Algorithmic Bayesian Epistemology

Eric Neyman

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy under the Executive Committee of the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2024
Abstract
Algorithmic Bayesian Epistemology
Eric Neyman

One aspect of the algorithmic lens in theoretical computer science is a view on other scientific disciplines that focuses on satisfactory solutions that adhere to real-world constraints, as opposed to solutions that would be optimal ignoring such constraints. The algorithmic lens has provided a unique and important perspective on many academic fields, including molecular biology, ecology, neuroscience, quantum physics, economics, and social science.

This thesis applies the algorithmic lens to Bayesian epistemology. Traditional Bayesian epistemology provides a comprehensive framework for how an individual’s beliefs should evolve upon receiving new information. However, these methods typically assume an exhaustive model of such information, including the correlation structure between different pieces of evidence. In reality, individuals might lack such an exhaustive model, while still needing to form beliefs. Beyond such informational constraints, an individual may be bounded by limited computation, or by limited communication with agents that have access to information, or by the strategic behavior of such agents. Even when these restrictions prevent the formation of a perfectly accurate belief, arriving at a reasonably accurate belief remains crucial. In this thesis, we establish fundamental possibility and impossibility results about belief formation under a variety of restrictions, and lay the groundwork for further exploration.
Table of Contents

Acknowledgments ................................................................. vii

Dedication .............................................................................. ix

Preface .................................................................................. 1

Chapter 1: Introduction .......................................................... 3
  1.1 What is algorithmic Bayesian epistemology? ......................... 3
  1.2 Prior work in algorithmic Bayesian epistemology .................... 7
  1.3 Key takeaways .................................................................. 18

Chapter 2: Preliminaries .......................................................... 20
  2.1 Proper scoring rules and the Bregman divergence .................. 20
  2.2 Forecast aggregation methods ........................................... 29
  2.3 Information structures ..................................................... 36
  2.4 Bayesian justifications for generalized linear and logarithmic pooling ........................................... 46

Chapter 3: Incentivizing precise forecasts .................................... 51
  3.1 Introduction ................................................................... 51
  3.2 Our model and preliminaries ............................................. 57
  3.3 An incentivization index ................................................... 61
9.4 Desiderata beyond linearity and respect for proofs .................. 246
9.5 Mechanistic anomaly detection for neural networks .............. 252
9.6 Conclusion ..................................................................... 255

Epilogue ............................................................................. 257

References ......................................................................... 261

Appendix A: Details omitted from Chapter 3 ......................... 275

Appendix B: Details omitted from Chapter 5 ......................... 318

Appendix C: Details omitted from Chapter 6 ......................... 337

Appendix D: Details omitted from Chapter 8 ......................... 351
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Thesis dependency structure</td>
<td>2</td>
</tr>
<tr>
<td>2.1</td>
<td>Savage representation of a proper scoring rule</td>
<td>25</td>
</tr>
<tr>
<td>2.2</td>
<td>Bregman divergence</td>
<td>26</td>
</tr>
<tr>
<td>2.3</td>
<td>Coarser estimates as orthogonal projections</td>
<td>43</td>
</tr>
<tr>
<td>3.1</td>
<td>Optimal scoring rules under different error measures (plot 1)</td>
<td>73</td>
</tr>
<tr>
<td>3.2</td>
<td>Optimal scoring rules under different error measures (plot 2)</td>
<td>74</td>
</tr>
<tr>
<td>3.3</td>
<td>Comparison of common proper scoring rules to optimal ones</td>
<td>75</td>
</tr>
<tr>
<td>3.4</td>
<td>Incentivization indices of scoring rules relative to optimal values</td>
<td>76</td>
</tr>
<tr>
<td>3.5</td>
<td>Incentivization indices relative to optimal values, continuous version</td>
<td>77</td>
</tr>
<tr>
<td>5.1</td>
<td>Quadratic score vs. log score: expert preference over the outcome</td>
<td>99</td>
</tr>
<tr>
<td>7.1</td>
<td>Robust aggregation: results summary</td>
<td>160</td>
</tr>
<tr>
<td>7.2</td>
<td>Approximation guarantees as a function of the number of experts</td>
<td>163</td>
</tr>
<tr>
<td>7.3</td>
<td>Suggested extremization factor as a function of the number of experts</td>
<td>187</td>
</tr>
<tr>
<td>9.1</td>
<td>Scatter plot of CNF acceptance probability estimates</td>
<td>233</td>
</tr>
</tbody>
</table>
List of Tables

3.1 Incentivization indices of various scoring rules and error measures . . . . . . . . . . 76

7.1 Approximation guarantees for small numbers of experts . . . . . . . . . . . . . . . 162

9.1 Analogy between deductive circuit estimation and formal proof verification . . . . 227

A.1 Simulation results for incentivization index accuracy . . . . . . . . . . . . . . . . 317
Acknowledgements

First, I would like to thank my Ph.D. advisor, Tim Roughgarden. Tim’s expansive knowledge of and insight into theoretical computer science and surrounding areas helped me to choose my research directions and understand my work in the context of others’ contributions. I benefited greatly from his impeccable advice on communication and presentation. Most of all, Tim was incredibly supportive in my exploration of my interests, both within and outside of grad school.

I would also like to thank my undergraduate mentor, Matt Weinberg. Matt’s class on economics and computing inspired me to do research with him. Matt was an incredible mentor, and the research we did together inspired me to go to grad school in theoretical computer science. Moreover, our project on proper scoring rules (Chapter 3) was instrumental in my decision to study algorithmic Bayesian epistemology in particular.

Thanks as well to all of my other research collaborators: Paul Christiano, Raf Frongillo, Jacob Hilton, George Noarov, Václav Rozhoň, Bo Waggoner, and Mark Xu. Working with them was great, and I have learned so much from them.

Thanks also to the National Science Foundation, whose graduate research fellowship program funded my research throughout my time as a grad student.

I would like to thank Scott Aaronson, whose blog post and paper on Aumann’s agreement theorem I found truly inspiring. Scott may have inspired not just Chapter 8 (which directly follows up on his paper), but this whole direction of my research.

I am so grateful to my family: my mom and my dad and my sister and my grandma. Their never-ending and unconditional love and support means so much to me.
And I am grateful to my longtime friends. Thank you to Mike and Cathy and Dylan and Sarah and Jenny and Ben and Yafah and Mia and Drake and Sam. I owe a great deal of my personal growth to the intellectual conversations and unforgettable adventures that they have shared with me.
To Baba Katya

— who gave me my first lessons on Aumann’s agreement theorem.
Preface

For me as for most students, college was a time of exploration. I took many classes, read many academic and non-academic works, and tried my hand at a few research projects. Early in graduate school, I noticed a strong commonality among the questions that I had found particularly fascinating: most of them involved reasoning about knowledge, information, or uncertainty under constraints. I decided that this cluster of problems would be my primary academic focus. I settled on calling the cluster *algorithmic Bayesian epistemology*: all of the questions I was thinking about involved applying the “algorithmic lens” of theoretical computer science to problems of Bayesian epistemology.

This thesis showcases my work on these problems. It starts with an introduction (Chapter 1), followed by technical preliminaries (Chapter 2). Chapters 3 through 9 describe some of my technical contributions. See Figure 1 for the dependence structure of the thesis: which chapters and sections are necessary for which others.

While not strictly necessary for the technical content, I recommend at least skimming the introduction, where I try to convey what exactly I mean by “algorithmic Bayesian epistemology” and why I’m excited about it. I also recommend reading Chapter 2 (Preliminaries), which is intended to be accessible to readers with a general college-level mathematical background.\(^1\) While the nominal purpose of Chapter 2 is to introduce the mathematical tools that we use in later chapters, the topics covered there are interesting in their own right.

Different readers will of course have different opinions about which technical chapters are the

---

\(^1\)Chapter 2 relies most heavily on basic probability theory.
most interesting. Each technical chapter begins with a short summary of the content, which may be useful for figuring out whether you want to read it. Naturally, I have my own opinions: I think the most interesting chapters are 5, 7, and 9, so if you are looking for direction, you may want to tiebreak toward reading those.

Figure 1: The thesis dependency structure. Solid arrows represent required background; dashed arrows represent recommended background.
Chapter 1: Introduction

The title and subject of this thesis is *algorithmic Bayesian epistemology*. This is an original term, and so in this introduction I will define, explain, and motivate it.

I also hope to convey my excitement about this topic. Problems in algorithmic Bayesian epistemology are theoretically fascinating and practically important. And yet, in many cases, they are surprisingly neglected! This makes the area ripe for exploration. In this introduction and the thesis in general, I will place significant emphasis on exploration: pointing out under-explored areas and asking questions about them. And so, after defining and motivating algorithmic Bayesian epistemology, I will give a whirlwind tour of existing work in the area, with an emphasis on the gaps in our understanding. In the later chapters, I will fill just a few of those many gaps. My primary hope for this thesis is that it will spur new research into directions that I leave uncovered: topics that I acknowledge but don’t explore, as well as questions that I didn’t even think to ask.

But before all that: what is algorithmic Bayesian epistemology?

1.1 What is algorithmic Bayesian epistemology?

Epistemology is the study of knowledge and uncertainty. Bayesian epistemology – named for Thomas Bayes – is a particular framework for studying knowledge that aims to understand uncertainty using the tools of probability. In Bayesian epistemology, an observer assigns probabilities to uncertain events (typically called *prior* probabilities) and updates those probabilities in light of new evidence (to get *posterior* probabilities). For example, suppose that the incidence of some disease in the general population is 0.1%. If I have no further information about whether I have the
disease,\(^1\) then in the framework of Bayesian epistemology I might say that there’s a 0.1% chance that I have the disease. Now, suppose that my doctor tests me for the disease, and that the test is ten times more likely to come up positive for people who have the disease than for people who don’t. If my test comes up positive, I will update my probability to (approximately) 1% in light of the new evidence.

Unfortunately, forming beliefs in the framework of Bayesian epistemology is often much more complex. Perhaps there is not one test for the disease, but five different tests. Suppose that I take all of the tests, some of which come up positive and others negative. Updating my probability in light of all those tests is simple enough if the tests constitute independent evidence – but they might not. Maybe Test B is more likely to be a false positive if Test A is a false positive, so treating them as independent would cause me to over-update my probability. Maybe Tests B and C test for different sub-types of the disease, such that they complement each other. If I knew all of the statistical relationships between the tests, then I could correctly infer the probability that I have the disease. But in practice, I might not have access to that data.

For another example, consider the task of predicting the weather. State-of-the-art weather forecasting is based on numerical weather prediction (NWP), a method that takes as input observational data about the current state of the atmosphere and solves differential equations in order to simulate future atmospheric conditions. The input data comes from millions of observations made by weather balloons, Doppler radars, satellites, buoys, aircraft, and human volunteers around the world. The forecasting models that implement NWP are incredibly complex, running on some of the world’s fastest supercomputers (NOAA, 2017). Technical breakthroughs have led to a substantial improvement in NWP models over the last three decades (Bauer et al., 2015).

Despite all that, NWP models are far from perfect: indeed, NWP forecasts beyond ten days perform worse than simply relying on historical averages (Seman et al., 2018). The accuracy of NWP forecasts is limited by informational and computational constraints. First, NWP models rely on observational data that is not very granular: for example, there are only about 1000 automated

\(^1\)Perhaps the disease is latent until old age, so my lack of symptoms does not constitute evidence against having the disease.
surface observing systems (ASOS) and fewer than 200 Doppler radar towers in the United States (NOAA, 2017). Second, even that data is not perfectly precise, as equipment may be miscalibrated. Third, even if perfectly accurate, extremely granular data were available, NWP would be severely limited by computational constraints.

In recent years, weather prediction methods based on machine learning have shown promise as an alternative to NWP methods. These methods involve training neural networks to achieve high predictive performance on past data. A recently-introduced machine learning-based method called GraphCast was shown to improve upon frontier NWP models at low resolution. GraphCast was also able to forecast severe weather events such as atmospheric rivers and tracks of tropical cyclones more accurately (Lam et al., 2023). While still outperformed by NWP models on high-resolution forecasts, machine learning-based models are significantly less computationally intensive and are likely to rapidly improve in the coming years.

Even if machine learning-based approaches come to mitigate the issue of computational constraints, weather forecasting will still be limited by informational constraints (imperfect and coarse-grained observational data). Furthermore, these approaches face new statistical challenges, such as overfitting, because they are based on learning from historical data rather than simulating physical laws. Finally, both NWP and machine learning-based approaches face questions of aggregation. Given several disagreeing weather forecasts, how does it make sense to combine them into a single forecast?

In light of all of these challenges, it makes sense to ask: how can we reason about uncertainty, in light of computational, informational, and other challenges that would-be perfect Bayesian reasoners face in practice? This is the domain of algorithmic Bayesian epistemology.

The relationship between Bayesian epistemology and algorithmic Bayesian epistemology is the same as the relationship between game theory and algorithmic game theory, and as the relationship between mechanism design and algorithmic mechanism design.

Mechanism design – traditionally a sub-discipline of economics – asks the question: how can
we design systems containing strategic agents pursuing their own incentives, in a way that produces good outcomes? For example, how can we auction off multiple items to multiple bidders in a way that produces the optimal social welfare for the bidders? The traditional answer from economic theory is the Vickrey-Clarke-Groves (VCG) auction, which elicits bids, computes the optimal allocation of items, and charges each bidder based on their externality on the remaining bidders.

Computer scientists find this answer dissatisfying, for a simple reason: computing the optimal allocation is not feasible, from the standpoint of both communication and computation. First, the bidders’ preferences may not be compactly representable, in which case it is infeasible to communicate them to the auctioneer. Second, even if the bidders’ preferences are compactly representable, actually computing the optimal allocation may still be intractable. And so algorithmic mechanism design asks the question: how can we design a *computationally and communicationally tractable* auction mechanism that attains a large fraction of the optimal social welfare (Nisan et al., 2007, §11)?

Algorithmic mechanism design belongs to a longstanding tradition in theoretical computer science: considering problems from other disciplines through an algorithmic lens. That is, instead of asking for the optimal solution to a problem, computer scientists ask: what is the best solution that *can actually be implemented*, given real-world (or real-world-inspired) constraints?

Sometimes, these constraints are computational: what is the best solution that can be found in polynomial time? Other times, the constraints are communicational: what is the best solution if parties are limited in how much they can communicate? Other kinds of constraints are also common. For example:

- Constraints on *information*. For example, the subfield of *online algorithms* studies sequential decision making under uncertainty (incomplete information). Often, the goal of an online

---

2See e.g. the case of single-minded bidders (Nisan et al., 2007, §11.2).

3The term *algorithmic lens* was coined at Berkeley by members of the Theory of Computing research group around the year 2000 (*private communication with Christos Papadimitriou*).
algorithm is to guarantee a result that is almost as good as the best possible result in hind-sight, e.g. the prophet inequality from optimal stopping theory (Krengel & Sucheston, 1987; Samuel-Cahn, 1984), or no-regret algorithms in online learning (Orabona, 2023).

- Constraints imposed by the strategic behavior of agents in a system. For example, many computer scientists study the price of anarchy: how much the welfare of a system degrades because of self-interested actors, as compared to a welfare-optimizing central planner (Koutsoupias & Papadimitriou, 1999; Roughgarden & Tardos, 2000).

The study of real-world problems through the algorithmic lens has significantly impacted a variety of disciplines, including molecular biology, ecology, neuroscience, quantum physics, and various social sciences – see Wigderson (2019, §20) for a detailed discussion.

And so, algorithmic Bayesian epistemology is simply the application of the algorithmic lens to the discipline of Bayesian epistemology. It is perhaps best to define algorithmic Bayesian epistemology by its examples, but to attempt a loose description:

> A question belongs to the field of algorithmic Bayesian epistemology (henceforth ABE) if it involves reasoning about uncertainty from a Bayesian perspective, but under constraints that prevent complete assimilation of all existing information.

### 1.2 Prior work in algorithmic Bayesian epistemology

To further motivate ABE, this section will give a whirlwind tour of prior work in ABE, with a focus on particularly neglected directions and gaps in our understanding. Since ABE involves reasoning about uncertainty under constraints, this section will be sub-categorized by type of constraint. In particular, we will highlight:

- Bayesian epistemology (BE) under computational constraints. This sub-topic is extremely important, but also relatively well-studied. Most of our thesis focuses on more neglected parts of ABE, so our summary of prior work on BE under computational constraints will be brief. Computational constraints will be most relevant for Chapter 9.
• BE under informational constraints. This sub-topic is important, relatively neglected, and (in my opinion) extremely interesting. Chapters 6, 7, and 9 will all focus on BE under informational constraints.

• BE under communication constraints. This sub-topic is the most neglected of the four, despite having many interesting questions. It will be the focus of Chapter 8.

• BE under constraints imposed by the strategic behavior of experts. This sub-topic is reasonably well-studied, though – I would argue – not sufficiently well-studied relative to its importance. Chapters 3, 4, and 5 will all focus on this topic.

• BE under a combination of the types of constraints listed above.

1.2.1 Computational constraints

Approximating Bayesian inference  Exact Bayesian inference – that is, incorporating evidence in order to exactly compute a posterior probability – is computationally intractable because it requires taking a sum or integral over a high-dimensional space (one dimension per category of evidence). As a result, there has been extensive research into developing methods for computationally efficient approximate Bayesian inference. Classical results of the field include Markov Chain Monte Carlo (MCMC) methods, such as the Metropolis-Hastings algorithm, approximate Bayesian computation (ABC) methods, and hierarchical Bayesian methods. Such methods are used extensively in many disciplines, such as physics, biology, machine learning, and public health. There is a vast literature on approximate Bayesian inference – see Gelman et al. (2013) for a thorough exposition – but it is not a focus of this thesis.

Bounded rationality  While theoretical work in economics generally assumes agents to be rational – that is, to choose the best action in their situation given their knowledge – behaving fully rationally may involve solving computationally intractable problems. For example, Chen et al. (2009) showed that computing a Nash equilibrium in a game is PPAD-hard. Additionally, empirical studies show that in a variety of settings, people do not behave rationally (Conlisk, 1996).
This has led to the study of *bounded rationality*: formal models in which agents do not necessarily take the optimal action. There are many models of bounded rationality, including satisficing agents and evolutionary economics; see Conlisk (1996) for an overview. Perhaps the one that fits best within the purview of ABE is a model of *level-k rationality* formulated by Nagel (1995). In this model, a level 0 agent takes a random action; a level 1 agent best responds under the assumption that all other agents are level 0 agents; a level 2 agent best responds under the assumption that all other agents are level 1 agents; and so on. In a sense, each agent is computing their optimal strategy, but stopping after \( k \) steps.\(^4\) Camerer et al. (2004) extended this model to allow agents to best-respond under a hypothesized distribution of the levels of other agents, instead of assuming that all agents are one level lower.

### 1.2.2 Informational constraints

**Forecast aggregation under incomplete information** Suppose that an aggregator receives precipitation forecasts from several different experts. How should the aggregator combine those forecasts into a single number? If the aggregator has a perfect understanding of the likelihood of various possible states of the world and what information each expert has in every world state, then the aggregator can deduce the correct aggregate forecast (given unlimited computation). In practice, however, the aggregator does not have this knowledge and must aggregate the forecasts under incomplete information. Forecast aggregation under incomplete information will be the focus of Chapters 6 and 7.

One class of approaches to this problem is called *Bayesian* forecast aggregation: the aggregator makes some modeling assumptions about the experts’ information and deduces the correct posterior within that model. If the aggregator is tasked with combining probability distributions over a set of outcomes, then perhaps the simplest model would assume that the experts’ forecasts are independent conditional on the outcome. In this model, applying Bayes’ rule straightforwardly gives an answer. If instead experts are asked to forecast the expectation of a real number \( Y \) (such as

\(^4\)Note, however, that the limit as \( k \) approaches infinity of the strategy of a level \( k \) agent is not necessarily well-defined; the strategy may end up cycling among a list of options.
the amount of rainfall), then one could choose to model each expert as receiving a noisy estimate of $Y$ from some distribution; see Winkler (1981) for an early work with this flavor. More recently, Frongillo et al. (2015) considered a setting in which a principal wishes to learn the distribution of a random variable (which is modeled as belonging to a particular parameterized family of distributions) based on information provided by experts, each of whom sees some number of samples from the distribution. Lichtendahl Jr. et al. (2022) consider aggregation in a model in which experts share some information and additionally receive private i.i.d. samples from an exponential family. See Lichtendahl Jr. (2011) for an introduction to Bayesian forecast aggregation.

Another class of approaches is called axiomatic forecast aggregation: the goal of such approaches is to describe a set of axioms that an aggregation method ought to satisfy, and then to characterize methods that satisfy those axioms. Dietrich and List (2016) describe several such axioms, including unanimity preservation, eventwise independence, and external Bayesianality. No pooling method satisfies all three of these axioms: only linear pooling (i.e. a weighted average of the experts’ forecasts) satisfies the first two axioms (Aczél & Wagner, 1980), but it does not satisfy the third axiom. On the other hand, Genest (1984) showed that another natural aggregation method called logarithmic pooling satisfies the third axiom. In addition to satisfying natural axioms, linear and logarithmic pooling are the two most well-studied forecast aggregation methods, and they will come up many times in this thesis.

A third class of approaches – the one most relevant to this thesis – is called robust forecast aggregation. Because the aggregator lacks complete information about the information structure describing the experts’ knowledge, it makes sense to seek an aggregation method that performs well in the worst case over a wide class of information structures (see Section 2.3 for an introduction to information structures). For example, Arieli et al. (2018) find the aggregation method that performs best for the class of information structures of two Blackwell-ordered experts and (separately) for the class of information structures with two conditionally independent experts. Levy and Razin (2020) take a different approach, analyzing robust aggregation methods under bounds on the correlation between experts’ signals. Oliveira et al. (2021) take a similar robust approach.
to decision problems more generally. Overall, there has been relatively little prior work in this
direction, a gap that this thesis aims to fill.

**Online learning from expert advice**  In online learning, a decision-maker must choose from a
fixed range of options for each of $T$ time steps. On each time step, after choosing, the decision-
maker receives a reward that depends on their chosen option, and also learns the reward they would
have received under every other choice they could have made. The decision-maker’s goal is to get
a total reward that is almost as high as if they had chosen the fixed option that had the highest total
reward over the $T$ time steps. This is an example of an algorithmic problem under informational
constraints: the constraint here is that the decision-maker knows nothing about the quality of each
option beforehand, yet must do nearly as well as if they had known the best fixed option (on average
over the time steps). Online learning is a well-studied field; see Orabona (2023) for an overview.

We are interested in the subproblem of online prediction from expert advice. At each time step,
$m$ experts forecast probability distributions over $n$ possible outcomes. The decision-maker then
chooses their own probability distribution over the outcomes. Afterward, an outcome is realized
and the decision-maker is given a reward based on their distribution and the outcome (generally
using a proper scoring rule – see Section 1.2.4). The decision-maker aims to compete with the
best expert in hindsight, or perhaps even with the best possible mixture (e.g. weighted average)
of experts in hindsight. See Cesa-Bianchi and Lugosi (2006) for a survey of this topic. We will
explore online prediction from expert advice in Chapter 6.

**Estimation theory**  In statistics, estimation theory is the study of estimating the parameters of
a distribution given samples from the distribution. A simple example is estimating the mean of
a normal distribution with variance 1 based on samples. Given a prior over the true mean, one
can apply Bayes’ rule to exactly compute the expected value of the mean, using the sample mean
as evidence. However, estimation theory typically concerns itself with estimating parameters in
settings in which the prior is not known. One common approach is the minimax estimator, which
is the estimator with the best performance (e.g. as measured by expected squared error) in the worst
case over possible priors (or equivalently, in the worst case over parameters of the distribution). An alternative framing on minimax estimation is one of adversarial robustness: the minimax estimator is the optimal estimator if the prior over parameters is chosen by an adversary. Minimax estimators are thus a classic example of Bayesian epistemology under informational constraints: while not Bayesian in the traditional sense (they do not assume a prior), they can be thought of as the correct Bayesian estimate in the context of an adversarially chosen prior.

The minimax estimator is one example of a natural approach to estimation when lacking a prior over distribution parameters. Other approaches include unbiased minimum-variance estimators and maximum likelihood estimators. See Kay (1993) for a thorough treatment of the topic.

**Reasoning about Uncertainty**  *Reasoning about Uncertainty* (Halpern, 2003) is a book on uncertainty in the context of different notions of probability that depart from the standard one. For example, how might one reason about uncertainty if one is only able to assign probabilities to some subsets of possible events but not other subsets? How should we deal with probabilities that are defined under axioms that are weaker than the standard ones? These perspectives are more distantly related to the topics of this thesis than most topics covered in this introduction.

### 1.2.3 Communication constraints

**Agreement protocols**  There has been little work on Bayesian epistemology under communication constraints, but one stand-out example is Aaronson (2005). Aumann (1976) famously showed that two individuals with a common prior (but possibly different private information) cannot “agree to disagree”. To be more precise, suppose that there is “common knowledge” that Alice’s estimate of the chance of rain is $p$, meaning that Alice’s estimate is $p$, Bob’s estimate of Alice’s estimate is $p$, Alice’s estimate of Bob’s estimate of Alice’s estimate is $p$, and so on. And suppose that there is common knowledge that Bob’s estimate for the chance of rain is $q$. Then it must be that $p = q$. This result suggests that if Alice and Bob disagree, then they ought to be able to exchange information to reach agreement.
In this model, Alice and Bob can reach agreement by exchanging all of their information; however, this might require a prohibitively large amount of communication. Aaronson (2005) showed that simply by repeatedly exchanging their estimates (Alice shares her estimate; Bob shares his estimate after updating on Alice’s estimate; and so on), Alice and Bob will quickly reach near-agreement. This leaves many questions open: for example, under what circumstances can Alice and Bob reach agreement in a computationally efficient way? Or, under what circumstances is the agreed-upon estimate approximately correct? We address this second question in Chapter 8.

1.2.4 Strategic constraints

Information elicitation  So far we have focused on how an individual with access to information can form beliefs. However, in some settings, the individual must first learn the information, e.g. by eliciting it from experts. If the experts are self-interested, this can raise strategic questions. The most basic question is one of truthful elicitation: how can you pay an expert for information in a way that incentivizes the expert to tell the truth?

For example, suppose that you wish to elicit the probability of rain from a meteorologist. To incentivize the meteorologist, you decide to pay them as a function of the forecast they give you and whether or not it ends up raining. Such a function is called a scoring rule. Scoring rules must be chosen carefully. For instance, here is one natural (but poor) choice of scoring rule: if the meteorologist reports a probability \( p \) of rain, you will pay them \( p \) if it rains and \( 1 - p \) if it doesn’t. (That is, you reward the meteorologist proportionally to the probability that they assigned to the eventual outcome.) This scoring rule incentivizes the meteorologist to report a probability of 100% if their true belief is anywhere greater than 50% and to report 0% if their true belief is anywhere less than 50%.

A scoring rule is called proper if the optimal strategy of an expert who wishes to maximize their expected score is to report their true belief. One commonly used proper scoring rules is the quadratic score, which penalizes an expert according to the squared difference between their report and the correct answer (either 0 or 1 depending on the outcome). Another is the logarithmic score,
which is rewards the expert with the logarithm of the probability that the expert assigns to the eventual outcome. We provide an exposition to proper scoring rules in Section 2.1; see Gneiting and Raftery (2007) for a more thorough survey.

Similarly, one can aim to truthfully elicit properties of probability distributions. That is, suppose that you wish to know how much it will rain tomorrow. The meteorologist has a probability distribution over tomorrow’s rainfall, and you wish to know the mean of the distribution. If you ask the meteorologist for this mean, how do you pay them (as a function of their report and the eventual rainfall amount – i.e. a draw from the distribution) to truthfully elicit their belief? What if you want to know the median of the distribution, rather than the mean? What about the second moment? The variance? It turns out that the variance is not straightforwardly elicitable – no reward function will incentivize a truthful report – though you can compute the variance by separately eliciting the mean and the second moment (both of which are straightforwardly elicitable). The foundations of property elicitation were laid by Osband (1985) and Savage (1971). See Abernethy and Frongillo (2012), Frongillo and Kash (2020), Gneiting (2011), and Lambert, Pennock, and Shoham (2008) for more recent work in this area. Overall, most research on property elicitation is fairly recent and a lot of low-hanging fruit remains.

Recently, there has been interest in contract functions, which are scoring rules for multiple experts. Chun and Shachter (2011) showed that simply using a proper scoring rule for each expert invites collusion between experts. They asked whether contract functions that do not allow for collusion are possible. We resolve this question in Chapter 4 by exhibiting such a contract function. However, interesting questions remain: there are natural definitions of collusion that are broader than the one given by Chun and Shachter (2011), and we leave open the question of whether any contract function disallows collusion under these broader definitions.

There has also been recent work on wagering mechanisms, introduced by Lambert, Langford, et al. (2008): contract functions that additionally elicit a wager from each expert and redistribute the wagers according to expert performance. The authors defined several desiderata for wagering mechanisms and identified the unique wagering mechanism that satisfied those desiderata. Subse-
quentely, Chen et al. (2014) explored wagering mechanisms that do not allow collusion (by analogy to the same question for contract functions). More recently, Freeman et al. (2017) noted that wagering mechanisms can be thought of as experts trading securities; recent work on wagering mechanisms has often taken advantage of this framing. While this area is not a focus of the thesis, I see wagering mechanisms as a particularly fruitful direction for future work. We further discuss wagering mechanisms in the epilogue.

**Prediction markets** Prediction markets are a solution concept to the dual problems of forecast elicitation and forecast aggregation. In a prediction market, experts express their beliefs about the probability of some event (or the value of some unknown quantity) by making profit-maximizing trades. When an expert has private information, they can trade in the market, thus integrating their information into the consensus view. The most well-studied prediction markets in the computer science literature (but not the most common markets in practice) are *market scoring rules* (MSRs), introduced by Hanson (2003). An MSR is based on a proper scoring rule: the experts report their beliefs in sequence, and – once it is known whether the event happened – are their score minus the previous expert’s score. Chen and Pennock (2007) showed that MSRs can be thought of in terms of experts trading Arrow-Debreu securities (contracts that are worth 1 if the event happens and 0 otherwise). See Abernethy et al. (2013) for follow-up work in this vein.

One downside of MSRs, which perhaps accounts for the lack of use of MSRs in practice, is that market liquidity does not increase with total trading volume. Abernethy et al. (2014) define *volume-parameterized markets*, an MSR-inspired mechanism that aims to fix this issue.

Much work on MSRs and related mechanisms assumes that experts are myopic: that they maximize their expected reward from their current report or trade, without regard to potential future trades. Several recent papers explore conditions under which MSRs incentivize experts to reveal information even when they are not myopic (Anunrojwong et al., 2019; Chen & Waggoner, 2016; Chen et al., 2010; Kong & Schoenebeck, 2018).

An additional line of work explores prediction markets in the context of experts with mutable
beliefs, i.e. beliefs that may change in light of information revealed by the market (Chen et al., 2012; Ostrovsky, 2012). This is, of course, how experts’ beliefs behave in practice. Exploring the dynamics of prediction markets in which participants have mutable beliefs seems like a particularly neglected direction relative to its importance.

**Information design** So far we have taken the perspective of an individual who wishes to elicit information from strategic agents. We can also take the perspective of a strategic agent who is interested in giving away partial information so as to influence decision-makers. This topic is called information design. In a sense, information design is the mechanism design of ABE: while economic theory focuses on the behavior of strategic agents, mechanism design asks how to design a strategic landscape that is favorable to the designer. Similarly, information design asks how a favorable informational landscape can be created.

**Bayesian persuasion** is the sub-case of information design that involves a single information sender and a single receiver (Kamenica & Gentzkow, 2011). The authors an example of Bayesian persuasion in the context of a prosecutor (the sender) and a judge (the receiver). The judge is tasked with determining the innocence or guilt of 100 defendants, and knows that exactly 30 of the defendants are guilty (but doesn’t know which ones). The judge’s utility function is taken to be +1 for every correctly-classified defendant. The prosecutor knows which defendant are guilty, and their utility is +1 for every defendant whom the judge convicts. If the prosecutor reveals all of their information, then the judge will convict the 30 guilty defendants. However, suppose that the prosecutor randomly chooses 29 innocent defendants in addition to the 30 guilty ones and tells the judge that 30 of the 59 selected defendants are guilty. Then the judge will convict all 59 of them, which is a better outcome for the prosecutor. See Dughmi and Xu (2021) for a survey of Bayesian persuasion.

Information design more generally concerns itself with a sender giving partial information to multiple receivers, where the sender’s utility function depends on the receivers’ actions. Additionally, the sender may not know the receivers’ utility functions (although the sender has a probability
distribution over the utility functions). See Bergemann and Morris (2019) for a survey of recent work in information design.

1.2.5 Multiple kinds of constraints

Robust mechanism design While much of auction theory focuses on buyers with independent valuations, in practice a buyer’s valuation may be informed by other buyers’ valuations. For example, consider an auction for drilling rights in an oil field, where different companies (buyers) each have private information about the amount of oil in the field. Or consider a used car auction in which different participants have different impressions of the reliability of each car. The study of interdependent value auctions goes back to Wilson (1969), who studied common value auctions: auctions in which the item’s true value is the same for every participant (but is not known for sure by the participants). Myerson (1981) studied a “weighted sum” setting, in which each buyer’s value is equal to their private signal plus some constant \( \beta \in [0, 1] \) times the sum of all other buyers’ signals.

A seller may wish to create an auction that maximizes the buyers’ welfare or the seller’s revenue. The optimal auction depends heavily on the (potentially extremely complex) joint probability distribution over all buyers’ signals and values. Studying auctions in such a complex setting has often led to theoretical results that work poorly in practice or are too complex to implement (Wilson, 1987). This has led to work in robust mechanism design, which seeks to relax the assumption that the seller knows the probability distribution over signals and values, instead aiming for mechanisms that have strong guarantees for both incentive compatibility and social welfare (or revenue) under weaker assumptions (Bergemann & Morris, 2013). See Roughgarden and Talgam-Cohen (2019) for a survey of this area. Such work is an example of Bayesian epistemology under simultaneous computational, informational, and strategic constraints.

Incentivizing prediction without ground truth access Our discussion of information elicitation in Section 1.2.4 (e.g. asking a meteorologist how much it will rain tomorrow) crucially relied on
access to the eventual outcome (knowing how much it ended up raining). However, sometimes we wish to elicit forecasts for far-future events ("How much will the Earth’s average temperature increase over the 21st century?"). Prelec (2004) introduced a mechanism called *Bayesian truth serum* for eliciting forecasts in the absence of a knowable ground truth. The basic idea is to elicit from each expert both a forecast and a prediction of other experts’ forecasts, and to reward each forecaster both for accurately predicting others’ forecasts and for having a “surprisingly popular” forecast: a forecast whose frequency among the population of experts was underestimated by other experts. The key idea is that a Bayesian expert should predict that others will underestimate the frequency of that expert’s view. Prelec (2004) showed that truth-telling is a Nash equilibrium in the Bayesian truth serum mechanism.

The *peer prediction* mechanism (Miller et al., 2005) is designed for a similar setting, in which e.g. an instructor does not have time to grade all students’ homework, and so assigns students to grade each other’s homework. Truth-telling is a Nash equilibrium of peer prediction, though under somewhat stricter modeling assumptions. Both of these mechanisms are designed for the elicitation of information under both strategic constraints and computational/informational constraints.

### 1.3 Key takeaways

Now that I have introduced the concept of algorithmic Bayesian epistemology and given a variety of examples, I will conclude with some overall takeaways.

**ABE is important**  The ability to form accurate beliefs is self-evidently important. Often, there are obstacles to doing so: maybe you don’t have the computational resources to do so perfectly. Or maybe your data is biased but you don’t know exactly how. Or maybe you’re trying to aggregate estimates from different sources, but you don’t know whether your sources’ estimates are based on disjoint or overlapping information. Or maybe your information comes from market prices determined by strategic agents. Or maybe you want to incorporate the beliefs of someone who has approached the same question from an unfamiliar perspective. Coming up with solutions in the
face of these challenges is the domain of ABE.

**ABE is often neglected**  Consider the problem of how best to aggregate different forecasts or estimates for a quantity. This question is ubiquitous: it comes up in essentially every branch of science. Yet until quite recently, there was very little empirical work on this question and even less theoretical work. Or consider the problem of forming beliefs under communication constraints: a well-motivated question with very little theoretical work. Proper scoring rules have been relatively well-studied, yet the natural question of *which* proper scoring rule should be used in a given situation has not received much attention. Of the large number of important and well-motivated questions in ABE, relatively few have received a lot of attention. As a consequence, work in ABE is often relatively tractable: much of the low-hanging fruit is left to be picked.

**ABE is really interesting**  My opinion on this is, of course, subjective. But the question of how to form beliefs is fundamental, and the question of how to form beliefs under constraints seems like a fundamental sub-problem. Also, the mathematical notions that arise from the formal study of this problem are – at least in my opinion – very elegant.

In the coming chapters, we will look at just a few of the many interesting and well-motivated questions in ABE. Let’s begin!
Chapter 2: Preliminaries

While each chapter of this thesis will have its own introduction, some preliminary concepts in ABE will find use throughout the thesis. This chapter introduces those concepts.

2.1 Proper scoring rules and the Bregman divergence

This section is necessary for Chapters 3, 4, 5, and 8, and is also somewhat useful for Chapter 6.

2.1.1 Introduction to proper scoring rules

Suppose that I want to know whether it will be sunny, cloudy, or rainy tomorrow. I don’t know much but weather forecasting, but my friend Skylar does. It’s natural for me to ask her for a forecast probability distribution over the options {sunny, cloudy, rainy}. If I want to give Skylar an incentive to give me a well-reasoned probability distribution that reflects her true beliefs, then I might promise to pay her according to the accuracy of her forecast. Put another way, I could ask for her forecast, wait until tomorrow, and pay her according to “how right she was” – that is, according to how close her forecast was to the ideal forecast (putting 100% on whichever outcome actually ends up happening).

Any payment scheme that’s a function of Skylar’s forecast and the eventual outcome is called a scoring rule. For example, one scoring rule gives Skylar a payment (score) equal to the probability that she assigned to the eventual outcome. For example, if Skylar’s forecast is (70% sunny, 20% cloudy, 10% rainy), and it ends up being sunny, she would receive a score of 0.7.

Does this scoring rule “make sense”? Imagine that Skylar’s actual belief is (70% sunny, 20% cloudy, 10% rainy). If Skylar is interested in maximizing the expected value of her score, does it
make sense for Skylar to tell the truth?

If Skylar tells the truth, then with probability 70%, it will be sunny and she will receive a score of 0.7; with probability 20%, it will be cloudy and she will receive a score of 0.2; and with probability 10%, it will be rainy and she will receive a score of 0.1. Thus, the expected value of her score is

\[ 70\% \times 0.7 + 20\% \times 0.2 + 10\% \times 0.1 = 0.49 + 0.04 + 0.01 = 0.54. \]

Now, by contrast, suppose that Skylar lies and reports that there’s a 100% chance that it will be sunny tomorrow. If she does this, then her score will be 1 if it’s sunny and 0 if it’s not. Overall, this makes Skylar come out ahead: her expected score is now 70\%(1) = 0.7 > 0.54. Thus, this scoring rule gives Skylar an incentive to lie, which is clearly an undesirable property.

Are there any scoring rules that incentivize Skylar to report her true belief? As we will see, the answer is yes. Any such scoring rule is called a proper scoring rule. To talk precisely about proper scoring rules, let’s introduce some notation. We will be using this notation through the thesis.

- We will use the term principal to refer to the person eliciting (asking for) the forecast, and we use the term expert to refer to the person reporting their forecast. In the above example, I am the principal and Skylar is the expert.
- We will let \( n \) be the number of possible outcomes (above, \( n = 3 \)). We will number the outcomes 1 through \( n \) and use the letter \( j \) to denote any particular outcome. Thus, \( j \in \{1, \ldots, n\} \). (In the future, we will write \([n]\) in place of \(\{1, \ldots, n\}\) for brevity.)
- As is standard notation, we will let \( \Delta_n \) be the space of all probability distributions over \( n \) outcomes. We can think of a probability distribution over \( n \) outcomes as a vector of \( n \) probabilities: non-negative numbers that add to 1. In other words, \( \Delta_n = \{(x_1, \ldots, x_n) \in \mathbb{R}^n : x_j \geq 0 \ \forall j, \sum_j x_j = 1\} \).
- Typically, we will use \( x = (x_1, \ldots, x_n) \) to denote the expert’s report (i.e. the probability distribution that the expert reports to the principal) and will use \( p = (p_1, \ldots, p_n) \) to denote the expert’s true belief. Both \( x \) and \( p \) belong to \( \Delta \).
• We will use the notation \( s(x; j) \) for the expert’s score, a function of the expert’s report \( x \) and the eventual outcome \( j \). Scores are allowed to be negative (or even negative infinity).

What does it mean for a scoring rule \( s \) to truthfully elicit an expert’s belief? If an expert’s true belief is \( p \) and the expert reports \( x \), then their expected score is \( \mathbb{E}_{j \sim p} [s(x; j)] \). (Here, \( j \sim p \) means that \( j \in [n] \) is selected at random according to the distribution \( p \).) The expert is incentivized to report the \( x \) that maximizes this expected value. Therefore:

**Definition 2.1.1.** A scoring rule \( s \) for \( n \) outcomes is proper if for all \( p \in \Delta_n \), the expression \( \mathbb{E}_{j \sim p} [s(x; j)] \) has a unique maximum at \( x = p \).

(Note that many sources would call a scoring rule “proper” even if the maximum at \( x = p \) is not unique, and would use the term “strictly proper” where we say “proper.” Under this usage, the scoring rule \( s(x; j) = 0 \) (which gives the expert reward 0 no matter what) would be considered proper. Generally, we will only be interested in scoring rules that satisfy our stronger definition. When necessary, we will use the term *weakly proper* for the weaker notion.)

There are many proper scoring rules (we characterize them all below), two of which are very well-known. The most well-known is the *quadratic* scoring rule:

**Definition 2.1.2.** The quadratic scoring rule, also known as the Brier score, is defined by

\[
s_{quad}(x; j) := -(1 - x_j)^2 - \sum_{j' \neq j} x_{j'}^2.
\]

Put otherwise, if \( \delta_j \) is the vector with a 1 in position \( j \) and 0 elsewhere, then \( s_{quad}(x; j) \) is equal to \(-\|x - \delta_j\|_2^2\).

The quadratic scoring rule is best thought of as a penalty on the expert equal to the squared distance between their forecast (\( x \)) and the “perfect” forecast \( \delta_j \). This scoring rule is proper. (Note that the quadratic score can be made nonnegative by adding 2 to it, if it’s desirable for the expert’s score to be guaranteed to be nonnegative.)

The second most well-known proper scoring rule is the *logarithmic* scoring rule:
Definition 2.1.3. The logarithmic scoring rule (often, colloquially, the “log score”) is defined by

\[ s_{\log}(x; j) := \ln(x_j). \]

Instead of rewarding the expert with the probability assigned to the eventual outcome (which we saw earlier to be improper), the log score rewards the expert with the log of that probability. Note that, unlike the quadratic score, the log score only depends on the probability that the expert assigns to the eventual outcome. It turns out the the log score is the only proper scoring rule with this property (up to adding and multiplying by constants) (Parry et al., 2012).

Unlike the quadratic score, the log score harshly penalizes an expert for assigning an extremely low probability to the eventual outcome (and gives a score of \(-\infty\) to an expert who assigns probability zero to the eventual outcome). It makes sense to use the log scoring rule if the principal cares about differentiating between very low and extremely low probabilities. This is because the log scoring rule incentivizes the expert to think carefully about assigning very low probabilities to possible outcomes, and to make sure that their forecast is not overconfident.

Remark 2.1.4. Any positive affine transformation of a proper scoring rule is proper. That is, for any \(a > 0\) and \(b \in \mathbb{R}\), if \(s\) is proper then so is \(as + b\).

2.1.2 The Savage representation

Perhaps the best way to think of proper scoring rules is in terms of their Savage representation, named after mathematician Leonard Savage. To introduce the Savage representation, we will first define the expected score function of a proper scoring rule:

Definition 2.1.5. The expected score function \(G_s : \Delta_n \to \mathbb{R}\) of a proper scoring rule \(s\) over \(n\) outcomes is defined by

\[ G_s(x) := \sum_{j \in [n]} x_js(x; j). \]

\(G_s\) is thus the expected score of an honest expert, as a function of their belief. For example:
• If $s$ is $s_{\text{quad}}$, then $G_s(x) = \sum_j x_j^2 - 1 = \|x\|^2 - 1$. Note that $\|x\|^2$ is large for probability distributions that are more concentrated on particular outcomes, so an expert’s expected score is higher if they are more certain about which outcome will happen. This should be intuitive: more informed experts should have higher expected scores.

• If $s$ is $s_{\text{log}}$, then $G_s(x) = \sum_j x_j \ln(x_j)$. Note that this is exactly the negative of the Shannon entropy of the probability distribution $x$. This means that an expert’s expected score is higher for lower-entropy distributions, i.e. when the expert is more certain about the outcome.

Both of these expected score functions are convex. This is not a coincidence:

**Theorem 2.1.6** (Savage, 1971). The expected score function of a proper scoring rule is strictly convex. Given a strictly convex function $G : \Delta_n \to \mathbb{R}$, consider a function $s(x; j)$ as follows: at each $x \in \Delta_n$, draw a tangent plane to $G$ at $x$, and let $s(x; j)$ be the value of this tangent plane at $\delta_j$. (Algebraically, $s(x; j) = G(x) + \langle \delta_j - x, \nabla G(x) \rangle$, where $\nabla G$ is the gradient\(^1\) of $G$ and $\langle \cdot, \cdot \rangle$ is the dot product.) Then $s$ is a proper scoring rule with expected score function $G$ – and in fact is the only\(^2\) proper scoring rule with expected score function $G$.

It follows from Theorem 2.1.6 that for any proper scoring rule $s$, we can rewrite $s$ in terms of its expected score function $G_s$:

$$s(x; j) = G_s(x) + \langle \delta_j - x, \nabla G_s(x) \rangle.$$ \hspace{1cm} (2.1)

This form is known as the **Savage representation** of $s$.

In the case of $n = 2$ – i.e. the setting in which the principal wishes to elicit the probability of a single yes/no outcome – we can easily visualize the Savage representation. In this setting, we can identify any report with the probability assigned to the “Yes” outcome. In Figure 2.1, $G$ is the expected score function of some proper scoring rule $s$, and the proper scoring rule can be recovered from $G$ by drawing a tangent line at the report $x$ and noticing where it intersects 1 (that’s

\footnote{\(\text{Or a subgradient, if } G \text{ is not differentiable.}\)}

\footnote{\(\text{Unless } G \text{ is not differentiable, in which case multiple tangent planes can exist at a point.}\)}
the score if the “Yes” outcome happens) and where it intersects 0 (that’s the score if the “No” outcome happens).

Figure 2.1: For the proper scoring rule derived from the function $G$ shown here, suppose that an expert reports a probability $x = 0.4$ on the “Yes” outcome. The expert will receive a score equal to the $y$-value of the red point on the right (roughly 0.3) if the “Yes” outcome happens, and a score equal to the $y$-value of the red point on the left (roughly 0.7) if the “No” outcome happens.

It is helpful to have some intuition for Theorem 2.1.6. First, why is $G_s$ strictly convex? Suppose for contradiction that there are two forecasts $p$ and $q$, and a constant $0 < \alpha < 1$, such that $G_s(\alpha p + (1 - \alpha)q) \geq \alpha G_s(p) + (1 - \alpha)G_s(q)$. Then it would follow that either an expert with belief $p$ would be at least as well off reporting $\alpha p + (1 - \alpha)q$, or that an expert with belief $q$ would be at least as well off reporting $\alpha p + (1 - \alpha)q$. This can be verified algebraically by expanding out the definition of $G_s$.

Second, given a strictly convex $G$, why is the scoring rule $s(x; j) = G(x) + \langle \delta_j - x, \nabla G(x) \rangle$ proper? While this fact can be verified algebraically, there is also an intuitive geometric proof. We give this proof below, but to do so we will first introduce concept of a Bregman divergence.

### 2.1.3 The Bregman divergence

The Bregman divergence is a notion of distance that is defined with respect to a convex function.

**Definition 2.1.7.** For some $n \geq 1$, let $D$ be a convex subset of $\mathbb{R}^n$, and let $G : D \to \mathbb{R}$ be a differentiable, convex function. Given $x, y \in D$, the Bregman divergence from $y$ to $x$ with respect
to $G$ is defined as
\[
D_G(y \parallel x) := G(y) - G(x) - \langle y - x, \nabla G(x) \rangle.
\] (2.2)

While this formal definition is algebraic, the Bregman divergence is fundamentally a geometric object that can be understood without parsing the formal definition. The Bregman divergence from $y$ to $x$ has the following interpretation. Draw the tangent plane to $G$ at $x$; the Bregman divergence is the vertical distance between the plane and the function $G$ at $y$. We show an example (in one dimension) in Figure 2.2.

![Figure 2.2: The vertical distance shown is the Bregman divergence from $y$ to $x$ with respect to $G$, and is denoted $D_G(y \parallel x)$.](image)

Note that the Bregman divergence is not symmetric: $D_G(y \parallel x)$ does not in general equal $D_G(x \parallel y)$. (We discuss the order of arguments more below.) The one exception (up to positive affine transformations) is the function $G(x) = \|x\|^2$ (or, if in one dimension, $G(x) = x^2$). In that case, it can be verified that $D_G(y \parallel x) = \|y - x\|^2$, the squared distance between $x$ and $y$.

The Bregman divergence with respect to $G(x) = \|x\|^2$ is thus a very natural notion of distance. A different natural notion of distance is the Bregman divergence with respect to the negative of Shannon entropy (i.e. $\sum_j x_j \ln(x_j)$), which came up earlier as the expected score function of the log scoring rule. For this $G$, $D_G(y \parallel x)$ is the KL divergence $D_{\text{KL}}(y \parallel x)$, which is ubiquitous in information theory and statistics.

Now let’s return to Theorem 2.1.6, which claimed that if a function $G$ is strictly convex, then the scoring rule $s(x; j)$ defined as the value of the tangent plane to $G$ at $x$, evaluated at $\delta_j$, is proper.
Why is this true?

Suppose that an expert’s true belief is $y$. Then by reporting $y$, they will have expected score $G(y)$. Now, what if, instead, the expert reports some $x \neq y$? In that case, the expected value of their score is equal to the weighted average of the expert’s possible scores depending on the outcome, with weights given by $y$. This is just the value of the tangent plane to $G$ at $x$, evaluated at $y$. (For example, if $x$ and $y$ are as in Figure 2.2, then the expected score of an expert who believes $y$ but reports $x$ is the height of the point on the blue line below the point labeled $y$.)

Because $G$ is strictly convex, $G(y)$ is strictly larger than this value. In fact, it is larger by exactly $D_G(y \parallel x)$. Therefore, the expert is best off reporting their true belief $y$.

Note that this gives us some interpretations of the Bregman divergence in the context of proper scoring rules. For the points below, let $s$ be a proper scoring rule and $G$ be its expected score function, and consider an expert who is rewarded according to $s$.

- The Bregman divergence $D_G(y \parallel x)$ is the expected amount that the expert loses by lying and reporting $x$, if their true belief is $y$.
- Alternatively, if there is a “true” probability distribution $y$, but the expert mistakenly has belief $x$, then $D_G(y \parallel x)$ is the expected amount that the expert loses by reporting $x$, as compared to reporting $y$.

  - Note that Equations 2.1 and 2.2 together tell us that for a proper scoring rule $s$, we have $s(x; j) = G_s(\delta_j) - D_{G_s}(\delta_j \parallel x)$. This is a specialization of the above point to the case where we take the “true” probability distribution to be $\delta_j$, i.e. the distribution that puts 100% probability on the outcome that actually happens.

- Or alternatively, if the expert initially believes $x$, and then receives new information that causes them to update their belief to $y$, then $D_G(y \parallel x)$ is the value of this new information to the expert (in terms of their expected increase in score).

These interpretations give an important intuition about the order of arguments to a Bregman divergence. It generally makes sense to think of the second argument as a less refined (i.e. less
informed) estimate, and the first argument as a more refined estimate.

Bregman divergences allow us to generalize proper scoring rules beyond the setting in which the realized outcome is simply one of $n$ options. Suppose that a principal wants to elicit some real-valued (or perhaps even vector-valued) quantity from an expert – for example, the amount that it will rain tomorrow in each of five cities. More precisely, the expert has a probability distribution over the outcome, and the principal wants to elicit the expected value of the distribution. For a strictly convex function $G$, consider the scoring rule

$$s(x; y) = -D_G(y \parallel x),$$

where $x$ is the expert’s reported expected value and $y$ is the realized outcome (e.g. the amount of rainfall in the five cities). This scoring rule is proper!

**Proposition 2.1.8** (Banerjee et al., 2005). *Given a (vector-valued) random variable $y$, the quantity $\mathbb{E} [D_G(y \parallel x)]$ is minimized by $x = \mathbb{E} [y]$. (This is true by a straightforward generalization of our argument above for the case where $y = \delta_j$ for some $j$. See e.g. Banerjee et al. (2005) for a full proof.)

We can also add an arbitrary function $f(y)$ to the scoring rule while maintaining properness, since the expert has no control over the realized outcome $y$. Therefore, scoring rules of the form

$$s(x; y) = f(y) - D_G(y \parallel x),$$

for strictly convex functions $G$ are proper. Indeed, this is an exhaustive characterization: all proper scoring rules take this form (Abernethy & Frongillo, 2012, Theorem 12).

Proper scoring rules, Bregman divergences, and the intuition behind them are ubiquitous in ABE, and in this thesis in particular.
2.2 Forecast aggregation methods

This section (and particularly Subsection 2.2.1) is necessary for Chapters 5, 6, and 7. It is also likely to be interesting in its own right.

The question of forecast aggregation – how to aggregate forecasts from two or more experts into a single forecast – is so natural that it hardly needs motivating. But to be concrete, here are three examples. First, from Neyman and Roughgarden (2023):

**Example 2.2.1.** You are a meteorologist tasked with advising the governor of Florida on hurricane preparations. A hurricane is threatening to make landfall in Miami, and the governor needs to decide whether to order a mass evacuation. The governor asks you for the likelihood of a direct hit, so you decide to consult several weather models. These models all give you different answers: 10%, 25%, 70%. You trust the models equally, but your job is to come up with one number for the governor: your best guess, all things considered. What is the most sensible way for you to aggregate these numbers?

Second, from Neyman and Roughgarden (2022a):

**Example 2.2.2.** Suppose that you wish to estimate how much the GDP of the United States will grow next year: perhaps you are making financial decisions and want to know whether to expect a downturn. You don’t personally know much about the question – just that the historical average rate of GDP growth has been 3% – but you look online and find several forecasts made by machine learning models. One model predicts 3.5% growth next year; another predicts 1.5%; a third predicts a downturn: -1% growth. How might you take this information into account and turn it into one number: your best guess for next year’s growth rate, all things considered?

Third, a new example:

**Example 2.2.3.** Three different neural networks – perhaps with different architectures – are trained to classify pictures of a thousand different animals. Each neural net takes an image as input and
outputs a probability distribution over the thousand classes. You want to ensemble (combine) these neural nets into a single classifier by aggregating the distributions that they output. What is the best way to do this?

These examples are diverse in two ways. First, the application domain: the problem of forecast aggregation is very broadly applicable. Beyond these examples, forecast aggregation finds uses in almost every natural and social science. And second, the mathematical domain of the forecasts themselves: the first aggregates probabilities; the second, arbitrary real numbers; and the third, probability distributions over many outcomes.

The “correct” way to aggregate forecasts very much depends on the setting. In this section, we will discuss some of the most common aggregation methods.

2.2.1 Linear and logarithmic pooling

Let’s introduce some basic notation:

- We will refer to the forecasters as experts. We will let $m$ be the number of experts, and will number them 1 through $m$.
- The forecasts will live inside some space $\mathcal{D}$, which itself lies in $\mathbb{R}^n$ for some $n$. In Example 2.2.2, $\mathcal{D}$ can be taken to be $\mathbb{R}$; in Example 2.2.3 $\mathcal{D} = \Delta_{1000} \subseteq \mathbb{R}^{1000}$. Example 2.2.1 can be thought of as either $\mathcal{D} = [0, 1] \subseteq \mathbb{R}$, or as $\mathcal{D} = \Delta_2 \subseteq \mathbb{R}^2$.
- In this section, we will call the forecasts $x_1, \ldots, x_m$ (these are elements of $\mathcal{D}$, and thus are vectors in $\mathbb{R}^n$). We will let $x_i(j)$ denote the $j$-th component of $x_i$. (Our notation will vary slightly throughout the thesis.)
- The experts will have weights $w_1, \ldots, w_m$. The intuition for weights is that some experts may be more informed or reliable than others, so it is natural to weigh their forecasts more heavily in the aggregate.

The most straightforward (and most common) way to aggregate forecasts is to average them. In the context of aggregation, averaging forecasts is often called linear pooling.
Definition 2.2.4. *The linear pool of forecasts* \( x_1, \ldots, x_m \) *with weights* \( w_1, \ldots, w_m \) *is their weighted arithmetic mean:* \[ \sum_{i=1}^{m} w_i x_i. \]

An important virtue of linear pooling is its simplicity. Linear pooling frequently outperforms attempts at more sophisticated pooling methods, which often lead to overfitting. Linear pooling is also *eventwise-independent*, meaning that the aggregate probability of outcome \( j \) only depends on the probabilities that the various experts assign to outcome \( j \), and not on the probabilities that they assign to other outcomes. (But, as we will see, this is not always desirable.)

In the context of probabilistic forecasts, a common alternative to linear pooling is called *logarithmic pooling*. Logarithmic pooling involves taking the geometric mean of forecasts, instead of the arithmetic mean. As a simple example, suppose that two experts forecast distributions over three possible outcomes (such as whether tomorrow will be sunny, cloudy, or rainy). Expert 1 forecasts \( x_1 = (60\%, 36\%, 4\%) \), while Expert 2 forecasts \( x_2 = (75\%, 5\%, 20\%) \). The logarithmic pool of these two forecasts (with equal weights) first takes the geometric mean of the two forecasts outcome by outcome (in this case, \( \sqrt{0.6 \cdot 0.75}, \sqrt{0.36 \cdot 0.05}, \sqrt{0.04 \cdot 0.2} \)), and then rescales the resulting forecast so that the probabilities add to 1 (which in this case will give \( (75\%, 15\%, 10\%) \)).

Definition 2.2.5. *Let* \( x_1, \ldots, x_m \) *be probability distributions over* \( n \) *outcomes that assign nonzero probability to every outcome. The logarithmic pool of* \( x_1, \ldots, x_m \) *with weights* \( w_1, \ldots, w_m \) *is obtained by taking the weighted geometric mean of the forecasts (componentwise) and rescaling the result to add to 1. That is, the logarithmic pool is the distribution* \( x^* \) *defined as*

\[ x^*(j) = c \prod_{i=1}^{m} (x_i(j))^{w_i}, \]

*for all outcomes* \( j \), *where* \( c \) *is the appropriate normalizing constant.*

Why take the geometric mean instead of the arithmetic mean? To illustrate, suppose that a monster is hiding under one of three beds, and two experts are forecasting a probability distribution over which bed the monster is hiding under. Expert 1 checks under Bed 1, doesn’t see a

\[ 3 \text{ Specifically, } c = \left( \sum_{j=1}^{n} \prod_{i=1}^{m} (x_i(j))^{w_i} \right)^{-1}. \]
monster, and so reports \( x_1 = (0.04\%, 49.98\%, 49.98\%) \). Meanwhile, Expert 2 checks under Bed 2, doesn’t see a monster, and so reports \( x_2 = (49.98\%, 0.04\%, 49.98\%) \). Then ideally, the aggregate forecast would put almost all probability mass on the monster being under Bed 3. This is not achievable with a linear pool, which will only assign a 49.98% probability to the monster being under Bed 3. By contrast, a logarithmic pool with equal weights gives an aggregate of roughly \((2.7\%, 2.7\%, 94.6\%)\), which is much more reasonable in this situation.

In other words, logarithmic pooling takes experts seriously when they assign low probabilities to outcomes – unlike linear pooling. Perhaps for this reason, logarithmic pooling has been found to perform very well on real-world data, typically better than linear pooling (Satopää et al., 2014).

Another, perhaps more natural, perspective on the logarithmic pool is that it is the geometric mean of forecasts – without any normalization – when the forecasts are viewed as odds. Consider our earlier example: \( x_1 = (60\%, 36\%, 4\%) \), \( x_2 = (75\%, 5\%, 20\%) \). To take the logarithmic pool, we write each forecast in terms of odds: \((60\%, 36\%, 4\%)\) becomes 60:36:4 odds, or (to simplify) 15:9:1, while \((75\%, 5\%, 20\%)\) becomes 15:1:4 odds. Then we take the geometric mean, which gives 15:3:2. We can then reinterpret these odds as a vector of probabilities: \((75\%, 15\%, 10\%)\).

But perhaps the most natural perspective on logarithmic pooling is as an arithmetic mean of the experts’ log-odds (or “logits,” in machine learning terminology). That is, while linear pooling takes the arithmetic mean of forecasts in probability space, logarithmic pooling takes the arithmetic mean of forecasts in log-odds space. In other words: the logarithmic pool of \( x_1, \ldots, x_m \) with weights \( w_1, \ldots, w_m \) can also be written as

\[
\ln x^*(1), \ldots, \ln x^*(n) \equiv \sum_{i=1}^{m} w_i \ln x_i(1), \ldots, \ln x_i(n),
\]

where \( \equiv \) denotes equality up to translation by a multiple of the all-ones vector.\(^4\)

To see why this is a natural perspective, let us for simplicity consider the case of a binary event, such that every expert reports the probability \( x_i \) of the “Yes” outcome. Then the logarithmic pool

\[^4\text{Log-odds vectors are most naturally thought of belonging to } \mathbb{R}^n \text{ modulo the all-ones vector. Translation plays the same role that rescaling plays in Definition 2.2.5.}\]
of $x_1, \ldots, x_m$ with weights $w_1, \ldots, w_m$ can be defined as the probability $x^*$ that satisfies
\[
\ln \frac{x^*}{1 - x^*} = \sum_{i=1}^{m} w_i \ln \frac{x_i}{1 - x_i}.
\] (2.3)

Thinking of logarithmic pooling in terms of log-odds is natural because log-odds are units of the strength of Bayesian evidence. For example, suppose that our $m$ forecasters are estimating the probability of some event $X$, and start with some common prior. Each expert $i$ receives Bayesian evidence $E_i$ and performs a Bayesian update to obtain their posterior probability:
\[
\frac{\Pr [X | E_i]}{\Pr [\neg X | E_i]} = \frac{\Pr [X]}{\Pr [\neg X]} \cdot \frac{\Pr [E_i | X]}{\Pr [E_i | \neg X]}.
\]
Or, taking the log of both sides:
\[
\ln \frac{\Pr [X | E_i]}{\Pr [\neg X | E_i]} = \ln \frac{\Pr [X]}{\Pr [\neg X]} + \ln \frac{\Pr [E_i | X]}{\Pr [E_i | \neg X]}.
\]
That is, in log-odds space, every expert’s posterior is equal to their prior, plus a term that represents the strength of the evidence $E_i$ in favor of (or against) $X$.

And so, what does taking the average of the experts’ posterior log-odds represent? This average is equal to
\[
\ln \frac{\Pr [X]}{\Pr [\neg X]} + \frac{1}{m} \sum_{i=1}^{m} \ln \frac{\Pr [E_i | X]}{\Pr [E_i | \neg X]}.
\]
In other words, the logarithmic pool represents the posterior probability of a hypothetical expert who received evidence whose strength in favor of (or against) $X$ was the average of the strengths of all $m$ experts’ evidence.

However, this framing raises an important question: why average the experts’ Bayesian evidence? Would it not be more appropriate to add them, so as to fully incorporate all experts’ evidence? This brings us to the concept of extremization.
2.2.2 Extremization

Suppose that two experts are forecasting the probability of an event $X$, and that the experts have a common prior of $\frac{1}{2}$. Then, each expert receives a piece of evidence that causes them to update to $\frac{2}{3}$, and so both experts report a probability of $\frac{2}{3}$ to the aggregator. Both linear and logarithmic pooling will output an aggregate probability of $\frac{2}{3}$. But is that the correct way to aggregate the experts’ forecasts?

The answer is: it depends! Suppose, for example, that a coin with unknown bias (i.e. probability of heads) – uniformly selected from $[0, 1]$ – will be flipped tomorrow, and the experts are forecasting the probability that the coin will come up heads. The experts’ prior is, of course, $\frac{1}{2}$.

Now, first, suppose that the evidence that the experts see is one flip of the coin – the same flip. If the coin comes up heads, then each expert’s posterior probability will be $\frac{2}{3}$.

And because both experts saw the same coin flip, the correct aggregate is $\frac{2}{3}$ as well. The linear and logarithmic pools get this one right.

But now, suppose that the experts see different, independent flips of the coin, both of which come up heads. Each expert’s posterior will still be $\frac{2}{3}$, but this time the correct aggregate will be more than $\frac{2}{3}$ ($\frac{3}{4}$, as it happens). That’s because the experts observed different pieces of evidence, both of which caused them to update in the same direction. Accounting for both experts’ evidence – the fact that both flips came up heads – results in an even higher probability.

The key insight from these two contrasting examples is this: if the experts’ evidence is identical or heavily overlapping, then taking some sort of average – perhaps a linear pool (average of probabilities) or logarithmic pool (average of log-odds) – is sensible. But if the experts’ evidence is heavily non-overlapping or disjoint, then these pooling methods are insufficiently extremizing: they take a sort of average of the experts’ evidence, when it would make more sense to do something more like adding the evidence. The result is an estimate that is too close to the prior.

A natural solution to this problem is to pool the experts’ forecasts and then to extremize the

---

5This can be verified by performing a Bayesian update on this new evidence. It also follows from a theorem known as Laplace’s rule of succession.
pooled forecast – meaning, to push it away from the prior. The smaller the overlap between the experts’ information, the larger the appropriate amount of extremization.\(^6\) The benefits of extremization are not just theoretical: extremization has been found to improve the quality of aggregate forecasts in practice (Satopää et al., 2014; Sevilla, 2021). Indeed, taking the logarithmic pool of forecasts and then extremizing is a state-of-the-art forecast aggregation method.

2.2.3 The generalized linear and logarithmic pools

One natural way to extremize linear and logarithmic pools is to allow weights to vary freely, rather than requiring them to add to one. For example, we might attempt to define the “linear pool” of \(x_1, \ldots, x_m\) with weights \(w_1, \ldots, w_m\) (not necessarily adding to 1) as before: \(\sum_{i=1}^{m} w_i x_i\).

This does not quite make sense, as it is not invariant to adding a constant. That is, suppose that \(x_1, \ldots, x_m\) are all forecasts of some quantity \(x\). Define \(y := x + c\) for some known vector \(c\). Then the experts’ forecasts for \(y\) (call them \(y_i\)) will be \(y_i = x_i + c\). The linear pool of \(y_1, \ldots, y_m\) with weights \(w_1, \ldots, w_m\) ought to be equal to the linear pool of \(x_1, \ldots, x_m\) with the same weights, plus \(c\). This is the case if \(\sum_i w_i = 1\), but not in general.

We can fix this problem if we have access to a prior \(x_0\). (In Example 2.2.2 above, for example, the prior might be the historical average growth rate, 3%.) In that case, we can treat the experts’ forecasts in terms of their updates from the prior, and aggregate those updates with arbitrary weights. We will call this the generalized linear pool.

**Definition 2.2.6.** The generalized linear pool of forecasts \(x_1, \ldots, x_m\) with weights \(w_1, \ldots, w_m\), given a prior \(x_0\), is defined as

\[
x_0 + \sum_{i=1}^{m} w_i (x_i - x_0).
\]

If \(\sum_i w_i > 1\), we can think of this aggregation strategy as linearly pooling the forecasts and then extremizing the result by pushing it away from the prior by a constant factor. Satopää and

\(^6\)If the experts’ pieces of evidence are independent conditional on the outcome, then adding the sizes of their updates from the prior in log-odds space – or, in other words, extremizing the logarithmic pool by a factor of \(m\) – gives exactly the right answer.
Ungar (2015) called this technique *linear extremization*.

We can define the generalized logarithmic pool similarly. Recall that logarithmic pooling can be thought of as taking an average (i.e. linear pool) of the experts’ log-odds. We can instead take a *generalized* linear pool of the log-odds.

**Definition 2.2.7.** Given a forecast \( x \), let \( \ell(x) := (\ln x(1), \ldots, \ln x(n)) \). The **generalized logarithmic pool** of forecasts \( x_1, \ldots, x_m \) with weights \( w_1, \ldots, w_m \), given a prior \( x_0 \), is the forecast \( x^* \) satisfying

\[
\ell(x^*) \equiv \ell(x_0) + \sum_{i=1}^{m} w_i (\ell(x_i) - \ell(x_0)),
\]

where \( \equiv \) denotes equality up to translation by a multiple of the all-ones vector.

If each expert’s evidence is independent conditioned on the eventual outcome, then the generalized logarithmic pool with weights \( w_1 = \cdots = w_m = 1 \) gives exactly the correct aggregate. Since experts’ evidence typically has substantial overlap, it usually makes sense to choose much smaller weights. Satopää et al. (2014) found that setting all weights to \( 2/m \) (so that the sum of all of the weights is 2) results in good performance on real-world data.

### 2.3 Information structures

*This section is necessary for Chapters 7 and 8, and is also useful for Chapter 6.*

#### 2.3.1 Introduction to information structures

Perhaps the most fundamental concept of ABE is the *information structure*. Informally speaking, an information structure is a full description of all information possessed by a set of experts in all possible states of the world.

To be concrete, suppose that we are interested in the value of some (possibly real-valued, possibly vector-valued) random variable \( Y \): perhaps the amount that it will rain tomorrow. There are various pieces of information in the world that are relevant to \( Y \) (e.g. the current dew point, or
the air pressure forecast by the GFS weather model). We call these pieces of information *signals*.

From a Bayesian standpoint, there is some joint prior probability distribution over the signals and the value of $Y$. We call such a prior an *information structure*.

**Definition 2.3.1.** An information structure $\mathcal{I} = (\Omega, \mathbb{P}, \sigma, Y)$ consists of:

- A set of states of the world $\Omega$, together with a probability distribution $\mathbb{P}$ over $\Omega$.
- A collection of $m$ signals $\sigma = (\sigma_1, \ldots, \sigma_m)$, which are random variables defined on $\Omega$. That is, for $i = 1 \ldots m$, we have a signal $\sigma_i : \Omega \to S_i$, where $S_i$ – the set of possible values that $\sigma_i$ can take on – is called the $i$-th signal set. (We will often think of each signal as belonging to a different expert.)
- A random variable $Y : \Omega \to \mathbb{R}^n$ for some $n$.

Let’s unpack this definition through a series of examples.

**Example 2.3.2.** There is a coin that has bias (i.e. probability of heads) $Y$, which is either $\frac{1}{3}$ or $\frac{2}{3}$; these two possibilities are equally likely. Two experts each see a different flip of the coin.

Here, we can think of $S_1 = \{H, T\}$, with $\sigma_1 = H$ if Expert 1 sees heads and $\sigma_1 = T$ if Expert 1 sees tails. $S_2$ and $\sigma_2$ are defined analogously for Expert 2. We can define $\Omega$ as having eight states that together describe $Y$, $\sigma_1$, and $\sigma_2$. Each of these states has a certain probability. For example, the probability of the state $\{Y = \frac{2}{3}, \sigma_1 = H, \sigma_2 = T\}$ is $\frac{1}{2} \cdot \frac{2}{3} \cdot \frac{1}{3} = \frac{1}{9}$. That’s because there’s a 50% chance that $Y = \frac{2}{3}$; and conditional on that, there’s a $\frac{2}{3}$ chance that $\sigma_1 = H$ and a $\frac{1}{3}$ chance that $\sigma_2 = T$.

Often we are interested the expected value of $Y$ conditioned on the experts’ information. For this reason, it is often useful to summarize information structures like this one using two tables:

$$
\begin{array}{c|cc}
\mathbb{E}[Y] : & \sigma_2 = H & \sigma_2 = T \\
\sigma_1 = H & 3/5 & 1/2 \\
\sigma_1 = T & 1/2 & 2/5
\end{array}
\quad
\begin{array}{c|cc}
\mathbb{P}[\sigma_1, \sigma_2] : & \sigma_2 = H & \sigma_2 = T \\
\sigma_1 = H & 5/18 & 2/9 \\
\sigma_1 = T & 2/9 & 5/18
\end{array}
$$

For example, the expected value of $Y$ conditioned on $\sigma_1 = H$ and $\sigma_2 = H$ is $\frac{3}{5}$, and the probability that $\sigma_1 = H$ and $\sigma_2 = H$ is $\frac{5}{18}$.  

This table also allows us to compute the expected value of $Y$ conditioned on some individual expert’s signal. For example, $\mathbb{E}[Y \mid \sigma_1 = H] = \frac{3/5 + 1/2 + 2/9}{5/18 + 2/9} = \frac{5}{9}$.

**Example 2.3.3.** Just like in the previous example, there is a coin with bias $Y$ that is either $\frac{1}{3}$ or $\frac{2}{3}$. But this time, the two experts see the same flip of the coin.

In this case, we can think of $\Omega$, $S_1$, and $S_2$ the same way, but now $\mathbb{P}$ is different. In particular, the probability of $\{Y = \frac{2}{3}, \sigma_1 = H, \sigma_2 = T\}$ is now 0, because $\sigma_1$ and $\sigma_2$ are guaranteed to be either both $H$ or both $T$ in this information structure. The following table summarizes this information structure:

<table>
<thead>
<tr>
<th>$\mathbb{E}[Y]$</th>
<th>$\sigma_2 = H$</th>
<th>$\sigma_2 = T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma_1 = H$</td>
<td>5/9</td>
<td>-</td>
</tr>
<tr>
<td>$\sigma_1 = T$</td>
<td>-</td>
<td>5/9</td>
</tr>
<tr>
<td>$\mathbb{P}[\sigma_1, \sigma_2]$</td>
<td>$\sigma_2 = H$</td>
<td>$\sigma_2 = T$</td>
</tr>
<tr>
<td>$\sigma_1 = H$</td>
<td>1/2</td>
<td>0</td>
</tr>
<tr>
<td>$\sigma_1 = T$</td>
<td>0</td>
<td>1/2</td>
</tr>
</tbody>
</table>

These contrasting examples illustrate that an information structure captures not just the probability distribution of each expert’s information individually, but also the interaction between the experts’ information, e.g. how their signals are correlated.

Next we’ll introduce a particular type of information structure that we will find useful.

**Definition 2.3.4.** *(Satopää et al., 2016)* An information structure in the partial information framework (henceforth, a PIF information structure) is an information structure that takes the following form: for each subset $S \subseteq [m]$, there is a random variable $X_S$ (all of these random variables are independent), and $\sigma_i$ is the tuple of random variables $X_S$ for all $S$ containing $i$. The value of $Y$ is equal to $\sum_S X_S$.

If we think of the signals as belonging to experts, each $X_S$ as a piece of evidence (a real number, or perhaps a vector of reals), and expert $i$ has access to $X_S$ if $i \in S$. These pieces of evidence behave additively, in the sense that $Y$ is the sum of all of random variables $X_S$.

PIF information structures are interesting to study in the context of aggregation. Expert $i$’s estimate of $Y$ – that is, the expected value of $Y$ conditioned on $\sigma_i$ – is equal to $\sum_{S \ni i} X_S + \sum_{S \not\ni i} \mathbb{E}[X_S]$. For convenience, we typically assume that each $X_S$ has mean zero; this is the case without loss of
generality in all of our applications. Under this assumption, expert $i$’s estimate of $Y$ reduces to the sum of the $X_S$’s that Expert $i$ sees: $\sum_{S \ni i} X_S$.

Now consider an aggregator who sees each expert’s estimates. If the aggregator knew not just each expert’s estimate, but also their signal in full (i.e. all of the tuples of $X_S$-values), then their job would be straightforward: just add all the $X_S$-values. But if (as is typical) the aggregator cannot access this information, then aggregation becomes nontrivial. In Section 2.4, we will discuss the optimal aggregation strategy in the specific case that each $X_S$ is normally distributed.

2.3.2 Informational substitutes

The space of information structures is vast: there’s a huge number of ways in which different experts can have overlapping information about the value of a quantity. Relatively few nontrivial facts are known about all information structures. So a typical theorem about information structures imposes conditions: if an information structure $I$ satisfies [condition], then....

One natural constraint is that, for the purposes of accurately estimating $Y$, there are diminishing marginal returns to learning new signals. An example of diminishing marginal returns would be that learning Signal 7 is more useful if you only know Signal 2, than if you know both Signal 2 and Signal 4. This condition is particularly intuitive in the context of experts with overlapping information: the greater the information overlap between different experts, the fewer returns there are to learning additional experts’ information.

The general name for diminishing marginal returns to learning new signals is informational substitutes. The notion of informational substitutes was first explored by Börgers et al. (2013), though we will mostly be interested in building on definitions introduced by Chen and Waggoner (2016).\footnote{I recommend the ArXiv version of Chen and Waggoner (2016) for the most up-to-date introduction to informational substitutes.}

The concept of diminishing marginal returns to estimating $Y$ only makes sense in the context of some sort of “value function” – that is, a function that describes the quality of an estimate of $Y$. For example, we could judge an estimate of $Y$ based on its squared error. That is, we could say
that an information structure satisfies informational substitutes if there are diminishing marginal
returns to learning new signals, as measured by the squared error when estimating $Y$. Formalizing
this idea results in a concept called \textit{weak informational substitutes} (with respect to squared error).

In the following definition – and more generally, throughout the thesis – we will use the fol-
lowing notation: given a subset $A \subseteq [m]$ of signals, $Y_A$ is the expected value of $Y$ conditioned on
the signals in $A$. For example, $Y_{\emptyset}$ is the prior $\mathbb{E}[Y]$; for $i \in [m]$, $Y_i$ is the expected value of $Y$
conditioned on $\sigma_i$; and $Y_{[m]}$ is the expectation of $Y$ conditioned on all information that is present.

\textbf{Definition 2.3.5.} (Weak substitutes w.r.t. squared error (Chen & Waggoner, 2016)) Let $I = (\Omega, \mathbb{P}, \sigma, Y)$
be an information structure. $I$ satisfies \textit{weak informational substitutes} (or simply weak substitutes)
with respect to squared error if, for all $B \subseteq A \subseteq [m]$ and $i \notin A$, we have\footnote{More abstractly, we could say that $I$ satisfies weak substitutes with respect to squared error if $-(Y - Y_{A})^2$ is a submodular set function on the subsets $A$ of $[m]$.}

\begin{equation}
\mathbb{E} \left[ (Y - Y_A)^2 \right] - \mathbb{E} \left[ (Y - Y_{A\cup\{i\}})^2 \right] \leq \mathbb{E} \left[ (Y - Y_B)^2 \right] - \mathbb{E} \left[ (Y - Y_{B\cup\{i\}})^2 \right].
\end{equation}

That is, the reduction in squared error gained by learning $\sigma_i$ if you already know all the signals
in $A$ (that’s the left-hand side) is smaller than the reduction in squared error gained by learning $\sigma_i$
if you only know the signals in $B$, which is a subset of $A$ (that’s the right-hand side).

\textbf{Example 2.3.6.} It is straightforward to see that the information structure in Example 2.3.3 satisfies
weak substitutes with respect to squared error. That’s because $\sigma_1$ and $\sigma_2$ are identical. The value
of $\sigma_2$, in terms of reduction in squared error, is positive when no signal is known, but is exactly
zero when $\sigma_1$ is already known.

\textbf{Example 2.3.7.} Every PIF information structure satisfies weak substitutes with respect to squared
error. This is left as an exercise for now, though we will prove a stronger statement in Chapter 7.

The following non-example of informational substitutes, which we call the XOR information
structure, will be useful in future discussions.
Definition 2.3.8. The XOR information structure consists of two signals, \( \sigma_1 \) and \( \sigma_2 \), that are independent, random bits (i.e. either 0 or 1, with equal probability). The value of \( Y \) is equal to the binary XOR of \( \sigma_1 \) and \( \sigma_2 \), i.e. 0 if \( \sigma_1 = \sigma_2 \) and 1 if \( \sigma_1 \neq \sigma_2 \).

To see that the XOR information structure does not satisfy informational substitutes, we observe that the prior is \( \frac{1}{2} \) and also the expected value of \( Y \) conditioned on any one signal is always \( \frac{1}{2} \). That is, having just one signal is completely uninformative! On the other hand, knowing both signals gives away the value of \( Y \) exactly. Informally speaking, the opposite of informational substitutes is known as informational complements (see Chen and Waggoner (2016)), and XOR is a prototypical example of informational complements.

We can also define weak substitutes more generally, for other notions of error. Really, for any differentiable convex function \( G \), the Bregman divergence with respect to \( G \) from \( Y \) to \( Y_A \) makes sense in place of \( (Y - Y_A)^2 \). Bregman divergences are sensible in this context because they *elicit the mean*: for every \( Y \), \( \mathbb{E} [D_G(Y \parallel x)] \) is minimized at \( x = \mathbb{E} [Y] \) (see Proposition 2.1.8 above).

Thus, more generally, we say:

Definition 2.3.9 (Weak substitutes (Chen & Waggoner, 2016)). Let \( \mathcal{I} = (\Omega, \mathbb{P}, \mathbf{\sigma}, Y) \) be an information structure. \( \mathcal{I} \) satisfies weak substitutes with respect to a differentiable convex function \( G \) if, for all \( B \subseteq A \subseteq [m] \) and \( i \notin A \), we have

\[
\mathbb{E} [D_G(Y \parallel Y_A)] - \mathbb{E} [D_G(Y \parallel Y_{A \cup \{i\}})] \leq \mathbb{E} [D_G(Y \parallel Y_B)] - \mathbb{E} [D_G(Y \parallel Y_{B \cup \{i\}})].
\]

Why weak substitutes? Chen and Waggoner (2016) define stronger notions of substitutes as well. These notions consider partial revelation of signals, thus requiring submodularity over a finer space. We won’t go into further detail here: while these definitions are interesting, the work presented in this thesis builds specifically on the notion of weak substitutes.
2.3.3 Random variables as vectors and the Pythagorean theorem

It is often useful to think of random variables as vectors with the inner product $\langle X, Y \rangle := \mathbb{E}[XY]$. (More formally: given a probability space $(\Omega, \mathcal{F}, P)$, the set of random variables with finite variance on $(\Omega, \mathcal{F}, P)$, endowed with the inner product $\langle X, Y \rangle := \mathbb{E}[XY]$, is a Hilbert space over $\mathbb{R}$.)

If the state space $\Omega$ is finite, it often makes sense to think of a random variable $X$ geometrically, as a vector with one coordinate per state $\omega \in \Omega$, where the value of the coordinate is the value of $X$ on $\omega$. This perspective on random variables is particularly useful because it gives a very natural notion of an orthogonal projection. As we will soon show, when random variables are thought of as vectors in this way, orthogonal projections correspond to conditional expectations.

Consider an information structure $I = (\Omega, \mathcal{P}, \sigma, Y)$. Recall our notation $Y_A$ from the previous section, which means the expectation of $Y$ conditioned on all signals in $A$ (a subset of $[m]$). Each $Y_A$ is a random variable that only depends on the values of the signals in $A$. For example, $Y_\emptyset$ does not depend on the value of any signals: it is the same across all of $\Omega$. For any $i \in [m]$, $Y_{\{i\}}$ depends on the value of $\sigma_i$, but on no other signals: if $\sigma_i(\omega_1) = \sigma_i(\omega_2)$, then $Y_{\{i\}}(\omega_1) = Y_{\{i\}}(\omega_2)$.

Now, consider two subsets of signals $A$ and $B$, such that $B$ is a subset of $A$. This means that $Y_B$ is a coarser estimate for $Y$ than $Y_A$ is: it is an estimate (conditional expectation) based on a smaller set of signals. In the aforementioned inner product space, $Y_B$ is the orthogonal projection of $Y_A$ onto the subspace of all random variables whose values only depend on the signals in $B$. (Figure 2.3 illustrates this relationship.)

**Proposition 2.3.10.** Let $I = (\Omega, \mathcal{P}, \sigma, Y)$ be an information structure, $B \subseteq A \subseteq [m]$, $Y_A = \mathbb{E}[Y | \{\sigma_i\}_{i \in A}]$, and $Y_B = \mathbb{E}[Y | \{\sigma_i\}_{i \in B}]$. In the inner product space$^9$ of random variables on $\Omega$ defined by $\langle X_1, X_2 \rangle = \mathbb{E}[X_1X_2]$, $Y_B$ is the orthogonal projection of $Y_A$ onto the subspace of all random variables whose values only depend on the signals in $B$.

**Proof.** We need to show that $Y_A - Y_B$ has inner product zero with every random variable whose

---

$^9$Technically a Hilbert space, rather than an inner product space.
Figure 2.3: For a random variable $Y$, let $B \subseteq A$ be sets of signals. Let $Y_A$ be the expected value of $Y$ conditioned on the signals in $A$, and define $Y_B$ analogously. In the space of random variables with inner product $\langle X_1, X_2 \rangle := \mathbb{E} [X_1 X_2]$, $Y_B$ is the orthogonal projection of $Y_A$ onto the subspace of random variables whose values only depend on the signals in $B$.

value depends on the signals in $B$. First, note that

$$Y_B = \mathbb{E} [Y \mid \{\sigma_i\}_{i \in B}] = \mathbb{E} [\mathbb{E} [Y \mid \{\sigma_i\}_{i \in A}] \mid \{\sigma_i\}_{i \in B}] = \mathbb{E} [Y_A \mid \{\sigma_i\}_{i \in B}].$$

In other words, $Y_B$ is the expected value of $Y_A$ conditioned on the signals in $B$.\(^\text{10}\) This means that

$$\mathbb{E} [Y_A - Y_B \mid \{\sigma_i\}_{i \in B}] = \mathbb{E} [Y_A \mid \{\sigma_i\}_{i \in B}] - Y_B = 0.$$

Thus, for a variable $Z$ that only depends on the signals in $B$, we can write

$$\mathbb{E} [(Y_A - Y_B)Z] = \mathbb{E} [\mathbb{E} [(Y_A - Y_B)Z \mid \{\sigma_i\}_{i \in B}]] = \mathbb{E} [Z \mathbb{E} [Y_A - Y_B \mid \{\sigma_i\}_{i \in B}]] = \mathbb{E} [Z \cdot 0] = 0,$$

as desired. \qed

\(^{10}\)Formally, Proposition 2.3.10 is a special case of the following more general fact: given a ran-
dom variable $X$ in a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, and a sub-sigma-algebra $\mathcal{H} \subseteq \mathcal{F}$, the orthogonal projection of $X$ onto the subspace of $\mathcal{H}$-measurable random variables is exactly the conditional expectation $\mathbb{E} [X \mid \mathcal{H}]$ (Šidák, 1957). However, we only need the more specific fact that we just proved.)

Now, this orthogonal projection property means that we can use the Pythagorean theorem! Concretely, if $Z$ is some random variable whose value depends only on the signals in $A$, then $Z$, $Y_B$, and $Y_A$ form a “right triangle” with the right angle at $Y_B$ (see Figure 2.3). In our inner product space, the squared distance between two variables $X$ and $Y$ is $\langle X - Y, X - Y \rangle = \mathbb{E} [(X - Y)^2]$. And so, we have

$$\mathbb{E} [(Y_A - Z)^2] = \mathbb{E} [(Y_A - Y_B)^2] + \mathbb{E} [(Y_B - Z)^2].$$

More formally (and also more generally):

**Proposition 2.3.11.** (Pythagorean theorem for squared distance). In a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, let $A$ be a real-valued random variable, let $B = \mathbb{E} [A \mid \mathcal{H}]$ where $\mathcal{H}$ is a sub-sigma-algebra, and $C$ be a real-valued random variable defined on $\mathcal{H}$. Then

$$\mathbb{E} [(A - C)^2] = \mathbb{E} [(A - B)^2] + \mathbb{E} [(B - C)^2].$$

**Proof.** Observe that $\mathbb{E} [AB] = \mathbb{E} [\mathbb{E} [AB \mid \mathcal{H}]] = \mathbb{E} [B \mathbb{E} [A \mid \mathcal{H}]] = \mathbb{E} [B^2]$, so $\mathbb{E} [(A - B)^2] = \mathbb{E} [A^2] - \mathbb{E} [B^2]$. Also, note that $\mathbb{E} [AC] = \mathbb{E} [\mathbb{E} [AC \mid \mathcal{H}]] = \mathbb{E} [\mathbb{E} [A \mid \mathcal{H}] C] = \mathbb{E} [BC]$, where in the second step we use the fact that $C$ is defined on $\mathcal{H}$. Therefore, we have


$$= \mathbb{E} [(A - B)^2] + \mathbb{E} [(B - C)^2],$$

as desired. \qed

---

11Readers unfamiliar with sigma-algebras can think of $\mathcal{H}$ as supplying partial information about the state of the world $\omega$ (much like a signal). A random variable is defined on $\mathcal{H}$ if its value depends only on the information that $\mathcal{H}$ provides (i.e. the value of the signal).
In fact, the Pythagorean theorem turns out to be true for any Bregman divergence, not just for squared distance! Further, the result extends to vector-valued random variables.

**Proposition 2.3.12.** (Pythagorean theorem for Bregman divergence) In a probability space \((\Omega, \mathcal{F}, \mathbb{P})\), let \(A : \Omega \to \mathbb{R}^n\) be a random variable, let \(B = \mathbb{E} [A | \mathcal{H}]\) where \(\mathcal{H}\) is a sub-sigma-algebra, and let \(C : \Omega \to \mathbb{R}^n\) be a random variable that is defined on \(\mathcal{H}\). Let \(G\) be a differentiable, convex function defined on a convex set that includes all values taken on by \(A\), \(B\), and \(C\). Then

\[
\mathbb{E} [D_G(A \parallel C)] = \mathbb{E} [D_G(A \parallel B)] + \mathbb{E} [D_G(B \parallel C)].
\]

**Proof.** We have

\[
\begin{align*}
\mathbb{E} [D_G(A \parallel B)] + \mathbb{E} [D_G(B \parallel C)] &- \mathbb{E} [D_G(A \parallel C)] \\
= \mathbb{E} [G(A) - G(B) - (A - B) \cdot \nabla G(B) + G(B) - G(C) - (B - C) \cdot \nabla G(C)] \\
&- G(A) + G(C) + (A - C) \cdot \nabla G(C) \\
= \mathbb{E} [(A - B) \cdot (\nabla G(C) - \nabla G(B))] = \mathbb{E} [\mathbb{E} [(A - B) \cdot (\nabla G(C) - \nabla G(B)) | \mathcal{H}]] \\
= \mathbb{E} [\nabla G(C) - \nabla G(B)] \cdot \mathbb{E} [A - B | \mathcal{H}] = \mathbb{E} [(\nabla G(C) - \nabla G(B)) \cdot (\mathbb{E} [A | \mathcal{H}] - B)] = 0.
\end{align*}
\]

The third-to-last step follows from the fact that \(g(C) - g(B)\) is \(\mathcal{H}\)-measurable (we are using the “pulling out known factors” property of conditional expectations). The last step follows from the fact that \(\mathbb{E} [A | \mathcal{H}] = B\).

While Proposition 2.3.12 is a natural extension of the well-known Proposition 2.3.11, and its proof is straightforward, to my knowledge it first appeared in Frongillo et al. (2023) (which is presented in Chapter 8 of this thesis).

Note that these Pythagorean theorems allow us to rewrite the definitions of weak substitutes from the previous section:

**Proposition 2.3.13.** Let \(I = (\Omega, \mathbb{P}, \sigma, Y)\) be an information structure. \(I\) satisfies weak substitutes
with respect to squared error if, for all $B \subseteq A \subseteq [m]$ and $i \notin A$, we have

$$\mathbb{E} \left[ (Y_{A \cup \{i\}} - Y_A)^2 \right] \leq \mathbb{E} \left[ (Y_{B \cup \{i\}} - Y_B)^2 \right].$$

More generally, given a differentiable convex function $G$, $I$ satisfies weak substitutes with respect to $G$ if, for all $B \subseteq A \subseteq [m]$ and $i \notin A$, we have

$$\mathbb{E} \left[ D_G(Y_{A \cup \{i\}} \parallel Y_A) \right] \leq \mathbb{E} \left[ D_G(Y_{B \cup \{i\}} \parallel Y_B) \right].$$

**Proof.** We prove the first statement; the proof of the second statement is exactly analogous. We can rewrite the left-hand side of Equation 2.4 as $\mathbb{E} \left[ (Y_{A \cup \{i\}} - Y_A)^2 \right]$ by using the Pythagorean theorem. (Specifically, we set $A := Y$, $B := Y_{A \cup \{i\}}$, and $C := Y_A$ in the statement of Proposition 2.3.11.) Similarly, we can rewrite the right-hand side of Equation 2.4 as $\mathbb{E} \left[ (Y_{B \cup \{i\}} - Y_B)^2 \right].$ \hfill $\square$

### 2.4 Bayesian justifications for generalized linear and logarithmic pooling

This section offers further justification for generalized linear and logarithmic pooling. While not required for any chapter, it provides useful context for Chapters 5, 6, 7, and 9. This section should also be of independent interest, and it contains some original work.

#### 2.4.1 A Bayesian justification for generalized linear pooling

We are often interested in aggregating forecasts in the context of an information structure. That is, we are interested in a (real-valued) random variable $Y$, and each expert $i$ receives a signal $\sigma_i$ and reports $\mathbb{E} \left[ Y \mid \sigma_i \right]$ to the aggregator. (Our notation for this quantity has been $Y_{\{i\}}$, but in this section we will instead use $Y_i$.)

For simplicity, let us suppose that the prior on the value of $Y$ is zero, i.e. $Y_0 := \mathbb{E} [Y] = 0.$
In this case, a generalized linear pool of the experts’ forecasts (as introduced in Section 2.2.3) is simply some weighted sum of the \( Y_i \)’s.

Which generalized linear pool is best for a given information structure? This depends on the notion of error, but suppose we care about expected squared error. Then the optimal linear combination of the \( Y_i \)’s is the orthogonal projection of \( Y \) onto the vector subspace of random variables spanned by \( Y_1, \ldots, Y_m \). Or in other words, it is the linear regression of \( Y \) onto the features \( Y_1, \ldots, Y_m \). The formula for this linear combination is

\[
\text{diag}(\Sigma)^\top \Sigma^{-1} Y,
\]

where \( \Sigma \) is the covariance matrix of the \( Y_i \)’s and \( Y \) is the vector of \( Y_i \)’s.\(^{12}\)

Is there a natural class of information structures for which the linear regression estimate for \( Y \) is not just the best linear estimate, but indeed the best estimate? Such a class of information structures would offer a Bayesian justification for generalized linear pooling, in the sense that a perfect Bayesian with knowledge of the information structure would choose to estimate \( Y \) with a particular generalized linear pool of the \( Y_i \)’s.

Now, the best estimate of \( Y \) given \( Y_1, \ldots, Y_m \) is just its conditional expectation: \( \mathbb{E} [Y \mid Y_1, \ldots, Y_m] \). Thus, we are looking for an information structure in which \( \mathbb{E} [Y \mid Y_1, \ldots, Y_m] \) is exactly equal to the linear regression estimate for \( Y \).

Recall partial information framework (PIF) information structures from Definition 2.3.4.

**Claim 2.4.1** (Satopää et al., 2017). *Let \( \mathcal{I} \) be a PIF information structure in which every \( X_S \) is Gaussian with mean zero. Then the expected value of \( Y \) conditioned on \( Y_1, \ldots, Y_m \) is equal to the linear regression estimate for \( Y \):*

\[
\mathbb{E} [Y \mid Y_1, \ldots, Y_m] = \text{diag}(\Sigma)^\top \Sigma^{-1} Y.
\]

*Proof.* Let us write \( Y \) as its projection onto the subspace of \( Y_i \)’s plus the orthogonal remainder,

\(^{12}\)If \( \Sigma \) is not invertible, we take the Moore-Penrose pseudoinverse of \( \Sigma \) instead of the inverse.
which we will call \( Z \):

\[
Y = \text{diag}(\Sigma)^\top \Sigma^{-1} Y + Z. \tag{2.5}
\]

Note that \( Y = \sum_S X_S \) and \( Y_i = \sum_{S \ni i} X_S \). Thus, \( Z \) is also a linear combination of the \( X_S \)'s, and so \( Y_1, \ldots, Y_m, Z \) are jointly Gaussian. Further, every \( Y_i \) is uncorrelated with \( Z \), i.e. \( \mathbb{E} [ Y_i Z ] = 0 \), since \( Z \) is orthogonal to the subspace of \( Y_i \)'s. Note also that \( \mathbb{E} [ Z ] = 0 \), as \( Z \) is a linear combination of \( Y \) and the \( Y_i \)'s.

Now, jointly Gaussian and uncorrelated vectors of random variables are independent (Mardia et al., 1979). This means that \( \mathbb{E} [ Z \mid Y_1, \ldots, Y_m ] = 0 \). Taking the expectation of both sides of Equation 2.5 conditioned on \( Y_1, \ldots, Y_m \) gives us the desired result.

### 2.4.2 A Bayesian justification for generalized logarithmic pooling

To my knowledge, the main result of this section (Claim 2.4.2) is original to this thesis.

Suppose that \( m \) experts are forecasting a yes/no outcome, and assume for simplicity that they have a common prior of \( \frac{1}{2} \). In this case, the generalized logarithmic pool is defined as

\[
\ln \frac{x^*}{1 - x^*} = \sum_{i=1}^{m} w_i \ln \frac{x_i}{1 - x_i}
\]

for some weights \( w_1, \ldots, w_m \). Can we come up with a Bayesian justification for generalized logarithmic pooling, much as Gaussian PIF information structures gave us a Bayesian justification for generalized linear pooling? The answer is yes!

**Claim 2.4.2.** Consider the following information structure: \( Y \) is either 0 or 1, uniformly. If \( Y = 1 \), then the tuple of signals, \((\tau_1, \ldots, \tau_m)\), is drawn from a multivariate normal distribution with some mean \( \mu \) and covariance matrix \( \Sigma \). If \( Y = 0 \), then \((\tau_1, \ldots, \tau_m)\) is drawn from the multivariate normal distribution with mean \(-\mu\) and some invertible covariance matrix \( \Sigma \). Then the optimal aggregate
of the experts’ estimates $Y_1, \ldots, Y_m$ is a generalized logarithmic pool:

$$
\ln \frac{Pr \left[ Y = 1 \mid \tau \right]}{Pr \left[ Y = 0 \mid \tau \right]} = \left( \frac{\sigma_1^2}{\mu_1} \ln \frac{Y_1}{1-Y_1}, \ldots, \frac{\sigma_m^2}{\mu_m} \ln \frac{Y_m}{1-Y_m} \right)^\top \Sigma^{-1} \mu.
$$

(We use $\tau_i$ in place of $\sigma_i$ to avoid collision with our notation for signal variances.) Note that this information structure is a kind of Gaussian mixture model (see e.g. Bishop (2006, §9.2)).

**Proof.** First, note that $Y_i := \mathbb{E} \left[ Y \mid \tau_i \right]$ is simply the probability that $Y = 1$ conditioned on expert $i$’s signal $\tau_i$. Let $\sigma_i^2$ be the variance of $\tau_i$ (so $\sigma_i^2$ is the $i$-th entry of the diagonal of $\Sigma$). Then $\tau_i$ is distributed with mean $\mu_i$ and variance $\sigma_i^2$ if $Y = 1$, and with mean $-\mu_i$ and variance $\sigma_i^2$ if $Y = 0$. Using the formula for a Gaussian PDF, we have that

$$
\frac{Pr \left[ Y = 1 \mid \tau_i \right]}{Pr \left[ Y = 0 \mid \tau_i \right]} = \frac{\exp \left( -\frac{(\tau_i - \mu_i)^2}{2\sigma_i^2} \right)}{\exp \left( -\frac{(\tau_i + \mu_i)^2}{2\sigma_i^2} \right)} = \exp \left( \frac{2\mu_i}{\sigma_i^2} \tau_i \right).
$$

Thus, we can write $Y_i$ in log-odds space as

$$
\ln \frac{Y_i}{1-Y_i} = \frac{2\mu_i}{\sigma_i^2} \tau_i. \tag{2.6}
$$

What about the optimal aggregate, $\mathbb{E} \left[ Y \mid Y_1, \ldots, Y_m \right]$? Note that we can recover the value of $\tau_i$ from the value of $Y_i$ (using the equation we just wrote down\textsuperscript{13}), so $\mathbb{E} \left[ Y \mid Y_1, \ldots, Y_m \right] = \mathbb{E} \left[ Y \mid \tau \right]$, where $\tau = (\tau_1, \ldots, \tau_m)$. Using the formula for the PDF of a multivariate Gaussian, we have:

$$
\frac{Pr \left[ Y = 1 \mid \tau \right]}{Pr \left[ Y = 0 \mid \tau \right]} = \frac{\exp \left( \frac{-1}{2} (\tau - \mu) \Sigma^{-1} (\tau - \mu) \right)}{\exp \left( \frac{-1}{2} (\tau + \mu) \Sigma^{-1} (\tau + \mu) \right)} = \exp (2\tau^\top \Sigma^{-1} \mu),
$$

where in the last step we used the expanded out $(\tau - \mu)^\top \Sigma^{-1} (\tau - \mu)$ and $(\tau + \mu)^\top \Sigma^{-1} (\tau + \mu)$ as sums of four terms and noticed cancellations. (We also used the fact that $\Sigma^{-1}$ is symmetric, so

\textsuperscript{13}Unless $\mu_i = 0$, but in that case $\tau_i$ provides no information about $Y$.}
\( \mu^T \Sigma^{-1} \tau = \tau^T \Sigma^{-1} \mu \). Therefore, we have

\[
\ln \frac{\Pr [Y = 1 | \tau]}{\Pr [Y = 0 | \tau]} = 2\tau^T \Sigma^{-1} \mu,
\]

which is a particular linear combination of the \( \tau_i \)'s. Combining this equation with Equation 2.6, we have

\[
\ln \frac{\Pr [Y = 1 | \tau]}{\Pr [Y = 0 | \tau]} = \left( \frac{\sigma_1^2}{\mu_1} \ln \frac{Y_1}{1 - Y_1}, \ldots, \frac{\sigma_m^2}{\mu_m} \ln \frac{Y_m}{1 - Y_m} \right)^T \Sigma^{-1} \mu.
\]

This is indeed a generalized logarithmic pool, as it is a particular linear combination of the values of \( \ln \frac{Y_i}{1 - Y_i} \).

We note that a different Bayesian justification of generalized logarithmic pooling was given by Babichenko and Garber (2021). Suppose that \( Y \) is uniformly either 0 or 1 and that there are \( k \) signals that are independent conditioned on \( Y \), each known by a subset of experts. If the posterior probability that \( Y = 1 \) conditioned on each signal individually can be recovered from the experts’ forecasts, then the optimal aggregate is a generalized linear pool.\(^{14}\) By contrast, the justification given by Claim 2.4.2 does not rely on such a “recovery” assumption, instead relying on properties of normal distributions.

\(^{14}\)More formally, let \( A \) be the \( m \times k \) matrix whose \((i, j)\)-entry is 1 if expert \( i \) knows signal \( j \). If there is a vector \( h \) such that \( A^T h = 1_k \), then taking a generalized logarithmic pool of the experts’ forecasts with weight vector \( h \) is optimal.
Chapter 3: Incentivizing precise forecasts

This chapter presents “Binary Scoring Rules that Incentivize Precision” (Neyman et al., 2021). It assumes background on proper scoring rules presented in Section 2.1.

Summary: Proper scoring rules – by definition – incentivize an expert to predict accurately (report their true belief). However, not all proper scoring rules equally incentivize precision. In this chapter, we will consider a model in which a rational expert can refine their belief by repeatedly paying a fixed cost, and is incentivized to do so by a proper scoring rule.

Specifically, our expert aims to predict the probability that a biased coin flipped tomorrow will land heads, and can flip the coin any number of times today at a cost of $c$ per flip. Our first main result defines an incentivization index for proper scoring rules,\(^1\) and proves that this index measures the expected error of the expert’s reported probability (where the number of flips today is chosen to maximize the expert’s expected payoff, i.e. score minus cost). Our second main result finds proper scoring rule that has the lowest (i.e. optimal) incentivization index among all proper scoring rules.

3.1 Introduction

The space of proper scoring rules is vast: as we saw in Section 2.1, there is (roughly speaking) one proper scoring rule for every strictly convex function. A principal who wishes to elicit a forecast from an expert must select one such scoring rule to use – but how? The quadratic and logarithmic scores are common choices because of their simplicity, but is there a more principled way to make this choice?

In many settings, the principal may care about the precision of the experts’ forecast. As a

\(^1\)We focus specifically on symmetric proper scoring rules for binary outcomes – more details below.
motivating example, consider the problem of guessing the probability that one of two competing advertisements will be clicked by a user. With zero effort, an expert could blindly guess that each is equally likely. But the expert can expend some cost in order to refine their forecast, i.e. make it more precise. For example, the expert could run a crowdsourcing experiment, paying users to see which link they would click. Any proper scoring rule will equally incentivize the expert to accurately report their resulting belief, but not all proper scoring rules equally incentivize the costly gathering of information.

And so, the motivating question of this work is: Which proper scoring rule most incentivizes the costly gathering of information?

We propose a simple model to formally measure the extent to which a proper scoring rule incentivizes costly refinement of the expert’s beliefs. Specifically, we consider a two-sided coin that comes up heads with probability $p$, and $p$ is drawn uniformly from $(0, 1)$ (we refer to $p$ as the bias of the coin). Tomorrow the coin will be flipped, and we ask the expert to guess the probability that it lands heads. Today, the expert can flip the coin (with bias $p$) any number of times, at cost $c$ per flip. While we choose this model for its mathematical simplicity, it captures examples like the one above surprisingly well: tomorrow, a user will be shown the two advertisements (clicking one). Today, the expert can run a crowdsourcing experiment and pay any number of workers $c$ to choose between the two ads. This simple model also captures weather forecasting using ensemble methods surprisingly well, and we expand on this connection in Section 3.1.1.

With this model in mind, consider the following two extreme forecasts: on the one hand, the expert could never flip the coin, and always output a guess of $1/2$. On the other, the expert could flip the coin infinitely many times to learn $p$ exactly, and output a guess of $p$. Note that both forecasts are accurate: the expert is truthfully reporting their belief, and that belief is correct given the observed flips. However, the latter forecast is more precise. All proper scoring rules incentivize the expert to accurately report their true forecast in both cases, but different scoring rules incentivize the expert to flip the coin a different number of times. More specifically, every proper scoring rule induces a different optimization problem for the expert, thereby leading them to
produce forecasts of different quality. In this model, the key question we answer is the following: *which scoring rules best incentivize the expert to produce a precise forecast?*

As our setting indicates, we will be considering *binary* proper scoring rules – meaning that there are two possible outcomes, which we will label “Yes” (heads) and “No” (tails). Further, we will restrict attention *symmetric* scoring rules – meaning that the scoring rule treats “Yes” and “No” symmetrically. That is, we are interested in proper scoring rules $s$ with the property that for all $x \in [0, 1]$, the score of an expert who assigns probability $x$ to Yes, if Yes happens, is equal to the score of an expert who assigns probability $x$ to No, if No happens:

$$
 s((x \text{ Yes}, 1 - x \text{ No}); \text{Yes}) = s((1 - x \text{ Yes}, x \text{ No}); \text{No}).
$$

This condition allows us to simplify notation: we will write $s(x)$ to mean the score of an expert who assigns probability $x$ to whichever outcome is realized. That is, our notation $s(x)$ refers to both $s((x \text{ Yes}, 1 - x \text{ No}); \text{Yes})$ and $s((1 - x \text{ Yes}, x \text{ No}); \text{No})$.

### 3.1.1 Motivation: Relationship to ensemble weather forecasts

A major shift occurred in the field of weather forecasting around the turn of the 21st century. In the previous century, weather forecasting was viewed as inherently deterministic: a forecasting model would take as input some initial conditions and use differential equations to simulate future states of the atmosphere. However, atmospheric conditions are never perfectly known: our observational equipment only gives us data about bits and pieces of the Earth’s atmosphere, while the rest of the picture must be completed with educated guesswork (see our discussion in Chapter 1). Additionally, the chaotic nature of atmospheric phenomena meant that even small inaccuracies in initial conditions would produce substantial forecast inaccuracies even a few days into the future.

Starting in the early 1990s and continuing into the early 2000s, there was a paradigm shift away from deterministic forecasts and toward *ensemble forecasts*. An ensemble is a collection of simulations based on different perturbations of a best guess about the initial conditions. Generally,
ensembles consist of five to 100 simulations. The results of these simulations are then used to
generate an overall forecast (Gneiting & Raftery, 2005).

The initial conditions used in ensemble models are typically chosen by “ensemble prediction
systems,” which attempt to sample initial conditions from a probability distribution based on real-
world uncertainty. Each simulation can be thought of as a sample from the probability distribution
over the future weather. For instance, if 60% of simulations predict rain in New York seven days
from now, then the ensemble model might estimate the chance of rain in New York seven days from
now at 60%, perhaps slightly adjusted based on a prior inferred from historical climate data (Gneit-
ing et al., 2007).

Each simulation can be thought of as a coin flip whose cost is measured in time, energy, or
computational resources. Each additional simulation has a constant cost. The final forecast for a
weather event is (roughly speaking) the fraction of simulations in which the event occurred. In this
way, ensemble forecasting strongly parallels our coin flip-based model of expert learning.

3.1.2 Our results

For a real number $\ell \geq 1$, let $\text{Error}_\ell^c(s)$ be the expected value of the $\ell$-th power of the absolute
error that a rational expert makes when incentivized with scoring rule $s$ with cost $c$ per flip. For
example, $\text{Error}_2^c(s)$ is the expected squared error of the expert (i.e. the squared difference between
the true bias of the coin and the expert’s report).\(^2\)

Our first main result is the existence of an incentivization index. Specifically, for every $\ell \geq 1$,
we give a closed-form index $\text{Ind}_\ell^c(s)$ with the following remarkable property: for all respectful
(see Definition 3.3.1) proper scoring rules $s_1$ and $s_2$, the inequality $\text{Ind}_\ell^c(s_1) < \text{Ind}_\ell^c(s_2)$ implies
the existence of a sufficiently small $c_0 > 0$ such that $\text{Error}_\ell^c(s_1) < \text{Error}_\ell^c(s_2)$ for all $c \leq c_0$
(Theorem 3.3.3). We formally introduce this index in Definition 3.3.2, but remark here that it is

\(^2\)In a sense, $\ell = 2$ is the most “internally consistent” choice: the value minimizing the expected squared distance
to the true bias is exactly the mean of the expert’s probability distribution over the bias, which is the number that $s$
elicits. However, it is reasonable to ask about other values of $\ell$ as well.
not a priori clear that such an index should exist at all, let alone that it should have a closed form.\footnote{Indeed, a priori it is possible that $\text{Error}^\ell_{0.1}(s_1) < \text{Error}^\ell_{0.1}(s_2)$, but $\text{Error}^\ell_{0.01}(s_1) > \text{Error}^\ell_{0.01}(s_2)$, and $\text{Error}^\ell_{0.001}(s_1) < \text{Error}^\ell_{0.001}(s_2)$, but $\text{Error}^\ell_{0.0001}(s_1) > \text{Error}^\ell_{0.0001}(s_2)$, and so on. The existence of an incentivization index rules out this possibility.}

With an index in hand, we can now pose a well-defined optimization problem for any given $\ell$: which proper scoring rule minimizes the incentivization index? Our second main result nails down this scoring rule precisely; we call it $s_{\ell, \text{Opt}}$ (see Theorem 3.4.1).

Some optimal rules $s_{\ell, \text{Opt}}$ have a particularly nice closed form (for example, as $\ell \to \infty$, the optimal rule pointwise converges to a polynomial), but many do not. We also prove, using techniques similar to the Weierstrass approximation theorem (Weierstrass, 1885), that each of these rules can be approximated by polynomial proper scoring rules whose incentivization indices approach the optimum.

Finally, beyond characterizing the optimal rules, the incentivization indices themselves allow for comparison among popular scoring rules, such as logarithmic, quadratic, and spherical $(s_{\text{sph}}(x) := x/\sqrt{x^2 + (1-x)^2})$. We plot the predictions made by our incentivization index (which provably binds only as $c \to 0$) for various values of $c$, and also confirm via simulation that the index is predictive for reasonable choices of $c$.

To summarize these results, we find that for all values of $\ell$, some relatively well-known proper scoring rule is very close to optimal. For small values of $\ell = 1, 2$, we find that the scoring rule $s_{\text{hs}}(x) := -\sqrt{1-x}/x$ – prominently used by Ben-David and Blais (2020) to prove a minimax theorem for randomized algorithms – is very close to optimal. For larger values of $\ell$ (e.g. $\ell = 4$), the log score is near-optimal. For every large values of $\ell$ (e.g. $\ell = 16$), the quadratic score is near-optimal. And for much larger values of $\ell$ (e.g. $\ell = 128$), the spherical scoring rule is near-optimal. See Table 3.1 and Figure 3.4 for more details.

Generally, these scoring rules are in decreasing order of how harshly they penalize assigning a very low probability to the eventual outcome (compare e.g. the log and quadratic scoring rules): the smaller the value of $\ell$, the more preferable it is to have a scoring rule that penalizes incorrect forecasts near the extremes.
3.1.3 Related work

To the best of our knowledge, Osband (1989) was the first to consider scoring rules as motivating the predictor to seek additional information about the distribution before reporting their belief. This direction is revisited in Clemen (2002), and has gained more attention recently (Hartline et al., 2020; Roughgarden & Schrijvers, 2017; Tsakas, 2019). While these works (and ours) each study the same phenomenon, there is little technical overlap and the models are distinct: each explores a different aspect of this broad agenda. For example, Roughgarden and Schrijvers (2017) consider the predictor’s incentive to outperform competing predictors (but there is no costly effort: the predictors’ beliefs are still exogenous). Hartline et al. (2020) (which is contemporaneous and independent of our work) is the most similar in motivation, but still has significant technical differences (beyond the two subsequent examples). On one hand, their model is more general than ours in that they consider multi-dimensional state spaces (rather than binary ones, in our model). On another hand, it is more restrictive in that they consider only two levels of effort (versus infinitely many, in our model).

Our work also fits into the broad category of principal-agent problems. For example, works such as Cai et al. (2015), Chen, Immorlica, et al. (2018), Chen and Zheng (2019), and Liu and Chen (2016) consider a learning principal who incentivizes agents to make costly effort and produce an accurate data point. Again, the models are fairly distinct, as these works focus on more sophisticated learning problems (e.g. regression), whereas we perform a more comprehensive dive into the problem of simply eliciting the (incentivized-to-be-precise) belief.

In summary, there is a sparse, but growing, body of work addressing the study of incentivizing effort in forming predictions, rather than just accuracy in reporting them. The above-referenced works pose various models to tackle different aspects of this agenda. In comparison, our model is arguably the simplest, and we develop a deep understanding of optimal scoring rules in this setting.
3.2 Our model and preliminaries

3.2.1 Modeling the expert’s behavior

We model the expert as Bayesian. Specifically, the expert initially believes the coin bias is uniformly distributed in $(0, 1)$. Today, the expert may flip the coin any number of times in order to gauge its true bias, and pays $c$ per flip.

**Fact 3.2.1** (Laplace’s rule of succession). After having flipped the coin $n$ times, and seen $k$ heads, the expert believes that the coin’s bias is $\frac{k+1}{n+2}$.

Once done flipping, the expert reports their belief about the coin’s bias. Tomorrow, the coin is flipped once, and the expert is scored with a proper scoring rule $s$ (known to the expert in advance).

It remains to define when the expert should stop flipping. Below, an *adaptive strategy* simply refers to a (possibly randomized) stopping rule for the expert, i.e. a rule that, given any number of past flips and the proper scoring rule $s$, tells the expert whether to stop or to flip the coin again. The payoff of an adaptive strategy is simply the expected score of an expert who follows that strategy, minus $c$ times the expected number of coin flips.

**Definition 3.2.2.** A globally-adaptive expert uses the payoff-maximizing adaptive strategy.

Nailing down the expert’s optimal behavior as a function of $c$ is quite unwieldy. Thus, we derive our characterizations up to $o(1)$ terms (as $c \to 0$). When $c$ is large, one may reasonably worry that these $o(1)$ terms render our theoretical results irrelevant. In Appendix A.6 we simulate the expert’s optimal behavior for large $c$, and confirm that our results hold qualitatively in this regime.

Finally, we define a natural measure of precision for the expert’s prediction.

**Definition 3.2.3.** The expected $\ell$-th power error associated with a proper scoring rule $s$ and cost $c$ is $\text{Error}_\ell^c(s) := \mathbb{E} \left[ |p - q|^\ell \right]$. The expectation is taken over the true bias $p$ of the coin, drawn

---

4By this, we mean the expert believes the coin would land heads with probability $\frac{k+1}{n+2}$, if it were flipped again.
uniformly from \((0, 1)\), and \(q\), the prediction of a globally-adaptive expert after flipping the coin as many times as they choose.

### 3.2.2 Scoring rule preliminaries

Our proofs will make use of fairly heavy single-variable analysis, and therefore will require making some assumptions on \(s\) such as differentiability, but also more technical ones. We will clearly state them when necessary, and confirm that all scoring rules of interest satisfy them. For these preliminaries, we need only assume that \(s\) is continuously differentiable so that everything which follows is well-defined.

Lemma 3.2.4 provides a characterization of proper (and weakly proper) scoring rules in our (binary, symmetric) setting.

**Lemma 3.2.4.** A continuously differentiable scoring rule \(s\) is weakly proper if and only if for all \(p \in (0, 1)\), \(ps'(p) = (1 - p)s'(1 - p)\) and \(s'(p) \geq 0\). It is (strictly) proper if and only if additionally \(s'(p) > 0\) almost everywhere\(^5\) in \((0, 1)\).

**Proof sketch.** If the expert believes that the true probability is \(p\) and reports \(x\), then their expected score is \(ps(x) + (1 - p)s(1 - x)\). For every \(p\), we want this expression to reach a maximum at \(x = p\). The derivative of the expression with respect to \(x\) is \(ps'(x) - (1 - p)s'(1 - x)\). Thus, for \(s\) to be proper, we should have that \(ps'(p) = (1 - p)s'(1 - p)\) for all \(p\). \(\square\)

We defer the full proof of Lemma 3.2.4 to Appendix A.1.

**Corollary 3.2.5.** Let \(s\) be strictly increasing almost everywhere (resp., nondecreasing everywhere) and continuously differentiable on \((0, \frac{1}{2}]\). Then \(s\) can be extended to a continuously differentiable proper (resp., weakly proper) scoring rule on \((0, 1)\) by defining \(s'(p) = \frac{1 - p}{p} s'(1 - p)\) for \(p \in (\frac{1}{2}, 1)\).

Put another way: every continuously differentiable proper scoring rule can be defined by first providing a strictly increasing function on \((0, \frac{1}{2}]\), and then extending it as in Corollary 3.2.5. For

---

\(^5\)Almost everywhere on \((0, 1)\) refers to the interval \((0, 1)\) except a set of measure zero.
example, consider the function \( s(x) = x \), which is strictly increasing on \((0, \frac{1}{2}]\). Defining \( s'(x) = \frac{1-x}{x} \cdot 1 = \frac{1}{x} - 1 \) for \( x \in [\frac{1}{2}, 1) \) results in \( s(x) = \ln x - x + 1 + \ln 2 \) (where \( 1 + \ln 2 \) is the necessary constant to make \( s \) continuous at \( x = \frac{1}{2} \)). Clearly \( s'(x) > 0 \) on \((0, 1)\) (as promised by Corollary 3.2.5), so we have just constructed a proper scoring rule:

\[
s(x) = \begin{cases} 
  x & \text{for } x \leq \frac{1}{2} \\
  \ln x - x + 1 + \ln 2 & \text{for } x \geq \frac{1}{2}
\end{cases}
\]

### 3.2.3 First steps towards understanding incentivization

We will be working with both \( s \) and its expected score function \( G_s(x) = xs(x) + (1-x)s(1-x) \) (see Section 2.1). Note that \( G_s \) is necessarily symmetric about \( \frac{1}{2} \) (this is a consequence of our assumption that \( s \) is symmetric). Additionally, the fact that \( s \) is continuously differentiable implies that \( G_s \) is as well.

**Remark 3.2.6.** For a weakly proper scoring rule \( s \), we have \( G'_s(x) = s(x) - s(1-x) \) and \( G''_s(x) = s'(x) + s'(1-x) = \frac{s'(x)}{1-x} \geq 0 \) on \((0, 1)\).

(The last equality can be inferred by using Lemma 3.2.4.) Lemma 3.2.7 observes how an expert’s expected score evolves with an additional flip of the coin.

**Lemma 3.2.7.** For a proper scoring rule \( s \), if the expert has already flipped the coin \( n \) times, seeing \( k \) heads, then their expected increase in score for exactly one additional flip is

\[
\frac{k + 1}{n + 2} G_s \left( \frac{k + 2}{n + 3} \right) + \frac{n - k + 1}{n + 2} G_s \left( \frac{k + 1}{n + 3} \right) - G_s \left( \frac{k + 1}{n + 2} \right).
\]

This quantity is positive.

**Proof.** This is a direct application of Laplace’s rule of succession (Fact 3.2.1). Currently, the expert believes the probability of heads to be \( \frac{k+1}{n+2} \). So their expected score if they stop flipping now is exactly \( G_s \left( \frac{k+1}{n+2} \right) \). If they flip once more and stop, then with probability \( \frac{k+1}{n+2} \) they will get a
heads, updating their belief to \( \frac{k+2}{n+3} \), and yielding expected score \( G_s \left( \frac{k+2}{n+3} \right) \). With probability \( \frac{n-k+1}{n+2} \) they will get a tails, updating their belief to \( \frac{k+1}{n+3} \) and yielding expected score \( G_s \left( \frac{k+1}{n+3} \right) \).

The fact that the quantity in Lemma 3.2.7 is positive follows directly from the fact that \( G_s \) is strictly convex.

Proper scoring rules remain proper when scaled by a positive affine transformation. This presents an issue for comparing scoring rules based on their incentivization properties. Because we are interested in incentivizing the expert to take costly actions, so the scale of a proper scoring rule will be relevant. For example, if \( s \) is proper, then so is \( 2s \), and \( 2s \) clearly does a better job of incentivizing the expert (since the quantity in Lemma 3.2.7 is larger by a factor of 2). As such, we will want to first normalize any scoring rules under consideration to be on the same scale.

A natural normalization is to consider two scoring rules to be on the same scale if they provide the same expected score to a perfect expert (one who knows the bias exactly). This is a natural choice because the expected score of a perfect expert is an upper bound on the expected payment that the principal must make. Further, as \( c \to 0 \), the expected score of a globally adaptive expert in fact approaches the expected score of a perfect expert (see Proposition A.1.2). Intuitively, this is because the number of flips approaches infinity as \( c \) approaches 0, so the expert’s forecast becomes perfectly precise in this limit.

The expected score of a perfect expert is \( \int_0^1 G_s(x)dx \), since a perfect expert has expected payoff \( G_s(x) \) if the coin has bias \( x \), and the coin’s bias is chosen uniformly from \([0, 1]\). For this reason, when evaluating a proper scoring rule in terms of its incentivization properties, we will scale it so that \( \int_0^1 G_s(x)dx = 1 \).

Scaling proper scoring rules in this way addresses one potential issue, but there is another as well: consider a proper scoring rule \( s \) satisfying \( \int_0^1 G_s(x)dx = 1 \). Then \( \tilde{s}(x) := 2s(x) - 1 \) is also proper and satisfies \( \int_0^1 G_{\tilde{s}}(x)dx = 1 \). However, \( \tilde{s} \) clearly does a better job incentivizing the expert (again, by the positivity of the quantity in Lemma 3.2.7). As such, we will also normalize \( s \) so that the score of a completely uninformed expert – one who flips the coin zero times and says \( \frac{1}{2} \) – is zero.
Definition 3.2.8. A weakly proper scoring rule \( s \) is normalized if \( \int_0^1 G_s(x) \, dx = 1, \) and \( s(1/2) = 0. \)

We often use the following equivalent condition (the proof of equivalence is given in Appendix A.1).

Claim 3.2.9. For a weakly proper scoring rule \( s \), we have

\[
\int_0^1 G_s(x) \, dx = s \left( \frac{1}{2} \right) + \int_{\frac{1}{2}}^1 (1 - x) s'(x) \, dx.
\]

Thus, \( s \) is normalized if and only if \( s(1/2) = 0 \) and \( \int_{\frac{1}{2}}^1 (1 - x) s'(x) \, dx = 1. \)

3.3 An incentivization index

This section presents our first main contribution: an incentivization index that characterizes the expert’s expected error. The main result of this section, Theorem 3.3.3, requires scoring rules to be analytically nice in a specific way. We term such scoring rules respectful.

Definition 3.3.1. A proper scoring rule \( s \) is respectful if:

1. \( G_s \) is strongly convex on \( (0, 1) \). That is, \( G_s''(x) \geq a \) on \( (0, 1) \) for some \( a > 0. \)
2. \( G_s''' \) is Riemann integrable on any closed sub-interval of \( (0, 1).^6 \)
3. There exists \( t > \frac{1}{4} \), and \( c_0 > 0 \) such that for all \( c \in (0, c_0) \): \( \left| G_s'''(x) \right| \leq \frac{1}{c^{0.16} \sqrt{x(1-x)}} G_s''(x) \) on \( [c^t, 1 - c^t].^7 \)

Recall that \( G_s \) is strictly convex for any (strictly) proper scoring rule, so strong convexity is a minor additional assumption. Likewise, the second condition is a minor “niceness” assumption.

We elaborate on the third condition in detail in Appendix A.2, and confirm that frequently used proper scoring rules are indeed respectful. We briefly note here that intuitively, the third condition asserts that \( G_s''' \) does not change too quickly (except possibly near zero and one) for small enough

\(^6\)Note this does not necessarily require \( G_s''' \) be defined on the entire \((0, 1)\), just that it is defined almost everywhere.

\(^7\)Except in places where \( G_s''' \) is undefined.
coin-flipping costs $c$. The particular choice of 0.16 is not special, and could be replaced with any constant less than $1/6$.

**Definition 3.3.2** (Incentivization Index). For $\ell \geq 1$, we define the $\ell$-th power incentivization index of a proper scoring rule $s$:

$$\text{Ind}_c^\ell(s) := \int_0^1 \left( \frac{x(1-x)}{G'_s(x)} \right)^{\ell/4} dx.$$  

**Theorem 3.3.3.** If $\mu_\ell := \frac{2^{\ell/2} \Gamma(\frac{\ell+1}{2})}{\sqrt{\pi}}$ is the $\ell$-th moment of the standard normal distribution, then

$$\lim_{c \to 0} c^{-\ell/4} \cdot \text{Error}_c^\ell(s) = \mu_\ell \cdot 2^{\ell/4} \cdot \text{Ind}_c^\ell(s).$$

Intuitively, the incentivization index captures the expert’s error as $c \to 0$. More formally, for any two respectful proper scoring rules $s_1, s_2$, $\text{Ind}_c^\ell(s_1) < \text{Ind}_c^\ell(s_2)$ implies that there exists a sufficiently small $c_0 > 0$ such that $\text{Error}_c^\ell(s_1) < \text{Error}_c^\ell(s_2)$ for all $c \leq c_0$. As previously mentioned, Theorem 3.3.3 says nothing about how big or small this $c_0$ might be, although simulations in Appendix A.6 confirm that it does not appear to be too small for typical scoring rules.

Theorem 3.3.3 says that the expert’s expected $\ell$-th power error is proportional to $c^{\ell/4}$. What is the intuition for this asymptotic relationship? Speaking informally, the key facts are that the number of times that the expert flips the coin is proportional to $c^{-1/2}$, and that the expected absolute error is proportional to $n^{-1/2}$, where $n$ is the number of flips. The first fact follows from the fact that the expected improvement in the expert’s score from an extra flip is proportional to $\frac{1}{n^2}$ (Claim 3.3.5 below); this quantity thus falls below the cost $c$ when $n$ is on the order of $c^{-1/2}$. The second fact follows from the fact that the expert’s estimate of the bias $p$ is (roughly) $\frac{1}{n}$ times a binomial random variable with $n$ trials and probability $p$ of success; the standard deviation of this quantity is proportional to $n^{-1/2}$.

The rest of this section is organized as follows. Sections 3.3.1 through 3.3.6 outline our proof of Theorem 3.3.3. The key steps are given as precisely-stated technical lemmas with mathematical
intuition alongside them, to illustrate where precision is needed for the proof to carry through. Complete proofs of these lemmas can be found in Appendix A.3. In Appendix A.2, we confirm that natural scoring rules are respectful (which is mostly a matter of validating the third condition in Definition 3.3.1).

### 3.3.1 Proof outline of Theorem 3.3.3

Below, we provide an executive overview of our approach. The concrete steps are separated out as formally-stated technical lemmas in the following sections, with proofs deferred to Appendix A.3. Before beginning, we highlight the main challenge: to prove Theorem 3.3.3, we need to capture the *precise asymptotics* of the expert’s expected error. Upper bounds can be easily shown via concentration inequalities; however, traditional lower bounds via anti-concentration results would simply state that the expected error tends to 0 as \( c \to 0 \) (which holds for every proper scoring rule, and doesn’t distinguish among them). So not only are we looking for two-sided bounds on the error, but we need to gauge the precise *rate* at which it approaches zero. Moreover, even obtaining the order of magnitude of the error as \( c \to 0 \), which turns out to be \( c^{-\ell/4} \), still does not suffice: we need to compute the exact coefficient of \( c^{-\ell/4} \). This difficulty motivates the need for the technical lemmas stated in this section to be very precise. Our outline is as follows:

- All of our analysis first considers a locally-adaptive expert, who flips the coin one additional time if and only if the expected increase in score from that single flip exceeds \( c \).
- Our first key step, Section 3.3.2, provides a loose asymptotic *lower bound* on the number of times an expert flips the coin, for all respectful \( s \).
- Our second key step, Section 3.3.3, provides a coupling of the expert’s flips across all possible true biases \( p \). This helps prove uniform convergence bounds over all \( p \) for the expert’s error: we can now define an unlikely “bad” event of overly-slow convergence without reference to \( p \).
- Our third key step, Section 3.3.4, provides tight bounds on the number of flips by a locally-adaptive expert, up to \((1 \pm o(1))\) factors. Note that the first three steps have not referenced
an error measure at all, and only discuss the expert’s behavior.

- Our fourth key step, Section 3.3.5, shows how to translate the bounds in Section 3.3.4 to
tight bounds on the error of a locally-adaptive expert, again up to \((1 \pm o(1))\) factors.

- Finally our last step, Section 3.3.6, shows that the globally-adaptive expert behaves nearly-
identically to the locally-adaptive expert, up to an additional \(o(1)\) factor of flips.

We now proceed to formally state the main steps along this outline, recalling that the first
several steps consider a locally-adaptive expert, whose definition is restated formally below:

**Definition 3.3.4 (Locally-Adaptive Expert).** *The locally-adaptive expert flips one more time if and
only if making a single additional coin flip (and then stopping) increases their expected payoff.*

### 3.3.2 Step one: Lower bounding the expert’s number of flips

We begin by tying the expert’s expected marginal score from one additional flip to the second
derivative of the expected score function, \(G_s''\). Below, \(Q(n)\) denotes the random variable which is
the expert’s belief after \(n\) flips. The important takeaway from Claim 3.3.5 is that for fixed \(n\), the
expert’s expected belief as a function of \(Q(n)\) changes (roughly) as \(Q(n) \cdot (1 - Q(n)) \cdot G_s''(Q(n))\)
– this takeaway will appear in later sections.

**Claim 3.3.5.** Let \(\Delta_{n+1}(q) := \mathbb{E}[G_s(Q(n + 1)) \mid Q(n) = q] - G_s(q)\) be the expected increase in the
expert’s score (not counting the paid cost \(c\)) from the \((n + 1)\)th flip of the coin, given current belief
\(Q(n) = q\). Then there exist \(c_1, c_2 \in [q - 1/n, q + 1/n]\) such that:

\[
\Delta_{n+1} = \frac{q \cdot (1 - q)}{2(n + 3)^2} (q \cdot G_s''(c_1) + (1 - q) \cdot G_s''(c_2))
\]

Recalling that the locally-adaptive expert decides to flip the coin for the \((n+1)^{th}\) time if and only
if \(\Delta_{n+1} \geq c\), and assuming that \(G_s''\) is bounded away from zero (Condition 1 in Definition 3.3.1),
we arrive at a simple lower bound on the number of coin flips.

**Claim 3.3.6.** For all \(s\) such that \(G_s''\) is bounded away from zero, there exists \(\alpha, c_0\) such that the
expert is guaranteed to flip the coin at least \(\frac{1}{\alpha c_0^{1/3}}\) times for all \(c \leq c_0\) (no matter the true bias).
Using basic concentration inequalities, Claim 3.3.6 immediately implies an asymptotic upper bound on the expert’s error. Recall, however, that we need a two-sided bound, and moreover that we need precise asymptotics of the error. Still, Claim 3.3.6 is the first step towards this.

### 3.3.3 Step two: Ruling out irregular coin-flipping trajectories

The expert’s coin-flipping behavior depends on \( Q(n) \), which depends on the fraction of realized coin flips which are heads, which itself depend on the coin’s true bias \( p \). Note, of course, that \( Q(n) \to p \) as \( n \to \infty \). If instead we had that \( Q(n) = p \) exactly, we could leverage Claim 3.3.5 to better understand the number of flips as a function of \( p \). Unfortunately, \( Q(n) \) will not equal \( p \) exactly, and it is even possible to have \( Q(n) \) far from \( p \), albeit with low probability.

The challenge, then, is then how to handle these low-probability events, and importantly how to do so uniformly over \( p \). To this end, we consider the following coupling of coin-flipping processes over all possible biases. Specifically, rather than first drawing bias \( p \) and then flipping coins with bias \( p \), we use the following identically distributed procedure:

(1) Generate an infinite sequence \( r_1, r_2, \ldots \) of uniformly random numbers in \([0, 1]\).

(2) Choose \( p \) uniformly at random from \([0, 1]\).

(3) For each \( n \), coin \( n \) comes up heads if and only if \( r_n \leq p \).

Under this sampling procedure, \( Q_p(n) := \frac{h_p(n)+1}{n+2} \) is the expert’s estimate after flipping \( n \) coins, where \( h_p(n) \) is the number of heads in the first \( n \) flips, if \( p \) is the value chosen in step (2).

With this procedure, we can now define a single bad event uniformly over all \( p \). Intuitively, \( \Omega_N \) holds when, no matter what \( p \) is chosen in step (2), the expert’s Bayesian estimate of \( p \) never strays too far from \( p \) after \( N \) flips. More formally, the complement of \( \Omega_N \) is our single bad event:

\[
\bar{\Omega}_N := \bigcup_{n=N}^{\infty} \bigcup_{j=1}^{n-1} \left\{ Q_{j/n}(n) - \frac{j}{n} > \frac{\sqrt{j(n-j)}}{2n^{1.49}} \right\}.
\]

The expression on the right-hand side of the inequality can be rewritten as \( \sqrt{\frac{j(1-j)}{n}} \cdot \frac{n^{0.01}}{2} \), where
the radical term gives the order of the expected difference between $Q_{j/n}(n)$ and $\frac{j}{n}$. So intuitively, $\Omega_N$ holds unless the actual difference between $Q_{j/n}(n)$ and $\frac{j}{n}$ far exceeds its expected value.

We have defined $\Omega_N$ so that, on the one hand, our subsequent analysis becomes tractable when $\Omega_N$ holds, and on the other hand, $\Omega_N$ fails to hold with probability small enough that our asymptotic results are not affected. Below, Claim 3.3.7 gives the property we desire from $\Omega_N$, and Claim 3.3.8 shows that $\overline{\Omega_N}$ is unlikely. The key takeaway from Claim 3.3.7 is that when $\Omega_N$ holds, the expert’s prediction is close to $p$ for all $n \geq N$ and $p \in (0, 1)$ and this closeness shrinks with $n$.

**Claim 3.3.7.** The exists a sufficiently large $N_0$ such that for all $N \geq N_0$: if $\Omega_N$ holds, then

$$|Q_p(n) - p| \leq \frac{\sqrt{p(1-p)}}{n^{49}} \text{ for all } n \geq N \text{ and } p \in [1/n, 1 - 1/n].$$

**Claim 3.3.8.**

$$\Pr[\overline{\Omega_N}] = O\left(e^{-N^{0.01}}\right).$$

While it is trivial to see that $Q_p(n)$ approaches $p$ as $n \to \infty$, we reiterate that Claims 3.3.7 and 3.3.8 guarantee quantitatively that: (a) when $\Omega_N$ holds, $|Q_p(n) - p|$ shrinks with $n$, (b) the probability that $\Omega_N$ fails shrinks exponentially fast in $N$, and (c) both previous bounds are uniform over $p$.

### 3.3.4 Step three: Tightly bounding the expert’s number of flips

We now nail down the precise asymptotics of the number of the expert’s flips as a function of the true bias $p$. This becomes significantly more tractable after assuming $\Omega_N$ holds. Below, the random variable $n_{\text{stop}}$ denotes the number of flips that a locally-adaptive expert chooses to make.

**Proposition 3.3.9.** Assume that $\Omega_N$ holds for some $N$, and let $t$ be as in Definition 3.3.1. There exists a constant $\gamma$ and cost $c_0 > 0$ such that for all $c \leq c_0$ and all $p \in [2c^t, 1 - 2c^t]$, we have

$$\sqrt{\frac{p(1-p)G''_s(p)}{2c}}(1 - \gamma c^{1/300}) \leq n_{\text{stop}} \leq \sqrt{\frac{p(1-p)G''_s(p)}{2c}}(1 + \gamma c^{1/300}).$$
Proposition 3.3.9 has two key aspects. First, the upper and lower bounds on $n_{\text{stop}}$ match up to a $1 \pm o(1)$ factor. Second, the $o(1)$ term is independent of $p$. To get intuition for why $n_{\text{stop}} \approx \sqrt{\frac{p(1-p)G''_s(p)}{2c}}$, recall that Claim 3.3.5 shows after $n$ flips, the expected marginal gain is $\Delta_{n+1} \approx \frac{p(1-p)}{2n^2}G''_s(p)$. This quantity first falls below $c$, the cost per flip, after $n = \sqrt{\frac{p(1-p)G''_s(p)}{2c}}$ flips.

### 3.3.5 Step four: Translating number-of-flips bounds to error bounds

Having pinned down $n_{\text{stop}}$ quite precisely, we will now obtain a tight bound on the error of the locally-adaptive expert’s reported prediction. By contrast, the previous three steps performed an analysis of the locally-adaptive expert’s coin-flipping behavior, which does not depend on the choice of error metric. Lemma 3.3.10 below is a formal statement of the main step of this process, which nails down the asymptotics of the error conditioned on $\Omega_N$. Below, $\text{Err}_c(p)$ denotes a random variable equal to the locally-adaptive expert’s error (i.e. absolute difference between their report and the true bias) when the cost is $c$ and the true bias is $p$ (and the scoring rule $s$ is implicit).

**Lemma 3.3.10.** Let $\ell \geq 1$ and $\mu_\ell := \frac{2^{\ell/2} \Gamma(\ell/4)}{\sqrt{\pi}}$ be the $\ell^{\text{th}}$ moment of a standard Gaussian. Let $N = \frac{1}{ac^{1/3}}$ (so $N$ is implicitly a function of $c$). For all $p \in [2c^\ell, 1 - 2c^\ell]$ we have

$$(1 - o(1)) \cdot \mu_\ell \cdot \left(\frac{2p(1-p)}{G''_s(p)}\right)^{\ell/4} \leq e^{-\ell/4} \cdot \mathbb{E} \left[(\text{Err}_c(p))^\ell \mid \Omega_N\right] \leq (1 + o(1)) \cdot \mu_\ell \cdot \left(\frac{2p(1-p)}{G''_s(p)}\right)^{\ell/4}$$

where the $o(1)$ term is a function of $c$ (but not $p$) that approaches zero as $c$ approaches zero.

Lemma 3.3.10 is the key, but far from only, step in translating Proposition 3.3.9 to tight bounds on the locally-adaptive expert’s error. Intuitively, it states that the value of the expert’s error will be, up to a $1 \pm o(1)$ factor, consistent with what one would expect from using a quantitative central limit theorem in conjunction with the bound on $n_{\text{stop}}$ from Proposition 3.3.9.

### 3.3.6 Step five: From locally-adaptive to globally-adaptive behavior

Finally, we extend our previous analysis from locally-adaptive to globally-adaptive experts. In particular, for a scoring rule that gives a finite expected score to a perfect expert, we prove that the
globally-adaptive expert does not flip significantly more than a locally-adaptive expert would, and therefore their achieved errors are equal up to a $1 \pm o(1)$ factor. Below, the random variable $n_g$ denotes the number of flips by the globally-adaptive expert.

**Lemma 3.3.11.** Assume $s$ is respectful and normalizable (i.e. $\int_0^1 G_s(x)dx < \infty$). Let $\gamma$ be as in Proposition 3.3.9. There exists a $c_0 > 0$, such that for all $c \leq c_0$: If $\Omega_{n_{\text{stop}}}$ holds and $4c' \leq Q(n_{\text{stop}}) \leq 1 - 4c'$, then

$$n_{\text{stop}} \leq n_g \leq (1 + 6\gamma c^{1/300})n_{\text{stop}}.$$  

Lemma 3.3.11 is the key step in this portion of the analysis. The remaining work is to bound the impact of negligible events (such as $\Omega_{n_{\text{stop}}}$ failing, or $Q(n_{\text{stop}})$ being extremely close to 0 or 1) on our analysis. This completes our outline of the proof of Theorem 3.3.3.

### 3.4 Finding optimal scoring rules

Now that we have shown that the incentivization index characterizes how well any respectful scoring rule incentivizes a globally-adaptive expert to minimize error, we have a well-defined optimization problem: which normalized proper scoring rule has the lowest incentivization index (and therefore minimizes the expert’s expected error)? Recall the following necessary and sufficient set of conditions for a continuously differentiable and normalized scoring rule $s$ to be weakly proper:\(^8\)

- (Lemma 3.2.4) For all $x \in (0, 1)$, $xs'(x) = (1 - x)s'(1 - x)$ and $s'(x) \geq 0$.
- (Definition 3.2.8, Claim 3.2.9) $s(1/2) = 0$ and $\int_1^1 (1 - x) s'(x)dx = 1$.

So our goal is just to find the scoring rule which satisfies these constraints and minimizes the incentivization index:

$$\text{Ind}^\ell(s) = \int_0^1 \left( \frac{x(1 - x)}{G_s'(x)} \right)^{t/4} dx = \int_0^1 \left( \frac{x(1 - x)^2}{s'(x)} \right)^{t/4} dx.$$  

\(^8\)Including weakly proper scoring rules in our optimization domain makes the analysis simpler. The optimal scoring rules are in fact strictly proper.
(The last step follows from Remark 3.2.6.) The main result of this section is the following theorem, whose proof we defer to Appendix A.4.

**Theorem 3.4.1.** The unique continuously differentiable normalized proper scoring rule which minimizes \( \text{Ind}^\ell(s) \) is:

\[
s_{\ell,\text{Opt}}(x) := \begin{cases} 
\kappa \int_{\frac{1}{2}}^{x} (t^\ell - 8(1 - t)^2 t^{(\ell + 4)})^{1/4} \, dt & x \leq \frac{1}{2} \\
\kappa \int_{\frac{1}{2}}^{x} (t^\ell (1 - t)^2 t^{(\ell + 4)})^{1/4} \, dt & x \geq \frac{1}{2}.
\end{cases}
\]

While \( s_{\ell,\text{Opt}} \) is certainly challenging to parse, importantly it has a closed form, and can thus be numerically evaluated. Section 3.5.2 contains several plots of these scoring rules, alongside traditional ones. Below we give an overview of our proof of Theorem 3.4.1 (the full details of the proof can be found in Appendix A.4).

### 3.4.1 Proof overview of Theorem 3.4.1

As shown in Corollary 3.2.5, the equation \( x s'(x) = (1 - x) s'(1 - x) \) lets us extend \( s \) uniquely in a continuous manner to \((0, 1)\) if we know \( s \) on \([\frac{1}{2}, 1)\). Thus, we can simply consider \( s \) on \([\frac{1}{2}, 1)\). For \( s' \) to be nonnegative everywhere, it suffices for it to be nonnegative on \([\frac{1}{2}, 1)\), because of the relation \( x s'(x) = (1 - x) s'(1 - x) \). Also, observe that the integrand in the definition of the incentivization index is symmetric about \( \frac{1}{2} \); this is clear from the fact that \( G''_s \) is symmetric about \( \frac{1}{2} \). This means that

\[
\text{Ind}^\ell(s) = 2 \int_{\frac{1}{2}}^{1} \left( \frac{x(1 - x)^2}{s'(x)} \right)^{\ell/4} \, dx.
\]

Thus, our question can be phrased as follows: find the continuously differentiable function \( s : [\frac{1}{2}, 1) \rightarrow \mathbb{R} \) satisfying \( s\left(\frac{1}{2}\right) = 0, s'(x) \geq 0 \), and \( \int_{\frac{1}{2}}^{1} (1 - x) s'(x) \, dx = 1 \), that minimizes

\[
\int_{\frac{1}{2}}^{1} \left( \frac{x(1 - x)^2}{s'(x)} \right)^{\ell/4} \, dx.
\]

From this point, our problem is simply a continuous mathematical program. It is not obvious that the program should admit a closed-form solution, but it does. We defer all details to Ap-
pendix A.4, and just briefly note that we can formulate the problem exclusively as a function of \( s' \), and then uniquely reconstruct \( s \) using \( s \left( \frac{1}{2} \right) = 0 \). Once we have done this, we can take a Lagrangian relaxation by putting a multiplier on the constraint \( \int_{\frac{1}{2}}^{1} (1 - x)s'(x)dx = 1 \), and hope that the solution to the relaxation is continuous and satisfies \( s'(x) \geq 0 \). While this is not guaranteed to succeed, this method does in fact nail down the optimum. Below are the main technical lemmas that yield Theorem 3.4.1. Note that \( h \) plays the role of \( s' \) in these lemmas.

**Lemma 3.4.2.** For any \( \ell \geq 1 \), a function \( h : [\frac{1}{2}, 1) \rightarrow \mathbb{R}_{\geq 0} \) satisfying \( \int_{\frac{1}{2}}^{1} (1 - x)h(x)dx = 1 \) that minimizes \( \int_{\frac{1}{2}}^{1} \left( \frac{x(1-x)^2}{h(x)} \right)^{\ell/4} dx \) is \( \tilde{h}_\ell(x) = \kappa_\ell \left( x^\ell (1-x)^{2\ell-4} \right)^{1/(\ell+4)} \), where \( \kappa_\ell = \left( \int_{\frac{1}{2}}^{1} (x(1-x)^3)^{\ell/(\ell+4)} dx \right)^{-1} \).

(Note that \( \kappa_\ell \) is simply a normalization constant, so as to make \( \int_{\frac{1}{2}}^{1} (1 - x)h(x)dx \) equal 1.)

**Corollary 3.4.3.** The unique continuous function \( h : [\frac{1}{2}, 1) \rightarrow \mathbb{R}_{\geq 0} \) satisfying \( \int_{\frac{1}{2}}^{1} (1 - x)h(x)dx = 1 \) that minimizes \( \int_{\frac{1}{2}}^{1} \left( \frac{x(1-x)^2}{h(x)} \right)^{\ell/4} dx \) is \( \tilde{h}_\ell \).

Theorem 3.4.1 then follows from Corollary 3.4.3 by setting \( s_{\ell, \text{Opt}} \) to the integral of \( \tilde{h}_\ell \) on \([1/2, 1)\), and extending it to \((0, 1/2)\) via Corollary 3.2.5. For some choices of \( \ell \), the particular scoring rule \( s_{\ell, \text{Opt}} \) has an interesting closed form (see Section 3.5), but this is not true for all \( \ell \). Even in cases where the particular closed form is not illuminating, the fact that \( s_{\ell, \text{Opt}} \) even exists is already interesting, and the fact that Theorem 3.4.1 nails down the closed form allows us to compare other scoring rules to the optimum. We conclude with a remark, confirming that our analysis in Section 3.3 indeed is meaningful for all derived optimal scoring rules.

**Remark 3.4.4.** For every \( \ell \in [1, 8] \), \( s_{\ell, \text{Opt}} \) is respectful. For \( \ell > 8 \), and all \( \varepsilon > 0 \), there exists a respectful normalized proper scoring rule \( s \) such that \( |s(x) - s_{\ell, \text{Opt}}(x)| \leq \varepsilon \) for all \( x \in (0, 1) \), with \( \text{Ind}^\ell(s) \leq \text{Ind}^\ell(s_{\ell, \text{Opt}}) + \varepsilon \).

We give a proof for \( \ell \in [1, 8] \) in Appendix A.3. Meanwhile, the proof for \( \ell > 8 \) follows from the proof of Theorem 3.6.1 in Appendix A.5.\(^9\)

The following corollary follows from Remark 3.4.4.

\(^9\)More specifically, the scoring rules \( s_\varepsilon \) defined in the proof of Theorem 3.6.1 uniformly converge to \( s_{\ell, \text{Opt}} \).
Corollary 3.4.5. For $\ell \geq 1$, let

$$\text{Error}^\ell_{\text{Opt}} := \inf_s \lim_{c \to 0} c^{-\ell/4} \cdot \text{Error}^\ell_c(s)$$

where $s$ ranges over all normalized, respectful, continuously differentiable proper scoring rules. Let

$$\text{Ind}^\ell_{\text{Opt}} := \inf_s \text{Ind}^\ell(s)$$

where $s$ ranges over all normalized, continuously differentiable proper scoring rules. Then:

1. $\text{Error}^\ell_{\text{Opt}} = \mu_\ell \cdot 2^{\ell/4} \cdot \text{Ind}^\ell_{\text{Opt}}$.
2. For $1 \leq \ell \leq 8$, the first infimum is uniquely achieved by $s = s^\ell_{\text{Opt}}$.
3. For $\ell > 8$, no (respectful) function achieves the first infimum, but the infimum is reached in the limit by uniform approximations of $s^\ell_{\text{Opt}}$ (which are normalized, respectful, and continuously differentiable).

3.5 Comparing scoring rules

In this section we compare various scoring rules by their incentivization indices, for various values of $\ell$. Of particular interest are the values $\ell = 1$ (expected absolute error), $\ell = 2$ (expected squared error), and the limit as $\ell \to \infty$ (which penalizes bigger errors “infinitely more” than smaller ones, so this regime corresponds to minimizing the probability of being very far off).

3.5.1 Optimal scoring rules for particular values of $\ell$

We begin by noting some values of $\ell$ for which the function $s^\ell_{\text{Opt}}$ takes a nice closed form. $\ell = 1$ happens to not be one such value. For $\ell = 2, 4, 8$, the functions $s^\ell_{\text{Opt}}$ can be written in terms of elementary functions on the entire interval $(0, 1)$. For $\ell = 2$, the closed form on $(1/2, 1)$ is a polynomial, although its extension via Corollary 3.2.5 to $(0, 1/2)$ is not. For $\ell = 8$, the closed form on both $(0, 1/2)$ and $(1/2, 1)$ is a polynomial, although they are different. Interestingly, as $\ell \to \infty$, 71
the closed form converges pointwise to a single polynomial. Specifically, for these values of $\ell$:

**For $\ell = 2$:** On $[\frac{1}{2}, 1)$, we have

$$s_{2, \text{Opt}}(x) = k_2 \int_{\frac{1}{2}}^{x} t^{2/3} \, dt = \frac{3}{5} k_2 \left( x^{5/3} - \left( \frac{1}{2} \right)^{5/3} \right).$$

**For $\ell = 8$:** On $(0, \frac{1}{2}]$, we have

$$s_{8, \text{Opt}}(x) = k_8 \int_{\frac{1}{2}}^{x} (1-t)^{5/3} \, dt = \frac{3}{8} k_8 \left( \frac{1}{2} \right)^{8/3} - (1-x)^{8/3}$$

and on $[\frac{1}{2}, 1)$, we have

$$s_{8, \text{Opt}}(x) = k_8 \int_{\frac{1}{2}}^{x} (t^{2/3} - t^{5/3}) \, dt = k_8 \left( \frac{3}{5} \left( x^{5/3} - \left( \frac{1}{2} \right)^{5/3} \right) - \frac{3}{8} \left( x^{8/3} - \left( \frac{1}{2} \right)^{8/3} \right) \right).$$

Finally, as $\ell \to \infty$: on the entire interval $(0, 1)$, $s_{\ell, \text{Opt}}$ pointwise converges to

$$\lim_{\ell \to \infty} k_\ell \cdot \int_{\frac{1}{2}}^{x} t(1-t)^2 \, dt = \frac{320}{3} \left( \frac{1}{4} x^4 - \frac{2}{3} x^3 + \frac{1}{2} x^2 - \frac{11}{192} \right) = \frac{5}{9} (48 x^4 - 128 x^3 + 96 x^2 - 11).$$

We refer to this last rule as $s_{\infty, \text{Opt}}$. Intuitively, minimizing the expected value of error raised to a power that approaches infinity penalizes any error infinitely more than an even slightly smaller error. Put otherwise, this metric judges a scoring rule by the maximum (over $p \in (0, 1)$) of the spread of the distribution of expert error. The scoring rule $s_{\infty, \text{Opt}}$ has a very special property, which is that the quantity $\frac{x(1-x)}{G'_{s_{\infty, \text{Opt}}}(x)} = \frac{x(1-x)^2}{s_{s_{\infty, \text{Opt}}}(x)}$, which appears in the incentivization index, is a constant regardless of $x$. This means that, in the limit as $c \to \infty$, the distribution of the expert’s error is the same regardless of $p$. It makes intuitive sense that making the spread of the distribution of expert error uniform over all $p$ also minimizes the maximum of these spreads, which explains why $s_{\infty, \text{Opt}}$ has this interesting property.

As some of these rules are not infinitely differentiable, a natural question to ask is: what infinitely differentiable normalized function minimizes $\text{Ind}^\ell$? While (as we have shown by virtue
of \( s_{\ell, \text{Opt}} \) being the unique minimizer) achieving an incentivization index equal to \( \text{Ind}^{\ell}(s_{\ell, \text{Opt}}) \) with an infinitely differentiable scoring rule is impossible, it turns out that it is possible to get arbitrarily close – and in fact it is possible to get arbitrarily close with polynomial scoring rules. The main idea of the proof is to use the Weierstrass approximation theorem to approximate \( s_{\ell, \text{Opt}} \) with polynomials. See Section 3.6 for a full proof.

### 3.5.2 Plots of some relevant scoring rules

Figure 3.1 plots \( s_{\ell, \text{Opt}} \) for \( \ell = 1, 2, 8, \infty \). It demonstrates that optimal scoring rules for larger values of \( \ell \) are “flatter,” choosing to sacrifice rewarding precision near 0 and 1, in favor of rewarding precision closer to \( \frac{1}{2} \). An expert who is scored by \( s_{\infty, \text{Opt}} \) does not particularly care to distinguish between 98% and 99% probabilities, since the scoring rule is basically flat near the tails; this is not the case for \( s_{1, \text{Opt}} \). Conversely, because \( s_{\infty, \text{Opt}} \) is steeper than \( s_{1, \text{Opt}} \) near \( \frac{1}{2} \), an expert cares more about differentiating between a 50% and a 51% chance if scored with \( s_{\infty, \text{Opt}} \) than with \( s_{1, \text{Opt}} \).

![Figure 3.1: Plots of \( s_{\ell, \text{Opt}} \) for \( \ell = 1, 2, 8, \infty \).](image)

Another, perhaps more enlightening way to view these scoring rules is through the quantity \( \sqrt{\frac{s(x)(1-x)}{G^2_s(x)}} \). Up to a constant factor depending on the cost \( c \) of a flip, this is the variance of the normal distribution that approximates the distribution of the expert’s response if the true bias of the coin is \( x \) (for small \( c \)) – or, put otherwise, the expected squared error. Figure 3.2 plots this quantity for a variety of the scoring rules we have discussed.
Figure 3.2: Plots of \( \sqrt{\frac{x(1-x)}{G'_s(x)}} \) for \( s = s_{\ell, \text{Opt}} \) for a variety of values of \( \ell \). Up to a constant factor, this quantity is the variance of the normal distribution that approximates the distribution of the expert’s response if the true bias of the coin is \( x \) (for small \( c \)).

Figure 3.2 reinforces our previous point: optimal rules for small value of \( \ell \) result in very small errors near 0 and 1, but relatively large errors in the middle. In Section 3.5.1, we discussed in brief why it makes sense that the value of \( \frac{x(1-x)}{G'_s(x)} \) is constant for the scoring rule \( s_{\infty, \text{Opt}} \). We see this in Figure 3.2: since our normalization constraints force a trade-off between minimizing expert error for different values of the coin’s bias \( p \), the scoring rule whose error is independent of \( p \) will have the minimum possible value of the maximum error over all \( p \).

Finally, Figure 3.3 is similar to Figure 3.2, except that it also includes the (normalized) logarithmic, quadratic, and spherical scoring rules. In Section 3.5.3, we noted that the logarithmic rule is near-optimal for \( \ell \approx 4 \), the quadratic rule for \( \ell \approx 16 \), and the spherical rule for even larger \( \ell \). Figure 3.3 helps provide some intuition for this fact: the log scoring rule is similar in shape to \( s_{4, \text{Opt}} \) and similarly for the quadratic scoring rule and \( s_{16, \text{Opt}} \), and for the spherical scoring rule and \( s_{\infty, \text{Opt}} \).

### 3.5.3 Comparison of incentivization indices of scoring rules

We compare commonly studied scoring rules such as quadratic, logarithmic, and spherical, and in this section we refer to their normalizations as \( s_{\text{quad}}, s_{\text{log}}, s_{\text{sph}} \), respectively. Additionally
Figure 3.3: Plots of $\sqrt{\frac{s(1-x)}{G_s'(x)}}$ for $s = s_{\log}$, $s_{\text{quad}}$, and $s_{\text{sph}}$, as well as some optimal rules. Up to a constant factor, this quantity is the variance of the normal distribution that approximates the distribution of the expert’s response if the true bias of the coin is $x$ (for small $c$).

we include for comparison $s_{\text{hs}}$, which is the normalization of the $hs$ scoring rule: $-\sqrt{\frac{1-x}{x}}$. This scoring rule was prominently used by Ben-David and Blais (2020) to prove their minimax theorem for randomized algorithms.

Table 3.1 states $\text{Ind}_\ell(s)$ for various scoring rules $s$ (the lower the better). It lets us compare the performance of various scoring rules by our metric for any particular value of $\ell$. However, as one can see, $\text{Ind}_\ell$ decreases as $\ell$ increases. This makes sense, since $\text{Ind}_\ell$ measures the expected $\ell$-th power of error. For this reason, if we wish to describe how a given scoring rule performs over a range of values of $\ell$, we need to normalize these values. We do so by taking the $\ell$-th root and dividing these values by the $\ell$-th root of the optimal (smallest) index (and take the inverse so that larger numbers are better). This gives us the following measure of scoring rule precision, which makes sense across different values of $\ell$:

$$\left(\frac{\text{Ind}_\ell(s_{\ell,\text{opt}})}{\text{Ind}_\ell(g)}\right)^{1/\ell}.$$

Figure 3.4, which evaluates this expression for a selection of proper scoring rules and values of $\ell$, reveals some interesting patterns. Of the $hs$, logarithmic, quadratic, and spherical scoring
rules, the hs scoring rule is the best one for the smallest values of $\ell$ and is in fact near-optimal for $\ell = 2$. The logarithmic rule is the best one for somewhat larger values of $\ell$ and is in near-optimal for $\ell \approx 4$. For larger values of $\ell$, the quadratic scoring rule is best, and is near-optimal for $\ell \approx 16$. For even larger values of $\ell$, the spherical scoring rule is the best of the four. This pattern suggests that for any given proper scoring rule there is a trade-off between incentivizing precision at low and at high values of $\ell$; it would be interesting to explore this further. Figure 3.5 is a continuous version of Figure 3.4. It shows how the numbers in the table vary over a continuum of values of $\ell$.

Table 3.1: A table of incentivization index values for various proper scoring rules under various error measures.

<table>
<thead>
<tr>
<th>Scoring Rule</th>
<th>$\ell = 1$</th>
<th>$\ell = 2$</th>
<th>$\ell = 4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_{\text{log}}$</td>
<td>0.260</td>
<td>0.0732</td>
<td>0.00644</td>
</tr>
<tr>
<td>$s_{\text{quad}}$</td>
<td>0.279</td>
<td>0.0802</td>
<td>0.00694</td>
</tr>
<tr>
<td>$s_{\text{sph}}$</td>
<td>0.296</td>
<td>0.0889</td>
<td>0.00819</td>
</tr>
<tr>
<td>$s_{\text{hs}}$</td>
<td>0.255</td>
<td>0.0723</td>
<td>0.00658</td>
</tr>
<tr>
<td>$s_{1,\text{Opt}}$</td>
<td>0.253</td>
<td>0.0728</td>
<td>0.00719</td>
</tr>
<tr>
<td>$s_{2,\text{Opt}}$</td>
<td>0.255</td>
<td>0.0718</td>
<td>0.00661</td>
</tr>
<tr>
<td>$s_{4,\text{Opt}}$</td>
<td>0.261</td>
<td>0.0732</td>
<td>0.00639</td>
</tr>
<tr>
<td>$s_{\infty,\text{Opt}}$</td>
<td>0.311</td>
<td>0.0968</td>
<td>0.00974</td>
</tr>
</tbody>
</table>

Figure 3.4: Incentivization indices of various proper scoring rules relative to the optimal proper scoring rule, for a variety of error measures. The values in the table are $\left( \frac{\text{Ind}^\ell(s_{\text{Opt}})}{\text{Ind}^\ell(s)} \right)^{1/\ell}$. 

76
Figure 3.5: Incentivization indices of various proper scoring rules relative to the optimal proper scoring rule. On the left, $\ell$ varies between 1 and 200. The chart on the right is a zoomed-in version of the same plot, showing values of $\ell$ between 1 and 10. The plotted values are $\left( \frac{\text{Ind}^\ell(s_{\ell, \text{Opt}})}{\text{Ind}^\ell(s)} \right)^{1/\ell}$.

3.6 Polynomial scoring rules with almost-optimal incentivization indices

The main result of this section is the following theorem, stating that polynomial, respectful proper scoring rules suffice to get arbitrarily close to the optimal incentivization index.

**Theorem 3.6.1.** For $\ell \geq 1$ and $\varepsilon > 0$, there exists a respectful polynomial normalized proper scoring rule $s$ satisfying $\text{Ind}^\ell(s) \leq \text{Ind}^\ell(s_{\ell, \text{Opt}}) + \varepsilon$.

The proof of Theorem 3.6.1 uses ideas from the Weierstrass approximation theorem. However, the Weierstrass approximation theorem gives a particular measure of “distance” between two functions, which does not translate to these functions having similar incentivization indices. So one challenge of the proof is ensuring convergence of a sequence of polynomials to $s_{\ell, \text{Opt}}$ in a measure related to $\text{Ind}^\ell$. A second challenge is to ensure that all polynomials in this sequence are themselves proper, respectful scoring rules. Like previous technical sections, we include a few concrete lemmas to give a sense of our proof outline.

---

10When we say a scoring rule $s$ is polynomial, we mean simply that $s$ is a polynomial function.
For example, one step in our proof is to characterize all *analytic* proper scoring rules (that is, proper scoring rules that have a Taylor expansion which converges on their entire domain \((0, 1)\)).

A necessary condition to be analytic is to be infinitely differentiable, which rules of the form \(s_{\ell, \text{Opt}}\) are not, for any fixed \(\ell\). We therefore seek to approximate such scoring rules with polynomial scoring rules (which are analytic), which are also respectful and proper.

**Theorem 3.6.2.** Let \(s : (0, 1) \rightarrow \mathbb{R}\) be analytic (and symmetric, as we have been assuming). Then \(s\) is a proper scoring rule if and only if \(s\) is nonconstant, \(s'(x) \geq 0\) everywhere, and

\[
s(x) = c_0 + \sum_{k \geq 0 \text{ odd}} c_k (2k + 1 - 2kx) \left( x - \frac{1}{2} \right)^k
\]

for some \(c_0, c_1, c_3, c_5, \cdots \in \mathbb{R}\).

As an example to help parse Theorem 3.6.2, the quadratic scoring rule has \(c_1 < 0\), and \(c_i = 0\) for all other \(i\). Using Theorem 3.6.2, we can conclude the following about \(G''_s\) for any proper scoring rule \(s\):

**Lemma 3.6.3.** Let \(s : (0, 1) \rightarrow \mathbb{R}\) be analytic and symmetric. Then \(s\) is a proper scoring rule if and only if \(G''_s\) is not uniformly zero, nonnegative everywhere, and can be written as

\[
G''_s(x) = \sum_{k \geq 0 \text{ even}} d_k \left(x - \frac{1}{2}\right)^k.
\]

Lemma 3.6.3 provides clean conditions on what functions \(G''_s\) are safe to use in our sequence of approximations, and our proof follows by following a Weierstrass approximation-type argument while keeping track of these conditions. The rest of the details for the proof of Theorem 3.6.1 can be found in Appendix A.5.

### 3.7 Conclusion

We proposed a simple model in which an expert can expend costly effort to refine their prediction, and studied the effectiveness of different proper scoring rules in incentivizing the expert
to form a precise belief. Our first main result (Theorem 3.3.3) identified the existence of a closed-form incentivization index: scoring rules with a lower index incentivize the expert to be more precise. Our second main result (Theorem 3.4.1) identified the unique optimal scoring rule with respect to this index. Section 3.5 then used the incentivization index to compare common proper scoring rules (including our newly-found optimal ones), and Section 3.6 showed that one can get arbitrarily close to the optimal incentivization index with polynomial proper scoring rules.

Our model is mathematically simple to describe, and yet it captures realistic settings surprisingly well (see Section 3.1). As such, there are many interesting directions for future work.

First: what about measures of error other than the $\ell$-th power of absolute distance? In Section 2.1, we introduced Bregman divergences and explained their relationship to proper scoring rules. In particular, if an expert is scored according to a proper scoring rule $s$, then the expert’s expected loss relative to perfectly reporting the coin’s true bias is equal to the Bregman divergence from the true bias to their report with respect to $G_s$. It is thus natural to consider the expected Bregman divergence as a notion of the expert’s error. One might ask several questions related to this:

- For a given $G$, what is the optimal proper scoring rule for Bregman divergence with respect to $G$? (We have already answered this question for $G(x) = x^2$: namely, $s_{2,\text{opt}}$.) The case of $G$ being the negative of binary entropy (in which case the Bregman divergence is KL divergence) seems particularly interesting.

- Conversely, is there a Bregman divergence with respect to which the quadratic scoring rule is optimal? What about the logarithmic scoring rule?

- As a general rule, will a proper scoring rule $s$ be unusually good at incentivizing precision, if the measure of error is Bregman divergence with respect to $G_s$? We already know that a proper scoring rule $s$ isn’t necessarily optimal for Bregman divergence with respect to $G_s$; in particular, $s_{\text{quad}}$ isn’t optimal for $\ell = 2$. However, perhaps a weaker statement to this effect could be true. Is there a proper scoring rule $s$ that is optimal for Bregman divergence with respect to $G_s$?
Other parameters of our model can also be varied to yield interesting questions:

- In our model, the expert receives information in discrete chunks (one coin flip at a time). The expert’s probability distribution over the bias of the coin is always a beta distribution with integer parameters $\alpha$ and $\beta$. Instead, we could model an expert as continually receiving information, so that their probability distribution over the bias of the coin is modeled as a beta distribution with continuously changing real parameters $\alpha$ and $\beta$. (This is inspired by an approach taken by Frongillo et al. (2015) for a related problem.) In addition to being an interesting variant of our model, a continuous model like this one may simplify the analysis. Other natural priors and processes of information gain may also be interesting to explore.

- Our work considers a globally-adaptive expert, and establishes that they behave nearly identically to a locally-adaptive expert. What about a non-adaptive expert, who must decide a priori how many coin flips to make before seeing the flip outcomes?
Chapter 4: Arbitrage-free contract functions

This chapter presents “Strictly Proper Contract Functions Can Be Arbitrage-Free” (Neyman & Roughgarden, 2022c), although all contents starting with Definition 4.5.3 are new. It assumes background on proper scoring rules presented in Section 2.1.

Summary: We consider mechanisms for truthfully eliciting probabilistic predictions from a group of experts. The standard approach – using a proper scoring rule to separately reward each expert – is not robust to collusion: experts may collude to misreport their beliefs in a way that guarantees them a larger total reward no matter the eventual outcome. It is a long-standing open question whether there is a truthful elicitation mechanism that makes any such collusion (also called arbitrage) impossible. We resolve this question positively, exhibiting a class of proper, arbitrage-free contract functions. These contract functions have two parts: one ensures that the total reward of a coalition of experts depends only on the average of their reports; the other ensures that changing this average report hurts the experts under at least one outcome.

4.1 Introduction

If a principal wishes to elicit a probabilistic forecast from an expert, they may pay the expert using a proper scoring rule. But in many settings, the principal may want to elicit forecasts from multiple experts, so as to get a better sense of expert opinion and the extent to which there is a consensus. The principal could use a proper scoring rule to elicit each expert’s forecast. If experts are not allowed to collude, then this strategy is incentive-compatible; however, French (1983) observed that experts can collude in a way that increases the sum total profit of all experts, no matter the final outcome.
For example, recall the quadratic scoring rule from Section 2.1:

$$s_{\text{quad}}(x; j) := -(1 - x_j)^2 - \sum_{j' \neq j} x_{j'}^2.$$  

Suppose that three experts believe that there is a 40%, 50%, and 90% chance of rain, and that they are paid according to the proper scoring rule $s(x; j) = s_{\text{quad}}(x; j) + 1$. If it ends up raining, then the sum of their scores will be 0.28 + 0.5 + 0.98 = 1.76; if it doesn’t rain, then the sum of their scores will be 0.68 + 0.5 − 0.62 = 0.56. However, suppose that the experts all collude to report 60%. Then the sum of their scores is 0.68 + 0.68 + 0.68 = 2.04 if it rains and 0.28 + 0.28 + 0.28 = 0.84 if it doesn’t: a larger number in both cases! The experts can agree beforehand to a redistribution of their rewards that guarantees every expert a larger profit than if they had not colluded.

Chun and Shachter (2011) called this phenomenon – in which experts collude to misreport in a way where their total reward is larger no matter the outcome – *arbitrage*. They showed that every proper scoring rule admits arbitrage – indeed, that there is an arbitrage opportunity for any group of experts so long as they do not all agree on the probability of the event. Specifically, a coalition of experts can risklessly make a profit by deviating to report an aggregate of their beliefs (in the case of the quadratic scoring rule, this aggregate is the arithmetic mean). In Chapter 5, we will extend this observation to probability distributions over more than two possible outcomes.

For many reasons, the expert may wish to make arbitrage impossible. First, the principal may wish to know whether the experts are in agreement: if they are not, for instance, the principal may want to elicit opinions from more experts. If the experts collude to report an aggregate value (as in our example), the principal does not find out whether they originally agreed. Second, even if the principal only seeks to act based on some aggregate of the experts’ opinions, their method of aggregation may be different from the one that experts use to collude. For instance, the principal may have a private opinion on the trustworthiness of each expert and wishes to average the experts’ opinions with corresponding weights. Collusion among the experts denies the principal this opportunity. Third, a principal may wish to track the accuracy of each individual expert (to figure
out which experts to trust more in the future, for instance), and collusion makes this impossible. Fourth, the space of collusion strategies that constitute arbitrage is large. In our example above, any report in $[0.546, 0.637]$ would guarantee a profit; and this does not even mention strategies in which experts report different probabilities. As such, the principal may not even be able to recover basic information about the experts’ beliefs from their reports.

As we have discussed, preventing arbitrage is impossible if the principal chooses a proper scoring rule and uses it to reward all of the experts. However, the principal has more freedom than this: they may choose to make each expert’s reward depend not only on that expert’s report but also other experts’ reports. Chun and Shachter (2011, §5) ask whether there is any mechanism for rewarding experts that makes arbitrage impossible, concluding that this “seems unlikely.” Freeman et al. (2020) explore this question further, proposing a mechanism that prevents arbitrage if the experts’ reports are guaranteed to be in the range $[\epsilon, 1 - \epsilon]$ for some positive $\epsilon$ (though their mechanism may require very large payments if $\epsilon$ is small). However, they leave open Chun and Shachter’s question of whether an incentive-compatible, arbitrage-free reward mechanism exists.

We resolve this question in the affirmative by exhibiting a class of incentive-compatible mechanisms in which arbitrage from collusion is impossible. Our mechanism takes inspiration from Brier’s quadratic scoring rule, but modifies it to take into account the aggregate performance of the remaining experts.

4.2 Related work

Freeman et al. (2020) explore the question of whether proper arbitrage-free mechanisms exist by proving positive results under different relaxations of these constraints. Their main result is a proper arbitrage-free mechanism under the restriction that the range of allowed reports is restricted to $[\epsilon, 1 - \epsilon]$. However, their mechanism necessitates payments that are exponentially large in $1/\epsilon$. Alternatively, these payments can be scaled down, but at the expense of giving essentially zero reward to each expert on the vast majority of the interval of possible reports, thus providing little
incentive for truthful reporting. They also exhibit a positive result if the properness criterion is somewhat relaxed to allow for some contract functions that are weakly proper but not proper.

Chen et al. (2014) explore the different but related topic of *arbitrage-free wagering mechanisms*. In a wagering mechanism, each expert wagers a certain amount of money along with their report, and the pool of wagers is redistributed among the experts depending on each expert’s report and wager and the eventual outcome. In this setting, they define arbitrage as any opportunity for an individual to risklessly make a profit. That is, an arbitrage opportunity is one in which an expert may unilaterally deviate by submitting a report that guarantees a profit no matter the final outcome. This differs from Chun and Shachter’s definition of arbitrage, which is concerned with riskless profit opportunities stemming from collusion between experts.

The most well-known wagering mechanism is the *weighted score wagering mechanism*, which rewards each expert based on their performance compared to other experts according to a proper scoring rule. An expert may risklessly profit from a weighted score wagering mechanism by reporting an aggregate of other experts’ reports. This is the same aggregate as the one that a coalition of experts who are rewarded with a proper scoring rule may report in order to risklessly make a profit in our setting. Chen et al. (2014, §4.1) define *no-arbitrage wagering mechanisms*, which modify the reallocation rule of weighted score wagering mechanisms to reward each expert based on their performance relative to the performance of the aggregate of all other experts’ reports. No-arbitrage wagering mechanisms can be re-interpreted in our setting as contract functions that prevent the *entire group of experts* from colluding. However, as we discuss in the next section, this is easy to accomplish; we are instead faced with the challenge of preventing collusion between any coalition of experts of any size. Thus, our mechanism and theirs share some of the same spirit, but are different mechanisms that solve different problems.
4.3 Preliminaries on contract functions

Contract functions, defined by Chun and Shachter (2011), generalize scoring rules to multiple experts. We say that there are \( m \) experts; for \( i \in [m] \), expert \( i \) reports a probability distribution \( p_i \in \Delta_n \) over \( n \) outcomes. We denote the \( j \)-th coordinate of \( p_i \) as \( p_{i,j} \).

A contract function is any function that takes as input the \( m \) experts’ reports and the outcome, and outputs the reward of each expert. Formally, a contract function is any function \( \Pi : (\Delta_n)^m \times [n] \rightarrow \mathbb{R}^m \); if the experts report distributions \( p_1, \ldots, p_m \) and the outcome is \( j \), then the vector of expert rewards is \( \Pi(p_1, \ldots, p_m; j) \). We let \( \Pi_i(\cdot) \) denote the \( i \)-th coordinate of \( \Pi(\cdot) \), i.e. expert \( i \)’s reward. We will generally use \( P \) to denote the \( m \)-tuple of reports \( (p_1, \ldots, p_m) \).

A contract function is weakly proper if for each \( i \in [m] \), expert \( i \) maximizes their expected reward by reporting their belief \( b_i \), no matter the reports \( p_{-i} \) of the other experts. Formally, \( \Pi \) is weakly proper if for all \( i \in [m] \), for all \( b_i \) and all \( p_{-i} \), \( \sum_j b_{i,j} \Pi_i(x, p_{-i}; j) \) is maximized at \( x = b_i \). We say that \( \Pi \) is proper if \( x = b_i \) is the unique maximizer, i.e. that an expert does strictly worse by misreporting their belief.

Our goal is to exhibit a proper contract function that does not permit arbitrage from collusion. We use the definition of arbitrage given by Freeman et al. (2020), which was adapted from Chun and Shachter (2011).

A contract function \( \Pi \) admits arbitrage if there is a coalition (i.e. subset) \( C \subseteq [m] \) of experts and \( m \)-tuples of expert reports \( P \) and \( Q \), with \( p_i = q_i \) for all \( i \notin C \), such that

\[
\sum_{i \in C} \Pi_i(Q; j) \geq \sum_{i \in C} \Pi_i(P; j)
\]

for all \( j \in [n] \), and the inequality is strict for some \( j \). We say that \( \Pi \) is arbitrage-free if it does not admit arbitrage. Intuitively, \( \Pi \) admits arbitrage if it is possible for a coalition of experts to collude to misreport their values in such a way that the total reward of the experts in the coalition ends up larger, no matter the outcome. (Above, the misreport is \( Q \); the constraint that \( p_i = q_i \) for \( i \notin C \).
means that only experts in $C$ change their reports.) If this is possible, then the experts in $C$ can commit beforehand to a redistribution of the extra reward in a way that makes every expert in the coalition better off no matter the eventual outcome $j$.

**Remark 4.3.1.** Positive affine transformations preserve both properness and arbitrage-freeness. That is, if $\Pi$ is proper then so is $a\Pi + b$ for any $a > 0$ and $b$, and this is likewise true for arbitrage-freeness.

The question posed by Chun and Shachter (2011) and explored by Freeman et al. (2020), which we answer affirmatively in this work, is: **Does there exist a proper arbitrage-free contract function?**

In the case of $m = 2$ experts, there is a straightforward solution:

$$
\Pi(p_1, p_2; j) = (s_{\text{quad}}(p_1; j) - s_{\text{quad}}(p_2; j), s_{\text{quad}}(p_2; j) - s_{\text{quad}}(p_1; j)).
$$

(4.1)

This contract function is proper because expert 1’s reward is the (proper) quadratic score of their report plus a term that does not depend on their report, and likewise for expert 2. It is arbitrage-free because the total reward of the two experts is 0 no matter what. Indeed, this contract function is arbitrage-free with any proper scoring rule in place of the quadratic scoring rule.

This idea does not extend to $m > 2$ experts, because an arbitrage-free contract function must not admit arbitrage by a coalition of experts of any size. While it is easy to construct a contract function that does not admit arbitrage by a coalition of size $m$ (by making the total reward always equal to 0), this does not automatically make the contract function free of arbitrage opportunities for coalitions of sizes between 2 and $m - 1$. In the next section we address this challenge and exhibit a proper contract function that is arbitrage-free for $m > 2$ experts.
4.4 A class of proper arbitrage-free contract functions

Suppose that – as before – there are \( m \geq 2 \) experts who are forecasting an event with \( n \geq 2 \) outcomes. Given experts with reports \( P = (p_1, \ldots, p_m) \) and a nonempty subset \( S \subseteq [m] \) of the experts, we will let \( \bar{p}_S := \frac{1}{|S|} \sum_{i \in S} p_i \) be the average of the experts’ reports. We will use \( \bar{p}_{-i} \) to denote \( \bar{p}_{[m]\setminus\{i\}} \).

We now state our main theorem, which exhibits a class of proper, arbitrage-free contract functions.

**Theorem 4.4.1.** Let \( \alpha \) be a real number such that \( \alpha < 0 \) or \( \alpha \geq 2(m-1)^2n \). Let \( \Pi \) be the contract function defined by

\[
\Pi_i(P; j) = s_{\text{quad}}(p_i; j) - (m-1)^2s_{\text{quad}}(\bar{p}_{-i}; j) + \alpha\bar{p}_{-i,j}
\]

for each \( i, j \). Then \( \Pi \) is proper and arbitrage-free.

Note that in the case of \( m = 2 \), setting \( \alpha = 0 \) yields our aforementioned solution for two experts in Equation 4.1. Unfortunately, setting \( \alpha = 0 \) for \( m > 2 \) experts causes arbitrage-freeness to fail in certain edge cases.

One can think of the contract function in Theorem 4.4.1 as having two parts. The first part, \( s_{\text{quad}}(p_i; j) - (m-1)^2s_{\text{quad}}(\bar{p}_{-i}; j) \), ensures that any coalition’s total reward depends only on the average of the coalition’s reports. In effect this significantly limits the degrees of freedom that a coalition has when colluding. The second part, \( \alpha\bar{p}_{-i,j} \), ensures that any deviation in this average report causes a decrease in total reward under at least one outcome.

We first present the proof of Theorem 4.4.1 for \( n = 2 \) outcomes, as this allows us to simplify notation while still explaining the core ideas.

**Proof of Theorem 4.4.1 for \( n = 2 \).** First, note that \( \Pi \) is proper, because expert \( i \)’s reward is their quadratic score plus a term that does not depend on their report. It remains to show that \( \Pi \) is arbitrage-free.
Let $C \subseteq [m]$ be a coalition of experts. Properness entails that no expert can unilaterally find an arbitrage opportunity, so we may assume that $|C| \geq 2$.

For an outcome $j$ and a subset $S \subseteq [m]$, let $p_{S,j} := \sum_{i \in S} p_{i,j}$.

**Lemma 4.4.2.** Let $d = m - 1 - \frac{\alpha}{4(m-1)}$. The expression for $\Pi_i(P; j)$ is equal to

$$2(p_{[m],j} - d - 1)(p_{[m],j} - 2p_{i,j} - d + 1) + f(m, \alpha),$$

(4.2)

for some function $f$.

**Proof of Lemma 4.4.2.** For some $f$ (whose exact form does not concern us), we have

$$\Pi_i(P; j) = s_{quad}(p_{i}; j) - (m - 1)^2s_{quad}(\overline{p}_{-i}; j) + \alpha\overline{p}_{-i,j}$$

$$= -2(1 - p_{i,j})^2 - (m - 1)^2 \left( -2 \left(1 - \frac{p_{-i,j}}{m-1}\right) \right) + \frac{\alpha}{m-1} p_{-i,j}$$

$$= -2(1 - p_{i,j})^2 + 2(m - 1 - p_{-i,j})^2 + \frac{\alpha}{m-1} p_{-i,j}$$

$$= -2(1 - p_{i,j})^2 + 2 \left( d - p_{-i,j} \right)^2 + f(m, \alpha)$$

$$= 2 \left( d - p_{-i,j} + (1 - p_{i,j}) \right) \left( d - p_{-i,j} - (1 - p_{i,j}) \right) + f(m, \alpha)$$

$$= 2(p_{[m],j} - d - 1)(p_{[m],j} - 2p_{i,j} - d + 1) + f(m, \alpha),$$

as desired. \qed

Equation 4.2 makes it evident that rewards add nicely across experts in a coalition $C$, as the first term of the product is the same for all experts in $C$. We will use the notation $\Pi_C(P; j)$ to denote $\sum_{i \in C} \Pi_i(P; j)$. The key idea is that, as we are about to show, if the reports of experts not in $C$ are held fixed, $\Pi_C(P; j)$ depends only on $p_{C,j}$. Thus, the experts in $C$ have only one degree of freedom available for colluding: the sum of their reports.

We write $\overline{C}$ to mean $[m] \setminus C$. We have

$$\Pi_C(P; j) = 2 \sum_{i \in C} (p_{[m],j} - d - 1)(p_{[m],j} - 2p_{i,j} - d + 1) + |C| f(m, \alpha)$$

88
for some function $g$. Now, recall the constraints on $\alpha$ in Theorem 4.4.1, and note that $\alpha < 0 \iff d > m - 1$ and $\alpha \geq 4(m - 1)^2 \iff d \leq 0$. With this in mind, we now prove the following claim, which is sufficient to complete our proof.

**Claim 4.4.3.** If $d \leq 0$, then for each $j$ and for all possible reports of experts not in $C$, $\Pi_C(P; j)$ is a strictly increasing function of $p_{C,j}$. If $d > m - 1$, it is a strictly decreasing function of $p_{C,j}$.

By virtue of deriving Equation 4.3, we have already proven the most difficult part of Claim 4.4.3, which is that $\Pi_C(P; j)$ is a function of (i.e. determined by) $p_{C,j}$. Why is this function’s monotonicity sufficient to complete our proof of Theorem 4.4.1? Since $p_{C,1} + p_{C,2} = |C|$, it follows from Claim 4.4.3 that for $d \leq 0$ and $d > m - 1$, colluding in a way that increases the total reward in the case of one outcome necessarily decreases it in the case of the other outcome.

**Proof of Claim 4.4.3.** We first consider the case of $|C| = 2$. In this case we have

$$\Pi_C(P; j) = 4(p_{\overline{C},j} - d + 1)p_{C,j} + g(m, \alpha, |C|, p_{\overline{C},j}).$$

Now, $0 \leq p_{\overline{C},j} \leq m - 2$, which means that $1 - d \leq p_{\overline{C},j} - d + 1 \leq m - 1 - d$. If $d \leq 0$, this quantity is guaranteed to be strictly positive, so $\Pi_C(P; j)$ is a strictly increasing function of $p_{C,j}$; if $d > m - 1$, it is guaranteed to be negative, so $\Pi_C(P; j)$ is a strictly decreasing function of $p_{C,j}$.

Now assume that $|C| > 2$. In this case, it follows from Equation 4.3 that $\Pi_C(P; j)$ is a parabola with a minimum at

$$\frac{(|C| - 1)(d - p_{\overline{C},j}) - 1}{|C| - 2}.$$ 

We wish to show that if $d \leq 0$ then this quantity is at most 0, and that if $d > m - 1$ then it is at least
|C| (since the range of possible values of \(p_{C,j}\) is \([0, |C|]\)). If \(d \leq 0\) then, since \(p_{C,j} \geq 0\), we have

\[
\frac{(|C| - 1)(d - p_{\overline{C},j}) - 1}{|C| - 2} \leq \frac{-1}{|C| - 2} \leq \frac{-1}{m - 2} \leq 0.
\]

If \(d > m - 1\) then, since \(p_{\overline{C},j} \leq m - |C|\), we have

\[
\frac{(|C| - 1)(d - p_{\overline{C},j}) - 1}{|C| - 2} \geq \frac{(|C| - 1)^2 - 1}{|C| - 2} = |C|.
\]

\[\square\]

Having proved the claim, we have completed the proof of Theorem 4.4.1 for \(n = 2\).

\[\square\]

We note that setting \(\alpha = 0\) results in a contract function that is arbitrage-free except in one edge case: in the event that all but two experts assign a probability of zero to some outcome \(j\), the remaining experts can collude to adjust their probabilities – in particular, lowering the total probability they assign to outcome \(j\) – in a way that increases their total reward under outcome \(j\) and leaves the remaining rewards unchanged. If we are willing to put this exception aside (e.g. if we only allow reports strictly between 0 and 1), then we may regard the resulting contract function \(\Pi_i(\mathbf{P}; j) = s_{\text{quad}}(\mathbf{p}_i; j) - (m - 1)^2 s_{\text{quad}}(\mathbf{p}_{-i}; j)\) as arbitrage-free. This contract function has a natural interpretation: it rewards an expert for the accuracy of their forecast but penalizes the expert if others are accurate in aggregate. This rule is reminiscent of the no-arbitrage wagering mechanism for the quadratic scoring rule given by Chen et al. (2014), except that the penalty is multiplied by a factor of \((m - 1)^2\).

We now present the proof of Theorem 4.4.1 in full generality.

**Proof of Theorem 4.4.1 for general \(n\).** First, note that \(\Pi\) is proper, because expert \(i\)’s reward is their quadratic score plus a term that does not depend on their report. It remains to show that \(\Pi\) is arbitrage-free.

Let \(C \subseteq [m]\) be a coalition of experts. Properness entails that no expert can unilaterally find an arbitrage opportunity, so we may assume that \(|C| \geq 2\).
For an outcome \( j \) and a subset \( S \subseteq [m] \), let \( p_{S,j} := \sum_{i \in S} p_{i,j} \). Let \( d = m - 1 - \frac{\alpha}{2(m-1)} \). For some \( f \) whose particular form does not concern us, we have

\[
\Pi_i(\mathbf{P}, j) = s_{\text{quad}}(\mathbf{p}_i; j) - (m - 1)^2 s_{\text{quad}}(\mathbf{P}_{-i}; j) + \alpha \mathbf{P}_{-i,j}
\]

\[
= (m - 1)^2 \left( 1 - \frac{1}{m - 1} p_{-i,j} \right)^2 - (1 - p_{i,j})^2 + \sum_{\ell \neq j} \left( (m - 1)^2 \left( \frac{1}{m - 1} p_{-i,\ell} \right)^2 - p_{i,\ell}^2 \right) + \frac{\alpha}{m - 1} p_{-i,j}
\]

\[
= (d - p_{-i,j})^2 - (1 - p_{i,j})^2 + \sum_{\ell \neq j} \left( p_{-i,\ell}^2 - p_{i,\ell}^2 \right) + f(m, n, \alpha)
\]

\[
= (d - p_{-i,j} + (1 - p_{i,j}))(d - p_{-i,j} - (1 - p_{i,j})) + \sum_{\ell \neq j} (p_{-i,\ell} + p_{i,\ell})(p_{-i,\ell} - p_{i,\ell}) + f(m, n, \alpha)
\]

\[
= (p_{[m],j} - d - 1)(p_{[m],j} - 2p_{i,j} - d + 1) + \sum_{\ell \neq j} p_{[m],\ell}(p_{[m],\ell} - 2p_{i,\ell}) + f(m, n, \alpha).
\]

We will use the notation \( \Pi_C(\mathbf{P}; j) \) to denote \( \sum_{i \in C} \Pi_i(\mathbf{P}; j) \). We also write \( \overline{C} \) to mean \( [m] \setminus C \) and \( \mathbf{P}_{\overline{C}} \) to mean the collection of reports \( \mathbf{p}_i \) for \( i \in \overline{C} \). We have

\[
\Pi_C(\mathbf{P}; j) = \sum_{i \in C} \left( (p_{[m],j} - d - 1)(p_{[m],j} - 2p_{i,j} - d + 1) + \sum_{\ell \neq j} p_{[m],\ell}(p_{[m],\ell} - 2p_{i,\ell}) \right) + |C| f(m, n, \alpha)
\]

\[
= (p_{C,j} + p_{\overline{C},j} - d - 1)((|C| - 2)p_{C,j} + |C| (p_{\overline{C},j} - d + 1))
\]

\[
+ \sum_{\ell \neq j} (p_{C,\ell} + p_{\overline{C},\ell})(|C| p_{\overline{C},\ell} + (|C| - 2)p_{C,\ell}) + |C| f(m, n, \alpha)
\]

\[
= (|C| - 2)p_{\overline{C},j}^2 + ((2 |C| - 2)(p_{\overline{C},j} - d) + 2)p_{C,j}
\]

\[
+ \sum_{\ell \neq j} \left( (|C| - 2)p_{\overline{C},\ell}^2 + (2 |C| - 2)p_{\overline{C},\ell}p_{C,\ell} \right) + g(m, n, \alpha, |C|, \mathbf{P}_{\overline{C}})
\]

\[
= (2 - (2 |C| - 2)d)p_{C,j} + \sum_{\ell} \left( (|C| - 2)p_{\overline{C},\ell}^2 + (2 |C| - 2)p_{\overline{C},\ell}p_{C,\ell} \right) + g(m, n, \alpha, |C|, \mathbf{P}_{\overline{C}})
\]

for some function \( g \). Consider a different vector \( \mathbf{Q} \) that agrees with \( \mathbf{P} \) on \( \overline{C} \).
**Case 1:** \( \alpha < 0 \). In this case, \( 2 - (2|C| - 2)d < 2 - (2|C| - 2)(m-1) \). Let \( \tilde{j} = \arg \max_{\ell} q_{C,\ell} - p_{C,\ell} \), and let \( \epsilon = q_{C,j} - p_{C,j} \). We note that

\[
\sum_{\ell} q_{C,\ell}^2 - p_{C,\ell}^2 = \sum_{\ell} (q_{C,\ell} - p_{C,\ell}) (q_{C,\ell} + p_{C,\ell}) \leq \epsilon \sum_{\ell} (q_{C,\ell} + p_{C,\ell}) = 2 \epsilon |C|.
\]

Thus, we have

\[
\Pi_C(Q; \tilde{j}) - \Pi_C(P; \tilde{j}) = (2 - (2|C| - 2)d) \epsilon + \sum_{\ell} (|C| - 2)(q_{C,\ell}^2 - p_{C,\ell}^2) + (2|C| - 2)(q_{C,\ell} - p_{C,\ell}) p_{C,\ell}
\]

\[
\leq (2 - (2|C| - 2)d) \epsilon + (|C| - 2) \cdot 2 \epsilon |C| + (2|C| - 2)(m - |C|) \epsilon
\]

\[
\leq (2 - (2|C| - 2)(m-1)) \epsilon + (|C| - 2) \cdot 2 \epsilon |C| + (2|C| - 2)(m - |C|) \epsilon
\]

\[
= 2 \epsilon (1 + (|C| - 2)|C| + (|C| - 1)(1 - |C|)) = 0,
\]

with equality in the second step only when \( \epsilon = 0 \), i.e. \( q_{C,\ell} = p_{C,\ell} \) for all \( \ell \). Thus, either the total reward of the experts in \( C \) is the same under \( Q \) as under \( P \) for every outcome, or it is strictly smaller under \( Q \) in the case of outcome \( \tilde{j} \).

**Case 2:** \( \alpha \geq 2(m-1)^2 n \). In this case, \( 2 - (2|C| - 2)d \geq 2 + (2|C| - 2)(m-1)(n-1) \). Let \( \tilde{j} = \arg \max_{\ell} p_{C,\ell} - q_{C,\ell} \), and let \( \epsilon = p_{C,j} - q_{C,j} \). Since \( \sum_{\ell} (q_{C,\ell} - p_{C,\ell}) = 0 \), it follows that \( q_{C,\ell} - p_{C,\ell} \leq (n-1) \epsilon \) for all \( \ell \). We note that

\[
\sum_{\ell} q_{C,\ell}^2 - p_{C,\ell}^2 \leq \sum_{\ell} (q_{C,\ell} + p_{C,\ell}) \max(q_{C,\ell} - p_{C,\ell}, 0)
\]

\[
\leq 2 |C| \sum_{\ell} \max(q_{C,\ell} - p_{C,\ell}, 0) \leq 2 |C| (n-1) \epsilon.
\]

We also have that

\[
\sum_{\ell} (q_{C,\ell} - p_{C,\ell}) p_{C,\ell} \leq (m - |C|) \sum_{\ell} \max(q_{C,\ell} - p_{C,\ell}, 0) \leq (m - |C|)(n-1) \epsilon.
\]
Therefore,

\[ \Pi_C(Q; \tilde{j}) - \Pi_C(P; \tilde{j}) = -(2 - (2 |C| - 2)d)\epsilon + \sum_{\ell} (|C| - 2)(q_{C,\ell}^2 - p_{C,\ell}^2) + (2 |C| - 2)(q_{C,\ell} - p_{C,\ell})p_{\overline{C},\ell} \]

\[ \leq -(2 - (2 |C| - 2)d)\epsilon + (|C| - 2) \cdot 2 |C| (n - 1)\epsilon + (2 |C| - 2) \cdot (m - |C|)(n - 1)\epsilon \]

\[ = 2\epsilon(-1 - (|C| - 1)(m - 1)(n - 1) + (n - 1)(m(|C| - 1) - |C|)) \]

\[ = 2\epsilon(-1 - (n - 1)) = -2\epsilon n \leq 0, \]

with equality in the last step only when \( \epsilon = 0 \), i.e. \( q_{C,\ell} = p_{C,\ell} \) for all \( \ell \). As in the previous case, this means that either the total reward of the experts in \( C \) is the same under \( Q \) as under \( P \) for every outcome, or it is strictly smaller under \( Q \) in the case of outcome \( \tilde{j} \). This completes the proof. \( \square \)

### 4.5 Future directions in arbitrage-freeness

While the contract functions defined in Theorem 4.4.1 are proper and arbitrage-free, there are other desirable notions of arbitrage-freeness that they do not satisfy. To see this, consider the following concrete example: suppose that there are two outcomes, and all \( m > 2 \) experts think that Outcome 1 will happen with probability 1. Suppose that \( \alpha \) is chosen to be only slightly negative – very close to zero. In that case, all experts reporting their true belief is just about the worst possible outcome for the experts: they each believe that they will each receive a score of \( \alpha \). By comparison, if all experts were to instead lie and report \( p_i = (0, 1) \) (i.e. that Outcome 2 will happen with probability 1), then each expert believes that they will receive reward \( 2((m - 1)^2 - 1) \). That is, the experts are much better off if everyone lies than if everyone tells the truth.

Why is this not a counterexample to arbitrage-freeness? Well, if Outcome 2 happens, then the experts will get reward \( 2((m - 1)^2 - 1) \) if they tell the truth and \( \alpha \) if they lie. The experts believe that Outcome 2 will happen with probability 0, but if Outcome 2 \textit{does} happen, then they will end up better off if they tell the truth. However, since every expert believes that Outcome 1 is guaranteed to happen, it seems that in practice the experts would want to collude in this situation: they believe...
that they can collude to guarantee themselves a greater profit with probability 100%.

How might we expand the definition of arbitrage to include the collusion scenario we just described? One way is to say that $\Pi$ admits arbitrage if there is a collusion strategy for the experts in $C$ that, in the opinion of every expert in $C$, increases the expected total reward of the experts in $C$. Formally:

**Definition 4.5.1.** A contract function $\Pi$ admits expected arbitrage if there is a coalition $C \subseteq [m]$ of experts and vectors of reports $P = (p_1, \ldots, p_m)$, $Q = (q_1, \ldots, q_m)$, with $p_i = q_i$ if $i \notin C$, such that for all $i \in C$ we have

$$\sum_{j \in [n]} p_{i,j} \sum_{k \in C} \Pi_k (P; j) \leq \sum_{j \in [n]} p_{i,j} \sum_{k \in C} \Pi_k (Q; j),$$

and the inequality is strict for some $i$. We say that $\Pi$ is free of expected arbitrage if it does not admit expected arbitrage.

Up to edge scenarios,\(^1\) if a contract function admits arbitrage then it also admits expected arbitrage. On the other hand, in the case of $m > 2$ experts, the scoring rules described by Theorem 4.4.1 (which do not admit arbitrage) do admit expected arbitrage. As an example, consider two outcomes and $m$ experts with beliefs $(\frac{1}{2}, \frac{1}{2})$. If all experts report their beliefs, then each expert’s reward is $\frac{a}{2} + \frac{1}{2}((m-1)^2 - 1)$, no matter the outcome. If all experts instead report $(1, 0)$ then each expert expects a reward of $\frac{a}{2} + (m - 1)^2 - 1$, which is larger. This raises the following question:

**Question 4.5.2.** Is there a proper scoring rule that does not admit expected arbitrage?

We can define an even stronger notion of arbitrage-freeness:

**Definition 4.5.3.** We say that $\Pi$ admits redistributional arbitrage if there is:

- A coalition $C \subseteq [m]$ of experts

---

\(^1\)It is possible for a coalition of experts to collude in a way that increases their total reward under an outcome to which they all assign probability 0. If their reward in the case of all other outcome is unchanged, such a deviation would constitute arbitrage but not expected arbitrage.
• A redistribution agreement \( A \) that takes as input a list of probability distributions \( \mathbf{X} = (x_1, \ldots, x_m) \), another list of probability distributions \( \mathbf{Y} = (y_1, \ldots, y_m) \) such that \( x_i = y_i \) for any \( i \notin C \),\(^2\) and an outcome \( j \), and outputs a list of rewards, one for each \( i \in C \), such that

\[
\sum_{i \in C} A_i(\mathbf{X}; \mathbf{Y}; j) = \sum_{i \in C} \Pi_i(\mathbf{X}; j).
\]

(We call this the budget balance property of \( A \).

\[
\sum_{j \in [n]} \Pi_i(\mathbf{X}; j) = \sum_{i \in C} \Pi_i(\mathbf{X}; j).
\]

\[
\sum_{j \in [n]} \Pi_i(\mathbf{X}; j) = \sum_{i \in C} \Pi_i(\mathbf{X}; j).
\]

... such that there are vectors of reports \( \mathbf{P} = (p_1, \ldots, p_m) \), \( \mathbf{Q} = (q_1, \ldots, q_m) \), with \( p_i = q_i \) if \( i \notin C \), so that for all \( i \in C \) we have

\[
\sum_{j \in [n]} p_{i,j} A_i(\mathbf{Q}; \mathbf{P}; j) \geq \sum_{j \in [n]} p_{i,j} \Pi_i(\mathbf{P}; j),
\]

and the inequality is strict for some \( i \). We say that \( \Pi \) is free of redistribution arbitrage if it does not admit expected arbitrage.

We show that every \( \Pi \) that admits expected arbitrage also admits redistributional arbitrage – that is, that if there is a misreport that each experts believes will make the pool of colluding experts better off in expectation, then there is a way to redistribute the earnings in a way that makes each expert believe that they will be better off in expectation.

**Theorem 4.5.4.** Every contract function \( \Pi \) that admits expected arbitrage admits redistributional arbitrage.

**Proof.** Let \( \Pi \) be a contract function that admits expected arbitrage, and let \( C \), \( \mathbf{P} \), and \( \mathbf{Q} \) be as in Definition 4.5.1. For any \( \mathbf{X} = (x_1, \ldots, x_m) \) and \( \mathbf{Y} = (y_1, \ldots, y_m) \) that coincide outside of \( C \), let

\[
W(\mathbf{X}; \mathbf{Y}; j) := \sum_{i \in C} \Pi_i(\mathbf{X}; j) - \sum_{i \in C} \Pi_i(\mathbf{Y}; j)
\]

\(^2\)Here, the \( x_i \) represent the reports given to \( \Pi \), the \( y_i \) represent the pre-collusion probabilities (these can be thought of as the “true beliefs”), and \( j \) represents the outcome that happens.
be the total gain of experts in $C$ by colluding to report $X$ instead of $Y$. Thus, $\sum_j p_{i,j} W(Q; P; j) \geq 0$ for all $i$ and the inequality is strict for some $i$. Define

$$A_i(X; Y; j) := \Pi_i(Y; j) + \frac{1}{|C|} W(X; Y; j).$$

(Clearly $A$ is budget-balanced.) Then for any $i \in C$ we have

$$\sum_j p_{i,j} A_i(Q; P; j) = \sum_j p_{i,j} \left( \Pi_i(P; j) + \frac{1}{|C|} W(Q; P; j) \right) \geq \sum_j p_{i,j} \Pi_i(P; j),$$

with the inequality strict for some $i$, as desired. \hfill \square

We close with the question analogous to Question 4.5.2 for this stronger notion:

**Question 4.5.5.** Is there a proper scoring rule that does not admit redistributional arbitrage?
Chapter 5: Quasi-arithmetic pooling

This chapter presents “From Proper Scoring Rules to Max-Min Optimal Forecast Aggregation” (Neyman & Roughgarden, 2023), although the contents of Section 5.5 are original to this thesis. It assumes background on proper scoring rules presented in Section 2.1, as well as background on forecast aggregation presented in Section 2.2.

Summary: This chapter forges a strong connection between two seemingly unrelated forecasting problems: incentive-compatible forecast elicitation and forecast aggregation. We have previously discussed proper scoring rules as a solution to the former problem (see Section 2.1). To each proper scoring rule $s$ we associate a corresponding method of aggregation, mapping expert forecasts and expert weights to a “consensus forecast,” which we call quasi-arithmetic (QA) pooling with respect to $s$. We justify this correspondence in several ways:

- QA pooling with respect to the two most well-studied scoring rules (quadratic and logarithmic) corresponds to the two most well-studied forecast aggregation methods (linear and logarithmic).
- Given a scoring rule $s$ used for payment, a forecaster agent who sub-contracts several experts, paying them in proportion to their weights, is best off aggregating the experts’ reports using QA pooling with respect to $s$, meaning this strategy maximizes its worst-case profit (over the possible outcomes).
- The score of an aggregator who uses QA pooling is concave in the experts’ weights. As a consequence, online gradient descent can be used to learn appropriate expert weights from repeated experiments with low regret.
- QA pooling can be used to define overconfidence with respect to a scoring rule. The re-
sulting notion of overconfidence turns out to be equivalent to another natural definition of overconfidence.

- The class of all QA pooling methods is characterized by a natural set of axioms (generalizing classical work by Kolmogorov on quasi-arithmetic means).

### 5.1 Introduction and motivation

#### 5.1.1 Choice of scoring rule as a value judgment

There are infinitely many proper scoring rules. How might a principal go about deciding which one to use? To gain some intuition, recall the quadratic and logarithmic scoring rules from Section 2.1:

\[
s_{\text{quad}}(x; j) := -(1 - x_j)^2 - \sum_{j' \neq j} x_{j'}^2,
\]

\[
s_{\text{log}}(x; j) := \ln(x_j)
\]

Let us consider the quadratic and logarithmic scoring rules in the case of a binary Yes/No outcome. In Figure 5.1, for both scoring rules, we show the difference between the expert’s score if a given outcome happens and their score if it does not happen, as a function of the probability that they assign to the outcome.\(^1\)

For the quadratic scoring rule, this difference scales linearly with the expert’s report. Meanwhile, for the logarithmic rule, the difference changes more slowly than for the quadratic rule for probabilities near 50%, but much more quickly at the extremes. Informally speaking, this means that the logarithmic rule indicates a preference (of the elicitor) for high precision close to 0 and 1, while the quadratic rule indicates a more even preference for precision across [0, 1]. Put another way, an elicitor who chooses to use the logarithmic scoring rule renders a judgment that the prob-

\(^1\)We scale down the logarithmic rule by a factor of \(2 \ln 2\) to make the two rules comparable. The factor \(2 \ln 2\) was chosen to make the range of values taken on by the expected score functions of the two scoring rules the same.
Figure 5.1: Difference between expert’s score if an outcome happens and if it does not happen, as a function of the expert’s report, for the quadratic and logarithmic scoring rules. For example, if the expert reports a 70% probability of an outcome, then under the quadratic rule they receive a score of \(-\| (0.7, 0.3) - (1, 0) \|_2^2 = -0.18\) if the outcome happens and \(-\| (0.7, 0.3) - (0, 1) \|_2^2 = -0.98\) if it does not: a difference of 0.8. If scored with the logarithmic rule, this difference would be 0.61.

abilities 0.01 and 0.001 are qualitatively quite different; one who uses the quadratic rule indicates that these probabilities are very similar.

Now, recall from Section 2.2 the two most widely-used forecast aggregation methods: linear and logarithmic pooling. Given probabilistic forecasts \(x_1, \ldots, x_m\), the linear pool with weights \(w_1, \ldots, w_m\) (adding to 1) is given by the weighted arithmetic mean: \(x^* = \sum_i w_i x_i\). Meanwhile, the logarithmic pool is given by the (renormalized) geometric mean: \(x^*(j) = c \prod_{i=1}^m (x_i(j))^{w_i}\).

On its surface, the elicitation of forecasts has seemingly little to do with their aggregation. However, given that the choice of scoring rule implies a subjective judgment about how different probabilities compare to one another, it makes sense to apply this judgment to the aggregation of forecasts as well.

As an example, consider the setting of weather prediction, with models playing the role of experts. In such contexts we often care about low-probability extreme events: a 0.1% chance of an imminent major hurricane may not be worth preparing for; a 1% chance could mean significant preparations, and a 10% chance could mean mandatory evacuations. The need to distinguish very unlikely events from somewhat unlikely events has two consequences. First, as discussed above, this is a reason to use the logarithmic scoring rule to assess the quality of weather models. Second,
we wish to avoid the failure mode in which an ill-informed forecaster assigns a high probability due to lack of evidence and thereby drowns out a better-informed low-probability forecast. We would expect to encounter this failure mode with linear pooling; for example, if the more informed model predicts a 0.1% chance and the less informed model predicts a 20% chance, linear pooling with equal weights\(^2\) would predict roughly a 10% chance. Logarithmic pooling, by contrast, assigns roughly a 1.6% chance to the event, avoiding this failure mode. In general, a calibrated model that predicts a very low probability must have good evidence, so it may make sense to give the model more weight.

By contrast, a consultant whose job is to determine the closest races in an election may not much care about the difference between a 0.1% and a 1% chance of victory. After all, attention and resources are generally devoted to races with highly uncertain outcomes. As such, it might make sense to assess the qualities of forecast models using the quadratic scoring rule. Similarly, without a compelling reason to pay extra attention to extreme probabilities, it may make more sense to simply take the average of the forecasts' opinions.

In the case of extreme weather prediction, we have argued in favor of using the logarithmic scoring rule to assess the models and the logarithmic pooling method to aggregate them, for similar reasons. In the case of political prediction for targeting close races, we have argued in favor of the quadratic scoring rule and linear pooling, also for similar reasons. Could there be a formal connection between proper scoring rules and forecast aggregation methods that captures this intuition? This brings us to the main focus of this work: namely, we prove a novel correspondence between proper scoring rules and forecast aggregation methods.

### 5.1.2 Our definitions

Recall from Section 2.1 the *Savage representation* of a proper scoring rule:

\[
    s(x; j) = G(x) + \langle \delta_j - x, g(x) \rangle
\]

\(^2\)Assigning equal weights makes sense when there is not enough information to predict in advance which model will be less informed.
where $G$ is the expected score function of $s$ and $g$ is the gradient of $G$. The function $g$, which will be central to our work, describes the difference in the expert’s score depending on which outcome happens. More precisely, the vector $(s(x; j_1), \ldots, s(x; j_n))$ is exactly the vector $g(x)$, except possibly for a uniform translation in all coordinates. Put otherwise, for any two outcomes $j_1$ and $j_2$, we have $g_1(x) - g_2(x) = s(x; j_1) - s(x; j_2)$ (this can be verified from Equation 5.1). This is precisely the quantity plotted in Figure 5.1 for the quadratic and logarithmic scoring rules, if $j_1$ is “Yes” and $j_2$ is “No.” This observation about the function $g$ motivates the correspondence that we will establish between proper scoring rules and forecast aggregation methods.

**Quasi-arithmetic pooling**  
We can now define the aforementioned correspondence between proper scoring rules and forecast aggregation methods. Given a proper scoring rule $s$ used for elicitation, and given $m$ probability distributions $p_1, \ldots, p_m$ and expert weights $w_1, \ldots, w_m$, the aggregate distribution $p^*$ that we suggest is the one satisfying

$$g(p^*) = \sum_{i=1}^{m} w_i g(p_i).$$

(In Section 5.2 we will define this notion more precisely using subgradients of $G$ instead of gradients; this will ensure that $p^*$ is well defined, i.e. that it exists and is unique.) This definition of $p^*$ can be restated as the forecast that minimizes the weighted average Bregman divergence (with respect to $G$) to all experts’ forecasts.

We refer to this pooling method as **quasi-arithmetic pooling** with respect to $g$ (or the scoring rule $s$), or QA pooling for short.\(^3\) To get a sense of QA pooling, let us determine what this method looks like for the quadratic and logarithmic scoring rules.

**QA pooling with respect to the quadratic scoring rule**  
Recall from Section 2.1 that the expected score function $G_{\text{quad}}$ of the quadratic scoring rule $s_{\text{quad}}$ is equal to $\sum_j x_j^2 - 1$. We thus have

\[^3\text{This term comes from the notion of quasi-arithmetic means: given a continuous, strictly increasing function } f \text{ and values } x_1, \ldots, x_m, \text{ the quasi-arithmetic mean with respect to } f \text{ of these values is } f^{-1}(1/m \sum_i f(x_i)).\]
\[ g_{\text{quad}}(x) = (2x_1, \ldots, 2x_n), \] so we are looking for the \( p^* \) such that
\[ (2p^*(1), \ldots, 2p^*(n)) = \sum_{i=1}^m w_i (2p_i(1), \ldots, 2p_i(n)). \]

This is \( p^* = \sum_{i=1}^m w_i p_i \). Therefore, QA pooling for the quadratic scoring rule is precisely linear pooling, which we introduced in Section 2.2 as the simplest and most widely-used forecast aggregation method.

**QA pooling with respect to the logarithmic scoring rule** Recall that the expected score function \( G_{\log} \) of the logarithmic scoring rule \( s_{\log} \) is equal to \( \sum_j x_j \ln(x_j) \). We thus have \( g_{\log}(x) = (\ln x_1 + 1, \ldots, \ln x_n + 1) \), so we are looking for the \( p^* \) such that
\[ (\ln p^*(1) + 1, \ldots, \ln p^*(n) + 1) = \sum_{i=1}^m w_i (\ln p_i(1) + 1, \ldots, \ln p_i(n) + 1). \]

By exponentiating the components on both sides, we find that \( p^*(j) = c \prod_{i=1}^n (p_i(j))^{w_i} \) for all \( j \), for some proportionality constant \( c \). This is precisely the definition of the logarithmic pooling method, which we introduced in Section 2.2 as a simple, well-motivated, and effective way of aggregating probabilistic forecasts. (The constant \( c \) comes from the fact that values of \( g(\cdot) \) should be interpreted modulo translation by the all-ones vector; see Remark 5.2.2.)

The fact that this pooling scheme maps the two most well-studied scoring rules to the two most well-studied forecast aggregation methods has not been noted previously, to our knowledge. This correspondence suggests that – beyond just our earlier informal justification – QA pooling with respect to a given scoring rule may be a fundamental concept. The rest of this work argues that this is indeed the case.

This correspondence may have practical implications for forecasters. While the quadratic and logarithmic scoring rules are both ubiquitous in practice, linear pooling is far more common than logarithmic pooling (Ranjan & Gneiting, 2010). This is despite empirical evidence that logarithmic
pooling often outperforms linear pooling (Satopää et al., 2014). The connection that we establish between the logarithmic scoring rule and logarithmic pooling provides further reason to think that logarithmic pooling has been somewhat overlooked.

5.1.3 Our results

(Section 5.3) Max-min optimality Suppose that a principal asks you to issue a forecast and will pay you according to $s$. You are not knowledgeable on the subject but know some experts whom you trust on the matter (perhaps to varying degrees). You sub-contract the experts, promising to pay each expert $i$ according to $w_i \cdot s$. By using QA pooling according to $s$ on the experts’ forecasts, you guarantee yourself a profit; in fact, this strategy maximizes your worst-case profit, and is the unique such report. Furthermore, this profit is the same for all outcomes. This fact can be interpreted to mean that you have, in a sense, pooled the forecasts “correctly”: you do not care which outcome will come to pass, which means that you have correctly factored the expert opinions into your forecast. We give an additional interpretation of this optimality notion as maximizing an aggregator’s guaranteed improvement over choosing an expert at random.

(Section 5.4) Learning expert weights Pooling forecasts entails assigning weights to experts. Where do these weights come from? How might one learn them from experience?

Suppose we have a fixed proper scoring rule $s$, and further consider fixing the reports of the $m$ experts as well as the eventual outcome. One can ask: what does the score of the aggregate distribution (per QA pooling with respect to $s$) look like as a function of $w$, the vector of expert weights? We prove that this function is concave. This is useful because it allows for online convex optimization over expert weights.

Theorem (informal). Let $s$ be a bounded proper scoring rule.\(^4\) For time steps $t = 1 \ldots T$, $m$ experts report forecasts to an aggregator, who combines them into a forecast $p^t$ using QA pooling with respect to $s$ and suffers a loss of $-s(p^t; j^t)$, where $j^t$ is the outcome at time step $t$. If the aggregator

\(^4\)For which QA pooling is well defined (we discuss this below).
updates the experts’ weights using online gradient descent, then the aggregator’s regret compared to the best weights in hindsight is $O(\sqrt{T})$.

The aforementioned concavity property is a nontrivial fact that demonstrates an advantage of QA pooling with respect to the proper scoring rule used for elicitation, as compared with using e.g. linear or logarithmic pooling regardless of the method of elicitation: linear and logarithmic pooling satisfy the concavity property for some proper scoring rules $s$ but not others.

(Section 5.5) Overconfidence Informally, an expert is overconfident if the expert’s forecasts would be more accurate if they were less extreme (closer to uniform). Much as it makes sense to aggregate forecasts in a way that depends on the scoring rule, it also makes sense to define overconfidence in a way that depends on the scoring rule. We give two natural definitions of overconfidence with respect to a proper scoring rule. The first definition says that a series of forecasts made by an expert is overconfident if the expert’s total score is lower than the score the expert expects. The second definition says that a series of forecasts made by an expert is overconfident if using QA pooling to make the forecasts less extreme would increase the expert’s total score. We prove that these two definitions are equivalent.

(Section 5.6) Natural axiomatization for QA pooling methods Kolmogorov (1930) and Nagumo (1930) independently came up with a simple axiomatization of quasi-arithmetic means. We show how to change these axioms to allow for weighted means; the resulting axiomatization is a natural characterization of all quasi-arithmetic pooling methods in the case of $n = 2$ outcomes. Furthermore, although quasi-arithmetic means are typically defined for scalar-valued functions, we demonstrate that these axioms can be extended to describe quasi-arithmetic means with respect to vector-valued functions, as is necessary for our purposes if $n > 2$. This extension is nontrivial but natural, and to our knowledge has not previously been described.\(^5\)

\(^5\)For $n > 2$, these axioms characterize the class of all QA pooling methods with respect proper scoring rules that satisfy convex exposure, a natural condition that we introduce in Section 5.2.
5.1.4 Related work

Since this work’s contribution is a connection between two well-studied problems (elicitation and aggregation of forecasts), there is naturally a large volume of related work, some of which we have discussed in previous chapters. In this section, we briefly discuss the most closely-related work to ours.

Quasi-arithmetic means Our notion of quasi-arithmetic pooling is an adaptation (and extension to higher dimensions) of the existing notion of quasi-arithmetic means. These were originally defined and axiomatized independently by Kolmogorov (1930) and Nagumo (1930). Aczél (1948) generalized this work to include weighted quasi-arithmetic means, though these means have weights baked in rather than taking them as inputs, which is different from our setting. See Grabisch et al. (2011, §3.1) for an overview of this topic.

Arbitrage from collusion In Chapter 4 we discussed the problem of colluding experts: suppose that a principal uses a proper scoring rule $s$ to elicit forecasts from multiple experts. If the experts have different opinions, they can collude to report the forecast, thus earning them a larger total score than if they had been honest, no matter the outcome. We did not detail how exactly this collusion works, but the answer turns out to be precisely QA pooling: for the case of $n = 2$ outcomes, Chun and Shachter (2011) showed that if all experts report what we are calling the QA pool of their opinions with respect to $s$, then their total score is guaranteed to increase. Our Theorem 5.3.1 recovers this result as a special case.

Connections to Bregman divergence In part, our work provides an alternative interpretation of prior work on forecast aggregation via minimizing Bregman divergence, see e.g. Adamčík (2014) and Pettigrew (2019). Concretely, Pettigrew (2019, §4) defines a notion of aggregation analogous to ours, though in a different context. The main focus of their line of work is on connecting forecast aggregation to Bregman divergence; our approach connects aggregation with proper scoring rules,
and a connection to Bregman divergence falls naturally out of this pursuit.

**Dual averaging** One perspective on QA pooling is that, instead of directly averaging experts’ forecasts, QA pooling prescribes considering forecasts as elements in the dual space of gradients (of the function \( G \)) and taking the average in this space before converting the result back to the primal space of probabilities. Gradient methods in online machine learning often take the *sum* of gradients of losses. Taking the average is of gradients is a less ubiquitous technique known as *dual averaging*, which was introduced by Nesterov (2009) and generalized further by Xiao (2010). However, the contexts of QA pooling and dual averaging are quite different.

**Aggregation via prediction markets** In Chapter 1, we saw market scoring rules (MSRs) as a way of aggregating forecasts. Chen and Pennock (2007) introduced *cost-function markets*, in which a market maker sells \( n \) types of shares – one for each outcome – where the price of a share depends on the number of shares sold thus far according to some cost function. They established a connection between cost-function markets and MSRs, where a market with a given cost function will behave the same way as a certain MSR. In particular, the cost function \( C \) of a cost function market is the convex dual of the expected score function \( G \) of the proper scoring rule associated with the corresponding MSR (Abernethy et al., 2013, §8.3).

QA pooling has a simple interpretation in terms of cost function markets: for the cost function market corresponding to the scoring rule \( s \), let \( q_i \) be the quantity vector that implies each expert \( i \)’s probability \( p_i \) (or in other terms, \( i \) would buy a bundle \( q_i \) of shares as the first participant in the market). Then the QA pool with respect to \( s \) is the probability implied by the weighted average quantity vector \( \sum_i w_i q_i \).

Our work is also superficially similar to Abernethy et al. (2013) and Chen and Vaughan (2010), which tied cost-function market making to online learning of probability distributions. Our results on online learning are in a different context: the goal of their online learning problem is to learn a *probability distribution over outcomes*, whereas our goal in Section 5.4 is to learn *expert weights*. 
5.2 Preliminaries

Throughout this work, we will let \( m \) be the number of experts and use the index \( i \) to refer to any particular expert. We will let \( n \) be the number of outcomes and use the index \( j \) to refer to any particular outcome. We will let \( \Delta_n \) be the standard in \( \mathbb{R}^n \) and \( \delta_j \) be the basis-aligned unit vector with a 1 in the \( j \)-th coordinate.

5.2.1 Proper scoring rules

We have already introduced proper scoring rules, along with their Savage representations, in Section 2.1. However, it will serve us well to be more precise in this chapter. First, we will clarify the range of possible scores: we will allow \( s(x; j) \), for \( x \in \Delta_n \) and \( j \in [n] \), to be any real number, or negative infinity. For example, \( s_{\log}((1, 0); 2) = -\infty \).

We define the forecast domain \( D \) associated with \( s \) to be the set of forecasts \( x \) such that \( s(x; j) \) is real-valued for all \( j \). For example, if \( s = s_{\log} \), then \( D \) is the interior of \( \Delta_n \). When discussing forecast aggregation, we will assume that all forecasts belong to \( D \).

We will let \( G \) denote the expected score function of \( s \). We will assume that \( s(\cdot; j) \) is continuous on \( \Delta_n \) for all \( j \), as is \( G(\cdot) \); to our knowledge, this is the case for all frequently-used proper scoring rules.

In Section 2.1 we introduced the Savage representation of a proper scoring rule. We do so here again, this time more carefully.

**Proposition 5.2.1** (Gneiting and Raftery, 2007, Theorem 2). A regular scoring rule \( s \) is proper if and only if

\[
s(x; j) = G(x) + \langle g(x), \delta_j - x \rangle, \tag{5.2}
\]

6This choice removes from consideration cases such as two experts reporting \((1, 0)\) and \((0, 1)\) under the logarithmic scoring rule; aggregating these forecasts using our method is tantamount to adding positive and negative infinity.

7It will typically be clear which scoring rule we are working with, so there is no need to write \( G_s \).

8For \( s \), continuity is with respect to the standard topology on \( \mathbb{R} \cup \{-\infty\} \), i.e. the one that includes sets of the form \([-\infty, r) \) as open sets. This means that e.g. the log scoring rule is continuous on all of \( \Delta_n \).
for some convex function $G : \Delta_n \to \mathbb{R}$ and subgradient function$^9$ $g$ of $G$. The function $G$ is then the expected score function of $s$.

Since we are assuming that $s$ and $G$ are continuous, it follows from Equation 5.2 that $g$ must be continuous as well.$^{10}$ A convex function with a continuous finite subgradient is differentiable (Bauschke & Combettes, 2011, Proposition 17.41), which means that $G$ is differentiable on the interior of $\Delta_n$, with gradient $g$.

As discussed in Section 2.1, the key intuition of Equation 5.2 is that the score of an expert who reports probability distribution $p$ is determined by drawing the tangent plane to $G$ at $p$; the value of this plane at $\delta_j$, where $j$ is the outcome that happens, is the expert’s score.

We refer to $g$ as the exposure function of $s$. We borrow this term from finance, where exposure refers to how much an agent stands to gain or lose from various possible outcomes – informally speaking, how much the agent cares about which outcome will happen. If we view $G(p) - \langle g(p), p \rangle$ as the agent’s “baseline profit,” then the $j$-th component of $g(p)$ is the amount that the agent stands to gain (or lose) on top of the baseline profit if outcome $j$ happens.

Remark 5.2.2. Because the domain of $G$ is a subset of $\Delta_n$ (and thus lies in a plane that is orthogonal to the all-ones vector $1_n$), it makes the most sense to think of its gradient function $g$ as taking on values in $\mathbb{R}^n$ modulo translation by the all-ones vector $1_n$; we will denote this space by $\mathbb{R}^n / T(1_n)$.

Sometimes we find it convenient to treat $G$ as a function of $n$ variables rather than $n - 1$ variables out of convenience, thus artificially extending the domain of $G$ outside of the plane containing $\Delta_n$. The component of the gradient of $G$ that is parallel to $1_n$ is not relevant.$^{11}$

Next, recall the Bregman divergence from Section 2.1.3:

$^9$That is, $g$ satisfies $G(y) \geq G(x) + \langle g(x), y - x \rangle$ for all $x, y$. Note that $g_i(x)$ may be $-\infty$ if $x_i = 0$; see Waggoner (2021) for an examination of subgradients of functions to the extended reals.

$^{10}$Continuity of each component $g_\ell$ of $g$ (for $\ell \in [n]$) is (as with $s$) with respect to the standard topology on $\mathbb{R} \cup \{-\infty\}$. To see that $g_\ell$ is continuous at a given $p$, consider the limit of Equation 5.2 as $x \to p$ with $j = \ell$ if $p_j = 0$ and $j \neq \ell$ if $p_j \neq 0$.

$^{11}$Formally, consider the change of coordinates given by $z_j = x_n - x_j$ for $j \leq n - 1$ and $z_n = \sum_j x_j$, so that the domain of $G$ lies in the plane $z_n = 1$. Then for $j \leq n - 1$, $\frac{\partial G}{\partial z_j}$ at a given point in the domain of $G$ does not change if 1 is substituted for $z_n$; only $\frac{\partial G}{\partial z_n}$ changes (to zero). Equivalently in terms of our original coordinates, the change that $g$ undergoes when we consider $G$ to be a function only defined on $\mathcal{D}$ instead of $\mathbb{R}^n$ is precisely a projection of $g$ onto $\{x \in \mathbb{R}^n : \sum_i x_i = 0\}$.
**Definition 2.1.7.** For some \( n \geq 1 \), let \( D \) be a convex subset of \( \mathbb{R}^n \), and let \( G : D \to \mathbb{R} \) be a differentiable, convex function. Given \( x, y \in D \), the Bregman divergence from \( y \) to \( x \) with respect to \( G \) is defined as

\[
D_G(y \parallel x) := G(y) - G(x) - \langle y - x, \nabla G(x) \rangle.
\]  

(2.2)

We note the following well-known facts about Bregman divergence:

**Proposition 5.2.3** (Well-known facts about Bregman divergence). Let \( G \) be a differentiable, strictly convex function. Then:

- \( D_G(p \parallel q) \geq 0 \), with equality only when \( p = q \).
- For every \( q \), \( D_G(x \parallel q) \) is a strictly convex function of \( x \).

Finally, we make a note about interpreting the \( n = 2 \) outcome case in one dimension.

**Remark 5.2.4.** Because \( \Delta_n \) is \( (n - 1) \)-dimensional, we can think of the case of \( n = 2 \) outcomes in one dimension. All probabilities in are of the form \((p, 1 - p)\); we map \( \Delta_2 \) to \([0, 1]\) via the first coordinate. Thus, we let \( G(p) := G(p, 1 - p) \). We let \( g(p) := G'(p) = \langle g(p, 1 - p), (1, -1) \rangle \). The tangent line to \( G \) at \( p \) will intersect the line \( x = 1 \) at \( s(p; 1) \) (i.e. the score if Outcome 1 happens) and intersect the line \( x = 0 \) at \( s(p; 2) \) (i.e. the score if Outcome 2 happens). See Figure 2.1 for an illustration.

The formulation in Remark 5.2.4 will be helpful when discussing the two-outcome case, e.g. in Section 5.6.

### 5.2.2 Quasi-arithmetic pooling

We now introduce the central concept of this chapter: quasi-arithmetic pooling.

**Definition 5.2.5** (quasi-arithmetic pooling). Let \( s \) be a proper scoring rule with expected score function \( G \) and exposure function \( g \). Given forecasts \( p_1, \ldots, p_m \in D \) with non-negative weights \( w_1, \ldots, w_m \) adding to 1, the quasi-arithmetic (QA) pool of these forecasts with respect to \( s \) (or
with respect to \( g \), denoted by \( \bigoplus_{i=1}^{m} (p_i, w_i) \), is the unique \( p^* \in \Delta_n \) such that \( \sum_{i=1}^{m} w_i g(p_i) \) is a subgradient of \( G \) at \( p^* \).

If the forecasts and weights are clear from context, we may simply write \( p^* \) to refer to their quasi-arithmetic pool; or, if only the forecasts are clear, we may write \( p^*_w \), where \( w \) is the vector of weights.

**Remark 5.2.6.** Equivalently (Rockafellar, 1970a, Theorem 23.5), we can define

\[
\bigoplus_{i=1}^{m} (p_i, w_i) := \arg \min_{x} G(x) - \left< x, \sum_{i=1}^{m} w_i g(p_i) \right>.
\]

Also equivalently, we can define

\[
\bigoplus_{i=1}^{m} (p_i, w_i) := \arg \min_{x} \sum_{i=1}^{m} w_i D_G(x || p_i).
\]

The expression being minimized here differs from the one in Equation 5.3 by a constant: namely, \( \sum_i w_i(\langle p_i, g(p_i) \rangle - G(p_i)) \).

The QA-pool is well-defined, i.e. exists and is unique. It is unique because a strictly convex function cannot have the same subgradient at two different points (Waggoner, 2021, Lemma 3.11). It exists because \( G(x) - \left< x, \sum_i w_i g(p_i) \right> \) is continuous and thus attains its minimum on the (compact) domain \( \Delta_n \).

In light of the fact that \( D_G(p || q) \) is the expected loss in score by an expert who, believing \( p \), instead reports \( q \) (as discussed in Section 2.1.3), the Bregman divergence formulation of QA pooling gives another natural interpretation.

**Remark 5.2.7.** Consider a proper scoring rule \( s \) with forecasts \( p_1, \ldots, p_m \) with weights \( w_1, \ldots, w_m \). The QA pool of these forecasts is the forecast \( p^* \) that, if it is the correct answer (i.e. if the outcome is drawn according to \( p^* \)), would minimize the expected loss of a randomly chosen (according to \( w \)) expert relative to reporting \( p^* \).
In this sense, QA pooling reflects a *compromise* between experts: it is the probability that, if it were correct, would make the experts’ forecast least wrong overall.

**Remark 5.2.8.** Since the Bregman divergence is convex in its first argument, computing the QA pool is a matter of convex optimization. In particular, given oracle access to $g$, the ellipsoid method (see e.g. Grötschel et al. (1993)) can be used to efficiently find the QA pool of a list of forecasts.

Note that although Definition 5.2.5 only specifies that $p^* \in \Delta_n$, in fact it lies in $\mathcal{D}$:

**Claim 5.2.9.** For any $p_1, \ldots, p_m$ and $w_1, \ldots, w_m$, the QA pool $p^*$ lies in $\mathcal{D}$.

**Proof.** Let $x \in \Delta_n \setminus \mathcal{D}$. We show that $\sum_i w_i g(p_i)$ is not a subgradient of $G$ at $x$.

We have $s(x; j) = -\infty$ for some $j$ (satisfying $x_j = 0$), so $\langle g(x), \delta_j - x \rangle = -\infty$. Since $g$ is continuous, for sufficiently small $\epsilon$, we have

$$\langle g(x + \epsilon' (\delta_j - x)), \delta_j - x \rangle < \left( \sum_i w_i g(p_i), \delta_j - x \right)$$

for all $\epsilon' \leq \epsilon$. This means that

$$G(x + \epsilon (\delta_j - x)) - G(x) < \left( \sum_i w_i g(p_i), \epsilon (\delta_j - x) \right),$$

so $\sum_i w_i g(p_i)$ is not a subgradient of $G$ at $x$, as desired. \hfill \square

While our max-min optimality result (Section 5.3) holds unconditionally, our results in Section 5.4 and 5.6 require that our proper scoring rule $s$ satisfy a property that we term *convex exposure*.

**Definition 5.2.10** (convex exposure). A proper scoring rule $s$ with forecast domain $\mathcal{D}$ has convex exposure if the range of its exposure function $g$ on $\mathcal{D}$ is a convex set.

The key fact about proper scoring rules with convex exposure is that for all $p_1, \ldots, p_m$ and $w_1, \ldots, w_m$, $\sum_i w_i g(p_i) = g(x)$ for some $x \in \mathcal{D}$. This means that $\sum_i w_i g(p_i)$ is a subgradient of $G$. 

111
at \( x \), so \( x \) is the weighted QA pool of the \( p_i \)'s. In other words, if \( s \) has convex exposure, then we may write

\[
g(p^*) = \sum_i w_ig(p_i),
\]

(5.4)

where \( p^* \) is the weighted QA pool of the given forecasts. The convex exposure property thus allows us to write down relations between exposures of forecasts that would otherwise not necessarily be true.

The quadratic and logarithmic scoring rules, as well as all proper scoring rules for binary outcomes, have convex exposure. In Appendix B.3 we explore in more depth the question of which commonly used proper scoring rules have convex exposure property.

### 5.3 QA pooling as a max-min optimal method

Our goal is to give a formal justification for quasi-arithmetic pooling. Remark 5.2.7 established that the QA pool is optimizes (i.e. minimizes) the weighted average Bregman divergence to the experts’ forecasts. This section gives another justification for QA pooling in terms of max-min optimality. We will give additional justifications in later sections.

**Theorem 5.3.1.** Let \( s \) be a proper scoring rule and let \( g \) be the exposure function of \( s \). Fix any forecasts \( p_1, \ldots, p_m \in D\) with non-negative weights \( w_1, \ldots, w_m \) adding to 1. Define

\[
u(p; j) := s(p; j) - \sum_{i=1}^m w_is(p_i; j).
\]

Then the quantity \( \min_j \nu(p; j) \) is uniquely maximized by setting \( p \) to \( p^* := \bigoplus_{i=1}^m (p_i, w_i) \). Furthermore, this minimum (across outcomes \( j \)) is achieved simultaneously by all \( j \) with \( p^*_j > 0 \). This quantity is non-negative, and is positive unless all reports \( p_i \) with positive weights are equal.

One interpretation for this theorem statement is as follows. Consider an agent who is tasked with submitting a forecast, and who will be paid according to \( s \). The agent decides to sub-contract
$m$ experts to get their opinions, paying expert $i$ the amount $w_is(p_i; j)$ if the expert reports $p_i$ and outcome $j$ happens. (Perhaps experts whom the agent trusts more have higher $w_i$'s.) Finally, the agent reports some (any) forecast $p$. Then $u(p; j)$ is precisely the agent’s profit (utility).

The quantity $\min_j u(p; j)$ is the agent’s minimum possible profit over all outcomes. It is natural to ask which report $p$ maximizes this quantity. Theorem 5.3.1 states that this maximum is achieved by the QA pool of the experts’ forecasts with respect to $s$, and that this is the unique maximizer.

A possible geometric intuition to keep in mind for the proof (below): for each expert $i$, draw the plane tangent to $G$ at $p_i$. For any $j$, the value of this plane at $\delta_j$ is $s(p_i; j)$. Now take the weighted average of all $m$ planes; this is a new plane whose total score received by the experts if $j$ happens. Since $G$ is convex, this plane lies below $G$. To figure out which point maximizes the agent’s guaranteed profit, push the plane upward until it hits $G$. It will hit $G$ at $p^*$ and the agent’s worst-case profit will be the vertical distance that the plane was pushed.

**Proof of Theorem 5.3.1.** By Equation 5.3, computing the QA pool amounts to finding the minimizer $p^*$ of the function $G(x) - \langle x, \sum_{i=1}^m w_i g(p_i) \rangle$ over $\Delta_n$. If $p^*$ is in the interior of $\Delta_n$, then this expression is differentiable at $p^*$. If $p^*$ is on the boundary, then the expression can be extended to a differentiable function in a neighborhood around $p^*$.

12 This follows e.g. from Azagra and Mudarra (2015, Theorem 1.8), where we take $C$ in the theorem statement to be a compact subset of $\Delta_n$ containing $p^*$ where $G$ is differentiable. Here we use that $p^* \in \mathcal{D}$; by continuity of $g$, such a subset necessarily exists.
\[
G(p^*) - \sum_i w_i G(p_i) + \left( \sum_i w_i g(p_i) + \lambda 1_n + \mu, \delta_j - p^* \right) - \sum_i w_i \langle g(p_i), \delta_j - p_i \rangle
\]

\[
G(p^*) - \sum_i w_i G(p_i) + \sum_i w_i \langle g(p_i), p_i - p^* \rangle + \langle \lambda 1_n + \mu, \delta_j - p^* \rangle
\]

\[
G(p^*) - \sum_i w_i G(p_i) + \sum_i w_i \langle g(p_i), p_i - p^* \rangle + \mu_j
\]

\[
\sum_i w_i D_G(p^* \parallel p_i) + \mu_j.
\]

The second-to-last step follows from the fact that \( \langle \lambda 1_n, \delta_j - p^* \rangle = 0 \) and \( \langle \mu, p^* \rangle = 0 \), and the last step follows by the definition of Bregman divergence. Since \( \mu_j = 0 \) for every \( j \) such that \( p_j^* > 0 \) (of which there is at least one), the minimum of \( u(p^*; j) \) over \( j \) is achieved simultaneously for all \( j \) with \( p_j^* > 0 \), and this minimum is equal to \( \sum_i w_i D_G(p^* \parallel p_i) \). This quantity is non-negative, and positive except when all \( p_i \)'s with positive weights are equal.

Finally we show that \( x = p^* \) maximizes \( \min_j u(x; j) \). Suppose that for some report \( q \) we have that \( \min_j u(q; j) \geq \min_j u(p^*; j) \). Then \( u(q; j) \geq u(p^*; j) \) for every \( j \) such that \( p_j^* > 0 \). But this means that the expected score (according to \( s \)) of an expert who believes \( p^* \) is larger if the expert reports \( q \) than if the expert reports \( p^* \). This contradicts the fact that \( s \) is proper.

\[ \square \]

Remark 5.3.2. We can reformulate Theorem 5.3.1 as follows: suppose that an agent has access to forecasts \( p_1, \ldots, p_m \) and needs to issue a forecast, for which the agent will be scored using a proper scoring rule \( s \). The agent can improve upon selecting an expert at random according to weights \( w_1, \ldots, w_m \), no matter the outcome \( j \), by reporting \( p^* \). This improvement is the same no matter the outcome – so long as the outcome is assigned positive probability by \( p^* \) – and is a strict improvement unless all forecasts with positive weights are the same.

Remark 5.3.3. Theorem 5.3.1 is closely related to work by Grünwald and Dawid (2004) that establishes a connection between entropy maximization and worst-case expected loss minimization. Their work studies a generalized notion of entropy functions that, for the case of a proper scoring rule \( s \), is equal to the negative of the expected score function \( G \). They show that the forecast \( x \) that maximizes entropy (minimizes \( G \)) also maximizes an expert’s worst-case score (over outcomes \( j \)).
Considering instead the entropy function \( \langle x, \sum_i w_i g(p_i) \rangle - G(x) \) yields results that are very similar to our Theorem 5.3.1. In particular, if \( D \) is a closed set or an open set, then the max-min result in Theorem 5.3.1 can be derived from Grünwald and Dawid (2004, Theorem 5.2) and Grünwald and Dawid (2004, Theorem 6.2), respectively. However, our proof below captures cases that their results do not address. See also Chambers et al. (2019, Lemma 1), from which (upon considering the scoring rule with expected score function \( G(x) - \langle x, \sum_i w_i g(p_i) \rangle \)) yields the simultaneity result of Theorem 5.3.1 if \( p^* \) lies in the interior of \( \Delta_n \).

### 5.4 Convex losses and learning expert weights

Thus far, when discussing QA pooling, we have regarded expert weights as given. Where do these weights come from? As demonstrated by the results in this section, if the proper scoring rule \( s \) has convex exposure (see Definition 5.2.10), these weights can be learned from experience. This learning property for weights falls out of the following key observation, which states that an agent’s score is a concave function of the weights it uses for the experts.

**Theorem 5.4.1.** Let \( s \) be a proper scoring rule with convex exposure and forecast domain \( D \), and fix any \( p_1, \ldots, p_m \in D \). Given a weight vector \( w = (w_1, \ldots, w_m) \in \Delta_m \), define the weight-score of \( w \) for an outcome \( j \) as

\[
WS_j(w) := s \left( \bigoplus_{i=1}^m g(p_i, w_i); j \right).
\]

Then for every \( j \in [n] \), \( WS_j(w) \) a concave function of \( w \).

**Proof.** Let \( v \) and \( w \) be two weight vectors. We wish to show that for any \( c \in [0, 1] \), we have

\[
WS_j(c v + (1-c)w) - c WS_j(v) - (1-c)WS_j(w) \geq 0.
\]
Recall the notation $p^*_w$ from Definition 5.2.5. Note that

$$g(p^*_{cv+(1-c)w}) = \sum_{i=1}^{m} (cv_i + (1 - c)w)g(p_i) = cg(p^*_v) + (1 - c)g(p^*_w). \quad (5.5)$$

We have

$$WS_f(cv + (1 - c)w) - cWS_f(v) - (1 - c)WS_f(w)$$

$$= s(p^*_w; j) - cs(p^*_v; j) - (1 - c)s(p^*_w; j)$$

$$= G(p^*_w) + \langle g(p^*_w), \delta_j - p^*_w \rangle$$

$$- c(G(p^*_v) + \langle g(p^*_v), \delta_j - p^*_v \rangle) - (1 - c)(G(p^*_w) + \langle g(p^*_w), \delta_j - p^*_w \rangle)$$

$$= G(p^*_w) - \langle g(p^*_w), p^*_w \rangle$$

Step 1 follows from the definition of WS. Step 2 follows from Equation 5.2. Step 3 follows from Equation 5.5, and specifically that the inner product of each side with $\delta_j$ is the same (so the $\delta_j$ terms cancel out, leaving a quantity that does not depend on $j$). Continuing where we left off:

$$WS_f(cv + (1 - c)w) - cWS_f(v) - (1 - c)WS_f(w)$$

$$= G(p^*_w) - c \langle g(p^*_v), p^*_w \rangle - (1 - c) \langle g(p^*_w), p^*_w \rangle$$

$$- c(G(p^*_v) - \langle g(p^*_v), p^*_v \rangle) - (1 - c)(G(p^*_w) - \langle g(p^*_w), p^*_w \rangle)$$

$$= c(G(p^*_w) - G(p^*_v)) + (1 - c)(G(p^*_w) - G(p^*_v))$$

$$+ (1 - c)(G(p^*_w) - G(p^*_v))$$

$$= cD_G(p^*_w \parallel p^*_v) + (1 - c)D_G(p^*_w \parallel p^*_v) \geq 0.$$

Step 4 again follows from Equation 5.5. Step 5 is a rearrangement of terms. Finally, step 6 follows from the definition of Bregman divergence, and step 7 follows from the fact that Bregman
divergence is always non-negative. This completes the proof.

**Remark 5.4.2.** Theorem 5.4.1 can be stated in more generality: $s$ need not have convex exposure; it suffices to have that for the particular $p_1, \ldots, p_m$, the QA pool of these forecasts exists for every weight vector.

Beyond Theorem 5.4.1’s instrumental use for no-regret online learning of expert weights (Theorem 5.4.5 below), the result is interesting in its own right. For example, the following fact – loosely speaking, that QA pooling cannot benefit from weight randomization – follows as a corollary. (Recall the definition of $p^*_w$ from Definition 5.2.5.)

**Corollary 5.4.3.** Consider a randomized algorithm $A$ with the following specifications:

- **Input:** a proper scoring rule $s$ with convex exposure, expert forecasts $p_1, \ldots, p_m$.
- **Output:** a weight vector $w \in \Delta_m$.

For any input $s, p_1, \ldots, p_m$ and for every $j$, we have

$$s(p^*_{E[A][w]}; j) \geq E_A [s(p^*_w; j)],$$

where $p^*_x$ denotes the QA pool of $p_1, \ldots, p_m$ with weight vector $x$.

**Remark 5.4.4.** Theorem 5.4.1 would not hold if the pooling operator in the definition of WS were replaced by linear pooling or by logarithmic pooling.\(^\text{13}\) This is an advantage of QA pooling over using the linear or logarithmic method irrespective of the scoring rule.

We now state the no-regret result that we have alluded to. This result is quite strong in that it does not merely achieve low regret compared to the best expert, but in fact compared to the best possible weighted pool of experts in hindsight. This is a substantial distinction, as it is possible for a mixture of experts to substantially outperform any individual expert.

\(^{13}\)For a counterexample to logarithmic pooling, consider $n = 2$, let $s$ be the quadratic scoring rule, $p_1 = (0.1, 0.9)$, $p_2 = (0.5, 0.5)$, $j = 1$, $v = (1, 0)$, $w = (0, 1)$, and $c = \frac{1}{2}$. For a counterexample to linear pooling, consider $n = 2$, let $s$ be given by $G(p_1, p_2) = \sqrt{p_1^2 + p_2^2}$ (this is known as the spherical scoring rule), $p_1 = (0, 1)$, $p_2 = (0.2, 0.8)$, $j = 1$, $v = (1, 0)$, $w = (0, 1)$, and $c = \frac{1}{2}$. 117
Theorem 5.4.5. Let $s$ be a bounded proper scoring rule with convex exposure and forecast domain $\mathcal{D}$. For time steps $t = 1 \ldots T$, an agent chooses a weight vector $w_t^i \in \Delta_m$. The agent then receives a score of

$$s \left( \bigoplus_{i=1}^{m} (p^i_t, w^i_t); j^i \right),$$

where $p^i_1, \ldots, p^i_m \in \mathcal{D}$ and $j^i \in [n]$ are chosen adversarially. By choosing $w^i$ according to Algorithm 5.4.6 (online gradient descent on the experts’ weights), the agent achieves $O(\sqrt{T})$ regret in comparison with the best weight vector in hindsight. In particular, if $M$ is an upper bound\footnote{This bound $M$ exists because $s$ is bounded by assumption, and so $g$ is also bounded (this follows from Equation 5.2).} on $\|g\|_2$, then for every $w^* \in \Delta_m$ we have

$$\sum_{t=1}^{T} s \left( \bigoplus_{i=1}^{m} (p^i_t, w^*_i); j^i \right) - s \left( \bigoplus_{i=1}^{m} (p^i_t, w^i_t); j^i \right) \leq 3\sqrt{mM} \sqrt{T}.$$ 

**Algorithm 5.4.6:** Online gradient descent algorithm for Theorem 5.4.5

- \( w^1 \leftarrow (1/m, \ldots, 1/m); \)
- \( \text{for } t = 1 \text{ to } T \text{ do} \)
  - \( \eta_t \leftarrow \frac{1}{M\sqrt{mt}}; \)
  - Play $w^t$ and observe loss $L^t(w^t)$;
  - $\tilde{w}^{t+1} \leftarrow w^t - \eta_t \nabla L^t(w^t)$;
  - $w^{t+1} \leftarrow$ projection of $\tilde{w}^{t+1}$ onto $\Delta_m$;
- \( \text{end} \)

Algorithm 5.4.6, referenced in the theorem statement, is an application of the standard online gradient descent algorithm (see e.g. Hazan (2023, Theorem 3.1)) to our particular setting. In our context, the loss function is $L^t(w) = -\text{WS}_{j^t}(w)$, where WS is as in Theorem 5.4.1, relative to forecasts $p^1_t, \ldots, p^m_t$. We defer the proof of Theorem 5.4.5 to Appendix B.1. The proof amounts to applying the standard bounds for online gradient descent, though with an extra step: we use the bound $M$ on $\|g\|$ to bound the gradient of the loss as a function of expert weights.
5.5 QA pooling connects two notions of overconfidence

Overconfidence is a well-studied phenomenon in which forecasters are systematically biased toward reporting probabilities that are too extreme. For example, suppose that an expert assigns probabilities to a large number of events, and that, among those events to which the expert assigns a 1% probability, 10% end up happening. This is a sign of overconfidence: the expert’s forecasts would be better if all of their forecasts of 1% were systematically increased.

What, formally, is meant by overconfidence? Arguably, a definition of overconfidence ought to depend on the proper scoring rule used to elicit or grade the expert’s forecast. This is because – as we have already discussed – a proper scoring rule expresses a subjective opinion about which probabilistic distinctions are important. If an expert assigns a 0.01% probability to an outcome that has a 1% chance of occurring, is this a larger mistake than if the expert assigns a 40% probability to an outcome that has a 41% chance of occurring? The logarithmic scoring rule says yes; the quadratic scoring rule says no.\(^{15}\)

In this section, we present two definitions of overconfidence with respect to a proper scoring rule. The first definition is natural and well-motivated, but (to our knowledge) has not previously appeared in the literature. The second definition adapts a standard notion of overconfidence to respect the intuition of the previous paragraph by making use of QA pooling. We then prove that the two definitions are equivalent.

For both definitions, we consider the following setting: an expert forecasts \(p_1, \ldots, p_N \in \Delta_n\) for \(N\) events. Outcomes \(j_1, \ldots, j_N \in [n]\) are realized. We are defining what it means for the expert to be overconfident on this sample of \(N\) events.\(^{16}\)

The first definition uses a quite simple principle: an overconfident expert expects a larger score than they will receive.

\(^{15}\)Formally, recall that the Bregman divergence from the true probability to the reported forecast with respect to the expected score function \(G\) measures the score that the expert loses out on due to their inaccuracy. We have \(D_{\text{log}}(1\% \parallel 0.01\%) \gg D_{\text{quad}}(41% \parallel 40\%),\) whereas \(D_{\text{quad}}(1\% \parallel 0.01\%) \approx D_{\text{quad}}(41\% \parallel 40\%).\)

\(^{16}\)We define overconfidence with respect to a particular realization of the outcomes (as opposed to on average over realizations) so that our definitions do not require a prior over the probabilities of the outcomes.
**Definition 5.5.1** (Expected score definition of overconfidence). Suppose that an expert forecasts $p^1, \ldots, p^N \in \Delta_n$, and outcomes $j^1, \ldots, j^N \in [n]$ are realized. The expert is expected-score overconfident with respect to a proper scoring rule $s$ (with expected score function $G$) if their expected score exceeds their actual score:

$$\sum_{k=1}^{N} G(p^k) > \sum_{k=1}^{N} s(p^k; j^k).$$

On a basic level, the intuition behind Definition 5.5.1 is straightforward: an overconfident expert will overestimate their score. But to gain some further intuition, recall Figure 2.1 (repeated here). It shows the score of an expert who reports a probability of $x = 40\%$, under the No outcome (the red dot on the left) and under the Yes outcome (the red dot on the right). Suppose that, in the setting of Definition 5.5.1, an expert reports 40\% for all $N$ events. Then the expert expects a score of $G(0.4)$ per event on average: more than that under a No outcome (assuming $G$ is symmetric about 0.5, as shown) and less than that under a Yes outcome, but $G(0.4)$ on average.

If in fact 40\% of the events happen, then the expert is correctly calibrated: neither over- nor underconfident. Their total score will be $N \cdot G(0.4)$ and so the left-hand side of Equation 5.6 will equal the right-hand side. If only 20\% of the events happen, then the expert is underconfident (should have assigned more extreme probabilities) – and indeed, the expert’s total score will be *higher* than the expert expects. And if more than 40\% of the events happen, then the expert is overconfident, and indeed, that is the case in which Equation 5.6 holds.
Our second definition also uses a simple principle: *an overconfident expert would be better off assigning less extreme probabilities.* This basic notion has previously appeared in the literature. For example, Schanbacher (2014) uses the following definition: an expert is overconfident with respect to a proper scoring rule if shrinking all of the expert’s forecasts toward the uniform distribution by some positive constant factor will improve the expert’s score. Formally, an expert is overconfident by this definition if there exists \( w \in (0, 1) \) such that

\[
\sum_k s\left(w\mathbf{p}^k + (1-w)(1/n, \ldots, 1/n); j^k\right) > \sum_k s(\mathbf{p}^k; j^k).
\]

Our definition is similar in spirit, but importantly different: why shrink the expert’s report toward the uniform distribution by a constant factor? Again, it seems that the correct form of shrinkage ought to depend on \( s \). Perhaps the quadratic score treats 26\% as being equally far from 2\% as it is from 50\% (and so shrinking 2\% halfway toward 50\% would produce a result of 26\%), but the log score certainly does not: shrinking 2\% halfway toward 50\% “with respect to” the log scoring rule should give something much less than 26\%. And so our definition instead uses the exposure function \( g \) of \( s \). Much as QA pooling averages the exposures of the experts’ forecasts, our method of shrinking a forecast is to average its exposure with the zero vector. Intuitively this means uniformly making the forecast “more indifferent” to the eventual outcome.

**Definition 5.5.2.** For a proper scoring rule \( s \), we define \( \text{Shrink}_s(\mathbf{p}, w) \) to be the forecast \( \mathbf{p}^* \) such that \( wg(\mathbf{p}) \) is a subgradient of \( G \) at \( \mathbf{p}^* \), where \( g \) is the exposure function of \( s \).

Note that \( \text{Shrink}_s(\mathbf{p}, 1) = \mathbf{p} \), i.e. \( w = 1 \) means no shrinkage (whereas \( w = 0 \) means complete shrinkage).

**Remark 5.5.3.** In the case that there exists \( \mathbf{q} \) such that \( g(\mathbf{q}) = 0 \) – i.e. a forecast \( \mathbf{q} \) such that the score of \( \mathbf{q} \) does not depend on the outcome – we have

\[
\text{Shrink}_s(\mathbf{p}, w) = (\mathbf{p}, w) \oplus (\mathbf{q}, 1-w).
\]
For this reason, we will refer to Shrink$_s$ as QA shrinkage with respect to $s$. Such a $q$ typically exists for proper scoring rules that are used in practice. Indeed, for any proper scoring rule that treats all outcomes symmetrically, $q$ is simply the uniform distribution $(1/n, \ldots, 1/n)$. In particular, QA shrinkage with respect to the quadratic scoring rule is equivalent to averaging with the uniform distribution. That is, in the case of $s_{quad}$, our definition coincides with that of Schanbacher (2014). On the other hand, QA shrinkage with respect to $s_{log}$ is equivalent to taking a logarithmic pool of the forecast with the uniform distribution.

**Definition 5.5.4** (QA shrinkage definition of overconfidence). Suppose that an expert forecasts $p^1, \ldots, p^N \in \Delta_n$, and outcomes $j^1, \ldots, j^N \in [n]$ are realized. The expert is QA-shrinkage overconfident with respect to a proper scoring rule $s$ if for some $w \in (0, 1)$, we have

$$\sum_{k=1}^N s(\text{Shrink}_s(p^k, w); j^k) > \sum_{k=1}^N s(p^k; j^k).$$

(5.7)

We can now state our main theorem, which is that (under some weak assumptions about $s$) these two definitions are the same.

**Theorem 5.5.5.** Let $s$ be a proper scoring rule with convex exposure, for which there is a $q \in \Delta_n$ such that $g(q) = 0$. Suppose that an expert forecasts $p^1, \ldots, p^N \in \Delta_n$, and outcomes $j^1, \ldots, j^N \in [n]$ are realized. The expert is expected-score overconfident with respect to $s$ if and only if they are QA-shrinkage overconfident with respect to $s$.

**Proof.** First, observe that by using the Savage representation of $s$, we may rewrite the condition for expected-score overconfidence (Equation 5.6) as

$$\sum_{k=1}^N \langle g(p^k), \delta_{j^k} - p^k \rangle < 0.$$

We now show that Definition 5.5.4 is equivalent. By Remark 5.5.3, we may write $\text{Shrink}_s(p^k, w) = (p^k, w) \oplus (q, 1 - w)$, where $q$ is as in the theorem statement. Furthermore, because $s$ has convex exposure, Equation 5.4 tells us that $g(\text{Shrink}_s(p^k, w)) = wzg(p^k)$.  

122
We now use Theorem 5.4.1, which tells us that for all $k$, the quantity

$$s \left( (p^k, w) \oplus_s (q, 1 - w); j^k \right)$$

is concave in $w$. This means that the left-hand side of Equation 5.7 is a concave function (call it $f$) of $w$. On the other hand, the right-hand side of Equation 5.7 is $f(1)$. The question of whether there exists $w \in (0, 1)$ such that $f(w) > f(1)$ is therefore equivalent to the question of whether the derivative of $f$ at $w = 1$ is negative.\footnote{Or more precisely, the left derivative, since $f$ is only defined on $[0, 1]$.} To evaluate this derivative, we note that

$$s(\operatorname{Shrink}_s(p^k, w); j^k) = G(\operatorname{Shrink}_s(p^k, w)) + \left< g(\operatorname{Shrink}_s(p^k, w)), \delta_{j^k} - \operatorname{Shrink}_s(p^k, w) \right>$$

$$= G(\operatorname{Shrink}_s(p^k, w)) + w \left< g(p^k), \delta_{j^k} - \operatorname{Shrink}_s(p^k, w) \right>.$$

Therefore, we have

$$\frac{\partial s(\operatorname{Shrink}_s(p^k, w); j^k)}{\partial w} = \frac{\partial G(\operatorname{Shrink}_s(p^k, w))}{\partial w} + \frac{\partial \left( w \left< g(p^k), \delta_{j^k} - \operatorname{Shrink}_s(p^k, w) \right> \right)}{\partial w}.$$

Considering $\operatorname{Shrink}_s(p^k, w)$ as a function from $[0, 1]$ to $\mathbb{R}^n$, let $J(w)$ be its Jacobian, i.e. the vector of partial derivatives of the $n$ coordinates with respect to $w$. Applying the rules of differentiation, we have that

$$\frac{\partial s(\operatorname{Shrink}_s(p^k, w); j^k)}{\partial w} = \left< g(\operatorname{Shrink}_s(p^k, w)), J(w) \right> + \left< g(p^k), \delta_{j^k} - \operatorname{Shrink}_s(p^k, w) \right>$$

$$- w \left< g(p^k), J(w) \right>.$$

Now, we are specifically interested in the derivative at $w = 1$. Plugging in $w = 1$, we have

$$\frac{\partial s(\operatorname{Shrink}_s(p^k, w); j^k)}{\partial w} \bigg|_{w=1} = \left< g(p^k), J(1) \right> + \left< g(p^k), \delta_{j^k} - p^k \right> - \left< g(p^k), J(1) \right> = \left< g(p^k), \delta_{j^k} - p^k \right>.$$
Recalling that the expert is QA-shrinkage overconfident if and only if the derivative of \( f \) at \( w = 1 \) is negative, this is the case if and only if

\[
\sum_{k=1}^{N} \langle g(p^k), \delta_j - p^k \rangle < 0.
\]

As we have already showed, this is the case precisely when the expert is expected-score overconfident. This completes the proof.

\[\square\]

### 5.6 Axiomatization of QA pooling

In this section, we aim to show that the class of all quasi-arithmetic pooling operators is a natural one, by showing that these operators are precisely those which satisfy a natural set of axioms.

Kolmogorov (1930) and Nagumo (1930) independently considered the class of quasi-arithmetic means. Given an interval \( I \subseteq \mathbb{R} \) and a continuous, injective function \( f : I \rightarrow \mathbb{R} \), the quasi-arithmetic mean with respect to \( f \), or \( f \)-mean, is the function \( M_f \) that takes as input \( x_1, \ldots, x_m \in I \) (for any \( m \geq 1 \)) and outputs

\[
M_f(x_1, \ldots, x_m) := f^{-1}\left( \frac{f(x_1) + \cdots + f(x_m)}{m} \right).
\]

For example, the arithmetic mean corresponds to \( f(x) = x \); the quadratic to \( f(x) = x^2 \); the geometric to \( f(x) = \log x \); and the harmonic to \( f(x) = \frac{1}{x} \).

Kolmogorov proved that the class of quasi-arithmetic means is precisely the class of functions \( M : \bigcup_{m=1}^{\infty} I^m \rightarrow I \) satisfying the following natural properties:\(^{\text{18}}\)

1. \( M(x_1, \ldots, x_m) \) is continuous and strictly increasing in each variable.
2. \( M \) is symmetric in its arguments.
3. \( M(x, x, \ldots, x) = x \).

\(^{\text{18}}\)Nagumo also provided a characterization, though with slightly different properties.
\begin{align*}
(4) \quad & M(x_1, \ldots, x_k, x_{k+1}, \ldots, x_m) = M(y, \ldots, y, x_{k+1}, \ldots, x_m), \text{ where } y := M(x_1, \ldots, x_k) \text{ appears } \\
& \quad k \text{ times on the right-hand side. Informally, a subset of arguments to the mean function can be replaced with their mean.}
\end{align*}

The four properties listed above can be viewed as an \textit{axiomatization} of quasi-arithmetic means.

Our notion of quasi-arithmetic pooling is exactly that of a quasi-arithmetic mean, except that it is more general in two ways. First, it allows for weights to accompany the arguments to the mean. Second, we are considering quasi-arithmetic means with respect to \textit{vector-valued} functions \( g \). In the \( n = 2 \) outcome case, \( g \) can be considered a scalar-valued function since it is defined on a one-dimensional space (see Remark 5.2.4 for details); but in general we cannot treat \( g \) as scalar-valued.\(^{19}\)

Our goal is to extend the above axiomatization of quasi-arithmetic means in these two ways: first (in Section 5.6.1) to include weights as arguments, and second (in Section 5.6.2) to general \( n \) (while still allowing arbitrary weights).

\section*{5.6.1 Generalizing to include weights as arguments}

The objects that we will be studying in this section are ones of the form \((p, w)\), where \( w \geq 0 \) and \( p \in D \). In this subsection, \( D \) is a two-outcome forecast domain, whose elements we will identify with by the probability of the first outcome (see Remark 5.2.4).\(^{20}\) We will fix the set \( D \) for the remainder of the subsection. Our results generalize to any interval of \( \mathbb{R} \) (as in Kolmogorov’s work), but we focus on forecast domains since that is our application.

\textbf{Definition 5.6.1.} A \textbf{weighted forecast} is an element of \( D \times \mathbb{R}_{>0} \): a probability and a positive weight. Given a weighted forecast \( \Pi = (p, w) \) we define \( pr(\Pi) := p \) and \( wt(\Pi) := w \).

\(^{19}\) Another difference is that in the \( n = 2 \) case, we are restricting \( D \) to be an interval from 0 to 1, though this is not a fundamental difference for the purposes of this section.

\(^{20}\) Proper scoring rules for two outcomes can have four possible forecast domains: \([0, 1]\), \([0, 1)\), \((0, 1]\), and \((0, 1)\).
We will thinking of the output of pooling operators as weighted forecasts. This is a simple extension of our earlier definition of quasi-arithmetic pooling (Definition 5.2.5), which only output a probability.

**Definition 5.6.2** (Quasi-arithmetic pooling with arbitrary weights \((n = 2)\)). Given a continuous, strictly increasing function \(g : \mathcal{D} \to \mathbb{R}\), and weighted forecasts \(\Pi_1 = (p_1, w_1), \ldots, \Pi_m = (p_m, w_m)\), define the quasi-arithmetic pool of \(\Pi_1, \ldots, \Pi_m\) with respect to \(g\) as

\[
\bigoplus_{i=1}^m g(p_i, w_i) := \left(g^{-1}\left(\frac{\sum_i w_i g(p_i)}{\sum_i w_i}\right), \sum_i w_i\right).
\]

**Remark 5.6.3.** The equivalence of this notion to our earlier one uses the fact that all (continuous) proper scoring rules for binary outcomes have convex exposure (Proposition B.3.1), so the relationship between \(g(p^*)\) and \(g(p)\) may be written as in Equation 5.4.

In the case that \(\sum_i w_i = 1\), Definition 5.6.2 reduces to Definition 5.2.5. In general, by linearly scaling the weights in Definition 5.6.2 to add to 1, we recover quasi-arithmetic pooling as previously defined.

We find the following fact useful.

**Proposition 5.6.4.** Given two continuous, strictly increasing functions \(g_1\) and \(g_2\), \(\oplus_{g_1}\) and \(\oplus_{g_2}\) are the same if and only if \(g_2 = ag_1 + b\) for some \(a > 0\) and \(b \in \mathbb{R}\).

**Proof.** Clearly if \(g_2 = ag_1 + b\) for some \(a > 0\) and \(b \in \mathbb{R}\) then \(\oplus_{g_1}\) and \(\oplus_{g_2}\) are the same. For the converse, suppose that no such \(a\) and \(b\) exist. Let \(x < y\) be such that \(g_1\) and \(g_2\) are not equal (even up to positive affine transformation) on \([x, y]\). Let \(g'_2\) be the positive affine transformation of \(g_2\) that makes it equal to \(g_1\) at \(x\) and \(y\), and let \(z \in (x, y)\) be such that \(g_1(z) \neq g'_2(z)\). Let \(\alpha\) be such that \(g_1(z) = \alpha g_1(x) + (1 - \alpha)g_1(y)\). Then \((x, \alpha) \oplus_{g_1} (y, 1 - \alpha) = (z, 1)\), but \((x, \alpha) \oplus_{g_2} (y, 1 - \alpha) \neq (z, 1)\), so \(\oplus_{g_1}\) and \(\oplus_{g_2}\) are different. \(\square\)

We now define properties (i.e. axioms) of a pooling operator \(\oplus\), such that these properties are satisfied if and only if \(\oplus\) is \(\oplus_g\) for some \(g\). Our axiomatization will look somewhat different from
Kolmogorov’s, in part because we choose to define $\oplus$ as a binary operator that (if it satisfies the associativity axiom) extends to the $m$-ary case. This is a simpler domain and will simplify notation. In Appendix B.2 we exhibit an equivalent set of axioms that more closely resembles Kolmogorov’s.

**Definition 5.6.5** (Axioms for pooling operators ($n = 2$)). For a pooling operator $\oplus$ on $D$ (i.e. a binary operator on weighted forecasts), we define the following axioms.

1. **Weight additivity**: $wt(\Pi_1 \oplus \Pi_2) = wt(\Pi_1) + wt(\Pi_2)$ for every $\Pi_1, \Pi_2$.

2. **Commutativity**: $\Pi_1 \oplus \Pi_2 = \Pi_2 \oplus \Pi_1$ for every $\Pi_1, \Pi_2$.

3. **Associativity**: $\Pi_1 \oplus (\Pi_2 \oplus \Pi_3) = (\Pi_1 \oplus \Pi_2) \oplus \Pi_3$ for every $\Pi_1, \Pi_2, \Pi_3$.

4. **Continuity**: For every $p_1, p_2$, the quantity $pr((p_1, w_1) \oplus (p_2, w_2))$ is a continuous function of $(w_1, w_2)$ on $\mathbb{R}^2_{\geq 0} \setminus \{(0, 0)\}$.

5. **Idempotence**: For every $\Pi_1, \Pi_2$, if $pr(\Pi_1) = pr(\Pi_2)$ then $pr(\Pi_1 \oplus \Pi_2) = pr(\Pi_1)$.

6. **Monotonicity**: Let $w > 0$ and let $p_1 > p_2 \in D$. Then for $x \in (0, w)$, the quantity $pr((p_1, x) \oplus (p_2, w - x))$ is a strictly increasing function of $x$.

The motivation for the weight additivity axiom is that the weight of a weighted forecast can be thought of as the *amount of evidence* for its prediction. When pooling weighted forecasts, the weight of an individual forecast can be thought of as the strength of its vote in the aggregate.

The monotonicity axiom essentially states that if one pools two forecasts with different probabilities and a fixed total weight, then the larger the share of the weight belonging to the larger of the two probabilities, the larger the aggregate probability.

We now state and prove this section’s main result: these axioms describe the class of QA pooling operators.

**Theorem 5.6.6.** A pooling operator is a QA pooling operator (as in Definition 5.6.2) with respect to some $g$ if and only if it satisfies the axioms in Definition 5.6.5.\(^{22}\)

\(^{21}\)We allow one weight to be 0 by defining $(p, w) \oplus (q, 0) = (q, 0) \oplus (p, w) = (p, w)$.

\(^{22}\)As we mentioned, for an associative pooling operator $\oplus$, $\Pi_1 \oplus \Pi_2 \cdots \oplus \Pi_m$ is a well-specified quantity, even without indicating parenthesization. This lets us use the notation $\bigoplus_{i=1}^{m} \Pi_i$. This is why the statement of Theorem 5.6.6 makes sense despite pooling operators not being $m$-ary by default.
We will use \( \oplus \) (without a \( g \) subscript) to denote an arbitrary pooling operator that satisfies the axioms in Definition 5.6.5. Before presenting the proof of Theorem 5.6.6, we will note a few important facts about weighted forecasts and pooling operators. First, we find it natural to define a notion of multiplying a weighted forecast pair by a positive constant.

**Definition 5.6.7.** Given a weighted forecast \( \Pi = (p, w) \) and \( c > 0 \), define \( c\Pi := (p, cw) \).

Note that \( m\Pi = \bigoplus_{i=1}^{m} \Pi \) for any positive integer \( m \), by idempotence; this definition is a natural extension to all \( c > 0 \). We note the following (quite obvious) fact.

**Proposition 5.6.8.** For every weighted forecast \( \Pi \) and \( c_1, c_2 > 0 \), we have \( c_1(\Pi) \oplus (c_2\Pi) = (c_1c_2)\Pi \).

A natural property that is not listed in Definition 5.6.5 is **scale invariance**, i.e. that \( \Pr((p_1, w_1) \oplus (p_2, w_2)) = \Pr((p_1, cw_1) \oplus (p_2, cw_2)) \) for any positive \( c \); or, equivalently, that \( c(\Pi_1 \oplus \Pi_2) = c\Pi_1 \oplus c\Pi_2 \). This in fact follows from the listed axioms.

**Proposition 5.6.9** (Distributive property/scale invariance). For every \( \Pi_1, \Pi_2 \) and any operator \( \oplus \) satisfying the axioms in Definition 5.6.5, we have \( c(\Pi_1 \oplus \Pi_2) = c\Pi_1 \oplus c\Pi_2 \).

**Proof.** First suppose \( c \) is an integer. Then

\[
c\Pi_1 \oplus c\Pi_2 = \bigoplus_{i=1}^{c} \Pi_1 \oplus \bigoplus_{i=1}^{c} \Pi_2 = \bigoplus_{i=1}^{c} (\Pi_1 \oplus \Pi_2) = c(\Pi_1 \oplus \Pi_2).
\]

Here, the first and last steps follow by weight additivity and idempotence. Now suppose that \( c = \frac{k}{\ell} \) is a rational number. Let \( \Pi_1' = \frac{1}{\ell}\Pi_1 \) and \( \Pi_2' = \frac{1}{\ell}\Pi_2 \). We have

\[
\frac{k}{\ell}(\Pi_1 \oplus \Pi_2) = \frac{k}{\ell}(\ell\Pi_1' \oplus \ell\Pi_2') = \frac{k}{\ell} \cdot \ell(\Pi_1' \oplus \Pi_2') = k(\Pi_1' \oplus \Pi_2') = k\Pi_1' \oplus k\Pi_2' = \frac{k}{\ell}\Pi_1 \oplus \frac{k}{\ell}\Pi_2.
\]

Here, the second and second-to-last steps follow from the fact that the distributive property holds for integers.

Finally, make use of the continuity axiom to extend our proof to all positive real numbers \( c \). In particular, it suffices to show that \( \Pr(\Pi_1 \oplus \Pi_2) = \Pr(c\Pi_1 \oplus c\Pi_2) \). Let \( p \) be the former...
quantity; note that \( \text{pr}(r \Pi_1 \oplus r \Pi_2) = p \) for positive rational numbers \( r \). Since the rationals are dense among the reals, it follows that for every \( \epsilon > 0 \), we have \( |\text{pr}(c \Pi_1 \oplus c \Pi_2) - p| \leq \epsilon \). Therefore, \( \text{pr}(c \Pi_1 \oplus c \Pi_2) = p \). This completes the proof.

Armed with these facts, we present a proof of Theorem 5.6.6.

**Proof of Theorem 5.6.6.** We first prove that any QA pooling operator \( \oplus_g \) satisfies the axioms in Definition 5.6.5. Weight additivity, commutativity, and idempotence are trivial. Associativity is also clear: given \( \Pi_1 = (p_1, w_1) \) and likewise \( \Pi_2, \Pi_3 \), we have

\[
g(\text{pr}(\Pi_1 \oplus_g \Pi_2 \oplus_g \Pi_3)) = \frac{(w_1 + w_2)g(\text{pr}(\Pi_1 \oplus_g \Pi_2)) + w_3g(p_3)}{(w_1 + w_2) + w_3}
= \frac{(w_1 + w_2)\frac{w_1g(p_1) + w_2g(p_2)}{w_1 + w_2} + w_3g(p_3)}{w_1 + w_2 + w_3}
= \frac{w_1g(p_1) + w_2g(p_2) + w_3g(p_3)}{w_1 + w_2 + w_3}
\]

and likewise for \( g(\text{pr}(\Pi_1 \oplus_g (\Pi_2 \oplus_g \Pi_3))) \), so \( \text{pr}((\Pi_1 \oplus_g \Pi_2) \oplus_g \Pi_3) = \text{pr}(\Pi_1 \oplus_g (\Pi_2 \oplus_g \Pi_3)) \) (since \( g \) is strictly increasing and therefore injective). The fact that the weights are also the same is trivial. Continuity follows from the fact that

\[
\text{pr}(\Pi_1 \oplus_g \Pi_2) = g^{-1}\left(\frac{w_1g(p_1) + w_2g(p_2)}{w_1 + w_2}\right)
\]

is continuous in \((w_1, w_2)\) (when \(w_1, w_2\) are not both zero). Here we are using the fact that \( g \) is strictly increasing, which means that \( g^{-1} \) is continuous.

Finally, regarding the monotonicity axiom, for any fixed \( w \) and \( p_1 > p_2 \) (as in the axiom statement), we have

\[
g(\text{pr}((p_1, x) \oplus_g (p_2, w-x))) = \frac{yg(p_1) + (w-x)g(p_2)}{x + w - x} = \frac{yg(p_1) + (w-x)g(p_2)}{w}.
\]

Since \( p_1 > p_2 \), we have \( g(p_1) > g(p_2) \), so the right-hand side strictly increases with \( x \). Since \( g^{-1} \) is also strictly increasing, it follows that \( \text{pr}((p_1, x) \oplus_g (p_2, w-x)) \) strictly increases with \( x \).
The converse – that every pooling operator satisfying the axioms in Definition 5.6.5 is $\oplus_g$ for some $g$ – works by constructing $g$ by fixing it at two points and constructing $g$ at all other points. Right now we show how to do this when the forecast domain is $[0, 1]$; see the proof of Theorem B.2.7 for the argument in full generality.

Let $\oplus$ be a pooling operator that satisfies our axioms. Define $g$ as follows: let $g(0) = 0$ and $g(1) = 1$. For $0 < p < 1$, define $g(p) = w$ where $(1, w) \oplus (0, 1 - w) = (p, 1)$. (This $w$ exists by continuity and the intermediate value theorem; it is unique by the “strictly” increasing stipulation of monotonicity.) Note that $g$ is continuous and increasing by monotonicity.\(^{23}\)

We wish to show that for any $\Pi_1 = (p_1, w_1)$ and $\Pi_2 = (p_2, w_2)$, we have that $\Pi_1 \oplus \Pi_2 = \Pi_1 \oplus_g \Pi_2$. Clearly the weight of both sides is $w_1 + w_2$, so we wish to show that the probabilities on each side are the same. We have\(^{24}\)

\[
\begin{align*}
\text{pr}(\Pi_1 \oplus \Pi_2) &= \text{pr}(w_1(p_1, 1) \oplus w_2(p_2, 1)) \\
&= \text{pr}(w_1((1, g(p_1)) \oplus (0, 1 - g(p_1))) \oplus w_2((1, g(p_2)) \oplus (0, 1 - g(p_2)))) \\
&= \text{pr}(w_1(1, g(p_1)) \oplus (0, 1 - g(p_1)) \oplus w_2(1, g(p_2)) \oplus w_2(0, 1 - g(p_2))) \\
&= \text{pr}((1, w_1 g(p_1)) \oplus (0, w_1(1 - g(p_1))) \oplus (1, w_2 g(p_2)) \oplus (0, w_2(1 - g(p_2)))) \\
&= \text{pr}((1, w_1 g(p_1) + w_2 g(p_2)) \oplus (0, w_1(1 - g(p_1)) + w_2(1 - g(p_2)))) \\
&= \text{pr} \left( \frac{1}{w_1 + w_2} \left( (1, w_1 g(p_1) + w_2 g(p_2)) \oplus (0, w_1(1 - g(p_1)) + w_2(1 - g(p_2))) \right) \right) \\
&= \text{pr} \left( \frac{1}{w_1 + w_2} \left( \frac{w_1 g(p_1) + w_2 g(p_2)}{w_1 + w_2} \right) \oplus \left( 0, \frac{w_1(1 - g(p_1)) + w_2(1 - g(p_2))}{w_1 + w_2} \right) \right),
\end{align*}
\]

which by definition of $g$ is equal to the probability $p$ such that $g(p) = \frac{g(p_1)w_1 + g(p_2)w_2}{w_1 + w_2}$. That is, $\text{pr}(\Pi_1 \oplus \Pi_2) = \text{pr}(\Pi_1 \oplus_g \Pi_2)$.

---

\(^{23}\)As a matter of fact, $g$ is strictly increasing because it is impossible for $g(p_1)$ to equal $g(p_2)$ for $p_1 \neq p_2$, as that would mean that $(1, g(p_1)) \oplus (0, 1 - g(p_1)) = (p_1, 1) = (p_2, 1)$. Another way to look at this is that it comes from the fact that $(1, w) \oplus (0, 1 - w)$ is continuous in $w$ by the continuity axiom. In a sense, the continuity of $g$ corresponds to the strictness of increase in the monotonicity axiom and the strictness of increase of $g$ corresponds to the continuity axiom.

\(^{24}\)Steps 3 and 7 uses the distributive property (Proposition 5.6.9).
Showing that $\oplus$ and $\oplus_g$ are equivalent for more than two arguments is now trivial:

$$
\bigoplus_{i=1}^{m} \Pi_i = \Pi_1 \oplus_g \Pi_2 \oplus_g \Pi_3 \cdots \oplus_g \Pi_m = \Pi_1 \oplus \Pi_2 \oplus_g \Pi_3 \cdots \oplus_g \Pi_m = \cdots = \bigoplus_{i=1}^{m} \Pi_i.
$$

(Here we are implicitly using the fact that $\oplus_g$ is associative, as we proved earlier.) This completes the proof. □

### 5.6.2 Generalizing to higher dimensions

In Appendix B.2.1, we generalize our axioms from $n = 2$ outcomes to arbitrary values of $n$. An important challenge is extending the monotonicity axiom: in higher dimensions, what is the appropriate generalization of an increasing function? We show that the correct notion is cyclically monotonicity, which we define and discuss. We then present our axiomatization (Definition B.2.2) and prove that the axioms constitute a characterization of the class of QA pooling operators (Theorem B.2.7). On a high level, the proof is not dissimilar to that of Theorem 5.6.6, though the details are fairly different and more technical.

In conclusion, in Definition 5.6.5 we made a list of natural properties that a pooling operator may satisfy. Theorem 5.6.6 shows that the pooling operators satisfying these properties are exactly the QA pooling operators. In Appendix B.2.1, we generalize this theorem to higher dimensions, thus fully axiomatizing QA pooling. This result gives us an additional important reason to believe that QA pooling with respect to a proper scoring rule is a fundamental notion.

### 5.7 Conclusions and future directions

While in this work we have focused on proper scoring rules for eliciting probability distributions over possible outcomes, scoring rules can be used to elicit various other properties, such as the expectation or the median of a random variable (Gneiting & Raftery, 2007; Lambert, Pen-
nock, & Shoham, 2008; Savage, 1971). The topic of property elicitation studies such scoring rules. The Savage representation of a proper scoring rule generalizes to arbitrary linear properties (Frongillo & Kash, 2015, Theorem 11). That is, consider a random variable (or n-tuple of random variables) $X$ and a convex function $G$ whose domain is (a superset of) the possible values of $X$. If $p$ is an expert’s forecast for $\mathbb{E}[X]$ and $x$ is the realized outcome, then the scoring rule $s(p; x) := G(p) + \langle g(p), x - p \rangle$ is proper.\footnote{In the setting of this work, $X$ is the vector of random variables where the $j$-th variable is 1 if outcome $j$ happens and 0 otherwise, and $x = \delta_j$ where $j$ is the outcome that happens.} Modulo a fairly straightforward generalization, all proper scoring rules for $\mathbb{E}[X]$ take this form (see Section 2.1.3).

Our definition of QA pooling extends verbatim to the setting of eliciting linear properties. In this more general setting, for any proper scoring rule that has convex exposure, it remains the case $g(p^*) = \sum_i w_i g(p_i)$; as a consequence, Theorems 5.3.1, 5.4.1, 5.4.5, and 5.5.5 generalize.

We now discuss several promising directions for future work. First: as we discussed in Section 2.2.3, it is natural to generalize QA pooling by dropping the requirement that weights add to 1. In Section 2.4, we gave a Bayesian justification for linear pooling with arbitrary weights, and gave a different and novel Bayesian justification for logarithmic pooling with arbitrary weights. This raises a few questions:

- Which of our results generalize if the requirement that weights add to 1 is dropped? Note that dropping this requirement means that the idempotence axiom of Section 5.6 is no longer satisfied. Is there an axiomatization that characterizes the class of all generalized QA pools?
- Just as we gave Bayesian justifications for generalized linear and logarithmic pooling, is there a Bayesian justification for generalized QA pooling with respect to an arbitrary proper scoring rule? What does it look like?

Second: it is natural to wonder about connections to prediction markets. After all, a prediction market is a mechanism for eliciting forecasts from multiple experts and aggregating them! As mentioned in Section 5.1.4, QA pooling can be interpreted in terms of cost function markets: if
\( q_i \) denotes the quantity of shares that each expert would buy in order to bring a market into line with the expert’s beliefs, then the QA pool of the experts’ beliefs is the probability distribution corresponding to the quantity vector \( q^* := \sum_i w_i q_i \). This follows from Abernethy et al. (2013, Eq. 22); the underlying reason is the convex duality between an expected score function \( G \) and the corresponding cost function \( C \). Exploring this connection further may yield insights into forecast aggregation through market mechanisms.

Third: some of our results may be able to be extended or generalized. In fact, we do so in Chapter 6, where we relax the assumption made in Theorem 5.4.5 that the scoring rule is bounded. Additionally, many of our theorem statements depend on the convex exposure property, but it is possible that some of our definitions and theorem statements could be modified so as to not require this assumption.

QA pooling connects forecast elicitation and aggregation in a way that seems quite fundamental. Further research in this area may yield important insights.
Chapter 6: Learning weights for logarithmic pooling

This chapter presents “No-Regret Learning with Unbounded Losses: The Case of Logarithmic Pooling” (Neyman & Roughgarden, 2022b). It assumes knowledge of logarithmic pooling, presented in Section 2.2.1. Background on proper scoring rules (Section 2.1), as well as on information structures (Section 2.3), is useful for context but not required. Additionally, Chapter 5 is useful for context, as this work builds on it directly.

Summary: For each of $T$ time steps, $m$ experts report probability distributions over $n$ outcomes; we wish to learn to aggregate these forecasts in a way that attains a no-regret guarantee. We focus on logarithmic pooling, which is in a certain sense the optimal choice of pooling method if one is interested in minimizing log loss (see Theorem 5.3.1). We consider the problem of learning the best set of parameters (i.e. expert weights) in an online adversarial setting. We assume (by necessity) that the adversarial choices of outcomes and forecasts are consistent, in the sense that experts report calibrated forecasts. Imposing this constraint creates a (to our knowledge) novel semi-adversarial setting in which the adversary retains a large amount of flexibility. In this setting, we present an algorithm based on online mirror descent that learns expert weights in a way that attains $O(\sqrt{T} \log T)$ expected regret as compared with the best weights in hindsight.

6.1 Introduction

6.1.1 Learning expert weights for logarithmic pooling

In Section 2.2, we motivated logarithmic pooling as a natural method of forecast aggregation that has compelling theoretical properties and works well in practice. To recall, the logarithmic
pool of forecasts $\mathbf{p}^1, \ldots, \mathbf{p}^m \in \Delta_n$ with weight vector $\mathbf{w} \in \Delta_m$ is defined by

$$
p_j^*(\mathbf{w}) = c(\mathbf{w}) \prod_{i=1}^{m} (p_j^i)^{w_i},
$$

for all $j \in [n]$, where $c(\mathbf{w})$ is a normalizing constant (that depends on the weights).

But where do the expert weights come from? That is, how should an aggregator determine what weight to assign to each expert? It is common for the aggregator to learn the correct vector of weights over time, based on the experts’ track record (Tetlock & Gardner, 2015). Learning expert weights over time is a sequential decision making problem: for each of $T$ time steps, experts submit forecasts to the aggregator, and the aggregator aggregates them using logarithmic pooling, with weights that the aggregator chooses. Then an outcome is realized and the aggregator receives a score depending on the accuracy of the aggregate forecast. The aggregator adjusts expert weights for the next time step, perhaps increasing the weights of experts who did well while decreasing the weights of experts who did poorly.

This problem falls into the well-studied and wide-reaching field of online learning – and more specifically, online prediction with expert advice (Cesa-Bianchi & Lugosi, 2006). Generally, algorithms for online learning problems are assessed based on their regret relative to some baseline. In our case, the natural baseline is the best possible weight vector in hindsight. That is, we are looking for a weight selection algorithm whose overall performance over the $T$ time steps is guaranteed to be almost as good as using the best weight vector $\mathbf{w}^*$ in hindsight.

We previously discussed online learning of expert weights in Chapter 5. In that chapter, we introduced quasi-arithmetic (QA) pooling as a way of aggregating forecasts with respect to a given proper scoring rule $s$. Specifically, the QA pool with respect to a proper scoring rule $s$ is the forecast that guarantees the largest possible expected overperformance (as judged by the scoring rule) compared with the strategy of choosing a random expert to trust.\footnote{More formally, the QA pool maximizes the minimum improvement in the score (over possible outcomes).} We saw that QA pooling with respect to the quadratic scoring rule is linear pooling (averaging the experts’ forecasts), while QA pooling with respect to the logarithmic scoring rule is logarithmic pooling.
In Section 5.4, we gave an algorithm for online-learning expert weights for QA pooling with respect to any **bounded** proper scoring rule \( s \). The algorithm had the property that, asymptotically with the number of time steps \( T \), the average score of the algorithm (according to \( s \)) was guaranteed to be almost as large as the average score of an aggregator who chose the best vector of weights in hindsight. In online learning terminology, this means that the algorithm has *vanishing regret* (or, synonymously, *no regret*).

However, the log scoring rule is not bounded, so the aforementioned result does not give a no-regret algorithm for learning weights for logarithmic pooling (if regret is defined with respect to the log scoring rule). The purpose of this chapter is to fill that gap.

*A note on terminology*: In this chapter, we will use the *log loss* instead of the log score, as is standard in the learning literature. The log loss is simply the negative of the log score: while the log score of the forecast \( p \) under outcome \( j \) is \( \ln(p_j) \), the log loss is \( -\ln(p_j) \).

### 6.1.2 Choosing the right benchmark

In Chapter 5, we established a connection between the log loss and logarithmic pooling, arguing that if the log loss is used for elicitation, then it makes sense to use logarithmic pooling for aggregation. The goal of this work is to develop an algorithm for learning weights for logarithmic pooling in a way that achieves vanishing regret as judged by the log loss function. Within the field of online prediction with expert advice, this is a particularly challenging setting. In part, this is because the losses are potentially unbounded. However, that is not the whole story: finding weights for *linear* pooling, by contrast, is a well-studied problem that has been solved even in the context of log loss. On the other hand, because logarithmic pooling behaves more as a geometric than an arithmetic mean, if some expert assigns a very low probability to the eventual outcome (and the other experts do not) then the logarithmic pool will also assign a low probability, incurring a large loss. This makes the combination of logarithmic pooling with log loss particularly difficult.

We require that our algorithm not have access to the experts’ forecasts when choosing weights:
an algorithm that chooses weights in a way that depends on forecasts can output an essentially arbitrary function of the forecasts, and thus may do something other than learn optimal weights for logarithmic pooling. For example, suppose that \( m = n = 2 \) and an aggregator wishes to subvert our intentions and take an equally weighted linear pool of the experts’ forecasts. Without knowing the experts’ forecasts, this is impossible; on the other hand, if the aggregator knew that e.g. \( \mathbf{p}_1 = (90\%, 10\%) \) and \( \mathbf{p}_2 = (50\%, 50\%) \), they could assign weights for logarithmic pooling so as to produce the post-hoc desired result, i.e. \((70\%, 30\%)\). We wish to disallow this.

One might suggest the following setup: at each time step, the algorithm selects weights for each expert. Subsequently, an adversary chooses each expert’s forecast and the outcome, after which the algorithm and each expert incur a log loss. Unfortunately – due to the unboundedness of log loss and the behavior of logarithmic pooling – vanishing regret guarantees in this setting are impossible.

**Example 6.1.1.** Consider the case of \( m = n = 2 \). Without loss of generality, suppose that the algorithm assigns Expert 1 a weight \( w \geq 0.5 \) in the first time step. The adversary chooses reports \((e^{-T}, 1 - e^{-T})\) for Expert 1 and \(\left(\frac{1}{2}, \frac{1}{2}\right)\) for Expert 2, and for Outcome 1 to happen. The logarithmic pool of the forecasts turns out to be approximately \((e^{-wT}, 1 - e^{-wT})\), so the algorithm incurs a log loss of approximately \(wT \geq 0.5T\), compared to \(O(1)\) loss for Expert 2. On subsequent time steps, Expert 2 is perfect (assigns probability 1 to the correct outcome), so the algorithm cannot catch up.

What goes wrong in Example 6.1.1 is that the adversary has full control over experts’ forecast and the realized outcome, and is not required to couple the two in any way. This unreasonable amount of adversarial power motivates assuming that the experts are calibrated: for example, if an expert assigns a 10% chance to an outcome, there really is a 10% chance of that outcome (conditional on the expert’s information).

We propose the following setting: an adversary chooses a joint probability distribution over the experts’ beliefs and the outcome – subject to the constraint that each expert is calibrated. Loosely speaking, this means that each expert correctly assesses the probability each outcome based on the information they have: if an expert reports \( \mathbf{p} \), then each event \( j \) really does have a probability \( p_j \)
of happening, based on the expert’s information. (We formally define calibration in Section 6.3.1.)

The adversary retains full control over correlations between forecasts and outcomes, subject to this calibration property. Subsequently, nature randomly samples each expert’s belief and the eventual outcome from the distribution. In this setting, we seek to prove upper bounds on the expected value of our algorithm’s regret.

Why impose this constraint, instead of a different one? Our reasons are twofold: theoretical and empirical. From a theoretical standpoint, the assumption that experts are calibrated is natural because experts who form Bayesian rational beliefs based on evidence will be calibrated, regardless of how much or how little evidence they have. The assumption is also motivated if we model experts as learners rather than Bayesian agents: even if a forecaster starts out completely uninformed, they can quickly become calibrated in a domain simply by observing the frequency of events (Foster & Vohra, 1997).

Second, recent work has shown that modern deep neural networks are calibrated when trained on a proper loss function such as log loss. This is true for a variety of tasks, including image classification (Hendrycks et al., 2020; Minderer et al., 2021) and language modeling (Desai & Durrett, 2020; Kadavath et al., 2022; OpenAI, 2023); see Blasiok et al. (2023) for a review of the literature. We may wish to use an ensemble of off-the-shelf neural networks for some prediction or classification task. If we trust these networks to be calibrated (as suggested by recent work), then we may wish to learn to ensemble these experts (models) in a way that has strong worst-case theoretical guarantees under the calibration assumption.

Logarithmic pooling is particularly sensible in the context of calibrated experts because it takes confident forecasts “more seriously” as compared with linear pooling (simple averaging). If Expert 1 reports probability distribution $(0.1\%, 99.9\%)$ over two outcomes and Expert 2 reports $(50\%, 50\%)$, then the logarithmic pool (with equal weights) is approximately $(3\%, 97\%)$, as compared with a linear pool of roughly $(25\%, 75\%)$. If Expert 1 is calibrated (as we are assuming), then the $(0.1\%, 99.9\%)$ forecast entails very strong evidence in favor of Outcome 2 over Outcome 1. Meanwhile, Expert 2’s forecast gives no evidence either way. Thus, it is sensible for the aggregate
to point to Outcome 2 over Outcome 1 with a fair amount of confidence.

As another example, suppose that Expert 1 reports (0.04%, 49.98%, 49.98%) and Expert 2 reports (49.98%, 0.04%, 49.98%) (a natural interpretation: Expert 1 found strong evidence against Outcome 1 and Expert 2 found strong evidence against Outcome 2). If both experts are calibrated, a sensible aggregate should arguably assign nearly all probability to Outcome 3. Logarithmic pooling returns roughly (2.7%, 2.7%, 94.6%), which (unlike linear pooling) accomplishes this.

Since we are allowing our algorithm to learn the optimal logarithmic pool, perhaps there is hope to compete not just with the best expert in hindsight, but the optimally weighted logarithmic pool of experts in hindsight. We will aim to compete with this stronger benchmark.

This work demonstrates that the “calibrated experts” condition allows us to prove regret bounds when no such bounds are possible for an unrestricted adversary. While that is our primary motivation, the relaxation may also be of independent interest. For example, even in settings where vanishing regret is attainable in the presence of an unrestricted adversary, even stronger regret bounds might be achievable if calibration is assumed.

6.1.3 Our main result

Is vanishing regret possible in our setting? Our main result is that the answer is yes. We exhibit an algorithm that attains expected regret that scales as $O(\sqrt{T \log T})$ with the number of time steps $T$. Our algorithm uses online mirror descent (OMD) with the Tsallis entropy regularizer $R(w) = \frac{1}{\alpha} (w_1^\alpha + \cdots + w_m^\alpha)$ and step size $\eta \approx \frac{1}{\sqrt{T \ln T}}$, where any choice of $\alpha \in (0, 1/2)$ attains the regret bound.

Our proof has two key ideas. One is to use the calibration property to show that the gradient of loss with respect to the weight vector is likely to be small (Section 6.4.4). This is how we leverage the calibration property to turn an intractable setting into one where – despite the unboundedness of log loss and the behavior of logarithmic pooling – there is hope for vanishing regret.

The other key idea (Section 6.4.3) involves keeping track of a function that, roughly speaking, reflects how much “regret potential” the algorithm has. We show that if the aforementioned gradi-
ent updates are indeed small, then this potential function decreases in value at each time step. This allows us to upper bound the algorithm’s regret by the initial value of the potential function.

This potential argument is an important component of the proof. A naïve analysis might seek to use our bounds on the gradient steps to myopically bound the contribution to regret at each time step. Such an analysis, however, does not achieve our $O(\sqrt{T \log T})$ regret bound. In particular, an adversary can force a large accumulation of regret if some experts’ weights are very small (specifically by making the experts with small weights more informed than those with large weights) – but by doing so, the small weights increase and the adversary “spends down” its potential. Tracking this potential allows us to take this nuance into consideration, improving our bound.

We extend our main result by showing that the result holds even if experts are only approximately calibrated: so long as no expert understates the probability of an outcome by more than a constant factor, we still attain the same regret bound (see Corollary 6.4.11). We also show in Section 6.5 that no OMD algorithm with a constant step size can attain expected regret better than $\Omega(\sqrt{T})$.

6.2 Related work

In the subfield of prediction with expert advice, for $T$ time steps, experts report “predictions” from a decision space $\mathcal{D}$ (often, as in our case, the space of probability distributions over a set of outcomes). A forecaster must then output their own prediction from $\mathcal{D}$. Then, predictions are assessed according to a loss function. See Cesa-Bianchi and Lugosi (2006) for an survey of this field.

We are particularly interested in mixture forecasters: forecasters who, instead of choosing an expert to trust at each time step, aggregate the expert’ reports. Linear mixtures, i.e. convex combinations of predictions, have been especially well-studied, generally with the goal of learning weights for the convex combination to compete with the best weights in hindsight. Standard convex optimization algorithms achieve $O(\sqrt{T})$ regret for bounded, convex losses, but it is sometimes
possible to do better. For example, if the loss function is bounded and exp-concave, then logarithmic regret in $T$ is attainable (Cesa-Bianchi & Lugosi, 2006, §3.3).

Portfolio theory studies optimal stock selection for maximizing return on investment, often in a no-regret setting. Cover (1991) introduced the “universal portfolio” algorithm, which, for each of $T$ time steps, selects a portfolio (convex combination of stocks). Our setting translates naturally to Cover’s: experts play the role of stocks, and the return of a stock corresponds to the probability that and expert assigns to the eventual outcome. The universal portfolio algorithm achieves logarithmic regret compared with the best portfolio in hindsight (Cover & Ordentlich, 1996); in our terms, this means that logarithmic regret (for log loss) is attainable for the linear pooling of experts. See Li and Hoi (2014) for a survey of this area.

To our knowledge, learning weights for logarithmic pooling has not been previously studied. As shown in Example 6.1.1, it is not possible to achieve vanishing regret if the setting is fully adversarial. We relax our setting by insisting that the experts be calibrated (see Section 6.3.1). To our knowledge, online prediction with expert advice has also not previously been studied under this condition.

The calibration condition can be equivalently restated as follows: the experts’ forecasts are based on signals drawn from an information structure – possibly a different one at each time step – that is unknown to the aggregator, and experts report accurate probabilities conditional on their signals (see Remark 6.3.2). Babichenko and Garber (2021) investigated online prediction with expert advice under the assumption that the experts’ signals are drawn from a particular type of information structure, which they called the “partial evidence environment.”\(^2\) Under the paper’s assumptions, the Bayesian optimal aggregate is a generalized logarithmic pool (as introduced in Section 2.2.3), and the authors show that an aggregator can often learn this optimal aggregate with vanishing regret. By contrast, we do not make any assumptions about the information structures,

\(^{2}\) Essentially, the partial evidence environment is an adaptation of the partial information framework (see Section 2.3) to the case of binary outcomes. There is a set of signals, which are independent conditional on the outcome, and every expert knows a subset of the signals (the same subset on each time step). Further, the authors assume that knowledge of the information structure would allow a perfect aggregator to infer all relevant information about the experts’ signals from their forecasts.
and instead aim to compete not with the optimal aggregate but with the best weighted mixture of experts in hindsight.

6.3 Preliminaries

6.3.1 Calibration property

We define calibration as follows. Note that the definition is in the context of our setting, i.e. $m$ experts reporting probability distributions $\mathbf{p}^1, \ldots, \mathbf{p}^m$ over $n$ outcomes. We will use $J$ to denote the random variable corresponding to the outcome, i.e. $J$ takes values in $[n]$.

**Definition 6.3.1.** Consider a joint probability distribution $\mathbb{P}$ over experts’ reports and the outcome. We say that expert $i$ is calibrated if for all $\mathbf{p} \in \Delta_n$ and $j \in [n]$, we have that

$$
\Pr \left[ J = j \mid \mathbf{p}^i = \mathbf{p} \right] = p_j.
$$

That is, expert $i$ is calibrated if the probability distribution of $J$ conditional on their report $\mathbf{p}^i$ is precisely $\mathbf{p}^i$. We say that $\mathbb{P}$ satisfies the calibration property if every expert is calibrated.

The key intuition behind the usefulness of calibration is that if an expert claims that an outcome is very unlikely, this is strong evidence that the outcome is in fact unlikely. In Section 6.4.4 we will use the calibration property to show that the gradient of the loss with respect to the weight vector is likely to be relatively small at each time step.

6.3.2 Our online learning setting

The setting for our online learning problem is as follows. For each time step $t \in [T]$:

1. Our algorithm reports a weight vector $\mathbf{w}^t \in \Delta_m$.
2. An adversary (with knowledge of $\mathbf{w}^t$) constructs a probability distribution $\mathbb{P}$, over reports and the outcome, that satisfies the calibration property.
(3) Reports $p^{t,1}, \ldots, p^{t,m}$ and an outcome $j^t$ are sampled from $\mathbb{P}$.

(4) The loss of a weight vector $w$ is defined as $L^t(w) := -\ln(p^*_j(w))$, the log loss of the logarithmic pool of $p^{t,1}, \ldots, p^{t,m}$ with weights $w$. Our algorithm incurs loss $L^t(w')$.

Remark 6.3.2. An equivalent, more mechanistic definition of calibration views $\mathbb{P}$ instead as a joint probability distribution over signals received by each expert and the outcome, i.e. as an information structure (see Section 2.3). Each expert’s probability distribution is then the actual probability distribution over the outcome conditioned on their signal. We can reinterpret our online learning setting in light of this view. Specifically, instead of directly selecting a distribution over the experts’ forecasts and the outcome, the adversary selects an information structure describing the probability distribution over the outcome and each expert’s signal. The signals and outcome are then drawn at random from the information structure, and each expert’s report is the true probability distribution over the outcome conditioned on their signal.

Formally, the adversary selects an information structure $\mathcal{I}_t = (\Omega^t, \mathbb{P}^t, (\sigma^t_1, \ldots, \sigma^t_m), Y^t)$, where $Y^t$ is a random variables taking on one of $n$ possible values, namely the standard basis vectors in $\mathbb{R}^n$. Then, nature randomly samples $\omega^t$ according to $\mathbb{P}^t$. The outcome $j^t$ is the coordinate in which $Y^t(\omega^t)$ has a 1, and each expert $i$’s report $p^{t,i}$ is $\mathbb{E}[Y^t \mid \sigma^t_i(\omega^t)]$.

We define the regret of our algorithm as

$$\text{Regret} = \sum_{t=1}^T L^t(w^t) - \min_{w \in \Delta_m} \sum_{t=1}^T L^t(w).$$

That is, the benchmark for regret is the best weight vector in hindsight. Since our setting involves randomness, our goal is to provide an algorithm with vanishing expected regret against any adversarial strategy, where the expectation is taken over the sampling in Step (3).

Even subject to the calibration property, the adversary has a large amount of flexibility, because the adversary retains control over the correlation between different experts’ forecasts. An unrestricted adversary has exponentially many degrees of freedom (as a function of the number of
experts), whereas the calibration property imposes a mere linear number of constraints.³

6.3.3 Our algorithm

We use Algorithm 6.3.3 to accomplish this goal. The algorithm is online mirror descent (OMD) on the weight vector. Fix any \( \alpha \in (0, 1/2) \). We use the regularizer

\[
R(w) := \frac{-1}{\alpha} (w_1^\alpha + \cdots + w_m^\alpha).
\]

This is known as the Tsallis entropy regularizer; see e.g. Zimmert and Seldin (2021) for previous use in the online learning literature. We obtain the same result (up to a multiplicative factor that depends on \( \alpha \)) regardless of the choice of \( \alpha \). Because no choice of \( \alpha \) stands out, we prove our result for all \( \alpha \in (0, 1/2) \) simultaneously.

We will generally use a step size \( \eta = \frac{1}{\sqrt{T \ln T}} \cdot \frac{1}{12m^{(1+\alpha)/2}n} \). However, in the (unlikely, as we show) event that some expert’s weight becomes unusually small, we will reduce the step size.

In Appendix C, we prove that Algorithm 6.3.3 is efficient, taking \( O(mn) \) time per time step.

³This follows from the perspective of the adversary choosing an information structure from which experts’ signals are drawn. The information structure specifies the probability of every possible combination of signals received by the experts, and thus has dimension that is exponential in the number of experts. The calibration property imposes linearly many constraints on this space.
Theorem 6.3.4 formally states our no-regret guarantee.

**Theorem 6.3.4.** For any adversarial strategy, the expected regret\(^4\) of Algorithm 6.3.3 is at most

\[
O\left(\frac{m^{(3-\alpha)/2} n \sqrt{T} \log T}{\eta}\right).
\]

### 6.4 Proof of no-regret guarantee

In this section, we prove Theorem 6.3.4.

#### 6.4.1 Outline of proof

We use the following fact, which follows from the fact that the score of a QA pool of forecasts is concave in the forecasts’ weights (see Chapter 5).

**Proposition 6.4.1** (Follows from Theorem 5.4.1). Let \(p_1, \ldots, p_m\) be forecasts over \(n\) outcomes, \(j \in [n]\) be an outcome, and \(w \in \Delta_m\) be a weight vector. Let \(p^*(w)\) be the logarithmic pool of the forecasts with weight vector \(w\) and let \(L(w) := -\ln(p^*_j(w))\) be the log loss of \(p^*(w)\) if Outcome \(j\) is realized. Then \(L\) is a convex function.

In particular, all of our loss functions \(L'\) are convex, which means that standard regret bounds apply. In particular, to bound the expected regret of Algorithm 6.3.3, we will use a well-known regret bound for follow the regularized leader (FTRL) with linearized losses (Hazan, 2023, Lemma 5.3), which in our case is equivalent to OMD.\(^5\)

**Lemma 6.4.2** (Follows from Hazan (2023, Lemma 5.3)). If \(\eta_t = \eta\) for all \(t\), the regret of Algorithm 6.3.3 is at most

\[
\frac{1}{\eta} \left( \max_{w \in \Delta_m} R(w) - \min_{w \in \Delta_m} R(w) \right) + \sum_{t=1}^{T} \langle \nabla L'(w'), w' - w^{t+1} \rangle.
\]

\(^4\)The given asymptotics assume that \(T \gg m, n\), i.e. ignore terms that are lower-order in \(T\).

\(^5\)This equivalence is due to our choice of regularizer, as we never need to project \(w'\).
Informally, this bound means that if the vectors $\nabla L_t'(w^t)$ are small in magnitude, our regret is also small. Conversely, if some $\nabla L_t'(w^t)$ is large, this may be bad for our regret bound. We expect the gradient of the loss to be large if some expert is very wrong (assigns a very low probability to the correct outcome), since the loss would then be steeply increasing as a function of that expert’s weight. Fortunately, the calibration property guarantees this to be unlikely. Specifically, we define the small gradient assumption as follows.

**Definition 6.4.3.** Define $\gamma := 12n \ln T$. The small gradient assumption holds for a particular run of Algorithm 6.3.3 if for every $t \in [T]$ and $i \in [m]$, we have

$$\frac{-\gamma}{w^t_i} \leq \partial_i L_t'(w^t) \leq \gamma,$$

where $\partial_i$ denotes the partial derivative with respect to the $i$-th weight.$^6$

In Section 6.4.4, we prove that the small gradient assumption is very likely to hold. This is a key conceptual step in our proof, as it is where we leverage the calibration property to prove bounds that ultimately let us bound our algorithm’s regret. We then use the low likelihood of the small gradient assumption failing in order to bound the contribution to the expected regret from the case where the assumption fails to hold.

In Sections 6.4.2 and 6.4.3, we bound regret under the condition that the small gradient assumption holds. We show that under the assumption, for all $i, t$ we have $(w^t_i)^\alpha \geq \eta$. Consequently, $\eta = \frac{1}{\sqrt{T} \ln T} \cdot \frac{1}{12m^{(1+\alpha)/2}n}$ at all time steps, so we can apply Lemma 6.4.2. The first term in the bound is $O(1/\eta) = O(\sqrt{T} \log T)$. As for the summation term, we upper bound it by keeping track of the following quantity:

$$\varphi(t) := \sum_{s=1}^t \langle \nabla L_s'(w^s), w^s - w^{s+1} \rangle + 19m^2\gamma^2 \eta(T-t) - 4\gamma \sum_{i=1}^m \ln w^t_i + 1.$$  

The first term is exactly the summation in Lemma 6.4.2 up through step $t$. The $19m^2\gamma^2 \eta$ is some-

$^6$See Equation C.5 for an expression of this quantity in terms of the experts’ reports and weights.
thing akin to an upper bound on the value of $\langle \nabla L_t'(w^t), w^t - w^{t+1} \rangle$ at a given time step (times $T-t$ remaining time steps). This upper bound is not strict: in particular, large summands are possible when some weights are small (because of the fact that the lower bound in the small gradient assumption is inversely proportional to $w'_i$). However, attaining a large summand requires these small weights to increase, thus “spending potential” for future large summands. The last term keeps track of this potential.

We show that under the small gradient assumption, $\varphi(t)$ necessarily decreases with $t$. This argument, which we give in Section 6.4.3, is another key conceptual step, and is arguably the heart of the proof. Since $\varphi(T)$ is equal to the summation term in Lemma 6.4.2 (plus a positive number), and $\varphi(0) \geq \varphi(T)$, the summation term is less than or equal to $\varphi(0)$, which is at most $O(m^{(3-\alpha)/2}\sqrt{T} \log T)$. This completes the proof.

**6.4.2 Bounds on $w^t$ under the small gradient assumption**

In this section, we state bounds on expert weights and how quickly they change from one time step to the next, conditional on the small gradient assumption. We use the following lemma, whose proof we defer to Appendix C.

**Lemma 6.4.4.** Consider a particular run of Algorithm 6.3.3. Let $\zeta$ be a constant such that $-\frac{\zeta}{w'_i} \leq \partial_i L_t'(w^t) \leq \zeta$ for all $i, t$. Then for every $i, t$, we have

$$(w'_i)^{\alpha-1} - \left(\frac{1}{w'_i} + 1\right) \eta_t \zeta \leq (w'^{t+1}_i)^{\alpha-1} \leq (w'_i)^{\alpha-1} + \left(\frac{1}{\min_k w_k} + 1\right) \eta_t \zeta.$$ 

Furthermore, if $\eta_t \zeta \leq (1-\alpha)^2 (w'_i)^{\alpha}$ for all $i$, then for every $i$ we have

$$(w'^{t+1}_i)^{\alpha-1} \leq (w'_i)^{\alpha-1} + (m+1) \eta_t \zeta.$$ 

Intuitively, this result states that when the gradient update is small, $w'^{t+1}_i$ is not too different from $w'_i$. Note that the lower bound $-\frac{\zeta}{w'_i}$ that we place on the gradient is not a simple Lipschitz
bound but instead depends on $w'_i$; this makes the bounds in Lemma 6.4.4 less straightforward to prove. In particular, we bound each component $w'_i$ individually, using bounds on the gradient of the loss for all other components and convexity arguments.

Lemma 6.4.4 can be translated into bounds on each $w'_i$ and on the change between $w'_i$ and $w'_{i+1}$:

**Corollary 6.4.5.** Under the small gradient assumption, for sufficiently large $T$ we have for all $i \in [m], t \in [T]$ that:

(I) $(w'_i)^\alpha \leq 4\eta \gamma$ and $w'_i \geq \frac{1}{10 \sqrt{m}} T^{1/(2(\alpha-1))}$.

(II) $-32(w'_i)^{1-\alpha} \eta \gamma \leq w'_i - w'_{i+1} \leq 2(w'_i)^{2-\alpha}(m + 1) \eta \gamma$.

We defer the proof of Corollary 6.4.5 to Appendix C. The key idea for (I) is to proceed by induction on $t$ on the two sub-statements in parallel: so long as $(w'_i)^{\alpha} \geq 4\eta \gamma$, we may use the second part of Lemma 6.4.4 with $\zeta = \gamma$ to bound $(w'_{i+1})^{\alpha-1}$ in terms of $(w'_i)^{\alpha-1}$, which we can leverage to prove both sub-statements for $t + 1$. (II) then follows from (I) by routine (though nontrivial) algebra.

Armed with the bounds of Corollary 6.4.5, we are now able to show that under the small gradient assumption, Algorithm 6.3.3 attains vanishing regret.

### 6.4.3 Bounding regret under the small gradient assumption

Assume the small gradient assumption. Note that since $4\gamma \geq 1$, by Corollary 6.4.5 (I) we have that $\eta_t = \eta$ for all $t$. This means that we may apply the bound in Lemma 6.4.2, and in particular we have

$$\frac{1}{\eta} \left( \max_{w \in \Delta_m} R(w) - \min_{w \in \Delta_m} R(w) \right) = \frac{1}{\eta} \cdot \frac{m}{\alpha} \left( \frac{1}{m} \right)^\alpha \geq \frac{m^{1-\alpha}}{\alpha \eta \sqrt{n} \sqrt{T} \ln T}. $$

It remains to bound the summation component of the regret bound in Lemma 6.4.2. To do so, we prove the following lemma, which we alluded to in Section 6.4.1 as the heart of the proof of Theorem 6.3.4.
Lemma 6.4.6. For $t \in \{0, 1, \ldots, T\}$, let

$$
\varphi(t) := \sum_{s=1}^{t} \langle \nabla L^s(w^s), w^s - w^{s+1} \rangle + 19m^2 \gamma^2 \eta (T - t) - 4\gamma \sum_{i=1}^{m} \ln w_i^{t+1}.
$$

Under the small gradient assumption, for sufficiently large $T$, $\varphi(t)$ is a decreasing function of $t$.

To prove this claim, consider a particular $t \in [T]$. We may write

$$
\varphi(t) - \varphi(t - 1) = \sum_{i=1}^{m} \left( (w_i^t - w_i^{t+1}) \partial_i L^t(w^t) - 19m \gamma^2 \eta + 4\gamma (\ln w_i^t - \ln w_i^{t+1}) \right) \tag{6.1}
$$

and we wish to show that this quantity is negative. In fact, we show that the contribution from every $i \in [m]$ is negative. The key idea is to consider two cases: $w_i^{t+1} \leq w_i^t$ and $w_i^{t+1} \geq w_i^t$. In each case, Corollary 6.4.5 provides an upper bound on the magnitude of the difference between $w_i^t$ and $w_i^{t+1}$. If $w_i^{t+1} \leq w_i^t$ then the first and third terms in the summation are positive but small, and are dominated by the middle term. If $w_i^{t+1} \geq w_i^t$ then the first term may be quite large, because of the asymmetric bound in the small gradient assumption (and the consequently asymmetric bound in Corollary 6.4.5). However, in this case the contribution of the third term is very negative, enough to make the overall expression negative. In this sense, the third term keeps track of unspent potential for future regret, which gets “spent down” whenever a large amount of regret is realized (as measured by the first term).

We now prove formally that each term of the summation in Equation 6.1 is negative.

Proof. First assume that $w_i^t - w_i^{t+1} \leq 0$. Note that by combining (#1) and (#2) of Corollary 6.4.5, we have

$$
w_i^{t+1} - w_i^t \leq 32(w_i^t)^{1-\alpha} \eta \gamma \leq 8w_i^t.
$$

By the small gradient assumption we have that

$$
(w_i^t - w_i^{t+1}) \partial_i L^t(w^t) \leq \frac{\gamma(w_i^{t+1} - w_i^t)}{w_i^t}.
$$
On the other hand, we have

\[ 4\gamma (\ln w_i^T - \ln w_i^{t+1}) = -4\gamma \ln \left( 1 + \frac{w_i^{t+1} - w_i^t}{w_i^t} \right) \leq -\frac{\gamma (w_i^{t+1} - w_i^t)}{w_i^t} \]

for \( T \) large enough. (Here we use that \( w_i^{t+1} - w_i^t \leq 8w_i^t \) and that \( \ln(1 + x) \geq \frac{x}{4} \) for \( x \leq 8 \).) Thus, the first and third terms in Equation 6.1 are net negative; meanwhile, the second term is also negative, so the expression is negative.

Now assume that \( w_i^t - w_i^{t+1} \geq 0 \). Again by the small gradient assumption, we have that

\[ (w_i^t - w_i^{t+1})\partial_t L^t(w^t) \leq \gamma (w_i^t - w_i^{t+1}) \leq 2(m + 1)\eta \gamma^2 (w_i^t)^{2-\alpha} \leq 3m\eta \gamma^2 \]

and

\[ 4\gamma (\ln w_i^t - \ln w_i^{t+1}) = -4\gamma \ln \left( 1 - \frac{w_i^t - w_i^{t+1}}{w_i^t} \right) \leq -4\gamma \ln(1 - 2(m + 1)\eta \gamma (w_i^t)^{1-\alpha}) \]

\[ \leq -4\gamma \ln(1 - 2(m + 1)\eta \gamma) \leq -4\gamma \ln(1 - 3m\eta \gamma) \leq 16m\eta \gamma^2 \]

for \( T \) sufficiently large, where in the last step we use that \( \ln(1 - x) \geq -\frac{4}{3}x \) for \( x > 0 \) sufficiently small (and we have \( \lim_{T \to \infty} 3m\eta \gamma = 0 \)). Since \( 16 + 3 \leq 19 \), the right-hand side of Equation 6.1 is negative. This concludes the proof.

\[ \square \]

**Corollary 6.4.7.** For sufficiently large \( T \), under the small gradient assumption, the regret of Algorithm 6.3.3 is at most \( 240 + \frac{12}{\alpha} \) \( m(3-\alpha)/2n\sqrt{T} \ln T \).

**Proof.** We have already bounded the first term in the regret bound in Lemma 6.4.2. It remains only to bound the second term. This term is exactly equal to \( \varphi(T) + 4\gamma \sum_{i=1}^m \ln w_i^{T+1} \leq \varphi(T) \), and \( \varphi(T) \leq \varphi(0) \), by Lemma 6.4.6. We have

\[ \varphi(0) = 19m^2\gamma^2 \eta T + 4m\gamma \ln m \leq 20m^2\gamma^2 \eta T \]

150
for sufficiently large $T$. Plugging in $\gamma = 12n \ln T$ and $\eta = \frac{1}{\sqrt{T \ln T}} \cdot \frac{1}{12m^{(1+\alpha)/2} n}$ concludes the proof.

### 6.4.4 The case where the small gradient assumption fails

It remains to consider the case in which the small gradient assumption does not hold. This part of the proof consists primarily of technical lemmas, which we defer to Appendix C. The key lemma is a bound on the probability that the small gradient assumption fails by a given margin:

**Lemma 6.4.8.** For any weight vector $w$, $i \in [m]$, and $\zeta \geq 0$, we have that

$$\Pr \left[ \partial_i L(w) \geq \zeta \right] \leq ne^{-\zeta}$$

(6.2)

and

$$\Pr \left[ \partial_i L(w) \leq -\frac{\zeta}{w_i} \right] \leq mn^2 e^{-\zeta/n}.$$  

(6.3)

Note that plugging in $\zeta = \gamma$ yields a bound of $mT(ne^{-\gamma} + mn^2e^{-\gamma/n})$ on the probability that the small gradient assumption fails to hold. (Since $\gamma = 12n \ln T$, this quantity is on the order of $T^{-11}$.)

The proof of Lemma 6.4.8 is the only part of the proof of Theorem 6.3.4 that uses the calibration property. While we defer the full proof to Appendix C, we highlight how the calibration property is used to prove Equation 6.2. In brief, it is straightforward to show that $\partial_i L(w) \leq -\ln p_j$, where $J$ is the random variable corresponding to the realized outcome.\footnote{Writing out the expression for $L(w)$ and differentiating leaves us with $-\ln p_j$ plus a negative term (see Equation C.5).}

Therefore, we have

$$\Pr \left[ \partial_i L(w) \geq \zeta \right] \leq \Pr \left[ -\ln p_j \geq \zeta \right] = \Pr \left[ p_j \leq e^{-\zeta} \right] = \sum_{j=1}^n \Pr \left[ J = j \ & p_j \leq e^{-\zeta} \right]$$

$$= \sum_{j=1}^n \Pr \left[ p_j \leq e^{-\zeta} \right] \Pr \left[ J = j \ | p_j \leq e^{-\zeta} \right] \leq \sum_{j=1}^n \Pr \left[ J = j \ | p_j \leq e^{-\zeta} \right] \leq ne^{-\zeta},$$

where the last step follows by the calibration property, thus proving Equation 6.2.
Combining Lemma 6.4.8 with an analysis of our algorithm using the standard regret bound for online mirror descent (Orabona, 2021, Theorem 6.8) gives us the following result as a corollary.

**Corollary 6.4.9.** The expected total regret of our algorithm conditional on the small gradient assumption not holding, times the probability of this event, is at most $\tilde{O}(T^{(5-\alpha)/(1-\alpha)-1})$.

It follows that the contribution to expected regret from the case that the small gradient assumption does not hold is $\tilde{O}(T^{-1})$, which is negligible. Together with Corollary 6.4.7 (which bounds regret under the small gradient assumption), this proves Theorem 6.3.4.

### 6.4.5 Approximate calibration

Theorem 6.3.4 holds even if experts are only *approximately* calibrated.

**Definition 6.4.10.** For $\tau \geq 1$, we say that expert $i$ is $\tau$-calibrated if for all $p \in \Delta_n$ and $j \in [n]$, we have that $\Pr[J = j \mid p^j = p] \leq \tau p_j$. We say that $\mathbb{P}$ satisfies the $\tau$-approximate calibration property if every expert is $\tau$-calibrated.

**Corollary 6.4.11.** For any $\tau$, Theorem 6.3.4 holds even if the calibration property is replaced with the $\tau$-approximate calibration property.

(Note that the $\tau$ is subsumed by the big-$O$ notation in Theorem 6.3.4; Corollary 6.4.11 does not allow experts to be arbitrarily miscalibrated.)

Technically, Corollary 6.4.11 is a corollary of the *proof* of Theorem 6.3.4, rather than a corollary of the theorem itself.\(^8\)

**Proof of Corollary 6.4.11.** We only used the calibration property in the proofs of Equations 6.2 and 6.3. In the proof of Equation 6.2, we used the fact that $\Pr[J = j \mid p^j \leq e^{-\zeta}] \leq e^{-\zeta}$; the right-hand side now becomes $\tau e^{-\zeta}$, and so the right-hand side of Equation 6.2 changes to $\tau ne^{-\zeta}$. Similarly, the right-hand side of Equation 6.3 changes\(^9\) to $\tau mn e^{-\zeta}/n$.

\(^{8}\)Fun fact: the technical term for a corollary to a proof is a *porism*.

\(^{9}\)Specifically, in the proof of Equation 6.3, we use the calibration property in the proof of Lemma C.2.4; the right-hand side of the lemma changes to $\tau mnq$. 

152
Lemma 6.4.8 is only used in the proof of Corollary 6.4.9, where $2m^2n^2T$ is replaced by $2\tau m^2n^2T$. Since $\tau$ is a constant, Corollary 6.4.9 holds verbatim.

6.5 Lower bound

In this section, we prove a lower bound result for our setting. Specifically, we show that no OMD algorithm with a constant step size\(^{10}\) substantially outperforms Algorithm 6.3.3.

**Theorem 6.5.1.** *For every strictly convex function $R : \Delta_m \to \mathbb{R}$ that is continuously twice differentiable at its minimum, and $\eta \geq 0$, online mirror descent with regularizer $R$ and constant step size $\eta$ incurs $\Omega(\sqrt{T})$ expected regret.***

*Proof.* Our examples will have $m = n = 2$. The space of weights is one-dimensional; let us call $w$ the weight of the first expert. We may treat $R$ as a (convex) function of $w$, and similarly for the losses at each time step. We assume that $R'(0.5) = 0$; this allows us to assume that $w_1 = 0.5$ and does not affect the proof idea.

It is straightforward to check that if Experts 1 and 2 assign probabilities $p$ and $\frac{1}{2}$, respectively, to the correct outcome, then

$$L'(w) = \frac{(1 - p)^w}{p^w + (1 - p)^w} \ln \frac{1 - p}{p}.$$ 

If roles are reversed (they say $\frac{1}{2}$ and $p$ respectively) then

$$L'(w) = -\frac{(1 - p)^{1-w}}{p^{1-w} + (1 - p)^{1-w}} \ln \frac{1 - p}{p}.$$ 

We first prove the regret bound if $\eta$ is small ($\eta \leq T^{-1/2}$). Consider the following setting: Expert 1 always reports (50%, 50%); Expert 2 always reports (90%, 10%); and Outcome 1 happens with probability 90% at each time step. It is a matter of simple computation that:

---

\(^{10}\)While Algorithm 6.3.3 does not always have a constant step size, it does so with high probability. The examples that prove Theorem 6.5.1 cause $\Omega(\sqrt{T})$ regret in the typical case, rather than causing unusually large regret in an atypical case. This makes our comparison of Algorithm 6.3.3 to this class fair.
• $L'(w) \leq 2$ no matter the outcome or the value of $w$.

• If $w \geq 0.4$, then $p^*_1(w) \leq 0.8$.

The first point implies that $R'(w_t) \geq -2\eta t$ for all $t$. It follows from the second point that the algorithm will output weights that will result in an aggregate probability of less than 80% for values of $t$ such that $-2\eta t \geq R'(0.4)$, i.e. for $t \leq \frac{-R'(0.4)}{2\eta}$. Each of these time steps accumulates constant regret compared to the optimal weight vector in hindsight (which with high probability will be near 1). Therefore, the expected total regret accumulated during these time steps is $\Omega(1/\eta) = \Omega(\sqrt{T})$.

Now we consider the case in which $\eta$ is large ($\eta \geq \sqrt{T}$). In this case our example is the same as before, except we change which expert is “ignorant” (reports $(50\%, 50\%)$) and which is “informed” (reports $(90\%, 10\%)$). Specifically the informed expert will be the one with a lower weight (breaking ties arbitrarily).

We will show that our algorithm incurs $\Omega(\eta)$ regret compared to always choosing weight 0.5. Suppose without loss of generality that at a given time step $t$, Expert 1 is informed (so $w^t \leq 0.5$). Observe that

$$L(w^t) - L(0.5) = -(0.5 - w^t)L'(0.5) + O((0.5 - w)^2)$$

$$= -(0.5 - w^t) \frac{\sqrt{1-p}}{\sqrt{p} + \sqrt{1-p}} \ln \frac{1-p}{p} + O((0.5 - w)^2),$$

where $p$ is the probability that Expert 1 assigns to the event that happens (so $p = 0.9$ with probability 0.9 and $p = 0.1$ with probability 0.1). This expression is (up to lower order terms) equal to $c(0.5 - w^t)$ if $p = 0.9$ and $-3c(0.5 - w^t)$ if $p = 0.1$, where $c \approx 0.55$. This means that an expected regret (relative to $w = 0.5$) of $0.6c(0.5 - w^t)$ (up to lower order terms) is incurred.

Let $D$ be such that $R''(w) \leq D$ for all $w$ such that $|w - 0.5| \leq \frac{\sqrt{T}}{4D}$. (Such a $D$ exists because $R$ is continuously twice differentiable at 0.5.) If $|w^t - 0.5| \geq \frac{\sqrt{T}}{4D}$, we just showed that an expected regret (relative to $w = 0.5$) of $\Omega\left(\frac{\sqrt{T}}{4D}\right)$ is incurred. On the other hand, suppose that $|w^t - 0.5| \leq \frac{\sqrt{T}}{4D}$. We show that $|w^{t+1} - 0.5| \geq \frac{\sqrt{T}}{4D}$.

154
To see this, note that \( |L'(w')| \geq 0.5 \), we have that \( |R'(w^{t+1}) - R'(w')| \geq 0.5\eta \). We also have that \( D |w^{t+1} - w'| \geq |R'(w^{t+1}) - R'(w')| \), so \( D |w^{t+1} - w'| \geq 0.5\eta \). Therefore, \( |w^{t+1} - w'| \geq \frac{\eta}{2D} \geq \frac{\sqrt{T}}{2D} \), which means that \( |w^{t+1} - 0.5| \geq \frac{\sqrt{T}}{4D} \).

This means that an expected regret (relative to \( w = 0.5 \)) of \( \Omega\left(\frac{\sqrt{T}}{4D}\right) \) is incurred on at least half of time steps. Since \( D \) is a constant, it follows that a total regret of at least \( \Omega(\sqrt{T}) \) is incurred, as desired.

\[\Box\]

### 6.6 Conclusion

In this work, we have considered the problem of learning optimal weights for the logarithmic pooling of expert forecasts. It quickly became apparent that under the usual fully adversarial setup, attaining vanishing regret is impossible (Example 6.1.1). We chose to relax the environment by imposing the constraint on the adversary that experts must be calibrated. Put otherwise, the adversary is allowed to choose a joint probability distribution over the experts’ reports and the outcome however it wants to, so long as the experts’ reports are calibrated, after which the realized reports and outcome are selected at random from this distribution. To our knowledge, this setting is a novel contribution to the literature on prediction with expert advice. The setting may be of independent interest: we have demonstrated that no-regret bounds are possible in this setting when they are otherwise impossible, and it seems plausible that even in settings where no-regret bounds are attainable in a fully adversarial setting, the calibration property allows for stronger results.

Another important direction for future work is learning weights for generalized logarithmic pooling: loosely speaking, logarithmic pooling but without the requirement that weights add to 1. In Chapter 2, we introduced generalized logarithmic pooling and exhibited a class of information structures for which generalized logarithmic pooling is the Bayesian optimal aggregation method. This raises the natural question of whether our methods can be adapted to this more general setting.

Finally, we are interested in learning weights for other pooling methods. In particular, it is natural to ask which proper loss functions \( \ell \) have the property that it is possible to achieve vanishing
regret when learning weights for QA pooling with respect to $\ell$, and under what assumptions. While in Chapter 5 we showed that no-regret learning is possible for bounded loss functions, extending our techniques to unbounded loss functions beyond the log loss is a promising avenue for future exploration.
Chapter 7: Robust aggregation of substitutable signals

This chapter presents “Are You Smarter Than a Random Expert? The Robust Aggregation of Substitutable Signals” (Neyman & Roughgarden, 2022a), although the contents of Section 7.3 are original to this thesis. It assumes background on forecast aggregation methods (Section 2.2) and information structures (Section 2.3). Although the relevant definitions will be restated, I strongly recommend reading those sections before reading this chapter in order to gain context and intuition.

Summary: The problem of aggregating expert forecasts is ubiquitous in fields as wide-ranging as machine learning, economics, climate science, and national security. Despite this, our theoretical understanding of this question is fairly shallow. The work discussed in this chapter initiates the study of forecast aggregation in a context where experts’ knowledge is chosen adversarially from a broad class of information structures. While in full generality it is impossible to achieve a nontrivial performance guarantee, we show that doing so is possible under a condition on the experts’ information structure that we call projective substitutes. The projective substitutes condition is a notion of informational substitutes: that there are diminishing marginal returns to learning the experts’ signals. We show that under the projective substitutes condition, taking the average of the experts’ forecasts improves substantially upon the strategy of trusting a random expert. We then consider a more permissive setting, in which the aggregator has access to the prior. We show that by averaging the experts’ forecasts and then extremizing the average by moving it away from the prior by a constant factor, the aggregator’s performance guarantee is substantially better than is possible without knowledge of the prior. Our results give a theoretical grounding to past empirical research on extremization and help give guidance on the appropriate amount to extremize.
7.1 Introduction

Suppose that you wish to estimate how much the GDP of the United States will grow next year: perhaps you are making financial decisions and want to know whether to expect a downturn. You don’t personally know much about the question – just that the historical average rate of GDP growth has been 3% – but on the internet you find several forecasts made by machine learning models. One model predicts 3.5% growth next year; another predicts 1.5%; a third predicts a downturn: -1% growth. How might you take this information into account and turn it into one number: your best guess, all things considered?¹

Because of the ubiquity of its applications, forecast aggregation is of critical importance to many fields: economics, climate science, public health, meteorology, ecology, and sociology, to name a few (McAndrew et al., 2021). Despite this, the theoretical tools we have for understanding this problem are fairly limited.

What should we ask of a framework for comparing competing aggregation methods? First, for each fixed setup, it should allow us to quantitatively assess an aggregation method based on its performance relative to a natural benchmark (analogous to, for example, assessing an online learning algorithm via its regret with respect to the best fixed action in hindsight, as we saw in Chapter 6). Second, the framework should be general: rather than evaluating an aggregation method based on its performance under a particular assumption about the experts’ information sets, it should assess the method based on its performance over a broad range of possible setups.

We can model each expert as having partial information over the state of the world, and thus the quantity being estimated (which we denote $Y$). The experts’ information sets may overlap in essentially arbitrary ways, which we formalize using information structures (introduced in Section 2.3).

No aggregation method is simultaneously optimal for every information structure. As such, it is natural to ask which aggregation method optimizes worst-case performance over a broad class of

¹Note the difference in setting from Chapters 5 and 6: we are now interested in aggregating real-valued forecasts, as opposed to probability distributions.
information structures. This is the approach we take, because it has the aforementioned advantages: it assesses aggregation methods based on their performance, but does so broadly rather than under specific assumptions.

### 7.1.1 Our results

Without any conditions on the experts’ information structure, no aggregation strategy can achieve a nontrivial performance guarantee.\(^2\) In this work, we optimize for worst-case performance over all information structures that satisfy a condition that we call *projective informational substitutes*. In Section 2.3.2, we introduced the notion of *informational substitutes*: that the value of learning an additional signal has diminishing marginal returns. The projective substitutes condition is a particular formalization of this concept that builds on the notion of “weak information substitutes” (Definition 2.3.5), as we will show that the weak substitutes condition is insufficient for our purposes.\(^3\) Intuitively, substitutable signals allow for effective aggregation because signal interactions are more predictable, so it is possible to infer more from forecasts alone without knowing the information structure.

We consider two settings: the prior-free setting and the known prior setting. In the prior-free setting, an aggregator receives only the experts’ forecasts as input; in the known prior setting, the aggregator additionally knows the prior, i.e. the overall expected value of \(Y\) (3% in our leading example). In both settings, the expert must then output an aggregate forecast.

One simple strategy is to pick an expert at random and “aggregate” by outputing that expert’s forecast. In expectation, this aggregate performs at least as well as the prior; and under the weak substitutes condition, the strategy does at least \(1/m\) times as well as someone who knew every expert’s signal and the information structure, where \(m\) is the number of experts.\(^4\) That is, choosing

\(^2\)For example, consider the “XOR information structure” in which two experts receive independent, random bits, and \(Y\) is their XOR. See Section 7.2.3 for further discussion.

\(^3\)In Section 7.2.5 we introduce a “secret sharing” information structure that shows that with no further assumptions beyond the weak substitutes condition, no aggregation strategy achieves a better performance guarantee than the strategy of choosing a random expert to trust.

\(^4\)We judge the performance of an aggregation strategy based on its improvement over the prior.
a random expert attains an approximation ratio of $1/m$. Unfortunately, we exhibit an information structure that satisfies weak substitutes but on which no aggregation strategy can outperform a $1/m$-approximation (even in the known prior setting).

However, under our slightly stronger assumption of projective substitutes, it is possible to improve upon this $1/m$ baseline. Thus, while one can ask about robust aggregation in many different settings, the projective substitutes condition appears to be a sweet spot: it allows for a broad array of possible information structures while still allowing at nontrivial performance guarantees in both the prior-free and known prior settings. These results are summarized in Figure 7.1.

<table>
<thead>
<tr>
<th>Information structures</th>
<th>Weak substitutes</th>
<th>Projective substitutes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example: XOR</td>
<td>Example: secret sharing</td>
<td>Example: PIF info. structures</td>
</tr>
<tr>
<td>Prior-free: [0, 0]</td>
<td>Prior-free: $\left[\frac{1}{m}, \frac{1}{m}\right]$</td>
<td>Prior-free: $\left[\frac{1.866}{m}, \frac{2}{m}\right]$</td>
</tr>
<tr>
<td>Known prior: [0, 0]</td>
<td>Known prior: $\left[\frac{1}{m}, \frac{1}{m}\right]$</td>
<td>Known prior: $\left[\frac{2.598}{m}, \frac{4}{m}\right]$</td>
</tr>
</tbody>
</table>

Figure 7.1: We are interested in the innermost setting, where nontrivial positive results are possible. Intervals given for each setting indicate positive and negative results, respectively (and are stated as asymptotic approximations in $m$, the number of experts, for the innermost setting).

In Section 7.3, we begin our investigation with a surprisingly strong negative result. Let $m$ be the number of experts. We show that no aggregation strategy can hope to perform even $\frac{4}{m}$ times as well as someone who knew every expert’s signal, even if the aggregation strategy knows the precise information structure. We do this by exhibiting a single information structure (one that satisfies the projective substitutes condition) in which the optimal aggregate of the experts’ forecasts performs less than $\frac{4}{m}$ times as well as the optimal aggregate given all experts’ signals. The information structure is quite natural (it is a PIF information structure – see Definition 2.3.4, also repeated later in this chapter), and so our $\frac{4}{m}$ bound serves as a basis of comparison for our other results. How close to this $\frac{4}{m}$ bound can we get with positive results?
In Section 7.4, we investigate the prior-free setting. In this setting, we show that under the projective substitutes condition, the aggregation strategy that averages all experts' forecasts improves upon the random expert strategy, attaining an approximation guarantee of roughly $1.866/m$. We also show that our bound is tight for two experts and close to tight for any number of experts.

In Section 7.5, we investigate the known prior setting. We prove that it is possible to improve upon the aforementioned guarantee of the prior-free setting by extremizing the average of the experts' beliefs, i.e. moving it away from the prior. Additionally, our results suggest a particular amount by which to extremize. Specifically, we show that by linearly extremizing – moving the average of the experts' forecasts away from the prior by a particular constant factor\(^5\) (that depends on \(m\)) – it is possible to attain an approximation ratio of roughly $2.598/m$. We show that our positive result is tight for two experts.

The aforementioned results are stated asymptotically in \(m\) for convenience; however, these asymptotics are not our focus. Instead, our goal is to understand which methods of aggregation work well in which settings. When is the tried and true method of averaging forecasts about optimal, and when is it possible to attain a substantial improvement? Table 7.1 and Figure 7.2 summarize our findings for small values of \(m\) (which are plausible for many of the applications that motivate this work). The high-level takeaways are:

- Under the projective substitutes condition, it is possible to improve substantially upon selecting a random expert simply by averaging the experts’ forecasts.
- When only the forecasts are known, no technique can substantially improve upon averaging.
- But when the prior is known, extremizing appropriately is substantially better than averaging, and in fact better than any possible aggregation strategy that does not use the prior.
- While there is potentially room for improvement on our extremization technique, one cannot hope for a very substantial improvement in the projective substitutes setting – even for an aggregator who knows the entire information structure.

\(^5\)This factor approaches $\sqrt{3}$ as \(m\) approaches infinity.
Weak subs. Proj. subs. (prior-free) Proj. subs. (known prior)

<table>
<thead>
<tr>
<th>m</th>
<th>Weak subs.</th>
<th>Proj. subs. (prior-free)</th>
<th>Proj. subs. (known prior)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Pos. &amp; neg.</td>
<td>Averaging (positive)</td>
<td>Negative</td>
</tr>
<tr>
<td>2</td>
<td>0.500</td>
<td>0.706</td>
<td>0.706</td>
</tr>
<tr>
<td>3</td>
<td>0.333</td>
<td>0.520</td>
<td>0.556</td>
</tr>
<tr>
<td>4</td>
<td>0.250</td>
<td>0.409</td>
<td>0.438</td>
</tr>
<tr>
<td>5</td>
<td>0.200</td>
<td>0.336</td>
<td>0.360</td>
</tr>
<tr>
<td>6</td>
<td>0.167</td>
<td>0.285</td>
<td>0.306</td>
</tr>
<tr>
<td>7</td>
<td>0.143</td>
<td>0.248</td>
<td>0.265</td>
</tr>
</tbody>
</table>

Table 7.1: For $m = 2 \ldots 7$: the approximation ratio guaranteed by choosing a random expert, which is the best one can do under weak substitutes, followed by some of our positive and negative results under the projective substitutes condition. Note that the negative results for the known prior setting hold even if the aggregator knows the entire information structure, not just the prior.

7.1.2 Related work

Robust aggregation In Chapter 1, we discussed the two most common approaches to theoretical questions about forecast aggregation: axiomatic approaches and Bayesian ones. The axiomatic approach seeks to define desirable properties of aggregation methods, and asks which methods satisfy these properties. By contrast, Bayesian approaches take a parametric view of aggregation: experts are modeled as Bayesians whose signals are drawn from a particular parameterized family of distributions, and an aggregation method is chosen to optimize an objective function within the model.

Our approach can be thought of as a hybrid of the axiomatic and Bayesian approaches, blending what we believe to be the most appealing parts of each. We draw from the Bayesian approach in using information structures as a formalism for the experts’ knowledge, whereas the goal of producing a single role that satisfies some global property (in our case, worst-case optimality) is reminiscent of the axiomatic approach. Our model is non-parametric: rather than assuming a parameterized family of distributions, we seek to optimize our aggregation method against a broad class of information structures.

Our work is most similar to Arieli et al. (2018), which likewise seeks to optimize an aggregation method against an adversarially selected information structure. However, the class of information structures that we consider is broader: while they consider the case of two Blackwell-ordered ex-
Figure 7.2: For \( m = 2 \ldots 20 \): positive and negative results for approximation ratios under various settings (similar to Table 7.1). Grey: weak substitutes (matching positive and negative results). Green: projective substitutes in the prior-free setting (positive). Red: projective substitutes in the prior-free setting (negative). Blue: projective substitutes in the known prior setting (positive). Yellow: projective substitutes in the known prior setting, and in fact the known information structure setting (negative).

Experts (i.e. two experts, an unknown one of whom knows strictly more than the other) and two conditionally independent experts, we consider an arbitrary number of experts from any information structure that satisfies the projective substitutes condition. Levy and Razin (2020) have a similar model, but are also quite restrictive in terms of the information structures they consider. Oliveira et al. (2021) use a similar model, but more distantly related: they consider arbitrary decision problems but restrict the aggregator to a finite number of decisions (just two decisions for many of their results) – in our setting this would mean forcing the aggregator to choose among finitely many output choices. Our notion of robustness is also similar to that of prior-independent algorithm design (see e.g. Hartline and Roughgarden (2008, §3)), though in a quite different setting.

Another important difference is that most of the previously mentioned work specifically considers the aggregation of probabilistic forecasts, whereas we are interested in aggregating expected value forecasts for arbitrary real-valued quantities.
**Extremization**  In Section 2.2, we discussed the merits of extremization: pushing the average of the experts’ forecasts away from the prior. Past empirical work has demonstrated that extremizing the average of the experts’ forecasts often improves the aggregate forecast (Baron et al., 2014; Satopää & Ungar, 2015; Satopää et al., 2014). Baron et al. (2014) explain this by noting that any individual forecaster should incorporate the fact that they may be missing useful information available to other forecasters, and that simply averaging forecasts would fail to incorporate the full wisdom of the crowd. Studying aggregation in the context of information structures as well, Satopää and Ungar (2015) note that the forecast average lacks resolution, meaning that its variance is provably too low, and finds that extremization helps to solve this issue. Satopää et al. (2016) note that the more disjoint the experts’ information, the more it makes sense to extremize. Satopää (2021) refines this approach, suggesting that the variance of forecasts be taken into consideration.

However, none of the aforementioned works aim to show that extremization produces better results across a broad class of information structures; instead, the authors consider particular structures, such as experts with correlated Gaussian signals. Satopää and Ungar (2015) note that the information structure framework is in full generality “too abstract to be applied in practice.” On the other hand, our approach of robust aggregation is able not only to provide a theoretical justification for extremization, but also to suggest a particular factor of extremization (Theorem 7.5.1), thus giving rigorous backing to what had previously been justified either by empirical heuristics or by optimization over a quite narrow class of information structures.

### 7.2 Key definitions and preliminaries

#### 7.2.1 Information structures

We introduced information structures in Section 2.3. Information structures will be essential to this chapter, so we recommend reviewing that discussion. For convenience, we restate the definition below.

**Definition 2.3.1.** An information structure $\mathcal{I} = (\Omega, \mathbb{P}, \sigma, Y)$ consists of:
• A set of states of the world $\Omega$, together with a probability distribution $\mathbb{P}$ over $\Omega$.

• A collection of $m$ signals $\sigma = (\sigma_1, \ldots, \sigma_m)$, which are random variables defined on $\Omega$. That is, for $i = 1 \ldots m$, we have a signal $\sigma_i : \Omega \rightarrow S_i$, where $S_i$ – the set of possible values that $\sigma_i$ can take on – is called the $i$-th signal set. (We will often think of each signal as belonging to a different expert.)

• A random variable $Y : \Omega \rightarrow \mathbb{R}^n$ for some $n$.

In this chapter, $Y$ will specifically be a real-valued random variable. We will also assume that $Y$ has finite variance. As usual, we will interpret each signal $\sigma_i$ as belonging to expert $i$.

We also recall the XOR information structure:

**Definition 2.3.8.** The XOR information structure consists of two signals, $\sigma_1$ and $\sigma_2$, that are independent, random bits (i.e. either 0 or 1, with equal probability). The value of $Y$ is equal to the binary XOR of $\sigma_1$ and $\sigma_2$, i.e. 0 if $\sigma_1 = \sigma_2$ and 1 if $\sigma_1 \neq \sigma_2$.

Just as in Section 2.3, given a subset $A \subseteq [m]$, we define $Y_A := \mathbb{E} [Y | \sigma_i : i \in A]$. That is, $Y_A$ is the random variable whose value is the expectation of $Y$ given the signals of the experts in $A$. If $A = \{i\}$, we write $Y_i$ in place of $Y_{\{i\}}$.

### 7.2.2 The Pythagorean theorem

Recall from Section 2.3.3 the Pythagorean theorem:

**Proposition 2.3.11.** (Pythagorean theorem for squared distance). In a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, let $A$ be a real-valued random variable, let $B = \mathbb{E} [A | \mathcal{H}]$ where $\mathcal{H}$ is a sub-sigma-algebra, and $C$ be a real-valued random variable defined on $\mathcal{H}$. Then

$$\mathbb{E} [(A - C)^2] = \mathbb{E} [(A - B)^2] + \mathbb{E} [(B - C)^2].$$

Readers unfamiliar with sigma-algebras can think of $\mathcal{H}$ as supplying partial information about the state of the world $\omega$ (much like a signal). A random variable is defined on $\mathcal{H}$ if its value depends only on the information that $\mathcal{H}$ provides (i.e. the value of the signal).
As discussed in Section 2.3.3, random variables can be thought of as vectors in a Hilbert space with inner product $\langle X, Y \rangle := \mathbb{E}[XY]$. In this space, uncorrelated random variables correspond to orthogonal vectors, and conditional expectations correspond to orthogonal projections. Proposition 2.3.11 is simply an application of the usual Pythagorean theorem in the context of this space.

### 7.2.3 Improving on the prior

In this work, we will be taking the perspective of an aggregator who receives estimates of $Y$ from each expert.\(^7\) The aggregator then produces an estimate $Z$ of $Y$ which is as accurate as possible. In particular, we care about the robust estimation of $Y$: a single estimate that is simultaneously as accurate as possible across all possible information structures (satisfying the projective substitutes condition, which we discuss below).

We assess an aggregator’s performance by the squared distance between their estimate $Z$ and the true value $Y$. That is, the aggregator wishes to minimize $\mathbb{E}[(Y - Z)^2]$. We define the function $\nu(Z)$ as follows to reflect the quality of $Z$ as an estimate of $Y$.

**Definition 7.2.1.** Given an information structure $I = (\Omega, \mathbb{P}, \sigma, Y)$ and a random variable $Z$, we define

$$
\nu(Z) := \mathbb{E}[(Y - \mathbb{E}[Y])^2] - \mathbb{E}[(Y - Z)^2].
$$

Thus, $\nu(\cdot)$ is the improvement in loss provided by $Z$ over an uninformed estimate. For example, $\nu(Y_0) = 0$ and $\nu(Y) = \mathbb{E}[(Y - \mathbb{E}[Y])^2]$ is the variance of $Y$. We cannot possibly hope for any $Z$ such that $\nu(Z) > Y_{[m]}$, since $Y_{[m]}$ is the estimate produced by knowing all information that exists. This motivates comparing $\nu(Z)$ against the benchmark $\nu(Y_{[m]})$.

However, the aggregator does not know the underlying information structure – only the experts’ estimates. Specifically, we will consider two settings:

1. The **prior-free setting**: the aggregator’s estimate is only based on the experts’ estimates. That is, $Z$ is a function of $Y_1, \ldots, Y_m$.

---

\(^7\)These estimates are each expert’s expectation of $Y$ conditioned on their signal. The aggregator does not receive signals from experts.
(2) The known prior setting: the aggregator knows the experts’ estimates and the prior. That is, $Z$ is a function of $Y_1, \ldots, Y_m$ and $\mathbb{E}[Y]$.

That is, $Z$ is a function of $m$ real numbers (or $m + 1$, in the known prior setting); we call this function the aggregator’s aggregation strategy. The aggregator’s goal is to come up with an aggregation strategy that performs well across information structures (we formalize this below).

In the known prior setting, the aggregator can report $Z = \mathbb{E}[Y]$; then $\nu(Z) = 0$ (we call this the trivial aggregation strategy). In both settings it is possible to do at least as well as the trivial aggregation strategy by reporting e.g. $Z = Y_1$. On the other hand, without any conditions on the information structure, it is not always possible to do strictly better: in the XOR information structure, the aggregator is guaranteed to receive $Y_1 = Y_2 = \frac{1}{2}$, and it is impossible for the aggregator to improve upon simply reporting the prior of $\frac{1}{2}$.

### 7.2.4 Informational complements and substitutes

Intuitively, in the XOR information structure, the aggregator is impeded by the fact that the experts’ signals are informational complements: each signal (and the estimate it produces) is not valuable by itself, but the two signals are valuable when taken together. Perhaps if we assume that the experts’ signals are instead informational substitutes, then we will be able to prove nontrivial guarantees about some aggregation strategies. And so, we recall the notion of weak informational substitutes from Section 2.3.2.

**Definition 2.3.5. (Weak substitutes w.r.t. squared error (Chen & Waggoner, 2016))** Let $\mathcal{I} = (\Omega, \mathcal{F}, \sigma, Y)$ be an information structure. $\mathcal{I}$ satisfies weak informational substitutes (or simply weak substitutes) with respect to squared error if, for all $B \subseteq A \subseteq [m]$ and $i \notin A$, we have

$$\mathbb{E}[(Y - Y_A)^2] - \mathbb{E}[(Y - Y_{A \cup \{i\}})^2] \leq \mathbb{E}[(Y - Y_B)^2] - \mathbb{E}[(Y - Y_{B \cup \{i\}})^2].$$

(2.4)

More abstractly, we could say that $\mathcal{I}$ satisfies weak substitutes with respect to squared error if $-(Y - Y_A)^2$ is a submodular set function on the subsets $A$ of $[m]$. 

167
Remark 7.2.2. Equivalently, we can say that $I$ satisfies weak informational substitutes if for all $B \subseteq A \subseteq [m]$ and $i \notin A$, we have

$$v(Y_{A \cup \{i\}}) - v(Y_A) \leq v(Y_{B \cup \{i\}}) - v(Y_B). \tag{7.1}$$

7.2.5 Random expert strategy under weak substitutes

It is not surprising that with no knowledge of the information structure, it is impossible to outperform the trivial strategy. Perhaps it would be possible to do better with only a coarse constraint on the information structure. It is not 

*a priori* obvious that this should be possible. However, if $I$ satisfies weak substitutes, then it is possible to outperform the trivial strategy by reporting a random expert’s belief:

**Proposition 7.2.3.** Suppose that $I = (\Omega, \mathbb{P}, \sigma, Y)$ satisfies weak substitutes, and let $Z$ be equal to $Y_i$ for a uniformly random $i \in [m]$ (we call this the random expert strategy). Then $v(Z) \geq \frac{1}{m} v(Y_{[m]})$.

**Proof.** For $j \in [m]$, plug $A = [j - 1], B = \emptyset, i = j$ into Equation 7.1. Adding these $m$ inequalities (and noting that $v(\emptyset) = 0$), we find that $\sum_{j=1}^{m} v(Y_j) \geq v(Y_{[m]})$. Therefore, for $Z$ as in the proposition statement, we have

$$v(Z) = \frac{1}{m} \sum_{i=1}^{m} v(Y_i) \geq \frac{1}{m} v(Y_{[m]}),$$

as desired. \qed

Put otherwise, the random expert strategy attains an approximation ratio of $1/m$.

**Definition 7.2.4.** Given an information structure $I = (\Omega, \mathbb{P}, \sigma, Y)$ with $m$ experts, the approximation ratio of a random variable $Z$ is given by the quantity $v(Z)/v(Y_{[m]})$.

The Pythagorean theorem lets us rewrite the approximation ratio in a more convenient form.
Claim 7.2.5. If a random variable $Z$ depends only on $\sigma_1, \ldots, \sigma_m$, the approximation ratio of $Z$ (i.e. $v(Z)/v(Y_m)$) is equal to
$$1 - \frac{\mathbb{E}[(Y_m - Z)^2]}{\mathbb{E}[(Y_m - \mathbb{E}[Y])^2]}.$$ 

Proof. We have
$$\frac{v(Z)}{v(Y_m)} = \frac{\mathbb{E}[(Y - \mathbb{E}[Y])^2] - \mathbb{E}[(Y - Z)^2]}{\mathbb{E}[(Y - \mathbb{E}[Y])^2] - \mathbb{E}[(Y - Y_m)^2]} = \frac{\mathbb{E}[(Y_m - \mathbb{E}[Y])^2] - \mathbb{E}[(Y_m - Z)^2]}{\mathbb{E}[(Y_m - \mathbb{E}[Y])^2]} = 1 - \frac{\mathbb{E}[(Y_m - Z)^2]}{\mathbb{E}[(Y_m - \mathbb{E}[Y])^2]}.$$

In the second step, we use the Pythagorean theorem twice: one time we plug in $A = Y, B = Y_m, C = \mathbb{E}[Y]$, and the other time, $A = Y, B = Y_m, C = Z$. In the third step, we again use the Pythagorean theorem, plugging in $A = Y, B = Y_m, C = \mathbb{E}[Y]$. 

Unfortunately, the following result shows that with no further assumptions, it is not possible to guarantee an approximation ratio larger than $1/m$:

Proposition 7.2.6. For every $m$, there is an information structure that satisfies the weak substitutes condition, such that in both the prior-free and known prior settings, no aggregation strategy attains an approximation ratio greater than $1/m$ on the information structure.

The key idea is to use Shamir secret sharing (Shamir, 1979) to create an $(m,r)$-threshold scheme for a uniformly random $r \in [m]$. Then $v(\cdot)$ is additive (and thus submodular) on the subsets of $[m]$, but an aggregator who only knows the experts’ reports will only be able to recover the secret if $r = 1$.

Proof. Let $p > m$ be a prime. Consider the following information structure (the secret sharing information structure).

- An integer $r \in [m]$ is selected uniformly at random and announced.
• A random \((r - 1)\)-th degree polynomial \(P(x) = a_0 + a_1x + \cdots + a_{r-1}x^{r-1}\) over \(\mathbb{F}_p\) is selected, with coefficients chosen uniformly at random from \(\mathbb{F}_p\), except that \(a_0\) is either \(-1\) or \(1\) (also uniformly). For each \(i \in [m]\), expert \(i\) is told \(P(i)\).

• The quantity \(Y\) is equal to \(1\) if \(a_0 = 1\) and \(-1\) if \(a_0 = -1\).

Note that for a fixed choice of \(r\) and for any \(A \subseteq [m]\), we have \(Y_A = 0\) if \(|A| < r\) and \(Y_A = \pm 1\) if \(|A| \geq r\). Therefore, for any \(A\) we have that \(Y_A = \pm 1\) with probability \(|A|/m\) and 0 otherwise. Therefore, we have that \(v(Y_A) = \frac{|A|}{m}\), so \(v(\cdot)\) is additive (and thus submodular). Thus, this information structure satisfies weak substitutes.

On the other hand, note that with probability \(\frac{m-1}{m}\), all experts report 0 to the aggregator, in which case the aggregator cannot do better in expectation than also reporting 0. Thus, it is impossible for the aggregator to report an estimate \(Z\) with \(v(Z) > \frac{1}{m}\), and so an approximation ratio larger than \(\frac{1}{m}\) is not attainable.

The secret sharing information structure is a lottery over \(m\) different information structures, for each of which \(v(\cdot)\) is a threshold function: any \(r - 1\) experts know nothing \((v(A) = 0\) if \(|A| < r\)), while any \(r\) experts know everything \((v(A) = 1\) if \(|A| \geq r\)). Except for \(r = 1\), these information structures have experts that should be intuitively regarded as complementary. Indeed, these structures generalize the XOR information structure (which is the case of \(m = r = 2\)). This suggests that properties of \(v(\cdot)\) as a set function on \([m]\) are insufficient to capture what we intuitively mean by substitutable signals. This motivates us to seek a natural but stronger notion of informational substitutes – one that is well-motivated and not too restrictive, but which rules out information structures such as this one and allows an aggregator to outperform the guarantee of the random expert strategy.

### 7.2.6 Projective substitutes

The following fact helps to motivate our stronger notion of informational substitutes.

170
Proposition 7.2.7. The weak substitutes condition may be rewritten as: for any $i$ and $B \subseteq A$, we have

$$\mathbb{E} \left[ (Y_A - Y_B)^2 \right] \geq \mathbb{E} \left[ (Y_{A\cup\{i\}} - Y_{B\cup\{i\}})^2 \right].$$

Proof. By the Pythagorean theorem, we have that

$$\mathbb{E} \left[ (Y_A - Y_B)^2 \right] = \mathbb{E} \left[ (Y - Y_A)^2 \right] - \mathbb{E} \left[ (Y - Y_B)^2 \right],$$

and similarly for the right-hand side. Rearranging terms gives us Equation 2.4.

Intuitively, this interpretation of substitutes says: **For any expert $i$, a set $B$ of experts becomes better at predicting the belief of a superset of experts $A$ if $i$’s signal is announced.** Here, by the belief of a set $T$ of experts we mean the expected value of $Y$ conditioned on all experts’ signals, i.e. $Y_T$.

This matches the intuition of substitutes as diminishing marginal returns: if signal $i$ becomes known, the “information gap” between $A$ and $B$ decreases.

A more general notion of substitutes would require this to hold even when $A$ is not a superset of $B$. That is: for all $i$, $A$, $B$, the experts in $B \cup \{i\}$ can collectively predict the belief of $A \cup \{i\}$ better than the experts in $B$ can collectively predict the belief of $A$. This captures the spirit of diminishing marginal returns in a somewhat broader context.

Let us formalize the notion of $B$’s prediction of $A$’s belief. By this we mean the expected value of $Y_A$ given the signal outcomes of the experts in $B$, i.e. $\mathbb{E} \left[ Y_A | \{\sigma_i : i \in B\} \right]$.

**Definition 7.2.8.** Given an information structure $\mathcal{I} = (\Omega, \mathbb{P}, \mathbf{\sigma}, Y)$ for $m$ experts and subsets $A, B \subseteq [m]$, $B$’s prediction of $A$’s belief is defined as the expected value of $Y_A$ given the signal outcomes of the experts in $B$, i.e.

$$Y_{A \to B} := \mathbb{E} \left[ Y_A | \{\sigma_i : i \in B\} \right].$$

We now state our substitutes definition, which strengthens the weak substitutes condition.
Definition 7.2.9. An information structure $\mathcal{I} = (\Omega, \mathbb{P}, \sigma, Y)$ for $m$ experts satisfies projective substitutes if for all $A, B \subseteq [m]$ and $i \in [m]$, we have

$$\mathbb{E} \left[ (Y_A - Y_{A \rightarrow B})^2 \right] \geq \mathbb{E} \left[ (Y_{A \cup \{i\}} - Y_{A \cup \{i\} \rightarrow B \cup \{i\}})^2 \right].$$

(7.2)

The secret sharing information structure does not satisfy projective substitutes: take $A, B, i$ with $|A| \geq |B|$ (but $A \not\supset B$) and $i \in B \setminus A$. On the other hand, the example below does satisfy projective substitutes.

Example 7.2.10. Consider the following information structure, in which the value of $Y$ is determined by the values of $\sigma_1$ and $\sigma_2$. The table on the left specifies the value of $Y$ depending on the pair of signals, and the table on the right specifies the probability of each pair of signals. (Note that the signal values $w, x, y, z$ are arbitrary labels.)

\[
\begin{align*}
Y = & \begin{cases} 
\sigma_2 = y & \sigma_2 = z \\
\sigma_1 = w & 0 \quad 1 \\
\sigma_1 = x & 1 \quad 2
\end{cases} \\
\mathbb{P} = & \begin{cases} 
\sigma_2 = y & \sigma_2 = z \\
\sigma_1 = w & 0.3 \quad 0.2 \\
\sigma_1 = x & 0.2 \quad 0.3
\end{cases}
\end{align*}
\]

Let $A = \{1\}$ and $B = \{2\}$. It is not difficult to compute that

\[
Y_A = \begin{cases} 
\sigma_2 = y & \sigma_2 = z \\
\sigma_1 = w & 0.4 \quad 0.4 \\
\sigma_1 = x & 1.6 \quad 1.6
\end{cases} \quad Y_{A \rightarrow B} = \begin{cases} 
\sigma_2 = y & \sigma_2 = z \\
\sigma_1 = w & 0.88 \quad 1.12 \\
\sigma_1 = x & 0.88 \quad 1.12
\end{cases}
\]

For example, in the $(\sigma_1, \sigma_2) = (w, y)$ case we have that $Y_A = \frac{0.3 \cdot 0.4 + 0.2 \cdot 0.1}{0.3 + 0.2} = 0.4$. $Y_A$ is then used to compute $Y_{A \rightarrow B}$: for example, in the $(\sigma_1, \sigma_2) = (w, y)$ case we have that $Y_{A \rightarrow B} = \frac{0.3 \cdot 0.4 + 0.2 \cdot 1.6}{0.3 + 0.2} = 0.88$, as this is the expected value of $Y_A$ conditioned on $\sigma_2 = y$.

It can be computed that $\mathbb{E} \left[ (Y_A - Y_{A \rightarrow B})^2 \right] = 0.3456$, whereas $\mathbb{E} \left[ (Y_{A \cup \{2\}} - Y_{A \cup \{2\} \rightarrow B \cup \{2\}})^2 \right] = \mathbb{E} \left[ (Y - Y_B)^2 \right] = 0.24$, so Equation 7.2 is satisfied for $A = \{1\}, B = \{2\}, i = 2$. It can be verified that Equation 7.2 is in fact satisfied for all $A, B, i$, so this information structure satisfies projective substitutes.

The projective substitutes definition can be interpreted as describing the class of information structures in which full information revelation is a dominant strategy. While in general we are
interested in aggregation, not elicitation, we present the following thought experiment in order to motivate the projective substitutes condition.

Consider a central party (call them the *elicitor*) who knows the information structure but does not know the experts’ signals. Experts are truthful, but may be strategic: they will not lie about their signal, but may decide not to reveal it. The elicitor wishes to structure incentives that will encourage each expert to reveal their signal. The elicitor puts experts on teams (but does not immediately announce the teams). Then:

1. Each expert either reveals their signal to the elicitor, or does not.
2. The elicitor announces which experts revealed their signals and announces the teams.
3. Each team makes a prediction about the elicitor’s posterior belief (after learning the signals of all experts who decided to reveal) and is scored using a quadratic scoring rule (i.e. penalized by the squared distance between their prediction of the elicitor’s belief and the elicitor’s actual belief).  

This mechanism incentivizes experts to reveal their signals if and only if the information structure satisfies projective substitutes. Formally:

**Proposition 7.2.11.** An information structure satisfies projective substitutes if and only if in the above mechanism, revealing one’s signal is a dominant strategy for every expert, regardless of who is on their team.

**Proof.** First suppose that the information structure satisfies projective substitutes. Consider any expert *i*, let *B* be *i*’s team, and let *A* be the set of all other experts who reveal their signals. If *i* does not reveal their signal, then the elicitor’s belief will be *Y_A* and *B*’s prediction of the elicitor’s belief will be *Y_{A→B}*. If *i* reveals their signal, then the elicitor’s belief will be *Y_{A∪{i}}* and *B*’s prediction of the elicitor’s belief will be *Y_{A∪{i}→B} = Y_{A∪{i}→B∪{i}}*. Therefore, by the projective substitutes, condition, *B*’s expected prediction error is smaller if *i* reveals their signal to the expert.

---

9 Why not elicit *Y* directly? Eliciting each team’s best guess about the *elicitor*’s belief is particularly compelling in situations in which the true value of *Y* will never be known, or will be learned in the far future. Under these circumstances, the elicitor’s belief serves as an approximation for *Y* given the available information.
Conversely, suppose that the information structure does not satisfy projective substitutes. Then there are sets $A, B \subseteq [m]$ and $i \in B$ (see Remark 7.2.12 (ii)) such that

$$\mathbb{E} \left[ (Y_A - Y_{A \rightarrow B})^2 \right] < \mathbb{E} \left[ (Y_{A \cup \{i\}} - Y_{A \cup \{i\} \rightarrow B \cup \{i\}})^2 \right].$$

Consider expert $i$, suppose their team is $B$, and suppose that the set of experts excluding $i$ who reveal their signal is $A$. Then $i$ is incentivized not to reveal their signal to the elicitor, as revealing their signal will increase $B$’s expected prediction error. \(\square\)

**Remark 7.2.12** (Facts about projective substitutes).

(i) The weak substitutes condition is equivalent to Equation 7.2 holding for all $B \subseteq A$, so the projective substitutes condition is stronger. In fact it is strictly stronger, as it excludes the secret sharing information structure.

(ii) There are several equivalent formulations of projective substitutes. One definition replaces $\{i\}$ with an arbitrary set $X$. Another modifies the definition by only requiring Equation 7.2 to hold if $i \in B$.\(^{10}\)

(iii) The notation $Y_{A \rightarrow B}$ comes from the fact that $Y_{A \rightarrow B}$ is the orthogonal projection of $Y_A$ onto the space of random variables that depend only on the signals of the experts in $B$. As a consequence of this alternative formulation, $Y_{A \rightarrow B}$ is the closest random variable (by expected squared distance) to $Y_A$ among all random variables that depend only on the values $(\sigma_i)_{i \in B}$.

### 7.2.7 PIF information structures

Recall the definition of a PIF information structure from Section 2.3:

**Definition 2.3.4.** (Satopää et al., 2016) An information structure in the partial information framework (henceforth, a PIF information structure) is an information structure that takes the following form: for each subset $S \subseteq [m]$, there is a random variable $X_S$ (all of these random variables are

\(^{10}\)To see that this is equivalent, for any $B, i$ with $i \notin B$ define $B' := B \cup \{i\}$. Then Equation 7.2 for $A, B', i$ has the same right-hand side but a smaller or equal left-hand side, and is thus more difficult to satisfy.
independent), and $\sigma_i$ is the tuple of random variables $X_S$ for all $S$ containing $i$. The value of $Y$ is equal to $\sum_S X_S$.

PIF information structures are natural because each $X_S$ can be thought of as an (additive) piece of evidence about $Y$ that is known by the experts in $S$. They are also quite versatile, as they capture the idea that different experts can have overlapping pieces of evidence in a variety of ways. As we are about to show, all PIF information structures satisfy the projective substitutes condition. This fact further motivates the notion of projective substitutes: it means that the class of projective substitutes information structures is broad enough to include all PIF information structures.

**Proposition 7.2.13.** Every PIF information structure satisfies projective substitutes.

**Proof.** Consider a PIF information structure, and without loss of generality, assume that every $X_S$ is zero-mean. Note that given $A, B \subseteq [m]$, we have

$$Y_A = \sum_{S : S \cap A \neq \emptyset} X_S.$$  

Also,

$$Y_{A \rightarrow B} = \sum_{S : S \cap A \neq \emptyset, S \cap B \neq \emptyset} X_S,$$

because $X_S$ counts toward $Y_{A \rightarrow B}$ if both someone in $B$ knows $X_S$ and someone in $A$ knows $X_S$. Therefore,

$$\mathbb{E} \left[ (Y_A - Y_{A \rightarrow B})^2 \right] = \sum_{S : S \cap A \neq \emptyset, S \cap B = \emptyset} \mathbb{E} \left[ X_S^2 \right].$$

Meanwhile,

$$\mathbb{E} \left[ (Y_{A \cup \{i\}} - Y_{A \cup \{i\} \rightarrow B \cup \{i\}})^2 \right] = \sum_{S : S \cap (A \cup \{i\}) \neq \emptyset, S \cap (B \cup \{i\}) = \emptyset} \mathbb{E} \left[ X_S^2 \right].$$

Note that if $S \cap (B \cup \{i\}) = \emptyset$, then $S \cap B = \emptyset$ and also $i \notin S$. If, furthermore, $S \cap (A \cup \{i\}) \neq \emptyset$, it follows that $S \cap A \neq \emptyset$. Thus, any such $S$ also satisfies $S \cap A \neq \emptyset, S \cap B = \emptyset$. Therefore, we have

$$\mathbb{E} \left[ (Y_A - Y_{A \rightarrow B})^2 \right] \geq \mathbb{E} \left[ (Y_{A \cup \{i\}} - Y_{A \cup \{i\} \rightarrow B \cup \{i\}})^2 \right].$$
so the information structure indeed satisfies projective substitutes.

We will use PIF information structures for two of our negative results.

### 7.3 A strong impossibility result for forecast aggregation

In this section, we exhibit a single PIF information structure on which no aggregation strategy can achieve an approximation ratio of $\frac{4}{m}$. This result thus shows that achieving an approximation ratio of $\frac{4}{m}$ is impossible even by an aggregator who knows the entire information structure. *(Thanks to Mark Xu of the Alignment Research Center for the discussion that led to this result.)*

**Theorem 7.3.1.** Consider the PIF information structure $\mathcal{I}$ defined with $X_{\{i\}} \sim \mathcal{N}(0, 1)$, $X_{[m]} \sim \mathcal{N}(0, 1)$, and all other $X_S$’s uniformly zero. No aggregation strategy achieves an approximation ratio of more than $\frac{4m}{(m+1)^2}$ on $\mathcal{I}$.

**Proof.** For convenience, we will write $X_i$ in place of $X_{\{i\}}$. We have $Y_i = X_i + X_{[m]}$ for every $i$.

Let $Z(Y_1, \ldots, Y_m)$ be the output of some aggregation strategy. Let $\hat{Y} := \mathbb{E}[Y | Y_1, \ldots, Y_m]$. By the Pythagorean theorem, we have that $\mathbb{E}[(Y - Z)^2] = \mathbb{E}[(Y - \hat{Y})^2] + \mathbb{E}[(\hat{Y} - Z)^2] \geq \mathbb{E}[(Y - \hat{Y})^2]$, with equality if $Z$ is uniformly equal to $\hat{Y}$.

We claim that $\hat{Y} = \frac{2}{m+1} \sum_i Y_i$. To see this, observe that for all $j$, $Y_j$ is uncorrelated with $Y - \frac{2}{m+1} \sum_i Y_i$:

$$
\mathbb{E} \left[ Y_j \left( Y - \frac{2}{m+1} \sum_i Y_i \right) \right] = \mathbb{E} \left[ (X_j + X_{[m]}) \left( \frac{m-1}{m+1} \sum_i X_i - \frac{m-1}{m+1} X_{[m]} \right) \right] = \frac{m-1}{m+1} \left( \mathbb{E} \left[ X_j^2 \right] - \mathbb{E} \left[ X_{[m]}^2 \right] \right) = 0.
$$

Because uncorrelated, jointly multinormal vectors are independent (Mardia et al., 1979), we in fact have that $Y - \frac{2}{m+1} \sum_i Y_i$ is independent of every $Y_j$. Therefore, the expected value of $Y - \frac{2}{m+1} \sum_i Y_i$ conditioned on $Y_1, \ldots, Y_m$ is uniformly zero, and so $\hat{Y} = \frac{2}{m+1} \sum_i Y_i$. 

176
Thus, let $Z = \frac{2}{m+1} \sum_i Y_i$. This is the optimal $Z$, and the approximation ratio achieved by $Z$ is

$$1 - \frac{\mathbb{E} [(Y - Z)^2]}{\mathbb{E} [Y^2]} = 1 - \frac{\left(\frac{m-1}{m+1}\right)^2 (m + 1)}{m + 1} = \frac{4m}{(m + 1)^2},$$

as desired. \qed

7.4 The prior-free setting

Now that we have established that no aggregator can hope to achieve an approximation ratio of $\frac{4}{m}$, even under the projective substitutes condition and even if the aggregator knows the information structure, we ask: how close to $\frac{4}{m}$ can we get? We begin our investigation with the prior-free setting: that is, we will assume that the aggregator knows nothing about the information structure, except that it satisfies the projective substitutes condition. In this setting, what is the largest approximation ratio that the aggregator can guarantee?

In this section we give a positive result and a negative result. The positive result is that averaging the experts’ reports attains an approximation ratio of at least $\left(1 + \sqrt{3}/2\right)/m - O(1/m^2) \approx \frac{1.866}{m}$. The negative result is that for all $m$, no aggregation strategy attains an approximation ratio of more than $2/m - 1/m^2$. Thus, projective substitutes enables a significant improvement over the $1/m$-approximation guarantee of the random expert strategy, but no more than by a factor of two.

7.4.1 Positive result for the prior-free setting

**Theorem 7.4.1.** Let $I = (\Omega, \mathcal{F}, S, Y)$ be an information structure for $m$ experts that satisfies projective substitutes, and let $Z = \frac{1}{m} \sum_{i=1}^m Y_i$. Then $Z$ attains an approximation ratio of at least

$$\frac{2}{m} - \frac{m - 1}{2m(2m - 1 + \sqrt{3}m^2 - 3m + 1)} - \frac{1}{m^2} \geq \left(1 + \frac{\sqrt{3}}{2}\right) \cdot \frac{1}{m} - O\left(\frac{1}{m^2}\right).$$
In other words, knowing nothing about an information structure other than the fact that it satisfies the projective substitutes condition, one can significantly improve upon the $1/m$-approximation guarantee of choosing a random expert.

To prove this result, we use the projective substitutes condition to show that one of two things must be true: either (a) the experts’ forecasts are (in expectation) fairly different from each other, or (b) the forecasts are somewhat accurate, meaning that they improve substantially upon the prior. (This is a reinterpretation of Lemma 7.4.2 below.) In case (a), averaging the experts’ forecasts guarantees substantial improvement upon a random forecast; in case (b), even though averaging the forecasts does not improve substantially upon a random forecast, a random forecast already substantially outperforms the prior. (Our proof does not rely on casework, instead showing that these two cases quantitatively trade off against each other.)

**Proof.** Let $Z = \frac{1}{m} \sum_{i=1}^{m} Y_i$. By Claim 7.2.5, showing that $Z$ achieves an approximation ratio of $\alpha$ is equivalent to showing that

$$
\mathbb{E} [(Y_{[m]} - Z)^2] \leq (1 - \alpha) \mathbb{E} [(Y_{[m]} - \mathbb{E}[Y])^2].
$$

(7.3)

The first step in our proof uses the following fact: for any numbers $y_1, \ldots, y_m$ and $y$, we have

$$
(y - \frac{y_1 + \cdots + y_m}{m})^2 = \frac{1}{m} \sum_{i=1}^{m} (y - y_i)^2 - \frac{1}{m^2} \sum_{1 \leq i < j \leq m} (y_i - y_j)^2.
$$

This equality follows from rearranging terms, and applying it in expectation for $y = Y_{[m]}$ and $y_i = Y_i$ gives us the following equality.

$$
\mathbb{E} [(Y_{[m]} - Z)^2] = \frac{1}{m} \sum_{i=1}^{m} \mathbb{E} [(Y_{[m]} - Y_i)^2] - \frac{1}{m^2} \sum_{1 \leq i < j \leq m} \mathbb{E} [(Y_i - Y_j)^2].
$$

(7.4)

The left-hand side here is the same as in Equation 7.3; meanwhile, the right-hand side has a term representing the average error of a random expert and another term representing the average expected distance between the experts’ forecasts. The following lemma allows us to get a handle on
Lemma 7.4.2. For all \(i, j\), and for all \(a, b \geq 0\) such that \(b \geq \frac{(2a-1)^2}{4a}\), we have

\[
\mathbb{E} [(Y_i - Y_j)^2] \geq a \left( \mathbb{E} [(Y_{i,j} - Y_i)^2] + \mathbb{E} [(Y_{i,j} - Y_j)^2] \right) - b \left( \mathbb{E} [(Y_i - \mathbb{E} [Y])^2] + \mathbb{E} [(Y_j - \mathbb{E} [Y])^2] \right).
\]  

(7.5)

The proof of Lemma 7.4.2 relies on the projective substitutes assumption. The lemma lets us flexibly lower bound the expected distance between \(Y_i\) and \(Y_j\) in terms of the average expected distance from \(Y_{i,j}\) to \(Y_i\) and \(Y_j\). Intuitively, the projective substitutes condition guarantees such a bound because expert \(i\) must be able to forecast \(Y_{i,j}\) better than they can forecast \(Y_j\). For now we assume the truth of Lemma 7.4.2 and return to the proof of Theorem 7.4.1. We note that we may rewrite

\[
\mathbb{E} [(Y_i - \mathbb{E} [Y])^2] = \mathbb{E} [(Y_{[m]} - \mathbb{E} [Y])^2] - \mathbb{E} [(Y_{[m]} - Y_i)^2],
\]

by the Pythagorean theorem. Additionally, we note that by weak substitutes (which follows from projective substitutes), for all \(i\) we have

\[
\sum_{j \neq i} \mathbb{E} [(Y_{i,j} - Y_i)^2] \geq \mathbb{E} [(Y_{[m]} - Y_i)^2].
\]  

(7.7)

To see this, consider for example \(i = 1\). By weak substitutes, \(\mathbb{E} [(Y_{(1,j)} - Y_1)^2] \geq \mathbb{E} [(Y_{(j)} - Y_{(j-1)})^2]\), so the left-hand side of Equation 7.7 is greater than or equal to \(\sum_{j > 1} \mathbb{E} [(Y_{(j)} - Y_{(j-1)})^2]\), which (by \(m - 2\) applications of the Pythagorean theorem) is equal to \(\mathbb{E} [(Y_{[m]} - Y_1)^2]\). Now, combining Equations 7.4, 7.5, 7.6, and 7.7 gives us that

\[
\mathbb{E} [(Y_{[m]} - Z)^2] \leq \frac{b(m-1)}{m} \mathbb{E} [(Y_{[m]} - \mathbb{E} [Y])^2] + \left( \frac{1}{m} - \frac{a}{m^2} - \frac{b(m-1)}{m^2} \right) \sum_{i=1}^{m} \mathbb{E} [(Y_{[m]} - Y_i)^2].
\]  

(7.8)
for any \(a, b\) satisfying Lemma 7.4.2. Now, note that by weak substitutes we have

\[
\sum_{i=1}^{m} \mathbb{E} \left[ (Y_{[m]} - Y_i)^2 \right] = m \mathbb{E} \left[ (Y_{[m]} - \mathbb{E} [Y])^2 \right] - \sum_{i=1}^{m} \mathbb{E} \left[ (Y_i - \mathbb{E} [Y])^2 \right] \leq (m-1) \mathbb{E} \left[ (Y_{[m]} - \mathbb{E} [Y])^2 \right],
\]

(7.9)

where the first step uses the Pythagorean theorem and the second step follows from Proposition 7.2.3. Therefore, if \(\frac{1}{m} - \frac{a}{m^2} - \frac{b(m-1)}{m^2} \geq 0\), we may write Equation 7.8 as

\[
\mathbb{E} \left[ (Y_{[m]} - Z)^2 \right] \leq \frac{m-1}{m} \left( 1 - \frac{a-b}{m} \right) \mathbb{E} \left[ (Y_{[m]} - \mathbb{E} [Y])^2 \right].
\]

To make this inequality as tight as possible, we wish to make \(a - b\) as large as possible; our constraints are that \(b \geq \frac{(2a-1)^2}{4a}\) and \(\frac{1}{m} - \frac{a}{m^2} - \frac{b(m-1)}{m^2} \geq 0\). The optimal values are

\[
a = \frac{2m - 1 + \sqrt{3m^2 - 3m + 1}}{2m} \quad \text{and} \quad b = \frac{(2a - 1)^2}{4a}.
\]

This gives us

\[
\alpha = 1 - \frac{m-1}{m} \left( 1 - \frac{a-b}{m} \right) = \frac{2}{m} - \frac{m-1}{2m(2m - 1 + \sqrt{3m^2 - 3m + 1})} - \frac{1}{m^2},
\]

as desired.  

\(\square\)

**Proof of Lemma 7.4.2.** Thinking of random variables as vectors (as in Section 7.2.2), let \(T_i\) be the projection of \(Y_j\) onto the space of all affine combinations of \(Y_i\) and \(Y_\emptyset\), i.e. \(\{\beta Y_i + (1-\beta)Y_\emptyset : \beta \in \mathbb{R}\}\). (Recall that \(Y_\emptyset\) is the random variable that is always equal to \(\mathbb{E} [Y]\).) Define \(T_j\) analogously. Note that \(\mathbb{E} \left[ (Y_j - Y_{j \rightarrow i})^2 \right] \leq \mathbb{E} \left[ (Y_j - T_j)^2 \right]\), since \(Y_{j \rightarrow i}\) is the closest point to \(Y_j\) of the subspace of random variables that depend only on \(\sigma_i\), and the aforementioned affine space is a subset of that subspace. Additionally, by projective substitutes (with \(A = \{i\}, B = \{j\}, i = i\) in Definition 7.2.9), we have that \(\mathbb{E} \left[ (Y_{[i,j]} - Y_i)^2 \right] \leq \mathbb{E} \left[ (Y_j - Y_{j \rightarrow i})^2 \right]\). Therefore, we have that \(\mathbb{E} \left[ (Y_{[i,j]} - Y_i)^2 \right] \leq \mathbb{E} \left[ (Y_j - T_j)^2 \right]\), and similarly that \(\mathbb{E} \left[ (Y_{[i,j]} - Y_j)^2 \right] \leq \mathbb{E} \left[ (Y_i - T_i)^2 \right]\). It therefore suffices to show

180
that
\[
\mathbb{E}
\left[
(Y_i - Y_j)^2
\right]
\geq
a
\left(
\mathbb{E}
\left[
(Y_i - T_i)^2
\right] + \mathbb{E}
\left[
(Y_j - T_j)^2
\right]
\right) - b
\left(
\mathbb{E}
\left[
(Y_i - Y_0)^2
\right] + \mathbb{E}
\left[
(Y_j - Y_0)^2
\right]
\right).
\]
(7.10)

By the Pythagorean theorem,\(^{11}\) we know the following four facts.

\[
\mathbb{E}
\left[
(Y_i - Y_j)^2
\right] = \mathbb{E}
\left[
(Y_i - T_i)^2
\right] + \mathbb{E}
\left[
(Y_j - T_j)^2
\right] ;
\mathbb{E}
\left[
(Y_i - Y_j)^2
\right] = \mathbb{E}
\left[
(Y_j - T_j)^2
\right] + \mathbb{E}
\left[
(Y_i - T_i)^2
\right]
\]
\[
\mathbb{E}
\left[
(Y_i - Y_0)^2
\right] = \mathbb{E}
\left[
(Y_i - T_i)^2
\right] + \mathbb{E}
\left[
(T_i - Y_0)^2
\right] ;
\mathbb{E}
\left[
(Y_j - Y_0)^2
\right] = \mathbb{E}
\left[
(Y_j - T_j)^2
\right] + \mathbb{E}
\left[
(T_j - Y_0)^2
\right]
\]

These let us rewrite Equation 7.10 as such:

\[
\frac{1}{2}
\left(
\mathbb{E}
\left[
(Y_j - T_i)^2
\right] + \mathbb{E}
\left[
(Y_i - T_j)^2
\right]
\right) + \left(a - \frac{1}{2}\right)
\left(
\mathbb{E}
\left[
(Y_i - Y_0)^2
\right] + \mathbb{E}
\left[
(Y_j - Y_0)^2
\right]
\right)
\geq
\left(a - b - \frac{1}{2}\right)
\left(
\mathbb{E}
\left[
(Y_i - Y_j)^2
\right] + \mathbb{E}
\left[
(Y_j - Y_0)^2
\right]
\right).
\]
(7.11)

We wish to show that this inequality holds so long as \(b \geq \frac{(2a-1)^2}{4a}\). To do so, we note the following fact: for any random variables \(Q, R\) and non-negative reals \(c_1, c_2\), we have that

\[
c_1\mathbb{E}
\left[
Q^2
\right] + c_2\mathbb{E}
\left[
R^2
\right] \geq \frac{c_1c_2}{c_1 + c_2}\mathbb{E}
\left[
(Q + R)^2
\right].
\]

This follows (after multiplying through by \(c_1 + c_2\) and cancelling terms) from the fact that for all \(q, r\) we have \((c_1q)^2 + (c_2r)^2 \geq 2(c_1q)(c_2r)\). Now, we apply this identity to \(Q := Y_j - T_i\) and \(R := T_i - Y_0\), with \(c_1 = \frac{1}{2}\) and \(c_2 = a - \frac{1}{2}\), and also to \(Q := Y_i - T_j\) and \(R := T_j - Y_0\). This tells us that

\[
\frac{1}{2}
\left(
\mathbb{E}
\left[
(Y_j - T_i)^2
\right] + \mathbb{E}
\left[
(Y_i - T_j)^2
\right]
\right) + \left(a - \frac{1}{2}\right)
\left(
\mathbb{E}
\left[
(T_i - Y_0)^2
\right] + \mathbb{E}
\left[
(T_j - Y_0)^2
\right]
\right)
\geq \frac{2a - 1}{4a}
\left(
\mathbb{E}
\left[
(Y_i - Y_0)^2
\right] + \mathbb{E}
\left[
(Y_j - Y_0)^2
\right]
\right).
\]

\(^{11}\)While in most cases by “Pythagorean theorem” we mean Proposition 2.3.11, in this case we are referring to the fact that for orthogonal vectors \(x\) and \(y\), we have \(\|x\|^2 + \|y\|^2 = \|x + y\|^2\) (and applying this fact to e.g. \(x = Y_i - T_i\), \(y = Y_j - T_i\)).
Therefore, Equation 7.11 holds so long as \( \frac{2a-1}{4a} \geq a - b - \frac{1}{2} \), which is equivalent to \( b \geq \frac{(2a-1)^2}{4a} \). \(\square\)

### 7.4.2 Negative results for the prior-free setting

**Theorem 7.4.3.** Fix any \( m \geq 1 \). For \( \mu \in \mathbb{R} \), let \( I_\mu \) be the PIF information structure defined with \( X_{(i)} \sim \mathcal{N}(0, 1) \), \( X_{[m]} = m\mu \) deterministically, and all other \( X_i \)'s uniformly zero. No aggregation strategy achieves an approximation ratio of more than \( \frac{2}{m} - \frac{1}{m^2} \) on every \( I_\mu \).

Note that by Proposition 7.2.13, \( I_\mu \) satisfies the projective substitutes condition, and so Theorem 7.4.3 serves as a negative result for the prior-free setting. The result would still apply if we replaced the projective substitutes assumption with any assumption that permits the class of all PIF information structures (or indeed, any assumption that permits the set of all information structures \( I_\mu \)).

**Proof.** For simplicity, we write \( X_i \) in place of \( X_{(i)} \). We prove the theorem in two steps. First, we show that taking the average of the experts’ reports yields an approximation ratio of exactly \( \frac{2}{m} - \frac{1}{m^2} \) for all \( \mu \). Second, we show that no aggregation strategy beats taking the average for every \( \mu \).

For the first step, assume without loss of generality that \( \mu = 0 \). Then \( Y_{[m]} = \sum_i X_i \), and the average of the \( Y_i \)'s (which we will denote \( Z \)) is equal to \( \frac{1}{m} \sum_i X_i \). Therefore we have

\[
\mathbb{E} \left[ (Y_{[m]} - Z)^2 \right] = \mathbb{E} \left[ \left( \frac{m - 1}{m} \sum_i X_i \right)^2 \right] = \left( \frac{m - 1}{m} \right)^2 \mathbb{E} \left[ \left( \sum_i X_i \right)^2 \right].
\]

On the other hand, we have that \( \mathbb{E} \left[ (Y_{[m]} - \mathbb{E}[Y])^2 \right] = \mathbb{E} \left[ (\sum_i X_i)^2 \right] \), so (using Claim 7.2.5) we have that the approximation ratio is

\[
1 - \left( \frac{m - 1}{m} \right)^2 = \frac{2}{m} - \frac{1}{m^2}.
\]

This completes the first step. To complete the second step, we use the following well-known result from statistical theory.\(^{12}\)

\(^{12}\)This fact does not generalize to more than two dimensions, meaning that if the \( x_i \) are vectors in three or more
Proposition 7.4.4 (Blyth, 1951; Girshick and Savage, 1951; Hodges and Lehmann, 1951). Let \( x_1, \ldots, x_m \) be drawn independently from a normal distribution with unknown mean \( \mu \) and standard deviation 1. Let \( \hat{\mu} := \frac{1}{m} \sum_{i=1}^{m} x_i \). Then for every function \( Z \) of \( x_1, \ldots, x_m \), there exists \( \mu \) such that
\[
\mathbb{E} \left[ (Z - \mu)^2 \right] \geq \mathbb{E} \left[ (\hat{\mu} - \mu)^2 \right].
\]

Since \( \mathbb{E} \left[ (\hat{\mu} - \mu)^2 \right] = \frac{1}{m} \) for every \( \mu \), we have the following fact as a corollary.

Corollary 7.4.5. Let \( x_1, \ldots, x_m \) be drawn independently from a normal distribution with unknown mean \( \mu \) and standard deviation 1. Then for every aggregation strategy \( Z \) that takes as input \( x_1, \ldots, x_m \), we have
\[
\max_{\mu} \mathbb{E} \left[ (Z - \mu)^2 \right] \geq \frac{1}{m}.
\]

(The only subtlety is that aggregation strategies are not required to be deterministic; however, replacing a randomized aggregation strategy with the deterministic strategy that outputs the expected value of the randomized strategy given the inputs can only reduce expected squared error.)

Returning to our proof, observe that each \( Y_i \) is equal to \( m\mu + X_i \), which is an independent draw from the normally distribution with mean \( m\mu \) and standard deviation 1. Let \( Z \) be any aggregation strategy on inputs \( Y_1, \ldots, Y_m \). We define a new aggregation strategy: \( \tilde{Z} := \frac{1}{m-1} (\sum_i Y_i - Z) \). We claim that if \( Z \) achieves an approximation ratio of more than \( \frac{2}{m} - \frac{1}{m^2} \) for every \( \mu \), then \( \tilde{Z} \) violates Corollary 7.4.5. Consider \( \tilde{Z} \) as an estimator for \( m\mu \). We have
\[
\mathbb{E} \left[ (m\mu - \tilde{Z})^2 \right] = \mathbb{E} \left[ \left( m\mu - \frac{1}{m-1} \left( \sum_i Y_i - Z \right) \right)^2 \right]
\]
\[
= \frac{1}{(m-1)^2} \mathbb{E} \left[ \left( \sum_i Y_i - m(m-1)\mu - Z \right)^2 \right] = \frac{1}{(m-1)^2} \mathbb{E} \left[ (Y - Z)^2 \right],
\]
where in the last step we use the fact that \( Y = m\mu + \sum_i X_i = \sum_i Y_i - m(m-1)\mu \). Now, suppose for contradiction that \( Z \) achieves an approximation ratio of more than \( \frac{2}{m} - \frac{1}{m^2} \) on every \( \mu \). Then for all dimensions drawn independently from a normal distribution with unknown mean and known covariance matrix, then there is an estimator for the mean that Pareto dominates the sample mean according to expected squared vector distance. One such estimator is the James-Stein estimator (Stein, 1956).
we have
\[
\left( \frac{m-1}{m} \right)^2 = 1 - \left( \frac{2}{m} - \frac{1}{m^2} \right) > \frac{\mathbb{E} \left[ (Y - Z)^2 \right]}{\mathbb{E} \left[ (Y - \mathbb{E}[Y])^2 \right]} = \frac{\mathbb{E} \left[ (Y - Z)^2 \right]}{m} = \frac{(m-1)^2}{m} \mathbb{E} \left[ (m\mu - \tilde{Z})^2 \right],
\]
so \( \mathbb{E} \left[ (m\mu - \tilde{Z})^2 \right] < \frac{1}{m} \) for every value of \( m\mu \) (and thus for every \( \mu \)). This contradicts Corollary 7.4.5 and completes the proof.

Theorems 7.4.1 and 7.4.3 give us non-matching lower and upper bounds on the optimal approximation ratio under the projective substitutes condition. In particular, for \( m = 2 \) experts, Theorem 7.4.1 tells us that averaging achieves an approximation ratio of \( \frac{3+\sqrt{7}}{8} \approx 0.706 \), while Theorem 7.4.1 tells us that no aggregation strategy can achieve an approximation ratio larger than 0.75. We now show that for two experts, our positive result is tight.

**Theorem 7.4.6.** In the prior-free setting, no aggregation strategy achieves an approximation ratio larger than \( \frac{3+\sqrt{7}}{8} \) on every two-expert information structure that satisfies projective substitutes.

**Proof.** Let \( I_+ \) be the following information structure, where \( p = 1 - \frac{\sqrt{7}}{4} \) and \( x = \sqrt{14} - 2\sqrt{2} \). We label the signals \(-1\) and \(1\) because these are the expected values conditional on the respective signals.

\[
I_+ := \left\{ \begin{array}{c|c|c}
\sigma_1 = 1 & \sigma_2 = 1 & \sigma_2 = -1 \\
\sigma_1 = -1 & \frac{1-(1-2p)x}{2p} & x \\
& \frac{-1-(1-2p)x}{2p} & \end{array} \right\}
\]

Let \( I_- \) be the same information structure, but with \( x = 2\sqrt{2} - \sqrt{14} \). It is a matter of calculation to verify that these information structures satisfy projective substitutes.\(^{13}\)

Note that any aggregation strategy that outputs a number other than 1 on input \((Y_1, Y_2) = (1, 1)\) has an approximation ratio of negative infinity on an information structure where \( Y = 1 \) deterministically. This is likewise true for \(-1\) in place of 1. Thus, if Theorem 7.4.6 were false, it

\(^{13}\)These information structures were found by finding values that would make the inequalities in the proofs of Theorem 7.4.1 and Lemma 7.4.2 hold with equality.
would be disproved by an information structure that outputs 1 on $I_+$ and $I_-$ if $(\sigma_1, \sigma_2) = (1, 1)$ and $-1$ if $(\sigma_1, \sigma_2) = (-1, -1)$. Conditional on this, the aggregation strategy that minimizes the maximum expected squared distance to $Y_{[m]}$ on $I_+$ and $I_-$ is the one that returns 0 when $(\sigma_1, \sigma_2) = (1, -1)$ or $(\sigma_1, \sigma_2) = (-1, 1)$. It is a matter of calculation to verify that this aggregation strategy achieves an approximation ratio of exactly $\frac{3 + \sqrt{7}}{8}$.

7.5 The known prior setting

Let us now expand the information available to the aggregator by allowing them knowledge of the prior $\mathbb{E}[Y]$. How might this change the optimal aggregation strategy?

In Section 2.2, we saw that it often makes sense to extremize an aggregate forecast – that is, to push it away from the prior. We gave the following information structure as a motivating example: a coin comes up heads $Y$ fraction of the time, where $Y$ is selected uniformly from $[0, 1]$; each of two experts sees an independent flip of the coin. It can be calculated that an expert who sees heads has a posterior of $\frac{2}{3}$. However, consider the situation in which both experts report heads: collectively they have seen two heads and zero tails, conditional on which the expected value of $Y$ is $\frac{3}{4}$, rather than $\frac{2}{3}$.

More generally, extremization is useful when experts have private information. For example, if many experts update upward from the prior as a result of each of their pieces of evidence, then it stands to reason that observing all of the evidence would cause an update that is larger than the average of the experts’ individual updates.

7.5.1 The extremization factor

Consider the following aggregation strategy, parameterized by a constant $d$ which we will call the extremization factor.

$$Z := \frac{1}{m} \sum_i Y_i + (d - 1) \left( \frac{1}{m} \sum_i Y_i - \mathbb{E}[Y] \right).$$

(7.12)
Setting \( d = 1 \) recovers the average of the reports; setting \( d = 0 \) simply returns the prior. In general, setting \( d > 1 \) extremizes the average (i.e. pushes it away from the prior) by a factor of \( d \). As an example, consider the class of information structures in Theorem 7.4.3, where averaging achieved an approximation ratio of \( \frac{2}{m} - \frac{1}{m^2} \). On the other hand, extremizing by a factor of \( m \) (i.e. setting \( d = m \) above) recovers \( Y \) exactly (thus achieving an approximation ratio of 1). This approach, which is a special case of generalized linear pooling (introduced in Section 2.2.3), is known as linear extremization (Satopää & Ungar, 2015).

We now prove that by extremizing, we can achieve an approximation ratio that is higher than what we could hope to attain without knowledge of the prior. In particular, we find that by applying an appropriate amount of linear extremization, it is possible to achieve an approximation ratio of at least \( \frac{3\sqrt{3}}{2m} - O(1/m^2) \approx 2.598/m \). This is a substantial improvement not only over our positive result in the prior-free setting, but also over our negative result in that setting.

### 7.5.2 Positive result for the known prior setting

**Theorem 7.5.1.** Let \( \mathcal{I} = (\Omega, \mathbb{P}, S, Y) \) be an information structure for \( m \) experts that satisfies projective substitutes, and let \( Z = \frac{1}{m} \sum_{i=1}^{m} Y_i + (d - 1) \left( \frac{1}{m} \sum_{i=1}^{m} Y_i - \mathbb{E}[Y] \right) \), where \( d = \frac{m(\sqrt{3m^2 - 3m + 1} - 2)}{m^2 - m - 1} \).

Then \( Z \) attains an approximation ratio of at least

\[
\frac{(3m^2 - 3m + 1)^{3/2} - 9m^2 + 9m + 1}{2(m^2 - m - 1)^2} \geq \frac{3\sqrt{3}}{2m} - O\left(\frac{1}{m^2}\right).
\]

In Figure 7.3, we plot the values of \( d \) suggested by Theorem 7.5.1. While \( d \) increases with \( m \), it reaches a limit – namely, \( \sqrt{3} \approx 1.732 \). By contrast, the optimal response to the information structures in our negative result for the prior-free setting (Theorem 7.4.3) was to extremize by a factor of \( m \) (i.e. to add up the experts’ updates from the prior). This is a consequence of the fact that the signals received by each expert were independent. By contrast, Theorem 7.5.1 suggests a smaller amount of extremization, because it is concerned with the optimal strategy in the worst case over information structures. This means that it must compromise between doing well in
settings with independent signals (where a large extremization factor makes sense) and settings in which experts’ signals are highly dependent (where little or no extremization is optimal). The extremization factor suggested by Theorem 7.5.1 is also consistent with the Bayesian modeling results in Satopää (2022) and the empirical findings in Satopää and Ungar (2015).

Figure 7.3: For $m = 2 \ldots 20$, the value $d = \frac{m(\sqrt{3}m^2 - 3m + 1) - 2}{m^2 - m - 1}$ suggested by Theorem 7.5.1, together with the asymptotic (with $m$) value $\sqrt{3}$.

The proof of Theorem 7.5.1 is similar to the proof of Theorem 7.4.1, though with an additional degree of freedom ($d$) to optimize over.

**Proof of Theorem 7.5.1.** Let $d > 0$ and let $Z := \frac{1}{m} \sum_i Y_i + (d - 1) \left( \frac{1}{m} \sum_i Y_i - \mathbb{E}[Y] \right)$. As with the proof of Theorem 7.4.1, we start by upper bounding $\mathbb{E} \left[ (Y[m] - Z)^2 \right]$. We have

$$
\mathbb{E} \left[ (Y[m] - Z)^2 \right] = \mathbb{E} \left[ d \left( Y[m] - \frac{1}{m} \sum_i Y_i \right) - (d - 1) \left( Y[m] - \mathbb{E}[Y] \right) \right]^2 \\
= d^2 \mathbb{E} \left[ Y[m] - \frac{1}{m} \sum_i Y_i \right]^2 + (d - 1)^2 \mathbb{E} \left[ (Y[m] - \mathbb{E}[Y])^2 \right] \\
- \frac{2d(d - 1)}{m} \mathbb{E} \left[ mY[m] - \sum_i Y_i \right] \left( Y[m] - \mathbb{E}[Y] \right) \\
= d^2 \left( \frac{1}{m} \sum_i \mathbb{E} \left[ (Y[m] - Y_i)^2 \right] - \frac{1}{m^2} \sum_{1 \leq i < j \leq m} \mathbb{E} \left[ (Y_i - Y_j)^2 \right] \right) \\
+ (d - 1)^2 \mathbb{E} \left[ (Y[m] - \mathbb{E}[Y])^2 \right] - \frac{2d(d - 1)}{m} \sum_i \mathbb{E} \left[ (Y[m] - Y_i)^2 \right].
$$

187
In the last step, we adapt the first term using Equation 7.4 and adapt the last term by observing that

\[
\mathbb{E} \left[ (Y_{[m]} - Y_i)(Y_{[m]} - \mathbb{E} [Y_i]) \right] = \mathbb{E} \left[ Y_{[m]}(Y_{[m]} - Y_i) \right] - \mathbb{E} [Y_i] \mathbb{E} [Y_{[m]} - Y_i]
\]

\[
= \mathbb{E} \left[ Y_{[m]}(Y_{[m]} - Y_i) \right] = \mathbb{E} \left[ (Y_{[m]} - Y_i)^2 \right]
\]

(where the last step holds because for any given \( Y_i \), \( \mathbb{E} [Y_{[m]} | Y_i] = Y_i \), so \( \mathbb{E} [Y_i(Y_{[m]} - Y_i)] = 0 \)).

Grouping like terms, we have

\[
\mathbb{E} \left[ (Y_{[m]} - Z)^2 \right] = (d-1)^2 \mathbb{E} \left[ (Y_{[m]} - \mathbb{E} [Y])^2 \right] - \frac{d(d-2)}{m} \sum_i \mathbb{E} \left[ (Y_{[m]} - Y_i)^2 \right] - \frac{d^2}{m^2} \sum_{1 \leq i < j \leq m} \mathbb{E} \left[ (Y_i - Y_j)^2 \right].
\]

Now, recall Lemma 7.4.2. Consider any \( a, b \geq 0 \) satisfying \( b \geq \frac{(2a-1)^2}{4a} \); then for all \( i, j \) we have

\[
\mathbb{E} \left[ (Y_i - Y_j)^2 \right] \geq a \left( \mathbb{E} \left[ (Y_{(i,j)} - Y_i)^2 \right] + \mathbb{E} \left[ (Y_{(i,j)} - Y_j)^2 \right] \right) - b \left( \mathbb{E} \left[ (Y_i - \mathbb{E} [Y])^2 \right] + \mathbb{E} \left[ (Y_j - \mathbb{E} [Y])^2 \right] \right).
\]

Therefore we have

\[
\mathbb{E} \left[ (Y_{[m]} - Z)^2 \right] \leq (d-1)^2 \mathbb{E} \left[ (Y_{[m]} - \mathbb{E} [Y])^2 \right] - \frac{d(d-2)}{m} \sum_i \mathbb{E} \left[ (Y_{[m]} - Y_i)^2 \right]
\]

\[
- \frac{d^2}{m^2} \sum_{1 \leq i < j \leq m} \left( a \left( \mathbb{E} \left[ (Y_{(i,j)} - Y_i)^2 \right] + \mathbb{E} \left[ (Y_{(i,j)} - Y_j)^2 \right] \right) - b \left( \mathbb{E} \left[ (Y_i - \mathbb{E} [Y])^2 \right] + \mathbb{E} \left[ (Y_j - \mathbb{E} [Y])^2 \right] \right) \right)
\]

\[
= (d-1)^2 + \frac{bd^2(m-1)}{m} \mathbb{E} \left[ (Y_{[m]} - \mathbb{E} [Y])^2 \right] - \left( \frac{d(d-2)}{m} + \frac{bd^2(m-1)}{m^2} \right) \sum_i \mathbb{E} \left[ (Y_{[m]} - Y_i)^2 \right]
\]

\[
- \frac{ad^2}{m^2} \sum_{1 \leq i < j \leq m} \left( \mathbb{E} \left[ (Y_{(i,j)} - Y_i)^2 \right] + \mathbb{E} \left[ (Y_{(i,j)} - Y_j)^2 \right] \right),
\]

where in the last step we use the Pythagorean theorem to write \( \mathbb{E} \left[ (Y_i - \mathbb{E} [Y])^2 \right] \) as \( \mathbb{E} \left[ (Y - \mathbb{E} [Y])^2 \right] - \mathbb{E} \left[ (Y - Y_i)^2 \right] \). Now we use Equation 7.7:

\[
\mathbb{E} \left[ (Y_{[m]} - Z)^2 \right] \leq (d-1)^2 + \frac{bd^2(m-1)}{m} \mathbb{E} \left[ (Y_{[m]} - \mathbb{E} [Y])^2 \right]
\]

188
\[-\left(\frac{d(d-2)}{m} + \frac{bd^2(m-1)}{m^2} + \frac{ad^2}{m^2}\right) \sum_i \mathbb{E}\left[(Y_{[m]}-Y_i)^2\right].\]

Now, supposing that \(\frac{d(d-2)}{m} + \frac{bd^2(m-1)}{m^2} + \frac{ad^2}{m^2}\) is not positive, we may use Equation 7.9 to obtain:

\[
\mathbb{E}\left[(Y_{[m]}-\mathbb{E}[Y])^2\right] \leq \left((d-1)^2 + \frac{bd^2(m-1)}{m} - \frac{m-1}{m}\left(\frac{d(d-2)}{m} + \frac{bd^2(m-1)}{m} + \frac{ad^2}{m}\right)\right) \mathbb{E}\left[(Y_{[m]}-\mathbb{E}[Y])^2\right]
\]

\[= \left(1 + \frac{d^2 - 2d}{m} - (a-b)\frac{m-1}{m^2}d^2\right) \mathbb{E}\left[(Y_{[m]}-\mathbb{E}[Y])^2\right].\]

With \(d\) held fixed, our goal is to maximize \(a - b\), just as in the proof of Theorem 7.4.1. This time, our constraints are \(b \geq \frac{(2a-1)^2}{4a}\) (as before) and \(\frac{d(d-2)}{m} + \frac{bd^2(m-1)}{m^2} + \frac{ad^2}{m^2} \leq 0\), which can be rewritten as \(a + b(m-1) \leq \frac{2-2d}{d}m\). The optimal values are

\[a = \frac{-2m - 1 + \sqrt{(\frac{-2m - 1}{d})^2 - m(m-1) + \frac{2d^2 - 2d}{m^2} - (a-b)m-1}}{2m}\]

\[b = \frac{(2a-1)^2}{4a}\]

Now, let \(a\) and \(b\) be as above. We may select \(d\) as we please and seek to minimize the expression

\[1 + \frac{d^2 - 2d}{m} - (a-b)\frac{m-1}{m^2}d^2.\]

We choose the value of \(d\) in the theorem statement (which one can verify is optimal using a computer algebra system). This yields the desired approximation ratio. \(\square\)

### 7.5.3 Negative result for the known prior setting for \(m = 2\)

We already have a negative result for the known prior setting: namely, Theorem 7.3.1. In the special case of \(m = 2\), Theorem 7.3.1 shows that an approximation ratio larger than \(\frac{8}{5} \approx 0.889\) cannot be achieved. By contrast, Theorem 7.5.1 tells us that averaging and extremizing by a factor of \(2(\sqrt{7} - 2) \approx 1.292\) achieves an approximation ratio of \(\frac{7\sqrt{7} - 17}{2} \approx 0.760\). We prove that this positive result is in fact tight.
Theorem 7.5.2. In the known prior setting, no aggregation strategy achieves an approximation ratio larger than \( \frac{7\sqrt{7}-17}{2} \) on every two-expert information structure that satisfies projective substitutes.

Proof. Let \( I_+ \) be the following information structure, where \( p = \frac{2+\sqrt{7}}{12} \) and \( x = \frac{\sqrt{2+\sqrt{7}}}{3} \). We label the signals \(-1\) and \(1\) because these are the expected values conditional on the respective signals.

\[
I_+ := \left\{ Y = \begin{array}{c|c|c}
\sigma_1 = 1 & \sigma_2 = 1 & 1 - (1 - 2p)x \\
\sigma_1 = -1 & \sigma_2 = -1 & 1 - (1 - 2p)x \\
\end{array} \right. \quad I = \left\{ \begin{array}{c|c|c}
\sigma_1 = 1 & \sigma_2 = 1 & \frac{1}{2} - p \\
\sigma_1 = -1 & \sigma_2 = -1 & \frac{1}{2} - p \\
\end{array} \right.
\]

Let \( I_- \) be the same information structure, but with \( x = -\frac{\sqrt{2+\sqrt{7}}}{3} \). It is a matter of calculation to verify that these information structures satisfy projective substitutes. The quantity \( \mathbb{E} \left[ (Y - \mathbb{E}[Y])^2 \right] \) is the same for \( I_+ \) and \( I_- \), so the aggregation strategy \( Z \) that guarantees the largest possible approximation ratio when the information structure is one of \( I_+ \) and \( I_- \) is the one that minimizes the maximum value of \( \mathbb{E} \left[ (Y - Z)^2 \right] \) over these two information structures. This is achieved by outputting 0 when \((Y_1, Y_2)\) is \((1, -1)\) or \((-1, 1)\), \(\frac{1}{2p}\) when \((Y_1, Y_2) = (1, 1)\), and \(\frac{-1}{2p}\) when \((Y_1, Y_2) = (-1, -1)\). It is a matter of calculation to verify that this aggregation strategy achieves an approximation ratio of exactly \( \frac{7\sqrt{7}-17}{2} \).

\[ \square \]

7.6 Future directions in robust aggregation

Robust forecast aggregation is a new area. To our knowledge, this work is the first to demonstrate nontrivial guarantees about forecast aggregation under a truly broad class of information structures. But the space of questions to be asked about robust forecast aggregation is much broader than the space of questions that we have considered.

To highlight one example: we found that averaging followed by linear extremization is a robust aggregation method, if the aggregator's error is their squared distance to the truth. However, this aggregation method does not in general make sense for probabilistic forecasts: in particular,
aggregating probabilities in this way may result in aggregates outside of $[0, 1]$. A natural question to ask, then, is: what is a robust and sensible way to aggregate probabilistic forecasts?

If we change our error measure from squared distance to KL divergence, then answers outside of $[0, 1]$ become unacceptable, as they accrue infinite error for the aggregator. And so we can ask: what aggregation methods achieve a high worst-case approximation ratio, if we use KL divergence as our error measure? A natural first guess might be generalized logarithmic pooling, as introduced in Section 2.2.3. Is this aggregation method in fact robust?

To step back from this particular suggestion for future work, let us consider the question that we answered in this chapter. The question was: what approximation ratio can be achieved by an aggregator who learns expected value estimates of a real-valued quantity $Y$ from $m$ truthful experts whose signals are drawn from an information structure that satisfies projective substitutes, if the aggregator’s loss is their squared error and the aggregator knows nothing about the information structure or only knows the prior? \(^{14}\)

All of the emphasized phrases in this question can be varied! For example:

- What if the aggregator is judged based on their KL divergence from the true value of $Y$, or a different Bregman divergence? What if the aggregator is judged based on an altogether different distance function (perhaps absolute error), such that the aggregator no longer wants to guess about the expectation of $Y$, but a different property (perhaps as the median)?
- What if we make a different set of assumptions about the information structure? We could restrict attention to PIF information structures. Or we could use a different notion of informational substitutes. Or – following the lead of Arieli et al. (2018) – we could assume that the experts’ signals are independent conditioned on the value of $Y$, or that they are Blackwell ordered.
- What if the aggregator knows more about the information structure, such as the covariance matrix of the experts’ estimates, or perhaps the entire joint probability distribution of their

\(^{14}\)We also showed a negative result in the setting where the aggregator knows the entire information structure.
estimates?\textsuperscript{15}

• What if we choose a different benchmark? The approximation ratio as we have defined it seems like a natural choice, but there may be others as well.

• What if the aggregator learns information other than the experts’ expected values? Perhaps the aggregator instead learns the median of each expert’s probability distribution over $Y$. Or, if the aggregator has detailed knowledge of the information structure, then it may make sense to ask what happens if the aggregator learns partial information about the experts’ \textit{signals}.

• What if the experts are strategic? Perhaps they want to influence the aggregate forecast. What if the experts are biased in some way?

• What if $Y$ is vector-valued, rather than real-valued?

Many of these settings (and the vast majority of combinations of these settings) have yet to be considered. The field of robust forecast aggregation has great potential to grow, and to produce results that are both theoretically interesting and practically useful.

\textsuperscript{15}While our $\frac{1}{m}$ negative result applies in our setting regardless of how much the aggregator knows, in other settings there may be strong approximation guarantees if the expert knows more information than just the prior.
Chapter 8: When does agreement imply accuracy?

This chapter presents “Agreement Implies Accuracy for Substitutable Signals” (Frongillo et al., 2023). It assumes background on Bregman divergence (Section 2.1.3) and information structures (Section 2.3).

Summary: Inspired by Aumann’s agreement theorem, Aaronson (2005) studied the amount of communication necessary for two Bayesian experts to approximately agree on the expectation of a random variable. Aaronson showed that, remarkably, the number of bits does not depend on the amount of information available to each expert. However, in general the agreed-upon estimate may be inaccurate: far from the estimate they would settle on if they were to share all of their information. We show that if the experts’ signals satisfy a particular notion of informational substitutes, then it is the case that if the experts are close to agreement then they are close to the truth. We prove this result for a broad class of agreement and accuracy measures that includes squared distance and KL divergence. Additionally, we show that although these measures capture fundamentally different kinds of agreement, Aaronson’s agreement result generalizes to them as well.

8.1 Introduction

Suppose that Alice and Bob are honest, rational Bayesians who wish to estimate some quantity – say, the unemployment rate one year from now. Alice is an expert on historical macroeconomic trends, while Bob is an expert on contemporary monetary policy. They convene to discuss and share their knowledge with each other until they reach an agreement about the expected value of the future unemployment rate. Alice and Bob could reach agreement by sharing everything they
had ever learned, at which point they would have the same information, but the process would take years. How, then, should they proceed?

In the seminal work “Agreeing to Disagree,” Aumann (1976) observed that Alice and Bob can reach agreement simply by taking turns sharing their current expected value for the quantity. In addition to modeling communication between Bayesian agents, protocols similar to this one model financial markets: each trader shares partial information about their expected value on their turn (discussed in Section 8.5). A remarkable result by Aaronson (2005) shows that if Alice and Bob follow certain protocols of this form, they will agree to within $\epsilon$ with probability $1 - \delta$ by communicating $O\left(\frac{1}{\delta \epsilon^2}\right)$ bits.\(^1\) Notably, this bound only depends on the error Alice and Bob are willing to tolerate, and not on the amount of information available to them.

Absent from Aaronson’s results, however, is what estimate Alice and Bob end up agreeing on. In particular, there is no guarantee that Alice and Bob will be accurate, meaning their agreed-upon estimate will be close (in e.g. expected squared distance) to what they would believe if they shared all of their information. In fact, they might agree on an estimate that is highly inaccurate: suppose that Alice and Bob have independent, uniformly random bits $b_A, b_B$, and wish to estimate their XOR $b_A \oplus b_B$.\(^2\) Alice and Bob agree from the onset, as from each of their perspectives, the expected value of $b_A \oplus b_B$ is $\frac{1}{2}$. Yet this expectation is far from the best estimate given their collective knowledge, which is either 0 or 1. So while agreement is fundamental to understanding communication between Bayesians – in Aumann’s terms, they cannot “agree to disagree” – agreement is far from the whole story. An important open problem, therefore, is what assumptions guarantee that Alice and Bob are accurate once they agree.

We address this open problem by introducing a natural condition, called rectangle substitutes, under which agreement implies accuracy. Rectangle substitutes is a notion of informational substitutes (discussed in Section 2.3.2): the property that additional information has diminishing marginal returns. The notion of substitutes is ubiquitous in optimization problems, and infor-

\(^1\)To ensure that each message is short, Alice and Bob share discretized versions of their estimates; we discuss this in Section 8.2.
\(^2\)In Section 2.3.2 we introduced this example and called it the “XOR information structure.”
mational substitutes conditions have recently been used to analyze equilibria in markets (Chen & Waggoner, 2016). In that context, Kong and Schoenebeck (2023) showed for conditionally independent signals convergence of the popular LMSR market implies full information aggregation, i.e. accuracy. We show that under the rectangle substitutes condition, any protocol leading to agreement will also lead to accuracy. We then extend these results beyond the case of squared error, to a broad family of measures of agreement and accuracy including KL divergence.\(^3\)

### 8.1.1 Overview of approach and results

In Aaronson (2005), Alice and Bob are said to agree if the squared difference between their estimates is small. Likewise, we can say that Alice and Bob are accurate if the squared distance between each of their estimates and the truth is small. In Section 8.3 we present our first main result: under these definitions, if the information structure describing Alice and Bob’s signals satisfies the rectangle substitutes condition, then agreement implies accuracy. In other words, under this assumption, when two Bayesians agree – regardless of how little information they have shared – they necessarily agree on the truth.

The proof involves carefully partitioning the space of posterior beliefs induced by the protocol. Agreement is used to show that Alice and Bob usually fall into the same partition element, which means that Bob would not learn much from learning the partition element of Alice’s expectation. Then, the rectangle substitutes condition is used to show that if Bob were to learn Alice’s partition element, then he would be very close to knowing the truth.

Aaronson measures agreement in terms of squared error, yet other measurements like KL divergence may be better suited for some settings. For example, if Alice and Bob estimate the probability of a catastrophic event as 10\(^{-10}\) and 10\(^{-2}\), respectively, then under squared error they are said to agree closely, but arguably they disagree strongly, as reflected by their large KL divergence. Motivated by these different ways to measure agreement, we next ask:

(1) Can Aaronson’s protocols be generalized to other notions of agreement, such that the number

\(^3\)Specifically, agreement and accuracy with respect to (almost) arbitrary Bregman divergences.
of bits communicated is independent of the amount of information available to Alice and Bob?

(2) Do other notions of agreement necessarily imply accuracy under rectangle substitutes?

In Section 8.4, we give our second and third main results: **the answer to both questions is yes.** Specifically, the positive results apply when measuring agreement and accuracy using Bregman divergences, a class of error measures that includes both squared distance and KL divergence.4

Aaronson’s proof of his agreement theorem turns out to be specific to squared distance. Our agreement theorem (Theorem 8.4.8) modifies Aaronson’s protocol to depend on the particular Bregman divergence, i.e. the relevant error measure. It then proceeds in a manner inspired by Aaronson but using several new ideas. Our proof that agreement implies accuracy under rectangle substitutes for general Bregman divergences also involves some nontrivial changes to our proof for squared distance. In particular, the fact that the length of an interval cannot be inferred from the Bregman divergence between its endpoints necessitates a closer analysis of the partition of Alice’s and Bob’s beliefs.

We conclude in Section 8.5 with a discussion of connections between agreement protocols and information revelation in financial markets, and discuss an interesting potential avenue for future work.

### 8.1.2 Related work

Geanakoplos and Polemarchakis (1982) discussed the distinction between agreement and full information revelation. One result shown is that under a natural probability measure on information structures, full agreement and information revelation occur in a single round of communication with probability one. However, conditions for accuracy and the concept of substitutes are not discussed.

Our setting is related to but distinct from communication complexity. In that field (e.g. Rao 196)

---

4The third result holds under an “approximate triangle inequality” condition on the Bregman divergence, which is satisfied by most or all natural choices; indeed, it is nontrivial to construct a Bregman divergence that does not satisfy this property.
and Yehudayoff (2020)), the goal is for Alice and Bob to correctly compute a function of their inputs while communicating as few bits as possible and using any protocol necessary. By contrast, Aaronson (2005) considered a goal of agreement, not correctness, and focused on specific natural protocols, which he showed achieve this goal in a constant number of bits. Our work focuses on Aaronson’s setting. We discuss how our results might be framed in terms of communication complexity in Appendix D.5.

Our introduction of the substitutes condition is inspired by its usefulness in prediction markets (Chen & Waggoner, 2016). The “expectation-sharing” agreement protocols we study bear a strong similarity to dynamics of market prices. Ostrovsky (2012) introduced a condition under which convergence of prices in a market implies that all information is aggregated. This can be viewed as an “agreement implies accuracy” condition. Similarly, Kong and Schoenebeck (2023) presented a result that, for the logarithmic market scoring rule (LMSR) and conditionally independent signals, convergence of the market implies full information revelation. Our results are conceptually similar, although they are technically quite different as we rely on the novel condition of rectangle substitutes. In the context of the LMSR, the rectangle substitutes notion includes conditionally independent signals as a special case (see discussion in Section 8.4.1). We discuss the connection of our work to markets in Section 8.5.

8.2 Preliminaries

8.2.1 Information structures

In Section 2.3, we introduced the concept of an information structure as a tuple consisting of a probability distribution \( P \) over a set \( \Omega \) of states of the world, a quantity \( Y \), and a tuple of \( m \) signals that give partial information about the state of the world \( \omega \) (and thus about \( Y \)).

In this chapter, we specifically consider the case of \( m = 2 \) signals and choose notation accordingly. We will say that there are two experts, Alice and Bob. Alice’s signal is \( \sigma : \Omega \rightarrow S \) and Bob’s signal is \( \tau : \Omega \rightarrow T \). Thus, we will think of information structures as 5-tuples \((\Omega, P, \sigma, \tau, Y)\). We
additionally assume (following Aaronson) that \( Y \) takes values in \([0, 1]\).

We denote by \( \mu_{\sigma \tau} := \mathbb{E} [Y \mid \sigma, \tau] \) the random variable that is equal to the expected value of \( Y \) conditioned on both Alice’s signal \( \sigma \) and Bob’s signal \( \tau \).\(^5\) We also define \( \mu_{\sigma} := \mathbb{E} [Y \mid \sigma] \) and \( \mu_{\tau} := \mathbb{E} [Y \mid \tau] \). For a measurable set \( S \subseteq S \), we define \( \mu_{S} := \mathbb{E} [Y \mid \sigma \in S] \); we define \( \mu_{T} \) analogously for \( T \subseteq T \). Additionally, for \( T \subseteq T \), we define \( \mu_{\sigma T} := \mathbb{E} [Y \mid \tau \in T, \sigma] \), i.e. the expected value of \( Y \) conditioned on the particular value of \( \sigma \) and the knowledge that \( \tau \in T \). If Alice knows that Bob’s signal belongs to \( T \) (and nothing else about his signal), then the expected value of \( Y \) conditional on her information is \( \mu_{\sigma T} \); we refer to this as Alice’s expectation. Likewise, for \( S \subseteq S \), we define \( \mu_{S \tau} := \mathbb{E} [Y \mid \sigma \in S, \tau] \). Finally, we define \( \mu_{ST} := \mathbb{E} [Y \mid \sigma \in S, \tau \in T] \). This is the expectation of a third party who only knows that \( \sigma \in S \) and \( \tau \in T \).

In general we often wish to take expectations conditioned on \( \sigma \in S, \tau \in T \) (for some \( S \subseteq S, T \subseteq T \)). We will use the shorthand \( \mathbb{E} [\cdot \mid S, T] \) for \( \mathbb{E} [\cdot \mid \sigma \in S, \tau \in T] \) in such cases.

### 8.2.2 Agreement protocols

The notion of agreement between Alice and Bob is central to our work. We first define agreement in terms of squared error, and generalize to other error measures in Section 8.4.

**Definition 8.2.1** (\( \epsilon \)-agreement). Let \( a \) and \( b \) be Alice’s and Bob’s expectations, respectively (\( a \) and \( b \) are random variables defined on \( \Omega \)). Alice and Bob \( \epsilon \)-agree if \( \frac{1}{4} \mathbb{E} \left[ (a - b)^2 \right] \leq \epsilon \).

The constant \( \frac{1}{4} \) makes the left-hand side represent Alice’s and Bob’s squared distance to the average of their expectations.

Our setting follows Aaronson (2005), which examined communication protocols that cause Alice and Bob to agree. In a \((deterministic) \) communication protocol, Alice and Bob take turns sending each other messages (strings of bits). On Alice’s turns, Alice communicates a message that is a deterministic function of her input (i.e. her signal \( \sigma \)) and all previous communication, and likewise for Bob on his turns. A rectangle is a set of the form \( S \times T \) where \( S \subseteq S \) and \( T \subseteq T \).

\(^5\)The value of \( Y \) need not be determined by \( \sigma \) and \( \tau \), although for our purposes the case in which it is determined is essentially equivalent.
The communication transcript is the ordered tuple of all messages that have been sent. The transcript at time step \( t \) refers to the tuple consisting of the first \( t \) messages. The transcript at time step \( t \) partitions \( \Omega \) into rectangles: for any given sequence of \( t \) messages, there are subsets \( S_t \subseteq S, T_t \subseteq T \) such that the protocol transcript at time \( t \) is equal to this sequence if and only if \( (\sigma, \tau) \in S_t \times T_t \).

For a given communication protocol, we may think of \( S_t \) and \( T_t \) as random variables. Alice’s expectation at time \( t \) (i.e. after the \( t \)-th message has been sent) is \( \mu_{\sigma T_t} \) and Bob’s expectation at time \( t \) is \( \mu_{S_t \tau} \). Finally, the protocol terminates at a certain time (which need not be known in advance of the protocol). While typically in communication complexity a protocol is associated with a final output, in this case we are interested in Alice’s and Bob’s expectations, so we do not require an output.

It will be convenient to hypothesize a third party observer, whom we call Charlie, who observes the protocol but has no other information. At time \( t \), Charlie has expectation \( \mu_{S_t T_t} \). Charlie’s expectation can also be interpreted as the expectation of \( Y \) according to Alice and Bob’s common knowledge. Note that Alice and Bob each know Charlie’s expectation at any given time.

The following definition formalizes the relationship between communication protocols and agreement.

**Definition 8.2.2 (\( \epsilon \)-agreement protocol).** Given an information structure \( I \), a communication protocol causes Alice and Bob to \( \epsilon \)-agree on \( I \) if Alice and Bob \( \epsilon \)-agree at the end of the protocol, i.e., if \( \frac{1}{4} \mathbb{E} [ (\mu_{\sigma T_t} - \mu_{S_t \tau})^2 ] \leq \epsilon \), where the expected value is over Alice’s and Bob’s inputs. We say that a communication protocol is an \( \epsilon \)-agreement protocol if the protocol causes Alice and Bob to \( \epsilon \)-agree on every information structure.

Aaronson defines and analyzes two \( \epsilon \)-agreement protocols. A minor difference to our framing is that Aaronson (2005) focuses on probable approximate agreement: protocols that cause the absolute difference between Alice and Bob to be at most \( \epsilon \) with probability all but \( \delta \). While the results as presented in this section are stronger than those in Aaronson (2005) (the original results follow from these as a consequence of Markov’s inequality), these results follow from a straightforward modification of his proofs.
protocol, in which Alice and Bob take turns stating their expectations for a number of time steps that can be computed by Alice and Bob independently in advance of the protocol, and which is guaranteed to be at most $O(1/\epsilon)$.

The fact that exchanging their expectations for $O(1/\epsilon)$ time steps results in $\epsilon$-agreement is profound and compelling. However, the standard protocol may require an unbounded number of bits of communication, since Alice and Bob are exchanging real numbers. To address this, Aaronson defines another agreement protocol that is truly polynomial-communication (which we slightly modify for our purposes):

**Definition 8.2.3 (Discretized protocol, Aaronson (2005)).** Choose $\epsilon > 0$. In the discretized protocol with parameter $\epsilon$, on her turn (at time $t$), Alice sends “low” if her expectation is smaller than Charlie’s by more than $\epsilon/4$, i.e. if $\mu_{S, t-1} < \mu_{S, t-1}T_{t-1} - \epsilon/4$; “high” if her expectation is larger than Charlie’s by more than $\epsilon/4$; and “medium” otherwise. Bob acts analogously on his turn. At the start of the protocol, Alice and Bob use the information structure to independently compute the time $t_{end} \leq 1000/\epsilon$ that minimizes $\mathbb{E} \left[ (\mu_{\sigma T_{end}} - \mu_{S_{end}})^2 \right]$. The protocol ends at this time.

**Theorem 8.2.4 (Aaronson, 2005, Theorem 4).** The discretized protocol with parameter $\epsilon$ is an $\epsilon$-agreement protocol with transcript length $O(1/\epsilon)$ bits.

In general, we refer to Aaronson’s standard and discretized protocols as examples of expectation-sharing protocols. We will define other examples in Section 8.4, similar to Aaronson’s discretized protocol but with different cutoffs for low, medium, and high. We also interpret expectation-sharing protocols in the context of markets in Section 8.5.

### 8.2.3 Accuracy and informational substitutes

Most of our main results give conditions such that if Alice and Bob $\epsilon$-agree, then Alice’s and Bob’s estimates are accurate. By *accurate*, we mean that Alice’s and Bob’s expectations are close to $\mu_{\sigma T}$, i.e., what they would believe if they knew each other’s signals. (After all, they cannot hope
to have a better estimate of \( Y \) than \( \mu_{\sigma \tau} \); for this reason we sometimes refer to \( \mu_{\sigma \tau} \) as the “truth.”

Formally:

**Definition 8.2.5** (\( \epsilon \)-accuracy). Let \( a \) be Alice’s expectation. Alice is \( \epsilon \)-accurate if
\[
\mathbb{E} \left[ (\mu_{\sigma \tau} - a)^2 \right] \leq \epsilon.
\]
We define \( \epsilon \)-accuracy analogously for Bob.

One cannot hope for an unconditional result stating that if Alice and Bob agree, then they are accurate. Consider for instance the XOR information structure from the introduction: Alice and Bob each receive independent random bits as input, and \( Y \) is the XOR of these bits. Then from the start Alice and Bob agree that the expected value of \( Y \) is exactly \( \frac{1}{2} \), but this value is far from \( \mu_{\sigma \tau} \), which is either 0 or 1.

Intuitively, this situation arises because Alice’s and Bob’s signals are informational complements: each signal is not informative by itself, but they are informative when taken together. On the other hand, we say that signals are informational substitutes if learning one signal is less valuable if you already know the other signal. We introduced and motivated the concept of informational substitutes in Section 2.3.2. In particular, we defined weak substitutes (first introduced by Chen and Waggoner (2016)) as a formalization of the notion of diminishing marginal returns to learning an extra signal. An extreme example of informational substitutes is if \( \sigma \) and \( \tau \) both specify the value of \( Y \) exactly. In that case, \( \sigma \) becomes useless upon learning \( \tau \) and vice versa. Our definition is inspired by the definition of weak substitutes, but we require a stronger notion for our results to hold. For the following definition, recall that we write \( |S, T| \) as shorthand for \( |\sigma \in S, \tau \in T| \).

**Definition 8.2.6.** An information structure \( I = (\Omega, \mathbb{P}, \sigma, \tau, Y) \) satisfies rectangle substitutes if for every \( S \subseteq S, T \subseteq T \) such that \( \mathbb{P}[\sigma \in S, \tau \in T] > 0 \), we have
\[
\mathbb{E} \left[ (Y - \mu_{ST})^2 \mid S, T \right] - \mathbb{E} \left[ (Y - \mu_{\sigma \tau})^2 \mid S, T \right] \leq \mathbb{E} \left[ (Y - \mu_{ST})^2 \mid S, T \right] - \mathbb{E} \left[ (Y - \mu_{\sigma \tau})^2 \mid S, T \right].
\]
(8.1)

This definition is a strengthening of weak substitutes for two agents: an information structure satisfies weak substitutes if Equation 8.1 holds specifically for \( S = S \) and \( T = T \). We will show
that under the rectangle substitutes condition, if Alice and Bob approximately agree, then they are approximately accurate.

**Interpreting substitutes** Both sides of Equation 8.1 represent the “value” of learning $\sigma$ as measured by a decrease in error. The left-hand side gives the decrease if one already knows $\tau$ and that $\sigma \in S$; the right-hand side gives the decrease if one only knows that $\sigma \in S, \tau \in T$. Substitutes thus says: the marginal value of learning $\sigma$ is smaller if one already knows $\tau$ than if one does not. This statement should hold for every sub-rectangle $S, T$. Note that the inequality can be rearranged to focus instead on the marginal value of $\tau$ rather than $\sigma$. Note also that in the XOR information structure, the left-hand side of the inequality is $1/4$ while the right-hand side is zero: a large violation of the substitutes condition. In the example where $\sigma$ and $\tau$ both specify the value of $Y$, the left side is always zero.

Chen and Waggoner (2016) discusses three interpretations of substitutes, which motivate it as a natural condition. (1) Each side of the inequality measures an improvement in prediction error, here the squared loss, due to learning $\sigma$. Under substitutes, the improvement is smaller if one already knows $\tau$. (2) Each side measures a decrease in uncertainty (here, measured roughly by variance) due to learning $\sigma$. Under substitutes, $\sigma$ provides less information about $Y$ if one already knows $\tau$. (3) Each side measures the decrease in distance of a posterior expectation from the truth when learning $\sigma$. The distance to $Y$ changes less if one already knows $\tau$.

**Restrictiveness of substitutes** It is natural to ask about the strength of the rectangle substitutes assumption. In the case that $|S| = |T| = 2$, the condition reduces to the aforementioned and well-established weak substitutes condition. For larger signal sets, the set of information structures satisfying rectangle substitutes remains nontrivial. For example, it is satisfied by a positive fraction of information structures (for a natural choice of measure). We show this fact in Appendix D.1 by exhibiting an information structure in which Equation 8.1 holds strictly for all $S, T$.

---

8Here, uncertainty is measured by variance of one’s belief. Under the KL divergence analogue covered in Section 8.4.1, uncertainty is measured in bits via Shannon entropy.
with \(|S|, |T| \geq 2\) (and thus, that all nearby information structures also satisfy rectangle substitutes).

Finally, we note that although the rectangle substitutes condition is strong due to the quantification over sub-rectangles, in Section 8.3.3 we prove that our main results decay gracefully for information structures that are close to but do not quite satisfy the rectangle substitutes condition.

### 8.2.4 The Pythagorean theorem

We recall the Pythagorean theorem from Section 2.3.3:

**Proposition 2.3.11.** (Pythagorean theorem for squared distance). In a probability space \((\Omega, \mathcal{F}, \mathbb{P})\), let \(A\) be a real-valued random variable, let \(B = \mathbb{E}[A \mid \mathcal{H}]\) where \(\mathcal{H}\) is a sub-sigma-algebra,\(^9\) and \(C\) be a real-valued random variable defined on \(\mathcal{H}\). Then

\[
\mathbb{E}[(A - C)^2] = \mathbb{E}[(A - B)^2] + \mathbb{E}[(B - C)^2].
\]

One application of the Pythagorean theorem in our context takes \(A = Y\), \(B = \mu_{o\tau}\) (the expected value of \(Y\) conditioned on the experts’ signals), and \(C = \mu_{o\tau}\) (Alice’s expected value, which only depends on her signal and thus on the signal pair). This particular application, along with the symmetric one taking \(C = \mu_{S\tau}\), allows us to rewrite the rectangle substitutes condition in a form that we will find more convenient:

**Remark 8.2.7.** An information structure \(I\) satisfies rectangle substitutes if and only if

\[
\mathbb{E}[(\mu_{o\tau} - \mu_{S\tau})^2 \mid S, T] \leq \mathbb{E}[(\mu_{o\tau} - \mu_{ST})^2 \mid S, T]
\]

for all \(S, T\) such that \(\mathbb{P}[\sigma \in S, \tau \in T] > 0\).

\(^9\)Readers unfamiliar with sigma-algebras can think of \(\mathcal{H}\) as supplying partial information about the state of the world \(\omega\) (much like a signal). A random variable is defined on \(\mathcal{H}\) if its value depends only on the information that \(\mathcal{H}\) provides (i.e. the value of the signal).
8.3 Results for squared distance

Our main results show that, under the rectangle substitutes condition, any communication protocol that causes Alice and Bob to agree also causes them to be accurate. We now show the first of these results, which is specific to the squared distance error measure that we have been discussing.

8.3.1 Agreement implies accuracy

**Theorem 8.3.1.** Let \( I = (\Omega, \mathbb{P}, \sigma, \tau, Y) \) be an information structure that satisfies rectangle substitutes. For any communication protocol that causes Alice and Bob to \( \epsilon \)-agree on \( I \), Alice and Bob are \( 10\epsilon^{1/3} \)-accurate after the protocol terminates.

The crux of the argument is the following lemma.

**Lemma 8.3.2.** Let \( I = (\Omega, \mathbb{P}, \sigma, \tau, Y) \) be an information structure that satisfies rectangle substitutes. Let \( \epsilon = \mathbb{E} \left[ (\mu_\sigma - \mu_\tau)^2 \right] \). Then

\[
\mathbb{E} \left[ (\mu_\sigma - \mu_\tau)^2 \right] \leq 6\epsilon^{1/3}.
\]

Let us first prove Theorem 8.3.1 assuming Lemma 8.3.2 is true.

**Proof of Theorem 8.3.1.** Consider any protocol that causes Alice and Bob to \( \epsilon \)-agree on \( I \). Let \( S \) be the set of possible signals of Alice at the end of the protocol which are consistent with the protocol transcript, and define \( T \) likewise for Bob. Intuitively, \( S \times T \) is the set of plausible signal pairs \( (\sigma, \tau) \) according to an external observer of the protocol. Observe that \( S \) and \( T \) are random variables, each a function of both \( \sigma \) and \( \tau \). We have

\[
\mathbb{E} \left[ (\mu_\sigma - \mu_S)^2 \right] = \mathbb{E}_{S,T} \left[ \mathbb{E} \left[ (\mu_\sigma - \mu_S)^2 \mid S, T \right] \right] \\
\leq \mathbb{E}_{S,T} \left[ 6 \left( \mathbb{E} \left[ (\mu_\tau - \mu_S)^2 \mid S, T \right] \right)^{1/3} \right] \\
\leq 6 \mathbb{E}_{S,T} \left[ \mathbb{E} \left[ (\mu_\tau - \mu_S)^2 \mid S, T \right] \right]^{1/3}
\]
\[
= 6 \mathbb{E} \left[ (\mu_{\sigma T} - \mu_{S_T})^2 \right]^{1/3} \leq 6(4\epsilon)^{1/3} \leq 10\epsilon^{1/3}.
\]

In the second step, we apply Lemma 8.3.2 to the information structure \( I \) restricted to \( S \times T \) – that is, to \( I' = (\Omega', \mathbb{P}', S, T, Y) \), where \( \Omega' = \{ \omega \in \Omega : \sigma \in S, \tau \in T \} \) and \( \mathbb{P}'[\omega] = \mathbb{P}[\omega | \sigma \in S, \tau \in T] \). (Note that we use the fact that if \( I \) satisfies rectangle substitutes, then so does \( I' \); this is because a rectangle of \( I' \) is also a rectangle of \( I \).) The third step follows by the concavity of \( x^{1/3} \). Therefore, Bob is \( 10\epsilon^{1/3} \) accurate (and Alice is likewise by symmetry).

The proof of Lemma 8.3.2 relies on the following claim.

**Claim 8.3.3.** In the setting of Lemma 8.3.2, for any \( N \geq 1 \), it is possible to partition \([0, 1]\) into \( N \) intervals \([0, x_1), [x_1, x_2), \ldots, [x_{N-1}, 1]\) in a way so that each interval has length at most \( \frac{2}{N} \), and

\[
\mathbb{P}[k(\sigma) \neq k(\tau)] \leq \sqrt{\epsilon}N,
\]

where \( k(\sigma) \) denotes the \( k \in [N] \) such that \( x_{k-1} \leq \mu_{\sigma} < x_k \), and \( k(\tau) \) is defined analogously.\(^{10}\)

Intuitively, Claim 8.3.3 is true because if \( \mathbb{E} \left[ (\mu_{\sigma} - \mu_{\tau})^2 \right] \) is small, then \( \mu_{\sigma} \) and \( \mu_{\tau} \) are likely to fall into the same interval.

**Proof.** We claim that in fact we can choose the \( x_i \)'s so that each \( x_i \) is in \( \left[ \frac{i}{N} - \frac{1}{2N}, \frac{i}{N} + \frac{1}{2N} \right] \). This ensures that each interval has length at most \( \frac{2}{N} \).

For \( x \in [0, 1] \), let \( \rho(x) \) be the probability that \( x \) is between \( \mu_{\sigma} \) and \( \mu_{\tau} \), inclusive. Note that

\[
\mathbb{P}[k(\sigma) \neq k(\tau)] \leq \sum_{i=1}^{N-1} \rho(x_i).
\]

Observe that if \( x \) is selected uniformly from \([0, 1]\), the expected value of \( \rho(x) \) is equal to \( |\mu_{\sigma} - \mu_{\tau}| \), because both quantities are equal to the probability that \( x \) is between \( \mu_{\sigma} \) and \( \mu_{\tau} \). Therefore, if \((\sigma, \tau)\) is additionally chosen according to \( \mathbb{P} \), we have

\[
\mathbb{E}_{x \sim [0,1]} [\rho(x)] = \mathbb{E} [ |\mu_{\sigma} - \mu_{\tau}|] \leq \sqrt{\mathbb{E} \left[ (\mu_{\sigma} - \mu_{\tau})^2 \right]} = \sqrt{\epsilon}.
\]

\(^{10}\)For convenience we define \( x_0 = 0 \) and \( x_N \) to be some number greater than 1.
This means that
\[
\sum_{i=1}^{N-1} \mathbb{E}_{x \leftarrow \left[\frac{i}{N}, \frac{i+1}{N}\right]} [\rho(x)] = (N - 1) \mathbb{E}_{x \leftarrow \left[\frac{1}{2N}, \frac{1}{2N} + \frac{1}{N}\right]} [\rho(x)] \leq \sqrt{\epsilon N}.
\]

Thus, if each \(x_i\) is selected uniformly at random from \(\left[\frac{i}{N} - \frac{1}{2N}, \frac{i}{N} + \frac{1}{2N}\right]\), the expected value of \(\mathbb{P}[k(\sigma) \neq k(\tau)]\) would be at most \(\sqrt{\epsilon N}\). In particular this means that there exist choices of the \(x_i\)'s such that \(\mathbb{P}[k(\sigma) \neq k(\tau)] \leq \sqrt{\epsilon N}\).

We now prove Lemma 8.3.2.

**Proof of Lemma 8.3.2.** Fix a large positive integer \(N\) (we will later find it optimal to set \(N = \epsilon^{-1/6}\)). Consider a partition of \([0, 1]\) into \(N\) intervals \([0, x_1), [x_1, x_2), \ldots, [x_{N-1}, 1]\) satisfying the conditions of Claim 8.3.3. Let \(S^{(k)} := \{\sigma \in S : x_{k-1} \leq \mu_\sigma < x_k\}\). Additionally, let \(k(\sigma)\) and \(k(\tau)\) be as defined in Claim 8.3.3.

Our goal is to upper bound the expectation of \((\mu_{\sigma T} - \mu_\tau)^2\). In pursuit of this goal, we observe that by the Pythagorean theorem, we have

\[
\mathbb{E} \left[ (\mu_{\sigma T} - \mu_\tau)^2 \right] = \mathbb{E} \left[ (\mu_{\sigma T} - \mu_{S^{(k(\sigma))} T})^2 \right] + \mathbb{E} \left[ (\mu_{S^{(k(\sigma))} T} - \mu_\tau)^2 \right] .
\]

We now use the rectangle substitutes assumption: for any \(k\), by applying Equation 8.2 to \(S = S^{(k)}\) and \(T = T\), we know that

\[
\mathbb{E} \left[ (\mu_\sigma - \mu_{S^{(k)}})^2 \mid \sigma \in S^{(k)} \right] \geq \mathbb{E} \left[ (\mu_{\sigma T} - \mu_{S^{(k) T}})^2 \mid \sigma \in S^{(k)} \right] .
\]

Taking the expectation over \(k\) (i.e. choosing each \(k\) with probability equal to \(\mathbb{P}[\sigma \in S^{(k)}]\)), we have that

\[
\mathbb{E} \left[ (\mu_\sigma - \mu_{S^{(k(\sigma))}})^2 \right] \geq \mathbb{E} \left[ (\mu_{\sigma T} - \mu_{S^{(k(\sigma)) T}})^2 \right] . \tag{8.3}
\]
Therefore, we have

$$\mathbb{E}[(\mu_{\sigma^T} - \mu_T)^2] \leq \mathbb{E}[(\mu_{\sigma^T} - \mu_{S(k(\sigma))})^2] + \mathbb{E}[(\mu_{S(k(\sigma))\tau} - \mu_T)^2] \quad (8.4)$$

We will use Claim 8.3.3 to argue that each of these two summands is small. The argument regarding the first summand is straightforward: for any $\sigma$, we have that $x_k(\sigma) \leq \mu_\sigma, \mu_{S(k(\sigma))} < x_k(\sigma) + 1 \leq x_k(\sigma) + \frac{2}{N}$, which means that $\mathbb{E}[(\mu_{\sigma^T} - \mu_{S(k(\sigma))})^2] \leq \frac{4}{N^2}$.

We now upper bound the second summand.\(^{11}\) For any $\hat{\tau} \in T$, let $p(\hat{\tau}) = \mathbb{P}[\tau = \hat{\tau}]$ and $q(\hat{\tau}) = \mathbb{P}[\tau = \hat{\tau}, k(\sigma) \neq k(\tau)]$. Then $\sum_{\hat{\tau} \in T} p(\hat{\tau}) = 1$ and $\sum_{\hat{\tau} \in T} q(\hat{\tau}) \leq \sqrt{\varepsilon N}$. Observe that

$$\mathbb{E}[(\mu_{S(k(\sigma))\tau} - \mu_T)^2] = \sum_{\hat{\tau}} p(\hat{\tau}) \mathbb{E}[(\mu_{S(k(\sigma))\hat{\tau}} - \mu_{\hat{\tau}})^2 \mid \tau = \hat{\tau}]$$

$$= \sum_{\hat{\tau}} (p(\hat{\tau}) - q(\hat{\tau})) \mathbb{E}[(\mu_{S(k(\sigma))\hat{\tau}} - \mu_{\hat{\tau}})^2 \mid \tau = \hat{\tau}, k(\sigma) = k(\hat{\tau})] + q(\hat{\tau}) \mathbb{E}[(\mu_{S(k(\sigma))\hat{\tau}} - \mu_{\hat{\tau}})^2 \mid \tau = \hat{\tau}, k(\sigma) \neq k(\hat{\tau})] \quad (8.5)$$

To handle the first expectation, we note that if $k(\sigma) = k(\hat{\tau})$, then $|\mu_{S(k(\sigma))\hat{\tau}} - \mu_{\hat{\tau}}| \leq \frac{q(\hat{\tau})}{p(\hat{\tau})}$. To see this, observe

$$p(\hat{\tau}) \mu_{\hat{\tau}} = (p(\hat{\tau}) - q(\hat{\tau})) \mu_{S(k(\sigma))\hat{\tau}} + q(\hat{\tau}) \mu_{S \setminus S(k(\sigma))\hat{\tau}}.$$ 

Rearranging and taking absolute values, we conclude

$$p(\hat{\tau}) |\mu_{S(k(\sigma))\hat{\tau}} - \mu_{\hat{\tau}}| = q(\hat{\tau}) |\mu_{S(k(\sigma))\hat{\tau}} - \mu_{S \setminus S(k(\sigma))\hat{\tau}}| \leq q(\hat{\tau}).$$

Therefore, recalling $q(\hat{\tau}) \leq p(\hat{\tau})$, we have

$$p(\hat{\tau}) - q(\hat{\tau}) \mathbb{E}[(\mu_{S(k(\sigma))\hat{\tau}} - \mu_{\hat{\tau}})^2 \mid \tau = \hat{\tau}, k(\sigma) = k(\hat{\tau})] \leq (p(\hat{\tau}) - q(\hat{\tau})) \left( \frac{q(\hat{\tau})}{p(\hat{\tau})} \right)^2 \leq \frac{q(\hat{\tau})^2}{p(\hat{\tau})} \leq q(\hat{\tau}).$$

On the other hand, we can bound the second expectation in Equation 8.5 by 1. Therefore we

\(^{11}\)The proof below takes sums over $\hat{\tau} \in T$ and thus implicitly assumes that $T$ is finite, but the proof extends to infinite $T$, with sums over $\tau$ replaced by integrals with respect to the probability measure over $T$. 

207
have
\[
\mathbb{E} \left[ (\mu_{S(k(\sigma))} - \mu_\tau)^2 \right] \leq \sum_\hat{\tau} (q(\hat{\tau}) + q(\hat{\tau})) = 2 \sum_\hat{\tau} q(\hat{\tau}) \leq 2 \sqrt{\varepsilon N}.
\]

To conclude, we now know that
\[
\mathbb{E} \left[ (\mu_\sigma - \mu_\tau)^2 \right] \leq \frac{4}{N^2} + 2 \sqrt{\varepsilon N}.
\]

Setting \( N = \varepsilon^{-1/6} \) makes the right-hand side equal to \( 6\varepsilon^{1/3} \), completing the proof.

Now that we have proven Lemma 8.3.2, our proof of Theorem 8.3.1 is complete.

### 8.3.2 Consequences of Theorem 8.3.1

Theorem 8.3.1 is a general result about agreement protocols. Applying the result to Aaronson’s discretized protocol gives us the following result.

**Corollary 8.3.4.** Let \( I \) be any information structure that satisfies universal rectangle substitutes. For any \( \varepsilon > 0 \), Alice and Bob will be \( \varepsilon \)-accurate after running Aaronson’s discretized protocol with parameter \( \varepsilon^3 / 1000 \) (and this takes \( O(1/\varepsilon^3) \) bits of communication).

**Remark 8.3.5.** The discretized protocol is not always the most efficient agreement protocol. For example, Proposition D.2.1 shows that if the rectangle substitutes condition holds, agreement (and therefore accuracy) can be reached with just \( O(\log(1/\varepsilon)) \) bits, an improvement on Corollary 8.3.4. We discuss communication complexity further in Appendix D.5. Even if more efficient protocols are sometimes possible, expectation-sharing protocols are of interest because they model naturally-occurring communication processes. For example, they capture the dynamics of prices in markets, which we also discuss in Section 8.5. More generally, we find it remarkable that Alice and Bob become accurate by running Aaronson’s agreement protocol (or indeed any agreement protocol), despite such protocols being designed with only agreement in mind.

Finally, we observe the following important consequence of Theorem 8.3.1: once Alice and Bob agree, they continue to agree.
Corollary 8.3.6. Let $I = (\Omega, \mathbb{P}, \sigma, \tau, Y)$ be an information structure that satisfies rectangle substitutes. Consider a communication protocol with the property that Alice and Bob $\epsilon$-agree after round $t$. Then Alice and Bob $10\epsilon^{1/3}$-agree on all subsequent time steps.

Proof. If Alice and Bob $\epsilon$-agree then they are $10\epsilon^{1/3}$-accurate, so in particular $\mathbb{E} \left[ (\mu_{\sigma \tau} - \mu_{\sigma T_s})^2 \right] \leq 10\epsilon^{1/3}$. Note that $\mathbb{E} \left[ (\mu_{\sigma \tau} - \mu_{\sigma T_s})^2 \right]$ is a decreasing function of $s$, since for any $s_1 \leq s_2$ we have

$$\mathbb{E} \left[ (\mu_{\sigma \tau} - \mu_{\sigma T_{s_1}})^2 \right] = \mathbb{E} \left[ (\mu_{\sigma \tau} - \mu_{\sigma T_{s_2}})^2 \right] + \mathbb{E} \left[ (\mu_{\sigma T_{s_2}} - \mu_{\sigma T_{s_1}})^2 \right]$$

by the Pythagorean theorem. Therefore, for any $t' > t$, we have $\mathbb{E} \left[ (\mu_{\sigma \tau} - \mu_{\sigma T_{t'}})^2 \right] \leq 10\epsilon^{1/3}$. Symmetrically, we have $\mathbb{E} \left[ (\mu_{\sigma \tau} - \mu_{S_{t'} \tau})^2 \right] \leq 10\epsilon^{1/3}$. Therefore, $\mathbb{E} \left[ (\mu_{\sigma T_{t'}} - \mu_{S_{t'} \tau})^2 \right] \leq 40\epsilon^{1/3}$, which means that after round $t'$, Alice and Bob $10\epsilon^{1/3}$-agree.

Corollary 8.3.6 stands in contrast to the more general case, in which it is possible that Alice and Bob “nearly agree for the first $t - 1$ time steps, then disagree violently at the $t$-th step” (Aaronson, 2005, §2.2). Thus, while the main purpose of Theorem 8.3.1 is a property about accuracy, an agreement property falls out naturally: under the rectangle substitutes condition, once Alice and Bob are close to agreement, they will remain in relatively close agreement into the future.

### 8.3.3 Graceful decay under closeness to rectangle substitutes

In a sense, the rectangle substitutes condition is quite strong: it requires that the weak substitutes condition be satisfied on every sub-rectangle. One might hope for a result that generalizes Theorem 8.3.1 to information structures that almost-but-not-quite satisfy the rectangle substitutes. Let us formally define a notion of closeness to rectangle substitutes.

**Definition 8.3.7.** An information structure $I = (\Omega, \mathbb{P}, \sigma, \tau, Y)$ satisfies $\delta$-approximate rectangle substitutes if for every partition of $S \times T$ into rectangles,\(^{12}\) the rectangle substitutes condition

---

\(^{12}\)There are partitions into rectangles that cannot arise from a communication protocol. Our results would apply equally if this condition were instead defined for every partition that could arise from a communication protocol, but we state this condition more generally so that it could be applicable in a broader context than the analysis of communication protocols.
holds in expectation over the partition, up to an additive constant of \( \delta \), i.e., if we have

\[
\mathbb{E}_{\sigma, \tau} \left[ (\mu_{\sigma \tau} - \mu_{S_{\sigma, \tau}})^2 \right] \leq \mathbb{E}_{\sigma, \tau} \left[ (\mu_{\sigma \tau}^T - \mu_{S_{\sigma, \tau} \tau})^2 \right] + \delta,
\] (8.6)

where \( S_{\sigma, \tau} \times T_{\sigma, \tau} \) is the rectangle containing \((\sigma, \tau)\).

**Remark 8.3.8.** The \( \delta \)-approximate rectangle substitutes property is a relaxation of the rectangle substitutes property, in the sense that the two are equivalent if \( \delta = 0 \). To see this, first observe that if \( I \) satisfies rectangle substitutes, then it satisfies Equation 8.6 with \( \delta = 0 \) pointwise across all \( S_{\sigma, \tau}, T_{\sigma, \tau} \), and thus in expectation. In the other direction, suppose that \( I \) satisfies 0-approximate rectangle substitutes. Let \( S \subseteq S, T \subseteq T \) and consider the partition of \( I \) into rectangles that contains \( S \times T \) and, separately, every other signal pair \((\sigma, \tau)\) in its own rectangle. For this partition, Equation 8.6 reduces precisely to Equation 8.2 (the rectangle substitutes condition for \( S \) and \( T \)).

Theorem 8.3.1 generalizes to approximate rectangle substitutes as follows.

**Theorem 8.3.9.** Let \( I = (\Omega, \mathcal{F}, \sigma, \tau, Y) \) be an information structure that satisfies \( \delta \)-approximate rectangle substitutes. For any communication protocol that causes Alice and Bob to \( \varepsilon \)-agree on \( I \), Alice and Bob are \((10\varepsilon^{1/3} + \delta)\)-accurate after the protocol terminates.

**Proof.** We first observe that Lemma 8.3.2 can be modified as follows.

**Lemma 8.3.10.** Let \( I = (\Omega, \mathcal{F}, \sigma, \tau, Y) \) be an information structure that satisfies \( \delta \)-approximate rectangle substitutes. Let \( \varepsilon = \mathbb{E} \left[ (\mu_{\sigma} - \mu_{\tau})^2 \right] \). Then

\[
\mathbb{E} \left[ (\mu_{\sigma \tau} - \mu_{\tau})^2 \right] \leq 6\varepsilon^{1/3} + \delta.
\]

The proof of Lemma 8.3.10 is exactly the same as that of Lemma 8.3.2, except that Equation 8.3 includes an additive \( \delta \) term on the left-hand side:

\[
\mathbb{E} \left[ (\mu_{\sigma} - \mu_{S_{(k(\sigma))}})^2 \right] + \delta \geq \mathbb{E} \left[ (\mu_{\sigma \tau} - \mu_{S_{(k(\tau)) \tau}})^2 \right].
\]
This modified inequality follows immediately from the $\delta$-approximate rectangle substitutes condition, noting that one partition of $S \times T$ into rectangles is $\{S_1 \times T, \ldots, S_N \times T\}$. The extra $\delta$ term produces the $\delta$ term in the lemma statement.

To prove the theorem, let $S$ be the set of possible signals of Alice at the end of the protocol which are consistent with the protocol transcript, and define $T$ likewise for Bob. Let $\delta_{ST}$ be the minimum $\delta$ such that $S \times T$ satisfies $\delta$-approximate rectangle substitutes. Note that $E_{S,T} [\delta_{ST}] \leq \delta$: otherwise, by taking the union over the worst-case partitions for each $S, T$ we would exhibit a partition of $S \times T$ into rectangles that would violate the $\delta$-approximate rectangle substitutes property. Therefore we have

$$
E \left[ (\mu_{\sigma T} - \mu_{ST})^2 \right] = E_{S,T} \left[ E \left[ (\mu_{\sigma T} - \mu_{ST})^2 \mid S, T \right] \right] \\
\leq E_{S,T} \left[ 6 \left( E \left[ (\mu_{\sigma T} - \mu_{ST})^2 \mid S, T \right] \right)^{1/3} + \delta_{ST} \right] \\
\leq 6E_{S,T} \left[ E \left[ (\mu_{\sigma T} - \mu_{ST})^2 \mid S, T \right] \right]^{1/3} + \delta \\
= 6E \left[ (\mu_{\sigma T} - \mu_{ST})^2 \right]^{1/3} + \delta = 6(4\epsilon)^{1/3} + \delta \leq 10\epsilon^{1/3} + \delta.
$$

As in the proof of Theorem 8.3.1, the second step follows by applying Lemma 8.3.2 to the information structure $I$ restricted to $S \times T$.

8.4 Results for other divergence measures

Squared distance is a compelling error measure because it elicits the mean. That is, if you wish to estimate a random variable $Y$ and will be penalized according to the squared distance between $Y$ and your estimate, the strategy that minimizes your expected penalty is to report the expected value of $Y$ (conditional on the information you have). This is in contrast to e.g. absolute distance as an error measure, which would instead elicit the median of your distribution. The class of error measures that elicit the mean is precisely the class of Bregman divergences, which we introduced in Section 2.1.3. (See Proposition 2.1.8.)
In this section, our main result is a generalization of Theorem 8.3.1 to (almost) arbitrary Bregman divergences (see e.g. Theorem 8.4.11). Additionally, we provide a generalization of Aaronson’s discretized protocol to arbitrary Bregman divergences (Theorem 8.4.8).

8.4.1 Preliminaries on Bregman divergences

Recall from Section 2.1.3 the definition of Bregman divergence:

**Definition 2.1.7.** For some \( n \geq 1 \), let \( \mathcal{D} \) be a convex subset of \( \mathbb{R}^n \), and let \( G : \mathcal{D} \rightarrow \mathbb{R} \) be a differentiable, convex function. Given \( x, y \in \mathcal{D} \), the Bregman divergence from \( y \) to \( x \) with respect to \( G \) is defined as

\[
D_G(y \parallel x) := G(y) - G(x) - \langle y - x, \nabla G(x) \rangle.
\] (2.2)

In this chapter, we are dealing with scalar quantities, so \( n = 1 \) and in particular \( \mathcal{D} = [0, 1] \). So for us, the Bregman divergence from \( y \) to \( x \) is

\[
D_G(y \parallel x) := G(y) - G(x) - (y - x)G'(x).
\]

Recall that the Bregman divergence with respect to \( G(x) = x^2 \) is precisely the squared distance. Another common Bregman divergence is the KL divergence, which corresponds to \( G(x) = x \log x + (1 - x) \log(1 - x) \), the negative of the binary entropy function.

We generalize relevant notions such as agreement and accuracy to arbitrary Bregman divergences as follows. In the definitions below, \( G : [0, 1] \rightarrow \mathbb{R} \) is a differentiable, strictly convex function.

**Definition 8.4.1.** Let \( a \) be Alice’s expectation. Alice is \( \epsilon \)-accurate if \( \mathbb{E} [D_G(\mu_{\sigma_T} \parallel a)] \leq \epsilon \), and likewise for Bob.\(^{13}\)

We now define \( \epsilon \)-agreement, and to do so we first define the Jensen-Bregman divergence.

\(^{13}\)We discuss our choice of the order of these two arguments (i.e. why we do not instead consider the expectation of \( D_G(a \parallel \mu_{\sigma_T}) \)) in Appendix D.4.
**Definition 8.4.2.** For $a, b \in [0, 1]$, the Jensen-Bregman divergence between $a$ and $b$ with respect to $G$ is

$$JB_G(a, b) := \frac{1}{2} \left( D_G \left( a \parallel \frac{a + b}{2} \right) + D_G \left( b \parallel \frac{a + b}{2} \right) \right) = \frac{G(a) + G(b)}{2} - G \left( \frac{a + b}{2} \right).$$

The validity of the second equality can be easily derived from the definition of Bregman divergence. Note that the Jensen-Bregman divergence, unlike the Bregman divergence, is symmetric in its arguments. The Jensen-Bregman divergence is a lower bound on the average Bregman divergence from Alice and Bob to any other point (see Proposition D.3.1 (i)).

**Definition 8.4.3.** Let $a$ and $b$ be Alice’s and Bob’s expectations, respectively. Alice and Bob $\epsilon$-agree with respect to $G$ if $JB_G(a, b) \leq \epsilon$.

In Appendix D.4 we discuss alternative definitions of agreement and accuracy. The upshot of this discussion is that our definition of agreement is the weakest reasonable one, and our definition of accuracy is the strongest reasonable one. This means that the main result of this section – that under a wide class of Bregman divergence, agreement implies accuracy – is quite powerful: it starts with a weak premise and proves a strong conclusion.

**Definition 8.4.4.** Given an information structure $I$, a communication protocol causes Alice and Bob to $\epsilon$-agree on $I$ with respect to $G$ if Alice and Bob $\epsilon$-agree with respect to $G$ at the end of the protocol. A communication protocol is an $\epsilon$-agreement protocol with respect to $G$ if the protocol causes Alice and Bob to $\epsilon$-agree with respect to $G$ on every information structure.

We also generalize the notion of rectangle substitutes to this domain, following Chen and Waggoner (2016), which explored notions of substitutes for arbitrary Bregman divergences.

**Definition 8.4.5.** Let $G : [0, 1] \to \mathbb{R}$ be a differentiable, strictly convex function. An information structure $I = (\Omega, \mathcal{F}, \sigma, \tau, Y)$ satisfies rectangle substitutes with respect to $G$ if for every $S \subseteq S, T \subseteq T$, we have

$$\mathbb{E} \left[ D_G(Y \parallel \mu_{ST}) \mid S, T \right] - \mathbb{E} \left[ D_G(Y \parallel \mu_{\sigma T}) \mid S, T \right] \leq \mathbb{E} \left[ D_G(Y \parallel \mu_{ST}) \mid S, T \right] - \mathbb{E} \left[ D_G(Y \parallel \mu_{\sigma T}) \mid S, T \right].$$
Chen and Waggoner (2016) explore the notion of weak substitutes with respect to arbitrary convex functions $G$ as well; just as before, $I$ is said to satisfy the weak substitutes condition if the above inequality holds for $S = S$ and $T = T$. The authors additionally explore in detail the weak substitutes condition with respect to negative entropy, i.e. for $D_G$ equal to the KL divergence. They show that if Alice and Bob have independent signals conditioned on $Y$, then the information structure satisfies weak substitutes with respect to this $G$. In fact, any such information structure also satisfies rectangle substitutes, because an information structure with conditionally independent signals retains the conditional independence when restricted to any sub-rectangle. The rectangle substitutes condition thus covers the specific case of conditionally independent signals under which Kong and Schoenebeck (2023) prove their accuracy result. On the other hand, the greater generality of our setting necessitates a different proof strategy.

Recall from Section 2.3.3 that the Pythagorean theorem generalizes to arbitrary Bregman divergences:

**Proposition 2.3.12.** (Pythagorean theorem for Bregman divergence). In a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, let $A : \Omega \rightarrow \mathbb{R}^n$ be a random variable, let $B = \mathbb{E}[A \mid \mathcal{H}]$ where $\mathcal{H}$ is a sub-sigma-algebra, and let $C : \Omega \rightarrow \mathbb{R}^n$ be a random variable that is defined on $\mathcal{H}$. Let $G$ be a differentiable, convex function defined on a convex set that includes all values taken on by $A$, $B$, and $C$. Then

\[
\mathbb{E}[D_G(A \parallel C)] = \mathbb{E}[D_G(A \parallel B)] + \mathbb{E}[D_G(B \parallel C)].
\]

(In this chapter we are specifically interested in real-valued random variables, i.e. we take $n = 1$.) Just as we did with squared error, this general Pythagorean theorem allows us to rewrite the rectangle substitutes condition for Bregman divergences.

**Remark 8.4.6.** An information structure $I$ satisfies rectangle substitutes with respect to $G$ if and only if for all $S \subseteq \mathcal{S}, T \subseteq \mathcal{T}$ we have

\[
\mathbb{E}[D_G(\mu_{\sigma_T} \parallel \mu_{S_T}) \mid S, T] \leq \mathbb{E}[D_G(\mu_{\sigma_T} \parallel \mu_{S_T}) \mid S, T]. \tag{8.7}
\]
Given the interpretation of Bregman divergences as measures of error, we can interpret the left side as Bob’s expected error in predicting the truth while the right side is Charlie’s expected error when predicting Alice’s expectation (with Charlie as defined in Section 8.2.2). Both sides measure a prediction error due to not having Alice’s signal, but from different starting points.

### 8.4.2 Generalizing the discretized protocol

In Section 8.4.4, we will show that under some weak conditions, protocols that cause Alice and Bob to agree with respect to $G$ also cause Alice and Bob to be accurate with respect to $G$. However, this raises an interesting question: are there protocols that cause Alice and Bob to agree with respect to $G$? In particular, we are interested in natural expectation-sharing protocols. Aaronson’s discretized protocol is specific to $G(x) = x^2$, and it is not immediately obvious how to generalize it. We present the following generalization.

**Definition 8.4.7.** Let $G$ be a differentiable, strictly convex function, and let $M := \max_x G(x) − \min_x G(x)$. Choose $\epsilon > 0$. In the discretized protocol with respect to $G$ with parameter $\epsilon$, on her turn (at time $t$), Alice sends “medium” if $D_G(\mu_{\tau_{t-1}} \parallel \mu_{S_{t-1}T_{t-1}}) < \frac{\epsilon}{2}$, and otherwise either “low” or “high”, depending on whether $\mu_{\tau_{t-1}}$ is smaller or larger (respectively) than $\mu_{S_{t}T_{t}}$. Bob acts analogously on his turn. At the start of the protocol, Alice and Bob use the information structure to independently compute the time $t_{\text{end}} \leq \frac{24M(4M+\epsilon)}{\epsilon^2}$ that minimizes $\mathbb{E} \left[ D_G(\mu_{\tau_{t_{\text{end}}}} \parallel \mu_{S_{t_{\text{end}}}}) \right]$. The protocol ends at this time.

**Theorem 8.4.8.** The discretized protocol with respect to $G$ with parameter $\epsilon$ is an $\epsilon$-agreement protocol with respect to $G$ that requires $O \left( \frac{M(M+\epsilon)}{\epsilon^2} \right)$ bits of communication.

Our proof draws inspiration from Aaronson’s proof of the discretized protocol, but has significant differences. The key idea is to keep track of the monovariant $\mathbb{E} \left[ D_G(Y \parallel \mu_{S_{t}}) \right]$. This is Charlie’s expected error (as measured by the Bregman divergence from the correct answer $Y$) after time step $t$ – recall that Charlie is our name for a third-party observer of the protocol. Note that this quantity is at most $M$ and at least 0. Hence, if we show that the quantity decreases by at least
some value $\beta$ every time Alice and Bob do not $\epsilon$-agree, then we will have shown that Alice and Bob must $\epsilon$-agree within $\frac{\beta}{M}$ time steps. We defer the proof to Appendix D.3.

8.4.3 Approximate triangle inequality

Our accuracy results in Section 8.4.4 will hold for a class of Jensen-Bregman divergences that satisfy an approximate version of the triangle inequality. Specifically, we will require $J_B G$ to satisfy the following $c$-approximate triangle inequality for some $c > 0$.

Definition 8.4.9. Given a differentiable, strictly convex function $G : [0, 1] \rightarrow \mathbb{R}$ and a positive number $c$, we say that $J_B G(\cdot, \cdot)$ satisfies the $c$-approximate triangle inequality if for all $a, b, x \in [0, 1]$ we have

$$J_B G(a, x) + J_B G(x, b) \geq c J_B G(a, b).$$

It is possible to construct functions $G$ such that there is no positive $c$ for which $J_B G$ satisfies the $c$-approximate triangle inequality. However, $J_B G$ satisfies the $c$-approximate triangle inequality for some positive $c$ for essentially all natural choices of $G$.

Proposition 8.4.10. Let $G : [0, 1] \rightarrow \mathbb{R}$ be a differentiable, strictly convex function.

(i) If $\sqrt{J_B G(\cdot, \cdot)}$ satisfies the triangle inequality, then $J_B G$ satisfies the $\frac{1}{2}$-approximate triangle inequality.

(ii) If $G(x) = x^2$ (i.e. $D_G$ is squared distance) or if $G(x) = x \log x + (1 - x) \log(1 - x)$ (i.e. $D_G$ is KL divergence), then $\sqrt{J_B G}$ satisfies the triangle inequality (and so $J_B G$ satisfies the $\frac{1}{2}$-approximate triangle inequality).

Proof. Regarding Fact (i), suppose that $\sqrt{J_B G}$ satisfies the triangle inequality. Then for all $a, b, x$ we have $\sqrt{J_B G(a, x)} + \sqrt{J_B G(x, b)} \geq \sqrt{J_B G(a, b)}$. Squaring both sides and observing that $J_B G(a, x) + J_B G(x, b) \geq 2 \sqrt{J_B G(a, x)J_B G(x, b)}$ completes the proof.

Fact (ii) is trivial for $G(x) = x^2$, since $\sqrt{J_B G}$ is the absolute distance metric (times a constant factor). As for $G(x) = x \log x + (1 - x) \log(1 - x)$, see Endres and Schindelin (2003). \qed
The question of which convex functions $G$ have the property that $\sqrt{JB_G}$ satisfies the triangle inequality has been explored in previous work (Acharyya et al., 2013; Chen et al., 2008).

8.4.4 Generalized agreement implies generalized accuracy

In all of the results in this subsection, we consider the following setting: $G$ is a differentiable convex function; $c$ is a positive real number such that $JB_G$ satisfies the $c$-approximate triangle inequality; and $I = (\Omega, \mathbb{P}, \sigma, \tau, Y)$ is an information structure that satisfies rectangle substitutes with respect to $G$.

We prove generalizations of Theorem 8.3.1, showing that under the rectangle substitutes condition, if a protocol ends with Alice and Bob in approximate agreement, then Alice and Bob are approximately accurate.

**Theorem 8.4.11.** Assume that $G$ is symmetric about the line $x = \frac{1}{2}$. For any communication protocol that causes Alice and Bob to $\epsilon$-agree on $I$, and for any $\beta \geq \frac{2}{c}\epsilon$, Alice and Bob are

$$
\left(\frac{8}{c^2}\beta + 16 \left( G(0) - G\left(\frac{\epsilon}{\beta}\right)^{1/(1-\log_2 c)}\right)\right) - \text{accurate}
$$

with respect to $G$ after the protocol terminates.

This result is not our most general, as it assumes that $G$ is symmetric, but this assumption likely holds for most use cases. To apply the result optimally, one must first optimize $\beta$ as a function of $G$. For example, setting $\beta = \epsilon^r/(r+1-\log_2 c)$ (with $r$ defined below) gives us the following corollary:

**Corollary 8.4.12.** Assume that $G(0) - G(x), G(1) - G(1-x) \leq O(x^r)$. For any communication protocol that causes Alice and Bob to $\epsilon$-agree on $I$, Alice and Bob are $O\left(\epsilon^{r/(r+1-\log_2 c)}\right)$-accurate after the protocol terminates, where the constant hidden by $O(\cdot)$ depends on $G$.

**Remark 8.4.13.** Concretely, if $G'$ is bounded then we can choose $r = 1$, in which case our bound simplifies to $O\left(\epsilon^{1/(2-\log_2 c)}\right)$. If instead we assume that $c = \frac{1}{2}$ (as is the case if $\sqrt{JB_G(\cdot, \cdot)}$ is a

\[14\]Corollary 8.4.12 as stated (without the symmetry assumption) is actually a corollary of Theorem 8.4.15.
metric), then the bound is $O\left(e^{r/(r+2)}\right)$. If both of these are true, as is the case for $G(x) = x^2$, then the bound is $O(e^{1/3})$, which recovers our result in Theorem 8.3.1.

For $G$ equal to the negative of Shannon entropy (i.e. the $G$ for which $D_G$ is KL divergence), setting $\beta = e^{1/3}(\log 1/e)^{2/3}$ in Theorem 8.4.11 gives us the following corollary.

**Corollary 8.4.14.** If $G(x) = x \log x + (1 - x) \log(1 - x)$, then for any communication protocol that causes Alice and Bob to $\epsilon$-agree on $I$, Alice and Bob are $O(e^{1/3}(\log 1/e)^{2/3})$-accurate after the protocol terminates.

Theorem 8.4.11 follows from our most general result about agreement implying accuracy:

**Theorem 8.4.15.** Let $\tilde{G}(x) := \max_{a,b} |a - b| \leq x (G(a) - G(b))$ be the maximum possible difference in $G$-values of two points that differ by at most $x$, and let $\tilde{G}^*(x)$ be the concave envelope of $\tilde{G}$, i.e.

$$\tilde{G}^*(x) := \max_{0 \leq a,b,w \leq 1 : w a + (1 - w) b = x} w \tilde{G}(a) + (1 - w) \tilde{G}(b).$$

For any communication protocol that causes Alice and Bob to $\epsilon$-agree on $I$, and for any $\beta > 0$, Alice and Bob are

$$\left(\frac{8}{c^2} \beta + 16 \tilde{G}^* \left(\left(\frac{\epsilon}{\beta}\right)^{1/(1 - \log_2 c)}\right)\right)$$

accurate after the protocol terminates.

**Proof.** To prove Theorem 8.4.15, it suffices to prove the following lemma.

**Lemma 8.4.16.** Let $G$ be a differentiable convex function on $[0, 1]$ and $c \in (0, 1)$ be such that $J_B G$ satisfies the $c$-approximate triangle inequality. Let $I = (\Omega, \mathbb{P}, \sigma, \tau, Y)$ be an information structure that satisfies rectangle substitutes with respect to $G$. Let $\epsilon = \mathbb{E} [J_B G(\mu_{\tau} \parallel \mu_{\tau})]$. Then for any $\beta > 0$, we have

$$\mathbb{E} [D_G(\mu_{\tau\tau} \parallel \mu_{\tau})] \leq \frac{8}{c^2} \beta + 16 \tilde{G}^* \left(\left(\frac{\epsilon}{\beta}\right)^{1/(1 - \log_2 c)}\right).$$

Let us first prove Theorem 8.4.15 assuming Lemma 8.4.16 is true.
Consider any protocol that causes Alice and Bob to ε-agree on $I$. Let $S$ be the set of possible signals of Alice at the end of the protocol which are consistent with the protocol transcript, and define $T$ likewise for Bob.

Let $\varepsilon_{ST} = \mathbb{E} [\text{JB}_G(\mu_{\sigma T}, \mu_{S\tau}) \mid S, T]$. Note that

$$\mathbb{E}_{S,T}[\varepsilon_{ST}] = \mathbb{E}_{S,T} [\mathbb{E} [\text{JB}_G(\mu_{\sigma T}, \mu_{S\tau}) \mid S, T]] = \mathbb{E} [\text{JB}_G(\mu_{\sigma T}, \mu_{S\tau})] \leq \varepsilon.$$ 

Therefore, for any $\beta > 0$ we have

$$\mathbb{E} [D_G(\mu_{\sigma \tau} \parallel \mu_{S\tau})] \leq \mathbb{E}_{S,T} \left[ \frac{8}{c^2\beta} + 16\tilde{G}^* \left( \frac{\varepsilon_{ST}}{\beta} \right)^{1/(1-\log_2 c)} \right]$$

$$\leq \frac{8}{c^2\beta} + 16\tilde{G}^* \left( \mathbb{E}_{S,T} \left[ \frac{\varepsilon_{ST}}{\beta} \right] \right)^{1/(1-\log_2 c)}$$

$$\leq \frac{8}{c^2\beta} + 16\tilde{G}^* \left( \frac{\varepsilon_{ST}}{\beta} \right)^{1/(1-\log_2 c)}$$

$$\leq \frac{8}{c^2\beta} + 16\tilde{G}^* \left( \frac{\varepsilon_{ST}}{\beta} \right)^{1/(1-\log_2 c)}.$$

In the first step, we apply Lemma 8.4.16 to the information structure $I$ restricted to $S \times T$ – that is, to $I' = (\Omega', \mathcal{P}', S, T, Y)$, where $\Omega' = \{\omega \in \Omega : \sigma \in S, \tau \in T\}$ and $\mathcal{P}'[\omega] = \mathcal{P}[\omega \mid \sigma \in S, \tau \in T]$. The next two steps follow by the convexity of $\tilde{G}^*$ and $x^{1/(1-\log_2 c)}$, respectively.

The basic outline of the proof of Lemma 8.4.16 is similar to that of Lemma 8.3.2. Once again, we partition $[0, 1]$ into $N$ intervals. Analogously to Equation 8.4, and with $S^{(k(\sigma))}$ defined analogously, we find that

$$\mathbb{E} [D_G(\mu_{\sigma \tau} \parallel \mu_{T})] \leq \mathbb{E} [D_G(\mu_{\sigma} \parallel \mu_{S^{(k(\sigma))}})] + \mathbb{E} [D_G(\mu_{S^{(k(\sigma))} \tau} \parallel \mu_{T})].$$

As before, we wish to upper bound each summand. However, the fact that the Bregman divergence is now arbitrary introduces complications. First, it is no longer the case that we can directly relate the length of an interval to the Bregman divergence between its endpoints. Second, we
consider functions $G$ that become infinitely steep near 0 and 1 (such as the negative of Shannon entropy), which makes matters more challenging. This means that we need to be more careful when partitioning $[0, 1]$ into $N$ intervals: see Algorithm D.3.3 for our new approach. Additionally, bounding the second summand involves reasoning carefully about the behavior of the function $G$, which is responsible for the introduction of $\tilde{G}^*$ into the lemma statement. We defer the full proof of Lemma 8.4.16 to Appendix D.3.

8.5 Connections to markets

In this work, we established a natural condition on information structures, rectangle substitutes, under which any agreement protocol results in accurate beliefs. Expectation-sharing protocols, where Alice and Bob take turns stating their current expected value (or discretizations thereof) are a particularly natural class of agreement protocols.

Expectation-sharing protocols have close connections to financial markets. In markets, the actions of traders reveal partial information about their believed value for an asset, i.e., their expectation. Specifically, a trader’s decision about whether to buy or sell, and how much, can be viewed as revealing a discretization of this expectation. In many theoretical models of markets (e.g. Ostrovsky (2012)) traders eventually reach agreement. The intuition behind this phenomenon is that a trader who disagrees with the price leaves money on the table by refusing to trade. Our work thus provides a lens into a well-studied question:\footnote{This is related to the efficient market hypothesis, the problem of when market prices reflect all available information, which traces back at least to Fama (1970) and Hayek (1945). Modern models of financial markets are often based on Kyle (1985); see e.g. Ostrovsky (2012) and references therein for further information.} when are market prices accurate? Our results can be viewed as generalizing and conceptually supporting the result presented in Kong and Schoenebeck (2023), under which convergence in a popular prediction market design implies full information revelation in the prices.

An important caveat, however, is that traders behave strategically, and may not disclose their true expected value. For example, a trader may choose to withhold information until a later point when doing so would be more profitable. Therefore, to interpret the actions of traders as revealing
discretized versions of their expected value, one first has to understand the Bayes-Nash equilibria of the market. Chen and Waggoner (2016) study conditions under which traders are incentivized to reveal all of their information on their first trading opportunity. They call a market equilibrium *all-rush* if every trader is incentivized to reveal their information immediately. Their main result, roughly speaking, is that there is an all-rush equilibrium if and only if the information structure satisfies *strong substitutes* – another strengthening of their weak substitutes condition. This result is specific to settings in which traders have the option to reveal all of their information on their turn – a setting that would be considered trivial from the standpoint of communication theory.

An exciting question for further study is therefore: under what information structure conditions and market settings is it a Bayes-Nash equilibrium to follow an agreement protocol that leads to accurate beliefs? In other words, what conditions give not only that agreement implies accuracy, but also that the market incentivizes participants to *follow* the protocol? Together with Chen and Waggoner (2016), our work suggests that certain substitutes-like conditions could suffice.
Chapter 9: Deductive circuit estimation

This chapter presents work done at the Alignment Research Center, in collaboration with Paul Christiano, Jacob Hilton, Václav Rozhoň, and Mark Xu. It builds on the ideas of “Formalizing the Presumption of Independence” (Christiano et al., 2022). The chapter assumes basic familiarity with theoretical computer science. The idea of forecast aggregation in the context of experts holding partial information – especially as in Section 2.4 – may be useful for context.

Summary: In this chapter, we turn our attention to estimating formal mathematical expressions – such as the acceptance probability of a boolean circuit – using deductive arguments. Much as it is possible to be uncertain about whether it will rain next week due to missing information or bounded computational resources, one can also be uncertain about the fraction of inputs on which a boolean circuit will output 1. While there are simple inductive methods for estimating this number, such as by running the circuit on a random sample of inputs, in this chapter we are interested in deductive estimation. In other words, we are interested in designing an algorithm that takes as input formal arguments and observations about the structure of a circuit, and uses those observations in order to estimate the circuit’s acceptance probability. Deductive estimates have the advantage that they may give insight about the reasons why a circuit tends to accept certain classes of inputs.

Deductive estimates of mathematical quantities abound in fields such as number theory, theoretical computer science, and discrete mathematics, but there is little understanding of the rules that govern such estimation. Much as there is a formal definition of a mathematical proof and an algorithm that checks proofs for validity, there may be a formal definition of a valid deductive argument and an algorithm that uses such arguments to estimate mathematical quantities.

In this chapter, we search for such a formalization, focusing our attention on deductively esti-
mating the acceptance probabilities of boolean circuits. We define two properties – \textit{linearity} and \textit{respect for proofs} – that a deductive estimation algorithm ought to satisfy, and then provide an efficient algorithm that satisfies those properties. We then show a negative result: that no efficient algorithm can satisfy linearity, respect for proofs, and another property that we call \textit{0-1 boundedness}, assuming that $P \neq PP$. We discuss additional desirable properties for a deductive estimation algorithm and then conclude with a discussion of potential applications to detecting anomalous neural network behavior.

More so than any previous chapter, this one shows a work in progress. It raises more questions than it answers, and some of the raised questions are not formally stated. However, the question of whether deductive estimation can be formalized seems like a fundamental one, and further progress in this direction may have important applications for building advanced AI systems safely.

9.1 Introduction

Suppose we wish to estimate the acceptance probability $p(C)$ of a boolean circuit $C : \{0, 1\}^n \rightarrow \{0, 1\}$ on a uniformly random input – that is, the fraction of inputs $x$ for which $C(x) = 1$. We can contrast two different types of approaches to such an estimation problem: \textit{inductive} approaches and \textit{deductive} approaches.

By an \textit{inductive} approach, we mean an approach based on observing the input-output behavior of $C$ on some inputs. For example, one inductive approach is to sample inputs to $C$ uniformly at random and observe the fraction of inputs that $C$ accepts (outputs $1$ on).\footnote{We consider pseudorandom sampling to also be inductive. The use of randomness is not required for an argument to be considered inductive.} Such an estimation procedure is very effective, at least in terms of estimating $p(C)$ with a small additive error. However, it does not provide any insight about \textit{why} $C$’s acceptance probability is what it is.

By a \textit{deductive} approach, we mean an approach that uses structural observations about $C$ in order to estimate $p(C)$. We call such structural observations \textit{deductive arguments}.\footnote{We use the term “deductive argument” instead of the term “heuristic argument” (as used in e.g. Christiano et al. (2022)) to stress the difference with inductive arguments (which we do not consider in this chapter) and to emphasize...} While an
inductive approaches would generally treat a circuit as a black box and estimate its acceptance probability using input-output behavior, deductive arguments treat circuits as white boxes. Let us illustrate with a few examples.

**Example 9.1.1.** Suppose that $C$ takes as input a triple $(a, b, c)$ of positive integers, computes $\max(a, b)$ and $\max(b, c)$, and accepts if they are equal. A deductive argument about $p(C)$ might point out that if $b$ is the largest of the three integers, then $\max(a, b) = b = \max(b, c)$ and so $C$ will accept, and that this happens with probability roughly $\frac{1}{3}$.

**Example 9.1.2.** Suppose that $C(x)$ computes SHA-256($x$) (the output of SHA-256 is a 256-bit string) and accepts if the first 128 bits (interpreted as an integer) are larger than the last 128 bits. One can make a deductive argument about $p(C)$ by making repeated use of the presumption of independence (Christiano et al., 2022). In particular, the SHA-256 circuit consists of components (such as the majority circuit, or addition modulo $2^{32}$) that produce uniformly random outputs on independent, uniformly random inputs. Thus, a deductive argument that repeatedly presumes that the inputs to each component are independent concludes that the output of SHA-256 consists of independent, uniformly random bits. It would then follow that the probability that the first 128 bits of the output are larger than the last 128 bits is $\frac{1}{2}$.

**Example 9.1.3.** Suppose that $C$ is a particular CNF with $k$ clauses of three literals each. A deductive argument about $p(C)$ might point out that $C$ is a CNF with $k$ clauses of three literals each, and that on average, CNFs with this structure accept $\left(\frac{2}{3}\right)^k$ fraction of inputs. A more sophisticated argument might point out particular structural patterns in $C$ (e.g. that the variable $x_1$ appears with positive sign in every clause) and reason about the average acceptance probability of CNFs with that structure.

**Example 9.1.4.** Suppose that $C$ takes as input an integer $k$ between $e^{100}$ and $e^{101}$ and accepts if $k$ and $k + 2$ are both prime. A deductive argument about $p(C)$ might point out that the density of primes in this range is roughly 1%, so if we presume that the event that $k$ is prime and the event that a proof is a special type of deductive argument.
$k + 2$ is prime are independent, then we get an estimate of 0.01%. A more sophisticated argument might take this one step further by pointing out that if $k$ is prime, then $k$ is odd, so $k + 2$ is odd, which makes $k + 2$ more likely to be prime (by a factor of 2), suggesting an estimate of 0.02%. A yet more sophisticated argument might point out that additionally, if $k$ is prime, then $k$ is not divisible by 3, which makes $k + 2$ more likely to be divisible by 3, which reduces the chance that $k + 2$ is prime, and would make similar arguments for divisibility by 5, 7, and so on.

In a sense, deductive arguments generalize proofs. A formal proof about the value of $p(C)$ is a type of deductive argument, but in many cases it is infeasible to prove even basic facts about a circuit’s acceptance probability. In Example 9.1.2, for instance, there is likely no short proof that $p(C)$ is between $\frac{1}{3}$ and $\frac{2}{5}$, but there is a very simple deductive argument. Deductive arguments let us draw conclusions based on circuits’ structural properties in many cases where proofs do not.

Inductive approaches to circuit estimation hold some obvious advantages over deductive ones. First, they are fast and straightforward. Second, they have strong accuracy guarantees: by sampling random inputs, one can efficiently estimate $p(C)$ with a small additive error with high probability. By comparison, a deductive estimate can be incorrect (for example, the naïve argument that gave the 0.01% estimate in Example 9.1.4 was substantially wrong). So why bother with deductive estimates?

One answer is that deductive estimation may be helpful in situations where sampling-based approaches will fail. For example, suppose we wish to estimate the acceptance probability of a CNF $C$ that has a low acceptance probability, with a small multiplicative error. Sampling inputs will likely fail to find any satisfying assignments. On the other hand, one could hope to be able to reason deductively about $C$’s acceptance probability if given deductive arguments based on $C$’s structure.

More importantly, however, deductive arguments can give us a *mechanistic understanding* of a circuit and provide insight about why the circuit has a certain acceptance probability. Such understanding allows us to answer questions like: does $C$ accept inputs $x$ and $y$ “for the same reason” or
“for different reasons”? In Example 9.1.1, \( C \) might accept an input because \( b \) is larger than both \( a \) and \( c \), or because \( a = c \). Deductive arguments can allow us to draw such distinctions; inductive arguments do not. Gaining such a mechanistic understanding is an important motivation for exploring deductive circuit estimation.

The purpose of this line of work is to create a deductive estimation algorithm. A deductive estimation algorithm takes as input a boolean circuit,\(^3\) together with a list of deductive arguments about the circuit, and outputs an estimate of the circuit’s acceptance probability based on the provided arguments. Creating such an algorithm requires surmounting several challenges. First, it is necessary to specify a formal language in which deductive arguments may be stated; deductive arguments given as input to the estimation algorithm will be written in this formal language. Second, desiderata for the algorithm should be defined: what properties should a deductive estimation algorithm satisfy? Finally, the third challenge is to create an algorithm that satisfies those properties.

An analogy to formal proof verification may be instructive. The central task of proof verification is the (solved) problem of finding a verification algorithm that, given a mathematical statement and an alleged proof of the statement, verifies the proof. In our setting, the circuit (whose acceptance probability we want to estimate) is analogous to the mathematical statement, and the list of deductive arguments is analogous to the proof. Table 9.1 describes the analogy in more detail.

In Section 9.2, we begin with a discussion of deductive circuit estimation via linear regression. That is, we will be interested in producing a linear estimator for a class of circuits in terms of a list of features. We illustrate this approach with an example of CNF estimation. The purpose of this section is to provide an example of what deductive arguments for circuit estimation might look

\(^3\)We are also interested in the more general problem of estimating the expected outputs of arithmetic circuits and of computer programs more generally, but in this chapter we focus on boolean circuits. Boolean circuits are a special case that may capture the core difficulties of the more general problem. Further, boolean circuits have a canonical choice for the distribution of inputs (uniformly random bit strings), and some of our discussion (e.g. our definition of linearity in Section 9.3) is simplified in the case of boolean circuits.
Table 9.1: We are interested in developing a deductive estimation algorithm for boolean circuits. There are similarities between this task and the (solved) task of developing an algorithm for verifying formal proofs of mathematical statements. This table illustrates the analogy. Importantly, the purpose of a deductive estimation algorithm is to incorporate the deductive arguments that it has been given as input, rather than to generate its own arguments. The output of a deductive estimation algorithm is only as sophisticated as the arguments that it has been given.

In Section 9.3, we examine the problem of deductive circuit estimation in more generality, asking the question: what properties should a deductive estimation algorithm satisfy? We introduce two properties – which we call linearity and respect for proofs – and describe an algorithm that satisfies both properties. Then, in Section 9.4, we discuss additional properties. We introduce a basic property called 0-1 boundedness and show that it is impossible to satisfy in addition to linearity and respect for proofs, unless $P = PP$. We discuss two other properties – iterated estimation and pulling out known factors – that it would be desirable to satisfy, and leave open the question of whether they can be satisfied.

Finally, in Section 9.5, we conclude with a discussion of potential applications to detecting anomalous behavior in neural networks.

### 9.1.1 Related work

The work most closely related to ours is Christiano et al. (2022), which posed the question of whether it is possible to formalize the process of deductively estimating the value of a mathematical expression, such as the expectation of a random variable. The authors posited the presumption of independence as an underlying principle of deductive estimation: two random variables should
be presumed independent until an argument to the contrary is presented. Thus, a deductive estimation algorithm might start by assuming that all sub-expressions are independent, and then update its estimate after incorporating knowledge about various dependencies. This chapter attacks the problem of deductive estimation from a different perspective, but is a continuation of that work.

There has not been much other prior work on formalizing deductive estimation. Perhaps the closest is Tao (2012), which also posits that two quantities should be presumed independent unless there is a good reason to the contrary, and uses this principle to give a heuristic justification for the ABC conjecture from number theory. Heuristic arguments for mathematical claims are common in fields such as number theory, theoretical computer science, and discrete mathematics. For example, Erdős and Ulam (1971) give a heuristic argument for the truth of Fermat’s last theorem; Mézard and Zecchina (2002) use heuristic methods to analyze random $k$-SAT instances; and Cornu and Hilhorst (2019) heuristically estimate the frequencies of certain patterns in Conway’s game of life. However, little prior work attempts to formalize the rules governing such arguments.

Barak (2015, 2016) are also closely related to our work. Barak asks whether it is possible to construct an estimation algorithm for a given quantity that appears reasonable to a broad class of observers. By contrast, in our framing, “observers” are deductive arguments that are given as input. Our perspective is in some ways less ambitious and in some ways more: on the one hand, we want our estimate to appear reasonable to an observer (i.e. incorporate a deductive argument) only if the observer is explicitly given as input; on the other hand, this may allow us to consider a much wider range of possible observers.

More recently, Gowers (2023) posited a no-coincidence principle: “If an apparently outrageous coincidence happens in mathematics, then there is a reason for it.” In the language of this chapter, we might instead write: “If a circuit exhibits a surprising behavior, then there is a short deductive argument that explains the behavior.” (An example of a surprising behavior might be that the acceptance probability of a circuit is much larger than a naïve estimate would suggest.\footnote{Informally speaking: for a circuit that always outputs 1, there should be a deductive argument that makes it seem plausible that the circuit always outputs 1. On the other hand, for most circuits that do not always output 1, no such deductive argument should exist. See Christiano et al. (2022, §C) for further discussion.})
There is a significant literature on the problem of circuit estimation. Inductive estimates are common, often under the name *approximate model counting*. Beyond straightforward uniform sampling, there are more algorithms based on more sophisticated methods such as Markov chain Monte Carlo (Wei & Selman, 2005). If given access to an NP oracle, hashing-based approaches can give strong guarantees (Yang & Meel, 2023). See Chakraborty et al. (2021) for a survey of this area. There has also been work on derandomization-based deterministic algorithms for approximate DNF counting (Gopalan et al., 2013).

There has also been extensive research into heuristic methods for the satisfiability problem (see e.g. Gu et al. (1996)). This work is similar in spirit to ours, but our goal is different. Rather than finding particular heuristic algorithms and validating them on particular instances, we are seeking a unified theoretical framework for making estimates based on a very large class of arguments.

### 9.2 Circuit estimation via linear regression

Often, families of circuits are easier to reason about than the individual circuits they contain. For example, it may be difficult to compute the acceptance probability of a given 3CNF with \(k\) clauses, but it is easy to compute the average acceptance probability of *all* 3CNFs with \(k\) clauses. For this reason, estimating the acceptance probability of a circuit \(C\) is made easier by instead estimating the acceptance probabilities of *all* circuits in a family that contains \(C\).

Let us denote the average acceptance probability of a (finite) family of boolean circuits \(\mathcal{C}\) as \(p(\mathcal{C})\). That is, \(p(\mathcal{C}) := \mathbb{E}_{C \sim \mathcal{C}} [p(C)]\). Consider a circuit \(C : \{0, 1\}^n \rightarrow \{0, 1\}\) for which \(p(C)\) is difficult to compute. Let \(\mathcal{C}\) be a family of circuits containing \(C\). In a sense, \(p(\mathcal{C})\) is a reasonable (if uninformative) estimate for \(p(C)\), because this estimate is correct on average over \(\mathcal{C}\).

We can ask for more informative estimates: ones that use more information about a circuit than just the fact that it belongs to \(\mathcal{C}\). In general, we can hope to estimate the acceptance probabilities of the circuits in \(\mathcal{C}\) by using features of those circuits.

To be more precise, suppose that we have \(m\) features: efficiently computable functions \(\varphi_1, \ldots, \varphi_m\):
\(C \rightarrow \mathbb{R}\). Suppose further that we can compute the average value of each feature on \(C\) (that is, \(\mathbb{E}_{C \sim \mathcal{C}} [\varphi_i(C)]\) for all \(i \in m\), covariances between the features (that is, \(\text{Cov}_{C \sim \mathcal{C}} (\varphi_i(C), \varphi_j(C))\) for \(i, j \in m\)), and the covariance of each feature with the acceptance probability (that is, \(\text{Cov}_{C \sim \mathcal{C}} (p(C), \varphi_i(C))\) for all \(i \in m\)). Then the best linear estimator of \(p(C)\) in terms of the features, as measured by average squared error over \(\mathcal{C}\), is given by the least squares regression formula:

\[
\hat{p}(C) = p(\mathcal{C}) + \left(\begin{array}{c}
\text{Cov}_{C' \sim \mathcal{C}} (p(C'), \varphi_1(C')) \\
\vdots \\
\text{Cov}_{C' \sim \mathcal{C}} (p(C'), \varphi_m(C'))
\end{array}\right) \Sigma_\varphi^+ \left(\begin{array}{c}
\varphi_1(C) - \mathbb{E}_{C' \sim \mathcal{C}} [\varphi_1(C')] \\
\vdots \\
\varphi_m(C) - \mathbb{E}_{C' \sim \mathcal{C}} [\varphi_m(C')]
\end{array}\right),
\]

where \(\Sigma_\varphi\) denotes the covariance matrix of the features and \(\Sigma_\varphi^+\) is its Moore-Penrose pseudoinverse.

### 9.2.1 Example: CNF estimation

In this section, we work with a family \(\mathcal{C}\) consisting of \(2^{3k}\) 3CNFs on \(n\) variables with \(k\) clauses: specifically, these CNFs will be identical except for the sign of each literal. More formally, for every \(j \in [k]\) and \(\ell \in [3]\), fix an index \(i_{j,\ell}\) in \([n]\). Then, for every \(b \in \{0, 1\}^{k \times 3}\), we define

\[
C_b(x) := (x_{i_{1,1} = b_{1,1}} \lor x_{i_{1,2} = b_{1,2}} \lor x_{i_{1,3} = b_{1,3}}) \land \cdots \land (x_{i_{k,1} = b_{k,1}} \lor x_{i_{k,2} = b_{k,2}} \lor x_{i_{k,3} = b_{k,3}}).
\]

That is, the indices \(i_{j,\ell}\) specify which variable appears at each position in the CNF, and \(b_{j,\ell}\) specifies the sign of the \(\ell\)-th literal in clause \(j\). Then, for a particular choice of indices \(i_{1,1}, \ldots, i_{k,3}\) (which is implicit in the notation), \(\mathcal{C}\) is the set of CNFs \(C_b\) for all \(b \in \{0, 1\}^{k \times 3}\).

**Claim 9.2.1.** \(p(\mathcal{C}) = (7/8)^k\).

**Proof.** The quantity \(p(\mathcal{C})\) is equal to the fraction of all choices of \((x, b)\) such that \(C_b(x) = 1\). Choose any given \(x\); we ask: if we select a random \(b\), what is the probability that \(C_b(x) = 1\)? For each of the \(k\) clauses, there is a \(7/8\) probability that the clause will be satisfied by the chosen \(b\), and these probabilities are independent. \(\square\)
Thus, we could estimate \( p(C_b) \) as \( (7/8)^k \) for all \( b \). As we have discussed, this is a reasonable estimate, but not a very informative one. To refine this estimate, we consider \( n \) features – one for each variable – which we call \( \varphi_1, \ldots, \varphi_n \):

### Definition 9.2.2

Given a circuit \( C \in \mathcal{C} \), let \( \mathcal{C}_i(C) \) (for \( i \in [n] \)) be the set of circuits in \( \mathcal{C} \) that agree define with \( C \) on the signs of all instances of the variable \( x_i \). Define \( \varphi_i(C) := p(\mathcal{C}_i(C)) \), i.e. the average acceptance probability of these circuits.

### Example 9.2.3

Suppose that \( C(x) = (x_1 = 1 \lor x_2 = 1 \lor x_3 = 1) \land (x_1 = 0 \lor x_2 = 1 \lor x_4 = 1) \).

There are \( 2^4 = 16 \) circuits in \( \mathcal{C}_1(C) \): specifically all circuits of the form

\[
(x_1 = 1 \lor x_2 = b_{1,2} \lor x_3 = b_{1,3}) \land (x_1 = 0 \lor x_2 = b_{2,2} \lor x_4 = b_{2,3}),
\]

for some choice of bits \( b_{1,2}, b_{1,3}, b_{2,2}, b_{2,3} \). We can compute \( \varphi_1(C) \) by conditioning on the value of \( x_1 \):

\[
\varphi_1(C) = \frac{1}{2} \Pr_{x,b} [x_2 = b_{1,2} \lor x_3 = b_{1,3}] + \frac{1}{2} \Pr_{x,b} [x_2 = b_{2,2} \lor x_4 = b_{2,3}] = \frac{3}{4}.
\]

We can think of each \( \varphi_i(C) \) as a more informative estimate of \( p(C) \) than \( p(\mathcal{C}) \) is: it is still correct on average, but now uses more information about \( C \).\(^5\) An analogy to forecast aggregation may be instructive. Suppose that a circuit \( C \) is drawn at random from \( \mathcal{C} \), and that there are \( n \) experts: Expert \( i \) knows the signs of all instances of \( x_i \) and no other signs. Then \( p(\mathcal{C}) \) is the prior for the value of \( p(C) \), and \( \varphi_i(C) \) is Expert \( i \)’s estimate of the value of \( p(C) \).

Note that \( \varphi_i(C) \) is easy to compute. In particular, we have

\[
\varphi_i(C) = \Pr_{x \sim \{0,1\}^n, C' \sim \mathcal{C}_i(C)} [C'(x) = 1] = \frac{1}{2} \left( \Pr_{x_1 = 1, C' \sim \mathcal{C}_i(C)} [C'(x) = 1] + \Pr_{x_1 = 0, C' \sim \mathcal{C}_i(C)} [C'(x) = 1] \right).
\]

\(^5\)In general, features do not need to be reasonable estimates for acceptance probability – they just happen to be in this case.
Now, let $C[x_i = 1]$ be the CNF obtained by plugging $x_i = 1$ into $C$ and simplifying. Note that $\Pr_{x \sim \mathcal{E}_i} [C'(x) = 1]$ is equal to the average acceptance probability of all circuits that are identical to $C[x_i = 1]$ except for the signs of the literals – and finding this average probability is easy (see Claim 9.2.1). Computing $\Pr_{x \sim \mathcal{E}_i} [C'(x) = 1]$ is similarly easy.

Thus, we can obtain $n$ estimates $\varphi_1(C), \ldots, \varphi_n(C)$, each of which is more informative than $p(\mathcal{E})$. We can combine them into a single estimate of $p(C)$ with linear regression, as per Equation 9.1. In particular:

- $\mathbb{E}_{C \sim \mathcal{E}} [\varphi_i(C')] = p(\mathcal{E})$ for all $i$.
- The $n$ features are uncorrelated. This is because the $\mathcal{E}_i(C)$ depends only on the signs of the variable $x_i$ in $C$, and these bits are independent for each value of $i$. Thus, $\Sigma_\varphi$ is a diagonal matrix containing the feature variances.
- $\varphi_i(C')$ is a calibrated estimate of $p(C')$, meaning that $\mathbb{E}_{C \sim \mathcal{E}} [p(C') | \varphi_i(C')] = \varphi_i(C')$.

This means that $\text{Cov}_{C \sim \mathcal{E}} (p(C'), \varphi_i(C')) = \text{Var}_{C \sim \mathcal{E}} (\varphi_i(C'))$. Therefore, we have

$$
\begin{pmatrix}
\text{Cov}_{C \sim \mathcal{E}} (p(C'), \varphi_1(C')) \\
\vdots \\
\text{Cov}_{C \sim \mathcal{E}} (p(C'), \varphi_m(C'))
\end{pmatrix} \Sigma_\varphi^{-1} =
\begin{pmatrix}
1 \\
\vdots \\
1
\end{pmatrix}.
$$

Putting these facts together tells us that the linear regression estimator for $p(C)$ is given by

$$
\hat{p}(C) = p(\mathcal{E}) + \sum_{i=1}^{n} (\varphi_i(C) - p(\mathcal{E})) = \sum_{i=1}^{n} \varphi_i(C) - (n - 1) p(\mathcal{E}).
$$

(9.2)

Intuitively, the estimator $\hat{p}$ treats $p(\mathcal{E})$ as a prior and treats the quantities $\varphi_i(C) - p(\mathcal{E})$ as independent updates from the prior, which it combines additively.\(^6\)

It turns out that $\hat{p}(C)$ is a substantial improvement upon the prior $p(\mathcal{E})$. Figure 9.1 plots $\hat{p}(C)$ versus $p(C)$ for ten thousand randomly generated 3CNFs on ten variables with five clauses. In this

---

\(^6\)We can think of this additive estimate in the context of Section 2.4. There, we saw that if experts report estimates $Y_1, \ldots, Y_m$ for a mean-zero quantity $Y$, then the best linear estimate of $Y$ in terms of $Y_1, \ldots, Y_m$ is given by $\text{diag}(\Sigma) \Sigma^{-1} Y$, where $Y = (Y_1, \ldots, Y_m)$ and $\Sigma$ is the covariance matrix of the experts’ estimates. If we consider $Y = (p(C) - p(\mathcal{E}))$, then $Y_i = \varphi_i(C) - p(\mathcal{E})$, and $\Sigma$ is a diagonal matrix (since the $Y_i$’s are independent). This perspective also yields the estimate given by Equation 9.2.
sample, $\hat{p}(C)$ explains 78% of the variance in $p(C)$.

By adding more features, it would be possible to further refine the estimate, so that it explains even more variance in $p(C)$. For example, one could add features based on the signs of pairs of variables, or based on other structural properties of the CNFs in $\mathcal{C}$.

Figure 9.1: Ten thousand 3CNFs on $n = 10$ variables with $k = 5$ clauses were randomly generated. This scatter plot shows each CNF’s estimated acceptance probability $\hat{p}(C)$ on the $x$-axis and its true acceptance probability $p(C)$ on the $y$-axis. In this sample, $\hat{p}(C)$ explains 78% of the variance in $p(C)$ (that is, $r^2 \approx 0.78$).

9.2.2 Challenges for linear regression

One challenge faced by the linear regression estimator given in Equation 9.1 is that it may output estimates $\hat{p}(C)$ that are not in $[0, 1]$. This makes sense, given that we sought an estimator that minimizes average squared error over $\mathcal{C}$: if $p(C) = 0.1$, the estimate $\hat{p}(C) = -0.2$ is no worse than the estimate $\hat{p}(C) = 0.4$. On the other hand, such errors mean that the linear regression estimator may output estimates that are obviously incorrect. These errors can be corrected for individual circuits by introducing additional features. However, in Section 9.4.1, we show that
requiring all estimates to lie between 0 and 1 must come at the expense of a different desirable property.

A more significant challenge to using Equation 9.1 for deductive circuit estimation is that one may want to use features whose average values cannot be easily computed, or whose covariances with \( p(C) \) or with each other cannot be easily computed. In such situations, it makes sense to deductively estimate those quantities. How can we do this?

We can think of each feature \( \varphi_i \) as being computed by some arithmetic circuit.\(^7\) Equation 9.1 applies just as well for estimating expected values of arithmetic circuits, so we can hope to use linear regression in turn to estimate the quantities we need. However, this is difficult for several reasons. First, it is unclear what family of circuits \( \varphi_i \) should be thought of as belonging to (that is, over what family of circuits to do the recursive linear regression step). Second, if the features are almost as complex as \( C \) itself, then estimating the covariances between them means estimating quantities that are more complex than \( C \). Third, this recursive estimation scheme would need a base case: what if no features are supplied? That is, we want a reasonable “default guess” for the expressions in Equation 9.1. These challenges are priorities for future work, because fully solving them might yield a much more generic deductive estimation algorithm.

### 9.3 Linearity and respect for proofs

So far we have discussed how a deductive estimation algorithm might behave when given certain kinds of arguments (features for linear regression). In this section, we take a different perspective: what properties should a deductive estimation algorithm satisfy? We define two properties, which we call linearity and respect for proofs, and give a deductive estimation algorithm that satisfies both properties.

Let us introduce notation (mostly following Christiano et al. (2022)). We will use the letter \( G \) to denote a deductive estimation algorithm. \( G \) will take two inputs: first, a circuit \( C \) (whose accep-

\(^7\)The input to \( \varphi_i \) is ordinarily a circuit \( C \); when thinking of \( \varphi_i \) as a circuit, we can think of its input as a string of bits that describes \( C \).
tance probability it needs to estimate), and second, a set of deductive arguments \( \{ \pi_1, \ldots, \pi_m \} \). We will write \( G(C \mid \pi_1, \ldots, \pi_m) \) to denote the output of \( G \) on these inputs.\(^8\) For convenience, we will use \( \Pi \) to denote a set of arguments and write \( G(C \mid \Pi) \).

How do features for linear regression, as in Section 9.2, translate into arguments? Since we have not specified a formal language for deductive arguments, we cannot give an exact answer; however, the arguments should specify \( C \) and the features \( \varphi_i(C) \). The arguments should also verify the values of the quantities in Equation 9.1 (namely, \( p(C), \text{Cov}(p(C'), \varphi_i(C')), \Sigma_{\varphi}, \varphi_i(C), \) and \( \mathbb{E}[\varphi_i(C')] \)), so that \( G \) can compute \( \hat{p}(C) \). For example, there should be a proof of the value of \( p(C) \) among the arguments given to \( G \).\(^9\)

We now define the following notation for circuit substitutions, and then introduce the aforementioned linearity and respect for proofs properties.

**Definition 9.3.1.** Let \( C : \{0, 1\}^n \rightarrow \{0, 1\} \) be a circuit with input variables \( x_1, \ldots, x_n \). For \( b \in \{0, 1\} \), we define \( C[x_i = b] \) to be the circuit on \( n - 1 \) variables \( x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n \) obtained by setting \( x_i = b \) in \( C \). More generally, given a partial assignment \( A \) to some \( \ell \) of the \( n \) variables, we define \( C[A] \) to be the circuit on the remaining \( n - \ell \) variables that is obtained from \( C \) by setting the variables in \( \ell \) to their corresponding values in \( A \). A particular input \( x \) can be thought of as a full assignment; thus, \( C[x] \) is a circuit that takes no inputs and outputs \( C(x) \).

**Definition 9.3.2.** A deductive estimation algorithm \( G \) satisfies linearity if for all \( n, C : \{0, 1\}^n \rightarrow \{0, 1\}, \ i \in [n], \) and \( \Pi \), we have

\[
G(C \mid \Pi) = \frac{1}{2} (G(C[x_i = 0] \mid \Pi) + G(C[x_i = 1] \mid \Pi)).
\]

In other words: a basic fact about circuits is that \( p(C) = \frac{1}{2}(p(C[x_i = 0]) + p(C[x_i = 1])) \), and \( G \)'s estimates should respect this fact. Note that linearity entails the following quite strong

\(^8\)We choose to use the \( | \) symbol because \( G \) needs to estimate \( p(C) \) given – i.e. in light of – the arguments \( \pi_1, \ldots, \pi_m \). As we will discuss later, we may want \( G \) to behave akin to a conditional expectation.

\(^9\)More generally, \( G \) should be able to accept deductive arguments about the values of these quantities, instead of exact proofs, as briefly discussed in Section 9.2.2.
property: if \( C \) is a circuit on \( n \) inputs, then

\[
G(C \mid \Pi) = \frac{1}{2^n} \sum_{x \sim \{0,1\}^n} G(C[x] | \Pi).
\]

We note, however, that \( G(C[x] | \Pi) \) need not equal \( C(x) \). Although calculating \( C(x) \) is easy, \( G \) does not necessarily do so: \( G \)'s task is not to estimate a circuit as well as it can, but rather to fully incorporate the arguments in \( \Pi \) into its estimate. Unless \( \Pi \) contains an explicit computation of \( C(x) \), \( G(C[x] | \Pi) \) may be different from \( C(x) \).

We now introduce the “respect for proofs” property. Recall from the introduction that deductive arguments can be thought of as generalizing proofs. The respect for proofs property essentially states that proofs constitute valid deductive arguments that \( G \) must respect.

**Definition 9.3.3.** A deductive estimation algorithm \( G \) satisfies respect for proofs if there is a polynomial-time algorithm that, given boolean circuits \( C_1, \ldots, C_k \) and a proof that \( \sum_{i=1}^{k} a_i p(C_i) \leq b \) for some rational constants \( a_1, \ldots, a_k, b \), outputs an argument \( \pi \) such that for every set of arguments \( \Pi \) containing \( \pi \), it is the case that \( \sum_{i=1}^{k} a_i G(C_i \mid \Pi) \leq b \).

In other words, if \( G \) is given a proof of a linear constraint on the acceptance probabilities of some circuits, then \( G \)'s estimates of the acceptance probabilities must also obey that linear constraint. For example:

- For a circuit \( C \), a proof that \( p(C) \leq \frac{1}{2} \) can be translated into an argument \( \pi \) such that \( G(C \mid \pi) \leq \frac{1}{2} \). In fact, Definition 9.3.3 gives a stronger guarantee: for all further arguments \( \pi_1, \pi_2, \ldots, \pi_m \), we must still have \( G(C \mid \pi, \pi_1, \ldots, \pi_m) \leq \frac{1}{2} \).
- Similarly, a proof that \( p(C) \geq \frac{1}{2} \) can be translated into an argument \( \pi' \). If \( G \) is given both \( \pi \) (as in the previous bullet point) and \( \pi' \), then \( G(C \mid \pi, \pi', \pi_1, \ldots, \pi_m) \) must equal \( \frac{1}{2} \) for any further arguments \( \pi_1, \ldots, \pi_m \).
- For circuits \( C_1 \) and \( C_2 \), the property \( p(C_1) = p(C_2) \) can likewise be stated as an intersection of two linear constraints: \( p(C_1) - p(C_2) \leq 0 \) and \( p(C_2) - p(C_1) \leq 0 \). A proof of this
property can be translated into arguments $\pi$ and $\pi'$ such that $G(C_1 \mid \pi, \pi', \pi_1, \ldots, \pi_m) = G(C_2 \mid \pi, \pi', \pi_1, \ldots, \pi_m)$ for all $\pi_1, \ldots, \pi_m$.

Definition 9.3.3 requires the constraints to be linear. Why not allow all constraints? Suppose we have a proof that $p(C)$ is either less than $\frac{1}{3}$ or greater than $\frac{2}{3}$. What should $G$’s estimate of $p(C)$ be, in light of such a proof? Answers between $\frac{1}{3}$ and $\frac{2}{3}$ are quite reasonable: while it is known that $p(C) < \frac{1}{3}$ or $p(C) > \frac{2}{3}$, it is not known which of these is true, and so it would be reasonable for $G$ to “split the difference” and say $\frac{1}{2}$.

Meanwhile, if it is known that $\sum_i a_i p(C_i) \leq b$, then any reasonable tuple of estimates for $p(C_1), \ldots, p(C_k)$ ought to satisfy this constraint as well. One formal justification for this is that for any probability distribution over tuples $(p(C_1), \ldots, p(C_k))$ that satisfy the constraint, the mean of the distribution will also satisfy the constraint. In fact, this property holds for all convex constraints, not just linear ones, and it would be reasonable to state a stronger version of Definition 9.3.3 that requires $G$ to respect arbitrary convex constraints. The results of this section generalize to this stronger definition, although with some caveats; see Remark 9.3.10 for further discussion.

Essentially, linearity is a basic consistency condition on $G$’s estimates for different circuits derived from the same base circuit – that is, circuits that are derived from the same circuit via different variable substitutions (i.e. $C[A_1]$ and $C[A_2]$ for partial assignments $A_1$ and $A_2$). By contrast, respect for proofs is a more general consistency condition on $G$’s estimates, but it only applies if $G$ is supplied a proof. For example, if $C_1$ and $C_2$ provably implement the same function, then $G$ does not need to output the same estimate for $C_1$ and $C_2$ by default – but if it is given a proof that $p(C_1) = p(C_2)$, then its estimates must respect that proof. Taken together, linearity and respect for proofs are fairly powerful.\(^{10}\)

On the other hand, the respect for proofs property is importantly constrained, because it cannot force $G$ to obey a universal quantification of proofs. For example, a proof that $p(C[x]) \geq 0$ for

\(^{10}\)For example, given a network of circuits that relate to each other by elementary equivalence transformations (such as distributing AND over OR) and variable substitution, one can turn this network into a set of arguments $\Pi$ that forces $G$ to give consistent estimates for all circuits in the network.
all \( x \) cannot be turned into an argument \( \pi \) such that \( G(C[x] \mid \pi) \geq 0 \) for all \( x \). The respect-for-proofs property merely guarantees that for all \( x \), the proof that \( p(C[x]) \geq 0 \) can be turned into an argument \( \pi_x \) such that \( G(C[x] \mid \pi_x) \geq 0 \).\(^{11}\) In Section 9.4.1, we will show that one cannot hope to satisfy a stronger version of respect for proofs that allows for universal quantification (in conjunction with linearity).

### 9.3.1 An efficient algorithm that satisfies linearity and respect for proofs

In this section, we give a polynomial-time deductive estimation algorithm that satisfies linearity and respect for proofs. Before we do so, we clarify some conventions and definitions that will become relevant:

- Circuits will always have labeled inputs and gates. For example, if \( C' \) is an exact gate-for-gate copy of \( C \), we still consider \( C \) and \( C' \) to be different circuits. A linear constraint involving \( C \) does not force \( G \)'s estimate of \( p(C') \) to satisfy the same constraint. However, a proof that \( p(C) = p(C') \) will force \( G \) to output the same estimate for \( C \) as for \( C' \).
- We can formally define a circuit as a DAG of labeled inputs and gates with one sink (output) – we will call this DAG the base circuit – together with a (possibly empty) partial assignment to the input variables. For example, if \( C' = C[x_1 = 0] \) and \( C'' = C'[x_2 = 1] \), then we formally define \( C'' \) to be the base circuit \( C \) together with the partial assignment \( (x_1 = 0, x_2 = 1) \). That is, \( C'' := C[x_1 = 0, x_2 = 1] \). If a circuit \( C \) is defined as \( \tilde{C}[A] \), then we call \( \tilde{C} \) the base circuit of \( C \). Every circuit has exactly one base circuit.
- The size of a partial assignment \( A \), denoted \( |A| \), is the number of variables that it defines. The measure of a partial assignment \( A \), denoted \( \mu(A) \), is defined as \( 2^{-|A|} \) (this is the fraction of full assignments that satisfy \( A \)). Given partial assignments \( A_1 \) and \( A_2 \), we define \( \mu(A_1 \cup A_2) \) to be the fraction of full assignments satisfying both \( A_1 \) and \( A_2 \); we have \( \mu(A_1 \cup A_2) \leq \mu(A_1), \mu(A_2) \), and in particular \( \mu(A_1 \cup A_2) = 0 \) if \( A_1 \) and \( A_2 \) assign opposite values to the

\(^{11}\)In theory, \( G \) can be given an argument \( \pi_x \) for every \( x \), but in that case the runtime of \( G \) may be exponential in the size of \( C \), even if \( G \) is a polynomial-time algorithm.
same variable.

Our deductive estimation algorithm, which we call \( G_{\text{lin}} \), is defined in Algorithm 9.3.4. We call this algorithm \( G_{\text{lin}} \) because it creates a linear model of a base circuit’s behavior on individual inputs, and then computes acceptance probabilities based on that model. (See the discussion immediately after the proof of Theorem 9.3.6 for details of this interpretation.) The algorithm’s implementation details are somewhat complex, but the basic idea behind it is fairly simple. Before proving the algorithm’s correctness and efficiency, we describe the intuition behind it.

**Algorithm 9.3.4: Deductive estimation algorithm \( G_{\text{lin}} \)**

**Input**: Boolean circuit \( C \), set of deductive arguments \( \Pi \)

**Output**: \( G_{\text{lin}}(C \mid \Pi) \), a deductive estimate of \( p(C) \)

1. Initialize an empty list \( \ell \) of linear constraints. For each \( \pi \in \Pi \), check whether \( \pi \) is a valid proof of a rational linear constraint \( \sum_{i=1}^{k} a_i p(C_i) \leq b \). If so, add the constraint to \( \ell \).
2. For every circuit \( C' \) that is the base circuit of some circuit that appears in some constraint in \( \ell \): let \((A_1, \ldots, A_m)\) be the tuple of all partial assignments \( A \) such that \( C'[A] \) appears in some constraint in \( \ell \). Let \( M \) be the \( m \times m \) matrix defined by \( M_{i,j} = \mu(A_i \cup A_j) \). Compute vectors \( v_1, \ldots, v_k \) that span the orthogonal complement of the column space of \( M \). For each \( i \in [k] \), add the constraint \( \langle v_i, (\mu(A_1)p(C'[A_1]), \ldots, \mu(A_m)p(C'[A_m])) \rangle = 0 \) to \( \ell \).
3. Use the ellipsoid method to find a point \( \hat{p} \) satisfying all constraints in \( \ell \) – that is, an estimate \( \hat{p}(C') \) of \( p(C') \) for every \( C' \) appearing in at least one constraint in \( \ell \) (not just base circuits), such that \( \hat{p} \) simultaneously satisfies all of the constraints in \( \ell \).
4. Let \( C = \tilde{C}[A] \), where \( \tilde{C} \) is \( C \)’s base circuit. Let \((A_1, \ldots, A_m)\) be the (possibly empty) tuple of all partial assignments \( A' \) such that \( \tilde{C}[A'] \) appears in some constraint in \( \ell \). Let \( M \) be the \( m \times m \) matrix defined by \( M_{i,j} = \mu(A_i \cup A_j) \). Let \( \beta \) be the lowest-norm solution to the equation \( M\beta = (\mu(A_1)\hat{p}(\tilde{C}[A_1]), \ldots, \mu(A_m)\hat{p}(\tilde{C}[A_m])) \). (Below we show that this equation has a solution.) Return \( \sum_{j=1}^{m} \frac{\mu(A_i \cup A_j)}{\mu(A_j)} \beta_j \).

In Step 1 and 2, the algorithm creates a list \( \ell \) of constraints on its estimates, which it will then work to satisfy. In Step 1, it searches through all linear constraints that have been proven in \( \Pi \) and adds them to \( \ell \). Its job will be to satisfy all of those constraints. Then, in Step 2, the algorithm adds some additional constraints: ones that it can infer must hold simply from how the acceptance probabilities of circuits derived from the same base circuit must relate to each other. For example, we know that \( p(C[x_1 = 0]) + p(C[x_1 = 1]) = 2p(C) \), and that \( p(C[x_1 = 0, x_2 = 0]) + p(C[x_1 = 0, x_2 = 1]) + 2p(C[x_1 = 1]) = 4p(C) \), and so on. Adding these constraints to \( \ell \) is necessary to
make sure that the linear system in Step 4 is consistent.

Then, in Step 3, the algorithm solves all of the constraints that it found in \( \Pi \) or subsequently appended to \( \ell \). The result is a point \( \hat{p} \) whose coordinates correspond to estimated acceptance probabilities of all circuits that appeared in some constraint in \( \ell \). For a circuit \( C' \), we use \( \hat{\rho}(C') \) to denote the coordinate of \( \hat{p} \) corresponding to the circuit \( C' \).\(^\text{12}\)

Finally, in Step 4, the algorithm looks at the circuit \( C \) that it was provided as input – it is only at this final step that the algorithm’s behavior depends on the input circuit \( C \). Let \( C = \tilde{C}[A] \), where \( \tilde{C} \) is \( C \)'s base circuit.\(^\text{13}\) While the algorithm will only output an estimate of \( C \), it must satisfy linearity, which means that its estimate for \( p(C) \) must be consistent with the counterfactual estimates that it would have given, had it been asked to estimate the acceptance probability of a different circuit with the same base circuit \( \tilde{C} \). In order to do this, it finds coefficients \( \beta_i \) – one for every circuit \( \tilde{C}[A_i] \) that appeared in \( \ell \) – with the following property: if for every \( x, \tilde{C}(x) \) were equal to the sum of \( \beta_j \) over all \( j \) such that \( x \) is consistent with \( A_j \), then for every \( A_i \), the average output of \( \tilde{C}[A_i] \) would be equal to \( \hat{\rho}(\tilde{C}[A_i]) \). That is, the algorithm creates a linear model of \( \tilde{C}' \)'s behavior on individual inputs that is consistent with the estimates given by \( \hat{p} \). From there, it is straightforward to compute the average output of \( C \) based on this linear model – and this is the number that the algorithm outputs.

**Example 9.3.5.** Recall from Section 9.2.1 our class of \( 2^{3^k} \) 3CNFs on \( n \) variables with \( k \) clauses, parameterized by \( b \in \{0, 1\}^{k \times 3} \):

\[
\tilde{C}_b(x) = (x_{i_1,1} = b_{1,1} \lor x_{i_1,2} = b_{1,2} \lor x_{i_1,3} = b_{1,3}) \land \cdots \land (x_{i_k,1} = b_{k,1} \lor x_{i_k,2} = b_{k,2} \lor x_{i_k,3} = b_{k,3}).
\]

(Here, the indices \( i_j, \ell \in [n] \) are fixed.) Define \( \tilde{C}(x, b) := C_b(x) \).

Now, suppose we want to estimate \( p(C) \), where \( C = C_z \) for some particular \( z \). We can recover the estimate from Section 9.2.1 as a simple example of Algorithm 9.3.4. Note that \( C = \tilde{C}[b =

\text{We use the ellipsoid algorithm for concreteness, but any worst-case polynomial-time algorithm for solving linear programs would work.}

\text{Note that } C \text{ did not necessarily appear as a constraint in } \ell, \text{ so } \hat{\rho}(C) \text{ may not be defined.}
For \( i \in [n] \), define \( A_i \) to be the partial assignment that sets all bits in \( b \) corresponding to appearances of the variable \( x_i \) in the CNF formula the same way that they are set in \( z \). Note that \( C = \tilde{C}[A_1, \ldots, A_n] \). For convenience, let \( \varphi_i := p(\tilde{C}[A_i]) \).

Let \( \pi \) be the proof that \( p(\tilde{C}) = (7/8)^k \) (\( \pi \) might look essentially like the proof of Claim 9.2.1, though in a formal language).\(^{14}\) Let \( \pi_1, \ldots, \pi_n \) be proofs of the values of \( \varphi_i \) (these proofs are straightforward, as discussed in Section 9.2.1). Let \( \Pi = \{ \pi, \pi_1, \ldots, \pi_n \} \).

What is \( G_{\text{lin}}(C \mid \Pi) \)? In Step 1, the algorithm adds constraints that enforce the values of \( \hat{p}(\tilde{C}), \hat{p}(\tilde{C}[A_1]), \ldots, \hat{p}(\tilde{C}[A_n]) \). Assuming that every variable appears at least once in the CNF formula, no further constraints are added in Step 2, because the matrix \( M \) defined in that step has full rank. Step 3 is trivial, because the constraints in Step 1 define a polytope consisting of a single point – at this point we have found \( \hat{p} \), and we have \( \hat{p}(\tilde{C}) = p(\tilde{C}) \) and \( \hat{p}(\tilde{C}[A_i]) = \varphi_i \) for \( i \in [n] \).

What happens in Step 4? Call \( \beta_0, \beta_1, \ldots, \beta_n \) the \( \beta \)-values corresponding to \( \tilde{C}, \tilde{C}[A_1], \ldots, \tilde{C}[A_n] \), respectively. Because \( C = \tilde{C}[A_1, \ldots, A_n] \), the algorithm will return \( \beta_0 + \beta_1 + \cdots + \beta_n \). Now, let’s compute \( \beta \). Note that \( \mu(A_i \cup A_j) = \mu(A_i) \mu(A_j) \) for \( i \neq j \). This fact lets us simplify the matrix equation, giving us the following \( n + 1 \) linear equations:

\[
\begin{align*}
\beta_0 + \sum_{j=1}^{n} \mu(A_j) \beta_j &= p(\tilde{C}) \\
\beta_0 + \beta_i + \sum_{j \neq i}^{n} \mu(A_j) \beta_j &= \varphi_i \quad \forall i \in [n].
\end{align*}
\]

Subtracting the first equation from the last \( n \) equations, we find that any solution must have \( (1 - \mu(A_i)) \beta_i = \varphi_i - p(\tilde{C}) \) for all \( i \), and so \( \beta_i = \frac{\varphi_i - p(\tilde{C})}{1 - \mu(A_i)} \). This means that

\[
\beta_0 = p(\tilde{C}) - \sum_{j=1}^{n} \mu(A_j) \beta_j = p(\tilde{C}) - \sum_{j=1}^{n} \mu(A_j) \cdot \frac{\varphi_j - p(\tilde{C})}{1 - \mu(A_j)}.
\]

\(^{14}\)More precisely, we should have two arguments: \( \pi^\leq \), which proves that \( p(\tilde{C}) \leq (7/8)^k \), and \( \pi^\geq \), which proves that \( -p(\tilde{C}) \leq (7/8)^k \), and similarly for \( \pi_1, \ldots, \pi_n \).
Therefore, the algorithm returns

$$\beta_0 + \beta_1 + \cdots + \beta_n = p(\tilde{C}) + \sum_{j=1}^{n} (1 - \mu(A_j)) \cdot \frac{\varphi_j - p(\tilde{C})}{1 - \mu(A_j)} = \sum_{j=1}^{n} \varphi_j - (n - 1)p(\tilde{C}).$$

This is exactly the same as Equation 9.2, our linear regression estimate from Section 9.2.1!

We now prove that Algorithm 9.3.4 is efficient, and that it satisfies linearity and respect for proofs.

**Theorem 9.3.6.** The deductive estimation algorithm $G_{\text{lin}}$, as defined in Algorithm 9.3.4, runs in polynomial time in the length of its input, and satisfies linearity and respect for proofs.

**Proof.** We begin by proving that Algorithm 9.3.4 runs in polynomial time. Step 1 is straightforwardly fast, because checking a proof for validity takes linear time. Step 2 is also fast: first, computing $M$ is straightforward. To compute $v_1, \ldots, v_k$, we first (greedily) find a subset of $M$’s column space that spans the column space (which will have size $m - k$) and use the Gram-Schmidt process to find $m - k$ orthogonal vectors that have the same span. Then we continue the Gram-Schmidt process in order to find $k$ vectors in $\mathbb{R}^m$ that are orthogonal to these $m - k$ orthogonal vectors. These are our $v_1, \ldots, v_k$.

Step 3 is an application of the ellipsoid method, which – given rational linear constraints that define a nonempty region – can find a point in the region in polynomial time (Grötschel et al., 1993). The only detail that we must check is that the intersection of the constraints in $\ell$ is nonempty. We show this by showing that the vector $p$ of true acceptance probabilities satisfies all constraints.

Clearly, $p$ satisfies the constraints that are added to $\ell$ in Step 1 (after all, $\Pi$ contains proofs that $p$ satisfies the constraints). What about the constraints added in Step 2? For $C'$, $(A_1, \ldots, A_m)$, and $M$ as defined in Step 2, we must check that if $v$ is orthogonal to the column space of $M$, then $\langle v, (\mu(A_1)p(C'[A_1]), \ldots, \mu(A_m)p(C'[A_m])) \rangle = 0$. In order to do so, it suffices to show that $(\mu(A_1)p(C'[A_1]), \ldots, \mu(A_m)p(C'[A_m]))$ lies in the column space of $M$. 242
Claim 9.3.7. For $C'$, $(A_1, \ldots, A_m)$, and $M$ as in Step 2, $(\mu(A_1)p(C'[A_1]), \ldots, \mu(A_m)p(C'[A_m]))$ lies in the column space of $M$.

Proof. Let $n$ be the number of inputs to $C'$. Associate to each coordinate of $\mathbb{R}^{2^n}$ a unique element of $\{0, 1\}^n$. Let $\Phi$ be the $2^n \times m$ matrix whose $i$-th column is the indicator vector for the partial assignment $A_i$ — that is, $\Phi_i$ contains a 1 in the coordinate corresponding to $x \in \{0, 1\}^n$ if $x$ is consistent with $A_i$. Let $y \in \mathbb{R}^{2^n}$ be the indicator vector for $C'$ — that is, $y$ contains a 1 in the coordinate corresponding to $x$ if $C'(x) = 1$.

Let $\beta \in \mathbb{R}^m$ be such that $\Phi \beta = \text{proj}_\Phi(y)$ — that is, $\Phi \beta$ is the projection of $y$ onto the column space of $\Phi$. We claim that $M \beta = (\mu(A_1)p(C'[A_1]), \ldots, \mu(A_m)p(C'[A_m]))$. To see this, we first observe that $M = \frac{1}{2^n} \Phi^T \Phi$. This is because $\Phi^T \Phi$ is an $m \times m$ matrix whose $(i, j)$-entry is the number of assignments satisfying both $A_i$ and $A_j$, which is $2^n \mu(A_i \cup A_j)$. Therefore, we have

$$M \beta = \frac{1}{2^n} \Phi^T \Phi \beta = \frac{1}{2^n} \Phi^T \Phi \text{proj}_\Phi(y) = \frac{1}{2^n} \Phi^T y,$$

where the last step follows from the fact that $\Phi^T(y - \text{proj}_\Phi(y)) = 0$, as $y - \text{proj}_\Phi(y)$ is orthogonal to the column space of $\Phi$. Finally, observe that $\Phi^T y$ is a vector in $\mathbb{R}^m$ whose $i$-th coordinate counts the number of inputs $x \in \{0, 1\}^n$ that satisfy $C$ and are consistent with partial assignment $A_i$. This means that $\frac{1}{2^n} \Phi^T y$ is the fraction of inputs with this property, so the $i$-th coordinate of $\frac{1}{2^n} \Phi^T y$ is equal to $\mu(A_i)p(C'[A_i])$. This completes the proof. \hfill \Box

Finally, we consider Step 4. The equation $M \beta = (\mu(A_1)\hat{\rho}(\tilde{C}[A_1]), \ldots, \mu(A_m)\hat{\rho}(\tilde{C}[A_m]))$ has a solution. After all, we enforced this in Step 2! Assuming that $\tilde{C}$ is the base circuit of some constraint in $\ell$,\textsuperscript{15} when we considered $C' = \tilde{C}$ in Step 2, we added linear constraints that enforced that $(\mu(A_1)\hat{\rho}(\tilde{C}[A_1]), \ldots, \mu(A_m)\hat{\rho}(\tilde{C}[A_m]))$ is orthogonal to every vector in the orthogonal complement of the column space of $M$. This means that $(\mu(A_1)\hat{\rho}(\tilde{C}[A_1]), \ldots, \mu(A_m)\hat{\rho}(\tilde{C}[A_m]))$ lies in $M'$s column space.

\textsuperscript{15}And if not, then the empty vector is a solution to the linear system (which has zero equations and zero variables).
Now, since a solution exists, the space of solutions is an affine subspace of \( \mathbb{R}^m \). The lowest-norm solution is just the projection of \( \mathbf{0} \) onto the space, which can be found in polynomial time. Computing the output is also straightforward. This completes the proof that Algorithm 9.3.4 runs in polynomial time.

We now prove that \( G_{\text{lin}} \) satisfies linearity and respect for proofs.

**Claim 9.3.8.** \( G_{\text{lin}} \) satisfies respect for proofs.

*Proof.* We begin by noting the following fact: if \( C \) appears in a constraint in \( \ell \), then \( G_{\text{lin}}(C \mid \Pi) = \hat{p}(C) \). To see this, let \( \tilde{C}, A_1, \ldots, A_m, M, \beta \) be as in Step 4 of the algorithm, and without loss of generality assume that \( C = \tilde{C}[A_1] \). The matrix equation defining \( \beta \) gives us \( m \) linear constraints, the first of which says that

\[
\sum_{j=1}^{m} \mu(A_1 \cup A_j)\beta_j = \mu(A_1)\hat{p}(C),
\]

and so indeed we have

\[
G_{\text{lin}}(C \mid \Pi) = \sum_{j=1}^{m} \frac{\mu(A_1 \cup A_j)}{\mu(A_1)}\beta_j = \hat{p}(C).
\]

Now, to prove that \( G_{\text{lin}} \) satisfies respect for proofs, let us consider boolean circuits \( C_1, \ldots, C_k \) and a proof that \( \sum_{i=1}^{k} a_i p(C_i) \leq b \). The polynomial-time algorithm in Definition 9.3.3 that converts the proof into an argument \( \pi \) is the trivial algorithm that just outputs the proof that it receives as input. Now, consider any \( \Pi \) containing \( \pi \). Let us consider the behavior of Algorithm 9.3.4 if given \( \Pi \) as input.

Note that the vector \( \hat{p} \) computed in Step 3 only depends on \( \Pi \), and not at all on \( C \): after all the algorithm first looks at \( C \) in Step 4. Note also that \( C_1, \ldots, C_k \) all appear in some constraint in \( \ell \) (because the algorithm receives \( \pi \) as part of \( \Pi \)), and so the quantities \( \hat{p}(C_1), \ldots, \hat{p}(C_k) \) are well-defined and do not depend on the input \( C \) given to Algorithm 9.3.4.

Because the constraint \( \sum_{i=1}^{k} a_i p(C_i) \leq b \) appeared in \( \ell \), we have \( \sum_{i=1}^{k} a_i \hat{p}(C_i) \leq b \). Therefore, we have that \( \sum_{i=1}^{k} a_i G_{\text{lin}}(C_i \mid \Pi) = \sum_{i=1}^{k} a_i \hat{p}(C_i) \leq b \), as desired. \( \square \)
Claim 9.3.9. $G_{lin}$ satisfies linearity.

Proof. Let $C$, $\Pi$, and $i$ be as in Definition 9.3.2. We are interested in the algorithm’s outputs when run on $(C, \Pi)$, $(C[x_i = 0], \Pi)$, and $C[x_i = 1], \Pi)$. Let $\tilde{C}$ be the base circuit of $C$, which is also the base circuit of $C[x_i = 0]$ and $C[x_i = 1]$. Observe that $\beta$ as defined in Step 4 is the same in all three cases. This is because the steps before Step 4 only depend on $\Pi$ and not on the input circuit, and $\beta$ depends only $\Pi$ and the base circuit defined in Step 4, which is $\tilde{C}$ in all three cases.

Let $A_1, \ldots, A_m$ be as in Step 4, which again are the same in all three cases. Let $A$ be such that $C = \tilde{C}[A]$; then $C[x_i = 0] = \tilde{C}[A \cup (x_i = 0)]$ and $C[x_i = 1] = \tilde{C}[A \cup (x_i = 1)]$. We want to show that

$$\sum_{j=1}^{m} \frac{\mu(A \cup A_j)}{\mu(A)} \beta_j = \frac{1}{2} \left( \sum_{j=1}^{m} \frac{\mu(A \cup (x_i = 0) \cup A_j)}{\mu(A \cup (x_i = 0))} \beta_j + \sum_{j=1}^{m} \frac{\mu(A \cup (x_i = 1) \cup A_j)}{\mu(A \cup (x_i = 1))} \beta_j \right).$$

We have that $\mu(A \cup (x_i = 0)) = \mu(A \cup (x_i = 1)) = \frac{1}{2} \mu(A)$. We also have that $\mu(A \cup (x_i = 0) \cup A_j) + \mu(A \cup (x_i = 1) \cup A_j) = \mu(A \cup A_j)$. Equation 9.3 follows straightforwardly from these two facts.

We have now proven that Algorithm 9.3.4 runs in polynomial time and that it satisfies linearity and respect for proofs, and so we have completed the proof of Theorem 9.3.6.

Observe that inputs to a circuit $C$ are in correspondence with full assignments to the variables in $C$. That is, we have $C(x) = p(C[x])$, where on the right-hand side we think of $x$ as a partial assignment that happens to be a full assignment. This lets us define $G_{lin}$’s estimate of $C(x)$ on a particular input $x$: namely, it is the quantity $G_{lin}(C[x] \mid \Pi)$. So, what is this quantity? For $A_1, \ldots, A_m, \beta$ as in Step 4 of Algorithm 9.3.4, we have

$$G_{lin}(C[x] \mid \Pi) = \sum_{j=1}^{m} \frac{\mu(x \cup A_j)}{\mu(x)} \beta_j = \sum_{j: x \text{ is consistent with } A_j} \beta_j.$$

This gives us a different perspective on $G_{lin}$: instead of thinking of $G_{lin}(C[x] \mid \Pi)$ as a special
case in which we substitute values for all variables, we can instead think of $G_{\text{lin}}$ as constructing a linear model for $\tilde{C}$’s behavior on individual inputs: namely, the sum of all $\beta_j$ such that the input is consistent with $A_j$. Then, for a given circuit $C = \tilde{C}[A]$, $G_{\text{lin}}(C \mid \Pi)$ is simply the average value of $G_{\text{lin}}(\tilde{C}[x] \mid \Pi)$ over all $x$ that are consistent with $A$.

**Remark 9.3.10.** As discussed earlier, it makes sense to extend the definition of respect for proofs to allow for convex constraints. Can we can extend Algorithm 9.3.4 to deal with convex constraints? The answer is essentially yes, but with a few caveats:

- In order to run the ellipsoid method, we must have access to a separation oracle for the convex constraints. It is easy for $G$ to implement a separation oracle under the following assumptions: first, the convex constraints are expressed as $f(C(x_1), \ldots, C(x_k)) \leq b$ for a convex function $f$. Second, there is a fast algorithm for evaluating $f$ and the gradient of $f$ at a point (or a subgradient, if $f$ is not differentiable). Third, $G$ is given a proof that $f$ is convex, as well as algorithms for evaluating $f$ and its gradient and proofs that the algorithms are fast and correct.

- Even so, for general convex constraints, the ellipsoid method may not be able to find an exact solution to all constraints. However, the ellipsoid method can quickly find a point that satisfies all constraints to within $\epsilon$ tolerance. Thus, we can satisfy “near”-respect-for-proofs, but not necessarily exact respect for proofs.

## 9.4 Desiderata beyond linearity and respect for proofs

Although $G_{\text{lin}}$ satisfies linearity and respect for proofs, its estimates are in some sense arbitrary. The ellipsoid method finds some $\hat{p}$ satisfying the constraints in $\Pi$, but this $\hat{p}$ is by no means canonical.

As a more canonical choice, we could instead find the centroid of the feasible region.\[^{16}\] While

\[^{16}\text{In order to make the centroid well-defined, we can bound the feasible region by adding the constraints } 0 \leq p(C') \leq 1 \text{ for every } C' \text{ appearing in some constraint in } \ell.\]
we cannot efficiently compute the exact centroid, we can compute it approximately by sampling from the feasible region, which can be done efficiently (see e.g. Chen, Dwivedi, et al. (2018)).

However, choosing the centroid of the feasible region has some undesirable properties, even if it can be computed exactly. Let $G_{\text{cent}}$ be the modified version of $G_{\text{lin}}$ that defines $\hat{p}$ to be the centroid of the feasible region. Consider three circuits $C_1, C_2, C_3$, about which no information is supplied except that $0 \leq p(C_1), p(C_2), p(C_3) \leq 1$. Then $G_{\text{cent}}$ calculates $\hat{p}(C_1) = \hat{p}(C_2) = \hat{p}(C_3) = \frac{1}{2}$—so far so good.

Now, let $C = C_1 \land C_2 \land C_3$. Note that $0 \leq p(C) \leq p(C_1), p(C_2), p(C_3)$ and also that $p(C) \geq p(C_1) + p(C_2) + p(C_3) - 2$. This second fact follows from the fact that

$$1 - p(C) = p(\neg C_1 \lor \neg C_2 \lor \neg C_3) \leq p(\neg C_1) + p(\neg C_2) + p(\neg C_3) = 3 - p(C_1) - p(C_2) - p(C_3).$$

These are the only linear constraints on $p(C)$ in terms of $p(C_1), p(C_2), p(C_3)$ that always hold, no matter what $C_1, C_2, C_3$ are. Now, if $G$ is supplied this additional information, it turns out $\hat{p}(C_1) = \frac{14}{25}$ (as can be shown by taking an integral). In other words, giving $G_{\text{cent}}$ irrelevant information—information about some auxiliary circuit that was guaranteed to be true no matter what the circuits $C_1, C_2, C_3$ are—changed its estimate of $p(C_1)$.

This source of potential dissatisfaction with $G_{\text{lin}}$ and $G_{\text{cent}}$ raises the question: what other properties should a deductive estimation algorithm $G$ have, beyond linearity and respect for proofs? Christiano et al. (2022) informally discuss an “independence of irrelevant arguments” property as a potential desideratum: that $G$’s estimates ought not to depend “irrelevant” information. However, formally defining what it means for information to be irrelevant appears to be difficult. In this section we will consider other desiderata for $G$.

### 9.4.1 0-1 boundedness

In this section, we propose a simple property that we call 0-1 boundedness.

**Definition 9.4.1.** A deductive estimation algorithm $G$ satisfies 0-1 boundedness if for every boolean
circuit $C$ and set of arguments $\Pi$, we have $0 \leq G(C \mid \Pi) \leq 1$.

As we discussed in Section 9.3, the respect for proofs property does not force $G$ to respect universal proofs. For example, a proof that $0 \leq p(C) \leq 1$ for all boolean circuits $C$ cannot be turned into an argument $\pi$ such that $0 \leq G(C \mid \pi) \leq 1$ for all $C$. The 0-1 boundedness property is weaker than “respect for universal proofs” – and yet, it cannot be satisfied together with linearity and respect for proofs.

**Theorem 9.4.2.** Unless $P = PP$, there is no polynomial-time deductive estimation algorithm that satisfies linearity, respect for proofs, and 0-1 boundedness.

**Proof.** Let $G$ be a deductive estimation algorithm that satisfies linearity, respect for proofs, and 0-1 boundedness. We will show that a polynomial-time algorithm with oracle access to $G$ can solve #3CNF in polynomial time. This means that either $P = PP$ or $G$ is not a polynomial-time algorithm.

Recall the circuit $\tilde{C}(x, b)$ from Example 9.3.5:

$$
\tilde{C}(x, b) = (x_{i_{1,1}} = b_{1,1} \lor x_{i_{1,2}} = b_{1,2} \lor x_{i_{1,3}} = b_{1,3}) \land \cdots \land (x_{i_{k,1}} = b_{k,1} \lor x_{i_{k,2}} = b_{k,2} \lor x_{i_{k,3}} = b_{k,3}),
$$

for some fixed indices $i_{1,1}, \ldots, i_{k,3}$ that are implicit in the notation.

Consider the proof that $p(\tilde{C}) = (7/8)^k$. Because $G$ satisfies respect for proofs, this proof can be turned into an argument $\pi_0$ such that $G(\tilde{C} \mid \Pi) = (7/8)^k$ for all $\Pi$ containing $\pi_0$.

We now consider $8k$ additional arguments: eight arguments for every clause. Let $j \in [k]$ and $y = (y_1, y_2, y_3) \in \{0, 1\}^3$, and define $A_{j,y}$ to be the partial assignment

$$
A_{j,y} = (b_{j,1} = y_1, b_{j,2} = y_2, b_{j,3} = y_3, x_{i_{j,1}} = \neg y_1, x_{i_{j,2}} = \neg y_2, x_{i_{j,3}} = \neg y_3).
$$

Then $p(\tilde{C}[A_{j,y}]) = 0$. This is because $A$ makes clause $j$ false, and thus – no matter how the rest of the variables are set – the entire formula is false. The proof of this fact can be turned into an argument $\pi_{j,y}$ such that $G(\tilde{C}[A_{j,y}] \mid \Pi) = 0$ for all $\Pi$ containing $\pi_{j,y}$.
Let $\tilde{\Pi}$ consist of $\pi_0$ together with $\pi_{j,y}$ for all $k$ choices of $j$ and eight choices of $y$.

**Claim 9.4.3.** For all assignments $(x, b)$ to $\tilde{C}$, we have

$$G(\tilde{C}[x, b] \mid \tilde{\Pi}) = \tilde{C}(x, b).$$

**Proof.** We first show that $G(\tilde{C}[x, b] \mid \tilde{\Pi}) \leq \tilde{C}(x, b)$ for all $(x, b)$. First, if $\tilde{C}(x, b) = 1$, then $G(\tilde{C}[x, b] \mid \tilde{\Pi}) \leq 1 = \tilde{C}(x, b)$ by 0-1 boundedness.

Now, let $(x, b)$ be such that $\tilde{C}(x, b) = 0$, and suppose for contradiction that $G(\tilde{C}[x, b] \mid \tilde{\Pi}) > 0$. $(x, b)$ violates some clause; call it $j$. Let $y = (b_{j,1}, b_{j,2}, b_{j,3})$. By respect for proofs, we have that $G(\tilde{C}[A_{j,y}] \mid \tilde{\Pi}) = 0$. By (repeated applications of) linearity, this means that

$$\mathbb{E}_{x', b'} \text{ consistent with } A_{j,y} \left[ G(\tilde{C}[x', b'] \mid \tilde{\Pi}) \right] = G(\tilde{C}[A_{j,y}] \mid \tilde{\Pi}) = 0.$$

But $(x, b)$ is consistent with $A_{j,y}$ and $G(\tilde{C}[x, b] \mid \tilde{\Pi}) > 0$ by assumption. It follows that for some $(x', b')$, $G(\tilde{C}[x', b'] \mid \tilde{\Pi}) < 0$. But this contradicts 0-1 boundedness.

We now know that $G(\tilde{C}[x, b] \mid \tilde{\Pi}) \leq \tilde{C}(x, b)$ for all $(x, b)$. Now, suppose for contradiction that $G(\tilde{C}[x, b] \mid \tilde{\Pi}) < \tilde{C}(x, b)$ for some $(x, b)$. By (repeated applications of) linearity, it follows that

$$G(\tilde{C} \mid \tilde{\Pi}) = \mathbb{E}_{x, b} \left[ G(\tilde{C}[x, b] \mid \tilde{\Pi}) \right] < \mathbb{E}_{x, b} \left[ \tilde{C}(x, b) \right] = p(\tilde{C}) = (7/8)^k.$$

But this contradicts respect for proofs, because $\pi_0 \in \tilde{\Pi}$. \qed

By linearity, it follows from Claim 9.4.3 that for every $b$, we have

$$G(\tilde{C}[b] \mid \tilde{\Pi}) = \mathbb{E}_x \left[ \tilde{C}(x, b) \right] = p(\tilde{C}[b]).$$

We can now give an algorithm that, with oracle access to $G$, solves $\#3CNF$ in polynomial time. Consider any 3CNF $C$ on $n$ variables. We have that $C = \tilde{C}[b]$ for a particular $\tilde{C}$ (i.e. for particular choices of indices $i_{1,1}, \ldots, i_{k,3}$) and $b$. Create $\tilde{\Pi}$ as above. Return $2^n G(\tilde{C}[b] \mid \tilde{\Pi})$. The resulting
output will be $2^n \hat{p}(\tilde{C}[b])$, which is equal to the number of satisfying assignments to $C$. 

### 9.4.2 Desiderata inspired by conditional expectation

We have now shown that we cannot hope for $G$ to satisfy linearity and respect for proofs while requiring $G$ to output estimates in $[0, 1]$. What other properties might we hope for $G$ to satisfy? In this section, we informally discuss two such properties that are inspired by properties of conditional expectations.

We call the first of these properties *iterated estimation*, because it is inspired by the law of iterated expectations from probability theory. Roughly speaking, the law of iterated expectations states that for a probability space $(\Omega, \mathcal{F}, P)$, a random variable $X$ defined on the space, and $\sigma$-algebras $\mathcal{H}_1 \subseteq \mathcal{H}_2 \subseteq \mathcal{F}$, we have

$$
\mathbb{E} [\mathbb{E} [X \mid \mathcal{H}_2] \mid \mathcal{H}_1] = \mathbb{E} [X \mid \mathcal{H}_1].
$$

Why is this true? We can think of $\mathcal{H}_1$ as revealing partial information about the state of the world $\omega \in \Omega$, and $\mathcal{H}_2$ as revealing more fine-grained information than $\mathcal{H}_1$. The right-hand side is a random variable that is the expectation of $X$ conditioned on knowing the information in $\mathcal{H}_1$. The left-hand side is the expected value of what you will think the expectation of $X$ is after learning the information in $\mathcal{H}_2$, if you only know the information in $\mathcal{H}_1$. These are the same because, if you knew that in expectation you would update your guess about $X$ in a particular direction once you learned $\mathcal{H}_2$, it would make sense to update your guess about $X$ before learning $\mathcal{H}_2$.

**Definition 9.4.4 (Informal).** A deductive estimation algorithm $G$ satisfies iterated estimation if for all boolean circuits $C$ and for all sets of arguments $\Pi$ and $\Pi'$, we have

$$G(G(C \mid \Pi, \Pi') \mid \Pi) = G(C \mid \Pi).$$

---

17For readers unfamiliar with $\sigma$-algebras, we suggest thinking of $\mathcal{H}_1$ as partial information about $X$ and $\mathcal{H}_2$ as more fine-grained partial information about $X$. 

250
In other words: if $G$ only knows the arguments in $\Pi$, then its estimate for what its estimate of $p(C)$ would be if it also considered $\Pi'$, is just its current estimate of $p(C)$.

The statement of iterated estimation does not quite make sense, because we said that the first argument to $G$ is a boolean circuit – whereas $G(C \mid \Pi, \Pi')$ is not quite a boolean circuit.

However, we can make sense of this property by expanding the scope of the first argument to $G$. Suppose we allow $G$ to accept not just boolean circuits, but arbitrary computer programs (Turing machines) that output a real number. Then $G(C \mid \Pi, \Pi')$ is one such computer program – namely, it takes no inputs, runs $G$ on the input $(C, (\Pi, \Pi'))$, and outputs the output of $G$. While in this chapter we have dealt with boolean circuits for simplicity, we are interested in the deductive estimation of the outputs of computer programs in general, and do not see a fundamental obstacle to enriching the space of inputs to $G$ to allow for arbitrary programs.

Returning to the iterated estimation property, it makes sense to ask: does the parallel to the law of iterated estimation make sense, or should we want $G$ to behave differently from an iterated expectation? One reason for skepticism is that it requires the outer $G$ in the expression $G(G(C \mid \Pi, \Pi') \mid \Pi)$ to be “blinded” to $\Pi'$ when estimating $G(C \mid \Pi, \Pi')$. That is, the outer $G$ sees the expression $G(C \mid \Pi, \Pi')$, but does not “process” it, beyond the extent allowed by $\Pi$.

But in a sense, this is the behavior we desire from $G$. We do not expect $G$ to be able to deduce properties of the circuit (or program) that it is estimating, except for the properties that are given to $G$ as arguments. In that sense, we should not expect the outer $G$ to be able to draw conclusions from $\Pi'$, except to the extent that it can do so based on the arguments that it has been given (i.e. $\Pi$).

The second property is inspired by a different property of conditional expectations, called “pulling out known factors.” The pulling out known factors property of conditional expectations states roughly that for an $\mathcal{H}$-measurable random variable $X$ and a random variable $Y$, we have $E[XY \mid \mathcal{H}] = XE[Y \mid \mathcal{H}]$. In other words, if the information supplied by $\mathcal{H}$ is sufficient to determine the value of $X$, then $X$ can be pulled out of the conditional expectation.

---

18The fact that this program takes no inputs is nothing new: we have already considered expressions such as $G(C[x] \mid \Pi)$, where $x$ is a full assignment to $C$, and so $C[x]$ is a circuit with no inputs that outputs $C(x)$.
Definition 9.4.5 (Informal). A deductive estimation algorithm $G$ satisfies the pulling out known factors property if for boolean circuits $C_1, C_2$ and sets of arguments $\Pi, \Pi'$, we have

$$G(G(C_1 \mid \Pi)p(C_2) \mid \Pi, \Pi') = G(C_1 \mid \Pi)G(C_2 \mid \Pi, \Pi').$$

Again, we require $G$ to accept arbitrary programs as input, and interpret $G(C_1 \mid \Pi)p(C_2)$ as the program that computes $G(C_1 \mid \Pi)$, computes $p(C_2)$, and returns their product. The intuition is the same as for the corresponding property of conditional expectations: in the expression $G(G(C_1 \mid \Pi)p(C_2) \mid \Pi, \Pi')$, the outer $G$ knows $\Pi$, so it knows what its estimate of $p(C_1)$ would be if it only knew $\Pi$. To the outer $G$, the value $G(C_1 \mid \Pi)$ is a particular constant, rather than an unknown, so it can be “pulled out.”

9.5 Mechanistic anomaly detection for neural networks

Let $G_{\text{good}}$ be a hypothetical deductive estimation algorithm that we would find satisfying: perhaps it satisfies linearity, respect for proofs, and some other important desiderata. One potential use case for $G_{\text{good}}$ is to tell apart different reasons for why a circuit might output 1. Recall Example 9.1.1 from the introduction: $C$ takes as input a triple $(a, b, c)$ of positive integers and accepts if $\max(a, b) = \max(b, c)$. Note that $C$ can accept an input for one of two “reasons”: either $b \geq a, c$ (which happens with probability $\frac{1}{3}$ or so), or $a = c$ (which happens extremely rarely).

Let $\pi$ be the proof that if $b \geq a, c$ then $C(a, b, c) = 1$. We might expect the following behavior from $G_{\text{good}}$: for all $(a, b, c)$ such that $b \geq a, c$, we have $G_{\text{good}}(C[a, b, c] \mid \pi) = 1$. Meanwhile, if $b < a$ or $b < c$, then $G_{\text{good}}(C[a, b, c] \mid \pi)$ is close to zero – even if in fact $a = c$ – because $\pi$ does not point out the fact that if $a = c$, then $C(a, b, c) = 1$. Thus, $G_{\text{good}}$ allows us to distinguish inputs on which $C$ outputs 1 for the “usual” reason $(b \geq a, c)$ from inputs on which $C$ outputs 1 for a different reason ($a = c$).

---

19 This program may not run in polynomial time, but that is fine: $G$ can estimate the outputs of programs that may take a long time to run.
Of course, this approach is only useful insofar as it can be applied to circuits that we don’t already understand. For such circuits, there is an obvious barrier to using $G_{\text{good}}$ in this way: how do we find the deductive arguments to give to $G_{\text{good}}$? In general, finding deductive arguments is as hard as finding proofs of mathematical statements. However, in the particular case of machine learning, we may be able to find deductive arguments that allow us to understand the neural networks that we have trained.

The key fact about neural networks is that they are trained from random initialization via gradient descent. We can hope to learn deductive arguments for the behavior of a neural network also via gradient descent, in parallel with training the network. These arguments might be features in some continuous parameterized space, somewhat akin to the features $\varphi_i$ in Section 9.2. One possibility is that these features will be simple functions of the neural network’s weights and other features. The features would be learned to maximize predictive accuracy of the neural network’s behavior: for example, if the neural network (call it $N$) is trained to get a high reward according to a reward model (call it $R$), and it learns to get high reward on the input distribution $D$, then we would hope to learn features $\varphi_1, \ldots, \varphi_m$ such that $G_{\text{good}}(\mathbb{E}_{x \sim D} [R(x, N(x))] \mid \varphi_1, \ldots, \varphi_m)$ is (correctly) high.\footnote{How does respect for proofs figure into this? The respect for proofs property is meant to capture that the space of possible features must be extremely expressive: enough that features can encode formal proofs about $N$ and $R$. In practice, the features that we learn are unlikely to be proofs; however, respect for proofs is an easily formalizable property that is necessary (even if not sufficient) for the features to be arbitrarily expressive. It is thus a convenient property to work with when taking a theoretical approach to finding a satisfying deductive estimation algorithm.}

We may then be able to use these features for mechanistic anomaly detection (Christiano, 2022). The mechanistic anomaly detection problem can be stated informally as follows: suppose that a neural network $N$ gets high average reward from a reward model $R$ over an input distribution $D$ (that is, $\mathbb{E}_{x \sim D} [R(x, N(x))]$ is large). Let $x^* \notin D$ be a new input on which $N$ again gets high reward (that is, $R(x^*, N(x^*))$ is also large). Did $N$ get high reward on $x^*$ for the same reasons that it got high average reward on $D$, or for a different reason?

For example, suppose that $N$ is an financial assistant that takes actions such as buying and selling stocks, transferring money between bank accounts, and paying taxes. Suppose that $R$ sanity-
checks $N$’s actions for legal compliance, and then rewards $N$ if it has turned a profit. Then $N$ may have two ways of getting high reward: *successful task completion* (making good trades on the stock market) and *successful deception* (for example, making a profit via money laundering and then covering its tracks). Suppose that $D$ is a “clean” distribution of input scenarios: on each scenario in $D$, $N$ achieves high reward via the “successful task completion” route. If $x^*$ is a new input scenario on which $N$ again achieves high reward, we want to be able to determine whether $N$ successfully completed the task (got high reward *for the same reasons* as it did on $D$) or if it instead successfully deceived $R$ (got high reward *for a different reason*). In the latter case, we would like to classify $N$’s behavior on $x^*$ as “anomalous” (and perhaps refer the input for more extensive oversight).

Mechanistic anomaly detection is a particularly challenging problem, because it cannot in general be solved by looking at $N$’s input-output behavior. After all, a sufficiently advanced reward model $R$ would be able to integrate observations of input-output behavior into its oversight process. To catch successful deception, it could be necessary to analyze $N$’s internals. This is why sophisticated machinery such as deductive estimation may be necessary.

How can deductive estimation be used to solve the mechanistic anomaly detection problem? Suppose that we successfully learn features $\varphi_1, \ldots, \varphi_m$ that explain $N$’s high average reward to $G_{\text{good}}$ – that is, $G_{\text{good}}(\mathbb{E}_{x \sim D} [R(x, N(x))] \mid \varphi_1, \ldots, \varphi_m)$ is appropriately large. Then consider the quantity

$$G_{\text{good}}(R(x^*, N(x^*)) \mid \varphi_1, \ldots, \varphi_m).$$

If $N$ gets high reward on $x^*$ for the same reasons than it got high reward on inputs in $D$, then $\varphi_1, \ldots, \varphi_m$ should be sufficient to explain $N$’s high reward on $x^*$ as well, and so we expect $G_{\text{good}}(R(x^*, N(x^*)) \mid \varphi_1, \ldots, \varphi_m)$ to be large as well. On the other hand, suppose that $N$ gets high reward on $x^*$ for a *different* reason than it got high reward on $D$. Then $G_{\text{good}}(R(x^*, N(x^*)) \mid \varphi_1, \ldots, \varphi_m)$ will probably be small, because the reasons why $N$ did well on $D$ (which are represented as features $\varphi_1, \ldots, \varphi_m$) are *insufficient* to explain $N$’s high reward on $x^*$. So if $R(x^*, N(x^*))$ is large, but $G_{\text{good}}(R(x^*, N(x^*)) \mid \varphi_1, \ldots, \varphi_m)$ is small, then we classify $N$’s behavior on $x^*$ as
anomalous.\textsuperscript{21}

\section*{9.6 Conclusion}

In my view, the problem of deductive circuit estimation – finding an efficient deductive estimation algorithm that satisfies important properties such as linearity, respect for proofs, and more – has two compelling motivations.

The first is theoretical elegance and importance. In the introduction, we gave several examples of deductive arguments about a circuit. Those arguments were not proofs, but in an important sense, they were sound. Barring further arguments, it is reasonable to guess that the first 128 bits of the output of SHA-256 are as likely to be larger than the last 128 bits as they are to be smaller. Barring further observations about the structure of a 3CNF with \(k\) clauses, it is reasonable to guess that its acceptance probability is \((7/8)^k\). Such deductive reasoning is commonplace in computer science and mathematics.

Christiano et al. (2022) asked whether it is possible to formalize such reasoning: that is, to design an algorithm that estimates quantities given formal arguments similar to the ones in the introduction. This question seems fundamental: much as mathematicians have found a formalization of a mathematical proof, one could hope for a formalization of a much broader class of deductive arguments.

In this chapter, we continued this formalization project by designing an algorithm for boolean circuit estimation that satisfies linearity and respect for proofs, and by suggesting further properties that deductive estimation algorithms ought to satisfy. However, we have not satisfied those further properties, nor am I convinced that we have found the right set of properties to satisfy. We have raised more questions than we have answered. But in my opinion, the questions that we have raised are fundamental and exciting.

\textsuperscript{21}In fact, given the input \(x^*\), we may further optimize our features \(\varphi_1, \ldots, \varphi_m\), with the goal of finding features that still cause \(G\) to estimate that \(N\)'s average reward on \(D\) is large, while causing \(G\) to estimate that \(N\)'s reward on \(x^*\) is small. If we find any such features, we label \(N\)'s behavior on \(x^*\) anomalous. Of course, care must be taken to not over-optimize, as this runs the danger of labeling every input “anomalous.”
The second motivation for this work is practical relevance to the AI alignment problem. AI labs such as OpenAI and Anthropic are putting substantial effort into scalable oversight: methods of overseeing training that will continue to be reliable even as AI capabilities advance (Anthropic, 2023; Leike & Sutskever, 2023). However, a sufficiently advanced AI could find ways to exploit loopholes in its oversight process. While oversight approaches that rely on observations of input-output behavior may work, this is not guaranteed. Approaches that use model internals as part of the oversight process may be necessary to build safe advanced AI.

In Section 9.5, we saw how deductive estimation algorithms could be used to oversee an AI by flagging anomalous behavior. There are many challenges ahead for this approach, and this path forward is more complex and speculative than most approaches to scalable oversight. However, this approach has the potential to succeed even under somewhat pessimistic assumptions about the difficulty of the AI alignment problem.

For both of these reasons – the theoretical and the practical – I find deductive estimation to be a compelling research direction. I am excited to see what the future holds for this area.

---

22 See Ji et al. (2023) for a survey of the AI alignment problem.
23 See https://www.alignment.org/blog/ for further discussion.
Epilogue

Each technical chapter of this thesis explored a different facet of algorithmic Bayesian epistemology. Chapters 3 and 4 explored the elicitation of knowledge under strategic constraints. Chapter 5 bridged the problem of knowledge elicitation with the problem knowledge aggregation. Chapters 6 and 7 focused on the knowledge aggregation problem, now under informational constraints. Chapter 8 also explored aggregation, but now under communication constraints. Finally, Chapter 9 explored belief formation under both computational and informational constraints.

We closed every chapter with a discussion of future directions. Some chapters had a few suggestions, while others had many. Depending on how you count, Chapter 7 had as many as 1,152 suggestions! Some closed with fully formal problem statements (Chapter 4), others with problem statements so informal that formalizing them would in itself be a breakthrough (Chapter 9). Some suggestions were intellectual curiosities, while others had significant potential for application. Some were merely interesting, while others were (in my opinion) utterly fascinating.

In light of all these differences, I’d like to close by highlighting the directions in ABE that I find most exciting. These directions are listed in order of the chapter(s) to which they are relevant.

**Bayesian justifications for generalized QA pooling** In Section 2.4, we gave Bayesian justifications for generalized linear and logarithmic pooling (i.e. linear and logarithmic pooling but with weights that do not necessarily add to 1). That is, we presented information structures in which a generalized linear pool is the *exactly correct* aggregation method, and similarly for generalized

---

1After all, the question of Chapter 8 can be framed as: under what circumstances can Alice and Bob aggregate their knowledge by sharing a small amount of their information?
logarithmic pooling. Can we do this for every QA pooling method – that is, generalized QA pooling with respect to every proper scoring rule?

**Directions in robust forecast aggregation**  As discussed in Section 7.6, robust forecast aggregation seems like a *particularly* exciting area for future work. Highlights include:

- *Generalization to KL divergence.* In our exploration of the known prior setting, we found that averaging experts’ forecasts and then moving the average away from the prior by a constant factor results in a robustly high-quality forecast. However, if the value $Y$ being forecast is a probability, then this procedure can result in aggregate forecasts outside of $[0, 1]$. If the error measure were changed from squared error to KL divergence, then such forecasts would naturally be disallowed, as they would incur an infinite penalty. So, what is a robust way to aggregate probabilistic forecasts if the error measure is KL divergence? The answer to this question may have important implications for the aggregation of probabilistic forecasts!

- *Giving the aggregator additional information.* In our problem, the aggregator learns each expert’s expectation of $Y$ and (sometimes) the prior. How much better can the aggregator do if they learn additional information, either about the experts’ beliefs or about the information structure itself? Note that the answer might vary considerably depending on which assumptions are made about the information structure.

**Sophisticated Bayesian models for forecast aggregation**  While this thesis focused primarily on robust forecast aggregation, there is significant low-hanging fruit in Bayesian aggregation: creating a Bayesian model of experts’ information and finding the optimal aggregate (or a good aggregate) under that model. While much more well-studied than robust aggregation, most work on Bayesian aggregation focuses on specific application domains (see McAndrew et al. (2021) for a survey).

One particularly compelling direction is to build on the Gaussian partial information framework introduced by Satopää et al. (2016) (see our exposition in Section 2.3). In this framework, the optimal aggregate is a linear combination of the experts’ forecasts, with weights determined by the overlap between the experts’ information sets (see Section 2.4). Satopää et al. (2017) explore...
how this overlap may be inferred when multiple quantities are being forecast. But how might one aggregate forecasts within this framework if only one quantity is being forecast?

Here is one possible Bayesian model, inspired by this question. Let us say that the experts are forecasting a quantity $Y = X_1 + X_2 + \ldots$, where each $X_j$ is independently and normally distributed, with mean 0 and standard deviation $\frac{1}{j}$. (Essentially, this means that the factors contributing to $Y$ have “importances” that follow a Zipfian distribution. Note that $Y$ is almost surely finite.) There are $m$ experts, each of which knows a subset of the $X_j$’s. In particular, we can model each expert $i$ as having an expertise score $e_i$ (over which we have some prior), and each $X_j$ as having an obviousness score $o_j$ (over which we have some prior). The odds that Expert $i$ knows the value of $X_j$ are $e_i \cdot o_j$ to 1 (or in other words, the probability that expert $i$ knows $X_j$ is $\frac{e_i o_j}{e_i o_j + 1}$). Each expert tells the aggregator their estimate of $Y$, which is the sum of all $X_j$’s whose value the expert knows. The aggregator can then do (approximate) Bayesian inference to find the optimal way to combine the experts’ estimates into a single estimate of $Y$.

Note that it is straightforward to adapt this model to the context of forecasting a binary event. Just as before, we have $Y = X_1 + X_2 + \ldots$, with each expert knowing a subset of the $X_j$’s, but now the experts are estimating the probability that $Y \geq 0$, conditioned on their information. The aggregator’s task is now to combine these probabilities into an overall probability that $Y \geq 0$.

**Finding a good deductive estimator** In Chapter 9, we found a deductive estimation algorithm that satisfies linearity and respect for proofs. We then exhibited an informal property (independence of irrelevant information) that the algorithm does not satisfy, thus prompting the question of whether a “better” deductive estimation algorithm exists. This question decomposes into two sub-questions. First, what does it mean for a deductive estimation algorithm to be good? A formalization of “independence of irrelevant information” would help answer this question, as would formulating other desirable properties of an estimation algorithm (perhaps “iterated estimation" and “pulling out known factors," as discussed in Section 9.4.2). Second, how do we create a deductive estimation algorithm that is good (as defined by the answer to the first question)? As
discussed in Section 9.6, these questions seem quite fundamental.

**Wagering mechanisms that produce good aggregate forecasts** In Chapter 1, we briefly discussed wagering mechanisms, which are a type of mechanism for eliciting probabilistic forecasts from multiple experts. Each expert submits a forecast and a *wager*, and the wagers get redistributed according to each expert’s performance.

Wagering mechanisms are similar to prediction markets, in that they elicit forecasts by giving experts opportunities to place bets. Prediction markets also aggregate experts’ forecasts into a single forecast, which can be inferred from market prices. It is natural to wonder whether wagering mechanisms can also be used for aggregation.

Lambert et al. (2015, §6) ask the natural question of what Bayesian experts’ wagers look like in equilibrium.\(^2\) For a given information structure, we could say that a wagering mechanism produces good aggregate forecasts if pooling the experts’ forecasts according to their equilibrium wagers outperforms pooling the forecasts with equal weights.\(^3\) Perhaps it is possible to create a wagering mechanism that produces good aggregate forecasts for a large, natural class of information structures.

It seems to me that wagering mechanisms have the potential to be a powerful tool for eliciting and aggregating forecasts, alongside prediction markets. However, such wagering mechanisms would need to be designed so that optimal wagers are reflective of confidence.

In summary, algorithmic Bayesian epistemology presents a wide range of fascinating questions and possible directions for further research. I hope that this thesis has conveyed my enthusiasm for ABE, and that it has sparked a similar curiosity in you as well.

---

\(^2\)To circumvent the no-trade theorem, Lambert et al. (2015) posit that the experts get utility from gambling. Alternatively, we could circumvent the no-trade theorem by subsidizing the wagering mechanism.

\(^3\)Intuitively, a wagering mechanism should produce good aggregate forecasts for an information structure if each expert’s equilibrium wager reflects their confidence. By *confidence*, we (informally) mean the extent to which the expert expects their forecast to remain unchanged upon learning other experts’ signals. If experts’ wagers reflect their confidence, then an expert who knows almost all of the available information will make a large wager, while an expert who knows that other experts have important private information will make a small wager. In that case, using wagers as weights for aggregation should produce a sensible result.
References


Christiano, P. (2022, November). *Mechanistic anomaly detection and elk [Joint work with Mark Xu; Follow-up to Eliciting Latent Knowledge]*. (Cit. on p. 253).


NOAA. (2017). 6 tools our meteorologists use to forecast the weather. *NOAA Research* (cit. on pp. 4, 5).


271


Appendix A: Details omitted from Chapter 3

A.1 Details omitted from Section 3.2

**Fact A.1.1.** If a coin’s bias is chosen uniformly from $[0, 1]$, then for all $n$ and $0 \leq k \leq n$, the probability that exactly $k$ of $n$ flips come up heads is $\frac{1}{n+1}$.

*Proof.* Consider the following method of choosing the $n$ flip outcomes: first, choose a number $r$ in $[0, 1]$ (the bias); then, choose $n$ numbers in $[0, 1]$, each of which corresponds to a heads flip if it is less than $r$. The probability of $k$ heads is the probability that the first of the $n+1$ numbers is the $(k + 1)$-th smallest, which is $\frac{1}{n+1}$. \hfill $\square$

**Lemma 3.2.4.** A continuously differentiable scoring rule $s$ is weakly proper if and only if for all $p \in (0, 1)$, $ps'(p) = (1 - p)s'(1 - p)$ and $s'(p) \geq 0$. It is (strictly) proper if and only if additionally $s'(p) > 0$ almost everywhere\(^1\) in $(0, 1)$.

*Proof.* Fix a scoring rule $s$ and let $r_p(x)$ be the expected score of an expert with belief $p$ who reports $x$ (so $r_p(x) = ps(x) + (1 - p)s(1 - x)$).

We first prove that if $s$ is weakly proper then it satisfies the two stated constraints. Suppose that $s$ is weakly proper. It is clear that $s$ satisfies the first equation: for all $p$, in order for $r_p(x)$ to have a maximum at $x = p$, its derivative $ps'(x) - (1 - p)s'(1 - x)$ must be 0 at $x = p$. So we first conclude that we must have $xs'(x) = (1 - x)s'(1 - x)$ for all $x \in (0, 1)$. Next, observe that

$$r'_p(x) = ps'(x) - (1 - p)s'(1 - x) = ps'(x) - (1 - p)\frac{x s'(x)}{1 - x} = s'(x) \left(p - (1 - p) \frac{x}{1 - x}\right). \tag{A.1}$$

Suppose for contradiction that for some $p \in (0, 1)$, we have $s'(p) < 0$. Since $s'$ is continuous,

\(^1\)Almost everywhere on $(0, 1)$ refers to the interval $(0, 1)$ except a set of measure zero.
s′(x) < 0 on some open interval containing p. On that open interval, then, the sign of r′ₚ(x) is the opposite of the sign of p – (1 – p)\frac{x}{1-x} – that is, negative when x < p and positive when x > p. But then rₚ(x) is strictly minimized, rather than maximized, at x = p on this interval, contradicting that s is weakly proper.

To prove the stronger claim when s is proper, assume for contradiction that s′(p) is not strictly positive almost everywhere. Then because s′ is continuous, there is an interval of non-zero length in which s′(x) = 0 on the entire interval. Let p lie on the interior of this interval. Equation (A.1) then establishes that r′ₚ(x) is 0 in an interval around x = p, meaning that p is not the unique maximizer of rₚ, contradicting that s is proper.

Conversely, suppose that s satisfies the two stated constraints. We show that s is weakly proper by showing a stronger statement: that for all p, rₚ(x) weakly increases on (0, p] and weakly decreases on [p, 1). By the first constraint, (A.1) holds. By the second constraint, for all x, r′ₚ(x) is either 0 or has the sign of p – (1 – p)\frac{x}{1-x}, i.e. positive if x < p and negative if x > p. This means that rₚ(x) is weakly increasing on (0, p] and weakly decreasing on [p, 1), and so attains a weak global maximum at x = p, as desired.

To prove the stronger claim when s′(x) > 0 almost everywhere, we show that rₚ(x) strictly increases almost everywhere on (0, p] and strictly decreases almost everywhere on [p, 1). Again, (A.1) holds, so by the second constraint we have that r′ₚ(x) has the sign of p – (1 – p)\frac{x}{1-x} almost everywhere, i.e. positive if x < p and negative if x > p. Thus, rₚ(x) is strictly increasing almost everywhere on (0, p] and strictly decreasing almost everywhere on [p, 1), and so attains a strict global maximum at x = p, as desired. □

**Corollary 3.2.5.** Let s be strictly increasing almost everywhere (resp., nondecreasing everywhere) and continuously differentiable on (0, \frac{1}{2}]. Then s can be extended to a continuously differentiable proper (resp., weakly proper) scoring rule on (0, 1) by defining s′(p) = \frac{1-p}{p} s′(1-p) for p ∈ (\frac{1}{2}, 1).

**Proof.** As \frac{1-p}{p} is strictly positive on (\frac{1}{2}, 1), and s′(1 – p) is strictly (resp., weakly) positive almost everywhere on (\frac{1}{2}, 1), we immediately conclude that s′(p) is also strictly (resp., weakly) positive almost everywhere on (\frac{1}{2}, 1). Therefore, s is proper (resp., weakly proper) by Lemma 3.2.4. □
Proposition A.1.2. Let $s$ be a proper scoring rule and define $\text{Score}_c(s)$ to be the expected score of a globally-adaptive expert who is scored with $s$. Then $\lim_{c \to 0} \text{Score}_c(s) = \int_0^1 G_s(x)dx$. That is, $\text{Score}_c(s) = \int_0^1 G_s(x)dx \pm o(1)$.

Proof. First, we wish to argue that as $c \to 0$, no matter the true bias, the number of flips any expert will choose to make approaches $\infty$. To see this, observe that after $n$ flips, the expert’s current belief will always be an integer multiple of $\frac{1}{n+1}$. So if $c(m) := \min \left\{ \frac{k+1}{n+2} G_s\left(\frac{k+2}{n+3}\right) + \frac{n-k+1}{n+2} G_s\left(\frac{k+1}{n+3}\right) - G_s\left(\frac{k+1}{n+2}\right) \right\}$ then the expert will always flip the coin at least $m$ times as long as $c \leq c(m)$. Observe also that the minimum is taken over finitely many terms, all of which are strictly positive, so $c(m)$ is strictly positive. Therefore, for all $m$, there is a sufficiently small $c(m) > 0$ such that whenever the cost is at most $c(m)$, the expert flips at least $m$ times no matter the true bias. Note that while these calculations are done for an adaptive expert, they hold for a non-adaptive expert as well because the non-adaptive expert would want to flip at least $m$ coins no matter the outcomes.

Now, let’s consider the expected score of an expert who makes exactly $m$ flips no matter what. For all $k$, such an expert sees $k$ total heads with probability $\frac{1}{m+1}$ (Fact A.1.1). And conditioned on seeing $k$ heads, the expert’s expected score tomorrow is $G_s\left(\frac{k+1}{m+2}\right)$. Therefore, we can conclude that for all $m$, the expert’s expected score after $m$ coin flips is $\frac{1}{m+1} \sum_{k=0}^{m} G_s\left(\frac{k+1}{m+2}\right)$.

Now, we want to understand the limit of this sum as $m \to \infty$. Observe that for each $m$, the sum is a Riemann sum for the function $G_s(x)$ on $[0, 1]$ (i.e. each $\frac{k+1}{m+2}$ lies inside the interval $[\frac{k}{m+1}, \frac{k+1}{m+1}]$). Therefore, the limit as $m \to \infty$ is just the Riemann integral, and we get:

$$\lim_{m \to \infty} \frac{1}{m+1} \sum_{k=0}^{m} G_s\left(\frac{k+1}{m+2}\right) = \int_0^1 G_s(x)dx.$$

So now we can conclude that the non-adaptive expert gets expected payoff $\int_0^1 G_s(x)dx + o(1)$. As $c \to 0$, the number of flips $m \to \infty$, and the expected payoff as $m \to \infty$ approaches $\int_0^1 G_s(x)dx$. For an adaptive expert, observe that by Lemma 3.2.7, as long as they flip the coin at
least \( m \) times with probability 1, their expected score is at least as large as if they flipped it exactly \( m \) times with probability 1. As their expected score can certainly not exceed \( \int_0^1 G_s(x)dx \) (as this is the score of a perfect expert who knows exactly the bias), their expected score must also approach \( \int_0^1 G_s(x)dx \) as \( m \to \infty \) (and therefore as \( c \to \infty \) as well).

\[ \square \]

**Claim 3.2.9.** For a weakly proper scoring rule \( s \), we have

\[
\int_0^1 G_s(x)dx = s\left(\frac{1}{2}\right) + \int_{\frac{1}{2}}^1 (1-x)s'(x)dx.
\]

Thus, \( s \) is normalized if and only if \( s(1/2) = 0 \) and \( \int_{\frac{1}{2}}^1 (1-x)s'(x)dx = 1 \).

**Proof.** We have

\[
\int_0^1 G_s(x)dx = \int_0^1 (xs(x) + (1-x)s(1-x))dx = \int_0^1 2xs(x)dx
\]

where the last step follows by separating \((1-x)s(1-x)\) into its own integral and substituting \( u = 1-x \). Now we integrate by parts, letting \( u = s(x) \) and \( dv = 2dx \), so that \( du = s'(x) \) and \( v = x^2 \), to get

\[
\int_0^1 G_s(x)dx = x^2s(x) \big|_0^1 - \int_0^1 x^2s'(x)dx = s(1) - \int_0^1 x^2s'(x)dx
\]

\[
= s(1) - \int_0^{\frac{1}{2}} x^2s'(x)dx - \int_{\frac{1}{2}}^1 x^2s'(x)dx
\]

\[
= s(1) - \int_0^{\frac{1}{2}} x(1-x)s'(1-x)dx - \int_{\frac{1}{2}}^1 x^2s'(x)dx
\]

\[
= s(1) - \int_{\frac{1}{2}}^1 (x(1-x) + x^2)s'(x)dx = s(1) - \int_{\frac{1}{2}}^1 xs'(x)dx.
\]

Expressing \( s(1) - s\left(\frac{1}{2}\right) \) as \( \int_{\frac{1}{2}}^1 s'(x)dx \), we obtain the desired equality. \[ \square \]
A.2 Properties of respectful scoring rules

We state several sufficient conditions for a scoring rule to be respectful, confirm that typical scoring rules are respectful, and provide a brief discussion.

Claim A.2.1. If Conditions 1 and 2 of Definition 3.3.1 hold and $G_s''(x)$ is bounded on $(0, 1)$ then $s$ is respectful.

This should be clear: take $c$ to be small enough such that $c^{-16}$ times the lower bound on $G_s''$ exceeds the upper bound on $|G_s'''|$. Scoring rules such as the quadratic scoring rule and the spherical scoring rule satisfy the hypotheses of Claim A.2.1. One well-known generalization of the quadratic scoring rule is the Tsallis rule (Tsallis, 1988). This rule, parametrized by $\gamma > 1$, is defined to be the scoring rule $s$ for which $G_s(x) = x^\gamma + (1 - x)^\gamma$. When $\gamma = 2$, this yields the quadratic scoring rule.

For $\gamma \geq 3$, it is evident that the Tsallis rule satisfies the hypotheses of Claim A.2.1. However, this is not so for $\gamma < 3$ (except for $\gamma = 2$). Perhaps more importantly, the logarithmic scoring rule does not satisfy Claim A.2.1 either. This motivates the following result (the proof appears at the end of this section).

Claim A.2.2. Suppose that Conditions 1 and 2 of Definition 3.3.1 are satisfied. Suppose further that $G_s'''$ is bounded on any closed sub-interval of $(0, 1)$, and that there exist constants $k \neq 0$ and $r$ such that $\lim_{x \to 0} x^r G_s'''(x) = k$. Then $s$ is respectful.

The logarithmic scoring rule satisfies the hypotheses of Claim A.2.2 ($r = 2$ and $k = -1$). The Tsallis rule with $\gamma < 3$ (and $\gamma \neq 2$) also satisfies these hypotheses ($r = 3 - \gamma$ and $k = \gamma(\gamma - 1)(\gamma - 2)$). The proof of Claim A.2.2 is more involved, and deferred to Appendix A.3.

We briefly discuss ways in which proper scoring rules can fail to be respectful. One way a scoring rule can be disrespectful is if $G_s''$ grows extremely quickly near zero (e.g. $G_s''(x) = e^{1/x}$). Such functions, however, are outside the scope of this entire exercise because they are not
normalizable. That is, such $G_s$ have $\int_0^1 G_s(x)\,dx = \infty$, and provide infinite expected payment to the expert. So this “limitation” of respectfulness is more of a restatement of normalizability.

Another way a scoring rule could be disrespectful is if $G''_s$ is not bounded away from zero. For example: $G''_s(x) = \left(x - \frac{1}{2}\right)^2$ or $G''_s(x) = x(1 - x)$. If $G''_s$ remains “very flat” near 0 for a “large interval”, then $\text{Ind}_\ell(s)$ is infinite anyway. This implies that we should expect the error to be a larger order of magnitude than $c^{-\ell/4}$, and for such functions to not incentivize precision well at all (although we do not explicitly prove this). This makes sense: if $G''_s(x) \approx 0$, then the expert gains $\approx 0$ by flipping the coin to refine their current belief (Lemma 3.2.7). It is also possible that $G''_s$ is not bounded away from zero, but also not “very flat”. We conjecture that Theorem 3.3.3 does hold for such functions, but that our approach does not establish this. While it is possible to come up with such functions (e.g., the two above) which elude Theorem 3.3.3’s precise statement, this does not affect commonly-studied scoring rules, nor the scoring rules designed in this paper (sometimes leaning on Theorem 3.6.1).

Finally, as with any exercise in analysis, there are continuous functions that behave erratically near zero, such as $G''_s(x) = \sin \left(\frac{1}{x}\right) + \sin \left(\frac{1}{1-x}\right) + 3$. While it may or may not be the case that Theorem 3.3.3 extends to such functions, this does not seem particularly relevant.

**Proof of Claim A.2.2.** Let $r, k$ be as in Claim A.2.2. If $r \leq 0$ then the claim is uninteresting: $G''_s$ is bounded on $(0, 1)$ and so the statement is subsumed by Claim A.2.1. The interesting case is when $r > 0$.

We first consider the case when $r > 1$. Note that $\lim_{x \to 0} G''_s(x) = \infty$. To see this, suppose for contradiction that this limit is finite. We may write

$$\lim_{x \to 0} G''_s(x) = \lim_{x \to 0} \frac{x G''_s(x)}{x} = \lim_{x \to 0} x G''_s(x) + G''_s(x)$$

by L’Hôpital’s rule, so $\lim_{x \to 0} x G'''_s(x) = 0$, contradicting that $\lim_{x \to 0} x r G'''_s(x) \neq 0$. 

280
Now, the fact that \( \lim_{x \to 0} G''_s(x) = \infty \) lets us apply L’Hôpital’s rule:

\[
\lim_{x \to 0} x^{r-1} G''_s(x) = \lim_{x \to 0} \frac{G''_s(x)}{x^{1-r}} = \lim_{x \to 0} \frac{G'''_s(x)}{(1-r)x^{-r}} = \frac{1}{1-r} \lim_{x \to 0} x^r G'''_s(x) = \frac{k}{1-r}.
\]

This means that

\[
\lim_{x \to 0} x^{r-1} G''_s(x) = 1 - r
\]

so in particular, there exists \( \delta > 0 \) such that for all \( x \leq \delta \) we have \( x^{r-1} G''_s(x) \) is bounded on \([\delta, 1-\delta] \) by assumption.

To finish, let \( t \) as in Definition 3.3.1 equal \( 0 \).

3. Assume \( c \) is small enough that the following conditions are satisfied:

- \( \frac{|G'''_s(x)|}{G''_s(x)} \leq c^{-16} \) on \([\delta, 1-\delta]\).
- \( c^{-0.01} \geq r \).

Then the last condition of Definition 3.3.1 will be satisfied on \([\delta, 1-\delta]\); it will also be satisfied on \([c^{-3}, 1-\delta]\) for any \( c \) satisfying the second condition above, because on that interval we have

\[
\frac{|G'''_s(x)|}{G''_s(x)} \leq \frac{r}{x} \leq \frac{1}{c^{16}\sqrt{x(1-x)}} \cdot \frac{c^{16}r}{\sqrt{x}} \leq \frac{1}{c^{16}\sqrt{x(1-x)}} \cdot \frac{c^{16}r}{c^{15}} \leq \frac{1}{c^{16}\sqrt{x(1-x)}}.
\]

By symmetry of \( G''_s \) about \( \frac{1}{2} \) (and antisymmetry of \( G'''_s \)) we have that the condition also holds on \([1-\delta, 1-c^t]\), as desired.

Now we consider the case that \( r = 1 \). As above, we have \( \lim_{x \to 0} G''_s(x) = \infty \). Proceeding similarly, we have

\[
\lim_{x \to 0} \frac{G''_s(x)}{\ln x} = \lim_{x \to 0} \frac{G'''_s(x)}{x} = \lim_{x \to 0} xG'''_s(x) = k.
\]

This means that there exists \( \delta > 0 \) such that for all \( x \leq \delta \) we have \( x \ln x \frac{G'''_s(x)}{G''_s(x)} \in [0, 2] \) and so

\[
\frac{|G'''_s(x)|}{G''_s(x)} \leq \frac{2}{x \ln x}. \quad \text{We finish as before.}
\]

Finally, consider the case that \( 0 < r < 1 \). Let \( a \) be a lower bound on \( G''_s \), as in the statement
of Claim A.2.2. It suffices to show that for for \( c \) small enough, we have \( |G''_s(x)| \leq \frac{a}{c^{16}\sqrt{x}} \) on \([c^{-3}, 1 - c^{-3}]\). Let \( \delta \) be such that \( x^c G''_s(x) \in [k - 1, k + 1] \) for all \( x \leq \delta \). On \([c^{-3}, \delta]\) we have

\[
|G''_s(x)| \leq \frac{|k| + 1}{x^c} \leq \frac{a}{c^{16}\sqrt{x}} \cdot \frac{|k| + 1}{ac^{-16}\sqrt{x}} \leq \frac{a}{c^{16}\sqrt{x}} \cdot \frac{|k| + 1}{ac^{-16} \cdot c^{15}}
\]

if \( c \) is small enough that \( c^{-0.1} \geq \frac{|k| + 1}{a} \). (As before, we also need to make sure that \( c \) is small enough that the condition is satisfied on \([\delta, 1 - \delta]\).) This concludes the proof.

\[ \square \]

### A.3 Details omitted from Section 3.3

#### A.3.1 Details omitted from Section 3.3.2

**Claim 3.3.5.** Let \( \Delta_{n+1}(q) := \mathbb{E} [G_s(Q(n + 1)) \mid Q(n) = q] - G_s(q) \) be the expected increase in the expert’s score (not counting the paid cost \( c \)) from the \((n + 1)\)th flip of the coin, given current belief \( Q(n) = q \). Then there exist \( c_1, c_2 \in [q - 1/n, q + 1/n] \) such that:

\[
\Delta_{n+1} = \frac{q \cdot (1-q)}{2(n+3)^2} (q \cdot G''_s(c_1) + (1-q) \cdot G''_s(c_2))
\]

**Proof.** By Lemma 3.2.7, we have

\[
\Delta_{n+1} = \frac{n+1}{n+2} G_s \left( \frac{h+2}{n+3} \right) + \frac{n-h+1}{n+2} G_s \left( \frac{h+1}{n+3} \right) - G_s \left( \frac{h+1}{n+2} \right).
\]

Since \( G_s \) is twice differentiable, we may use Taylor’s approximation theorem to write

\[
G_s \left( \frac{h+1}{n+3} \right) = G_s \left( \frac{h+1}{n+2} \right) + \left( \frac{h+1}{n+3} - \frac{h+1}{n+2} \right) G'_s \left( \frac{h+1}{n+2} \right) + \frac{1}{2} \left( \frac{h+1}{n+3} - \frac{h+1}{n+2} \right)^2 G''_s(c_1)
\]

for some \( c_1 \in \left[ \frac{h+1}{n+3}, \frac{h+1}{n+2} \right] \). Similarly we have

\[
G_s \left( \frac{h+2}{n+3} \right) = G_s \left( \frac{h+1}{n+2} \right) + \left( \frac{h+2}{n+3} - \frac{h+1}{n+2} \right) G'_s \left( \frac{h+1}{n+2} \right) + \frac{1}{2} \left( \frac{h+2}{n+3} - \frac{h+1}{n+2} \right)^2 G''_s(c_1)
\]

282
for some \( c_2 \in \left[ \frac{h+1}{n+2}, \frac{h+2}{n+3} \right] \). When we plug these expressions into the formula for \( \Delta_{n+1} \) above, the zeroth- and first-order terms cancel. We are left with

\[
\Delta_{n+1} = \frac{n - h + 1}{n + 2} \cdot \frac{1}{2} \left( \frac{h + 1}{n + 3} - \frac{h + 1}{n + 2} \right)^2 G''_s(c_1) + \frac{h + 1}{n + 2} \cdot \frac{1}{2} \left( \frac{h + 2}{n + 3} - \frac{h + 1}{n + 2} \right)^2 G''_s(c_2)
\]

\[
= \frac{(h + 1)^2(n - h + 1)}{2(n + 2)^3(n + 3)^2} G''_s(c_1) + \frac{(h + 1)(n - h + 1)^2}{2(n + 2)^3(n + 3)^2} G''_s(c_2)
\]

\[
= \frac{q(1 - q)}{2(n + 3)^2} \left( q G''_s(c_1) + (1 - q) G''_s(c_2) \right).
\]

Note that \( |c_1 - q| \leq \frac{h+1}{n+2} - \frac{h+1}{n+3} \leq \frac{1}{n} \), so \( c_1 \in \left[ q - \frac{1}{n}, q + \frac{1}{n} \right] \), and similarly for \( c_2 \). This completes the proof. \( \square \)

**Claim 3.3.6.** For all \( s \) such that \( G''_s \) is bounded away from zero, there exists \( \alpha, c_0 \) such that the expert is guaranteed to flip the coin at least \( \frac{1}{\alpha c^{1/3}} \) times for all \( c \leq c_0 \) (no matter the true bias).

**Proof.** Suppose that \( G''_s(x) \geq a \) for all \( x \in (0, 1) \). By Claim 3.3.5 we have

\[
\Delta_{n+1} \geq \frac{q(1 - q)a_1}{2(n + 3)^2}.
\]

Now, we have that \( \frac{1}{n+2} \leq q \leq \frac{n-1}{n+2} \), and \( q(1 - q) \) decreases as \( q \) gets farther from \( \frac{1}{2} \). This means that

\[
\Delta_{n+1} \geq \frac{n+1}{(n+2)^2} a \geq \frac{a_1}{72(n + 1)^3}.
\]

Therefore, if \( \Delta_{n+1} < c \) then \( n + 1 > \frac{1}{(\frac{2}{3})^{1/3} c^{1/3}} \), so \( n > \frac{1}{\alpha c^{1/3}} \) for some \( \alpha \) (not to be confused with \( a \)), if \( c \) is small enough. \( \square \)

### A.3.2 Details omitted from Section 3.3.3

**Claim 3.3.7.** The exists a sufficiently large \( N_0 \) such that for all \( N \geq N_0 \): if \( \Omega_N \) holds, then

\[
|Q_p(n) - p| \leq \frac{\sqrt{p(1-p)}}{n^{0.49}} \quad \text{for all } n \geq N \text{ and } p \in [1/n, 1 - 1/n].
\]

283
Proof. Let $1 \leq j_p \leq n - 2$ be such that $\frac{j_p}{n} \leq p \leq \frac{j_p + 1}{n}$. We have

$$
\frac{j_p}{n} - \frac{\sqrt{\frac{j_p(n - j_p)}{2n^{1.49}}}}{2n^{1.49}} \leq Q_{j_p/n}(n) \leq Q_p(n) \leq Q_{(j_p + 1)/n}(n) \leq \frac{j_p + 1}{n} + \frac{\sqrt{(j_p + 1)(n - 1 - j_p)}}{2n^{1.49}}
$$

so

$$
|Q_p(n) - p| \leq \max \left( \frac{1}{n} + \frac{1}{2n^{1.49}} \max \left( \sqrt{\frac{j_p(n - j_p)}{2n^{1.49}}} - \sqrt{(j_p + 1)(n - 1 - j_p)} \right) \right).
$$

For fixed $n$ and for $\frac{1}{n} \leq p \leq 1 - \frac{1}{n}$, this maximum divided by $n\sqrt{p(1 - p)}$ is maximized when $p = \frac{1}{n}$ (or $p = 1 - \frac{1}{n}$), in which case the ratio is $\sqrt{\frac{2(n - 2)}{n - 1}} \leq \sqrt{2}$. Therefore we have

$$
\frac{|Q_p(n) - p|}{\sqrt{p(1 - p)}} \leq \frac{1}{n\sqrt{p(1 - p)}} + \frac{n\sqrt{2}}{2n^{1.49}} \leq \frac{1}{n^{0.49}}
$$

for $n$ large enough. (Here we again use that $p \geq \frac{1}{n}$, so $\sqrt{p(1 - p)}$ is minimized at $p = \frac{1}{n}$.)

Claim 3.3.8.

$$
Pr \left[ \bar{\Omega}_N \right] = O \left( e^{-N^{0.01}} \right).
$$

Proof. We have

$$
Pr \left[ \bar{\Omega}_N \right] \leq \sum_{n=N}^{\infty} \sum_{j=1}^{n-1} Pr \left[ \left| Q_{j/n}(n) - \frac{j}{n} \right| > \frac{\sqrt{j(n-j)}}{2n^{1.49}} \right].
$$

Now, let $\mathcal{H}_{j/n}(n)$ be the fraction of the first $n$ coin flips that were heads (so $\mathcal{H}_{j/n}(n)$ is an average of $n$ i.i.d. Bernoulli random variables that are 1 with probability $\frac{j}{n}$). Note that $Q_{j/n}(n)$ is within $\frac{1}{n}$ of $\mathcal{H}_{j/n}(n)$, and for large $n$ we have $\frac{1}{n} \leq \frac{\sqrt{j(n-j)}}{2n^{1.49}}$ for all $j$. This means that for large $n$, by the triangle inequality we have that if $|\mathcal{H}_{j/n}(n) - \frac{j}{n}| > \frac{\sqrt{j(n-j)}}{n^{1.49}}$ then $|Q_{j/n}(n) - \frac{j}{n}| > \frac{\sqrt{j(n-j)}}{2n^{1.49}}$. Therefore, for large $N$ we have

$$
Pr \left[ \bar{\Omega}_N \right] \leq \sum_{n=N}^{\infty} \sum_{j=1}^{n-1} Pr \left[ |\mathcal{H}_{j/n}(n) - \frac{j}{n}| > \frac{\sqrt{j(n-j)}}{n^{1.49}} \right].
$$
We bound each of these probabilities. Recall the following version of the Chernoff bound: for 
\(0 \leq \delta \leq 1\), if \(X = \sum_{i=1}^{n} X_i\) is a sum of i.i.d. Bernoulli random variables with \(\mathbb{E}[X] = \mu\), then

\[
\Pr[|X - \mu| \geq \delta \mu] \leq 2e^{-\mu\delta^2/3}.
\]

We apply this to our random variables (so \(X = nH_j/n(n)\) and \(\mu = j\)). Assume \(j \leq n/2\). Let \(\delta = n^{-49/2} \sqrt{n-j}/j\). Then

\[
\Pr \left| \frac{X}{n} - \frac{j}{n} \right| \geq \frac{\sqrt{j(n-j)}}{n^{1.49}} \leq 2e^{-n^{0.02}/6}.
\]

If \(j \geq n/2\), a symmetry argument yields the same result. Therefore, for sufficiently large \(N\) we have

\[
\Pr \left[ \Omega_N \right] \leq \sum_{n=N}^{\infty} \sum_{j=1}^{n-1} 2e^{-n^{0.02}/6} \leq \sum_{n=N}^{\infty} 2ne^{-n^{0.02}/6} \leq \sum_{n=N}^{\infty} e^{-n^{0.02}/7} = e^{-N^{0.02}/7} \sum_{n=N}^{\infty} e^{-n^{0.02}-n^{0.02}}/7 \leq \sum_{n=N}^{\infty} e^{-n^{0.02}/14} \leq e^{-n^{0.02}/7} \sum_{n=N}^{\infty} \frac{14^{100} \cdot 100!}{n^2}.
\]

The last step comes from observing that \(e^x \geq \frac{x^{100}}{100!}\) for positive \(x\) and plugging in \(x = \frac{n^{0.02}}{14}\). Now, this summand is bounded by a constant, since \(\sum_{n=1}^{\infty} \frac{1}{n^2}\) converges, and so we have

\[
\Pr \left[ \Omega_N \right] \leq O \left( e^{-N^{0.02}/7} \right) \leq O \left( e^{-N^{0.01}} \right),
\]

as desired.

\(\square\)

### A.3.3 Details omitted from Section 3.3.4

**Proposition 3.3.9.** Assume that \(\Omega_N\) holds for some \(N\), and let \(t\) be as in Definition 3.3.1. There exists a constant \(\gamma\) and cost \(c_0 > 0\) such that for all \(c \leq c_0\) and all \(p \in [2c, 1 - 2c]\), we have

\[
\sqrt{\frac{p(1-p)G''(p)}{2c}} (1 - \gamma c^{1/300}) \leq n_{\text{stop}} \leq \sqrt{\frac{p(1-p)G''(p)}{2c}} (1 + \gamma c^{1/300}).
\]

**Proof.** We assume for convenience that \(t < 0.3\) (which is safe, as Definition 3.3.1 holds for all
\( t \leq t' \) whenever it holds for \( t' \).

Fix \( p \in [2c^t, 1 - 2c^t] \). Let \( \alpha \) be as in Claim 3.3.6. As before, let \( Q(n) \) be the predictor’s estimate for the bias of the coin after \( n \) flips. Then for \( c \) small enough that \( \frac{1}{\alpha c^{1/3}} \geq N \) and \( c^{1/30} \leq \frac{2}{\alpha} \), for \( n \geq \frac{1}{\alpha c^{1/3}} \), we have

\[
 p - \sqrt{p(1 - p)(\alpha c^{1/3})^{49}} \leq Q(n) \leq p + \sqrt{p(1 - p)(\alpha c^{1/3})^{49}}.
\]

This follows from Claim 3.3.7, noting that if \( c^{1/30} \leq \frac{2}{\alpha} \) and \( n \geq \frac{1}{\alpha c^{1/3}} \) then \( \frac{1}{n} \leq p \leq 1 - \frac{1}{n} \).

Now, recall Claim 3.3.5:

\[
\Delta_{n+1} = \frac{Q(n)(1 - Q(n))}{2(n + 3)^2} (Q(n)G''_s(c_1) + (1 - Q(n))G''_s(c_2))
\]

for some \( c_1, c_2 \in [Q(n) - \frac{1}{n}, Q(n) + \frac{1}{n}] \). In the remainder of this proof, what we essentially argue is that \( G''_s \) on this interval is not too far from \( G''_s(p) \), because of our bound on \( Q(n) \) as \( p \) plus or minus a small quantity.

We ask: for a given (possibly negative) \( \epsilon \), how far from \( G''_s(p) \) can \( G''_s(p + \epsilon) \) be? Well, since \( G''_s \) is integrable, we have

\[
\left| G''_s(p + \epsilon) - G''_s(p) \right| = \left| \int_p^{p+\epsilon} G''_s(x) \, dx \right| \leq \left| \int_p^{p+\epsilon} \left| G''_s(x) \right| \, dx \right|.
\]

Now, since \( s \) is respectful we have that for \( c \) small enough, if \( p, p + \epsilon \in [c^t, 1 - c^t] \) then

\[
\left| G''_s(x) \right| \leq \frac{1}{c^{16} \sqrt{x(1-x)}} G''_s(x) \leq \frac{1}{c^{16} \sqrt{\hat{\rho}(1-\hat{\rho})}} G''_s(x),
\]

where \( \hat{\rho} \) is defined to be the number on the interval between \( p \) and \( p + \epsilon \) minimizing \( \sqrt{x(1-x)} \) (i.e. farthest from \( \frac{1}{2} \)).

Define \( r := \frac{1}{c^{16} \sqrt{\hat{\rho}(1-\hat{\rho})}} \). Then \( \left| G''_s(x) \right| \leq r G''_s(x) \). Below, we will use this fact to prove the following claim.
Claim A.3.1. \(|G''_s(p + \epsilon) - G''_s(p)| \leq G''_s(p)(e^{\epsilon} - 1)|

Assuming the claim for now: how large of an \(\epsilon\) do we care about? The farthest that \(c_1\) and \(c_2\) can be from \(p\) is
\[
\sqrt{p(1-p)}(\alpha c^{1/3})^{49} + \alpha c^{1/3} \leq 2\sqrt{p(1-p)}(\alpha c^{1/3})^{49},
\]
for small \(c\). (This is because we assumed for convenience that \(t < 0.3\), which means that \(p(1-p) \geq c^{-3}\), so \(\alpha c^{1/3} \leq \sqrt{p(1-p)}(\alpha c^{1/3})^{49}\).) Therefore, by Claim A.3.1 we have
\[
|G''_s(p) - G''_s(c_1)|, |G''_s(p) - G''_s(c_2)| \leq G''_s(p)(e^{2\sqrt{p(1-p)}(\alpha c^{1/3})^{49}} - 1)
\]
\[
= G''_s(p) \exp \left( \frac{2\sqrt{p(1-p)}(\alpha c^{1/3})^{49}}{c^{16} \sqrt{\hat{p}(1-\hat{p})}} \right) - G''_s(p)
\]
where \(\hat{p}\) is either \(p \pm 2\sqrt{p(1-p)}(\alpha c^{1/3})^{49}\), whichever is farther from \(\frac{1}{2}\). It is easy to check\(^2\) that for small enough \(c\) we have that \(\frac{\sqrt{p(1-p)}}{\sqrt{p(1-\hat{p})}} \leq 2\), and so we have
\[
|G''_s(p) - G''_s(c_1)|, |G''_s(p) - G''_s(c_2)| \leq G''_s(p) \exp \left( \frac{1}{2} \beta c^{1/300} \right) - G''_s(p) \leq \beta c^{1/300} G''_s(p)
\]
for small enough \(c\), where \(\beta = 8\alpha^{49}\). (Here we use that \(e^x \leq 1 + 2x\) for small positive \(x\).) It follows, then, by Claim 3.3.5, that
\[
\frac{Q(n)(1 - Q(n))}{2(n + 3)^2} G''_s(p)(1 - \beta c^{1/300}) \leq \Delta_{n+1} \leq \frac{Q(n)(1 - Q(n))}{2(n + 3)^2} G''_s(p)(1 + \beta c^{1/300}).
\]
Note that since \(Q(n) \geq p - \sqrt{p(1-p)}(\alpha c^{1/3})^{49}\) and \(1 - Q(n) \geq 1 - p - \sqrt{p(1-p)}(\alpha c^{1/3})^{49}\), we may write
\[
Q(n)(1 - Q(n)) \geq p(1 - p)(1 + (\alpha c^{1/3})^{98}) - \sqrt{p(1-p)}(\alpha c^{1/3})^{49} = p(1 - p) \left( 1 - \frac{(\alpha c^{1/3})^{49}}{\sqrt{p(1-p)}} \right)
\]
\(^2\)Without loss of generality assume \(p \leq \frac{1}{2}\), so \(\hat{p} = p - 2\sqrt{p(1-p)}(\alpha c^{1/3})^{49}\). Then \(\frac{\sqrt{p(1-p)}}{\sqrt{p(1-\hat{p})}} \leq \frac{p(1-p)}{p(1-p)} \leq \frac{p}{\hat{p}}\), so it suffices to show that \(2\sqrt{p(1-p)}(\alpha c^{1/3})^{49} \leq p\). This is indeed the case, as \(p \geq c^{-3} \geq 4(\alpha c^{1/3})^{-49}\) for small \(c\), so \(2\sqrt{p(\alpha c^{1/3})^{49}} \leq p\).
\[ \geq p(1-p) \left( 1 - \frac{2(\alpha c^{1/3})^{49}}{e^{t/2}} \right) \geq p(1-p)(1 - \alpha^{49} c^{.01}) \]

for \( c \) small enough that the second-to-last step holds. (In the last step we use that \( t < 0.3 \).) A similar calculation shows that \( Q(n)(1 - Q(n)) \leq p(1-p)(1 + \alpha^{49} c^{.01}) \) for \( c \) small enough.\(^3\) Also note that \( n^2 \leq (n + 3)^2 \leq n^2(1 + 4\alpha c^{1/3})^2 \). Putting these approximations all together, we note that the \( c^{1/300} \) approximation is the dominant one, which means that there is a constant \( \gamma \) such that

\[
\frac{p(1-p)}{2n^2} G_s''(p)(1 - \gamma c^{1/300}) \leq \Delta_{n+1} \leq \frac{p(1-p)}{2n^2} G_s''(p)(1 + \gamma c^{1/300}). \tag{A.2}
\]

Therefore, since the expert stops flipping when \( \Delta_{n+1} < c \), we have

\[
\sqrt{\frac{p(1-p) G_s''(p)}{2c}} (1 - \gamma c^{1/300}) \leq n_{\text{stop}} \leq \sqrt{\frac{p(1-p) G_s''(p)}{2c}} (1 + \gamma c^{1/300}).
\]

This holds for any \( p \) such that \( p \pm 2\sqrt{p(1-p)}(\alpha c^{1/3})^{49} \in [c', 1 - c'] \); a sufficient condition is \( p \in [2c', 1 - 2c'] \). \( \square \)

**Proof of Claim A.3.1.** We prove this for positive \( \epsilon \). The result then follows for negative \( \epsilon \) because if some \( G_s \) is a counterexample for some negative \( \epsilon \), then a function \( G_\tilde{s} \) defined so that \( G_\tilde{s}''(p+x) := -G''(p-x) \) for \( x \in [0, -\epsilon] \) serves as a counterexample for \(-\epsilon \). (This is because \( G_\tilde{s}''(p+x) = G_s''(p-x) \) for all \( x \in [0, -\epsilon] \), by the fundamental theorem of calculus.) Additionally, we may assume that \( G_s''(p) = 1 \), because if there is a counterexample function \( G_s \) to the claim then \( G_\tilde{s}(x) := \frac{G_s(x)}{G_s(p)} \) also serves as a counterexample.

We prove that

\[ e^{-r\epsilon} - 1 \leq G_s''(p + \epsilon) - 1 \leq e^{r\epsilon} - 1. \]

The left inequality suffices because \( 1 - e^{-x} \leq e^x - 1 \) for all \( x \), so in particular \( 1 - e^{r\epsilon} \leq e^{-r\epsilon} - 1 \).

We begin with the right inequality. Suppose for contradiction that \( G_s''(p + \epsilon) > e^{r\epsilon} \). Let \( S \) be the set of points in \([p, p + \epsilon]\) where \( G_s''(x) > e^{r(x-p)} \). Since \( S \) contains \( p + \epsilon \), it is nonempty; let

\(^3\)An extra \( \alpha^{49} c^{49/3} \) appears, but this term is dominated by \( \alpha^{49} c^{.01} \) for small \( c \).
\[ p_1 = \inf S. \] Since \( G''_s \) is continuous, we have \( G''_s(p_1) - e^{r(p_1-p)} = 0. \) Pick \( \delta > 0 \) small enough that the set \( T \) of points \( x \in [p_1, p_1 + \min(\epsilon, \frac{1}{3\delta})] \) with \( G''_s(x) - e^{r(x-p)} > \delta \) is nonempty. Let \( p_2 = \inf T, \) so \( G''_s(p_2) - e^{r(p_2-p)} = \delta. \) Note that

\[
\delta = G''_s(p_2) - e^{r(p_2-p)} = G''_s(p_1) - e^{r(p_1-p)} + \int_{p_1}^{p_2} \frac{d}{dx} \left( G''_s(x) - e^{r(x-p)} \right) dx = \int_{p_1}^{p_2} \left( G''_s(x) - re^{r(x-p)} \right) dx.
\]

It follows that \( G''_s(p_3) - re^{r(p_3-p)} \geq \frac{\delta}{p_3-p_1} \geq 2r\delta \) for some \( p_3 \in [p_1, p_2]. \) (Otherwise the value of the integral would be at most the integral of \( 2r\delta \) from \( p_1 \) to \( p_2 \), which is at most \( \frac{2}{3}\delta \), since \( p_2 - p_1 \leq \frac{1}{3\delta} \).) Therefore, because \( |G'''_s(x)| \leq rG''_s(x) \) for all \( x \in [p, p+\epsilon] \), we have that

\[
rG''_s(p_3) - re^{r(p_3-p)} \geq 2r\delta
\]

so \( G''_s(p_3) - e^{r(p_3-p)} \geq 2\delta. \) But then we have that \( p_2 < p_3 \) and \( p_3 \in T, \) contradicting the definition of \( p_2 \) as the infimum of \( T. \)

The proof of the left inequality above proceeds similarly, but is not exactly analogous. Suppose for contradiction that \( t := G''_s(p+\epsilon) - e^{-re} < 0. \) Define \( p_1 \) to be the supremum of points in \( [p, p+\epsilon] \) where \( G''_s(x) \geq e^{-r(x-p)} \) (so \( G''_s(p_1) = e^{-r(p_1-p)} \)). Then \( G''_s(x) - e^{-r(x-p)} \) is zero at \( x = p_1 \) and \( t \) at \( x = p + \epsilon, \) so

\[
\int_{p_1}^{p+\epsilon} (G''_s(x) + re^{-r(x-p)}) dx = t < 0,
\]

which means that for some \( p_2 \in [p_1, p+\epsilon] \) we have that \( G'''_s(p_2) + re^{-r(p_2-p)} \leq \frac{t}{2(p+\epsilon-p_1)} \) (otherwise the value of the integral would be at least \( \frac{t}{2} \)). Since \( |G'''_s(x)| \leq rG''_s(x) \) for all \( x \in [p, p+\epsilon], \) we have that

\[
-rG''_s(p_2) + re^{-r(p_2-p)} \leq \frac{t}{2(p+\epsilon-p_1)} < 0,
\]

so \( G''_s(p_2) > e^{-r(p_2-p)}. \) This is a contradiction, since on the one hand we have \( p_2 > p_1, \) but on the other hand \( p_1 \) was defined as the supremum of points where \( G''_s(x) \geq e^{-r(x-p)}. \) This completes the proof. \( \Box \)
Lemma 3.3.10. Let $\ell \geq 1$ and $\mu_\ell := \frac{2^{\ell/2} \Gamma(\frac{\ell+1}{2})}{\sqrt{\pi}}$ be the $\ell$th moment of a standard Gaussian. Let $N = \frac{1}{ac^{1/3}}$ (so $N$ is implicitly a function of $c$). For all $p \in [2c^\ell, 1 - 2c^\ell]$ we have

$$(1 - o(1)) \cdot \mu_\ell \cdot \left( \frac{2p(1-p)}{G''_s(p)} \right)^{\ell/4} \leq c^{-\ell/4} \cdot \mathbb{E} \left[ (\text{Err}_c(p))^\ell \mid \Omega_N \right] \leq (1 + o(1)) \cdot \mu_\ell \cdot \left( \frac{2p(1-p)}{G''_s(p)} \right)^{\ell/4}$$

where the $o(1)$ term is a function of $c$ (but not $p$) that approaches zero as $c$ approaches zero.

A note on terminology: We will sometimes speak of limits that hold “uniformly over $p$.” If we say that a function $g(c)$ is $o(h(c))$ uniformly over $p$, we mean that $g$ and $h$ are implicitly functions of $p$ as well, and that $\frac{h(x)}{g(x)}$ approaches zero uniformly in $p$ (i.e. for all $\epsilon$ there exists $c_\epsilon$ such that for all $c \leq c_\epsilon$, we have $\frac{h(x)}{g(x)} < \epsilon$ for all (relevant) values of $p$). So for instance, the $o(1)$ in the statement of Lemma 3.3.10 is uniform in $p$.

Proof. Fix any $c$ and $p \in [2c^\ell, 1 - 2c^\ell]$. Let

$$n_0 := \sqrt{\frac{p(1-p)G''_s(p)}{2c}(1 - \gamma c^{1/300})}.$$  

Let $d_1$ be the expert’s error after $n_0$ flips, i.e. $|Q(n_0) - p|$. Let $d_2$ be the distance from their guess after $n_0$ flips to their guess after $n_{\text{stop}}$ flips. Then the expert’s error after $n_{\text{stop}}$ flips lies between $d_1 - d_2$ and $d_1 + d_2$ by the triangle inequality. That is, we have

$$\max(0, d_1 - d_2) \leq \text{Err}_c(p) \leq d_1 + d_2,$$

so

$$\mathbb{E} \left[ \max(0, d_1 - d_2)^\ell \mid \Omega_N \right] \leq \mathbb{E} \left[ (\text{Err}_c(p))^\ell \mid \Omega_N \right] \leq \mathbb{E} \left[ (d_1 + d_2)^\ell \mid \Omega_N \right]$$
which means that

\[
\frac{\mathbb{E} \left[ \max(0, d_1 - d_2) \ell \mid \Omega_N \right]}{\mathbb{E} \left[ d_1^\ell \mid \Omega_N \right]} \leq \frac{\mathbb{E} \left[ (\text{Err}_c(p))^\ell \mid \Omega_N \right]}{\mathbb{E} \left[ d_1^\ell \mid \Omega_N \right]} \leq \frac{\mathbb{E} \left[ (d_1 + d_2)^\ell \mid \Omega_N \right]}{\mathbb{E} \left[ d_1^\ell \mid \Omega_N \right]}.
\]

Below, we will prove the following facts:

**Claim A.3.2.** We have

\[
\lim_{c \to 0} \frac{\mathbb{E} \left[ d_2^\ell \mid \Omega_N \right]}{\mathbb{E} \left[ d_1^\ell \mid \Omega_N \right]} = 0
\]

uniformly over \( p \in [2c^\ell, 1 - 2c^\ell] \). That is, for all \( \epsilon \) there exists \( c_\epsilon \) such that for all \( c < c_\epsilon \), the fraction above is less than \( \epsilon \) for all \( p \in [2c^\ell, 1 - 2c^\ell] \).

**Proposition A.3.3.** Let \( X_{c,p} \) and \( Y_{c,p} \) be random variables taking values in \([0, 1]\) for each real number \( c > 0 \) and \( p \in \mathcal{P}_c \) (some arbitrary set that depends on \( c \)). Let \( \ell > 0 \). If \( \lim_{c \to 0} \frac{\mathbb{E}[Y_{c,p}^\ell]}{\mathbb{E}[X_{c,p}^\ell]} = 0 \) uniformly over \( p \in \mathcal{P}_c \), then \( \lim_{c \to 0} \frac{\mathbb{E}[(X_{c,p} + Y_{c,p})^\ell]}{\mathbb{E}[X_{c,p}^\ell]} = 1 \) uniformly over \( p \in \mathcal{P}_c \). Separately, if \( \lim_{c \to 0} \frac{\mathbb{E}[X_{c,p}^\ell]}{\mathbb{E}[(X_{c,p} + Y_{c,p})^\ell]} = 1 \) uniformly over \( p \in \mathcal{P}_c \), then \( \lim_{c \to 0} \frac{\mathbb{E}[Y_{c,p}^\ell]}{\mathbb{E}[(X_{c,p} + Y_{c,p})^\ell]} = 0 \) uniformly over \( p \in \mathcal{P}_c \).

Now, by combining Claim A.3.2 with the first statement of Proposition A.3.3 (with \( X_{c,p} = d_1 \) and \( Y_{c,p} = d_2 \)), we have

\[
\lim_{c \to 0} \frac{\mathbb{E} \left[ (d_1 + d_2)^\ell \mid \Omega_N \right]}{\mathbb{E} \left[ d_1^\ell \mid \Omega_N \right]} = 1.
\]

By the second statement of Proposition A.3.3 (with \( X_{c,p} = \max(0, d_1 - d_2) \) and \( Y_{c,p} = d_1 - \max(0, d_1 - d_2) \)), we have

\[
\lim_{c \to 0} \frac{\mathbb{E} \left[ \max(0, d_1 - d_2)^\ell \mid \Omega_N \right]}{\mathbb{E} \left[ d_1^\ell \mid \Omega_N \right]} = 1.
\]

Note that the premise of the second statement of Proposition A.3.3 holds for these \( X_{c,p} \) and \( Y_{c,p} \), because \( Y_{c,p} = d_1 - \max(0, d_1 - d_2) \leq d_2 \) and so certainly if \( \lim_{c \to 0} \frac{\mathbb{E}[d_1^\ell \mid \Omega_N]}{\mathbb{E}[d_1^\ell \mid \Omega_N]} = 0 \) then

\[
\lim_{c \to 0} \frac{\mathbb{E} \left[ (d_1 - \max(0, d_1 - d_2))^\ell \mid \Omega_N \right]}{\mathbb{E} \left[ d_1^\ell \mid \Omega_N \right]} = 0.
\]
By the squeeze theorem, it follows that
\[
\lim_{c \to 0} \frac{\mathbb{E} \left[ (\text{Err}_c(p))^\ell \mid \Omega_N \right]}{\mathbb{E} \left[ d_1^\ell \mid \Omega_N \right]} = 1.
\]

Note that this limit holds uniformly over \( p \in [2c^\ell, 1 - 2c^\ell] \).

To complete the proof of Lemma 3.3.10, we use the following fact, which we will also prove later.

**Claim A.3.4.**
\[
\mathbb{E} \left[ d_1^\ell \mid \Omega_N \right] = \mu_\ell \left( \frac{p(1-p)}{n_0} \right)^{\ell/2} (1 + o(1))
\]

where the \( o(1) \) term is a function of \( c \) (but not \( p \)) that approaches zero as \( c \) approaches zero.

We have
\[
n_0 = \sqrt{\frac{p(1-p)G''_s(p)}{2c}} \left( 1 - \gamma c^{1/300} \right) = \sqrt{\frac{p(1-p)G''_s(p)}{2c}} (1 + o(1)).
\]

Therefore we have
\[
\mathbb{E} \left[ d_1^\ell \mid \Omega_N \right] = \mu_\ell \left( \frac{2c}{p(1-p)G''_s(p)} \right)^{\ell/4} (1 + o(1)),
\]
as desired.

We now return to the proofs that we deferred.

**Proof of Proposition A.3.3.** The condition that \( X_{c,p}, Y_{c,p} \in [0, 1] \) is simply a convenient one to guarantee that all relevant expectations are finite. Now, for any \( a \in (0, 1) \), we have
\[
\mathbb{E} \left[ (X_{c,p} + Y_{c,p})^\ell \right] = \int_0^\infty \Pr \left[ (X_{c,p} + Y_{c,p})^\ell > z \right] dz = \int_0^\infty \Pr \left[ X_{c,p} + Y_{c,p} > z^{1/\ell} \right] dz
\]
\[
\leq \int_0^\infty \left( \Pr \left[ X_{c,p} > (1-a)z^{1/\ell} \right] + \Pr \left[ Y_{c,p} > az^{1/\ell} \right] \right) dz
\]
\[
= \int_0^\infty \left( \Pr \left[ \frac{X_{c,p}^\ell}{(1-a)^\ell} > z \right] + \Pr \left[ \frac{Y_{c,p}^\ell}{a^\ell} > z \right] \right) dz = \mathbb{E} \left[ X_{c,p}^\ell \right] \frac{(1-a)^\ell}{a^\ell} + \mathbb{E} \left[ Y_{c,p}^\ell \right], \quad (A.3)
\]
where the inequality follows by a union bound.

We start with the first statement. Dividing Equation A.3 by $\mathbb{E} \left[ X_{c,p}^\ell \right]$, we have

$$1 \leq \frac{\mathbb{E} \left[ (X_{c,p} + Y_{c,p})^\ell \right]}{\mathbb{E} \left[ X_{c,p}^\ell \right]} \leq \frac{1}{(1-a)^\ell} + \frac{\mathbb{E} \left[ Y_{c,p}^\ell \right]}{\mathbb{E} \left[ X_{c,p}^\ell \right]} \cdot \frac{1}{a^\ell}.$$ 

The limit of $\frac{\mathbb{E} \left[ Y_{c,p}^\ell \right]}{\mathbb{E} \left[ X_{c,p}^\ell \right]} \cdot \frac{1}{a^\ell}$ as $c$ approaches zero is 0 by assumption, so for $c$ small enough we have that $\frac{\mathbb{E} \left[ Y_{c,p}^\ell \right]}{\mathbb{E} \left[ X_{c,p}^\ell \right]} \cdot \frac{1}{a^\ell} \leq a$ for all $p \in \mathcal{P}_c$. In other words, for every $a$ there exists $c_a$ such that for all $c \leq c_a$ we have that $1 \leq \frac{\mathbb{E} \left[ (X_{c,p} + Y_{c,p})^\ell \right]}{\mathbb{E} \left[ X_{c,p}^\ell \right]} \leq \frac{1}{(1-a)^\ell} + a$. Since $\lim_{a \to 0} \frac{1}{(1-a)^\ell} + a = 1$, we have that for all $\varepsilon > 0$ there exists $c_\varepsilon$ such that for all $c \leq c_\varepsilon$ and $p \in \mathcal{P}_c$ we have that $1 \leq \frac{\mathbb{E} \left[ (X_{c,p} + Y_{c,p})^\ell \right]}{\mathbb{E} \left[ X_{c,p}^\ell \right]} \leq 1 + \varepsilon$.

This proves the first statement.

As for the second statement, we divide Equation A.3 by $\mathbb{E} \left[ (X_{c,p} + Y_{c,p})^\ell \right]$ to obtain

$$1 \leq \frac{1}{(1-a)\ell} \cdot \frac{\mathbb{E} \left[ X_{c,p}^\ell \right]}{\mathbb{E} \left[ (X_{c,p} + Y_{c,p})^\ell \right]} + \frac{1}{a^\ell} \cdot \frac{\mathbb{E} \left[ Y_{c,p}^\ell \right]}{\mathbb{E} \left[ (X_{c,p} + Y_{c,p})^\ell \right]}$$

so

$$(1-a)^\ell \left( 1 - \frac{1}{a^\ell} \cdot \frac{\mathbb{E} \left[ Y_{c,p}^\ell \right]}{\mathbb{E} \left[ (X_{c,p} + Y_{c,p})^\ell \right]} \right) \leq \frac{\mathbb{E} \left[ X_{c,p}^\ell \right]}{\mathbb{E} \left[ (X_{c,p} + Y_{c,p})^\ell \right]} \leq 1.$$ 

The limit of $\frac{\mathbb{E} \left[ Y_{c,p}^\ell \right]}{\mathbb{E} \left[ (X_{c,p} + Y_{c,p})^\ell \right]}$ as $c$ approaches zero is 0 by assumption, so for $c$ small enough we have that $\frac{1}{a^\ell} \cdot \frac{\mathbb{E} \left[ Y_{c,p}^\ell \right]}{\mathbb{E} \left[ (X_{c,p} + Y_{c,p})^\ell \right]} \leq a$ for all $p \in \mathcal{P}_c$. In other words, for every $a$ there exists $c_a$ such that for all $c \leq c_a$ we have that $(1-a)^{\ell+1} \leq \frac{\mathbb{E} \left[ X_{c,p}^\ell \right]}{\mathbb{E} \left[ (X_{c,p} + Y_{c,p})^\ell \right]} \leq 1$. Since $\lim_{a \to 0} (1-a)^{\ell+1} = 1$, we have that for all $\varepsilon > 0$ there exists $c_\varepsilon$ such that for all $c \leq c_\varepsilon$ and $p \in \mathcal{P}_c$ we have that $1 - \varepsilon \leq \frac{\mathbb{E} \left[ X_{c,p}^\ell \right]}{\mathbb{E} \left[ (X_{c,p} + Y_{c,p})^\ell \right]} \leq 1$.

This proves the second statement.

Proof of Claim A.3.4. We use the Berry-Esseen theorem, a result about the speed of convergence of a sum of i.i.d. random variables to a normal distribution.

Theorem A.3.5 (Berry-Esseen theorem). Let $X_1, \ldots, X_n$ be i.i.d. random variables with $\mathbb{E} \left[ X_1 \right] = 0$, $\mathbb{E} \left[ X_1^2 \right] \equiv \sigma^2 > 0$, and $\mathbb{E} \left[ |X_1|^3 \right] = \rho < \infty$. Let $Y = \frac{1}{n} \sum_i X_i$ and let $F$ be the CDF of $\frac{Y \sqrt{n}}{\sigma}$. Let
Let \( \varphi(x) \) be the standard normal distribution. Then for all \( x \) we have

\[
|F(x) - \varphi(x)| \leq \frac{C \rho}{\sigma^3 \sqrt{n}},
\]

for some universal constant \( C \) independent of \( n \) and the distribution of the \( X_i \).

Define \( X_i \) to be \( 1 - p \) if the expert flips heads (which happens with probability \( p \)) and \(-p\) if the expert flips tails (which happens with probability \( 1 - p \)). Let \( Y = \sum_i X_i \). Then \( \sigma = \sqrt{p(1 - p)} \) and \( \rho = p(1 - p) \left( p^2 + (1 - p)^2 \right) \leq p(1 - p) \). Plugging in these \( X_i \) and \( n = n_0 \) into the Berry-Esseen theorem, we have

\[
|F(x) - \varphi(x)| \leq \frac{C p(1 - p)}{(p(1 - p))^{3/2} \sqrt{n_0}} = \frac{C}{\sqrt{p(1 - p)n_0}}.
\]

Now, we want to approximate \( \mathbb{E} \left[ d_1^\ell \right] = \mathbb{E} \left[ |Q(n_0) - p|\ell \right] \). Note that \( Q(n_0) \) is within \( \frac{1}{n_0} \) of \( Y + p \), the number of heads flipped divided by \( n_0 \). This means that \( Y - \frac{1}{n_0} \leq Q(n_0) - p \leq Y + \frac{1}{n_0} \). For this reason, we focus on computing \( \mathbb{E} \left[ |Y|\ell \right] \) and subsequently correct for this small difference.

Observe that

\[
\mathbb{E} \left[ |Y|\ell \right] = \int_0^\infty \Pr \left[ |Y| > x \right] dx = \int_0^\infty \Pr \left[ |Y| > u \right] \cdot \ell u^{\ell - 1} du
\]

\[
= \int_0^\infty \left( \Pr \left[ Y > u \right] + \Pr \left[ Y < -u \right] \right) \cdot \ell u^{\ell - 1} du = \int_0^\infty \left( \Pr \left[ \frac{Y \sqrt{n_0}}{\sigma} > \frac{u \sqrt{n_0}}{\sigma} \right] + \Pr \left[ \frac{Y \sqrt{n_0}}{\sigma} < \frac{-u \sqrt{n_0}}{\sigma} \right] \right) \cdot \ell u^{\ell - 1} du
\]

\[
= \int_0^\infty \left( 1 - F \left( \frac{u \sqrt{n_0}}{\sigma} \right) + F \left( \frac{-u \sqrt{n_0}}{\sigma} \right) \right) \cdot \ell u^{\ell - 1} du.
\]

Now, observe on the other hand that

\[
\int_0^\infty \left( 1 - \varphi \left( \frac{u \sqrt{n_0}}{\sigma} \right) + \varphi \left( \frac{-u \sqrt{n_0}}{\sigma} \right) \right) \cdot \ell u^{\ell - 1} du = \mathbb{E} \left[ |Z|\ell \right],
\]

where \( Z \) is a random variable drawn from a normal distribution with mean zero and variance \( \frac{\sigma^2}{n_0} \).
Now we ask: how different is this second quantity (the integral involving $\varphi$) from the first one (the integral involving $F$)?

The answer is: not that different. Indeed, as we derived, $F\left(\frac{u\sqrt{n_0}}{\sigma}\right)$ is within $\frac{C}{\sqrt{p(1-p)n_0}}$ of $\varphi\left(\frac{u\sqrt{n_0}}{\sigma}\right)$ for all $u$. Furthermore, for any $r < \frac{1}{2}$, if $u \leq -\sqrt{p(1-p)}n_0^{-\alpha}$ then $F\left(\frac{u\sqrt{n_0}}{\sigma}\right)$ and $\varphi\left(\frac{u\sqrt{n_0}}{\sigma}\right)$ are both $e^{-n_0^{\Omega(1)}}$. (For $\varphi$ this follows by concentration of normal distributions; the claim for $F$ follows from Claim 3.3.8, realizing the fact that there is nothing special about the 0.49 in the exponent except that it is less than $\frac{1}{2}$.) Similarly, if $u \geq \sqrt{p(1-p)}n_0^{-\alpha}$ then both $F\left(\frac{u\sqrt{n_0}}{\sigma}\right)$ and $\varphi\left(\frac{u\sqrt{n_0}}{\sigma}\right)$ are exponentially close to 1. Finally, for $u \leq -1$ we have $F\left(\frac{u\sqrt{n_0}}{\sigma}\right) = 0$ and $\varphi\left(\frac{u\sqrt{n_0}}{\sigma}\right) = O(e^{-u^2n_0})$.

This means that $E \left[|Y|^\ell\right]$ is within

$$\int_0^{\sqrt{p(1-p)}n_0^{-\alpha}} \frac{2C}{\sqrt{p(1-p)n_0}} \cdot tu^{\ell-1} du + e^{-n_0^{\Omega(1)}} = 2C(p(1-p))^{(\ell-1)/2}n_0^{-r\ell-1/2} + e^{-n_0^{\Omega(1)}}$$

of $E \left[|Z|^\ell\right]$. Now, since $p(1-p) \geq c^\ell \geq c_3$ and $n_0 \geq \frac{1}{ac^{1/3}}$, we have that $p(1-p) \geq (an_0)^{-9}$. It is easy to check that setting any $r > \frac{1}{2} - \frac{1}{20\ell}$ shows that

$$E \left[|Y|^\ell\right] = E \left[|Z|^\ell\right] (1 + o(1))$$

where the $o(1)$ depends only on $c$, not on $p$. Note that $E \left[|Z|^\ell\right] = \Omega \left(\frac{p(1-p)}{n_0^{\ell}}\right) = \omega \left(\frac{1}{n_0^{\ell}}\right)$ uniformly over $p$. This means that $\frac{1}{n_0^{\ell}} = o \left(E \left[|Y|^\ell\right]\right)$ uniformly over $p$.

Now, recall that $Y - \frac{1}{n_0} \leq Q(n_0) - p \leq Y + \frac{1}{n_0}$, so

$$\max \left(0, |Y| - \frac{1}{n_0}\right) \leq |Q(n_0) - p| \leq |Y| + \frac{1}{n_0},$$

and thus

$$E \left[\max \left(0, |Y| - \frac{1}{n_0}\right)^\ell\right] \leq E \left[|Q(n_0) - p|^\ell\right] = E \left[d_1\right] \leq E \left[\left(|Y| + \frac{1}{n_0}\right)^\ell\right].$$
By the first statement of Proposition A.3.3 (with $X_{c,p} = |Y|$ and $Y_{c,p} = \frac{1}{n_0}$), we have that
\[
\lim_{c \to 0} \frac{\E \left[ (|Y| + \frac{1}{n_0})^\ell \right]}{\E \left[ |Y|^\ell \right]} = 1
\]
uniformly over $p$. By the second statement (with $X_{c,p} = \max \left( 0, |Y| - \frac{1}{n_0} \right)$ and $Y_{c,p} = |Y| - \max \left( 0, |Y| - \frac{1}{n_0} \right)$), we have that
\[
\lim_{c \to 0} \frac{\E \left[ \max \left( 0, |Y| - \frac{1}{n_0} \right)^\ell \right]}{\E \left[ |Y|^\ell \right]} = 1
\]
uniformly over $p$. (The premise of the second statement is satisfied because $|Y| - \max \left( 0, |Y| - \frac{1}{n_0} \right) \leq \frac{1}{n_0}$.) Therefore, the squeeze theorem tells us that
\[
\lim_{c \to 0} \frac{\E [d_1]}{\E \left[ |Y|^\ell \right]} = 1
\]
uniformly over $p$. Therefore, we have $\E [d_1] = \E \left[ |Z|^\ell \right] (1 + o(1))$ where the $o(1)$ term only depends on $c$.

Finally, note that
\[
\E [d_1] = \E [d_1 \mid \Omega_N] \Pr [\Omega_N] + \E [d_1 \mid \overline{\Omega_N}] \Pr [\overline{\Omega_N}].
\]
Since $0 \leq \E [d_1 \mid \overline{\Omega_N}] \leq 1$ and $\Pr [\overline{\Omega_N}] = e^{-\frac{\Omega(1)}{n_0}}$ (where the $\Omega(1)$ does not depend on $p$), we have that $\E [d_1 \mid \Omega_N]$ is within $e^{-\frac{\Omega(1)}{n_0}}$ of $\E [d_1]$. Applying Proposition A.3.3 in the same way as earlier, we find that $\E [d_\ell^\ell \mid \Omega_N] = \E [d_\ell^\ell] (1 + o(1)) = \E \left[ |Z|^\ell \right] (1 + o(1))$. We know that $\E \left[ |Z|^\ell \right] = \mu_\ell \left( \frac{n_0(1-p)}{n_0} \right)^{\ell/2}$. This completes the proof. \hfill \Box

Proof of Claim A.3.2. Let $k = n_{\text{stop}} - n_0$. Define $\{Y_i\}_{i=0}^k$ as follows: $Y_0 = 0$ and for $i > 0$, $Y_i$ is either $Y_{i-1} + 1 - p$ (if the $n_0 + i$-th flip is heads, i.e. with probability $p$) or $Y_{i-1} - p$ (if the $n_0 + i$-th
flip is tails, i.e. with probability $1 - p$. Note that $\{Y_i\}$ is a martingale.

Now, observe that for any $0 \leq i \leq k$, we have

$$Q(n_0 + i) = \frac{(n_0 + 2)Q(n_0) + Y_i + pi}{n_0 + i + 2}.$$  

This is because $(n_0 + 2)Q(n_0)$ is one more than the number of heads in the first $n_0$ flips and $(n_0 + i + 2)Q(n_0 + i)$ is one more than the number of heads in the first $n_0 + i$ flips. Thus,

$$|Q(n_0 + i) - Q(n_0)| = \left| \frac{Y_i + i(p - Q(n_0))}{n_0 + i + 2} \right| \leq \frac{\max_i |Y_i| + k |p - Q(n_0)|}{n_0} = \max_i |Y_i| + k d_1.$$

Therefore we have

$$\mathbb{E} [d_1^\ell] \leq \mathbb{E} \left[ \max_{i \leq k} \left( |Q(n_0 + i) - Q(n_0)|^\ell \right) \right] \leq \mathbb{E} \left[ (\max_i |Y_i| + k d_1)^\ell \right] = \frac{2^\ell \mathbb{E} \left[ \left( \frac{\max_i |Y_i| + k d_1}{2} \right)^\ell \right]}{n_0^\ell}.$$  

Here, the last inequality follows from the fact that the arithmetic mean of $\max_i |Y_i|$ and $k d_1$ is less than or equal to the $\ell$-power mean (since $\ell \geq 1$).

Now, it is clear that $\frac{(2^{\ell-1}k)^\ell}{n_0^\ell} \mathbb{E} [d_1^\ell] = o(\mathbb{E} [d_1^\ell])$, since $k = o(n_0)$ by Proposition 3.3.9. We now show that $\frac{(2^{\ell-1}k)^\ell}{n_0^\ell} \mathbb{E} [\max_i |Y_i|^\ell] = o(\mathbb{E} [d_1^\ell])$. We make use of a tool called the Burkholder-Davis-Gundy inequality.

**Definition A.3.6.** Let $Y = \{Y_i\}_{i=0}^k$ be a martingale. The **quadratic variation** of $Y$, denoted $[Y]$, is equal to

$$[Y] = \sum_{i=1}^k (Y_i - Y_{i-1})^2.$$  

Note that $[Y]$ is a random variable, not a number.

**Theorem A.3.7** (Burkholder-Davis-Gundy inequality). Let $\ell \geq 1$. There is a constant $C_\ell$ such that
for every martingale \( Y = \{Y_i\}_{i=0}^k \) with \( Y_0 = 0 \), we have

\[
\mathbb{E} \left[ \left( \max_{i=0}^k |Y_i| \right)^\ell \right] \leq C_\ell \mathbb{E} \left[ |Y|^{\ell/2} \right].
\]

We wish to bound \( \mathbb{E} \left[ \left( \max_{i=0}^k |Y_i| \right)^\ell \right] \) above. To do so, we bound \( \mathbb{E} \left[ |Y|^{\ell/2} \right] \) above. Observe that \( |Y| \) is a sum of \( k \) independent random variables that are each either \( p^2 \) (with probability \( 1 - p \)) or \( (1 - p)^2 \) (with probability \( p \)). Thus, \( \mathbb{E} \left[ |Y| \right] = kp(1 - p) \equiv \mu \). Observe that

\[
\mathbb{E} \left[ |Y|^{\ell/2} \right] = \int_0^\infty \mathbb{P} \left[ |Y|^{\ell/2} \geq x \right] dx = \int_0^\infty \mathbb{P} \left[ |Y| \geq x^{2/\ell} \right] dx \leq (3\mu)^{\ell/2} + \int_{(3\mu)^{\ell/2}}^\infty \mathbb{P} \left[ |Y| \geq x^{2/\ell} \right] dx.
\]

The last line comes from a Chernoff bound. In particular, we have that \( \mathbb{P} \left[ |Y| \geq \mu(1 + \delta) \right] \leq e^{-\delta^2 \mu/(2+\delta)} \leq e^{-\delta \mu/2} \) for \( \delta \geq 2 \). Setting \( \delta = \frac{x^{2/\ell}}{\mu} - 1 \) gives us the expression above. Now, we can bound the integral as follows:

\[
\int_{(3\mu)^{\ell/2}}^\infty e^{-\frac{x^{2/\ell}}{2}} dx \leq (5\mu)^{\ell/2} - (3\mu)^{\ell/2} e^{-\mu} + (7\mu)^{\ell/2} - (5\mu)^{\ell/2} e^{-2\mu} + \cdots \equiv B(\mu).
\]

Note that \( B(\mu) \) is continuous, converges on \([0, \infty)\), and approaches zero as \( \mu \to \infty \). It follows that \( B(\mu) \) is bounded on \([0, \infty)\); in other words, our integral is \( O(1) \) (i.e. possibly depends on \( \ell \) but is at most a constant for fixed \( \ell \)). Therefore, we have

\[
\mathbb{E} \left[ \left( \max_{i=0}^k |Y_i| \right)^\ell \right] \leq \mathbb{E} \left[ |Y|^{\ell/2} \right] \leq 3^{\ell/2} (kp(1 - p))^{\ell/2} + O(1).
\]

Therefore we have

\[
\frac{2^{\ell-1}}{n_0^\ell} \mathbb{E} \left[ \left( \max_{i} |Y_i| \right)^\ell \right] = O \left( \left( \frac{p(1 - p)}{n_0} \right)^{\ell/2} \left( \frac{k}{n_0} \right)^{\ell/2} \right) = O \left( \left( \frac{k}{n_0} \right)^{\ell/2} \mathbb{E} \left[ d_1^\ell \right] \right) = o \left( \mathbb{E} \left[ d_1^\ell \right] \right).
\]

Therefore, we have that \( \mathbb{E} \left[ d_2^\ell \right] = o \left( \mathbb{E} \left[ d_1^\ell \right] \right) \). By the same reasoning as in the proof of Claim A.3.4
below, it follows that $\mathbb{E} \left[ d_2 \mid \Omega_N \right] = o(\mathbb{E} \left[ d_1^\ell \right])$. We previously showed that $\mathbb{E} \left[ d_1^\ell \right] = \Theta(\mathbb{E} \left[ d_1^\ell \mid \Omega_N \right])$.

This completes the proof. \(\square\)

## A.3.5 Details omitted from Section 3.3.6 and proof of Theorem 3.3.3

In this section, we complete the proof of Theorem 3.3.3. We begin by proving the analog of Theorem 3.3.3 for a locally adaptive expert.

**Theorem A.3.8.** If $s$ is a respectful, normalizable, continuously differentiable proper scoring rule, and $\text{Err}_c(p)$ is the expected error of a locally adaptive expert scored by $s$ when the coin has bias $p$ and the cost of a flip is $c$, then

$$
\lim_{c \to 0} c^{-\ell/4} \mathbb{E}_{p \sim U[0,1]} \left[ \text{Err}_c(p)^\ell \right] = \mu \ell \int_0^1 \left( \frac{2x(1-x)}{G''_s(x)} \right)^{\ell/4} dx.
$$

**Proof.** Let $N = \frac{1}{\alpha c^{1/3}}$, i.e. a large enough function of $c$ that it is guaranteed that the expert flips the coin at least $N$ times. We have

$$
c^{-\ell/4} \mathbb{E}_{p \sim U[0,1]} \left[ \text{Err}_c(p)^\ell \right] = c^{-\ell/4} \left( \mathbb{E}_{p \sim U[0,1]} \left[ \text{Err}_c(p)^\ell \mid \Omega_N \right] \mathbb{P} \left[ \Omega_N \right] + \mathbb{E}_{p \sim U[0,1]} \left[ \text{Err}_c(p)^\ell \mid \Omega_N^c \right] \mathbb{P} \left[ \Omega_N^c \right] \right).
$$

We wish to compute the limit of this quantity as $c$ approaches zero. Note that

$$
\lim_{c \to 0} c^{-\ell/4} \mathbb{E}_{p \sim U[0,1]} \left[ \text{Err}_c(p)^\ell \mid \Omega_N \right] \mathbb{P} \left[ \Omega_N \right] = 0.
$$

This is because $\text{Err}_c(p)$ is bounded between 0 and 1 and $\mathbb{P} \left[ \Omega_N \right] = O(e^{-N^{0.01}}) = O(e^{-\Omega(c^{-1/300})})$, which goes to zero faster than $c^{-\ell/4}$ goes to infinity. Therefore we have

$$
\lim_{c \to 0} c^{-\ell/4} \mathbb{E}_{p \sim U[0,1]} \left[ \text{Err}_c(p)^\ell \right] = \lim_{c \to 0} c^{-\ell/4} \mathbb{E}_{p \sim U[0,1]} \left[ \text{Err}_c(p)^\ell \mid \Omega_N \right].
$$

(We may ignore the $\mathbb{P} \left[ \Omega_N \right]$ term above because it approaches 1 in the limit.) We may write this
quantity as

\[
\lim_{c \to 0} c^{-\ell/4} \left( (1 - 4c^t)\mathbb{E}_{p \leftarrow \mathcal{U}[2c^t,1-2c^t]} \left[ \text{Err}_c(p)^\ell \mid \Omega_N \right] + 4c^t \mathbb{E}_{p \leftarrow \mathcal{U}[0,2c^t],[1-2c^t,1]} \left[ \text{Err}_c(p)^\ell \mid \Omega_N \right] \right).
\]

Let us focus on the second summand. Let \( p \in [0, 2c^t] \cup [1 - 2c^t, 1] \). We assume \( p \in [0, 2c^t] \); the other case is analogous.

We consider two sub-cases: \( p \in [0, \alpha c^{1/3}] \) and \( p \in [\alpha c^{1/3}, 2c^t] \). First suppose that \( p \in [\alpha c^{1/3}, 2c^t] \). Note that since \( \Omega_N \) holds, we have for all \( n \geq N \) that

\[
|Q(n) - p| \leq \frac{\sqrt{p(1-p)}}{n^{49}} \leq \frac{\sqrt{2} c^t}{N^{49}} = \sqrt{2} \alpha^{49} c^{t/2 + 49/3}.
\]

This in particular is true of \( n = n_{\text{stop}} \), so

\[
\text{Err}_c(p)^\ell \leq (\sqrt{2} \alpha^{49})^\ell c^{\ell(t/2 + 49/3)} = o(c^{\ell/4})
\]

since \( t > \frac{1}{4} \) and so \( \frac{\ell}{2} + \frac{49}{3} > \frac{1}{4} \).

Now suppose that \( p \in [0, \alpha c^{1/3}] \). Recall the notation \( Q_p(n) \) from the discussion preceding the definition of \( \Omega_N \). For any \( n \geq N \), we have

\[
|Q(n) - p| \leq |Q(n) - Q_{\alpha c^{1/3}}(n)| + |Q_{\alpha c^{1/3}}(n) - \alpha c^{1/3}| + |\alpha c^{1/3} - p|
\]

\[
\leq Q_{\alpha c^{1/3}}(n) + |Q_{\alpha c^{1/3}}(n) - \alpha c^{1/3}| + |\alpha c^{1/3} |
\]

\[
\leq 2\alpha c^{1/3} + 2\sqrt{\alpha c^{1/3}} + 2\sqrt{\alpha c^{1/6} + 49/3} = o(c^{1/4})
\]

so \( \text{Err}_c(p)^\ell = o(c^{\ell/4}) \).

This means that

\[
\lim_{c \to 0} c^{-\ell/4} \cdot 4c^t \mathbb{E}_{p \leftarrow \mathcal{U}[0,2c^t],[1-2c^t,1]} \left[ \text{Err}_c(p)^\ell \mid \Omega_N \right] = 0
\]
so we can ignore this summand. Therefore, we have

\[
\lim_{c \to 0} c^{-\ell/4} \mathbb{E}_p \left[ \text{Err}_c(p)^\ell \right] = \lim_{c \to 0} c^{-\ell/4} (1 - 4c^\ell) \mathbb{E}_p \left[ \text{Err}_c(p)^\ell \mid \Omega_N \right].
\]

From Lemma 3.3.10, we have that

\[
(1 - o(1)) \int_{2c^\ell}^{1-2c^\ell} \mu_\ell \left( \frac{2x(1-x)}{G''_s(x)} \right)^{\ell/4} dx \leq c^{-\ell/4} (1 - 4c^\ell) \mathbb{E}_p \left[ \text{Err}_c(p)^\ell \mid \Omega_N \right] \leq (1 + o(1)) \int_{2c^\ell}^{1-2c^\ell} \mu_\ell \left( \frac{2x(1-x)}{G''_s(x)} \right)^{\ell/4} dx.
\]

By the squeeze theorem, we conclude that.

\[
\lim_{c \to 0} c^{-\ell/4} \mathbb{E}_p \left[ \text{Err}_c(p)^\ell \right] = \lim_{c \to 0} \int_{2c^\ell}^{1-2c^\ell} \mu_\ell \left( \frac{2x(1-x)}{G''_s(x)} \right)^{\ell/4} dx = \mu_\ell \int_0^1 \left( \frac{2x(1-x)}{G''_s(x)} \right)^{\ell/4} dx.
\]

Lemma 3.3.11. Assume \( s \) is respectful and normalizable (i.e. \( \int_0^1 G_s(x) dx < \infty \)). Let \( \gamma \) be as in Proposition 3.3.9. There exists a \( c_0 > 0 \), such that for all \( c \leq c_0 \): If \( \Omega_{n_{\text{stop}}} \) holds and \( 4c^\ell \leq Q(n_{\text{stop}}) \leq 1 - 4c^\ell \), then

\[
\text{n}_{\text{stop}} \leq n_g \leq (1 + 6\gamma c^{1/300})n_{\text{stop}}.
\]

Proof. Our approach will be to compare the behavior of a locally adaptive expert to that of a globally adaptive one. We will assume that the experts observe the same stream of coin flips (each heads with probability \( p \) unknown to the experts) but that they may decide to stop at different times. As before, we will let \( Q(n) = \frac{h+1}{n+2} \) where \( h \) is the number of the first \( n \) flips to have come up heads; since the experts see the same coin flips, we do not need to distinguish between \( Q(n) \) for the locally adaptive expert and for the globally adaptive expert. We will let \( n_l \) and \( n_g \) be the number of times the locally and globally adaptive experts flip the coin, respectively (so \( n_g \geq n_l \)). (We used the notation \( n_{\text{stop}} \) in place of \( n_l \) in Proposition 3.3.9.) Let \( t \) be as in the definition of respectful scoring rules, and in particular we will assume that \( t < 0.3 \) as before (for any \( t \) that witnesses that
a scoring rule is respectful, any smaller \( t > \frac{1}{4} \) also works).

Suppose the globally adaptive expert flips the coin \( N := (1 + 6\gamma c^{1/300})n \) times. We show that they do not flip the coin another time.

By definition of \( \Omega_n \), we have that
\[
p - \frac{\sqrt{p(1-p)}}{n_i^{49}} \leq Q(n_i) \leq p + \frac{\sqrt{p(1-p)}}{n_i^{49}}.
\]

It is easy to check that because \( n_i = \Omega(c^{-1/3}) \) (by Claim 3.3.6) and \( 4c^t \leq Q(n_i) \leq 1 - 4c^t \) (so \( Q(n_i), 1 - Q(n_i) = \Omega(n_i^{-9}) \)), the above relationship between \( Q(n_i) \) and \( p \) implies that \( 2c^t \leq p \leq 1 - 2c^t \). This allows us to use some results from our analysis of locally adaptive experts. In particular, by Equation A.2 in the proof of Proposition 3.3.9, we have that
\[
\frac{p(1-p)}{2n_i^2}G''(p)(1 - \gamma c^{1/300}) \leq \Delta_{n_i+1} \leq \frac{p(1-p)}{2n_i^2}G''(p)(1 + \gamma c^{1/300}).
\]

Conditional on \( \Omega_N \) (and by definition \( \Omega_n \) implies \( \Omega_N \)), we also have
\[
\frac{p(1-p)}{2N^2}G''(p)(1 - \gamma c^{1/300}) \leq \Delta_{n+1} \leq \frac{p(1-p)}{2N^2}G''(p)(1 + \gamma c^{1/300}).
\]

In particular this means that
\[
\Delta_{n+1} \leq \frac{n_i^2}{N^2} \cdot \frac{1 + \gamma c^{1/300}}{1 - \gamma c^{1/300}} \Delta_{n_i+1} \leq \left( \frac{n_i}{N} \right)^2 (1 + 3\gamma c^{1/300}) \Delta_{n_i+1}
\]

for \( c \) small enough. This means that if \( \left( \frac{n_i}{N} \right)^2 \leq \frac{1}{1 + 6\gamma c^{1/300}} \) then \( \Delta_{n+1} \leq (1 - 2\gamma c^{1/300}) \Delta_{n_i+1} < c(1 - 2\gamma c^{1/300}) \). Furthermore, for any \( n \geq N \) we will have \( \Delta_{n+1} < c(1 - 2\gamma c^{1/300}) \).

However, this does not mean that the globally adaptive expert won’t flip the coin for the \( N+1 \)-th time, because they don’t know that \( \Omega_n \) is true. From the expert’s perspective, if they knew \( \Omega_n \) (or even \( \Omega_N \)) to be true, they would stop flipping the coin; but perhaps they should keep flipping the coin because of the outside chance that \( \Omega_N \) is false.
This turns out not to be the case, because the probability that $\Omega_N$ is false is so small. In particular, from the expert’s perspective, if $\Omega_N$ is false, they cannot achieve a score that is better than the expectation of $G_s(p)$ conditional on the coins they’ve flipped and on $\Omega_N$ being false. We show that if the scoring rule $s$ is normalizable (i.e. $\int_0^1 G_s(x)dx$ is finite), then this quantity isn’t too large. In particular, we show the following:

**Claim A.3.9.** Let $H_N$ be the random variable corresponding to the number of heads flipped in the first $N$ flips. Then for any $3c^tN \leq h \leq (1 - 3c^t)N$, we have

$$E \left[ G_s(p) \mid \Omega_N, H_N = h \right] \leq \frac{2}{c^t} \int_0^1 G_s(x) dx.$$

**Proof.** We have

$$E \left[ G_s(p) \mid \Omega_N, H_N = h \right] \leq E \left[ G_s(p) \mid \Omega_N, H_N = h, p < \frac{1}{2} \right] + E \left[ G_s(p) \mid \Omega_N, H_N = h, p > \frac{1}{2} \right].$$

Let us consider the expectation conditioned on $p < \frac{1}{2}$. Consider the distribution $D$ of $p$ conditioned on $\Omega_N$, $H_N = h$, and $p < \frac{1}{2}$. Consider also the uniform distribution $D'$ on $[0, c^t]$. We claim that $D$ stochastically dominates $D'$, i.e. $Pr_{x \sim D} [x \leq y] \leq Pr_{x \sim D'} [x \leq y]$ for all $y$. To see this, observe that the PDF of $D$ is an increasing function on $[0, c^t]$. This is because $D$ on $[0, c^t]$ is a constant multiple of the distribution $D''$ of $p$ conditioned on $\Omega_N$, $H_N = h$, and $p \leq c^t$; but in this case the condition $\Omega_N$ is redundant because if $p \leq c^t$ then $\Omega_N$ holds. So $D''$ is the distribution of $p$ conditioned on $H_N = h$ and $p \leq c^t$. Clearly the PDF of $D''$ increases on $[0, c^t]$ (because the expert starts with uniform priors and updates more strongly on against values of $p$ farther from $\frac{h+1}{N+2}$, which is greater than $2.9c^t$ for $c$ small enough).

Now, the expectation of $G_s(p)$ if $p$ were drawn from $D'$ instead of $D$ is equal to $\frac{1}{c^t} \int_0^{c^t} G_s(x)dx \leq \frac{1}{c^t} \int_0^1 G_s(x)dx$. On the other hand, the actual expectation of $G_s(p)$ (i.e. with $p$ drawn from $D$) is necessarily smaller. This is because $G_s$ is convex and symmetric about $\frac{1}{2}$, meaning that $G_s$ is
decreasing on \((0, \frac{1}{2})\). Since \(D\) stochastically dominates \(D'\), we conclude that

\[
\mathbb{E} \left[ G_s(p) \mid \overline{\Omega_N}, H_N = h, p < \frac{1}{2} \right] \leq \frac{1}{c_t} \int_0^1 G_s(x) dx.
\]

The same inequality holds conditional instead on \(p > \frac{1}{2}\), which concludes the proof.

From the expert’s perspective, this means that if they flip the coin for the \(N + 1\)-th time, then:

- In the case that \(\Omega_N\) is true, the best case is that they never flip the coin again, in which case they will pay a total cost of \(c\) and get expected score at most \(c(1 - 2\gamma c^{1/300})\).
- In the case that \(\Omega_N\) is false, the best case is that they get a score of \(\frac{2}{c_t} \int_0^1 G_s(x) dx\).

In other words, the expert’s expected score if they flip the coin for the \(N + 1\)-th time and pursue the optimal strategy from there is at most

\[
\frac{2}{c_t} \int_0^1 G_s(x) dx \cdot \Pr \left[ \overline{\Omega_N} \right] - 2\gamma c^{1/300} \Pr [\Omega_N] \leq O \left( \frac{e^{-N^{0.01}}}{c^t} \right) - \gamma c^{1/300} = O \left( \frac{e^{-\Omega(c^{1/300})}}{c^t} \right) - \gamma c^{1/300}.
\]

The first step is nontrivial: it uses the fact that the probability that the expert assigns to \(\overline{\Omega_N}\) after the first \(N\) flips is \(O(e^{-N^{0.01}})\). This doesn’t immediately follow from Claim 3.3.8 because the claim only states that the prior probability of \(\overline{\Omega_N}\), i.e. before any flips, is \(O(e^{-N^{0.01}})\). To see that the posterior probability (after the first \(N\) flips) is also of this order, we first observe that the posterior probability cannot depend on the order of the flip outcomes; this is apparent from the definition of \(\Omega_N\). However, perhaps the number of heads, i.e. the value of \(H_N\), affects the posterior probability of \(\overline{\Omega_N}\). This may be so, but it cannot increase the probability by more than a factor of \(N + 1\). That is because the prior for \(H_N\) is uniform over \(\{0, \ldots, N\}\) (Fact A.1.1).

Now, the quantity on the right is negative for \(c\) small enough, so the expert will not flip the \(N + 1\)-th coin. This proves the claim.

The following corollary is essentially identical to Proposition 3.3.9 but for globally adaptive experts.
Corollary A.3.10. Assume that $\Omega_N$ holds for some $N$. For sufficiently small $c$, for all $p \in [8c', 1 - 8c']$, we have
\[
\sqrt{\frac{p(1 - p)G''_s(p)}{2c}} (1 - 14\gamma c^{1/300}) \leq n_g \leq \sqrt{\frac{p(1 - p)G''_s(p)}{2c}} (1 + 14\gamma c^{1/300}).
\]

Proof. Because $\Omega_N$ holds for some $N$, for sufficiently small $c$ the fact that $p \in [8c', 1 - 8c']$ implies that $Q(n_t) \in [4c', 1 - 4c']$. This means that we may apply Lemma 3.3.11 to say that $n_g \leq N$. Consequently we have that
\[
\sqrt{\frac{p(1 - p)G''_s(p)}{2c}} (1 - \gamma c^{1/300}) \leq n_t \leq n_g \leq (1 + 6\gamma c^{1/300})n_t
\]
\[
\leq (1 + 6\gamma c^{1/300})\sqrt{\frac{p(1 - p)G''_s(p)}{2c}} (1 + \gamma c^{1/300}) \leq \sqrt{\frac{p(1 - p)G''_s(p)}{2c}} (1 + 14\gamma c^{1/300})
\]
for $c$ small enough. Therefore we have
\[
\sqrt{\frac{p(1 - p)G''_s(p)}{2c}} (1 - 14\gamma c^{1/300}) \leq n_g \leq \sqrt{\frac{p(1 - p)G''_s(p)}{2c}} (1 + 14\gamma c^{1/300}).
\]

Theorem 3.3.3 follows as a simple corollary.

Theorem 3.3.3. If $\mu_{\ell} := \frac{2^{\ell/2} \Gamma\left(\frac{\ell + 1}{2}\right)}{\sqrt{\pi}}$ is the $\ell$th moment of the standard normal distribution, then
\[
\lim_{c \to 0} c^{-\ell/4} \cdot \text{Error}_c^\ell(s) = \mu_{\ell} \cdot 2^{\ell/4} \cdot \text{Ind}_c^\ell(s).
\]

Proof. The lemma analogous to Lemma 3.3.10 but for expected globally adaptive error, and for $p \in [8c', 1 - 8c']$, follows immediately from Corollary A.3.10. This is because the proof of Lemma 3.3.10 makes no assumptions about the specific value of $\gamma$ (other than that it is positive), which means that the proof goes through just as well for $14\gamma$ in place of $\gamma$. Theorem 3.3.3 follows from this fact exactly in the same way that Theorem A.3.8 followed from Lemma 3.3.10.
A.4 Details omitted from Section 3.4

Lemma 3.4.2. For any $\ell \geq 1$, a function $h : \left[\frac{1}{2}, 1\right] \to \mathbb{R}_{\geq 0}$ satisfying $\int_{\frac{1}{2}}^{1} (1-x)h(x)dx = 1$ that minimizes $\int_{\frac{1}{2}}^{1} \left(\frac{x(1-x)^2}{h(x)}\right)^{\ell/4} dx$ is $\tilde{h}_\ell(x) = \kappa_\ell \left(x^\ell (1-x)^{2\ell-4} \right)^{1/(\ell+4)}$, where $\kappa_\ell = \left(\int_{\frac{1}{2}}^{1} x(1-x)^3 \right)^{-1}.$

Proof. Let $\lambda_\ell = \frac{\ell}{4 \kappa_\ell^{1/(\ell+4)}}$. Consider the functional

$$\chi(h) := \int_{\frac{1}{2}}^{1} \left(\frac{x(1-x)^2}{h(x)}\right)^{\ell/4} dx + \lambda_\ell \left(\int_{\frac{1}{2}}^{1} (1-x)h(x)dx - 1\right)$$

$$= \int_{\frac{1}{2}}^{1} \left(\frac{x(1-x)^2}{h(x)}\right)^{\ell/4} dx + \lambda_\ell (1-x)h(x) dx - \lambda_\ell.$$

It suffices to show that among all $h$ satisfying $\int_{\frac{1}{2}}^{1} (1-x)h(x)dx = 1$, $\tilde{h}$ minimizes $\chi(h)$. This is because among such $h$, the second summand in the definition of $\chi$ is always zero. In fact, we prove something stronger: $\tilde{h}$ minimizes $\chi$, among all functions from $\left[\frac{1}{2}, 1\right]$ to $\mathbb{R}$. To show this, it suffices to show that for every $x \in \left[\frac{1}{2}, 1\right]$, the value $y$ that minimizes

$$\left(\frac{x(1-x)^2}{y}\right)^{\ell/4} + \lambda_\ell (1-x)y$$

is $y = \tilde{h}(x)$. The derivative with respect to $y$ of this expression is

$$\frac{-\ell}{4} (x(1-x)^2)^{\ell/4} y^{-(\ell/4+1)} + \lambda_\ell (1-x),$$

which is an increasing function of $y$ (since $y^{-(\ell/4+1)}$ is a decreasing function of $y$ and $\frac{-\ell}{4} (x(1-x)^2)^{\ell/4}$ is negative). It is equal to 0 precisely when

$$y = \left(\frac{\ell}{4 \lambda_\ell}\right)^{4/(\ell+4)} x^\ell (1-x)^{2\ell-4} = \kappa_\ell x^\ell (1-x)^{2\ell-4} 1/(\ell+4) = \tilde{h}(x).$$

It remains only to note that $\int_{\frac{1}{2}}^{1} (1-x)\tilde{h}(x)dx = 1$, and this follows immediately from the definition of $\tilde{h}$ and $\kappa_\ell$. \qed
Corollary 3.4.3. The unique continuous function $h : [\frac{1}{2}, 1) \to \mathbb{R}_{\geq 0}$ satisfying $\int_{\frac{1}{2}}^{1} (1-x)h(x)dx = 1$ that minimizes $\int_{\frac{1}{2}}^{1} \left( \frac{(1-x)^2}{h(x)} \right)^{\ell/4} dx$ is $\tilde{h}_\ell$.

Proof. Suppose for contradiction that there is another continuous function $\hat{h}$ satisfying the above properties that achieves the minimum. Then $\chi(\hat{h}) = \chi(\tilde{h})$, with $\chi$ as in the proof of Lemma 3.4.2, since $\tilde{h}$ minimizes $\chi$ and the second summand in the definition of $\chi$ is zero for both $\hat{h}$ and $\tilde{h}$. In particular, we have that $\chi(\hat{h}) - \chi(\tilde{h}) = 0$, i.e.

$$\int_{\frac{1}{2}}^{1} \left( \left( \frac{(1-x)^2}{\hat{h}(x)} \right)^{\ell/4} + \lambda_\ell(1-x)\hat{h}(x) \right) - \left( \left( \frac{(1-x)^2}{\tilde{h}(x)} \right)^{\ell/4} + \lambda_\ell(1-x)\tilde{h}(x) \right) dx = 0.$$

Let $\Delta(x)$ be the integrand. Note that $\Delta$ is always nonnegative, and is zero precisely for those values of $x$ where $\hat{h}(x) = \tilde{h}(x)$ (since, as we showed earlier, $y = \tilde{h}(x)$ is the unique value minimizing $\left( \frac{(1-x)^2}{y} \right)^{\ell/4} + \lambda_\ell(1-x)y$). Since $\hat{h} \neq \tilde{h}$, $\Delta$ is positive at some $x_0$; say $\Delta(x_0) = y_0$. Also, note that $\Delta$ is continuous because $\hat{h}$ and $\tilde{h}$ are continuous. This means that for some $\delta > 0$, $|\Delta(x) - y_0| < \frac{y_0}{2}$ for all $x$ such that $x_0 \leq x \leq x_0 + \delta$. But this means that the integral of $\Delta$ on $[x_0, x_0 + \delta]$ is at least $\frac{\delta \cdot y_0}{2} > 0$, so $\int_{\frac{1}{2}}^{1} \Delta(x) > 0$, a contradiction. Therefore, $\tilde{h}$ is indeed the unique continuous function satisfying the stated constraints. \qed

Theorem 3.4.1. The unique continuously differentiable normalized proper scoring rule which minimizes $\text{Ind}_\ell^k(s)$ is:

$$s_{\ell, \text{Opt}}(x) := \begin{cases} \kappa_\ell \int_{\frac{1}{2}}^{x} (t^\ell - 8(t^\ell - 1)^{2\ell+4})^{1/(\ell+4)} dt & x \leq \frac{1}{2} \\ \kappa_\ell \int_{\frac{1}{2}}^{x} (t^\ell (1-t)^{2\ell+4})^{1/(\ell+4)} dt & x \geq \frac{1}{2}. \end{cases}$$

Proof. We have reasoned that $s_{\ell, \text{Opt}}$ is the antiderivative of $\tilde{h}$ on $[\frac{1}{2}, 1)$, which gives us $s_{\ell, \text{Opt}}$ for $x \geq \frac{1}{2}$. For $x < \frac{1}{2}$, we have

$$s'_{\ell, \text{Opt}}(x) = \frac{1-x}{x} s_{\ell, \text{Opt}}(1-x) = \kappa_\ell \frac{1-x}{x} ((1-x)^{\ell} x^{2\ell+4})^{1/(\ell+4)} = \kappa_\ell (x^{\ell-8} (1-x)^{2\ell+4})^{1/(\ell+4)},$$

which extends to the stated function $s_{\ell, \text{Opt}}$ by the fundamental theorem of calculus. The constant we want to add (upon taking the antiderivative) is zero so that $s_{\ell, \text{Opt}} \left( \frac{1}{2} \right) = 0$. We need to check
that \( s_{\ell, \text{Opt}} \) is continuously differentiable at \( \ell \), which means checking that \( \tilde{h} \) is continuous at \( \ell \) when extended to \((0, 1)\). This is indeed the case because

\[
\lim_{x \to \frac{1}{2}^-} s'_{\ell, \text{Opt}}(x) = \lim_{x \to \frac{1}{2}^-} \frac{1 - x}{x} s'_{\ell, \text{Opt}}(1 - x) = \lim_{x \to \frac{1}{2}^+} s'_{\ell, \text{Opt}}(1 - x) = \lim_{x \to \frac{1}{2}^+} s'_{\ell, \text{Opt}}(x).
\]

Finally, \( s_{\ell, \text{Opt}} \) is the unique continuous normalized minimizer because its derivative is unique, by Corollary 3.4.3. Note that \( s_{\ell, \text{Opt}} \) is in fact strictly proper since \( \tilde{h}(x) \) is positive on \([\frac{1}{2}, 1)\).

**Remark 3.4.4.** For every \( \ell \in [1, 8] \), \( s_{\ell, \text{Opt}} \) is respectful. For \( \ell > 8 \), and all \( \varepsilon > 0 \), there exists a respectful normalized proper scoring rule \( s \) such that \( |s(x) - s_{\ell, \text{Opt}}(x)| \leq \varepsilon \) for all \( x \in (0, 1) \), with \( \text{Ind}^\ell(s) \leq \text{Ind}^\ell(s_{\ell, \text{Opt}}) + \varepsilon \).

**Proof of Remark 3.4.4** for \( 1 \leq \ell \leq 8 \). This proof handles the case of \( \ell \in [1, 8] \). Let us write \( G_{\ell, \text{Opt}} \) to mean \( G_{s_{\ell, \text{Opt}}} \). Then

\[
G''_{\ell, \text{Opt}}(x) = \frac{s'_{\ell, \text{Opt}}(x)}{1 - x} = \begin{cases} 
\kappa_\ell (x^{\ell-8} (1-x)^{\ell})^{1/(\ell+4)} & x \leq \frac{1}{2} \\
\kappa_\ell (x^{\ell} (1-x)^{\ell-8})^{1/(\ell+4)} & x \geq \frac{1}{2}.
\end{cases}
\]

First note that \( G_{\ell, \text{Opt}} \) is strongly convex. Since \( G''_{\ell, \text{Opt}} \) is symmetric, it suffices to show this on \((0, \frac{1}{2}]\). We have \( 1 - x \geq \frac{1}{2} \) on this interval, and \( x^{(\ell-8)/(\ell+4)} \) is bounded away from zero when \( x \leq 8 \). Next, the fact that \( G''_{\ell, \text{Opt}} \) is Riemann integral on any closed sub-interval of \((0, 1)\) is evident. Finally, there are constants \( k \neq 0 \) and \( r \) such that \( \lim_{x \to 0} x^r G'''_{\ell, \text{Opt}}(x) = k \): in particular, \( r = \frac{12}{\ell+4} \) and \( k = \frac{(\ell-8)\kappa_\ell}{\ell+4} \).

(Note that for \( \ell > 8 \), \( s_{\ell, \text{Opt}} \) is not respectful, since \( \lim_{x \to 0} G''_{\ell, \text{Opt}}(x) = 0 \).)
A.5 Details omitted from Section 3.6

Theorem 3.6.2. Let \( s : (0, 1) \to \mathbb{R} \) be analytic (and symmetric, as we have been assuming). Then

\[ s \] is a proper scoring rule if and only if \( s \) is nonconstant, \( s'(x) \geq 0 \) everywhere, and

\[
s(x) = c_0 + \sum_{k > 0 \text{ odd}} c_k (2k + 1 - 2kx) \left(x - \frac{1}{2}\right)^k
\]

for some \( c_0, c_1, c_3, c_5, \ldots \in \mathbb{R} \).

Proof. Suppose that \( s \) is a proper scoring rule. Then \( s \) is nonconstant, \( s'(x) \geq 0 \) everywhere by Lemma 3.2.4, and by the same lemma we have that

\[
xs'(x) = (1 - x)s'(1 - x).
\]

Taking successive derivatives of both sides, we have

\[
s'(x) + xs''(x) = -s'(1 - x) - (1 - x)s''(1 - x)
\]

\[
2s''(x) + xs'''(x) = 2s''(1 - x) + (1 - x)s'''(1 - x)
\]

\[
3s'''(x) + xs^{(4)}(x) = -3s'''(1 - x) - (1 - x)s^{(4)}(1 - x)
\]

and so on. Plugging in \( x = \frac{1}{2} \), we have

\[
s'(1/2) + 1/2s''(1/2) = -s'(1/2) - 1/2s''(1/2)
\]

\[
2s''(1/2) + 1/2s'''(1/2) = 2s''(1/2) + 1/2s'''(1/2)
\]

\[
3s'''(1/2) + 1/2s^{(4)}(1/2) = -3s'''(1/2) - 1/2s^{(4)}(1/2)
\]

and so on. These equations alternate between giving us tautologies and simplifying to the following
identities:

\[ s'' (1/2) = -2s' (1/2); \quad  s^{(4)} (1/2) = -6s'' (1/2); \quad  s^{(6)} (1/2) = -10f^{(5)} (1/2); \]

and so on, the general form of the identities being that for \( k \) odd, we have

\[ s^{(k+1)} (1/2) = -2k s^{(k)} (1/2). \]

Since \( s \) is analytic, we have

\[
s(x) = \sum_{n=0}^{\infty} \frac{1}{n!} s^{(n)} \left( \frac{1}{2} \right) \left( x - \frac{1}{2} \right)^n
\]

\[
= s \left( \frac{1}{2} \right) + \sum_{k>0 \text{ odd}} \frac{1}{(k+1)!} \left( k+1 \right) s^{(k)} \left( \frac{1}{2} \right) \left( x - \frac{1}{2} \right)^k + s^{(k+1)} \left( \frac{1}{2} \right) \left( x - \frac{1}{2} \right)^{k+1}. 
\]

Letting \( c_k = \frac{1}{(k+1)!} s^{(k)} \left( \frac{1}{2} \right) \) for \( k = 0, 1, 3, 5, \ldots \), we have

\[
s(x) = c_0 + \sum_{k>0 \text{ odd}} c_k \left( k+1 \right) \left( x - \frac{1}{2} \right)^k - 2k \left( x - \frac{1}{2} \right)^{k+1}
\]

\[
= c_0 + \sum_{k>0 \text{ odd}} c_k (2k + 1 - 2kx) \left( x - \frac{1}{2} \right)^k.
\]

This proves the forward direction. Conversely, we claim that if \( s \) is nonconstant, \( s'(x) \geq 0 \) everywhere, and \( s \) can be written in the stated form for some \( c_0, c_1, c_3, \ldots \), then \( s \) is a proper scoring rule. We only have to verify that \( xs'(x) = (1 - x)s'(1 - x) \) everywhere (by Lemma 3.2.4 and Lemma A.5.1 (stated and proven below)). Taking the derivative of \( (1 - x)s'(1 - x) \) term by term, we have

\[
(1 - x)s'(1 - x) = (x - 1)(s(1 - x))' = (x - 1) \frac{d}{dx} \left( c_0 + \sum_{k>0 \text{ odd}} c_k (2kx + 1) \left( \frac{1}{2} - x \right)^k \right)
\]

\[
= (x - 1) \sum_{k>0 \text{ odd}} c_k \left( 2k \left( \frac{1}{2} - x \right)^k - k(2kx + 1) \left( \frac{1}{2} - x \right)^{k-1} \right)
\]
\[
\begin{align*}
&= \sum_{k > 0 \text{ odd}} (x-1)kc_k \left(\frac{1}{2} - x\right)^{k-1} (1 - 2x - 2kx - 1) \\
&= \sum_{k > 0 \text{ odd}} 2k(k+1)c_kx(1-x) \left(\frac{1}{2} - x\right)^{k-1}.
\end{align*}
\]

Similarly, we have

\[
x's(x) = x \frac{d}{dx} \left( c_0 + \sum_{k > 0 \text{ odd}} c_k \left( 2k + 1 - 2kx \right) \left( x - \frac{1}{2} \right)^{k-1} \right)
= x \sum_{k > 0 \text{ odd}} c_k \left( -2k \left( x - \frac{1}{2} \right)^k + k(2k + 1 - 2kx) \left( x - \frac{1}{2} \right)^{k-1} \right)
= \sum_{k > 0 \text{ odd}} xkc_k \left( x - \frac{1}{2} \right)^{k-1} (1 - 2x + 2k + 1 - 2kx)
= \sum_{k > 0 \text{ odd}} 2k(k+1)c_kx(1-x) \left( x - \frac{1}{2} \right)^{k-1} = (1-x)s'(1-x),
\]

as desired. \qed

**Lemma A.5.1.** The only infinitely differentiable scoring rules that are weakly proper but not proper are constant functions.

**Proof.** Let \(s\) be an infinitely differentiable scoring rule that is weakly proper but not proper. Recall that in the proof of Lemma 3.2.4, we showed that for all \(p\), the function \(r_p(x)\) weakly increases on \((0, p]\) and weakly decreases on \([p, 1)\). Since \(s\) is not proper, there is some \(p\) such that \(r_p(x)\) does not strictly increase on \((0, p]\) or does not strictly decrease on \([p, 1)\). But this means that \(r_p(x)\) is constant on some open interval, which means that \(r_p(x)\) is constant (because \(s\) is infinitely differentiable, which means that \(r_p(x)\) is also infinitely differentiable). Thus, for some \(c \in \mathbb{R}\) we have that \(ps(x) + (1 - p)s(1 - x) = c\) for all \(x\). Taking the derivative, we have that \(ps'(x) = (1 - p)s'(1 - x)\) for all \(x\). But we also have that \(xs'(x) = (1 - x)s'(1 - x)\) for all \(x\). The only way for both of these equations to hold is for \(s'(x)\) to be uniformly zero, so \(s\) is indeed constant. \qed

**Lemma 3.6.3.** Let \(s : (0, 1) \to \mathbb{R}\) be analytic and symmetric. Then \(s\) is a proper scoring rule if
and only if $G''_s$ is not uniformly zero, nonnegative everywhere, and can be written as

$$G''_s(x) = \sum_{k \geq 0 \text{ even}} d_k \left(x - \frac{1}{2}\right)^k.$$ 

**Proof.** By Theorem 3.6.2, $s$ is a proper scoring rule if and only if $s$ is nonconstant, $s'(x) \geq 0$ everywhere, and

$$s(x) = c_0 + \sum_{k > 0 \text{ odd}} c_k (2k + 1 - 2kx) \left(x - \frac{1}{2}\right)^k$$

for some $c_0, c_1, c_3, c_5, \ldots \in \mathbb{R}$. Equivalently,

$$s'(x) = \sum_{k > 0 \text{ odd}} kc_k (2k + 1 - 2kx) \left(x - \frac{1}{2}\right)^{k-1} - 2kc_k \left(x - \frac{1}{2}\right)^k$$

$$= \sum_{k > 0 \text{ odd}} k c_k \left(2k + 1 - 2kx - 2 \left(x - \frac{1}{2}\right)\right) \left(x - \frac{1}{2}\right)^{k-1}$$

$$= \sum_{k > 0 \text{ odd}} 2kc_k (k + 1)(1 - x) \left(x - \frac{1}{2}\right)^{k-1}$$

$$= \sum_{k \geq 0 \text{ even}} 2(k + 1)c_{k+1}(k + 2)(1 - x) \left(x - \frac{1}{2}\right)^k = (1 - x) \sum_{k \geq 0 \text{ even}} d_k \left(x - \frac{1}{2}\right)^k,$$

where $d_k = 2(k + 1)(k + 2)c_{k+1}$. Noting that $s$ is constant if and only if $G''_s$ is uniformly zero, and that $s'(x) \geq 0$ if and only if $G''_s(x) \geq 0$, this completes the proof. 

In proving Theorem 3.6.1, we will make substantial use of $G''_s$, the second derivative of the expected score function of $s$. For convenience, we will write $\varphi$ instead of $G''_s$.

Note that if $s$ is a continuously differentiable (but not necessarily infinitely differentiable) proper scoring rule, then we have

$$\text{Ind}^\ell(s) = 2 \int_{\frac{1}{2}}^1 \left(\frac{x(1-x)}{\varphi(x)}\right)^{\ell/4} dx.$$
Also, note that \( s \) is normalized if and only if \( s \left( \frac{1}{2} \right) = 0 \) and (by Claim 3.2.9 and Remark 3.2.6)
\[
\int_{\frac{1}{2}}^{1} (1-x)s'(x)dx = \int_{\frac{1}{2}}^{1} (1-x)^2 \varphi(x)dx = 1.
\]

**Theorem 3.6.1.** For \( \ell \geq 1 \) and \( \varepsilon > 0 \), there exists a respectful polynomial normalized proper scoring rule \( s \) satisfying \( \text{Ind}^\ell(s) \leq \text{Ind}^\ell(s_{\ell,\text{Opt}}) + \varepsilon \).

**Proof.** The Weierstrass approximation theorem says that any continuous function can be uniformly approximated by polynomials on a closed interval. A constructive proof of this theorem (for the interval \([0, 1]\)) is given by the Bernstein polynomials: \( b_{i,n}(x) = \binom{n}{i} x^i (1-x)^{n-i} \). Given a continuous function \( \psi : [0, 1] \rightarrow \mathbb{R} \), define
\[
B_n(\psi)(x) = \sum_{i=0}^{n} \psi \left( \frac{i}{n} \right) b_{i,n}(x).
\]

Then the polynomials \( B_n(\psi) \) converge uniformly to \( \psi \) (Estep, 1998, §36). Suppose that \( \psi \) also satisfies \( \psi(x) = \psi(1-x) \). Then \( \psi \left( \frac{i}{n} \right) = \psi \left( \frac{n-i}{n} \right) \), which means \( B_n(\psi) \) can be written as a linear combination of polynomials \( (b_{i,n} + b_{n-i,n})(x) \). These polynomials are equal at \( x \) and \( 1-x \), and thus \( B_n(\psi)(x) = B_n(\psi)(1-x) \). From this we conclude that \( \psi \) can be uniformly approximated on \([0, 1]\) by a sequence polynomials \( p_i \) that satisfy \( p_i(x) = p_i(1-x) \).

Let \( \varphi_\ell \) be the \( \varphi \) corresponding to \( s_{\ell,\text{Opt}} \). Recall that
\[
\varphi_\ell(x) = \begin{cases} 
\kappa_\ell(x^\ell-8 (1-x)^\ell)^{1/(\ell+4)} & x \leq \frac{1}{2} \\
\kappa_\ell(x^\ell (1-x)^{\ell-8})^{1/(\ell+4)} & x \geq \frac{1}{2}.
\end{cases} \tag{A.4}
\]

Let \( 0 < \varepsilon < \min\left( \frac{1}{2}, \varphi_\ell\left( \frac{1}{2} \right) \right) \). Consider the following function \( \varphi_{\ell,\varepsilon} : [0, 1] \rightarrow \mathbb{R} \).
\[
\varphi_{\ell,\varepsilon}(x) = \begin{cases} 
\varphi_\ell(\varepsilon) & x \leq \varepsilon \\
\varphi_\ell(x) & \varepsilon \leq x \leq 1 - \varepsilon \\
\varphi_\ell(1 - \varepsilon) & x \geq 1 - \varepsilon.
\end{cases}
\]

Observe that \( \varphi_{\ell,\varepsilon}(x) = \varphi_{\ell,\varepsilon}(1-x) \) for all \( x \in [0, 1] \); this is a straightforward consequence
of the fact that \( \varphi_{\ell} \) is symmetric about \( \frac{1}{2} \). Per our discussion above, there exists a polynomial \( p_\epsilon \) satisfying \( p_\epsilon(x) = p_\epsilon(1 - x) \) such that for all \( x \in [0, 1] \), \( |p_\epsilon(x) - \varphi_{\ell, \epsilon}(x)| \leq \epsilon \). In particular, we take \( p_\epsilon = B_{n(\epsilon)}(\varphi_{\ell, \epsilon}) \), where \( n(\epsilon) \) is any \( n \) large enough that \( p_\epsilon \) is uniformly within \( \epsilon \) of \( \varphi_{\ell, \epsilon} \).

Observe that such a polynomial, when written as a sum of powers of \( x - \frac{1}{2} \), must only contain even powers of \( x - \frac{1}{2} \), since \( p_\epsilon(x) - p_\epsilon(1 - x) \) must be the zero polynomial. Consequently, by Lemma 3.6.3, \((1 - x)p_\epsilon(x)\) is the derivative of a proper scoring rule.\(^4\) To find the associated normalized proper scoring rule (call it \( s_\epsilon \)), we take the antiderivative (taking the constant coefficient in the \((x - \frac{1}{2})\)-expansion to be zero), and divide by \( \int_{\frac{1}{2}}^{1} (1 - x)^2 p_\epsilon(x) \, dx \). Thus, corresponding to each \( \epsilon \) we have a normalized polynomial scoring rule \( s_\epsilon \) with incentivization index

\[
\text{Ind}^\ell(s_\epsilon) = 2 \int_{\frac{1}{2}}^{1} \left( \frac{x(1 - x)}{p_\epsilon(x)} \right) \frac{\ell/4}{\int_{\frac{1}{2}}^{1} (1 - x)^2 p_\epsilon(x) \, dx} \, dx = 2 \left( \int_{\frac{1}{2}}^{1} (1 - x)^2 p_\epsilon(x) \, dx \right)^{\ell/4} \int_{\frac{1}{2}}^{1} \left( \frac{x(1 - x)}{p_\epsilon(x)} \right)^{\ell/4} \, dx.
\]

**Claim A.5.2.** \( s_\epsilon \) is respectful.

**Proof.** Since \( s_\epsilon \) is polynomial (and thus bounded and infinitely differentiable), it suffices to show that the second derivative of its expected score function is bounded away from zero. The second derivative of \( s_\epsilon \)'s expected score function is a positive multiple of \( p_\epsilon \), so it suffices to show that \( p_\epsilon \) is bounded away from zero. This is indeed the case. To see this, note that \( \varphi_{\ell, \epsilon} \) is bounded away from zero (as \( \varphi_{\ell} \) is bounded away from zero on \([\epsilon, 1 - \epsilon] \)); let \( L > 0 \) be such that \( \varphi_{\ell, \epsilon}(x) \geq L \) on \([0, 1] \). Then

\[
p_\epsilon(x) = \sum_{i=0}^{n(\epsilon)} \varphi_{\ell, \epsilon} \left( \frac{i}{n(\epsilon)} \right) \left( \begin{array}{c} n(\epsilon) \\ i \end{array} \right) x^i (1 - x)^{n(\epsilon) - i} \geq \sum_{i=0}^{n(\epsilon)} L \left( \begin{array}{c} n(\epsilon) \\ i \end{array} \right) x^i (1 - x)^{n(\epsilon) - i} = L.
\]

\[\square\]

Our goal is to upper bound \( \text{Ind}^\ell(s_\epsilon) \) in a way that shows that \( \lim_{\epsilon \to 0} \text{Ind}^\ell(s_\epsilon) = \text{Ind}^\ell(s_{\ell, \text{Opt}}) \).

To do this, it suffices to show that the first of the two integrals in our formula for \( s_\epsilon \) converges to

---

\(^4\)The fact that \( p_\epsilon \) is nonnegative everywhere follows from the fact that it is a uniform \( \epsilon \)-approximation of \( \varphi_{\ell, \epsilon} \), which is greater than 1 on \([0, 1] \).
1 as $\epsilon \to 0$ and that twice the second integral converges to $\text{Ind}^\ell(s_\ell, \text{Opt})$ as $\epsilon \to 0$. We begin by working with the first of the two integrals.

**Claim A.5.3.**

$$\limsup_{\epsilon \to 0} \int_{\frac{1}{2}}^{1} (1-x)^2 p_\epsilon(x) dx \leq 1.$$

**Proof.** First observe that, since $s_\ell, \text{Opt}$ is normalized, we have $\int_{\frac{1}{2}}^{1} (1-x)^2 \varphi_\ell(x) dx = 1$. Next, note that for $1 \leq \ell \leq 8$, $\varphi_\ell$ is increasing on $[\frac{1}{2}, 1)$ (as is evident from Equation A.4), which means that

$$\int_{\frac{1}{2}}^{1} (1-x)^2 \varphi_{\ell, \epsilon}(x) dx \leq \int_{\frac{1}{2}}^{1} (1-x)^2 \varphi_\ell(x) dx = 1. \quad (1 \leq \ell \leq 8)$$

On the other hand, for $\ell > 8$, observe that $\varphi_\ell$ is bounded above on $[\frac{1}{2}, 1)$, say by a constant $M_\ell$, which means that in this case

$$\int_{\frac{1}{2}}^{1} (1-x)^2 \varphi_{\ell, \epsilon}(x) dx \leq \int_{\frac{1}{2}}^{1} (1-x)^2 \varphi_\ell(x) dx + \epsilon M_\ell = 1 + \epsilon M_\ell. \quad (\ell > 8)$$

Now, we have

$$\left(\int_{\frac{1}{2}}^{1} (1-x)^2 p_\epsilon(x) dx\right)^{\ell/4} \leq \left(\int_{\frac{1}{2}}^{1} (1-x)^2 (\varphi_{\ell, \epsilon}(x) + \epsilon) dx\right)^{\ell/4} \leq \left(\epsilon + \int_{\frac{1}{2}}^{1} (1-x)^2 \varphi_{\ell, \epsilon}(x) dx\right)^{\ell/4}$$

which is at most $(1 + \epsilon)^{\ell/4}$ (for $\ell \leq 8$) and at most $(1 + (M_\ell + 1)\epsilon)^{\ell/4}$ (for $\ell > 8$).

Next we work with the second integral.

**Claim A.5.4.**

$$\limsup_{\epsilon \to 0} 2 \int_{\frac{1}{2}}^{1} \left(\frac{x(1-x)}{p_\epsilon(x)}\right)^{\ell/4} dx \leq I(s_\ell, \text{Opt}).$$

**Proof.** We have

$$\int_{\frac{1}{2}}^{1} \left(\frac{x(1-x)}{p_\epsilon(x)}\right)^{\ell/4} dx \leq \int_{\frac{1}{2}}^{1} \left(\frac{x(1-x)}{\varphi_{\ell, \epsilon}(x) - \epsilon}\right)^{\ell/4} dx = \int_{\frac{1}{2}}^{1} \left(\frac{x(1-x)}{\varphi_{\ell, \epsilon}(x) - \epsilon}\right)^{\ell/4} dx$$
\[ \left( 1 - \max_{x \in [\frac{1}{2}, 1]} \frac{\epsilon}{\varphi_{\ell, \epsilon}(x)} \right)^{\ell/4} \int_{\frac{1}{2}}^{1} \left( \frac{x(1-x)}{\varphi_{\ell, \epsilon}(x)} \right)^{\ell/4} dx. \]

Let us consider \( \max_{x \in [\frac{1}{2}, 1]} \frac{\epsilon}{\varphi_{\ell, \epsilon}(x)} \) as a function of \( \epsilon \). For \( \ell \leq 8 \), since \( \varphi_{\ell} \) is increasing on \([\frac{1}{2}, 1]\), this is just \( \frac{\epsilon}{\varphi_{\ell}(\frac{1}{2})} \), a quantity that approaches zero as \( \epsilon \) approaches zero.

For \( \ell > 8 \), as \( \epsilon \) approaches zero we have that \( \min_{x \in [\frac{1}{2}, 1]} \varphi_{\ell, \epsilon}(x) \) approaches \( \omega(\epsilon) \). This means that \( \max_{x \in [\frac{1}{2}, 1]} \frac{\epsilon}{\varphi_{\ell, \epsilon}(x)} \) approaches zero as \( \epsilon \) approaches zero.

Therefore, we have

\[ \limsup_{\epsilon \to 0} 2 \int_{\frac{1}{2}}^{1} \left( \frac{x(1-x)}{p_{\epsilon}(x)} \right)^{\ell/4} dx \leq \limsup_{\epsilon \to 0} 2 \int_{\frac{1}{2}}^{1} \left( \frac{x(1-x)}{\varphi_{\ell, \epsilon}(x)} \right)^{\ell/4}. \]

Next, note that for \( x \in [1 - \epsilon, 1] \), we have

\[ \left( \frac{x(1-x)}{\varphi_{\ell, \epsilon}(x)} \right)^{\ell/4} = \left( \frac{x(1-x)}{\varphi_{\ell}(1-\epsilon)} \right)^{\ell/4} \leq \left( \frac{\epsilon(1-\epsilon)}{\varphi_{\ell}(1-\epsilon)} \right)^{\ell/4} = \left( \frac{\epsilon(1-\epsilon)}{\kappa_{\ell}((1-\epsilon)^{\ell-8})^{1/(\ell+4)}} \right)^{\ell/4} \]

\[ \leq \frac{1}{\kappa_{\ell}^{\ell/4}}. \]

Therefore we have

\[ \int_{\frac{1}{2}}^{1} \left( \frac{x(1-x)}{\varphi_{\ell, \epsilon}(x)} \right)^{\ell/4} dx \leq \int_{\frac{1}{2}}^{1} \left( \frac{x(1-x)}{\varphi_{\ell}(x)} \right)^{\ell/4} dx + \frac{\epsilon}{\kappa_{\ell}^{\ell/4}} = \frac{1}{2} \text{Ind}^{\ell}(s_{\ell, \text{Opt}}) + \frac{\epsilon}{\kappa_{\ell}^{\ell/4}}, \]

so

\[ \limsup_{\epsilon \to 0} 2 \int_{\frac{1}{2}}^{1} \left( \frac{x(1-x)}{\varphi_{\ell, \epsilon}(x)} \right)^{\ell/4} \leq \text{Ind}^{\ell}(s_{\ell, \text{Opt}}). \]

It therefore follows that \( \limsup_{\epsilon \to 0} \text{Ind}^{\ell}(s_{\ell}) \leq \text{Ind}^{\ell}(s_{\ell, \text{Opt}}) \). But in fact, the inequality is an equality; this is because no continuously differentiable function has incentivization index less than that of \( s_{\ell, \text{Opt}} \). This completes the proof of Theorem 3.6.1.
## A.6 Simulation results

<table>
<thead>
<tr>
<th>Cost</th>
<th>Rule</th>
<th>Avg. Error</th>
<th>Predicted Avg. Error</th>
<th>Ratio</th>
<th>Avg. # Flips</th>
<th>Max. # Flips</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>$s_{\text{quad}}$</td>
<td>0.1616</td>
<td>0.1490</td>
<td>1.0845</td>
<td>2.3341</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>$s_{\text{log}}$</td>
<td>0.1609</td>
<td>0.1389</td>
<td>1.1582</td>
<td>2.3317</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>$s_{1,\text{Opt}}$</td>
<td>0.1553</td>
<td>0.1348</td>
<td>1.1522</td>
<td>2.6658</td>
<td>3</td>
</tr>
<tr>
<td>0.03</td>
<td>$s_{\text{quad}}$</td>
<td>0.1136</td>
<td>0.1103</td>
<td>1.0298</td>
<td>6.0933</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>$s_{\text{log}}$</td>
<td>0.1093</td>
<td>0.1028</td>
<td>1.0636</td>
<td>6.7133</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>$s_{1,\text{Opt}}$</td>
<td>0.1110</td>
<td>0.0997</td>
<td>1.1126</td>
<td>7.1547</td>
<td>10</td>
</tr>
<tr>
<td>0.01</td>
<td>$s_{\text{quad}}$</td>
<td>0.0850</td>
<td>0.0838</td>
<td>1.0147</td>
<td>11.9745</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>$s_{\text{log}}$</td>
<td>0.0816</td>
<td>0.0781</td>
<td>1.0444</td>
<td>13.2780</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>$s_{1,\text{Opt}}$</td>
<td>0.0802</td>
<td>0.0758</td>
<td>1.0577</td>
<td>15.3399</td>
<td>23</td>
</tr>
<tr>
<td>0.003</td>
<td>$s_{\text{quad}}$</td>
<td>0.0626</td>
<td>0.0620</td>
<td>1.0096</td>
<td>23.2076</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>$s_{\text{log}}$</td>
<td>0.0590</td>
<td>0.0578</td>
<td>1.0199</td>
<td>26.3918</td>
<td>27</td>
</tr>
<tr>
<td></td>
<td>$s_{1,\text{Opt}}$</td>
<td>0.0580</td>
<td>0.0561</td>
<td>1.0349</td>
<td>31.2093</td>
<td>52</td>
</tr>
<tr>
<td>0.001</td>
<td>$s_{\text{quad}}$</td>
<td>0.0472</td>
<td>0.0471</td>
<td>1.0014</td>
<td>41.5592</td>
<td>52</td>
</tr>
<tr>
<td></td>
<td>$s_{\text{log}}$</td>
<td>0.0448</td>
<td>0.0439</td>
<td>1.0193</td>
<td>47.4845</td>
<td>48</td>
</tr>
<tr>
<td></td>
<td>$s_{1,\text{Opt}}$</td>
<td>0.0434</td>
<td>0.0426</td>
<td>1.0179</td>
<td>57.5323</td>
<td>107</td>
</tr>
<tr>
<td>0.0003</td>
<td>$s_{\text{quad}}$</td>
<td>0.0349</td>
<td>0.0349</td>
<td>1.0016</td>
<td>77.2931</td>
<td>97</td>
</tr>
<tr>
<td></td>
<td>$s_{\text{log}}$</td>
<td>0.0329</td>
<td>0.0325</td>
<td>1.0113</td>
<td>89.2927</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>$s_{1,\text{Opt}}$</td>
<td>0.0320</td>
<td>0.0315</td>
<td>1.0130</td>
<td>108.8362</td>
<td>230</td>
</tr>
<tr>
<td>0.0001</td>
<td>$s_{\text{quad}}$</td>
<td>0.0265</td>
<td>0.0265</td>
<td>0.9999</td>
<td>134.6477</td>
<td>171</td>
</tr>
<tr>
<td></td>
<td>$s_{\text{log}}$</td>
<td>0.0248</td>
<td>0.0247</td>
<td>1.0047</td>
<td>157.1403</td>
<td>158</td>
</tr>
<tr>
<td></td>
<td>$s_{1,\text{Opt}}$</td>
<td>0.0241</td>
<td>0.0240</td>
<td>1.0043</td>
<td>192.5842</td>
<td>460</td>
</tr>
<tr>
<td>0.00003</td>
<td>$s_{\text{quad}}$</td>
<td>0.0196</td>
<td>0.0196</td>
<td>1.0013</td>
<td>247.0952</td>
<td>314</td>
</tr>
<tr>
<td></td>
<td>$s_{\text{log}}$</td>
<td>0.0184</td>
<td>0.0183</td>
<td>1.0056</td>
<td>289.8925</td>
<td>291</td>
</tr>
<tr>
<td></td>
<td>$s_{1,\text{Opt}}$</td>
<td>0.0177</td>
<td>0.0177</td>
<td>0.9980</td>
<td>356.4421</td>
<td>979</td>
</tr>
</tbody>
</table>

Table A.1: Results of a simulation showing the behavior of a locally adaptive expert for small-to-medium costs $c$. “Predicted average error” means the error that Theorem 3.3.3 predicts in the limit as $c$ approaches 0 (but multiplied by $c^{-1/4}$ in the stated value of $c$). “Ratio” refers to the ratio between the average error and the predicted average error. “Maximum number of flips” refers to the maximum number of flips for that cost and rule in the 100,000 simulations. (Note that because of the large number of simulations and comparatively small number of flips, these are likely to be universal upper bounds on the number of flips given that cost and rule.)
Appendix B: Details omitted from Chapter 5

B.1 Details omitted from Section 5.4

Theorem 5.4.5. Let $s$ be a bounded proper scoring rule with convex exposure and forecast domain $D$. For time steps $t = 1 \ldots T$, an agent chooses a weight vector $w^t \in \Delta_m$. The agent then receives a score of

$$s \left( \bigoplus_{i=1}^{m} g(p^t_i, w^t_i); j^t \right),$$

where $p^t_1, \ldots, p^t_m \in D$ and $j^t \in [n]$ are chosen adversarially. By choosing $w^t$ according to Algorithm 5.4.6 (online gradient descent on the experts’ weights), the agent achieves $O(\sqrt{T})$ regret in comparison with the best weight vector in hindsight. In particular, if $M$ is an upper bound\footnote{This bound $M$ exists because $s$ is bounded by assumption, and so $g$ is also bounded (this follows from Equation 5.2).} on $\|g\|_2$, then for every $w^* \in \Delta_m$ we have

$$\sum_{t=1}^{T} s \left( \bigoplus_{i=1}^{m} g(p^t_i, w^t_i); j^t \right) - s \left( \bigoplus_{i=1}^{m} g(p^*_i, w^*_i); j^t \right) \leq 3\sqrt{mM \sqrt{T}}.$$

Proof. We apply Theorem 3.1 of Hazan (2023); this theorem tells us that in order to prove the stated bound, it suffices to show that for all $t$ and $w$, $\|\nabla L^t(w)\|_2 \leq \sqrt{2mM}$. Let $L$ be an arbitrary loss function, i.e. $L(w) = -WS_j(w)$ for some $j, p_1, \ldots, p_m$. Let $p^*(w) = \bigoplus_{i=1}^{m} g(p_i, w_i)$. We claim that

$$\nabla L(w) = \begin{pmatrix} -g(p_1) \\ \vdots \\ -g(p_m) \end{pmatrix} \left( p^*(w) - \delta_j \right),$$

(B.1)
where this \( m \)-dimensional vector should be interpreted modulo translation by \( \mathbf{I}_m \) (see Remark 5.2.2). To see this, observe that

\[
\nabla L(w) = -\nabla WS_j(w) = -\nabla_w s(p^*(w); j) = -\nabla_w \left(G(p^*(w)) + \langle g(p^*(w)), \delta_j - p^*(w) \rangle \right),
\]

where \( \nabla_w \) denotes the gradient with respect to change in the weight vector \( w \) (as opposed to change in the probability vector). Now, by the chain rule for gradients, we have

\[
\nabla_w G(p^*(w)) = (J_{p^*}(w))^{\top} g(p^*(w)),
\]

where \( J_{p^*} \) denotes the Jacobian matrix of the function \( p^*(w) \). Also, we have

\[
g(p^*(w)) = \sum_{i=1}^{m} w_i g(p_i),
\]

so (again by the chain rule) we have

\[
\nabla_w \langle g(p^*(w)), \delta_j - p^*(w) \rangle = \begin{pmatrix} g(p_1) \\ \vdots \\ g(p_m) \end{pmatrix} (\delta_j - p^*(w)) - (J_{p^*}(w))^{\top} g(p^*(w)).
\]

This gives us Equation B.1.\(^2\) Now, for any \( i \), we have

\[
|\langle g(p_i), p^*(w) - \delta_j \rangle| \leq \|g(p_i)\|_2 \|p^*(w) - \delta_j\|_2 \leq \sqrt{2}M.
\]

Therefore,

\[
\|\nabla L(w)\|_2 \leq \sqrt{m \cdot \left( \sqrt{2}M \right)^2} = \sqrt{2mM},
\]

completing the proof.

\(^2\)Note that the cancellation of the Jacobian terms stems not from the specific relationship between \( p^* \) and \( w \) but from the nature of proper scoring rules. We obtain the same cancellation if we consider \( \nabla s(p; j) \), where after differentiating \( G(p) + \langle g(p), \delta_j - p \rangle \) we find that the \( g(p) \) terms cancel.
B.2 Details omitted from Section 5.6

We claim that our axioms in Definition 5.6.2 can be restated equivalently in a form similar to that of Kolmogorov introduced at the top of Section 5.6 (though with weights.)

Claim B.2.1. Given a pooling operator $\oplus$ on $D$ satisfying Definition 5.6.2, the function $M$ defined on arbitrary tuples of weighted forecasts defined by $M(\Pi_1, \ldots, \Pi_m) := \bigoplus_{i=1}^{m} \Pi_i$ satisfies the following axioms:

1. $M(\Pi_1, \ldots, \Pi_m)$ is strictly increasing in each $pr(\Pi_i)$ and continuous in its inputs.\(^{3}\)
2. $M$ is symmetric in its arguments.
3. $M((p, w_1), \ldots, (p, w_m)) = (p, \sum_i w_i)$.
4. $M(\Pi_1, \ldots, \Pi_k, \Pi_{k+1}, \ldots, \Pi_m) = M(\Pi', \Pi_{k+1}, \ldots, \Pi_m)$, where $y := M(\Pi_1, \ldots, \Pi_k)$.
5. $M((p_1, w_1), \ldots, (p_m, w_m))$ has weight $w_1 + \cdots + w_m$.

Additionally, given any $M$ defined on arbitrary tuples of weighted forecasts, the operator $\oplus$ defined by $\Pi_1 \oplus \Pi_2 := M(\Pi_1, \Pi_2)$ satisfies Definition 5.6.2.

Proof. We first prove that given $\oplus$ satisfying Definition 5.6.2, $M$ satisfies the stated axioms. The last four axioms are clear, so we prove the first one. The fact that $M$ is strictly increasing in each probability follows immediately by considering the continuous, strictly increasing function $g$ such that $\oplus = \oplus_g$, which exists by Theorem 5.6.6. Continuity likewise follows, since the quantity in Definition 5.6.2 is continuous.

We now prove that given $M$ satisfying the stated axioms, $\oplus$ satisfies Definition 5.6.2. Weight additivity, commutativity, continuity, and idempotence are clear. To prove associativity, note that

$$\Pi_1 \oplus (\Pi_2 \oplus \Pi_3) = M(\Pi_1, M(\Pi_2, \Pi_3)) = M(\Pi_1, \Pi_2, \Pi_3) = M(M(\Pi_1, \Pi_2), \Pi_3) = (\Pi_1 \oplus \Pi_2) \oplus \Pi_3.$$

To prove monotonicity, let $p_1 > p_2$ and $w > x > y$. We wish to prove that $pr((p_1, x) \oplus (p_2, w - \ldots$\(^{3}\) That is, it is a continuous function of its input in $D^m \times (\mathbb{R}_0^m \setminus 0)$, where weighted forecasts with weight 0 are ignored when computing $M$.\)
\[ \text{We have} \]
\[
\begin{align*}
\text{pr}((p_1, x) \oplus (p_2, w-x)) &= \text{pr}(M((p_1, x), (p_2, w-x))) = \text{pr}(M((p_1, y), (p_1, x-y), (p_2, w-x))) \\
&> \text{pr}(M((p_1, y), (p_2, x-y), (p_2, w-x))) = \text{pr}(M((p_1, y), (p_2, w-y))) \\
&= \text{pr}((p_1, y) \oplus (p_2, w-y)).
\end{align*}
\]

\[ \square \]

**B.2.1 Extending the results of Section 5.6 to \( n > 2 \) outcomes**

We now discuss extending our axiomatization to arbitrary values of \( n \) in a way that, again, describes the class of QA pooling operators. Just as we fixed a two-outcome forecast domain \( \mathcal{D} \) in Section 5.6, we now fix an \( n \)-outcome forecast domain \( \mathcal{D} \) for any \( n \geq 2 \). Our definition of weighted forecasts remains the same (except that now \( \text{pr}(\Pi) \) is a vector). Our definition of quasi-arithmetic pooling, however, needs to change to make \( g \) vector-valued. This raises the question: what is the analogue of “increasing” for vector-valued functions? It turns out that the relevant notion for us is *cyclical monotonicity*, introduced by Rockafellar (1970b) (see also Rockafellar (1970a, §27)).

We will define this notion shortly, but first we give the definition of quasi-arithmetic pooling with arbitrary weights (analogous to Definition 5.6.2) for this setting. Throughout this section, we will use the notation \( H_n(c) := \{ \mathbf{x} \in \mathbb{R}^n : \sum_i x_i = c \} \). Recall from Remark 5.2.2 that the range of the gradient of a function defined on \( \mathcal{D} \) is a subset of \( H_n(0) \).

**Definition B.2.2** (Quasi-arithmetic pooling with arbitrary weights). *Given a continuous, strictly cyclical monotone vector-valued function \( g : \mathcal{D} \to H_n(0) \) whose range is a convex set, and weighted forecasts \( \Pi_1 = (p_1, w_1), \ldots, \Pi_m = (p_m, w_m) \), define the quasi-arithmetic pool of \( \Pi_1, \ldots, \Pi_m \) with respect to \( g \) as*

\[
\bigoplus_{i=1}^m g(p_i, w_i) := \left( g^{-1}\left( \frac{\sum_i w_i g(p_i)}{\sum_i w_i} \right), \sum_i w_i \right).
\]

Note that QA pooling as defined in Definition 5.2.5 can be written in the form of Defi-
tion B.2.2 if and only if the scoring rule has convex exposure; if it does not, then for some choices of parameters, $\sum_i w_i g(p_i)$ will be equal to a subgradient – but not the gradient – of $G$ at some point.

**Definition B.2.3** (Cyclical monotonicity). A function $g : U \subseteq \mathbb{R}^n \to \mathbb{R}^n$ is cyclically monotone if for every list of points $x_0, x_1, \ldots, x_{k-1}, x_k = x_0 \in U$, we have

$$\sum_{i=1}^{k} \langle g(x_i), x_i - x_{i-1} \rangle \geq 0.$$ 

We also say that $g$ is strictly cyclically monotone if the inequality is strict except when $x_0 = \cdots = x_{k-1}$.

To gain an intuition for this notion, consider the case of $k = 2$; then this condition says that $\langle g(x_1) - g(x_0), x_1 - x_0 \rangle \geq 0$. In other words, the change in $g$ from $x_0$ to $x_1$ is in the same general direction as the direction from $x_0$ to $x_1$. This property is called 2-cycle (or weak) monotonicity.

Cyclical monotonicity is a stronger notion, and has applications to mechanism design and revealed preference theory (see e.g. Ashlagi et al. (2010), Frongillo and Kash (2014), Lavi and Swamy (2007), and Vohra (2007)). In such settings, it is usually the case that two-cycle and cyclical monotonicity are equivalent. Indeed, Saks and Yu showed that these conditions are equivalent in settings where the set of outcomes (i.e. the range of $g$) is finite (Saks & Yu, 2005). However, cyclical monotonicity is substantially stronger than two-cycle monotonicity when the range of $g$ is infinitely large, as in our setting. In fact, the difference between these two conditions is that a two-cycle monotone function is cyclically monotone if and only if it is also vortex-free (Archer & Kleinberg, 2014, Theorem 3.9). Vortex-freeness means that the path integral of $g$ along any triangle vanishes. See Archer and Kleinberg (2014) for a detailed comparison of these two notions.

The immediately relevant fact for us is that cyclically monotone functions are gradients of convex functions (and vice versa). Speaking more precisely:
**Theorem B.2.4.** A vector-valued function $g$ is continuous and strictly cyclically monotone if and only if it is the gradient of a differentiable, strictly convex function $G$.

**Proof.** Per a theorem of Rockafellar (Rockafellar (1970b), see also Rockafellar (1970a, Theorem 24.8)), a function $g$ is cyclically monotone if and only if it is a subgradient of a convex function $G$. The proof of this fact shows just as easily that a function is strictly cyclically monotone if and only if it is a subgradient of a strictly convex function.

Consider a differentiable, strictly convex function $G$. Its gradient is continuous (see Rockafellar (1970a, Theorem 25.5)). Conversely, consider a continuous, strictly cyclically monotone vector-valued function $g$. As we just discussed, it is a subgradient of some strictly convex function $G$. A convex function with a continuous subgradient is differentiable (Bauschke & Combettes, 2011, Proposition 17.41).

This means that the conditions on $g$ in Definition B.2.2 are precisely those necessary to let $g$ be any function that it could be in our original definition of quasi-arithmetic pooling (Definition 5.2.5). Our new definition is thus equivalent to the old one (after normalizing weights to add to 1).

We now discuss our axioms for pooling operators that will again capture the class of QA pooling operators. We will keep the weight additivity, commutativity, associativity, and idempotence verbatim from our discussion of the $n = 2$ case. We will slightly strengthen the continuity argument (see below).

We will also add a new axiom, subtraction, which states that if $\Pi_1 \otimes \Pi_2 = \Pi_1 \otimes \Pi_3$ then $\Pi_2 = \Pi_3$. Subtraction in the $n = 2$ case follows from monotonicity; in this case, however, we the subtraction axiom will help us state the monotonicity axiom. In particular, it allows us to make the following definition, which essentially extends the notion of pooling to allow for negative weights.

**Definition B.2.5.** Let $\otimes$ be a pooling operator satisfying weight additivity, commutativity, associativity, and subtraction. Fix $p_1, \ldots, p_k \in D$. Define a function $p : \Delta_k \to D$ (with $p_1, \ldots, p_k$
serving as implicit arguments) defined by

\[ p(w_1, \ldots, w_k) = pr\left( \bigoplus_{i=1}^{k} (p_i, w_i) \right). \]

We extend the definition of \( p \) to a partial function on \( H_k(1) \), as follows: given input \( (w_1, \ldots, w_k) \), let \( S \subseteq [k] \) be the set of indices \( i \) such that \( w_i < 0 \) and \( T \subseteq [k] \) be the set of indices \( i \) such that \( w_i > 0 \). We define \( p(w_1, \ldots, w_k) \) to be the \( q \in \mathcal{D} \) such that

\[ (q, 1) \oplus \left( \bigoplus_{i \in S} (p_i, -w_i) \right) = \bigoplus_{i \in T} (p_i, w_i). \]

Note that \( q \) is not guaranteed to exist, which is why we call \( p \) a partial function. However, if \( q \) exists then it is unique, by the subtraction axiom.

We can now state the full axiomatization, including the monotonicity axiom.

**Definition B.2.6 (Axioms for pooling operators).** For a pooling operator \( \oplus \) on \( \mathcal{D} \), we define the following axioms.

1. **Weight additivity:** \( wt(\Pi_1 \oplus \Pi_2) = wt(\Pi_1) + wt(\Pi_2) \) for every \( \Pi_1, \Pi_2 \).
2. **Commutativity:** \( \Pi_1 \oplus \Pi_2 = \Pi_2 \oplus \Pi_1 \) for every \( \Pi_1, \Pi_2 \).
3. **Associativity:** \( \Pi_1 \oplus (\Pi_2 \oplus \Pi_3) = (\Pi_1 \oplus \Pi_2) \oplus \Pi_3 \) for every \( \Pi_1, \Pi_2, \Pi_3 \).
4. **Continuity:** For every positive integer \( k \) and \( p_1, \ldots, p_k \), the quantity\(^4\)

\[ pr\left( \bigoplus_{i=1}^{k} (p_i, w_i) \right) \]

is a continuous function of \( (w_1, \ldots, w_k) \) on \( \mathbb{R}_{\geq 0}^k \setminus \{0\} \).

5. **Idempotence:** For every \( \Pi_1 \) and \( \Pi_2 \), if \( pr(\Pi_1) = pr(\Pi_2) \) then \( pr(\Pi_1 \oplus \Pi_2) = pr(\Pi_1) \).
6. **Subtraction:** If \( \Pi_1 \oplus \Pi_2 = \Pi_1 \oplus \Pi_3 \) then \( \Pi_2 = \Pi_3 \).

\(^4\)The continuity axiom is only well-defined conditioned on \( \oplus \) being associative, which is fine for our purposes. We allow a proper subset of weights to be zero by defining the aggregate to ignore forecasts with weight zero.
7. **Monotonicity:** There exist vectors $p_1, \ldots, p_n \in D$ such that $p$ (as in Definition B.2.5) is a strictly cyclically monotone function from its domain to $\mathbb{R}^n$.

This monotonicity axiom essentially extends our previous monotonicity axiom (in Definition 5.6.5) to a multi-dimensional setting. It states that there are $n$ “anchor points” in $D$ such that the function $p$ from weight vectors to $D$ that pools the anchor points with the weights given as input obeys a notion of monotonicity (namely cyclical monotonicity). Informally, this means that the vector of weights that one would need to give to the anchor points in order to arrive at a forecast $p$ “correlates” with the forecast $p$ itself.

We now state the main theorem of our axiomatization.

**Theorem B.2.7.** A pooling operator satisfies the axioms in Definition B.2.6 if and only if it is a QA pooling operator as in Definition B.2.2.\(^5\)

**Proof.** We begin by noting the following fact, which follows from results in Rockafellar (1970a, §26).

**Proposition B.2.8.** A strictly cyclically monotone function $g : D \rightarrow \mathbb{R}^n$ is injective, and its inverse $g^{-1}$ is strictly cyclically monotone and continuous.\(^6\)

We provide a partial proof below; it relies on the following observation.

**Remark B.2.9.** We can instead write the condition as

$$\sum_{i=1}^{k} \langle g(x_i) - g(x_{i-1}), x_i \rangle \geq 0.$$

\(^5\) Recall that Definition B.2.2 is narrower than Definition 5.2.5, since it excludes QA pools with respect to scoring rules that do not have convex exposure. Without convex exposure, the associativity, subtraction, and monotonicity axioms may be violated. For example, consider $\Pi_i = (\delta_i, 1)$ for $i = 1, 2, 3$, for the scoring rule with expected score function $G(x) = x_1^4 + x_2^4 + x_3^4$. We have that $(\Pi_1 \oplus_g \Pi_2) \oplus_g \Pi_3 = ((1/2, 1/2, 0), 2) \oplus_g (\delta_3, 1) \approx ((0.182, 0.182, 0.635), 3)$, whereas $\Pi_1 \oplus_g (\Pi_2 \oplus_g \Pi_3) = ((0.635, 0.182, 0.182), 3)$.

\(^6\) Why can’t we apply this result again to $g^{-1}$ to conclude that $g$ is continuous, even though we did not assume it to be? The reason is that the proof of continuity relies on the convexity of $D$; if $g$ is discontinuous then the domain of $g^{-1}$ may not be convex (or even connected), so we cannot apply the result to $g^{-1}$.
This is equivalent to the condition in Definition B.2.3, because it is the same statement (with rearranged terms) when the $x_i$'s are listed in reverse order.

Proof. First, suppose that $g(x) = g(y)$. Then

$$g(x)(x - y) + g(y)(y - x) = 0.$$ 

Since $g$ is strictly cyclically monotone, this implies that $x = y$. (Note that we only use two-cycle monotonicity.)

We now show that $g^{-1}$ is strictly cyclically monotone. That is, we wish to show that

$$\sum_{i=1}^{k} \langle x_i, g^{-1}(x_i) - g^{-1}(x_{i-1}) \rangle > 0$$

for any $x_1, \ldots, x_k = x_0$ that are not all the same. (See Remark B.2.9.) By the cyclical monotonicity of $g$, we have that

$$\sum_{i=1}^{k} \langle g(p_i), p_i - p_{i-1} \rangle > 0$$

(the strictness of the inequality follows by the injectivity of $g$: if $x_i \neq x_j$ then $p_i \neq p_j$). This means that

$$\sum_{i=1}^{k} \langle x_i, g^{-1}(x_i) - g^{-1}(x_{i-1}) \rangle > 0,$$

as desired. As for continuity, we defer to Rockafellar (1970a, Theorem 26.5). 

Back to the proof of Theorem B.2.7, we first prove that any such $\oplus_g$ satisfies the stated axioms. Weight additivity, commutativity, associativity, and idempotence are clear. Continuity follows from the formula

$$\text{pr}((p_1, w_1) \oplus_g (p_2, w_2)) = g^{-1} \left( \frac{w_1 g(p_1) + w_2 g(p_2)}{w_1 + w_2} \right),$$

noting that $g^{-1}$ is continuous by Proposition B.2.8. Likewise, subtraction follows from the fact that $g$ is injective (by Proposition B.2.8), as is $g^{-1}$ (likewise). Monotonicity remains.
The range of \( g \) contains an open subset\(^7\) of \( H_n(0) \), so in particular it contains the vertices of some translated and dilated copy of the standard simplex. That is, there are \( n \) points \( x_1, \ldots, x_n \) in the range of \( g \) for which there is a positive scalar \( a \) and vector \( b \) such that \( a\delta_i + b = x_i \) for every \( i \). (Here \( \delta_i \) is the \( i \)-th standard basis vector in \( \mathbb{R}^n \).) We will let \( p_i \) be the pre-image of \( x_i \) under \( g \), so that \( g(p_i) = a\delta_i + b \).

Observe that for any \( w \) in the domain of \( p \), we have

\[
g(p(w)) = \sum_{i=1}^{n} w_i g(p_i) = \sum_{i=1}^{n} w_i (a\delta_i + b) = aw + b,
\]

so

\[
p(w) = g^{-1}(aw + b).
\]

We have that \( g^{-1} \) is strictly cyclically monotone (by Proposition B.2.8), and it is easy to verify that for any strictly cyclically monotone function \( f \) and any \( a > 0 \) and \( b \), \( f(ax + b) \) is a strictly cyclically monotone function of \( x \). Therefore, \( p(w) = g^{-1}(aw + b) \) is strictly cyclically monotone, as desired.

Now we prove the converse. Assume that we have a pooling operator \( \oplus \) satisfying the axioms in Definition B.2.6. We wish to show that \( \oplus \) is \( \oplus_g \) for some \( g : D \to H_n(0) \).

For the remainder of this proof, let \( p_1, \ldots, p_n \) be vectors certifying the monotonicity of \( \oplus \), and let \( p(\cdot) \) be as in Definition B.2.5.

For any \( q \in D \), let \( g(q) := w - \frac{1}{n}1_n \), where \( w \in H_n(1) \) is such that \( p(w) = q \) and \( 1_n \) is the all-ones vector. This raises the question of well-definedness: does this \( w \) necessarily exist, and if so, is it unique? The following claim shows that this is indeed the case.

**Claim B.2.10.** The function \( p \), from the subset of \( H_n(1) \) where it is defined to \( D \), is bijective.

**Proof.** The fact that \( p \) is injective follows from the fact that it is strictly cyclically monotone (see Proposition B.2.8). We now show that \( p \) is surjective.

---

\(^7\)This follows from the invariance of domain theorem, which states that the image of an open subset of a manifold under an injective continuous map is open.
Let \( q \in \mathcal{D} \). Define the function \( \tilde{p} : \Delta_{n+1} \to \mathcal{D} \) by

\[
\tilde{p}(w_1, \ldots, w_{n+1}) := \text{pr} \left( \bigoplus_{i=1}^{n} (p_i, w_i) \oplus (q, w_{n+1}) \right).
\]

Since \( \tilde{p} \) is a continuous map\(^8\) from \( \Delta_{n+1} \) (an \( n \)-dimensional manifold) to \( \mathcal{D} \) (an \((n-1)\)-dimensional manifold), \( \tilde{p} \) is not injective.\(^9\) So in particular, let \( w_1 \neq w_2 \in \Delta_{n+1} \) be such that \( \tilde{p}(w_1) = \tilde{p}(w_2) \).

That is, we have

\[
\left( \bigoplus_{i=1}^{n} (p_i, w_{1,i}) \right) \oplus (q, w_{1,n+1}) = \left( \bigoplus_{i=1}^{n} (p_i, w_{2,i}) \right) \oplus (q, w_{2,n+1}).
\]

(B.2)

Observe that \( w_{1,n+1} \neq w_{2,n+2} \); for otherwise it would follow from the subtraction axiom that two different combinations of the \( p_i \)'s would give the same probability, contradicting the fact that \( p \) is injective. Without loss of generality, assume that \( w_{1,n+1} > w_{2,n+1} \). We can rearrange the terms in Equation B.2 to look as follows.

\[
(q, w_{1,n+1} - w_{2,n+1}) \oplus \left( \bigoplus_{i \in S} (p_i, v_i) \right) = \bigoplus_{i \in T \subseteq [n] \setminus S} (p_i, v_i)
\]

for some positive \( v_1, \ldots, v_n \). By the distributive property, we may multiply all weights by \( \frac{1}{w_{1,n+1} - w_{2,n+1}} \).

The result will be an equation as in Definition B.2.5, certifying that \( q \) is in the range of the function \( p \), as desired. \( \square \)

We return to our main proof, now that we have shown that our function \( g(q) := w - \frac{1}{n} \mathbf{1} \), where \( w \in H_n(1) \) is such that \( p(w) = q \), is well-defined. In fact, we can simply write \( g(q) = p^{-1}(q) - \frac{1}{n} \mathbf{1} \).

(The vector \( \frac{1}{n} \mathbf{1} \) is fairly arbitrary; it only serves the purpose of forcing the range of \( g \) to lie in \( H_n(0) \) instead of \( H_n(1) \).)

We first show that the equation that defines \( \oplus_q \) holds – that is, that \( (q_1, v_1) \oplus (q_2, v_2) = \)

---

\(^8\)By the continuity axiom; here we use the more generalized form we stated earlier.

\(^9\)This follows e.g. from the Borsuk-Ulam theorem.
\((q, v_1 + v_2)\) (with \(v_1, v_2 \geq 0\), not both zero), then

\[
g(q) = \frac{v_1 g(q_1) + v_2 g(q_2)}{v_1 + v_2}.
\]

Let \(w_1, w_2 \in H_n(1)\) be such that \(q_1 = p(w_1)\) and \(q_2 = p(w_2)\). It is intuitive that \(q = p \left( \frac{v_1 w_1 + v_2 w_2}{v_1 + v_2} \right)\), but we show this formally.

**Claim B.2.11.** Given \(q_1, q_2 \in \mathcal{D}\) with \(q_1 = p(w_1)\), \(q_2 = p(w_2)\), and \(0 \leq \alpha \leq 1\), we have

\[
p(\alpha w_1 + (1 - \alpha) w_2) = (q_1, \alpha) \oplus (q_2, 1 - \alpha).
\]

**Proof.** Note that

\[
(q_1, 1) \oplus \bigoplus_{i: w_1, i < 0} (p_i, w_1, i) = \bigoplus_{i: w_1, i > 0} (p_i, w_1, i)
\]

\[
(q_2, 1) \oplus \bigoplus_{i: w_2, i < 0} (p_i, w_2, i) = \bigoplus_{i: w_2, i > 0} (p_i, w_2, i).
\]

Applying the distributive property to the two above equations with constants \(\alpha\) and \(1 - \alpha\), respectively, and adding them, we get that

\[
(q_1, \alpha) \oplus (q_2, 1 - \alpha) \oplus \bigoplus_{i: w_1, i < 0} (p_i, \alpha w_1, i) \oplus \bigoplus_{i: w_2, i < 0} (p_i, (1 - \alpha) w_2, i)
\]

\[
= \left( \bigoplus_{i: w_1, i > 0} (p_i, \alpha w_1, i) \right) \oplus \left( \bigoplus_{i: w_2, i > 0} (p_i, (1 - \alpha) w_2, i) \right).
\]

We have that \((q_1, \alpha) \oplus (q_2, 1 - \alpha) = (q, 1)\). It follows (after rearranging terms, from Definition B.2.5) that \(q = p \left( \frac{v_1 w_1 + v_2 w_2}{v_1 + v_2} \right)\). \(\square\)
Applying Claim B.2.11 with $\alpha = \frac{v_1}{v_1 + v_2}$, we find that

$$g(q) = \frac{v_1 w_1 + v_2 w_2}{v_1 + v_2} - \frac{1}{n} = \frac{v_1 \left(g(q_1) + \frac{1}{n} \right) + v_2 \left(g(q_2) + \frac{1}{n} \right)}{v_1 + v_2} - \frac{1}{n} = \frac{v_1 g(q_1) + v_2 g(q_2)}{v_1 + v_2},$$

as desired.

It remains to show that $g$ is continuous, strictly cyclically monotone, and has convex range. By the monotonicity axiom, $p$ is strictly cyclically monotone. It follows by Proposition B.2.8 that its inverse is continuous and strictly cyclically monotone. Therefore, $g$ is continuous and cyclically monotone (as it is simply a translation of $p^{-1}(q)$ by $\frac{1}{n}1$).

Finally, to show that $g$ has convex range, we wish to show that $p^{-1}$ has convex range; or, in other words, that the domain on which $p$ is defined is convex. And indeed, this follows straightforwardly from Claim B.2.11. Let $w_1, w_2$ be in the domain of $p$, with $p(w_1) = q_1, p(w_2) = q_2$. Then for any $0 \leq \alpha \leq 1$, we have that

$$p(\alpha w_1 + (1-\alpha)w_2) = (q_1, \alpha) \oplus (q_2, 1-\alpha),$$

so in particular $\alpha w_1 + (1-\alpha)w_2$ is in the domain of $p$. This concludes the proof.

\[\square\]

### B.3 The convex exposure property

Several of our results have been contingent on the convex exposure property. In this e-companion, we consider when the convex exposure property holds. Our first result is that it always holds in the case of a binary outcome (i.e. $n = 2$).

**Proposition B.3.1.** If $n = 2$, every (continuous) proper scoring rule has convex exposure.

**Proof.** Consider a proper scoring rule $s$ with forecast domain $\mathcal{D}$. Since $\mathcal{D}$ is connected and $g$ is continuous on $\mathcal{D}$, the range of $g$ over $\mathcal{D}$ is connected. In the $n = 2$ outcome case, the range of $g$
lies on the line \((x_1, x_2) : x_1 + x_2 = 0\), and a connected subset of a line is convex.

As we shall see, the convex exposure property holds for nearly all of the most commonly used scoring rules even in higher dimensions.

(A note on notation: in this section we use \(p_j\) instead of \(p(j)\) to refer to the \(j\)-th coordinate of a probability distribution \(p\).)

We now show that scoring rules that – like the logarithmic scoring rule – “go off to infinity” have convex exposure.

**Proposition B.3.2.** Let \(s\) be a proper scoring rule whose forecast domain is the interior of \(\Delta_n\), such that for any point \(x\) on the boundary of \(\Delta_n\), and for any sequence \(x_1, x_2, \ldots\) converging to \(x\),

\[
\lim_{k \to \infty} \|g(x_k)\|_2 = \infty.
\]

Then \(s\) has convex exposure.

This is a statement of convex analysis – namely that if \(\|g\|_2\) approaches \(\infty\) on the boundary of a convex set, then the range of \(g\) is convex (assuming \(g\) is the gradient of a differentiable convex function). See Rockafellar (1970a, Theorem 26.5) for the proof. In non-pathological cases, the basic intuition is that every \(v \in \{x : \sum_i x_i = 0\}\) is the gradient of \(G\) at some point. In these cases, \(\nabla G(x) = v\) where \(x\) minimizes \(G(v) - \langle v, x \rangle\); the \(\lim_{k \to \infty} \|g(x_k)\|_2 = \infty\) condition means that this minimum does not occur on the boundary of \(\Delta_n\).

**Corollary B.3.3.** The following scoring rules have convex exposure:

- The logarithmic scoring rule.
- The scoring rule given by \(G(p) = -\sum_j p_j^{y} \) for \(y \in (0, 1)\).
- The scoring rule given by \(G(p) = -\sum_j \ln p_j\), which can be thought of as the limit of the \(G\) in the previous bullet point as \(y \to 0\).
- The scoring rule \(hs\) given by \(G_{hs}(p) = -\prod_j p_j^{1/n}\).

\(^{10}\)Abernethy et al. (2013) say that \(G\) is a pseudo-barrier function if this condition is satisfied.

\(^{11}\)This is a natural way to think of this scoring rule because \(\nabla G(p) = -(p_1^{-1}, \ldots, p_n^{-1})\).
The *hs* scoring rule  The last of these scoring rules is a generalization of the scoring rule $hs(q) = 1 - \sqrt{\frac{1-q}{q}}$ used by Ben-David and Blais (2020) as part of proving their minimax theorem for randomized algorithms.\textsuperscript{12} The authors used this scoring rule as a key ingredient in their minimax theorem for randomized algorithms. The key property of the scoring rule was a result about its amplification (Ben-David & Blais, 2020, Lemma 3.10). The authors define a forecasting algorithm to be a generalization of a randomized algorithm that outputs an estimated probability that an output should be accepted. Then, roughly speaking, the authors show that given a forecasting algorithm $R$, it is possible to create a forecasting algorithm $R'$ that has a much larger expected score from the scoring rule $hs$ by combining running $R$ a small number of times and combining the outputs. This is an important new result in theoretical computer science and suggests that $hs$ deserves more attention.

Since additive and multiplicative constants are irrelevant, we may treat $hs(q) = -\frac{1}{2} \sqrt{\frac{1-q}{q}}$. Observe that (in the case of two outcomes), the expected score $G_{hs}$ on a report of $q$ is

$$
G_{hs}(q) = q \left( -\frac{1}{2} \sqrt{\frac{1-q}{q}} \right) + (1-q) \left( -\frac{1}{2} \sqrt{\frac{1-q}{q}} \right) = -\sqrt{q(1-q)}.
$$

That is, $G_{hs}$ is precisely negative the geometric mean of $q$ and $1-q$. This motivates us to generalize $hs$ to a setting with $n$ outcomes by setting

$$
G_{hs}(p) := -\prod_{i=1}^{n} p_i^{1/n}.
$$

It should not be obvious that this function is convex, but it turns out to be; this is the precise statement of an inequality known as Mahler’s inequality (Wikipedia contributors, 2018).

Next we note that the quadratic scoring rule has convex exposure, since its exposure function $g(p) = 2p$ (modulo $1_n$ as discussed in Remark 3.12 of the main article) maps any convex set to a convex set.

\textsuperscript{12}Here we are using the shorthand notation for the $n = 2$ outcome case discussed in Remark 3.5 of the main article.
**Proposition B.3.4.** The quadratic scoring rule has convex exposure.

Spherical scoring rules – the third most studied proper scoring rules, after the quadratic and logarithmic rules – also have convex exposure.

**Definition B.3.5** (Spherical scoring rules). (Gneiting & Raftery, 2007, Example 2) For any $\alpha > 1$, define the spherical scoring rule with parameter $\alpha$ to be the scoring rule given by

$$G_{sph,\alpha}(p) := \left( \sum_{i=1}^{n} p_i^\alpha \right)^{1/\alpha}.$$ 

If the “spherical scoring rule” is referenced with no parameter $\alpha$ given, $\alpha$ is presumed to equal 2.

**Proposition B.3.6.** For any $\alpha > 1$, the spherical scoring rule with parameter $\alpha$ has convex exposure.

**Proof.** Fix $\alpha > 1$. We will write $G$ in place of $G_{sph,\alpha}$. We have

$$g(p) = \left( \sum_{j=1}^{n} p_j^\alpha \right)^{(1/\alpha)-1} (p_1^{\alpha-1}, \ldots, p_n^{\alpha-1}). \quad (B.3)$$

Now, define the $n$-dimensional unit $\beta$-sphere, i.e. \( \{x : \sum_j x_j^\beta = 1\} \), and define the $n$-dimensional unit $\beta$-ball correspondingly (i.e. with $\leq$ in place of $=$). The range of $g$ is precisely the part of the $n$-dimensional unit $\frac{\alpha}{\alpha-1}$-sphere with all non-negative coordinates. Indeed, on the one hand, for any $p$ we have

$$\sum_j g_j(p)^{\alpha/(\alpha-1)} = \left( \sum_j p_j^\alpha \right)^{-1} \cdot \sum_j p_j^\alpha = 1$$

(where $g_j(p)$ denotes the $j$-th coordinate of $g(p)$ as in Equation B.3). On the other hand, given a point $x$ on the unit $\frac{\alpha}{\alpha-1}$-sphere with all non-negative coordinates,

$$p = \left( \sum_j x_j^{1/(\alpha-1)} \right)^{-1} \left( x_1^{1/(\alpha-1)}, \ldots, x_n^{1/(\alpha-1)} \right)$$

\[\text{As discussed in Remark 3.12 of the main article, the range of } g \text{ should be thought of as modulo } T(1_n). \text{ However, we find it convenient for this proof to think of it as lying in } \mathbb{R}^n \text{ and project later.}\]
lies in $\Delta_n$ and satisfies $g(p) = x$.

The crucial point for us is that for $\beta > 1$, the unit $\beta$-ball is convex. This means that for any such $\beta$, the convex combination of any number of points on the unit $\beta$-sphere will lie in the unit $\beta$-ball. Since $\frac{\alpha}{\alpha-1} > 1$ for $\alpha > 1$, we have that for arbitrary $p, q \in \Delta_n$ and $w \in [0, 1]$, $wg(p) + (1 - w)g(q)$ lies in the unit $\beta$-ball – in fact, in the part with all non-negative coordinates. Now, consider casting a ray from this convex combination point in the positive $1_n$ direction. All points on this ray are equivalent to this point modulo $T(1_n)$, and this ray will intersect the unit $\beta$-sphere at some point $x$ with all non-negative coordinates. The point $p \in \Delta_n$ with $g(p) = x$ satisfies

$$g(p) = g(p) + (1 - w)g(q).$$

This completes the proof.

**Remark B.3.7.** The above proof gives a geometric interpretation of the QA pooling with respect to the spherical scoring rule, particularly for $\alpha = 2$. In the $\alpha = 2$ case, pooling amounts to taking the following steps:

1. Scale each forecast so it lies on the unit sphere.
2. Take the weighted average of the resulting points in $\mathbb{R}^n$.
3. Shift the resulting point in the positive $1_n$ direction to the unique point in that direction that lies on the unit sphere.
4. Scale this point so that its coordinates add to 1.

Finally we consider the parametrized family known as Tsallis scoring rules (Tsallis, 1988).

**Definition B.3.8** (Tsallis scoring rules). For $\gamma > 1$, the Tsallis scoring rule with parameter $\gamma$ is the rule given by

$$G_{\text{Tsall}, \gamma}(p) = \sum_{j=1}^{m} p_j^\gamma.$$

Setting $\gamma = 2$ above yields the quadratic scoring rule. Note also that we have already addressed the scoring rule given by $G(p) = \pm \sum_j p_j^\gamma$ for $\gamma \leq 1$ (except $\gamma = 0, 1$, which are degenerate), with
the sign chosen to make $G$ convex: these scoring rules have convex exposure by Proposition B.3.2.

The following proposition completes our analysis for this natural class of scoring rules.

**Proposition B.3.9.** For $\gamma \leq 2$, the Tsallis scoring rule with parameter $\gamma$ has convex exposure. For $\gamma > 2$, this is not the case if $n > 2$.

**Proof.** Fix $\gamma > 1$. We will write $G$ in place of $G_{\text{Tsa},\gamma}$. Up to a multiplicative factor of $\gamma$ that we are free to ignore, we have

$$g(p) = (p_1^{\gamma-1}, \ldots, p_n^{\gamma-1}).$$

Let $p, q \in \Delta_n$ and $w \in [0, 1]$. We wish to find an $x \in \Delta_n$ such that $g(x) = wg(p) + (1-w)g(q)$, i.e.

$$wp_j^{\gamma-1} + (1-w)q_j^{\gamma-1} + c = x_j^{\gamma-1},$$

for all $j \in [n]$, for some $c$. Since $\sum_j x_j = 1$, this $c$ must satisfy

$$\sum_j (wp_j^{\gamma-1} + (1-w)q_j^{\gamma-1} + c)^{1/(\gamma-1)} = 1. \quad (B.4)$$

Let $h(x) := \sum_j (wp_j^{\gamma-1} + (1-w)q_j^{\gamma-1} + x)^{1/(\gamma-1)}$. Note that $h$ is increasing in $x$.

First consider the case that $\gamma \leq 2$. By concavity, we have that $wp_j^{\gamma-1} + (1-w)q_j^{\gamma-1} \leq (wp_j + (1-w)q_j)^{\gamma-1}$. This means that

$$h(0) = \sum_j (wp_j^{\gamma-1} + (1-w)q_j^{\gamma-1})^{1/(\gamma-1)} \leq \sum_j (wp_j + (1-w)q_j) = 1.$$

On the other hand, $\lim_{x \to \infty} h(x) = \infty$. Since $h$ is continuous, there must be some $x \in [0, \infty)$ such that $h(x) = 1$; call this value $c$. Then let

$$x_j = (wp_j^{\gamma-1} + (1-w)q_j^{\gamma-1} + c)^{1/(\gamma-1)}.$$

Then every $x_j$ is nonnegative and $\sum_j x_j = 1$, so we have succeeded.
Now consider the case that $\gamma > 2$, and consider as a counterexample $p = (1, 0, \ldots, 0)$, $q = (0, 1, 0, \ldots, 0)$, and $w = \frac{1}{2}$. To satisfy Equation B.4, we are looking for $c$ such that

$$h(c) = 2 \left( \frac{1}{2} + c \right)^{1/(\gamma - 1)} + (n - 2)c^{1/(\gamma - 1)} = 1.$$ 

Note that $h(0) = 2 \cdot 2^{-1/(\gamma - 1)} = 2^{(\gamma - 2)/(\gamma - 1)} > 1$, so $c < 0$ (as $h$ is increasing). But in that case $x_j^{\gamma - 1} < 0$ for any $j \geq 3$, a contradiction (assuming $n > 2$). \qed

Note that because $\nabla G_{\text{Tsallis}}(p) = (p_1^{\gamma - 1}, \ldots, p_n^{\gamma - 1})$ (up to a constant factor), QA pooling with respect to the Tsallis scoring rule can be thought of as an appropriately scaled coordinate-wise $(\gamma - 1)$-power mean. For $\gamma = 2$ it is the coordinate-wise arithmetic average. For $\gamma = 3$ it is the coordinate-wise root mean square, but with the average of the squares scaled by an appropriate additive constant so that, upon taking the square roots, the probabilities add to 1. (However, as the Tsallis score with parameter 3 does not have convex exposure, this is not always well-defined.)

In Corollary B.3.3 we mentioned that the scoring rule given by $G(p) = -\sum_j \ln p_j$ can be thought of as an extension to $\gamma = 0$ of (what we are now calling) the Tsallis score, because the derivative of $\ln x$ is $x^{-1}$. QA pooling with respect to this scoring rule is, correspondingly, the $-1$-power mean, i.e. the harmonic mean. This pooling method is appropriately referred to as \textit{harmonic pooling}, see e.g. Dawid et al. (1995, §4.2).

Finally, we note that the logarithmic scoring rule can likewise be thought of as an extension of the Tsallis score to $\gamma = 1$, in that the second derivative of $x \ln x$ is $x^{-1}$. It is likewise natural to call the geometric mean the 0-power mean; notice that logarithmic pooling is precisely an appropriately scaled coordinate-wise geometric mean.
Appendix C: Details omitted from Chapter 6

C.1 Efficiency of Algorithm 6.3.3

The only nontrivial step of the algorithm is finding the weight vector satisfying the equation on the last line of the algorithm. To do so, it is first necessary to compute the gradient of the loss. This gradient, given by Equation C.5 below, can clearly be computed in time $O(mn)$. After that, it is necessary to find the weight vector $w^{t+1}$ that satisfies the equation on the last line. This can be done efficiently through local search: the goal amounts to find weights $(w_1, \ldots, w_m)$ such that the vector $(w_1^{\alpha-1}, \ldots, w_m^{\alpha-1})$ is equal to a target vector (call it $v$) plus a constant $c$ times the all-ones vector. That is, we need to simultaneously solve the equation $w_i^{\alpha-1} = v_i + c$ for all $i$, with weights that add to 1. (Here, the $v_i$ are knowns and the $w_i$ and $c$ are unknowns.)

We start by finding $c$, by solving the equation $\sum_i (v_i + c)^{1/(\alpha-1)} = 1$. Such a $c$ exists because the left-hand side of this equation is continuous and monotone decreasing, going from infinity to zero as $c$ ranges from $-\min_i v_i$ to infinity. We can solve for $c$ very efficiently, e.g. with Newton’s method. Once we know $c$, we know each $w_i$: we have $w_i = (v_i + c)^{1/(\alpha-1)}$. Thus, Algorithm 1 takes $O(mn)$ time.

C.2 Details omitted from Section 6.4

Lemma 6.4.4. Consider a particular run of Algorithm 6.3.3. Let $\zeta$ be a constant such that $-\frac{\zeta}{w^t_i} \leq \partial_i L^t(w^t) \leq \zeta$ for all $i, t$. Then for every $i, t$, we have

$$(w^{t}_i)^{\alpha-1} - \left(\frac{1}{w^{t}_i} + 1\right) \eta_t \zeta \leq (w^{t+1}_i)^{\alpha-1} \leq (w^{t}_i)^{\alpha-1} + \left(\frac{1}{\min_k w_k} + 1\right) \eta_t \zeta.$$
Furthermore, if \( \eta_i \zeta \leq (1 - \alpha)^2 (w_i^t)^\alpha \) for all \( i \), then for every \( i \) we have

\[
(w_i^{t+1})^{\alpha-1} \leq (w_i^t)^{\alpha-1} + (m + 1)\eta_i \zeta.
\]

**Proof.** Fix any \( t \). Note that since the space of possible weights is \( \Delta_m \), it is most natural to think of \( \nabla R \) as a function from \( \Delta_m \) to \( \mathbb{R}^m / T(\mathbf{1}_m) \), i.e. \( \mathbb{R}^m \) modulo translation by the all-ones vector (which is orthogonal to \( \Delta_m \) in \( \mathbb{R}^m \)). That is, \( \nabla R(\mathbf{w}) = -((w_1)^{\alpha-1}, \ldots, (w_m)^{\alpha-1}) \), where this vector may be thought of as modulo translation by the all-ones vector. Nevertheless, we find it convenient to define \( \partial_t R(\mathbf{w}) := -(w_i)^{\alpha-1} \). We define \( \partial_t L_i^t(\mathbf{w}) \) similarly (see Section 6.4.4).

Define \( \mathbf{h} \in \mathbb{R}^m \) to have coordinates \( h_i := \partial_t R(\mathbf{w}^t) - \eta_i \partial_t L_i^t(\mathbf{w}^t) \). Per the update rule, we have that \( h_i \equiv R(\mathbf{w}^{t+1}) \mod T(\mathbf{1}_m) \). We have

\[
-(w_i^t)^{\alpha-1} - \eta_i \zeta = \partial_t R(\mathbf{w}^t) - \eta_i \zeta \leq h_i \leq \partial_t R(\mathbf{w}^t) + \frac{\eta_i \zeta}{w_i^t} = -(w_i^t)^{\alpha-1} + \frac{\eta_i \zeta}{w_i^t} \tag{C.1}
\]

Applying the first and last claims of Lemma C.2.1 (below) with \( a = \alpha - 1 \), \( \mathbf{v} = \mathbf{w}^t \), \( \kappa = \eta_i \zeta \), and \( \mathbf{g} = -\mathbf{h} \), we have that there exists a unique \( c \in \mathbb{R} \) such that

\[
\sum_{i=1}^m (-h_i + c)^{1/(\alpha-1)} = 1,
\]

and in fact that \( -\eta_i \zeta \leq c \leq m\eta_i \zeta \). (Equation C.1 is relevant here because it is equivalent to the \( v_i^\alpha - \frac{\kappa}{w_i^t} \leq g_i \leq v_i^\alpha + \kappa \) conditions in Lemma C.2.1. This is also where we use that \( \eta_i \zeta \leq (1 - \alpha)^2 (w_i^t)^\alpha \), which is equivalent to \( \kappa \leq a^2 v_i^{\alpha+1} \).) The significance of this fact is that \( -(h_i + c)^{1/(\alpha-1)} \) is precisely \( w_i^{t+1} \), since (in \( \mathbb{R}^m \)) we have that \( (\partial_t R(\mathbf{w}^{t+1}), \ldots, \partial_t R(\mathbf{w}^{t+1})) = \mathbf{h} - c \cdot \mathbf{1} \) for some \( c \), and in particular this \( c \) must be such that \( \sum_i w_i^{t+1} = 1 \). In particular, this means that for all \( i \), we have

\[
(w_i^{t+1})^{\alpha-1} = -h_i + c \leq (w_i^t)^{\alpha-1} + \eta_i \zeta + \frac{1}{\min_k w_k} \eta_i \zeta = (w_i^t)^{\alpha-1} + \left( \frac{1}{\min_k w_k} + 1 \right) \eta_i \zeta.
\]

Here, the inequality comes from the left inequality of Equation C.1 and the fact that \( c \leq \frac{1}{\min_k w_k} \eta_i \zeta \).
If we also have that $\eta \zeta \leq (1 - \alpha)^2 (w_i')^{a-1}$, then the last claim of Lemma C.2.1 gives us that

$$(w_i')^{a-1} = -h_i + c \leq (w_i')^{a-1} + \eta \zeta + m \eta \zeta = (w_i')^{a-1} + (m + 1) \eta \zeta.$$ 

Similarly, we have

$$(w_i')^{a-1} = -h_i + c \geq (w_i')^{a-1} - \eta \zeta = (w_i')^{a-1} - \left(\frac{1}{w_i'} + 1\right) \eta \zeta.$$ 

Lemma C.2.1. Let $-1 < a < 0$ and $g \in \mathbb{R}^m$. There is a unique $c \in \mathbb{R}$ such that $\sum_i (g_i + c)^{1/a} = 1$.

Furthermore, let $v \in \Delta_m$ and $\kappa \geq 0$. Then:

- If $g_i \leq v_i^a + \kappa$ for all $i$, then $c \geq -\kappa$.
- If $g_i \geq v_i^a - \frac{\kappa}{v_i}$ for all $i$, then $c \leq \frac{\kappa}{\min_i v_i}$.
  
  And if, furthermore, $\kappa \leq a^2 v_i^{a+1}$ for all $i$, then $c \leq m \kappa$.

Proof. Observe that $\sum_i (g_i+c)^{1/a}$ is a continuous, monotone decreasing function on $c \in (-\min_i g_i, \infty)$; the range of the function on this interval is $(0, \infty)$. Therefore, there is a unique $c \in (-\min_i g_i, \infty)$ such that the sum equals 1.

We now prove the first bullet. Since $x^{1/a}$ decreases in $x$ and $g_i \leq v_i^a + \kappa$, we have that

$$1 = \sum_i (g_i + c)^{1/a} \geq \sum_i (v_i^a + \kappa + c)^{1/a}.$$ 

Suppose for contradiction that $c < -\kappa$. Then $v_i^a + \kappa + c < v_i^a$ for all $i$, so

$$\sum_i (v_i^a + \kappa + c)^{1/a} \geq \sum_i (v_i^a)^{1/a} = \sum_i v_i = 1.$$ 

This is a contradiction, so in fact $c \geq -\kappa$.

The first claim of the second bullet is analogous. Since $x^{1/a}$ decreases in $x$ and $g_i \geq v_i^a - \kappa v_i$, 

339
we have that
\[ 1 = \sum_i (g_i + c)^{1/a} \leq \sum_i \left( v_i^a - \frac{\kappa}{v_i} + c \right)^{1/a}. \quad \text{(C.2)} \]
Suppose for contradiction that \( c > \frac{\kappa}{v_i} \) for every \( i \). Then \( v_i^a - \frac{\kappa}{v_i} + c > v_i^a \) for all \( i \), so
\[
\sum_i \left( v_i^a - \frac{\kappa}{v_i} + c \right)^{1/a} < \sum_i (v_i^a)^{1/a} = \sum_i v_i = 1.
\]
This is a contradiction, so in fact \( c \leq \frac{\kappa}{\min_i v_i} \).

We now prove the second claim of the second bullet. To do so, we note the following technical lemma (proof below).

**Lemma C.2.2.** For \(-1 < a < 0\) and \( \kappa, c \geq 0 \), the function \( f(x) = (x^a - \frac{\kappa}{x} + c)^{1/a} \) is defined and concave at any value of \( x > 0 \) such that \( a^2 x^{a+1} \geq \kappa \).

Since for a general concave function \( f \) it holds that \( \frac{1}{m} \sum_{i=1}^{m} f(x_i) \leq f\left( \frac{1}{m} \sum_{i=1}^{m} x_i \right) \), the following inequality follows from Lemma C.2.2:
\[
\sum_i \left( v_i^a - \frac{\kappa}{v_i} + c \right)^{1/a} \leq m \left( \left( \frac{1}{m} \right)^a - \kappa m + c \right)^{1/a}.
\]
(Here we are using the fact that \( \sum_i v_i = 1 \).) Now, combining this fact with Equation C.2, we have that
\[
m \left( \left( \frac{1}{m} \right)^a - \kappa m + c \right)^{1/a} \geq 1
\]
\[
\left( \frac{1}{m} \right)^a - \kappa m + c \leq \left( \frac{1}{m} \right)^a
\]
so \( c \leq m \kappa \), as desired.

**Proof of Lemma C.2.2.** To show that \( f \) is defined for any \( x \) such that \( a^2 x^{a+1} \geq \kappa \), we need to show
that $x^a - \frac{\kappa}{x} + c > 0$ for such values of $x$. This is indeed the case:

$$x^a - \frac{\kappa}{x} + c \geq x^a - a^2x^a + c = (1-a^2)x^a + c > c \geq 0.$$ 

Now we show concavity. We have

$$f''(x) = \frac{1}{-a}\left(\left(1 + \frac{1}{-a}\right)\left(x^a - \frac{k}{x} + c\right)^{1/a-2}\left(ax^{a-1} + \frac{\kappa}{x^2}\right) - \left(x^a - \frac{k}{x} + c\right)^{1/a-1}\left(a(a-1)x^{a-2} - \frac{2\kappa}{x^3}\right)\right)$$

so we wish to show that

$$\left(1 + \frac{1}{-a}\right)\left(x^a - \frac{k}{x} + c\right)^{1/a-2}\left(ax^{a-1} + \frac{\kappa}{x^2}\right)^2 \leq \left(x^a - \frac{k}{x} + c\right)^{1/a-1}\left(a(a-1)x^{a-2} - \frac{2\kappa}{x^3}\right)$$

for every $x$ such that $a^2x^{a+1} \geq \kappa$. Fix any such $x$, and let $d = \frac{\kappa}{x^{a+1}}$ (so $0 \leq d \leq a^2$). We have

$$d \leq a^2$$

$$(1+a)(a^2-d)d \geq 0$$

$$(1-a)(a+d)^2 \leq -a(1-d)(a(a-1) - 2d)$$

(rearrange terms)

$$\left(1 - \frac{1}{a}\right)(a+d)^2x^a \leq ((1-d)x^a)(a(a-1) - 2d)$$

(multiply by $\frac{x^a}{-d}$)

$$\left(1 - \frac{1}{a}\right)(a+d)^2x^a \leq ((1-d)x^a+c)(a(a-1) - 2d)$$

(c$(a(a-1) - 2d) \geq 0$)

$$\left(1 - \frac{1}{a}\right)((a+d)x^{a-1})^2 \leq ((1-d)x^a+c)(a(a-1) - 2d)x^{a-2}$$

(multiply by $x^{a-2}$)

$$\left(1 - \frac{1}{a}\right)\left(ax^{a-1} + \frac{\kappa}{x^2}\right)^2 \leq \left(x^a - \frac{\kappa}{x} + c\right)\left(a(a-1)x^{a-2} - \frac{2\kappa}{x^3}\right).$$

(substitute $d = \kappa x^{-a-1}$)

Note that the fifth line is justified by the fact that $c \geq 0$ and $a(a-1) \geq 2d$ (because $a^2 \geq d$ and $-a > a^2 \geq d$). Now, multiplying both sides by $(x^a - \frac{k}{x} + c)^{1/a-2}$ completes the proof.

**Corollary 6.4.5.** Under the small gradient assumption, for sufficiently large $T$ we have for all $i \in [m], t \in [T]$ that:
(#1) \((w_i^t)^α ≥ 4\eta γ\) and \(w_i^t ≥ \frac{1}{10\sqrt{m}}T^{1/(2(α−1))}\).

(#2) \(-32(w_i^t)^{1−α}γ ≤ w_i^t − w_i^{t+1} ≤ 2(w_i^t)^{2−α}(m + 1)\eta γ\).

Proof. Note that \(\eta γ = \frac{1}{\sqrt{Tm^{(1+α)/2}}}\) and also that \(\eta_t ≤ \eta\) for all \(t\); we will be using these facts.

To prove (#1), we proceed by induction on \(t\). In the case of \(t = 1\), all weights are \(1/m\), so the claim holds for sufficiently large \(T\). Now assume that the claim holds for a generic \(t < T\); we show it for \(t + 1\).

By the small gradient assumption, we may use Lemma 6.4.4 with \(ζ = γ\). By the inductive hypothesis (and the fact that \(\eta_t ≤ \eta\)), we may apply the second part of Lemma 6.4.4:

\[
(w_i^{t+1})^{α−1} ≤ (w_i^t)^{α−1} + (m + 1)\eta γ ≤ \cdots ≤ (1/m)^{α−1} + t(m + 1)\eta γ.
\]

\[
≤ (1/m)^{α−1} + \frac{(T−1)(m + 1)}{m^{(1+α)/2}√T} ≤ 3m^{(1−α)/2}√T.
\]

Since \(-\frac{1}{2} < α - 1 < 0\), this means that \(w_i^t ≥ \frac{1}{10\sqrt{m}}T^{1/(2(α−1))}\).

We also have that

\[
(w_i^{t+1})^α ≥ \frac{1}{(10\sqrt{m})^α}T^{α/(2(α−1))} ≥ \frac{4}{m^{(1+α)/2}}T^{-1/2} = 4\eta γ
\]

for \(T\) sufficiently large, since \(\frac{α}{2(α−1)} > \frac{1}{2}\). This completes the inductive step, and thus the proof of (#1).

To prove (#2), we use the following technical lemma (see below for the proof).

Lemma C.2.3. Fix \(x > 0\) and \(-1 < a < 0\). Let \(f(y) = (x^a + y)^{1/a}\). Then for all \(y > -x^a\), we have

\[
x − f(y) ≤ \frac{−1}{a}x^{1−a}y
\]

(C.3)

and for all \(-1 < c ≤ 0\), for all \(cx^a ≤ y ≤ 0\), we have

\[
f(y) − x ≤ \frac{1}{a}(1 + c)^{1/a−1}x^{1−a}y.
\]

(C.4)
We apply Equation C.3 to $x = w^t_i$, $y = (m + 1)\eta\gamma$, and $a = \alpha - 1$. This tells us that

$$w^t_i - w^{t+1}_i \leq w^t_i - ((w^t_i)^{\alpha-1} + (m + 1)\eta\gamma)^{1/(\alpha-1)} \leq 2(w^t_i)^{2-\alpha}(m + 1)\eta\gamma.$$ 

The first step follows by the second part of Lemma 6.4.4 and the fact that $\eta_t \leq \eta$. The second step follows from Equation C.3 and uses the fact that $\frac{1}{1-\alpha} > 2$.

For the other side of (#2), we observe that since by (#1) we have $(w^t_i)^{\alpha} \geq 4\eta\gamma$, it follows that $\frac{1}{2}(w^t_i)^{\alpha} \geq (w^t_i + 1)\eta\gamma$, and so $\left(\frac{1}{w^t_i + 1}\right)\eta\gamma \leq \frac{1}{2}(w^t_i)^{\alpha-1}$. Therefore, we can apply Equation C.4 to $x = w^t_i$, $y = -(\frac{1}{w^t_i} + 1)\eta\gamma$, $a = \alpha - 1$, and $c = -\frac{1}{2}$. This tells us that

$$w^{t+1}_i - w^t_i \leq \left(\frac{1}{w^t_i + 1}\right)\eta\gamma^{1/(\alpha-1)} - w^t_i \leq 16(w^t_i)^{2-\alpha}\left(\frac{1}{w^t_i + 1}\right)\eta\gamma$$

$$\leq 32(w^t_i)^{1-\alpha}\eta\gamma.$$ 

This completes the proof.

Proof of Lemma C.2.3. For all $y > -x^a$, we have

$$f'(y) = \frac{1}{a}(x^a + y)^{1/a-1}$$

and

$$f''(y) = \frac{1}{a}\left(\frac{1}{a} - 1\right)(x^a + y)^{1/a-2} > 0,$$

so $f'$ is increasing. Thus, for positive values of $y$ we have

$$f'(0) \leq \frac{f(y) - f(0)}{y} = \frac{f(y) - x}{y} \leq f'(y)$$

and for negative values of $y$ we have

$$f'(y) \leq \frac{f(y) - f(0)}{y} = \frac{f(y) - x}{y} \leq f'(0).$$

343
Regardless of whether \( y \) is positive or negative, this means that \( x - f(y) \leq -y f'(0) = \frac{1}{a} x^{1-a} y \).

Now, let \(-1 < c \leq 0\) and suppose that \( c x^a \leq y \leq 0\). Since \( f' \) is increasing, we have that

\[
f'(y) \geq f'(c x^a) = \frac{1}{a} ((1 + c) x^a)^{1/a-1} = \frac{1}{a} (1 + c)^{1/a-1} x^{1-a},
\]

so

\[
f(y) - x \leq y f'(y) \leq \frac{1}{a} (1 + c)^{1/a-1} x^{1-a} y.
\]

\[\Box\]

**Lemma 6.4.8.** For any weight vector \( w \), \( i \in [m] \), and \( \zeta \geq 0 \), we have that

\[
Pr \left[ \partial_i L(w) \geq \zeta \right] \leq ne^{-\zeta}
\]  
(6.2)

and

\[
Pr \left[ \partial_i L(w) \leq -\frac{\zeta}{w_i} \right] \leq mn^2 e^{-\zeta/n}.
\]  
(6.3)

**Proof.** We first derive an expression for \( \partial_i L(w) \) given expert reports \( p_1, \ldots, p_m \), where \( L(w) \) is the log loss of the logarithmic pool \( p^*(w) \) of \( p_1, \ldots, p_m \) with weights \( w \), and \( j \) is the realized outcome. We have\(^1\)

\[
\partial_i L(w) = -\partial_i \ln \frac{\prod_{k=1}^m (p_i^k)^{w_k}}{\sum_{\ell=1}^n \prod_{k=1}^m (p_{\ell}^k)^{w_k}} = \partial_i \ln \left( \sum_{\ell=1}^n \prod_{k=1}^m (p_{\ell}^k)^{w_k} \right) - \partial_i \ln \left( \prod_{k=1}^m (p_i^k)^{w_k} \right)
\]

\[
= \frac{\sum_{\ell=1}^n \ln p_{\ell}^j \cdot \prod_{k=1}^m (p_{\ell}^k)^{w_k}}{\sum_{\ell=1}^n \prod_{k=1}^m (p_{\ell}^k)^{w_k}} - \ln p_j^i = \sum_{\ell=1}^n p_{\ell}^j \ln p_{\ell}^j - \ln p_j^i.
\]  
(C.5)

Equation 6.2 now follows fairly straightforwardly. Equation C.5 tells us that \( \partial_i L(w) \leq -\ln p_j^i \),

\(^1\)It should be noted that \( \nabla L(w) \) is most naturally thought of as living in \( \mathbb{R}^m / T(1_m) \), i.e. \( m \)-dimensional space modulo translation by the all-ones vector, since \( w \) lives in a place that is orthogonal to the all-ones vector. As an arbitrary but convenient convention, we define \( \partial_i L(w) \) to be the specific value derived below, and define the small gradient assumption accordingly.
where $J$ is the random variable corresponding to the realized outcome. Therefore, we have

$$\Pr \left[ \partial_t L(w) \geq \zeta \right] \leq \Pr \left[ - \ln p_j^i \geq \zeta \right] = \Pr \left[ p_j^i \leq e^{-\zeta} \right] = \sum_{j=1}^{n} \Pr \left[ J = j \land p_j^i \leq e^{-\zeta} \right]$$

$$= \sum_{j=1}^{n} \Pr \left[ p_j^i \leq e^{-\zeta} \right] \Pr \left[ J = j \mid p_j^i \leq e^{-\zeta} \right] \leq \sum_{j=1}^{n} \Pr \left[ J = j \mid p_j^i \leq e^{-\zeta} \right] \leq ne^{-\zeta},$$

where the last step follows by the calibration property. This proves Equation 6.2.

We now prove Equation 6.3. The proof has a similar idea, but is somewhat more technical. We begin by proving the following lemma; we again use the calibration property in the proof.

**Lemma C.2.4.** For all $q$, we have

$$\Pr \left[ \forall j \exists i : p_j^i \leq q \right] \leq mnq.$$

**Proof.** Let $J$ be the random variable corresponding to the index of the outcome that ends up happening. We have

$$\Pr \left[ \forall j \exists i : p_j^i \leq q \right] \leq \Pr \left[ \exists i : p_j^i \leq q \right] = \sum_{j \in [n]} \Pr \left[ J = j \land \exists i : p_j^i \leq q \right]$$

$$\leq \sum_{j \in [n]} \sum_{i \in [m]} \Pr \left[ J = j \land p_j^i \leq q \right]$$

$$= \sum_{j \in [n]} \sum_{i \in [m]} \Pr \left[ p_j^i \leq q \right] \Pr \left[ J = j \mid p_j^i \leq q \right] \leq \sum_{j \in [n]} \sum_{i \in [m]} 1 \cdot q = mnq,$$

where the fact that $\Pr \left[ J = j \mid p_j^i \leq q \right] \leq q$ follows by the calibration property. \qed

**Corollary C.2.5.** For any reports $p^1, \ldots, p^n$, weight vector $w$, $i \in [m]$, and $j \in [n]$, we have

$$\Pr \left[ p_j^i(w) \geq \frac{(p_j^i)^{w_i}}{q} \right] \leq mnq.$$
Proof. We have

\[ p_j^*(w) = \frac{\prod_{k=1}^m (p_j^k)^{w_k}}{\sum_{\ell=1}^n \prod_{k=1}^m (p_{\ell k}^k)^{w_k}} \leq \frac{(p_j^i)^{w_i}}{\sum_{\ell=1}^n \prod_{k=1}^m (p_{\ell k}^k)^{w_k}}. \]

Now, assuming that there is an \( \ell \) such that for every \( k \) we have \( p_{\ell k}^k > q \), the denominator is greater than \( q \), in which case we have \( p_j^*(w) < \frac{(p_j^i)^{w_i}}{q} \). Therefore, if \( p_j^*(w) \geq \frac{(p_j^i)^{w_i}}{q} \), it follows that for every \( \ell \) there is a \( k \) such that \( p_{\ell k}^k \leq q \). By Lemma C.2.4, this happens with probability at most \( mnq \).

We now use Corollary C.2.5 to prove Equation 6.3. Note that the equation is trivial for \( \zeta < n \), so we assume that \( \zeta \geq n \). By setting \( q := e^{-\frac{\zeta}{n}} \), we may restate Equation 6.3 as follows: for any \( q \leq \frac{1}{e} \), any \( i \in [m] \), and any weight vector \( w \),

\[ \Pr \left[ \partial_i L(w) \leq -\frac{n \ln 1/q}{w_i} \right] \leq mn^2 q. \]

(Note that the condition \( q \leq \frac{1}{e} \) is equivalent to \( \zeta \geq n \).) We prove this result.

From Equation C.5, we have

\[ \partial_i L(w) = \sum_{j=1}^n p_j^*(w) \ln p_j^i - \ln p_j^i \geq \sum_{j=1}^n p_j(w) \ln p_j^i. \]

Now, it suffices to show that for each \( j \in [n] \), the probability that \( p_j(w) \ln p_j^i \leq -\frac{n \ln 1/q}{w_i} = \frac{\ln q}{w_i} \) is at most \( mnq \); the desired result will then follow by the union bound. By Corollary C.2.5, for each \( j \) we have that

\[ \Pr \left[ p_j(w) \ln p_j^i \leq \frac{(p_j^i)^{w_i}}{q} \ln p_j^i \right] \leq mnq. \]

Additionally, we know for a fact that \( p_j(w) \ln p_j^i \geq \ln p_j^i \) (since \( p_j(w) \leq 1 \)), so in fact

\[ \Pr \left[ p_j(w) \ln p_j^i \leq \max \left( \frac{(p_j^i)^{w_i}}{q} \ln p_j^i, \ln p_j^i \right) \right] \leq mnq. \]

It remains only to show that \( \max \left( \frac{(p_j^i)^{w_i}}{q} \ln p_j^i, \ln p_j^i \right) \geq \frac{\ln q}{w_i} \). If \( p_j^i \geq q^{1/w_i} \) then this is clearly true,
since in that case $\ln p_j^i \geq \frac{\ln q}{w_i}$. Now suppose that $p_j^i < q^{1/w_i}$. Observe that $\frac{x^{w_i}}{q} \ln x$ decreases on $(0, e^{-1/w_i})$, and that (since $q \leq \frac{1}{e}$) we have $q^{1/w_i} \leq e^{-1/w_i}$. Therefore,

$$
\frac{(p_j^i)^{w_i}}{q} \ln p_j^i \leq \frac{(q^{1/w_i})^{w_i}}{q} \ln q^{1/w_i} = \frac{\ln q}{w_i}.
$$

This completes the proof of Equation 6.3, and thus of Lemma 6.4.8. 

The following lemma lower bounds the regret of Algorithm 6.3.3 as a function of $\zeta$.

**Lemma C.2.6.** Consider a run of Algorithm 6.3.3. Let $\zeta$ be such that $-\frac{\zeta}{w_i} \leq \partial_t L'(w^t) \leq \zeta$ for all $i, t$. The total regret is at most

$$
O\left(\zeta^{2(2-\alpha)/(1-\alpha)} T^{(5-\alpha)/(1-\alpha)}\right).
$$

**Proof.** We first bound $w_i^t$ for all $i, t$. From Lemma 6.4.4, we have that

$$(w_i^t)^{\alpha-1} \leq (w_i^t)^{\alpha-1} + \left(\frac{1}{\min_i w_i^t} + 1\right) \eta_t \zeta \leq (w_i^t)^{\alpha-1} + 2 \zeta.
$$

Here we use that $\frac{1}{\min_i w_i} + 1 \leq \frac{2}{\min_i w_i}$ and that $\eta_t \leq \min_i w_i$. Therefore, we have that

$$(w_i^t)^{\alpha-1} \leq (w_i^{t-1})^{\alpha-1} + 2 \zeta \leq \cdots \leq m^{1-\alpha} + 2 \zeta(t-1) \leq m^{1-\alpha} + 2 \zeta T.
$$

Thus, $w_i^t \geq (m^{1-\alpha} + 2 \zeta T)^{1/(\alpha-1)} \geq \Omega((\zeta T)^{1/(\alpha-1)})$ for all $i, t$.

We now use the standard regret bound for online mirror descent, see e.g. Orabona (2021, Theorem 6.8):

$$
\text{Regret} \leq \max_i \frac{B_R(u; w^t)}{\eta_T} + \frac{1}{2 \lambda} \sum_{t=1}^T \eta_t \|\nabla L'(w^t)\|_2^2,
$$

where $B_R(\cdot; \cdot)$ is the Bregman divergence of with respect to $R$, $u$ is the optimal (overall loss-minimizing) point, $\lambda$ is a constant such that $R$ is $\lambda$-strongly convex with respect to a norm of our
choice over $\Delta_m$, and $\|\cdot\|_*$ is the dual norm of the aforementioned norm.

Note that for any $x \in \Delta_m$, we have

$$\max_{v \in \Delta_m} B_R(v; x) = \max_{v \in \Delta_m} R(v) - R(x) - \langle \nabla R(x), v - x \rangle \leq \frac{m^{1-\alpha}}{\alpha} + (\min_i x_i)^{\alpha-1}.$$  

In the last step, we use the fact that $-\nabla R(x) = (x_1^{\alpha-1}, \ldots, x_m^{\alpha-1})$ (all of these coordinates are positive), so $-\langle \nabla R(x), v - x \rangle \leq \langle (x_1^{\alpha-1}, \ldots, x_m^{\alpha-1}), v \rangle$, and that all coordinates of $v$ are non-negative and add to 1.

Therefore, given our bound on $w'_i$, this first component of our regret bound (C.6) is at most

$$\frac{1}{\eta_T} \left( \frac{m^{1-\alpha}}{\alpha} + m^{1-\alpha} + 2\xi T \right) \leq O \left( \frac{\xi T}{\eta_T} \right) \leq O \left( \frac{\xi T}{(\xi T)^{1/(\alpha-1)}} \right) = O \left( (\xi T)^{(2-\alpha)/(1-\alpha)} \right).$$

To bound the second term, we choose to work with the $\ell_1$ norm. To show that $R$ is $\lambda$-convex it suffices to show that for all $x, y \in \Delta_m$ we have $\langle (\nabla^2 R(x)y, y \rangle \geq \lambda \|y\|^2$, where $\nabla^2 R$ is the Hessian matrix of $R$ (Shalev-Shwartz (2007, Lemma 14); see also Orabona (2021, Theorem 4.3)). Equivalently, we wish to find a $\lambda$ such that

$$(1 - \alpha) \sum_i x_i^{\alpha-2} y_i^2 \geq \lambda.$$  

Since $x_i^{\alpha-2} \geq 1$ for all $i$, the left-hand side is at least $(1 - \alpha) \sum_i y_i^2 \geq \frac{1-\alpha}{m}$, so $\lambda = \frac{1-\alpha}{m}$ suffices.

Now, given $\theta \in \mathbb{R}^m$, we have $\|\theta\|_* = \max_{x: \|x\| \leq 1} \langle \theta, x \rangle$. In the case of the $\ell_1$ primal norm, the dual norm is the largest absolute component of $\theta$. Thus, we have

$$\|\nabla L'(x')\|_* \leq \frac{\xi}{w'_i} \leq O \left( \xi (\xi T)^{1/(1-\alpha)} \right) = O \left( \xi^{(2-\alpha)/(1-\alpha)} T^{1/(1-\alpha)} \right).$$

Since $\eta_T \leq O(T^{-1/2})$, we have that the second component of our regret bound (C.6) is at most

$$O \left( T \cdot T^{-1/2} \cdot \xi^{2(2-\alpha)/(1-\alpha)} T^{2/(1-\alpha)} \right) \leq O \left( \xi^{2(2-\alpha)/(1-\alpha)} T^{5(2-\alpha)/(1-\alpha)} \right).$$
This component dominates our bound on the regret of the first component, in both $\zeta$ and $T$. This concludes the proof.

**Corollary 6.4.9.** The expected total regret of our algorithm conditional on the small gradient assumption not holding, times the probability of this event, is at most $\tilde{O}(T^{(5-\alpha)/(1-\alpha)-10})$.

**Proof.** Let $Z$ be the minimum value of $\zeta$ such that $-\frac{\zeta}{w_i} \leq \partial_i L(t(w^i)) \leq \zeta$ for all $i, t$. Note that by Lemma 6.4.8, we have that

$$\Pr[Z \geq x] \leq \sum_{i=1}^{m} \sum_{t=1}^{T} (mn^2 e^{-\frac{x}{T}} + ne^{-x}) \leq 2m^2 n^2 Te^{-\frac{x}{T}}.$$  

Let $\mu$ be the constant hidden in the big-O of Lemma C.2.6, i.e. a constant (dependent on $m, n$, and $\alpha$) such that

$$\text{Regret} \leq \mu Z^{2(2-\alpha)/(1-\alpha)T^{(5-\alpha)/(1-\alpha)}}.$$  

Let $r(Z, T)$ be the expression on the right-hand side. The small gradient assumption not holding is equivalent to $Z > 12n \ln T$, or equivalently, $r(Z, T) > r(12n \ln T, T)$. The expected regret of our algorithm conditional on the small gradient assumption not holding, times the probability of this event, is therefore at most the expected value of $r(Z, T)$ conditional on the value being greater than $r(12n \ln T, T)$, times this probability. This is equal to

$$r(12n \ln T, T) \cdot \Pr[Z > 12n \ln T] + \int_{x=r(12n \ln T, T)}^{\infty} \Pr[r(Z, T) \geq x] \, dx$$  

$$\leq \sum_{k=11}^{\infty} r((k+1)n \ln T, T) \cdot \Pr[Z \geq kn \ln T]$$  

$$\leq \sum_{k=11}^{\infty} \mu \cdot ((k+1)n \ln T)^{2(2-\alpha)/(1-\alpha)T^{(5-\alpha)/(1-\alpha)}} \cdot 2m^2 n^2 T \cdot T^{-k}$$  

$$\leq \sum_{k=11}^{\infty} \tilde{O}(T^{1+(5-\alpha)/(1-\alpha)-k}) = \tilde{O}(T^{(5-\alpha)/(1-\alpha)-10}),$$  

as desired. (The first inequality follows by matching the first term with the $k = 11$ summand and upper-bounding the integral with subsequent summands, noting that $r((k+1)n \ln T, T) \geq 1$.)
Note that \( \frac{\alpha}{\alpha-1} - 10 \leq \frac{5-1/2}{1-1/2} - 10 = -1 \). Therefore, the contribution to expected regret from the case that the small gradient assumption does not hold is \( \tilde{O}(T^{-1}) \), which is negligible. Together with Corollary 6.4.7 (which bounds regret under the small gradient assumption), this proves Theorem 6.3.4.
Appendix D: Details omitted from Chapter 8

D.1 Details omitted from Section 8.2

In Section 8.2, we claimed that a positive measure of \( n \times n \) information structures satisfy rectangle substitutes. To formalize this claim, we choose a natural measure over \( n \times n \) information structures, specified via the following probability distribution over the values of \( Y \) and \( \mathbb{P} [\sigma, \tau] \):

- Alice has signals labeled \( \sigma_0, \ldots, \sigma_{n-1} \); Bob has signals labeled \( \tau_0, \ldots, \tau_{n-1} \). Correspondingly, there are \( n^2 \) states which we identify with the pair \((i, j)\). For each \( i, j \), whenever \( \sigma = \sigma_i \), \( \tau = \tau_j \), \( Y = y(i, j) \) where \( y(i, j) \) is uniformly random in \([0, 1]\).
- The probability distribution over states \((i, j)\) is selected uniformly from the space of probability distributions over \( n^2 \) states.

Theorem D.1.1. For every \( n \), a positive measure of \( n \times n \) information structures (per the above measure) satisfy the rectangle substitutes condition.

Proof. The proof is conceptually quite simple. It suffices to exhibit an information structure in which the weak substitutes condition (i.e. Equation 8.1) holds strictly for every \( S, T \) such that \(|S|, |T| \geq 2\). It then follows that for a sufficiently small \( \delta \), every information structure in the \( \delta \)-ball around this one\(^1\) also satisfies rectangle substitutes, completing the proof.\(^2\)

The information structure \( I \) that we exhibit is as follows: choose any increasing, strictly concave function \( f : [0, 2(n-1)] \rightarrow \mathbb{R} \) (for example, \( f(x) = \sqrt{x} \)). Let \( y(i, j) = \frac{i+j}{2n} \), and let \( \mathbb{P}[(i, j)] \)

\(^1\)We can for example define the distance between information structures \( I \) and \( I' \) as \( \sum_{i,j} (y(i, j) - y'(i, j))^2 + (\mathbb{P}[(i, j)] - \mathbb{P}'[(i, j)])^2 \).

\(^2\)This uses the continuity of the terms in Equation 8.1. Note that the continuity of conditional expectations relies on the conditioning events having positive probability, as is the case in the information structure that we exhibit. Note also that we need not concern ourselves with cases in which \(|S| = 1 \) or \(|T| = 1\), since in those cases the equation is necessarily an equality.
be proportional to $e^{f(i+j)}$.

For convenience, define the substitutes slack of an information structure to be the additive margin by which the information structure satisfies weak substitutes, i.e. the right-hand side of Equation 8.1 minus the left-hand side for $S = S$ and $T = T$.

Fix a particular $S$ and $T$ such that $|S|, |T| \geq 2$. We wish to show that for sufficiently small positive values of $\epsilon$, Equation 8.1 holds strictly. We will show that the substitutes slack of $\mathcal{I} |_{S,T}$, i.e. $\mathcal{I}$ restricted to $S \times T$, is positive when $\epsilon$ is sufficiently small.

In order to prove this, we first consider the following (different) information structure for values $v, a, a', b, b', c, x, x', y, y'$ (obeying comparisons that we specify below). Each row corresponds to a possible signal value $\sigma$ for Alice, and each column a possible signal value $\tau$ for Bob.

$$Y = \begin{bmatrix} v & v + b & b' \\ v + a & v + a + b & - \\ a' & - & - \end{bmatrix} \text{ with probability proportional to } \begin{bmatrix} 1 & y & y' \\ x & cxy & 0 \\ x' & 0 & 0 \end{bmatrix}$$

In this information structure, suppose that $x' \leq x \ll 1$; $y' \leq y \ll 1$; and $1 \ll c \leq 1, 1$ (so $xy \ll cxy \leq x, y$). It can be verified (e.g. with a computer algebra system) that the substitutes slack of this information structure is $2abcxy + O(xy)$.

We will transform this information structure into $I_{S,T}$ while (approximately) preserving substitutes slack. To foreshadow the correspondence, define $i_S$ and $i'_S$ be the smallest and second smallest values of $i$ such that $\sigma_i \in S$, and define $j_T$ and $j'_T$ analogously. The rows of the information structure above will correspond to $\sigma = \sigma_{i_S}, \sigma'_{i'_S}$, and all other values of $\sigma \in S$, in that order; the columns will correspond to $\tau = \tau_{j_T}, \tau'_{j'_T}$, and all other values of $\tau \in T$, in that order.

Set $x := e^{f(i'_S+j_T)-f(i_S+j_T)}$, $y := e^{f(i_S+j'_T)-f(i_S+j_T)}$, and $c := e^{f(i_S+j_T)+f(i'_S+j'_T)-f(i_S+j'_T)-f(i'_S+j_T)}$, so that $cxy = e^{f(i'_S+j'_T)-f(i_S+j_T)}$. Note that these values satisfy the aforementioned inequalities involving $x$, $y$, and $c$. (The fact that $1 \ll c$ follows from the strict concavity of $f$.) Set $v := \frac{i_S}{2n}$, $a := \frac{i'_S-i_S}{2n}$, and $b := \frac{j_T-j}{2n}$. Set $x'$ so that $P[i > i'_S \mid i \in S, j = j_T] = \frac{x'}{1+x+x'}$ and $y'$ so that $P[j > j'_T \mid i = i_S, j \in T] = \frac{y'}{1+xy+xy'}$. We set $a' := \mathbb{E}[y(i, j_T) \mid i > i'_S, i \in S]$ and set $b'$ :=

352
We now make the following transformation to this information structure: we replace the third row with $|S| - 2$ rows, each corresponding to a different $i > i_S'$. As before, each signal will only be possible in conjunction with Bob’s first signal; the value of $Y$ for the signal corresponding to $\sigma_I$ in $I$ will be $\frac{i + j_T}{2n}$, and the probability will be $\mathbb{P}[(i, j_T)]$. Note that this simply “splits” Alice’s third signal into multiple (more informative) signals while preserving the total probability and expectation (this is due to how we picked $a', b', x'$, $y'$ above). This does not affect the substitutes slack of the information structure, because the value of Bob’s signal does not change as a result of the transformation (regardless of whether Alice’s signal is known).

We make the same transformation but this time to Bob, replacing the third column with $|T| - 2$ columns. The transformation is otherwise analogous, and the substitutes slack again does not change.

Finally, in our last transformation we make this information structure match $I$ exactly. Note that the information structures already match in the first row ($i = i_S$), and in the first column ($j = j_T$), and in the (second row, second column) entry ($(i, j) = (i_S', j_T')$). All other entries in $I$ have probabilities that are $o(c_{xy})$ (recall that $c_{xy} = e^{f(i_S' + j_T') - f(i_S + j_T)}$). As a consequence, adding these entries to the information structure that we are transforming only changes the substitutes slack by $o(c_{xy})$.

Therefore, $I$ has substitutes slack $2abc_{xy} + o(c_{xy}) \geq \frac{2}{\mu^2}e^{f(i_S' + j_T') - f(i_S + j_T)}(1 + o(1))$. This is positive for $\mu$ sufficiently small, as desired.

We complete the proof by setting $\mu$ to be such that it is sufficiently small (in the above argument) for all $S, T$ such that $|S|, |T| \geq 2$. □

### D.2 Details omitted from Section 8.3

**Proposition D.2.1.** Consider the following protocol, parametrized by $\epsilon > 0$. Alice and Bob send their initial expectations to each other, rounding to the nearest multiple of $\epsilon$. This protocol entails
communicating $O(\log 1/\epsilon)$ bits. At the end of the protocol, Alice and Bob $\epsilon^2/2$-agree and are $\epsilon^2$-accurate (with respect to $G(x) = x^2$).

**Proof.** Let $S$ be the set of possible signals of Alice at the end of the protocol which are consistent with the protocol transcript, and define $T$ likewise for Bob. Recall that we use $S$ and $T$ to denote the sets of all of Alice’s and Bob’s possible signals, respectively. We have

$$
\mathbb{E}[(\mu_{ST} - \mu_{S})^2] \leq \mathbb{E}[(\mu_{S} - \mu_{S})^2] \leq \epsilon^2,
$$

since $\mu_{ST}$ and $\mu_{S}$ are guaranteed to be within $\epsilon$ of each other by construction. Thus, Bob is $\epsilon^2$-accurate, and likewise for Alice. By the $\frac{1}{2}$-approximate triangle inequality for $G(x) = x^2$ (see Section 8.4), it follows that Alice and Bob $\epsilon^2/2$-agree. \qed

**D.3 Details omitted from Section 8.4**

**Proposition D.3.1.** Let $G$ be a differentiable convex function on the interval $[0, 1]$. For all $0 \leq a \leq b \leq 1$, we have

(i) $\frac{1}{2}(D_G(a \parallel x) + D_G(b \parallel x)) \geq JB_G(a, b)$ for every $x \in [0, 1]$.

(ii) $JB_G$ satisfies the reverse triangle inequality: for every $x \in [a, b]$, we have $JB_G(a, x) + JB_G(x, b) \leq JB_G(a, b)$.

(iii) For all $a \leq a' \leq b' \leq b$, we have $JB_G(a', b') \leq JB_G(a, b)$.

(iv) For a random variable $X$ supported on $[a, b]$, we have

$$
\mathbb{E}[D_G(X \parallel \mathbb{E}[X])] = \mathbb{E}[G(X)] - G(\mathbb{E}[X]) \leq 2JB_G(a, b).
$$

**Proof.** Fact (i) follows from Proposition 2.1.8. Regarding Fact (ii), without loss of generality assume that $x \leq \frac{a+b}{2}$ and that $G(x) = G\left(\frac{a+b}{2}\right)$ (uniformly adding a constant to the derivative of $G$ does not change any Jensen-Bregman divergence, hence the second assumption). Then
\[ G \left( \frac{a+b}{2} \right) \geq G(x) \], so \( J_B G(a, x) \leq \frac{G(a) - G(x)}{2} \). Since \( \frac{b+a}{2} \geq \frac{a+b}{2} \), we also have that \( G \left( \frac{b+a}{2} \right) \geq G(x) \), so \( J_B G(b, x) \leq \frac{G(b) - G(x)}{2} \). Thus, we have

\[ J_B G(a, x) + J_B G(b, x) \leq \frac{G(a) + G(b)}{2} - G(x) = \frac{G(a) + G(b)}{2} - G \left( \frac{a+b}{2} \right) = J_B G(a, b). \]

Fact (iii) follows from Fact (ii): we have

\[ J_B G(a, b) = J_B G(a, a') + J_B G(a', b') + J_B G(b', b) \geq J_B G(a', b'). \]

Regarding the equality in Fact (iv), we have

\[ E \left[ D_G(X \mid E[X]) \right] = E \left[ G(X) - G(E[X]) \right] - (X - E[X]) G'(E[X]) \]

\[ = E \left[ G(X) - G(E[X]) \right] = E \left[ G(X) - G(E[X]) \right], \]

where the first step follows from the fact that \( E \left[ (X - E[X]) G'(E[X]) \right] = G'(E[X]) E \left[ X - E[X] \right] \), and \( E \left[ X - E[X] \right] = 0 \).

Regarding the inequality in Fact (iv), without loss of generality assume that \( E[X] \leq \frac{a+b}{2} \). By convexity we have that

\[ G \left( \frac{a+b}{2} \right) \leq \frac{b-a}{b - E[X]} G(E[X]) + \frac{a+b - E[X]}{b - E[X]} G(b), \]

so

\[ J_B G(a, b) = G(a) + G(b) - 2G \left( \frac{a+b}{2} \right) \]

\[ \geq G(a) + G(b) - \frac{b-a}{b - E[X]} G(E[X]) - \frac{a+b - 2E[X]}{b - E[X]} G(b) \]

\[ = G(a) + \frac{E[X] - a}{b - E[X]} G(b) - \frac{b-a}{b - E[X]} G(E[X]) \]

\[ = \frac{b-a}{b - E[X]} \left( \frac{b - E[X]}{b-a} G(a) + \frac{E[X] - a}{b-a} G(b) - G(E[X]) \right). \]
\[ \geq \frac{b - \mathbb{E}[X]}{b - a} G(a) + \frac{\mathbb{E}[X] - a}{b - a} G(b) - G(\mathbb{E}[X]) \geq \mathbb{E}[G(X)] - G(\mathbb{E}[X]). \]

In the last step we use the fact that for a convex function \( f \) and a random variable \( X \) defined on an interval \([a, b]\) with mean \( \mu \), the maximum possible value of \( \mathbb{E}[f(X)] \) is attained if \( X \) is either \( a \) or \( b \) with the appropriate probabilities. \( \square \)

**Theorem 8.4.8.** The discretized protocol with respect to \( G \) with parameter \( \epsilon \) is an \( \epsilon \)-agreement protocol with respect to \( G \) that requires \( O\left(\frac{M(M+\epsilon)}{\epsilon^{2}}\right) \) bits of communication.

**Proof.** Suppose that Alice and Bob do not \( \epsilon \)-agree at time step \( t \), and without loss of generality assume that the next turn (number \( t + 1 \)) is Alice’s. We begin by observing that, by Proposition D.3.1 (i), we have

\[ \mathbb{E}\left[ D_G(\mu_{\sigma T_i} \parallel \mu_{S_i T_i}) + D_G(\mu_{S_i T_i} \parallel \mu_{S_i T_i}) \right] \geq 2 \mathbb{E}\left[J_B G(\mu_{\sigma T_i}, \mu_{S_i T_i})\right] > 2 \epsilon. \]

Therefore, either \( \mathbb{E}\left[ D_G(\mu_{\sigma T_i} \parallel \mu_{S_i T_i}) \right] \geq \frac{2 \epsilon}{3} \) or \( \mathbb{E}\left[ D_G(\mu_{S_i T_i} \parallel \mu_{S_i T_i}) \right] \geq \frac{4 \epsilon}{3} \).

**Case 1:** \( \mathbb{E}\left[ D_G(\mu_{\sigma T_i} \parallel \mu_{S_i T_i}) \right] \geq \frac{2 \epsilon}{3} \). Let us use “hi,” “lo,” and “md” to denote the events that Alice says “high,” Alice says “low,” and Alice says “medium,” respectively. We have

\[ \frac{2 \epsilon}{3} \leq \mathbb{E}\left[ D_G(\mu_{\sigma T_i} \parallel \mu_{S_i T_i}) \right] = \mathbb{E}\left[ \mathbb{E}\left[ D_G(\mu_{\sigma T_i} \parallel \mu_{S_i T_i}) | S_i, T_i \right] \right] \]

\[ = \mathbb{E}\left[ \mathbb{E}\left[ D_G(\mu_{\sigma T_i} \parallel \mu_{S_i T_i}) \cdot 1_{hi or lo} | S_i, T_i \right] \right] + \mathbb{E}\left[ \mathbb{E}\left[ D_G(\mu_{\sigma T_i} \parallel \mu_{S_i T_i}) \cdot 1_{md} | S_i, T_i \right] \right] \]

\[ \leq \mathbb{E}\left[ \mathbb{E}\left[ D_G(\mu_{\sigma T_i} \parallel \mu_{S_i T_i}) \cdot 1_{hi or lo} | S_i, T_i \right] \right] + \frac{\epsilon}{2}, \]

where \( 1_{S_i, T_i} \) is short for \( 1_{\sigma \in S_i, \tau \in T_i} \), a notation we use throughout the proof. We thus have

\[ \mathbb{E}\left[ \mathbb{E}\left[ D_G(\mu_{\sigma T_i} \parallel \mu_{S_i T_i}) \cdot 1_{hi} | S_i, T_i \right] \right] + \mathbb{E}\left[ \mathbb{E}\left[ D_G(\mu_{\sigma T_i} \parallel \mu_{S_i T_i}) \cdot 1_{lo} | S_i, T_i \right] \right] \geq \frac{\epsilon}{6}. \quad (D.1) \]

We now make use of the following lemma.
Lemma D.3.2. Suppose that turn $t+1$ is Alice’s. Let “hi” denote the event that Alice says “high.” Let $\alpha := \mathbb{E} \left[ D_G(\mu_{\sigma T_i} \| \mu_{S_i T_i}) \cdot 1_{hi} \mid S_i, T_i \right]$. Then

$$\mathbb{E} \left[ D_G(\mu_{S_{t+1} T_{t+1}} \| \mu_{S_i T_i}) \cdot 1_{hi} \mid S_i, T_i \right] \geq \frac{\alpha \varepsilon}{8M + 2\varepsilon}.$$ 

The analogous statement is true if Alice says “low,” and likewise if it is instead Bob’s turn.

We assume Lemma D.3.2 and return to prove it afterward. This lemma translates Equation D.1 into a statement about how much Charlie learns. Specifically, we have that

$$\mathbb{E} \left[ D_G(\mu_{S_{t+1} T_{t+1}} \| \mu_{S_i T_i} + 1_T \| \mu_{S_i T_i}) \cdot 1_{hi} \mid S_i, T_i \right] \geq \mathbb{E} \left[ D_G(\mu_{S_{t+1} T_{t+1}} \| \mu_{S_i T_i}) \cdot 1_{lo} \mid S_i, T_i \right] \geq \frac{\varepsilon^2}{6(8M + 2\varepsilon)}.$$ 

**Case 2:** $\mathbb{E} \left[ D_G(\mu_{S_{t+1}} \| \mu_{S_i T_i}) \right] \geq \frac{4\varepsilon}{3}$. Using the Pythagorean theorem to write the same Bregman divergence in two ways, we have that

$$\mathbb{E} \left[ D_G(\mu_{s_{t+1}} \| \mu_{S_i T_i}) \right] + \mathbb{E} \left[ D_G(\mu_{s_{t+1}} \| \mu_{S_i T_i}) \right] = \mathbb{E} \left[ D_G(\mu_{s_{t+1}} \| \mu_{S_i T_i}) \right] \geq \frac{4\varepsilon}{3}.$$

This means that one of the two summands on the left-hand side is at least $\frac{2\varepsilon}{3}$.

**Case 2a:** $\mathbb{E} \left[ D_G(\mu_{s_{t+1}} \| \mu_{S_i T_i}) \right] \geq \frac{2\varepsilon}{3}$. In that case we have that

$$\mathbb{E} \left[ D_G(\mu_{s_{t+2}} \| \mu_{S_i T_i}) \right] \geq \frac{\varepsilon^2}{6(8M + 2\varepsilon)}$$

by the same logic as in Case 1.
**Case 2b:** \( \mathbb{E} \left[ D_G(\mu_{S_{i+1}T_{i+1}} \| \mu_{S_iT_i}) \right] \geq \frac{2\epsilon}{3} \geq \frac{\epsilon^2}{12\epsilon} \geq \frac{\epsilon^2}{6(8M + 2\epsilon)}. \)

In each of our cases, we have that

\[
\mathbb{E} \left[ D_G(Y \| \mu_{S_iT_i}) - D_G(Y \| \mu_{S_{i+2}T_{i+2}}) \right] = \mathbb{E} \left[ D_G(\mu_{S_{i+2}T_{i+2}} \| \mu_{S_iT_i}) \right] \\
= \mathbb{E} \left[ D_G(\mu_{S_{i+2}T_{i+2}} \| \mu_{S_{i+1}T_{i+1}}) \right] + \mathbb{E} \left[ D_G(\mu_{S_{i+1}T_{i+1}} \| \mu_{S_iT_i}) \right] \geq \frac{\epsilon^2}{6(8M + 2\epsilon)}.
\]

Therefore, the total number of steps until agreement is first reached cannot be more than

\[
2 \cdot \frac{M}{\epsilon^2} = \frac{24M(4M + \epsilon)}{\epsilon^2}.
\]

This completes the proof. \( \square \)

We now prove Lemma D.3.2.

**Proof of Lemma D.3.2.** We will restrict our probability space to outcomes where Charlie knows \( S_i, T_i \) at time \( t \) (and thus omit \( | S_i, T_i \) from here on). For convenience, we will let \( A := \mu_{\sigmaT_i} \) be Alice’s expectation (a random variable) and \( c := \mu_{S_iT_i} \) be Charlie’s expectation (which is a particular number in \([0, 1]\)). We will let \( \epsilon' := \frac{\epsilon}{2} \), so that if Alice says “high” then Charlie knows that \( A > c \) and that \( D_G(A \| c) \geq \epsilon' \).

Let \( D(x) := D_G(x \| c) = G(x) - G(c) - G'(c)(x - c) \), and let \( \hat{a}_h := \mathbb{E} [A \mid hi] \). Note that if Alice says “high” then \( \mu_{S_{i+1}T_{i+1}} = \hat{a}_h \). In our new notation, we may write \( \alpha = \mathbb{E} [D(A) \mid hi] \cdot \mathbb{P}[hi] \), and we wish to show that \( D(\hat{a}_h) \cdot \mathbb{P}[hi] \geq \frac{\alpha \epsilon'}{2(M + \epsilon')}. \) Put otherwise, our goal is to show that

\[
\frac{D(\hat{a}_h)}{\mathbb{E} [D(A) \mid hi]} \geq \frac{\epsilon'}{2(M + \epsilon')}. 
\]

For convenience we will let \( B \) denote the quantity on the left-hand side.

Let \( a_{h\text{min}} \) be the number larger than \( c \) such that \( D(a) = \epsilon' \), so that \( A \geq a_{h\text{min}} \) whenever
Alice says “high.”³ Observe that since $D$ is convex (Bregman divergences are convex in their first argument), for a fixed value of $\hat{a}_h$, the value of $\mathbb{E} \left[ D(A) \mid \text{hi} \right]$ is maximized when $A$ is either $a_{\text{hmin}}$ or $1$ (with probabilities $\frac{1-\hat{a}_h}{1-a_{\text{hmin}}}$ and $\frac{\hat{a}_h-a_{\text{hmin}}}{1-a_{\text{hmin}}}$, respectively). Therefore we have

$$B = \frac{D(\hat{a}_h)}{\mathbb{E} \left[ D(A) \mid \text{hi} \right]} \geq \frac{D(\hat{a}_h)(1-a_{\text{hmin}})}{(1-\hat{a}_h)e' + (\hat{a}_h-a_{\text{hmin}})D(1)}.$$  
(D.2)

**Case 1:** $(1-\hat{a}_h)e' \geq (\hat{a}_h-a_{\text{hmin}})D(1)$. In that case we have

$$B \geq \frac{D(\hat{a}_h)(1-a_{\text{hmin}})}{2(1-\hat{a}_h)e'} \geq \frac{e'(1-a_{\text{hmin}})}{2(1-\hat{a}_h)e'} \geq \frac{1}{2} \geq \frac{e'}{2(M+e')}.$$  

**Case 2:** $(1-\hat{a}_h)e' \leq (\hat{a}_h-a_{\text{hmin}})D(1)$. In that case we have

$$B \geq \frac{D(\hat{a}_h)(1-a_{\text{hmin}})}{2(\hat{a}_h-a_{\text{hmin}})D(1)}.$$  
(D.3)

**Case 2a:** $D(1) \leq \frac{1-c}{\hat{a}_h-c}(M+e')$. Then we have

$$B \geq \frac{D(\hat{a}_h)(1-a_{\text{hmin}})}{2(\hat{a}_h-a_{\text{hmin}})} \cdot \frac{1-c}{\hat{a}_h-c}(M+e') \geq \frac{e'}{2(M+e')} \cdot \frac{(1-a_{\text{hmin}})(\hat{a}_h-c)}{(\hat{a}_h-a_{\text{hmin}})(1-c)}.$$  

(In the last step we again use that $D(\hat{a}_h) \geq e$.) Now, it is easy to verify that the second fraction is at least 1 (this comes down to the fact that $a_{\text{hmin}} \geq c$), so we indeed have that $B \geq \frac{e'}{2(M+e')}$. 

**Case 2b:** $D(1) \geq \frac{1-c}{\hat{a}_h-c}(M+e')$. We claim that for all $x \geq c$, we have that

$$D(x) \geq \frac{x-c}{1-c}D(1) - M.$$  
(D.4)

To see this, suppose for contradiction that for some $x$ we have $D(x) < \frac{x-c}{1-c}D(1) - M$. Then

$$G(x) - G(c) - G'(c)(x-c) < \frac{x-c}{1-c} (G(1) - G(c) - G'(c)(1-c)) - M$$

³If $D(a) < e'$ for all $a > c$ then Alice never says “high” and the lemma statement is trivial.
\[(1 - c)G(x) - (1 - c)G(c) < (x - c)G(1) - (x - c)G(c) - (1 - c)M\]
\[G(x) + M < \frac{(1 - x)G(c) + (x - c)G(1)}{1 - c}.
\]

On the other hand, we have that both \(G(c)\) and \(G(1)\) are less than or equal to \(G(x) + M\), by definition of \(M\). This means that
\[G(1), G(c) < \frac{(1 - x)G(c) + (x - c)G(1)}{1 - c}\]
but this implies that \(G(1) < G(c)\) and that \(G(c) < G(1)\), a contradiction.

Plugging in \(x = \hat{a}_h\) into Equation D.4, we find that
\[D(\hat{a}_h) \geq \frac{\hat{a}_h - c}{1 - c} D(1) - M.
\]

Plugging this bound into Equation D.3, we get that
\[B \geq \frac{\hat{a}_h - c}{1 - c} D(1) - M \geq \frac{1 - a_{\text{min}}}{2(\hat{a}_h - a_{\text{min}})} \cdot \frac{\hat{a}_h - c}{1 - c} \left(1 - \frac{M}{M + \epsilon'}\right) \geq \frac{\epsilon'}{2(M + \epsilon')}\]
where in the second-to-last step we use that \(D(1) \geq \frac{1 - c}{\hat{a}_h - c} (M + \epsilon')\) and in the last step we again use the fact that \(\frac{1 - a_{\text{min}}}{\hat{a}_h - a_{\text{min}}}(1 - c) \geq 1\).

**Lemma 8.4.16.** Let \(G\) be a differentiable convex function on \([0, 1]\) and \(c \in (0, 1)\) be such that \(JB_G\) satisfies the \(c\)-approximate triangle inequality. Let \(I = (\Omega, \mathcal{F}, \sigma, \tau, Y)\) be an information structure that satisfies rectangle substitutes with respect to \(G\). Let \(\epsilon = \mathbb{E} [JB_G(\mu_\sigma \parallel \mu_\tau)]\). Then for any \(\beta > 0\), we have
\[\mathbb{E} [D_G(\mu_\sigma \parallel \mu_\tau)] \leq \frac{8}{c^2}\beta + 16\widetilde{G}^* \left(\left(\frac{\epsilon}{\beta}\right)^{1/(1 - \log_2 c)}\right).
\]

**Proof.** We will partition \([0, 1]\) into a number \(N\) of small intervals \(I_1 = [x_0 = 0, x_1), I_2 = [x_1, x_2),\)
\( I_3 = [x_2, x_3], \ldots, I_N = [x_{N-1}, x_N = 1] \) with certain desirable properties (which we will describe below). For \( k \in [N], \) we will let \( S^{(k)} := \{ \sigma \in S : \mu_\sigma \in I_k \} \). For a given \( \sigma \in S \), we will let \( k(\sigma) \) be the \( k \) such that \( \sigma \in S^{(k)} \).

Our goal is to upper bound the expectation of \( D_G(\mu_{\sigma \tau} \parallel \mu_\tau) \). In pursuit of this goal, we observe that by Proposition 2.3.12 we have

\[
\mathbb{E} [D_G(\mu_{\sigma \tau} \parallel \mu_\tau)] = \mathbb{E} [D_G(\mu_{\sigma \tau} \parallel \mu_{S^{(k(\sigma))} \tau})] + \mathbb{E} [D_G(\mu_{S^{(k(\sigma))} \tau} \parallel \mu_\tau)]. \tag{D.5}
\]

Now, for any \( k \), by applying Equation 8.7 to \( S = S^{(k)} \) and \( T = T \), we know that

\[
\mathbb{E} [D_G(\mu_\sigma \parallel \mu_{S^{(k)}}) \mid S^{(k)}] \geq \mathbb{E} [D_G(\mu_{\sigma \tau} \parallel \mu_{S^{(k(\sigma))} \tau}) \mid S^{(k)}].
\]

(Here, “\( \mid S^{(k)} \)” is short for “\( \mid \sigma \in S^{(k)} \).”) This is our only use of the rectangle substitutes assumption. Now, taking the expectation over \( k \) (i.e. choosing each \( k \) with probability equal to \( \mathbb{P}[\sigma \in S^{(k)}] \)), we have that

\[
\mathbb{E} [D_G(\mu_\sigma \parallel \mu_{S^{(k(\sigma))}})] \geq \mathbb{E} [D_G(\mu_{\sigma \tau} \parallel \mu_{S^{(k(\sigma))} \tau})].
\]

Together with Equation D.5, this tells us that

\[
\mathbb{E} [D_G(\mu_{\sigma \tau} \parallel \mu_\tau)] \leq \mathbb{E} [D_G(\mu_\sigma \parallel \mu_{S^{(k(\sigma))}})] + \mathbb{E} [D_G(\mu_{S^{(k(\sigma))} \tau} \parallel \mu_\tau)]. \tag{D.6}
\]

Our goal will be to bound the two summands in Equation D.6. We will specify the boundaries of the intervals \( I_1, \ldots, I_N \) with this goal in mind.

On an intuitive level, we are hoping for two things to be true:

- In order for the first summand to be small, we want \( \mu_\sigma \) and \( \mu_{S^{(k(\sigma))}} \) to be similar in value.
  
In other words, we want each interval is “short” (for a notion of shortness with respect to \( G \).
that we are about to discuss).

- In order for the second summand to be small, we want $\mu_{S(k(\sigma))\tau}$ and $\mu_\tau$ to be similar in value. In other words, the estimate of a third party who knows $\tau$ shouldn’t change much upon learning $k(\sigma)$. One way to ensure this is by creating the intervals in a way that makes the third party very confident about the value of $k(\sigma)$ before learning it. Intuitively this should be true because Alice and Bob approximately agree, so Alice’s estimate is likely to be close to Bob’s. However, we must be careful to strategically choose the boundaries of our intervals $x_1, \ldots, x_{N-1}$ so that Alice’s and Bob’s estimates are unlikely to be on opposite sides of a boundary.\(^4\)

What, formally, do we need for the first summand to be small? For any $k$, we have $\mu_{S(k(\sigma))} = \mathbb{E} [\mu_\sigma | \sigma \in S^{(k)}]$. We can apply Proposition D.3.1 (iv) to the random variable $X = \mu_\sigma$ on the probability subspace given by $\sigma \in S^{(k)}$. Since $X$ takes on values in $I_k$, we have that

$$
\mathbb{E} \left[ D_G(\mu_\sigma \| \mu_{S^{(k)}}) | S^{(k)} \right] \leq 2JB_G(I_k),
$$

(D.7)

where $JB_G(I_k)$ is shorthand for the Jensen-Bregman divergence between the endpoints of $I_k$. Therefore, if $JB_G(I_k)$ is small for all $k$, then the first summand (which is an expected value of $\mathbb{E} \left[ D_G(\mu_\sigma \| \mu_{S^{(k)}}) | S^{(k)} \right]$ over $k \in [N]$) is also small.

What about the second summand? As per the intuition above, we wish to choose our boundary points $x_1, \ldots, x_{N-1}$ so that Alice’s and Bob’s estimates are unlikely to be on opposite sides of any boundary. Let $\mu_- = \min(\mu_\sigma, \mu_\tau)$ be the smaller of the two estimates and $\mu_+ = \max(\mu_\sigma, \mu_\tau)$ be the larger one. We say that $\mu_-, \mu_+$ thwart a point $x \in (0, 1)$ if $\mu_- \leq x \leq \mu_+$ and $\mu_- \neq \mu_+$. We define the thwart density of $x$ to be

$$
\rho(x) := \mathbb{P}[\mu_-, \mu_+ \text{ thwart } x].
$$

\(^4\)This limits how many intervals we can reasonably use, which is why we cannot make our intervals arbitrarily short to satisfy the first of our two criteria.
Roughly speaking, we will choose $x_1, \ldots, x_{N-1}$ such that $\rho(x_k)$ is small on average.

We will approach this problem by first creating intervals to satisfy the first criterion (short intervals), without regard to the second, and then modifying them to satisfy the second without compromising the first. Formally, we choose our intervals according to the following algorithm.

**ALGORITHM D.3.3:** Partitioning $[0, 1]$ into intervals $I_1, \ldots, I_N$

1. Choose points $0 < x'_1 < x'_2 < \cdots < x'_{N-2} < 1$ such that the $N - 1$ intervals thus created all have Jensen-Bregman divergence between $\beta$ and $\frac{2\beta}{c}$, inclusive, where $\beta$ and $c$ are as in the statement of Lemma 8.4.16. ($N$ is not pre-determined; it is defined as one more than the number of intervals created.) (See footnote for why this is possible.\(^5\))
2. Let $x_0' := 0, x'_{N-1} := 1$ for convenience. Define $I'_k := [x'_{k-1}, x'_k]$. For $k \in [N - 1]$, let $\alpha_k := \inf_{x \in I'_k} \rho(x)$. Let $x_k \in I'_k$ be such\(^6\) that $\rho(x_k) \leq 2\alpha_k$.
3. Return the intervals $I_1 = [0, x_1), I_2 = [x_1, x_2), \ldots, I_N = [x_{N-1}, 1]$.

We begin by observing that for any $k \in [N]$, we have

$$JB_G(I_k) = JB_G(x_{k-1}, x_k) \leq JB_G(x'_{k-2}, x'_k) \leq \frac{1}{c}(JB_G(x'_{k-2}, x'_{k-1}) + JB_G(x'_{k-1}, x'_k)) \leq \frac{4\beta}{c^2}$$

where for convenience we define $x'_{-1} := 0, x'_N := 1$. Therefore, by Equation D.7, we have

$$\mathbb{E} \left[ D_G(\mu_\sigma \| \mu_{S(k(\sigma))}) \right] \leq \frac{8\beta}{c^2}. \quad \text{(D.8)}$$

It remains to bound the second summand of Equation D.6, $\mathbb{E} \left[ D_G(\mu_{S(k(\sigma)) \tau} \| \mu_\tau) \right]$, which is the bulk of the proof. We proceed in two steps:

\(^5\)Define $x'_1$ so that $JB_G(0, x'_1) = \frac{2\beta}{c}$ (this is possible because $JB_G$ is continuous in its arguments). Define $x'_2$ so that $JB_G(x'_1, x'_2) = \frac{2\beta}{c}$. Keep going until an endpoint $x'_{N-3}$ is defined such that adding $x'_{N-2}$ as before would leave an interval $(x'_{N-2}, 1)$ with Jensen-Bregman divergence less than $\frac{2\beta}{c}$. Now, instead of defining $x'_{N-2}$ in this way, define it so that $JB_G(x'_{N-3}, x'_{N-2}) = JB_G(x'_{N-2}, 1)$. Since $JB_G(x'_{N-3}, 1) \geq \frac{2\beta}{c}$, the $c$-approximate triangle inequality that we have by assumption tells us that $JB_G(x'_{N-3}, x'_{N-2}) = JB_G(x'_{N-2}, 1) \geq \beta$.

\(^6\)If the infimum is achieved (e.g. if the space of signals to Alice and Bob is finite), then we can set $x_k := \arg \min_x \rho(x)$. Our algorithm works in more generality, at the expense of a factor of $2$ in our final bound. Note that by replacing $2$ with a smaller constant can arbitrarily reduce this factor.
(1) (Lemma D.3.4) We show that \( \sum_{k=1}^{N} \alpha_k \) is small. This means that Alice’s and Bob’s estimates are unlikely to lie on opposite sides of some boundary point \( x_k \). As a consequence, Bob is highly likely to know \( k(\sigma) \) with a lot of confidence.

(2) (Lemma D.3.7) We bound the second summand as a function of \( \sum_{k=1}^{N} \alpha_k \). The intuition is that if \( \sum_{k} \alpha_k \) is small, then Bob is highly likely to know \( k(\sigma) \) with a lot of confidence, which means that he does not learn too much from learning \( k(\sigma) \).

We begin with the first step; recall our notation \( \mu^- := \min(\mu_{\sigma}, \mu_\tau) \) and \( \mu^+ := \max(\mu_{\sigma}, \mu_\tau) \).

**Lemma D.3.4.**
\[
2 \sum_{k=1}^{N} \alpha_k \leq 4 \left( \frac{\varepsilon}{\beta c} \right)^{1/(1-\log_2 c)}.
\]

**Proof.** We use the following claim, whose proof we provide afterward.

**Claim D.3.5.** Let \( I = [x^-, x^+] \) be any sub-interval of \( [0, 1] \) and let \( \alpha = \inf_{x \in I} \rho(x) \). Then there is an increasing sequence of points \( z_0 := x^-, z_1, z_2, \ldots, z_{L-1}, z_L := x^+ \), such that for every \( \ell \in [L] \), \( \Pr[\mu^- \leq z_{\ell-1}, \mu^+ \geq z_\ell] \geq \frac{a}{2} \), and where
\[
L \leq 2 \sum_{\ell \in [L]} \Pr[\mu^- \leq z_{\ell-1} < \mu^+ \leq z_\ell].
\]

We apply Claim D.3.5 to the intervals \( I'_1, \ldots, I'_{N-1} \), with \( \alpha = \alpha_k \). Let \( z_{k,0}, \ldots, z_{k,L_k} \) be the points whose existence the claim proves, and let \( r_k := \sum_{\ell \in [L_k]} \Pr[\mu^- \leq z_{k,\ell-1} < \mu^+ \leq z_{k,\ell}] \), so that \( L_k \leq \frac{2}{\alpha_k} r_k \). Observe that \( \sum_{k} r_k \leq 1 \), because the intervals \((z_{k,\ell-1}, z_{k,\ell})\) are disjoint for all \( k, \ell \). We make the following claim (we provide the proof afterward).

**Claim D.3.6.**
\[
\sum_{k \in [N-1]} r_k \left( \frac{\alpha_k}{2r_k} \right)^{1-\log_2 c} \leq \frac{\varepsilon}{\beta c}. \tag{D.9}
\]

We may rewrite Equation D.9 as
\[
\left( \sum_{k \in [N-1]} r_k \left( \frac{\alpha_k}{2r_k} \right)^{1-\log_2 c} \right)^{1/(1-\log_2 c)} \leq \left( \frac{\varepsilon}{\beta c} \right)^{1/(1-\log_2 c)}.
\]
Recall that $\sum r_k \leq 1$. Scaling the $r_k$'s to add to 1 decreases the left-hand side above, so we may assume that $\sum r_k = 1$. Note that $x^{1-\log_2 c}$ is convex. Thus, by using a weighted Jensen inequality on the left-hand side with weights $r_k$, we find that

$$\frac{1}{2} \sum_k \alpha_k = \sum_k r_k \cdot \frac{\alpha_k}{2r_k} \leq \left( \sum_{k \in [N-1]} r_k \left( \frac{\alpha_k}{2r_k} \right)^{1-\log_2 c} \right)^{1/(1-\log_2 c)} \leq \left( \frac{e}{\beta c} \right)^{1/(1-\log_2 c)}.$$

This completes the proof of Lemma D.3.4.

**Proof of Claim D.3.5.** Let $z_1 = \inf\{z : \mathbb{P}[\mu_0 \leq z_0 < \mu_+ \leq z] \geq \frac{\alpha}{2}\}$, or $x^+$ if this number does not exist or is larger than $x^+$. Note that $\mathbb{P}[\mu_0 \leq z_0 < \mu_] \geq \alpha$, as we have $\rho(z_0) = \mathbb{P}[\mu_0 \leq z_0 < \mu_] + \mathbb{P}[\mu_0 < z_0 = \mu_] \geq \alpha$, so if the first term were less than $\alpha$ we would have some $z' > z_0$ with $\rho(z') < \alpha$. On the other hand, $\mathbb{P}[\mu_0 \leq z_0 < \mu_+ < z_1] \leq \frac{\alpha}{2}$, since

$$\mathbb{P}[\mu_0 \leq z_0 < \mu_+ < z_1] = \lim_{z \to z_1 \text{ from below}} \mathbb{P}[\mu_0 \leq z_0 < \mu_+ \leq z]$$

and if the right-hand side were more than $\frac{\alpha}{2}$ then that would contradict the definition of $z_1$ as an infimum. Therefore, $\mathbb{P}[\mu_0 \leq z_0, \mu_+ \geq z_1] \geq \frac{\alpha}{2}$.

If $z_1 = x^+$, we are done. Otherwise, let $z_2 = \inf\{z : \mathbb{P}[\mu_0 \leq z_1 < \mu_+ \leq z] \geq \frac{\alpha}{2}\}$. Then $\mathbb{P}[\mu_0 \leq z_1, \mu_+ \geq z_2] \geq \frac{\alpha}{2}$. Define $z_3$ analogously, and so forth.

All that remains to show is the upper bound on $L$. This is where we use the fact that (by construction) $\mathbb{P}[\mu_0 \leq z_{\ell-1} < \mu_+ \leq z_\ell] \geq \frac{\alpha}{2}$. Summing over all $\ell$, we have

$$\sum_{\ell \in [L]} \mathbb{P}[\mu_0 \leq z_{\ell-1} < \mu_+ \leq z_\ell] \geq \frac{\alpha}{2} L,$$

which (after rearranging) completes the proof.

**Proof of Claim D.3.6.** First note that by construction, $\text{JB}_G(I'_k) \geq \beta$ for all $k$. By repeated use of
the $c$-approximate triangle inequality,\footnote{We sub-divide $I'_k$ into $[z_{k, 0}, z_{k, L_k/2}]$ and $[z_{k, L_k/2}, z_{k, L_k}]$, then subdivide each of these, and so on.} we find that

$$\sum_{\ell \in [L_k]} JB_G(z_{k, \ell - 1}, z_{k, \ell}) \geq c \frac{\log_2 L_k}{\alpha_k} JB_G(I'_k) \geq c \frac{1 + \log_2 \frac{2r_k}{\alpha_k}}{\alpha_k} \beta = c \left( \frac{2r_k}{\alpha_k} \right)^{\log_2 c} \beta.$$ 

On the other hand, we have

$$\epsilon \geq \mathbb{E} [JB_G(\mu_\sigma, \mu_\tau)] = \sum_{\sigma, \tau} P[\sigma, \tau] JB_G(\mu_\sigma, \mu_\tau) \geq \sum_{\sigma, \tau} \sum_{k, \ell: \mu \leq z_{k, \ell - 1}} P[\sigma, \tau] JB_G(z_{k, \ell - 1}, z_{k, \ell})$$

$$= \sum_{k, \ell} P[\mu_\sigma \leq z_{k, \ell - 1}, \mu_\tau \geq z_{k, \ell}] JB_G(z_{k, \ell - 1}, z_{k, \ell}) \geq \sum_{k, \ell} \frac{\alpha_k}{2} JB_G(z_{k, \ell - 1}, z_{k, \ell}).$$

Here, the third step follows by the reverse triangle inequality (Fact (ii) of Proposition D.3.1) and the fourth step follows by rearranging the order of summation.\footnote{The case that the space of signals is infinite is identical except that the summation is replaced by an integral over the probability space.} Combining the last two facts gives us that

$$\epsilon \geq \sum_k \frac{\alpha_k}{2} \cdot c \left( \frac{2r_k}{\alpha_k} \right)^{\log_2 c} \beta = \sum_k r_k \left( \frac{2r_k}{\alpha_k} \right)^{\log_2 c - 1} \beta c,$$

which rearranges to the desired identity. \hfill \Box

We are now ready to bound the second summand, i.e. $\mathbb{E} [D_G(\mu_\sigma, \mu_\tau)]$, where $k(\sigma)$ is the $k$ such that Alice’s estimate $\mu_\sigma$ lies in $I_k$. For convenience we will define $k(\tau)$ for Bob by analogy as the $k$ such that $\mu_\tau$ lies in $I_k$. By Lemma D.3.4 and the preceding discussion, we know that

$$P[k(\sigma) \neq k(\tau)] \leq 4 \left( \frac{\epsilon}{\beta c} \right)^{1/(1 - \log_2 c)}.$$ 

**Lemma D.3.7.** Let $Q = P[k(\sigma) \neq k(\tau)]$. Then

$$\mathbb{E} [D_G(\mu_{S(k(\sigma))}, \mu_\tau)] \leq 2 \tilde{G}^*(Q).$$

The key idea is that because $k(\sigma) = k(\tau)$ with probability near 1, learning $k(\sigma)$ is unlikely to
make Bob update his estimate much.

**Proof.** Consider any signal $\hat{\tau} \in \mathcal{T}$ and let $p(\hat{\tau}) = \mathbb{P}[\tau = \hat{\tau}]$. We have\footnote{This proof takes sums over $\hat{\tau} \in \mathcal{T}$ and thus implicitly assumes that $\mathcal{T}$ is finite, but the proof extends to infinite $\mathcal{T}$, with sums over $\tau$ replaced by integrals with respect to the probability measure over $\mathcal{T}$.}

$$
\mathbb{E} \left[ D_G(\mu_{S(k(\sigma))\hat{\tau}} \| \mu_{\tau}) \right] = \sum_{\hat{\tau} \in \mathcal{T}} p(\hat{\tau}) \mathbb{E} \left[ D_G(\mu_{S(k(\sigma))\hat{\tau}} \| \mu_{\hat{\tau}}) \mid \tau = \hat{\tau} \right].
$$

Note that $\mu_{\hat{\tau}} = \mathbb{E} \left[ \mu_{S(k(\sigma))\hat{\tau}} \mid \tau = \hat{\tau} \right]$, so by Proposition D.3.1 we have that

$$
\mathbb{E} \left[ D_G(\mu_{S(k(\sigma))\hat{\tau}} \| \mu_{\tau}) \right] = \sum_{\hat{\tau} \in \mathcal{T}} p(\hat{\tau}) \left( \mathbb{E} \left[ G(\mu_{S(k(\sigma))\hat{\tau}}) \mid \tau = \hat{\tau} \right] - G(\mu_{\hat{\tau}}) \right).
$$

Let $q(\hat{\tau}) = \mathbb{P}[\tau = \hat{\tau}, k(\sigma) \neq k(\hat{\tau})]$, so $\sum_{\hat{\tau} \in \mathcal{T}} q(\hat{\tau}) = Q$. Then

$$
\mathbb{E} \left[ G(\mu_{S(k(\sigma))\hat{\tau}}) \mid \tau = \hat{\tau} \right] - G(\mu_{\hat{\tau}}) =
\frac{p(\hat{\tau}) - q(\hat{\tau})}{p(\hat{\tau})} \left( \mathbb{E} \left[ G(\mu_{S(k(\hat{\tau}))\hat{\tau}}) - G(\mu_{\hat{\tau}}) \right] \right) + \frac{q(\hat{\tau})}{p(\hat{\tau})} \left( \mathbb{E} \left[ G(\mu_{S(k(\sigma))\hat{\tau}}) \mid \tau = \hat{\tau}, k(\sigma) \neq k(\hat{\tau}) \right] - G(\mu_{\hat{\tau}}) \right).
$$

The second term is at most $\frac{q(\hat{\tau})}{p(\hat{\tau})} M$, since $M$ is the range of $G$. To bound the first term, we note that $\mu_{S(k(\hat{\tau}))\hat{\tau}}$ cannot differ from $\mu_{\hat{\tau}}$ by more than $\frac{q(\hat{\tau})}{p(\hat{\tau}) - q(\hat{\tau})}$, as otherwise the average value of $\mu_{S(k(\sigma))\hat{\tau}}$ could not be $\mu_{\hat{\tau}}$. Therefore, $\mathbb{E} \left[ G(\mu_{S(k(\hat{\tau}))\hat{\tau}}) - G(\mu_{\hat{\tau}}) \right]$ is bounded by the largest possible difference in $G$-values of two points that differ by at most $\frac{q(\hat{\tau})}{p(\hat{\tau}) - q(\hat{\tau})}$. Therefore, we have

$$
\mathbb{E} \left[ D_G(\mu_{S(k(\sigma))\tau} \| \mu_{\tau}) \right] \leq \sum_{\hat{\tau} \in \mathcal{T}} p(\hat{\tau}) \left( \frac{p(\hat{\tau}) - q(\hat{\tau})}{p(\hat{\tau})} \tilde{G} \left( \frac{q(\hat{\tau})}{p(\hat{\tau}) - q(\hat{\tau})} \right) + \frac{q(\hat{\tau})}{p(\hat{\tau})} M \right)
\leq Q M + \sum_{\hat{\tau} \in \mathcal{T}} (p(\hat{\tau}) - q(\hat{\tau})) \tilde{G} \left( \frac{q(\hat{\tau})}{p(\hat{\tau}) - q(\hat{\tau})} \right),
$$

where $\tilde{G}$ is defined as in the statement of Lemma D.3.7. If $G$ is symmetric on $[0, 1]$, then $\tilde{G}(x) = G(0) - G(x)$ for $x \leq \frac{1}{2}$ and $M$ otherwise. This is a concave function, but $\tilde{G}$ is not in general concave. However, consider $\tilde{G}^*$ as defined in the lemma statement, so $\tilde{G}(x) \leq \tilde{G}^*(x)$ for all $x$.\footnotetext{This proof takes sums over $\hat{\tau} \in \mathcal{T}$ and thus implicitly assumes that $\mathcal{T}$ is finite, but the proof extends to infinite $\mathcal{T}$, with sums over $\tau$ replaced by integrals with respect to the probability measure over $\mathcal{T}$.}
Then

\[ \mathbb{E} \left[ D_G (\mu_{\hat{\tau}} \| \mu_\tau) \right] \leq QM + \sum_{\hat{\tau} \in \hat{T}} (p(\hat{\tau}) - q(\hat{\tau})) \tilde{G}^* \left( \frac{q(\hat{\tau})}{p(\hat{\tau}) - q(\hat{\tau})} \right) \]

\leq QM + \left( \sum_{\hat{\tau} \in \hat{T}} (p(\hat{\tau}) - q(\hat{\tau})) \right) \cdot \tilde{G}^* \left( \frac{\sum_{\hat{\tau} \in \hat{T}} q(\hat{\tau})}{\sum_{\hat{\tau} \in \hat{T}} (p(\hat{\tau}) - q(\hat{\tau}))} \right) \]

\[ = QM + (1 - Q) \tilde{G}^* \left( \frac{Q}{1 - Q} \right) \leq QM + \tilde{G}^*(Q) \leq 2 \tilde{G}^*(Q). \]

Here, the second step follows by Jensen’s inequality with terms \( \frac{q(\hat{\tau})}{p(\hat{\tau}) - q(\hat{\tau})} \) and weights \( p(\hat{\tau}) - q(\hat{\tau}) \), the second-to-last step follows from the fact that \( \tilde{G}^* \) is convex and \( \tilde{G}^*(0) = 0 \), and the last step follows from the fact that \( \tilde{G}^* \) is convex and \( \tilde{G}^*(1) = M \).

Since \( Q \leq 4 \left( \frac{\varepsilon}{\beta c} \right)^{1/(1 - \log_2 c)} \), combining Lemma D.3.7 with Equation D.8 gives us the following result.

\[ \mathbb{E} \left[ D_G (\mu_{\sigma \tau} \| \mu_\tau) \right] \leq \frac{8\beta}{c^2} + 2 \tilde{G}^* \left( 4 \left( \frac{\varepsilon}{\beta c} \right)^{1/(1 - \log_2 c)} \right). \]

Noting that \( \tilde{G}^* \) is concave and \( c^{-1/(1 - \log_2 c)} \leq 2 \) (which is true for all \( 0 < c < 1 \)) completes the proof of Lemma 8.4.16.

\[ \square \]

### D.4 Alternative definitions of agreement and accuracy

For arbitrary Bregman divergences, there are several notions of agreement and accuracy that are worth considering. Before we discuss these, we make a note about the order of arguments in a Bregman divergence. In our context, it makes the most sense to talk of the Bregman divergence \( \textit{from a more informed estimate to a less informed estimate} \). By a “more informed estimate” we mean a finer-grained one, i.e. one that is informed by more knowledge. For example, in terms of estimating \( Y \) in the context of this work explores, \( Y \) is more informed than \( \mu_{\sigma \tau} \), which is more informed than \( \mu_{\sigma} \) and \( \mu_{S\tau} \), which are each more informed than \( \mu_{ST} \), which is more informed than \( \mathbb{E}[Y] \).

To see that this is the natural order of the arguments, recall that Bregman divergences are
motivated by the property that they elicit the mean (see Proposition 2.1.8): if an agent who gives an estimate of \( x \) for the value of a random variable \( Y \) incurs a loss of \( D_G(Y \parallel x) \), then the agent minimizes their expected loss by reporting \( x = \mathbb{E} [Y] \). This means that the expert ought to report the expected value of \( Y \) given the information that the expert knows.

This means that given two estimates of \( Y \), \( Z_1 \) and \( Z_2 \), of which \( Z_1 \) is more informed, the quantity \( D_G(Z_1 \parallel Z_2) \) has a natural interpretation: it is the expected amount the expert gains by learning more and refining their estimate from \( Z_2 \) to \( Z_1 \). This follows by the Pythagorean theorem:

\[
\mathbb{E} \left[ D_G(Z_1 \parallel Z_2) \right] = \mathbb{E} \left[ D_G(Y \parallel Z_2) \right] - \mathbb{E} \left[ D_G(Y \parallel Z_1) \right].
\]

**D.4.1 Alternative definitions of agreement**

One important motivation for using the Jensen-Bregman divergence to the midpoint as the definition of agreement is that this quantity serves as a lower bound on the expected amount that Charlie disagrees with Alice and Bob. Formally:

**Definition D.4.1.** Let \( a, b, \) and \( c \) be Alice’s, Bob’s, and Charlie’s expectations, respectively (these are random variables on \( \Omega \)). Alice and Bob \( \epsilon \)-agree with Charlie if

\[
\frac{1}{2} \left( \mathbb{E} \left[ D_G(a \parallel c) + D_G(b \parallel c) \right] \right) \leq \epsilon.
\]

(This is the order of arguments because Alice and Bob are more informed than Charlie.) By Proposition D.3.1 (i), we know that **if Alice and Bob \( \epsilon \)-agree with Charlie then they \( \epsilon \)-agree**.

As it happens the fact that under this (stronger) definition of agreement implies accuracy under rectangle substitutes follows immediately:

**Proposition D.4.2.** Let \( I = (\Omega, \mathcal{F}, \sigma, \tau, Y) \) be an information structure that satisfies rectangle substitutes. For any communication protocol that causes Alice and Bob to \( \epsilon \)-agree with Charlie on \( I \), Alice and Bob are \( 2\epsilon \)-accurate after the protocol terminates.

**Proof.** Let \( S \) be the set of possible signals of Alice at the end of the protocol which are consistent with the protocol transcript, and define \( T \) likewise for Bob. Recall that Charlie’s expectation is

369
\( \mu_{ST} \). We have

\[
\mathbb{E} [ D_G(\mu_{\sigma T} \parallel \mu_{ST}) ] \leq \mathbb{E} [ D_G(\mu_{\sigma T} \parallel \mu_{ST}) ] \leq \mathbb{E} [ D_G(\mu_{\sigma T} \parallel \mu_{ST}) ] + \mathbb{E} [ D_G(\mu_{ST} \parallel \mu_{ST}) ] \leq 2\epsilon,
\]

where the first inequality follows by rectangle substitutes and the last inequality follows because Alice and Bob \( \epsilon \)-agree with Charlie. 

The drawback of Definition D.4.1 is that it is not so much a definition of Alice and Bob’s agreement with each other, so much as a definition of agreement with respect to the protocol being run (since Charlie only exists within the context of the protocol). Put otherwise, it is impossible to determine whether Alice and Bob \( \epsilon \)-agree with Charlie simply by knowing Alice and Bob’s expectations; one must also know Charlie’s expectation, which cannot be determined from Alice’s and Bob’s expectations. The question “how far from agreement are Alice and Bob if Alice believes 25\% and Bob believes 30\%?” makes sense in the context of \( \epsilon \)-agreement, but not in the context of \( \epsilon \)-agreement with Charlie.

A different notion of agreement, which (like \( \epsilon \)-agreement) only depends on Alice’s and Bob’s expectations, uses the symmetrized Bregman divergence between these expectations:

\[
\frac{1}{2}(D_G(a \parallel b) + D_G(b \parallel a)).
\]

**Definition D.4.3.** Let \( a \) and \( b \) be Alice’s and Bob’s expectations, respectively (these are random variables on \( \Omega \)). Alice and Bob satisfy symmetrized \( \epsilon \)-agreement if \( \frac{1}{2}(D_G(a \parallel b) + D_G(b \parallel a)) \).

By Proposition D.3.1 (iii), we know that if Alice and Bob satisfy symmetrized \( \epsilon \)-agreement then they \( \epsilon \)-agree.

In our context, symmetrized Bregman divergence is less natural than Jensen-Bregman divergence. This is symmetrized Bregman divergence (unlike Jensen-Bregman divergence) does not seem to closely relate to our previous discussion of the Bregman divergence from a more informed to a less informed estimate being most natural.
D.4.2 Alternative notions of accuracy

Our definition of Alice’s accuracy as the expected Bregman divergence from the truth $\mu_{rT}$ to Alice’s expectation seems like the most natural one. However, one may desire a definition of accuracy that takes both Alice’s and Bob’s expectations into account, judging the pair’s accuracy based on their consensus belief, rather than each of their individual beliefs. For instance, one could say that Alice and Bob are $\epsilon$-midpoint-accurate if $\mathbb{E} \left[ D_G \left( \mu_{rT} \parallel \frac{a+b}{2} \right) \right] \leq \epsilon$. By this definition, Alice’s and Bob’s expectations could individually be far from the truth, but they are considered accurate because the average of their expectations is close to correct.

**Proposition D.4.4.** If Alice and Bob are $\epsilon$-accurate, then they are $2\epsilon$-midpoint-accurate.

**Proof.** Observe that for all $a, b, y$ it is the case that

$$D_G \left( y \parallel \frac{a+b}{2} \right) \leq \max(D_G(y \parallel a), D_G(y \parallel b)) \leq D_G(y \parallel a) + D_G(y \parallel b).$$

The first inequality is true simply because $\frac{a+b}{2}$ lies in between $a$ and $b$. Therefore,

$$\mathbb{E} \left[ D_G \left( y \parallel \frac{a+b}{2} \right) \right] \leq \mathbb{E} [D_G(y \parallel a) + D_G(y \parallel b)] \leq 2\epsilon.$$

Another natural choice for Alice’s and Bob’s consensus belief is the QA pool (see Neyman and Roughgarden (2023)). Proposition D.4.4 likewise holds for the QA pool in place of the midpoint, and indeed holds for any choice of consensus belief that is guaranteed to lie in between Alice’s and Bob’s expectations. Thus, any such definition will be weaker than our definition of $\epsilon$-accuracy for Alice and Bob (up to a constant factor).

To summarize, among the above definitions of agreement, $\epsilon$-agreement is the weakest; and among the above definitions of accuracy, Alice’s and Bob’s $\epsilon$-accuracy is the strongest. This is an

371
indication of strength for Theorem 8.4.15: it starts from a relatively weak premise and reaches a relatively strong conclusion.

D.5 Implications for communication complexity

Our results can be framed in a communication complexity context, where they imply that “substitutable” functions can be computed with probability $1 - \delta$ (over the inputs) with a transcript length depending only on $\delta$. This is a nonstandard and weak notion of computing the function, but sketching the reduction may inspire future work on connections between substitutes and communication complexity.

In a classic deterministic communication complexity setup (e.g. Rao and Yehudayoff (2020)), Alice holds $\sigma \in S$, Bob holds $\tau \in T$, and the goal is to compute some function $g : S \times T \to \{0, 1\}$ using a communication protocol (see Section 8.2.2). Our setting captures this model when $Y = g(\sigma, \tau)$. Observe that in this case, $Y = \mu_{ST}$, i.e. Alice and Bob’s information together determine $Y$ completely. A communication protocol defines its output by a function $h : \Pi \to \{0, 1\}$ where $\Pi$ is the space of transcripts. We can simply let $h(\pi) = \text{round}(\mu_{ST})$, i.e. rounding the ex post expectation $\mathbb{E}[Y | \pi] = \mu_{ST}$ to either zero or one. This is equivalent to the belief of “Charlie”, or the common knowledge of Alice and Bob after the protocol is completed.

**Definition D.5.1** (Rectangle substitutes, $(1 - \delta)$-computes). Given a function $g$ and a distribution $\mathcal{D}$ over $S \times T$, we say $(g, \mathcal{D})$ satisfy rectangle substitutes if the corresponding information structure with $Y = g(\sigma, \tau)$ satisfies rectangle substitutes (Definition 8.2.6). We say a protocol $(1 - \delta)$-computes $g$ over $\mathcal{D}$ if, with probability at least $1 - \delta$ over $(\sigma, \tau) \sim \mathcal{D}$, the protocol has $h(\pi) = g(\sigma, \tau)$.

By our results, under rectangle substitutes $(g, \mathcal{D})$, any agreement protocol approximately computes $g$ over $\mathcal{D}$. More precisely, using a fast substitutes-agreement protocol similar to Proposition D.2.1, we obtain the following.
Corollary D.5.2. Suppose \((g, D)\) satisfy rectangle substitutes. Then for every \(\delta \in (0, 1)\), there is a deterministic communication protocol using \(O(\log(1/\delta))\) bits of communication that \((1 - \delta)\)-computes \(g\) over \(D\).

Proof. In round one, Alice sends her current expectation \(\mu_\sigma\) rounded to a multiple of \(\epsilon\); call this message \(A\). In round two, Bob sends his updated expectation \(\mu_{S_T}\) rounded to a multiple of \(\epsilon\); call this message \(B\). The protocol then halts, and the output is \(B\) rounded to either zero or one. It uses \(O(\log(1/\epsilon))\) bits. Let \(S, T\) be the random rectangle associated with the protocol.

By construction, \(|\mu_\sigma - A| \leq \epsilon\), and \(\mu_S\) is the expectation of \(Y\) conditioned on \(A\), so it follows that \(|\mu_\sigma - \mu_S| \leq \epsilon\). Using substitutes (just as in Proposition D.2.1),

\[
\mathbb{E} \left[ (\mu_{\sigma_T} - \mu_{S_T})^2 \right] \leq \mathbb{E} \left[ (\mu_\sigma - \mu_S)^2 \right] \leq \epsilon^2.
\]

By construction, \(|B - \mu_{S_T}| \leq \epsilon\). Therefore, by the \(\frac{1}{2}\)-approximate triangle inequality for squared distance (e.g. Proposition 8.4.10)),

\[
\mathbb{E} \left[ (\mu_{\sigma_T} - B)^2 \right] \leq 2\mathbb{E} \left[ (\mu_{\sigma_T} - \mu_{S_T})^2 \right] + 2\mathbb{E} \left[ (\mu_{S_T} - B)^2 \right] \leq 2\epsilon^2.
\]

Now, the protocol is incorrect if \(|B - \mu_{\sigma_T}| \geq \frac{1}{2}\). Using Markov’s inequality,

\[
P[|B - \mu_{\sigma_T}| \geq \frac{1}{2}] = P[(B - \mu_{\sigma_T})^2 \geq \frac{1}{4}] \\
\leq 4\mathbb{E} \left[ (B - \mu_{\sigma_T})^2 \right] \\
\leq 8\epsilon^2.
\]

Therefore, given \(\delta \in (0, 1)\), we run the protocol with \(\epsilon = \sqrt{\delta}/8\). The probability of an incorrect output is at most \(\delta\), and we use \(O(\log(1/\epsilon) = O(\log(1/\delta))\) bits of communication. \(\square\)