Published in final edited form as:
Pac Symp Biocomput. 2013 ; : 356–367.

METASEQ: PRIVACY PRESERVING META-ANALYSIS OF SEQUENCING-BASED ASSOCIATION STUDIES*

ANGAD PAL SINGH, **SAMREEN ZAFER**, and **ITSIK PE'ER**†
Department of Computer Science, Columbia University, New York, New York 10027-7003, USA

ANGAD PAL SINGH: aps2157@columbia.edu; SAMREEN ZAFER: sz2317@columbia.edu

Abstract

Human genetics recently transitioned from GWAS to studies based on NGS data. For GWAS, small effects dictated large sample sizes, typically made possible through meta-analysis by exchanging summary statistics across consortia. NGS studies groupwise-test for association of multiple potentially-causal alleles along each gene. They are subject to similar power constraints and therefore likely to resort to meta-analysis as well. The problem arises when considering privacy of the genetic information during the data-exchange process. Many scoring schemes for NGS association rely on the frequency of each variant thus requiring the exchange of identity of the sequenced variant. As such variants are often rare, potentially revealing the identity of their carriers and jeopardizing privacy. We have thus developed MetaSeq, a protocol for meta-analysis of genome-wide sequencing data by multiple collaborating parties, scoring association for rare variants pooled per gene across all parties. We tackle the challenge of tallying frequency counts of rare, sequenced alleles, for meta-analysis of sequencing data without disclosing the allele identity and counts, thereby protecting sample identity. This apparent paradoxical exchange of information is achieved through cryptographic means. The key idea is that parties encrypt identity of genes and variants. When they transfer information about frequency counts in cases and controls, the exchanged data does not convey the identity of a mutation and therefore does not expose carrier identity. The exchange relies on a 3rd party, trusted to follow the protocol although not trusted to learn about the raw data. We show applicability of this method to publicly available exome-sequencing data from multiple studies, simulating phenotypic information for powerful meta-analysis. The MetaSeq software is publicly available as open source.

1. Introduction

Human genetics has recently undergone a transition from genomewide association studies (GWAS) based on genotyping common polymorphisms[1–4] to studies based on next generation sequencing (NGS) data[5–7], that ascertains common and rare variants across individuals[8]. For GWAS, low effect sizes of most of the causal common alleles on common diseases and quantitative traits dictated large sample sizes to achieve statistical power[9]. In many studies, such sizes were made possible by consortia of multiple collaborating groups, each contributing hundreds or thousands of samples, together amassing tens or hundreds of thousands of genotyped samples to detect minute effects on various phenotypes[10]. Computational methods for meta-analysis of such collated GWAS datasets have been instrumental in facilitating their joint analysis[11].

*This work is supported by NSF grants CCF-0829882, CCF-0845677 and NIH grant U54-CA121852

†Author to which all correspondence should be addressed ip2169@columbia.edu.

NGS studies met initial success using only a handful of samples for sequencing exomes[12,13] or whole genomes[14,15] to detect novel, fully-penetrant alleles that disrupt genes and cause disease. Yet, detecting disease genes with rare alleles of partial penetrance, that explain only a small fraction of the cases, is more challenging. First, the limited power to detect such alleles on their own motivates testing for association of multiple alleles along the gene [16]. Indeed, multiple methods for groupwise testing of alleles have been developed to optimize power of detecting such multiply disrupted genes[17–22]. Second, the tautological problem with rare variants is their low frequency. Large numbers of samples are still required in order to observe such alleles and detect their significant association. Fortunately, the cost of NGS keeps dropping, and the throughput keeps increasing. Sequencing exomes now require reagent-cost and labor resources comparable to early GWAS, with genomes likely to soon follow. This paper is motivated by the assumption that these power constraints along with throughput opportunities will lead to large-scale disease sequencing studies[23] that would be more rapidly, and therefore more competitively executed by groups operating in parallel, but jointly meta-analyzing their data.

Privacy had been a thorny issue in genetics research[24–26]. The irreversible labeling of individuals if their genetic information is known requires broad consent by study participants in order for researchers to have the ethical right and legal permit to expose their genotype data or even to share it with peers and collaborators[27,28]. This, along with some investigators' sense of ownership of their data and cohorts typically makes data-access in human genetics (unlike other fields[29,30]) restricted, at least initially, often to the investigator. In GWAS, large consortia had preserved such access restriction, as meta-analysis required only exchange of summary statistics across collaborating groups and institutional barriers, rather than sharing explicit genotype data[31]. Such summary statistics typically include essentially allele frequencies (and their confidence levels) per marker. Although formally individuals and their relatives can be identified as members of a cohort just based on these summary statistics[32], this identification requires expert computation, and may be underpowered, depending on study parameters such as number of SNPs, sample size and allele frequencies[33].

Meta-analysis of sequencing data poses unique challenges in terms of subject privacy. Specifically, such data includes hundreds of thousands of rare alleles per genome[34], among them de novo mutations[35], one or two of which can uniquely identify an individual among the entire world population. Even exome sequences typically include thousands of alleles that are currently novel[13]. Even assuming future expansion of variant databases, a typical human exome will have thousands of very rare (frequency $< 10^{-4}$) alleles, typically singletons within a cohort of size in the low thousands. Such alleles, alone or in concert, readily provide unambiguous identification of carrier of the sample. The classical summary statistic for meta-analysis, which is the list of allele frequencies in a sample, therefore provides clear indication of membership for each and every sample in the cohort if applied genomewide with the exception of monozygotic twins, simply by virtue of including the singleton alleles carried by this sample. A similar rationale would decide or rule out membership in an exome-sequenced cohort based on presence of rare mutations. Yet, allele frequencies in cases and controls across the entire set of analyzed samples are a key ingredient in multiple methods for association to rare alleles[18,19,21]. Exchange of allele frequencies between consortium members in order to tally alleles across datasets is instrumental for meta-analysis of sequencing data, posing an apparent conflict with ethical requirements to protect against identification of samples.

This paper tackles the challenge of facilitating the tally of frequency counts of rare, sequenced alleles between consortium members, thus enabling meta-analysis of sequencing data while not disclosing the allele identity and counts, therefore providing considerable

protection of sample identity. This apparent paradoxical exchange of information is achieved through cryptographic means. The key idea is that parties hide the identity of the variants. When they transfer information about frequency counts in cases and controls, it does not convey the identity of a mutation, therefore not exposing the identity of the carriers. The parties do use an identical encryption key, thus identical variants will be encrypted identically. One could therefore sum up the counts for identical variants, without knowing the identity of the alleles whose counts are being tallied.

2. Methods

2.1. Notation

We hereby describe MetaSeq, a privacy preserving protocol for meta-analysis of sequencing data coming from C collaborators such that:

- Each collaborator c has data on a set $S[c]$ of samples.
- Such data includes a set $V_m[c]$ of positions along each gene g_m among the $M \sim$ 20,000 genes $g_1, g_2 \ldots g_M$. $V_m[c]$ specifies all positions where variant (no-reference) calls had been made for at least one sequenced individual $i \in S[c]$.
- The data further includes for each individual $i \in S[c]$, and each variant position $v \in V_m[c]$ the actual genotype of i at v: heterozygote or non-reference homozygote, denoted by $h_m[c](v,i)$, represented in a standard vcf format[36]. We define $H_m[c]$ to be full matrix of genotype values, across all rows $v \in V_m[c]$, and columns $i \in S[c]$. Effectively, $H_m[c]$ is a matrix of values 0,1, or 2 for each position and individual.
- For each individual $i \in S[c]$, the data also includes the affection status or the phenotype value of i, denoted by $p(i) \in \{1,0\}$ for cases and controls, respectively. We denote $P[c]$ as the list of phenotype values $p(i)$ for each $i \in S[c]$.

We assume $V_m[c]$ is listed as genomic coordinates: chromosome and position along the chromosome. For each such position x, we define the coordinate, $\phi_m(x)$, which is its offset from the start of the chromosome. We naturally extend $\phi_m(.)$ to operate on sets of positions. In practice we assume $\phi_m(x)$ is a 32-bit integer.

We define the set of all variable positions along the chromosome for gene, g_m, and the total set of individuals respectively, as follows:

$$V_m = \cup_c V_m[c] \quad (1)$$

$$S = \cup_c S[c] \quad (2)$$

We further define the full listing P of phenotype values for all individuals across all cohorts and the full set G of genes, $g_1 .. g_M \mid M \sim 20000$. H_m is defined as the genotype matrix across all cohorts, with columns for all $i \in S$, and rows for all $v \in V_m$. $H_m[c]$ is the minor of H_m induced on $V_m[c] \times S[c]$. The data for gene m is $D_m = \{V_m, H_m\}$, and the entire genetic dataset is given as:

$$D = \cup_m D_m \quad (3)$$

2.1.1. Association score—Let $F(D_m = (V_m, H_m), P)$ be the scoring function used for testing association of g_m. We assume F has certain properties that are shared by standard methods for testing association[18].

Specifically, F remains fixed when swapping rows (variants) of H_m along with V_m, if we assume all variants considered by the test are similarly likely to be causal (this assumption can be relaxed). Also, the set of scores for all genes by definition remains fixed when swapping genes g_m.

$$F(D,P)=\{F(D_m,P)\}_{m=1}^{M} \quad (4)$$

The goal of the protocol is to encrypt the data using a secret key k, such that gene labels and variant labels are swapped (or permuted). Specifically, we define key-dependent permutations g_k and ρ_k on gene labels and potential coordinates (32-bit integers), respectively. The permuted data for each gene is denoted by the following equations:

$$\rho_k(D_m)=(\rho_k(V_m),\rho_k(H_m)) \quad (5)$$

$$\rho_k(V_m)=\rho_k(\phi_m(v))|v \in V_m \quad (6)$$

where, $\rho_k(V_m)$ is the set of permuted coordinates and $\rho_k(H_m)$ is the matrix of genotype calls with permuted rows, i.e., with values $h_m[\rho_k(\phi_m(v)),i]$ for all $v \in V_m$, $i \in S$. We observe that the score is unchanged by this transformation: $F(\rho_k(D_m),P) = F(D_m,P)$. Yet, if one were to observe only a minor of $\rho_k(D_m)$, corresponding to a subset of individuals and the corresponding subsets of variants that they carry, one does not obtain any information on the individuals not in this subset, nor on the variants not carried by these individuals. Specifically, for each cohort c, the relevant subset of the data, $D_m[c] = (V_m[c], H_m[c])$, when encrypted into $\rho_k(D_m)[c]$, does not provide information regarding any other cohort $c' \neq c$, nor on any variants not in $V_m[c]$. In this sense, the encryption is privacy preserving. Finally, if gene labels are permuted, then receiver of the permuted data $D_{g(k)} = \{D_{gm(k)} | m \in 1..M\}$ cannot learn anything about the identity of any gene.

We have developed a 5-step protocol for meta-analysis of genomewide sequencing data, computing association scores for pooled rare variants. The protocol is presented here in simplified form, with the following leniencies:

1. We discuss only two-way meta-analysis, where two investigators (collaborators), Alice and Bob (or c_1 and c_2), each have their own sequenced association cohorts.
2. We consider case-control association testing.
3. We present the calculation of a simple variable allele-frequency threshold score[21].
4. Alice and Bob rely on the assistance of a semi-trusted third party, Trent, to help compute the score.

The protocol preserves privacy of the subjects in the following respects:

1. The only information Alice and Bob learn about each other's cohort is the scores of top-associated genes.
2. Trent does not have direct or practical information that could expose the identity of the subject in Alice and Bob's cohorts. Specifically, Trent does not learn which genes harbor which mutations in each cohort, and given an exome of an individual, cannot determine whether that individual is a member of any of the cohorts. Even upon publication of the research results by Alice and Bob, the information that Trent learns, is limited.

2.1.2. Protocol—The protocol proceeds as follows:

1. Key Exchange:

   Alice and Bob choose a shared secret key k, that can serve as an encryption key

2. Annotation and Encryption:

   Alice and Bob each encrypt their data as follows:

   a. Variants are annotated for the genes they belong to and variant classification, e.g. *known* or *nonsense*, needed for scoring. Such classification is kept unencrypted.

   b. Alice and Bob generate a secret permutation $g(k)$ over the set of genes $g_1 \ldots g_M$, creating permuted gene identifiers, $g_1(k) \ldots g_M(k)$.

   c. They further secretly permute the set of variants V_m, creating $V_m(k)$.

3. Data Transfer:

   Alice and Bob send Trent their encrypted gene names $g_1(k) \ldots g_M(k)$ and variant positions, $V_m(k)$ along with the (unencrypted) (frequency) counts $f_{V_m(k)[ci]}$.

4. Merging and association testing:

   Trent computes, for each (permuted) gene $g_m(k)$ a total count for each (encrypted) variant, $V_m(k)$ by summing the two counts $f_{V_m(k)[c1]}$ and $f_{V_m(k)[c2]}$ if both Alice and Bob report the variant in $g_m(k)$ or collapsing the association score for the variants otherwise.

5. Decrypt results:

   Trent sends Alice and Bob the (top) association scores assigned to specific (encrypted) gene names, that they are able to decrypt.

Note that many rare-variant association tests focus on particular type of variants, e.g. non-synonymous, or loss-of-function variants. Such information is lost upon encryption, and Trent will thus be unable to restrict analysis to a particular class of variants. A convenient workaround is to communicate a set of per-variant weights by both Alice and Bob. Weights depend on classification of variant type that is agreed upon in advance, i.e. Alice and Bob decide on a weight function $W: T \rightarrow [0,1]$ on the domain of all variant types $T = \{$*missense, synonymous coding* …..$\}$. Each variant v is assigned type $t(v) \in T$ and therefore a real-valued weight $W(t(v)) \in [0,1]$, is communicated to Trent in clear text. We make note of the fact that since both gene names and variant positions are encrypted, for a sufficiently large class of variant types it becomes difficult for Trent to make any concrete inferences on variant identities using this information.

2.1.3. Implementation: MetaSeq—We implemented this protocol as MetaSeq, an open source PERL package. A step-by-step illustration of the protocol as is in the MetaSeq code is given in Figure 1. We assume that the collaborators have their data stored on a server that is remotely accessible using the server name. We also require tools for annotation and encryption of the data on the server. MetaSeq works on variant call files (*.vcf format) that include genotypes and phenotypes for each collaborating party, and is available as open source at https://github.com/angadps/Rare-Variant-Association.

We provide implementation details regarding specific steps of MetaSeq:

Step 1: Registration & key exchange: MetaSeq guides the collaborating parties through the key exchange procedure using the PERL encryption modules Crypt::DES[37], Crypt::CFB[38] and Crypt::CBC[39], and allows an arbitrary number of collaborators, instead of

just the pair of Alice and Bob. In detail, the collaborators register with Trent using their server names. Communication between the servers is via the use of sockets. A specific port is designated on the servers for all data exchange and communication between the servers. Trent signals the key generation process after registration. All collaborators contribute a seed towards the generation of the key, of which Trent has no information about or contributes in any way towards the generation of either. We use the MD5 algorithm to generate a 32-bit key.

Step 2: Annotation & encryption: Each party then encrypts the data, which are first annotated by the vcfCodingSnps tool[40] on a per gene basis. The purpose of annotation is two-fold. Firstly, it helps us prepare the genotype and phenotype files separately for every gene as required by the association test. Secondly, it helps us in restricting the analysis to certain class of variants, or in assigning different weights to different classes. For that purpose, additional input to MetaSeq is a file of weights that needs to be agreed upon in advance. Variant data is encrypted per the protocol, and communicated as numeric 32-bit dumps – sufficient to uniquely index positions along any chromosome. At the same time we would like to point out that we have tested MetaSeq to work with gene level annotations only, although the idea could be extended to any general definitions of region for annotations as long as it is consistent across studies.

Step 4: Merging and association testing: MetaSeq is implemented with the Variance Threshold (VT) test[21] of association, but can in principle include other tests as well. The encrypted files received by Trent from Alice and Bob are first merged by their (encrypted) gene names. This prepares the data from all collaborators for the pooled association test.

2.2. Simulation Testing

We used simulation to evaluate the power of meta-analysis assuming different numbers of causal variants in a single gene. Power here is defined as the fraction of successful association tests. Specifically, for each such number, we simulated 100 datasets of 50 cases and 50 controls collected by each of C=10 collaborators. We tested association by each single-collaborator vs. pooled across collaborators in a privacy-preserving manner. We tallied the fraction of successful association tests, but note that reporting a success requires more care in this study than usual. In detail, a conservative definition of success is when the true gene is the unique top-scoring gene (for either single-collaborator or pooled testing modes). A more lenient definition allows other top-scoring genes to tie with the true gene (again, for both modes). Finally, without privacy-preserving data analysis, one can consider independent PIs running the association test, and then decide about the associated gene based on the individual results of all of them, by taking a majority vote. We report power based on each of these 5 modes of analysis. We repeated this for $1,2,2^2, \ldots 2^{10}$ causal variants for the causal gene, in addition to 1000 neutral variants for each gene. We simulated the case and control sequencing data using an implementation of the Wright-Fisher model[41], that allows setting particular numbers of causal and neutral variants. The Wright-Fisher Model gives the probability density function $f(p)$, of the probability of encountering a mutation, p as follows:

$$f(p)=c * p^{b_s-1} * (1-p)^{b_n-1} * e^{s(1-p)} \quad (7)$$

Here, $f(p)$ is the probability function of the mutation-probability p, b_s is the scaled mutation rate of disease mutations, b_n is the scaled back-mutation rate, s is the scaled selection rate and c is the constant that normalizes the integral of $f(p)$ to 1.

3. Results

3.1. Power of pooled-collaborators vs. single-PI testing

We report results from all the variants of the power tests stated above. Plots for the same are shown in Figure 2. Throughout the range of parameters, pooled tests are better powered compared to single-PI tests. This advantage is most pronounced when there are only few causal variants along the truly causal gene. At the extreme, 1–8 causal variants in a gene, we observe decently powered pooled test (up to 55% power for the conservative test) compared to a severely (<5%) underpowered single-PI test, an improvement of up to 50 percentage points or 10 – 30 times with the pooled tests. Naturally, lenient reporting of success enjoys higher power, but would potentially require following up multiple promising genes, rather than only one.

We note that the number of causal, case-only variants is a natural parameter here – the rare-allele analog of the size of effect to be detected. Power is further influenced by nuisance parameters, such as the span of a gene in basepairs (hence, the number of neutral variants along it, here normalized to be 1,000), and the genetic length of a gene in centimorgans (hence, the effective number of independent variants along it). This explains some of the genes being hard to find as associated, even with many rare case-only variants simulated. Potential false positives or false negatives in the context of meta-analysis alone are expected to be minimal (otherwise, the same concerns may apply as in the case of single cohort tests). Since variant frequencies are collapsed across all cohorts and for all variants in a gene, such loss of data, which is the primary input for the protocol is not expected. Also, encryption is performed in a loss-less manner i.e., no genes or variant ids are expected to be lost in the due course of execution of the protocol.

3.2. MetaSeq requirements of computing resources

We state the time and space requirements for MetaSeq in Table 1. The tests were run on a Sun Grid Engine controlled cluster with sufficient number of compute cores and maximum 8GB of RAM given to a single test at any time. We state the time and space that was required for a single run of MetaSeq with 1000 neutral variants per gene, broken down by small steps of the protocol. Some of the steps, i.e., registration, key generation, transfer of results, and decryption are insignificant both in terms of time and space. Yet, these steps are reported here for completion. In total, MetaSeq can be completed in 3.5 hours of elapsed time using less than 30 hours of CPU resources, using at its peak 150MB of space in total. Network footprint is even smaller, as transmitted files are archived and zipped. The most intensive parts in terms of computing resources are the annotation and encryption stages that need to I/O information in 200,000 files (one per gene per collaborator). The most CPU is used during association testing, for permuting the data 100,000 times to assess significance. We parallelize this stage over 20 cores.

4. Discussion

We developed MetaSeq, a protocol that relies on a trusted third-party to compute the association scores over the intersection of the variant set. We implemented the protocol in PERL and have made it available as an open source package. Our protocol is designed to be robust in securing private genetic information, while at the same time making only minimal assumptions about compliance of the parties to the protocol.

In securing private genetic information, we try to preserve privacy against participating collaborators knowing individual-level identifying information, such as private mutations. This is achieved by computing an association score, not by one of the parties, but rather by a designated third-party, who also needs to stay in the dark and not learn the identity of the

study participants and their private mutations. The third-party, after collating data and performing the desired computations, is assumed to follow protocol, and not to share variant information with any of the collaborators. The third-party is considered to be “trusted” in this regard. At the same time we need to secure information from the third-party as well. We achieve this by this party only working with encrypted data, never having access to the secret key that was used to encrypt all the genetic information. Hence, while the third-party has access to all the data, it is still meaningless to that party, since the data is in encrypted form and the encryption key is not available to it.

We make the assumption that no collaborator conspires with the third-party to share the key, as that would violate the desired privacy requirements. Another potential breach that can arise is when more than one collaborator plan to collate their datasets so as to draw inferences regarding the data from the remaining collaborators. However, such estimations can only be effectively made only if all but one of the collaborators get together and conspire against the remaining one. Even then, the coalition would, at best, learn limited information about the cohort of the conspired-upon collaborator, e.g., presence of variants that they already have in their cohort. The coalition will not learn the identity of private variants.

Another way that collaborators can violate protocol to learn the alleles is to send monomorphic data to the third-party for their own dataset. In this way they are sure that any identified carrier alleles are coming only from the datasets of other collaborators. This is possible only if all but one of the collaborators is sending monomorphic data, and we assume the parties follow protocol. At the same time it is assumed that there may be (approximately) a minimum of 5 collaborators in any run of the protocol. Under this assumption it is difficult for a single collaborator to learn the datasets of any other single collaborator by employing such mechanisms.

Finally, a collaborator may try to estimate datasets by computing a prior distribution of the results obtained from the final computation of scores, which is OK, and then use their own dataset to obtain a better posterior distribution. However, they only have a chance to learn about variants that are shared, rather than private to a cohort, and only within the top-scoring genes. A theoretical analysis of the privacy guarantees of the protocol may resemble the one by Sankararaman[42] to some extent although we are now working in the MAF < 0.5 range. A complete analysis however remains out of scope for this paper and will be considered for future work.

Privacy preserving protocols of this sort have been investigated in the cryptography literature as secure multiparty computation[43]. Over the last decade, protocols have been proposed for joint computation of the intersection of two or more subsets[44] that can be employed to compute the intersection of the variant set. More generally, theoretical results guarantee the ability to simulate any privacy-preserving protocol that uses a third trusted party without the need of such a party[45]. Similar to meta-analysis techniques in GWAS, the application of similar techniques for NGS studies is expected to reveal the role of many rare variants in Mendelian diseases.

Acknowledgments

We are grateful to Rajan Banerjee for his work in building some of the supporting cryptographic modules in MetaSeq.

We are also thankful to the NSF and the NIH for their funding provided via the NSF grants CCF-0829882, CCF-0845677 and the NIH grant U54-CA121852.

References

1. Burton PR, Clayton DG, et al. Nature. 2007; 447:661–78. [PubMed: 17554300]
2. Easton DF, et al. Nature. 2007; 447:1087–93. [PubMed: 17529967]
3. Hirschhorn JN, Daly MJ. Nat Rev Genet. 2005; 6:95–108. [PubMed: 15716906]
4. Sladek R, et al. Nature. 2007; 445:881–5. [PubMed: 17293876]
5. Harismendy O, et al. Genome Biol. 2009; 10:R32. [PubMed: 19327155]
6. Schuster SC. Nat Methods. 2008; 5:16–8. [PubMed: 18165802]
7. Shendure J. Genome Biol. 2011; 12:408. [PubMed: 21920048]
8. Shen Y, et al. Genome Res. 2010; 20:273–80. [PubMed: 20019143]
9. Spencer CC, Su Z, Donnelly P, Marchini J. PLoS Genet. 2009; 5:e1000477. [PubMed: 19492015]
10. Speliotes EK, et al. Nat Genet. 2010; 42:937–48. [PubMed: 20935630]
11. Evangelou E, Maraganore DM, Ioannidis JP. PLoS One. 2007; 2:e196. [PubMed: 17332845]
12. Bilguvar K, et al. Nature. 2010; 467:207–10. [PubMed: 20729831]
13. Ng SB, et al. Nature. 2009; 461:272–6. [PubMed: 19684571]
14. Lupski JR, et al. N Engl J Med. 2010; 362:1181–91. [PubMed: 20220177]
15. Roach JC, et al. Science. 2010; 328:636–9. [PubMed: 20220176]
16. Cohen JC, et al. Science. 2004; 305:869–72. [PubMed: 15297675]
17. Ionita-Laza I, Ottman R. Genetics. 2011; 189:1061–8. [PubMed: 21840850]
18. Li B, Leal SM. Am J Hum Genet. 2008; 83:311–21. [PubMed: 18691683]
19. Madsen BE, Browning SR. PLoS Genet. 2009; 5:e1000384. [PubMed: 19214210]
20. Neale BM, et al. PLoS Genet. 2011; 7:e1001322. [PubMed: 21408211]
21. Price AL, et al. Am J Hum Genet. 2010; 86:832–8. [PubMed: 20471002]
22. Zeggini E, et al. Nat Genet. 2008; 40:638–45. [PubMed: 18372903]
23. KAW. [Accessed Apr 17, 2012] DNA Sequencing Costs: Data from the NHGRI Large-Scale Genome Sequencing Program. 2012. Available at: http://www.genome.gov/sequencingcosts/
24. Fuller BP, et al. Science. 1999; 285:1359–61. [PubMed: 10490410]
25. Lin Z, Owen AB, Altman RB. Science. 2004; 305:183. [PubMed: 15247459]
26. Regalado A. Wall St J (East Ed). 2002:R10. [PubMed: 12542058]
27. Caulfield T, et al. PLoS Biol. 2008; 6:e73. [PubMed: 18366258]
28. Kaye J, Heeney C, Hawkins N, de Vries J, Boddington P. Nat Rev Genet. 2009; 10:331–5. [PubMed: 19308065]
29. Berman HM, et al. Nucleic Acids Res. 2000; 28:235–42. [PubMed: 10592235]
30. Edgar R, Domrachev M, Lash AE. Nucleic Acids Res. 2002; 30:207–10. [PubMed: 11752295]
31. de Bakker PI, et al. Hum Mol Genet. 2008; 17:R122–8. [PubMed: 18852200]
32. Homer N, et al. PLoS Genet. 2008; 4:e1000167. [PubMed: 18769715]
33. Jacobs KB, et al. Nat Genet. 2009; 41:1253–7. [PubMed: 19801980]
34. Durbin RMAG, Altshuler DL, et al. Nature. 2010; 467:1061–73. [PubMed: 20981092]
35. Au KS, et al. Genet Med. 2007; 9:88–100. [PubMed: 17304050]
36. Danecek P, et al. Bioinformatics. 2011; 27:2156–8. [PubMed: 21653522]
37. http://search.cpan.org/~dparis/Crypt-DES-2.05/
38. http://search.cpan.org/~kjh/Crypt-CFB-0.02/
39. http://search.cpan.org/~lds/Crypt-CBC-2.30/
40. http://www.sph.umich.edu/csg/liyanmin/vcfCodingSnps/index.shtml
41. Cheung YH, Wang G, Leal SM, Wang S. Genet Epidemiol. 2012
42. Sankararaman S, Obozinski G, Jordan MI, Halperin E. Nat Genet. 2009; 41:965–7. [PubMed: 19701190]
43. Yao, A. Protocols for secure computation. 23rd FOCS; 1982. p. 160-164.
44. Song, LKaD. School of Computer Science. Carnegie Mellon University; 2004. CMU-CS-04-182

45. Chow SSM, JL, Subramanian L. Proc NDSS. 2009

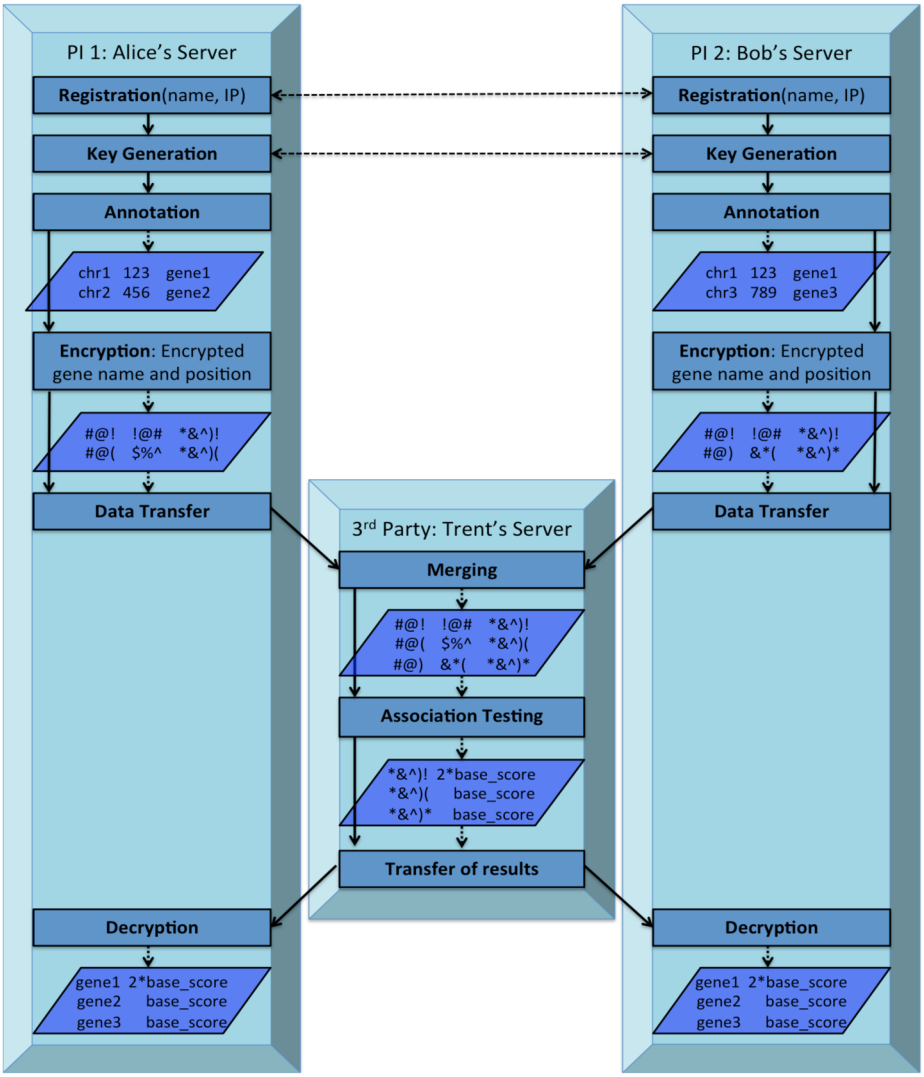

Figure 1.

Flow diagram of MetaSeq: Two Investigators, Alice and Bob compute per-gene scores on their pooled data without revealing the data to one another nor to a third party, Trent, who computes association scores "blindfolded". The figure describes a simple scenario using three genes, one of which, including a single variant in it, is common to Alice and Bob. The gene name and variant position for this is encrypted to the same text, thus being merged together by the 3rd party before association testing. This gene scores higher compared to other genes, as shown in the results decrypted by individual collaborators on their servers. Note that the generation of the key to be used for encryption is coordinated between Alice and Bob, excluding Trent in the process. Also note that while the figure does not point out phenotype information explicitly, the association testing step of the protocol receives the frequency data segregated for case and control cohorts, respectively.

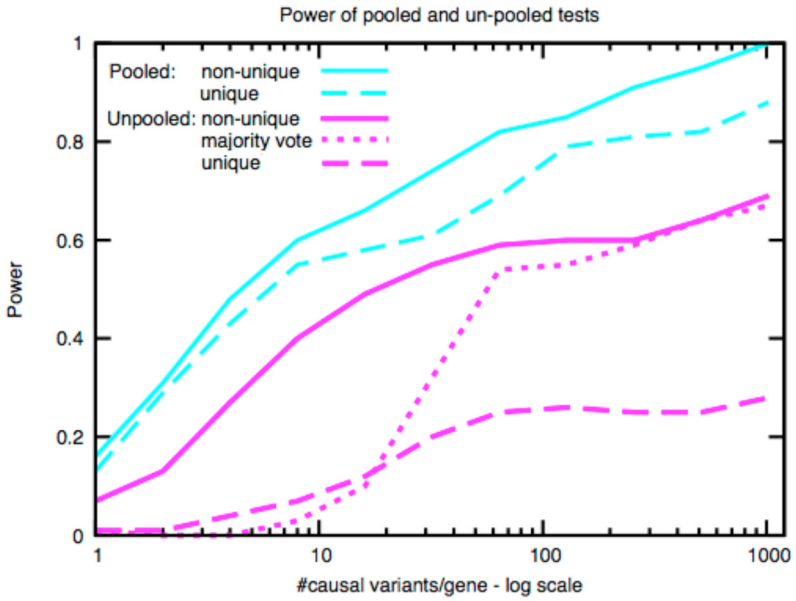

Figure 2.

Log-scaled power plot for pooled and single-PI tests. Results are plotted for all three definitions of success (unique and non-unique causal gene for pooled and single-PI tests, majority vote for single-PI tests only). In the unique and non-unique gene plots for the single-PI tests, the final success rate is calculated by averaging the number of successes across all PIs per dataset. In the majority vote plot, a majority vote of the number of successes is taken per single-PI per dataset. The different nature of success here explains the region in the figure where the plot for single-PI unique gene tests is higher than the majority vote plot.

Table 1

Space and time requirements of MetaSeq. The benchmark runs included 10 collaborators, with each one contributing 100 samples to the pooled analysis including 1000 neutral variants per gene. Runs include all ~20,000 genes along the genome. Steps performed by the collaborating parties ("Alice & Bob", though in this benchmark also 8 other collaborators) are evaluated for resources required per party. Also time taken for the association test mentioned is with a parallelism of 20. A total of 20 CPU hours were effectively needed for the association testing, although total memory required is less than 1MB. Note that decryption takes negligible time as opposed to encryption since the parties only need to decrypt the list of top-scoring gene names.

Step	Performed by	Elapsed time [min]	CPU time [min]	Memory [MB]
1.1 Register	Alice & Bob	Nil	Nil	Nil
1.2 Generate key		Nil	Nil	Nil
2.1 Annotate		18	180	15
2.2 Encrypt		27	270	12
3 Transfer data		1	10	12
4.1 Merge	Trent	80	80	105
4.2 Test association		60	1200	1
5.1 Transfer results		Nil	Nil	Nil
5.2 Decrypt	Alice & Bob	Nil	Nil	Nil