In neuroscience, Hopfield networks [1] are the classical biologically plausible model of long-term memory, relying on Hebbian plasticity for storage and attractor dynamics for recall. In contrast, memory networks in machine learning commonly use a key-value mechanism [2], but it has remained unclear whether it can be implemented by biological systems. Here, we propose a set of biologically plausible three-factor plasticity rules [3] for a simple key-value memory network [4] which performs on par with classical Hopfield networks, and can be naturally extended to more complex tasks -- including manipulation of individual stored memories -- or embedded in a larger neural system. Overall, our results suggest a compelling alternative to the classical Hopfield network as a model of biological long-term memory.

**Key-value memory storage and retrieval**

\[ h_t = \text{softmax}(K_t \bar{x}) \]

\[ \bar{y} = V_t h_t = \sum_{i=1}^{N} h_{ti} y_i, \]

\[ K_{t+1} = (1 - \eta_t) \odot K_t + \eta_t \odot [1 \odot \gamma_t] \]

\[ \gamma_t \]

[global third factor (external input)]

[local third factor (sequential or random)]

\[ h_{t+1} = \text{softmax}(K_{t+1} x_t) \]

**Fast memory sub-network**

Using our network in conjunction with a gradient-based network to learn smooth functions. With few samples, the fast-learning key-value network outperforms the gradient network, but with more samples the slower gradient network generalizes. Weighting the sub-network outputs based on their confidence allows inference of the output better than either alone.

**Meta-learned plasticity rules**

Plasticity rules meta-learned via gradient descent are qualitatively similar to the hand-designed network: presynaptic non-Hebbian for writing keys, and Hebbian for writing values. Performance and capacity of the meta-learned network remain unchanged.

\[ K_{t+1} = (1 - \eta_t) \odot K_t + \eta_t \odot [\odot \lambda_t (h_{t+1}^T \odot (y_{t+1}))] \]

\[ V_{t+1} = (1 - \eta_t) \odot V_t + \eta_t \odot [\odot \lambda_t (h_{t+1}^T \odot (y_{t+1}))] \]

**Heteroassociative tasks**

With distinct input and output layers, our network can be naturally extended to heteroassociative memory. Including recurrent connections from the output to the input, the network performs sequence learning. Embedding our network in a larger system enables storing and recalling arbitrary input sequences.