Boltzmann Distribution Optimization

Andrew Moore

Summer 2023

Consider an Ising circuit f with N inputs, M outputs, and A auxilliaries. The total spin space is thus $\Sigma := \Sigma_N \times \Sigma_M \times \Sigma_A \cong \mathbb{Z}_2^{N+M+A}$. Now, we would like to design and optimization problem that searches for the Hamiltonian polynomial on Σ that minimizes the worst-case Boltzmann probability of a wrong answer. For any input $\sigma \in \Sigma_N$, define I_σ to be the input level, R_σ to be the set of right answers and W_σ to be the set of wrong answers:

$$I_\sigma := \{\sigma\} \times \Sigma_M \times \Sigma_A \subset \Sigma \tag{1}$$

$$R_\sigma := \{\sigma\} \times \{f(\sigma)\} \times \Sigma_A \subset I_\sigma \tag{2}$$

$$W_\sigma := \{\sigma\} \times (\Sigma_M \setminus \{f(\sigma)\}) \times \Sigma_A \subset I_\sigma \tag{3}$$

The optimization objective can therefore be explicitly written as

$$\arg\min_H \max_{\sigma \in \Sigma_N} \frac{\sum_{\xi \in W_\sigma} \exp(-\beta H(\xi))}{\sum_{\xi \in I_\sigma} \exp(-\beta H(\xi))} \tag{4}$$

Note that since log is an increasing function, taking the log of the inner expression will not affect the result, since it will preserve the ordering of all values. Therefore, we can solve the equivalent problem

$$\arg\min_H \max_{\sigma \in \Sigma_N} \log \frac{\sum_{\xi \in W_\sigma} \exp(-\beta H(\xi))}{\sum_{\xi \in I_\sigma} \exp(-\beta H(\xi))} \tag{5}$$

$$= \arg\min_H \max_{\sigma \in \Sigma_N} \log \sum_{\xi \in W_\sigma} \exp(-\beta H(\xi)) - \log \sum_{\xi \in I_\sigma} \exp(-\beta H(\xi)) \tag{6}$$

Now, replace that max with a soft max for a large weight parameter λ:

$$\arg\min_H \frac{1}{\lambda} \log \sum_{\sigma \in \Sigma_N} \exp\left(\lambda \log \sum_{\xi \in W_\sigma} \exp(-\beta H(\xi)) - \lambda \log \sum_{\xi \in I_\sigma} \exp(-\beta H(\xi))\right) \tag{7}$$

$$= \arg\min_H \sum_{\sigma \in \Sigma_N} \exp\left(\lambda \log \sum_{\xi \in W_\sigma} \exp(-\beta H(\xi)) - \lambda \log \sum_{\xi \in I_\sigma} \exp(-\beta H(\xi))\right) \tag{8}$$

This presents the opportunity for stochastic gradient descent, using where the set Σ_N is the "dataset". At this point, the only thing stopping us from just attempting the optimization is the potentially large computational complexity of computing the two inner sums: even if we cache the values for W_σ while computing the second sum over I_σ, we are still evaluating the Hamiltonian 2^{M+A} times just to find the value of the loss function on a single datapoint. This is not very practical, even for relatively small problems. At this point we can employ a trick. Suppose that we use a SGD batch size of 1. Then we only need

$$\nabla \exp\left(\lambda \log \sum_{\xi\in W_\sigma} \exp(-\beta H(\xi)) - \lambda \log \sum_{\xi\in I_\sigma} \exp(-\beta H(\xi))\right) \tag{9}$$

$$= \lambda \exp(\cdots)\, \nabla \left(\log \sum_{\xi\in W_\sigma} \exp(-\beta H(\xi)) - \log \sum_{\xi\in I_\sigma} \exp(-\beta H(\xi))\right) \tag{10}$$

$$\propto \nabla \left(\log \sum_{\xi\in W_\sigma} \exp(-\beta H(\xi)) - \log \sum_{\xi\in I_\sigma} \exp(-\beta H(\xi))\right) \tag{11}$$

Thus if we sacrifice the step size information from the gradient and settle for SGD with a batch size of 1 and a normalized step vector, we can ignore the proportionality constant, which is always positive and therefore does not affect the step vector direction. The SGD problem thus becomes

$$\arg\min_H \sum_{\sigma\in\Sigma_N} \sum_{S\in\{W_\sigma, I_\sigma\}} (-1)^{\iota(S)} \log \sum_{\xi\in S} \exp(-\beta H(\xi)) \tag{12}$$

Where $\iota(S) = 1$ if $S = I_\sigma$ and -1 if $S = W_\sigma$. But since we've already decided that we're doing SGD with batch size 1 and normalized steps, we can just pull the same trick again, at the cost of even more stochastic noise:

$$\nabla \left[(-1)^{\iota(S)} \log \sum_{\xi\in S} \exp(-\beta H(\xi))\right] = (-1)^{\iota(S)} \frac{1}{\sum_{\xi\in S} \exp(-\beta H(\xi))} \sum_{\xi\in S} \nabla \exp(-\beta H(\xi)) \tag{13}$$

$$\propto \sum_{\xi\in S} \nabla \left[(-1)^{\iota(S)} \exp(-\beta H(\xi))\right] \tag{14}$$

Our (now extremely stochastic) gradient descent problem thus becomes

$$\arg\min_H \sum_{\sigma\in\Sigma_N, S\in\{W_\sigma, I_\sigma\}, \xi\in S} (-1)^{\iota(S)} \exp(-\beta H(\xi)) \tag{15}$$

Where we will be normalizing every step and using a batch size of 1 over the above sum. Due to the high noise in this system, the learning rate should be kept very small. However, each step is now extremely cheap.