On Helmholtz Machines

The Helmholtz machine [1] is a generative model that consists of two networks, a bottom-up recognition network that takes the data as input and produces a distribution over hidden variables (inference), and a top-down generative network that maps values of the hidden variables to the data space (generator). In an unsupervised learning regime, the wake-sleep algorithm may be used for training.

1 Variational Inference

Suppose we have a generative model of some data x characterized by some hidden variables z and parameters θ. The log likelihood function can be written as:

$$\log p_\theta(x) = \log \int p_\theta(x,z)dz = \log \int p_\theta(x \mid z)p_\theta(z)dz. \tag{1}$$

Equivalently, we can express the above in terms of an energy function:

$$\mathcal{E}_z = -\log p_\theta(x,z). \tag{2}$$

In terms of the energy function, we can write the posterior of z given x as:

$$P_z = p(z \mid x, \theta) = \frac{p_\theta(x,z)}{\int p_\theta(x,z')dz'} = \frac{\exp(-\mathcal{E}_z)}{\int \exp(-\mathcal{E}_{z'})dz'}. \tag{3}$$

This is known as the Boltzmann distribution. The denominator is known as the partition function $Z = \int \exp(-\mathcal{E}_z)dz$. We can express Z in terms of $\mathcal{E}_z$ and P_z as $Z = \frac{\exp(-\mathcal{E}_z)}{P_z}$. Then:

$$\log Z = \log \left[\frac{\exp(-\mathcal{E}_z)}{P_z}\right] = -\mathcal{E}_z - \log P_z \tag{4}$$

which holds for all z. Equation (4) also holds if we take the expectation with respect to P_z:

$$\log Z = \langle -\mathcal{E}_z - \log P_z \rangle_{P_z} \tag{5}$$

where the left hand-side is identical since it does not depend on z.

In fact, since taking the expectation of the left-hand side of (4) has no effect, we can do this for any arbitrary distribution Q_z over z. Then:

$$\log Z = \langle -\mathcal{E}_z - \log P_z \rangle_{Q_z} = \underbrace{-\int Q_z \mathcal{E}_z dz - \int Q_z \log Q_z)}_{-\mathcal{F}(\mathcal{E},Q)} + \underbrace{\int Q_z \log \frac{Q_z}{P_z}}_{KL(Q,P)}. \tag{6}$$

This equation gives us a prescription for obtaining the Q that approximates P best. Note, the left-hand side of (6), $\log Z$, is independent of Q, it is just the marginal log likelihood of the data. The second term on right,is the Kullback-Leibler divergence $KL(Q, P) > 0$ between Q and P, which we know is non-negative. ideally, we want to minimize $KL(Q, P)$ as much as possible. (6) suggests that we can minimize $KL(Q, P)$ by maximizing the Helmholtz free energy $-\mathcal{F}(\mathcal{E}, Q)$. Since the terms $KL(Q, P)$ and $-\mathcal{F}(E, Q)$ add up to the constant $\log Z$, maximizing $-\mathcal{F}(E, Q)$, necessarily minimizes $KL(Q, P)$.

2 Helmholtz Machine

The Helmholtz machine is a particular instantiation of the variational inference framework described above. It is a traditional neural network, in the sense that it has an input layer and a number of stacked hidden layers on top of each other. However, the consecutive layers are connected through both feedforward and feedback connections. The feedforwared connections are also known as recognition weights and the feedback connections are known as generative weights.

Recall, we need to calculate the Helmholtz free energy $\mathcal{F}(\mathcal{E}, Q)$ where $\mathcal{E}$ is the energy function and Q is the approximative posterior. To make this calculation tractable, we are going to assume that Q is a separable distribution in each layer. In other words, given the activities of all units in layer l, the units in layer $l+1$ are conditionally independent. More specifically, Q_z takes the following form:

$$Q_z = \prod_{l>1} \prod_{j} [q_j^l(\phi, s^{l-1})]^{s_j^l} [1 - q_j^l(\phi, s^{l-1})]^{s_j^l} \tag{7}$$

where s_j^l denotes the binary stochastic activity of the j-th unit in the l=th layer, s^l is the vector of activities of all units in the l-th layer and q_j^l is the probability of being active for the j-th neuron in the l-th layer:

$$q_j^l(\phi, s^{l-1}) = \sigma \left(\sum_i s_i^{l-1} \phi_{i,j}^{l-1,l} \right) \tag{8}$$

where $\sigma(\cdot)$ denotes the sigmoid function and $\phi_{i,j}^{l-1,l}$ is the feedforward (recognition) weight between the i-th unit in layer $l-1$ and the j-th unit in layer l. In practive, [1] use an approximation where the stochastic activies s_j^l are replaced by their means q_j^l, as in mean field models:

$$q_j^l(\phi, q^{l-1}) = \sigma \left(\sum_i q_i^{l-1} \phi_{i,j}^{l-1,l} \right) . \tag{9}$$

We make a similar separability assumption for the joint distribution $p_\theta(x, z)$ that determines the energy function $\mathcal{E}_z$ (2):

$$p_\theta(x,z) = \prod_{l\geq 1}\prod_{j}[p_j^l(\theta, s^{l-1})]^{s_j^l}[1 - p_j^l(\theta, s^{l-1})]^{s_j^l} \tag{10}$$

where p_j^l is the activation probability of the j-th unit in the l-th layer in the generative model for which a mean-field type approximation similar to the one used for q_j^l above:

$$p_j^l(\theta, q^{l+1}) = \left(1 - \frac{1}{1 + \sum_k q_k^{l+1}\theta_{k,j}^{l+1,l}}\right)\left(1 - \prod_k \left[1 - q_k^{l+1}\frac{\theta_{k,j}^{l+1,l}}{1 + \theta_{k,j}^{l+1,l}}\right]\right). \tag{11}$$

It turns out that the obvious analogue of (9) for p_j^l does not work well in practice due to local minima. Hence, (**??**) is a remedy to escape local minima.

The first layer activations q^1 constitue the data x in the model. With this factorized approximation of $p_\theta(x,z)$, we can write $-\mathcal{F}(\mathcal{E}, Q)$ as:

$$-\mathcal{F}(\mathcal{E}, Q) = \langle -\mathcal{E}_z - \log Q_z\rangle_{Q_z} \propto \sum_x \sum_l \sum_j q_j^l \log\frac{q_l^l}{p_j^l} + (1 - q_j^l)\log\frac{1 - q_j^l}{1 - p_j^l} \tag{12}$$

where q_j^l and p_j^l are given by (9) and (11), respectively, and the sum over x represents an average over the data. The derivatives of $\mathcal{F}(\mathcal{E}, Q)$ with respect the recognition and generative weights, ϕ and θ, can be easily calculated using the chain rule and are given in the Appendix of [1].

References

[1] P. Dayan, G. E. Hinton, R. N. Neal, and R. S. Zemel. The Helmholtz machine. *Neural Computation*, 7:889–904, 1995.