

19.5 Learned Approximate Inference

We have seen that inference can be thought of as an optimization procedure that increases the value of a function \mathcal{L} . Explicitly performing optimization via iterative procedures such as fixed point equations or gradient-based optimization is often very expensive and time-consuming. Many approaches to inference avoid this expense by learning to perform approximate inference. Specifically, we can think of the optimization process as a function f that maps an input \mathbf{v} to an approximate distribution $q^* = \arg \max_q \mathcal{L}(\mathbf{v}, q)$. Once we think of the multi-step iterative optimization process as just being a function, we can approximate it with a neural network that implements an approximation $\hat{f}(\mathbf{v}; \boldsymbol{\theta})$.

19.5.1 Wake-Sleep

One of the main difficulties with training a model to infer \mathbf{h} from \mathbf{v} is that we do not have a supervised training set with which to train the model. Given a \mathbf{v} , we do not know the appropriate \mathbf{h} . The mapping from \mathbf{v} to \mathbf{h} depends on the choice of model family, and evolves throughout the learning process as $\boldsymbol{\theta}$ changes. The wake-sleep algorithm (Hinton *et al.*, 1995b; Frey *et al.*, 1996) resolves this problem by drawing samples of both \mathbf{h} and \mathbf{v} from the model distribution. For example, in a directed model, this can be done cheaply by performing ancestral sampling beginning at \mathbf{h} and ending at \mathbf{v} . The inference network can then be trained to perform the reverse mapping: predicting which \mathbf{h} caused the present \mathbf{v} . The main drawback to this approach is that we will only be able to train the inference network on values of \mathbf{v} that have high probability under the model. Early in learning, the model distribution will not resemble the data distribution, so the inference network will not have an opportunity to learn on samples that resemble data.

In section 18.2 we saw that one possible explanation for the role of dream sleep in human beings and animals is that dreams could provide the negative phase samples that Monte Carlo training algorithms use to approximate the negative gradient of the log partition function of undirected models. Another possible explanation for biological dreaming is that it is providing samples from $p(\mathbf{h}, \mathbf{v})$ which can be used to train an inference network to predict \mathbf{h} given \mathbf{v} . In some senses, this explanation is more satisfying than the partition function explanation. Monte Carlo algorithms generally do not perform well if they are run using only the positive phase of the gradient for several steps then with only the negative phase of the gradient for several steps. Human beings and animals are usually awake for several consecutive hours then asleep for several consecutive hours. It is