

meaning we define the joint probability distribution using an energy function:

$$P(\mathbf{x}) = \frac{\exp(-E(\mathbf{x}))}{Z}, \quad (20.1)$$

where  $E(\mathbf{x})$  is the energy function and  $Z$  is the partition function that ensures that  $\sum_{\mathbf{x}} P(\mathbf{x}) = 1$ . The energy function of the Boltzmann machine is given by

$$E(\mathbf{x}) = -\mathbf{x}^\top \mathbf{U} \mathbf{x} - \mathbf{b}^\top \mathbf{x}, \quad (20.2)$$

where  $\mathbf{U}$  is the “weight” matrix of model parameters and  $\mathbf{b}$  is the vector of bias parameters.

In the general setting of the Boltzmann machine, we are given a set of training examples, each of which are  $n$ -dimensional. Equation 20.1 describes the joint probability distribution over the observed variables. While this scenario is certainly viable, it does limit the kinds of interactions between the observed variables to those described by the weight matrix. Specifically, it means that the probability of one unit being on is given by a linear model (logistic regression) from the values of the other units.

The Boltzmann machine becomes more powerful when not all the variables are observed. In this case, the latent variables, can act similarly to hidden units in a multi-layer perceptron and model higher-order interactions among the visible units. Just as the addition of hidden units to convert logistic regression into an MLP results in the MLP being a universal approximator of functions, a Boltzmann machine with hidden units is no longer limited to modeling linear relationships between variables. Instead, the Boltzmann machine becomes a universal approximator of probability mass functions over discrete variables (Le Roux and Bengio, 2008).

Formally, we decompose the units  $\mathbf{x}$  into two subsets: the visible units  $\mathbf{v}$  and the latent (or hidden) units  $\mathbf{h}$ . The energy function becomes

$$E(\mathbf{v}, \mathbf{h}) = -\mathbf{v}^\top \mathbf{R} \mathbf{v} - \mathbf{v}^\top \mathbf{W} \mathbf{h} - \mathbf{h}^\top \mathbf{S} \mathbf{h} - \mathbf{b}^\top \mathbf{v} - \mathbf{c}^\top \mathbf{h}. \quad (20.3)$$

**Boltzmann Machine Learning** Learning algorithms for Boltzmann machines are usually based on maximum likelihood. All Boltzmann machines have an intractable partition function, so the maximum likelihood gradient must be approximated using the techniques described in chapter 18.

One interesting property of Boltzmann machines when trained with learning rules based on maximum likelihood is that the update for a particular weight connecting two units depends only the statistics of those two units, collected under different distributions:  $P_{\text{model}}(\mathbf{v})$  and  $\hat{P}_{\text{data}}(\mathbf{v})P_{\text{model}}(\mathbf{h} \mid \mathbf{v})$ . The rest of the