



Figure 16.14: An RBM drawn as a Markov network.

and

$$p(\mathbf{v} \mid \mathbf{h}) = \prod_i p(v_i \mid \mathbf{h}). \quad (16.12)$$

The individual conditionals are simple to compute as well. For the binary RBM we obtain:

$$P(h_i = 1 \mid \mathbf{v}) = \sigma \left( \mathbf{v}^\top \mathbf{W}_{:,i} + b_i \right), \quad (16.13)$$

$$P(h_i = 0 \mid \mathbf{v}) = 1 - \sigma \left( \mathbf{v}^\top \mathbf{W}_{:,i} + b_i \right). \quad (16.14)$$

Together these properties allow for efficient **block Gibbs** sampling, which alternates between sampling all of  $\mathbf{h}$  simultaneously and sampling all of  $\mathbf{v}$  simultaneously. Samples generated by Gibbs sampling from an RBM model are shown in figure 16.15.

Since the energy function itself is just a linear function of the parameters, it is easy to take its derivatives. For example,

$$\frac{\partial}{\partial W_{i,j}} E(\mathbf{v}, \mathbf{h}) = -v_i h_j. \quad (16.15)$$

These two properties—efficient Gibbs sampling and efficient derivatives—make training convenient. In chapter 18, we will see that undirected models may be trained by computing such derivatives applied to samples from the model.

Training the model induces a representation  $\mathbf{h}$  of the data  $\mathbf{v}$ . We can often use  $\mathbb{E}_{\mathbf{h} \sim p(\mathbf{h}|\mathbf{v})}[\mathbf{h}]$  as a set of features to describe  $\mathbf{v}$ .

Overall, the RBM demonstrates the typical deep learning approach to graphical models: representation learning accomplished via layers of latent variables, combined with efficient interactions between layers parametrized by matrices.

The language of graphical models provides an elegant, flexible and clear language for describing probabilistic models. In the chapters ahead, we use this language, among other perspectives, to describe a wide variety of deep probabilistic models.