

so that the burn in operation does not take as many steps.

The **contrastive divergence** (CD, or CD- $k$  to indicate CD with  $k$  Gibbs steps) algorithm initializes the Markov chain at each step with samples from the data distribution (Hinton, 2000, 2010). This approach is presented as algorithm 18.2. Obtaining samples from the data distribution is free, because they are already available in the data set. Initially, the data distribution is not close to the model distribution, so the negative phase is not very accurate. Fortunately, the positive phase can still accurately increase the model's probability of the data. After the positive phase has had some time to act, the model distribution is closer to the data distribution, and the negative phase starts to become accurate.

---

**Algorithm 18.2** The contrastive divergence algorithm, using gradient ascent as the optimization procedure.

---

Set  $\epsilon$ , the step size, to a small positive number.

Set  $k$ , the number of Gibbs steps, high enough to allow a Markov chain sampling from  $p(\mathbf{x}; \boldsymbol{\theta})$  to mix when initialized from  $p_{\text{data}}$ . Perhaps 1-20 to train an RBM on a small image patch.

**while** not converged **do**

    Sample a minibatch of  $m$  examples  $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\}$  from the training set.

$\mathbf{g} \leftarrow \frac{1}{m} \sum_{i=1}^m \nabla_{\boldsymbol{\theta}} \log \tilde{p}(\mathbf{x}^{(i)}; \boldsymbol{\theta})$ .

**for**  $i = 1$  to  $m$  **do**

$\tilde{\mathbf{x}}^{(i)} \leftarrow \mathbf{x}^{(i)}$ .

**end for**

**for**  $i = 1$  to  $k$  **do**

**for**  $j = 1$  to  $m$  **do**

$\tilde{\mathbf{x}}^{(j)} \leftarrow \text{gibbs\_update}(\tilde{\mathbf{x}}^{(j)})$ .

**end for**

**end for**

$\mathbf{g} \leftarrow \mathbf{g} - \frac{1}{m} \sum_{i=1}^m \nabla_{\boldsymbol{\theta}} \log \tilde{p}(\tilde{\mathbf{x}}^{(i)}; \boldsymbol{\theta})$ .

$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \epsilon \mathbf{g}$ .

**end while**

---

Of course, CD is still an approximation to the correct negative phase. The main way that CD qualitatively fails to implement the correct negative phase is that it fails to suppress regions of high probability that are far from actual training examples. These regions that have high probability under the model but low probability under the data generating distribution are called **spurious modes**. Figure 18.2 illustrates why this happens. Essentially, it is because modes in the model distribution that are far from the data distribution will not be visited by