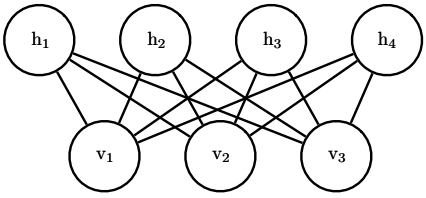
Learning in Probabilistic Models: RBMs, DBMs, DBNs (chapters 18, 20)

Restricted Boltzmann Machines: recap



Restricted Boltzmann Machine

Energy-based Undirected Model

$$P(\mathbf{v} = \boldsymbol{v}, \mathbf{h} = \boldsymbol{h}) = \frac{1}{Z} \exp\left(-E(\boldsymbol{v}, \boldsymbol{h})\right)$$

$$E(\boldsymbol{v}, \boldsymbol{h}) = -\boldsymbol{b}^\top \boldsymbol{v} - \boldsymbol{c}^\top \boldsymbol{h} - \boldsymbol{v}^\top \boldsymbol{W} \boldsymbol{h}$$

visible units: observable variables

hidden units: not observable variables

partition function

$$Z = \sum_{\boldsymbol{v}} \sum_{\boldsymbol{h}} \exp\left\{-E(\boldsymbol{v}, \boldsymbol{h})\right\}$$

Binary visible and hidden variables

$p(\boldsymbol{v})$ is intractable, however both $p(\boldsymbol{h}|\boldsymbol{v})$ and $p(\boldsymbol{v}|\boldsymbol{h})$ are *factorial* and easy to compute!

$$p(\boldsymbol{h}|\boldsymbol{v}) = \prod_{j=1}^{n_h} p(h_j|\boldsymbol{v})$$

$$p(h_j = 1|\boldsymbol{v}) = \sigma(c_j + \boldsymbol{v}^\top \boldsymbol{W}_{:,j})$$

Training Restricted Boltzmann Machines

Now that we know how to perform Gibbs sampling, we are ready to understand how a RBM can be trained!

- Given training data $\mathcal{T} = \{\boldsymbol{x}^{(i)}\}_{i=1}^{n}$ (to be clamped to the visible units) we want to find parameters $\boldsymbol{\theta} = \{\boldsymbol{W}, \boldsymbol{b}, \boldsymbol{c}\}$ that maximize $p(\mathcal{T}|\boldsymbol{\theta})$
- We know it is easier to maximize the Log-Likelihood

$$\log p(\mathcal{T}|\boldsymbol{\theta}) = \log \frac{1}{Z(\boldsymbol{\theta})} \tilde{p}(\mathcal{T}|\boldsymbol{\theta}) = \log \tilde{p}(\mathcal{T}|\boldsymbol{\theta}) - \log Z(\boldsymbol{\theta}),$$

where

$$\log \tilde{p}(\mathcal{T}|\boldsymbol{\theta}) = \log \underbrace{\prod_{i=1}^{n} \tilde{p}(\boldsymbol{x}^{(i)}|\boldsymbol{\theta})}_{i.i.d.\ samples} = \sum_{i=1}^{n} \log \tilde{p}(\boldsymbol{x}^{(i)}|\boldsymbol{\theta})$$

- We proceed by gradient ascent, so we need to compute

$$\begin{aligned} \nabla_{\boldsymbol{\theta}} \log p(\mathcal{T}|\boldsymbol{\theta}) &= \nabla_{\boldsymbol{\theta}} \log \tilde{p}(\mathcal{T}|\boldsymbol{\theta}) - \nabla_{\boldsymbol{\theta}} \log Z(\boldsymbol{\theta}) \\ &= \sum_{i=1}^{n} \nabla_{\boldsymbol{\theta}} \log \tilde{p}(\boldsymbol{x}^{(i)}|\boldsymbol{\theta}) - \nabla_{\boldsymbol{\theta}} \log Z(\boldsymbol{\theta}) \end{aligned}$$

Training Restricted Boltzmann Machines

- The term $\sum_{i=1}^{n} \nabla_{\boldsymbol{\theta}} \log \tilde{p}(\boldsymbol{x}^{(i)}|\boldsymbol{\theta})$ is not difficult to deal with

$$\begin{aligned} \sum_{i=1}^{n} \nabla_{\boldsymbol{\theta}} \log \tilde{p}(\boldsymbol{x}^{(i)}|\boldsymbol{\theta}) &= \sum_{i=1}^{n} \frac{1}{\tilde{p}(\boldsymbol{x}^{(i)}|\boldsymbol{\theta})} \nabla_{\boldsymbol{\theta}} \tilde{p}(\boldsymbol{x}^{(i)}|\boldsymbol{\theta}) \\ &= (\boldsymbol{x}^{(i)} \text{ is clamped to } \boldsymbol{v}^{(i)}, \text{ i.e. } \boldsymbol{x}^{(i)} = \boldsymbol{v}^{(i)}) \\ &= \sum_{i=1}^{n} \frac{1}{\sum_{\boldsymbol{h}} \exp(-E(\boldsymbol{v}^{(i)}, \boldsymbol{h}))} \nabla_{\boldsymbol{\theta}} \sum_{\boldsymbol{h}} \exp(-E(\boldsymbol{v}^{(i)}, \boldsymbol{h})) \\ &= -\sum_{i=1}^{n} \sum_{\boldsymbol{h}} \underbrace{\frac{\exp(-E(\boldsymbol{v}^{(i)}, \boldsymbol{h}))}{\sum_{\boldsymbol{h}} \exp(-E(\boldsymbol{v}^{(i)}, \boldsymbol{h}))}}_{= p(\boldsymbol{h}|\boldsymbol{v}^{(i)})} \nabla_{\boldsymbol{\theta}} E(\boldsymbol{v}^{(i)}, \boldsymbol{h}) \\ &= -\sum_{i=1}^{n} \sum_{\boldsymbol{h}} p(\boldsymbol{h}|\boldsymbol{v}^{(i)}) \nabla_{\boldsymbol{\theta}} E(\boldsymbol{v}^{(i)}, \boldsymbol{h}) \end{aligned}$$

$$\frac{\exp(-E(\boldsymbol{v}^{(i)}, \boldsymbol{h}))}{\sum_{\boldsymbol{h}} \exp(-E(\boldsymbol{v}^{(i)}, \boldsymbol{h}))} = \frac{\tilde{p}(\boldsymbol{h}, \boldsymbol{v}^{(i)})}{\tilde{p}(\boldsymbol{v}^{(i)})} \frac{Z}{Z} = \frac{p(\boldsymbol{h}, \boldsymbol{v}^{(i)})}{p(\boldsymbol{v}^{(i)})} = p(\boldsymbol{h}|\boldsymbol{v}^{(i)})$$

Training Restricted Boltzmann Machines

Knowing that $p(\boldsymbol{h}|\boldsymbol{v}^{(i)}) = \prod_{j=1}^{n_h} p(h_j|\boldsymbol{v}^{(i)})$, we have (recall that $\boldsymbol{x}^{(i)} = \boldsymbol{v}^{(i)}$)

$$\sum_{i=1}^{n} \nabla_{\boldsymbol{\theta}} \log \tilde{p}(\boldsymbol{x}^{(i)}|\boldsymbol{\theta}) \quad = \quad -\sum_{i=1}^{n} \sum_{\boldsymbol{h}} \prod_{j=1}^{n_h} p(h_j|\boldsymbol{v}^{(i)}) \nabla_{\boldsymbol{\theta}} E(\boldsymbol{v}^{(i)}, \boldsymbol{h})$$

Recalling that $\forall j, h_j \in \{0,1\}$, $\nabla_{\boldsymbol{\theta}} \log \tilde{p}(\boldsymbol{x}^{(i)}|\boldsymbol{\theta}) = \nabla_{\boldsymbol{W},\boldsymbol{b},\boldsymbol{c}} \log \tilde{p}(\boldsymbol{x}^{(i)}|\boldsymbol{\theta})$, if we now focus on $\nabla_{\boldsymbol{W}}$, we have for any q and z (notice that $\frac{\partial E(\boldsymbol{v},\boldsymbol{h})}{\partial w_{qz}} = \frac{\partial(-\boldsymbol{b}^\top \boldsymbol{v} - \boldsymbol{c}^\top \boldsymbol{h} - \boldsymbol{v}^\top \boldsymbol{W} \boldsymbol{h})}{\partial w_{qz}} = -v_q h_z$)

$$\begin{aligned}
\sum_{i=1}^{n} \frac{\partial \log \tilde{p}(\boldsymbol{x}^{(i)}|\boldsymbol{\theta})}{\partial w_{qz}} &= -\sum_{i=1}^{n} \sum_{\boldsymbol{h}} \prod_{j=1}^{n_h} p(h_j|\boldsymbol{v}^{(i)}) \frac{\partial E(\boldsymbol{v}^{(i)}, \boldsymbol{h})}{\partial w_{qz}} \\
&= \sum_{i=1}^{n} \sum_{h_1,\ldots,h_{n_h}} \prod_{j=1}^{n_h} p(h_j|\boldsymbol{v}^{(i)}) v_q^{(i)} h_z \\
&= \sum_{i=1}^{n} \sum_{h_z \in \{0,1\}} p(h_z|\boldsymbol{v}^{(i)}) v_q^{(i)} h_z \underbrace{\sum_{\substack{h_1,\ldots,\\ h_{z-1},h_{z+1},\\ \ldots,h_{n_h}}} \prod_{\substack{j=1\\ j\neq z}}^{n_h} p(h_j|\boldsymbol{v}^{(i)})}_{=1}
\end{aligned}$$

Training Restricted Boltzmann Machines

$$\begin{aligned}
\sum_{i=1}^{n} \frac{\partial \log \tilde{p}(\boldsymbol{x}^{(i)}|\boldsymbol{\theta})}{\partial w_{qz}} &= \sum_{i=1}^{n} \sum_{h_z \in \{0,1\}} p(h_z|\boldsymbol{v}^{(i)}) v_q^{(i)} h_z \\
&= \sum_{i=1}^{n} p(h_z = 1|\boldsymbol{v}^{(i)}) v_q^{(i)}
\end{aligned}$$

and recalling that $p(h_z = 1|\boldsymbol{v}) = \sigma(c_z + \boldsymbol{v}^\top \boldsymbol{W}_{:,z})$

$$\sum_{i=1}^{n} \frac{\partial \log \tilde{p}(\boldsymbol{x}^{(i)}|\boldsymbol{\theta})}{\partial w_{qz}} \quad = \quad \sum_{i=1}^{n} \sigma(c_z + {\boldsymbol{v}^{(i)}}^\top \boldsymbol{W}_{:,z}) v_q^{(i)}$$

Training Restricted Boltzmann Machines

- As we already discussed, the term $\nabla_{\boldsymbol{\theta}} \log Z(\boldsymbol{\theta})$ is difficult to deal with. In fact, recalling that $Z(\boldsymbol{\theta}) = \sum_{\boldsymbol{v},\boldsymbol{h}} \exp(-E(\boldsymbol{v},\boldsymbol{h}))$, we have

$$\begin{aligned}
\nabla_{\boldsymbol{\theta}} \log Z(\boldsymbol{\theta}) &= \frac{1}{\sum_{\boldsymbol{v},\boldsymbol{h}} \exp(-E(\boldsymbol{v},\boldsymbol{h}))} \nabla_{\boldsymbol{\theta}} \sum_{\boldsymbol{v},\boldsymbol{h}} \exp(-E(\boldsymbol{v},\boldsymbol{h})) \\
&= -\sum_{\boldsymbol{v},\boldsymbol{h}} \underbrace{\frac{\exp(-E(\boldsymbol{v},\boldsymbol{h}))}{\sum_{\boldsymbol{v},\boldsymbol{h}} \exp(-E(\boldsymbol{v},\boldsymbol{h}))}}_{= p(\boldsymbol{v},\boldsymbol{h})} \nabla_{\boldsymbol{\theta}} E(\boldsymbol{v},\boldsymbol{h}) \\
&= -\sum_{\boldsymbol{v},\boldsymbol{h}} p(\boldsymbol{v},\boldsymbol{h}) \nabla_{\boldsymbol{\theta}} E(\boldsymbol{v},\boldsymbol{h}) \\
&= -\sum_{\boldsymbol{v},\boldsymbol{h}} \underbrace{p(\boldsymbol{v}) p(\boldsymbol{h}|\boldsymbol{v})}_{\substack{product \\ rule}} \nabla_{\boldsymbol{\theta}} E(\boldsymbol{v},\boldsymbol{h}) \text{ (alternative: } p(\boldsymbol{h}) p(\boldsymbol{v}|\boldsymbol{h})\text{)} \\
&= -\sum_{\boldsymbol{v}} p(\boldsymbol{v}) \sum_{\boldsymbol{h}} p(\boldsymbol{h}|\boldsymbol{v}) \nabla_{\boldsymbol{\theta}} E(\boldsymbol{v},\boldsymbol{h})
\end{aligned}$$

Training Restricted Boltzmann Machines

If we now focus on $\nabla_{\boldsymbol{W}}$, we have for any q and z

$$\begin{aligned}
\frac{\partial \log Z(\boldsymbol{\theta})}{\partial w_{qz}} &= -\sum_{\boldsymbol{v}} p(\boldsymbol{v}) \underbrace{\sum_{\boldsymbol{h}} p(\boldsymbol{h}|\boldsymbol{v}) \frac{\partial E(\boldsymbol{v},\boldsymbol{h})}{\partial w_{qz}}}_{= -\sigma(c_z + \boldsymbol{v}^\top \boldsymbol{W}_{:,z}) v_q} \\
&= \sum_{\boldsymbol{v}} p(\boldsymbol{v}) \sigma(c_z + \boldsymbol{v}^\top \boldsymbol{W}_{:,z}) v_q
\end{aligned}$$

which is intractable for regular sized RBMs because its complexity is still exponential in the size of $\boldsymbol{v}$ (or $\boldsymbol{h}$)

Recalling that we are assuming binary input/hidden units:

$size(\boldsymbol{v})$ ➡ with k input units the size is 2^k

Training Restricted Boltzmann Machines

Summing up

due to partition function needs Gibbs sampling

$$\frac{\partial \log p(\boldsymbol{x}|\boldsymbol{\theta})}{\partial w_{qz}} = \underbrace{\sum_{i=1}^{n} \sigma(c_z + \boldsymbol{v}^{(i)\mathsf{T}} \boldsymbol{W}_{:,z}) v_q^{(i)}}_{data} - \underbrace{\sum_{\boldsymbol{v}} p(\boldsymbol{v}) \sigma(c_z + \boldsymbol{v}^{\mathsf{T}} \boldsymbol{W}_{:,z}) v_q}_{model}$$

$$\propto \langle v_q h_z \rangle_{\text{data}} - \langle v_q h_z \rangle_{\text{model}}$$

and in a similar way it is possibile to get the $\nabla_{\boldsymbol{b}}$ and $\nabla_{\boldsymbol{c}}$ components of the gradient

$$\frac{\partial \log p(\boldsymbol{x}|\boldsymbol{\theta})}{\partial b_q} = \sum_{i=1}^{n} v_q^{(i)} - \sum_{\boldsymbol{v}} p(\boldsymbol{v}) v_q$$

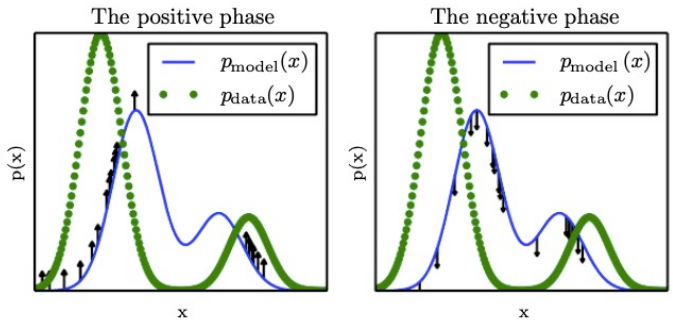
due to partition function needs Gibbs sampling

$$\frac{\partial \log p(\boldsymbol{x}|\boldsymbol{\theta})}{\partial c_z} = \sum_{i=1}^{n} \sigma(c_z + \boldsymbol{v}^{\mathsf{T}} \boldsymbol{W}_{:,z}) - \sum_{\boldsymbol{v}} p(\boldsymbol{v}) \sigma(c_z + \boldsymbol{v}^{\mathsf{T}} \boldsymbol{W}_{:,z})$$

Training Restricted Boltzmann Machines

Algorithm 18.1 A naive MCMC algorithm for maximizing the log-likelihood with an intractable partition function using gradient ascent

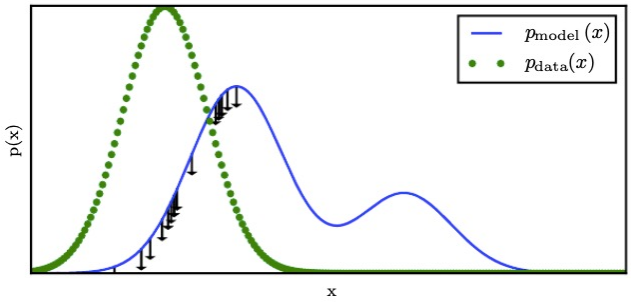
Set ϵ, the step size, to a small positive number.
Set k, the number of Gibbs steps, high enough to allow burn in. Perhaps 100 to train an RBM on a small image patch.
while not converged **do**
  Sample a minibatch of m examples $\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}\}$ from the training set
  $\mathbf{g} \leftarrow \frac{1}{m} \sum_{i=1}^{m} \nabla_{\boldsymbol{\theta}} \log \tilde{p}(\mathbf{x}^{(i)}; \boldsymbol{\theta})$.
  Initialize a set of m samples $\{\tilde{\mathbf{x}}^{(1)}, \ldots, \tilde{\mathbf{x}}^{(m)}\}$ to random values (e.g., from a uniform or normal distribution, or possibly a distribution with marginals matched to the model's marginals)
  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



Training Restricted Boltzmann Machines

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

Set ϵ, 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_{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**
      $\tilde{\mathbf{x}}^{(j)} \leftarrow$ 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



Training Restricted Boltzmann Machines

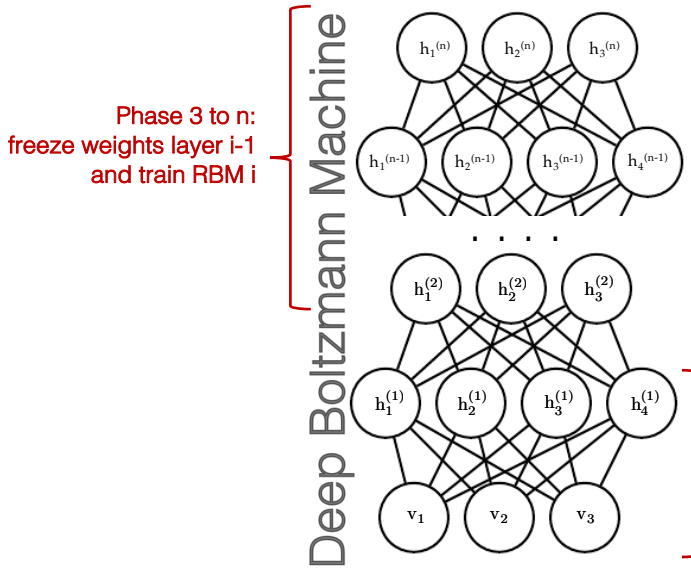
Algorithm 18.3 The stochastic maximum likelihood / persistent contrastive divergence algorithm using gradient ascent as the optimization procedure

Set ϵ, the step size, to a small positive number.
Set k, the number of Gibbs steps, high enough to allow a Markov chain sampling from $p(\mathbf{x};\boldsymbol{\theta}+\epsilon\mathbf{g})$ to burn in, starting from samples from $p(\mathbf{x};\boldsymbol{\theta})$. Perhaps 1 for RBM on a small image patch, or 5–50 for a more complicated model like a DBM.

Initialize a set of m samples $\{\tilde{\mathbf{x}}^{(1)},\dots,\tilde{\mathbf{x}}^{(m)}\}$ to random values (e.g., from a uniform or normal distribution, or possibly a distribution with marginals matched to the model's marginals).
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 k **do**
    for $j = 1$ to m **do**
      $\tilde{\mathbf{x}}^{(j)} \leftarrow$ 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

RBM-based Deep Models

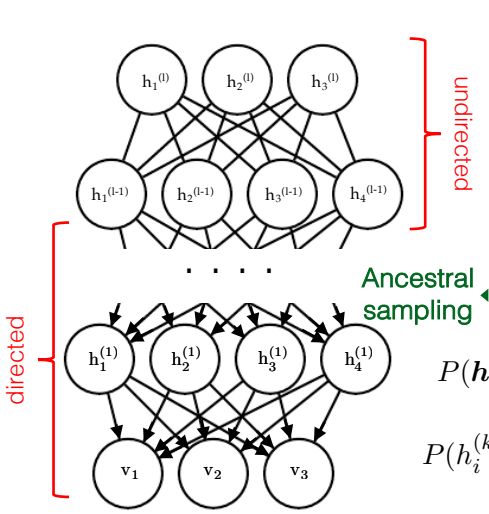
Layer-wise approach: greadily stack RMBs (suboptimal)



Training procedure can also be used as (pre-)trainig of a Deep Belief Network

$$p(\boldsymbol{v}, \boldsymbol{h}^{(1)}, \boldsymbol{h}^{(2)}, \ldots, \boldsymbol{h}^{(l-1)}, \boldsymbol{h}^{(l)}) = \frac{1}{Z(\boldsymbol{\theta})} \exp(-E(\boldsymbol{v}, \boldsymbol{h}^{(1)}, \boldsymbol{h}^{(2)}, \ldots, \boldsymbol{h}^{(l-1)}, \boldsymbol{h}^{(l)} | \boldsymbol{\theta}))$$

Deep Belief Networks



Mixed model:
- top undirected
- rest directed towards visible units

binary hidden units

$$P(\boldsymbol{h}^{(l)}, \boldsymbol{h}^{(l-1)}) \propto \exp\left(\boldsymbol{b}^{(l)\top}\boldsymbol{h}^{(l)} + \boldsymbol{b}^{(l-1)\top}\boldsymbol{h}^{(l-1)} + \boldsymbol{h}^{(l-1)\top}\boldsymbol{W}^{(l)}\boldsymbol{h}^{(l)}\right), \quad \text{(undirected)}$$

$$P(h_i^{(k)} = 1 \mid \boldsymbol{h}^{(k+1)}) = \sigma\left(b_i^{(k)} + \boldsymbol{W}_{:,i}^{(k+1)\top}\boldsymbol{h}^{(k+1)}\right) \forall i, \forall k \in 1, \ldots, l-2, \quad \text{(directed)}$$

$$P(v_i = 1 \mid \boldsymbol{h}^{(1)}) = \sigma\left(b_i^{(0)} + \boldsymbol{W}_{:,i}^{(1)\top}\boldsymbol{h}^{(1)}\right) \forall i.$$

In the case of real-valued visible units, substitute

$$\mathbf{v} \sim \mathcal{N}\left(\boldsymbol{v}; \boldsymbol{b}^{(0)} + \boldsymbol{W}^{(1)\top}\boldsymbol{h}^{(1)}, \boldsymbol{\beta}^{-1}\right)$$

diagonal