

RBM: Log-likelihood gradient

Sargur N. Srihari
srihari@cedar.buffalo.edu

Topics in Partition Function

- Definition of Partition Function
 1. The log-likelihood gradient
 1. RBM Log-likelihood gradient
 2. Stochastic maximum likelihood and contrastive divergence
 3. Pseudolikelihood
 4. Score matching and Ratio matching
 5. Denoising score matching
 6. Noise-contrastive estimation
 7. Estimating the partition function

RBM with visible and hidden units

- Joint configuration (\mathbf{v}, \mathbf{h})
 - visible and hidden units has an energy (Hopfield 1982)

$$E(\mathbf{v}, \mathbf{h}) = - \sum_{i \in \text{visible}} a_i v_i - \sum_{j \in \text{hidden}} b_j h_j - \sum_{i,j} v_i h_j w_{ij}$$

- Network assigns a probability to every pair of hidden and visible vectors

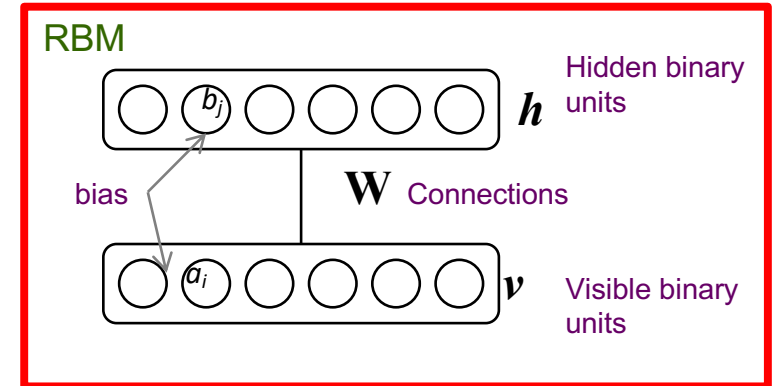
$$p(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} e^{-E(\mathbf{v}, \mathbf{h})}$$

- where partition function Z is a sum over all possible pairs of visible/hidden vectors

$$Z = \sum_{\mathbf{v}, \mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h})}$$

- Probability that network assigns to a visible vector \mathbf{v} is

$$p(\mathbf{v}) = \frac{1}{Z} \sum_{\mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h})}$$



Stochastic binary pixels \mathbf{v} connected to stochastic binary feature detectors \mathbf{h} using symmetrically weighted connections

Changing probability of image

- Probability network assigns to a training image is raised by adjusting weights and biases
 - Lower the energy of that image & raise energy of other images
 - Especially those that have low energies and make a high contribution to the partition function
- Maximum likelihood approach to determine $W, \mathbf{h}, \mathbf{v}$

$$\text{Likelihood: } P(\{\mathbf{v}^{(1)}, \dots, \mathbf{v}^{(M)}\}) = \prod_m p(\mathbf{v}^{(m)})$$

Log-likelihood:

$$\ln P(\{\mathbf{v}^{(1)}, \dots, \mathbf{v}^{(M)}\}) = \sum_m \ln p(\mathbf{v}^{(m)}) = \sum_m \ln \left(\frac{1}{Z} \sum_{\mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h})^{(m)}} \right) = \sum_m \ln \left(\sum_{\mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h})^{(m)}} \right) - \sum_m \ln \left(\sum_{\mathbf{v}, \mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h})} \right)$$

Derivative of the log-probability of a training vector wrt a weight:

$$\frac{\partial \ln p(\mathbf{v})}{\partial w_{ij}} = \mathbb{E}_{\text{data}}(v_i h_j) - \mathbb{E}_{\text{model}}(v_i h_j)$$

Learning rule for stochastic steepest ascent

$$\Delta w_{ij} = \varepsilon \left(\mathbb{E}_{\text{data}}(v_i h_j) - \mathbb{E}_{\text{model}}(v_i h_j) \right), \text{ where } \varepsilon \text{ is the learning rate}$$

$$p(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} e^{-E(\mathbf{v}, \mathbf{h})}$$

$$p(\mathbf{v}) = \frac{1}{Z} \sum_{\mathbf{h}} e^{-E(\mathbf{v}, \mathbf{h})}$$

$$E(\mathbf{v}, \mathbf{h}) = - \sum_{i \in \text{visible}} a_i v_i - \sum_{j \in \text{hidden}} b_j h_j - \sum_{i,j} v_i h_j w_{ij}$$

$$\frac{\partial}{\partial w_{i,j}} E(\mathbf{v}, \mathbf{h}) = -v_i h_j$$

$$\frac{d}{dx} \ln x = \frac{1}{x}$$

Samples for Computing Expectations

- Getting unbiased samples for $E_{\text{data}}(v_i h_j)$

- Given a random training image \mathbf{v} , the binary state h_j for each hidden unit is set to **1** with probability

$$p(h_j = 1 \mid \mathbf{v}) = \sigma \left(b_j + \sum_i v_i w_{ij} \right)$$

- Given a random training image \mathbf{v} , the binary state v_i for a visible unit is set to **1** with probability

$$p(v_i = 1 \mid \mathbf{h}) = \sigma \left(a_i + \sum_j h_j w_{ij} \right)$$

- Getting unbiased samples for $E_{\text{model}}(v_i h_j)$

- Can be done by starting at a random state of visible units and performing Gibbs sampling for a long time
- One iteration of alternating Gibbs sampling consists of updating all hidden units in parallel followed by updating all visible units

Summary of RBM training

Probability Distribution of Undirected model (Gibbs)

$$p(\mathbf{x}; \boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \tilde{p}(\mathbf{x}, \boldsymbol{\theta})$$

$$Z(\boldsymbol{\theta}) = \sum_{\mathbf{x}} \tilde{p}(\mathbf{x}, \boldsymbol{\theta})$$

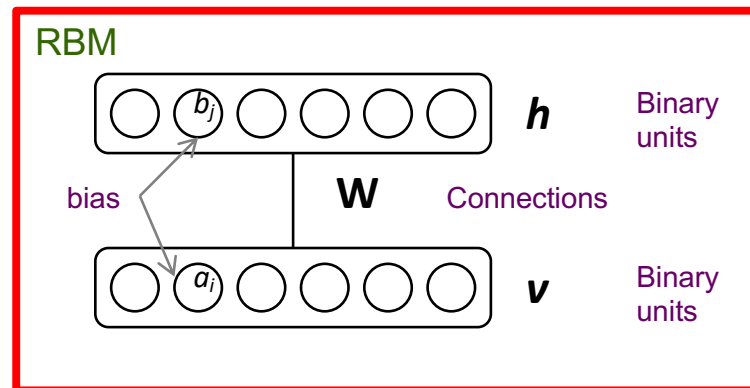
Intractable
Partition function

For an RBM: $\mathbf{x} = \{\mathbf{v}, \mathbf{h}\}$ $\boldsymbol{\theta} = \{\mathbf{W}, \mathbf{a}, \mathbf{b}\}$

Determine parameters $\boldsymbol{\theta}$ that maximize log-likelihood (negative loss)

$$\max_{\boldsymbol{\theta}} L(\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(M)}\}; \boldsymbol{\theta}) = \sum_m \log p(\mathbf{x}^{(m)}; \boldsymbol{\theta})$$

$$L(\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(M)}\}; \boldsymbol{\theta}) = \sum_m \log \tilde{p}(\mathbf{x}^{(m)}; \boldsymbol{\theta}) - \sum_m \log Z(\boldsymbol{\theta})$$



$$E(\mathbf{v}, \mathbf{h}) = -\mathbf{h}^T \mathbf{W} \mathbf{v} - \mathbf{a}^T \mathbf{v} - \mathbf{b}^T \mathbf{h} = \sum_i \sum_j W_{i,j} v_i h_j - \sum_i a_i v_i - \sum_j b_j h_j$$

$$p(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} \exp(-E(\mathbf{v}, \mathbf{h}))$$

$$\tilde{p}(\mathbf{x}) = \exp(-E(\mathbf{v}, \mathbf{h}))$$

$$Z = \sum_{\mathbf{v}, \mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h}))$$

For stochastic gradient ascent, take derivatives:

$$g_m = \nabla_{\boldsymbol{\theta}} \log p(\mathbf{x}^{(m)}; \boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}} \log \tilde{p}(\mathbf{x}^{(m)}; \boldsymbol{\theta}) - \nabla_{\boldsymbol{\theta}} \log Z(\boldsymbol{\theta}) \quad \boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \varepsilon g$$

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

Derivative of positive phase:

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

Summation is over samples
from the training set

Since it is summed m times $1/m$ has no effect

Derivative of negative phase:

$$\nabla_{\boldsymbol{\theta}} \log Z(\boldsymbol{\theta}) = \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \nabla_{\boldsymbol{\theta}} \log \tilde{p}(\mathbf{x})$$

An identity

$$\mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \nabla_{\boldsymbol{\theta}} \log \tilde{p}(\mathbf{x}) = \frac{1}{M} \sum_{i=1}^M \nabla_{\boldsymbol{\theta}} \log \tilde{p}(\mathbf{x}^{(m)}; \boldsymbol{\theta})$$

Summation is over samples from the RBM