Probabilistic Modelling and Reasoning, Tutorial Answer Sheet 5 (for Week 8)

School of Informatics, University of Edinburgh

Instructor: Amos Storkey
Prob. Sheet. ID: 27114.1

This tutorial is too long and you should pick and choose what to go through.
A Boltzmann Machine takes the form

$$P(\mathbf{x}|\mathbf{W},\mathbf{a},\mathbf{b}) = \frac{1}{Z}\exp\left(\frac{1}{2}\mathbf{x}^T\mathbf{W}\mathbf{x} + \mathbf{b}^T\mathbf{x}\right)$$

1. Show that the probability $P(x_i|x_{rest}) = 1/(1+\exp(\mathbf{W}\mathbf{x}+\mathbf{b}))$. Write a matlab function `samplebm(w,b,n)` to generate from a Boltzmann machine with weight matrix w, and b a column vector of biases. The function should return a $d*n$ matrix, where n is the number of samples. You should do the sampling by choosing a random initialisation and Gibbs sampling (can be done in 6 lines of clean Matlab code). This will help you understand the working of the Boltzmann machine, and the working of Gibbs sampling. If you do this you will probably remember both.

   Now download the mnist data from the tutorial website. Run the code on the PMR website. It will set a weight matrix directly (using a Hopfield Learning Rule rather than a Boltzmann Learning rule). This enables you to see the process of sampling from the Boltzmann Machine. What do you notice?

   The code is available at `http://www.inf.ed.ac.uk/teaching/courses/pmr/code/samplebm.m`. If you run this code you will notice that this form of Boltzmann machine has a number of different high probability regions (corresponding to different forms of characters). You will also notice that the Gibbs sampling procedure does not mix properly - it finds one local probability peak and samples in the region of that. You should discuss the different issues with Gibbs Sampling. Ask students: why can't we just use this process and start at multiple starts and then run Gibbs sampling from there to get the samples? Discuss the balance between the samples from the different modes of the Energy function.

2. Now build a Gibbs sampling process for the restricted Boltzmann Machine with 784 visible units and 784 hidden units, and implement a simple batch contrastive divergence rule. Run this on the mnist training data from the website. Then sample for the hidden units, and run another RBM on that. Keep track of all the weight matrices. Finally run a simple classifier on all the features and test your classification performance on the mnist test set. Alternatively if you are short of time, find read and try out code that does this... **This is really just a familiarisation exercise for students. You can discuss**

aspects around the practical aspects of deep learning. Things to talk about are different encoidng methods (RBM, noisy autoencoders), different learning procedures (contrastive divergence, contrastive divergence-n, persistive contrastive divergence, the use of minibatches and online methods), initialisation issues. The benefits of node sparsity (e.g. interpretability, modelling scenarios with sparse factorial components e.g. images).

3. Show that we can write the same model in the same form but where the binary variables have a $\{-1, 1\}$ representation instead. What is the transformation for the parameters in one form to the parameters in the other?

   To show this plug in for the $\mathbf{y} = 2\mathbf{x} - 1$ i.e. $\mathbf{x} = (\mathbf{y} + 1)/2$ to the quadratic form and rebuild the new quadratic form.

$$(\mathbf{y} + 1)(\mathbf{W}/4)(\mathbf{y} + 1) + \mathbf{b}(\mathbf{y} + 1)/2 = \mathbf{y}(\mathbf{W}/4)\mathbf{y} + (\mathbf{b}/2 + \mathbf{W}\mathbf{1}/2)\mathbf{y} + const$$

   The constant term just contributed to the normalisation constant. The rest is in the same form, but where the new weight matrix is quarter the old one, and the new bias is given as above.

4. Show that we can add a diagonal term to the $\mathbf{W}$ without affecting the model (hint - it is easiest to do this in the $\{-1, 1\}$ representation.

   $x_i^2 = 1$, and so the diagonal terms are constant wrt $\mathbf{x}$ and so just contribute to the normalisation.

5. Derive the learning rule for a restricted Boltzmann machine with zero biases where both $\mathbf{x}$ and $\mathbf{y}$ are visible variables. **Solution:** This can follow the general derivation of the general gradient learning rule for models given in the lecture.

$$P(\mathbf{x}, \mathbf{y}|\mathbf{W}) = \frac{1}{Z} \exp\left(\mathbf{x}^T \mathbf{W} \mathbf{y}\right) \tag{1}$$

   So for a whole dataset $D = \{(\mathbf{x}^n, \mathbf{y}^n)\}$ of length N

$$\log P(D|\mathbf{W}) = \sum_n \left((\mathbf{x}^n)^T \mathbf{W}(\mathbf{y}^n)\right) - N \log \sum_{\mathbf{x},\mathbf{y}} \exp\left(\mathbf{x}^T \mathbf{W} \mathbf{y}\right) \tag{2}$$

   where we have written $Z(\mathbf{W})$ out explicitly.

   Now taking derivatives wrt W

$$\frac{\partial}{\partial w_{ij}} \log P(D|\mathbf{W}) = \sum_n \left(x_i^n y_j^n\right) - N \frac{1}{Z} \left(\sum_{\mathbf{x},\mathbf{y}} x_i y_j \exp\left(\mathbf{x}^T \mathbf{W} \mathbf{y}\right) \right) \tag{3}$$

   Which can be rewritten

$$\frac{\partial}{\partial w_{ij}} \log P(D|\mathbf{W}) = \sum_n \left(x_i^n y_j^n\right) - N \sum_{\mathbf{x},\mathbf{y}} x_i y_j P(\mathbf{x}, \mathbf{y}) \tag{4}$$

Often this is written using the shorthand

$$\frac{\partial}{\partial w_{ij}} \log P(D|\mathbf{W}) \quad = \quad N(\langle x_i y_j \rangle_D \quad - \quad \langle x_i y_j \rangle_M) \tag{5}$$

where the first term means the empirical expectation wrt the data, and the second term means the actual expectation wrt to the model $P(\mathbf{x}, \mathbf{y})$.

This gives the derivatives. The BM learning rule is typically a gradient ascent procedure of the form

$$w_{ij} \to w_{ij} + \gamma(\langle x_i y_j \rangle_D - \langle x_i y_j \rangle_M)$$

which moves in the direction of steepest descent.

6. What is a restricted Boltzmann Machine with one hidden unit? (Draw and manipulate the graphical model). **It is a Naive Bayes Model: Draw the graphical model it is one hidden node with conditionally independent visibles.**

7. . Consider a variant of the restricted Boltzmann machine:

$$P(\mathbf{x}, \mathbf{y}) = \frac{1}{Z} \exp\left(-\frac{1}{2}\mathbf{x}^T\mathbf{x} + \mathbf{x}^T\mathbf{W}\mathbf{y} + \mathbf{b}^T\mathbf{y}\right) \tag{6}$$

where $\mathbf{x}$ is real valued and $\mathbf{y} = \pm 1$.

Write the content of the exponent in the form $-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}(\mathbf{y}))^T(\mathbf{x} - \boldsymbol{\mu}(\mathbf{y})) + F(\mathbf{y})$. What are $\boldsymbol{\mu}(\mathbf{y})$ and $F(\mathbf{y})$. Rewrite (6) using this exponent, separating out the terms with $\mathbf{x}$ in and the terms without. Notice that the terms with $\mathbf{x}$ in are Gaussian and integrate over them to get a constant. What are you left with? Compare this with a Boltzmann Machine.

This is an exercise in manipulation of integrals. This will be tough for most students.

$$P(\mathbf{x}, \mathbf{y}) = \frac{1}{Z} \exp\left(-\frac{1}{2}\mathbf{x}^T\mathbf{x} + \mathbf{x}^T\mathbf{W}\mathbf{y} + \mathbf{b}^T\mathbf{y}\right) \tag{7}$$

$$P(\mathbf{x}, \mathbf{y}) = \frac{1}{Z} \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{W}\mathbf{y})^T(\mathbf{x} - \mathbf{W}\mathbf{y}) + \frac{1}{2}\mathbf{y}^T\mathbf{W}\mathbf{W}^T\mathbf{y} + \mathbf{b}^T\mathbf{y}\right) \tag{8}$$

(note that $\mathbf{x}^T\mathbf{W}\mathbf{y} = \mathbf{y}^T\mathbf{W}^T\mathbf{x}$ as they are both scalars).

Now we can integrate the $\mathbf{x}$ out as it is just a Gaussian with an identity covariance. So

$$\int d\mathbf{x} \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{W}\mathbf{y})^T(\mathbf{x} - \mathbf{W}\mathbf{y})\right) = (2\pi)^{d/2}$$

which can just be absorbed into the normalisation constant.

That leaves

$$P(\mathbf{x}, \mathbf{y}) = \frac{1}{Z} \exp \frac{1}{2}(\mathbf{y}^T\mathbf{W}\mathbf{W}^T\mathbf{y} \ + \mathbf{b}^T\mathbf{y}) \tag{9}$$

which is just a Boltzmann Machine with weight matrix $\mathbf{W}^T\mathbf{W}$ and bias $\mathbf{b}$ (as we know we can assume a positive definite weight matrix for a General Boltzmann Machine as discussed in class).

8. Now rewrite (6) as

$$P(\mathbf{x}^*, \mathbf{y}) = \frac{1}{Z'} \exp\left(-\frac{1}{2}(\mathbf{x}^*)^T(\mathbf{W}^T\mathbf{W})^{-1}\mathbf{x}^*\right) \prod_i \exp\left(x_i^* y_i - b_i y_i\right) \tag{10}$$

where $\mathbf{x}^* = \mathbf{W}^T\mathbf{x}$. Marginalise over the y_i to get the induced distribution over the $\mathbf{x}^*$. This shows how we can represent the Boltzmann distribution in terms of real valued augmented variables.

To Marginalise over the y_i is simply use $\sum_{y_i} \exp\left(x_i^* y_i - b_i y_i\right) = \cosh\left(x_i^* - b_i\right)$ and plug in:

$$P(\mathbf{x}) = \frac{1}{Z'} \exp\left(-\frac{1}{2}(\mathbf{x}^*)^T(\mathbf{W}^T\mathbf{W})^{-1}\mathbf{x}^*\right) \prod_i \cosh\left(x_i^* - b_i\right) \tag{11}$$