Exercise 7 (Graded homework 3): 7th of May 2019
CS-526 Learning Theory

The deadline is Tuesday, May 21, 2019. Please hand in your homework during the lecture (May 20) or the exercise session (May 21). **No scan of handwritten homework is accepted**.

Problem 1: MCMC, Gibbs sampling, and application to the Ising model

Let $p(\underline{x})$, $\underline{x} = (x_1, \cdots, x_K) \in \mathcal{A}^K$, for a discrete alphabet $\mathcal{A}$, be the probability distribution (of some graphical model say) from which we want to sample. In class we discussed the general Metropolis-Hastings prescription: (i) Choose a *base or proposal chain* with transition probability $\widetilde{q}(\underline{x}' \mid \underline{x})$; (ii) If at time t the state of the Markov chain is $\underline{x}_t$ propose the move $\underline{x}_t \to \underline{x}_{t+1}$ where $\underline{x}_{t+1} \sim \widetilde{q}(\cdot \mid \underline{x}_t)$; (iii) Accept the new state with probability

$$A(\underline{x}_{t+1}, \underline{x}_t) = \min\left(1, \frac{\widetilde{q}(\underline{x}_t \mid \underline{x}_{t+1}) p(\underline{x}_{t+1})}{\widetilde{q}(\underline{x}_{t+1} \mid \underline{x}_t) p(\underline{x}_t)}\right).$$

Let $q(\underline{x}_{t+1} \mid \underline{x}_t)$ the transition probability of this chain.

1) Show that the detailed balance condition is satisfied, i.e.,

$$q(\underline{x}_{t+1} \mid \underline{x}_t) p(\underline{x}_t) = q(\underline{x}_t \mid \underline{x}_{t+1}) p(\underline{x}_{t+1})$$

and that therefore $p(\underline{x})$ is a stationary distribution.

2) Now consider the following base chain: select $i \in \{1, \cdots, K\}$ uniformly at random and do the move $\underline{x} \to \underline{x}'$ with probability $\widetilde{q}(\underline{x}' \mid \underline{x}) = p(x_i' \mid \{x_j\}_{j \neq i})$, if $\underline{x}'$ and $\underline{x}$ differ at most at coordinate i, and $\widetilde{q}(\underline{x}' \mid \underline{x}) = 0$ otherwise. This means that the new proposal state differs from the old state at most at one random selected coordinate.

   Show that the acceptance probability $A(\underline{x}', \underline{x}) = 1$.

3) The sampling method of the previous question is called Gibbs sampling (or heat bath dynamics or Glauber dynamics). Show that for the Ising model on an arbitrary graph with distribution:

$$p(\underline{s}) = \frac{1}{Z} \exp(\sum_{i,j \in E} J_{kl} s_k s_l + \sum_{k \in V} h_k s_k)$$

   where $(s_1, \cdots, s_K) \in \{-1, +1\}^K$, the Gibbs sampling algorithm reduces to the following simple rule:

   (i) At time t select $i \in \{1, \cdots, K\}$ uniformly at random.

(ii) Given the spin state $\underline{s}_t$, the new spin state at time $t+1$ is the same for all $j \neq i$ and has $s_{i,t+1} = \pm 1$ with probability

$$p(s_{i,t+1} = \pm 1 \mid (s_{j,t})_{j \in MB(i)}) = \frac{1}{2}(1 \pm \tanh(\sum_{j \in MB(i)} J_{ij} s_{j,t} + h_i))$$

where $MB(i) = \{j \in V \mid J_{ij} \neq 0\}$ is the Markov blanket of vertex i.

Problem 2: KL divergence (Barber 8.42)

Consider a "Boltzman machine" distribution on binary variables $x_i \in \{0,1\}, i = 1, \dots, D$

$$p(\mathbf{x}|\mathbf{W}) = \frac{1}{Z_p(\mathbf{W})} \exp(\mathbf{x}^T \mathbf{W} \mathbf{x})$$

We wish to fit p with another distribution q having the same form, i.e.,

$$q(\mathbf{x}|\mathbf{U}) = \frac{1}{Z_q(\mathbf{U})} \exp(\mathbf{x}^T \mathbf{U} \mathbf{x})$$

1) Show that

$$\arg\min_U \mathrm{KL}(p|q) = \arg\max_U \big\{ \mathrm{Tr}(\mathbf{UC}) - \log Z_q(\mathbf{U}) \big\},$$

where $C_{i,j} = \mathbb{E}_p[x_i x_j]$. Explain from there, that in theory at least, the second-moment matrix C is enough to fully specify p.

Problem 3: Naive Bayes classifier. Learning by counting. (Barber 10.4)

The Naive Bayes Classifier has a joint probability distribution over feature vectors $\underline{x} = (x_1, \cdots, x_K)$ and their *class* label of the form:

$$p(\underline{x}, class) = p(class) \prod_{i=1}^{K} p(x_i \mid class)\,.$$

This is a belief network with parent *class* and children $x_1, \cdots, x_K$. For binary attributes $x_i \in \{0,1\}$ and two classes 0, 1 it is parametrized by:

$$\theta_i^1 = p(x_i = 1|class = 1),\ \theta_i^0 = p(x_i = 1|class = 0),\ p_1 = p(class = 1),\ p_0 = p(class = 0)$$

Given a data set $(\underline{x}^{(n)}, c^{(n)})$, $n \in \{1, \cdots, N\}$, we can learn these parameters by counting. Once this is done we classify a new sample $\underline{x}^*$ in $class = 0$ if $p(\underline{x}^*|class = 0) > p(\underline{x}^*|class = 1)$ and classify it in $class = 1$ if $p(\underline{x}^*|class = 1) > p(\underline{x}^*|class = 0)$.

1) Show that the decision to classify a datapoint $\mathbf{x}^*$ as class 1 holds if $\mathbf{w}^T \mathbf{x}^* + b > 0$ for some $\mathbf{w}$ and b. State explicitly $\mathbf{w}$ and b as a functions of $\theta^1, \theta^0, p_1, p_0$.

Problem 4: Sigmoid Belief Network (Barber 11.7)

The sigmoid Belief Network is defined by the layered network

$$
\begin{aligned}
&p(\mathbf{x}^L)\prod_{l=1}^{L} p(\mathbf{x}^{l-1}|\mathbf{x}^l) \\
p(\mathbf{x}^{l-1}|\mathbf{x}^l) &= \prod_{i=1}^{w_l} p(x_i^{l-1}|\mathbf{x}^l) \\
p(x_i^{l-1} = 1|\mathbf{x}^l) &= \sigma(\mathbf{w}_{i,l}^T \mathbf{x}^l),
\end{aligned}
$$

where w_l is the width of layer l, $\mathbf{x}^l \in \{0,1\}^{w_l}$ and $\sigma(x) = 1/(1+e^{-x})$ is the sigmoid function. The top layer $p(\mathbf{x}^L)$ describes a factorized distribution $p(x_1^L)\dots p(x_{w_L}^L)$.

1) Draw the Belief Network structure of this distribution

2) For layer $\mathbf{x}^0$, what is the computational complexity of computing the likelihood $p(\mathbf{x}^0)$, assuming that all layers have equal width w?

3) We assume that we have data where $l = 0$ is the only visble layer, the other ones $l = 1, \cdots, L$ being hidden, and that furthermore we have only one data point $\mathbf{x}_0$. Assuming a fully factorized approximation for an equal width network (all $w_l = w$)

$$
p(\mathbf{x}^1, \dots, \mathbf{x}^L|\mathbf{x}^0) \approx \prod_{l=1}^{L}\prod_{i=1}^{w} q(x_i^l),
$$

write down the energy term of the Variational EM procedure (for a single data observation $\mathbf{x}^0$) and the complexity of its computation.

Problem 5: EM algorithm for mixtures of Gaussians

Consider a mixture of D-dimensional isotropic Gaussians defined by

$$
\begin{aligned}
p(\mathbf{x}) &= \sum_{i=1}^{H} p(\mathbf{x}|\mathbf{m}_i, \sigma_i^2)p(i) \\
p(\mathbf{x}|\mathbf{m}_i, \sigma_i^2) &= (2\pi\sigma_i^2)^{-D/2} \exp\Big(-\frac{1}{2\sigma_i^2}\|\mathbf{x} - \mathbf{m}_i\|^2\Big)
\end{aligned}
$$

1) Derive the optimal update rule for $\mathbf{m}_i$ and σ_i^2 in the M-step of EM algorithm.

Problem 6: On gradient ascent for RBM's

We want to learn the weight parameters $W_{ij} \in \mathbb{R}$ of a Restricted Boltzmann Machine with visible variables $(v_1, \cdots, v_K)$ and hidden (unobserved) variables $(h_1, \cdots, h_M)$. We assume that all variables are in a finite alphabet.

$$
p(\underline{v}, \underline{h} \mid W) = \frac{1}{Z}\exp\left(\sum_{i=1}^{K}\sum_{j=1}^{M} W_{ij} v_i h_j\right)
$$

Here we have assumed the bias terms $\sum_{i=1}^{K} b_i v_i + \sum_{j=1}^{M} c_i h_i$ are zero for simplicity, i.e., $b_i = c_i = 0$ (but the exercise can be generalized). We have a set of visible data points $v_1^{(1)}, \cdots, v_K^{(N)}$ with log-likelihood $L(W)$ (we assume that the data points are iid):

1) First, without assuming anything about the alphabet (apart that it is discrete and finite) show that

$$\frac{\partial}{\partial W_{ij}} L(W) = \sum_{n=1}^{N} \Big(\mathbb{E}_{p(h_j | \underline{v}_1^{(1)} \cdots \underline{v}_K^{(N)}, W)} [v_i^{(n)} h_j] - \langle v_i h_j \rangle \Big)$$

In the literature the first expectation is called the "clamped average". The second one is the average w.r.t $p(\underline{v}, \underline{h} \mid W)$ written with the "Gibbs bracket notation" $\langle - \rangle$. What is the Markov blanket of the pair of nodes i, j in $\langle v_i h_j \rangle$?

2) Now assuming that the hidden variables are binary in $\{-1, +1\}$ show that this reduces to

$$\frac{\partial}{\partial W_{ij}} L(W) = \sum_{n=1}^{N} \Big(v_i^{(n)} \tanh(\sum_{k=1}^{K} W_{kj} v_k^{(n)}) - \langle v_i h_j \rangle \Big)$$

3) *Note*: When performing gradient ascent the first term is easy to compute. The second one would typically be computed by MCMC but this is costly because in principle one has to run the chain for a long time. The "contrastive divergence" algorithm of Hinton uses a MCMC sampling method where the chain is cut after very few (in practice one or two) time steps to obtain a stochastic estimator of the gradient.