CSE 250a. Assignment 5

Out: *Tue Nov 03* **Due:** *Tue Nov 10*

5.1 EM algorithm

For this problem, consider the belief network shown below, with observed nodes B and D and hidden nodes A and C.



(a) Posterior probability

Compute the posterior probability P(a, c|b, d) in terms of the conditional probability tables (CPTs) of the belief network, namely P(a), P(b|a), P(c|a, b) and P(d|b, c).

(b) Posterior probability

Compute the posterior probabilities P(a|b,d) and P(c|b,d) in terms of your answer from part (a).

(c) Log-likelihood

Consider a partially complete data set of *i.i.d.* examples $\{b_t, d_t\}_{t=1}^T$ drawn from the joint distribution of the above belief network. The log-likelihood of the data set is given by:

$$\mathcal{L} = \sum_{t} \log P(B = b_t, D = d_t).$$

Compute this log-likelihood in terms of the CPTs of the belief network.

(d) EM algorithm

Derive the EM updates for the CPTs of this belief network. Simplify your answers as much as possible, expressing them in terms of the posterior probabilities $P(a, c|b_t, d_t)$, $P(a|b_t, d_t)$, and $P(c|b_t, d_t)$, as well as the binary-valued functions $I(b, b_t)$, and $I(d, d_t)$, where

$$I(x, x') = \begin{cases} 1 & \text{if } x = x', \\ 0 & \text{otherwise.} \end{cases}$$

5.2 EM algorithm for noisy-OR

Consider the belief network on the right, with binary random variables $X \in \{0, 1\}^n$ and $Y \in \{0, 1\}$ and a noisy-OR conditional probability table (CPT). The noisy-OR CPT is given by:

$$P(Y = 1|X) = 1 - \prod_{i=1}^{n} (1 - p_i)^{X_i},$$

which is expressed in terms of the noisy-OR parameters $p_i \in [0, 1]$.

In this problem, you will derive and implement an EM algorithm for estimating the noisy-OR parameters p_i . It may seem that the EM algorithm is not suited to this problem, in which all the nodes are observed, and the CPT has a parameterized form. In fact, the EM algorithm can be applied, but first we must express the model in a different but equivalent form.

Consider the belief network shown to the right. In this network, a binary random variable $Z_i \in \{0, 1\}$ intercedes between each pair of nodes X_i and Y. Suppose that:

$$P(Z_i = 1 | X_i = 0) = 0,$$

$$P(Z_i = 1 | X_i = 1) = p_i.$$

Also, let the node Y be *determined* by the logical-OR of Z_i . In other words:

$$P(Y=1|Z) = \begin{cases} 1 \text{ if } Z_i = 1 \text{ for any } i, \\ 0 \text{ if } Z_i = 0 \text{ for all } i. \end{cases}$$

.



(a) Show that this "extended" belief network defines the same conditional distribution P(Y|X) as the original one. In particular, starting from

$$P(Y=1|X) = \sum_{Z \in \{0,1\}^n} P(Y=1,Z|X),$$

show that the right hand side of this equation reduces to the noisy-OR CPT with parameters p_i . To perform this marginalization, you will need to exploit various conditional independence relations.

(b) Consider estimating the noisy-OR parameters p_i to maximize the (conditional) likelihood of the observed data. The (normalized) log-likelihood in this case is given by:

$$\mathcal{L} = \frac{1}{T} \sum_{t=1}^{T} \log P(Y = y^{(t)} | X = \vec{x}^{(t)}),$$

where $(\vec{x}^{(t)}, y^{(t)})$ is the *t*th joint observation of X and Y, and where for convenience we have divided the overall log-likelihood by the number of examples T. From your result in part (a), it follows that we can estimate the parameters p_i in either the original network or the extended one (since in both networks they would be maximizing the same equation for the log-likelihood).

Notice that in the extended network, we can view X and Y as observed nodes and Z as hidden nodes. Thus in this network, we can use the EM algorithm to estimate each parameter p_i , which simply defines one row of the "look-up" CPT for the node Z_i .

Compute the posterior probability that appears in the E-step of this EM algorithm. In particular, for joint observations $x \in \{0, 1\}^n$ and $y \in \{0, 1\}$, use Bayes rule to show that:

$$P(Z_i=1, X_i=1|X=x, Y=y) = \frac{yx_ip_i}{1-\prod_j(1-p_j)^{x_j}}$$

(c) For the data set $\{\vec{x}^{(t)}, y^{(t)}\}_{t=1}^T$, show that the EM update for the parameters p_i is given by:

$$p_i \leftarrow \frac{1}{T_i} \sum_t P\left(Z_i = 1, X_i = 1 | X = x^{(t)}, Y = y^{(t)}\right),$$

where T_i is the number of examples in which $X_i = 1$. (You should derive this update as a special case of the general form presented in lecture.)

(d) Download the ASCII files hw5_X.txt and hw5_Y.txt from the course web site. These files contain T = 20000 examples of (x, y) pairs (with n = 10). Use the EM algorithm to estimate the parameters p_i that maximize the log-likelihood of this data set.

To check your solution, initialize all $p_i = 0.5$ and perform 64 iterations of the EM algorithm. At each iteration, compute the log-likelihood shown in part (b). *Turn in a completed version of this table:*

#	0	1	2	4	8	16	32	64
\mathcal{L}	-2.5709	-0.6938	?	?	?	?	?	-0.5363

If you have implemented the EM algorithm correctly, this log-likelihood will always increase from one iteration to the next. You may use the already completed entries of this table to check your work.

(e) Turn in your source code. You may program in the language of your choice.

5.3 Auxiliary function

In class we derived an auxiliary function for optimizing the log-likelihood in belief networks with hidden variables. In this problem you will derive an auxiliary function for optimizing a simpler function that is nearly quadratic near its minimum, but nearly linear far away from its minimum.

- (a) Consider the function $f(x) = \log \cosh(x)$. Show that the minimum occurs at x = 0.
- (b) Show that $f''(x) \leq 1$ for all x.
- (c) Consider the function $Q(x, y) = f(y) + f'(y)(x y) + \frac{1}{2}(x y)^2$. Plot f(x), Q(x, -2), and Q(x, 3) as a function of x.
- (d) Prove that Q(x, y) is an auxiliary function for f(x). In particular, show that it satisfies:
 (i) Q(x, x) = f(x)
 (ii) Q(x, y) ≥ f(x)
- (e) Derive the form of the update rule $x_{n+1} = \operatorname{argmin}_x Q(x, x_n)$.
- (f) Write a simple program to show that your update rule in (e) converges numerically for the initial guesses $x_0 = -2$ and $x_0 = 3$. Turn in your source code as well as plots of x_n versus n.
- (g) Repeat parts (e) and (f) using the update rule for Newton's method: namely, $x_{n+1} = x_n f'(x_n)/f''(x_n)$. What happens and why? Determine an upper bound on $|x_0|$ so that Newton's method converges. (*Hint:* require $|x_1| < |x_0|$.)
- (h) Plot the function $g(x) = \frac{1}{10} \sum_{k=1}^{10} \log \cosh \left(x + \frac{1}{\sqrt{k}} \right)$. Is it still simple to find the exact minimum?
- (i) Consider the function $R(x, y) = g(y) + g'(y)(x-y) + \frac{1}{2}(x-y)^2$. Prove that R(x, y) is an auxiliary function for g(x).
- (j) Derive the form of the update rule $x_{n+1} = \operatorname{argmin}_{x} R(x, x_n)$.
- (k) Use the update rule from part (j) to locate the minimum of g(x) to four significant digits. In addition to your answer for the minimum, **turn in your source code as well as plots of** x_n **versus** n.