CS446: Machine Learning **SPRING 2017**

Problem Set 7

Handed Out: April 20^{th}, 2017 *Due: May* 1^{st}, 2017

- Feel free to talk to other members of the class in doing the homework. I am more concerned that you learn how to solve the problem than that you demonstrate that you solved it entirely on your own. You should, however, write down your solution yourself. Please try to keep the solution brief and clear.
- Please use Piazza first if you have questions about the homework. Also feel free to send us e-mails and come to office hours.
- Please, no handwritten solutions. You will submit your solution manuscript as a single pdf file.
- The homework is due at **11:59 PM** on the due date. We will be using Compass for collecting the homework assignments. Please submit an electronic copy via Compass2g (`http://compass2g.illinois.edu`). Please do NOT hand in a hard copy of your write-up. Contact the TAs if you are face technical difficulties in submitting the assignment.
- You cannot use the late submission credit hours for this problem set.
- No code is needed for any of these problems. You can do the calculations however you please. You need to turn in only the report. Please name your report as ⟨`NetID`⟩`-hw7.pdf`.

1. **[EM Algorithm - 70 points]**

   Assume we have a set D of m data points, where for each data point x from D, $x \in \{0,1\}^{n+1}$. Denote the i-th bit of the j-th example as $x_i^{(j)}$. Thus, the index i ranges from $0 \dots n$, and the index j ranges from $1 \dots m$.

   Assume these data points were generated according to the following distribution:

   Postulate a hidden random variable Z with values $z = 1, 2$, where the probability of $z = 1$ is α and the probability of $z = 2$ is $1 - \alpha$, where $0 < \alpha < 1$.

   For a specific example $x^{(j)}$, a random value of Z is chosen, but its true value z is hidden. Note that each example $x^{(j)}$ has a fixed underlying z. If $z = 1$, $x_i^{(j)}$ is set to 1 with probability p_i. If $z = 2$, the bit is set to 1 with probability q_i. Thus, there are $2n + 3$ unknown parameters. You will use EM to develop an algorithm to estimate these unknown parameters.

   (a) **[10 points]** Express $\Pr(x^{(j)})$ first in terms of conditional probabilities and then in terms of the unknown parameters α, p_i, and q_i.
   Using the total probability rule and the fact that the x_i s are conditionally indepenedeny given Z:

$$\begin{aligned}
\Pr(x^{(j)}) &= \Pr(x^{(j)}|Z^{(j)}=1)\Pr(Z^{(j)}=1) + \Pr(x^{(j)}|Z^{(j)}=0)\Pr(Z^{(j)}=0) \\
&= \Pr(Z^{(j)}=1)\prod_{i=0}^{n}\Pr(x_i^{(j)}|Z^{(j)}=1) + \Pr(Z^{(j)}=0)\prod_{i=0}^{n}\Pr(x_i^{(j)}|Z^{(j)}=0) \\
&= \alpha\prod_{i=0}^{n}\left(p_i^{x_i^{(j)}}(1-p_i)^{1-x_i^{(j)}}\right) + (1-\alpha)\prod_{i=0}^{n}\left(q_i^{x_i^{(j)}}(1-q_i)^{1-x_i^{(j)}}\right)
\end{aligned}$$

(b) **[10 points]** Let $f_z^{(j)} = \Pr(Z = z \mid x^{(j)})$, i.e. the probability that the data point $x^{(j)}$ has z as the value of its hidden variable Z. Express $f_1^{(j)}$ and $f_2^{(j)}$ in terms of the unknown parameters.

Using Bayes rule, we have

$$\begin{aligned} f_1^{(j)} &= \Pr(Z^{(j)} = 1 | x^{(j)}) \\ &= \frac{\Pr(x^{(j)} | Z^{(j)} = 1) \Pr(Z^{(j)} = 1)}{\Pr(x^{(j)})} \\ &= \frac{\alpha \prod_{i=0}^{n} \left(p_i^{x_i^{(j)}} (1-p_i)^{1-x_i^{(j)}} \right)}{\alpha \prod_{i=0}^{n} \left(p_i^{x_i^{(j)}} (1-p_i)^{1-x_i^{(j)}} \right) + (1-\alpha) \prod_{i=0}^{n} \left(q_i^{x_i^{(j)}} (1-q_i)^{1-x_i^{(j)}} \right)} \end{aligned}$$

Similarly,

$$\begin{aligned} f_2^{(j)} &= \Pr(Z^{(j)} = 2 | x^{(j)}) \\ &= \frac{\Pr(x^{(j)} | Z^{(j)} = 2) \Pr(Z^{(j)} = 2)}{\Pr(x^{(j)})} \\ &= \frac{(1-\alpha) \prod_{i=0}^{n} \left(q_i^{x_i^{(j)}} (1-q_i)^{1-x_i^{(j)}} \right)}{\alpha \prod_{i=0}^{n} \left(p_i^{x_i^{(j)}} (1-p_i)^{1-x_i^{(j)}} \right) + (1-\alpha) \prod_{i=0}^{n} \left(q_i^{x_i^{(j)}} (1-q_i)^{1-x_i^{(j)}} \right)} \end{aligned}$$

(c) **[10 points]** Derive an expression for the expected log likelihood ($E[LL]$) of the entire data set D and its associated z settings given new parameter estimates $\tilde{\alpha}, \tilde{p}_i, \tilde{q}_i$.

Since the x^j are independent, the expected log likelihood is:

$$\begin{aligned} E[LL] &= E\left[\sum_{j=1}^{m} \ln(\Pr(x^{(j)}, Z^{(j)} | \tilde{\alpha}, \tilde{p}_i, \tilde{q}_i)) \right] \\ &= \sum_{j=1}^{m} E\left[\ln(\Pr(x^{(j)}, Z^{(j)} | \tilde{\alpha}, \tilde{p}_i, \tilde{q}_i)) \right] \\ &= \sum_{j=1}^{m} f_1^{(j)} \ln\left(\tilde{\alpha} \prod_{i=0}^{n} \tilde{p}_i^{x_i^{(j)}} (1-\tilde{p}_i)^{1-x_i^{(j)}} \right) + f_2^{(j)} \ln\left((1-\tilde{\alpha}) \prod_{i=0}^{n} \tilde{q}_i^{x_i^{(j)}} (1-\tilde{q}_i)^{1-x_i^{(j)}} \right) \\ &= \sum_{j=1}^{m} f_1^{(j)} \left(\ln \tilde{\alpha} + \sum_{i=0}^{n} (x_i^{(j)} \ln \tilde{p}_i + (1-x_i^{(j)}) \ln(1-\tilde{p}_i)) \right) + \\ &\qquad f_2^{(j)} \left(\ln(1-\tilde{\alpha}) + \sum_{i=0}^{n} (x_i^{(j)} \ln \tilde{q}_i + (1-x_i^{(j)}) \ln(1-\tilde{q}_i)) \right) \end{aligned}$$

(d) **[10 points]** Maximize the log likelihood (LL) and determine the update rules for the parameters according to the EM algorithm.

To maximize the log likelihood, we set equal to 0 the partial derivatives with respect to the parameters. Note that $f_1^{(j)} = 1 - f_2^{(j)}$.

$$\frac{\delta E}{\delta \tilde{\alpha}} = \sum_{j=1}^{m} \frac{f_1^{(j)}}{\tilde{\alpha}} - \frac{f_2^{(j)}}{1-\tilde{\alpha}} = 0 \Rightarrow \tilde{\alpha} = \frac{1}{m}\sum_{j=1}^{m} f_1^{(j)}$$

$$\frac{\delta E}{\delta \tilde{p}_i} = \sum_{j=1}^{m} f_1^{(j)} \left(\frac{x_i^{(j)}}{\tilde{p}_i} - \frac{1-x_i^{(j)}}{1-\tilde{p}_i} \right) = 0 \Rightarrow \tilde{p}_i = \frac{\sum_{j=1}^{m} x_i^{(j)} f_1^{(j)}}{\sum_{j=1}^{m} f_1^{(j)}}$$

$$\frac{\delta E}{\delta \tilde{q}_i} = \sum_{j=1}^{m} f_2^{(j)} \left(\frac{x_i^{(j)}}{\tilde{q}_i} - \frac{1-x_i^{(j)}}{1-\tilde{q}_i} \right) = 0 \Rightarrow \tilde{q}_i = \frac{\sum_{j=1}^{m} x_i^{(j)} f_2^{(j)}}{\sum_{j=1}^{m} f_2^{(j)}}$$

(e) **[10 points]** Examine the update rules explain them in English. Describe in pseudocode how you would run the algorithm: initialization, iteration, termination. What equations would you use at which steps in the algorithm?

From the results in (d), the update rule indicates that the best estimate for $\tilde{\alpha}$ is the average of the $f_1^{(j)}$ over all data, and the best estimates for the p_i and q_i are weighted averages of $x_i^{(j)}$ by $f_1^{(j)}$ and $f_2^{(j)}$ respectively.

To run the algorithm:

i. Initialize with random values for α, p_i, and q_i for all i.
ii. Calculate $f_1^{(j)}$ and $f_2^{(j)}$ as shown in (b).
iii. Find the new values for α, p_i, and q_i using the update rules derived in (d).
iv. Repeat (ii) and (iii) until convergence.

(f) **[10 points]** Assume that your task is to predict the value of x_0 given an assignment to the other n variables and that you have the parameters of the model. Show how to use these parameters to predict x_0. (*Hint:* Consider the ratio between $P(X_0 = 0)$ and $P(X_0 = 1)$.)

When X_0 is unobserved, the conditional odds that $X_0 = 1$ can be written as:

$$O(X_0 = 1 | X_1 = x_1, ..., X_n = x_n) \equiv$$

$$\frac{\Pr(X_0 = 1, X_1 = x_1, ..., X_n = x_n)}{\Pr(X_0 = 0, X_1 = x_1, ..., X_n = x_n)}$$

$$= \frac{\Pr(Z=1)\Pr(1, x_1, ..., x_n | Z = 1) + \Pr(Z=2)\Pr(1, x_1, ..., x_n | Z = 2)}{\Pr(Z=1)\Pr(0, x_1, ..., x_n | Z = 1) + \Pr(Z=2)\Pr(0, x_1, ..., x_n | Z = 2)}$$

$$= \frac{\Pr(Z=1)p_0 \Pr(x_1, ..., x_n | Z = 1) + \Pr(Z=2)q_0 \Pr(x_1, ..., x_n | Z = 2)}{\Pr(Z=1)(1-p_0)\Pr(x_1, ..., x_n | Z = 1) + \Pr(Z=2)(1-q_0)\Pr(x_1, ..., x_n | Z = 2)}$$

$$= \frac{p_0 L_1 + q_0 L_2}{(1-p_0)L_1 + (1-q_0)L_2}$$

where

$$L_1 = \alpha \prod_{i=1}^{n} p_i^{x_i}(1-p_i)^{1-x_i}$$

and

$$L_2 = (1-\alpha) \prod_{i=1}^{n} q_i^{x_i}(1-q_i)^{1-x_i}$$

Thus, we can predict the value of x_0 by checking the condition:

$$O(X_0 = 1 | X_1 = x_1, ..., X_n = x_n) > 1$$

or

$$p_0 L_1 + q_0 L_2 > (1-p_0)L_1 + (1-q_0)L_2$$

Note that if both $p_0 > \frac{1}{2}$ and $q_0 > \frac{1}{2}$, the rule is linear but degenerate—we always predict 1 (and 0 respectively if they are both below $\frac{1}{2}$).

(g) **[10 points]** Show that the decision surface for this prediction is a linear function of the x_i's.

Continuing from (f), the hypothesis for the value of X_0 can be written as:

$$(2p_0 - 1)L_1 > (1-2q_0)L_2$$

To show that this decision has a linear surface, it suffices to show that we can write it as a sum like

$$\sum_{i=1}^{n} w_i x_i > \theta$$

Since we have a product involving x_i and not a sum, we start by applying *log* to both sides of the inequality:

$$\begin{aligned}
&\log(2p_0-1) + \log\left(\alpha \textstyle\prod_{i=1}^{n} p_i^{x_i}(1-p_i)^{1-x_i}\right) > \log(1-2q_0) + \log\left((1-\alpha)\textstyle\prod_{i=1}^{n} q_i^{x_i}(1-q_i)^{1-x_i}\right) \\
\implies & c_0 + \textstyle\sum_{i=1}^{n}[x_i \log(p_i) + (1-x_i)\log(1-p_i)] > c_1 + \sum_{i=1}^{n}[x_i \log(q_i) + (1-x_i)\log(1-q_i)] \\
\implies & c_0 + \textstyle\sum_{i=1}^{n}\left[\log\left(\frac{p_i}{1-p_i}\right)x_i + \log(1-p_i)\right] > c_1 + \sum_{i=1}^{n}\left[\log\left(\frac{q_i}{1-q_i}\right)x_i + \log(1-q_i)\right] \\
\implies & c_0 + \textstyle\sum_{i=1}^{n}[x_i \log(p_i) + (1-x_i)\log(1-p_i)] > c_1 + \sum_{i=1}^{n}[x_i \log(q_i) + (1-x_i)\log(1-q_i)] \\
\implies & c_0 + \textstyle\sum_{i=1}^{n}\left[\log\left(\frac{p_i}{1-p_i}\right)x_i + \log(1-p_i)\right] > c_1 + \sum_{i=1}^{n}\left[\log\left(\frac{q_i}{1-q_i}\right)x_i + \log(1-q_i)\right] \\
\implies & c_2 + \textstyle\sum_{i=1}^{n}\log\left(\frac{p_i}{1-p_i}\right)x_i > c_3 + \sum_{i=1}^{n}\log\left(\frac{q_i}{1-q_i}\right)x_i \\
\implies & \textstyle\sum_{i=1}^{n}\log\left(\frac{p_i(1-q_i)}{q_i(1-p_i)}\right)x_i > c_3 - c_2
\end{aligned}$$

Along the way, we have substituted c_j constants for terms that do not depend on any x_i. Note once again that if both $p_0 > \frac{1}{2}$ and $q_0 > \frac{1}{2}$, the rule is linear but degenerate—we always predict 1 (and 0 respectively if they are both below $\frac{1}{2}$).

2. [**Tree Dependent Distributions - 30 points**]

A tree dependent distribution is a probability distribution over n variables, $\{x_1, \ldots, x_n\}$ that can be represented as a tree built over n nodes corresponding to the variables. If there is a directed edge from variable x_i to variable x_j, then x_i is said to be the parent of x_j. Each directed edge $\langle x_i, x_j \rangle$ has a weight that indicates the conditional probability $\Pr(x_j \mid x_i)$. In addition, we also have probability $\Pr(x_r)$ associated with the root node x_r. While computing joint probabilities over tree-dependent distributions, we assume that a node is independent of all its non-descendants given its parent. For instance, in our example above, x_j is independent of all its non-descendants given x_i.

To learn a tree-dependent distribution, we need to learn three things: the structure of the tree, the conditional probabilities on the edges of the tree, and the probabilities on the nodes. Assume that you have an algorithm to learn an *undirected* tree T with all required probabilities. To clarify, for all *undirected* edges $\langle x_i, x_j \rangle$, we have learned both probabilities, $\Pr(x_i \mid x_j)$ and $\Pr(x_j \mid x_i)$. (There exists such an algorithm and we will be covering that in class.) The only aspect missing is the directionality of edges to convert this undirected tree to a directed one.

However, it is okay to not learn the directionality of the edges explicitly. In this problem, you will show that choosing any arbitrary node as the root and directing all edges away from it is sufficient, and that two directed trees obtained this way from the same underlying undirected tree T are equivalent.

(a) [**10 points**] State exactly what is meant by the statement: "*The two directed trees obtained from T are equivalent.*"

Two directed trees T_0 and T_1 over variables $x_1, x_@, \ldots, x_n$ are equivalent iff the joint probability distributions they represent are the same. In other words:

$$Pr_{T_0}(x_1, x_2, \ldots, x_n) = Pr_{T_1}(x_1, x_2, \ldots, x_n)$$

This implies that for every event E over $x_1, x_2, \ldots, x_n$, $\Pr_{T_0}(E) = \Pr_{T_1}(E)$.

(b) [**20 points**] Show that no matter which node in T is chosen as the root for the "direction" stage, the resulting directed trees are all equivalent (based on your definition above).

Let T_i and T_j be the two directed trees obtained by choosing two different roots x_i and x_j ($i \neq j, 1 \leq i, j \leq n$) from the undirected tree T. Denoting $\mathrm{x} = (x_1, \ldots, x_n)$, we would like to show that $\Pr_{T_i}(\mathrm{x}) = \Pr_{T_j}(\mathrm{x})$.

Let π_{x_k} be the parent of node x_k. Note that there is a unique path $\mathcal{P}$ between nodes i and j. Assume for now that the path is of length 1. That is, there is an edge in T between x_i and x_j. Thus, the only difference between T_i and T_j is the direction of this edge (convince yourself of that).

$$
\begin{aligned}
Pr_{T_i}(\mathrm{x}) &= \Pr(x_i) \prod_{\substack{k=1 \\ k \neq i}}^{n} \Pr(x_k | \pi_{x_k}) \\
&= \Pr(x_i) \Pr(x_j | x_i) \prod_{\substack{k=1 \\ x_k \notin \mathcal{P}}}^{n} \Pr(x_k | \pi_{x_k}) \\
&= \Pr(x_i, x_j) \prod_{\substack{k=1 \\ x_k \notin \mathcal{P}}}^{n} \Pr(x_k | \pi_{x_k}) \\
&= \Pr(x_j) \Pr(x_i | x_j) \prod_{\substack{k=1 \\ k_{x_k \notin \mathcal{P}}}}^{n} \Pr(x_k | \pi_{x_k}) \\
&= \Pr(x_j) \prod_{\substack{k=1 \\ k \neq j}}^{n} \Pr(x_k | \pi_{x_k}) \\
&= Pr_{T_j}(\mathrm{x})
\end{aligned}
$$

Next, notice that is the path $\mathcal{P}$ between x_i and x_j is longer, we maintain the property that the esges not on the path $\mathcal{P}$ have the same directionality in T_i and T_j. We can therefore use the argument above inductively, transforming a tree rooted at x_i to one rooted at x_j by switching the directions of the edges in $\mathcal{P}$ one edge at a time. As shown above, each of these steps amintains the equivalent joint distribution.