CS446: Machine Learning **Fall 2014**

Problem Set 7

Nikhil Unni *Handed In: December 4, 2014*

1. EM Algorithm

a.

$$P(w_j, d_i) = P(d_i)P(w_j|d_i)$$
$$= P(d_i)\sum_{k=1}^{2} P(c_k|d_i)P(w_j|c_k)$$

b.

$$P(c_k|w_j, d_i) = \frac{P(c_k, w_j, d_i)}{P(w_j, d_i)}$$
$$= \frac{P(d_i)P(c_k|d_i)P(w_j|c_k)}{P(d_i)\sum_{k=1}^{2} P(c_k|d_i)P(w_j|c_k)}$$
$$= \frac{P(c_k|d_i)P(w_j|c_k)}{\sum_{k=1}^{2} P(c_k|d_i)P(w_j|c_k)}$$

c. Likelihood of entire data is given by:

$$L = \prod_i \prod_j P(d_i, w_j)^{n(d_i, w_j)}$$

So then log-likelihood is:

$$LL = \sum_i \sum_j n(d_i, w_j) log[P(d_i, w_j)]$$

And then expected value of the log-likelihood with respect to the posterior:

$$E[LL] = \sum_i \sum_j n(d_i, w_j) E[log[\sum_k P(d_i)P(c_k|d_i)P(w_j|c_k)]]$$

$$= \sum_i \sum_j n(d_i, w_j)[P(c_1|w_j, d_i) log[P(w_j|c_1)P(c_1|d_i)] + P(c_2|w_j, d_i) log[P(w_j|c_2)P(c_2|d_i)]]$$

d. Using lagrange multipliers, with the optimizing function as:

$$f : E[LL] = \sum_i \sum_j n(d_i, w_j)[P(c_1|w_j, d_i) log[P(w_j|c_1)P(c_1|d_i)] + P(c_2|w_j, d_i) log[P(w_j|c_2)P(c_2|d_i)]]$$

With constraints:

$$g_1 : \sum P(c_k|d_i) = 1$$
$$g_2 : \sum P(w_j|c_k) = 1$$

$$g_3 : \sum P(d_i) = 1$$

If we combine them all with the constraints, after a bit of messy math, we get the equations:

$$P(w_j|c_k) = \frac{\sum_{i=1}^{M} n(d_i, w_j)P(c_k|d_i, w_j)}{\sum_i \sum_{j2=1}^{V} n(d_i, w_{j2})P(c_k|d_i, w_{j2})}$$

$$P(c_k|d_i) = \frac{\sum_{j=1}^{V} n(d_i, w_j)P(c_k|d_i, w_j)}{\sum_{j=1}^{V} n(d_i, w_j)}$$

And then, estimate $P(d_i)$ with just $P(d_i) = \frac{1}{M}$.

e. $P(w_j|c_k)$ is just given by iterating through all the documents, and counting how many times w_j has appeared with category c_k and dividing it by the total number of appearances of c_k in all of the documents.
$P(c_k|d_i)$ is just given by iterating through the given document d_i, and counting how many times the category c_k appeared, and dividing it by the total number of words in the document.
And then $P(d_i)$ is just the likelihood of choosing a specific document, and with no information about that, the likelihood is equal among all documents.

1. Make initial guess of our parameters, $P(d_i)$, $P(c_k|d_i)$, $P(w_j|c_k)$
2. While not converged:
3. Find log-likelihood of the data given initial guess of our parameters using our equation
4. Find posterior of the latent variable, using our equation from part b.
5. Calculate the expected value of the log-likelihood, with respect to the posterior, using
6. Maximize the expected value, using the techniques from part d.
7. Set our parameters to the new argmax values.
8. Return our final parameters.

2. Tree Dependent Distributions

a. It merely means that the choice of our root node is irrelevant for the final joint probability distribution of the tree. Explicitly, it means:
For all choices of root nodes, x_r, in the graph, all $P(x_r) \prod_{x_i \in T - \{x_r\}} P(x_i)P(x_i|parent(x_i))$ are equal.

b. Because the Chow-Liu Algorithm uses the function $I(x, y) = \sum_{x,y} P(x, y) log \frac{P(x,y)}{P(x)P(y)}$ as the metric for the tree generation, we see that if we were to use either $P(x_i|x_j)$ or $P(x_j|x_i)$, it wouldn't matter, since I is symmetric for both. And since we know that either one of the two (for all i and j) will be included in the calculation, the resulting joint probability distribution is the same, no matter where the starting root is.