MACHINE LEARNING COMS 4771, HOMEWORK 4
Assigned November 6, 2014. Due November 20, 2014 before 1.00pm.

1 Problem 1 (10 points): EM Derivation

Consider a random variable x that is categorical with M possible values $1, \ldots, M$. Suppose x is represented as a vector such that $x(j) = 1$ if x takes the j^{th} value, and $\sum_{j=1}^{M} x(j) = 1$. The distribution of x is described by a mixture of K discrete multinomial distributions such that:

$$p(x) = \sum_{k=1}^{K} \pi_k p(x|\mu_k)$$

where

$$p(x|\mu_k) = \prod_{j=1}^{M} \mu_k(j)^{x(j)}$$

where π_k denotes the mixing coefficients for the k^{th} component (aka the prior probability that the hidden variable $z = k$), and μ_k specifies the parameters of the k^{th} component. Specifically, $\mu_k(j)$ represents the probability $p(x(j) = 1|z = k)$ (and, therefore, $\sum_j \mu_k(j) = 1$). Given an observed data set $\{x_n\}$, $n = 1, \cdots, N$, derive the E and M step equations of the EM algorithm for optimizing the mixing coefficients and the component parameters $\mu_k(j)$ for this distribution. For your reference, here is the generic formula for the E and M steps. Note that θ is used to denote all parameters of the mixture model.

E-step. For each n, calculate $\tau_{nj} = p(z_n = j|x_n, \theta)$, i.e., the probability that observation i belongs to each of the K clusters.

M-step. Set

$$\theta := \arg\max_{\theta} \sum_{n=1}^{N} \sum_{j=1}^{K} \tau_{nj} \log \frac{p(x_n, z_n = j|\theta)}{\tau_{nj}}.$$

2 Problem 2 (30 points): EM for Multinomial Mixtures

Start by downloading the implementation of the Expectation-Maximization algorithm for Gaussian mixture-models for clustering d-dimensional vector data using a mixture of M multivariate Gaussian models. The code is available form the tutorials link as mixmodel.m. You will also need the four .m files below it. This includes randInit.m to initialize the parameters randomly and the functions plotClust.m and plotGauss.m to show the Gaussians overlayed on a plot of the first two dimensions of the data sets after EM converges. Try this code out on datasetA and datasetB by showing a good fit of these two data-sets with 3 Gaussians.

Next implement a new EM algorithm for clustering multinomial models rather than Gaussians. Test this code on the dataset ShakespeareMiddleton.data. Just type 'load ShakespeareMiddleton.data' in Matlab to load it. This dataset is a matrix of 18 columns and 10025 rows. The data are the word counts from 9 plays are by William Shakespeare and 9 plays by Thomas Middleton. The first 9 columns correspond to the Shakespeare plays Antony, Coriolanus, Hamlet, Julius, Lear, Othello, Romeo, Timon, and Titus. The last 9 columns correspond to the Middleton plays: Cheapside, Gallants, Maiden, Nowit, Phoenix, Puritan, Revenger, Trick, and Witch. The texts were downloaded (and parsed to undo capitalization, punctuation, etc.) from

```
http://www.tech.org/~cleary/middhome.html
http://the-tech.mit.edu/Shakespeare/
```

The words (from a dicitonary of 10025 words total covering all the above 18 plays) for each row are shown in the first column of ShakespeareMiddleton.txt if you are curious what the word counts correspond to. We also added a 1 to the counts so words that never appear in a document get 1, words that appear once get a count of 2, and so forth.

The work for this question will be to produce code to do a mixture of multinomials instead of a mixture of Gaussians. Use this mixture of multinomials to cluster the Shakespeare and Middelton documents for $M = 2$ different clusters. Show which documents go with which cluster and the training log-likelihood as you start EM and iterate it. Then report the log-likelihood as you vary the number of clusters for various random initializations. Then cross-validate to determine the best number of multinomials in the mixture ($M = 1, 2, 3$ and 4) by splitting the documents into training and testing (only use 2 documents of Shakespeare and 2 from Middleton for testing since the data is so small). Report average and standard deviation of training and testing log-likelihoods over 10 different random initializes for $M = 1, 2, 3$ and 4 multinomials in the mixture model.

Note on Numerical Issues. You may need to add a small amount (i.e. do MAP estimation) to the multinomials so that they don't give you numerical problems and you will need to work with log probabilities to avoid NaN and Inf numerical issues. Write a brief discussion about some of the peculiarities you noted with EM, numerical issues, convergence issues, etc. For the E-step you will need to calculate the following responsibility values (tau) which is the ratio $\tau_k = p_k / \sum_{i=1}^{M} p_i$. To avoid numerical problems and 0/0 problems, compute $l_k = \log(p_k)$ for all $k = 1, \ldots, M$ probabilities. Find the largest value $z = \max_{i=1}^{M} l_i$ and then compute tau using this formula: $\tau_k = \frac{\exp(l_k - z)}{\exp(l_1 - z) + \exp(l_2 - z) + \ldots \exp(l_M - z)}$ instead.

Note on Random Initialization. The search space is very large so if we're not careful we're likely to start very far from our clusters and may experience peculiar convergence characteristics. One neat way to handle this is to initialize the $\tau_{n,k}$ responsiblities rather than the distribution parameters directly, then start by running an M step.

Note on Convergence. As discussed in lectures, theoretically we should see log likelihood converging to a local maximum by rising monotonically at each iteration. On this data set, convergence is rapid, although potentially to quite different solutions.

3 Problem 3 (10 points): Jensen's inequality

Prove the following statements:
a) The the arithmetic mean of non-negative numbers is at least their geometric mean.

b) $\sum_{i=1}^{m} \exp(\theta^\top f_i) \geq \exp\left(\theta^\top \sum_{i=1}^{m} \alpha_i f_i - \sum_{i=1}^{m} \alpha_i \log \alpha_i\right)$, where $\alpha_i = \frac{\exp\left(\hat{\theta}^\top f_i\right)}{\sum_{j=1}^{m} \exp(\hat{\theta}^\top f_j)}$.

HINT: Use Jensen's inequality.