Mixture Models and K-means Clustering

COS424: Assignment # 2

Due : Thursday, March 15, 2012

Turn in your written assignments in class on Thursday, March 15th. Submit your electronic files for the R programing questions to CS DropBox at http://dropbox.cs.princeton.edu/COS424_S2012/Homework_2 by the end of the same day.

Written Exercises

Question 1: Mixtures of Gaussians (15 points)

Consider a mixture of Gaussians model defined by K means $\mu_1, \ldots, \mu_K$, variance σ^2, and proportions $\boldsymbol{\pi} = \langle \pi_1, \ldots, \pi_K \rangle$. In such a model, each (real-valued) X_n is generated as follows: First, one of the mixture components $Z_n \in \{1, \ldots, K\}$ is chosen at random according to $\boldsymbol{\pi}$ (so that $Z_n = z$ with probability π_z). Then, given that $Z_n = z$, X_n is chosen according to a Gaussian distribution with mean μ_z and variance σ^2. Note that only X_n is visible; Z_n is hidden. We assume that $\sigma > 0$ is known and fixed. Given data $X_1 : N$, recall the EM algorithm for estimating $\mu_1, \ldots, \mu_K$ and $\boldsymbol{\pi}$.

a. Argue, from the algorithmic perspective, that as $\sigma^2 \to 0$, this algorithm approaches the K-means algorithm.

b. Argue now that the EM objective approaches the K-means objective.

Question 2: Naive Bayes with mixed features (15 points) (adapted from MLAPA)

Consider a 3 class naive Bayes classifier with one binary feature and one Gaussian feature:

$$y \sim Mu(y|\pi, 1), x_1|y = c \sim Ber(x_1|\theta_c), x_2|y = c \sim N(x_2|\mu_c, \sigma_c^2)$$

Let the parameter vectors be as follows:

$$\pi = (0.5, 0.25, 0.25), \theta = (0.5, 0.75, 0.5), \mu = (-1, 0, 1), \sigma^2 = (1, 1, 1)$$

a. Compute $p(y|x_1 = 0, x_2 = 0)$ (the result should be a vector of 3 numbers that sums to 1).

b. Compute $p(y|x_1 = 0)$.

c. Compute $p(y|x_2 = 0)$.

(Show your computation for a, b and c.)

Programming Exercises

Question 3: K-means clustering (30 points)

In this problem, you will use K-means clustering on pixels of images for image compression. We have provided you with a true-color image and a greyscale image which are stored in the files PatchPanels.jpg and PatchPanelsGrey.jpg respectively.

a. Display the images after data compression using K-means clustering for different values of K (2, 5, 10, 15, 20).

b. What are the compression ratios for different values of K?

c. You will see that there is a trade-off between degree of compression and image quality. What would be a good value of K for each of the two images?

Here are some functions that might be useful for this problem:

- kmeans
- read.jpeg in package ReadImages
- imagematrix in package ReadImages

Electronic submissions for this question:

- *A file named Question3KmeansColor.R which performs K-means clustering on the true-color image.*
- *A file named Question3KmeansGrey.R which performs K-means clustering on the greyscale image.*
- *A file named Homework2.pdf containing your write-up and your plots. This file is for both question 3 and 4.*

Question 4: Mixtures of multinomials (40 points)

In this problem, you will implement parameter estimation using expectation-maximization for a mixture of multinomial distributions.

This model will take a fixed number of clusters as input, and find cluster proportions and per-cluster multinomial distributions. Given a model, in the E-step, compute the posterior cluster distribution for each data point. In the M-step, compute maximum likelihood estimates using expected counts, where the expectation is taken with respect to the distributions computed in the E-step.

Make sure that the expected complete log likelihood goes up at each step, and declare convergence when the relative change in this objective is smaller than 0.01%.

Implementation tips and tricks:

- In the E-step, to prevent underflow, compute the joint distribution and normalization in log space. Then exponentiate. In more detail, if $w_{1:N}$ are the words in a document and Θ are the model parameters, then the log posterior is

  $$\log p(z = k \mid w_{1:N}, \Theta) = \log p(z = k \mid \Theta) + \sum_{n=1}^{N} \log p(w \mid z = k, \Theta) - \log \sum_{i=1}^{K} p(z = i, w_{1:N} \mid \Theta)$$

  Note that the first two terms are $\log p(z = i, w_{1:N} \mid \Theta)$ for each i. To compute the log normalizer, and staying in log-space, we have provided a useful function that computes $\log(a + b)$ from $\log a$ and $\log b$.

- We have tried to arrange things so that $\log 0$ does not come up. However, in case it does, we suggest implementing a function called `safe.log` that returns `log` if the argument is non-zero and returns -100000 if the argument is 0.

- Initialize the kth cluster by choosing a document at random d_k, and setting the cluster to be:

  $$\beta_k \propto \vec{w}_{d_k} + \vec{\epsilon} + 10$$

  where note that $\vec{w}_{d_k}$ is the vector of counts for the d_kth document and $\vec{\epsilon}$ is a vector of random values between 0 and 1.

  Initialize the cluster proportions π to be uniform, i.e., $\pi_i = 1/K$ at the beginning of EM.

We have provided two discrete data sets on which to implement this mixture model and fold assignments for each data point. These data sets are in the files `corp1.Rdat` and `corp2.Rdat`. Each one contains objects `corp` and `vocab`.

a. For a fixed value of K and one of the data sets, plot the expected complete log likelihood as a function of iteration.

b. Just for kicks, start out each mixture component to the uniform distribution. Note that this is a bad idea. What happens? Why?

c. For a fixed value of K and for each data set, make a table of the top 15 terms from each cluster distribution and indicate their probabilities. What kinds of regularities have the models captured in the data? What kinds of data do you think these corpora are?

d. For $k \in \{2, 5, 10, 20, 30\}$, compute the held-out *perplexity*. Perplexity is a quantity used in the field of language modeling, which measures how well a model has captured the underlying distribution of language. For a particular document $w_{1:N}$, the perplexity is

$$\text{perplexity}(w_{1:N}) = 2^{\left\{-\frac{\log_2 p(w_{1:N} \mid \Theta)}{N}\right\}}$$

In this question, you will compute the average perplexity of the documents in the data set as a function of the number of clusters. For each data set, create folds with the following command

```
folds <- sample(rep(1:5, length=nrows))
```

Note that `nrows` is the number of rows in the corpus.

For each fold, fit a model on the *out-of-fold* data. Then, with this model, compute the perplexities of the *in-fold* documents.

Note that this will yield a perplexity value for each document in the collection. Further note that the dth document's perplexity is computed from a model that was not trained on a data set that contains the dth docment. The average perplexity is the mean of these per-document perplexities.

Electronic submissions for this question:

- *A file named Question4EM.R containing your R code for this question.*
- *A file named Homework2.pdf containing your write-up, your tables and your plots. This file is for both question 3 and 4.*