Mixture Models and Regression

COS424: Assignment # 3

Due : Wednesday, April 16, 2008

Written Exercises

Mixtures of Gaussians

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.

a. Given data $X_1 : N$, describe in detail the EM algorithm for estimating $\mu_1, \ldots, \mu_K$ and $\boldsymbol{\pi}$.

b. Argue that as $\sigma^2 \to 0$, this algorithm approaches the K-means algorithm.

c. Argue directly that as $\sigma^2 \to 0$, the EM objective approaches the K-means objective.

Regularized Regression

As is usual for linear regression, suppose we are given training data $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)$ where $y_i \in \mathbb{R}$ and $\mathbf{x}_i \in \mathbb{R}^n$ (with components x_{ij}). In this problem, we seek linear models of the form $\hat{f}(\mathbf{x}) = w_0 + \mathbf{w} \cdot \mathbf{x}$ where w_0 is the scalar intercept term, and $\mathbf{w} = \langle w_1, \ldots, w_n \rangle$ is a (column) vector of weights over the n inputs. Consider the problem in ridge regression of minimizing

$$\sum_{i=1}^{m} (w_0 + \mathbf{w} \cdot \mathbf{x}_i - y_i)^2 + \lambda \left\| \mathbf{w} \right\|_2^2 . \tag{1}$$

Here, as in Hastie et al. (but unlike in class), we include an explicit intercept term w_0, but omit this term from the regression penalty.

a. Suppose *for this part only* that $\sum_{i=1}^m x_{ij} = 0$ for all j. Let $\mathbf{X}$ be the $m \times n$ matrix of all inputs in which the i-th row is equal to (the transpose of) $\mathbf{x}_i$, and let $\mathbf{y}$ be the (column) vector whose i-th entry is y_i. Show that the solution of (1) is given by

$$\begin{aligned}\hat{w}_0 &= \frac{1}{m}\sum_{i=1}^m y_i \\ \hat{\mathbf{w}} &= (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{y}\end{aligned}$$

where $\mathbf{I}$ is the $n \times n$ identity matrix.

b. Returning to the general case (in which the input vectors do not sum to zero), let

$$a_j = \frac{1}{m}\sum_{i=1}^m x_{ij}$$

and define $\mathbf{x}'_i$ by $x'_{ij} = x_{ij} - a_j$. Note that, after centering in this fashion, the new input vectors sum to zero so that the technique in the last part can be applied. Show that minimizing (1) is equivalent to minimizing

$$\sum_{i=1}^m (w'_0 + \mathbf{w}' \cdot \mathbf{x}'_i - y_i)^2 + \lambda \left\|\mathbf{w}'\right\|_2^2 . \qquad (2)$$

In other words, if $\hat{w}_0, \hat{\mathbf{w}}$ is the solution that minimizes (1), and $\hat{w}'_0, \hat{\mathbf{w}}'$ is the solution that minimizes (2), show that $\hat{w}_0 + \hat{\mathbf{w}} \cdot \mathbf{x} = \hat{w}'_0 + \hat{\mathbf{w}}' \cdot \mathbf{x}'$ for any $\mathbf{x}$ and its transform $\mathbf{x}'$. Moreover, given a solution $\hat{w}'_0, \hat{\mathbf{w}}'$ to (2), show explicitly how to transform it directly into a solution $\hat{w}_0, \hat{\mathbf{w}}$ to (1).

c. Suppose that the inputs are both centered *and* scaled. In other words, suppose we instead define $\mathbf{x}'_i$ by $x'_{ij} = (x_{ij} - a_j)/s_j$ for some constants s_j. Show that the minimization problems (1) and (2) need no longer be equivalent (in the sense described above). Show nevertheless how a solution $\hat{w}'_0, \hat{\mathbf{w}}'$ to (2) can be transformed back into $\hat{w}_0, \hat{\mathbf{w}}$, not necessarily a solution to (1), but for which $\hat{w}_0 + \hat{\mathbf{w}} \cdot \mathbf{x} = \hat{w}'_0 + \hat{\mathbf{w}}' \cdot \mathbf{x}'$ for any $\mathbf{x}$ and its transform $\mathbf{x}'$.

Programming Exercises

Mixtures of Multinomials

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

$$\text{perplexity}(w_{1:N}) = \exp\left\{-\frac{\log_2 p(w_{1:N} \,|\, \Theta)}{N}\right\}$$

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.

e. (Extra credit.) Collect a continuous data set. Implement the mixture of Gaussians model and plot a number of visualizations and analyses of your data with it.