

CS 445/545: Machine Learning Winter, 2009

Homework 3: Due Wednesday March 11, 5:00pm.

1. Kmeans clustering.

(a) Use the *kmeans* function in Matlab (part of the Statistics toolbox, which is already loaded in Matlab on the CECS machines) to cluster the optdigits training data (minus the class values) into 10 clusters. To do this, create a data matrix called “optdigits.train” and load it into Matlab. This should be a 3823 x 64 matrix, in which each row is a training example with 64 features.

Run the command `[idx, C, sumd, D] = kmeans(optdigits, 10)`. The resulting matrix **idx** will be a 3823 x 1 matrix of the cluster assignments of each training example. E.g.,

`idx =`

```
7
7
9
5
1
3
5
10
.
.
.
```

This means that the first training example was assigned to cluster 7, the second training example was assigned to cluster 7, and so on.

(b) By comparing with the training data with the true class assignments, determine which cluster best corresponds to which digit. For example, if a majority of instances in cluster 7 correspond to digit 0, then assume that cluster 7 corresponds to digit 0. Given this classification method, give the overall error rate of this classifier on the training data as well as a confusion matrix for this classifier on the training data.

(c) The resulting matrix **sumd** is 10 x 1 matrix giving within-cluster sums of point-to-centroid distances for all the points in each cluster. Describe how these relative distances relate to the relative error rate on each digit (computed in part (b)).

(d) The resulting matrix **D** is a 3823 x 10 matrix that gives the distance from each training example to each centroid. Create a scatter plot: for each centroid, plot the distance from each example to that centroid. If you can, color-code the examples according to their true class. Write a few sentences on your observations about this plot.

(d) The resulting matrix \mathbf{C} is be a 10 x 64 matrix giving the cluster center coordinates. Using these coordinates, use Matlab to calculate the Euclidean distance from each example in the **test set** `optdigits.test` to each cluster, and assign each example to the closest cluster. Using these assignments, compute the overall error of this classification method on the test set as well as a confusion matrix. How does this compare with your previous SVM results on this data set?

2. Kmeans vs. Gaussian Mixture Models and EM.

(a) Download and read about the “wine” database from

<http://archive.ics.uci.edu/ml/datasets/Wine>

Create a file that contains the wine examples without the class labels. (Note, there is no separation here between “training” and “test” data.)

(b) Run *kmeans* on this dataset, as above, specifying 3 clusters. Give the classification error of the kmeans clustering and a scatter plot of the distance from each example to each of the three centroids.

(c) Use the *gmdistribution.fit* function in Matlab to create a Gaussian mixture model of this data (this uses the EM algorithm). This will produce an object of type “Gaussian mixture distribution with 3 components in 13 dimensions”.

(d) Use the *cluster* function to cluster the wine data according to the Gaussian mixture model created in step (c). Give the classification error of the GMM clustering and a scatter plot of the Mahalanobis distance from each example to each of the three centroids. (See the documentation for the *cluster* function for GMMs in the Statistics toolbox to find out how to do this.) Write a few sentences with your observations about the classification error and scatter plot using the GMM versus the classification and scatter plot using Kmeans. Why do you think one seems to work better than the other?

3. **Hidden Markov Models.** This part of the homework will be handed out on Wednesday.

What to turn in

A writeup with all results asked for above.

Policy on late homework: If you are having trouble completing the assignment on time for any reason, please see me **before** the due date to find out if you can get an extension. Any homework turned in late without an extension from me will be graded down one grade (e.g., “A” to “B”).