

## CS 374 Assignment #11 Clustering

Due the week of April 28, 2008

This week you will have the opportunity to explore two algorithms for unsupervised learning – specifically, two clustering algorithms. In unsupervised learning, data are presented to an algorithm in the form of training examples. The difference between these training examples and those you have seen earlier this semester, is that they are unlabeled. That is, the training examples are described by attribute information, but they have no associated class. The goal in clustering is to find groups of examples that are similar to each other but distinct from other groups of examples. In other words, this is one way to think about finding patterns in an unsupervised setting.

This will be your last structured assignment of the semester. Next week each of you will give a brief (10-15 minute) presentation on a project of your choice. (See my recent email for more on this.)

### 1 EM and k Means

The two algorithms you will study this week are the simple k-means algorithm and the EM algorithm.

The basic idea of the k-means algorithm is as follows. Randomly select k points in your example space. These are taken to be the initial centers of clusters. (So there will be k clusters.) The training instances are then assigned to a cluster, based on their distance from each of the k centers. Once all examples have been assigned to clusters, k new centers are found by computing the mean of each cluster. The process then repeats. All training instances are assigned to clusters based on their distance to the new centers, etc. Once the centers become stable, you're done. A brief description of this algorithm and a visualization can be found at:

[http://home.dei.polimi.it/matteucc/Clustering/tutorial\\_html/kmeans.html](http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/kmeans.html)

There are clearly advantages and disadvantages to this algorithm. There are also alternatives, such as the EM algorithm. EM assumes that the training instances were generated by a mixture of Gaussians and uses this assumption to perform clustering in a manner roughly analogous to k means. It hypothesizes the values of the parameters and then revises those hypotheses with each iteration of the algorithm.

#### 1.1 Reading

There are many sources from which you can learn about the EM algorithm and about simple k means. They include:

- Mitchell, Section 6.12,
- Alpaydin, pages 140-144,
- Witten and Frank, pages 137-138, 262-266, 337-338, and
- Bishop, 187-190, 65-72 (note that this final reading is quite dense).

You aren't required to read all of these. It's up to you to read as much as you need in order to do the implementation and to understand the research paper (both described below).

## 1.2 Exercises

This week your primary exercise will involve the implementation of the k-means algorithm. This algorithm is conceptually fairly simple, but as you've seen with other algorithms you've implemented this semester, some details can be tricky and debugging is non-trivial. Please don't wait until the last minute to start this.

As the readings make clear, there is no known theoretical way to select an optimal value for  $k$ , the number of clusters. For this exercise, however, you will be training on data sets for which we have class labels. *You should not treat the class labels as attributes!!!* However, you can use this information to specify a good value for  $k$ .

The two data sets with which you will be working are `iris` and `vehicle`. I've selected these as they have only real-valued attributes. This is essential as you will need to compute the Euclidean distance from a training instance to the center of a cluster. (There are versions of clustering algorithms that work with discrete-valued attributes, but we won't consider them this week.) In order to handle real-valued data, you will undoubtedly need to make some changes to the code you wrote previously to read data from an "ARFF" file. The changes should be fairly small, however.

Your k-means clustering program should do the following:

- Read the training instances from the file. Don't discard the class information. While you can't use it for clustering, you will need it later for assigning names to the clusters and for checking the accuracy of the clusters. You do not need to normalize the attribute values.
- Apply the k-means algorithm to find clusters (three in the case of `iris` and four for `vehicles`).
- Assign each final cluster a name by choosing the most frequently occurring class label of examples in the cluster.
- Find the number of examples that were put in clusters in which they didn't belong. You can check your results by comparing with Weka. Note that your results for `iris` will likely be close to Weka's in most cases. The `vehicle` results will vary more.

## 2 Cluster Outliers

### 2.1 Reading

Please read

- "LOF: Identifying Density-Based Local Outliers", a paper by Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jorg Sander.

You can find a link to the paper by going to the Assignments page for this course. This paper describes a method for determining the degree to which an object is an outlier of a cluster. The paper refers frequently to density-based clustering. While it isn't essential for understanding this paper that you know how density-based clustering is done, if you'd like to learn more, you can read "Density-Based Clustering in Spatial Databases: The Algorithm DBSCAN and its Applications" (also available through a link from the Assignments page).

### 2.2 Exercise

Please write a review of the paper. As usual, you should follow the general format that researchers use when reviewing papers for conferences or journals. The review should begin with a very brief assessment

of your own expertise/comfort level in reviewing the paper. It should then give a summary of the paper, including all of the major results. Next it should include a critique. You should include both positive and negative comments. You should comment on such areas as importance, technical soundness, and clarity.

Please type your report, which should be a minimum of two pages of text (12-point font, 1.5 spacing).

In your tutorial session, expect to discuss the paper and to share your written critiques with each other. (That means that the critiques must be written beforehand.) This will allow you to provide your classmates with constructive criticism of their work.