CS 401R Lab 5
Data Clustering
Due 10 December 2015

For your experience, you must do this lab using the R software, as it is one of the richest and most versatile freeware tools for statistical analysis and data mining. Some information is given here to get you started. There are also a number of online tutorials for R that you can refer to as you complete this lab.

1. Download and install the latest version of R on your computer. The following is a direct link: https://www.r-project.org. [There also exists a rather nice IDE for R known as RStudio (https://www.rstudio.com). While not necessary for this lab, you may still wish to download it and use it. Finally, there is a platform known as Revolution R (http://www.revolutionanalytics.com/products) that extends R to run on big data in parallel and distributed environments. It is not needed for this lab, but you may wish to have a look].

2. Install the following packages: datasets, stats, and dbscan. To do so, click on Packages & Data in the menu bar, and select Package Installer. Click on Get List. You will be prompted to select a CRAN mirror. Select (HTTP mirrors) [It should be the last option in the drop-down list]. Select USA (CA1) in the new drop down list. A list of packages will then appear. Click on the above packages in the list (if they are not there, they have probably been loaded by default when R installed; skip to the next step). Make sure you tick the Install Dependencies box before you click Install Selected. After the packages have been installed, click on Packages & Data again in the menu bar, and this time select Package Manager. Click on the above packages so they get loaded [Some may have been loaded already].

You are now ready to complete this lab. Details on the various functions implemented in each package, as well as examples of usage may be found in the Package Manager by selecting the package of interest.

3. Run the k-means algorithm (kmeans in R) on the iris dataset (iris was loaded above when you loaded the R datasets package). Of course, iris has a target attribute. It must be excluded from the clustering. The simplest way to do this is to create a copy of iris consisting of only the first 4 attributes. This can be accomplished with the command: 'iris_copy <- subset(iris, select=c(1:4))'. You can then run kmeans on iris_copy.

   a. Run k-means for k = 2, 3, 4, 5, 7, 9, 11.

   b. For each value of k, report the size of the clusters and the F-measure (see the slide deck Basic Clustering for details; there is a link to it on our class schedule). Both size and cluster assignments are available in variables computed during k-means (see the documentation in the R stats package under kmeans). You will need the target values from the original iris dataset to compute the F-score. You may write a small program in R to do this (this is the preferred method as it will give you
further experience with R and you will use that program again below) or export the data and compute elsewhere.

c. Report the value of $k$ that produces the highest F-score.

d. Comment on anything interesting about your experiment.

4. Run the hierarchical agglomerative clustering algorithm (hclust in R) on the iris dataset using complete link for the distance. Be mindful that hclust requires a distance matrix rather than a set of points as input. You can easily transform a set of points into its equivalent distance matrix using the `dist()` function. From `iris_copy`, you could thus construct `iris_dist` with the command: `'iris_dist <- dist(iris_copy)'`. You can then run `hclust` on `iris_dist`.

   a. Display and include in your report the result of hierarchical agglomerative clustering, including the dendrogram. Use the function `plot()` to graph the dendrogram. You can copy the result to a postscript file using the following commands: `'postscript("nameoffile.eps")'`, then `'plot(resultofhac)'`, and finally `'dev.off()'`.

   b. By looking at the display or using the values of clustering heights, select a threshold at which you feel the clustering would be optimal and justify your choice. (In principle, we would do this by computing some quality measure during the clustering process, but for simplicity, we are just eyeballing here).

   c. How does the corresponding number and nature/composition of clusters compare with that obtained with k-means clustering above? Cluster composition (or assignments) is available in variables computed by the clustering algorithms.

5. Run the density-based clustering algorithm (dbscan in R) on the iris dataset. Be mindful that dbscan requires a numeric matrix rather than a set of points as input. You can easily transform a set of points into its equivalent numeric matrix using the `as_matrix()` function. From `iris_copy`, you could thus construct `iris_mat` with the command: `'iris_mat <- as.matrix(iris_copy)'`. You can then run `dbscan` on `iris_mat`.

   a. Run `dbscan` for epsilon (eps in `dbscan()`) = 0.2, 0.3, 0.4, 0.5, 0.6, 0.8, and 1.0.

   b. For each value of eps, report the size of the clusters and the F-measure (see the slide deck Basic Clustering for details; there is a link to it on our class schedule). Both size and cluster assignments are available in variables computed during `dbscan` (see the documentation in the R stats package under `dbscan`). You will need the target values from the original iris dataset to compute the F-score. You may use the same program as for k-means or export the data and compute elsewhere.

   c. Report the value of eps that produces the highest F-score.
d. Comment on anything interesting about your experiment.

e. How does the corresponding number and nature/composition of clusters compare with that obtained with k-means and hierarchical agglomerative clustering above? Cluster composition (or assignments) is available in variables computed by the clustering algorithms.

6. Consider the swiss dataset (swiss was loaded above when you loaded the R datasets package). Use a clustering algorithm, whichever of the three studied here seems to make most sense, to produce a list of the Swiss cities predominantly protestant and those predominantly catholic. You may produce a graph or simply a list of the cities.