Cory Mollenhour
CSCI 4350
Open Lab 4 - K-Means Clustering
12/5/2018

K-Means Clustering

Summary

When presented with a large data set consisting of defining attributes, it is possible to use a predictive algorithm to determine the underlying pattern from the data alone. The K-Means algorithm can learn these patterns and classify the data accordingly, as well as predict the classification of future data after training. This is known as Unsupervised Learning, because the test data does not require any specification of class prior to being tested.

Overview

The process begins with supplying the K-Means algorithm with a set of “training” data, which usually consists of attribute data. When supplied with enough attribute types, the algorithm begins to correlate the data as clusters. The number of clusters (K) can be specified at the beginning of the program to increase prediction accuracy. The clusters center around nodes called “centroids” which move through the data to determine the best fit. As the centroids move, their nearest neighboring nodes become part of its cluster. The centroid stops changing position once all of the clusters remain the same for two consecutive iterations. At this time the class values can be counted to determine the majority. The majority class becomes the prediction and is recorded with the centroid.

Once the algorithm is trained, the next input can be a set of test data. This data is not given any class label, but the program can attempt to determine, by probability, what is the most likely class of each set of attributes. As the number of specified clusters increases, the accuracy can be further and further refined.

This process is further illustrated on the next page.

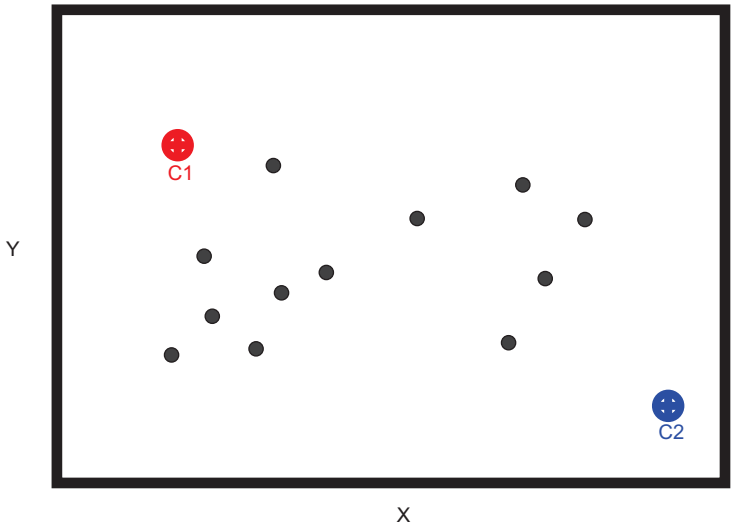
Challenges

The process of debugging this code was the most challenging as there are many data structures being used and many loops to iterate through them. This could probably have been improved using a more class/method based approach, rather than coding the majority of the program in the main() function. However, coding each stage seemed to work well as it allowed me to define potential issues at each stage separately. Such issues were found with segfaults during the for loops, and stalling during the while loop.

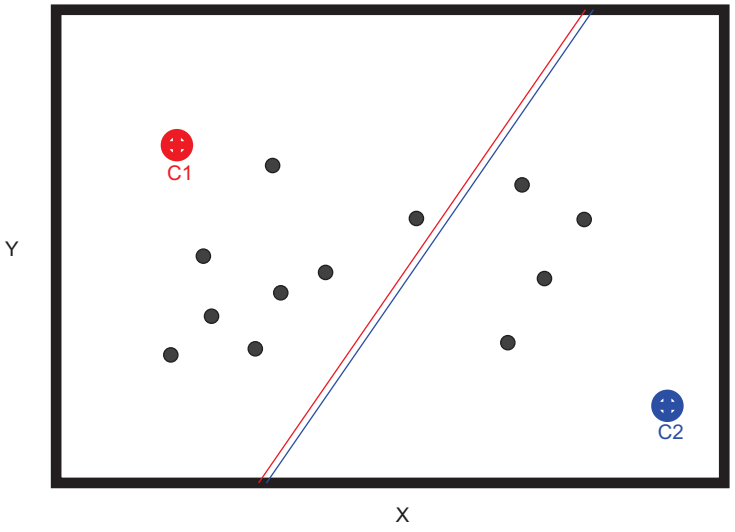
What I learned

I have gained a better understanding of logic behind the algorithm, and how to use them to predict probable classifications. I also learned that K means algorithms are used in detecting high risk crime zones in cities, predicting fantasy sport statistics, and detecting insurance fraud (Kaushik). However, it seems we are just scratching the surface on the implementation of K means clustering and other unsupervised learning algorithms.

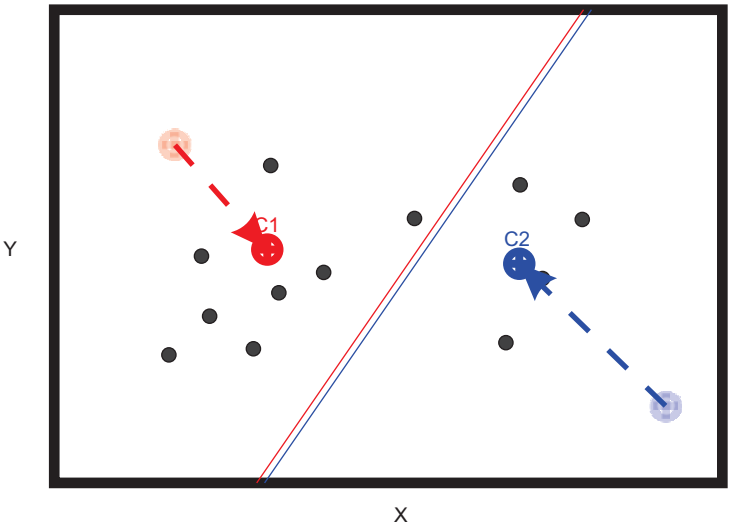
Kmeans Clustering with 2 Attributes before calculating distances.



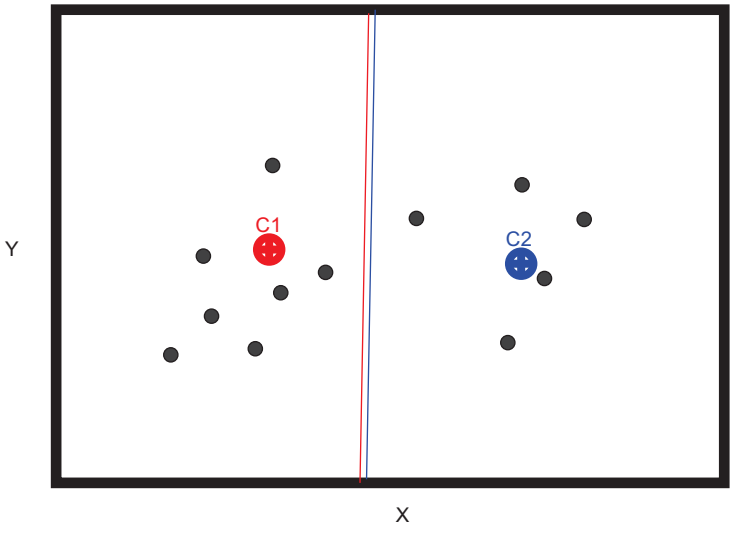
Node associations after calculating distance.



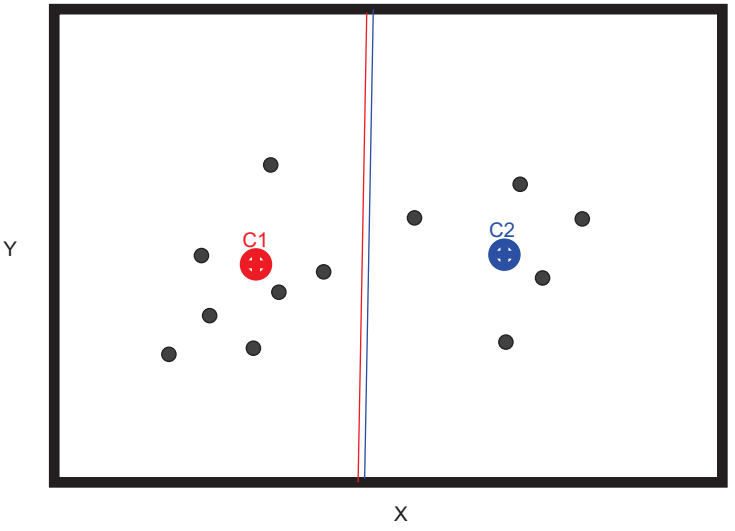
Centroids moved after averaging distances to cluster nodes.



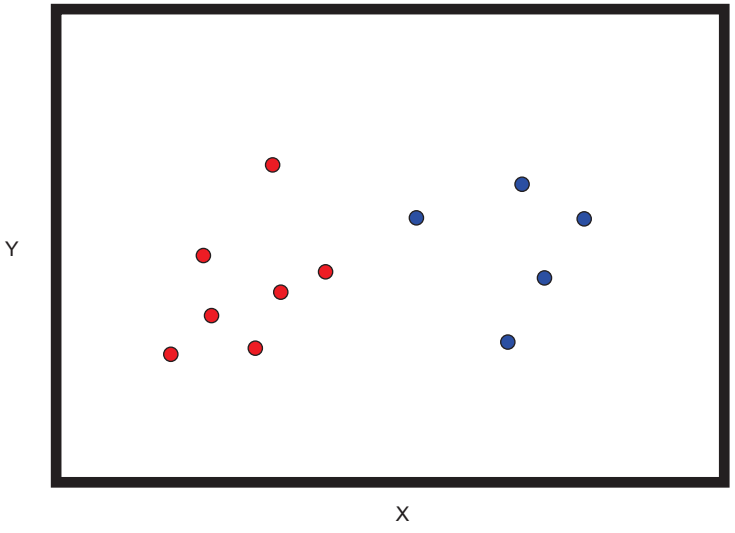
Distance is calcuated again to determine which nodes belong in cluster.



Centroids move again to average position of their cluster nodes



No more changes necessary, centroids remain in position
Now we assign the centroid class label to all cluster nodes to make predictions



Cory Mollenhour
CSCI 4350
Open Lab 4 - K-Means Clustering
12/5/2018

The Code In Summary

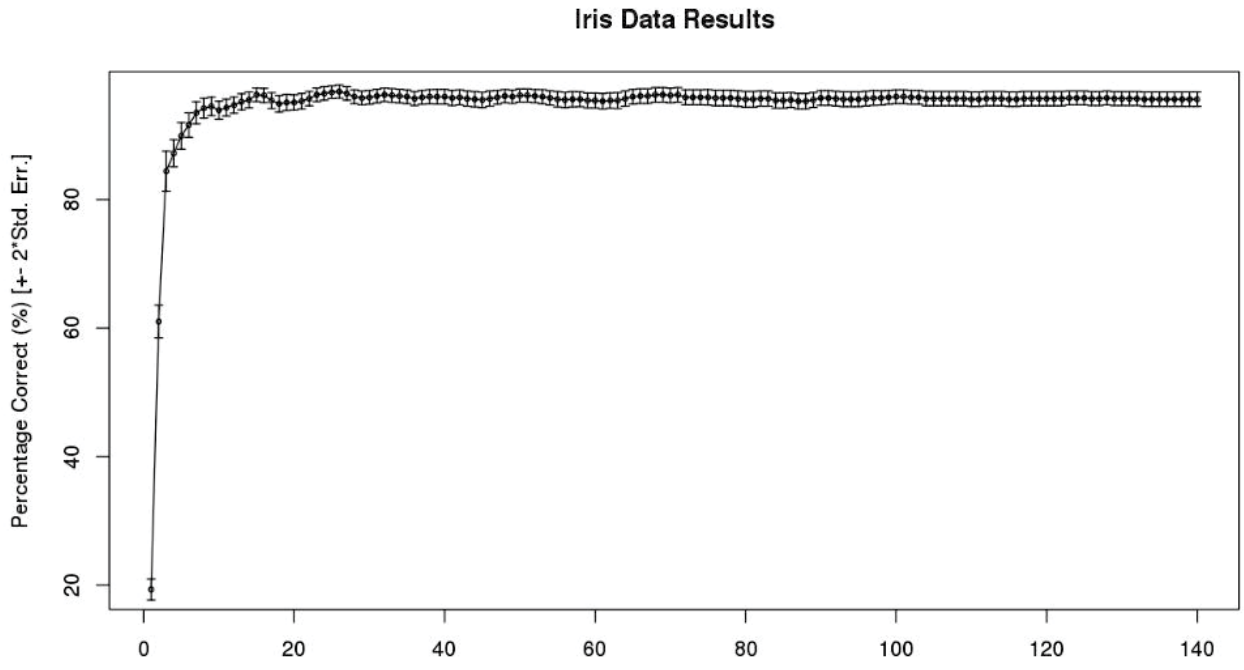
In order to code the K-Means algorithm, I found it helpful to break the process into defined stages.

1. Generate centroid node objects
    a. Use randomly selected points from the training data.
    b. Ensure each centroid position is only used once.
2. Begin repeating process (while clusters change data members)
    a. Create a vector containing each class amount initialized to 0 for each centroid object.
    b. Determine distances of all nodes to all centroids using Euclidean distance.
    c. Populate each centroid with the nodes nearest to it while updating the class counts each time for the centroid.
    d. Make a new centroid at a new position using the average distance from the original centroid to each node in all dimensions.
    e. If the cluster contents do not change twice in a row, then stop repeating.
3. Get total predictions made correctly using the current K cluster configuration by checking which centroid each testing node is closest to
4. Output the amount of correct predictions made.

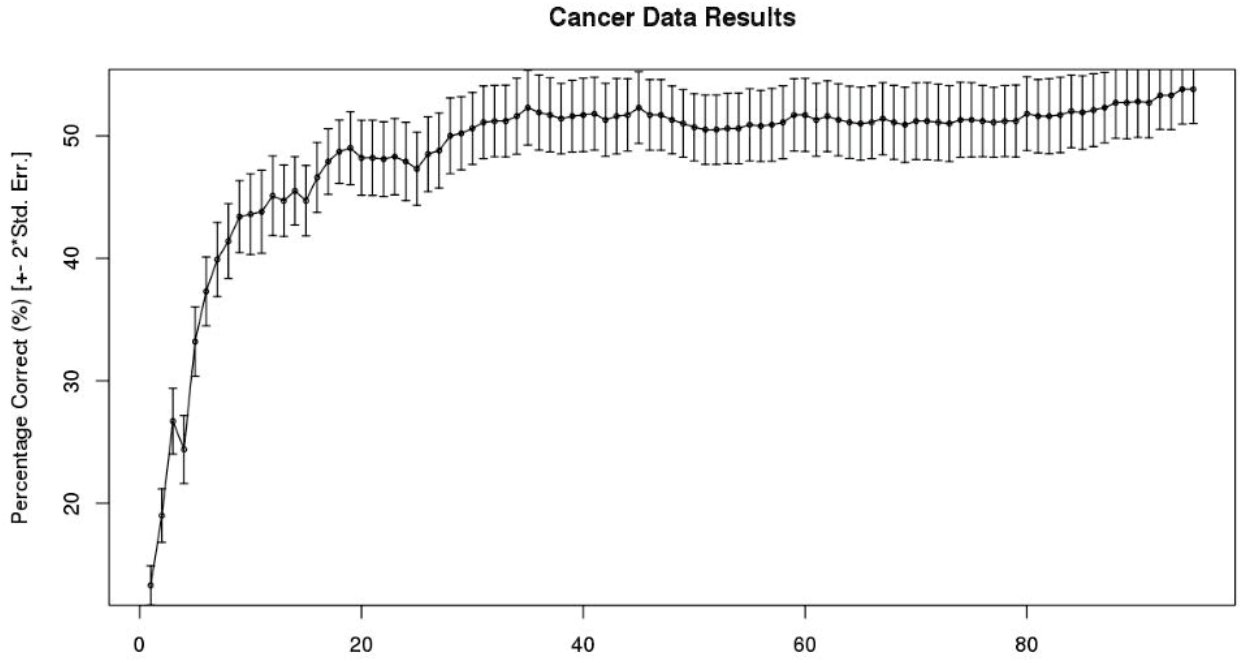
Cory Mollenhour
CSCI 4350
Open Lab 4 - K-Means Clustering
12/5/2018

My Results

The following results were made with 100 shuffles of the iris data and the first 140 lines used for training with the last 10 lines for testing.



The following results were made with 100 shuffles of the cancer data and the first 140 lines used for training with the last 10 lines for testing.



Cory Mollenhour
CSCI 4350
Open Lab 4 - K-Means Clustering
12/5/2018

Works Cited

Raghupathi, Kaushik. “10 Interesting Use Cases for the K-Means Algorithm - DZone AI.” Dzone.com, 27 Mar. 2018, dzone.com/articles/10-interesting-use-cases-for-the-k-means-algorithm.