Kevin Peterson
OLA4

K Means Clustering Algorithm

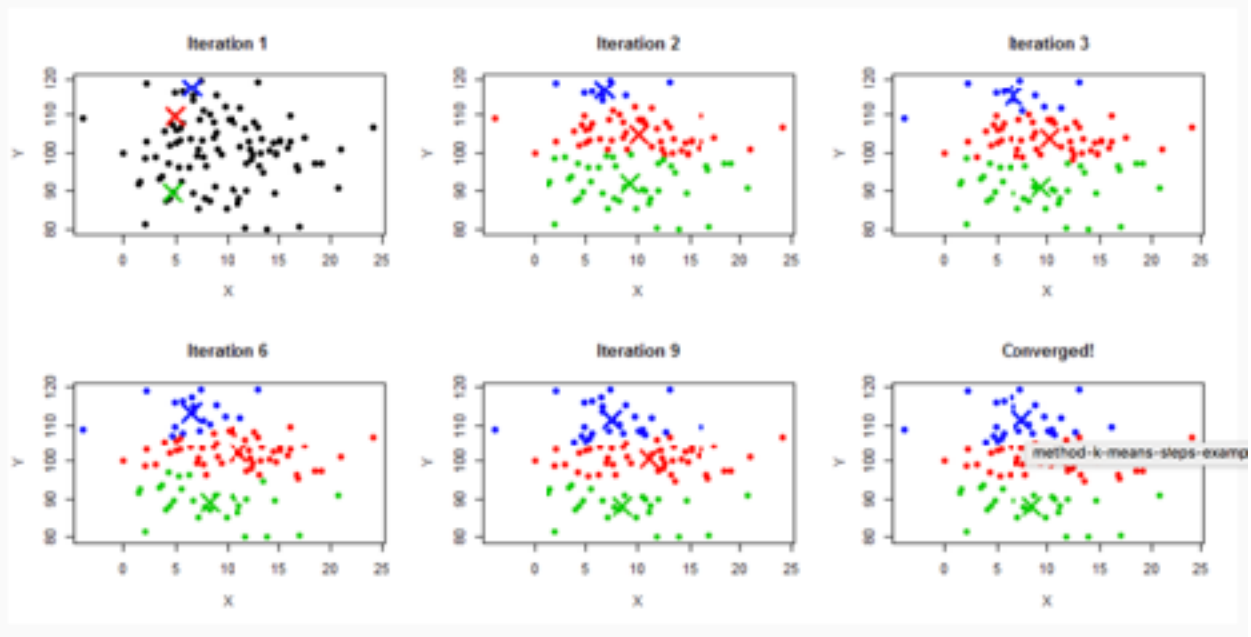
Introduction

K means clustering is a Unsupervised machine learning algorithm that separates data into K clusters by their attribute values. Because K means is unsupervised, this means that it has no training data to learn from. It cannot see the classification of certain data's class labels before it classifies the data into its own clusters.

The algorithm works by picking K random points out of the data and creating "centroids" at those points. These centroids are where the eventual cluster means will be. The algorithm repeats a simple process over and over again in order to move these centroids to their correct places within the data.

1. All points in data set are assigned a centroid by whichever centroid has the least Euclidian distance value away from the point
2. The centroids are moved to the location of the mean of all the points assigned to that given centroid. (It is moved to the center of the cluster).

The algorithm keeps looping until all of the points are not assigned to a new centroid and have not moved. We can then say the centroids have converged and found their final location. An illustration of the K means algorithm with two attributes is provided below: (source: http://www.learnbymarketing.com/methods/k-means-clustering/):



Approach

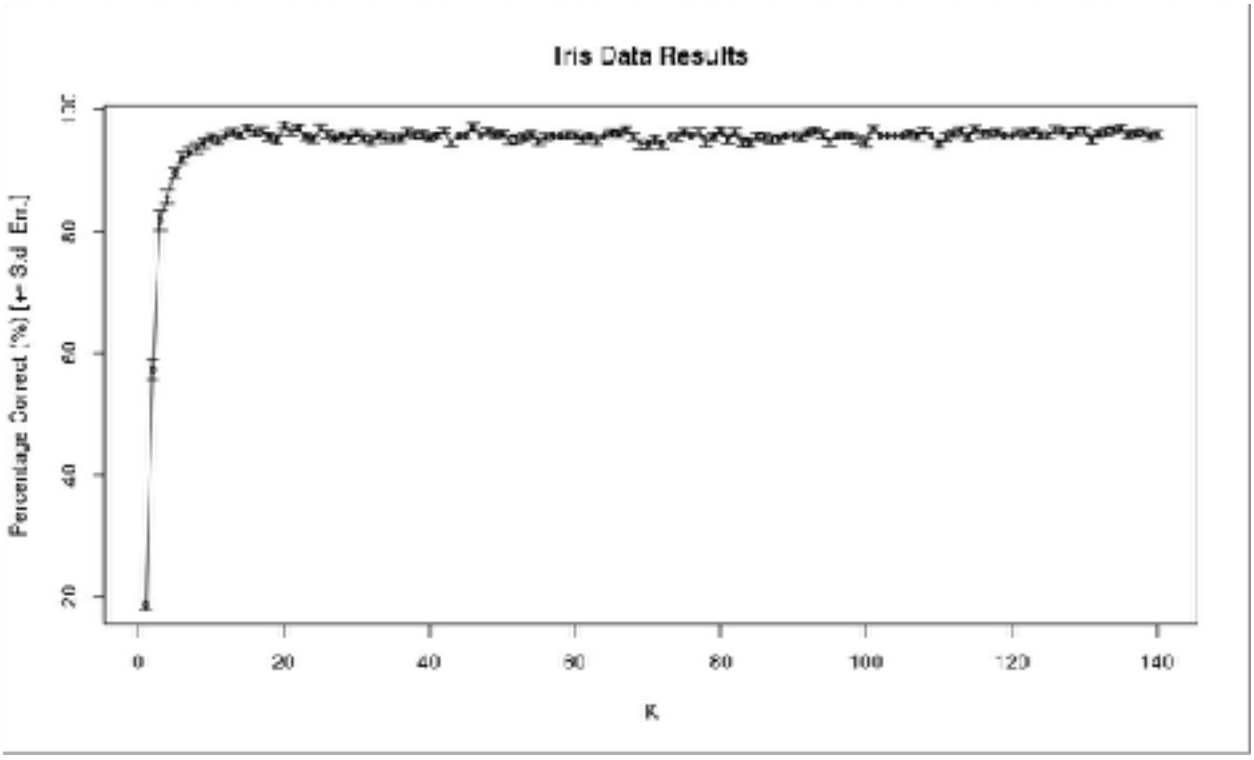
In order to implement the K means algorithm, corresponding vectors were used for the data points and their centroids. The algorithm was run and each point was assigned a centroid over and over until the centroids were stationary and had converged. Then each centroid was assigned a class label, based on the points class labels from the training data. A simple majority vote was taken in order to assign each centroid a class label. For example, if most of the points in the cluster represented by centroid 0 had a class value of 1 in the training data, centroid 0 was assigned a class label of 1.

In order to test the algorithm, 10 testing data points would be read and assigned to a cluster based on the closest centroid. They would then be classified by the label that was previously assigned to that

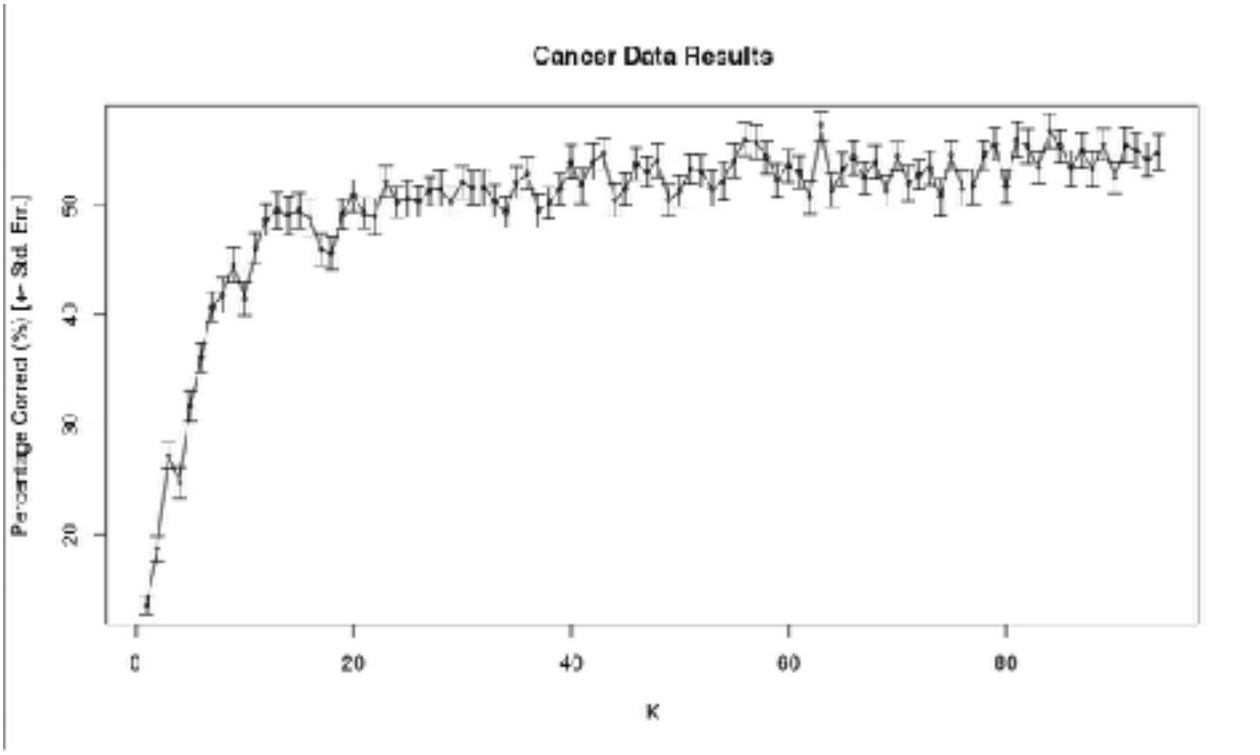
centroid after the K means algorithm had converged. The program would output how many of the testing points were correctly classified.

Results and Statistics

Statistics were gathered for the K means algorithm on two different data sets. The algorithm was tested on the cancer data set and the iris data set. The program was run 100 times for each number of clusters and the mean and standard error were calculated and plotted for each number of clusters.



K means clustering performed very well on the Iris Data. The Iris data contained 3 different class labels so the clustering algorithm performed poorly up until it had 3 or more clusters. The algorithm reached a max performance fairly quickly at approximately 9 clusters with a mean of 94.5% accuracy and .81 standard error. Even at 100+ clusters, the mean never reached above 96.7% accuracy. This means that if we were using this algorithm for classification, it would be most useful for us to use somewhere between 8-10 clusters. Based on the standard error, we cannot be statistically confident that higher amounts of clusters will give us more accurate results. Using more clusters would be an inefficient use of data and cpu cycles.



K means clustering did not perform quite as well on the cancer data, as it is a harder problem that is less easily classified. At around 12 clusters, the cancer data means started to flatten out. At 12 clusters, there was a mean of 48.7% accuracy and a standard error of 1.45. Although the data started to flatten out at this point, it does not completely flatten out like the Iris data does. At 140 clusters, the mean is 54.8% accuracy and the standard error is 1.6. It becomes a difficult question then what the best number of clusters to use for this data is. Because there is a statistically significant difference in accuracy between 12 clusters and 140 clusters. However, there is also a large decrease in time and space efficiency.

In conclusion, the K means clustering algorithm works very well on problems that have distinct data inputs and are easily classified. However it has a much harder time working on data that is more ambiguous and less easily classified. This is nicely illustrated in the difference in the Iris data set, which reached ~95% accuracy, and the cancer data set, which reached ~55% accuracy. Another issue with the K means algorithm is the decision on how many clusters to use. With real world problems like the cancer data, there seems to be a constant tradeoff in efficiency of the algorithm and accuracy of the classifications. There is no clear point as to how many clusters to use. The power of the K means algorithm lies in the fact that it does not have to be told the correct answer and trained like supervised learning methods do. It can form classifications without seeing the correct class labels of the training data before testing.