HW #1 - Clustering

Implementing the K-means Algorithm

Youchen Ren

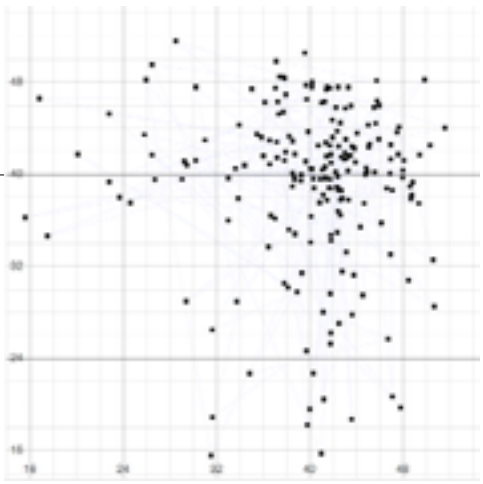
March 10, 2014

A. How to randomly generate the starting centroids in Step 2.

I am using the "Random" class for generating the random "double", and store these 3 points into an array. But the random points must be in the range of the clusters, meaning the points cannot exceeds the "edge" of the clusters. Because if so, the cluster assignment which based on the randomly generated centroids can be easier become as an unique cluster(all the points belong to one cluster).

B. How to manually pick the starting centroid in Step 6.

I am using a graph software on Mac OS X called "Grapher" to plot all the points and visually picked the approximate centroids.(see Figure)



C. Details of implementation that may be important.

1. Since the delimiter of the original data file is "tab", therefore, when reading the file, I need to use "regex" to determine my delimiter("\\s") instead of the regular "space" (" ")delimiter;
2. Randomly select the centroids, this action must be doing within the "range" of the points (as stated in B above);
3. When implementing the method of "FindClusterCentroid", the important point is that I need to know how many points in each cluster so that I may use this number as denominator for averaging the centroids points.
4. For calculating the IV and EV, it's likely to mis-coding the for loop, especially the EV method, I need to focus on the model for the algorithm.

D. All sets of Starting Centroids, Final Centroids, IV, EV, and IV/EV in a table.

(see Appendix attached with this report.)

E. Discuss results! Which set of starting centroids generates the best results? Any observation why this set of centroids is good?

Obviously, the manually picked centroids. That's because as human justifying, we could approximately know the centroids at some place, but the computer, it does not know where the centroids are, just based no the randomly generated numbers.

F. Any other observations you may have on your experimental results.

Based on E, I think if we have some much more talent way to choose the centroid(for instance, as what the human eyes did) at the beginning, the algorithm could reduce the computing and running time.

E. K-means Algorithm Intro:

Based on Professor's lecture, I summarized the algorithm by following:

1. Generating the randomly centroids;
2. Assign the points to the clusters based on the closest centroids.
3. Recalculate the centroids based on the points which has already been clustered.
4. Compare the centroids, if the Centroids do not change any more from step 2 to 3, then we find the final centroids, else repeat step 2 to 3.