CS434 Assignment 4 Report

Jiongcheng Luo

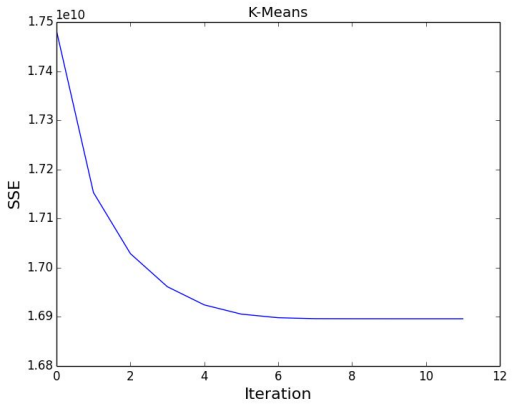
Jiayi Du

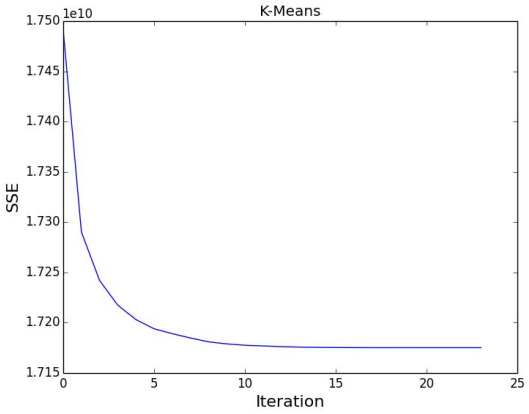
Tao Chen

1. Non-hierarchical clustering - K-Means Algorithm

a. Implementation (single K-value):

We chose k = 2 (assign data to 2 clusters) and kept looping over to re-assign data to clusters until convergence. Our program determines convergence by calculating the sum SSE from both cluster. The following graphs (with different initial centroids) plots the SSE (y-axis) versus the number of iterations (x-axis). At the 11th and 25th iteration, the SSE converged.



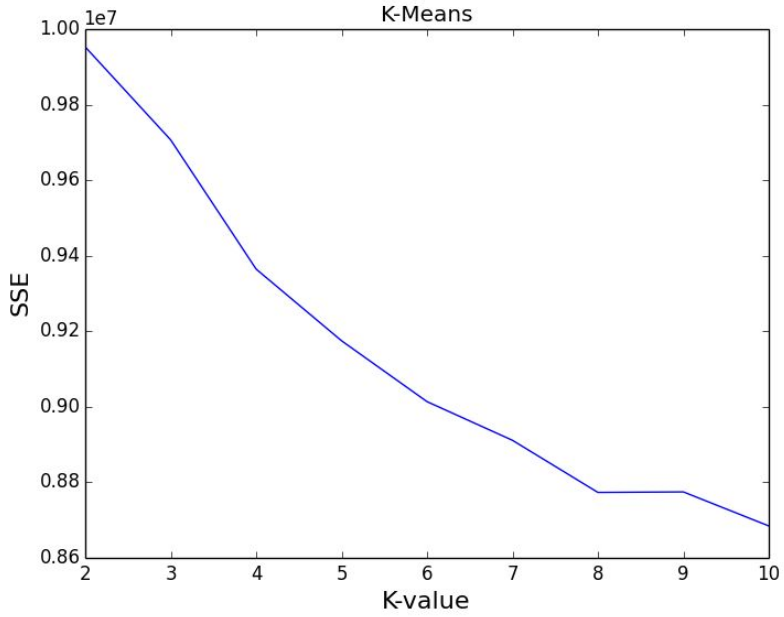


b. Observation:

The two graphs displayed the same trend. It makes sense because SSE always decreases as points are reassigned to another cluster.

2. Implementation (Multiple K-values):

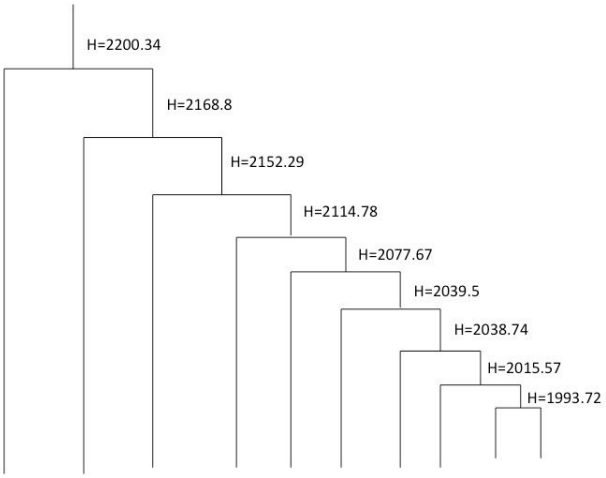
The second question was implemented in the same way as the first question. We added a for loop so that we could test with different K values (2~10).



This plot clearly illustrates that SSE decreases as K increases. When we have a large K, we are splitting the data into more categories. Too many categories might lead to overfitting. It is similar to KNN. When K reaches the number of points in a given data set, the SSE will become 0. Based on the graph above, we believe 4 or 8 will be promising K values, because the absolute value of the slope of the curve decreased at those two points, meaning that more clusters beyond these points would not necessarily return better results.

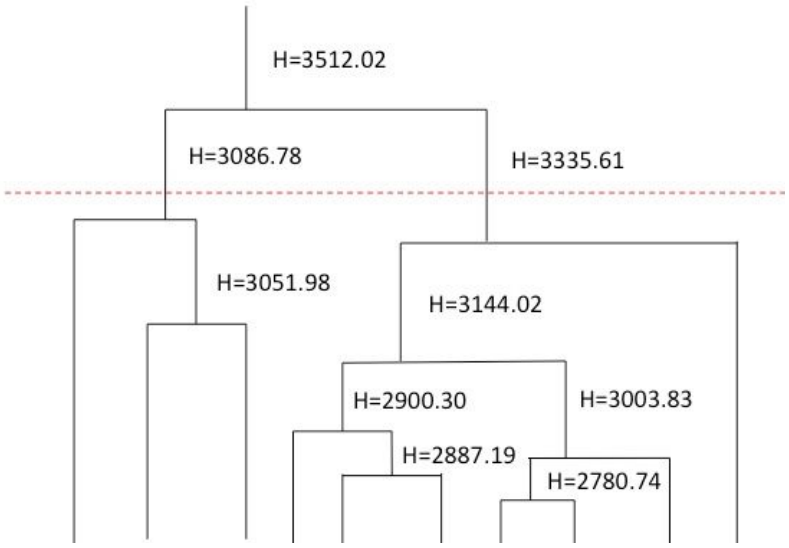
3. Hierarchical agglomerative clustering (HAC)

a. HAC Single Link



From the dendrogram above, we believe that 10 clusters will lead to a better fit for the data, because the dendrogram looks like a chain link where every one of the ten clusters consists of one cluster from below the visible tree. Because we know the data represents the ten digits 0 to 9, we are confident that K = 10 is a good choice for the data.

b. HAC Complete Link



From the dendrogram of complete link HAC, we may determine the number of clusters to be 2 (as cut by the red dashed line in the figure above) but we are not sure whether it is a good estimate. For this problem if we split the data into 2 clusters, where we can consider that one of the two clusters contains digits with edges and sharp corners (e.g. 1, 4, and 7) and the other cluster contains digit with rounded edges (e.g. 0, 2, 3, 5, 6, 8, and 9).