Name: Vigneshwaarar CR

CWID: A20392185

K-Means Clustering Implementation

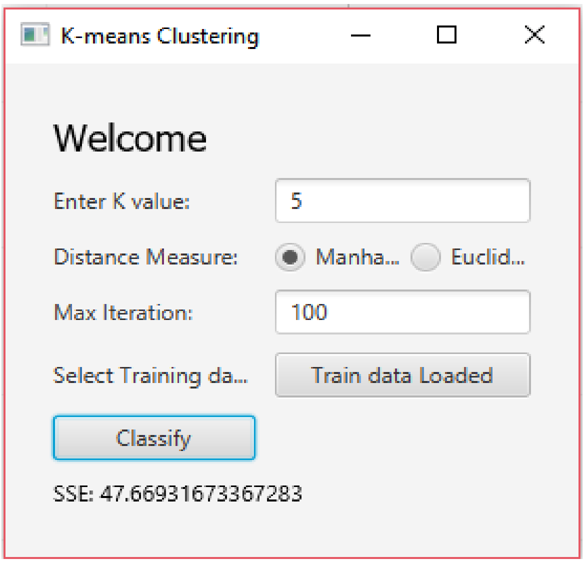
K-means Algorithm:

- Before we use the data to build the model, we normalize the dataset on an equal scale and discretization must be done on the classification features, other than the feature which we use for prediction.
- From the dataset, we select a random data points for each cluster. For example, if we consider 3 clusters, then, select a random point from dataset and consider as centroids for each cluster
- Now, we use any distance measure like, Manhattan distance, Euclidean distance for calculating the distance from each data point to the centroids. The data point with lesser distance to the centroid, will be oriented to the cluster. We iterate through all the data points
  - Manhattan distance = abs$|X_1-Y_1|$ + … + abs$|X_n-Y_n|$
  - Euclidean distance = sqrt$((X_1-Y_1)^2 + \ldots + (X_n-Y_n)^2)$
- For the next iteration, the centroids values will be the mean value of the data in each cluster
- We repeat this process until the clusters converge, i.e., the movement of data points between the clusters doesn't happen.
- We can also set max iteration limit and we can stop iterating and consider the clusters have converged.
- Now, we calculate SSE for the model. It is the sum of the square of the distance of the data point to the centroid of the cluster and we sum up for all the cluster. This is our SSE measure which shows how good our learning process is but know how useful our clustering is, it requires user intervention manually look at the clustering and make a conclusion

Execution steps:

- Run the .jar file
- Enter the number of clusters in K value field
- Select the distance measure to be considered
  - Manhattan distance
  - Euclidean distance
- Select dataset for clustering by clicking on file browser controls
- Click in classify button to see the SSE of the model for that preferred combination
- The clustered output file is saved to the folder path where the .jar exists

Sample input:



Welcome

Enter K value: 5

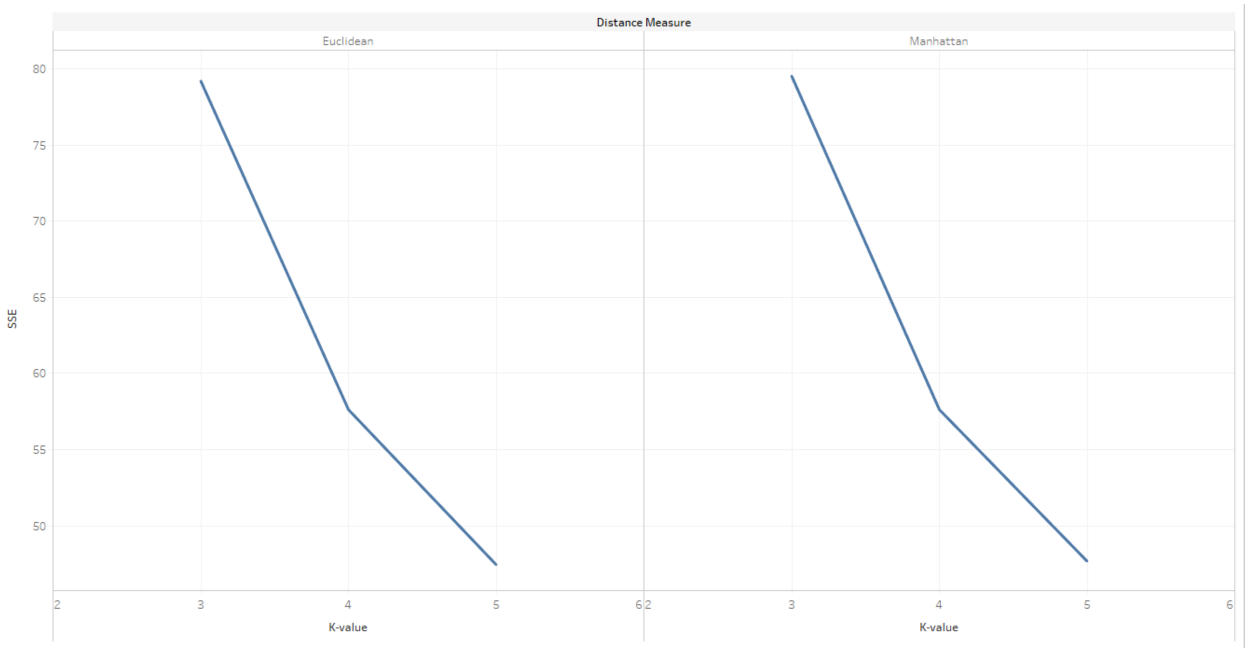
Distance Measure: ● Manha... ○ Euclid...

Max Iteration: 100

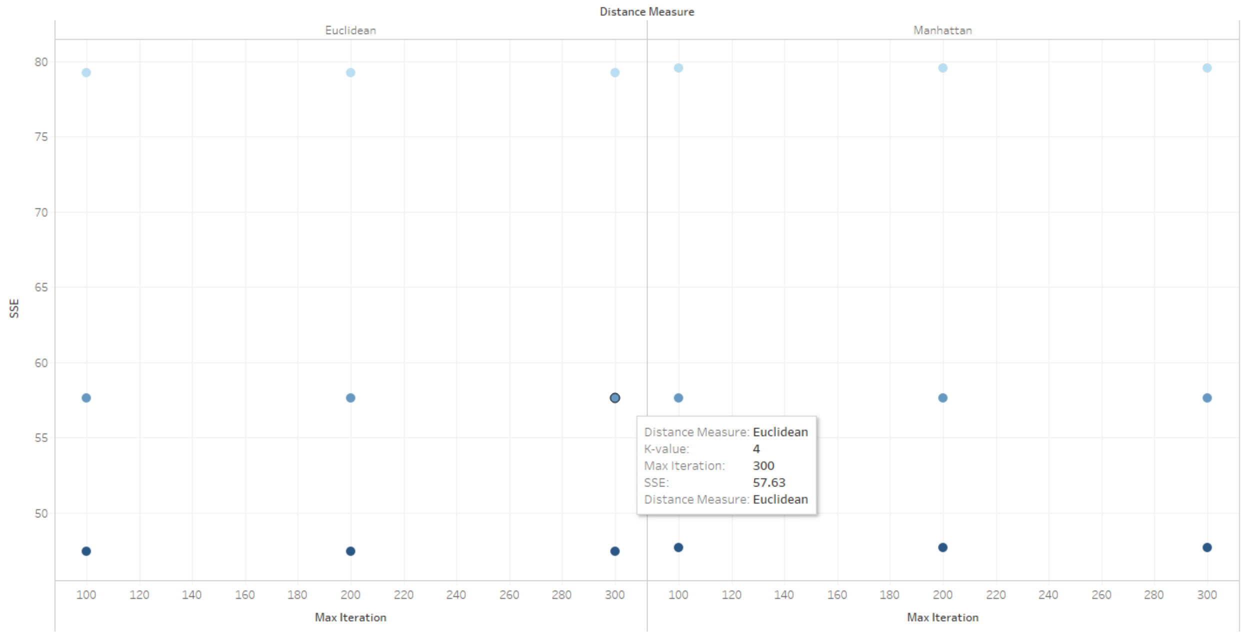
Select Training da... Train data Loaded

SSE: 47.66931673367283

SSE trend for different K-value and Distance Measures:



SSE trend for different K-value, Max iteration and Distance Measures:


Distance Measure
Euclidean
Manhattan
SSE
Max Iteration
Distance Measure: Euclidean
K-value: 4
Max Iteration: 300
SSE: 57.63
Distance Measure: Euclidean