CSE: 575 PROJECT 2

-Harshith Chittajallu, ASU ID- 1218707243

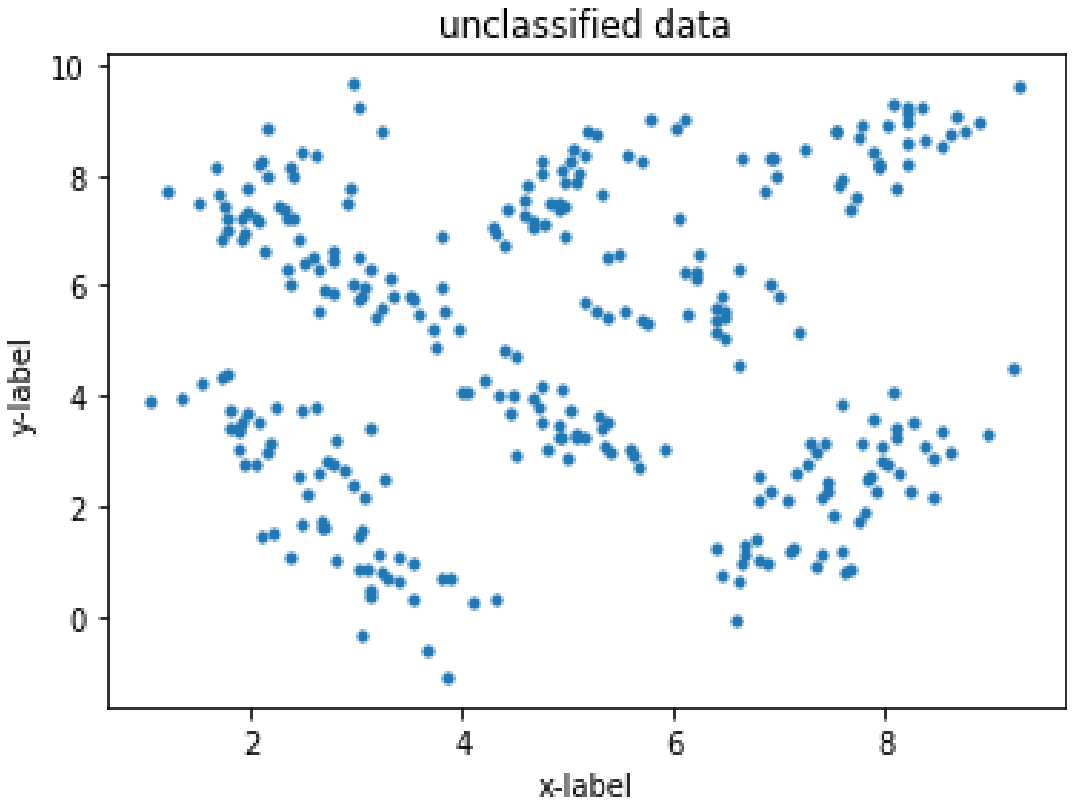
March 26, 2020

INTRODUCTION

In this project, we have 1 data set, containing 300 unlabeled 2-D points:

The data set has 300 rows and 2 columns, with each element being a real number. We cluster those data points accordingly to a ‘k’ number of cluster centers using 2 strategies and thus ‘classify’ them without the help of labels.

Here is how the data set looks, unlabeled, on a scatter plot:



PROCEDURE FOR K-MEANS, STRATEGY 1

The algorithm for k-means works in the following steps:

1. Take ‘k’ random cluster centers from the given set of points.
   Cluster Centers= $(\mu_{1\ to\ k}, \mu_{1\ to\ k})$
2. Compute the Euclidean distances of the points from each cluster center and ‘classify’ a point as belonging to a certain cluster center if its distance from that particular cluster center is minimal compared to other cluster centers.
   Euclidean distance= $\sqrt{(x_i - \mu_i)^2 + (y_i - \mu_i)^2}$

3. After classification, you must change the cluster centers to new values (which are basically the centroids of the points classified to that particular cluster)

Centroid or updated cluster center $(\bar{\mu}_x, \bar{\mu}_y)$ = $(\sum_1^n x_i \div n, \sum_1^n y_i \div n)$ where n is number of points belonging to a particular cluster

4. Update the values of the cluster centers and continue. If the case arises that no points are classified to a particular cluster center, do not calculate the centroid OR update the value of that particular cluster center.
5. After certain cycles, the cluster centers stop changing after each cycle. This is the point of convergence of the algorithm with optimal distance of the cluster centers to classify the clusters.

PROCEDURE TO CALCULATE THE COST FUNCTION AND ELBOW GRAPH:

1. After the k-means algorithms converges, compute the sum of squared distances of all points classified under a particular cluster center to that center
2. Add all of such computed sum of squared distances for each cluster

3. That gives the cost function of ‘k’ clusters. Note down all such cost functions for clusters starting from 2 to 10 so that we can plot a graph of the obtained cost functions to the corresponding ‘k’ number of clusters.
4. The x coordinate where the graph suddenly stoops low is known as the elbow point and that X coordinate gives us the optimal number of clusters required.

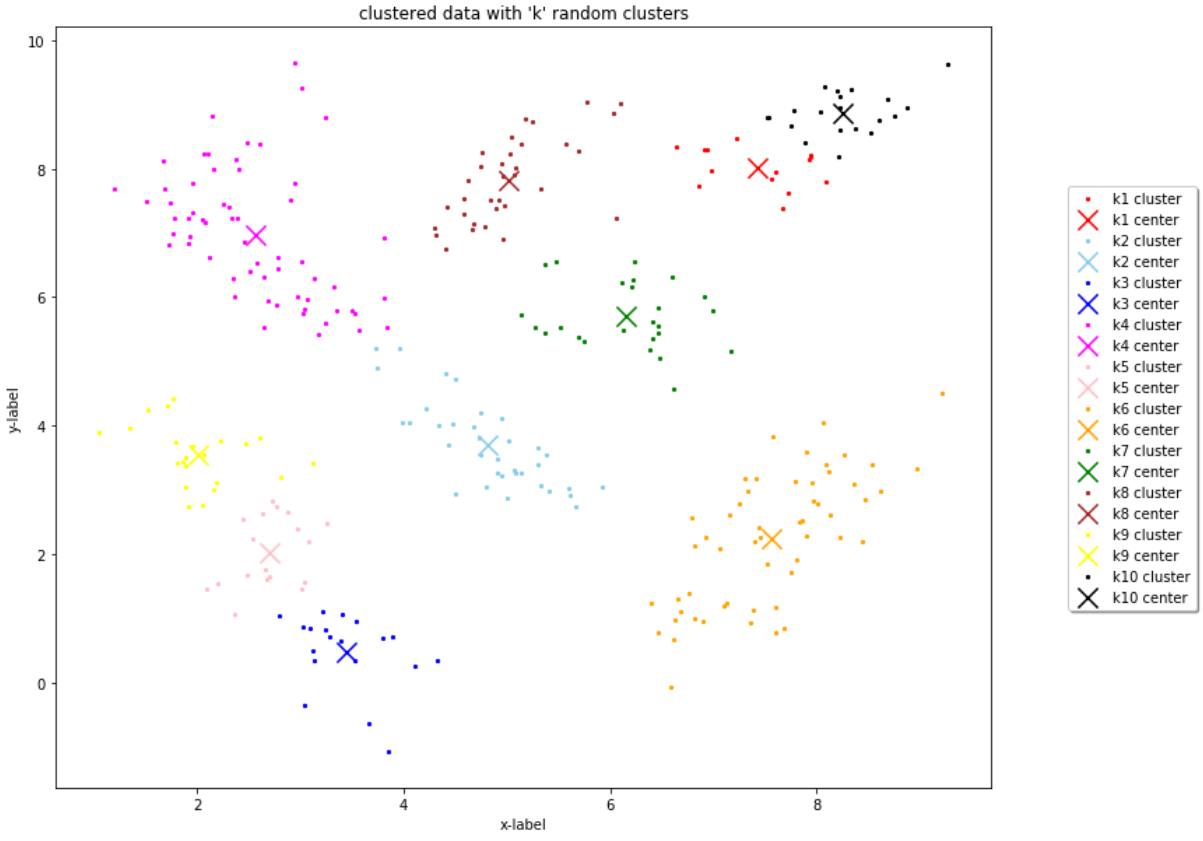
Procedure for K-means, Strategy 2: K++

1) In this strategy, instead of generating ‘k’ random cluster centers, we generate only 1 initial cluster center and generate the remaining cluster centers in the following steps:
2) After generating a random cluster center, we consider the next cluster center to be the maximum distance of a point from the initial cluster center.
3) From then on, the following cluster centers will be the maximum average distance of a point from all previous cluster centers
4) After generating the initial cluster centers, use the k means algorithm as mentioned in strategy 1 but using these cluster centers as initial cluster centers instead of random ones.
5) Doing so will reduce clustering errors

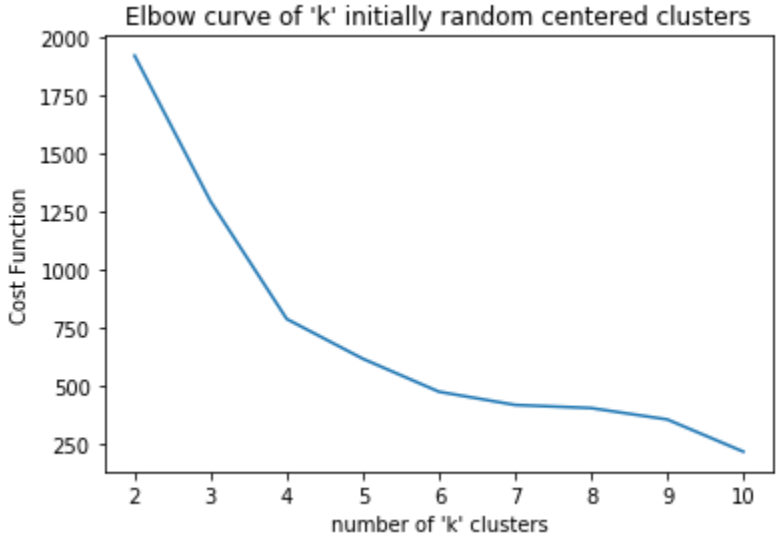
The procedure to find the cost function graph is the same as in strategy 1 except that we use the updated initial cluster centers in calculating the cost functions.

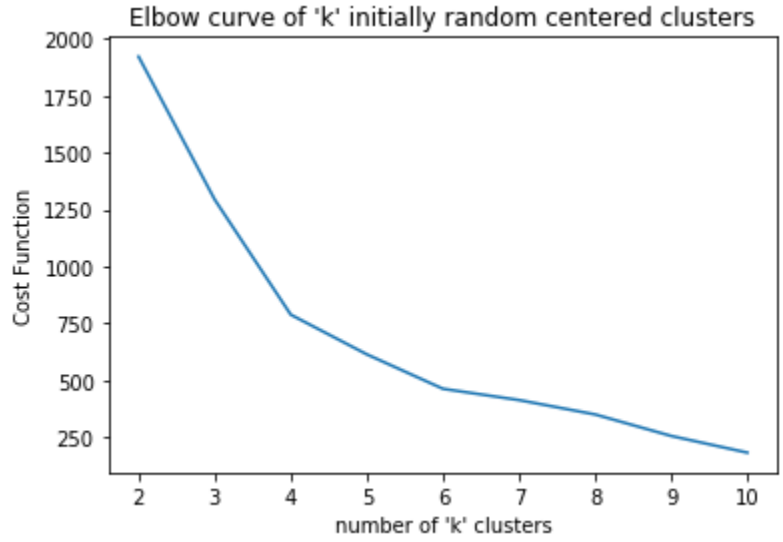
RESULTS

After running the code using strategy 1, this is our result for 10 clusters:



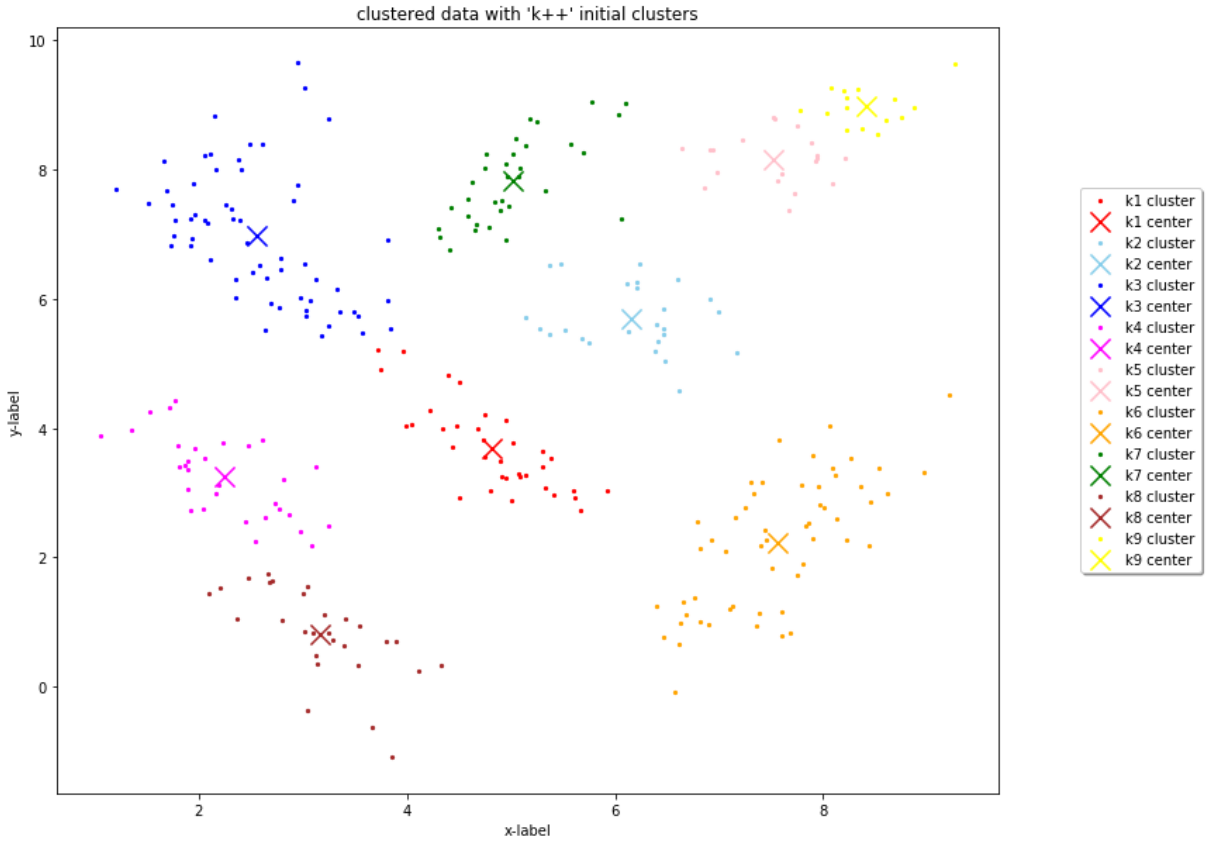
After running the code for calculating the cost function graph twice, these are the graphs obtained:



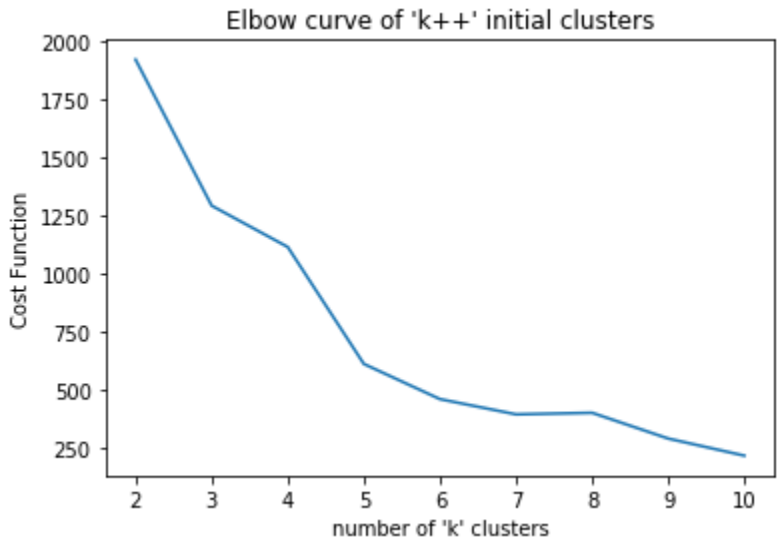


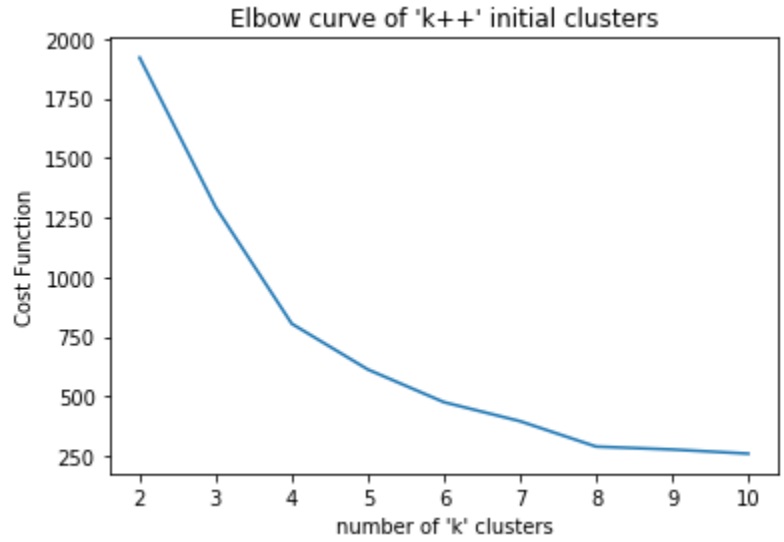
As we can see, in both cases, the graph stoops down sharply at k = 4. Hence, using strategy 1, the optimal number of clusters for the given data is 4.

After running the code using strategy 2, this is our result for 9 clusters:



After running the code for calculating the cost function graph using strategy 2 twice, these are the graphs obtained:





Here the result is more ambiguous. In one graph, the optimal number of clusters is 5 and in the other, it is 4. After running the code multiple times to find the average optimal output cluster, it turned out to be 4 as well.

Therefore, we can say that the optimal number of clusters for the given data set using K-means for both strategies is equal to 4.