Unsupervised Learning(K-Means)

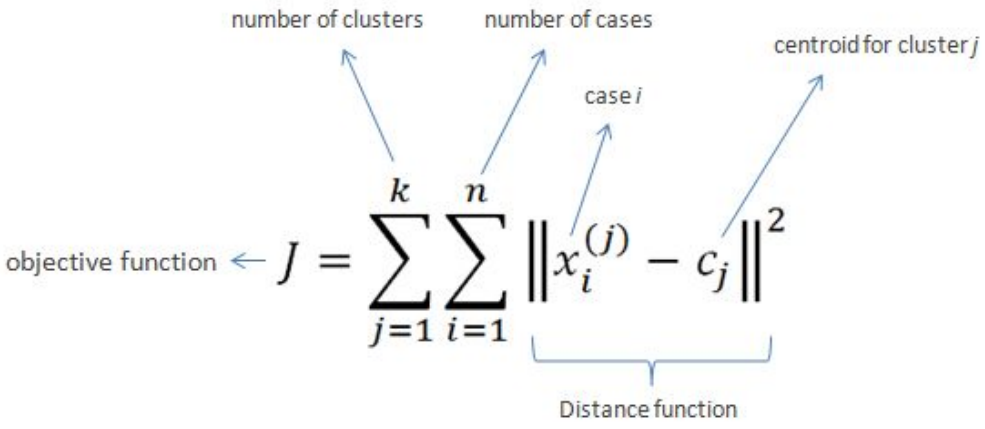
Aditya Das

K-means algorithm is an unsupervised learning algorithm which is used to divide a large data into different clusters of similar types. The objective here is to divide a N size dataset with P dimensional observations into K different clusters such that the sum of squared distances within the cluster is minimum. Since there can be many clusters involved here, it can come up with a huge number of cluster combinations. Thus, it might not be the optimal solution always. The resulting solution is highly dependent on the number of clusters chosen and the initial configurations set before running the algorithm. A pragmatic approach to finding a good solution is to compare the results of multiple runs with different K and choose the best one among them based on a predefined criterion. Generally, a large K would minimise the error to a considerable extent, however it poses the risk of overfitting.

Pseudocode of K-means Algorithm

```
1. Choose the number of clusters(K) and obtain the data points
2. Place the centroids c_1, c_2, ..... c_k randomly
3. Repeat steps 4 and 5 until convergence or until the end of a fixed
number of iterations
4. for each data point x_i:
     - find the nearest centroid(c_1, c_2 .. c_k)
     - assign the point to that cluster
5. for each cluster j = 1..k
     - new centroid = mean of all points assigned to that cluster
6. End
```

Objective function of K-means Algorithm



K-means algorithm is designed to minimize the above objective function. It is bound to decrease after every iteration till the stopcase is reached.

Initialization Strategies

1. Pick random K initial centers from the given sample
2. Pick the first center randomly; for the i-th center (i>1), choose a sample (among all possible samples) such that the average distance of this chosen one to all previous (i-1) centers is maximal.

Objective function Vs K Cluster graphical plots

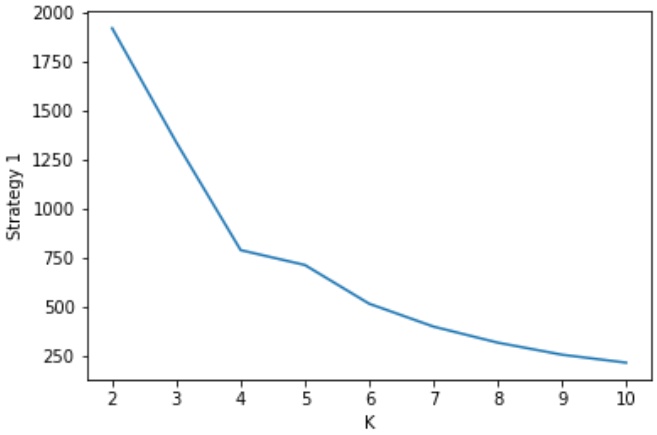


Fig 1. Objective function vs K plot for Strategy 1

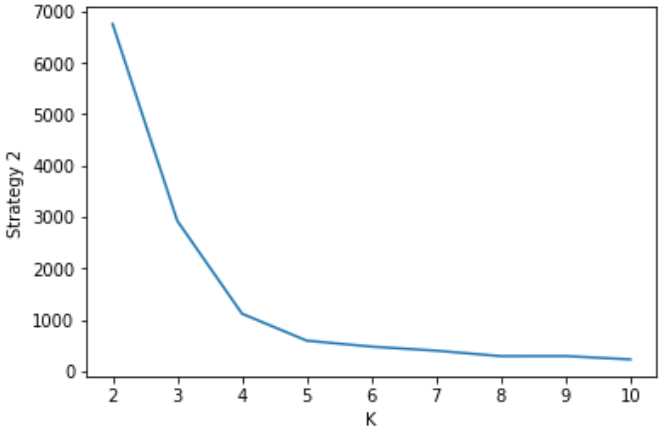


Fig 2. Objective function vs K plot for Strategy 2

Visualization of Clustered Points

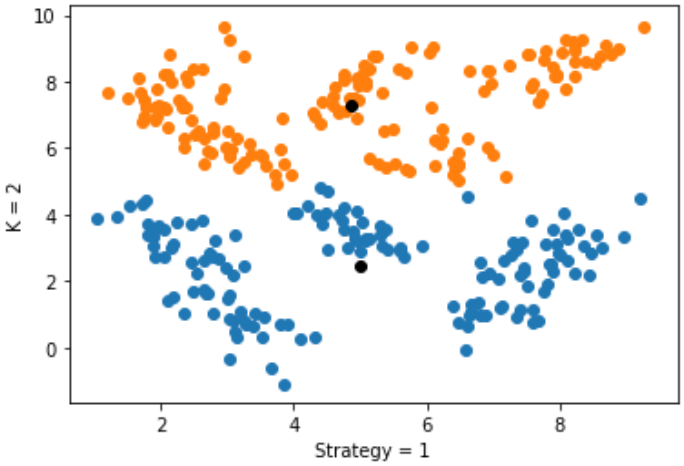


Fig 3. Scatter plot of clusters for K=2, Strategy 1

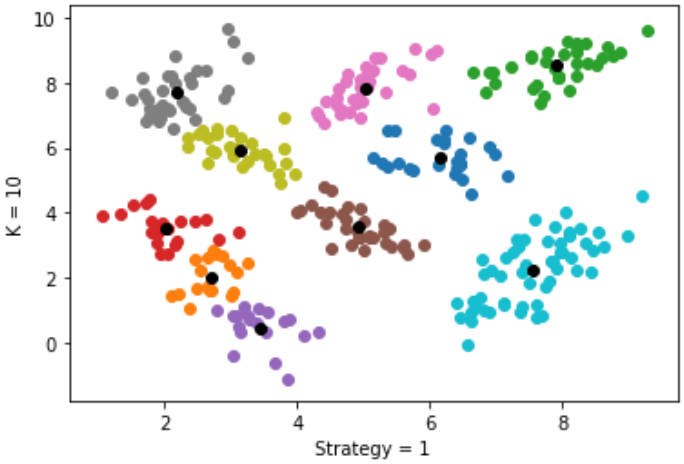


Fig 4. Scatter plot of clusters for K=10, Strategy 1

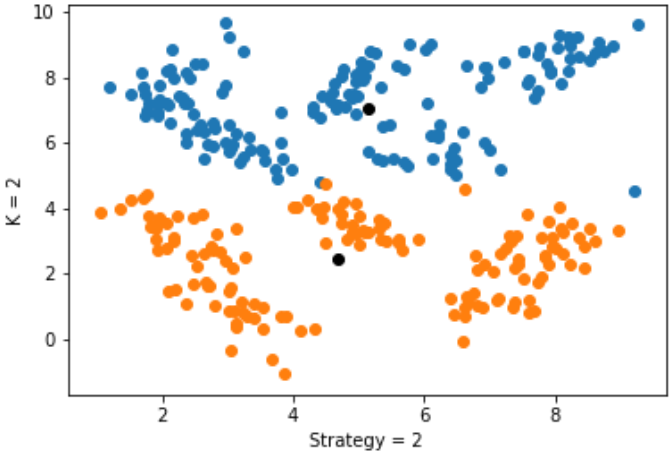


Fig 5. Scatter plot of clusters for K=2, Strategy 2

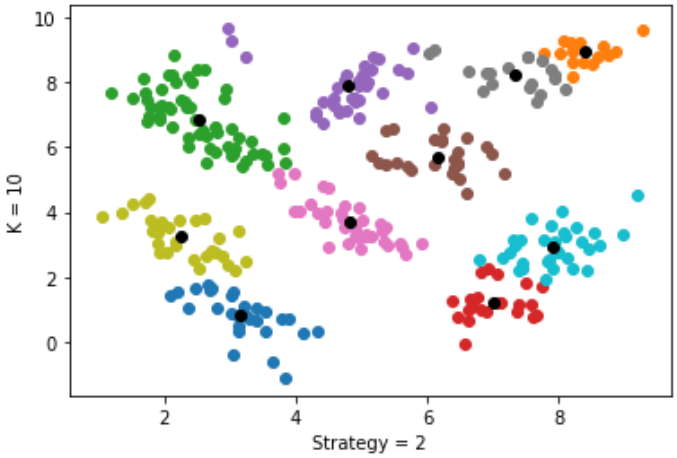


Fig 6. Scatter plot of clusters for K=10, Strategy 2

Analysis of both strategies

K-means algorithm in Strategy 1 starts with choosing the initial cluster centers randomly and then finding out the better solution in each consecutive iteration. K-means algorithm in Strategy 2, on other hand starts with one initial random cluster center and then searches for other centers given the first one. The second strategy may not necessarily perform better on all occasions because the subsequent centers chosen could be outliers. This would significantly impact the sum of squared errors in this case. So this strategy works well when we are sure that our dataset has less noise. It can also fail miserably in large dataset where the K-means could converge to one large big cluster. The first strategy is totally random. If the points selected are good, it can give us a very good result. On the contrary, it can also leave us with a bad result as well. Hence we can expect to see spikes in the graph in some cases. Its sum of squared errors may increase or decrease at any instant during the convergence process.