Unsupervised Learning (K-means)

CSE 575: Statistical Machine Learning
Tejaswi Paruchuri – 1213268054

1. Introduction:

The main aim of this project is to implement unsupervised learning using K-means algorithm on the given dataset of 2-D points with 2 different strategies of initializing centroids. With the two strategies these data points have to be clustered into k clusters with k ranging from 2 to 10. After the clustering is done objective function has to be calculated for each data point with the corresponding centroid for all the k. The resultant objective function values have to be plotted with the number of clusters k for both the strategies to summarize the results.

2. Dataset:

The given dataset contains 300 2-D data points which means the data will be an array of size 300 X 2 which will kind of represent the co-ordinates of the data points by giving the x and y co-ordinates of the data points.

The program is implemented using python 3.5 programming language. The program takes the parameter value as below while execution based on the requirement to plot scatter graphs or not. Plots of objective function will be plotted irrespective of parameter value input, but the scatter plot graphs will not be plotted if the parameter value of plot_scatters is not given as 'yes'. Default value of plot_scatters is No which means scatter plot graphs will not be plotted by default.

```
E:\Courses\SML\programming_assignment\2\submission>python3 main_Kmeans.py -h
usage: main_Kmeans.py [-h] [-plot_scatters No]

optional arguments:
  -h, --help         show this help message and exit
  -plot_scatters No  give Yes to plot scatter graphs else No

E:\Courses\SML\programming_assignment\2\submission>python3 main_Kmeans.py -plot_scatters yes
```

3. K-means Algorithm:

K-means algorithm is one of the mostly used algorithms for unsupervised learning. In unsupervised learning inferences must be made from the given dataset of unlabeled samples. In K-means we will have some fixed number of centroids (k) at the beginning and all the samples in the dataset will be clustered to a particular centroid based on the similarity of the sample to the centroid which is also called membership. This membership can be defined in multiple ways. One of the methods of defining membership is Euclidean distance which is

invariant to translation and rotation of the feature space. In K-means we consider hard membership i.e., one sample belongs to only one centroid. Given n data sample the goal of this algorithm is to optimize by minimizing (making it zero) the sum of squared error (i.e., for each cluster total distance between the sample and the centroid should be as small as possible). As the membership of sample is determined by distance to the means μ_i the task is to find the optimal set of { μ_i} which is NP-hard. But heuristic approaches can be used to quickly converge to local optimum.

K-means clustering:

Input: Given n data samples

Goal: Partition them into k clusters/sets D_i with respect to center/mean vector μ_1, μ_2, ... μ_k to minimize

$$\sum_{i=1}^{k} \sum_{x \in D_i} \|x - \mu_i\|^2$$

Comparing with the mixture models: Here hard assignment of membership to a sample is done (simply based on its distance from cluster center)

4. **Methodology:**

K-means Algorithm:

Step1: Initialize the centroids of k clusters (μ_1, μ_2, ... μ_k)

Step2: Begin

Membership assignment: Assign each sample in the data set to a cluster μ_i which is nearest to the data sample

Recomputing centroids: Recompute the centroids based on the samples in the new clusters using below formula

$$\mu_i = \frac{\sum_{x \in D_i} x}{n_i}$$

Repeat step 2 until no change in μ_1, μ_2, ... μ_k

Step3: Return μ_1, μ_2, ... μ_k

Below are the strategies used in initializing the centroids in step1:

Strategy1: Randomly selecting k centroids (μ_1, μ_2, ... μ_k) from the given dataset without replacement

Strategy2 Pick the first centroid μ_1 randomly. To choose μ_i select a sample such that the average distance of the sample from the remaining i-1 centroids is maximum

This algorithm has to be implemented for the values of k ranging from 2 to 10

Objective Function Calculation:

For the values of μ_1, μ_2, ... μ_k returned from K-means algorithm calculate objective function using the below formula for values of k ranging from 2 to 10

$$\sum_{i=1}^{k} \sum_{x \in D_i} \|x - \mu_i\|^2$$

5. Results:

5.1. Strategy1:

In this k centroids were selected randomly by selecting k samples (μ_1, μ_2, ... μ_k) from the given dataset without replacement. Objective function values for the k ranging from 2 to 10 were recording in 2 runs and below are the graphs and values for 2 runs.

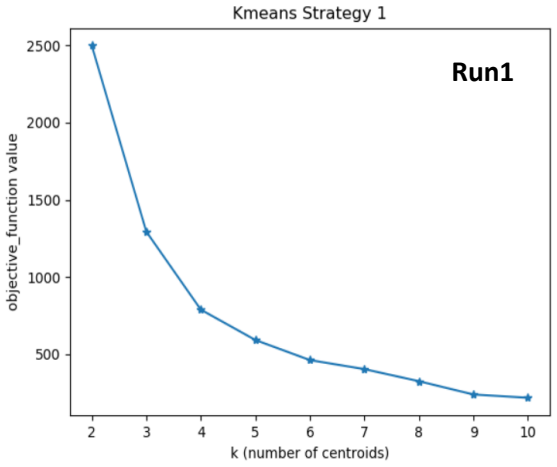

Fig: **Run 1**

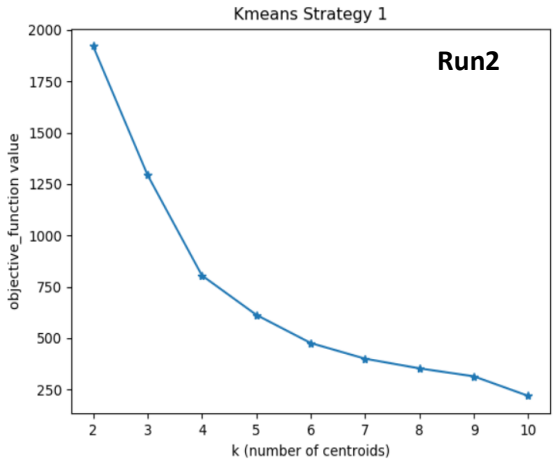

Fig: **Run 2**

Values of k	Values of objective function (Run 1)	Values of objective function (Run 2)
2	2497.990517122937	1921.033485856206
3	1294.2984174853177	1294.2984174853177
4	789.237972217795	805.116645747261
5	592.9375729660761	613.2824392056042
6	462.9263558248374	476.118751676353
7	404.3744685245436	399.8224706715241
8	326.2650293699702	352.6941527848963
9	240.3132887970991	314.0790106473261
10	219.01649528367935	218.70640669599916

5.2. Strategy2:

In this first centroid were selected randomly whereas remaining k-1 centroids were selected by selecting μ_i such that the average distance of the sample from the remaining i-1 centroids is maximum. Objective function values for the k ranging from 2 to 10 were recording in 2 runs and below are the graphs and values for 2 runs.

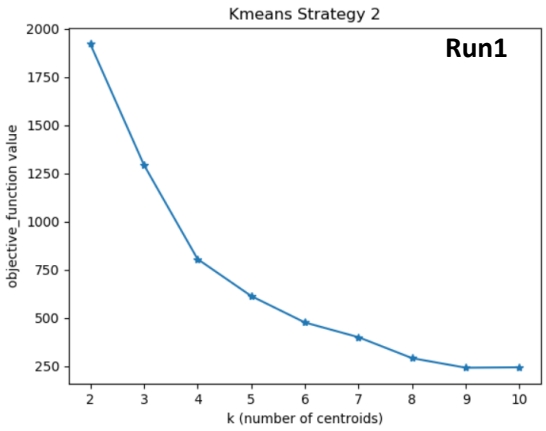

Fig: **Run 1**

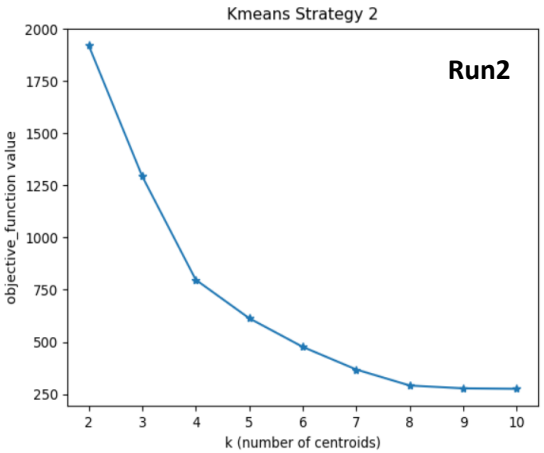

Fig: **Run 2**

Values of k	Values of objective function (Run 1)	Values of objective function (Run 2)
2	1921.033485856206	1921.033485856206
3	1293.7774523911348	1293.7774523911348
4	805.116645747261	797.960184078995
5	613.2824392056041	613.2824392056041
6	476.118751676353	476.118751676353
7	399.7003015793046	367.66584649464943
8	290.9243344744376	290.9243344744376
9	241.3719224573046	277.39143397662207
10	243.45056737269	275.19842718008664

6. Conclusion:

Using strategy1 and strategy2 K-means algorithm is implement with the number of clusters k ranging from 2 to 10. At 3 it was observed that the value of objective function is dropping dramatically indicating that k=3 is the optimal number of clusters for the given data set.