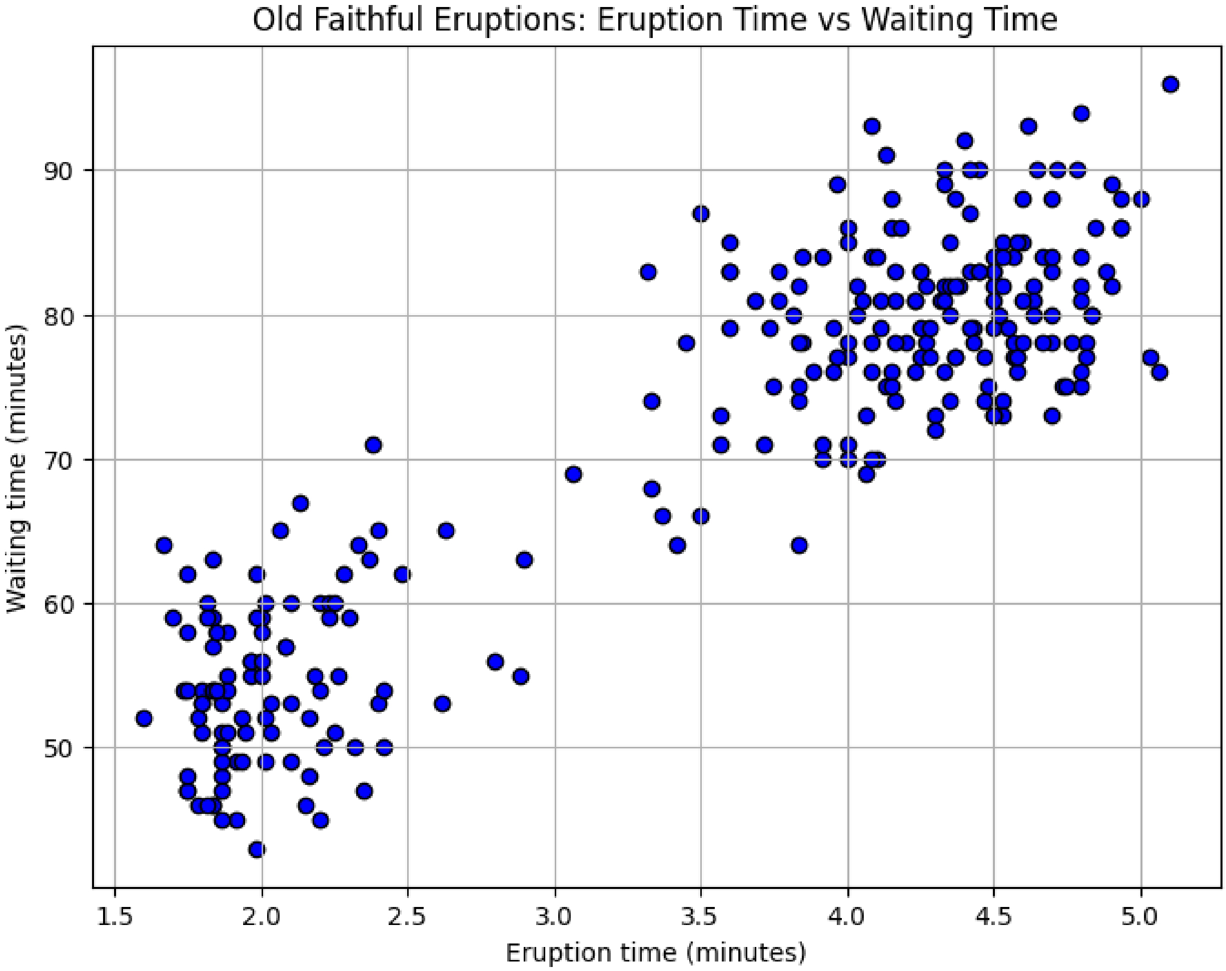
IS 733 Homework 3 **Reena Kale XB96457**

Problem 1 (25 points)

(a) Create and print out a scatter plot of this dataset, eruption time versus waiting time. (10 points)

Ans: Based on the faithful dataset provided, the scatterplot is generated using python code



The python code to generate the above scatter plot for faithful dataset has been uploaded to the GitHub repository. Please find the GitHub link below to get the code and scatter plot: **https://github.com/reenakale3109/IS_733_Homework3/blob/main/IS_733_HW_3_XB96457.ipynb**

(b) How many clusters do you see based on your scatter plot? For the purposes of this question, a cluster is a "blob" of many data points that are close together, with regions of fewer data points between it and other "blobs"/clusters. (5 points)

Ans:

From the above scatterplot, we can clearly see that there are “2” major blobs or clusters. These two clusters have significant space between them and appear to be distinct from each other. From the two clusters we can clearly draw a few conclusions such as;
One Cluster namely Cluster 1 is located at the starting of the axis or plot, i.e. this cluster has Shorter Waiting Time (Starting from around 35-40 mins to 65-70 mins) and even has shorter duration of Eruption time as well (Approximately from 1.5 min to 2.5 mins). Whereas the second Cluster, namely Cluster 2, is located a bit far away from the start of the axis and has a significantly longer time. Cluster 2 data points have Longer Waiting time (i.e. Approximately from 70 mins to 90 mins) and even have longer duration of Eruptions (Roughly from 3.5 mins to 5 minutes). Both clusters have a significant distance between them, which makes them two separate clusters.

(c) Describe the steps of a hierarchical clustering algorithm. Based on your scatter plot, would this method be appropriate for this dataset? (10 points)

Ans:

Hierarchical Clustering Algorithm is simply an unsupervised learning technique which is used to group similar data points into clusters or blobs based on factors such as how much they are alike or how close they are to each other. (Distance calculated between data points using methods like Euclidean distance).

Hierarchical Clustering creates a hierarchy of several clusters, i.e. nesting of clusters together to form a family of clusters or tree-like structure of clusters. This tree-like structure of clusters is called “Dendrogram”.

Hierarchical Clustering algorithm follows two types of approaches to creating clusters basically, which are: -
1.) Agglomerative (Bottom – Up Most Commonly Used)
2.) Divisive (Top – Bottom)

In the Agglomerative approach we generally follow the following steps, firstly we identify each data point as an individual cluster. Then find the next closest cluster (this can be done by using distance calculation methods like Euclidean). Once it is found, they are merged into a single cluster. This process continues until they all merge into a single cluster and the formed dendrogram is cut to the required level to get the desired number of clusters.
Coming to Divisive method, it is a kind of opposite to the above one, first consider all the data points as one cluster. Then recursively cut the cluster into smaller clusters based on distances between each other. Continue this process until each data point is its own cluster or there is a stop condition.
From the scatterplot obtained for the data set, we can clearly conclude that Yes, Hierarchical Clustering is a very appropriate method for this data set. Cause we can clearly see that there

are 2 separate blobs or clusters, and hierarchical clustering works well for separated clusters. And another added advantage is that there is no need to specify the number of clusters in advance.
https://github.com/reenakale3109/IS_733_Homework3/blob/main/IS_733_HW_3_XB96457.ipynb

Problem 2 (75 points)

Implement the k-means algorithm in Python and use it to perform clustering on the Old Faithful dataset. Use the number of clusters that you identified in Problem 1. Be sure to ignore the first column, which contains instance ID numbers. In your notebook, including the following items:

(a) Your source code for the k-means algorithm. You need to implement the algorithm from scratch. (45 points)

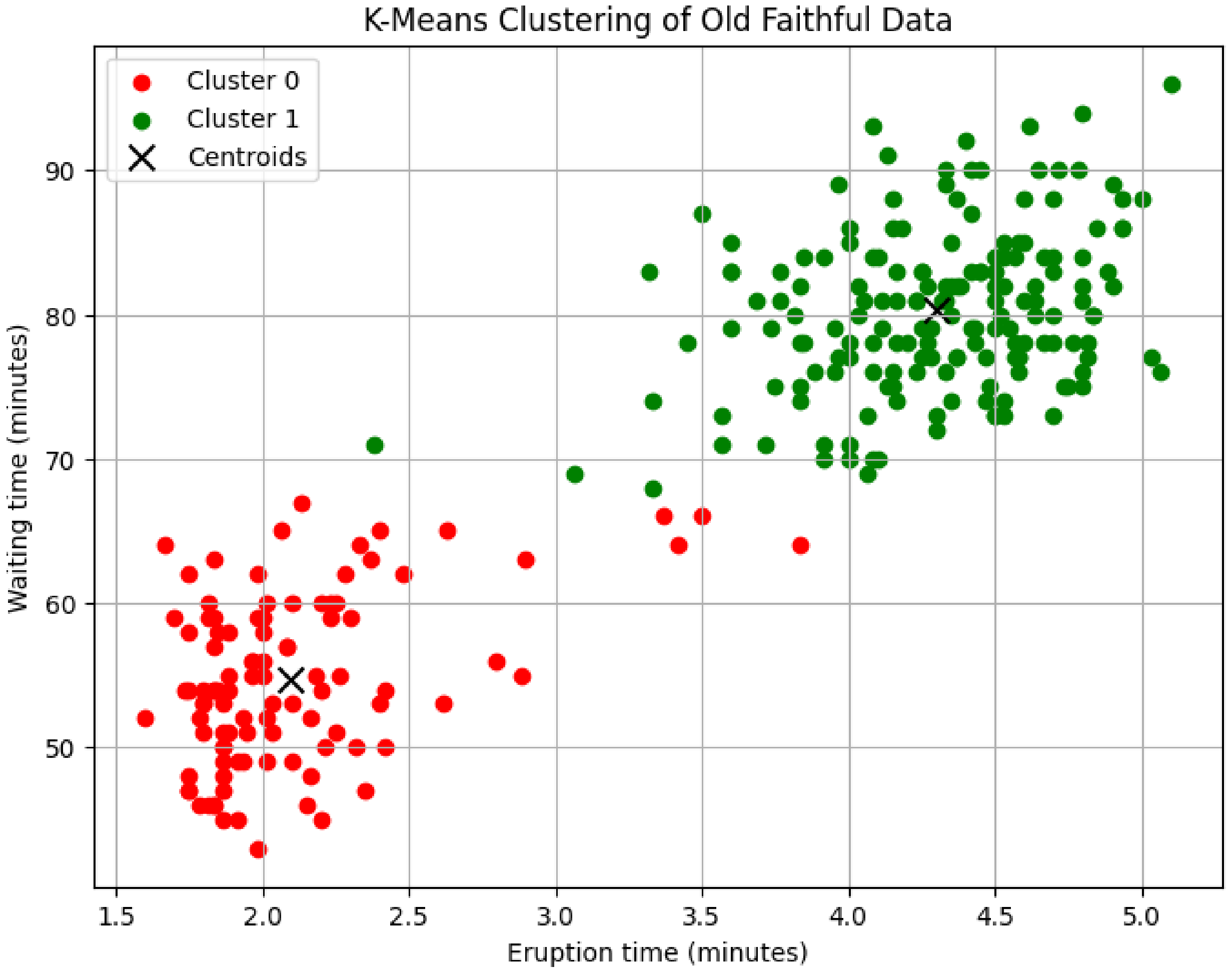
The output after running the k-means algorithm would be as follows.

```
K-Means converged in 4 iterations.
Final Cluster Centroids:
Cluster 1: Eruption = 2.09 mins, waiting = 54.75 mins
This is Cluster 1 which has Shorter Eruption and Waiting Times.
Cluster 2: Eruption = 4.30 mins, waiting = 80.28 mins
This is Cluster 2 which has Longer Eruption and Waiting Times.
Points per Cluster:
Cluster 1: 100 points
Cluster 2: 172 points

Final Inertia (Sum of Squared Distances): 8901.77
Sample points from each cluster:
Cluster 1 examples:
   eruptions  waiting
1     1.800       54
3     2.283       62
5     2.883       55
8     1.950       51
10    1.833       54

Cluster 2 examples:
   eruptions  waiting
0     3.600       79
2     3.333       74
4     4.533       85
6     4.700       88
7     3.600       85
```

(b) A scatter plot of your final clustering, with the data points in each cluster color-coded, or plotted with different symbols. Include the cluster centers in your plot. (10 points)

Ans:

The scatter plot of final clustering with k=2 would be as shown in the below diagram



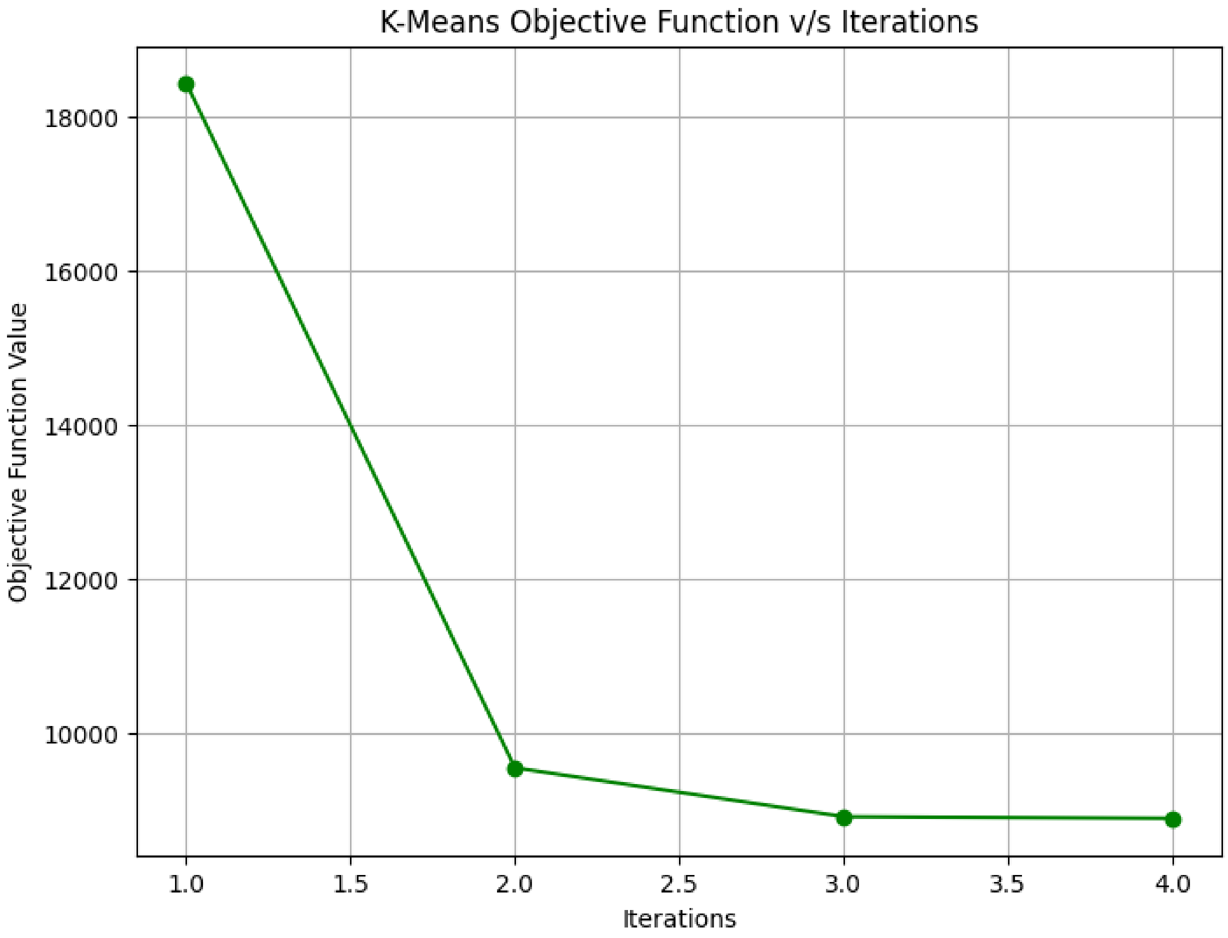
Code link for scatter plot: **https://github.com/reenakale3109/IS_733_Homework3/blob/main/IS_733_HW_3_XB96457.ipynb**

(c) A plot of the k-means objective function versus iterations of the algorithm. Recall that the objective function is (10 points)

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} \|p - c_i\|^2 \ ,$$

where k is the number of clusters, C_i is the set of instances assigned to the ith cluster, and c_i is the cluster center for the ith cluster. Note that the objective function should always decrease. If this is not the case, look for a bug in your code.

Ans:



Code for the same: **https://github.com/reenakale3109/IS_733_Homework3/blob/main/IS_733_HW_3_XB96457.ipynb**

(d) Did the method manage to find the clusters that you identified in Problem 1? If not, did it help to run the method again with another random initialization? (10 points)

Ans:

Yes, the method managed to find the clusters that are identified in Problem 1.

In Problem 1 we have identified 2 Clusters, i.e. Cluster 1 with Shorter Eruption and Waiting times, and Cluster 2 with Longer Eruption and Waiting times. By identifying the centers or centroids of the clusters, the separation can be clearly seen which is;

For Cluster 1: - Eruption time ≈ 1.5 minutes and Waiting Time ≈ 55 minutes
For Cluster 2: - Eruption time ≈ 4.5 minutes and Waiting Time ≈ 80 minutes

These values closely match the values observed in problem 1, and in case the k-means algorithm with given value of k=2 did not work, then in that case k-means algorithm can be run again with another random value. K-means is very sensitive to initial placement of the centroids, which are chosen randomly. If in case, it did not find the correct clusters in the first go, then the algorithm is re-run with different values and multiple initializations help in these types of cases. But in the above case, it is obvious from the scatterplot that there are only 2 clusters or blobs, and these two clusters are well separated. That means, K-means will be always converging to the right answer, even if it is initialized with some random value during the start. Hence, in the above case k-means algorithm successfully identified the clusters that were found out in problem 1