**CSE 6363: Machine Learning**

**Project 3: Report**

**About the dataset:**

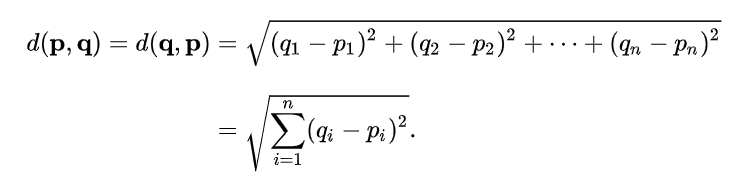
The dataset being used is the ​Iris Flower Dataset​, which is available for download for free at <http://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data>. This dataset is one of the best-known datasets used for statistical classification techniques.

The dataset consists of 50 samples from 3 species of Iris, namely: ***​Iris virginica***, ***​Iris setosa*** and ***Iris versicolor***. There is a total of 150 records in the dataset under 5 attributes. The attributes measured and collected from each species are the sepal length, sepal width, petal length, petal width in centimeters, and the class of the species.

**Method:**

The algorithm that is applied to this is the **K-Means Clustering** algorithm, which is one of the most popular unsupervised learning algorithms. Unsupervised learning algorithms are those which do not require labelled datasets to make inferences.

K-Means works on the principle of grouping together similar data points according to a similarity measure such as Euclidean-based distance or correlation-based distance.



Interestingly, simply to set the number of iterations to a fixed value (say, 10 or 20) is among reasonable ways. K-means is dedicated to being a fast method, therefore if you want a convergence criterion to be checked after every iteration that criterion must be easy/fast to compute.

**Implementation:**

This model has been implemented using Python 3.6 The code consists of following functions:

* Kmeans\_clustering()
* Centroid\_init()
* Paired\_distance()
* Kmeans\_clustering function is the main function, which is responsible for initializing cluster centroids and calculating the Euclidean distance by calling the respective functions.
* Centroid\_init function initializes random centroid points.
* Paired\_distance function calculates the Euclidean distance and returns it back to the Kmeans\_clustering function.

We follow the following steps:

* Select K points as the initial centroids.
* Repeat until the centroids become constant
  + Form K clusters by assigning all points to the closest centroid.
  + Recompute the centroid of each cluster.

We also calculate the accuracy and the error for the clustering process.

**Note:**

Python libraries used in this model are:

* Numpy
* Pandas
* Math

**Results**

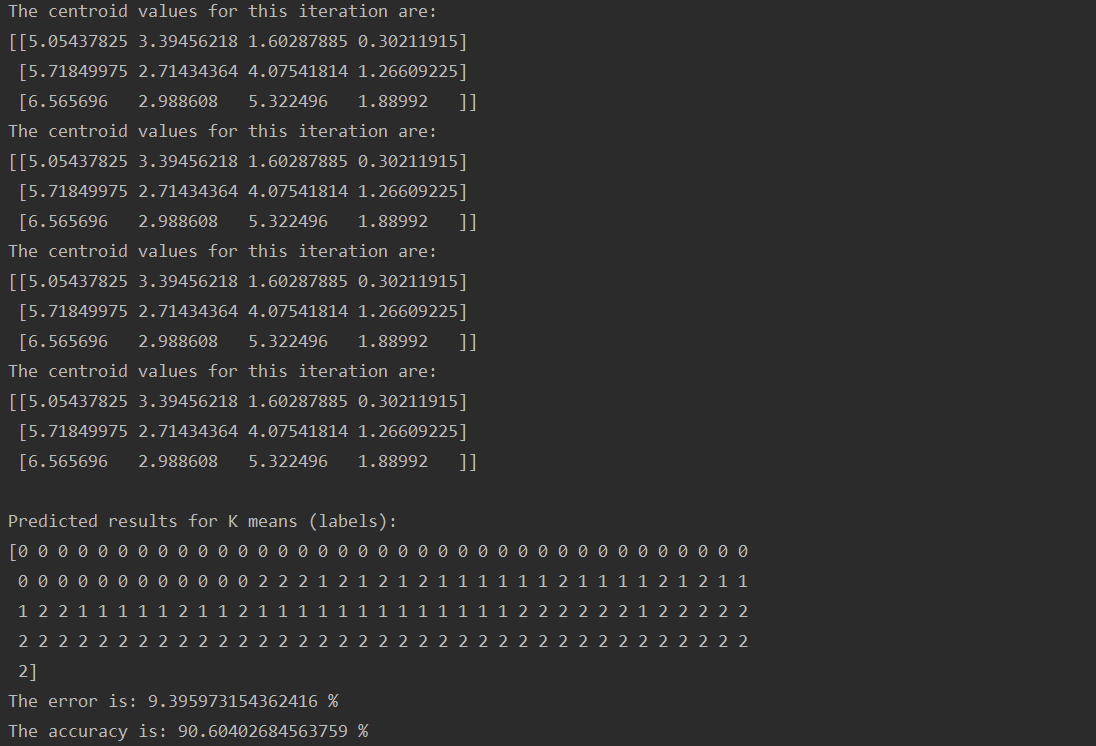
To select the value of K, I have used the Elbow method, which is the best way to select the value of K. I have selected the value for K as K = 3 and ran the code for different number of iterations. The results are as follows:

|  |  |  |
| --- | --- | --- |
| **Sl. No.** | **Iteration No.** | **Accuracy** |
| 1 | 3 | 85.235% |
| 2 | 5 | 89.933% |
| 3 | 10 | 90.604% |

Since the accuracy for the trial with 10 iterations is better than the other two, I have selected that as the final number of iterations.

The **final error** for K = 3 and number of iterations = 10 is found to be **9.395%.**

The **final accuracy** for K = 3 and number of iterations = 10 is found to be **90.604%.**



**References:**

1) <https://towardsdatascience.com/k-means-clustering-algorithm-applications-evaluation-methods-and-drawbacks-aa03e644b48a>  
2) <http://guidetodatamining.com/chapter8/>

3) Stackoverflow

4) Wikipedia: <https://en.wikipedia.org/wiki/K-means_clustering>

5) <https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1>

6) <http://madhugnadig.com/articles/machine-learning/2017/03/04/implementing-k-means-clustering-from-scratch-in-python.html>

7) <https://stats.stackexchange.com/questions/261836/k-means-how-many-iterations-in-practical-situations>

8) https://www.kaggle.com/ranjan42/use-of-elbow-technique-k-means-iris-dataset