Assignment 9

@author: @ruhend (Mudigonda Himansh)

AP19110010169

K-Mean algorithm

K-Mean algorithm is an unsupervised algorithm that solved the clustering problem. Data points inside a cluster are homogeneous and outside the cluster are heterogeneous.

Determining the value of 'K'

In k-means, we ahve clusters and each cluster has its own centeroid. Sum of square of difference between centroid and the data points within a cluster constitutes within sum of square value for that cluster.

And also, when the sum of square values for all clusters are added, it becomes total within sum of square value for the cluster solution.

The optimum number for k is taken when the chang in slope of the value doesnt change much with increase in the number of clusters.

In [1]:

```
import pandas as pd
from sklearn.cluster import KMeans
```

In [2]:

```
train_data = pd.read_csv('train-data.csv')
test_data = pd.read_csv('test-data.csv')
```

In [3]:

```
print('Head of the training dataset  : ', train_data.head())
print('Head of the testing dataset   : ', test_data.head())
```

```
Head of the training dataset  :     Age  Annual Income (k$)  Spending
Score (1-100)  Genre_Female  Genre_Male
0   30                  34                      73             1
0
1   36                 103                      85             1
0
2   54                 101                      24             1
0
3   28                 101                      68             0
1
4   24                  39                      65             1
0
Head of the testing dataset   :     Age  Annual Income (k$)  Spending
Score (1-100)  Genre_Female  Genre_Male
0   53                  46                      46             0
1
1   22                  17                      76             1
0
2   35                  24                      35             0
1
3   32                 137                      18             0
1
4   31                  43                      54             1
0
```

In [4]:

```
print('Shape of the training dataset  : ', train_data.shape)
print('Shape of the testing dataset   : ', test_data.shape)
```

```
Shape of the training dataset  :  (100, 5)
Shape of the testing dataset   :  (100, 5)
```

In [5]:

```
model = KMeans()
model.fit(train_data)
```

Out[5]:

```
KMeans()
```

In [6]:

```
print('Default number of clusters      : ', model.n_clusters)
```

```
Default number of clusters      :  8
```

In [7]:

```
predict_train = model.predict(train_data)
print('Cluster on train data           : ', predict_train)
```

```
Cluster on train data           :  [4 0 3 0 4 7 7 0 0 2 5 5 1 3 0 5 3
6 7 4 6 1 1 3 6 3 5 5 3 1 6 1 1 3 3 5 2
 5 2 1 3 2 5 5 3 1 1 3 0 5 1 3 7 5 3 5 7 2 4 6 1 4 7 3 1 3 6 1 1 5 7
5 4 7
 0 0 4 1 7 1 7 1 5 7 2 7 3 1 2 1 6 0 5 3 5 7 1 4 5 4]
```

In [8]:

```
model_n3 = KMeans(n_clusters = 3)
model_n3.fit(train_data)
```

Out[8]:

```
KMeans(n_clusters=3)
```

In [9]:

```
print('Number of Cluster               : ', model_n3.n_clusters)
```

```
Number of Cluster               :  3
```

In [10]:

```
predict_train_3 = model_n3.predict(train_data)
print('Clusters in the training dataset : ', predict_train_3)
```

```
Clusters in the training dataset :  [0 2 1 2 0 1 0 2 2 0 2 2 0 1 2 2
1 0 0 0 0 0 0 1 0 1 2 2 1 0 0 0 0 1 1 2 0
 2 0 0 1 0 2 2 1 0 0 1 2 2 0 1 0 2 1 2 0 0 0 0 0 0 0 1 0 1 0 0 0 2 1
2 0 0
 2 2 2 0 2 0 0 0 2 0 0 0 1 0 0 0 0 2 2 1 2 0 0 0 2 0]
```

In [11]:

```
predict_test_3 = model_n3.predict(test_data)
print('Clusters in the testing dataset  : ', predict_test_3)
```

```
Clusters in the testing dataset  :  [0 0 0 1 0 0 1 0 0 0 0 0 0 1 1 0
0 0 0 2 1 1 0 0 0 0 1 0 0 1 2 0 0 0 0 1 2
 2 0 1 1 0 0 1 0 0 0 2 0 0 0 0 0 2 0 0 2 0 0 0 0 0 0 2 2 0 2 0 0 0 2
0 1 2
 0 1 0 1 0 2 0 0 0 1 0 0 0 0 0 1 0 0 0 0 1 0 0 0 2 1]
```