LAB ASSIGNMENT – 9

K-Means Clustering Algorithm

NAME : PRATHAPANI SATWIKA

REG.NO. : 20BCD7160

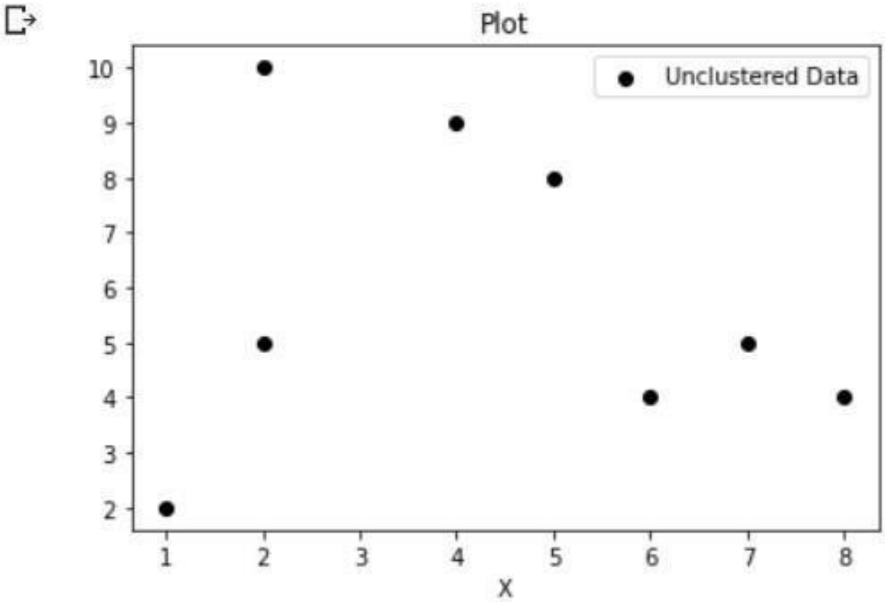
1.Develop k-Means Clustering algorithm to apply clustering on the following data objects referred by (x, y) pair: (k =3) A1(2, 10), A2(2, 5), A3(8, 4), A4(5, 8), A5(7, 5), A6(6, 4), A7(1, 2), A8(4, 9) Use Euclidian distance metric to determine closest centroid.

CODE AND OUTPUT :

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import random as rd
```

```python
X = np.array([[2,10],[2,5],[8,4],[5,8],[7,5],[6,4],[1,2],[4,9]])
```

```python
[3] plt.scatter(X[:,0],X[:,1],c='Black',label='Unclustered Data')
    plt.xlabel('X')
    plt.legend()
    plt.title('Plot')
    plt.show()
```



```python
m=X.shape[0]
n=X.shape[1]
n_iter=50
k=3
```

[9]
```python
centroids = np.array([]).reshape(n,0)
for i in range(k):
    rand = rd.randint(0,m-1)
    centroids = np.c_[centroids,X[rand]]
centroids
```

```
array([[ 5.,  7.,  2.],
       [ 8.,  5., 10.]])
```

[11]
```python
output={}
for i in range(n_iter):
    dist = np.array([]).reshape(m,0)
    for j in range(k):
        tempdist = np.sum((X - centroids[:,j])**2,axis=1)
        dist = np.c_[dist,tempdist]
    C = np.argmin(dist,axis=1) + 1
```

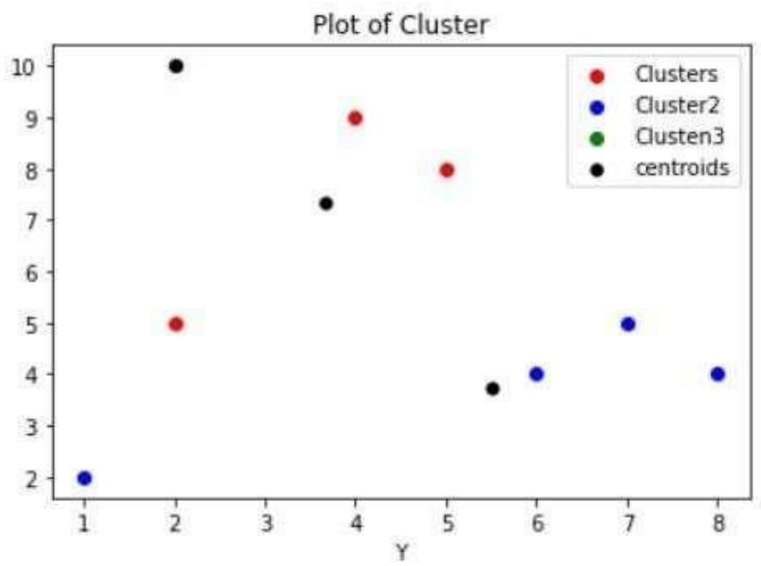
[12]
```python
Y={}
for j in range(k):
    Y[j+1] = np.array([]).reshape(2,0)
for i in range(m):
    Y[C[i]] = np.c_[Y[C[i]],X[i]]
for j  in range (k) :
    Y[j +1] = Y[j+1].T
```

[13]
```python
for j in range ( k) :
    centroids[:,j] = np.mean(Y[j+1],axis=0)
output = Y
```

[15]
```python
print('Outputs for iterations:',i)
print(output)
```

```
Outputs for iterations: 7
{1: array([[2., 5.],
       [5., 8.],
       [4., 9.]]), 2: array([[8., 4.],
       [7., 5.],
       [6., 4.],
       [1., 2.]]), 3: array([[ 2., 10.]])}
```

```python
colour=['red','blue','green']
labels=['Clusters','Cluster2', 'Clusten3']
for i in range(k):
    plt.scatter(output[i+1][:,0],output[i+1][:,1],c=colour[i],label=labels[i])
plt.scatter(centroids[0,:],centroids[1,:],s=30,c='Black',label= 'centroids')
plt.xlabel('X')
plt.xlabel('Y')
plt.legend()
plt.title('Plot of Cluster')
plt.show()
```



2. Load IRIS data set (IRIS.csv) - Remove Class Label column from IRIS data set – Apply developed kMeans clustering in Question 1 on the unlabelled IRIS data set with k.

CODE AND OUTPUT :

```python
import seaborn as sns
```

```python
[22] df = pd.read_csv("iris.csv")
df.head()
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

```
np.unique(df['Species'])
```

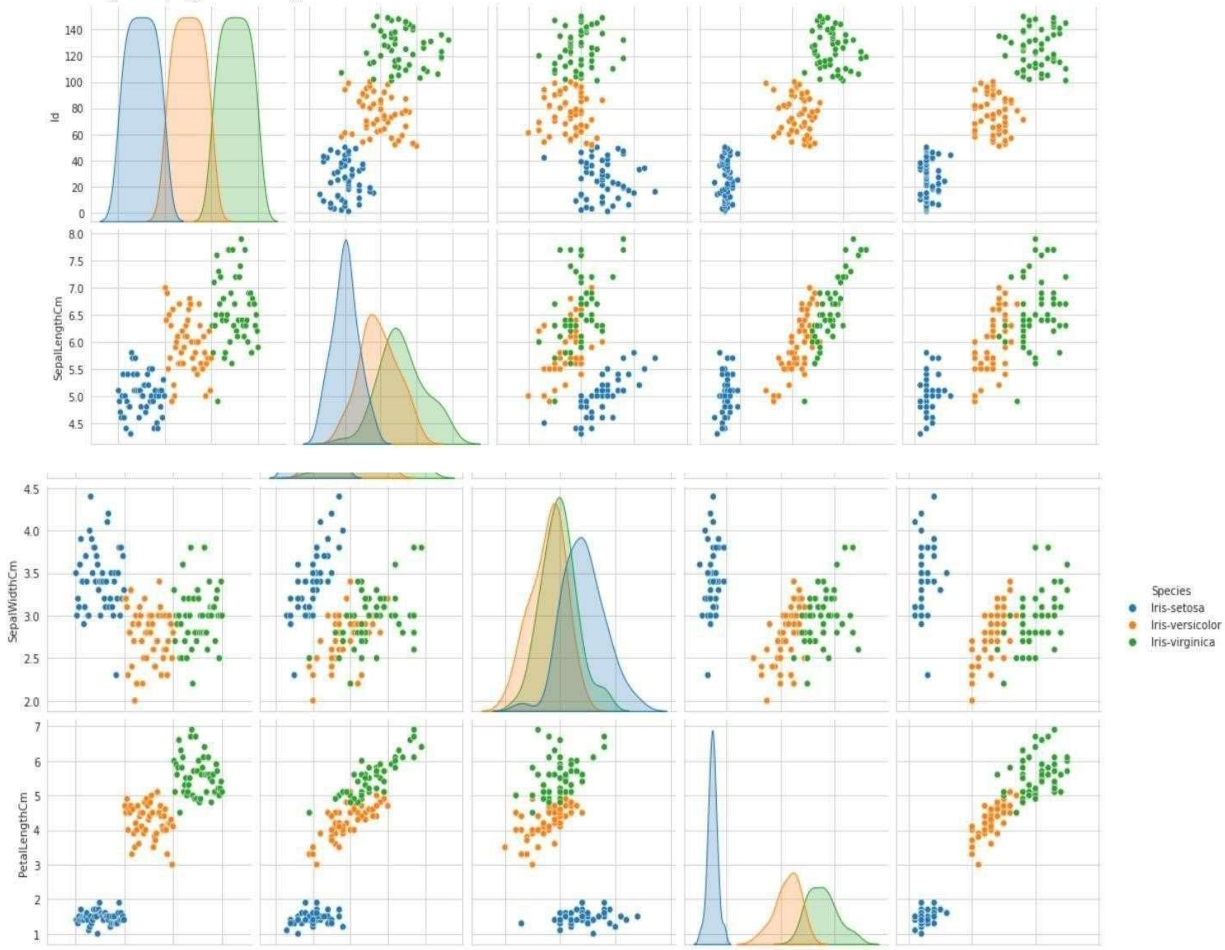
```
array(['Iris-setosa', 'Iris-versicolor', 'Iris-virginica'], dtype=object)
```

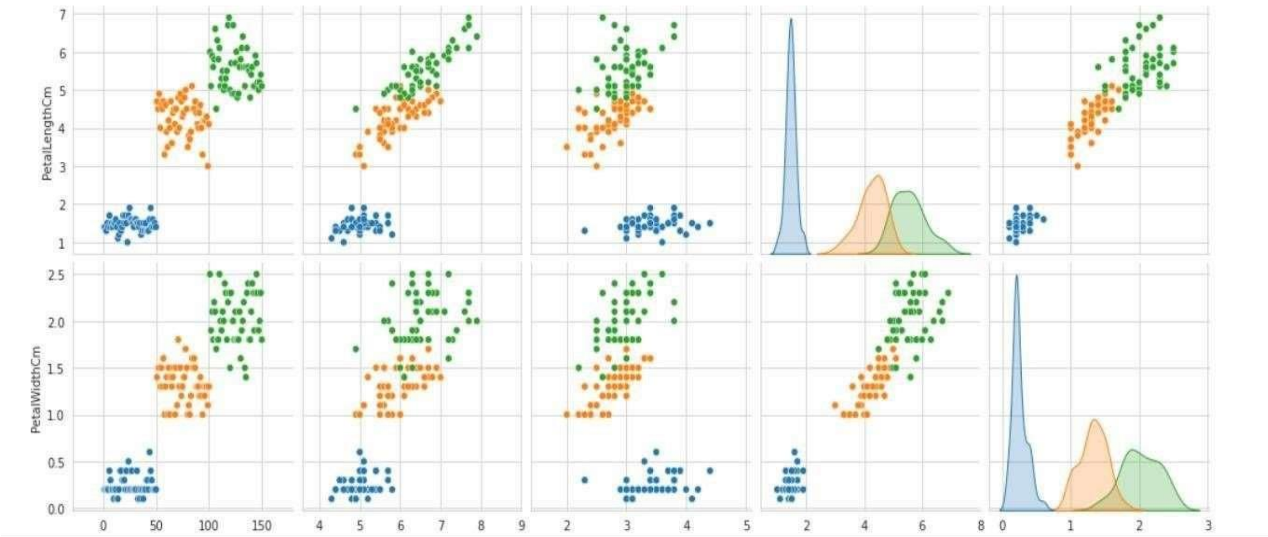
```
[25] Cluster = pd.crosstab(index=df['Species'],columns='count')
     Cluster
```

col_0	count
Species	
Iris-setosa	50
Iris-versicolor	50
Iris-virginica	50

```
sns.set_style("whitegrid")
sns.pairplot(df,hue="Species",size=3);
plt.show()
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/axisgrid.py:2076: UserWarning: The `size` parameter has been renamed to `height`; please update your
  warnings.warn(msg, UserWarning)
```






```
X=df.values
X = X[ :, 0: 4]
m = X.shape[0]
n=X.shape [1]
n_iter=50
k=3
```

[31]
```
centroids = np.array([]).reshape(n,0)
for i in range (k) :
    rand = rd . randint(0,m- 1)
    centroids = np.c_[centroids,X[rand]]
centroids
```

```
array([[62, 119, 78],
       [5.9, 7.7, 6.7],
       [3.0, 2.6, 3.0],
       [4.2, 6.9, 5.0]], dtype=object)
```

[32]
```
output={}
for i in range(n_iter):
    dist = np.array([]).reshape(m,0)
    for j in range(k):
        tempdist = np.sum((X - centroids[:,j])**2,axis=1)
        dist = np.c_[dist,tempdist]
    C = np.argmin(dist,axis=1) + 1
```

```
Y={}
for j  in  range( k) :
    Y [j+1] = np.array ([ ]).reshape(4, 0)
for i in range (m) :
    Y[C[i]] = np.c_[Y[C[i]],X[i]]
for j  in range ( k) :
    Y[j+1] = Y[j+1].T
```

[36]
```
for j in range(k):
    centroids[:,j] = np.mean(Y[j+1],axis=0)
output = Y
print('Outputs for iterations:',i)
print(output)
```

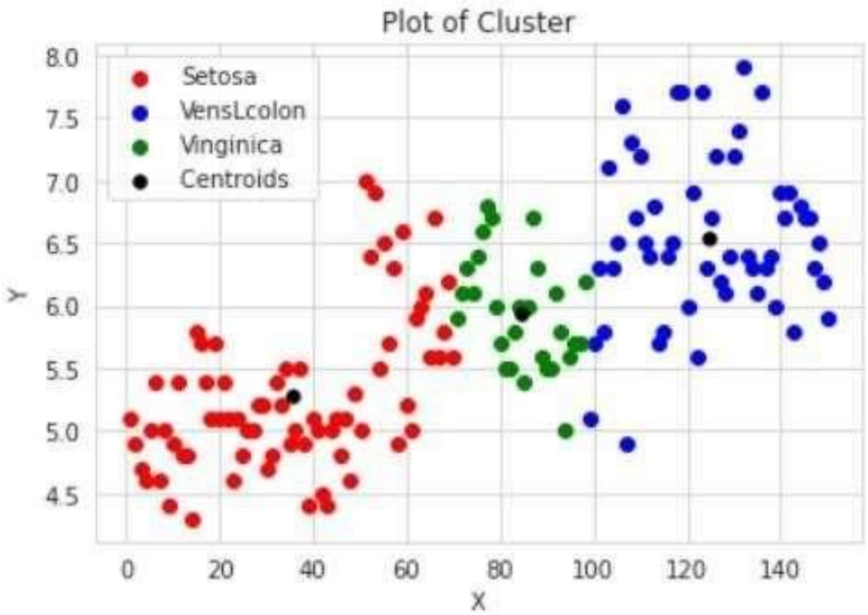
```
        [120, 6.0, 2.2, 5.0],
        [121, 6.9, 3.2, 5.7],
        [122, 5.6, 2.8, 4.9],
        [123, 7.7, 2.8, 6.7],
        [124, 6.3, 2.7, 4.9],
        [125, 6.7, 3.3, 5.7],
        [126, 7.2, 3.2, 6.0],
        [127, 6.2, 2.8, 4.8],
        [128, 6.1, 3.0, 4.9],
        [129, 6.4, 2.8, 5.6],
        [130, 7.2, 3.0, 5.8],
        [131, 7.4, 2.8, 6.1],
        [132, 7.9, 3.8, 6.4],
        [133, 6.4, 2.8, 5.6],
        [134, 6.3, 2.8, 5.1],
        [135, 6.1, 2.6, 5.6],
        [136, 7.7, 3.0, 6.1],
        [137, 6.3, 3.4, 5.6],
        [138, 6.4, 3.1, 5.5],
        [139, 6.0, 3.0, 4.8],
        [140, 6.9, 3.1, 5.4],
        [141, 6.7, 3.1, 5.6],
        [142, 6.9, 3.1, 5.1],
        [143, 5.8, 2.7, 5.1],
```

```
[143, 5.8, 2.7, 5.1],
[144, 6.8, 3.2, 5.9],
[145, 6.7, 3.3, 5.7],
[146, 6.7, 3.0, 5.2],
[147, 6.3, 2.5, 5.0],
[148, 6.5, 3.0, 5.2],
[149, 6.2, 3.4, 5.4],
[150, 5.9, 3.0, 5.1]], dtype=object), 3: array([[71, 5.9, 3.2, 4.8],
[72, 6.1, 2.8, 4.0],
[73, 6.3, 2.5, 4.9],
[74, 6.1, 2.8, 4.7],
[75, 6.4, 2.9, 4.3],
[76, 6.6, 3.0, 4.4],
[77, 6.8, 2.8, 4.8],
[78, 6.7, 3.0, 5.0],
[79, 6.0, 2.9, 4.5],
[80, 5.7, 2.6, 3.5],
[81, 5.5, 2.4, 3.8],
[82, 5.5, 2.4, 3.7],
[83, 5.8, 2.7, 3.9],
[84, 6.0, 2.7, 5.1],
[85, 5.4, 3.0, 4.5],
[86, 6.0, 3.4, 4.5],
[87, 6.7, 3.1, 4.7],
[88, 6.3, 2.3, 4.4],
[89, 5.6, 3.0, 4.1],
[90, 5.5, 2.5, 4.0],
[91, 5.5, 2.6, 4.4],
[92, 6.1, 3.0, 4.6],
[93, 5.8, 2.6, 4.0],
[94, 5.0, 2.3, 3.3],
```


```
[94, 5.0, 2.3, 3.3],
```


```
[95, 5.6, 2.7, 4.2],
[96, 5.7, 3.0, 4.2],
[97, 5.7, 2.9, 4.2],
[98, 6.2, 2.9, 4.3]], dtype=object)}
```

```
colour=['red','blue','green']
labels=['Setosa','VensLcolon','Vinginica']
for i in range(k):
    plt.scatter(output[i+1][:,0],output[i+1][:,1],c=colour[i],label=labels[i])
plt.scatter(centroids[0,:],centroids[1,:],s=30,c='Black',label='Centroids ')
plt.xlabel('X')
plt.ylabel('Y')
plt.legend()
plt.title('Plot of Cluster')
plt.show()
```


Plot of Cluster
Setosa
VensLcolon
Vinginica
Centroids
8.0
7.5
7.0
6.5
6.0
5.5
5.0
4.5
Y
0
20
40
60
80
100
120
140
X