PAGE NO : 23
DATE : 5
EXP.NO. : 8

8 Apply EM algorithm to cluster a set of data stored in a .csv file. Use the same data set for clustering using K-Means algorithm Compare the results of these two algorithms & comment on the quality of clustering You can add Java/Python ML Library classes / API in the program.

```
import matplotlib.pyplot as plt
from sklearn import datasets
from sklearn.cluster import KMeans
import sklearn.metrics as sm
import pandas as pd.
import numpy as np.

iris = datasets.load_iris()
X = pd.DataFrame(iris.data)
X.columns = ['sepal_length', 'sepal_width'
             ,'Petal_length', 'Petal_width']

y = pd.DataFrame(iris.target)
y.columns = ['Targets']

model = KMeans(n_clusters=3)
model.fit(X)
```

ARUN'S

```
model.Labels_
plt.figure(figsize=(14,7))
colormap = np.array(['red','lime','black'])

plt.subplot(1,2,1)
plt.scatter(x.Petal_length, x.Petal_width,
        c=colormap[y.Targets], s=40)
plt.title('Real Classification')

plt.subplot(1,2,2)
plt.scatter(x.Petal_length, x.Petal_width,
        c=colormap[model.labels_] s=40)

plt.title('K Mean classification')
plt.figure(figsize=(14,7))

predy = np.choose(model.Labels_,
            [0,1,2]).astype(np.int64)
print(predy)
plt.subplot(1,2,1)
plt.scatter(x.Petal_length, x.Petal_width,
        c=colormap[y.Targets], s=40)

plt.title('Real classification')
plt.subplot(1,2,2)
```

ARUN'S

```
plt.scatter(x.Petal_length, x.Petal_width,
            c=colormap[pred_y], s=40)
plt.title('K Mean classification')

Print('The accuracy score of K-Mean:',
      sm.accuracy_score(y, model.labels_))
print('The confusion matrix of K-Mean:',
      sm.confusion_matrix(y,model.
                          labels_))
from sklearn import preprocessing
scaler = preprocessing.standard_scaler()
Scaler.fit(x)
xsa = scalar.transform(x)
xs = pd.DataFrame(xsa, columns = x.columns)
from sklearn.mixture import GaussianMixture
gmm = GaussianMixture(n_components=3)
gmm.fit(xs)

y_cluster_gmm = gmm.predict(xs)

plt.subplot(2,2,3)
plt.scatter(x.petal_length, x.Petal_width,
            c=colormap[y_cluster_gmm], s=40)
plt.title('GMM classification')
```

```
print ('The accuracy score of EM :',
        sm.accuracy_score (y, y_cluster_
                           gmm))
print ('The confusion matrix of EM :', sm.
        confusion_matrix (y, y_cluster_gmm))
```

[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 2 2 1 2 2 2 2 2 2
2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2 1 2 1 1 1 1 2 1 1 1 1
1 1 2 2 1 1 1 1 2 1 2 1 2 1 2 1 1 2 2 1 1 1 1 1 2 1
2 1 1 1 2 1 1 1 2 1 1 2]

The accuracy score of K-Mean : 0.44
The confusion matrix of K-Mean :

[[50 0 0]
[0 2 48]
[0 36 14]]

The accuracy score of EM : 0.3333333333333
The confusion matrix of EM : [[0 50 0]
[45 0 5]
[0 0 50]]