8 Apply EM algorithm to cluster a set of data stored in a csv file use the same dataset for clustering. using k-means algorithm. compare the result of these too algorithm and coment on the quality of clustering. You can add Java/python ML library classes.

```
import matplotlib. pyplot as plot.
from sklearn import datasets.
from sklearn. cluster import kmeans.
import sklearn. metrices as sm
import. pandas as pd.
import numpy as up.
Iris = daterets. load. iris()
x = pd.DataFrame (iris data)
x. columns = [' Sepal length', 'sepal width', 'petal
           length', 'petal width'].
y = pd.DataFrame (iris. target)
y. columns = [' Target']

model = kmeans (n_ clusters = 3)
model. fit (x)
model. labels
plt. figure (figure = (14,7))
Colormap = up. array [['red', 'the', 'black']]
plt. subplot (1,1).
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 means classification')
plt. figsize = (14,7)
pred y = np.choose (model. labels-, [0,1,2]. astype.
(np. int 64)
print (pred y)
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 [pred y], s=40)
plt. title ('k means classification').
print('The accuracy Score of k means:', sm.
accuracy-score (y, model. labels-))
print('The confusion matrix of k means:'
sm. confusion-matrix (y. model-labels))
from sklearn import preprocessing
scaler = preprocessing. Standard scalar()
scaler. fit (x)


x2a = scalar. transform (x)
x3 = pd. Dataframe (x2a, colum = x. columns)
from sklearn. mixture import gaussian
mixture
        gmm = Gaussianmixture (n-components=3
```

```
gmm. fitst(xs)
y_cluster-gmm = gmm.predict(xs)
plt-subplot(2,2,3)
plt.scatter(x.petal_length. x.petal-width, c=color.
    map[y_cluster_gmm],s=45).
plt.title('GMM classification')
print('the accuracy score of Em:', sm.accuracy.
    score (y, y_cluster-gmm))
print('The confusion matrix of Em:', sm.conf
usion matrix (y, y cluster.gmm))
```

output:

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

The accuracy sore of k-means = 62q.
The confusion of k-means : [[0 5 0 0]
                            [4 8 0 2]
                            [1 4 0 36]]

The accuracy score of EM: 0.9666666667
The confusion matrix : [[50 0 0 0]
                        [0 45 5]
                        [0 0 50]]