GAS SENSORS - QCM DATASET

KATHIA TERAN

IMPORT DATASETS SEPARATELY

```
%matplotlib inline
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn import metrics

dataset = pd.read_csv('/Users/kathiateran/Documents/Machine Learning/A1. FINAL/datasets/QCM3.csv', sep=',')
dataset
```

	0.799_0.201	0.799_0.201.1	0.700_0.300	0.700_0.300.1	0.600_0.400	0.600_0.400.1	0.501_0.499	0.501_0.499.1	0.400_0.600	0.400_0.600.1	Class
0	-10.06	-10.62	-14.43	-18.31	-24.64	-30.56	-38.62	-45.59	-54.89	-62.28	0
1	-9.69	-10.86	-16.73	-21.75	-28.47	-35.83	-43.65	-52.43	-61.92	-71.27	0
2	-12.07	-14.28	-21.54	-27.92	-35.19	-43.94	-52.04	-62.49	-71.97	-83.10	0
3	-14.21	-17.41	-25.91	-33.36	-41.29	-51.27	-59.94	-71.55	-81.51	-93.83	0
4	-16.57	-20.35	-29.97	-37.84	-47.03	-57.29	-67.13	-78.96	-90.01	-102.65	0
5	-75.61	-64.10	-122.08	-102.17	-174.79	-145.50	-214.69	-177.30	-250.83	-207.99	1
6	-76.28	-64.53	-123.40	-103.06	-174.74	-144.98	-206.75	-170.68	-240.14	-200.05	1
7	-76.96	-65.37	-124.50	-103.87	-169.93	-140.56	-200.89	-165.88	-232.82	-194.70	1
8	-78.29	-66.49	-125.78	-104.77	-165.42	-136.17	-196.10	-161.99	-226.89	-189.57	1
9	-79.62	-66.85	-126.61	-104.38	-162.73	-133.26	-194.87	-160.68	-221.33	-183.81	1
10	-88.03	-76.88	-128.47	-124.36	-177.13	-175.21	-220.91	-218.92	-274.27	-268.55	2
11	-91.08	-79.42	-131.32	-125.91	-182.66	-178.76	-224.73	-221.49	-272.67	-266.15	2
12	-92.08	-80.56	-132.50	-126.99	-183.33	-179.25	-222.60	-219.52	-264.81	-259.99	2

- Class columns were merged into a single one for data processing simplification purposes.

Q3 - SVM POLYNOMIAL

```
X = dataset.iloc[:, 0:10].values
y = dataset.iloc[:, 10].values
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 0)
```

```
from sklearn.svm import SVC

classifier = SVC(kernel='poly',C=1000, degree=3, gamma='auto')

classifier.fit(X_train, y_train)

pred_train_poly = classifier.predict(X_train)
pred_test_poly = classifier.predict(X_test)

print('\nPrediction accuracy for the training dataset')
print('{:.2%}\n'.format(metrics.accuracy_score(y_train, pred_train_poly)))


print('Prediction accuracy for the test dataset')
print('{:.2%}\n'.format(metrics.accuracy_score(y_test, pred_test_poly)))

print('Confusion Matrix')
print(metrics.confusion_matrix(y_test, pred_test_poly))
```

- Data split: 80% training, 20% testing

RESULTS

Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	1
1	1.00	1.00	1.00	1
2	1.00	1.00	1.00	1
3	1.00	1.00	1.00	2
accuracy			1.00	5
macro avg	1.00	1.00	1.00	5
weighted avg	1.00	1.00	1.00	5

```
True Positive:  1
False Positive:  0
False Negative:  0
True Negative:  1
```

Overall Accuracy	100%
Sensitivity	100%
Specificity	100%

```
Prediction accuracy for the training dataset
100.00%

Prediction accuracy for the test dataset
100.00%

Confusion Matrix
[[1 0 0 0]
 [0 1 0 0]
 [0 0 1 0]
 [0 0 0 2]]
```

Q3 - KNN

```
X = dataset.iloc[:, 0:10].values
y = dataset.iloc[:, 10].values
```

```
# Splitting the dataset:
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=0)
```

```
#K-NEIGHBOR
from sklearn.neighbors import KNeighborsClassifier

knn_clf = KNeighborsClassifier(n_neighbors=5, metric='minkowski', p=2)

knn_clf.fit(X_train, y_train)
```

- Data split: 80% training, 20% testing

RESULTS

Classification Report:

	precision	recall	f1-score	support
0	0.79	0.94	0.86	16
1	0.70	1.00	0.82	14
2	0.75	1.00	0.86	15
3	0.78	1.00	0.88	14
4	0.77	0.77	0.77	13
micro avg	0.76	0.94	0.84	72
macro avg	0.76	0.94	0.84	72
weighted avg	0.76	0.94	0.84	72
samples avg	0.78	0.96	0.83	72

```
True Positive:  1
False Positive:  0
False Negative:  0
True Negative:  1
```

Overall Accuracy	100%
Sensitivity	100%
Specificity	100%

```
Prediction accuracy for the training dataset
100.00%

Prediction accuracy for the test dataset
100.00%

Confusion Matrix
[[1 0 0 0]
 [0 1 0 0]
 [0 0 1 0]
 [0 0 0 2]]
```

Q3 – HMM

```
from sklearn import metrics
from hmmlearn import hmm
from hmmlearn.hmm import GaussianHMM

hmm_clf = GaussianHMM(n_components=5, covariance_type='diag', n_iter=10000).fit(X)

pred_train_hmm = hmm_clf.predict(X_train)
pred_test_hmm = hmm_clf.predict(X_test)

print('\nPrediction accuracy for the training dataset')
print('{:.2%}\n'.format(metrics.accuracy_score(y_train, pred_train_hmm)))

print('Prediction accuracy for the test dataset')
print('{:.2%}\n'.format(metrics.accuracy_score(y_test, pred_test_hmm)))

print('Confusion Matrix')
print(metrics.confusion_matrix(y_test, pred_test_hmm))
```

- Data split: 80% training, 20% testing

RESULTS

Classification Report:

	precision	recall	f1-score	support
0	0.00	0.00	0.00	1
1	0.50	1.00	0.67	1
2	0.00	0.00	0.00	1
3	1.00	1.00	1.00	2
accuracy			0.60	5
macro avg	0.38	0.50	0.42	5
weighted avg	0.50	0.60	0.53	5

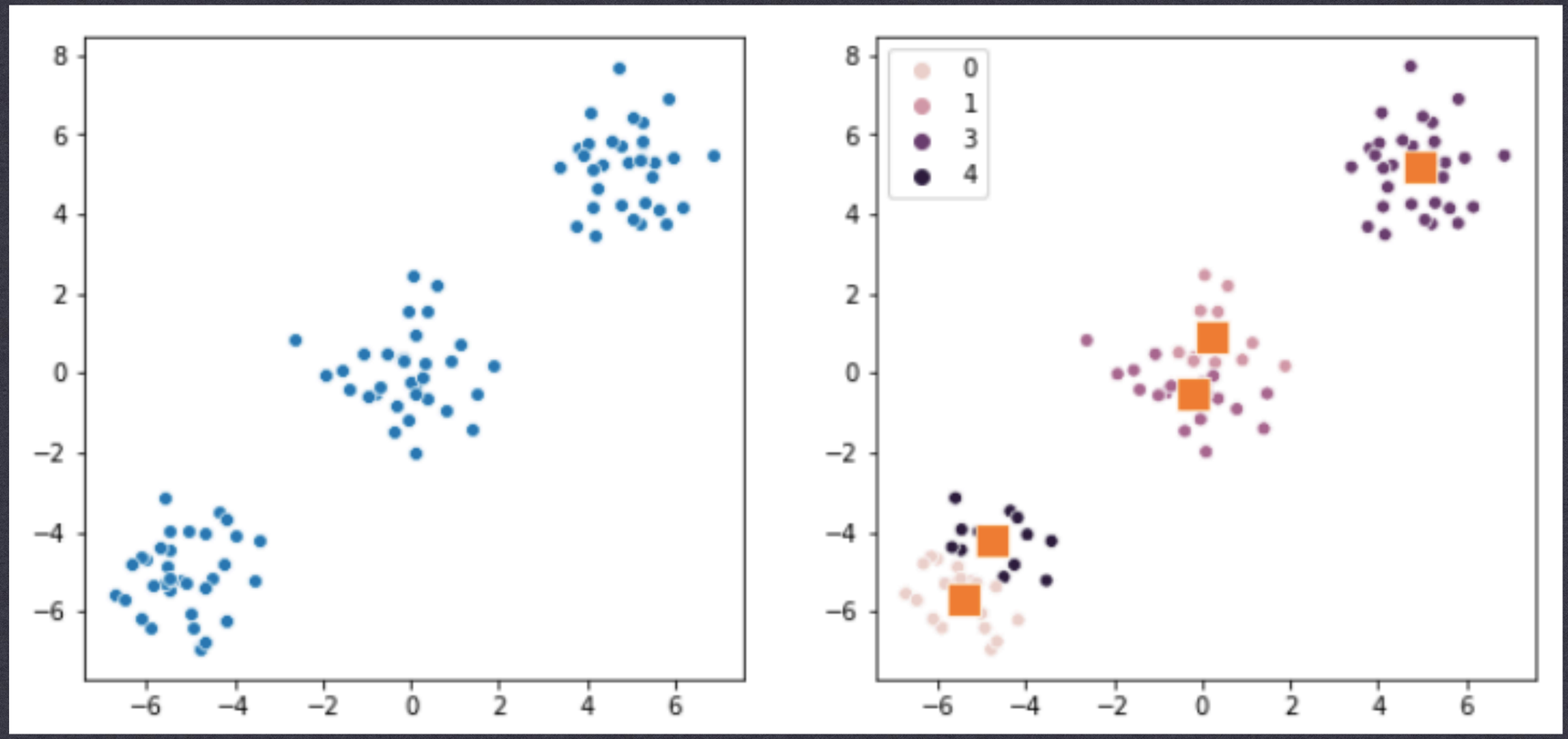
```
True Positive:  0
False Positive:  1
False Negative:  0
True Negative:  1
```

Overall Accuracy	**60%**
Sensitivity	**60%**

```
Prediction accuracy for the training dataset
25.00%

Prediction accuracy for the test dataset
60.00%

Confusion Matrix
[[0 1 0 0]
 [0 1 0 0]
 [1 0 0 0]
 [0 0 0 2]]
```

Q3 – FCM

```
from fcmeans import FCM

n_samples = 100
fcm = FCM(n_clusters=5)

fcm.fit(X)
```



- Data split: 80% training, 20% testing

RESULTS

Classification Report:

	precision	recall	f1-score	support
0	0.00	0.00	0.00	1
1	0.00	0.00	0.00	1
2	0.00	0.00	0.00	1
3	1.00	1.00	1.00	1
4	0.00	0.00	0.00	1
accuracy			0.20	5
macro avg	0.20	0.20	0.20	5
weighted avg	0.20	0.20	0.20	5

```
True Positive:  0
False Positive:  0
False Negative:  0
True Negative:  0
```

Overall Accuracy	20%
Sensitivity	20%

```
Prediction accuracy for the training dataset
20.00%

Prediction accuracy for the test dataset
20.00%

Confusion Matrix
[[0 0 1 0 0]
 [0 0 0 0 1]
 [1 0 0 0 0]
 [0 0 0 1 0]
 [1 0 0 0 0]]
```

Q6, Q10 - SVM LINEAR & KNN

Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	1
1	1.00	1.00	1.00	1
2	1.00	1.00	1.00	1
3	1.00	1.00	1.00	2
accuracy			1.00	5
macro avg	1.00	1.00	1.00	5
weighted avg	1.00	1.00	1.00	5

```
True Positive:  1
False Positive:  0
False Negative:  0
True Negative:  1
```

	SVM	KNN
Overall Accuracy	100%	100%
Sensitivity	100%	100%

```
Prediction accuracy for the training dataset
100.00%

Prediction accuracy for the test dataset
100.00%

Confusion Matrix
[[1 0 0 0]
 [0 1 0 0]
 [0 0 1 0]
 [0 0 0 2]]
```

Q6 – HMM

Classification Report:

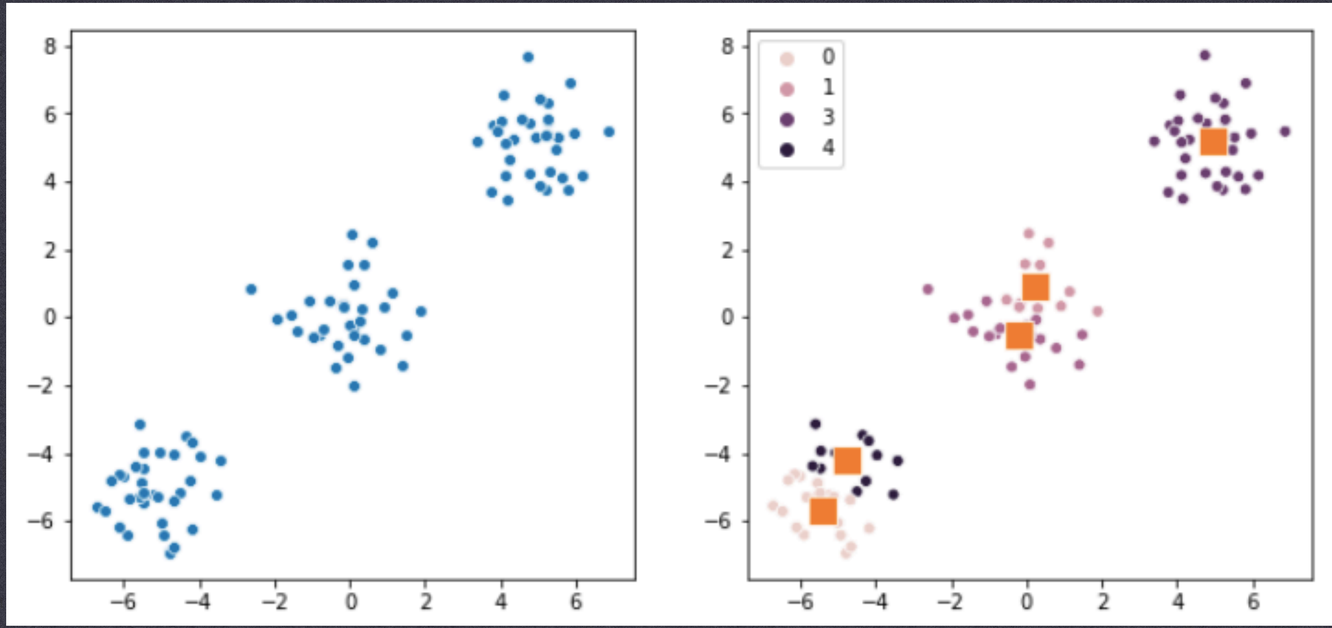
	precision	recall	f1-score	support
0	0.00	0.00	0.00	1
1	0.00	0.00	0.00	1
2	0.20	1.00	0.33	1
3	0.00	0.00	0.00	2
accuracy			0.20	5
macro avg	0.05	0.25	0.08	5
weighted avg	0.04	0.20	0.07	5

```
True Positive:  0
False Positive:  0
False Negative:  0
True Negative:  0
```

Overall Accuracy	20%
Sensitivity	20%

```
Prediction accuracy for the training dataset
30.00%

Prediction accuracy for the test dataset
20.00%

Confusion Matrix
[[0 0 1 0]
 [0 0 1 0]
 [0 0 1 0]
 [0 0 2 0]]
```

Q6 – FCM




```
True Positive:  0
False Positive:  1
False Negative:  0
True Negative:  0
```

```
Prediction accuracy for the training dataset
20.00%

Prediction accuracy for the test dataset
20.00%

Confusion Matrix
[[0 1 0 0 0]
 [0 0 0 1 0]
 [1 0 0 0 0]
 [0 0 1 0 0]
 [0 0 0 0 1]]
```

```
Classification Report:

              precision    recall  f1-score   support

           0       0.00      0.00      0.00         1
           1       0.00      0.00      0.00         1
           2       0.00      0.00      0.00         1
           3       0.00      0.00      0.00         1
           4       1.00      1.00      1.00         1

    accuracy                           0.20         5
   macro avg       0.20      0.20      0.20         5
weighted avg       0.20      0.20      0.20         5
```

Overall Accuracy	20%
Sensitivity	20%

Q7, Q12 - SVM POLY & KNN

```
Classification Report:

              precision    recall  f1-score   support

           0       1.00      1.00      1.00         1
           1       1.00      1.00      1.00         1
           2       1.00      1.00      1.00         1
           3       1.00      1.00      1.00         2

    accuracy                           1.00         5
   macro avg       1.00      1.00      1.00         5
weighted avg       1.00      1.00      1.00         5
```

```
Prediction accuracy for the training dataset
95.00%

Prediction accuracy for the test dataset
100.00%
```

- KNN training accuracy differed with 95% instead of 100%

```
True Positive:  1
False Positive:  0
False Negative:  0
True Negative:  1
```

	SVM	KNN
Overall Accuracy	100%	100%
Sensitivity	100%	100%

```
Prediction accuracy for the training dataset
100.00%

Prediction accuracy for the test dataset
100.00%

Confusion Matrix
[[1 0 0 0]
 [0 1 0 0]
 [0 0 1 0]
 [0 0 0 2]]
```

Q7 – HMM

Classification Report:

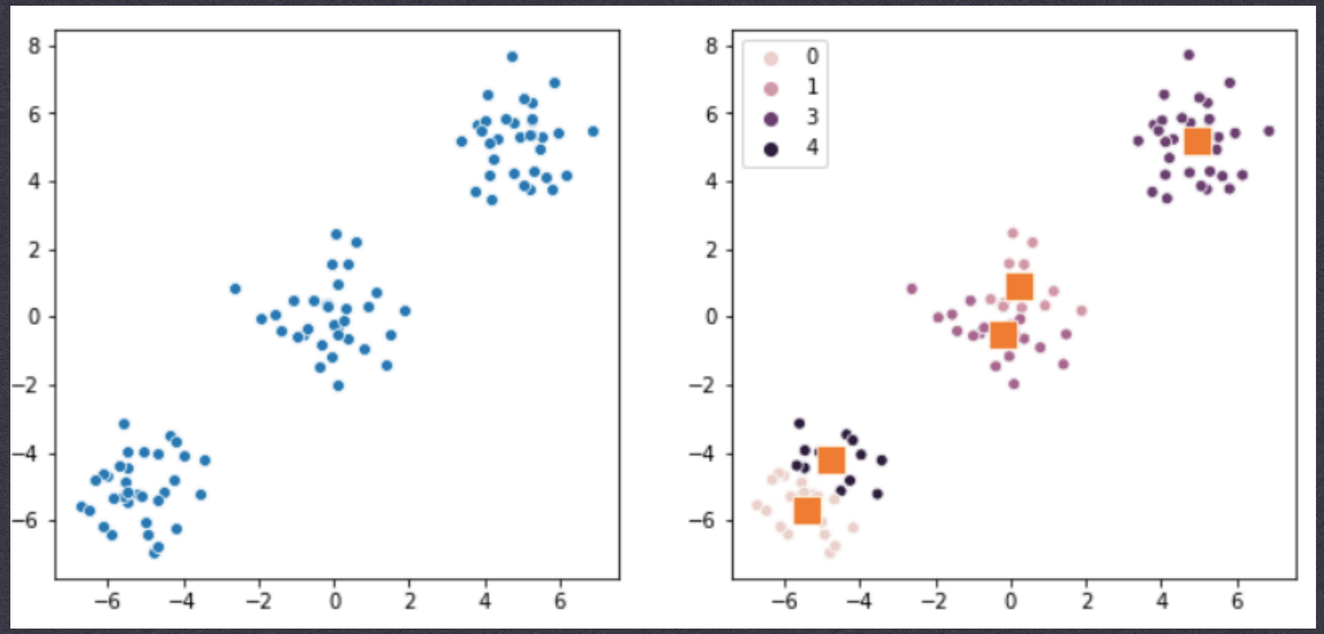
	precision	recall	f1-score	support
0	0.00	0.00	0.00	1
1	0.25	1.00	0.40	1
2	1.00	1.00	1.00	1
3	0.00	0.00	0.00	2
accuracy			0.40	5
macro avg	0.31	0.50	0.35	5
weighted avg	0.25	0.40	0.28	5

```
True Positive:  0
False Positive:  1
False Negative:  0
True Negative:  1
```

Overall Accuracy	**40%**
Sensitivity	**40%**

```
Prediction accuracy for the training dataset
25.00%

Prediction accuracy for the test dataset
40.00%

Confusion Matrix
[[0 1 0 0]
 [0 1 0 0]
 [0 0 1 0]
 [0 2 0 0]]
```

Q7, Q10 - FCM




```
True Positive:  0
False Positive:  0
False Negative:  0
True Negative:  0
```

- Q7

```
True Positive:  0
False Positive:  1
False Negative:  0
True Negative:  0
```

- Q10

```
Prediction accuracy for the training dataset
20.00%

Prediction accuracy for the test dataset
20.00%

Confusion Matrix
[[0 1 0 0 0]
 [0 0 0 1 0]
 [1 0 0 0 0]
 [0 0 1 0 0]
 [0 0 0 0 1]]
```

```
Classification Report:

              precision    recall  f1-score   support

           0       0.00      0.00      0.00         1
           1       0.00      0.00      0.00         1
           2       0.00      0.00      0.00         1
           3       0.00      0.00      0.00         1
           4       1.00      1.00      1.00         1

    accuracy                           0.20         5
   macro avg       0.20      0.20      0.20         5
weighted avg       0.20      0.20      0.20         5
```

Overall Accuracy	20%
Sensitivity	20%

Q10, Q12 - HMM

Classification Report:

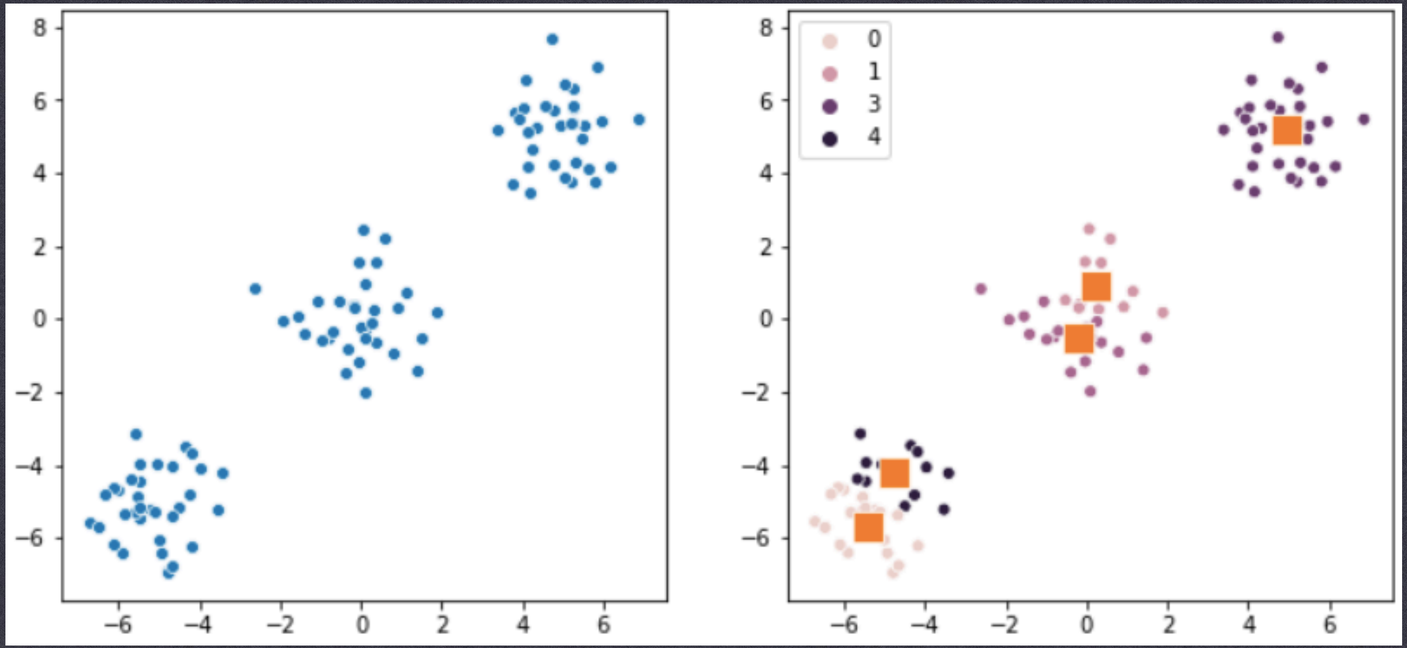
	precision	recall	f1-score	support
0	0.00	0.00	0.00	1
1	0.33	1.00	0.50	1
2	0.00	0.00	0.00	1
3	0.00	0.00	0.00	2
4	0.00	0.00	0.00	0
accuracy			0.20	5
macro avg	0.07	0.20	0.10	5
weighted avg	0.07	0.20	0.10	5

```
True Positive:  0
False Positive:  1
False Negative:  0
True Negative:  1
```

Overall Accuracy	**20%**
Sensitivity	**20%**

```
Prediction accuracy for the training dataset
25.00%

Prediction accuracy for the test dataset
20.00%

Confusion Matrix
[[0 1 0 0 0]
 [0 1 0 0 0]
 [0 0 0 1 0]
 [0 1 0 0 1]
 [0 0 0 0 0]]
```

Q12 - FCM




```
True Positive:  0
False Positive:  1
False Negative:  0
True Negative:  0
```

```
Prediction accuracy for the training dataset
40.00%

Prediction accuracy for the test dataset
40.00%

Confusion Matrix
[[0 1 0 0 0]
 [0 0 0 0 1]
 [1 0 0 0 0]
 [0 0 0 1 0]
 [0 0 0 0 1]]
```

Classification Report:

	precision	recall	f1-score	support
0	0.00	0.00	0.00	1
1	0.00	0.00	0.00	1
2	0.00	0.00	0.00	1
3	1.00	1.00	1.00	1
4	0.50	1.00	0.67	1
accuracy			0.40	5
macro avg	0.30	0.40	0.33	5
weighted avg	0.30	0.40	0.33	5

Overall Accuracy	**40%**
Sensitivity	**40%**

THANKS!