CS5691: Pattern Recognition and Machine Learning
Programming Assignment 2
Project Report

Reetwik Das - CS20D402
Shubhanshu Sharma - EE20D413
Vamsi Sai Krishna Malineni - OE20S302

May 7, 2021

Contents

List of Tables

List of Figures

1 Dataset 1a

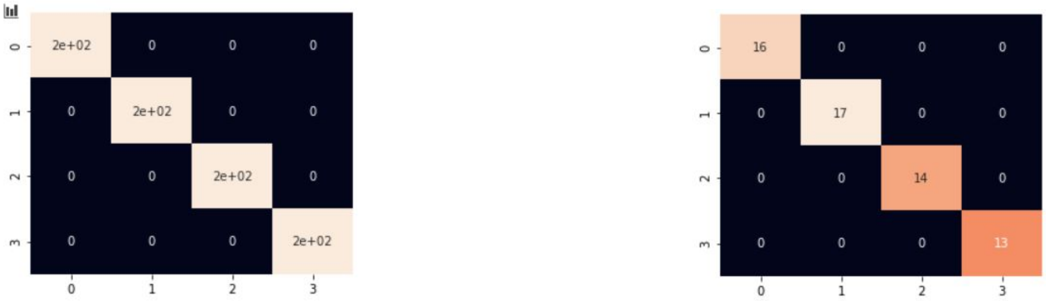
1.1 K-nearest neighbours classifier

Hyperparameters $K = \{1, 7, 15\}$

K	Training Dataset	Validation Dataset	Test Dataset
1	100	100	100
7	100	100	100
15	100	100	100

Table 1.1.1: Classification accuracy of KNN classifier on Dataset 1a

As we can see from the table above the model with the best performance on the test data is with K=1 with an accuracy of 100% (same accuracy with least computation cost).



(a) for Training data

(b) for Test data

Figure 1.1.1: Confusion matrix for datatset 1a using KNN classifier, $K = 1$

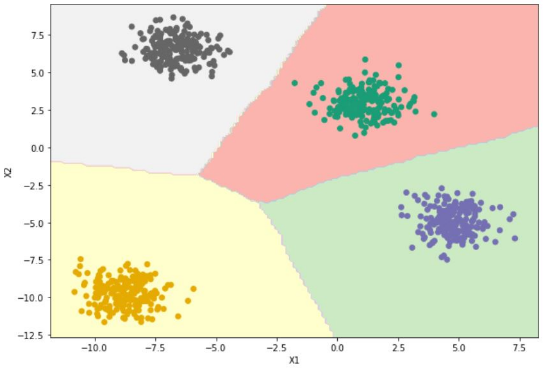
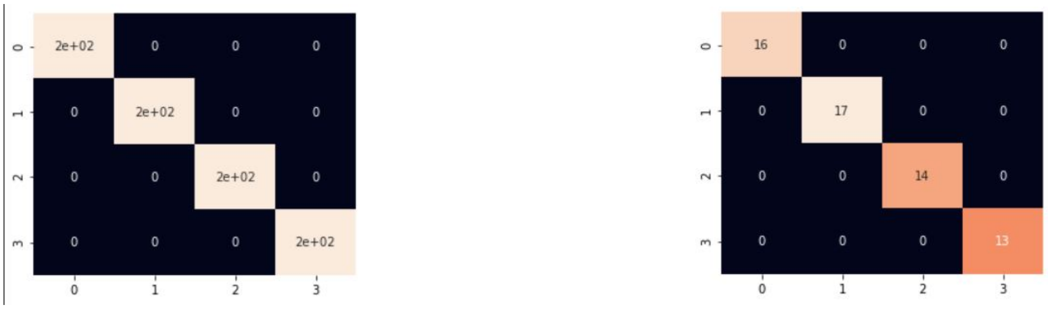


Figure 1.1.2: Decision region plots superposed with training examples for datatset 1a using KNN classifier, $K = 1$

1.2 Naive bayes Classifier, covariance matrix for all the classes is $\sigma^2 I$

Training Dataset	Validation Dataset	Test Dataset
100	100	100

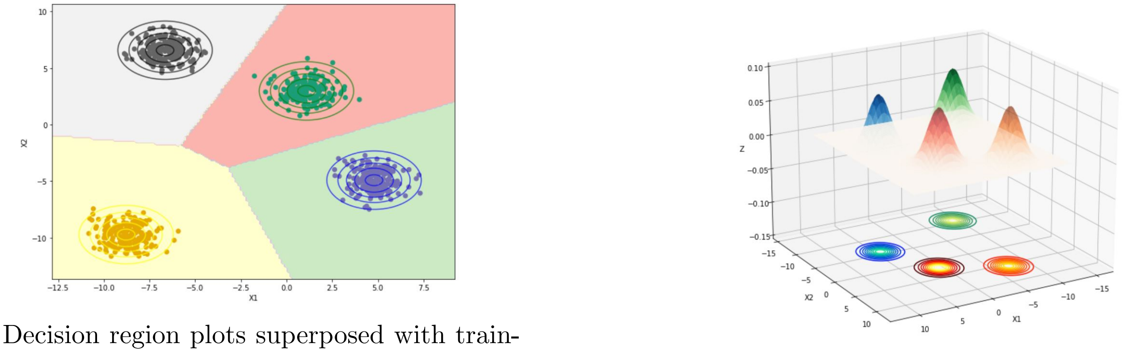
Table 1.2.1: Classification accuracy of Naive bayes classifier on Dataset 1a, the covariance matrices are same and $= \sigma^2 I$



(a) for Training data

(b) for Test data

Figure 1.2.1: Confusion matrix for datatset 1a using Naive bayes classifier, the covariance matrices are same and $= \sigma^2 I$



(a) Decision region plots superposed with training examples

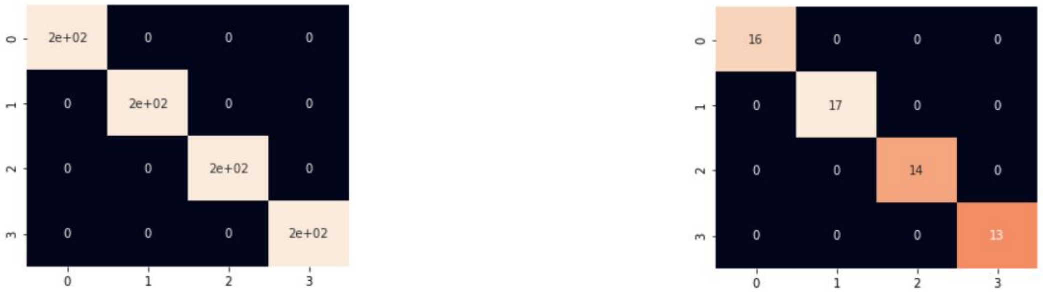
(b) Plots of the level curves

Figure 1.2.2: Plots for datatset 1a using Naive bayes classifier, the covariance matrices are same and $= \sigma^2 I$

1.3 Naive bayes Classifier, covariance matrix for all the classes is C

Training Dataset	Validation Dataset	Test Dataset
100	100	100

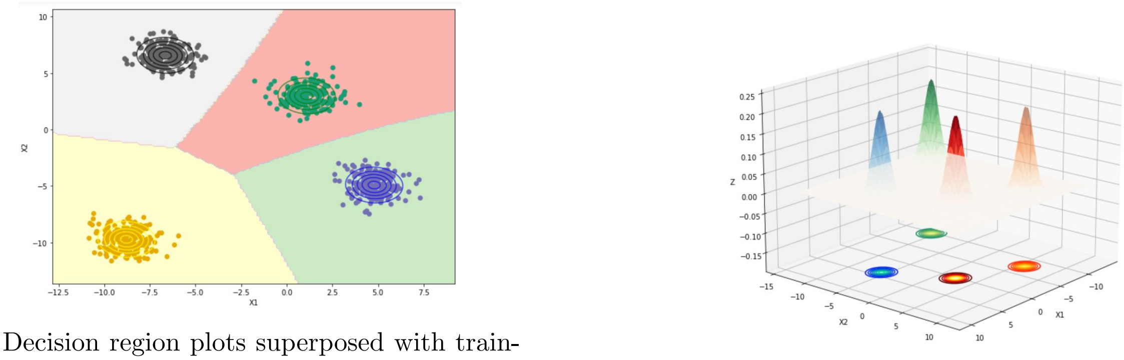
Table 1.3.1: Classification accuracy of Naive bayes classifier on Dataset 1a, the covariance matrices are same $= C$



(a) for Training data

(b) for Test data

Figure 1.3.1: Confusion matrix for datatset 1a using Naive bayes classifier, the covariance matrices are same $= C$



(a) Decision region plots superposed with training examples

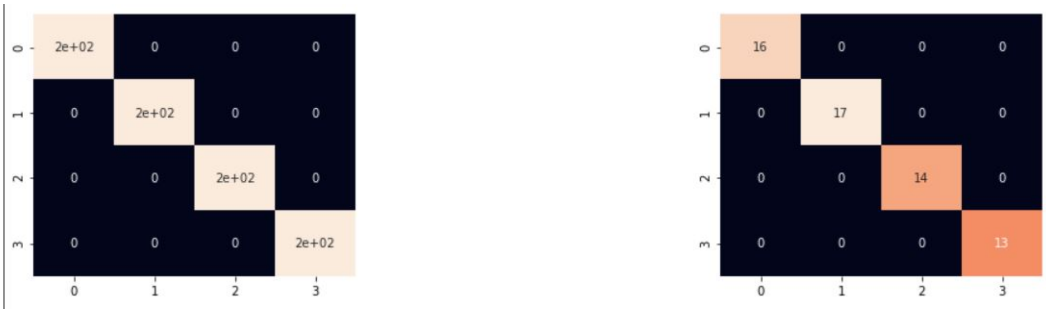
(b) Plots of the level curves

Figure 1.3.2: Plots for datatset 1a using Naive bayes classifier, the covariance matrices are same $= C$

1.4 Naive bayes Classifier, covariance matrix for all the classes is different

Training Dataset	Validation Dataset	Test Dataset
100	100	100

Table 1.4.1: Classification accuracy of Naive bayes classifier on Dataset 1a, the covariance matrices are different



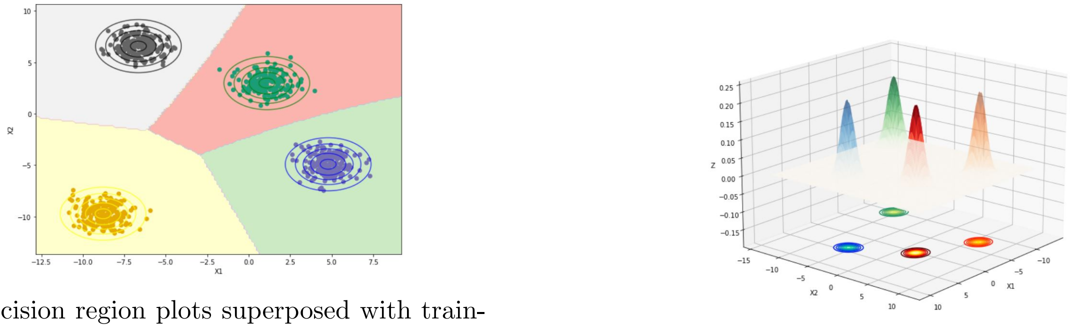
	0	1	2	3
0	2e+02	0	0	0
1	0	2e+02	0	0
2	0	0	2e+02	0
3	0	0	0	2e+02

(a) for Training data

	0	1	2	3
0	16	0	0	0
1	0	17	0	0
2	0	0	14	0
3	0	0	0	13

(b) for Test data

Figure 1.4.1: Confusion matrix for datatset 1a using Naive bayes classifier, the covariance matrices are different



(a) Decision region plots superposed with training examples

(b) Plot of level curves

Figure 1.4.2: Plots for datatset 1a using Naive bayes classifier, the covariance matrices are different

We observed that since all the classes are well separated all the classification approaches gave us very good results.

2 Dataset 1b

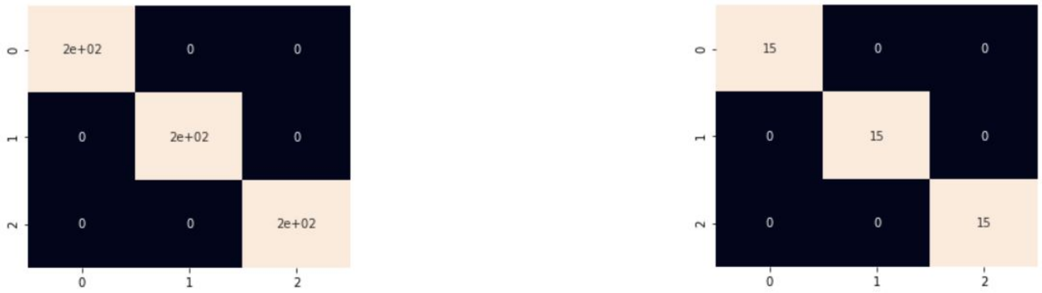
2.1 K-nearest neighbours classifier

Hyperparameters $K = \{1, 7, 15\}$

K	Training Dataset	Validation Dataset	Test Dataset
1	100	100	100
7	100	100	100
15	99.67	100	100

Table 2.1.1: Classification accuracy of KNN classifier on Dataset 1b

As we can see from the table above the model with the best performance on the test data is with K=1 with an accuracy of 100% (same accuracy with least computation cost).



(a) for Training data

(b) for Test data

Figure 2.1.1: Confusion matrix for datatset 1b using KNN classifier, $K = 1$

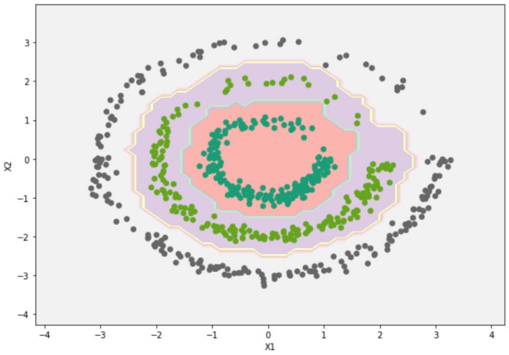


Figure 2.1.2: Decision region plots superposed with training examples for datatset 1b using KNN classifier, $K = 1$

Although the classes are not linearly separable they are well separated for KNN classifier to give good results.

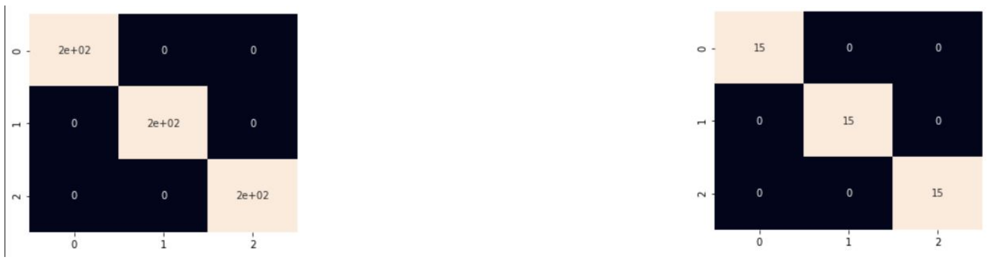
2.2 Bayes classifier with GMM, using full covariance matrices

Hyperparameters $Q = \{3, 5, 10\}$

Q	Training Dataset	Validation Dataset	Test Dataset
3	91.67	97.78	91.1
5	98.66	97.78	97.78
10	100	100	100

Table 2.2.1: Classification accuracy of GMM classifier using full covariance matrix on Dataset 1b

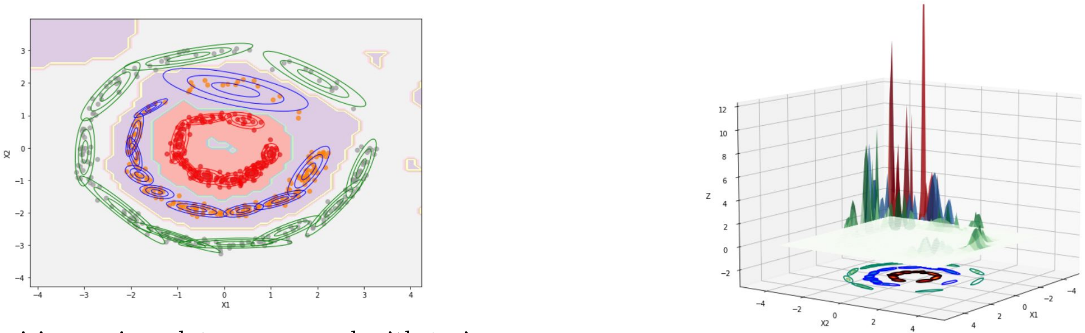
As we can see from the table above the model with the best performance on the test data is with Q=10 with an accuracy of 100%



(a) for Training data

(b) for Test data

Figure 2.2.1: Confusion matrix for datatset 1b using GMM with full covariance matrix, $Q = 10$



(a) Decision region plots superposed with training examples

(b) Plots of the level curves

Figure 2.2.2: Plots for datatset 1b using GMM with full covariance matrix, $Q = 10$

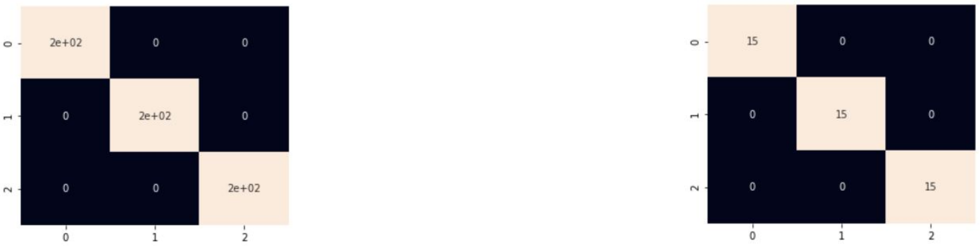
2.3 Bayes classifier with GMM, using diagonal covariance matrices

Hyperparameters $Q = \{3, 5, 10\}$

Q	Training Dataset	Validation Dataset	Test Dataset
3	91.67	97.78	91.1
5	98.66	97.78	97.78
10	99.67	100	100

Table 2.3.1: Classification accuracy of GMM classifier using diagonal covariance matrices on Dataset 1b

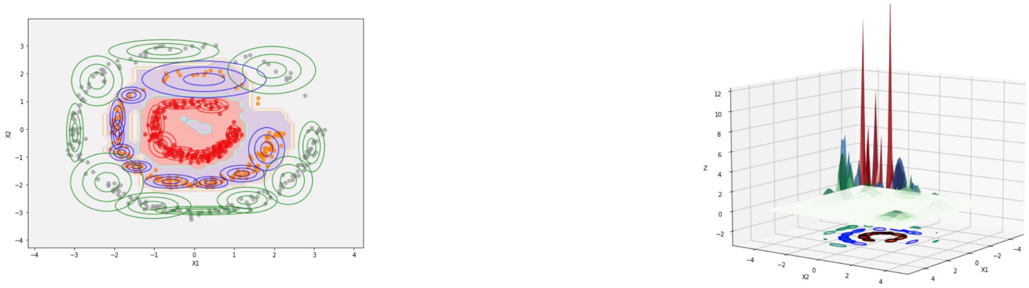
As we can see from the table above the model with the best performance on the test data is with Q=10 with an accuracy of 100%



(a) for Training data

(b) for Test data

Figure 2.3.1: Confusion matrix for datatset 1b using GMM classifier with diagonal covariance matrices



(a) Decision region plots superposed with training examples

(b) Plots of the level curves

Figure 2.3.2: Plots for datatset 1b using GMM with diagonal covariance matrix, $Q = 10$

We saw that with full covariance matrix we get a slightly better result as compared to the diagonal covariance matrix since the GMMs are not restricted to be aligned with the principal axes.

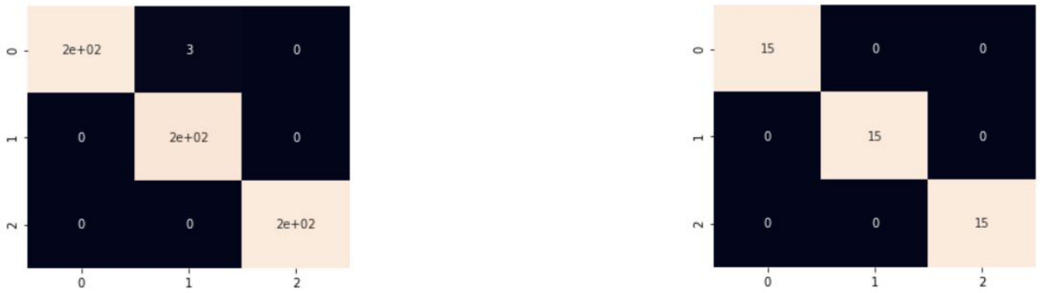
2.4 Bayes classifier with K-nearest neighbours method for estimation of class-conditional probability density function

Hyperparameters $K = \{10, 20\}$

K	Training Dataset	Validation Dataset	Test Dataset
10	99.5	100	100
20	96	100	96.56

Table 2.4.1: Classification accuracy of KNN density estimator classifier on Dataset 1b

As we can see from the table above the model with the best performance on the test data is with K=10 with an accuracy of 100%



(a) for Training data

(b) for Test data

Figure 2.4.1: Confusion matrix for datatset 1b using KNN density estimator classifier, $K = 10$

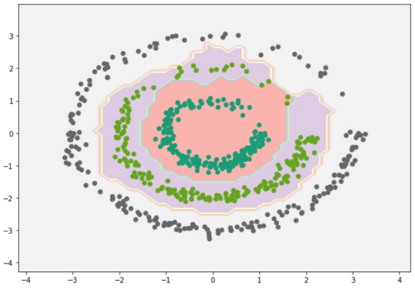


Figure 2.4.2: Decision region plots superposed with training examples for datatset 1b using KNN density estimator, $K = 10$

For KNN using density estimators the decision surface was able to classify better with lesser K as the regions with low density of training points were easily misclassified.

3 Dataset 2a

3.1 Bayes classifier with GMM, using full covariance matrices

The classes we had to distinguish between were:

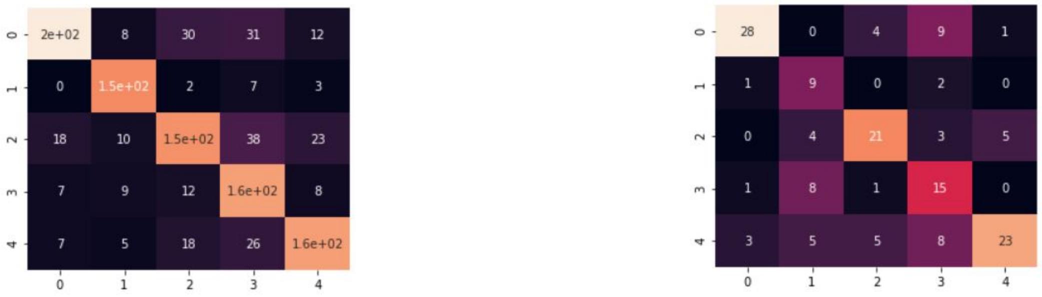
1. Forest
2. Highway
3. Inside city
4. Mountain
5. Street

Hyperparameters $Q = \{3, 5, 10\}$

Q	Training Dataset	Validation Dataset	Test Dataset
3	68.83	54.77	55.12
5	72.04	61.14	59.61
10	74.88	59.23	61.53

Table 3.1.1: Classification accuracy of GMM classifier using full covariance matrices on Dataset 2a

As we can see from the table above the model with the best performance on the test data is with Q=10 with an accuracy of 61.53%



(a) for Training data

(b) for Test data

Figure 3.1.1: Confusion matrix for datatset 2a using GMM with full covariance matrix

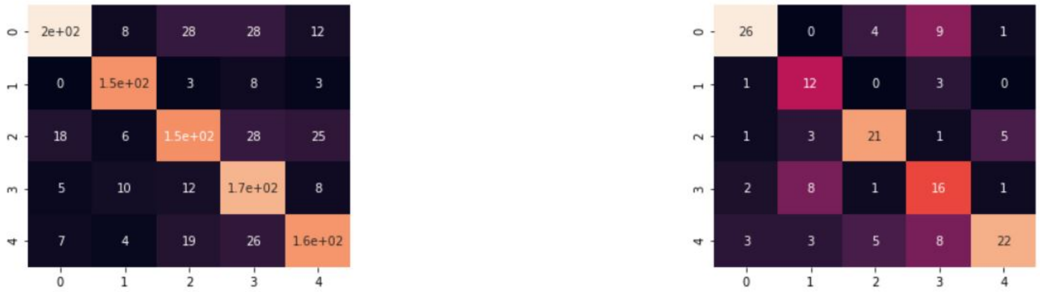
3.2 Bayes classifier with GMM, using diagonal covariance matrices

Hyperparameters $Q = \{3, 5, 10\}$

Q	Training Dataset	Validation Dataset	Test Dataset
3	69.29	56.68	57.05
5	72.86	60.50	59.61
10	76.35	60.50	62.17

Table 3.2.1: Classification accuracy of GMM classifier using diagonal covariance matrices on Dataset 2a

As we can see from the table above the model with the best performance on the test data is with Q=10 with an accuracy of 62.17%



(a) for Training data

(b) for Test data

Figure 3.2.1: Confusion matrix for datatset 2a using GMM with diagonal covariance matrix

For the real dataset the GMM with higher number of clusters was also unable to classify the examples with very high accuracy.
As we increased the number of clusters the accuracy increased to a point after which we started getting some empty clusters as well during the model training. We added an epsilon value to the covariance matrices to ensure that they don't become singular.

4 Dataset 2b

4.1 Bayes classifier with GMM, using full covariance matrices

The classes we had to distinguish between were:

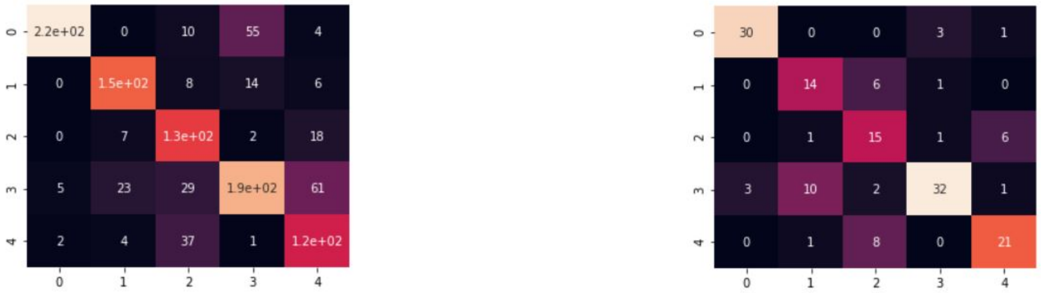
1. Forest
2. Highway
3. Inside city
4. Mountain
5. Street

Hyperparameters $Q = \{3, 7, 13\}$

Q	Training Dataset	Validation Dataset	Test Dataset
3	68.10	69.42	71.79
7	74.61	72.61	70.51
13	73.78	74.52	71.79

Table 4.1.1: Classification accuracy of GMM classifier using full covariance matrices on Dataset 2b

As we can see from the table above the model with the best performance on the test data is with Q=13 with an accuracy of 71.79%



(a) for Training data

(b) for Test data

Figure 4.1.1: Confusion matrix for datatset 2b using GMM classifier with full covariance matrices

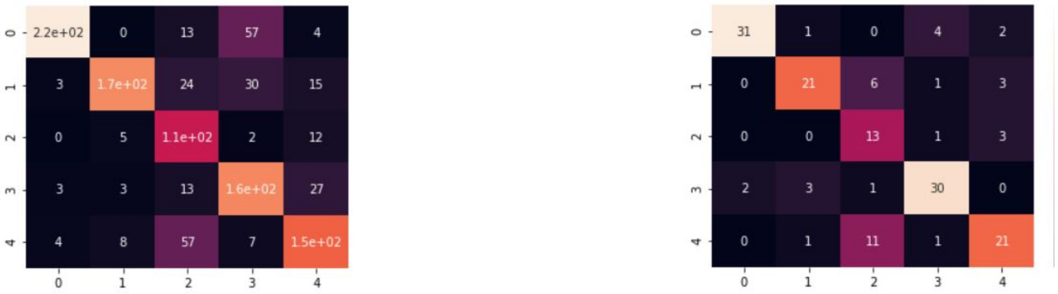
4.2 Bayes classifier with GMM, using diagonal covariance matrices

Hyperparameters $Q = \{3, 7, 13\}$

Q	**Training Dataset**	**Validation Dataset**	**Test Dataset**
3	65.99	69.42	67.94
7	73.69	69.42	73.71
13	73.69	73.24	74.35

Table 4.2.1: Classification accuracy of GMM classifier using diagonal covariance matrices on Dataset 2b

As we can see from the table above the model with the best performance on the test data is with Q=13 with an accuracy of 74.35%



(a) for Training data

(b) for Test data

Figure 4.2.1: Confusion matrix for datatset 2b using GMM classifier with diagonal covariance matrices

We faced similar issues with this dataset as well but since we had more training data $(36X)$ of the previous example we could increase the number of clusters to a higher point to get better classification accuracy.