February-May 2021 Semester

CS5691: Pattern recognition and Machine Learning

Programming Assignment 2

Report

Team 32

Varun Srinivas Venkatesh

MM17B036

Hrishikesh Kambale

ME18B142

S Sabesh Vishwanath

MM18B112

INDEX

Section 1 : Naive Bayes

1.1 Task:

To classify the given data using a Naive Bayes Classifier with a Gaussian distribution with different values of covariance matrix.

a. Covariance matrix for all the classes is the same and is sigma^2 * I
b. Covariance matrix for all the classes is the same and is C
c. Covariance matrix for each class is different

1.2 Steps:

1. Load the dataset, shuffle and split train, test and validation sets.
2. Split the data by class and calculate class specific mean
3. For the 3 different cases, calculate the covariance matrix
4. Using the mean and covariance matrix for each class, calculate posterior probabilities for each input value.
5. Ignore prior probabilities since they are equal
6. Classify the points based on maximum posterior probability
7. Plot the decision surfaces

1.3 Classification Accuracies of the Model:

Table 1.1: Classification Accuracies of the Naive Bayes Classifier (Cases mentioned in 1.1)

Covariance Matrix	Case (a)	Case (b)	Case (c)
Training Data	1.0	1.0	1.0
Testing Data	1.0	1.0	1.0
Validation Data	1.0	1.0	1.0

1. We see that the accuracy of the Naive Bayes Classifier is great in this case primarily because the data is distinctly separable and clustered.
2. Confusion matrices are presented for the third case

1.4 Confusion Matrix for all cases:

Table 1.2: Confusion Matrix for all cases:

Class	Class 0	Class 1	Class 2	Class 3
Class 0	230	0	0	0
Class 1	0	230	0	0
Class 2	0	0	230	0
Class 3	0	0	0	230

3. The accuracy in all cases is 1, so the confusion matrix has no false positives or negatives

1.5 Decision Region Plots:

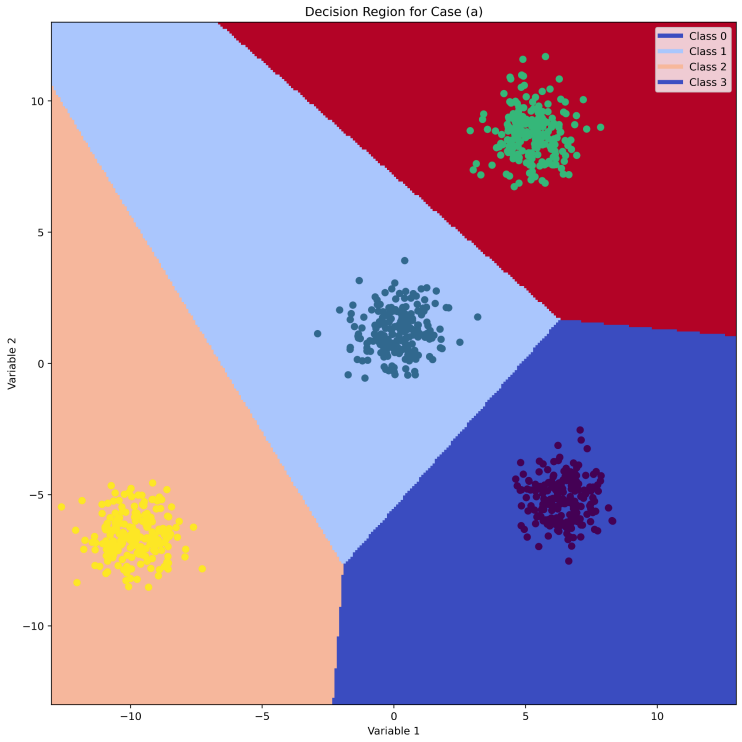


Figure 1.1: Decision Region for Case (a)

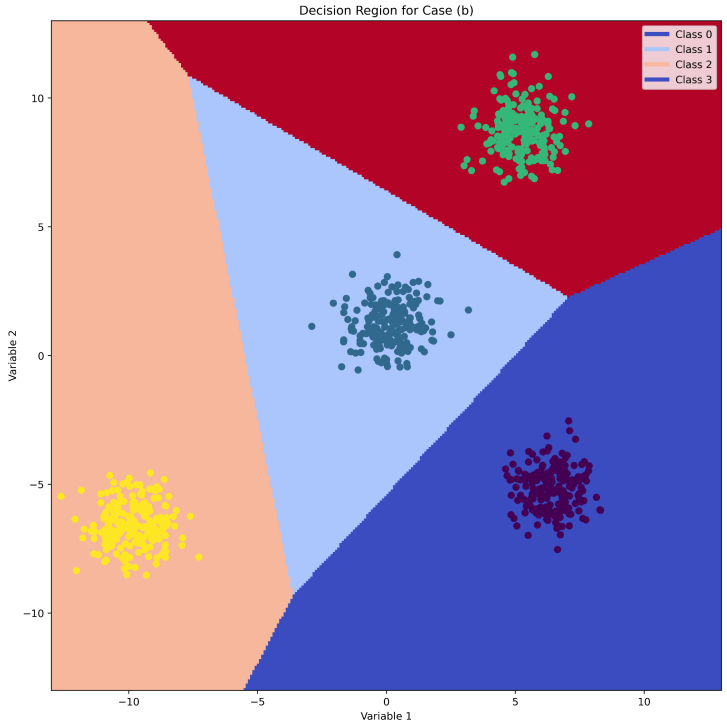


Figure 1.2: Decision Region for Case (b)

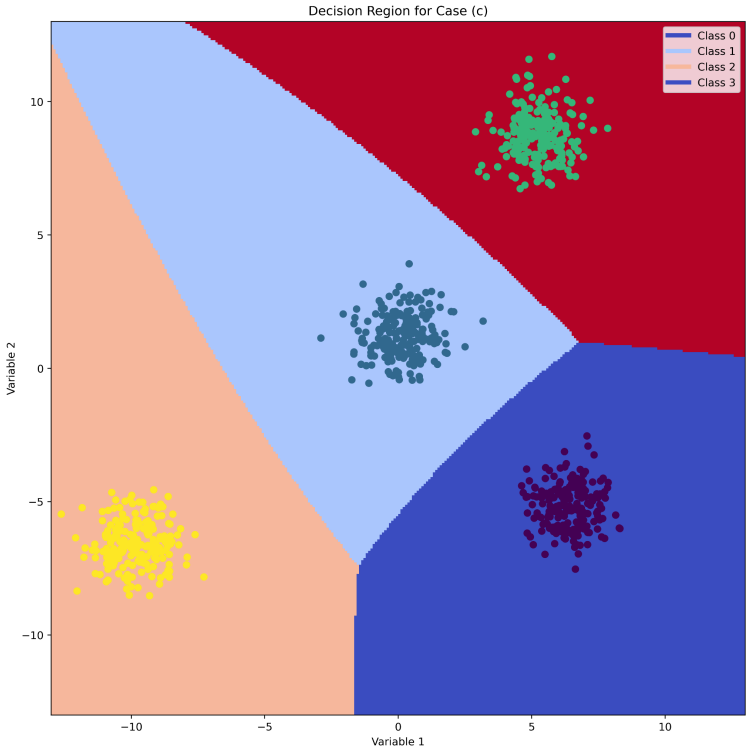


Figure 1.3: Decision Region for Case (c)

Section 2 : Bayes GMM

2.1 Task:

To classify datasets 1b(synthetic set) ,2A and 2B(real dataset) using Bayes GMM model

1. Using Bayes GMM with Full covariance matrix (for 1b,2a,2b)
2. Using Bayes GMM with Diagonal covariance matrix (for 1b,2a,2b)
3. Using Bayes KNN model (for 1b dataset)

2.2 Steps:

1. Import libraries, load the dataset (1B,2A,2B) also split class wise for further trainings,
2. Normalise data and split into train/val/test using 80/10/10.
3. Constructed class structure with functions for initializing using clustering ,E step, M step and optimisation for different GMM cases(Full/diagonal covariance).
4. Classified the points using GMM models per class and assigned label of the class giving max probability.
5. For real data sets with 36 feature vectors, probability for each feature vector is calculated and multiplied to find overall probability of the image belonging to a particular class.
6. Evaluation Accuracy tables/ confusion matrix and Decision region/level curves plots are formed.

2.3. Bayes classifier with GMM and Full covariance matrix (Data:1b)

Table 2.1: Accuracy Tables for Training/Validation/Test Data

(Accuracy between 0-1 given: Multiply with 100 for %Accuracy)

Accuracy/Clusters	Q/k=2	k=3	k=4	k=5	k=6	k=7
Training set	0.9117	0.9867	0.9883	0.9967	1.0	1.0
Validation set	0.8889	0.9778	0.9778	0.9778	1.0	1.0
Test set	0.8889	0.9556	0.9778	0.9778	1.0	1.0

Conclusion = Best model(*Highlighted) for this case is (**k=6**)as all accuracy are 100%. k=7 would take slightly more time in computation but will give the same result like k=6.

Table 2.2: Confusion matrix for best model

- **Confusion matrix for Training Data:**

Class	Class 1	Class 2	Class 3
Class 1	200	0	0
Class 2	0	200	0
Class 3	0	0	200

- **Confusion matrix for Validation Data:**

Class	Class 1	Class 2	Class 3
Class 1	16	0	0
Class 2	0	15	0
Class 3	0	0	14

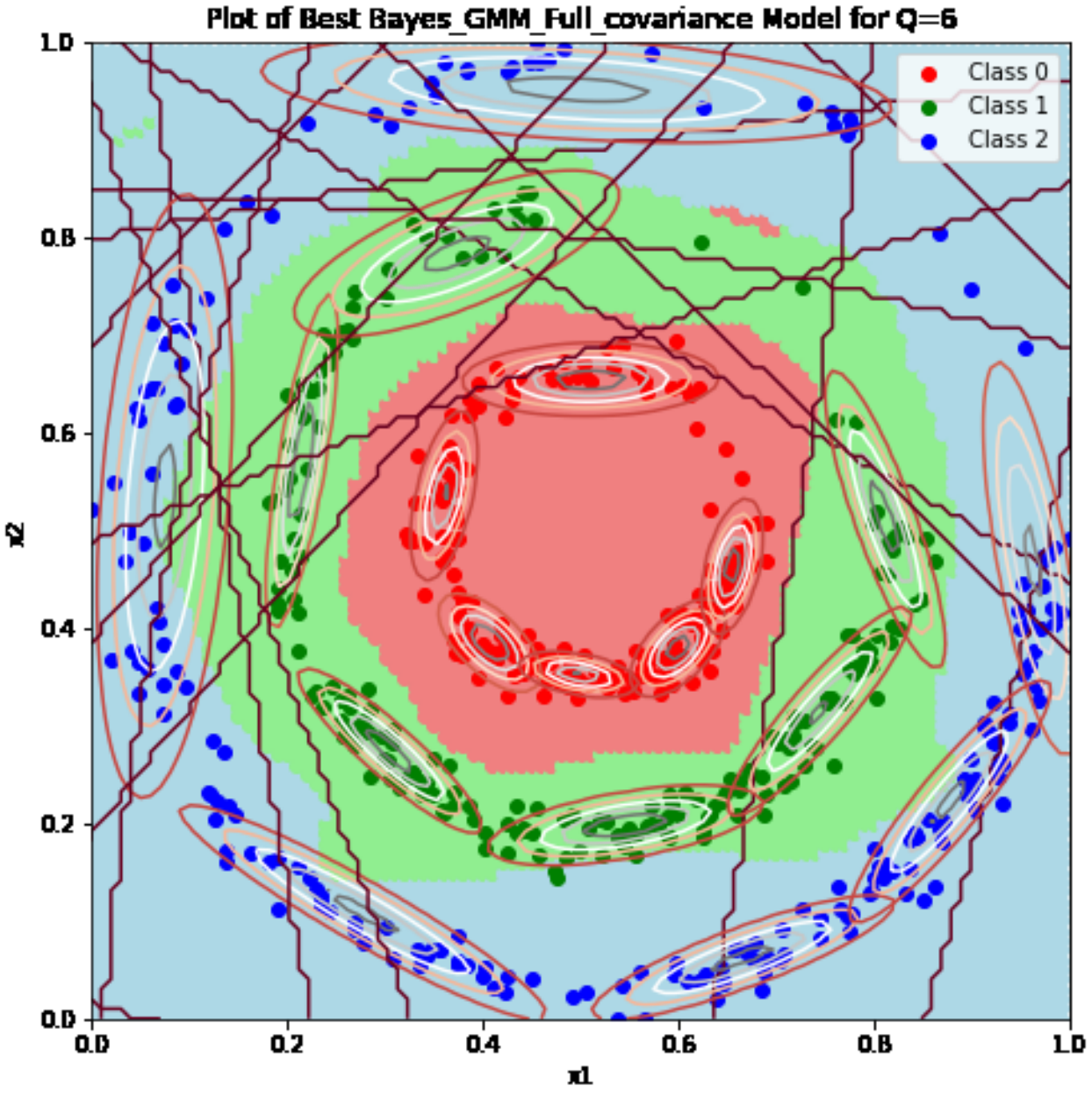
- **Confusion matrix for Test Data:**

Class	Class 1	Class 2	Class 3
Class 1	14	0	0
Class 2	0	15	0
Class 3	0	0	16

Conclusion: Shows all points are true positives and true negatives, 0 False positives and negatives.

Figure 2.1 Graph for best model (Plotted 1.Decision region, 2.Training pts 3.Level curves)

Full covariance GMM CASE (Q=6)



Conclusion:

- Level curves for **k=6** (**Best model**) are plotted per class , **Shape=Ellipse**
- The bigger lines are just part of level curves which are out of scope of image and kept coming in the plot even though I fixed contour no.=5 for each ellipse and cant be removed.
- It's seen that all Training points are classified right as shown in decision regions

2.4: Bayes classifier with GMM and Diagonal covariance matrix(Data:1b)

Table 2.3: Accuracy Tables for Training/Validation/Test Data

(Accuracy between 0-1 given: Multiply with 100 for %Accuracy)

Accuracy/Cluster	Q/k =2	k=3	k=4	k=5	k=6	k=7
Training set	0.9000	0.9067	0.9483	0.9417	0.95	0.895
Validation set	0.9556	0.9111	0.9333	1.0000	0.8889	0.3556
Test set	0.8889	0.8	0.8667	0.8667	0.8889	0.3311

Conclusion:

- **Best model**(Highlighted) for this case is (k=5 *highlighted) as validation accuracy is best among all other models.
- Above k=5 model started overfitting giving low accuracy on validation and test set

Table 2.4: Confusion matrix for best model (k=5)

- **Confusion matrix for Training Data:**

Class	Class 1	Class 2	Class 3
Class 1	200	0	0
Class 2	25	175	0
Class 3	0	10	190

- **Confusion matrix for Validation Data:**

Class	Class 1	Class 2	Class 3
Class 1	16	0	0
Class 2	0	15	0
Class 3	0	0	14

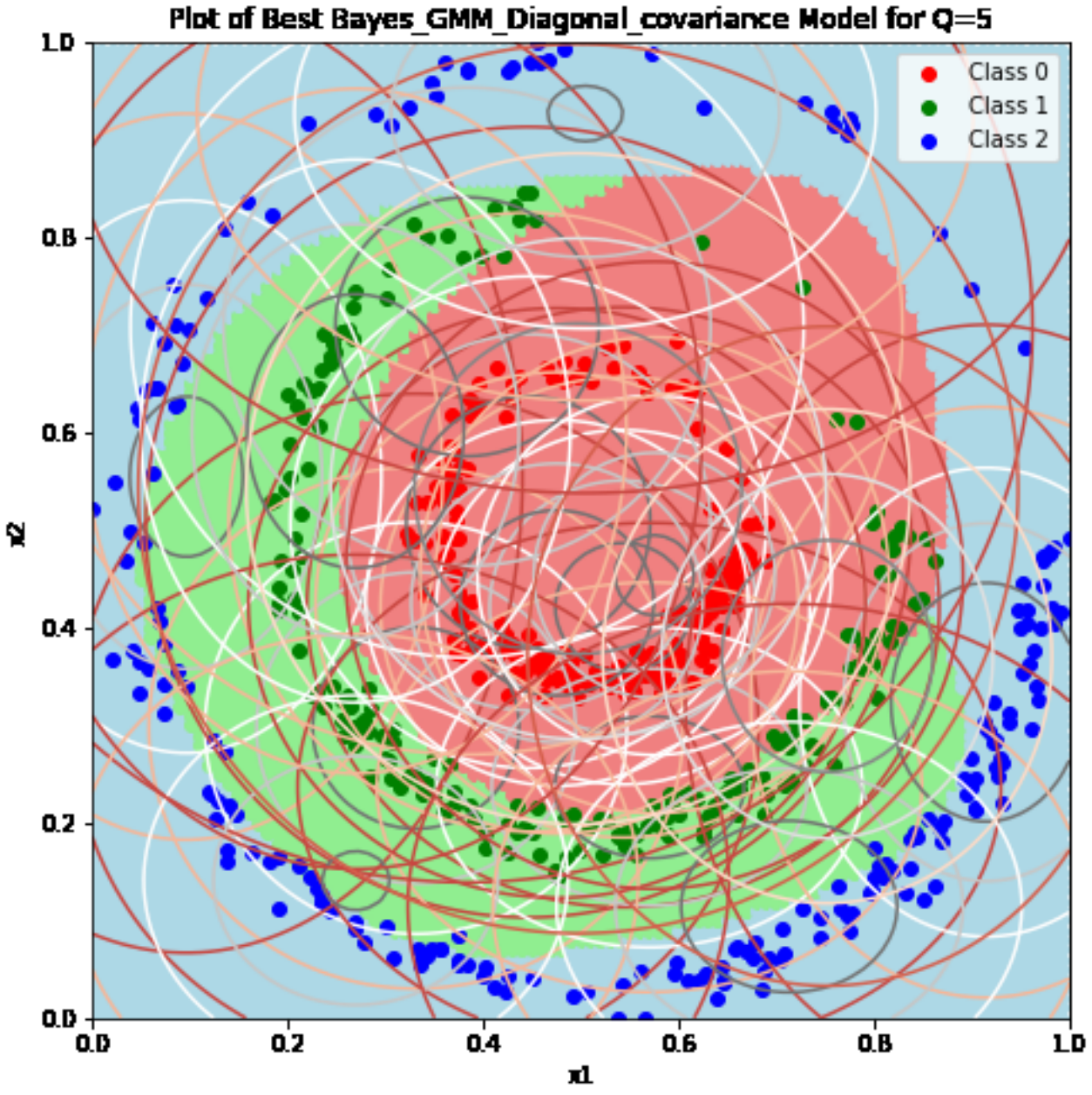
- **Confusion matrix for Test Data:**

Class	Class 1	Class 2	Class 3
Class 1	14	0	0
Class 2	6	9	0
Class 3	0	0	16

Conclusion: Compared to the full covariance matrix, In this case confusion matrix have some points False positive and False negative in training and test data but not in validation data.

Figure 2.2 Graph for best model(Plotted 1.Decision region, 2.Training pts 3.Level curves)

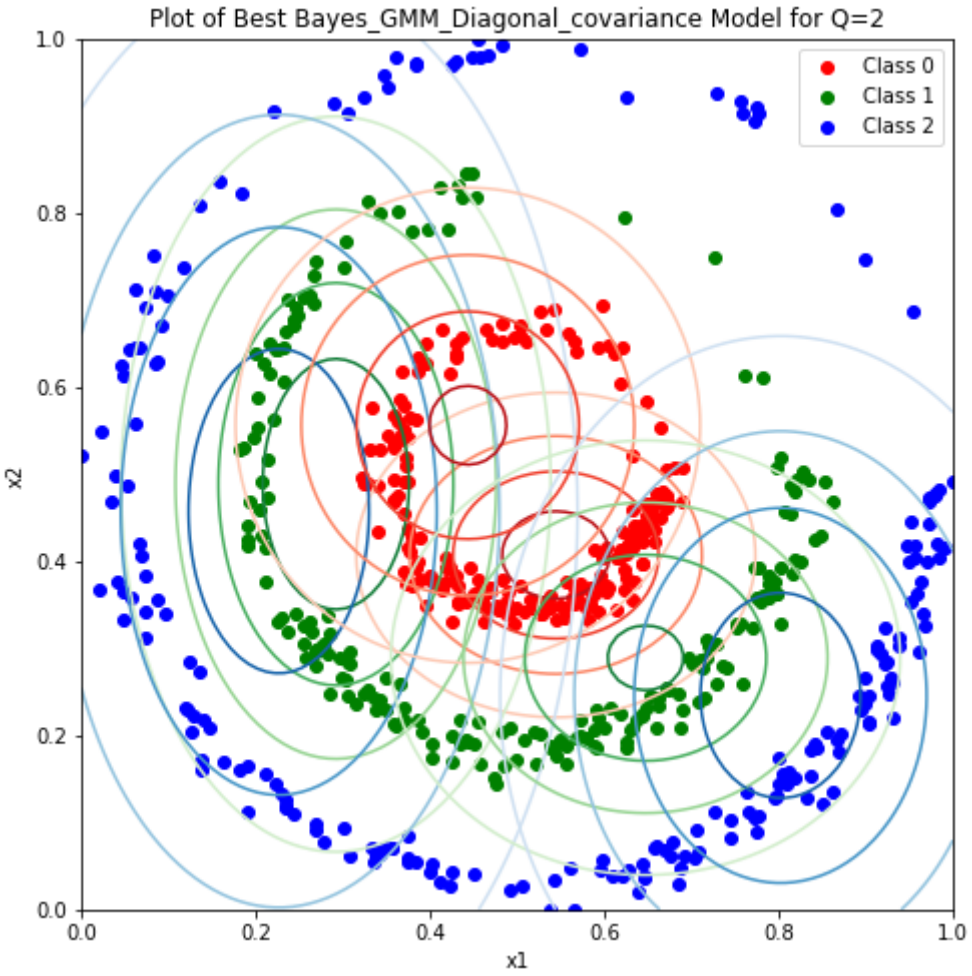
Diagonal covariance GMM CASE (k/Q=5)



Conclusion:

- This time level curves are 5 per class,SHAPE=Ellipses with axis parallel to coordinate axes.
- Ellipses have a diagonal covariance matrix so they are more wider and bigger so the figure looks too congested.
- Yet decision region plots show that most of the points are correctly classified

- Just to show that ellipses axes are parallel to x,y axis, we have shown level curves plot with (Q=2 per class) for less congestion.



- Here we can see that for Q/K=2 case ellipses are clearly seen and are parallel to the (x,y) coordinate axis and are wider and bigger to the case where the covariance matrix is full.

2.5. Bayes classifier with KNN (Data:1b)

Table 2.5: Accuracy Tables for Training/Validation/Test Data

(Accuracy between 0-1 given: Multiply with 100 for %Accuracy)

Accuracy/ Knn	knn = 10	knn = 20
Training set	0.9917	0.9533
Validation set	1.0	0.9333
Test set	0.9778	0.9556

Conclusion: Best model is (knn=10 *highlighted) as for knn=20 it overfits as validation accuracy decreases.

Table 2.6: Confusion matrix for best model

- **Confusion matrix for Training Data:**

Class	Class 1	Class 2	Class 3
Class 1	200	0	0
Class 2	2	198	0
Class 3	0	3	197

- **Confusion matrix for Validation Data:**

Class	Class 1	Class 2	Class 3
Class 1	16	0	0
Class 2	0	15	0
Class 3	0	0	14

- **Confusion matrix for Test Data:**

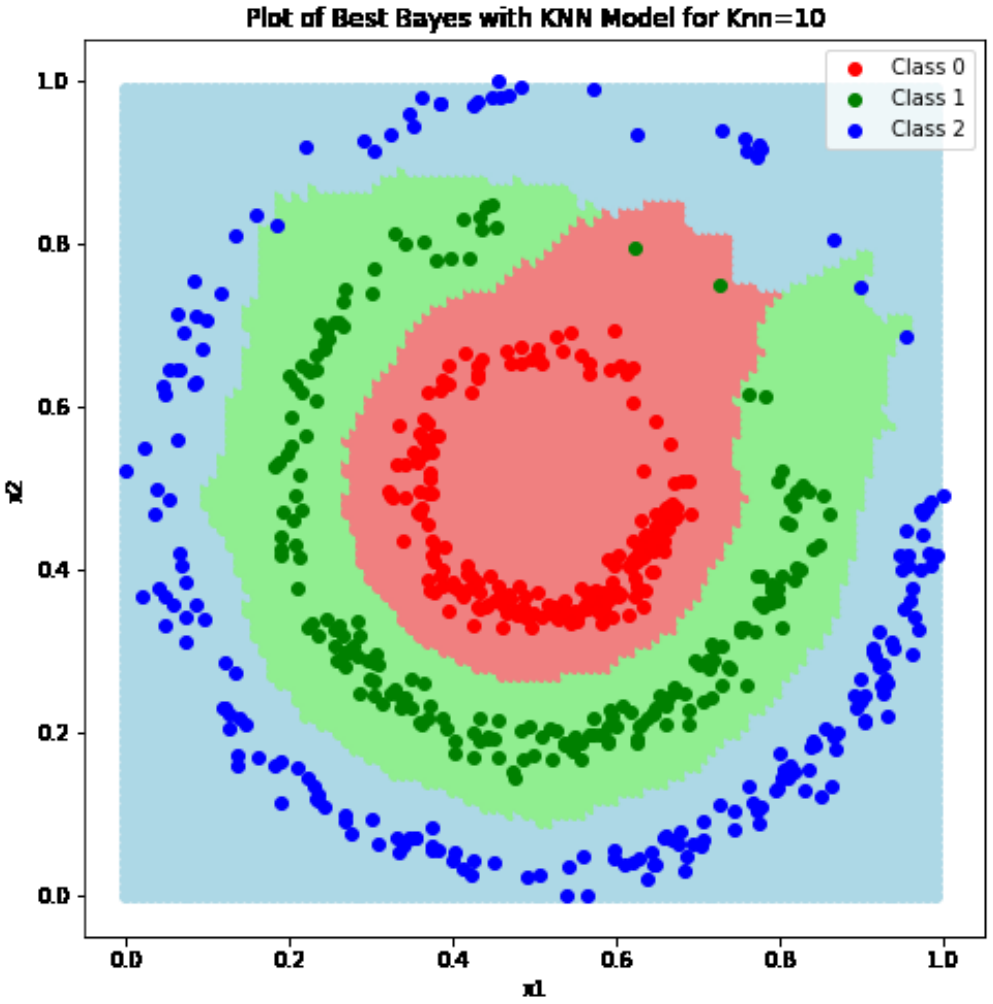
Class	Class 1	Class 2	Class 3
Class 1	14	0	0
Class 2	0	14	1
Class 3	0	0	16

Conclusion:

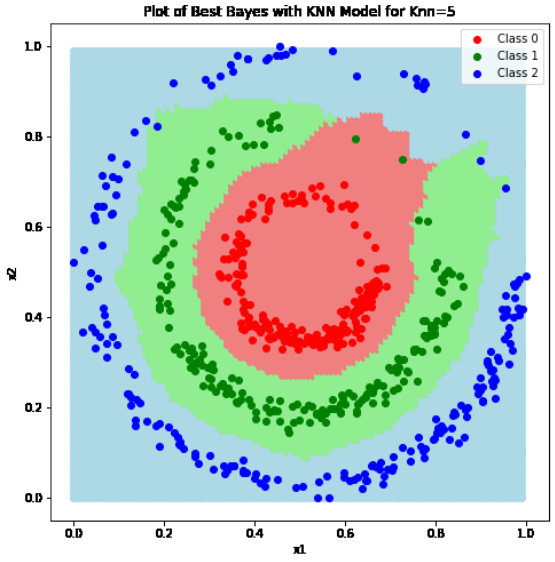
- There are no false positives or false negatives for validation data.
- Very less false positives and negatives in case of train and test data as accuracy is high for the model as seen in table.

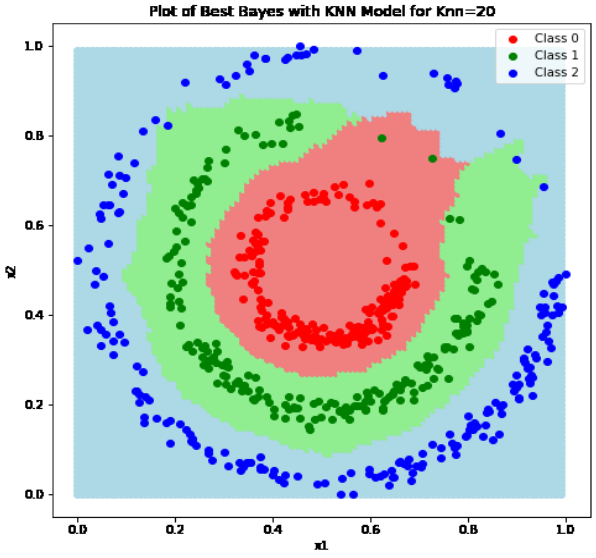
Figure 2.3 Graph for best model (Plotted 1.Decision region, 2.Training pts 3.Level curves)

Bayes with KNN CASE (knn=10)



Other e.g k=5,20 (Decision boundaries almost look same)





2.6: Bayes classifier with GMM and Full covariance matrix (Data:2a)

Table 2.7: Accuracy Tables for Training/Validation/Test Data

(Accuracy between 0-1 given: Multiply with 100 for %Accuracy)

Accuracy/Clusters	Q=1	Q=2	Q=3	Q=4
Training set	1.0	0.1853	0.228	0.228
Validation set	0.2152	0.2152	0.2152	0.1835
Test set	0.2468	0.2468	0.2468	0.1456

Conlusion:

- Best Model= (Q=1 *highlighted), giving 100% accuracy on training set but 21.52% on validation set and 24.68% on test set, I also compared results with scikit learn classifiers and results were like this only, those were also not giving good results on val/test set.

Table 2.8: Confusion matrix for best model (Q=1)

- **Confusion matrix for Training Data:**

Class	Class 1	Class 2	Class 3	Class 4	Class 5
Class 1	251	0	0	0	0
Class 2	0	182	0	0	0
Class 3	0	0	215	0	0
Class 4	0	0	0	204	0
Class 5	0	0	0	0	249

- **Confusion matrix for Test Data:**

Class	Class 1	Class 2	Class 3	Class 4	Class 5
Class 1	39	0	0	0	0
Class 2	23	0	0	0	0
Class 3	28	0	0	0	0
Class 4	32	0	0	0	0
Class 5	36	0	0	0	0

Conclusion:

- Model did well on the training set but performed poorly on validation and test dataset, seems like val/test data is not generalized well and overfitting happened on training set as val/test accuracy is very less compared to training set accuracy.
- Val/test Data was not giving good results on scikit learn models too(just for comparing) but train set was giving good results.

2.7: Bayes classifier with GMM and Diagonal covariance matrix (Data:2a)

Given data has 1 feature vector/image and Dimension=24

Table 2.9: Accuracy Tables for Training/Validation/Test Data
(Accuracy between 0-1 given: Multiply with 100 for %Accuracy)

Accuracy/Clusters	Q=1	Q=2	Q=3	Q=4	Q=5
Training set	0.6104	0.6594	0.7021	0.7075	0.4620
Validation set	0.3038	0.4177	0.4557	0.4304	0.3544
Test Set	0.2848	0.443	0.4620	0.4620	0.4304

Conlusion:

- (Q=3 *HIGHLIGHTED), giving 70% accuracy on the training set but 45.57% on validation set and 46.2% on test set better than other cases.
- Performed better than Full covariance case for the same data.

Table 2.10: Confusion matrix for best model (Q=1)

- **Confusion matrix for Training Data:**

Class	Class 1	Class 2	Class 3	Class 4	Class 5
Class 1	230	5	9	4	3
Class 2	5	122	20	14	21
Class 3	15	7	128	21	44
Class 4	10	5	22	148	19
Class 5	18	14	49	20	148

- **Confusion matrix for Validation Data:**

Class	Class 1	Class 2	Class 3	Class 4	Class 5
Class 1	11	2	6	7	8
Class 2	2	14	1	4	8
Class 3	0	2	12	14	6
Class 4	0	4	1	17	4
Class 5	3	1	5	7	19

- **Confusion matrix for Test Data:**

Class	Class 1	Class 2	Class 3	Class 4	Class 5
Class 1	8	7	5	7	12
Class 2	4	11	1	3	4
Class 3	0	5	9	11	3
Class 4	0	1	3	27	1
Class 5	6	1	11	6	12

Conclusion:

- Best model in this case performed better on the val/test set and reduced overfitting as training accuracy was reduced to 70% and val/test accuracy increased to (45-46%) compared to the full covariance case.
- Less False positives and negatives can be seen in Val/Test Data Confusion matrix.

2.8: Bayes classifier with GMM and Full covariance matrix (Data:2B)

Given data has 36 feature vectors/image and Dimension=23

Table 2.11: Accuracy Tables for Training/Validation/Test Data

(Accuracy between 0-1 given: Multiply with 100 for %Accuracy)

Accuracy/Cluster	Q=1	Q=2	Q=3	Q=4
Training set	0.7321	0.7943	0.7320	0.4561
Validation set	0.6772	0.7231	0.6532	0.5632
Testing set	0.6456	0.7145	0.6456	0.5678

Conlusion:

- Best model= (Q=2 *Highlighted) Best results for image data till now (out of all models used till now for 2A data).

Table 2.12: Confusion matrix for best model (Q=2)

- **Confusion matrix for Training Data:**

Class	Class 1	Class 2	Class 3	Class 4	Class 5
Class 1	185	46	5	4	11
Class 2	17	147	5	9	4
Class 3	9	11	127	35	33
Class 4	4	13	12	161	14
Class 5	5	19	16	23	186

- **Confusion matrix for Validation Data:**

Class	Class 1	Class 2	Class 3	Class 4	Class 5
Class 1	25	7	1	1	0
Class 2	3	19	0	6	1
Class 3	1	1	19	11	2
Class 4	0	0	2	24	0
Class 5	2	5	5	3	20

- **Confusion matrix for Test Data:**

Class	Class 1	Class 2	Class 3	Class 4	Class 5
Class 1	26	10	1	1	1
Class 2	4	14	0	4	1
Class 3	0	2	11	13	2
Class 4	0	0	4	26	2
Class 5	1	2	2	6	25

2.9: Bayes classifier with GMM and Diagonal covariance matrix (Data:2B)

Given data has 36 feature vectors/image and Dimension=23

Table 2.11: Accuracy Tables for Training/Validation/Test Data

(Accuracy between 0-1 given: Multiply with 100 for %Accuracy)

Accuracy/Cluster	Q=1	Q=2	Q=3	Q=4
Training set	0.2279	0.228	0.2105	0.18
Validation set	0.2151	0.2152	0.2150	0.2067
Testing set	0.2468	0.2468	0.2421	0.2421

Conclusion:

- (Q=2 * Highlighted one), (Q=1) is almost the same but both models are not that good compared to Full Covariance case .

Table 2.12: Confusion matrix for best model (Q=1,2)

- **Confusion matrix for Training Data:**

Class	Class 1	Class 2	Class 3	Class 4	Class 5
Class 1	251	0	0	0	0
Class 2	182	0	0	0	0
Class 3	215	0	0	0	0
Class 4	204	0	0	0	0
Class 5	249	0	0	0	0

- **Confusion matrix for Validation Data:**

Class	Class 1	Class 2	Class 3	Class 4	Class 5
Class 1	34	0	0	0	0
Class 2	29	0	0	0	0
Class 3	34	0	0	0	0
Class 4	26	0	0	0	0
Class 5	35	0	0	0	0

- **Confusion matrix for Test Data:**

Class	Class 1	Class 2	Class 3	Class 4	Class 5
Class 1	39	0	0	0	0
Class 2	23	0	0	0	0
Class 3	28	0	0	0	0
Class 4	32	0	0	0	0
Class 5	36	0	0	0	0

Conclusion:

- Full covariance case gave better results on all train/val/test data compared to diagonal covariance case.
- Confusion matrix also doesn't look good for a diagonal case.

Section 3 : KNN

3.1 Task:

To classify dataset based on KNN Method for

1 : Linearly separable data

2: Nonlinearly separable data

3.2 Steps:

1. Extract the Data.
2. Split the given test data for Validation and Testing.
3. Run it through the KNN Algorithm and get the Accuracies.
4. Plot the Decision boundary plot for visualizing the dataset.

3.3 Linearly Separable Data (dataset 1a):

Table 3.1 : Accuracy Table:

Dataset / K	**K = 1**	**K = 7**	**K = 15**
Training Set	1.0	1.0	1.0
Validation Set	1.0	1.0	1.0
Test Set	1.0	1.0	1.0

Best Configuration :

- All the above cases have maximum Accuracy.
- Computation wise K=1 will be the faster one

Table 3.2 : Confusion Matrix

All the above cases have maximum Accuracy but it's good to opt. for a medium range value of k to be trusted
Below is the matrix for K = 7.

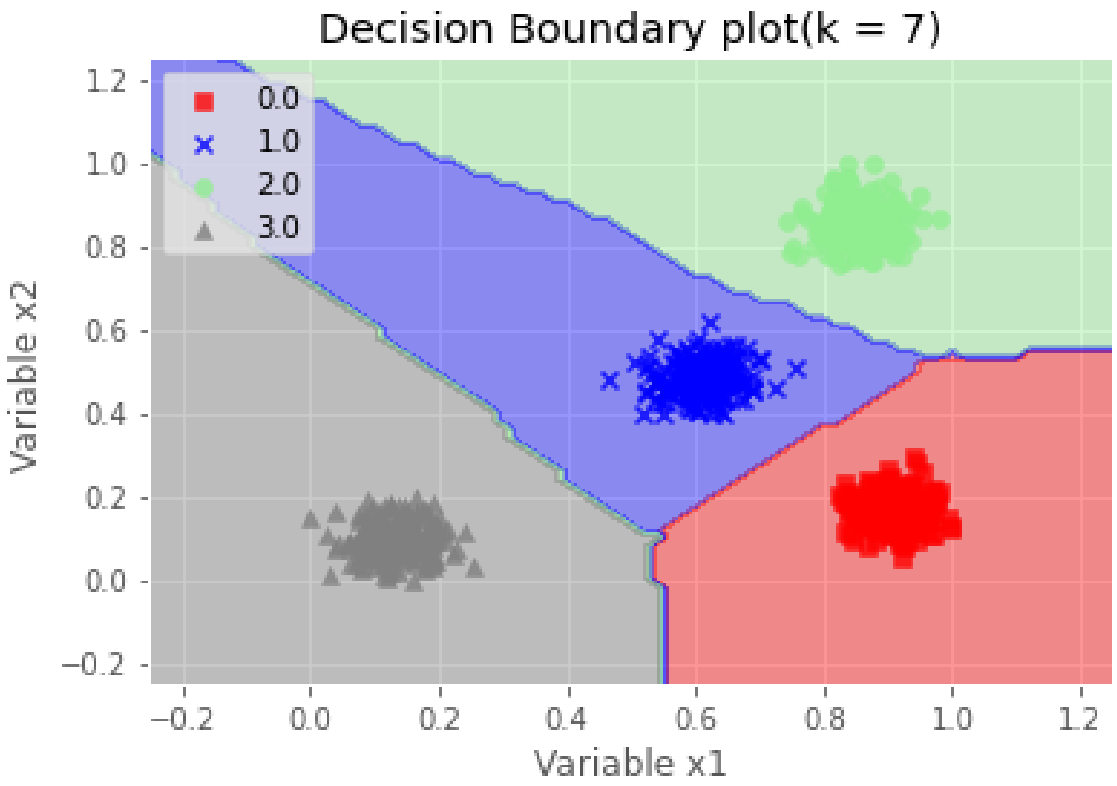
- **Validation Dataset**

Class	Class 1	Class 2	Class 3	Class 4
Class 1	13	0	0	0
Class 2	0	14	0	0
Class 3	0	0	17	0
Class 4	0	0	0	16

- **Test Dataset**

Class	Class 1	Class 2	Class 3	Class 4
Class 1	17	0	0	0
Class 2	0	16	0	0
Class 3	0	0	13	0
Class 4	0	0	0	14

Figure 3.1 : Plot for the best Model :



3.4 Linearly Separable Dataset (Dataset 1b)

Table 3.3 : Accuracy Table

Dataset / K	K = 1	K = 7	K = 15
Training Set	1.0	1.0	1.0
Validation Set	1.0	1.0	1.0
Test Set	1.0	1.0	1.0

Best Configuration :

All the above cases have maximum Accuracy. For Real world applications higher K value is advised .

Table 3.4 : Confusion Matrix

All the above cases have maximum Accuracy but it's good to opt. for a medium range value. Below is the matrix for K = 7.

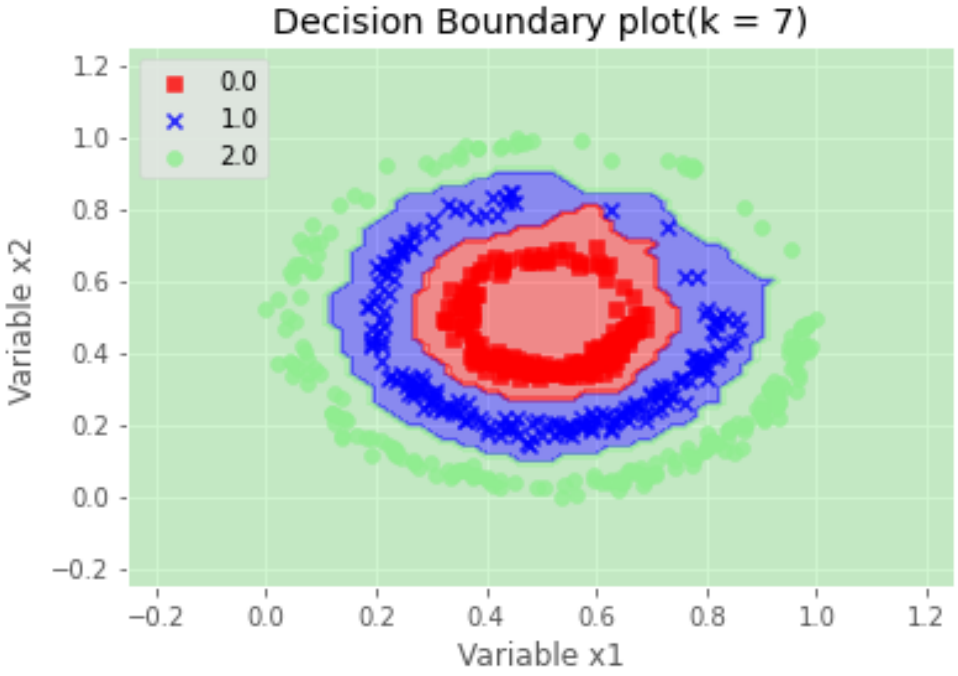
- **Validation Dataset**

Classes	Class 1	Class 2	Class 3
Class 1	13	0	0
Class 2	0	14	0
Class 3	0	0	18

- **Test Dataset**

Classes	Class 1	Class 2	Class 3
Class 1	17	0	0
Class 2	0	12	0
Class 3	0	0	16

Figure 3.2 : Plot for the best Model
For k = 7


Decision Boundary plot(k = 7)
0.0
1.0
2.0
Variable x2
Variable x1
1.2
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.2
0.0
0.2
0.4
0.6
0.8
1.0
1.2