Assignment 2 & 3

(Bayes Classifier using GMM)

CS669: Pattern Recognition

<u>Group -16</u>

Tulsi Upadhyay V22024
Vaibhav Sharma T23156

Objective 2
Gaussian Mixture Model 2
Density Estimation using GMM 3
Bayes Decision Rule 4
Results and Plots Dataset-1 6
Iterations V/S Likelihood Plots 9
Class-3 9
Comparison with Assignment-1 11
Results and Plots Dataset-2(a) 13

Objective

We are given two datasets:

Dataset 1: Nonlinearly separable classes: 2-dimensional data of 2 or 3 classes that are non linearly separable. (Same data used in Assignment 1.)
Dataset 2: Real world data set:
(a) Two-dimensional speech dataset (vowel data) used in Assignment 1
(b) 3 class scene image datasets
(c) Cervical cytology (cell) image dataset

- **Task** :
  1. Build the Bayes classifier using GMM on Dataset-1, Dataset-2(a) and, Dataset-2(b) Parameters of GMM are to be initialized using K-means clustering
  2. Segment the cell images by clustering the local feature vectors from cell image datasets into 3 groups using
     (a) K-means clustering and
     (b) Modified K-means clustering (using Mahalanobis distance).

Gaussian Mixture Model

In Assignment -1,we made the assumption that data of each class coming from gaussian distribution that rarely happens in real scenario. GMM helps in capturing the underlying density/structure of data effectively.It can effectively approximate any distribution that data may follow with proper choice of components.

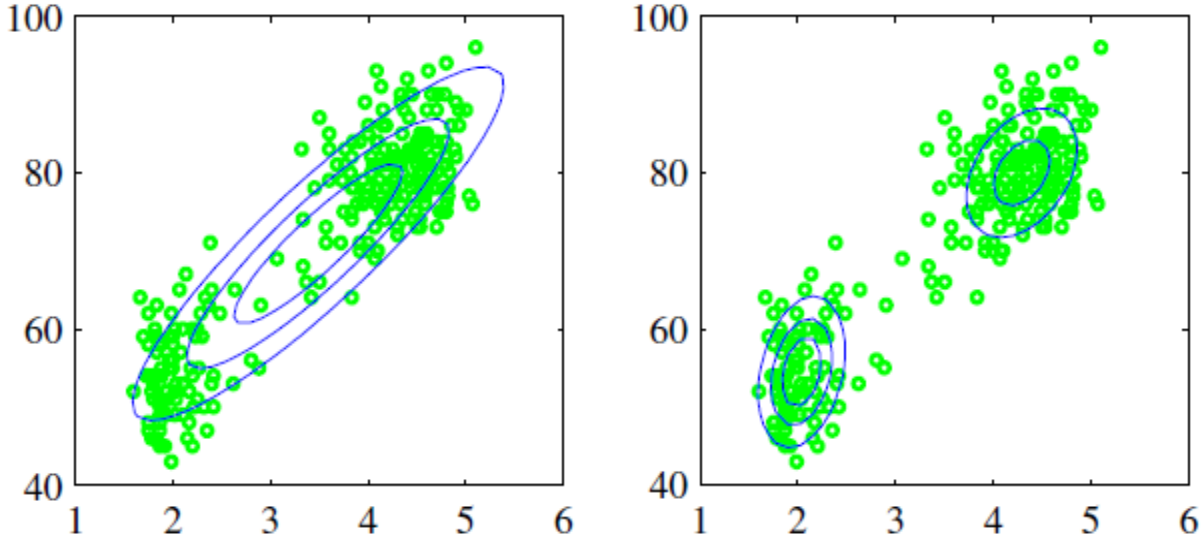


Fig : (a) **Fig : (b)**

Fig : 1Contours of (a) SIngle Gaussian distribution
(b) combination of 2 Gaussian distribution

From Fig :1, it is clear that the above data's distribution can be better approximated by a combination of 2 gaussian distributions

GMM is used for both density estimation and clustering.GMM is quite similar to the KMeans, it's a probabilistic version of KMeans. KMeans uses a distance based approach, and GMM uses a probabilistic approach.In KMeans each data point belongs to only one cluster i.e it is hard clustering while GMM is soft clustering i.e a data point can belong to multiple clusters with varying degrees of membership

Density Estimation using GMM

1. Decide the number of clusters.
2. Initiate mean, covariance, and weight parameter per cluster. (Initialized using KMeans clustering)

3. Expectation step (E-Step):Calculate the probability of each data point belonging to each cluster (*Responsibility term*).

$$\gamma(z_{nk}) = \frac{w_k \; N(\overline{x_n} \mid \overline{\mu_k}, \Sigma_k)}{\sum_{k=1}^{K} w_k \; N(\overline{x_n} \mid \overline{\mu_k}, \Sigma_k)}$$

then evaluate the total-datalikelihood function using the current estimate for the parameters.

$$l(\overline{\theta}) = \sum_{n=1}^{N_i} \ln \sum_{k=1}^{K} w_k \; N(\overline{x_n} \mid \overline{\mu_k}, \Sigma_k)$$

4. Maximization step (M-Step) : Update the previous mean, covariance, and weight parameters to maximize the expected likelihood found in E-Step.

- $$\overline{\mu_{ik}} = \frac{\sum_{k=1}^{K} \gamma(z_{nk}) \, \overline{x_{in}}}{\sum_{k=1}^{K} \gamma(z_{nk})}$$
- $$\Sigma_{ik} = \frac{\sum_{n=1}^{N_i} \gamma(z_{nk}) \, (\overline{x_{in}} - \overline{\mu_{ik}})(\overline{x_{in}} - \overline{\mu_{ik}})^T}{\sum_{n=1}^{N_i} \gamma(z_{nk})}$$
- $$w_{ik} = \frac{\sum_{n=1}^{N_i} \gamma(z_{nk})}{N_i}$$

5. Repeat steps 3 & 4 until the model converges.

Bayes Decision Rule

$$P(C_i \mid x) = \frac{p(x \mid C_i) \, P(C_i)}{P(x)}$$

$$p(x|C_i) = \sum_{k=1}^{K} w_k \; N(\overline{x_n} \mid \overline{\mu_k}, \Sigma_k \;) \qquad \Rightarrow GMM$$

Class Label for x $= arg\, max\, P(C_i|x) \qquad \forall\, i = 1, 2, \ldots, M$

Results and Plots

Dataset-1

K = 1

Accuracy : 86.444%
Mean - precision: 0.86515
Mean - recall: 0.86445
Mean - F measure: 0.86474

	Class-1	Class-2	Class-3
Precision	0.79084	0.90667	0.89795
Recall	0.80667	0.90667	0.88
F Measure	0.79867	0.90667	0.88889

K = 2

Accuracy : 99.55%
Mean - precision: 0.99561
Mean - recall: 0.99555
Mean - F measure: 0.99555

	Class-1	Class-2	Class-3
Precision	1.0	1.0	0.98684
Recall	0.98667	1.0	1.0
F Measure	0.99328	1.0	0.99337

K = 4

Accuracy : 100%
Mean - precision: 1.0
Mean - recall: 1.0
Mean - F measure: 1.0

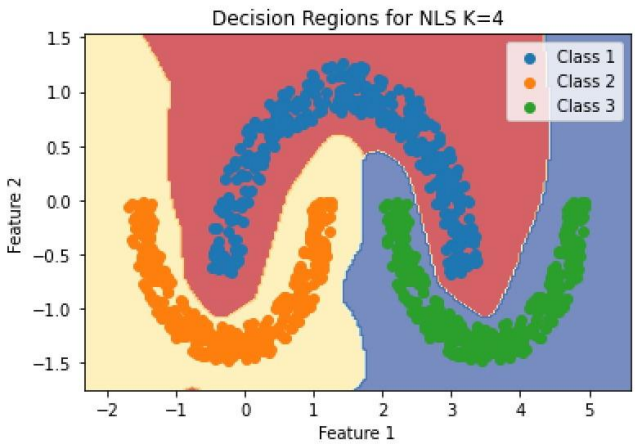
	Class-1	Class-2	Class-3
Precision	1.0	1.0	1.0
Recall	1.0	1.0	1.0
F Measure	1.0	1.0	1.0

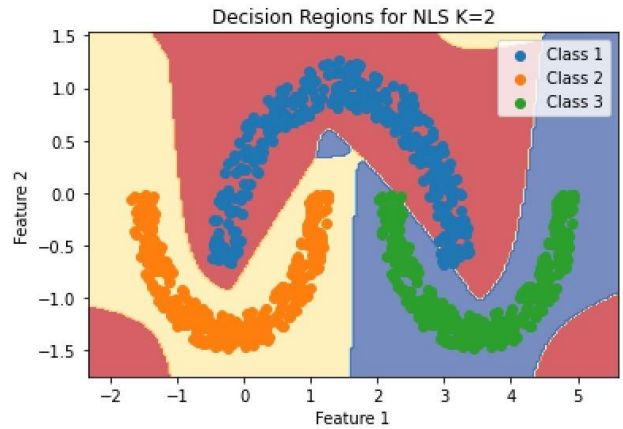
Confusion Matrix for K=4

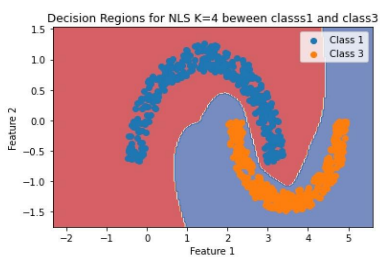
	Class1	Class2	Class2
Class1	150	0	0
Class2	0	150	0
Class3	0	0	150

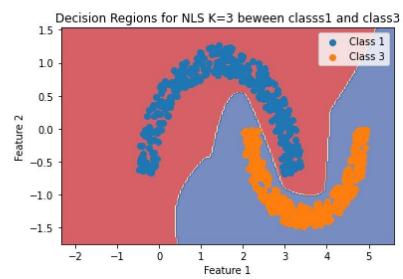
Confusion Matrix for K=2

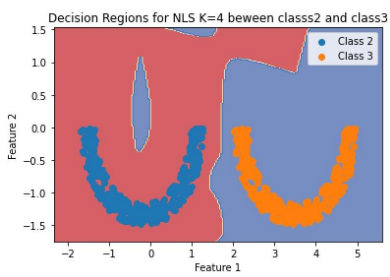
	Class1	Class2	Class3
Class1	148	0	0
Class2	0	150.0	0
Class3	2	0	150.0

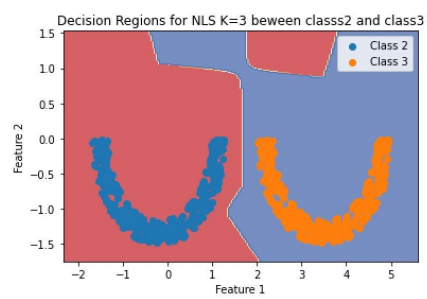


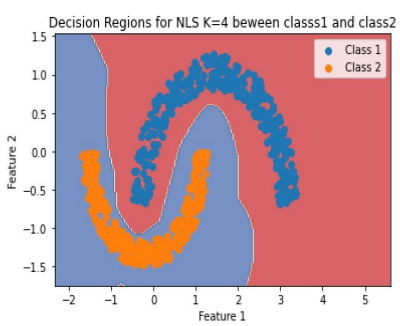


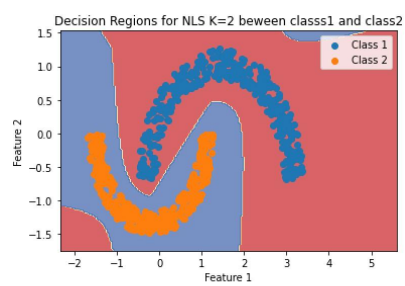


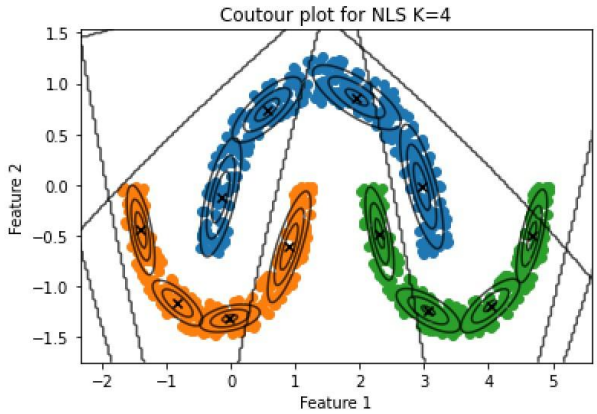


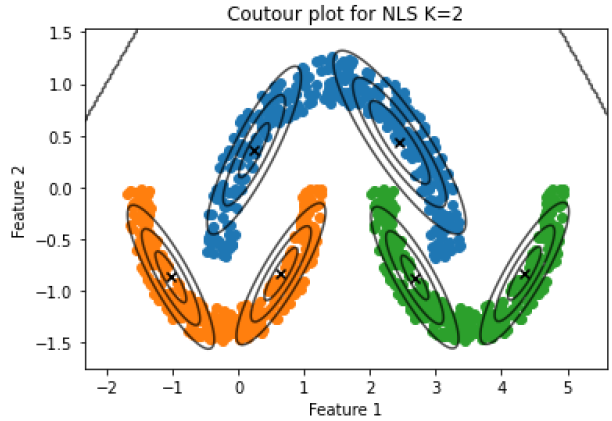






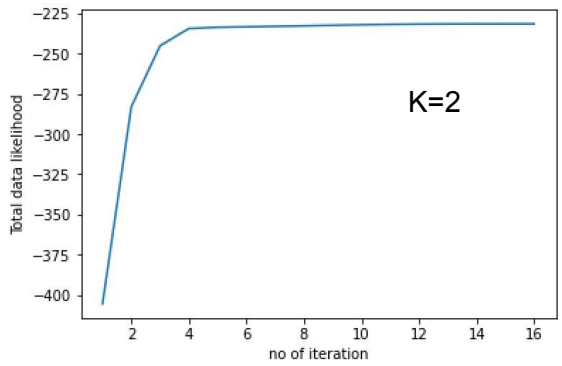


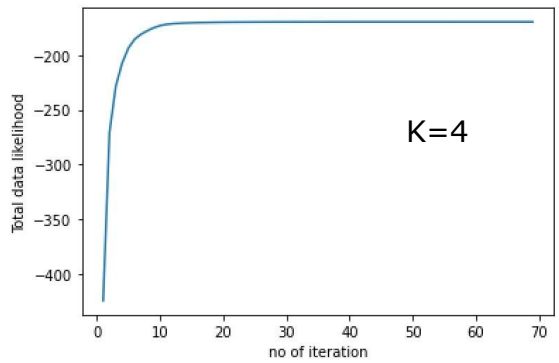

Coutour plot for NLS K=4
Feature 2
Feature 1


Coutour plot for NLS K=2
Feature 2
Feature 1

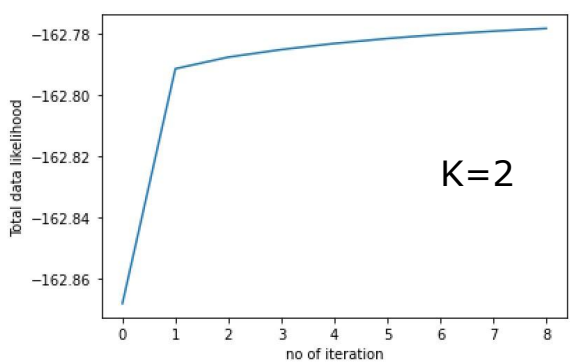
Iterations V/S Likelihood Plots

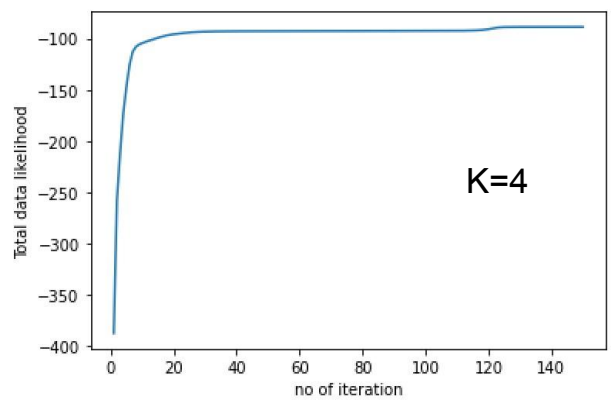
Class -1



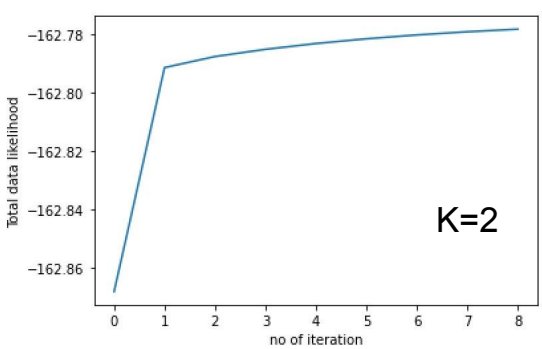


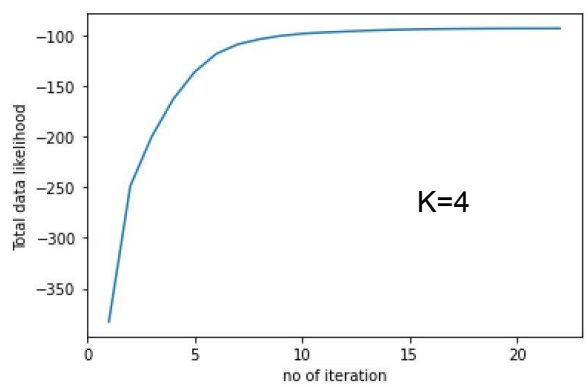
Class -2





Class-3





For k=2 we have 99.55% accuracy and for k=4 we have 100% accuracy. However, the number of iterations required for convergence of the E-M algorithm in K=4 is more than that for k=2.

So there is **Trade off between accuracy and speed between the** two models i.e k=2 and k=4.

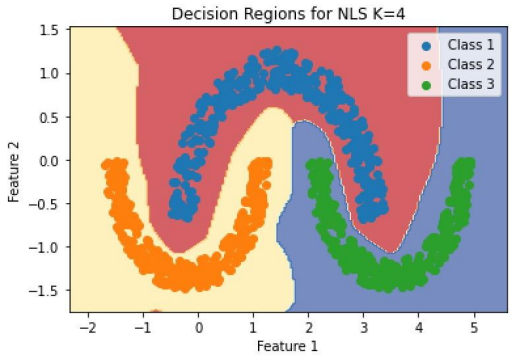
Thus,

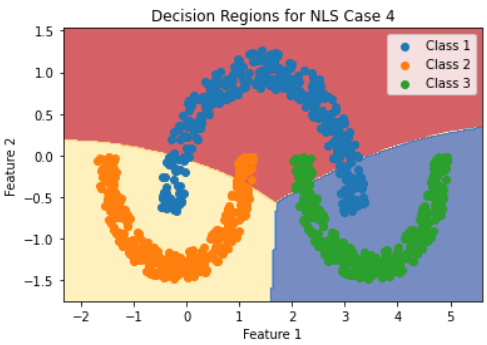
- If slightly lower accuracy is acceptable then 2-GMM is best.
- If computational resources and time constraints are limiting factors then also 2-GMM is best.

In summary, we can say if accuracy is of paramount importance then 4-GMM gives 100% accuracy and if speed is a critical factor and slightly reduced accuracy is acceptable then prefer 2-GMM.

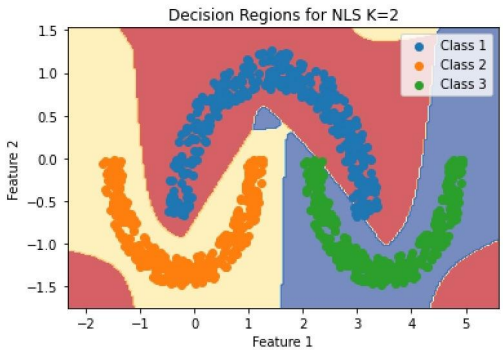
Comparison with Assignment-1

Plot comparison





From Assignment-1



Confusion Matrix comparison

	Class1	Class2	Class3
Class1	121	14	18
Class2	14	136	0
Class3	15	0	132

	Class1	Class2	Class3
Class1	150	0	0
Class2	0	150	0
Class3	0	0	150

	Class1	Class2	Class3
Class1	148	0	0
Class2	0	150.0	0
Class3	2	0	150.0

From confusion matrix and Plots it is clear GMM based Bayes classifier work better than Unimodal Bayes Classifier.

Results and Plots

Dataset-2(a)

K = 1

Accuracy :97.025%
Mean - precision: 0.97224
Mean - recall: 0.97124
Mean - F measure: 0.97162

	Class-1	Class-2	Class-3
Precision	0.96112	0.95703	0.99857
Recall	0.95223	0.98525	0.97625
F Measure	0.95665	0.97093	0.98729

K=2

Accuracy : 97.58%
Mean - precision: 0.97554
Mean - recall: 0.97553
Mean - F measure: 0.97552

	Class-1	Class-2	Class-3
Precision	0.96154	0.96787	0.99719
Recall	0.96302	0.96916	0.99441
F Measure	0.96227	0.96851	0.99580

K = 4

Accuracy : 97.39%
Mean - precision: 0.97357
Mean - recall: 0.97372
Mean - F measure: 0.97366

	Class-1	Class-2	Class-3
Precision	0.95022	0.97051	1.0
Recall	0.97072	0.97051	0.98044
F Measure	0.96036	0.97050	0.99013

K = 8

Accuracy : 97.86%
Mean - precision: 0.97896
Mean - recall: 0.97798
Mean - F measure: 0.97839

	Class-1	Class-2	Class-3
Precision	0.97496	0.96335	0.99858
Recall	0.95993	0.98659	0.98743
F Measure	0.96739	0.97483	0.99297

K = 16

Accuracy : 96.96%
Mean - precision: 0.96922
Mean - recall: 0.96968
Mean - F measure: 0.9693

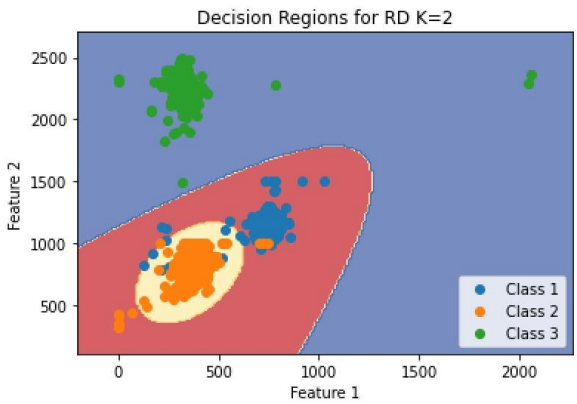
	Class-1	Class-2	Class-3
Precision	0.94277	0.96491	1.0
Recall	0.96456	0.95845	0.98603
F Measure	0.95354	0.96166	0.99296

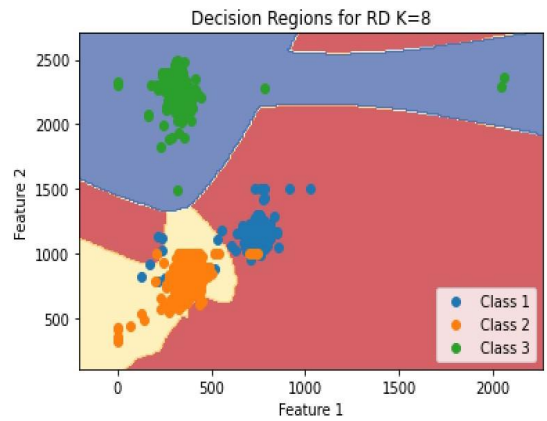
Confusion Matrix for K=2

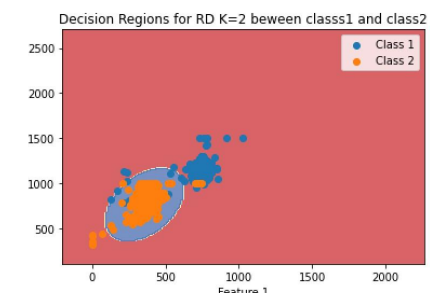
	Class1	Class2	Class3
Class1	625	23	2
Class2	22	723	2
Class3	2	0	712

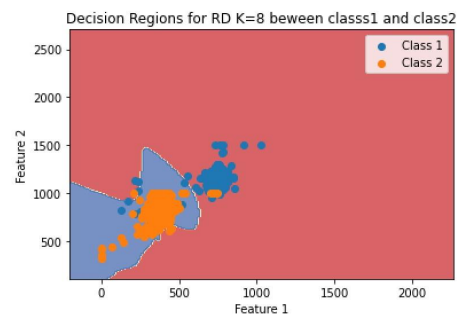
Confusion Matrix for K=8

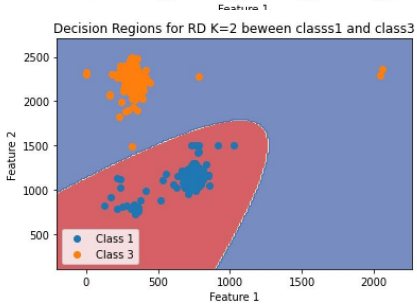
	Class1	Class2	Class2
Class1	623	10	6
Class2	25	736	3
Class3	1	0	707

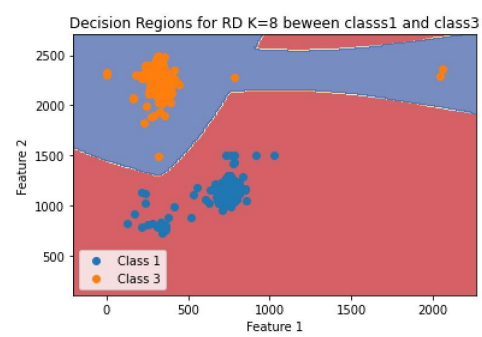


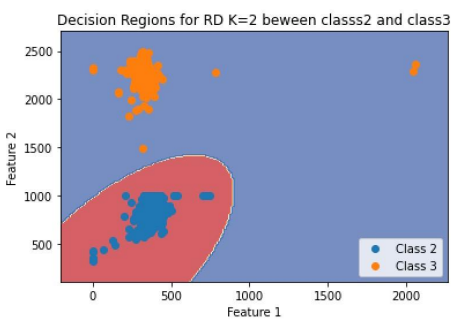


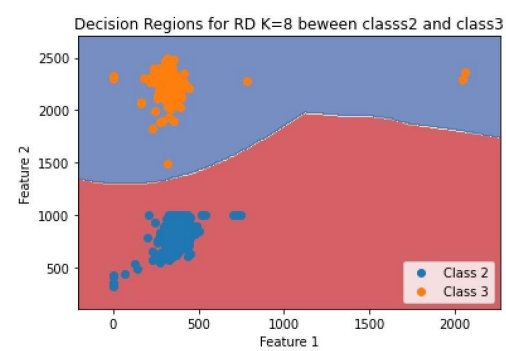


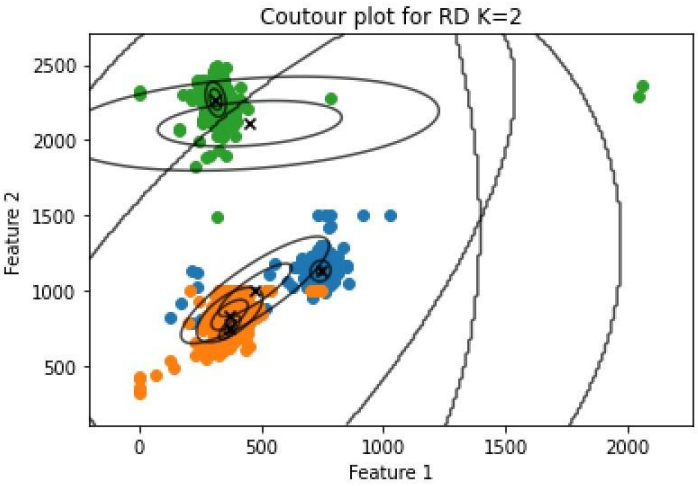


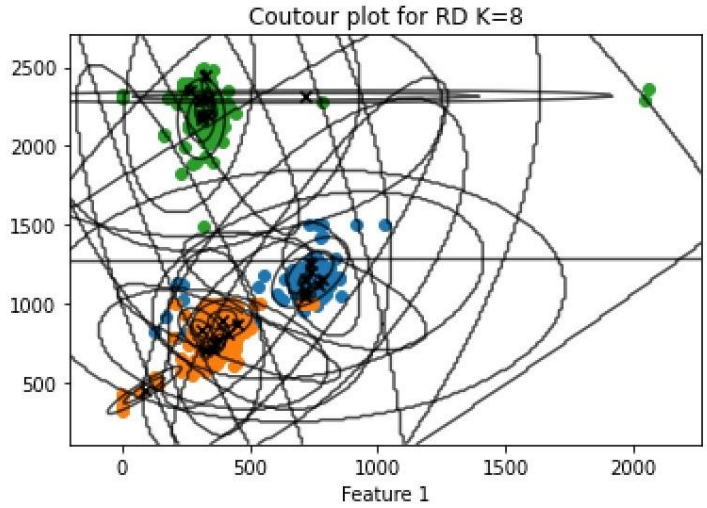






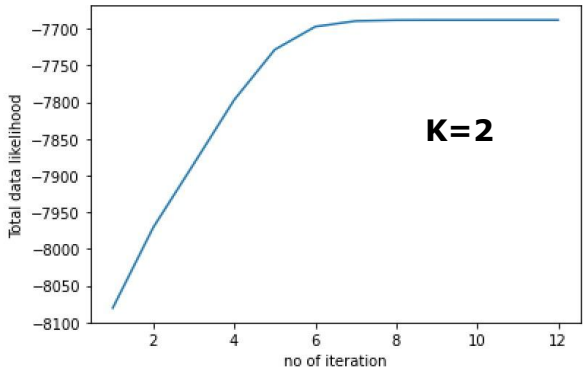


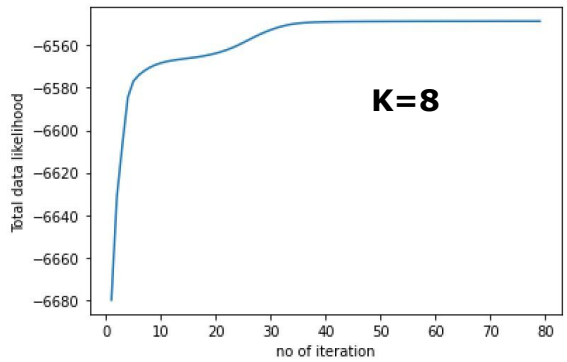

Coutour plot for RD K=2
Feature 2
Feature 1
2500
2000
1500
1000
500
0
500
1000
1500
2000


Coutour plot for RD K=8
Feature 1
2500
2000
1500
1000
500
0
500
1000
1500
2000

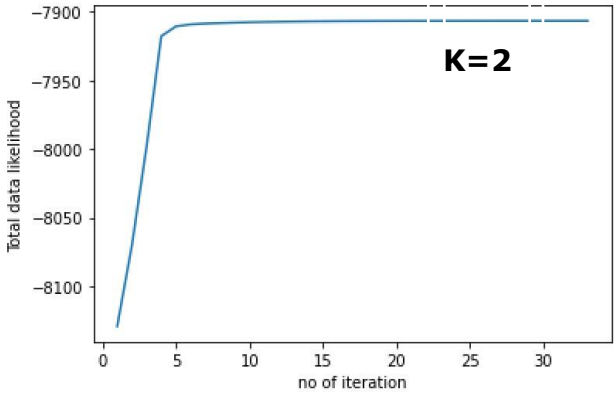
Iterations V/S Likelihood Plots

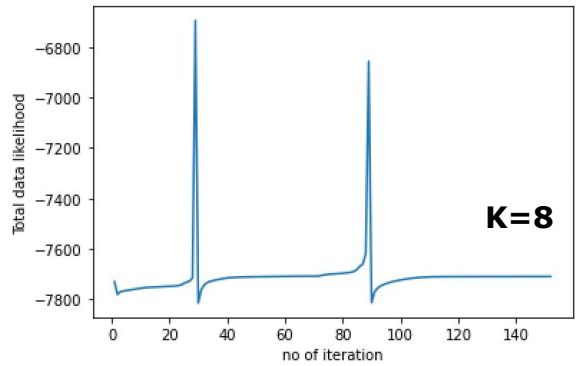
Class-1



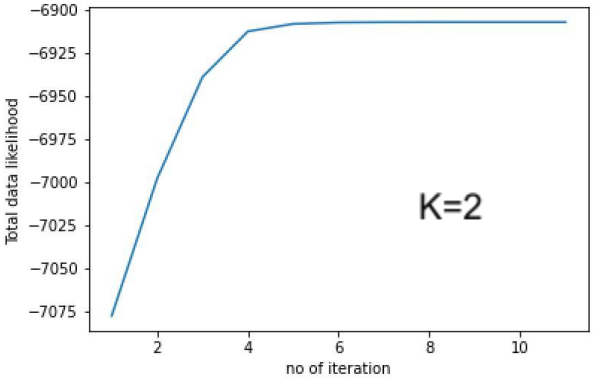


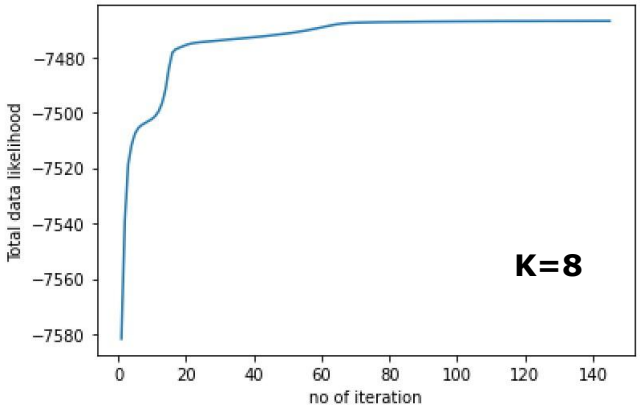
Class-2





Class-3





- Likelihood plot is highly fluctuating for class-2 k=8 this is because data for class 2 exhibit high variability

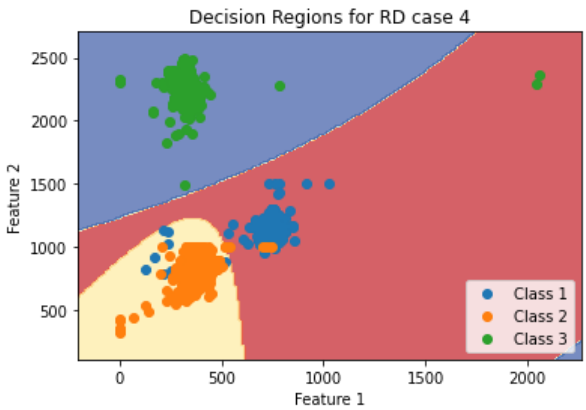
For k=2 we have 97.58% accuracy and for k=8 we have 97.86% accuracy. However, the number of iterations required for convergence of the E-M algorithm in K=8 is more than that for k=2.
So there is **Trade off between accuracy and speed between the** two models i.e k=2 and k=8.
Thus,

- ➢ If slightly lower accuracy is acceptable then 2-GMM is best.
- ➢ If computational resources and time constraints are limiting factors then also 2-GMM is best.

In summary, we can say if accuracy is of paramount importance then 8-GMM gives 97.86% accuracy and if speed is a critical factor and slightly reduced accuracy is acceptable then prefer 2-GMM.

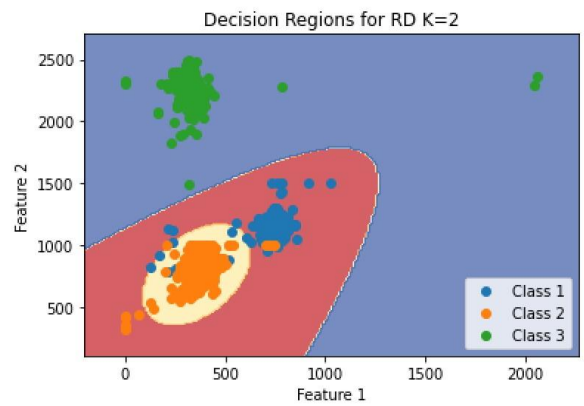
Comparison with Assignment-1

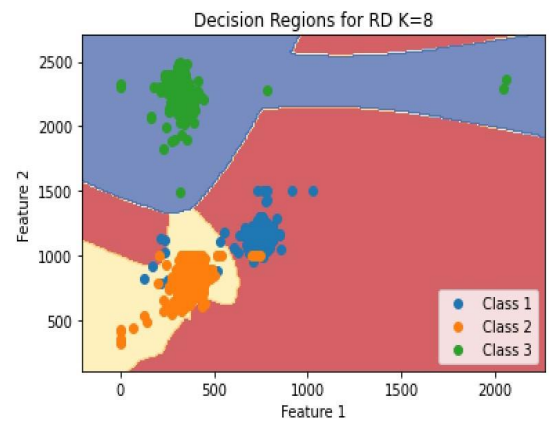
Plot comparison



From Assignment-1

From Assignment-1





Confusion Matrix comparison

	Class1	Class2	Class3
Class1	618	11	12
Class2	30	735	3
Class3	1	0	701

	Class1	Class2	Class3
Class1	150	0	0
Class2	0	150	0
Class3	0	0	150

	Class1	Class2	Class3
Class1	121	14	18
Class2	14	136	0
Class3	15	0	132

From confusion matrix and Plots it is clear GMM based Bayes classifier work better than Unimodal Bayes Classifier.

Dataset-2(b)

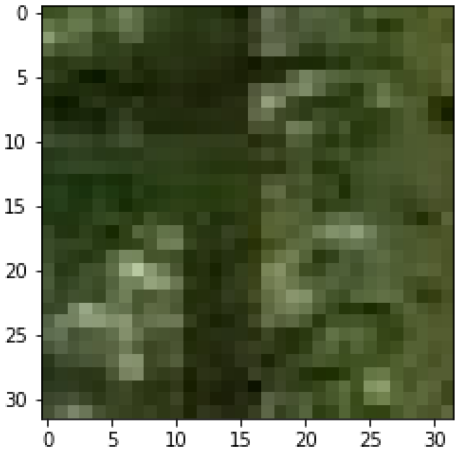
Dataset 2(b) contains Images of 3 classes to do classification on it. We are given two techniques to apply on this dataset to classify the images.

Technique 1

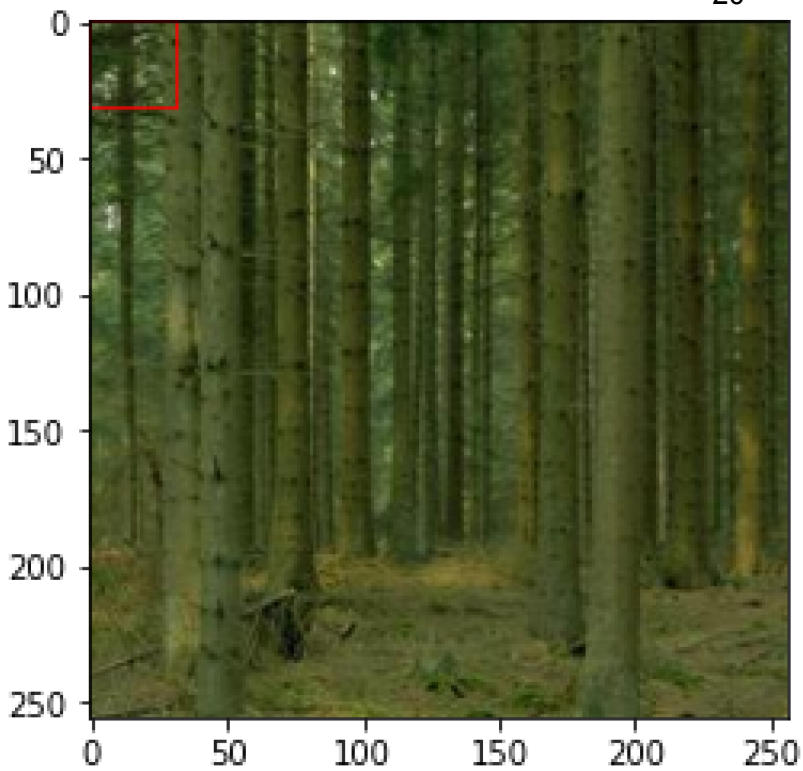
The First technique is converting an image to set of 24 dimensional feature vectors representing a 32*32 non overlapping patches in which 24 dimensions represent the 8-bin color histogram from Red Green and Blue channel for that patch.

The following pages include the example of how we have calculated the corresponding feature vectors of size 24 from each patch of each Image of the three data sets we are given.

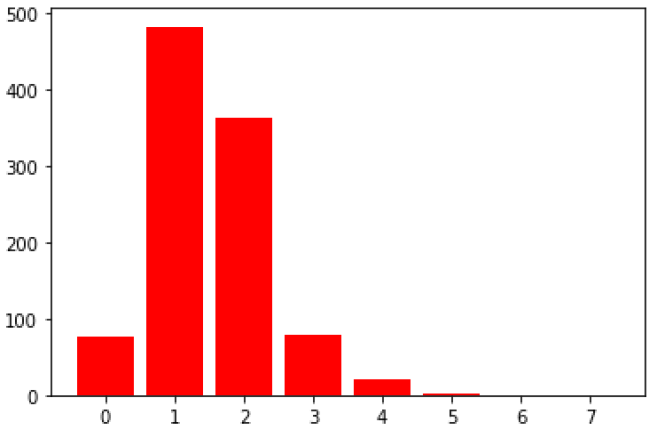
After converting the images to vectors we can do training on these vectors of the Bayes classifier with the gaussian mixtures to get the parameters of our model.

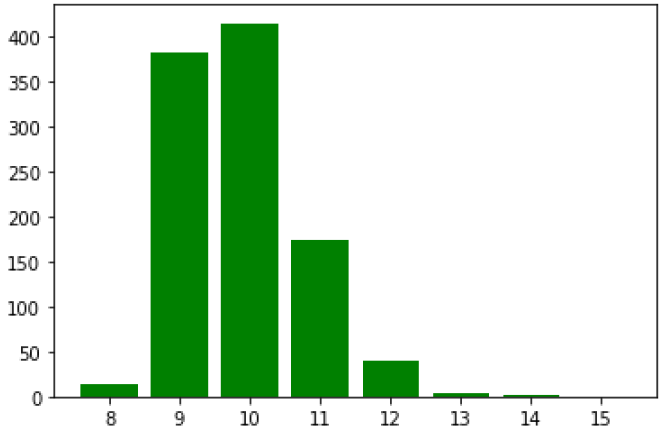


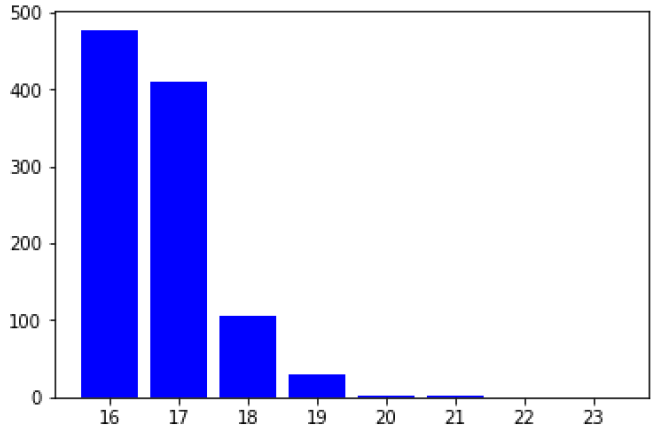
A patch of the Image



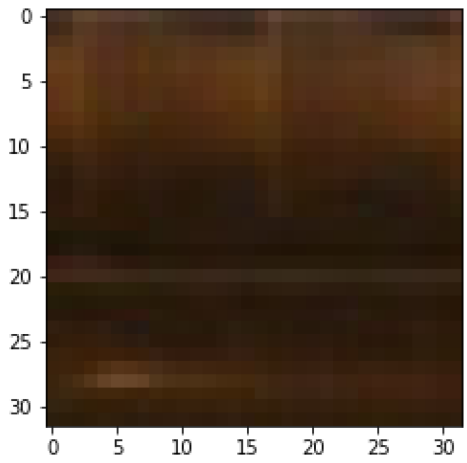
An Image of Forest





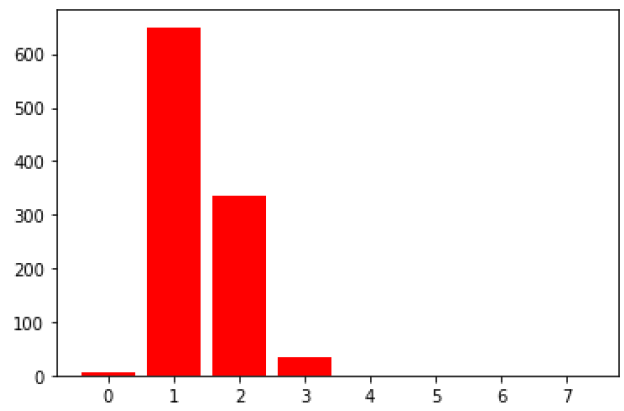


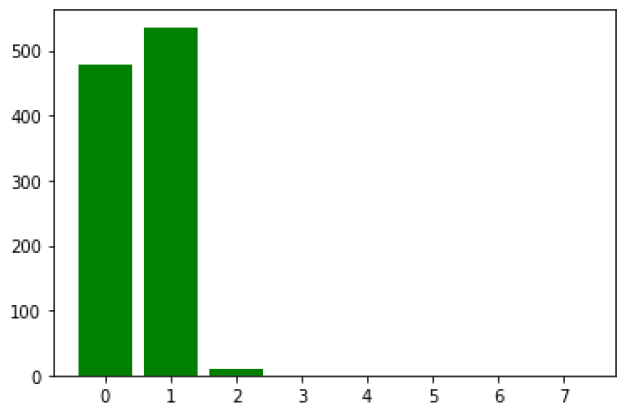
8-bin colour histogram for every colour channel (R, G and B) for a patch of an image of class 1

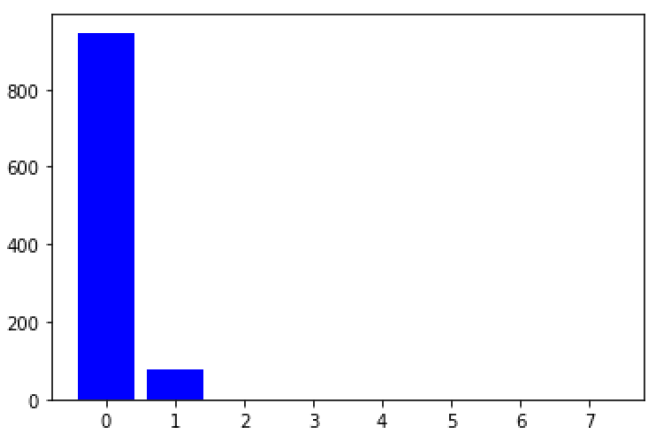


A patch of the Image

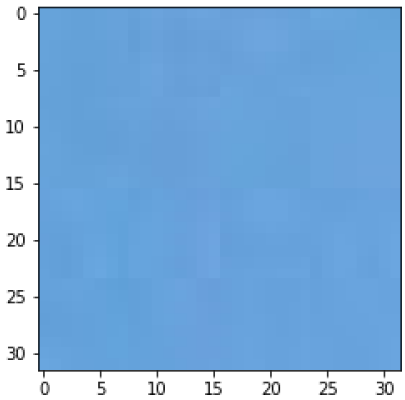
An Image of Ice Cream Parlor



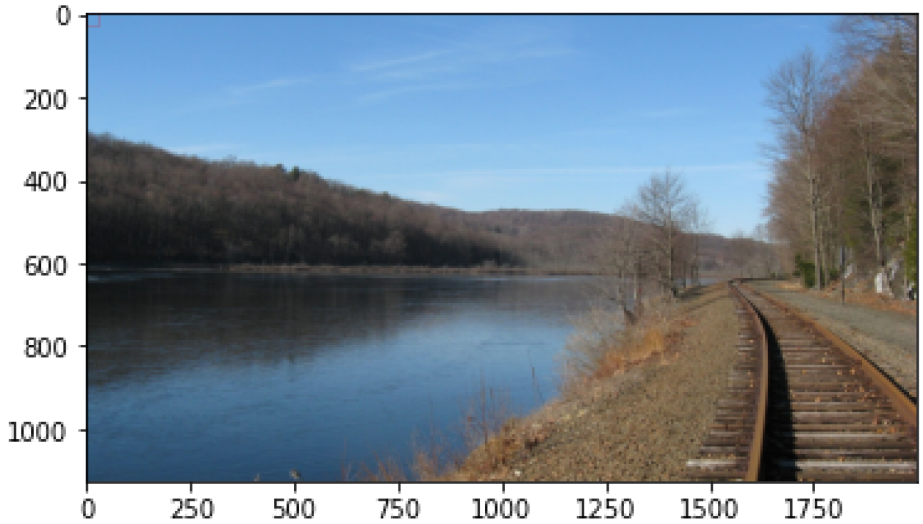




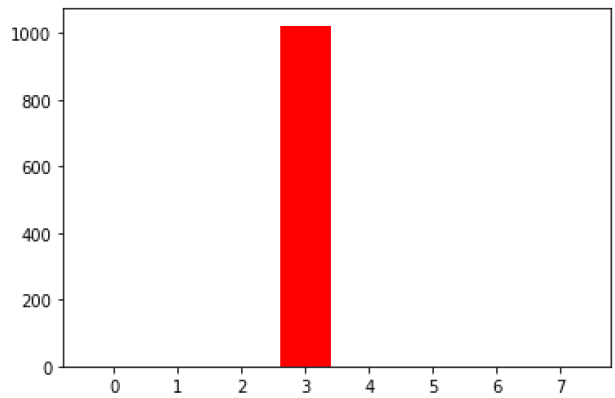
8-bin colour histogram for every colour channel (R, G and B) for a patch of an image of class 2

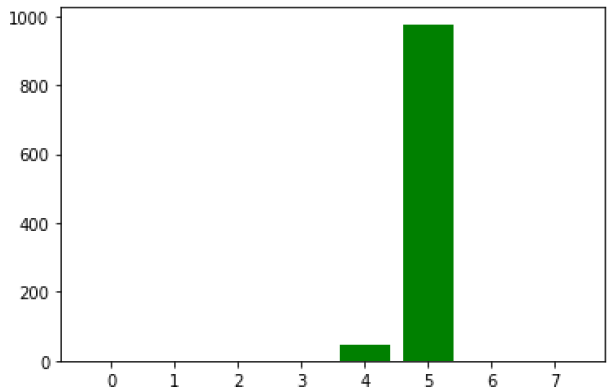


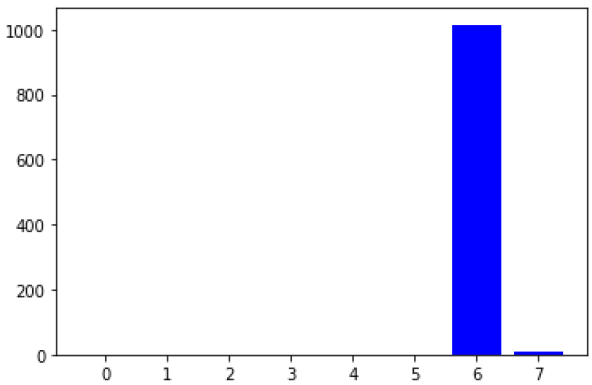
A patch of the Image



An Image of Pond







8-bin colour histogram for every colour channel (R, G and B) for a patch of an image of class 3

The following is the best results we get from applying the classification parameters estimation and doing testing on it with help of the test data set.

The Confusion Matrix and other measures

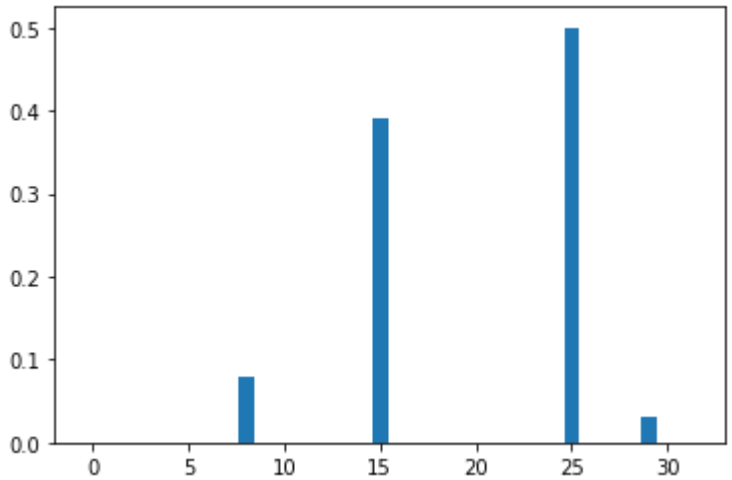
	Class1	Class2	Class3
Class1	31	3	4
Class2	6	36	3
Class3	13	11	43

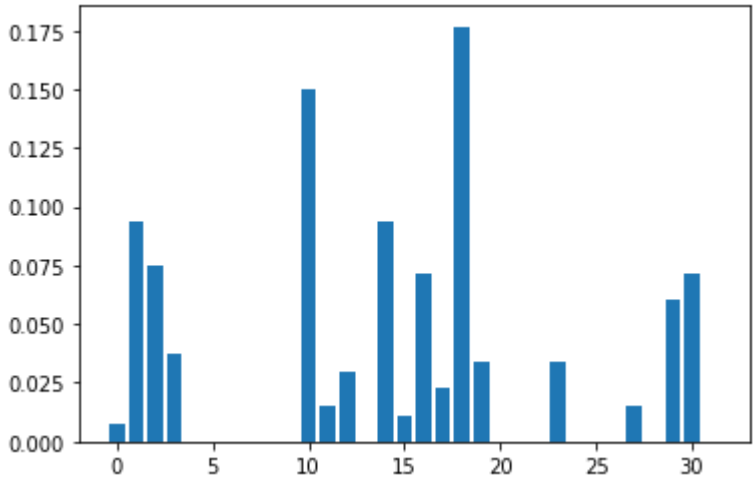
Accuracy : 73.33%

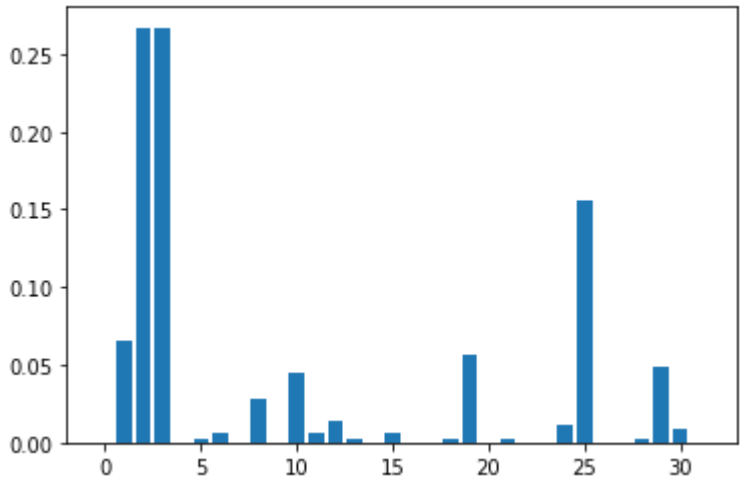
	Class-1	Class-2	Class-3
Precision	0.79084	0.90667	0.89795
Recall	0.80667	0.90667	0.88
F Measure	0.79867	0.90667	0.88889

Technique 2

Each image is represented by a 32 dimensional vector, so we have here 150 vectors of 3 classes with 50 vectros in each class.







	Class1	Class2	Class3
Class1	31	3	4
Class2	6	36	3
Class3	13	11	43

Confusion matrix for K=1

	Class1	Class2	Class3
Class1	31	6	23
Class2	12	42	12
Class3	7	2	15

Confusion matrix for K=2

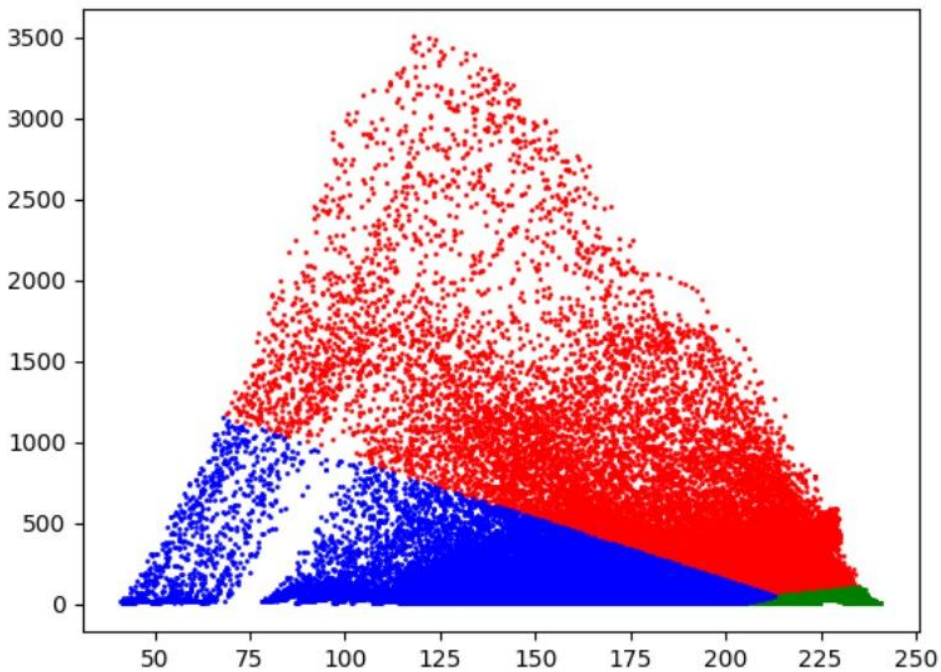
	Class1	Class2	Class3
Class1	37	17	32
Class2	6	29	12
Class3	7	4	6

Confusion matrix for K=3

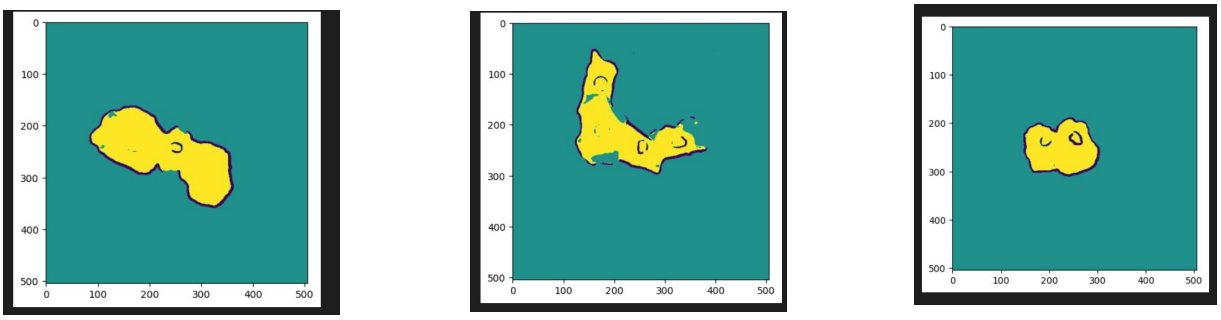
We observe that the results are not appealing. This is due to after converting the images to a single vector we are left with only 50 vectors of each class to apply the GMM based Bayes classifier which causes the overfitting and its effect increases with higher values of K.

Dataset-2(c)

In this dataset we have to apply clustering on the cell images. This is the result from the clustering using simple K-means clustering:



Clusters of the 2d vectors each vector representing a patch



Segmentation of Test Images by calculating the minimum distant cluster with help of Euclidean distance.