Pattern Recognition
CS669

ASSIGNMENT 4

Dimension Reduction (FDA) And Classication
(Perceptron, SVM, Baye's Classier using GMM)

Suryavanshi Virendrasingh B16037

Contents

1. Problem Description

Data-sets:

- **Dataset 1:** 2-dimensional articial data
  - Linearly separable data set
  - Non-linearly separable data set
- **Dataset 2:** 3 class scene image dataset: Consider the 32-dimensional BoVW representation from Assignment-2

Classifiers:

- Apply Fisher linear discriminant analysis (FDA) on Dataset-1 and Dataset-2. Use Bayes classier using both unimodal Gaussian and GMM.
- Perceptron-based classier on Dataset-1(a).
- SVM-based classier using (a) linear kernel, (b) polynomial kernel and (c) Gaussian/RBF kernel on Dataset-1 and Dataset-2.

2. Theory

1 Problem Definition

The curse of Dimensionality: For the datasets with feature vectors having more number of dimensions, we need to calculate more number of parameters. For instance in a unimodal Gaussian PDF, the number of parameters to be calculated are as follows:
For full covariance matrix,

$$NumberofParameters \;=\; d + d(d+1)/2 \tag{2..1}$$

where the rst term d indicates the means and the second term d(d+1)/2 indicates the covariance terms.
For diagonal covariance matrix,

$$NumberofParameters \;=\; d + d \tag{2..2}$$

Similarly for multimodal Gaussian Mixture Model with K mixtures,
For full covariance matrix,

$$NumberofParameters \;=\; Kd + Kd(d+1)/2 + K \tag{2..3}$$

where, rst term is means for all the mixtures, second term is means for all the mixtures and the third term are priors.
For diagonal covariance matrix,

$$NumberofParameters \;=\; Kd + Kd + K \tag{2..4}$$

Hence larger d requires calculation of larger number of parameters which requires more training data. Since training data is not always large, we use dimensionality reduction techniques like PCA and FDA.

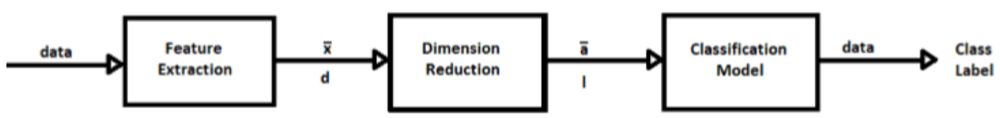

Figure 2..1. Steps in dimensionality reduction methods of classication

1.1 Fisher Discriminant Analysis

FDA is a dimension reduction tool in which a single direction of projection in which the separability of projected data is maximum is found. It takes into account the separability criteria which was missing in PCA. Though the mathematical expressions given below are valid for only two class classier the FDA itself can be extended to multiple classes using techniques like Max Voting Scheme etc.

Max Voting Scheme: One Multiclass classication problem is converted into multiple two class classication problem. The class in which a given test point gets classied into most number of times is taken as its predicted class.

D→ Data of 2 classes

$$D = \{\mathbf{x_n}, y_n\}_{n=1}^{N}$$

where, $y_n = +1, -1$ is the class label

(here we are considering only 2 classes. But we can extend the concept for multiple classes)

$$\begin{aligned}
D_+ &= \text{data of +ve class(class1)} \\
D_- &= \text{data of -ve class(class2)} \\
N_+ &= \text{number of examples in +ve class} \\
N_+ &= \text{number of examples in -ve class} \\
N &= N_+ + N_- \\
a_n &= \mathbf{w^T x_n}
\end{aligned}$$

Now, here we need to pick the direction such that the seperability is maximum. So we will have to quantify seperability.

Let,
m_+ be the mean of projected data of +ve class.
m_- be the mean of projected data of -ve class.
σ_+^2 be the variance of projected data of +ve class.
σ_-^2 be the variance of projected data of -ve class.

Scatter = Total deviation of x_n from mean, where μ is the mean.

For Multivariate Data,
Scater matrix

$$S = \sum_{n=1}^{N} (x-\mu)(x-\mu)^T$$

s_+^2 is the scatter for projected data of +ve class.
s_-^2 is the scatter for projected data of -ve class.

$$\begin{aligned}
s_+^2 &= \sum_{n=1}^{N_+} (a_n - m_+)^2 \\
s_-^2 &= \sum_{n=1}^{N_-} (a_n - m_-)^2
\end{aligned}$$

Separability
Separation between means: $(m_+ - m_-)^2 \rightarrow$ as large as possible

$$J(\mathbf{w}) = \frac{(m_+ - m_-)^2}{(s_+^2 + s_-^2)}$$

we need to write J in terms of $\mathbf{w}$

$$\begin{aligned}
J(\mathbf{w}) &= \frac{\mathbf{w}^T(\mu_+ - \mu_-)(\mu_+ - \mu_-)^T\mathbf{w}}{\mathbf{w}^T(s_+ + s_-)\mathbf{w}} \\
&= \frac{\mathbf{w}^T S_B \mathbf{W}}{\mathbf{w}^T S_W \mathbf{w}}
\end{aligned}$$

where, S_B is deviation of mean one class w.r.t. other class(Between class scatter matrix).
And S_W is within class scatter matrix.

We need to Maximize J

$$\frac{\partial J(\mathbf{w})}{\partial \mathbf{w}} = \bar{0}$$
$$S_B\mathbf{w} = \lambda S_w \mathbf{w}$$
$$S_W^{-1} S_B \mathbf{w} = \lambda \mathbf{w}, \quad \lambda \geq 0$$

Hence to find optimum value of **w** we need to do Eigen Analysis of $S_W^{-1} S_B$ and select the Eigen vector corresponing to maximum Eigen value.

1.2 Perceptron Learning Algorithm

Non Discriminant Learning: We estimate the parameter for a distribution that we assume for the data. While estimating the parameters we consider the data of one class alone. We don't give any importance to the discriminant features between classes while training the classier. Thus we call it, Non Discriminative Learning. Example, Bayes classier.

Discriminant Learning: Learning can be done by considering the data of classes together and learn from the discriminant features between the classes to directly come up with g(x). This is known as Discriminative learning.

Perceptron Learning: Perceptron learning does not involve any assumption about the shape of the distribution. In Perceptron learning, the only assumption is that the classes should be linearly separable. The main task is to nd a linear discriminant function from the training data of 2 class. We assume some arbitrary initial linear boundary and then in subsequent iterations try to minimize the error caused due to misclassications.

Perceptron learning consists of following steps:

- Initilaize values of w and w_o
- Compute error
- Try to move in the direction that minimizes error

The learning rate represents the jump in the direction of error reduction i.e. it controls the speed of gradient decent. More is the learning rate the more will be the jump. Initially learning rate should be large so that the error function is minimized quickly. But if learning rate is kept large, the decision boundary might keep oscillating between the two classes, not falling in the separating region, thus never making the error term zero. So learning rate should be decreased gradually to obtain a separating hyperplane.

Limitations:

- Efficient only when data is linearly separable.
- Algorithm suffers from local minima problem.
- The perceptron learning just focuses on nding the separating hyperplane, it does not guarantee that the separating hyperplane obtained is optimal. An optimal separating hyperplane is the hyperplane which maximises the distance of the nearest data points of either class from the hyperplane (i.e. the margin is maximum). To obtain a optimal Hyperplane we need the next model Support Vector Machine.

1.3 Support Vector Machine

Support Vector Machine is an improvement over Perceptron Learning as it tries to nd the optimal Hyperplane by maximising margin. SVM can also be applied to a dataset which are slightly linearly nonseparable.

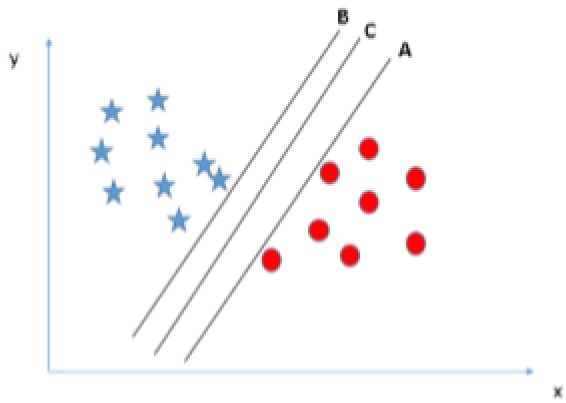

Figure 2..2. Existence of multiple hyperplane

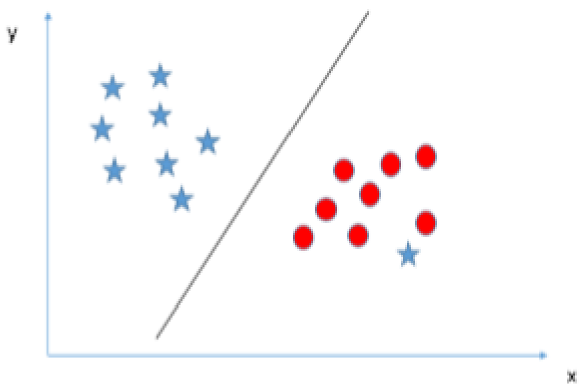

Figure 2..3. Optimum hyperplane

As shown in gure 2.2, Perceptron can give any one of the many existing hyperplanes, whereas, hyperplane gives us the optimum hyperplane by maximising the margin as shown in figure 2.3.

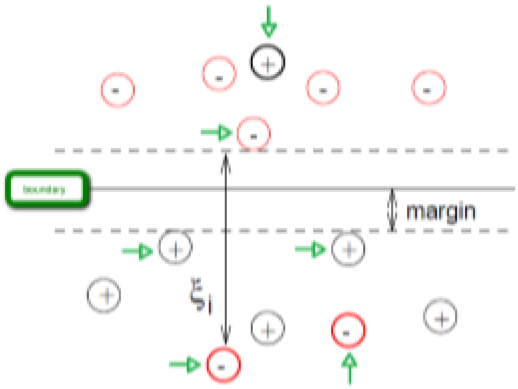

Figure 2..4. Margins and Support vectors

Support vectors: Support vectors are those data points which lie on the margin an those which violate the hyperplane. As indicated in gure 4 by the green arrows.

Margin: It is the distance of nearest points of either class from the Hyperplane. As illustrated in gure 2.4.

By using kernel trick SVM can be extended to non linearly separable data.Kernel trick involves taking low dimensional input space and transforming it to a higher dimensional space in which they are linearly separable. This is known as Cover's Theorem. The kernel functions used for transformation must satisfy Mercer Theorem. Commonly used kernel functions are Linear, Polynomial and Gaussian kernel functions.

Advantages:
1. The number of support vectors do not depend on the dimension of the feature vectors of the data set, it only depends on the number of training data points. The Gaussian Mixture model etc. all suer from curse of dimensionality as the number of parameters to be estimated depend on the dimensionality of data . SVM however does not depend on the dimensions of data hence it does not suer from curse of dimensionality. Therefore transforming the data to higher dimensions poses no problem.
2. Since it uses a subset of training points in the decision function (called support vectors), so it is also memory ecient.

1.4 Library and Classification Method

The Perceptron has been implemented from scratch and I have used ”one vs one” to classify among the present 3 classes. One Vs One approach helps handle the problem of class imbalance hence I am using this approach. For SVM I am using Python’s Sklearn library.

3. Observations and Results

1 Bayes Classifier using FDA

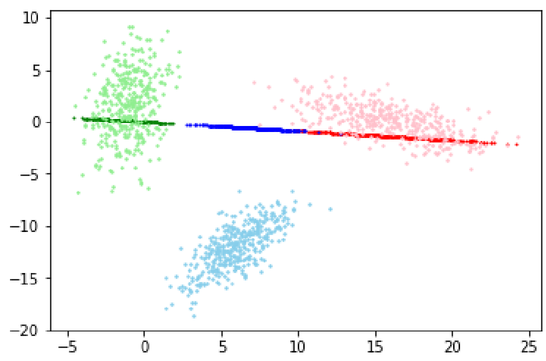

Figure 3..1. Original and Projected Data for Linearly Separable Dataset

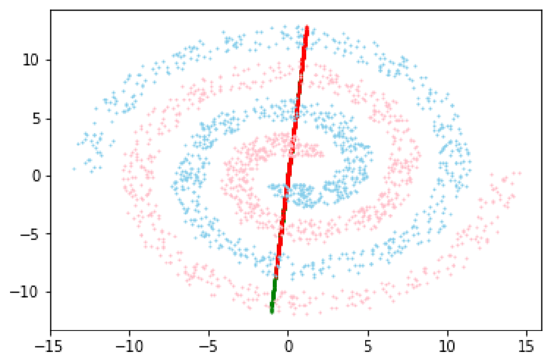

Figure 3..2. Original and Projected Data for Non-Linearly Separable Dataset

1.1 Results for Dataset 1

K-value : 1
Average Accuracy : 97.33%
Mean Precision: 97.37%
Mean Recall: 97.33%
Mean F-Measure: 97.34%

	Class1	Class2	Class 3
Class1	119	6	0
Class2	3	122	0
Class3	0	1	124

(a) Confusion Matrix

	Class1	Class2	Class3
Accuracy	97.60	97.33	99.73
Precision	97.54	94.57	100
Recall	95.20	97.60	99.20
F-Measure	96.35	96.06	99.60

(b) Analysis (in Percentage)

Table 3..1. Result for all the 3 classes

K-value : 2
Average Accuracy : 97.33%
Mean Precision: 97.37%
Mean Recall: 97.33%
Mean F-Measure: 97.34%

	Class1	Class2	Class 3
Class1	119	6	0
Class2	3	122	0
Class3	0	1	124

(a) Confusion Matrix

	Class1	Class2	Class3
Accuracy	97.60	97.33	99.73
Precision	97.54	94.57	100
Recall	95.20	97.60	99.20
F-Measure	96.35	96.06	99.60

(b) Analysis (in Percentage)

Table 3..2. Result for all the 3 classes

K-value : 4
Average Accuracy : 97.33%
Mean Precision: 97.37%
Mean Recall: 97.33%
Mean F-Measure: 97.34%

	Class1	Class2	Class 3
Class1	119	6	0
Class2	3	122	0
Class3	0	1	124

(a) Confusion Matrix

	Class1	Class2	Class3
Accuracy	97.60	97.33	99.73
Precision	97.54	94.57	100
Recall	95.20	97.60	99.20
F-Measure	96.35	96.06	99.60

(b) Analysis (in Percentage)

Table 3..3. Result for all the 3 classes

K-value : 8
Average Accuracy : 97.33%
Mean Precision: 97.37%
Mean Recall: 97.33%
Mean F-Measure: 97.34%

	Class1	Class2	Class 3
Class1	119	6	0
Class2	3	122	0
Class3	0	1	124

(a) Confusion Matrix

	Class1	Class2	Class3
Accuracy	97.60	97.33	99.73
Precision	97.54	94.57	100
Recall	95.20	97.60	99.20
F-Measure	96.35	96.06	99.60

(b) Analysis (in Percentage)

Table 3..4. Result for all the 3 classes

1.2 Result for Dataset 2

K-value : 1
Average Accuracy : 55.21%
Mean Precision: 55.23%
Mean Recall: 55.21%
Mean F-Measure: 55.17%

	Class1	Class2
Class1	190	136
Class2	156	170

(a) Confusion Matrix

	Class1	Class2
Accuracy	55.21	55.21
Precision	54.91	55.55
Recall	58.28	52.14
F-Measure	56.54	53.79

(b) Analysis (in Percentage)

Table 3..5. Result for all the 3 classes

K-value : 2
Average Accuracy : 59.81%
Mean Precision: 59.25%
Mean Recall: 59.82%
Mean F-Measure: 59.66%

	Class1	Class2
Class1	215	111
Class2	151	175

(a) Confusion Matrix

	Class1	Class2
Accuracy	59.81	59.81
Precision	58.74	61.11
Recall	65.95	53.68
F-Measure	62.13	57.18

(b) Analysis (in Percentage)

Table 3..6. Result for all the 3 classes

K-value : 4
Average Accuracy : 68.56%
Mean Precision: 68.88%
Mean Recall: 68.86%
Mean F-Measure: 68.85%

	Class1	Class2
Class1	229	97
Class2	106	220

(a) Confusion Matrix

	Class1	Class2
Accuracy	68.86	68.86
Precision	68.35	69.40
Recall	70.24	67.48
F-Measure	69.28	68..42

(b) Analysis (in Percentage)

Table 3..7. Result for all the 3 classes

K-value : 8
Average Accuracy : 67.33%
Mean Precision: 67.51%
Mean Recall: 67.33%
Mean F-Measure: 67.24%

	Class1	Class2
Class1	236	90
Class2	123	203

(a) Confusion Matrix

	Class1	Class2
Accuracy	67.33	67.33
Precision	65.73	69.28
Recall	72.39	62.26
F-Measure	68.90	65.58

(b) Analysis (in Percentage)

Table 3..8. Result for all the 3 classes

1.3 Result for Image Scene Dataset

K-value : 1
Average Accuracy : 44.00%
Mean Precision: 45.60%
Mean Recall: 44.00%
Mean F-Measure: 41.33%

	Class1	Class2	Class 3
Class1	36	4	10
Class2	25	10	15
Class3	24	6	20

(a) Confusion Matrix

	Class1	Class2	Class3
Accuracy	58.00	66.67	63.33
Precision	42.35	50.00	44.44
Recall	72.00	20.00	40.00
F-Measure	53.33	28.57	42.10

(b) Analysis (in Percentage)

Table 3..9. Result for all the 3 classes

K-value : 2
Average Accuracy : 43.33%
Mean Precision: 45.27%
Mean Recall: 43.33%
Mean F-Measure: 43.09%

	Class1	Class2	Class 3
Class1	17	25	8
Class2	11	28	11
Class3	8	22	20

(a) Confusion Matrix

	Class1	Class2	Class3
Accuracy	65.33	54.00	67.33
Precision	47.22	37.33	51.28
Recall	34.00	56.00	40.00
F-Measure	39.53	44.80	44.94

(b) Analysis (in Percentage)

Table 3..10. Result for all the 3 classes

K-value : 4
Average Accuracy : 42.00%
Mean Precision: 42.15%
Mean Recall: 41.88%
Mean F-Measure: 42.01%

	Class1	Class2	Class 3
Class1	21	13	16
Class2	17	20	13
Class3	16	12	22

(a) Confusion Matrix

	Class1	Class2	Class3
Accuracy	58.67	63.33	62.00
Precision	38.89	44.44	43.13
Recall	42.00	40.00	44.00
F-Measure	40.38	42.10	43.56

(b) Analysis (in Percentage)

Table 3..11. Result for all the 3 classes

K-value : 8
Average Accuracy : 39.33%
Mean Precision: 39.33%
Mean Recall: 39.33%
Mean F-Measure: 39.33%

	Class1	Class2	Class 3
Class1	18	16	16
Class2	17	20	13
Class3	15	14	21

(a) Confusion Matrix

	Class1	Class2	Class3
Accuracy	57.33	60.00	61.33
Precision	36.00	40.00	42.00
Recall	36.00	40.00	42.00
F-Measure	36.00	40.00	42.00

(b) Analysis (in Percentage)

Table 3..12. Result for all the 3 classes

2 Perceptron Classifier

Perceptron has been applied on Dataset 1.

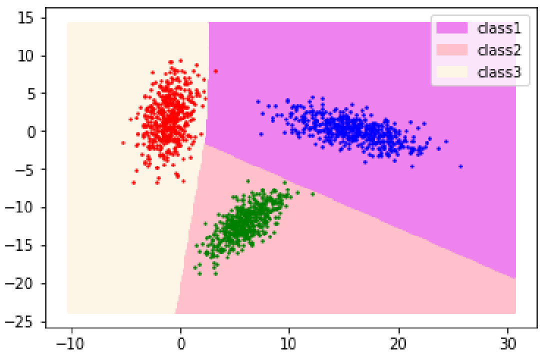

Figure 3..3. Perceptron applied on Dataset 1.

Average Accuracy : 99.73%
Mean Precision: 99.73%
Mean Recall: 99.73%
Mean F-Measure: 99.73%

	Class1	Class2	Class 3
Class1	125	0	0
Class2	0	125	0
Class3	1	0	124

(a) Confusion Matrix

	Class1	Class2	Class3
Accuracy	99.73	100	99.73
Precision	99.20	100	100
Recall	100	100	99.19
F-Measure	99.60	100	99.59

(b) Analysis (in Percentage)

Table 3..13. Result for all the 3 classes

3 SVM Classifier

3.1 Results for Dataset 1

RBF Kernel

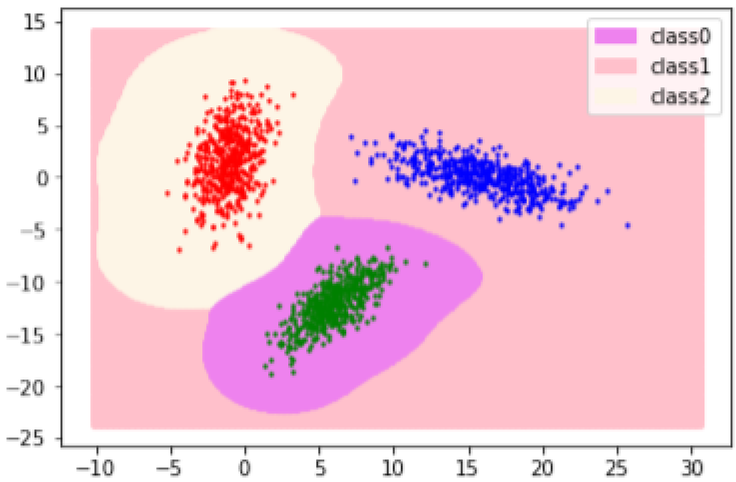

Figure 3..4. SVM applied on Dataset 1 using rbf kernel

Average Accuracy : 100%
Mean Precision: 100%
Mean Recall: 100%
Mean F-Measure: 100%

	Class1	Class2	Class 3
Class1	125	0	0
Class2	0	125	0
Class3	0	0	125

(a) Confusion Matrix

	Class1	Class2	Class3
Accuracy	100	100	100
Precision	100	100	100
Recall	100	100	100
F-Measure	100	100	100

(b) Analysis (in Percentage)

Table 3..14. Result for all the 3 classes

Linear Kernel

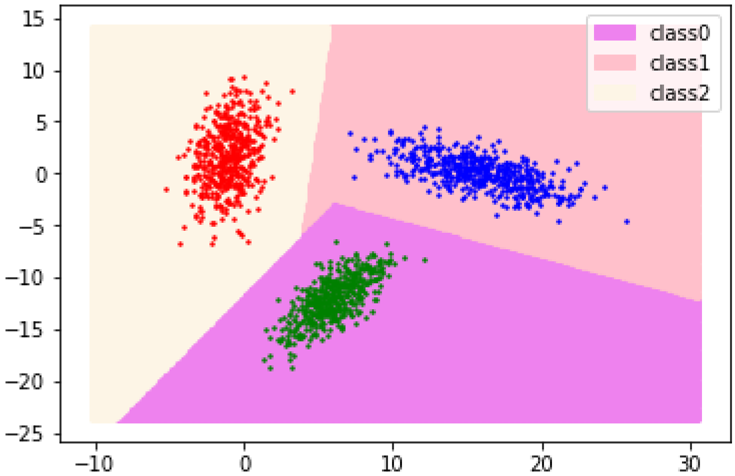

Figure 3..5. SVM applied on Dataset 1 using Linear kernel

Average Accuracy : 100%
Mean Precision: 100%
Mean Recall: 100%
Mean F-Measure: 100%

	Class1	Class2	Class 3
Class1	125	0	0
Class2	0	125	0
Class3	0	0	125

(a) Confusion Matrix

	Class1	Class2	Class3
Accuracy	100	100	100
Precision	100	100	100
Recall	100	100	100
F-Measure	100	100	100

(b) Analysis (in Percentage)

Table 3..15. Result for all the 3 classes

Polynomial Kernel

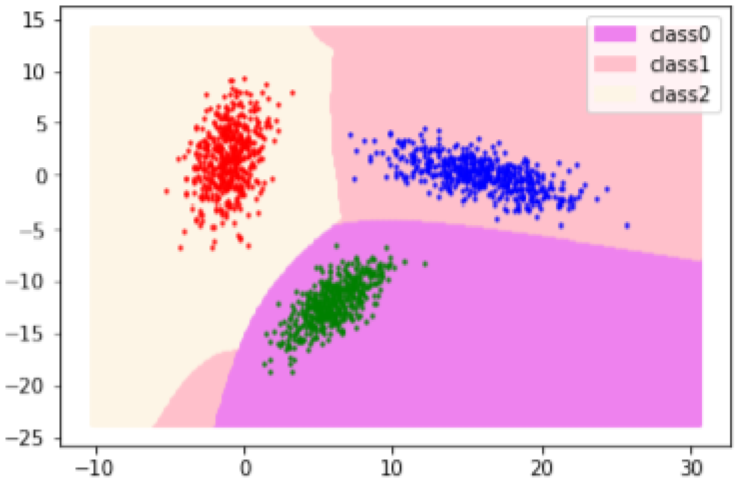

Figure 3..6. SVM applied on Dataset 1 using polynomial kernel of degree 2

Average Accuracy : 100%
Mean Precision: 100%
Mean Recall: 100%
Mean F-Measure: 100%

	Class1	Class2	Class 3
Class1	125	0	0
Class2	0	125	0
Class3	0	0	125

(a) Confusion Matrix

	Class1	Class2	Class3
Accuracy	100	100	100
Precision	100	100	100
Recall	100	100	100
F-Measure	100	100	100

(b) Analysis (in Percentage)

Table 3..16. Result for all the 3 classes

3.2 Results for Dataset 2

RBF Kernel

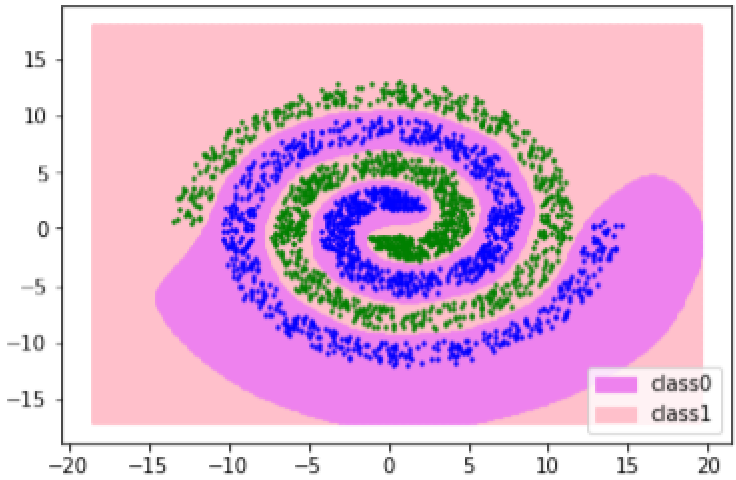

Figure 3..7. SVM applied on Dataset 2 using rbf kernel

Average Accuracy : 100%
Mean Precision: 100%
Mean Recall: 100%
Mean F-Measure: 100%

	Class1	Class2
Class1	326	0
Class2	0	326

(a) Confusion Matrix

	Class1	Class2
Accuracy	100	100
Precision	100	100
Recall	100	100
F-Measure	100	100

(b) Analysis (in Percentage)

Table 3..17. Result for all the 3 classes

Linear Kernel

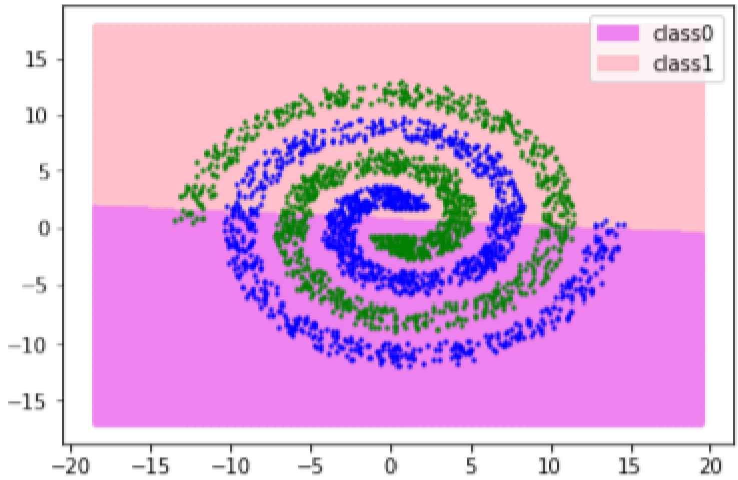

Figure 3..8. SVM applied on Dataset 2 using Linear kernel

Average Accuracy : 55.06%
Mean Precision: 55.07%
Mean Recall: 55.06%
Mean F-Measure: 55.03%

	Class1	Class2
Class1	188	138
Class2	155	171

(a) Confusion Matrix

	Class1	Class2
Accuracy	55.06	55.06
Precision	54.81	55.33
Recall	57.67	52.45
F-Measure	56.20	53.85

(b) Analysis (in Percentage)

Table 3..18. Result for all the 3 classes

Polynomial Kernel

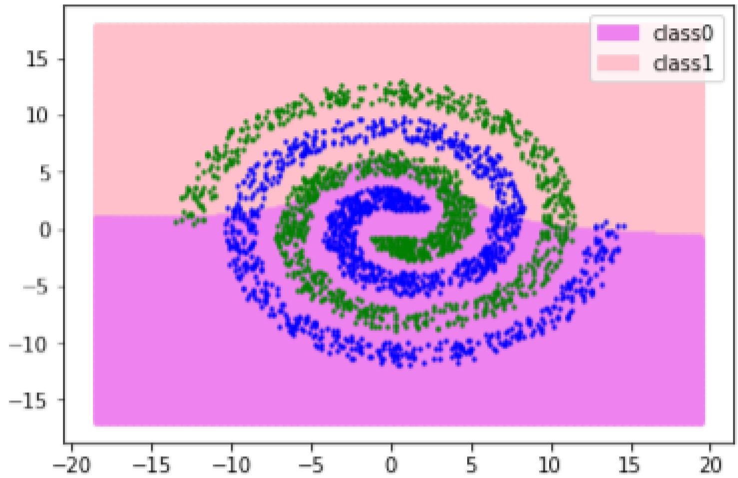

Figure 3..9. SVM applied on Dataset 2 using Polynomial kernel of degree 3

Average Accuracy : 55.21%
Mean Precision: 56.16%
Mean Recall: 55.21%
Mean F-Measure: 53.41%

	Class1	Class2
Class1	244	82
Class2	210	116

(a) Confusion Matrix

	Class1	Class2
Accuracy	55.21	55.21
Precision	53.74	58.59
Recall	74.85	35.58
F-Measure	65.26	44.27

(b) Analysis (in Percentage)

Table 3..19. Result for all the 3 classes

3.3 Results for 3 class scene Image Dataset

RBF Kernel

Average Accuracy : 34.00%
Mean Precision: 55.63%
Mean Recall: 34.00%
Mean F-Measure: 18.05%

	Class1	Class2	Class3
Class1	0	0	50
Class2	0	1	49
Class3	0	0	50

(a) Confusion Matrix

	Class1	Class2	Class3
Accuracy	66.67	67.33	34.00
Precision	33.33	100	33.56
Recall	0	2	100
F-Measure	0	3.92	50.25

(b) Analysis (in Percentage)

Table 3..20. Result for all the 3 classes

Linear Kernel

Average Accuracy : 47.33%
Mean Precision: 48.14%
Mean Recall: 47.33%
Mean F-Measure: 46.43%

	Class1	Class2	Class3
Class1	33	11	6
Class2	19	21	10
Class3	21	12	17

(a) Confusion Matrix

	Class1	Class2	Class3
Accuracy	62	65.33	67.33
Precision	45.20	45.72	51.51
Recall	66	42	34
F-Measure	53.66	44.68	40.96

(b) Analysis (in Percentage)

Table 3..21. Result for all the 3 classes

Polynomial Kernel

Average Accuracy : 49.33%
Mean Precision: 50.56%
Mean Recall: 49.33%
Mean F-Measure: 49.30%

	Class1	Class2	Class3
Class1	29	11	10
Class2	19	22	9
Class3	19	8	23

(a) Confusion Matrix

	Class1	Class2	Class3
Accuracy	60.67	68.67	69.33
Precision	43.28	53.65	54.76
Recall	58	44	46
F-Measure	49.57	48.35	50

(b) Analysis (in Percentage)

Table 3..22. Result for all the 3 classes

4. Different SVMs by Varying Parameters of Sklearn svm Library

1 Variation of C in Radial Basis kernal Function

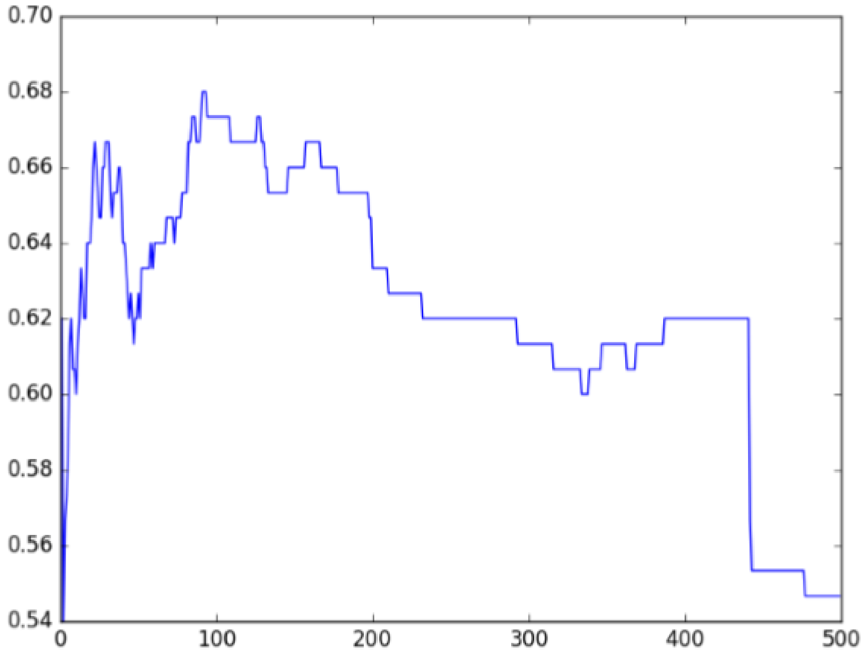

Figure 4..1. Accuracy Vs. C

2 Variation of σ in Radial Basis kernal Function

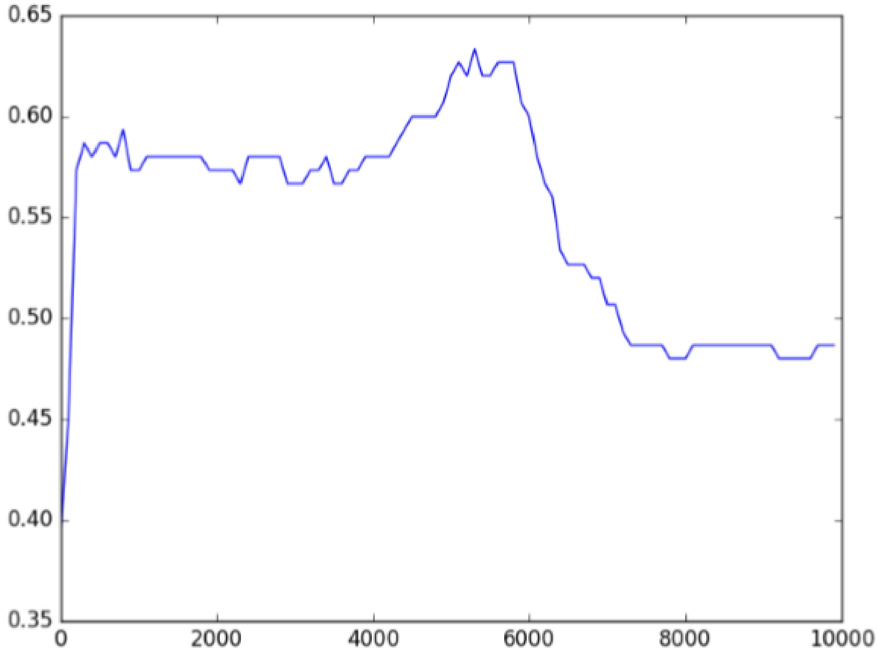

Figure 4..2. Accuracy Vs. Sigma

As we can see in both gure 4.1 and gure 4.2, for variation in C and σ in radial basis function, the accuracy first increases to obtain a maximum value and then decreases further. For low value of σ, the exponential term in radial basis function tends to 0. This leads to most samples being classied into one class, thus the poor accuracy. When we increase we rectify this problem as the term mow instead of tending to 0, is more spread in the range [0 1]. On further increasing the value of we see a decrease in accuracy again as all the term are now shifted towards 1. Hence classication into other class increases signicantly, reducing the accuracy again.

5. Inferences

1 Fisher Discriminant Analysis (LDA)

1. The accuracy increases as the number of cluster increases. This is due to better segregation of the data into different classes with increasingh K.

2. LDA is preferred when we have small samples and features are correlated with redundant data.

3. The reduction of dimension helps process large data in lesser time by reducing the number of dimensions but the ability to interpret the influence of individual features goes down.

4. **Comparison With PCA:** PCA performs better in case where number of samples per class is less whereas LDA works better with large dataset having multiple classes and class separability is an important factor factor while reducing dimensionality.

2 Perceptron Classifier

The data is linearly separable so it is ensured that perceptron learning can obtain the values for omega vector that completely separates the data but the number of iterations can be more or less depending on the data.

3 SVM Classifier

1. For Linearly separable data, 100% accuracy is observed for every kernel because the data can be easily separated by all the kernels.

2. For Non Linear Data and BoVW data the precision for various kernels is as per this order: RBF Kernel ¿ Polynomial Kernel ¿ Linear Kernel.

3. For polynomial Kernel best results were obtained for degree equal to 3.

4. For RBF Kernel, I observed that higher the value of gamma parameter better it will try to exact fit as per the training dataset i.e.generalization error and cause over-fitting problem. C(Penalty paramater or error term) controls the trade off between smooth decision boundary and classifying the training points correctly. Final values for obtaining best results were gamma = 1e-1 and C = 1.

5. In BoVW the feature dimension is large. SVM is extremely helpful when number of features is larger than number of samples and obtain better result than other classification techniques performed in the course earlier.

6. Conclusions

Comparisons and conclusions from results

1 Linearly Separable Dataset

The Linearly separable data set has always proved to be the easiest to classify and all classiers including bayes classier (using unimodal gaussian distribution), Bayes classier using GMM (after FDA), Bayes classier using GMM (after PCA), Perceptron, SVM etc. have given almost 100% accuracy. Note however that the decision boundary obtained in case of the bayes classier was better as compared to that of perceptron. This is because perceptron continued to look for the optimum decision boundary till the point all the points in the training data were correctly classied. Thus in most cases where perceptron was used the decision boundary ended up being very close to the training data points of one of the classes. Thus any point slightly more shifted towards another class would be misclassified leading to drop in accuracy. The SVM classier also provided a better decision. This happens due to the margin that we consider in this case.

2 Non Linear Separable Dataset

When we had tried to classify the non linearly separable data using the linear boundary obtained using bayes classier the results were poor. On considering the boundary to be of quadratic nature the results improved a little. Using GMM in bayes classier, signicantly increased the classication accuracy and the accuracy almost touched 100%. The accuracy given by SVM for the non linear dataset depends on the kind of kernel function used. The Radial basis kernel function led to 100% accuracy in this case, whereas the other kernel functions performed poorly. The Bayes classier using GMM (after FDA) provided the classification with an accuracy of around 70%.

3 Image Dataset

I used the Bayes classier using GMM and SVM classier for the classication of images. The dimensionality reduction was also tried on the data before trying to classify the data using the bayes classier. The two dimensionality reduction techniques tried are PCA an FDA. It was noticed that the results obtained when using the FDA were better than the ones obtained using PCA. We also noticed that the classication accuracy provided by the SVM classier were also better than the ones when we tried to use the Bayes classier using GMM. However Bayes classier using GMM on Image data set provided a better accuracy as compared to the K-nearest neighbour classier and DHMM classier. Among the various kernel functions used in the SVM, the radial basis kernel tended to give the best results.