CMSC828C: Statistical Pattern Recognition

Project1 Report

Author: Amoghavarsha Prasanna (UID:116952910)

Date:20 November 2020

1. Introduction

The premise of the project was to implement various popular classifiers like the Bayesian, K-nearest Neighbors(KNN), Support Vector Machines perform the required classification task for the data sets provided.

In addition, the project also involves the application of the Linear Discriminant Analysis(LDA) and Principal Component Analysis(PCA) on the data and to study the effect that these transformations have on the classification task with different classifiers.

2. Dataset

The data-set that has been predominantly used for this project is:

Data.mat: Cropped images of 200 subjects, 3 images each, each image of size 24x21. The file data.mat has a variable "face" of size (24x21x600). The images corresponding to the person labeled n, n = 1, . . . , 200 can be indexed in Matlab as face(:,:,3*n-2), face(:,:,3*n-1) and face(:,:,3*n). The first image is a neutral face, the second image is a face with facial expression, and the third image has illumination variations. The data was split into training and testing sets as required for each classifier and task.

3. Task: Neutral Vs Expression Classification

3.1 Bayesian Classifier:

A Bayesian classifier was implemented for the binary classification task of neutral vs expression. For the initial test of the classifier a 50-50 split of training and testing data was used. The classifier was also tested for different proportions of split of the data-set to study the effectiveness and accuracy of the data. Since the data was easily separable the Bayesian classifier performed relatively well.

The results observed are as follows:

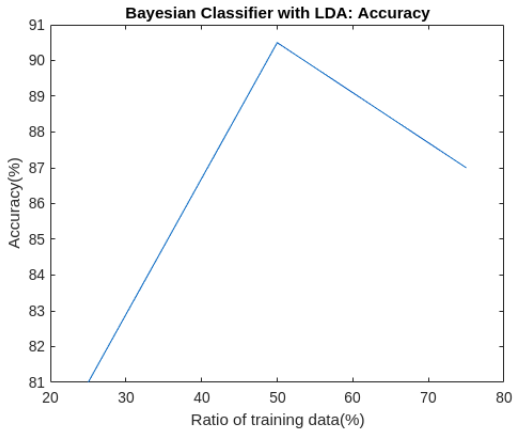
Train-Test Split (%):	Accuracy (%):
25-75	84.6
50-50	86.5
75-25	81

The trend that was observed was with the increase in the number of training images there was a slight increase in the accuracy as well. However at 75-25 split, the accuracy of the classifier decreased as compared to previous configuration of train-test split. So, one has to find the right trade-off when training the data to avoid over-fitting to training data.

3.1.1 LDA with Bayesian Classifier:

The Linear Discriminant analysis(LDA) transformation was applied to the data-set, which helped in reducing dimesionality of the data. In addition, a salient feature of LDA is help to reduce with-class scatter and increase between class scatter. This helped improve the accuracy of the Bayesian classifier. This also tested for 3 different ratio of data split. The trend that was observed for normal Bayesian holds true here as well.
The observed results are as follows:

Train-Test Split (%):	Accuracy (%):
25-75	81.25
50-50	90.5
75-25	87



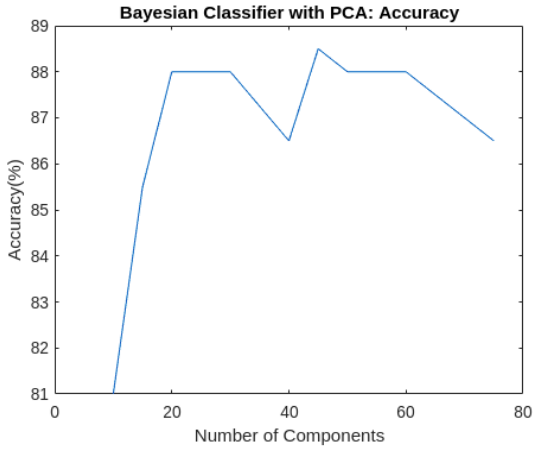
3.1.2 PCA with Bayesian Classifier

Principal Component Analysis is a also a method used for dimensionality reduction and tries to maintain the original data variance,PCA was applied to data to study effectiveness of the Bayesian classifier. Since PCA is dependent on the number of components corresponding to maximum eigen values, the accuracy of the Bayesian was calculated for different number of components and recorded.

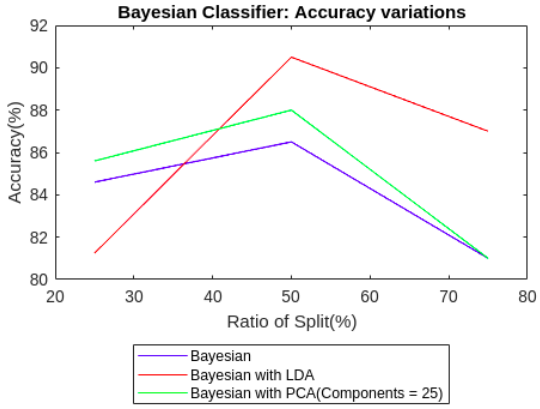
For the results provided below, the train-test split used was 50-50. In addition, by varying the train-test split it was observed that it follows the general trend. Initially, with the increase in the number of components, the accuracy also improves, then after a point with further increase in components, the accuracy actually starts to fall, hence its is important to find the right trade off for the data-set. For data.mat with 50-50 split, 25-40 would be the optimal number of components for this classifier.

The observed results:

Number of Components :	Accuracy (%):
10	81
15	85.6
20	88
25	88
30	88
40	88
45	86.5
50	88
60	88
75	86.5



The accuracy variations for all the the variations of the classifier with changing train-test split of data was recorded, the general trend holds true in this case as well.

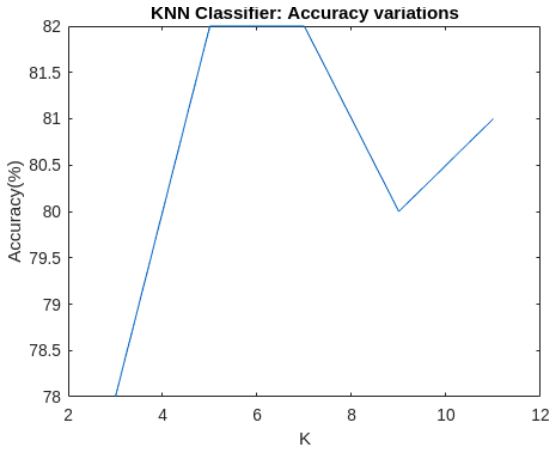


3.2 K-Nearest Neighbors:

The K-Nearest Neighbor classifier was employed for the neutral vs expression binary classification. The results were observed for different values of K.The classifier produced relatively well for the data-set.

The observed results:

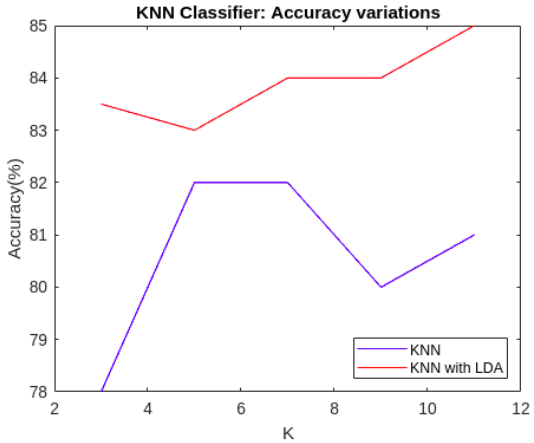
K value :	Accuracy (%):
3	78
5	82
7	82
9	80
11	81



3.2.1 LDA with KNN classifier:

LDA performed on the data and fed into the KNN classifier. As expected there was an increase in the effectiveness of the classifier. It can be observed in the results below.

K value :	Accuracy (%):
3	83.5
5	83
7	84
9	84
11	85

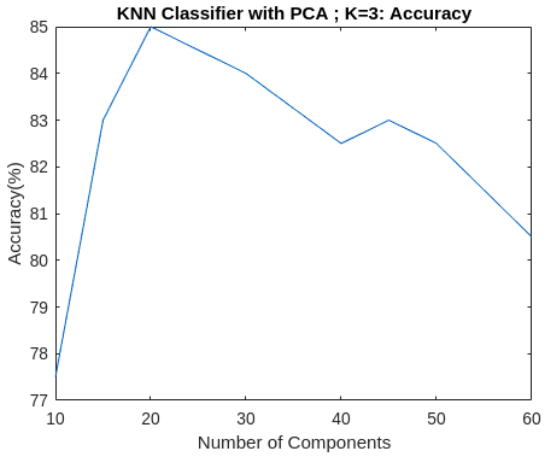


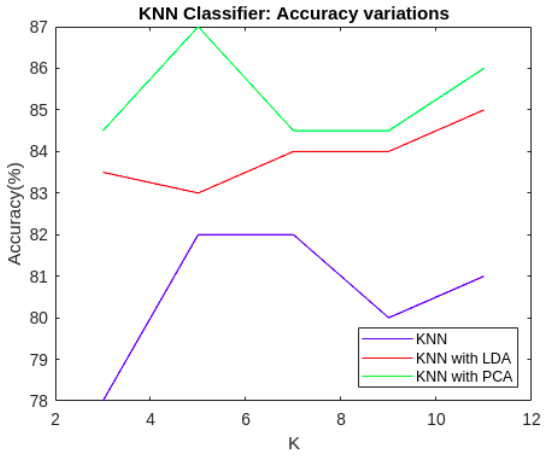
3.2.2 PCA with KNN classifier:

For the KNN with PCa there are multiple parameters that can be tuned to get the 'best' possible results for the classification task.
The number of components for PCA and the number of K value for the classifier were varied to study the effect on the classifier.

The observed results:

K value :	Accuracy (%):
3	84.5
5	87
7	84.5
9	84.5
11	86





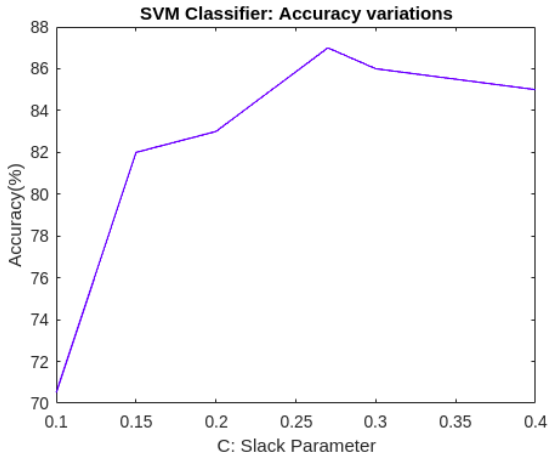
From, the observed results, it can be seen that the trends observed for bayesian hold true for this classifier as well.

3.3 Linear Support Vector Machine (SVM):

A Linear support vector machine was implemented for the data-set data.mat for the neutral vs expression binary classification task. The dual optimization problem for the SVM was solved using Quadratic Programming(Quadprog) on Matlab.

A slack parameter 'C' was chosen incorporated to ensure improvement in the linear separability of the data in the data-set to help improve performance. The accuracy of the classifier was calculated for different values of C. Ii observed by increasing the value of the slack parameter the accuracy also increases. Its is also important to note after a certain threshold of the slack parameter, it will have no effect on the accuracy, which will remain constant, indicating maximum possible separability of data for that data-set has been achieved.

Slack Parameter C:	Accuracy (%):
0.1	70.5
0.15	82
0.2	83
0.25	85.5
0.3	86
0.4	85.5

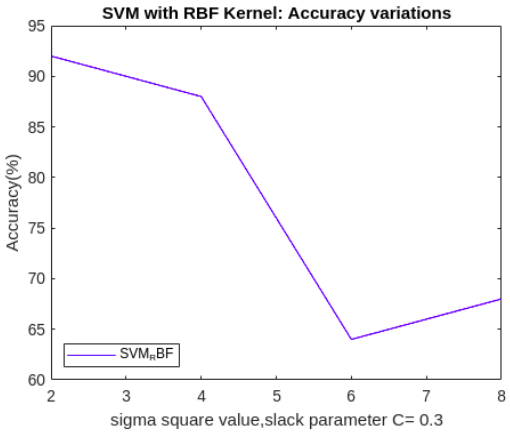


3.3.1 SVM with kernels: RBF and Polynomial:

Two kernels were used along with the SVM to study the effect on the accuracy of the classifier. As expected, these classifiers performed very well on the dataset, since it was highly separable to begin with.

In the case of the RBF SVM the results were tabulated by varying the sigma-square value, but keeping the slack parameter constant (C=0.3). The trend that was observed was with the increase in the sigma-square value, the accuracy decreased, which was expected considering the structure of the kernel. Therefore, smaller the value of sigma-square the better the accuracy of the classifier.
Note: Proper tuning of the slack parameter and the sigma-square parameters are required to find 'best' possible classifier.

Sigma-square: C =0.3	Accuracy (%):
2	92
4	88
6	64
8	68



In case of the kernel for polynomial, the accuracy of the classifier as also high. the parameter that was tuned was r. The highest accuracy that was obtained was 92%. Since the data was linearly separable the accuracy remained constant for different r values.

r: C =0.3	Accuracy (%):
2	92
4	92
6	92
7	75

4. Task: Subject Classification

4.0.1 Bayesian Classifier:

For the task of subject classification using Bayesian classifier was computationally intensive. In addition the data-set data.mat contained 200 subjects, with only 3 images per subject. So the data available was limited to say the least.

The train-test split was done in such a way that the training set would include 2 images and the testing set would include 1 image per subject. As a result, for each subject about 200 posteriors were calculated to determine the class of the subject.

The classification was carried out 3 different combinations of train-test split to study the effectiveness of the classifier.
The observed results are as follows:

Train-set	Test-set	Bayes Accuracy (%)	Bayes with MDA (%)	Bayes with PCA (%)
Neutral + Expression	Illumination Variation	64	58	64
Neutral +Illumination variation	Expression	67	70.5	66.5
Expression + Illumination	Neutral	71	78.5	69.5

The best set of accuracies were obtained when the illumination variation images in combination with either expression or neutral images were in the training set. The combination of Expression+ illumination images in train and neutral images in the test yielded the best subject classification accuracy.

Based on the results, it can be inferred that the illumination images added variation to the training set as a result it performed better for test sets. In the first case, the training set has lower bias as both neutral and expression images are quite similar as a result when tested with the completely unknown illumination images it performed badly. Moreover, that ,may not be the only reason why the classifier performance was relatively bad, lack of access to good amount of images per subject also contributes to this case.

The Bayes with LDA performed as expected and yielded the best performance as a whole when compared to all the other classifiers.
The Bayes with PCA performed similar to normal Bayes classifier, even though there was a reduction dimensions, I think that this is because PCA preserves natural variance of the original data and the lack of sufficient images per subject.

4.0.2 KNN Classifier:

For the subject classification task the classifier was implemented and the results were recorded. The trends observed for the Bayesian subject classifier hold true here as well. In addition, This classifier performs works than the Bayesian Classifier as it is supposed to.

Train-set	Test-set	KNN Accuracy (%)	KNN with MDA (%)	KNN with PCA (%)
Neutral + Expression	Illumination Variation	59	63	55
Neutral +Illumination variation	Expression	65	69.5	65
Expression + Illumination	Neutral	56	78.5	69.5

5. Conclusion

For this project various classifiers like the Bayesian, K-Nearest Neighbors(KNN), Support Vector Machines were implemented for the tasks of binary classification(neutral vs expression) with the data-set: data.mat. In addition for the task of subject classification was implemented through Bayesian and KNN with the data-set: data.mat In addition, dimension reduction techniques like LDA and PCA were employed along with the classifiers to study and record the changes in the effectiveness of the classifiers.

Note:*All my observations and inferences for all the classifiers have been explained in their corresponding sections.*

Word Count: 1452