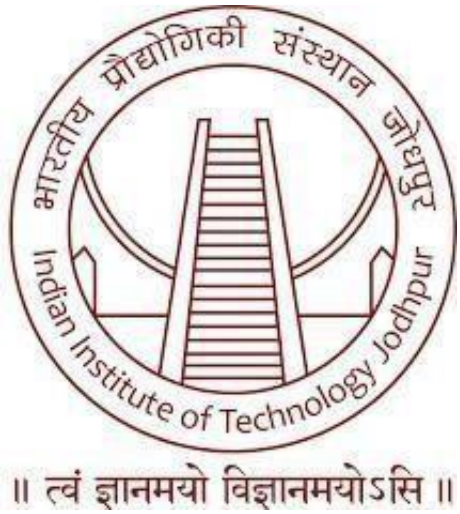
Pattern Recognition and Machine Learning

Indian Institute of Technology, Jodhpur



LAB 7

Report

Harshil Kaneria
Department of Computer Science, IIT Jodhpur

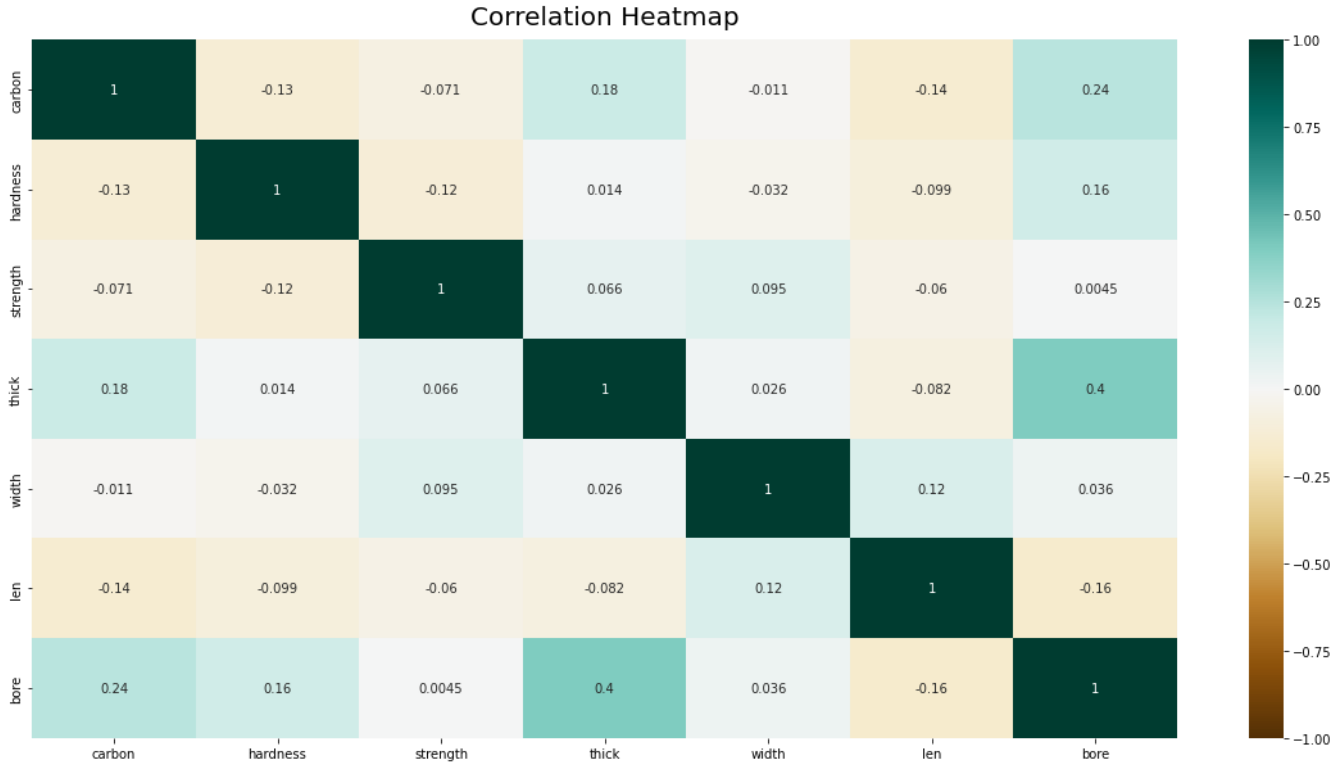
March 21, 2023

1)

1) Pre-Processing and Data Visualisation:

There were many missing values in the dataset given. For the same, I dropped the columns having more than 25% missing values. After that dropped the rows that has missing values. Finally, I was left with 728 rows of dataset. Then encoded the data for further processes.

Correlation Heatmap:



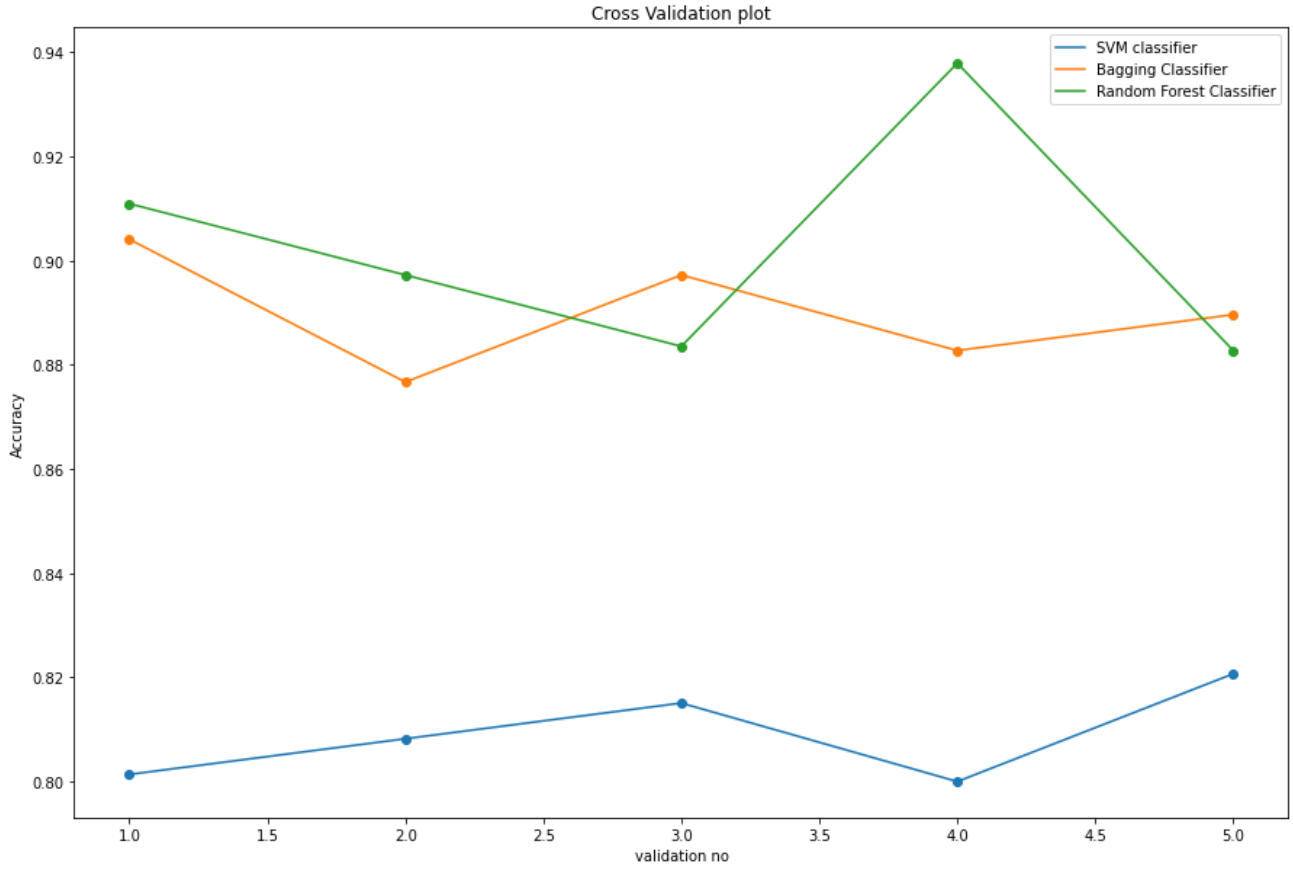
There were no features that were highly correlated.

2) Performed all Classification with and without Data Standardisation:

With Standardisation:

Classifier	Accuracy
SVM Classifier	80.78
Bagging Classifier	86.66
Random Forest Classifier	89.02

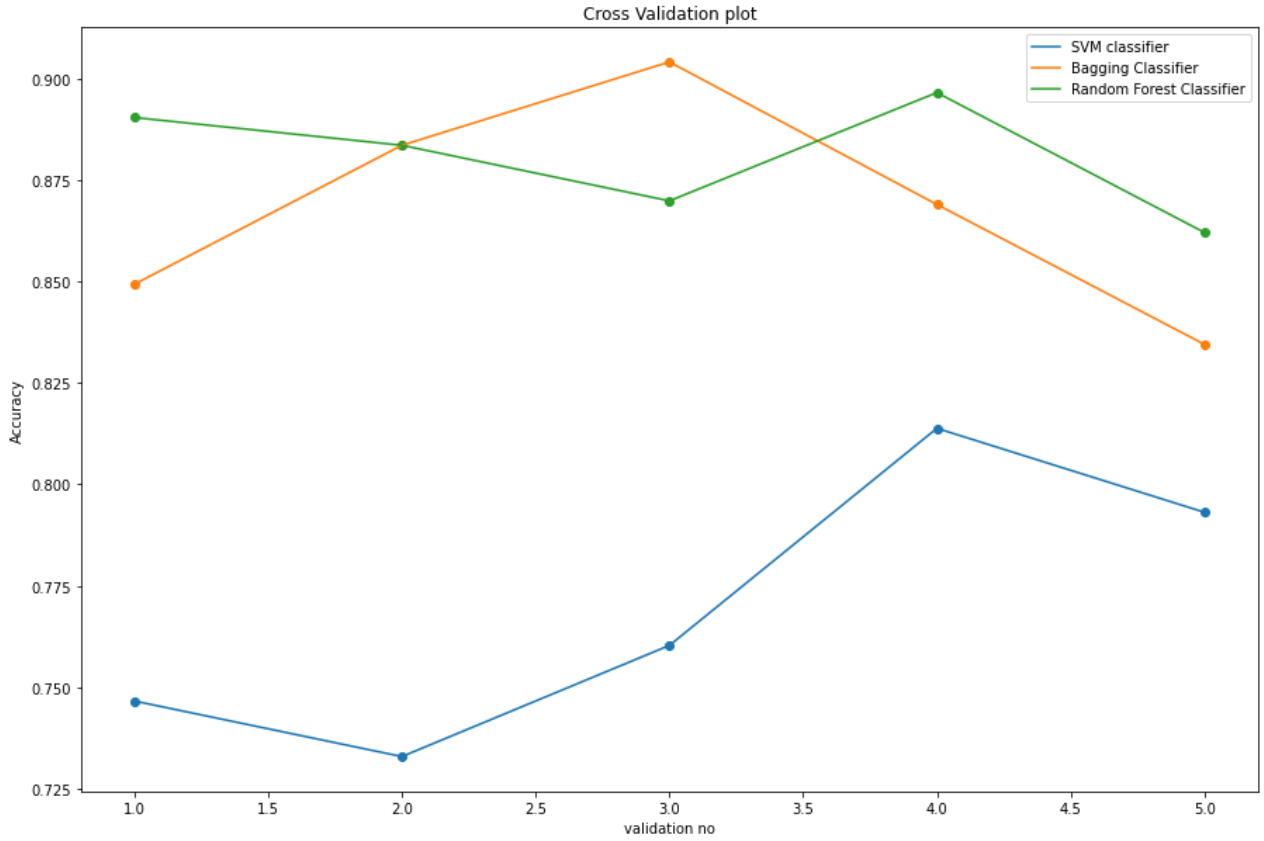
5-Fold Cross-Validation Plot:



Without Standardisation:

Classifier	Accuracy
SVM Classifier	74.90
Bagging Classifier	87.45
Random Forest Classifier	89.80

5-Fold Cross-Validation Plot:



3) I have chosen Bagging Classifier and Random Forest Classifier other than SVM Classifier for comparison. Also, you can see that I have compared them with respect to data standardisation too.

We can observe there is decrease of accuracy for all the classifier when non standardised dataset is used with respect to when standardised dataset is used.

Also, major change in accuracy is seen for SVM Classifier (about 5 %). One of the assumptions of SVM is that the data is centred and standardized. By standardizing the data, we ensure that all the features are on the same scale, and none of them dominate the optimization process. This leads to better performance of the SVM classifier as it is able to find a better hyperplane that can separate the classes more effectively. That's why accuracies after standardisation are good.

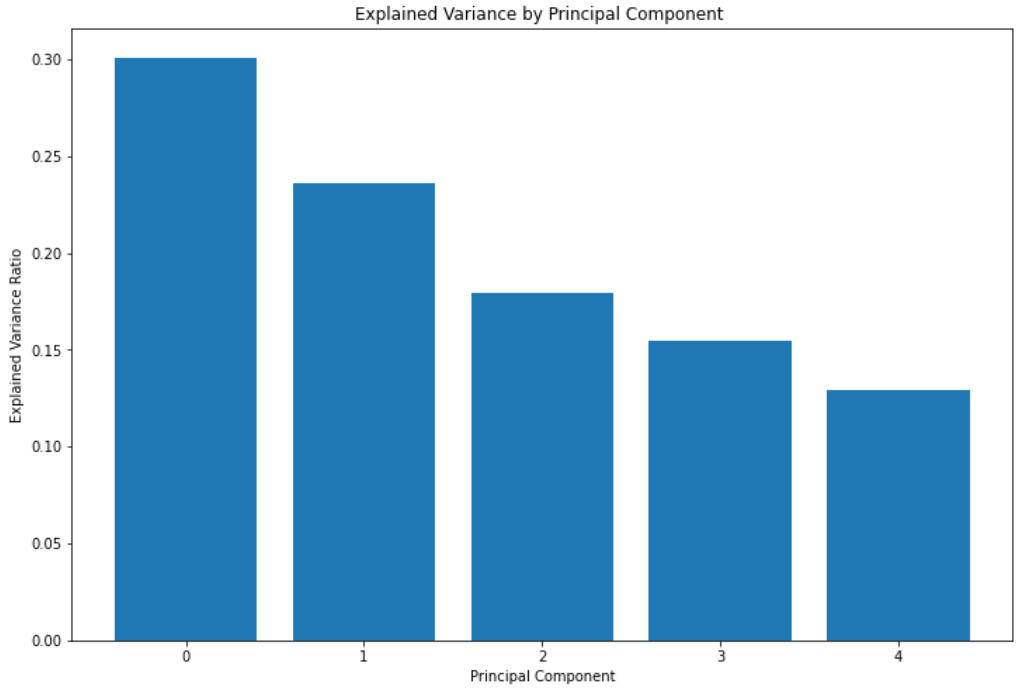
4) Implemented PCA from Scratch:

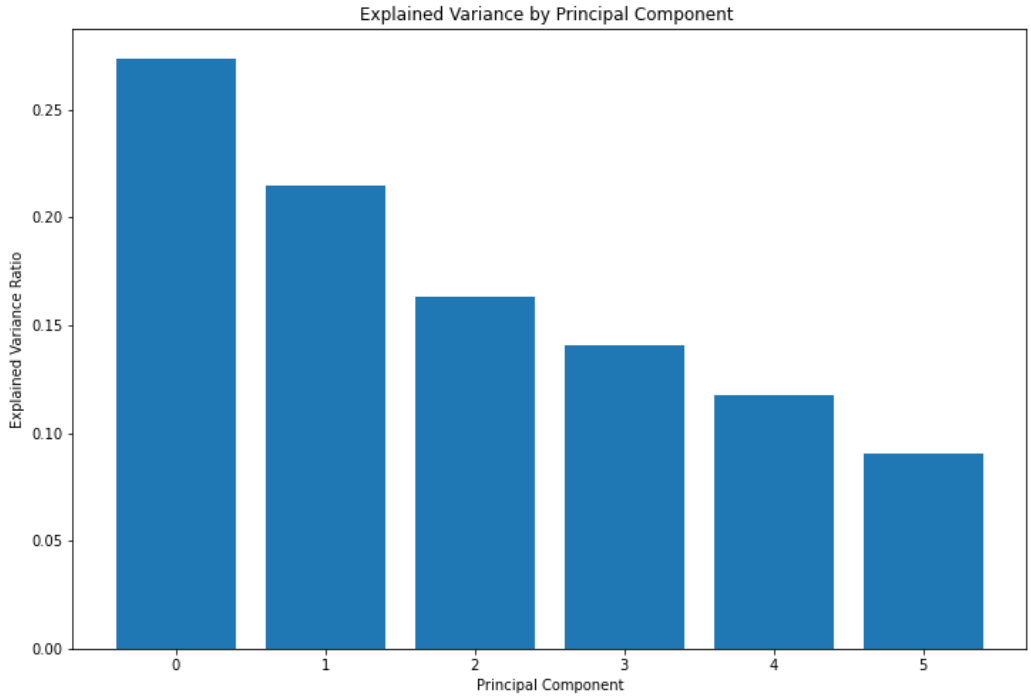
Covariance Matrix is being calculated from scratch

Eigen Vectors and Values is calculated with numpy functions but the Principle components are being identified from scratch.

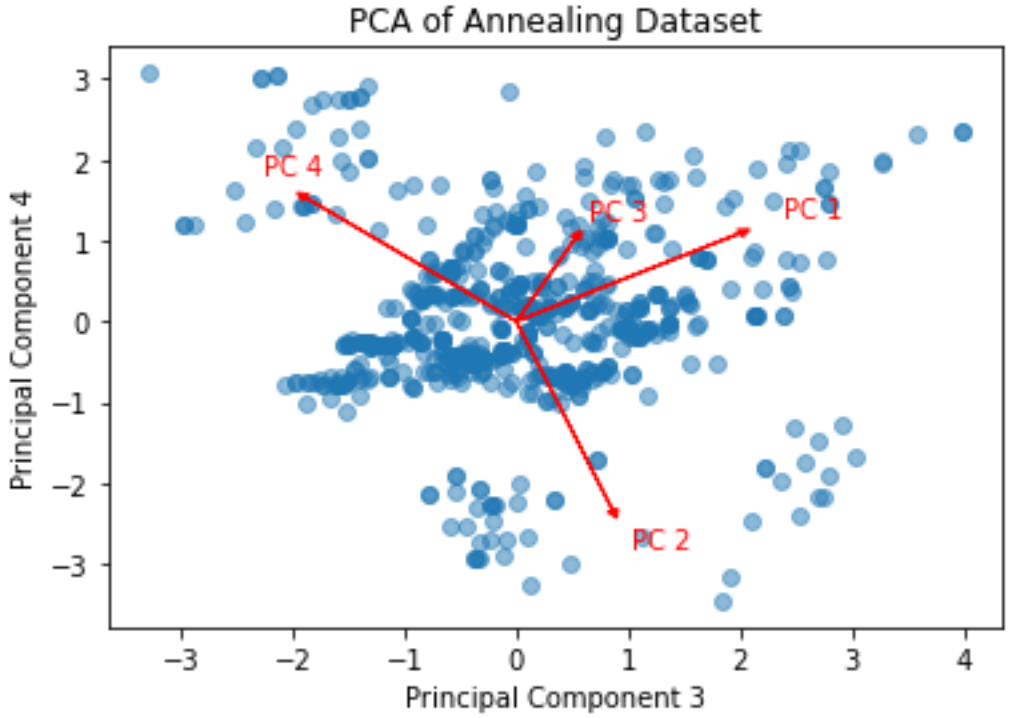
5) The Implementation of PCA:

Some of the Bar plots according to n_components in PCA:





Scatter plot from the reduced Data along with Eigen-Vectors along Principal Components:



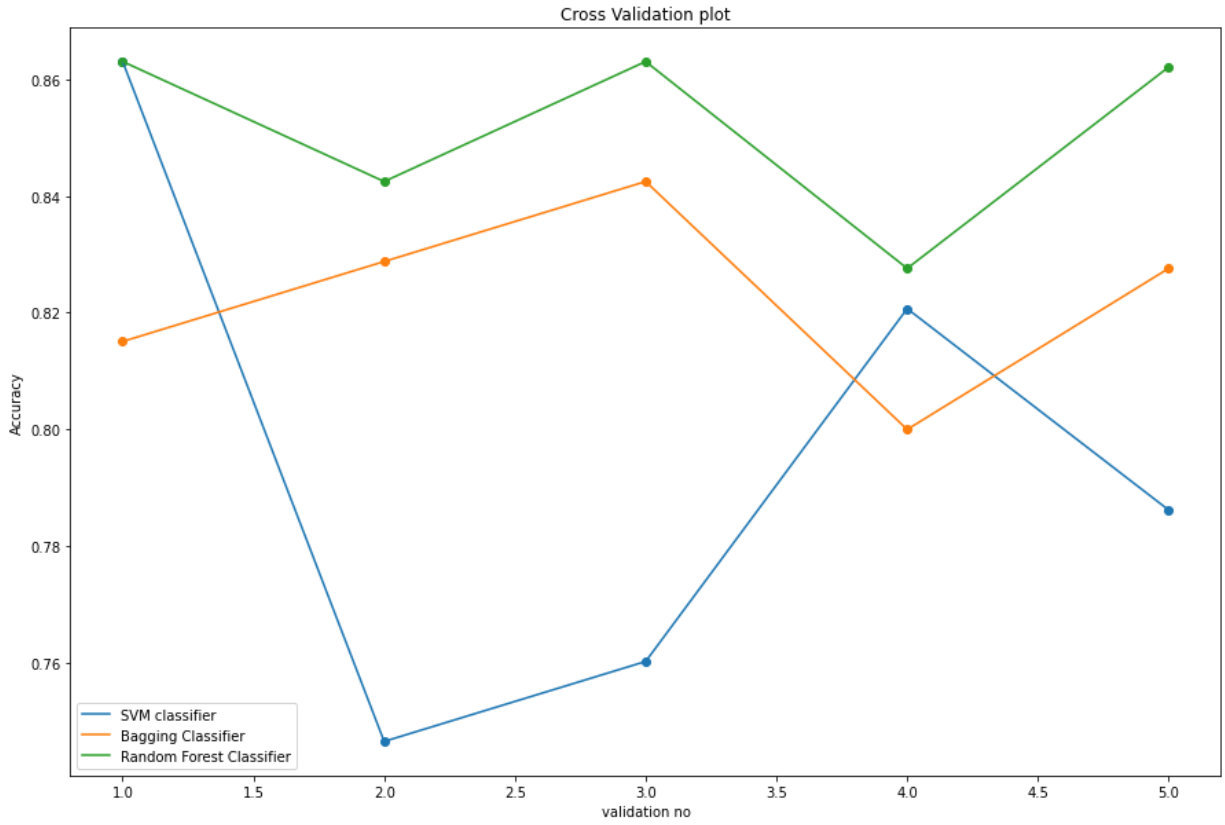
6) Model Training with Reduced Dataset and Plotting:

Trained SVM Classifier, Bagging Classifier and Random Forest Classifier for the Reduced Dataset by PCA.

Accuracy Table:

Classifier	Accuracy
SVM Classifier	76.86
Bagging Classifier	81.57
Random Forest Classifier	86.27

Cross Validation Plot:



2)

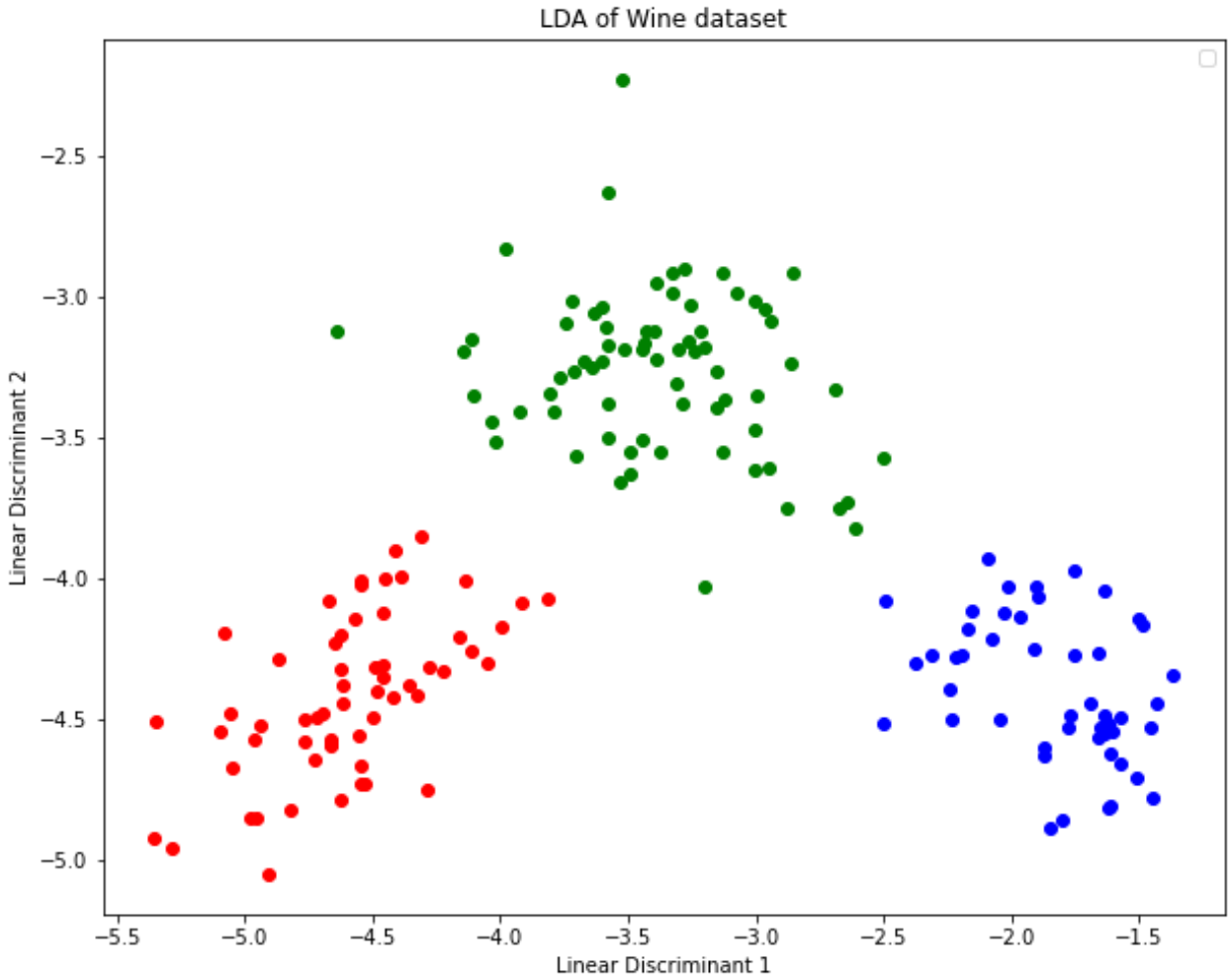
Pre-Processing the Data: The data is already pre-processed and ready to use.

1) Implement LDA from Scratch:

- It has a fit function that calculates “within class and between class scatter matrices”
- It has function that automatically select number of linear discriminants based on percentage of variance that needs to be conserved.(here > 95%)

2) Visualisation:

Scatter plot of Dataset after LDA is implemented over it:



3) Comparing PCA and LDA:

LDA:

Classifier	Accuracy
Bagging Classifier	88.88
Random Forest Classifier	100

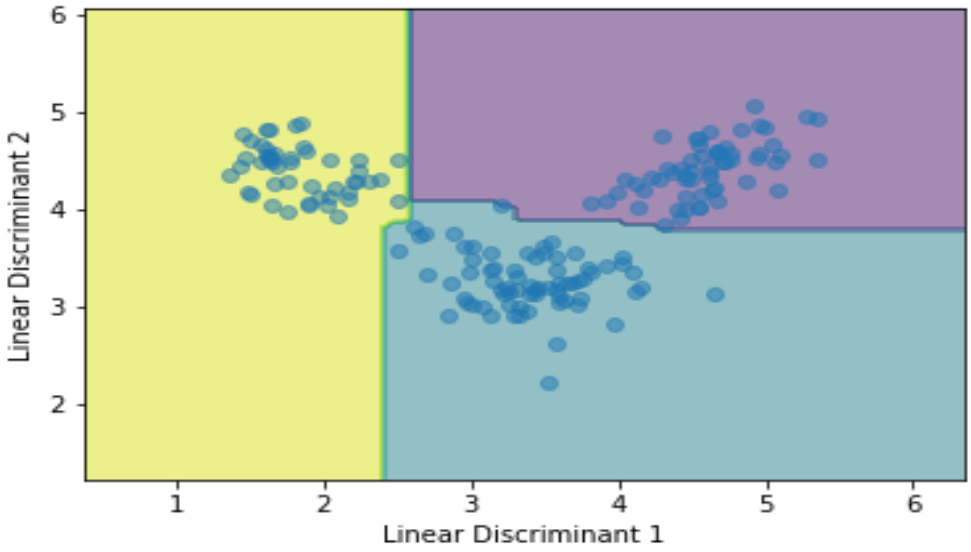
PCA:

Classifier	Accuracy
Bagging Classifier	66.66
Random Forest Classifier	81.48

Reasons why LDA can be better than PCA:

1. LDA takes into account the class labels of the data, and therefore it is better suited for classification tasks where the goal is to separate the data into different classes.
2. LDA aims to preserve the information that is most relevant to the class separation, while discarding the information that is not useful. PCA, on the other hand, preserves the information that explains the most variance in the data, regardless of its relevance to the classification task.
3. LDA can handle datasets with small sample sizes, while PCA may not be effective in such cases.

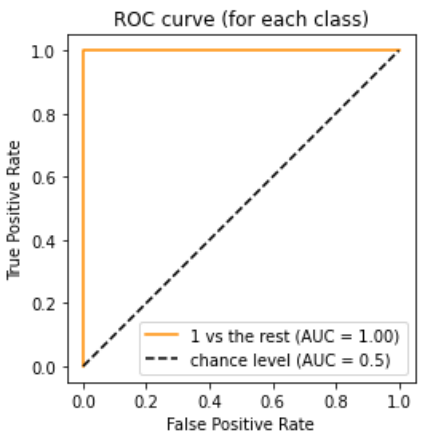
4) The Decision Boundary and Scatter Plot:

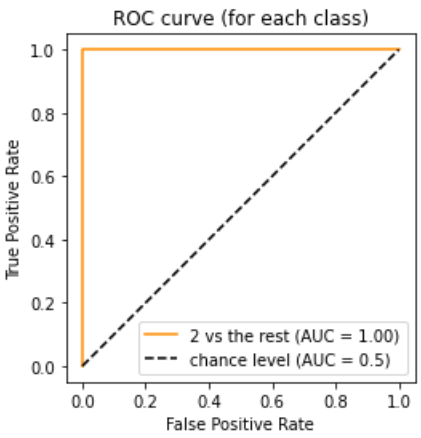


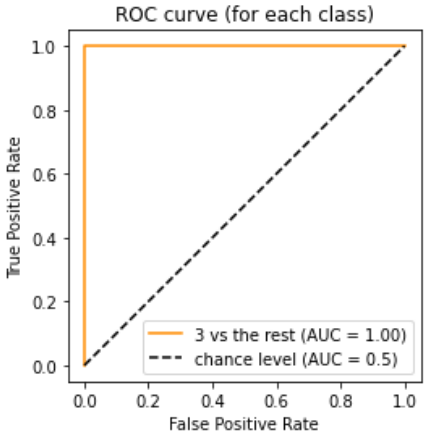
5) LDA as Classifier:

Performed 5-fold Cross Validations and plotted ROC Curves with AUC values. Below shown are some of the ROC curve plots:

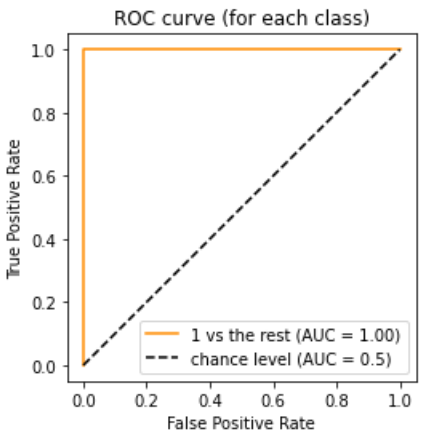
Fold 1:

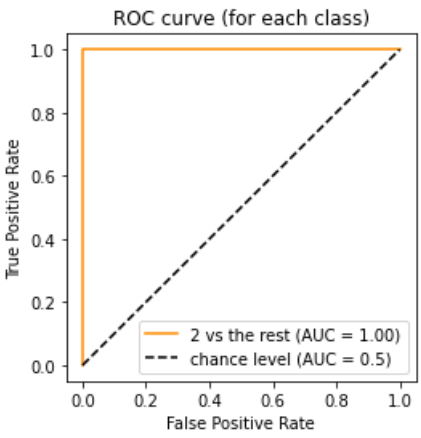


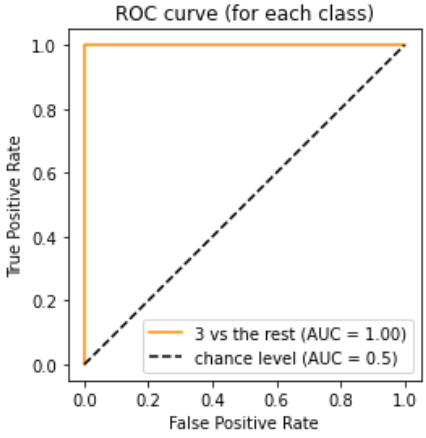




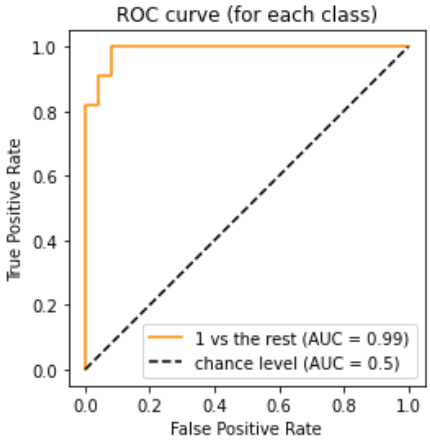
Fold 2:

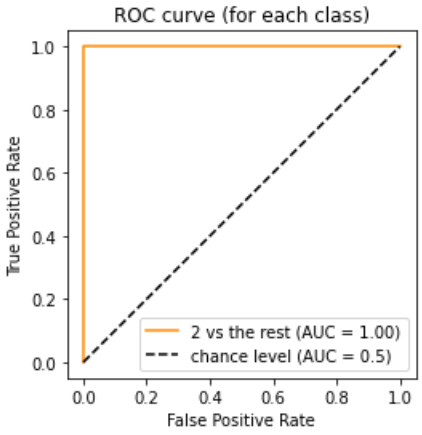


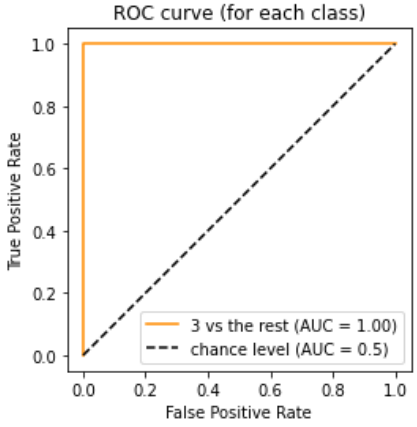




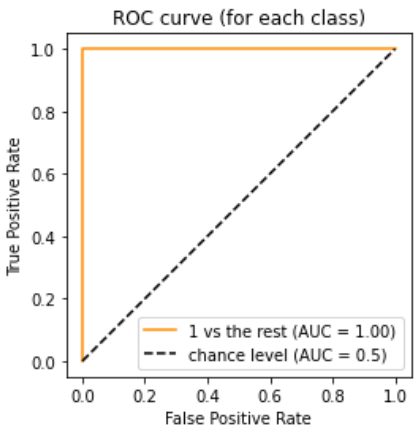
Fold 3:

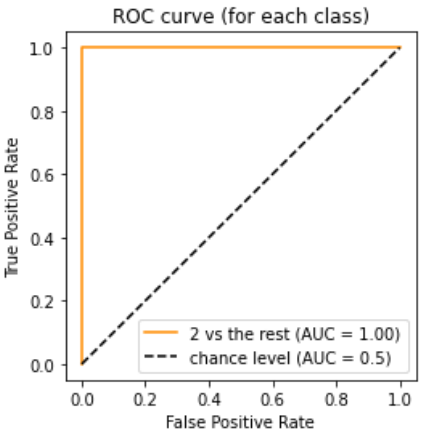


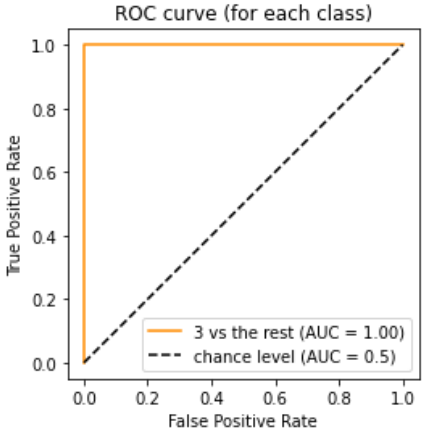




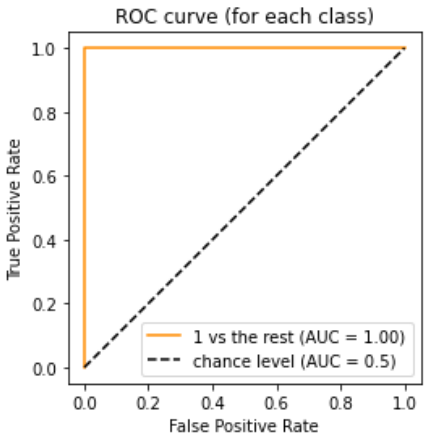
Fold 4:

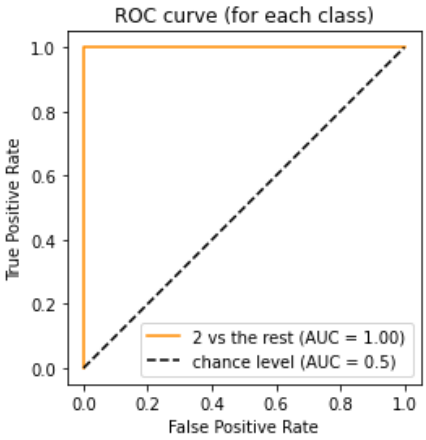

ROC curve (for each class)
True Positive Rate
False Positive Rate
1 vs the rest (AUC = 1.00)
chance level (AUC = 0.5)

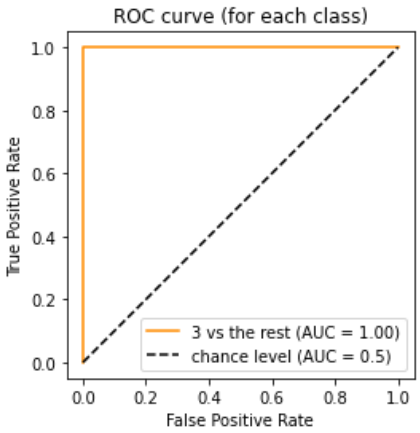

ROC curve (for each class)
True Positive Rate
False Positive Rate
2 vs the rest (AUC = 1.00)
chance level (AUC = 0.5)


ROC curve (for each class)
True Positive Rate
False Positive Rate
3 vs the rest (AUC = 1.00)
chance level (AUC = 0.5)

Fold 5:


ROC curve (for each class)
True Positive Rate
False Positive Rate
1 vs the rest (AUC = 1.00)
chance level (AUC = 0.5)


ROC curve (for each class)
True Positive Rate
False Positive Rate
2 vs the rest (AUC = 1.00)
chance level (AUC = 0.5)


ROC curve (for each class)
True Positive Rate
False Positive Rate
3 vs the rest (AUC = 1.00)
chance level (AUC = 0.5)