Maniya Yash Rajeshbhai
B20CS033

ML Lab 6 Report

Question 1 :

1. Converting Data into Valid Format :

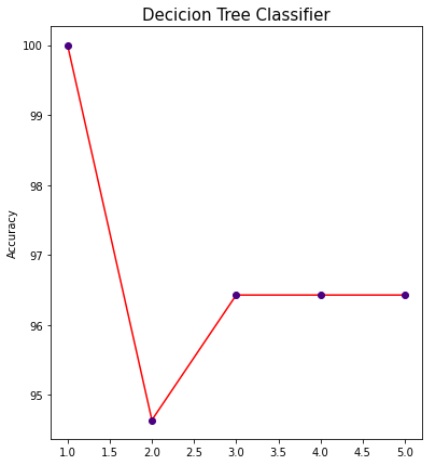
- Converted the data into DataFrames and saw that there were too many trash columns.
- Due to a large number of unuseful features, PCA and LDA are best options to decrease the number of features along with maintaining the accuracy of predictions.

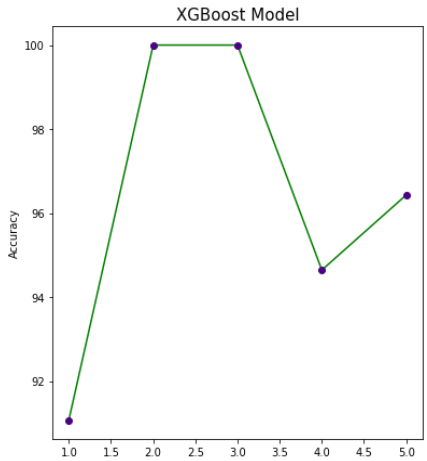
2. Preprocessing :

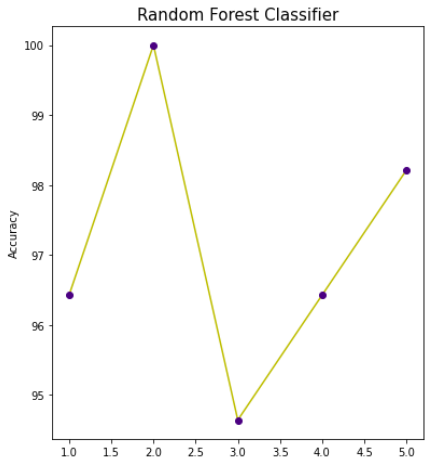
- Unnecessary columns were dropped (filled with a large number of '?').
- Did LabelEncoding where there was a need, and completed preprocessing.
- Split the data into train:test ratio of 65:35.

3. Classification on Raw Data :

- Classification was done using three models on the raw data (without PCA).
- These three models were Decision Tree, XGBoost, Random Forest Classifier.
- The accuracies obtained were 98.21%, 98.57% and 98.93 % respectively.
- Performed Cross Validation for all the models trained. Here are the CV plots:







4. Principal Component Analysis :

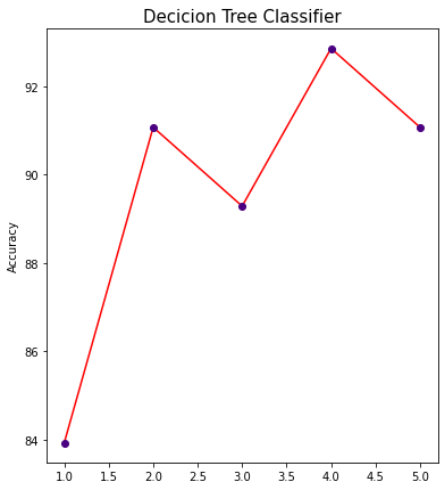
- Centralised the data via feature-wise means and standard deviations.

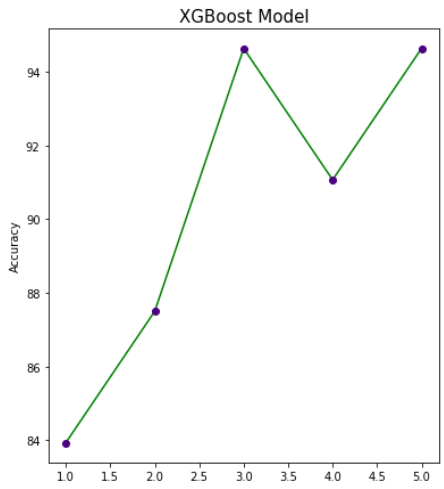
$$\bar{X} = \frac{X - \mu}{\sigma}$$

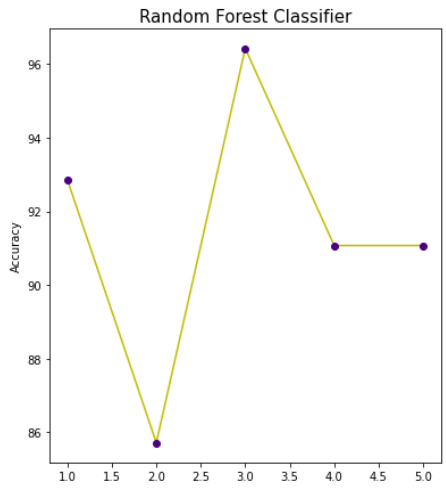
- Made Singular_Value_Decomp() function from scratch which takes training Dataset as input and gives Projection matrix (Principal Components) and EigenValues.

5. Cross Validation after PCA :

- Trained the above mentioned three models after applying PCA to the dataset and did 5-Fold Cross validation.
- The CV plots obtained from it are shown below :







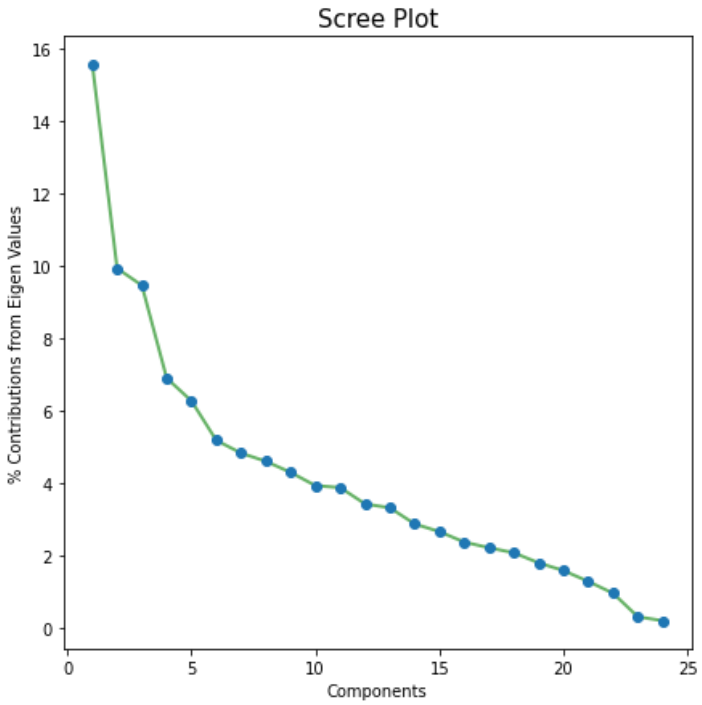
6. Testing the Models :

- Since we have already applied PCA to the training data, we can train the models and put them to predict classes of the Testing data.
- SO I tested the above-mentioned three models on the testing dataset and found the following metrics : Accuracy Scores, Class-wise Sensitivities, and class-wise F1 scores.
- Here are the results we get :

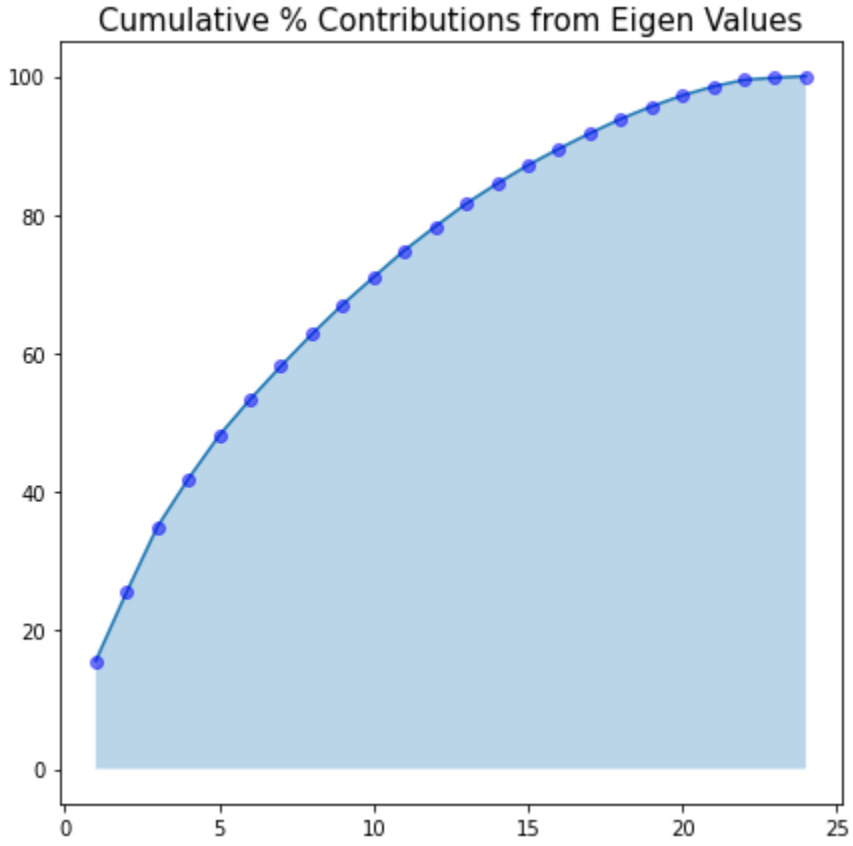
```
Metrics on Testing Data after applying PCA

Accuracy = 92.00 %      [ Decision Tree Classifier ]
Accuracy = 96.00 %      [ XGBoost Classifier ]
Accuracy = 96.00 %      [ Random Forest Classifier ]

Sensitivity for all classes = 0.8182, 0.9474, 0.8571, 0.8333    [ Decision Tree Classifier ]
Sensitivity for all classes = 1.0, 0.9737, 0.8571, 0.8333       [ XGBoost Classifier ]
Sensitivity for all classes = 1.0, 0.9737, 1.0, 0.6667          [ Random Forest Classifier ]

F1 Score for all classes = 0.8182, 0.96, 0.75, 0.8333           [ Decision Tree Classifier ]
F1 Score for all classes = 0.9565, 0.9737, 0.8571, 0.9091       [ XGBoost Classifier ]
F1 Score for all classes = 0.9565, 0.9737, 0.9333, 0.8          [ Random Forest Classifier ]
```

7. Analysis (Before & After PCA):

- We see that the accuracies before applying the PCA are a bit greater than after applying PCA.
- This happens as we have dropped a number of less useful features and hence reduced the dimensions of the dataset.
- However the final accuracies are good enough (0.9 +) considering the fact that we were left just with a small number of features.
- Then after we have plotted the Scree Plot to see the contributions made by the Principal Components. [a scree plot is a line plot of the eigenvalues of factors or principal components in an analysis]



We have also plotted the cumulative percentage contributions of Principal Components. We see that using 10 features can result in recovering the 60% of the original data.



Question 2 :

A. Implement LDA from Scratch :

- Implemented Linear Discriminant Analysis from Scratch using two main functions.
- First function is to calculate the within class and between class scatter matrices.
- Second function calculates the linear discriminants which are necessary to gain a threshold amount of variance conserved to maintain the accuracy of the resulting model.
- The training data is then modified in a way that separation between the classes is maximised.

B. Performing PCA:

- Performed PCA after performing the LDA and calculated metrics on predicted data on the testing dataset.
- Then compared the results between PCA and LDA.
- We see that LDA gives better accuracy in all the three models.

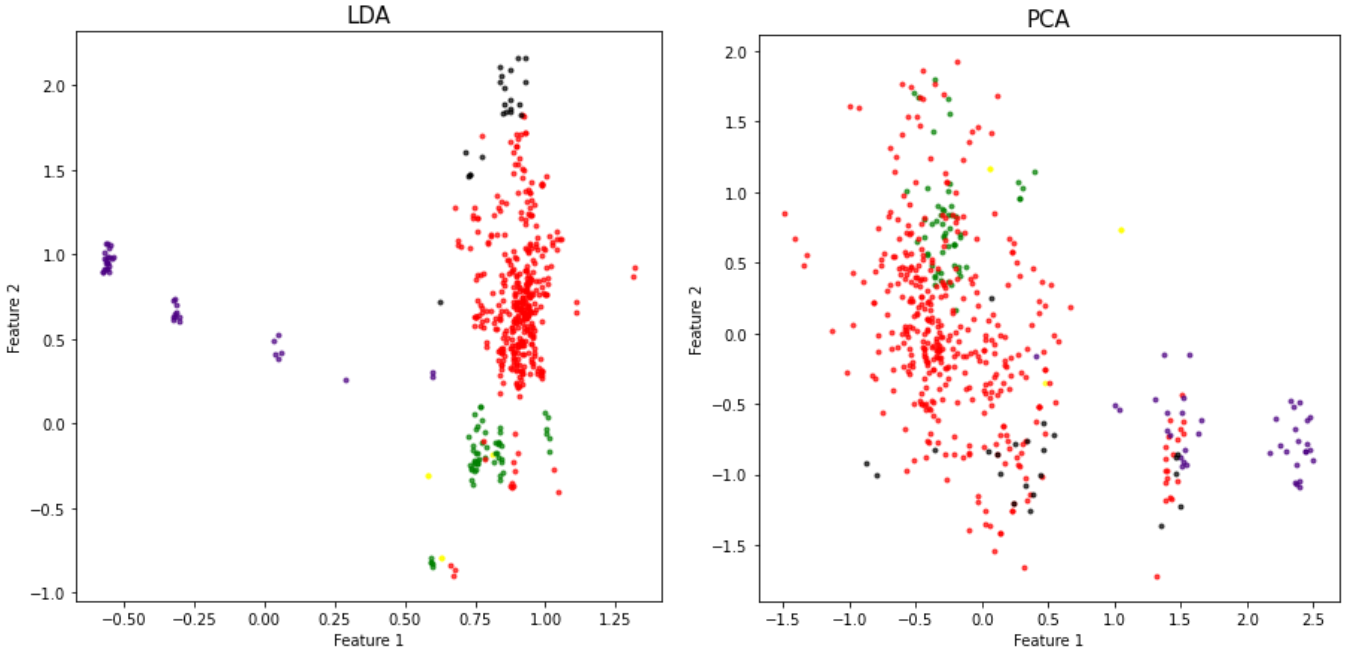
```
Metrics on Testing Data after applying LDA

Accuracy = 98.00 %      [ Decision Tree Classifier ]
Accuracy = 96.00 %      [ XGBoost Classifier ]
Accuracy = 98.00 %      [ Random Forest Classifier ]

Metrics on Testing Data after applying PCA

Accuracy = 92.00 %      [ Decision Tree Classifier ]
Accuracy = 96.00 %      [ XGBoost Classifier ]
Accuracy = 96.00 %      [ Random Forest Classifier ]
```

C. Features having high impact :

From the most contributing Principal Components in PCA and most contributing linear Discriminant in LDA, we found out the descending order of the contributions from the features.

```
The descending order of feature indices having high impact is :

For PCA : 0 1 2 3 4 7 9 11 19 20 21 23 22 18 17 16 15 14 13 12 10 8 6 5
For LDA : 0 1 2 3 4 5 6 7 8 10 9 11 12 13 14 15 16 17 18 19 20 22 21 23

We see that first 5 features have maximum impact on classification common to both methods.
These features are : family, steel, carbon, hardness, temper_rolling
```

And Also plotted the separations of the top two features which were highly impacting the dataset. [**'family'** and **'steel'**]



We realise that LDA gives better separation than that of PCA.

D. Analysis Table :

	Decision Tree Classifier	Random Forest Classifier
PCA	92 %	96 %
LDA	98 %	98 %

We see that LDA performs far better for both the models. This is because LDA maximises the class separability and which is an important factor while reducing dimensionality. This highly improves the accuracy of the model.

End of the Report !