PRML Minor Project

Huzefa Aiyub Ansari B21CS035

Himanshu Gupta B21CS034

Dhyan Yajnik B21AI054

Introduction:

It is important that credit card companies are able to recognize fraudulent credit card transactions so that customers are not charged for items that they did not purchase.

The dataset contains transactions made by credit cards in September 2013 by European cardholders. This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. Feature 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset. The feature 'Amount' is the transaction Amount, this feature can be used for example-dependent cost-sensitive learning. Feature 'Class' is the response variable and it takes value 1 in case of fraud and 0 otherwise.

Methods:

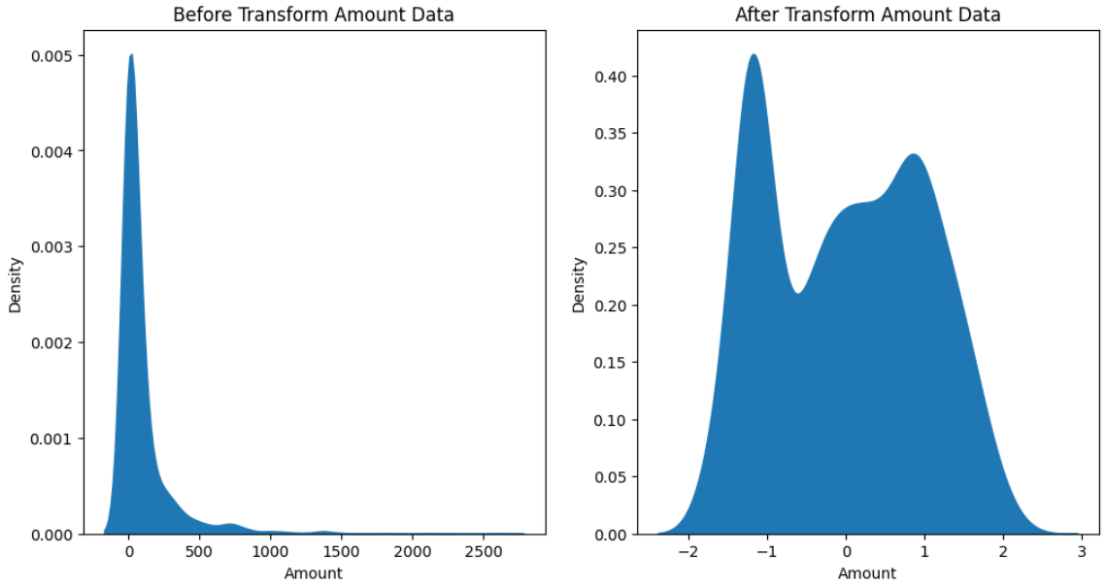
The dataset contained transactions, some of which were fraudulent while others were legitimate. Firstly,we had to create a balanced dataset that could be used to train and test a machine-learning model.

We first imported basic libraries and observed the dataset to gather basic information. We found that there were no null values in any of the columns. We then printed the number of fraud and legal entries, which revealed that the dataset was highly unbalanced, with the positive class accounting for only 0.172% of all transactions.

To create a balanced dataset, we used the Undersampling method and selected 500 samples of the legal class and 492 samples of the fraud class to create a new dataset. We then plotted a heatmap to gain insights into the dataset, which suggested positive covariance between V4 and Class, V11 and Class, and negative covariance between V14 and Class. We also plotted a histogram to visualize the dataset and the KDE (Kernel Density Estimation) of the “Amount” column in the data frame to see how the data was spread out.We observed that the column was highly right skewed. Hence to normalize the data, we used a power transformer.

Next, we split the data into features and targets and created training, validation, and test sets. The ratio of the data in the train, validation, and test sets was 70%, 10%, and 20%, respectively. To standardize the data, we used a standard scalar.

Then we applied various machine learning algorithms to our dataset like logistic regression, decision tree, random forest, naïve bayes, PCA, LDA, XgBoost, etc, and observed the results of these algorithms and then plotted them.

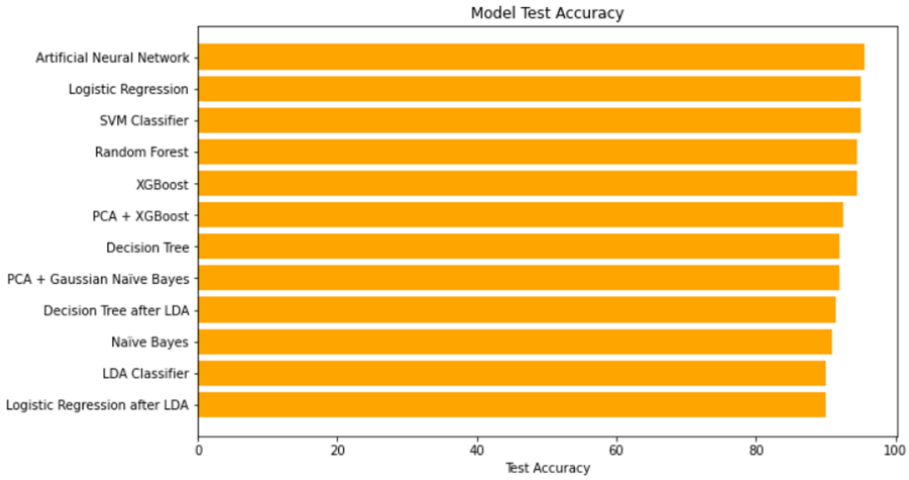


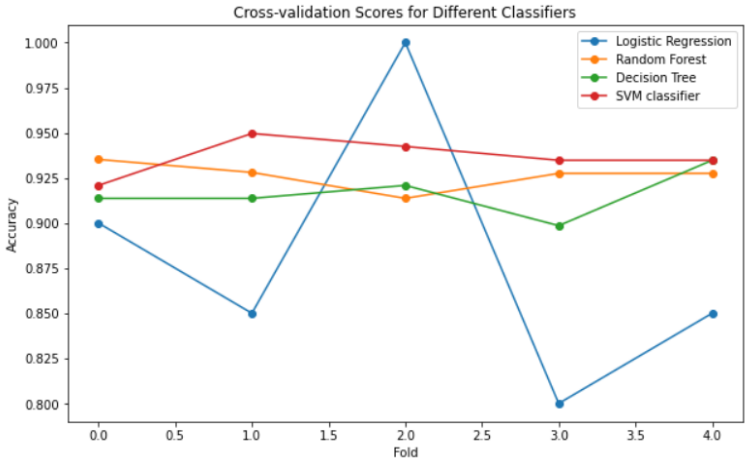
Results:

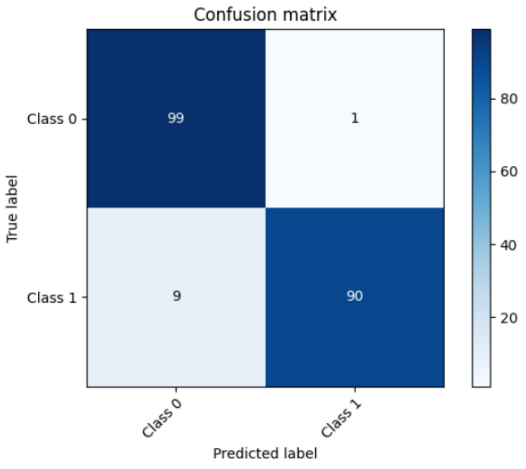
Following are the results obtained for the various models on the dataset. The table below shows the best hyperparameters, the mean validation accuracies and the test accuracies of all the models.

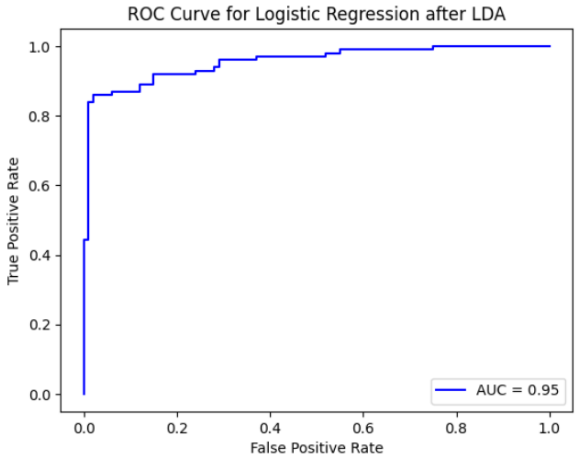
Model	Best Hyperparameters	Validation Accuracy	Test Accuracy
Logistic Regression	**{'C': 1, 'penalty': 'l2'}**	**95%**	**94.97%**
Random Forest	**{'max_depth': 10, 'min_samples_split': 10, 'n_estimators': 50}**	**92.64%**	**94.47%**

Decision Tree	**{'criterion': 'gini', 'max_depth': 5, 'min_samples_leaf': 2, 'min_samples_split': 2}**	**91.63%**	**91.96%**
Naïve Bayes	**{'var_smoothing': 1e-09}**	**90%**	**90.95%**
XGBoost	**{'colsample_bytree': 0.8, 'learning_rate': 0.1, 'max_depth': 3, 'n_estimators': 100, 'subsample': 1.0}**	**94.00%**	**94.47%**
PCA + Gaussian Naïve Bayes	**-**	**90%**	**91.96%**
PCA + XGBoost	**-**	**94.00%**	**92.46%**
LDA Classifier	**-**	**90.00%**	**90.95%**
Logistic Regression after LDA	**-**	**87.00%**	**90.00%**
Decision Tree after LDA	**-**	**91.46%**	**91.46%**
SVM Classifier	**-**	**94%**	**94.97%**
Artificial Neural Network	**-**	**-**	**95.48%**







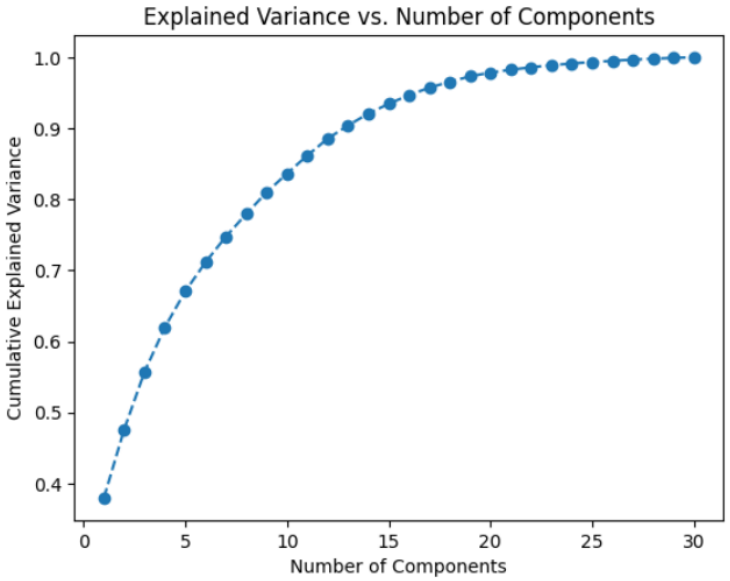


Confusion Matrix for SVM classifier

DISCUSSION:

Initially we saw that the number of fraudulent transactions were very less in comparison to the legal transactions. Hence for an unbiased training of our machine learning models, it was necessary to undersample them and bring them to a comparable level. Once we had the balanced dataset, using EDA we could observe some key differences between the fraud and legal transactions such as the mean time of a fraud transaction was noticeably higher than that of the legal transactions, similar pattern could be observed for the 'Amounts' column and many others. Out of all the models we used, ANN gave the maximum accuracy. Moving forward we also tried to reduce the dimensions of the dataset without compromising much with the information. Hence, we chose the optimal number of components which could capture around 90 percent of the variance and projected the data values onto those directions to form a new dataset with reduced columns.We found that for 13 features were able to explain 90% of the variance of the data. It also showed approximately the same accuracy with the models but there was some loss of information which led to the slight decline in the accuracy of the models. Even after applying LDA there was no improvement in the accuracy, therefore we can also infer that the data which we got lacked some good discriminative information.

Lastly we came up with Artificial Neural Networks, since ours was a case of binary classification, hence we chose 'binary cross entropy' as our loss function and 'sigmoid' as our activation function. Due to time constraints we were able to achieve an accuracy of around 96 percent in just 50 epochs.



CONCLUSION:

In conclusion, we successfully preprocessed the imbalanced dataset by using undersampling to create a balanced dataset, performing data visualization to gain insights, and standardizing the data using standard scalar. These steps prepared the dataset for machine learning analysis.

Then we were able to successfully classify the data into ‘fraud’ or ‘legitimate’ class by training and testing on various models. Based on the results of different machine learning algorithms, we can see that the Logistic Regression and SVM Classifier achieved the highest accuracy of 95% on the test set. Random Forest and XGBoost also performed well with an accuracy of 94%. Naïve Bayes, Decision Tree, and LDA Classifier achieved an accuracy of around 90%. The best accuracy we found was equal to 95.48 percent when we trained the data on Artificial Neural Networks.

It is worth noting that the cross-validation scores of different models were relatively consistent, indicating that the models are not overfitting the training data. Among the models that were optimized with hyperparameter tuning, we can see that they generally performed better than their default configurations.

In conclusion, we have seen that different machine learning algorithms have varying levels of accuracy for the given dataset and the choice of the best algorithm would vary from problem to problem.