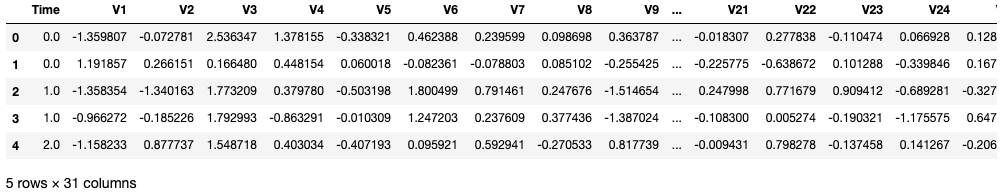
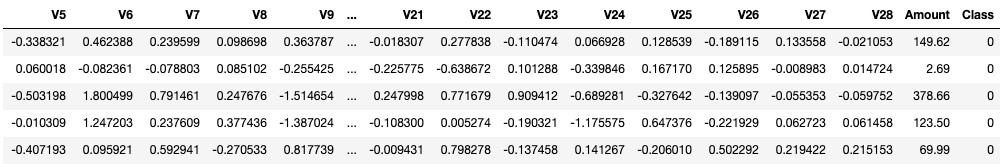
**ML Group 3 Report (Credit Card Fraud)**

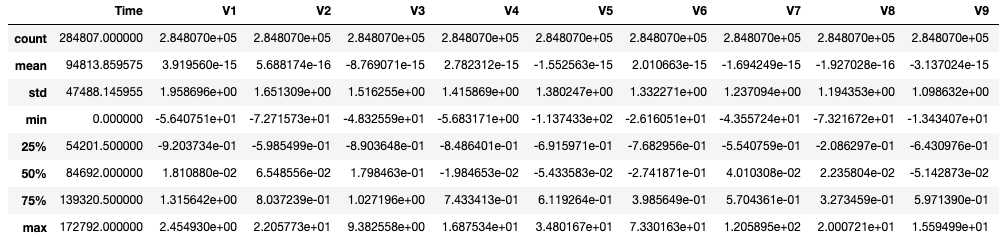
Writeup

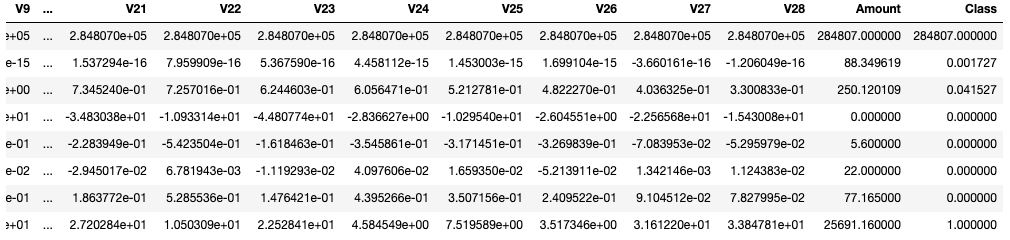
a) Firstly, df.head() is to have a sneak peak of the data. There are 30 features in total, V1-V28, time and amount. The ground truth labels are under the “Class” column.





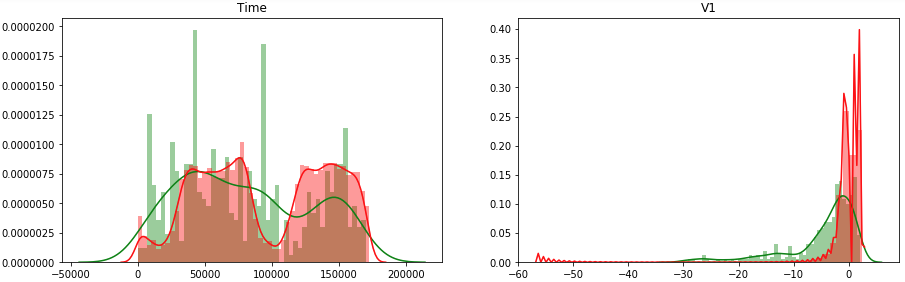
We use df.describe() to take a look at some of the statistical features of the data. Notice that the means of V1-V28 are very small suggesting that they have been normalized. This is consistent with the fact that they have been transformed using PCA. Time and Amount have not been normalized so we have to normalize them before fitting into our model.



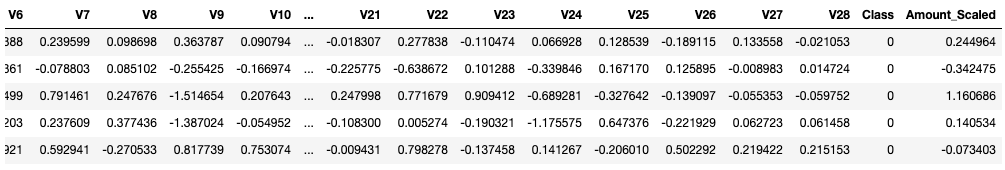


b) Next, we check for null values in the data using df.isnull().values.any(), which there are none.

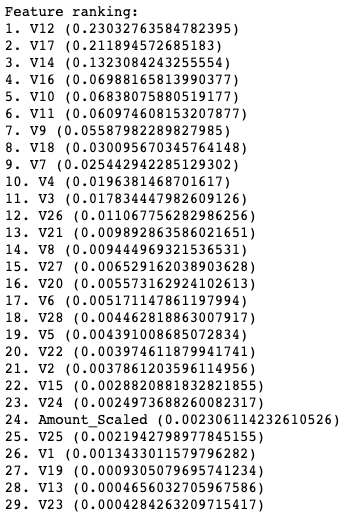
To understand more about the features quantitatively, we plot their distributions (only a preview is shown due to space constraints, refer to the workbook for the entire plot).



The distributions of “Fraud” and “Non-fraud” classes are very similar for the time feature and is not very meaningful, thus we drop it. We also drop the “Amount’ feature after normalizing it using StandardScalar and forming a new column called “Amount\_Scaled”.



Additionally, feature importance is calculated using a Random Forest Classifier for us to identify which are the best features to be fitted into the model.

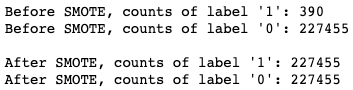


Following that, the distribution of classes is plotted to check if the data is imbalanced. Indeed, the data is severely imbalanced, with the “Non-fraud” class overwhelming the “Fraud” class.

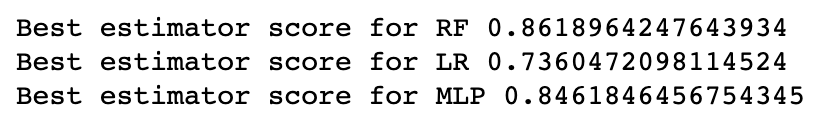


To address the problem of imbalanced data, we perform oversampling on the underrepresented class using SMOTE after splitting the data into train/test sets (80/20 split).

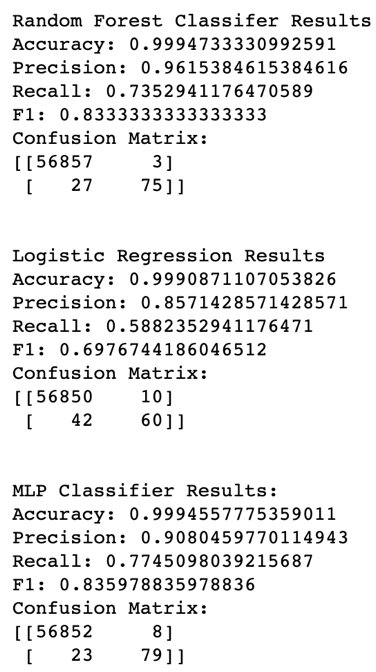




c) Afterwards, we are ready to fit the training data into our model. We will test three algorithms for our model, namely Random Forest, Logistic Regression and the MLP Classifier. To evaluate the performance of the algorithms, we perform 5-fold cross validation, and the best performing one will be used as our actual model. We use f1 as a scoring metric since it optimizes both recall and precision. To ensure that our models perform close to their maximum potential, we do hyperparameter tuning using RandomizedSearchCV. From the cross validation results, the best performing model is Random Forest.



Lastly, we use the trained model to perform prediction on testing data. Looking at the prediction results, Random Forest model has the highest f1 score of 0.83333. It also has a very high accuracy score of 0.99947 and a precision score of 0.96154. The recall score is relatively low at 0.735294, but this is expected since the “Fraud” class is under-represented. From the confusion matrix of the Random Forest Model, we can see that out of 102 “Fraud” samples, 27 of them we incorrectly predicted. However, out of 56860 “Non-fraud” samples, only 3 of them were incorrectly predicted.



d) The limitation of the model is that it misses approximately 25% of “Fraud” transactions, even though it can almost perfectly predict “Non-fraud” transactions. The predictability of the model will improve if more samples of the “Fraud” class is given. The cross-validation f1 score of the training set for the Random Forest model is 0.86190 while the f1 score of the test set is 0.83333, which are very close. This suggests that overfitting is very minimal. To further decrease the gap between the two scores, we can implement a feature selection algorithm in our pipeline to further reduce overfitting.

Questions

1. Dropping of ‘Time’ feature, normalizing of ‘Amount’ feature, oversampling the under-represented minority class (Fraud).
2. The top 10 most important features as classified by Random Forest are:

1. V17 (0.2522050588538928)

2. V14 (0.15414863294285908)

3. V12 (0.14290099328976075)

4. V16 (0.07694328080977224)

5. V11 (0.0711999769706915)

6. V10 (0.06949965973052667)

7. V18 (0.049480062130690824)

8. V4 (0.03127366423649514)

9. V9 (0.02485646255046628)

10. V7 (0.021995515730454357)

1. Random Forest, Logistic Regression and Multilayer Perceptron.
2. More data from the ‘Fraud’ class would be ideal to reduce overfitting.
3. To prevent data leakage, data preprocessing and feature engineering should be redone within each fold of the cross-validation cycle. To make the model more generalized and reduce overfitting, a feature selection algorithm such as RFECV can be included.