IEEE CIS FRAUD DETECTION

Abhinav Singh
MT20202001

Labdhi Kapasi
MT20202066

Shivam Jain
MT2020002

***Abstract*—This is a detailed report on our work on IEEE CIS fraud detection problem, with the aim to predict if a transaction is Legit or Fraud.**

***Index Terms*—EDA, Feature Engineering, Preprocessing, Random Search CV, Decision Tree Classifier, Random Forest classifier, LightGBM Classifier, XGBoost .**

PROBLEM STATEMENT

Financial statement fraud has been a difficult problem for both the public and government regulators, so various data mining methods have been used for financial statement fraud detection to provide decision support for stakeholders.
In day today's electronic world, everyone makes a transaction either in shopping mall or in booking movie tickets or flight tickets etc. So, fraud transaction can happen in any of these places which will penalize the customer heavily as well as the businesses. Hence improving the efficacy of fraudulent transaction alerts for millions of people around the world and helping hundreds of thousands of businesses reduce their fraud loss and increase their revenue.

DATASET

- The dataset contains 4 csv file one for transaction and another for identity in test and train.
- Train file of transaction contains 354324 rows and 394 features and identity contain 86935 rows and 41 features.
- Test file of transaction contains 236216 rows and 393 features and identity contain 57298 rows and 41 features.

Dataset contains 403 numerical features and 31 categorical features.
In Dataset some of the features contains NULL values.
Few important columns are -

- **`IsFraud-`** Ground truth
- **`TransactionID`**- Unique Machine ID
- **`ProductCD`** - product code, the product for each transaction
- **`P_ & R_emaildomain`** – purchaser and recipient email domain
- **`card1 - card6`** - payment card information, such as card type, card category, issue bank, country, etc.
- **`D1-D15`** - timedelta, such as days between previous transaction, etc.
- **`TransactionAMT`** - transaction payment amount in USD
- **`Addr-`** Addr columns represent Purchaser Billing Region and Country

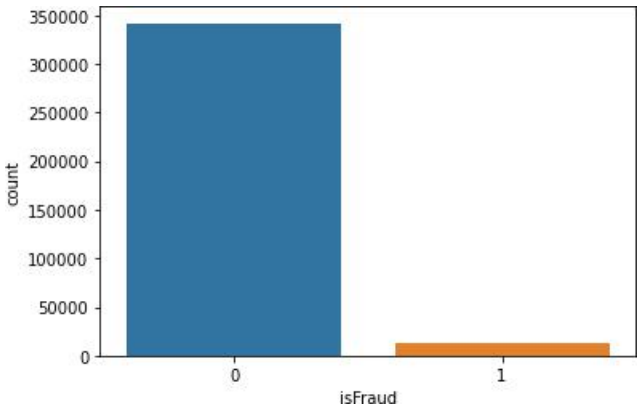
I. INTRODUCTION

With the rapid evolution of the technology, the world is turning to use credit cards instead of cash in their daily life, which opens the door to many new ways for fraudulent people to use these cards in a bad way. According to the Nilson report, global card losses are expected to exceed $35 billion by 2020. To ensure the safety of users for these credit cards, the credit card's provider should provide a service to protect users from any risk they may
face.

The challenges of fraudulent activities increased the demand for systems to detect credit card fraud. Credit card fraud detection is becoming more complex since the fraudulent transactions for the cards are more and more like legal ones. To solve this issue, credit cards´ providers must use more sophisticated techniques to detect fraud transactions. One of the biggest problems in this field is the lack of good datasets since the datasets available for his problem are imbalanced datasets and have a lot of unknown fields for private insurance. Which makes it harder for the programmers to understand the dataset and build the best model that solves this problem.

Consequently, we present our approach to predict legitimate or fraud transactions on the IEEE-CIS Fraud Detection dataset provided by Kaggle.

II. DATA PREPROCESSING



The raw data is very high dimensional and has many missing values. The size of data is also comparatively large. Each row in this dataset corresponds to a Transaction, uniquely identified by a TranasactionId . IsFraud is the ground truth and indicates that Transaction is legit or not. As we can see from above image that IsFraud is highly imbalanced, so while training the model we need to select the model's which could handle imbalanced data and evaluate the predictions using the right performance metrices otherwise results may be misleading.

As we all know the biggest challenge before training the model is selecting the relevant features. Second, preparing the data so that it can properly fit the algorithm. Hence in our project here as well, we tried to preprocess the data. Hence reducing the size of the data and removing non relevant or unnecessary columns was very crucial. Here we have provided the data cleaning and preprocessing steps with description -

- **`Mostly-missing Feature`**- In the given data set there are columns that have more than 99% of missing values removing them will not affect accuracy of model. Below are the features having NULL value percentage > 90%

	Total	Percent
id_24	351487	99.199320
id_25	351248	99.131868
id_07	351237	99.128764
id_08	351237	99.128764
id_26	351232	99.127352
id_21	351232	99.127352
id_27	351228	99.126223
id_22	351228	99.126223
id_23	351228	99.126223
Logdist2	331717	93.619681
D7	330931	93.397851
id_18	327049	92.302243

- **`Too-skewed features`** - There are the columns in dataset which are highly skewed means majority of occurrences are covered by one type of categories so these columns will not contain any useful information hence we can remove them. Below are the mentioned highly skewed features with their skewness value.

V311	303.457524
V129	243.936487
V309	229.943855
V206	180.274756
V319	168.038493
V266	157.560669
V334	140.929155
V131	111.946856
V227	100.807054
V321	100.624658

- **`Highly-correlated features-`** Correlations between columns show how related the two columns are, we have used heatmap to check highly correlated features and removed highly correlated features.
    1. C2, C6, C11, C12 columns
    2. D2, D6, D12 columns

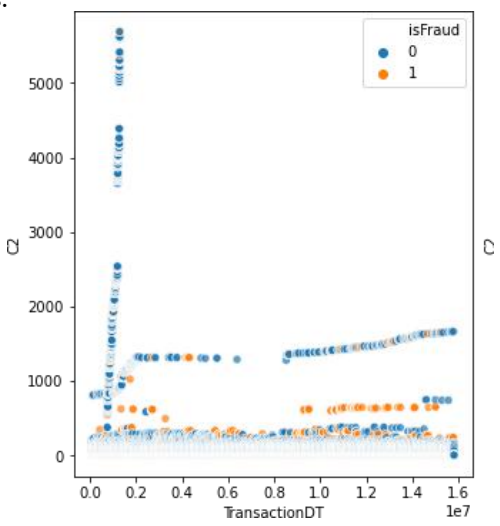
These are the features which can be dropped since they can be derived from another features. So, in total we have removed 29 columns without affecting the performance of model.

Handling Categorical Features

In Dataset there are some categorical features so we need to encode them before giving it to model we have used one hot encoding for features having less than 5 categories followed by label encoding to encode remaining features.
We also tried frequency encoding to encode categorical features but we saw no improvement in Roc score.

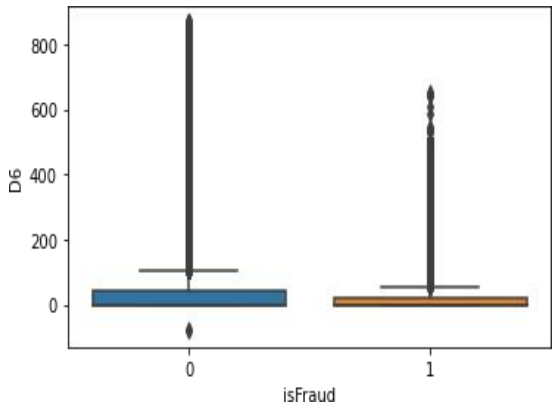
III. Data Visualization

We observed the output to be predicted is binary and highly imbalanced we are provided with 434 features based on which we need to predict whether Transaction is legit or not with some of these features being categorical.
While analyzing these features we have seen some of them are highly skewed and some of them contain high null value percentage and some features have outliers.



C columns

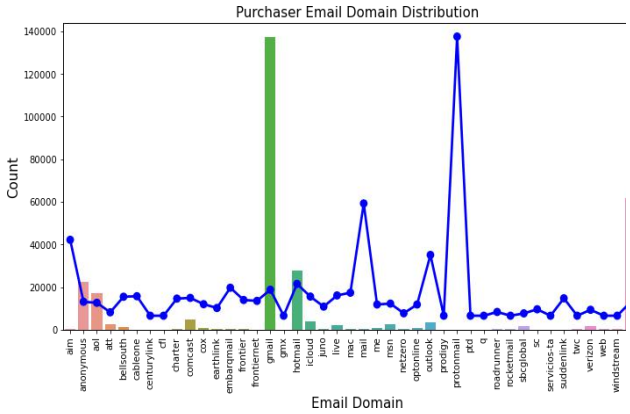
From the EDA we have seen that some of C columns contains outliers above shown in the figure of one of C columns. The outliers are visible from figure for value above 3000. To handle outlier, we have replaced their value with median. This same pattern was observed across all C columns.



D columns

From the EDA we have seen that some of D columns contains negative values as shown in figure of one of D columns and negative value are visible, these negative values can be treated as outliers.

To handle outliers, we have replaced their value with median.



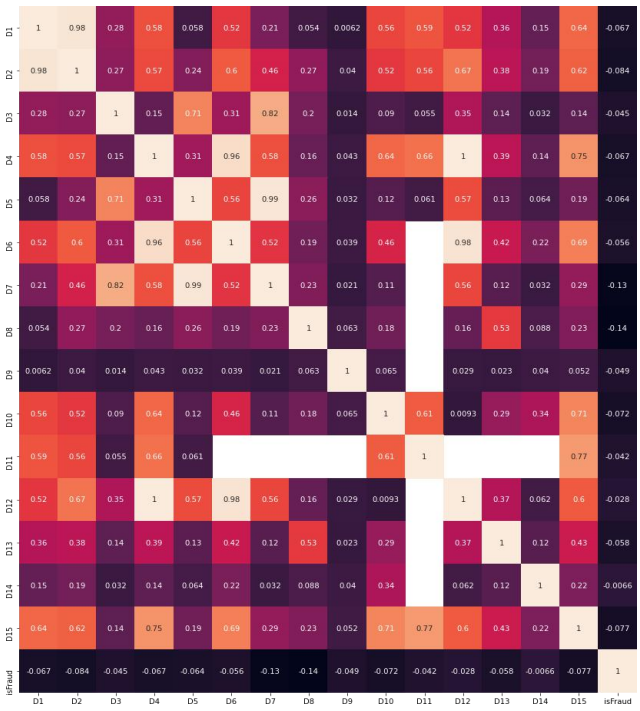
PEmail Domain

Purchaser Email domain is an interesting feature as we can see from above figure, the rate of fraud transaction using proton mail is extremely high.

We also tried to plot heatmap to analyze correlation between features.

From the heatmap we observed that there are some features which have high correlation.

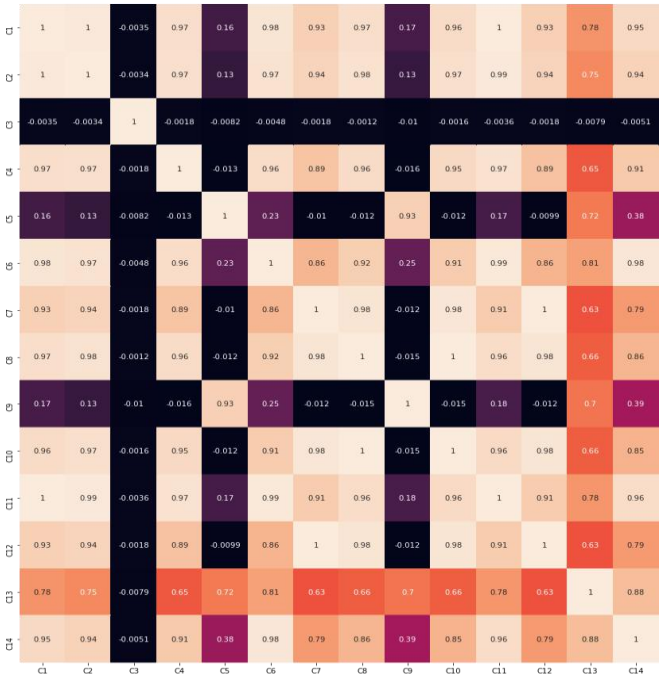
Heatmap for D columns



We can see from above heatmap there are some highly correlated D columns- mention which ones

We have removed columns D2, D6, D12

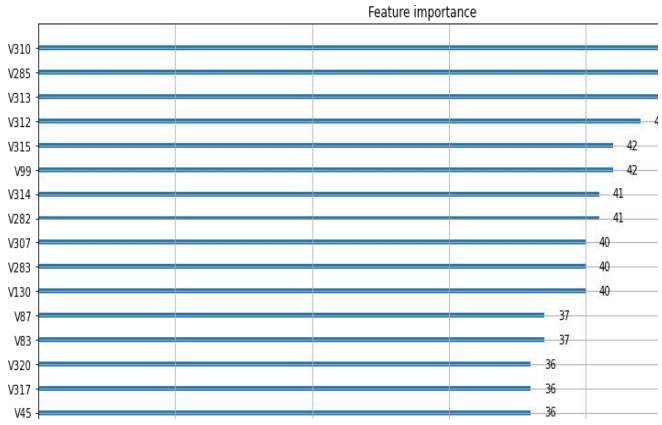
Heatmap for C columns



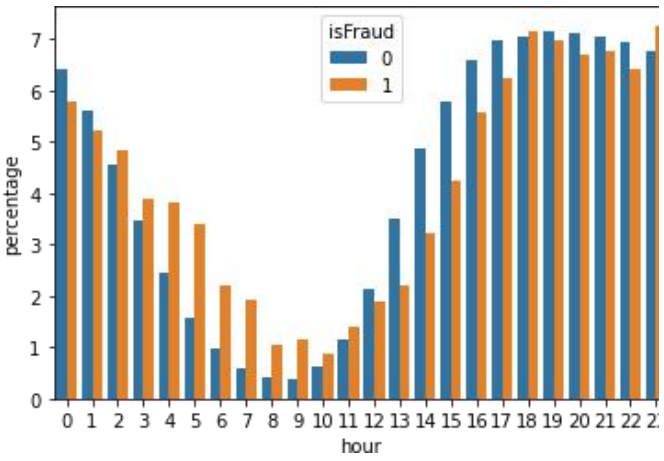
We can see from above heatmap there are some highly correlated C columns – mention names

We have Removed Columns C2, C6, C10, C11

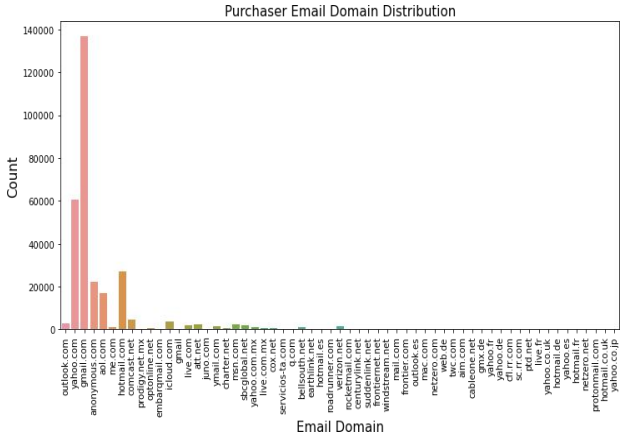
IV. FEATURE ENGINEERING

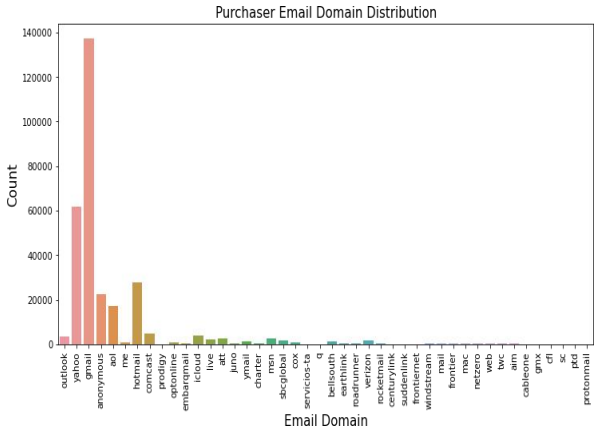


Here to determine the important V features we used LGBM classifier on data. The results we found after plotting the feature importance graph are as shown above. We have found that some of the V features have extremely low importance so we drop the V columns having feature importance less than 3. We also tried to calculate model accuracy including and excluding these V columns but there was no change in accuracy.

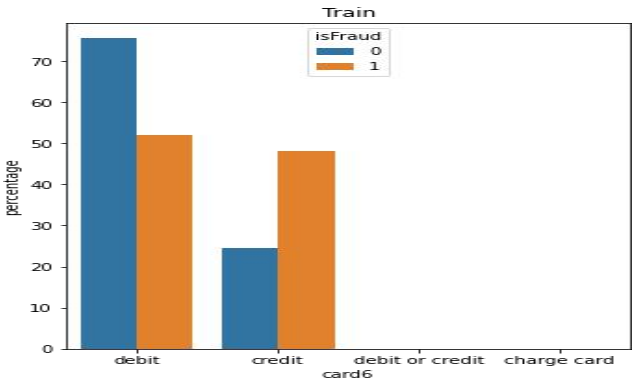


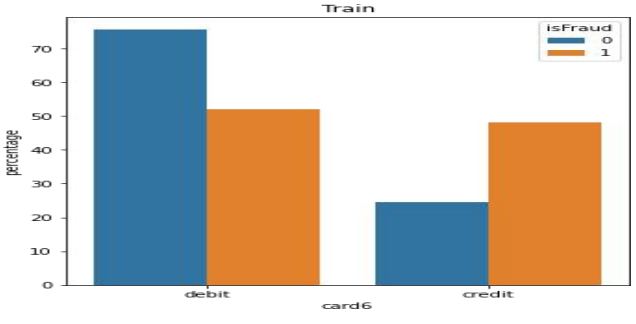
TransactionDT which is a time related feature and the time was in seconds so we created a new feature which represents time in hours after converting it into hours, we saw a very interesting finding as show in graph above where we can see that for hours >=3 and <=9 rate of fraud transactions is high.





In Features Purchaser Email Domain and Recipient Email domain lot of domains came from same distributors like hotmail.com and hotmail.fr, yahoo.com and yahoo.fr. So, we grouped them under the parent distributors.





While Analyzing the Card6 features we have observed count of transactions through debit or credit and change card are extremely low so we combined their count in transaction through debit.

V. Training and Results

After performing data preprocessing where we removed less relevant features and performed encoding of categorical features. We have now total 300 features in our dataset on which we need to train the model, while training the model we calculated the best hyperparameters for each model using Randomized search CV. parameters returned by Randomized search CV to model and fit our model on the training data accordingly. Finally, we made our predictions for the test data using the best hyperparameters.

We trained our data on the following algorithms -

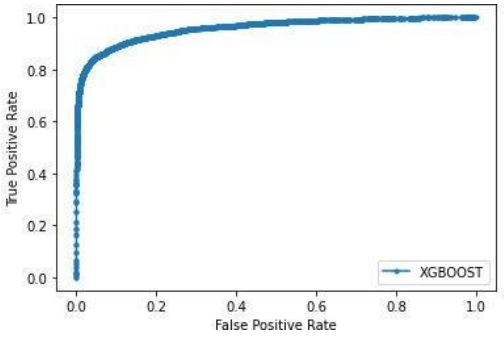
- Decision Tree Classifier
- Random Forest Classifier
- LGBM Classifier
- XGBoost Classifier

The AUC SCORE on these models were the following

AUC SCORE ON VARIOUS ALGORITHMS

Classifier	AUC SCORE
Decision Tree Classifier	0.8361
Random Forest classifier	0.908
LGBM Classifier	0.959
XGBoost Classifier	0.9639

Roc Auc = 0.959



Best Results – Model XGBoost , Train – 0.959 Test – 0.9639

VI. Conclusion

After training our dataset on many different models, we observed that such a real-life dataset, which is highly imbalanced is best fitted by ensemble methods. Even in these ensemble methods the ones which gave the best results were Decision tree based such as Random Forest, Light Gradient Boost and XG Boost. After training and analyzing these models via RandomSearch CV we found that XG Boost gives the best AUC ROC score.

VII. Acknowlegement

We would like to thank our teaching assistant Saurabh Jain for helping us out on numerous occasions, Special thanks to Raghavan sir. His highly detailed lectures on several topics helped us understand concepts very well.

VIII. Refrences

- https://www.kaggle.com/robikscube/ieee-fraud-detection-first-look-and-eda
- https://www.kaggle.com/cdeotte/xgb-fraud-with-magic-0-9600
- https://medium.com/@gtavicecity581/ieee-fraud-detection-469398ce1ac4
- https://nycdatascience.com/blog/student-works/ieee-cis-fraud-detection-detecting-fraud-from-customer-transactions/
- https://www.kaggle.com/suoires1/fraud-detection-eda-and-modeling
- https://www.kaggle.com/rajeshcv/understanding-v-columns
- https://www.kaggle.com/c/ieee-fraud-detection/discussion/104973