IEEE-CIS Fraud Detection Data Analysis & Visualization

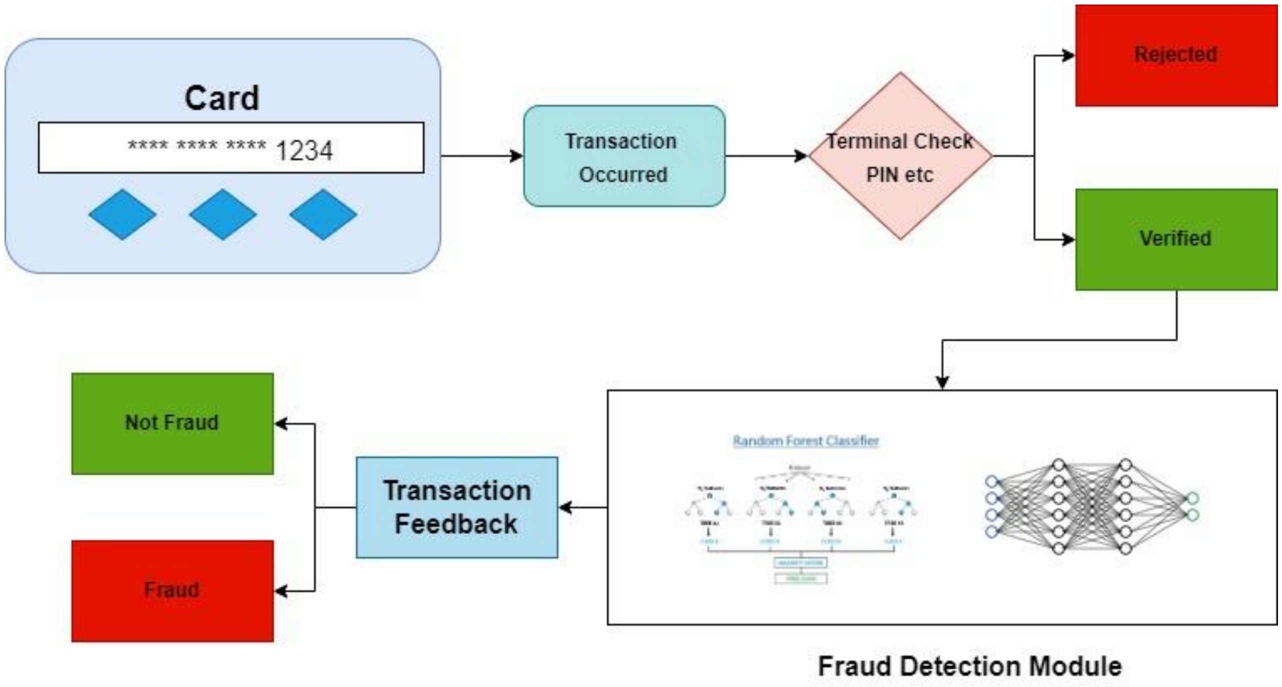
Nikhil Balwani, Zain Merchant, Qingcheng Yu, Haozhong Jia

(Group 19)

Overview

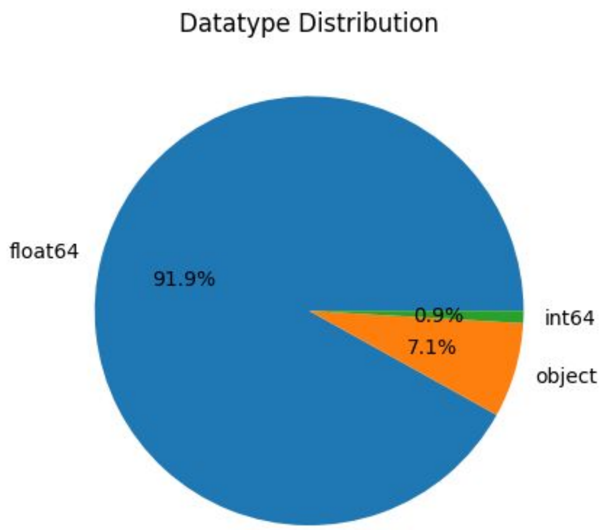
- Initial Data Exploration
  - Imbalance of Target Variable (Fraud)
  - Proportion of Fraudulent Transactions
- Data Cleaning + Sampling
- Insights from Data Exploration
  - Central Tendency Plot
  - Drop columns with too many missing values
  - Correlation Matrix
  - Feature importance via Random Forest
- Proposed ML Techniques to Implement

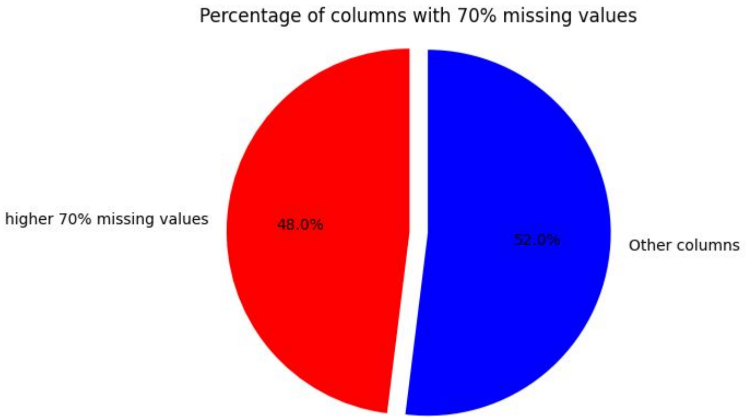
Background


Card
**** **** **** 1234
Transaction Occurred
Terminal Check PIN etc
Rejected
Verified
Random Forest Classifier
Fraud Detection Module
Transaction Feedback
Not Fraud
Fraud

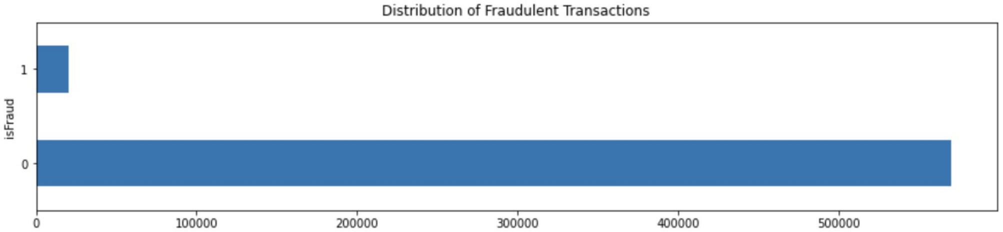
Initial Data Exploration

- The IEEE-CIS Fraud data set contains 433 feature:
  - Numerical: 402 columns
  - Categorical: 31 columns
- About half of the columns contain >70% missing values
- The dataset contains one target variable, describing if the transaction is fraud
  - 1 represent fraud and 0 represent non-fraud
  - most of the samples are non-fraud
- Given the large number of features, but a few important ones were be chosen for EDA

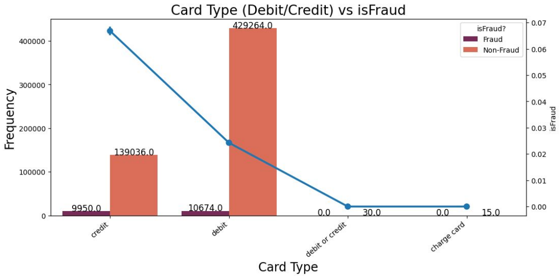


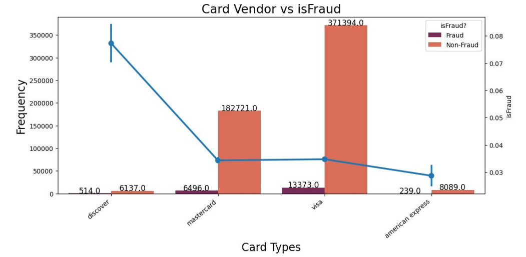


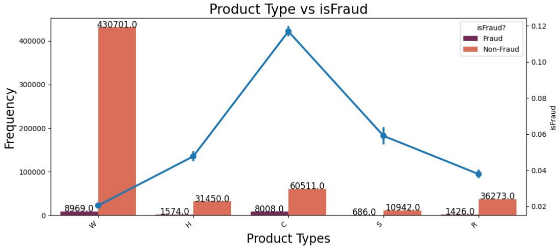
Initial Data Exploration

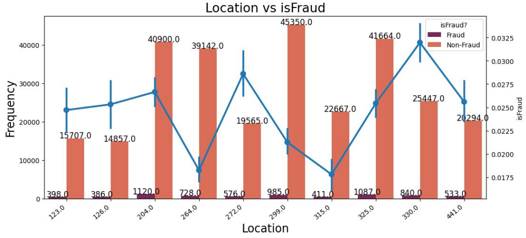


- Mostly non-fraudulent data
- Distribution of target variable:
    - Fraud: 3.5%
    - Non-fraud: 96.5%
- Categorical variable insights:
    - “Discover” vendor has the highest proportion of fraud followed by “Visa”.
    - “Credit” cards tend to have a higher chance of fraud than “Debit” cards.
    - The product type “C” has the highest proportion of frauds.
    - The location encoded with 330.0 has the highest proportion of frauds.





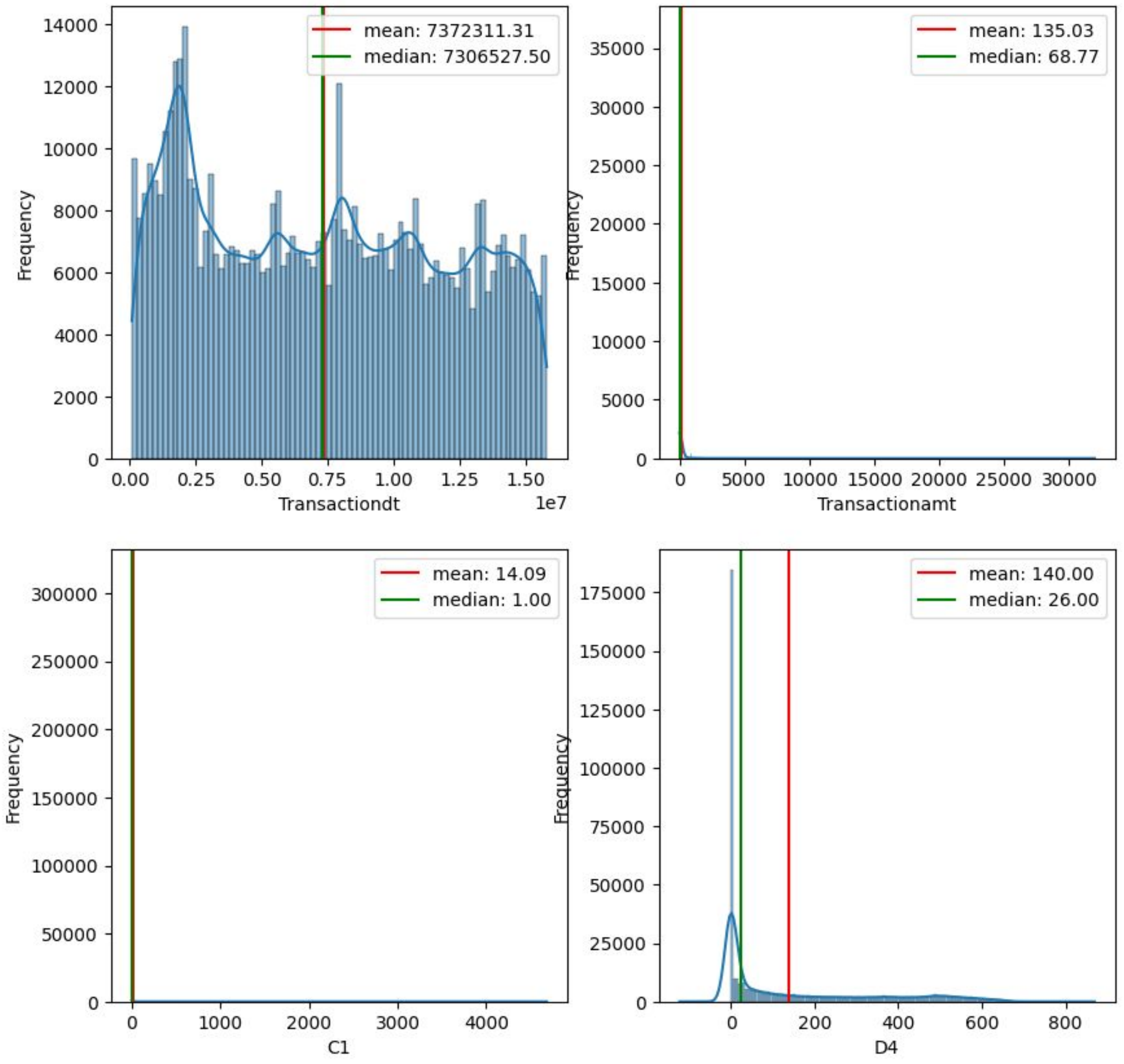




Bar plot - frequency (left y-axis)
Line plot - proportion of fraudulent transactions (right y-axis)

Central Tendency Plots

- Distribution and central tendencies for:
  - Transactiondt
  - Transactionamt
  - C1
  - D4
- Large difference between the mean and the median:
  - Suggests that the features are highly skewed (evident from the figures)



Data Cleaning, Sampling, and Preprocessing

- Some categorical features are encoded as float64
  - We convert them into int32 to save memory
- Filling Missing data
  - Dropping columns with more than 70% missing values
  - Numerical features: missing values replaced with the median
  - Categorical features: an extra “missing” category added
- Standardization
  - A separate dataset for softmax regression, feed forward neural network classifiers
- Resampling
  - SMOTE
  - Random Forest With Class Weighting

Dropping columns with too many missing values

missing value percentages before drop

```
id_24            99.196159
id_25            99.130965
id_08            99.127070
id_07            99.127070
id_21            99.126393
                   ...
C6                0.000000
C5                0.000000
C1                0.000000
C2                0.000000
TransactionID     0.000000
Length: 433, dtype: float64
```

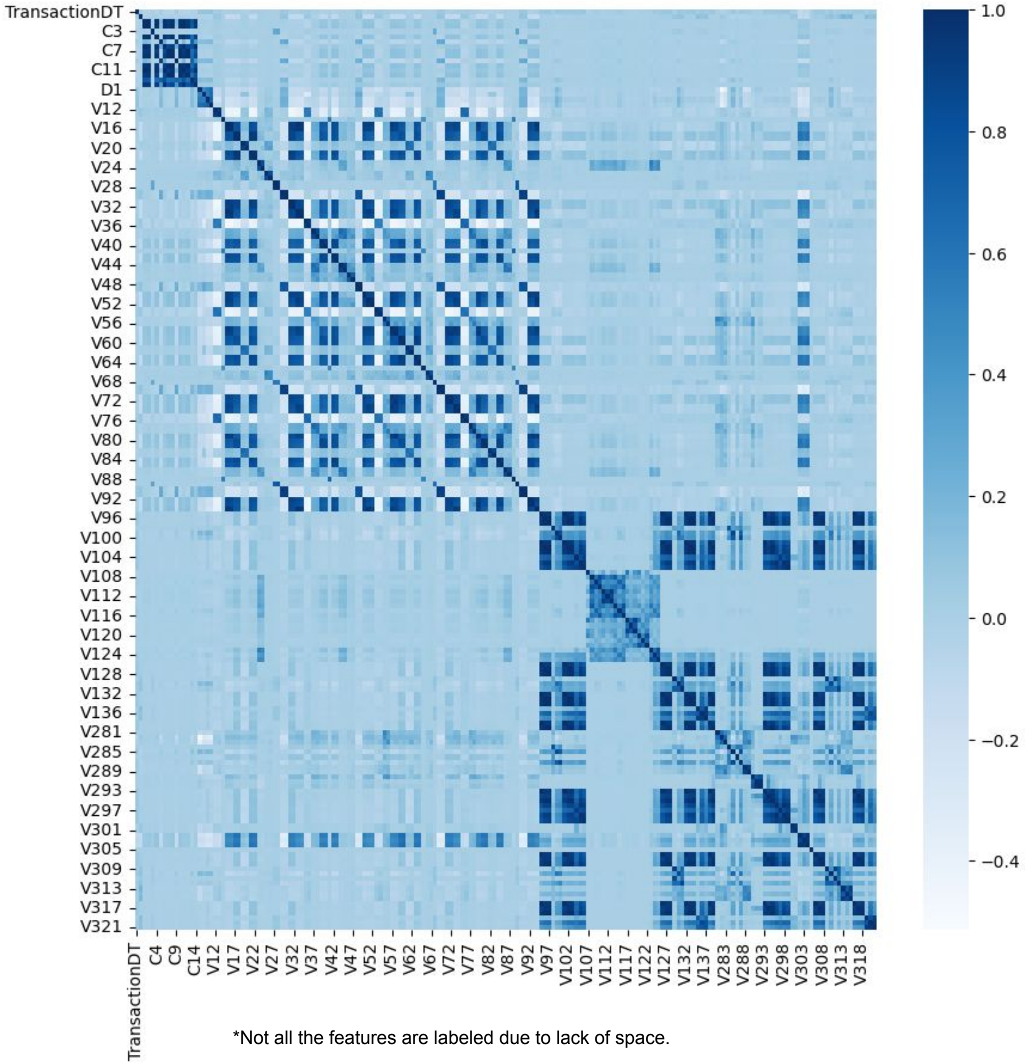
missing value percentages after drop

```
M6               28.678836
V43              28.612626
V51              28.612626
V50              28.612626
V49              28.612626
                   ...
C12               0.000000
C13               0.000000
C14               0.000000
TransactionDT     0.000000
TransactionID     0.000000
Length: 201, dtype: float64
```

about half columns were dropped

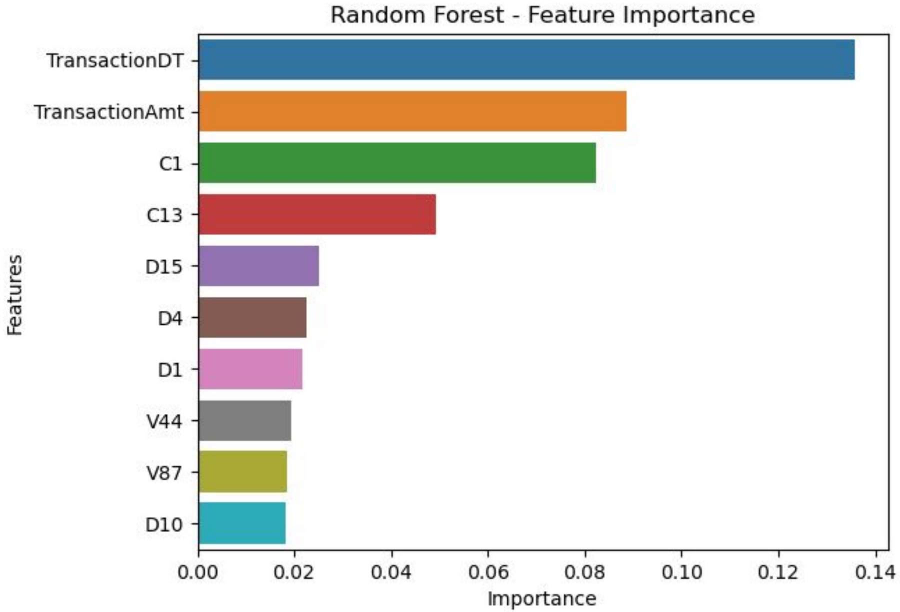
Correlation Matrix

- According to the correlation heatmap, we figured out some highly correlated features exist in the original dataset, which may cause:
  - overfitting
  - make it harder to draw meaningful insights from data
- Therefore, we dropped highly correlated features based on correlation threshold of 0.9



*Not all the features are labeled due to lack of space.

Feature Importance in Random Forest



Top-10 most important features

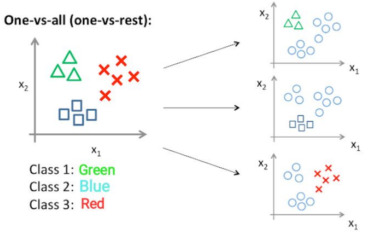
Why do we calculate feature importance scores?

- Reduce dimensionality:
  - Reducing the time to train the model without reducing the performance
- Better interpretation:
  - Identifying which features are most closely related to the target variable

Proposed ML techniques

Models:

- Baselines
  - One vs Rest logistic regression
  - Softmax classifier
  - K-NN classifier
- Tree based models
  - Random Forest
  - Histogram-based Gradient Boosting Classifier
  - Extreme Gradient Boosting (XGBoost)
- Neural Network
  - Feedforward Neural Network with Softmax activation



Metrics:

- Recall
  - special focus - we want to prevent fraud aggressively
- F1 score
- Area Under Receiver Operating Characteristic Curve
- Area Under Precision Recall Curve