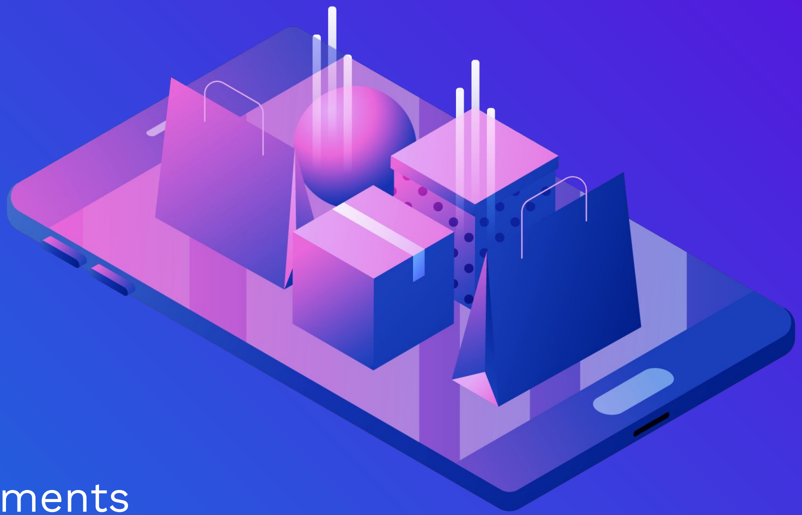
Credit Card Fraud Detection

Fight Against Financial Crime with Machine Learning

DSIF3 - 07 April 2022
Vincent Chua

Table of Contents



Background

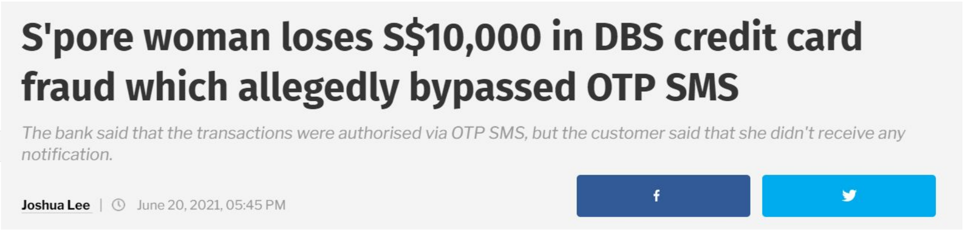
THE STRAITS TIMES SINGAPORE LOG IN SUBSCRIBE

Reports of unauthorised online banking and card transactions in Singapore jump 460% in 2020

Top Stories Latest News Discover Singapore Asia Commentary Sustainability CNA Insider Lifestyle Watch Listen + All Sections

Singapore

About S$500,000 stolen in fraudulent card payments involving diversion of SMS one-time passwords



S'pore woman loses S$10,000 in DBS credit card fraud which allegedly bypassed OTP SMS

The bank said that the transactions were authorised via OTP SMS, but the customer said that she didn't receive any notification.

Joshua Lee | June 20, 2021, 05:45 PM

Evolving of internet and popularity in using credit card for payments, **numbers of credit card fraud has been increased as compared to old days**. Although there is very low crime rate in Singapore, there is still fraud cases happens and it affect the credit card users as well as the bank. According to Straits Times, there were 1,848 police reports of transactions involving criminals phishing for banking and card details from victims - **up 462% from 2019's 329 cases**.

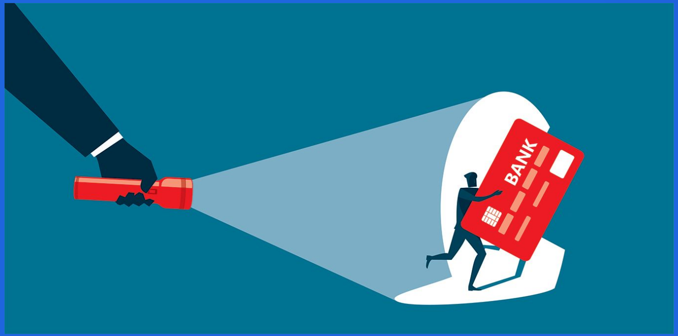
The purpose of fraud detection system is to **detect the anomaly credit card transactions and on halt the fraud transactions** on time while letting the normal transactions to be processed automatically.

With an effective fraud detection model in place, bank can **save huge losses from fraud transactions, gain credibility from users** and it also **enable the algorithms to identify and adapt to new anomaly pattern from fraudster.**

Problem Statement

To develop a fraud detection model aim to identify credit card fraud transactions via classification modeling. Model will be evaluated based on Recall Score and F1-Score.

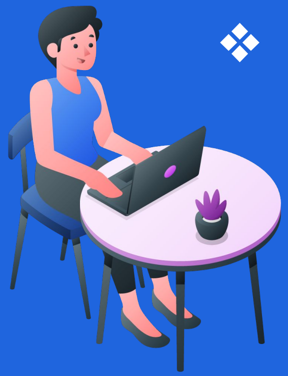
Target Audience: MAS Regulators / Risk and Compliance Department Heads and Managers



Datasets

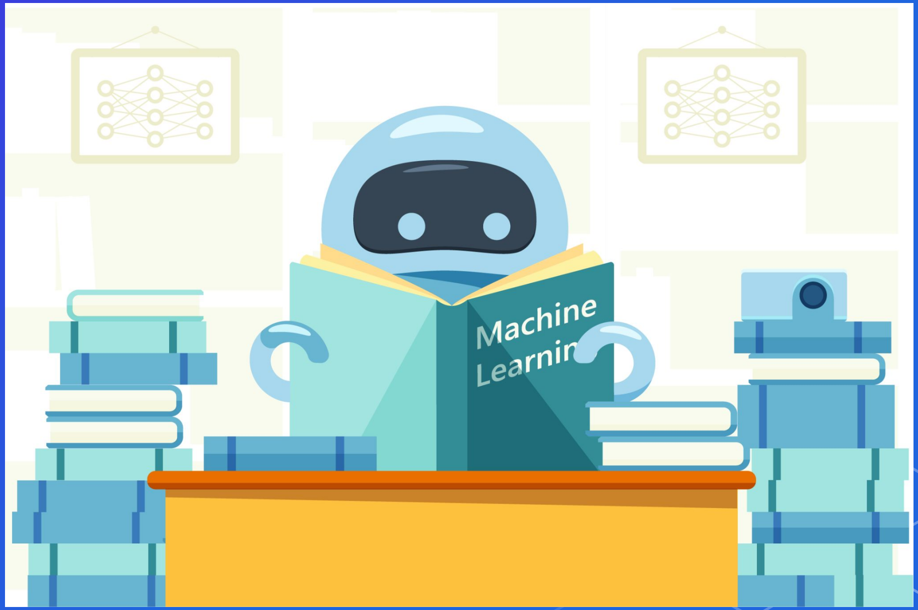
Credit Card Transactions Fraud Detection Dataset - Kaggle

- Train datasets of 1.3m rows
- Test datasets of 556k rows
- Transactions from 1st Jan 2019 - 31st Dec 2020
- 1,000 customers with a pool of 800 merchants



Modeling Approach

- ❑ Logistic Regression
- ❑ Gaussian Naive Bayes
- ❑ Random Forest Classifier
- ❑ XGB Classifier
- ❑ CatBoost Classifier
- ❑ LGBM Classifier

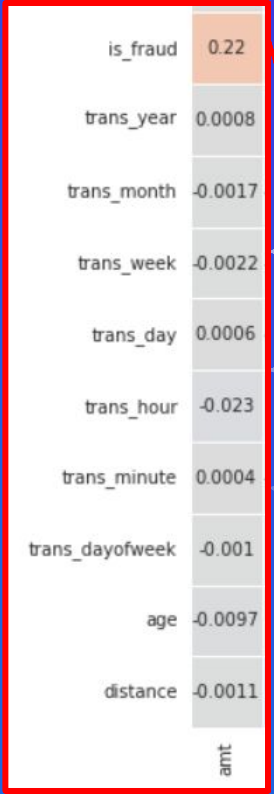


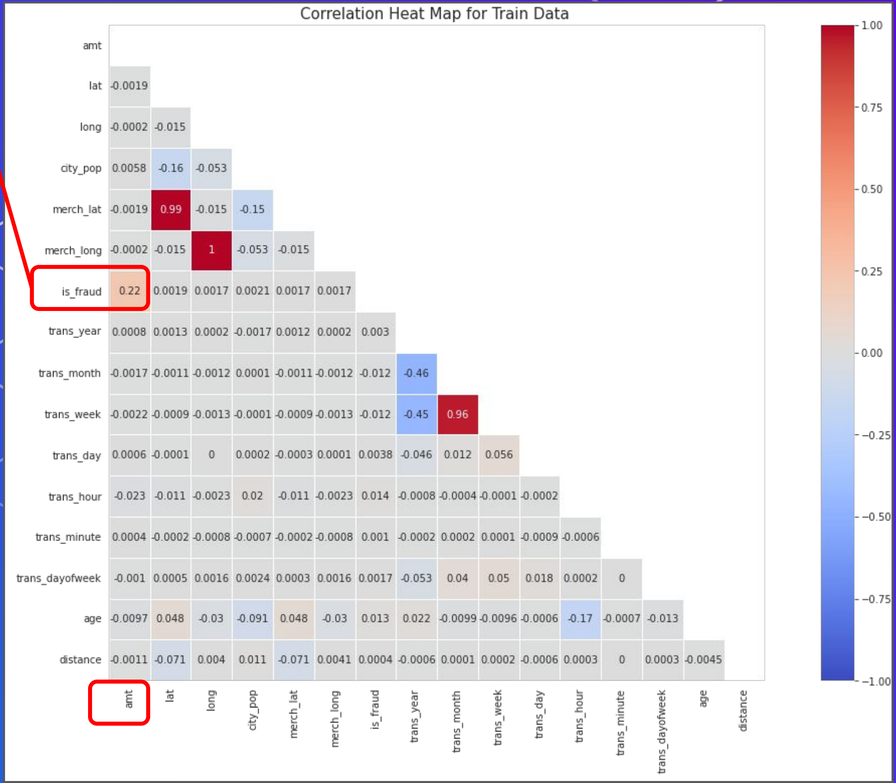
01

Exploratory Data Analysis (EDA)

Dataset - Target

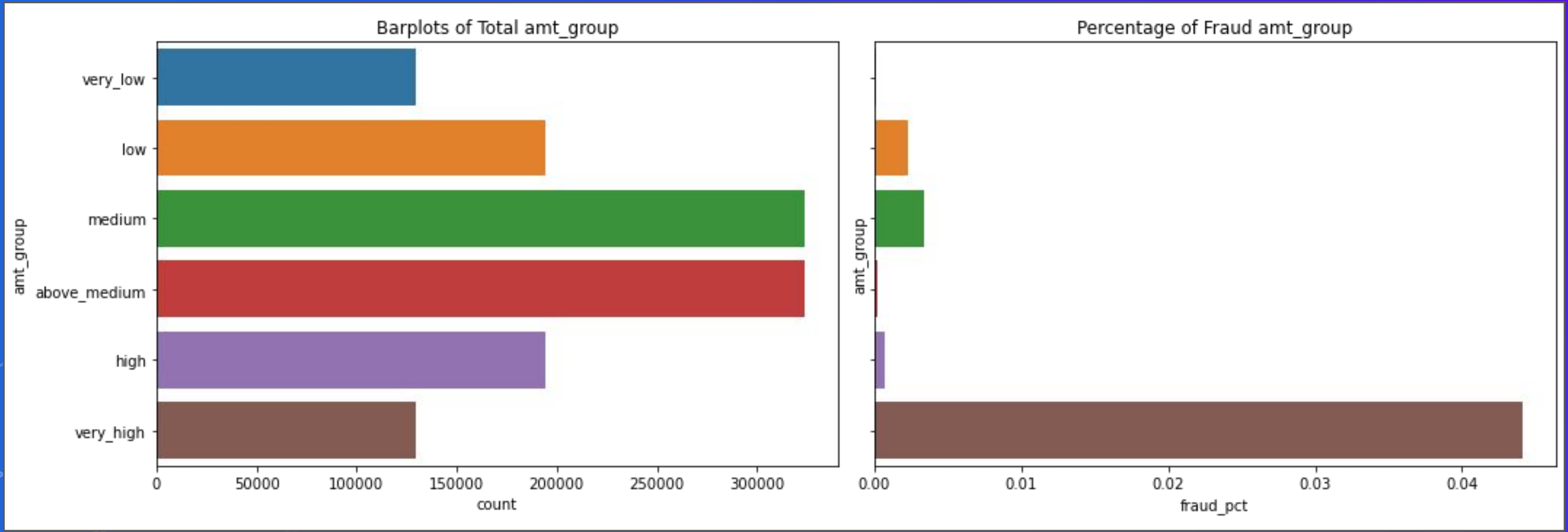
- Train datasets with **imbalance** target class, only **0.58% of transactions labeled as 'Fraud'**
- No numerical feature showing strong correlation against target (**is_fraud**)
- **Transaction Amount (amt)** is the only numerical feature have moderately positive correlation with **Pearson Correlation Coefficient of 0.22** against target





Transaction Amount Groups

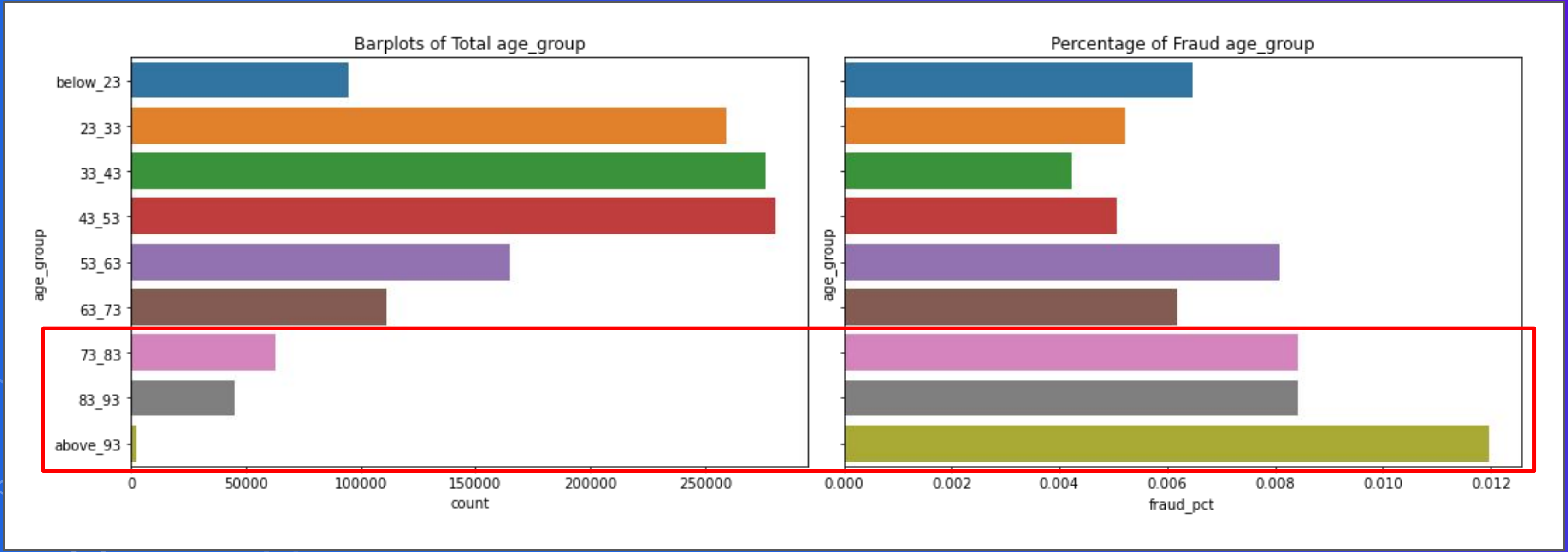
- **Very_high** amount group have the **less** transactions, it have the **highest** fraud rate
- Fraud rate of **very_high** is about **300 times higher** than **very_low** transaction amount group



- *Chi-Square p-value < 0.05 (amt_group & target)*
- *T-Test p-value < 0.05 (fraud rate for very_high & very_low)*

Age Groups

- **Above_93** age group have the less transactions, it have the **highest** fraud rate
- **Age about 73** have relatively higher fraud rate, the lowest fraud rate in age group between **33 to 43**

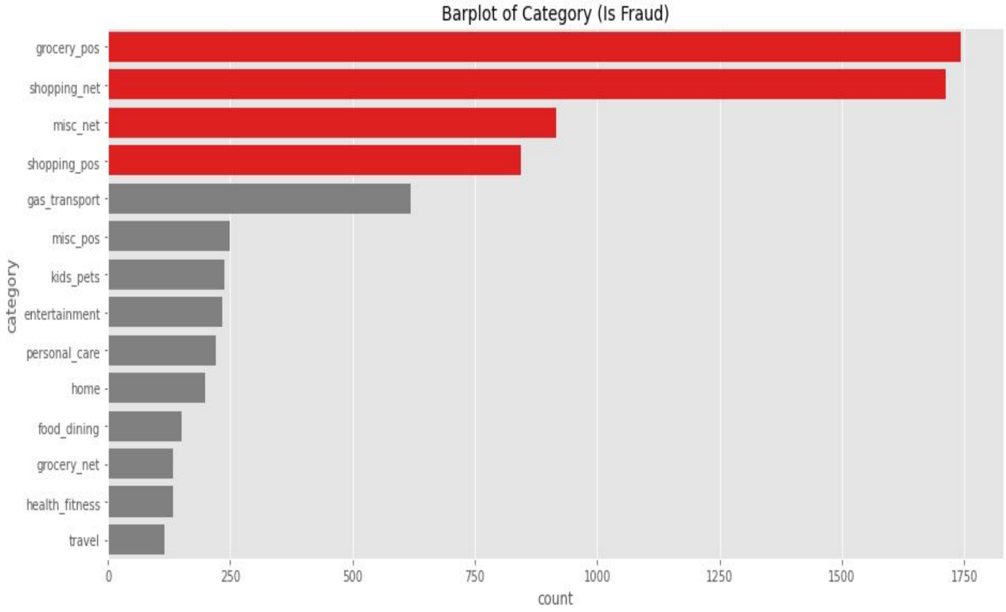


- ❖ *Chi-Square p-value < 0.05 (amt_group & target)*
- ❖ *T-Test p-value < 0.05 (fraud rate for very_high & very_low)*

Category

- **2 out of 3 net transactions** (Shopping & Misc) are the **Top 2 fraud rate** categories
- **Grocery_pos & Shopping_pos** are the **Top 4 fraud transactions count and fraud rate** categories

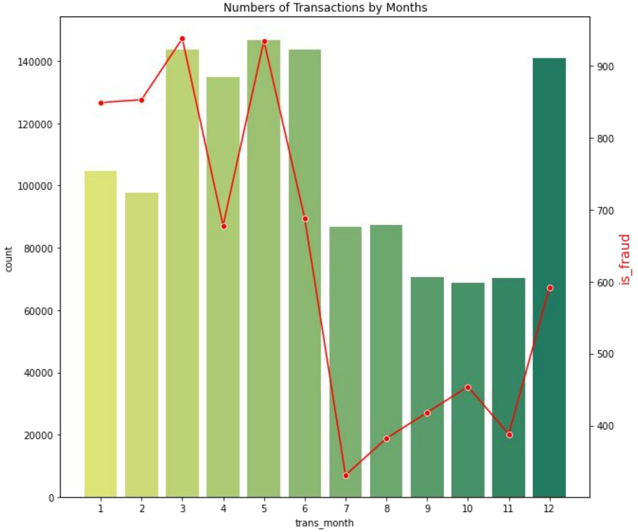
❖ *Chi-Square p-value < 0.05 (category & target)*

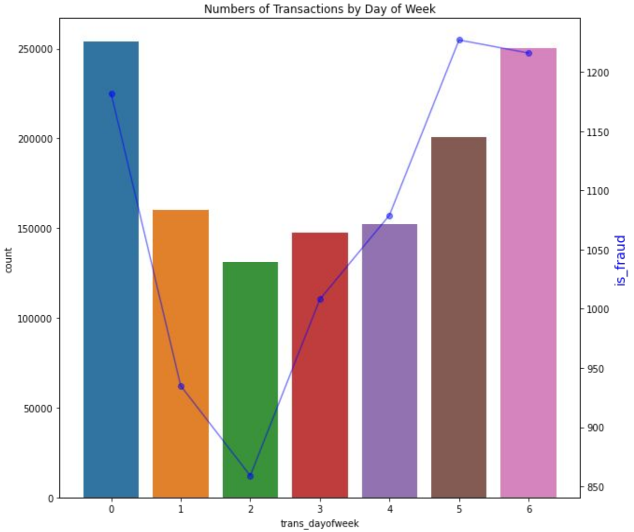


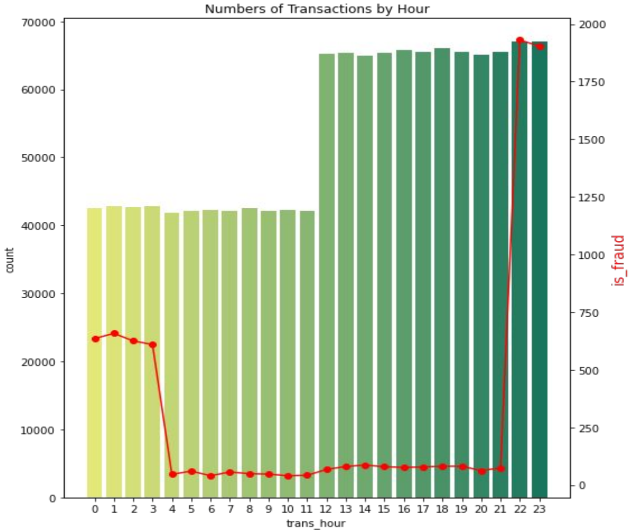
is_fraud / category	0	1	fraud_pct
shopping_net	95830	1713	0.017561
misc_net	62372	915	0.014458
grocery_pos	121895	1743	0.014098
shopping_pos	115829	843	0.007225
gas_transport	131041	618	0.004694
misc_pos	79405	250	0.003139
grocery_net	45318	134	0.002948
travel	40391	116	0.002864
entertainment	93781	233	0.002478
personal_care	90538	220	0.002424
kids_pets	112796	239	0.002114
food_dining	91310	151	0.001651
home	122917	198	0.001608
health_fitness	85746	133	0.001549

	is_fraud / category	0	1
amt	entertainment	63.118913	503.535579
	food_dining	50.972816	120.076954
	gas_transport	63.675782	12.288204
	grocery_net	53.792817	12.162836
	grocery_pos	114.172234	311.989616
	health_fitness	54.234775	20.248421
	home	57.949319	257.432626
	kids_pets	57.620386	18.121674
	misc_net	70.359176	797.012852
	misc_pos	62.401757	218.284080
	personal_care	48.020643	26.170545
	shopping_net	72.142606	999.254939
	shopping_pos	73.977575	876.921815
	travel	111.798920	9.064569

Transaction Month, Day of Week, Hour



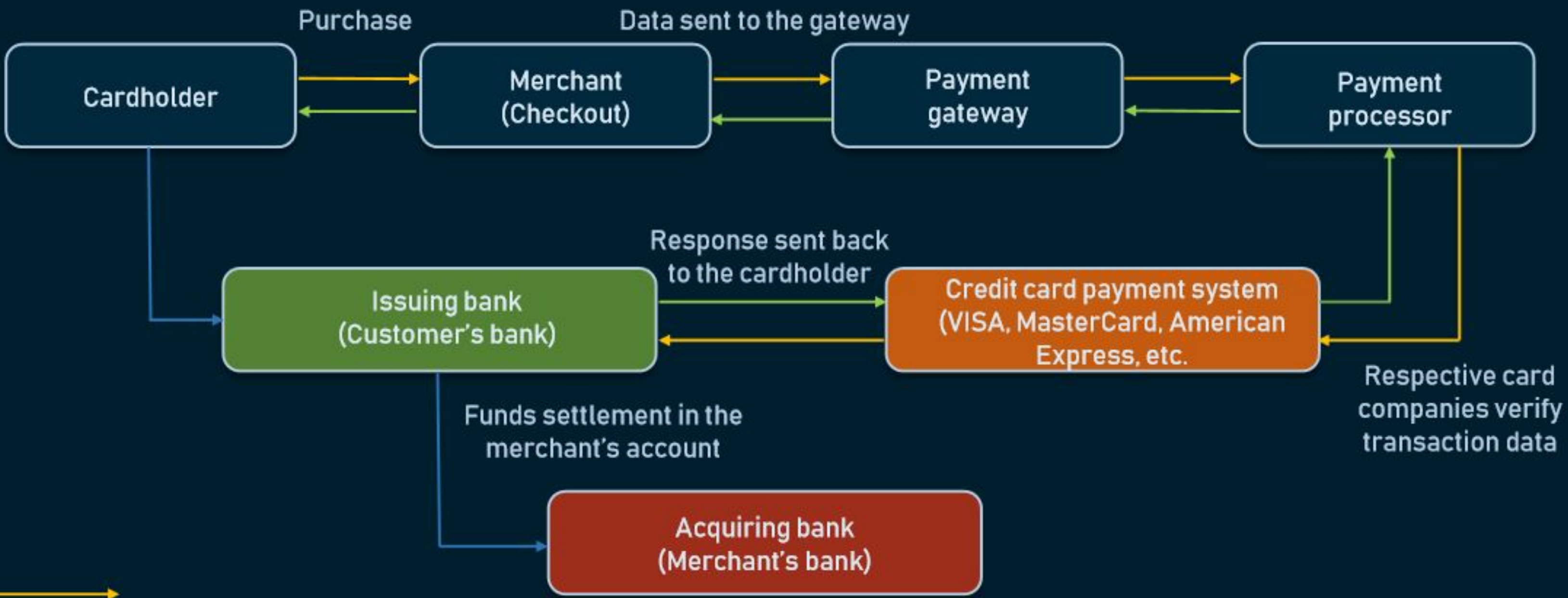




- **Month 3 and 5** have the highest fraud rate, while **month 7** has the lowest fraud rate
- **Saturday and Sunday** have the highest number of transactions, **Friday** have the highest fraud rate, **Tuesday** have both lowest number of transactions and fraud rate
- Higher **fraud rate** happens **from 10pm and started reducing at 3am**

02
Feature Engineering & RFM Analysis

HOW PAYMENT PROCESSING WORKS



*Orange arrow indicates card data verification flow

*Green arrow indicates the response from banks and a credit card associations returned to a cardholder

*Blue arrow indicates funds settlement in the acquiring bank

Image Source: https://www.altexsoft.com/blog/credit-card-fraud-detection/

altexsoft
software r&d engineering

Feature Engineering

Cardholder Level

- **Cumulative and Average** Transaction Amount
- **Difference** in Transaction <u>Amount</u> from Last Transaction
- **Difference** in Transaction <u>Datetime</u> from Last Transaction
- **Difference** in Merchant <u>Distance</u> from Last Transaction

Cardholder Spending Behavior

- Number of Transactions
- Average Transaction Amount
- Minimum Transaction Amount
- Maximum Transaction Amount

1. Last 5 Minutes
2. Last 1 Hour
3. Last 24 Hours
4. Last 7 Days
5. Last 30 Days

Merchant Transaction Behavior

- Number of Transactions
- Average Transaction Amount
- Minimum Transaction Amount
- Maximum Transaction Amount

1. Last 24 Hours
2. Last 7 Days
3. Last 14 Days
4. Last 30 Days

Feature Engineering - Others

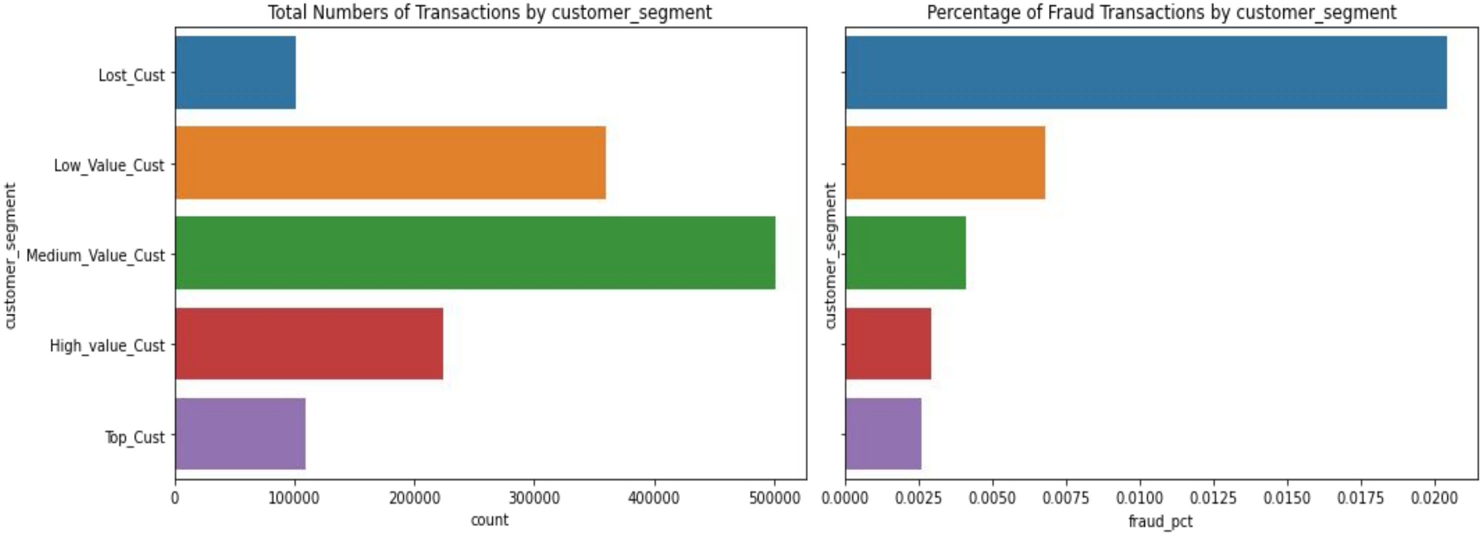
Internet Transaction

Clustering Long, Lat & Amount (reduce from 970 to 44) via **DBSCAN**

Customer Segmentation via **RFM Analysis**

RFM Analysis

- Identify **Recency Frequency & Monetary** to segmentize into different credit cardholders segmentations
- **4%** Top_Cust, **9%** High_value_cust, **27%** Medium_Value_Cust, **34%** Low_Value_Cust, **26%** Lost_Cust
- **Lost_Cust** segment have about **10 times higher** fraud rate than **Top_Cust** segment



- ❖ *Chi-Square p-value < 0.05 (customer_segment & target)*
- ❖ *T-Test p-value < 0.05 (top_cust & lost_cust)*

03

Modeling & Evaluation

Modeling Process

Preprocessing - Remove multicollinearity, Dummify categorical variables, Train-Test Split

Model Selection - Baseline model: Logistic Regression

Hyperparameters Tuning with AutoML - PyCaret

Model Evaluation

Final Model & Fraud Detection

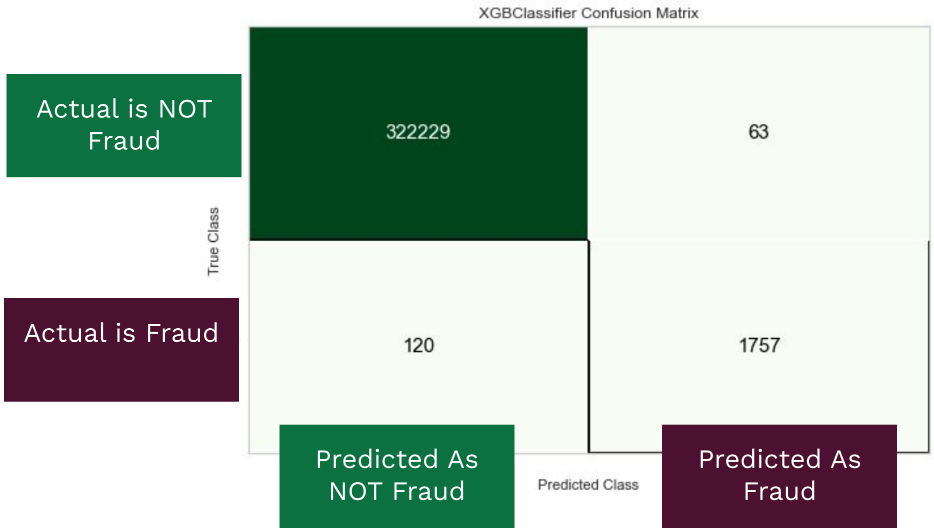
Model Selection

model	train accuracy	test accuracy	precision	recall	average precision	f1_score	roc_auc
CatBoostClassifier	0.999987	0.999429	0.983982	0.916356	0.902162	0.948966	0.999736
XGBClassifier	0.999969	0.999374	0.982699	0.907832	0.892659	0.943783	0.999546
Random Forest Classifier	1.000000	0.998760	0.985517	0.797549	0.787170	0.881625	0.995251
Logistic Regression	0.997269	0.997082	0.887594	0.567928	0.506591	0.692658	0.979961
LGBMClassifier	0.987712	0.986825	0.281170	0.819393	0.231434	0.418674	0.947420
Gaussian Naive Bayes	0.979104	0.978863	0.181049	0.752264	0.137631	0.291856	0.947807

- Maximize **Recall Score** to detect as many Fraud transactions as possible
- Then optimize **F1-Score** to get a balance with **Precision Score**, it is to minimize Type I (False Positive) Errors.
- **CatBoost & XGBoost** are the only models with all **3 metrics above 90%**

Hyperparameters Tuning

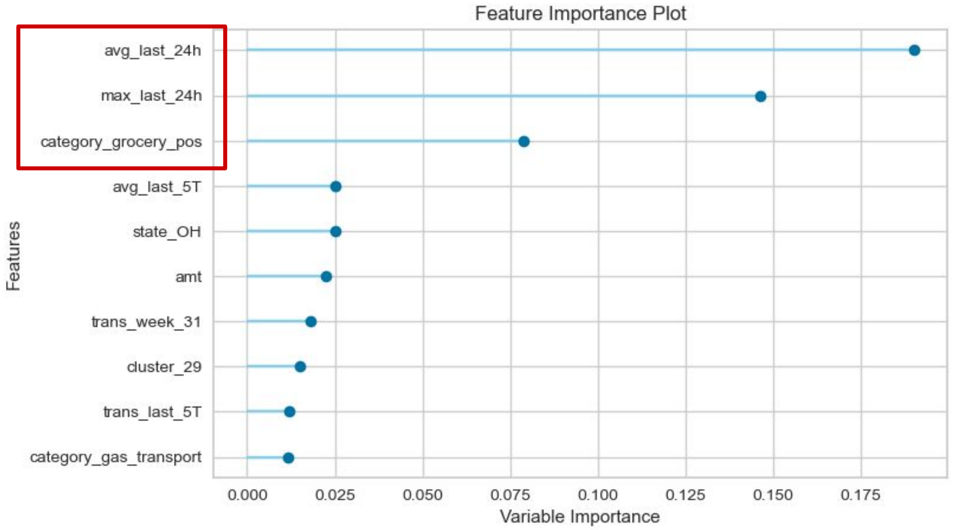
	Accuracy	Precision	Recall	F1
CatBoost	0.9994	0.984	0.9163	0.949
XGB	0.9994	0.9827	0.9078	0.9438
XGB (tuned)	0.9994	0.9561 ↓	0.94 ↑	0.9479

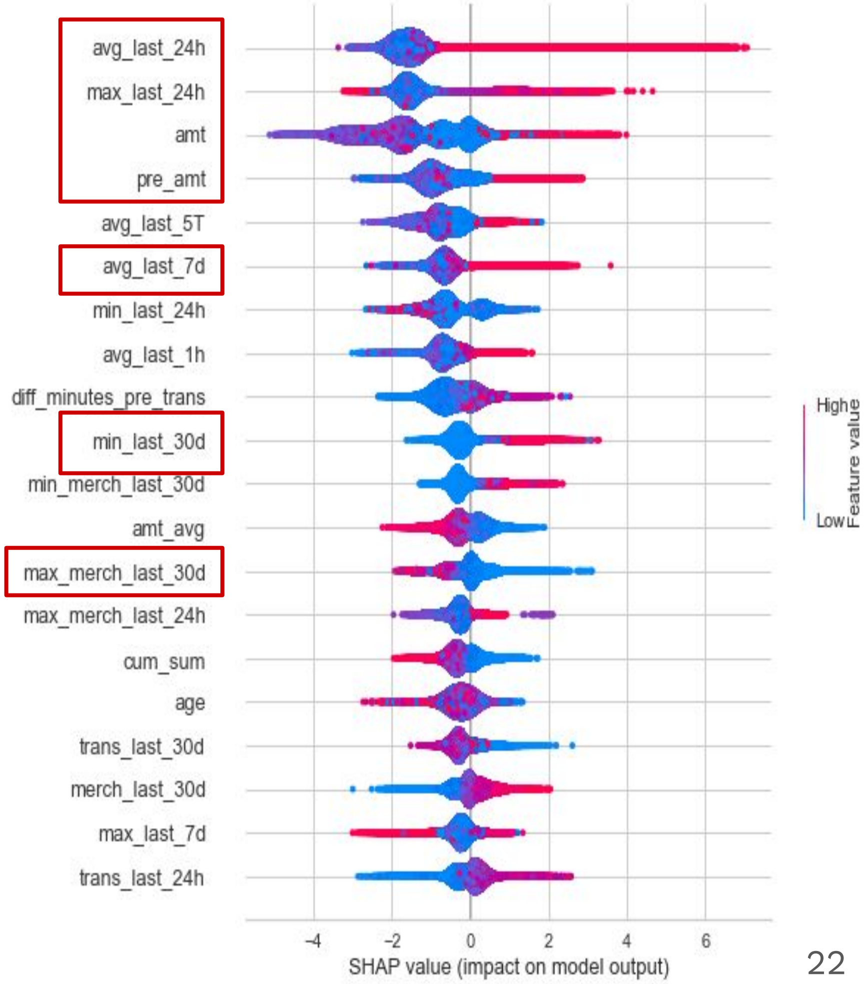


- Perform Hyperparameters tuning using **AutoML - PyCaret**
- **XGB** turned out to be the best model after tuned with 20 iterations
- Although Precision Score and F1-Score reduce a bit, Recall Score has a huge improvement
- About **3% Type I** (False Positive) Errors, which predicted as fraud but those are actual non-fraud transactions
- About **6% Type II** (False Negative) Erros, which predicted as non-fraud but it those are actual fraud transactions

Top Predictors

- Most important features are **average and maximum transactions amount in last 24 hour**, and **Grocery_pos Category**
- Features that have strong positive impact on target are **avg_last_24h**, **max_last_24h**, **amt**, **pre_amt**, **avg_last_7d**, **min_last_30d**, **max_merch_last_30d**





Final Model - Test Dataset (Unseen)

Recall Score: **90.72%** F1 Score: **91.58%** Precision Score: **92.45%**

94%

Reducing in Fraud Transactions Amount

- Stopping total **$ 1.07 million** of fraud transaction amount from went through

90%

Reducing in Numbers of Fraud Transactions

- Detecting **1946** fraud transactions out of **2145** actual fraud transactions

92%

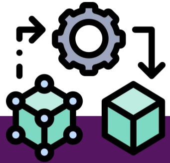
Reducing in False Alarm Triggered

- **159** False Alarm (Fraud Alert) Triggered to the cardholders instead of **2105**

04

Recommendations & Future Improvements

Recommendations



Effective Fraud Detection Model

- **Improve the efficiency and effectiveness on fraud detection** by Risk and Compliance Team
- Reduce unnecessary false alarm triggered to the credit cardholders
- Allow bank **save around 94% of losses** from fraud transactions

Top Predictors

- **Identify Top Predictors and Features with strong impact on target** for the team to implement action plan strategically to fight against financial crime

App Verification

- The team can implement the additional authentication via phone app / sms to **reduce the negative implications and user experience** with the false alarm triggered
- Avoid transactions on halt unnecessarily

Future Improvements

Features Engineering	Explore other possibilities to engineer new features eg. demographically & geographically (jobs, income per capita by states)
Deep Learning Solutions	Explore Deep Learning Solutions and aim to achieve target recall score at minimal 95%
Time-Dependent Graph	Identify and capture the potential anomaly and fraud pattern
Real-Time Alert	Deploy the model for real-time alert and detection

Tableau: Credit Card Fraud Monitoring

Check out my Tableau Dashboard here!

https://tabsoft.co/376Gr5a

THANKS!

Do you have any questions?

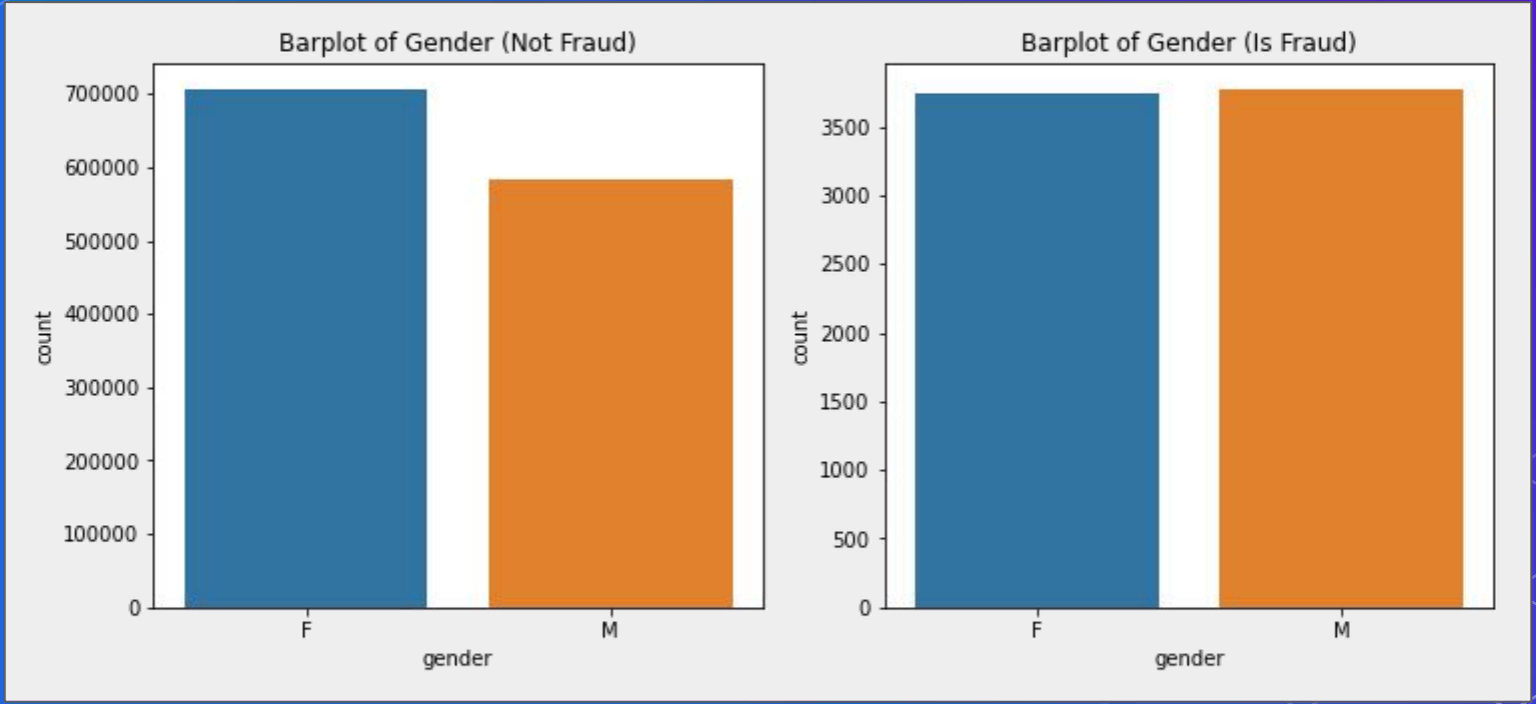
chua_vincent@live.com

/in/vincentchua1989/

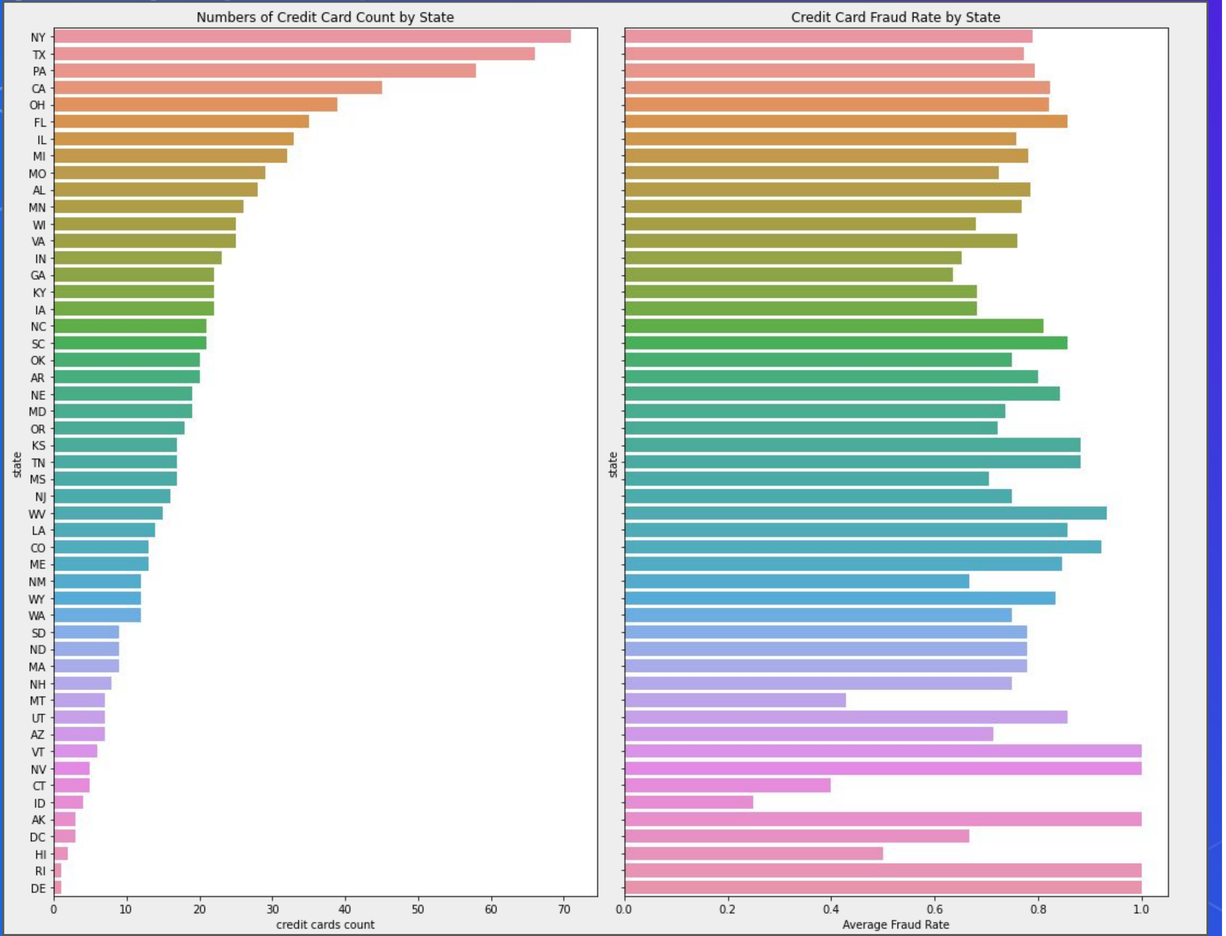
APPENDIX 1 – RAW DATASET FEATURES

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1296675 entries, 0 to 1296674
Data columns (total 23 columns):
 #   Column                 Non-Null Count    Dtype
---  ------                 --------------    -----
 0   Unnamed: 0             1296675 non-null  int64
 1   trans_date_trans_time  1296675 non-null  object
 2   cc_num                 1296675 non-null  int64
 3   merchant               1296675 non-null  object
 4   category               1296675 non-null  object
 5   amt                    1296675 non-null  float64
 6   first                  1296675 non-null  object
 7   last                   1296675 non-null  object
 8   gender                 1296675 non-null  object
 9   street                 1296675 non-null  object
 10  city                   1296675 non-null  object
 11  state                  1296675 non-null  object
 12  zip                    1296675 non-null  int64
 13  lat                    1296675 non-null  float64
 14  long                   1296675 non-null  float64
 15  city_pop               1296675 non-null  int64
 16  job                    1296675 non-null  object
 17  dob                    1296675 non-null  object
 18  trans_num              1296675 non-null  object
 19  unix_time              1296675 non-null  int64
 20  merch_lat              1296675 non-null  float64
 21  merch_long             1296675 non-null  float64
 22  is_fraud               1296675 non-null  int64
dtypes: float64(5), int64(6), object(12)
memory usage: 227.5+ MB
```

APPENDIX 2 – GENDER BAR PLOT

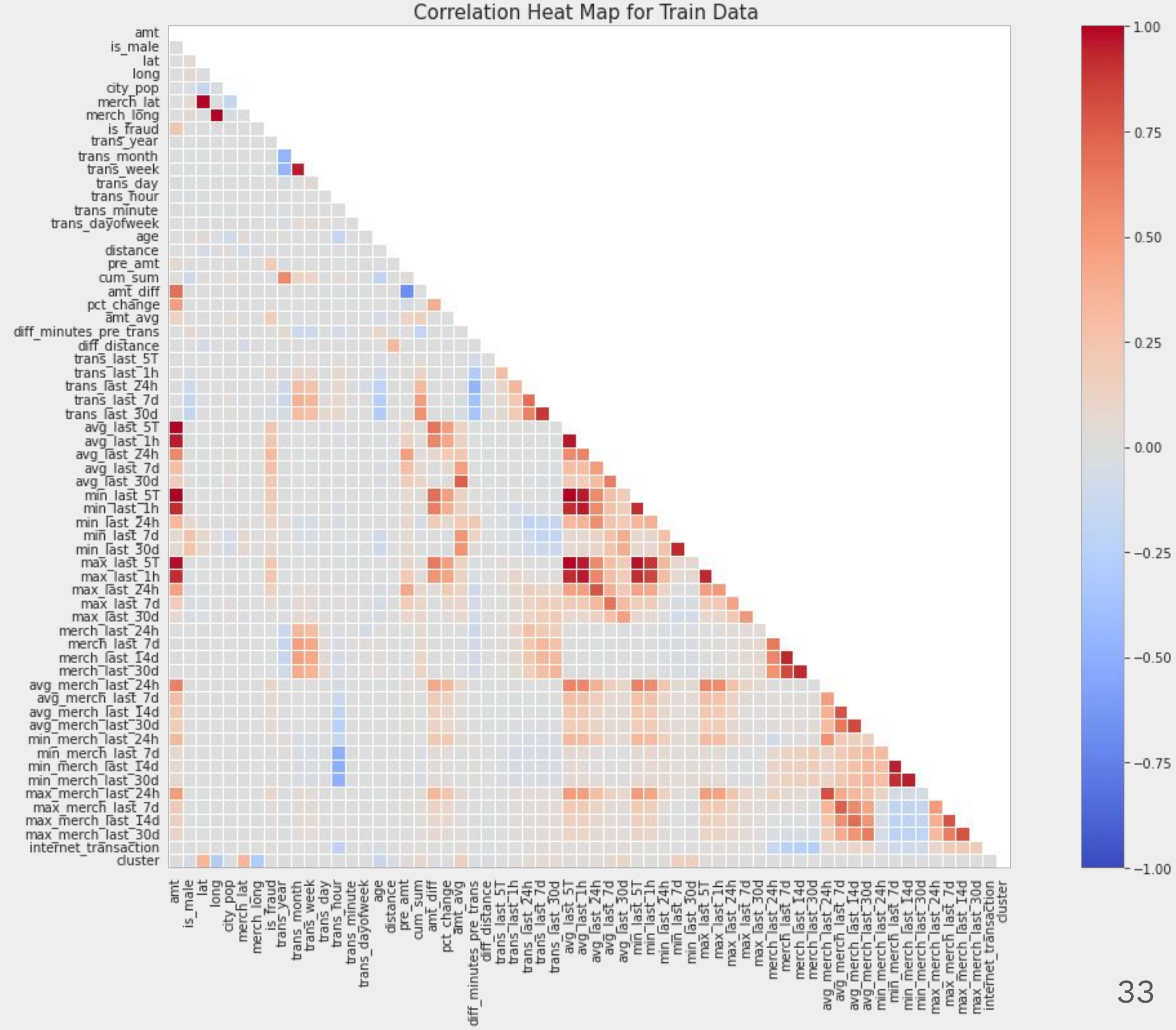


APPENDIX 3 – STATE BAR PLOT



is_fraud	0	1	fraud_pct
state			
DE	0.0	9.0	1.000000
RI	535.0	15.0	0.027273
AK	2084.0	36.0	0.016981
NV	5560.0	47.0	0.008382
CO	13767.0	113.0	0.008141
OR	18448.0	149.0	0.008012
TN	17414.0	140.0	0.007975
NE	23988.0	180.0	0.007448
ME	16386.0	119.0	0.007210
NH	8219.0	59.0	0.007127
OH	46159.0	321.0	0.006906
KS	22840.0	156.0	0.006784
VA	29052.0	198.0	0.006769
NY	82946.0	555.0	0.006647
SC	28997.0	193.0	0.006612
FL	42390.0	281.0	0.006585
MN	31507.0	207.0	0.006527
VT	11696.0	72.0	0.006118

APPENDIX 4 – HEATMAP



References

- **Icons:** https://www.flaticon.com/
- **Slides Template:** https://slidesgo.com/
- **Dataset:** https://www.kaggle.com/datasets/kartik2112/fraud-detection