Credit Card Fraud detection - Capstone Project

Team Members Page

Vivek SD

Senior Manager,
Schneider Electric

Yash Khatavkar

Data Analyst
Blue Star

Vikas Bhartiya

Database
Manager

Introduction

- Problem statement thro’ 5W-How Analysis
- Dataset(Highly Imbalanced)
- Exploratory Data Analysis
- Model Building and Evaluation
- Testing the best model
- Cost Benefit Analysis
- Recommendation

Problem Statement

Who

Who is involved in this process?

The fraudster attempting to steal the money, the customer whose credit card information is being used without his/her knowledge, and the credit card company responsible for detecting and preventing such fraudulent transactions.

What

What do they do with it?

Fraudster tries to steal money while the customers are unaware of such fraudulent activities made using their credit card.

Where

Where do the transactions happen?

Fraudulent transactions can happen anywhere credit cards are accepted for payment be it in-person at a physical store or online.

When

When does it happen?

Fraudulent transactions can happen anytime but they occur more frequently during online transactions and/or during holiday seasons

Why

Why do credit card fraud transaction occur?

Fraudsters seek to gain unauthorized access to funds, goods or services without being detected.

How

How business is affected?

Can lead to financial losses for credit card companies and customers, damage to their reputation, and increased costs for implementing fraud detection measures.

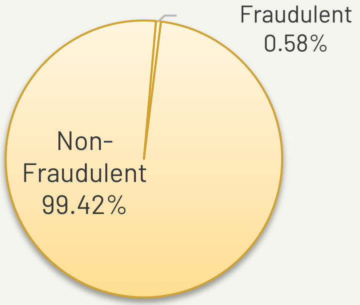
Dataset and Data imbalance

DATASET

Train & Test dataset provided to build and come up with the best model for Credit Card fraud detection.

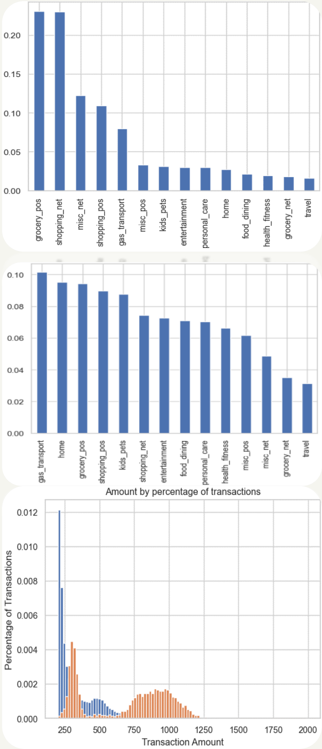
	Rows	Columns
Train dataset	1296675	23
Test dataset	555719	23

DATA IMBALANCE

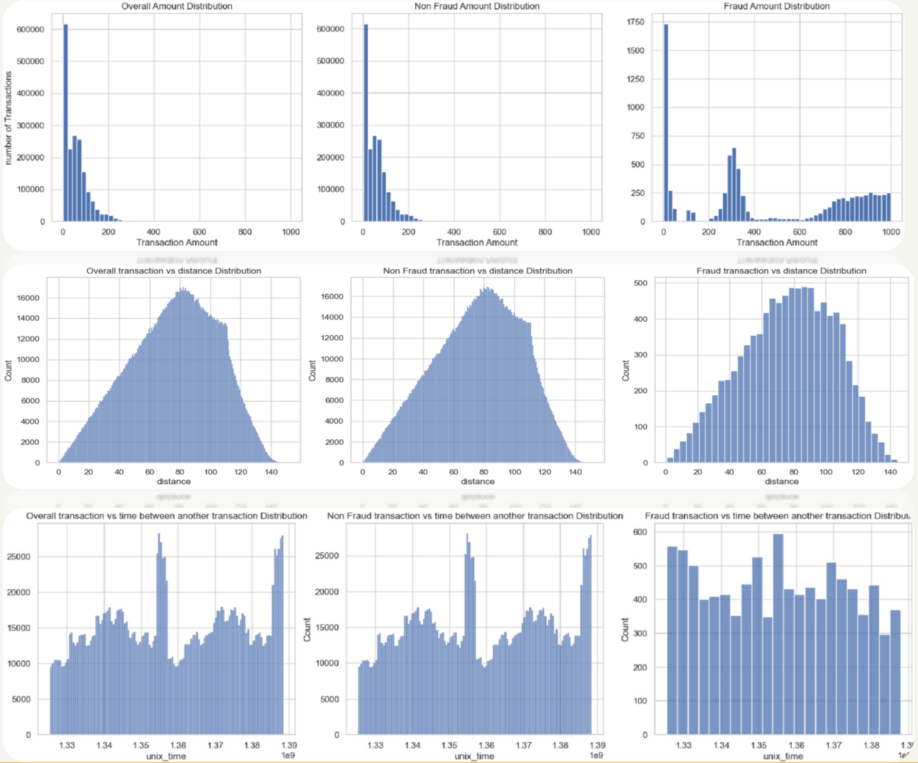


Exploratory Data Analysis

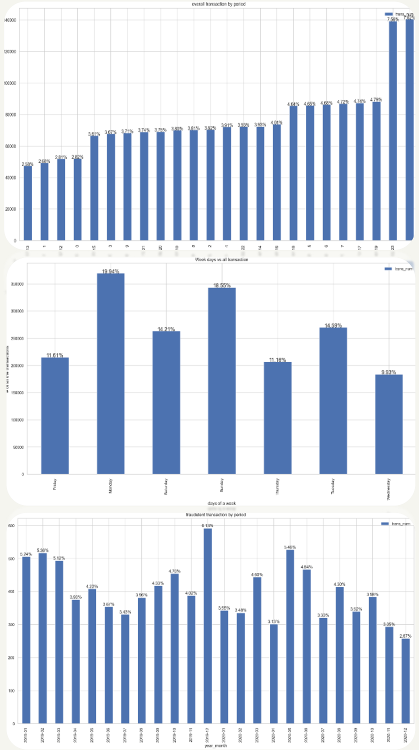
Category wise comparison



Fraudulent Vs Non-Fraudulent



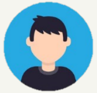
Fraudulent transactions by period



Exploratory Data Analysis

GAS _ TRANSPORT have the **HIGHEST** number of transactions 

TRAVEL the **LEAST** number of transactions

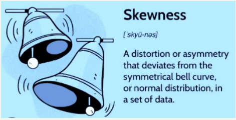
 made more number of transactions than 

MORE number of transactions at **NIGHT** 

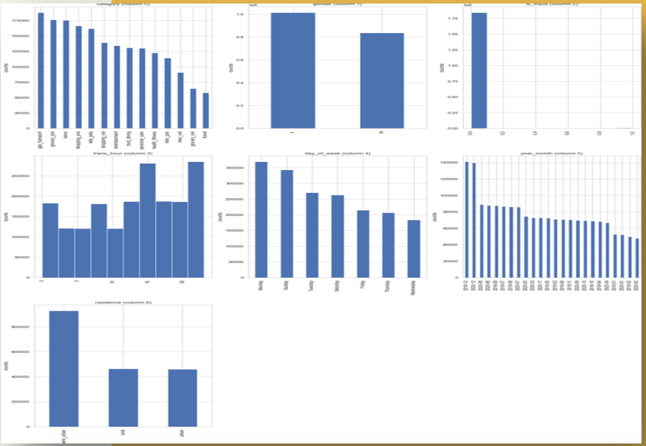
MORE on Sunday and MONDAY and the **LEAST on Wednesday**

MORE number of transactions observed towards **YEAR END** 

Dataset is **HIGHLY SKEWED****



Lastly, **NO SIGNIFICANT CORRELATION** observed between variables



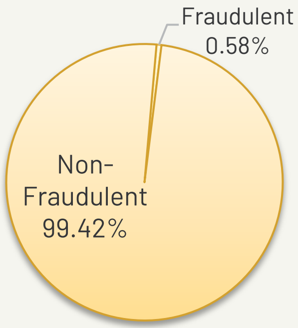
Model Building and Evaluation

	Model Name	Training Score	Testing Score	Accuracy	F1 Score	Precision	Recall
0	Logistic Regression - without balancing	0.993720	0.995609	0.995609	0.993948	0.000000	0.000000
1	Logistic Regression - Random Under Sampling	0.830796	0.833259	0.925450	0.957731	0.037356	0.739394
2	Logistic Regression - Random Over Sampling	0.829232	0.833809	0.927084	0.958615	0.037750	0.730536
3	Logistic Regression - SMOTE	0.827151	0.834617	0.928095	0.959163	0.038336	0.731935
4	Decision Tree - Random Under Sampling	0.980301	0.967362	0.958567	0.975576	0.082867	0.966900
5	Decision Tree - Random Over Sampling	0.986404	0.955598	0.956868	0.974671	0.080146	0.971096
6	Decision Tree - SMOTE	0.989901	0.962845	0.954709	0.973514	0.075860	0.959907
7	Random Forest - Random Under Sampling	1.000000	0.975799	0.977503	0.985723	0.142492	0.962238
8	Random Forest - Random Over Sampling	1.000000	0.968918	0.975673	0.984734	0.133570	0.966434
9	Random Forest - SMOTE	1.000000	0.967377	0.973625	0.983624	0.123767	0.959441

RANDOM FOREST - yields the BEST RESULT amongst the models tested

Model Building and Evaluation

Data Imbalance posed difficulty in training the models as it resulted in overfitting.



Logistic regression – Precision and recall are almost same when applying random under sampling ,

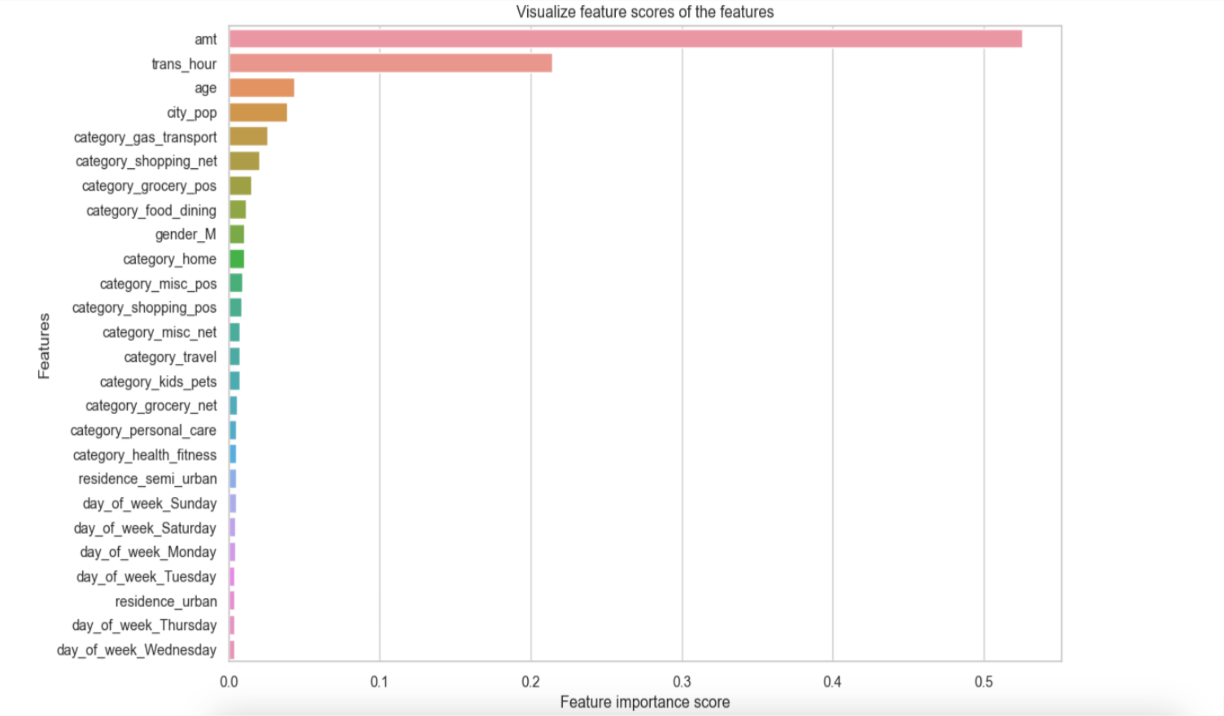
Oversampling and SMOTE but better than applying without any sampling.

Decision tree- preformed better than the logistic regression using all 3 method.

Random forest performs better than all the other models

RANDOM FOREST - yields the BEST RESULT amongst the models tested

Features Importance




```
amt                         0.525422
trans_hour                  0.214341
age                         0.043452
city_pop                    0.038603
category_gas_transport      0.025702
category_shopping_net       0.020065
category_grocery_pos        0.014919
category_food_dining        0.011062
gender_M                    0.010137
category_home               0.010059
category_misc_pos           0.009227
category_shopping_pos       0.008335
category_misc_net           0.007459
category_travel             0.007114
category_kids_pets          0.007009
category_grocery_net        0.005645
category_personal_care      0.005004
category_health_fitness     0.004824
residence_semi_urban        0.004802
day_of_week_Sunday          0.004687
day_of_week_Saturday        0.003976
day_of_week_Monday          0.003956
day_of_week_Tuesday         0.003749
residence_urban             0.003527
day_of_week_Thursday        0.003490
day_of_week_Wednesday       0.003435
dtype: float64
```

Amount , transaction hour and age are the top features contributing in model.

Testing the Best Model

	precision	recall	f1-score	support
0	1.00	0.97	0.99	1289169
1	0.18	0.98	0.30	7506
accuracy			0.97	1296675
macro avg	0.59	0.98	0.64	1296675
weighted avg	1.00	0.97	0.98	1296675

	precision	recall	f1-score	support
0	1.00	0.97	0.99	553574
1	0.12	0.96	0.22	2145
accuracy			0.97	555719
macro avg	0.56	0.97	0.60	555719
weighted avg	1.00	0.97	0.98	555719

PRECISION and RECALL scores are **HIGH**

COST BENEFIT ANALYSIS to be done to identify the affordability of the model

Cost Benefit Analysis

Cost Benefit Analysis		
S. No	Questions	Answer
a	Average number of transactions per month	77,183
b	Average number of fraudulent transaction per month	402
c	Average amount per fraud transaction	531

S. No	Questions	Answer
1	Cost incurred per month before the model was deployed (b*c)	2,13,392.22
2	Average number of transactions per month detected as fraudulent by the model (TF)	2400
3	Cost of providing customer executive support per fraudulent transaction detected by the model	1.5
4	Total cost of providing customer support per month for fraudulent transactions detected by the model (TF*$1.5)	8778
5	Average number of transactions per month that are fraudulent but not detected by the model (FN)	10
6	Cost incurred due to fraudulent transactions left undetected by the model (FN*c)	12,205.21
7	Cost incurred per month after the model is built and deployed (4+6)	$20,983
8	Final savings = Cost incurred before - Cost incurred after(1-7)	$1,92,409

$1,92,409

Savings due to our model

Thank you!