Credit Card Fraud Detection

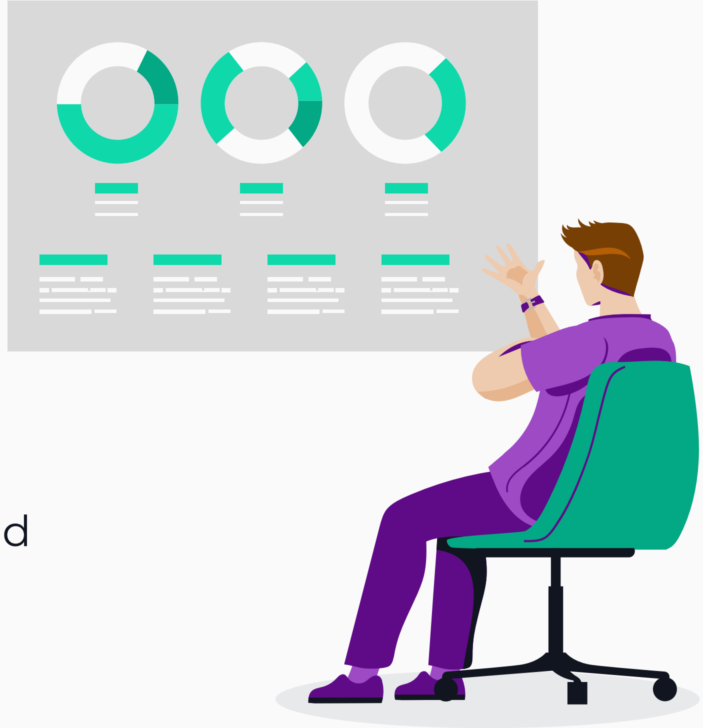
Ahmad Aldhaheri
Devin Thakker
Yash Gollapudi

Table of contents

01

Background

What is the importance and history of fraud detection?



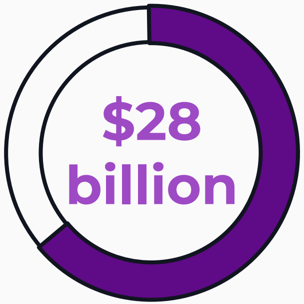
$28.7 Trillion

Total value of credit card
transactions as of 2019

Fraud VS Transactions

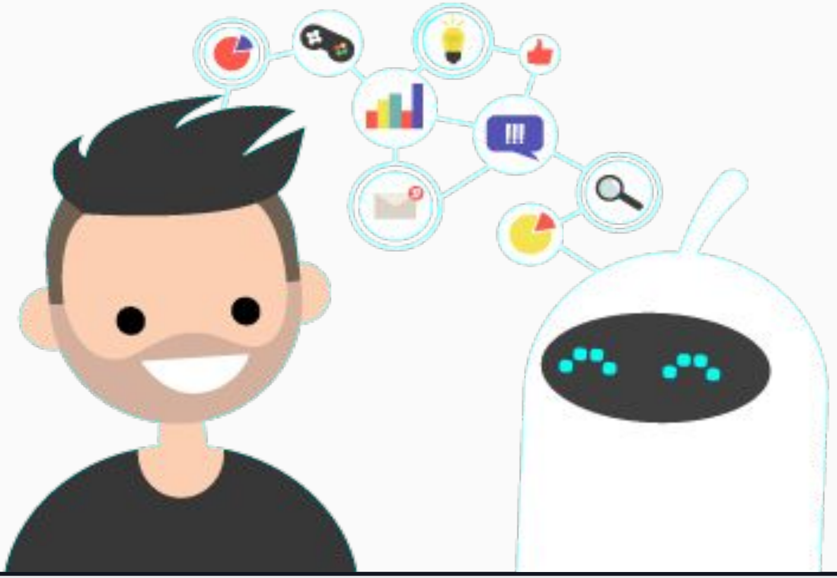
As credit card use increases,
fraud increases.


1234 5678 901
3456



Total Losses From Credit Card Fraud

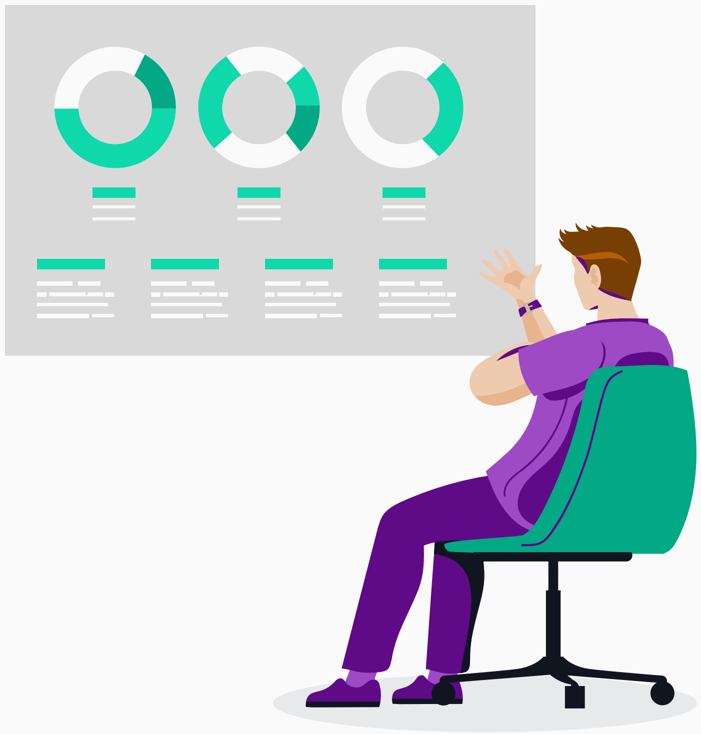
It Has Taken a Toll on EVERYONE!



02

Dataset

What dataset did we use?



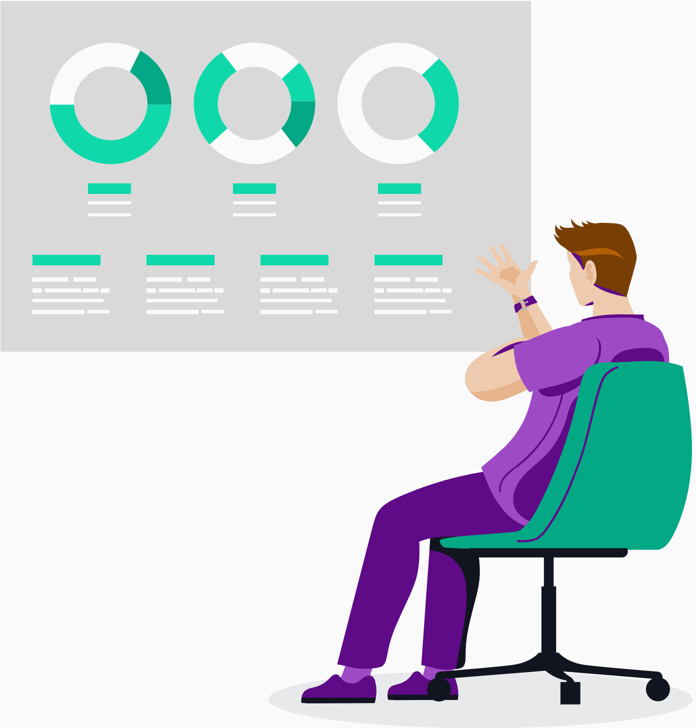
Simulated Dataset

- ➔ Dataset obtained from a Kaggle page
- ➔ Generated from the Sparkov simulator
  - ◆ Created by Brandon Harris
- ➔ The simulator is tuned to the parameters which included:
  - ◆ Date range from: January 1st 2019 to December 31st 2020
  - ◆ Around 1000 customers and cards
  - ◆ 800 different merchants for purchases
  - ◆ Included legitimate and fraudulent for testing
- ➔ Prior to starting, we also split into the test and train set

03

Attributes

What were the main variables to look at?



state	zip	lat	long	city_pop	job	dob	trans_num	unix_time	merch_lat	merch_long
SC	29209	33.9659	-80.9355	333497	Mechanical engineer	1968-03-19	2da90c7d74bd46a0caf3777415b3ebd3	1371816865	33.986391	-81.200714

trans_date...	cc_num	merchant	category	amt	first	last	gender	street	city
2020-06-21 12:14:25	2291163933867244	fraud_Kirlin and Sons	personal_care	2.86	Jeff	Elliott	M	351 Darlene Green	Columbia

Our model utilized many attributes to decide on whether a transaction is fraudulent or not.

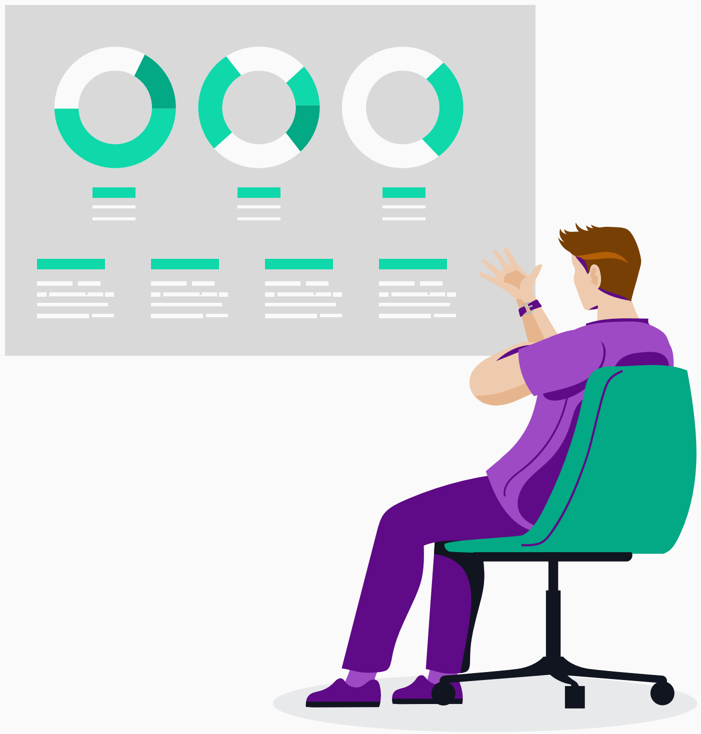
The main ones we used are:

- Merchant
- Amount
- Date

04

Methods Used & Results

What was the outcome?



Implementation

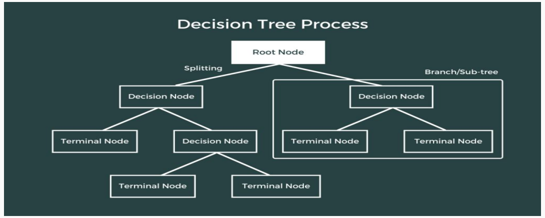
```
import pandas as pd
import numpy as np
from sklearn.tree import DecisionTreeClassifier, plot_tree, export_text
from sklearn.metrics import mean_squared_error, r2_score, mean_absolute_error, mean_absolute_percentage_error
from sklearn.metrics import confusion_matrix, precision_score, recall_score, f1_score, accuracy_score
from sklearn.preprocessing import StandardScaler
```

	Unnamed: 0	cc_num	merchant	category	amt	gender	zip	lat	long	merch_lat	merch_long	year	month	day	hour	minute	second
0	0	2703186189652095	514	8	4.97	0	28654	36.0788	-81.1781	36.011293	-82.048315	2019	1	1	0	0	18
1	1	630423337322	241	4	107.23	0	99160	48.8878	-118.2105	49.159047	-118.186462	2019	1	1	0	0	44
2	2	38859492057661	390	0	220.11	1	83252	42.1808	-112.2620	43.150704	-112.154481	2019	1	1	0	0	51
3	3	3534093764340240	360	2	45.00	1	59632	46.2306	-112.1138	47.034331	-112.561071	2019	1	1	0	1	16
4	4	375534208663984	297	9	41.96	1	24433	38.4207	-79.4629	38.674999	-78.632459	2019	1	1	0	3	6

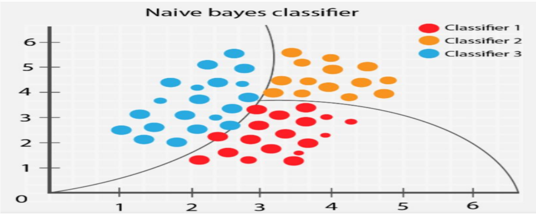
+ Code + Markdown

Methods Used

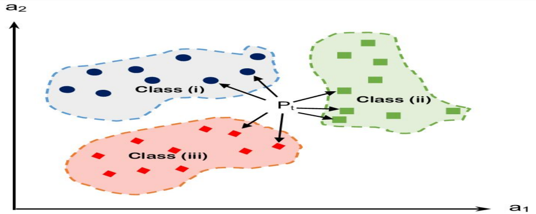
Decision Tree



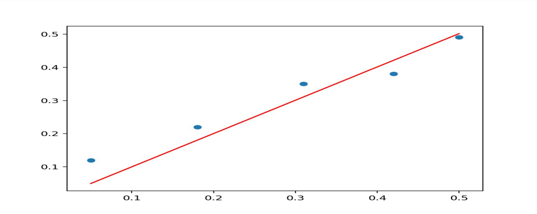
Naive Bayes



K-NN



LNR-Regression



Decision Tree

- Supervised learning method for classification or regression.
- Data set required a classification model so utilized a decision tree classifier.
- Tuned the classifier to a max depth of 9 which produced the best results.

Sample of Tree

```
|--- feature_4 <= 695.45
|   |--- feature_4 <= 259.04
|   |   |--- feature_14 <= 21.50
|   |   |   |--- feature_14 <= 3.50
|   |   |   |   |--- feature_4 <= 24.04
|   |   |   |   |   |--- feature_3 <= 3.50
|   |   |   |   |   |   |--- feature_3 <= 1.50
|   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |--- feature_3 >  1.50
|   |   |   |   |   |   |   |--- feature_3 <= 2.50
|   |   |   |   |   |   |   |   |--- feature_4 <= 21.71
|   |   |   |   |   |   |   |   |   |--- class: 1
|   |   |   |   |   |   |   |   |--- feature_4 >  21.71
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- feature_3 >  2.50
|   |   |   |   |   |   |   |   |--- feature_4 <= 16.03
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- feature_4 >  16.03
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |--- feature_3 >  3.50
|   |   |   |   |   |   |--- feature_4 <= 6.30
|   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |--- feature_4 >  6.30
|   |   |   |   |   |   |   |--- feature_3 <= 10.00
|   |   |   |   |   |   |   |   |--- feature_3 <= 8.50
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |   |--- feature_3 >  8.50
|   |   |   |   |   |   |   |   |   |--- class: 0
|   |   |   |   |   |   |   |--- feature_3 >  10.00
|   |   |   |   |   |   |   |   |--- class: 0
```

Results

Accuracy

99.83%

Precision

82.17%

Recall

69.83%

F1 Score

75.5%

R2 Score

54.51%

Matrix

[[553249   325]
[647 1498]]

Other Models

LNR-Regression

1. Mean Absolute Error: 0.995%
2. Mean Squared Error: 0.385%
3. Root Mean Squared Error: 6.204%
4. R2 Score: -0.0012893538391397942

K-NN

1. Accuracy: 99.54%
2. Precision: 0.2.98%
3. Recall: 0.559%
4. F1 Score: 0.942%
5. Confusion Matrix: [[553184, 390] [2133, 12]]

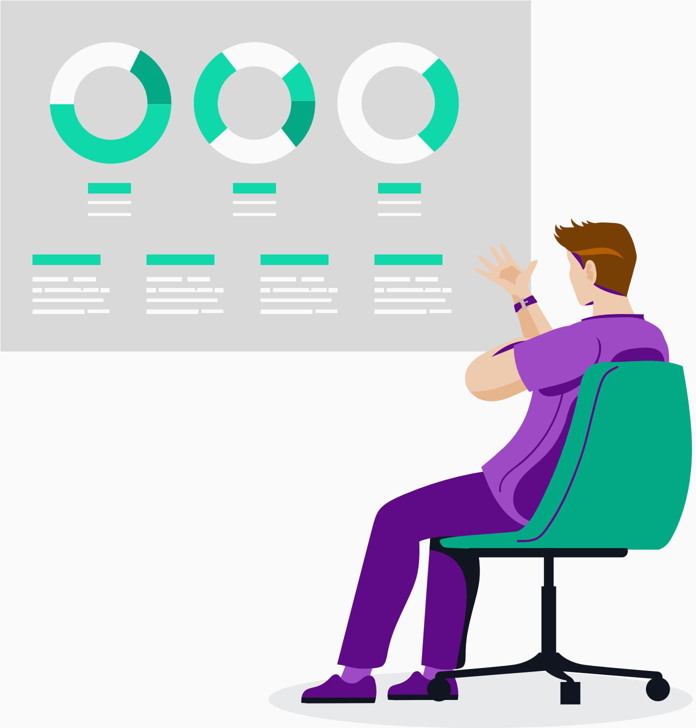
Naive Bayes

1. Accuracy: 99.61%
2. Mean Absolute Error: 0.386%
3. Mean Squared Error: 0.386%
4. Root Mean Squared Error: 6.213%
5. Confusion Matrix: [[553574, 0] [2145, 0]]

06

Challenges & Key Takeaways

What have we learned and what obstacles did we face?



Dataset

Very small proportion of actual fraudulent data (0.3% of transactions were fraudulent)

Model Selection

Very large data set doesn’t allow for tuning of hyperparameters

Thank You!