# CREDIT CARD FRAUD DETECTION

*Jade Bell*
*CSC 325 001*

# TABLE OF CONTENTS

# 01.

## CREDIT CARD FRAUD

**What is Credit Card Fraud?**

Credit Card Fraud occurs when an unauthorized individual gains access to another person's information and uses it to make purchases or request cash advances without their consent.

Some ways scammers get your information:

- Lost/stolen credit cards
- Phishing/scam techniques
- Hacking devices

02.

# PROBLEM & POSSIBLE SOLUTION


# PROBLEM

How can we detect fraudulent transactions to decrease the risk of card fraud?

## CREDIT CARD FRAUD DETECTION

The process of identifying purchase attempts that are fraudulent

## POSSIBLE SOLUTION:

Create a system/models that tracks the patterns of all abnormal transactions

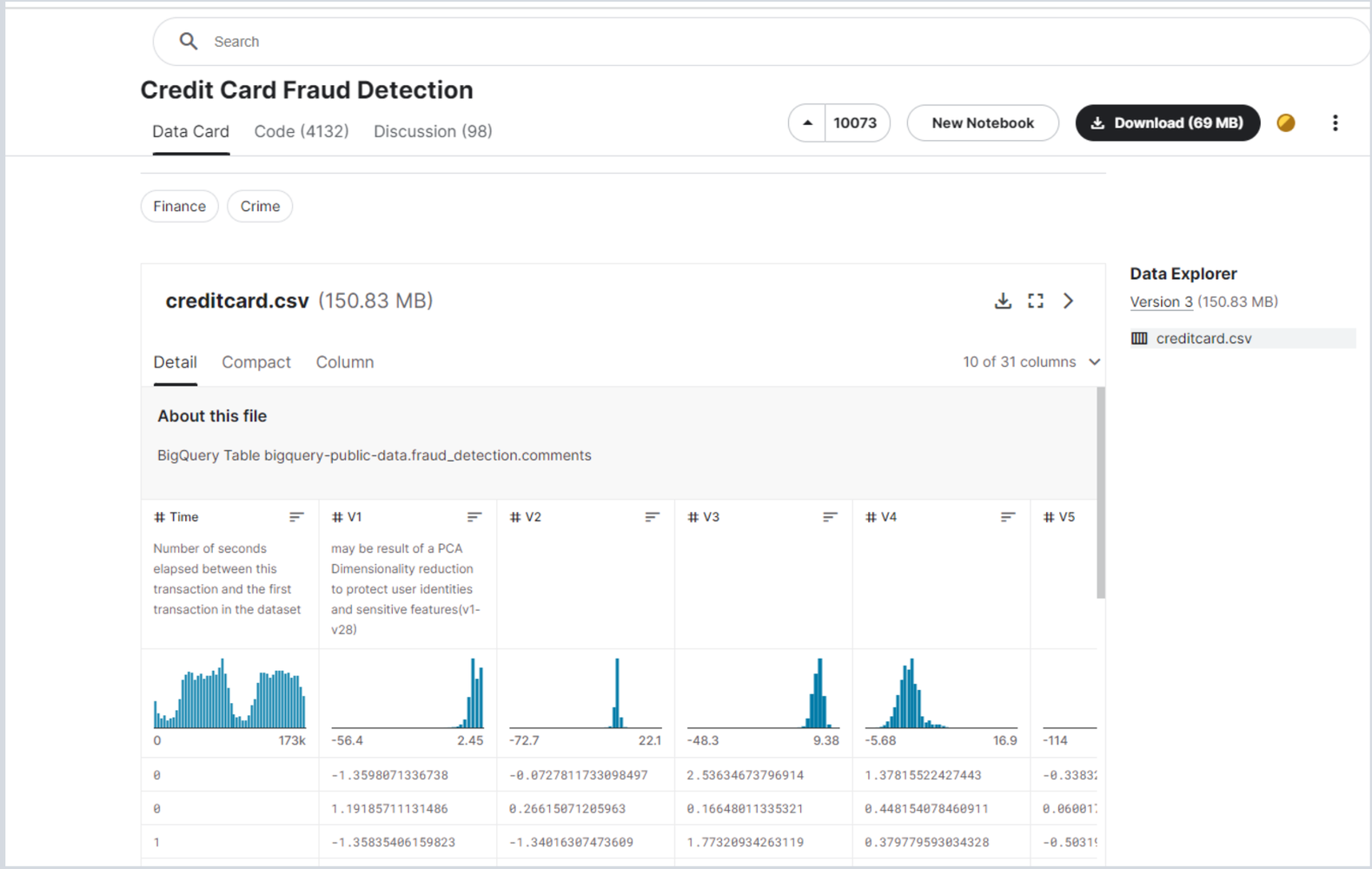
Main Goal -> Detect credit card fraud through dataset behavior

# DATA INFORMATION

# KAGGLE DATASET

Kaggle Dataset:

Contains card transactions made in September 2013 by European cardholders.



# KAGGLE DATASET: STEPS/ LIBRARIES

LIBRARIES:

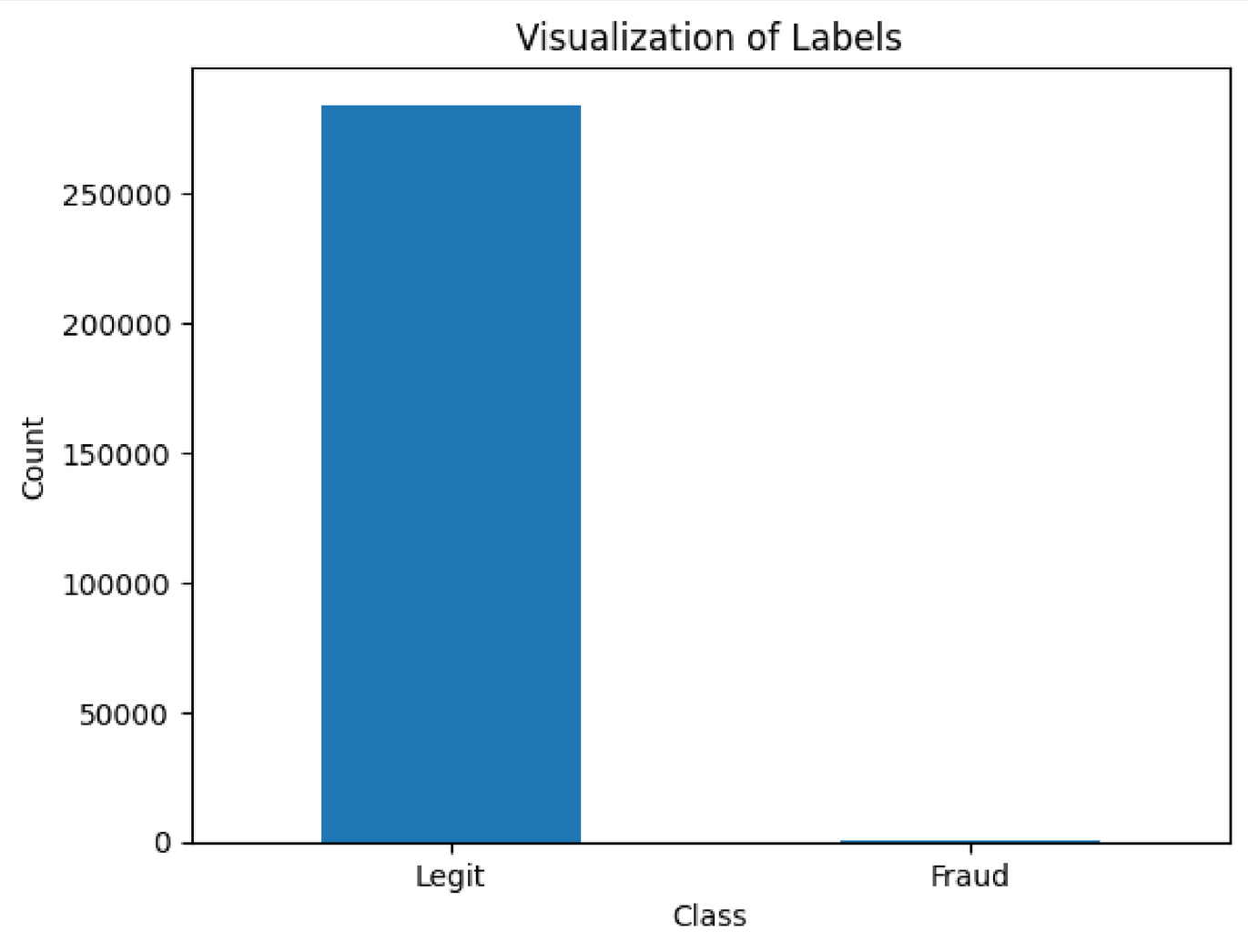
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, confusion_matrix
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import cross_val_score
import seaborn as sns
```

MAIN STEPS:

1 Classify credit card transactions as regular/fraud
2 Using detection methods to identify transactions from normal to fraud
3 Split data Train 80% & Test 20%
4 Find which model performs the best

# UNBALANCED DATA?

**Number of Legit transactions:  284315**
**Number of Fraud transactions:  492**
**Percentage of Fraud transactions: 0.1727(17%)**




- **Highly Unbalanced Data**

**Unbalanced Data ->One class of data is recognized instead of the other**

- **We cannot fit this current data into a ML model because if we trained the model based on this data, it will not be able to recognize the fraudulent transaction.**

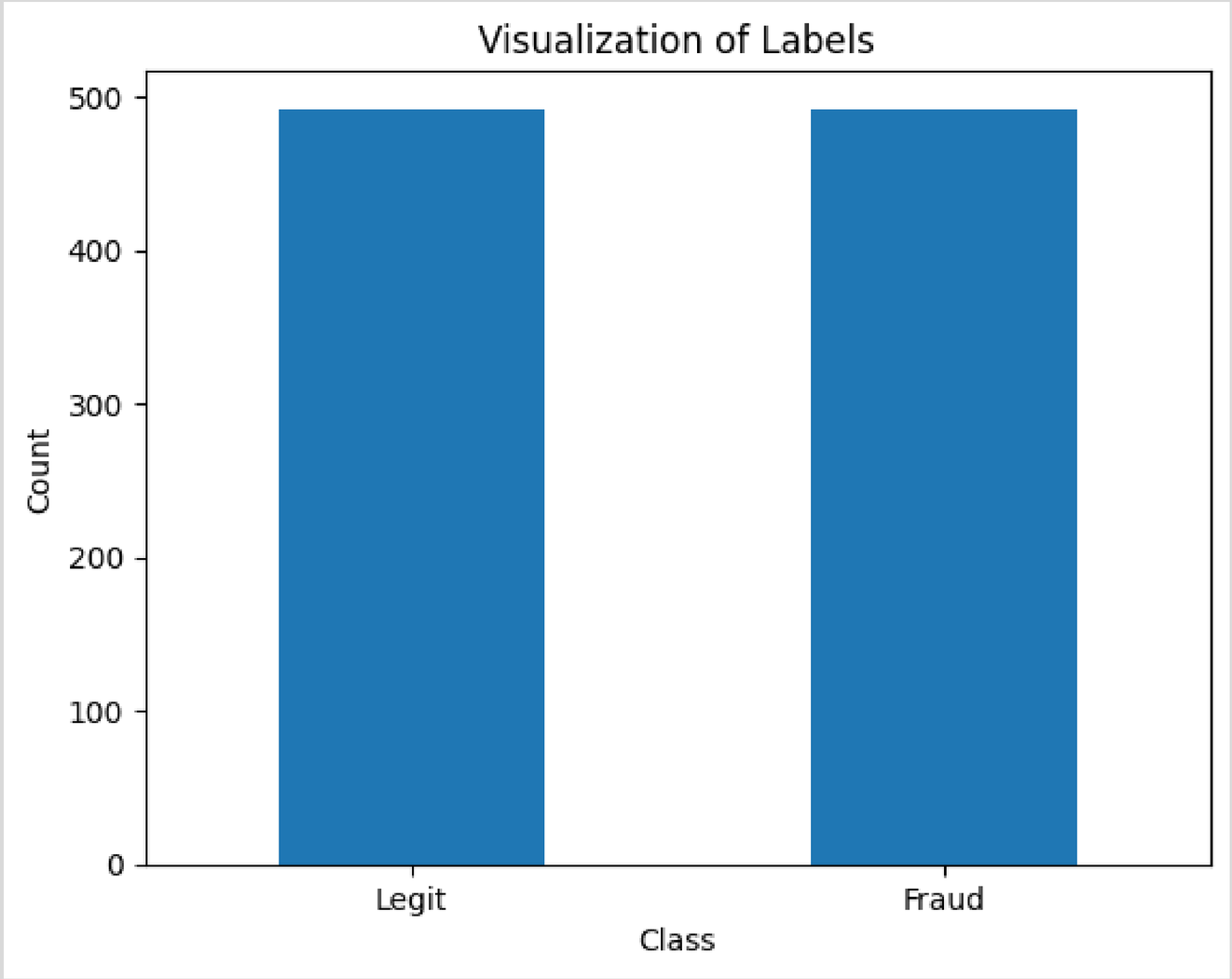
# How to fix the unbalanced data

- Change the shape & number of legit transactions to equal fraud

```
legit_sample = legit.sample(n=492)
```

- Conconate two data frames

```
new_ccd = pd.concat([legit_sample, fraud],  axis=0)
```



# Model 1: Logistic Regression

```
▾ LogisticRegression
LogisticRegression()
```

Accuracy Score: Train & Test

```
[ ]  # Accuracy on training data
     X_train_prediction = lr.predict(X_train)
     training_data_accuracy = accuracy_score(X_train_prediction, Y_train)

     print('Accuracy on Training Data:  {:.3f}%'.format(training_data_accuracy * 100))
```

```
Accuracy on Training Data:  93.393%
```

```
[ ]  # Accuracy on test data
     X_test_prediction = lr.predict(X_test)
     test_data_accuracy = accuracy_score(X_test_prediction, Y_test)

     print('Accuracy score on Test Data: {:.3f}%'.format(test_data_accuracy * 100))

     df_acc = pd.DataFrame(columns=['model_name', 'train_accuracy', 'test_accuracy'])
     df_acc = df_acc.append({'model_name': lr.__class__.__name__,
                             'train_accuracy': training_data_accuracy,
                             'test_accuracy': test_data_accuracy}, ignore_index=True)
     df_acc.head()
```

```
Accuracy score on Test Data: 92.386%
<ipython-input-35-2476300a2e65>:8: FutureWarning: The frame.append method is deprecated and will be removed from panc
  df_acc = df_acc.append({'model_name': lr.__class__.__name__,
```

| | model_name | train_accuracy | test_accuracy |
|---|---|---|---|
| 0 | LogisticRegression | 0.933926 | 0.923858 |

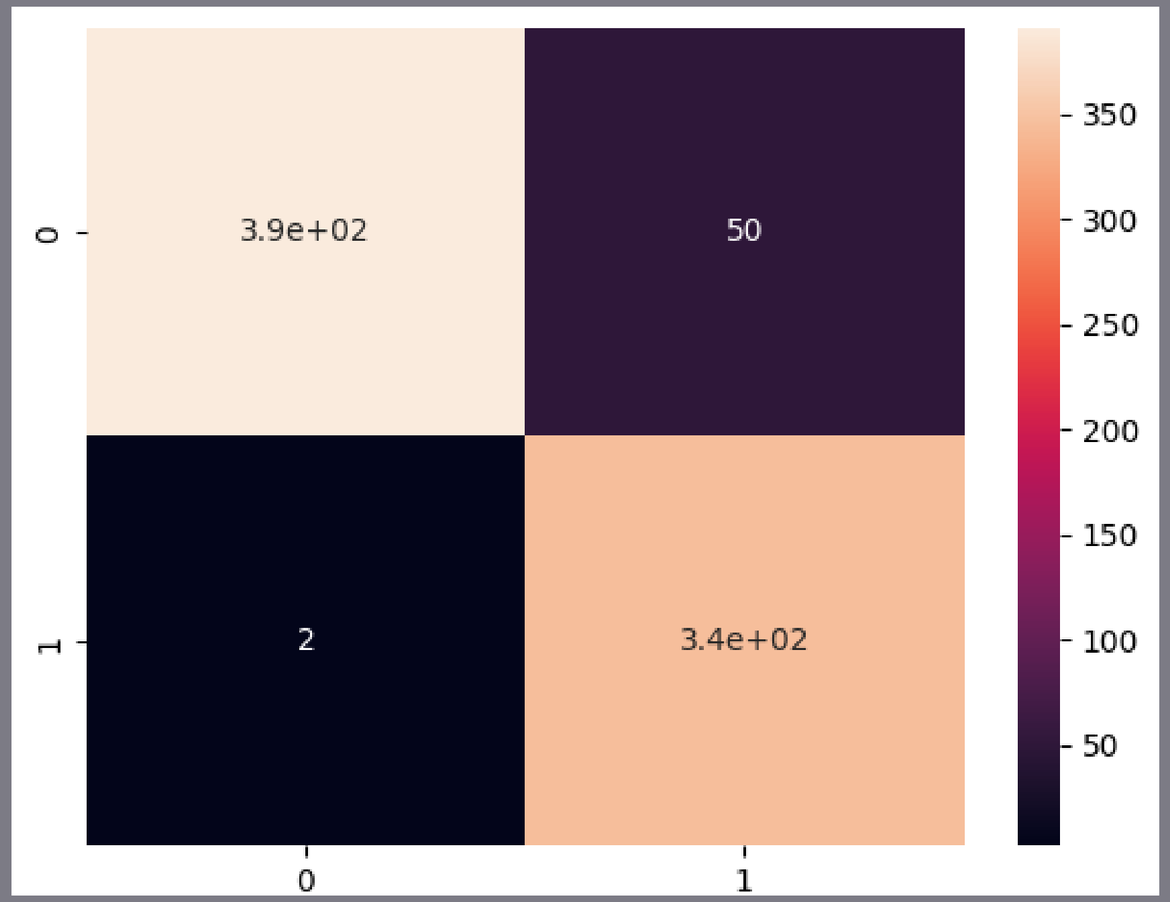
Accuracy on Training Data:
**93.393%**

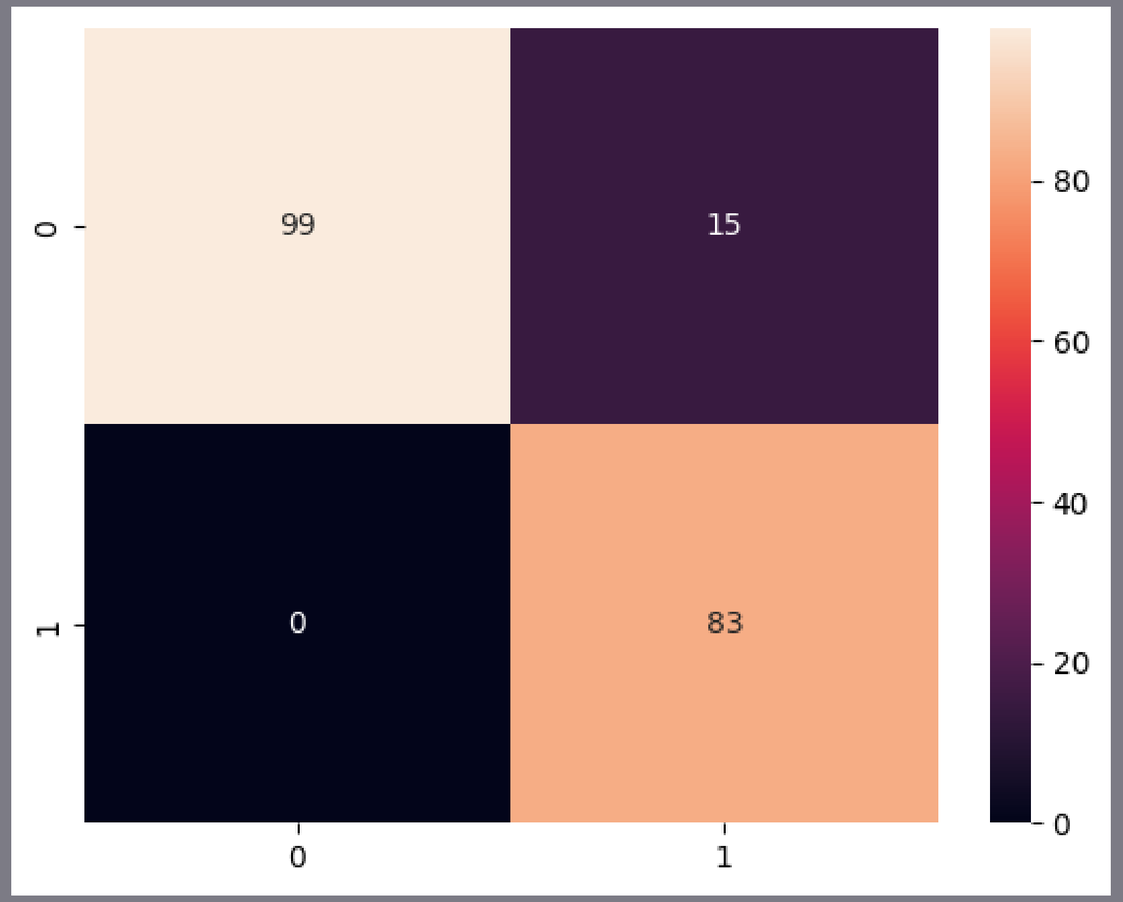
Accuracy on Test Data:
**92.385%**

**Cross Validation**
Scores: [0.91370558 0.93384224]
Mean: 0.9237739114710479
Standard deviation: 0.0100683277147027 3

# Logistic Regression (cont)

## Confusion Matrix & Seaborn



03.

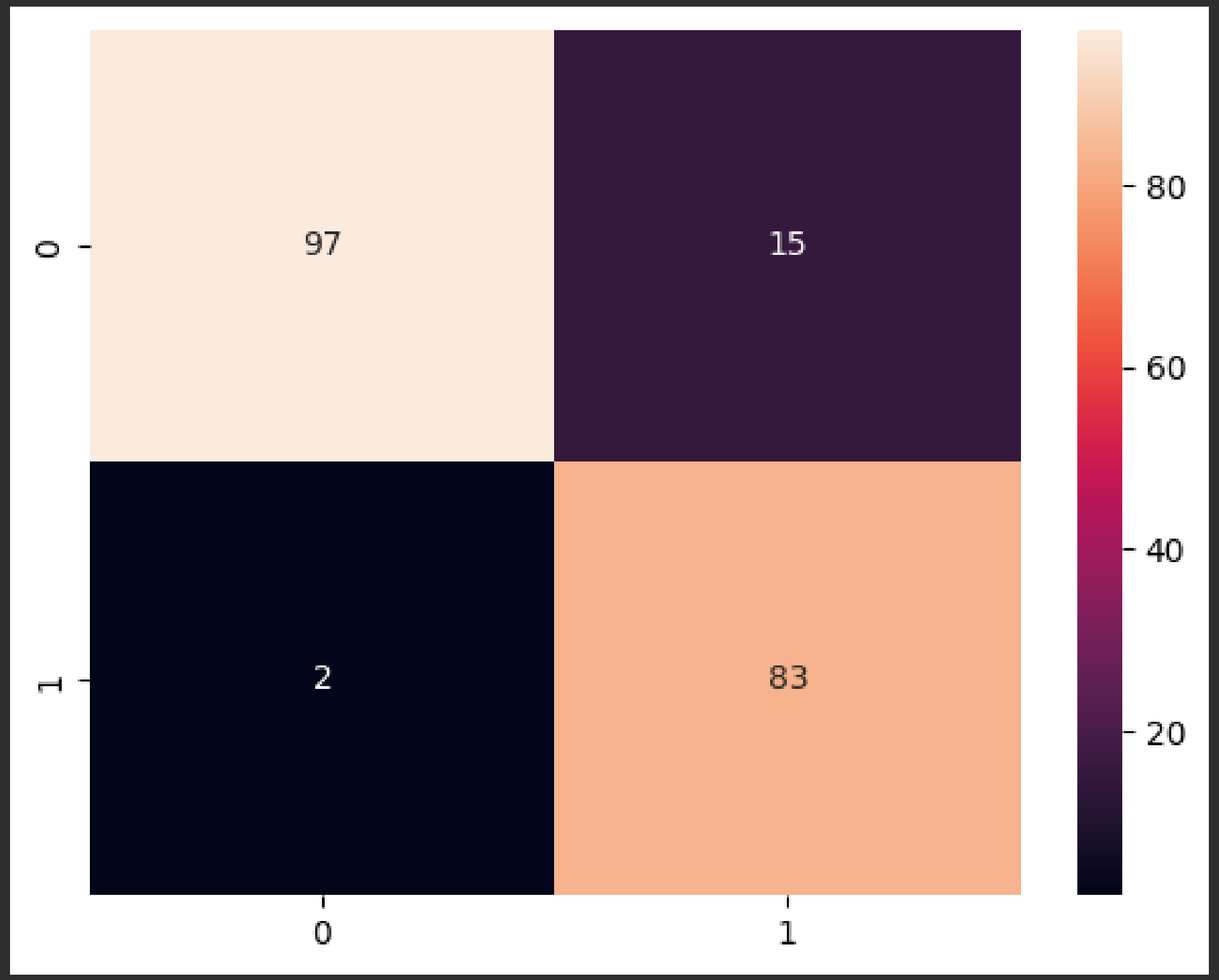
# MODEL COMPARISONS

*Which model has the best performance?*

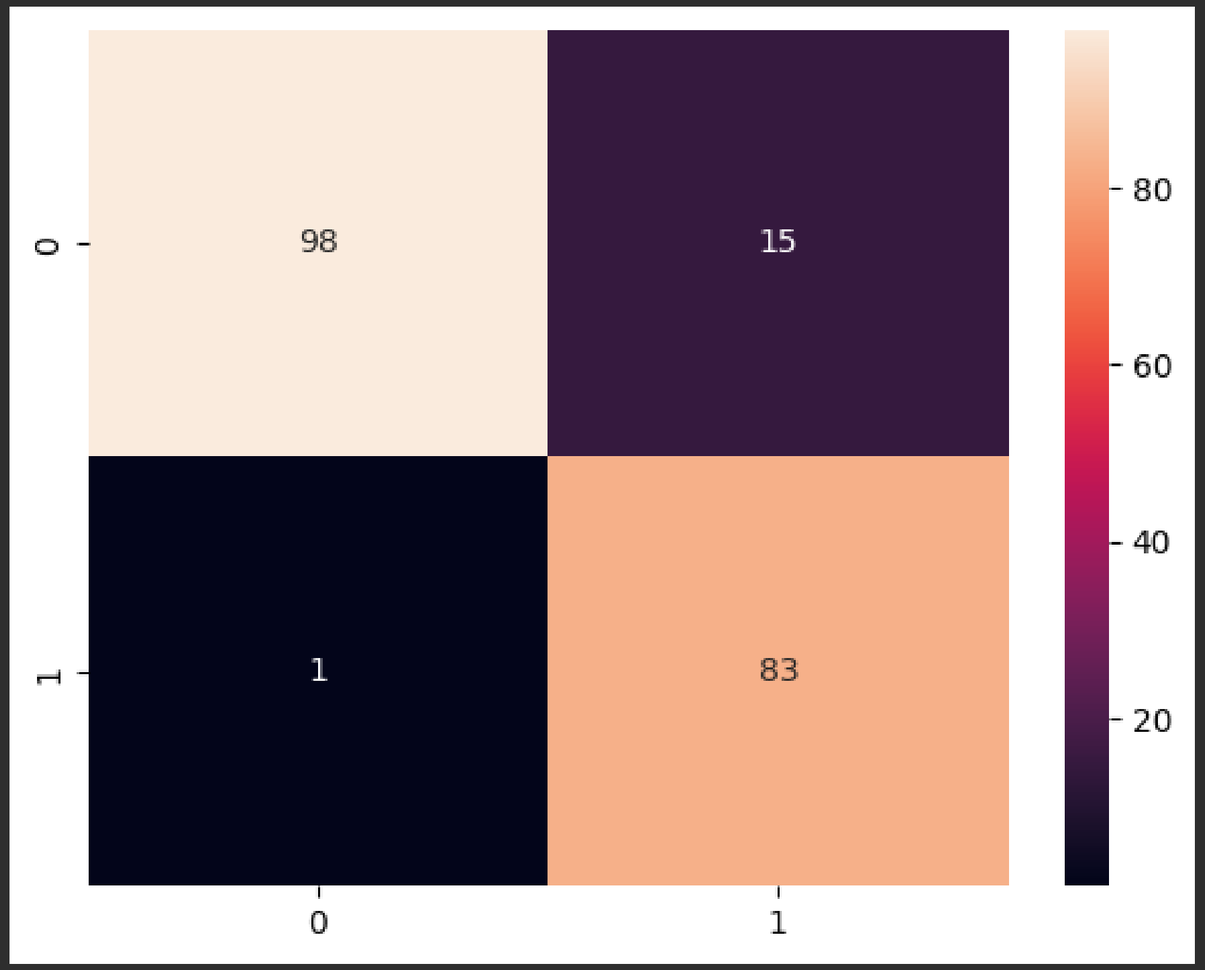
| Model Name | Training Set Acc | Test Set Acc | CV Mean | CV Highest Score(m+std) |
|---|---|---|---|---|
| Logistic R | 93.77% | 92.38% | 92.75% | 93.38% |
| SVM | 95.04% | 91.87% | 93.77% | 94.40% |
| RBF | 95.55% | 92.38% | 93.64% | 94.64% |
| Decision Tree | 97.83% | 91.37% | 89.83% | 91.65% |
| RNF | 97.45% | 92.38% | 93.64% | **94.69%** |

# *TEST SET*

*DT*

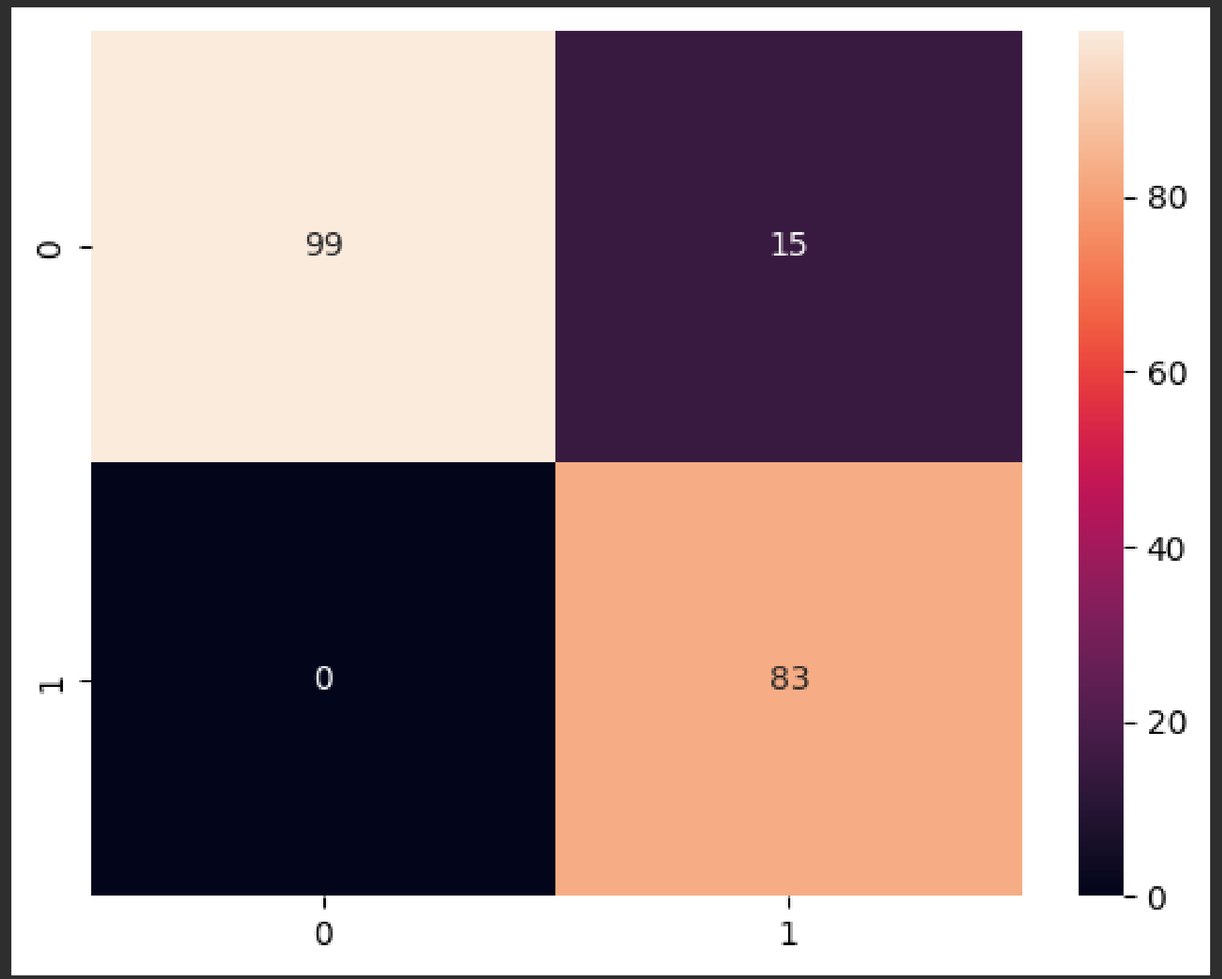



*SVM*

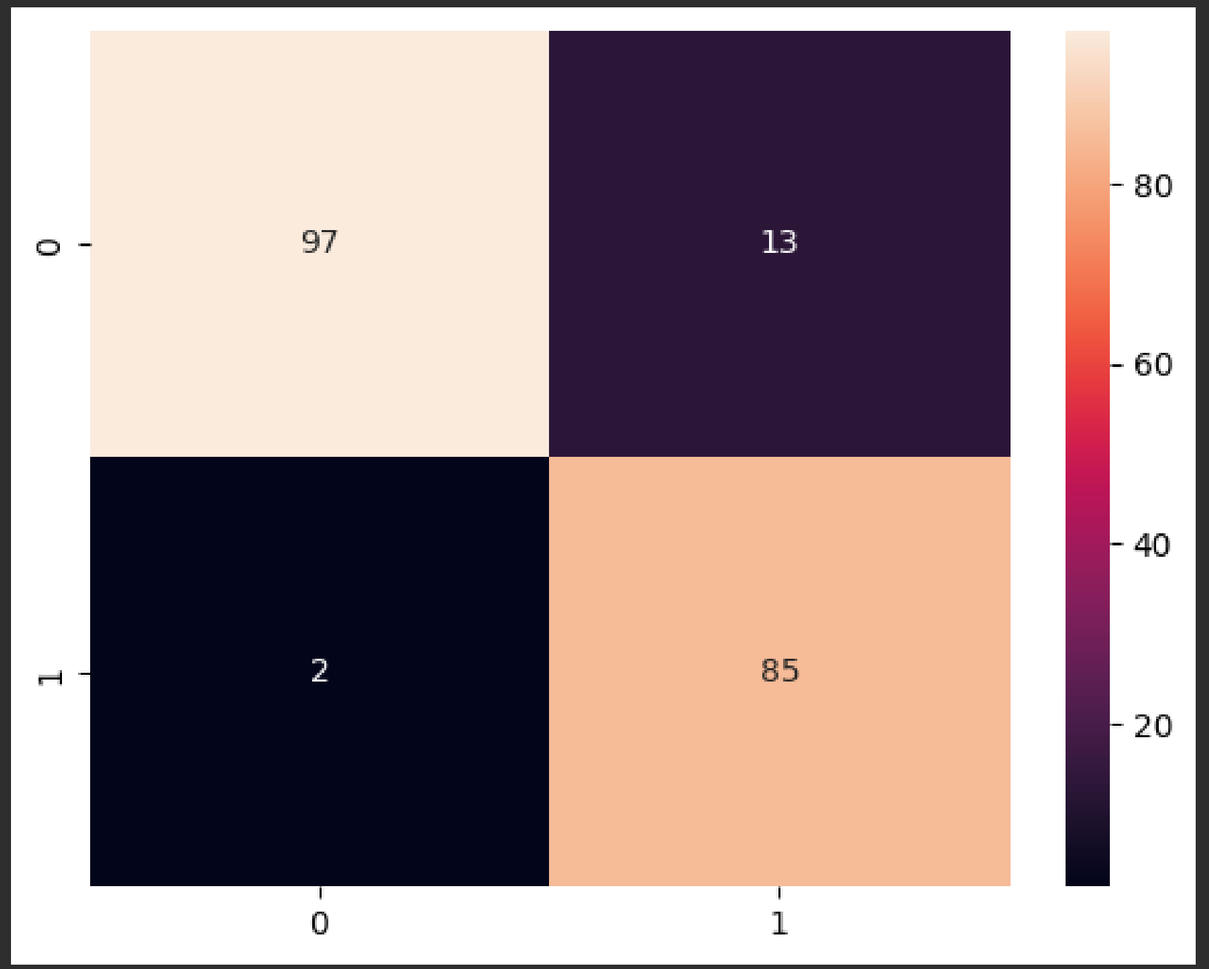


# *TEST SET*

## *RBF*




## *RNF*



# 04.

# IMPORTANCE?


## Reason 1

- Companies are able to inspect fraudulent credit card transactions so that customers are not charged with items not purposely purchased

  - Fraud detection techniques & models help merchants identify whether a purchase is legitimate


## Reason 2

- Fraud detection and prevention stops scammers from stealing your personal information and make outside purchases

# SUMMARY

- Credit Card Fraud Detection: Classification Task(legit & fraud)
- The key to preventing credit card fraud is to detect suspicious events that seem abnormal
- It is important to decipher what model best performs well in order for others to utilize
- *Even though some models obtained the same accuracy score for a test run, the model with the best performance in terms of cross-validation high score is Random Forest with 94.69%*
- Credit card fraud detection can prevent the risk of financial losses & increased fraud attacks

# THANK YOU

Have any question?

# REFERENCES


Inscribe. “Credit Card Fraud Detection: Everything You Need To Know.” Fraudulent Document Detection & Automation, Inscribe, 11 Apr. 2023, https://www.inscribe.ai/fraud-detection/credit-fraud-detection#:~:text=Credit%20card%20fraud%20detection%20is,fraud%20and%20stop%20fraudulent%20transactions.

Saxena, Pranjal. “Credit Card Fraud Detection Using Machine Learning & Python.” Medium, Towards Data Science, 13 Sept. 2022, https://towardsdatascience.com/credit-card-fraud-detection-using-machine-learning-python-5b098d4a8edc.

ULB, Machine Learning Group -. “Credit Card Fraud Detection.” Kaggle, 23 Mar. 2018, https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud.

White, Alexandria. “Here's How Credit Card Fraud Happens and Tips to Protect Yourself.” CNBC, CNBC, 27 May 2022, https://www.cnbc.com/select/credit-card-fraud/#:~:text=Credit%20card%20fraud%20occurs%20when,Hacking%20your%20computer.