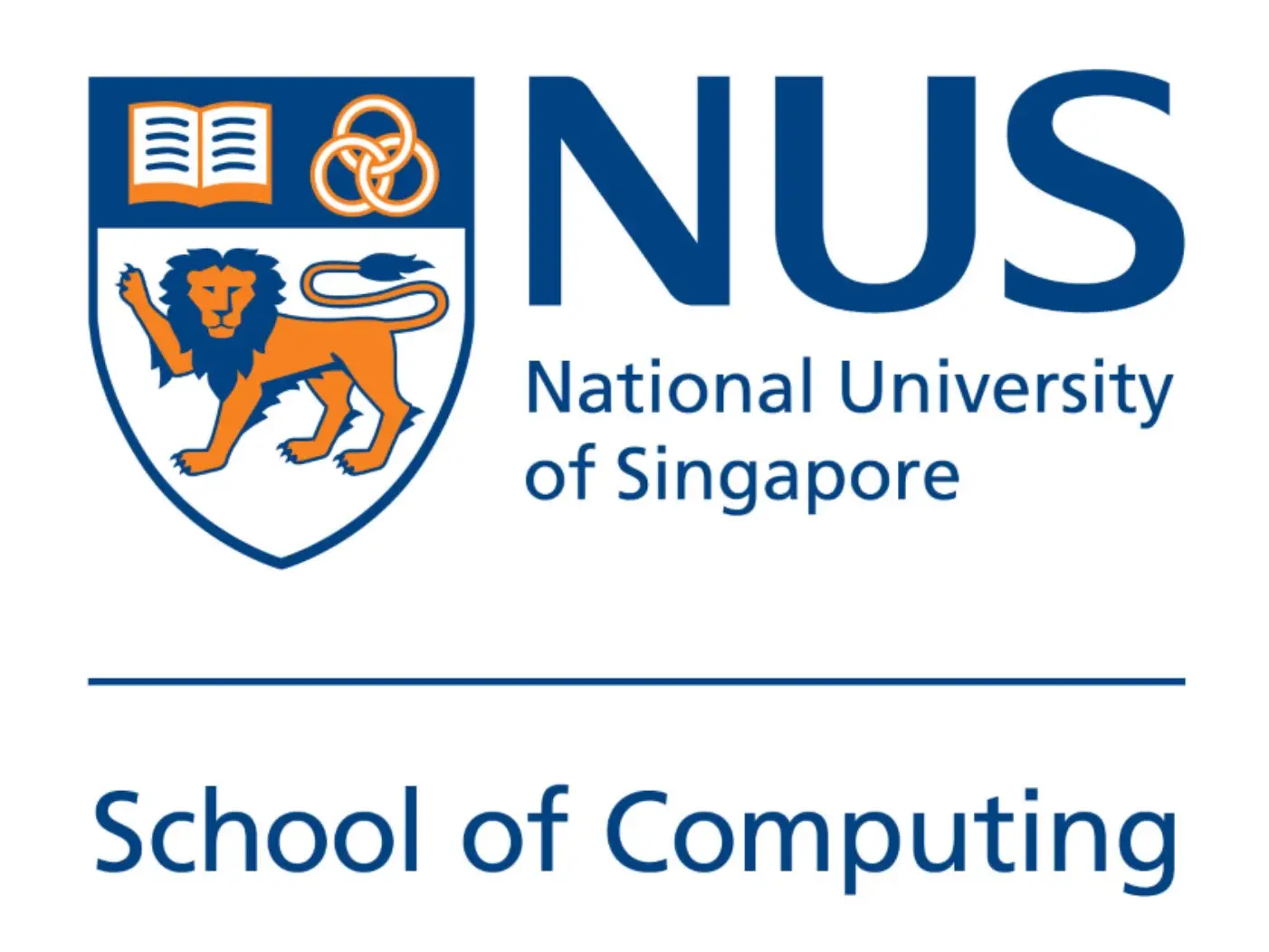
****

**BT4012 Fraud Analytics**

| **Name** | **Matriculation Number** |
| --- | --- |
| Kenny Low Sheng Wei | A0233869N |
| Ng Wei Han | A0234338E |
| Jeyakumar Jegan | A0217456B |
| Dylan Wo Qiying | A0233634J |
| Khoo Wu Chuan | A0217391E |

**Github Link:** [**https://github.com/NWH21/BT4012-Group-24**](https://github.com/NWH21/BT4012-Group-24)

**Content Page**

[**1. Motivation 3**](#_xdnvp2n67lw7)

[**2. Data Overview 4**](#_u9ff8tgmkmyn)

[**3. Data Pre-processing 5**](#_rh7m6dgh2tvt)

[3.1 Feature Selection 5](#_nafdrpm3mbpm)

[3.2 Handling Imbalanced Classes 6](#_ystn8uebrfut)

[3.3 Scaling Numerical Features 6](#_1398lojcwvwv)

[**4. Models 6**](#_92u4zav45op6)

[4.1 Gaussian Naive Bayes 6](#_x0enutc9qomw)

[4.2 Decision Tree 7](#_jolqfzf2ag80)

[4.3 Random Forest 7](#_3uhlgpx5n3v1)

[4.4 AdaBoost 7](#_tluc8wxwtybw)

[4.5 LightGBM 7](#_gmh0dyl7pdo5)

[**5. Performance 8**](#_emryqju1hqj3)

[**6. Interpretation 8**](#_fimxk794b2gz)

[6.1 Feature Importance 8](#_v2i4wkxty3ve)

[6.2 Integration Into Business Process 9](#_24v50w7l6thz)

[**7. Conclusion 10**](#_d06nmn8wmmjn)

[**References 11**](#_t06vb14ll9kl)

# Motivation

Credit card fraud is widespread within the finance industry and has resulted in approximately $30 billion in losses for merchants and card acquirers globally from 2014 to 2021[[1]](#footnote-0) The implementation of fraud detection is paramount in protecting individuals from financial losses and alleviating the stress associated with unauthorized transactions. It also builds trust in financial transactions, encouraging people to continue using credit cards for convenience.

On the business front, fraud detection strategies play a vital role in safeguarding customers' bank accounts and ensuring the financial integrity of transactions. Automated fraud detection systems are cost-effective, reducing the expenses associated with manual detection processes and contributing to overall cost reduction.

Beyond financial considerations, credit card fraud detection improves the user experience by quickly identifying and preventing fraudulent transactions. Furthermore, businesses with strong security measures and low fraud risks build a positive brand reputation, attracting more customers and establishing trust with partners.

Therefore, it is crucial to address this problem through the implementation of effective classification models that can accurately differentiate between fraudulent and non-fraudulent transactions, with an emphasis on minimizing false positives and false negatives.

# Data Overview

Real credit card transaction datasets often contain sensitive and personal information, which makes it extremely challenging to find a public credit card transaction dataset without anonymous features. Therefore, we decided to make use of a simulated credit card transaction dataset generated using Sparkov Data Generation. The simulator used to create the dataset uses multiple statistical distributions to capture real financial data. This approach enhances the interpretability of our models and ensures their applicability in real-world scenarios.

We have two [datasets](https://www.kaggle.com/datasets/kartik2112/fraud-detection/data)[[2]](#footnote-1) one for training (fraudTrain.csv) and one for testing (fraudTest.csv). The datasets contain legitimate and fraudulent transactions from the duration of 1st Jan 2019 - 31st Dec 2020. They cover the credit cards of 1000 customers doing transactions with a pool of 800 merchants. Both datasets have 22 variables (columns). fraudTrain has 1296675 rows while fraudTest has 555719 rows.

The table below outlines the variables that are in the dataset.

| **Variable name** | **Type** | **Description** |
| --- | --- | --- |
| index | Integer | Unique Identifier for each row |
| trans\_date\_trans\_time | Date/Time | Transaction DateTime |
| cc\_num | Integer | Credit Card Number of Customer |
| merchant | String | Merchant Name |
| category | String | Category of Merchant |
| amt | Float | Amount of Transaction |
| first | String | First Name of Credit Card Holder |
| last | String | Last Name of Credit Card Holder |
| gender | String | Gender of Credit Card Holder |
| street | String | Street Address of Credit Card Holder |
| city | String | City of Credit Card Holder |
| state | String | State of Credit Card Holder |
| zip | String | Zip of Credit Card Holder |
| lat | Float | Latitude Location of Credit Card Holder |
| long | Float | Longitude Location of Credit Card Holder |
| city\_pop | Integer | Credit Card Holder's City Population |
| job | String | Job of Credit Card Holder |
| dob | Date | Date of Birth of Credit Card Holder |
| trans\_num | String | Transaction Number |
| unix\_time | Integer | UNIX Time of transaction |
| merch\_lat | Float | Latitude Location of Merchant |
| merch\_long | Float | Longitude Location of Merchant |
| is\_fraud | Integer | Target class |

# Data Pre-processing

In order to prepare our dataset, we undertook several crucial pre-processing steps aimed at feature selection, handling class imbalance, and scaling numerical features.

## 3.1 Feature Selection

Based on insights gained from our exploratory data analysis (EDA), we selected relevant features for our modeling efforts. Features such as 'Transaction Amount,' 'Age,' 'Category,' 'Time,' and 'Location' exhibited varying degrees of correlation with credit card fraud and were considered in our models.

## 3.2 Handling Imbalanced Classes

Credit card fraud datasets inherently exhibit imbalance, which stems from the infrequent occurrence of fraudulent transactions in the real world compared to the significantly more abundant instances of legitimate transactions.

This imbalance can cause machine learning models to lean towards learning more about the majority class (non-fraudulent transactions) during training, making it challenging for them to effectively recognize and handle the minority class (fraudulent transactions). To address this imbalance and ensure the effectiveness of our fraud analytics models, we employed techniques such as Synthetic Minority Over-sampling Technique (SMOTE), which generates synthetic samples of the minority class to rebalance the dataset. This approach is crucial for enhancing the model's ability to detect and accurately classify the rare instances of credit card fraud.

Furthermore, our detailed testing showed that SMOTE has the best performance out of the other resampling methods, achieving a balanced performance across all key metrics, recall and precision.

## 3.3 Scaling Numerical Features

Given the varying scales of numeric features in our dataset, we employed the Robust Scaler to mitigate the impact of outliers on our models. This scaling method is particularly useful when dealing with datasets that may contain extreme values, providing robustness to the presence of outliers.

# Models

A comprehensive evaluation was conducted using five distinct machine learning techniques (Gaussian Naive Bayes, Decision Tree, Random Forest, AdaBoost, and LightGBM) to discern their individual strengths and weaknesses in enhancing fraud detection capabilities. Prominent features consistently identified across multiple models include 'amt' (Transaction Amount) and 'category\_grocery\_pos', suggesting their significance in detecting potential credit card fraud.

## 4.1 Gaussian Naive Bayes

Gaussian Naive Bayes (GNB) is a supervised classification model that leverages Bayes' theorem. It assumes feature independence given the class label, making it particularly useful for quick decisions in real-time scenarios like fraud detection.

## 

## 4.2 Decision Tree

Decision Tree, a supervised learning technique helps construct a tree-like model that facilitates clear decision rules, aiding in understanding the features and criteria used for classification. In the context of fraud analytics, the Decision Tree model offers valuable insights into the factors influencing the identification of fraudulent transactions.

## 4.3 Random Forest

Random Forest is a supervised machine learning model for fraud detection, leveraging an ensemble of decision trees. It combines the strengths of multiple decision trees to provide a comprehensive and reliable solution. With 100 estimators and parallel processing (n\_jobs = -1), Random Forest aims to enhance predictive accuracy and robustness.

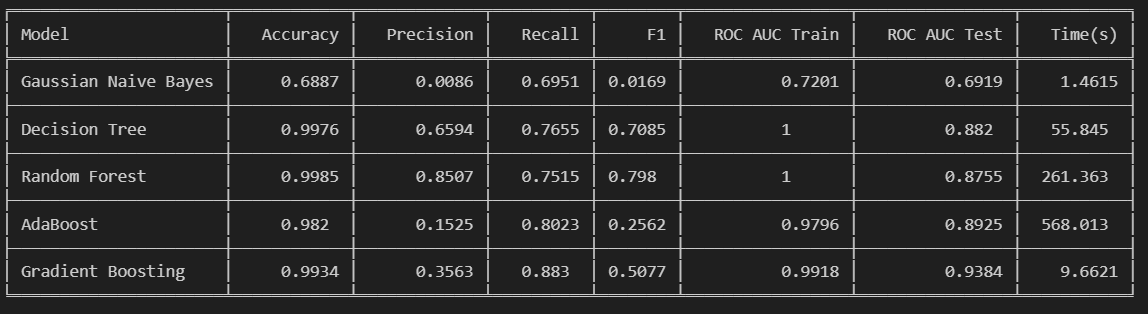
## 4.4 AdaBoost

The AdaBoost Classifier is implemented as a supervised machine learning model for fraud detection, employing an ensemble of weak learners, specifically Decision Trees with a maximum depth of 1. AdaBoost iteratively adjusts the weights of misclassified instances, emphasizing the challenging cases and creating a strong learner. With 100 estimators, AdaBoost aims to enhance the model's generalization capabilities, making it adept at identifying patterns indicative of fraudulent transactions.

## 4.5 LightGBM

The LightGBM model is a gradient boosting framework designed for efficient training and high performance. Implemented as a binary classification model, it leverages the boosting technique to create an ensemble of decision trees. In this supervised learning approach, the model aims to accurately classify instances as either fraudulent or non-fraudulent based on the provided features. The hyperparameters of the LightGBM model, including the number of leaves, learning rate, and feature fraction, are optimized through a grid search to enhance its predictive capabilities.

# Performance



In the evaluation of machine learning models for fraud detection, the Gaussian Naive Bayes model achieved a moderate level of accuracy, registering at 68.87%, but faced challenges in precision. The model displayed moderate discrimination ability, as indicated by its AUC scores. The low precision could be due to the classifier’s assumption of feature independence. In fraud detection, features are usually not completely independent, therefore impacting the performance of the Gaussian Naive Bayes model. Moving on to the Decision Tree model, it showcased high accuracy at 99.76%, with a balanced high recall. However, this was accompanied by a trade-off in precision, and despite its interpretability with clear decision rules, the model exhibited a longer processing time. The Random Forest Classifier demonstrated outstanding comprehensive performance, achieving high accuracy, balanced recall, and an impressive F1 Score. It excelled in discrimination, as reflected in both AUC scores. AdaBoost exhibited high accuracy and recall, with a notable trade-off between precision and recall. The model's longer processing time was offset by its strong discrimination ability, as indicated by AUC scores. Finally, LightGBM stood out with outstanding recall and a balanced F1 Score, coupled with efficient processing and excellent discrimination.

# Interpretation

## 6.1 Feature Importance

The importance of the 'amt' feature in credit card fraud detection models is rooted in its role as a strong indicator of potential fraud. Unusual or extreme transaction values are deemed crucial markers, aligning with the financial impact of credit card fraud. The models prioritize 'amt' to swiftly identify transactions deviating from typical spending patterns, recognizing it as a key determinant in distinguishing between fraudulent and non-fraudulent activities.

The significance of 'category\_grocery\_pos' in credit card fraud detection arises from its role in identifying anomalies within routine transactions. As a common and frequent spending category, fraudsters may exploit grocery purchases to disguise unauthorized activities. The models prioritized this category to capture deviations in spending patterns, recognizing them as vital indicators of potential fraud.

## 6.2 Integration Into Business Process

Gaussian Naive Bayes (GNB) brings notable strengths to the table regarding social and business implications. Its computational efficiency makes it well-suited for real-time fraud detection, contributing to improved security by quickly identifying potential fraudulent transactions. This rapid decision-making process enhances trust in financial systems, providing customers with a sense of security during their transactions. However, GNB's limitations should be acknowledged. Its challenges in handling imbalanced datasets may result in higher false positives and missed fraudulent transactions, impacting its effectiveness in loss prevention. The lower precision of GNB might lead to false alarms, potentially causing inconvenience to customers and affecting brand reputation.

The Decision Tree model excels in loss prevention due to its high accuracy, effectively identifying and preventing fraudulent transactions. Its clear decision rules also contribute to an improved user experience by facilitating real-time fraud detection, allowing for a smoother process for legitimate transactions. On the flip side, the longer processing time of Decision Tree may impact cost reduction efforts compared to faster models. This could be a limitation for businesses aiming for efficient automated fraud detection.

Random Forest's exceptional accuracy significantly enhances security by effectively capturing a substantial portion of actual fraud cases. This robustness contributes to a positive brand reputation, assuring customers of strong security measures in place. However, similar to other models, the longer processing time of Random Forest may impact cost reduction efforts compared to faster models, potentially influencing the efficiency of automated fraud detection.

AdaBoost's strength lies in its focus on challenging instances, contributing to improved security by identifying complex patterns indicative of fraudulent activities. Its ability to capture a substantial proportion of actual fraud cases enhances trust in the effectiveness of financial systems. Yet, AdaBoost's longer processing time may pose challenges in terms of cost reduction efforts, impacting the efficiency of automated fraud detection.

LightGBM, with its exceptional accuracy and high recall, significantly contributes to improved security by effectively identifying a significant proportion of actual fraud cases. Its balance between precision and recall ensures accurate identification of fraudulent transactions, contributing to effective loss prevention. However, the processing time of LightGBM, while relatively efficient, needs consideration for real-time applications, influencing cost reduction efforts compared to faster models.

Our project's goals of improving security, building trust in financial systems, and reducing costs through automated fraud detection are well-addressed by the ensemble models, Random Forest, AdaBoost, and LightGBM. These models showcase strong discrimination, high accuracy, and an effective balance between precision and recall. While Decision Tree offers interpretability, its longer processing time may pose challenges for real-time applications. GNB, while computationally efficient, faces limitations in handling imbalanced datasets and achieving high precision.

# Conclusion

Based on our analysis of the different models, Random Forest achieved the highest F1-score of 0.798 and among the highest for AUC on the testing set. In fraud detection, minimizing false negatives is essential because it ensures that fraudulent transactions are not overlooked, which would erode the trust that consumers have in credit cards. Minimizing false positives is also important to avoid unnecessary disruptions or inconvenience to credit card holders. Therefore, Random Forest is our recommendation among the models proposed. However, it is important to note that the dynamic nature of fraud patterns demands regular updates to models as fraudsters change their patterns. Continuous monitoring, retraining, and incorporating the latest data and features are paramount for models to maintain relevance over time.

# Limitations

The experiment utilizes a simulated credit card transaction dataset from Sparkov Data Generation, enhancing interpretability and privacy but risking a lack of real-world transaction complexity. Additionally, it doesn't explicitly address the dynamic nature of evolving fraud patterns, necessitating continuous monitoring and model updates for long-term effectiveness. Lastly, the inherent class imbalance in credit card fraud datasets, despite employing techniques like SMOTE, may limit the model's generalization to real-world scenarios with varying class imbalances.

1. Statista. (2023, August 23). Card fraud - credit cards and debit cards combined - worldwide 2014-2021. <https://www.statista.com/statistics/1394119/global-card-fraud-losses/> [↑](#footnote-ref-0)
2. Kaggle. (2020). Credit Card Transactions Fraud Detection Dataset <https://www.kaggle.com/datasets/kartik2112/fraud-detection/data> [↑](#footnote-ref-1)