Project Methodology

Hasan Khwaja, Yifan Wang, atreyapurapu.k

March 2025

1 Purpose of the Methodology

In this project, we employ multiple machine learning classification algorithms (Naïve Bayes, Random Forest Classifiers, and Neural Networks) to detect fraudulent credit card transactions. Each algorithm offers unique advantages:

- **Naïve Bayes:** A fast, probabilistic classifier that works well even with high-dimensional data and limited training times, making it suitable for real-time detection.
- **Random Forest Classifiers:** Provide robust performance on imbalanced data and offer clearer insights into the model's decision-making process, via feature importance analysis.
- **Neural Networks:** Capable of modeling non-linear relationships effectively with large datasets and can potentially capture subtle fraud patterns.

In comparing these approaches, we rely on metrics such as *accuracy*, *precision*, *recall*, and *F1-score* to reveal each model's strengths and weaknesses in classifying minority (fraud) cases. Comparing them is necessary to:

- Identify which algorithm or combination of algorithms (ensemble) best handles the severe class imbalance.
- Understand trade-offs between true fraud detection and false alarms, given the risk and cost implications of each.
- Optimize hyperparameters and select the most suitable approach for integration into a real-time fraud detection system.

2 Problem Statement

2.1 Defining the Problem

This project addresses *credit card fraud detection*, a binary classification problem classifying each transaction as `fraud` or `legitimate`. Because actual fraud

accounts for a small fraction of overall transactions, high class imbalance poses a major challenge.

2.2 Significance and Relevance

Mitigating fraudulent transactions is critical for:

- **Financial Institutions & Users:** Fraud results in significant monetary losses and damages consumer trust.
- **Industry Applications:** Real-time detection systems must be both accurate and efficient, often operating on streaming data from large transaction networks.
- **Academic Research:** Novel algorithms and techniques for highly imbalanced data are a focus for machine learning and data mining communities.

Our work extends prior methods by applying multiple classification algorithms, focusing on feature engineering (e.g., distance between merchant and user, transaction time) to enhance fraud detection rates. Model performance is evaluated through metrics (F1-score, recall, precision, etc.) to ensure robust detection of fraudulent activity while minimizing disruption to legitimate card usage.

3 Data collection and preparation

3.1 Data sources

The dataset used for this project is a simulation of credit card fraud transactions used for the purpose of predicting credit card fraud. This dataset was taken from the simulated data on Kaggle in order to ensure that sensitive or personally identifiable information cannot be used against any individual.

3.2 Data description

The dataset covers the credit card activity with 23 columns and roughly 1.3 million rows. The dataset contains characteristics related to demographics, location, transaction details, and target data. This data is suitable for building a model aimed at predicting fraudulent credit card transactions due to its diversity in features.

3.3 Preprocessing steps

The dataset preparation for fraud detection involved multiple steps to ensure efficient and meaningful feature selection. First, essential libraries such as `pandas`, `numpy`, and `scikit-learn` were imported for data manipulation, preprocessing,

and dataset splitting. The dataset was then loaded from `fraudTrain.csv` into a Pandas DataFrame, forming the foundation for subsequent processing.

Next, irrelevant columns such as `cc_num`, `trans_num`, and `street` were removed to streamline the dataset. A key feature engineering step involved calculating the cardholder's age by extracting the year from `trans_date_trans_time` and subtracting it from the birth year in `dob`. The `dob` column was then replaced with the derived `age` feature, making the data more interpretable. Additionally, the geographical distance between the customer and the merchant was computed using the Haversine formula, leveraging latitude and longitude values. This newly created `distance` feature provided a spatial measure useful for fraud detection.

Categorical variables, including `merchant`, `category`, `gender`, and `job`, were transformed into numerical values using label encoding. This ensured that qualitative data could be processed effectively by machine learning models. Furthermore, geographical coordinates (`lat`, `long`, `merch_lat`, `merch_long`) were bucketed into ten equal bins using the `pd.cut()` function. This bucketing process helped retain spatial patterns while reducing excessive variability in the raw coordinate values.

To prepare the dataset for model training, an **80-20%** split was performed using `train_test_split`, with a fixed random state of 42 for reproducibility. Additionally, numerical features were normalized using `StandardScaler` to ensure consistent scaling across all variables. This standardization prevented features with larger scales from disproportionately influencing the model.

Finally, after feature engineering, redundant columns such as `trans_date_trans_time`, `first`, `last`, `city`, `state`, `zip`, `lat`, `long`, `merch_lat`, and `merch_long` were removed, as they were either replaced by derived features or deemed unnecessary. The resulting dataset was optimized for fraud detection, retaining only the most relevant and informative features.

4 Selection of Machine Learning Models

The models that were considered for this project were Naïve Bayes, Random Forest Classifiers, and Neural Networks. Naïve Bayes was considered here due to its computational efficiency, allowing for the algorithm to make fast predictions, suitable for real-time detection. Neural networks were considered here due to how they are able to learn and distinguish between legitimate transactions and patterns of fraud, in terms of non-linear relationships, when given a sufficiently large dataset such as the one used for this project. Random Forest Classifiers were used here due to how they clearly give insights into the model's decision-making process through the use of feature importance, allowing for better tuning and efficiency.

When comparing each of these models the metrics accuracy, precision, recall, and F1-score were used for comparison. All models exhibited near-perfect accuracy at roughly 0.98 – 0.99 with the F1 score resulting in 0.70 for the randomized forest classifier, 0.65 for the neural network and 0.24 for Naïve Bayes.

This discrepancy between the accuracy and F1-score shows that the models tend to identify legitimate transactions as fraud which can be attributed to how less than one percent of the data is made up as fraud. Since Randomized Forest Classifiers and Neural Networks have the best performance, in terms of identifying true fraudulent transactions, they will be chosen for implementation within the system.

5 Model Development and Training

The Random Forest Classifier is made up of decision trees built using a sample of the data at every node where features are considered at each split. The model uses the features, merchant, category, amount, city population, job, time, age, and distance as they are the most important features found through feature selection in order to help reduce the model complexity. These same features were also used for hyperparameter tuning. The model complexity was controlled by controlling the total number of trees, the maximum depth of trees, the samples needed to require a split, and whether or not bootstrap samples should be used when building trees.

To handle overfitting, cross-validation was used to ensure consistency in performance across different subsets of the training data. Feature selection was also used to select features with high importance as well as bootstrapping.

In terms of hyperparameters, randomized search was used to sample a defined number of parameter combinations from the distribution in order to find optimal parameters for the model.

The architecture of the consists of a **Sequential model** with multiple dense layers, incorporating **Dropout layers** to mitigate overfitting. The **Adam optimizer** was used for efficient weight updates, and a **binary classification output layer** was employed, indicating that the model is designed to detect fraudulent transactions.

For feature selection and processing, the dataset underwent preprocessing using **StandardScaler** to standardize numerical features. Categorical variables were transformed using **Label Encoding** to ensure they could be processed by the neural network. A dedicated function, `preprocess_fraud_data()`, was implemented to automate the preprocessing pipeline.

The dataset was split into **training and validation sets** using an **80-20%** split, ensuring that the model was trained on a significant portion of the data while reserving a subset for validation. The split was performed using the `train_test_split()` function from `sklearn.model_selection` with a fixed random state to ensure reproducibility.

To address class imbalance, techniques such as **oversampling and undersampling** were considered, as fraud detection datasets often contain significantly fewer fraudulent transactions compared to non-fraudulent ones. Additionally, **threshold tuning** was proposed as a method to improve recall for the minority class. The model was trained using **binary cross-entropy loss**, which is commonly used in fraud detection scenarios.

6 Evaluation and Comparison

The key evaluation metric used was F1-score due to how it is a balance between both precision and recall where precision measures the percentage of correct predictions and recall measures relevant predictions made. Targeting the balance between these two metrics is the reason F1-score is being used. Accuracy is also being evaluated although, since a small fraction of the dataset is made up of fraudulent transactions, it is critical that legitimate transactions are not labeled as fraudulent as the current models are performing with regard to high levels of accuracy.