# Machine Learning Model For Financial Crime Transaction Risk Classification

Pasindu De Silva (012279499)

[Machine Learning Model For Financial Crime Transaction Risk Classification 1](#_Toc185949945)

[**Part A: Letter of Transmittal** 3](#_Toc185949946)

[**Part B: Project Proposal Plan** 4](#_Toc185949947)

[Project Summary 4](#_Toc185949948)

[Data Summary 5](#_Toc185949949)

[Implementation 5](#_Toc185949950)

[Business Understanding – 5](#_Toc185949951)

[Data Understanding – 6](#_Toc185949952)

[Data Preparation – 6](#_Toc185949953)

[Modeling – 6](#_Toc185949954)

[Evaluation – 6](#_Toc185949955)

[Deployment – 6](#_Toc185949956)

[Timeline 7](#_Toc185949957)

[Evaluation Plan 7](#_Toc185949958)

[Resources and Costs 8](#_Toc185949959)

[Hardware and Software – 8](#_Toc185949960)

[Labor Costs – 9](#_Toc185949961)

[Environment Costs – 9](#_Toc185949962)

[**Part C: Application** 9](#_Toc185949963)

[**Part D: Post-implementation Report** 10](#_Toc185949964)

[Solution Summary 10](#_Toc185949965)

[Data Summary 10](#_Toc185949966)

[Overview 10](#_Toc185949967)

[Data Preparation 11](#_Toc185949968)

[Data Visualizations and Insights 12](#_Toc185949969)

[Machine Learning 15](#_Toc185949970)

[Logistic Regression: 15](#_Toc185949971)

[Random Forest: 17](#_Toc185949972)

[Final Results: 18](#_Toc185949973)

[User guide 19](#_Toc185949974)

## **Part A: Letter of Transmittal**

December 23, 2024

To: Emily Sullivan

DFCC Bank

500 Main Street, Columbia, SC

Dear Emily,

As you are well aware, our current transaction monitoring systems often generate excessive false positives, overwhelm investigators, and leave high-risk activities undetected. This results in inefficiencies and also exacerbates the massive backlog in case investigations that we currently have. To address these issues, I propose implementing a machine learning-based data product that assigns risk scores to financial transactions, enabling investigators to focus on the most critical cases.

The proposed solution leverages synthetic transaction data to train a supervised logistic regression model, producing transaction risk classifications. This data product will enhance compliance efforts and streamline investigation processes by automating the identification of suspicious activities. My team will implement the CRISP-DM methodology to ensure a standardized and speedy workflow.

I estimate the total cost of this project to be $32,500, including hardware, personnel, and cloud hosting expenses. The majority of the cost will be for the hardware and developer pay, as we will be using open-source libraries. The dataset used for this initiative will be a reputed dataset sourced from an online repository of datasets known as “Skaggle.” The data is completely synthetic and does not contain any personally identifiable information of our clients. The data will also be altered so it aligns with the bank’s internal transactions.

By investing in this initiative, we can significantly reduce false positives, prioritize high-risk transactions, and improve the efficiency of AML operations. Additionally, the solution will adapt to emerging financial crime patterns, ensuring its long-term utility.

As you know, I have an extensive background in financial crime investigations as well as a proficiency in machine learning; I am equipped to lead this project successfully, ensuring alignment with organizational goals and compliance standards.

Sincerely,

Pasindu De Silva

Pasindu De Silva – Senior Advisory Automation Manager

## **Part B: Project Proposal Plan**

### Project Summary

The global financial ecosystem faces significant challenges in combating money laundering and financial crime. Financial institutions, regulatory bodies, and governments must continuously innovate to identify and prevent illicit transactions.

As an employee of DFCC Bank, I have firsthand experience with the challenges faced by financial institutions in combating money laundering. Our bank handles a significant volume of transactions daily, leading to a massive backlog of cases flagged for manual review. This backlog slows the investigative process and increases the risk of delayed responses to genuine money laundering activities. The current rule-based systems, while useful, often generate numerous false positives, making it difficult for investigators to focus on high-priority cases effectively.

An ML solution is to be built that will leverage a synthetic dataset of transaction data to learn patterns associated with the risk of financial transactions. Using supervised learning algorithms such as logistic regression and random forests, the model will be trained on labeled data, where transactions are categorized as high-risk or low-risk based on historical money laundering patterns. By learning from these labeled examples, the model can assign risk scores to new transactions, predicting their likelihood of involving illicit activities.

The training process will involve feature engineering, where important transaction features such as transaction amounts, the parties involved, geographic location (originator and beneficiary), and industry sector are extracted. These features will allow the model to understand the factors contributing to high-risk behaviors, ensuring the model can generalize well when analyzing new data. The features will be preprocessed through techniques such as normalization and scaling to ensure that the model treats all data points fairly.

Once trained, the machine learning model will provide a risk score for each incoming transaction, indicating the probability that it is associated with money laundering. The executables and model will come with detailed documentation for future ML engineers should the model require alterations.

For it to be successful, we will have the required accuracy of the model to exceed 0.75 (with a macro average of 0.75 or higher).

For a full-scale model (if the project is successful), we can use our bank’s historical transaction data and risk decisioning to help train the model. Note that this is for the future, and the scope of this proof-of-concept project only involves synthetic data.

The risk score will reflect the likelihood of the transaction being associated with money laundering or other financial crimes, making AML/BSA investigators' jobs much easier as they wouldn’t have to sift through false positives and irrelevant transactions. With this solution, DFCC Bank will be at the forefront of leveraging advanced technology to combat financial crime effectively.

### Data Summary

The data will be sourced from Skaggle.com.

https://www.kaggle.com/datasets/waqi786/global-black-money-transactions-dataset

The synthetic dataset has 10,000 records with details on transaction IDs, countries, amounts, types, dates, etc. (Ali, 2022). This dataset was created by Ali (2022) and is publicly available for download and use. The dataset is pre-collected and publicly hosted on Kaggle. No additional data collection will be required, as this synthetic dataset was designed for analysis in financial transaction scenarios. The data used has no connection to our clients and is completely synthetic. Fortunately, there are no ethical or legal considerations.

The data will be downloaded from the site directly in CSV (comma delimited) format.

The data will undergo several processing steps throughout the application development lifecycle. During the design phase, I will remove any fields that I deem irrelevant to the machine learning model and data that would not exist (or be available) in our bank.

During development, I will handle missing or incomplete values by replacing them with appropriate placeholders or imputed values. The features will be encoded into numerical values and scaled for model optimization. Outliers will not be removed as they are numerous in real-time transaction data; hence, they should be part of the model.

The dataset will remain static, and maintenance will not be required for this proof-of-concept. However, provisions will be in place to update the dataset with real-world data in the future.

### Implementation

The development will follow the CRISP-DM methodology.

#### Business Understanding –

This project aims to automate the detection and prioritization of high-risk financial transactions to assist investigators in identifying potential money laundering activities. In this phase, discussions will be held with my colleagues in the bank, such as the BSA officer and the CAMLO (chief AML Officer), who oversees the AML program, to gather detailed requirements.

The primary business objectives include improving the efficiency of transaction reviews and ensuring compliance with AML regulations.

#### Data Understanding –

The data understanding phase involves collecting and exploring the dataset to gain insights into its characteristics, quality, and any potential issues that may arise during modeling. The dataset used is synthetic and open source and includes a variety of information, including transaction amounts, involved entities, dates, and geographical information. The data will be explored to identify patterns that could indicate high-risk transactions.

#### Data Preparation –

I will preprocess the data by normalizing continuous features like transaction amounts, transforming variables (e.g., country of origin) into numerical values, and performing feature engineering to create new features that may improve the model’s predictive power. For example, features like transaction frequency, amount ranges, and patterns of involved parties may be derived from the raw data to enhance the model’s ability to identify money laundering behaviors.

#### Modeling –

The modeling phase will apply machine learning algorithms to the prepared data to build a predictive model to assess transaction risk. Based on labeled training data, supervised logistic regression will be used to classify transactions as high-risk or low-risk. Multiple algorithms might be tested to identify the one that provides the best performance in accuracy and reliability.

#### Evaluation –

In the evaluation phase, the model’s performance will be assessed using several metrics to determine how well it classifies transactions. A confusion matrix will also be used to analyze the errors made by the model, such as false positives and false negatives, which is crucial for fine-tuning the model’s effectiveness. The evaluation will ensure that the model meets the predefined business objectives, such as achieving a high level of classification accuracy and minimizing false positives.

#### Deployment –

The final deployment phase involves integrating the trained model into the organization’s existing financial monitoring system. This will enable real-time or batch processing of transaction data, with the model providing risk scores for new transactions. The deployment will include creating a simple user interface for investigators to easily access and review flagged transactions and a backend integration that communicates with the financial institution’s data systems. Additionally, the system will be designed for ongoing updates, allowing the model to be retrained periodically with new transaction data to keep it relevant and effective in detecting emerging patterns of money laundering.

### Timeline

The proposed timeline spans from 1 January 2025 to 1 March 2025. This two-month timeframe is initially blocked out in the event of delays; however, it is likely that the project may be concluded in close to 40 days.

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestone or deliverable** | **Duration** | **Projected start date** | **Anticipated end date** |
| Gather requirements and collect datasets | 5 days | January 1, 2025 | |  | | --- | |  |  |  | | --- | | January 5, 2025 | |
| Perform data analysis on collected data | 5 days | |  | | --- | |  |  |  | | --- | | January 6, 2025 | | |  | | --- | |  |  |  | | --- | | January 10, 2025 | |
| Clean and preprocess data for training | 14 days | |  | | --- | |  |  |  | | --- | | January 11, 2025 | | |  | | --- | |  |  |  | | --- | | January 25, 2025 | |
| Build and train the machine-learning model | 5 days | |  | | --- | |  |  |  | | --- | | January 26, 2025 | | |  | | --- | |  |  |  | | --- | | January 30, 2025 | |
| Evaluate model performance and identify improvements | 5 days | |  | | --- | |  |  |  | | --- | | January 31, 2025 | | |  | | --- | |  |  |  | | --- | | February 4, 2025 | |
| Prepare for model deployment | 5 days | |  | | --- | |  |  |  | | --- | | February 5, 2025 | | |  | | --- | |  |  |  | | --- | | February 9, 2025 | |
| Deploy model and finalize documentation | 7 days | |  | | --- | |  |  |  | | --- | | February 9, 2025 | | |  | | --- | |  |  |  | | --- | | February 15, 2025 | |

### Evaluation Plan

* Exploratory data analysis will be conducted to check for any missing values and irrelevant outliers during the data understanding and preparation phase. Histograms, box plots, etc., would be used to glean this data.
* Evaluate the machine learning model using accuracy, precision, recall, and F1 score. The model will be considered successful if it achieves at least 75% accuracy and a macro average (precision, recall, and F1-score) of 0.75 or higher.
* Use the confusion matrix to identify false positives and false negatives. Focus on minimizing these errors to improve the model's ability to accurately classify high-risk transactions while minimizing false alerts by the existing TM system.
* Validate the model with a test set to assess its ability to generalize. Ensure that the model maintains strong performance on this unseen data. The model will be validated using an 80/20 train-test split, where 80% of the data is used for training and the remaining 20% is used for testing. The model will be tested on this unseen data to validate its generalizability.
* Build a Python program that can run on all common operating systems to work the data.

The model will be considered a success if it provides accurate risk predictions and is able to help investigators prioritize high-risk transactions efficiently, with minimal false positives. The evaluation will be based on the criteria mentioned above, and if the model demonstrates the desired accuracy (75%) and performance in real-world data, I will push for full-scale deployment.

### Resources and Costs

#### Hardware and Software –

Hardware – Ryzen 9 7950x3D processor + 32 gigabytes RAM

Computing infrastructure for model training and testing, costing $2,000.

Software –

* Windows 11
* Python 3.12 + Libraries
* Anaconda Navigator
* Jupyter Notebook
* PyCharm Professional + notebook integration
* Google Collab
* Machine learning frameworks (Pandas, NumPy) are available as open-source tools.
* Github for version control

No cost for any of the above (PyCharm Professional is already available to employees – no extra cost)

#### Labor Costs –

* Software Developer: Responsible for model implementation and data preprocessing, estimated at $60/hour for 300 hours, totaling $18,000.

#### Environment Costs –

* Cloud Hosting: AWS services for training, storage, and deployment, costing $500 per month.
* Dataset: Synthetic transaction dataset sourced from Skaggle, available for free.

## **Part C: Application**

The application is hosted on Google Collab. You will require a free Google account to access it.

<https://colab.research.google.com/github/CheMBurN695/FinCrime_ML_Model/blob/master/FC_ML_Model_NB.ipynb>

You will find my whole source code hosted on GitHub.

<https://github.com/CheMBurN695/FinCrime_ML_Model>

The dataset used is completely synthetic – no security features are required to protect the data. Furthermore, the machine learning model is a proof-of-concept – no maintenance of this model is necessary at this time.

## **Part D: Post-implementation Report**

### Solution Summary

In order to rectify a backlog of investigations in DFCC bank, a machine learning solution was built to leverage a synthetic dataset of transaction data to learn patterns associated with the risk of financial transactions. Using supervised learning algorithms, the model was trained on labeled data, where transactions are categorized as high-risk or low-risk based on historical money laundering patterns. By learning from these labeled examples, the model can assign risk scores to new transactions, predicting their likelihood of involving illicit activities.

Several machine learning algorithms were used to train the data, and the performance of each of them was evaluated for their performance. The best performance was yielded by using a random forest algorithm, which exceeded our required metrics with reasonably high accuracy.

### Data Summary

#### Overview

The dataset used for the application was a synthetic dataset sourced from Skaggle with 10,000 transactions, each having the following fields:

* Transaction ID
* Country
* Amount (USD)
* Transaction Type
* Date of Transaction
* Person Involved
* Industry
* Destination Country
* Reported by Authority
* Source of Money
* Money Laundering Risk Score
* Shell Companies Involved
* Financial Institution
* Tax Haven Country

As noted in the proposal, all transactions here are synthetic and have no bearing on any real individual, containing no names or identifiable information. However, the transactions and patterns are indicative of normal banking transactions.

#### Data Preparation

The data was downloaded from the site directly in CSV (comma delimited) format. All of the fields above were populated, and no null values or unintended whitespace was found.

To suit the study and to align more with the transactions we have at DFCC Bank, the data underwent several alterations.

I added a field to the data called “Risk\_Classification.” This is merely a binary value, 0 and 1, denoting a “Low Risk” and “High Risk” transaction. The value is derived from the Money Laundering Risk Score value (which is between 0 and 10), where anything above a risk score of 4 would be classified as high risk and anything below that as low risk.

The following fields were removed:

* Tax Haven Country – the country in this field does not appear to have any correlation to the stated transaction, and the dataset does not provide a good explanation of the relevance of the data in this column.
* Reported by Authority – we are not able to identify reporting data during model training when using real-world data due to legal and compliance reasons. Hence, this column was removed.

The above two fields were removed completely from the main dataset and will not be included in the study. However, this being said, not all fields above will be used to train the model (to ensure that the model is valid).

The data was then split into ‘training’ and ‘testing’ parts with an 80:20 ratio. I used a small Python script to the above (included in the repository), which shuffles the rows from the original dataset and writes to two separate CSV files (https://github.com/CheMBurN695/FinCrime\_ML\_Model/blob/master/Data\_Splitter.py).

The training data contains 8000 transactions, and the testing data contains 2000 transactions.

#### Data Visualizations and Insights

A graph of different colored bars

Description automatically generatedSeveral visualizations were drawn up to better understand the dataset and to gauge the suitability of the data.

The above stacked-bar chart shows a good distribution of countries/transactions where the source of funds was deemed illegal.

A graph of a number of blue bars

Description automatically generated with medium confidence

A graph with blue squares

Description automatically generated with medium confidence

The above histogram shows a normal distribution of risk scores within the dataset. This is indicative of a real-world scenario and would assist with the model’s validity. In terms of risk classification, a majority of the transactions are legal transactions, and illegal transactions account for roughly 30% of the total dataset – this distribution too closely follows what is expected from post-AML monitoring transactions at the bank.

A pie chart with different colored circles

Description automatically generated

The pie chart above shows a good equal distribution of transaction types in the dataset, which will help reduce sample bias.

Using pandas, a data frame was taken, and the data was encoded (using Scikit-Learn LabelEncoder) into numerical values. The data was used to draw up a correlation heatmap to identify whether any immediate correlations show up within the data.

As shown above, there is generally a very weak correlation between the features across the board. The high correlation between the Money Laundering Risk Score and Risk Classification is artificial (as I mentioned above, it is merely [if (riskScore > 5) { riskClass = 1 } else {riskClass = 0}]). There is also an (above average) correlation between the number of shell companies involved and the risk score, which is expected.

### Machine Learning

Following data preparation, the following machine learning methods were used.

All data was encoded using LabelEncoder, and the following features were used for the variables.

Independent variables (X) = ['Country,' 'Destination Country,' 'Industry,' 'Shell Companies Involved,' 'Transaction Type']

Target Variable (Y) = ['Risk\_Classification']

I did not use ‘Money Laundering Risk Score’ or ‘Source of Money’ to avoid overfitting.

These variables were used for all the machine-learning algorithms detailed below.

#### Logistic Regression:

Logistic regression was the first method used, as our classification is binary and dichotomous in nature. We simply need to classify high risk as low risk.

Using Sckit-Learn’s logistic regression library, I fit the model using the variables and predicted the outcomes. The maximum iterations for the model instance was set to 1000.

The confusion matrix and the classification report of the results are shown below:

A screenshot of a computer screen

Description automatically generatedA screenshot of a computer

Description automatically generated

The model's overall accuracy is 0.66, meaning it correctly classified 66% of all instances. This, unfortunately, was a bit below our requirements as we aimed for 75% accuracy. The model performs better at identifying Low-Risk transactions but struggles with High-Risk cases. It has a high recall for Low Risk but a low recall for High Risk, suggesting it tends to classify more instances as Low Risk.

#### Random Forest:

The next model used to train the data was a random forest. Considering the number of features and values as well as the high dimensionality in the dataset, I felt a random forest algorithm might be more suitable to train it.

Using Sckit-Learn’s random forest classifier library, I fit the model using the variables and predicted the outcomes. The number of estimators used was 300, and the last random seed used was 420.

A screenshot of a computer screen

Description automatically generatedThe confusion matrix and the classification report of the results are shown below:

A screenshot of a computer

Description automatically generated

As you can see, the random forest learning model was much more successful at classifying transactions than logistic regression. The classification report indicates a generally good performance of the model, with an overall accuracy of 0.7915 (79.15%). The model correctly classified roughly about 4 out of 5 transactions in the dataset.

The model shows roughly a 20% improvement over logistic regression.

The macro average scores (0.77 for precision, recall, and F1-score) show that the model performs consistently across both classes when giving equal weight to each class.

The model demonstrates good predictive capability, more for low-risk transactions while maintaining reasonable performance on high-risk transactions and certainly exceeding both our targets of 75% accuracy and 0.75 macro average.

### Final Results:

The machine learning models implemented in this study demonstrated varying levels of effectiveness in classifying financial transactions as high-risk or low-risk.

Logistic regression was chosen as the initial model due to its suitability for binary classification tasks. The model achieved an overall accuracy of 66%, correctly classifying two-thirds of the transactions in the dataset. While the model performed well in identifying low-risk transactions, it struggled with high-risk cases, as evidenced by its high recall for low-risk and low recall for high-risk transactions.

A random forest classifier was subsequently employed to address the limitations observed with logistic regression. The random forest model significantly outperformed logistic regression, achieving an overall accuracy of 79.15%, exceeding the target threshold of 75%. Furthermore, the macro average scores for precision, recall, and F1-score were all 0.77, indicating balanced performance across both classes, even when weighted equally.

The random forest classifier successfully met the bank's requirements for an accurate classification model for financial transactions. By achieving a 79.15% accuracy and consistent macro average metrics, it demonstrated its capability to classify transactions effectively.

### User guide

Step 1: If you do not have one, create a free Google account

Step 2: Navigate to the Google Collab URL (and sign in to your Google account)- <https://colab.research.google.com/github/CheMBurN695/FinCrime_ML_Model/blob/master/FC_ML_Model_NB.ipynb#scrollTo=b849f199acd4101d>

Step 3: Navigate to the bottom of the page.

Step 4: Click on Runtime -> Run all (or alternatively use the default shortcut Ctrl+F9)

A screenshot of a computer program

Description automatically generated

Step 5: Wait until the program runs, and the widgets UI should appear below.

Step 6: Use the dropdowns of the widget to select the details of the transaction you wish to classify. Press the Predict Risk button. The risk classification should populate below the button.

A screenshot of a computer

Description automatically generated