UTM
UNIVERSITI TEKNOLOGI MALAYSIA

MASTER PROJECT

MONEY LAUNDERING TRANSACTION DETECTION USING MACHINE LEARNING ALGORITHMS

PRESENTED BY:
NUR ADRIANA BATRISYIA BINTI MOHD SUBRI (MCS241007)

SUPERVISOR: DR NOR HAIZAN BINTI MOHAMED RADZI

Innovating Solutions

CONTENT OUTLINES

01 INTRODUCTION

02 LITERATURE REVIEW

03 RESEARCH METHODOLOGY

04 DATA PREPARATION

05 MODEL DEVELOPMENT AND EVALUATION

06 DASHBOARD DEVELOPMENT

07 CONCLUSION

UTM
UNIVERSITI TEKNOLOGI MALAYSIA

CHAPTER 1

INTRODUCTION

MONEY LAUNDERING

- A serious global issue since the 1980s. It is recognized as a **threat to economic stability and security worldwide.**

- **Linked to crimes** like fraud, drug trafficking – to disguise the origin of illegal funds

- Lead to **higher crime rates, threaten national security risks, increase corruption**

- Financial institutions are **under pressure to detect and prevent money laundering**

- Traditional AML systems are **rule-based, requiring frequent updates and human review**

- As laundering techniques become more complex, traditional approach is no longer sufficient. Hence, the need to integrate **machine learning** into AML efforts.

PROBLEM BACKGROUND

Money laundering involves **three stages**:

🟢 Placement → 🔄 Layering → 💰 Integration

GLOBAL MONEY LAUNDERING

(source: UNDOC & Global Financial Crime Report 2023)

- **2%-5%** of world's GDP laundered annually
- In 2023, around **$3.1 trillion** are laundered globally fueling various crimes such as:
  - **$346.7 billion** in human trafficking
  - **$782.9 billion** in drug trafficking
  - **$11.5 billion** in terrorist financing

According to BNM Annual Report 2023,

- **317,435** Suspicious Transaction Reports (STRs) was recorded
  - Key Offences: **Fraud, money laundering, tax evasion**
  - Over **100 individuals arrested and RM290 millions seized**
  - Detected and deactivated **59,684 mule accounts**

Based on Switzerland's 2015 National Risk Assessment,

- **only 511 out of 47,000** suspicious transactions triggered SARs in which **99% were false positives**

Most financial institutions still rely on **rule-based systems**, which generate **many false positives** and overwhelm investigators. To improve this, financial institutions are now starting to adopt **machine learning.** Studies show that:

- **Supervised learning models like Random Forest** can achieve over **94% accuracy and F1-score**, significantly outperforming traditional methods.
- For unlabeled data, **unsupervised models like Isolation Forest** effectively detect transaction anomalies.
- Analyzing **feature importance** helps make these models more transparent and justifiable in real-world financial settings.

PROBLEM STATEMENT

Millions of transactions every seconds make manual monitoring impractical.

Rule-based system generates **excessive false positives** which overwhelm compliance teams and slowing down real investigations

Machine learning promises better accuracy and fewer false alarms, but challenges remains:

- **Lack of accessibility to real datasets**
- **Limited interpretability of models reduces trust and adoption**

Therefore, there is a need to develop intelligent and adaptive machine learning systems for large-scale transaction analysis to accurately classify suspicious activities with low false-positive rates and examine key predictive features.

RESEARCH OBJECTIVES

01 **To perform data preprocessing and exploratory data analysis (EDA)** to understand data distributions and transaction patterns.

02 **To develop classification models** to predict suspicious transactions and conduct comprehensive evaluation on the performance of the models.

03 **To develop interactive dashboard** to present insights and findings related to money laundering.

SCOPE OF THE PROJECT

DATASET

- Synthetic Anti-Money Laundering Dataset (SAML-D)

SUPERVISED MACHINE LEARNING ALGORITHMS

- XGBoost
- Random Forest

TOOLS

- Google Colab
- Microsoft Power BI

SIGNIFICANCE OF RESEARCH

The outcomes from this project may contribute in the effort to combat money laundering:

- ➢ A well-trained model that can detect money laundering especially in complex scenarios can **strengthen financial institution's defence against money laundering**
- ➢ Model with low false positive rates can reduce unnecessary alerts and help authorities to focus on truly suspicious cases hence **improve the efficiency of enforcement agency to handle money laundering cases**
- ➢ **Protect the integrity of global financial system** through an innovative and data-driven approach to money laundering detection.

UTM
UNIVERSITI TEKNOLOGI MALAYSIA

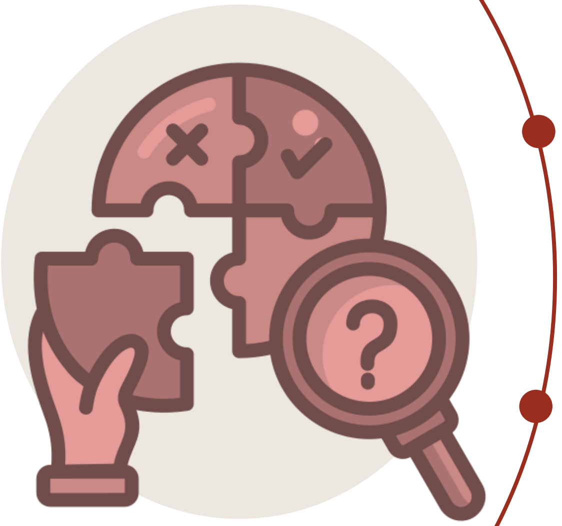
CHAPTER 2

LITERATURE REVIEW

PREVIOUS STUDIES

No.	Author	Method	Dataset	Highlights
1.	**Tundis et al. (2021)**	Decision Tree, Support Vector Machines, Random Forest (RF), Linear Regression, Naïve Bayes	PaySim (Synthetic)	- RF achieved >94% in accuracy, recall, F1; low false positives - Included analysis on feature importance
2.	**Zhang & Trubey (2019)**	Bayes Logistic Regression, Decision Tree, Random Forest, Support Vector Machines, Artificial Neural Network (ANN)	US Bank Data (Real)	- Handled data imbalance - ANN performs best
3.	**Reite et al. (2024)**	XGBoost	Norwegian Bank SME's clients (Real)	- Improved client risk classification while reducing false alarms
4.	**Yang et al. (2023)**	Combination of Long Short Term Memory and Graph Convolutional Neural Network	Elliptic Bitcoin (Synthetic)	- Effective for classifying Bitcoin-related anomalies
5.	**Labanca et al. (2022)**	Combination of Random Forest and Isolation Forest	Capital market (Synthetic)	- Active learning improves anomaly detection & reduces compliance cost
6.	**X. Luo et al. (2022)**	Neural Network	Dataset from Law Enforcement Agency (Real)	- Captures transaction behaviors & transfer evolutions
7.	**Cheng et al. (2023)**	Neural Network	UnionPay (Real)	- Detects organized laundering via user transaction graphs
8.	**J. Luo et al. (2024)**	Edge-Node Fusion	AMLSim (Synthetic)	- Improves illicit edge prediction using advanced training
9.	**Pambudi et al. (2019)**	Support Vector Machines	PaySim (Synthetic)	- F1-score ↑ 22.8%, precision ↑ 40.8% with RUS for imbalance

RESEARCH GAPS



Limited real-world dataset on money laundering

- Due to privacy & security concerns
- Use synthetic dataset that simulates realistic behaviors while protecting sensitive information

Imbalance transaction dataset

- Can lead to biased predictions
- Apply imbalanced-data handling method

Lack of research experimented on typologies of money laundering

- Most only focus on behavioral features
- Plan to include typology features to enhance detection

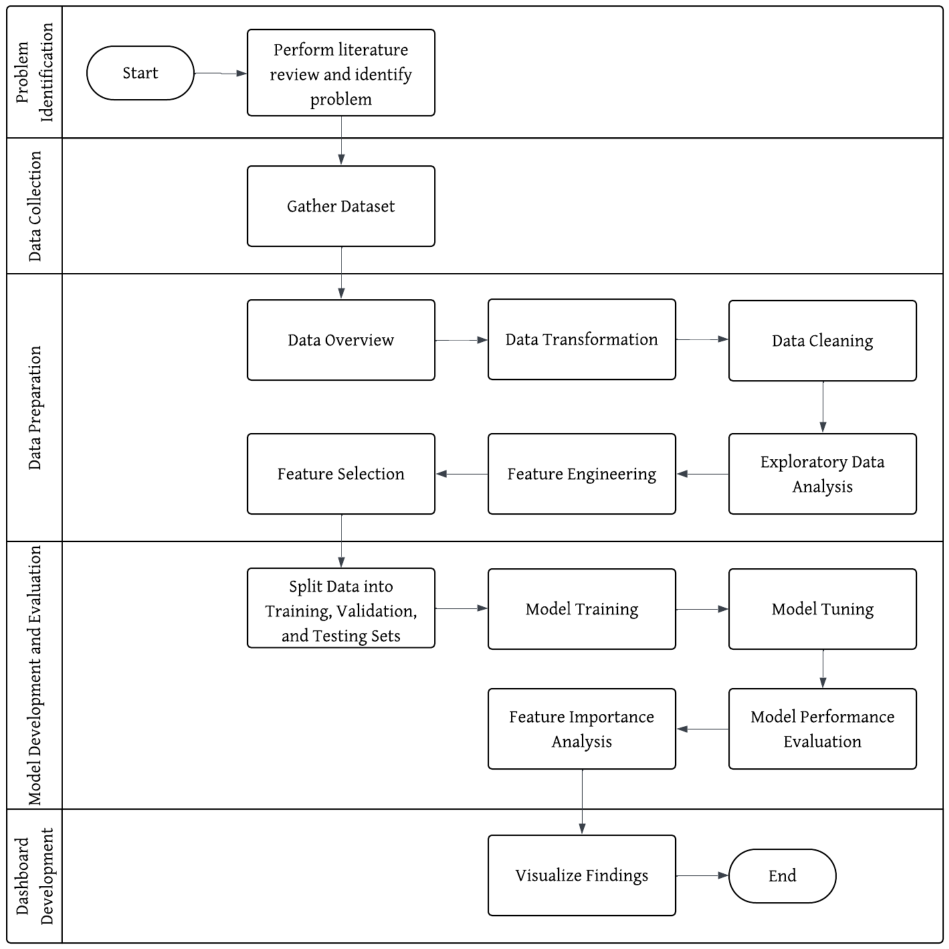
Lack of interpretability of the model used in money laundering detection

- Many models are black boxes, accurate but not explainable.
- Partially address the gap by conducting feature importance analysis to identify key features that contribute to the model's decisions.

CHAPTER 3

RESEARCH METHODOLOGY

PROJECT LIFECYCLE



Problem Identification

Data Collection

Data Preparation

Model Development and Evaluation

Dashboard Development

Start

Perform literature review and identify problem

Gather Dataset

Data Overview

Data Transformation

Data Cleaning

Exploratory Data Analysis

Feature Engineering

Feature Selection

Split Data into Training, Validation, and Testing Sets

Model Training

Model Tuning

Model Performance Evaluation

Feature Importance Analysis

Visualize Findings

End

1 PROBLEM IDENTIFICATION

- ➢ Understand the current challenges in money laundering detection through literature review

2 DATA COLLECTION

- ➢ Obtain dataset for model development

3 DATA PREPARATION

- ➢ Data Overview : No. of attributes, data types, categories for each features
- ➢ Data Transformation: Convert data type for certain columns (e.g: integer → string) or derive new features from existing feature (e.g: ‘Day_of_week’ and ‘Is_weekend’ are derived from ‘Date’ column)
- ➢ Data Cleaning: Check for missing and duplicate data and remove irrelevant columns
- ➢ Exploratory Data Analysis (EDA) : identify the patterns, and trends in the dataset
- ➢ Feature Engineering: Label Encoding
- ➢ Feature Selection: Correlation Analysis

4 MODEL DEVELOPMENT AND EVALUATION

- ➢ Split Train-Test: 70% Training, 15% Validating, 15% Testing
- ➢ Train using Random Forest and XGBoost
- ➢ Tune hyperparameter and adjust threshold to improve the model
- ➢ Evaluate model using precision, recall, F1-score and accuracy
- ➢ Perform feature importance analysis

5 DASHBOARD DEVELOPMENT

- ➢ - Visualize findings using interactive dashboards

CHAPTER 4

DATA PREPARATION

DATASET

DATASET

- Synthetic transaction dataset called SAML-D
- Produced by research paper entitled ‘Enhancing Anti-Money Laundering: Development of a Synthetic Transaction Monitoring Dataset’ by Berkan Oztas et al. in 2023
- Publicly available in Kaggle

Size	9,504,851 rows
Attributes	12 Attributes 1) Time 2) Date 3) Sender_account 4) Receiver_account 5) Amount 6) Payment_currency 7) Received_currency 8) Sender_bank_location 9) Receiver_bank_location 10) Payment_type 11) Is_laundering 12) Laundering_type

DATA TRANSFORMATION

DERIVED NEW FEATURES

1. DateTime	Combine Date and Time to obtain timestamp of the transaction to derive new features related to time.
1. TimeOfDay	Indicates whether it is ‘Day’ (6AM to 6PM) or ‘Night’ 6PM to 6 AM.
2. Is_Weekend	Assign ‘1’ if weekend, ‘0’ if weekday.
3. Tx_per_Day	Daily transaction frequency per sender.
4. Tx_Interval	Transactions interval from previous transactions grouped by sender (in sec).
5. Avg_Amount_Sent	Average transaction amount per sender.
6. Pair_bank_location	Combine Sender Bank Location and Receiver Bank Location.
7. Pair_currency	Combine Payment Currency and Received Currency.

DATA TRANSFORMATION

CATEGORICAL SIMPLIFICATION

‘Laundering_Structure’ (New Feature)	‘Laundering_type’ (Original Feature)
1. Cash_Deposit	Normal_Cash_Deposits
2. Fan-Out	Normal_Fan_Out, Normal_Small_Fan_Out, Layered_Fan_Out, Fan_Out
3. Fan-In	Normal_Fan_In, Layered_Fan_In, Fan_In
4. Cash_Withdrawal	Normal_Cash_Withdrawal, Cash_Withdrawal
5. Behavioral_Group	Normal_Group, Behavioural_Change_1, Behavioural_Change_2
6. Mutual	Normal_Mutual, Normal_Plus_Mutual
7. Forward	Normal_Foward, Deposit-Send
8. Periodical	Normal_Periodical
9. Structuring	Smurfing, Structuring
10. Cycle	Cycle
11. Bipartite	Stacked Bipartite, Bipartite
12. Scatter-Gather	Scatter-Gather
13. Gather-Scatter	Gather-Scatter
14. Single_Large	Normal_single_large, Single_large
15. Over-Invoicing	Over-Invoicing

DATA CLEANING

Missing and Duplicated Data

- No duplicated data but has 292,715 missing values in column Tx_Interval
- Because it is the first transaction for the account, hence it has no previous transaction to compute the interval.
- Therefore, missing values are imputed with ‘0’

Remove Irrelevant Columns

- Sender_account
- Receiver_account
- Date
- Time
- DateTime
- Payment_currency
- Received_currency
- Sender_bank_location
- Receiver_bank_location
- Laundering_type

Because it have been used to derive new features. The info already captured in derived features.

FEATURE ENGINEERING

Label Encoding: Transform categorical variables into numerical variables

- Payment_type
- TimeOfDay
- Pair_bank_location
- Pair_currency
- Laundering_Structure

Example:

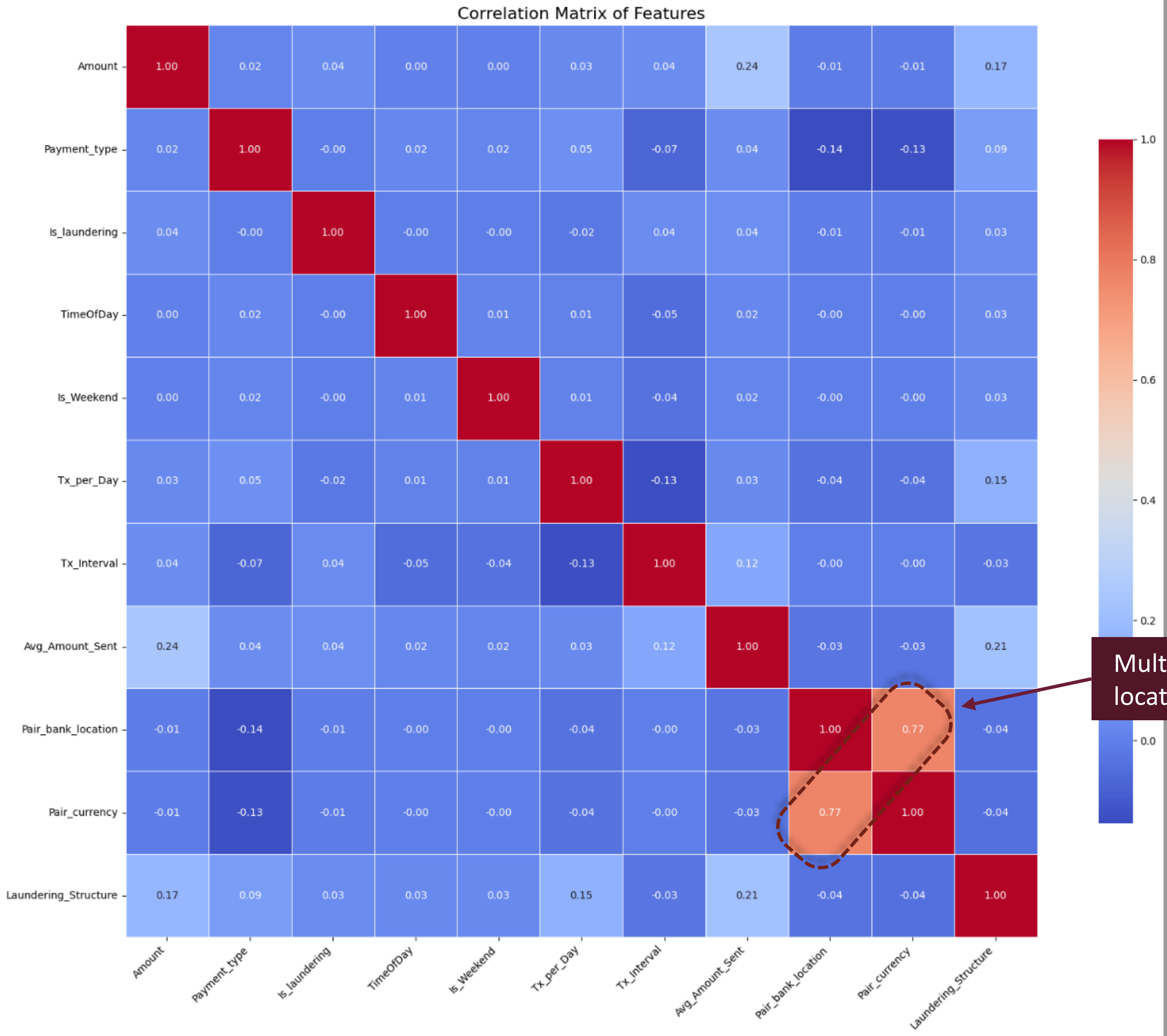
Pair Bank Location
UK - UK
UK – Morocco
UK - Nigeria
UK - Albania
UK -Mexico

→

Pair Bank Location
304
296
298
288
295

Even though it is not ordinal data, label encoding is selected due to its high cardinality (324 total pairs)

FEATURE SELECTION



Multicollinearity exist between Pair bank location and Pair currency (>0.75)

Correlation with target (Is_laundering)	
1. Avg_Amount_Sent	0.0431
2. Amount	0.0401
3. Tx_Interval	0.0360
4. Laundering_Structure	0.0321
5. Tx_per_Day	-0.0188
✗ 6. Pair_currency	-0.0105
7. Pair_bank_location	-0.0065
8. Payment_type	-0.0017
9. TimeOfDay	-0.0015
10. Is_Weekend	-0.0011

All features have very weak linear correlations with target feature (<0.05)

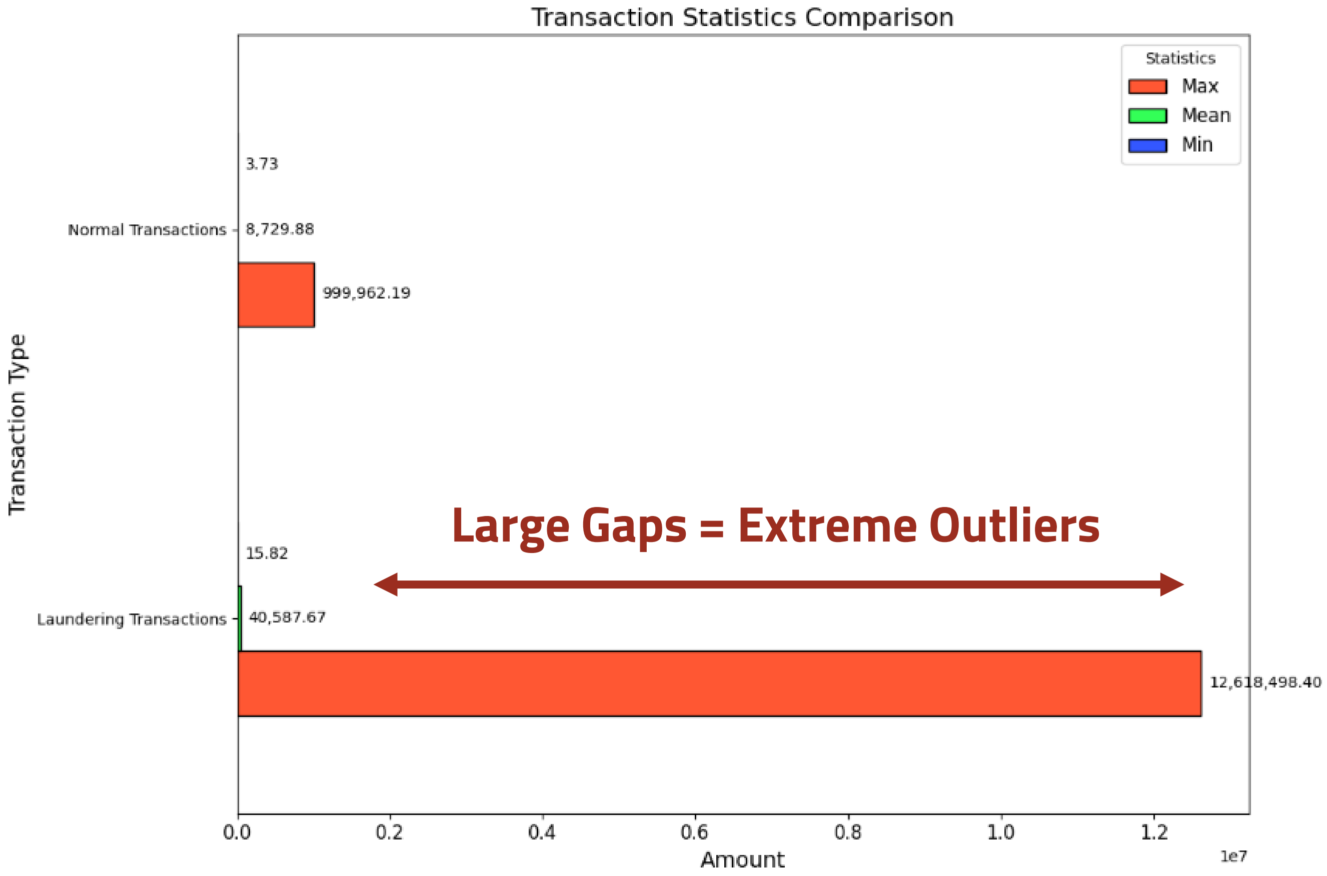
Remain all features except for **Pair_currency** due to multicollinearity.

Therefore, the model is now has 9 independent variables (x) and 1 dependent variable ready for training.

The dependent/target variable:
Is_laundering where ‘0’ is normal transaction and ‘1’ suspicious transaction

EXPLORATORY DATA ANALYSIS

1 Identify Min, Max & Mean for Laundering and Normal Transactions



Large Gaps = Extreme Outliers

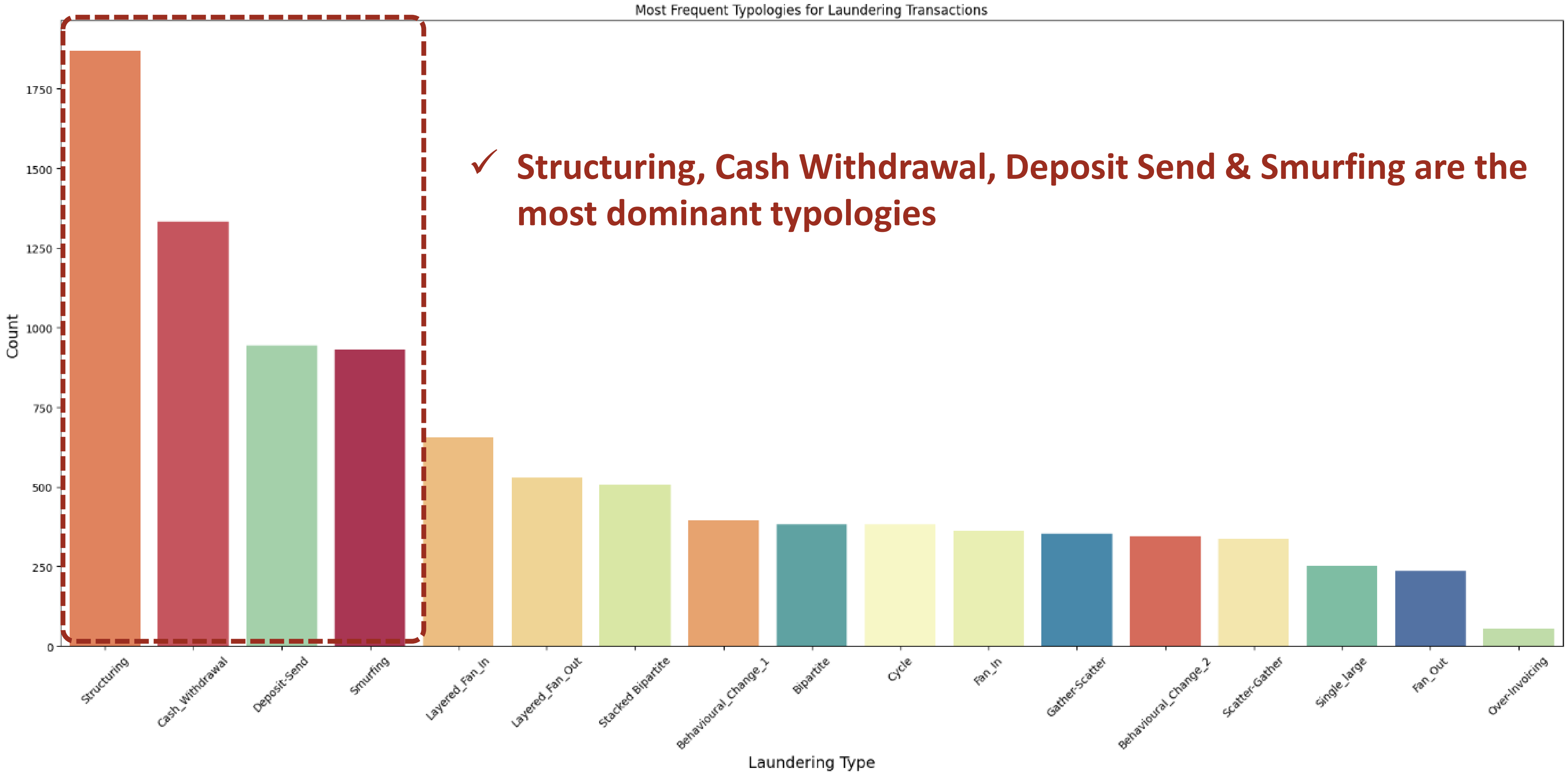
- ✓ Maximum amount in laundering transactions is significantly higher than normal transactions
- ✓ Both transactions have extremely small minimum amount

- ✓ **Laundering transactions often involve extreme values**

EXPLORATORY DATA ANALYSIS

UTM UNIVERSITI TEKNOLOGI MALAYSIA

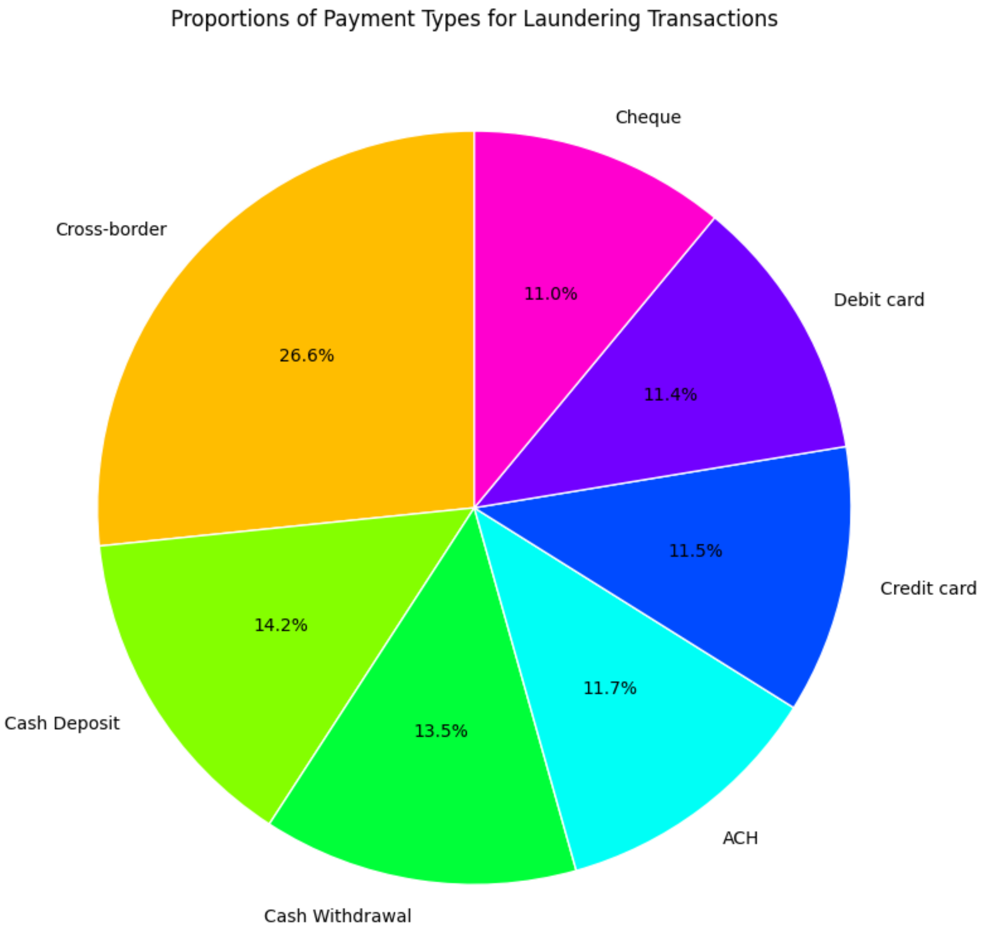
2 Identify Most Frequent Typologies for Laundering Transactions



✓ **Structuring, Cash Withdrawal, Deposit Send & Smurfing are the most dominant typologies**

www.utm.my

3 Identify Most Frequent Payment Types for Laundering Transactions

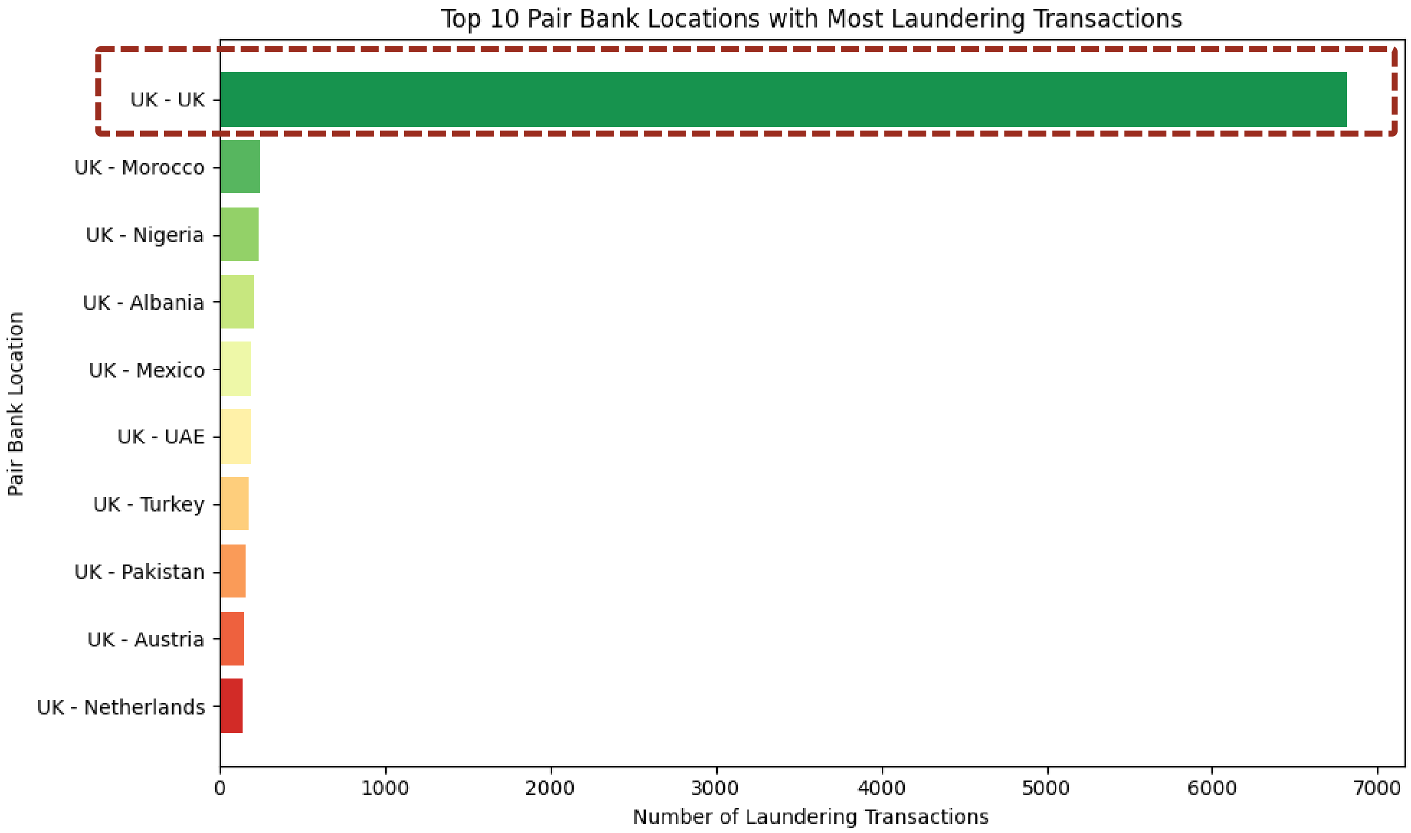


- ✓ Cross-border transactions has the largest proportions followed by Cash Deposit and Cash Withdrawal
- ✓ ACH, Credit Card, Debit Card, and Cheque have relatively similar proportion

- ✓ **Cross-border transactions is the most preferred payment method by launderers**

EXPLORATORY DATA ANALYSIS

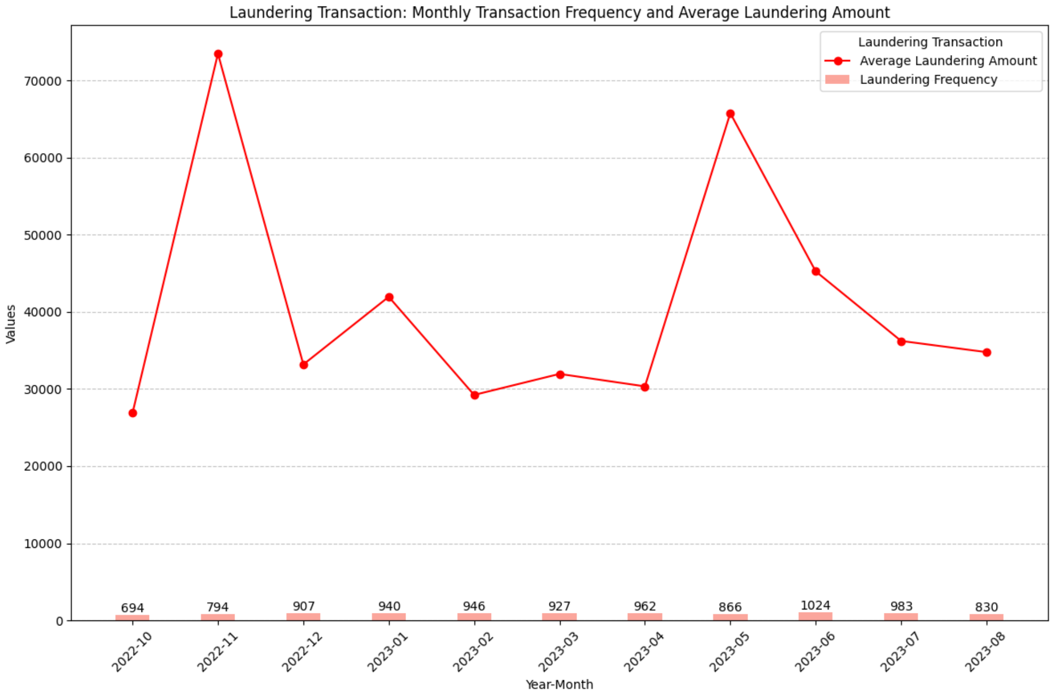
4 Identify High Risk Pair Bank Locations

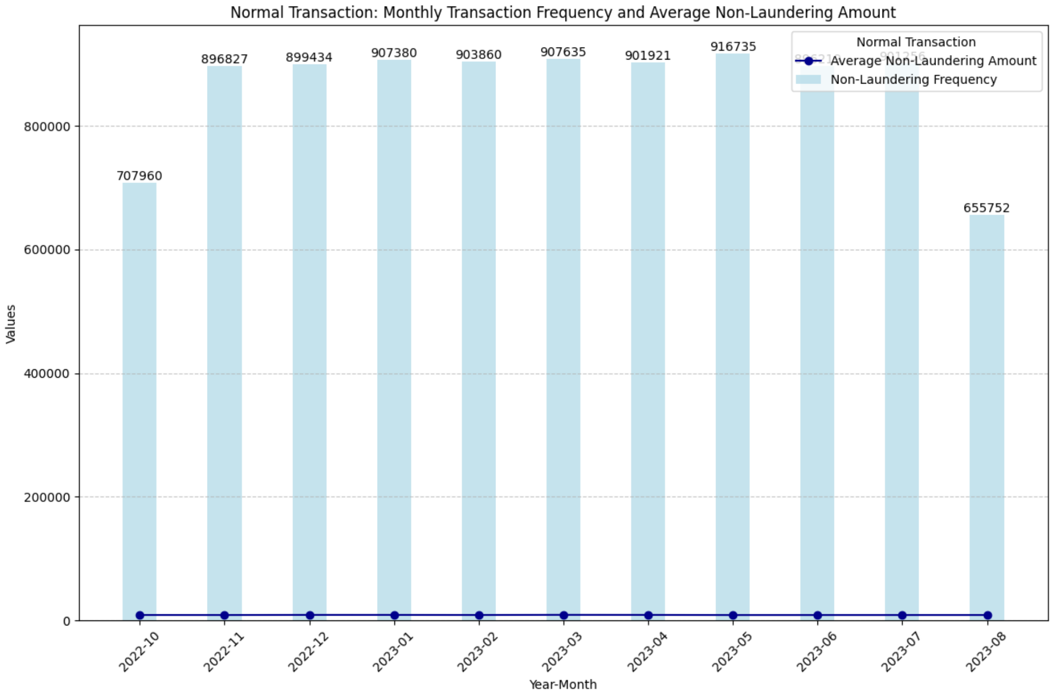


- ✓ **UK is the most high-risk pair bank locations as it seems to be a central hub for both sending and receiving illicit money from laundering transactions**

EXPLORATORY DATA ANALYSIS

5 Identify Monthly Transaction Frequency and Average Laundering Amount by Transaction Type


Laundering Transaction: Monthly Transaction Frequency and Average Laundering Amount


Normal Transaction: Monthly Transaction Frequency and Average Non-Laundering Amount

✓ **The sharp contrast in frequency and amount emphasize that laundering transactions occurrence are rare but usually involve larger amounts of money.**

UNIVERSITI TEKNOLOGI MALAYSIA

CHAPTER 5

MODEL DEVELOPMENT AND EVALUATION

SPLIT TRAIN-VALIDATE-TEST DATASET USING STRATIFIED SAMPLING

UTM
UNIVERSITI TEKNOLOGI MALAYSIA

Ratio : 70 : 15 : 15

Training Set	Validation Set	Testing Set
6,657,198 records	1,421,926 records	1,425,728 records

Default Parameters:

Random Forest	XGBoost
(a) n_estimators =100	(a) n_estimators = 100
(b) max_depth = None	(b) max_depth = 6
(c) min_samples_split = 2	(c) learning_rate = 0.3
(d) random_state = 42	(d) subsample = 1
(e) n_jobs = -1	(e) random_state = 42
	(f) n_jobs=-1

Sampling method:

Stratified sampling was selected as sampling method throughout this project ensuring the class proportions are preserved across training, validation, and testing set.

		Precision	Recall	F1-Score	Accuracy
Random Forest	Random Sampling	0.9829	0.7033	0.8199	0.9997
	Stratified Sampling	0.9894	0.6919	0.8143	0.9997
XGBoost	Random Sampling	0.8412	0.6327	0.7222	0.9995
	Stratified Sampling	0.8923	0.6845	0.7747	0.9996

Random Forest:

- Minor difference between strategies.
- Show stable performance regardless of different strategies.

XGBoost:

- Stronger dependence on sampling strategy.
- More balanced performance using stratified sampling.

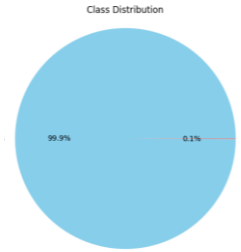
Therefore, the result from stratified sampling with default setting become the **baseline model** to compare throughout the project.

www.utm.my

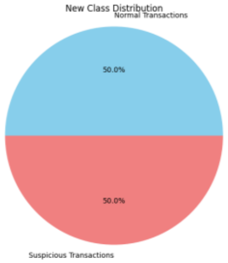
HANDLING CLASS IMBALANCE (during model training only)

Random Under Sampling

Reducing the majority samples (normal transactions) to match the size of minority class (laundering transactions).



Tranining Set before RUS



Training Set after RUS

	Tranining Set before RUS	Training Set after RUS
Normal Transactions	6,650,283	6,915
Suspicious Transactions	6,915	6,915

Class Weighting

Assign higher weight to minority class so that the model penalizes more severely on the misclassification of suspicious transactions

Random Forest

- Parameter: class_weight = ‘balanced’
- Automatically compute class weight:

Normal = 0.50
Suspicious = 481.43

$$w_c = \frac{N}{K \cdot n_c}$$

- Suspicious transactions were treated 481 times more important than normal transactions during splitting in tree construction

XGBoost

- Parameter: scale_pos_weight
- Value computed using formula:

Total Majority / Total Minority = 961.71

- Suspicious transactions are 962 times rarer than normal transactions.

Result Comparison:

		Precision	Recall	F1-Score	Accuracy
Random Forest	**Baseline**	0.9894	0.6919	0.8143	0.9997
	With RUS	0.0214	0.9479	0.0419	0.9550
	With Class Weight	0.9929	0.6669	0.7979	0.9996
XGBoost	**Baseline**	0.8923	0.6845	0.7747	0.9996
	With RUS	0.0234	0.9553	0.0456	0.9585
	With Class Weight	0.0648	0.9242	0.1211	0.9861

Random Under Sampling:

- Degraded both model performance (based on F1-score)
- Recall increase from 0.6 to 0.9, but precision drop significantly

Class Weighting:

- **XGBoost**: Very high recall, very low precision. (Drastic effect)
- **Random Forest:**. Maintain high precision, with slightly decline in recall. Still less balance than baseline.

Therefore, these imbalanced handling methods will not be applied in the next stage as the result is not satisfactory.

2-STAGE MODEL OPTIMIZATION

Tune Hyperparameter using Randomized SearchCV

- Control how the model learn form the data and to balance overfitting and underfitting.
- To reduce computational cost:
  - Use subsample of 200k from training dataset with stratified sampling. Then, refitted on the full training dataset once the best hyperparameters are identified.
  - Use Randomized Search with 3-fold CV. Randomly select a subset of hyperparameter combination from the parameter grid.

- **Best hyperparameters:**

Random Forest	XGBoost
(a) n_estimators =100	(a) n_estimators = 100
(b) max_depth = None	(b) max_depth = 6
(c) min_samples_split = 2	(c) learning_rate = 0.1
	(d) subsample = 1

- **Result after Tuning:**

	Precision	Recall	F1-Score	Accuracy
Random Forest	0.9894	0.6919	0.8143	0.9997
XGBoost	0.9614	0.6750	0.7932	0.9996

Same result as baseline.
Default parameter = Best parameter

Result improved than baseline

Random Forest achieved slightly better F1-score, fewer false positives and fewer misclassifications

Adjust Threshold

- By default, classifiers algorithm use threshold = 0.5 on the predicted probability to decide the class.
- Evaluate different probability threshold between 0.1 to 0.9 to select the best threshold that maximize F1-score.

- **Best Threshold:**

  Random Forest = 0.24
  XGBoost = 0.28

- **Result after Tuning & Adjust Threshold:**

	Precision	Recall	F1-Score	Accuracy
Random Forest	0.9185	0.7630	0.8336	0.9997
XGBoost	0.9236	0.7041	0.7991	0.9996

- Adjusting threshold below 0.5 makes the model classify more true suspicious case, hence the increase in recall. Even the precision have slightly dropped, it still above 0.90. Hence, the F1-Score improved for both model.

- Threshold tuning also proven to handle imbalanced dataset. High accuracy in this model is due to dominance of majority class. Hence, adjusting threshold allowed the models to make better trade-offs between detecting suspicious transactions and minimizing false positives

FINAL EVALUATION

Retrained models on the combined Training+Validation Set using Best Hyperparameters → Predict the Testing Set using Best Probability Threshold

Final Result:

Random Forest

	Precision	Recall	F1-Score	Accuracy
Training Set	0.9978	1.0000	0.9989	1.0000
Validation Set	0.9185	0.7630	0.8336	0.9997
Testing Set	0.9113	0.7698	0.8346	0.9997

- Nearly perfect performance in Training Set.
- Slightly lower performance in Validation & Testing Set.
  - Not much difference in the performance.
  - Precision > 0.91, Recall > 0.76, F1-Score > 0.83
  - Effective in minimizing false alarms but less effective in capturing all suspicious transactions.
- Consistent and reliable result in validation and testing prove that the model can generalize well

XGBoost

	Precision	Recall	F1-Score	Accuracy
Training Set	0.9340	0.7138	0.8092	0.9997
Validation Set	0.9236	0.7041	0.7991	0.9996
Testing Set	0.9261	0.6935	0.7931	0.9996

- The model produced consistent result with minor variations throughout training until testing. Maintaining:
  - Precision (0.92-0.93), Recall (0.69-0.71),F1-Score (0.79-0.80)
- Performs well in reducing false alarms but a bit struggle to capture all true suspicious (High Precision Low Recall)
- Model generalizes well and not overfitting.

Random Forest outperform XGBoost in terms of recall and F1 Score across validation and testing set. Hence, it becomes the preferred model since it offers overall effectiveness in detecting money laundering transactions

FEATURE IMPORTANCE ANALYSIS BASED ON RANDOM FOREST

Feature Importance

- Represents the contributions of each feature to the model‘s predictions.
- In Random Forest, it is calculated based on how much each feature decreases the Gini impurity across all decision trees.
- Features that consistently create better split have higher importance score.

Feature	Importance Score
1. Laundering_Structure	0.4666
2. Amount	0.1413
3. Avg_Amount_Sent	0.1386
4. Tx_Interval	0.1143
5. Payment_type	0.0710
6. Pair_bank_location	0.0291
7. Tx_per_Day	0.0253
8. Is_Weekend	0.0079
9. TimeOfDay	0.0059

Most Influential Features

Least Influential Features

Random Forest is most sensitive to structural and monetary characteristics of transactions while temporal features have lesser contribution to the overall prediction.

It corresponds with expectation in money laundering where structuring and manipulation of transaction value are key red flags

UTM
UNIVERSITI TEKNOLOGI MALAYSIA

CHAPTER 6

DASHBOARD DEVELOPMENT

DATA PREPARATION FOR DASHBOARD

Include Transcation ID

- Primary key for linking different tables

Separate the dataset into Fact Tables, Dimension Tables, and Result Tables

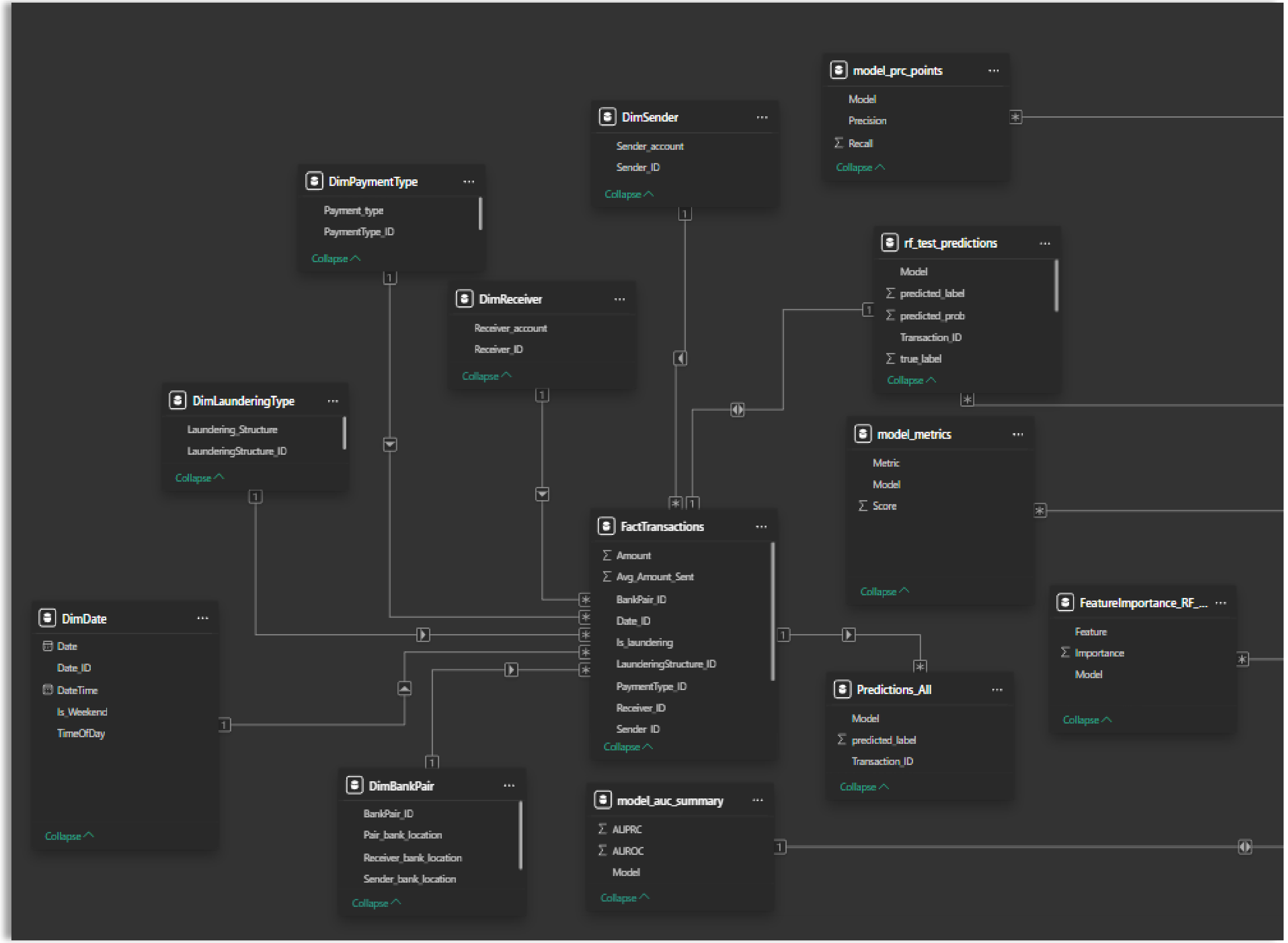
- Fact Table: contain IDs and transactional details (Amount, Transaction_ID)
- Dimension Tables: contain descriptive information (Location, Payment_type)
- Result Tables: stores predicted labels and metrics

WHY?

To improve query efficiency and makes the dashboard easier to maintain and navigate

Construct Star Schema

- Fact tables at the centre connecting with dimension tables and certain result tables through foreign key.
- Enables aggregation, filtering, and slicing data to be done efficiently when developing dashboard.



UTM
UNIVERSITI TEKNOLOGI MALAYSIA

CHAPTER 7

CONCLUSION

SUMMARY

Achieved Objective 1 – Completed data preprocessing and EDA

- ✓ Cleaned, transformed, and derived new features
- ✓ Explore transaction patterns such as most preferred payment types and high-risk pair countries

Achieved Objective 2 – Built and evaluated classification models

- ✓ Developed Random Forest & XGBoost, then tuned and tested
- ✓ Result shows that Random Forest performed slightly better

Achieved Objective 3 – Developed interactive dashboard

- ✓ Built dashboard using PowerBI with fact, dimension, and result tables
- ✓ Visualize insights based on descriptive, diagnostic, predictive, and prescriptive analytics

LIMITATION

Synthetic data may not fully capture the complexity of real-world financial transactions.

Only Random Forest and XGBoost are tested (no deep learning or graph-based method).

Models still has **limited interpretability.** Feature importance offers limited transparency to compliance officers.

FUTURE WORK

Collaborate with financial institutions to apply methodology on **real dataset.**

Expand the models approach to **deep learning, graph-based, and hybrid.**

Use SHAP to enhance explainability for transparency and regulatory trust.

THANK YOU