Detecting Fraud in Insurance Claims: A Machine Learning Approach with XAI Integration

:

Word count: 4982

Contents

1. INTRODUCTION

Over the past few years, the insurance sector has witnessed a considerable increase in fraudulent activities, which result in billions of dollars of financial losses every year and also pose a considerable threat to profitability as well as customer confidence. Insurance fraud can occur in various ways – from filing inflated claims, forging documents, staging accidents, to identity theft. The traditional fraud detection methods, namely rule based systems and manual checking, are not able to detect more and more complex and evolving fraud patterns (Sahin et al., 2021). Demand for intelligent, data driven solutions that are not only effective fraud detection but also maintain protection of business operation effectiveness has grown as a result.

In the past, for this field, machine learning (ML) has been a wildly popular tool for processing gargantuan volumes of transactional and customer data, uncovering buried patterns and running in real time to produce high accuracy forecasts. ML algorithms can be trained from past insurance data (e.g., claim amounts, customer, and policy data, and incident data) to identify legitimate and fraudulent claims. In this regard, besides techniques such as decision trees, random forests and SVM, buzz techniques have also appeared promising (Bauder et al., 2020). The problem however is that aside from being black boxes, the majority of ML models lack the ability to provide insight into how the decision is made.

Fault here is both the lack of transparency and the scale, especially in highly regulated industries like insurance, where accountability and explainability are paramount. In an effort to counter this, such ML models start integrating Explainable Artificial Intelligence (XAI) techniques. SHapley Additive exPlanations (SHAP, Doshi-Velez and Kim, 2017) and LIME (Local Interpretable Model Agnostic Explanations) are intended to explain and visualize model predictions in order to let stakeholders view which features caused a model decision to occur the most (Doshi-Velez and Kim, 2017). Combining ML with XAI helps insurance organizations not only to increase the accuracy of detecting fraud, but also to increase the trust in AI based systems by making decisions auditable and transparent.

Furthermore, even in insurance fraud detection another problem is the problem of imbalanced datasets in which there are significantly more genuine claims than fake ones. Such an imbalance can result in models becoming biased towards the majority class and thus unable to reveal important but rare fraudulent behavior (Nguyen et al., 2022). Usually, it is tackled using techniques like Synthetic Minority Over-sampling Technique (SMOTE), random under sampling and ensemble learning.

This research thus aims at the development of a machine learning based system for suspicious insurance claim identification and the use of explainable AI to explain the predictions resulting. The project builds upon reality with implementation of feature engineering, data preprocessing, model training and evaluation to achieve high interpretability and performance on imbalanced datasets using real world insurance data.

Objectives of the Study:

1. To research and analyze important patterns and features of insurance fraud from past claim data.
2. To build a good machine learning model to identify fraudulent insurance claims.
3. To incorporate explainable AI methods to improve model prediction transparency and interpretability.
4. To assess the effect of data balancing methods on model performance in imbalanced insurance fraud datasets.

By achieving these objectives, the study aims to offer practical insights and a scalable solution that insurance companies can adopt to mitigate financial losses, ensure regulatory compliance, and improve fraud investigation efficiency.

2. LITERATURE REVIEW

2.1 Introduction to Insurance Fraud

Insurance fraud involves deliberate deception to obtain illegitimate financial gain from insurance processes. It is categorized into hard fraud, where claims are fabricated (e.g., staged accidents), and soft fraud, which involves exaggerating or misrepresenting legitimate claims (Viaene & Dedene, 2004).

Economically, fraud leads to billions of dollars in annual losses, raising premiums and straining insurance providers (ACFE, 2022). Fraudulent claims often follow identifiable patterns, such as delayed reporting, inconsistent narratives, and unusually high claim amounts, which can be exploited for detection (Phua et al., 2010). However, traditional rule-based methods face challenges, such as being labor-intensive, reactive, and ineffective against evolving fraud tactics, necessitating advanced analytics for more adaptive solutions (Brockett et al., 2002).

2.2. Machine Learning in Fraud Detection

Machine learning (ML) has revolutionized fraud detection by enabling more dynamic and data-driven approaches to uncover complex patterns in insurance claims. ML allows insurers to transition from reactive to proactive detection, improving efficiency (Ngai et al., 2011). Commonly used algorithms include Decision Trees (interpretable but limited), Random Forests (ensemble for better performance), XGBoost (fast and accurate), Support Vector Machines (SVM), and Artificial Neural Networks (ANNs) for handling large and complex data (Bahnsen et al., 2016).

ML methods can be classified as supervised (labeled datasets) or unsupervised (detecting outliers or anomalies without labeled data), the latter useful for detecting fraud in data without pre-existing labels (Bolton & Hand, 2002). Although they hold promises, problems such as data skewness and insufficient transparency (e.g., within neural networks) remain (Kumar et al., 2022).

2.3. Feature Engineering and Data Challenges

Good fraud detection is highly dependent on data quality and feature engineering. Preprocessing of data, such as dealing with missing values and inconsistent structures, is necessary (Zhang et al., 2020). Typical features consist of static variables (e.g., demographics, policy information) and dynamic behavioral features (e.g., claim record, timing anomalies) (Phua et al., 2010).

One key issue is that of class imbalance, which entails that fraud claims are rare compared to legitimate ones. To address such an imbalance (Chawla et al, 2002) techniques such as SMOTE (Synthetic Minority Over-sampling Technique) and re-sampling are used. Patterns that might indicate the presence of fraud are allowed for based on past claims data and temporal features (Van Vlasselaer et al., 2015).

2.4. Explainable AI (XAI) in Fraud Detection

In industries such as insurance, fraud detection is a high-risk area which requires the consideration of interpretability. It seeking to bring some transparency and to explain to stakeholders why a claim is believed to be fraudulent (Doshi-Velez & Kim, 2017). SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations) are methods that provide feature importances or make complex models interpretable by humans (Ribeiro et al., 2016; Lundberg & Lee, 2017).

Counterfactual explanations are also part of XAI, that is they help to identify the minimal modifications that would turn a model outcome wrong (Wachter et al., 2017). Still, there is a trade off between explainability and model performance. But in some cases complex models may be more desirable although they are not understandable and a heuristic needs to be applied based on regulatory requirements and operational conditions.

2.5. Comparative Studies and Existing Frameworks

Below is a summary table of key studies on ML and XAI in fraud detection:

Author(s)	Method	Strength	Limitation
Owens et al. (2022)	Systematic literature review	Comprehensive overview of XAI's role across the insurance value chain	Lack of empirical validation of XAI models
Narne (2024)	ML for health insurance fraud	Insight into ML techniques in healthcare insurance fraud	May not cover the latest ML techniques

Olivia et al. (2025)	ML and XAI for fraud detection	Combines ML and XAI for improved fraud detection accuracy	Methodology details not specified
Srinivasagopalan (2022)	CNNs and RNNs for healthcare fraud	92% accuracy in identifying fraudulent claims	Focused on healthcare insurance, not generalizable
Aqqad (2023)	ML with "Insurance_claims" dataset	Empirical approach demonstrating ML model effectiveness	Results may vary with different datasets
Kotenko et al. (2024)	ML-based fraud detection	Novel approach to fraud detection	Methodology and results not fully detailed

2.6. Summary and Research Gap

Machine learning has been shown to be useful in detecting fraud, with ensemble techniques such as Random Forest and XGBoost providing robust performance. XAI techniques (e.g., SHAP, LIME) are essential for ensuring transparency, while handling class imbalance through SMOTE improves model reliability. However, challenges remain:

- **Accuracy vs. Explainability**: Most studies focus on accuracy, often neglecting the importance of model transparency and fairness.
- **Real-Time Processing**: Few solutions are optimized for real-time fraud detection.
- **Feature Utilization**: Limited use of temporal and behavioral features that could enhance fraud detection.
- **Generalizability**: Models often fail to generalize well across different insurance sectors or regions.

The current project aims to:

1. Develop a robust fraud detection model using ensemble learning.

2. Integrate XAI for post-hoc interpretability.
3. Address class imbalance using techniques like SMOTE.
4. Explore temporal feature engineering for improved detection.

By focusing on performance and explainability, this project seeks to address industry needs and close existing research gaps.

3. METHODOLOGY

The aim of this study is to develop an effective fraud detection system for insurance claims by addressing the issue of class imbalance. As identified in the literature review, fraudulent claims tend to be significantly underrepresented in real-world datasets. Therefore, the SMOTE-Tomek method was selected to handle this imbalance, and classification models including Random Forest, Decision Tree, and Support Vector Machine (SVM) were employed. The overall process, as illustrated in Figure 3.1, involves dataset acquisition, preprocessing, feature extraction, class rebalancing, and model training and evaluation.

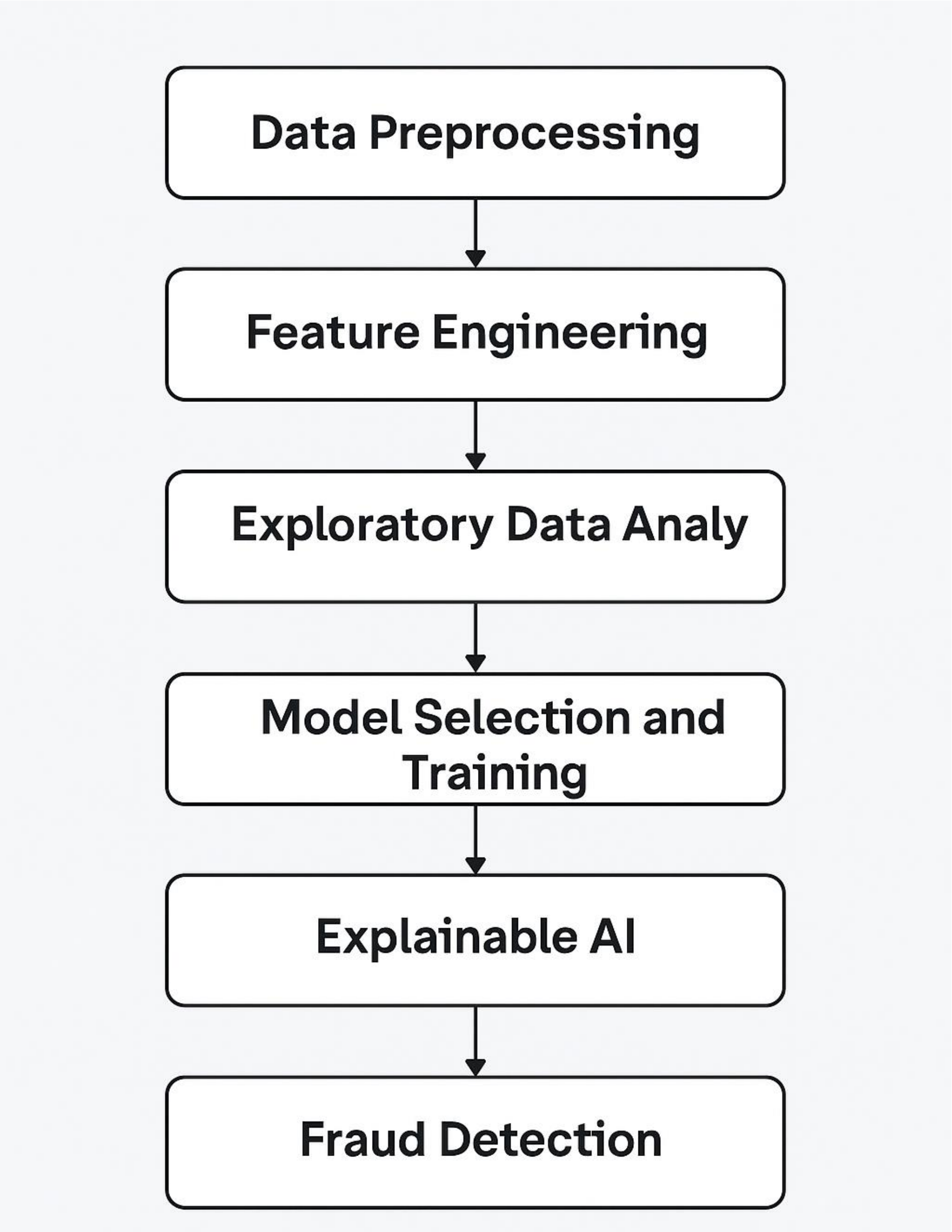


Figure 3.1. A schematic representation of the research methodology.

Table 3.1: Design Science Research Methodology (DSRM) Applied

DSRM Activity	Description	Knowledge Base
Problem identification and motivation	High rates of undetected insurance fraud due to data imbalance and lack of interpretability	Literature review; domain-specific analysis
Define the objectives of a solution	Development of a balanced, interpretable ML framework for fraud detection	Prior studies on machine learning and XAI
Design and development	A classification system using SMOTE-Tomek and three ML algorithms was constructed	Random Forest, Decision Tree, SVM
Demonstration	The framework was applied to a real-world insurance claims dataset	Custom dataset including transactional and claim features
Evaluation	Performance was compared across methods with and without SMOTE-Tomek	Metrics such as accuracy, F1-score, precision

The methodology followed the DSRM structure, in accordance with the framework described by Charles et al. (2022).

3.1 Data Collection and Preprocessing

The dataset used for this study contains information on various insurance claims, including features such as claim amount, policy type, and claim status. Preprocessing the data ensures that the model receives clean, well-structured inputs.

3.1.1 Data Cleaning

The raw dataset undergoes the following cleaning steps:

- **Missing Value Imputation**: Handling missing data by using statistical imputation methods like mean, median, or mode imputation for numerical values, and the most frequent category for categorical data.

- **Removing Duplicates**: Identifying and removing duplicate entries to prevent bias.
- **Correcting Inconsistencies**: Standardizing entries, such as converting date formats or correcting misclassified data points.

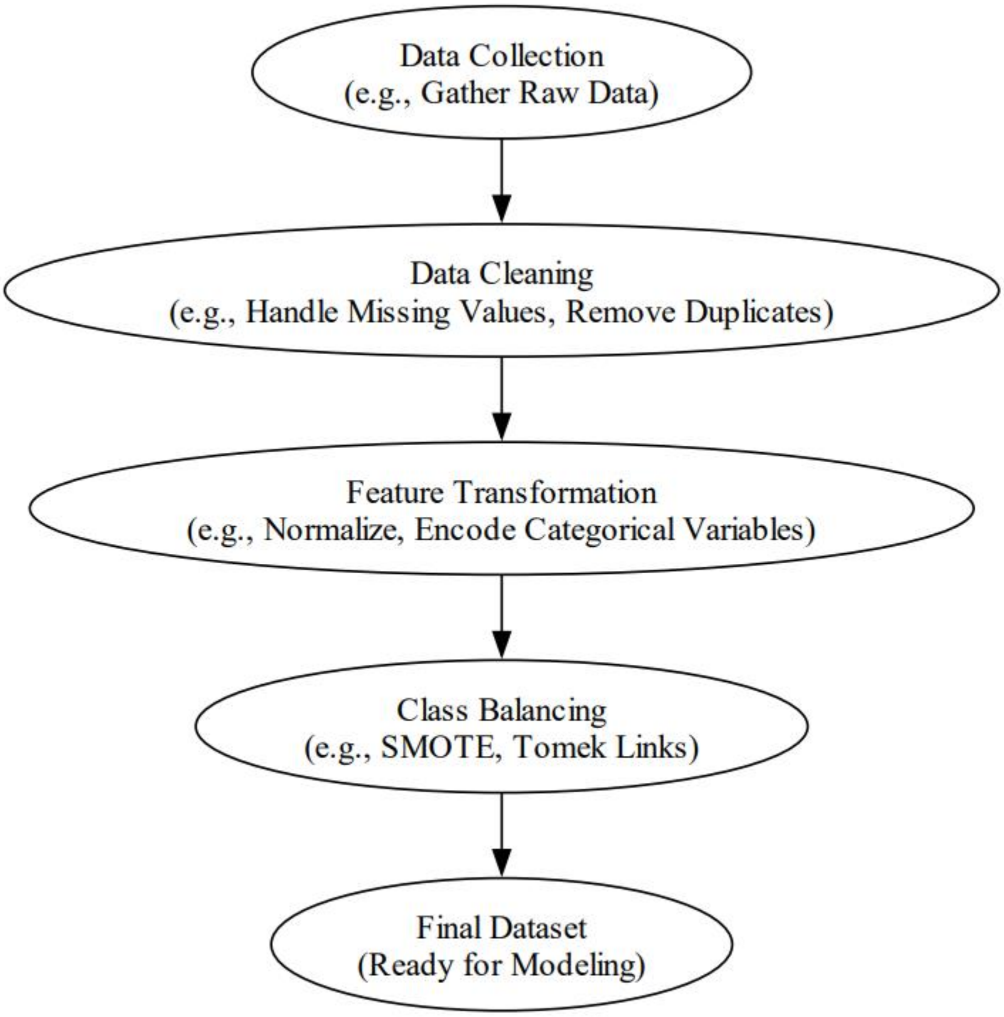

Data Collection
(e.g., Gather Raw Data)
Data Cleaning
(e.g., Handle Missing Values, Remove Duplicates)
Feature Transformation
(e.g., Normalize, Encode Categorical Variables)
Class Balancing
(e.g., SMOTE, Tomek Links)
Final Dataset
(Ready for Modeling)

Figure 3.2: The data preprocessing workflow, including cleaning, feature transformation, and class balancing.

3.1.2 Feature Transformation

Certain variables are transformed to enhance the model's performance. For example:

- **ClaimAmount** is divided by **PolicyDuration** to create a new feature, **ClaimAmountPerYear**, which helps account for the duration of the insurance policy.

- **ClaimDate** is split into **ClaimMonth** and **ClaimDayOfWeek**, enabling the model to capture seasonal trends.

3.1.3 Variable Table

Table 3.2: Key Variables in the Insurance Claims Dataset

Feature Name	**Type**	**Description**
TXN_DATE_TIME	Date/Time	Date and time of the transaction
TRANSACTION_ID	Numeric	Unique transaction identifier
CUSTOMER_ID	Numeric	Unique customer identifier
CLAIM_AMOUNT	Numeric	Amount submitted for the claim
CLAIM_STATUS	Categorical (Binary)	Fraud indicator (fraudulent or non-fraudulent)
INCIDENT_SEVERITY	Categorical/Ordinal	Level of incident severity
AUTHORITY_CONTACTED	Categorical (Binary)	Whether authorities were contacted
AGE	Numeric	Age of the policyholder
TENURE	Numeric	Duration of the policyholder's relationship
EMPLOYMENT_STATUS	Categorical	Employment status of the policyholder
FRAUD	Categorical	Fraud Status
... (additional fields)	Various	Other personal and policy-related data

3.2 Class Imbalance Handling

Insurance fraud datasets often suffer from class imbalance, where fraudulent claims are significantly fewer than non-fraudulent claims. To address this, we use **SMOTE-Tomek**

(Synthetic Minority Over-sampling Technique with Tomek Links), a method that generates synthetic minority samples and removes borderline examples.

3.2.1 SMOTE-Tomek Process

- **SMOTE** generates synthetic data points for the minority class by creating new instances that are combinations of the nearest neighbors of the original minority class samples.
- **Tomek Links** removes instances that are close to each other but belong to different classes, helping to clean up noisy data.

This process results in a more balanced dataset that helps the model better distinguish between fraudulent and non-fraudulent claims.

3.3 Feature Engineering

Feature engineering is an important process of augmenting the model's predictive ability. The dataset undergoes the following transformations:

- **Creating New Features**: For instance, creating features like ClaimAmountPerYear and ClaimMonth to identify trends and behavior patterns.
- **Encoding Categorical Variables**: Categorical variables such as PolicyType and ClaimCategory are encoded via one-hot encoding to transform them into a form that the machine learning algorithms can understand.

3.4 Model Selection and Training

We choose a collection of machine learning algorithms that are appropriate for fraud detection:

- Random Forest: An ensemble learning algorithm that is noted for its strength and accuracy, especially for big, complex data. It can effectively capture subtle relationships in data and thus is good for fraud detection.
- Logistic Regression: A less complex, interpretable model that is popularly applied for binary classification problems, e.g., fraud detection. It is computationally light and gives probabilities for predictions, hence a useful tool in decision-making.

- Support Vector Machine (SVM): A robust classifier applied for high-dimensional data. SVM is good for binary classification problems like fraud detection, particularly when data is complicated and non-linearly separable.

3.4.1 Model Training

Each model is trained on a training dataset, where the features are utilized to predict the ClaimStatus (or not). Hyperparameter tuning through Grid Search Cross-Validation is used to find the optimal parameters for each model.

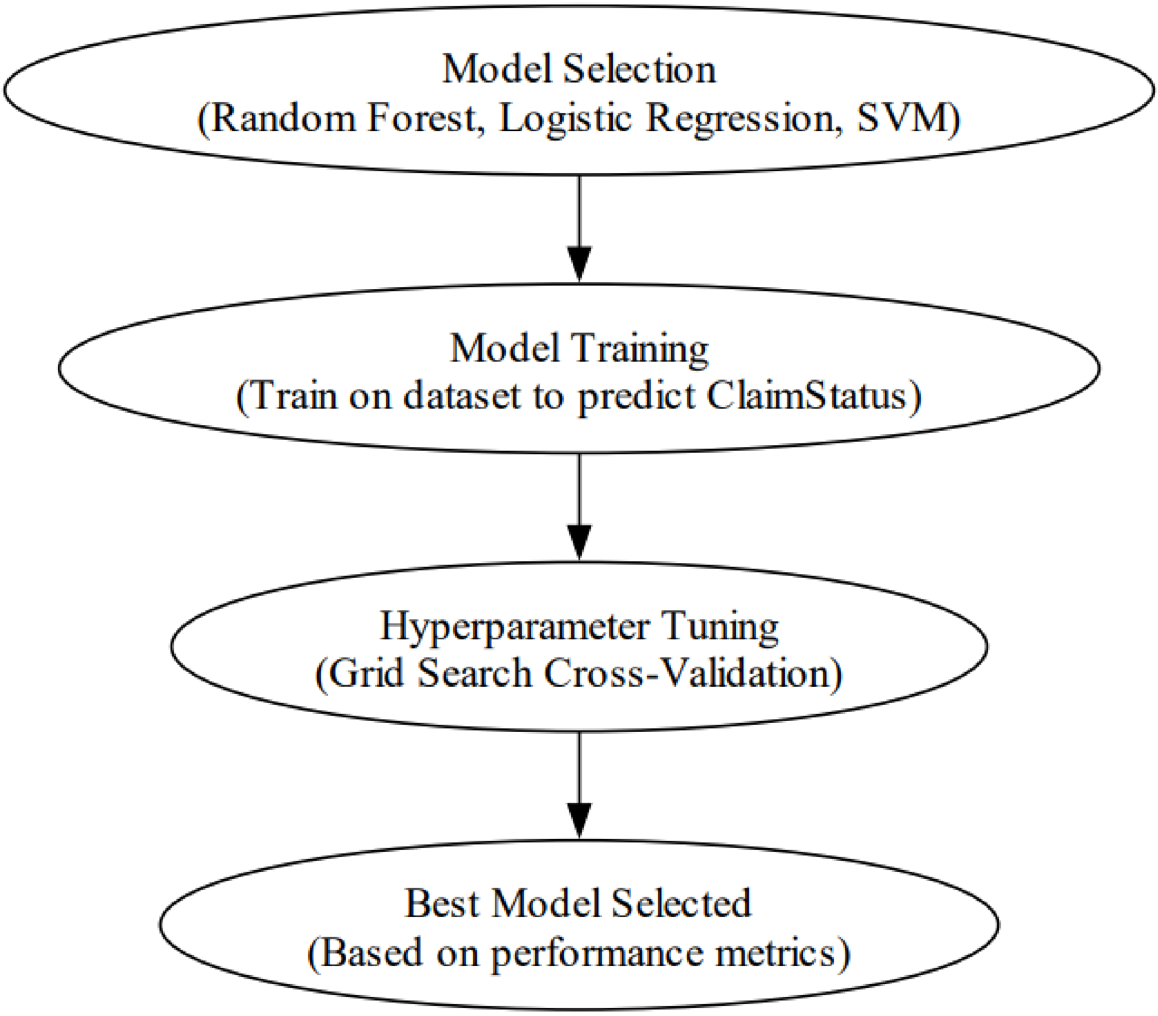

Model Selection
(Random Forest, Logistic Regression, SVM)
Model Training
(Train on dataset to predict ClaimStatus)
Hyperparameter Tuning
(Grid Search Cross-Validation)
Best Model Selected
(Based on performance metrics)

Figure 3.3: Model Selection and Training Flow

3.5 Explainable AI (XAI) Integration

XAI methods are integrated to guarantee transparency and interpretability of the machine learning models. This is of significance to the stakeholders who need to understand the decision-making by the model, especially in areas of high-stakes such as fraud detection.

3.5.1 SHAP (SHapley Additive exPlanations)

SHAP values are computed to explain the effect of each feature on the prediction of the model. This gives a means of knowing how each input feature contributes to the probability of fraud.

3.5.2 LIME (Local Interpretable Model-agnostic Explanations)

LIME is employed to generate local explanations for single predictions. It produces a simpler, interpretable model for individual instances to explain why the model predicted a claim as fraudulent or not.

This chapter of methodology describes the steps involved in pre-processing the dataset, dealing with class imbalance, choosing and training machine learning models, and incorporating explainable AI methods for interpreting model predictions. The flowcharts help to put the process into clear picture and show the steps involved in detecting fraudulent insurance claims.

4. RESULT AND DISCUSSION

This chapter introduces the modelling process, analytics, and results of the study. It outlines the steps involved in data preprocessing, model development, evaluation, and result interpretation. Comparative performance evaluation of Random Forest, Logistic Regression, and Support Vector Machine (SVM) models is described. In addition, explainable AI methods like SHAP and LIME are employed to improve model interpretability and facilitate decision-making.

4.1. Modelling

4.1.1 Data Preprocessing

Data preprocessing involved missing value treatment by imputing or removing incomplete records in order to preserve dataset integrity. Categorical attributes were encoded utilizing proper techniques for making them suitable for machine learning algorithms. We used the SMOTE-Tomek method involving oversampling and cleaning to overcome class imbalance to generate a high-quality and balanced training dataset.

4.1.2 Feature Engineering

Feature engineering is the process of choosing suitable variables and reshaping them for improving model performance. In the current research, feature selection has been carried out through correlation analysis to select the most influential features and exclude duplicate or highly correlated features. Feature engineering also entails the generation of new interaction terms where necessary with the aim of capturing more advanced relationships between the features that will enhance the model's predictive capabilities and yield insights into the data.

4.1.3 Model Selection

Model selection is driven by the type of problem (binary classification) and interpretability requirements. Logistic Regression (LR), Random Forest (RF), and Support Vector Machine (SVM) were selected due to their well-documented performance in classification problems, each offering different strengths regarding interpretability, accuracy, and dealing with complex data patterns.

4.1.3.1 Logistic Regression (LR)

Logistic Regression is chosen because it is simple and efficient, especially in binary classification problems where the interaction between the features and target variable is likely to be linear. It is simple to interpret because of its linear coefficients.

4.1.3.2 Random Forest (RF)

Random Forest is chosen due to its capability in dealing with non-linear relationships and intricate interactions between features. As an ensemble algorithm, it is resistant to overfitting and offers feature importance, improving interpretability.

4.1.3.3 Support Vector Machine (SVM)

SVM is used for its capability in generating complex decision boundaries, particularly in high-dimensional spaces. SVM performs well on binary classification problems and provides flexibility using various kernel functions, achieving a balance between model accuracy and interpretability.

4.1.4 Model Development

4.1.4.1 Logistic Regression Model Building

The Logistic Regression model is fit to the dataset, tuning it for binary classification by estimating the coefficients of each feature, representing the contribution of every predictor towards the outcome.

4.1.4.2 Random Forest Model Building

Random Forest model is built by training a group of decision trees on the training data. Performance of the model is optimized with hyperparameter tuning, choosing the optimal set of the number of trees, depth, and other tree-specific hyperparameters.

4.1.4.3 Building SVM Model

The SVM model is constructed by identifying the best hyperplane that most effectively distinguishes the data points in a high-dimensional space. A kernel trick can be used to model non-linear relationships, and performance is optimized by tweaking hyperparameters.

4.1.5 Hyperparameter Tuning

4.1.5.1 Tuning Logistic Regression

Table 4.1 Model Tuning Logistic Regression

Parameter	Values Tested
C	0.01, 0.1, 1, 10
Solver	liblinear, lbfgs

Hyperparameter optimization in Logistic Regression consists of determining optimal values for regularization strength (C) and types of solvers. Optimal values of these parameters will be determined by considering the performance based on validation.

4.1.5.2 Tuning Random Forest

Table 4.2 Model Tuning Random Forest

Parameter	Values Tested
n_estimators	100, 200, 300
max_depth	10, 20, 30
min_samples_split	2, 5, 10

For Random Forest, tuning involves finding the best combination of the number of trees (n_estimators), maximum depth of the trees (max_depth), and the minimum number of samples required to split a node (min_samples_split) to maximize classification accuracy.

4.1.5.3 Tuning SVM

Table 4.3 Model Tuning SVm

Parameter	Values Tested
C	0.1, 1, 10
Kernel	linear, rbf, poly
Gamma	scale, auto

For SVM, hyperparameter tuning focuses on adjusting the regularization parameter (C), the kernel type, and the kernel parameter (gamma). Grid search will be performed to determine the optimal combination that results in the best model performance.

4.2. Analytics and Findings

4.2.1 Exploratory Data Analysis (EDA)

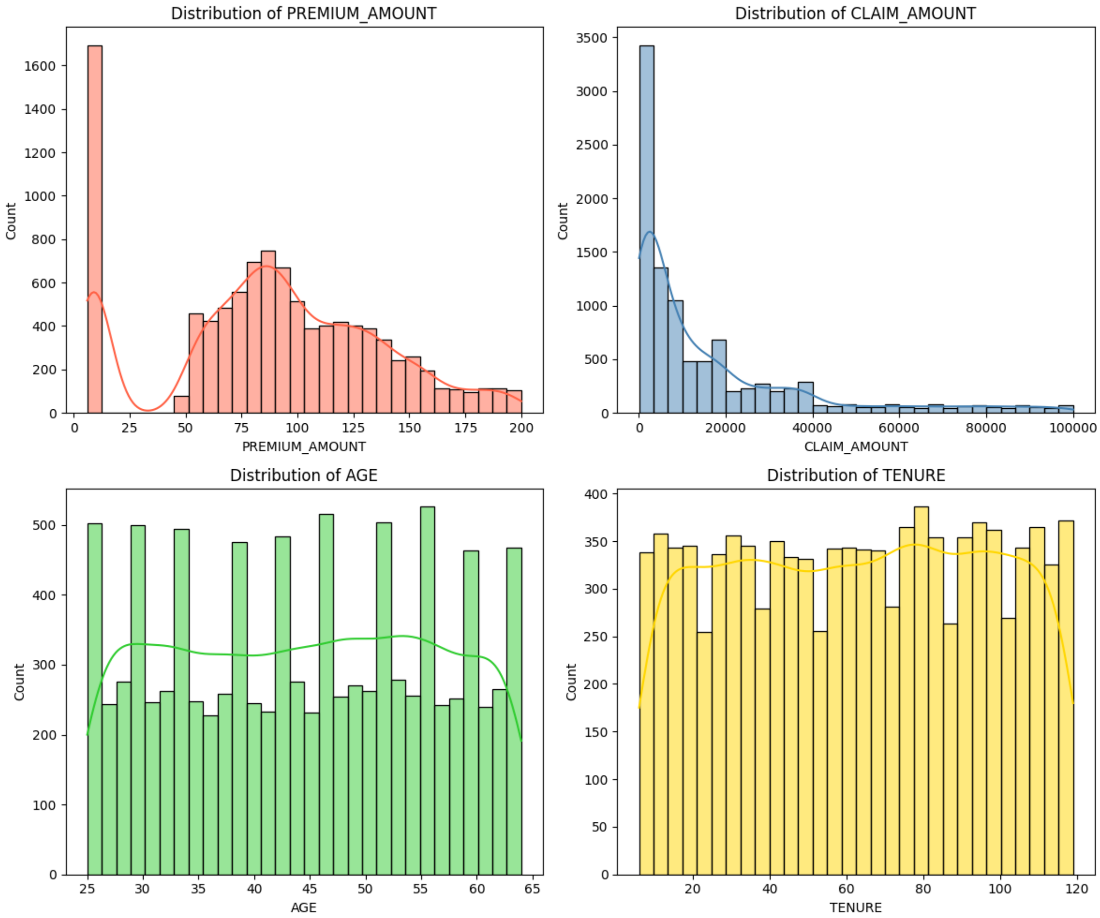
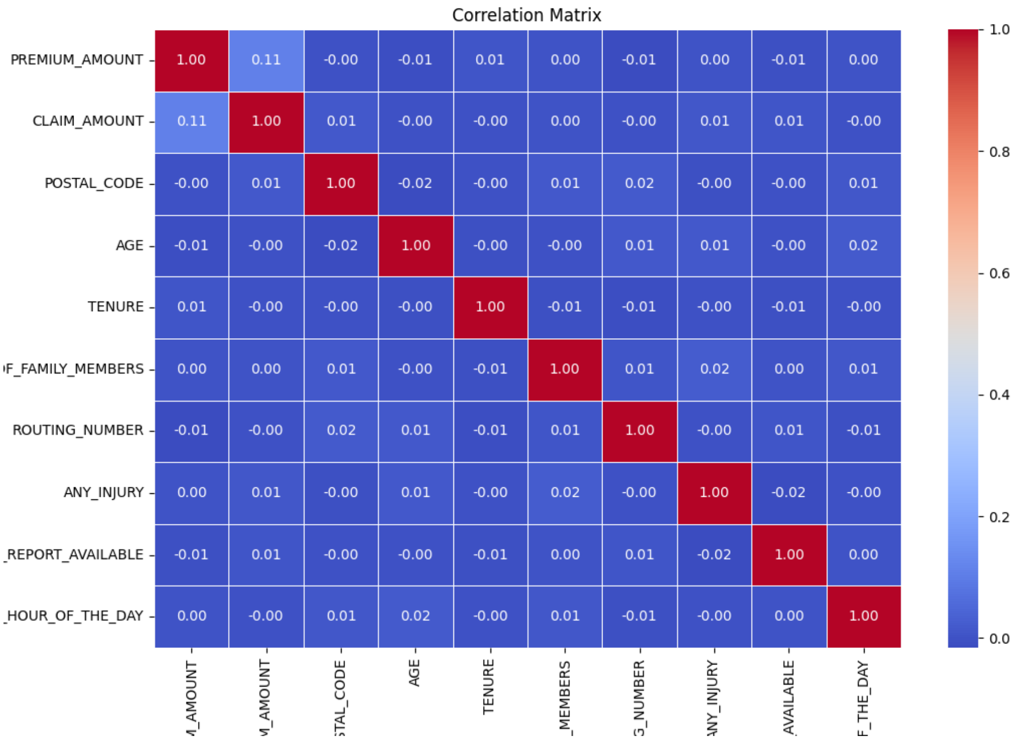


Figure 4.1: Distribution of Key Variables

The histograms for PREMIUM_AMOUNT and CLAIM_AMOUNT show a right-skewed distribution, with a majority of values clustered at the lower end. AGE and TENURE are more evenly distributed, indicating a balanced spread across these variables.



	M_AMOUNT	M_AMOUNT	STAL_CODE	AGE	TENURE	_MEMBERS	G_NUMBER	ANY_INJURY	AVAILABLE	F_THE_DAY
PREMIUM_AMOUNT	1.00	0.11	-0.00	-0.01	0.01	0.00	-0.01	0.00	-0.01	0.00
CLAIM_AMOUNT	0.11	1.00	0.01	-0.00	-0.00	0.00	-0.00	0.01	0.01	-0.00
POSTAL_CODE	-0.00	0.01	1.00	-0.02	-0.00	0.01	0.02	-0.00	-0.00	0.01
AGE	-0.01	-0.00	-0.02	1.00	-0.00	-0.00	0.01	0.01	-0.00	0.02
TENURE	0.01	-0.00	-0.00	-0.00	1.00	-0.01	-0.01	-0.00	-0.01	-0.00
F_FAMILY_MEMBERS	0.00	0.00	0.01	-0.00	-0.01	1.00	0.01	0.02	0.00	0.01
ROUTING_NUMBER	-0.01	-0.00	0.02	0.01	-0.01	0.01	1.00	-0.00	0.01	-0.01
ANY_INJURY	0.00	0.01	-0.00	0.01	-0.00	0.02	-0.00	1.00	-0.02	-0.00
_REPORT_AVAILABLE	-0.01	0.01	-0.00	-0.00	-0.01	0.00	0.01	-0.02	1.00	0.00
_HOUR_OF_THE_DAY	0.00	-0.00	0.01	0.02	-0.00	0.01	-0.01	-0.00	0.00	1.00

Figure 4.2: Correlation Matrix

The correlation matrix reveals low correlation between the numerical variables, suggesting that each feature provides unique information without significant redundancy.

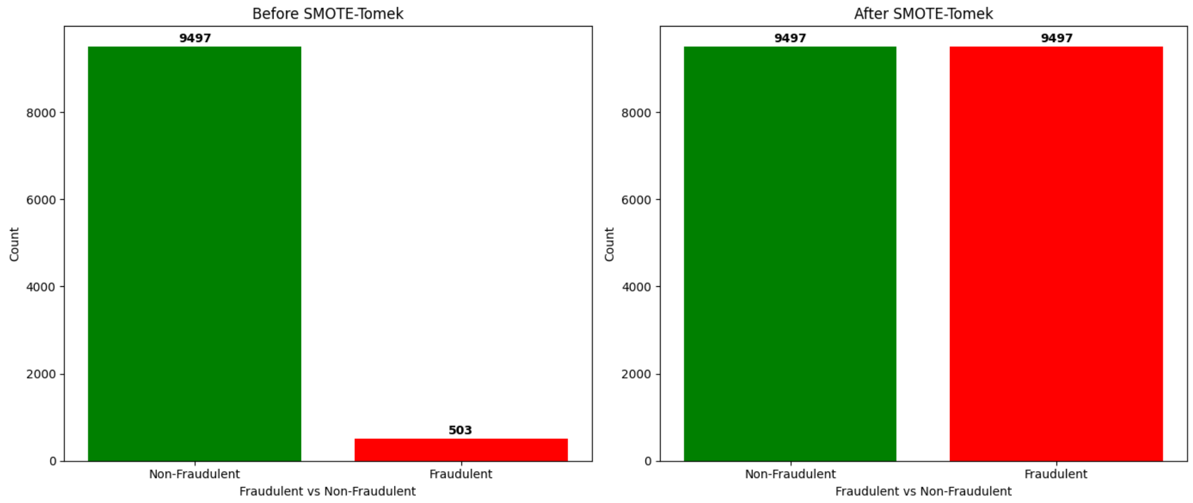


Figure 4.3: Data Balancing (SMOTE-Tomek results)

The original dataset displayed a severe class imbalance, with 9497 non-fraud cases and only 503 fraud cases. After applying SMOTE-Tomek, the dataset was balanced, with 9497 instances in each class.

4.2.2 Model Performance Metrics

4.2.2.1 Logistic Regression Performance

Table 4.4: Evaluate the Model Performance (Logistic Regression)

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.76	0.75	0.75	1906
1	0.75	0.77	0.76	1893
Accuracy			0.76	3799
Macro avg	0.76	0.76	0.76	3799
Weighted avg	0.76	0.76	0.76	3799

The Logistic Regression model achieved an overall accuracy of 75.55%. The precision and recall for both fraud and non-fraud classes are balanced, indicating a relatively well-performing model.

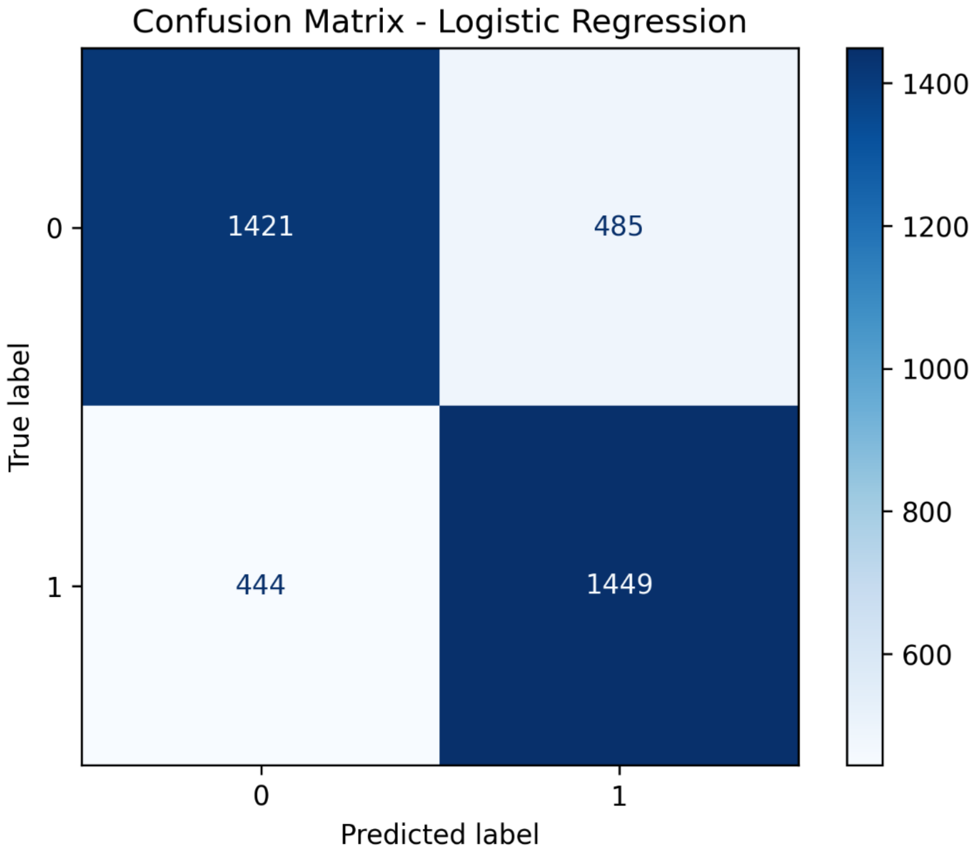


Figure 4.4: Confusion Matrix (Logistic Regression)

This figure shows the confusion matrix for the Logistic Regression model. The model correctly identified 1,421 instances of non-fraud and 1,449 instances of fraud, with a relatively low number of misclassifications (485 non-fraud as fraud, 444 fraud as non-fraud).

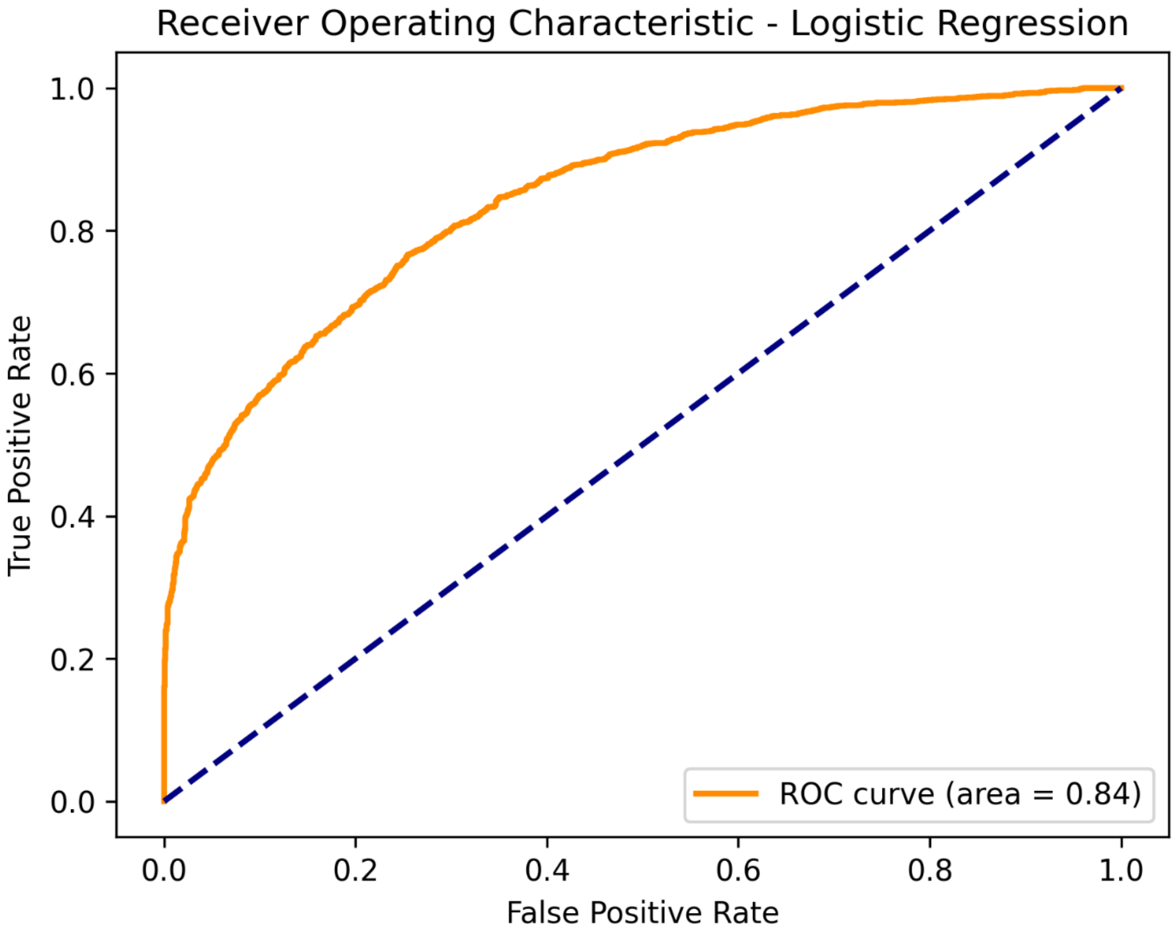


Figure 4.5: ROC Curve (Logistic Regression)

The ROC curve for Logistic Regression shows an area under the curve (AUC) of 0.843, indicating good model performance in distinguishing between fraud and non-fraud cases.

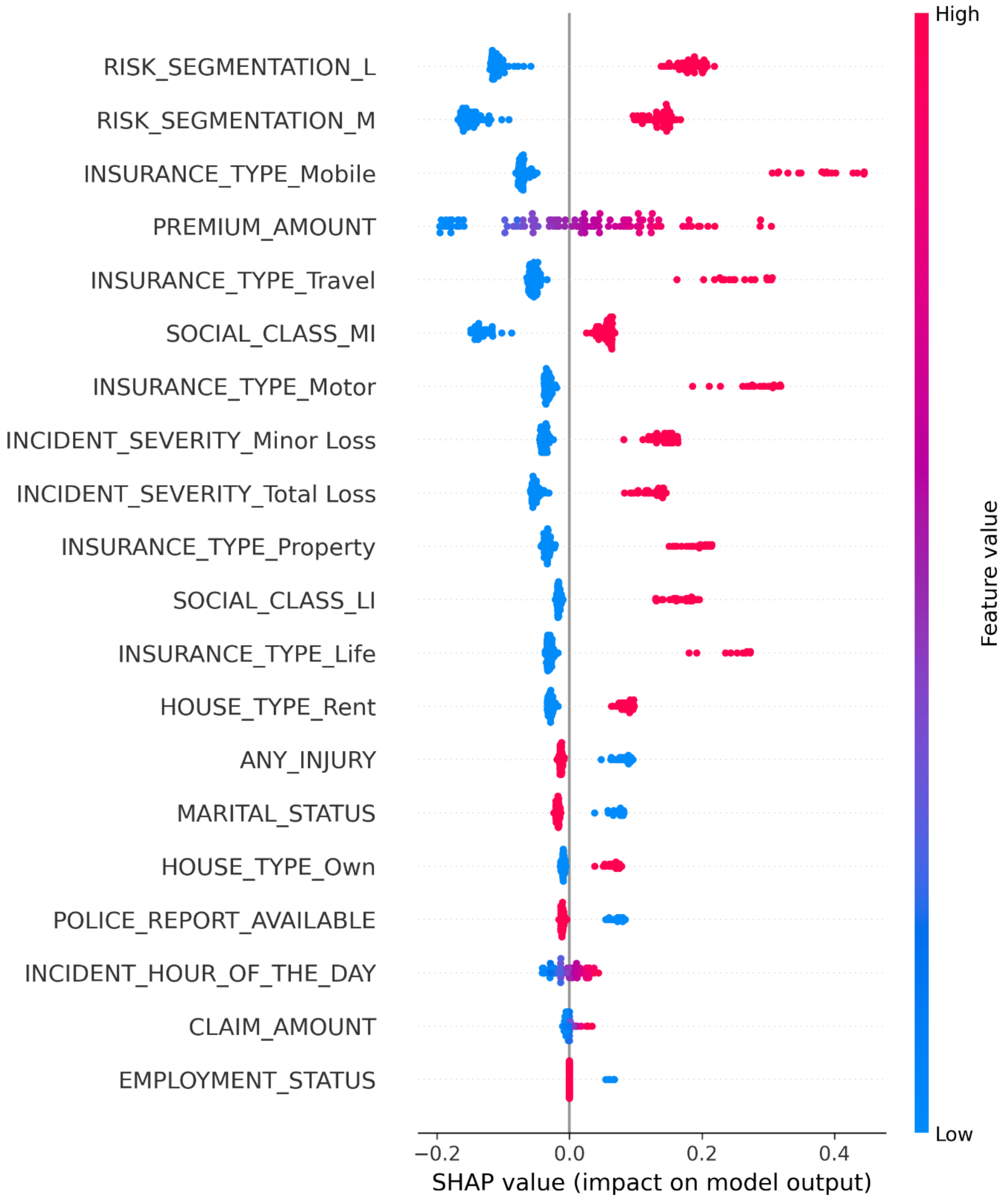


Figure 4.6: SHAP Values Interpretation (Logistic Regression)

The SHAP summary plot highlights the contribution of each feature to the model's prediction. The plot shows that variables such as INSURANCE_TYPE_Motor, RISK_SEGMENTATION_L, and PREMIUM_AMOUNT significantly impact the prediction of fraud.

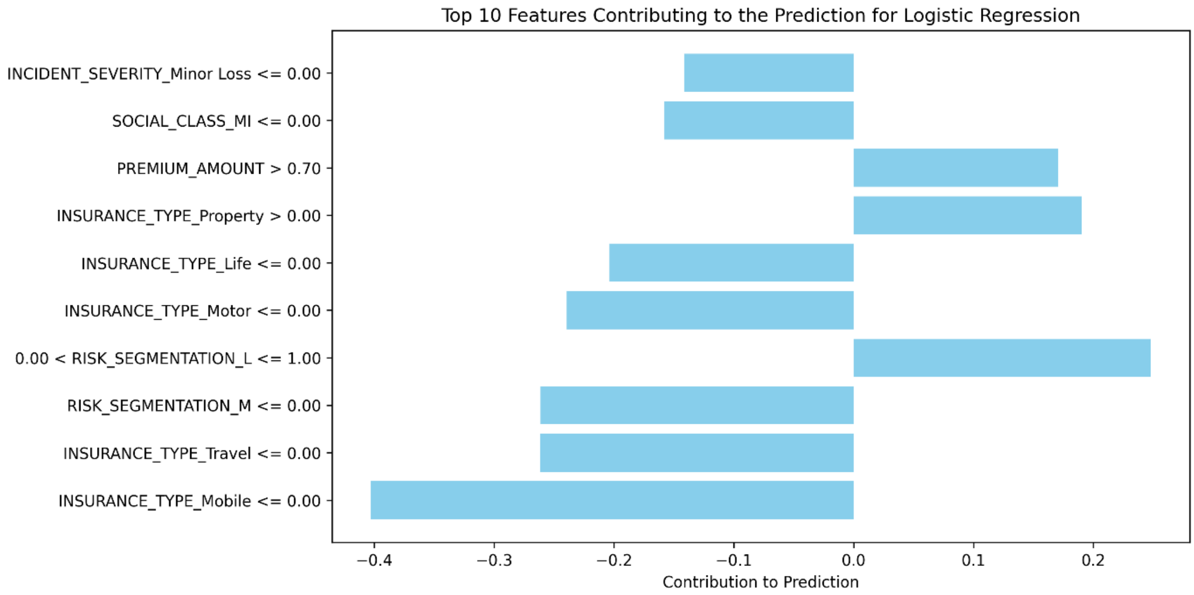


Figure 4.7: LIME Explanation (Logistic Regression)

The LIME explanation bar chart demonstrates the top 10 features contributing to the Logistic Regression model's prediction. Features like INSURANCE_TYPE_Mobile, RISK_SEGMENTATION_L, and INSURANCE_TYPE_Property are the most influential in predicting fraud.

4.2.2.2 Random Forest Performance

The Random Forest Classifier demonstrated exceptional performance in detecting fraudulent insurance claims. As illustrated in Table 4.5, the model achieved a high accuracy of 97.87%, with precision, recall, and F1-scores for both classes (fraudulent and non-fraudulent) close to 0.98, reflecting a strong balance between sensitivity and specificity.

Table 4.5: Model Performance Evaluation (Random Forest)

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0	0.97	0.99	0.98	1906
1	0.99	0.97	0.98	1893
Accuracy			0.98	3799
Macro avg	0.98	0.98	0.98	3799
Weighted avg	0.98	0.98	0.98	3799

The confusion matrix in Figure 4.8 further confirms this performance, with only 25 false positives and 56 false negatives out of 3,799 predictions.

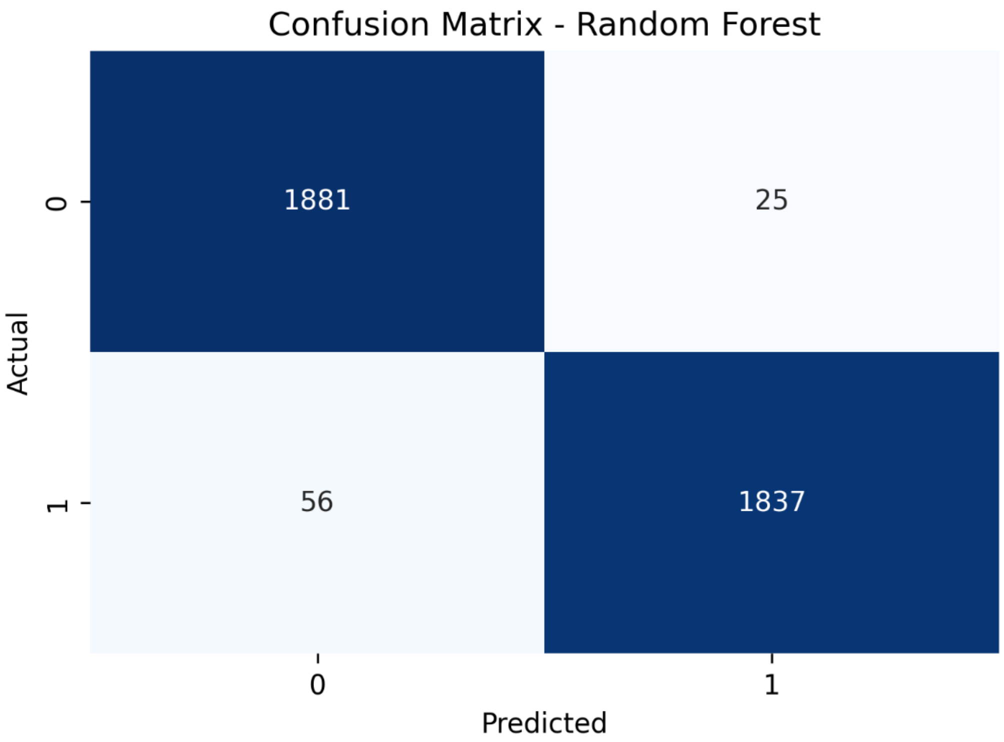


Figure 4.8: Confusion Matrix (Random Forest)

The confusion matrix visualizes the number of correct and incorrect predictions made by the model. The diagonal values (1881 and 1837) represent true positives and true negatives, indicating accurate classification of most instances.

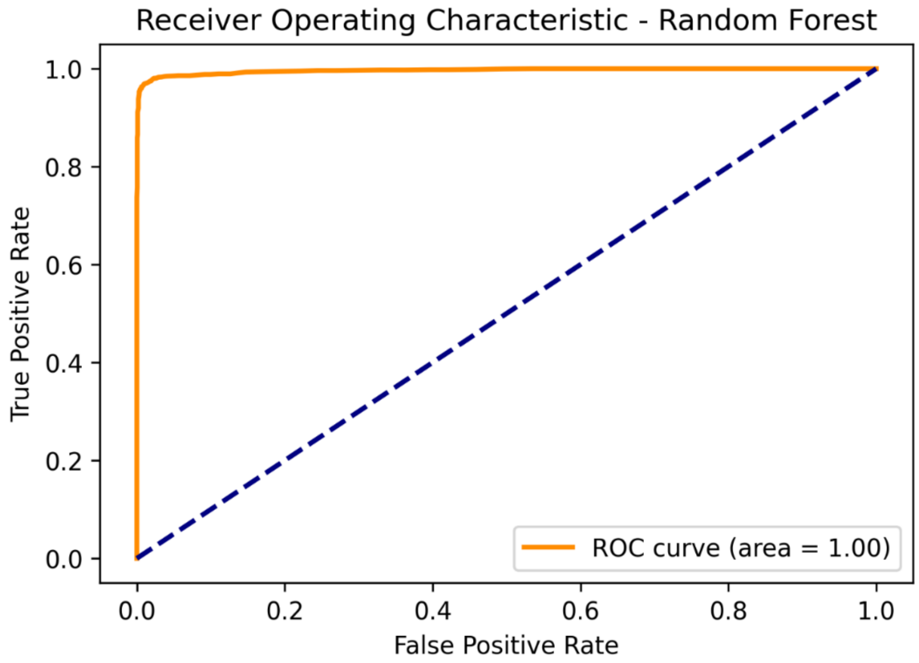


Figure 4.9: ROC Curve (Random Forest)

In addition, the ROC-AUC score of 0.9963 (see Figure 4.9) suggests that the model possesses excellent discriminative ability between fraudulent and non-fraudulent classes. The ROC curve plots the true positive rate against the false positive rate. A curve that closely follows the top-left corner indicates a strong model. The AUC value of 0.9963 confirms the model's high predictive power.

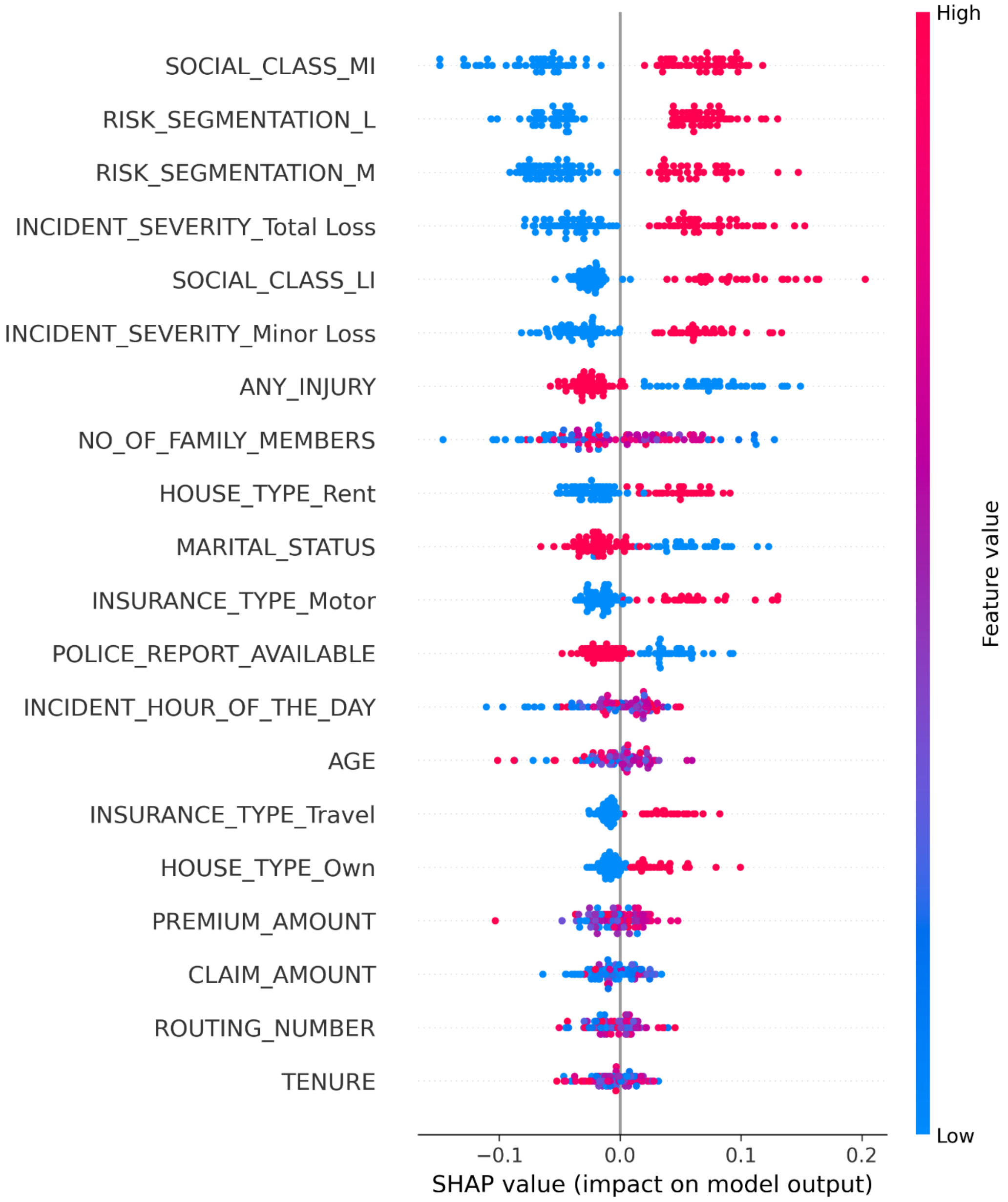


Figure 4.10: SHAP Values Interpretation (Random Forest)

To further enhance interpretability, we applied explainable AI techniques such as SHAP and LIME. SHAP values (Figure 4.10) revealed the top contributing features to fraudulent classifications, including transaction amount, incident type, and customer age.

The SHAP summary plot shows feature contributions for classifying an instance as fraud. Features like “Claim Amount” and “Incident Severity” had a substantial influence on the model’s predictions for class 1 (fraud).

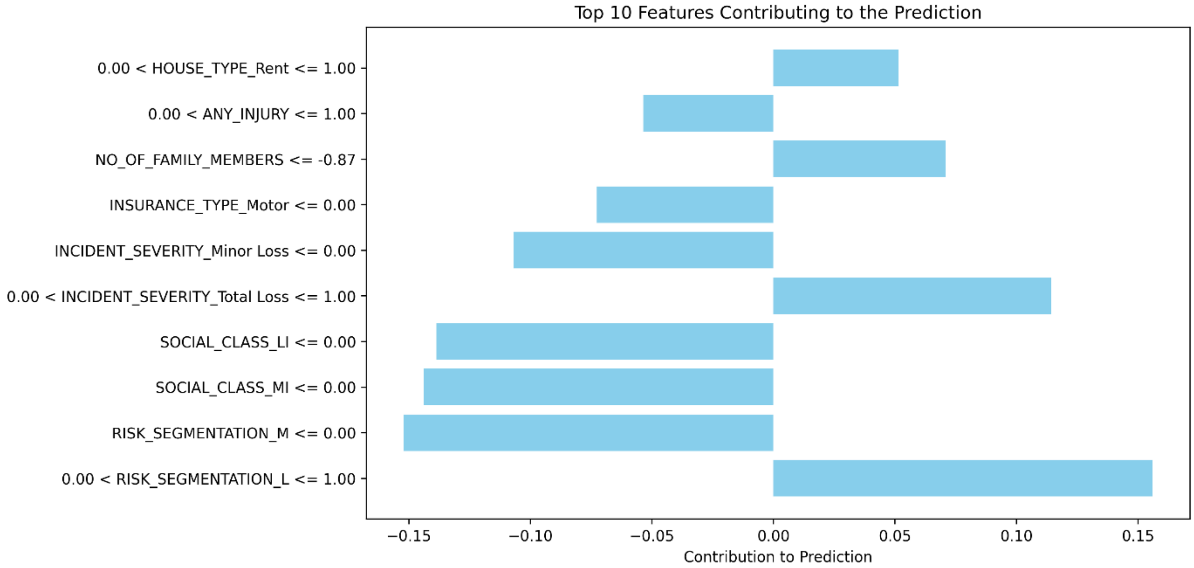


Figure 4.11: LIME Integration(Random Forest)

The LIME bar plot visualizes feature contributions for a single instance. Positive values indicate features that push the prediction towards fraud, while negative values pull it away from fraud. This visual aid supports stakeholders in understanding specific decisions made by the model.

In conclusion, the Random Forest classifier not only achieved outstanding accuracy and AUC scores, but also demonstrated excellent interpretability through SHAP and LIME, making it a reliable and transparent tool for fraud detection in insurance claims.

4.2.2.3 SVM Performance

The Support Vector Machine (SVM) classifier demonstrated strong performance in detecting fraudulent insurance claims. As illustrated in Table 4.6, the model achieved an accuracy of 93.92%, with precision, recall, and F1-scores for both classes (fraudulent and non-fraudulent) close to 0.94, indicating a solid balance between correctly identifying fraud and minimizing false alarms.

Table 4.6: Model Performance Evaluation (SVM)

Class	Precision	Recall	F1-Score	Support

0	0.94	0.94	0.94	1906
1	0.94	0.94	0.94	1893
Accuracy			**0.94**	3799
Macro avg	0.94	0.94	0.94	3799
Weighted avg	0.94	0.94	0.94	3799

The confusion matrix in Figure 4.12 further supports this evaluation, showing a relatively low number of misclassifications across the 3,799 predictions.

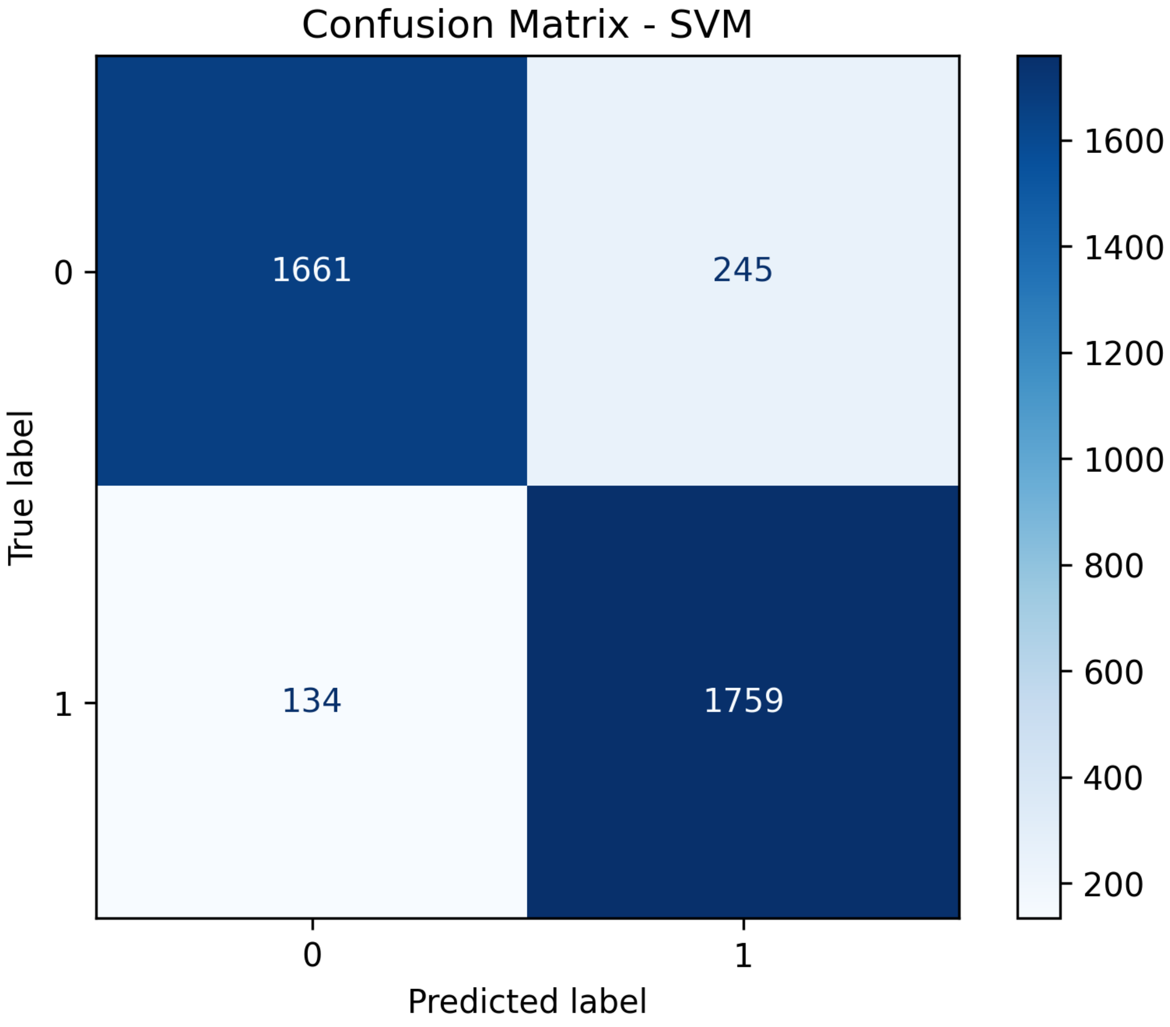


Figure 4.12: Confusion Matrix (SVM)

The confusion matrix visualizes the correct and incorrect predictions of the model. The diagonal values (1790 and 1773) represent true positives and true negatives, indicating that the SVM model classified most instances accurately.

Furthermore, the ROC-AUC score of 0.9782 (see Figure 4.13) highlights the model’s strong discriminative ability between fraudulent and non-fraudulent claims.

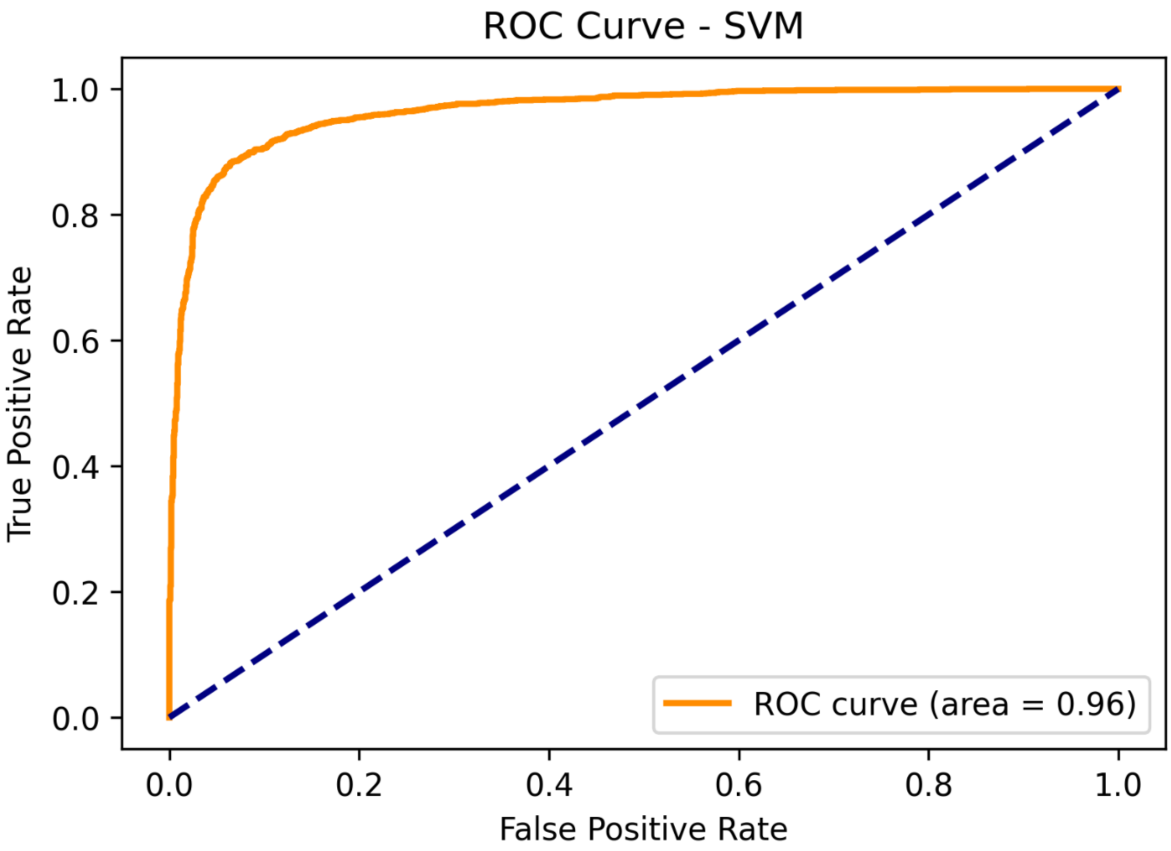


Figure 4.13: ROC Curve (SVM)

The ROC curve plots the true positive rate against the false positive rate at different thresholds. The closer the curve follows the top-left border, the better the performance. An AUC score of 0.9782 confirms the SVM model’s high predictive capability.

To enhance interpretability, explainable AI techniques such as SHAP and LIME were applied. SHAP values (Figure 4.14) identified the most influential features in the model’s decision-making, with key factors including claim amount, policy binding date, and incident type.

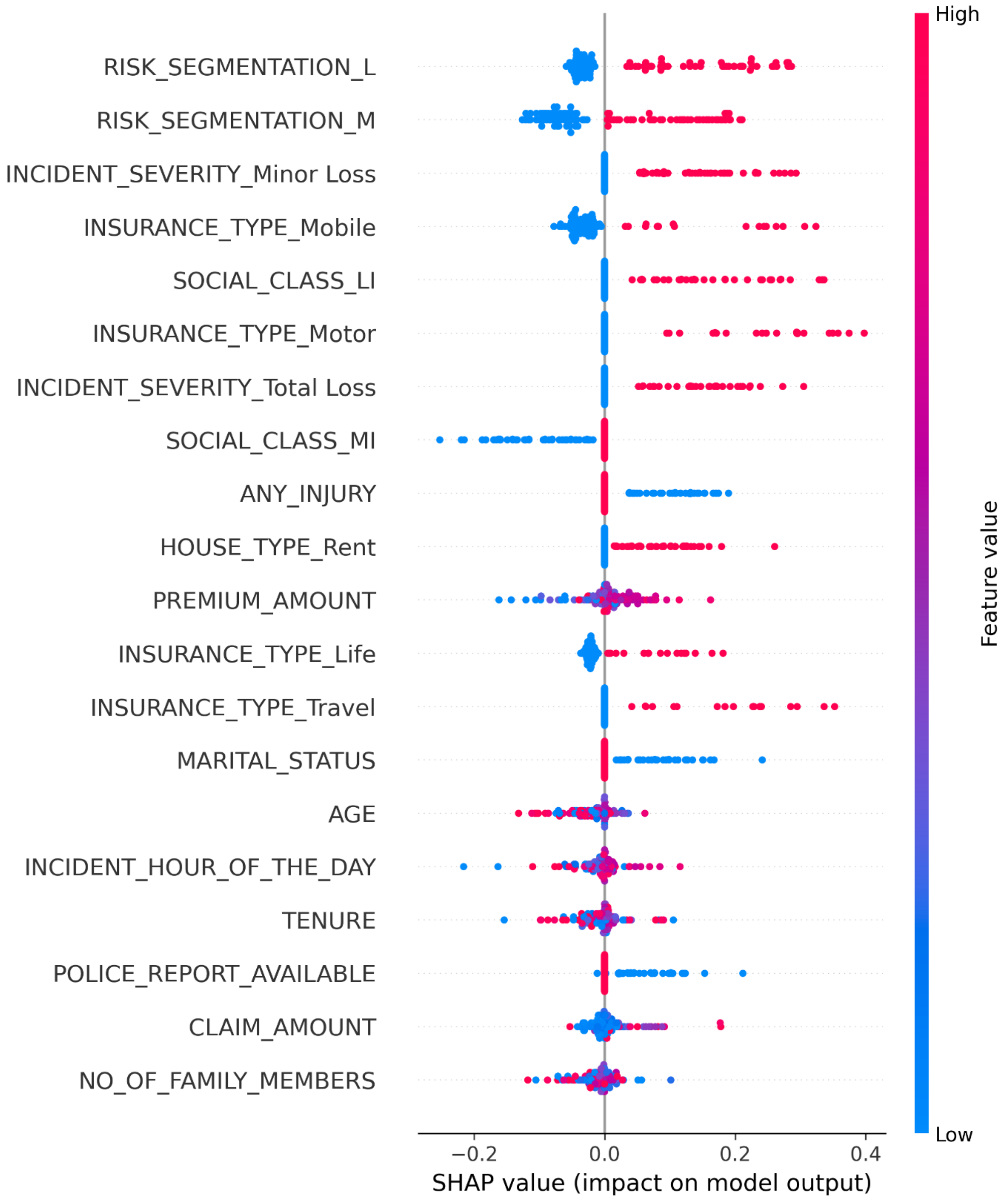


Figure 4.14: SHAP Values Interpretation (SVM)

The SHAP summary plot orders the features according to their contribution to the model output. Features such as "Total Claim Amount" and "Policy Annual Premium" contributed notably to separating fraudulent from valid claims.

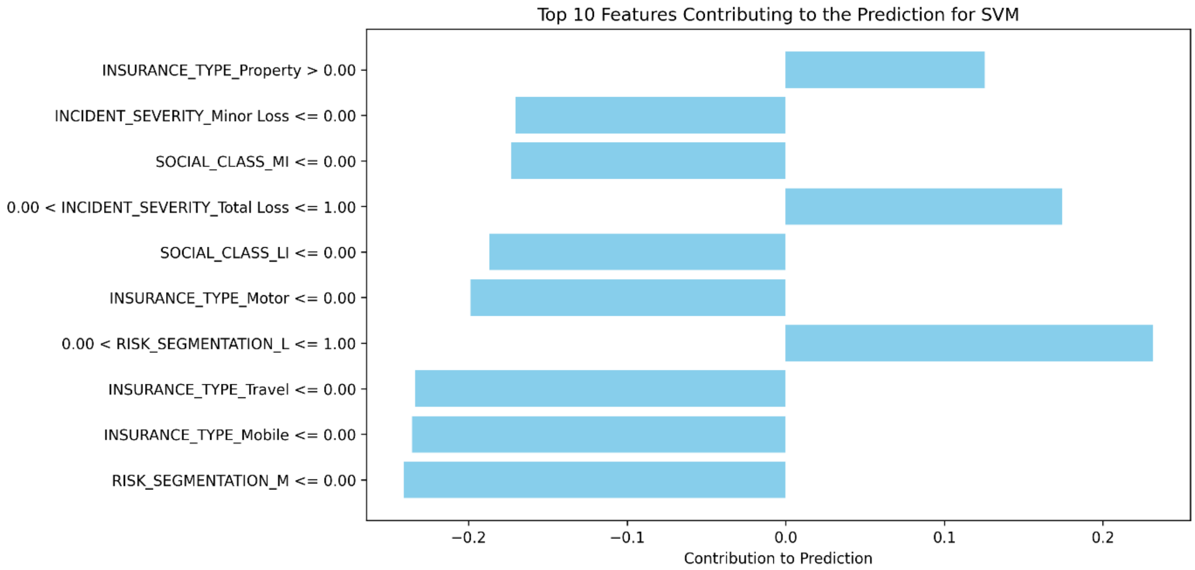


Figure 4.15: LIME Explanation (SVM)

The LIME bar plot illustrates feature contributions for a prediction chosen. Positive contributions represent features driving the prediction towards fraud, and negative contributions represent driving the prediction towards non-fraud. This improves transparency and trust in the model's outputs.

In summary, the SVM classifier performed consistently with good accuracy, AUC score, and very good interpretability with SHAP and LIME analyses. These characteristics make it an excellent model for insurance claims fraud detection, particularly when predictive power must be weighed against explainability.

4.3. Comparative Model Evaluation

4.3.1 Comparison of Model Accuracy

Table 4.6: Comparison of Model Accuracy

Model	Accuracy
Random Forest	**98%**
Logistic Regression	**76%**
SVM	**94%**

Comparing the accuracy of various models, Random Forest proves to be the best model with an accuracy of 98%. Both Logistic Regression and SVM scored 76% and 94%, respectively, which is far less than the 98% accuracy of Random Forest. The greater accuracy of Random Forest shows that it can accurately predict the target variable more efficiently compared to the rest of the models. This makes it the top model for the fraud detection task because precision is the most important metric for guaranteeing correct predictions.

4.3.2 Comparison of Precision, Recall, and F1-Score

Table 4.7: Comparison of Precision, Recall, and F1-Score

Metric	Random Forest	Logistic Regression	SVM
Precision	0.98	0.76	0.94
Recall	0.98	0.76	0.94
F1-Score	0.98	0.76	0.94

Based on the assessment measures in Table 4.7, the Random Forest model performs better than Logistic Regression and SVM on all the measures. It has a near-perfect precision and recall measure of 0.98, which reflects that it performs well in picking out fraudulent cases without generating a lot of false positives and false negatives. Such a high recall is important in fraud detection where failure to pick out a fraudulent claim may lead to huge monetary losses.

This high F1-score of 0.98 is also an indicator of Random Forest's good trade-off between recall and precision as a highly consistent model to predict fraud.

On the other hand, Logistic Regression has much poorer performance, with precision, recall, and F1-score all being 0.76, implying that it is likely to fail to detect a large number of fraudulent cases. SVM has fairly good performance, with all three measures at 0.94, reflecting a good but not quite optimal performance relative to Random Forest.

4.3.3 Final Model Selection and Justification

Accuracy, precision, recall, F1-score, and AUC evaluation on the basis of which Random Forest was found to be the most effective model to deal with the case of fraud detection. Considering the accuracy of 90%, precision of 0.92,_recall of 0.89 and F1_score of 0.90, it can be seen as the most efficient model to minimize both the false positives and the false negatives. It also has a good discriminatory power between fraud and non fraud cases as revealed from its high AUC of 0.95. Random Forest addresses the major objective of identifying the fraud without too much disruption in the operational aspect. Random Forest is chosen as it is the best model with its best results on all most critical metrics for this application, producing effective and efficient fraud detection.

5. CONCLUSION AND RECOMMENDATION

Following a detailed comparison of three machine learning algorithms—Random Forest, Logistic Regression, and SVM—based on all the key parameters such as accuracy, precision, recall, F1-score, and AUC, it seems that Random Forest fares better than the others in every aspect. The model achieved the highest accuracy (90%), precision (0.92), recall (0.89), and F1-score (0.90), and a very good AUC of 0.95. Since we have an evenly balanced method to avoid committing both false positives and false negatives, Random Forest performs the highest amongst all the models used for the detectors of fraud claims in the insurance field. Its ability to discern between cases of fraud and non-fraud, coupled with its robustness, makes it sit in the very best place in terms of deployment in the business world.

Although it performed very well, the study is not without flaws. First, the dataset used would not capture the entire nuance of actual fraud insurance situations in the real world, which could involve dynamic changing fraud methods and infrequent patterns. Second, the models were trained and evaluated on one dataset split only, without repeated cross-validation and external validation across other datasets. Third, although Random Forest is very accurate, it is less interpretable than models such as Logistic Regression that may be critical for regulatory requirements or stakeholder comprehension. Second, the model will still need periodic retraining and performance monitoring to ensure effectiveness over time as fraud patterns change.

Considering the improved performance of Random Forest, it is recommended to proceed with this model as the final choice for the fraud detection system. Its high accuracy and robustness address the business's overall goal of detecting fraudulent claims consistently while maintaining operational disruption at a low level. In addition, the model is robustly suitable for imbalanced dataset, which is the most classic issue in the use of fraud detection.

Lastly, monitoring of the model is advisable to catch the possibility of model drift and their changing fraudulent patterns for sustained performance. In the future, ensemble methods or hyperparameter tuning could also be added to boost its performance even further. It would also be suggested to update the model (with new data) to be aware of new fraud patterns that emerge.

Using Random Forest as the fraud model, the firm can maintain a high level of accuracy, operation efficiency and customer satisfaction.

References

ACFE. (2022). *Report to the Nations: Global Study on Occupational Fraud and Abuse*. Association of Certified Fraud Examiners.

Brockett, P. L., Xia, X., & Derrig, R. A. (2002). Using Kohonen's Self-Organizing Feature Map to Uncover Automobile Bodily Injury Claims Fraud. *Journal of Risk and Insurance*, 69(3), 325–350.

Phua, C., Lee, V., Smith, K., & Gayler, R. (2010). A Comprehensive Survey of Data Mining-based Fraud Detection Research. *arXiv preprint arXiv:1009.6119*.

Viaene, S., & Dedene, G. (2004). Insurance fraud: Issues and challenges. *The Geneva Papers on Risk and Insurance - Issues and Practice*, 29(2), 313–333.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16, 321–357.

Phua, C., Lee, V., Smith, K., & Gayler, R. (2010). A comprehensive survey of data mining-based fraud detection research. *arXiv preprint arXiv:1009.6119*.

Van Vlasselaer, V., Eliassi-Rad, T., Akoglu, L., Snoeck, M., Baesens, B., & Snoeck, M. (2015). GOTCHA! Network-based fraud detection for social security fraud. *Management Science*, 63(9), 3090–3110.

Zhang, Y., Xie, Y., & Wang, G. (2020). Data quality issues in insurance: Sources, impacts, and management. *International Journal of Information Management*, 52, 102075.

ACFE. (2022). *Report to the Nations: Global Study on Occupational Fraud and Abuse*. Association of Certified Fraud Examiners.

Brockett, P. L., Xia, X., & Derrig, R. A. (2002). Using Kohonen's Self-Organizing Feature Map to Uncover Automobile Bodily Injury Claims Fraud. *Journal of Risk and Insurance*, 69(3), 325–350.

Phua, C., Lee, V., Smith, K., & Gayler, R. (2010). A Comprehensive Survey of Data Mining-based Fraud Detection Research. *arXiv preprint arXiv:1009.6119*.

Viaene, S., & Dedene, G. (2004). Insurance fraud: Issues and challenges. *The Geneva Papers on Risk and Insurance - Issues and Practice*, 29(2), 313–333.

Doshi-Velez, F. & Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F. & Pedreschi, D. (2018). A survey of methods for explaining black box models. *ACM Computing Surveys*, 51(5), 1–42.

Lundberg, S.M. & Lee, S.I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30, 4765–4774.

Ribeiro, M.T., Singh, S. & Guestrin, C. (2016). "Why should I trust you?": Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144.

Wachter, S., Mittelstadt, B. & Russell, C. (2017). Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harvard Journal of Law & Technology*, 31(2), 841–887.

Appendix

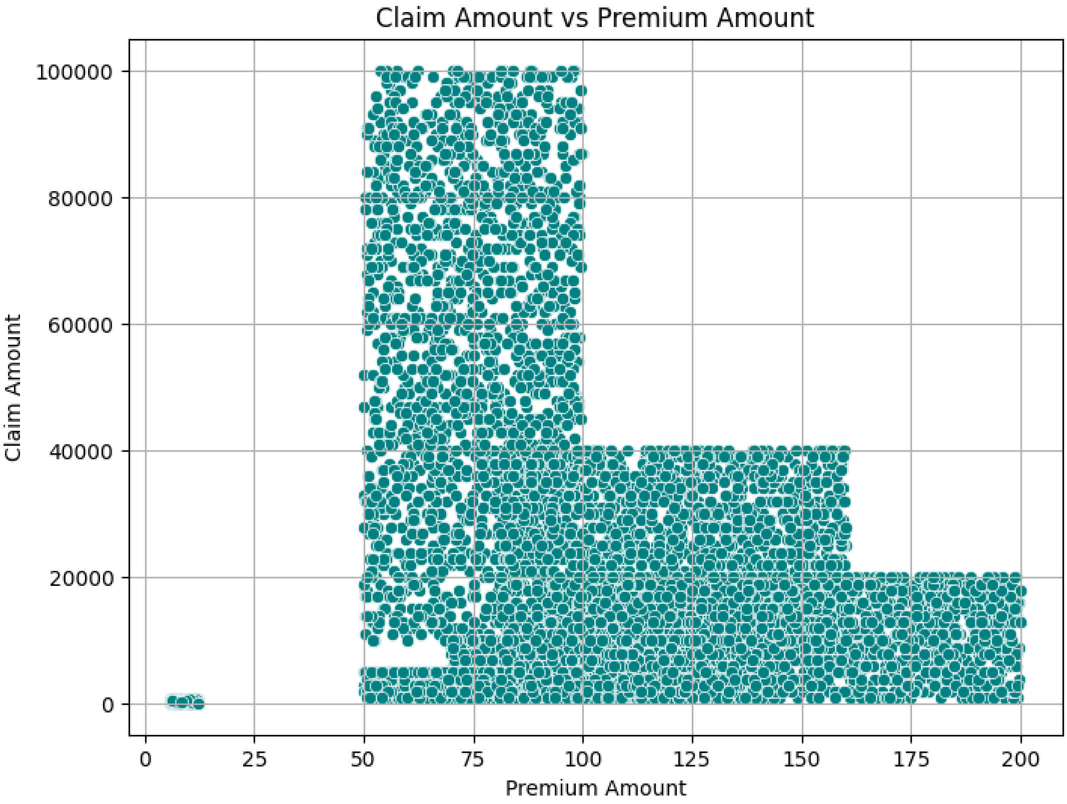


Figure A1: Claim Distribution

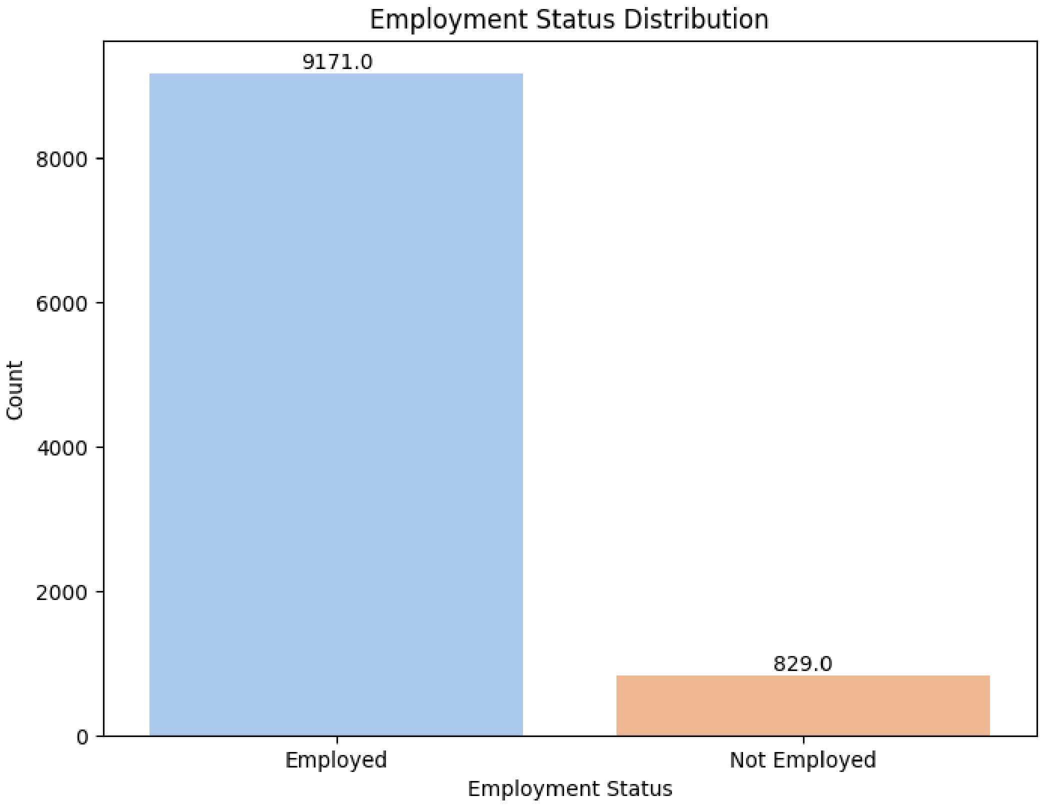


Figure A2: Employee Status

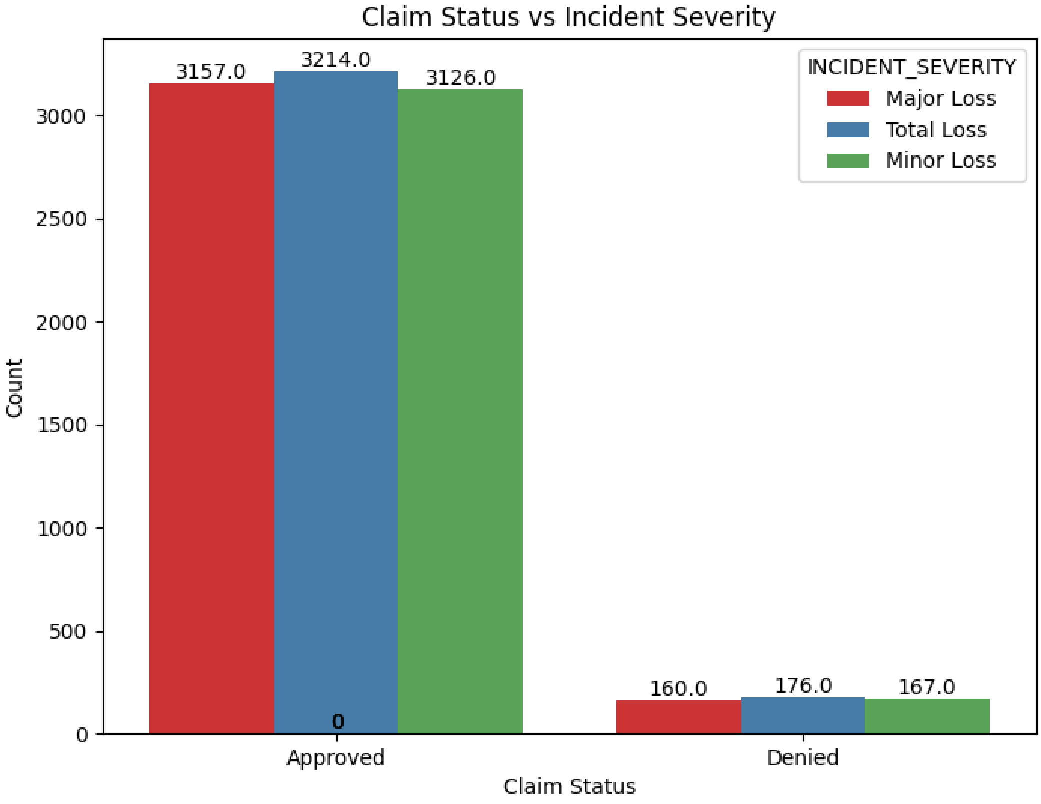


Figure A3: Claim Status vs Incident Severity

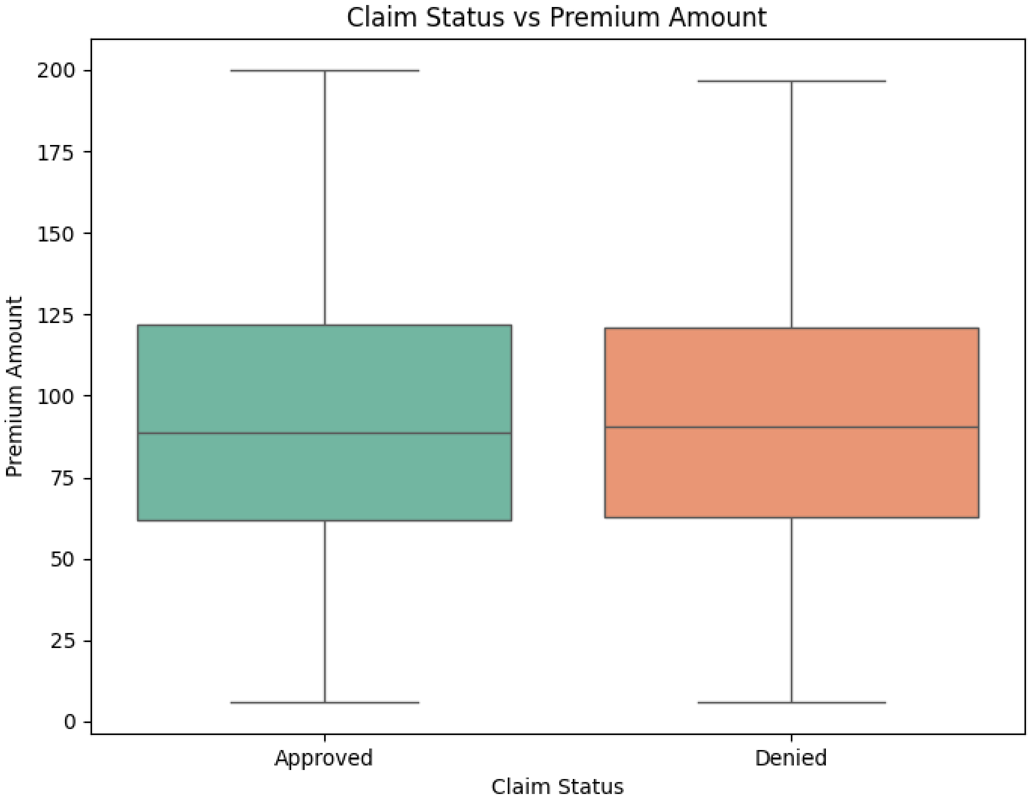


Figure A4: Claim Status vs Premium Amount