Insurance Fraud Detection: A Machine Learning Case Study

1. Introduction

Business Problem

Insurance fraud continues to be a major challenge for providers, contributing to substantial financial losses and affecting honest customers through higher premiums. Manual fraud detection methods are often time-consuming and error-prone, which creates a compelling case for leveraging data-driven approaches.

Objective

This case study explores how machine learning can be applied to detect fraudulent insurance claims using structured data. By training and comparing several classification models, both traditional and ensemble-based, **our goal was to develop a system that effectively identifies suspicious claims while minimizing false positives**.

Value Proposition

By improving the accuracy and efficiency of fraud detection, this **approach can significantly reduce financial losses**, **streamline claims processing**, and **protect the interests of honest policyholders**. Implementing such a data-driven system not only enhances operational efficiency but also contributes to a fairer, more sustainable insurance ecosystem.

2. Data Overview

The data set consists of 5,000 insurance claim records with a mix of numerical and categorical features such as Claim Type, Customer Age, Income, Claim Amount, and a binary target variable Fraudulent Claim.

a) Class Distribution

An initial inspection revealed that only 15% of the claims were labeled as fraudulent, highlighting a significant class imbalance. This imbalance can severely affect the model's ability to detect fraud, as most algorithms tend to favor the majority class. On the other hand, the claim types in the dataset are approximately equal in proportion, indicating a balanced distribution across different insurance products. However, the number of **Auto** claims is slightly higher compared to **Home** and **Life** claims, making it the most frequent claim type in the dataset.

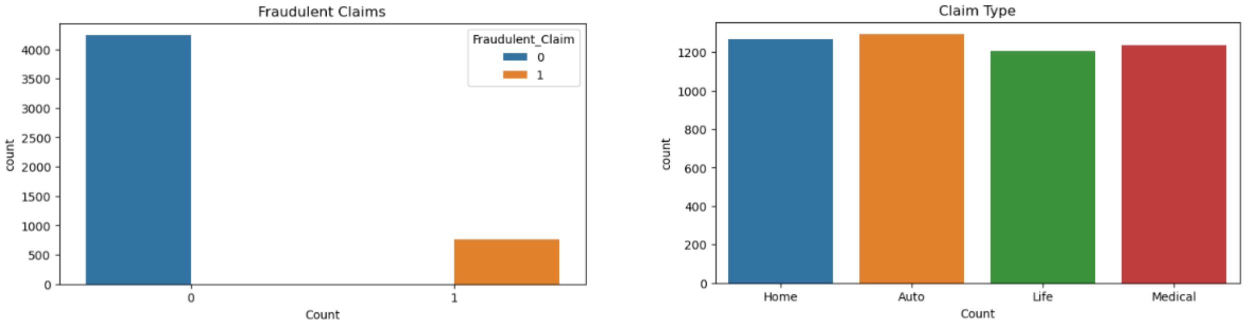

Figure 1. Class distribution and types of fraudulent and legitimate claims

b) Claim Amount and Delays vs Fraud

The analysis indicates that fraudulent claims tend to have slightly higher median amounts and more extreme upper values than legitimate ones. Although there is considerable overlap, this suggests that fraud cases may involve unusually large payouts. Fraudulent claims also appear more concentrated around mid-range submission delays, typically between 30 and 60 days. While neither claim amount nor delay is a strong predictor on its own, both variables provide valuable signals when used in combination with other features in a machine learning model.

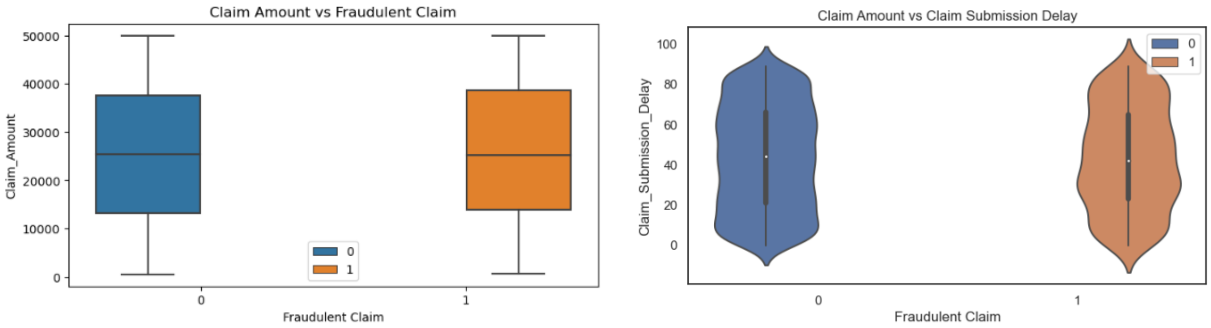

Figure 2. Amount of insurance claim and submission delay period (days) vs fraudulent claim

c) Age, Location and Income vs Fraud

There is no clear visual correlation between customer age and the likelihood of fraud, indicating that age alone is not a strong predictor. Fraudulent and legitimate claims are also evenly distributed across Rural, Urban, and Sub-Urban areas, with no noticeable geographic concentration. Similarly, customer income does not appear to differ significantly between fraudulent and legitimate claimants.

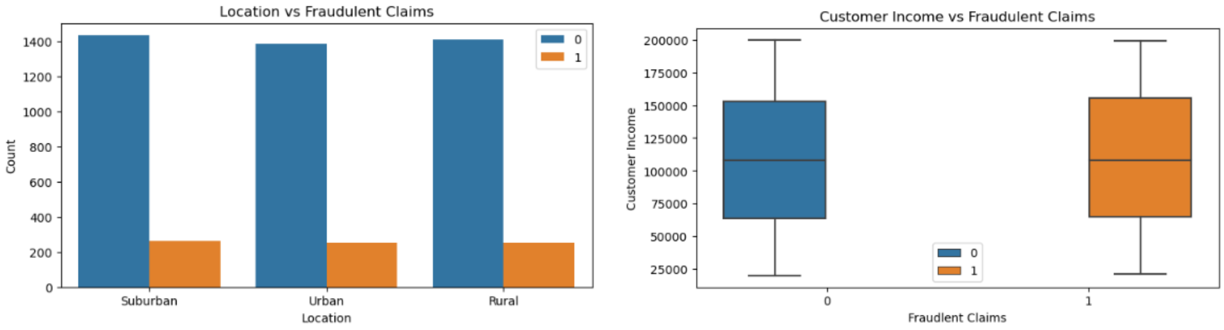

Figure 3. Customer income and location versus Fraudulent Claims

3. Data Preprocessing and Feature Engineering

The dataset was preprocessed to ensure it was suitable for machine learning. Categorical variables such as Claim Type, Marital Status, and Location were transformed using one-hot encoding. StandardScaler was applied to numerical features for models sensitive to feature scale, including KNN, SVM, and Logistic Regression. Tree-based models such as Random Forest and XGBoost were used with and without scaling, as they are inherently scale-invariant. To address class imbalance, SMOTE (Synthetic Minority Over-sampling Technique) was applied to the training set, generating synthetic examples of the minority (fraudulent) class and helping the models learn from a more balanced distribution.

4. Model Development and Evaluation

To predict fraudulent insurance claims, a range of machine learning models were developed, evaluated, and compared. The selection included both traditional algorithms (Logistic Regression, K-Nearest Neighbors, and Support Vector Machines) and ensemble-based models (Random Forest, XGBoost, and a Stacked Ensemble). A Multi-Layer Perceptron (MLP) was also tested to explore neural network-based approaches.

All models were trained using a systematic approach involving 10-fold cross-validation to ensure robust performance and reduce overfitting. Hyperparameter tuning was performed using GridSearchCV, with the F1-score used as the primary evaluation metric due to the imbalanced nature of the data, where both false positives and false negatives carry business consequences. This multi-model strategy aligns with the **No Free Lunch Theorem (NFL)**, which states that no single algorithm performs best across all problems. As such, evaluating diverse models was essential to identify the most suitable approach for fraud detection in this specific dataset.

Given the class imbalance in the dataset, two strategies were explored:

I. SMOTE (Synthetic Minority Over-sampling Technique): This technique was applied to the training data to generate synthetic examples of the minority (fraudulent) class. It enabled models to learn from a more balanced representation of fraud cases.

II. Cost-sensitive learning: For models that support class weighting (such as Logistic Regression, SVM, Random Forest, and XGBoost), class weights were applied to penalize the misclassification of the minority class more heavily. This allowed the model to give more attention to fraudulent claims without modifying the underlying data.

Model evaluation was based on multiple metrics, including accuracy, precision, recall, F1-score, and ROC AUC. However, due to the class imbalance, recall and F1-score for the fraudulent class were prioritized when assessing model effectiveness.

5. Model Performance Summary and Insights

A myriad of machine learning models were tested using both SMOTE-based resampling and cost-sensitive learning to address the significant class imbalance in fraudulent insurance claims.

Table 1. Performance metrics of different machine learning models

Model	Accuracy	Fraud Precision	Fraud Recall	Fraud F1-Score	ROC AUC	Precision-Recall AUC
KNN (SMOTE)	0.58	0.13	0.32	0.19	0.47	0.14
Logistic Regression (SMOTE)	0.48	0.15	**0.52**	0.23	0.50	0.15
SVM (SMOTE)	0.69	0.17	0.25	0.2	0.52	0.16
Random Forest (SMOTE)	**0.80**	0.17	0.08	0.11	0.49	0.15
XGBoost (SMOTE)	**0.79**	0.14	0.08	0.1	0.49	0.16
MLP (SMOTE)	0.73	0.15	0.17	0.16	0.53	0.16
Stacked Ensemble	**0.79**	0.1	0.05	0.06	0.48	0.15
Logistic Regression (Cost-Sensitive)	0.48	0.15	**0.53**	0.24	0.49	0.15
SVM (Cost-Sensitive)	0.55	0.16	**0.47**	0.24	0.49	0.16
Random Forest (Cost-Sensitive)	**0.81**	0.15	0.05	0.07	0.49	0.16
XGBoost (Cost-Sensitive)	0.6	0.15	**0.36**	0.22	0.51	0.16

The heatmap (**Figure 4**) visually highlights strengths and weaknesses of different models, **darker colors indicate better performance**. This is further supplemented by the bar chart in **figure 5**.

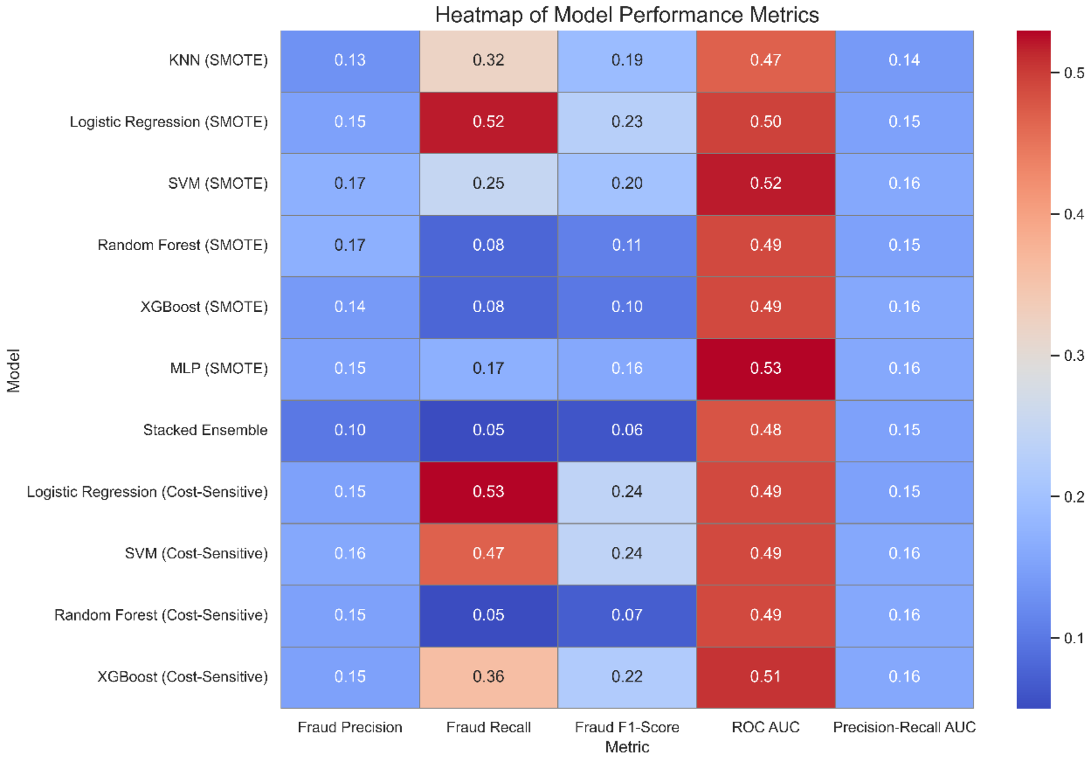

Figure 4. Heatmap showing the performance of different models. Darker colors indicate better performance.

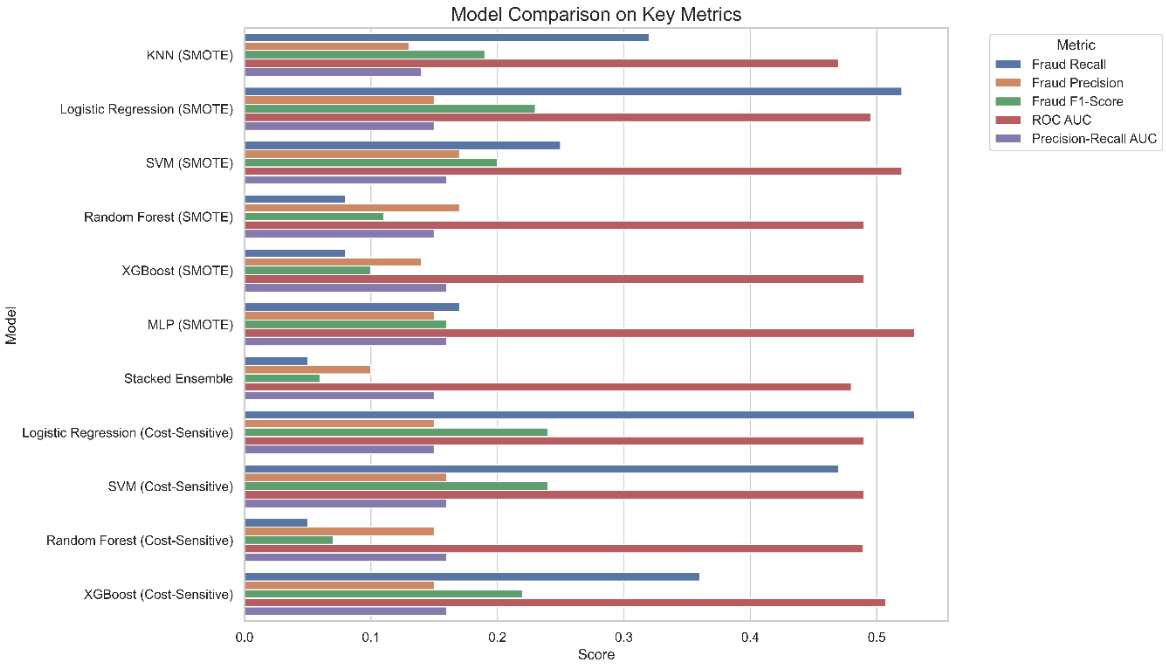

Figure 5. Bar chat showing the performance of different models and their respective metrics

Top performing models for detecting fraud (recall-focused)

I. **Logistic Regression (SMOTE)** achieved the **highest fraud recall (52%)** with a reasonable F1-score (0.23), making it a strong baseline for identifying fraudulent cases.

II. **SVM (Cost-Sensitive)** also performed well with **47% fraud recall** and a similar F1-score (0.24), showing that class weighting can be effective without altering the data distribution.

III. **XGBoost (Cost-Sensitive)** detected 36% of fraud cases and had a slightly higher ROC AUC (0.5074), offering a solid balance between model complexity and interpretability.

Models with high overall accuracy but low fraud detection

I. **Random Forest (Cost-Sensitive)** had the **highest overall accuracy (81%)** but only captured **5% of fraud cases**, highlighting a common trade-off where the model favors the majority class.

II. Similarly, **Random Forest (SMOTE)** and **XGBoost (SMOTE)** achieved high accuracy (~79%) but identified less than 10% of fraud cases.

Ensemble and neural network models

- **MLP (SMOTE)** and **Stacked Ensemble** achieved good overall performance but did not outperform simpler models in fraud recall, suggesting that more complex architectures may not yield better results under current data conditions

6. Feature Importance Analysis

Both Random Forest and SHAP (XGBoost) analyses consistently identified **Claim Amount, Customer Income**, and **Claim Submission Delay** as the top predictors of fraud **(Figure 6**). These features reflect key financial and behavioral signals.

- **High claim amounts** and **longer submission delays** were strongly associated with fraudulent claims.
- **Lower incomes** and **younger customers** showed increased fraud risk.
- SHAP provided deeper insights by showing how specific feature values (e.g., high delay, low income) pushed predictions toward fraud.

Although Random Forest ranked features by overall importance, SHAP explained their individual impact on predictions. Together, they confirm that a blend of financial behavior, demographics, and claim patterns drives fraud detection in this dataset.

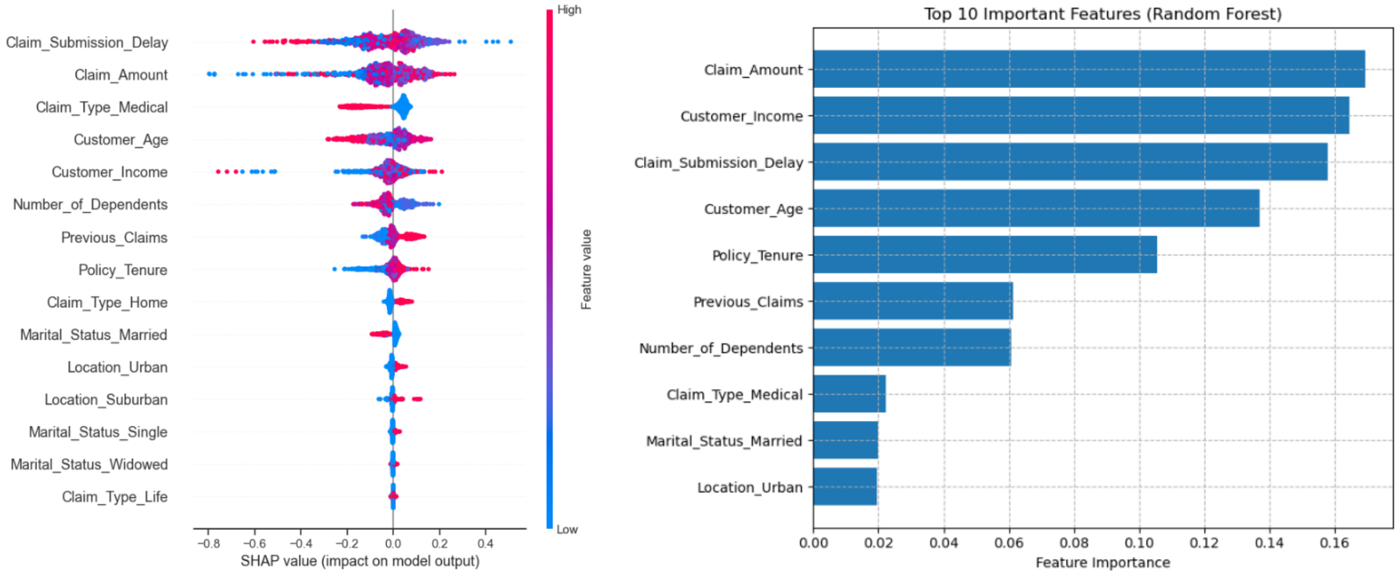

Figure 6. Top predictors of fraud as identified by XGBoost (left) and Random Forest (right) Models

7. Key Takeaway and Recommendations

Based on the evaluation of multiple machine learning models, the next steps aim to improve fraud detection while ensuring the system remains practical and reliable for business use. **Logistic Regression (SMOTE) and Support Vector Machine (SVM)** with **cost-sensitive learning** emerged as the most promising models, achieving the highest recall for fraudulent claims. These models are **recommended as baseline** options for initial deployment. To ensure continued performance, the models should not remain static. It is essential to incorporate real-time data, such as emerging claim patterns, and establish a periodic retraining schedule. This will enable the system to adapt to evolving fraud tactics and maintain its effectiveness over time.

To further enhance performance, tuning the decision threshold is necessary. This involves adjusting the probability cut-off used to classify fraud, allowing better control over the trade-off between detecting fraud and limiting false positives. Additionally, improving the quality of features can significantly strengthen model performance. New features, such as claim-to-income ratios or interactions between submission delay and claim amount, may help the model learn more meaningful patterns.

Although ensemble methods were explored, a custom voting ensemble combining the top-performing models may offer better results than a general stacked approach. This can provide more balanced predictions and reduce model variance.

Finally, due to the inherent challenge of achieving high precision in imbalanced datasets, it is advisable to use the model as a decision-support tool rather than for automated rejection. Predictions should be used to flag high-risk claims for further manual investigation by fraud analysts.