Title: Fraudulent Insurance Claim Detection

Submitted by: Saniya Baig and Chirag Panchal

Course/Module: Machine learning

Date: 11th May 2025

Table of Contents

- Problem Statement
- Objective
- Dataset Overview
- Methodology
- Exploratory Data Analysis (EDA)
- Feature Engineering
- Model Building and Evaluation
- Key Insights and Visualizations
- Assumptions
- Recommendations and Conclusion

Problem Statement

- Insurance fraud is a major challenge impacting the financial performance of insurance companies. Detecting fraudulent claims is crucial to prevent unnecessary payouts and improve operational efficiency.
- This project focuses on building a classification model to identify whether an insurance claim is fraudulent or legitimate based on historical data.

Objective

The primary objective of this assignment is to:

- Analyze historical insurance claims data.
- Identify patterns indicative of fraudulent behavior.
- Build and evaluate classification models for fraud detection.
- Compare model performances and recommend the best approach.

Dataset Overview

The dataset consists of 1000 insurance claims with 40 features, including:

- Customer demographics
- Policy information
- Claim details
- Incident descriptions
- Target variable: fraud_reported (Yes/No)

Methodology

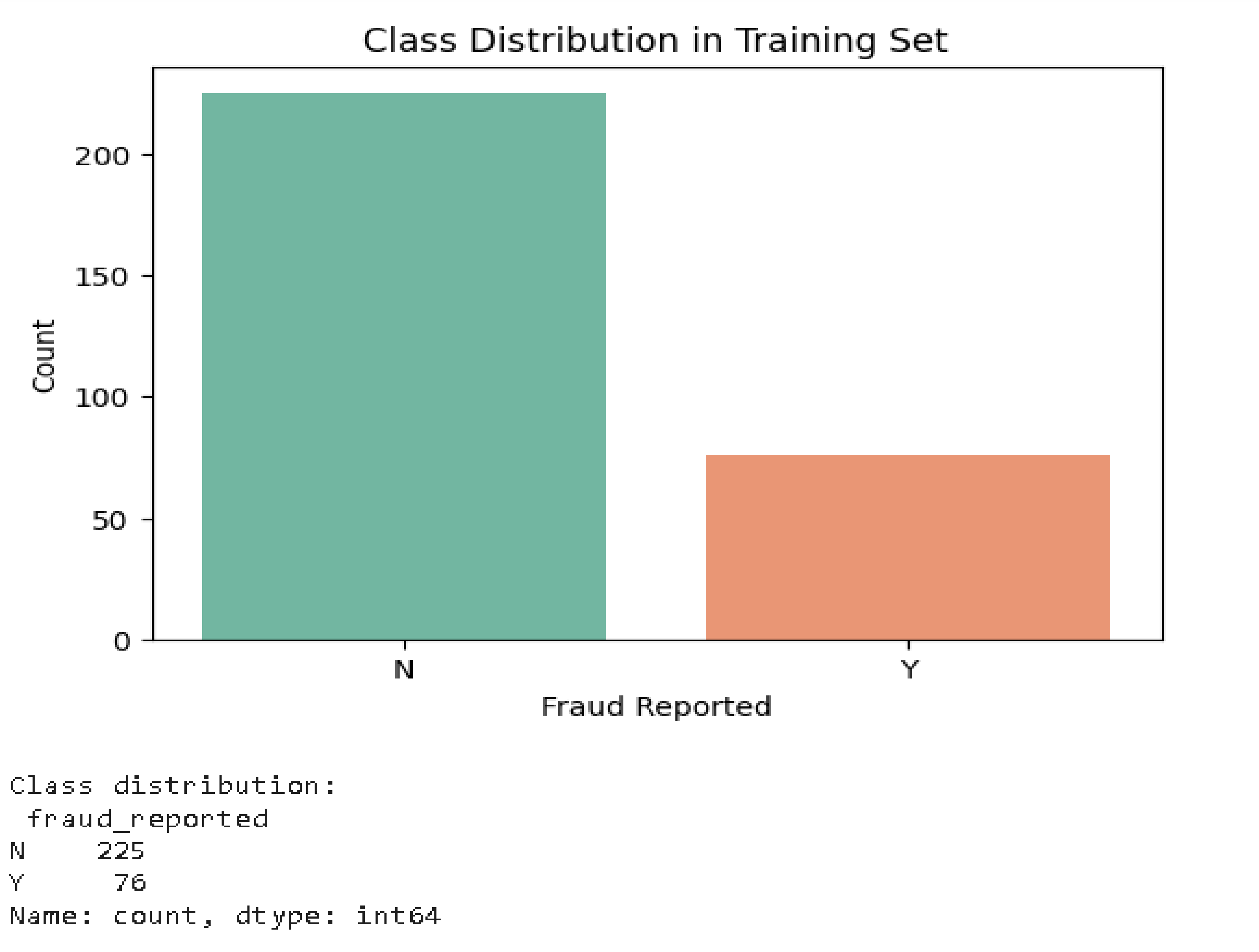
- Data Cleaning & Preprocessing: Handling missing values, encoding categorical variables.
- Exploratory Data Analysis: Visual exploration of fraud patterns.
- Feature Engineering: Creation of relevant dummy variables and selection of significant features.
- Modeling: Logistic Regression and Random Forest models were trained.
- Evaluation: Based on accuracy, precision, recall, specificity, sensitivity, F1-score.

Exploratory Data Analysis (EDA)

Key findings:

- Fraudulent claims were around 20% of total data.
- Certain states and occupations showed higher fraud rates.
- Visual trends observed in claim amounts and incident types.

➨ Insert bar plots or pie charts showing fraud distribution, state-wise incidents, etc.

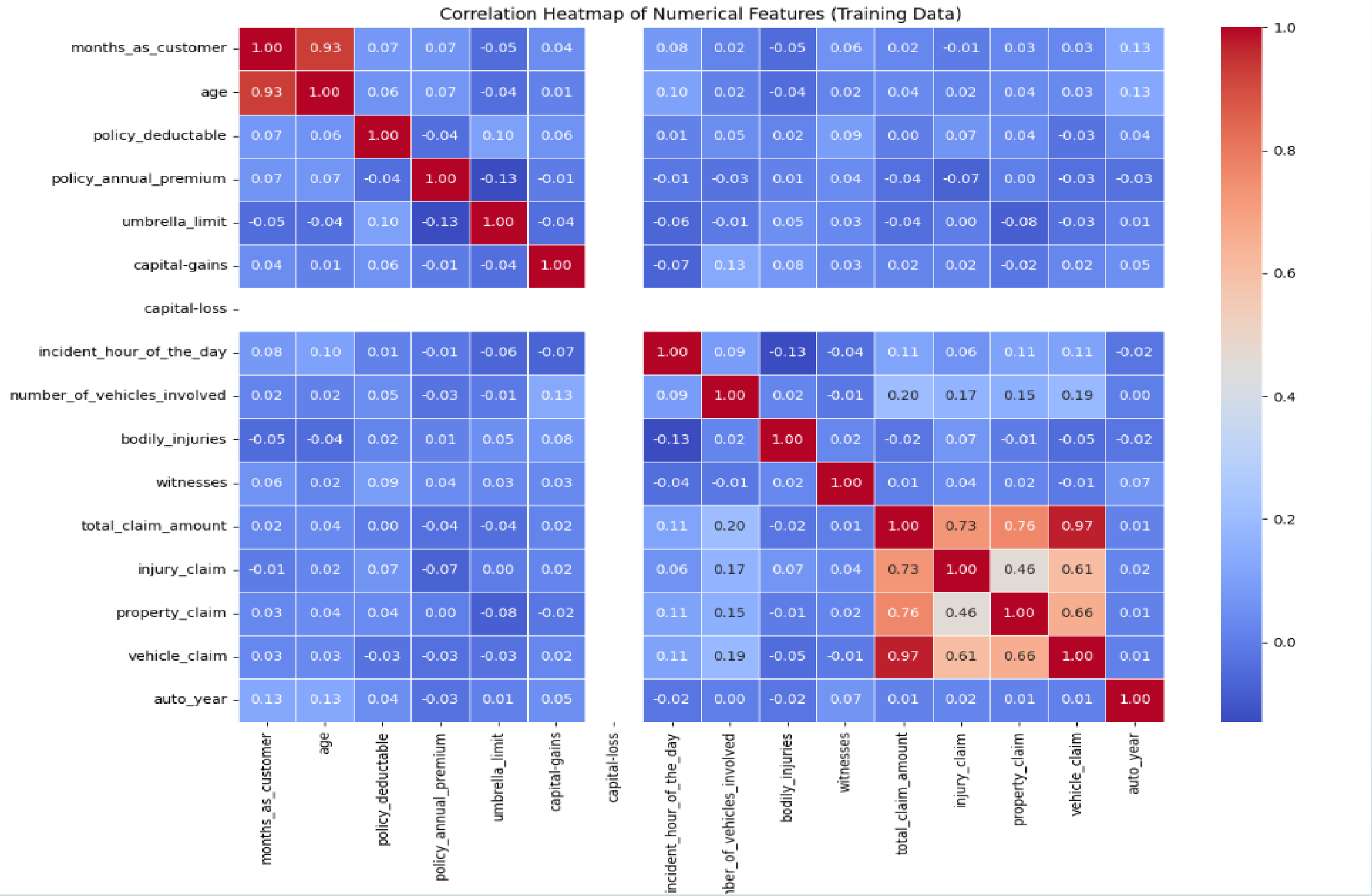


```
Class distribution:
 fraud_reported
N    225
Y     76
Name: count, dtype: int64
```

- The target variable is imbalanced, with significantly more non-fraudulent claims (around 73%) than fraudulent ones (about 27%). This may affect model performance, particularly recall for the minority class (fraud). Handling techniques such as resampling or class weighting may be considered in later steps.

Given the imbalance in the fraud_reported variable (approx. 73% No vs. 27% Yes), we will initially retain the original distribution for EDA and baseline model development. To mitigate bias during training, class weights will be applied. If performance on fraudulent claims remains poor, oversampling techniques such as SMOTE may be applied to the training data.

Correlation heatmap of numerical features



Correlation Heatmap of Numerical Features (Training Data)

	months_as_customer	age	policy_deductable	policy_annual_premium	umbrella_limit	capital-gains	capital-loss	incident_hour_of_the_day	number_of_vehicles_involved	bodily_injuries	witnesses	total_claim_amount	injury_claim	property_claim	vehicle_claim	auto_year
months_as_customer	1.00	0.93	0.07	0.07	-0.05	0.04		0.08	0.02	-0.05	0.06	0.02	-0.01	0.03	0.03	0.13
age	0.93	1.00	0.06	0.07	-0.04	0.01		0.10	0.02	-0.04	0.02	0.04	0.02	0.04	0.03	0.13
policy_deductable	0.07	0.06	1.00	-0.04	0.10	0.06		0.01	0.05	0.02	0.09	0.00	0.07	0.04	-0.03	0.04
policy_annual_premium	0.07	0.07	-0.04	1.00	-0.13	-0.01		-0.01	-0.03	0.01	0.04	-0.04	-0.07	0.00	-0.03	-0.03
umbrella_limit	-0.05	-0.04	0.10	-0.13	1.00	-0.04		-0.06	-0.01	0.05	0.03	-0.04	0.00	-0.08	-0.03	0.01
capital-gains	0.04	0.01	0.06	-0.01	-0.04	1.00		-0.07	0.13	0.08	0.03	0.02	0.02	-0.02	0.02	0.05
capital-loss																
incident_hour_of_the_day	0.08	0.10	0.01	-0.01	-0.06	-0.07		1.00	0.09	-0.13	-0.04	0.11	0.06	0.11	0.11	-0.02
number_of_vehicles_involved	0.02	0.02	0.05	-0.03	-0.01	0.13		0.09	1.00	0.02	-0.01	0.20	0.17	0.15	0.19	0.00
bodily_injuries	-0.05	-0.04	0.02	0.01	0.05	0.08		-0.13	0.02	1.00	0.02	-0.02	0.07	-0.01	-0.05	-0.02
witnesses	0.06	0.02	0.09	0.04	0.03	0.03		-0.04	-0.01	0.02	1.00	0.01	0.04	0.02	-0.01	0.07
total_claim_amount	0.02	0.04	0.00	-0.04	-0.04	0.02		0.11	0.20	-0.02	0.01	1.00	0.73	0.76	0.97	0.01
injury_claim	-0.01	0.02	0.07	-0.07	0.00	0.02		0.06	0.17	0.07	0.04	0.73	1.00	0.46	0.61	0.02
property_claim	0.03	0.04	0.04	0.00	-0.08	-0.02		0.11	0.15	-0.01	0.02	0.76	0.46	1.00	0.66	0.01
vehicle_claim	0.03	0.03	-0.03	-0.03	-0.03	0.02		0.11	0.19	-0.05	-0.01	0.97	0.61	0.66	1.00	0.01
auto_year	0.13	0.13	0.04	-0.03	0.01	0.05		-0.02	0.00	-0.02	0.07	0.01	0.02	0.01	0.01	1.00

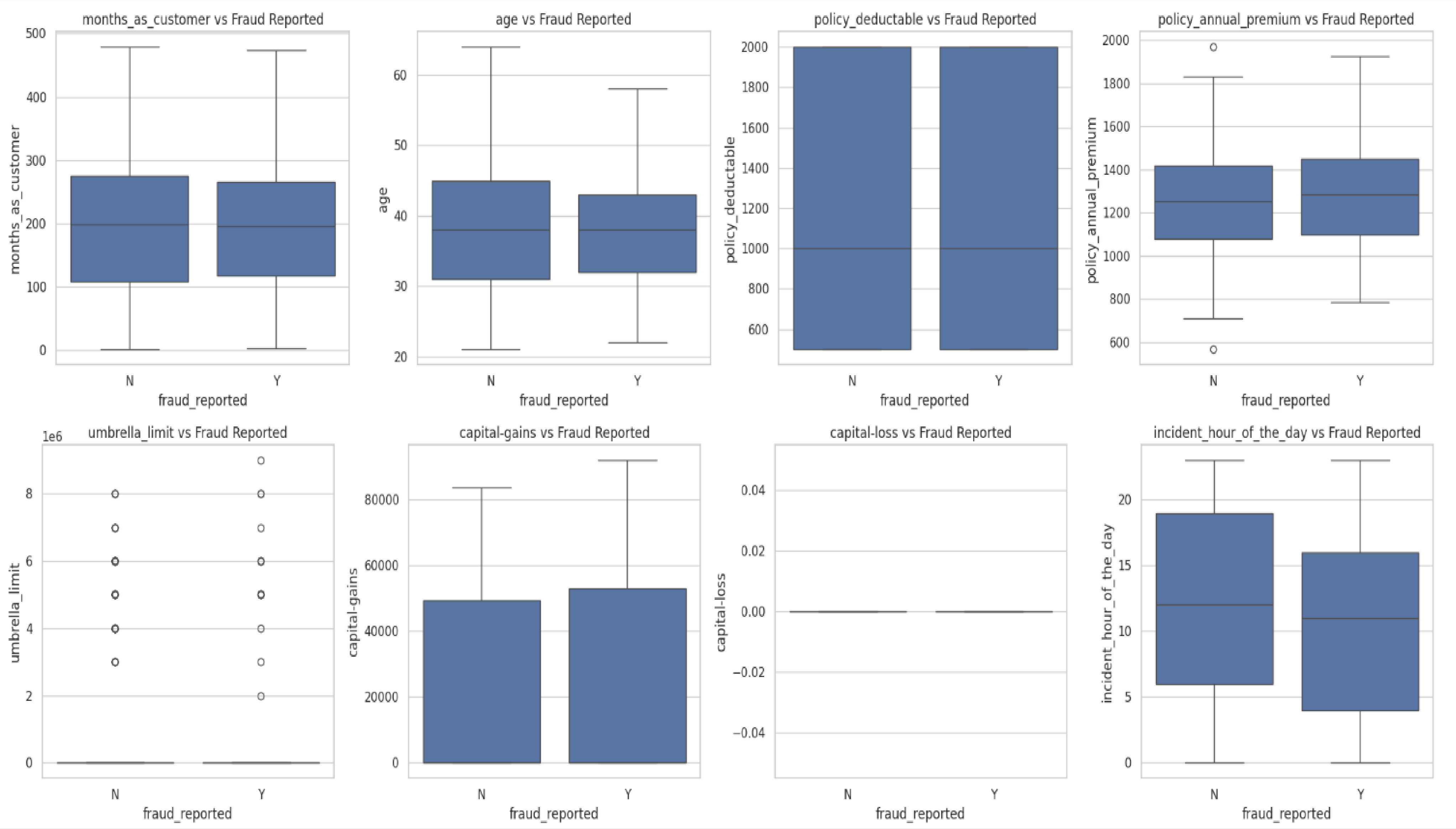
Correlations Identified:

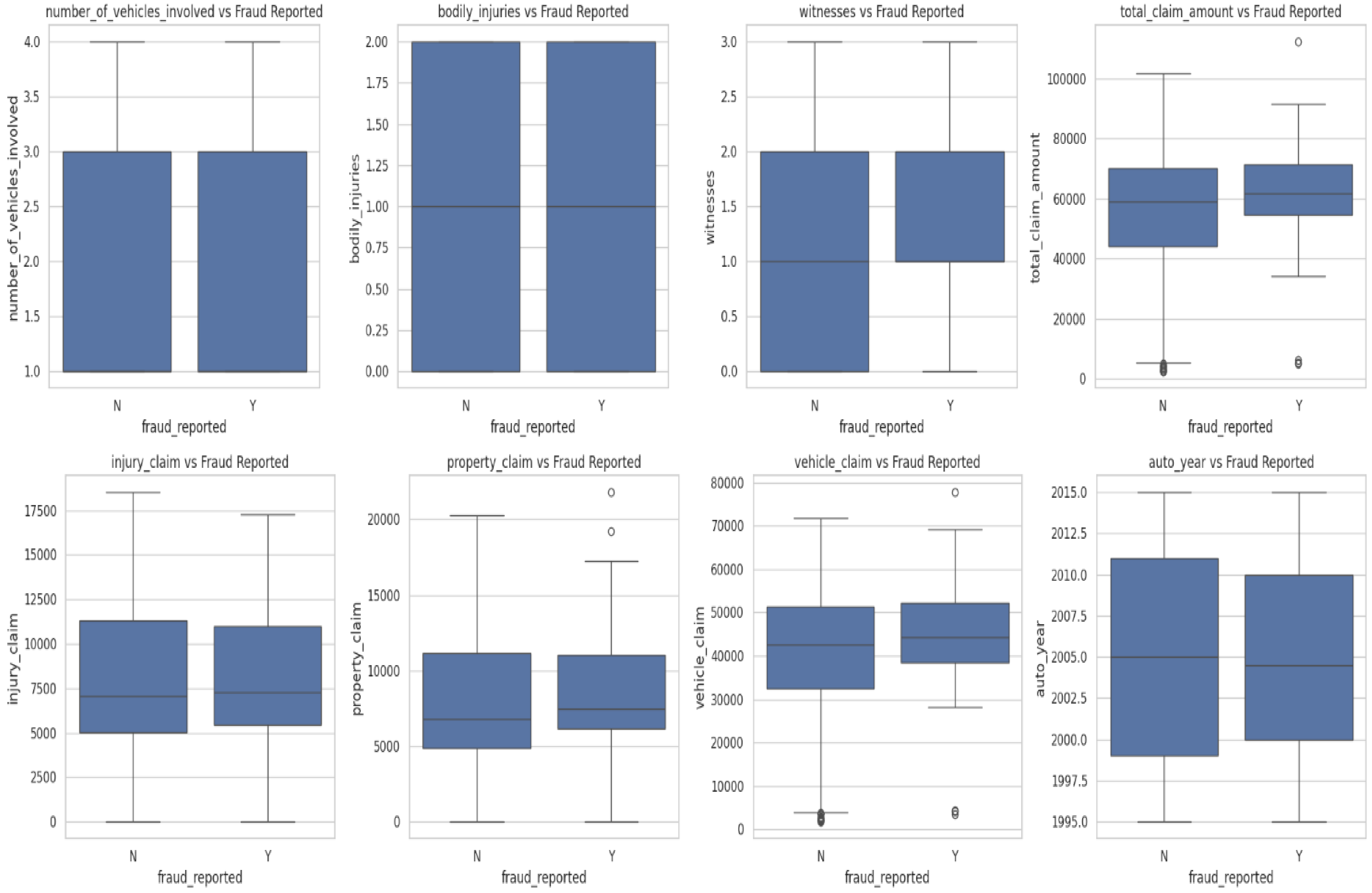
1. **Very strong correlations:**
   - csl_per_person ↔ csl_per_accident → **0.99**
   - vehicle_claim ↔ total_claim_amount → **0.98**
   - property_claim ↔ total_claim_amount → **0.77**
   - injury_claim ↔ total_claim_amount → **0.76**
2. **Strong pairwise correlations:**
   - vehicle_claim ↔ injury_claim → **0.67**
   - vehicle_claim ↔ property_claim → **0.67**
   - property_claim ↔ injury_claim → **0.47**
3. **Other notable one:** months_as_customer ↔ age → **0.92** → This is unexpectedly high and worth checking if there's data leakage or a data engineering artifact.

Feature Engineering

- Created dummies for multi-category variables like policy_csl, incident_severity.
- Selected top features using p-values and correlation with the target.
- Final selected features included:
    - policy_csl_incident_severity_250/500_Major Damage
    - policy_csl_incident_severity_500/1000_Major Damage
    - insured_occupation_handlers-cleaners
    - incident_state_OH

Box plot of numerical features vs Target variable




number_of_vehicles_involved vs Fraud Reported
number_of_vehicles_involved
1.0
1.5
2.0
2.5
3.0
3.5
4.0
N
Y
fraud_reported
bodily_injuries vs Fraud Reported
bodily_injuries
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
N
Y
fraud_reported
witnesses vs Fraud Reported
witnesses
0.0
0.5
1.0
1.5
2.0
2.5
3.0
N
Y
fraud_reported
total_claim_amount vs Fraud Reported
total_claim_amount
0
20000
40000
60000
80000
100000
N
Y
fraud_reported
injury_claim vs Fraud Reported
injury_claim
0
2500
5000
7500
10000
12500
15000
17500
N
Y
fraud_reported
property_claim vs Fraud Reported
property_claim
0
5000
10000
15000
20000
N
Y
fraud_reported
vehicle_claim vs Fraud Reported
vehicle_claim
0
10000
20000
30000
40000
50000
60000
70000
80000
N
Y
fraud_reported
auto_year vs Fraud Reported
auto_year
1995.0
1997.5
2000.0
2002.5
2005.0
2007.5
2010.0
2012.5
2015.0
N
Y
fraud_reported

1. **Strong Predictors**:
   - **total_claim_amount**, **witnesses**: Large differences between fraudulent and non-fraudulent claims, particularly with witnesses showing a **clear separation**. These features could be strong indicators of fraud.
2. **Moderate Predictors**:
   - **capital_loss**, **vehicle_claim**, **property_claim**: These features show **moderate differences** in medians and include outliers, which may suggest they have some predictive power but require further investigation.
3. **Weak Predictors**:
   - **auto_year**, **policy_annual_premium**: These features show only **slight differences** in medians between fraud and non-fraud, making them weaker predictors for the target variable.
4. **Outliers**:
   - Several features, including **vehicle_claim**, **property_claim**, **total_claim_amount**, and **umbrella_limit**, have **outliers**

Model Building and Evaluation

- **Model 1: Logistic Regression**
  - Accuracy: 79.2%
  - F1-Score: 0.526
  - Good balance of recall and precision.
- **Model 2: Random Forest**
  - Accuracy: 74.6%
  - F1-Score: Poor (due to overfitting, only predicted one class)
  - Despite high training performance, failed on validation.

➨ **Add confusion matrices and a comparison table of metrics here.**

Model Performance Comparison on Test Data

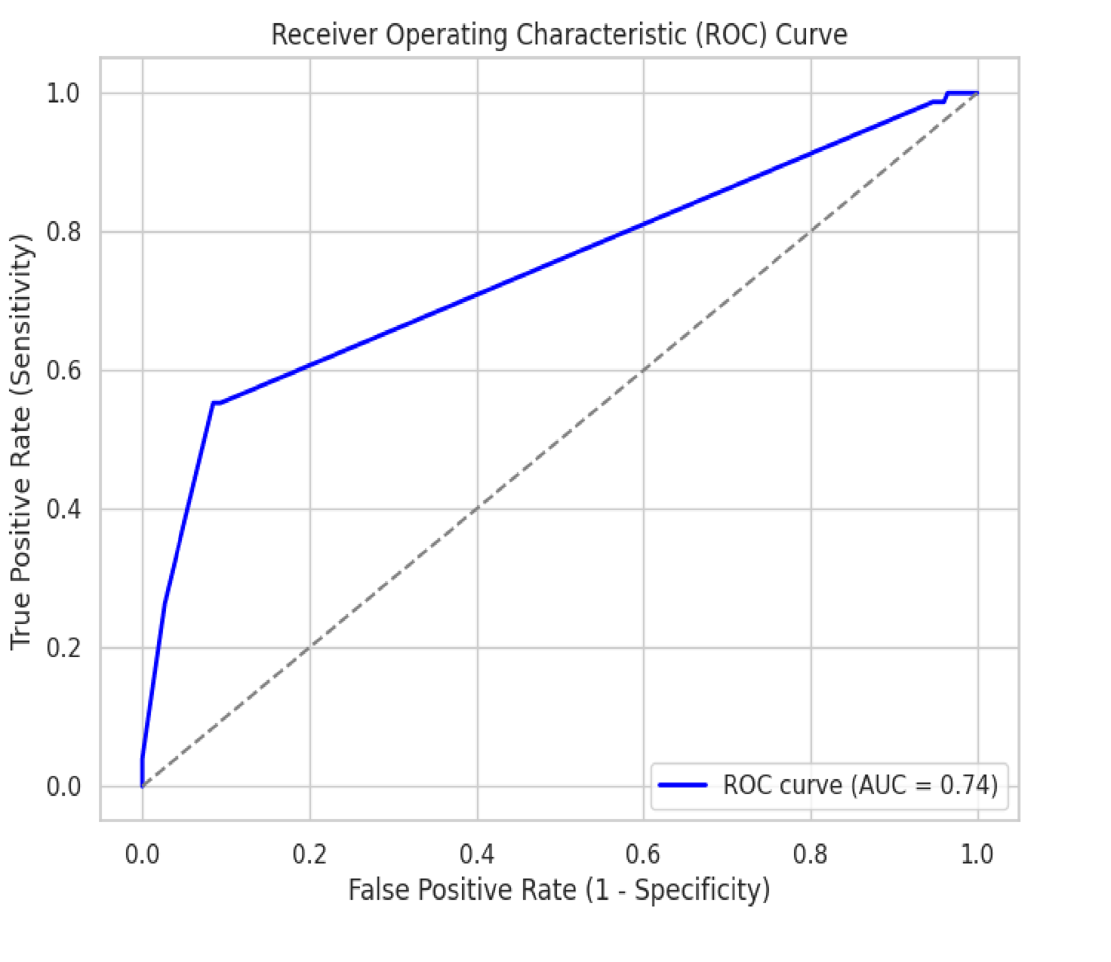
Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	0.79	0.625	0.45	0.53
Random Forest	0.75	0.00	0.00	0.00

Random Forest in terms of **precision, recall, and F1 score** on the test data. Random Forest failed to identify any positive (fraudulent) cases, leading to zero values for all key classification metrics. This indicates that it is either **overfitting** the majority class or **completely biased** toward predicting only legitimate claims. On the other hand, Logistic Regression demonstrates a much better balance, correctly identifying a reasonable portion of fraud cases while maintaining decent precision. Therefore, **Logistic Regression is the preferred model** for this task, especially given the class imbalance and the business importance of detecting fraud.

Key Insights and Visualizations

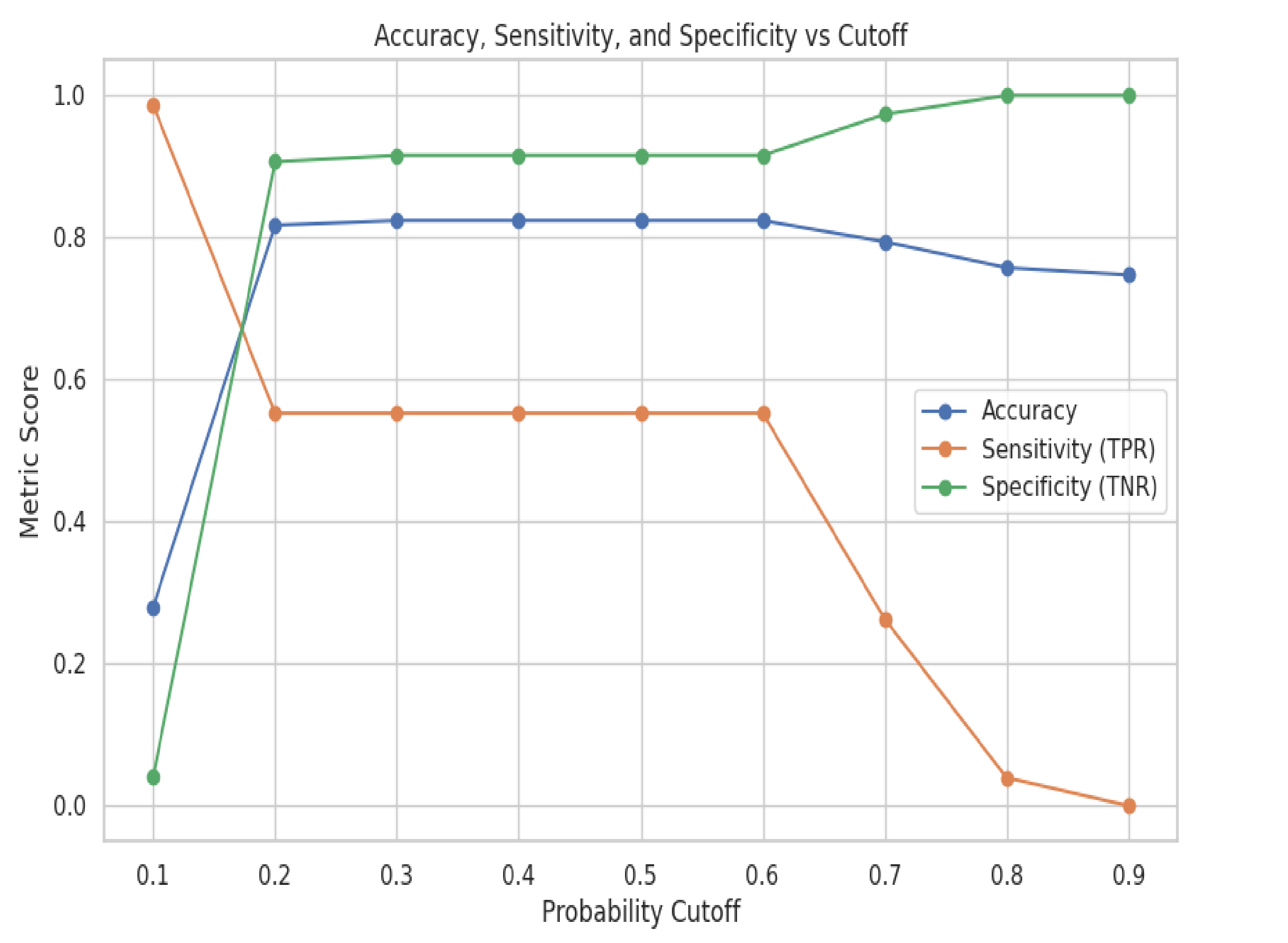
- Logistic regression outperformed Random Forest in balanced performance.
- Random Forest overfitted on training data.
- Features involving claim severity and occupation are strong fraud predictors.

ROC Curve



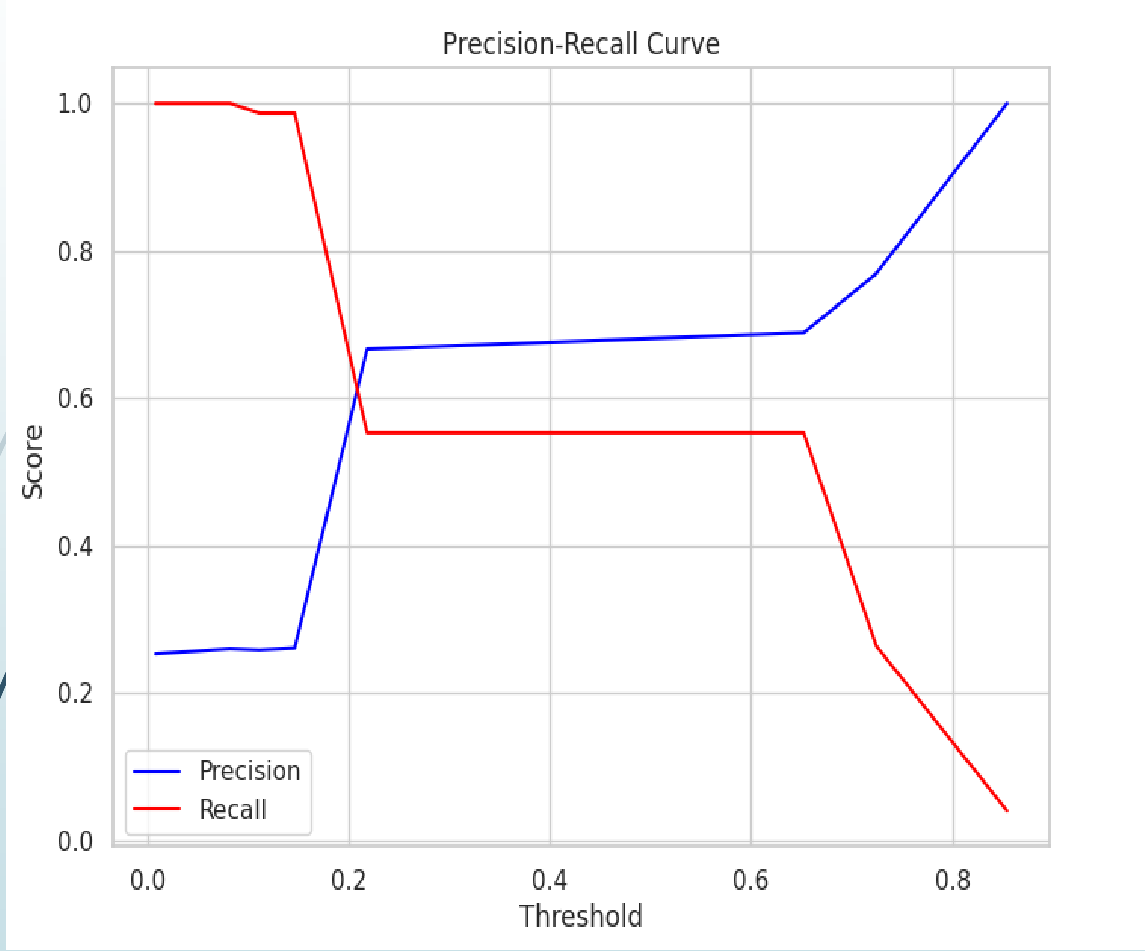
The ROC curve demonstrates that the logistic regression model has a fair ability to distinguish between fraudulent and legitimate claims, with an AUC of 0.74. This indicates a 74% probability that the model ranks fraudulent cases higher than legitimate ones. While this is a respectable performance for an initial model, especially in the presence of class imbalance, there is potential to improve the model's discriminatory power, possibly by exploring more complex algorithms, rebalancing techniques, or threshold tuning

Accuracy, sensitivity, specificity at different values of probability cutoffs



The cutoff analysis reveals that a threshold between 0.2 and 0.3 offers the best balance between sensitivity and specificity. At this range, the model maintains high accuracy (~82%), captures a reasonable portion of fraud cases (sensitivity ~55%), and correctly identifies most legitimate claims (specificity ~91%). Beyond this range, increasing the threshold improves specificity but sharply reduces the model's ability to detect fraud, making it unsuitable for a fraud detection scenario where recall is crucial

Precision-recall curve



- There's a clear **trade-off** between precision and recall.
- The **threshold range around 0.2 to 0.6** appears to give a **reasonable balance**:
  - Precision ~0.66
  - Recall ~0.55
- **Beyond 0.6**, precision improves, but **recall collapses**, making it less useful for detecting fraud.

As the threshold increases, precision improves but recall drops significantly. An optimal balance occurs around a threshold of 0.2 to 0.3, where both precision and recall are reasonably high (~0.66 and ~0.55, respectively), maximizing the F1 score. Beyond this range, recall drops too low, compromising the model's ability to detect fraud effectively."

Assumptions

- Validation data has similar distribution and features as training data.
- Cost of false negatives (fraud missed) is higher than false positives.
- The cutoff threshold for logistic regression was optimized based on F1-score.

Recommendations and Conclusion

- Use logistic regression for deployment due to better generalization.
- Focus fraud investigations on high-risk occupations and states.
- Implement a periodic retraining pipeline as fraud patterns evolve.

➨ Final Thought: A simple, interpretable model like logistic regression can offer reliable fraud detection when combined with domain insights.