Fraudulent Insurance Claim Detection

1. **Problem Statement**

Global Insure, a leading insurance company, processes thousands of claims annually. However, a significant percentage of these claims turn out to be fraudulent, resulting in considerable financial losses. The company's current process for identifying fraudulent claims involves manual inspections, which is time-consuming and inefficient. Fraudulent claims are often detected too late in the process, after the company has already paid out significant amounts. Global Insure wants to improve its fraud detection process using data-driven insights to classify claims as fraudulent or legitimate early in the approval process. This would minimise financial losses and optimise the overall claims handling process.

2. **Overall Approach & Methodology**

a. Data Understanding & Preprocessing

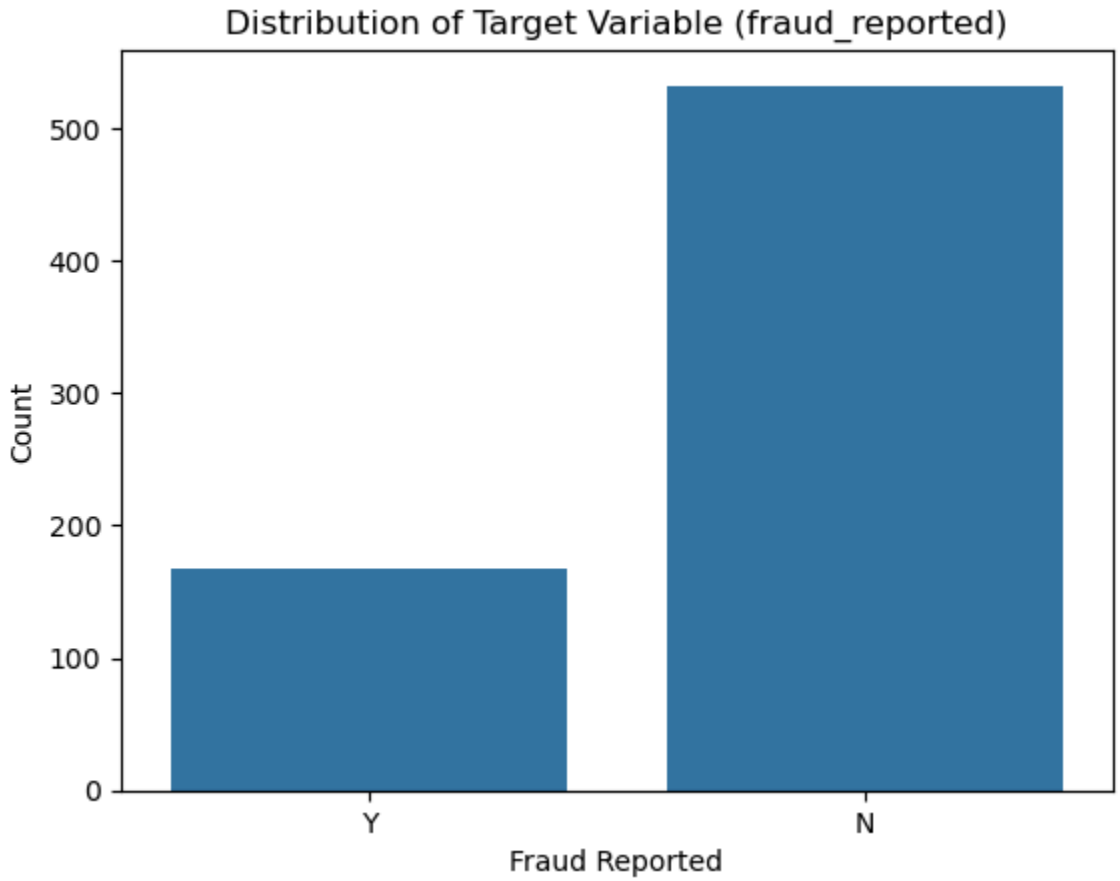
- Loaded and cleaned the dataset (insurance_claims.csv) containing 1,000 records.
- Examined the null values
- Replaced invalid or missing entries (e.g., '?') with Unknown and handled them which are present in these columns collision_type, property_damage, police_report_available.
- Dropped redundant or irrelevant columns such as policy_number, incident_location, etc.
- Created new features from date and categorical variables when applicable.

b. Assumptions

- Fraud is labeled as fraud_reported = Y/N.
- Data has class imbalance; hence accuracy alone is not enough.
- Random Forest can model complex interactions better than Logistic Regression.

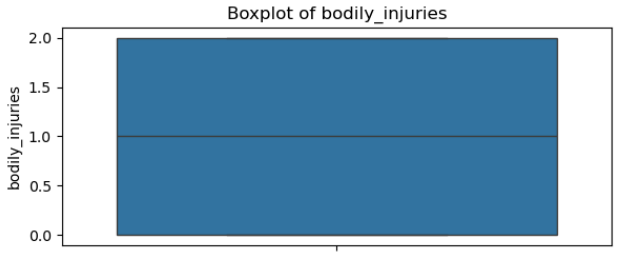
3. **Exploratory Data Analysis (EDA)**

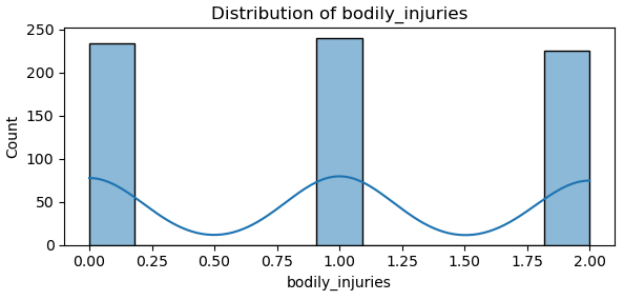
a. Fraud Distribution

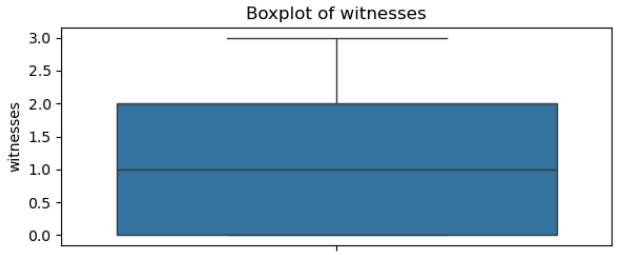


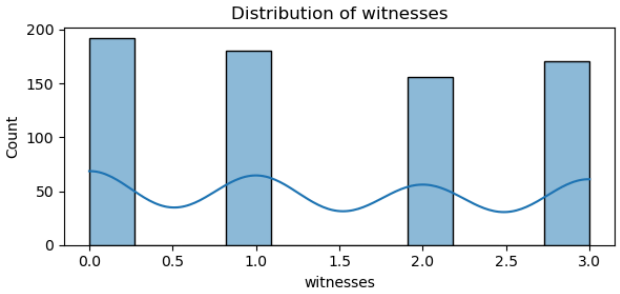
Clearly above graph is showing class imbalance approximately only 16% are reported as fraud

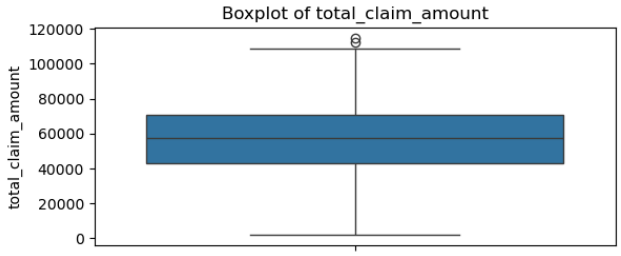
b. Univariate Analysis for Numerical Columns

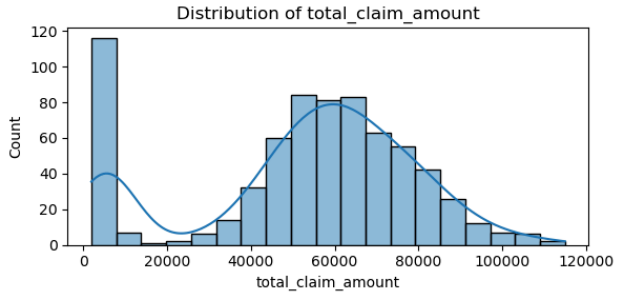


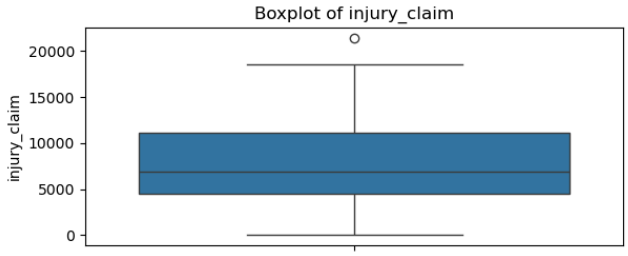


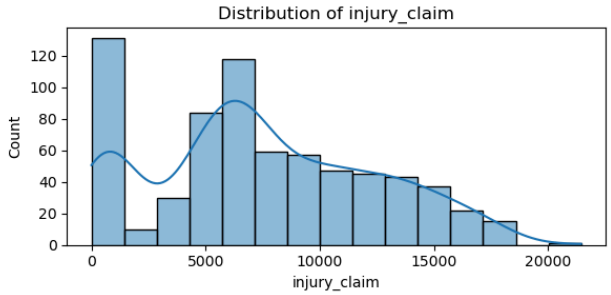


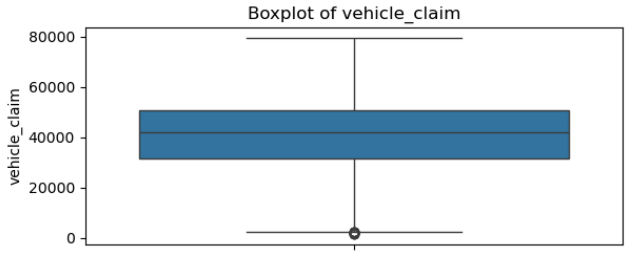


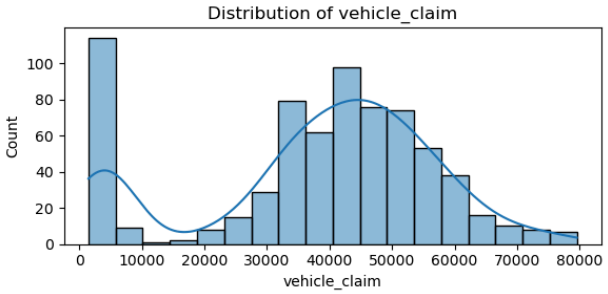


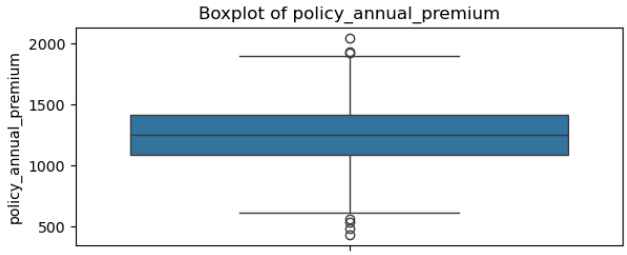


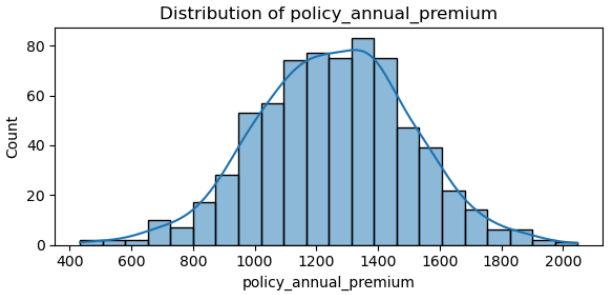


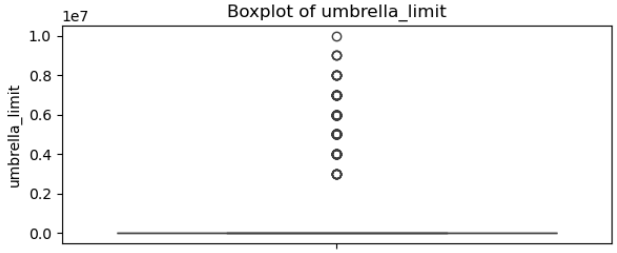


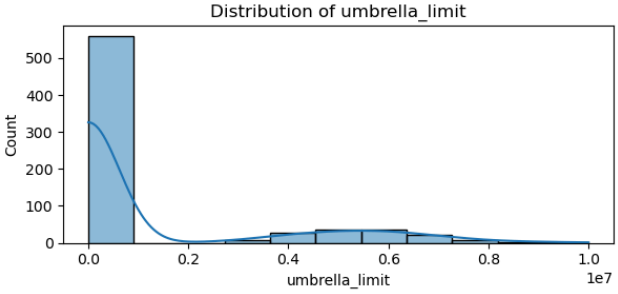

Boxplot of vehicle_claim
vehicle_claim
80000
60000
40000
20000
0

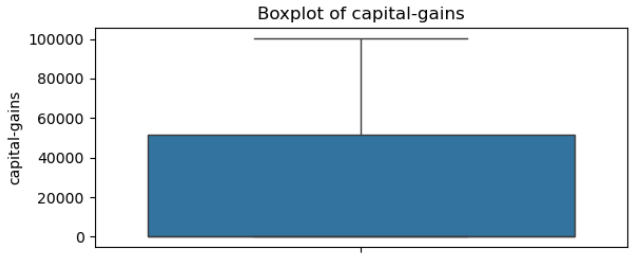

Distribution of vehicle_claim
Count
100
80
60
40
20
0
0
10000
20000
30000
40000
50000
60000
70000
80000
vehicle_claim

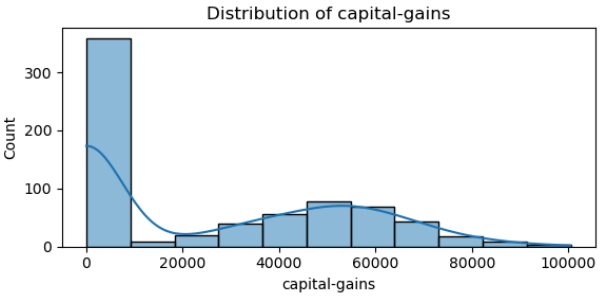

Boxplot of policy_annual_premium
policy_annual_premium
2000
1500
1000
500

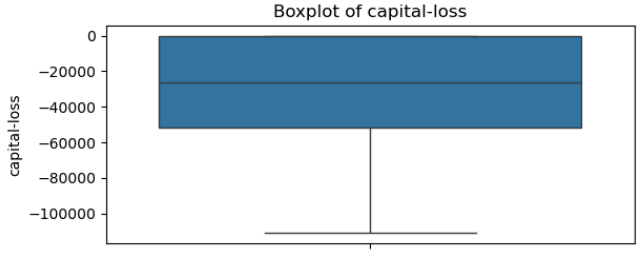

Distribution of policy_annual_premium
Count
80
60
40
20
0
400
600
800
1000
1200
1400
1600
1800
2000
policy_annual_premium

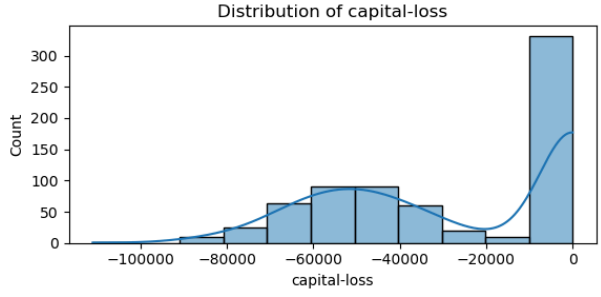

Boxplot of umbrella_limit
1e7
umbrella_limit
1.0
0.8
0.6
0.4
0.2
0.0

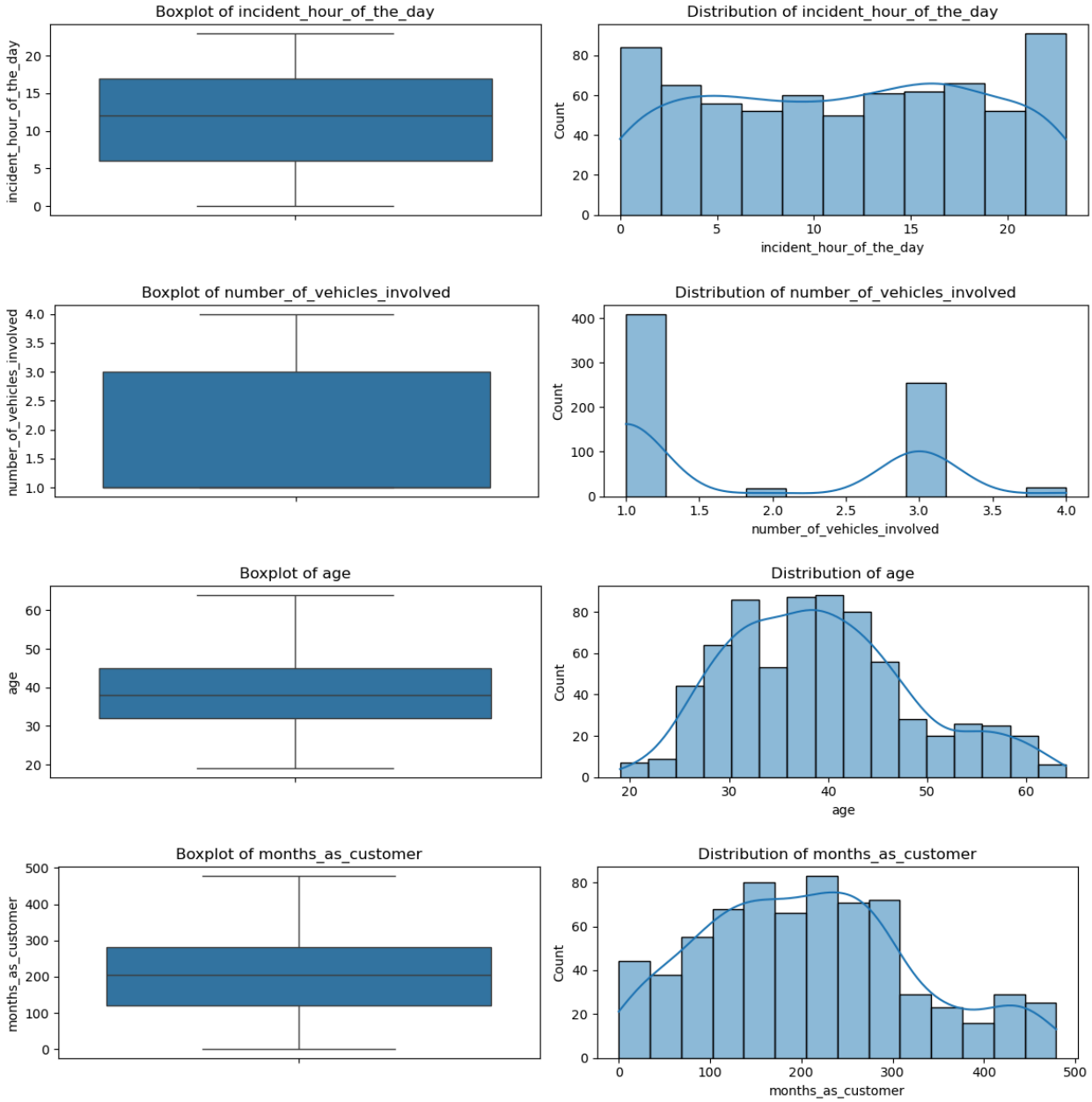

Distribution of umbrella_limit
Count
500
400
300
200
100
0
0.0
0.2
0.4
0.6
0.8
1.0
1e7
umbrella_limit


Boxplot of capital-gains
capital-gains
100000
80000
60000
40000
20000
0


Distribution of capital-gains
Count
300
200
100
0
0
20000
40000
60000
80000
100000
capital-gains

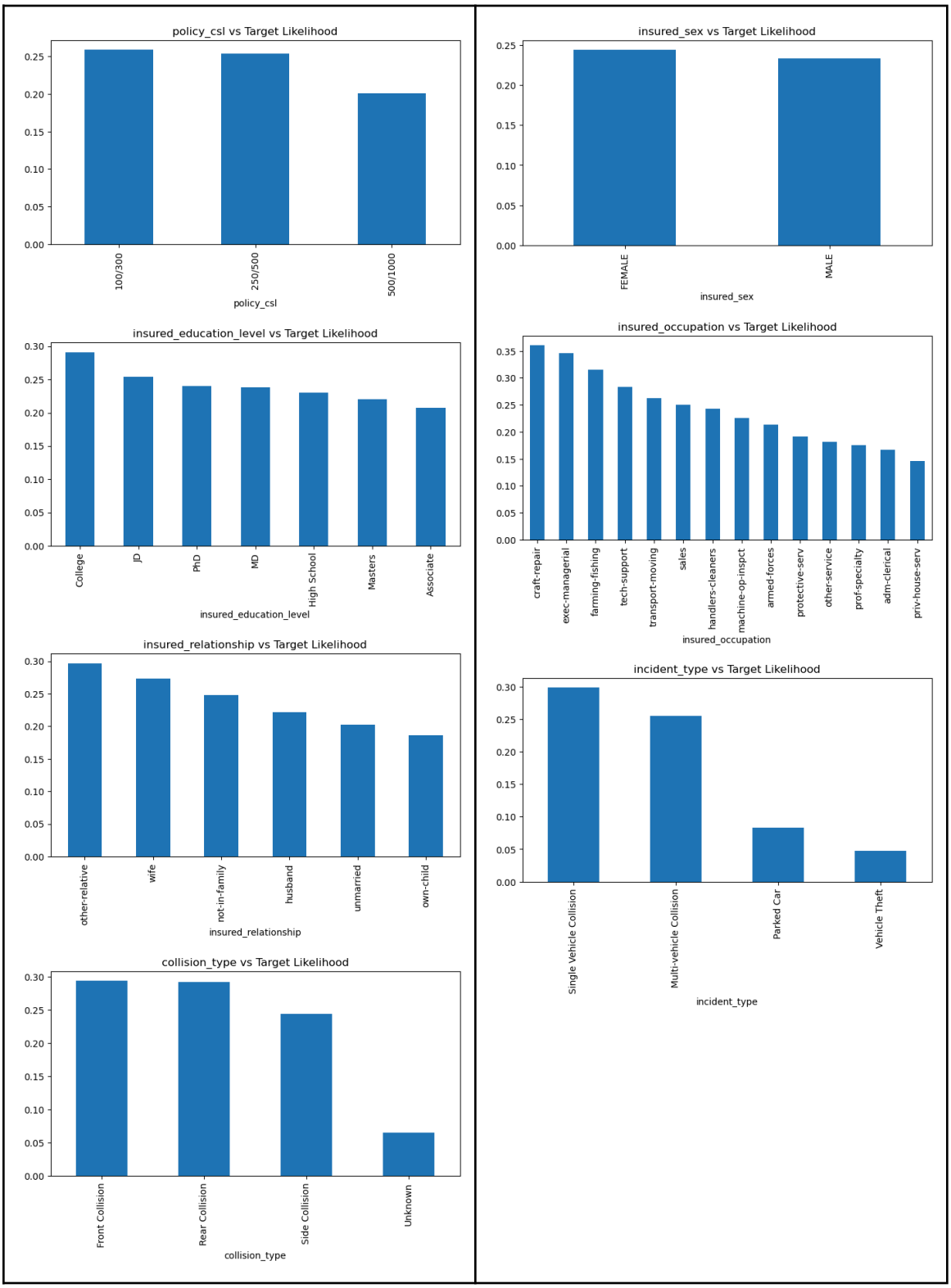

Boxplot of capital-loss
capital-loss
0
−20000
−40000
−60000
−80000
−100000

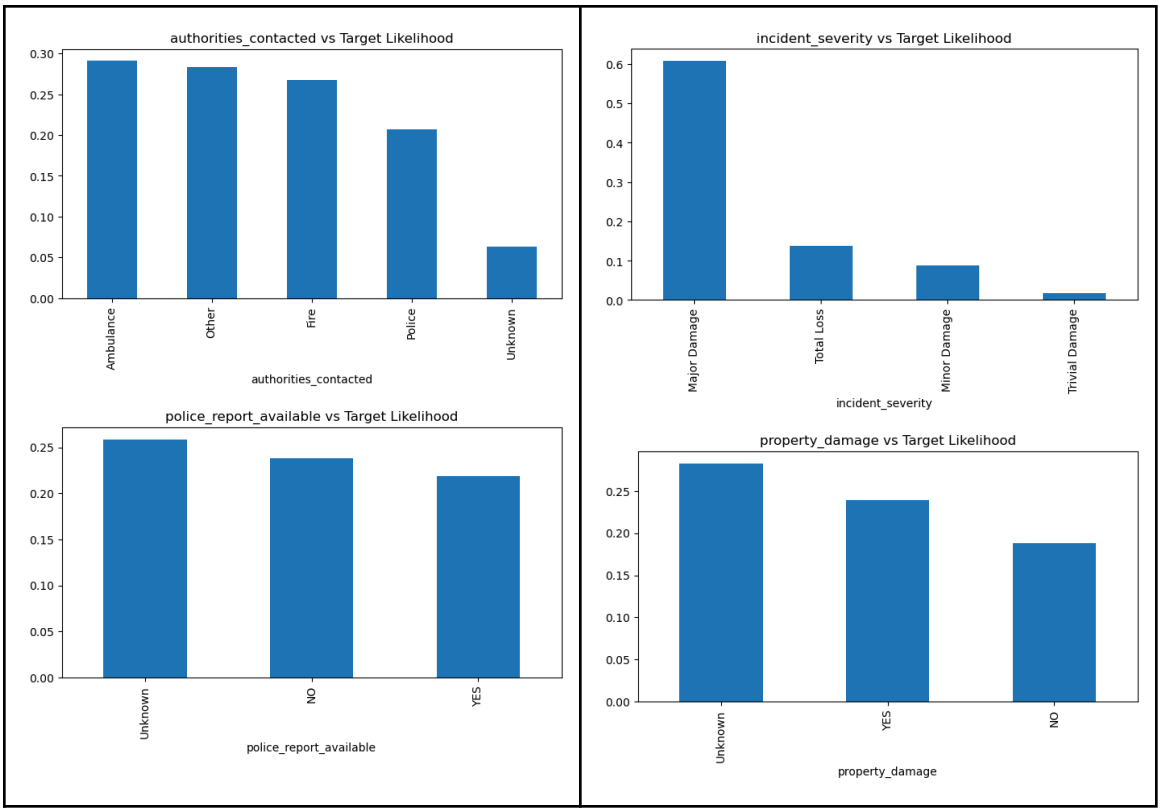

Distribution of capital-loss
Count
300
250
200
150
100
50
0
−100000
−80000
−60000
−40000
−20000
0
capital-loss



We found some outliers in umbrella_limit but we didn't removed as they are important according to data dictionary

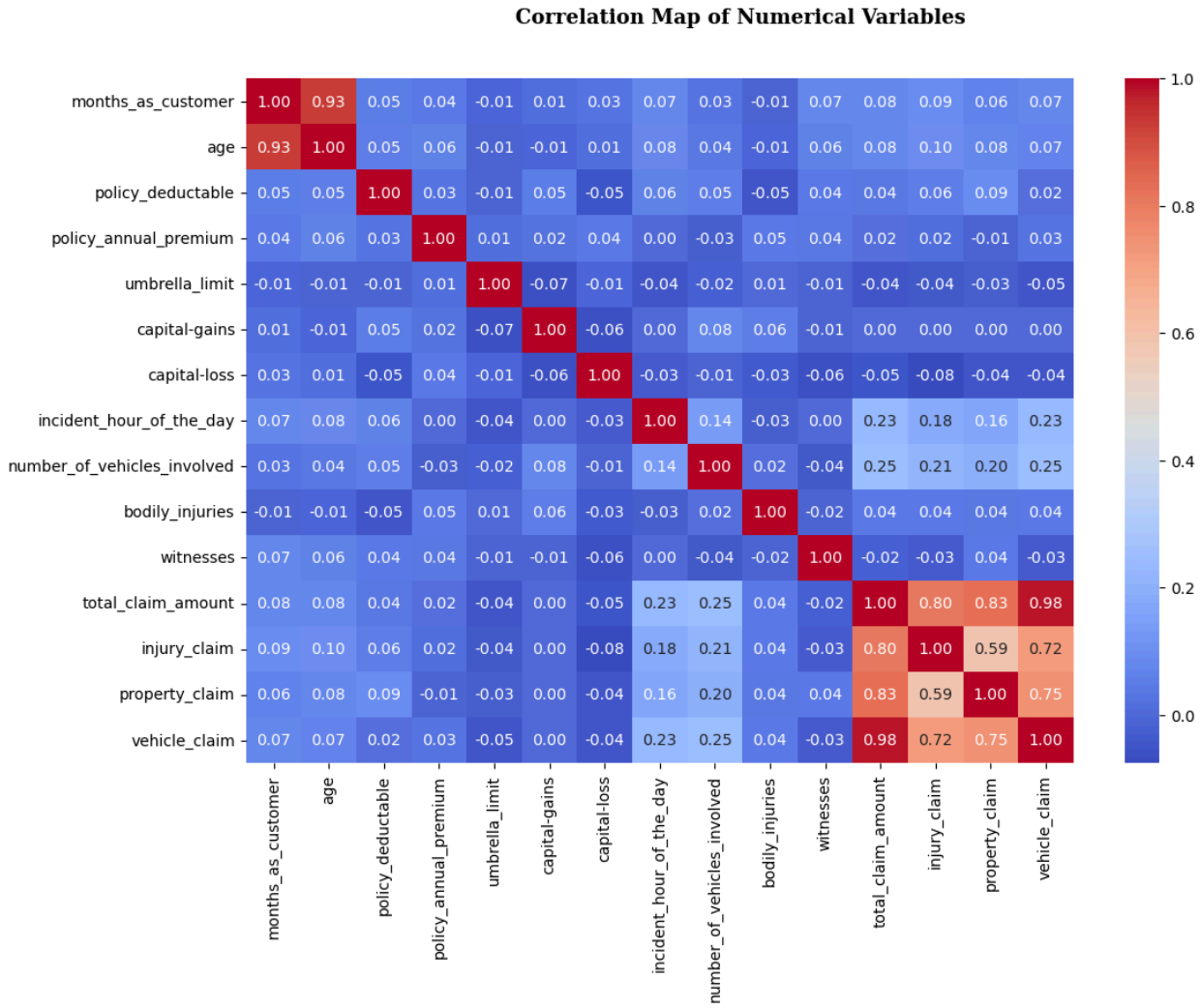
c. Bivariate Analysis for Numerical Columns





1. 100/300 and 25/500 is having high target likelihood in policy csl category
2. Almost equal target likelihood in insured sex category
3. College is having high target likelihood in insured education level category
4. Craft-repair is having high target likelihood in insured occupation category
5. Other-relative is having high target likelihood in insured relation category
6. Single Vehicle Collision is having high target likelihood in incident type category
7. Front collision and Rear collision is having high target likelihood in collision type category
8. Ambulance and Other is having high target likelihood in authorities contacted category
9. Unknown is having high target likelihood in police report available category and property damage category(Unknown is the value we added for the values of string “?”)

d. Correlation Analysis

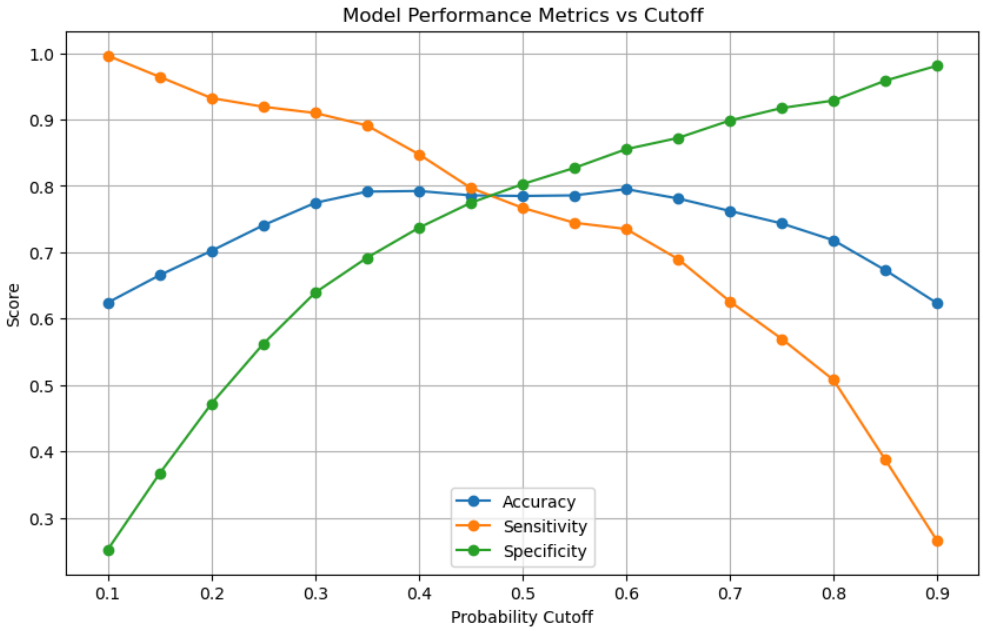


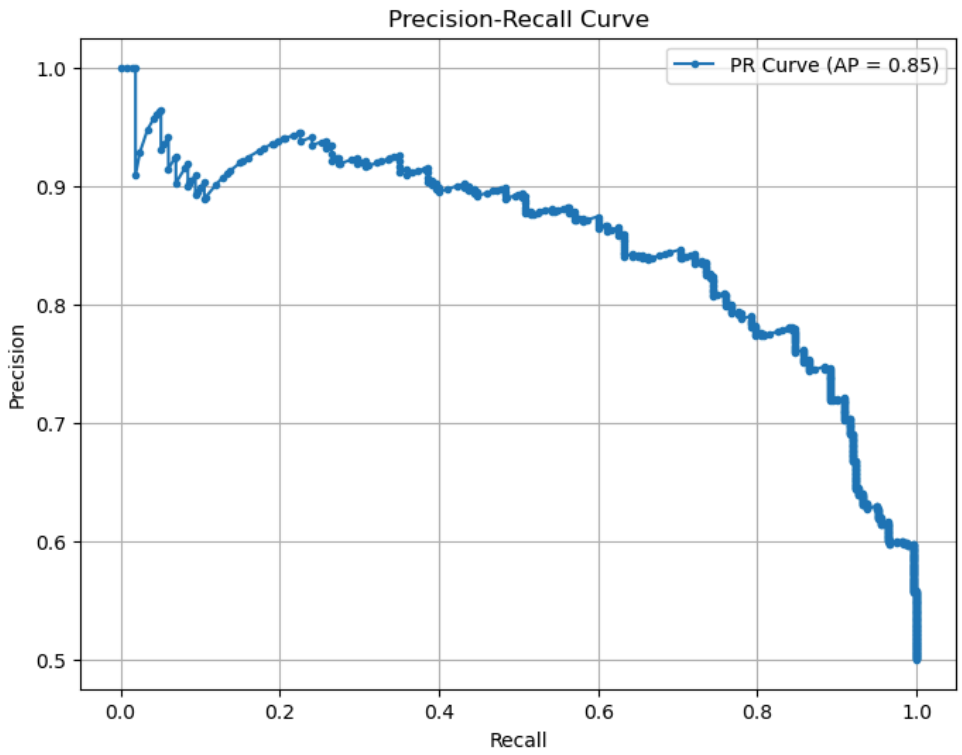
Dropped the age column as it is highly correlated column with months_as_customer

`total_claim_amount highly correlated column` was dropped as it was the sum of `injury`, `vehicle`, and `property` claims also

4. Modeling Techniques Used

Step	Logistic Regression	Random Forest
Scaling	StandardScaler	StandardScaler
Feature Selection	RFECV	Feature Importance
Evaluation	Accuracy, Precision, Recall, F1	Accuracy, Precision, Recall, F1
Hyperparameter Tuning	RFECV min 20 features	GridSearchCV (50+ combos)





- Based on Variance inflation factor and p values we also did some manual dropping rows in logistic regression
- We take optimal cut-off as 4.8 in logistic regression after plotting the `accuracy, sensitivity, and specificity at different values of probability cutoffs`
- The Precision-Recall curve starts with high precision (close to 1) and gradually lowers as recall increases.
- The Average Precision (AP) score of 0.85 indicates a strong performance — your model maintains a good balance between precision and recall.

5. Results & Comparison

Metric	Logistic Regression	Random Forest
Train Accuracy	78%	93%
Test Accuracy	73%	77%
CV Accuracy	~74%	~94%
Best Hyperparams	RFECV	{'criterion': 'gini', 'max_depth': 20, 'max_features': 'sqrt', 'min_samples_leaf': 10, 'n_estimators': 100}

6. Key Insights

- The Random Forest model significantly outperformed Logistic Regression in all key metrics.
- Features like `incident_type`, `authorities_contacted`, and `insured_relationship` were most predictive.
- Scaling and feature selection helped both models generalize better.
- Overfitting was reduced by tuning tree depth and leaf size

7. Final Recommendation

Based on the model performance and business impact, we recommend using the Random Forest model for deployment. It offers a strong balance between performance and generalization, achieving 77% test accuracy and excellent recall.