Insurance Fraud Detection using Machine Learning

A PREDICTIVE APPROACH TO FRAUDULENT CLAIMS

AUTHOR: ABDIHAKIM ISSACK

DATE: 08/12/2024

Introduction

- Objective:
- Identify fraudulent insurance claims using machine learning models.
- Reduce financial losses due to fraudulent activities.
- Scope:
- Exploratory data analysis (EDA), feature engineering, and predictive modeling.

Dataset Overview

- Dataset Details:
  - Number of records: 1000
  - Number of columns: 40
  - Target variable: fraud_reported
- Key Attributes:
  - Policy-related features: Policy Number, Policy Annual Premium.
  - Insured details: Insured Name, Insured Age, Insured Occupation,
  - Incident details: Incident Date, Incident Location, Incident State.

Data Preprocessing

- Steps Taken:
  - Removed irrelevant columns.
  - Imputed missing values.
  - Encoded categorical variables using one-hot encoding.
  - Scaled numerical variables using StandardScaler.
- Impact:
  - Cleaner dataset ready for modeling.

Exploratory Data Analysis (EDA)

- **Fraud Rates by Incident Type:**
- **Multi-vehicle Collision:** 27.2% of claims with this incident type are fraudulent.
- **Parked Car:** Only 9.5% of claims with this incident type are fraudulent.
- **Single Vehicle Collision:** 29% of claims are fraudulent.
- **Vehicle Theft:** 8.5% of claims are fraudulent, which is lower compared to other incident types.

Fraud Rates by Education Level:

- **Associate Degree:** 23.4% of claims from this group are fraudulent.
- **College:** 26.3% of claims are fraudulent.
- **High School:** 22.5% of claims are fraudulent.
- **JD:** 26.1% of claims are fraudulent.
- **MD:** 26.4% of claims are fraudulent.
- **Masters:** 22.4% of claims are fraudulent.
- **PhD:** 26.4% of claims are fraudulent.

Feature Engineering

- Techniques Used:
- One-hot encoding for categorical variables.
- Binning for policy_annual_premium.
- Removal of multicollinear features.
- Results: A robust feature set for better prediction.

Handling Class Imbalance

- Class Distribution:
  - Non-Fraudulent (N): 75.3%
  - Fraudulent (Y): 24.7%
- Challenge: Class imbalance can lead to biased predictions favoring the majority class.
- Solution: SMOTE (Synthetic Minority Over-sampling Technique) was used to oversample the minority class (fraudulent claims), improving model performance on predicting fraud.
- Result: More balanced predictions, improving the model's ability to identify fraudulent claims.

Model 1 - Logistic Regression

- Details:
- - Baseline model for comparison.
- - Simple and interpretable.
- Performance:
- Training Accuracy 0.757
- Test Accuracy 0.73
- ROC AUC Score: 0.694

Model 2 - Random Forest

- Details:
- - Captures non-linear relationships.
- - Handles high-dimensional data effectively.
- Performance:
- - Accuracy: 0.71
- - ROC AUC Score: 0.7517

Hyperparameter Tuning

- Technique: Used GridSearchCV to optimize Random Forest parameters.
- Hyperparameter tuning was applied to improve the Random Forest model's performance by optimizing key parameters. It helps to:
- **Reduce Overfitting**: Controls model complexity.
- **Improve Accuracy**: Enhances generalization to new data.
- **Address Class Imbalance**: Adjusts class weight for better handling of imbalanced data.

Model Evaluation Metrics

- Accuracy: Measures overall correctness but may be misleading with imbalanced data.
- Precision: Indicates how many of the predicted positive cases are actually positive.
- Recall: Measures how well the model identifies positive cases.
- F1-Score: Balances precision and recall for better performance evaluation.
- ROC AUC: Evaluates the model's ability to distinguish between classes, with a score closer to 1 indicating better performance.

Confusion Matrix

- The confusion matrix provides insights into the model's ability to distinguish between fraud and non-fraud cases. By analyzing it, we can assess:

- **Accuracy**: How many correct predictions were made.
- **Precision**: The accuracy of positive predictions.
- **Recall**: The model's ability to detect positive cases.
- **F1-Score**: Balance between precision and recall.

ROC Curve Analysis

- The ROC curve is useful for understanding the **trade-offs** between true positives and false positives across different threshold settings. A higher **AUC** value signifies a better-performing model, providing a better ability to distinguish between fraud and non-fraud cases.
- By using the ROC curve, we can select the optimal threshold to balance sensitivity and specificity based on the business context.

Conclusion and Future Work

- Conclusion:
- - Logistic Regression: Good baseline.
- - Random Forest with SMOTE: Best results.
- - Key metrics highlight effectiveness in fraud detection.
- Future Work:
- - Experiment with advanced models like XGBoost.
- - Incorporate cost-sensitive learning.
- - Improve feature engineering with external data sources.

Questions and Discussion

- Thank You!
- Any questions?