University of Luxembourg

UNIVERSITÉ DU LUXEMBOURG

Auto Insurance Fraud Prediction Using Machine Learning

Francis Osei

Abstract

Insurance fraud encompasses a wide range of illegal behaviors that an individual may engage in order to obtain a favorable outcome from an insurance company. Inflating claims, misrepresenting information on an insurance application, submitting claims for injuries or damage that never occurred, and staging accidents are the most common types of insurance fraud. This study proposes a framework for fraud detection in the auto insurance industry by using machine learning models. The data was cleaned before modeling, and exploratory data analysis was performed. Following that, the data was preprocessed for modeling. Three predictive models (random forest, stochastic gradient boosting, and extreme gradient boosting (XGBoost)) are applied to develop a fraud detection mechanism. The models were evaluated, and the best-fit model was selected using the F1 score, kappa, accuracy, sensitivity, and specificity. The results reveal that the XGBoost outperforms all the other models in terms of all the metrics used. The study's findings are useful for detecting fraud in the auto insurance business.

Keywords: auto-insurance, machine learning, random forest, stochastic gradient boosting, extreme gradient boosting

Introduction

With the increase in the automotive population, auto insurance has become an increasingly important sector tied to global economic growth and people's lives. It is largely responsible for covering the costs of losses caused by natural disasters and automobile accidents, including auto insurance and third-party motor vehicle liability insurance. Insurance companies, like all businesses, must address fraud and attempted fraud, which can result in significant losses and jeopardize their continued operations. Insurance fraud encompasses a wide range of illegal behaviors that an individual may engage in order to obtain a favorable outcome from an insurance company. This could include arranging the incident, misrepresenting the situation, including the relevant characters and the reason for the incident, and lastly exaggerating the level of damage caused. According to the Insurance Information Institute, insurance fraud costs vehicle insurers at least $29 billion each year in the United State. Unrecognized drivers topped the list, accounting for $10.3 billion of the total losses. Underestimated mileage ($5.4 billion), violations and accidents ($3.4 billion), and fake garaging to save on premiums ($2.9 billion) were the next most common causes of fraud.

Motor vehicle insurance scams are the most common sort of insurance fraud in the United Kingdom, causing £577 million in damages in 2021 (Association of British Insurers). This account for 60% of all fake claims caught in 2021, growing 10.7% from the previous year.
Due to the high volume of insurance claims, a vehicle insurance company cannot afford to manually verify each claim for fraud detection, therefore, machine learning and data mining methods are commonly used to detect fraud. The current study tries to categorize auto insurance fraud based on insurer attributes.

Materials and Methods

As fraud patterns evolve on a daily basis, it is critical to have a solid understanding of the technology utilized for fraud detection. As a result, this study presents a conceived comparative analysis in which multiple algorithms are applied and then classification algorithms are developed over the selected features to create the best model capable of reliably predicting fraud claims.

Description of Datasets

The data collection includes 1000 auto events and insurance claims from Indiana, Illinois, and Ohio between January 1 and March 1, 2015, with a total of 39 variables. From figure 1 and figure 2 frauds

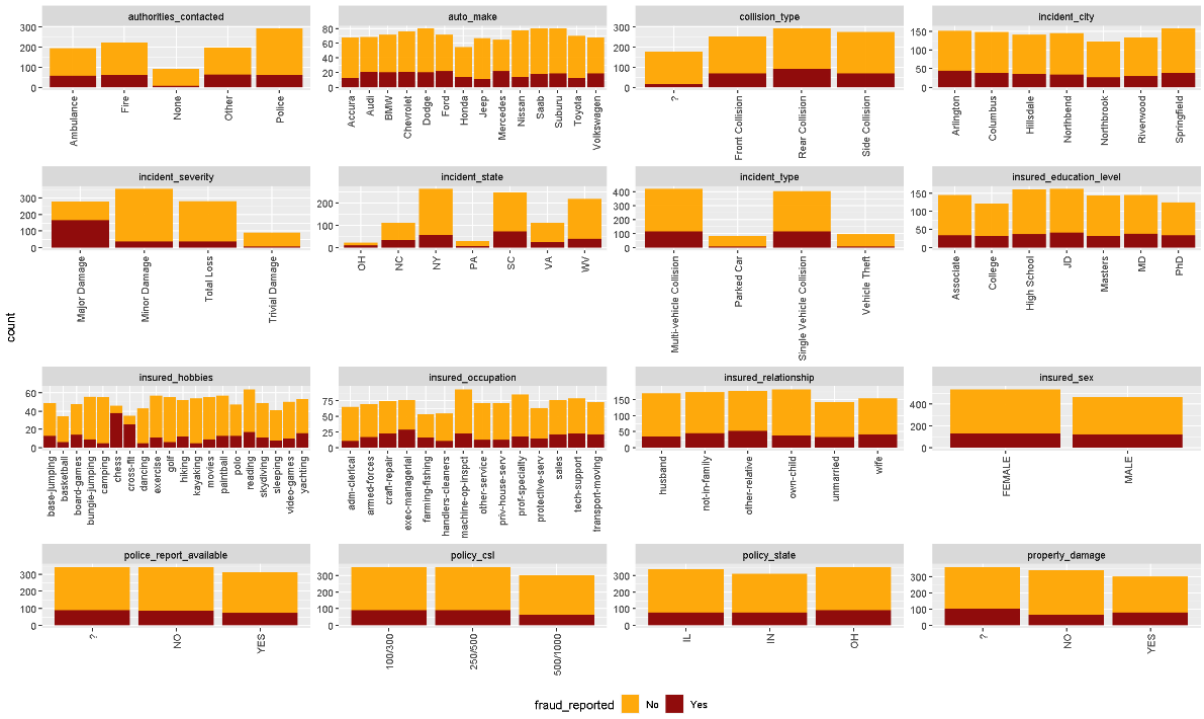

Figure 1: Distribution of some categorical features

are far less common as compared to legit insurance claims. Policyholders who play either chess or cross-fit are more in fraud cases. Cases with incident severity reported as major damages, seem to have the highest fraud cases compared to non-fraud cases. In figure 2, whiles most features are either positively skewed or negatively skewed, we observe that age and policy annual premiums are approximately normally distributed with an average age of policyholders ranging between 35-45.

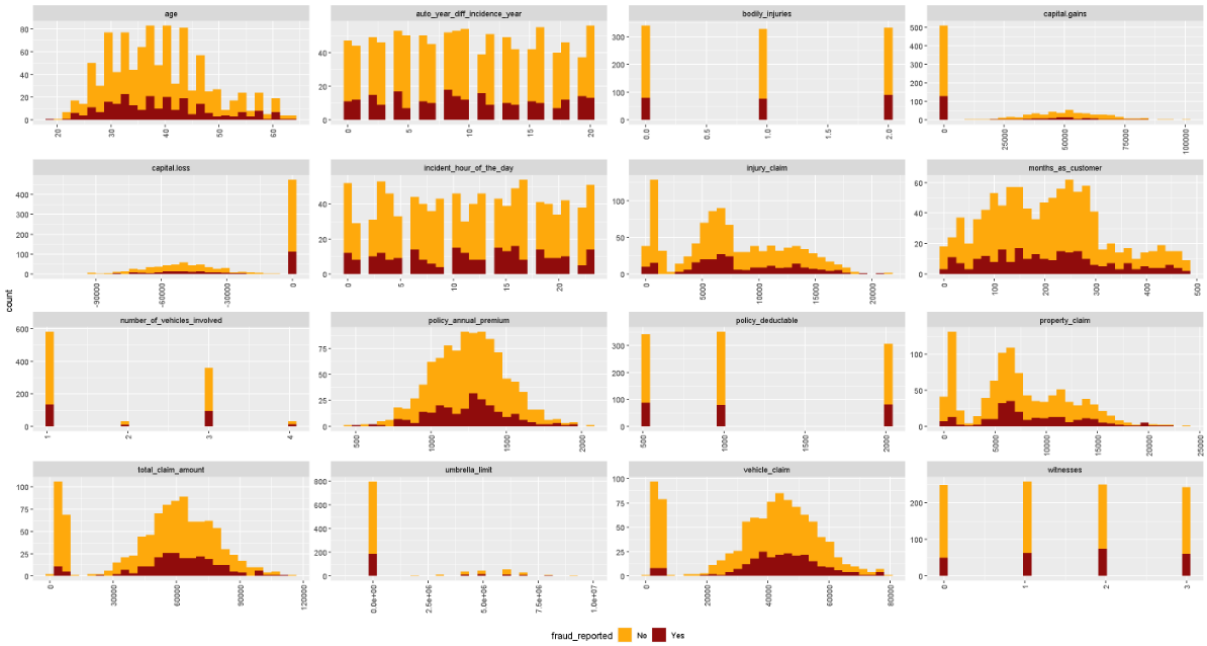

Figure 2: Distribution of numerical features

Feature Engineering and Selection

The success of machine learning algorithms is determined by how data is represented. The process of changing raw data into features that better describe the underlying problem to predictive models, resulting in enhanced model performance on unseen data, is known as feature engineering. Feature Selection is a technique for reducing the input variable to your model by using only relevant data and removing noise from the data.

Feature encoding

Two encoding techniques for our categorical features were implemented. Nominal features (no natural ordering to the categories) were handled using the one-hot-encoding while label encoding was applied to ordinal features (natural ordering to the categories). For example, the variable insured education level was labeled encoded as high school$\longrightarrow$1, Associate $\longrightarrow$2, college $\longrightarrow$3, JD$\longrightarrow$4, Masters$\longrightarrow$5, MD$\longrightarrow$6, and, PhD$\longrightarrow$7. The One-hot encoding technique converts the categorical data into numeric data by splitting the column into multiple columns. One-hot encoding policy_csl results in creating the columns policy_csl.100.300, policy_csl.250.500 and, policy_csl.500.1000 with each row containing either 0's or 1's depending on which column has what value.
Again, all data points that were represented by "?" were renamed as unknown.

Feature correlation

We compute the correlation of our new variables. When the explanatory variables which are assumed to be independent of each other are revealed to be closely related to each other, increases standard error leading to overfitting and reduced model accuracy. Figure 3 depicts top 10 with the highest correlation among features. Computing the correlation helps in feature selection by dropping one of the two features that correlate. This reduces the dimension of our dataset. The red bar in figure 3 shows that there is a negative correlation between the two features and the blue bar shows a positive correlation. Vehicle claims, the number of vehicles involved, and months as a

customer was dropped from our features. There is no threshold for the correlation value to be used to drop a column. In our case, we chose a threshold of 0.85.

Figure 3: Correlation plot of features

Data preparation

Data splitting is a technique in machine learning where data is divided into two or more subsets. One part (train data) is used for training our model and the other part (test data) is used for model evaluation. In machine learning, data splitting is typically done to avoid overfitting. In this study, a random sample split of 70/30 was considered, 70% of the data would be used for the training model, with 20% of the data being used for the testing model. Figure 4 shows that non-fraud cases make up a large proportion (76%) of the training dataset. The problem with imbalanced data is that the model being trained will be dominated by the majority, and so will predict the majority class (non-fraud) more effectively than the minority class (fraud), resulting in a high specificity rate and a low sensitivity. The simple technique to reduce the negative impact of this problem is

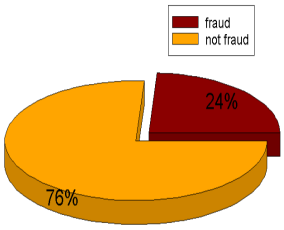

Figure 4: Unbalance class

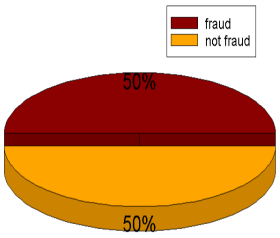

Figure 5: Balance class

by subsampling the data. Subsampling the data is a basic strategy for reducing the detrimental impact of this situation. In this research, the upsampling method is used to increase the size of the minority class by sampling with replacement so that the classes will have the same size. This method is applied to only the training data to avoid data leakage (information from outside the training dataset is used to create the model).

Machine Learning Models and Performance Metrics

In this study, three machine learning models i.e., random forest (RF), stochastic gradient boosting (sgb), and extreme gradient boosting (XGBoost) are implemented for classification purposes. The main objective is to determine the most robust machine learning models by comparing their performance on the test dataset for fraud detection.

Random Forest

A decision tree is a flowchart-like structure in which each internal node represents an attribute test, each branch represents the test's outcome, and each leaf node represents a class label.The random forest algorithm is made up of a collection of decision trees, and each tree in the ensemble is comprised of a data sample drawn from a training set with replacement. Decision trees consider all the possible feature splits while random forests only select a subset of those features. The advantage of using random forest includes flexibibity, easy to determine feature importance, and finally reduces overfitting.

Stochastic Gradient Boosting

The concept of boosting arose from the question of whether a poor learner could be modified to become a better learner. At each iteration, a random subsample of the training data (without replacement) is drawn from the entire training dataset. The randomly selected subsample is then used to fit the base learner rather than the entire sample. There are various variant of subsampling; Subsample rows before creating each tree, Subsample columns before considering each split and Subsample columns before creating each tree.

Extreme Gradient Boosting (XGBoost)

The primary distinction between random forests and XGBoost is in the manner in which decision trees are generated and aggregated. Unlike random forest each decision tree is built one after another. The two main benefits of XGBoost are model performance and execution speed. When compared to other gradient-boosting implementations, XGBoost is typically quick.

1 Result and Discusion

The confusion matrix and Six evolution metrics are taken into account when evaluating the effectiveness of the three machine learning models. Accuracy, AUC/ROC, precision, kappa, sensitivity, and specificity are the evaluation metrics.

Model		Actual not Fraud	Actual Fraud
Random forest			
	Predicted not Fraud	193	17
	Predicted Fraud	30	60
GBM			
	Predicted not Fraud	194	15
	Predicted Fraud	29	62
XGBoost			
	Predicted not Fraud	192	8
	Predicted Fraud	31	69

Table 1: Confusion matrix

The comparison of six matrix scores (Accuracy, auc/roc, precision, recall, kappa, sensitivity, and specificity) of all three models are reported in Table 2. The finding shows that the ratio of the number of correct predictions and the total number of predictions (accuracy) is high for the XGBoost (87%). This shows how often the XGBost classifier correctly predicts each class compared to the other classifiers. The probability that model ranks a random positive example more highly

Model	accuracy	auc/roc	precision	recall	kappa	sensitivity	specificity
Random Forest	0.8433	0.7929	0.6667	0.7792	0.6109	0.7792	0.8655
GMB	0.8533	0.8048	0.6813	0.8052	0.6372	0.8052	0.8700
XGBoost	0.87	0.8048	0.6900	0.8961	0.6897	0.8961	0.8610

Table 2: Comparison of results

than a random negative (auc/roc) is higher for both GBM and XGBost (80%). Again, the proportion to identify fraud cases that were reported as fraud (precision) is higher for XGBoost (69%) compared to the other two models. XGBoost (90%) outperform GBM and XGBost in terms of detecting the proportion of actual fraud cases that were identified correctly. Since we have imbalance clases in our test dataset, predictions at random could give a high accuracy score. Kappa compared the model accuracy to the expected accuracy based on the number of observations in each class. The highest kappa among the three predictive models is achieved by XGBost, which is 0.69%.The highest probability of detecting for a fraud cases if the case is truly a fraud (sensititvity) is achieved using the XGBoost with a value of 0.89. GMB (0.87) outperform random forest and XGBoost interms of the probability to detect non fraud cases if indeed the case is not fraud.

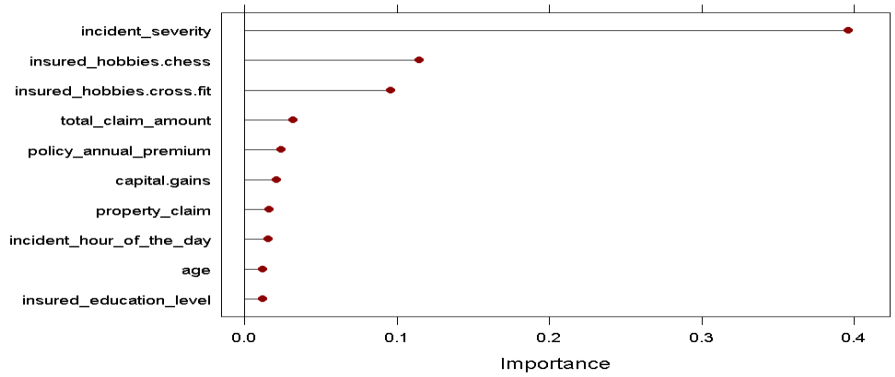

Figure 6: Vaiable Importance of the features

The top 10 features, ranked by feature importance and weight, are shown in the figure 6. One of the most crucial factors is the severity of the incident. Hobbies like CrossFit and chess are also among the top characteristics. The Variable importance is calculated by the sum of the decrease in error when split by a variable.

Conclusion

An auto insurance fraud detection model has been developed as part of this project. The model can lower losses for insurance companies in this way. The difficulty with machine learning for fraud detection is that fraudulent insurance claims are much less frequent than legitimate ones. Upsampling techniques were used to solve the problem of imbalanced classes. Three different classifiers were used in this project: random forest, stochastic gradient boosting, and extreme gradient boosting. The best and final fitted model XGBoost yields an accuracy of 87%, auc/roc of 84%, a precision of 69% recall of 90%, kappa of 69%, and sensitivity of 90%. GMB and random forest outperforms XGBoosting in detecting non-fraud cases if indeed the case is not a fraud.
Long-term expansion of auto insurance companies would be made possible by the anticipated advantages of machine learning implementation. The overall objective of this study is to facilitate human investigators' work in the auto insurance sector, leading to more accurate fraud detection,

identification, and examination. Therefore, our suggested framework may help insurance companies select the models and features for advanced machine learning and artificial intelligence-based fraud detection methodologies.

References

R. A. Bauder, T. M. Khoshgoftaar (2017), *Medicare Fraud Detection Using Machine Learning Methods.*
J. Perols (2011), *Financial Statement Fraud Detection: An Analysis of Statistical and Machine Learning Algorithms.*
S. Ramraj, N. Uzirb R. Sunil and S. Banerjee (2016), *Experimenting XGBoost Algorithm for Prediction and Classification of Different Datasets*
R. Blagus, L. Lusa (2015), *Boosting for high-dimensional two-class prediction.*