Auto Insurance Fraud Claim Detection

Fraudulent auto insurance accident claims are very popular in India nowaday. Insurance companies are paying very large amounts of of wrong insurance payout and this hurts their profits significantly. Also, insurance companies have to raise their insurance premiums and this also hurts their current customers. Therefore, there is a big demand for a good prediction model to detect fraudulent vehicle accident claims from insurance companies. In this project, I will create a classification model that help predict whether or not vehicle insurance accident claims in India are fraudulent and reduce the amount of wrong insurance payout for insurance companies.

1. Data Cleaning

1.1. **Data.** There are 28836 auto insurance claims in the original data which are represented by five csv files with the following information:

Insurance Claim Information: This table contains Customer ID, Type Of Incident, Type Of Collission, Severity Of Incident, Authorities Contacted, Incident State, Incident City, Incident Addresses, Incident Time, Number Of Vehicles Involved, Property Damage, Body Injuries, Witnesses, Police Reports, Amount Of Total Claims, Amount Of Injury Claim, Amount Of Property Claim, and Amount Of Vehicle Damage.

Customer Demographics: This table contains Customer ID, Insured Age, Insured ZipCode, Insured Gender, Insured Education Level, Insured Occupation, Insured Hobbies, Capital Gains, Capital Loss, and Country.

Insurance Policy: This table contains Insurance Policy Number, Customer Loyalty Period, Date Of Policy Coverage, Insurance Policy State, Policy Combined-Single Limit, Policy Deductible, Policy Annual Premium, Umbrella Limit, Insured Relationship, and Customer ID.

Vehicle Information: This table contains Customer ID, Vehicle Attribute, and Vehicle Attribute details.

Reported Fraud: This table only contains Customer ID and Reported Fraud.

1.2. **Data Cleaning.** I created a new and cleaned dataset using the following steps:

Combine all data into a table: I pivoted the Vehicle Information table into a new table to with more columns for vehicle features. Then I combined all four tables (Insurance Claim Information, Customer Demographics, Insurance Policy, and Vehicle Information) along Customer ID.

Clean some obvious data errors: I dropped the Country columns since it only contains India. Figured out different forms that represents the missed data in each column. Identify columns with small amount of missed data and then drop rows that contains those missed data. Identified and fixed columns that refect numeric data in nature but their data in incorrect data types.

Clean data in catetorical columns: Three columns (Type Of Collission, PropertyDamage, and PoliceReport) contains many missing values so we drop the entire columns. For each column we identify rare occured objects and then we drop rows containing them.

Clean data in numerical column: In each column we checked the statistics to figure out the outliers. Then we removed rows containing the outlier or replace the outliers with other appropriate values.

I finally obtains a new and cleaned data set with 27149 rows and 36 columns.

2. Exploratory Data Analysis Data Processing

Through the EDA above, we basically perform the following steps:

(1) I first explore the distribution of each variable. In the exploration, I also the showed the distribution of data of fraudulent claims and data of non-fraudulent claims separately.
(2) I explore correlations between numeric variables and identify pairs of strong correlation variables.
(3) Investigate I invested categorical variables and drop the ones with many values which are the columns: Date Of Incident, Incident Address, Date Of Policy Coverage, and VehicleID.
(4) I convert categorical variables into multiple numeric variables using order encoder and One-Hot-Encoder.
(5) I split the data into training set (80%) and testing set (20%) with predictor variables and target separately.
(6) I standized the training set and then appled the transformation into the test set.

3. Algorithms & Machine Learning

3.1. **Modeling.** Since the data set was unbalanced (25% positive and 75% negative), I chose AUC score for model evaluation and selection. By using cross validation, I obtained the following performance model results with default parameters:

Models	AUC scores
KNeighborsClassifier	0.90
Logistic regression	0.79
SVM (with polynomial kernel)	0.91
SVM (with radial basis function kernel)	0.91
Decision Tree	0.79
Random Forrest	0.91
AdaBoost with random forests	0.91
GradientBoostingClassifier	0.88
XGBoost	0.91

For the simplicity and interpretability I choose the Random forest model to proceed and then I run the Grid Search to choose the best parameters for the model.

3.2. **Model performance on the test set.** The chosen model achieves the 0.92 AUC scores. We also achieved the accuracy 92% and the precision, recall, and F1-scores as follows:

Report for the performance on the test set				
	Precision	Recall	F1-score	Support
Positive	0.92	0.98	0.95	3953
Negative	0.94	0.77	0.85	1477

3.3. **Feature Importance.** Features that came up as important in the modeling as follows:

Feature Importance		
Rank	Feature	Contribution
1	Severity Of Incident	0.091433
2	Insured Hobbies chess	0.041979
3	Amount Of Property Claim	0.041606
4	Amount Of Injury Claim	0.039432
5	Amount Of Vehicle Damage	0.038610
6	Insured ZipCode	0.037906
7	Insured Hobbies cross-fit	0.036741
8	Policy Annual Premium	0.035482
9	Date Of Incident	0.034266
10	Customer Loyalty Period	0.033565
11	Year Of Policy	0.028093
12	Insured Age	0.027673
13	Incident Time	0.027563
14	Policy Deductible	0.026136
15	Vehicle Year of Model	0.025515
16	Capital Loss	0.019094
17	Capital Gains	0.018804
18	Insured Education Level	0.016937
19	Witnesses	0.014973
20	Bodily Injuries	0.012492

4. Acknowledgements

I would like to thank my Springboard mentor Tony Paek for very helpful comments and great suggestions on this project.