An Article on Insurance Claim Fraud Detection Project

Samrat Dey
Batch - DS2312

Problem Definition

Insurance claim fraud is a persistent issue that plagues the insurance industry, costing billions of dollars annually and eroding trust in insurers. Fraudulent claims come in various forms, ranging from staged accidents and exaggerated damages to deliberate misrepresentation of information. Such fraudulent activities not only impact insurers' profitability but also drive-up premiums for honest policyholders, ultimately harming consumers.

The problem of insurance claim fraud detection revolves around identifying and mitigating fraudulent activities within the claims process. Insurers must sift through vast amounts of data and discern legitimate claims from fraudulent ones to minimize financial losses and maintain the integrity of their operations.

Fraudulent activities can occur across different types of insurance, including auto, health, property, and casualty insurance. Each type presents unique challenges and requires tailored approaches for detection. For instance, detecting fraudulent health insurance claims may involve analysing medical records and billing codes, while identifying staged accidents in auto insurance claims may rely on analysing accident reports and vehicle damage assessments.

The consequences of failing to detect insurance claim fraud are substantial. Insurers incur direct financial losses from paying out fraudulent claims, which can lead to increased premiums and reduced profitability. Moreover, fraudulent activities contribute to a culture of mistrust within the insurance industry, undermining the relationship between insurers and policyholders.

In addition to financial losses, insurance claim fraud also has broader societal implications. It places an undue burden on law enforcement agencies and regulatory bodies tasked with investigating and prosecuting fraudulent activities. Moreover, it can contribute to a rise in insurance premiums for consumers, as insurers seek to recoup losses incurred from fraudulent claims by passing on the costs to policyholders.

Detecting insurance claim fraud is a multifaceted challenge that requires a combination of data analytics, machine learning algorithms, and domain expertise. Insurers must leverage advanced technologies and analytical tools to sift through vast amounts of data and identify suspicious patterns or anomalies indicative of fraud. Moreover, they must continually adapt and evolve their detection methods to stay ahead of increasingly sophisticated fraudsters.

In this project, we are provided with a dataset which has the details of the insurance policy along with the customer details. It also has the details of the accident on the basis of which the claims have been made. We will be working with some auto insurance data to demonstrate how we can create a predictive model that predicts if an insurance claim is fraudulent or not.

Data Analysis

The relevant data including historical claims data, policyholder information is gathered from the source link https://raw.githubusercontent.com/FlipRoboTechnologies/ML_-Datasets/main/Insurance%20Claim%20Fraud%20Detection/Automobile_insurance_fraud.csv and created a dataset. This dataset has the details of the accident on the basis of which the claims have been made where both numerical and categorical data are present. Here 'fraud_reported' is the target variable which contains 2 categories (Y and N), so it will be termed as 'Classification Problem' where we need to predict if an insurance claim is fraudulent or not.

The size of the dataset is 1000 rows x 39 columns. There are three different types of data (int64, object and float64) present in the dataset and no null values are present. The table below shows the number of unique values in each column.

months_as_customer	391	age	46	policy_number	1000	policy_bind_date	951
policy_state	3	policy_csl	3	policy_deductable	3	policy_annual_premium	991
umbrella_limit	11	insured_zip	995	insured_sex	2	insured_education_level	7
insured_occupation	14	insured_hobbies	20	insured_relationship	6		
capital-gains	338	capital-loss	354	incident_date	60	incident_type	4
collision_type	4	incident_severity	4	authorities_contacted	5		
incident_state	7	incident_city	7	incident_location	1000	incident_hour_of_the_day	24
number_of_vehicles_involved	4	property_damage	3	bodily_injuries	3		
witnesses	4	police_report_available	3	total_claim_amount	763		
injury_claim	638	property_claim	626	vehicle_claim	726	auto_make	14
auto_model	39	auto_year	21	fraud_reported	2		

Statistical Summary of the numerical columns in the dataset shows-

1. The counts of all the columns are same which means there are no missing values in the dataset
2. The mean value is greater than the median (50%) in some of the columns, which means the data is skewed to right in these columns and the mean value is smaller than the median (50%) in some of the columns, which means the data is skewed to left in these columns
3. By summarizing the data, we can observe that there are difference between 75% and max in maximum number of columns, hence there are outliers present in the data

Data is then visualized with the help of count plot, bar plot and the outliers in the data are visualised with the help of box plot. In the feature engineering part, checked the skewness, aggregated data, encoded categorical variables. Checked the correlation between the features and the target variable and visualize the correlation matrix by plotting the heatmap. Visualized the correlation between label and features using bar plot.

Under Normalization and Scaling, the numerical features are scaled to ensure they have similar ranges and distributions, which helps improve the performance of machine learning models.

Variance Inflation Factor was checked in each scaled column. By checking VIF values we have found that the features causing multicollinearity problem. Here, we have found the features like age, months_as_customer, number_of_vehicles_involved and incident_type have VIF value very high which means they have high correlation with other features. So, by dropping one of the columns first and next by removing the column with high VIF, we have reduced the VIF.

EDA Concluding Remarks

The dataset comprises comprehensive details of insurance policies and customer information, including accident records for filed claims. With a blend of numerical and categorical data, the primary task revolves around predicting fraudulent insurance claims based on the provided features.

Summary of Findings:

1. Data Exploration: Initial exploration revealed a dataset of 1000 rows and 39 columns, with one being the target variable, 'fraud_reported'.
2. Data Integrity: No missing values were detected across the dataset, ensuring data completeness.
3. Skewed Distributions: Several features displayed skewness, indicating the need for preprocessing to address potential biases.
4. Outlier Identification: Outliers were identified in numerous numerical columns, necessitating further investigation and potential treatment to mitigate their impact on model performance.

Key Insights:

1. Fraud Prevalence: The dataset exhibits class imbalance, with fraudulent claims being a minority.
2. Feature Importance: Through correlation analysis, several features displayed significant correlations with the target variable, providing valuable insights for predictive modeling.
3. Model Performance: Multiple classification algorithms were trained and evaluated, with ExtraTrees Classifier emerging as the best-performing model, achieving an accuracy of 90.72%.

Recommendations:

1. Data Balancing: Employ techniques such as SMOTE to address class imbalance for more robust model training.
2. Feature Engineering: Further exploration into feature interactions and transformations may enhance predictive power and generalization.
3. Model Refinement: Fine-tuning hyperparameters through techniques like GridSearchCV can optimize model performance for better fraud detection.

Implications:

1. Operational Efficiency: Accurate fraud detection can streamline claims processing, reducing financial losses and preserving company reputation.
2. Customer Trust: Robust fraud detection mechanisms instill confidence in policyholders, fostering long-term relationships and loyalty.
3. Regulatory Compliance: Effective fraud prevention aligns with industry regulations, ensuring adherence to legal and ethical standards.

Future Directions:

1. Ensemble Techniques: Investigate the effectiveness of ensemble methods such as stacking or boosting to further enhance model performance.
2. Real-time Monitoring: Explore possibilities for implementing real-time fraud detection systems to proactively identify suspicious activities.
3. External Data Integration: Incorporate external data sources such as social media or public records to enrich feature sets and improve predictive accuracy.

Conclusion:

In conclusion, the exploratory data analysis provided valuable insights into the insurance claim fraud detection domain. By leveraging advanced analytics techniques and robust modeling approaches, insurers can bolster their fraud detection capabilities, mitigate risks, and uphold trust and integrity within the industry.

Pre-processing Pipeline

The pre-processing pipeline of the Insurance Claim Fraud Detection project involves several steps to prepare the data for modeling. Here's a breakdown of the pre-processing steps:

1. Loading the Dataset: Initially, the dataset is loaded using pandas read_csv function. The dataset contains information about insurance policies, customer details, and accident/incident information.
2. Data Exploration and Understanding:
    - Shape and Structure: Understanding the dimensions (shape) and structure (columns) of the dataset.
    - Data Types: Checking the data types of each column to identify numerical and categorical variables.
    - Missing Values: Checking for missing values in the dataset and handling them appropriately.
    - Unique Values: Exploring the number of unique values in each column to understand the diversity of data.
3. Visualization:
    - Countplots: Visualizing categorical variables using countplots to understand the distribution of categories.
    - Histograms: Plotting histograms for numerical variables to observe their distribution and identify skewness.
    - Pairplot: Using pairplot to visualize relationships between numerical variables and the target variable.
    - Boxplots: Visualizing numerical variables using boxplots to identify outliers.

4. Data Cleaning and Transformation:
    - Handling Skewness: Applying transformations (such as cuberoot) to address skewness in numerical variables.
    - Encoding Categorical Variables: Using OrdinalEncoder to convert categorical variables into numerical format.
    - Feature Scaling: Standardizing numerical features using StandardScaler to ensure all features have the same scale.
5. Feature Selection:
    - Checking Multicollinearity: Identifying and addressing multicollinearity using techniques like Variance Inflation Factor (VIF) and dropping highly correlated features.
6. Handling Class Imbalance:
    - Over-sampling: Balancing the target variable classes using SMOTE (Synthetic Minority Over-sampling Technique) to address class imbalance.
7. Train-Test Split:
    - Splitting the dataset into training and testing sets to evaluate model performance.
8. Model Building and Evaluation:
    - Classification Algorithms: Implementing various classification algorithms like Random Forest, Logistic Regression, Gradient Boosting, etc.
    - Hyperparameter Tuning: Fine-tuning the hyperparameters of the best-performing model using techniques like GridSearchCV.
    - Model Evaluation: Evaluating model performance using accuracy score, confusion matrix, and classification report.
    - Cross-Validation: Checking cross-validation scores to ensure model stability and reliability.
9. Final Model Selection and Saving:
    - Selecting the best-performing model based on accuracy scores and cross-validation results.
    - Saving the final model using joblib for future predictions.
10. Prediction:
    - Loading the saved model and making predictions on the test data.
    - Comparing predicted values with the original target values to evaluate model performance.

This pre-processing pipeline ensures that the dataset is properly prepared and the best model is selected for predicting insurance claim fraud detection accurately.

Building Machine Learning Models

The process of building machine learning models for insurance claim fraud detection involves the following steps:

- Feature Selection:

    - Used correlation analysis to identify features correlated with the target variable.
    - Visualized correlations using a heatmap and bar plot.

- Model Building:

- The data is split into training and testing sets.
- Various classification algorithms were used such as Random Forest Classifier, Logistic Regression, Gradient Boosting Classifier, Support Vector Machine (SVM) Classifier, AdaBoost Classifier, Bagging Classifier, and ExtraTrees Classifier.
- Each model's performance was evaluated using accuracy, confusion matrix, and classification report.
- Cross-validation scores are checked to assess the models' generalization performance.
- The best performing model is selected based on accuracy and cross-validation scores.

• Hyperparameter Tuning:

- Hyperparameter tuning was performed using GridSearchCV to find the optimal parameters for the chosen model (Gradient Boosting Classifier).

• Model Evaluation:

- The final model's performance was evaluated using accuracy, ROC curve, and AUC (Area Under the Curve).
- The ROC curve was plotted to visualize the model's performance.

• Model Saving and Prediction:

- The final trained model was saved using joblib.
- The saved model was loaded for making predictions on new data.

Concluding Remarks

The project successfully addressed the classification problem of identifying fraudulent insurance claims.

By leveraging machine learning techniques and thorough data analysis, the model demonstrated high accuracy in predicting fraudulent claims.

The insights gained from the analysis can help insurance companies in automating claim approval processes and detecting fraudulent activities efficiently, thus reducing financial losses and improving customer satisfaction.

Overall, the project provided valuable insights into insurance fraud detection and showcased the effectiveness of machine learning algorithms in addressing real-world challenges in the insurance industry.

* * *