**Insurance Claim Fraud Detection**

Insurance fraud is a significant issue that leads to substantial financial losses for insurance companies, estimated to be billions of dollars annually. It also results in higher premiums for policyholders and undermines the integrity of the insurance system. Detecting and preventing fraud is essential to mitigate these impacts and maintain trust in the industry.

Insurance claim fraud detection is a multifaceted process that employs a variety of techniques to identify and prevent fraudulent activities. While challenges exist, advancements in data analytics, machine learning, and other technologies continue to enhance the effectiveness and efficiency of fraud detection systems, ultimately benefiting both insurance companies and policyholders.

Insurance claim fraud detection is crucial for maintaining the financial health and stability of insurance companies. Fraudulent claims result in substantial financial losses, draining resources that could otherwise be used for legitimate claims and business growth. By implementing effective fraud detection measures, insurers can significantly reduce unnecessary payouts, thereby protecting their financial reserves and ensuring the company's long-term viability.

**Customer trust and satisfaction are also significantly enhanced through robust fraud detection systems. When customers perceive that an insurance company is committed to fairness and integrity, their trust in the insurer increases.**

Effective fraud detection systems help insurance companies to:

* Reduce Financial Losses: Minimize payouts on fraudulent claims, protecting the company’s bottom line.
* Maintain Lower Premiums: Preventing fraud helps keep insurance premiums affordable for honest policyholders.
* Enhance Trust: Build and maintain trust with customers by ensuring fair handling of claims.
* Improve Operational Efficiency: Automating fraud detection processes can streamline operations and reduce the burden on human investigators.

In this project, we are provided 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. In this example, I worked with some auto insurance data to demonstrate how you can create a predictive model that predicts if an insurance claim is fraudulent or not. This dataset consists of 1000 rows, 40 features describing each policy characteristics and target variable. fraud\_reported is target variable to be predicted. As target variable is categorial in nature, this case study falls into classification machine learning problem. We have two objectives here:

1. Which key factors result in fraud being reported?
2. Building ML Model for predicting fraud.

**Data Preparation: Load, Clean and Format**

Let’s begin with importing libraries for EDA and dataset itself.

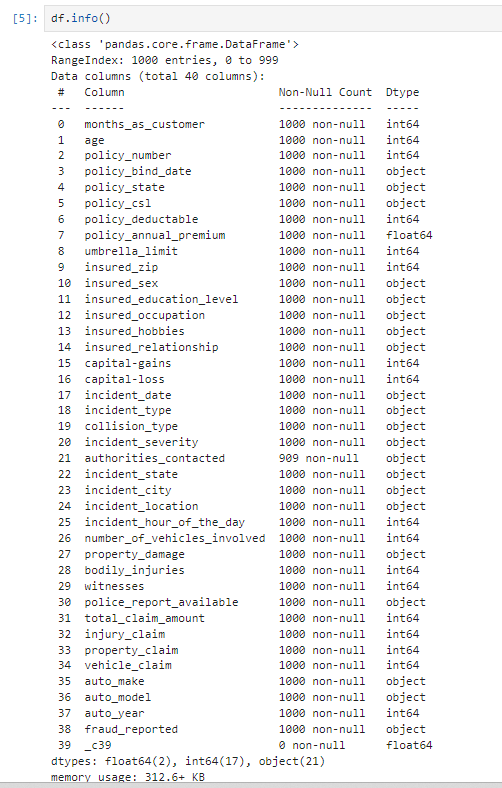
A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

Checking different datatypes in dataset: -

****

We have 21 features with object datatypes and rest are Numeric feature with int64and float64

Above nomenclature will help in better understanding of data when we perform EDA in this case study.

**Data Integrity Check:** Dataset can have missing values, duplicated entries and whitespaces. Now we will perform this integrity check of dataset.

A screenshot of a computer screen

Description automatically generated

there is missing data! So lets remove nulls and dropping unnecessary columns.

A screenshot of a computer code

Description automatically generated

Dataset doesn’t contain Any duplicate entry, whitespace, ‘NA’, or ‘-’.

A screenshot of a computer

Description automatically generated

**Splitting features**

****

**Exploratory data analysis**

**Exploratory Data Analysis refers to the critical process of performing initial investigations on data so as to discover PATTERNS, TO spot ANOMALIES, TO test hypothesis and to check assumptions with the help of summary statistics and graphical representations.**

## Univariate Analysis

A screenshot of a computer code

Description automatically generated A green bar graph with white text

Description automatically generated

'fraud\_reported' is our target variable to be predicted. From hist plot we can say dataset is imbalanced in nature. ***making our dataset to be consider as imbalanced***since much of the fraud was not reported.

In this dataset we have features like injury\_claim, property\_claim, vehicle\_claim which are inter related with each other. Let investigate this by visualisation of these features one by one to gain more insights.

****

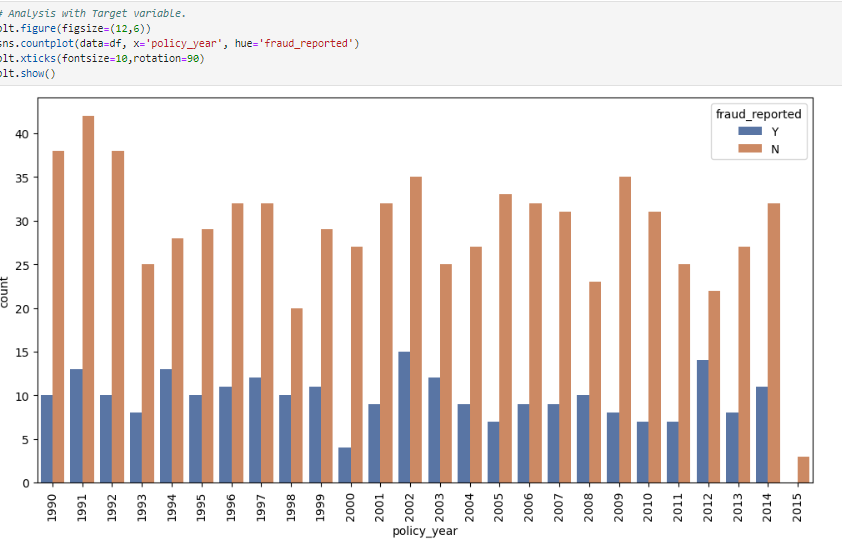
**A group of colorful bars

Description automatically generated with medium confidence**

**Observation :**

* Maximum fraud cases comes from people with age group of 31-50 year.
* Very few cases in 60+ year old peoples.
* Out of all cases around 24.7 % cases are Fraud.
* Almost same amout of cases come from each state.
* Maximum fraud cases come from state of Ohio.
* Number of claims come from female is higher than which reported by male insured.
* Almost same amount of fraud cases comes from same gender.

## Bivariate Analysis



A graph of different colored lines

Description automatically generated with medium confidence

Observation :

* Most of case comes from Multi-vehicle and single vehicle collision.
* Some claims are due to automobile robbery.
* **One claim out of three claim is fraud in multi or single vehicle collision incident.**

Key Insights on Department & Education level of from Pie Plot

**Feature Engineering: Data Pre-processing**

*Feature engineering is the process of transforming raw data into features that better represent the underlying problem to the predictive models, resulting in improved model accuracy on unseen data.*

Feature Engineering is very important step in building Machine Learning model. Some machine learning projects succeed and some fail. What makes the difference? Easily the most important factor is the features used. In Feature engineering can be done for various reason. **Some of them are mention below:**

1. **Feature Importance**: An estimate of the usefulness of a feature
2. **Feature Extraction**: The automatic construction of new features from raw data (Dimensionality reduction Technique like PCA)
3. **Feature Selection**: From many features to a few that are useful
4. **Feature Construction**: The manual construction of new features from raw data (For example, construction of new column for month out date - mm/dd/yy)

There are Varity of techniques use to achieve above mention means as per need of dataset. Some of Techniques important are as below:

* Handling missing values
* Handling imbalanced data using SMOTE
* Outliers’ detection and removal using Z-score, IQR
* Scaling of data using Standard Scalar or Minmax Scalar
* Binning whenever needed
* Encoding categorical data using one hot encoding, label / ordinal encoding
* Skewness correction using Boxcox or yeo-Johnson method
* Handling Multicollinearity among feature using variance inflation factor
* Feature selection Techniques:
* Correlation Matrix with Heatmap
* Univariate Selection – SelectKBest
* ExtraTreesClassifier method

In this case study we will use some of the mention feature engineering Techniques one by one.

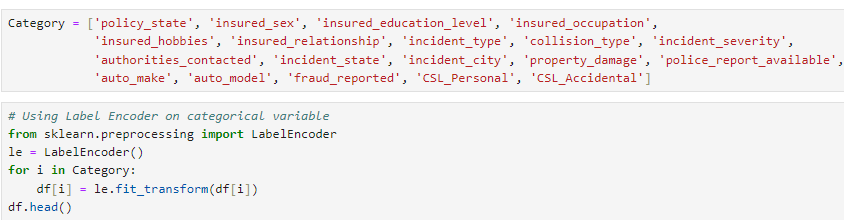
1. **Dropping unnecessary features**

Feature like ‘auto\_year’, ‘policy\_number’ are irrelevant from ML model building perspective. We will drop these features.



1. **Encoding Categorical & Ordinal Features**

Label Encoding is employed over categorical features.

****

Since now encoding is done we will move towards outliers’ detection and removal.

1. **Outliers’ detection and removal**

Identifying outliers and bad data in your dataset is probably one of the most difficult parts of data clean-up, and it takes time to get right. Even if you have a deep understanding of statistics and how outliers might affect your data, it’s always a topic to explore cautiously.

* Page 167, Data Wrangling with Python, 2016

Machine learning algorithms are sensitive to the range and distribution of attribute values. Data outliers can spoil and mislead the training process resulting in longer training times, less accurate models and ultimately poorer results. Outliers can be seen in boxplot of numerical feature. .

A screen shot of a computer code

Description automatically generated A computer screen shot of a computer code

Description automatically generated

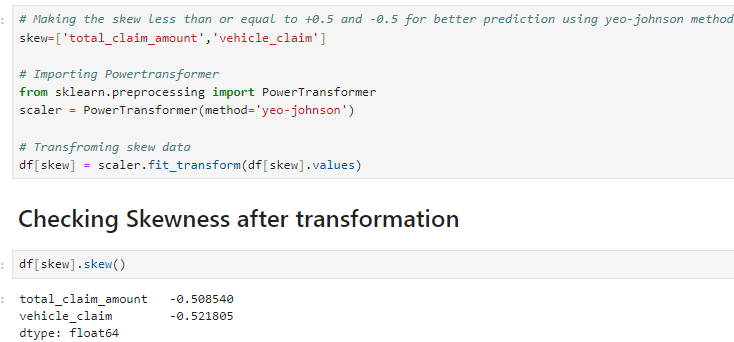
1. **Checking Skewness**

**A screenshot of a computer

Description automatically generated**

With the above plot, it's evident that the skewness in several columns exceeds the permissible limit of -0.5 to 0.5, indicating a need for removal.

**Approach:- Skewness removal through Power transformer**

****

1. **Correlation Heatmap**

Correlation Heatmap show in a glance which variables are correlated, to what degree, in which direction, and alerts us to potential multicollinearity problems. The bar plot of correlation coefficient of target variable with independent features shown below

**A graph with green squares

Description automatically generated with medium confidence**

1. **Multicollinearity between features**

A white background with black text

Description automatically generated Variance Inflation factor imported from statsmodels.stats.outliers\_influence to check multicollinearity between features.

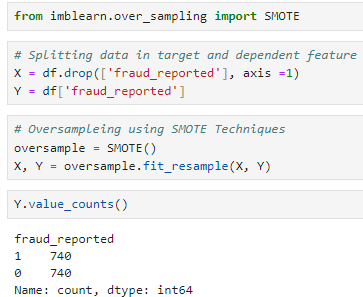
A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

1. **Handling imbalanced data using SMOTE**

This two-class dataset is imbalanced (76% vs 24%). As a result, there is a possibility that the model built might be biased towards to the majority and over-represented class. We can resolve this by Synthetic Minority Oversampling Technique (SMOTE) to over-sample the minority class.



1. **Scaling of data using Standard Scalar**

A screenshot of a computer code

Description automatically generated

1. **Dimensionality Reduction Using PCA**

PCA used find patterns and extract the latent features from our dataset.

A graph with a red dotted line

Description automatically generated

We can see that 28 principal components attribute for 90% of variation in the data. PCA applied for 28 components.

A screenshot of a computer code

Description automatically generated

**Machine Learning Model Building:**

In this section we will build Supervised learning ML model-based classification algorithm. As objective is to predict fraud\_reported in ‘Yes’ or ‘No’ leads to fall problem in domain of classification algorithm. train\_test\_split used to split data with size of 0.3

A screenshot of a computer code

Description automatically generated

First we will build base model using logistic regression algorithim. Best random state is investigated using for loop for random state in range of (1,250).

A screenshot of a computer program

Description automatically generated

Logistics regression model is train with random state 9. The evalution matrix along with classification report is as below :

A screenshot of a computer

Description automatically generated

As Now base model is ready with f1-score of 0.76, we will train model with different classification algorithm along with k-5 fold cross validation. The final evaluation matrix different classification algorithm is as shown table below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ML Algorithm | Accuracy Score | CVMean Score | f-1 Score | Recall | Precision |
| Logistics Regression | 0.76 | 0.756 | 0.76 | 0.75 | 0.78 |
| SVC | 0.8310 | 0.807 | 0.84 | 0.85 | 0.82 |
| GaussianNB | 0.7905 | 0.7722 | 0.80 | 0.80 | 0.79 |
| DecisionTreeClassifier | 0.7319 | 0.6905 | 0.72 | 0.68 | 0.77 |
| RandomForestClassifier | 0.8175 | 0.8087 | 0.82 | 0.84 | 0.81 |
| ExtraTreesClassifier | 0.8355 | 0.8216 | 0.84 | 0.84 | 0.84 |

(Min Value in column -Green, Max Value in column - Pink Colour )

We can see that Extra Trees Classifier gives us maximum f1-score & mean cross validation score. We will perform hyper parameter tuning on extra trees classifier to build final ML Model

A screenshot of a computer program

Description automatically generated

Next step is to build final machine learning model over best params in Hyper parameter tuning.

A screenshot of a computer program

Description automatically generated

We can see that Final model with hyper parameter tuning leads to slight decrease in accuracy score from 0.7612 in original model to 0.7522. This complete possible We will use model with default values as our final model. AOC-ROC score of final model is shown below:

A graph with a line

Description automatically generated

At last, we will save final model with joblib library, so it can be deploy on cloud platform.

A screen shot of a computer

Description automatically generated

**Predicting the Final Model**

**A screenshot of a computer

Description automatically generated**

**Concluding Remarks on EDA and ML Model**

* Individuals between the age of 60- 69 have second-cheapest car insurance rates, behind only those in their 50s., I can summarize that other than the age of 22, the older you are, the higher the auto insurance claim amount is..
* When comparing the longevity of customer membership to the total claims amount for auto insurance, I hypothesize that it would affect the amount. Customer loyalty is important, so benefits for being a customer for a long duration of time would have an effect to how much money you would get back.
* The majority of individuals are customers up to around 300 months. However, it is incorrect to assume that customer longevity has a significant impact on total claims amount.
* The eldest vehicles made in 1995 had the largest amount of auto insurance claims than the more recent year (2015). This could be for numerous reasons. Damages to older cars could be significant or even dangerous, so the insurance would need to reimburse their customers.
* Different feature engineering techniques like balancing data, outliers’ removal, label encoding, feature selection & PCA are perform on data.
* Extra trees Classifier model gives maximum Accuracy.

You can get code of this case study from my [GitHub Profile](https://github.com/paridhy/Internship/tree/main/Evaluation%20Phase%20Projects/Insurance%20Claim%20Fraud%20Detection)