Fraudulent Insurance Claim Detection – Report

Problem Statement

Global Insure, a leading insurance company, processes thousands of claims annually. However, a significant percentage of these claims turn out to be fraudulent, resulting in considerable financial losses. The company's current process for identifying fraudulent claims involves manual inspections, which is time-consuming and inefficient. Fraudulent claims are often detected too late in the process, after the company has already paid out significant amounts. Global Insure wants to improve its fraud detection process using data-driven insights to classify claims as fraudulent or legitimate early in the approval process. This would minimize financial losses and optimize the overall claims handling process.

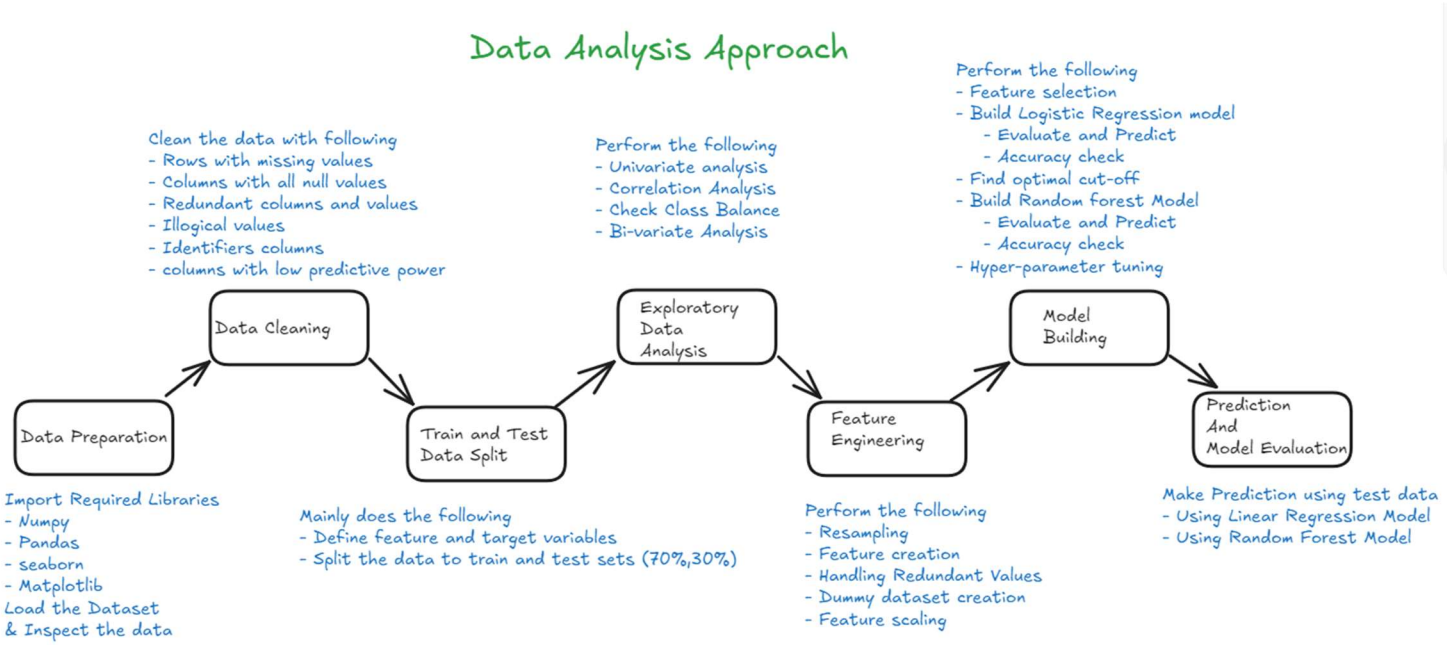
Business Objective

Global Insure wants to build a model to classify insurance claims as either fraudulent or legitimate based on historical claim details and customer profiles. By using features like claim amounts, customer profiles and claim types, the company aims to predict which claims are likely to be fraudulent before they are approved. Goals desired to be achieved are:

- Global Insure aims to enhance its ability to detect fraudulent insurance claims by leveraging historical claim data.
- The company seeks to identify patterns and key indicators that differentiate fraudulent claims from genuine ones.
- By developing a predictive model, it intends to assess the likelihood of fraud in incoming claims, enabling proactive fraud detection and reducing financial losses.

Solution Approach

For the analysis of fraudulent insurance claims, we followed a comprehensive data science methodology. The process began with data acquisition and initial setup, followed by thorough exploratory data analysis (EDA) and visualization to understand underlying patterns and relationships. The dataset was then systematically split into training and test sets, with careful attention to data preparation and feature scaling to ensure optimal model performance. Correlation analysis was performed to understand variable relationships, and feature selection was optimized using Recursive Feature Elimination with Cross-Validation (RFECV) to identify the most relevant predictors. The final stages involved building and training the predictive model, followed by rigorous evaluation and validation to assess its effectiveness in identifying fraudulent claims. This structured approach ensured a robust and reliable fraud detection system.



Step 1: Data Preparation

In this step we:

- Imported the required libraries like Numpy, Pandas, Seaborn and Matplotlibs.
- We read the dataset provided in CSV format and look at basic statistics of the data, including date types, preview of data, shape and general information of the columns.

Step 2: Data Cleaning

In this step we:

- Examined and handled the columns with null values *[_c39]*
- Examined the rows with missing values
- Examined and corrected columns with -ve values where it should be positive. *[umbrella_limit was corrected]*
- Checked for redundant and unique columns. *[None found]*
- Checked columns where large portion of values were unique and dropped. *[9 columns dropped]*
- Dropped for columns which are empty. *[One column _c39 was dropped.]*
- Fixed the data types for date columns which had a non-date type. *[incident_date, policy_bind_date]*

Step 3: Train and Test data split

We did split the dataset into 70% train and 30% validation and use stratification on the target variable.

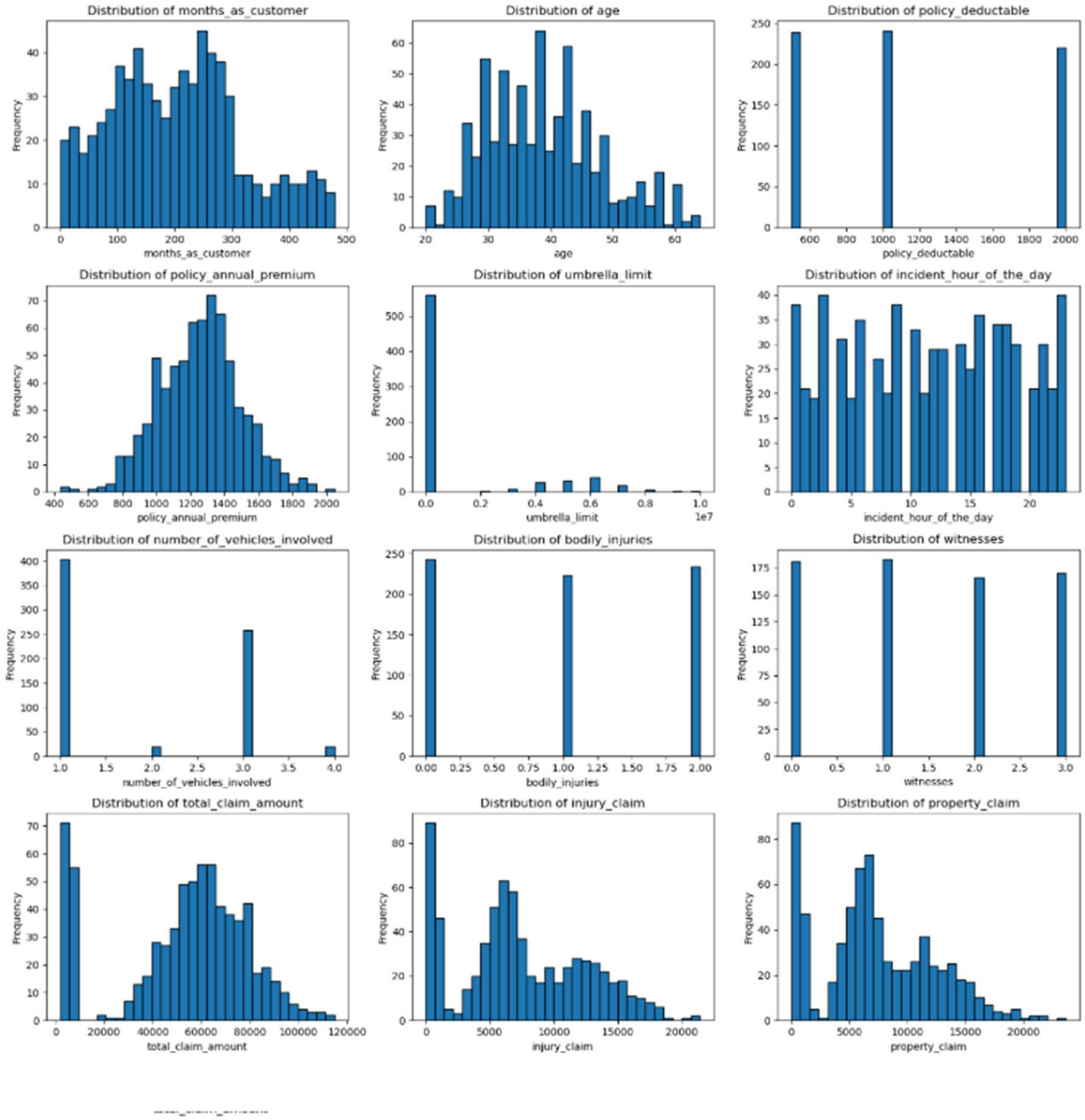
Step 4: Exploratory Data Analysis (EDA)

We conducted **univariate analysis** by creating a numerical dataset and plotting all those columns to visualize their distribution and understand their characteristics.

```
['months_as_customer',
 'age',
 'policy_deductable',
 'policy_annual_premium',
 'umbrella_limit',
 'incident_hour_of_the_day',
 'number_of_vehicles_involved',
 'bodily_injuries',
 'witnesses',
 'total_claim_amount',
 'injury_claim',
 'property_claim',
 'vehicle_claim']
```

As per our analysis:

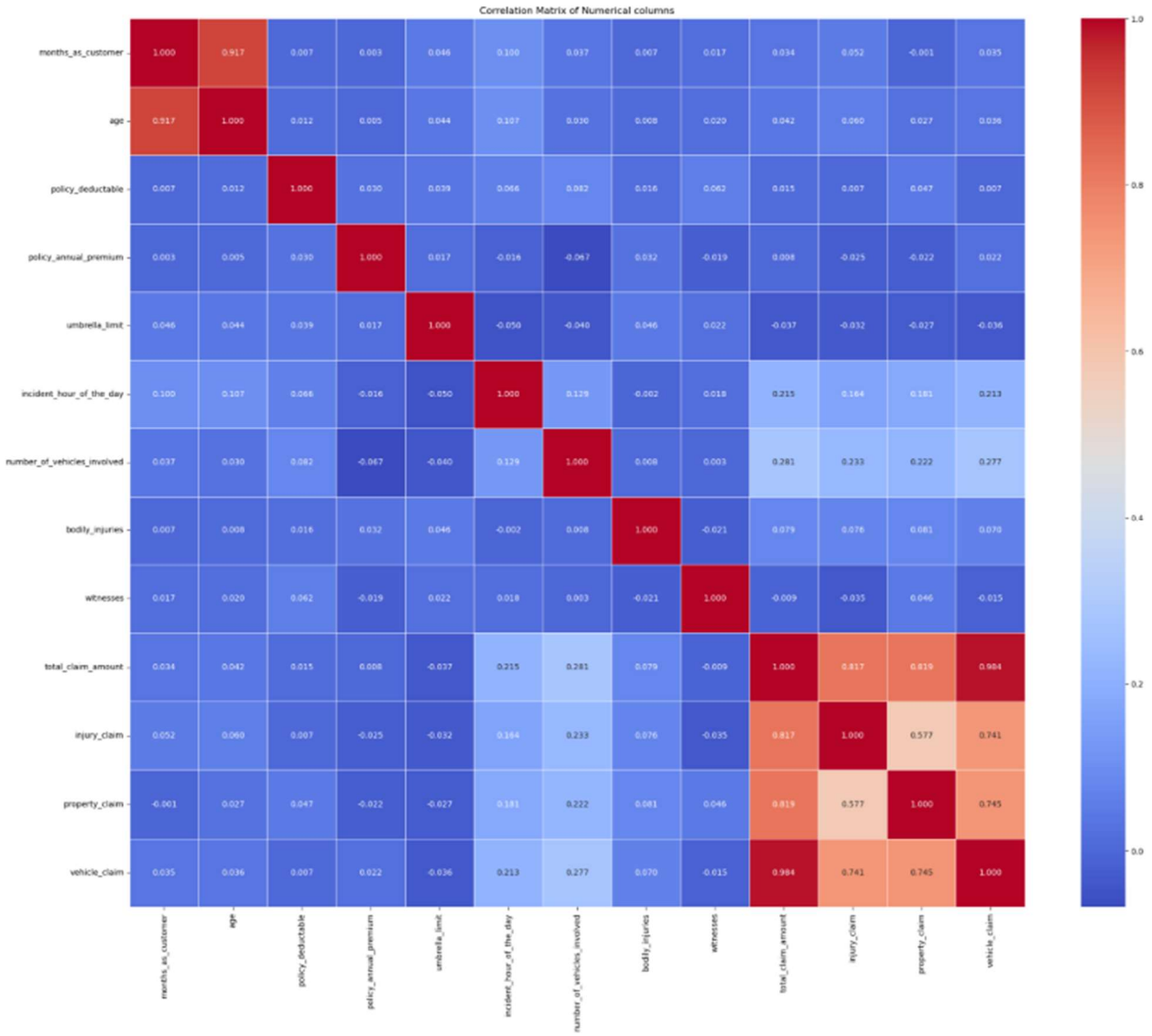
- There are customers in varied buckets of 'Months as a customer' with major numbers are within 100-300.
- Interesting patterns in vehicle_claim and policy_annual_premiums with major portion lying in the mid-range of 5 0K, 1200 -1400 respectively.
- There are specific higher buckets for property_claim, injury_claim and total_claim amount where major of the po pulation present.
- Distribution for age is between 20-70 years with major population between 30-50 years of age.
- Close to 80% of the claims are in umbrella limit 0.
- Witnesses are available for ~75% of the cases with number of witnesses ranging from 1-3.
- Bodily injuries are usually 1 or 2, and in many cases none.
- Number of vehicles involved are majorly 1 or 3.



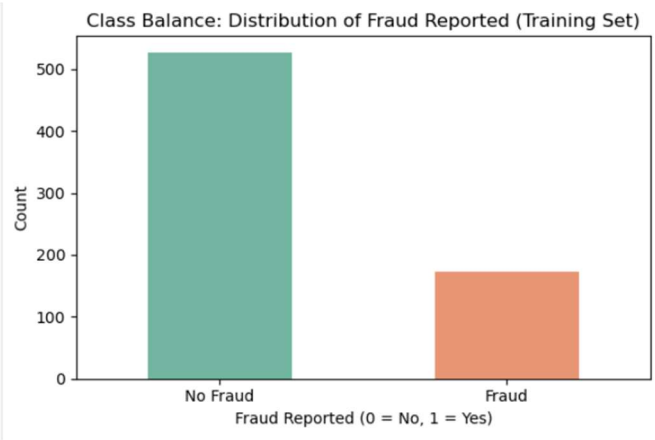
We conducted the **correlation matrix using heatmap** which helped to understand the relation between numeric variables to identify potential multicollinearity or dependencies.

The analysis shows that there is highest positive collinearity:

- total_claim_amount with injury_claim, property_claim and vehicle_claim.
- Age and months_as_customer.

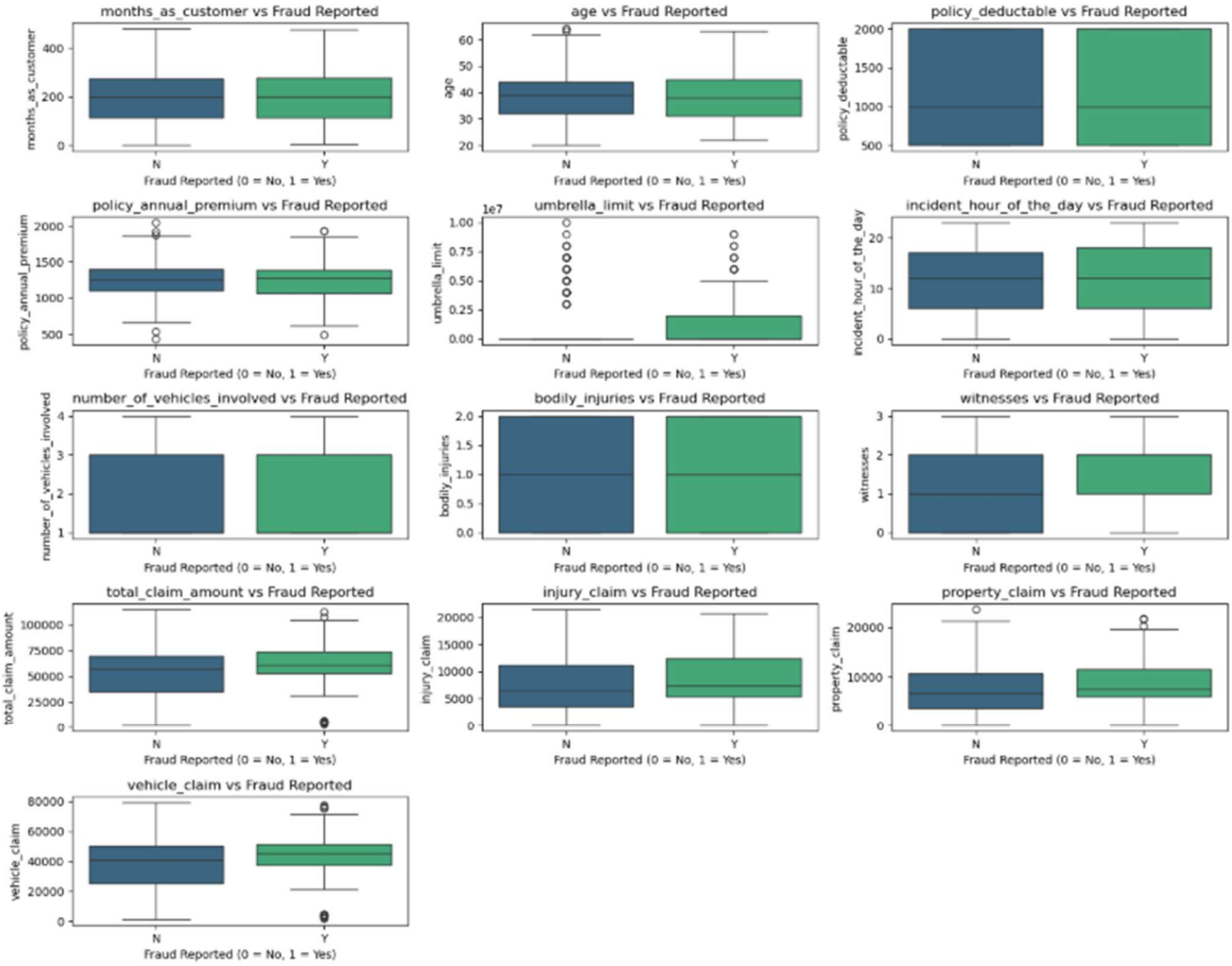


Used **countplot function (Histograms)** to examine the distribution of the target variable to identify potential class imbalances using visualization for better understanding.



Conducted **Bi-variate analysis** by plotting boxplots between the numerical variables and target variables in the dataset. This helped to understand the impact of the numeric variable on the target variable by using appropriate visualization techniques to identify trends and relationship strength.

The boxplot visualization reveals distinct patterns between fraudulent and non-fraudulent claims, where fraudulent claims demonstrate notably higher median amounts, greater variability (shown by larger box size and wider interquartile range), and more extreme outliers skewed towards higher values. In contrast, non-fraudulent claims exhibit more concentrated, consistent amounts, suggesting that claim amount characteristics serve as valuable indicators in fraud detection, with fraudulent claims typically showing higher amounts and more irregular patterns compared to legitimate claims.



Step 5: Feature Engineering

We performed the following in this step:

- Conducted resampling RandomOverSampler technique to handle class imbalance. Based on the data analysis we observed the distribution of fraud is ~35% whereas non-fraud is ~65%. Resampling helps to reduce the impact that could happen due to class imbalance in fraud_reported data.
- Created new features incident_day_of_week, incident_month, days_between_policy_and_incident from existing features in training and validation datasets.
- We dropped some of the redundant columns from both training and validation dataset.

- Combined categories that occur infrequently or exhibited similar behavior to reduce sparsity and improve model generalization.
- Created dummy variables for categorical columns 'insured_sex', 'insured_education_level', 'insured_occupation', 'insured_relationship', 'incident_type', 'collision_type', 'incident_severity', 'property_damage', 'police_report_available'. Also created dummy variables for target variables in training and validation datasets.
- Applied feature scaling to numerical variables using StandardScaler to prevent features with larger values from dominating the model.

Step 6: Model Building

We used Logistic Regression and Random Forest technique for building the learning models.

Using **Logistic Regression** model building technique, we:

- Applied RFECV to identify the most relevant features using Recursive Feature Elimination.
- Built the logistic regression model and analyzed statistical aspects such as p-values and VIFs to detect multicollinearity.
- We did fit the model on the training data and assessed the initial performance.
- Found the optimal cutoff to enhance sensitivity and improve model performance.
- Generated final predictions using the selected cutoff and evaluated model performance.

Using **Random Forest** model building technique, we:

- Built the initial model using RandomModelClassifier and got the importance scores to train the model.
- Trained the model with the selected features and generated the predictions.
- Checked the accuracy of the model and calculated the sensitivity, specificity, precision, recall and F1-score of the model.
- Performed hyperparameter tuning using GridSearchCV (parameters such as max_depth, min_samples_leaf, class_weight) to enhance the performance of the model.
- Built the final model using the best parameters.

Step 7: Prediction and Model Evaluation

- Built prediction by selecting relevant features from validation data using Logistics Regression model.
- Calculated the sensitivity, specificity, precision, recall and F1 score of the model. The calculated metrics are:

  Sensitivity (Recall): 0.6757
  Specificity: 0.8673
  Precision: 0.6250
  F1 Score: 0.6494

- Built predictions over the validation data using Random Forest model.
- Calculated the sensitivity, specificity, precision, recall and F1 score of the model. The calculated metrics are:

  Sensitivity (Recall): 0.6757
  Specificity: 0.8673
  Precision: 0.6250
  F1 Score: 0.6494

KEY INSIGHTS AND CONCLUSION

Derived the following insight from data analysis

- Model performance is relatively good with ~87% accuracy in identifying legitimate claims
- Top predictors:
    - Incident_severity and total_claim_amount highly predictive of the fraud.
    - Features like policy_annual_premium, days_between_policy_and_incident, and incident_hour_of_the_day also show strong importance in the Random Forest model.
- Class imbalance was found in the target variable (65% to 35% distribution of No fraud and Fraud respectively). Addressing the class imbalance issue with resampling significantly improved the model.
- Hyper-tuning in Random Forest helped to improve the sensitivity and F1 scores in the model.
- Implement strategic actions such as update of fraud detection policies and improving data collection methods.
- Threshold tuning can allow better control over false positives and false negatives.
- Model should be monitored regularly to understand the changes in pattern and accordingly update the model.
- Train staff on fraud indicators, share identified patterns and improve documentation practices.