**Problem Statement**

Global Insure, a leading insurance company, processes thousands of claims annually. A significant percentage of these claims are fraudulent, leading to substantial financial losses. The company's existing fraud detection process relies on manual inspections, which are both time-consuming and inefficient. Fraudulent claims often go unnoticed until payouts have already been made.

To address this challenge, Global Insure aims to enhance its fraud detection process using **data-driven insights**. By developing a predictive model, the company seeks to identify fraudulent claims early in the approval process, minimizing financial losses and improving operational efficiency.

**Business Objective**

The goal is to build a machine learning model to classify insurance claims as **fraudulent or legitimate** based on historical claim details and customer profiles. The model will analyze key features such as **claim amounts, customer profiles, claim types, and approval times** to predict fraudulent activity **before** claims are approved.

**Methodology**

**1. Data Preparation & Cleaning**

 Handled Null Values: Missing data was addressed using appropriate strategies, such as mean/median imputation for numerical features and mode or categorical encoding for categorical variables.

 Identified and Removed Redundant Columns & Values: Duplicate entries were detected and removed to ensure data integrity. Highly correlated or unnecessary features were dropped to prevent redundancy.

 Fixed Data Types: Ensured proper data type formatting, converting numerical values stored as text, adjusting date/time formats, and ensuring categorical features were appropriately encoded.

**2. Train-Validation Split**

**• Defined Feature and Target Variables:**

• Selected feature variables: **"months\_as\_customer", "age", "policy\_bind\_date", "policy\_state", "policy\_csl", "policy\_deductable", "policy\_annual\_premium", "umbrella\_limit", "insured\_zip", etc.**

• Defined the target variable as **"fraud\_reported"**, indicating whether a claim is fraudulent (1) or legitimate (0).

**• Split the Data:**

• Used train\_test\_split() from **Scikit-learn**, setting test\_size = 0.3 to allocate **70% for training** and **30% for testing**.

• Applied **stratified sampling** (if necessary) to maintain class balance, ensuring equal representation of fraudulent and legitimate claims in both training and test sets.

**Exploratory Data Analysis (EDA)**

**• Identified & Selected Numerical Columns for Univariate Analysis**

• Focused on key numerical features (**months\_as\_customer, age, policy\_deductible, policy\_annual\_premium, umbrella\_limit, etc.**).

**• Visualized Feature Distributions**

• Used **histograms** to understand the distribution of numerical features and detect outliers.

**• Performed Correlation Analysis**

• Generated a **heatmap** to examine relationships between numerical variables.

• Identified highly correlated predictors relevant for fraud detection.

**• Checked Class Balance**

• Used **bar charts** to analyze the distribution of fraudulent vs. legitimate claims.

• Determined whether fraud cases were underrepresented and considered techniques to balance the dataset.

**• Explored Feature Relationships with the Target Variable**

• Analyzed trends in fraud rates based on numerical feature distributions.

**Feature Engineering**

**1. Resampling on Training Data**

• Resampled the dataset to handle class imbalance and improve fraud detection accuracy.

**2. Feature Creation**

• Extracted new date-based features:

**• policy\_bind\_year, policy\_bind\_month, policy\_bind\_weekday** from policy bind date.

**• incident\_year, incident\_month, incident\_weekday, incident\_is\_weekend** from incident date.

**3. Handle Redundant Columns**

• Dropped unnecessary or highly correlated columns:  **months\_as\_customer, vehicle\_claim, injury\_claim, property\_claim, auto\_model, auto\_make, auto\_year, insured\_hobbies, incident\_state, incident\_city, policy\_bind\_date, incident\_date, insured\_zip**.

**4. Combined Values in Categorical Columns**

• Simplified and grouped categories for better model performance:

**• incident\_severity, insured\_education\_level, insured\_relationship, incident\_type, collision\_type, authorities\_contacted**.

**5. Dummy Variable Creation & Feature Scaling**

• Encoded categorical variables using dummy variables.

• Standardized numerical features for optimal model performance using **StandardScaler**.

**4. Model Building & Evaluation**

• Tested multiple algorithms (**Random Forest, Logistic Regression**), selecting the most effective.

• Used **Precision, F1-score** for evaluation.

• Implemented **hyperparameter tuning** to optimize model performance.