Problem Statement

Objective

This assignment aims to give you a practical understanding of model selection in real-world scenarios. You will learn how to analyse and preprocess data, extract meaningful features, and apply various machine learning models to uncover patterns and insights. Through model evaluation and optimisation techniques, you will gain the ability to compare different approaches and select the most effective one. This process will enhance your skills in building reliable predictive models, improving decision-making, and ensuring model accuracy and generalisation in diverse applications.

Business Value

Developing a predictive model for insurance fraud detection offers significant business value. Fraud investigations currently rely on manual processes like reviewing claims, calling claimants, and conducting background checks, which are time-consuming and inefficient. These delays allow fraud to go undetected while legitimate claims face unnecessary scrutiny. Predictive modelling enables early identification of high-risk claims, streamlining fraud investigations, reducing financial losses, and improving operational efficiency. It also enhances customer experience by expediting legitimate claims. Ultimately, an effective fraud detection model leads to better decision-making, optimised resource allocation, and increased profitability.

The objective is 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, claim types, and approval times, the company aims to predict which claims are likely to be fraudulent before they are approved.

Dataset Overview

Context

The data contains records of various claims received by an insurance company and relevant features associated with those claims such as policy number, incident date, incident type, and so on.

Content

The data is stored as a CSV file and contains information such as customer-related details (age, occupation, hobbies), policy information (coverage limits, premiums), incident specifics (type, severity, location), claim details (amounts, types of damage), vehicle information (make, model, year), and whether the claim was reported as fraud.

Acknowledgements

This dataset is free and is publicly available on UCI Machine Learning Repository.

Scoring and Penalty

- **Total Marks**: **250** (130 for code notebook, 70 for report and 50 for PPT)
- **Extension and Penalty**: As given in your learner handbooks

Instructions

1. This is a group assignment.
2. The programming language is Python.
3. You will be provided with the dataset and a starter notebook. You have to perform all the tasksin the starter notebook only.
4. It is very important that you do not change any headings, subheadings, questions or tasks in yournotebook as it can cause problems with grading.
5. The data might have inconsistencies and outliers; please handle them as you see fit and mention them in your report.
6. You are encouraged to search the web and consult AI tools for conceptual understanding. However,using plagiarised or AI-generated code is strictly prohibited and strongly discouraged.
7. Submitting plagiarised and purely AI-generated code or reports will result in significant penalties.

Submission Guidelines

1. You are required to upload your solution files in the submission field.
2. You are required to submit **three** files:
   (a) an ***Interactive Python Notebook (.ipynb)*** that contains your code
   (b) a ***Report Document (.pdf)*** that presents your visualisations, analysis, results, insights, and outcomes
   (c) a ***PPT (.pdf)***, which provides answers to the questions asked with supporting visualisations and texts.
3. Note that the solution notebook should only be generated from the starter files provided to you, and the other files should be based on them too.
4. The submitted Jupyter notebook, report, and PPT should contain group members' names and the assignment title in the format:
   `"Fraudulent_Claim_Detection_<name_of_member1_name_of_member2>.<filetype>"`
5. Mention all assumptions made in the report.
6. Outline the problem statement and overall approach in the report document with visualisations and reasoning. Any graphs/plots you generate for analysis should also be attached to the report.
7. The PPT should strictly contain the answers to questions asked in this assignment with visualisations.

Results Expected from Learners

Outline the problem statement and overall approach in the report document with visualisations and reasoning. Any graphs/plots you generate for analysis should also be attached to the report.

Based on this assignment, you have to answer the following questions in the PPT with supporting visualisations and texts:

1. How can we analyse historical claim data to detect patterns that indicate fraudulent claims?
2. Which features are most predictive of fraudulent behaviour?
3. Can we predict the likelihood of fraud for an incoming claim, based on past data?
4. What insights can be drawn from the model that can help in improving the fraud detection process?

In the starter notebook, you will find headings, subheadings, and checkpoints stating the tasks you need to perform. The marks associated with each checkpoint will also be mentioned in the notebook. Keep in mind not to edit the cells with marking schemes and questions. You can find a brief description of the tasks below.

1. Data Preparation: Import necessary libraries and load the data
2. Data Cleaning [10 marks]
   (a) Handling null values [2 marks]
   (b) Handling redundant features [5 marks]
   (c) Fix data types [3 marks]
3. Train-Validation Split [5 marks]
   (a) Define feature and target(s) variables [2 marks]
   (b) Split data into training and validation sets [3 marks]
4. EDA on Training Data [20 marks]
   (a) Perform univariate analysis [5 marks]
   (b) Perform correlation analysis [3 marks]
   (c) Check class balance [2 marks]
   (d) Perform bivariate analysis [10 marks]
5. EDA on Validation Data [optional]
   (a) Perform univariate analysis
   (b) Perform correlation analysis
   (c) Check class balance
   (d) Perform bivariate analysis
6. Feature Engineering [25 marks]
   (a) Perform resampling [3 marks]
   (b) New feature creation [4 marks]
   (c) Handle redundant columns [3 marks]
   (d) Combine values in categorical columns [6 marks]
   (e) Dummy variable creation [6 marks]
   (f) Feature rescaling [3 marks]
7. Model Building [50 marks]
   (a) Feature selection [4 marks]
   (b) Building logistic regression model [12 marks]
   (c) Find the optimal cutoff [12 marks]
   (d) Building random forest model [12 marks]
   (e) Hyperparameter tuning [10 marks]
8. Predictions and model evaluation [20 marks]
   (a) Make predictions over validation data using logistic regression model [10 marks]
   (b) Make predictions over validation data using random forest model [10 marks]

Evaluation Rubrics

The following rubrics will be used while evaluating your solutions to the above tasks.

Table 1: Rubrics

Criteria	Meets expectations	Does not meet expectations
Data Cleaning	1. Missing values are handled correctly. 2. Redundant values within categorical columns are correctly identified and handled (if any). 3. Redundant columns are dropped correctly. 4. DataTypes have been corrected for columns with incorrect DataTypes.	1. Missing values are handled inadequately. 2. Redundant values within categorical columns are incorrectly identified and handled (if any). 3. Redundant columns are not dropped correctly. 4. DataTypes have not been corrected or have been incorrectly assigned for columns with incorrect DataTypes.
Train-Validation Split	1. Feature and target variables are defined correctly. 2. Data is split into training and validation sets maintaining the ratio of 70:30.	1. Feature and target variables are not defined or defined incorrectly. 2. Data splitting is implemented incorrectly or not implemented at all.
EDA on training data	1. Performed univariate analysis by visualising the distribution of all numerical columns. 2. Performed correlation analysis by visualising a heatmap of the correlation matrix. 3. Visualised class distribution of the target variable. 4. Bivariate analysis has been performed by analysing target likelihood for each category level and visualising the relationship between numerical columns and the target variable	1. Failed to plot the distributions of numerical columns or created incomplete or inaccurate plots without meaningful insights. 2. Incorrectly visualised the heatmap or failed to interpret correlations. 3. Failed to visualise the class distribution of the target variable. 4. Target Likelihood Analysis has not been performed or done incorrectly. The influence of numerical variables on the target variable has not been visualised. Incomplete or unclear visualisations and insights have been provided.

Continued on next page

Table 1: Rubrics (Continued)

Criteria	Meets expectations	Does not meet expectations
Feature Engineering	1. Performed resampling to handle class imbalance. 2. Created new feature(s) from existing features to enhance the model's ability to capture patterns. 3. Handled redundant columns which may be redundant or contribute minimal information toward prediction. 4. Identified and combined low frequency values in categorical columns to reduce sparsity and improve model generalisation. 5. Created dummy variables for independent and dependent columns in both training and validation sets. 6. Applied feature scaling to numerical columns effectively by scaling the features.	1. Failed to perform resampling to handle class imbalance. 2. Failed to create new feature(s) from existing features to enhance the model's ability to capture patterns. 3. Failed to handle redundant columns which may be redundant or contribute minimal information toward prediction. 4. Failed to identify and combine low frequency values in categorical columns to reduce sparsity and improve model generalisation. 5. Failed to identify or create appropriate dummy variables for independent and dependent columns in both training and validation sets. 6. Failed to apply or incorrectly performed feature scaling, leading to inconsistent data ranges.

Continued on next page

Table 1: Rubrics (Continued)

Criteria	Meets expectations	Does not meet expectations
Model Building	1. Selected the most important features using Recursive Feature Elimination Cross Validation (RFECV). 2. Built a logistic regression model using the selected features, evaluated multicollinearity with p-values and VIFs, made predictions, and assessed model performance. 3. Found the optimal cutoff by plotting the ROC curve, visualising trade-offs between sensitivity and specificity, precision and recall, and evaluated the final prediction with optimal cutoff. 4. Built a random forest model, made predictions and asssessed model performance. 5. Tuned the random forest model using appropriate technique, optimised hyperparameters, made predictions, and assessed performance with relevant metrics.	1. Failed to select appropriate features using Recursive Feature Elimination Cross Validation (RFECV). 2. Failed to correctly build the logistic regression model using selected features and evaluated or interpreted the performance metrics incorrectly. 3. Failed to identify the optimal cutoff or omitted key evaluations like plotting curves and calculated necessary performance metrics. 4. Failed to correctly build the random forest model, did not apply appropriate evaluation metrics, or misinterpreted the results. 5. Failed to tune the random forest model effectively, did not optimise hyperparameters, or misinterpreted the performance evaluation.
Prediction and Model Evaluation	1. Predictions were made on validation data using the selected relevant features in the logistic regression model. 2. Predictions were made on validation data using the random forest model. 3. Evaluated the performance of both the logistic regression and random forest models using the given evaluation metrics.	1. Failed to select relevant features or incorrectly made predictions on validation data in the logistic regression model. 2. Made incorrect predictions on validation data using the random forest model. 3. Failed to evaluate the performance of both the logistic regression and random forest models using the correct evaluation metrics or interpreted the results inaccurately.

Continued on next page

Table 1: Rubrics (Continued)

Criteria	Meets expectations	Does not meet expectations
Report and Recommendations	1. The report has a clear structure, is not too long, and explains the most important results concisely in simple language. 2. The recommendations to solve the problem are realistic, actionable and coherent with the analysis. 3. The report includes visualisations and insights derived from them. 4. If any assumptions are made, they are stated clearly.	1. The report lacks structure, is too long or does not put emphasis on the important observations. The language used is complicated for business people to understand. 2. The recommendations to solve the problem are either unrealistic, non-actionable or incoherent with the analysis. 3. The report is missing visualisations or fails to provide meaningful insights. 4. Assumptions made, if any, are not stated clearly.
PPT	1. Effectively answered the questions asked in the assignment using appropriate visualisations and texts. 2. Used clear, well-structured slides with relevant charts, graphs, and summaries to enhance understanding.	1. Did not effectively answer the questions asked in the assignment or used inappropriate/unclear visualisations. 2. PPT lacked structure, clarity, or relevant visual elements, making it difficult to understand key insights.
Conciseness and readability of the code	1. The code is concise and syntactically correct. Wherever appropriate, built-in functions and standard libraries are used instead of writing long code (if-else statements, loops, etc.). 2. Custom functions are used to perform repetitive tasks. 3. The code is readable with appropriately named variables and detailed comments are written wherever necessary.	1. Long and complex code is used instead of shorter built-in functions. 2. Custom functions are not used to perform repetitive tasks resulting in the same piece of code being repeated multiple times. 3. Code readability is poor because of vaguely named variables or lack of comments wherever necessary.