IBB31103 | ISB46703 Introduction to/Principles of Artificial Intelligence
Universiti Kuala Lumpur
February 2023
Dr. Faiz

Group Mini Project

Weight: 15%
Due: 9 June 2023

1 Introduction

The objective of this group assignment is to apply Support Vector Machine (SVM) binary classification techniques on a car insurance claim dataset to predict whether a customer will file an insurance claim or not based on given features. This assignment will cover basic exploratory data analysis, data visualization, data preprocessing, model evaluation, and parameter tuning for the SVM model.

1.1 Goal

Your final goal is to predict whether the policy holder will file a claim in the next 6 months or not.

2 Dataset

The dataset used for this assignment is a car insurance claim dataset containing policyholders attributes like policy tenure, age of the car, age of the car owner, the population density of the city, make and model of the car, power, engine type, etc, and the target variable indicating whether the policyholder files a claim in the next 6 months or not.

Dataset download link: https://bit.ly/3BWSxua

Variable	Description
`policy_id`	Unique identifier of the policyholder
`policy_tenure`	Time period of the policy
`age_of_car`	Normalised age of the car in years
`age_of_policyholder`	Normalised age of policyholder in years
`area_cluster`	Area cluster of the policyholder
`population density`	Population density of the city (Policyholder City)
`make`	Encoded Manufacturer/company of the car
`segment`	Segment of the car (A/ B1/ B2/ C1/ C2)
`model`	Encoded name of the car

fuel_type	Type of fuel used by the car
max_torque	Maximum Torque generated by the car (Nm@rpm)
max_power	Maximum Power generated by the car (bhp@rpm)
engine_type	Type of engine used in the car
airbags	Number of airbags installed in the car
is_esc	Boolean flag indicating whether Electronic Stability Control (ESC) is present in the car or not.
is_adjustable_steering	Boolean flag indicating whether the steering wheel of the car is adjustable or not.
is_tpms	Boolean flag indicating whether Tyre Pressure Monitoring System (TPMS) is present in the car or not.
is_parking_sensors	Boolean flag indicating whether parking sensors are present in the car or not.
is_parking_camera	Boolean flag indicating whether the parking camera is present in the car or not.
rear_brakes_type	Type of brakes used in the rear of the car.
engine_displacement	Engine displacement of the car (cc).
cylinder	Number of cylinders present in the engine of the car.
transmission_type	Transmission type of the car.
gear_box	Number of gears in the car.
steering_type	Type of the power steering present in the car.
turning_radius	The space a vehicle needs to make a certain turn (Meters).
length	Length of the car (Millimetre).
width	Width of the car (Millimetre).
height	Height of the car (Millimetre).
gross_weight	The maximum allowable weight of the fully-loaded car, including passengers, cargo and equipment (Kg).
is_front_fog_lights	Boolean flag indicating whether front fog lights are available in the car or not.
is_rear_window_wiper	Boolean flag indicating whether the rear window wiper is available in the car or not.
is_rear_window_washer	Boolean flag indicating whether the rear window washer is available in the car or not.
is_rear_window_defogger	Boolean flag indicating whether rear window defogger is available in the car or not.
is_brake_assist	Boolean flag indicating whether the brake assistance feature is available in the car or not.
is_power_door_lock	Boolean flag indicating whether a power door lock is available in the car or not.

`is_central_locking`	Boolean flag indicating whether the central locking feature is available in the car or not.
`is_power_steering`	Boolean flag indicating whether power steering is available in the car or not.
`is_driver_seat_height_adjustable`	Boolean flag indicating whether the height of the driver seat is adjustable or not.
`is_day_night_rear_view_mirror`	Boolean flag indicating whether day and night rearview mirror is present in the car or not.
`is_ecw`	Boolean flag indicating whether Engine Check Warning (ECW) is available in the car or not.
`is_speed_alert`	Boolean flag indicating whether the speed alert system is available in the car or not.
`ncap_rating`	Safety rating given by NCAP (out of 5)
`is_claim`	Outcome: Boolean flag indicating whether the policyholder file a claim in the next 6 months or not.

Table 1: Car policyholder dataset.

The dataset consists of a combination of attributes with different data types. Some of the attributes, such as `fuel_type` and `segment`, are categorical, while others, like `max_torque` and `cylinder`, are numeric. The dataset contains a total of $97,655$ examples, and each example is described by 44 features.

3 Instructions

3.1 Exploratory Data Analysis [10 points]

1. Load the car policyholder claim dataset.
2. Perform basic exploratory data analysis to gain insights into the dataset.
3. Analyse the distribution of features, identify any missing values, and handle them appropriately.
4. Explore relationships between features and the claim status (`is_claim`).
5. Interpret the findings and discuss any patterns or trends observed.

3.2 Data Visualisation [10 points]

1. Create visualisations to explore relationships between features and the claim status.
2. Use appropriate plots and charts to illustrate the patterns and trends in the data.
3. Analyse the impact of different features on the likelihood of filing an insurance claim.

3.3 Data Preprocessing [15 points]

1. Perform necessary preprocessing steps such as handling missing (NaN) values, encoding categorical variables, and scaling numeric features by computing its z-score:

$$z = \frac{x - \mu}{\delta}$$

2. Split the dataset into training and testing sets (70 : 30 ratio) using `train_test_split()` function.

3.4 Modelling [10 points]

1. Import the necessary libraries for SVM binary classification.
2. Create an SVM classifier using an appropriate kernel function.
3. Fit the classifier to the training data.

3.5 Model Evaluation [20 points]

1. Predict the claim status using the trained SVM model on the testing data.
2. Evaluate the performance of the model using the following evaluation metrics:
   - Accuracy
   - Precision
   - Recall
   - F1-score.
3. Interpret the results and discuss the model's effectiveness in predicting car insurance claims.

3.6 Parameter Tuning [25 points]

1. Perform hyperparameter tuning on the SVM model to improve its performance.
2. Use grid search to find the optimal values for hyperparameters.
3. Re-train the SVM model with the tuned hyperparameters.
4. Evaluate the tuned model and compare its performance with the initial model.

3.7 Conclusion and Discussion [10 points]

1. Summarise the findings of the assignment, including the initial model's performance and the impact of hyperparameter tuning on the SVM model.
2. Discuss the limitations of the study and suggest potential improvements.
3. Reflect on the importance and relevance of SVM binary classification in predicting car policyholder claim status.

4 Groupwork Policy

Each group member should actively participate in all stages of the assignment, including exploratory data analysis, data visualization, data preprocessing, model creation, evaluation, parameter tuning, and conclusion. Collaboration, division of tasks, and effective communication within the group are essential for successful completion of the assignment.

5 Presentation

Each group will present their findings, including an overview of the dataset, exploratory data analysis, data visualisation, data pre-processing steps, model creation, evaluation results, parameter tuning process, and final conclusions. Visual aids, such as charts and graphs, are encouraged to enhance the clarity of the presentation.

6 Submission

Submit the following materials:

- Jupyter Notebook or code files used for data analysis and modeling via VLE.
- Presentation slides summarising the key findings and insights from the assignment.