

*Faculty of Engineering and Technology*

*Computer Science Department*

*COMP338 - Artificial Intelligence*

**Project 1-Genetic Algorithm**

**Prepared by:**

Nour Manasrah, 1211163

Raymaa Elian, 1200798

**Instructor:** Dr. Mohammad Helal

**Table of Contents**

[**1.Introduction** 4](#_Toc187088638)

[**2.Dataset Description** 4](#_Toc187088639)

[**3. Methodology** 4](#_Toc187088640)

[**3.1 Dataset Preprocessing** 4](#_Toc187088641)

[**3.2 Model Construction** 4](#_Toc187088642)

[**3.3 Evaluation Metrics** 5](#_Toc187088643)

[**4. Results** 5](#_Toc187088644)

[**4.1 Target Class Distribution** 5](#_Toc187088645)

[**4.2 Model Training and Testing** 5](#_Toc187088646)

[**4.3 Comparison of Accuracy and F1 Score** 6](#_Toc187088647)

[**4.4 Decision Trees for Models M1 and M2** 6](#_Toc187088648)

**Table of Figures**

[Figure 1:Decision Tree for M1 7](#_Toc187093891)

[Figure 2:Decision Tree for M2 7](#_Toc187093892)

# **1.Introduction**

This report outlines our efforts to create a predictive model with the Decision Tree algorithm. We specifically utilized the C4.5 decision tree algorithm from the Weka library to classify a dataset focused on car evaluations. The aim of this report is to detail the steps of data preprocessing, model building, evaluation, and the comparison of two models using various train-test splits. Additionally, we will showcase the decision trees produced by the models and examine the outcomes.

# **2.Dataset Description**

The data for this project comes from the Car Evaluation dataset. It contains different categories related to how cars are evaluated. The aim is to predict the target class, which shows if a car is rated as "unacceptable," "acceptable," "good," or "very good." Each entry in the dataset has various attributes like price, maintenance cost, and safety, which help in determining the car's evaluation.

# **3. Methodology**

## **3.1 Dataset Preprocessing**

Before we trained the models, we took several steps to prepare the data:

- Loading the dataset: We imported the dataset from a CSV file using Weka’s DataSource class.

- Shuffling the dataset: To promote randomness and minimize bias, we shuffled the dataset before dividing it into training and test sets.

- Setting the target class: We identified the target class (evaluation) and set the class index accordingly.

## **3.2 Model Construction**

We built two models (M1 and M2) using the Decision Tree (C4.5) algorithm.

- Model M1: We divided the data into 70% for training and 30% for testing. The training data was used to create the decision tree, and we evaluated the model on the test data.

- Model M2: We split the data into 50% for training and 50% for testing, then constructed and assessed a new decision tree.

## **3.3 Evaluation Metrics**

We assessed the models using two key metrics:

- Accuracy: The proportion of correctly classified instances in the test set .

Accuracy= (Correctly Classified Instances/ Total Number of Instances) ​×100

- F1 Score: The weighted average of precision and recall, which considers both false positives and false negatives.

F1=2× ((Precision × Recall) / (Precision +Recall))

# **4. Results**

## **4.1 Target Class Distribution**

Before starting model training, we looked at how the target class is distributed in the dataset. The target class indicates the car's evaluation as follows:

Unacceptable: 1 sample, 0.06%

Acceptable: 1210 samples, 69.98%.

Good: 384 samples, 22.23%.

Very Good: 65 samples, 3.76%.

This distribution helps us see any class imbalance and understand how the model might perform across different classes.

## **4.2 Model Training and Testing**

Model M1 (70% Training, 30% Testing)

Training Set Size: 1210 instances.

Test Set Size: 519 instances.

We used 70% of the data to train the decision tree and the remaining 30% for testing. The model's performance was assessed using the test set.

Model M2 (50% Training, 50% Testing)

Training Set Size: 865 instances.

Test Set Size: 864 instances.

We followed a similar approach with a 50-50 split for training and testing, creating a new decision tree.

## **4.3 Comparison of Accuracy and F1 Score**

After training the models, we measured their accuracy and F1 score on the test data.

Model M1:

Accuracy: 94.61%.

F1 Score: 0.946.

Model M2:

Accuracy: 88.66%.

F1 Score: 0.881.

Model M1 is better than Model M2 in accuracy (94.61% compared to 88.66%) and F1 score (0.9457 versus 0.8815). This advantage comes from M1 having a larger training set (70% versus 50%), which helps it recognize patterns and generalize well. While both models have similar decision tree designs, M1's larger dataset likely leads to its better results. On the other hand, M2, with a 50/50 data split, has less information to learn from, which could be why its accuracy and F1 score are lower.

## **4.4 Decision Trees for Models M1 and M2**

Decision Tree for Model M1:

The decision tree for M1 was created using Weka, based on the C4.5 algorithm. It splits the data according to the attribute that best classifies the target class. The tree structure is as follows:

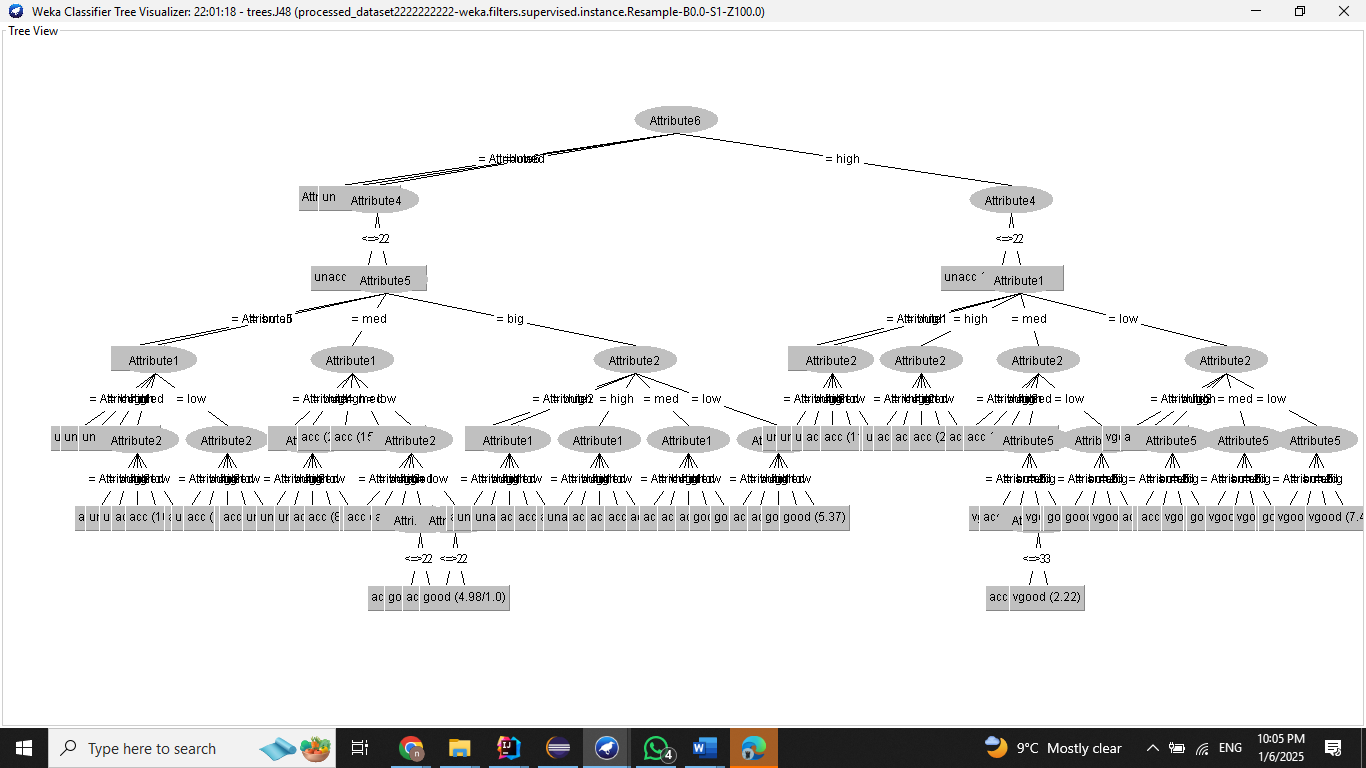


Figure 1:Decision Tree for M1

Decision Tree for Model M2:

The decision tree for M2 was generated in the same way.

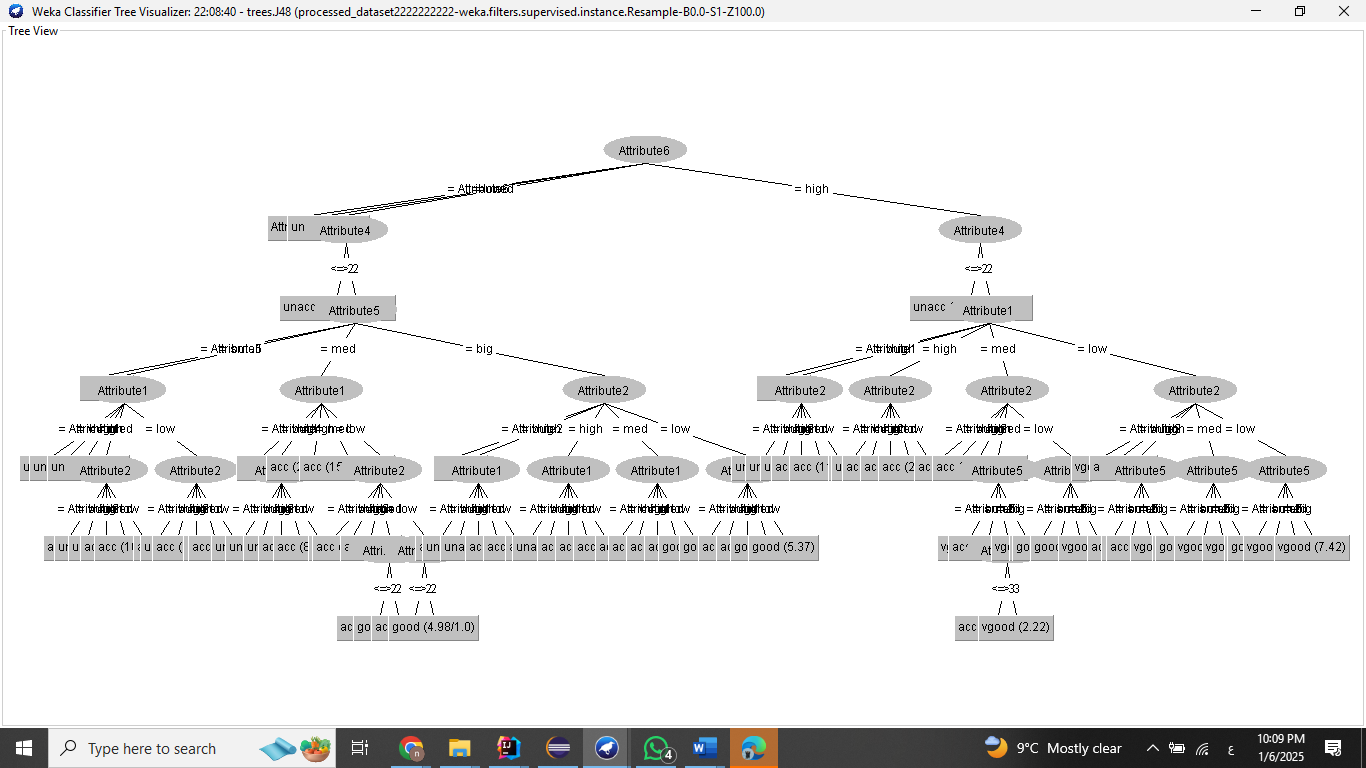


Figure 2:Decision Tree for M2