دانشگاه شیراز

Shiraz University

Machine Learning

Homework 3

Decision Tree

Supervisor:

Dr. Hashemi

Ali Mahmoodi

Atefe Rajabi

Spring 2024

Index

Problem Definition

Introduction:

The use of decision tree classifiers is tasked with predicting the survival outcomes of patients suffering from heart failure. This approach is part of a broader effort to utilize machine learning in enhancing the predictability of medical outcomes based on historical data, particularly under the added complexity of COVID-19.

Problem Context:

Heart failure is a significant global health issue, and the COVID-19 pandemic has compounded the risks associated with cardiovascular diseases. The dataset provided, "Heart Failure and Covid19.csv," includes information that could potentially illuminate correlations between various health indicators and patient survival rates during the pandemic.

Specific Task:

The task involves employing decision tree algorithms to model and predict the survival of patients. This involves understanding the intricacies of different decision tree methodologies (like ID3, C4.5, CART, CHAID, MARS) and implementing them from scratch to handle both binary and multiclass classification scenarios using One Versus One (OVO) and One Versus All (OVA) strategies.

Data Exploration:

The dataset "Heart Failure and Covid19.csv" from Kaggle is designed to help predict the survival of patients with heart failure in the context of the COVID-19 pandemic.

- Source: The dataset is part of a study published in BMC Medical Informatics and Decision Making[1], by Davide Chicco and Giuseppe Jurman.

- Context: It contains medical records of 299 patients with heart failure collected to study survival outcomes.

- Features: The dataset comprises 13 clinical features per patient.

- Target Variable: The primary outcome variable is the 'death event', indicating if the patient died during the follow-up period.

Feature Details

1. Age: Age of the patient (continuous variable ranging from 40 to 95 years).

2. Anaemia: Binary indicator (0 or 1) whether the patient has a decrease in red blood cells or hemoglobin.

3. Creatinine Phosphokinase (CPK): Enzyme level in the blood, measured in mcg/L (range: 23 to 7861 mcg/L).

4. Diabetes: Binary indicator (0 or 1) whether the patient has diabetes.

5. Ejection Fraction: Measures the percentage of blood leaving the heart at each contraction (range: 14% to 80%).

6. High Blood Pressure: Binary indicator (0 or 1) whether the patient has hypertension.

7. Platelets: Platelet count in the blood measured in kiloplatelets/mL (range: 25.01 to 850.00 kiloplatelets/mL).

8. Serum Creatinine: Level of creatinine in the blood measured in mg/dL (range: 0.50 to 9.40 mg/dL).

9. Serum Sodium: Sodium level in the blood measured in mEq/L (range: 114 to 148 mEq/L).

10. Sex: Binary indicator (0 for women, 1 for men).

11. Smoking: Binary indicator (0 or 1) whether the patient smokes.

12. Time: Follow-up period measured in days (range: 4 to 285 days).

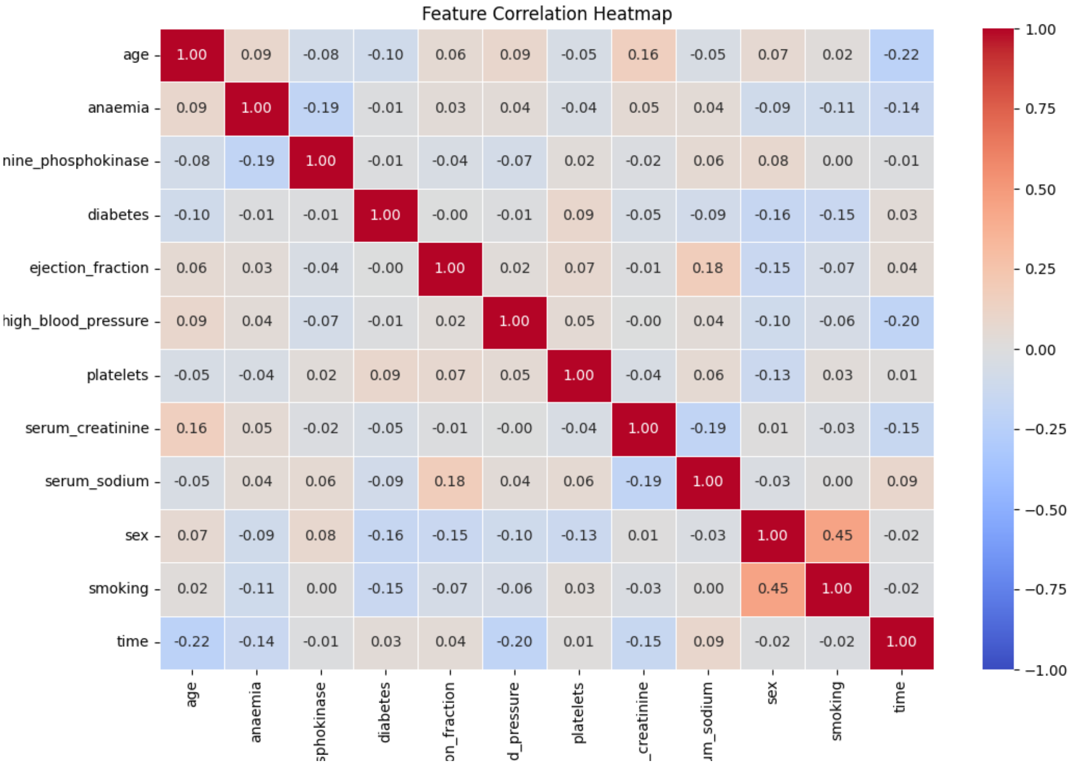
[1] https://doi.org/10.1186/s12911-020-1023-5

Statistical Details

- Instances: 299

- Features: 12 independent features and 1 dependent feature ('death event').

Feature Correlation Analysis:



The provided dataset contains various features related to heart disease patients, including their medical history and demographic information. To understand the relationships between these features, we examine their correlations.

1. Age and Heart Disease Outcomes: Age is positively correlated with the occurrence of heart disease events. Older patients tend to have a higher likelihood of experiencing adverse heart events.

2. Anaemia and Heart Disease Outcomes: Anaemia, indicated by lower red blood cell counts, is another factor positively correlated with heart disease. Patients with anaemia are more likely to suffer from heart conditions.

3. Creatinine Phosphokinase Levels: This enzyme's levels indicate muscle damage, including heart muscle. High levels are typically associated with increased risk of heart disease, reflecting the body's response to muscle injury.

4. Diabetes: Patients with diabetes have a higher risk of heart disease. This correlation highlights the interplay between metabolic disorders and cardiovascular health.

5. Ejection Fraction: This measure of heart function is inversely correlated with heart disease outcomes. Lower ejection fractions suggest poorer heart performance and are associated with higher risk of heart events.

6. High Blood Pressure: Hypertension is a significant risk factor for heart disease. Patients with high blood pressure are more prone to heart disease, underscoring the importance of blood pressure management.

7. Platelets Count: Platelets are involved in clotting, and their levels can impact heart disease risk. Abnormal platelet counts may signal underlying health issues related to heart disease.

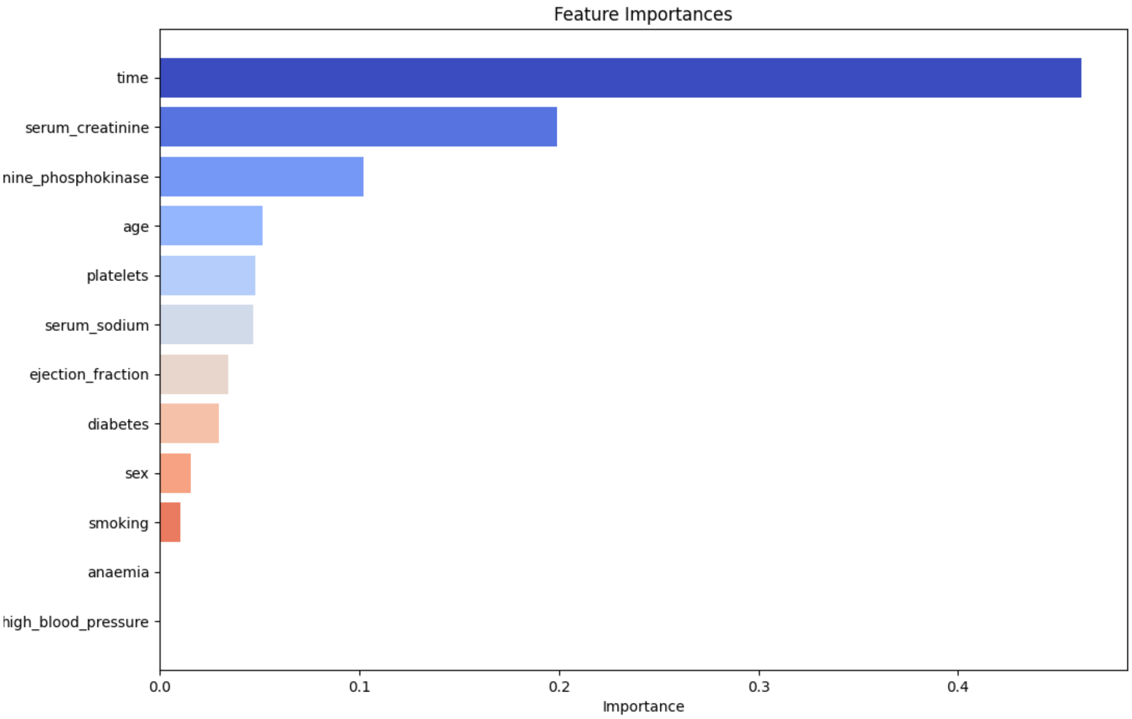
8. Serum Creatinine and Sodium Levels: These kidney function markers correlate with heart health. Elevated serum creatinine levels and low serum sodium levels are indicative of worse heart outcomes, as kidney and heart health are closely linked.

9. Sex: Gender differences exist in heart disease risk, with males typically at higher risk. This correlation helps tailor gender-specific prevention and treatment strategies.

10. Smoking: Smoking is a well-known risk factor for heart disease. Smokers are significantly more likely to experience heart disease, reinforcing the critical need for smoking cessation programs.

11. Time and DEATH_EVENT: The duration patients have been under observation is an essential factor. Longer observation periods provide more data on the progression and outcomes of heart disease.

Feature Importance Analysis:



Understanding the importance of different features in predicting heart disease outcomes is crucial for developing effective diagnostic and treatment strategies. Feature importance helps identify which variables significantly contribute to the risk of heart disease, allowing clinicians and researchers to focus on the most impactful factors.

1. Age: Age is a critical factor, as the risk of heart disease increases with advancing age. Older individuals are more likely to experience heart-related issues due to the natural aging process and the accumulation of risk factors over time.

2. Ejection Fraction: Ejection fraction, a measure of the heart's pumping efficiency, is a highly significant predictor. Lower ejection fractions indicate reduced heart function and are strongly associated with adverse heart outcomes.

3. Serum Creatinine: This marker of kidney function is crucial, as elevated serum creatinine levels suggest impaired kidney function, which is closely linked to heart health. Poor kidney function can exacerbate heart disease risk.

4. High Blood Pressure: Hypertension is a well-known risk factor for heart disease. High blood pressure can lead to various cardiovascular complications, making it a vital feature for predicting heart disease risk.

5. Serum Sodium: Sodium levels in the blood are important for maintaining fluid balance and proper heart function. Abnormal serum sodium levels can indicate underlying health issues that increase heart disease risk.

6. Diabetes: The presence of diabetes is a significant predictor of heart disease. Diabetes can cause damage to blood vessels and the heart, leading to an increased risk of cardiovascular events.

7. Anaemia: Anaemia, characterized by low red blood cell counts, is another important feature. Anaemia can strain the heart as it works harder to supply oxygen to the body, increasing the risk of heart disease.

8. Creatinine Phosphokinase: Elevated levels of this enzyme indicate muscle damage, including potential heart muscle damage. High levels can signal acute heart issues, such as myocardial infarction.

9. Platelets: Platelet count is essential in predicting heart disease risk. Abnormal platelet levels can indicate clotting disorders or other health issues that impact cardiovascular health.

10. Smoking: Smoking is a significant modifiable risk factor. It contributes to the development and progression of heart disease, making it an important feature for predicting outcomes.

11. Sex: Gender differences in heart disease risk are well-documented, with males generally at higher risk. Understanding these differences helps tailor prevention and treatment strategies.

12. Time: The duration of patient observation provides valuable longitudinal data. Longer observation periods allow for better tracking of disease progression and outcomes.

Data Balance Analysis

To analyze the balance of the dataset, we look at the distribution of the target variable 'DEATH_EVENT':

- Distribution of 'DEATH_EVENT':
  - DEATH_EVENT = 0: The patient survived during the follow-up period.
  - DEATH_EVENT = 1: The patient died during the follow-up period.

Class 0: 0.678928 (67.9%)

Class 1: 0.321072 (32.1%)

This distribution shows that the dataset is somewhat imbalanced, with a higher proportion of patients surviving (67.9%) compared to those who died (32.1%). This imbalance should be considered when developing predictive models, as it can influence model performance and bias.

COVID19.csv

The dataset contains 50,729 entries and 22 columns. Below is a brief description of each feature in the dataset:

1. sex: Categorical variable indicating the sex of the individual (0 for female, 1 for male).

2. age: Numerical variable representing the age of the individual.

3. country: Categorical variable representing the country code.

4. province: Categorical variable representing the province code.

5. city: Categorical variable representing the city code.

6. infection_case: Categorical variable representing the case type of infection.

7. infection_order: Numerical variable indicating the order of infection (e.g., primary, secondary).

8. elementary_school_count: Numerical variable representing the count of elementary schools in the area.

9. kindergarten_count: Numerical variable representing the count of kindergartens in the area.

10. university_count: Numerical variable representing the count of universities in the area.

11. academy_ratio: Numerical variable representing the ratio of academies in the area.

12. elderly_population_ratio: Numerical variable representing the ratio of the elderly population in the area.

13. elderly_alone_ratio: Numerical variable representing the ratio of elderly individuals living alone.

14. nursing_home_count: Numerical variable representing the count of nursing homes in the area.

15. avg_temp: Numerical variable representing the average temperature.

16. min_temp: Numerical variable representing the minimum temperature.

17. max_temp: Numerical variable representing the maximum temperature.

18. precipitation: Numerical variable representing the amount of precipitation.

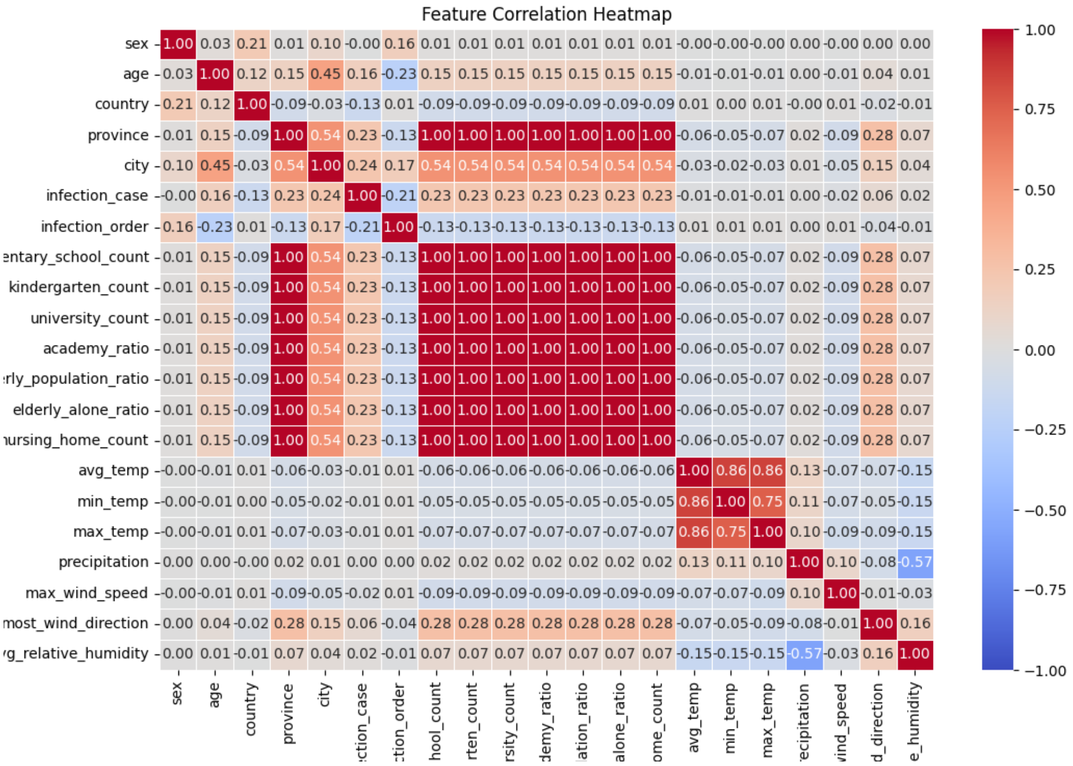
19. max_wind_speed: Numerical variable representing the maximum wind speed.

20. most_wind_direction: Categorical variable representing the most frequent wind direction.

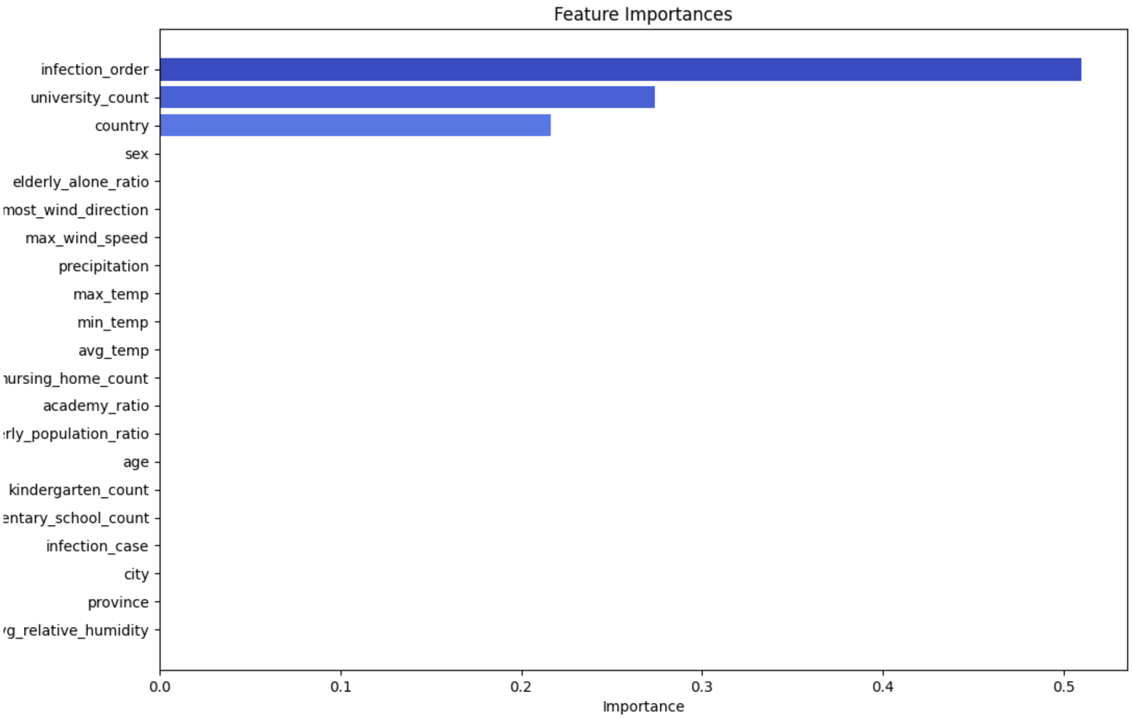
21. avg_relative_humidity: Numerical variable representing the average relative humidity.

22. label: Categorical variable indicating the label/class of the data (e.g., infected or not).

Data Correlation:



Feature Importance:



Data Balance Analysis

The dataset's balance can be evaluated by examining the distribution of the target variable, ‘label’, which indicates different classes:

- Class 0: 66.7% of the data

- Class 1: 25.4% of the data

- Class 2: 7.9% of the data

This distribution shows that the dataset is imbalanced. The majority class (Class 0) constitutes more than half of the data, while Class 2 is significantly underrepresented compared to the other classes. In the context of machine learning and data analysis, such imbalances can affect the performance of classification models, as they might become biased towards the majority class.

Decision Tree:

C4.5 and CART (Classification and Regression Trees) are two well-known algorithms used for creating decision trees, which are popular tools in machine learning and data mining for classification and regression tasks.

C4.5

C4.5 is an algorithm developed by Ross Quinlan, which is an extension of his earlier ID3 algorithm. It is used for generating a decision tree from a set of labeled training data.

1. Splitting Criteria: C4.5 uses information gain ratio as its splitting criterion. This is an improvement over ID3, which uses information gain. The information gain ratio aims to reduce the bias of information gain towards attributes with many distinct values.

2. Handling Continuous Data: Unlike ID3, which handles only categorical data, C4.5 can handle continuous data by creating a threshold and splitting the data into two partitions based on whether the data values are above or below this threshold.

3. Handling Missing Values: C4.5 can handle missing values in the dataset by using fractional instances, where the instance with missing value is split into different branches with weights proportional to the frequencies of those branches.

4. Pruning: C4.5 includes a pruning mechanism to remove branches that may reflect noise or outliers, helping to reduce overfitting. This is achieved through a process called error-based pruning.

5. Output: The output of C4.5 is a decision tree that can be used for classification purposes.

CART (Classification and Regression Trees)

CART is another popular decision tree algorithm introduced by Breiman et al. It is used for both classification and regression tasks, which distinguishes it from C4.5 that is primarily for classification.

1. Splitting Criteria: For classification tasks, CART uses the Gini impurity index as its splitting criterion. For regression tasks, it uses mean squared error (MSE) as the criterion to minimize.

2. Binary Splits: Unlike C4.5, which can create multi-way splits, CART only produces binary splits (i.e., each node has two children).

3. Handling Continuous and Categorical Data: CART can handle both continuous and categorical data. For continuous data, it searches for the best split by evaluating all possible thresholds.

4. Pruning: Similar to C4.5, CART also has a pruning mechanism to avoid overfitting. It uses a technique known as cost complexity pruning, which prunes the tree by balancing the complexity of the tree (i.e., number of leaves) against its accuracy on the training data.

5. Regression Trees: In the case of regression tasks, CART produces regression trees where each leaf represents a continuous value (typically the mean of the target values in that leaf).

6. Output: The output of CART can be either a classification tree (for classification tasks) or a regression tree (for regression tasks).

Pruning in Decision Trees

Pruning is a technique used to reduce the size of decision trees by removing parts of the tree that do not provide significant power in classifying instances. The goal of pruning is to reduce the complexity of the model and prevent overfitting, which occurs when the model captures noise in the training data instead of the actual patterns.

Pruning can be done in two main ways:

1. Pre-pruning (Early Stopping): This involves stopping the growth of the tree before it reaches the maximum depth. This can be controlled using parameters such as 'max_depth', 'min_samples_split', and 'min_samples_leaf'.

2. Post-pruning: This involves growing the tree to its maximum depth and then removing branches that have little importance. This can be done by setting a threshold on the complexity of the tree, often using cross-validation to determine the optimal threshold. This approach is implemented in the provided project.

Implementation Details:

Functions:

1. ‘__init__’

- ‘min_samples’: Minimum number of samples required to split a node.
- ‘method’: The algorithm to use for tree building ('c4.5' or 'cart').
- ‘max_depth’: The maximum depth of the tree. A negative value indicates no limit.

2. ‘_entropy’:

Calculates the entropy of a set of labels, which measures the disorder or impurity in the labels.

$$H(S) = -\sum_{i=1}^{n} p_i \log_2 p_i$$

3. ‘_split_info’: Computes the split information for a given feature, used in the calculation of the information gain ratio.

Information Gain:

$$IG(S, A) = H(S) - \sum_{t \in T} \frac{|S_t|}{|S|} H(S_t)$$

Split information:

$$SplitInfo(S, A) = -\sum_{t \in T} \frac{|S_t|}{|S|} \log_2 \left(\frac{|S_t|}{|S|} \right)$$

4. ‘_gain’: Calculates the information gain for a feature, optionally normalized by split info for continuous features.

$$GainRatio(S, A) = \frac{IG(S, A)}{SplitInfo(S, A)}$$

5. ‘_gini’: Calculates the Gini impurity of a set of labels.

$$Gini(S) = 1 - \sum_{i=1}^{n} p_i^2$$

6. ‘_impurity’: Computes the total impurity for potential splits of a feature, based on Gini impurity.

7. ‘_feature2expand_c4’ and ‘_feature2expand_cart’: Determines the best feature to split on next using either the C4.5 or CART method.

8. ‘rec_fit_c4’ and ‘rec_fit_cart’: Recursively builds the decision tree using the C4.5 or CART method.

10. ‘rec_predict_c4’ and ‘rec_predict_cart’: Recursively predicts class labels or probabilities for a given dataset using the built tree.

12. ‘score’: Calculates the accuracy of the classifier.

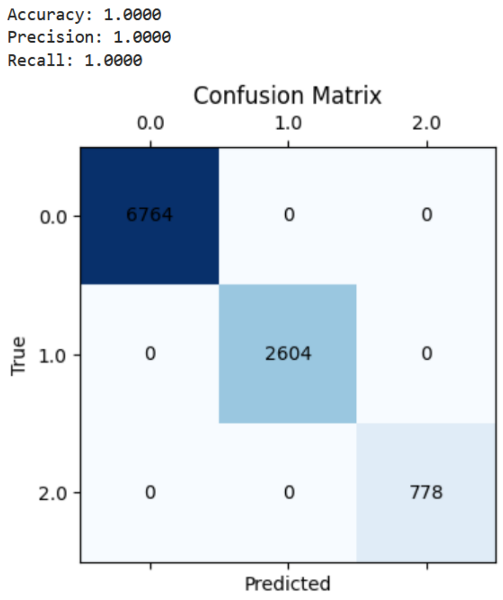
13. ‘rec_rules_c4’ and ‘rec_rules_cart’: Recursively extracts decision rules from the tree.

Result:

Dataset COVID:

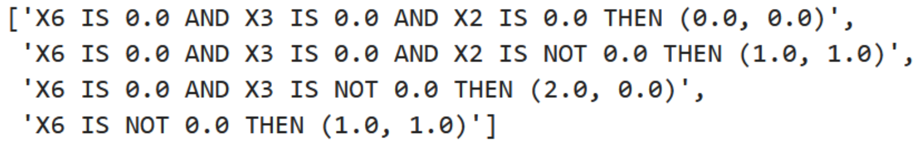
Multi-label tree:

CART:



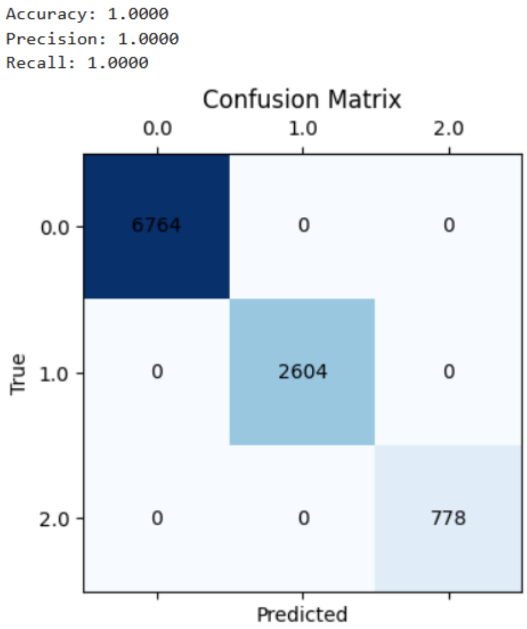
```
Accuracy: 1.0000
Precision: 1.0000
Recall: 1.0000
```

Extractred Rules:



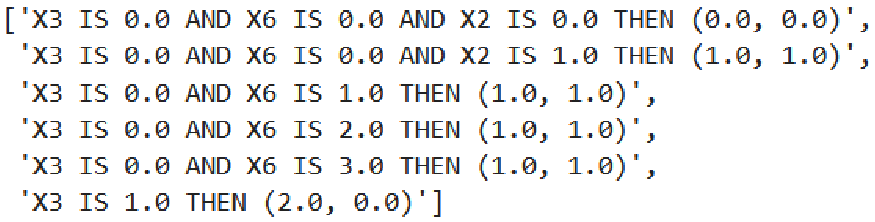
```
['X6 IS 0.0 AND X3 IS 0.0 AND X2 IS 0.0 THEN (0.0, 0.0)',
 'X6 IS 0.0 AND X3 IS 0.0 AND X2 IS NOT 0.0 THEN (1.0, 1.0)',
 'X6 IS 0.0 AND X3 IS NOT 0.0 THEN (2.0, 0.0)',
 'X6 IS NOT 0.0 THEN (1.0, 1.0)']
```

C4.5:



```
Accuracy: 1.0000
Precision: 1.0000
Recall: 1.0000
```

Extracted Rules:

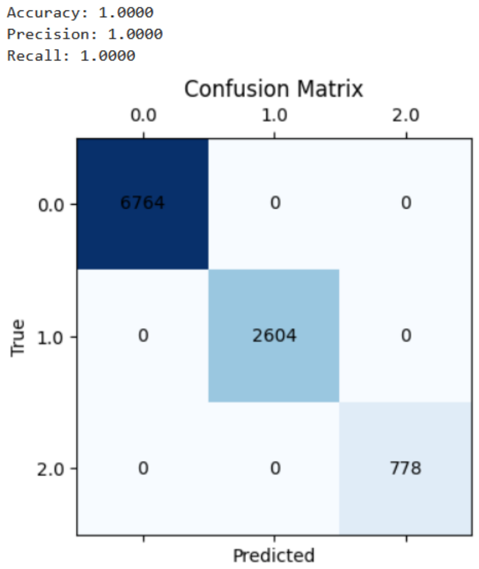


```
['X3 IS 0.0 AND X6 IS 0.0 AND X2 IS 0.0 THEN (0.0, 0.0)',
 'X3 IS 0.0 AND X6 IS 0.0 AND X2 IS 1.0 THEN (1.0, 1.0)',
 'X3 IS 0.0 AND X6 IS 1.0 THEN (1.0, 1.0)',
 'X3 IS 0.0 AND X6 IS 2.0 THEN (1.0, 1.0)',
 'X3 IS 0.0 AND X6 IS 3.0 THEN (1.0, 1.0)',
 'X3 IS 1.0 THEN (2.0, 0.0)']
```

OVA, OVO:

Using Decision tree as a base model:

OVA:

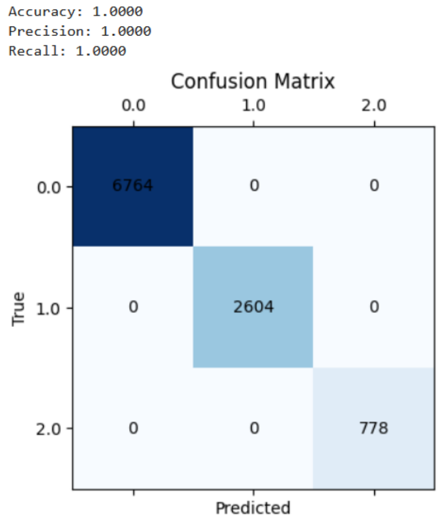


```
Accuracy: 1.0000
Precision: 1.0000
Recall: 1.0000
```

Confusion Matrix

True \ Predicted	0.0	1.0	2.0
0.0	6764	0	0
1.0	0	2604	0
2.0	0	0	778

OVO:



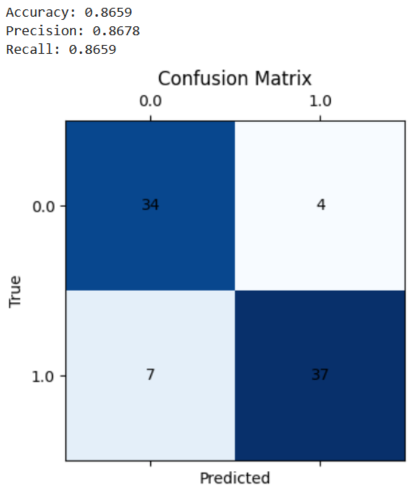
```
Accuracy: 1.0000
Precision: 1.0000
Recall: 1.0000
```

Confusion Matrix

True \ Predicted	0.0	1.0	2.0
0.0	6764	0	0
1.0	0	2604	0
2.0	0	0	778

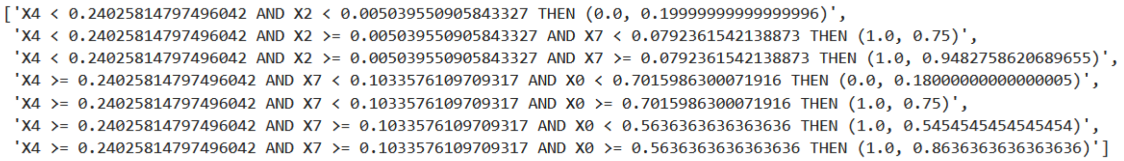
Dataset Heart.csv:

CART



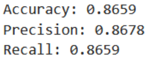
```
Accuracy: 0.8659
Precision: 0.8678
Recall: 0.8659
```

Extracted Rules:

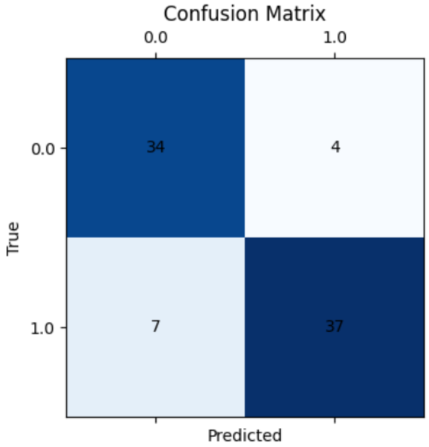


```
['X4 < 0.24025814797496042 AND X2 < 0.005039550905843327 THEN (0.0, 0.19999999999999996)',
 'X4 < 0.24025814797496042 AND X2 >= 0.005039550905843327 AND X7 < 0.0792361542138873 THEN (1.0, 0.75)',
 'X4 < 0.24025814797496042 AND X2 >= 0.005039550905843327 AND X7 >= 0.0792361542138873 THEN (1.0, 0.9482758620689655)',
 'X4 >= 0.24025814797496042 AND X7 < 0.1033576109709317 AND X0 < 0.7015986300071916 THEN (0.0, 0.18000000000000005)',
 'X4 >= 0.24025814797496042 AND X7 < 0.1033576109709317 AND X0 >= 0.7015986300071916 THEN (1.0, 0.75)',
 'X4 >= 0.24025814797496042 AND X7 >= 0.1033576109709317 AND X0 < 0.5636363636363636 THEN (1.0, 0.5454545454545454)',
 'X4 >= 0.24025814797496042 AND X7 >= 0.1033576109709317 AND X0 >= 0.5636363636363636 THEN (1.0, 0.8636363636363636)']
```

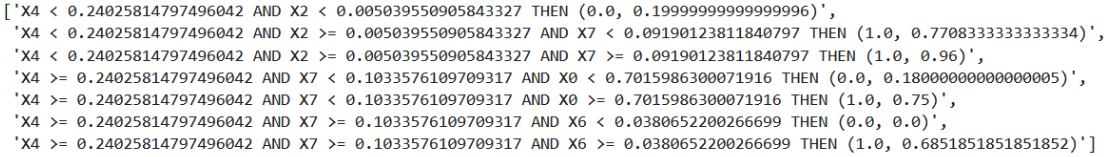
C4.5:



```
Accuracy: 0.8659
Precision: 0.8678
Recall: 0.8659
```



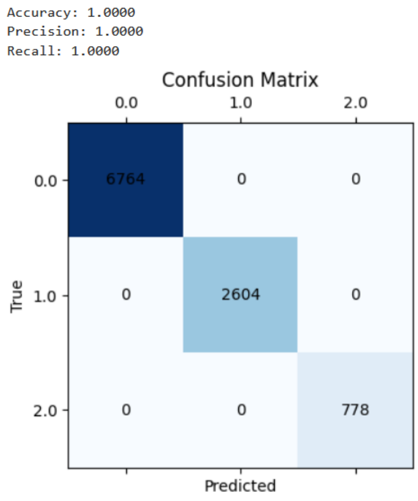
Extracted Rules:



```
['X4 < 0.24025814797496042 AND X2 < 0.005039550905843327 THEN (0.0, 0.19999999999999996)',
 'X4 < 0.24025814797496042 AND X2 >= 0.005039550905843327 AND X7 < 0.09190123811840797 THEN (1.0, 0.7708333333333334)',
 'X4 < 0.24025814797496042 AND X2 >= 0.005039550905843327 AND X7 >= 0.09190123811840797 THEN (1.0, 0.96)',
 'X4 >= 0.24025814797496042 AND X7 < 0.1033576109709317 AND X0 < 0.7015986300071916 THEN (0.0, 0.18000000000000005)',
 'X4 >= 0.24025814797496042 AND X7 < 0.1033576109709317 AND X0 >= 0.7015986300071916 THEN (1.0, 0.75)',
 'X4 >= 0.24025814797496042 AND X7 >= 0.1033576109709317 AND X6 < 0.0380652200266699 THEN (0.0, 0.0)',
 'X4 >= 0.24025814797496042 AND X7 >= 0.1033576109709317 AND X6 >= 0.0380652200266699 THEN (1.0, 0.6851851851851852)']
```

Comparison with the baseline:

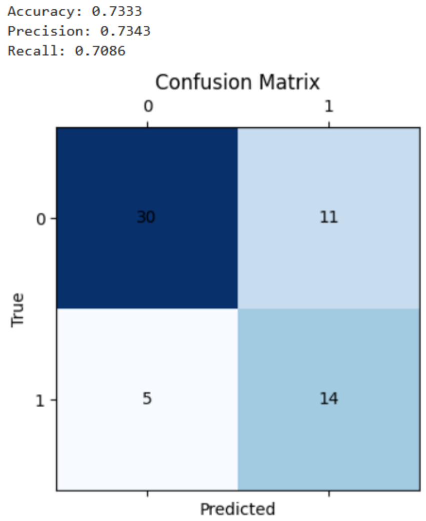
Decision Tree Classifier (scikit learn):
COVID.csv:



```
Accuracy: 1.0000
Precision: 1.0000
Recall: 1.0000
```

	0.0	1.0	2.0
0.0	6764	0	0
1.0	0	2604	0
2.0	0	0	778

Heart.csv:



```
Accuracy: 0.7333
Precision: 0.7343
Recall: 0.7086
```

	0	1
0	30	11
1	5	14

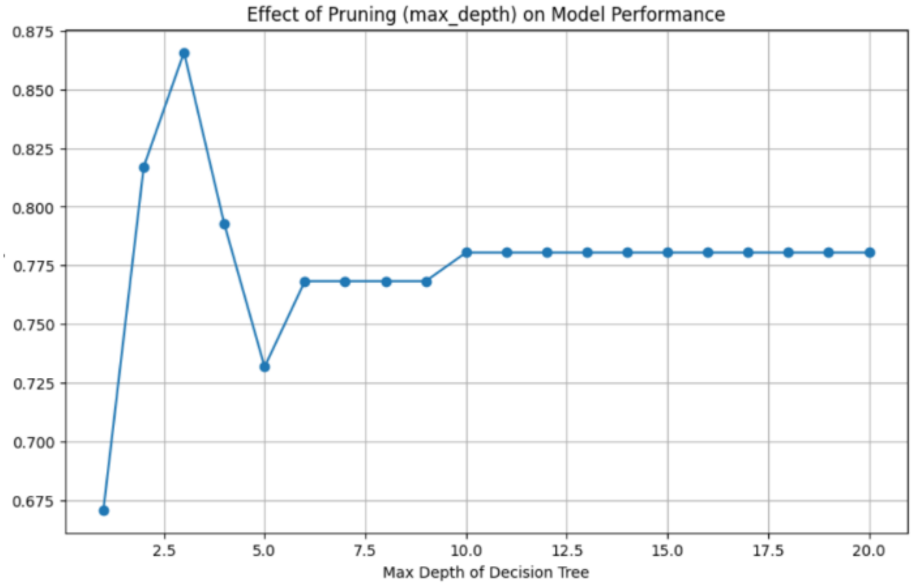
In this dataset, we achieved better results. Since the dataset was imbalanced, we used the SMOTE approach to resample before training our model.

Effect of Pruning on Model Performance

To examine the effect of pruning, we trained and evaluated DecisionTree models with different maximum tree heights (‘max_depth’).

Typically, as the depth increases, the model captures more complexity in the data, but after a certain point, further increasing the depth may lead to overfitting, which reduces the model's ability to generalize to new data.

The final accuracy on the test set using the best tree depth gives an indication of the model's performance after pruning. This process helps in finding a balanced model that neither overfits nor underfits the data.


Effect of Pruning (max_depth) on Model Performance
Max Depth of Decision Tree