Data Science and Data Mining

Spring 2023

Classification of Adult Income Using Decision Tree

Roland Fiagbe
University of Central Florida, fiagberoland@Knights.ucf.edu

Classification of Adult Income Using Decision Tree*

Roland Fiagbe
Department of Statistics and Data Science
University of Central Florida
Orlando, United States
fiagberoland@knights.ucf.edu

***Abstract*—Decision tree is a commonly used data mining methodology for performing classification tasks. It is a tree-based supervised machine learning algorithm that is used to classify or make predictions in a path of how previous questions are answered. Generally, the decision tree algorithm categorizes data into branch-like segments that develop into a tree that contains a root, nodes, and leaves. This project seeks to explore the decision tree methodology and apply it to the Adult Income dataset from the UCI Machine Learning Repository, to determine whether a person makes over $50K$ per year and determine the necessary factors that improve an individual's income. The model was evaluated using the classification metrics. The results show a good performance of the model. Also, the feature importance scores were computed to determine the contributing factors that improve an individual's income.**

***Index Terms*—Decision Tree, Income, Classification**

I. Introduction

One of the most important applications of machine learning is the use of classification algorithms to perform tasks. Several machine learning techniques have been developed over the years for classification. Decision tree is one of the commonly used machine learning techniques for establishing classification based on multiple variables. It is used for building a predictive algorithm for a categorical target variable. However, decision tree can be applied to both discrete and continuous variables as response variables or predictor variables. In recent years, decision tree methodology has become very useful in medical research, engineering, law, business, among others.

Generally, the decision tree algorithm categorizes data into branch-like segments that develop into a tree that contains a root, nodes, and leaves. The path of the tree begins from the root and the data is separated sequentially until a Boolean outcome is achieved at the leaf node [3]. It is considered a non-parametric method and it is applicable to large and complicated data. Some of the common usages of decision tree are variable selection, assessing relative importance variables, handling missing values, data manipulation, and prediction [4].

In this project, we want to apply the decision tree methodology to the adult income data to predict an individual's income and determine the necessary factors that improve an individual's income.

II. Data

The data for this project is a popular dataset available on the University of California Irvine (UCI) Machine Learning Repository [2]. It was extracted by Barry Becker from the 1994 Census database. The sets of records were extracted using the following conditions: $((AAGE > 16)\&(AGI > 100)\&(AFNLWGT > 1)\&(HRSWK > 0))$ and its prediction task is to determine whether a person makes over $50K$ per year. The data set is made up of 32561 observations and 14 predictor variables for 42 different countries around the world. The variables are listed below;

Feature Name	Type
Age	Continuous
Workclass	Categorical
Fnlwgt	Continuous
Education	Categorical
Education-num	Continuous
Marital Status	Categorical
Occupation	Categorical
Relationship	Categorical
Race	Categorical
Sex	Categorical
Capital-gain	Continuous
Capital-loss	Continuous
Hours-per-week	Continuous
Native-country	Categorical

TABLE I
List of Variables

The target variable is a binary class data that predicts whether an individual earns $> 50k$ or $\leq 50k$ per year based on the given attributes.

III. Exploratory Data Analysis and Data Visualization

In this section, we will perform some exploratory analysis and data visualization. This helps to have a summary and visual description of the data. Figure (1) shows the histogram of the income class. The histogram reveals an imbalance of the target variable in the data. However, this would not have a significant effect on the algorithm and hence would not be classified as a problem in our study.
Figure (2) shows the heatmap that visualizes the correlation

between the continuous variables and we can observe a weak correlation among the variables. Also, the Box and Whisker plot of all the continuous variables are shown below.

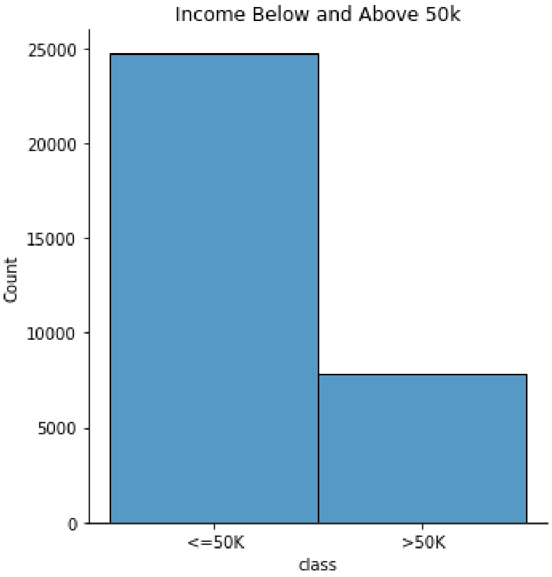


Fig. 1. Histogram for Income Class

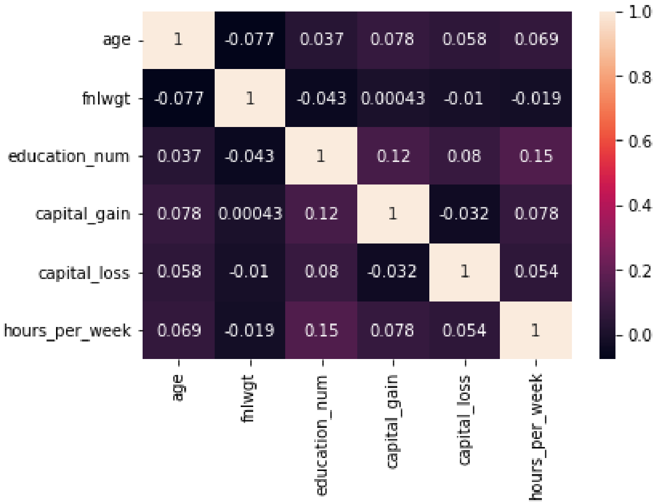
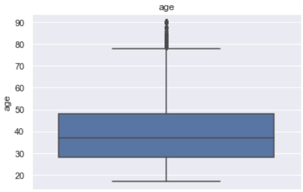
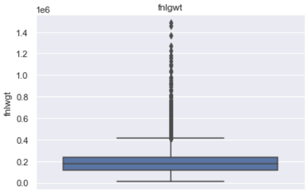


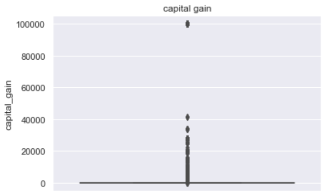
Fig. 2. HeatMap Matrix of Continuous Variables



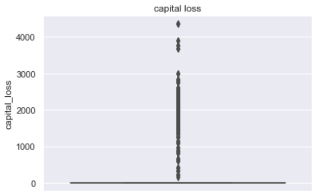
Boxplot for Age



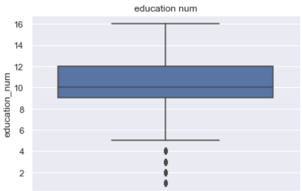
Boxplot for fnlgwt



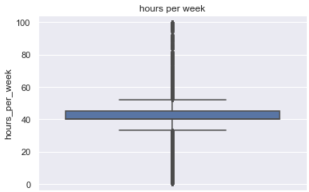
Boxplot for Capital Gain



Boxplot for Capital Loss



Boxplot for Education Num



Boxplot for Hours Per Week

IV. Methodology

A. *Decision Tree Algorithm*

In data mining, decision tree is one of the widely used classification techniques that can be applied to vast volumes of data. It is a tree-based supervised machine learning that is used to classify or make predictions in a path of how previous questions are answered [1]. Figure (3) illustrates the structure and main components of a decision tree model. (Image source [3]).

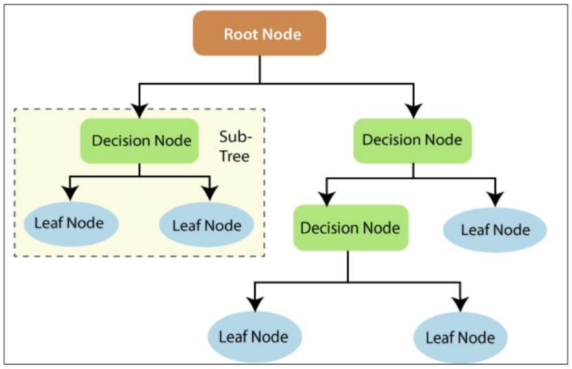


Fig. 3. Structure of Decision Tree Algorithm

As illustrated in Fig 3, the main components are nodes and branches and the steps in building the tree are splitting, stopping, and pruning the tree. The tree is made up of three types of nodes; that is, (1) the root node that represents a point that will be subdivided into two mutually exclusive subsets. (2) Internal node which represents the possible choices at each point of the tree structure. (3) The leaf node is also known as end node and it is the final point of the combination of decisions. In general, in the tree structure, each node represents a features (age, workclass, fnlwgt, education, etc) that is set to be classified into two mutually exclusive subsets and each subset defines a value that is observed by the node. The branches of the tree indicate the flow of outcomes from the root nodes and internal nodes.

B. Data Preprocessing

In this section, we consider preprocessing the data to make the features compatible with the algorithm. The data set consists of 32561 instances, 8 categorical features, 6 continuous features, and a categorical target variable that indicates whether an individual earns $> 50k$ or $\leq 50k$. The data set does not contain any missing values.

1) Encoding Categorical Features: For all the categorical features, we applied label encoding. The label encoding replaces the categories of a feature with a numeric value between 0 and the number of classes minus 1. For example, a feature with 3 classes is categorized into $0, 1, 2$. This is an important step for tree-based algorithms. Also, we applied label encoding to the target variable into a classification of $> 50k \longrightarrow 1$ and $\leq 50k \longrightarrow 0$

2) Data Splitting: Now, we split the dataset into a training and a test set into 70% 30% ratio. 70% of the dataset is used to train the model and 30% of the data is used to test the model.

C. Model Building

In this section, using sklearn module in Python, we develop a decision tree model to predict an individual's income level. We began by building a decision tree model with all the default hyperparameters; $criterion$ = 'gini', $max\_depth = 5$, $min\_samples\_leaf = 1$, $min\_samples\_split = 2$. Fig 4 is a visualization of the tree built with the default hyperparameters.

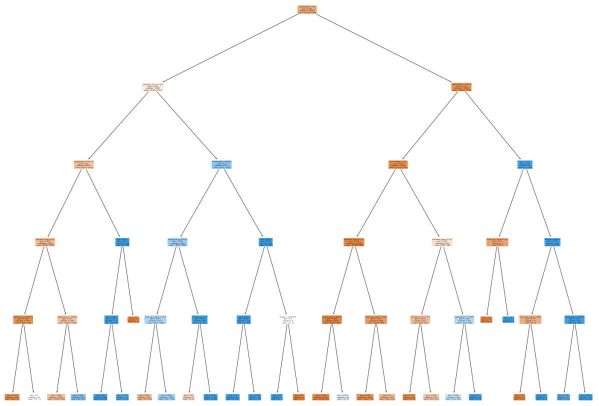


Fig. 4. Decision Tree Model for Default Hyperparameters

However, in the second case, we considered tuning the hyperparameters to build a simpler tree. The candidate hyperparameters were set up as follows;

```
    param_grid = {
    'max_depth': range(5, 15, 5),
    'min_samples_leaf': range(50, 150, 50),
    'min_samples_split': range(50, 150, 50),
    'criterion': ["entropy", "gini"]
}
```

We used Grid-Search with 5 folds to find the combination of multiple optimal hyperparameters. The optimal hyperparameters obtained were $criterion$ = 'gini', $max\_depth = 10$, $min\_samples\_leaf = 50$, $min\_samples\_split = 50$. Fig 5 is a visualization of the tree built with the optimal hyperparameters. In the decision tree shown below, the final leaf nodes classify whether an individual earns below or above $50k$ per year.

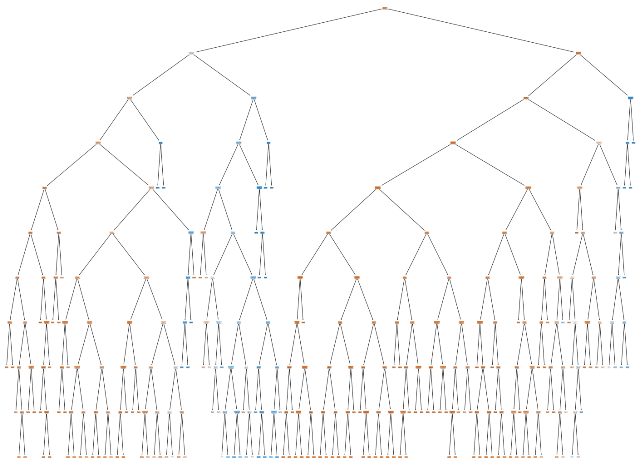


Fig. 5. Decision Tree Model with Best Hyperparameters

V. Results and Discussions

This section presents the results from the analysis discussed in the previous section. Out of the 32561 total observations in the dataset, 22792 observations were used to train the model and 9769 observations were used to test and evaluate the model. The performance of the model was evaluated using the following metrics.

- **Model accuracy:** The model accuracy measures the fraction of correctly classified observations by the model. It is computed as;

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}$$

From the model, the training accuracy is 0.85854 and the test accuracy is 0.85290

- **Sensitivity:** The sensitivity score of the model indicates the fraction of observations that were correctly classified as positives. It is computed as

$$TPR = \frac{TP}{TP + TN}$$

However, the model resulted in a sensitivity of 0.9337 for training and a sensitivity of 0.9286 for testing.

- **Precision:** This defines the fraction of observations that are correctly classified as positives out of the total positive observations. It is computed as

$$Precision = \frac{TP}{TP + FP}$$

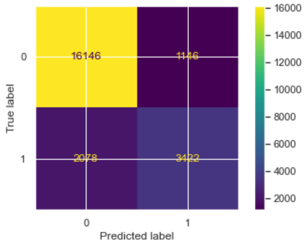
From the model, the precision for training is 0.8859 and 0.8837 for testing.

- **F1-Score:** This is the harmonic mean of precision and sensitivity and it is computed as

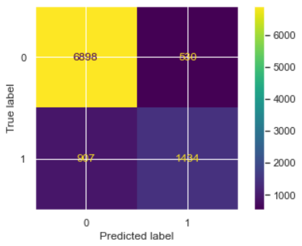
$$F_1 = 2\left(\frac{Precision \times Sensitivity}{Precision + Sensitivity}\right)$$

From the model, the F1-score for training is 0.6798 and testing is 0.6662.

The plot below shows the confusion matrix for the training data and test data. The confusion matrix was used to compute the model performance metrics. Moreover, table II and Fig 6 give the list and histogram of all the features and their importance scores.



Confusion Matrix (Train)



Confusion Matrix (Test)

Index	Feature Name	Variable Importance Score
0	age	0.04310
1	fnlwgt	0.01118
2	education-num	0.21607
3	capital-gain	0.20790
4	capital-loss	0.04447
5	hours-per-week	0.04300
6	workclass	0.00916
7	education	0.00274
8	marital-status	0.00061
9	occupation	0.00928
10	relationship	0.41143
11	race	0.00000
12	sex	0.00104
13	Native-country	0.00001

TABLE II
FEATURE IMPORTANCE SCORES

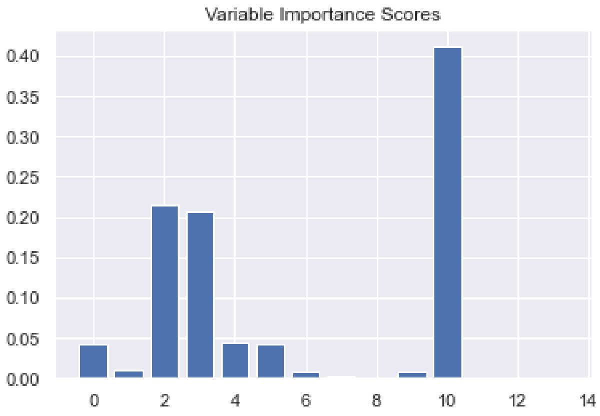


Fig. 6. Feature Importance Scores

VI. CONCLUSION

In this project, we have explored the decision tree methodology and applied it to predict the income of individuals based on collected features. The model was fitted with the optimal hyperparameters and the performance of the model was measured using the confusion matrix. From the results, we could observe that the model performs pretty well in predicting the income classes of individuals. An overall prediction accuracy of 85% was achieved by the model. The feature importance scores also show the variables that highly contribute to whether an individual earns above or below $50k$. Features with higher importance scores tend to contribute more to improving an individual's income whereas features with lower scores contribute less.

REFERENCES

[1] Emil Agbemade. "Predicting Heart Disease using Tree-based Model". In: (2023).

[2] *Census Income*. UCI Machine Learning Repository. 1996.

[3] Bahzad Charbuty and Adnan Abdulazeez. "Classification based on decision tree algorithm for machine learning". In: *Journal of Applied Science and Technology Trends* 2.01 (2021), pp. 20–28.

[4] Yan-Yan Song and LU Ying. "Decision tree methods: applications for classification and prediction". In: *Shanghai archives of psychiatry* 27.2 (2015), p. 130.