DAL Assignment 4: A Mathematical Essay On Decision Tree

C Vamshi Krishna Reddy
CH20B112
EE5708: Data Analytics Laboratory
IIT Madras, TamilNadu, India
ch20b112@smail.iitm.ac.in

***Abstract*—This document serves as a model for showcasing the application of the Decision Tree to build a model that helps us to classify a car into any one of the following classes: very good, good, acceptable, and unacceptable based on six different parameters like buying price, price of the maintenance, number of doors, capacity in terms of person to carry, size of luggage boot and the estimated value of the car.**

***Index Terms*—Classification, Decision Tree, Statistical Modeling, Supervised Learning**

I. Introduction

We all know that classification problems are one of the primary tasks solved with the help of Machine learning apart from regression. These algorithms create patterns between input and output variables using already existing data and classify the target variable based on the new input data provided. Broadly speaking there are 3 types of machine learning algorithms, namely, supervised, unsupervised and reinforcement learning. In this paper we are using Decision trees, which come under the supervised category.

A decision tree model is a supervised learning technique in which algorithms are taught from the labels associated with the data. It is a predictive model that uses a set of binary rules to determine the value of the target variable. Classification models are produced by decision trees in the shape of trees. By employing the potential outcomes of each attribute as a branch of the tree, this form aids in understanding the decision hierarchy and relationships between the qualities. Apart from these advantages, it is also important to note that these decision trees are prone to overfitting and they usually can not reach the level of accuracy provided by alternative classification methods and also any small change in the data can significantly affect the tree structure for predictions.

This paper aims to use Decision Trees in explaining the effect of buying price, price of maintenance, number of doors, etc., on the class of the car which can help us in choosing whether to buy a car or not.

The paper provides a detailed overview of Decision Trees by solving a real-world problem and gives insights into the relation between variables like buying price, price of maintenance, number of doors, capacity in terms of persons to carry, size of luggage boot, and the estimated safety value of the car and the class of the car.

II. Decision Trees

Classification is a widely employed application in machine learning, especially in supervised settings. It essentially helps us understand how certain independent factors determine which class it belongs to. In simpler terms, it's like a tool used in machine learning to make classifications. The core idea is to teach algorithms how different independent variables are linked to an outcome or dependent variable. Once the model grasps this relationship, it becomes capable of predicting what might happen when faced with new, unexpected data or completing missing parts in existing data.

A decision tree is a tree structure that resembles a flowchart, where each internal node represents a test on an attribute, each branch a test result, and each leaf node (terminal node) a class label. Because of their interpretability, decision trees are considered to be the most effective and well-liked technique for categorization and prediction. Any decision tree is composed of three sections: root node, branches, and leaves. The root node is the position of the first split. Branches are the subsequent questions that classify deeper and leaves are the end nodes of a decision tree where the prediction process is completed. Based on the type of output, In machine learning Decision trees are classified into two subgroups: Classification trees and regression trees. These two types of algorithms fall into the category of "classification and regression trees" and are popularly referred to as CART. Where their respective roles are to "classify" and to "predict."

• **Classification Trees:** In this Decision tree, the dependent variable determines whether an event happened or didn't happen. This is the most commonly used approach in cases of decision-making problems in real life. A simple example would be to predict if we need to carry an umbrella or not based on the temperature, humidity, and wind speed.

• **Regression Trees:** In these Decision trees, the dependent variable can take continuous values based on previous data or information sources. These types of decision trees are used more in programming algorithms, where our goal is to predict what is likely to happen, given previous behaviour or trends. A simple example would be to estimate the overall strength of a school based on the previous year's data and student feedback.

A. Cost Function

In the process of training decision trees, each split is determined by a local metric, such as information gain/entropy or the Gini index. This involves a greedy search rather than a cost function. Even when a global training metric like likelihood is defined, the evaluation of each training step still relies on these local metrics.

• **Entropy:** Entropy can be defined as the impurity in the data set provided. A decrease in entropy is referred to as information gain. Information gain calculates the difference between the average entropy after splitting and the entropy before splitting the data set.The attribute with the highest information gain is chosen as the splitting attribute at the node.

$$Entropy = \sum_{i=1}^{c} -p_i \log(p_i) \tag{1}$$

• **Gini Index:** Gini index measures the likelihood that a selected variable would be incorrectly classified when chosen at random. The degree of the Gini index lies between 0 and 1, where 0 denotes that all elements belong to a certain class and 1 denotes that the elements are randomly distributed across various classes. The attribute with a minimum Gini index for the given data is selected as a splitting attribute at that node.

$$GiniIndex = 1 - \sum_{i=1}^{c} p_i^2 \tag{2}$$

B. Evaluation Model

In classification models like Decision Trees, confusion matrix, and f1 score are used to evaluate the performance of the model. Here since our problem is a Logistic regression which has 4 output classes. We have a confusion matrix of 4 X 4 dimension, And F1-Score is defined as the harmonic mean of recall and precision values of the classification problem. The formula for F1-Score is as follows:

$$F1 = \frac{2 * precision * recall}{precision + recall} \tag{3}$$

III. The Problem

The objective is to develop a model to help the person in deciding whether to buy a car or not based on factors like buying price, price of maintenance, the number of doors, capacity in terms of persons to carry, size of luggage boot, and the estimated safety value of the car as the input features. The process initiates with data cleaning and then exploratory data analysis where we predict the 4 possible classes for cars: unacceptable, acceptable, good, and very good.

A. Data Cleaning

The given dataset contains around 1727 rows and 7 columns. Upon looking at the information of data we see that there are no missing values or nulls in the data set. But it is always a good idea to check if nulls are stored in some other format. Even after exploring these variables individually, we can see that there are no nulls in other forms too. So it is safe to say that our dataset is free from nulls or missing values and do not require any further steps of dropping rows or filling up missing data with central tendencies etc.

B. Exploratory Data Analysis and Visualisations

The provided dataset includes car information like maintenance and buying price, the capacity of the car, etc. These details of the car will be used to train a model that can classify a car as unacceptable (or) acceptable (or) good (or) very good, based on the "ground truth" i.e known data. Here we can assume that it would be acceptable to buy a car if it belongs to either acceptable, good, or very good and reject if it belongs to an unacceptable class.

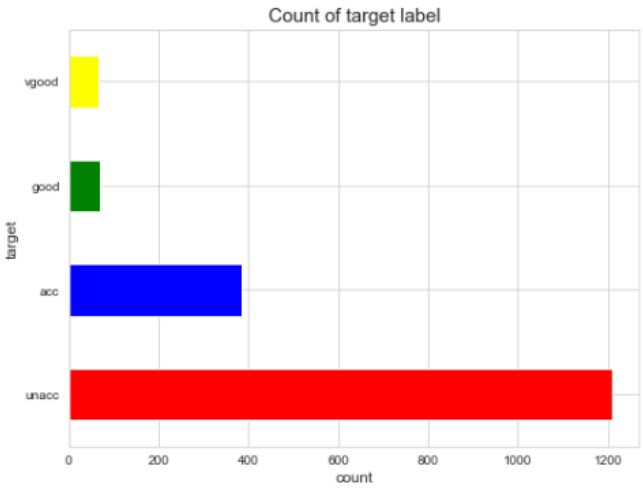

Fig. 1. Count of Target label.

Figure 1 shows a horizontal bar graph plot that tells the count of cars which belonged to four output labels. We can see that count of unacceptable cars is too high compared to remaining three classes count.

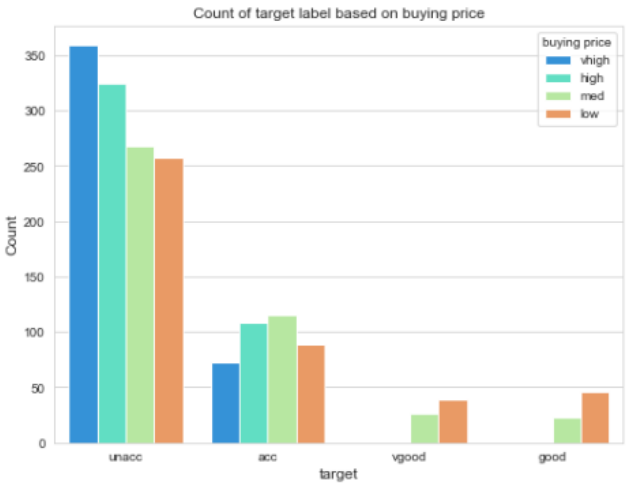

Fig. 2. Count of target labels based on buying price.

From Figure 2, we can see that it is not acceptable to buy a car if its price is too high or high i.e price of a car is inversely affects the probability of buying a car.

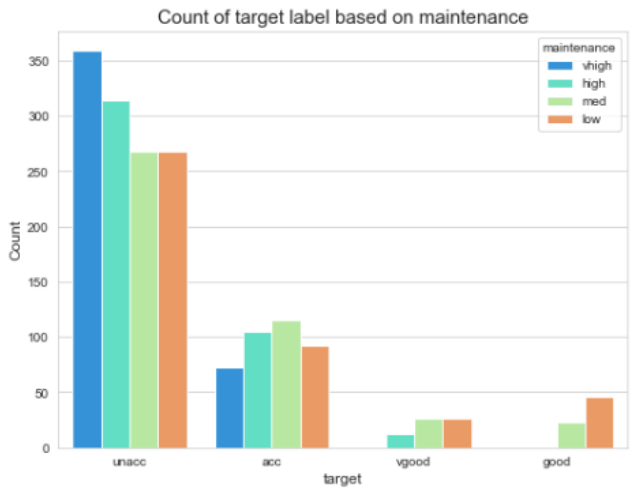

Fig. 3. Count of target labels based on the buying price of maintenance.

From Figure 3 we can see that if the maintenance price of a car is too high then the price of a car inversely affects the probability of buying a car and the count of unacceptable cars are also in good number.

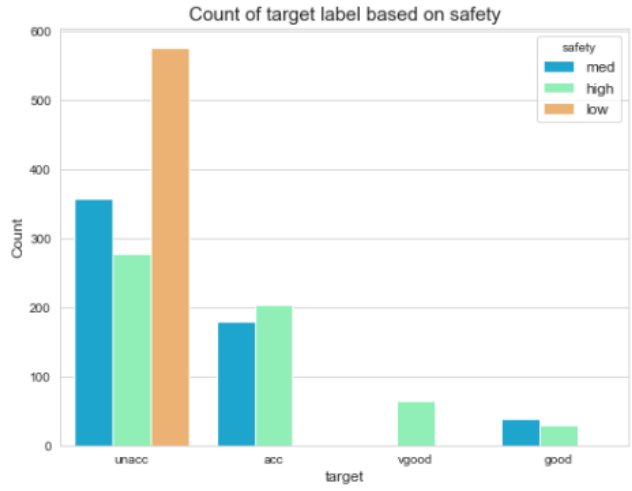

Fig. 4. Count of target labels based on safety of car.

Figure 4 gives idea of the effect of safety on the car's target label. We see that cars with low safety are more likely to be unacceptable to buy compared to others.

Fig.5 helps us to conclude that most of the cars that have a capacity of 2 persons are unacceptable to buy.

Fig.6 shows us that the size of the luggage boot directly affects the chance of acceptability as the cars with small luggage boots have a lesser acceptance rate.

Figure 7 we can see that the no of doors directly affects the chance of acceptability as the cars with less number of doors have a lesser acceptance rate.

Fig. 5. Count of target labels based on no of capacity of persons.

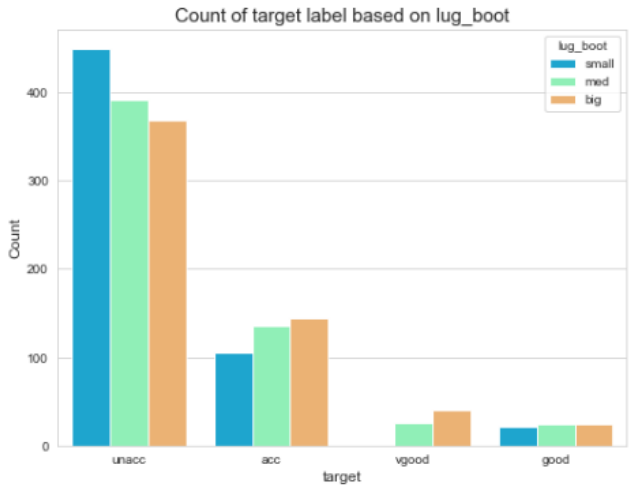

Fig. 6. Count of target labels based on no of lugboot.

C. Model

Now a Decision tree model as shown in Fig.7 is developed on this data set. This data set is first split into 2 data sets x and y. x comprises columns buying, maint, doors, persons, lugboot and safety and y comprises of the Target. We have decided to split these data sets using a train test split with a test size of 0.33 and a random state of 42. To train this model we need to encode the classes in x dataset. As all the columns in x are categorical I used Ordinal Encoder from the category encoders package. Now a model is developed by fitting X train and y train using a Decision tree classifier from sklearn. This trained model is tested by fitting the predicted classes and comparing them with the known classes of the Target column. Using accuracy score from sklearn.metrics we can see that our classification model has an accuracy of 0.8539 on the training data set and an accuracy of 0.82 on the test data set. As both accuracies are in the same range we can conclude that our classification model does not overfit the training data.

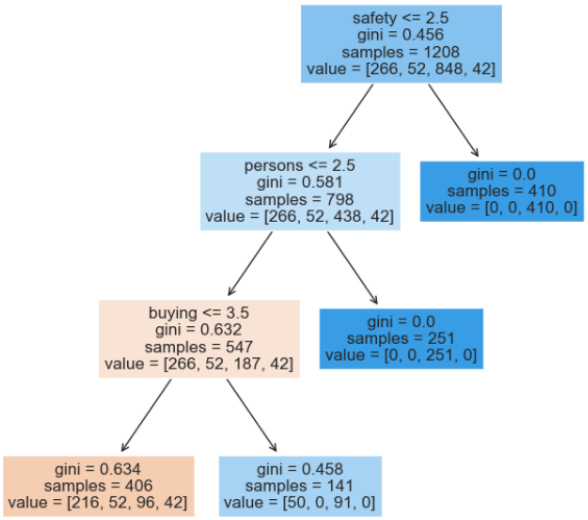

Fig. 7. Decision tree classifier.

And also from the classification report, we can see that the F1 score of our model is 0.82.

	precision	recall	f1-score	support
acc	0.81	0.56	0.67	170
good	0.00	0.00	0.00	0
unacc	0.91	0.94	0.92	349
vgood	0.00	0.00	0.00	0
accuracy			0.82	519
macro avg	0.43	0.38	0.40	519
weighted avg	0.88	0.82	0.84	519

Fig. 8. Classification Report.

A practical issue when creating a Decision-Tree model is overfitting. When an algorithm keeps going deeper and deeper to reduce training-set error but ends up increasing test-set error as we create a lot of branches because of outliers and other anomalies in the data resulting in the decline of our model's forecast accuracy. Overfitting in a decision tree can be avoided either by Pre Pruning or post-pruning. In pre-pruning, we stop splitting a node if its goodness measure is below a threshold value. whereas in post-pruning, we go deeper and deeper in the tree to build a complete tree and then reduce the depth of the model if the tree shows overfitting. Cross-validation data is used to check the effect of this pruning. If expanding a node shows an improvement, then we can continue by expanding that node. But if it shows a reduction in accuracy then the node is converted to a leaf node. In this model, we had fixed the max depth of the tree to be 4 to avoid overfitting as a part of pre-pruning.

D. Insights and Observations

From the heat map of correlation, we can see that there is a positive correlation between safety, the capacity of the car, the size of the luggage and Target class i.e car with more safety (or) more capacity (or) larger size of the luggage is more likely to belong to acceptable classes (acc, good, vgood). We can also observe a negative correlation between Buying price, maintenance price and the Target class i.e Cars with a larger buying price or maintenance cost are more likely to be in the unacceptable class.

Fig. 9. Heap map of correlation.

IV. Conclusion

Our Decision tree model classifies a car into any one of the four classes: very good, good, acceptable and unacceptable with the accuracy of 0.8539 or 85.4 percent. The findings show that safety is the most important factor for car consumers. A customer wouldn't purchase a car if they believe it is unsafe. The number of passengers it can accommodate also plays an important role in choosing a car as the data set suggests that customers do not purchase cars with less than 2 seats or more than 4 seats. The maintenance cost is also taken into account and cars with high maintenance costs are not favoured by the people. The purchase price also determines if it is viable to buy the car or not. The last factor, luggage capacity can also prevent people from buying as the majority of people in the data set rejected cars with small capacity. The training-set and test-set accuracy scores of our model shows that our model did not overfit the training data and the confusion matrix and classification reports also validate the goodness of our model.

References

[1] Bishop, Christopher M., Pattern Recognition and Machine Learning, 2006.
[2] https://seaborn.pydata.org/tutorial/categorical.html.
[3] Aurelien Geron., Hands on Machine Learning with Scikit-Learn and Tensorflow, 2019.
[4] https://www.geeksforgeeks.org/decision-tree/.
[5] https://towardsdatascience.com/the-f1-score-bec2bbc38aa6.