Decision Tree for Classification to evaluate Customer acceptability of Cars

Jeshlin Donna J
Department of Metallurgical and Materials Engineering
Indian Institute of Technology Madras
Chennai, India
mm20b029@smail.iitm.ac.in

***Abstract*—The key task of this paper is to evaluate customer acceptability of Cars. The aim is use and apply Decision Tree and perform Exploratory data analysis to uncover previously unknown or hidden facts in the data set available, visualize them using Data Visualization and make predictions on the validation data. This is followed by model evaluation on the Decision Tree model. The task is to conclude the study of which types of cars are more likely to have higher customer acceptability.**

***Index Terms*—Decision Tree model, Exploratory data analysis, Data Visualization, Model Evaluation**

I. INTRODUCTION

The data considered was extracted from the UCI Machine learning repository. The key task is to evaluate customer acceptability of Cars. The aim is use and apply Decision Tree and perform Exploratory data analysis to uncover previously unknown or hidden facts in the data set available, visualize them using Data Visualization and make predictions on the validation data. The task is to conclude the study of which types of cars are more likely to have a higher customer acceptability level.

Decision tree learning is one of the predictive modelling approaches used in statistics, data mining and machine learning. It uses a decision tree (as a predictive model) to go from observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves). Tree models where the target variable can take a discrete set of values are called classification trees; in these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels. Decision trees where the target variable can take continuous values (typically real numbers) are called regression trees.

The goal of this research paper is to correctly predict which kind of cars are more likely to have a higher customer acceptability given a set of information. A predictive classification model based on Decision Tree was built using the given data so that the 'Buying price', 'Price of the maintenance', 'Number of doors', 'Capacity', 'Estimated safety of the car' and other features contributed to their customer acceptability. The output of a cars customer acceptability is predicted using the Naive Bayes classifier based on different feature combinations of the data.

In this paper, we conducted exploratory data analysis to excavate different knowledge existing in the given data set and to perceive the impact of every field with respect to the customer acceptability by the use of 'Customer Acceptability' field analysis in between each field of the data set. Data pre-processing was done, data analysis was performed and, a Decision Tree Classifier model was built and trained on the training data, and the accuracy was tested on the validation dataset. After analyzing the dataset, we discovered insights and information on what the cars with high customer acceptability had in common that helped them do so. We were also able to predict if a particular car with certain feature values would have a good customer acceptability level by applying the tools of machine learning and statistical analysis.

II. DECISION TREE

A decision tree or a classification tree is a tree in which each internal (non-leaf) node is labeled with an input feature. The arcs coming from a node labeled with an input feature are labeled with each of the possible values of the target feature or the arc leads to a subordinate decision node on a different input feature. Each leaf of the tree is labeled with a class or a probability distribution over the classes, signifying that the data set has been classified by the tree into either a specific class, or into a particular probability distribution (which, if the decision tree is well-constructed, is skewed towards certain subsets of classes).

A tree is built by splitting the source set, constituting the root node of the tree, into subsets—which constitute the successor children. The splitting is based on a set of splitting rules based on classification features. This process is repeated on each derived subset in a recursive manner called recursive partitioning. The recursion is completed when the subset at a node has all the same values of the target variable, or when splitting no longer adds value to the predictions. This process of "top-down induction of decision trees" is an example of a greedy algorithm, and it is by far the most common strategy for learning decision trees from data.

Data comes in records of the form: $(\mathbf{x}, Y) = (x_1, x_2, x_3, ..., x_k, Y)$

The dependent variable, Y, is the target variable that we are trying to understand, classify or generalize. The vector $\mathbf{x}$ is composed of the features, x_1, x_2, x_3 etc., that are used for that task.

A. *Decision tree types*

Decision trees used in data mining are of two main types: Classification tree analysis is when the predicted outcome is the class (discrete) to which the data belongs.Regression tree analysis is when the predicted outcome can be considered a real number. The term classification and regression tree (CART) analysis is an umbrella term used to refer to either of the above procedures. The trees used for regression and trees used for classification have some similarities – but also some differences, such as the procedure used to determine where to split.

B. *Metrics*

Algorithms for constructing decision trees usually work top-down, by choosing a variable at each step that best splits the set of items. Different algorithms use different metrics for measuring "best". These generally measure the homogeneity of the target variable within the subsets. Some examples of metrics are given below. These metrics are applied to each candidate subset, and the resulting values are combined (e.g., averaged) to provide a measure of the quality of the split.

1) Gini impurity: Gini impurity is a measure of how often a randomly chosen element from the set would be incorrectly labeled if it was randomly labeled according to the distribution of labels in the subset. The Gini impurity can be computed by summing the probability p_i of an item with label i being chosen times the probability $\sum_{k \neq i} p_k = 1 - p_i$ of a mistake in categorizing that item. It reaches its minimum (zero) when all cases in the node fall into a single target category.

To compute Gini impurity for a set of items with J classes, suppose $i \in \{1, 2, ..., J\}$, and let p_i be the fraction of items labeled with class i in the set.

$$I_G(p) = \sum_{i=1}^{J} \left(p_i \sum_{k \neq i} p_k \right) = \sum_{i=1}^{J} p_i (1 - p_i) = \sum_{i=1}^{J} (p_i - p_i^2) \tag{1}$$

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

2) Information gain: Information gain is based on the concept of information entropy and information content.

Entropy is defined as below

$$(T) = \mathrm{I}_E\left(p_1, p_2, \ldots, p_J\right) = -\sum_{i=1}^{J} p_i \log_2 p_i \tag{3}$$

where $p_1, p_2, \ldots$ are fractions that add up to 1 and represent the percentage of each class present in the child node that results from a split in the tree.

$$\overbrace{IG(T,a)}^{\text{information gain}} = \overbrace{(T)}^{\text{entropy (parent)}} - \overbrace{(T \mid a)}^{\text{sum of entropies (children)}} \tag{4}$$

$$= -\sum_{i=1}^{J} p_i \log_2 p_i - \sum_{i=1}^{J} - \Pr(i \mid a) \log_2 \Pr(i \mid a) \tag{5}$$

Averaging over the possible values of A,

$$\overbrace{E_A(\mathrm{IG}(T,a))}^{\text{expected information gain}} = \overbrace{I(T;A)}^{\text{mutual information between } T \text{ and } A} \tag{6}$$

$$= \overbrace{(T)}^{\text{entropy (parent)}} - \overbrace{(T \mid A)}^{\text{weighted sum of entropies (children)}} \tag{7}$$

$$= -\sum_{i=1}^{J} p_i \log_2 p_i - \sum_{a} p(a) \sum_{i=1}^{J} - \Pr(i \mid a) \log_2 \Pr(i \mid a) \tag{8}$$

That is, the expected information gain is the mutual information, meaning that on average, the reduction in the entropy of "T" is the mutual information.

Information gain is used to decide which feature to split on at each step in building the tree. For simplicity, we want to keep our tree small. To do so, at each step we should choose the split that results in the most consistent child nodes. A commonly used measure of consistency is called Information theory which is measured in bits. For each node of the tree, the information value "represents the expected amount of information that would be needed to specify whether a new instance should be classified yes or no, given that the example reached that node".

To build the tree, the information gain of each possible first split would need to be calculated. The best first split is the one that provides the most information gain. This process is repeated for each impure node until the tree is complete.

3) Measure of "goodness": The measure of "goodness" is a function that seeks to optimize the balance of a candidate split's capacity to create pure children with its capacity to create equally-sized children. This process is repeated for each impure node until the tree is complete. The function $\varphi(s \mid t)$, where s is a candidate split at node t, is defined as below

$$\varphi(s \mid t) = 2P_L P_R \sum_{j=1}^{\text{class count}} |P(j \mid t_L) - P(j \mid t_R)| \quad (9)$$

where t_L and t_R are the left and right children of node t using split s, respectively; P_L and P_R are the proportions of records in t in t_L and t_R, respectively; and $P(j \mid t_L)$ and $P(j \mid t_R)$ are the proportions of class j records in t_L and t_R, respectively.

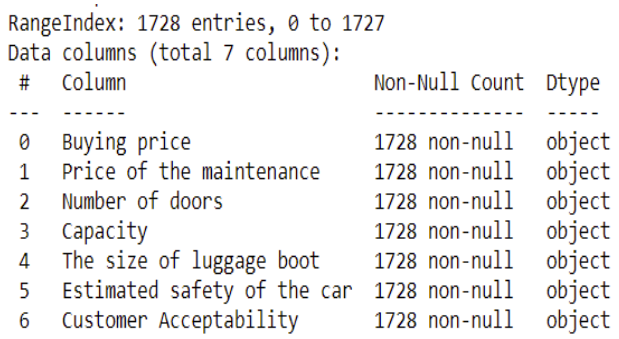
To build the tree, the ”goodness” of all candidate splits for the root node need to be calculated. The candidate with the maximum value will split the root node, and the process will continue for each impure node until the tree is complete.

Compared to other metrics such as information gain, the measure of ”goodness” will attempt to create a more balanced tree, leading to more-consistent decision time. However, it sacrifices some priority for creating pure children which can lead to additional splits that are not present with other metrics.

III. Using Decision Tree for Classification to Evaluate Customer Acceptability of Cars

A. Description of the Dataset

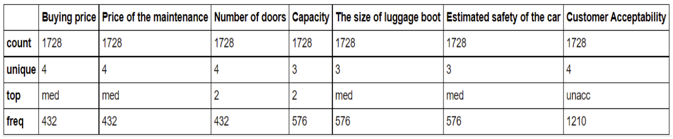
The data considered was is the Car evaluation dataset is taken from UCI Machine learning repository derived from simple hierarchical decision model.



```
RangeIndex: 1728 entries, 0 to 1727
Data columns (total 7 columns):
 #   Column                      Non-Null Count  Dtype 
---  ------                      --------------  ----- 
 0   Buying price                1728 non-null   object
 1   Price of the maintenance    1728 non-null   object
 2   Number of doors             1728 non-null   object
 3   Capacity                    1728 non-null   object
 4   The size of luggage boot    1728 non-null   object
 5   Estimated safety of the car 1728 non-null   object
 6   Customer Acceptability      1728 non-null   object
```

Fig. 1. Information about the dataset

The dataset comprises of 1728 observations and 7 variables. Out of which one variable(’Customer Acceptability’) is the dependent variable and the rest 6 are independent variables.



	Buying price	Price of the maintenance	Number of doors	Capacity	The size of luggage boot	Estimated safety of the car	Customer Acceptability
count	1728	1728	1728	1728	1728	1728	1728
unique	4	4	4	3	3	3	4
top	med	med	2	2	med	med	unacc
freq	432	432	432	576	576	576	1210

Fig. 2. Statistics of the dataset

It was noted that all the variables in the considered dataset are categorical in nature and had values of ’object’ datatype(Fig.1. and Fig.2.).

B. Data Cleaning

The raw dataset was found to be having vague column/variable names, these variables were renamed with meaningful variable names after performing a background research on the dataset.

C. Visualizations and insights from the analysis

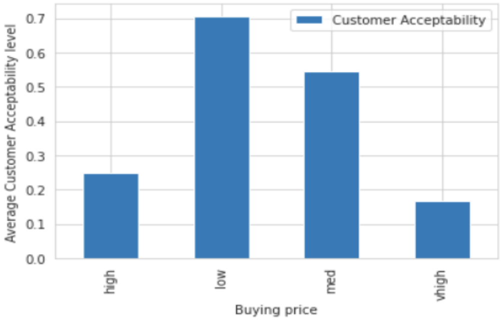


Fig. 3. Average Customer Acceptability level for different levels of Buying Price

From (Fig.3.) we infer that, lower the buying price, higher is the average customer acceptability level.

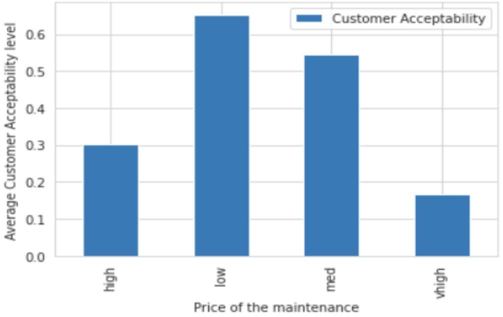


Fig. 4. Average Customer Acceptability level for different levels of Price of Maintenance

From (Fig.4.) we infer that, lower the price of maintenance, higher is the average customer acceptability level.

From (Fig.5.) we can conclude that, higher the number of doors in the car, higher is the average customer acceptability level.

From(Fig.6.) we can conclude that, cars with a capacity of 2 have zero customer acceptability level. And, cars with a capacity of 4 people was most preferred and had the highest average customer acceptability level.

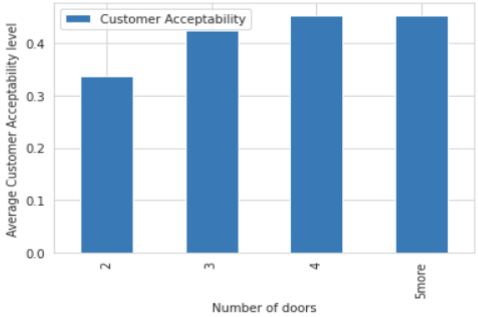


Fig. 5. Average Customer Acceptability level for different number of doors in the car

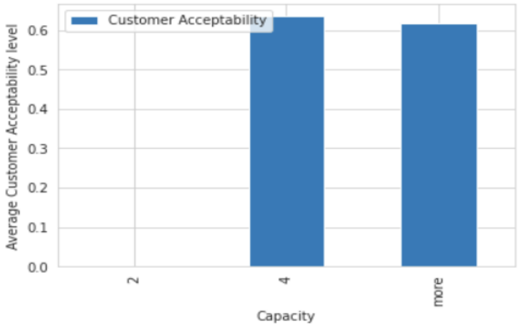


Fig. 6. Average Customer Acceptability level for different values of Car Capacity

From(Fig.7.) we can infer that the higher the size of luggage boot, higher is the average customer acceptability level.

From (Fig.8.), it can be said that, higher the estimated safety of the car, higher is the average customer acceptance level. Cars with 'low' safety estimation, had zero customer acceptability level.

D. Data Preprocessing

Categorical variables were created from the given variables of 'object' type. Since all the categories in the variables had a naturally increasing order, they were converted to numbers (e.g. 1, 2, 3, 4, 5) preserving the order. These are called ordinals.

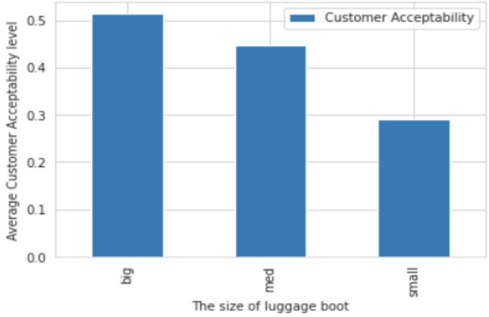


Fig. 7. Average Customer Acceptability level for different levels of Luggage Boot Size

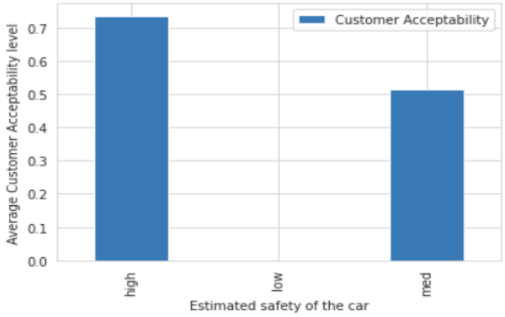


Fig. 8. Average Customer Acceptability level for different levels of Estimated Car Safety

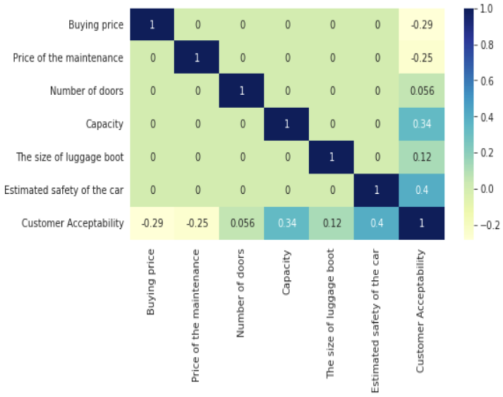


Fig. 9. Correlation heat-map between the various features of data

The correlation between no were found to be significant(Fig.9.). Hence there was no requirement to drop any variables before training to remove redundancy.

The dataset was also split into the training and validation parts with the validation data being used to calculate the accuracy and performance of the model before making predictions on the test data.

E. Building the Decision Tree Model

A Decision Tree Model(Fig.10.) has been built using the inbuilt 'DecisionTreeClassifier' from the Skikit-Learn library. The variables- 'Buying price', 'Price of the maintenance', 'Number of doors', 'Capacity', 'The size of luggage boot', 'Estimated safety of the car' were taken as the categorical inputs and were used to predict the categorical output variable-'Customer Acceptability'.

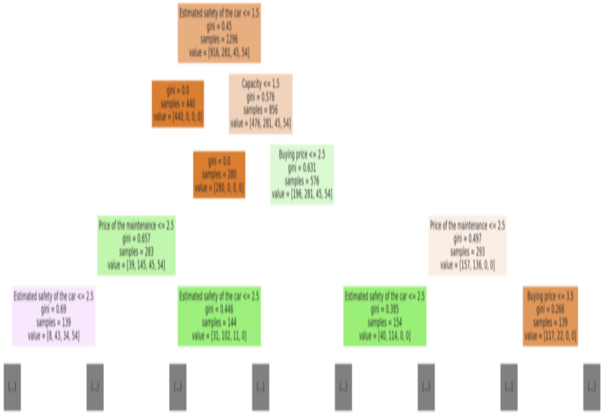


Fig. 10. Decision Tree Model after training on the training dataset

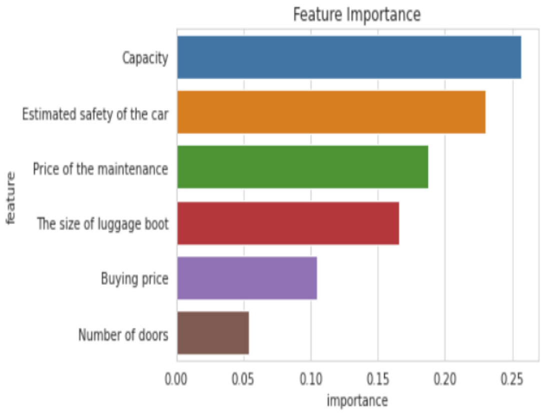


Fig. 11. Feature Imporatance

It was found out(Fig.11.) that the Capacity and Estimated Safety of the car were amongst the features/variables of utmost importance in the decision tree model.

F. Model performance

The dataset was split into the training and validation parts and the training dataset was used to train the model to fit the given data. The decision tree model was found to fit the training data with an astounding 100% accuracy.

The performance of the Decision Tree model was also evaluated on the validation dataset and, the accuracy of the model was found to be 97.22%. A confusion matrix of our model(Fig.12.) on the validation dataset was plotted to visualize the miss-classified datapoints.

IV. CONCLUSIONS

We analyzed the role of various features-'Buying price', 'Price of the maintenance', 'Number of doors', 'Capacity', 'The size of luggage boot', 'Estimated safety of the car' in estimating the 'Customer Acceptability' of cars. It was inferred that lower the buying price, higher was the average customer acceptability level. And, lower the price of maintenance, higher was the average customer acceptability level. It was also discovered that higher the number of doors in the car, higher was the average customer acceptability level. Cars with a capacity of 2 were found to have zero customer acceptability level. And cars with a capacity of 4 people was most preferred and had the highest average customer acceptability level. It was also found that the customer acceptability level was higher for higher the sizes of luggage boot. Safety of the car was found to be one of the features of utmost importance in customer acceptability. Cars with low safety estimation, had zero customer acceptability level. And cars with high estimated safety had a high average customer acceptance level.

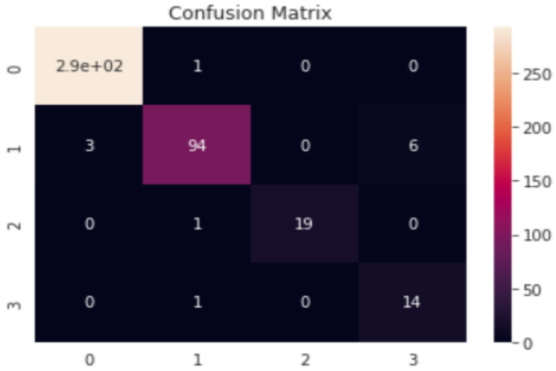


Fig. 12. Confusion matrix of our model on the validation dataset

Despite the high accuracy of the model on the validation dataset(97.22%), further improvements in the model are possible. In order to further improve the overall result, an extensive hyper-parameter tuning can be done to improve the accuracy of the model for it to better fit the data. It also is possible to further improve it by performing ensemble learning or possibly finding even better machine learning algorithms for the concerned task.

REFERENCES

[1] Patel, Harsh Prajapati, Purvi. (2018). Study and Analysis of Decision Tree Based Classification Algorithms. International Journal of Computer Sciences and Engineering. 6. 74-78. 10.26438/ijcse/v6i10.7478.

[2] Bramer, Max. (2020). Using Decision Trees for Classification. 10.1007/978-1-4471-7493-64.

[3] Engel, Joachim Erickson, Tim Martignon, Laura. (2020). TEACHING ABOUT DECISION TREES FOR CLASSIFICATION PROBLEMS.