DECISION TREES AND THEIR CLASSIFICATION USING THE CART ALGORITHM

Abstract

Exposeyour senses and get everywhere you will discover lots of machine learning apps that you use in your everyday life such as Facebook face recognition and getting recommendations forAmazon machine learning products used almost while online shopping. The decision tree is the throwing away of the practice of data withdrawal for planning and anticipation of Statistics. This proclamation tree is therefore a kind of editing process that is far from these controlled learning processes. In this chapter, we will clarify that association is the process of integrating between databases into separate classes or groups equally”. It is a strategy to classify perceptions into really different categories, retrieval of data, analyze it, and in terms of base, is well categorized into different categories and their different types with the help of the application case. Then we will see what the decision tree is and the various conditions associated with itby imagining the decision tree illuminating the unseen forest.In their preparation set, Naive Bayes, K-Nearest Neighbor, and this chapter also include a Decision Tree Classifier via Python using the CART Algorithm.

Keywords: Classification and Regression Trees (CART), Information Gain (IG), K-Nearest Neighbor (KNN), Probability (P), complexity parameter (CP

Authors

Vineeta Shrivastava
School of Computer Science
and Technology
LNCT University
Bhopal, India

Vaibhav Udgir
Software developer
Think Future Technologies Pvt.
Gurugram, India

I. INTRODUCTION

Expose your senses and gaze around you, you will find many machine learning tenders that you use and interact with in your daily life like facial recognition program on Face book and get recommendations for similar Amazon machine learning products used almost everywhere. This selection tree is therefore a type of organizational process that originates under these supervised information systems. So we should all be using Gmail. So you think how email is classified as non-Spam mail is not Spam there is nothing but segregation. "Separation is the process of dividing a database into different categories or groups by adding a standard". [1] It is a strategy of distinguishing perceived into different categories such as taking data, analyzing it and based on other factors is well divided into different categories. Now, why do we separate?

We break it down to make predictions predictable. such as when you receive a email the reader predicts that spam or non-spam email and based on that prediction adds unsolicited or spam emails to a specific folder, usually, this sorting algorithm deals with questions such as Who this facts belongs to. Whether it is a boy or a girl something like this now the query is wherein are you going to apply it? You can use this to hit upon fraud or test whether or not the movement is real or not.

Suppose I am and user of an acknowledgement score postcard in India now for some purpose I should visit Singapore they will question me to affirm their transaction[2].You can use it to differentiate one of a kind objects which includes fruit from satiated, color, length or weight of a well-educated device the usage of a easy category set of rules you could expect the class or sort of fruit on every occasion new statistics is provided.it is able to be anything which includes a car, a house, etc. This is called classification.

However, there are numerous exclusive methods to do the identical obligations together with predicting that a sure character gadget need to study first. But there are numerous methods to educate a gadget and you may pick any of them to get the predicted analysis. There are many different strategies but the most common is decision tree.

II. DECISION TREE

Graphical representation of how to take a decision is on a situation of hungriness and moneythis can be simply clarified as exposed in figure 1

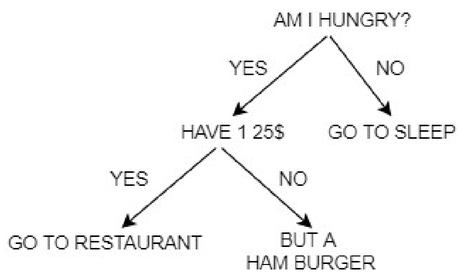


Figure 1: Decision tree

"The demisters tree work based on some conditions is shown in Figure 1."Eg. Here's occupation that speaks I have to energy to an eatery I have to purchase pizza. Confused about that? So, he will do it. You will create a choice tree that begins off&evolved with the foundation node can be the primary time you'll test in case you are hungry or now no longer. Unknown you aren't starving simply visit slumber. But in case you are hungry and feature 500 rs you'll determine to visit a eating place and in case you are hungry and do now no longer have 500 Rs. You'll simply pass and purchase pizza. So this is about the decision tree.

III. RANDOM FOREST

Build numerous decision trees and cover them there.Random forest methods give precisely more accurate forecast.The random decision forest is seamless for the decision tree repetition of overfilling in their exercise usual [3]. Competent with "bagging" method. The bagging technique is primarily founded totally at the concept that the mixture of gaining knowledge of the version complements the general effect. If you integrate gaining knowledge of from special fashions after which combining what you'll do it's going to boom the high-quality penalty. If the dimension of your database is large. However, if we use the balloting machine and ask special human beings to interpret the data. Then we are able to be capable of cowl the styles with exceptional care

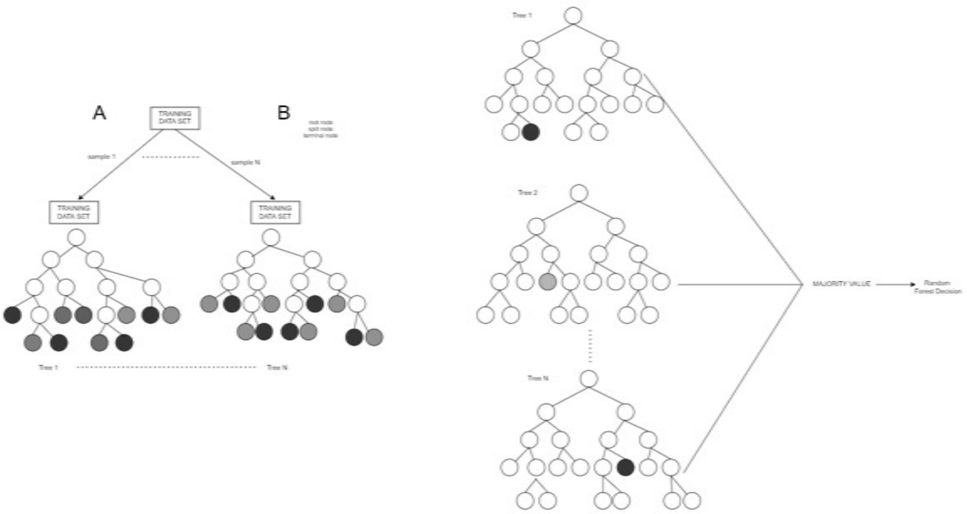


Figure 2: Decision tree with training data set

"The decision trees work by providing a large data set and dividing our training data set into n small samples as shown in Figure 2we can see that we've got a huge set of education records do we start by dividing our training data organized into n sub-sub-samples creating a decision tree for each such sample in section B, taking a vote on all decisions made by all decisive trees and disrespectfully voting for a vote in order to obtain compensation for a random forest decision.

IV. NAIVE BAYES

Naïve Bayes is an organisation system founded on Bayes Theorem

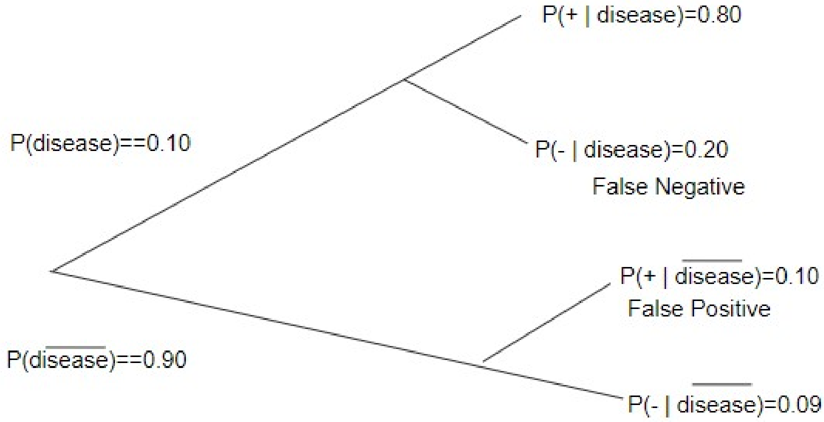
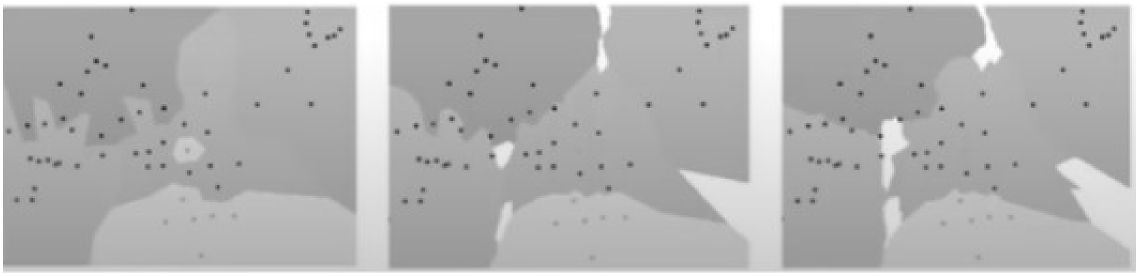


Figure 3: Naïve Bayes

"The Naive Bayes organization method founded on Bayes Theorem is shown in Figure 3."Undertake that the attendance of a specific component in the schoolroom is not connected to the company of any other component. Naive Bayes is humble and informal to custom algorithm and since of its effortlessness this algorithm may surpass where the records set length isn't always huge enough. Older use of naïve byes is the record separation you want to decide if the given textual content material corresponds to at the least one or more instructions with inside the text case the competencies used may be the presence or absence of any key-phrase so this turned into approximately Naïve Bayes so from Figure 2 you may see a way to use naïve byes we must determine if we've the ailment or now no longer. First, what we do is calling on the probabilities of having the ailment and now no longer getting the ailment. [3], [4]the risk of infection is 0.1 and on the other hand the risk of infection is 0.9 if you have the disease and visit the medical doctor. So in case you go to a medical doctor and feature a wonderful take a look at you're 0.eighty probabilities of getting a prognosis and 0.20 probabilities are in case you have already got the sickness. . [4] If you do now no longer devour the sickness at all of the odds of receiving it are 0.nine and in case you go to a medical doctor and the medical doctor is the equal, sure you've got got the sickness however you realize you do now no longer have it. I do now no longer have the sickness. So it's miles a fake record that. So the probability of life 0.ninety is similar to the probability which you do now no longer devour the sickness with side the take a look at suggests the equal consequences excellent declaration of 0.9.

1. K-Nearest Neighbour

The KNN or the K-Nearest Neighbour supplies all the obtainable bags and categorises novel cases based on a resemblance degree



K=1 K=3 K=5

Figure 4: Difference in the images of KNN

"The difference in the images when k=1, k=3 and k=5 and the “K” is the KNN algorithm is the nearest neighbors is shown in Figure 4."Eg. if K is equal to 1 then the thing is only allocated to the institution of that precise nearest neighbor from Figure 1.four you could break up the pics if k = 1, k = three and k = 5? to get the eye of the seen sample to check and find out hidden packages with inside the bottom cart of the shopping for cart in which the go out is observed whilst the precise identical object is in the shop in which the charge of the stained product may even robotically nationalize with inside the Customer Bill. Although this default price approach won't be extensively used in the meantime the technology has been superior and is available for use if you need to without a doubt do now no longer use it. Nearby friends are utilized in element to locate credit score card styles using many new functions of curtain locking software applications use the KNN algorithm to analyze registered data and detect unusual patterns that place a special function. [5]

Eg. if the register data shows that the customer information is being manually entered in lieu of automatic scanning and exchange it means that in that case the staff were using the register. I have personal customer experience or if I register data showing good returns or exchanges in many instances this will imply that personnel are abusing the go back coverage or are looking to make money by making false returns. Therefore this is about the KNN algorithm. [5-6]

These visual and easy-to-understand decision-making trees are just a few examples of interpreting where you can better understand why a divider makes that decision allowing a given statistics set that this set of rules works higher than ours like ours. It relies upon at the statistics you've got on the usage of the hit and the trial technique and all of the algorithms separately and evaluate the end result. The version offers end result healthy as a version you could make to get higher accuracy of your statistics set. So what's a reducing tree?

"The decision tree is a strong illustration of all conceivable keys founded on situations."

Now why is this called a cutting tree?

It is so referred to as as it begins off evolved with an arrow and sprouts numerous answers like a tree or even a tree begins off evolved at the foundation and starts off evolved to develop its branches whilst it has eaten the larger and larger timber the evergreen tree and a developing quantity of selection and conditions. [7]

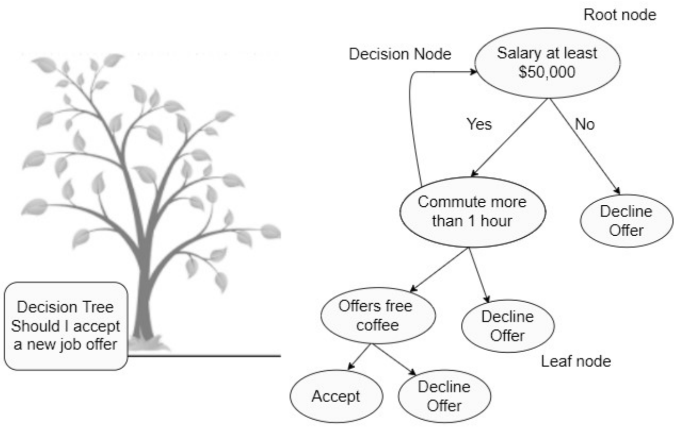


Figure 5: Decision Tree

"The own working of decision tree by taking one example is shown in Figure 5."Eg. bear in mind each time you dial to installation your credit score card employer quantity it takes you lower back to her clever laptop assistant while she requests you queries like media 1 to get English, press 2 to get the stop and Now one pick 1 and direct you to a selected set of questions along with press 1 in this situation press 2 for that to hold repeating whilst you attain the proper person. So the employer used a choice tree to take you to the proper person.

Just examine Figure 1.five in which the task is that I should receive a brand new task provide or now no longer so that you should determine whether or not with the aid of using developing a selection tree beginning with the simple shape or root node changed into the simple income or minimal salary must be. 50,000 if now no longer 50,000 way you do now no longer absolutely receive what's offered. So in case your income is greater than rs50,000 you'll additionally take a look at whether or not the go back and forth to paintings is greater than an hour or now no longer If it's far greater than an hour simply lessen the promise. If it's far much less than an hour you then definitely are near accepting the task. [8-9] then continue to test if the agency gives loose espresso or now no longer. If the agency does now no longer provide loose espresso you'll truly refuse the provide. And if the agency gives loose espresso you'll receive the provide that is simply an instance of a selection tree.

2. Understanding a Decision Tree

Here is a pattern facts set in order to be the usage of it to make the most their tree.

Table 1: Sample data set that will be using it to explore the decision tree

Colour	Diameter	Label
Green	3	Mango
Yellow	3	Mango
Red	1	Grapes
Red	1	Grapes
Yellow	3	Lemon

"**Table 1** displays the six sample dataset that will be using it to explore the decision tree. " Now records set every line is an instance and the primary columns offer capabilities, innovative descriptions and the ultimate column affords the marker or period we need to are expecting in case you want you may actually adjust this records with the aid of using including extra capabilities and greater. Our fashions and scheme will attempt the same way. Now, this records set is prematurely however for certainly considered one among them. The donkeys within side the 2d and 5th examples have comparable traits however the unique labels are each yellow in colour and as considerable as three. But the labels are mango and lemon. Let's see how the choice tree handles this example to create a tree with a view to use a choice tree set of rules known as cart set of rules. This cart represents a break up with a tree traction set of rules. Let's see first the way it works we are able to insert a tree root node and each root receives a listing of rows as established and the basis gets all of the education records set now. every node asks a ta rue query and a fake query approximately one issue and in solution to that query it's going to divide or separate the records accumulated into unique subsets of those subsets and grow to be an enter into the kid node that we upload to the tree and the intention of the query. is that you may ultimately blend the l. a. bel.as we pass down or in different phrases to supply a natural distribution of labels in every vicinity. There isn't any doubt approximately the sort of label because it best has graphs because it consists of best rapists. And so that it will do this we want to degree how plenty the query allows to put off the combination label and we are able to diploma the amount of uncertainty in a unmarried place using a metric called Gini pollution and we are able to diploma how lots the question reduces that uncertainty using the concept. Called information Gain will use the ones to select out the wonderful question to ask for each thing and what we are able to do will worsen the stairs we are able to again and again construct a tree in every new region will hold to break up the records till greater query is requested and ultimately attain our leaf position .so this turned into approximately the choice tree. [9].

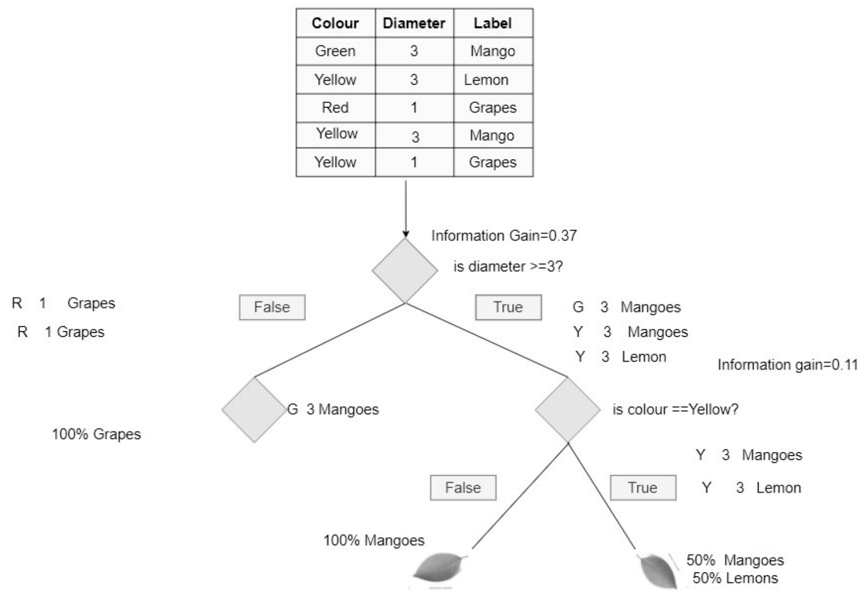


Figure 6: Decision tree identifying different sets

"The different datasets used to identify the decision tree is shown by **Figure 6.**"So with the intention to create a choice tree first we need, to perceive the specific units of questions you could ask the tree - like inexperienced.[6] And could you be asking if those questions can be decided with the aid of using your facts set consisting of this satiation is inexperienced in width extra than three is that yellow? Questions are just like your facts set. So if my satiation is inexperienced what it's going to do will cut up in first the inexperienced mango can be a part of the reality whilst the fake we have.

Figure 7: Decision tree identifying different sets of conditions [6]

Lemon and the mango.so if the color is inexperienced the diameter is extra than three or the color is yellow [9] "The decision tree identifying different sets of various conditions like based on the condition for mango its colour , shape and size is shown in Figure 7. "

3. Decision Tree Terminologies

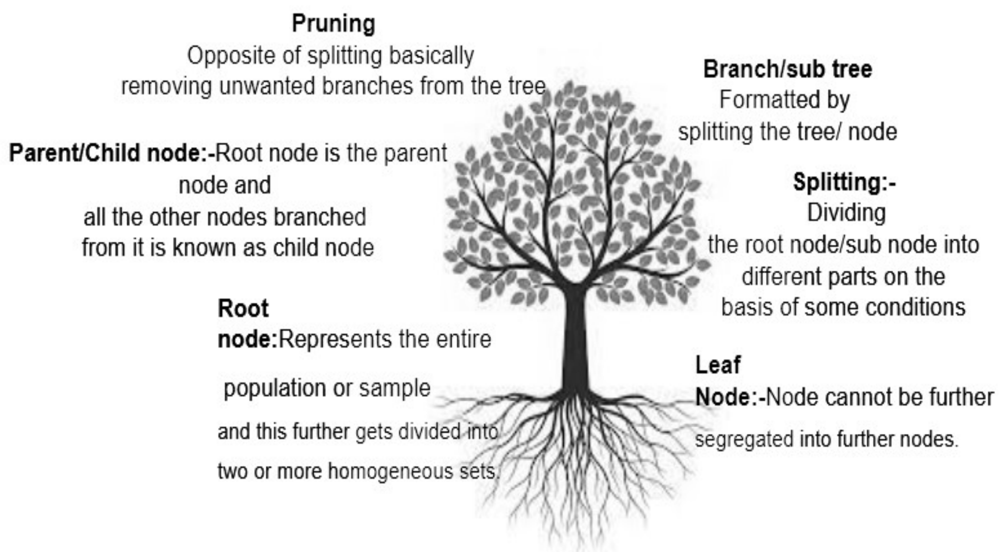


Figure 8: Decision tree terminologies

"The various terminologies of decision tree like root, leaf, splitting etc.are shown in Figure 8."

Root node: -The root node is the elementary node of a complete tree opening at the origin node. Signifies the entire people or sample and this lasts to be alienated into two or additional equal circles.
Leaf Node: -Node can't be reclassified. The leaf node is while you attain the cease of a tree which you cannot keep to serve one by one at another level.
Separation: -Differentiate to divide a root node / sub node into separate documents according to specific conditions.
Branch / Small Tree: -Designed by dividing the tree / area. This branch or substrate when dividing the tree proper when dividing the root node. Is alienated into two shares or two trivial plants
Pruning: -As opposite to trimming, it essentially eliminates unsolicited divisions after the tree.
Parent / Child Node: The root node is the figure node and altogether the extra brushwood because it are regarded as the kid node. We can recognize it in the sort of manner that each pinnacle node belongs to the figure node and the entire backside node. Found on the pinnacle node is a toddler node. The lump that plants extra node is the kid node and the basis node yields the kid node. [9- 10]

4. Cart Algorithm

Let's use a cart set of rules and layout a tree manually.
Step 1:-so at first we will decide which question to ask when to ask and when so how will you do that?

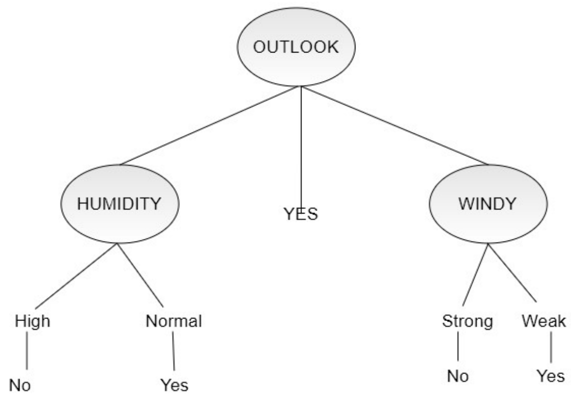


Figure 9: Example of decision tree [8]

"An explain of decision tree is given where outlook is either humid or windy depends on the condition is shown in **Figure 9**."

"**Table 2** displays a set of data that preserves the appearance, temperature, humidity, air, play based on your unique attribute and whether or not you could expect whether or not you could play or not so which one must you pick first? "

Table 2: Shows the data set which has a look, temp, humidity, windy, play as your different attribute-based

Outlook	Temp.	Humidity	Windy	Play
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
rainy	cool	normal	false	yes
rainy	cool	normal	true	no
overcast	cool	normal	true	yes
sunny	mild	high	false	no
sunny	cool	normal	false	yes
rainy	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
overcast	hot	normal	false	yes
rainy	mild	high	true	no

So first look at the decision tree here is you case decision tree that can be made in person Table 2 shows a set of data that preserves the appearance, temperature, humidity, air, play based on your unique attribute and whether or not you could expect whether or not you could play or not so which one must you pick first?

Find the attribute that best separates training data? But how can we choose the best quality? How does a tree determine its location?

So there are some words to understand before you can split a tree

- **Gini Index: -**The quantity of impurities (or purity) used with inside the production of the timber of the choice visible in CART with the aid of using the Gini Index. This Gini Indicator is a degree of Pollution or Hygiene used to construct a choice tree set of rules cart.
- **Information gain:** -Information benefit discounts entropy after the records set is constructed at the unruffled location of the power. Building a selection tree is set locating an characteristic that returns the very best stage of facts you benefit. So you may be taking a area that may stretch you the most Knowledge benefit.
- **Reduced Variation:** -The discount of variability is an set of rules used for non-stop centered variables (reversal problems) low variance variables are decided on as a determinant of populace division. Generally, the distinction is how exceptional your

facts is so in case your facts is grimy or easy if so the distinction might be as small as all of the facts are nearly the same.

- **Chi Square:** -It is a set of rules for locating the mathematical fee among the variations among sub-nodes and discern nodes.

Step 2: - How do you choose the best attribute? You essential to compute the gain of knowledge the quality with the maximum value of information is careful the top.

5. Entropy: -Entropy is a method of measuring NGS in other words or phrases, which can be calculated before solving a decision tree problem. Now it is common to understand what pollution is.

Suppose you've got were given have been given a basket complete of monkeys and every one of a kind bowl complete of the identical label now when you have requested to pick out one object for every basket and ball then the opportunities of having an apple and its accurate label are 1. So on this case, you may say that dust is 0. Now, what if there are 4 one-of-a-kind surrender result in a basket and 4 one-of-a-kind labels in a bowl. Then the opportunities of evaluating fruit with a label are slim. I can also furthermore have picked bananas withinside the basket as quickly as I determined directly to label the ball randomly. It says cherry any random modifications and combinations are possible. So on this case =! 0. [5-8]

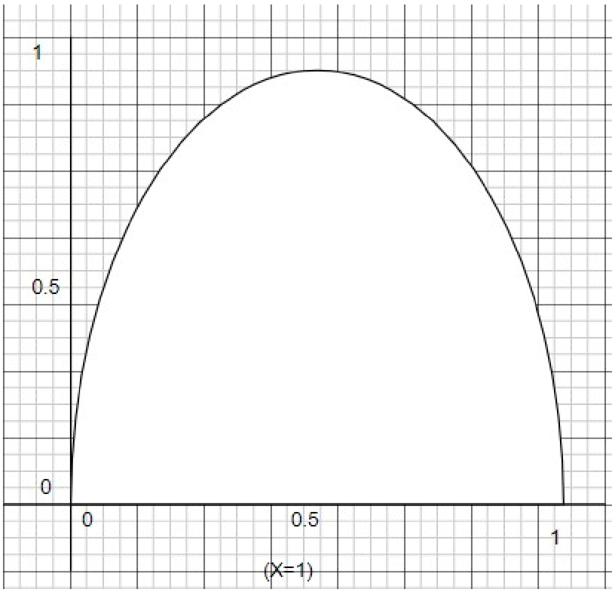


Figure 10: Entropy Graph

"The entropy x=1 amount of uncertainty with random outcome is shown in **Figure 10**." Entropy is a measure of contamination from a graph which is shown as a probability that 0 or 1 may be more polluted or cleaner than that. [10] Contamination is a random charge of random facts .so if the facts Is absolutely natural if so the randomness is equal to 0 or if the data is completely dirty then the pollution level is 0 .so the question is why the worth of entropy is advanced than 0.5?
So let's drive it mathematically

Therefore the graph is statistically proven in a single vicinity of our pattern and divided into components. Yes and no like in our records set the final results of the play is split into components sure or no we must expect whether or not we need to play or not. So in that case, we are able to outline an entropy formulation as an entropy.

The mathematical formula for entropy is: -

Entropy (s) = -P (yes) log2 P (yes) –P (no) log2 P (no)
There,
S is the sum of the sample space,
P (yes) chances are yes.
If the wide variety is yes = number P (S) = 0.5
Entropy = 1
If it contains all yes or all no ie P (S) = 1 or 0
Entropy (S) = 0
So let's look at the first scenario where the probability was 0.5
Entropy (s) = -P (yes) log2 P (yes)
When P (Yes) P (No) = 0.5 i.e. Yes + NO = Total sample (S)
So this is an entropy formula this is our first case
Where possibilities are sure = possibilities are no. that is, in our statistics set we've the same quantity of sure and no. consequently the chance of sure = chance once more and = 0.five.or can imply sure + and = Total pattern area. As the chance is 0.five so in case you set the values in a system you'll get some thing comparable below, and in case you calculate you'll get the entropy of the entire pattern area as 1.
E (Izi-) = 0.5 log2 0.5-0.5 log2 0.5
E (S) = 0.5 (log2 0.5-log2 0.5)
E (ZI) = 1
CASE 2 our next case is whether yes has a complete yes or no
E (S) = -P (Yes) log2 P (Yes)
When P (Yes) = 1 i.e. YES = Total sample (S)
So when you have a complete of sure let's examine the components if we've an entire sure, so we've each sure and zero no so chances are yes = 1and yes is the total space of the sample.
So in the formula when you set all the values you will get the entropy of the sample space as given below

E (S) = 1 log2 1
E (S) - 0
And as the value of log2 1is 0 so the total result will be 0
E (S) = -P (Cha) log2 P (Cha)
When P (No) = 1 i.e. No = Total Sample (S)
E (S) = 1 log2 1
E (S) - 0
The identical is real for zero even if so you'll get the entropy of the best pattern area as zero. so that is approximately Entropy
Gaining Knowledge
Measuring entropy reduction. Controls which feature would be designated as the decision bulge
If S is our complete collection,
Information Benefit = Entropy (S) - [(Weighted Avg.) X Entropy (per element)]

So let's count it by example and build our Decision Tree
Create a data decision tree provided below
"Table 3 displays the comparison of entropy for 14 datasets for outlook,Temp, Humidity,Windy and Play

Table 3: Compute the entropy for the entire data set

	Outlook	Temp.	Humidity	Windy	Play
D1	sunny	hot	high	false	no
D2	sunny	hot	high	true	no
D3	overcast	hot	high	false	yes
D4	rainy	mild	high	false	yes
D5	rainy	cool	normal	false	yes
D6	rainy	cool	normal	true	no
D7	overcast	cool	normal	true	yes
D8	sunny	mild	high	false	no
D9	sunny	cool	normal	false	yes
D10	rainy	mild	normal	false	yes
D11	sunny	mild	normal	true	yes
D12	overcast	mild	high	true	yes
D13	overcast	hot	normal	false	yes
D14	rainy	mild	high	true	no

"**Table 3** displays the compariosn of entropy for 14 datasets for outlook,Temp, Humidity,Windy and Play "

So that is our statistics set which includes 14 specific times out of which we've got nine sure and 5 no, so we've got a method for entropy so placing the values in it.

Step 1:-Compute the entropy for the entire data set
Out of 14 instances, we have 9 Yes and 5 No
So we have the formula,
E(s)=-P(yes) log2 P(yes) –P(no)log2 P(no)
E(S)= -(9/4)* log2 (9/14) –(5/14)* log2 (5/14)
E(S)=0.41+0.53=0.94

Step 2:-Which Node to Select as Root Node?
Now you need to pick out of outlook, temperature, humidity and windy which of the node you need to pick as the foundation node?

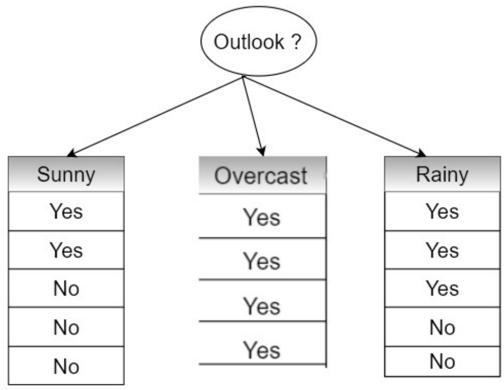


Figure 11: Calculates the Entropy and Information Gain

"The value for entropy and information gain is calculated in **Figure 11** whereas the concept has three different parameters as shown in **Figure 13" .**So you need to do it grade by grade you need to calculate the entropy and facts acquired from all of the one-of-a-kind nodes.[10]

So start with a vision. So Sunny, Overcast, and Rainy. so first choose how many yes and no there are in the solar system such as sunny how many yes and how many no numbers? So, in total, we have two yes and no three when it is sunny and when it covers we have yes. So if it is cloudy, we will definitely play. Next, it rains and the total value of yes is 3 and the total value of no is 2.

Next, we are able to calculate the entropy for each element that counts the entropy when Outlook is equal to the sun. First, we think that the idea is our root and as a result, we are counting the knowledge we have gained. So to calculate the Profit of Knowledge, we have a formula

We calculate the entropy of Outlook where the total number of yes while the sun was 2 and the total no was 3.
E (outlook = Sunny) = -2 / 5 log2 2 / 5-3 / 5 log2 3/5 = 0.971
Next, we count the cloudy entropy where the cloud was yes. So the probability of saying yes is equal to 1
E (view = cover) = -1 log2 1- 0 log2 0 = 0
And when it rains, it rains 3 yes and 2 no. therefore the probability of saying yes when it is sunny is 3/5 and the probability of saying no is 2/5.
E (outlook = Sunny) = -3 / 5 log2 3 / 5-2 / 5 log2 2/5 = 0.971
Now we have to calculate how much profit you get from watching it equal to the measured value
Gaining Knowledge
Me (view) = 5/14 * 0.971 + 4/14 * 0+ 5/14 * 0.971 = 0.693
So what changed into this estimate of the common price of sure and the full price of the co. so the records received from the view is identical to 5/14 from wherein does this 5? We calculate the full range of pattern areas inside that view whilst it's miles sunny. so with inside the case of the sun, there have been 2 sure and three no. so the common sun eclipse might be identical to 5/14.

Now calculate the value of the information to cover the sun, so we will equally calculate everything in relation to the sun. With a saved cover rate of 4/14 * entropy and combining all of these
Me (view) = 5/14 * 0.971 + 4/14 * 0+ 5/14 * 0.971 = 0.693
Now count the information gained by watching,
Profit (view) = E (S) -I (Outlook)
0.94-0.693 = 0.247

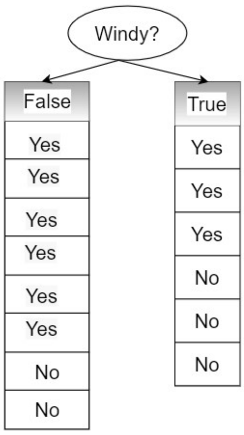


Figure 12: Calculate the Entropy for Windy [10]

"The value depend on the condition for decision tree for data windy gives value False or True is shown in **Figure 12."**

Now let us now imagine that the air is our root zone. so air contains two parameters True and False and how many and how many and how many yes if it happens true or false. therefore if there is a false positive in its parameter, if so, it has 6 yes and 2 no and if true as its parameter it has 3 yes and 3 no. therefore likewise count the information obtained from the wind and finally count the information obtained from the wind.

So first we will calculate the entropy of each element

So withinside the air condition, we've the identical variety of sure and no even as we've 0.five instances the sum of the cost of sure and the full cost of no. therefore, in that case, the entropy is identical to at least one in order that we will without delay input the fact whilst there's a wind like 1 and we've showed whilst the probability = 0.five entropy is a small cost identical to at least one

E (Spirit = True) = 1
Next is the entropy of fake if there's air so likewise simply positioned the possibilities of sure and no with inside the formulation and calculate the end result as we've six sure and a pair of no. in order an entire we get 6/eight sure and possibilities and no possibilities like.2/8. [10]

E (Wind = False) = 0.811
Therefore the complete information collected from the spirit is equal if the information obtained in the spirit is equal to true and false so we will calculate the approximate value of all and then summarize in order to obtain the complete information obtained from the spirit.

So, in this case, it is equal to 8/14 multiplied by 0.811 +6/14 multiplied by 1. here 8 is the sum of the number of yes and no in the case where the wind is equal to the false. therefore there was a false total value of 6 equal to the total value of 2 equal to 8. why rated estimates are the result of 8/14 equally taking information when the true equator is 3 + 3 i.e. 3 yes and 3 no 6 divided by the total value of the sample space i.e. 14 times 1 which is true entropy.

Information from windy

I (Windy) =8/14 * 0.811 +6/14 * 1=0.892
Now calculate information gained from windy
So total information gained from windy that equals total entropy –information gained from windy
Gain (Windy) =E(S)-I (Windy)
0.94-0.892=0.048
Similarly, we will calculate the rest of the two outlooks and temperature
Outlook:
Information from Outlook =0.693
The information gained from outlook=0.247
Temperature:
Information from Temperature =0.911
The information gained from Temperature =0.029
Humidity:
Information from Humidity =0.788
The information gained from Humidity =0.152

Windy:
Information from Windy =0.892
The information gained from Windy =0.048
Since Max gain =0.247
Outlook is our ROOT Node.

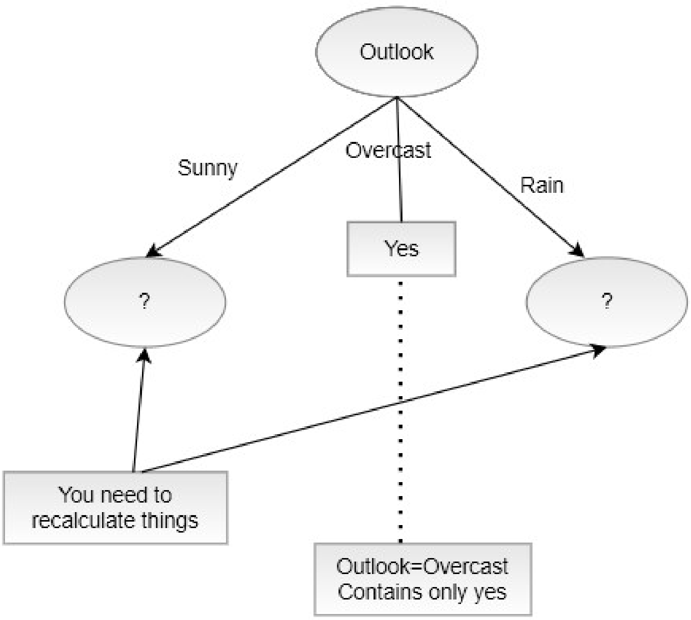


Figure 13: Calculates the Entropy for Outline Look

"Now we've decided on Outlook as our root node and it's miles similarly divided into 3 elements sunny, over, sky and rain in a variable kingdom we've visible that it consists of everything, sure to examine it as a leaf node is shown in **Figure 13"**

But with inside the case of solar and rain, it's miles dubious because it consists of each sure and no. so that you want to re-depend an object and on this node you need to re-depend items.

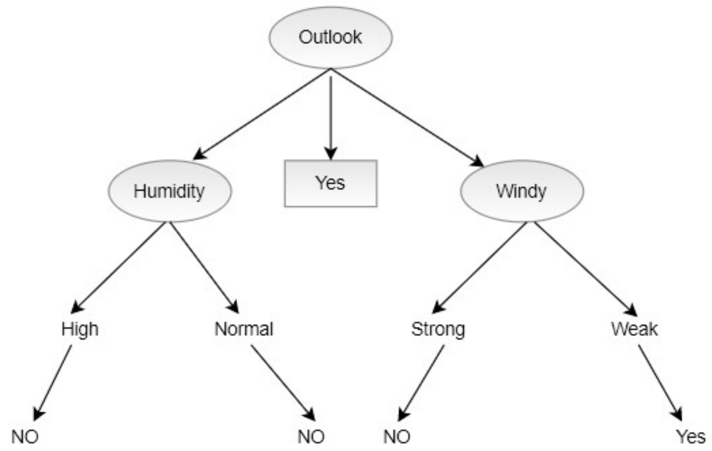


Figure 14: Calculates the Entropy for the Outlook [10]

"When a imaginative and prescient predicts it'll rain you may additionally take a look at whether or not it's miles windy or now no longer is shown in **Figure 14"**

So what is going to this ideal tree of yours appearance like? so let's see while you could play so that you can play while the view is overcast. Therefore, in that case, you could usually play. If the humidity is ordinary you could play, if the humidity is excessive you may now no longer play. When a imaginative and prescient predicts it'll rain you may additionally take a look at whether or not it's miles windy or now no longer. If it's miles a susceptible wind you may cross and provide the sport however when you have a sturdy wind you may now no longer play so that is what your choice tree can appearance like.

V. PRUNING

Does it say what I should do to play? There you have to make trees. Pruning will determine how you play.

"The decision tree is a clear illustration of all possible solutions based on specific circumstances"

Pruning is nothing but cutting nodes to find the right solutions. so pruning minimizes complexity.[10]

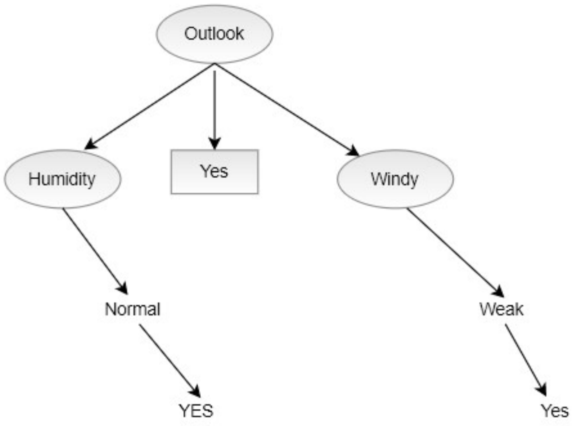


Figure 15: shows the results for the outlook [10]

"The result for the outlook depend on the condition will select humidity or windy is shown in Figure 15 which shows all the yes results you can play. "

Are tree-based simulations better than straight-line simulations?

So here you can think of a link if I use a regression factor for split problems and line back with regression problems. Then why is there a essential to use the medication? You can use any set of rules depending on the form of problem. Let's resolve a number of the critical troubles to help you determine which set of rules to apply and when.

1. If the association among dependent and independent variables is fine unhurried by the line model then the line aberration will overdo the tree-based perfect.
2. If there may be a non-linear mild and a complicated dating among based and impartial versions the tree version will surpass the antique retrospective version.
3. If you essential to figure an easy-to-explain perfect for persons the result tree model will continually do improved than the line model as the decision tree replicas are easier to understand than line support.

VI. WRITING A DECISION TREE CLASSIFIER IN PYTHON USING CART ALGORITHM

So here I initialize our training data set. This is our sample dataset for which each row is an example.

```
# Sample dataset.
# Format: each row is an example.
# The last column is the label.
# The first two columns are features.
# If you want you can add more features & examples.
# Interesting note: 2nd and 5th examples have the same features, but different labels -
# let's see how tree handles this case.

training_data = [
    ['Green', 3, 'Mango'],
    ['Yellow', 3, 'Mango'],
    ['Red', 1, 'Grape'],
    ['Red', 1, 'Grape'],
    ['Yellow', 3, 'Lemon'],
]

# Column labels.
# These are used only to print the tree.
header = ["color", "diameter", "label"]
```

Figure 16: Initialize our training data set

"The inilization of training dataset is shown in Figure 16. and its further implementation is shown in Figure 17"

The ultimate column is label and the primary columns are elements. If you need you could upload a few features .so let's have a look at how the tree handles this case. Both columns and the 5th column have comparable features. Next, upload column labels. So now we are going to upload titles to the columns because the first column is one 2d extensive and the 0.33 is a label column. So what we can do is outline characteristic as exceptional values to be able to exceed traces and columns. So this characteristic will begin with the aid of using locating exceptional column values withinside the database for an instance of this. So right here we transmit schooling facts as our rows and column wide variety as 0. So we get specific values in color. and in this, as the road is schooling facts and column 1 we get specific values in phrases of scope.

```
def unique_vals(rows, col):
    """Find the unique values for a column in a dataset."""
    return set([row[col] for row in rows])

#######
# Demo:
# unique_vals(training_data, 0)
```

Figure 17: Initialize our training data set

"The inilization of training dataset is shown in Figure 16. and its further implementation is shown in Figure 17"

We will outline a characteristic as a category calculation, and we can by skip traces on it for what we do. Remember the wide variety of every version kind withinside the information set so in this situation we remember the wide variety of every kind for instance withinside the information. set or what we can we calculate distinct values according to label withinside the database as sample.

```
def class_counts(rows):
    """Counts the number of each type of example in a dataset."""
    counts = {}  # a dictionary of label -> count.
    for row in rows:
        # in our dataset format, the label is always the last column
        label = row[-1]
        if label not in counts:
            counts[label] = 0
        counts[label] += 1
    return counts
```

Figure 18: Passing training data as our rows and column

"The implementation of passing rows and columns values to our training dataset is shown in Figure 18."As a sample you can see here we can transfer all of the training data set to this specific function such as class_count.

```
    counts = {}  # a dictionary of label -> count.
    for row in rows:
        # in our dataset format, the label is always the last column
        label = row[-1]
        if label not in counts:
            counts[label] = 0
        counts[label] += 1
    return counts

#######
# Demo:
# class_counts(training_data)
#######


def is_numeric(value):
    """Test if a value is numeric."""
    return isinstance(value, int) or isinstance(value, float)
```

Figure 19: Passing that entire training data set to this particular function as class count

"The implementation of entire training data set to this particular function as class count is shown in Figure 19 "Next, we will define a function as a number and transfer the value to it. For example, you can see that we are more than 7 so it is a whole number so it will reverse the numerical value and if we exceed read it is not a numerical value.

```
def is_numeric(value):
    """Test if a value is numeric."""
    return isinstance(value, int) or isinstance(value, float)

#######
# Demo:
# is_numeric(7)
# is_numeric("Red")
#######
```

Figure 20: Defining a function as numeric

"The implementation of is_numeric function and passing value as an argument is shown in Figure 20." We will now outline the magnificence question in order that this may be used to split the facts set. Will this magnificence simply report a column number? Example zero of the satiation mild and the price of the column as an instance in inexperienced subsequent describes the matching approach used to examine the detail price withinside the version with the detail values saved in its question.

```
class Question:
    """A Question is used to partition a dataset.

    This class just records a 'column number' (e.g., 0 for Color) and a
    'column value' (e.g., Green). The 'match' method is used to compare
    the feature value in an example to the feature value stored in the
    question. See the demo below.
    """

    def __init__(self, column, value):
        self.column = column
        self.value = value

    def match(self, example):
        # Compare the feature value in an example to the
        # feature value in this question.
        val = example[self.column]
```

Figure 21: Defining a class

"The implementation of defining a class names Question is shown in Figure 21."

```
def match(self, example):
    # Compare the feature value in an example to the
    # feature value in this question.
    val = example[self.column]
    if is_numeric(val):
        return val >= self.value
    else:
        return val == self.value

def __repr__(self):
    # This is just a helper method to print
    # the question in a readable format.
    condition = "=="
    if is_numeric(self.value):
        condition = ">="
    return "Is %s %s %s?" % (
        header[self.column], condition, str(self.value))
```

Figure 22: Defining an init function

"The implementation of defining init function names match and _repr_ and assign values in it is shown in Figure 22." We outline the init characteristic and internally we byskip the column itself and the cost so subsequent we outline the characteristic because the analogy will evaluate the detail cost as An example with the element rate in this question. Next, we're capable of define a characteristic as a correction that is honestly a helper's way of printing a question withinside the can format the following time we do it defining the branch of labor. However, this selection is used to break up the facts set for each line with within the facts set that you take a look at to look if it looks as if a query or now no longer in case you do upload it to the Truth traces or if now no longer then upload to the fake traces.

```
def partition(rows, question):
    """Partitions a dataset.

    For each row in the dataset, check if it matches the question. If
    so, add it to 'true rows', otherwise, add it to 'false rows'.
    """
    true_rows, false_rows = [], []
    for row in rows:
        if question.match(row):
            true_rows.append(row)
        else:
            false_rows.append(row)
    return true_rows, false_rows
```

Figure 23: Defining a partition function

"The implementation of partition function and passing value like rows, question as an augment is shown in Figure 23." When we divide the information primarily based totally on whether or not the traces are purple or now no longer right here we name a characteristic question and by skip a price to 0. so what it does will provide all of the purple traces to the True traces and the whole lot else may be assigned to the fake traces. Next, we can describe Gini's contaminant pastime and inside that, we can undergo the listing of traces.

```
# true_rows, false_rows = partition(training_data, Question(0, 'Red'))
# This will contain all the 'Red' rows.
# true_rows
# This will contain everything else.
# false_rows
#######

def gini(rows):
    """Calculate the Gini Impurity for a list of rows."""

    counts = class_counts(rows)
    impurity = 1
    for lbl in counts:
        prob_of_lbl = counts[lbl] / float(len(rows))
        impurity -= prob_of_lbl**2
    return impurity
```

Figure 24: Defining Gini function

" The implementation of Gini function and passing number of rows an augment is shown in Figure 24." So this will calculate the Gini impurity for the rest of the rows.

```
def info_gain(left, right, current_uncertainty):
    """Information Gain.

    The uncertainty of the starting node, minus the weighted impurity of
    two child nodes.
    """

    p = float(len(left)) / (len(left) + len(right))
    return current_uncertainty - p * gini(left) - (1 - p) * gini(right)
```

Figure 25: Calculate the Gini impurity

"The implementation of info_gain function and passing left,right,current_uncertaininty as an argument is shown in Figure 25." Next, we can describe the feature info_gain.so will calculate the advantage of data the usage of the first-diploma uncertainty — the burden of the kid node. The subsequent assignment is to locate the exceptional location this activity is used to locate the exceptional query you may ask time and again with the aid of using all factors of pricing and calculate the profitability of the data.


```
def find_best_split(rows):
    """Find the best question to ask by iterating over every feature / value
    and calculating the information gain."""
    best_gain = 0  # keep track of the best information gain
    best_question = None  # keep train of the feature / value that produced it
    current_uncertainty = gini(rows)
    n_features = len(rows[0]) - 1  # number of columns

    for col in range(n_features):  # for each feature

        values = set([row[col] for row in rows])  # unique values in the column

        for val in values:  # for each value

            question = Question(col, val)

            # try splitting the dataset
            true_rows, false_rows = partition(rows, question)
```

Figure 26: Define a functionfind_ best_split

"The implementation of find_best_split function and passing rows as an argument is shown in Figure 26." Next, we can outline a category as depart and used it for classifying the data. It holds a dictionary of instructions like mango for a way usually it seems withinside the row from the education data

```
class Leaf:
    """A Leaf node classifies data.

    This holds a dictionary of class (e.g., "Mango") -> number of times
    it appears in the rows from the training data that reach this leaf.
    """

    def __init__(self, rows):
        self.predictions = class_counts(rows)


class Decision_Node:
    """A Decision Node asks a question.

    This holds a reference to the question, and to the two child nodes.
    """
```

Figure 27: Define a class as leave

"The implementation of class nmae as leaf and defining init function passing rows as an argument is shown in Figure 27." Next is the choice node so this choice node will ask a query. This holds a connection with the query and the two-infant nodes on the bottom of it. You are identifying which node to feature in addition to which branch.

```
class Decision_Node:
    """A Decision Node asks a question.

    This holds a reference to the question, and to the two child nodes.
    """

    def __init__(self,
                 question,
                 true_branch,
                 false_branch):
        self.question = question
        self.true_branch = true_branch
        self.false_branch = false_branch


def build_tree(rows):
    """Builds the tree.
```

Figure 28: Define a decision node

"The implementation of defining decision tree by creating a class named it as Decision_Node is shown in Figure 28." Next, we're defining a characteristic of the construct tree that we're passing numerous rows so that is the characteristic this is used to construct the tree.

VII. CONCLUSION

The decision tree process is a leading digital separation tool to predict data fraud, interpretation, and classification is a major difficulty in data mining. Although segregation has been deliberate for a long time withinside the past, many proposed separation strategies do now no longer paintings in instances wherein records is immediately separated one piece is thought through one group, every other piece through every other assembly, and no collection is required to express their privacy. In this chapter, we illuminate the decision tree with various examples and use a training data set using the CART algorithm dividing the decision tree by python.

REFERENCES

[1] F. Mola, "Classification and Regression Trees Software and New Developments," pp. 311–318, 1998, doi: 10.1007/978-3-642-72253- 0_42.
[2] W. Du and Z. Zhan, "Building Decision Tree Classifier on Private Data," in Proceedings of the IEEE International Conference on Privacy, Security and Data Mining - Volume 14, 2002, pp. 1–8.
[3] R. Hull, M. Benedikt, V. Christophides, and J. Su, "E-Services: A Look behind the Curtain," in Proceedings of the Twenty-Second ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, 2003, pp. 1–14. doi: 10.1145/773153.773154.
[4] M. Anyanwu and S. Shiva, "Comparative Analysis of Serial Decision Tree Classification Algorithms," Int. J. Comput. Sci. Secur., vol. 3(3), 2009.
[5] P. E. Utgoff, N. C. Berkman, and J. A. Clouse, "Decision Tree Induction Based on Efficient Tree Restructuring," Mach. Learn., vol. 29, no. 1, pp. 5–44, 1997, doi: 10.1023/A:1007413323501.
[6] J. R. B. Cockett and J. A. Herrera, "Decision Tree Reduction," J. ACM, vol. 37, no. 4, pp. 815–842, Oct. 1990, doi: 10.1145/96559.96576.
[7] U. M. Fayyad and K. B. Irani, "On the handling of continuous-valued attributes in decision tree generation," Mach. Learn., vol. 8, no. 1, pp. 87–102, 1992, doi: 10.1007/BF00994007.

[8] A. Navada, A. N. Ansari, S. Patil, and B. A. Sonkamble, "Overview of use of decision tree algorithms in machine learning," in 2011 IEEE Control and System Graduate Research Colloquium, 2011, pp. 37–42. doi: 10.1109/ICSGRC.2011.5991826.
[9] M. Zorman, P. Kokol, M. Molan Stiglic, and A. Gregoric, "Are decision trees way around some statistic methods?," vol. 20, no. 3, pp. 1198–1201, 2002, doi: 10.1109/iembs.1998.747087.
[10] R. Samson, "Decision Making and Problem Solving," Leadersh. Manag. Nurs. Pract. Educ., no. September 2015, pp. 59–59, 2009, doi: 10.5005/jp/books/10444_10.