Decision Tree

Agenda/Session Outcomes

Decision Tree

At the end of this session, you will be able to:

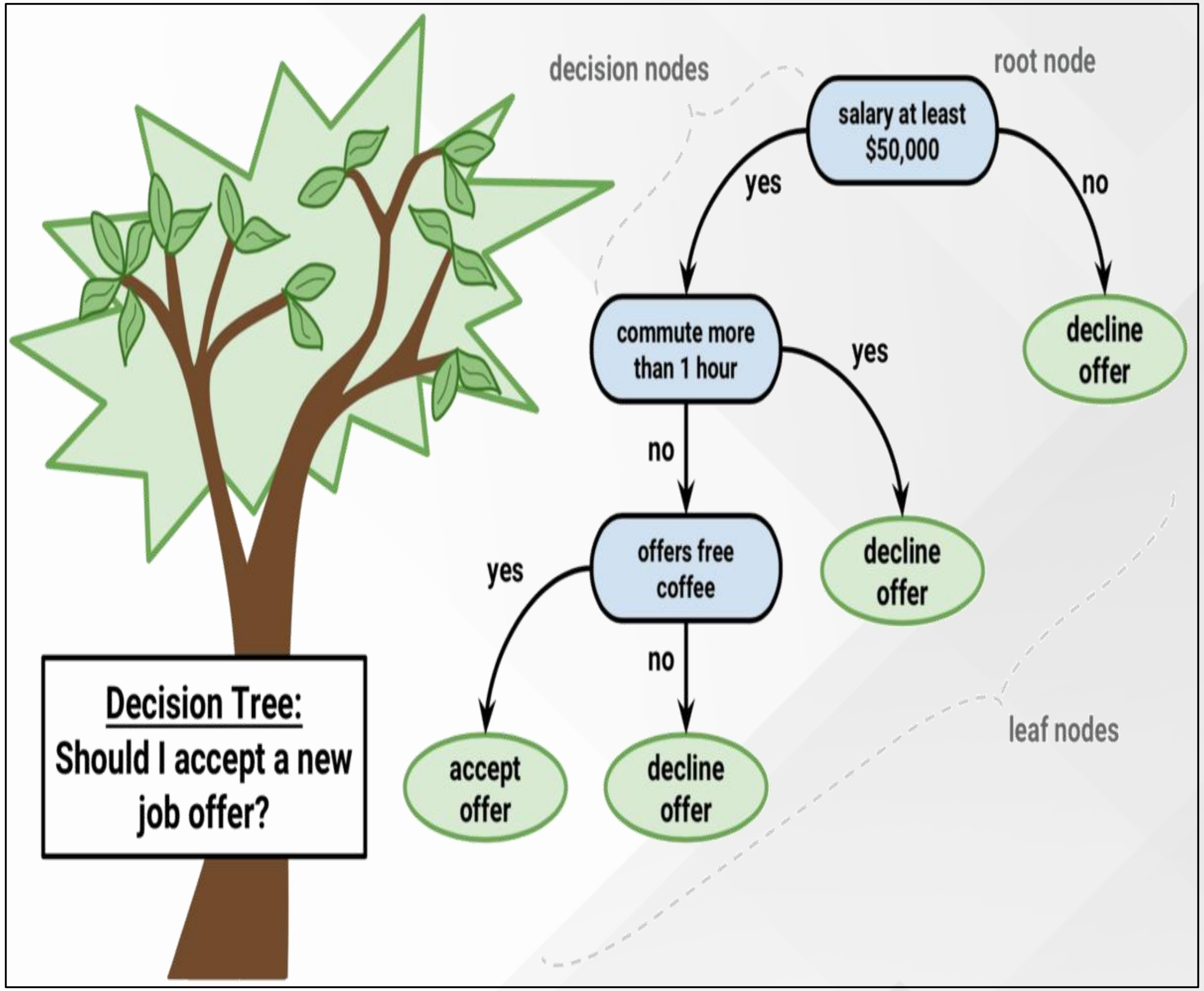
1. Evaluate the concept of decision trees and their usage for solving regression and classification problems.
2. Assess the concepts that make up the decision tree work, like Gini entropy, information gain, etc.

Applications of Decision Tree

- **Customer Relationship Management:** To classify customers' online behavior into specific groups.
- **Fraudulent Statement Decisions:** Auto-classification of banking transactions if they are fraudulent
- **Energy Consumption:** How much energy will be consumed can be predicted in the next billing cycle.
- **Healthcare Management:** In medical diagnosis, such as classifying patients as diabetic or not, or whether they have non-pneumonia or pneumonia, healthcare management is used.

Decision Tree

A **decision tree** is a supervised learning algorithm in machine learning used to categorize or make predictions based on the previous set of rules.



Parametric vs. Non-parametric Algorithms

A **decision tree** is a non-parametric algorithm. Let's understand what parametric and non-parametric algorithms are.

Parametric Algorithms

Parametric algorithms are based on a mathematical model defining a relationship between input and output variables.

Example of parametric algorithms

- Linear regression
- Logistic regression
- Neural network

Non-parametric Algorithms

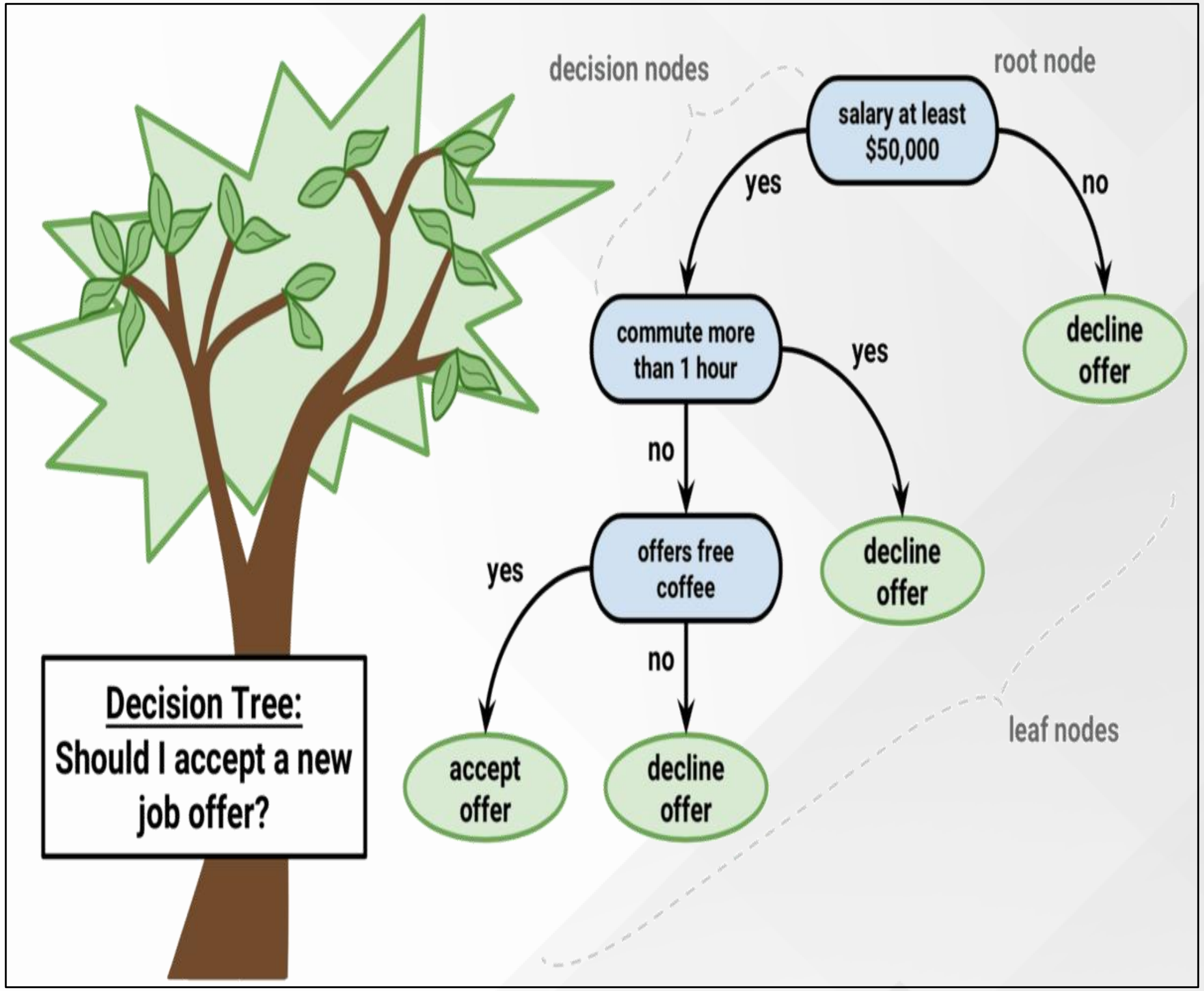
The non-Parametric algorithm tries to learn from the data itself and its patterns. They are computationally expensive.

Example of non-parametric algorithms

- Decision tree
- KNN
- Support vector machine
- Other ensemble models like (xgboost/adaboost/etc.)

Decision Tree

- Decision tree provides a way to present an algorithm with conditional control statements.
- Decision trees are more effective in handling nonlinear data sets effectively.
- There are two decision tree types: Categorical variable and Continuous variable decision tree.



Types of Decision Tree

Categorical Variable Decision Tree

A categorical variable decision tree will have the target variable as a category. The categories can be **Binary** or **Multiclass**.

Example:

Yes/No, High/Medium/Low, Non-Pneumonia/Pneumonia

Continuous Variable Decision Tree

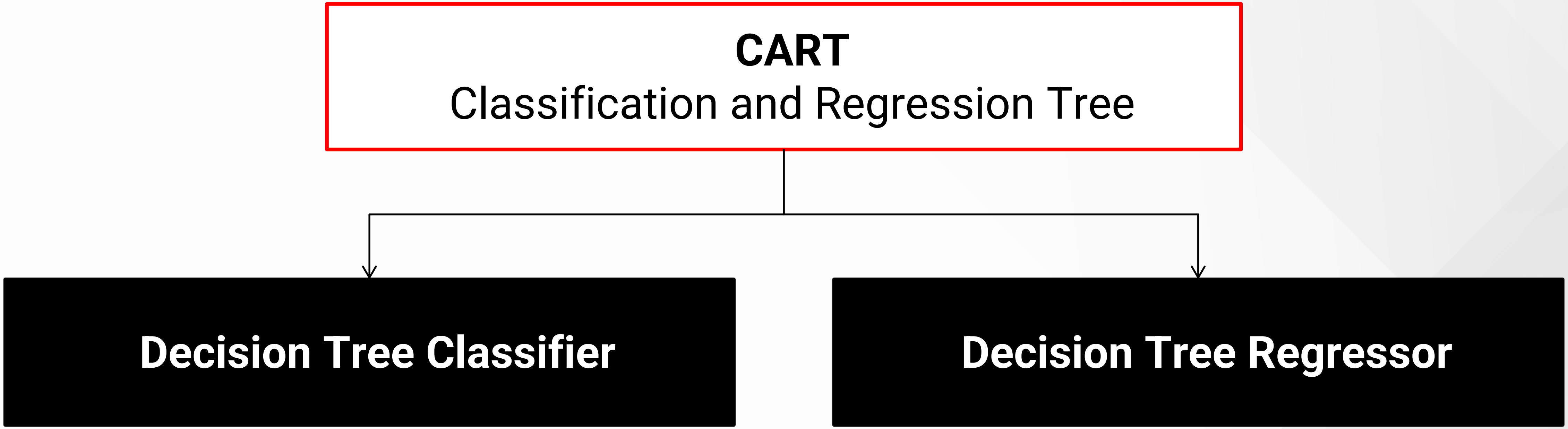
A continuous variable decision tree will have the target variable as **Continuous**.

Example:

Salary Prediction, Demand Forecasting, Stock Prediction.

Hero

CART



CART

Classification and Regression Tree

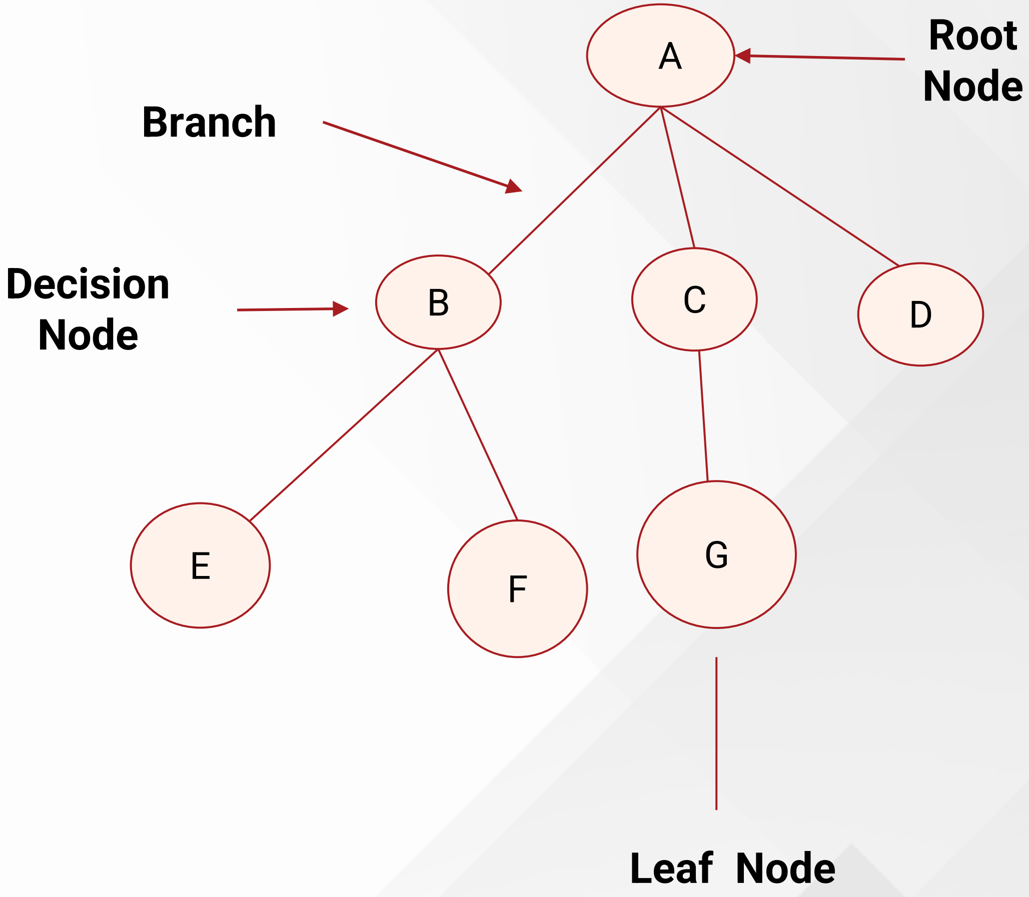
Decision Tree Classifier

Decision Tree Regressor

Structure of Tree

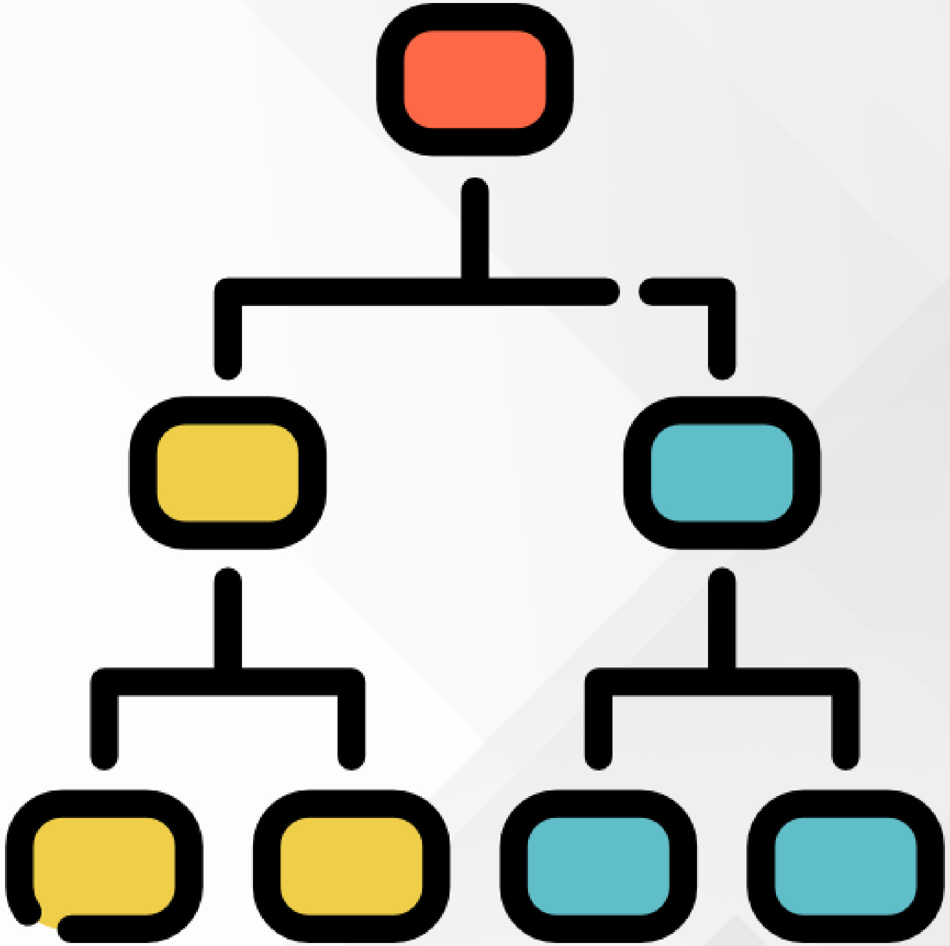
Before we learn about the decision tree algorithm, let’s first understand the tree structure.

- A tree has nodes & branches.
- Branches connect nodes.
- There is a parent-child relationship between the nodes connected with a branch. (node c is the child of node a)
- Depth of a node is defined as the number of branches from the node to the root node (depth of node g is 2).
- A node that has no child is called a leaf node. (Nodes e, f, g, and d are leaf nodes)



Make Decisions Using Tree

- Logic behind the decision tree is to make decisions and continually split the dataset until **Homogeneity**.
- For each question, we will add a node in the tree; the first node is called the **root node**.
- Asking questions based on the feature splits the dataset resulting in a tree.



Terminologies

- **Entropy**: Entropy can be defined as a measure of sub-split purity in machine learning. Entropy always lies between 0 and 1. The entropy of any partition can be calculated as follows:

$$H(s) = -P_{(} + \log_2 P_{(}+) - P_{(}-) \log_2 P_{(-})$$

$$\text{Here } P_{(}+) \,/\, P_{(-)} = \%\text{ of } + \text{ ve class } 1\%\text{ of - ve class}$$

- **Gini Impurity**: Gini Impurity is somewhat like entropy in internal working. Both are used for building the tree using the best split. But there is quite a difference in computational methods.

$$GI = 1 - \sum_{i=1}^{n} (p)^2$$

$$GI = 1 - \left[(P_{(}+))^2 + (P_{(}-))^2 \right]$$

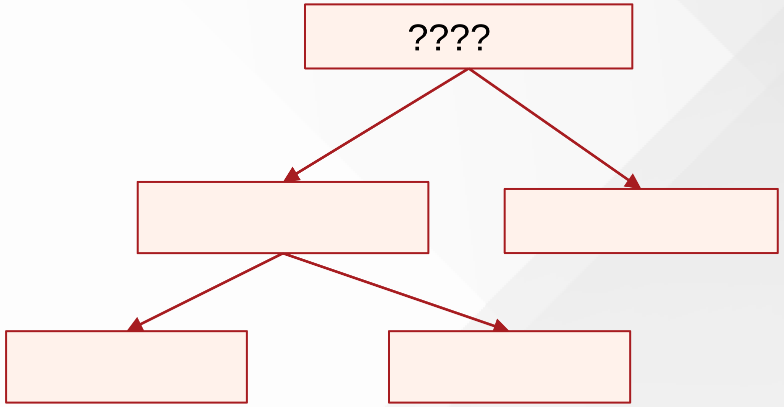
- **Information Gain**: Information gain tells us the best feature to split or the best feature to use as a root node for constructing a tree and then gradually creates an internal node and leaf node with each recursive split.

Let’s Implement a Decision Tree Classifier - 1

We will implement a decision tree using a small dataset, which has four features: loves popcorn, loves soda, age, and happy person, and remember the happy person is the target element for the dataset.

Loves Popcorn	Loves Soda	Age	Happy Person
Yes	Yes	7	No
yes	No	12	No
No	yes	18	yes
No	yes	35	yes
yes	yes	38	yes
yes	No	50	No
No	No	83	No

Step 1: Select the feature that will be the root node.

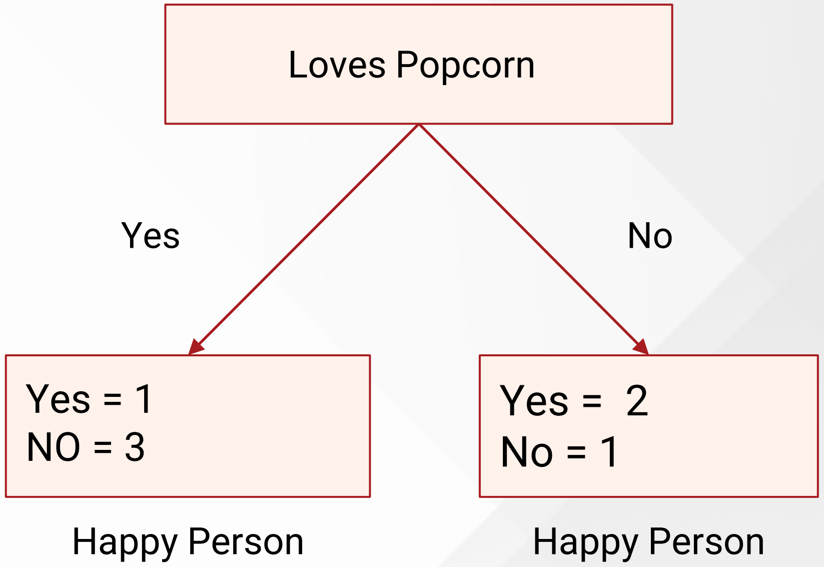


Let’s Implement a Decision Tree Classifier - 2

We will repeat the same process and fill all the leaf nodes recursively.

Loves Popcorn	Loves Soda	Age	**Happy Person**
Yes	Yes	7	No
yes	No	12	No
No	yes	18	yes
No	yes	35	yes
yes	yes	38	yes
yes	No	50	No
No	No	83	No

Select the feature that loves popcorn.

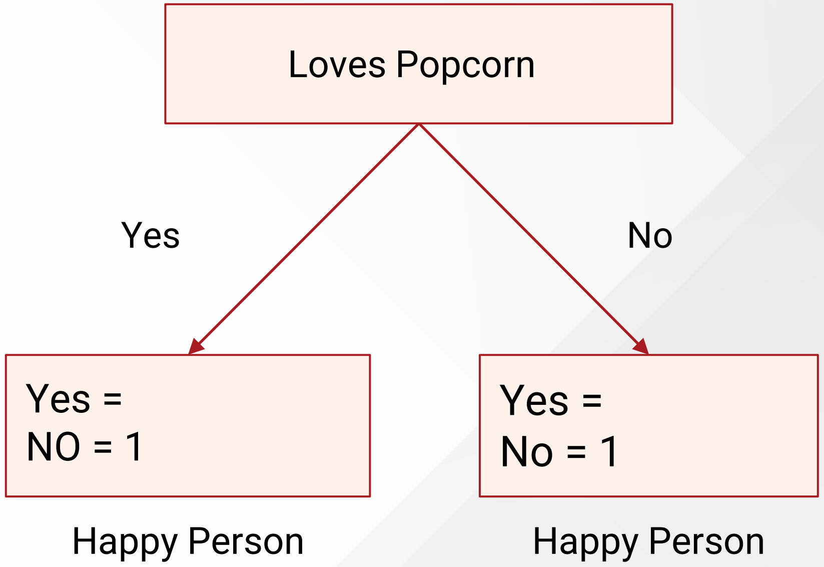


Let’s Implement a Decision Tree Classifier - 3

We will do the same thing for the Loves Soda we did for Loves Popcorn. And we will build a simple tree given below.

Loves Popcorn	**Loves Soda**	Age	**Happy Person**
Yes	Yes	7	No
yes	No	12	No
No	yes	18	yes
No	yes	35	yes
yes	yes	38	yes
yes	No	50	No
No	No	83	No

Select the feature that loves popcorn.



Loves Popcorn

Yes

No

Yes =
NO = 1

Yes =
No = 1

Happy Person

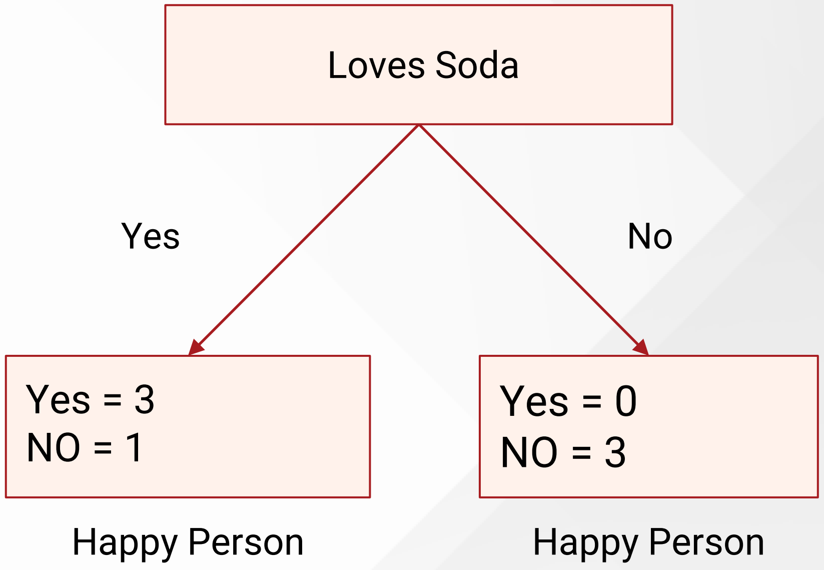
Happy Person

Let’s Implement a Decision Tree Classifier - 4

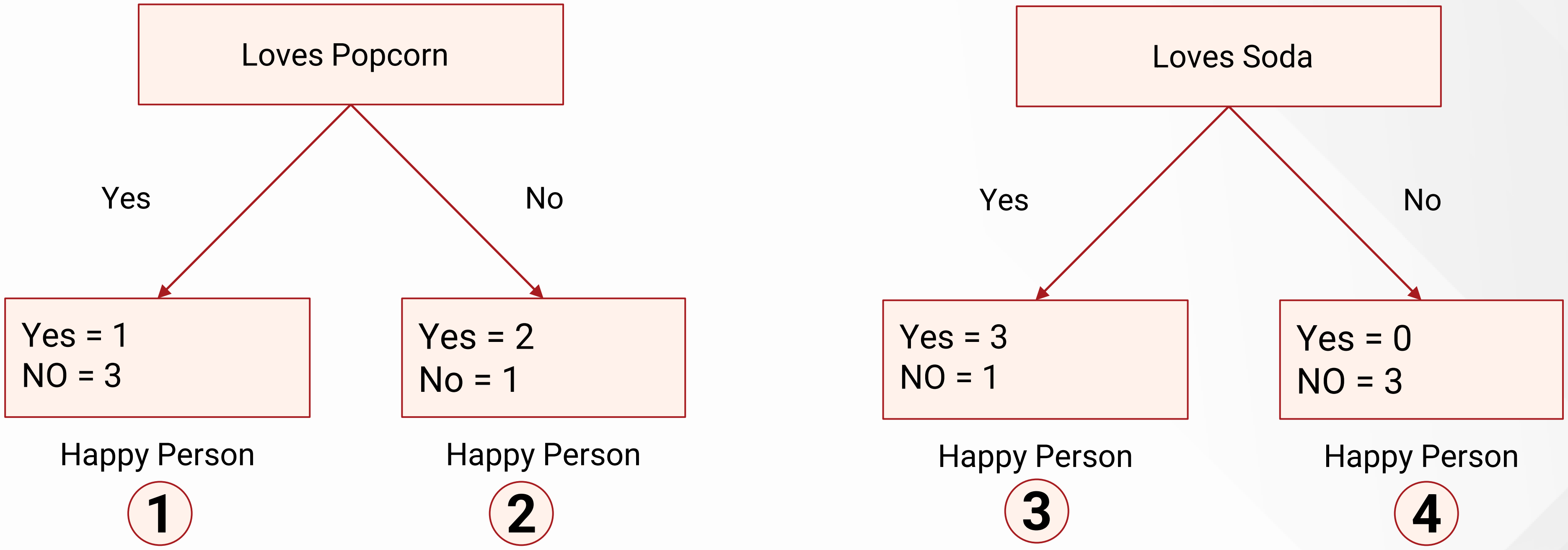
After we finish the rows, the simple tree below looks like when we have the Loves Soda attribute as a root node.

Loves Popcorn	**Loves Soda**	Age	**Happy Person**
Yes	Yes	7	No
yes	No	12	No
No	yes	18	yes
No	yes	35	yes
yes	yes	38	yes
yes	No	50	No
No	No	83	No

Now we will select the feature Loves Soda



Let’s Implement a Decision Tree Classifier - 5

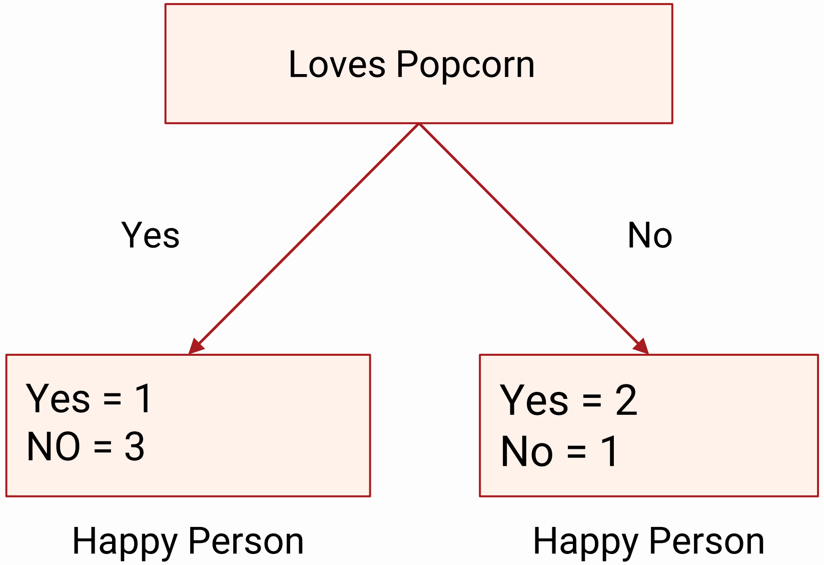


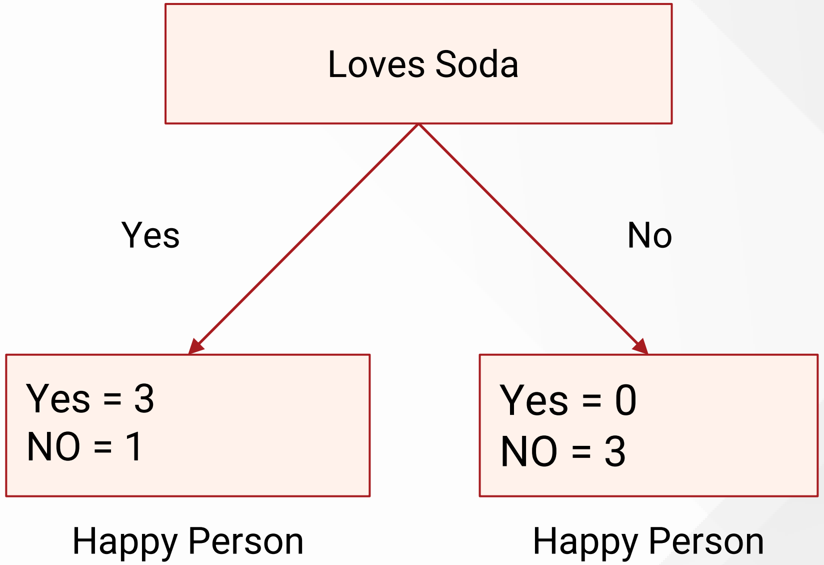
Note: Both the tree where the root node is Loves Popcorn and Loves Soda. They need to classify better who will be the happy person and who will not.

Leaf Nodes one, two, and three have little purity and are heterogeneous, whereas **Leaf Node four** has the purity. We aim **to remove heterogeneity and keep homogeneity.**

Let’s Implement a Decision Tree Classifier - 6

From the previous comparison between two simple trees. The Loves Soda tree is better at predicting whether a person will be happy. The impurity is in only one leaf, but we can see impurity in the Loves popcorn tree in both leaves.



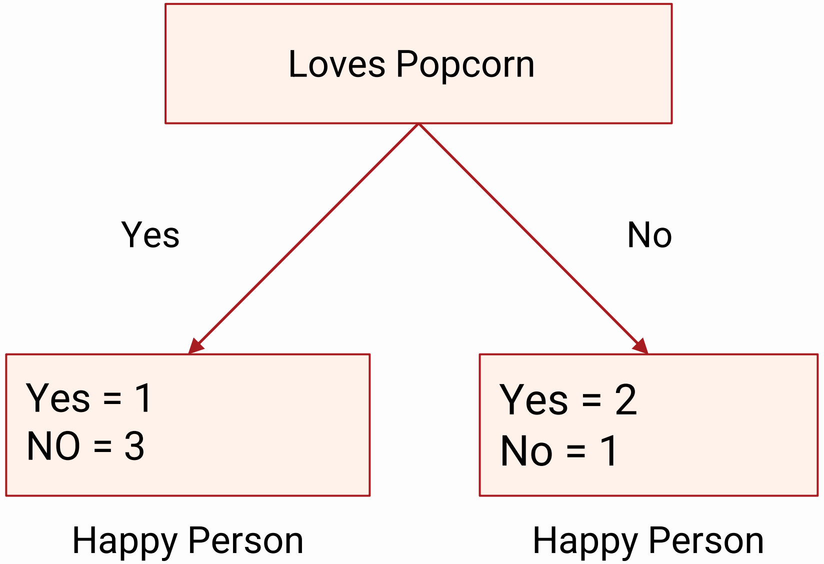


Good News is that there is always a way to measure/quantify the impurity in the leaves. Some methods are Entropy and Gini Impurity. Methodically they are identical, but they do differ in formulas.

Let's Implement a Decision Tree Classifier - 7

In this session, we will use Gini Impurity to understand how to calculate by formula. But we will also give the resources for Entropy as a post-read.

We will start calculating the Gini impurity for all the individual leaves for the Loves popcorn tree.



Gini impurity of left leaf = 1-(probability of "yes")²-(probability of "No")²

= 1-(1/1+3)²-(3/1+3)²

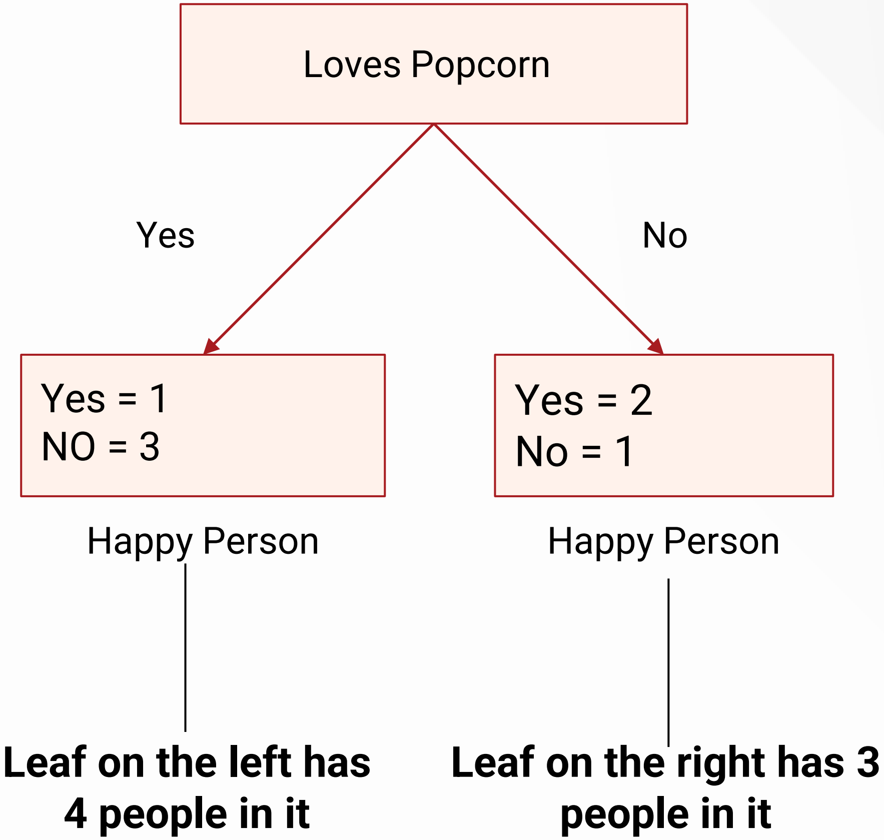
= 0.375

Gini impurity of right leaf = 1-(probability of "yes")²-(probability of "No")²

= 1- (2/2+1)²-(1/1+2)

= 0.444

Let’s Implement a Decision Tree Classifier - 8



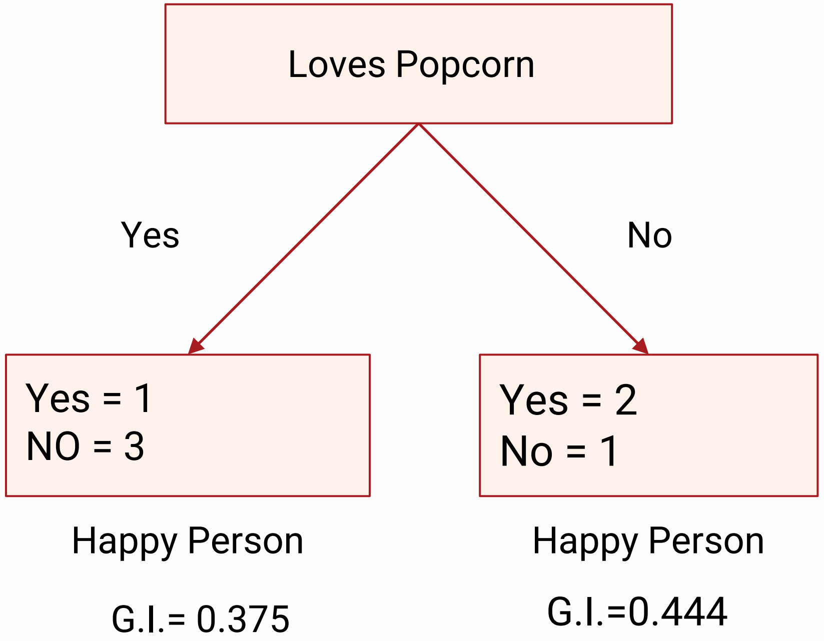
Leaf on the left has 4 people in it

Leaf on the right has 3 people in it

As we can observe, leaves do not represent the same number of people. Thus, to know the information gained to decide whether this feature is best for the root node, we must compute the weighted average of the Gini impurities for the leaves.

Let’s Implement a Decision Tree Classifier - 9

Thus, the total Gini Impurity for all the leaves for:



Loves Popcorn

Yes

No

Yes = 1
NO = 3

Yes = 2
No = 1

Happy Person

G.I.= 0.375

Happy Person

G.I.=0.444

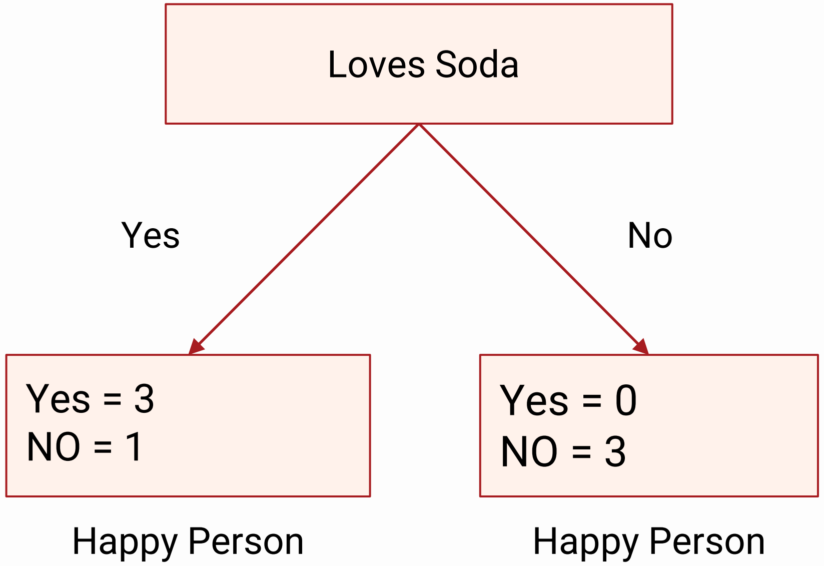
Total Gini impurity = Weighted average for Gini Impurities for the leaves.

Total G.I. = Weighted gini avg for left leaf + weighted gini avg for right leaf

= (4/4+3) *0.375 + (3 / 4 + 3) *0.444
= 0.405

Let's Implement a Decision Tree Classifier - 10

We will use the same method to compute the weighted gini avg for the entire leaves for **Love Soda.**



Total Gini impurity = Weighted average for Gini Impurities for the leaves.

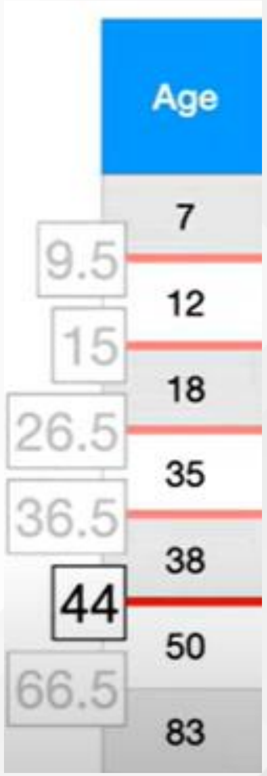
Total G.I. = Weighted gini avg for left leaf + weighted gini avg for right leaf

= 0.214

Let’s Implement a Decision Tree Classifier - 11

However, Age Contains numeric data, not just Yes/No values. Calculating the Gini Impurity is a little more involved. First, the values must be in ascending order, and then compute the average age for all the adjacent people.

Loves Popcorn	Loves Soda	Age	Happy Person
Yes	Yes	7	No
yes	No	12	No
No	yes	18	yes
No	yes	35	yes
yes	yes	38	yes
yes	No	50	No
No	No	83	No


Age
7
9.5
12
15
18
26.5
35
36.5
38
44
50
66.5
83

Let’s Implement a Decision Tree Classifier - 12

We will calculate the Gini Impurity values for each average age. Let’s look at an example by selecting one of the average and see how we can compute gini impurity.

Loves Popcorn	Loves Soda	**Age**	Happy Person
Yes	Yes	7	No
yes	No	12	No
No	yes	18	yes
No	yes	35	yes
yes	yes	38	yes
yes	No	50	No
No	No	83	No

Average Age

7 +12 = 9.5

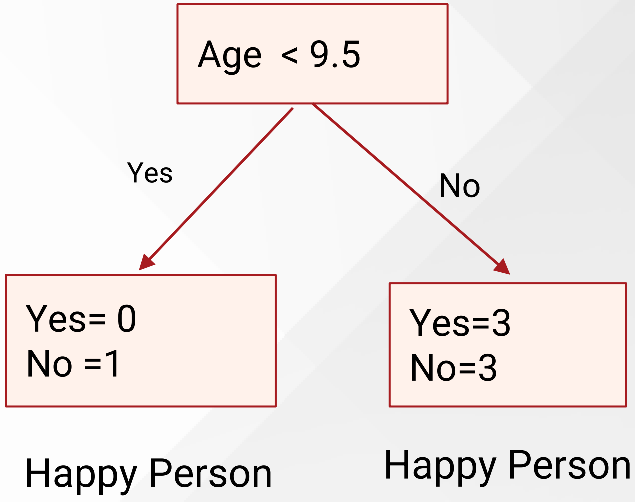
12+18 = 15

18+35 = 26.5

35+38 = 36.5

38+50 = 44

50+83 = 66.5


Age < 9.5

Yes

No

Yes= 0
No =1

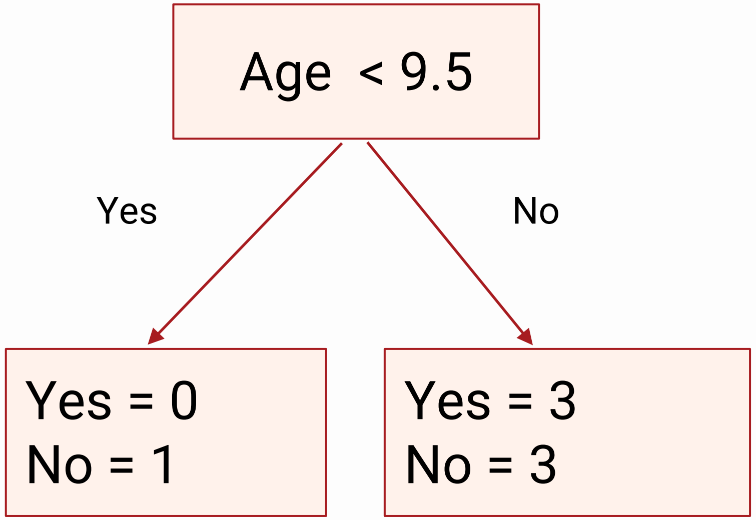
Yes=3
No=3

Happy Person

Happy Person

Let’s Implement a Decision Tree Classifier - 13

So, the Gini Impurity would be:



GI for a left leaf node = 1- (probability of “yes”)^2 - (probability of “No”)^2

$=1-(0/0+1)^2-(1/0+1)^2$

$= 0$

GI for a right leaf node =1-(probability of “yes”)^2 - (probability of“No”)^2 = $1-(3/3+3)^2- (3/3+3)^2$

$= 0.5$

The weighted Gini Impurity for Age < 9.5 would be: $(1/1+6)*0 + (6/1+6)*0.5 = 0.429$

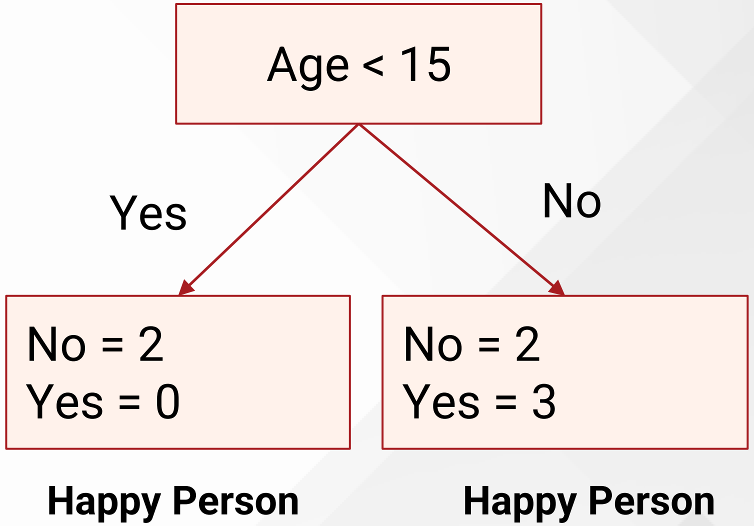
Let’s Implement a Decision Tree Classifier - 14

Likewise, we will calculate all other Gini impurities for all candidate values.

Average Age	Gini Impurity
9.5	0.429
15	**0.343**
26.5	0.476
36.5	0.476
44	**0.343**
66.5	0.429

We can observe that low impurity for Avg Ages 15 and 44 has the common lower contaminant 0.343

In this case, we will choose the Avg Age of 15 for a split.



Let’s Implement a Decision Tree Classifier - 15

Loves Popcorn	Loves Soda	Age	Happy Person
Yes	Yes	7	No
yes	No	12	No
No	yes	18	yes
No	yes	35	yes
yes	yes	38	yes
yes	No	50	No
No	No	83	No

There is a simple battle among **loves popcorn, loves soda, and age** who will be the tree’s root node.

Where a **happy person** is the target variable.

Let’s Implement a Decision Tree Classifier - 16

We will compare the weighted gini impurity for all the attributes from the selection method measures. Which point will be the root node?

Loves Popcorn	Loves Soda	Age	Happy Person
Yes	Yes	7	No
yes	No	12	No
No	yes	18	yes
No	yes	35	yes
yes	yes	38	yes
yes	No	50	No
No	No	83	No

Weighted gini impurities as follows :-

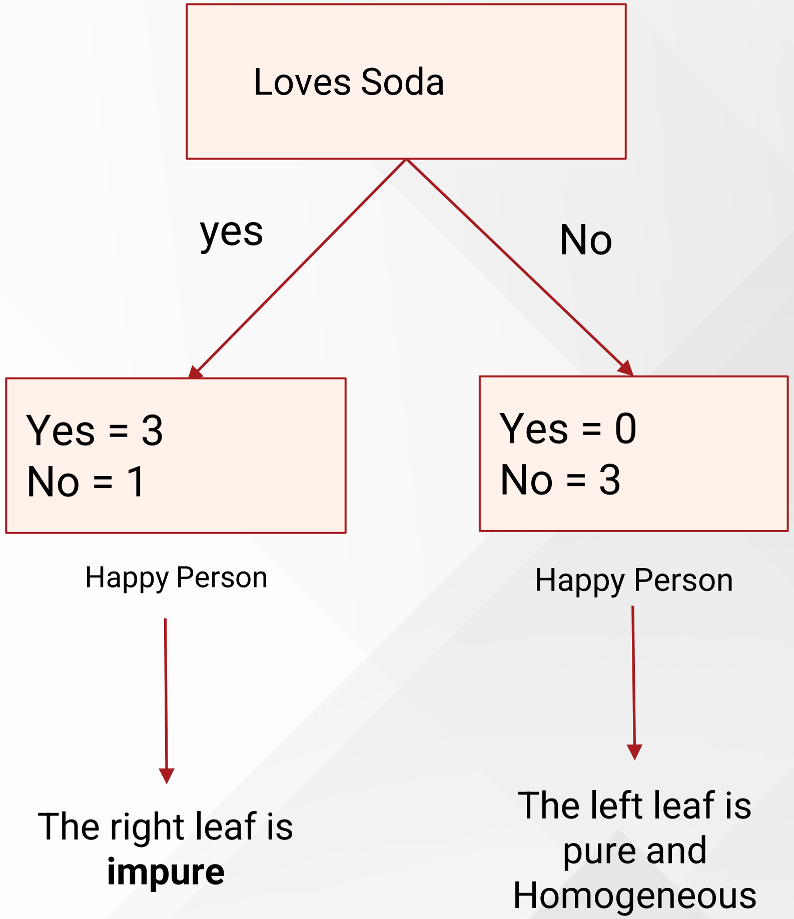
- Gini impurity for Loves Popcorn = 0.405
- Gini impurity for Age < 15 = 0.343
- Gini impurity for Loves Soda =0.214

We know that leaves has lowest impurity for **Loves Soda.** Hence we will select the **Loves Soda** as the root node of the final tree.

Let’s Implement a Decision Tree Classifier - 17

We will put Love Soda in the top as a root node and then we will try to construct a tree.

Loves Popcorn	Loves Soda	Age	Happy Person
Yes	Yes	7	No
yes	No	12	No
No	yes	18	yes
No	yes	35	yes
yes	yes	38	yes
yes	No	50	No
No	No	83	No

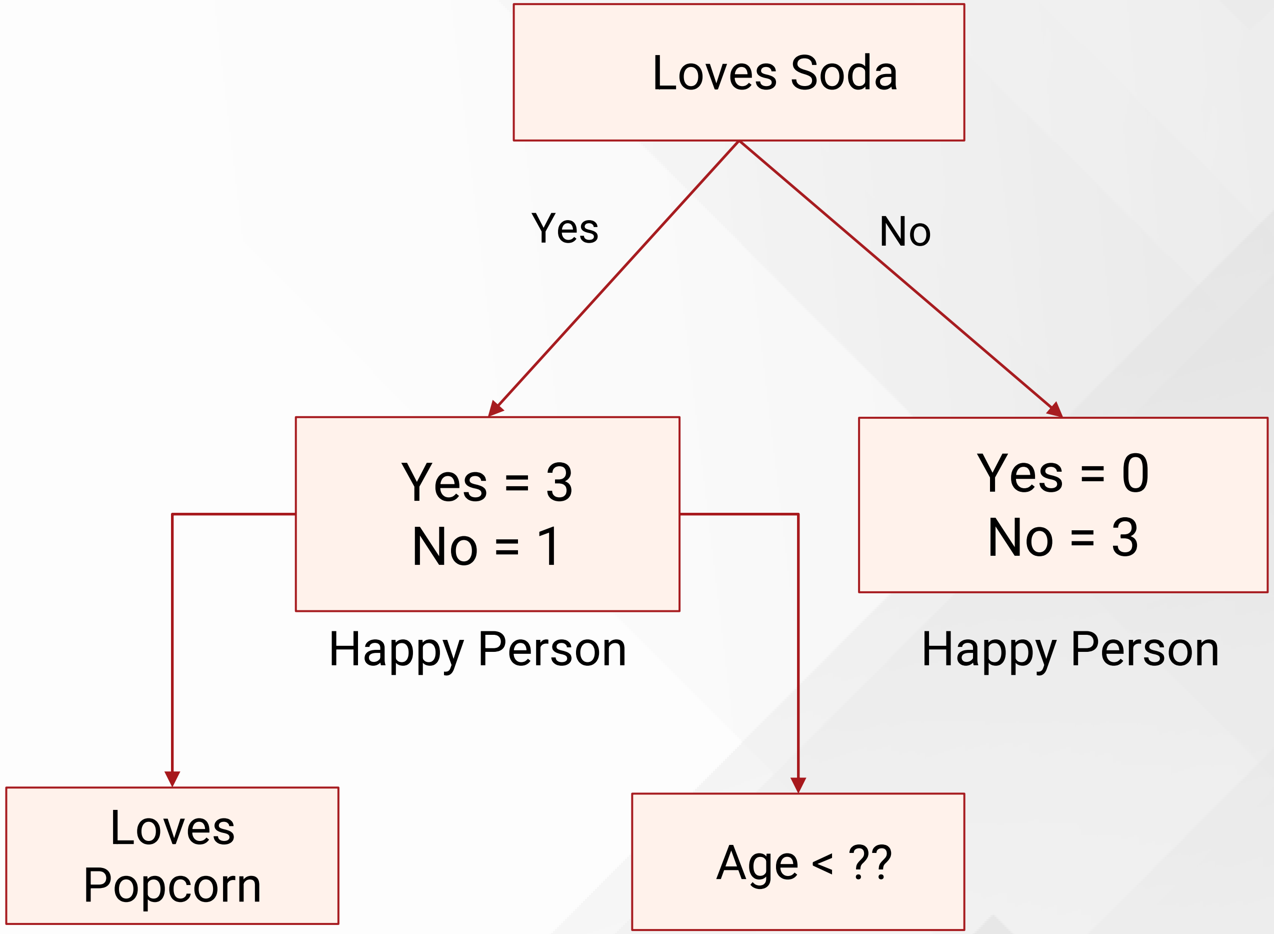


The right leaf is **impure**

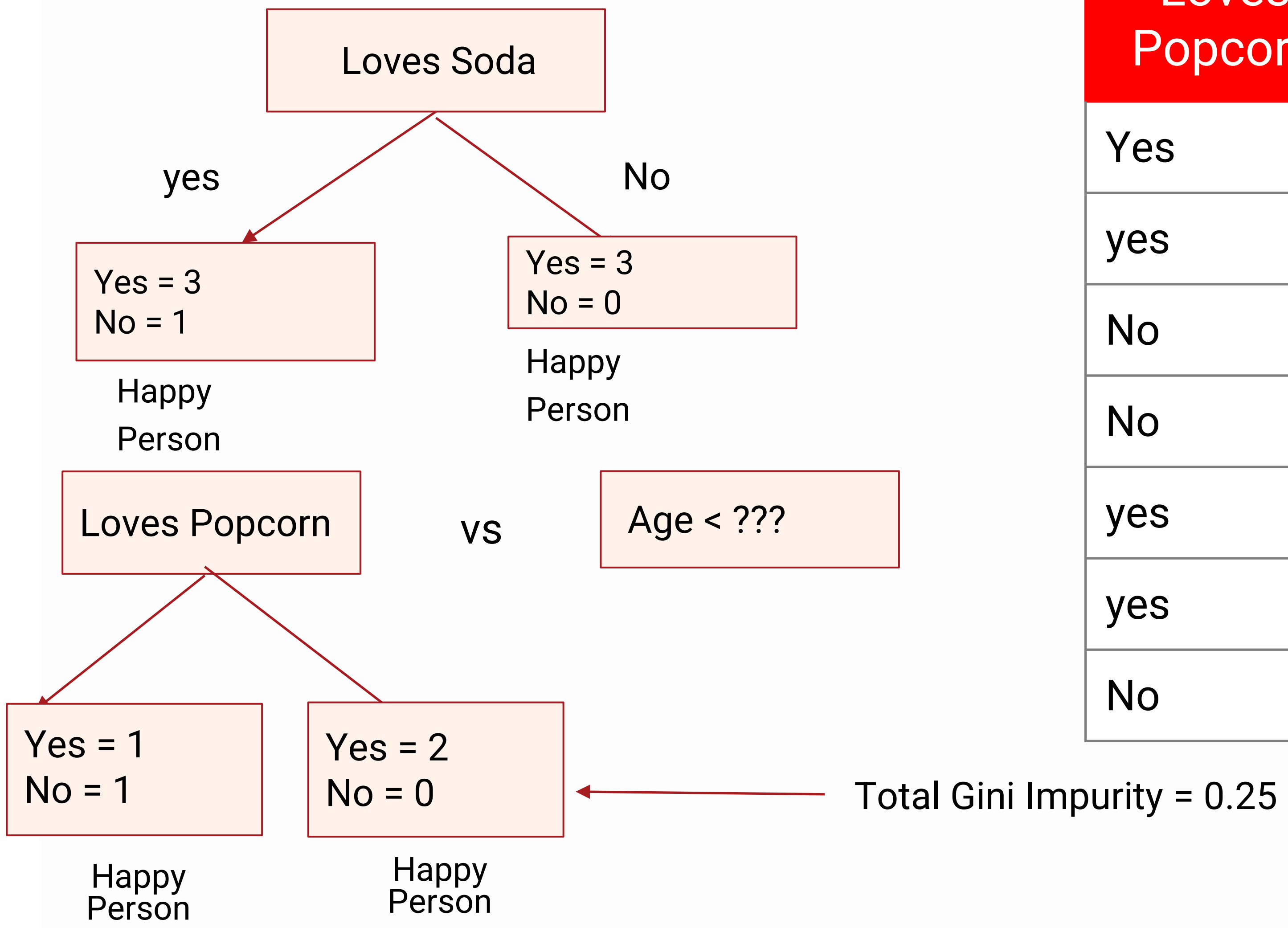
The left leaf is pure and Homogeneous

Let’s Implement a Decision Tree Classifier - 18

- As we can observe that in the Left Node there is impurity. To reduce the impurity we will have to go one more level or more set of questions to reduce impurity in the leaves.
- Let's see if we can reduce the impurity by splitting the people that Loves Soda based on Loves Popcorn and Age.

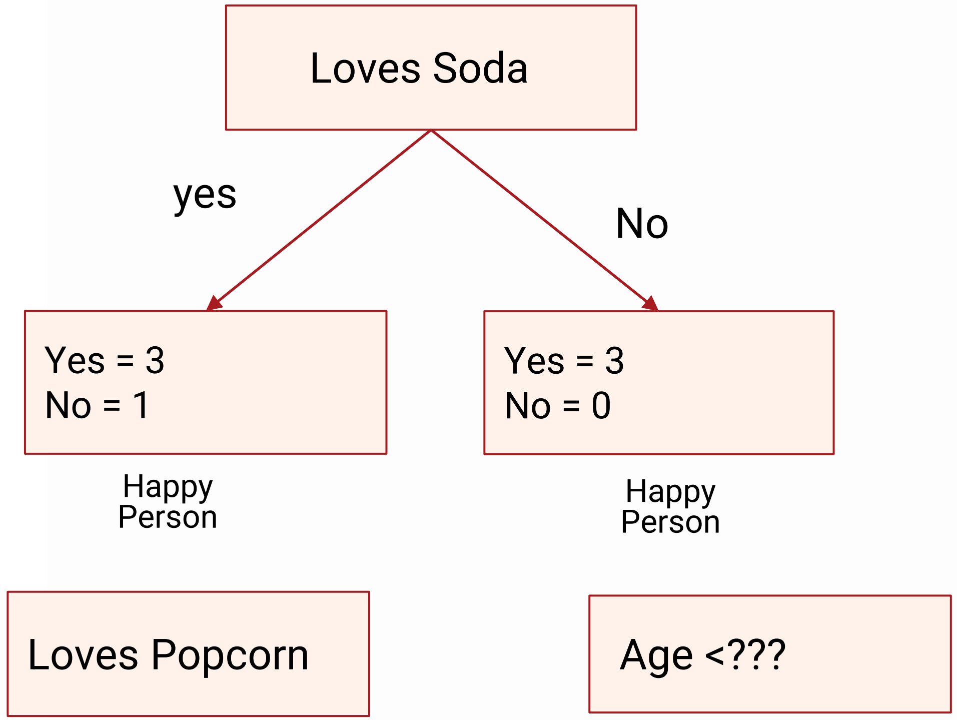


Let’s Implement a Decision Tree Classifier - 19



Loves Popcorn	Loves Soda	Age	Happy Person
Yes	**Yes**	**7**	No
yes	No	12	No
No	**yes**	**18**	yes
No	**yes**	**35**	yes
yes	**yes**	**38**	yes
yes	No	50	No
No	No	83	No

Let’s Implement a Decision Tree Classifier - 20



Gini Impurity = 0.25

Loves Popcorn	Loves Soda	Age	Happy Person
Yes	**Yes**	**7**	No
yes	No	12	No
No	**yes**	**18**	yes
No	**yes**	**35**	yes
yes	**yes**	**38**	yes
yes	No	50	No
No	No	83	No

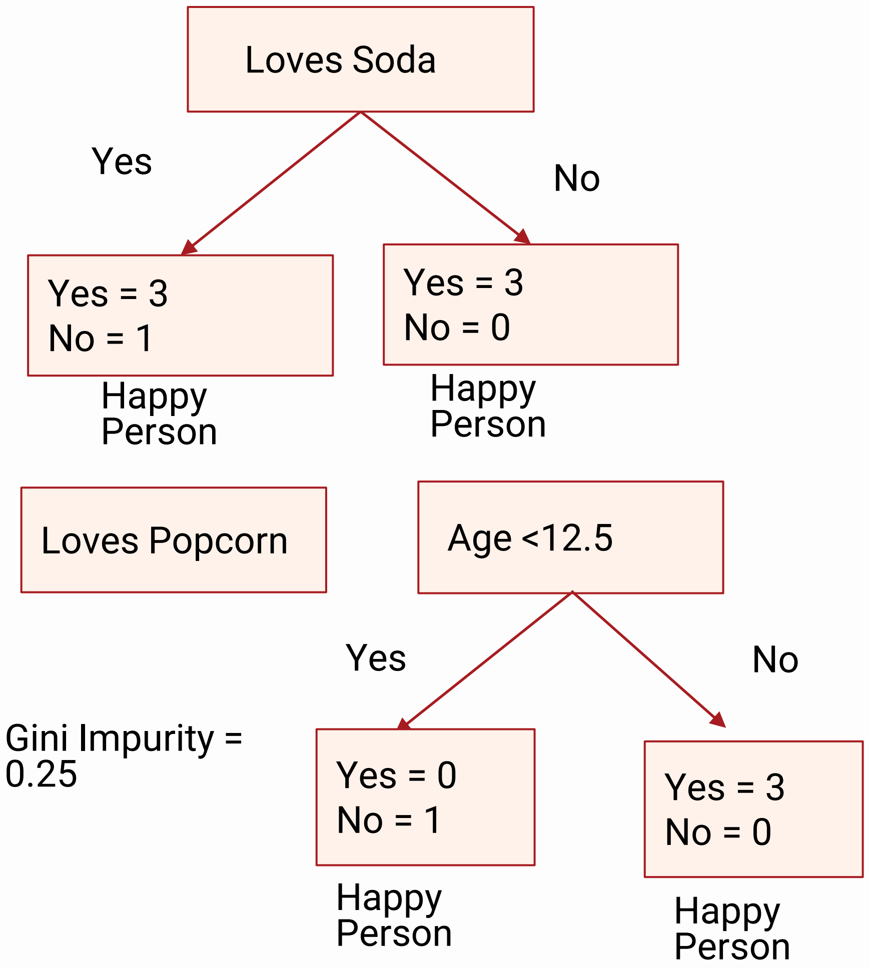
We will calculate the average age for those people who Loves Soda and its **yes.**

7 + 18 = 12.5

18 + 35 = 26.5

35 + 38 = 36.5

Let’s Implement a Decision Tree Classifier - 21



Loves Soda

Yes

No

Yes = 3
No = 1

Happy Person

Yes = 3
No = 0

Happy Person

Loves Popcorn

Age <12.5

Yes

No

Gini Impurity = 0.25

Yes = 0
No = 1

Happy Person

Yes = 3
No = 0

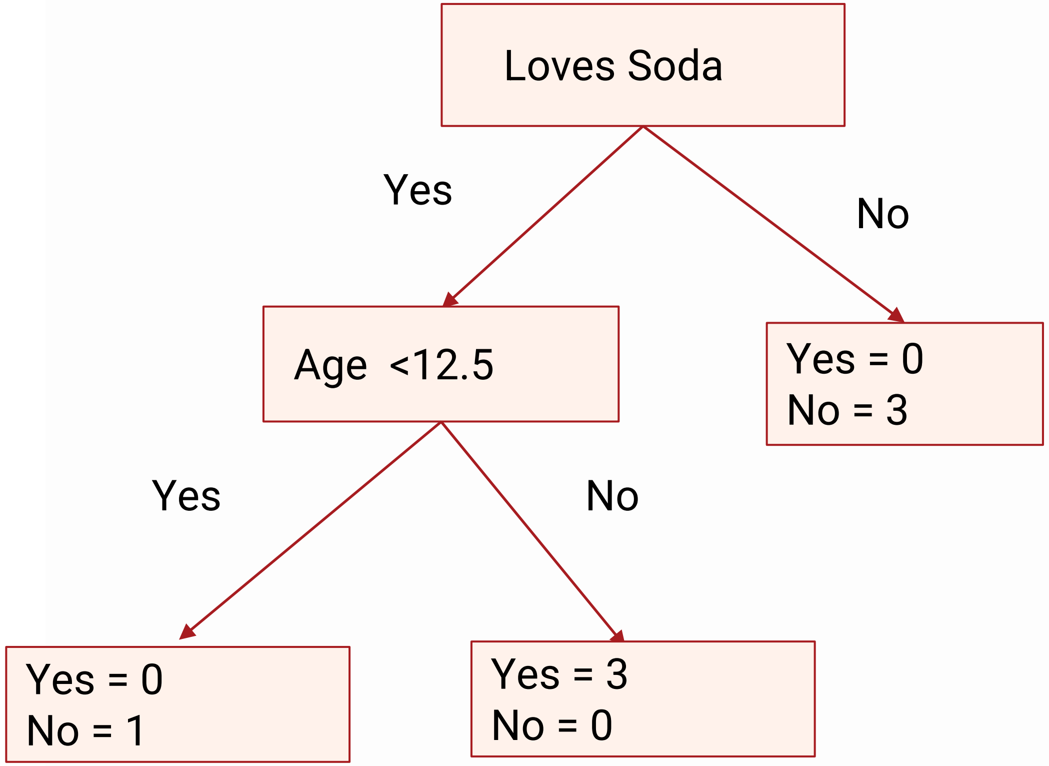
Happy Person

Over here we will take the Lowest impurity for Avg Age < 12.5 and will try to construct leaves for that node.

Loves Popcorn	Loves Soda	Age	Happy Person
Yes	**Yes**	**7**	No
yes	No	12	No
No	**yes**	**18**	yes
No	**yes**	**35**	yes
yes	**yes**	**38**	yes
yes	No	50	No
No	No	83	No

Let’s Implement a Decision Tree Classifier - 22

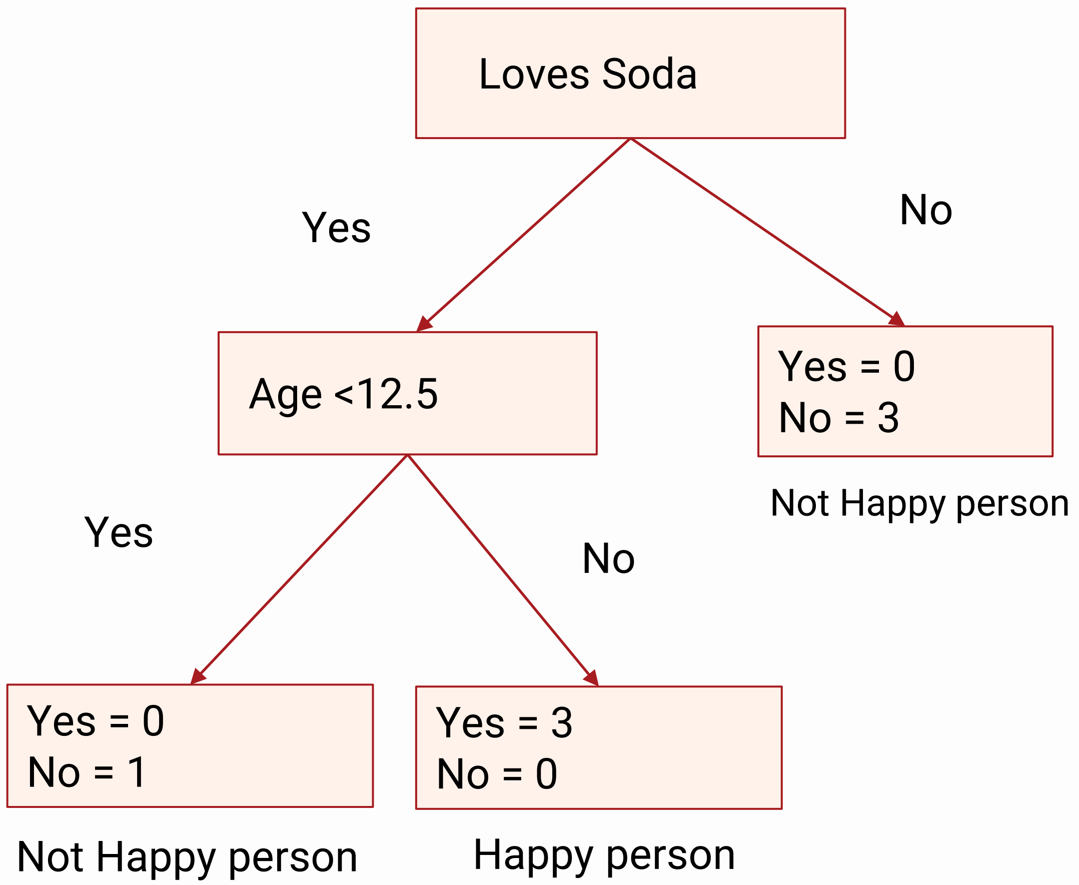
Now, we will finalize the Age as an internal node, person who loves soda and age < 12.5 can achieve impurity in the leaf nodes.



Loves Popcorn	Loves Soda	Age	Happy Person
Yes	**Yes**	**7**	No
yes	No	12	No
No	**yes**	**18**	yes
No	**yes**	**35**	yes
yes	**yes**	**38**	yes
yes	No	50	No
No	No	83	No

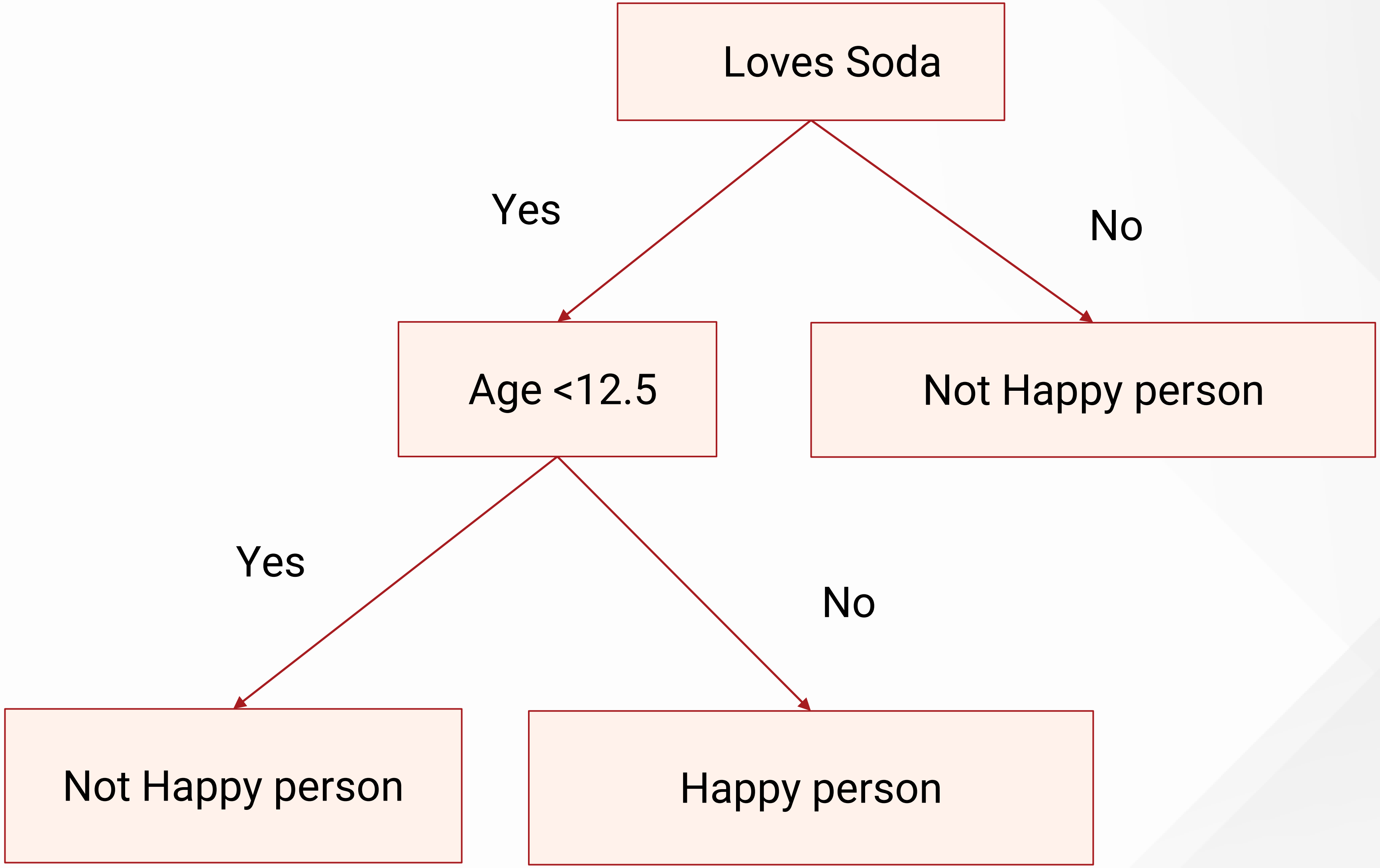
Let’s Implement a Decision Tree Classifier - 23

Once we are done with building the optimal tree, we will now assign the labels to the output leaf.



We need to assign each leaf Happy person or Not Happy person as per the majority of yes/no in the leaf nodes.

Final Tree



Loves Soda

Yes

No

Age <12.5

Not Happy person

Yes

No

Not Happy person

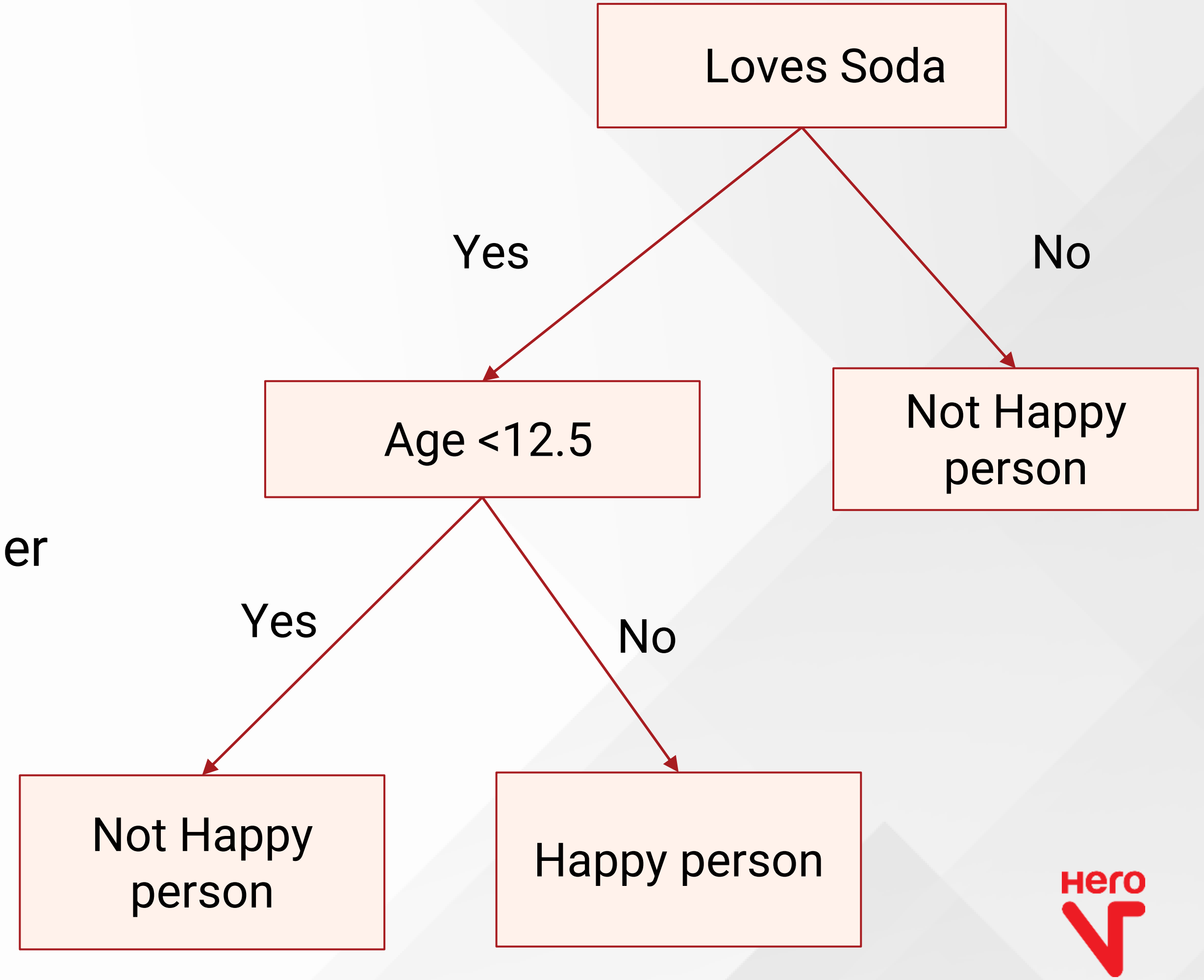
Happy person

Prediction of New Data Point

The below table shows new data provided to decision tree - let's see how decision tree will work.

Loves Popcorn	Loves Soda	Age	Happy Person
Yes	yes	15	???

On this very information, if we want to predict whether the person will be happy/not.



Quiz Poll

1. What are the metrics to measure the impurities of a decision tree classifier?

   a) Entropy

   b) Gini index

   c) Both I & II above

   d) Standard deviation reduction.

2. Suppose a dataset consist of 4 features where 3 features are predictors, and another feature is Target variable. Now suppose if we have to select a feature as root node, then will it be an arbitrarily selection feature ?

   a) Yes

   b) No

3. What will be the node that we will be calling in a decision tree where there will be no further splits happening?

   a) Root node

   b) Terminal node

   c) Decision node

   d) Branch node

Quiz Poll

4. Assume in a dataset, we have 4 features out of those one feature is continuous variable. In selecting the root node, we have to compute the gini index for each variable. So when it comes to a constant variable, how will the gini impurity be calculated?

a) Taking account of the continuous variable.

b) Halving the records and then computing the gini impurity.

c) Taking 70%, 30% records of the cont. Variable and compute gini impurity.

d) Take the average of two points a consecutively and then compute gini impurity.

5. Let's say you have built three decision trees on a small dataset dtree1 with max_depth 2 , dtree 2 with max_depth 5 and dtree 3 with max_depth 8. Which trees is more likely to overfit?

a) Dtree 1

b) Dtree 2

c) Dtree 3

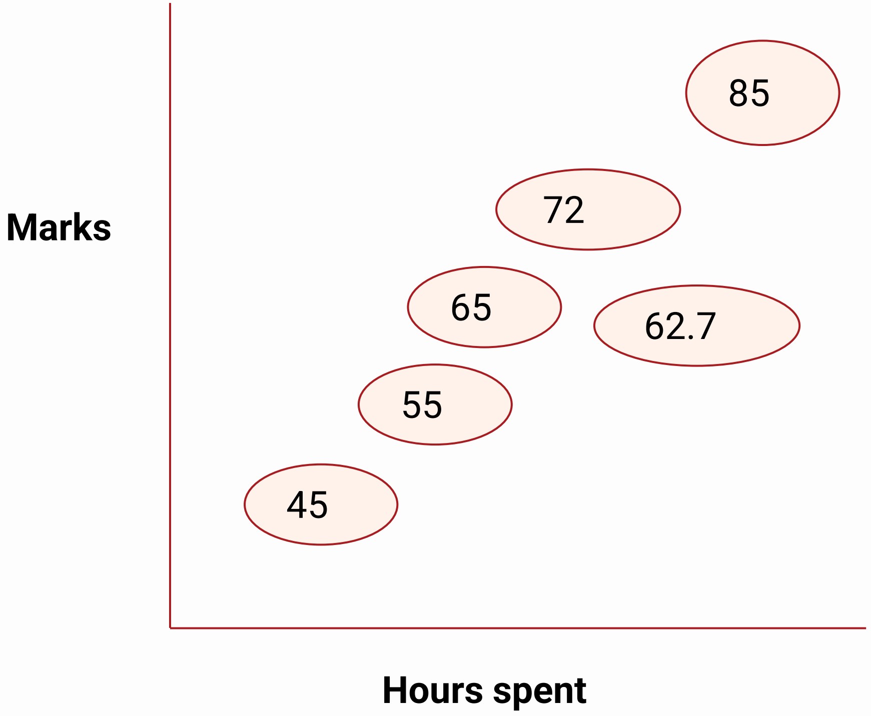
d) None

Demo

Decision Tree: Regression

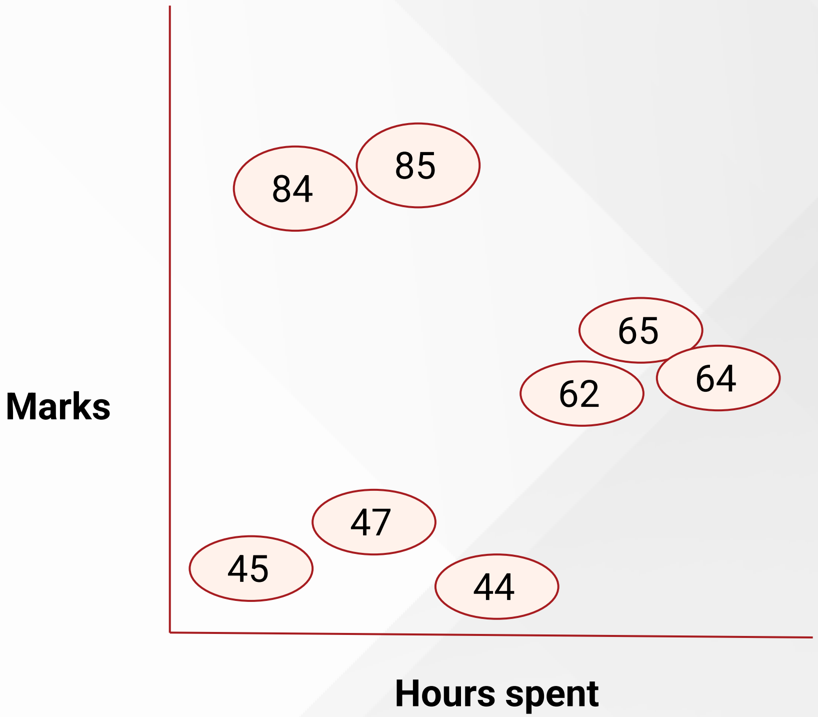
Decision Tree Regression Vs Linear Regression

Linear data & Nonlinear data



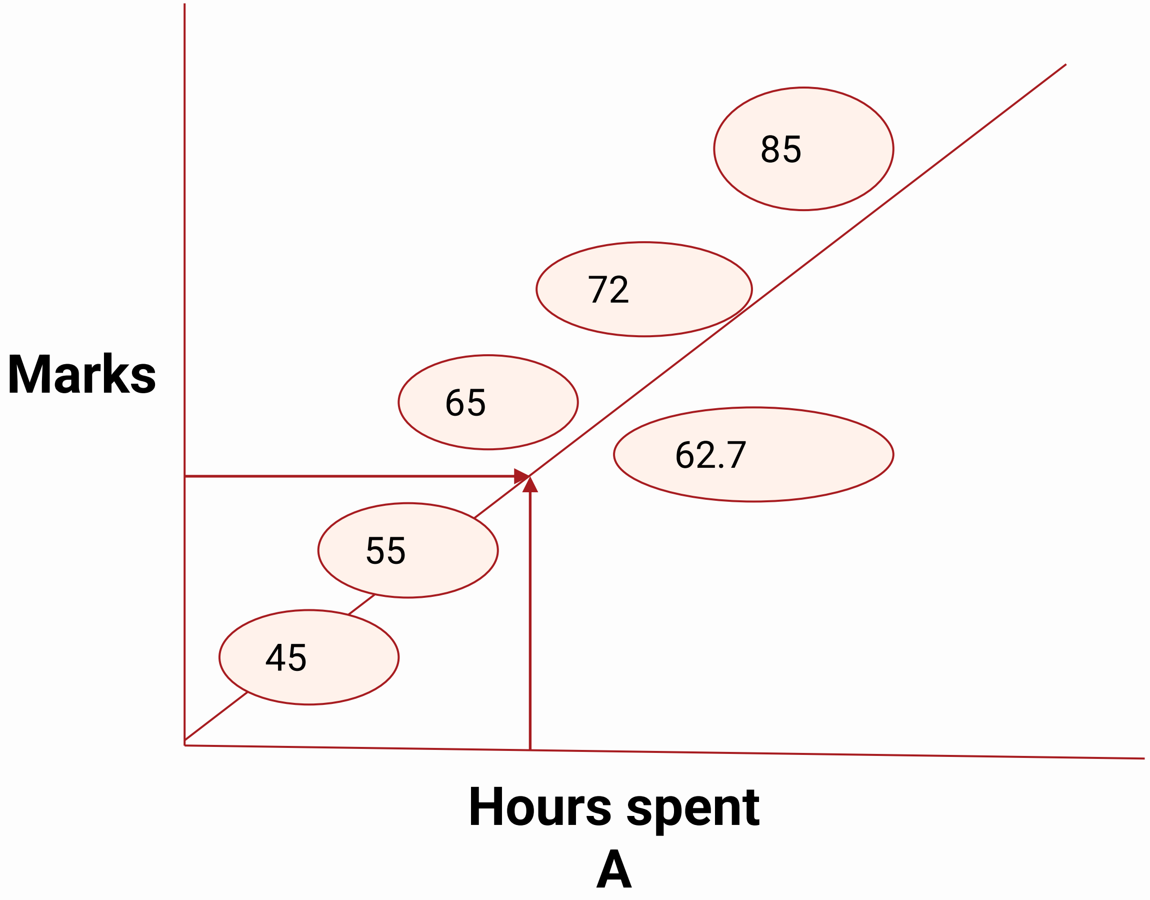
A

vs.



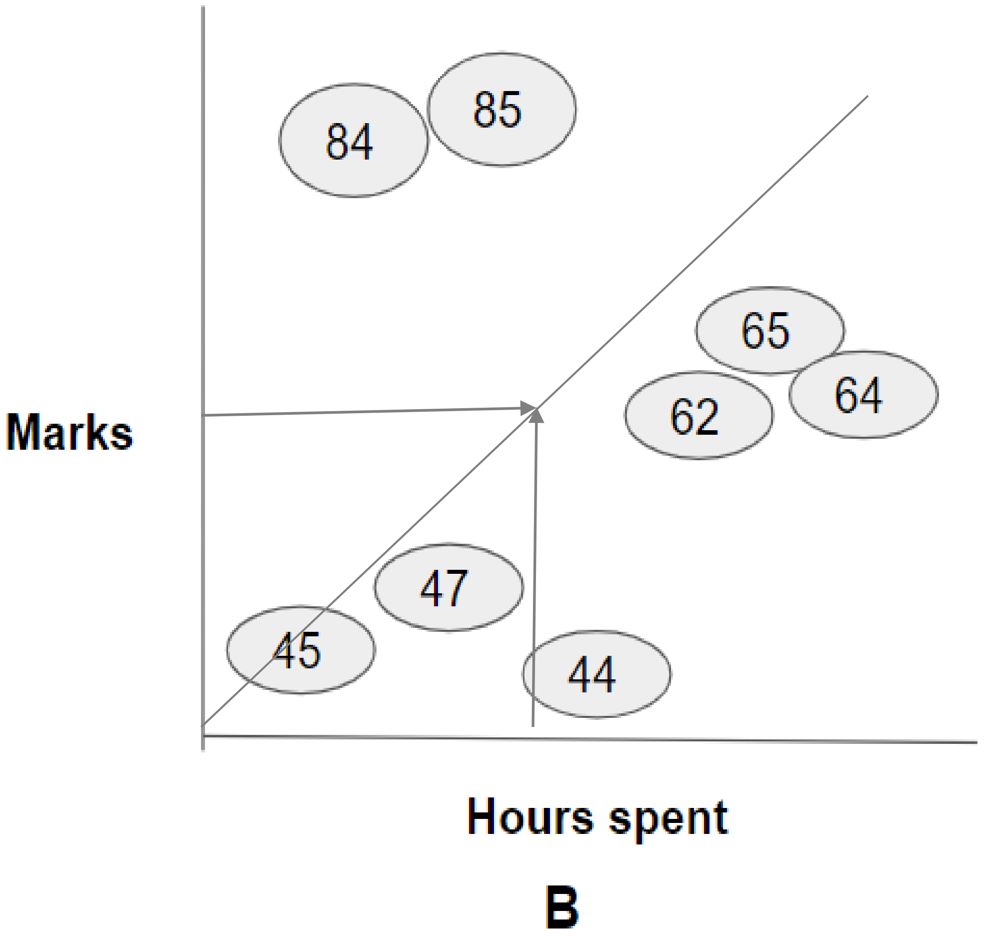
B

Prediction in Linear Data



- If we want to predict the marks obtained when a student has spent **around 6 hours**.
- This becomes easy to predict by a straight line because the nature of the data is linear, and the prediction is accurate.

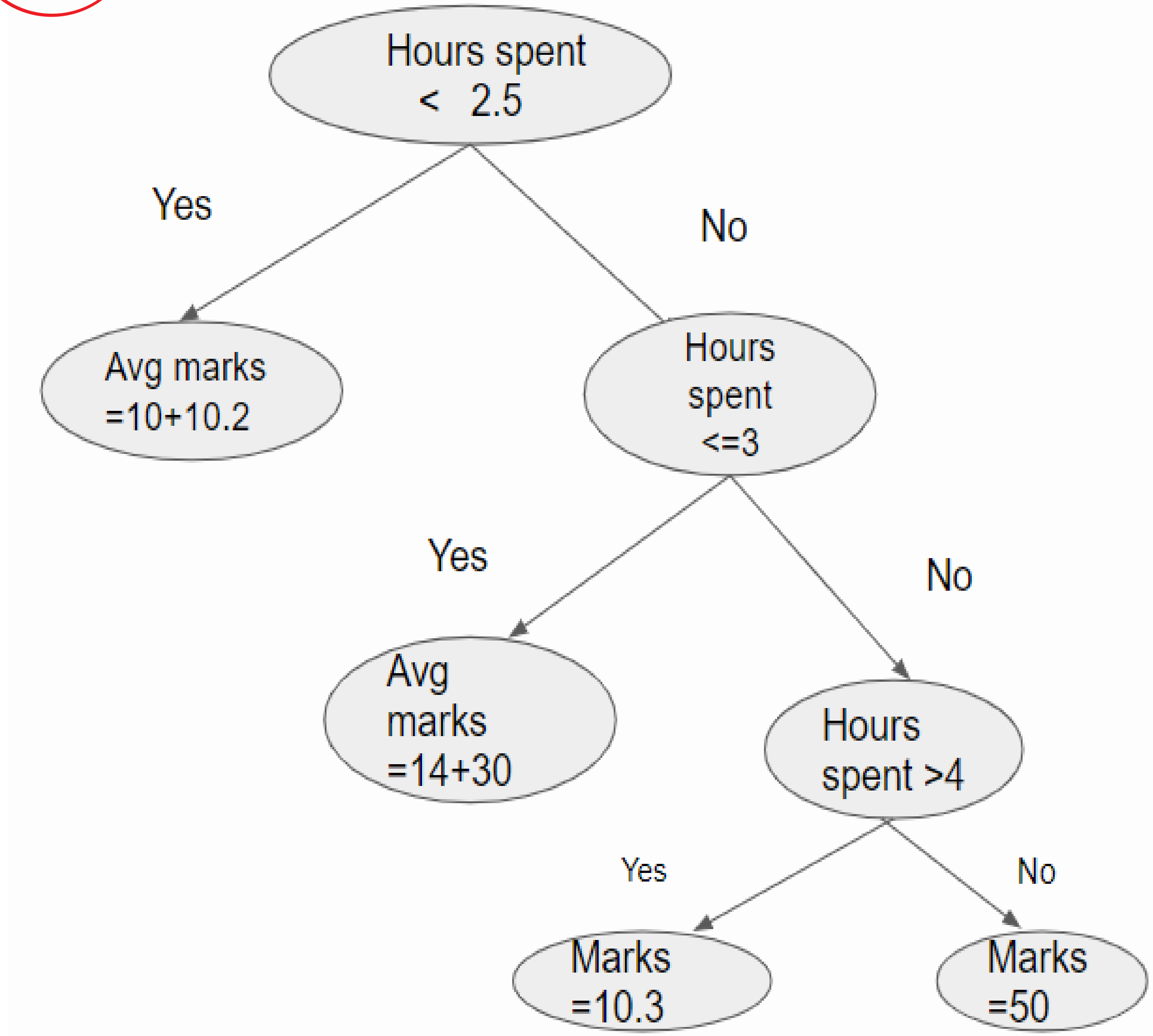
Prediction in Linear Data



Predicting with a straight line will be a big problem as data patterns are spread into clusters. If someone studies six hours a day will get average marks around **64** as per the straight line, it should be around **84.5.**

So, in this scenario, making assumptions with a straight line will be a bad idea. Regression trees will be the best option to handle this shortfall.

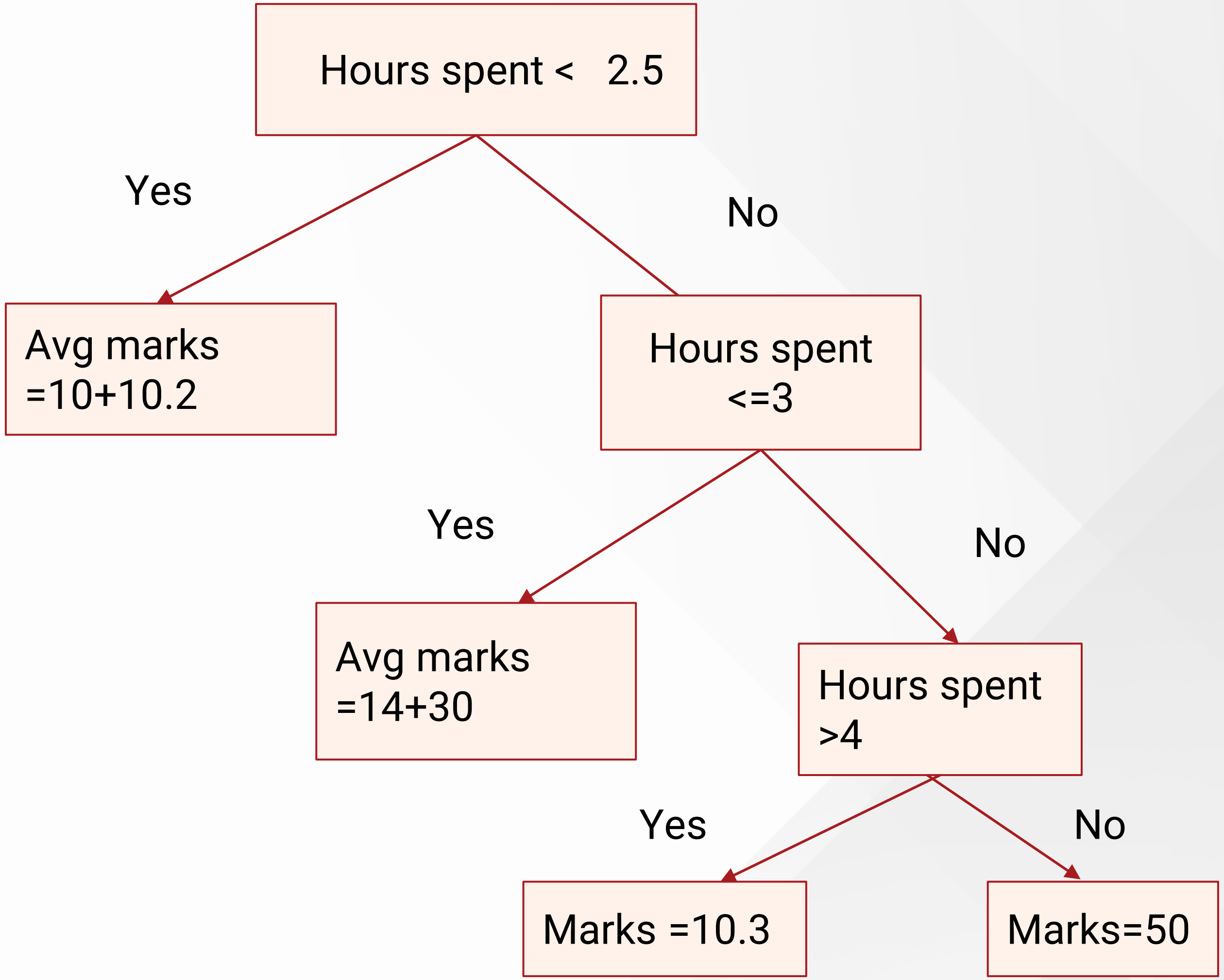
How Prediction Works in Decision Tree Regression - 1



- Let’s say a student spends 2.7 hours studying their expected marks would be.
- Based on the diagram, we can see that when hours spent is > two.5 and in the next level, if hours spent is <= 3, the output avg marks of (14+30) = 22.

How Prediction Works in Decision Tree Regression - 2

Hours Spent	Marks Secured
2	10
2.1	10.2
2.5	14
3	30
4	50
6	10.3



Decision Tree Regression Algorithm

- Assume we have one predictor feature X and target variable Y.
- First, it will average two samples from feature X. Then we will split the samples by the condition less than/more significant than the average, with some pieces left and right.
- We will compute the average of all the samples for the left leaf node and all the examples for the right leaf node. We will calculate the sum of squared residuals combinedly for the left and right leaf nodes.
- Step two to three is recursive until we get the least sum of squared residual for the individual split.
- Once we get the split with the least squared residual sum, we will select that split as the root node.
- Then, we will try to find the optimal split of all the left and right samples, keeping the first split as the root node to find the next decision node or further break.
- We aim to find the leaf nodes with less variance between samples.

Quiz

1. Pick the scenario which is best for the Decision tree regression.

   a) Mark secured by students wrt hours they spent studying.

   b) Weight measure wrt hours they spend in the gym.

   c) Customer retention on spending behavioral

   d) House sale prediction

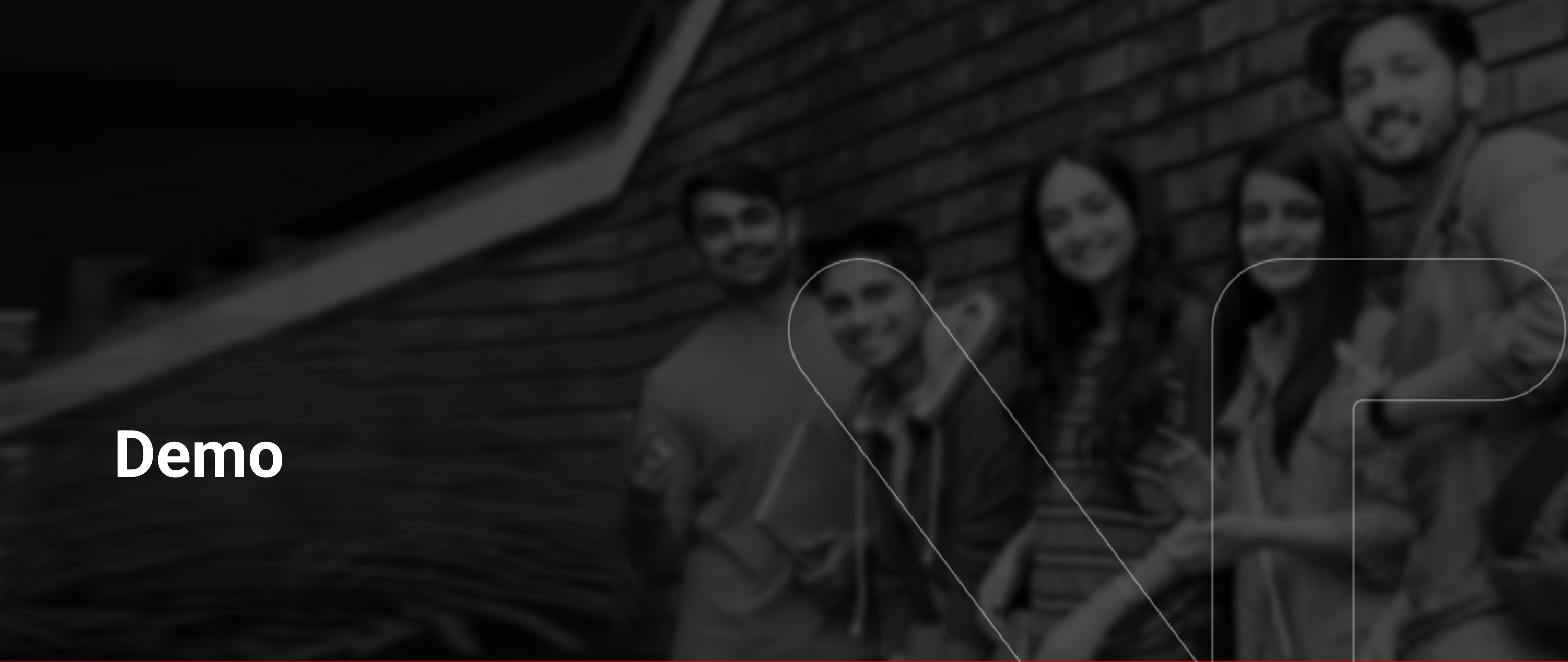
2. What is the measure/criteria to be used to construct a decision tree regression?

   a) Gini Index

   b) Entropy

   c) Information gain

   d) Standard deviation reduction



Demo

Summary

In the session, you learned to:

- Evaluate the concept of decision trees and their usage for solving regression and classification problems
- Assess the concepts that make up the decision tree work, like Gini entropy, information gain, etc.

Thank You!

Copyright © HeroX Private Limited, 2023. All rights reserved.