Decision Trees

Pragati Dharmale

Topics

What is Classification

Types of classification

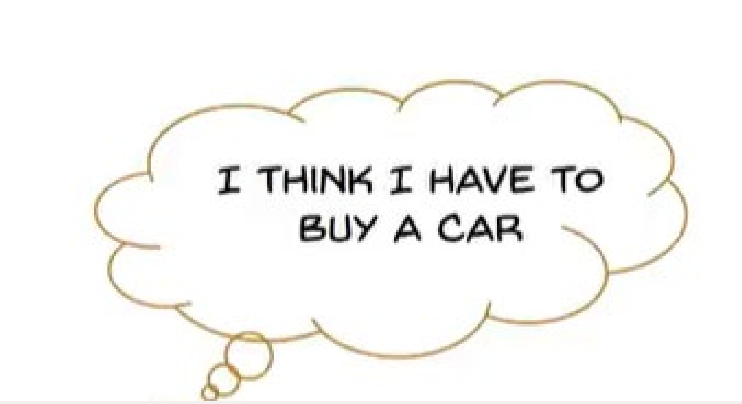
Classification use case

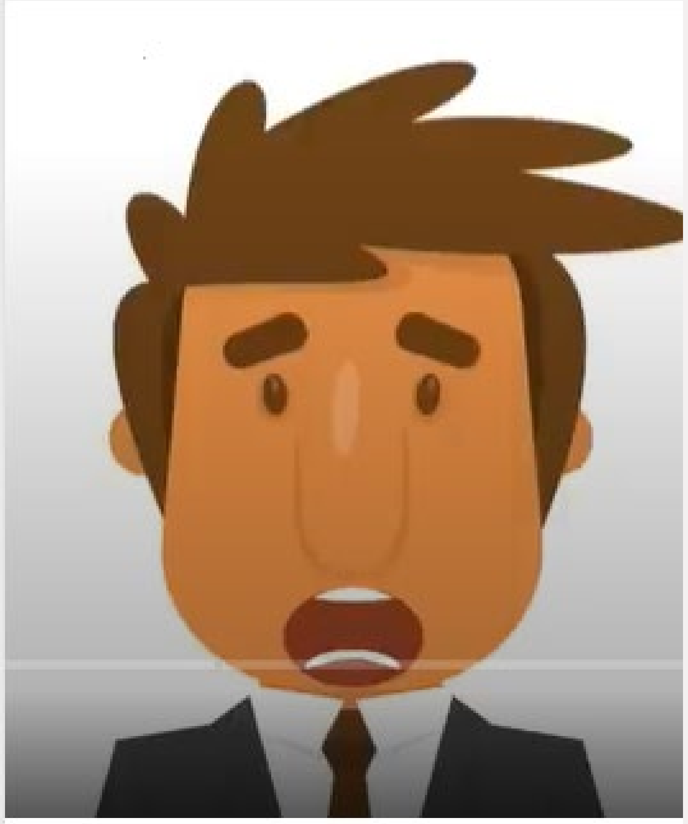
What is decision tree

Terminologies associated with Decision Trees

Visualization of a Decision Tree

Writing a Decision tree classifier using python


I THINK I HAVE TO
BUY A CAR



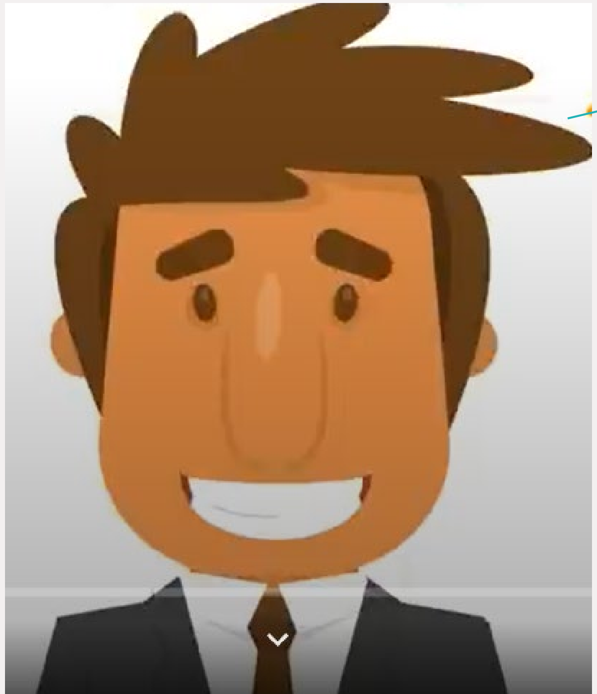
WILL IT BE SUFFICIENT FOR 6 PEOPLE?

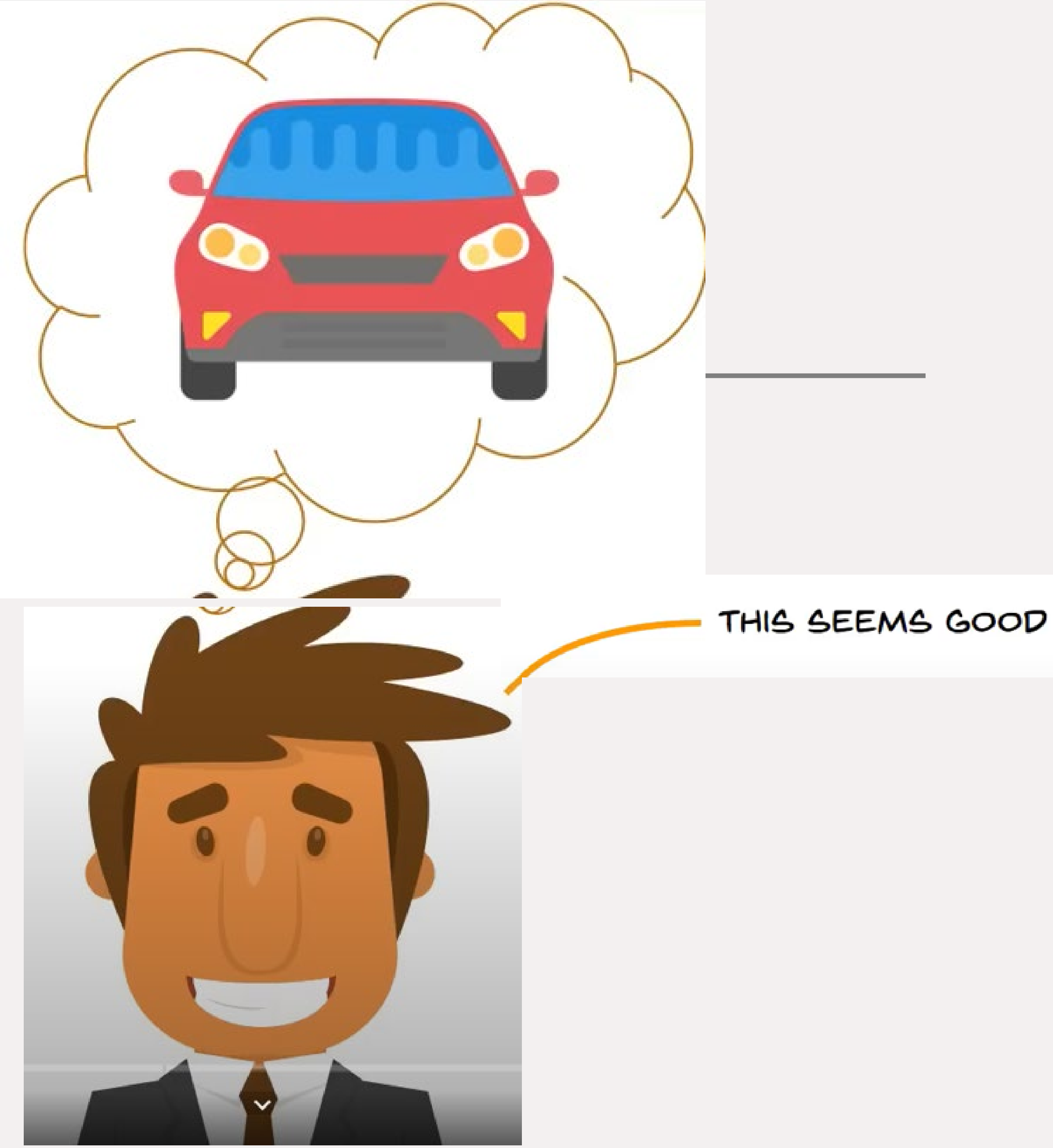
IS PRICE < 15 LAKHS?

NUMBER OF AIRBAGS = 4

IS MILEAGE > 20?

ANTI-LOCK BRAKES?




THIS SEEMS GOOD

What is a Decision Tree?

A decision tree is a map of the possible outcomes of a series of related choices. It allows an individual or organization to weigh possible actions against one another based on their costs, probabilities, and benefits.

As the name goes, it uses a tree-like model of decisions. They can be used either to drive informal discussion or to map out an algorithm that predicts the best choice mathematically.

A decision tree typically starts with a single node, which branches into possible outcomes. Each of those outcomes leads to additional nodes, which branch off into other possibilities. This gives it a tree-like shape.

What is Classification

Classification is the process of dividing the dataset into different categories or groups by adding labels

E.g. Fraud Detection

Classify fruits on the basis of size , color

,taste etc

Types of Classification

Decision Tree

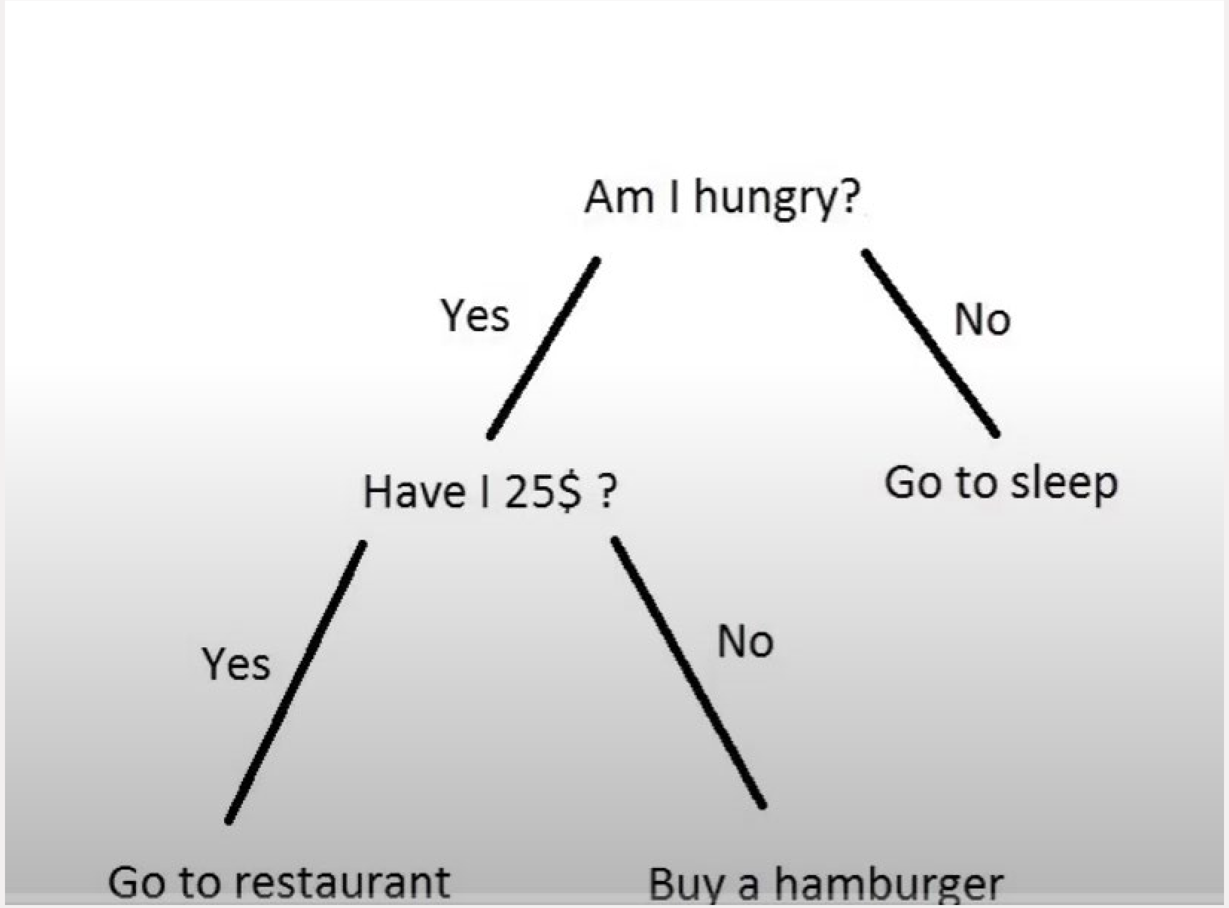
Random forest

Naïve Bias

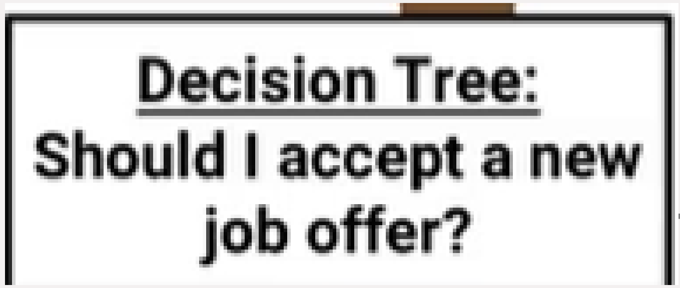
KNN

Decision Trees

- Graphical Representation of all possible solutions to a decision
- Decisions are based on some conditions
- Decision made can be easily explained

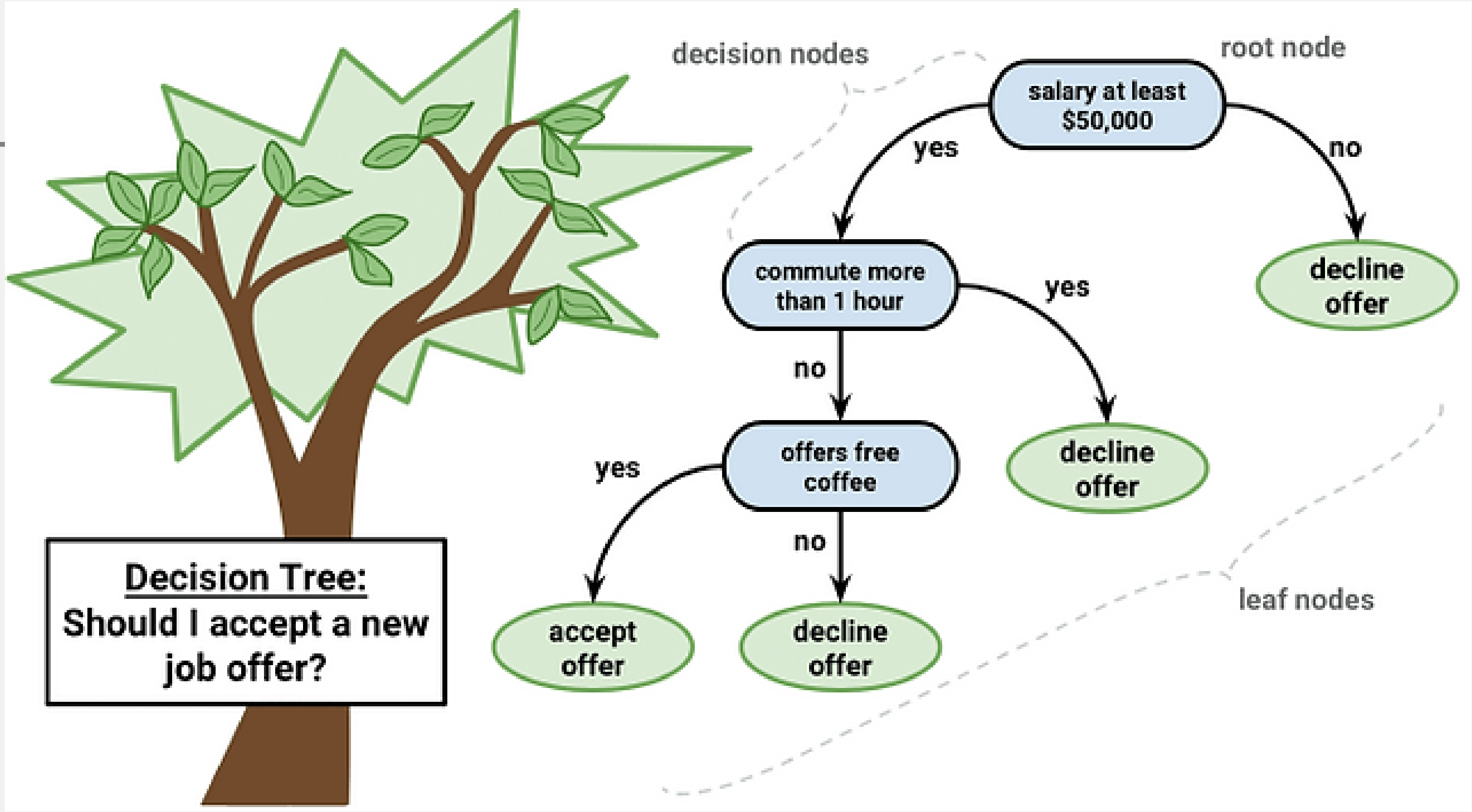


“A Decision Tree is a graphical representation of all the possible solutions to a decision based on certain conditions “



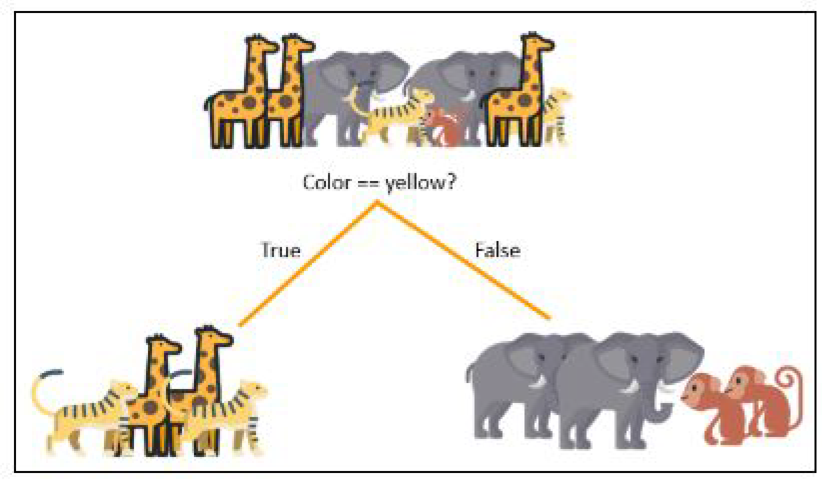
There are three different types of nodes:
chance nodes,
decision nodes,
and end nodes.

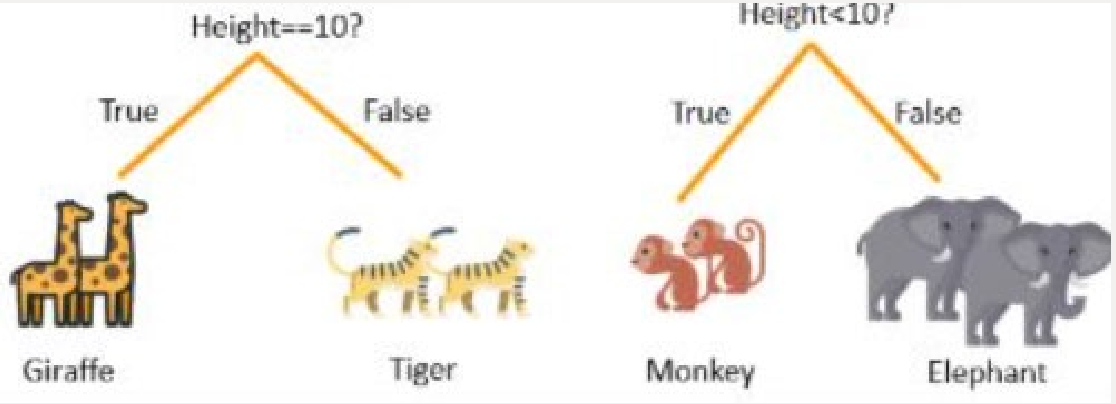
A chance node:
represented by a circle, shows the probabilities of certain results.
A decision node, represented by a square, shows a decision to be made,
and an end node shows the final outcome of a decision path.



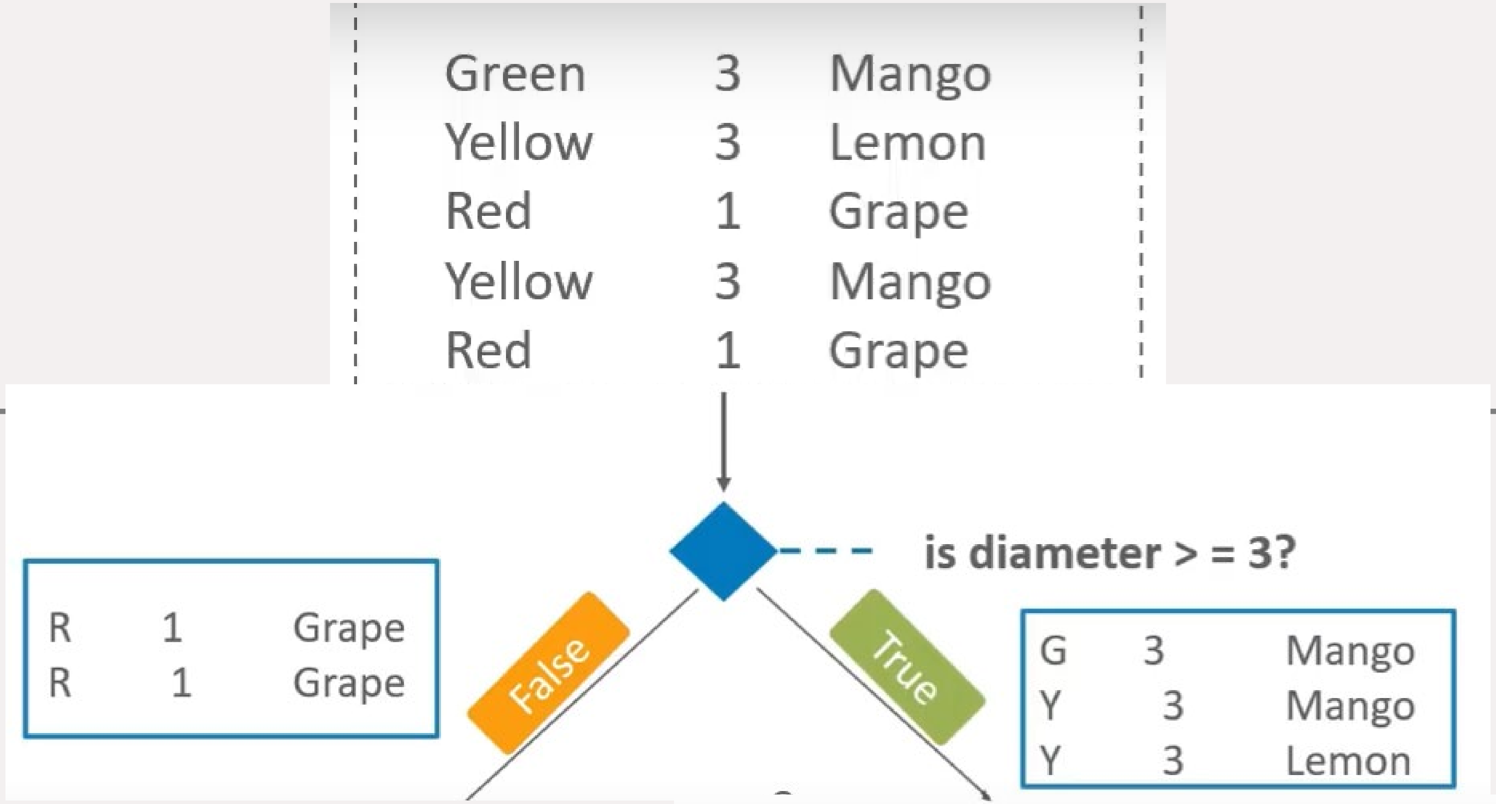
Training Dataset

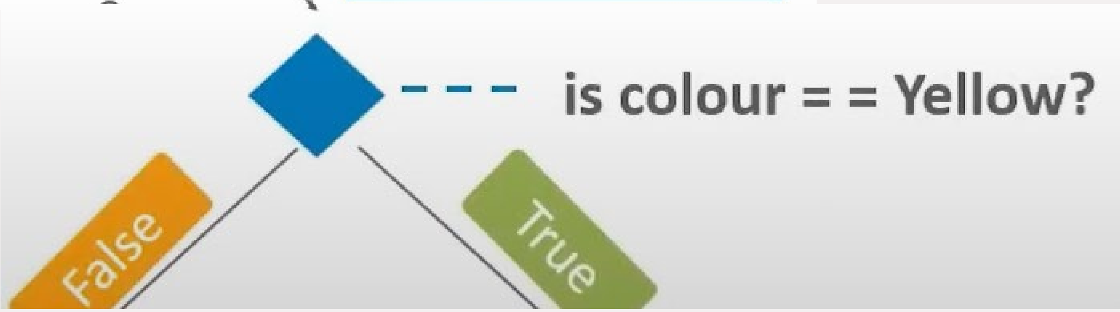
Color	Height	Label
Grey	10	Elephant
Yellow	10	Giraffe
Brown	3	Monkey
Grey	10	Elephant
Yellow	4	Tiger





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





How many Questions to Ask to unmix the label - GINI

Then quantify the questions as in how much a question reduces the uncertainty using a concept called information gain

Decision Tree Terminology

Pruning
Opposite of Splitting, basically removing unwanted branches from the tree

Branch/SubTree
Formed by splitting the tree/node

Parent/Child Node
Root node is the parent node and all the other nodes branched from it is known as child node

Splitting
Splitting is dividing the root node/sub node into different parts on the basis of some condition.

Root Node
It represents the entire population or sample and this further gets divided into two or more homogenous sets.

Leaf Node
Node cannot be further segregated into further nodes

But How do we choose the best attribute?

Or

How does a tree decide where to split?

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

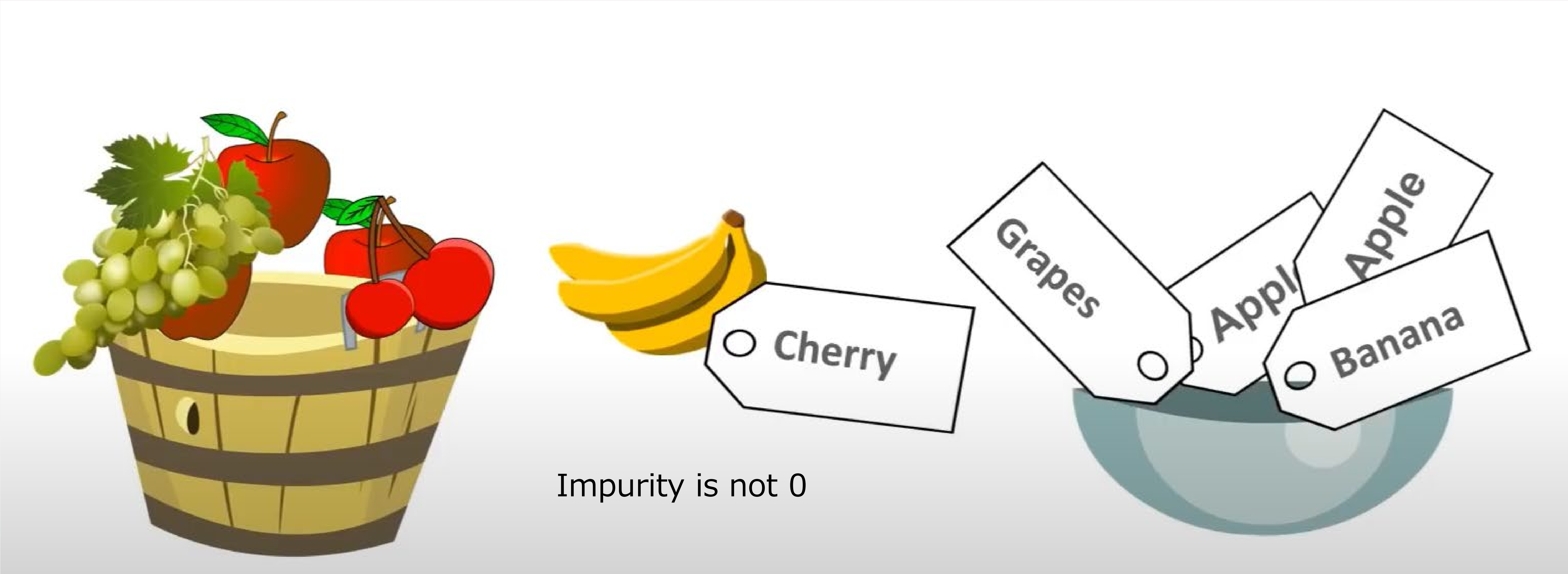
How Does a Tree decides where to split

Entropy: Entropy in Decision Tree stands for homogeneity. If the data is completely homogenous, the entropy is 0, else if the data is divided (50-50%) entropy is 1.
Information Gain: Information Gain is the decrease/increase in Entropy value when the node is split.

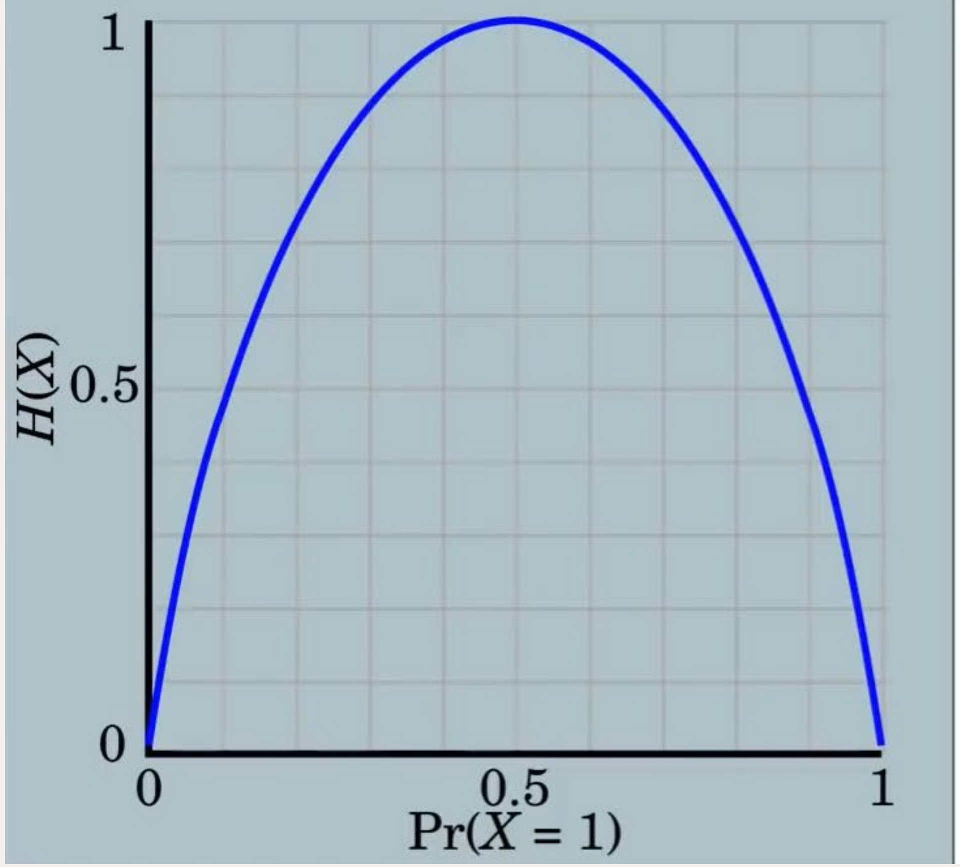
What is impurity

Degree of randomness . Here it is 0

Impurity – non zero



What is Entropy?



Entropy(s) =- P(yes) $\log_2 P(yes)$ – P(no) $\log_2 P(no)$

Where,

- S is the total sample space,
- P(yes) is probability of yes

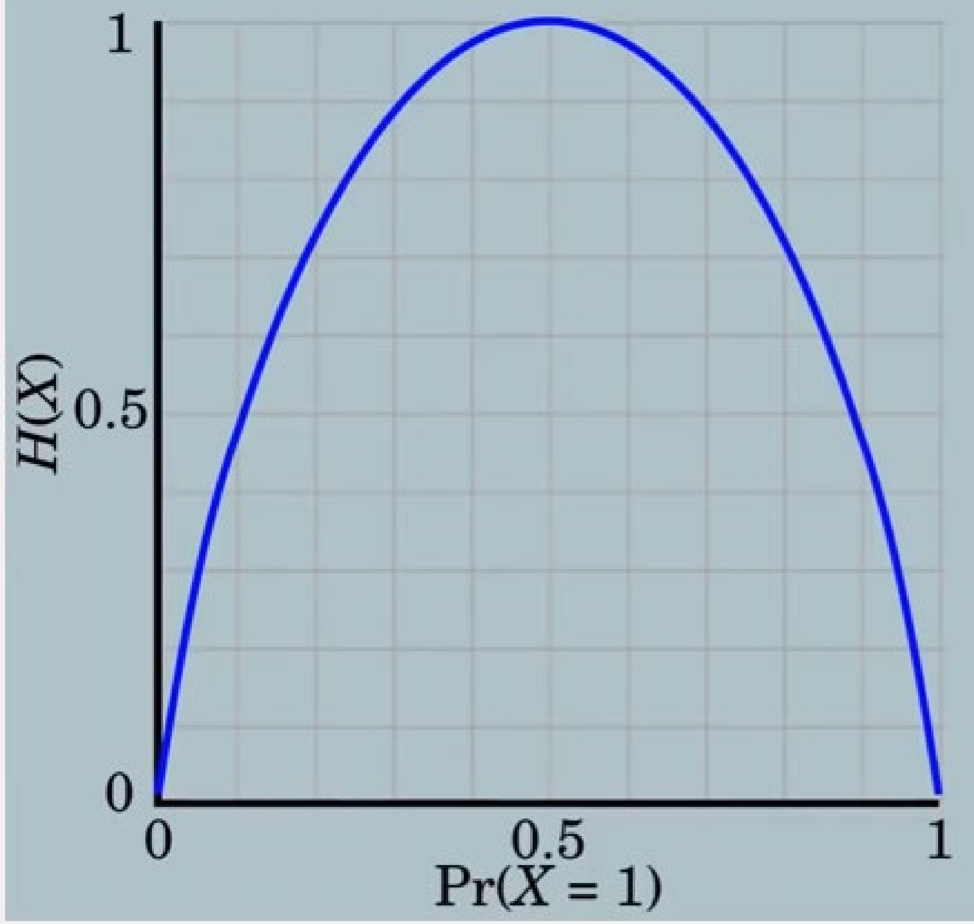
If number of *yes* = number of *no ie P(S) = 0.5*

$\Rightarrow$ *Entropy(s) = 1*

If it contains all yes or all no ie P(S) = 1 or 0

$\Rightarrow$ *Entropy(s) = 0*

What is Entropy?



$E(S) = -P(Yes)\log_2 P(Yes)$

When $P(Yes) = P(No) = 0.5$ ie YES + NO = Total Sample(S)

$E(S) = 0.5\ \log_2 0.5 - 0.5\ \log_2 0.5$

$E(S) = 0.5(\log_2 0.5 - \log_2 0.5)$

$E(S) = 1$

$E(S) = -P(No)\log_2 P(No)$

When $P(No) = 1$ ie No = Total Sample(S)

$E(S) = 1\ \log_2 1$

$E(S) = 0$

What is Information Gain?

- Measures the reduction in entropy
- Decides which attribute should be selected as the decision node

If S is our total collection,

Information Gain = Entropy(S) – [(Weighted Avg) x Entropy(each feature)]

GINI Impurity Vs Entropy

Gini impurity and entropy are not the same, but they are both commonly used measures of impurity in decision trees and other machine learning models.

Gini impurity is a measure of the probability of misclassifying a randomly chosen element in a dataset if it were randomly labeled according to the distribution of labels in the dataset. It is defined as:

Gini impurity = 1 - Σ(p_i)^2

where p_i is the proportion of instances in the dataset that belong to class i.

Entropy, on the other hand, is a measure of the disorder or unpredictability of a system. In the context of decision trees, it is a measure of the impurity of a set of labels. It is defined as:

Entropy = - Σ(p_i)log2(p_i)

where p_i is the proportion of instances in the dataset that belong to class i.

Both Gini impurity and entropy are used to determine the best split for a decision tree node. In general, Gini impurity is preferred when the classes are well-balanced, while entropy is preferred when the classes are highly imbalanced.

But How do we choose the best attribute?

Or

How does a tree decide where to split?

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

Out of 14 instances we have 9 YES and 5 NO

So we have the formula,

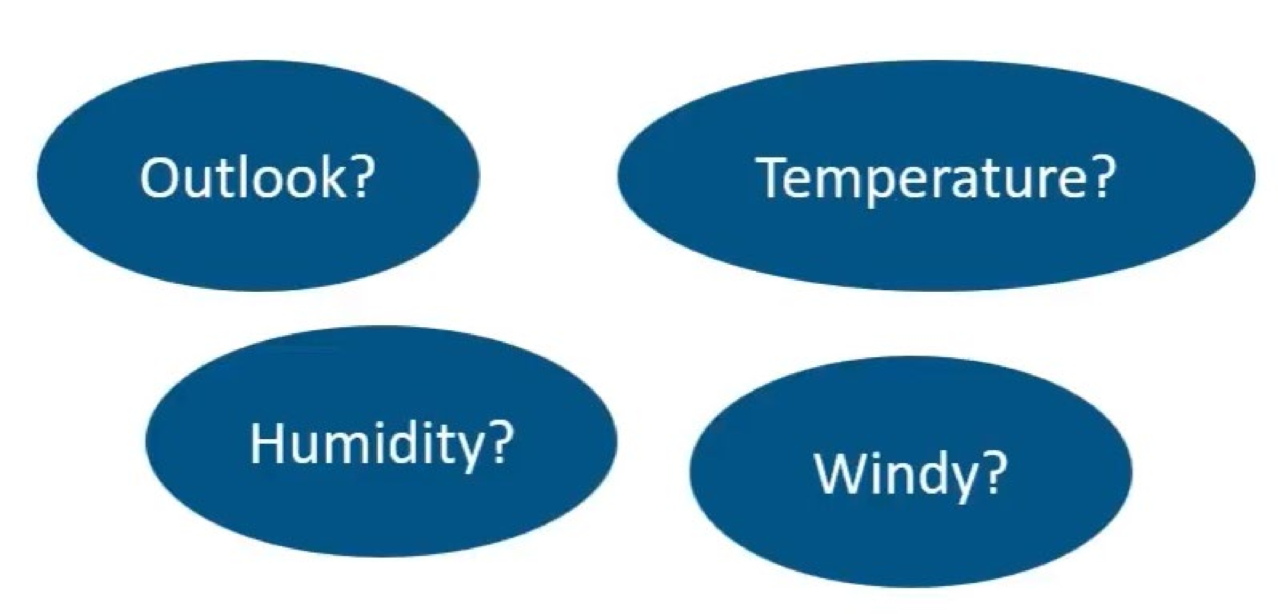
$$E(S) = -P(Yes)\log_2 P(Yes) - P(No)\log_2 P(No)$$

$$E(S) = -(9/14)^* \log_2 9/14 - (5/14)^* \log_2 5/14$$

$$E(S) = 0.41+0.53 = 0.94$$

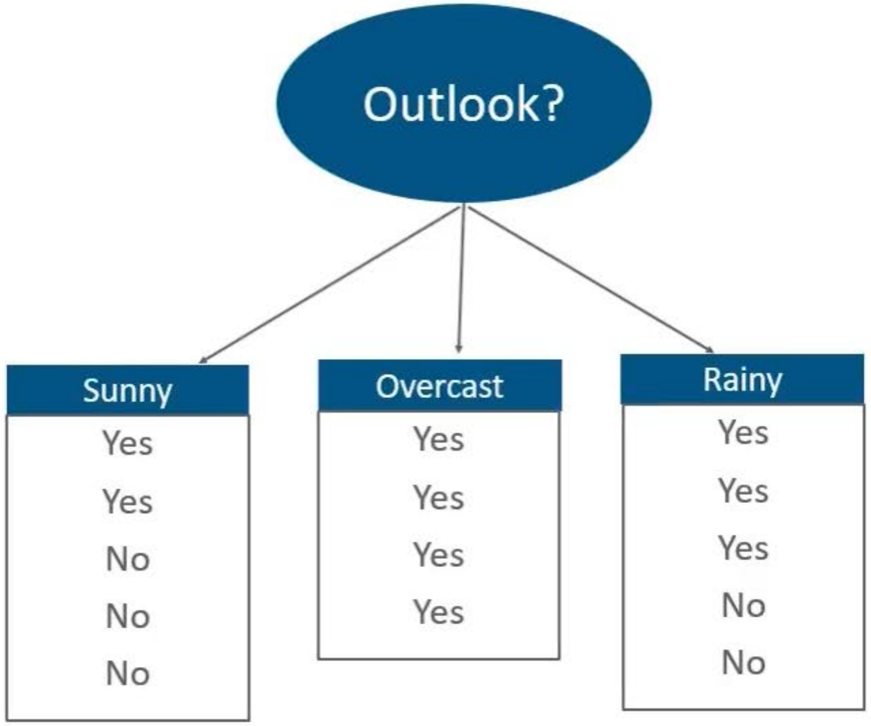
	outlook	temp.	humidity	windy	play
D1	sunny	hot	high	false	no
D2	sunny	hot	high	true	no
D3	overcast	hot	high	false	yes
D4	rainy	mild	high	false	yes
D5	rainy	cool	normal	false	yes
D6	rainy	cool	normal	true	no
D7	overcast	cool	normal	true	yes
D8	sunny	mild	high	false	no
D9	sunny	cool	normal	false	yes
D10	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

Which Node To Select As Root Node?



	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

Which Node To Select As Root Node: Outlook



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

Which Node To Select As Root Node?

$E(Outlook = Sunny) = -2/5 \log_2 2/5 - 3/5 \log_2 3/5 = 0.971$

$E(Outlook = Overcast) = -1 \log_2 1 - 0 \log_2 0 = 0$

$E(Outlook = rainy = -3/5 \log_2 3/5 - 2/5 \log_2 2/5 = 0.971$

Information from outlook,

$I(Outlook) = 5/14 \times 0.971 + 4/14 \times 0 + 5/14 \times 0.971 = 0.693$

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

Which Node To Select As Root Node?

E(Outlook = Sunny) = -2/5 $\log_2 2/5 - 3/5 \log_2 3/5$ = 0.971

E(Outlook = Overcast) = -1 $\log_2 1 - 0 \log_2 0$ = 0

E(Outlook = rainy = -3/5 $\log_2 3/5 - 2/5 \log_2 2/5$ = 0.971

Information from outlook,

I(Outlook) = 5/14 x 0.971 + 4/14 x 0 + 5/14 x 0.971 = 0.693

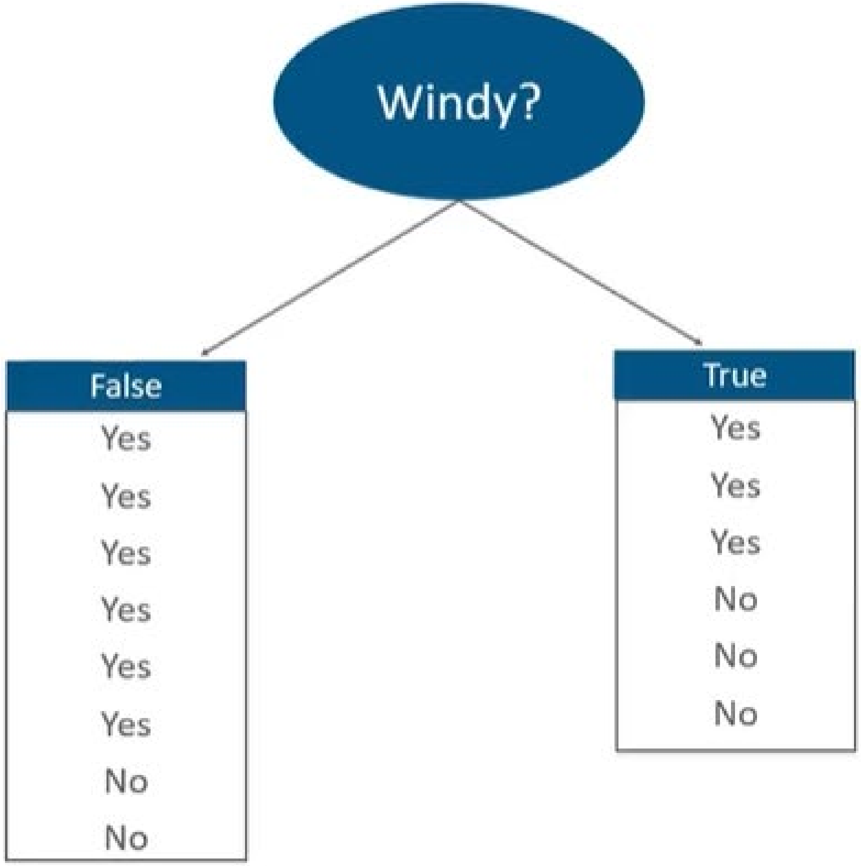
Information gained from outlook,

Gain(Outlook) = E(S) – I(Outlook)

0.94 – 0.693 = 0.247

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

Which Node To Select As Root Node: Windy



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

Which Node To Select As Root Node: Windy

E(Windy = True) = 1

E(Windy = False) = 0.811

Information from windy,

I(Windy) = 8/14 x 0.811+ 6/14 x 1 = 0.892

Information gained from outlook,

Gain(Windy) = E(S) – I(Windy)

0.94 – 0.892 = 0.048

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

Which Node To Select As Root Node

Outlook:
Info 0.693
Gain: 0.940-0.693 0.247

Temperature:
Info 0.911
Gain: 0.940-0.911 0.029

Humidity:
Info 0.788
Gain: 0.940-0.788 0.152

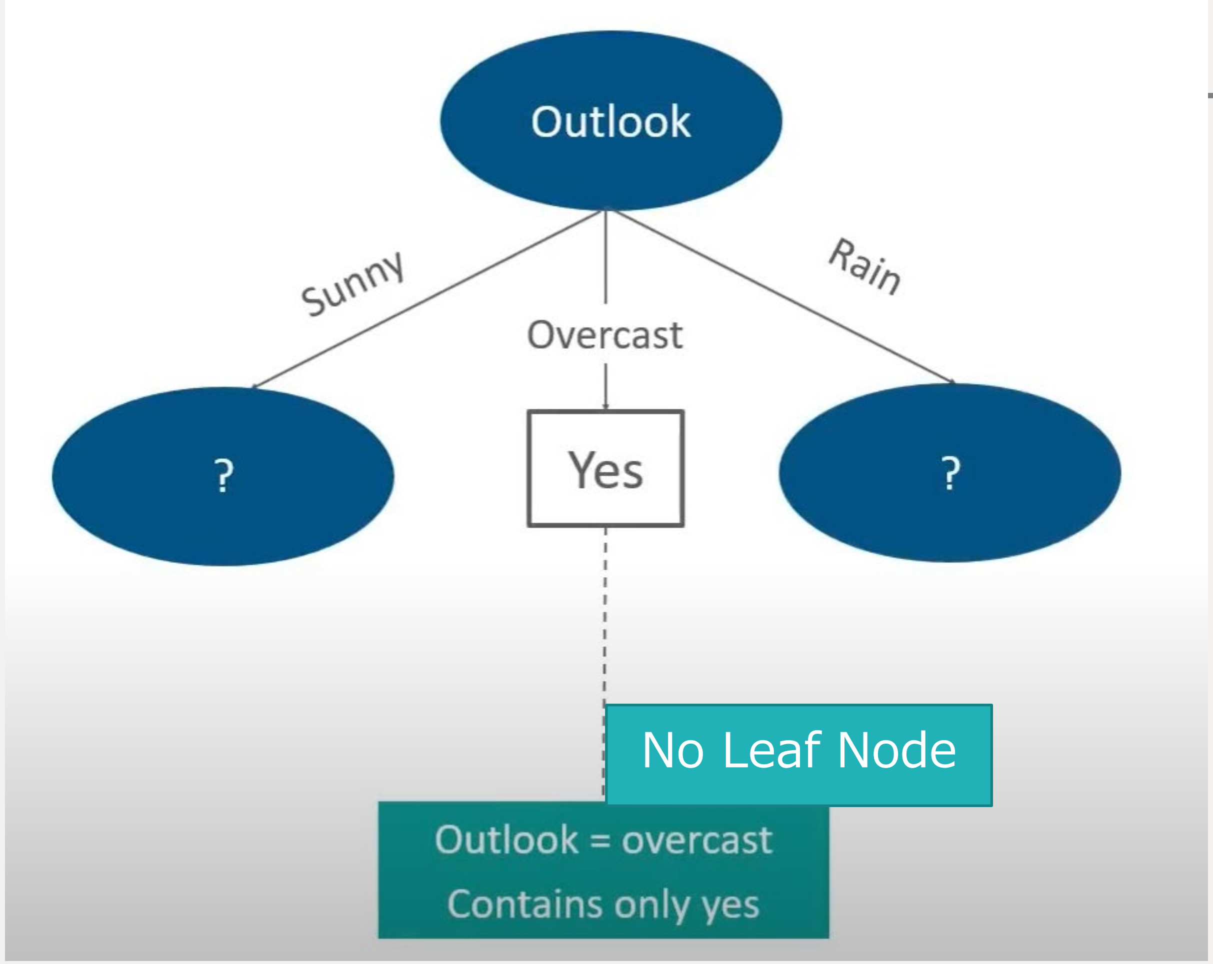
Windy:
Info 0.892
Gain: 0.940-0.982 0.048

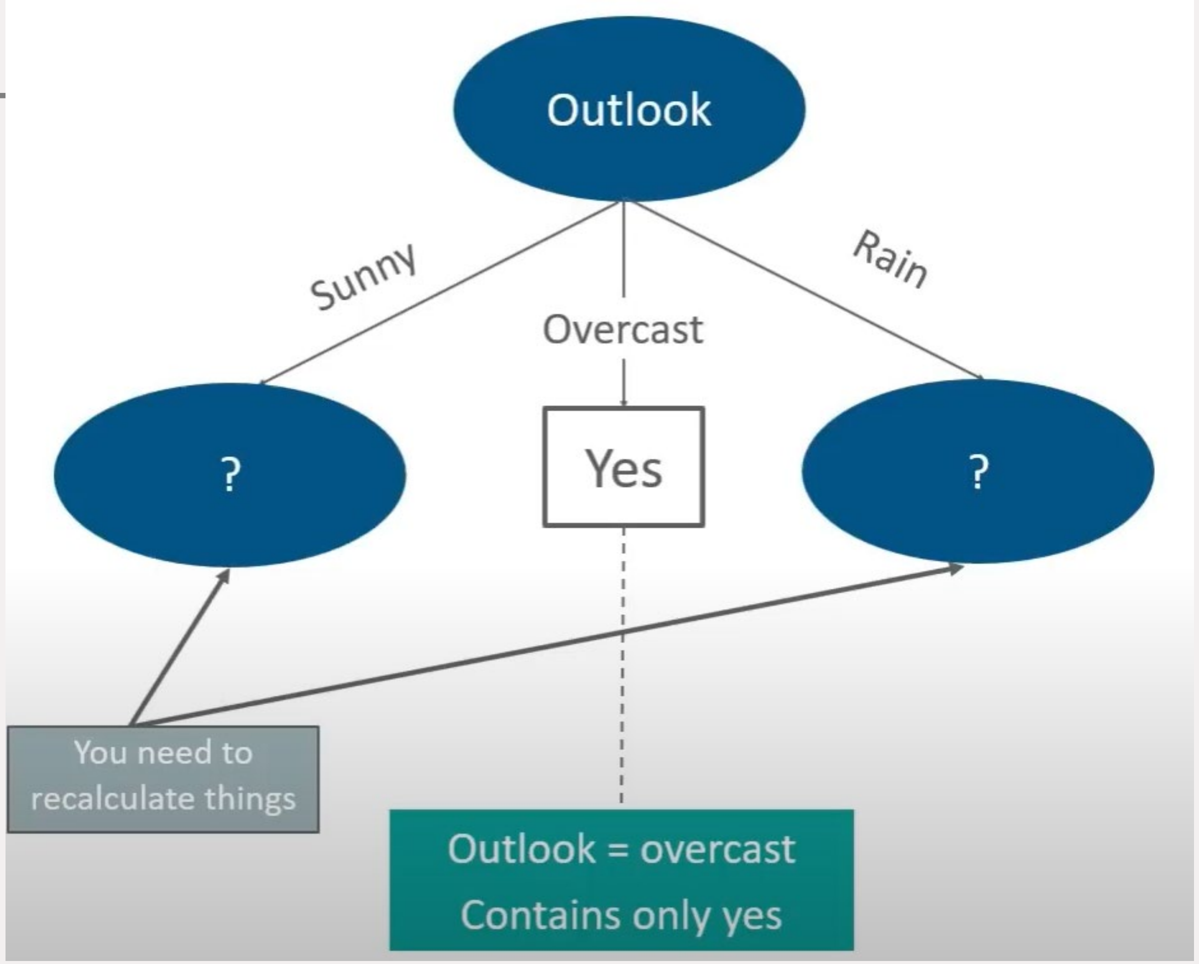
Since Max gain = 0.247,

Outlook is our ROOT Node

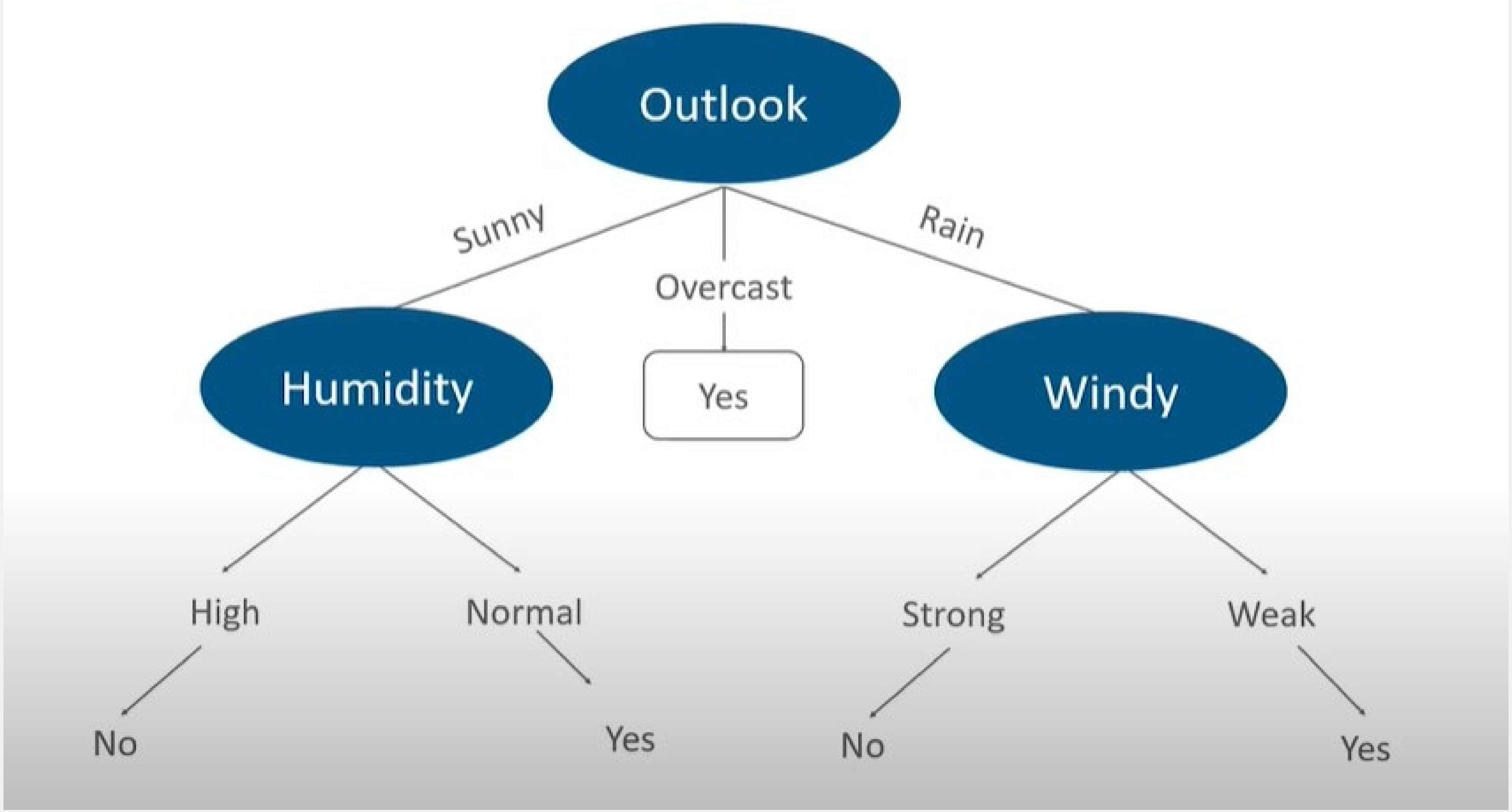
outlook	temp.	humidity	windy	play
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
rainy	cool	normal	false	yes
rainy	cool	normal	true	no
overcast	cool	normal	true	yes
sunny	mild	high	false	no
sunny	cool	normal	false	yes
rainy	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
overcast	hot	normal	false	yes
rainy	mild	high	true	no

Which Node To Select Further?





This Is How Your Complete Tree Will Look Like



What Should I Do To Play - Pruning

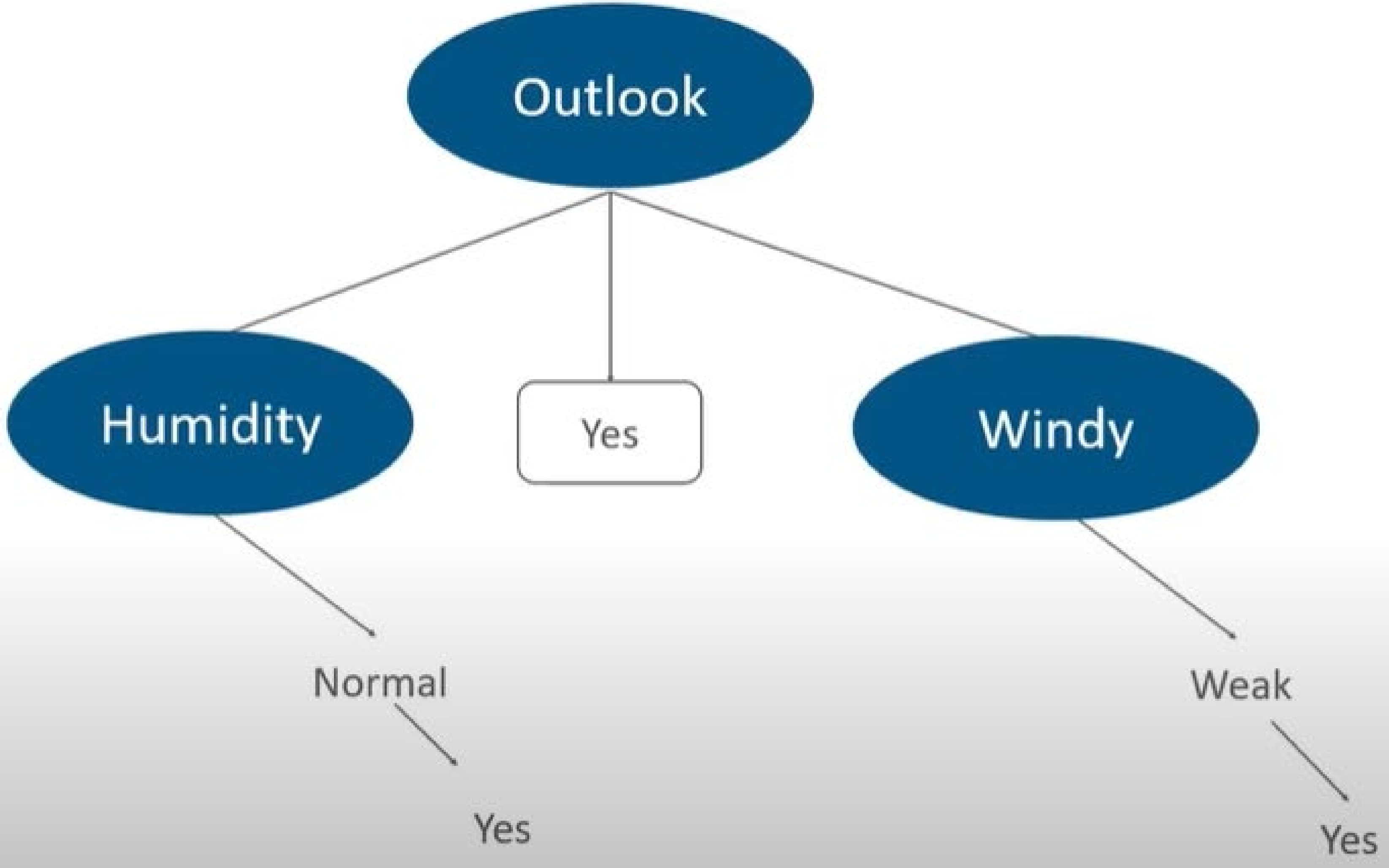
Cutting down the nodes to get optimal solution

Pruning refers to a technique used in machine learning to reduce the size and complexity of a neural network by removing unnecessary or redundant parts of the network.
The goal of pruning is to simplify the network architecture while maintaining or improving its performance.
Pruning can be performed in different ways, such as:

1. **Weight pruning:** This involves removing the weights in the neural network that have the smallest magnitude, as these weights have the least impact on the network's output.
2. **Neuron pruning:** This involves removing entire neurons from the network that are deemed unnecessary or redundant.
3. **Filter pruning:** This involves removing filters from the convolutional layers of a neural network that are not contributing significantly to the network's performance.

Pruning can be performed during the training phase of a neural network or after the network has been trained. When done correctly, pruning can significantly reduce the size of a neural network, making it more efficient in terms of memory and computation requirements, while maintaining or improving its accuracy and performance.

Pruning: Reducing The Complexity



Are Tree based model better than Linear model

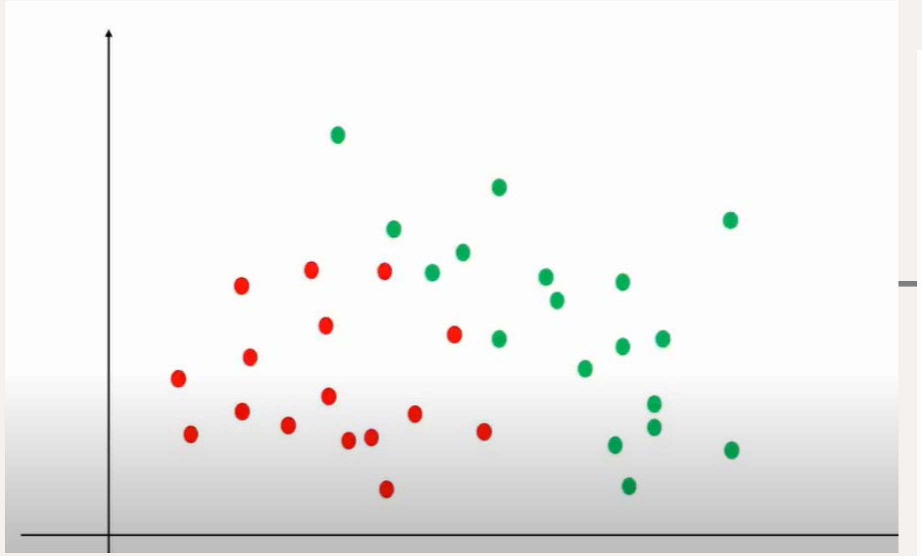
There is no clear-cut answer to whether tree-based models are better than linear models, as it depends on the specific problem and dataset being analyzed. Both types of models have their strengths and weaknesses, and the choice of model ultimately depends on the nature of the problem and the data available.

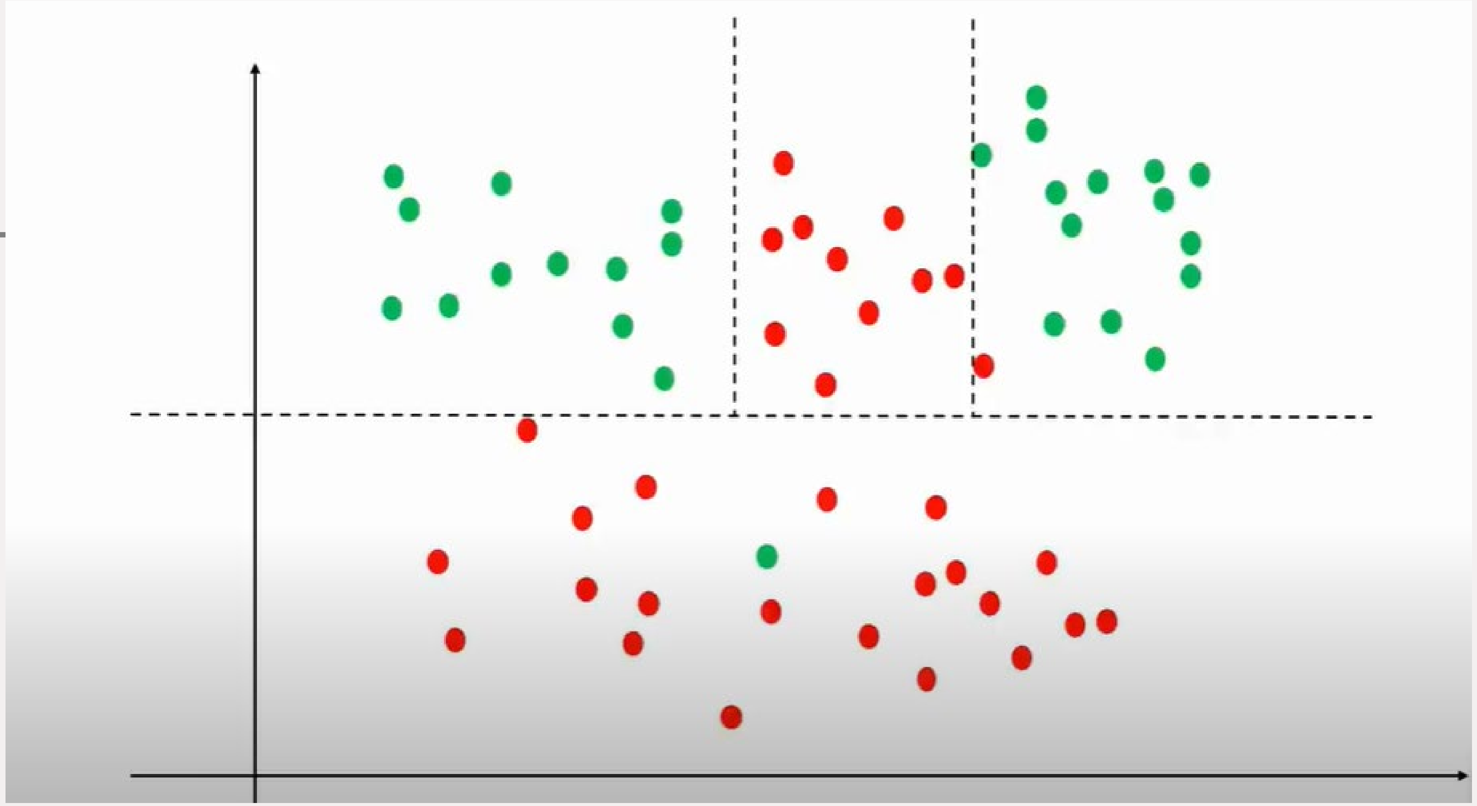
Tree-based models, such as decision trees, random forests, and gradient boosting machines, are powerful because they can model complex, non-linear relationships between variables and are often able to capture interactions and feature importance. They also handle categorical variables well and can work with both numerical and categorical data. However, they can be prone to overfitting if not properly tuned and can become complex and difficult to interpret.

Linear models, on the other hand, are simpler and easier to interpret than tree-based models. They work well when there is a linear relationship between the dependent variable and the independent variables, and can handle large datasets with high dimensional feature spaces. However, they may not capture non-linear relationships between variables as well as tree-based models and may require more feature engineering to capture interactions between variables.

In practice, it's common to try multiple models on a dataset and compare their performance using metrics such as accuracy, precision, recall, F1 score, or mean squared error (MSE). Depending on the dataset and the problem, either tree-based or linear models may perform better. It's also worth noting that hybrid models that combine both linear and tree-based models, such as generalized linear models with decision trees or random forests, can also be effective in certain situations.

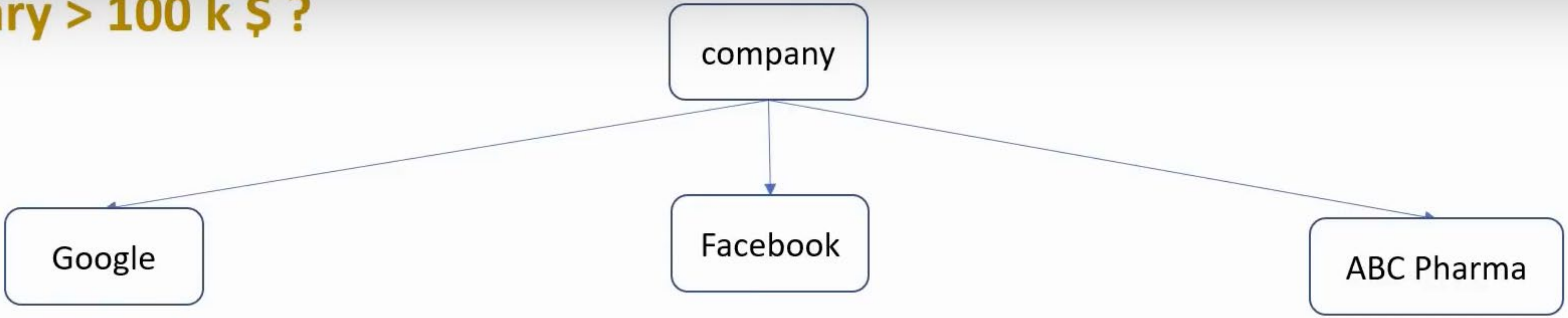
CART





Company	Job	Degree	Salary_more_then_100k
google	sales executive	bachelors	0
google	sales executive	masters	0
google	business manager	bachelors	1
google	business manager	masters	1
google	computer programmer	bachelors	0
google	computer programmer	masters	1
abc pharma	sales executive	masters	0
abc pharma	computer programmer	bachelors	0
abc pharma	business manager	bachelors	0
abc pharma	business manager	masters	1
facebook	sales executive	bachelors	1
facebook	sales executive	masters	1
facebook	business manager	bachelors	1
facebook	business manager	masters	1
facebook	computer programmer	bachelors	1

Salary > 100 k $?



google	sales executive	bachelors
google	sales executive	masters
google	business manager	bachelors
google	business manager	masters
google	computer programmer	bachelors
google	computer programmer	masters

?

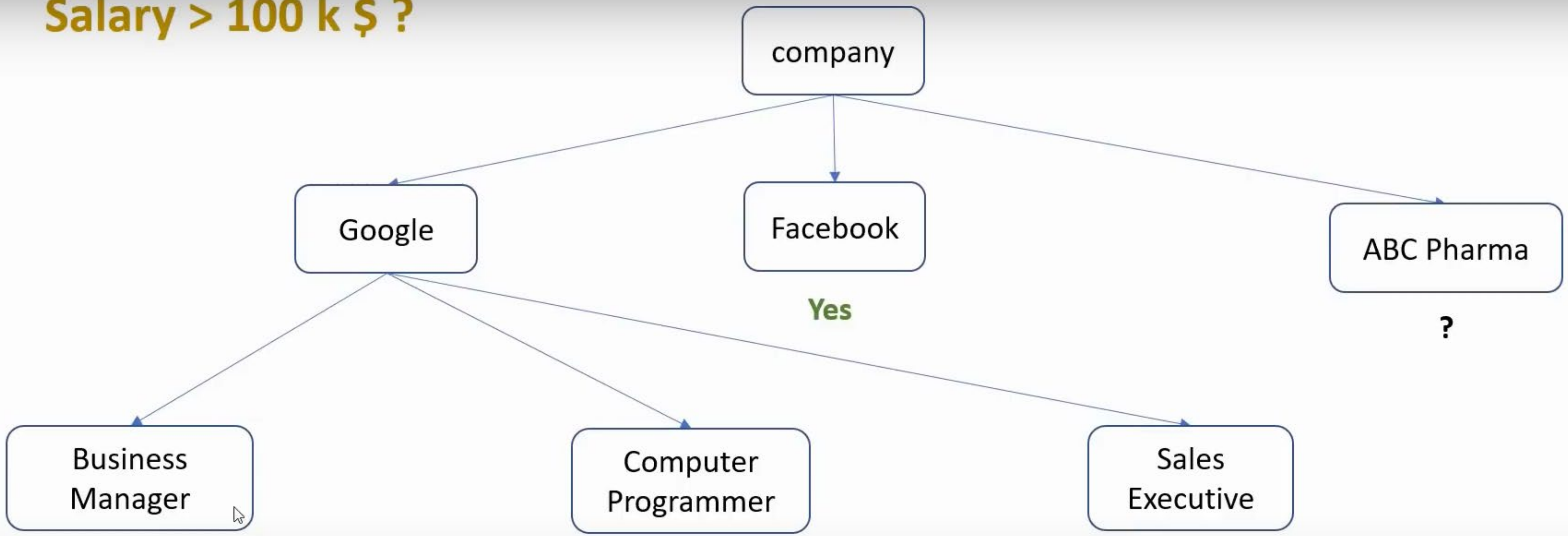
facebook	sales executive	bachelors
facebook	sales executive	masters
facebook	business manager	bachelors
facebook	business manager	masters
facebook	computer programmer	bachelors
facebook	computer programmer	masters

Yes

abc pharma	sales executive	masters
abc pharma	computer programmer	bachelors
abc pharma	business manager	bachelors
abc pharma	business manager	masters

?

Salary > 100 k $?



google	business manager	bachelors
google	business manager	masters

Yes

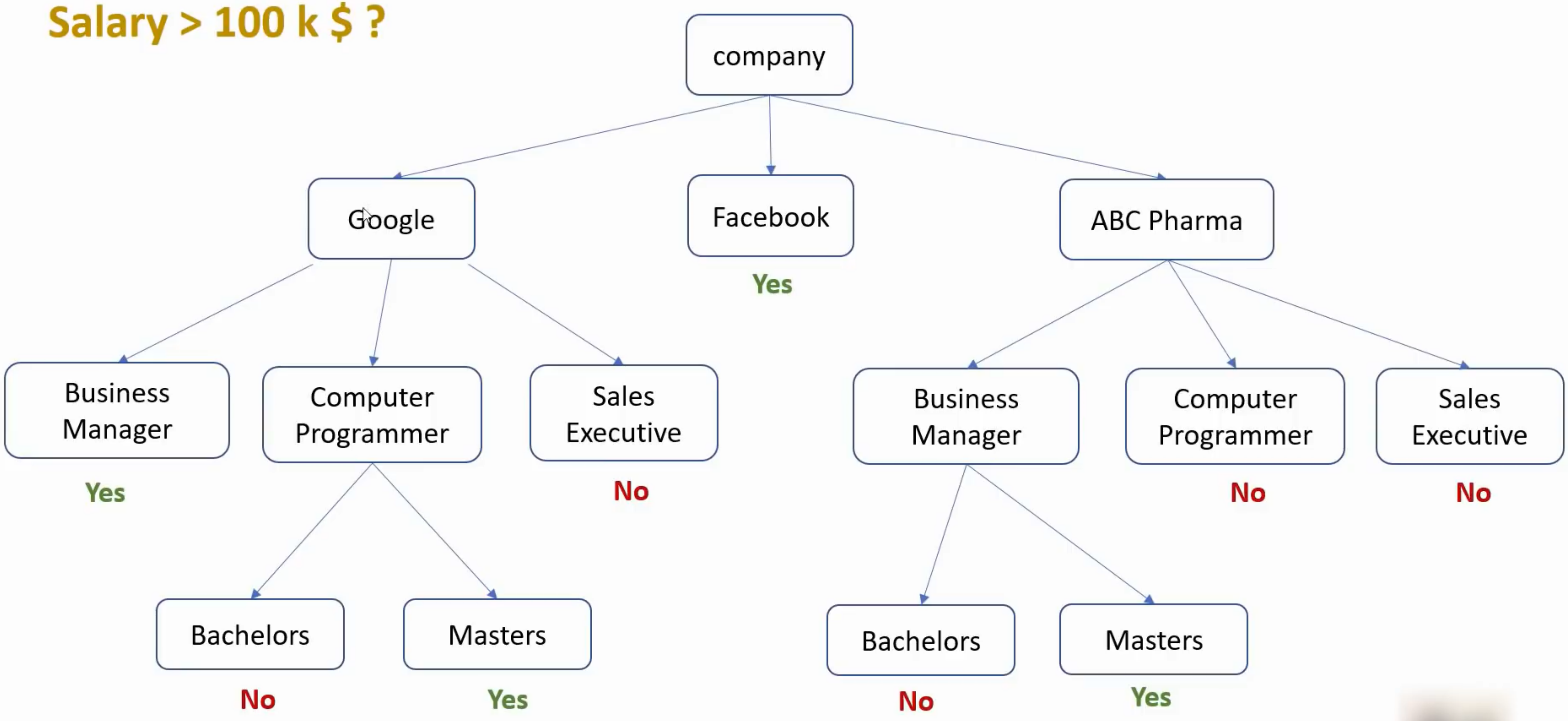
google	computer programmer	bachelors
google	computer programmer	masters

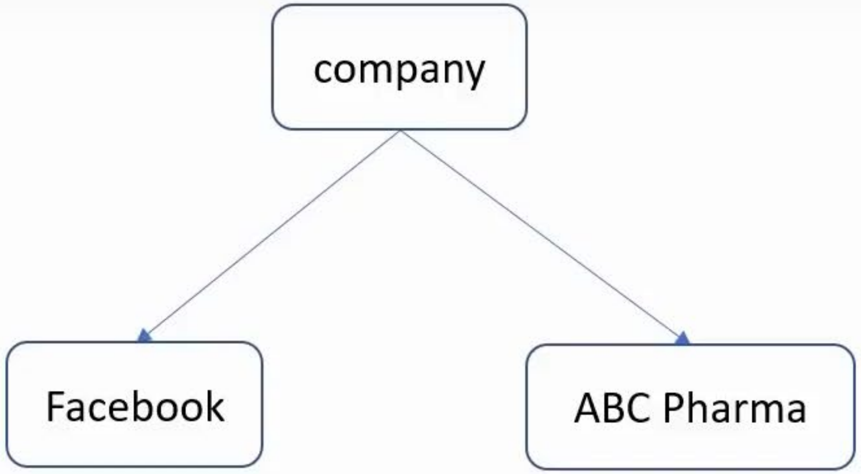
?

google	sales executive	bachelors
google	sales executive	masters

No

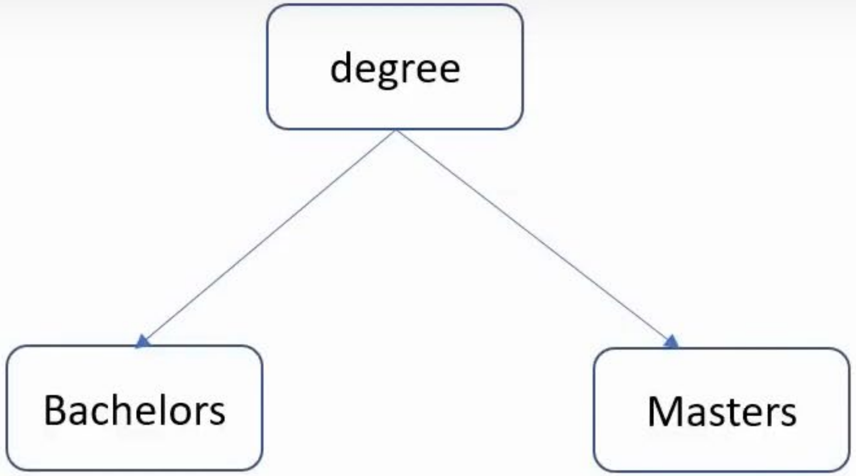
Salary > 100 k $?


company
Google
Facebook
ABC Pharma
Yes
Business Manager
Computer Programmer
Sales Executive
Business Manager
Computer Programmer
Sales Executive
Yes
No
No
No
Bachelors
Masters
Bachelors
Masters
No
Yes
No
Yes



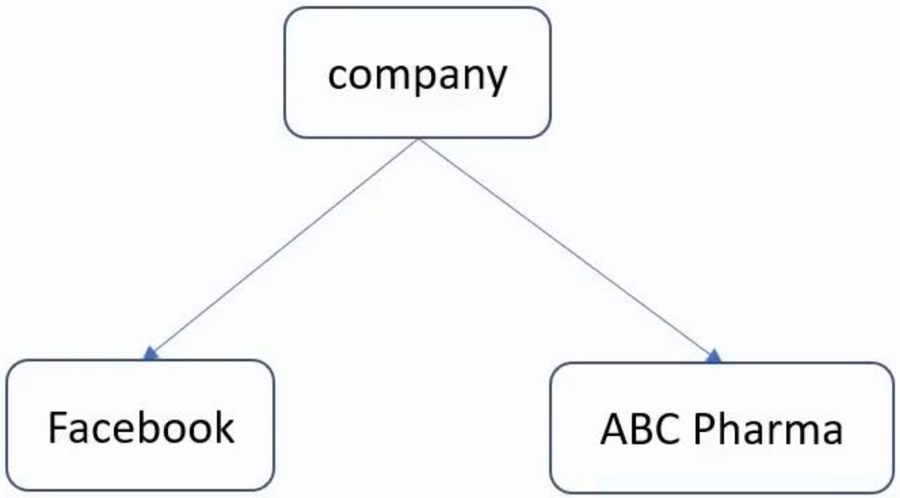
facebook	sales executive	bachelors
facebook	sales executive	masters
facebook	business manager	bachelors
facebook	business manager	masters
facebook	computer programmer	bachelors
facebook	computer programmer	masters

abc pharma	sales executive	masters
abc pharma	computer programmer	bachelors
abc pharma	business manager	bachelors
abc pharma	business manager	masters



google	sales executive	bachelors
google	business manager	bachelors
google	computer programmer	bachelors
abc pharma	computer programmer	bachelors
abc pharma	business manager	bachelors
facebook	sales executive	bachelors
facebook	business manager	bachelors
facebook	computer programmer	bachelors

google	sales executive	masters
google	business manager	masters
google	computer programmer	masters
abc pharma	sales executive	masters
abc pharma	business manager	masters
facebook	sales executive	masters
facebook	business manager	masters
facebook	computer programmer	masters

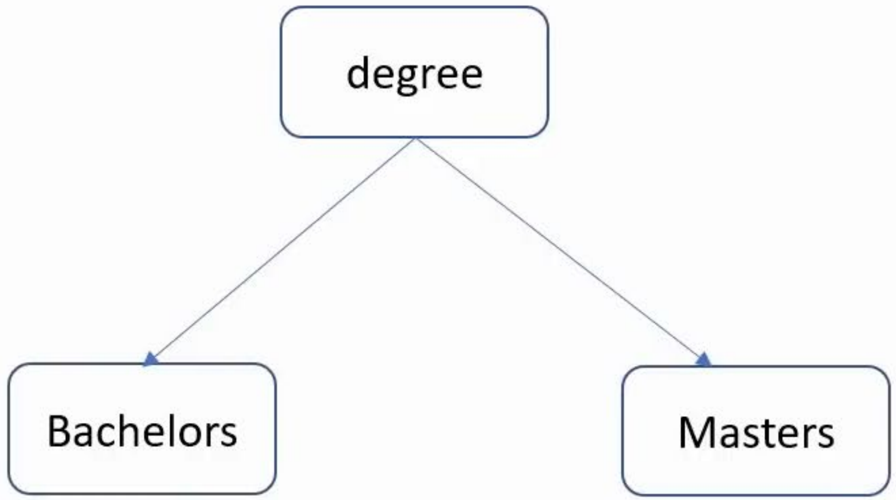


facebook	sales executive	bachelors
facebook	sales executive	masters
facebook	business manager	bachelors
facebook	business manager	masters
facebook	computer programmer	bachelors
facebook	computer programmer	masters

6 / 0 (low entropy)

abc pharma	sales executive	masters
abc pharma	computer programmer	bachelors
abc pharma	business manager	bachelors
abc pharma	business manager	masters

1 / 3

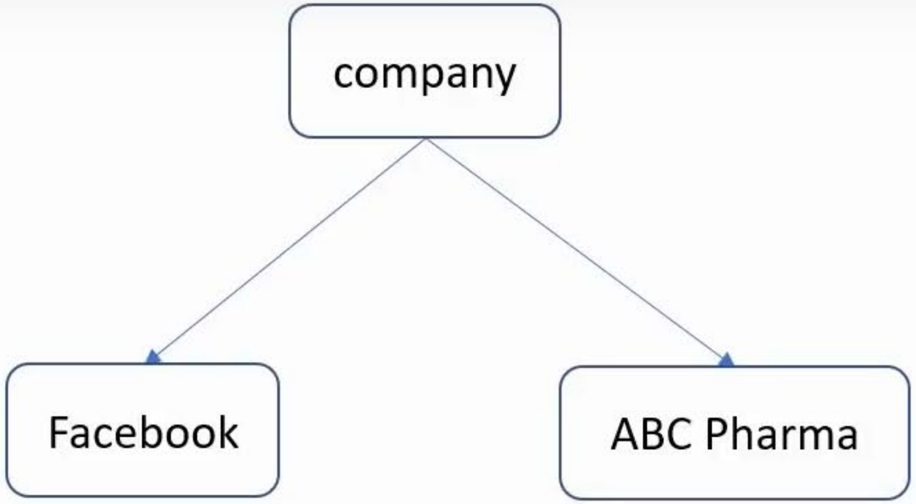


google	sales executive	bachelors
google	business manager	bachelors
google	computer programmer	bachelors
abc pharma	computer programmer	bachelors
abc pharma	business manager	bachelors
facebook	sales executive	bachelors
facebook	business manager	bachelors
facebook	computer programmer	bachelors

4 / 4 (high entropy)

google	sales executive	masters
google	business manager	masters
google	computer programmer	masters
abc pharma	sales executive	masters
abc pharma	business manager	masters
facebook	sales executive	masters
facebook	business manager	masters
facebook	computer programmer	masters

6 / 2



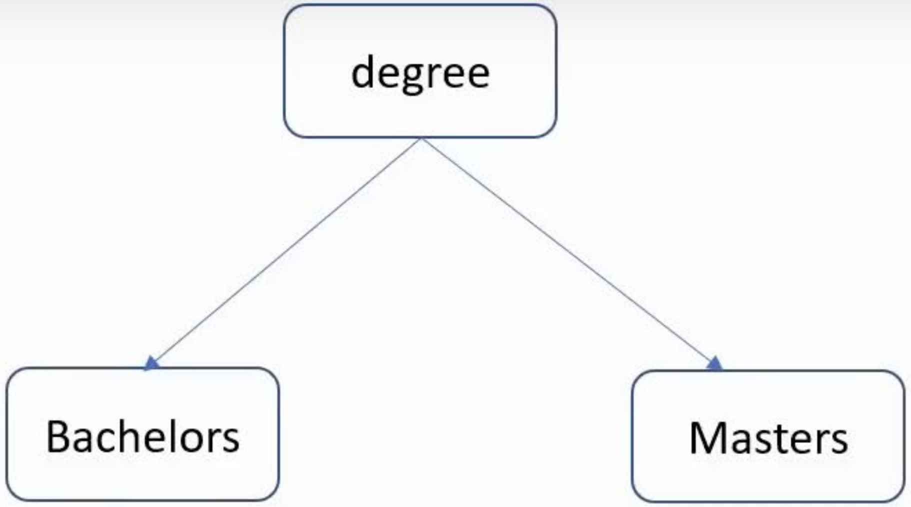
facebook	sales executive	bachelors
facebook	sales executive	masters
facebook	business manager	bachelors
facebook	business manager	masters
facebook	computer programmer	bachelors
facebook	computer programmer	masters

6 / 0 (low entropy)

abc pharma	sales executive	masters
abc pharma	computer programmer	bachelors
abc pharma	business manager	bachelors
abc pharma	business manager	masters

1 / 3

High Information Gain



google	sales executive	bachelors
google	business manager	bachelors
google	computer programmer	bachelors
abc pharma	computer programmer	bachelors
abc pharma	business manager	bachelors
facebook	sales executive	bachelors
facebook	business manager	bachelors
facebook	computer programmer	bachelors

4 / 4 (high entropy)

google	sales executive	masters
google	business manager	masters
google	computer programmer	masters
abc pharma	sales executive	masters
abc pharma	business manager	masters
facebook	sales executive	masters
facebook	business manager	masters
facebook	computer programmer	masters

6 / 2

Low Information Gain

Gini Impurity