What is a Decision Tree?

A decision tree is a non-parametric supervised learning algorithm for classification and regression tasks. It has a hierarchical tree structure consisting of a root node, branches, internal nodes, and leaf nodes.

How decision tree algorithms work?

1.**Starting at the Root:** Representing the entire dataset.

2.**Asking the Best Questions:** It looks for the most important feature or question that splits the data into the most distinct groups

3.**Branching Out:** Based on the answer to that question, it divides the data into smaller subsets, creating new branches.

4.**Repeating the Process:** The algorithm continues asking questions and splitting the data at each branch until it reaches the final “leaf nodes

Attribute Selection Measures

While implementing a Decision tree, the main issue arises that how to select the best attribute for the root node and for sub-nodes. So, to solve such problems there is a technique which is called as Attribute selection measure or ASM.

- **Information Gain**
- **Gini Index**

Information gain is the measurement of changes in entropy after the segmentation of a dataset based on an attribute.

A decision tree algorithm always tries to maximize the value of information gain, and a node/attribute having the highest information gain is split first. It can be calculated using the below formula:

Information Gain= Entropy(S)- [(Weighted Avg) *Entropy(each feature)

Entropy: Entropy is a metric to measure the impurity in a given attribute. It specifies randomness in data. Entropy can be calculated as:

$$\mathbf{Entropy(s)= -P(yes)log2\ P(yes)- P(no)\ log2\ P(no)}$$

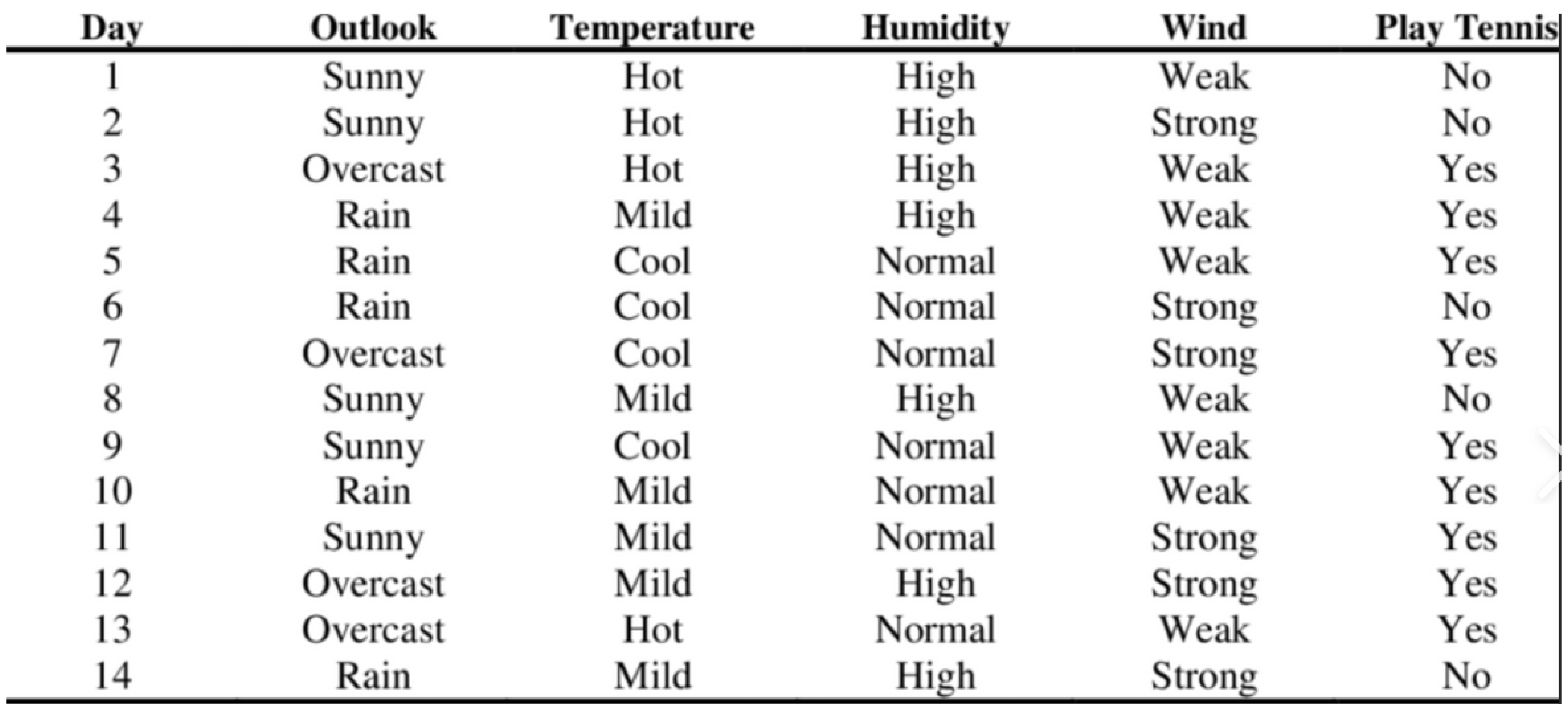
- S= Total number of samples
- P(yes)= probability of yes
- P(no)= probability of no

2. Gini Index:

Gini index is a measure of impurity or purity used while creating a decision tree

$$\mathbf{Gini\ Index= 1- \sum jPj2}$$

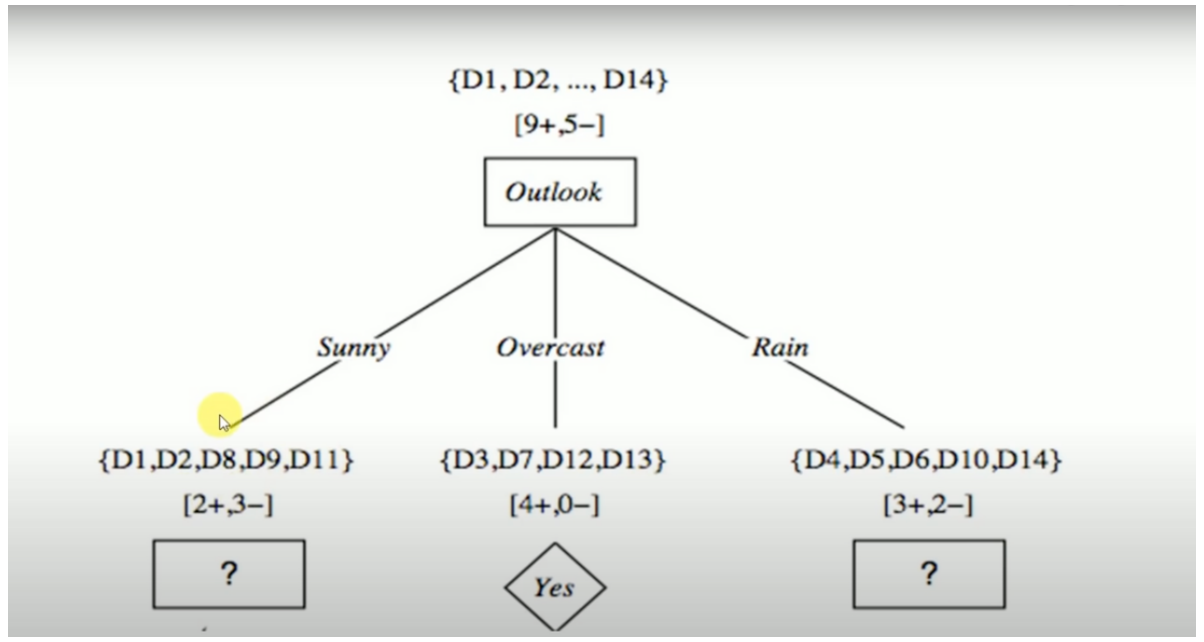
Example:



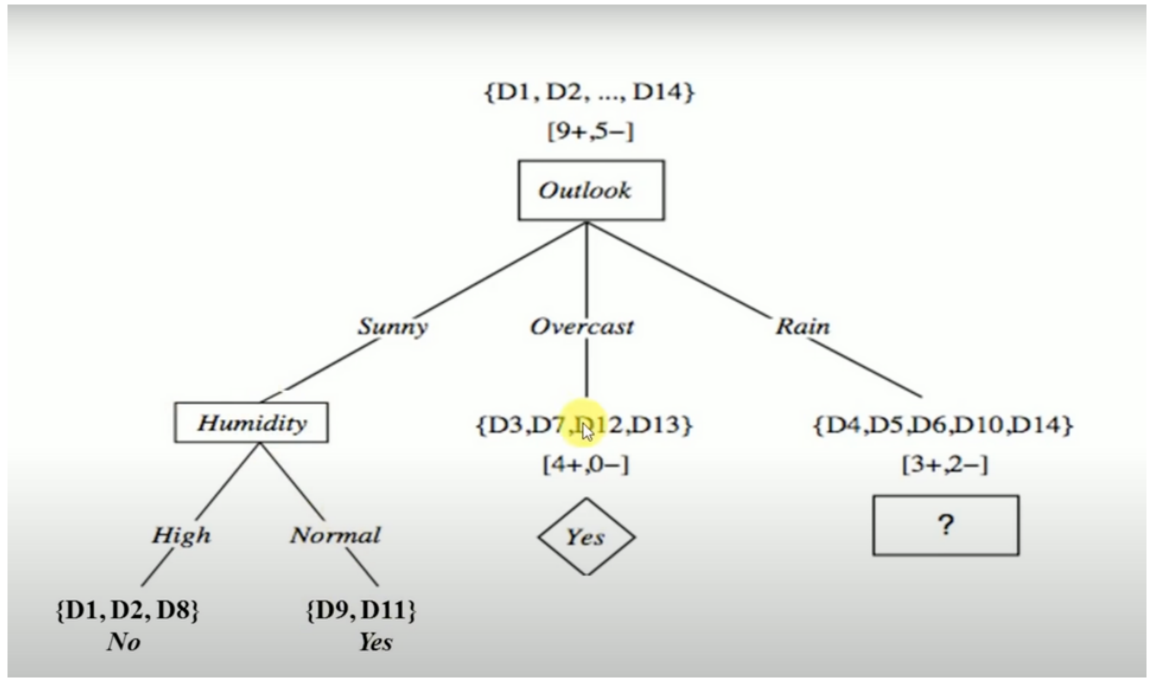
Day	Outlook	Temperature	Humidity	Wind	Play Tennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Step 1: We find the IG of each Feature.

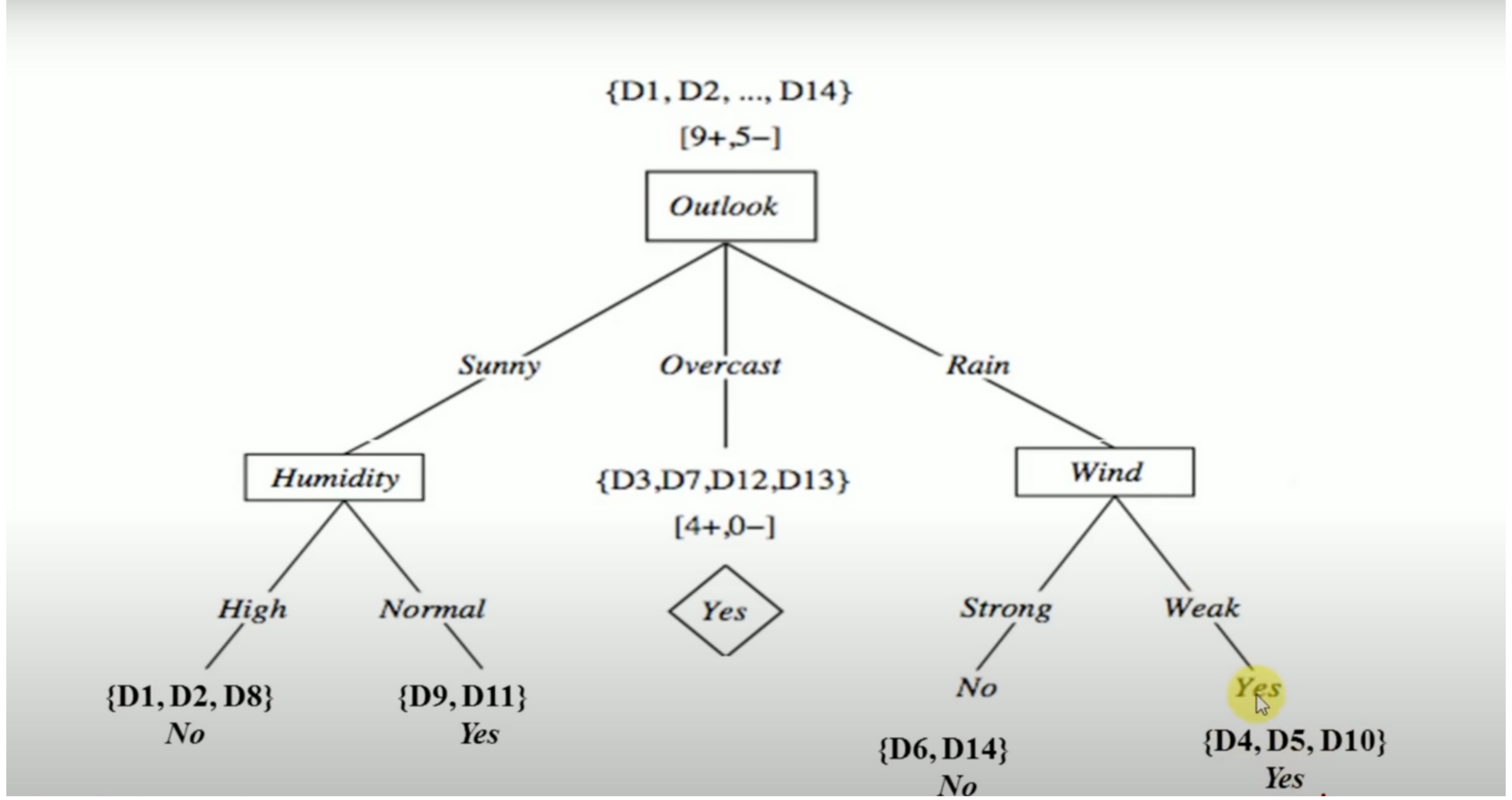
Step 2: The feature with highest IG consider that as root node



Step 3:


{D1, D2, ..., D14}
[9+,5−]
Outlook
Sunny
Overcast
Rain
Humidity
{D3,D7,D12,D13}
[4+,0−]
{D4,D5,D6,D10,D14}
[3+,2−]
High
Normal
Yes
?
{D1, D2, D8}
No
{D9, D11}
Yes

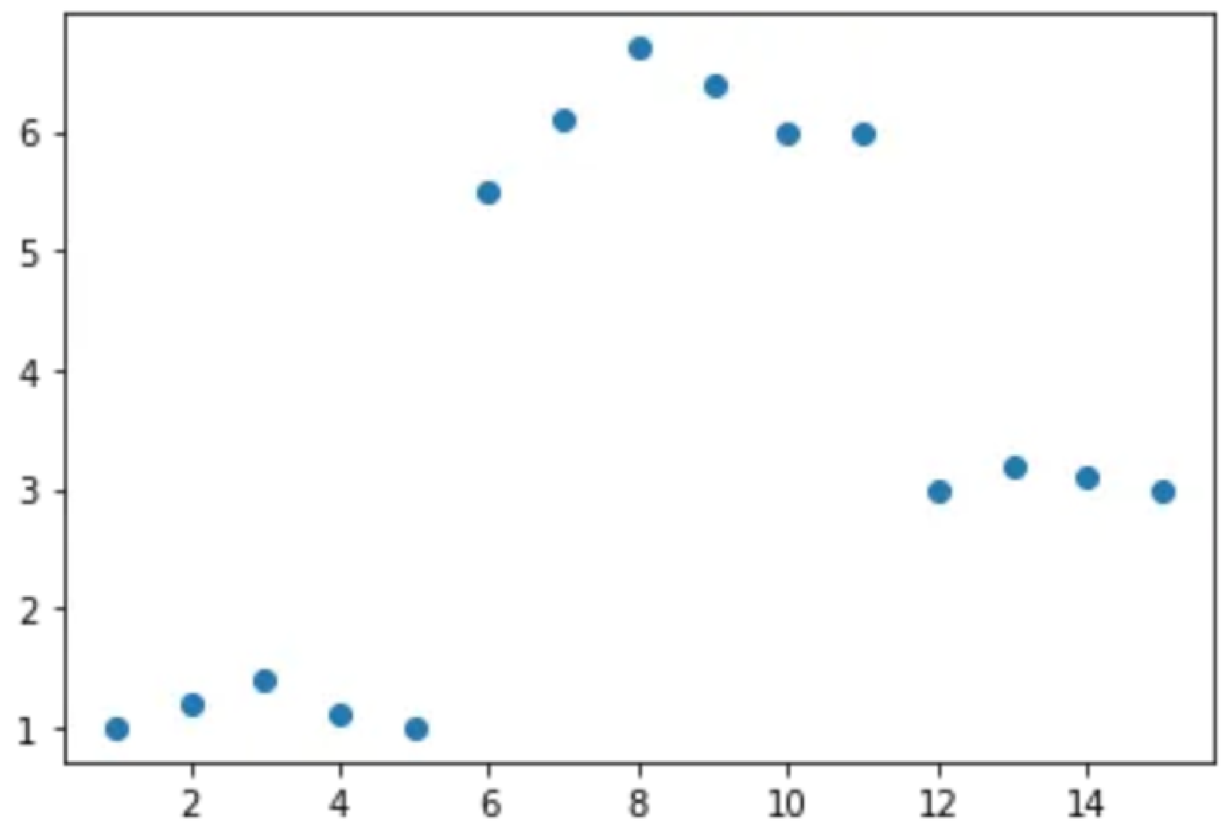
Step 4:

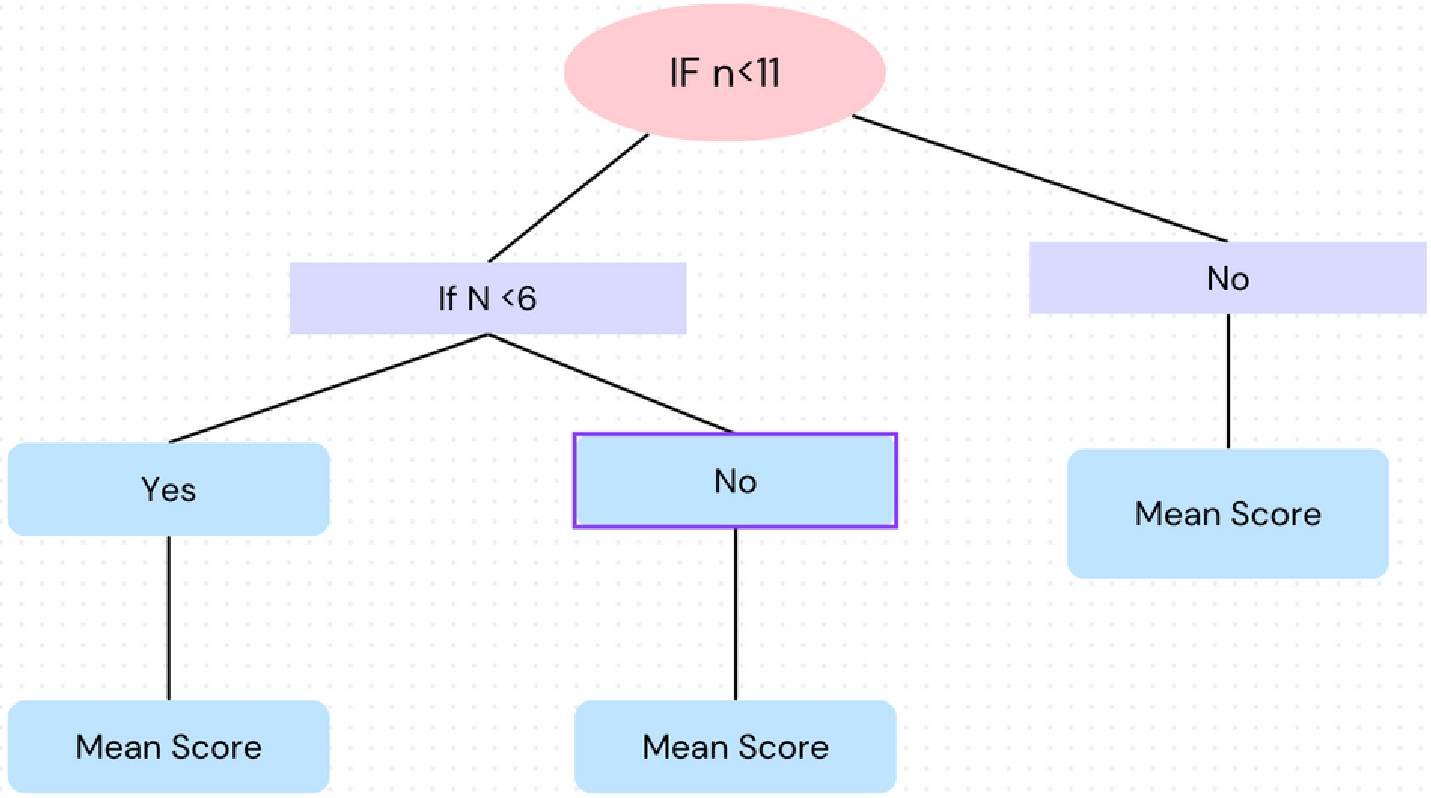

{D1, D2, ..., D14}
[9+,5−]
Outlook
Sunny
Overcast
Rain
Humidity
{D3,D7,D12,D13}
[4+,0−]
Wind
High
Normal
Yes
Strong
Weak
{D1, D2, D8}
No
{D9, D11}
Yes
No
Yes
{D6, D14}
No
{D4, D5, D10}
Yes

Regression Tree

Why we have to use Regression tree ?

Let’s consider a dataset where we have 2 variables, as shown below



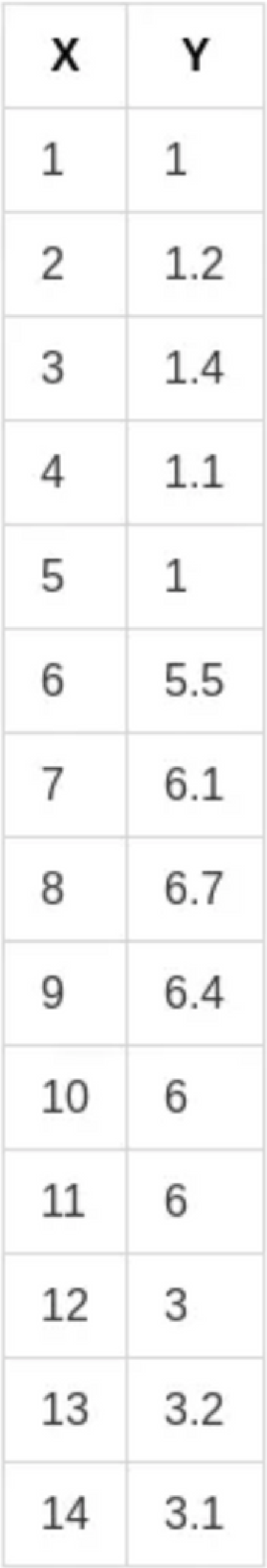


How we have to find the splitting ?

Step 1: take the average of the first 2 rows in variable X (which is (1+2)/2 = 1.5 according to the given dataset). Divide the dataset into 2 parts (Part A and Part B) , separated by x < 1.5 and X ≥ 1.5.

Now, Part A consist only of one point, which is the first row (1,1) and all the other points are in Part — B. Now, take the average of all the Y values in Part A and average of all Y values in Part B separately. These 2 values are the predicted output of the decision tree for x < 1.5 and x ≥ 1.5 respectively. Using the predicted and original values, calculate the mean square error and note it down.

Step 2: This process is repeated for the third 2 numbers, the fourth 2 numbers, the 5th, 6th, 7th till n-1th 2 numbers (where n is the number of records or rows in the dataset)



X	Y
1	1
2	1.2
3	1.4
4	1.1
5	1
6	5.5
7	6.1
8	6.7
9	6.4
10	6
11	6
12	3
13	3.2
14	3.1

Step 3: Now that we have n-1 mean squared errors calculated , we need to choose the point at which we are going to split the dataset. and that point is the point, which resulted in the lowest mean squared error on splitting at it. In this case, the point is x=5.5. Hence the tree will be split into 2 parts. x<5.5 and x≥ 5.5. The Root node is selected this way and the data points that go towards the left child and right child of the root node are further recursively exposed to the same algorithm for further splitting.

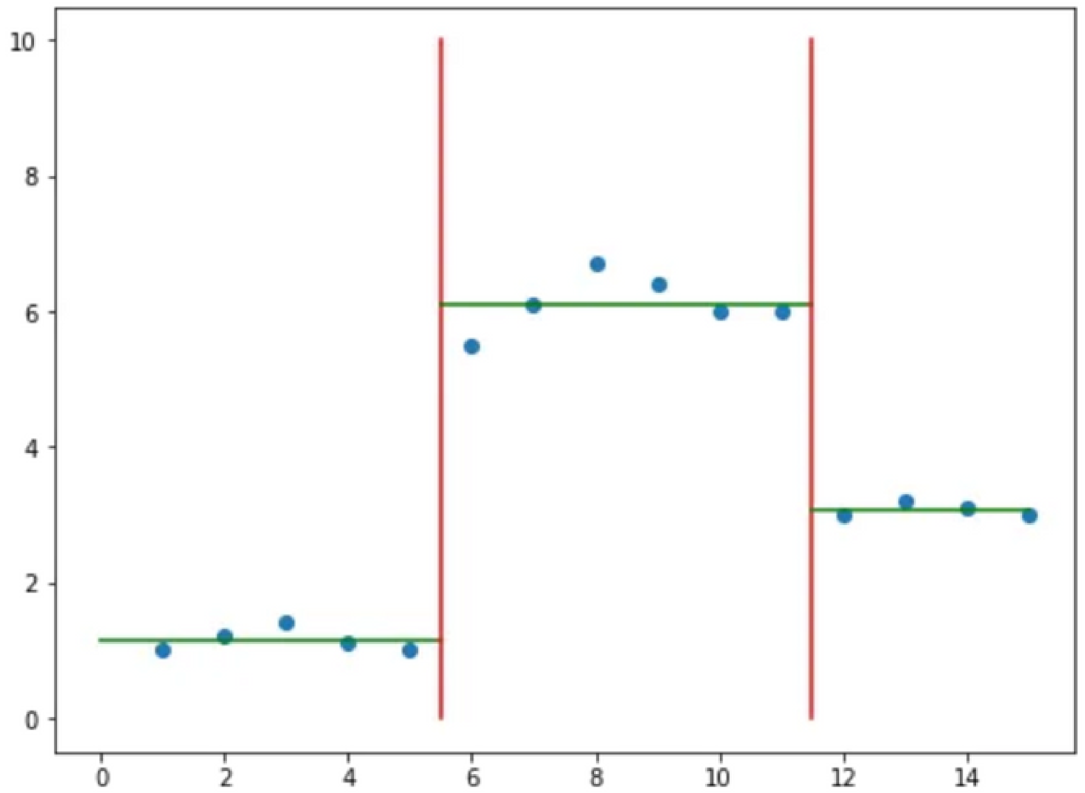
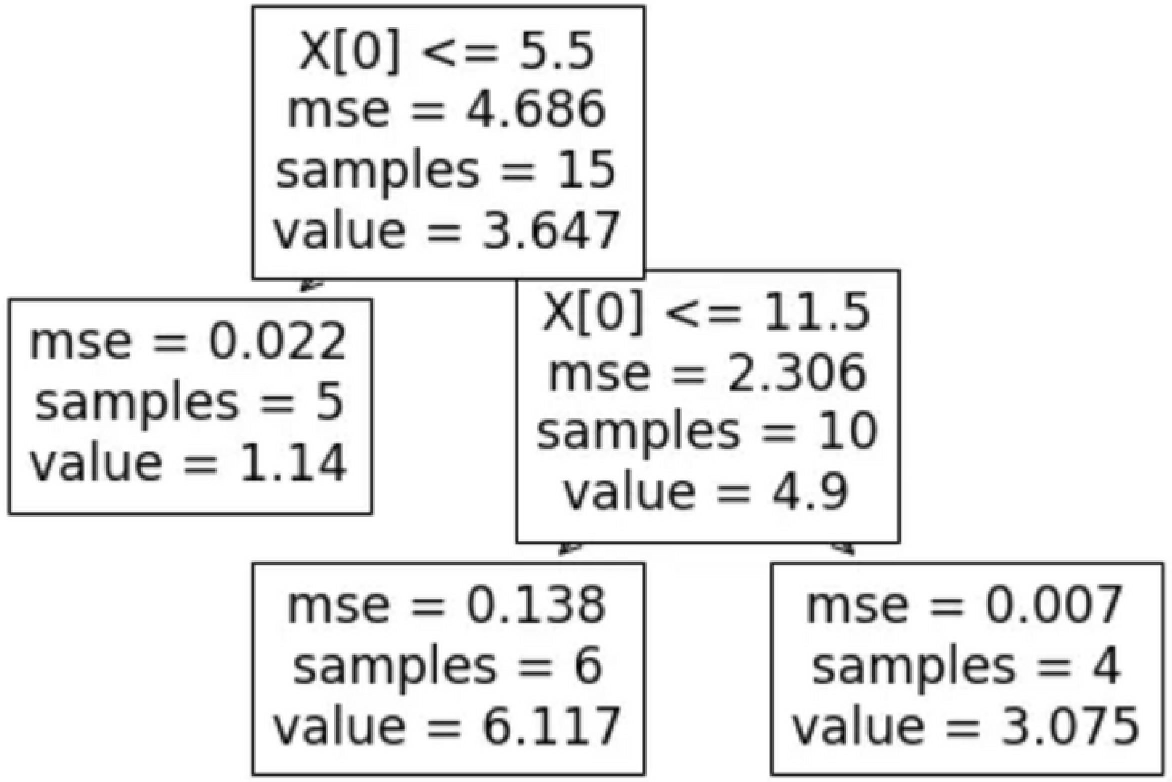


fig 3.2: The Decision Boundary



X[0] <= 5.5
mse = 4.686
samples = 15
value = 3.647

mse = 0.022
samples = 5
value = 1.14

X[0] <= 11.5
mse = 2.306
samples = 10
value = 4.9

mse = 0.138
samples = 6
value = 6.117

mse = 0.007
samples = 4
value = 3.075

fig 3.1: The resultant Decision Tree

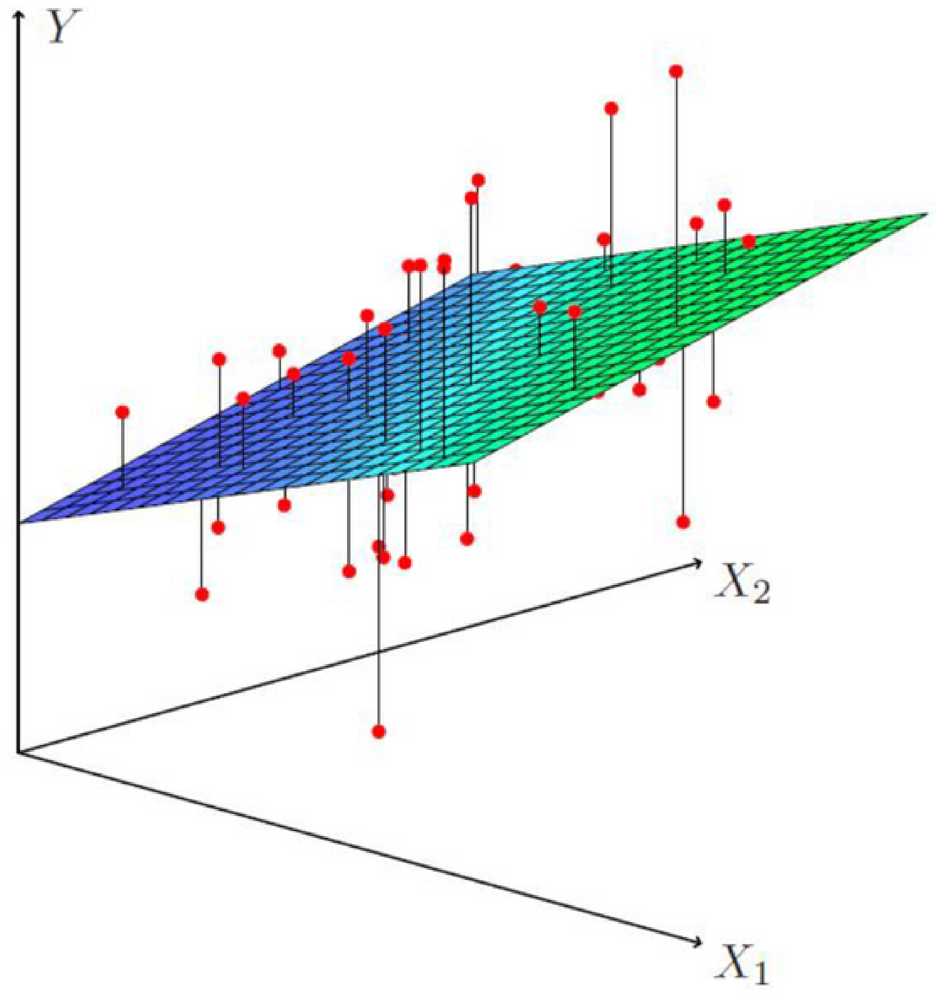
***Also we have to set a Threshold value to overcome overfitting**

What happens when there are multiple independent variables ?

Let us consider that there are 3 variables similar to the independent variable X

At each node, All the 3 variables would go through the same process as what X went through in the above example. The data would be sorted based on the 3 variables separately.

The points that minimises the mse are calculated for all the 3 variables. out of the 3 variables and the points calculated for them, the one that has the least mse would be chosen.



Pruning: Getting an Optimal Decision tree

Pruning is a process of deleting the unnecessary nodes from a tree in order to get the optimal decision tree.

- **Post Pruning**
- **Pre Pruning**