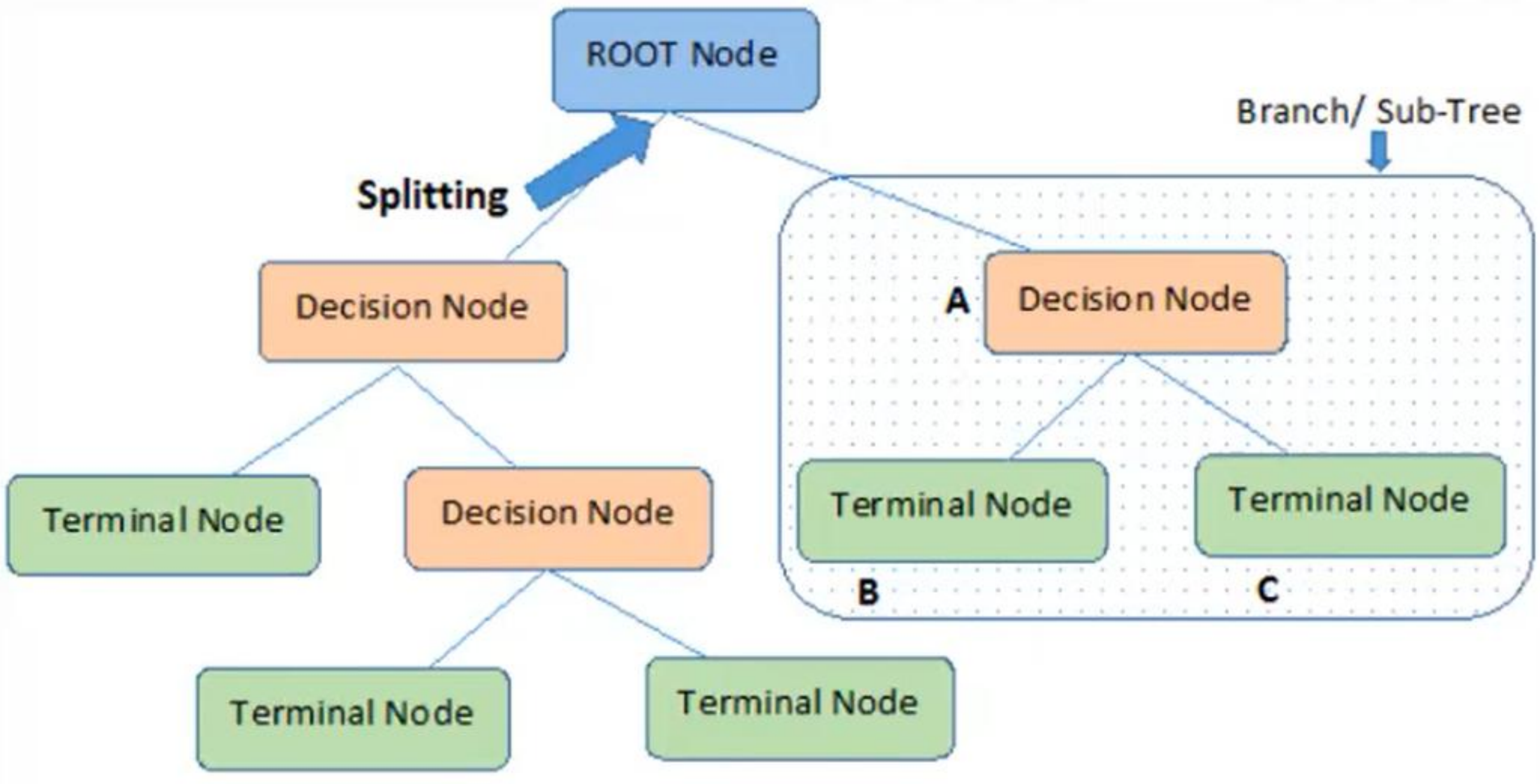
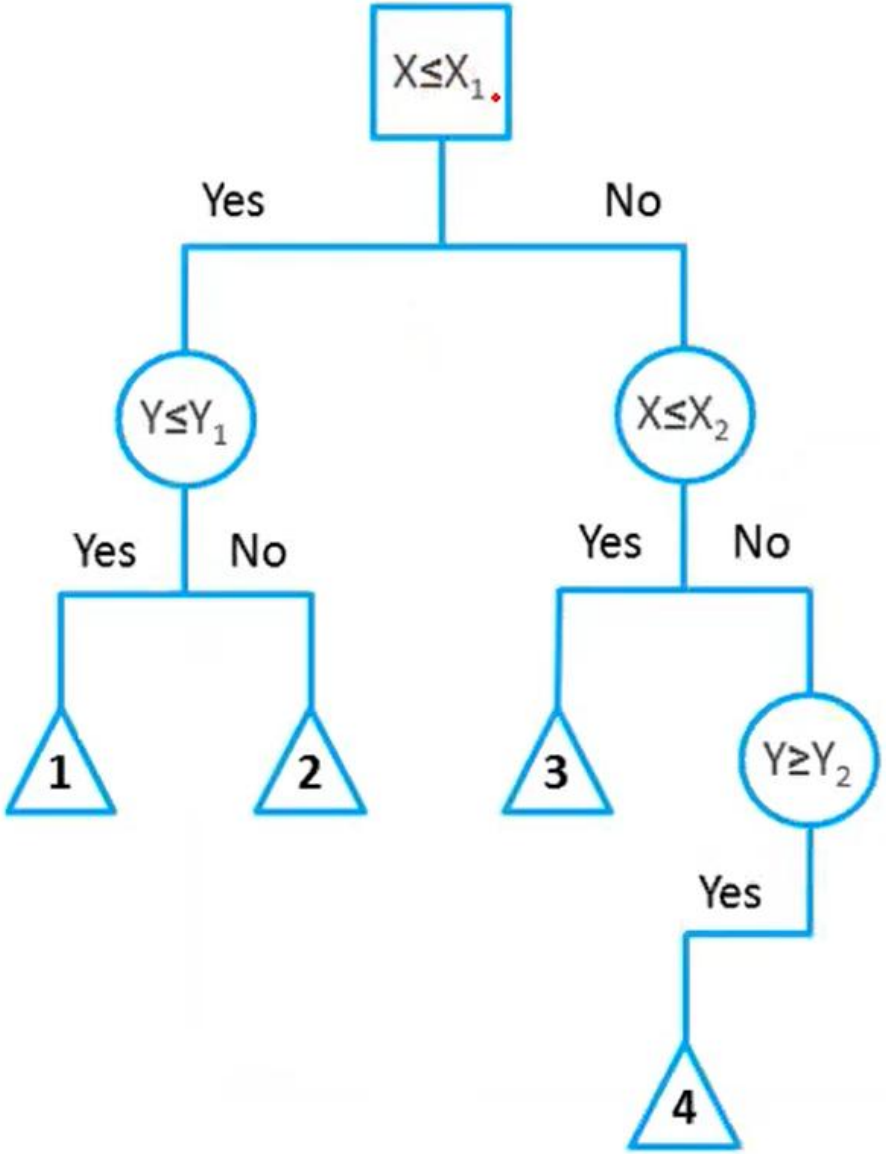
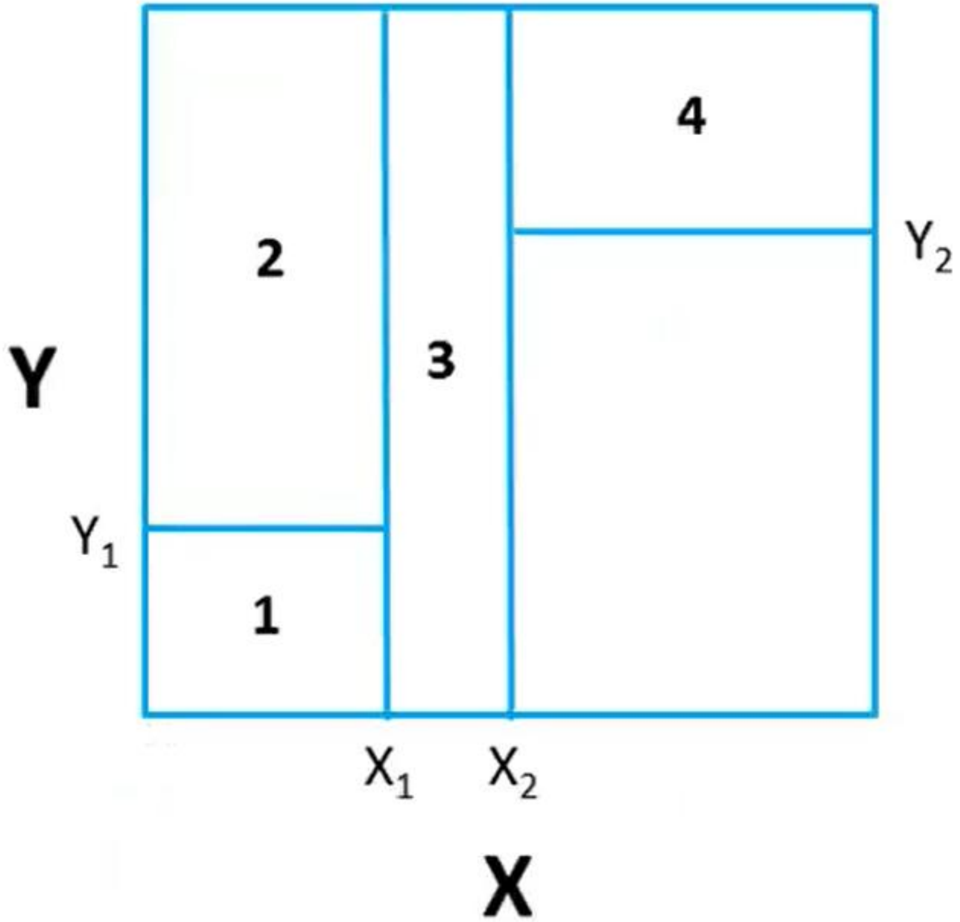
Decision Tree



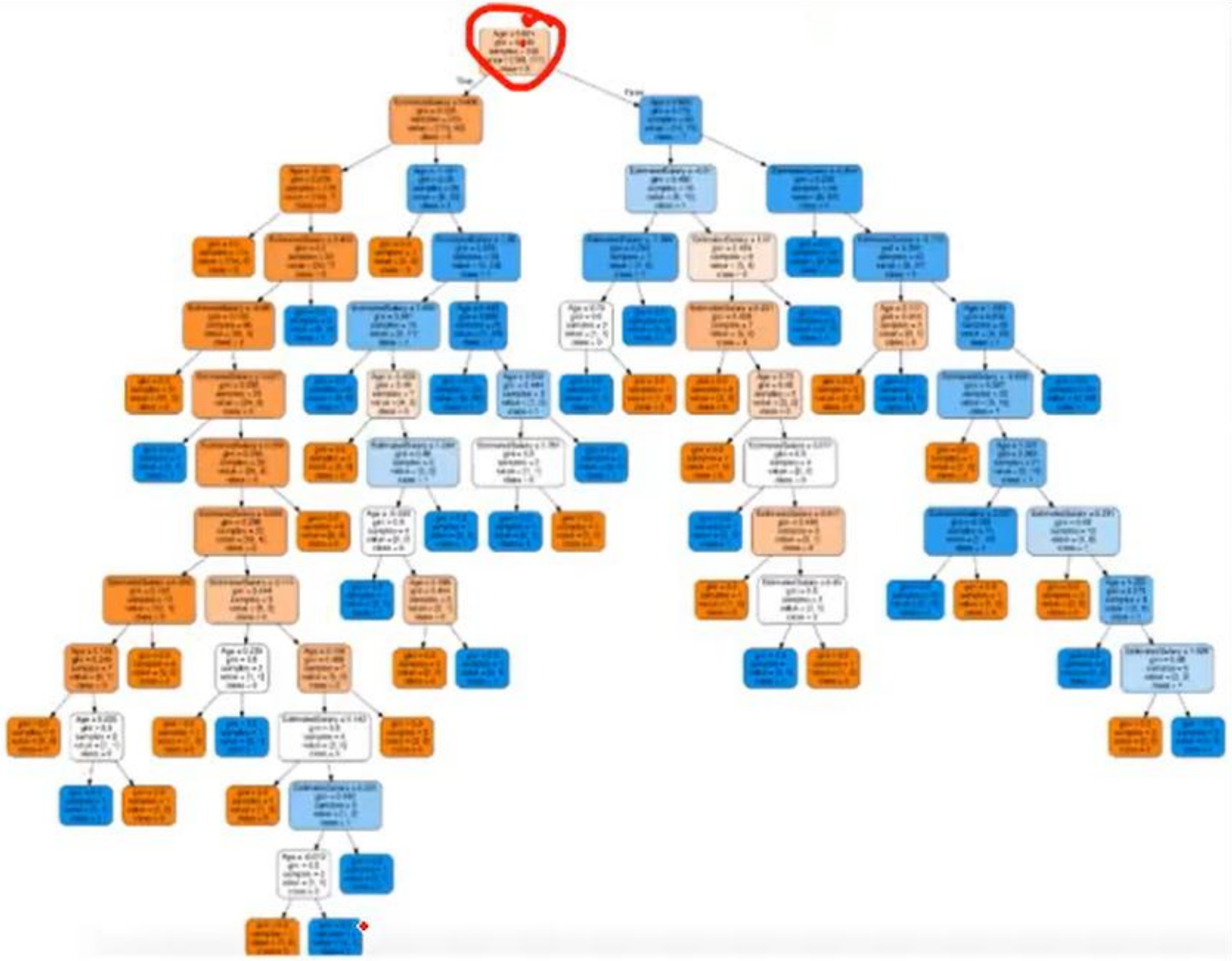
Note:- A is parent node of B and C.

Decision Tree





Decision Tree



Decision Tree.

Decision Tree

Important Terminology related to Decision Trees

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

2.Splitting: It is a process of dividing a node into two or more sub-nodes.

3.Decision Node: When a sub-node splits into further sub-nodes, then it is called the decision node.

4.Leaf / Terminal Node: Nodes do not split is called Leaf or Terminal node.

5.Pruning: When we remove sub-nodes of a decision node, this process is called pruning. You can say the opposite process of splitting.

6.Branch / Sub-Tree: A subsection of the entire tree is called branch or sub-tree.

7.Parent and Child Node: A node, which is divided into sub-nodes is called a parent node of sub-nodes whereas sub-nodes are the child of a parent node.

How to select the best variable from the dataset for Root Node

Randomness in the information	Effective Split of Target variable	Cost Function	Larger root Value
Entropy,	**Information gain,**	**Gini index,**	**Gain Ratio,**

Based on Variance

Reduction in Variance

How to select the best variable from the dataset for Root Node

Randomness in the information → **Entropy**

If ,
Entropy is larger→ Randomness is high → perfectly will not able to predict and Vice versa

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

$$E(T, X) = \sum_{c \in X} P(c) E(c)$$

Play Golf	
Yes	No
9	5

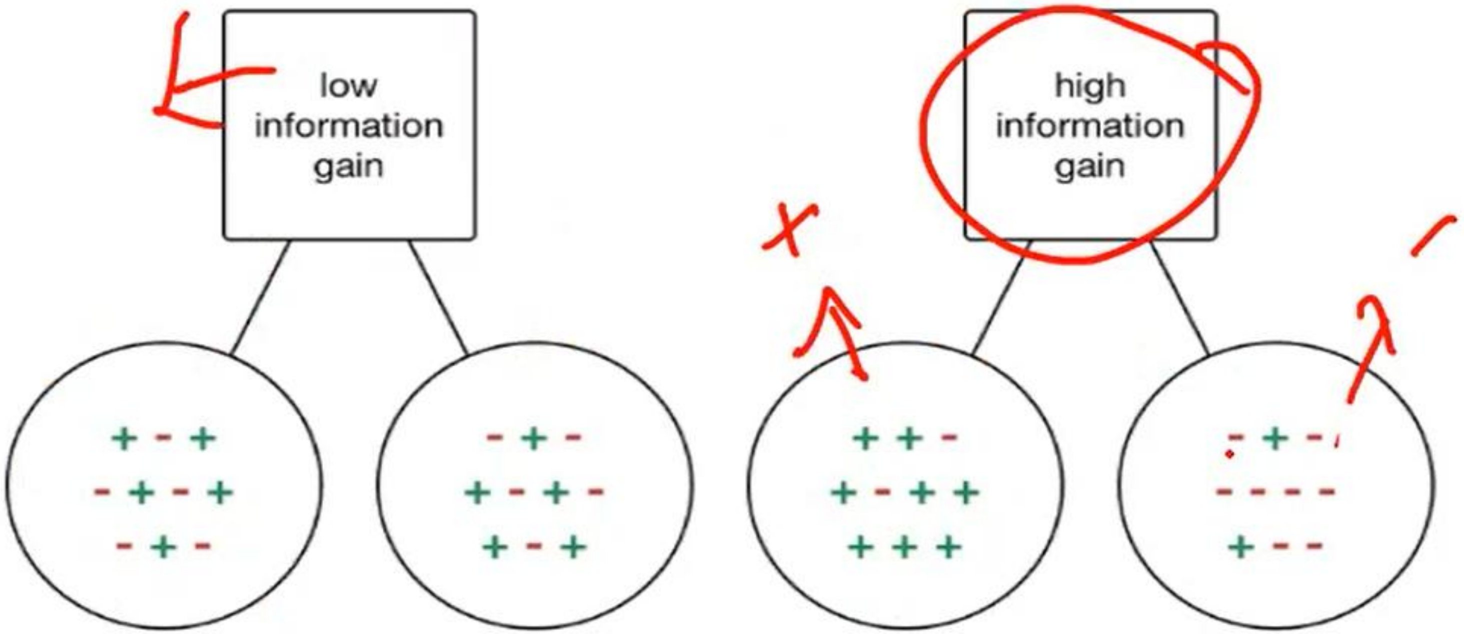
Entropy(PlayGolf) = Entropy (5,9)
= Entropy (0.36, 0.64)
= - (0.36 $\log_2$ 0.36) - (0.64 $\log_2$ 0.64)
= 0.94

		Play Golf		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5
				14

E(PlayGolf, Outlook) = **P**(Sunny)***E**(3,2) + **P**(Overcast)***E**(4,0) + **P**(Rainy)***E**(2,3)
= (5/14)*0.971 + (4/14)*0.0 + (5/14)*0.971
= 0.693

How to select the best variable from the dataset for Root Node

Effective Split of Target variable → **Information gain**



Constructing a decision tree is all about finding an attribute that returns the highest information gain and the smallest entropy.

How to select the best variable from the dataset for Root Node

Cost Function → Gini index,

You can understand the Gini index as a cost function used to evaluate splits in the dataset.
It is calculated by subtracting the sum of the squared probabilities of each class from one.
It favors larger partitions and easy to implement whereas information gain favors smaller partitions with distinct values.

$$Gini = 1 - \sum_{i=1}^{C} (p_i)^2$$

How to select the best variable from the dataset for Root Node

Larger root Value → **Gain Ratio,**

Gain ratio overcomes the problem with information gain by taking into account the number of branches that would result before making the split.
It corrects information gain by taking the intrinsic information of a split into account.

$$Gain\ Ratio = \frac{Information\ Gain}{SplitInfo} = \frac{Entropy\ (before) - \sum_{j=1}^{K} Entropy(j, after)}{\sum_{j=1}^{K} w_j \log_2 w_j}$$

How to select the best variable from the dataset for Root Node

Based on Variance → **Reduction in Variance**

This algorithm uses the standard formula of variance to choose the best split.
The split with lower variance is selected as the criteria to split the population:

$$\text{Variance} = \frac{\Sigma(X - \overline{X})^2}{n}$$

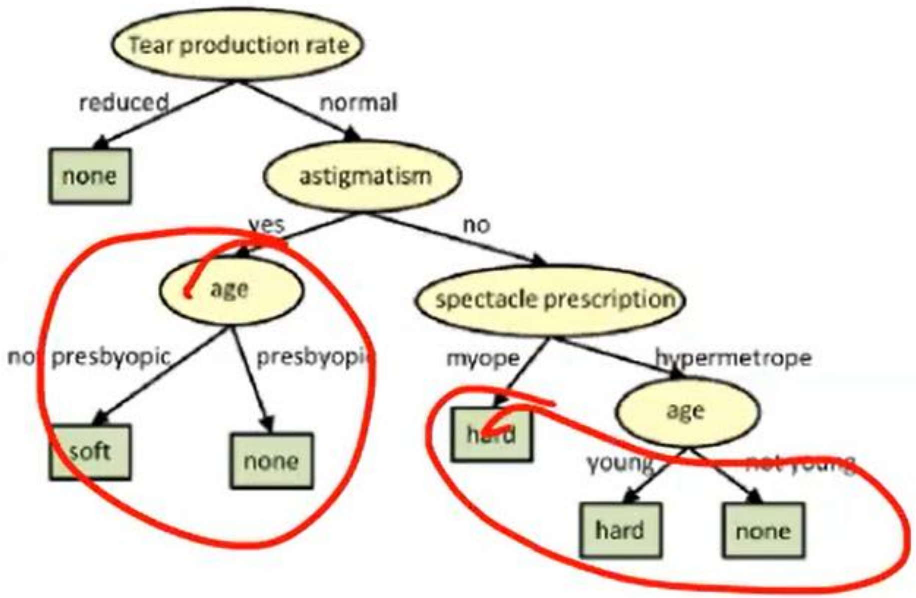
Steps to calculate Variance:

1. Calculate variance for each node.
2. Calculate variance for each split as the weighted average of each node variance.

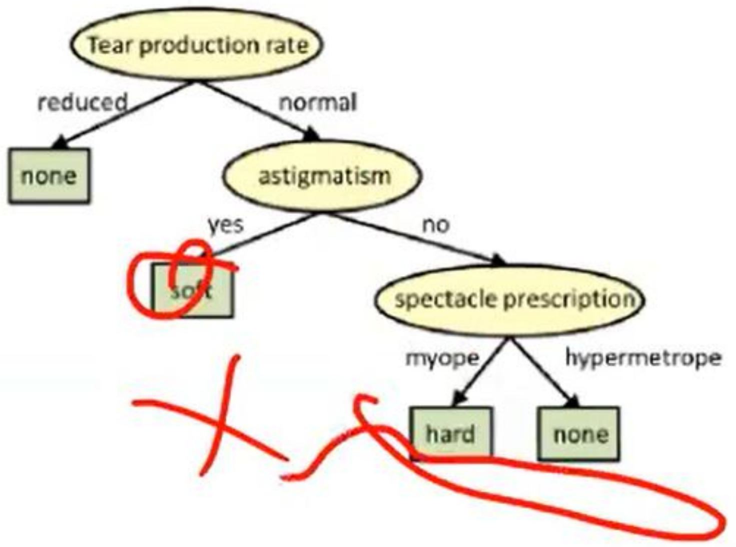
11:06

How to avoid/counter Overfitting in Decision Trees?

Pruning Decision Tree

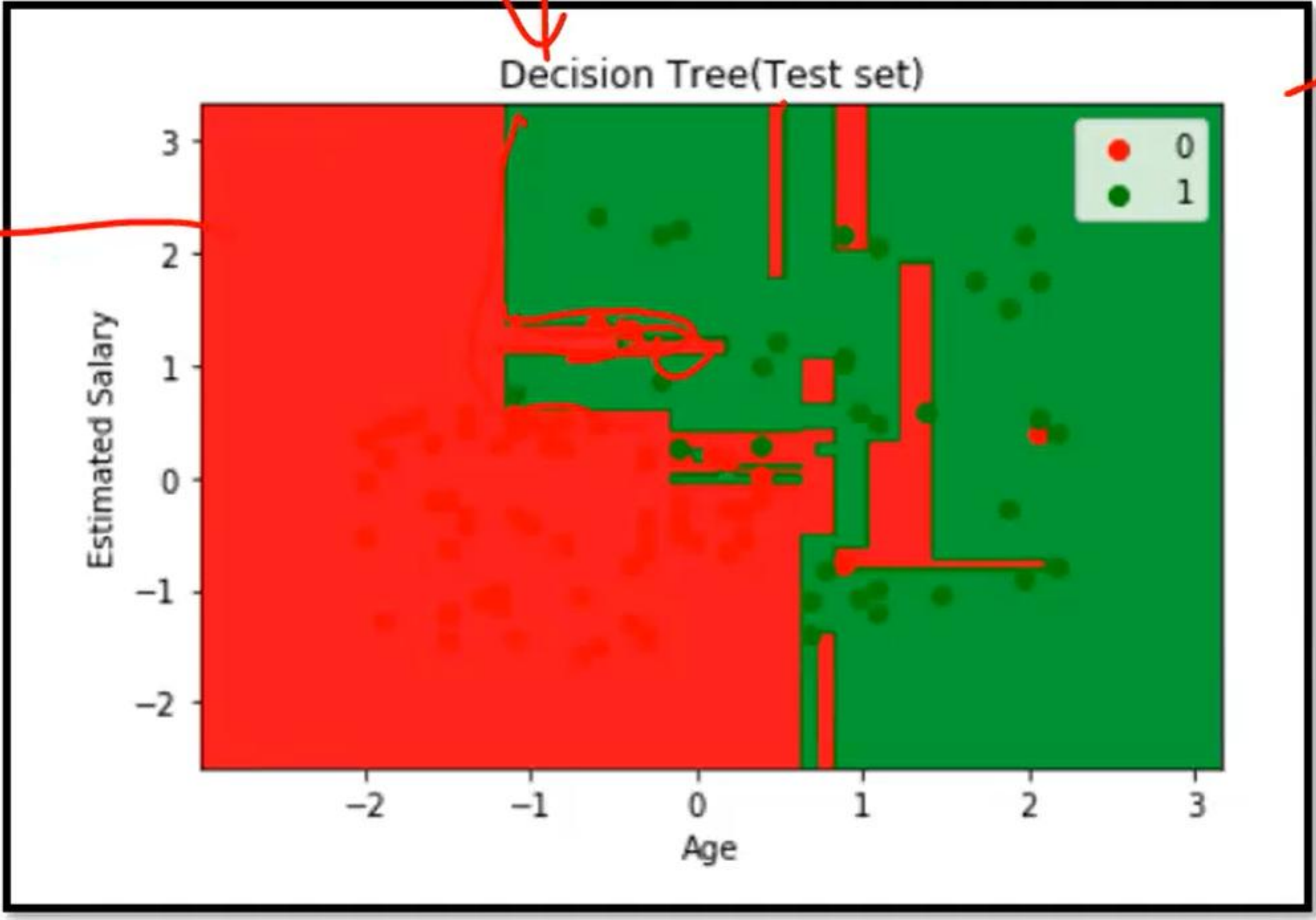


Original Tree

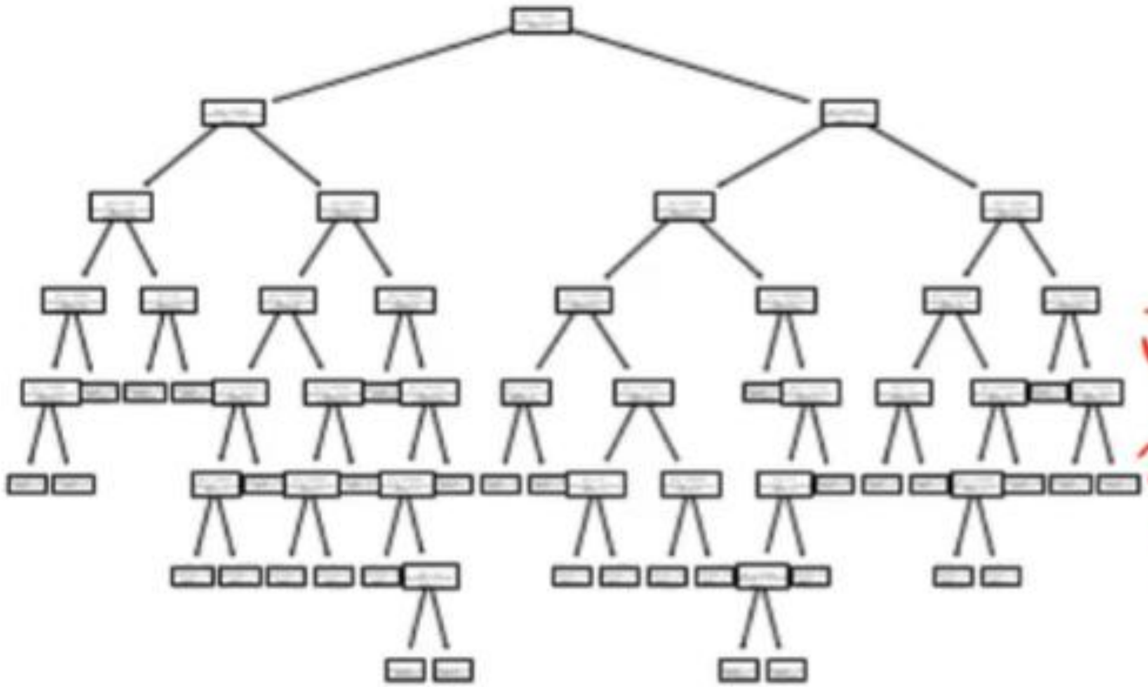


Pruned Tree

Decision Tree



```
plt.show()
```



Criterion	splitter	R-score
1) friedman	~~best~~	0.90
	best	0.91
2) squ		
3) Friedman	~~best~~ random	~~0.9~~

To find following the machine learning regression method using in r2 value

1 .MULTIPLE LINEAR REGRESSION (R^2 value) =0.7865

2. SUPPORT VECTOR MACHINE:

S.NO	HYPER PARAMETER	LINEAR (r value)	RBF (NON LINEAR) (r value)	POLY (r value)	SIGMOID (r value)
1	C10	0.4320	0.0480	0.027	0.0193
2	C100	0.6162	0.2913	0.6040	0.5056
3	C500	0.6803	0.6397	0.815	0.4638
4	C1000	0.7594	0.7915	0.8519	0.1842
5	C2000	0.7613	0.8460	0.8573	-0.5786
6	C3000	0.7612	0.8609	0.8577	-2.0119

The **SVM Regression** use R^2value (nonlinear (Rbf) and hyper parameter (C3000)) =0.8609

3. DECISION TREE:

SL.NO	CRITERION	MAX FEATURES	SPLITTER	R VALUE
1	Mse	auto	best	0.7100
2	Mse	auto	random	0.7009
3	Mse	sqrt	best	0.7318
4	Mse	sqrt	random	0.7032
5	Mse	Log2	best	0.7318
6	Mse	Log2	random	0.7032
7	Mae	auto	best	0.6672
8	Mae	auto	random	0.7537
9	Mae	sqrt	best	0.7326
10	Mae	sqrt	random	0.6635
11	Mae	Log2	best	0.7326
12	Mae	Log2	random	0.6635
13	Friedman_mse	auto	best	0.7088
14	Friedman_mse	auto	random	0.6988
15	Friedman_mse	sqrt	best	0.7281
16	Friedman_mse	sqrt	random	0.6926
17	Friedman_mse	Log2	best	0.7281
18	Friedman_mse	Log2	random	0.6926

The **Decision Tree** Regression use R^2value (Mae, auto, random) =0.7537