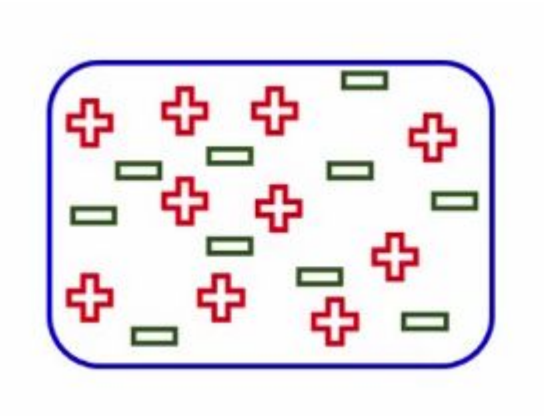
Decision Tree

- What is Decision Tree?
- Terminologies related to Decision Trees
- Different Splitting Criterion in Decision Trees
- Pros / Cons of Decision Tree
- Implementation of Decision Tree in Python

What is Decision Tree?



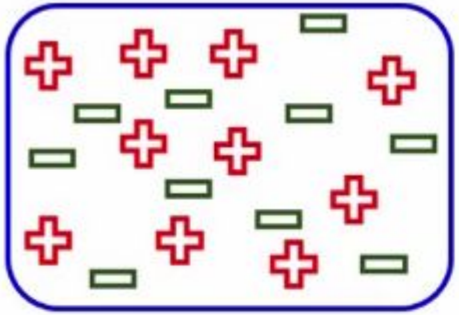
Total no.of students = 20

Play Cricket = 10

Do not play cricket = 10

What is Decision Tree?

- Height
- Performance in class
- Class



Total no.of students = 20

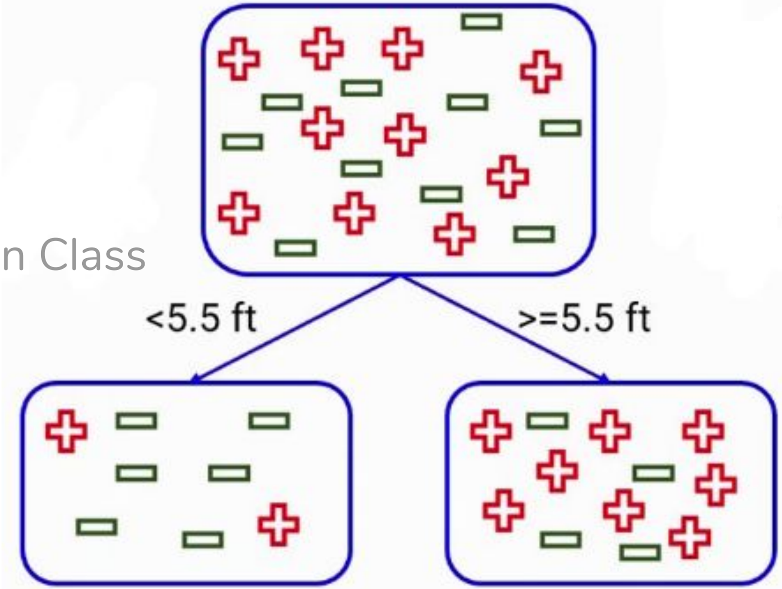
Play Cricket = 10

Do not play cricket = 10

What is Decision Tree?

- Split on Height
- Split on Performance in Class
- Split on Class

Students = 20
Play Cricket = 10
Percentage = 50%



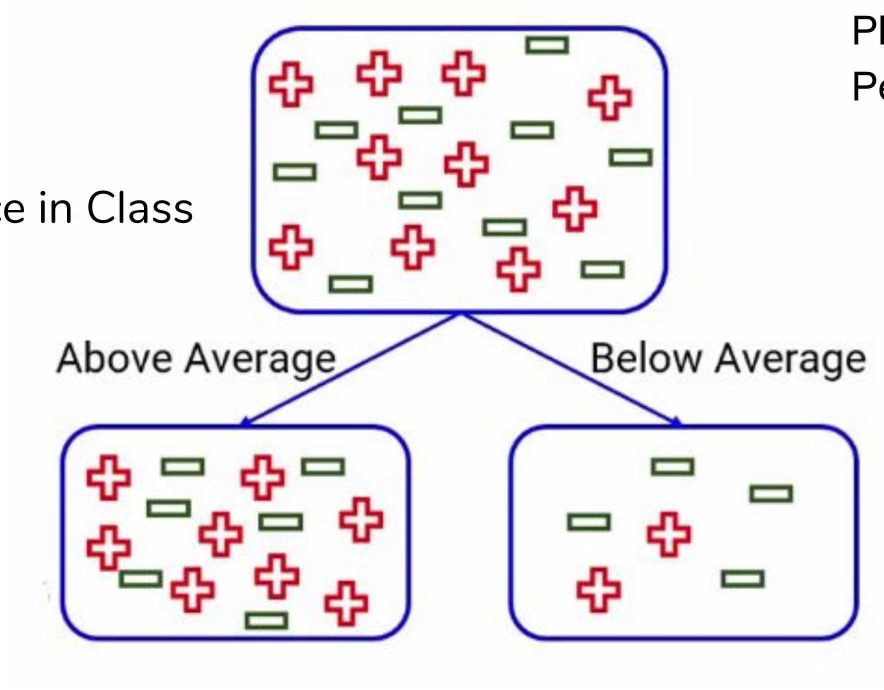
Students = 8
Play Cricket = 2
Percentage = 25%

Students = 12
Play Cricket = 8
Percentage = 66.67%

What is Decision Tree?

- Split on Height
- Split on Performance in Class
- Split on Class

Students = 20
Play Cricket = 10
Percentage = 50%



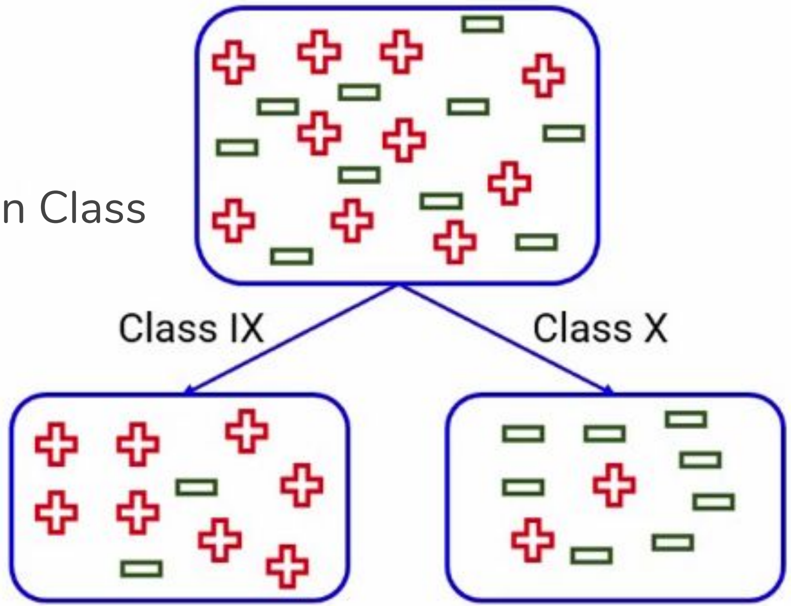
Students = 14
Play Cricket = 8
Percentage = 57.14%

Students = 6
Play Cricket = 2
Percentage = 33.33%

What is Decision Tree?

- Split on Height
- Split on Performance in Class
- Split on Class

Students = 20
Play Cricket = 10
Percentage = 50%

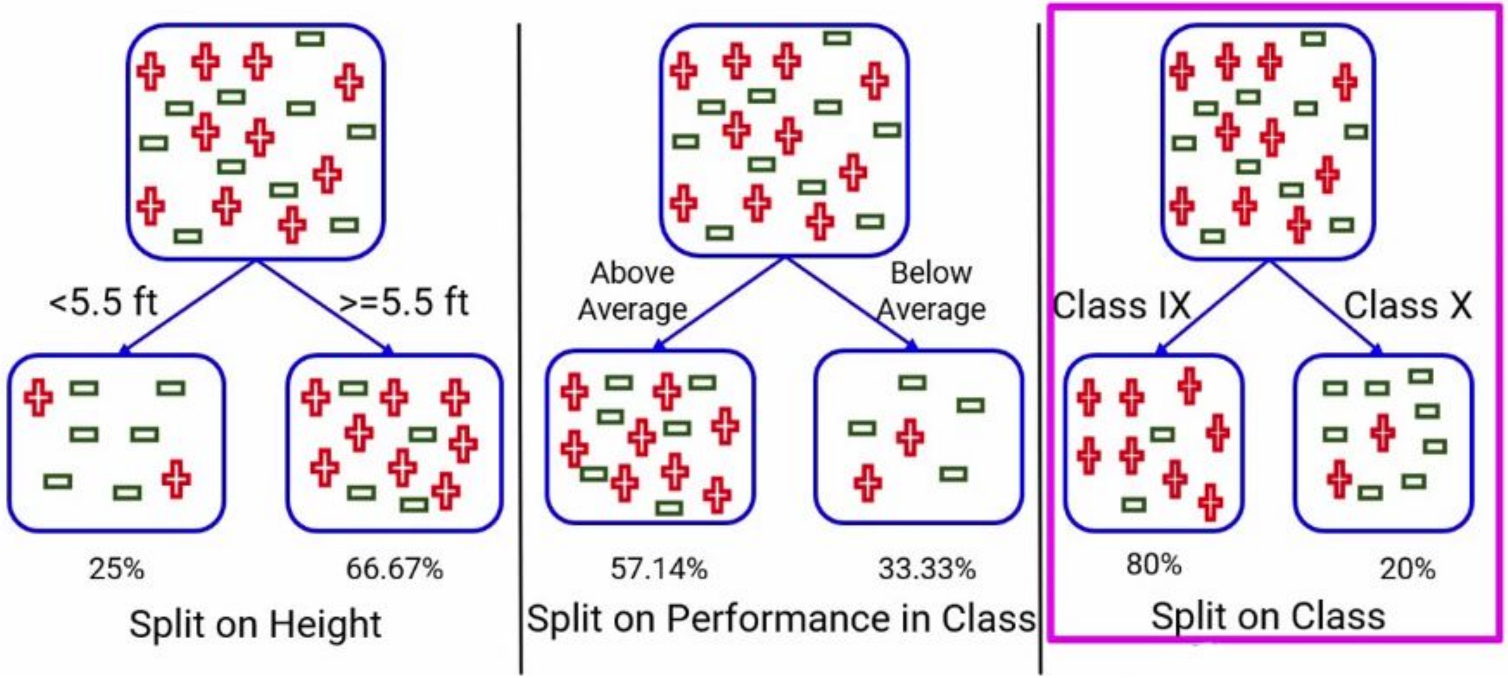


Students = 10
Play Cricket = 8
Percentage = 80%

Students = 10
Play Cricket = 2
Percentage = 20%

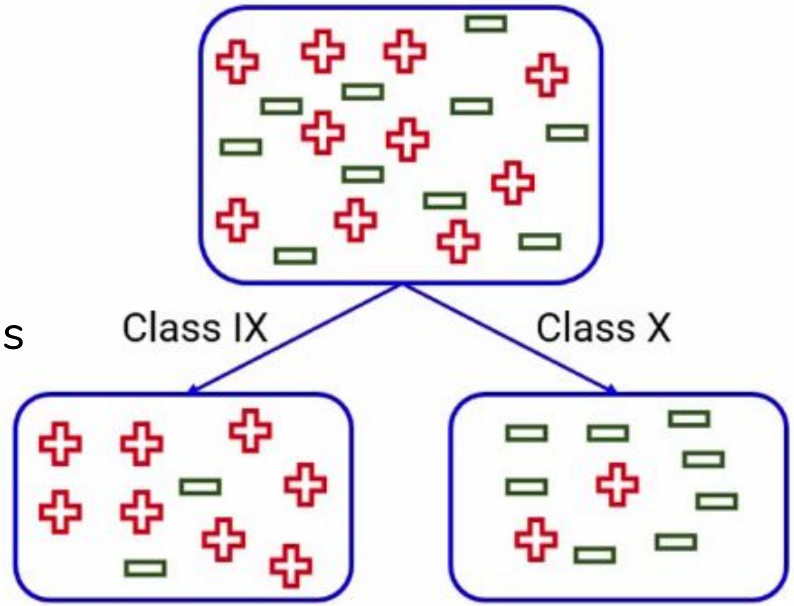
What is Decision Tree?

What is Decision Tree?



Purity in Decision Tree

- Height
- Performance in class
- Class



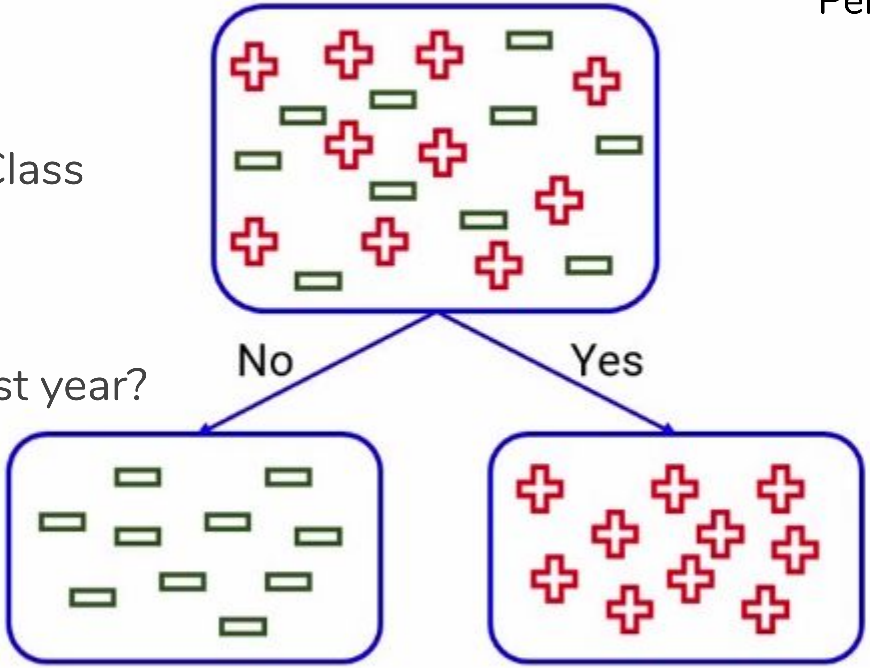
80%

20%

Split on Class

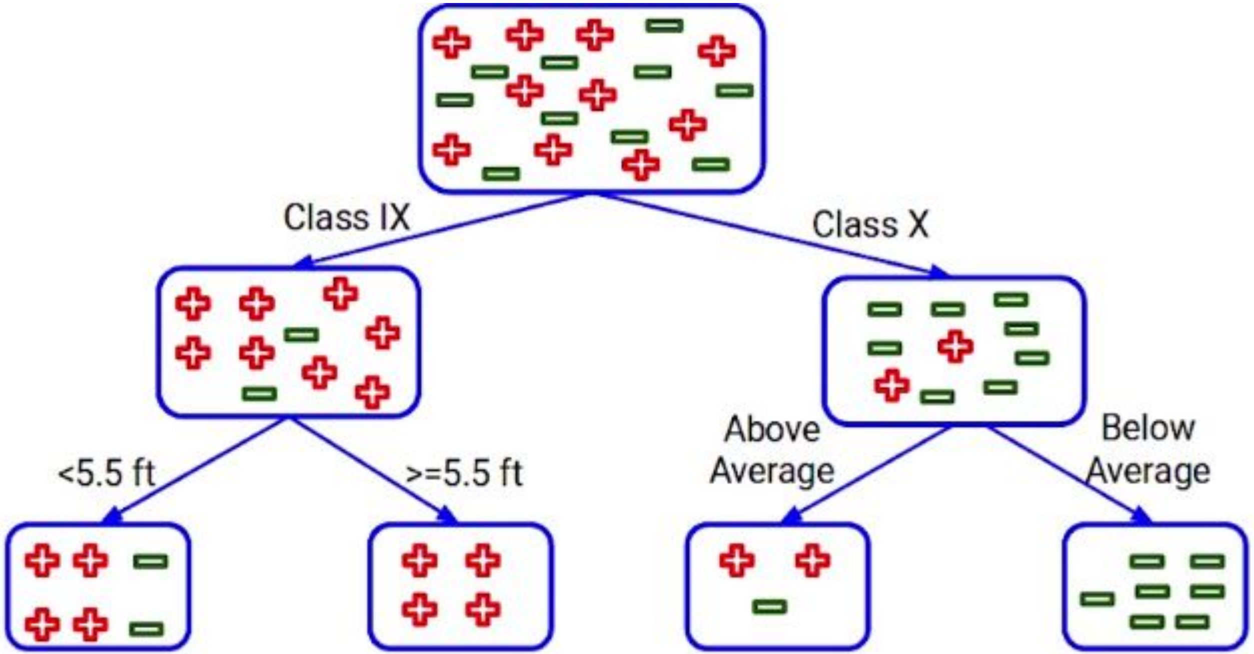
Students = 20
Play Cricket = 10
Percentage = 50%

- Split on Height
- Split on Performance in Class
- Split on Class
- Split on Played Cricket last year?



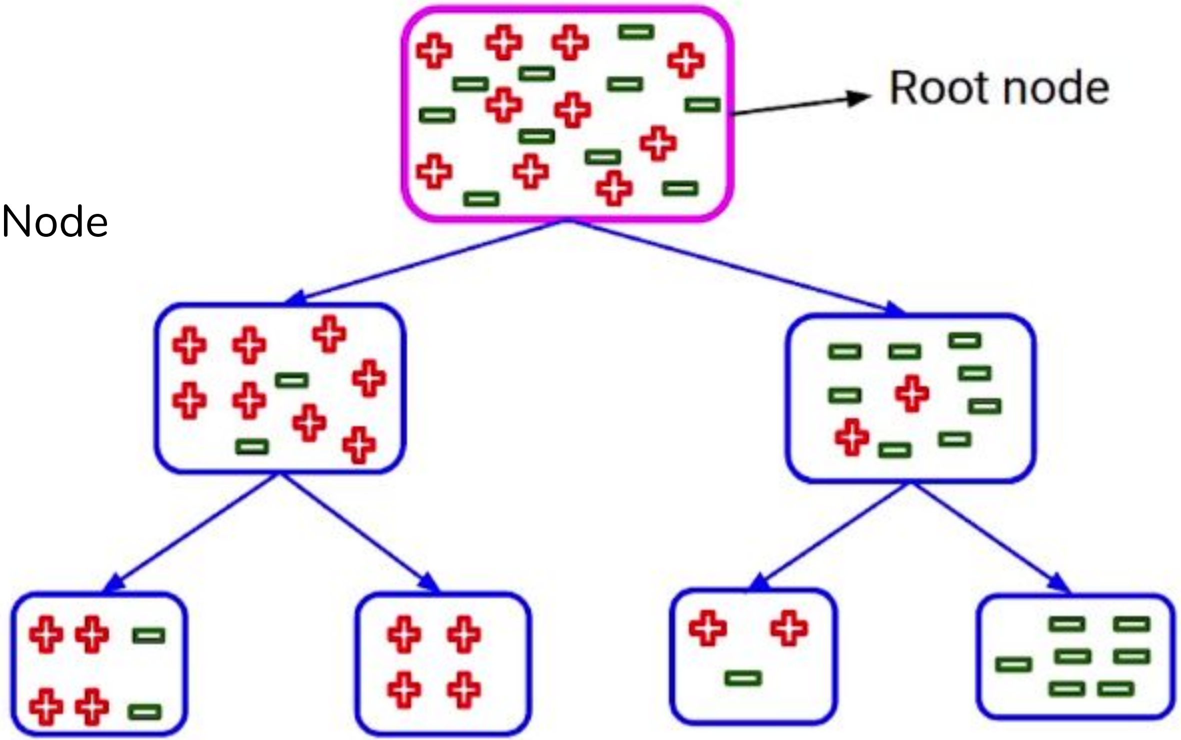
Students = 10
Play Cricket = 0
Percentage = 0%

Students = 10
Play Cricket = 10
Percentage = 100%

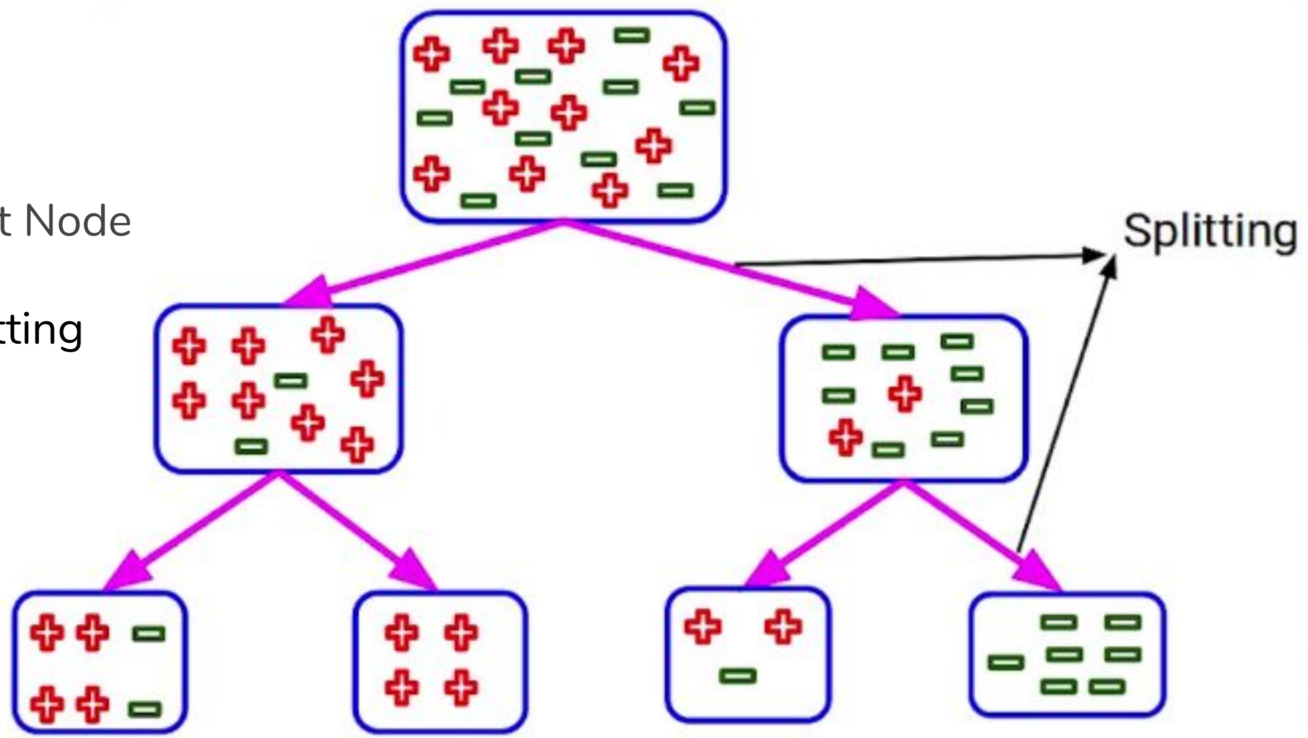

Class IX
Class X
<5.5 ft
>=5.5 ft
Above Average
Below Average

Terminologies related to Decision Tree

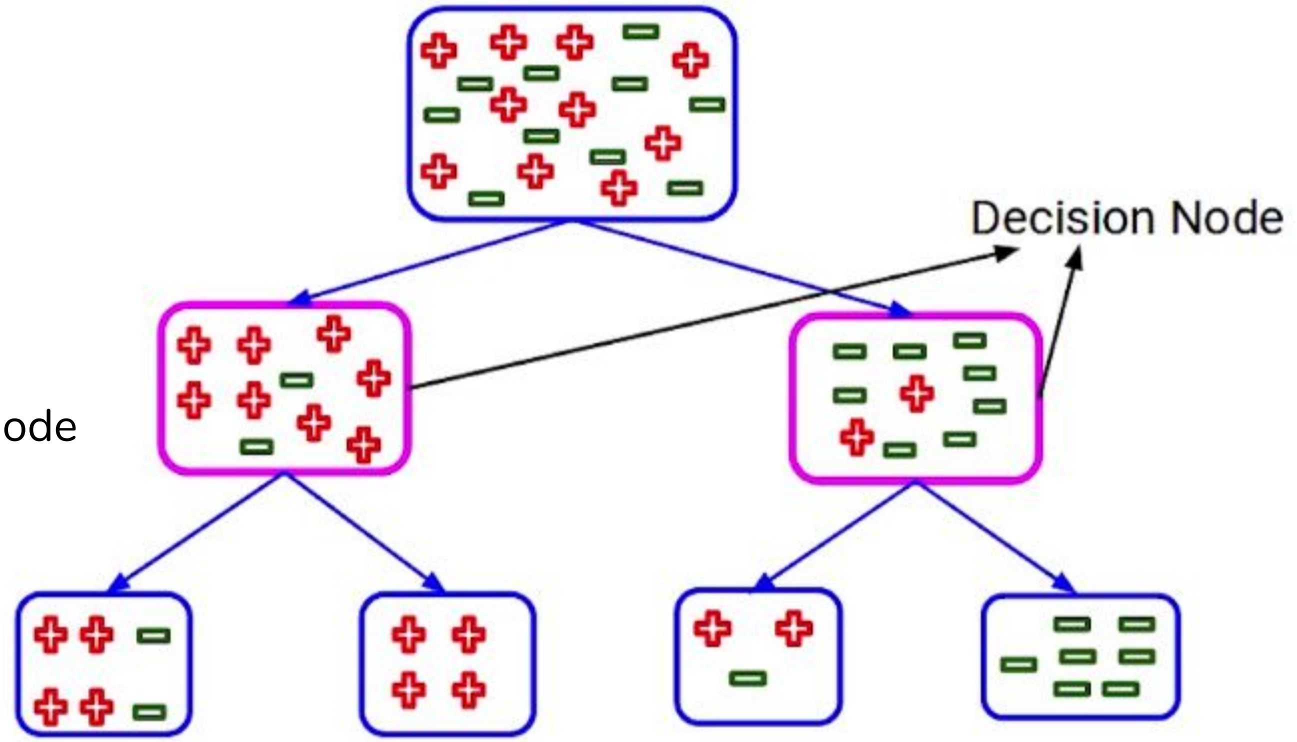
- Root Node



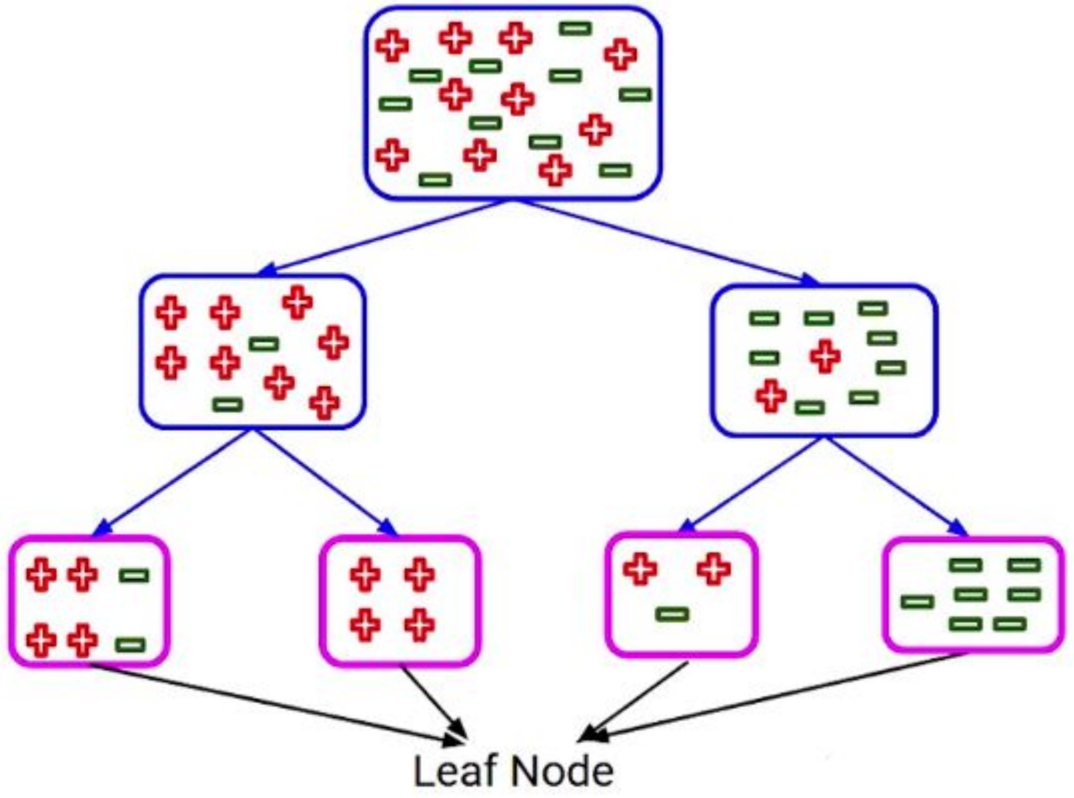
- Root Node
- Splitting



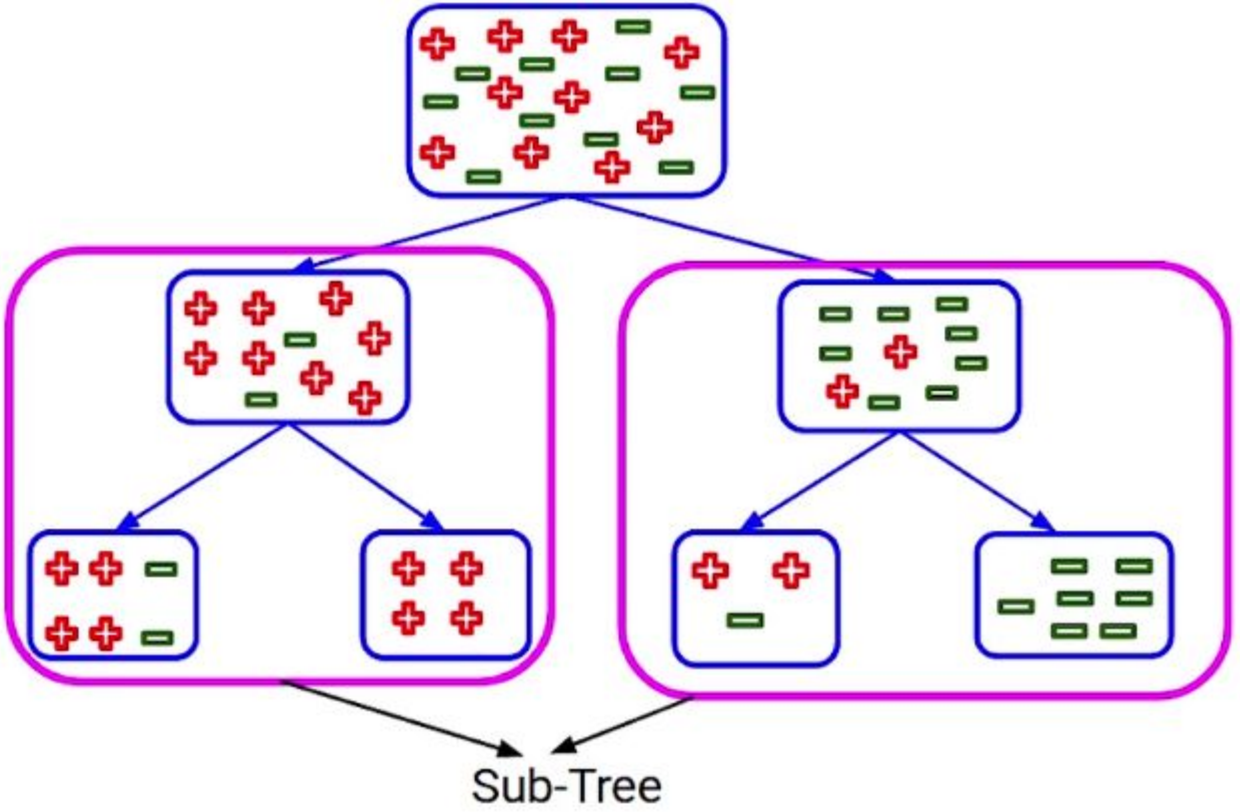
- Root Node
- Splitting
- Decision Node



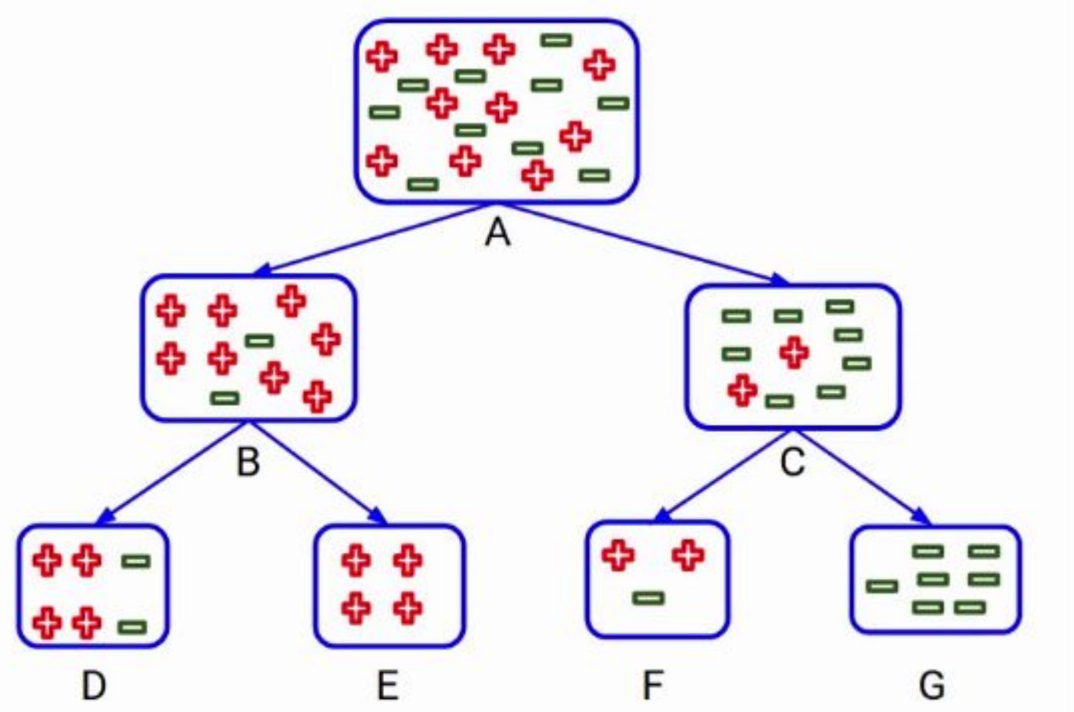
- Root Node
- Splitting
- Decision Node
- Leaf/ Terminal Node



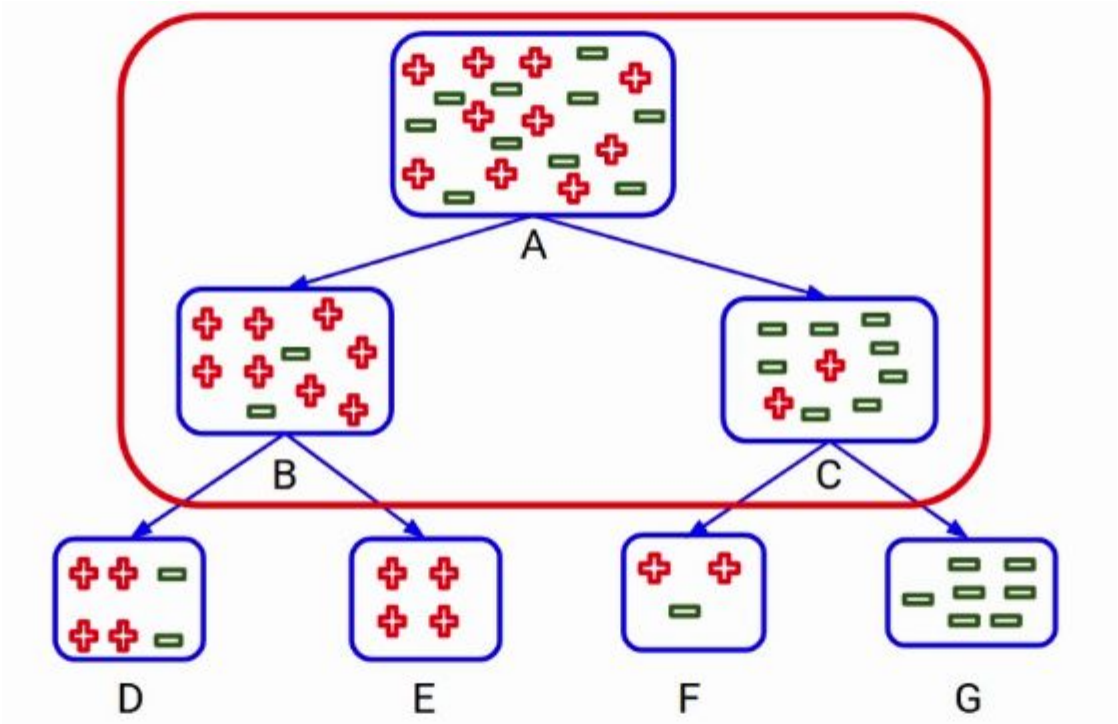
- Root Node
- Splitting
- Decision Node
- Leaf/ Terminal Node
- Branch/Sub-tree



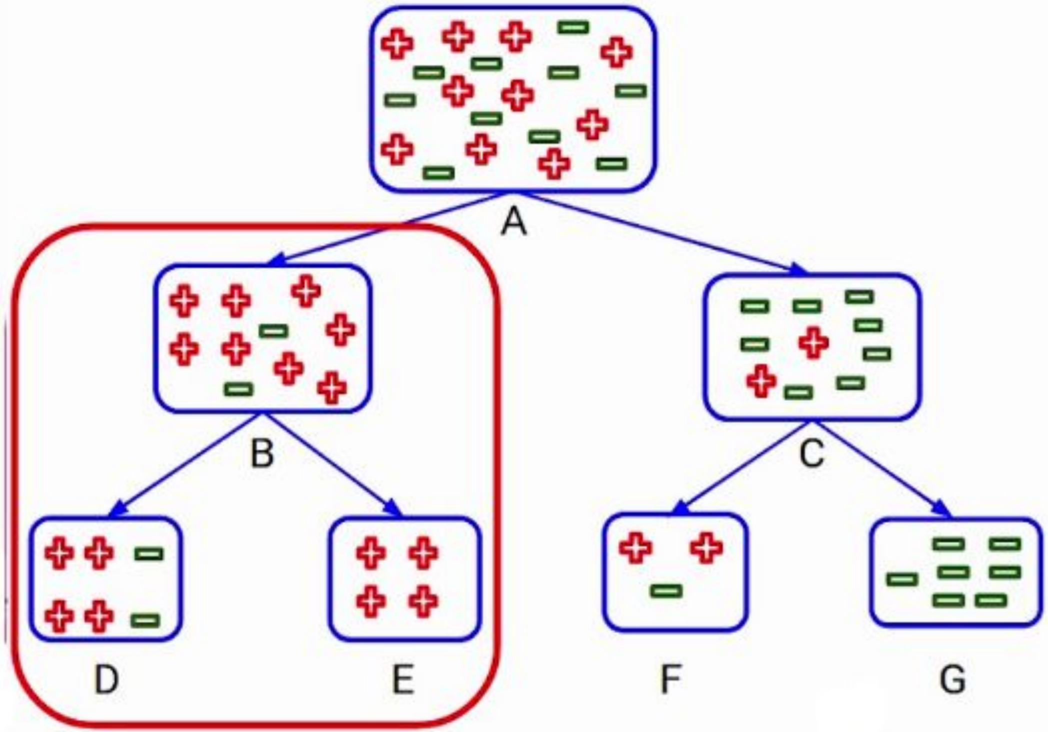
- Root Node
- Splitting
- Decision Node
- Leaf/ Terminal Node
- Branch/Sub-tree
- Parent and Child node



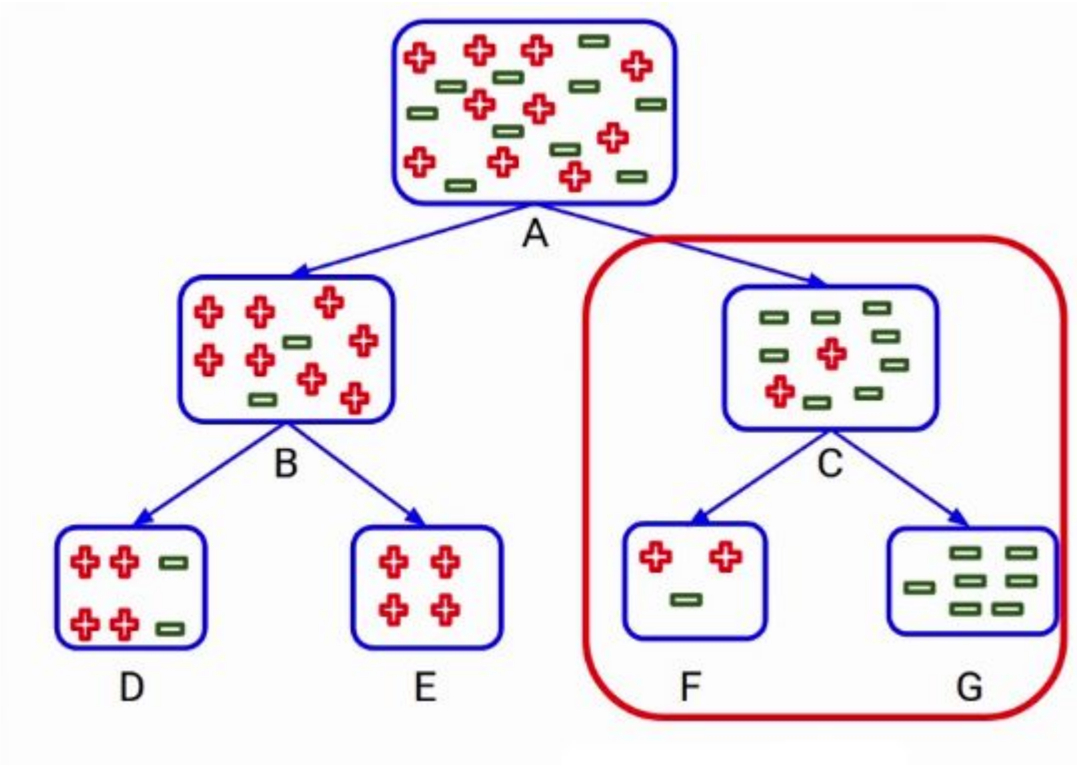
- Root Node
- Splitting
- Decision Node
- Leaf/ Terminal Node
- Branch/Sub-tree
- Parent and Child node



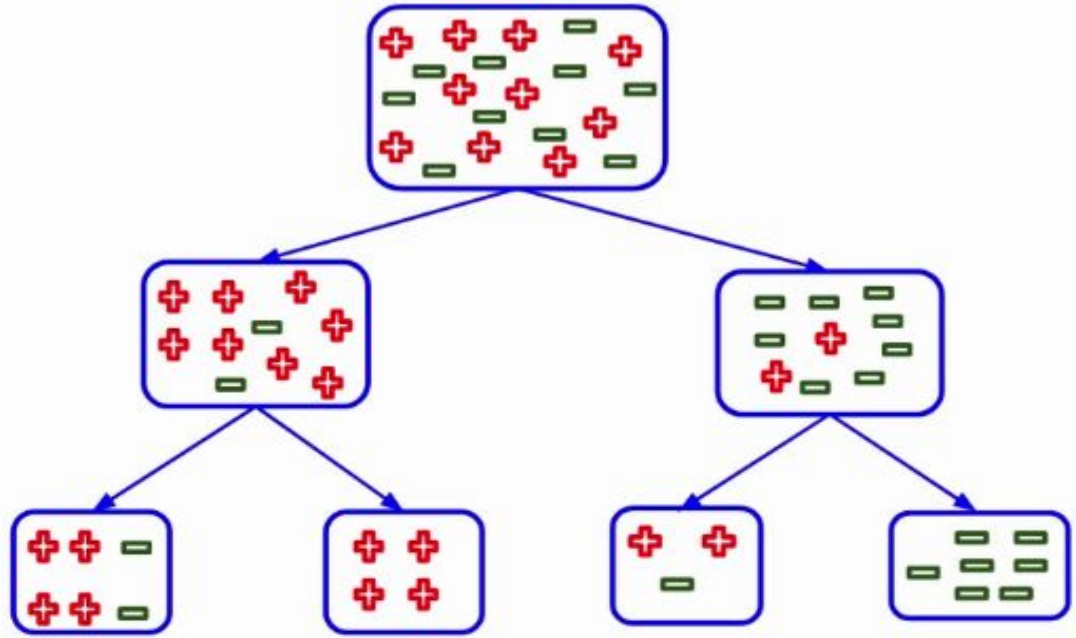
- Root Node
- Splitting
- Decision Node
- Leaf/ Terminal Node
- Branch/Sub-tree
- Parent and Child node



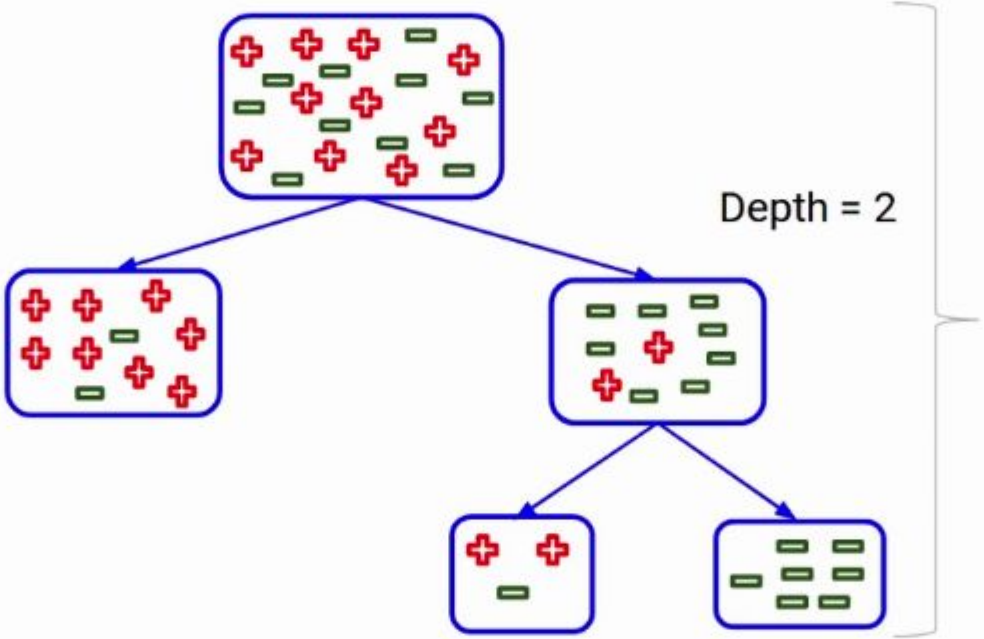
- Root Node
- Splitting
- Decision Node
- Leaf/ Terminal Node
- Branch/Sub-tree
- Parent and Child node



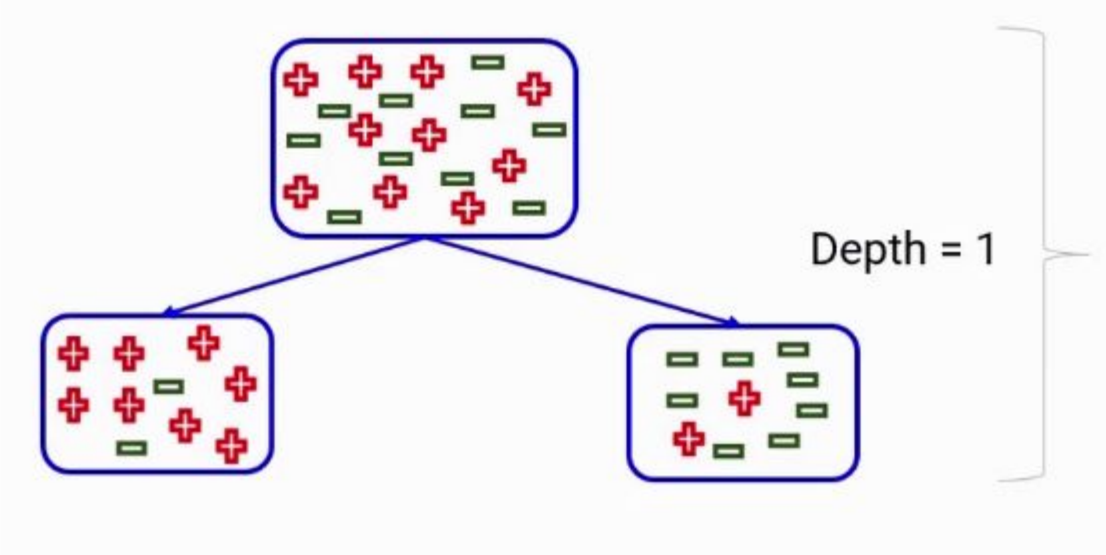
- Root Node
- Splitting
- Decision Node
- Leaf/ Terminal Node
- Branch/Sub-tree
- Parent and Child node
- Depth of Tree



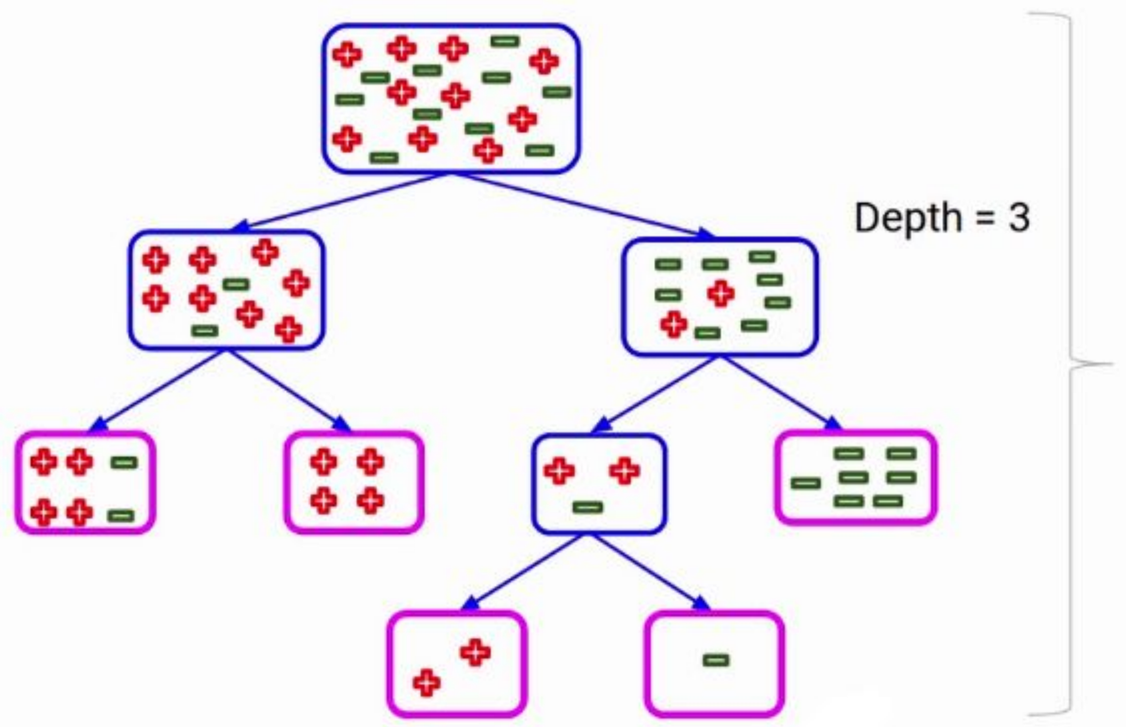
- Root Node
- Splitting
- Decision Node
- Leaf/ Terminal Node
- Branch/Sub-tree
- Parent and Child node
- Depth of Tree



- Root Node
- Splitting
- Decision Node
- Leaf/ Terminal Node
- Branch/Sub-tree
- Parent and Child node
- Depth of Tree



- Root Node
- Splitting
- Decision Node
- Leaf/ Terminal Node
- Branch/Sub-tree
- Parent and Child node
- Depth of Tree



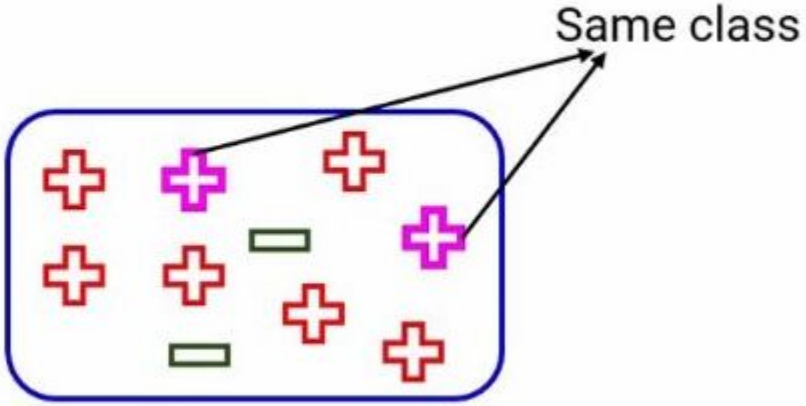
How to select the best split point in Decision Tree

1. Decision tree split all the node.
2. Select the split which result in most homogenous subnodes based on algorithm.

Gini Impurity

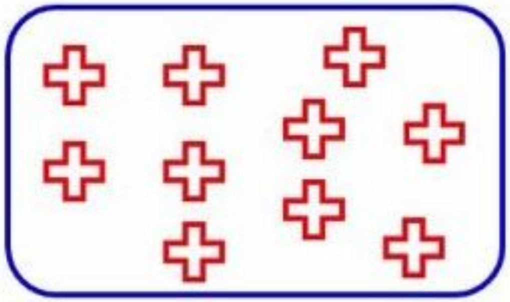
Gini impurity = 1-Gini(Measurement of the impurity of nodes and calculated using this)

Gini



If we select two items from a population at random,they must be of same class.

Gini



Probability that randomly picked points belong to same class

Probability = 1

Properties of Gini Impurity

- Node split is decided based on the gini impurity

$$Gini\ Impurity = 1 - Gini$$

- Lower the gini impurity, higher the homogeneity of nodes

- Works only with categorical targets

- Only performs binary splits

Steps to calculate Gini Impurity for a split

- Calculate the Gini impurity for sub-nodes:

  Gini impurity = 1- Gini

- Gini = Sum of square of probabilities for each class/category

  $$\textbf{Gini} = (p_1^2 + p_2^2 + p_3^2 + \ldots\ldots\ldots\ldots\ldots + p_n^2)$$

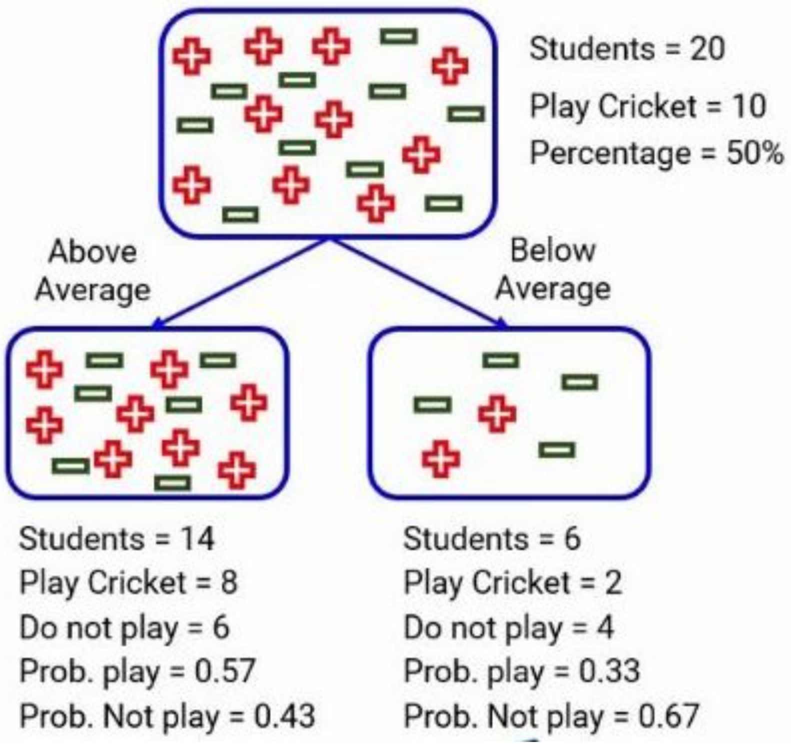
- To calculate the gini impurity for split, take weighted gini impurity of both sub-nodes of that split

Steps to Calculate Gini for a split

Split on Performance in Class

- Gini Impurity: Sub-node Above Average :

  1 - [(0.57)*(0.57)+(0.43)*(0.43)] = 0.49



Steps to Calculate Gini for a split

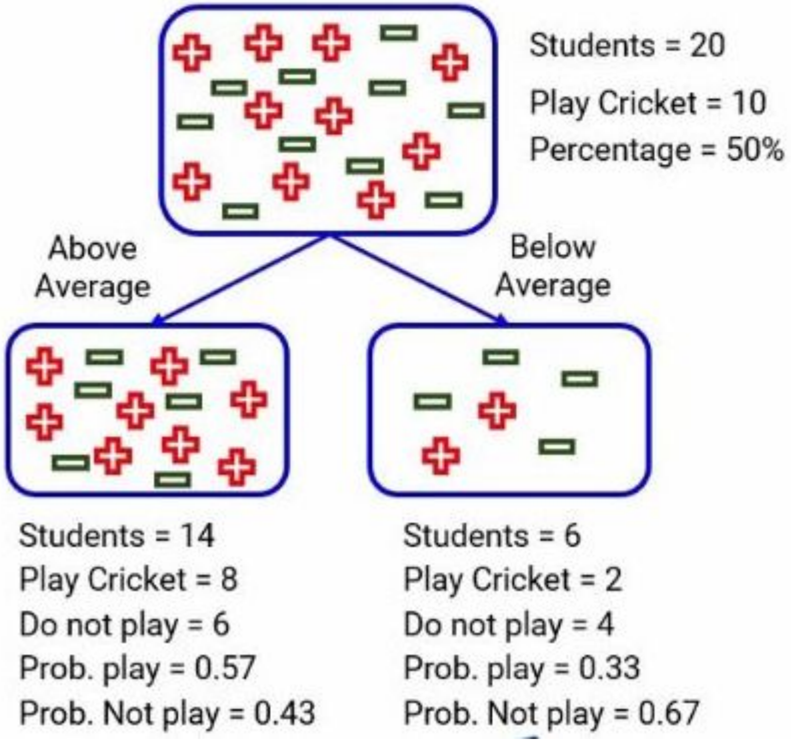
Split on Performance in Class

- Gini Impurity: Sub-node Above Average :

  1 - [(0.57)*(0.57)+(0.43)*(0.43)] = 0.49

- Gini Impurity: Sub-node Below Average :

  1 - [(0.33)*(0.33)+(0.67)*(0.67)] = 0.44



Steps to Calculate Gini for a split

Split on Performance in Class

- Gini Impurity: Sub-node Above Average :

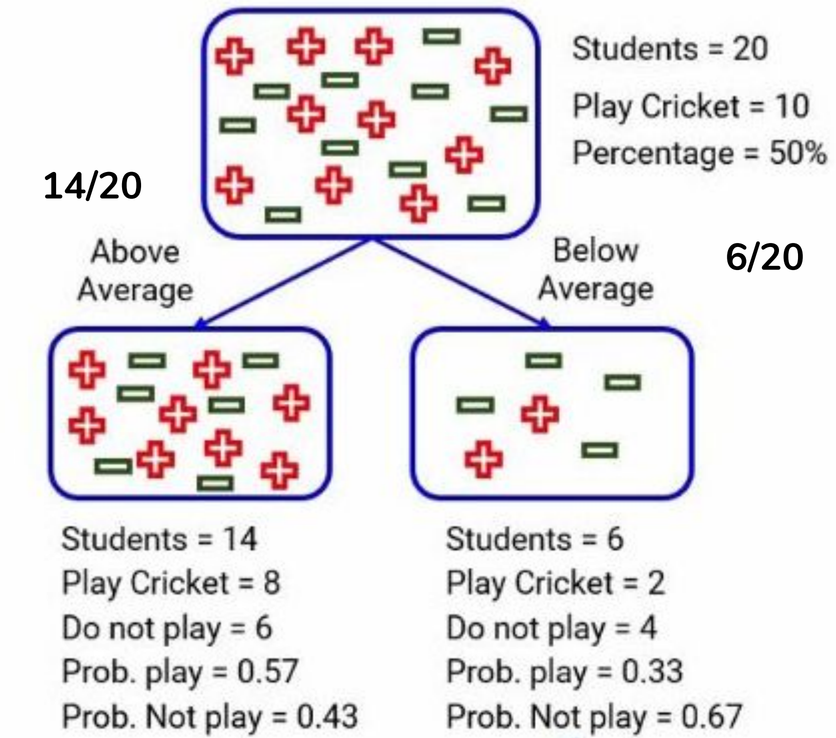
  1 - [(0.57)*(0.57)+(0.43)*(0.43)] = 0.49

- Gini Impurity: Sub-node Below Average :

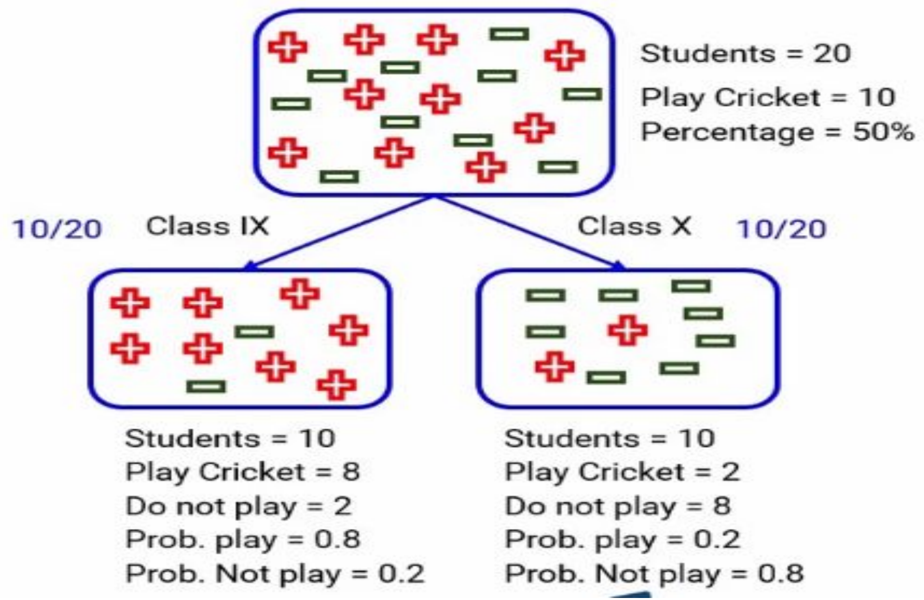
  1 - [(0.33)*(0.33)+(0.67)*(0.67)] = 0.44

- Weighted Gini Impurity: Performance in class :

  (14/20)*0.49+(6/20)*0.44=0.475



Steps to calculate Gini Impurity for a split



Steps to Calculate Gini for a split

Split on Class

- Gini Impurity: Sub-node Class IX:

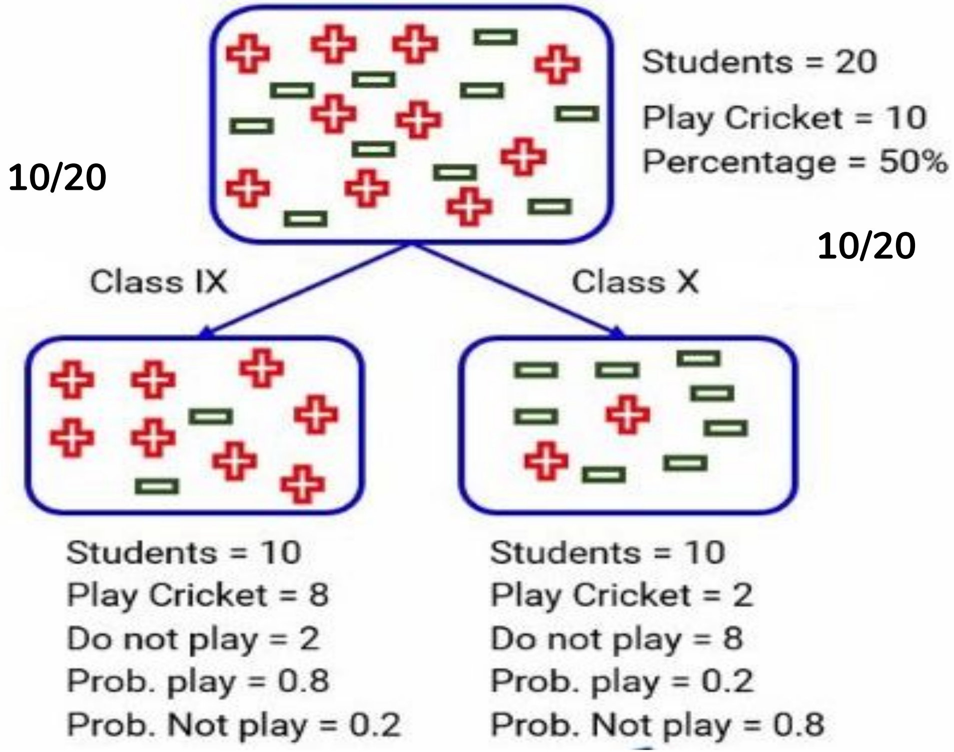
  1 - [(0.8)*(0.8)+(0.2)*(0.2)] = 0.32

- Gini Impurity: Sub-node Class X:

  1 - [(0.2)*(0.2)+(0.8)*(0.8)] = 0.32

- Weighted Gini Impurity: Class :

  (10/20)*0.32+(10/20)*0.32=0.32



Steps to Calculate Gini for a split

(Lower the Gini Impurity Higher the homogenity)

Split	Weighted Gini Impurity
Performance in Class	0.475
Class	0.32

Chi-Square

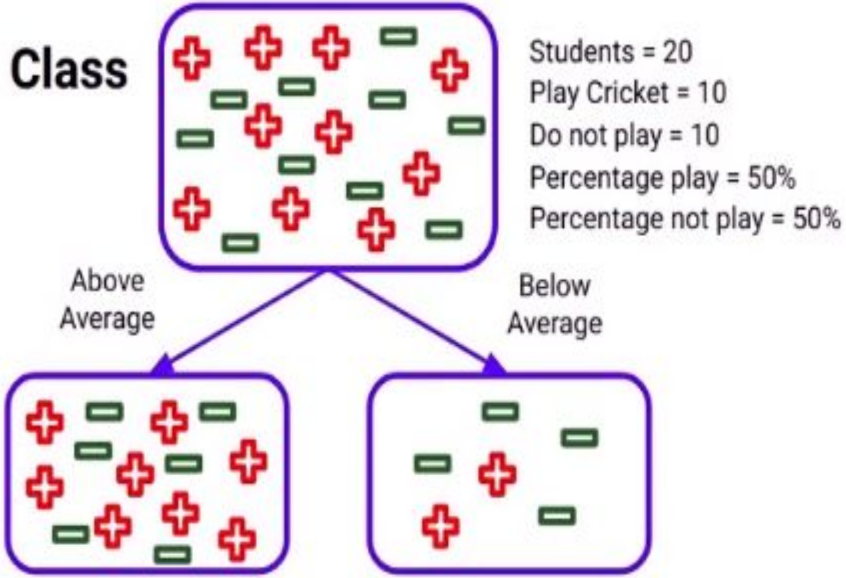
- Statistical significance between the differences between sub-nodes and parent node.
- Sum of squares of standardized differences between observed and expected frequencies of target variable.

$$\text{Chi-Square} = \sqrt{[(\text{Actual} - \text{Expected})^2 / \text{Expected}]}$$

- It generates tree called CHAID (Chi-square Automatic Interaction Detector)

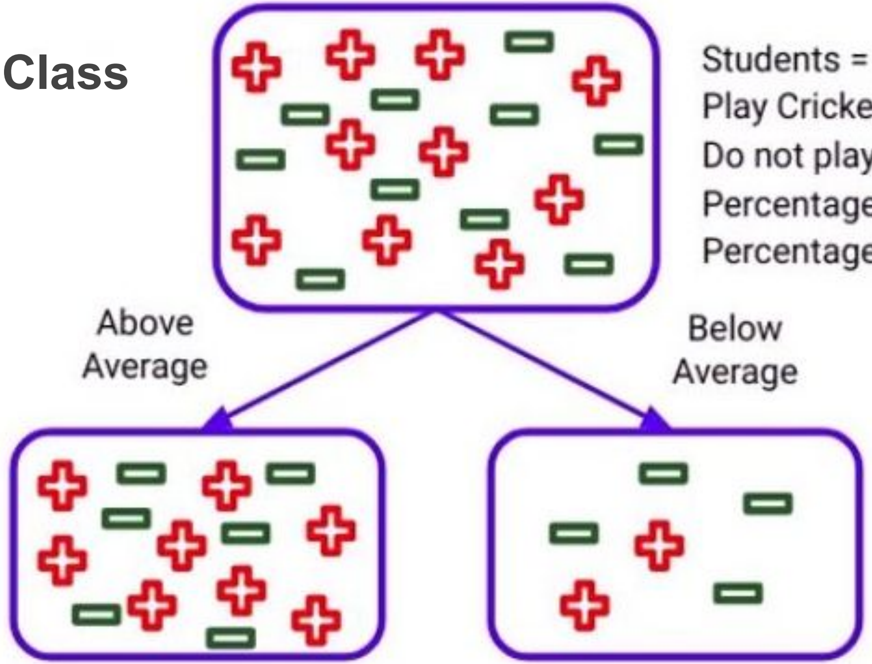
Steps to calculate Chi-Square for a split

Split on Performance in Class



Steps to Calculate Chi-square for a split

Split on Performance in Class



Students = 20
Play Cricket = 10
Do not play = 10
Percentage play = 50%
Percentage not play = 50%

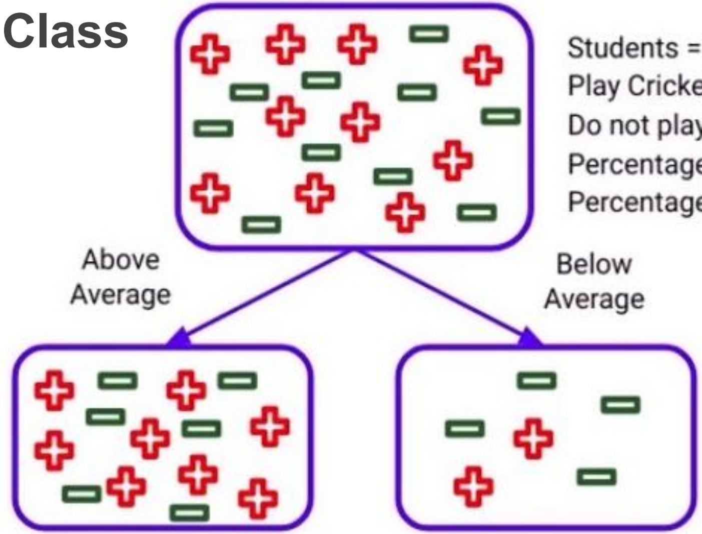
Students = 14

Expected Play = 7
Play Cricket = 8

Expected Not Play = 7
Not Play Cricket = 6

Steps to Calculate Chi-square for a split

Split on Performance in Class



Students = 20
Play Cricket = 10
Do not play = 10
Percentage play = 50%
Percentage not play = 50%

Students = 14

Expected Play = 7
Play Cricket = 8

Expected Not Play = 7
Not Play Cricket = 6

Students = 6

Expected Play = 3
Play Cricket = 2

Expected Not Play = 3
Not Play Cricket = 4

Chi-Square = $\sqrt{[(Actual - Expected)^2 / Expected]}$

Properties of Chi- Square

- Works only with categorical target variable
- Higher the Chi- Square value, higher the homogeneity of nodes

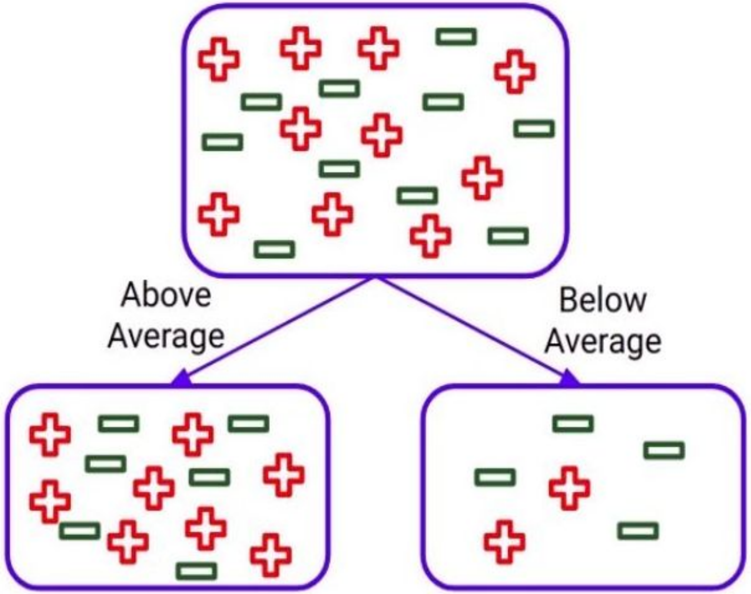
Steps to calculate Chi-Square for a split

- Calculate the expected values for each class for every child nodes
- Calculate the Chi-Square for every child node

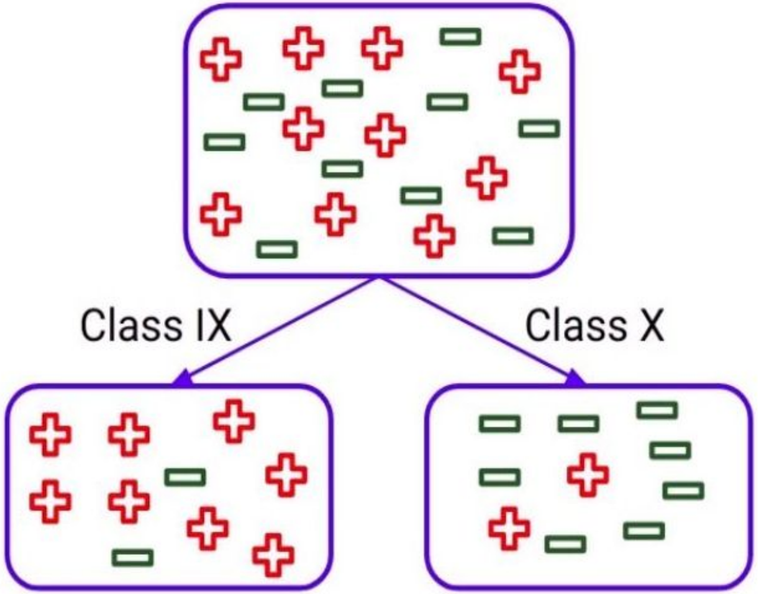
$$\text{Chi-Square} = \sqrt{[(\text{Actual} - \text{Expected})^2 / \text{Expected}]}$$

- Calculate Chi-Square for split using sum of Chi-Square of each child node of that split

Steps to calculate Chi-Square for a split



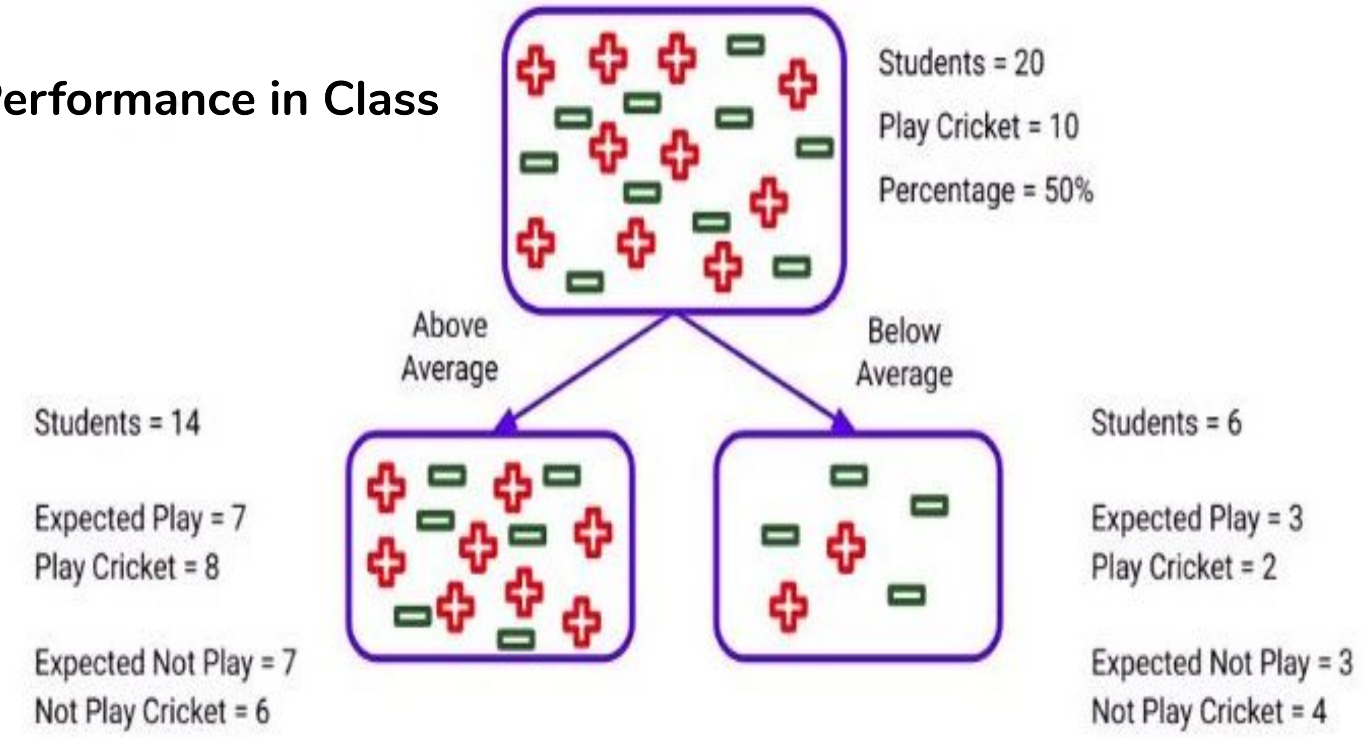
Split on Performance in Class



Split on Class

Steps to calculate Chi-Square for a split

Split on Performance in Class



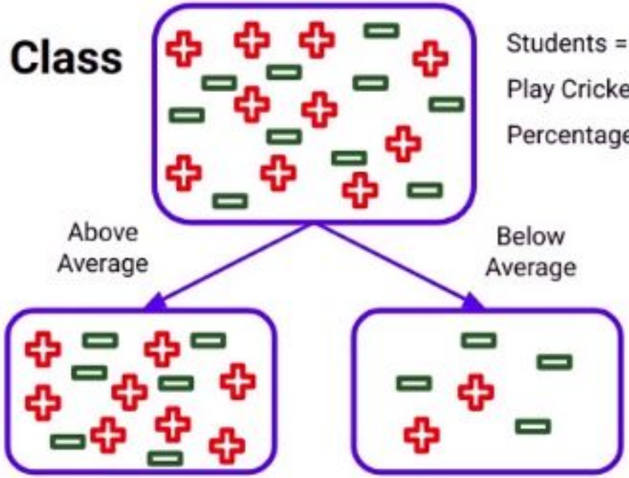
Steps to calculate Chi-Square for a split

Split on Performance in Class

Students = 20

Play Cricket = 10

Percentage = 50%



Students = 14

Expected Play = 7
Play Cricket = 8

Expected Not Play = 7
Not Play Cricket = 6

Students = 6

Expected Play = 3
Play Cricket = 2

Expected Not Play = 3
Not Play Cricket = 4

Node	Actual Play	Actual Not Play	Expected Play	Expected Not Play	Deviation Play	Deviation Not Play	Chi-Square (Play)	Chi-Square (Not Play)
Above Average								
Below Average								

Steps to calculate Chi-Square for a split

Node	Actual Play	Actual Not Play	Expected Play	Expected Not Play	Deviation Play	Deviation Not Play	Chi-Square(Play)	Chi-Square(Not Play)
Above Average	8	6	7	7				
Below Average	2	4	3	3				

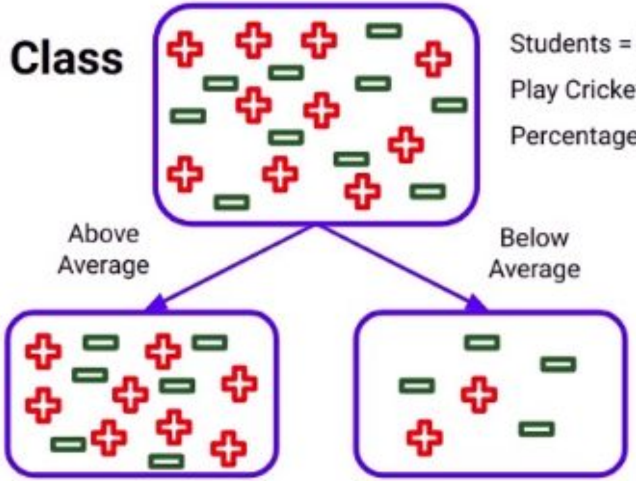
$$\text{Chi-Square} = \sqrt{[(\text{Actual} - \text{Expected})^2 / \text{Expected}]}$$

Steps to calculate Chi-Square for a split

Split on Performance in Class

Students = 20
Play Cricket = 10
Percentage = 50%

Total = 0.38 + 0.38 + 0.58 + 0.58 = 1.92



Students = 14

Expected Play = 7
Play Cricket = 8

Expected Not Play = 7
Not Play Cricket = 6

Students = 6

Expected Play = 3
Play Cricket = 2

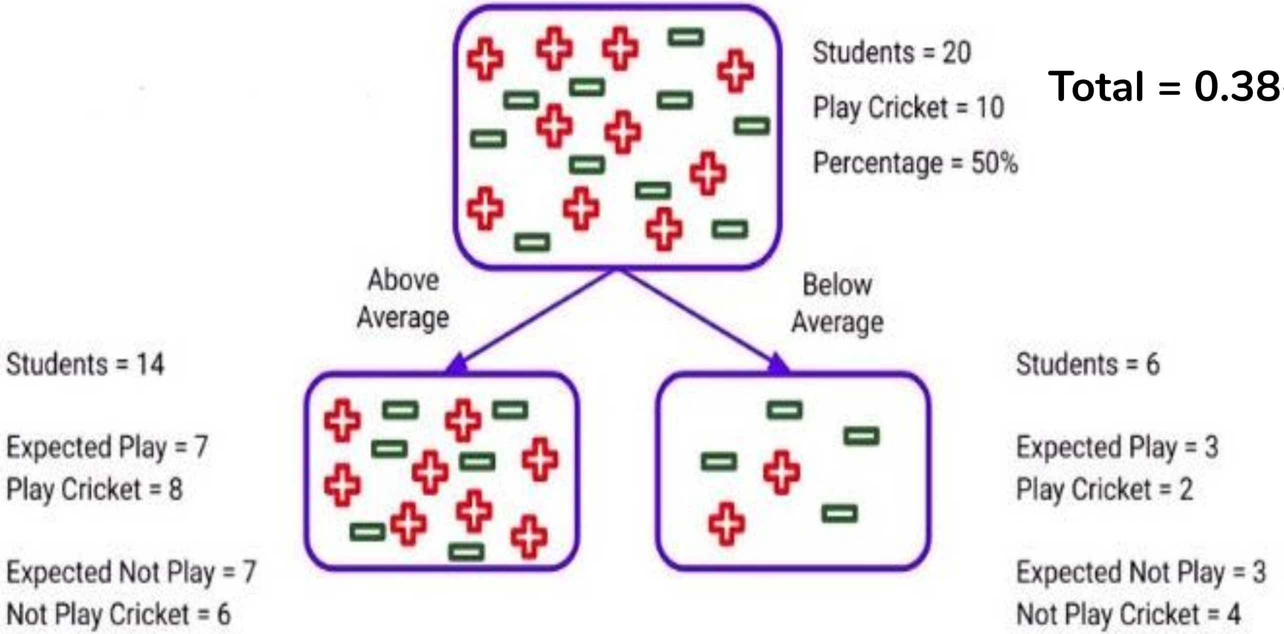
Expected Not Play = 3
Not Play Cricket = 4

Node	Actual Play	Actual Not Play	Expected Play	Expected Not Play	Deviation Play	Deviation Not Play	Chi-Square (Play)	Chi-Square (Not Play)
Above Average	8	6	7	7	1	-1	0.38	0.38
Below Average	2	4	3	3	-1	1	0.58	0.58

Steps to calculate Chi-Square for a split

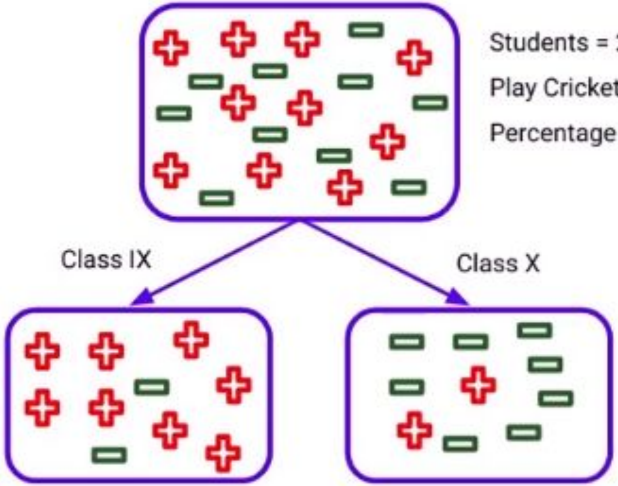
Split on Performance in Class

Total = 0.38+0.38+0.58+0.58 = 1.92



Steps to calculate Chi-Square for a split

Split on Class



Students = 20
Play Cricket = 10
Percentage = 50%

Students = 10

Expected Play = 5
Play Cricket = 8

Expected Not Play = 5
Not Play Cricket = 2

Students = 10

Expected Play = 5
Play Cricket = 2

Expected Not Play = 5
Not Play Cricket = 8

Steps to calculate Chi-Square for a split

Node	Actual Play	Actual Not Play	Expected Play	Expected Not Play	Deviation Play	Deviation Not Play	Chi-Square(Play)	Chi-Square(Not Play)
IX	8	2	5	5	3	-3	1.34	1.34
X	2	8	5	5	-3	3	1.34	1.34

Steps to calculate Chi-Square for a split

Split	Chi - Square
Performance in Class	1.92
Class	5.36

Pros / Cons of Decision Tree

Pros

- It is very interpretable, especially if we need to communicate our findings to a non-technical audience
- It deals well with noisy or incomplete data
- It can be used for both regression and classification problems

Pros / Cons of Decision Tree

Cons

- It can be unstable, meaning that a small change in your data can translate into a big change in your model
- It tends to overfit, which means low bias but high variance: i.e., might not perform as well on unseen data even if the score on the train data is great

Thank You!!!!!