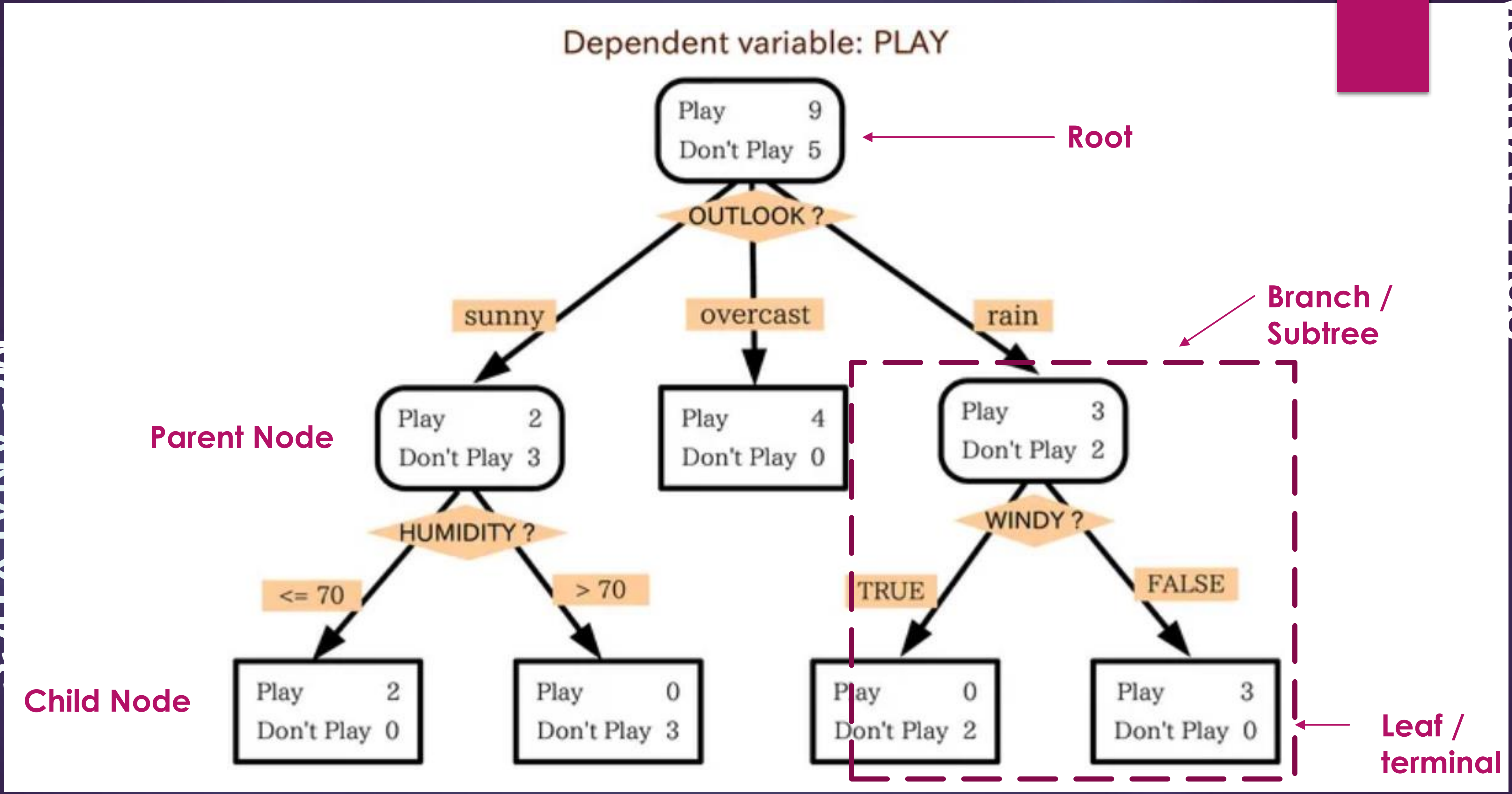
Decision Trees

BY MG ANALYTICS



MG ANALYTICS

IMPORTANT TERMS

- **Root Node:** It represents entire population or sample and this further gets divided into two or more homogeneous sets.
- **Splitting:** It is a process of dividing a node into two or more sub-nodes.
- **Decision Node:** When a sub-node splits into further sub-nodes, then it is called decision node.
- **Leaf/ Terminal Node:** Nodes do not split is called Leaf or Terminal node.
- **Branch / Sub-Tree:** A sub section of entire tree is called branch or sub-tree.
- **Parent and Child Node:** A node, which is divided into sub-nodes is called parent node of sub-nodes whereas sub-nodes are the child of parent node.

Process

- Trees are generated by selecting best rules possible at every split.
- Classification:
- Probability is obtained by proportion of values at leaf node
- Hard Class decisions are dependent on
  - majority vote.
  - Applying cut off on probability score

- Regression:
- Decision is dependent on average of target at terminal or leaf nodes
- Tree splitting is stopped when one of the set conditions is met.

Rules creation: Numeric Variables

- Numeric variables are split into random ranges to create rules.

Age	Rules
10	Age >10 , Age <10
13	Age > 13
5	Age <5
7	Age >7
17	Age<17
24	Age >24

MG ANALYTICS

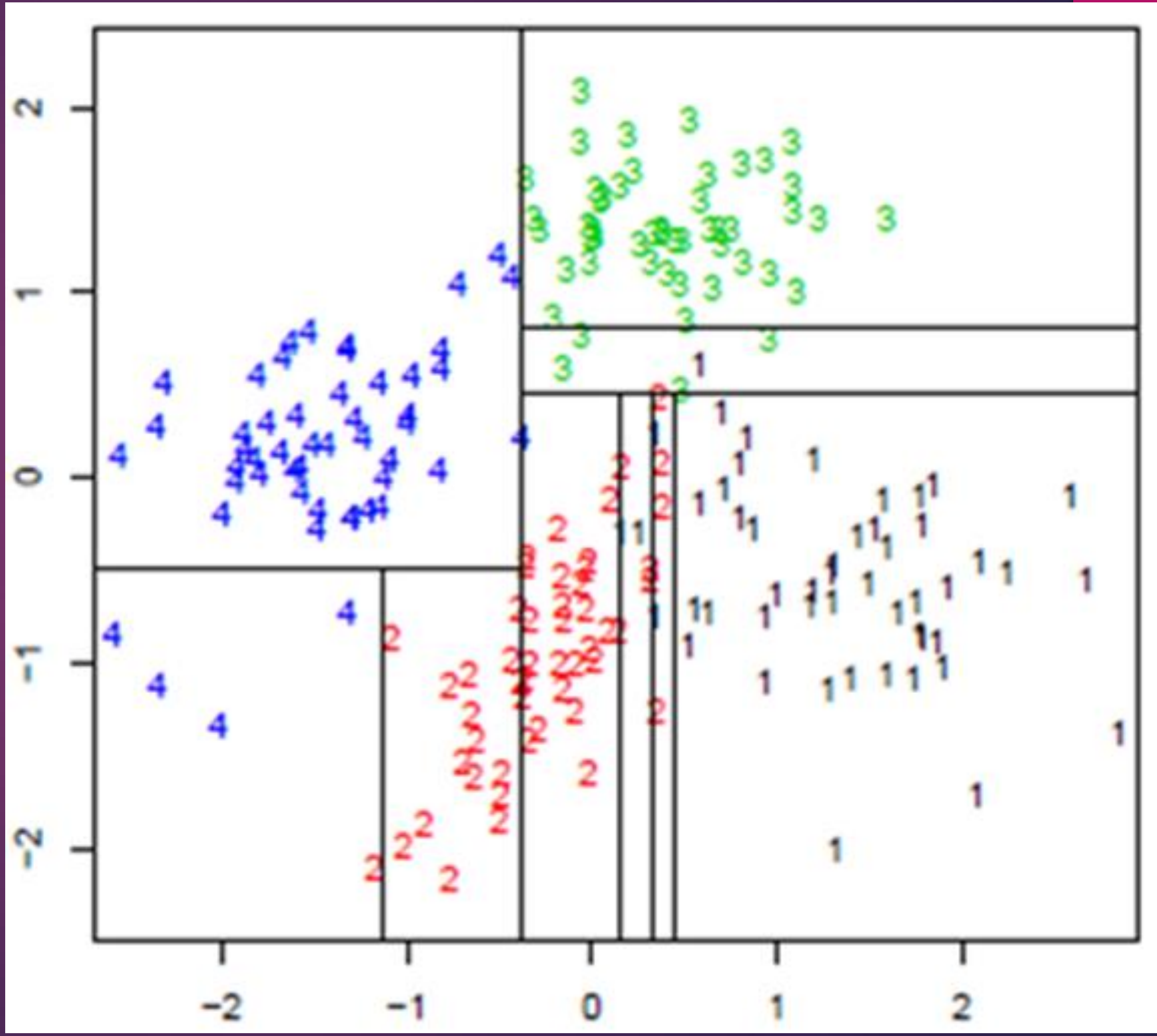
Rules creation: Categorical Variables

- Categorical variables are split on value >0.5 to create rules.
- Male > 0.5
- City_NY > 0.5
- City_Paris > 0.5

Male	City_NY	City_Paris
0	1	0
1	1	0
0	0	1
0	1	0
1	0	1
0	1	0

Axis Aligned Split

- Each split made would be aligned to an axis where each feature is an axis.



MG ANALYTICS

Rule Selection : Classification

- We are looking for a split which gives the most homogeneous child nodes.

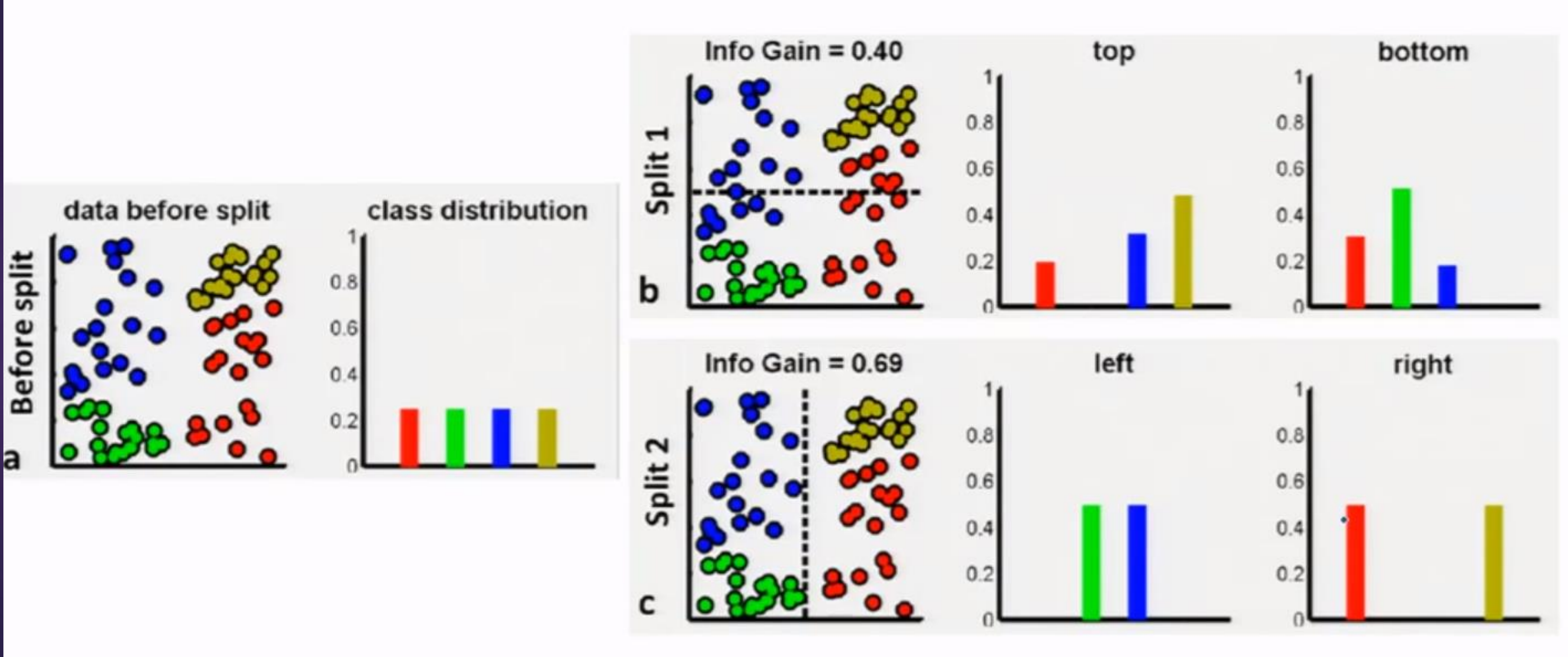
$$gini\ index = 1 - \sum_{i=1}^{k} p_i^2$$

$$entropy = -\sum_{i=1}^{k} p_i * log(p_i)$$

$$deviance = -\sum_{i=1}^{k} n_i * log(p_i)$$

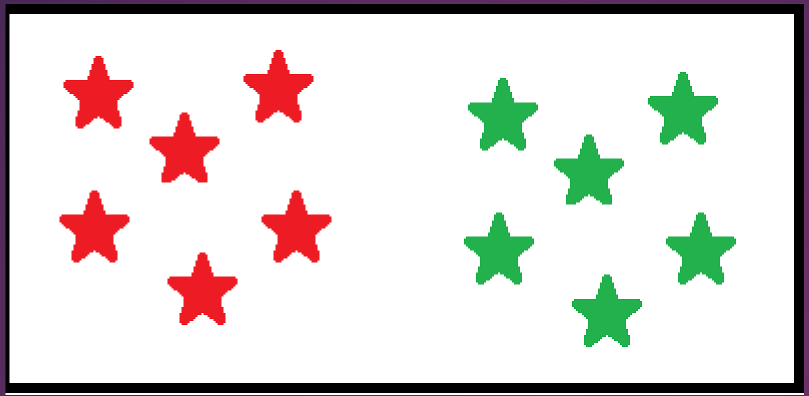
MG ANALYTICS

Using Information gain to Split

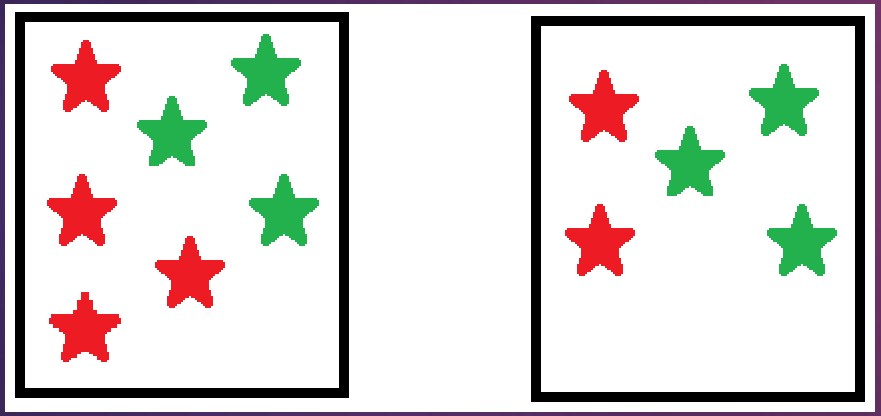


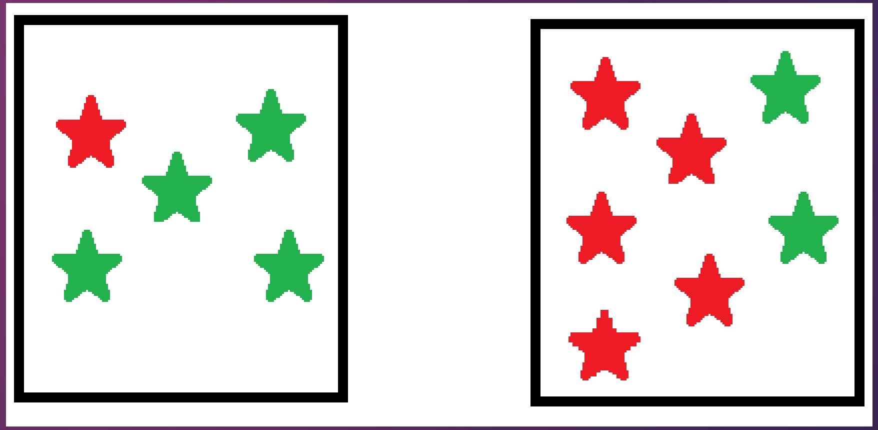
MG ANALYTICS

Probability of getting a result out of split



6/12 = .50





3/7 = .42

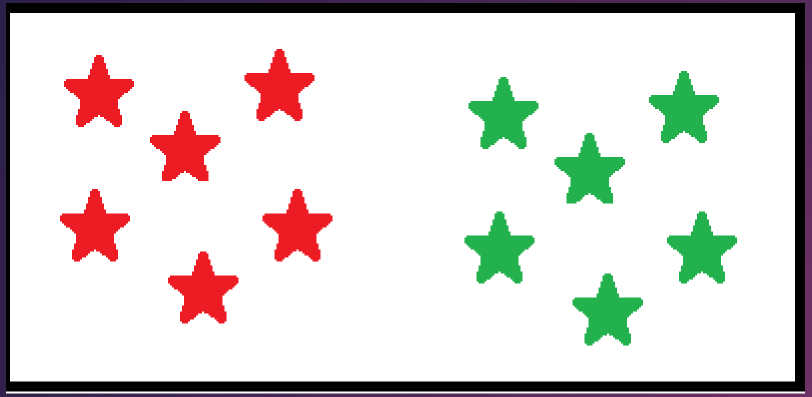
3/5 = .60

4/5 = 0.80

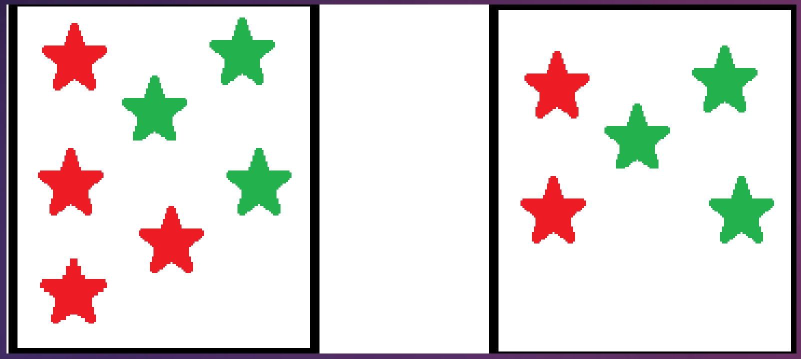
2/7 = .28

MG ANALYTICS

Information Gain Using Gini index

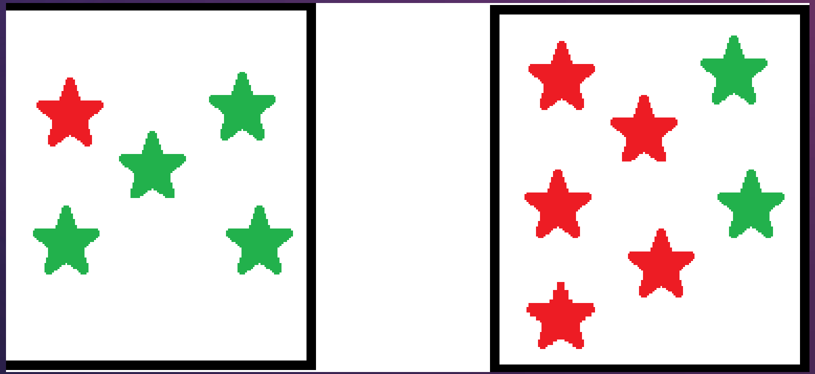


Parent Gini : $1 - (6/12)^2 + (6/12)^2 = \mathbf{0.5}$



Split 1: $GINI\_1 = 1 - (4/7)^2 + (3/7)^2 = 0.4898$
: $GINI\_2 = 1 - (2/5)^2 + (3/5)^2 = 0.480$

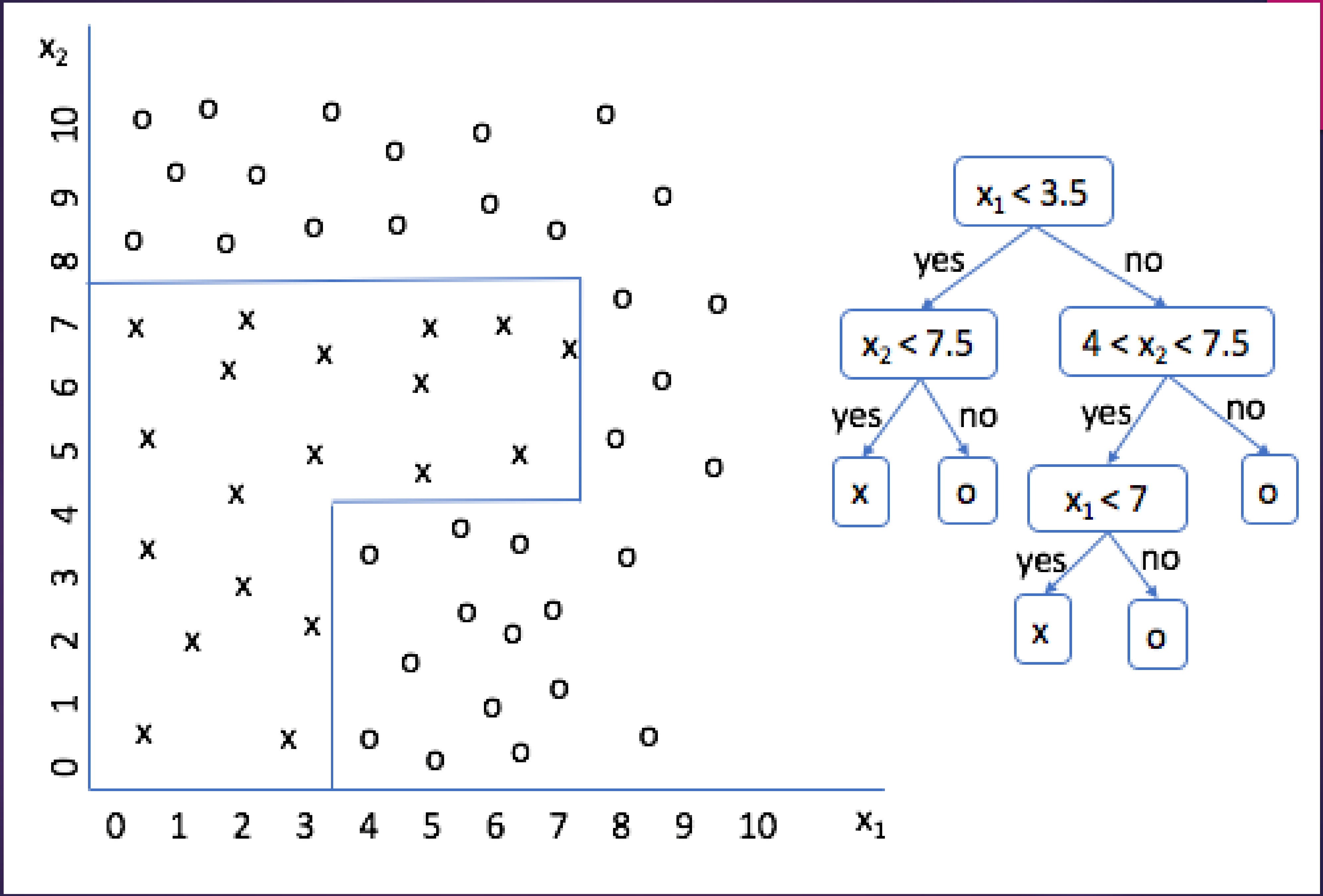
Wght **avg of GINIs** = $(7/12)*(0.4898) + (5/12)*(0.480) = \mathbf{0.486}$
GAIN = $0.5 – 0.486 = \mathbf{0.014}$



Split 2: $GINI\_1 = 1 - (4/5)^2 + (1/5)^2 = 0.320$
: $GINI\_2 = 1 - (5/7)^2 + (2/7)^2 = 0.4082$

Wght **avg of GINIs** = $(5/12)*(0.320) + (7/12)*(0.4082) = \mathbf{0.3715}$
GAIN = $0.5 – 0.3715 = \mathbf{0.1285}$

MG ANALYTICS



MG ANALYTICS

Regression Trees

- we are collecting very similar records at each leaf. So, we can use median or mean of the records at a leaf as the predictor value for all the new records that obey similar conditions.

- Such trees are called regression trees.

MG ANALYTICS

Rule Selection: Regression

- In case of regression the original SSE is compared with the sum of SSEs after the splits and the one with minimum is selected.

TARGET	PREDICTED	ERROR	Sqr(Error)
5	8.63	3.63	13.1769
6	8.63	2.63	6.9169
4	8.63	4.63	21.4369
6	8.63	2.63	6.9169
11	8.63	-2.37	5.6169
12	8.63	-3.37	11.3569
13	8.63	-4.37	19.0969
12	8.63	-3.37	11.3569
		SSE	**95.8752**

TARGET	PREDICTED	ERROR	Sqr(Error)
5	8.75	3.75	14.0625
6	8.75	2.75	7.5625
12	8.75	-3.25	10.5625
12	8.75	-3.25	10.5625
			42.75

TARGET	PREDICTED	ERROR	Sqr(Error)
11	8.5	-2.5	6.25
6	8.5	2.5	6.25
13	8.5	-4.5	20.25
4	8.5	4.5	20.25
			53

SSE = 42.75+53
= 95.75

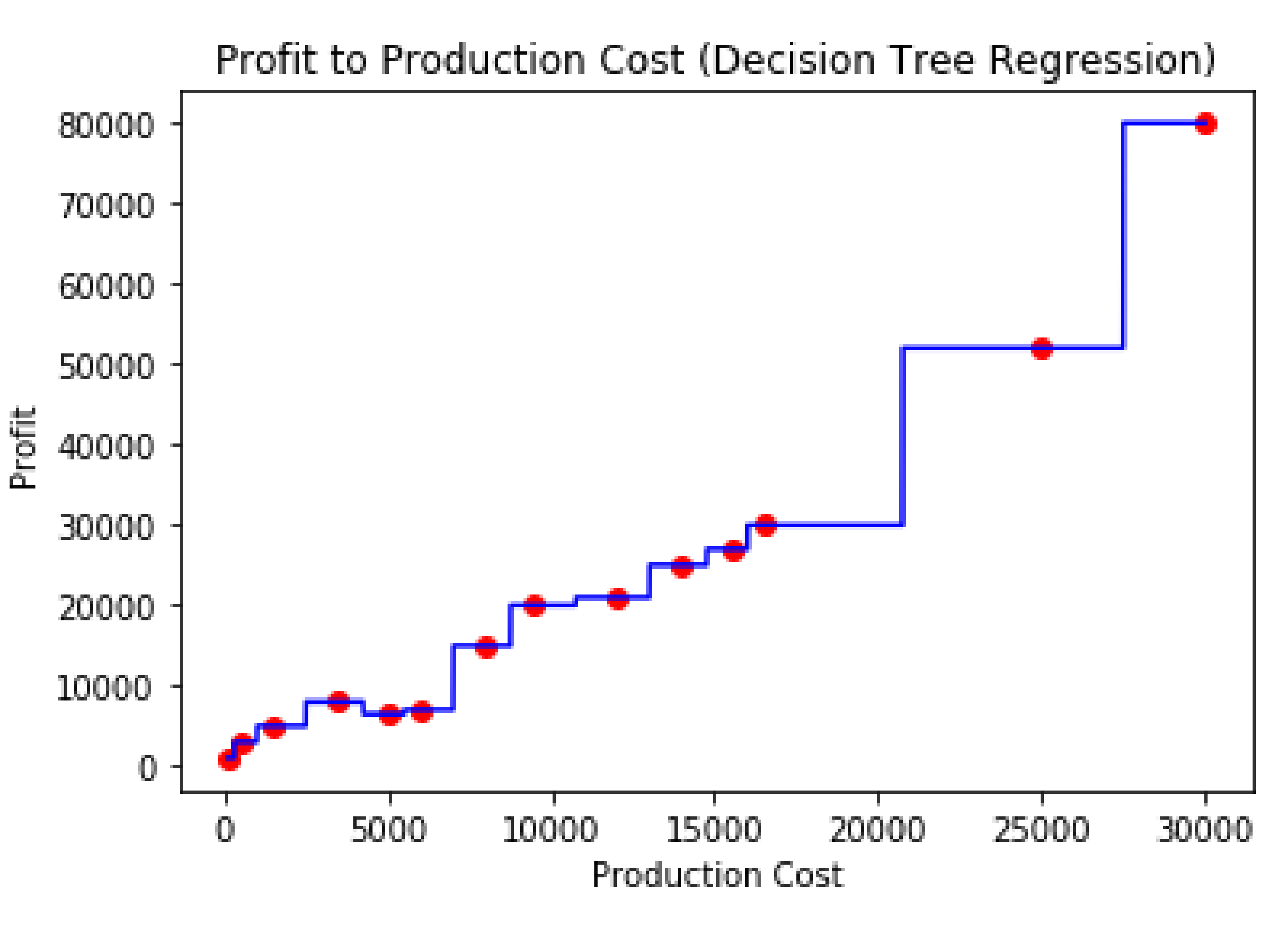
MG ANALYTICS

TARGET	PREDICTED	ERROR	Sqr(Error)
5	5.25	0.25	0.0625
6	5.25	-0.75	0.5625
6	5.25	-0.75	0.5625
4	5.25	1.25	1.5625
			2.75

TARGET	PREDICTED	ERROR	Sqr(Error)
11	12	1	1
12	12	0	0
13	12	-1	1
12	12	0	0
			2

SSE = 2.75 + 2
= 4.75

MG ANALYTICS



MG ANALYTICS

MG ANALYTICS

When to stop splitting:

1. Node becomes homogeneous.
2. Maximum depth of tree is reached.
3. Maximum number of leaf node limit is reached.
4. Leaf node has observations less than lower limit of values present.
5. The split results in leaf node having less than lower limit of values.

MG ANALYTICS

Hyperparameters :

- **criterion**{“gini”, “entropy”}, default=”gini”

The function to measure the quality of a split. Supported criteria are “gini” for the Gini impurity and “entropy” for the information gain.

- **Splitter**{“best”, “random”}, default=”best”

The strategy used to choose the split at each node. Supported strategies are “best” to choose the best split and “random” to choose the best random split.

- **max_depth**: int, default=None

The maximum depth of the tree. If None, then nodes are expanded until all leaves are pure or until all leaves contain less than min_samples_split samples.

MG ANALYTICS

- **min_samples_split**: int or float, default=2

  The minimum number of samples required to split an internal node:
- **min_samples_leaf**: int or float, default=1

  The minimum number of samples required to be at a leaf node.
- **max_features**: int, float or {“auto”, “sqrt”, “log2”}, default=None

  The number of features to consider when looking for the best split.
- **max_leaf_nodes :** Grow a tree with max_leaf_nodes in best-first fashion.
- **class_weight**: dict, list of dict or “balanced”, default=None

  Weights associated with classes in the form {class_label: weight}.

MG ANALYTICS

Overfitting

- Dtree's overfit due to :
  - high depth
  - Noisy observations
  - Noisy Variables

MG ANALYTICS