DT Algorithm

Be ready for

DT

Python 1 & 2

Session

Decision Tree Theory

Hyperparameters:

“min_samples_split” ***parameter:*** **(default=2)**

The minimum number of samples required to split an internal node.

“min_samples_leaf” ***parameter:*** **(default=1)**

The minimum number of samples required to be at leaf node.

Decision Tree Theory

Hyperparameters:

“splitter” *parameter:* (default=”best)

The strategy used to choose the split each node. (“best”, “random”)

“max_features” *parameter:* (default=None)

Number of features to consider when looking for the best split.

Decision Tree Theory

Hyperparameters:

“criterion” ***parameter:*** **(default=”gini”)**

The function to measure the quality of a split.

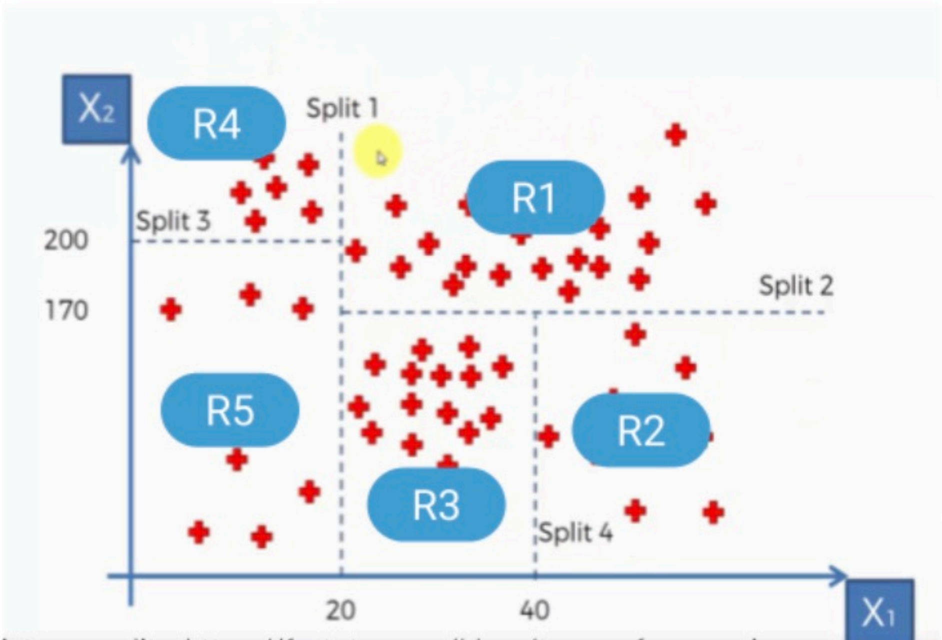
“max_depth” ***parameter:*** **(default=None)**

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_sample_split samples.

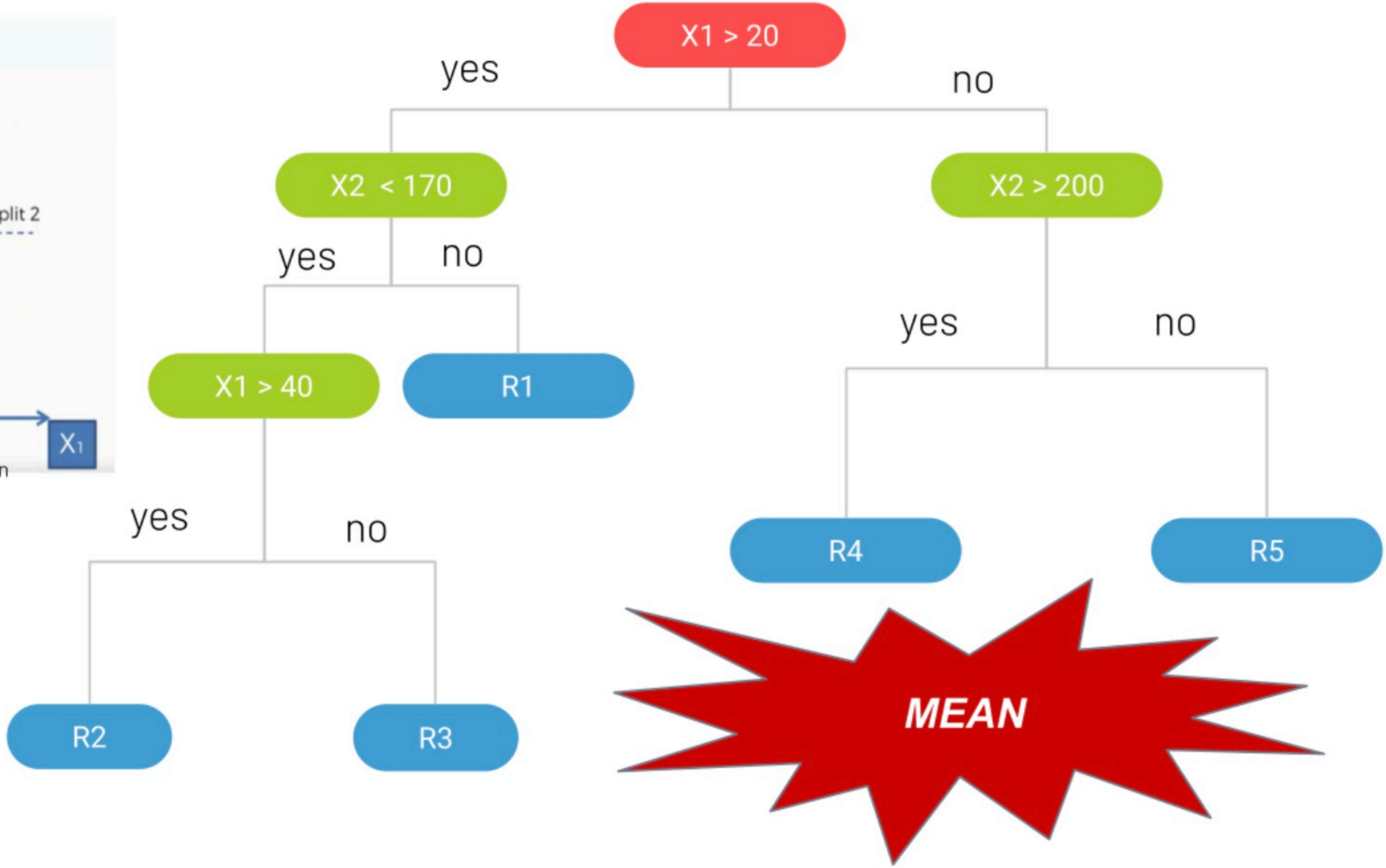
Decision Tree Theory

Regression- Variance



Picture credit : https://favtutor.com/blogs/types-of-regression

If the numerical sample is completely homogeneous its standard deviation is zero.

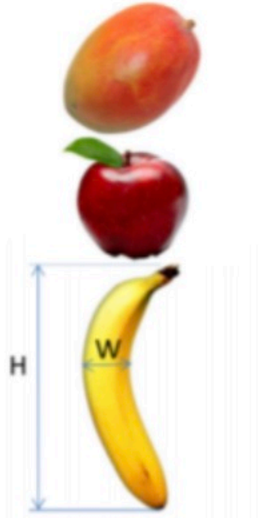


Mean is the value in the leaf nodes.

Decision Tree Theory

Information Gain - Entropy

- 2 features: width, height
- 3 classes: mango, apple, banana
- Data:



ID	Height (cm)	Width (cm)	Class
1	9	6	Mango
2	6	5.3	Apple
3	18	3.2	Banana
4	6.7	6.2	Apple
5	20	4	Banana
6	15	3.1	Banana
7	9	7.5	Apple
8	11.5	7.5	Mango
9	11	2.5	Banana
10	13	9	Mango

- Let's take the "height" with a threshold of "10 cm"

Height > 10 => Long
Height ≤ 10 => Short

$$IG(S) = I(S) - \frac{N_{left}}{N} I(S_{left}) - \frac{N_{right}}{N} I(S_{right})$$

entire dataset

$$IG(\textbf{Height}) = \overset{B\,M\,A}{I(4,3,3)} - \frac{6}{10}\overset{B\,M\,A}{I(4,2,0)} - \frac{4}{10}\overset{B\,M\,A}{I(0,1,3)}$$

$$I(4,3,3) = -(4/10)\log(4/10) - (3/10)\log(3/10) - (3/10)\log(3/10) = 1.571$$

$$I(4,2,0) = -(4/6)\log(4/6) - (2/6)\log(2/6) - 0 = 0.918$$

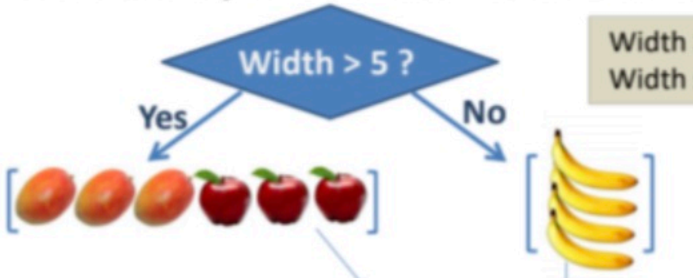
$$I(S) = -\sum_{i=1}^{N} p_i \log_2(p_i)$$

$$I(0,1,3) = 0 - (1/4)\log(1/4) - (3/4)\log(3/4) = 0.811$$

$$IG(\textbf{Height}) = 1.571 - \frac{6}{10}0.918 - \frac{4}{10}0.811 = 0.696$$

Root Node

- Now we try the "Width" with a threshold of "5"



Width > 5.0 => Wide
Width ≤ 5.0 => Narrow

$$IG(S) = I(S) - \frac{N_{left}}{N} I(S_{left}) - \frac{N_{right}}{N} I(S_{right})$$

$$IG(\textbf{Width}) = \overset{B\,M\,A}{I(4,3,3)} - \frac{6}{10}\overset{B\,M\,A}{I(0,3,3)} - \frac{4}{10}\overset{B\,M\,A}{I(4,0,0)}$$

I(4,3,3) computed before

$$I(0,3,3) = 0 - (3/6)\log(3/6) - (3/6)\log(3/6) = 1.0$$

$$I(4,0,0) = (4/4)\log(4/4) - 0 - 0 = 0.0$$

$$IG(\textbf{Width}) = 1.571 - \frac{6}{10}1.0 - \frac{4}{10}0.0 = 0.971$$

results

	Width > 5 ?	Height > 10 ?
Information Gain:	0.971	0.696

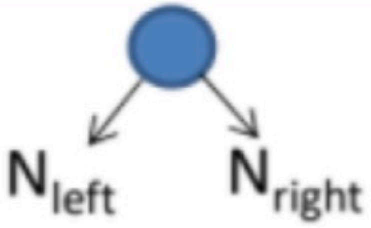
- Lower entropy => Higher gain
- Feature "Width of 5.0" creates a better gain
- Now we can proceed with the left branch of "Width > 5.0?" and split it further
- **Question:** How do we know that splitting with another threshold (for either feature) won't yield a better solution (higher gain)?

Decision Tree Theory

Information Gain - Entropy

For a **decision algorithm**, we start at the root node and split the data on the feature that results in **<u>the largest</u> Information Gain (IG)**

$$IG(S) = I(S) - \frac{N_{left}}{N} I(S_{left}) - \frac{N_{right}}{N} I(S_{right})$$



Entropy of the original (parent) collection before split

Number of Data points

Entropy of the left and right child collection

Decision Tree Theory

Information Gain - Entropy

Entropy of a group in which **all examples** belong to the **same class.**

$$H(S) = -1\log_2(1) = 0$$ (perfect purity, no uncertainty)

Entropy of a group with **50% in either class**:

$$H(S) = -0.5\log_2(0.5) - 0.5\log_2(0.5) = 1$$ (maximum uncertainty)

Decision Tree Theory

Information Gain - Entropy

Another criterion is **"Information Gain"** based on a purity measure called **Entropy*.**

Entropy Formula

# of classes

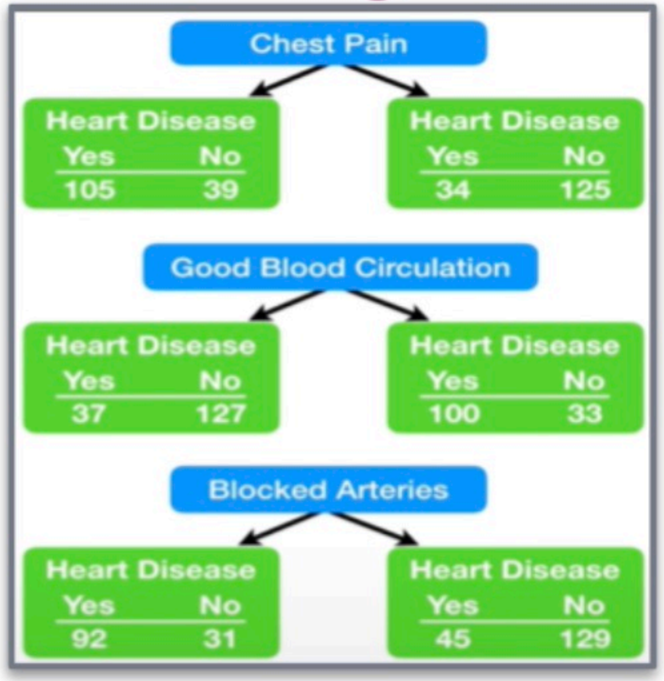
$$\underbrace{H}_{\text{Entropy}} = -\sum_{i=1}^{n} P(x_i)\log_2 \underbrace{P(x_i)}_{\text{Probability of class "i"}}$$

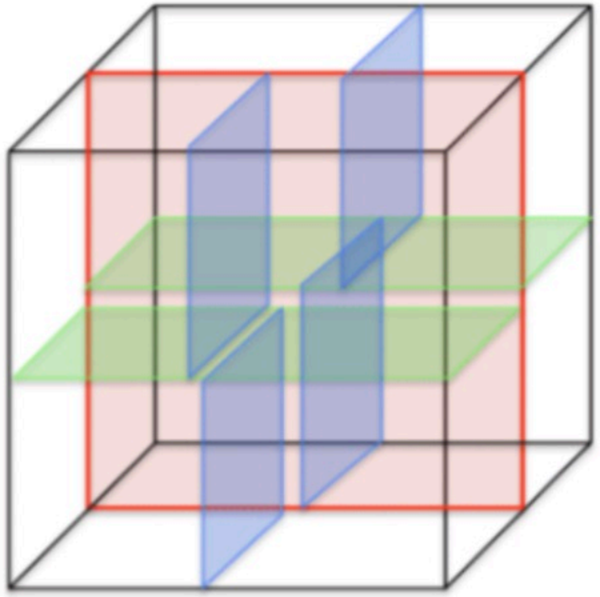
***Entropy** is a measure of uncertainty.

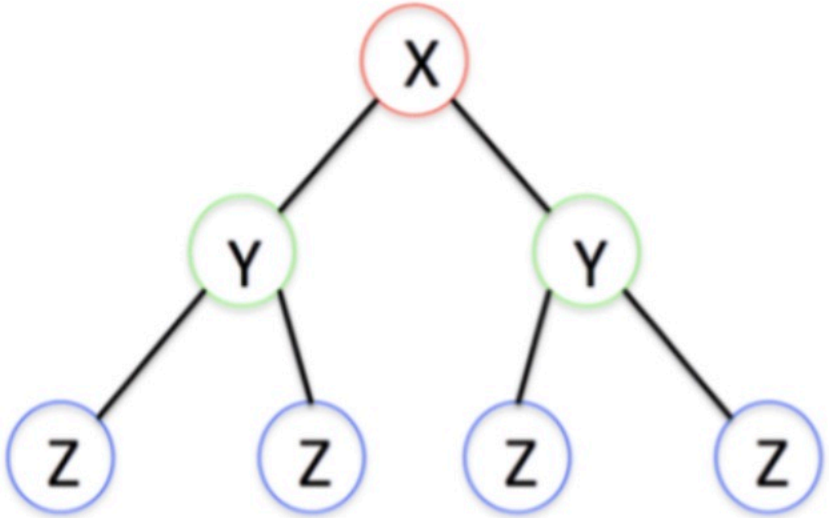
Decision Tree Theory

Gini Index Theory

Chest Pain	Good Blood Circulation	Blocked Arteries	Heart Disease
No	No	No	No
Yes	Yes	Yes	Yes
Yes	Yes	No	No
Yes	No	???	Yes
etc...	etc...	etc...	etc...





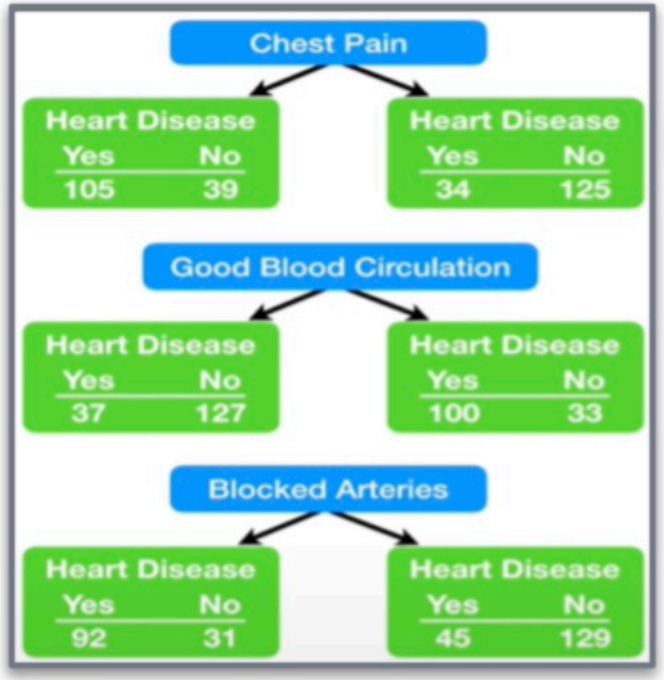


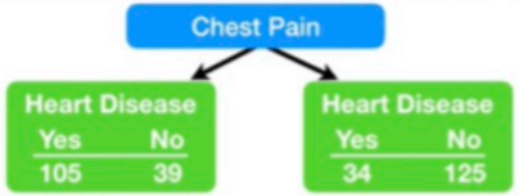
Sample from Statquest

Decision Tree Theory

Gini Index Theory

Chest Pain	Good Blood Circulation	Blocked Arteries	Heart Disease
No	No	No	No
Yes	Yes	Yes	Yes
Yes	Yes	No	No
Yes	No	???	Yes
etc...	etc...	etc...	etc...

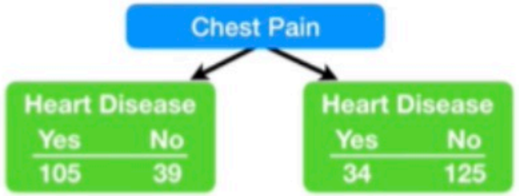




For this leaf, the Gini impurity = 1 - (the probability of "yes")^2 - (the probability of "no")^2

$$= 1 - \left(\frac{105}{105 + 39}\right)^2 - \left(\frac{39}{105 + 39}\right)^2$$

$$= 0.395$$



Gini impurity = 0.395 0.336

Gini impurity for Chest Pain = weighted average of Gini impurities for the leaf nodes

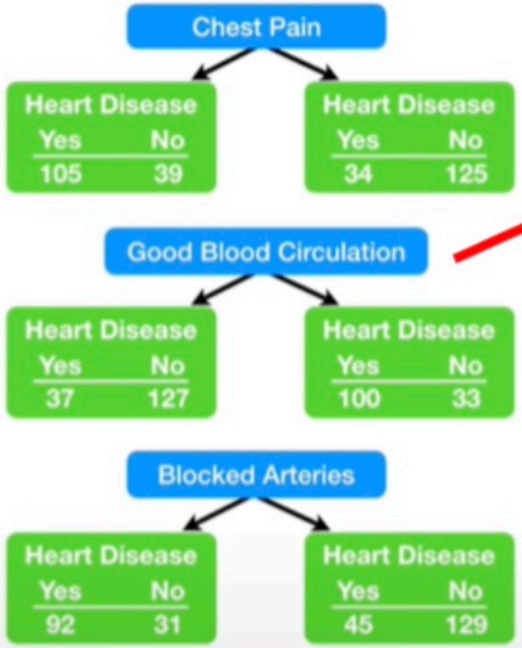
$$= \left(\frac{144}{144 + 159}\right) 0.395 + \left(\frac{159}{144 + 159}\right) 0.336$$

$$= 0.364$$

Gini impurity for Chest Pain = 0.364

Gini impurity for Good Blood Circulation = 0.360

Gini impurity for Blocked Arteries = 0.381



Root Node

Sample from Statquest

Decision Tree Theory

Gini Index Theory

Gini index (or Gini impurity index) is a criterion to *minimize* the *probability of misclassification.*

Gini Index Formula

$$Gini = 1 - \sum_{i=1}^{n} \underline{P^2(x_i)}$$

# of classes

Probability of class "i"

Decision Tree Theory

*– Which attribute to start with at the **main root ?***

*– **Where to split** the attribute from ?*

– What is the sequence of other attributes down the tree?

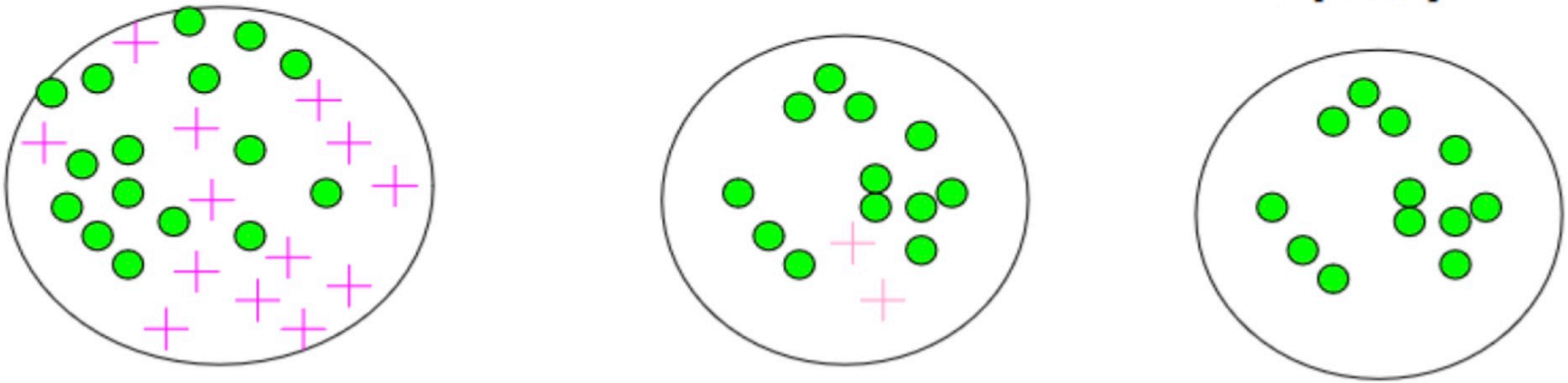
*We need some **sort of a measure** to decide which attribute to **start splitting** with.*

Gini Index (Gini Impurity Index)	Information Gain / Entropy

Decision Tree Theory

Impurity

Very impure group | **Less impure** | **Minimum impurity**



Decision Tree Theory

Main Question:

How do we determine how the tree will be split?

*The most important criterion when dividing is to ensure the **highest homogeneity** in the sub-decision nodes and to ensure that the leaf node is **pure** in terms of target variables.*

Other Questions:

– *Which attribute to start with at the **main root?***

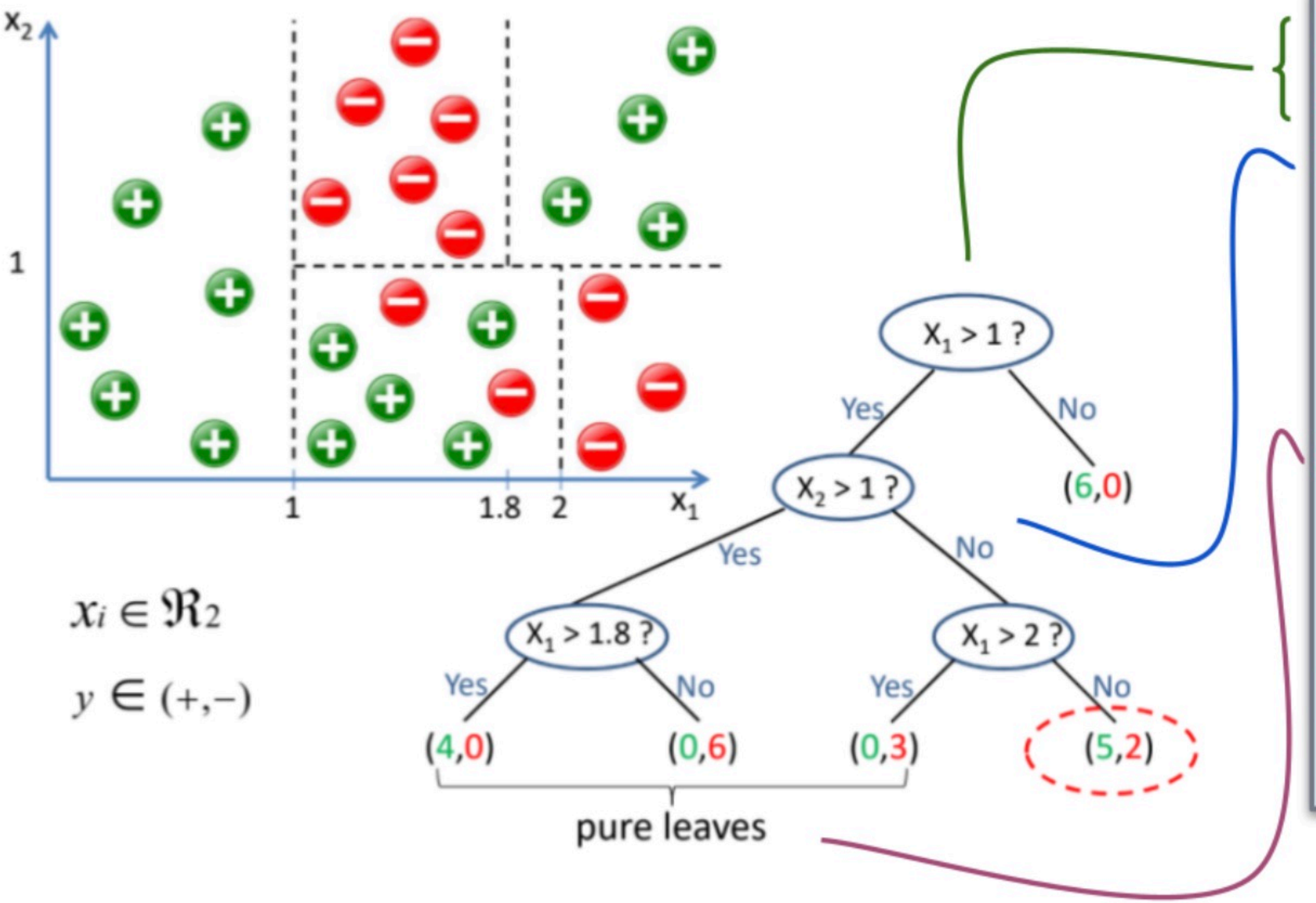
– ***Where to split** the attribute from?*

– *What is the sequence of other attributes down the tree?*

– *When to **stop branching** the tree?*

Decision Tree Theory

Decision Tree Algorithm Working Cycle:



0. Start at the root (with all training examples)

1. Select an attribute that best separates the classes

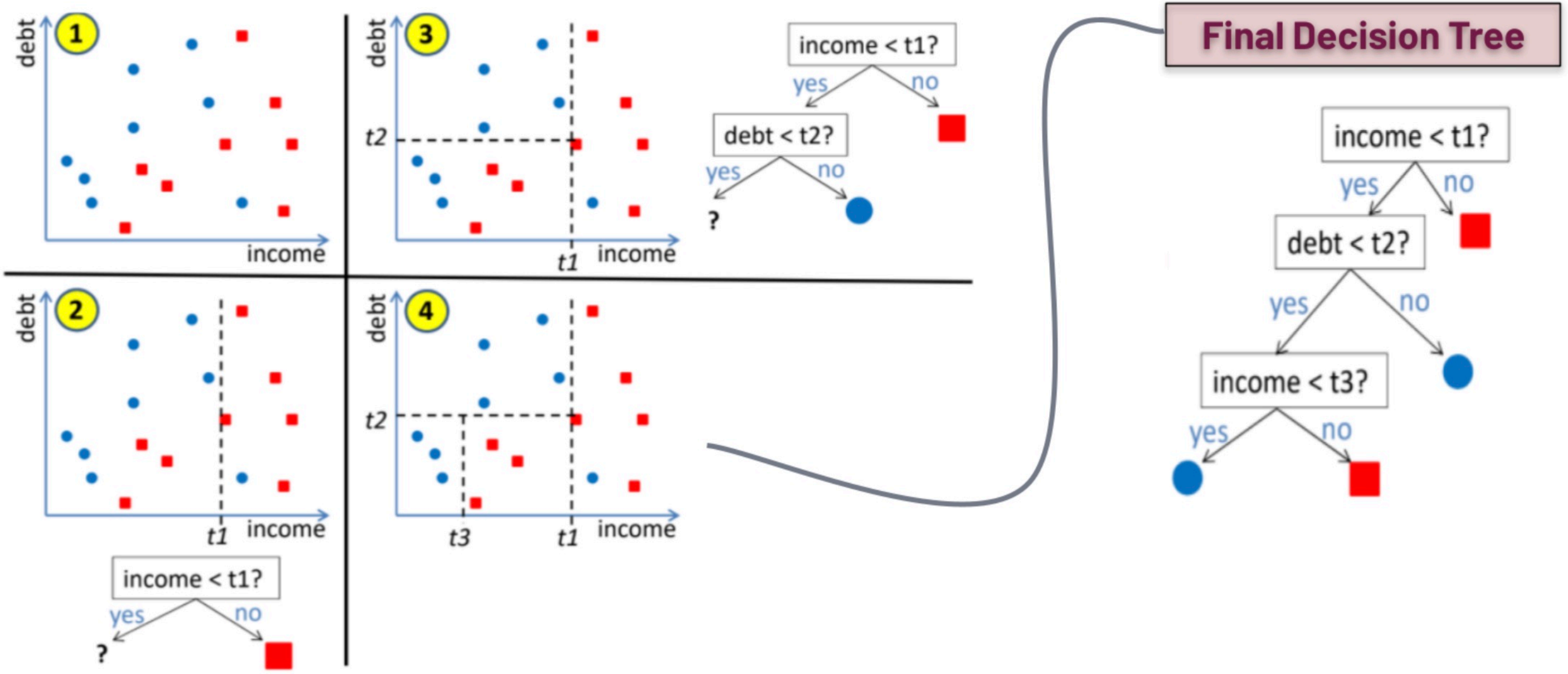
2. Split it into subsets (child nodes)
- If the attribute value is categorical:
  => select category(ies)
- If the attribute is numerical:
  => select a threshold

3. Are they pure (all samples from the same class)?
- If yes: stop (training set is perfectly classified)
- If no: select an attribute and split further, go to (3)

Classification: When you have a new data point, start at the root and traverse down the tree to get to the subset this new data point belongs to

Decision Tree Theory

DT Sample :

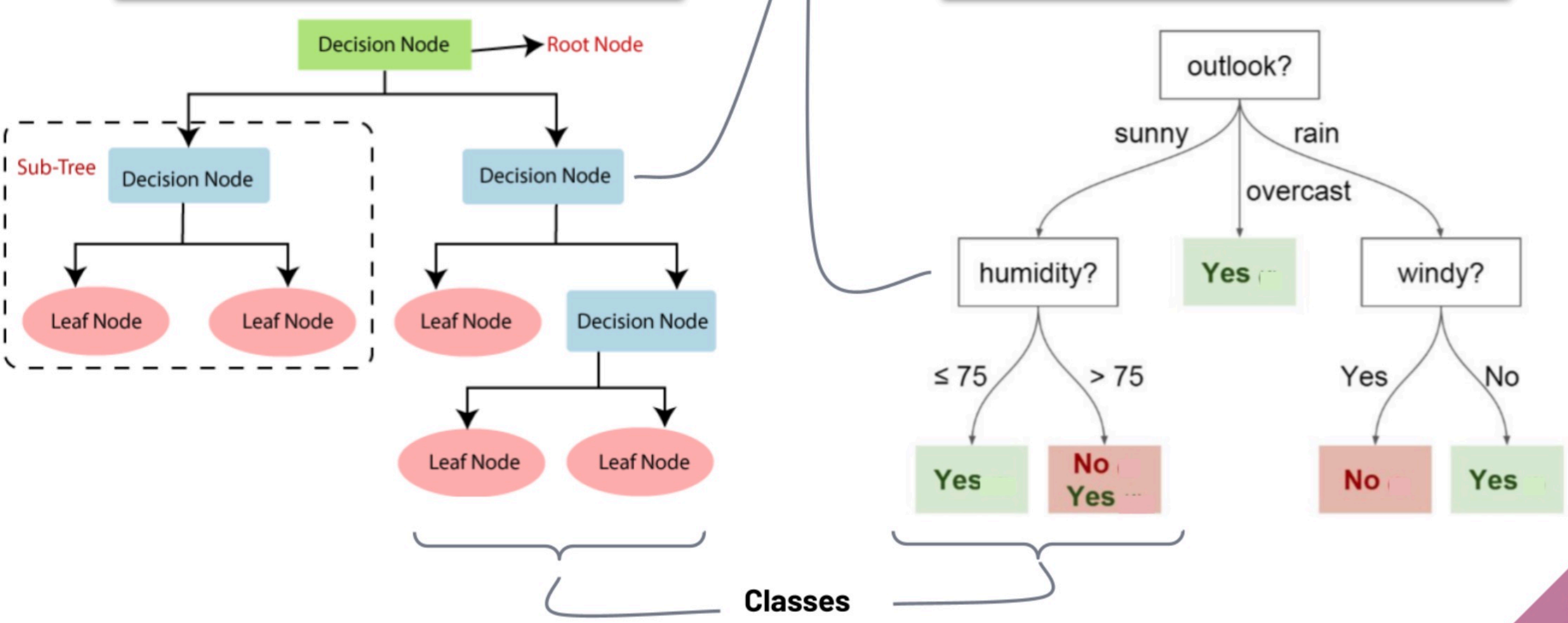


Decision Tree Theory

Question: Will your friends come to the picnic?

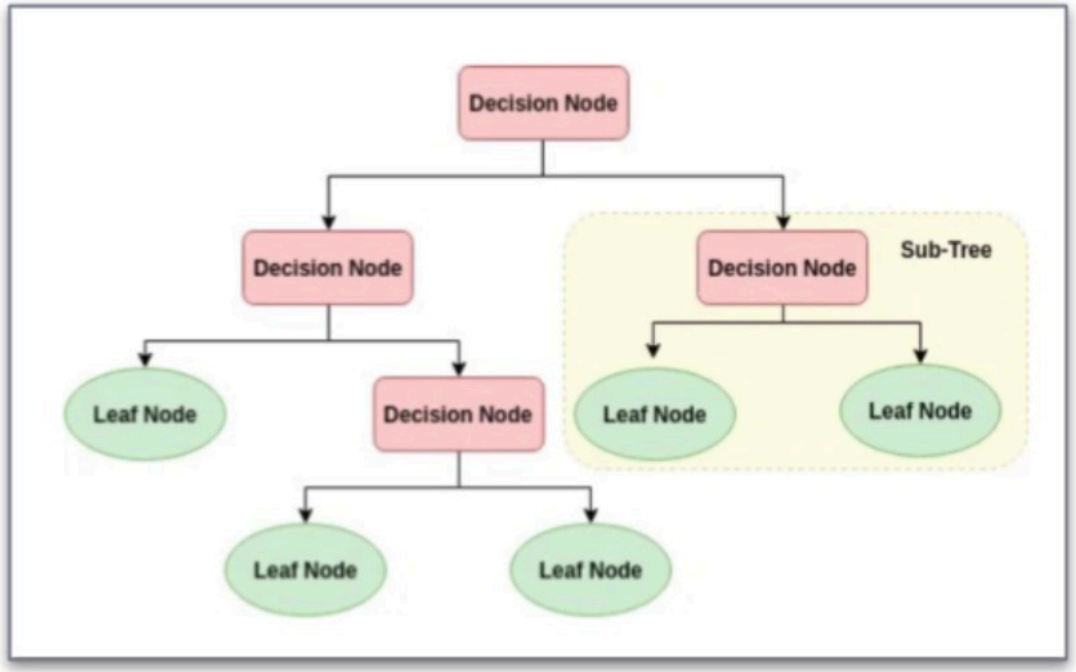
Decision Tree Diagram

Decision Tree Sample



Decision Tree Theory

DT is one of the **Tree-based machine learning algorithms that** are considered as the ***most widely used*** and ***successful*** algorithms.



Usage areas of Decision Tree classification problems:

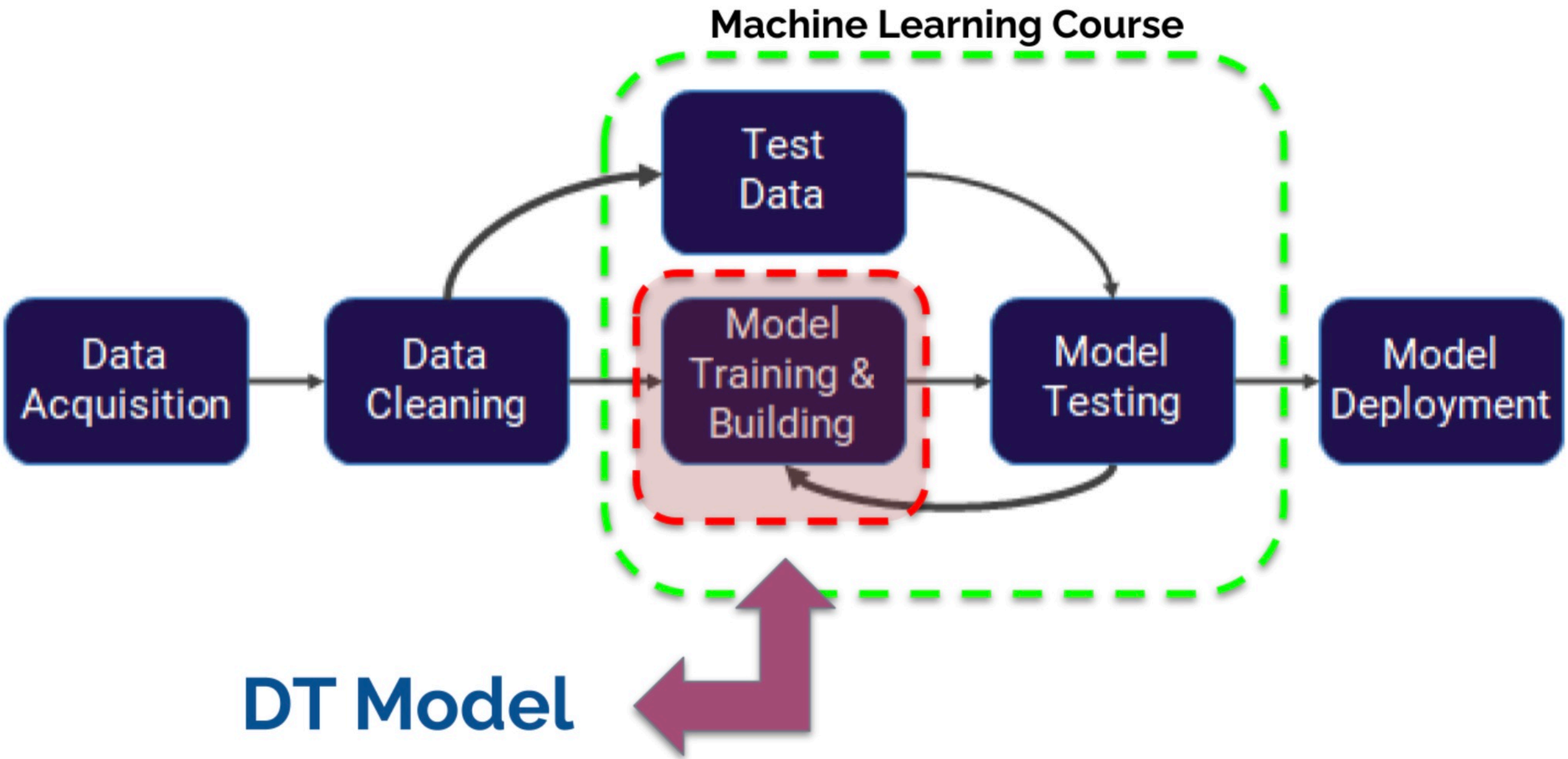
* Medical Diagnosis,

* Text Classification,

* Credit Risk Analysis etc.

Decision Tree Theory

Where are we?



Machine Learning Course
Test Data
Data Acquisition
Data Cleaning
Model Training & Building
Model Testing
Model Deployment
DT Model

Decision Tree Theory

Classification

Gini Index

Information Gain-Entropy

Regression

DT with Python

SUMMARY of PREVIOUS CLASS

- **SVM: Support vectors, Hyperplane**
- **Hard margin, soft margin classification, "C" parameter**
- **Kernel Trick**
- **Kernels:** {'linear', 'poly', 'rbf', 'sigmoid'}
- **"gamma" parameter:** Higher gamma, higher overfitting risk

Decision Tree Theory

Session-12