UNIVERSIDADE CATÓLICA PORTUGUESA

BRAGA

Machine Learning

Session 12 - T

Tree-Based Models – Part 1

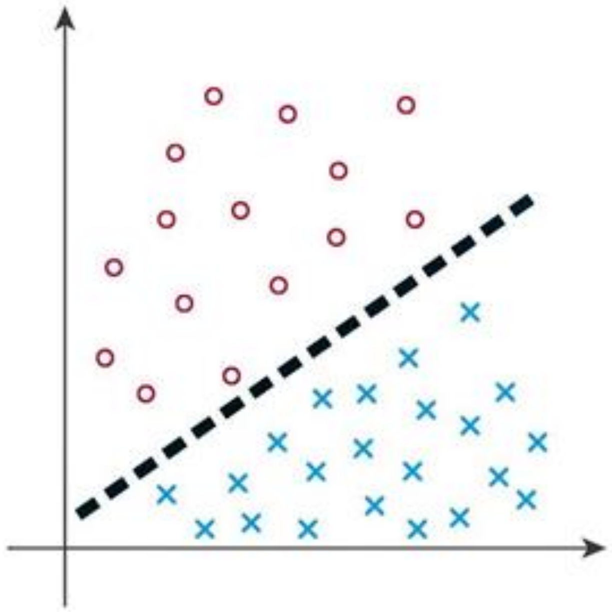
Ciência de Dados Aplicada

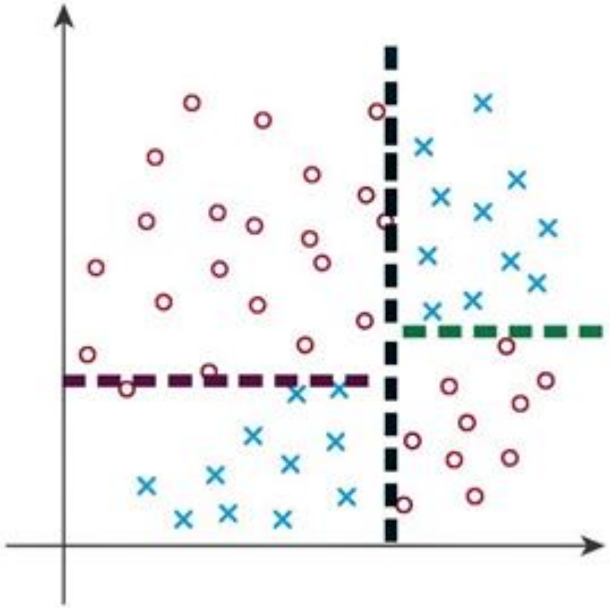
2023/2024

Feature Space

- **Linearly separable data** – the feature space can be well separated by a line or hyperplane;

- **Linearly inseparable** data – the feature space cannot be effectively divided by a single line or hyperplane.





Note that the classes are still well separated in the feature space, but the **decision boundaries cannot be described by single linear equations**.

Feature Space

UNIVERSIDADE CATÓLICA PORTUGUESA
BRAGA

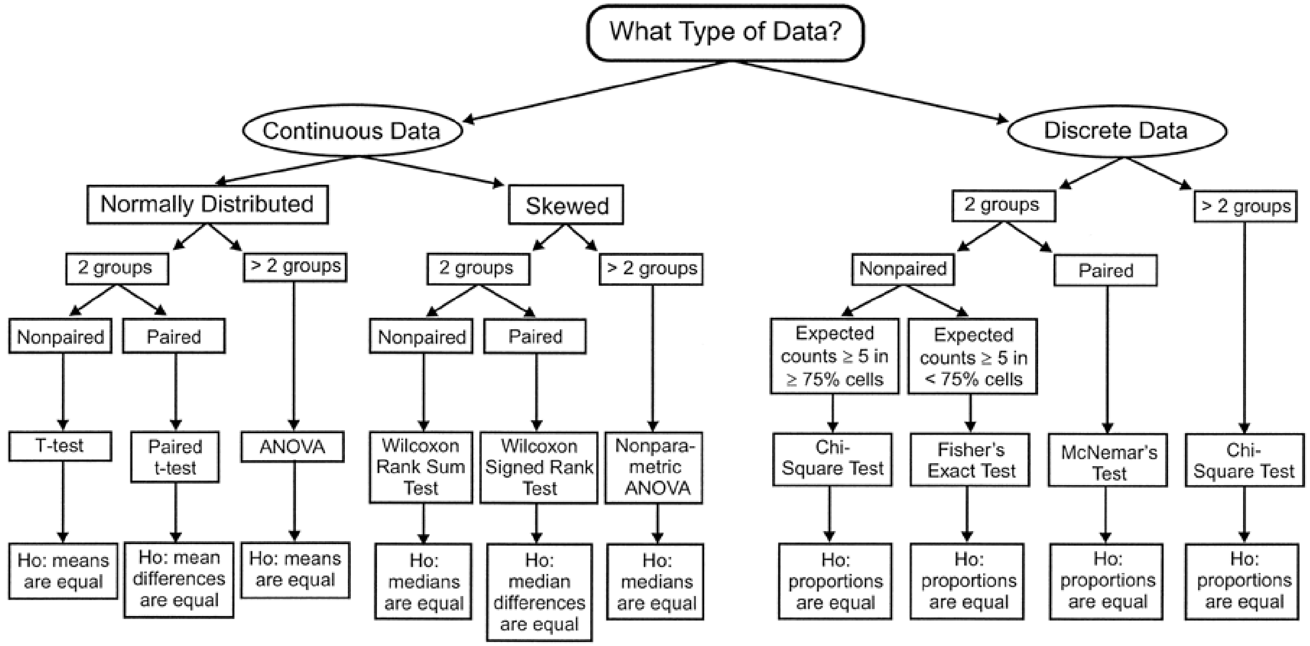
- Although linear models with linear boundaries offer intuitive interpretation, interpreting nonlinear decision boundaries presents challenges.

- Therefore, there is a need to build models that:
  - allow **complex decision boundaries**;
  - are **easy to interpret**.

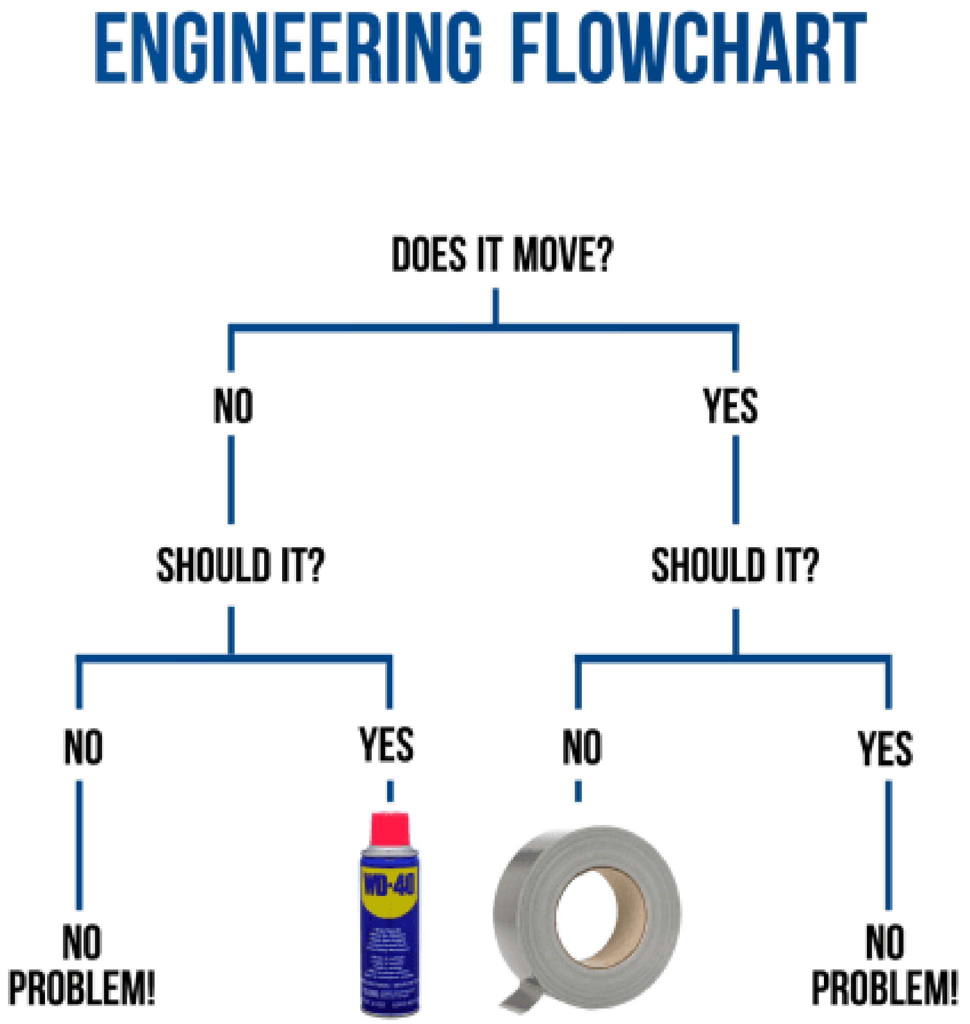
Interpretable Models

UNIVERSIDADE CATÓLICA PORTUGUESA
BRAGA

- People from diverse backgrounds have historically relied on **interpretable models** to distinguish between various classes of objects and phenomena.



Source: Waning B, Montagne M: *Pharmacoepidemiology: Principles and Practice*: http://www.accesspharmacy.com



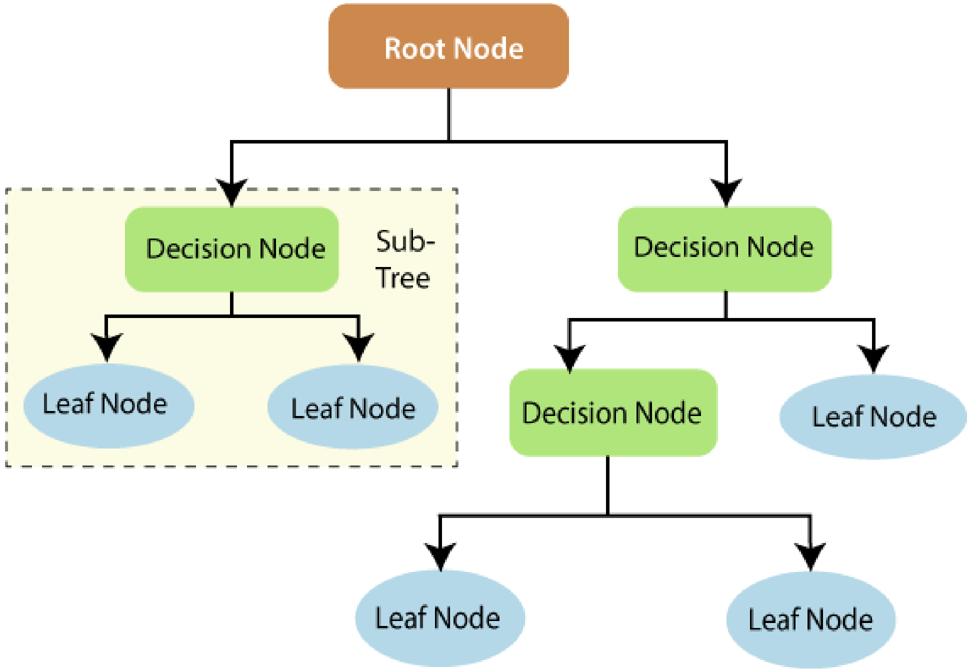
Tree-Based Models

- Flow charts like in the previous examples can be formulated as mathematical models (**graphs**) for classification and regression.

- These models are:
  - **Interpretable** by humans;
  - Have **complex decision boundaries**;
  - The decision boundaries are a **combination of linear boundaries** that are **mathematically simple to describe**.

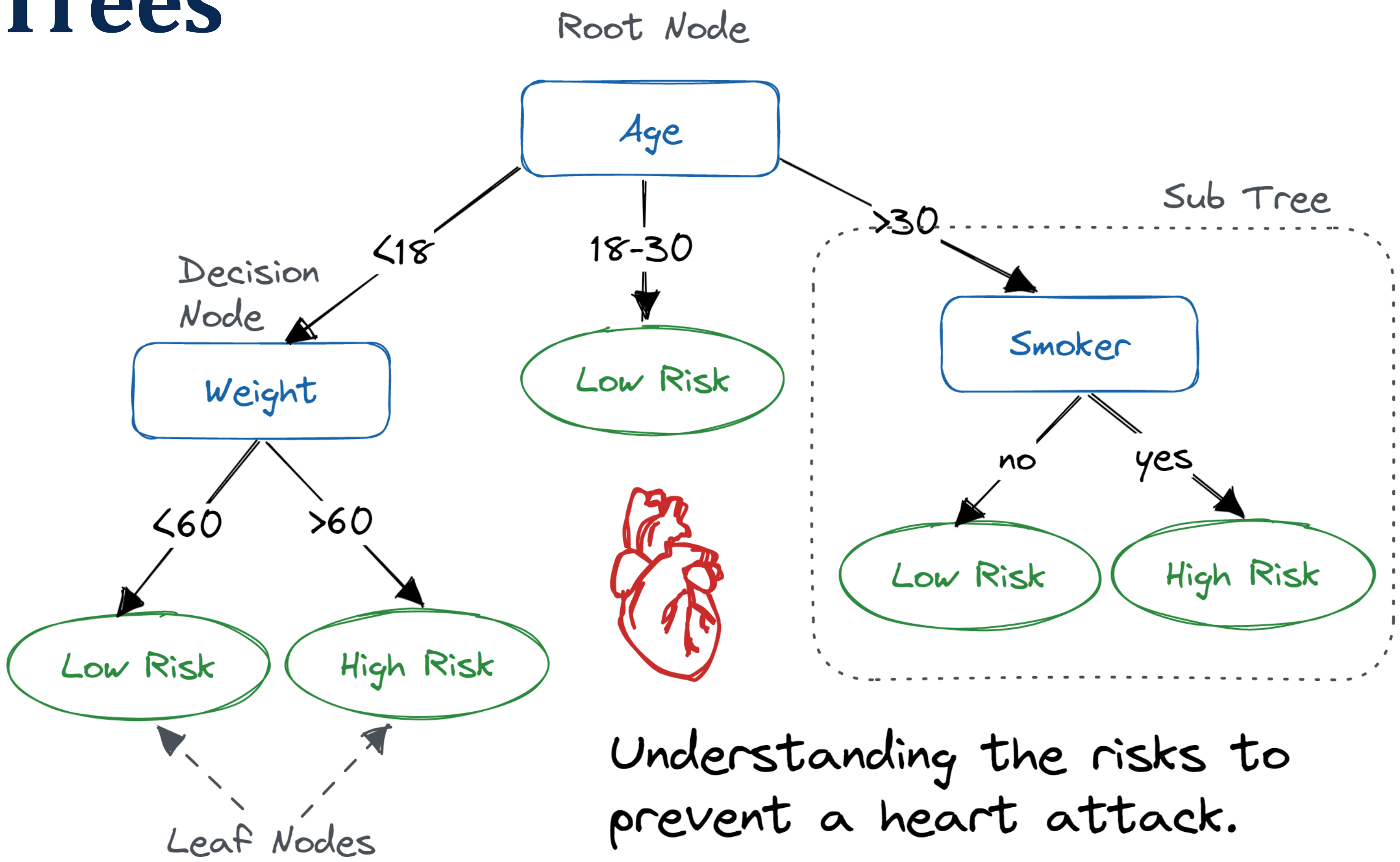
Decision Trees

- Mathematically, a decision tree can be defined as a **directed acyclic graph**, comprising:
  - **Nodes:** Represent decision points or conditions.
  - **Edges:** Connect nodes and represent the outcomes of decisions.
  - **Root Node:** The initial decision point, representing the entire dataset.
  - **Decision Nodes:** Decision points where a split is made based on a feature or attribute.
  - **Leaf Nodes:** Terminal nodes representing final outcomes or predictions.



Decision Trees

UNIVERSIDADE CATÓLICA PORTUGUESA
BRAGA

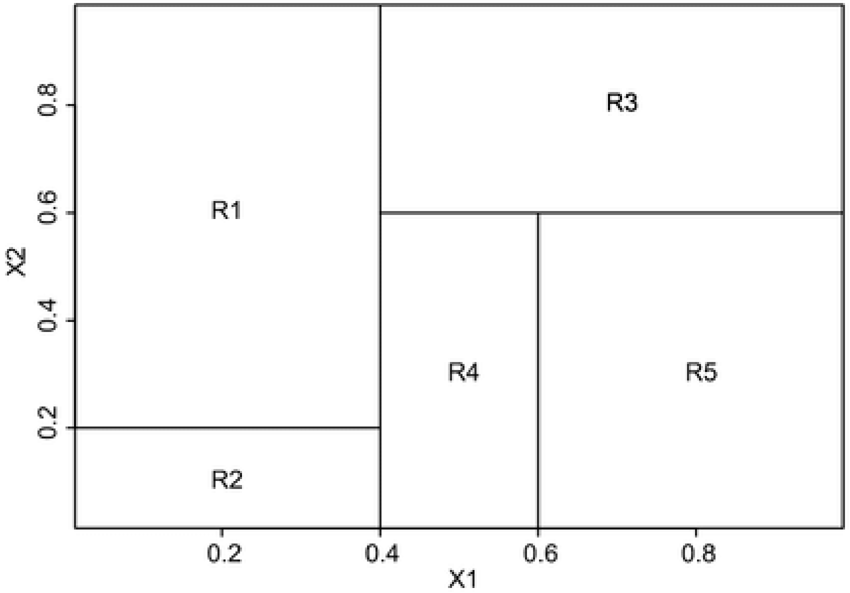


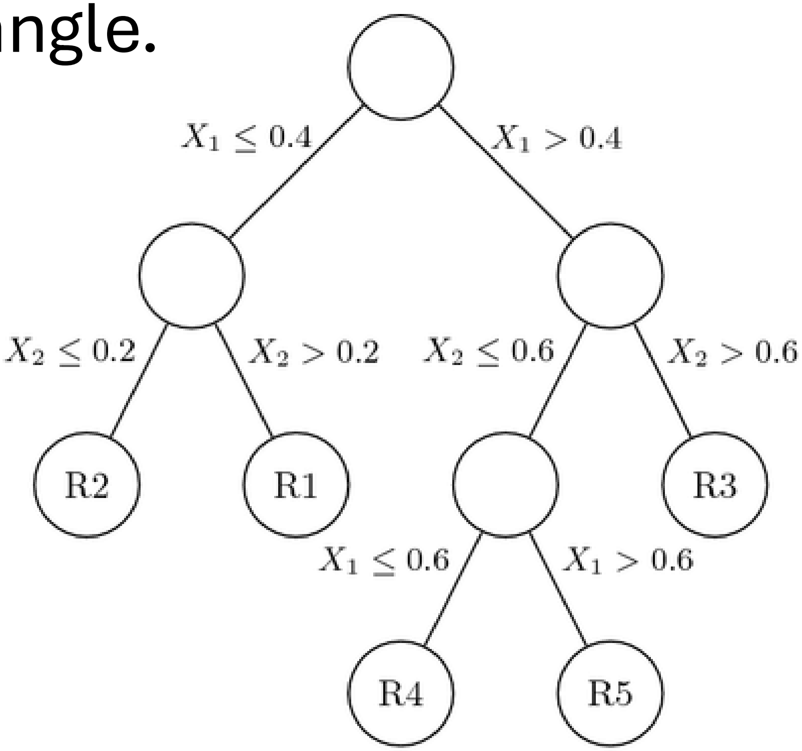
https://www.datacamp.com/tutorial/decision-tree-classification-python

Age	Weight	Smoker	Prediction
35	80	yes	High Risk
25	80	yes	?

Decision Trees

- Tree-based based methods work by **partitioning the feature space into rectangles**;
- Predictions are made by either **averaging values** or based on the **most frequently class** in each rectangle.

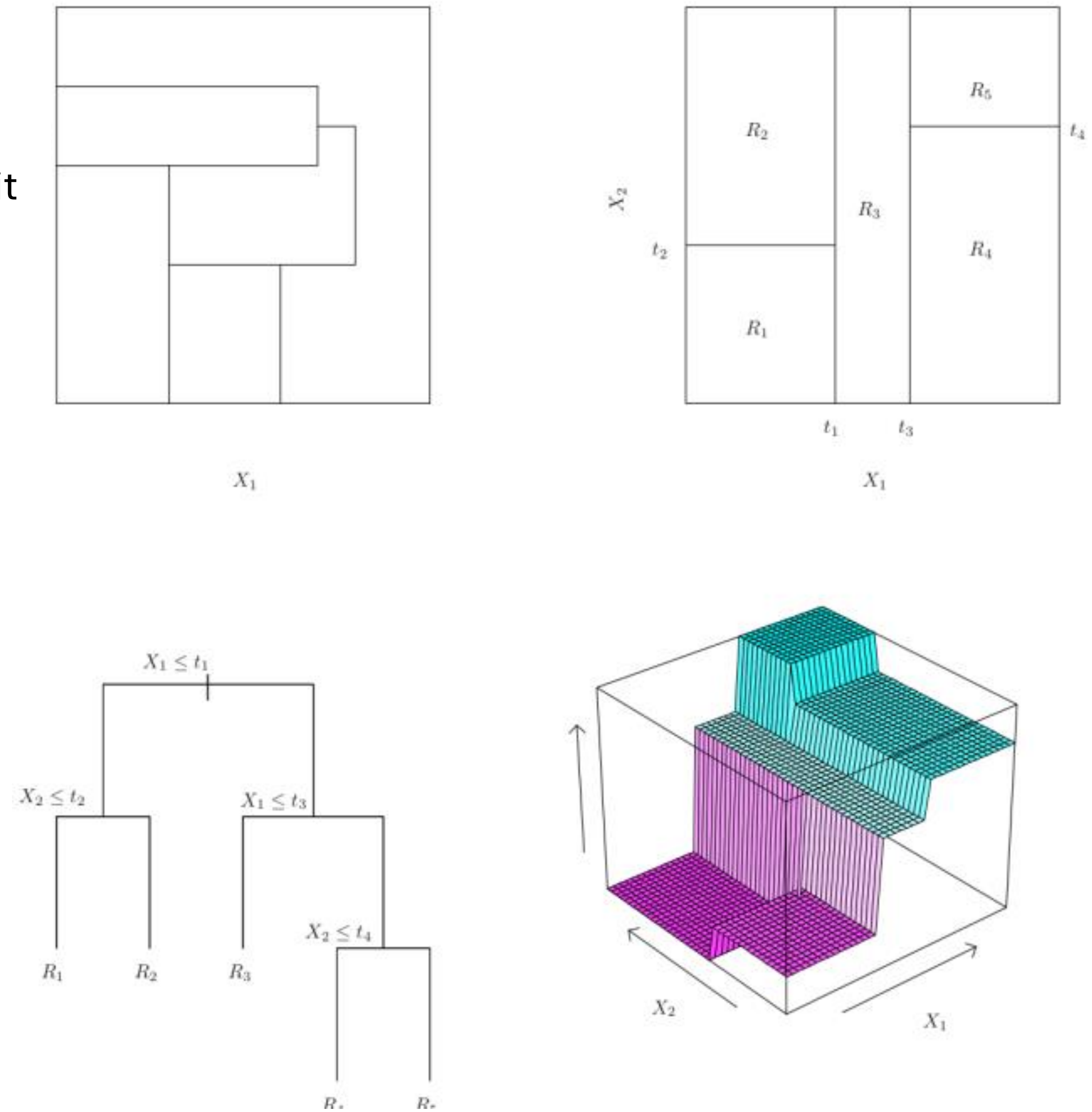




Beaulac, C., & Rosenthal, J. S. (2019). Predicting University Students' Academic Success and Major Using Random Forests. In Research in Higher Education (Vol. 60, Issue 7, pp. 1048–1064). Springer Science and Business Media LLC. https://doi.org/10.1007/s11162-019-09546-y

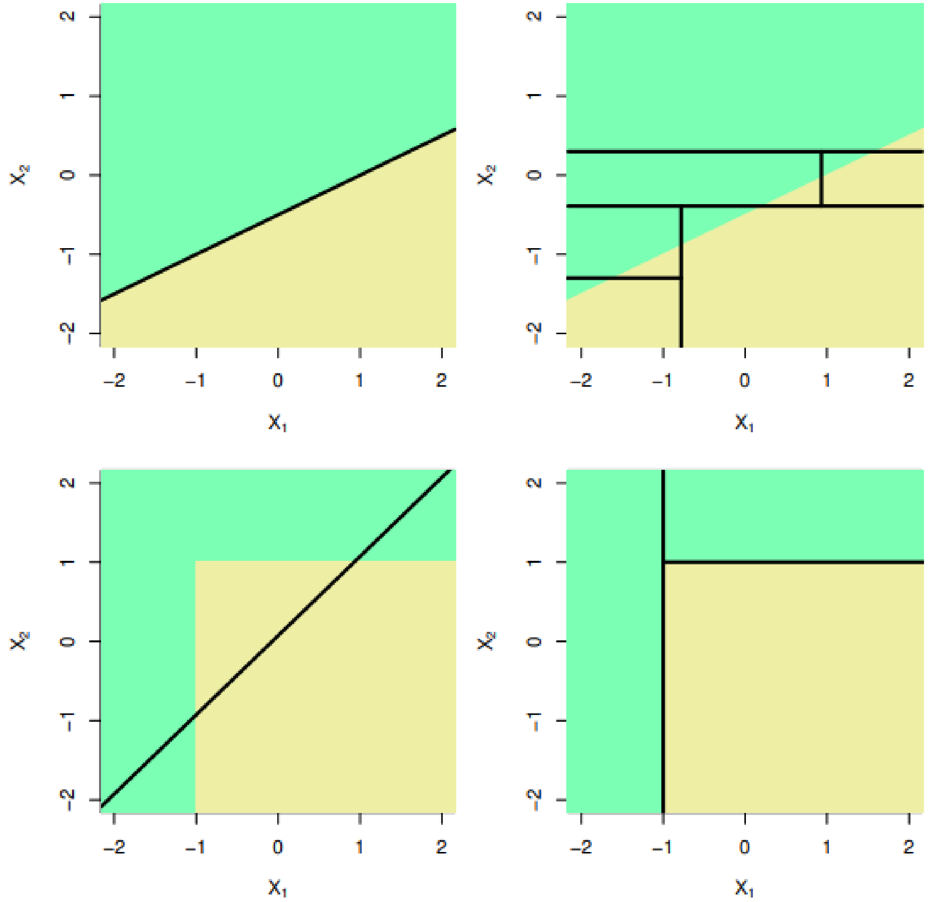
Decision Trees

We will never get a split like this one!



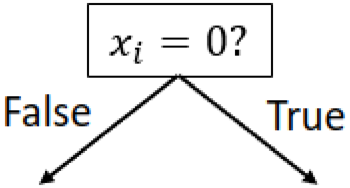
Decision Trees

- Linear models vs Decision Trees

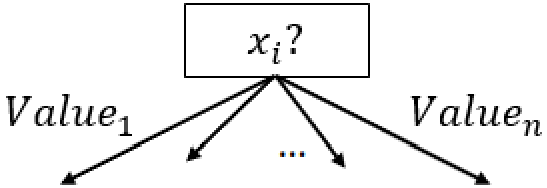


Decision Trees: Decision Nodes

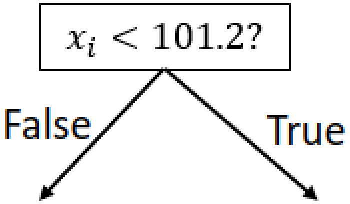
- Binary Feature



- Categorical Feature



- Numeric Feature



Decision Trees: Leaf Types

- Classification

y = 1

- Regression

y= 76.5

- Probability Estimate

P(y=0) = 0.2

P(y=1) = 0.3

P(y=2) = 0.5

Decision Trees: Algorithm

UNIVERSIDADE CATÓLICA PORTUGUESA
BRAGA

- Trees are built using a **greedy** algorithm: **Recursive binary partitioning**

Greedy means that each split is made in order to minimize a loss **without looking ahead** at future splits!

- This involves the following steps:
  - The definition of a **splititing criterion**;
  - The definition of a **stopping rule**;
  - Tree **pruning**.

Decision Trees: Splitting Criteria

- At each step, a new split is picked by **finding the featue x_j and split point s** that **best partitions the data into two half-spaces**.

$$\{\mathbf{x} : x_j < s\} \quad \{\mathbf{x} : x_j \geq s\}.$$

- For **regression** we want the split that **minimizes the residual sum os squares** (RSS)

$$RSS = \sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2,$$

where $\hat{y}_{R_j}$ is the mean values for the training data whithin the j^{th} box.

- For **classification**, we can use:
  - Entropy and Information Gain
  - Gini Index

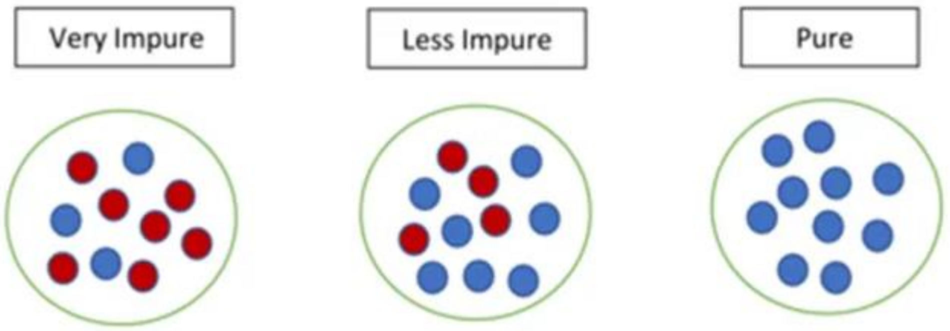
Decision Trees: Entropy and Information Gain

- "*In information theory, the entropy of a random variable is the average level of "information", "uncertainty" or "surprise", inherent in the variable's possible outcomes.*"

- In the context of Decision Trees, entropy measures the **disorder or impurity of a node.**

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

p_i is the probability of randomly picking an example of the class i.



Decision Trees: Entropy and Information Gain

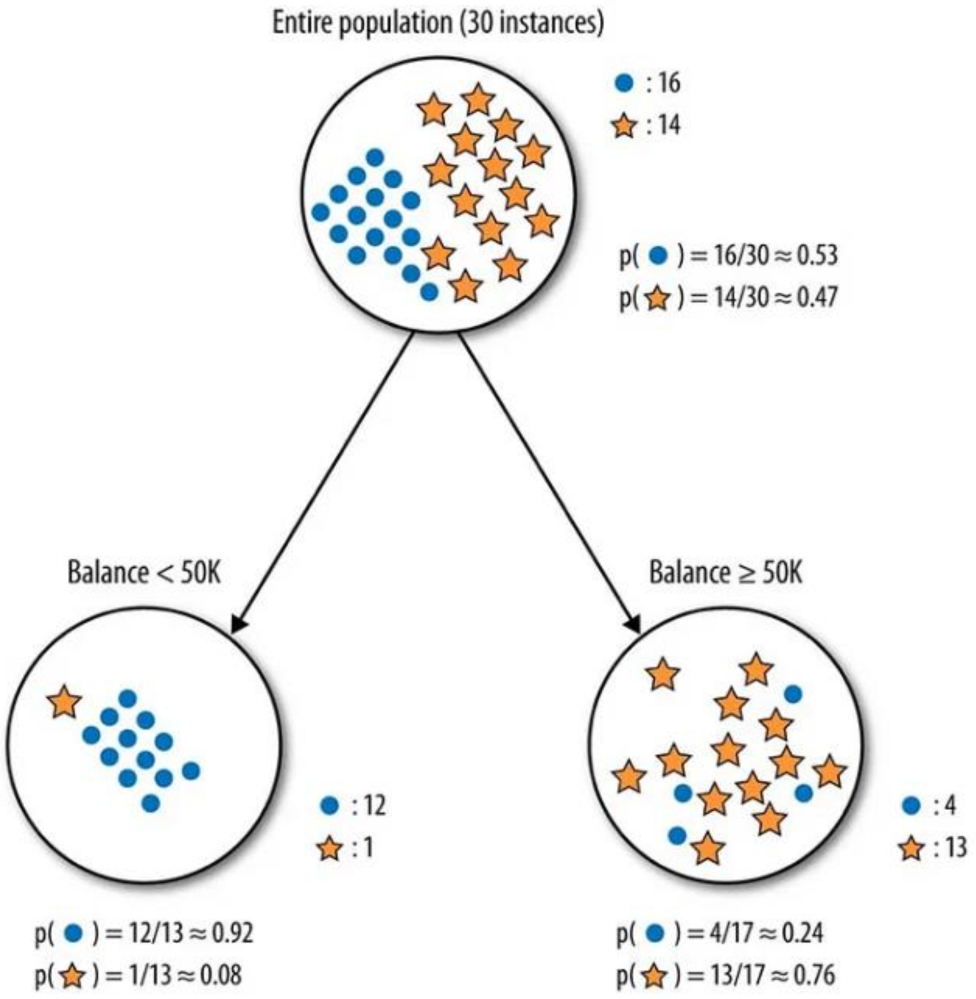
UNIVERSIDADE CATÓLICA PORTUGUESA
BRAGA

$$InformationGain = Entropy_{parent} - Entropy_{children}$$

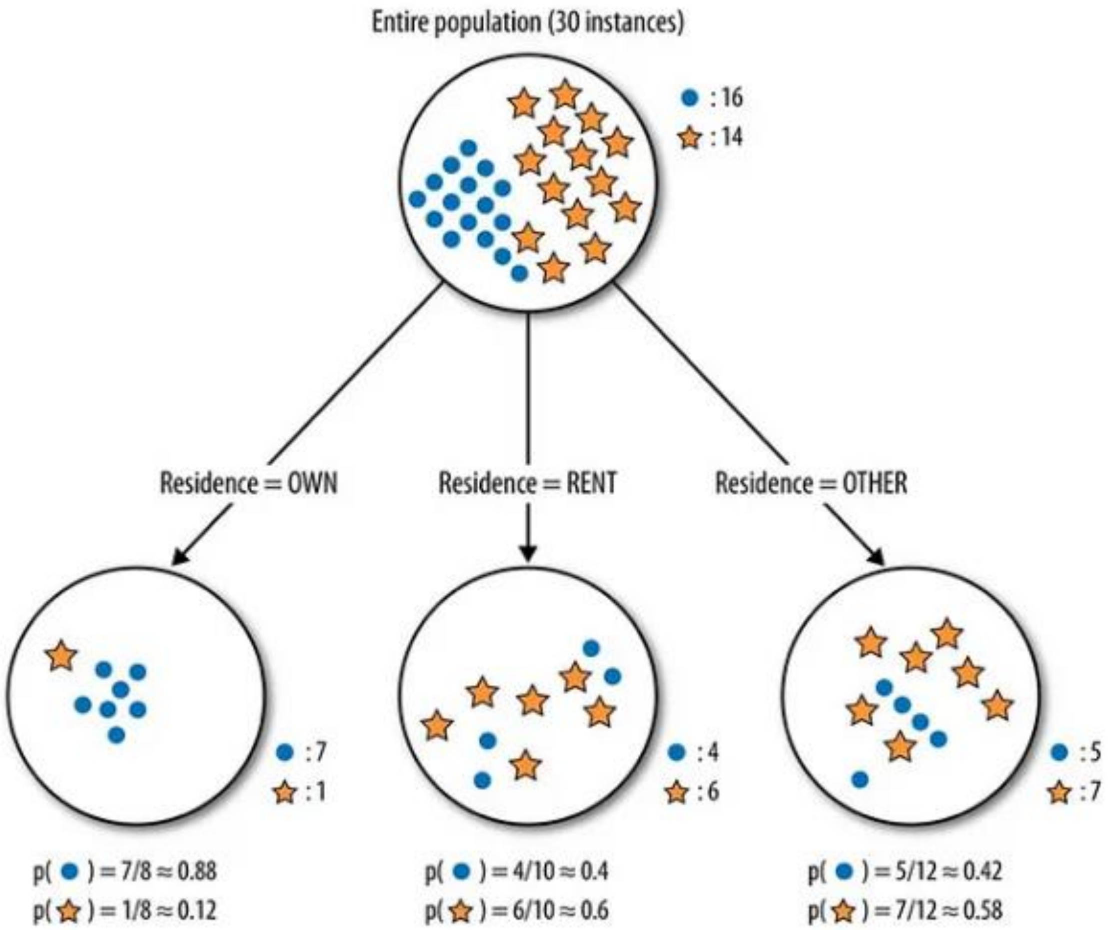
$$InformationGain = Entropy_{parent} - WeightedAvgEntropy_{children}$$

$$\text{Average Entropy} = \frac{n_{subnode_1}}{n_{parent}} E\_\text{subnode}_1 + \frac{n_{subnode_2}}{n_{parent}} E\_subnode_2 + \ldots + \frac{n_{subnode_n}}{n_{parent}} E\_subnode_n$$

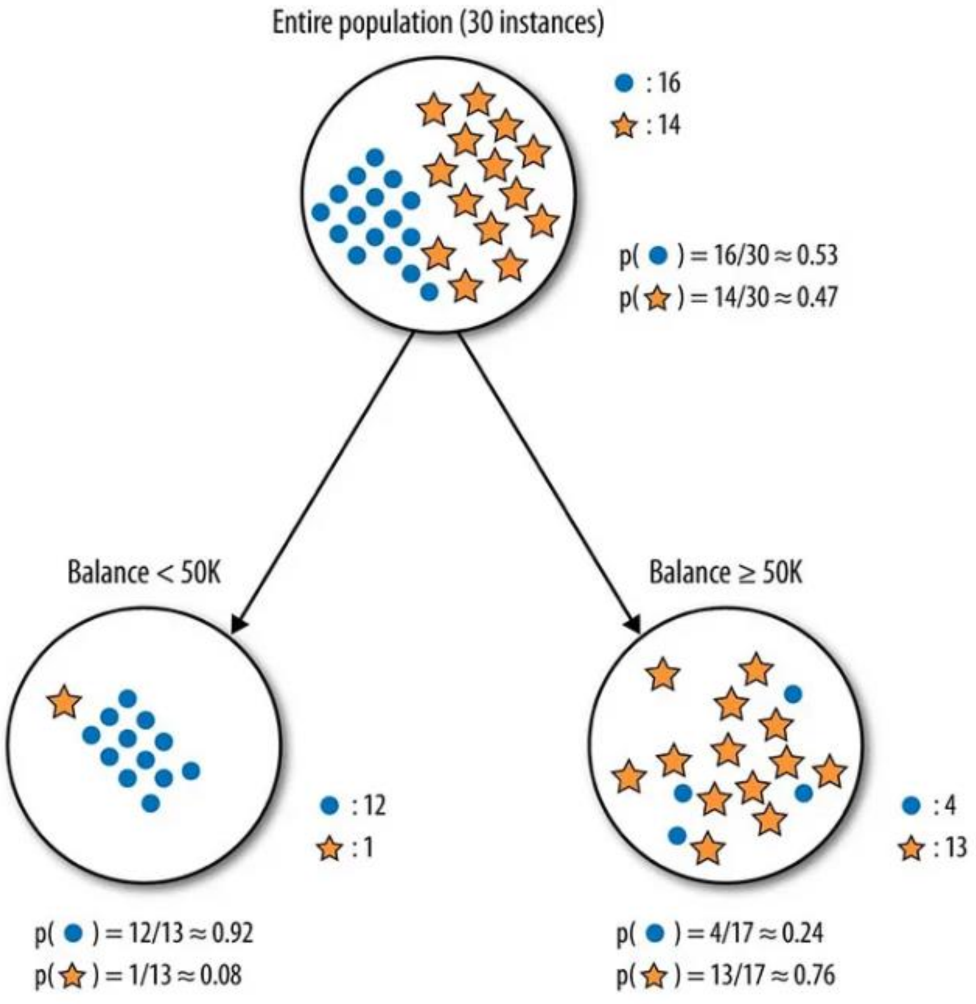
Decision Trees: Entropy and Information Gain



VS



Decision Trees: Entropy and Information Gain



$$E(Parent) = -\frac{16}{30}\log_2\left(\frac{16}{30}\right) - \frac{14}{30}\log_2\left(\frac{14}{30}\right) \approx 0.99$$

$$E(Balance < 50K) = -\frac{12}{13}\log_2\left(\frac{12}{13}\right) - \frac{1}{13}\log_2\left(\frac{1}{13}\right) \approx 0.39$$

$$E(Balance > 50K) = -\frac{4}{17}\log_2\left(\frac{4}{17}\right) - \frac{13}{17}\log_2\left(\frac{13}{17}\right) \approx 0.79$$

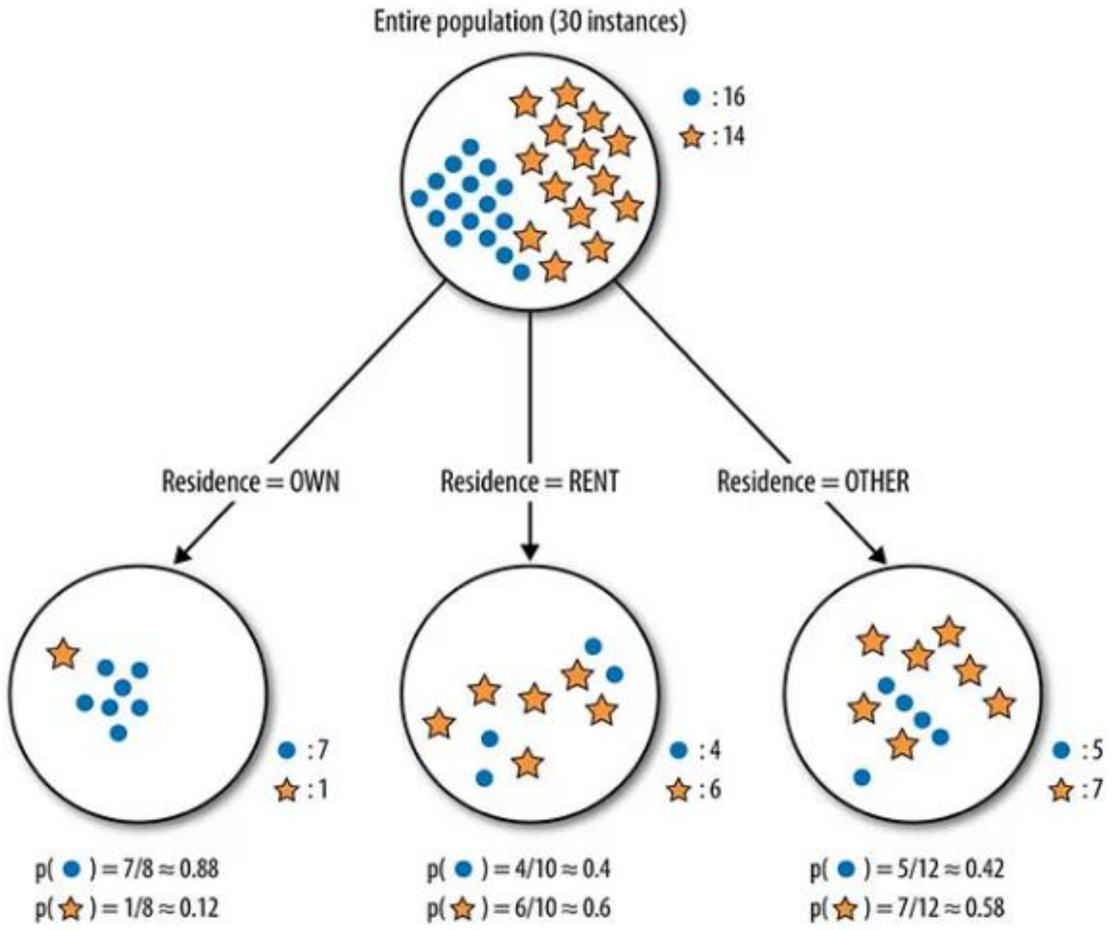
Weighted Average of entropy for each node:

$$\begin{aligned} E(Balance) &= \frac{13}{30}\times 0.39 + \frac{17}{30}\times 0.79 \\ &= 0.62 \end{aligned}$$

Information Gain:

$$\begin{aligned} IG(Parent, Balance) &= E(Parent) - E(Balance) \\ &= 0.99 - 0.62 \\ &= 0.37 \end{aligned}$$

Decision Trees: Entropy and Information Gain



$$E(\,Parent\,) \;=\; -\;\frac{16}{30}\log_2\!\left(\frac{16}{30}\right) - \frac{14}{30}\log_2\!\left(\frac{14}{30}\right) \approx 0.99$$

$$E(\,Residence\; = \;OWN\,) \;=\; -\;\frac{7}{8}\log_2\!\left(\frac{7}{8}\right) - \frac{1}{8}\log_2\!\left(\frac{1}{8}\right) \approx 0.54$$

$$E(\,Residence\; = \;RENT\,) \;=\; -\;\frac{4}{10}\log_2\!\left(\frac{4}{10}\right) - \frac{6}{10}\log_2\!\left(\frac{6}{10}\right) \approx 0.97$$

$$E(\,Residence\; = \;OTHER\,) \;=\; -\;\frac{5}{12}\log_2\!\left(\frac{5}{12}\right) - \frac{7}{12}\log_2\!\left(\frac{7}{12}\right) \approx 0.98$$

Weighted Average of entropies for each node:

$$E(\,Residence\,) \;=\; \frac{8}{30} \times 0.54 \;+\; \frac{10}{30}\times 0.97 \;+\; \frac{12}{30}\times 0.98 \;=\; 0.86$$

Information Gain:

$$\begin{aligned} IG(\,Parent,\; Residence\,) &= E(\,Parent\,) \;-\; E(\,Residence\,) \\ &= 0.99 \;-\; 0.86 \\ &= 0.13 \end{aligned}$$

Decision Trees: Gini Index

- The Gini Index measures the probability of misclassifying a randomly chosen element based on label distribution;

- Lower values indicate higher purity and better separation of classes in a decision tree node.

$$Gini = 1 - \sum_{i=1}^{j} P(i)^2$$

OR

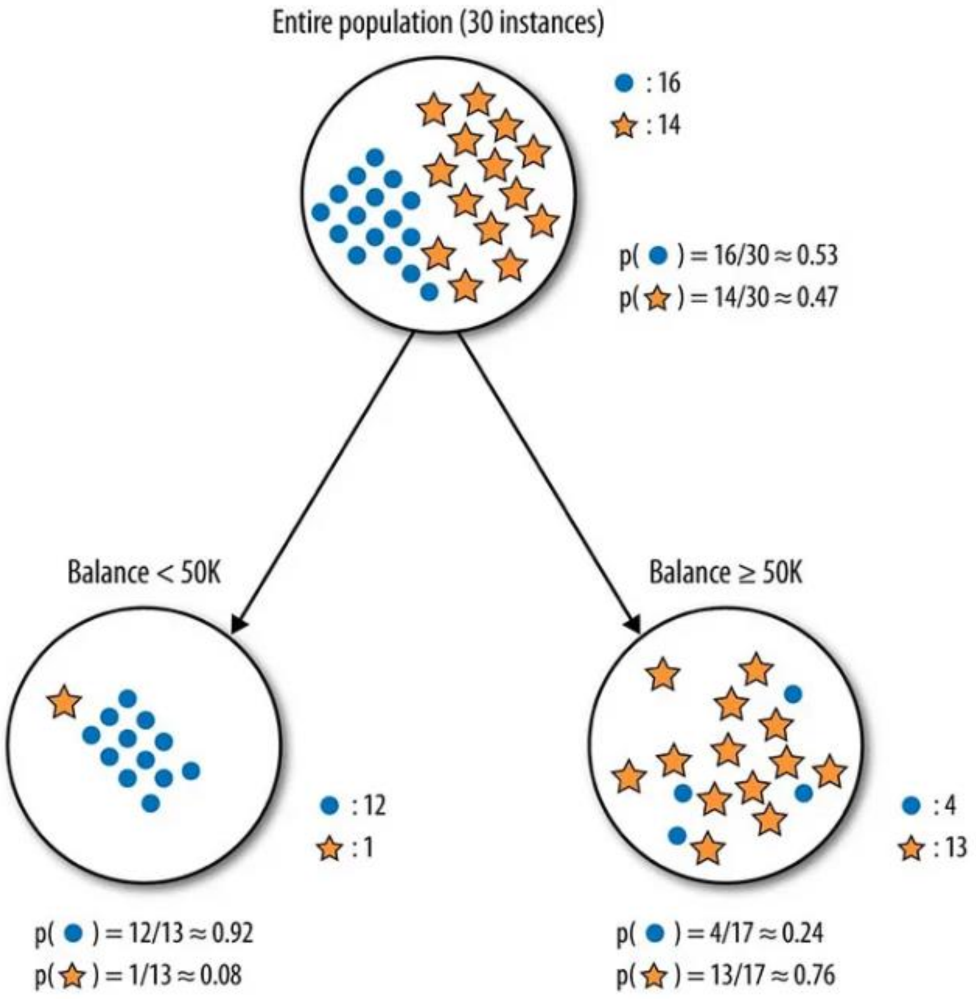
$$Gini = 1 - \sum_{i=1}^{j} P(i)(1 - P(i))$$

where j represents the number of classes in the target variable

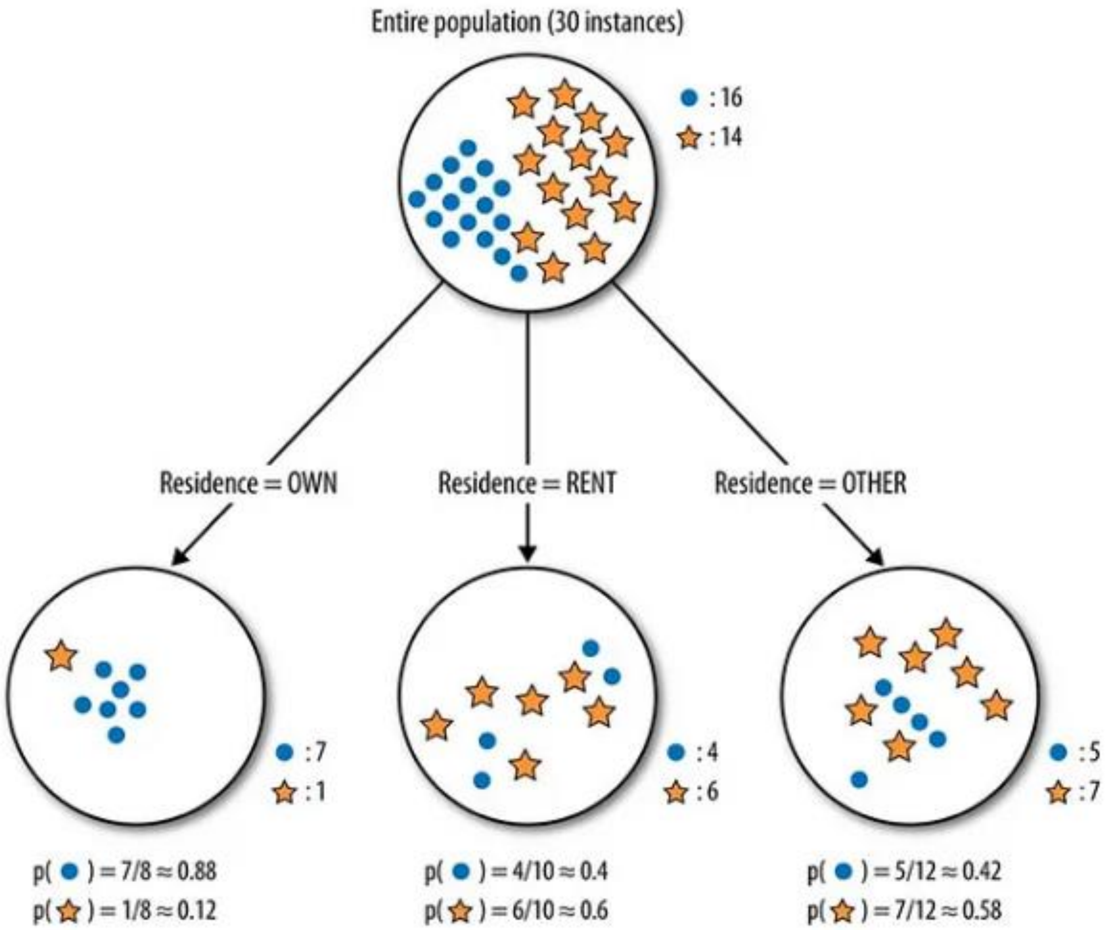
$$Gini_{split} = WeightedAvgGini_{nodes}$$

$$\text{WeightedAvgGini} = \frac{n_{subnode_1}}{n_{parent}} Gini_{subnode_1} + \frac{n_{subnode_2}}{n_{parent}} Gini_{subnode_2} + \ldots + \frac{n_{subnode_n}}{n_{parent}} Gini_{subnode_n}$$

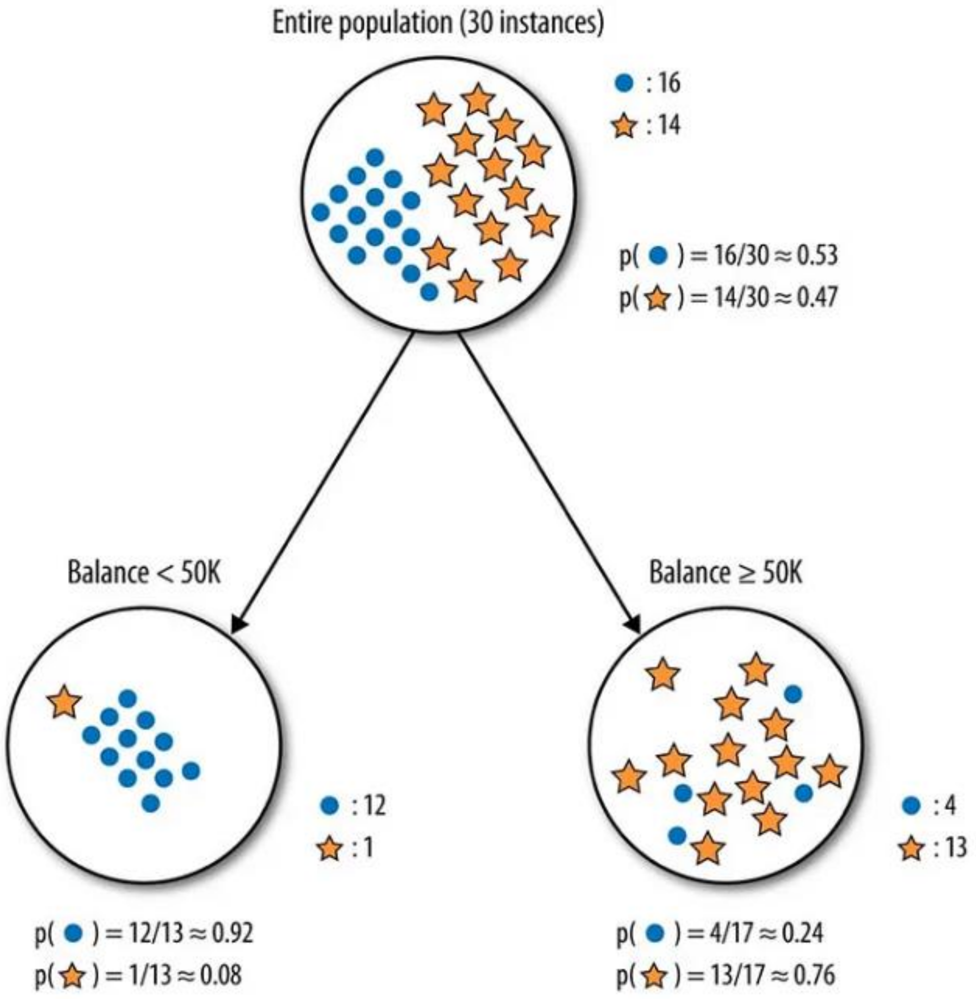
Decision Trees: Gini Index



VS



Decision Trees: Gini Index

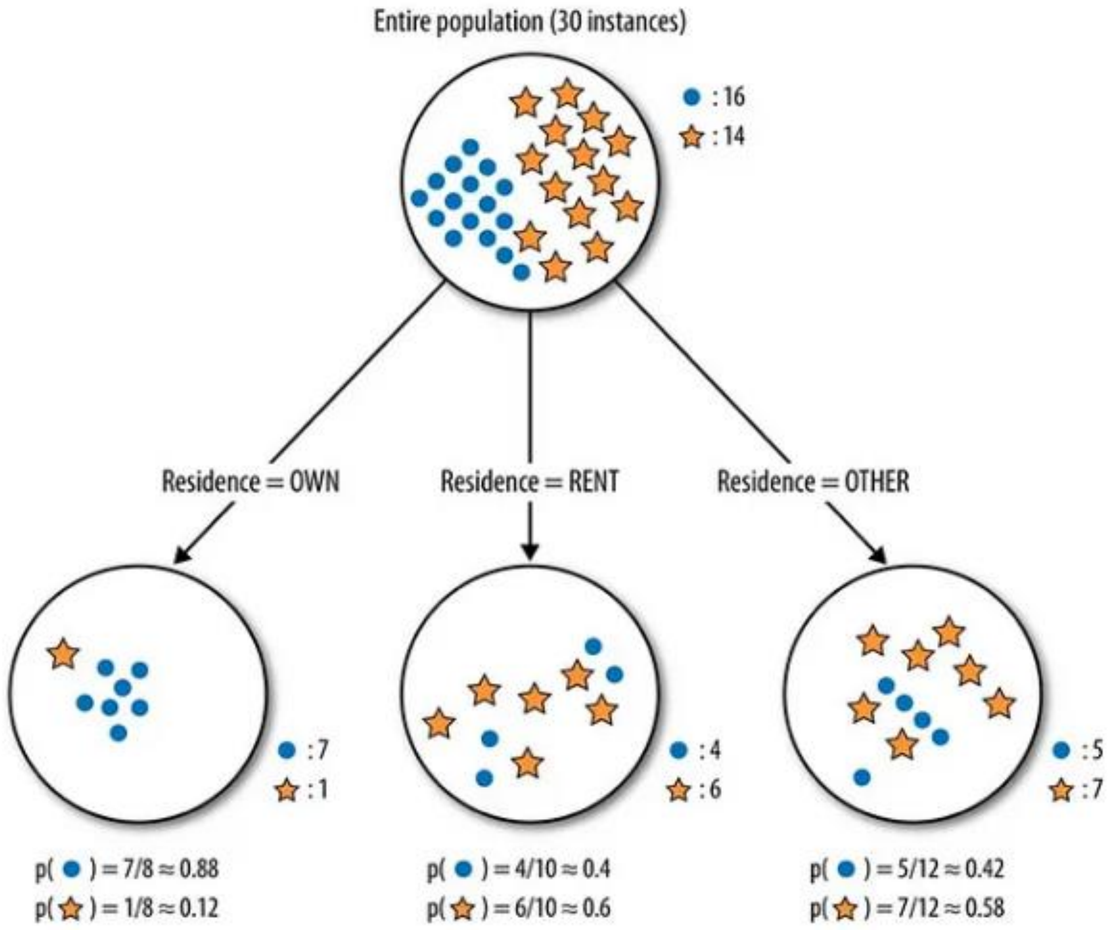


$$Gini_{(Balance<50)} = 1 - (\frac{12}{13})^2 - (\frac{1}{13})^2 = 0.142$$

$$Gini_{(Balance\geq50)} = 1 - (\frac{4}{17})^2 - (\frac{13}{17})^2 = 0.360$$

$$Gini = \frac{13}{30} * 0.142 + \frac{17}{30} * 0.360 = 0.266$$

Decision Trees: Gini Index



$$Gini_{(OWN)} = 1 - (\frac{7}{8})^2 - (\frac{1}{8})^2 = 0.219$$

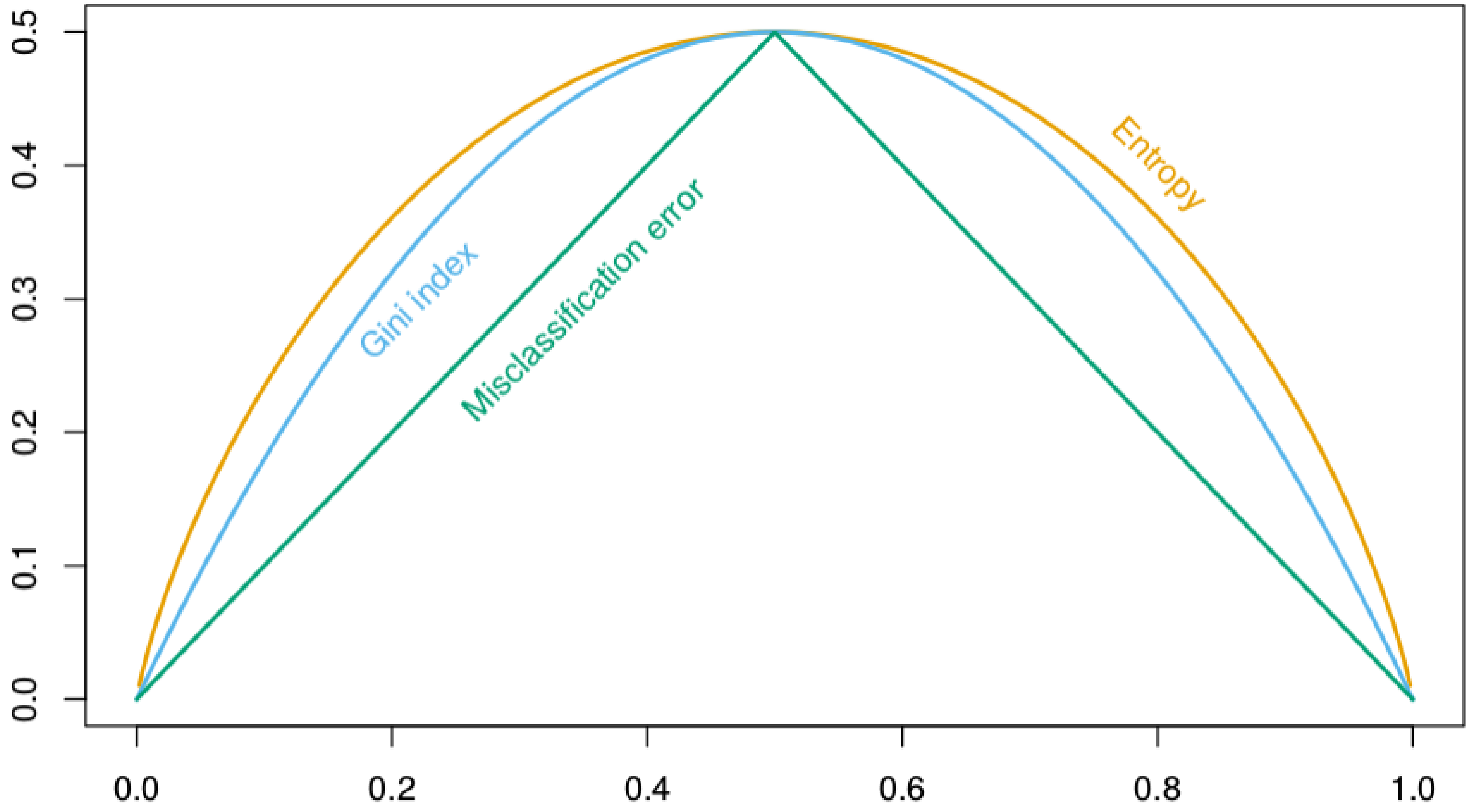
$$Gini_{(RENT)} = 1 - (\frac{4}{10})^2 - (\frac{6}{10})^2 = 0.48$$

$$Gini_{(OTHER)} = 1 - (\frac{5}{12})^2 - (\frac{7}{12})^2 = 0.486$$

$$Gini = \frac{8}{30} * 0.219 + \frac{10}{30} * 0.48 + \frac{12}{30} * 0.486 = 0.4128$$

Decision Trees: Splitting Criteria

- Why not minimize the missclassification error?



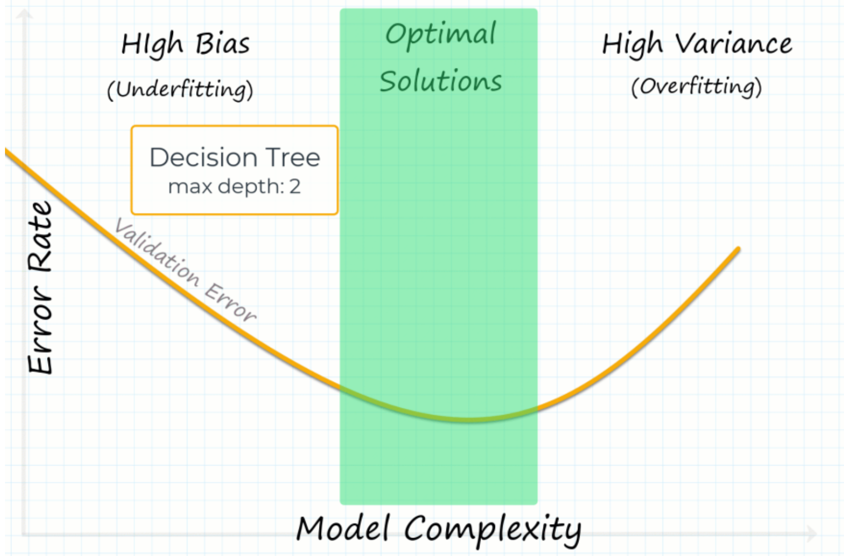
Decision Trees: Stopping Rules

UNIVERSIDADE CATÓLICA PORTUGUESA
BRAGA

- **Maximum depth:** limits the depth of the tree;
- **Minimum samples per leaf:** limits the minimum number of samples a leaf node can have;
- **Minimum samples per split:** limits the minimum number of samples required to perform a split;
- **Maximum number of leaf nodes:** caps the total number of lead nodes in a tree;
- **Impurity threshold:** a split is only performed if it reduces impurity by a certain amount;

Decision Trees

- The flexibility/complexity of decision trees is mainly decided by the **tree depth**:
  - To obtain a **small bias** we need a deep tree!
  - However, this results in **high variance**!



- To improve the performance:
  - **Pruning:** grow deep trees (small bias, high variance) which then are pruned into smaller ones (reduce variance);
  - **Ensemble methods (next session):** combine multiple simple trees.
    - Bagging and Random Forests
    - Boosted trees

Decision Trees: Tree Pruning

- **Deep trees often overfit** the training data resulting in **poor test performance**;

- We could stop spliting as soon the information gain does not improve at least a pre-specified amount;

- However, **"weak" splits early can sometimes lead to a really good split later**;

- **Solution:** Grow a deep tree and then **prune it back**.

Decision Trees: Cost Complexity Pruning

- **Cost complexity pruning** aka **weakest link pruning**:

- Mathematically, the cost complexity measure for a tree T is given by:

$$R_\alpha(T) = R(T) + \alpha|T|$$

Where:

* $R(T)$ is the risk of the tree T (overall RSS, Gini/Entropy/etc)
* $|T|$ is the number of leaf nodes in the tree T
* α is the penalty/regularization parameter

Decision Trees: Cost Complexity Pruning

$$R(T) = \sum_{t \in leaf\ \text{nodes}} (p_t \cdot M_t) = \sum_{t \in leaf\ \text{nodes}} R(t)$$

Risk of tree T

The smaler the better

Proportion of samples in leaf node t

Metric of leaf node t

Risk of leaf node t

- **Objective**: minimize $\mathrm{R}_\alpha(\mathrm{T}) = \mathrm{R}(\mathrm{T}) + \alpha|\mathrm{T}|$

Gives us cost

Gives us complexity

Cost complexity pruning

Decision Trees: Cost Complexity Pruning

UNIVERSIDADE CATÓLICA PORTUGUESA
BRAGA

- **Pruning rule:**
  - prune all child nodes of t if:

$$\underbrace{(|T_t| - 1)\alpha}_{\text{Penalty}} > \underbrace{R(t) - R(T_t)}_{\text{Reward}} \Rightarrow \alpha > \frac{R(t) - R(T_t)}{|T_t| - 1}$$

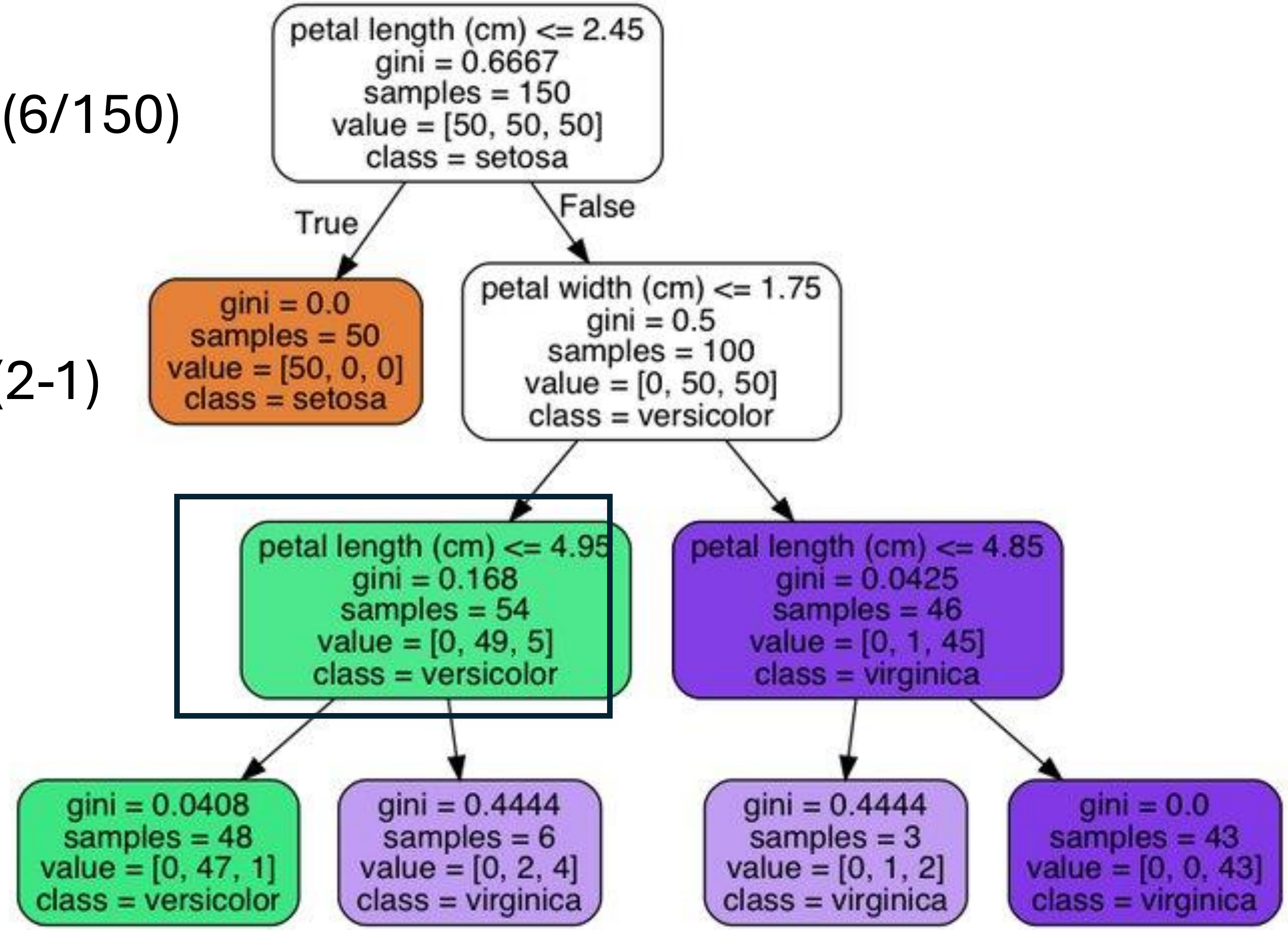
Prunning Rule

Decision Trees: Cost Complexity Pruning

- $R(t) = 0.168 * (54/150) = 0.06048$
- $R(T_t) = 0.0408 * (58/150) + 0.4444 * (6/150)$

  $= 0.033552$
- $|T| = 2$
- $\frac{R(t) - R(T_t)}{|T_t| - 1} = (0.06048 - 0.033552) / (2-1)$

  $= 0.026928$
- So, if:
  - $\alpha = 0.02$ we don't prune
  - $\alpha = 0.03$ we do prune

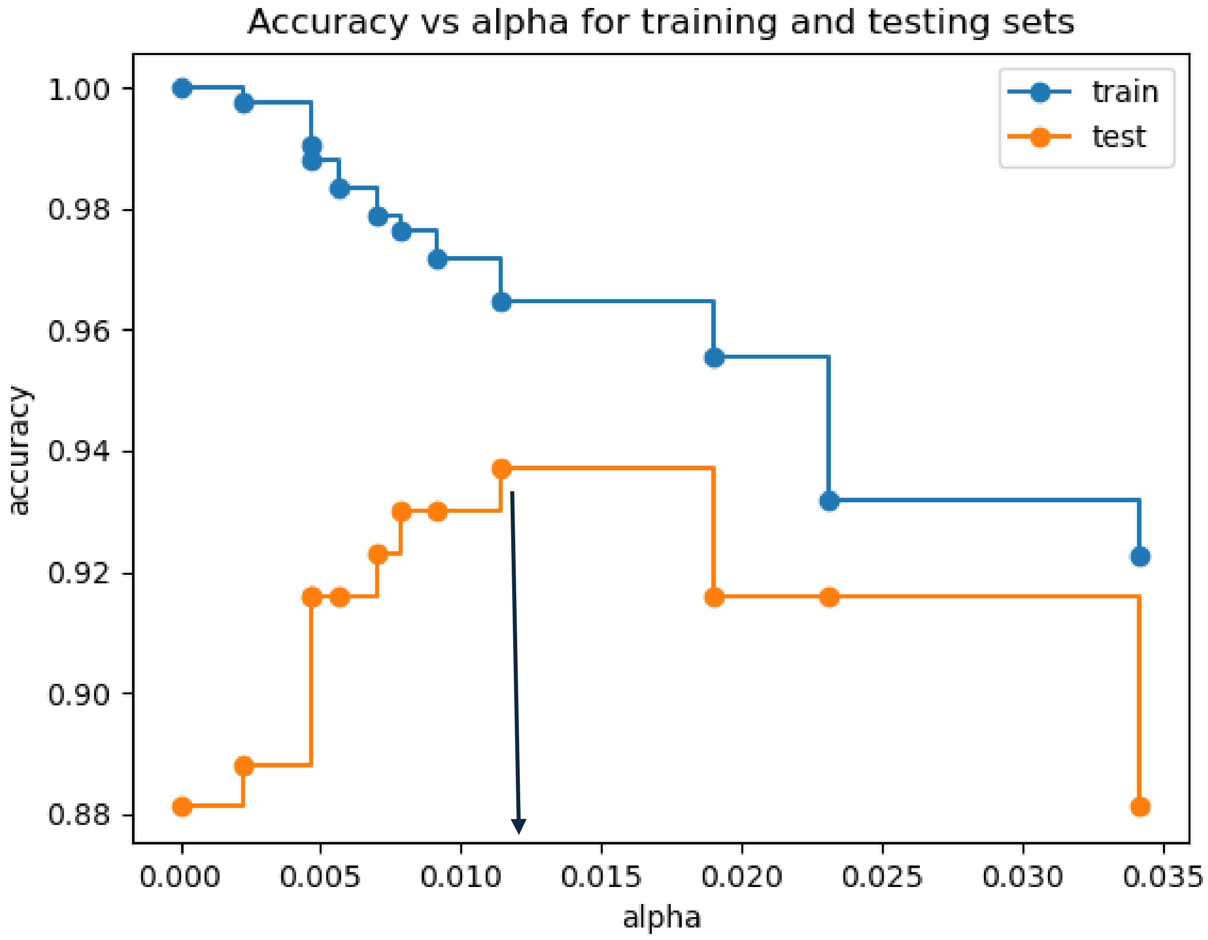


- **Question:**
  - **How to choose the value of α ?**

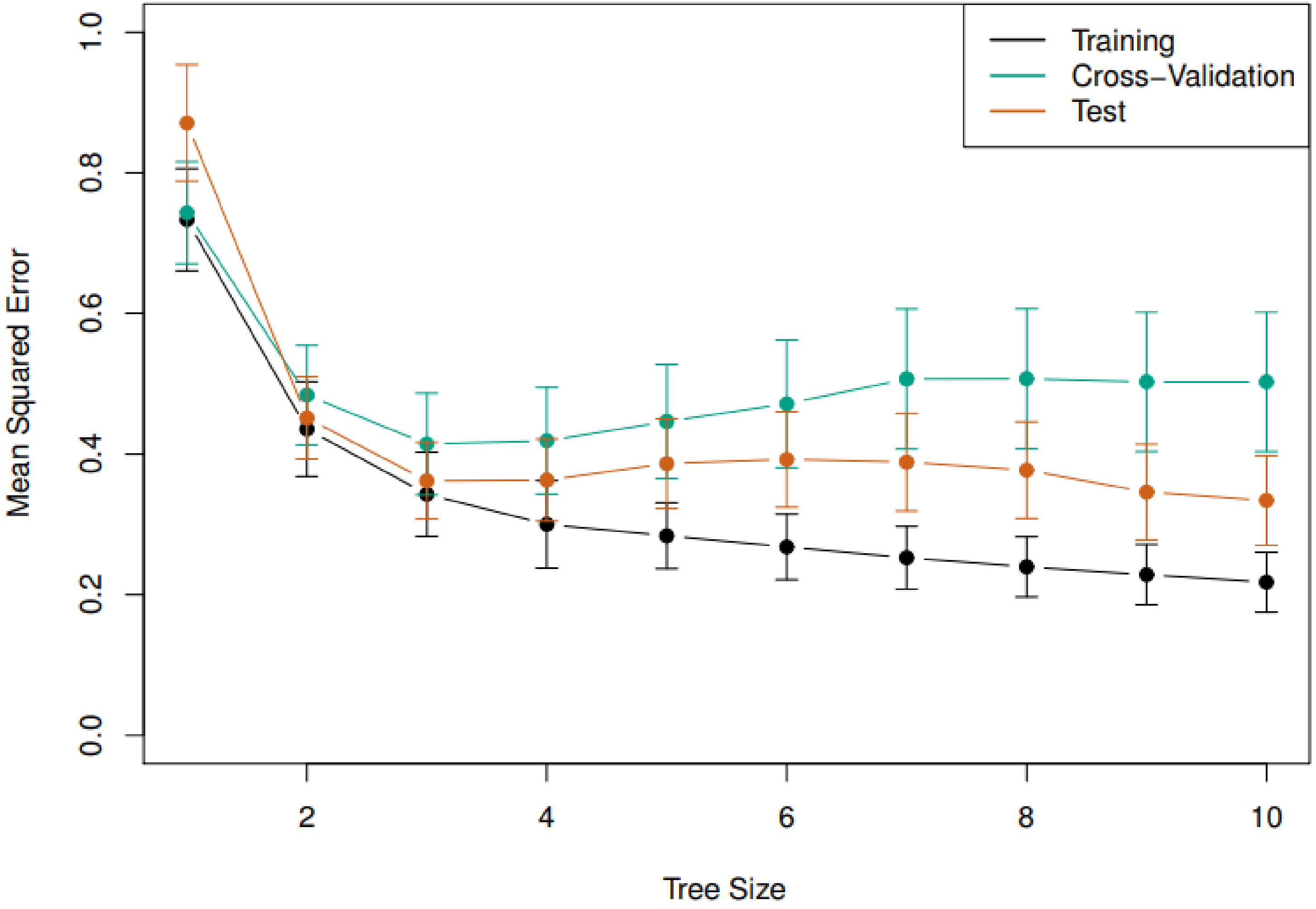
Decision Trees: Cost Complexity Pruning

- **Question:**
  - **How to choose the value of α?**

- Using cross-validation!



Decision Trees: Depth vs Error



- Looks like a small 3-leaf tree has the lowest CV error!

Decision Trees: Advantages

UNIVERSIDADE CATÓLICA PORTUGUESA
BRAGA

- **Interpretability:** easy to understand and interpret, making them suitable for explaining the reasoning behind decisions to non-experts.

- **No Data Preprocessing:** can handle both numerical and categorical data without requiring extensive preprocessing such as normalization or scaling.

- **Handles Non-linear Relationships:** can capture non-linear relationships between features and the target variable without explicitly modeling them.

- **Handles Missing Values:** can handle missing values by simply excluding them from the splitting process, making them robust to missing data.

- **Feature Importance:** provide a measure of feature importance, which can help identify the most influential features in the dataset.

- **Efficiency:** have a relatively fast training time, especially for smaller datasets, compared to more complex algorithms.

Decision Trees: Limitations

- **Overfitting:** are prone to overfitting, especially when they grow too deep or are not pruned properly, capturing noise or specific patterns in the training data that do not generalize well.

- **Instability:** small variations in the data can lead to different tree structures, making decision trees unstable and sensitive to changes in the training data.

- **Bias Toward Dominant Classes:** in classification tasks with imbalanced classes, decision trees may exhibit a bias toward the dominant classes, leading to poor performance on minority classes.

- **Greedy Nature:** use a greedy, top-down approach to recursively partition the feature space, which may not always lead to the globally optimal tree structure.

Resources

- Koning, M., & Smith, C. (2017). Decision trees and random forests. Independently Published.
- https://www.youtube.com/watch?v=_L39rN6gz7Y
- https://www.youtube.com/watch?v=_L39rN6gz7Y
- https://www.youtube.com/watch?v=wpNl-JwwplA
- https://www.youtube.com/watch?v=D0efHEJsfHo