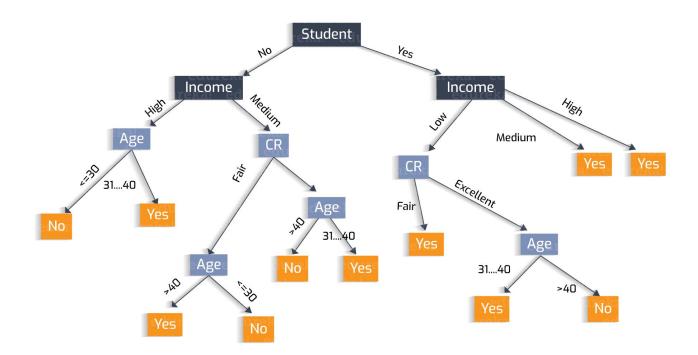
#### Tree Based Methods Fall 2024

#### **Contents**

- Tree Based Methods
  - o ID3, C4.5, CART
  - Mixed of numeric and categorical
  - Missing data
- Pruning
- Visualization
- Rule Generator



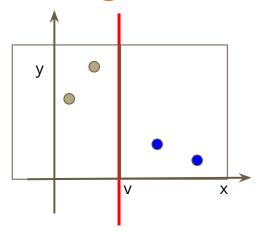
## **Explainable Rules**



# **Decision Tree Example**

	Pre	edictors		Target				Dec	ision Tree
Outlook	Temp.	Humidity	Windy	Play Golf			Outlook		
Rainy	Hot	High	Falce	No					
Rainy	Hot	High	True	No				_	
Overoast	Hot	High	Falce	Yes	Sur	nny		Do	iny
Sunny	Mild	High	Falce	Yes	30	mry .	Overcast	- No	mry .
Sunny	Cool	Normal	False	Yes	-				
Sunny	Cool	Normal	True	No	Fileson.		The second		
Overoast	Cool	Normal	True	Yes	Wi	ndy	Yes	Hun	idity
Rainy	Mild	High	False	No	0				
Rainy	Cool	Normal	Falce	Yes					
Sunny	Mild	Normal	False	Yes	FALSE	TRUE		High	Normal
Rainy	Mild	Normal	True	Yes					
Overoast	Mild	High	True	Yes			_		
Overoast	Hot	Normal	False	Yes	Yes	No		No	Yes
Sunny	Mild	High	True	No	100	197.000	0	140	103

#### **Handling Numeric Attributes**



- Suppose y is the prediction variable
- x is the feature
- The decision boundary is at value v
- If x > v then use samples on the right
- If x < v use samples on the left</li>
- For classification, voting
- For regression use uniform or weighted average

#### **Decision Tree**

- There are many trees possible
  - Always prefer the shortest one
- What is a good decision tree?
- For numeric attributes, it is important to decide the value to split
  - binary vs multiway splits
- For categorical variables its the set of the different values
- How to select between multiple attributes?
- How many attributes should be selected?
  - Single or multiple?



#### OCCAM'S RAZOR

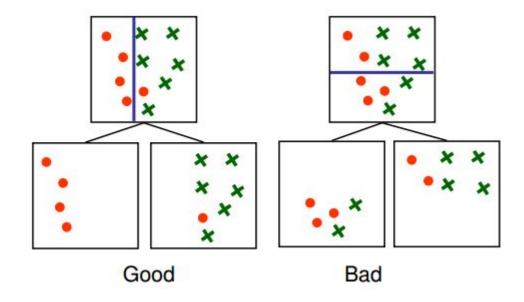
"WHEN FACED WITH TWO POSSIBLE EXPLANATIONS, THE SIMPLER OF THE TWO IS THE ONE MOST LIKELY TO BE TRUE."

#### **Decision Tree for Regression and Classification**

- Classification and Regression Trees
  - o Breiman et al 1984
  - Only Binary Splits
  - Uses Gini "measure of impurity"
- Iterative Dichotomiser 3
  - o Ross Quinlan, 1986
  - Uses Information Gain, Greedy Algorithm
- C 4.5 by
  - o Ross Quinlan, 1993
  - Improved version over ID3
    - Pruning, attributes with different costs, missing values, continuous attributes,

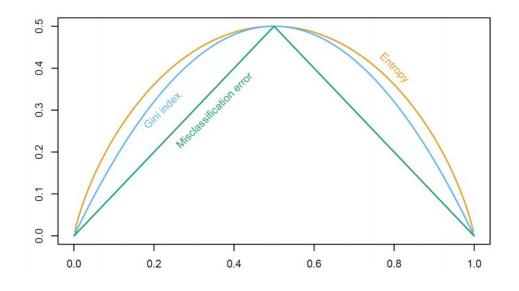
#### **Good Split vs Bad Split**

- What make a split good?
- The case for classification
  - Entropy
  - Information Gain
  - Gain Ratio
  - o Gini Index
- The case for regression
  - Squared Error



#### **Good Split vs Bad Split**

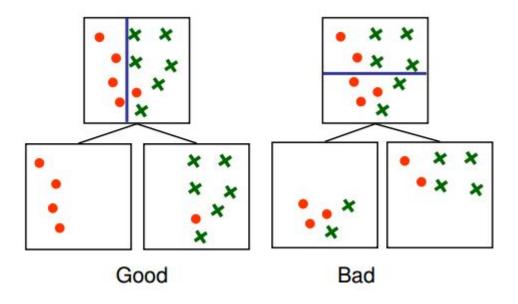
- What make a split good?
- The case for classification
  - Entropy
  - Information Gain
  - Gain Ratio
  - Gini Index
- The case for regression
  - Squared Error



## **Entropy**

- Measure of disorder in a set
- Find out entropy of each of the rectangles

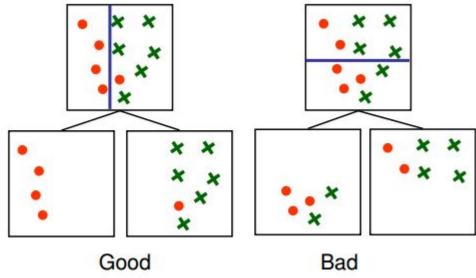
$$H(q) = -\sum_{i=1}^{\kappa} p_i log_2(p_i)$$



#### **Information Gain**

- How much information is gained by a split
- Originally a node have a measure of entropy H(q)
- After the split, the entropy is divided into sets. The gain is the difference.

$$H(q) = -\sum_{i=1}^{\kappa} p_i log_2(p_i)$$



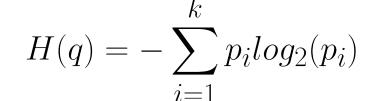
$$Gain(q, V) = H(q) - \sum_{i=1}^{|V|} \frac{N_i}{N_q} H(i)$$

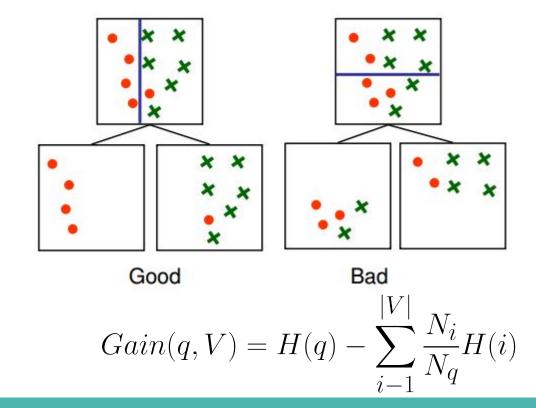
#### **Gain Ratio**

- IG biases the decision tree against considering attributes with a large number of distinct values
  - o E.g. credit card number
- Normalization of Information Gain
- Split Information

$$Split(q, V) = -\sum_{i=1}^{|V|} \frac{N_i}{N_q} log_2(\frac{N_i}{N_q})$$

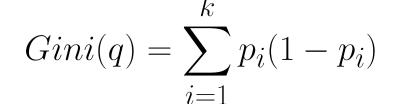
 Gain Ratio = Information Gain / Split Information

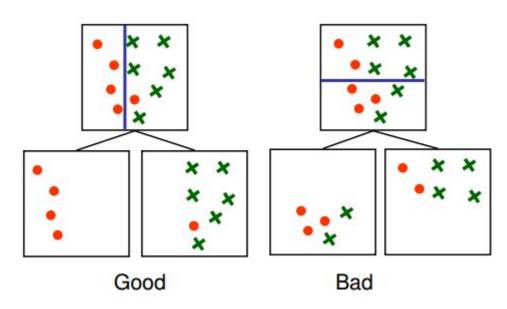




#### **Gini Index**

- Gini impurity is a measure of how often a randomly chosen element from the set would be incorrectly labeled if it was randomly labeled according to the distribution of labels in the subset
- Used by CART
- Gain is defined similarly





$$Gain(q, V) = Gini(q) - \sum_{i=1}^{|V|} \frac{N_i}{N_q} Gini(i)$$

## **An Example**

We will work on same dataset in ID3. There are 14 instances of golf playing decisions based on outlook, temperature, humidity and wind factors.

We will use gini index

Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

#### Outlook

Outlook is a nominal feature. It can be sunny, overcast or rain. I will summarize the final decisions for outlook feature.

Outlook	Yes	No	Number of instances
Sunny	2	3	5
Overcast	4	0	4
Rain	3	2	5

Gini(Outlook=Sunny) = 
$$1 - (2/5)^2 - (3/5)^2 = 1 - 0.16 - 0.36 = 0.48$$

Gini(Outlook=Overcast) = 
$$1 - (4/4)^2 - (0/4)^2 = 0$$

Gini(Outlook=Rain) = 
$$1 - (3/5)^2 - (2/5)^2 = 1 - 0.36 - 0.16 = 0.48$$

Then, we will calculate weighted sum of gini indexes for outlook feature.

Gini(Outlook) = 
$$(5/14) \times 0.48 + (4/14) \times 0 + (5/14) \times 0.48 = 0.171 + 0 + 0.171 = 0.342$$

#### **Temperature**

Temperature is a nominal feature and it could have 3 different values: Cool, Hot and Mild. Let's summarize decisions for temperature feature.

Temperature	Yes	No	Number of instances
Hot	2	2	4
Cool	3	1	4
Mild	4	2	6

Gini(Temp=Hot) = 
$$1 - (2/4)^2 - (2/4)^2 = 0.5$$

Gini(Temp=Cool) = 
$$1 - (3/4)^2 - (1/4)^2 = 1 - 0.5625 - 0.0625 = 0.375$$

Gini(Temp=Mild) = 
$$1 - (4/6)^2 - (2/6)^2 = 1 - 0.444 - 0.111 = 0.445$$

We'll calculate weighted sum of gini index for temperature feature

Gini(Temp) = 
$$(4/14) \times 0.5 + (4/14) \times 0.375 + (6/14) \times 0.445 = 0.142 + 0.107 + 0.190 = 0.439$$

## **Humidity**

Humidity is a binary class feature. It can be high or normal.

Humidity	Yes	No	Number of instances
High	3	4	7
Normal	6	1	7

Gini(Humidity=High) = 
$$1 - (3/7)^2 - (4/7)^2 = 1 - 0.183 - 0.326 = 0.489$$

Gini(Humidity=Normal) = 
$$1 - (6/7)^2 - (1/7)^2 = 1 - 0.734 - 0.02 = 0.244$$

Weighted sum for humidity feature will be calculated next

Gini(Humidity) = 
$$(7/14) \times 0.489 + (7/14) \times 0.244 = 0.367$$

#### Wind

Wind is a binary class similar to humidity. It can be weak and strong.

Wind	Yes	No	Number of instances
Weak	6	2	8
Strong	3	3	6

Gini(Wind=Weak) = 
$$1 - (6/8)^2 - (2/8)^2 = 1 - 0.5625 - 0.062 = 0.375$$

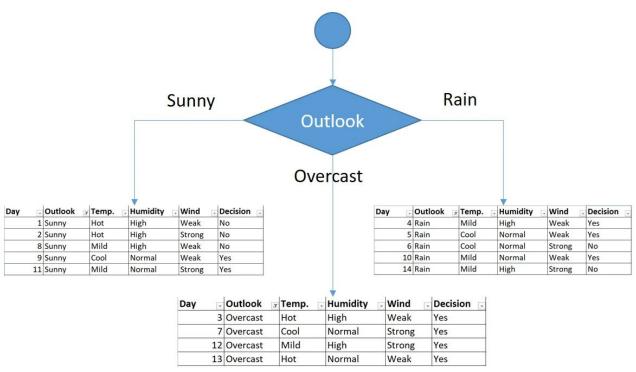
Gini(Wind=Strong) = 
$$1 - (3/6)^2 - (3/6)^2 = 1 - 0.25 - 0.25 = 0.5$$

Gini(Wind) = 
$$(8/14) \times 0.375 + (6/14) \times 0.5 = 0.428$$
  
https://sefiks.com/2018/08/27/a-step-by-step-cart-decision-tree-example/

# The first split

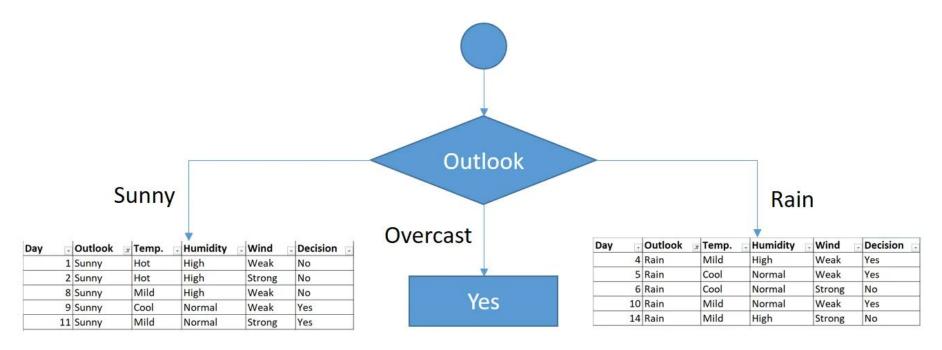
Feature	Gini index
Outlook	0.342
Temperature	0.439
Humidity	0.367
Wind	0.428

## The first split: Outlook



https://sefiks.com/2018/08/27/a-step-by-step-cart-decision-tree-example/

## The first split:Outlook



## **Recursive Partitioning**

#### A sub dataset

Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes

#### **Outlook sunny & Temperature**

Temperature	Yes	No	Number of instances
Hot	0	2	2
Cool	1	0	1
Mild	1	1	2

Gini(Outlook=Sunny and Temp.=Hot) =  $1 - (0/2)^2 - (2/2)^2 = 0$ 

Gini(Outlook=Sunny and Temp.=Cool) = 1 -  $(1/1)^2$  -  $(0/1)^2$  = 0

Gini(Outlook=Sunny and Temp.=Mild) =  $1 - (1/2)^2 - (1/2)^2 = 1 - 0.25 - 0.25 = 0.5$ 

Gini(Outlook=Sunny and Temp.) = (2/5)x0 + (1/5)x0 + (2/5)x0.5 = 0.2

https://sefiks.com/2018/08/27/a-step-by-step-cart-decision-tree-example/

## **Outlook sunny & Humidity**

Humidity	Yes	No	Number of instances
High	0	3	3
Normal	2	0	2

Gini(Outlook=Sunny and Humidity=High) = 
$$1 - (0/3)^2 - (3/3)^2 = 0$$

Gini(Outlook=Sunny and Humidity=Normal) = 
$$1 - (2/2)^2 - (0/2)^2 = 0$$

Gini(Outlook=Sunny and Humidity) = 
$$(3/5)x0 + (2/5)x0 = 0$$

## **Outlook sunny & Wind**

Wind	Yes	No	Number of instances
Weak	1	2	3
Strong	1	1	2

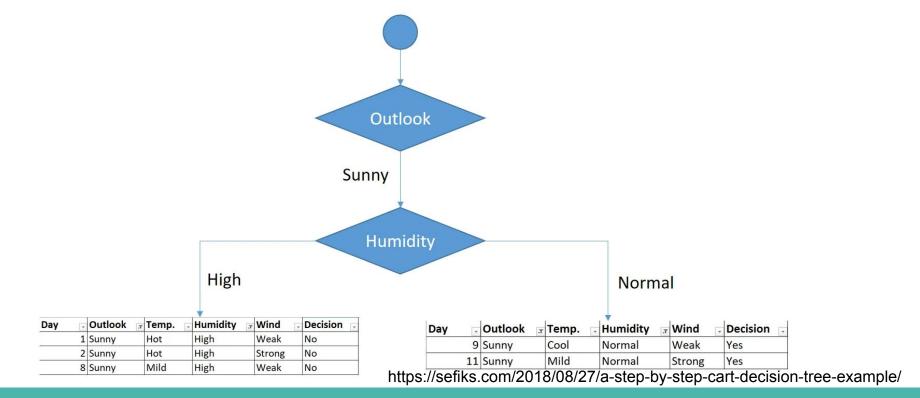
Gini(Outlook=Sunny and Wind=Weak) = 
$$1 - (1/3)^2 - (2/3)^2 = 0.266$$

Gini(Outlook=Sunny and Wind=Strong) = 1- 
$$(1/2)^2$$
 -  $(1/2)^2$  = 0.2

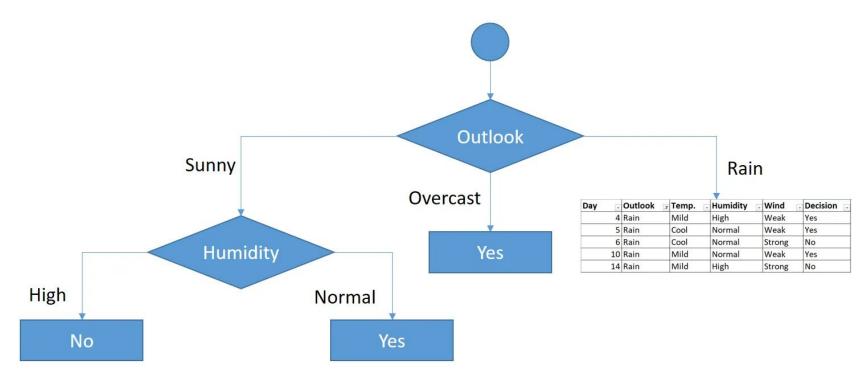
Gini(Outlook=Sunny and Wind) = 
$$(3/5)x0.266 + (2/5)x0.2 = 0.466$$

#### The second split

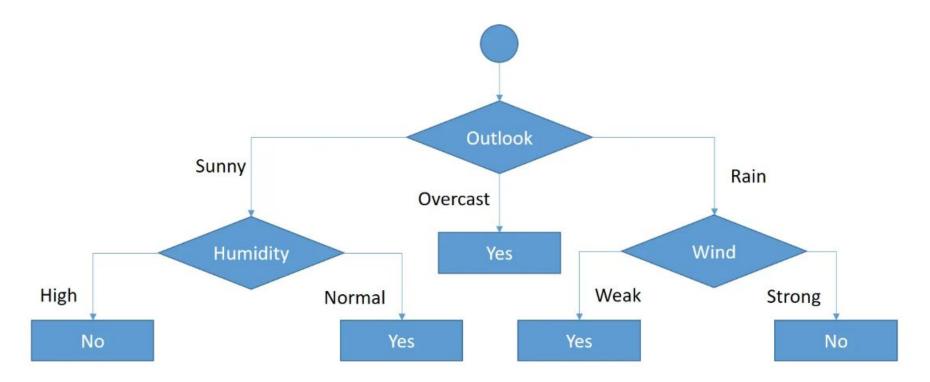
Feature	Gini index
Temperature	0.2
Humidity	0
Wind	0.466



## The second split



#### The final tree



#### **Decision Tree Overfitting**

#### Pre-Pruning

- Maximum number of leaf nodes.
- Maximum depth of the tree
- Minimum number of training instances at a leaf node

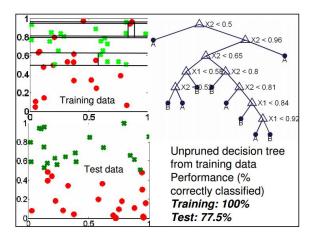
#### Post-Pruning

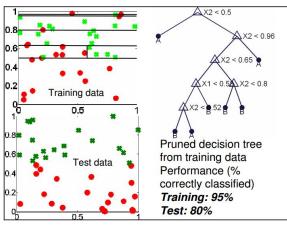
- Another strategy to avoid overfitting in decision trees is to first grow a full tree, and then prune it based on a previously held-out validation dataset.
- Use statistical Tests

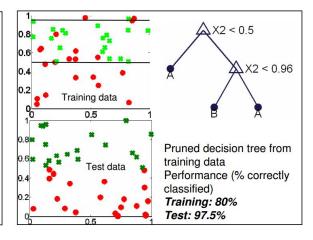


## **Tree Pruning: Validation Set**

Prune using a hold out validation dataset







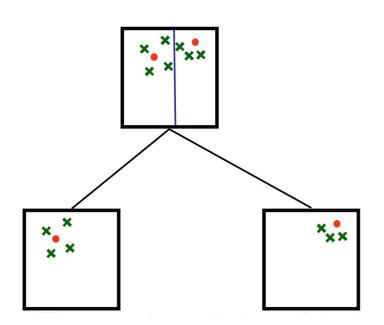
## **Detecting Useless Splits**

- Try Chi Square Test
- Check the statistic to find any significance gain achieved by the split
- Is there any difference with the arbitrary split?

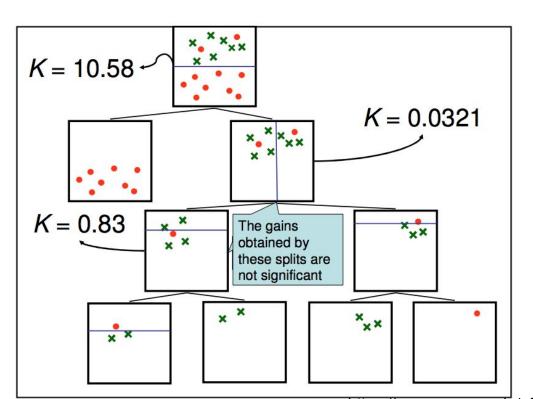
$$\Delta = \sum_{k=1}^{d} \frac{(p_k - \hat{p}_k)^2}{\hat{p}_k} + \frac{(n_k - \hat{n}_k)^2}{\hat{n}_k}$$

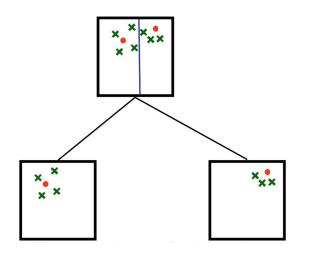
$$\hat{p}_k = p \times \frac{p_k + n_k}{p + n}$$

$$\hat{n}_k = n \times \frac{p_k + n_k}{p + n}$$



## **Detecting Useless Splits**





$$\hat{p}_k = p \times \frac{p_k + n_k}{p + n}$$

$$\hat{n}_k = n \times \frac{p_k + n_k}{p + n}$$

$$\Delta = \sum_{k=1}^{d} \frac{(p_k - \hat{p}_k)^2}{\hat{p}_k} + \frac{(n_k - \hat{n}_k)^2}{\hat{n}_k}$$

https://www.cs.cmu.edu/afs/cs.cmu.edu/academic/class/15381-s06/www/DTs2.pdf

#### **Decision Tree**

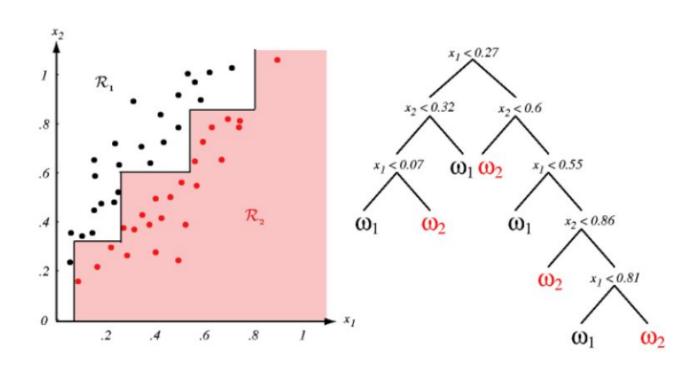
#### Pros

- Interpretable and Simple
- Handles all types of data
- Handles missing values
- Less pre-processing required
- Fast computation
- non-parametric

#### Cons

- NP-complete
- Not stable
- Often Overfits
- High bias
- Not suitable for unstructured data

## **Multi-Variable Split?**



# Multi-Variable Split?

