Decision Trees

Análise de Dados em Informática
Licenciatura em Engenharia Informática
ISEP/IPP

Ana Maria Madureira
2022/2023

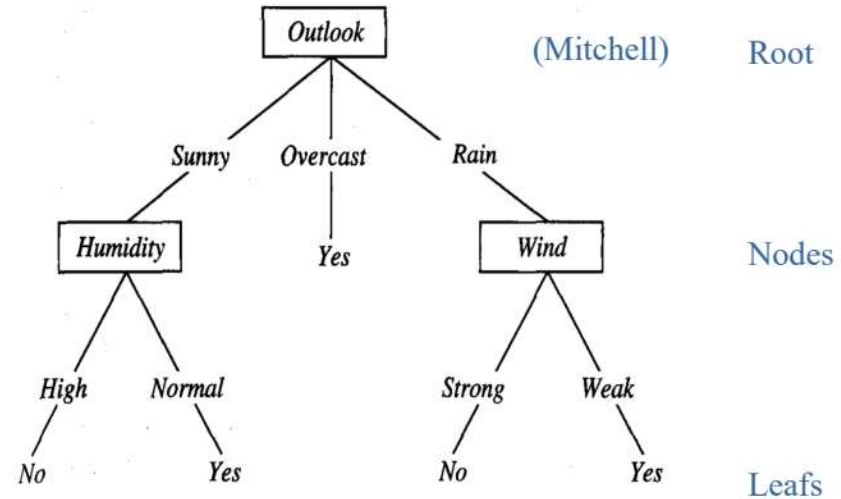
Fontes:

- Tom Mitchell, Machine Learning. McGraw-Hill, 1997.
- Catarina Silva e Bernardete Ribeiro, Aprendizagem Computacional em Engenharia, Imprensa da Universidade de Coimbra, 2018
- Sebastian Raschka, STAT479 FS18. L01: Intro to Machine Learning, Fall 2018

Decision Trees - Representation of concepts

- A decision tree (DT) consists of a set of decision nodes, connected by branches, extending downward from the root node until terminating in leaf nodes. Beginning at the root node, which by convention is placed at the top of the decision tree diagram, variables are tested at the decision nodes, with each possible outcome resulting in a branch. Each branch then leads either to another decision node or to a terminating leaf node.
- Learning in DT is a method of approximating discrete-value target function represented in a decision tree.
- The learned trees may also be represented by a set of if-then rules to improve human readability and interpretability.
- These learning methods are among the most used and have been applied to a large number of fields (since medial diagnosis to credit risk management in banking

Concept: "good day to play tennis"

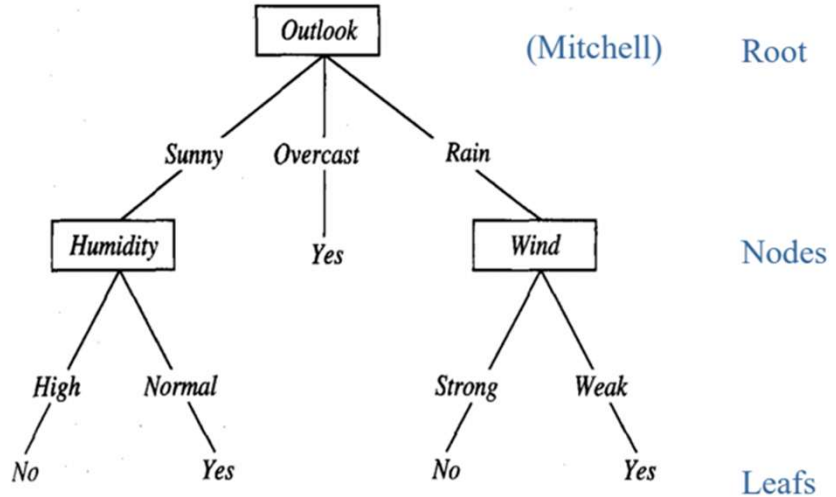


Binary decision function: Yes / No

Decision Trees

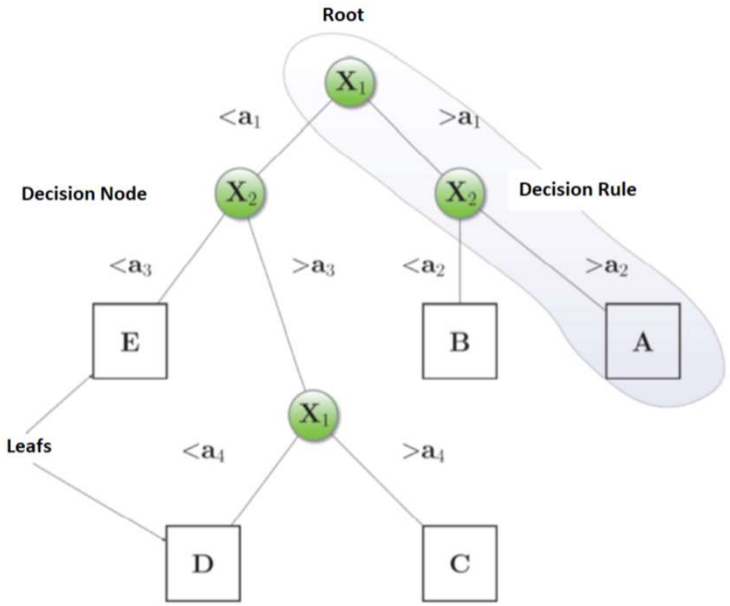
- The Decision Tree classifies instances of the problem by descending the tree from the root to some leaf that gives the classification of the instance.
- Each node executes a test over some attribute of the instance, and each descending branch from that node corresponds to one of the possible val ues of that attribute.
- The classification process of an instance starts at the root, testing the attr ibute specified by this node, descending the branch of the tree correspo nding to the value of the attribute in the example, and going to the extre me node of this branch, where the test of its attribute is made, and the process continues until a leaf of the tree is reached.

Concept: "good day to play tennis"



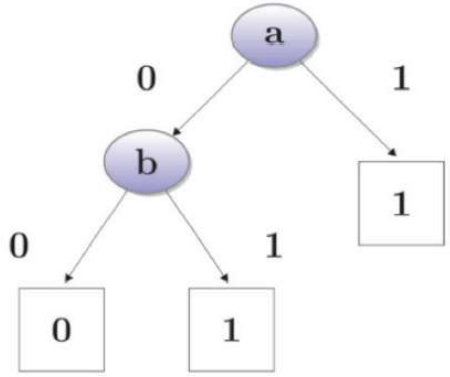
Binary decision function: Yes / No

Decision Trees

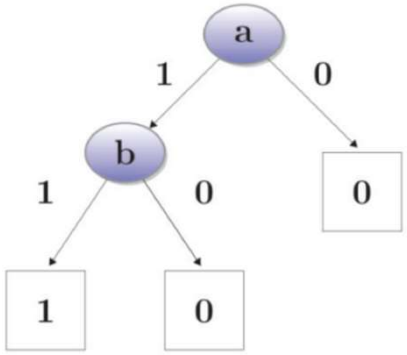


- For the DT representation, we use decision numbers that contain a test related to a certain attribute. Each descending branch corresponds to a possible value (or range of values) for that attribute. Each leaf is associated with a class, and each path in the tree (from the root to the leaf) corresponds to a classification rule.
- In the space defined by the attributes, each leaf corresponds to a region, being represented by a rectangle.
- The intersection of the rectangles is empty, and the union is the complete space
- A DT represents the disjunction of the set of restrictions in the values of the attributes
- Each path in the branch in the tree is a set of conditions, while the set of branches in the tree is disjoint.
- Any logical function can be represented by a decision tree. The result, is found in the right terminal rectangle.

Decision Trees



Representation of the logical function OR with a DT



Representation of the logical function AND with a DT

Decision Tree construction algorithm

Input: training examples
1. For each attribute 1.1. Calculate the information gain 1.2. Define the attribute with the highest information gain 2. Create a decision node that is divided into branches based on the values of this attribute 3. If the examples are not all separated by class in each branch, re-apply step 1 and step 2 to each node that is still to be divided
Output: Decision tree

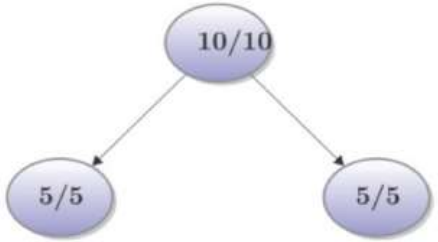
Decision Tree construction algorithm

- The process of building a decision tree initially goes through the
choosing an attribute that will be the root of the tree. In the example of playing golf of the figure, the chosen one was the weather. The choice of the attribute for the root of the tree is very important.

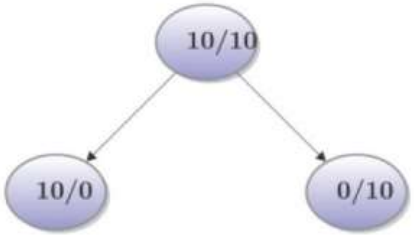
- It is usually based on information gain. Once chosen this attribute, the "tree" is extended by adding a branch for each value (or range of values) of the attribute, following the examples for the leaves (taking into account the attribute value).
- Then, it is evaluated if the tree is completely built, that is, if all the examples of each branch are of the same class. If not, the process is re-applied to all new nodes in the tree.

Decision Tree
Criteria for Choosing an Attribute

- How to measure the ability of a given attribute to discriminate classes?
- There are many measures. The choice of an attribute usually involves the use of heuristics that look a step ahead and normally do not reconsider the options already taken. These heuristics agree on two aspects:
  - A division that maintains the proportions of classes in all partitions that becomes useless, Figure a);
  - A division where in each partition all examples are of the same class has maximum utility, Figure b).
- Partition measures can be characterized in two ways:
  - By the measure of the difference given by the proportions of the classes between the current and the descendants. It has the advantage of enhancing the purity of the partitions.
  - By the measure of the difference given by a function based on the proportions of the classes between the descendants. In this case, the disparity between the partitions is valued, and being a measure of independence, it is also a measure of the degree of association between the attributes and the class.



a) Division into classes keeping the proportion (not very useful)



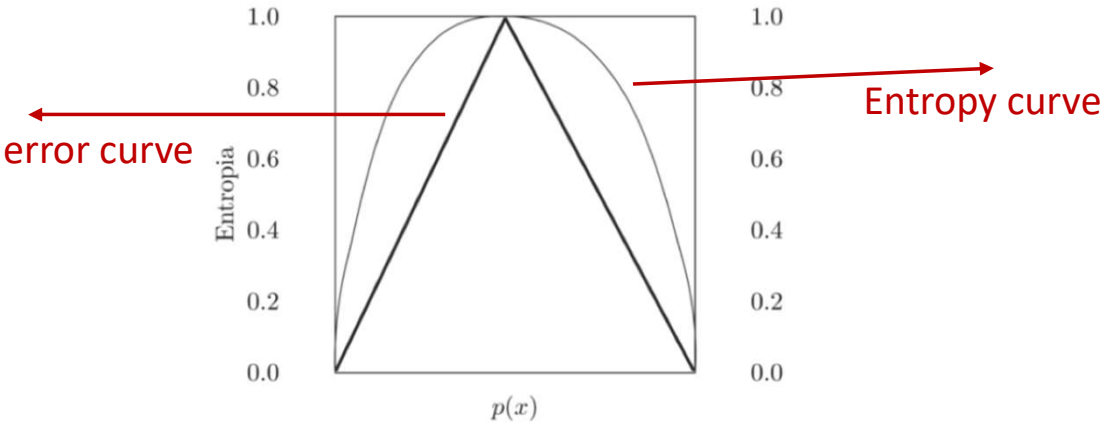
b) Division into maximum classes (very useful)

Entropy

- Entropy is a measure of the randomness of a variable.
- The entropy of a nominal variable X that can take ***i*** values can be calculated using the following equation:

$$Entropy\ (X) = -\sum_i p_i \times \log_2 p_i$$

- Entropy has a maximum ($log_2 p_i$) if $p_i = p_j$ if for any different *i* of *j* and *Entropy (X) = 0* if and only if there is an *i* that $p_i = 1$. It is assumed *that* $0 \times log_2 0 = 0$



Gain Information

- In the context of DT, entropy is used to estimate the German torment of the variable to predict: the class. The question that arises is to know for a given set of examples, which attribute to choose for test?
- In the context of DT, entropy is used to estimate the randomness of the variable to be predicted: the class.
- The question that arises is: **for a given set of examples, which attribute to choose for testing?**
- We know that the values of an attribute define partitions from the set of examples. On the other hand, the information gain measures the reduction in entropy caused by the partition of the examples according to the attribute values, as shown in the formula:

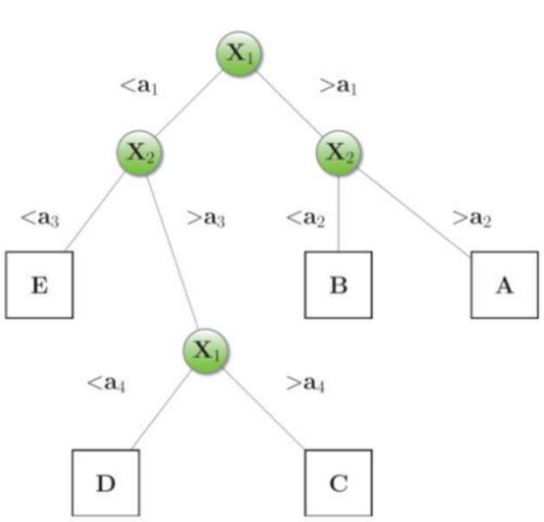
$$\text{Gain(Examples, Attributes)} = \text{Entropy(examples)} - \sum \frac{\#\text{Examples}_v}{\#\text{Examples}} \times \text{Entropy(examples)}$$

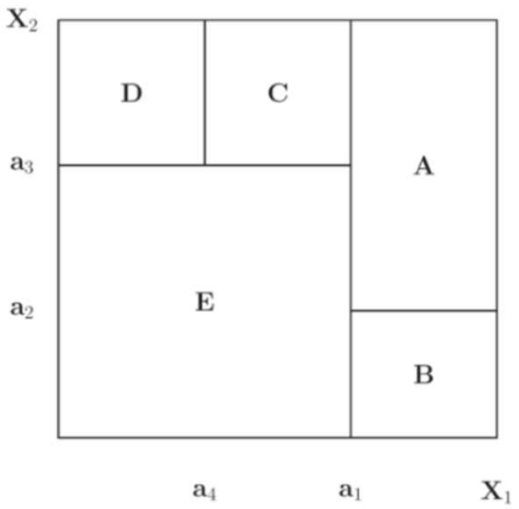
The construction of a DT is guided by the objective of decreasing entropy, that is, the randomness - prediction difficulty - of the objective variable.

Stopping criteria

The question that now arises is when should we stop dividing the examples. We can see the following four possibilities:

1. All examples belong to the same class
2. All examples have the same attribute values, but different classes
3. The number of examples is below a certain limit.
4. The merit of all possible partition tests of the examples is very below





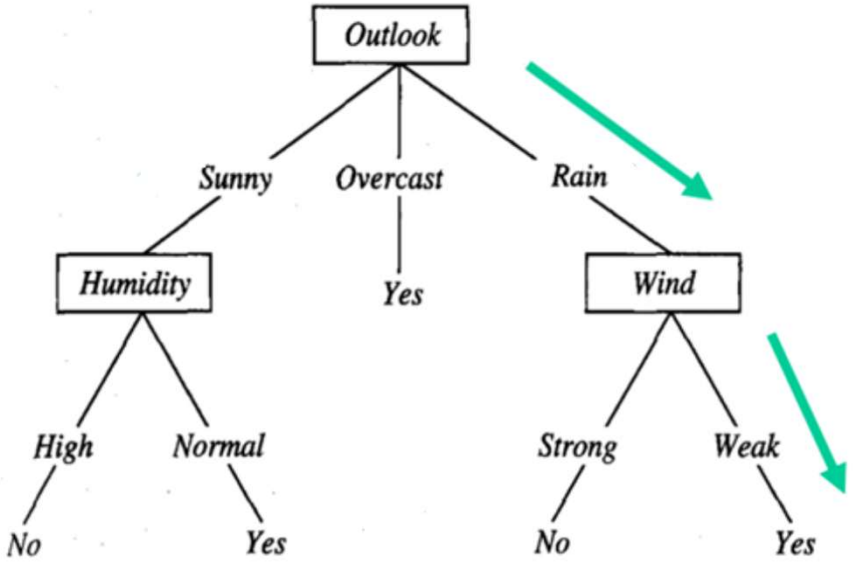
The DT effect a partition of the space of the attributes as shown in Figure above.
Let us see how partition A is obtained in the attribute space.
If the variable $X_1 > a_1$ and $X_2 > a_2$, that is, if the path in the tree is on its right side, from the point to the root, effectively the partition to that found in A.
we find this partition if we assign these values to the attributes. The reader is invited to check the other partitions from B to E, assigning the possible values of the attributes that configure the branches of the tree on the left side.

Decision Trees

Concept: “good day to play tennis”

- Given the instance <Outlook=Sunny, Wind=Weak>

- In this case, analyzing the tree, it is possible to classify (decide) each situation following the tree, for example if it is raining today, but it is not wind it means that the decision could be go to play.



Decision Trees

- Decision trees seek to create a set of leaf nodes that are as “pure” as possible, where each of the records in a leaf node has the same classification. The decision tree may provide classification assignments with the highest measure of confidence available.
- We shall examine two of the many methods for measuring leaf node purity, which lead to the two leading algorithms for constructing decision trees:
  - CART algorithm
    - produces DT that are strictly binary, containing exactly two branches for each decision node.
    - recursively partitions the records in the training data set into subsets of records with similar values for the target attribute.
    - grows the tree by conducting for each decision node, an exhaustive search of all available variables and all possible splitting values, selecting the optimal split according to the Gini Index
  - C5.0 algorithm
    - is the J. Ross Quinlan’s extension C4.5 algorithm for generating DT
    - is not restricted to binary splits
    - uses the concept of information gain or entropy reduction to select the optimal split. Suppose that we have a variable X whose k possible values have probabilities p1, p2 , …, pk. The smallest number of bits, on average per symbol, needed to transmit a stream of symbols representing the values of X observed is called the entropy of X, defined as $H(X) = -\sum_j p_j \log_2(p_j)$

J. Ross Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann, San Francisco, CA, 1992