Decision trees

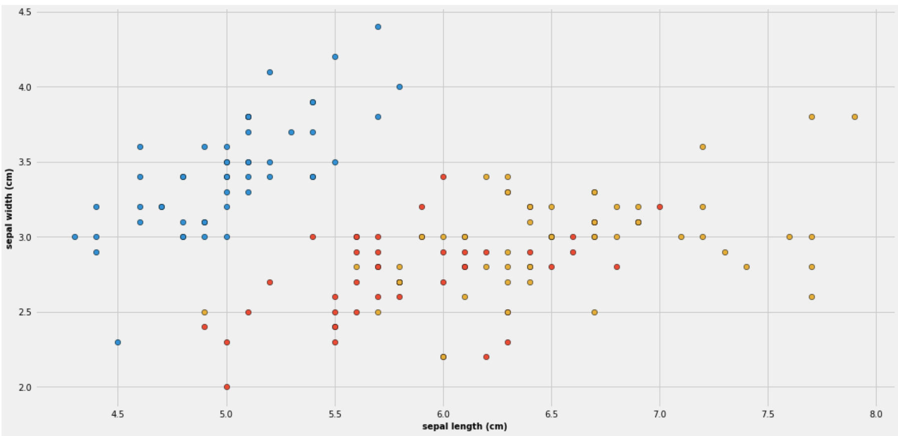
Course of Machine Learning
Master Degree in Computer Science
University of Rome "Tor Vergata"
a.a. 2024-20245

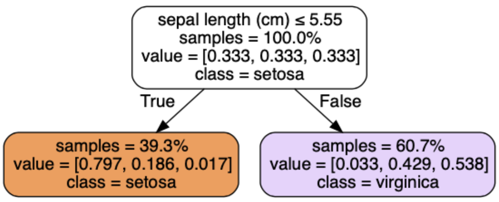
Giorgio Gambosi

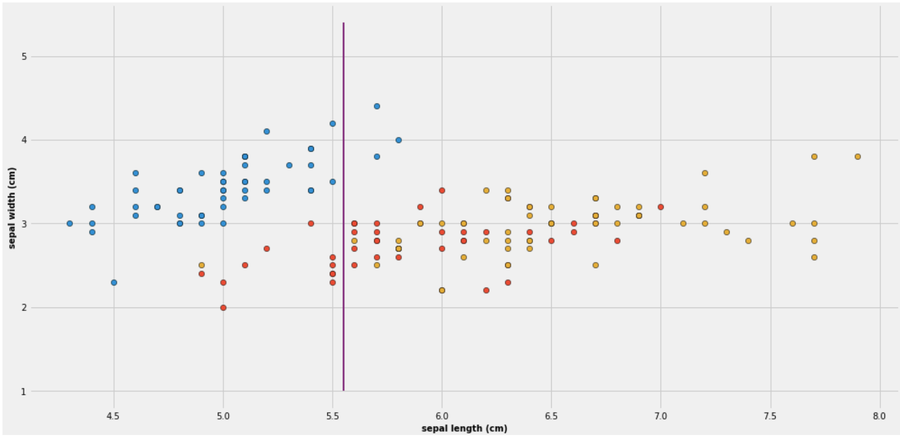
Decision tree

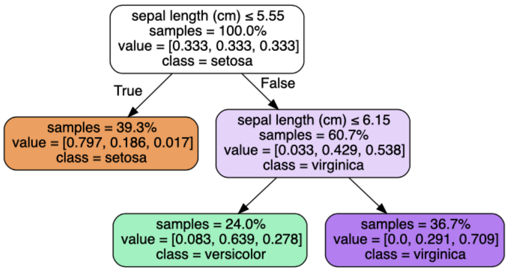
A **decision tree** is a classifier expressed as a recursive partition of the instance space.

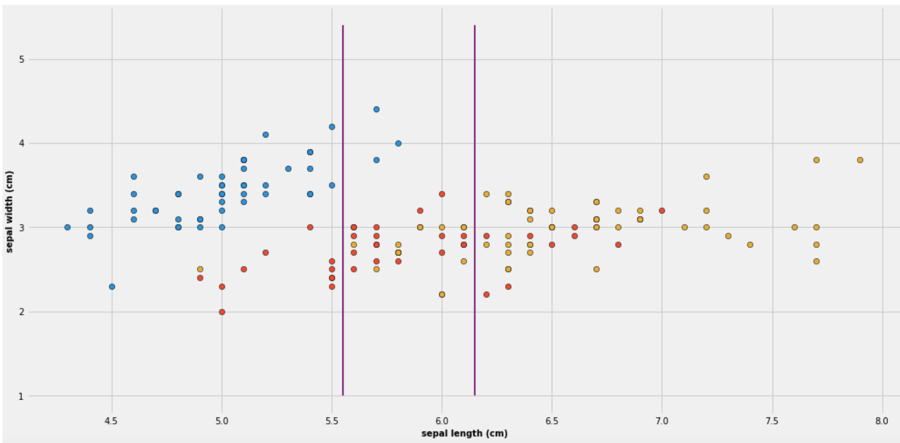
- It consists of nodes that form a rooted tree
- Each internal node splits the instance space into two or more subspaces, according to a given discrete function of the attributes values
- Usually, each node is associated to a partition wrt a single attribute
- Each leaf is associated to a subspace and:
  - either a class, representing the most appropriate prediction for all points in the subspace
  - or a vector of class probabilities

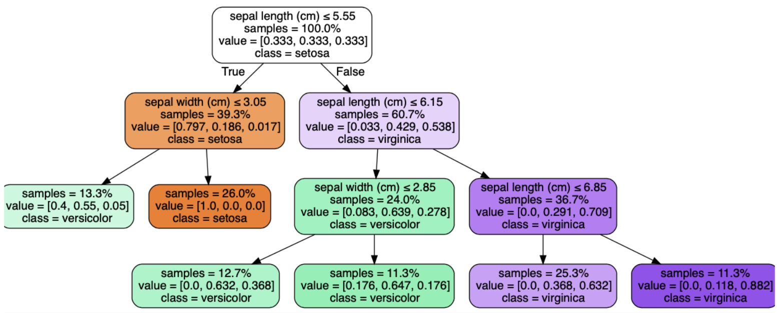


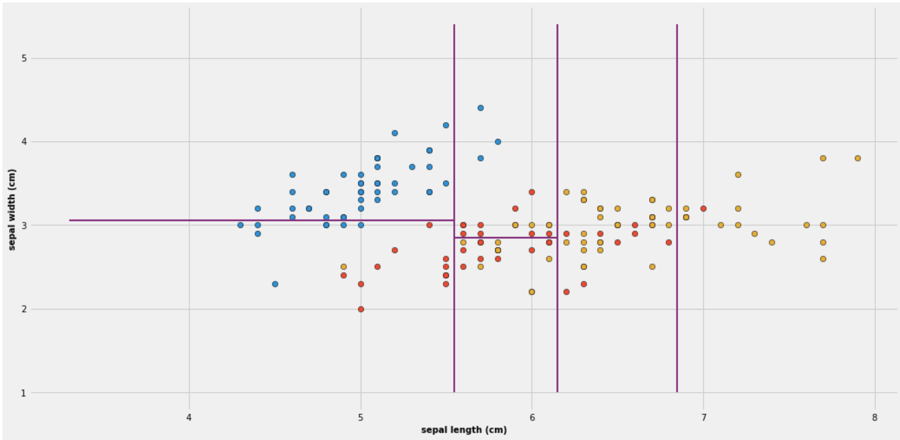



sepal width (cm)
sepal length (cm)
1
2
3
4
5
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0


sepal length (cm) ≤ 5.55
samples = 100.0%
value = [0.333, 0.333, 0.333]
class = setosa
True
False
samples = 39.3%
value = [0.797, 0.186, 0.017]
class = setosa
sepal length (cm) ≤ 6.15
samples = 60.7%
value = [0.033, 0.429, 0.538]
class = virginica
samples = 24.0%
value = [0.083, 0.639, 0.278]
class = versicolor
samples = 36.7%
value = [0.0, 0.291, 0.709]
class = virginica


sepal width (cm)
sepal length (cm)
1
2
3
4
5
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0


sepal length (cm) ≤ 5.55
samples = 100.0%
value = [0.333, 0.333, 0.333]
class = setosa
True
False
sepal width (cm) ≤ 3.05
samples = 39.3%
value = [0.797, 0.186, 0.017]
class = setosa
sepal length (cm) ≤ 6.15
samples = 60.7%
value = [0.033, 0.429, 0.538]
class = virginica
samples = 13.3%
value = [0.4, 0.55, 0.05]
class = versicolor
samples = 26.0%
value = [1.0, 0.0, 0.0]
class = setosa
sepal width (cm) ≤ 2.85
samples = 24.0%
value = [0.083, 0.639, 0.278]
class = versicolor
sepal length (cm) ≤ 6.85
samples = 36.7%
value = [0.0, 0.291, 0.709]
class = virginica
samples = 12.7%
value = [0.0, 0.632, 0.368]
class = versicolor
samples = 11.3%
value = [0.176, 0.647, 0.176]
class = versicolor
samples = 25.3%
value = [0.0, 0.368, 0.632]
class = virginica
samples = 11.3%
value = [0.0, 0.118, 0.882]
class = virginica



Decision tree: classification

- Given an item $\mathbf{z} = (z_1, \ldots, z_d)^T$, the decision tree is traversed starting from its root.
- At each node traversed, with associated feature x_i and function f_i, the value $f_i(z_i)$ is computed to decide which is the next node to be considered, among the set of children nodes. This is equivalent to considering smaller and smaller subregions of the space of data.
  - An important case is when a threshold θ_i is defined and a comparison between z_i and θ_i is performed to decide which is the next node to be considered, among two children nodes.
- The procedure halts when a leaf node is reached. The returned prediction is given by the corresponding class, or derived by the class probabilities vector.

Decision tree: construction

The space of data is recursively partitioned by constructing the decision tree from root to leaves.

At each node:

1. How to perform a partition of the corresponding region (choosing feature and function)?
2. When to stop partitioning? How to assign information to leaves?

Decision tree: partitioning at each node

Select the feature and function/threshold such that a given measure is maximized within the intersections of the training set with each subregion. Measures of class impurity within a set. To be minimized.

Inpurity measure

Given a random variable with discrete domain $\{a_1, \ldots, a_k\}$ and corresponding probabilities $p = (p_1, \ldots, p_k)$, an impurity measure $\phi : p \mapsto \mathbb{R}$ has the following properties

- $\phi(p) \geq 0$ for all possible p
- $\phi(p)$ is minimum if there exists $i, 1 \leq i \leq k$ such that $p_i = 1$
- $\phi(p)$ is maximum if $p_i = 1/k$ for all i
- $\phi(p) = \phi(p')$ for all p' deriving from a permutation of p
- $\phi(p)$ is differentiable everywhere

Goodness of split

In our case, we consider the class of each item in S.

- For any set S of items, the probability vector associated to S can be defined as $p = \left(\frac{|S_1|}{|S|}, \ldots, \frac{|S_k|}{|S|}\right)$, where $S_h \subseteq S$ is the set of elements of S belonging to class k.
- Given a function $f : S \mapsto \{1, \ldots, r\}$, let $s_i = \{x \in S | f(x) = i\}$ (that is, $x \in s_i$ iff $f(x) = i$). The goodness of split of S wrt f is given by

$$\Delta_\phi(S, f) = \phi(p_S) - \sum_{i=1}^{r} p_i \phi(p_{s_i})$$

that is, by the difference between the impurity of S and the mean of impurities of the subsets resulting from the application of f

Goodness of split

In practice, f is usually defined by considering a single feature and:

- if the feature is discrete, inducing a partition of its codomain in k subsets
  - as a special case, the partition is among items with the same value for the considered feature
- if it is continuous, inducing a partition of its codomain in a set of intervals, defined by thresholds
  - as a special case, a single threshold is given and f performs a binary partition on items in S

Entropy and information gain

- For any set S of items, let

  $$H_S = -\sum_{i=1}^{k} \frac{|S_i|}{|S|} \log_2 \frac{|S_i|}{|S|}$$

  be the **entropy** of S. Observe that the entropy is minimal (equal to 0) if all items of S belong to a same class, and maximal (equal to $\log_2 k$) if all classes are represented in S by the same number of items
- By using entropy as an impurity measure, the goodness of split is given by the **information gain**, defined as follows.

The **information gain** wrt to a partition function f is the decrease of entropy from S to the mean of entropies of s_i

$$IG(S, f) = H_S - \sum_{j=1}^{r} \frac{|s_j|}{|S|} H_{s_j}$$

Gini index

Gini index is used in many cases as a tool to measure divergence from equality. It is defined as

$$G_S = 1 - \sum_{i=1}^{k} \left(\frac{|S_i|}{|S|}\right)^2$$

- The **Gini gain** wrt to a partition function f is the decrease of Gini index from S to the weighted sum of Gini indices of s_i

$$GG(S, f) = G_S - \sum_{j=1}^{r} \frac{|s_j|}{|S|} G_{s_j}$$

Other goodness of split measures

DKM DKM is an impurity measure defined for binary classification

$$DKM_S = 2\sqrt{\left(\frac{|S_1|}{|S|}\right)\left(\frac{|S_2|}{|S|}\right)}$$

the corresponding gain is

$$DKMG(S, f) = DKM_S - \sum_{j=1}^{r} \frac{|s_j|}{|S|} DKM_{s_j}$$

Gain Ratio A version of the information gain normalized wrt the original entropy

$$GR_S = \frac{IG(S, f)}{H_S}$$

Other measures can be defined and applied

Decision tree: leaves

Often, conditions for deciding when partitioning has to stopped are predefined (maximum tree depth, maximum number of leaves, number of items in a subregion).

When a leaf is reached, the corresponding class can be defined as the majority class in the intersection of the subregion and the training set.

Pruning

- Early stopping tends to create small and underfitted decision trees.
- Loose stopping tends to generate large and overfitted trees.

Pruning methods can be applied to deal with the problem.

1. A loose stopping criterion is used, letting the decision tree overfit.
2. The overfitted tree is cut back into a smaller tree by suitably removing branches that seem not to contribute to the generalization accuracy. Different subtrees are merged into single nodes, thus reducing the tree size.