Machine learning

Decision trees

Corso di Laurea Magistrale in Informatica

Università di Roma Tor Vergata

Prof. Giorgio Gambosi

a.a. 2021-2022

Algoritmo che si presta bene per la classificazione.

Decision tree

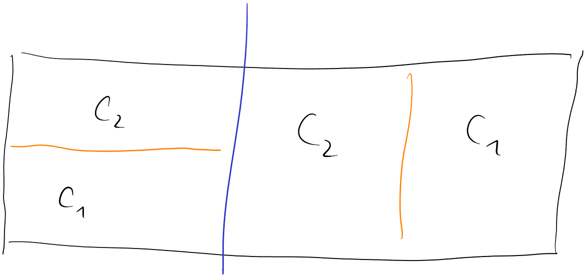
Si fa una decomposizione ricorsiva dello spazio delle feature, che sarà rappresentato da un albero. Otteniamo un albero di decomposizione.

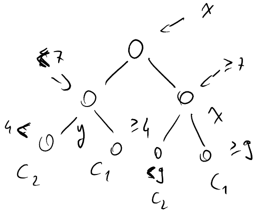
A decision tree is a classifier expressed as a recursive partition of the instance space.

- 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

Unione delle regioni delle foglie come tutto, per assegnare un punto ad una classe scendo lungo l'albero ed assegno alla regione. Sulla foglia posso anche avere una distribuzione di probabilità delle classi.

Piano delle feature.





Nell'albero perdo informazione rispetto al taglio: devo precisae rispetto a che asse sto tagliando ed inoltre rispetto a quale valore sto tagliando, es:

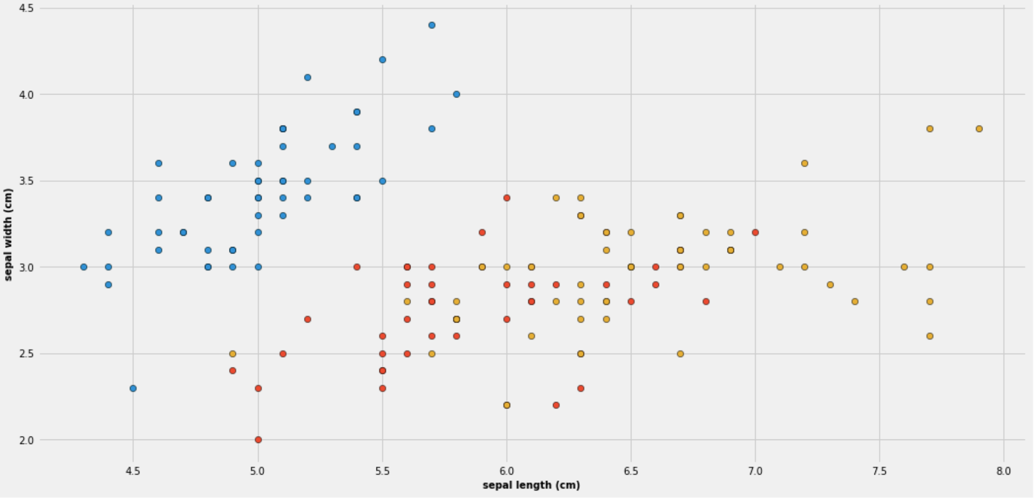
- rispetto all'asse x
- per valore 7

Supponiamo di avere anche assegnate le diverse zone alle due classi: le foglie avranno una etichetta che dice a quale classe va assegnato il punto.

Partendo da un x, es x=2.5 possiamo partire dalla radice e scendere facendo i confronti fino ad una foglia dove a questo punto trovo la classe associata.

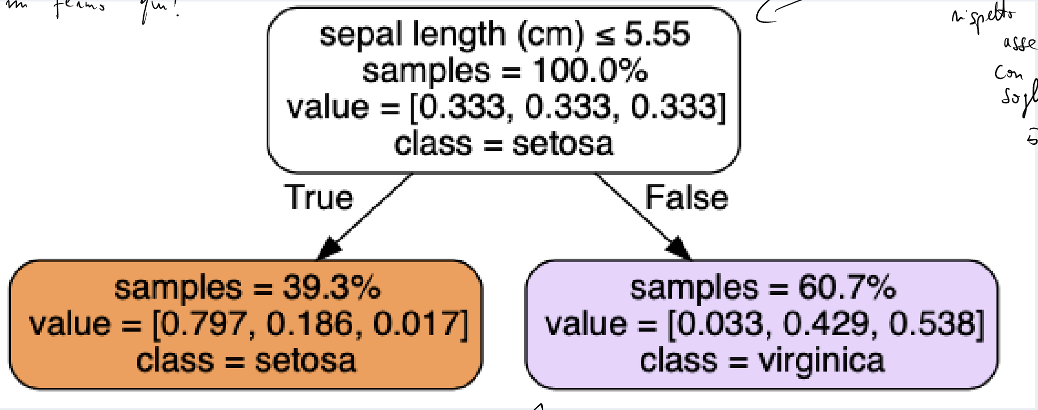
La decomposizione è sempre rispetto ad uno degli assi, sono tutte ortogonali fra loro, l'albero rappresenta la decomposizione ricorsiva che arriverà ad un certo numero di regioni con un certo criterio che dice quando fermarsi.

Decision tree



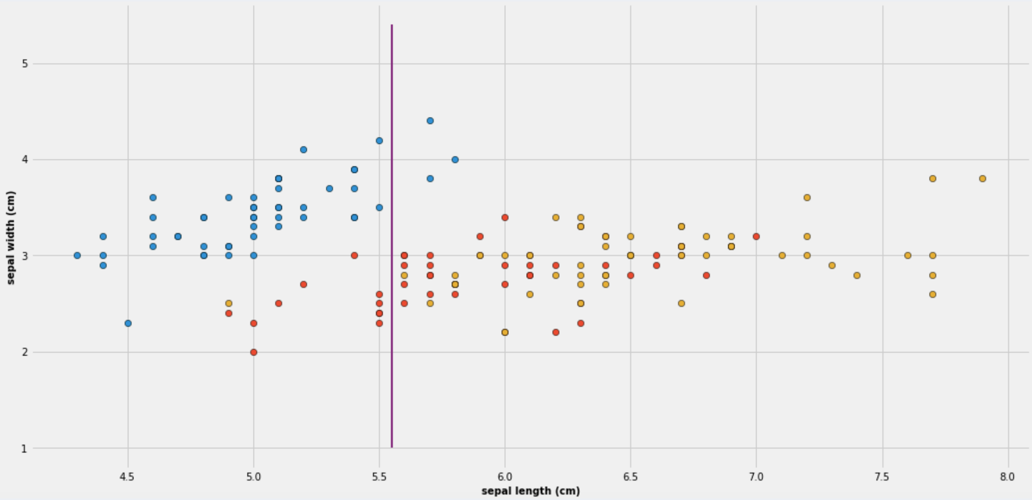
Come mai 5.55? Ma poi, mi fermo qui?

modo 1: taglia rispetto all'asse x con la soglia di 5.55



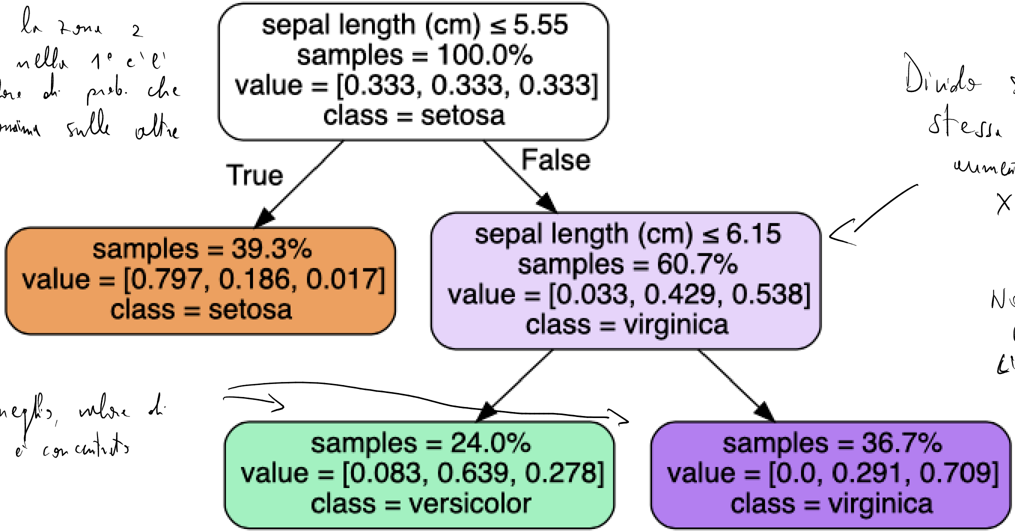
n° di elementi nelle due regioni risultanti.

Decision tree



Decision tree

Divido la zona 2 perché nella 1° c'è un valore di prob. che predomina sulle altre

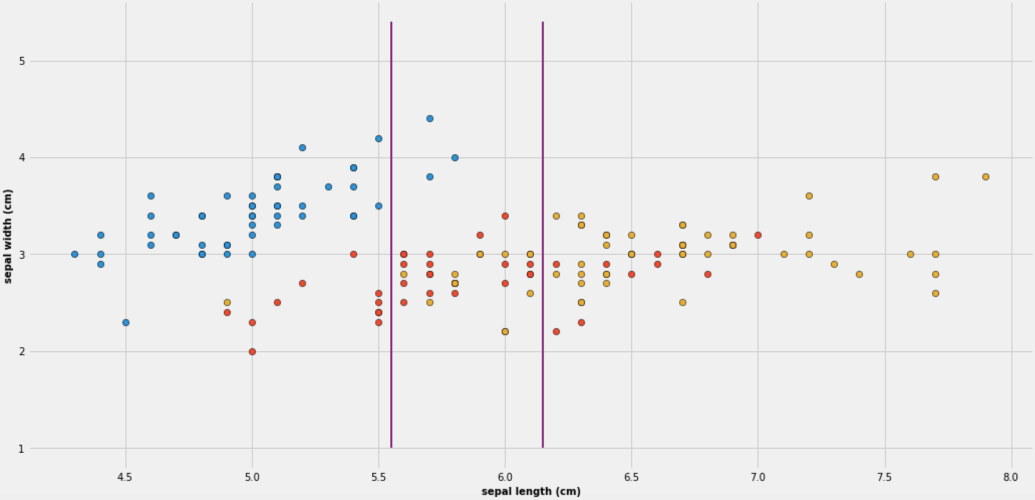


Divido sulla stessa feature aumentando X

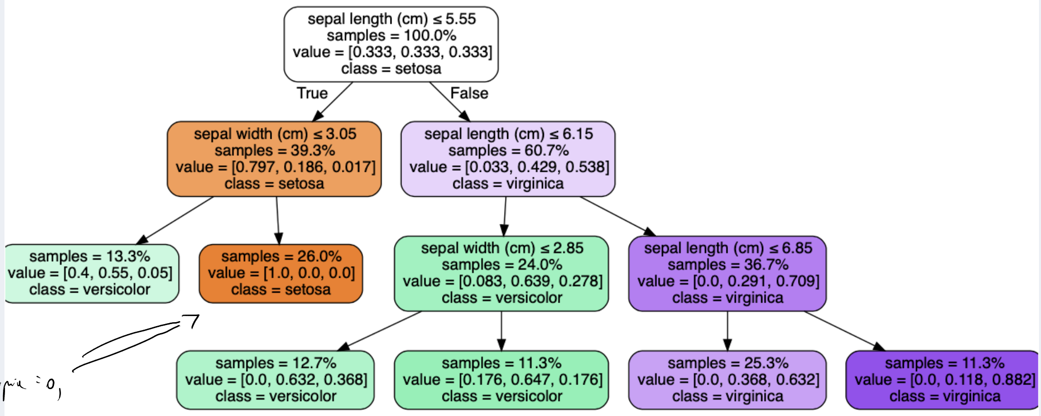
NON CI PIACE L'ENTROPIA: se $P(C_i|x)$ sono uguali tra loro la diminuisco tagliando

Va meglio, valore di prob. è concentrato

Decision tree

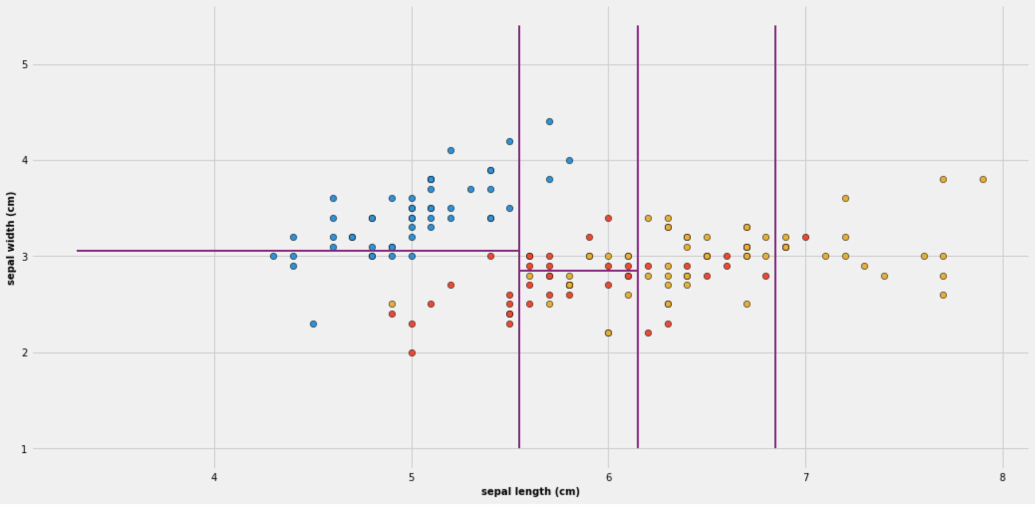


Decision tree



Entropia = 0,
mi fermo

Decision tree



Decision tree: classification

- Given an item $\mathbf{z} = (z_1, \dots, z_d)^T$, 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

Caratteristica: fornendo una classe, guardando al nodo dell'albero posso avere un'idea del perché viene etichettato come tale. es: punto è "setosa" perché ampiezza sepalo ≤ 3.5 e altezza < 5.5. Ho una spiegazione del perché: classificazione più explainable.

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

At each node:

su quale asse

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?

Non ha senso arrivare a regioni omogenee, mi fermo prima.

Costruiamo il classificatore, gli diamo un punto e questo ci dice che classe è.
Vedendolo ad esempio in ambito meidco, sulla base di dati come immagini di risonanze magnetiche il classificatore dice che puoi avere un problema.

Parlando però con un medico, manca il perché: quali sono gli elementi che fanno si che la risposta sia quella è l'explainability ovvero avere un sistema che non dia solo la predizione ma che dica anche sulla base di cosa è così.

Ci sono approcci che tendono a facilitarla ed approcci che la rendono più difficile: USA con sistema di classificazione che valutava se offrire benefici dando certe risposte. Guardando cosa accedeva si è scoperto che c'era un bias, offrendo meno queste opportunità alle persone di colore (MA GUARDA UN PO', PROPRIO IL BIAS RAZZIALE).

Occorre quindi avere alla base un modo per capire come il sistema è arrivato a quella decisione, nel caso del Decision Tree è più semplice.

Decision tree: partitioning at each node

entropia, in un caso

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.

Impurità: quanto sono lontano dal fatto che tutti gli elementi siano della stessa classe.

Inpurity measure

Given a random variable with discrete domain $\{a_1, \dots, a_k\}$ and corresponding probabilities $p = (p_1, \dots, p_k)$, 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

Se tutte le classi sono equiprobabili.

Col − : quanto tolgo di impurità tagliando così

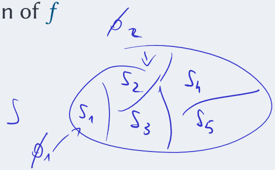
In our case, we consider the class of each item in S.

- 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

impurità prima del taglio

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

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



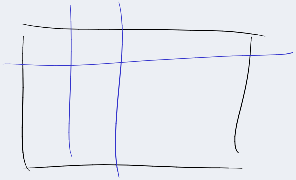
Livello di impurità dell'insieme pesa di più se ha più elementi.

Goodness of split

Il task dovrebbe essere dividere S così da avere $\Delta\phi(S,f)$ più alto possibile.

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



Dovrei confrontare i livelli di impurità dei vari modi di dividere

Entropy and information gain

$$= -\sum_i p_i \log_2 p_i$$

- For any set S of items, let

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

  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

Anche qui ci sono vari criteri empirici su quando fermarsi.

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.