Classificazione

Siamo alla ricerca di un modo sistematico per stabilire a priori la classe di appartenenza di una osservazione multidimensionale, in base ad una variabile target ed altre variabili ausiliarie

Le classi di potenziale appartenenza sono conosciute (grazie ad una sperimentazione preliminare) sia per numero che per caratteristiche.

$$C = \{c_1, c_2, \ldots, c_m\}, \quad c_i \cap c_j = \varnothing \;\; se \;\; i \neq j, \quad \bigcup_{i=1}^{m} c_i = C$$

Ogni entità può ricadere in uno -ed uno solo- gruppo di C

Un classificatore è una funzione **d** che associa una classe **c**, dell'insieme ***C*** di tutte le classi, ad ogni vettore **x** del data set ***X*** *e dei nuovi dati che si produrranno*

$$\mathbf{d(x)} = \mathbf{c} \qquad \mathbf{x} \in \boldsymbol{X}, \mathbf{c} \in \boldsymbol{C}$$

La matrice dei dati

E' una organizzazione righe-colonne (nxm) collocata al centro di molte tecniche di analisi multivariata

Variabili ⇓

$$entità \Rightarrow \mathbf{X} = \begin{array}{c|cccc} & X_1 & X_2 & X_j & X_m \\ \hline U_1 & x_{11} & x_{12} & x_{1j} & x_{1m} \\ U_2 & x_{21} & x_{22} & x_{21} & x_{2m} \\ & & \ldots & & \\ U_i & x_{i1} & x_{i2} & x_{ij} & x_{im} \\ \\ U_n & x_{n1} & x_{n2} & x_{in} & x_{nm} \end{array}$$

Alcuni valori di una variabile o di diverse variabili su di una o più entità potrebbero mancare

La matrice dei dati è suddivisa in due altre matrici nel rapporto, ad esempio, 1 a 4 per formare il data set di prova (train) e il data set di verifica (test).

La qualità del decisore si commisura alla performance sui dati test di ciò che si è ottenuti nei dati train.

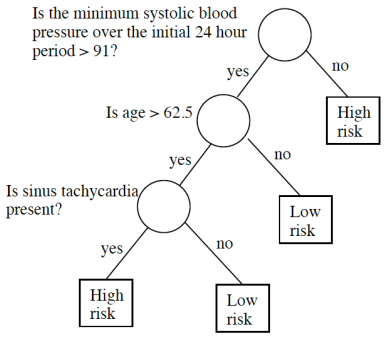
Classification and Regression Trees

E' la metodologia statistica introdotta da Breiman et al. nel 1984. Produce una classificazioni nella forma di diagramma ad albero rovesciato

Un caso clinico per il CART:
Prevedere quali tra i pazienti ad alto rischio sopravviveranno almeno 30 gg in base alle loro condizioni nelle prime 24 h

In tale frangente si rilevano 19 variabili tra le quali: pressione sanguigna, età, battiti cardiaci, etc.

Lo schema di decisione potrebbe essere l'albero sulla destra



Ciascuna unità percorre l'albero dalla radice (il nodo più alto dell'albero) verso il basso, procedendo ad una classificazione per ciascun nodo che trova sul suo cammino, sino al raggiungimento di un nodo terminale o foglia.

Tipologia delle variabili

Le variabili del data set sono considerate di due soli tipi

Ordinabili: le modalità sono espresse con dei numeri che esprimono il grado di contenuto del fenomeno rappresentato: sono incluse le scale a rapporti, intervallari e ordinali

Non ordinabili: le modalità sono espresse con delle etichette distintive (occasionalmente anche da numeri) che esprimono una qualità o una condizione distintiva rispetto alle altre, ma senza che sia possibile proporre un loro ordinamento

Le variabili sono rilevate su tutte le unità e se qualcuna non risponde o il valore non è conosciuto o conoscibile si considera mancante (missing)

Problema affrontato

Un data set contiene variabili rilevate con misura eterogenee

Una di queste variabili ha il ruolo di RISPOSTA rispetto altre variabili che hanno il ruolo di REGRESSORI

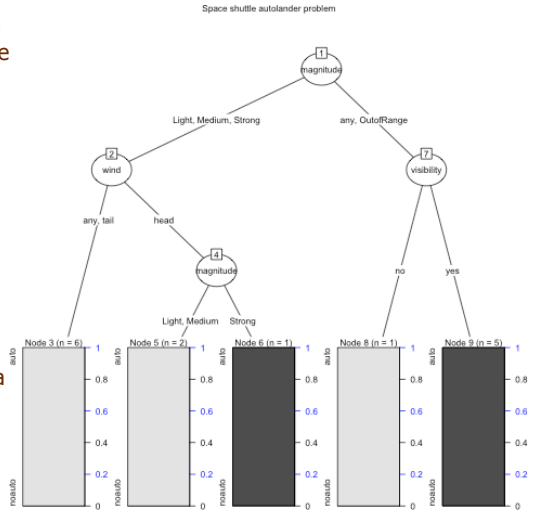
Si ipotizzano relazioni non lineari o indirette tra la risposta e regressori

E' necessario prevedere il valore della risposta di una unità non ancora analizzata (Regression)

Ovvero stabilire la classe più plausibile di appartenza della nuova unità (Classification)

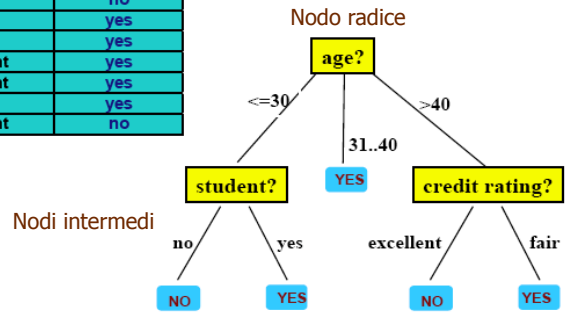
Si procede per bipartizioni successive di un gruppo in due sottogruppi iniziando da quello formato dall'intero data set

Shuttle landing problems



Esempio. Credito al consumo

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no



Idea guida del Cart

Passo_1: si sceglie un regressore X e si bipartiscono le unità in DUE gruppi in base ai valori o alle categorie di X.

Se il regressore è su scala ordinale, la suddivisione avviene per un valore di soglia X_s :

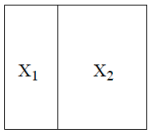
unità con $X \le X_s$ e unità con $X > X_s$

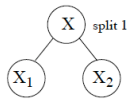
Se il regressore è una politomia non ordinata, la suddivisione avviene rispetto ad una delle categorie, diciamo A_s

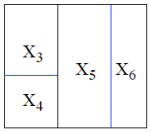
unità con $X = A_s$ e unità con $X \neq A_s$

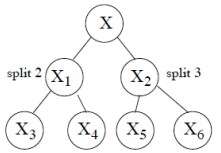
La stessa variabile può essere utilizzata più volte in livelli diversi dell'albero

Idea guida del Cart/2









Ogni cerchio rappresenta un nodo ed ogni nodo è denotato con **t**.

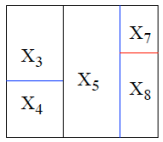
Il nodo può essere terminale oppure suscettibile di split: il subnodo di sinistra si indica con $\mathbf{t_L}$ quello di destra con $\mathbf{t_R}$.

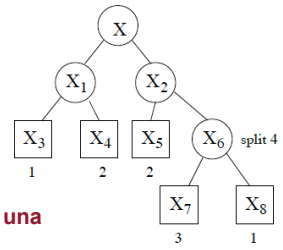
L'insieme di tutti i nodi e subnodi ha come simbolo **T**;

Uno split è indicato con **s**. E l'insieme di tutti gli split con **S**.

I nodi terminali sono indicati con dei quadrati.

I nodi terminali costituiscono la partizione del data set rispetto alla variabile target.





Ogni nodo terminale deve ricevere l'etichetta di una ed una sola delle classi previste nel data set

Idea guida del Cart/3

Passo_2: Ognuno dei due gruppi è a sua volta suddiviso in due sottogruppi con la stessa procedura.

La suddivisione è ricorsiva e termina quando l'ultima suddivisione ha raggiunto un grado di omogeneità soddisfacente nei gruppi finali.

Oppure termina quando gli annidamenti sono troppi per essere realmente utili.

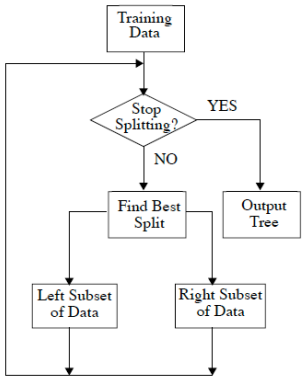


Figure 1. Flow chart Used for Splitting.

- **Come si definisce l'omogeneità/eterogeneità di un gruppo?**
- **Come si sceglie il regressore da bipartire?**
- **Come si decide se il nodo è terminale ovvero deve essere ancora suddiviso?**

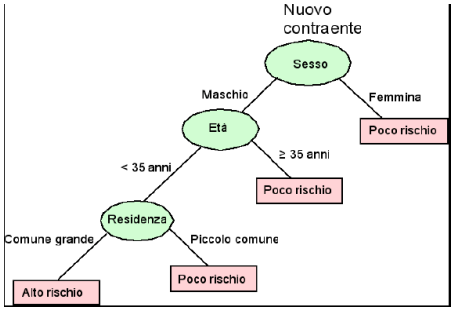
Esempio

Si realizzano delle osservazioni su alcuni assicurati sui quali si rilevano le variabili di interesse.

Si analizzano i dati così raccolti per costruire delle procedure stilizzate di individuazione delle polizze (percorsi).

Al presentarsi del nuovo contraente si acquisiscono i dati solo sulle variabili che sono state effettivamente coinvolte in almeno in un percorso.

Si segue il percorso e si colloca l'assicurando al livello di rischio che compare alla fine del percorso.



I nuovi contraenti sono divisi in due distinte categorie di rischio:

BASSO (donne, maschi di età ≥ 35, maschi di età <35 residenti in piccoli comuni)

ALTO: (Maschi di età inferiore a 35 anni residenti in comuni grandi).

Uso dell'albero

La nuova osservazione $X=(x_1,x_2,\ldots,x_m)$ deve essere assegnta ad una delle classi/categorie previste nel data set. Ogni split dipende dal valore di una delle variabili.

Se i valori di questa sono ordinati ci si chiede $\left(x_j \le M_j\right)?$

dove M_j è una delle modalità della j-esima variabile. Poiché i valori distinti sono al massimo n (numero di entità) le potenzialità di bipartizioni sono in numero limitato.

Se i valori sono categorie non ordinabili, queste sono distribuite in due sottoinsiemi A_j e A_j^c e ci si chiede

$$\left(x_j \in A_j\right)?$$

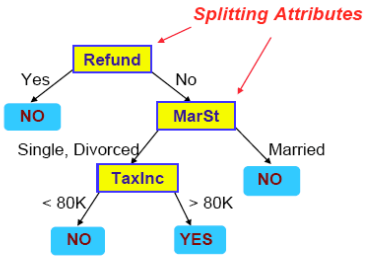
Anche i questo caso le bipartizioni possibili sono un numero finito.

L'insieme delle domande definisce la classificazione.

Altro esempio

Esempio di albero di classificazione

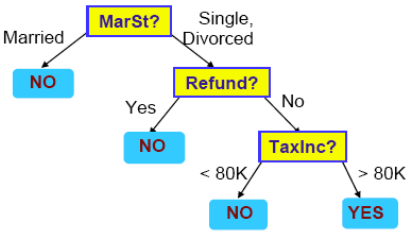
	categorical	categorical	continuous	class
Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



Esempio (continua)

Un altro albero di classificazione

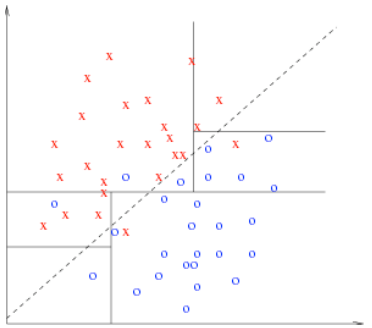
	categorical	categorical	continuous	class
Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



Si possono derivare più alberi dagli stessi dati!

Splitting monotetici

Gli split avvengono in base ad una variabile alla volta: sono perpendicolari per le variabili metriche.



Tali suddivisioni sono inefficienti se le variabili sono correlate.

Nel caso in figura ci vorranno molti split e tanti rettangoli per approssimare l'iperpiano che separa i due gruppi.

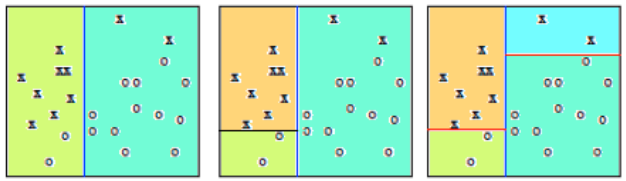
In questo caso funziona meglio la combinazione lineare

$$\sum a_j x_{.j} \leq c?$$

Un problema analogo si può riscontrare con variabili nominali che tendono ad stringere forti associazioni in tanti casi applicativi

Omogeneità ed eterogeneità nei gruppi

La suddivisione deve avvenire in modo tale che le entità nei due subnodi presentino, rispetto alla variabile di splitting, una omogeneità maggiore (minore eterogeneità) rispetto a quella posseduta dal nodo originale



La qualità dello split è misurata da una qualche funzione che esprime la omogeneità dei gruppi ovvero il miglioramento ottenuto nel passare dal nodo principale ai due nuovi subnodi.

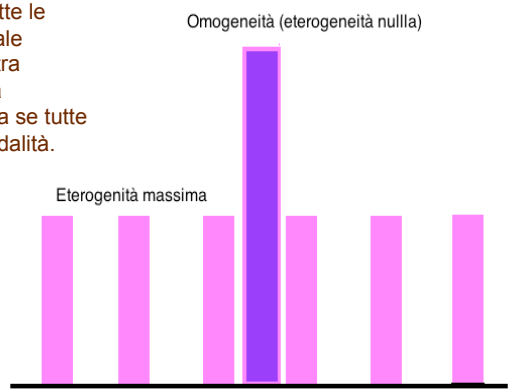
La misura deve anche essere utile per stabilire se la suddivisione è efficace ovvero conviene considerare il nodo come terminale

Variabilità e mutabilità

L'idea di differenziazione richiama in genere il concetto di variabilità che però è applicabile alle sole variabili metriche.

Quando la scala di misurazione non è metrica la variabilità ovvero la mutabilità, si può interpretare in termini di eterogeneità (per variabili qualitative) e di bipolarità (variabili quantitative ordinali).

Una variabile è eterogenea se tutte le categorie sono presenti con uguale frequenza: il fenomeno non mostra preferenza evidente per nessuna modalità; la variabile è omogenea se tutte le unità presentano la stessa modalità.



Misura della eterogeneità

E' un indice Φ definito sulle frequenze relative viste come probabilità dei vari gruppi. Supponiamo che siano presenti k classi incompatibili ed esaustive e sia

Classe	1	2	…	j	…	k
Probability	p_1	p_2	…	p_j	…	p_k

$$p_j \geq 0 \quad j = 1,2,\ldots,k; \quad \sum_{j=1}^{k} p_j = 1$$

L'indice di eterogeneità deve...

- essere basato sulle sole frequenze
- essere invariante rispetto all'ordine con cui si considerano le categorie
- Assumere valori crescenti in funzione del livello di eterogeneità
- raggiungere il valore massimo –dipendente dal numero di categorie- se le classi sono equiprobabili e cioè sono tutte presenti in egual misura nel nodo
- raggiungere il valore minimo allorché le classi hanno tutti tranne una probabilità zero ed una sola ha probabilità uno.

Indici di eterogeneità

$$\textit{Entropia}: \quad E_1 = -\sum_{j=1}^{k} p_j \log(p_j)$$

$$\textit{Disomogeneità}: \quad E_2 = 1 - \max_{1 \leq j \leq k} \{p_j\}$$

$$\textit{Diversità (Gini)}: E_3 = \sum_{j=1}^{k} p_j (1 - p_j) = 1 - \sum_{j=1}^{k} p_j^{\,2}$$

$$\textit{Dissimilarità}: \quad E_4 = 2\left(\frac{k-1}{k}\right) - \sum_{j=1}^{k} \left| p_j - \frac{1}{k} \right|$$

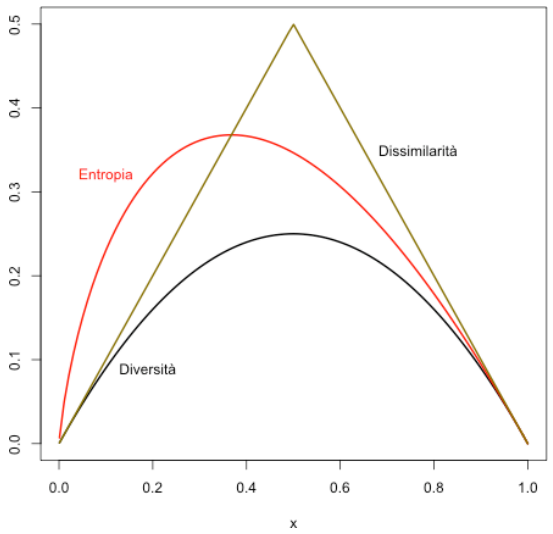
$$\textit{Cisbani - Frosini}: \quad E_5 = \sqrt{\left(\frac{k-1}{k}\right)} - \sqrt{\sum_{j=1}^{k} \left(p_j - \frac{1}{k} \right)^2}$$

La scelta tra questi indici dipende da quale caratteristica della mutabilità interessa.

Dipende anche dalla sensibilità che l'indice mostra nel distinguere le situazioni intermedie spesso così ravvicinate da non potersi analizzare senza un buon indicatore.

Si intende che la procedura CART agisce allo stesso modo a prescindere da come è misurata la eterogeneità

Indici di eterogeneità/2

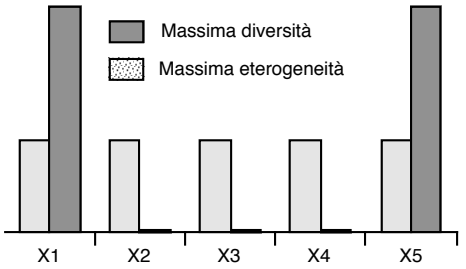


I nodi che contengono entità con un solo tipo di categoria hanno eterogeneità nulla (massima omogeneità)

La bipolarità

Si applica varaibili su scalae ordinali ed è analoga alla eterogeneità

Cambia il significato di "differenziazione massima" che si realizza quando la metà dei soggetti si colloca nel primo livello della variabile e l'altra metà nell'ultimo.



Per misurare la dispersione delle variabili ordinali si usano indici basati sulle frequenze cumulate

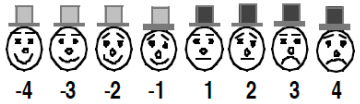
Misura della bipolarità

$$Indice\ di\ diversità\ di\ Gini: D_1 = \sum_{i=1}^{k-1} P_i(1-P_i);\ P_i = \sum_{j=1}^{i} p_j$$

Se le modalità si presentano con uguale frequenza (assenza di polarizzazione) si ha $D_1=[(k^2-1)/6k]$.

Se le frequenze si bipartiscono tra le categorie agli estremi si ha $p_1 = 0.5$, $p_k=0.5$ e quindi $P_i=0.5$ per i<k ed ovviamente $P_k=1$ per cui $D_1= (k-1)/4$.

Esempio: differenziale semantico con punteggi calibrati come una scala di Stapel e con i quali è stata acquisita l'opinione su di una nuova fiction.

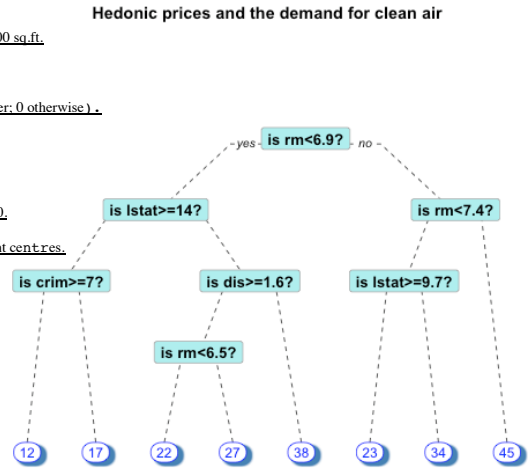


Modalità	-4	-3	-2	-1	1	2	3	4	
Freq.rel.	27	34	39	41	36	24	13	6	220

L'indice di bipolarità di Gini è $D_1=1.05$ che è inferiore al valore ottenibile in caso di massima eterogeneità: 1.31 ed è ovviamente inferiore al valore massimo 1.75.

Esempio: Boston houses e Gini index

crim
per capita crime rate by town.
zn
proportion of residential land zoned for lots over 25,000 sq.ft.
indus
proportion of non-retail business acres per town.
chas
Charles River dummy variable (= 1 if tract bounds river; 0 otherwise).
Nox
nitrogen oxides concentration (parts per 10 million).
rm
average number of rooms per dwelling.
age
proportion of owner-occupied units built prior to 1940.
dis
weighted mean of distances to five Boston employment centres.
rad
index of accessibility to radial highways.
tax
full-value property-tax rate per \$10,000.
ptratio
pupil-teacher ratio by town.
black
1000(Bk - 0.63)^2
where Bk is the proportion of blacks by town.
lstat
lower status of the population (percent).
medv
median value of owner-occupied homes in \$1000s.

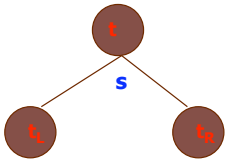


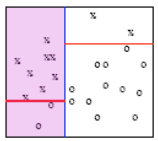
Valutazione della eterogeneità nel nodo

Quale che sia l'indice, possiamo quantificare il grado di eterogeneità presente nel nodo t-esimo. Indichiamo con i(t) (i sta per impurity)

$$i(t) = \phi\left[p(1|t), p(2|t), \ldots, p(k|t)\right];\quad dove \qquad p(j|t) = \left[\frac{p(j)}{p(t)}\right] p(t|j)$$

In base a questa espressione sul nodo t siamo in grado di valutare lo split s che divide il nodo t in due subnodi t_L e t_R.





L'efficacia dello split s del nodo t si desume dalla differenza tra la eterogeneità per il nodo t nel suo complesso, depurato dalla somma ponderata delle eterogeneità nei due subnodi derivati

$$\phi(s,t) = \Delta i(s,t) = i(t) - \left[p_L i(t_L) + p_R i(t_R)\right]$$

Eterogeneità nel nodo/2

Possiamo definire la eterogeneità ponderata del nodo t come il prodotto tra l'impurità del nodo t e la frazione di entità che ricadono nel nodo t.

$$I(t) = i(t)p(t)$$

In particolare, p(t) esprime la probabilità che una qualsiasi entità sia classificata come appartenente al nodo t, ignorando la classe. Può essere approssimata con la frazione di unità del data set incluse in t.

Quello che si è calcolato per il nodo t può essere esteso a tutti i nodi attualmente presenti nell'albero e calcolare l'eterogeneità complessiva dell'albero

$$I(T) = \sum_{t\in T} I(t) = \sum_{t\in T} i(t)p(t)$$

A livello di singolo nodo vale sempre la relazione

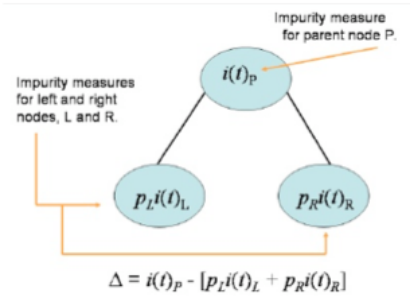
$$p(t_L) + p(t_R) = p(t);\ p_L = \frac{p(t_L)}{p(t)};\quad p_R = \frac{p(t_R)}{p(t)};\ p_L + p_R = 1$$

Eterogeneità nel nodo/3

Definiamo la differenza tra l'eterogeneità ponderate del nodo di origine ed i due nodi derivati

$$\Delta I(s,t) = I(t) - \left[I(t_L) + I(t_R)\right] = p(t)i(t) - \left[p(t_L)i(t_L) + p(t_R)i(t_R)\right]$$
$$= p(t)\left[i(t) - p_L i(t_L) - p_R i(t_R)\right] = p(t)\Delta i(s,t)$$

E' sulla base di tali quantità che dovremo decidere.



Probabilità a priori

Indichiamo con π_j la probabilità a priori della classe j. Tale probabilità esprime la tnedenza a sottoporre all'albero entità con classe j. Essa è fissata in base a cognizioni ed esperienze del fenomeno studiato.

Spesso è stimata con la proporzione delle entità del data set presenti in ogni classe:

$$\pi_j = \frac{entità\ \ in\ \ c_j}{entità\ \ totali} = \frac{n_j}{n}, \quad j = 1,2,\ldots,k$$

Le proporzioni nel data set potrebbero però essere diverse da quelle attese nelle applicazioni successive.

Infatti, la raccolta dati potrebbe essere soggetta a selection bias.

Se i dati sono stati ottenuti con uno studio di settore è verosimile che la proporzione di aziende in posizione critica sia maggiore che nella popolazione di tutte le aziende.

Se il data set può considerarsi un campione rappresentativo della popolazione allora le frequenze empiriche sono buone approssimazioni delle probabilità a priori

Probabilità condizionata

Sia n è il numero di entità del data set ed n_j, j=1,2,…,k il numero di entità nella classe/categoria c_j. Sia inoltre n(t) il numero di entità nel nodo t e $n_j(t)$ quelle entità in t che ricadono proprio la categoria j.

$$\sum_{j=1}^{k} n_j(t) = n(t); \quad n_j(t_L) + n_j(t_R) = n_j(t)$$

Ad ogni livello dell'albero, la somma di tutte le n(t) dovrebbe coincidere con n.

La probabilità che una entità con classe j si trovi nel nodo t è

$$p(t|j) = \frac{n_j(t)}{n_j} \quad con \quad p(t_L|j) + p(t_R|j) = p(t|j)$$

Queste probabilità possono essere calcolate in base alla composizione del nodo t-esimo ed alla presenza delle varie categorie j, j=1,2,…,k

Probabilità congiunta

Una semplice applicazione del teorema di Bayes consente di ottenere la probabilità congiunta che una qualsiasi entità sia di classe j e che ricada nel nodo t

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

$$p(j,t) = \pi_j p(j|t) = \pi_j \frac{n_j(t)}{n_j} \quad con \quad \sum_{j=1}^{k} p(j|t) = 1 \ \ \forall t$$

A questo punto la probabilità che una qualche entità ricada nel nodo t, a prescindere dalla sua classe di appartenenza, diventa.

$$p(t) = \sum_{j=1}^{k} p(j,t) = \sum_{j=1}^{k} \pi_j p(j|t) = \sum_{j=1}^{k} \pi_j \left(\frac{n_j(t)}{n_j}\right)$$

Quindi, le p(j|t)=p(j,t)/p(t) corrispondono alle probabilità che una entità sia di classe j-esima dato che si trova nel nodo t.

Qui sono stimate con le frequenze relative delle classi all'interno del nodo t e costituiscono delle quantità indispensabili per costruire l'albero.

Regole di stop

La costruzione dell'albero richiede due tipi di operazioni. Primo, per ogni nodo e per ogni classe dobbiamo calcolare la probabilità a posteriori di ogni classe

$$p(j|t) \quad \forall t \quad e \quad \forall j$$

Secondo, dobbiamo considerare ogni possibile split dei nodi per la variabili che rende massima la omogeneità dei nodi dell'albero.

Supponiamo di avere individuato 100 possibili split. Dobbiamo calcolare 100 distribuzioni a posteriori per le classi del subnodo a sinistra e 100 per il subnodo a destra e poi effettuare 100 valutazioni della misura di omogeneità prescelta.

Solo uno split verrà effettuato -se sarà effettuato- e solo per questo saranno eventualemente ammessi i subnodi che diventeranno poi nodi a tutti gli effetti.

Lo split NON sarà effettuato se

$$\max_{s \in S} \Delta I(s,t) < \beta$$

Se la soglia critica β on è superata, il nodo è considerato terminale dato che il decremento di eterogeneità è considerato trascurabile

Regole di stop/2

La regola appena data è considerata molto insoddisfacente. Infatti, la bipartizione è sempre "a vista" e accettare un cattivo split in una fase può anche preludere ad un ottimo spli in una fase successiva

x	x	x	0	0	0
x	x	x	0	0	0
x	x	x	0	0	0
0	0	0	x	x	x
0	0	0	x	x	x
0	0	0	x	x	x

Poiché il Cart procede per linee verticali e orizzontali, ogni taglio avrà in questo esempio circa la stessa proporzione di "x" e di "0". Ogni riga o colonna diverse da quelle segnate porterà a violare la soglia critica ed il Cart si fermerà ad una configurazione sbagliata.

Il problema non si risolve considerando più split in successione. E' necassaria una procedura aggiuntiva detta pruning

Regole di stop/3

Quando si interrompe la suddivisoine di un nodo?

- Se il miglior split del nodo non riduce in modo significativo (con una soglia predefinita).Ad esempio quando tutti i nodi ultimi hanno tutti la stessa modalità
- Se il numero di entità rimaste nel nodo è esigio
- Se il numero di nodi già ottenuti ha raggiunto un livello irragionevole di complessità
- Se l'errore complessivo dell'albero è stato ridotto al di sotto di una soglia prefissata

Assegnazione

Terminata la costruzione dell'albero dobbiamo assegnare una categoria, detta $\psi(t)$ ad ogni nodo terminale t .

In caso di classificazione,ad ogni entità che ricade nel nodo è attribuita la medesima categoria, in particolare quella che ha maggiore frequenza nel nodo terminale

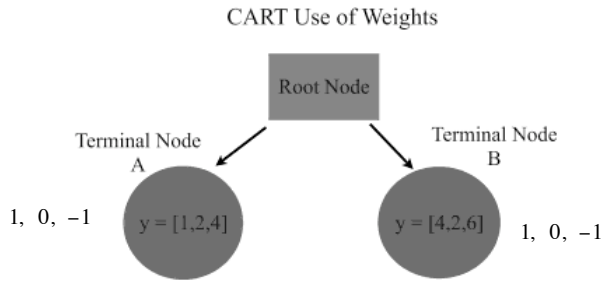
$$\psi(t) = \max_{1 \le j \le k} p(j|t)$$

Se il massimo è raggiunto da più di una categoria, quella effettiva di appartenenza sarà scelta casualmente tra quelle equivalenti.

Per un procedura di regressione, il valore tipico riscontrato nel nodo terminale potrebbe essere ottenuto con una media: aritmetica, potata, mediana, etc.

$$L(t) = \frac{\sum_{i \in t} y_{i,t}}{n_t}$$

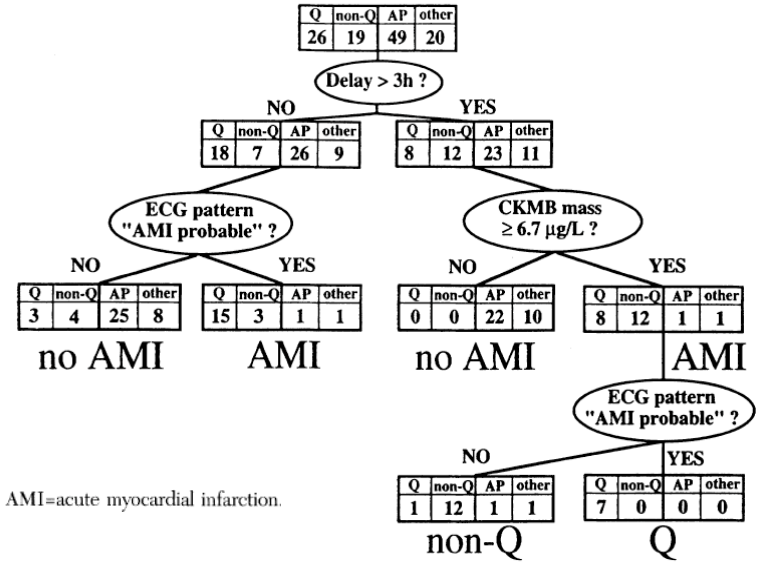
Assegnazione/2



$$y_A = 1\left(\frac{1}{6}\right)+0\left(\frac{2}{6}\right)-1\left(\frac{4}{6}\right)=-\frac{3}{6};\ y_B = 1\left(\frac{4}{12}\right)+0\left(\frac{2}{12}\right)-1\left(\frac{6}{12}\right)=-\frac{1}{6}$$

Se le variabili del data set sono di tipo metrico, allora il valore della variabile criterio da assegnare ad una unità nuova potrà basarsi sulla regressione lineare multipla applicata alle entità del nodo terminale (se queste sono sufficientemente numerose, ad esempio 5/6 volte il numero di regressori.

$$E(y|X) = \beta_0 + \sum_{j=1}^{m}\beta_j X_{i,j};\quad i \in t,\ n_t \geq 5m$$

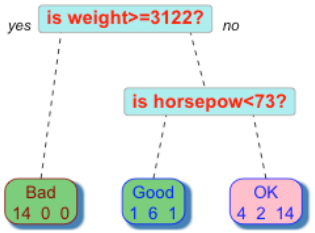


AMI=acute myocardial infarction.

FIGURE 2. Binary tree structured classifier for the diagnosis of acute myocardial infarction in nontraumatic chest pain patients obtained by classification and regression trees. For definition of the ECG pattern "probable AMI" see the Methods section. AP=angina pectoris; AMI=acute myocardial infarction.

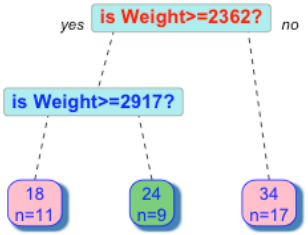
Esempio

predict Miles per Gallon for a set of cars



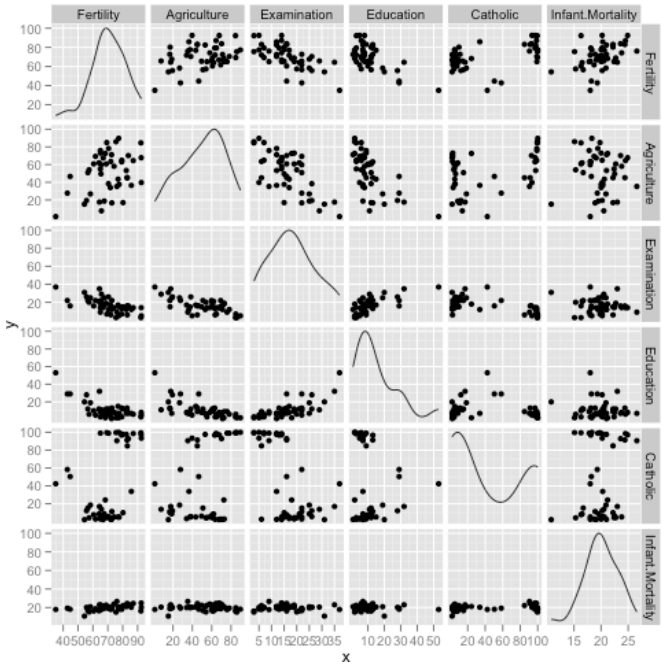
Classificazione

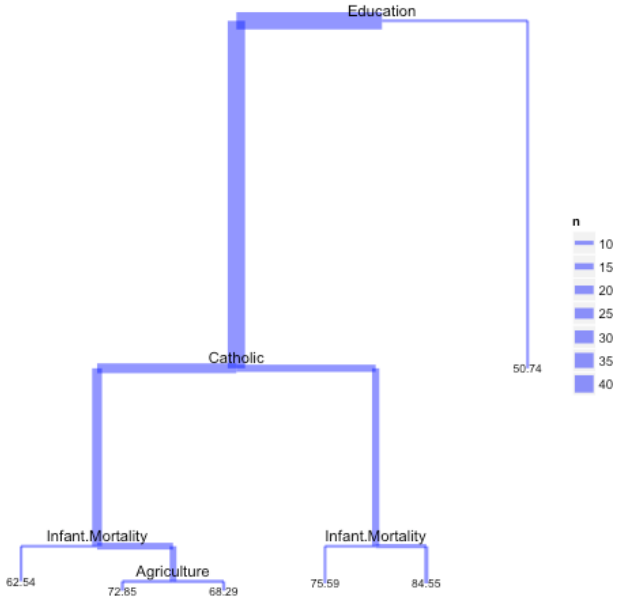
predict Miles per Gallon for a set of cars



Regressione

Swiss data set





Probabilità di errata classificazione

Supponiamo di aver terminato l'albero assegnando ad ogni nodo terminale la sua categoria. Qual'è l'errore di classicazione che si commette?

Dobbiamo introdurre il concetto di stima di risostituzione r(t) per la probabilità di sbagliare la categoria di una entità, dato che questa ricade nel nodo t-esimo

$$r(t) = 1 - \max_{1 \le j \le k} p(j|t) = 1 - p\left[\psi(t)|t\right]$$

Definiamo

$$R(t) = r(t)p(t) = p\left[\overline{\psi(t)}|t\right]p(t)$$

La stima di risostituzione per l'errore di classificazione globale dell'albero T è

$$R(T) = \sum_{t \in T} R(t) = \sum_{t \in T} r(t)p(t)$$

dove t è un nodo terminale dell'albero

Particolarità della stima di risostituzione

Nel dividere un nodo in due subnodi, l'errore di classificazione si riduce sempre.

Se tale errore è stimato con la risostituzione si può accertare che maggiore è il numero di split che si effettuanopiù e meno errori si commettono poi nella classificazione finale.

Qualunque sia lo split del nodo t nei subnodi t_L e t_R si ha

$$R(t) \ge R(t_L) + R(t_R)$$

Indichiamo con j* la classe di appartenenza del nodo t e consideriamo la probabilità condizionale di trovare la classe j* dato che siamo nel nodo t

$$p(j^*|t) = p(j^*, t_L|t) + p(j^*, t_R|t) = p(j^*|t_L)p(t_L|t) + p(j^*|t_R)p(t_R|t)$$
$$= p_L p(j^*|t_L) + p_R p(j^*|t_R) \le p_L \max_{1 \le j \le k} p(j|t_L) + p_R \max_{1 \le j \le k} p(j|t_R)$$

Particolarità della stima di risostituzione/2

Quindi

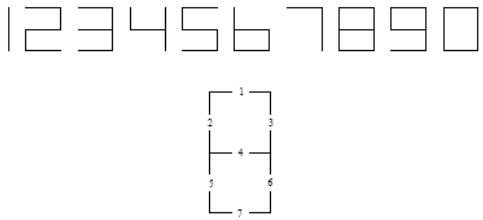
$$r(t) = 1 - p(j^*|t) \ge 1 - \left[p_L \max_{1 \le j \le k} p(j|t_L) + p_R \max_{1 \le j \le k} p(j|t_R)\right]$$
$$\ge p_L\left[1 - \max_{1 \le j \le k} p(j|t_L)\right] + p_R\left[1 - \max_{1 \le j \le k} p(j|t_R)\right]$$
$$\ge p_L r(t_L) + p_R r(t_R)$$

e finalmente

$$R(t) = p(t)r(t) \ge p(t)p_L r(t_L) + p(t)p_R r(t_R)$$
$$= p(t_L)r(t_L) + p(t_R)r(t_R)$$
$$= R(t_L) + R(t_R)$$

Example: Digit Recognition (CART)

Le cifre 0-9 su di un display possono essere visualizzate in base ad una matrice di 7 linee orizzontali e verticali.

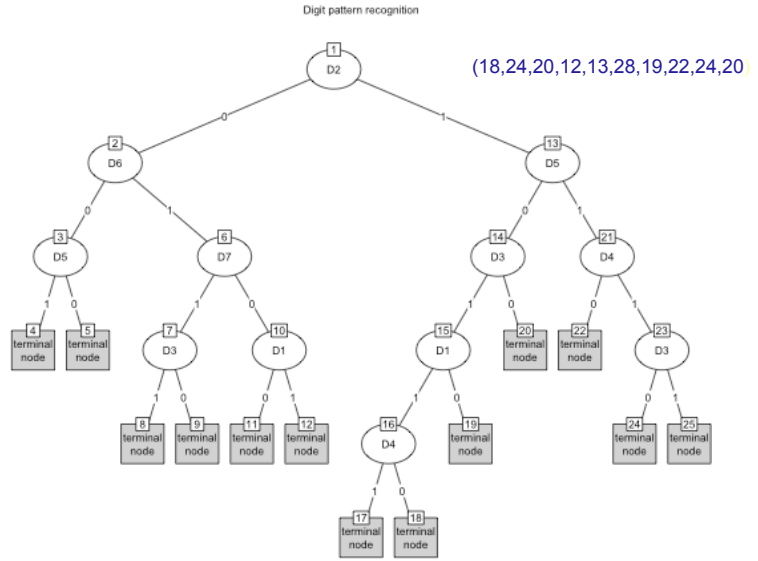


Ognuna di queste può essere presente o assente nella particolare cifra per cui la stessa può essere espressa da una configurazione di 0 e di 1 (luce on oppure off)

Digit	x_1	x_2	x_3	x_4	x_5	x_6	x_7
1	0	0	1	0	0	1	0
2	1	0	1	1	1	0	1
3	1	0	1	1	0	1	1
4	0	1	1	1	0	1	0
5	1	1	0	1	0	1	1
6	1	1	0	1	1	1	1
7	1	0	1	0	0	1	0
8	1	1	1	1	1	1	1
9	1	1	1	1	0	1	1
0	1	1	1	0	1	1	1

La sequenza corretta per le cifre è indicata nella matrice. Supponiamo che sussista una disfunzione che produce una probabilità del 10% di errore (luce on invece di off e luce off invece di on). Gli stati delle 7 linee sono indipendenti.

Il data set che serve per formare l'albero è formato da 200 rilevazioni.



Campione simulato da ogni cifra. I risultati differiscono dal testo di Breiman et al.

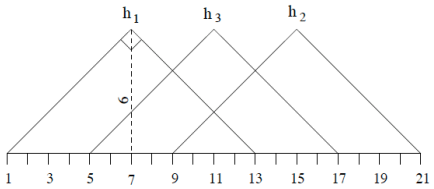
Alcune variabili sono adoperate più volte

Esempio: Waveform

Tre funzioni che definiscono dei triangoli equilateri centrati su punti diversi: 7, 11, 15.

ogni onda è caratterizzata da un vettore di m=13 dimensioni.

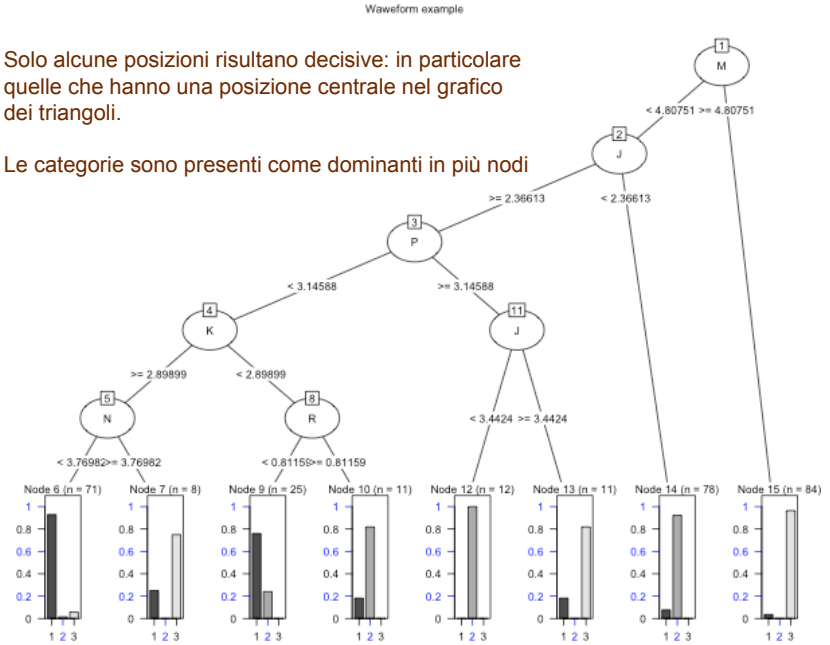
1. $y = x-1, y = 13-x,\ x \in 1:13,\ 0\ \ altrove$
2. $y = x-5, y = 17-x;\ x \in 5:17,\ 0\ \ altrove$
3. $y = x-9, y = 21-x; x \in 9:21,\ 0\ \ altrove$



Generiamo un data set di 100 entità per ognuno dei tre gruppi di onde così definite

1. $x_j = uh_1(j) + (1-u)h_2(j) + e_j;\ \ j = 1,2,\ldots,21$
2. $x_j = uh_1(j) + (1-u)h_3(j) + e_j;\ \ j = 1,2,\ldots,21$
3. $x_j = uh_2(j) + (1-u)h_3(j) + e_j;\ \ j = 1,2,\ldots,21$

Dove u proviene dalla uniforme (0,1) e e è una gaussiana standardizzata



Solo alcune posizioni risultano decisive: in particolare quelle che hanno una posizione centrale nel grafico dei triangoli.

Le categorie sono presenti come dominanti in più nodi

Ponderazione degli split (Cost)

vector of non-negative costs, one for each variable in the model. Defaults to one for all variables. These are scalings to be applied when considering splits, so the improvement on splitting on a variable is divided by its cost in deciding which split to choose.

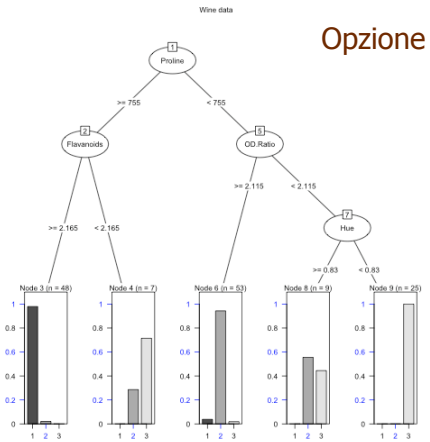
Cost=(1,1,1,1,1,1,1,1,1,1,1,1,1)

```
1) root 142 84 2 (0.34507042 0.40845070 0.24647887)
  2) Proline>=755 55  8 1 (0.85454545 0.05454545 0.09090909)
    4) Flavanoids>=2.165 48  1 1 (0.97916667 0.02083333 0.00000000) *
    5) Flavanoids< 2.165 7  2 3 (0.00000000 0.28571429 0.71428571) *
  3) Proline< 755 87 32 2 (0.02298851 0.63218391 0.34482759)
    6) OD.Ratio>=2.115 53  3 2 (0.03773585 0.94339623 0.01886792) *
    7) OD.Ratio< 2.115 34  5 3 (0.00000000 0.14705882 0.85294118)
     14) Hue>=0.83 9  4 2 (0.00000000 0.55555556 0.44444444) *
     15) Hue< 0.83 25  0 3 (0.00000000 0.00000000 1.00000000) *
```

Cost=(1,1,1,1,2,1,1,1,1,1,1,2)

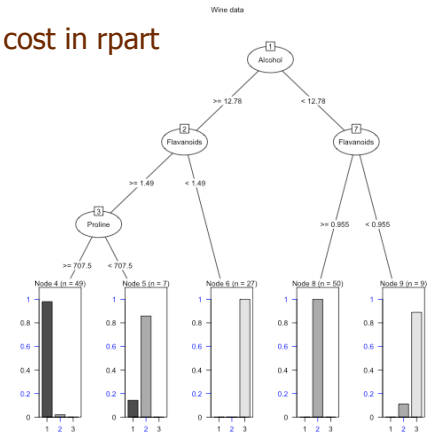
```
1) root 142 84 2 (0.34507042 0.40845070 0.24647887)
  2) Alcohol>=12.78 83 34 1 (0.59036145 0.08433735 0.32530120)
    4) Flavanoids>=1.49 56  7 1 (0.87500000 0.12500000 0.00000000)
      8) Proline>=707.5 49  1 1 (0.97959184 0.02040816 0.00000000) *
      9) Proline< 707.5 7  1 2 (0.14285714 0.85714286 0.00000000) *
    5) Flavanoids< 1.49 27  0 3 (0.00000000 0.00000000 1.00000000) *
  3) Alcohol< 12.78 59  8 2 (0.00000000 0.86440678 0.13559322)
    6) Flavanoids>=0.955 50  0 2 (0.00000000 1.00000000 0.00000000) *
    7) Flavanoids< 0.955 9  1 3 (0.00000000 0.11111111 0.88888889) *
```

Opzione cost in rpart




```
Digit.Tr4.pred      1      2      3
             1 0.33099 0.00704 0.00000
             2 0.01408 0.38732 0.03521
             3 0.00000 0.01408 0.21127
```

Pei uguali. Percentuale di successo nel campione training: 0.93




```
Digit.Tr4.pred      1      2      3
             1 0.33803 0.00704 0.00000
             2 0.00704 0.39437 0.00000
             3 0.00000 0.00704 0.24648
```

Magnesium e Proline peso doppio rispetto alle altre. Percentuale di successo nel campione training: 0.979

Criterio di split

Nel pacchetto rpart è limitato alle scelte: Gini (opzione di default) e Information cioè la misura di entropia.

In genere occorre esaminare entrambe le opzioni per poter effettuare una scelta più ragionata.

```
node), split, n, loss, yval, (yprob)
      * denotes terminal node
```

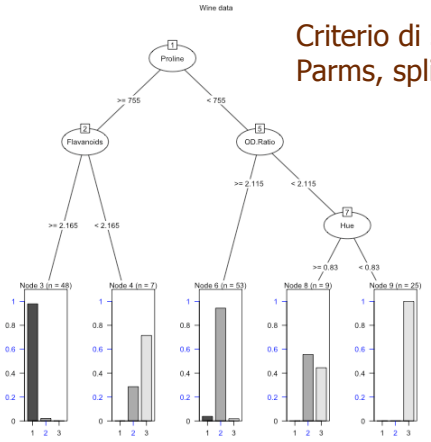
Information

```
1) root 142 84 2 (0.34507042 0.40845070 0.24647887)
  2) Flavanoids>=1.4 99 49 2 (0.49494949 0.50505051 0.00000000)
    4) Proline>=724.5 51  3 1 (0.94117647 0.05882353 0.00000000) *
    5) Proline< 724.5 48  1 2 (0.02083333 0.97916667 0.00000000) *
  3) Flavanoids< 1.4 43  8 3 (0.00000000 0.18604651 0.81395349)
    6) Color.intensity< 3.725 8  0 2 (0.00000000 1.00000000 0.00000000) *
    7) Color.intensity>=3.725 35  0 3 (0.00000000 0.00000000 1.00000000) *
```

Gini

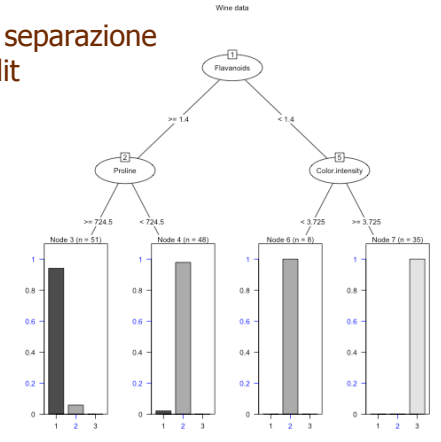
```
1) root 142 84 2 (0.34507042 0.40845070 0.24647887)
  2) Proline>=755 55  8 1 (0.85454545 0.05454545 0.09090909)
    4) Flavanoids>=2.165 48  1 1 (0.97916667 0.02083333 0.00000000) *
    5) Flavanoids< 2.165 7  2 3 (0.00000000 0.28571429 0.71428571) *
  3) Proline< 755 87 32 2 (0.02298851 0.63218391 0.34482759)
    6) OD.Ratio>=2.115 53  3 2 (0.03773585 0.94339623 0.01886792) *
    7) OD.Ratio< 2.115 34  5 3 (0.00000000 0.14705882 0.85294118)
     14) Hue>=0.83 9  4 2 (0.00000000 0.55555556 0.44444444) *
     15) Hue< 0.83 25  0 3 (0.00000000 0.00000000 1.00000000) *
```

Criterio di separazione Parms, split




```
Digit.Tr4.pred      1      2      3
             1 0.33099 0.00704 0.00000
             2 0.01408 0.38732 0.03521
             3 0.00000 0.01408 0.21127
```

Gini (e pesi uguali) Percentuale di successo nel campione training: 0.93




```
Digit.Tr4.pred      1      2      3
             1 0.33803 0.02113 0.00000
             2 0.00704 0.38732 0.00000
             3 0.00000 0.00000 0.24648
```

Information (e pesi uguali) Percentuale di successo nel campione training: 0.972

Scelta delle probabilità a priori

Esprime la la probabilità che una entità sconosciuta presente nel fenomeno, ma non neldata set abbia una modalità prefissata .

L'opzione di default è la frequenza relativa delle modalità nel data set di prova

$$\pi_j = \frac{entità\ \ in\ \ c_j}{entità\ \ totali} = \frac{n_j}{n}, \quad j = 1,2,\ldots,k$$

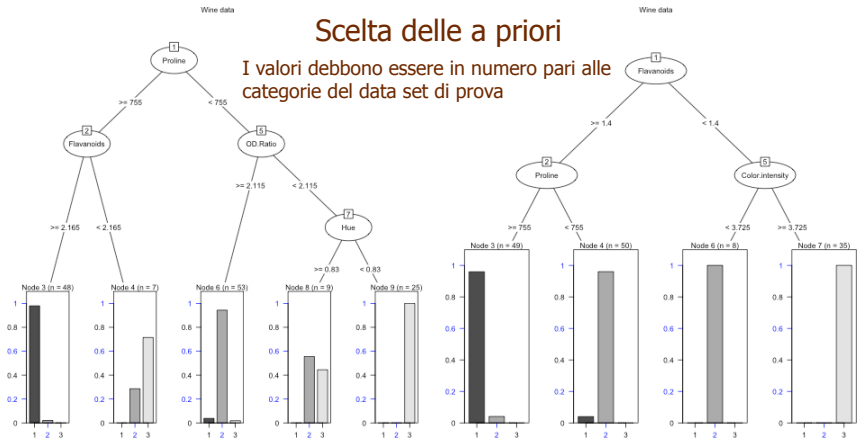
In alternative si possono scegliere delle probabilità (positive sommanti ad uno) che enfatizzino il ruolo di una classe di primaria importanza e comprimano quello di altre

Un'altra soluzione è di generare k numeri dalla uniforme unitaria e poi riscalarne i valori per renderli a somma uno.

Valutando un certo numero di volte la matrice di confusione nel campione di prova si sceglie la il vettore di k dimensioni che rende massima la classificazione corretta… Nell'auspicio che poi funzioni anche nel test e nei dati nuovi.

Scelta delle a priori

I valori debbono essere in numero pari alle categorie del data set di prova



Digit.Tr4.pred	1	2	3
1	0.33099	0.00704	0.00000
2	0.01408	0.38732	0.03521
3	0.00000	0.01408	0.21127

Gini, pesi uguali, priori=frequenze relative.
Percentuale di successo nel campione training: 0.93

Digit.Tr4.pred	1	2	3
1	0.3310	0.0141	0.0000
2	0.0141	0.3944	0.0000
3	0.0000	0.0000	0.2465

Gini, pesi uguali, Priori=0.13,0.45,0.42)
Percentuale di successo nel campione training: 0.993

Costi di errata classificazione diversi

Finora si è tacitamente ipotizzato che un'entità di classe j classificata per errore in una altra classe comportasse lo stesso costo o perdita di efficienza, a prescindere dalla catgegoria sbagliata.

Possiamo distinguere tali costi. Definiamo il costo di classificazione errata come

$$c(i|j) = \begin{cases} c_{ij} > 0 & se \quad i \neq j \\ 0 & se \quad i = j \end{cases}$$

Consideriamo il nodo t con probabilità p(j|t), j=1,2,…k. Se una entità di classe sconosciuta ricade in tale nodo ed è assegnata alla categoria i, allora il costo atteso di errata classificazione è

$$\sum_{j=1}^{k} c(i|j)\, p(j|t)$$

Possiamo stabilire una regola di assegnazione che tenga conto di tale costo e realizzi la scelta che lo renda minimo.

Costi di errata classificazione diversi/2

Si assegna l'entità sconosciuta alla classe i_0 se i_0 rende minimo il costo atteso di errata classificazione al nodo t, con un costo di risostituzione pari a

$$r(t) = \min_{1 \le i \le k} \sum_{j=1}^{k} c(i|j)\, p(j|t)$$

Supponiamo di aver completato l'albero assegnando ad ogni nodo terminale la sua categoria o valore. Qual'è l'errore di classicazione che si commette?

La stima di risostituzione per l'errore dell 'intero albero T è data da

$$R(T) = \sum_{t \in T} r(t)\, p(t)$$

La determinazione del miglior split dovrà terere conto in R(T) dei costi c(i,j)

Da notare che se i costi fossero tutti uguali si avrebbe:

$$\sum_{j=1}^{k} c(i|j)\, p(j|t) = 1 - p(i|t)$$

Matrice di perdita (loss matrix)

Indichiamo con L(i, j) la perdita di efficacia dovuta alla assegnazione di una entità alla classe j sebbene appartenente alla classe i.

Nel caso di modalità più rare, ma di grande impatto (ad esempio per accertare frodi) il falso negativo potrebbe costare molte volte di più del falso positivo.

La perdita attesa per l'errata classificazione è espressa da $L(i, j)p_ip_j$ che ci porta a definire un nuovo indice di eterogeneità

corrette/errate	1	2	…	*i*	…	*k*
1	0	c_{12}		c_{1i}		c_{1k}
2	c_{21}	0		c_{2i}		c_{2k}
…			…			
j	c_{j1}	c_{j2}		0		
…					…	
k	c_{k1}	c_{k2}				0

$$G(p) = \sum_i \sum_j L(i,j) p_i p_j$$

$$Gini = \sum_i \sum_j p_i p_j$$

$$G(p) \ \ per \ \ L(i,j) = 1$$

che però non è sempre usabile dato che simmetrizza comunque le perdite di qualità. Infatti

$$G(p) = (1/2) \sum \sum [L(i,j) + L(j,i)] \, p_i p_j$$

In effetti la classificazione con due catagorie rende inutile la matrice di perdita

Loss matrix/2

E' usata per dare un costo diverso all'errore di classificazione.

Il falso positivo ed il falso negativo non sempre sono equivalenti. Ad esempio, nelle analisi cliniche il falso positivo (giudichiamo sana una persona ammalata e non procediamo ad ulteriori verifiche) è più rischioso del falso positivo (giudichiamo ammalata una persona che è sana e procediamo ad altri test).

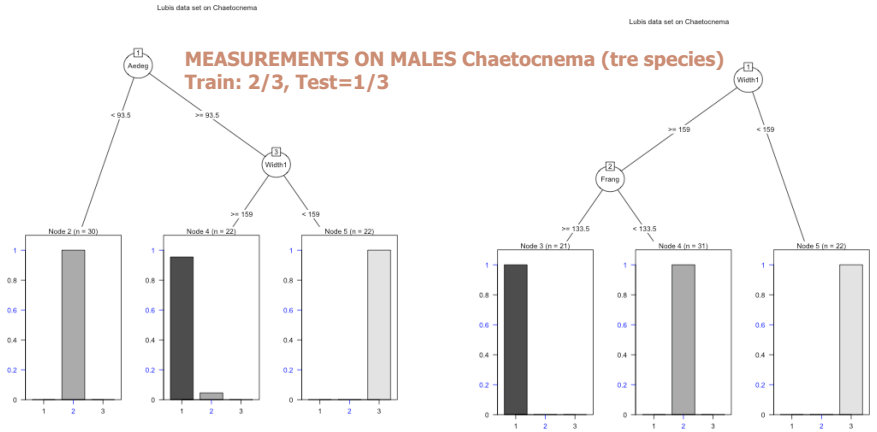
Allo stesso modo respingere ilcredito ad un cliente solvibile può essere meno dannoso che concederlo ad un altro che non sarà in grado di restituirlo

La costruzione dell'albero può tenere conto delle errate classificazioni dando una penalità diversa ad ogni classificazione errata per un dato split in ogni nodo

La loss matrix è molto utile in fase di addestramento del decisore alterando le probabilità a priori ed incidendo sulla scelta della variabili secondo cui splittare.

In particolare, quando esiste una classe molto dominante sulle altre

Lubishew data set 1



	[,1]	[,2]	[,3]
[1,]	0	1	3
[2,]	1	0	2
[3,]	3	2	0

In questo caso si è preferita una matrice simmetrica che ha permesso di eliminare gli errori nella seconda classe.

In generale, la loss matrix deve essere quadrata con dimensioni pari al numero di categorie e con diagonale nulla

Matrice di confusione

L'efficacia della procedura CART i può valutare comparando le informazioni accantonate nel data set di Test e quindi conosciute per certe rispetto alle previsioni/stime effettuate in base all'albero costruito sul data set di prova.

Casi predetti

Casi originali

	1	2	…	n
1	45	3	…	1
2	**0**	49	…	0
…	2	…	42	0
n	0	1	0	50

I numeri fuori diagonale sono la classificazione errata

I numeri sulla diagonale sono la classificazione corretta

Entropia quadratica di Rao

Se si sostituisce la matrice di perdita con una matrice delle distanze si ottiene il criterio dell'entropia quadratica

$$E_2(p) = \sum_i \sum_j d_{ij} p_i p_j$$

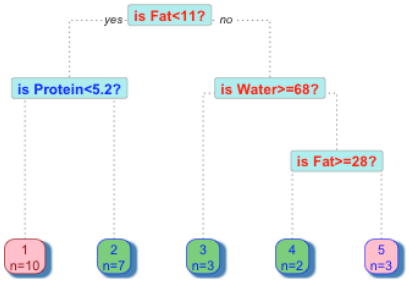
dove d_{ij} è il quadrato di una funzione di distanza che definisce una matrice euclidea delle distanze (ad esempio le metriche di Minkowski).

Se $d_{ij}=1$ per $i \neq j$ e $d_{ij}=0$ per $i=j$ allora $E_2(p)$=Gini

Il massimo di $E_2(p)$ si ottiene se le categorie più distanti hanno massima frequenza

L'entropia quadratica misura lo scarto atteso tra due entità scelte a caso nel data set.

Composizione latte dei mammiferi



La potatura (pruning)

Interrompere i frazionamenti successivi non è semplice dato che riducono, sia pure di poco e in ragione decrescnte, l'errore di classificazione

Alberi con molte foglie rischiano l'overfitting cioè adattano uno schema di interrogazioni troppo profondo e complesso per essere interpretabile in contesti non professionali.

Alberi con poche foglie possono mancare o sottovalutare aspetti importanti delle entità da classificare.

Questi problemi si possono affrontare con la potatura o pruning: si costruisce un albero molto frondoso per poi sfoltirlo, con regole opportune, dalle ramificazioni meno interessanti e meno utili.

Minbucket/Minsplit

In una prima fase si costruisce l'albero T_{max} senza alcun freno e si procede fin tanto che si sia rimasti con dei nodi che siano perfettamente omogenei oppure contengono un numero di entità inferiore alla soglia stabilita

$$N(t) \leq N_{\min}$$

Nel lessico del CART, N_{min} è detto minbucket ed esprime il numero minimo di entità che devono essere presenti perché un nodo possa essere considerato terminale.

Il parametro è appaiato a minsplit cioè il numero minimo di entità che debbono essere presenti nel nodo perché si posso tentare uno split. Se uno dei due nonè specificato, il software può ricavarlo dall'altro con le formule:

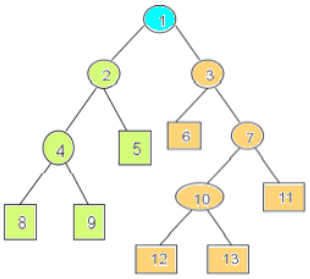
$$minsplit = 3minbucket; \quad minbucket = \frac{minsplit}{3}$$

Valori accettabili per N_{min} sono 5, 3 ed ovviamente N_{min} =1.

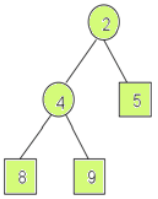
Successivamente I rami meno produttivi sono potati per ottenere un equilibrio accettabile tra il tasso di errore e il numero di nodi terminali (che potrebbero essere comuqnue troppi nostante i limiti imposti)

Taglio dei rami

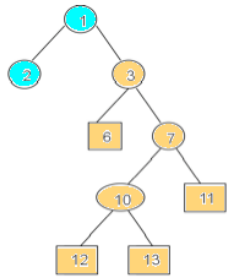
Se non sappiamo dove fermarci tanto vale continuare fino in fondo. Costruiamo un albero il più grande possibile (entro certi limiti) e poi tagliamo i rami secchi



Albero T_{max}



Ramo T_2



Albero potato T_{max}-T_2

La potatura da un albero T di un ramo T_t con nodo radice in t consiste nella cancellazione di tutti i nodi discendenti da t tranne il nodo radice t.

La potatura è necessaria per ridurre l'aleatorietà nella classificazione delle nuove entità ancora sconosiute.

Taglio dei rami/2

L'albero massimale –anche se costruito con prudenza- ha sempre un numero notevole di nodi (30/40 nodi non sarebbe un numero sorprendente)

Qual'è il ramo che cade per primo?

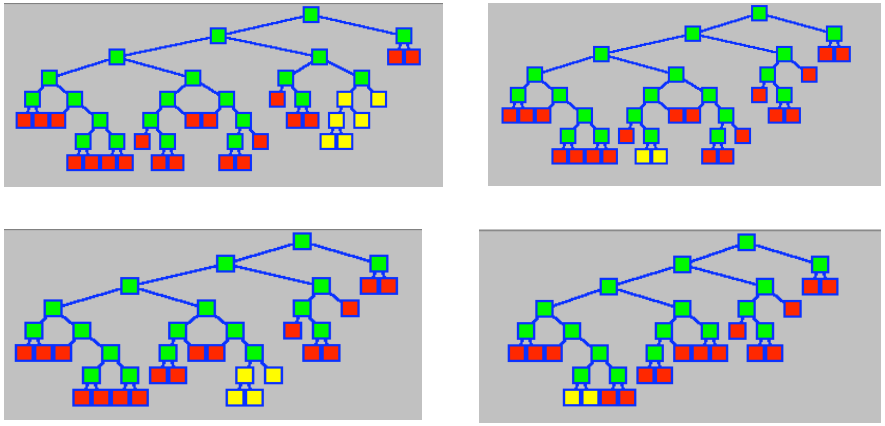
Si taglia il ramo che aggiunge meno alla qualità complessiva dell'albero, ad esempio quello che ha associato il costo di errata classificazione più alto.

La potatura selettiva parte allora dal calcolo di R(t) per ogni nodo $t \in T_{\max}$ avviando il taglio dal basso in modo che alla fine dei tagli, R(T) relativo all'albero che rimane, sia minimo.

Se più di un nodo si trova nelle stesse condizioni si tagliano entrambi.

E' possibile che l'intero albero sia tagliato alla radice e questo è un risultato desiderabile se la variabile target non è prevebile in base alle altre variabili del data set.

Taglio dei rami/3



Se le potature fossero annidate forse questa sarebbe una procedura ammissibile e cioè la migliore rescissione T_{k+1} non è necessariamente all'internto di T_k

Costo di complessità

Si minimizza una misura dell'albero che tiene conto sia dell'errore di classificazione che della complessità dello stesso:

$$R_{\alpha}(T) = R(T) + \alpha N(T)$$

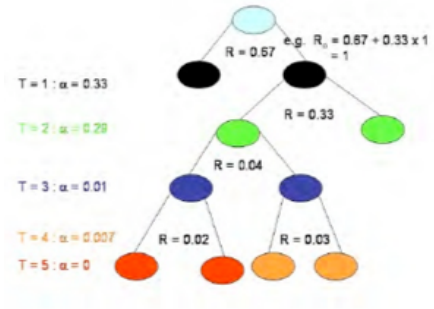
L'errore di classificazione $R(T)$ per un certo albero T corrisponde alla percentuale di osservazioni classificate erroneamente in ogni nodo terminale.

$N(T)$ indica il numero di foglie o nodi terminali, mentre $\alpha>0$ è un coefficiente che penalizza in ragione della complessità dell'albero, misurata dal numero di nodi terminali:

ogni nodo in più implica un incremento α nella complessità

$R_{\alpha}(T)$ è una combinazione lineare di errore e complessità. A questo si cerca, per ogni α fissato,l'albero con errore totar $T(\alpha) \leq T_{max}$ che rende minima la quantità

$$R_{\alpha}[T(\alpha)] = \underset{T \leq T_{\max}}{Min} \; R_{\alpha}(T)$$



Costo di Complessità/2

Ad ogni split si ottiene una stima del costo di risostituzione R(t) per il nodo t-esimo e questo diminuisce ad ogni split. Nel nodo terminale α è zero. Aumenta se risaliamo verso la radice primaria.

All'aumentare di α il peso della complessità aumenta per bilanciare la riduzione di errore dovuta alla minore presenza di nodi di decisione (è quei che si può sbagliare)

Se α è piccolo è bassa la penalità per avere molti nodi terminali cosicché $T(\alpha)$ sarà grande.

Se T_{max} fosse così vasto da formare un nodo terminale per ogni entità (classificazione perfetta) allora $R(T_{max})=0$ e quindi T_{max} minimizza $R_0(T)$.

All'aumentare di α l'albero migliore $T(\alpha)$ contiene sempre meno nodi terminali al punto che per α grande l'albero migliore consisterà solo della radice primaria.

Costo di complessità/3

Definiamo con $R_\alpha(T) = R(T) + \alpha N(T)$ il costo di complessità associato con l'albero T_α

$$Se\ \ T_1, T_2 \in T \ \ e \ \ R_\alpha(T_1) = R_\alpha(T_2) \Rightarrow (T_1 \subset T_2) \cup (T_2 \subset T_1)$$
$$\Rightarrow [N(T_1) < N(T_2)] \cup [N(T_2) < N(T_1)]$$

$$Se\ \ \alpha > \beta \Rightarrow (T_\alpha = T_\beta) \cup (T_\alpha \subset T_\beta)$$

La prima relazione implica che sia possibile definire univocamente l'albero T_α che minimizza $R_\alpha(T)$

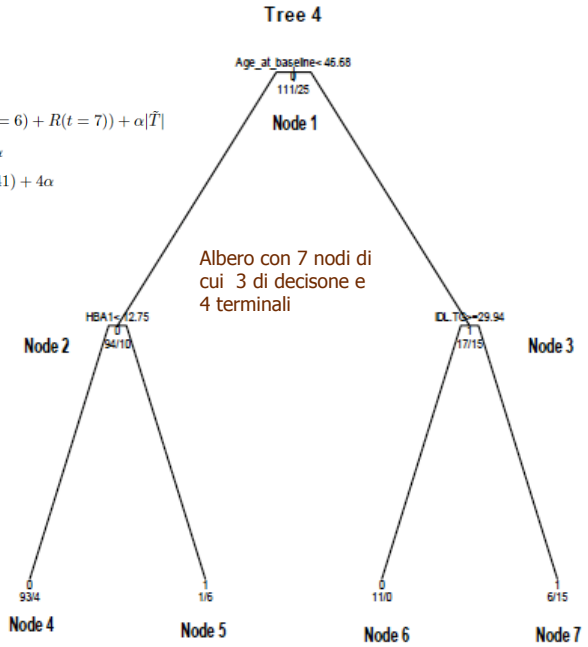
Poiché ogni successione di alberi annidati di T ha, al massimo, N(T) nodi terminali, allora la seconda relazione implica che i valori possibili di α possono essere raggruppati in m<N(T) intervalli

$$I_1 = [0, \alpha_1],\ I_2 = [\alpha_1, \alpha_2],\ I_3 = [\alpha_2, \alpha_3],\ \ldots,\ I_m = [\alpha_{m-1}, \infty]$$

E tutti gli $\alpha \in I_j$ condivono lo stesso albero che minimizza il costo di complessità

Esempio

$$\begin{aligned} R_\alpha(T) &= (R(t=4) + R(t=5) + R(t=6) + R(t=7)) + \alpha|\tilde{T}| \\ &= (\frac{16}{136} + \frac{1}{136} + 0 + \frac{6}{136}) + 4\alpha \\ &= (0.1176 + 0.0074 + 0 + 0.0441) + 4\alpha \\ &= 0.1691 + 4\alpha \end{aligned}$$



Albero con 7 nodi di cui 3 di decisone e 4 terminali

Esempio (continua)

The question is, what happens if an internal node becomes a terminal node? What is the consequence of pruning off all off spring nodes of an internal node?

For instance, if we cut the offspring nodes off the root node, we have the root-node tree whose cost-complexity is 0.7353+α

For it to have the same cost-complexity as the initial 7 tree, we need

$$0.1691 + 4\,\alpha = 0.7353 + \alpha$$

giving α = 0.1887.

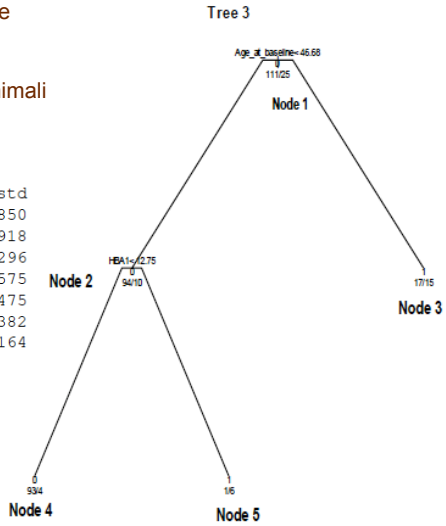
Node(t)	R(t)	R(T)	T		α
7	6/136	6/136			
6	0	0			
5	1/136	1/136			
4	16/136	16/136			
3	17/136	6/136	2	$0.0441 + 2\alpha = 0.125 + \alpha$	**0.081** (α_3)
2	40/136	17/136	2	$0.125 + 2\alpha = 0.2941 + \alpha$	0.1691
1	100/136	23/136	4	$0.1691 + 4\alpha = 0.7353 + \alpha$	0.1887

Esempio (potatura)

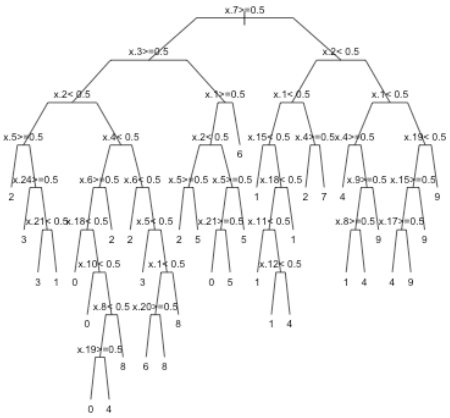
Il nodo inerno 3 diventa un nodo terminale con complessità 0.081.

Se procediamo oltre e per sottoalberi minimali annidati si determina una successione di possibili scelte guidate dalla tabella

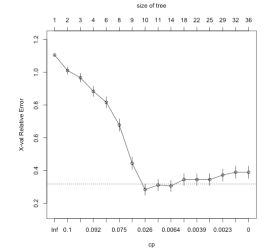
```
      CP nsplit rel error xerror      xstd
1 4.3e-01      0      1.00   1.00 0.1806850
2 2.3e-01      1      0.57   0.71 0.1333918
3 1.1e-01      2      0.34   0.66 0.1288296
4 7.5e-02      3      0.23   0.54 0.1185575
5 4.0e-02      5      0.08   0.49 0.1128475
6 1.0e-02      6      0.04   0.62 0.1287382
7 1.0e-06     10      0.00   0.70 0.1421164
```



Esempio: Digit pattern recognition



	CP	nsplit	rel error	xerror	xstd
1	0.1055556	0	1.00000	1.10556	0.0055416
2	0.1000000	1	0.89444	1.01111	0.0224846
3	0.0944444	2	0.79444	0.96667	0.0264225
4	0.0888889	3	0.70000	0.88333	0.0317178
5	0.0777778	5	0.52222	0.81667	0.0346744
6	0.0722222	7	0.36667	0.67778	0.0383213
7	0.0611111	8	0.29444	0.44444	0.0384900
8	0.0111111	9	0.23333	0.28333	0.0342445
9	0.0074074	10	0.22222	0.31111	0.0352767
10	0.0055556	13	0.20000	0.30556	0.0350815
11	0.0041667	17	0.17778	0.34444	0.0363369
12	0.0037037	21	0.16111	0.34444	0.0363369
13	0.0027778	24	0.15000	0.34444	0.0363369
14	0.0018519	28	0.13889	0.37222	0.0370831
15	0.0013889	31	0.13333	0.38889	0.0374743
16	0.0000000	35	0.12778	0.38889	0.0374743

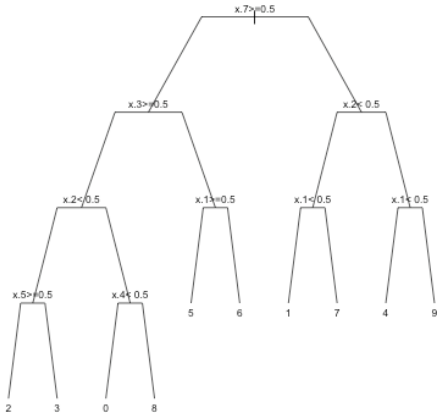


L'albero è stato ottenuto senza freni sulla complessità (cp=0)

$$cp = \frac{\alpha}{\sum_{i=1}^{n}\left(y_{t,i} - \bar{y}_t\right)^2}$$

xval è il numero di ricampionamenti del data set di prova

Esempio (continua)



	CP	nsplit	rel error	xerror	xstd
1	0.105556	0	1.00000	1.09444	0.0095501
2	0.100000	1	0.89444	1.02778	0.0206940
3	0.094444	2	0.79444	0.96667	0.0264225
4	0.088889	3	0.70000	0.90556	0.0305075
5	0.077778	5	0.52222	0.74444	0.0369434
6	0.072222	7	0.36667	0.62778	0.0389503
7	0.061111	8	0.29444	0.43333	0.0383213
8	0.025000	9	0.23333	0.30556	0.0350815

L'albero ottimizzato con cp=0.025 ha 10 nodi terminali (9 split)

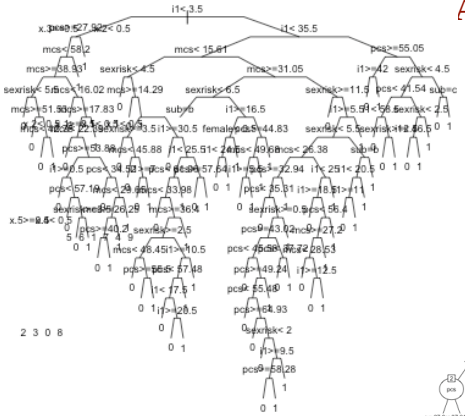
Nsplit= Numero di nodi terminali

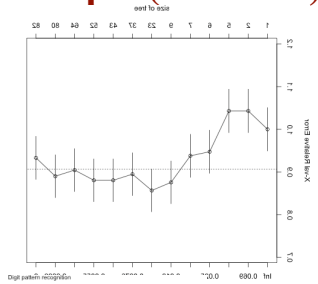
Rel.error= $F = \frac{\sum_{t\in T}\sum_{i=1}^{n}\left(y_{t,i} - \bar{y}_t\right)^2}{\sum_{t\in T}\sum_{i=1}^{n}\left(y_{t,i} - \bar{y}\right)^2}$

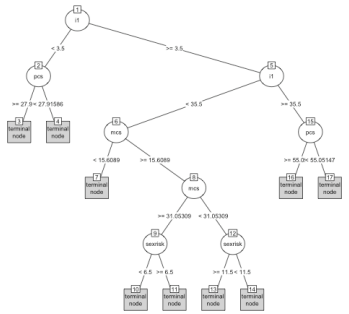
Xerror= analogo a rel.error ma basato su una diversa cross validation.

Xstd= deviazione standard per gli splits

Altro esempio (homeless)







	CP	nsplit	rel error	xerror	xstd
1	0.095694	0	1.00000	1.00000	0.050766
2	0.049442	1	0.90431	1.04785	0.050890
3	0.033493	4	0.75598	1.00000	0.050766
4	0.019139	5	0.72249	0.85646	0.049786
5	0.014354	6	0.70335	0.86603	0.049880
6	0.010000	8	0.67464	0.84689	0.049687

Altre opzioni

Il comando rpart prevede altri parametri che perfezionano la costruzione dell'albero

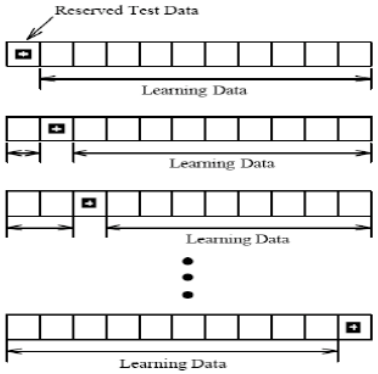
maxdepth. Limita la profondità dell'albero contando 0 il nodo radice. In pratica s tiene conto del numero massimo di nodi decisionali s che si incotrano percorrendo l'albero dall'inizio alla fine.

Meglio non superare 30.

maxcompete. Riduce il numero di split alternativi da riportare nell'output finale per ogni singolo nodo

Cross validation (xval)

Se xval=v allora il data set di prova è frazionato, con una selezione casuale, in v gruppi distinti L_1, L_2,...,L_v di numerosità approssimativamente uguale.



La procedura è applicata escludendo il gruppo v-esimo.

$$L^{(v)} = L - L_v, \; v = 1,\ldots,v$$

Il nuovo data set di prova contiene quindi una frazione (v-1)/v delle entità già incluse nel data set di prova. Se v=10 allora $L^{(v)}$ contiene il 90% dei dati di prova.

Per ogni valore del costo di complessità (α) si determina $T^{(v)}(\alpha)$ e cioè l'albero che minimizza il costo totale e per ogni v si valutano gli errori di classificazione testando l'esito finale sul segmento non incluso.

In questo modo si ottiene la stima degli errori di classificazione utile per determinare la sequenza di split più efficace.

Esempio

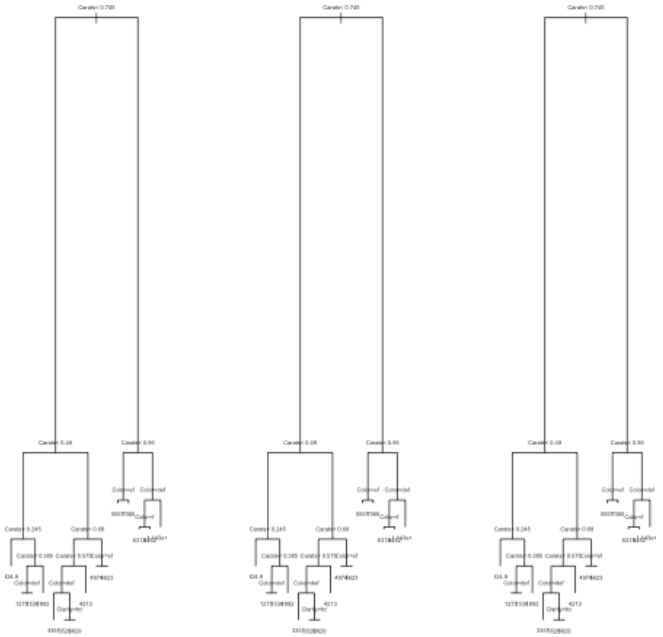
Diamonds data set

Xval=10

Xval=20

Xval=30

Nessun effetto

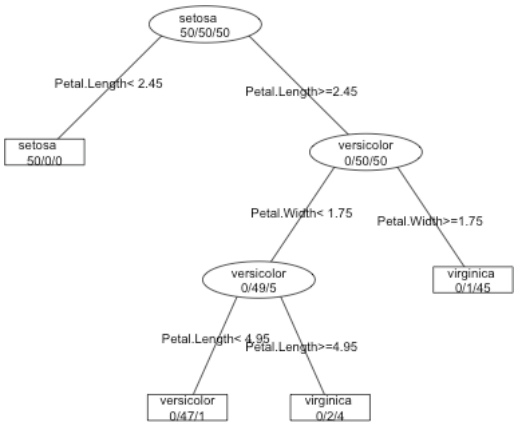


Esempio: rpartXse

This function is based on the tree-based framework provided by the rpart package (Therneau et. al. 2010).

It basically, integrates the tree growth and tree post-pruning in a single function call.

The post-pruning phase is essentially the 1-SE rule described in the CART book (Breiman et. al. 1984).



surrogate e valori mancanti

Ad ogni dato nodo è possibile tenere conto tanto del miglior split rispetto ad una variabile, ma anche di altri split rispetto alla stessa o ad altre variabili disposti in ordine decrescente di qualità.

In caso di variabili soggette a contaminazioni o difficili da reperire, la nuove entità potrebbero non riportare il valore della variabile target.

Qui si possono usare gli split ausiliari, stabiliti in una prima disamina a massima complessità dell'albero, per far passare oltre una entità lungo l'albero.

Nel caso l'entità manchi dei valori per tutti gli split ausiliari, l'entità non è classificata ovvero è inserita nella classe più numerosa e tutte le entitcon valore mancante nella variabile target sono considerate con la stessa classe.

Queste sono però scelte inefficienti perché le ragioni di una lacuna nei dati possono essere tante: un reddito può non essere dichiarato perché molto alto o perché molto basso.

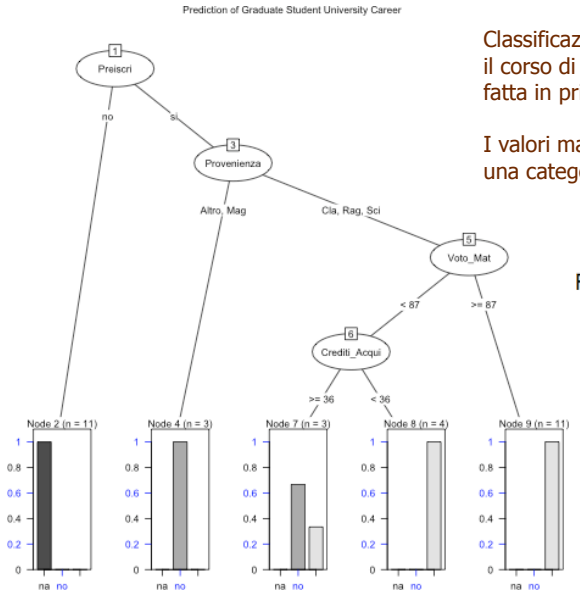
Parametri in rpart

Maxsurrogate
The number of surrogate splits retained in the output. If this is set to zero the compute time will be reduced, since approximately half of the computational time (other than setup) is used in the search for surrogate splits.

Usesurrogate.
how to use surrogates in the splitting process.
0 means display only; an observation with a missing value for the primary split rule is not sent further down the tree.
1 means use surrogates, in order, to split subjects missing the primary variable; if all surrogates are missing the observation is not split.
For value 2 , if all surrogates are missing, then send the observation in the majority direction.

Surrogatestyle
controls the selection of a best surrogate.
If set to 0 (default) the program uses the total number of correct classification for a potential surrogate variable;
if set to 1 it uses the percent correct, calculated over the non-missing values of the surrogate.
The first option more severely penalizes covariates with a large number of missing values.

Esempio



Classificazione della corrispondenza tra il corso di laurea di iscrizione e la scelta fatta in prima istanza.

I valori mancanti sono stati considerati una categoria terza

FG.Train.pred	na	no	si
na	0.556	0.000	0.000
no	0.000	0.111	0.000
si	0.000	0.000	0.333

Criticità del CART

Insignificant modification of the learning sample, such as eliminating several observations, could lead to radical changes in the decision tree.

As well, an increase or decrease in the ratio of misclassification costs may change the splitting variables.

I vantaggi del CART

La classificazione finale ha una forma semplice e di facile interpretazione

Le variabili possono essere sia qualitative che quantitative nello stesso data set

Si possono trattare set di dati molto ampi e con un numero elevato di variabili.

I risultati sono invarianti per trasformazioni monotone delle variabili (logaritmi, elevamento a potenza positiva, trasformazioni lineari)

E' possibile trattare anche osservazioni con dati mancanti

La tecnica è non parametrica per cui prescinde alla invadente ed invasiva ipotesi di gaussianità nelle variabili

No richiede l'indipendenza delle variabili e può gestire variabili correlate o associate

E' robusto rispetto ai valori remoti ed alla contaminazione nei dati