Alberi di Regressione

Caso di studio di Metodi Avanzati di Programmazione

AA 2015-2016

Corso A

Data Mining

Lo scopo del data mining è l'*estrazione* (semi) automatica di *conoscenza* nascosta in voluminose basi di dati al fine di renderla disponibile e direttamente utilizzabile

Aree di Applicazione

1. **previsione**

   utilizzo di valori noti per la previsione di quantità non note (es. stima del fatturato di un punto vendita sulla base delle sue caratteristiche)

2. **classificazione**

   individuazione delle caratteristiche che indicano a quale gruppo un certo caso appartiene (es. discriminazione tra comportamenti ordinari e fraudolenti)

3. **Regressione**

   Predizione del valore di un attributo numerico associato a un esempio sulla base di valori osservati per altri attributi dell'esempio medesimo

3. **segmentazione**

   individuazione di gruppi con elementi omogenei all'interno del gruppo e diversi da gruppo a gruppo (es. individuazione di gruppi di consumatori con comportamenti simili)

4. **associazione**

   individuazione di elementi che compaiono spesso assieme in un determinato evento (es. prodotti che frequentemente entrano nello stesso carrello della spesa)

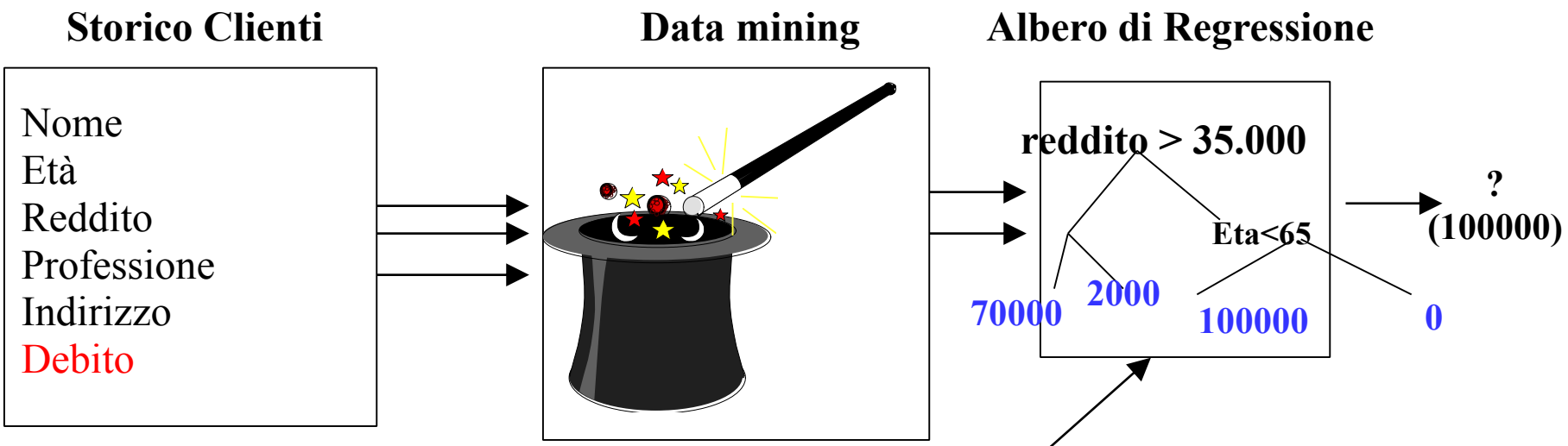
5. **sequenze**

   individuazione di una cronologia di associazioni (es. percorsi di visita di un sito web)

   …

Regressione

Considerando dati storici relativi a passati clienti e pagamenti, predire l'ammontare del debito del cliente con la banca



Dati di un nuovo cliente: Paolo Rossi, 35,37.000,architetto,Bari, ?

Regressione

- Apprendimento induttivo da esempi per imparare la definizione di una funzione di regressione
- Gli esempi usati per l'apprendimento sono descritti come vettori di coppie attributo-valore per i quali è nota l'attributo classe (target)
- Nella regressione l'attributo target è numerico

Regressione : Alberi di Regressione

Le funzioni di regressione sono apprese in forma di albero dove:

- ogni nodo interno rappresenta una variabile,
- un arco verso un nodo figlio rappresenta un possibile valore per quella proprietà, e
- una foglia il valore predetto per la classe a partire dai valori delle altre proprietà, che nell'albero è rappresentato del cammino (*path*) dalla nodo radice (*root*) al nodo foglia.

Un albero di regressione viene costruito utilizzando tecniche di apprendimento a partire dall'insieme dei dati iniziali (*training set*) per i quali è nota la classe

Induzione di Alberi di decisione

Input

Input: una collezione di esempi di apprendimento (training set), ciascun esempio è una tupla di valori per un prefissato insieme di attributi (variabili indipendenti)

$A = \{A_1, A_2, \ldots, A_m\}$

e un attributo di classe numerico (variabile dipendente/target). L'attributo A_i è descritto come continuo o discreto a seconda che i sui valori siano numerici o nominali.

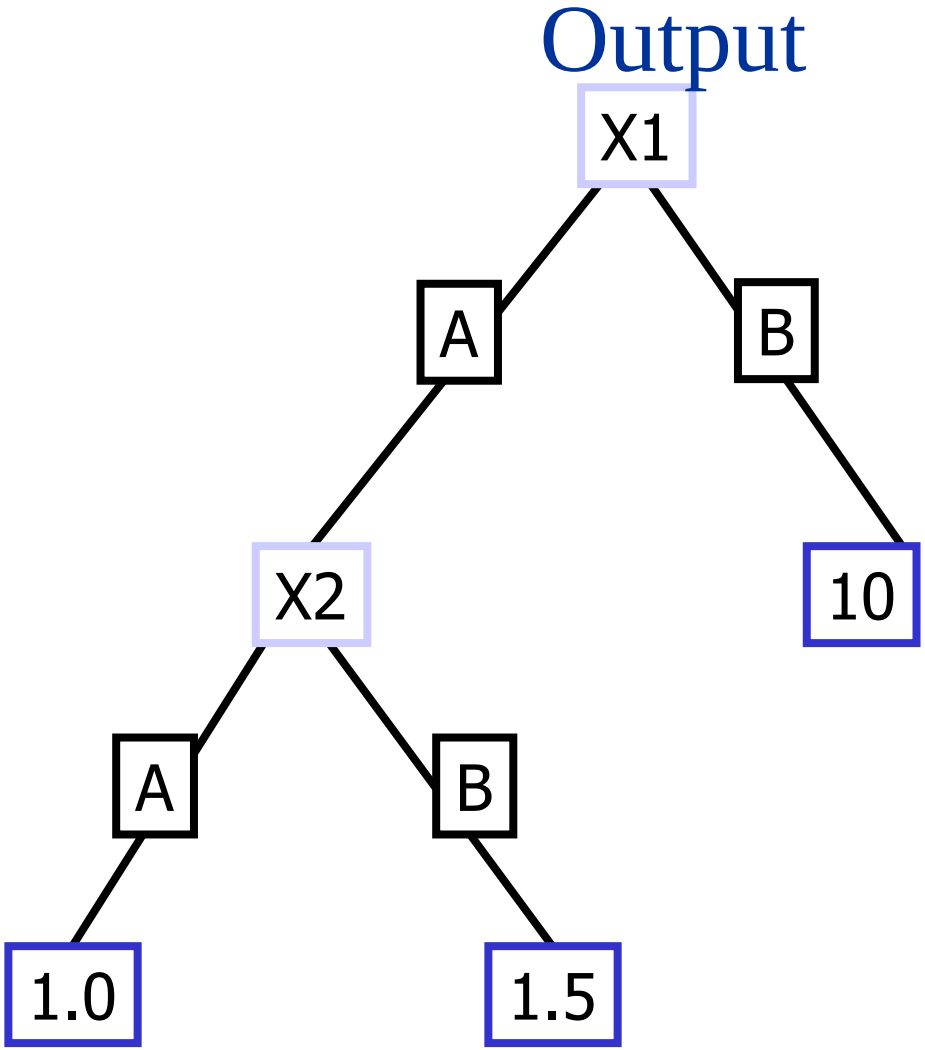
L'attributo di classe C è numerico e ha valori nell'insieme dei numeri reali

Induzione di Alberi di Regressione

Input

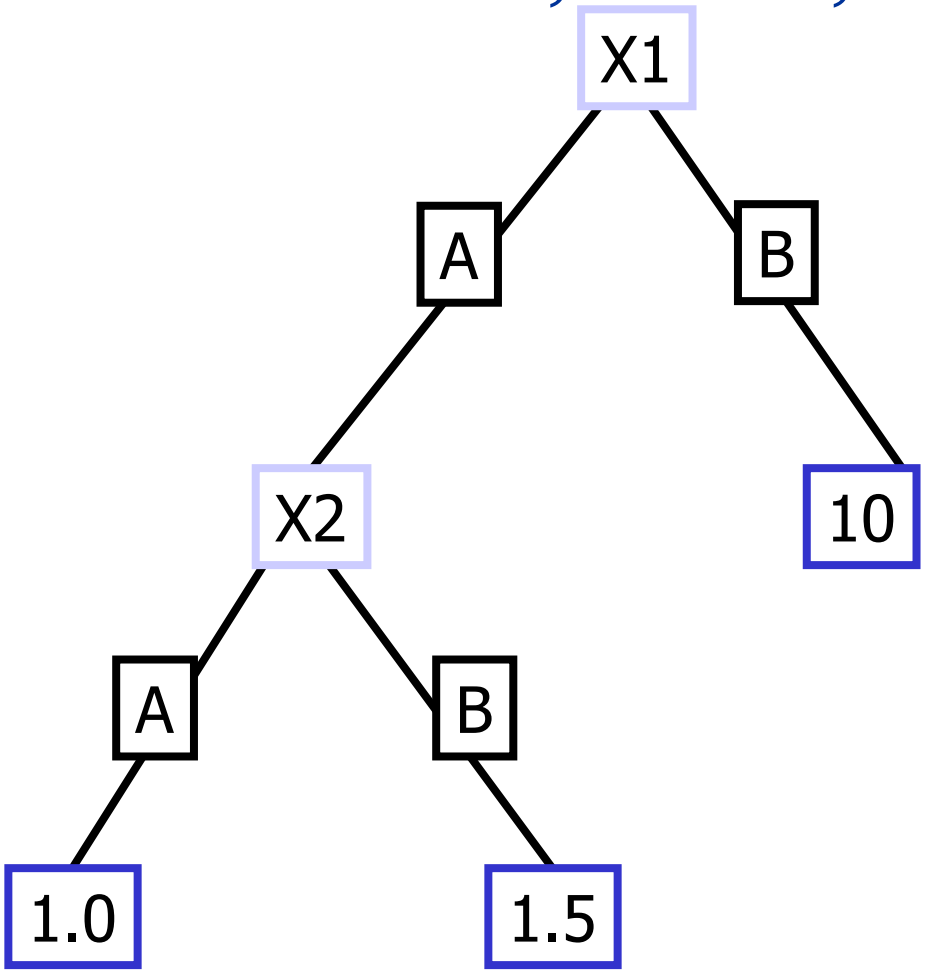
X1	X2	Y
A	A	1
A	A	1
A	A	1
A	A	1
A	B	1,5
A	B	1,5
A	B	1,5
B	B	10
A	B	1,5
A	B	1,5
B	C	10
B	B	10
B	C	10
B	C	10
A	A	1

Induzione di Alberi di Regressione



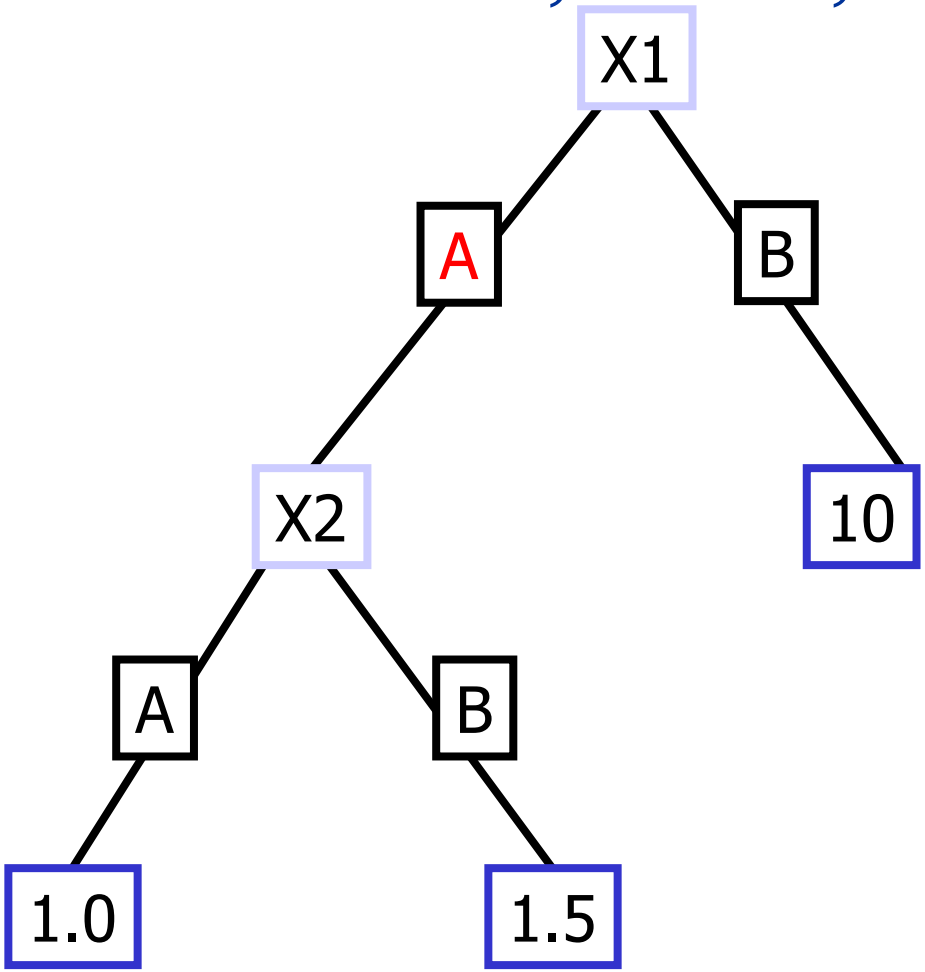
Alberi di Regressione: come usarli?

X1=A, X2=B, Y=?



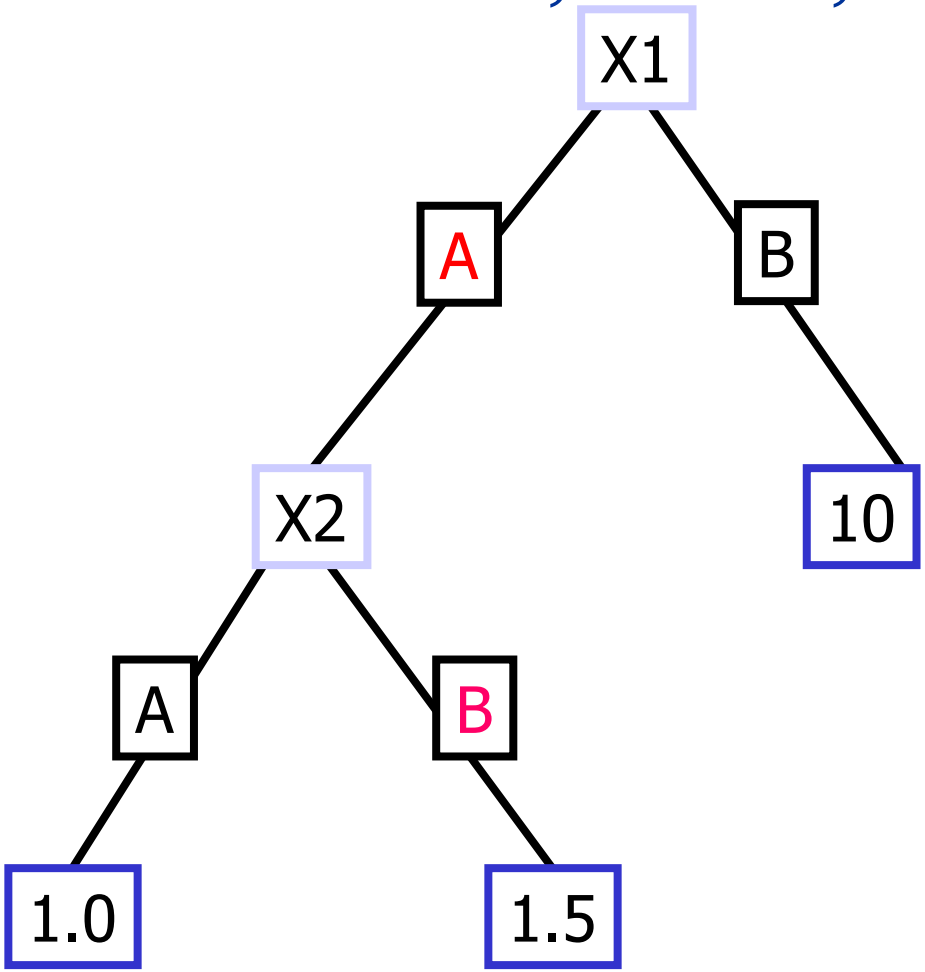
Alberi di Regressione: come usarli?

X1=A, X2=B, Y=?



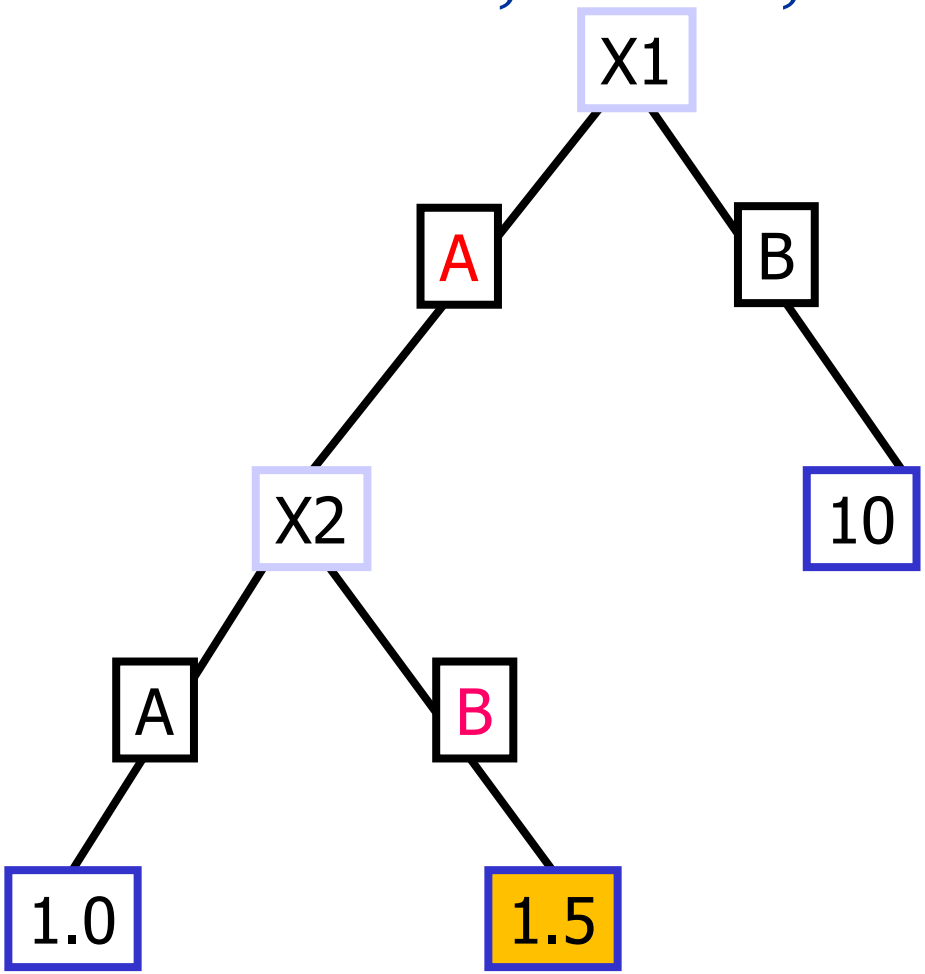
Alberi di Regressione: come usarli?

X1=A, X2=B, Y=?

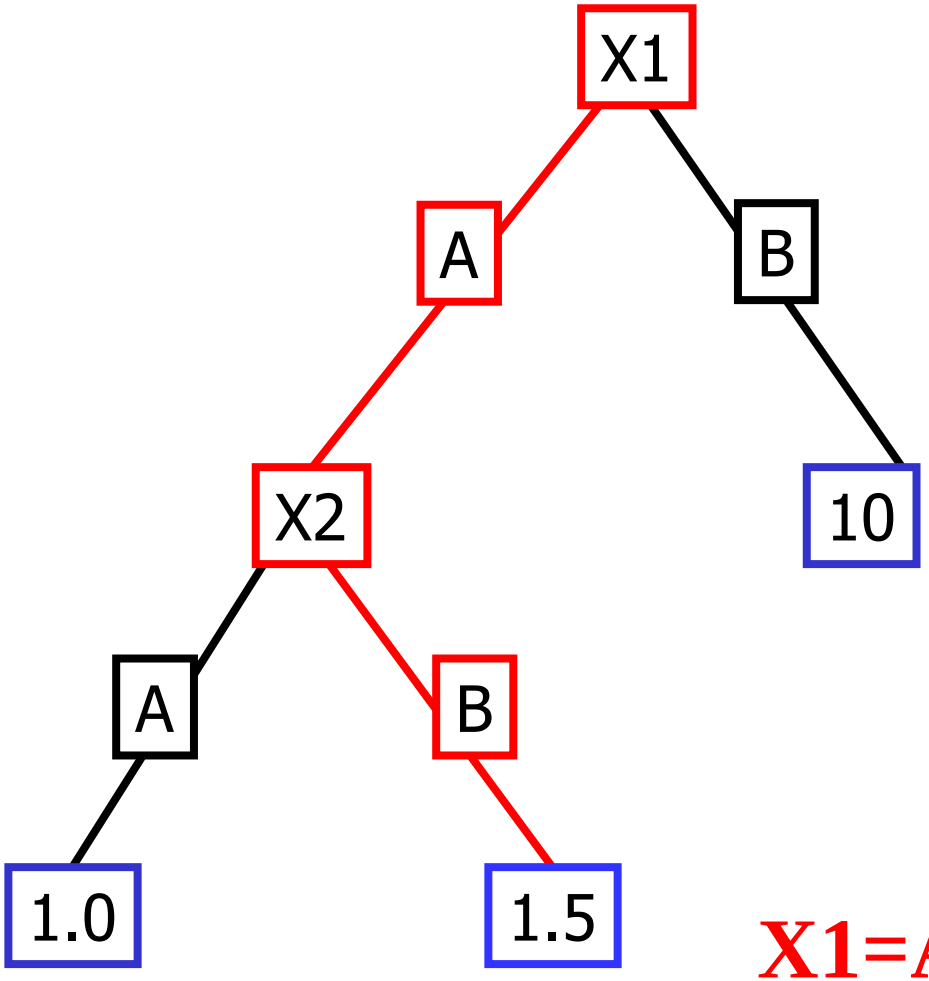


Alberi di Regressione: come usarli?

X1=A, X2=B, Y= 1.5

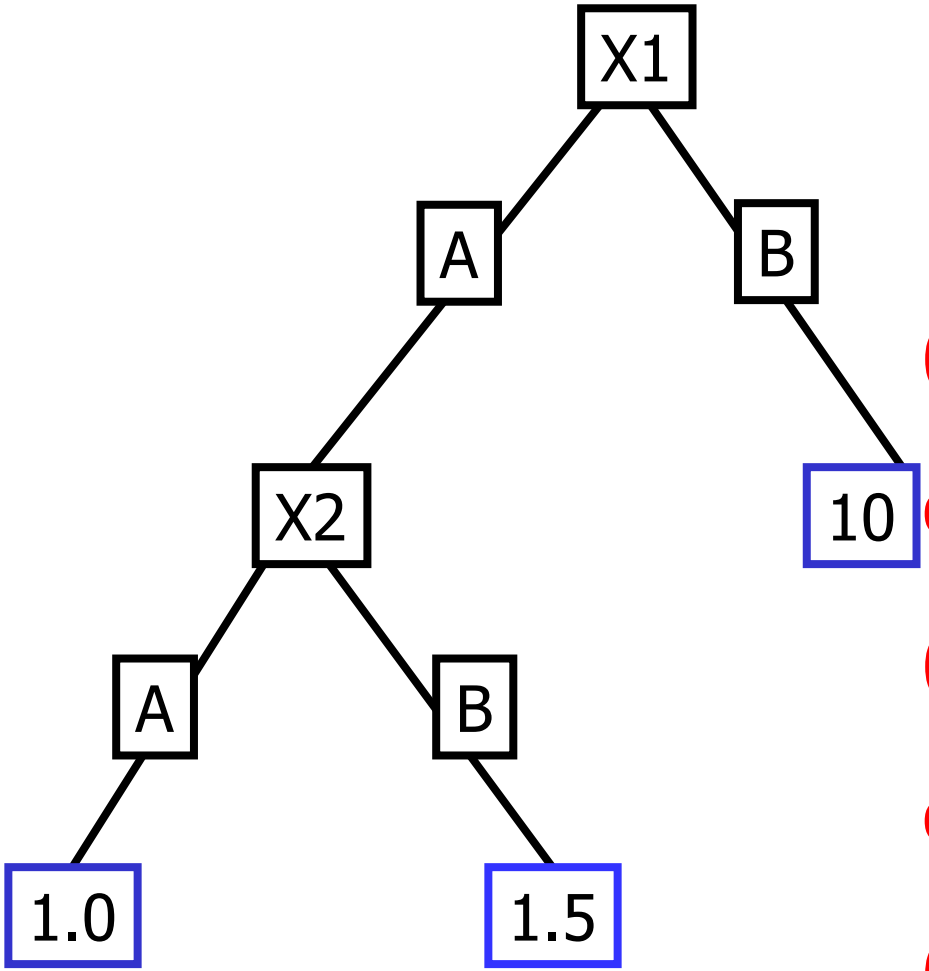


Alberi di Regressione: congiunzione di condizioni



X1=A and X2=B

Alberi di Regressione: disgiunzione di condizioni



(X1=A and X2=A)

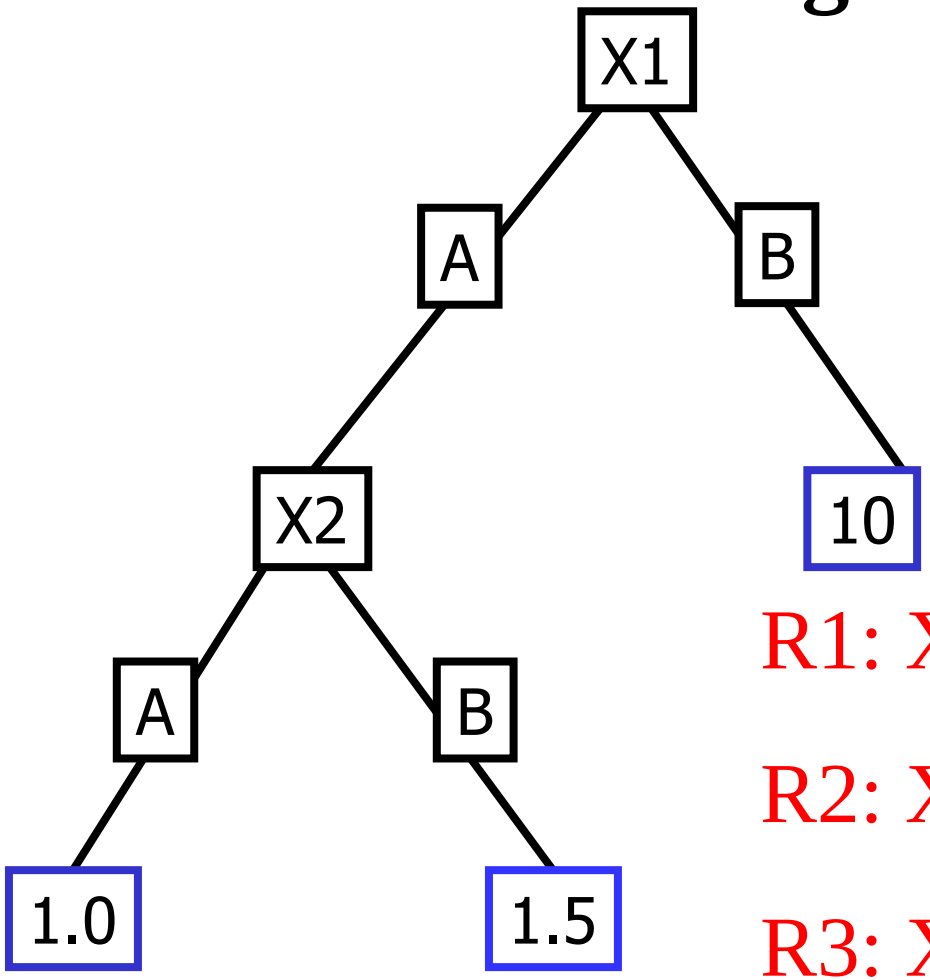
or

(X1=A and X2=B)

or

(X1=B)

Alberi di Regressione: regole di regressione



R1: X1=A and X2=A → Y=1.0

R2: X1=A and X2=B → Y=1.5

R3: X1=B → Y=10

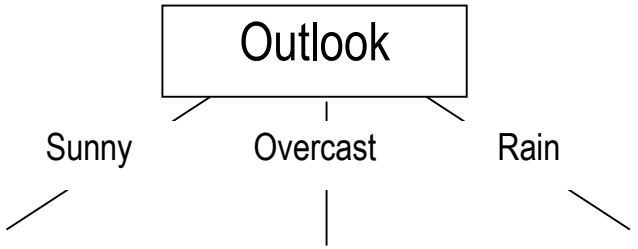
Alberi di Regressione

Tipo di test

Ciascun nodo interno è associato ad un test che coinvolge un attributo A_i.

Se A_i è discreto:

- un test con z alternative, una per ciascun valore assunto da A_i



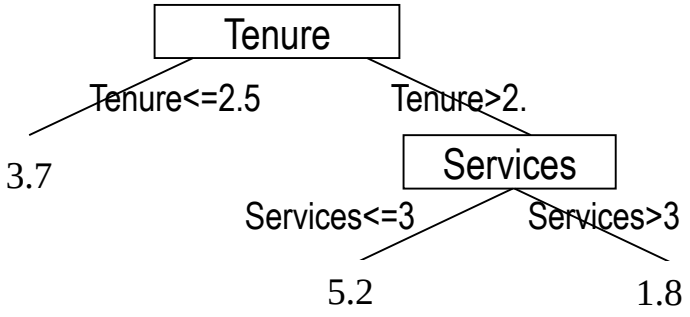
Alberi di Regressione

Tipo di test

Ciascun nodo interno è associato ad un test che coinvolge un attributo A_i.

Se Ai è continuo:

- un test con 2 alternative sulla base di una soglia θ: $Ai \leq \theta$ vs. $Ai > \theta$.



Alberi di Regressione

Selezionare i test

Domanda: Come determinare quale attributo permette di costruire la migliore funzione di regressione ?

Risposta: Varianza!!!

Sia:

- S la porzione di esempi di training correntemente analizzati
- Y la variabile di classe

La varianza di Y in S è calcolata come:

$$\mathrm{var}(S) = \sum_{i \in S} (Y(i) - \bar{Y})^2 = \sum_{i \in S} Y(i)^2 - \frac{(\sum_{i \in S} Y(i))^2}{size(S)}$$

Alberi di Regressione

Selezionare i test

var(S) è una misura della variabilità contenuta in S.

- Assume 0 se solo tutti gli eventi sono associati allo stesso valore di Y

Alberi di Regressione

Selezionare i test

- Sia $S_1, \ldots, S_t$ il partizionamento di S per il test t sull'attributo $A_{i,:}$

$$\mathrm{var}(S,t) = \sum_i \mathrm{var}(S_i)$$

Il criterio basato sulla varianza sceglie il test t che minimizza var(S,t)

Alberi di Regressione

Selezionare i test

Esempio

X1	X2	Y
A	A	1
A	A	1
A	A	1
A	A	1
A	B	1,5
A	B	1,5
A	B	1,5
B	B	10
A	B	1,5
A	B	1,5
B	C	10
B	B	10
B	C	10
B	C	10
A	A	1

Alberi di decisione

Selezionare i test

15 esempi di apprendimento

Y: 1-1-1-1-1.5-1.5-1.5-10-1.5-1.5-10-10-10-10-1

var(S) =255.833

Varianza per X1

X1: A (1-1-1-1-1.5-1.5-1.5-1.5-1.5-1),B (10-10-10-10-10)

Il test su X1 partiziona S come segue:

$var(S,X1)=var(S, X1=A)+var(S, X1=B)=0.625+0=0.625$

Alberi di decisione

Selezionare i test

15 esempi di apprendimento

Y: 1-1-1-1-1.5-1.5-1.5-10-1.5-1.5-10-10-10-10-1

var(S) =255.833

Varianza per X2

X2: A (1-1-1-1-1),B (1.5-1.5-1.5-10-1.5-1.5-10),C(10-10-10)

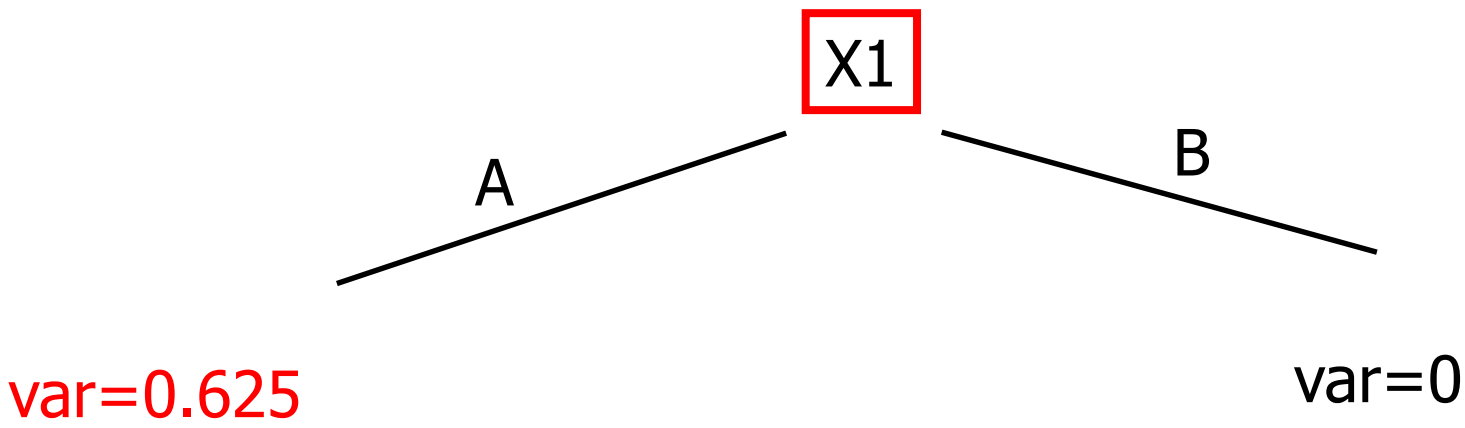
Il test su X2 partiziona S come segue:

var(S,X2)=var(S, X2=A)+var(S, X2=B) +var(S, X3=C)=0+103.21+0=103.21

Alberi di Regressione

Selezionare i test

var=255.833



Alberi di Regressione

Definire le soglie per test continui

- Come identificare le possibili soglie θ per l'attributo continuo A?
  1. ordinare gli esempi sulla base dei valori dell'attributo A (quicksort)
  2. per ciascuna valore distinto risultante dall'ordinamento considerare una possibile soglia per un test A<= soglia vs A>soglia.

Alberi di Regressione

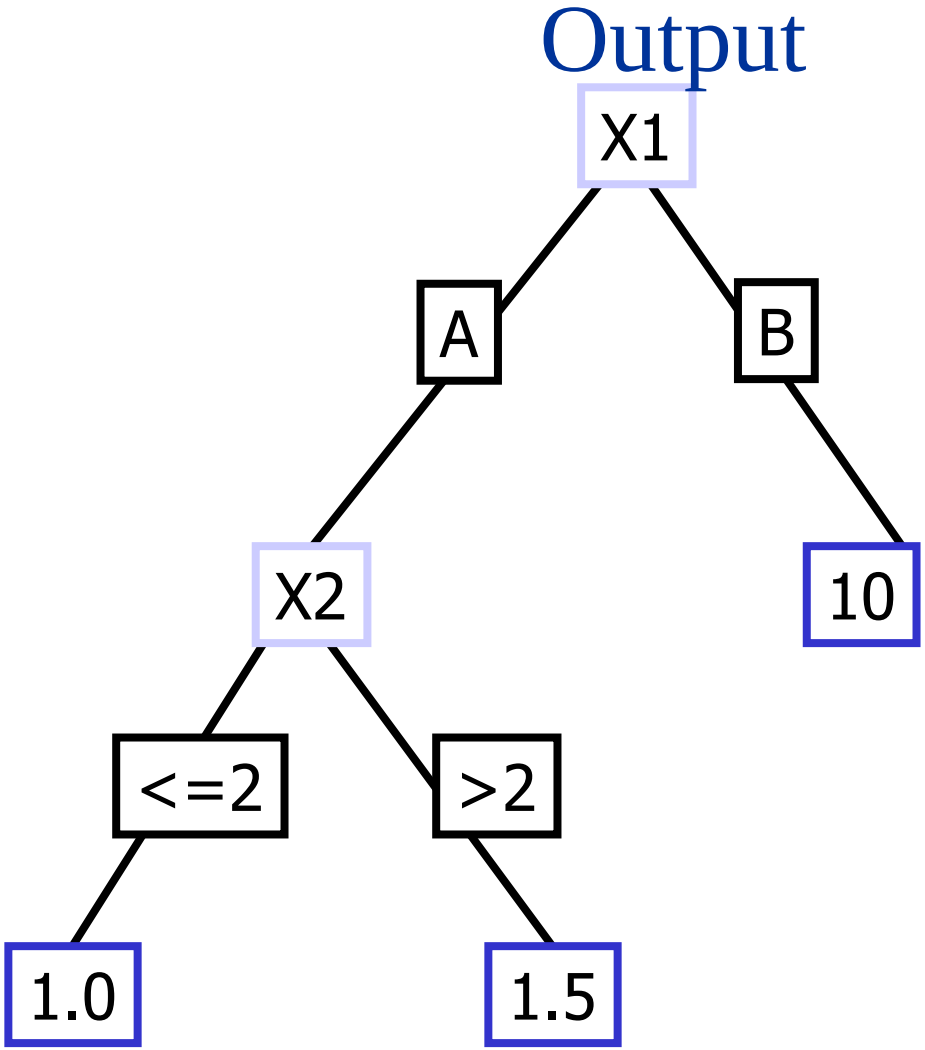
Definire le soglie per test continui

Esempio

1 → var(S,X2<=1,X2>1)=232.69
1
2 → var(S,X2<=2,X2>2)=180.625
2
2
5 → var(S,X2<=5,X2>5)=192.85
5
5
6 → var(S,X2<=6,X2>6)=128.22
6
6
6
10 → var(S,X2<=10,X2>10)=177.30
12 → var(S,X2<=12,X2>12)=219.375
14

X1	X2	Y
A	2	1
A	2	1
A	1	1
A	2	1
A	5	1,5
A	5	1,5
A	6	1,5
B	6	10
A	6	1,5
A	6	1,5
B	10	10
B	5	10
B	12	10
B	14	10
A	1	1

Induzione di Alberi di Regressione



Alberi di Regressione

Algoritmo

```
learnTree(Table S, int begin, int end){
   if( isLeaf(S begin, end)
        root=new LeafNode(S,begin,end);
   else //split node
   {
        root=determineBestSplitNode(S, begin, end);
        childTree=new DecisionTree[root.getNumberOfChildren()];
        for(int i=0;i<root.getNumberOfChildren();i++){
                childTree[i]=new RegressionTree();
                childTree[i].learnTree(trainingSet,root.begin,root.end);
        }
   }
}
```

Caso di studio

Progettare e realizzare un sistema client-server denomianto “Regression Tree Miner”.

Il server include funzionalità di data mining per l’apprendimento di alberi di regressione e uso degli stessi come strumento di previsione.

Il client è un applet Java che consente di effettuare previsioni usufruendo del servizio di predizione remoto

Istruzioni

1. Il progetto dell'a.a. 2015/16 riguarda il “Regression Tree miner" ed è valido solo per coloro che superano la prova scritta entro il corrente a.a. (appello Marzo 2017)
2. Ogni progetto può essere svolto da gruppi di al più TRE (3) studenti
   1. cognome A-L per immatricolati 2014-2015 (anno di corso =2)
   2. cognome A-Z per immatricolazioni antecedenti al 2014 (anno di corso >=3)
3. Coloro i quali superano la prova scritta devono consegnare il progetto ENTRO la data prevista per la corrispondente prova orale.
4. Il voto massimo assegnato al progetto è 33. Un voto superiore a 30 equivale a 30 e lode.
5. Il voto finale è la media del voto attribuito allo scritto e il voto attribuito al progetto.

Valutazione progetto

- Diagramma delle classi (2 punti)
- JavaDoc (3 punti)
- Guida di installazione (con Jar+ Bat+ Script SQL) (2 punti)
- Guida utente con esempi di test (2 punti)
- Sorgente del sistema (14 punti)
- Estensioni del progetto svolto in laboratorio (10 punti)

Istruzioni

- Non si riterrà sufficiente un progetto non sviluppato in tutte le su parti (client-server, applet, accesso al db, serializzazione,…
- Le estensioni aggiungono funzionalità, non le rimuovono