Intelligenza Artificiale

Anno accademico
2008-2009

Machine Learning: Classificazione e Predizione

Sommario

- Classificazione e Predizione
- Classificatori Separate and Conquer (PRISM)
- Classificatori Bayesiani
- Classificatori basati su istanze
- Valutazioni di classificatori
- Combinare classificatori

Classificazione e Predizione

- Classificazione e predizione sono processi che consistono nel creare dei modelli che possono essere usati:
  - per descrivere degli insiemi di dati;
  - per fare previsioni future
- Molti algoritmi sono stati sviluppati per risolvere problemi di classificazione e predizione da settori diversi della comunità scientifica:
  - **Machine Learning**
  - Statistica
  - Neurobiologia

Impieghi del machine learning

- **Data Mining** (estrazione di conoscenza): scoprire regolarità e patterns in dati multidimensionali e complessi (esempio cartelle cliniche)
- **Miglioramento delle performance**: macchine che migliorano le loro capacità (es. movimento di robot)
- **Software adattabili**: programmi che si adattano alle esigenze dell'utente (es. Letizia).

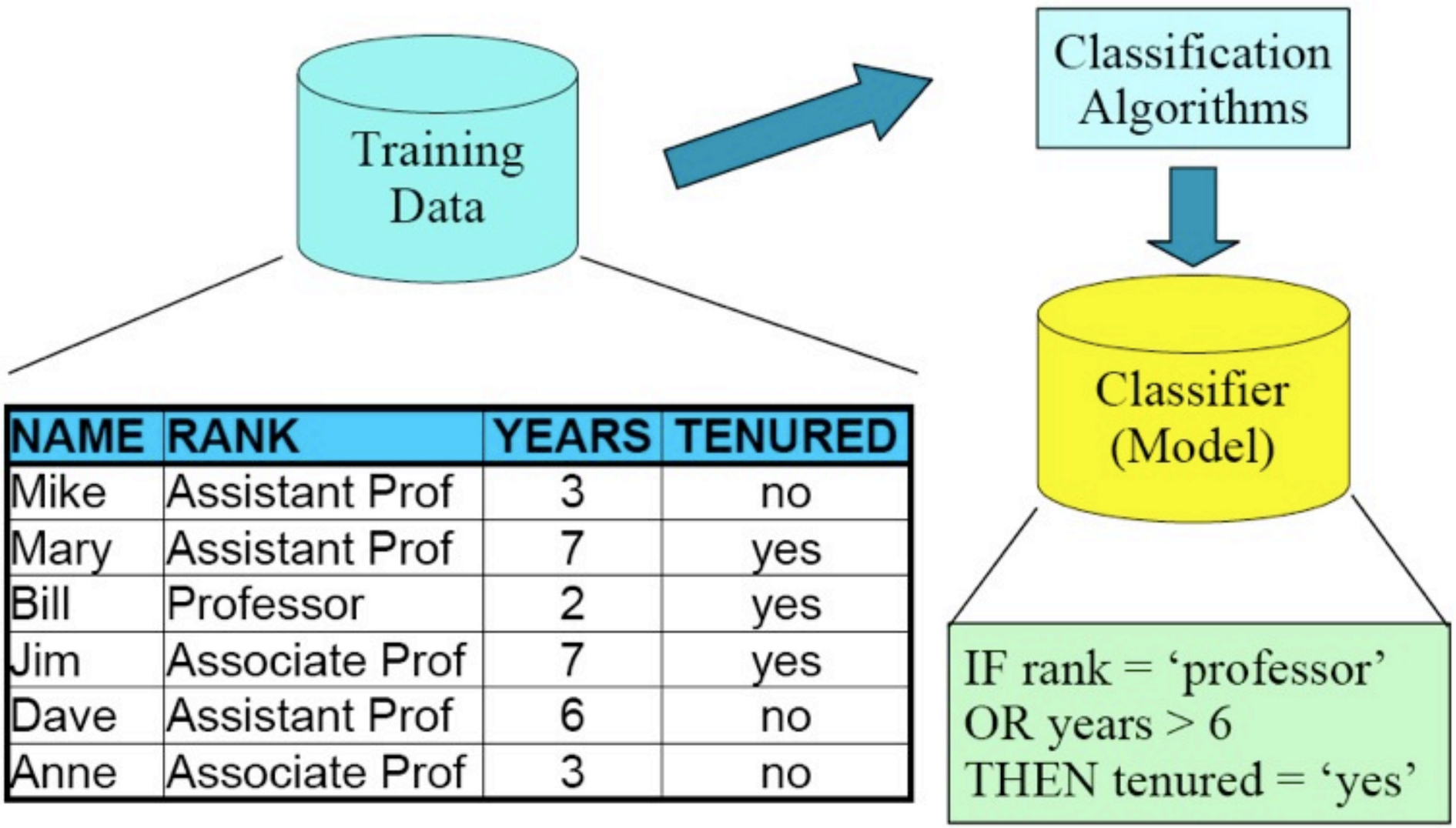
Un paio di casi aziendali

- Consorzio agrario
  - 30 agenzie distribuite sul territorio laziale, vendita concimi, fitofarmaci, sementi, ma anche carburanti e attrezzature
  - oltre 500000 record l'anno
  - Progetto: valutare la fedeltà dei clienti
- AWS
  - il principale corriere espresso a capitale tutto italiano
  - oltre 120 terminal in tutta italia
  - milioni di record all'anno
  - Progetto:
    - stimare il tempo di consegna reale man mano che il pacco procede verso la destinazione, valutando anche possibili percorsi alternativi da quello pianificato che offrano maggiori probabilità di ridurre il tempo di consegna

Classificazione
Fase 1

- Il processo di classificazione può essere visto come un processo a tre fasi:
  - Fase 1 (**Addestramento**): si produce un modello da insieme di addestramento.
    - Si assume che ogni istanza in input faccia parte di una tra un numero predefinito di classi diverse
    - Ogni istanza possiede un attributo che descrive la classe di appartenenza

Costruzione del modello
Fase 1



NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no

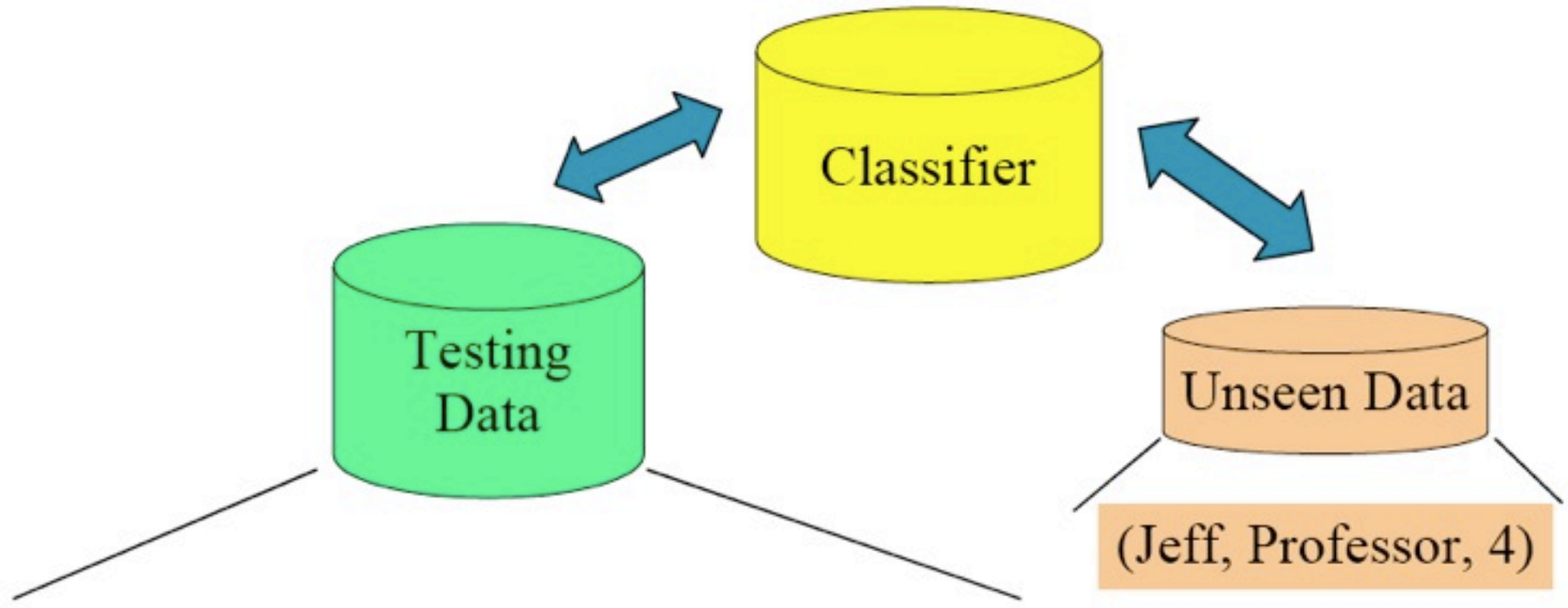
Classificazione
Fase 2

- Fase 2 (**Stima dell'accuratezza**): si stima l'accuratezza del modello usando un insieme di test.
  - ad esempio, si misura la percentuale di errori commessi sull'insieme di test
  - è importante che l'insieme di test sia indipendente dall'insieme usato per il campionamento, per evitare stime troppo ottimistiche.

Classificazione
Fase 3

- Fase 3 (**Utilizzo del modello**): si classificano istanze di classe ignota.
  - Siamo di fronte a istanze di cui si conoscono tutti gli attributi tranne quello di classificazione.

Test e Utilizzo del modello
Fase 2 e Fase 3

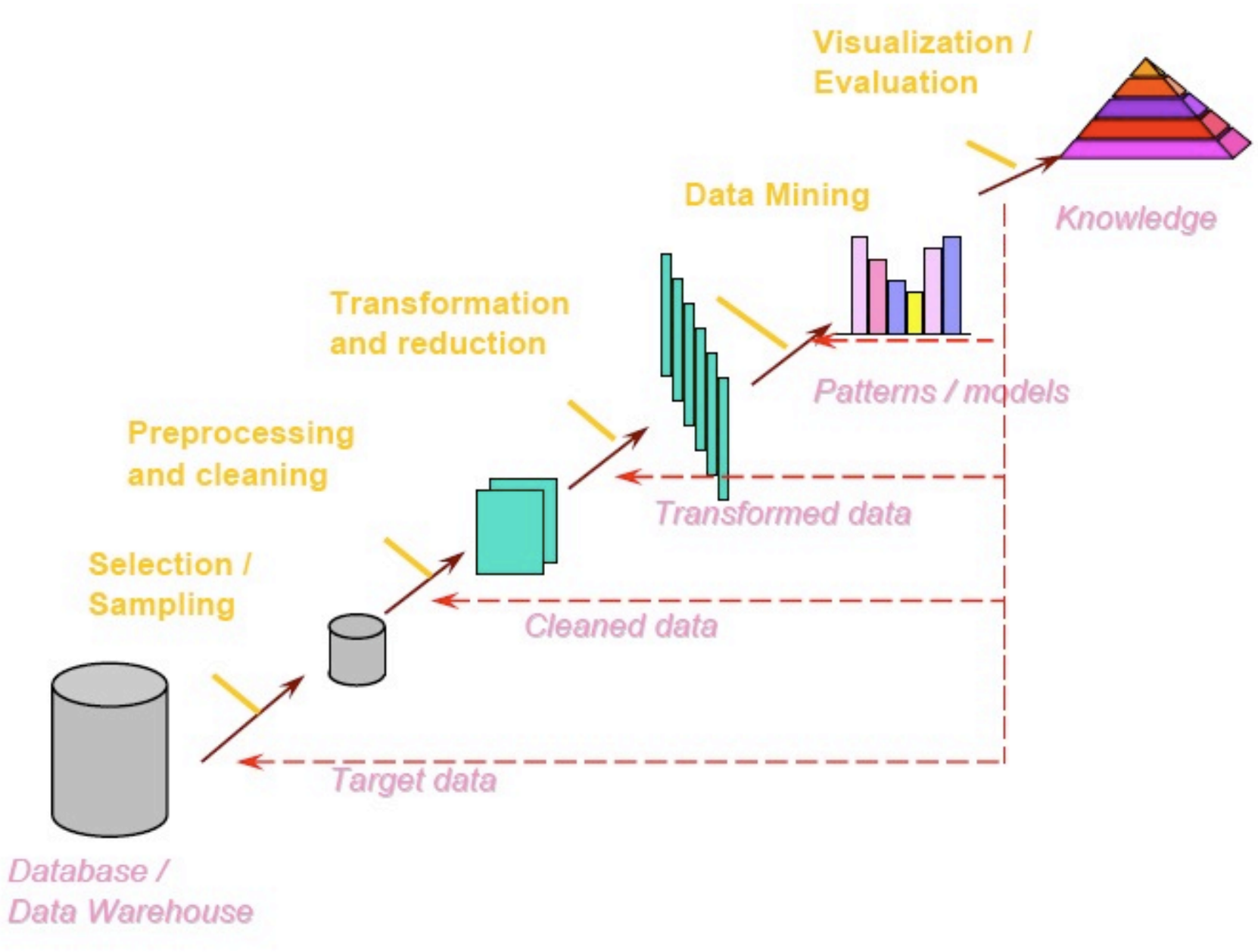


NAME	RANK	YEARS	TENURED
Tom	Assistant Prof	2	no
Merlisa	Associate Prof	7	no
George	Professor	5	yes
Joseph	Assistant Prof	7	yes

Classificazione VS Predizione

- Dal punto di vista etimologico, la **predizione** è il concetto generale, che si divide in
  - **Classificazione** quando la classe è un valore nominale;
  - **Regressione** quando la classe è un valore numerico;
- Nella pratica però il termine Predizione viene usato come sinonimo di Regressione.

Knowledge Discovery in Databases (KDD)



Preparazione dei dati (1/3)

- **Data Cleaning**: pre-elaborare i dati in modo da
  - eliminare il rumore
    - usando tecniche di smoothing
  - eliminare eventuali outliers
    - dati con caratteristiche completamente diverse dal resto degli altri e probabilmente dovuti ad errori nei dati o a casi limite
  - trattare gli attributi mancanti
- Anche se la maggior parte degli algoritmi di classificazione hanno dei meccanismi per eliminare il rumore, una pulizia ad-hoc produce un risultato migliore

Preparazione dei dati (2/3)

- **Analisi di rilevanza degli attributi**
  - Nota anche col termine di **Feature Selection** dal termine usato nella letteratura di Machine Learning
- L'obiettivo è ottimizzare le prestazioni.
  - l'idea è che il tempo impiegato per effettuare una analisi di rilevanza e l'addestramento sull'insieme di attributi ridotti è minore del tempo per effettuare l'addestramento su tutti gli attributi.
- Può anche migliorare la qualità del modello.

Preparazione dei dati (3/3)

- **Trasformazione dei dati**
  - ad esempio, generalizzare alcuni attributi secondo una gerarchia di concetti...
  - ... oppure normalizzarne altri
    - per esempio, ridurre il range di variabilità di un attributo numerico all'intervallo [0,1] sostituendo 0 al valore minimo, 1 al massimo e gli altri di conseguenza.

Valutazione degli algoritmi di classificazione

- **Accuratezza** della predizione
- **Velocità**
  - Tempo usato per costruire il modello e per usarlo
- **Robustezza**
  - Abilità del modello di fare previsioni corrette anche in presenza di dati errati o mancanti
- **Scalabilità**
  - Caratteristica degli algoritmi che sono efficienti non solo per piccoli insiemi di dati ma anche per grandi database.
- **Interpretabilità**
  - Possibilità di assegnare un significato intuitivo al modello generato

Cosa è un outlier

- Outlier: una istanza che è completamente differente dal restante insieme di dati o con esso inconsistente.
  - Cause di outliers
    - Errori
    - inerente variabilità dei dati
    - situazioni anomale (ad esempio tentativi di frode)
- Talvolta ci interessa individuare gli outliers! Si parla di outlier mining:
  - Riconoscimenti di frodi telefoniche
  - Riconoscimenti di attacchi ad un sistema informatico
  - Comportamenti anomali a farmaci

Outlier Mining

- Problema: date n istanze e il numero k, determinare le k istanze più dissimili dalle altre.
- Ma cosa è un dato dissimile dagli altri?
  - Non è facile da definire...
  - Alcune apparenti irregolarità (per esempio, un calo di vendite a Febbraio) potrebbero essere regolari se viste in un contesto più ampio.
  - Si può usare un metodo di visualizzazione grafica per evidenziare gli outlier, e lasciare il compito all'uomo?
    - Solo per dati con poche dimensioni e con attributi prevalentemente numerici

Metodi statistici

- Si assume che i dati siano generati secondo una certa distribuzione di probabilità.
- Si sviluppa un test per validare questa ipotesi
  - si tratta di calcolare qualche statistica di insieme di dati e di confrontare questa statistica con i vari oggetti
  - ad esempio, si può considerare outlier qualunque oggetto che dista più di 3 volte lo scarto quadratico medio dalla media
  - esempi di test famosi: t-student, X, etc.
- Svantaggi
  - la maggior parte dei test riguardano un singolo attributo, mentre nei casi pratici del data mining un outlier è riconoscibile solo guardando molti attributi contemporaneamente.
  - è necessario avere una idea della distribuzione dei dati

Metodi basati sulle distanze

- Un oggetto **o** in un insieme di dati **S** è un outlier basato sulle distanze con parametri **p** e **d** se almeno **p%** degli oggetti in **S** è più lontano di **d** da **o**
- Generalizza i metodi statistici
  - non è necessario conoscere il tipo di distribuzione
  - adatto per analisi multi-dimensionale
- Richiede di settare i parametri **p** e **d**
  - trovare i parametri giusti può richiedere vari tentativi

Regole di classificazione

- Come è stato esaminato per i sistemi di produzione (vedi lucidi) una regola di classificazione è una formula logica del tipo:
  - IF <antecedente> THEN <conseguente>
- L'antecedente è una serie di test, come i nodi di un albero di classificazione (vedi lucidi su alberi di decisione).
  - ... ma può essere una formula logica, in qualche formulazione evoluta.
    - Il conseguente dà la classe da assegnare alle instanze che soddisfano l'antecedente
- ESEMPIO: **If outlook=sunny and humidity=high then play=yes**

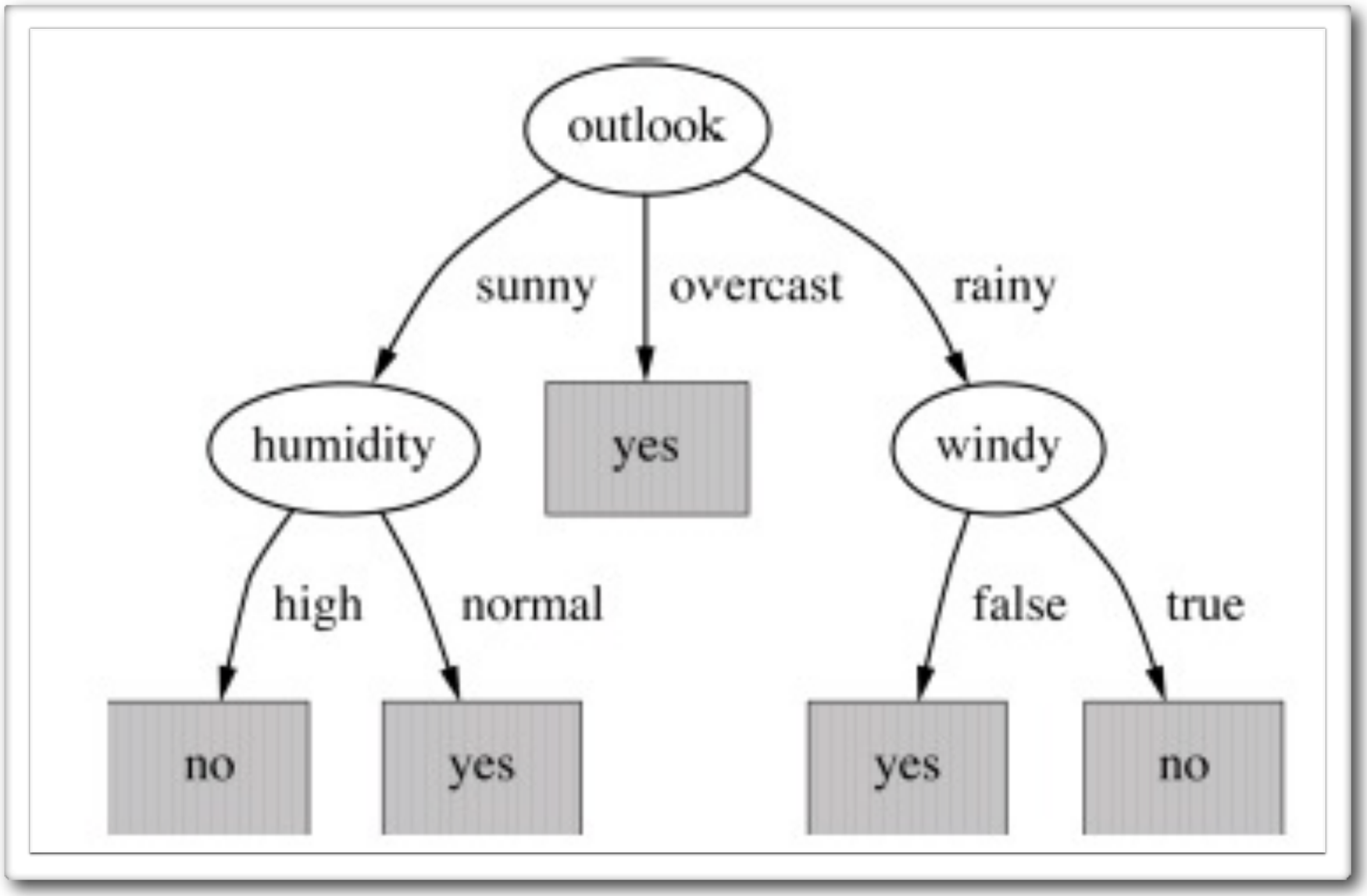
Alberi e regole di classificazione

- Un albero di classificazione si può trasformare facilmente in un insieme di regole di classificazione:
  - Una regola è creata per ogni percorso dalla radice alle foglie
  - Ogni nodo interno del percorso è un test dell'antecedente
  - La classe specificata sulla foglia è il conseguente

Esempio

	Outlook	Temperature	Humidity	Windy	Play
1	Sunny	Hot	High	No	No
2	Sunny	Hot	High	Yes	No
3	Overcast	Hot	High	No	Yes
4	Rainy	Mild	High	No	Yes
5	Rainy	Cool	Normal	No	Yes
6	Rainy	Cool	Normal	Yes	No
7	Overcast	Cool	Normal	Yes	Yes
8	Sunny	Mild	High	No	No
9	Sunny	Cool	Normal	No	Yes
10	Rainy	Mild	Normal	No	Yes
11	Sunny	Mild	Normal	Yes	Yes
12	Overcast	Mild	High	Yes	Yes
13	Overcast	Hot	Normal	No	Yes
14	Rainy	Mild	High	Yes	No

Albero generato dal dataset

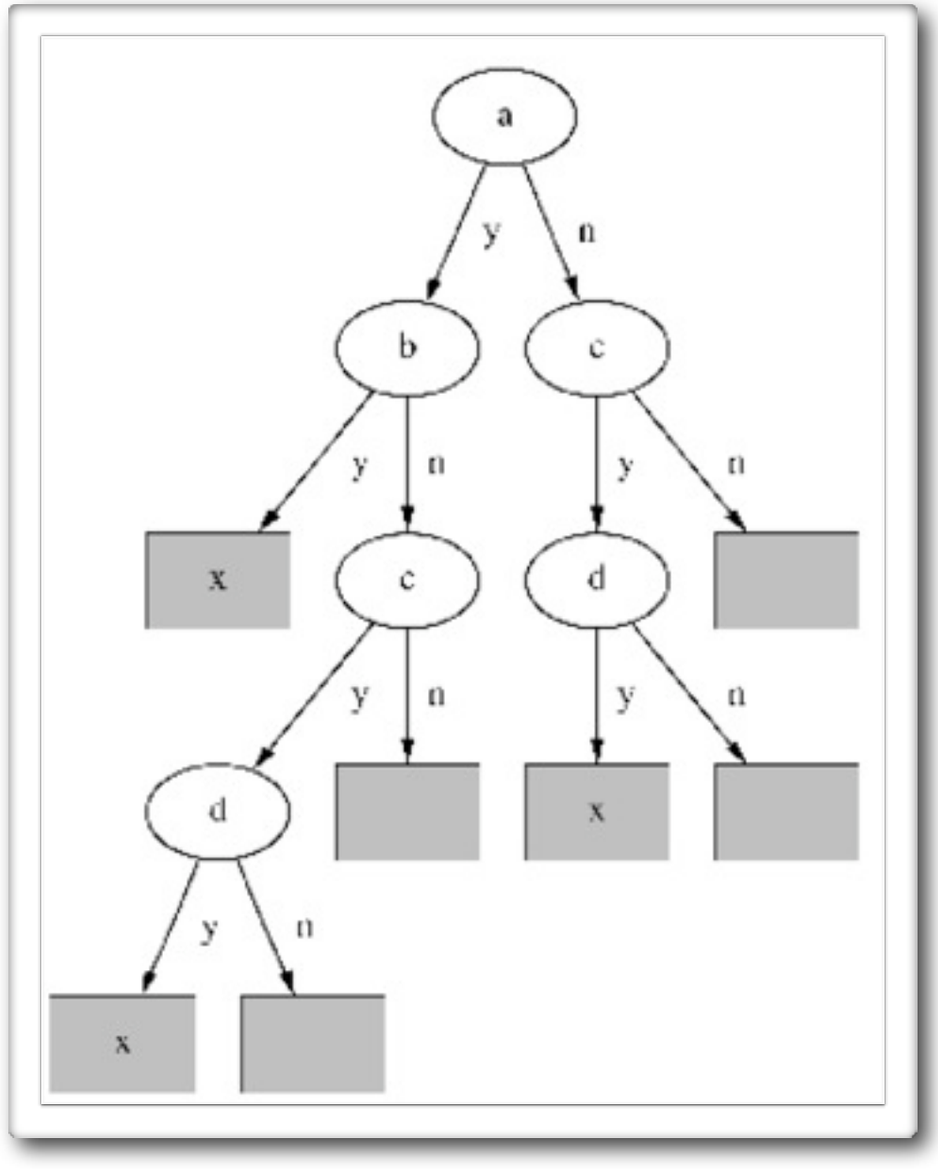


Con Regole di classificazione

- L'albero di decisione visto prima diventa:
  - IF Outlook="Sunny" AND Humidity="High" THEN Play="No"
  - IF Outlook="Sunny" AND Humidity="Normal" THEN Play="Yes"
  - IF Outlook="Overcast" THEN Play="Yes"
  - IF Outlook="Rain" AND Windy="True" THEN Play="no"
  - IF Outlook="Rain" AND Windy="False" THEN Play="yes"
- Si può effettuare il pruning delle singole regole, eliminando le condizioni che non peggiorano (o che magari migliorano) l'accuratezza su un insieme di test distinto.

Alberi e regole di classificazione

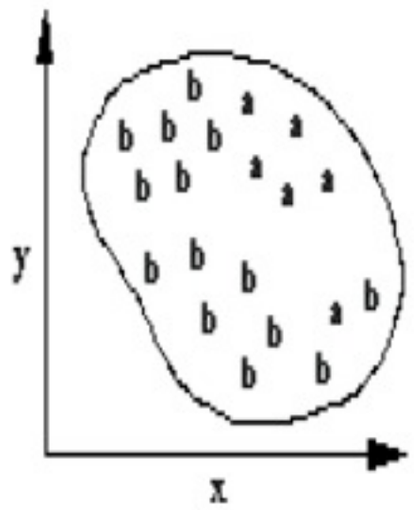
- Il procedimento inverso, da regole ad albero, non è così semplice, e può produrre alberi molto più grandi del previsto.
  - In particolare si ha il problema dei sottoalberi replicati.
- Supponiamo di avere le seguenti regole
  - `if a and b then x`
  - `if c and d then x`

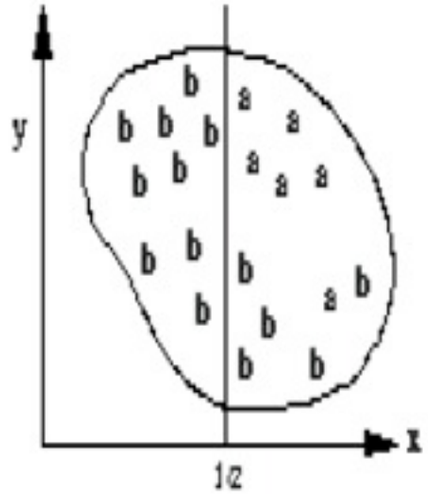


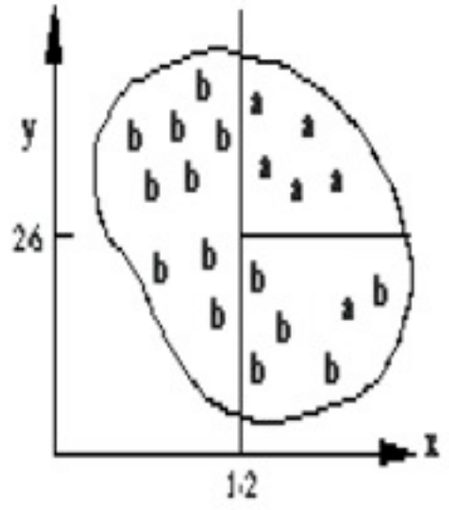
Algoritmi di copertura

- Spesso, la generazione di regole di decisione da alberi di decisione genera regole eccessivamente complicate.
- Ci sono algoritmi che generano direttamente un insieme di regole
  - per ogni classe, trova l'insieme di regole che copre tutte le istanze di quella classe.
    - Determina una prima regola che copre alcune istanze della classe
    - Trova un'altra regola che copre alcune delle rimanenti...
    - ... e così via
- Si parla di algorimti di copertura poiché ad ogni passo coprono un sottoinisieme delle istanze della classe.

Esempio: generazione di una regola








```
if true then class=a
```

```
if x>1.2 then class=a
```

```
if x>1.2 and y>2.6 then class=a
```

Copertura ed accuratezza

- Sia **R** una regola e
  - **t**: un numero di istanze che soddisfano la premessa
  - **p**: un numero di istanze che soddisfano premessa e conclusione
- Si definiscono allora i concetti di
  - **Copertura**: corrisponde al valore **t**
  - **Accuratezza**: corrisponde a **p/t**, ovvero la percentuale, tra le istanze che soddisfano la premessa, che soddisfano anche la conclusione.

Un semplice algoritmo di copertura

- Si parte da una regola di base senza condizioni.
- Si migliora la regola aggiungendo dei test che ne massimizzano l'accuratezza.
- Problema simile a quello degli alberi di decisione: decidere su che attributi dividere.
  - I metodi di copertura massimizzano l'accuratezza delle regole e considerano una sola classe. I test sono sempre binari!
- Ogni nuovo test riduce la copertura della regola.
- Ci si ferma se l'accuratezza è 1 o non si può dividere.

Esempio: lenti a contatto (1/7)

Age	Spectacle prescription	Astigmatism	Tear production rate	Recommended lenses
Young	Myope	No	Reduced	None
Young	Myope	No	Normal	Soft
Young	Myope	Yes	Reduced	None
Young	Myope	Yes	Normal	Hard
Young	Hypermetrope	No	Reduced	None
Young	Hypermetrope	No	Normal	Soft
Young	Hypermetrope	Yes	Reduced	None
Young	Hypermetrope	Yes	Normal	hard
Pre-presbyopic	Myope	No	Reduced	None
Pre-presbyopic	Myope	No	Normal	Soft
Pre-presbyopic	Myope	Yes	Reduced	None
Pre-presbyopic	Myope	Yes	Normal	Hard
Pre-presbyopic	Hypermetrope	No	Reduced	None
Pre-presbyopic	Hypermetrope	No	Normal	Soft
Pre-presbyopic	Hypermetrope	Yes	Reduced	None
Pre-presbyopic	Hypermetrope	Yes	Normal	None
Presbyopic	Myope	No	Reduced	None
Presbyopic	Myope	No	Normal	None
Presbyopic	Myope	Yes	Reduced	None
Presbyopic	Myope	Yes	Normal	Hard
Presbyopic	Hypermetrope	No	Reduced	None
Presbyopic	Hypermetrope	No	Normal	Soft
Presbyopic	Hypermetrope	Yes	Reduced	None
Presbyopic	Hypermetrope	Yes	Normal	None

Esempio: lenti a contatto (2/7)

- Regola cercata: if ? Then reccomendation = hard.
- Test possibili:
  - Age=Young 2/8
  - Age=Pre-presbyopic 1/8
  - Age=Presbyopic 1/8
  - Spectacle Prescription=Myope 3/12
  - Spectacle Prescription=Hypermetrope 1/12
  - Astigmatism=no 0/12
  - Astigmatism=yes 4/12
  - Tear Prod. Rate=reduced 0/12
  - Tear Prod. Rate=normal 4/12

Esempio: lenti a contatto (3/7)

- Regola migliore aggiunta:
  - if astigmatism=yes then reccomendation=hard
- Istanze coperte dalla nuova regola:

Age	Spectacle prescription	Astigmatism	Tear production rate	Recommended lenses
Young	Myope	Yes	Reduced	None
Young	Myope	Yes	Normal	Hard
Young	Hypermetrope	Yes	Reduced	None
Young	Hypermetrope	Yes	Normal	hard
Pre-presbyopic	Myope	Yes	Reduced	None
Pre-presbyopic	Myope	Yes	Normal	Hard
Pre-presbyopic	Hypermetrope	Yes	Reduced	None
Pre-presbyopic	Hypermetrope	Yes	Normal	None
Presbyopic	Myope	Yes	Reduced	None
Presbyopic	Myope	Yes	Normal	Hard
Presbyopic	Hypermetrope	Yes	Reduced	None
Presbyopic	Hypermetrope	Yes	Normal	None

Esempio: lenti a contatto (4/7)

- Stato corrente:
  - if astigmatism=yes and ? then reccomendation=hard
- Test possibili:

– Age=Young	2/4
– Age=Pre-presbyopic	1/4
– Age=Presbyopic	1/4
– Spectacle Prescription=Myope	3/6
– Spectacle Prescription=Hypermetrope	1/6
– Tear Prod. Rate=reduced	0/6
– Tear Prod. Rate=normal	4/6

Esempio: lenti a contatto (5/7)

- Regola migliore aggiunta
  - if astigmatism=yes and tear prod. rate=normal then reccomentation=hard
- Istanze coperte dalla nuova regola

Age	Spectacle prescription	Astigmatism	Tear production rate	Recommended lenses
Young	Myope	Yes	Normal	Hard
Young	Hypermetrope	Yes	Normal	hard
Pre-presbyopic	Myope	Yes	Normal	Hard
Pre-presbyopic	Hypermetrope	Yes	Normal	None
Presbyopic	Myope	Yes	Normal	Hard
Presbyopic	Hypermetrope	Yes	Normal	None

Esempio: lenti a contatto (6/7)

- Stato corrente
  - if astigmatism=yes and tear prod. rate=normal and ? then reccomendation=hard
- Test possibili:
  - Age=Young 2/2
  - Age=Pre-presbyopic 1/2
  - Age=Presbyopic 1/2
  - Spectacle Prescription=Myope 3/3
  - Spectacle Prescription=Hypermetrope 1/3
- Tra Age=Young e Spectacle Prescription=Myope scegliamo il secondo perché ha copertura maggiore.

Esempio: lenti a contatto (7/7)

- Regola finale:
  - if astigmatism=yes and tear prod. rate=normal and Spectacle Prescription=Myope then reccomendation=hard
- Seconda regola per “hard lenses”
  - ottenuta dalle istanze non coperte dalla prima regola
  - if age=young and astigmatism=yes and tear prod. rate=normal then reccomendation=hard
- Queste regole coprono tutte le istanze delle lenti rigide.
  - Il processo è ripetuto per gli altri valori della classe.

L'algoritmo PRISM

- Quello che abbiamo descritto è l'algoritmo PRISM.
  - Genera regole perfette, con accuratezza del 100%

```
Per ogni classe C
    inizializza E con l'insieme di tutte le istanze
    creare una regola R con una parte sinistra vuota che predice C
    finché R è perfetta (o non ci sono più attributi da usare)
        per ogni attributo A non menzionato in R, e ogni valore v
            considera di aggiungere la regola A=v al lato sinistro di R
            seleziona il valore v che massimizza l'accuratezza
                (in caso di più valori massimali, considera quello che
                massimizza anche il supporto)
            aggiungi A=v alla regola R
        rimuovi le istanze coperte da R in E
```

Regole contraddittorie?

- Se ci sono due istanze uguali, tranne che per l'attributo di classe, vengono generate regole contraddittorie

X	Y	Classe
a	b	y
a	b	n
a	b	n
a	a	y
b	b	n

- Si generano le seguenti regole:
  - `if Y=a then Classe=y`
  - `if X=a and Y=b then Classe=y`
  - `if X=b then Classe=n`
  - `if Y=b and X=a then Classe=n`

Regole contraddittorie?

- La 2 e la 4 sono contraddittorie!
  - Cosa fare nel caso X=a, Y=b ?
- Soluzione:
  - considero le regole nell'ordine in cui le ho determinate, ed applico la prima in cui la condizione è verificata.
  - in alternativa potrei considerare tutte le regole che si possono applicare e scegliere quella con accuratezza maggiore.
- Nota Bene: qualunque strategia di risoluzione si adotti, in caso di verifica di correttezza sull'insieme fornito in training, troveremo un errore di classificazione!

Separate and Conquer

- Metodi come PRISM sono noti come algoritmi *separate and conquer*.
  - Viene identificata una regola
    - Le istanze coperte dalla regola vengono separate dal resto
    - Le istanze che rimangono vengono conquistate
- La differenza con i metodi *divide and conquer* (come gli alberi di decisione)
  - I sottoinsiemi separati non devono più essere considerati

Cosa sono i classificatori Bayesiani?

- Sono metodi statistici di classificazione:
  - Hanno le seguenti carattersitiche:
    - Predicono la probabilità che una data istanza appartenga ad una certa classe
    - Sono metodi incrementali: ogni istanza dell'insieme di addestramento modifica in maniera incrementale la probabilità che una ipotesi sia corretta.
      - La conoscenza già acquisita può essere combinata facilmente con le nuove osservazioni basta aggiornare i conteggi)
      - Usati ad esempio in Mozilla o SpamAssassin per riconoscere le mail spam dalle mail ham.

Richiamo di teoria delle probabilità (1/3)

- Una probabilità è una misura su un insieme di eventi che soddisfa tre assiomi.

1. La misura di ogni evento è compresa tra 0 e 1
    - Questo si scrive $0 \leq P(E=e_i) \leq 1$, dove E è una variabile casuale che rappresenta un eventi ed ei sono i possibili valori di E. In generale, le variabili casuali si indicano con lettere maiuscole e i loro valori con lettere minuscole.

Richiamo di teoria delle probabilità (2/3)

- Una probabilità è una misura su un insieme di eventi che soddisfa tre assiomi.

2. La misura dell'intero insieme è 1:

$$\sum_{i=1}^{n} P(E = e_i) = 1$$

Richiamo di teoria delle probabilità (3/3)

- Una probabilità è una misura su un insieme di eventi che soddisfa tre assiomi.

3. La probabilità dell'unione di eventi disgiunti è pari alla somma delle probabilità dei singoli eventi:

$$P(E = e_1 \vee E = e_2) = P(E = e_1) + P(E = e_2)$$

Dove e_1 ed e_2 sono disgiunti

Modello probabilistico

- Un modello probabilistico consiste in uno spazio di possibili esiti mutualmente esclusivi insieme alla misura di probabilità associata ad ogni esito.
- Che tempo fa domani?
  - esiti: {SOLE, NUVOLE, PIOGGIA, NEVE}
- L'evento corrispondente ad una precipitazione è il sottoinsieme {PIOGGIA, NEVE}.

Notazioni

- Useremo P(E) per denotare il vettore di valori:

$$\langle P(E = e_1), \ldots, P(E = e_n) \rangle$$

- Useremo P(e_i) come abbreviazione di P(E=e_i)
- Useremo $\sum_e P(e)$ come abbreviazione per

$$\sum_{i=1}^{n} P(E = e_i)$$

Probabilità condizionale

- La probabilità condizionale P(B|A) è definita come:

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

- A e B sono condizionalmente indipendenti se P(B|A)=P(B) o il suo equivalente P(A|B)=P(A)

Funzionamento dei classificatori Bayesiani

- Sia X una istanza da classificare, e $C_1,\ldots,C_n$ le possibili classi. I classificatori Bayesiani calcolano $P(C_i|X)$ come

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)}$$

- Si sceglie la classe C_i che massimizza $P(C_i|X)$
  - P(X) è uguale per tutte le classi per cui non occorre calcolarla
  - $P(C_i)$ si può calcolare facilmente sull'insieme dei dati di addestramento
    - si conta la percentuale di istanze di classe C_i sul totale
  - Come si calcola $P(X|C_i)$?

Classificatori Naïve

- Assunzione dei classificatori naïve: indipendenza degli attributi.
- Se X è composta dagli attributi $A_1=a_1 \ldots A_m=a_m$, otteniamo

$$P(X|C_i) = \prod_{j=1}^{m} P(A_j = a_j|C_i)$$

- Se A_j è **categorico**, $P(A_j=a_j|C_i)$ viene stimato come la frequenza relativa delle istanze che hanno valore $A_j=a_j$ tra tutte le istanze di C_i
- Se A_j è **continuo**, si assume che $P(A_j=a_j|C_i)$ segue una distribuzione Gaussiana, con media e varianza stimata a partire dalle istanze di classe C_i.

Esempio (1/2)

Outlook	Temperature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N

P(p) = 9/14
P(n) = 5/14

outlook	
P(sunny|p) = 2/9	**P(sunny|n) = 3/5**
P(overcast|p) = 4/9	**P(overcast|n) = 0**
P(rain|p) = 3/9	**P(rain|n) = 2/5**
temperature	
P(hot|p) = 2/9	**P(hot|n) = 2/5**
P(mild|p) = 4/9	**P(mild|n) = 2/5**
P(cool|p) = 3/9	**P(cool|n) = 1/5**
humidity	
P(high|p) = 3/9	**P(high|n) = 4/5**
P(normal|p) = 6/9	**P(normal|n) = 2/5**
windy	
P(true|p) = 3/9	**P(true|n) = 3/5**
P(false|p) = 6/9	**P(false|n) = 2/5**

Esempio (2/2)

- Arriva un nuovo campione X=<rain, hot, high, false>
- P(X|p)P(p)=P(rain|p)P(hot|p)P(high|p)P(false|p)P(p) = 3/9*2/9*3/9*6/9*9/14 = 0.010582
- P(X|n)P(n)=P(rain|n)P(hot|n)P(high|n)P(false|n)P(n) = 2/5*2/5*4/5*2/5*5/14 = 0.018286
- Il campione X è classificato nella classe n (don’t play) in quanto la verosimiglianza per **n** è più elevata

Il problema delle frequenze nulle

- Cosa succede se per un certo valore di un attributo non si verifica mai per una data classe (ad esempio humidity=high è sempre falso per la classe p)
  - allora P(humidity="high"|p)=0
  - quando arriva una nuova istanza X con humidity=high, sarà sempre P(p|X)=0 indipendentemente da quanto siano probabili i valori per gli altri attributi.
- Per risolvere questo problema si usa sommare 1 ai conteggi per le coppie (attributo, valore)
  - in questo modo nessuna probabilità è mai nulla

Vantaggi e svantaggi dei classificatori Naïve

- L'assunzione di indipendenza (naive)
  - rende i calcoli possibili
  - consente di ottenere classificatori ottimali quando è soddisfatta...
  - ... ma è raramente soddisfata in pratica
- Una possibile soluzione: reti Bayesiane
  - Consentono di considerare le relazioni causali tra gli attributi
- In realtà, si è visto che anche quando l'ipotesi di indipendenza non è soddisfatta, il classificatore naïve Bayes spesso fornisce ottimi risultati.

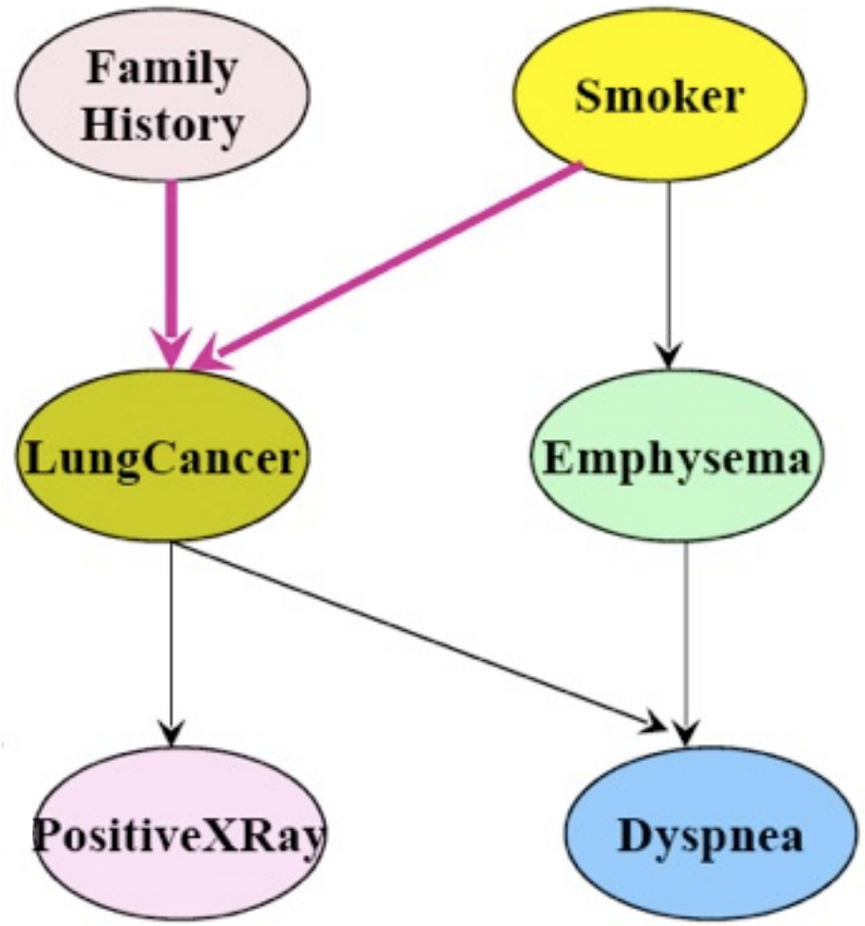
Reti bayesiane (1/2)

- Uno dei componenti di una rete bayesiana è un grafo diretto aciclico nel quale ogni nodo è una variabile casuale e ogni arco rappresenta una dipendenza probabilistica

- Sia X un nodo e Y l'insieme dei suoi genitori

  - X è indipendente da tutte le variabili aleatorie che non discendono da X se sono noti i valori di Y.

  - Se sono noti i valori di Family History e Smoker, allora LungCancer è indipendente da Emphysema



Reti bayesiane (2/2)

- Un altro componente di una rete Bayesiana è la tabella di probabilità condizionata

	(FH, S)	(FH, ~S)	(~FH, S)	(~FH, ~S)
LC	0.8	0.5	0.7	0.1
~LC	0.2	0.5	0.3	0.9

- Esprime i valori di P(LungCancer|FamilyHistory, Smoker)
- La probabilità totale di una tupla ($z_1,...,z_n$) corrispondente alle variabili casuali $Z_1,...,Z_n$) è

$$P(z_1, ..., z_n) = \prod_{i=1}^{n} P(z_i | Parents(Z_i))$$

Metodi basati sulle istanze

- Memorizziamo le istanze che hanno parte dell'insieme di addestramento e rimandiamo tutte le elaborazioni al momento in cui una nuova istanza deve essere classificata.
  - Si parla di lazy evaluation (valutazione pigra) contrapposta alla eager evaluation degli altri metodi
- Esempi tipici: k-nn, Case-Based Reasoning.

Vantaggi e Svantaggi

- Vantaggi:
  - Tipicamente dà luogo a metodi incrementali
    - Hanno accuratezza maggiore perchè lo “Spazio delle ipotesi” è più grande
- Svantaggi
  - Computazionalmente pesante

k-nearest neighbor (1/3)

- Supponiamo le istanze siano formulate da n attributi (escluso l'attributo di classe)
  - Ogni istanza è un punto in uno spazio n-dimensionale
  - Tra due istanze si definisce una distanza con la formula:

$$d(X,Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

- Nasce la necessità di normalizzare i dati in modo da dare a tutti gli attributi lo stesso peso.

k-nearest neighbor (2/3)

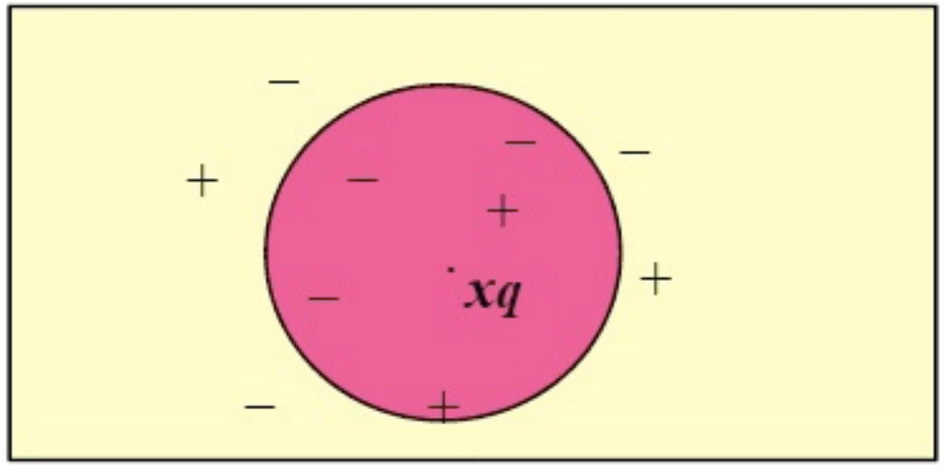
- Data una nuova istanza da classificare
  - si considerano le k istanze dell'insieme di addestramento più vicine
  - la classe predetta è quella più comune tra queste k istanze

k-nearest neighbor (3/3)

- Vantaggi
  - Robustezza
  - Incrementalità
- Svantaggi
  - Alto costo computazionale durante l'uso
    - Richiede tecniche di indicizzazione sofisticate per trovare i possibili vicini senza dover cercare tutte le istanze
  - Gli attributi irrilevanti vengono pesati come gli altri... può causare instabilità
    - Necessità di una fase iniziale che elimini gli attributi irrilevanti

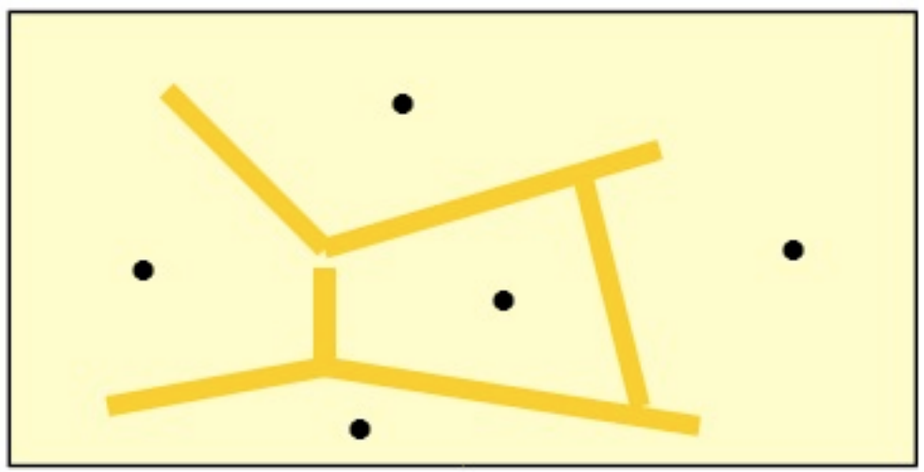
Esempio (1/2)

- Il punto x_q nella figura:
  - è classificato come + se k=1
  - è classificato come - se k=5



Esempio (2/2)

- Lo spazio delle ipotesi viene diviso in poligoni convessi
  - Ogni poligono contiene istanze che vengono classificate allo stesso modo
- a destra viene visualizzata la superficie di decisione indotta da un classificatore 1-NN (diagrammi di Vonoroi)



Distanza tra le istanze

- Gli algoritmi basati sulle istanze usano di solito rappresentare i dati in uno di questi due modi

- **Matrice dati**
  - x_{ij} = attributo i della istanza j

$$\begin{pmatrix} x_{11} & \dots & x_{1f} & \dots & x_{1p} \\ \dots & \dots & \dots & \dots & \dots \\ x_{1p} & \dots & \dots & \dots & x_{ip} \\ \dots & \dots & \dots & \dots & \dots \\ x_{n1} & \dots & x_{nf} & \dots & x_{np} \end{pmatrix}$$

Distanza tra le istanze

- Gli algoritmi basati sulle istanze usano di solito rappresentare i dati in uno di questi due modi

- **Matrice delle distanze:**
  - d(i,j)=distanza tra l'istanza i e j
  - d(i,j)=d(j,i) per cui si rappresenta solo metà matrice

$$\begin{pmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ d(n,1) & d(n,2) & \dots & \dots & 0 \end{pmatrix}$$

Distanze e tipi di dati

- d(i,j) misura la “dissimilarità” o distanza tra le istanze i e j.
- La definizione di d cambia molto a seconda del tipo di dato degli attributi
  - Intervallo
  - Nominali
  - Ordinali
- E ovviamente si possono avere situazioni in cui attributi diversi hanno tipo diverso!

Dati di tipo intervallo e normalizzazione

- Il primo passo per definire una misura di distanza su dati di tipo intervallo, è normalizzare i dati.
- Quasi sempre, si vuole che i vari attributi pesino in maniera uguale.
  - Esempio: una serie di istanze che rappresentano città
    - Attributi: temperatura media (gradi centigradi) e popolazione (numero di abitanti)
    - Il range di valori della popolazione è molto più ampio, ma si vuole che questo attributo non conti proporzionalmente di più

Normalizzazione
Zero score normalization

- Zero score normalization
  - Per ogni attributo f, calcolo la media m_f delle x_{if}

$$m_f = \frac{1}{n}(x_{1f} + x_{2f} + \cdots + x_{nf})$$

  - Calcolo lo scarto assoluto medio

$$s_f = \frac{1}{n}|x_{1f} - m_f| + |x_{2f} - m_f| + \ldots |x_{nf} - m_f|$$

  - Ottengo lo Z-score

$$z_{if} = \frac{x_{if} - m_f}{s_f}$$

Distanza su dati di tipo binario (1/2)

- Per calcolare la distanza tra l'istanza i e j, sia data la seguente tabella di contigenza
- In riga h, colonna k sta il numero di attributi per cui l'instanza i ha valore h e l'istanza j ha valore k

		Object *j*	
		1	0
Object *i*	1	*a*	*b*
	0	*c*	*d*

Distanza su dati di tipo binario (2/2)

- Attributi simmetrici
  - quando valori positivi e negativi contano allo stesso modo
- Attributi asimmetrici
  - quando valori positivi sono più importanti di valori negativi
  - ad esempio il risultato di un test su una malattia
- Indice di Russel-Sao per attriburi simmetrici

$$d(i,j) = \frac{b+c}{a+b+c+d}$$

- Indice di Jaccard per attributi asimmetrici

$$d(i,j) = \frac{b+c}{a+b+c}$$

Distanza su dati di tipo binario
Esempio

Name	Gender	Fever	Cough	Test-1	Test-2	Test-3	Test-4
Jack	M	Y	N	P	N	N	N
Mary	F	Y	N	P	N	P	N
Jim	M	Y	P	N	N	N	N

- Gender è un attributo simmetrico, gli altri sono asimmetrici
- Supponiamo di calcolare la distanza solo sulla base di attributi asimmetrici
  - Se Y e P equivalgono a 1 e N equivale a 0, abbiamo

$$d(jack, mary) = \frac{0+1}{2+0+1} = 0.33$$

$$d(jack, jim) = \frac{1+1}{1+1+1} = 0.67$$

$$d(jim, mary) = \frac{1+2}{1+1+2} = 0.75$$

Distanza su dati di tipo nominale

- Una semplice estensione del caso binario
  - L'attributo può assumere più di due valori.
- Metodo 1: matching semplice
  - **m**: numero di attributi che corrispondono
  - **p**: numero totale di attributi
  - distanza:

$$d(i,j) = \frac{p-m}{p}$$

Distanza su dati di tipo nominale

- Una semplice estensione del caso binario
  - L'attributo può assumere più di due valori.
- Metodo 2: trasformazione in attributi binari
  - Si trasforma una variabile nominale con N valori possibili in una serie di N variabili binarie asimmetriche.
  - La variabile binaria numero i è a 1 se la variabile nominale corrispondente assume il valore i, altrimenti è a 0.

Analisi di raggruppamento

- Un gruppo (cluster) è una collezione di istanze tali che:
  - le istanze dello stesso cluster sono simili tra loro
    - alta somiglianza inter-classe
  - le istanze di cluster diversi sono dissimili
    - bassa somiglianza intra-classe
- Analisi di raggruppamento
  - il processo di raggruppamento delle istanze in cluster
  - apprendimento non supervisionato (le istanze di addestramento non hanno una classe nota a priori)
- La qualità del raggruppamento dipenderà
  - dal parametro scelto per misurare la similarità inter e intra classe

Applicazioni dell'analisi di cluster

- Varie possibilità di utilizzo:
  - come analisi stand-alone
  - come processo preliminare ad altre analisi di dati
    - ad esempio assegnare un'etichetta ad ognuno e poi utilizzare un algoritmo di classificazione
  - come componente integrato di algoritmi per altri tipi di analisi
  - nella fase di pre-elaborazione dei dati
    - per esempio: outlier mining
    - riduzione della numerosità

Un classico algoritmo di clustering: k-means

- K-means ricade nella famiglia degli algoritmi di partizionamento:
  - dai un insieme di n istanze e un numero k, divide le istanze in k partizioni
    - usa tecniche di rilocazione iterativa per spostare le istanze da una partizione all'altra allo scopo di migliorare la qualità del cluster.

k-means (1/2)

- il metodo k-means adotta come centro di gravità di un cluster il suo punto medio.
- si tenta di minimizzare l'errore quadratico:

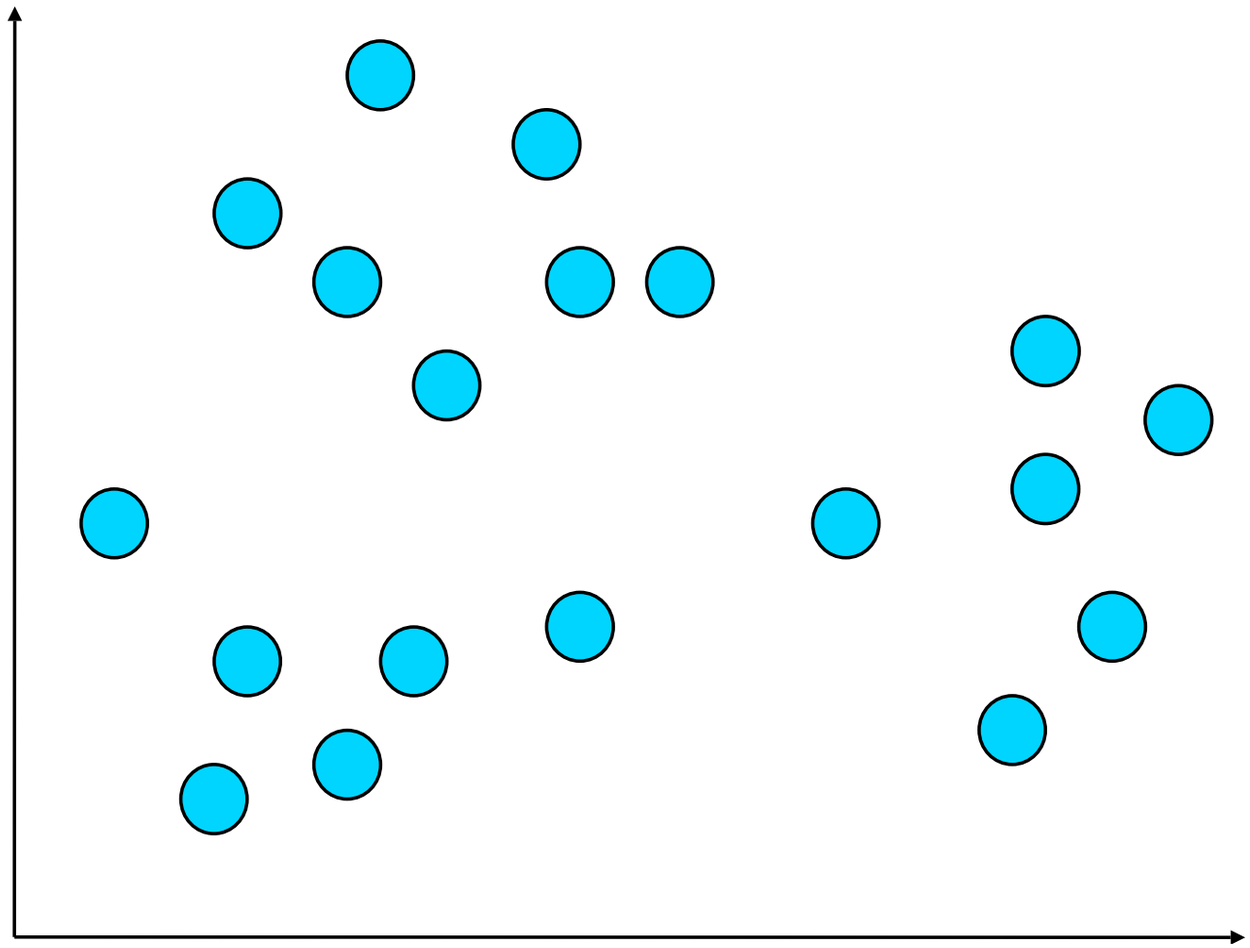
$$Err = \sum_{i=1}^{k} \sum_{p \in C_i} d(p, m_i)^2$$

- dove m_i è il punto medio del cluster C_i

k-means (2/2)

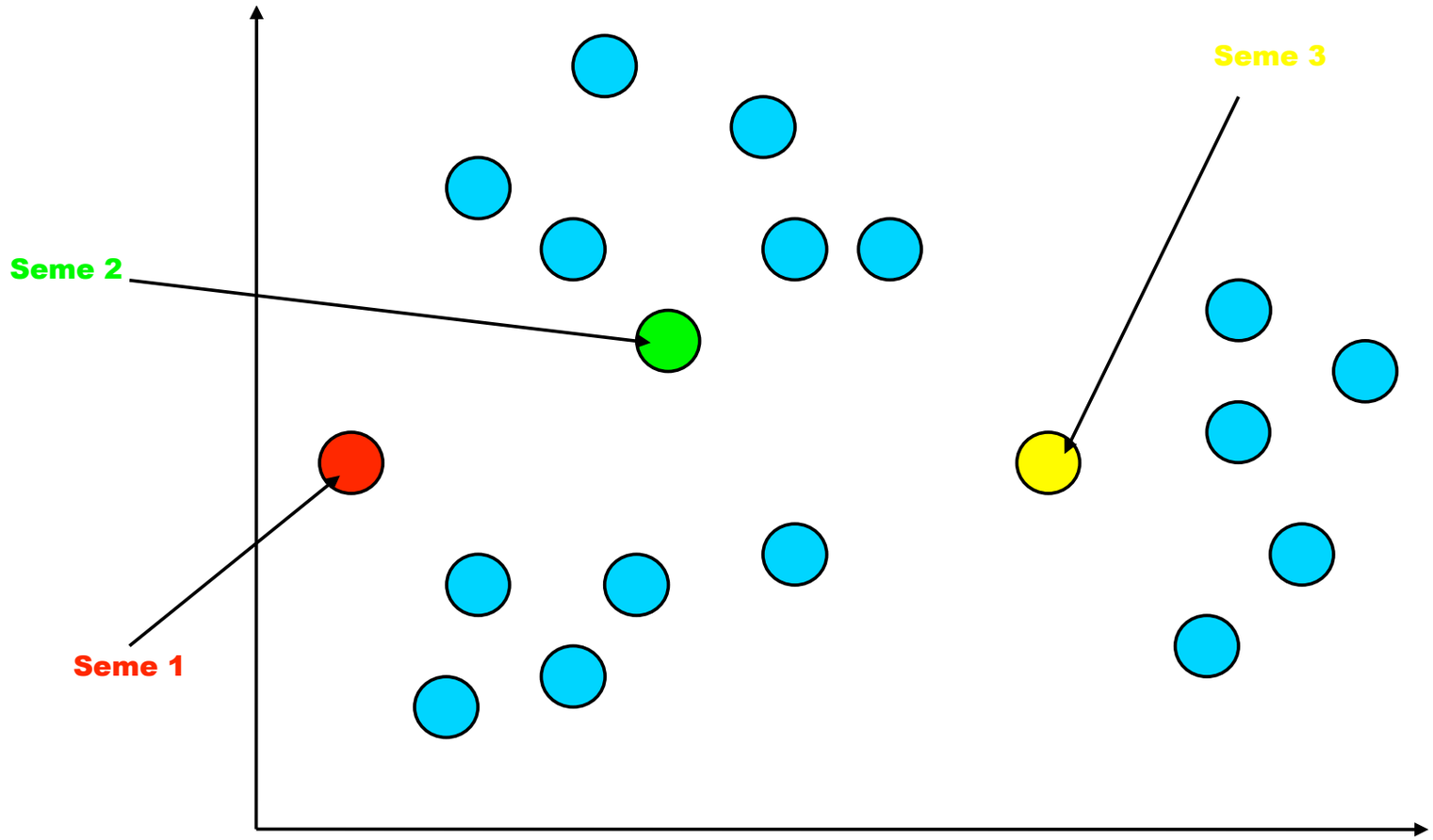
- Dato k, lo schema generale del k-means consta dei seguenti passi:
  - Scegli k oggetti come centri dei vari cluster
    - come avviene la scelta? (random?)
  - Ripeti
    - Assegna ogni oggetto al cluster il cui centro è più vicino
    - Ricalcola i nuovi centri del cluster
  - Finchè non c'è nessun cambiamento

Esempio
spazio di rappresentazione



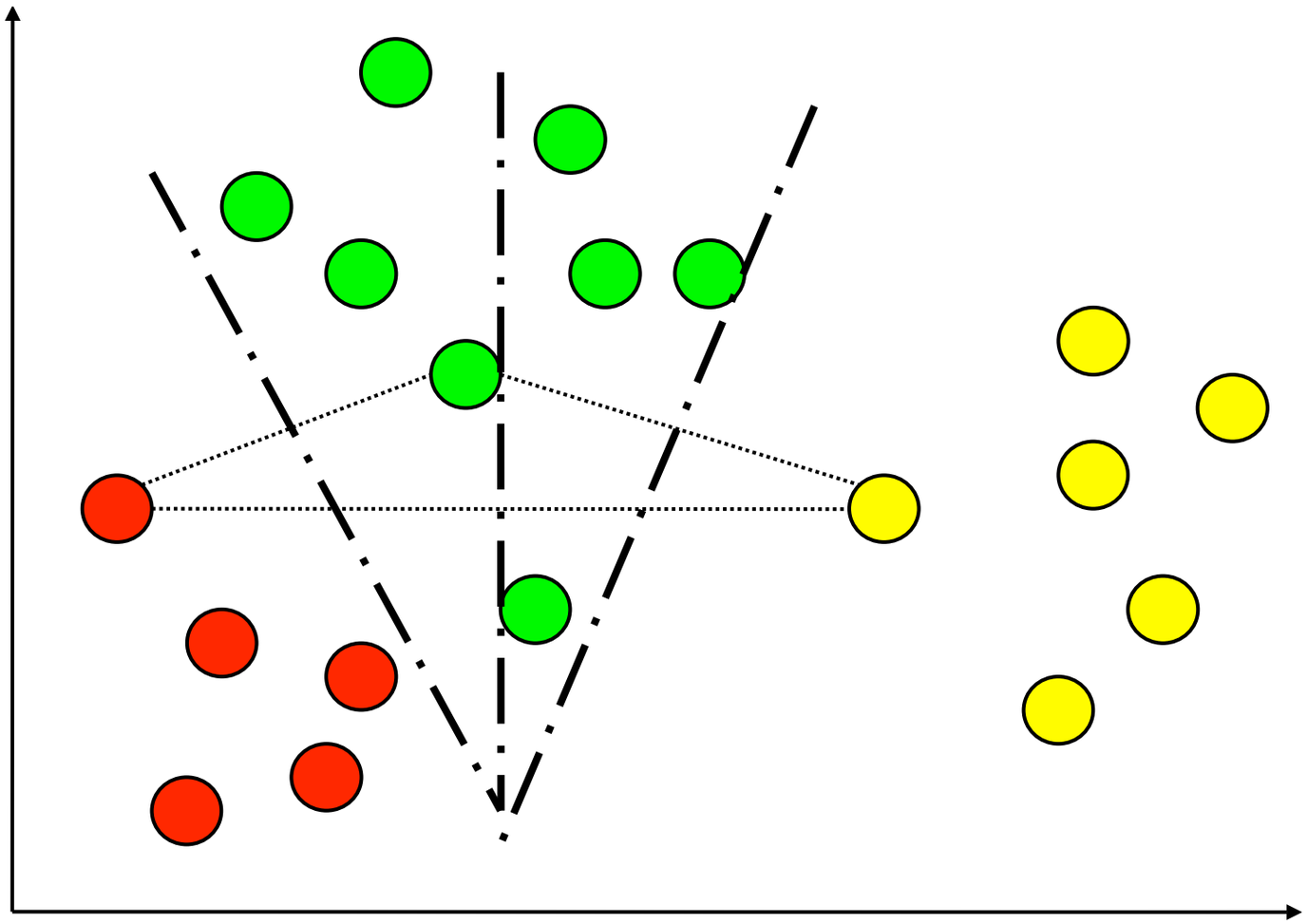
Step 1

scegliamo k=3 e i seguenti semi iniziali



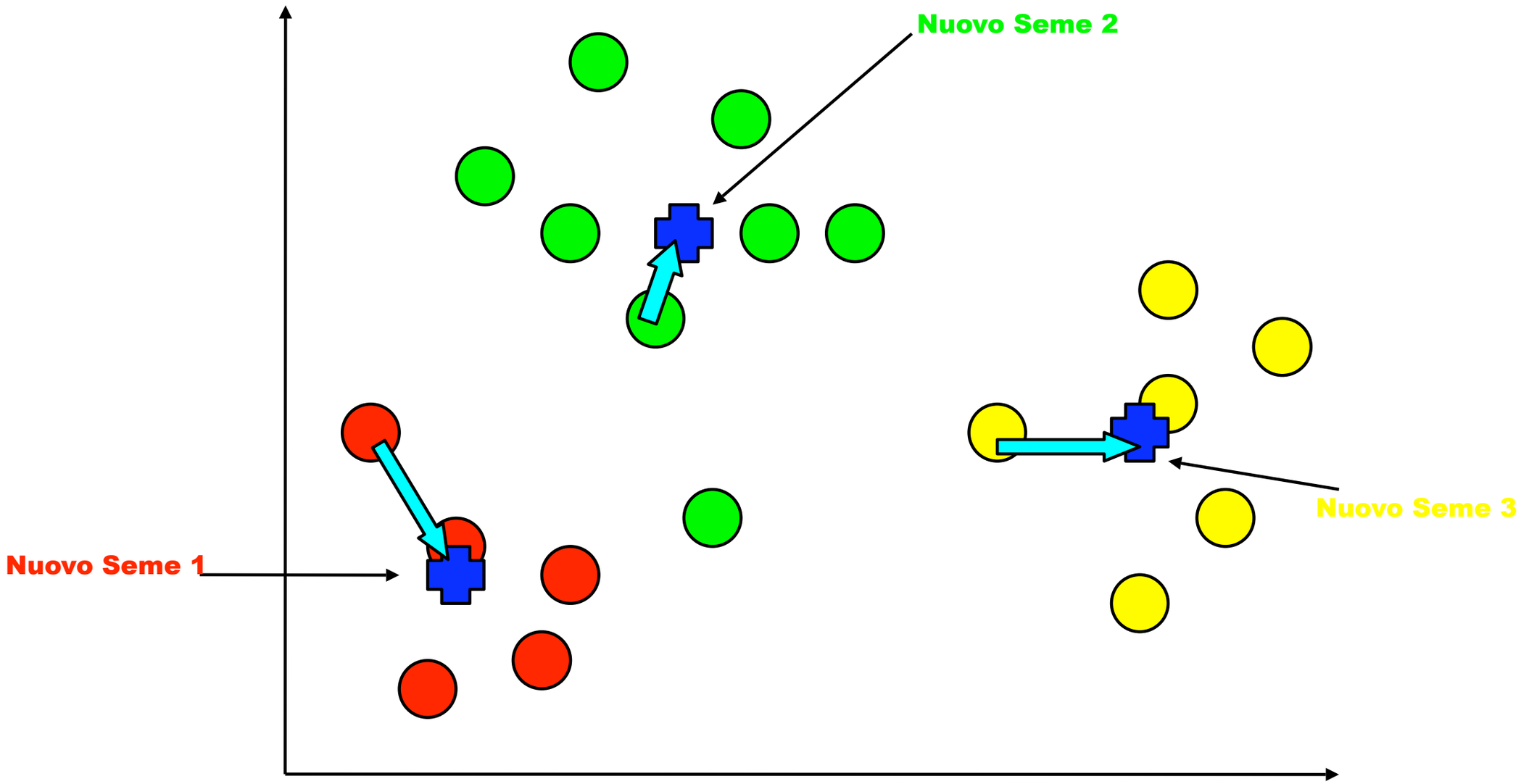
Step 2

assegnamo ogni record al cluster con il centroide pù vicino



Step 3

il passo 2 ha individuato i nuovi cluster. ci calcoliamo i centroidi della nuova configurazione

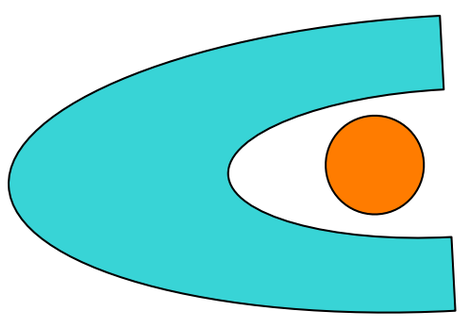


Pregi e difetti (1/2)

- Pregi
  - relativamente efficiente: O(tnk) dove t è il numero di iterazioni. Di solito t e k sono molto minori di n, per cui la complessità di può considerare O(n).
  - Spesso si termina in un ottimo locale. L'ottimo globale si può rincorrere con tecniche standard come il simulated annealing

Pregi e difetti (2/2)

- Difetti:
  - Applicabile solo se è possibile definire il centro.
  - Necessità di specificare k in anticipo.
  - Molto sensibile al rumore e ad outliers.
  - Non adatto per cluster con forme convesse.



Accuratezza di un classificatore

- Una volta ottenuto un classificatore, occorre stimarne l'accuratezza
  - ci serve una misura della prestazione del classificatore
  - per i problemi di classificazione, una misura naturale è il tasso di errore.

Tasso di errore vero

- Supponiamo che
  - I sia l'insieme di tutte le istanze possibili
  - Pr è un distribuzione di probabilità su I
  - c(I) è la classe "vera" di I
  - h(I) è la classe "predetta" di I
- Si definisce allora il tasso di errore vero come:

$$Pr\{i \in I | h(I) \neq c(I)\}$$

Tasso di errore su un campione

- Nella maggior parte dei casi I è infinito e il true error rate non è calcolabile
- Si calcola su un campione finito S

Stima del tasso di errore

- Il modo più semplice di stimare il tasso di errore vero è calcolare il tasso di errore sulle istanze di addestramento
  - si parla di **errore di sostituzione**
- La stima è troppo ottimistica!!!
  - occorre utilizzare due insiemi distinti, uno per l'addestramento e l'altro per il test
- E' importante che l'insieme di test non sia utilizzato in alcun modo nella fase di costruzione del modello.
- Una volta compiuta la stima, si può rimettere l'insieme di test dentro quello di addestramento, e ricalcolare il modello.

Metodo holdout

- Holdout: metodo in cui si divide l'insieme di dati in una parte usata per l'addestramento e una per il testing.
  - tipicamente 1/3 è usato per il test e 2/3 per l'addestramento
- Problema: il campione potrebbe non essere rappresentativo
  - ad esempio, alcune classi poco numerose potrebbero mancare nell'insieme di addestramento

Cross Validation

- Si divide l'insieme di dati in k parti
  - ripeto, per tutti i valori di i da 1 a k:
    - addestro il sistema con tutti i dati tranne quelli della partizione i
      - uso la partizione i per calcolare il tasso di errore
  - calcolo il tasso di errore finale come la media dei k tassi di errore che ho ottenuto in questo modo
  - si parla di k-fold cross validation
- Che valore di k scegliere?
  - da numerosi esperimenti 10 è un buon valore

Leave one out cross validation

- Se si sceglie k=N (numero totale delle istanze), si ha la leave one out cross validation
- Vantaggi:
  - fa il massimo uso dei dati a disposizione
  - non ci sono campionamenti casuali
- Svantaggi:
  - computazionalmente oneroso

Matrice di confusione

- Gli errori commessi dal classificatore possono essere visualizzati in una matrice di confusione

classi vere\predette	*Classe 1*	*Classe 2*	*Classe 3*	*...*	*Classe k*
Classe 1	23	3	4		2
Classe 2	..	17	..		..
Classe 3	..	..	15		..
....					
Classe k	..	..	..		93

- L'errore campionario si ottiene dalla somma dei valori di tutte le celle, esclusa la diagonale
- In generale, si possono pesare in modo diverso le varie celle

Matrici di confusione bidimensionali

	Truth: Yes	Truth: No
System: Yes	a	b
System: No	c	d

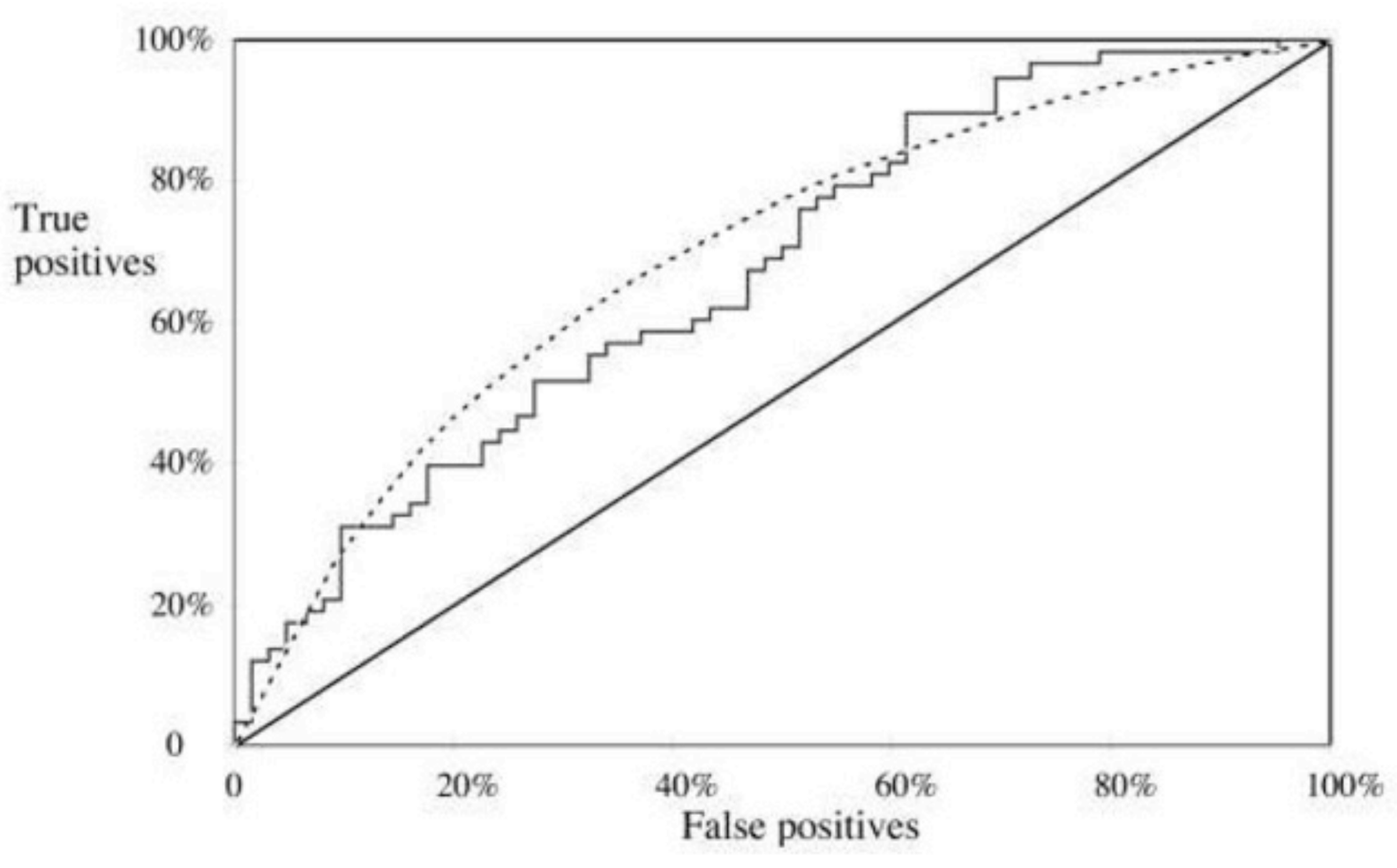
- Misure di performance per la classificazione binaria:
  - error rate = (b+c)/n
  - accuracy = 1 - error rate
  - precision (P) = a/(a+b)
  - recall (R) = a/(a+c)
  - break-even = (P+R)/2
  - F1-measure = 2PR/(P+R)

Curve ROC (1/2)

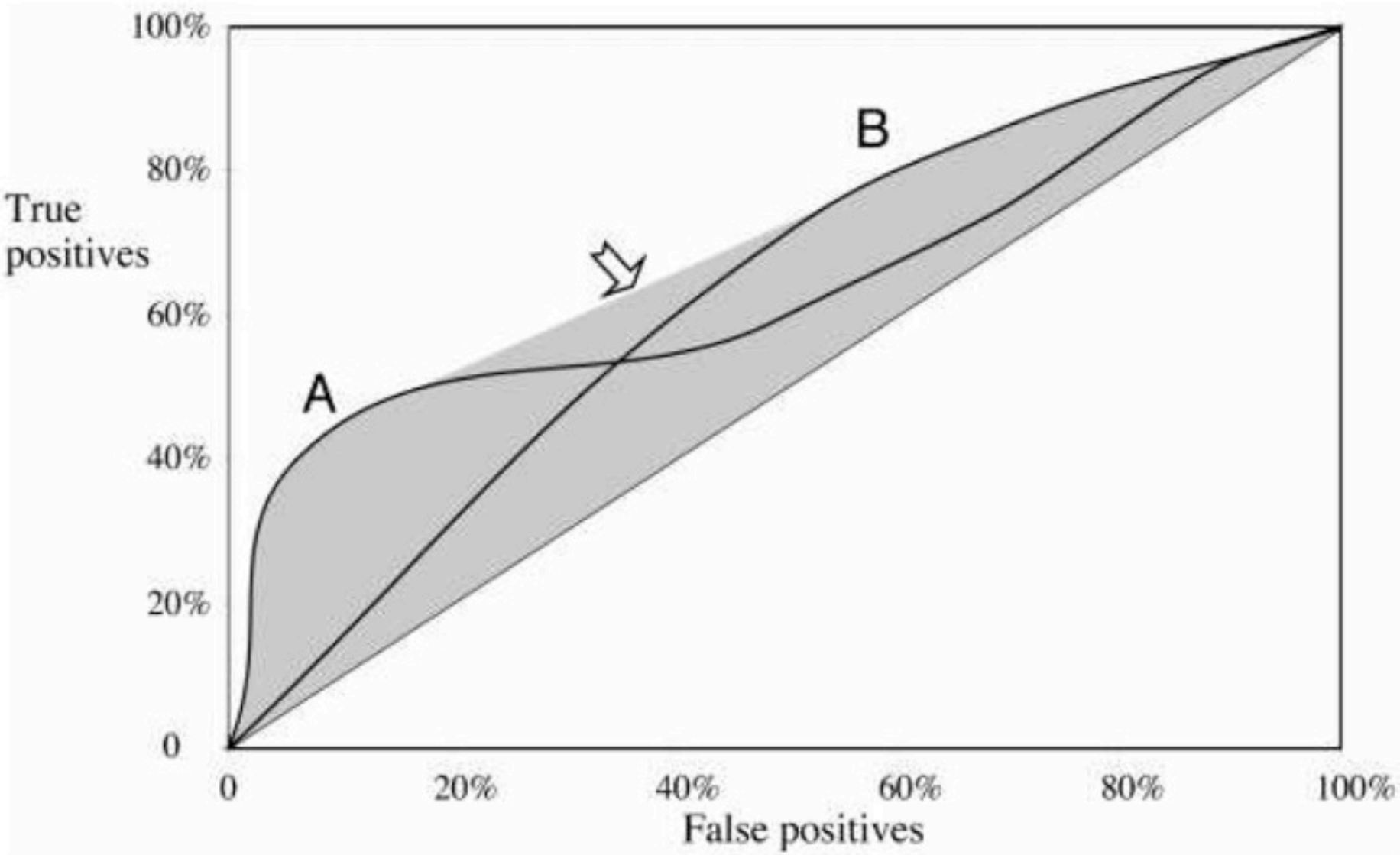
- ROC sta per “Receiver Operating Characteristics”
- Usato nel riconoscimento di segnali per mostrare il trade-off di segnali riconosciuti e numero di falsi allarmi
  - l’asse **X** è la percentuale di falsi positivi.
  - l’asse **Y** è la percentuale di veri positivi.

Curve ROC (2/2)

- Esempio di curva ROC



Curve ROC per due classificatori



Boosting e Bagging

- Sono due metodi standard per combinare dei classificatori C1,...,Ct per produrre un classificatore C* più accurato.
- Analogia con i medici
  - Supponiamo di voler diagnosticare una malattia. Possiamo rivolgerci a vari medici invece che ad uno solo.
  - **Bagging**: prendo le risposte di tutti i medici e considero come diagnosi valida quella prodotta in maggioranza.
  - **Boosting**: peso la diagnosi di ogni medico in base agli errori fatti in precedenza.