Lecture 12

Clustering

Martedì, 30 novembre 2004

Giuseppe Manco

Readings:

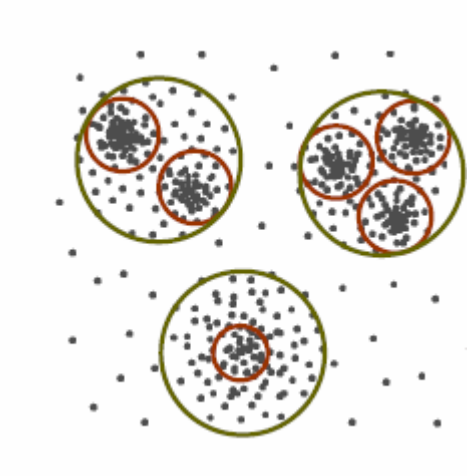
Chapter 8, Han and Kamber

Chapter 14, Hastie , Tibshirani and Friedman

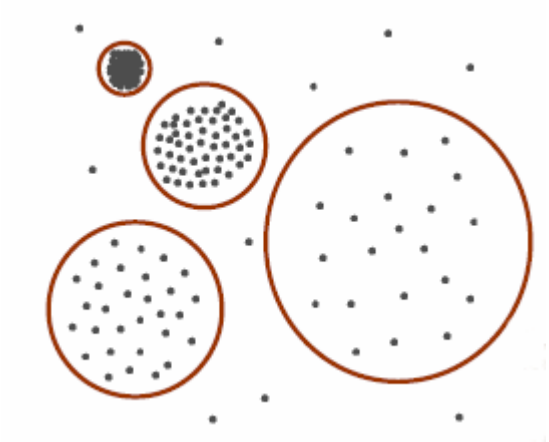
ICAR CNR
Consiglio Nazionale delle Ricerche
Istituto di Calcolo e Reti ad Alte Prestazioni

Clustering gerarchico

- **Il settaggio di parametri in alcune situazioni è complicato**



Cluster gerarchici

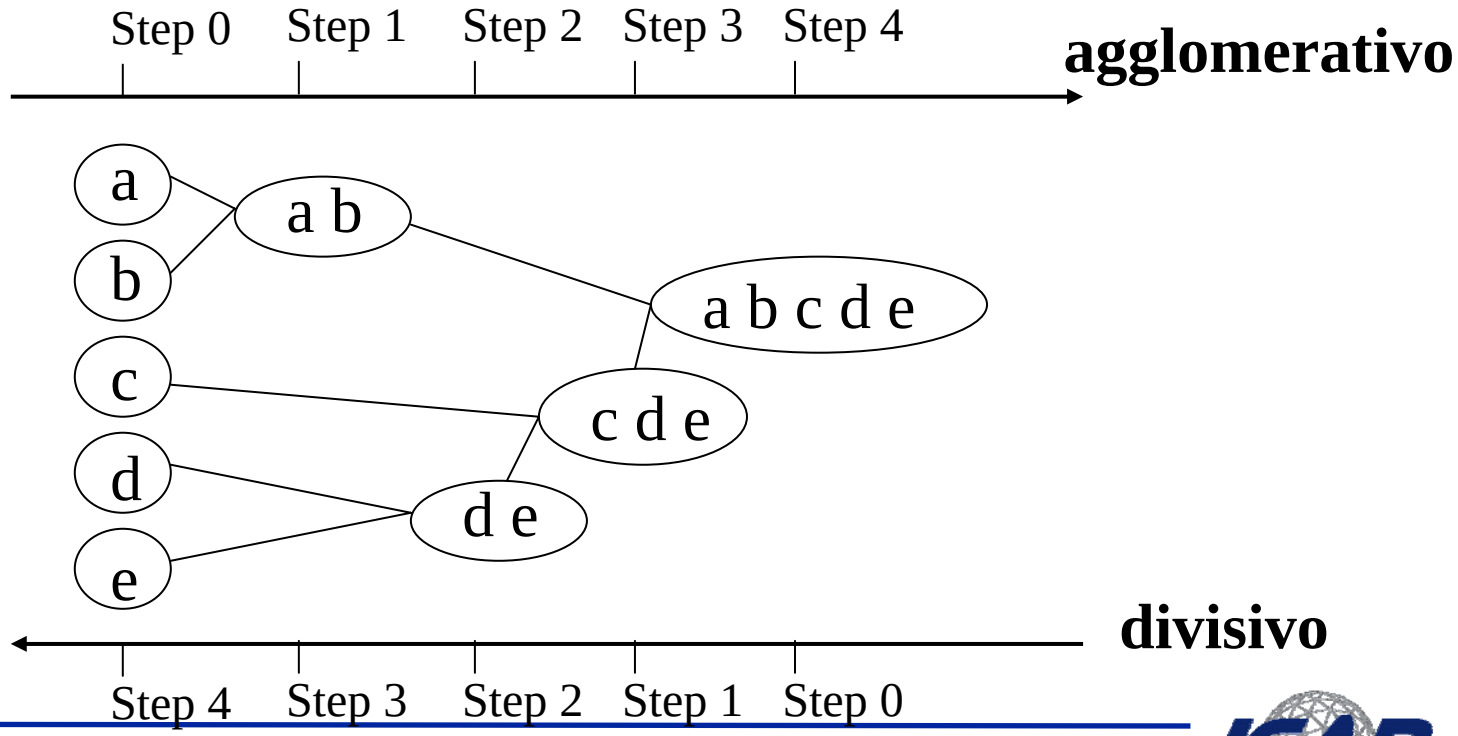


Cluster di densità differente

- **Soluzione: approcci gerarchici al clustering**

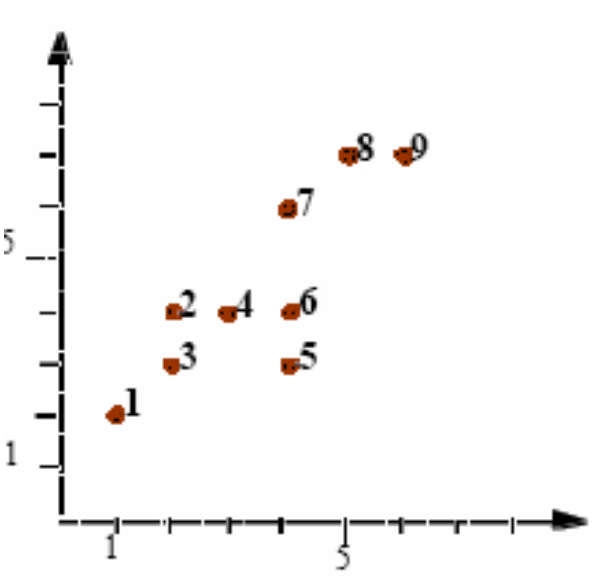
Clustering Gerarchico

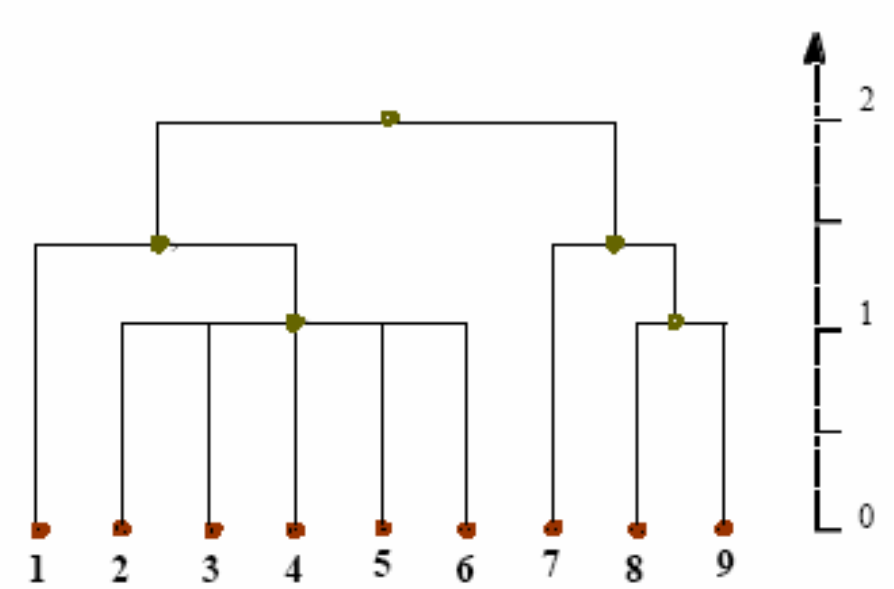
- **Decomposizione gerarchica del dataset in un insieme di clusters annidati**
  - **Il risultato è rappresentato sotto forma di dendogramma**
    - **I nodi rappresentano possibili clusters**
    - **Può essere costruito in maniera top-down o bottom-up**
  - **Basato sull'analisi della matrice di dissimilarità**



Esempio

- **Interpretazione del dendogramma**
  - La radice rappresenta l'intero dataset
  - Una foglia rappresenta un oggetto nel dataset
  - Un nodo interno rappresenta l'unione di tutti gli oggett nel sottoalbero
  - L'altezza di un nodo interno rappresenta la distanza tra i nodi figli





Clustering Gerarchico Agglomerativo

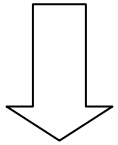
- **Input**
  - **Dataset D**
- **Output**
  - **Partizione S = {C_1, …, C_k}**
    - **N.B.: K non prefissato**

1. S_0 := D
2. Calcola la matrice di dissimilarità M_S
3. DO
4. Identifica la coppia (x,y) che esibisce il minimo valore dist(x,y) in M_s
5. Crea il cluster M = x∪y
6. S_{n+1} := S_n-{x,y} ∪M
7. Ricalcola Ms eliminando le righe relative a x,y e aggiungendo la riga relativa a M (calcolando le distanze di M da tutti gli altri oggetti)
8. UNTIL $|S_{n+1}|$ = 1
9. Scegli S_j che ottimizza il criterio di qualità

Aggiornamento della matrice

- **Quando rimuoviamo x,y e aggiungiamo M**
  - **M_s va aggiornata**

$$\begin{bmatrix} d(x_1,x_1) & d(x_1,x_2) & d(x_1,x_3) & d(x_1,x_4) & d(x_1,x_5) \\ d(x_2,x_1) & d(x_2,x_2) & d(x_2,x_3) & d(x_2,x_4) & d(x_2,x_5) \\ d(x_3,x_1) & d(x_3,x_2) & d(x_3,x_3) & d(x_3,x_4) & d(x_3,x_5) \\ d(x_4,x_1) & d(x_4,x_2) & d(x_4,x_3) & d(x_4,x_4) & d(x_4,x_5) \\ d(x_5,x_1) & d(x_5,x_2) & d(x_5,x_3) & d(x_5,x_4) & d(x_5,x_5) \end{bmatrix}$$



- **Tre varianti:**
  - **Single linkage**
  - **Complete linkage**
  - **Average linkage**

$$\begin{bmatrix} d(x_1,x_1) & d(x_1,\{x_2,x_4\}) & d(x_1,x_3) & d(x_1,x_5) \\ d(\{x_2,x_4\},x_1) & d(\{x_2,x_4\},\{x_2,x_4\}) & d(\{x_2,x_4\},x_3) & d(\{x_2,x_4\},x_5) \\ d(x_3,x_1) & d(x_3,\{x_2,x_4\}) & d(x_3,x_3) & d(x_3,x_5) \\ d(x_5,x_1) & d(x_5,\{x_2,x_4\}) & d(x_5,x_3) & d(x_5,x_5) \end{bmatrix}$$

Distanze inter-cluster

- **Single-Linkage**
  - **Chaining effect**
    - **Se due clusters hanno outliers vicini, vengono comunque fusi**
  - **La compattezza può essere violata**
    - **Clusters con diametri ampi**

$$dist(C_1, C_2) = \min_{x \in C_1, y \in C_2} dist(x, y)$$

- **Complete-linkage**
  - **Situazione duale**
  - **La vicinanza può essere violata**
    - **Clusters con diametri piccoli**

$$dist(C_1, C_2) = \max_{x \in C_1, y \in C_2} dist(x, y)$$

- **Average-linkage**
  - **Relativamente compatti**
  - **Relativamente vicini**

$$dist(C_1, C_2) = \frac{1}{|C_1||C_2|} \sum_{x \in C_1, y \in C_2} dist(x, y)$$

Clustering gerarchico: commenti

- **Maggiore debolezza**
  - **Non scalabili: $O(n^2)$, dove n è la dimensione del dataset**
  - **Ogni assegnamento è fissato una volta per tutte**
- **Approcci differenti al clustering gerarchico**
  - **BIRCH, CURE,CHAMELEON**
  - **Clustering gerarchico basato su densità: OPTICS**

Esercizio: PlayTennis

- Clusterizzare utilizzando DBScan
- Clusterizzare utilizzando un'istanza del gerarchico agglomerativo
  - Con le tre varianti

ID	Outlook	Temp.	Humidity	Windy
A	Sunny	Hot	High	False
B	Sunny	Hot	High	True
C	Overcast	Hot	High	False
D	Rainy	Mild	High	False
E	Rainy	Cool	Normal	False
F	Rainy	Cool	Normal	True
G	Overcast	Cool	Normal	True
H	Sunny	Mild	High	False
I	Sunny	Cool	Normal	False
J	Rainy	Mild	Normal	False
K	Sunny	Mild	Normal	True
L	Overcast	Mild	High	True
M	Overcast	Hot	Normal	False
N	Rainy	Mild	High	True

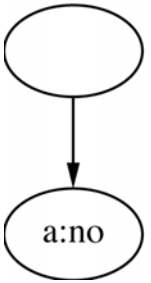
Clustering concettuale

- Approccio euristico (COBWEB/CLASSIT)
- Forma incrementalmente una gerarchia di clusters
- Inizializzazione
  - L'albero consiste del nodo radice (vuoto)
- Iterazione
  - Consideriamo le istanze incrementalmente
  - Aggiorniamo l'albero ad ogni passo
  - Per l'aggiornamento, troviamo la foglia più appropriata per l'istanza in considerazione
  - Può richiedere la ristrutturazione dell'albero
- (Ri)strutturazioni guidate dal concetto di *category utility*

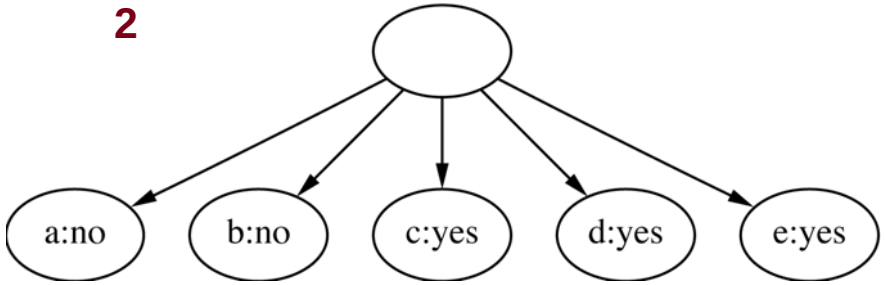
Clustering PlayTennis

ID	Outlook	Temp.	Humidity	Windy
A	Sunny	Hot	High	False
B	Sunny	Hot	High	True
C	Overcast	Hot	High	False
D	Rainy	Mild	High	False
E	Rainy	Cool	Normal	False
F	Rainy	Cool	Normal	True
G	Overcast	Cool	Normal	True
H	Sunny	Mild	High	False
I	Sunny	Cool	Normal	False
J	Rainy	Mild	Normal	False
K	Sunny	Mild	Normal	True
L	Overcast	Mild	High	True
M	Overcast	Hot	Normal	False
N	Rainy	Mild	High	True

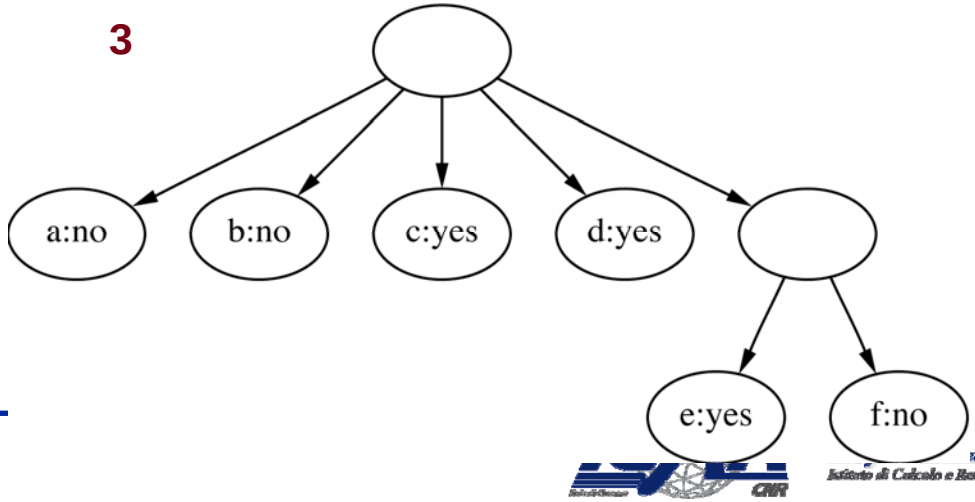
1



2



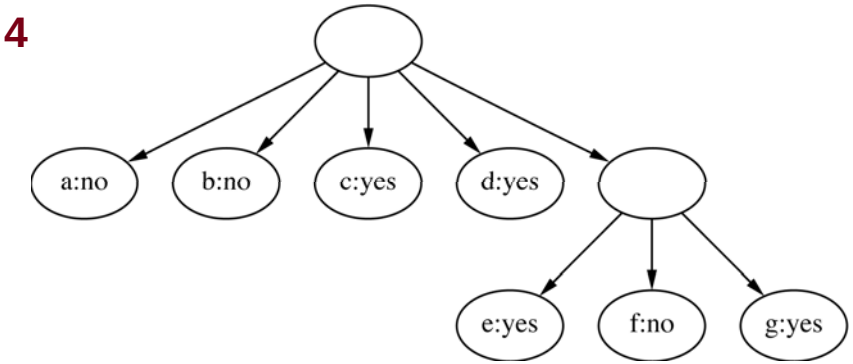
3



Clustering PlayTennis [2]

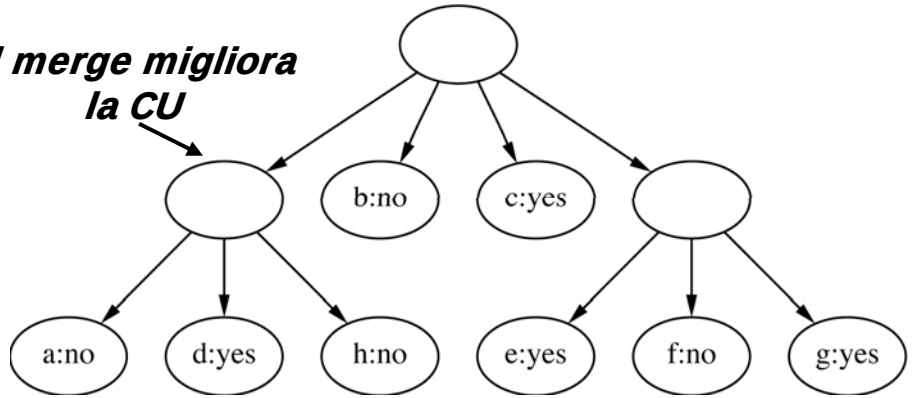
ID	Outlook	Temp.	Humidity	Windy
A	Sunny	Hot	High	False
B	Sunny	Hot	High	True
C	Overcast	Hot	High	False
D	Rainy	Mild	High	False
E	Rainy	Cool	Normal	False
F	Rainy	Cool	Normal	True
G	Overcast	Cool	Normal	True
H	Sunny	Mild	High	False
I	Sunny	Cool	Normal	False
J	Rainy	Mild	Normal	False
K	Sunny	Mild	Normal	True
L	Overcast	Mild	High	True
M	Overcast	Hot	Normal	False
N	Rainy	Mild	High	True

4



5

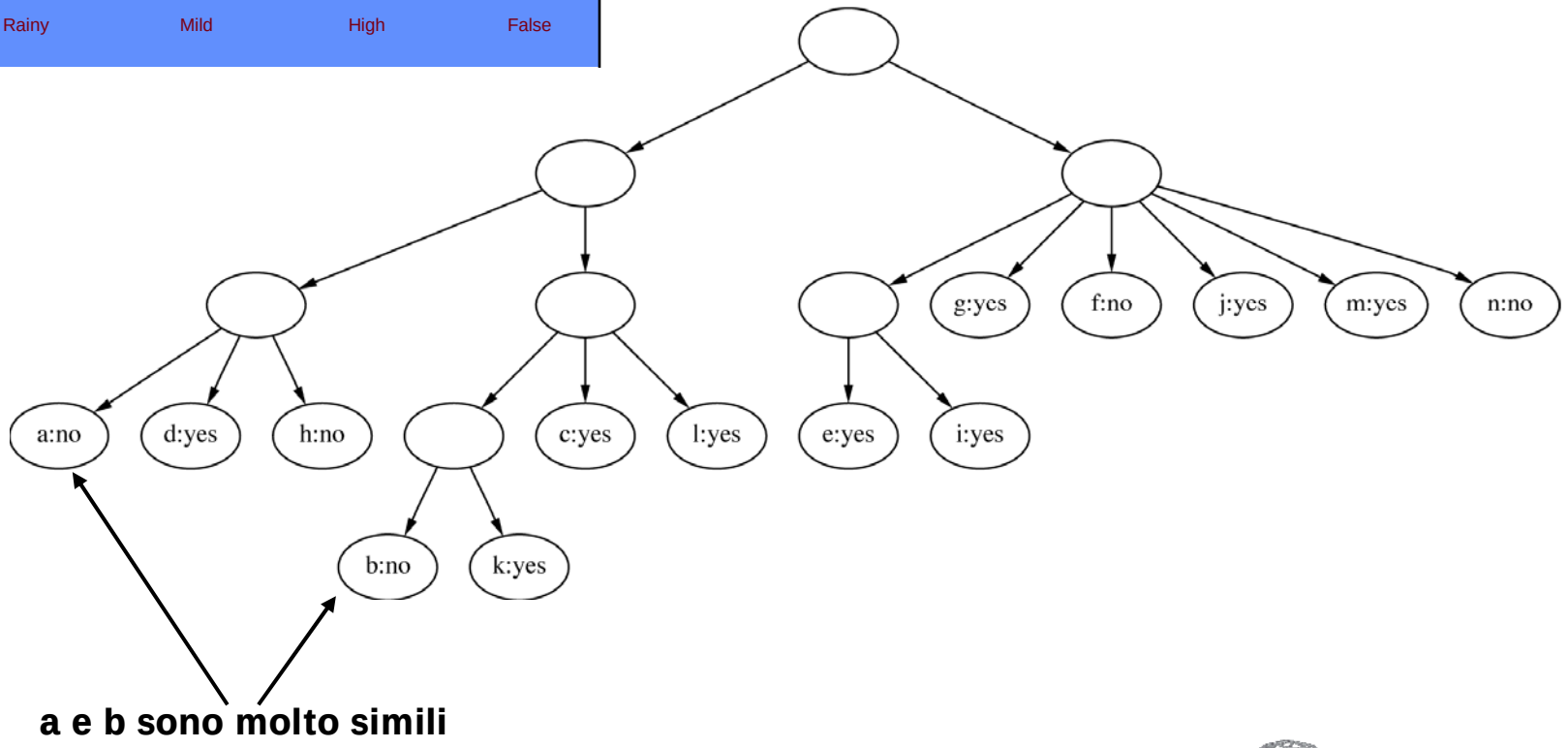
Il merge migliora la CU



Se il merging non aiuta, proviamo a fare splitting

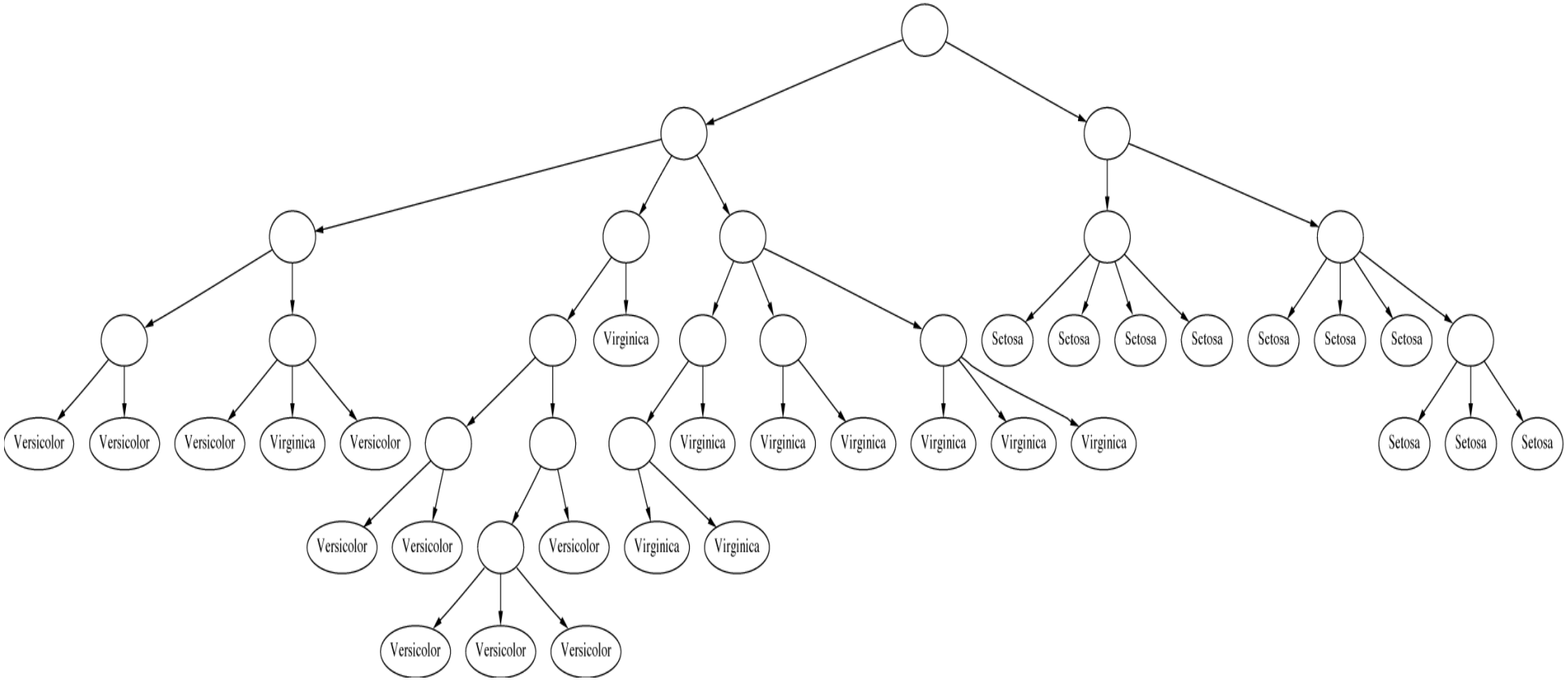
Clustering PlayTennis [3]

ID	Outlook	Temp.	Humidity	Windy
A	Sunny	Hot	High	False
B	Sunny	Hot	High	True
C	Overcast	Hot	High	False
D	Rainy	Mild	High	False

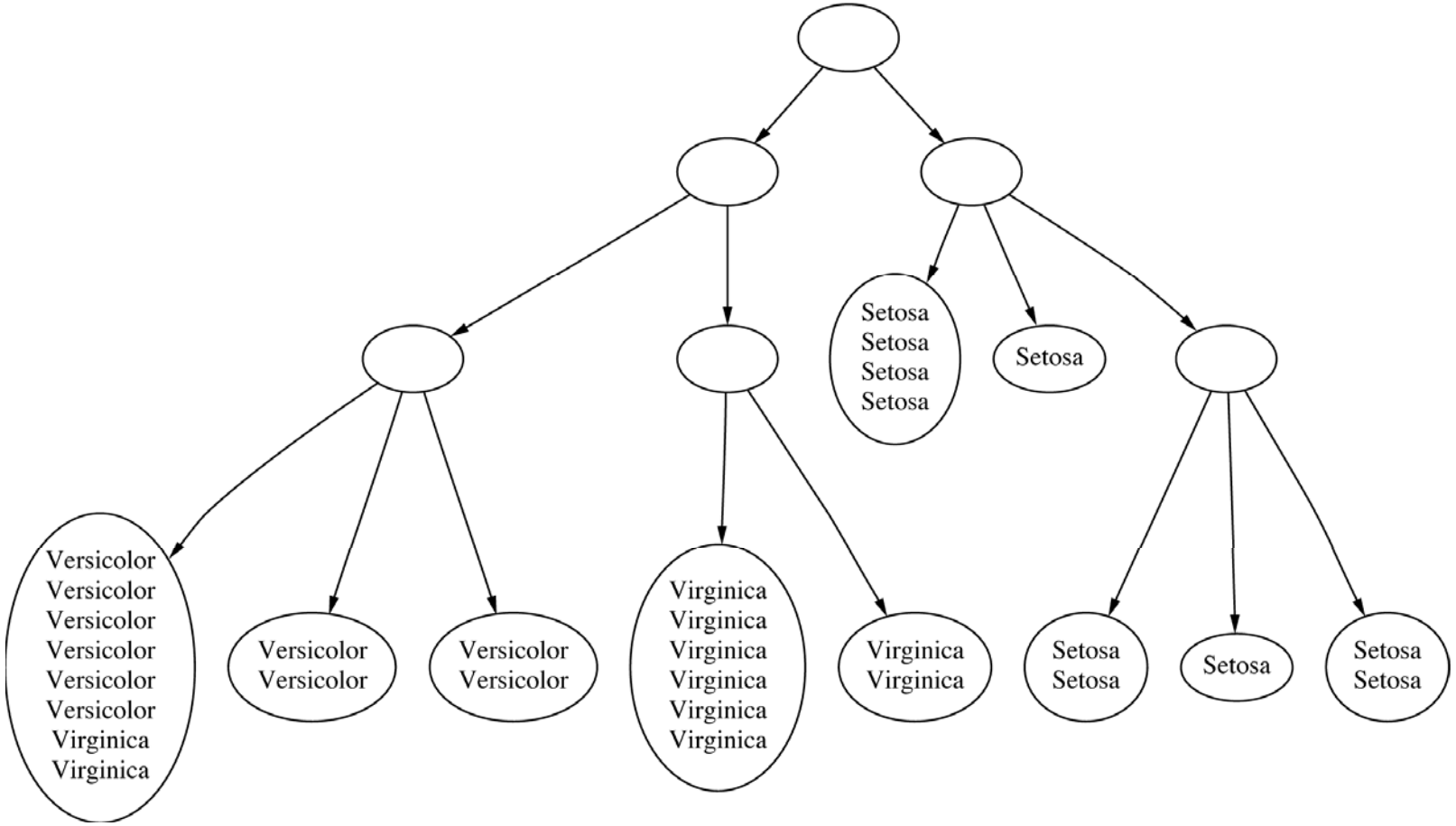


a e b sono molto simili

Esempio: Iris Data



Iris Riorganizzato



Category utility

- **Category utility: Funzione quadratica definita sulle probabilità condizionali:**

$$CU(C_1, C_2, \ldots, C_k) = \frac{\sum_l \Pr[C_l] \sum_i \sum_j (\Pr[a_i = v_{ij} \mid C_l]^2 - \Pr[a_i = v_{ij}]^2)}{k}$$

Overfitting

- **Se ogni istanza costituisce un'unica categoria, il numeratore si massimizza e diventa:**

$$n - \sum_i \sum_j \Pr[a_i = v_{ij}]^2$$

NB *n* è il numero di possibili valori per gli attributi.

- **Il k al denominatore mitiga questo effetto "overfitting"**