

# Clustering

Leggere la sezione 6.6 di Witten e Frank

# Clustering

---

- Raggruppare le istanze di un dominio in gruppi tali che gli oggetti nello stesso gruppo mostrino un alto grado di similarità e gli oggetti in gruppi diversi un alto grado di dissimilarità

# Misure di distanza o dissimilarità

---

- Variano tra 0 e + infinito
- Distanze per punti in  $\mathbb{R}^n$ : *distanza euclidea*

$$d_2(x, y) = \left( \sum_{k=1}^n (x_k - y_k)^2 \right)^{1/2} = \|x - y\|_2$$

- E' un caso particolare, con  $p=2$ , della *metrica di Minkowski*

$$d_p(x, y) = \left( \sum_{k=1}^n (x_k - y_k)^p \right)^{1/p} = \|x - y\|_p$$

# Misure di similarità

---

- Variano tra 0 e 1
- Funzione coseno

$$s_{\cos}(x, y) = \frac{x^T y}{\|x\| \|y\|}$$

- Coefficiente di Dice

$$s_{Dice}(x, y) = \frac{2x^T y}{(\|x\|^2 + \|y\|^2)}$$

- Similarita' esponente

$$s_{\exp}(x, y) = \exp\left(-\|x - y\|^\alpha\right)$$

# Relazione tra le misure di similarita' e dissimilarita'

---

- Un esempio:

$$s(x, y) = \frac{1}{1 + d(x, y)}$$

$$d(x, y) = \frac{1 - s(x, y)}{s(x, y)}$$

# K-means (versione di Forgy)

---

- Si applica a istanze appartenenti a  $R^n$
- Sia  $k$  il numero dei cluster che si vogliono trovare
- 1. Si scelgono  $k$  punti a caso in  $R^n$  come centri dei cluster
- 2. Le istanze sono assegnate al cluster avente il centro piu' vicino
- 3. Si calcola il centroide (la media) dei punti in ogni cluster: questo rappresenta il nuovo centro del cluster

- 4. Si riparte dal passo 2 finche' tutte le istanze non sono assegnate allo stesso cluster in due iterazioni successive
- $$c_j = \frac{\sum_{i=1}^{n_j} x_i}{n_j}$$

# K-means (versione di MacQueen)

---

- In questo caso i centroidi vengono ricalcolati dopo l'assegnazione di ogni pattern e non alla fine di un ciclo di riallocazione:
  1. Si scelgono  $k$  punti a caso in  $R^n$  come centri dei cluster
  2. Si assegnano le istanze ai cluster
  3. Si calcolano i nuovi centroidi dei cluster
  4. Ciascuna istanza e' assegnata al cluster avente il centroide piu' vicino. Dopo ogni assegnamento si deve ricalcolare il centroide del cluster che ha guadagnato l'elemento e di quello che l'ha perso
  5. Si riparte dal passo 4 finche' tutte le istanze non sono assegnate allo stesso cluster in due iterazioni successive

# Risultato del clustering

---

- Il k-means cerca di minimizzare la funzione obiettivo

$$e^2 = \sum_{j=1}^k \sum_{i \in \text{cluster}(j)} d^2(x_i, c_j)$$

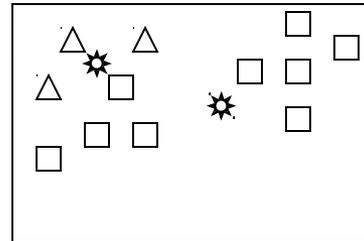
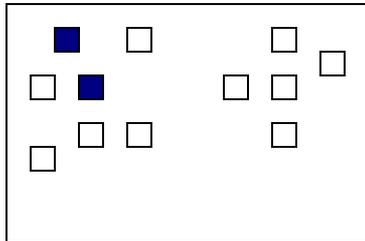
# Scelta dei punti iniziali

---

- Sono possibili varie scelte:
  - Le prime  $k$  istanze nel dataset
  - Etichetta le istanze con i numeri da 1 a  $m$  (numero delle istanze) e scegli quelle con numeri  $m/k, 2m/k, \dots, (k-1)m/k$  e  $m$
  - Scegliere a caso  $k$  istanze
  - Generare  $k$  punti scegliendo a caso i valori di ciascun coordinata nel range della coordinata
  - Genera un partizione del dataset in  $k$  sottoinsiemi mutuamente esclusivi e considera i centroidi dei sottoinsiemi

# Esempio (versione di Forgy)

---

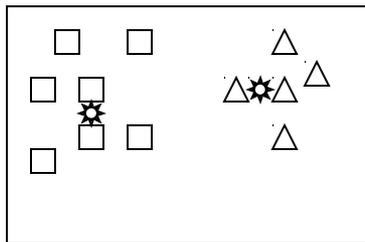


■ Punti iniziali

✱ Centri dei cluster

△ Membri del primo cluster

□ Membri del secondo cluster



Dopo la seconda iterazione

# PAM

---

- PAM (Partitioning Around Medoids)
- Per trovare k cluster, si determina un oggetto rappresentativo per ogni cluster. Questo oggetto, chiamato medoid, e' l'oggetto collocato piu' centralmente nel cluster.
- Una volta selezionati i medoid, gli altri oggetti vengono raggruppati intorno a quello piu' simile a loro
  1. Seleziona arbitrariamente k medoid iniziali
  2. Per ogni oggetto non selezionato, si vede se scambiandolo con uno dei medoid si ottiene un clustering migliore
  3. Continua fino a che nessuno scambio porta a miglioramento

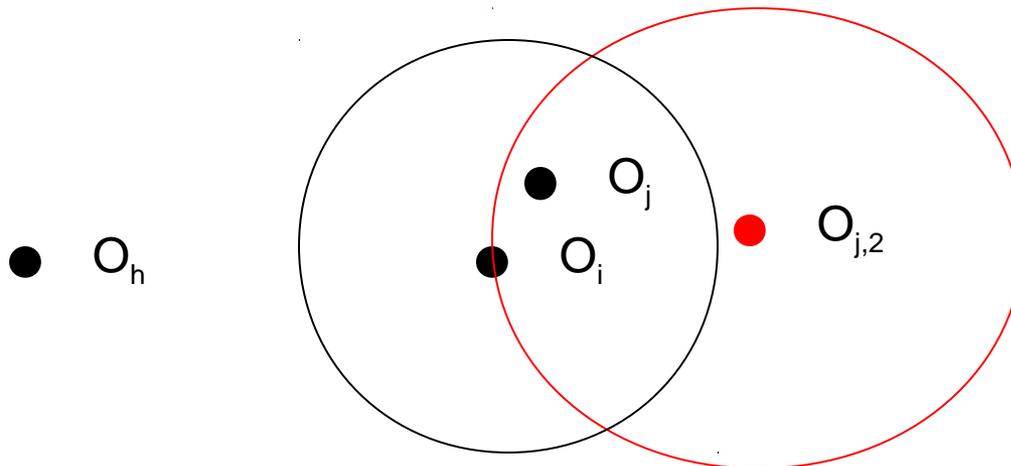
# PAM

---

- Sia  $O_i$  un oggetto selezionato come medoid e  $O_h$  l'oggetto non selezionato considerato per lo scambio
- PAM calcola il costo  $C_{jih}$  dello scambio tra  $O_h$  e  $O_i$  per ogni oggetto non selezionato  $O_j$
- Ci sono 4 casi

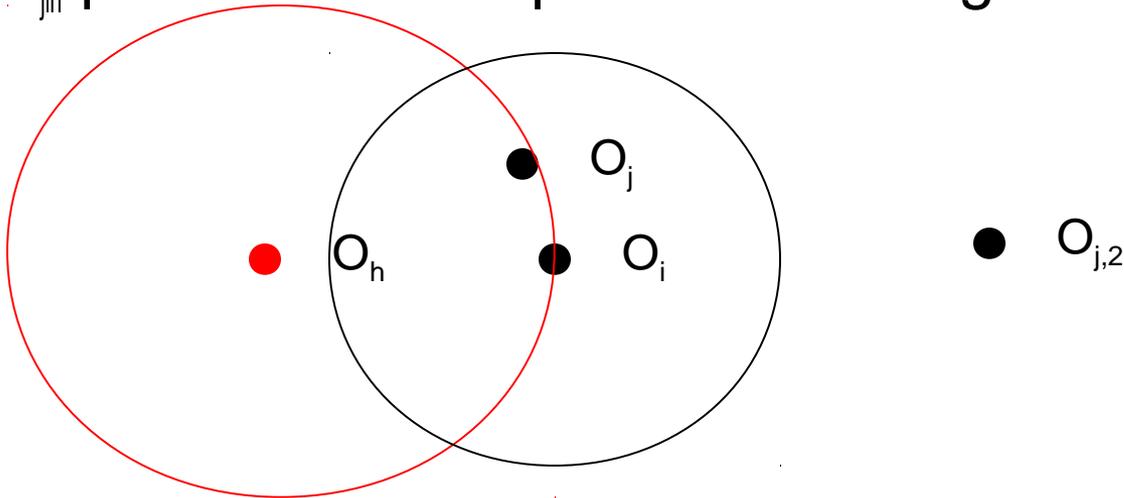
# Caso 1

1.  $O_j$  appartiene al cluster di  $O_i$ 
  - Sia  $O_j$  piu' vicino a  $O_{j,2}$  che a  $O_h$  cioe'  $d(O_j, O_h) \geq d(O_j, O_{j,2})$  dove  $O_{j,2}$  e' il secondo medoid piu' vicino a  $O_j$
  - $O_j$  va nel cluster di  $O_{j,2}$
  - $C_{jih} = d(O_j, O_{j,2}) - d(O_j, O_i)$
  - $C_{jih}$  e' sempre non negativo



## Caso 2

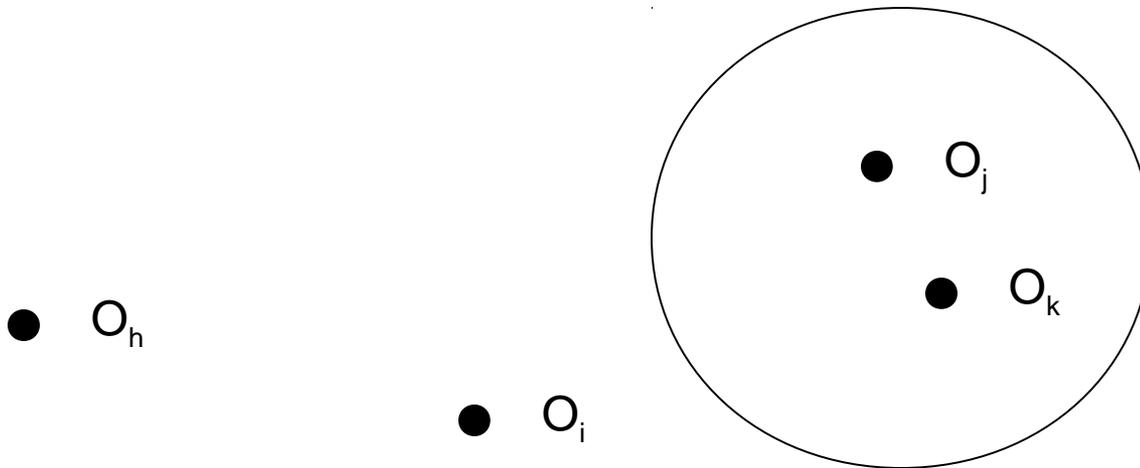
2.  $O_j$  appartiene al cluster di  $O_i$ . Ma  $O_j$  e' piu' distante da  $O_{j,2}$  che da  $O_h$  cioe'  $d(O_j, O_h) < d(O_j, O_{j,2})$
- $O_j$  va nel cluster di  $O_h$
  - $C_{jih} = d(O_j, O_h) - d(O_j, O_i)$
  - $C_{jih}$  puo' essere sia positivo che negativo



## Caso 3

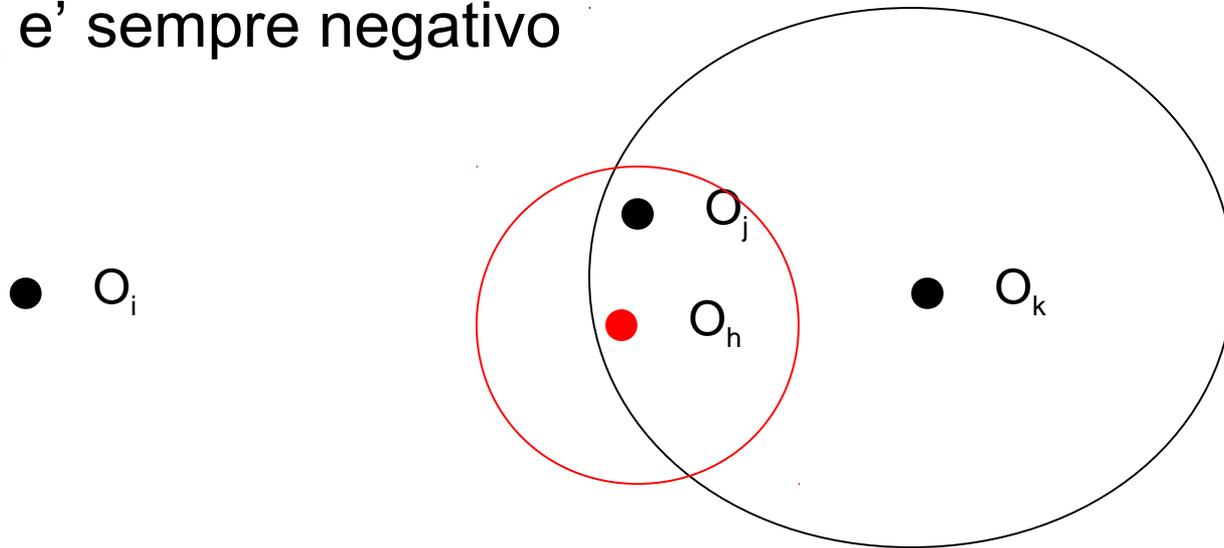
---

3.  $O_j$  appartiene al cluster di un  $O_k$  diverso da  $O_i$ . Sia  $O_j$  piu' vicino a  $O_k$  che a  $O_h$
- $O_j$  rimane nel cluster di  $O_k$
  - $C_{jh} = 0$



# Casi per $C_{jih}$

4.  $O_j$  appartiene al cluster di un  $O_k$ , ma  $O_j$  e' piu' lontano da  $O_k$  che da  $O_h$
- $O_j$  va nel cluster di  $O_h$
  - $C_{jih} = d(O_j, O_h) - d(O_j, O_k)$
  - $C_{jih}$  e' sempre negativo



# Costo totale

---

- Il costo totale di rimpiazzare  $O_i$  con  $O_h$  e'

$$TC_{ih} = \sum_j C_{jih}$$

# Algoritmo

---

1. Seleziona  $k$  oggetti arbitrariamente
2. Calcola  $TC_{ih}$  per tutte le coppie di oggetti  $O_i, O_h$  dove  $O_i$  e' correntemente selezionato e  $O_h$  no
3. Seleziona la coppia  $O_i, O_h$  che corrisponde al  $\min_{O_i, O_h} TC_{ih}$ . Se il minimo e' negativo, sostituisci  $O_i$  con  $O_h$  e torna al passo 2.
4. Altrimenti, assegna ciscun oggetto al suo cluster e termina

# Clustering basato sulla probabilita'

---

- Basato su un modello statistico che si chiama **finite mixture**
- Una mixture e' un insieme di k distribuzioni di probabilita, rappresentanti k cluster, ciascuna delle quali descrive la distribuzione dei valori per i membri di quel cluster
- Ogni distribuzione fornisce la probabilita' che una istanza abbia certi valori per i suoi attributi supponendo che sia noto a quale cluster appartiene
- Ogni istanza appartiene a un solo cluster ma non sappiamo quale
- I cluster non hanno la stessa probabilita'

# Finite mixture

---

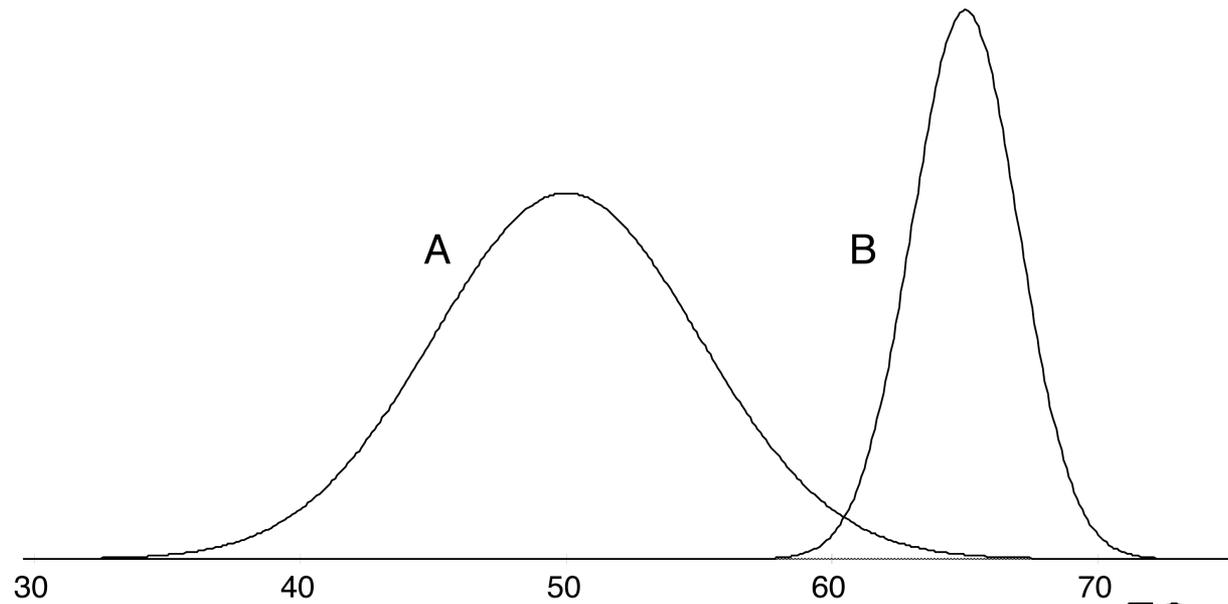
- Il caso piu' semplice e quello in cui si ha una sola variabile reale e due cluster con distribuzione normale.
- Media e varianza della distribuzione sono diverse per i due cluster
- Obiettivo del clustering e' quello di prendere un insieme di istanze e di trovare la media e la varianza delle due distribuzioni normali piu' la distribuzione delle istanze nei cluster

# Esempio

data

A	51	B	62	B	64	A	48	A	39	A	51
A	43	A	47	A	51	B	64	B	62	A	48
B	62	A	52	A	52	A	51	B	64	B	64
B	64	B	64	B	62	B	63	A	52	A	42
A	45	A	51	A	49	A	43	B	63	A	48
A	42	B	65	A	48	B	65	B	64	A	41
A	46	A	48	B	62	B	66	A	48		
A	45	A	49	A	43	B	65	B	64		
A	45	A	46	A	40	A	46	A	48		

model



# Finite mixture problem

---

- Ci sono due cluster A e B con medie e deviazioni standard  $\mu_A, \sigma_A$  e  $\mu_B, \sigma_B$
- I campioni sono presi da A con probabilita'  $p_A$  e da B con probabilita'  $p_B$  con  $p_A + p_B = 1$
- Il risultato e' il dataset mostrato
- Problema di clustering (detto anche finite mixture problem): date le istanze (senza le classi A o B), trovare il cinque parametri  $\mu_A, \sigma_A, \mu_B, \sigma_B$  e  $p_A$  ( $p_B$  puo' essere ricavato da  $p_A$ )

# Calcolo di media e varianza

---

- Se si conoscesse da quale distribuzione viene ogni istanza, si potrebbero calcolare  $\mu_A$ ,  $\sigma_A$ ,  $\mu_B$  e  $\sigma_B$  con le seguenti formule:

$$\mu = \frac{x_1 + x_2 + \dots + x_p}{p}$$
$$\sigma^2 = \frac{(x_1 - \mu)^2 + (x_2 - \mu)^2 + \dots + (x_p - \mu)^2}{p - 1}$$

- $p-1$  a denominatore e' usato perche'  $\sigma$  e' calcolata su un campione invece che sull'intera popolazione (si chiama stimatore senza bias). Se  $p$  e' grande c'e' poca differenza.

# Calcolo di $\Pr(A|x)$

---

- Se conoscessimo i 5 parametri, potremmo trovare le probabilita' che una istanza  $x$  appartenga a ciascuna distribuzione con la seguente formula

$$\Pr(A|x) = \frac{\Pr(x|A)\Pr(A)}{\Pr(x)} = \frac{f(x; \mu_A, \sigma_A)p_A}{\Pr(x)}$$

dove  $f(x; \mu, \sigma)$  e' la distribuzione normale:

$$f(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

- $\Pr(x)$  non e' noto

# Calcolo di $\Pr(A|x)$

---

- Allo stesso modo possiamo trovare il numeratore di  $\Pr(B|x)$ .
- Sappiamo che  $\Pr(A|x)+\Pr(B|x)=1$  quindi

$$\frac{f(x; \mu_A, \sigma_A)p_A + f(x; \mu_B, \sigma_B)p_B}{\Pr(x)} = 1$$

$$f(x; \mu_A, \sigma_A)p_A + f(x; \mu_B, \sigma_B)p_B = \Pr(x)$$

Quindi:

$$\Pr(A|x) = \frac{f(x; \mu_A, \sigma_A)p_A}{f(x; \mu_A, \sigma_A)p_A + f(x; \mu_B, \sigma_B)p_B}$$

# Algoritmo EM

---

- Il problema e' che non conosciamo ne' i 5 parametri ne' l'appartenenza delle istanze ai cluster
- Percio' adottiamo una procedura simile a quella del k-means:
  - Cominciamo con valori scelti a caso per i 5 parametri
  - Calcoliamo le probabilita' dei due cluster per ogni istanza usando i valori attuali dei parametri (passo di Expectation)
  - Usiamo la distribuzione delle istanze nei cluster per stimare i parametri (passo di Maximization (della verosimiglianza dei dati))
  - Ripartiamo dal passo di Expectation

# Stima dei parametri

---

- In EM non abbiamo l'appartenenza netta delle istanze ai vari cluster ma solo una probabilita'. Percio' le formule per la stima dei parametri diventano:

$$\mu_A = \frac{w_1 x_1 + w_2 x_2 + \dots + w_p x_p}{w_1 + w_2 + \dots + w_p}$$

$$\sigma_A^2 = \frac{w_1 (x_1 - \mu)^2 + w_2 (x_2 - \mu)^2 + \dots + w_p (x_p - \mu)^2}{w_1 + w_2 + \dots + w_p}$$

- Dove  $w_i$  e' la probabilita' che  $x_i$  appartenga ad A e dove gli  $x_i$  sono tutti gli  $x$ , non solo quelli che appartengono ad A

# Stima di $p_A$

---

$$p_A = \frac{w_1 + w_2 + \dots + w_p}{p}$$

$$p_B = 1 - p_A$$

- Dove  $w_i$  è la probabilità che  $x_i$  appartenga ad A e  $p$  è il numero totale di campioni

# Quando terminare?

---

- K-means si ferma quando le classi delle istanze non variano piu'
- EM converge verso un punto fisso ma non ci arriva mai effettivamente
- Possiamo pero' capire quanto e' vicino calcolando la **verosimiglianza (likelihood)** globale che i dati derivino da questo dataset, dati i valori dei 5 parametri
- La verosimiglianza globale si ottiene in questo modo

$$\prod_i (p_A \Pr(x_i | A) + p_B \Pr(x_i | B)) = \prod_i (\Pr(x_i, A) + \Pr(x_i, B)) = \prod_i \Pr(x_i)$$

# Quando terminare

---

- La verosimiglianza globale e' una misura della "bonta'" del clustering e aumenta a ogni iterazione dell'algoritmo EM
- Attenzione: per  $\Pr(x_i|A)$  si usa la funzione di distribuzione normale  $f(x; \mu_A, \sigma_A)$  che non e' una probabilita' ma una densita' di probabilita'.
- Quindi la verosimiglianza non e' una probabilita' ma comunque da' una misura della bonta' del clustering
- In pratica, viene calcolato il logaritmo della verosimiglianza che trasforma i prodotti in somme
- Per esempio, un criterio per fermarsi potrebbe essere: ci si ferma quando la differenza tra due valori successivi della verosimiglianza e' inferiore a  $10^{-10}$  per dieci iterazioni successive.

# Proprieta' dell' algoritmo EM

---

- Anche se e' garantito che EM converga a un massimo, non e' detto che sia un massimo globale, potrebbe essere un massimo locale.
- Per questo la procedura deve essere ripetuta diverse volte, partendo da diversi valori iniziali dei parametri
- La verosimiglianza globale viene poi usata per confrontare le diverse soluzioni ottenute e prendere quella con la verosimiglianza piu' alta

# Estensioni del mixture model

---

- Usare piu' di due distribuzioni: facile se il numero di distribuzioni e' dato come input
- Piu' di un attributo numerico: facile se si assume l'indipendenza tra gli attributi:
  - Le probabilita' di ciascuna classe dato un attributo sono moltiplicate insieme per ottenere la probabilita' congiunta della classe data l'istanza
- Coppie di attributi numerici correlati: difficile
  - I due attributi possono essere descritti da una distribuzione normale bivariata
  - Ha un media ma invece di due varianze ha una matrice di covarianza simmetrica con 4 parametri
  - Ci sono tecniche standard per stimare le probabilita' delle classi delle istanze e per stimare la media e la matrice di covarianza date le istanze e le loro probabilita' delle classi

# Estensioni del mixture model

---

- Più di due attributi correlati: difficile
  - Si usano ancora distribuzioni multivariate
  - Il numero dei parametri aumenta con il quadrato del numero di attributi
    - $n$  attributi indipendenti:  $2n$  parametri
    - $n$  attributi correlati:  $n+n(n+1)/2$  parametri (per la simmetria della matrice di covarianza)
  - Il numero di parametri causa overfitting

# Estensioni del mixture model

---

- Attributi nominali non correlati: un attributo con  $v$  valori possibili e' descritto da  $v$  numeri per ogni cluster ( $\Pr(x_i=v_j|\text{cluster}_h)$ ) che rappresentano la probabilita' di ogni valore
  - Passo di expectation:
    - si calcola  $\Pr(\text{cluster}_h|x_i=v_j)$  usando il teorema di Bayes
    - Si calcola  $\Pr(\text{cluster}_h|x)$  moltiplicando i vari  $\Pr(\text{cluster}_h|x_i=v_j)$  per ogni attributo (assunzione di indipendenza)
  - Passo di maximization
    - Si calcolano i vari  $\Pr(x_i=v_j|\text{cluster}_h)$  dai dati

# Estensioni del mixture model

---

- nel passo di maximization ci sono due problemi:
  - Non conosciamo l'appartenenza di una istanza ad una classe in maniera netta ma solo probabilistica (si considerano dei pesi)
  - Alcune stime di probabilita' possono risultare nulle. Queste stime annullano le  $\Pr(\text{cluster}_h | x_i=v_j)$  e quindi anche  $\Pr(\text{cluster}_h | x)$ 
    - Per questo si usa la stima di Laplace

# Estensioni del mixture model

---

- Due attributi nominali correlati: se hanno  $v_1$  e  $v_2$  possibili valori, possiamo sostituirli con un solo attributo covariante con  $v_1 v_2$  valori
  - Il numero di parametri aumenta esponenzialmente con l'aumentare del numero di attributi correlati
- Presenza di attributi sia numerici che nominali: non ci sono problemi se nessun attributo numerico è correlato con quelli nominali. In caso contrario il problema è difficile e non ce ne occupiamo

# Estensioni del mixture model

---

- Valori nominali mancanti: due possibilita':
  - Non si considerano ne' nel passo di expectation (non si moltiplica per  $\Pr(x_i=v_j|\text{cluster}_h)$ ) ne' nel passo di maximization
  - Si trattano come un valore in piu'
- Valori numerici mancanti: stesse possibilita' che per i valori nominali

# Estensioni del mixture model

---

- Al posto della distribuzione normale, altre distribuzioni possono essere usate:
  - Attributi numerici:
    - Se c'è un valore minimo (ad es. peso) e' meglio la distribuzione "log-normale"
    - Se c'è sia un minimo che un massimo e' meglio la distribuzione "log-odds"
    - Se sono conteggi interi invece che valori reali e' meglio la distribuzione di "Poisson"
- Distribuzioni diverse possono essere usate per attributi diversi

# Autoclass

---

- Autoclass e' un sistema di clustering sviluppato dalla NASA che utilizza l'algoritmo EM
- Prova diversi numeri di cluster e differenti distribuzioni di probabilita' per gli attributi numerici
- L'algoritmo e' computazionalmente pesante, per questo si definisce a priori un tempo di esecuzione e l'algoritmo termina una volta esaurito il tempo
  - Con piu' tempo possiamo ottenere risultati migliori

# Bibliografia

---

Ian Witten, Eibe Frank

Data Mining: Practical Machine Learning Tools and  
Techniques (Second Edition)

Morgan Kaufmann Publishers, 2005, ISBN 0-12-  
088407-0

(disponibile in biblioteca)