Cenni di apprendimento in Reti Bayesiane

Esistono diverse varianti di compiti di apprendimento

- La struttura della rete può essere *nota* o *sconosciuta*
- Esempi di apprendimento possono fornire dati per *tutte* le variabili della rete, o solo per *alcune*

Se si conosce la struttura e tutte le variabili sono “osservabili”

- allora l’allenamento è in sostanza equivalente a quello del classificatore Naive di Bayes

Cenni di apprendimento in Reti Bayesiane

Supponiamo che la struttura sia nota e le variabili parzialmente osservabili

per esempio, si hanno dati per *ForestFire, Storm, BusTourGroup, Thunder*, ma non per *Lightning, Campfire*...

- situazione simile al caso di reti neurali con unità nascoste: unità di output osservabili, unità nascoste non osservabili
- in effetti si possono apprendere le tabelle di probabilità condizionale per le variabili non osservabili tramite ascesa di gradiente
- approccio Maximum Likehood (ML): massimizzare $P(D|h)$

Ascesa di Gradiente ML

Come al solito è più semplice massimizzare $\ln P(D|h)$

Denotiamo con w_{ijk} una generica entry della tabella di probabilità condizionale per la variabile Y_i nella rete

$$w_{ijk} = P(Y_i = y_{ij} | Genitori(Y_i) = \text{la lista } u_{ik} \text{ di valori})$$

ad esempio, se $Y_i = Campfire$, allora u_{ik} potrebbe essere $\langle Storm = T, BusTourGroup = F \rangle$

Calcolo del gradiente:

$$
\begin{aligned}
\frac{\partial \ln P(D|h)}{\partial w_{ijk}} &= \frac{\partial}{\partial w_{ijk}} \ln \prod_{d \in D} P(d|h) \quad \text{(esempi estratti indipendentemente)} \\
&= \sum_{d \in D} \frac{\partial \ln P(d|h)}{\partial w_{ijk}} \\
&= \sum_{d \in D} \frac{1}{P(d|h)} \frac{\partial \ln P(d|h)}{\partial w_{ijk}} \\
&= \sum_{d \in D} \frac{1}{P(d|h)} \frac{\partial}{\partial w_{ijk}} \sum_{q,t} P(d|y_{iq}, u_{it}, h) P(y_{iq}, u_{it}, h) \\
&= \sum_{d \in D} \frac{1}{P(d|h)} \frac{\partial}{\partial w_{ijk}} \sum_{q,t} P(d|y_{iq}, u_{it}, h) \underbrace{P(y_{iq}|u_{it}, h) P(u_{it}|h)}_{\text{regola del prodotto}}
\end{aligned}
$$

Poiché $w_{ijk} = P(y_{iq}|u_{it})$, solo il termine della sommatoria per cui $q = j$ e $t = k$ il gradiente è non nullo

$$\begin{aligned}
\frac{\partial \ln P(D|h)}{\partial w_{ijk}} &= \sum_{d \in D} \frac{1}{P(d|h)} \frac{\partial}{\partial w_{ijk}} P(d|y_{ij}, u_{ik}, h) P(y_{ij}|u_{ik}, h) P(u_{ik}|h) \\
&= \sum_{d \in D} \frac{1}{P(d|h)} \frac{\partial}{\partial w_{ijk}} P(d|y_{ij}, u_{ik}, h) w_{ijk} P(u_{ik}|h) \\
&= \sum_{d \in D} \frac{1}{P(d|h)} P(d|y_{ij}, u_{ik}, h) P(u_{ik}|h)
\end{aligned}$$

Per il teorema di Bayes abbiamo $P(d|y_{ij}, u_{ik}, h) = \frac{P(y_{ij}, u_{ik}|d,h)P(d|h)}{P(y_{ij}, u_{ik}|h)}$ e quindi

$$\begin{aligned}
\frac{\partial \ln P(D|h)}{\partial w_{ijk}} &= \sum_{d \in D} \frac{1}{P(d|h)} \frac{P(y_{ij}, u_{ik}|d, h) P(d|h) P(u_{ik}|h)}{P(y_{ij}, u_{ik}|h)} \\
&= \sum_{d \in D} \frac{P(y_{ij}, u_{ik}|d, h) P(u_{ik}|h)}{P(y_{ij}, u_{ik}|h)} \\
&= \sum_{d \in D} \frac{P(y_{ij}, u_{ik}|d, h)}{P(y_{ij}|u_{ik}, h)} = \sum_{d \in D} \frac{P(y_{ij}, u_{ik}|d, h)}{w_{ijk}}
\end{aligned}$$

Ascesa di Gradiente ML

Eseguire ascesa di gradiente ripetendo le seguenti operazioni

1. aggiornare tutti i w_{ijk} usando i dati di apprendimento D:

$$w_{ijk} \leftarrow w_{ijk} + \eta \sum_{d \in D} \frac{P(y_{ij}, u_{ik}|d, h)}{w_{ijk}}$$

2. poi, normalizzare i w_{ijk} in modo da assicurare

- $\sum_j w_{ijk} = 1$
- $0 \leq w_{ijk} \leq 1$

Ancora sull'Apprendimento di Reti Bayesiane

Si può usare anche l'algoritmo **E**xpectation **M**aximization (EM)

Ripetere:

1. Calcolare le probabilità di variabili non osservabili, assumendo h corrente
2. Calcolare nuovi w_{ijk} (cioè una nuova ipotesi h') che massimizzano $E[\ln P(D|h)]$, dove D include ora sia i dati osservati che (le probabilità calcolate) di variabili non osservabili

Se la struttura non è conosciuta...

- algoritmi basati su ricerca greedy per aggiungere/togliere archi e nodi

Algoritmo Expectation Maximization (EM)

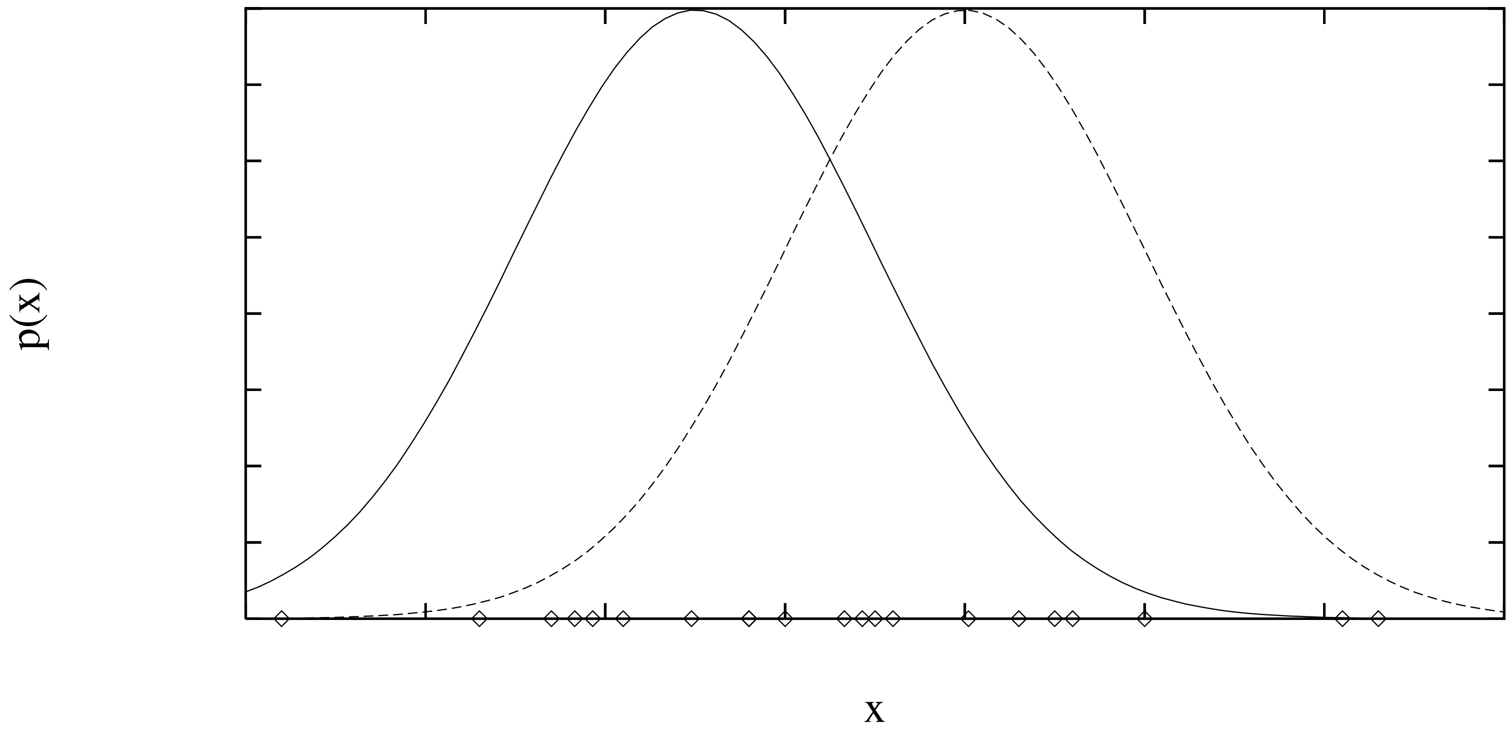
Quando usarlo:

- dati solo parzialmente osservabili
- clustering non supervisionato (valore target non osservabile)
- apprendimento supervisionato (alcuni attributi con valori mancanti)

Alcuni esempi:

- apprendimento Reti Bayesiane
- AUTOCLASS: clustering non supervisionato
- apprendimento di Modelli Markoviani Nascosti (Hidden Markov Models)

Cerchiamo di capire EM con un esempio...



Ogni istanza x generata

1. scegliendo una delle Gaussiane con probabilità uniforme
2. generando una istanza a caso secondo la Gaussiana scelta

EM per stimare k medie

Date:

- istanze da X generate da una mistura di k distribuzioni Gaussiane
- medie $\langle \mu_1, \dots, \mu_k \rangle$ sconosciute delle k Gaussiane (σ^2 conosciuto ed uguale per tutte le Gaussiane)
- non si sa quale istanza x_i è stata generata da quale Gaussiana

Determinare:

- stime maximum likelihood di $\langle \mu_1, \dots, \mu_k \rangle$

ogni istanza può essere pensata nella forma $y_i = \langle x_i, z_{i1}, z_{i2} \rangle$ (caso $k = 2$), dove

- z_{ij} è 1 se x_i è generata dalla j-esima Gaussiana
- x_i osservabile
- z_{ij} non osservabile

EM per stimare k medie

Algoritmo EM: scegliere a caso l'ipotesi iniziale $h = \langle \mu_1, \mu_2 \rangle$, poi ripetere

passo E: calcola il valore aspettato $E[z_{ij}]$ di ogni variabile non osservabile z_{ij}, assumendo che valga l'ipotesi corrente $h = \langle \mu_1, \mu_2 \rangle$

$$
\begin{aligned}
E[z_{ij}] &= \frac{p(x = x_i | \mu = \mu_j)}{\sum_{n=1}^{2} p(x = x_i | \mu = \mu_n)} \\
&= \frac{e^{-\frac{1}{2\sigma^2}(x_i - \mu_j)^2}}{\sum_{n=1}^{2} e^{-\frac{1}{2\sigma^2}(x_i - \mu_n)^2}}
\end{aligned}
$$

passo M: calcola la nuova ipotesi maximum likelihood $h' = \langle \mu'_1, \mu'_2 \rangle$, assumendo che il valore preso da ogni variabile non osservabile z_{ij} sia il suo valore aspettato $E[z_{ij}]$ (calcolato sopra). Rimpiazza $h = \langle \mu_1, \mu_2 \rangle$ con $h' = \langle \mu'_1, \mu'_2 \rangle$.

$$
\mu_j \leftarrow \frac{\sum_{i=1}^{m} E[z_{ij}]\ x_i}{\sum_{i=1}^{m} E[z_{ij}]}
$$

Algoritmo EM

Converge alla ipotesi h_{ML} locale (massimo locale) fornendo stime per le variabili non osservabili z_{ij}

Di fatto, trova un massimo locale di $E[\ln P(Y|h)]$, dove

- Y rappresenta tutti i dati (variabili osservabili e non)
- il valore aspettato è preso sui possibili valori di variabili non osservabili in Y

Problema EM in generale

Dati:

- dati osservati $X = \{x_1, \ldots, x_m\}$
- dati non osservabili $Z = \{z_1, \ldots, z_m\}$
- distribuzione di probabilità parametrizzata $P(Y|h)$, dove
  - $Y = \{y_1, \ldots, y_m\}$ è tutto l'insieme dei dati $y_i = x_i \cup z_i$
  - h sono i parametri

Determinare:

- h che massimizza (localmente) $E[\ln P(Y|h)]$

Metodo EM Generale

Definire la funzione di verosimiglianza (likelihood) $Q(h'|h)$ che calcola $Y = X \cup Z$ usando i dati osservati X ed i parametri correnti h per stimare Z

$$Q(h'|h) \leftarrow E[\ln P(Y|h')|h, X]$$

Algoritmo EM:

passo di stima (E): calcolare $Q(h'|h)$ usando l'ipotesi corrente h ed i dati osservati X per stimare la distribuzione di probabilità su Y

$$Q(h'|h) \leftarrow E[\ln P(Y|h')|h, X]$$

passo di massimizzazione (M): rimpiazza l'ipotesi h tramite l'ipotesi h' che massimizza la funzione Q

$$h \leftarrow \arg\max_{h'} Q(h'|h)$$