Cvičení z RPZ
Expectation-Maximization Algorithm

Jan Šochman

19. prosince 2005

1 Úvod

EM algoritmus je zástupcem algoritmů řešících úlohu učení bez učitele. Na vstup je mu dána množina pozorování a výstupem jsou parametry statistického modelu, z něhož byla pozorování vygenerována. Cílem tohoto cvičení je naprogramovat si EM algoritmus pro odhad parametrů směsi normálních rozdělení. Cvičení navazuje volně na cvičení o algoritmu k-means.

2 Formulace úlohy

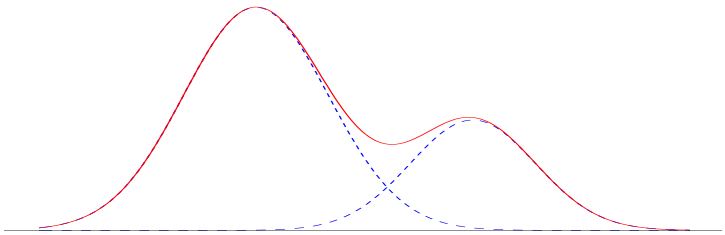
Mějme dánu množinu pozorování $X = \{x_1, \ldots, x_n\}$. Dále víme, že tato pozorování byla získána náhodným nezávislým výběrem z pravděpodobnosti $p(x, \theta)$. Známe tedy typ rozdělení, ale neznáme jeho parametry θ. Cílem algoritmu je na základě pozorování X nalézt nejvěrohodnější hodnoty parametrů θ.

2.1 Směs normálních rozdělení

V našem konkrétním případě bude mít pravděpodobnost $p(x, \theta)$ tvar směsi normálních rozdělení

$$p(x, \theta) = \sum_{c=1}^{k} P(c) p(x|\mu_c, \sigma_c), \tag{1}$$

kde $p(x|\mu_c, \sigma_c)$ je normální rozdělení se střední hodnotou μ_c a směrodatnou odchylkou σ_c. Směs je tedy lineární kombinací k normálních rozdělení s koeficienty $P(c)$. Příklad takové směsi pro $k = 2$ ukazuje následující obrázek.



Hledané parametry v případě směsi normálních rozdělení jsou tedy

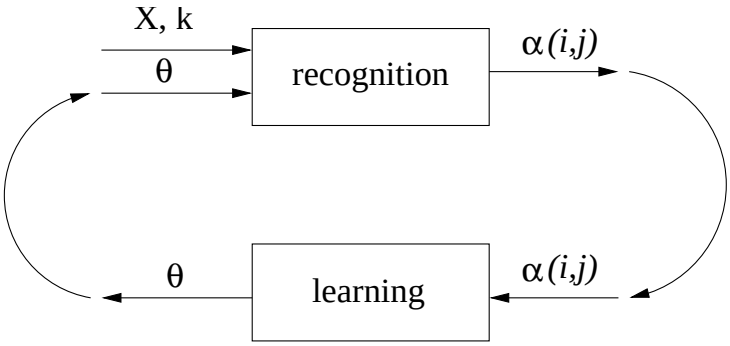
$$\theta = \{P(c), \mu_c, \sigma_c; c = 1, \ldots, k\}. \tag{2}$$

Parametry jedné složky směsi budeme dále značit θ_c.

3 Expectation-Maximization algoritmus

V dalším textu budeme používat označení *třída* jako identifikátor normálního rozdělení ve směsi, ze které bylo pozorování vygenerováno.

Stejně jako algoritmus k-means i EM algoritmus je iterativní algoritmus, který střídavě provádí dva kroky: učení a rozpoznávání (klasifikaci).



3.1 Rozpoznávání

V algoritmu k-means probíhalo rozpoznávání tak, že se určila přímo příslušnost daného pozorování x_i ke shluku (třídě), y_i. Namísto přímé klasifikace

pozorování do tříd počítá EM algoritmus aposteriorní pravděpodobnosti příslušnosti ke třídě

$$p(c|x_i) \sim \alpha(i,c) = \frac{P(c)p(x_i|c)}{\sum_{c=1}^{k} P(c)p(x_i|c)}. \tag{3}$$

Tedy pravděpodobnost klasifikace pozorování x_i do třídy c. Namísto pravděpodobnosti $p(c|x_i)$ budeme používat označení $\alpha(i,c)$, protože v jednotlivých iteracích nebudeme mít přímo aposteriorní pravděpodobnost, ale jen její přibližnou hodnotu spočtenou z odhadů apriorních pravděpodobností $P(c)$ a hustot pravděpodobnosti $p(x|c)$ ve kterých je skutečná hodnota parametrů θ_c nahrazena opět jen jejich odhadem.

3.2 Učení

Parametry θ hledá EM algoritmus metodou maximální věrohodnosti. Odvození provedeme jako bychom měli přímo k dispozici klasifikaci pozorování x_i do třídy y_i a teprve pak se podíváme, jak použít reálné hodnoty $\alpha(i,c)$, které vrací rozpoznávání. Věrohodnost můžeme díky nezávislosti pozorování a znalosti tvaru rozdělení rozepsat na

$$l(\theta) = P(X|\theta) = \prod_{i=1}^{n} p(x_i,\theta) = \prod_{i=1}^{n} P(y_i)p(x_i|\theta_{y_i}). \tag{4}$$

Označme si $L(\theta)$ logaritmus věrohodností funkce $l(\theta)$

$$L(\theta) = \sum_{i=1}^{n} \log P(y_i) + \sum_{i=1}^{n} \log p(x_i|\theta_{y_i}). \tag{5}$$

Definujeme-li si funkci α jako

$$\alpha(i,c) = \begin{cases} 0, & y_i \neq c \\ 1, & y_i = c \end{cases}, \tag{6}$$

můžeme logaritmus věrohodnosti přepsat do tvaru

$$L(\theta) = \sum_{i=1}^{n}\sum_{c=1}^{k} \alpha(i,c) \log P(c) + \sum_{i=1}^{n}\sum_{c=1}^{k} \alpha(i,c) \log p(x_i|\theta_c). \tag{7}$$

Nejlepší odhad apriorních pravděpodobností maximalizující $L(\theta)$ je (zderivujeme $L(\theta)$ podle $P(c)$ a položíme rovno nule)

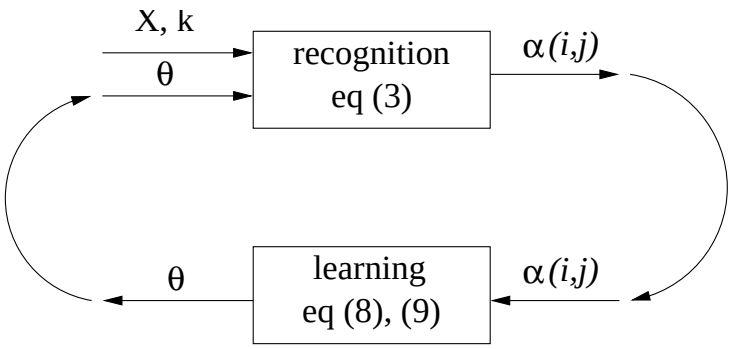
$$P(c) = \frac{\sum_{i=1}^{n} \alpha(i,c)}{\sum_{c=1}^{k} \sum_{i=1}^{n} \alpha(i,c)} = \frac{\sum_{i=1}^{n} \alpha(i,c)}{n}, \quad c = 1, \ldots, k \tag{8}$$

a hledání maximálně věrohodných parametrů se rozpadá na k nezávislých úloh

$$\theta_c^* = \operatorname*{argmax}_{\theta} \sum_{i=1}^{n} \alpha(i,c) \log p(x_i, \theta_c), \quad c = 1, \ldots, k. \tag{9}$$

Všimněte si, že optimalizační úloha (9) je definována nejenom pro 0/1 funkci $\alpha(i,c)$, ale i pro jakákoliv reálná čísla $\alpha(i,c) \in \langle 0,1 \rangle$ taková, že $\sum_{c=1}^{k} \alpha(i,c) = 1$ pro $i = 1, \ldots, n$. Tedy i pro $\alpha(i,c)$ získaná během učení.

Výsledný algoritmus lze vyjádřit následujícím diagramem



3.3 EM algoritmus pro směs normálních rozdělení

Optimalizační úloha (9) nemá jedno obecně platné řešení. Její vyřešení závisí na konkrétním zadání úlohy, na konkrétních hustotách pravděpodobnosti $p(x|c)$, ze kterých se směs skládá. V případě směsi normálních rozdělení lze vyjádřit řešení úlohy (9) analyticky

$$\mu_c = \frac{\sum_{i=1}^{n} \alpha(i,c) x_i}{\sum_{i=1}^{n} \alpha(i,c)} \tag{10}$$

$$\sigma_c^2 = \frac{\sum_{i=1}^{n} \alpha(i,c)(x_i - \mu_c)^2}{\sum_{i=1}^{n} \alpha(i,c)}. \tag{11}$$

3.4 Inicializace a ukončení

Podobně jako v algoritmu k-means provádíme inicializaci parametrů náhodně. Vhodná inicializace se často získává díky částečné znalosti úlohy, či lze použít nejakou vhodnou heuristiku.

EM algoritmus **nezaručuje nalezení globálního maxima** věrohodnostní funkce $L(\theta)$! Nalezené parametry závisí na startovacím bodu algoritmu. Prakticky se tento problém řeší spuštěním EM algoritmu z více náhodně generovaných bodů a výběrem řešení s konečnou největší věrohodností.

EM algorimus naopak **zaručuje monotónnost konvergence v** $L(\theta)$ (k lokálnímu optimu). V každé iteraci algoritmu se tedy najde model s větší věrohodností, než v iteraci předchozí. Tedy

$$L(\theta^{(1)}) < L(\theta^{(2)}) < \ldots < L(\theta^{(t)}) \tag{12}$$

Jako ukončovací podmínka se často volí

$$L(\theta^{(t)}) - L(\theta^{(t-1)}) < \varepsilon\,, \tag{13}$$

kde ε je dostatečně malé nezáporné číslo.

4 Zadání úlohy

1. Seznamte se z demonstračním programem `demo_emgmm` ze Statistical Pattern Recognition Toolboxu. Tento program demonstruje použití EM algoritmu pro odhad parametrů směsi dvourozměrných normálních rozdělení.

2. Implementujte EM algoritmus pro směs jednorozměrných normálních rozdělení a ověřte jeho činnost na synteticky generovaných datech. Pro generování dat použijte funkci `gmmsamp`. Pro vizualizaci výsledku použijte funkci `pgmm`. Funkce jsou opatřeny jednoduchými příklady použití.

3. Vykreslete věrohodnostní funkci $L(\theta)$.

Literatura

[1] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society*, 39:185–197, 1977. Základní článek (zájemcům okopíruji).

[2] G. McLachlan and T. Krishnan. *The EM Algorithm and Extensions*. John Wiley & Sons, New York, 1997. Knihovna CMP.

[3] S. Russell. The EM Algorithm, 1998. Naleznete Googlem.

[4] M. I. Schlesinger and V. Hlaváč. *Deset přednášek z teorie statistického a strukturního rozpoznávání, in Czech (Ten Lectures on Statistical and Structural Pattern Recognition)*. Czech Technical University Publishing House, Praha, Czech Republic, 1999. Knihovna CMP.