20. Grupiranje II

Strojno učenje 1, UNIZG FER, ak. god. 2023./2024.

Jan Šnajder, vježbe, v2.3

1 Zadatci za učenje

1. [***Svrha: Razumjeti model miješane gustoće i razlog zašto maksimizacija log-izglednost nije analitički rješiva. Razumjeti kako uvođenje latentnih varijabli rješava taj problem. Razumjeti, na općenitoj razini, E-korak i M-korak. Razumjeti rad algoritma kao maksimizacije log-izglednosti i razumjeti kako ishod ovisi o broju grupa i početnoj inicijalizaciji.***] Algoritam maksimizacije očekivanja (EM-algoritam), kada se koristi za grupiranje, zapravo je poopćenje algoritma K-sredina.

   (a) Što je prednost, a što nedostatak, algoritma maksimizacije očekivanja primijenjenog na GMM u odnosu na algoritam K-sredina?

   (b) Napišite izraz za gustoću $p(\mathbf{x})$ za model miješane gustoće (bez latentnih varijabli) i izraz za pripadnu (nepotpunu) log-izglednost.

   (c) Napišite izraz za mješavinu s latentnim varijablama i izvedite izraz za (potpunu) log-izglednost tog modela. Možemo li dalje raditi izravno s tom log-izglednošću? Zašto?

   (d) Definirajte E-korak i M-korak algoritma maksimizacije očekivanja primijenjenog na Gaussovu mješavinu.

   (e) Skicirajte vrijednost log-izglednosti $\ln \mathcal{L}(\boldsymbol{\theta}|\mathcal{D})$ modela Gaussove mješavine kao funkcije broja iteracija, i to za tri različite vrijednosti parametra K (broj grupa): $K = 1$, $K = 10$ i $K = 100$. Na istom grafikonu skicirajte krivulju za $K = 10$ kada se za inicijalizaciju središta koristi algoritam K-sredina.

2. [***Svrha: Isprobati rad algoritma hijerarhijskog aglomerativnog grupiranja (HAC) na konkretnom primjeru, za slučaj kada primjeri nisu vektori. Uočiti razliku između udaljenosti i sličnosti te razliku između jednostruke i potpune povezanosti.***] Jednako kao i algoritam K-medoida, algoritam hijerarhijskog aglomerativnog grupiranja može se primijeniti u slučajevima kada primjeri nisu prikazani kao vektori značajki te kada umjesto mjere udaljenosti između vektora raspolažemo općenitjom mjerom sličnosti (ili različitosti). Neka je *sličnost* primjera iz $\mathcal{D}$ definirana sljedećom matricom sličnosti:

$$S = \begin{array}{c} \\ a \\ b \\ c \\ d \\ e \end{array} \begin{array}{c} \begin{array}{ccccc} a & b & c & d & e \end{array} \\ \begin{pmatrix} 1.00 & 0.26 & 0.15 & 0.20 & 0.17 \\ 0.26 & 1.00 & 0.24 & 0.31 & 0.31 \\ 0.15 & 0.24 & 1.00 & 0.20 & 0.50 \\ 0.20 & 0.31 & 0.20 & 1.00 & 0.24 \\ 0.17 & 0.31 & 0.50 & 0.24 & 1.00 \end{pmatrix} \end{array}$$

   (a) Izgradite dendrogram uporabom jednostrukog povezivanja. Kada bi bilo potrebno napraviti particiju grupa, na kojoj biste razini presjekli taj dendrogram?

   (b) Izgradite dendrogram uporabom potpunog povezivanja. Kada bi bilo potrebno napraviti particiju grupa, na kojoj biste razini presljekli taj dendrogram?

3. [***Svrha: Razumjeti kako se unutarnji kriterij algoritma grupiranja može (pokušati) upotrijebiti za provjeru grupiranja (odabir optimalnog broja grupa). Razumjeti da Akaikeov kriterij u stvari oponaša regulariziranu funkciju pogreške, koja pak aproksimira pogrešku generalizacije.***]

   (a) Skicirajte krivulju log-izglednosti kod EM-algoritma kao funkciju broja grupa K. Obrazložite izgled krivulje. Možete li temeljem ove krivulje odrediti optimalan broj grupa? Kako?

(b) Optimizacija broja grupa K može se provesti nekim kriterijem koji kombinira funkciju pogreške (odnosno log-izglednost) i složenost modela. Takav kriterij odgovara strukturnome riziku modela, koji je minimalan za optimalan broj grupa. Jedan takav kriterij jest Akaikeov informacijski kriterij (AIC):

$$K^* = \operatorname*{argmin}_K \big(-2 \ln \mathcal{L}(K) + 2q(K) \big)$$

gdje je $-\ln \mathcal{L}(K)$ negativna log-izglednost podataka za K grupa, a $q(K)$ je broj parametara modela s K grupa.

Pretpostavite da podatci $\mathcal{D}$ u stvarnosti dolaze iz $K = 5$ grupa. Podatke grupiramo dvjema varijantama EM-algoritma: standardni algoritam i preinačeni algoritam s dijeljenom kovarijacijskom matricom (zajednička kovarijacijska matrica procijenjena nad čitavim skupom primjera $\mathcal{D}$ na početku izvođenja algoritma). Skicirajte za ta dva algoritma funkciju koju minimizira Akaikeov minimizacijski kriterij.

2 Zadatci s ispita

1. (P) Algoritam GMM koristimo za grupiranje $N = 10$ primjera u dvodimenzijskome ulaznom prostoru. Skup primjera koje grupiramo je sljedeći:

$$\mathcal{D} = \{(0,0), (1,1), (1,2), (2,2), (2,3), (5,0), (5,1), (6,0), (6,6), (7,7)\}$$

Razmatramo tri modela GMM:

$\mathcal{H}_1$: $K = 2$ grupa, puna kovarijacijska matrica
$\mathcal{H}_2$: $K = 2$ grupa, izotropna kovarijacijska matrica
$\mathcal{H}_3$: $K = 3$ grupe, izotropna kovarijacijska matrica

Za sva tri modela kovarijacijska matrica je nedijeljena, dakle svaka komponenta ima svoju kovarijacijsku matricu. Za početne centroide odabiremo nasumično dva odnosno tri primjera iz $\mathcal{D}$, ovisno o broju grupa K. Za svaki model grupiranje ponavljamo 100 puta te kao konačno grupiranje uzimamo ono s najvećom log-izglednošću na skupu $\mathcal{D}$. Zanima nas kojoj grupi najvjerojatnije pripada primjer $\mathbf{x}^{(5)} = (2,3)$, to jest zanima nas k koji maksimizira odgovornost $h_k^{(5)} = P(y = k|\mathbf{x}^{(5)})$. Ta vrijednost će biti različita za ova tri modela. Označimo sa h_α maksimalnu odgovornost za primjer $\mathbf{x}^{(5)}$ u modelu $\mathcal{H}_\alpha$, to jest vjerojatnost pripadanja tog primjera najvjerojatnijoj grupi dobivenoj grupiranjem pomoću modela $\mathcal{H}_\alpha$. **Što možemo zaključiti o odgovornostima h_α za ova tri modela?**

A $h_{\alpha_1} > h_{\alpha_2} > h_{\alpha_3}$ B $h_{\alpha_1} < h_{\alpha_2} < h_{\alpha_3}$ C $h_{\alpha_2} > h_{\alpha_1} > h_{\alpha_3}$ D $h_{\alpha_2} < h_{\alpha_1} < h_{\alpha_3}$

2. (P) Za grupiranje skupa primjera $\mathcal{D}$ koristimo algoritam GMM. Koristimo nekoliko varijanti tog modela:

$\mathcal{H}_1$: Model sa $K = 50$ središta inicijaliziranima algoritmom K-sredina
$\mathcal{H}_2$: Model sa $K = 50$ središta inicijaliziranima algoritmom K-sredina i dijeljenom kov. matricom
$\mathcal{H}_3$: Model sa $K = 50$ slučajno inicijaliziranim središtima i dijeljenom kov. matricom
$\mathcal{H}_4$: Model sa $K = 10$ središta inicijaliziranima algoritmom K-sredina i dijeljenom kov. matricom

Sa svakim modelom grupiranje ponavljamo 1000 puta i zatim za svaki model crtamo graf funkcije log-izglednosti kroz iteracije EM-algoritma, uprosječen kroz svih 1000 ponavljanja. Neka je LL_α^0 prosječna log-izglednost za model $\mathcal{H}_\alpha$ na početku izvođenja EM-algoritma, a neka je LL_α^* prosječna log-izglednost za taj model na kraju izvođenja EM-algoritma. **Što možemo unaprijed zaključiti o ovim log-izglednostima?**

A $LL_2^0 \geqslant LL_4^0$, $LL_1^* \geqslant LL_2^* \geqslant LL_3^*$
B $LL_3^0 \geqslant LL_4^0$, $LL_1^* \geqslant LL_3^* \geqslant LL_4^*$
C $LL_2^0 \geqslant LL_4^0 \geqslant LL_3^0$, $LL_1^* \geqslant LL_2^*$
D $LL_2^0 \leqslant LL_4^0$, $LL_2^* \leqslant LL_1^* \geqslant LL_3^*$

3. (P) Skup neoznačenih primjera $\mathcal{D}$ grupiramo modelom GMM treniranim EM-algoritmom. Koristimo nekoliko varijanti tog modela:

   $\mathcal{H}_1$: Model sa $K = 25$ središta inicijaliziranima algoritmom K-means++

   $\mathcal{H}_2$: Model sa $K = 50$ slučajno inicijaliziranim središtima i dijeljenom kovarijacijskom matricom

   $\mathcal{H}_3$: Model sa $K = 50$ središta inicijaliziranima algoritmom K-sredina i dijeljenom kovarijacijskom matricom

   Sa svakim modelom grupiranje ponavljamo 1000 puta i zatim za svaki model crtamo graf funkcije log-izglednosti kroz iteracije EM-algoritma, uprosječen kroz svih 1000 ponavljanja. Neka je LL^0_α prosječna log-izglednost za model $\mathcal{H}_\alpha$ na početku izvođenja EM-algoritma, LL^*_α prosječna log-izglednost za taj model na kraju izvođenja EM-algoritma te neka je k_α broj iteracija EM-algoritma za taj model. **Što možemo zaključiti o očekivanim odnosima između ovih vrijednosti?**

   A $LL^0_1 \geqslant LL^0_2$, $LL^*_2 \geqslant LL^*_3$, $k_1 \geqslant k_2$

   B $LL^0_1 \geqslant LL^0_3$, $LL^*_1 \geqslant LL^*_3$, $k_2 \geqslant k_1$

   C $LL^0_2 \geqslant LL^0_3$, $LL^*_3 \geqslant LL^*_2$, $k_3 \geqslant k_2$

   D $LL^0_3 \geqslant LL^0_2$, $LL^*_3 \geqslant LL^*_2$, $k_2 \geqslant k_3$

4. (P) Algoritmom GMM grupiramo primjere u dvodimenzijskome ulaznom prostoru. Skup podataka u stvarnosti je uzorkovan iz zajedničke distribucije koja se može opisati sljedećim mješavinskim modelom:

$$p(\mathbf{x}) = \sum_{j=1}^{3} \frac{1}{3} \mathcal{N}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) \quad \text{gdje} \quad \mu_1 = (5,5), \mu_2 = (5,10), \mu_3 = (-10,-10), \boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \boldsymbol{\Sigma}_3 = 2\mathbf{I}$$

   Skup $\mathcal{D}$ grupiramo u $K = 2$ grupe. Pritom isprobavamo tri modela, koji se međusobno razlikuju po pretpostavkama na kovarijacijsku matricu. Konkretno: dijeljena i puna kovarijacijska matrica ($\mathcal{H}_1$), nedijeljena i dijagonalna kovarijacijska matrica ($\mathcal{H}_2$) i nedijeljena i izotropna kovarijacijska matrica ($\mathcal{H}_3$). Neka je $\mathcal{L}_i$ izglednost parametara dobivena modelom $\mathcal{H}_i$ nakon konvergencije algoritma. Za inicijalizaciju središta koristi se algoritam K-means++. **Što su očekivani odnosi između izglednosti za ova tri modela?**

   A $\mathcal{L}_1 = \mathcal{L}_2 > \mathcal{L}_3$ B $\mathcal{L}_1 > \mathcal{L}_2 > \mathcal{L}_3$ C $\mathcal{L}_1 > \mathcal{L}_3$, $\mathcal{L}_2 > \mathcal{L}_3$ D $\mathcal{L}_2 > \mathcal{L}_1$, $\mathcal{L}_2 > \mathcal{L}_3$

5. (N) Algoritmom hijerarhijskog aglomerativnog grupiranja (HAC) grupiramo $N = 5$ primjera. Za grupiranje koristimo mjeru sličnosti, koja je za naših pet primjera definirana sljedećom matricom (matrica je simetrična, pa je donji trokut izostavljen):

$$\begin{array}{c} \\ \mathbf{x}^{(1)} \\ \mathbf{x}^{(2)} \\ \mathbf{x}^{(3)} \\ \mathbf{x}^{(4)} \\ \mathbf{x}^{(5)} \end{array} \begin{array}{c} \begin{array}{ccccc} \mathbf{x}^{(1)} & \mathbf{x}^{(2)} & \mathbf{x}^{(3)} & \mathbf{x}^{(4)} & \mathbf{x}^{(5)} \end{array} \\ \begin{pmatrix} 1 & 0.4 & 0.5 & 0.7 & 0.5 \\ & 1 & 0.9 & 0.3 & 0.6 \\ & & 1 & 0.7 & 0.1 \\ & & & 1 & 0.8 \\ & & & & 1 \end{pmatrix} \end{array}$$

   Provedite grupiranje algoritmom HAC s potpunim povezivanjem te nacrtajte pripadni dendrogram. Primijetite da dendrogram odgovara binarnom stablu, s pojedinim primjerima u listovima. **Kojem binarnom stablu odgovara dobiveni dendrogram?**

   A $((\mathbf{x}^{(2)}, \mathbf{x}^{(3)}), \mathbf{x}^{(4)}), (\mathbf{x}^{(5)}, \mathbf{x}^{(1)}))$

   B $((\mathbf{x}^{(2)}, \mathbf{x}^{(3)}), \mathbf{x}^{(1)}), (\mathbf{x}^{(4)}, \mathbf{x}^{(5)}))$

   C $((\mathbf{x}^{(2)}, \mathbf{x}^{(3)}), ((\mathbf{x}^{(4)}, \mathbf{x}^{(5)}), \mathbf{x}^{(1)})))$

   D $((\mathbf{x}^{(2)}, \mathbf{x}^{(3)}), ((\mathbf{x}^{(4)}, \mathbf{x}^{(1)}), \mathbf{x}^{(5)})))$

6. (N) Algoritmom hijerarhijskog aglomerativnog grupiranja (HAC) grupiramo $N = 5$ primjera. Za grupiranje koristimo mjeru sličnosti, definiranu sljedećom matricom:

$$\begin{pmatrix} 1.0 & 0.2 & 0.8 & 0.1 & 0.4 \\ 0.2 & 1.0 & 0.9 & 0.3 & 0.7 \\ 0.8 & 0.9 & 1.0 & 0.6 & 0.5 \\ 0.1 & 0.3 & 0.6 & 1.0 & 0.4 \\ 0.4 & 0.7 & 0.5 & 0.4 & 1.0 \end{pmatrix}$$

Provedite grupiranje algoritmom HAC s potpunim povezivanjem. Pritom u svakoj iteraciji bilježite na kojoj razini sličnosti se odvija stapanje dviju grupa. **Koliko iznosi zbroj po svim razinama sličnosti na kojima se odvija stapanje grupa?**

A 1.8 B 1.9 C 2.0 D 2.4

7. (N) Algoritmom HAC grupiramo riječi engleskog jezika. Neoznačeni skup podataka sastoji se od sljedećih riječi:

$$\mathcal{D} = \{\text{“water”, “watering”, “earth”, “air”}\}$$

Kao mjeru sličnosti između primjera koristimo jezgrenu funkciju nad znakovnim nizovima, definiranu kao $\kappa(\mathbf{x}_1, \mathbf{x}_2) = |\mathbf{x}_1 \cap \mathbf{x}_2|/|\mathbf{x}_1 \cup \mathbf{x}_2|$, gdje su operacije unije i presjeka definirane nad skupovima slova od kojih se riječi sastoje. Npr., $\kappa(\text{“water”, “watering”}) = 5/8 = 0.625$. Provedite prve dvije iteracije grupiranja algoritmom HAC uz prosječno povezivanje. **Na kojoj se razini sličnosti spajaju grupe u drugoj iteraciji algoritma HAC?**

A 0.292 B 0.478 C 0.535 D 0.583