<u>Shluková analýza (clustering)</u>

Petr Jurčík, Vít Vrba

Obsah

1 Úvod

Metody zabívající se nastavením klasifikátoru bývají souhrně označovány jako *učení bez učitele* . Jejich příkladem jsou metody *shlukové analýzy,* které umožňují nastavení klasifikátoru nejen bez údajů o správné klasifikaci, ale v krajním případě i bez apriorní znalosti počtu tříd.

Výchozí data pro metody shlukové analýzy tvoří množina, která obsahuje *neklasifikované vektory* přízanků. Úkolem *shlukové analýzy* je nalézt shluky těchto vektorů, tj. skupiny , jejichž prvky jsou si vzájemně *blízké, podobné.* Rozdělění vektorů do shluků není jednoznačné. Některá rozdělení pokládáme za lepší než jiná. Původní množinu rozkládáme na co možná nejkompaktnější podmnožiny. Úspěšného výsledku lze dosáhnout pouze tehdy, tvoří-li prvky trénovací množiny v příznakovém prostoru shluky.

Metody shlukové analýzy vycházejí z předpokladu, že dva vektory příznaků patří do stejného shluku, jsou-li si geometricky blízké. Shluky poté ztotožníme se třídami a dostáváme výslednou klasifikaci.

2 Rozbor

2.1 Formulace úlohy

Je dán soubor dat X={x_i}i=1....n. Cílem je rozdělit data do m shluků C$_j$ tak, aby:

1. C$_j$≠0 ; j=1...m
2. $\bigcup_{j=1}^{m} C_j = X$
3. C$_j$ $\bigcap$ C$_k$ = 0 ; j≠k (Fuzzy Clustering nemá bod 3)

a přitom minimalizovat kritérium:

$$R(d) = \int_x p(x).W(Q(x),x)dx$$

$$R'(d) = \sum_{i=1} W(Q(x_i),x_i)$$

kde d je funkce příslušnosti ke shluku C_j :Q(x)=j $\Leftrightarrow$ Xϵ C_j ; W(Q(x),x) je pokutová funkce charakterizující konkrétní úlohu.

(Baysovské rozhodování minimalizuje : $R = \int_x \sum_k p(x,k).W(Q(x),x)dx$).

2.2 Míry

Před prováděním shlukové analýzy je nezbytné zvolit některou z uvedených měr:

2.2.1 Míra podobnosti (popřípadě nepodobnosti)

Pro ohodnocení vzájemné „blízkosti, podobnosti" prvků se využívá ***měr podobnosti*** mezi dvěma vektory příznaků X_i a X_j. Nejčastěji se používá *Euklidovská vzdálenost* vekorů X_i a X_j, která oceňuje nepodobnost :

$$\sigma(X_i, X_j) = \| X_i - X_j \| = \sqrt{\sum_{k=1}^{N} (X_{ik} - X_{jk})^2} = \sqrt{(X_i - X_j)^T (X_i - X_j)} \tag{1}$$

Dva vektory jsou si tím podobnější, čím je jejich vzdálenost menší. Dále popisované metody shlukové analýzy využívají vesměs tuto míru.

Další míry: Mahalanobisova $((x_i - x_j)^T C(x_i - x_j))$, l-metrika a další

2.2.2 *Vzdálenost mezi dvěma shluky*

- je důležitá při postupném slučování shluků.

Označme C_i resp. C_j jako i-tý resp. j-tý shluk, tovřící podmnožinu všech shluků.Obvykle se pracuje s těmito mírami (vzdálenostni mezi obrazy shluků):

- minimální míra : $d_{min}(C_i, C_j) = min\{\| X - X' \|\}$, kde $X\epsilon C_i$, $X'\epsilon C_j$
- maximální míra : $d_{max}(C_i, C_j) = max\{\| X - X' \|\}$, kde $X\epsilon C_i$, $X'\epsilon C_j$
- průměrná míra : $d_{arg}(C_i, C_j) = \frac{1}{K_i K_j} \sum\sum\{\| X - X' \|\}$, kde $X\epsilon C_i$, $X'\epsilon C_j$, K_i resp. K_j je počet vektorů ve shluku C_i resp. C_j.
- střední míra (vzdálenost ideálních středů shluků - etalonů): $d_{mean}(C_i, C_j) = \| \mu_i - \mu_j \|$, kde μ_i resp. μ_j je střední hodnota vektorů příznaků shluku C_i resp. C_j.

$$\mu_i = \frac{1}{K_i} \sum_{k=1}^{K_i} X_k \tag{2}$$

Tyto míry poskytují obvykle stejné výsledk, pokud jsou shluky dostatečně kompaktní a dobře separovatelné.

2.2.3 Kritérium kvality (kriteriální funkce J)

Míra pro ohodnocování kvality rozdělení obrazů do shluků. Hledáme takové rozdělení, pro které nabývá toto krytérium extrém. Nejčastěji se používá kritérium minima kvadrátu odchylky:

$$J_e = \sum_{r=1}^{R} J_r = \sum_{r=1}^{R} \sum_{x\epsilon C_r} \| X - \mu_r \|^2 \tag{3}$$

kde

$$J_r = \sum_{x\epsilon C_r} \| X - \mu_r \|^2 \tag{4}$$

je rozptyl uvnitř shluku, R je počet shluků. Vybráním kritéria (například 3) se stává úloha shlukové analýzy **úlohou optimalizační**. Může být zadána např. takto:

Je dána množina vektorů C ={X_1, X_2,..........,X_k } a počet shluků R Úkolem je nalézt takový rozklad množiny C na C_1, C_2,.....C_R, aby hodnota kritéria (3) nabývala minima. Množiny C_1, C_2,.....C_R jsou potom hledané shluky. Řešení spočívá v postupných změnách přiřazení vktorů X do shluků tak, aby kritérium J stále zmenšovalo svoji hodnotu.

3 Algoritmy shlukové analýzy

Volba počtu tříd:
Při optimální volbě počtu tříd jde v podstatě o stanovení počtu „přirozených" shluků. U hierarchického slučování je možné například zvolit úroveň shlukování, na níž se proces shlukové analýzy zdola nahoru ukončí. Jinou možností je opakovat proceduru shlukování pro R=1, R=2, R=3,....a sledovat, jak se mění hodnota kriteriální funkce J se změnou R. Pokud jsou obrazy skutečně rozloženy do R=R' dobře separovatelných shluků, lze očekávat, že funkce J klesá velcie rychle až k R' a klesá podstatně pomaleji pro R>R'. Ze zlomu funkce J(R) lze pak jednoznačně určit optimální počet tříd R'.

3.1 Hierarchické shlukování

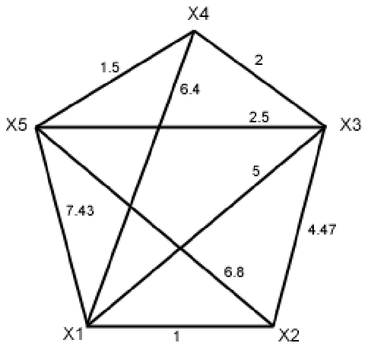
Na počátku máme n samostatných vektorů příznaků X. Předpokládejme, že každý vektor X tvoří samostatný shluk (máme n shluků) . V každém dalším kroku dva nejbližší shluky sjednotíme do jediného a tak snížíme jejich počet o jedna. Po *n-1* krocích dostaneme jediný shluk obsahující všechny vektory. Pro *hierarchické metody* je charakteristické, že pokud se vektory dostanou do společného shluku již v tomto shluku zůstanou po všechny následující kroky. Nejbližší shluky zjišťujeme podle některé z měr pro určování vzdáleností mezi shluky.

3.1.1 Popis algoritmu

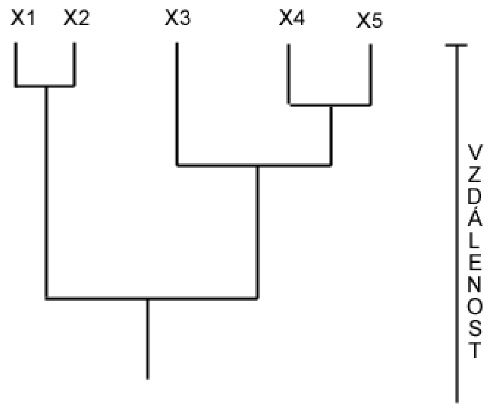
1. Inicializace - každý bod je jeden samostatný shluk
2. Nalezení dvou nejbližších shluků C_i a C_j.
3. Spojení C_i a C_j shluků do jednoho shluku
4. Zbyl-li více než jeden shluk skoč na bod 2., jinak konec

3.1.2 Dendogram

Při **hierarchickém shlukování** lze z výhodou použít popisu úlohy i řešení pomocí **grafů** : Nakreslíme-li úplný graf s pěti uzly, které označíme X_1, X_2, X_3, X_4, X_5, můžeme k hranám přiřadit vzdálenost mezi odpovídajícími uzly.



Nepodobnost mezi těmito body budeme měřit podle (1) jako jejich vzdálenost. Na počátku uvažujme pět shluků odpovídající jednotlivým bodům. Shluky budeme spojovat podle nějaké míry vzdálenosti (vybereme například minimální míru d_{min}). V prvním kroku spojíme X_1 a X_2 , protože jsou si nejméně podobné, jejich vzdálenost je rovna 1. Ve druhém kroku spojíme shluk X_4 a X_5 , které jsou od sebe vzdálené 1.5. Ve třetím kroku spojíme shluk X_3 se shlukem X_4 , X_5 vzhledem ke vzdálensoti 2 mezi těmito shluky. V posledním kroku vytvoříme jediný shluk. Spojování shluků můžeme znázornit stromem, který se nazývá **dendogram**:



Dendogram můžeme „rozříznout“podle počtu požadovaných shluků, vzdálenosti shluků, … a tím získáme výsledné shluky. V případě, že jako vzdálenost shluků je uvažována vzdálenost nejbližších bodů shluků, lze úlohu formulovat jako hledání minimální kostry v úplném grafu, jehož uzly jsou body a ohodnocení hran je jejich vzdálenost. Tento popsaný postup shlukování je hierarchickým *shlukováním zdola nahoru*, postup od jednoho shluku ke K shlukům je *shlukování shora dolů*.

3.2 EM algoritmus a ML(Maximum likelihood - maximálně věrohodný) odhad

Trénovací množina pro ML odhad:

$(X,k) = \left\{ \begin{array}{ccccc} X_1 & X_2 & X_3 & .... & X_n \\ k_1 & k_2 & k_3 & .... & k_n \end{array} \right\}$

Zavedeme : $\alpha_1(k)$,$\alpha_2(k)$,……$\alpha_n(k)$

kde $\alpha_i(k)$ říká, jestli í-tý vzorek patří do k-té třídy: $\alpha_i(k) = \left\{ \begin{array}{l} 1\ ;\ k = k_i \\ 0\ ;\ k \neq k_i \end{array} \right.$ kde k_i=1…M je počet tříd

ML-odhad hledá p(x_i,k)=p(k).p(x|k), tak aby maximálizoval pravděpodobnost trénovací množiny: p(X,K)=$\prod_{i=1}^{n} p(x_i, k_i)$

$$\text{ML odhad} \Longrightarrow \text{EM algoritmus}$$

-neznámá čísla $\alpha_i(k)$ nahradíme pravděpodobností p(k|x_i), kterou zpočteme z výchozího modelu.

3.2.1 EM algoritmus

- odhaduje směs K rozdělení s parametry a_k z množiny neklasifikovaných dat
- vychází z ML odhadu

$$p(x_i) = \sum_k p(k).p(x_i \mid k) = \sum_k p(k).f(x_i, a_k)$$

$$\text{Hledám parametry modelu : } \Theta = \{p(k), a_k\}$$

- skládá se ze dvou opakujících se kroků:

1. pravděpodobnosti $\alpha^t(k,i)$, že pozorování x_i bylo generováno rozdělením s parametry a_k

$$\alpha^t(k,i) = \frac{p^t(k).p^t(x_i \mid k)}{\sum_k p^t(k).p^t(x_i \mid k)} = \frac{p^t(x_i,k)}{p^t(x_i)} = p^t(k \mid x_i)$$

2. odhad parametrů pro čas t+1:

$$p(k)^{t+1} = \frac{\sum_{i=1}^{N} \alpha^t(k,i)}{\sum_{k=1}^{K} \sum_{i=1}^{N} \alpha^t(k,i)}$$

3.3 Algoritmus ISODATA = K - Means

- minimalizuje vzdálenost bodu od etalonu shluku
- rozděluje do předem daného počtu shluků (tříd) K : C_1,C_2, ...,C_k

(zjednodušený ML odhad, kde se nepracuje s pravděpodobnosmi, ale se vzdáleností)

3.3.1 Popis algoritmu

1. inicializace - nahodně zvolit μ_j^0 ,tak aby $C_j^0 \neq 0$, pro j=1...k
2. rozdělit data na shluky: $C^t(x_i)=\arg_j \min \| x_i - \mu_j^{t-1} \|$
3. odhadnout pro každý shluk ${C_j}^t$ novou hodnotu $\mu_j^t = \sum_{x \epsilon C_i} \frac{x}{|C_i|}$ (těžiště)
4. pokud $\mu_j^t = \mu_j^{t-1}$ pak konec, jinak krok 2.

Základní algoritmus používá Euklidovskou vzdálenost a μ_j je aritmetický průměr bodu ve shluku(střední hodnota ve třídě) - etalon

3.3.2 Vlastnosti

+ jednoduchý

+ konverguje v konečném počtu kroků k nějakému řešení C^*

- může existovat více řešení C^*v závislosti na počátečních podmínkách

Reference

[1] Vladimír Mařík, Zdeněk Kotek a kol. : Metody rozpoznávání a jejich aplikace, Academia 1993

[2] Zdeněk Kotek, Petr Vysoký, Zdeněk Zdráhal : Kybernetika, SNTL 1990

[3] Vladimír Mařík, Olga Štěpánková, Jiří Lažanský : Umělá inteligence I, Academia 1993