SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 6130

Usporedba algoritama grupiranja primjenom programske knjižnice Scikit-Learn

Dunja Aćimović

Zagreb, svibanj 2021.

Umjesto ove stranice umetnite izvornik Vašeg rada.
Da bi ste uklonili ovu stranicu obrišite naredbu `\izvornik`.

SADRŽAJ

1. Uvod

Strojno učenje grana je umjetne inteligencije koja se bavi oblikovanjem algoritama koji na osnovu empirijskih podataka poboljšavaju svoju učinkovitost. Dva osnovna pristupa strojnog učenja su nadzirano i nenadzirano učenje.

Nadzirano učenje se bavi problemima klasifikacije i regresije, a nenadzirano problemima grupiranja i smanjenja dimenzionalnosti. Nadzirano učenje zahtjeva ulazne podatke u obliku (x, y). Uz ulazne vrijednosti x, predaju se i ciljne vrijednosti y koje predstavljaju razrede u koje želimo svrstati ulazne vrijednosti. Suprotno tome nenadziranom učenju se predaju samo ulazne vrijednosti.

Grupiranje je oblik nenadziranog učenja za analizu statičkih podataka. Skup podataka dijeli u podskupove na temelju nekog zajedničkog obilježja. Grupiranja dijelimo na particijska i hijerarhijska.

Particijsko grupiranje dijeli skup primjera u grupe sličnih obilježja. Grupe se stvaraju istovremeno. Hijerarhijsko grupiranje skup primjera razdijeljuje u ugniježđene grupe koje čine hijerarhiju grupa. Za razliku od particijskog grupiranja, grupe se ne stvaraju istovremeno, već postepeno na osnovu prethodno stvorenih grupa.

2. Algoritmi grupiranja

2.1. Algoritam k-srednjih vrijednosti

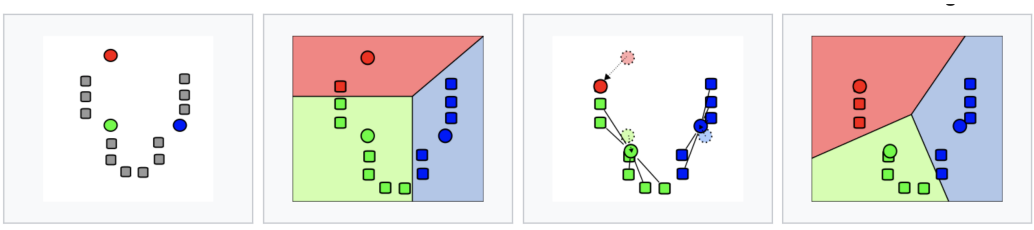
Algoritam k-srednjih vrijednosti je najjednostavniji algoritam grupiranja. Skup od n zadanih pojava dijeli u k grupa. Svaka grupa je predstavljena centroidom, koji je srednja vrijednost svih ulaznih vrijednosti unutar grupe. Svaka ulazna vrijednost se pridruži grupi s najsličnijim centroidom.

1. Slučajnim odabirom ili pomoću neke heuristike, odabire se k početnih srednjih vrijednosti.

2. Svaka se pojava pridjeljuje grupi s onom srednjom vrijednosti koja je u Euklidskom prostoru najbliža vrijednosti atributa te pojave.

3. Nakon što se sve pojave rasporede u grupe računaju se nove srednje vrijednosti svake grupe na temelju vrijednosti atributa pojava pridijeljenih svakoj od tih grupa.

Koraci 2 i 3 se ponavljaju dok se ne postigne stacionarno stanje.



Slika 2.1: Algoritam k-srednjih vriijednosti - tri koraka i rezultat iteracije

Kad za svaku pojavu t_1 vrijedi da je u iteraciji t i $(t+1)$ pridijeljena istoj grupi algoritam izlazi iz petlje.

Po složenosti ovaj algoritam je NP-težak problem. Složenost možemo definirati kao $O(n^{dk+1}log(n))$ samo uz uvjet da su dimenzionalnost Euklidskog prostora atributa pojava d i broj srednjih vrijednosti k unaprijed poznati. No bez obzira na veliku

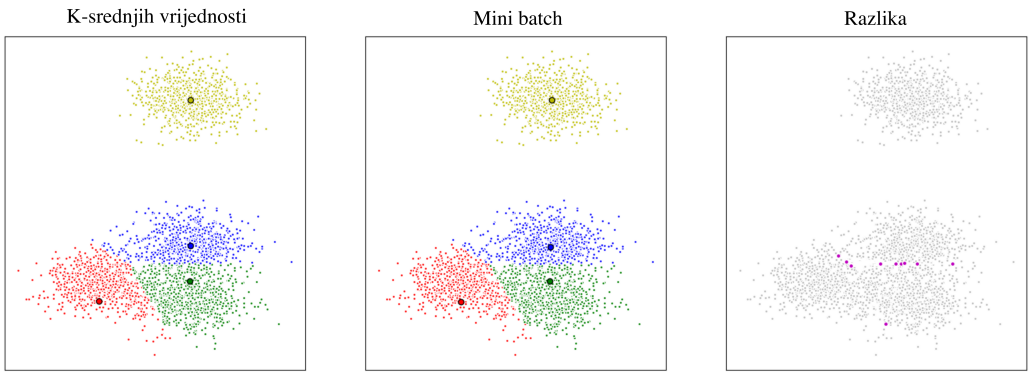
složenost, algoritam brzo konvergira prema stacionarnom stanju, pogotovo ako su početne srednje vrijednosti dobro izabrane.

Nedostatak ovog algoritma su lošiji rezultati za grupe različitih veličina, gustoća i grupe čiji oblik nije konveksan.

2.2. Mini batch

Algoritam k-srednjih vrijednosti istovremeno radi na cijelom skupu ulaznih vrijednosti. Što je algoritam veći potrebno mu je više vremena i memorije. Mini batch je varijacija algoritma k-srednjih vrijednosti . Za razliku od k-srednjih vrijednosti, mini batch radi s malim podskupom podataka. Svaka nova iteracija uzima novi proizvoljan podskup podataka, na temelju kojeg određuje grupe i njihove centroide. Postupak se ponavlja dok se ne postigne stacionarno stanje ili se ne izvede unaprijed zadan broj iteracija.

Mini batch algorithm je brži u usporedbi s algoritmom k-srednjih vrijednosti, ali su mu rezultati neznatno lošiji.



Slika 2.2: Razlika učinkovitosti algoritma srednjih vrijednosti i Mini batch algoritma

2.3. Hijerarhijsko aglomerativno grupiranje

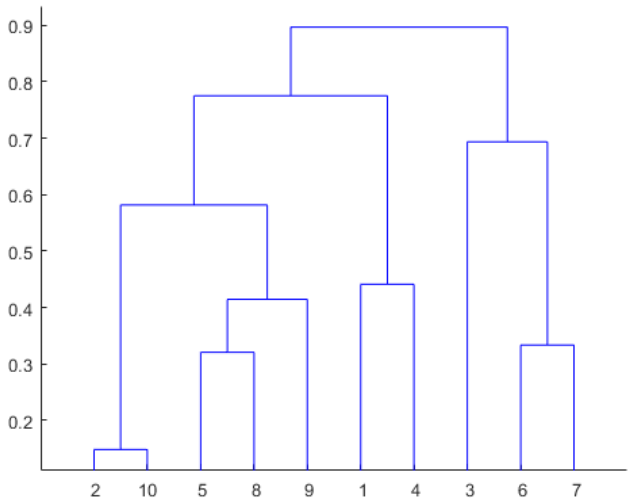
Hijerarhijsko grupiranje razdjeljuje skup primjera u ugniježđene grupe koje sačinjavaju hijerarhiju grupa. Za razliku od algoritma k-srednjih vrijednosti, ovo grupiranje je heuristički postupak, jer nema teorijsku osnovu. Dijeli se na divizivno i aglomerativno hijerarhijsko grupiranje.

Divizno grupiranje kreće od jedne grupe u kojoj su sve ulazne vrijednosti i postepeno razdjeljuje tu grupu u manje grupe. Suprotno tome, aglomerativno grupiranje

započinje sa svakom ulaznom vrijednošću u svojoj posebnoj grupi. Na kraju iteracije stapa dvije najbliže grupe, dok se ne dostigne K grupa.

$$
\begin{aligned}
&1.\ \textbf{inicijaliziraj}\ K, k \leftarrow N,\ \mathcal{G}_i \leftarrow \{\mathbf{x}^{(i)}\}\ za\ i = 1, ..., N \\
&2.\ \textbf{ponavljaj} \\
&3. \qquad k \leftarrow k - 1 \\
&4. \qquad (\mathcal{G}_i, \mathcal{G}_j) \leftarrow \underset{\mathcal{G}_a, \mathcal{G}_b}{\operatorname{argmin}}\, d(\mathcal{G}_a, \mathcal{G}_b) \\
&5. \qquad \mathcal{G}_i \leftarrow \mathcal{G}_i \cup \mathcal{G}_j \\
&6.\ \textbf{dok je}\ k\ >\ K
\end{aligned}
\tag{2.1}
$$

Rezultati hijerarhijskog grupiranja prikazuju se dendrogramom.



Slika 2.3: Primjer dendrograma

2.3.1. Kriterij povezanosti

Način računanja udaljenosti među skupovima podataka $d(\mathcal{G}_a, \mathcal{G}_b)$ određen je kriterijem povezanosti.

Grupiranje jednostrukom povezanošću (engl. *single-link clustering*) za mjeru koristi najmanju udaljenost između pojedinačnih primjera u tim grupama $d_{min}(\mathcal{G}_a, \mathcal{G}_b)$. dok potpuna povezanost (complete-link clustering) koristi najveću udaljenost za mjeru $d_{max}(\mathcal{G}_a, \mathcal{G}_b)$.

$$d_{min}(\mathcal{G}_a, \mathcal{G}_b) = \min_{\mathbf{x} \in \mathcal{G}_i, \mathbf{x'} \in \mathcal{G}_i} d(\mathbf{x}, \mathbf{x'}) \tag{2.2}$$

$$d_{max}(\mathcal{G}_a, \mathcal{G}_b) = \max_{\mathbf{x} \in \mathcal{G}_i, \mathbf{x'} \in \mathcal{G}_i} d(\mathbf{x}, \mathbf{x'}) \tag{2.3}$$

Ovisno o kompaktnosti grupe, ove dvije udaljenosti mogu dati približno jednake ili značajno različite rezultate. Ako je razlika u rezultatima velika, tada jednostruko povezivanje rezultira dugim ulančanim grupama, dok potpuno rezultira manjim grupama. Jednostruko i potpuno povezivanje su dva krajnja slučaja i oba su osjetljiva na šum, zato je uvedeno grupiranje temeljem prosječne povezanosti.

Prosječna udaljenost

Prosječna udaljenost se računa pomoću izraza (2.4.), u kojem N_i i N_j predstavljaju broj podataka u grupama $\mathcal{G}_a$ i $\mathcal{G}_b$.

$$d_{avg}(\mathcal{G}_a, \mathcal{G}_b) = \frac{1}{N_i N_j} \sum_{\mathbf{x} \in \mathcal{G}_i} \sum_{\mathbf{x'} \in \mathcal{G}_i} d(\mathbf{x}, \mathbf{x'}) \tag{2.4}$$

Ward

Postoje i drugi kriteriji povezanosti kao što je primjerice Wardova metoda. Ward uzima promjenu kvadarata udaljenosti točaka od središta grupe kao udaljenost između grupa.

$$\begin{aligned} \Delta(A, B) &= \sum_{i \in A \cup B} \| \overrightarrow{x}_i - \overrightarrow{m}_{A \cup B} \|^2 - \sum_{i \in A} \| \overrightarrow{x}_i - \overrightarrow{m}_A \|^2 - \sum_{i \in B} \| \overrightarrow{x}_i - \overrightarrow{m}_B \|^2 \\ &= \frac{n_A n_B}{n_A + n_B} \| \overrightarrow{m}_A - \overrightarrow{m}_B \|^2 \end{aligned} \tag{2.5}$$

2.4. DBSCAN

DBSCAN (engl. *density-based algorithm*) je nešto složenije grupiranje, od prethodno opisanih. Algoritam grupira podatke na osnovu gustoće. Predaju mu se dva parametra: ϵ i minimalan broj točaka u ϵ okolini, $MinPts$. Gustoća se definira kao broj pojava unutar kruga polumjera ϵ. Prema njoj se podaci dijele na središnje, granične i šum.

Središnja pojava u svojoj ϵ okolini mora imati barem minimalan broj točaka, $MinPts$. Ako nema, a nalazi se u ϵ okolini neke druge središnje pojave, tada se svrstava u granične pojave. Sve ostale pojave su šum.

Ovaj algoritam se izvodi u nekoliko koraka:

1. Svim točkama se dodjeljuje jedna od tri kategorije: središnja, granična ili šum.
2. Šum se uklanja.
3. Između središnjih pojava, koje su jedna drugoj unutar ϵ okolina, postavlja se brid.
4. Granične pojave se grupiraju s obzirom na njihovu središnju točku.

2.5. Model miješane gustoće

Model miješane gustoće pokušava prikazati ulazni skup podataka kao miješavinu Gaussovih razdioba.

2.5.1. Normalna ili Gaussova razdioba

Funkcija gustoće razdiobe za Gaussa u jednoj dimenziji dana je izrazom 2.6, gdje je μ očekivanje, a σ varijanca.

$$G(X|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2} \tag{2.6}$$

Za više dimenzija funkcija gustoće razdiobe je dana izrazom 2.7., gdje je Σ kovarijacijska matrica.

$$G(X|\mu,\Sigma) = \frac{1}{(\sqrt{(2\pi)}|\Sigma\|} exp(-\frac{1}{2}(X-\mu)^T\sigma^{-1}(X-\mu) \tag{2.7}$$

Za procjenu parametara $\pi,\ \Sigma,\ \mu$ maksimizira se funkcija log-izgledanosti (2.9).

$$p(X) = \sum_{k=1}^{K} \pi_k G(X|\mu_k,\Sigma_k) \tag{2.8}$$

$$ln\ p(X|\mu,\Sigma,\pi) = \sum_{I=1}^{N} P(X_i) \tag{2.9}$$

$$= \sum_{i=1}^{N} ln \sum_{k=1}^{K} \pi_k G(X|\mu_k,\Sigma_k) \tag{2.10}$$

Nakon toga se uvodi slučajna varijabla $\gamma_k(X) = p(k|X)$ i Bayesovim teoremom dobijamo izraz 2.12.

$$\gamma_k = \frac{p(X|k)\ p(k)}{\sum_{k=1}^{K} p(k)\ p(X|k)} \tag{2.11}$$

$$= \frac{(X|k)\pi_k}{\sum_{k=1}^{K} \pi_k\ p(X|k)} \tag{2.12}$$

Naposljetku se parametri računaju iz sljedećih izraza:

$$\mu_k = \frac{\sum_{n=1}^{N} \gamma_k(x_n) x_n}{\sum_{n=1}^{N} \gamma_k(x_n)} \tag{2.13}$$

$$\Sigma_k = \frac{\sum_{n=1}^{N} \gamma_k(x_n)(x_n-\mu_k)(x_n-\mu_k)^T}{\sum_{n=1}^{N} \gamma_k(x_n)} \tag{2.14}$$

$$\pi_k = \frac{1}{N} \sum_{n=1}^{N} \gamma_k(x_n) \tag{2.15}$$

2.5.2. Algoritam maksimizacije očekivanja

Algoritam maksimizacije očekivanja (engl. *expectation maximization algorithm*) ili EM-algoritam je optimizacijski postupak za problem najveće izglednosti kod modela s latentnim (skrivenim) varijablama. Sastoji se od dva osnovna koraka: E i M.

E-korak

Prvo E-korak inicijaizira parametre $\pi,\ \Sigma,\ \mu$ nasumičnim vrijednostima ili rezultatima algoritma k-srednjih vrijednosti. Zatim se izračuna γ_k i. s tom novom vrijednošću se procjene svi parametri.

M-korak

U ovom koraku računamo log-izglednost. Ako vrijednost te funkcije ili svih parametara konvergira, algoritam staje, inače ponovno počinje od E-koraka.

3. Opis postupka vrednovanja i rezultati

3.1. Unutarnja vrednovanja

Postupak vrednovanja učinkovitosti algoritama grupiranja se dijeli na: unutarnje i vanjsko vrednovanje. Razlika između ta dva vrednovanja je u inforamacijama koje koriste. Vanjsko vrednovanje koristi dodatne informacije koje se ne mogu pronaći u dobivenim podacima,ova već se te informacije predaju kao dodatne vrijednosti uz ulazne podatke.

No u praksi najčešće nisu dostupne dodatne informacije, tada se koriste postupci unutarnjeg vrednovanja. Suprotno vanjskom ono se oslanja samo na ulazne podatke, bez dodatnih informacija. Dva osnovna svojstva unutarnjeg vrednovanja su kompaktnost i razdvojenost.

Kompaktnost određuje koliko su bliski članovi neke grupe. Postoje dva načina određivanja te bliskosti. Prvi način koristi varijacije članova s obzirom na ostale članove grupe. Manja varijacija upućuje na veću kompaktnost. Drugi način je mjerenje udaljenosti članova unutar grupe. Koristi se međusobna udaljenost članova grupe ili njihova udaljenost s obzirom na središte grupe.

Razdvojenost mjeri koliko su grupe različite jedna od druge. Dobri pokazatelji razdvojenosti su udaljenost između središta grupa i najmanja udaljenost između elemenata različitih grupa.

3.1.1. Dunnov indeks

Dunnov indeks je izražen omjerom najmanje međusobne udaljenosti grupa i najvećeg promjera grupe. Promjer grupe može se definirati na više načina. Jedna od dobrih definicija je udaljenost između dvije najudaljenije točke neke grupe:

$$\Delta_i = \max_{x,y \in C_i} d(x, y) \tag{3.1}$$

Drugi način definiranja udaljenosti je srednja vrijednost udaljenosti svih parova u grupi:

$$\Delta_i = \frac{1}{|C_i|(|C_i|-1)} \sum_{x,y \in C_i, x \neq y} d(x,y), \tag{3.2}$$

Posljednja definicija je udaljenost svih točaka u grupi od središta grupe:

$$\Delta_i = \frac{\sum_{x \in C_i} d(x,\mu)}{|C_i|}, \mu = \frac{\sum_{x \in C_i} x}{|C_i|} \tag{3.3}$$

Slično se na više načina može definirati međusobna udaljenost grupa. Neki od mogućih opcija su: najbliže točke grupa, najudaljenije točke grupa, udaljenosti središta grupa, itd.

Dunnov index definira se sljedećim izrazom:

$$DI_m = \frac{\min_{1 \leq i < j \leq m} \delta(C_i, C_j)}{\max_{1 \leq k \leq m} \Delta_k} \tag{3.4}$$

gdje je $\delta(C_i, C_j)$ međusobna udaljenost grupa, a Δ_k promjer pojedine grupe. Bez obzira na odabrani način definiranja ovih udaljenosti, optimalni rezultat određen je najvećom vrijednošću Dunnovog indeksa.

3.1.2. Silhouette indeks

Silhouette indeks ocjenjuje učinkovitost algoritma grupiranja na osnovu razlike udaljenosti među objektima grupe i udaljenosti među samim grupama. Vrijednost indeksa se kreće između -1 i 1. Poželjno je da većina točaka ima visoku vrijednost indeksa, jer ona ukazuje da je točka bliska svojoj grupi. U slučaju da puno točaka ima niski indeks, točke su grupirane u premalo ili previše grupa.

Silhouette indeks se izračuna izrazom (3.5). Za svaku točku $i \in C_i$ izračuna se srednja vrijednost udaljenosti točke od svih ostalih točaka. Suma svih udaljenosti se dijeli s brojem preostalih točaka. Vrijednost $a(i)$ predstavlja kvalitetu grupiranja točke i.

$$a(i) = \frac{1}{|C_i|-1} \sum_{j \in C_i, i \neq j} d(i,j) \tag{3.5}$$

Zatim je potrebno izračunati udaljenost svake grupe od zadane točke i i izabrati najmanju udaljenost za vrijednost $b(i)$:

$$b(i) = \min_{i \neq j} \frac{1}{|C_j|} \sum_{j \in C_j} d(i,j) \tag{3.6}$$

Grupa s najmanjom udaljenosti od točke i se zove *susjedna grupa*.
$a(i)$ i $b(i)$ uvrštavamo u izraz (3.7) kako bi dobili vrijednost Silhouette indeksa.

$$s(i) = \begin{cases} \frac{b(i)-a(i)}{max\{a(i),b(i)\}}, & \text{za } |C_i| > 1 \\ 0, & \text{za } |C_i| = 0 \end{cases} \tag{3.7}$$

što se može zapisati i ovako:

$$s(i) = \begin{cases} 1 - a(i)/b(i), & \text{za } a(i) < b(i) \\ 0, & \text{za } a(i) = b(i) \\ a(i)/b(i) - 1, & \text{za } a(i) > b(i) \end{cases} \tag{3.8}$$

Iz izraza (3.8.) se može zaključiti sljedeće:

$$-1 \leq s(i) \leq 1 \tag{3.9}$$

Razlikuju se 3 slučaja: $s(i) < 1$ $s(i) = 0$ i $s(i) > 1$. Ako je $s(i) = 0$tada je promatrana točka jednako udaljena od svoje i od susjedne grupe, s(i) < 1 znači da je $a(i) \gg b(i)$, odnosno da je točka krivo grupirana jer je puno bliže susjednoj grupi nego svojoj grupi. Poželjno je da $s(i)$ teži u 1, da je $a(i) \ll b(i)$, kako bi točka bila puno bliža svojoj grupi nego susjednoj.

3.1.3. Davies-Bouldin indeks

Davies-Bouldin index je omjer udaljenosti među grupama i raspršenosti točaka unutar samih grupa. Računanje Davies-Bouldin indexa započinje izračunom koeficijenta raspršenosti objekata unutar grupe S_i. Raspršenost se računa pomoću izraza (3.10) u kojem X_j predstavlja proizvoljnu točku, A_i središte, a T_i veličinu grupe C_i. Za $p = 2$ izraz postaje Euklidova norma. Poželjno je da grupa bude što kompaktnija, odnosno da raspršenost bude što manja.

$$S_i = (\frac{1}{T_i} \sum_{j=1}^{T_i} |X_j - A_i|^p)^{\frac{1}{p}} \tag{3.10}$$

Osim S_i potrebno je izračunati i udaljenost između samih grupa, $M_{i,j}$. Udaljenost se uzima kao razlika A_i i A_j, koji predstavljaju središta grupa C_i i C_j. Računa se tako da se zbrajaju razlike parova točaka, $a_{k,i}$ i $a_{k,j}$ Ovaj izraz, kao i S_i, za $p = 2$ postaje Euklidova norma. Poželjno je da udaljenost među grupama bude što veća.

$$M_{i,j} = \|A_i - A_j\|_p = (\sum_{k=1}^{n} |a_{k,i} - a_{k,j}|)^p)^{\frac{1}{p}} \tag{3.11}$$

Mjera dobrog grupiranja $R_{i,j}$ (3.12) je omjer zbroja raspršenosti grupa S_i i S_j i njihove međusobne udaljenosti $M_{i,j}$.

$$R_{i,j} = \frac{S_i + S_j}{M_{i,j}} \tag{3.12}$$

Za dobro grupirane podatke raspršenosti S_i i S_j su male, udaljenosti samih grupa $M_{i,j}$ su velike. Iz toga slijedi da bi mjera dobrog grupiranja trebala težiti nuli i da ima sljedeća svojstva:

$$\begin{aligned}
&1.\ R_{i,j} \geq 0 \\
&2.\ R_{i,j} = R_{j,i} \\
&3.\ Za\ S_j \geq S_k\ i\ M_{i,j} = M_{i,k}\ vrijedi\ da\ je\ R_{i,j} > R_{i,k} \\
&4.\ Za\ S_j = S_k\ i\ M_{i,j} \leq M_{i,k}\ vrijedi\ da\ je\ R_{i,j} > R_{i,k}
\end{aligned} \tag{3.13}$$

Zatim se za svaku grupu i izdvaja najgori, odnosno najveći, $R_{i,j}$ kao D_i(3.14).

$$D_i = \max_{j \neq i} R_{i,j} \tag{3.14}$$

Naposljetku se Davies-Bouldin indeks računa kao prosječna vrijednost tih izdvojenih mjera dobrog grupiranja D_i (3.15).

$$DB = \frac{1}{N} \sum_{i=1}^{N} D_i \tag{3.15}$$

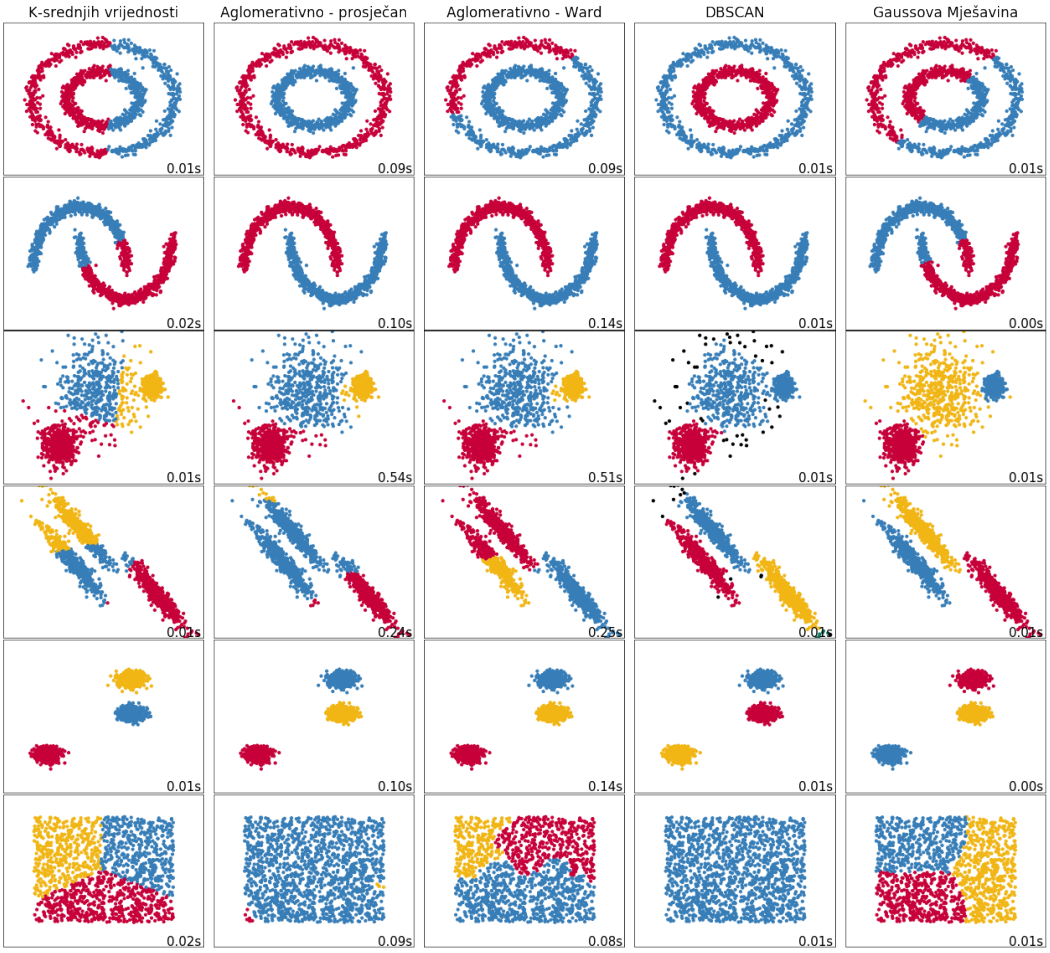
3.2. Usporedba učinkovitosti algoritama

Algoritam k-srednjih vrijednosti je najčešće korišteni algoritam, jer se svrstava u brže i jednostavnije algoritme grupiranja. Ipak, premda ima svoje prednosti, nije najpreciziji algoritam. Na slici 3.14 može se vidjeti da algoritam K-srednjih vrijednosti uvijek traži konveksne skupove. Zbog toga mu je teško prepoznati koncentrične kružnice ili grupe u obliku polumjeseca. Također grupe koje stvara najčešće poprimaju oblik kruga, zato ne može prepoznati duguljaste grupe u četvrtom primjeru. Zadnji nedostatak ovog algoritma jest potreba za unaprijed zadanim brojem grupa. Posljednji primjer sa slike 3.14 je jedna velika grupa, međutim algoritam pronalazi tri grupe, jer mu je tako zadan k. To znači da je za ovom algoritmy potrebno da su grupe koje treba pronaći konveksni, kružni skupovi podataka i da je njihov broj unaprijed poznat.

Hijerarhijsko aglomerativno grupiranje je pogodan za grupiranje konveksnih i nekonveksnih skupova podataka. Glavni nedostatak mu je što podatke grupira znatno sporije od ostalih algoritama, što je i vidljivo na slici 3.14. Zbog toga nije pogodan za grupiranje velikih skupova podataka.

DBSCAN algoritam daje dobre rezultate u kratkom vremenu. Uz to, ne zahtjeva unaprijed određeni broj grupa, pa nema problem s grupiranjem podataka bez strukture, kao što je vidljivo u posljednjem primjeru. Poteškoće nastaju kad grupira raspršene podatke, jer grupe određuje na osnovu slične gustoće.

Model Gaussove mješavine (GMM) oblikuje grupe kao realizacije slučajnih varijabli Gaussove razdiobe. Gaussova krivulja je konveksna i nikako se ne može prilagoditi nekonveksnim skupovima, kao što su koncentrične kružnice ili polumjeseci. Međutim, algoritam vrlo uspješno grupira sve konveksne skupove podataka, čak i ako nisu u kružnom obliku.



Slika 3.1: Usporedba rezultata grupiranja algoritmima

3.2.1. Vrijednosti indeksa

Indeksi unutarnje provjere su dobar pokazatelj kvalitete grupiranja. Vrijednosti Davies-Bouldinovog i Dunnovog indexa, te Silhouette koeficijenta se nalaze u tablici 3.1. Nji-

hovi rezultati uglavnom odgovaraju očekivanim vrijednostima, uz neka odstupanja. Uzroci odstupanja kod ova tri indeksa su šum i postojanje podgrupa.

Tablica 3.1: Unutarnja provjera

	Indeksi	**K**	**HA-prosjek**	**HA-Ward**	**DBSCAN**	**GMM**
Koncentrične	**DBI**	1.189	857.711	1.361	989.794	**1.194**
kružnice	**DI**	0.006	**0.064**	0.018	**0.069**	0.007
	SC	0.349	0.114	0.268	0.114	**0.351**
Polumjeseci	**DBI**	**0.805**	1.023	1.023	1.023	**0.804**
	DI	0.006	**0.157**	**0.157**	**0.157**	0.013
	SC	**0.500**	0.389	0.389	0.389	**0.500**
Raspršene	**DBI**	**0.633**	**0.642**	**0.644**	2.222	**0.662**
grupe	**DI**	0.014	**0.042**	**0.042**	0.013	0.006
	SC	**0.625**	**0.615**	**0.613**	0.532	0.596
Anisotropske	**DBI**	0.702	**0.526**	0.650	3.798	0.850
grupe	**DI**	0.002	0.010	0.009	0.009	**0.033**
	SC	**0.509**	0.429	0.477	0.397	0.472
Kružne konv.	**DBI**	0.270	0.270	0.270	0.270	0.270
grupe	**DI**	0.317	0.317	0.317	0.317	0.317
	SC	0.810	0.810	0.810	0.810	0.810

4. Zaključak

Kod grupiranja velike količine neoznačenih podataka, potrebni su algoritmi koji će podatke samostalno grupirati po nekim zajedničkim karakteristikama. Optimalni rezultati dobivaju se odabirom algoritma s obzirom na osobine ulaznih podataka.

Za manju količinu podataka pogodno je hijerarhijsko grupiranje. Za veće količine podataka, koji imaju konveksne skupove, algoritam k-srednjih vrijednosti i model Gaussove mješavine daju dobre rezultate. Većinu podataka najbolje grupira DBSCAN. On nije osjetljiv na količinu i oblik, ali je zato osjetljiv na različite gustoće grupa.

Bez obzira na odabrani algoritam, potrebno je provjeriti njegovu učinkovitost. Najbolju povratnu informaciju daju indeksi, samo treba uzeti u obzir da su neki od njih osjetljivi na šum, različitu gustoću grupa, podgrupe ili različite veličine grupa i da se u takvim slučajevima mogu dobiti neprecizni rezultati.

LITERATURA

[1]Pang-Ning Tan Michael Steinbach Vipin Kumar: Introduction to Data Mining, Pearson, New York, 2006

[2] Odilia Yim, Kylee T. Ramdeen: Hierarchical Cluster Analysis: Comparison of Three Linkage Measures and Application to Psychological Data, University of Ottawa, 2015

[3] Cosma Rohilla Shalizi: Distances between Clustering, Hierarchical Clustering, Carnegie Mellon University, Pittsburgh, 2009

[4] Yanchi Liu, Zhongmou Li, Hui Xiong, Xuedong Gao1, Junjie Wu: Understanding of Internal Clustering Validation Measures, University of Science and Technology Beijing, China, 2010

[5] Ethem Alpaydin: Introduction to Machine Learning, The MIT Press, Cambridge, Massachusetts London, England, 2014.

[6] Jan Šnajder, Bojana Dalbelo Bašić, Strojno učenje, Sveučilište u Zagrebu Fakultet elektrotehnike i računarstva, 2012.

[7] Clustering — scikit-learn 0.18.1 documentation, `http://scikit-learn.org/stable/modules/clustering.html`

[] Jonathan Baarsch and M. Emre Celebi: Investigation of Internal Validity Measures for K-Means Clustering, Hong Kong, 2012

Usporedba algoritama grupiranja primjenom programske knjižnice Scikit-Learn

Sažetak

Algoritme grupiranja se mogu usporediti na više načina, neki od njih su: iscrtavanje podataka, mjerenje trajanja grupiranja, izračun indeksa unutarnjeg i vanjskog vrednovanja. Rezultati ove usporedbe pokazuju da je algoritam k-srednjih vrijednosti brz i jednostavan, ali sklon greškama, te da najbolje grupira konveksne skupove u obliku kruga. Hijerarhijsko aglomerativno grupiranje je puno preciznije, ali potrebno mu je više vremena, te ga ne možemo koristiti za grupiranje velike količine podataka. DBSCAN je brz i precizan i velika mu je prednost što ne zahtjeva unaprijed zadan broj grupa, ali nije jednako učinkovit za raspršene skupove podataka. Model miješane gustoće (GMM) je brz i učinkovit sa svim konveksnim skupovima podataka, čak i kad nisu u kružnom obliku, ali ne grupira dobro nekonveksne skupove podataka, jer im ne može prilagoditi Gaussovu krivulju.

Ključne riječi: grupiranje, algoritam k-srednjih vrijednosti, DBSCAN, hijerarhijsko grupiranje, aglomerativno grupiranje, Dunnov indeks, Davies-Bouldin indeks, Silhouette koeficijent, Scikit-Learn, model miješane gustoće, GMM, Ward, kriterij povezanosti, mini batch

Comparison of clustering algorithms using the Scikit-Learn library

Abstract

There are many ways to compare clustering algorithms. Some of the ways it can be done are by plotting the results, timing the algorithms, using internal and external validation indices. Comparison has shown that k-means algorithm is fast and efficient but is prone to mistakes, unless datasets are convex and shaped as a circle. Hierarchical agglomerative clustering is much more efficient, but slower, so it is not fit for large datasets. DBSCAN is also fast and efficient. It doesn't require a number of clusters. Unfortunately it is not as efficient for clustering varied data. GMM is just as good for any type of convex datasets, even if they're not shaped as a circle however it can never fit Gaussian to a non-convex dataset, therefore it can never properly group them.

Keywords: clustering, k-means algorithm, DBSCAN, hierarchical clustering, agglomerative clustering, Dunn index, Davies-Bouldin index, Silhouette score, Scikit-Learn, Gaussian mixture model, GMM, Ward, linkage criteria