Klasterovanje 101

Milan Ilić 18/2015

Jun 2019

Sažetak

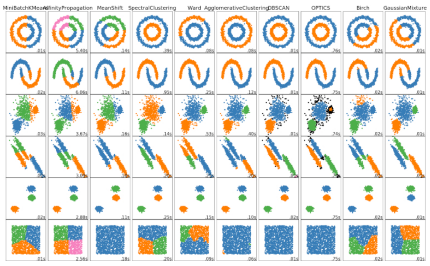
U ovom radu biće izvršene dve tehnike klasterovanja nad medicinskim skupom podataka koji sadrži podatke o genima i ćelijama koštane srži. Pre samog klasterovanja biće odrađena priprema i analiza skupa, kako bi se čitalac upoznao sa strukturom datih podataka. Na kraju rada biće reči o tome kako je moguće iskorisiti rezultate klasterovanja pri donošenju zaključaka o instancama skupa.

Sadržaj

1 Uvod

Klasterovanje predstavlja identifikaciju grupa u datim podacima. Jedno je od metoda nenadgledanog učenja i praktične primene se mogu pronaći u medicini, marketingu, lingvistici, biologiji, i drugim naučnim poljima. Postoje razne tehnike klasterovanja, od kojih su k-sredina, DBSCAN algoritam i algoritmi hijerarhijskog klasterovanja neki od najznačajnijih. Kao i u ostalim metodama nenadgledanog učenja, ne postoje obeleženi podaci, već je skup podataka dat kao matrica gde svaki red predstavlja jednu instancu podatka, dok su vrednosti kolona osobine podatka. Vizualizacija klasterovanja se može videti na slici 1.

U ovom radu biće predstavljen ceo put od sirovih podataka do klasterovanja na prilagođenim podacima skupa. Korišćen je python programski jezik i scikit-learn biblioteka za analizu, pretprocesiranje i klasterovanje podataka. Kôd korišćen za pripremu ovog rada, nalazi se na sledecem linku.



Slika 1: Vizualizacija razlicitih tehnika klasterovanja.

2 Analiza skupa podataka

Skup podataka nad kojim će se izvršiti klasterovanje u ovom radu jeste skup pod nazivom *"Multiple myeloma from he bone marrow from MM135 patients"*. U skupu podataka se nalaze podaci o ekspresiji 31221 različitih gena u 1339 ćelija. Ekspresija gena jeste numerička vrednost, pa su svi atributi skupa podataka isključivo numeričke vrednost.

Sledeća dva ispisa 1 2 koda prikazuju prih 5 redova baze i neke bitne statistike po kolona. Ceo postupak će biti prikazan i na originalnom skupu i na transponovanom skupu, tj. na skupu gde su redovi celije, a atributi aktivacije gena 3.

```
                1  2  3  4  5  ...  1334  1335  1336  1337  1338
     1339
hg38_A1BG       0  0  0  0  0  ...     0     0     0     0     0
        1
hg38_A1BG-AS1   0  0  0  0  0  ...     0     0     0     0     0
        0
hg38_A1CF       0  0  0  0  0  ...     0     0     0     0     0
        0
hg38_A2M        0  0  0  0  0  ...     0     0     1     0     0
        0
hg38_A2M-AS1    0  0  0  0  0  ...     0     0     0     0     0
        0

[5 rows x 1339 columns]
```

Listing 1: Ispis metode **.head()** *DataFrame* objekta koji sadži bazu.

```
                    1              2  ...           1338           1339
count   31221.000000   31221.000000  ...   31221.000000   31221.000000
mean        0.122482       0.374459  ...       0.121168       0.168060
std         1.954590       7.111641  ...       2.431342       2.098726
min         0.000000       0.000000  ...       0.000000       0.000000
25%         0.000000       0.000000  ...       0.000000       0.000000
50%         0.000000       0.000000  ...       0.000000       0.000000
75%         0.000000       0.000000  ...       0.000000       0.000000
max       187.000000     775.000000  ...     315.000000     170.000000

[8 rows x 1339 columns]
```

Listing 2: Ispis metode **.describe()** *DataFrame* objekta koji sadrži bazu

```
          hg38_A1BG  hg38_A1BG-AS1  ...     hg38_ZYX   hg38_ZZEF1
count  1339.000000    1339.000000  ...  1339.000000  1339.000000
mean      0.143391       0.004481  ...     0.132188     0.066468
std       0.464324       0.066815  ...     0.401420     0.255119
min       0.000000       0.000000  ...     0.000000     0.000000
25%       0.000000       0.000000  ...     0.000000     0.000000
50%       0.000000       0.000000  ...     0.000000     0.000000
```

```
8  75%        0.000000        0.000000   ...      0.000000      0.000000
9  max        5.000000        1.000000   ...      4.000000      2.000000
```

Listing 3: Ispis metode **.describe()** *DataFrame* objekta koji sadrži transponovanu bazu

Analizirajući ispis prethodnih metoda, može se zaključiti da su vrednosti ekspresija gena u celijama uglavnom 0, kao i da su proseci kolona jako mali. Ova informacija će pomoći pri pretprocesiranju podataka.

3 Pretprocesiranje podataka

Pretprocesiranje je izuzetno važan korak u spremanju podataka za klasterovanje jer algoritmi za klasterovanje računaju udaljenost podataka kako bi ih svrstali u određenu grupu. Zbog toga je korisno normalizovati (ili standardizovati) podatke i izbaciti elemente van granica, kao i druge tehnike koje se vrše tokom pretprocesiranja.

Zbog učestalosti nula u skupu, najpre ćemo prebrojati redove kod kojih je svaka vrednost jednaka nuli. Broj gena sa ovim svojstvom jeste 13639 (44% celog skupa). Ovi redovi se mogu izbaciti iz daljeg pretprocesiranja jer nemaju informativnu vrednost. Izbacivanjem ovih redova, trenutni skup podataka sadrži podatke o 17582 gena. U transponovanom slučaju, nijedan red nije sastavljen od 0 vrednosti za svaki gen.

Pre prelaska na uklanjanje elemenata van granica i redukciju dimenzionalnost, izvršena je standardizacija originalnog skupa podataka odnosno normalizacija transponovanog. Standardizacija jeste svođenje srednje vrednosti svake kolone na 0, a disperzije na 1, dok je normalizacija transformacija kolona u opseg [0, 1].

Standardizacija:

$$z = \frac{X - \mu}{\sigma}$$

Normalizacija:

$$z = \frac{X - min(X)}{max(X) - min(X)}$$

3.1 Uklanjanje elemenata van granica

Udaljenost između podataka igra važnu ulogu pri klasterovanju podataka, te podaci koji su značajno udaljeni u odnosu na druge mogu da utiču na krajnji rezultat klasterovanja. Takođe, prikupljanja medicinskih podataka o genima unutar ćelija je jako osetljivo i moguća je pojava grešaka. Iz ovih razloga, u ovom poglavlju će biti prikazano uklanjanje podataka van granica.

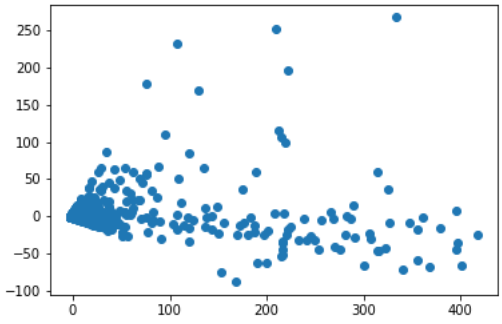
Zbog ne-normalne raspodele podataka i raštrkanih ne-nula vrednosti, detekcija vrednosti van granica je izvršena vizualnom analizom podataka redukovanih na tri dimenzije. Dobijeno je ukupno 20 gena koji su klasifikovani kao vrednosti van granica i oni su izbačeni iz skupa. U transponovanom skupu podataka nije doslo do pojave elemenata van granica.

Ovi redovi skupa podataka su interesantni za dalju analizu i određivanje uzroka postojanja ovakvih redova. Činjenica da su podaci medicinskog porekla povećava znacaj daljih analiza.

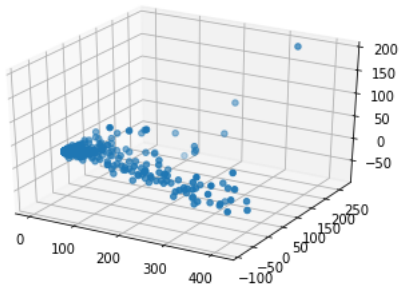
3.2 Redukcija dimenzionalnosti

Kako bi se metode klasterovanja izvršile u prihvatljivim vremenskim okvirima i izbacile redundantne kolone iz skupa podataka, vrši se neka od tehnika redukcije dimenzionalnosti poput PCA, metode slucajnih projekcija, autoenkodera, i drugih. U ovom radu je za smanjivanje broja dimenzija korišćena tehnika PCA, i broj dimenzija je redukovan na broj koji čuva 95% disperzije podataka.

Redukcijom dimenzionalnosti originalnog skupa, svaki red je dobio 25 kolona u originalnom supu odnosno 42 u transponovanom skupu podataka. Kako bismo videli raspodelu podataka u prostoru, izvršena je i redukcija na dve i tri dimenzije. Dobijeni rezultati redukcijom prikazani su na slikama 2, 3, 4 i 5.

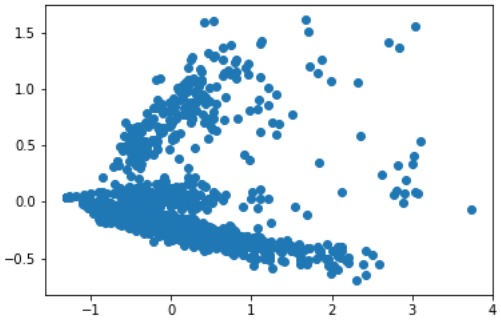


Slika 2: Redukcija originalnog skupa na dve dimenzije

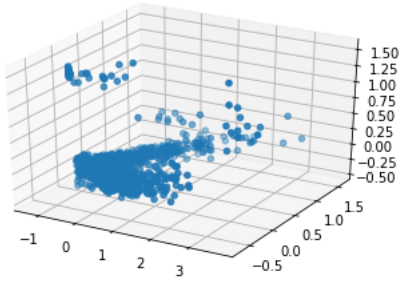


Slika 3: Redukcija originalnog skupa na tri dimenzije

Nakon pretprocesiranja, skup podataka je spreman za klasterovanje i sadrži 17563 redova i 25 kolona.



Slika 4: Redukcija transponovanog skupa na dve dimenzije



Slika 5: Redukcija transponovanog skupa na tri dimenzije

4 Klasterovanje

Nakon pretprocesiranja, trenutni skup podataka je spreman za klasterovanje. Od raznih tehnika klasterovanja, u ovom radu će biti prikazane tehnika hijerarhijskog klasterovanja i algoritam DBSCAN. Da bi se ocenio kvalitet rezultata, korišćena je metrika šilueta klastera".

4.1 Hijerarhijsko klasterovanje

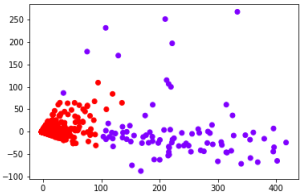
Hijerarhijsko klasterovanje se deli na hijerarhijsko klasterovanje odozdo nagore i hijerarhijsko klasterovanje odozgo nadole. U prvom slučaju, na početku algoritma su svi elementi skupa u svom posebnom klasteru i iterativno se spajaju najbliži klasteri dok se ne ispune uslovi zaustavljanja algoritma. Kod hijerarhijskog klasterovanja odozgo nadole, u početnom stanju su svi elementi u jednom klasteru i iterativno se klasteri dele na sve manje dok se ne ispune uslovi zaustavljanja. Na kraju oba algoritma dobijaja se skup podataka podeljen na određen broj klastera. Pre same upotrebe algoritma, kao uslov zaustavljanja se često koristi broj klastera na koje će dati skup biti podeljen.

Za ovu vrstu klasterovanja korišćen je algoritam *AgglomerativeCluste-*

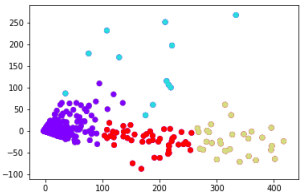
ring iz scikit-learn biblioteke. Algoritam je izvrsen za sve kombinacije parametara date u sledecoj listi:

- **n_clusters**: vrednosti iz skupa [2, 3, 4]
- **affinity**: vrednosti iz skupa ['euclidean', 'l1', 'l2', 'manhattan', 'cosine']
- **linkage**: vrednost iz skupa ['complete', 'average', 'single', 'ward']

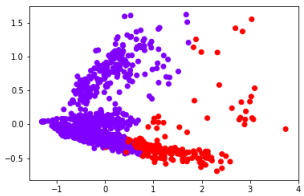
Neki od rezultata i odgovarajuca ocena siluete prikazani su na narednim slikama 6, 7, 8, 9.



Slika 6: Hijerarhijsko klasterovanje originalnog skupa 1, silueta je 0.99



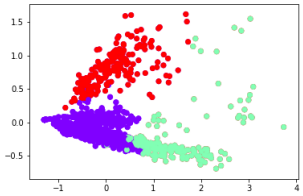
Slika 7: Hijerarhijsko klasterovanje originalnog skupa 2, silueta je 0.98



Slika 8: Hijerarhijsko klasterovanje transponovanog skupa 1, silueta je 0.43

4.2 DBSCAN klasterovanje

DBSCAN algoritam klasifikuje svaku instancu skupa u jednu od tri kategorije na osnovu parametara **eps** i **min_samples**:



Slika 9: Hijerarhijsko klasterovanje transponovanog skupa 2, silueta je 0.34

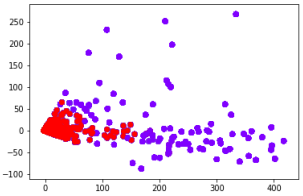
- Tačka je *tačka jezgra* ukoliko se u okolini **eps** nalazi najmanje **min_samples** drugih tacaka iz skupa.
- Tačka je *tačka na granici* ukoliko se u okolini **eps** nalazi manje od **min_samples** drugih tačaka iz skupa, ali se nalazi bar jedna.
- Inače, tacka je *šum*

Moče se videti da ne postoji parametar koji određuje broj klastera na izlazu, sto nije uobičajeno za algoritme klasterovanja.

Nad originalnim i transponovanim skupom podataka izvršen je algoritam *DBSCAN* iz scikit-learn biblioteke sa svim kombinacijama sledećih parametara:

- **eps**: vrednosti iz skupa [2, 4, 8, 16, 32]
- **min_samples**: vrednosti iz skupa [2, 4, 8, 16, 32]
- **metric**: vrednosti iz skupa ['cityblock', 'cosine', 'euclidean', 'l1', 'l2', 'manhattan']

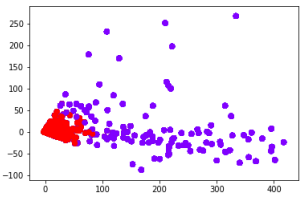
Neki od rezultata izvršavanja algoritma i odgovarajuća ocena siluete prikazani su na narednim slikama 10, 11, 12, 13.



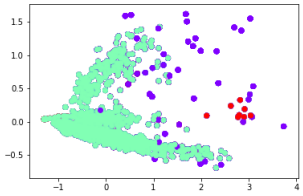
Slika 10: DBSCAN klasterovanje originalnog skupa 1, silueta je 0.98

5 Zaključak

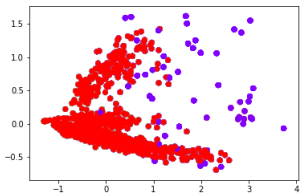
Nakon klaster analize, potebno je uraditi identifikiaciju klastera i šta oni predstavljaju u stvarnom svetu. Kod klasterovanja originalnog skupa, zbog velikog broja redova koji se redukcijom dimenzionalnosti svode na okolinu tačke (0, 0), silueta klasterovanja se jako malo razlikuje i uglavnom je iznad 0.98. Iz tog razloga, sve tačke koje su znatno udaljene od koordinatnog pocetka mogu se svrstati u gene za koje je potrebno dalje ispitivanje. Iz transponovanog skupa, mogu se izvojiti klasteri sa manjim brojem ćelija i odraditi se provera da li te ćelije imaju mutaciju.



Slika 11: DBSCAN klasterovanje originalnog skupa 2, silueta je 0.98



Slika 12: DBSCAN klasterovanje transponovanog skupa 1, silueta je 0.50



Slika 13: DBSCAN klasterovanje transponovanog skupa 2, silueta je 0.53