

Klasterovanje PBMC uzorka

Seminarski rad u okviru kursa

Istraživanje podataka 2

Matematički fakultet

Košanin Petar

mi15140@matf.bg.ac.rs

Gružanić Nemanja

mi16420@matf.bg.ac.rs

1. jun 2020.

Sažetak

PBMC (eng. Peripheral Blood Mononuclear Cells) su krvne ćelije kod kojih je prisutan jedan ovalni nukleus. PBMC se koriste u istraživanju raznih infektivnih bolesti, imunologiji, kao i u razvoju vakcina. U ovom radu predstavićemo nekoliko algoritama klasterovanja nad podacima dobijenim kombinovanjem više različitih skupova podataka različitih istraživanja.

Ključne reči: PBMC, klasterovanje spektralno klasterovanje, nenegativna faktorizacija

Sadržaj

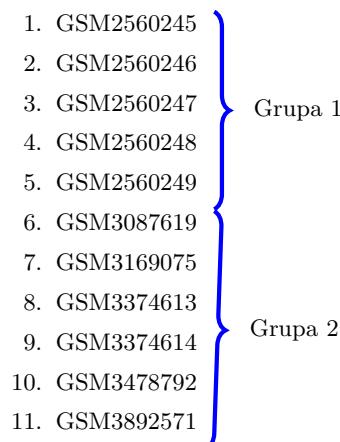
1	Uvod	3
2	Podaci	3
3	Preprocesiranje	3
3.1	Transformacija naziva gena	4
3.2	Transformacija naziva ćelija	4
3.3	Spajanje datoteka	4
3.4	Elementi van granica	4
4	Redukcija Dimenzionalnosti	4
4.1	Izuzetno promenljivi geni	4
4.2	Analiza glavnih komponenti	5
4.3	Nenegativna faktorizacija matrice	6
4.3.1	NMF nad tekstualnim podacima	6
4.3.2	NMF nad prvom grupom podataka	7
5	Vizuelizacija	7
6	Klasterovanje prve grupe podataka	8
6.1	K-sredina	8
6.2	Spektralno klasterovanje	9
6.2.1	Spektralno klasterovanje nad prvom verzijom prve grupe podataka . . .	10
6.2.2	Spektralno klasterovanje nad drugom verzijom prve grupe podataka . .	11
6.3	Hijerarhijsko klasterovanje	11
6.4	Klasterovanje zasnovano na gustini	14
6.4.1	DBSCAN	14
6.4.2	Određivanje parametara eps i minPts	15
6.4.3	DBSCAN nad prvom grupom podataka	15
6.5	BIRCH	16
7	Pregled dobijenih rezultata	17
8	Zaključak	18
A	Opisi datoteka koje čine skup podataka	19

1 Uvod

PBMC (eng. Peripheral Blood Mononuclear Cells) su krvne ćelije kod kojih je prisutan jedan ovalni nukleus. PBMC se koriste u istraživanju raznih infektivnih bolesti, imunologiji, kao i u razvoju vakcina. Podaci su sakupljeni iz više različitih izvora [9]. U nastavku biće reći o njihovojo inicijalnoj formi, načinu kako smo ih preprocesuirali, i konačno dobijeni rezultati klasterovanja.

2 Podaci

Skup podataka čini 11 datoteka i dve konsultacione datoteke. Podaci su podeljeni u dve grupe, pri čemu prvu grupu čine datoteke čiji nazivi počinju prefiksom GSM2, dok je prefiks druge grupe GSM3.

- 
1. GSM2560245
 2. GSM2560246
 3. GSM2560247
 4. GSM2560248
 5. GSM2560249
 6. GSM3087619
 7. GSM3169075
 8. GSM3374613
 9. GSM3374614
 10. GSM3478792
 11. GSM3892571

Konsultaciona datoteka **common_human_list** sadrži nazine ljudskih gena kao i njihove identifikatore. Isečak ove datoteke je prikazan u tabeli 1.

Tabela 1: Sadržaj konsultacione datoteke common_human_list

ENSG_ID	hg19	hg37	hg38	Ensembl_GRCh38.p12_rel94	GSM3717979
ENSG00000181638	hg19_ZFP41	grch37_ZFP41	grch38_ZFP41	#ZFP41	#ZFP41
ENSG00000111875	hg19 ASF1A	grch37 ASF1A	grch38 ASF1A	#ASF1A	#ASF1A
ENSG00000176142	hg19_TMEM39A	grch37_TMEM39A	grch38_TMEM39A	#TMEM39A	#TMEM39A
ENSG00000177186	hg19_OR2M7	grch37_OR2M7	grch38_OR2M7	#OR2M7	#OR2M7
ENSG00000135624	hg19_CCT7	grch37_CCT7	grch38_CCT7	#CCT7	#CCT7

Druga konsultaciona datoteka **SCT-10x-Metadata_readylist_merged-PBMC-tasks-short-Bgd.xlsx** sadrži metapodatke za datoteke koje čine skup podataka. U tabelama 4 i 5 prikazani su metapodaci redom za prvu i drugu grupu.

3 Preprocesiranje

Jedan od prvih problema koji se javlja je heterogenost podataka. Inicijalno, svaka datoteka za vrste ima gene, dok kolone predstavljaju ćelije. Broj gena, kao i broj ćelija, varira od datoteke do datoteke. Takođe, razlikuju se dve metode identifikacije gena. U nekim fajlovima geni su označeni njihovim ENSG-ovima(npr. ENSG00000243485), dok u drugim, označeni su njihovim nazivima(npr. RPS10). U nastavku biće opisani načini transformisanja naziva ćelija i gena. Zatim, podaci se transponuju, i izbacuju se ćelije i geni koji nezadovoljavaju kriterijume koji će biti opisani u nastavku. Opisane metode se koriste za konstrukciju velikog skupa podataka, kao i skupove podataka za svaku grupu pojedinačno.

3.1 Transformacija naziva gena

Za svaku datoteku određen je njen genom. Informacije o genomima nalaze se u konsultacionoj datoteci SCT-10x-Metadata_readylist_merged-PBMC-tasks-short-Bgd.xlsx. Tako na primer, za GSM2560245, genom je hg19 [4](#). Zatim, ažuriraju se identifikatori gena tako što se pravi novi identifikator spajanjem odgovarajućih vrednosti kolona ENSG.ID i kolone koja odgovara genomu datoteke koja se obrađuje iz konsultacione datoteke common_human_list. Za datoteku GSM2560245 i gen ZFP41, ažurirani identifikator je ENSG00000181638_ZFP41.¹ U slučaju da se gen javlja u nekoj datoteci, ali ne postoji u common_human_list, on biva eliminiran.

U konsultacionoj datoteci common_human_list, jedino je kolona ENSG.ID jedinstvena. To znači da postoji naziv gena kojem odgovara više različitih ENSG-ova. U ovim situacijama, izabrali smo prvi ENSG koji odgovara datom imenu gena, dok su drugi odbačeni.

3.2 Transformacija naziva celija

Nazivi celija se zamenjuju sa xxx_redniBroj, gde je xxx naziv datoteke. Na primer, ako je uzorak GSM2741551, celije su označene sa GSM2741551_1, GSM2741551_2, GSM2741551_3, ... Nakon transformisanja naziva celija i identifikatora gena, podaci se transponuju, celije postaju instance a geni atributi tako dobijenog skupa podataka.

3.3 Spajanje datoteka

Kako se broj gena razlikuje od datoteke do datoteke, odredili smo zajednički presek gena za sve datoteke. Upravo ti geni čine atribute velikog (eng. bulk) skupa podataka. Celije su se zatim iterativno dodavale na veliki skup podataka, zadržavajući vrednosti samo za zajedničke gene. Dimenzije tako dobijenog skupa podataka su 71095×12805 , tj. broj celija je 71095 a broj zajedničkih gena je **12805**.

3.4 Elementi van granica

Za veliki broj gena ne postoji neka celija sa nula vrednošću. Drugim rečima, postoje dosta nula kolona. Takođe, za neke gene postoji veoma mali broj celija kod kojih je očitana ekspresija tog gena. Slično pravilo važi i za celije. Zbog ovoga, eliminisali smo gene koji imaju manje od **%1** nula vrednosti u velikom skupu podataka. Izbačene su i celije koje ne zadovoljavaju bar jedan od sledeća dva uslova:

- Zbir transkriptata je veći od 1000
- Broj ne-nula gena je veći od 500

Tako na primeru grupe 1, primenom ovih pravila, dobijamo skup podataka dimenzija 15165×7545 .

4 Redukcija Dimenzionalnosti

Bioinformatički podaci mogu imati veliki broj atributa(gena). U našem slučaju, postoje datoteke sa preko 30 hiljada gena. Većina ovih gena su neinformativna, pri čemu je dosta njih ispunjeno nulama. Radi smanjenja vremena izvršavanja algoritama, pokušali smo da primenimo više različitih metoda redukcije dimenzionalnosti kao i metoda izbora atributa. U nastavku biće reči o izuzetno promenljivim genima(eng. Highly Variable Genes(HVG)), kao i o metodama nenegativne faktorizacije matrica(eng. Nonnegative Matrix Factorization(NMF)) i PCA.

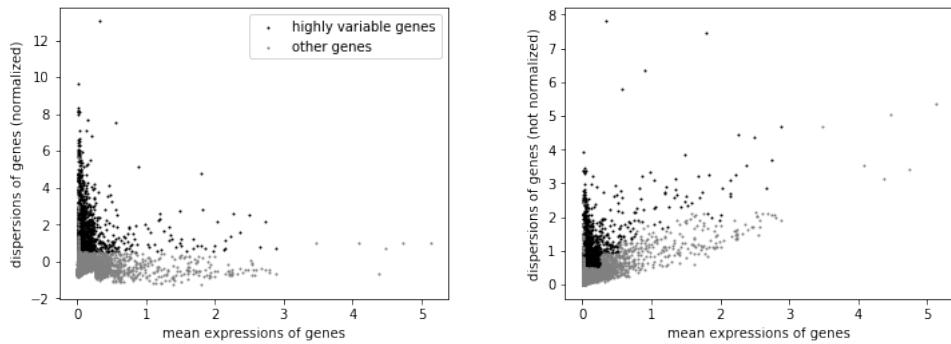
4.1 Izuzetno promenljivi geni

Prvi korak u smanjenju dimenzionalnosti podataka je izbor atributa(eng. feature selection). Izbor atributa pokušava da zadrži samo one atribute koji su *informativni*, dok atributi, poput nula kolona ili atributi niske varijabilnosti se izbacuju. Metoda koja se pokazala kao kvalitetna za ovaj korak preprocesiranja podataka jeste metoda *izuzetno promenljivih gena*(eng. Higly

¹Za datoteke čiji genome je grch38, imena gena su uzeta iz kolone Ensembl_GRCh38.p12_rel94 konsultacione datoteke.

variable Genes(HVG)). U zavisnosti od skupa podataka, bira se prvih 1000 do prvih 5000 HVG-a [5]. Izbor HVG gena se vrši na sledeći način: za svaki gen se računa njegova prosečna ekspresija, a zatim se geni rasporede u korpe(eng. bins) na osnovu njihovih prosečnih ekspresija. Iz svake korpe se bira gen sa najvećim odnosom varijanse i prosečne vrednosti ekspresije. Upravo ti geni su HVG.

Prvenstveno, naše podatke smo log-transformisali, i zatim odredili HVG gene pomoću ScanPy biblioteke. Na primeru prve grupe podataka, dobili smo 1030 HVG-a.

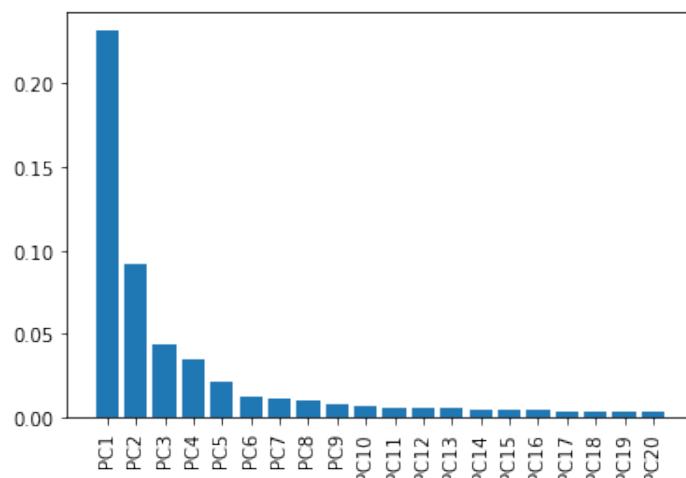


Slika 1: Slike prikazuju odnos varijanse i prosečne ekspresije gena prve grupe podataka. Na slikama su crnom bojom obojene vrednosti HVG gena, a sivom vrednosti ostalih gena.

4.2 Analiza glavnih komponenti

Analiza glavnih komponenti(eng. Principal component Analysis) se pokazala kao kvalitetna metoda redukcije dimenzionalnosti i kao takva nalazi široke primene.

Na slici 2 prikazan je procenat varijansi koje svaka od prvih 20 komponenti opisuje. Možemo primetiti da prva komponenta opisuje približno 25% varijanse, a da nakon osme komponente, koločina varijanse koje ostale komponente opisuju je zanemarljiva. Da bi se opisalo 80% varijansi skupa podataka, potrebno je čak 267 komponenti. U ovom slučaju, sve komponente počev od deveta se mogu odbaciti, ali problem nastaju u tome što prvih osam komponenti opisuje svega 47% varijanse. Zbog ovoga, u ovom radu nismo vršili detaljnije istraživanje nad PCA reprezentacijom podataka.



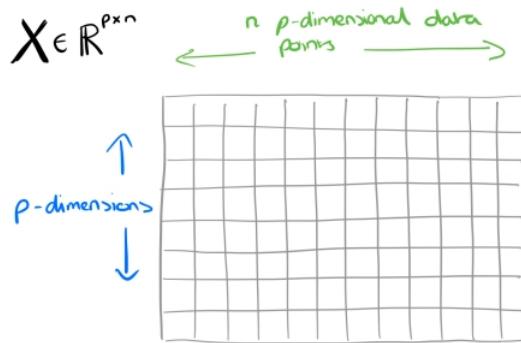
Slika 2: Procenat opisane varijansi prvih 20 komponenti dobijenih sa PCA

4.3 Nenegativna faktorizacija matrice

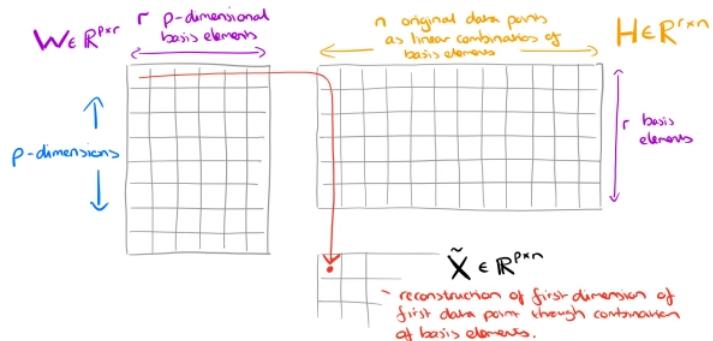
Nenegativna faktorizacija matrice(NMF) predstavlja metodu redukcije dimenzionalnosti koja automatski izvlači značajne atribute [4]. NMF nalazi primene u obradi teksta, obradi slika, a u ovom radu prikazaćemo rezultate primene nad bioinformatičkim podacima. Svojstvo koje matrica mora da zadovolji, kako bi nad njom bila primenjena NMF, jeste da je svaka vrednost te matrice nenegativna.

Neka je matrica $\mathbf{X} \in \mathbb{R}^{p \times n}$ koja čini skup podataka, pri čemu su kolone ove matrice instance, dok vrste atributi. Cilj NMF je dekompozicija matrice \mathbf{X} na dve matrice $\mathbf{W} \in \mathbb{R}^{p \times r}$ i $\mathbf{H} \in \mathbb{R}^{r \times n}$ tako da važi $\mathbf{X} \approx \mathbf{WH}$. Sve tri matrice su nenegativne.

Slika 3: Slika preuzeta sa [2]



Kolone matrice \mathbf{W} možemo interpretirati kao bazne vektore, dok kolone matrice \mathbf{H} predstavljaju koordinate instanci skupa podataka u toj novoj bazi.



Slika 4: Izvor slike [2]

4.3.1 NMF nad tekstualnim podacima

Neka je data nenegativna matrica $\mathbf{X} \in \mathbb{R}^{p \times n}$ takva da svakoj koloni odgovara jedan dokument, dok vrste čine reči. Na poziciji (i, j) te matrice nalazi se broj pojavljivanja i-te reči u j-tom dokumentu. Ovakve matrice nazivamo term-matricama. Primenom NMF nad \mathbf{X} dobijamo dve matrice (\mathbf{W}, \mathbf{H}) tako da važi:

$$\underbrace{X(:, j)}_{j\text{th document}} \approx \sum_{k=1}^r \underbrace{W(:, k)}_{\substack{\text{kth topic} \\ \text{importance of kth topic}}} \underbrace{H(k, j)}_{\substack{\text{in } j\text{th document}}} , \quad \text{with } W \geq 0 \text{ and } H \geq 0.$$

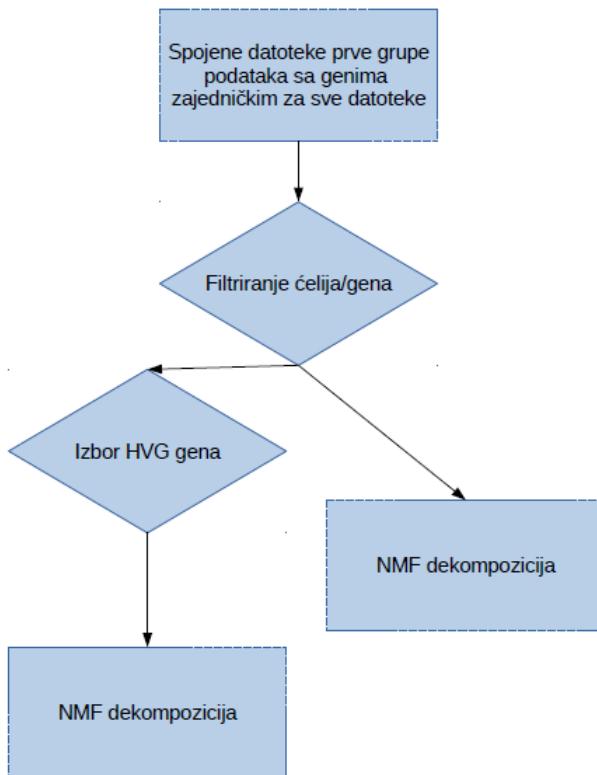
Kolone matrice \mathbf{W} možemo interpretirati kao teme(eng. topics) dok vrednosti u matrici \mathbf{H} čine težine koje govore koliko je neka tema bitna za neki dokument [4].

4.3.2 NMF nad prvom grupom podataka

U sekciji 4.3.1 opisan je rezultat primene NMF nad tekstulanim podacima. Dakle, bazni vektori(kolone matrice \mathbf{W}) čine različite teme(sport, politika...) a težine u matrici \mathbf{H} značajnost određenih tema za neki dokument. U slučaju naših podataka, kolone matrice \mathbf{W} čine *metagene*, dok vrednosti u \mathbf{H} značajnost metagena za pojedinačne ćelije.² Za broj metagena smo izabrali 50, pa dobijamo novu reprezentaciju podataka dimenzija 15165×50 .³

5 Vizuelizacija

U prethodnim sekcijama opisani su načini transformacija skupa podataka kako bi se dobila prikladna forma prihvatljiva od strane algoritama klasterovanja. U daljem radu biće korišćene dve verzije podataka 5. **Prvu verziju** predstavljaju podaci dobijeni izborom zajedničkih gena za sve datoteke, zatim filtriranjem ćelija i gena metodama opisanim u 3.4 i konačno promenom NMF dekompozicije. **Drugu verziju** karakteriše još jedan korak, a to je izbor HVG gena pre primene NMF dekompozicije



Slika 5: Dijagram dobijanja dve verzije podataka

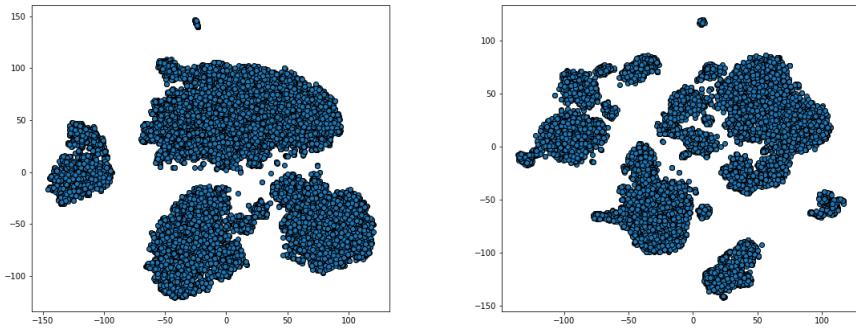
Za vizuelizaciju podataka smo koristili t-SNE algoritam [6]. Kao parametre algoritma, koristili smo:

- $perplexity = 30$
- $n_iters = 5000$

a dobijeni rezultat za obe verzije prve grupe podataka predstavljen je na slici 6. Možemo primetiti da je u drugoj verziji izdvojeno znatno više grupacija tačaka.

²Uzeti u obzir da radimo sa transponovanim podacima, tj. vrste matrice \mathbf{X} su instance(ćelije), a kolone geni. Zbog toga imamo novu reprezentaciju. $\mathbf{X}^T = \mathbf{H}^T \mathbf{W}^T$

³Korišćena je implementacija NMF u okviru sklearn biblioteke [8]



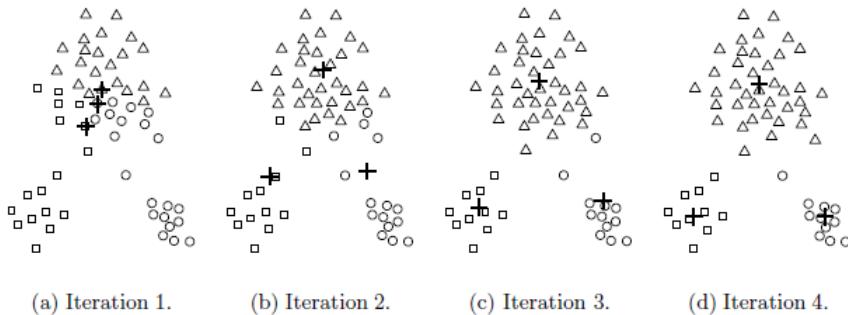
Slika 6: Na slici (levo) prikazana je t-SNE projekcija prve vrezije prve grupe podataka, dok na slici imamo t-SNE projekciju druge verzije

6 Klasterovanje prve grupe podataka

Klasterovanje predstavlja tehniku koja je jako zastupljena u eksplorativnoj analizi podataka. U nastavku biće opisano nekoliko algoritama klasterovanja, kao i njihovi rezultati nad prvom grupom podataka.

6.1 K-sredina

K-sredina predstavlja jedan od najstarijih i najčešće korišćenih algoritama klasterovanja [10]. Osnovni algoritam je jednostavan i iterativne prirode. U svakom koraku bira se K nasumičnih centroida, svaka tačka skupa podataka se pridružuje najbližem centroidu, i na taj način se konstruišu klasteri. Zatim se na osnovu nekog kriterijuma ažuriraju centroidi i postupak se ponavlja sve dok centroidi prestanu da se menjaju.



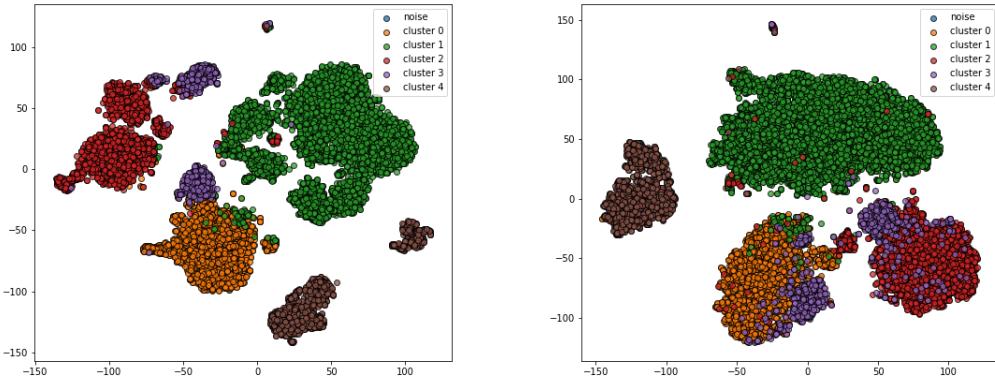
Slika 7: Primer izvršavanja algoritma k-sredina

Izbor centroida zavisi od izbora ciljnje funkcije. Standardna ciljnja funkcija za K-sredina je suma kvadratnih grešaka(eng. Sum of squared errors(SSE)). SSE definišemo kao:

$$SSE = \sum_{i=1}^K \sum_{x_i \in C_i} dist(c_i, x)^2 \quad (1)$$

gde je $dist$ euklidsko rastojanje.

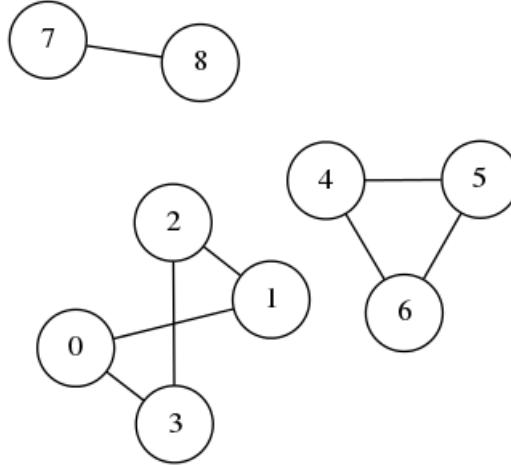
Na slikama 19 i 20 vidimo rezultat algoritma k-sredina za različite vrednosti parametra k nad verzijama, redom, jedan i dva. Konkretno, za pet klastera 8, možemo videti da k-sredina uspeva jasno da razdvaja klastere u slučaju druge verzije, mada prikazivanjem istih klastera nad prvom verzijom podataka, primećujemo da postoji dosta preklapanja.



Slika 8: Klasteri su dobijeni nad drugom verzijom podataka. Slika (levo) prikazuje vizuelizaciju klastera nad t-SNE projekcijom druge verzije, dok slika (desno) vizuelizaciju istih klastera ali nad t-SNE projekcijom prve verzije podataka.

6.2 Spektralno klasterovanje

Ako su podaci predstavljeni kao graf, gde su čvorovi instance, a grane povezuju bliske čvorove(zavisi od zadate mere bliskosti), tada jedan klaster čini komponentu povezanosti tog grafa. Spektralno klasterovanje je primer algoritma zasnovanog na grafovskoj reprezentaciji podataka.



Slika 9: Graf sa tri komponente povezanosti

Neka je $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ neusmeren težinski graf sa skupom čvorova $\mathbf{V} = \{v_1, v_2, \dots, v_n\}$ i skupom grana \mathbf{E} . Matricu povezanosti definisemo kao $\mathbf{W} = (w_{ij})$. Prepostavimo da $w_{ij} \geq 0, \forall i, j$, tj sve težine su nenegativne. U slučaju da je $w_{ij} = 0$, čvorovi i i j nisu povezani.

Stepen čvora $v_i \in V$ definisemo kao

$$\deg(i) = \sum_{j=1}^{|V|} w_{ij}$$

Postoji više načina za računanje matrice povezanosti. U ovom radu koristili smo pristupu k -najbližih suseda. Cilj ove metode je da poveže čvor v_i sa v_j ako je v_j među prvih k najbližih suseda čvora v_i (važi i obrnuto). Ovom metodom dobijamo usmeren graf, pa jednostavnim elemenisanjem usmerenja grana dobijamo neusmerni graf povezanosti.

Glavna komponenta spektralnog klasterovanja je **graf Laplasijan** [12]. *Nenormalizovani graf Laplasijan* definisemo kao

$$L = D - W$$

gde je D dijagonalna matrica takva da $d_{ii} = \deg(i)$ (eng. degree matrix), a matrica W matrica povezanosti.

Osnovni algoritam spektralnog klasterovanja je sledeći [12]:

Input: Skup podataka D , broj klastera c

Izračunati matricu povezanosti metodom k-najbližih suseda;

Izračunati Laplasijana L ;

Izračunati prvih c sopstvenih vektora s_1, s_2, \dots, s_c matrice L . Neka je matrica $S \in \mathbb{R}^{n \times c}$ koja sadrži sopstvene vektore s_1, s_2, \dots, s_c kao kolone. Tada je nova reprezentacija instance $a_i \in D, \forall i$ i-ta vrsta matrice S ;

Izvršiti klasterovanje metodom k-sredina nad novom reprezentacijom podataka;

Output: Klasterovani podaci

Algorithm 1: Spektralno klasterovanje, osnovni algoritam

6.2.1 Spektralno klasterovanje nad prvom verzijom prve grupe podataka

U okviru sekcije 4.3.2 prikazan je NMF oblik podataka prve grupe. Prvu grupu čini pet datoteka, tj datoteke GSM2560245, GSM2560246, GSM2560247, GSM2560248, GSM2560249. Koristili smo implementaciju spektralnog klasterovanja u okviru sklearn biblioteke. Lista bitnih parametara algoritma spektralno klasterovanja i njihove vrednosti su:

- $n_cluster \in \{4, 5, 6, 7\}$
- $affinity \in \{\text{nearest neighbors}, \text{precomputed}\}$
- $n_neighbors \in \{10, 50, 124\}$
- $metrika \in \{\text{euclidsko rastojanje}, \text{kosinusno rastojanje}, \text{korelacija}\}$

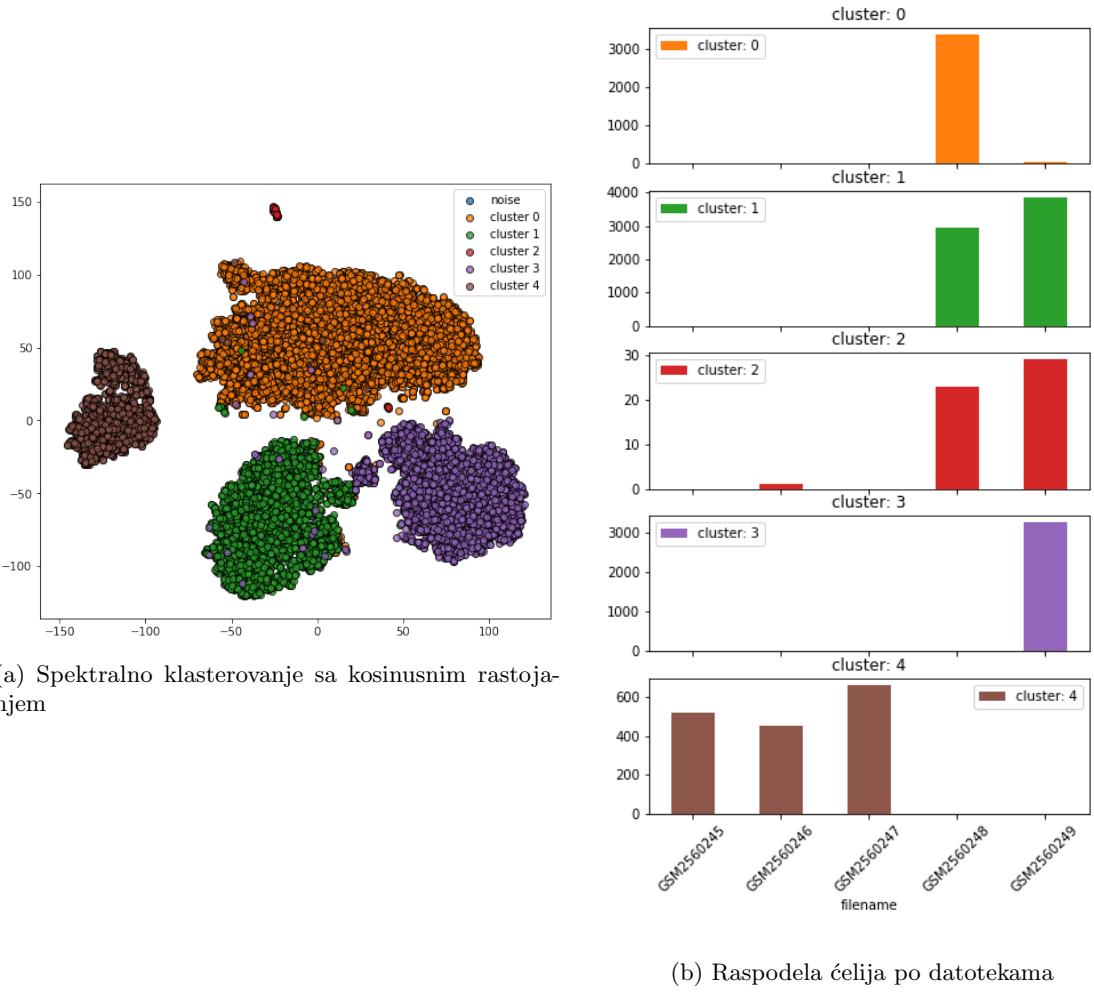
Prilikom konstruisanja matrice povezanosti, za broj suseda $n_neighbors$ korišćene su vrednosti iz skupa $\{10, 50, 124\}$.⁴ Eksperimentisanjem sa različitim metrikama, ispostavlja se da za vrednost $n_neighbors = 10$ dobijamo najbolje rezultate. Na slici 22 možemo videti kako se spektralno klasterovanje ponaša za različit broj klastera. Deluje da se za $n_cluster = 5$ dobijaju najbolji rezultati, pa u nastavku se fokusiramo na taj broj klastera.⁵ Kao mera kvaliteta korišćen je *silueta koeficijent* i t-SNE algoritam za vizuelizaciju.

Na osnovu slika 23, 24 i 25, dobijaju se dosta slični rezultati za sve tri metrike. Jedino euklidsko rastojanje daje manji ukupni silueta koeficijent (**0.204863**) u odnosu na silueta koeficijent dobijen kosinusnim rastojanjem (**0.444556**), odnosno silueta koeficijent dobijen korišćenjem koeficijenta korelacione (**0.438462**) kao metrike rastojanja.

Slike 10a i 10b prikazuju vezu između datoteka i klastera. Tako na primer, klaster 3 čine većinom ćelije iz datoteke GSM2560249 i nekolicina ćelija iz datoteke GSM2560248. Slično važi i za klaster 1, gde je većina ćelija iz jedne datoteke, tj iz GSM2560249. Uzeti u obzir da se skale grafikona razlikuju. Opis datoteka je prikazan u tabeli 4.

⁴Vrednost 124 dobijena je zaokruživanjem vrednosti $\sqrt{\text{broj_instanci}}$

⁵Kako spektralno klasterovanje koristi algoritam k-sredina, različita pokretanja algoritma mogu dati različite rezultate



6.2.2 Spektralno klasterovanje nad drugom verzijom prve grupe podataka

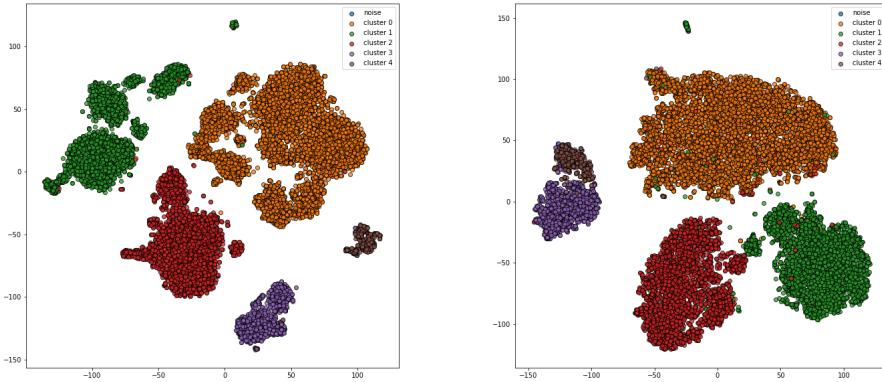
U prethodnoj sekciji videli smo da spektralno klasterovanje, u kombinaciji sa NMF dekompozicijom uspeva da pronađe jasno razdvojene (na osnovu t-SNE prikaza) klastere. Trend se nastavlja i na primeru druge verzije podataka 26. Na slici 11 možemo videti šta se dešava ako dobijene labele klastera vizuelizujemo i nad t-SNE projekcijom za prvu verziju podataka. Interesantno je da su dosta klastera skoro pa identična.

6.3 Hiperarhijsko klasterovanje

Hiperarhijsko klasterovanje predstavlja još jednu metodu klasterovanja čija je osnovna karakteristika, kao što i sam naziv govori, konstruisanje *hijerarhija*. Postoje dve vrste hiperarhijskog klasterovanja, i to:

- sakupljujući (eng. agglomerative)
- razdvajajući (eng. divisive)

Osnovna razlika ove dve vrste je to što, u sakupljujućem slučaju, svaka instanca predstavlja odvojen klastar. Iterativno se zatim, na osnovu neke mere bliskosti, spajaju najbliži klastari sve dok ne ostane samo jedan klastar. U slučaju razdvajajućeg hiperarhijskog algoritma, proces je analogan, pri čemu se počinje od jednog klastera, a zatim se instance iterativno razdvajaju. U nastavku biće više reči o sakupljujućem hiperarhijskom algoritmu, kao i njegovim varijacima. Opšti algoritam je oblika:



Slika 11: Spektralno klasterovanje dobijeno nad drugom verzijom podataka, prikaz nad t-SNE projekcijom druge verzije (levo) i prikaz nad t-SNE projekcijom prve verzije (desno).

Input: skup podataka D ili matrica bliskosti P

Ako je prosleđen skup podataka, izračunati matricu bliskosti P ;

repeat

Spoji dva "najbliža" klastera;
Transformiši matricu bliskosti P ;

until dok nije ostao samo jedan klaster;

Output: Klasterovani podaci

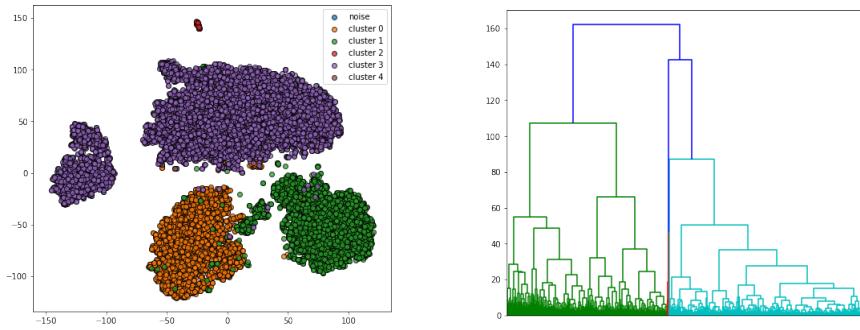
Algorithm 2: Osnovni sakupljajući hijerarhijski algoritam

Definicija mere bliskosti je ono što razlikuje različite vrste sakupljajućeg hijerarhijskog algoritma [10]. Standardne mere bliskosti su:

- Single vezu
- Complete vezu
- Average vezu
- Ward-ova vezu

Upoređene su sve četiri mere bliskosti nad prvom grupom podataka, pri čemu su najbolji rezultati postignuti sa Ward-ovom merom. Na slici 12b prikazan je dendrogram dobijen korišćenjem biblioteke SciPy [11], dok na slici 12a prikazan je rezultat dobijen funkcijom AgglomerativeClustering u okviru scikit-learn biblioteke i to sa parametrima:

- $n_clusters = 5$
- $linkage = "ward"$

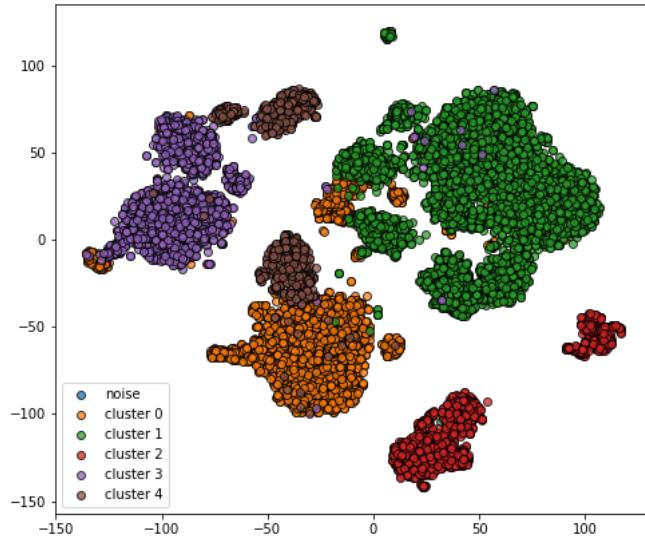


(a) Higerarhijsko klasterovanje sa Ward-ovom merom za 5 klastera

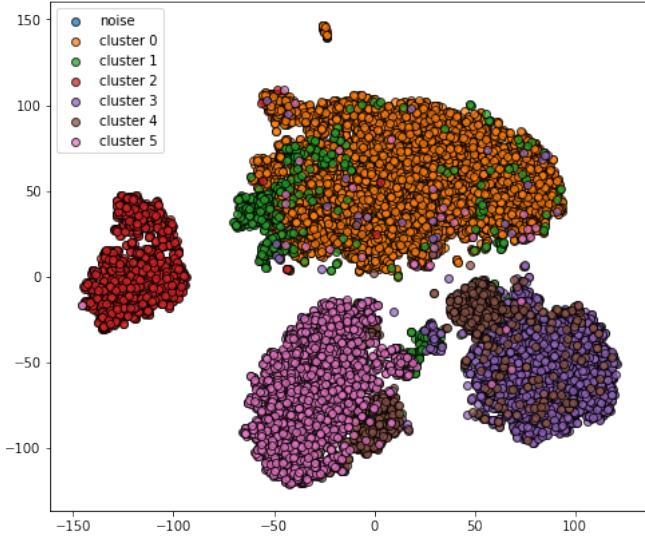
(b) Dendogram Ward-ovog higerarhijskog algoritma

Slika 12: Na slici (a) prikazani su dobijeni klasteri higerarhijskim algoritmom sa Ward-ovom mera, dok na slici (b) ceo dendrogram higerarhijskog algoritma, takođe sa Ward-ovom mera, u obzir da su slike nezavisne jedna od druge.

Za drugu verziju podataka, Ward-ova mera daje dosta slične, pa gotovo iste rezultate kao i algoritam k-sredina 13. Takođe, preklapanja koja su se javljala u slučaju k-sredina algoritma, javljaju se i ovde 14, pa se nameće pitanje, koja je od ove dve verzije podataka reprezentativnija.



Slika 13: Klasteri dobijeni Ward-ovim higerarhijskim klasterovanjem nad verzijom dva



Slika 14: Klasteri dobijeni Ward-ovim hijerarhijskim klasterovanjem nad verzijom dva, prikazani nad verzijom jedan.

6.4 Klasterovanje zasnovano na gustini

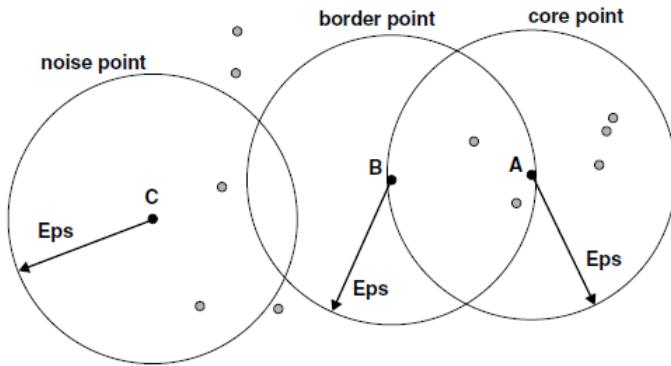
Algoritmi zasnovani na gustini lociraju "guste" regije prostora razdvojene regionima manje gustine [10]. Osnovna karakteristika ovih algoritama je to što mogu da prepoznaju klastere proizvoljnog oblika. Najpoznatiji algoritam koji se zasniva na ovom principu je DBSCAN [3].

6.4.1 DBSCAN

DBSCAN se pokazao kao izuzetno dobar algoritam za pronalaženje klastera različitih oblika. Algoritam se sastoji od dva parametra, $minPts$ i eps . Takođe, autori DBSCAN-a su predložili metodu k -dist plot-a kako bi se broj parametara algoritma sveo na svega jedan. Drugim rečima za dati $minPts$, pomoću k -dist plot-a, moguće je automatski odrediti prikladno eps . Osnovna karakteristika DBSCAN-a je podela tačaka(instanci) na 3 vrste, i to na tačke jezgra, granične tačke i šum [15].

Da bismo definisali tačke jezgra, uvodimo pojam *eps-okruženje*. Eps-okruženje tačke p predstavlja skup tačaka $N_{eps}(p)$ definisan kao

$$N_{eps}(p) = \{q \mid dist(p, q) \leq eps\} \quad (2)$$



Slika 15: Šum, granične tačke i tačke jezgra

Ako je za tačku p poznata njeno eps -okruženje i ako važi da

$$N_{eps}(p) \geq minPts \quad (3)$$

tada kažemo da je tačka p *tačka jezgra*. Za tačku q koja nije tačka jezgra ali važi da u njenom eps -okruženju pripada neka tačka jezgra, onda za q kažemo da je *granična tačka*. I na kraju, ako tačka x nije ni tačka jezgra, ni granična, onda nju smatramo kao *šum*. Kada se odrede tipovi tačaka, susedne tačke jezgra⁶ se spajaju u jedan klaster, a granične tačke se pridružuju njima najbližim klastерима.

6.4.2 Određivanje parametara eps i minPts

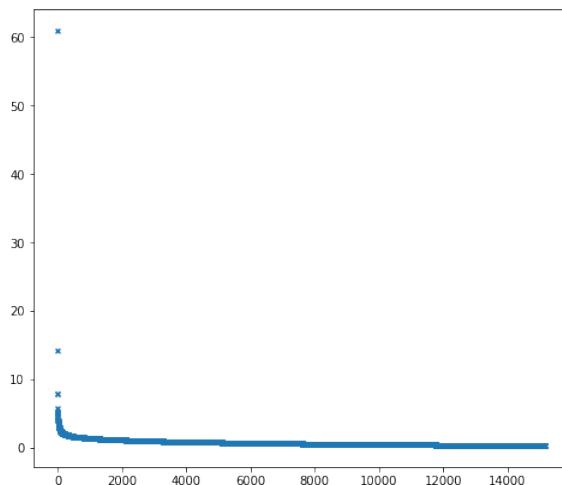
U sekciji 6.4.1 je spomenuto da DBSCAN čine dva parametra, i to minPts i eps . Zajedno, ova dva parametra određuju koliko *retke* klastere očekujemo. Tako za manje vrednosti eps i veće vrednosti minPts , dobijamo guste klastere, dok za veće vrednosti eps i manje vrednosti minPts dobijamo *retke* klastere. Odabir parametara ima ključnu ulogu na performanse DBSCAN-a. Takođe često se ne zna "gustina" klastera, što znatno otežava problem. Zbog ovoga, autori DBSCAN-a su predložili heuristiku k-dist plot⁷ za određivanje prikladnog eps -a na osnovu zadate vrednosti za minPts . Procedura je sledeća, za svaku tačku se nalazi njen minPts -najbliži sused. Zatim se dobijeni niz minPts -najbližih suseda sortira, i traži se vrednost u kojoj dobijena "kriva" naglo menja pravac. Na slici 16 prikazan je k-dist plot za prvu verziju prve grupe podataka za $\text{minPts} = 10$.

6.4.3 DBSCAN nad prvom grupom podataka

Na osnovu slike 16, odabrali smo da vrednosti eps budu iz intervala $[3, 8]$. Interval $[3, 8]$ smo ekvidistantno podelili na 5 delova i dobijene vrednosti smo koristili za vrednost eps -a, dok je vrednost minPts postavljena na 10. Sa ovim parametrima, primenili smo DBSCAN nad NMF reprezentacijom prve grupe podata. Ipak, DBSCAN nije uspeo da detektuje klastere, čak nije prepoznao šumove, pa je sve tačke stavio u jedan klaster. Slični rezultati se dobijaju ako DBSCAN primenimo nad formatu prve grupe opisanom u 3.4.

Takođe, primenili smo i algoritam OPTICS [1], ali rezultati su takođe bili nezadovoljavajući, pa neće biti detaljnije diskusije o navedenom algoritmu.

Na osnovu slike 16 može se zaključiti da su sve tačke relativno "blizu" jedna drugoj. To možemo takođe videti iz vrednosti silueta koeficijnata dobijenih spektralnim klasterovanjem. Naime, kada je silueta koeficijent blizak nuli, to ukazuje na preklapanje klastera. To možemo videti i u slučaju spektralnog klasterovanja, konkretno za spektralno klasterovanje i euklidsko rastojanje 23a. Interesantno je da je ipak, na osnovu vizuelizacije t-SNE-om, spektralno klasterovanje uspelo da razdvoji klastere.



Slika 16: K-dist plot za NMF reprezentaciju prve grupe podataka, za $\text{minPts} = 10$

Slične rezultate dobijamo i sa drugom verzijom podataka, pri čemu u ovom slučaju neko licina tačaka je označena kao šumovi, dok većina pripada jednom klasteru. Za parametre smo koristili

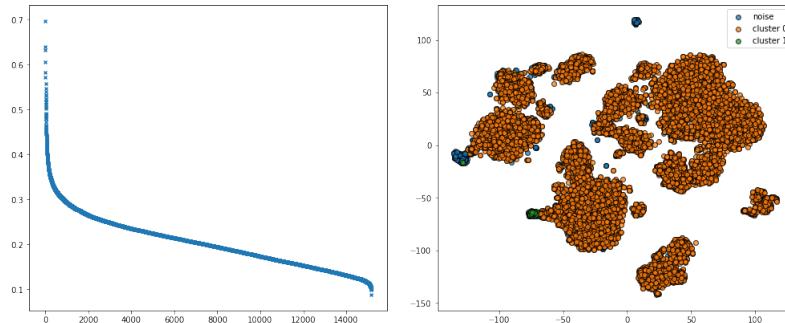
⁶Tačke p i q su susedne ako važi da $p \in N_{\text{eps}}(q)$ ili $q \in N_{\text{eps}}(p)$

⁷Vrednost k u nazivu k-dist je jednaka minPts .

- $\text{eps} = 0.3$
- $\text{minPts} = 50$

a vizuelni prikaz je predstavljen na slici 17.

Ovim dolazimo do zaključka da algoritmi zasnovani na gustini nisu primenljivi nad ovakvom vrstom podataka.



Slika 17: Na slici (levo) vidimo k-dist plot za drugu verziju prve grupe podataka, dok slika (desno) prikazuje t-SNE vizuelizaciju dobijenih klastera DBSCAN algoritmom

6.5 BIRCH

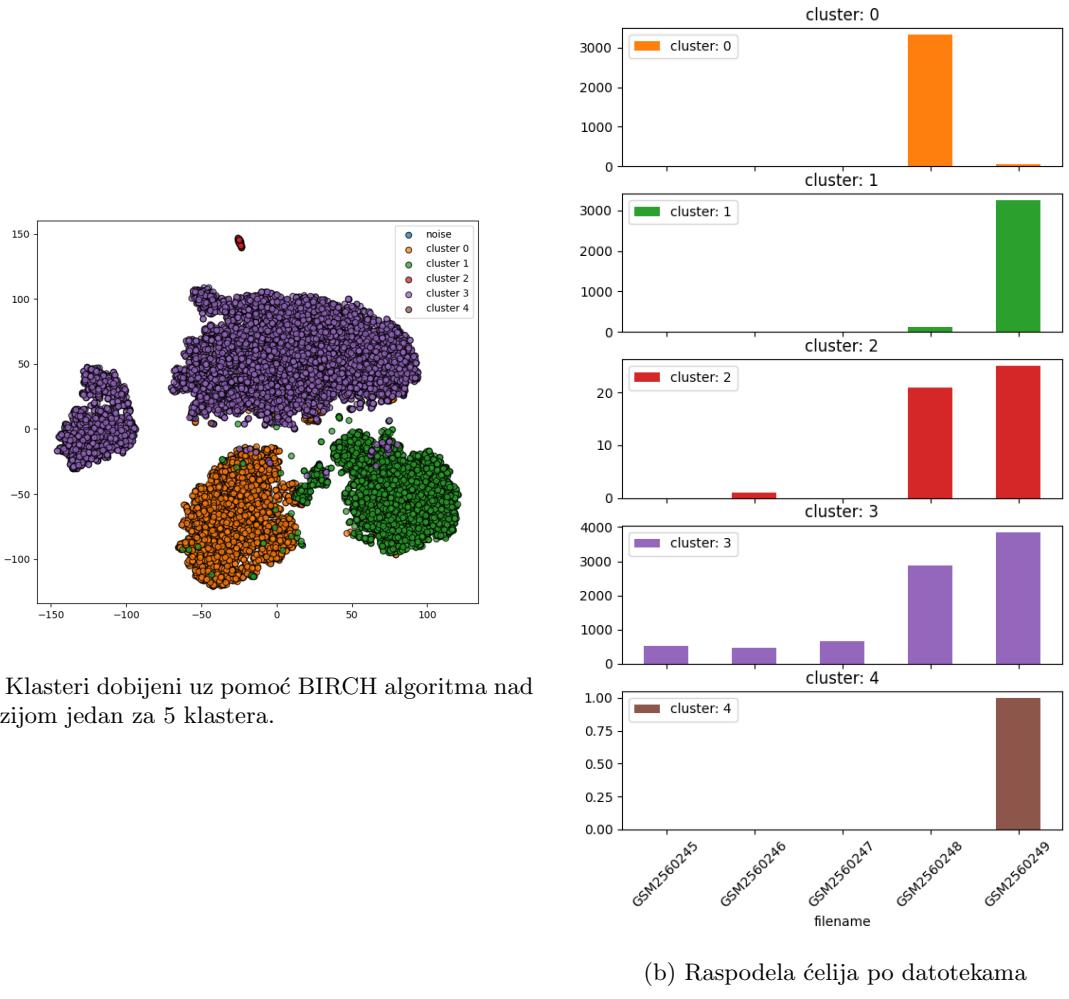
BIRCH algoritam se zasniva na CF drvetu. CF drvo je balansirano drvo koje sadrži faktor grananja B i prag T kao svoje parametre [7]. Unutrašnji čvorovi drveta sadrže niz od najviše B elemenata. Unutrašnji čvor predstavlja klaster koji sadrži sve potklasterne predstavljene njegovim elementima. Listovi sadrže pokazivače kako bi svi oni bili povezani radi efikasnijeg prolaska kroz njih. Listovi su klasteri koji se sastoje od svih potklastera predstavljenih njihovim elementima. Prečnik mora biti manji od T , je uslov koji moraju zadovoljiti svi elementi u listu.

Ovaj algoritam se sastoји од 4 faze, od kojih su neke opcione. U fazi 1 koja je obavezna prolazi se kroz sve podatke i cilj je u memoriji konstruisati početno CF drvo. U ovoj fazi se konstruiše pregled podataka u memoriji tako da su gusto grupisane tačke podataka potklasteri dok se sve druge razbacane tačke uklanjanju kao greške. Prva faza obezbeđuje da će izračunavanja u sledećim fazama biti brža jer nisu potrebne nikakve ulazne operacije kao i to da je problem klasterovanja originalnih podataka smanjen na manji problem klasterovanja potklastera u elementima listova. Takođe, kasnija izračunavanja će biti preciznija jer je većina grešaka uklonjena.

Faza 2 je opcionalna i u njoj se prolazi kroz elemente lista u inicijalnom CF drvetu, tako da se potklasteri koji su gusti grupišu u veće potklasterne. U ovoj fazi dolazi i do dodatnog uklanjanja grešaka.

U fazi 3 se koristi algoritam koji klasteruje elemente lista tako da se onemogući loše klasterovanje koje nastaje zbog lošeg redosleda ulaza i deljenjem zbog veličine stranice u memoriji.

Faza 4 je opcionalna. Klasteri koji su dobijeni u fazi 3 su bitni jer se u fazi 4 koriste njihovi centroidi kao semena za preraspodelu tačaka podataka prema najbližem semenu i tako nastaju novi skupovi klastera. U ovoj fazi je moguće odstraniti tačke koje predstavljaju greške.



(a) Klasteri dobijeni uz pomoć BIRCH algoritma nad verzijom jedan za 5 klastera.

(b) Raspodela célja po datotekama

Slika 18: Na slici (a) prikazani su klasteri dobijeni BIRCH algoritmom sa parametrima $n_{\text{cluster}} = 5$, branching factor = 30 i threshold = 0.2, dok na slici (b) raspored tako klasterovanih célja po datotekama.

7 Pregled dobijenih rezultata

S obzirom da su podaci nelabelirani, najviše smo se oslanjali na vizuelizaciju pomoću t-SNE kao na meru kvaliteta. Takođe, izračunali smo i silueta koeficijente za sve primenjene algoritme 2 i 3⁸. Interesantno je da nijedan algoritam nema silueta koeficijent veći od 0.5, ali isto tako nema algoritama koji imaju negativni silueta koeficijent, te je većina donekle uspešna da prepozna neku strukturu u podacima. Za računanje silueta koeficijenta koristili smo dve metrike, i to euklidsko i kosinusno rastojanje. Ubedljivo najbolje rezultate dobijamo sa spektralnim klasterovanjem u kombinaciji sa kosinusnim rastojanjem ($\text{sil_koe} = 0.445$).

Za svaki od navedenih algoritama u tabelama 2 i 3 izračunate su i raspodele datoteka po klasterima, kao na primeru 10b i svi podaci se mogu pronaći u direktorijumu *supplementary_data*.

⁸DBSCAN algoritam nije uzet u obzir jer nije uspeo da prepozna klaster

Tabela 2: Silueta koeficijent nad prvom verzijom podataka

Algoritam	Broj Klastera	Silueta koef / Euklidsko	Silueta koef / Kosinusno
Spektralno klast. / euklidsko rast.	5	0.204863	0.439041
Spektralno klast. / kosinusno rast.	5	0.199455	0.444556
K-sredina	4	0.396459	0.347671
K-sredina	5	0.397532	0.346946
K-sredina	6	0.310075	0.219312
K-sredina	7	0.153298	0.184236
Ward-ovo hijerarhijsko	4	0.339216	0.382940
Ward-ovo hijerarhijsko	5	0.339920	0.382643
Ward-ovo hijerarhijsko	6	0.261799	0.228352
BIRCH / branching factor=30	4	0.347014	0.386919
BIRCH / branching factor=30	5	0.347720	0.386620
BIRCH / branching factor=30	6	0.266185	0.240132

Tabela 3: Silueta koeficijent nad drugom verzijom podataka

Algoritam	Broj Klastera	Silueta koef / Euklidsko	Silueta koef / Kosinusno
Spektralno klast. / euklidsko	4	0.176517	0.279486
Spektralno klast. / euklidsko	5	0.217971	0.297713
Spektralno klast. / euklidsko	6	0.224402	0.300279
Spektralno klast. / euklidsko	7	0.238307	0.309415
Spektralno klast. / euklidsko	8	0.241764	0.292899
Spektralno klast. / euklidsko	9	0.204278	0.297957
K-sredina	4	0.239236	0.314944
K-sredina	5	0.248646	0.321342
K-sredina	6	0.248669	0.314202
K-sredina	7	0.252884	0.305646
K-sredina	8	0.225043	0.326530
K-sredina	9	0.215614	0.297955
Ward-ovo hijerarhijsko	4	0.216909	0.282551
Ward-ovo hijerarhijsko	5	0.239302	0.299316
Ward-ovo hijerarhijsko	6	0.239470	0.306849
BIRCH / branching factor=30	5	0.226926	0.308484
BIRCH / branching factor=30	6	0.233039	0.312238

8 Zaključak

U ovom radu primenili smo više algoritama za klasterovanje nad PBMC ćelijama. Zbog prirode podataka, velika pažnja je posvećena redukciji dimenzionalnosti. Koristili smo dve verzije reprezentacije podataka, pri čemu dobijamo dobre rezultate za obe verzije. Algoritam spektralnog klasterovanja se pokazao kao kvalitetan pristup ovom problemu, dok sa algoritmima zasnovanim na gustini nismo imali dosta uspeha. Za dalji rad planiramo upotrebu još nekoliko metoda za preprocesiranje podataka, poput tf-idf-a, kao i primeniti druge pristupe za redukciju dimenzionalnosti, poput autoenkodera.

A Opisi datoteka koje čine skup podataka

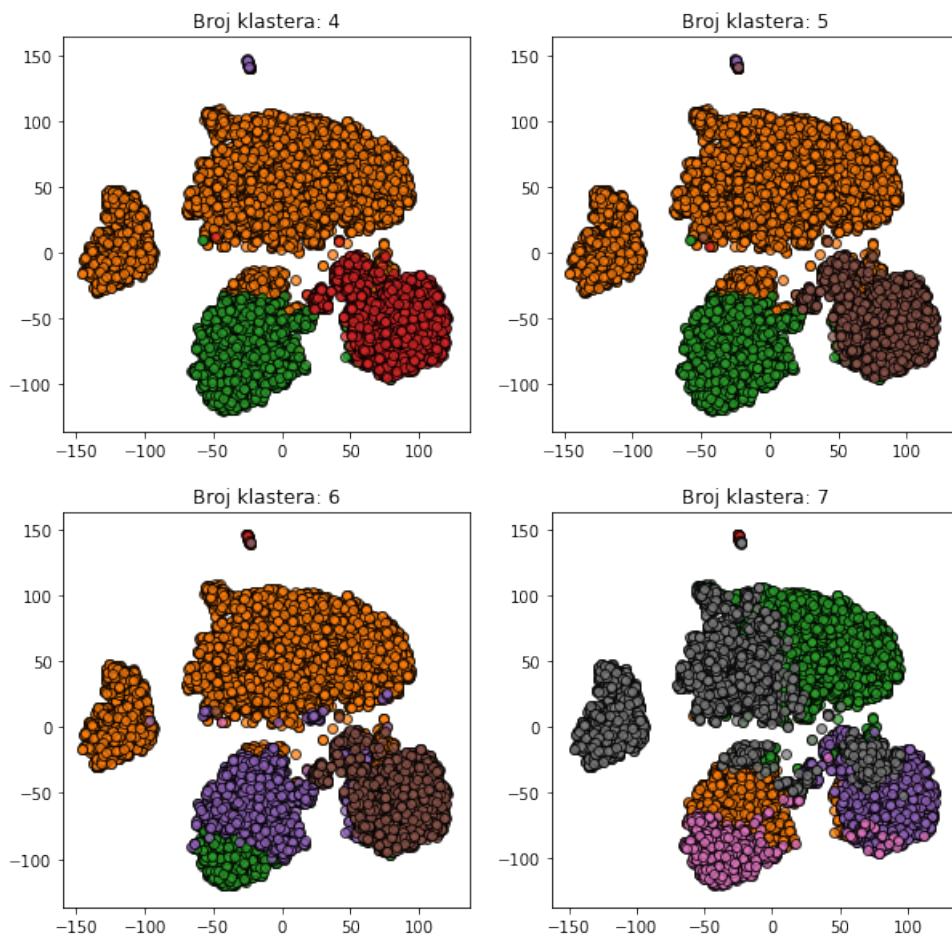
Tabela 4: Metapodaci za prvu grupu

SAMPLE	GENOME	DESCRIPTION
GSM2560245	hg19	batch 1 sample A; single cell RNA-seq_SLE patients (well 1); Homo sapiens; subject status: SLE patient; cell type: peripheral blood mononuclear cells (PBMCs);
GSM2560246	hg19	batch 1 sample B; single cell RNA-seq_SLE patients (well 2); Homo sapiens; subject status: SLE patient; cell type: peripheral blood mononuclear cells (PBMCs);
GSM2560247	hg19	batch 1 sample C; single cell RNA-seq_SLE patients (well 3); Homo sapiens; subject status: SLE patient; cell type: peripheral blood mononuclear cells (PBMCs);
GSM2560248	hg19	batch 2 control; single cell RNA-seq_SLE patients (6 hours control); Homo sapiens; subject status: SLE patient; cell type: peripheral blood mononuclear cells (PBMCs); stimulated with: none (control)
GSM2560249	hg19	batch 2 stim (IFN-beta); single cell RNA-seq_SLE patients (6 hours IFN-b stimulation); Homo sapiens; subject status: SLE patient; cell type: peripheral blood mononuclear cells (PBMCs); stimulated with: IFN-beta for 6hrs

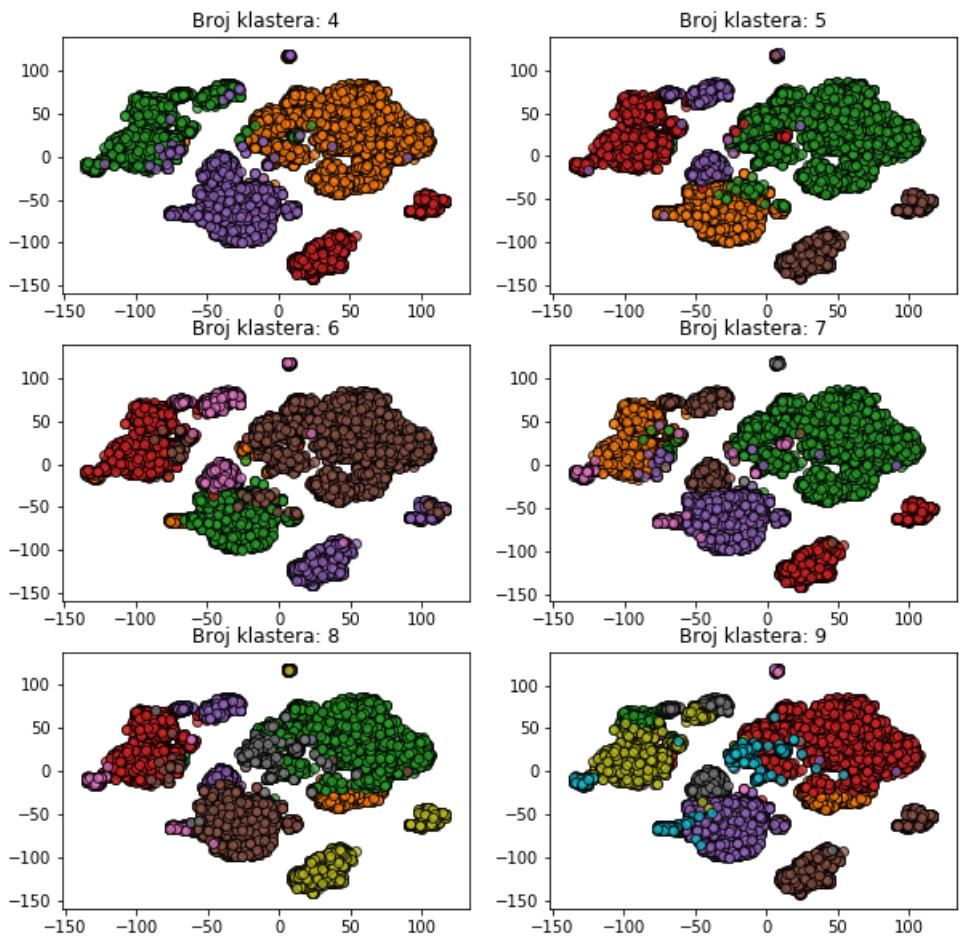
Tabela 5: Metapodaci za drugu grupu

SAMPLE	GENOME	DESCRIPTION
GSM3087619	GRCh38	DTM-X_PBMC_live; whole blood; Homo sapiens; isolation: Ficoll; fixation: Live; resuspension: PBS; cell type: PBMC
GSM3478792	GRCh38	Patient1_Nonmalignant; PBMC; Homo sapiens; condition: Normal PBMC T Cells
GSM3892571	GRCh38	PBMC_HV; PBMC from a sex- and age-matched healthy volunteer (HV); Homo sapiens; disease diagnosis: Healthy; tissue: PBMC; manipulation: Freshly isolated
GSM3169075	GRCh38 version 90	Healthy human PBMCs; PBMC_scRNA-seq; Homo sapiens; subject status: healthy donor; cell type: Peripheral blood mononuclear cells (PBMCs); barcode coordinate: tag CB; umi coordinate: tag UB
GSM3374613	hg38	Ina_cell_pbmc_original; Human PBMCs; Homo sapiens; sample type: Human PBMCs
GSM3374614	hg38	Ina_cell_pbmc_resampled; Human PBMCs; Homo sapiens; sample type: Human PBMCs

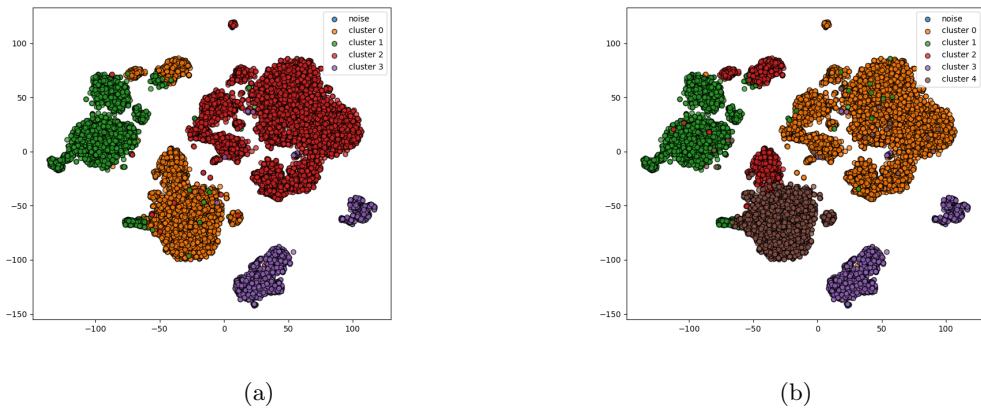
B Dodatne vizuelizacije klastera



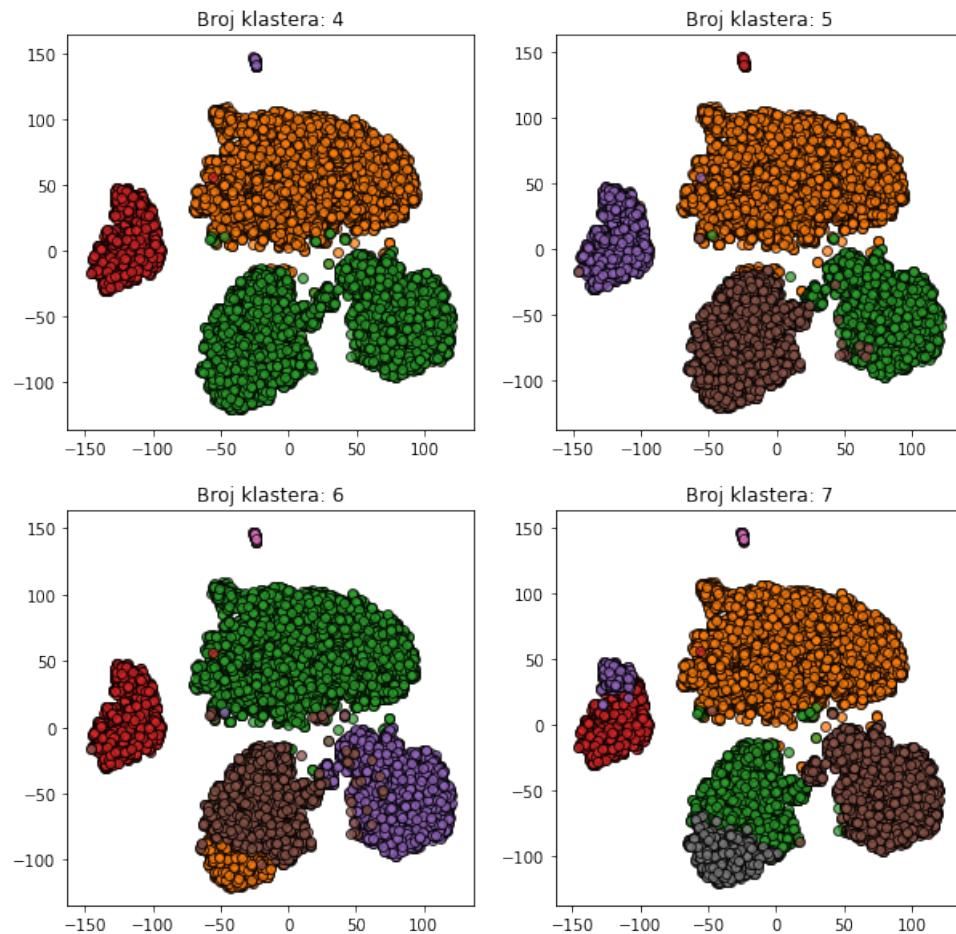
Slika 19: K-sredina nad prvom verzijom prve grupe podataka



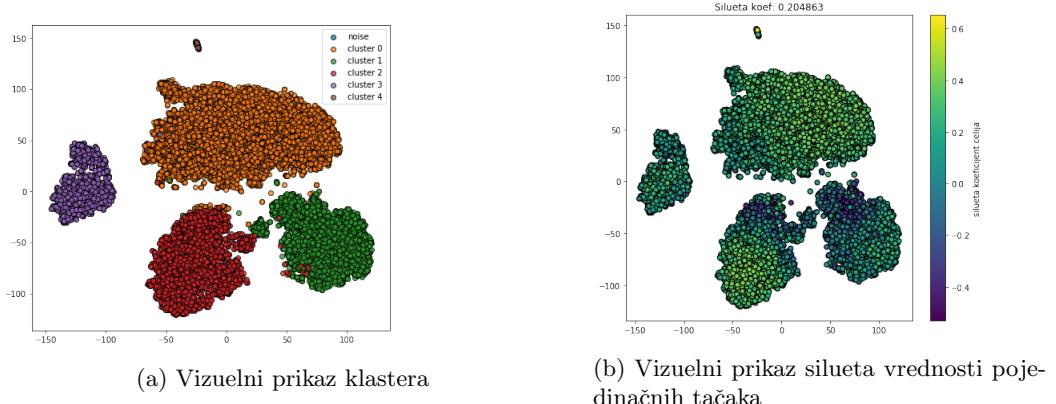
Slika 20: K-sredina nad drugom verzijom druge grupe podataka



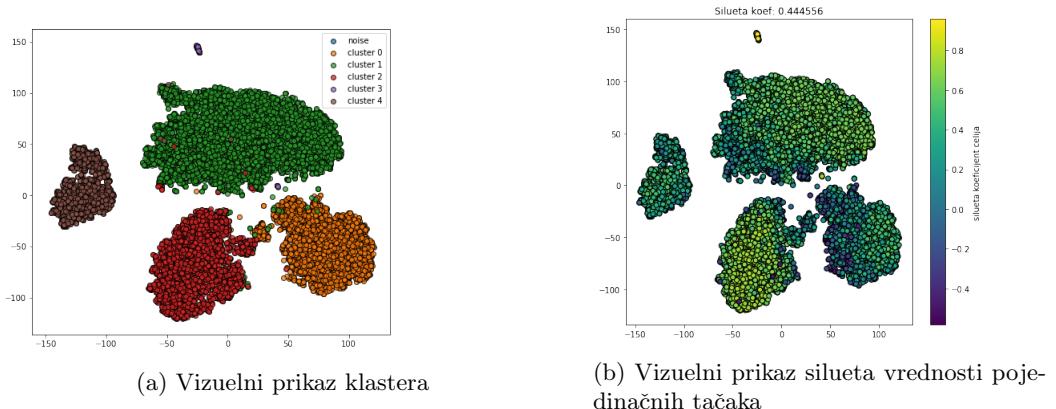
Slika 21: BIRCH algoritam pirmenjen nad drugom verzijom prve grupe podataka za (a) 4 i (b) 5 klatera



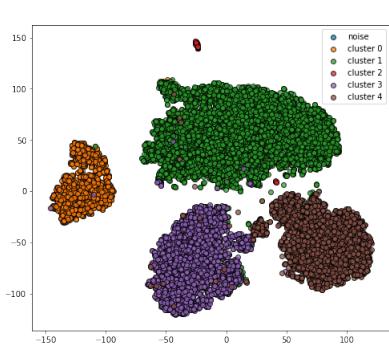
Slika 22: Spektralno klasterovanja nad prvom verzijom prve grupe podataka za različite vrednosti broja klastera



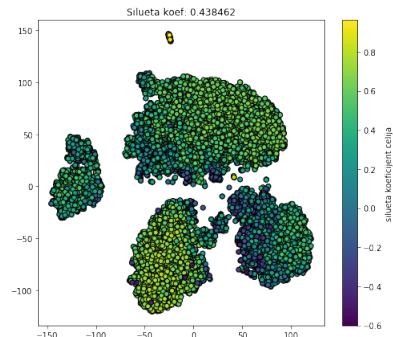
Slika 23: Spektralno klasterovanje sa *euklidskim* rastojanjem. Na slici (a) klasteri su prikazani različitim bojama, dok na slici (b) boja jedne tačke zavisi od njenog silueta koeficijenta. Svetlige boje ukazuju na veću vrednost silueta koeficijenta, dok tamnije manju. Ukupan silueta koeficijent je **0.204863**



Slika 24: Spektralno klasterovanje sa *kosinusnim* rastojanjem. Na slici (a) klasteri su prikazani različitim bojama, dok na slici (b) boja jedne tačke zavisi od njenog silueta koeficijenta. Svetlige boje ukazuju na veću vrednost silueta koeficijenta, dok tamnije manju. Ukupan silueta koeficijent je **0.444556**

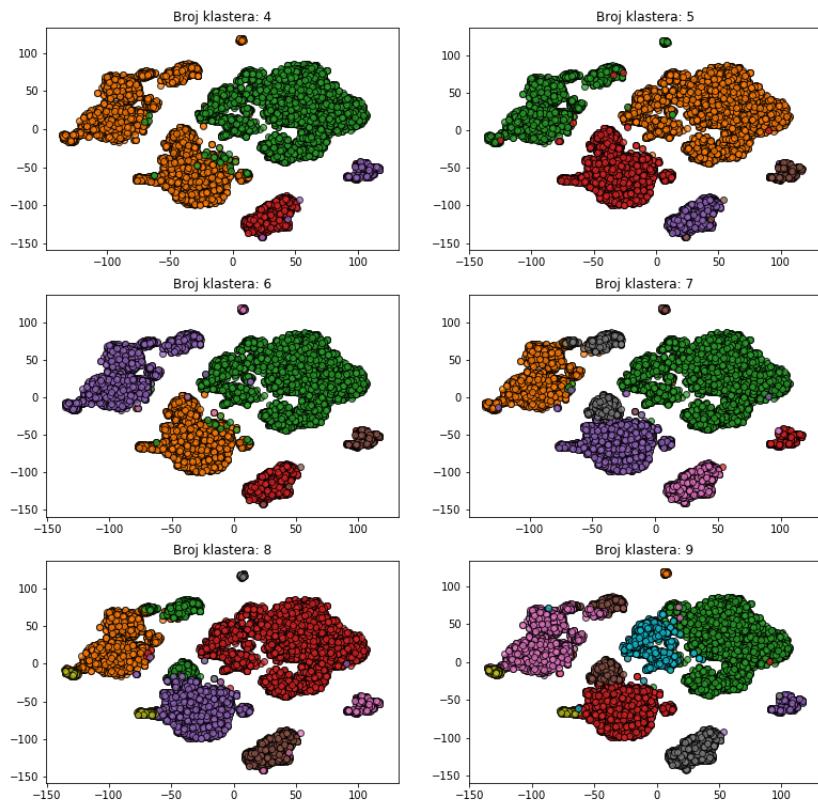


(a) Vizuelni prikaz klastera



(b) Vizuelni prikaz silueta vrednosti pojedinačnih tačaka

Slika 25: Spektralno klasterovanje sa *koefficijentom korelacije* kao mera rastojanja. Na slici (a) klasteri su prikazani različitim bojama, dok na slici (b) boja jedne tačke zavisi od njenog silueta koeficijenta. Svetlige boje ukazuju na veću vrednost silueta koeficijenta, dok tamnije manju. Ukupan silueta koeficijent je **0.438462**



Slika 26: Spektralno klasterovanja nad drugom verzijom prve grupe podataka za različit broj klastera

Literatura

- [1] Mihael Ankerst, Markus M Breunig, Hans-Peter Kriegel, and Jörg Sander. Optics: ordering points to identify the clustering structure. *ACM Sigmod record*, 28(2):49–60, 1999.
- [2] Lamacraft Austen. The why and how of nonnegative matrix factorization. <https://blog.acolyer.org/2019/02/18/the-why-and-how-of-nonnegative-matrix-factorization/>.
- [3] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. pages 226–231. AAAI Press, 1996.
- [4] Nicolas Gillis. The why and how of nonnegative matrix factorization. *Regularization, optimization, kernels, and support vector machines*, 12(257):257–291, 2014.
- [5] Malte D Luecken and Fabian J Theis. Current best practices in single-cell rna-seq analysis: a tutorial. *Molecular systems biology*, 15(6), 2019.
- [6] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.
- [7] Maljković Mirjana. Skalabilni klaster algoritmi. <http://alas.matf.bg.ac.rs/~mi05006/ip/seminarskiSkalabilniKlasterAlgoritmi.pdf>.
- [8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [9] Razin Abdulrauf Shaikh, Jiahui Zhong, Minjie Lyu, Sen Lin, Derin Keskin, Guanglan Zhang, Lou Chitkushev, and Vladimir Brusic. Classification of five cell types from pbmc samples using single cell transcriptomics and artificial neural networks. In *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 2207–2213. IEEE, 2019.
- [10] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to data mining*. Pearson Education India, 2016.
- [11] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, CJ Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake Vand erPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.
- [12] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.