Avila skup podataka

Lazar Dačkovič

30. maj 2023.

Sadržaj

1 Uvod

1.1 Upoznavanje sa skupom podataka

Avila skup podataka je izvučen iz 800 slika "Avila biblije", velike latinske kopije iz 12. veka. Cilj Avila skupa podataka je klasifikacija pisama na španskom jeziku, na osnovu određenih karakteristika pisama. Konkretno, skup podataka se sastoji od 20867 instanci pisama koje su prethodno obradili algoritmi za prepoznavanje teksta. Svaka instanca ima 10 numeričkih atributa koji opisuju karakteristike teksta. Cilj klasifikacije je da se na osnovu tih atributa odredi kojoj od 12 kategorija pisma (klasa) data instanca pripada. Ovaj skup podataka se često koristi za testiranje algoritama klasifikacije.

Podaci su normalizovani korišćenjem metode Z-normalizacije. Podeljeni su u dva skupa podataka: Trening skup podataka koji sadrži 10430 uzoraka, i testa koji sadrzi 10437 uzoraka

1.2 Informacije o atributima: 10 atributa i klasa

F1: Međustubno rastojanje
F2: Gornja margina
F3: Donja margina
F4: Eksploatacija
F5: Broj reda
F6: Modularni odnos
F7: Međuredni razmak
F8: Težina
F9: Broj vrha
F10: F6/F7
Klasa: A, B, C, D, E, F, G, H, I, W, X, Y

2 Eksplorativna analiza podataka

U ovom delu bavićemo se analizom podataka, pretprocesiranjem i pronalaženju nekih bitnih osobina, pre nego što pređemo na konstruisanje modela klasifikacije.

2.1 Prikaz podataka i da li ima nedostajućih vrednosti

Na slici 1 možemo videti primere instanci klase, sa njihovim vrednostima atributa.

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	Copyist
0	0.130292	0.870736	-3.210528	0.062493	0.261718	1.436060	1.465940	0.636203	0.282354	0.515587	A
1	-0.116585	0.069915	0.068476	-0.783147	0.261718	0.439463	-0.081827	-0.888236	-0.123005	0.582939	A
2	0.031541	0.297600	-3.210528	-0.583590	-0.721442	-0.307984	0.710932	1.051693	0.594169	-0.533994	A
3	0.229043	0.807926	-0.052442	0.082634	0.261718	0.148790	0.635431	0.051062	0.032902	-0.086652	F
4	0.117948	-0.220579	-3.210528	-1.623238	0.261718	-0.349509	0.257927	-0.385979	-0.247731	-0.331310	A
...	...	...	...	...	...	...	...	...	...	...	...
10424	0.080916	0.588093	0.015130	0.002250	0.261718	-0.557133	0.371178	0.932346	0.282354	-0.580141	F
10425	0.253730	-0.338346	0.352988	-1.154243	0.172340	-0.557133	0.257927	0.348428	0.032902	-0.527134	F
10426	0.229043	-0.000745	0.171611	-0.002793	0.261718	0.688613	0.295677	-1.088486	-0.590727	0.580142	A
10427	-0.301743	0.352558	0.288973	1.638181	0.261718	0.688613	0.069175	0.502761	0.625350	0.718969	E
10428	-0.104241	-1.037102	0.388552	-1.099311	0.172340	-0.307984	0.786433	-1.337547	0.999528	-0.551063	X

Slika 1: Primeri nekih instanci sa vrednostima atributa

Na slici 2 možemo videti neke opšte statistike i to da ne postoje nedostajuće vrednosti.

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
count	10429.000000	10429.000000	10429.000000	10429.000000	10429.000000	10429.000000	10429.000000	10429.000000	10429.000000	10429.000000
mean	0.000827	0.033630	-0.000556	-0.002433	0.006354	0.013948	0.005570	0.010234	0.012891	0.000803
std	0.991475	3.921056	1.120251	1.008564	0.992100	1.126296	1.313812	1.003515	1.087715	1.007141
min	-3.498799	-2.426761	-3.210528	-5.440122	-4.922215	-7.450257	-11.935457	-4.247781	-5.486218	-6.719324
25%	-0.128929	-0.259834	0.064919	-0.528002	0.172340	-0.598658	-0.044076	-0.542001	-0.372457	-0.516103
50%	0.043885	-0.055704	0.217845	0.095763	0.261718	-0.058835	0.220177	0.111754	0.064084	-0.034621
75%	0.204355	0.203385	0.352988	0.658210	0.261718	0.564038	0.446679	0.654900	0.500624	0.530885
max	11.819916	386.000000	50.000000	3.987152	1.066121	53.000000	83.000000	13.173081	44.000000	4.671232

Ne postoje nedostajuce vrednosti u trening skupu podataka, ni u test skupu

```
df_trening.isna().any().any()
```

```
False
```

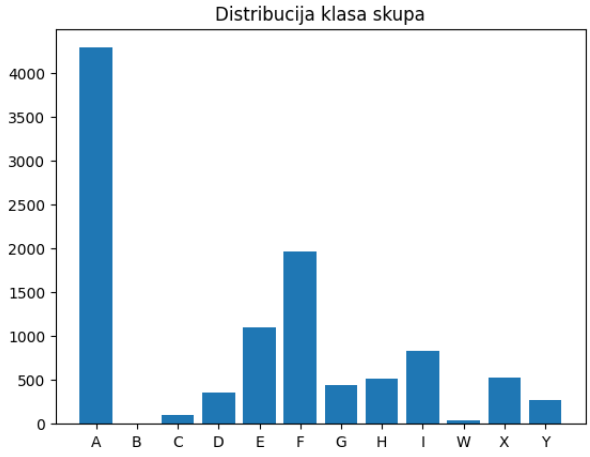
```
df_test.isna().any().any()
```

```
False
```

Slika 2: Opšte statistike skupa podataka

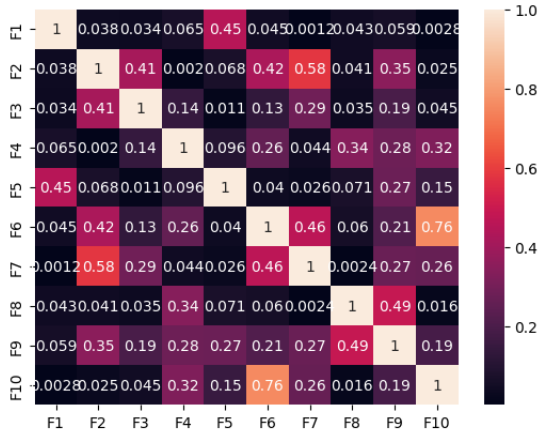
2.2 Distribucija klasa i matrica korelacije

Na slici 3 je prikazana distribucija klasa u trening skupu, potpuno slično je i na test skupu. Ono što možemo zaključiti da je broj klasa neravnomeran. Najdominantnija klasa je klasa A



Slika 3: Distribucija klasa u trening skupu

Na slici 4 je prikazana matrica korelacije. Iz nje možemo videti da su atributi F6, F7 i F10 u visokoj korelaciji. Što je i logično jer je atribut F10 zapravo F6/F7. A možemo primetiti da su u visokoj korelaciji još i F1 i F5, i F2 i F7.



Slika 4: Matrica korelacije

3 Klasifikacija

Klasifikaciju ćemo sprovesti na dva načina, odnosno konstruisaćemo dva modela i uporediti njihove rezultate. Prvi način, koristićemo ansambl metodu slučajne šume (RandomForestClassifier), koja konstruiše više manjih stabla odlu-

čivanja i pravi njihovu najbolju kombinaciju.

Drugi način, koristićemo algoritam K najbližih suseda (KNN). On radi tako što od K najbližih instanci instance koju testiramo dodelimo najbrojniju klasu od K najbližih

3.1 Slučajne šume

Prvo trebamo postaviti pitanje nad kakvim podacima ovaj model najbolje radi, odnosno da li je otporan na nedostajuće vrednosti, i kako se ponaša sa elementima van granica.

U našim skupovima nema nedostajućih vrednosti. Ova metoda je takođe i otporna na elemente van granica, tako da ih nećemo za sada specijalno obrađivati.

Nakon kreiranja modela, testirali smo ga na trening skupu. Na slici 5 možemo videti da za apsolutno svaku instancu tačno određuje njenu klasu. Postavlja se pitanje da li je možda došlo do preprilagođavanja.

Classification report for model RandomForestClassifier on training data

	precision	recall	f1-score	support
A	1.00	1.00	1.00	4285
B	1.00	1.00	1.00	5
C	1.00	1.00	1.00	103
D	1.00	1.00	1.00	352
E	1.00	1.00	1.00	1095
F	1.00	1.00	1.00	1961
G	1.00	1.00	1.00	446
H	1.00	1.00	1.00	519
I	1.00	1.00	1.00	831
W	1.00	1.00	1.00	44
X	1.00	1.00	1.00	522
Y	1.00	1.00	1.00	266
accuracy			1.00	10429
macro avg	1.00	1.00	1.00	10429
weighted avg	1.00	1.00	1.00	10429

Slika 5: Izveštaj za model slučajnih šuma na trening skupu

Testiramo kreirani model sada na test skupu podataka. Izveštaj možemo videti na slici 6

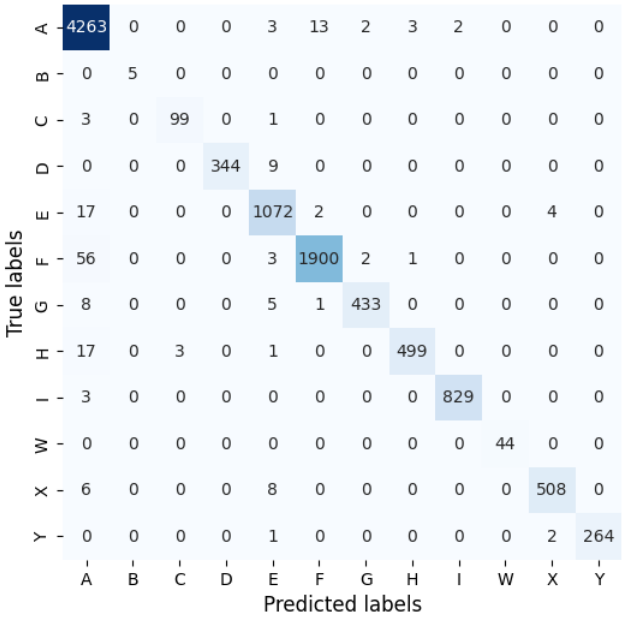
Možemo zaključiti da je model jako dobro istreniran jer i na test skupu pogađa sa 98 procenata tačnosti. Matricom konfuzije ćemo prikazati kako predviđa klase i koje greške pravi. Možemo primetiti da je najčešća greška koju pravi taj da kod klasa koje nisu A su svrstane u A, ali svejedno je taj broj zanemarljiv.

Prikazaćemo i važnost atributa na osnovu kojih pravi podele. Na osnovu gragikona na slici 8 najvažniji atribut je F1, osrednje važni atributi su F2, F3, F4, F5, F9, a najmanje važni F10, F8, F7 i F6

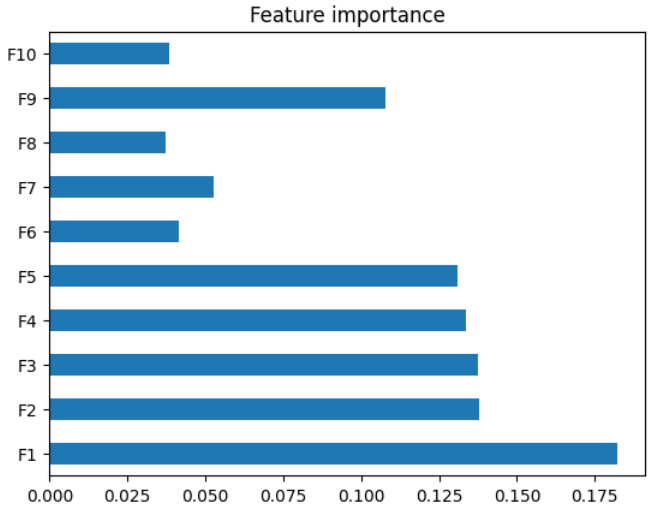
Classification report for model RandomForestClassifier on test data

	precision	recall	f1-score	support
A	0.97	0.99	0.98	4286
B	1.00	1.00	1.00	5
C	0.97	0.96	0.97	103
D	1.00	0.97	0.99	353
E	0.97	0.98	0.98	1095
F	0.99	0.97	0.98	1962
G	0.99	0.97	0.98	447
H	0.99	0.96	0.98	520
I	1.00	1.00	1.00	832
W	1.00	1.00	1.00	44
X	0.99	0.97	0.98	522
Y	1.00	0.99	0.99	267
accuracy			0.98	10436
macro avg	0.99	0.98	0.99	10436
weighted avg	0.98	0.98	0.98	10436

Slika 6: Izveštaj za model slučajnih šuma na trening skupu



Slika 7: Matrica konfuzije



Slika 8: Važnost atributa

3.2 K najbližih suseda

Na model koji se konstruiše KNN algoritmom, elementi van granica mogu imati značajniji uticaj. Zato prvo treba proveriti da li postoje takvi elementi. Njih ćemo pronaći pomoću IQR metode.

	lower	min	num_lower	upper	max	num_upper	percantage
F1	-0.628855	-3.498799	504	0.704281	11.819916	490	10
F2	-0.954663	-2.426761	366	0.898214	386.000000	117	5
F3	-0.367185	-3.210528	870	0.785092	50.000000	85	9
F4	-2.307320	-5.440122	227	2.437528	3.987152	23	2
F5	0.038273	-4.922215	1540	0.395785	1.066121	1093	25
F6	-2.342702	-7.450257	117	2.308082	53.000000	218	3
F7	-0.780208	-11.935457	1382	1.182811	83.000000	175	15
F8	-2.337352	-4.247781	258	2.450252	13.173081	48	3
F9	-1.682079	-5.486218	621	1.810245	44.000000	365	9
F10	-2.086585	-6.719324	234	2.101367	4.671232	231	4

Slika 9: Izveštaj o elementima van granica

Na osnovu slike 9 možemo videti da količina elemenata van granica u pojedinim atributima nije zanemarljiva. Po atributu F5 čak četvrtina elemenata je van granice, a značajniji procenat elemenata van granice je i po atributu F7.

Postoji više načina rada sa elementima van granica. Možemo izvršiti odsecanje, odnosno vrednosti tih atributa ćemo svesti na granične vrednosti. Možemo takođe i izbaciti te elemente iz skupa.

3.3 KNN: Rad sa odsecanjem

Za konstrukciju KNN modela su nam bitni pojedini parametri. Sa koliko najbližih suseda upoređujemo, i na koji način računamo rastojanje. Pomoću GridSearchCV funkcije određujemo te parametre. I ona nam govori da je najbolje da izaberemo parametre nneighbors=2, weights=distance.

Nakon treniranja modela sa datim parametrima, na slici 10 je prikazan izveštaj koliko dobro model radi sa trening podacima.

Classification report for model KNeighborsClassifier on training data

	precision	recall	f1-score	support
A	1.00	1.00	1.00	4285
B	1.00	1.00	1.00	5
C	1.00	1.00	1.00	103
D	1.00	1.00	1.00	352
E	1.00	1.00	1.00	1095
F	1.00	1.00	1.00	1961
G	1.00	1.00	1.00	446
H	1.00	1.00	1.00	519
I	1.00	1.00	1.00	831
W	1.00	1.00	1.00	44
X	1.00	1.00	1.00	522
Y	1.00	1.00	1.00	266
accuracy			1.00	10429
macro avg	1.00	1.00	1.00	10429
weighted avg	1.00	1.00	1.00	10429

Slika 10: Izveštaj za KNN model na trening skupu

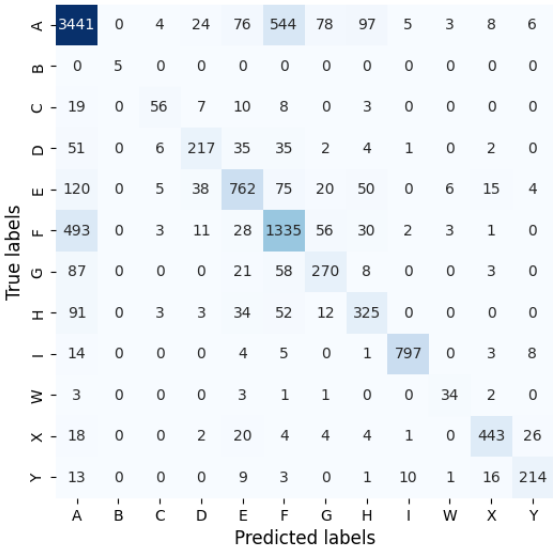
I možemo primetiti da za svaku instancu ispravno pogađa klasu. Opet se postavlja pitanje da li je došlo do preprilagođavanja. Testiramo model sada na test skupu, i rezultate možemo videti na slici 11.

Classification report for model KNeighborsClassifier on test data

	precision	recall	f1-score	support
A	0.79	0.80	0.80	4286
B	1.00	1.00	1.00	5
C	0.73	0.54	0.62	103
D	0.72	0.61	0.66	353
E	0.76	0.70	0.73	1095
F	0.63	0.68	0.65	1962
G	0.61	0.60	0.61	447
H	0.62	0.62	0.62	520
I	0.98	0.96	0.97	832
W	0.72	0.77	0.75	44
X	0.90	0.85	0.87	522
Y	0.83	0.80	0.82	267
accuracy			0.76	10436
macro avg	0.77	0.75	0.76	10436
weighted avg	0.76	0.76	0.76	10436

Slika 11: Izveštaj za KNN model na test skupu

Model radi sa 76 procenata tačnosti, osetno više pravi grešaka u odnosu na metodu slučajnih šuma. Na slici 12 ćemo prikazati matricu konfuzije.



Slika 12: Matrica konfuzije

Možemo primetiti da jako puno grešaka pravi na dve najbrojnije klase u test skupu, A i F.

3.4 KNN: Rad bez elemenata van granica

Kao i u prethodnom delu, pomoću funkcije GridSearchCV ćemo odrediti koje parametre treba izabrati. Ona nam govori da najbolje rezultate dobijamo sa parametrom nneighbors=4, weights=distance.

Nakon treniranja modela, prikazaćemo izveštaje modela i na trening i na test skupu.

Možemo primetiti (Slike 13 i 14) da model dosta greši čak i na trening skupu, a na test skupu su rezultati još slabiji, tek nešto preko pola ispravno određenih klasa. Što je donekle očekivano, jer preko jedne četvrtine podataka izbacujemo iz skupa što nije zanemarljiv broj.

Zaključak, kao najbolja metoda se pokazala ona gde koristimo slučajne šume, tu predviđa skoro pa nepogrešivo. Dok KNN zavisi od toga kako postupamo sa podacima pre samog treniranja modela.

Classification report for model KNeighborsClassifier on training data

	precision	recall	f1-score	support
A	0.80	0.85	0.82	4285
B	0.00	0.00	0.00	5
C	0.37	0.73	0.49	103
D	0.62	0.47	0.53	352
E	0.61	0.76	0.68	1095
F	0.73	0.80	0.76	1961
G	0.76	0.86	0.81	446
H	0.58	0.90	0.70	519
I	0.00	0.00	0.00	831
W	0.59	0.23	0.33	44
X	0.69	0.69	0.69	522
Y	0.00	0.00	0.00	266
accuracy			0.72	10429
macro avg	0.48	0.52	0.48	10429
weighted avg	0.65	0.72	0.68	10429

Slika 13: Izveštaj za KNN model na trening skupu

Classification report for model KNeighborsClassifier on test data

	precision	recall	f1-score	support
A	0.65	0.74	0.70	4286
B	0.00	0.00	0.00	5
C	0.22	0.32	0.26	103
D	0.35	0.22	0.27	353
E	0.48	0.54	0.51	1095
F	0.50	0.58	0.54	1962
G	0.51	0.50	0.51	447
H	0.40	0.57	0.47	520
I	0.00	0.00	0.00	832
W	0.25	0.05	0.08	44
X	0.69	0.68	0.69	522
Y	0.00	0.00	0.00	267
accuracy			0.57	10436
macro avg	0.34	0.35	0.33	10436
weighted avg	0.51	0.57	0.53	10436

Slika 14: Izveštaj za KNN model na test skupu

4 Klasterovanje

Klasterovanje ili nenadgledano učenje se primenjuje u situacijama kada nemamo unapred definisane oznake instanci, već želimo da pronađedmo prirodne strukture povezanosti među podacima. Klasterovanje je pronalaženje grupa objekata takvih da su objekti u grupi što sličniji, i da su objekti u različitim grupama što različitiji.

Da bismo što bolje razumeli klasterovanje, tj. grupisanje elemenata poželjno je da se uradi vizuelizacija tj. da se atributi svedu na 2 ili 3 komponente. Naš skup podataka sadrži čak 10 atributa, tako da je potrebno odraditi neke redukcije, agregacije, kombinacije i slično da bismo ih sveli na manji broj. U našim primerima svešćemo ih na dve komponente pomoću PCA metode.

Nakon toga primenićemo 2 algoritma klasterovanja. Prvi algoritam biće Kmeans, drugi DBSCAN.

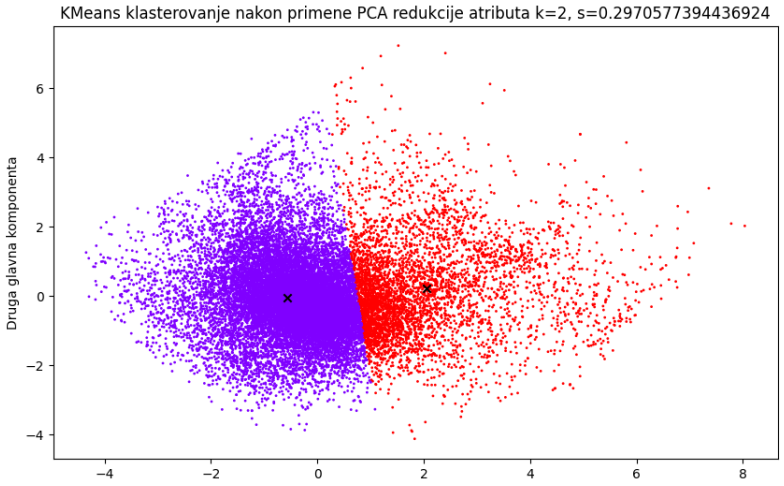
4.1 K-means

K-means pretpostavlja da su klasteri sferičnog oblika i da su konveksni. K-means je metoda particionisanja klasterovanja, što znači da deli skup podataka na predefinisan broj klastera. Pitanje je takođe kako izabrati taj broj. Postoji više metrika koje nam mogu bliže odrediti na koliko klastera treba podeliti dati skup podataka.

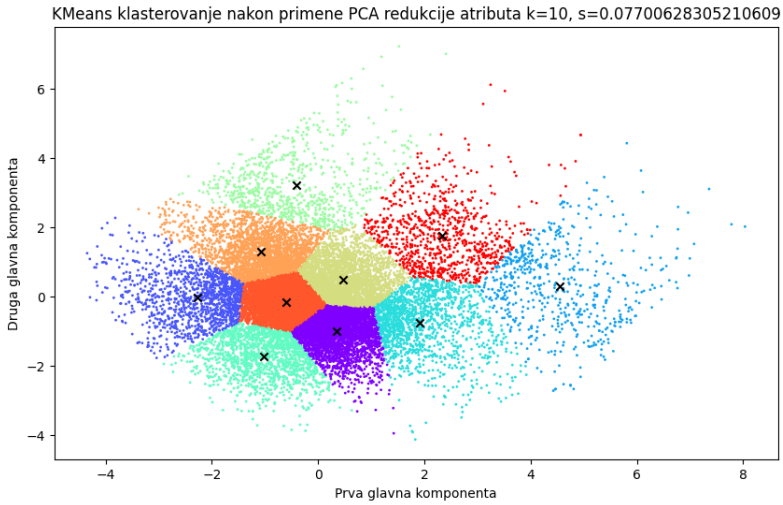
Jedna od tih metrika je: Silueta (Silhouette). Silueta je metrika koja procenjuje kvalitet klasterovanja na osnovu udaljenosti između instanci unutar klastera i udaljenosti do najbližeg susednog klastera. Vrednosti siluete se kreću od -1 do 1, gde vrednosti bliske 1 ukazuju na dobro klasterovanje, vrednosti bliske 0 ukazuju na preklapanje klastera, dok vrednosti bliske -1 ukazuju na pogrešno klasterovanje.

Pošto smo upoznati sa time da ovaj skup podataka ima 12 klasa, smatraćemo da je to najveći mogući broj klastera.

Metrika silueta nam govori da je najbolje podeliti skup na 2 klastera.



Slika 15: Vizuelizacija i primer klasterovanja



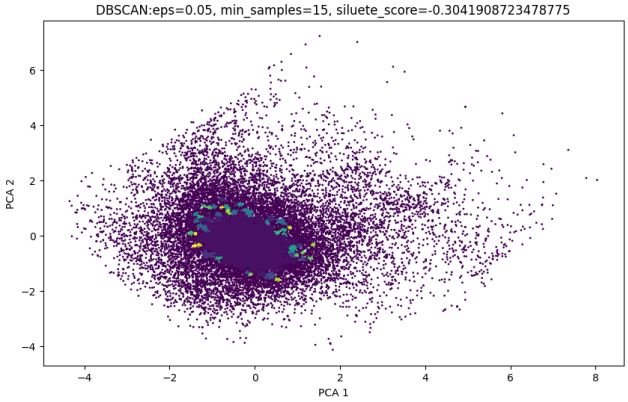
Slika 16: Vizuelizacija i podela na 10 klastera

4.2 DBSCAN

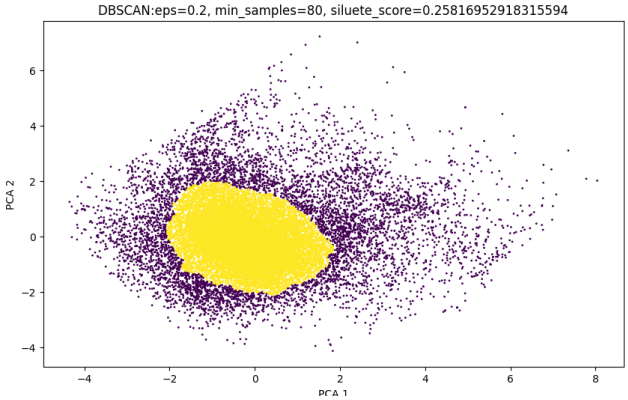
Za razliku od K-menas algoritma, DBSCAN-u ne treba unapred da zadamo broj klastera, već parametre epsilon i minSamples. Epsilon nam govori koliko najdalje mogu da budu udaljene instance da bi se smatrale da pripadaju istom klasteru. MinSamples nam govori koliki je minimalni broj instanci potreban u okolini da bismo ih smatrali dovoljnim za jednu grupaciju. A instance koje u svojoj okolini imaju neke druge instance za koje sa sigurnošću tvrdimo da pripadaju klasteru, a ipak nemaju dovoljan broj bliskih instanci smatramo da su na ivici klastera.

DBSCAN ne pretpostavlja sferične oblike klastera kao K-means, može da dobije skroz nepravilne oblike klastera. Takođe možemo upotrebiti siluetu kao meru kvaliteta izvršenog klasterovanja. U narednim slikama prikazaćemo neke načine klasterovanja.

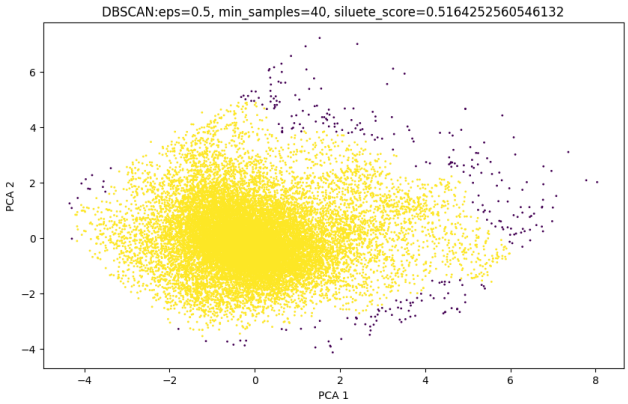
Možemo primetiti da ako izaberemo malo epsilon dolazi do mesanja klastera, što nam govori i silueta, jer ima negativan koeficijent. A ako izaberemo manji broj za minimum uzoraka, u ovom najgušćem delu ce se pojaviti mnoštvo manjih klastera.



Slika 17: DBSCAN sa konkretnim parametrima



Slika 18: DBSCAN sa konkretnim parametrima



Slika 19: DBSCAN sa konkretnim parametrima