Projekat iz Istrazivanja podataka 1

profesor:Nenad Mitic

astitent:Stefan Kapunac

student: Jelena Mitrovic 357/2020

Sadrzaj

1.Uvod

Podaci koji se nalaze u bazi podataka sa kojom radim su prikupljeni od učesnika u eksperimentalnim događajima brzih sastanaka od 2002.-2004. godine. Prisutne osobe bi imale četvorominutni "prvi sastanak" sa svakim drugim učesnikom suprotnog pola. Na kraju njihova četiri minuta, učesnici su upitani da li bi želeli da ponovo izadju na sastanak . Od njih je takođe zatraženo da ocenjuju svoj sastanak na osnovu šest atributa: privlačnost, iskrenost, inteligencija, zabava, ambicija i zajednički interesi. Skup podataka takođe uključuje podatke iz upitnika prikupljene od učesnika u različitim tačkama procesa. Ova polja uključuju: demografiju, navike za sastanke, uverenja o tome šta drugi smatraju dragocenim kod partnera i informacije o životnom stilu.

2.Analiza atibuta

Baza podataka se sastoji od 8378 instanci i 123 atributa.

Atributi baze podatka:

gender: Pol osobe

age: Broj godina osobe

age_o:Broj godina partnera

d_age: Razlika u godinama

race: Rasa osobe

race_o:Rasa partnera

samerace: Da li je vazno da osobe imaju istu rasu(1/0)

importance_same_race: Koliko je vazno da je partner iste rase

importance_same_religion:Koliko je vazno da je partner iste religije

field: Obrazovanje osobe

pref_o_attractive:Koliko je partneru vazna privlacnost

pref_o_sinsere: Koliko je partneru vazna iskrenost

pref_o_intelligence: Koliko je partneru vazna inteligencija

pref_o_funny: Koliko je partneru vazno da je osoba duhovita

pref_o_ambitious:Koliko je partneru vazna ambicija

pref_o_shared_interests: Koliko je partneru vazno da imaju zaj. inter.

attractive_o:Ocena osobe od strane partnera posle sastanka o atraktivnosti

sincere_o: Ocena osobe od strane partnera posle sastanka o iskrenosti

intelligence_o:Ocena osobe od strane partnera posle sastanka o inteligenciji

funny_o:Ocena osobe od strane partnera posle sastanka o duhovitosti

ambitous_o: Ocena osobe od strane partnera posle sastanka o ambicioznost

attractive: Ocena osobe za samu sebe - atraktivnost

sincere: Ocena osobe za samu sebe - iskrenost

intelligence: Ocena osobe za samu sebe -inteligencija

funny:Ocena osobe za samu sebe - duhovitost

ambition: Ocena osobe za samu sebe- ambicija

attractive_partner:Ocena osobe za partnera-atraktivnost

sincere_partner:Ocena osobe za partnera-iskrenost

intelligence_partner: Ocena osobe za partnera-inteligencija

funny_partner: Ocena osobe za partnera-duhovitost

ambition_partner: Ocena osobe za patnera-ambicija

shared_interests_partner: Ocena osobe za partnera-zaj.inter.

sports: Interesovanje osobe za sport

tv sports: Interesovanje osobe za tv sport

exercise: Interesovanje osobe za vezbanje

dining: Interesovanje osobe za izlazak na veceru

museum: Interesovanje osobe za obilazak muzeja

art:Interesovanje osobe za umetnost

hiking:Interesovanje osobe za planinarenje

gaming: Interesovanje osobe za igranje igrica

clubbing: Interesovanje osobe za nocne izlaske

reading:Interesovanje osobe za citanje

shared_interests_o:Ocena osobe od strane partnera posle sastanka o zaj.inter.

attractive_important:Kakvog partnera trazite - atraktivnost

sincere_important: Kakvog partnera trazite - iskrenost

intellicence_important: Kakvog partnera trazite - inteligencija

funny_important: Kakvog partnera trazite - duhovitost

ambtition_important: Kakvog partnera trazite - ambicija

shared_interests_important:Kakvog partnera trazite -zaj.inter.

interests_correlate: Korelacija interesovanja osobe i partnera

tv:Interesovanje osobe za gledanje televizije

theater: Interesovanje osobe za odlazak u pozoriste

movies: Interesovanje osobe za gledanje filmova

concerts: Interesovanje osobe za koncerte

music: Interesovanje osobe za muziku

shopping: Interesovanje osobe za shopping

yoga: Interesovanje osobe za jogu

expected_happy_with_sd_people: Koliko ocekujete da cete biti srecni sa osobama koje upoznate na brzom sastanku

expected_num_interested_in_me: Od 20 osoba koje cete upoznati, koliko ocekujete da ce biti zainteresovano za zabavljanje sa vama

expected_num_matches: Koliko ocekujete pozitivnih odgovora za sledeci sastanak

like: Da li vam se svidja partner

guess_prob_liked: Kolika je verovatnoca da se dopadnete vasem partneru

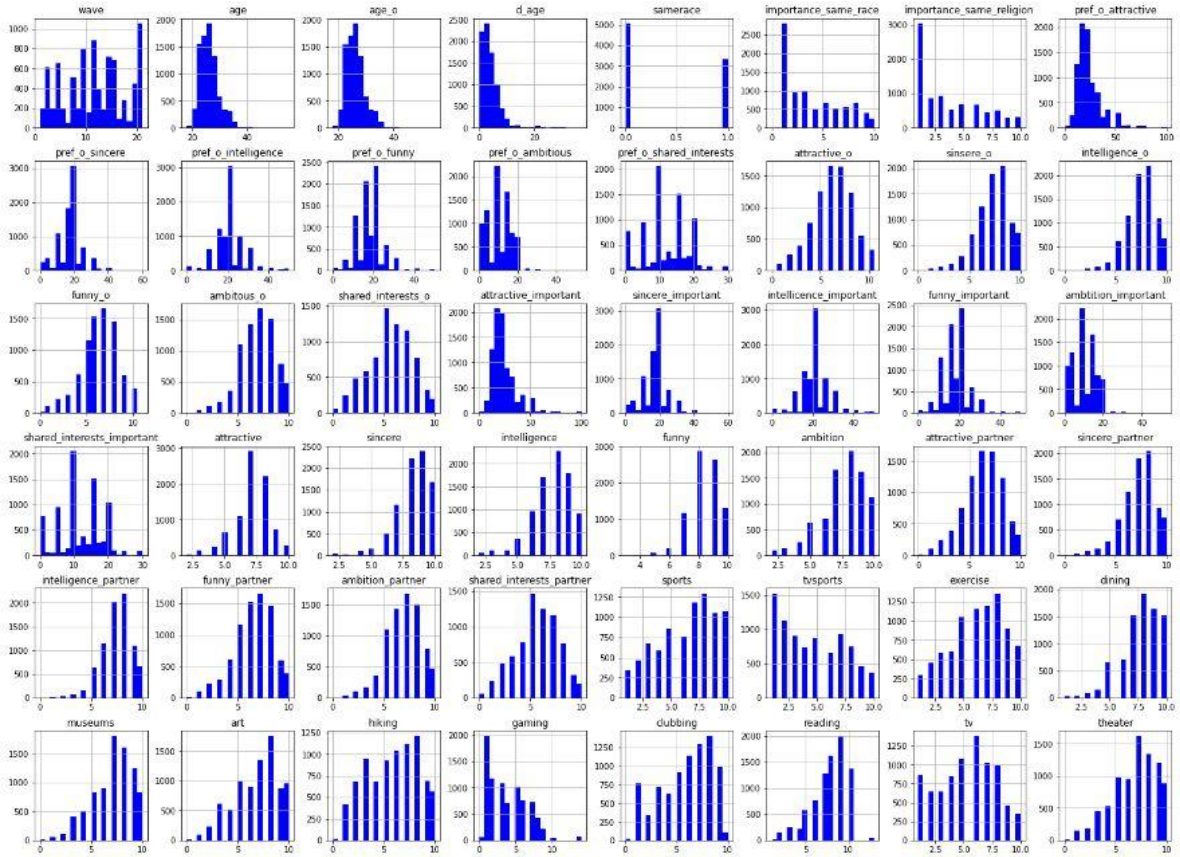
met: Da li ste sreli partera pre ovog sastanka

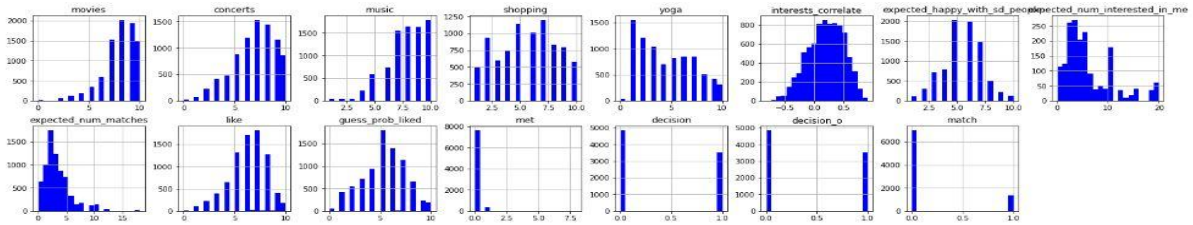
decision: Odluka osobe posle sastanka

decision_o: Odluka partnera posle sastanka

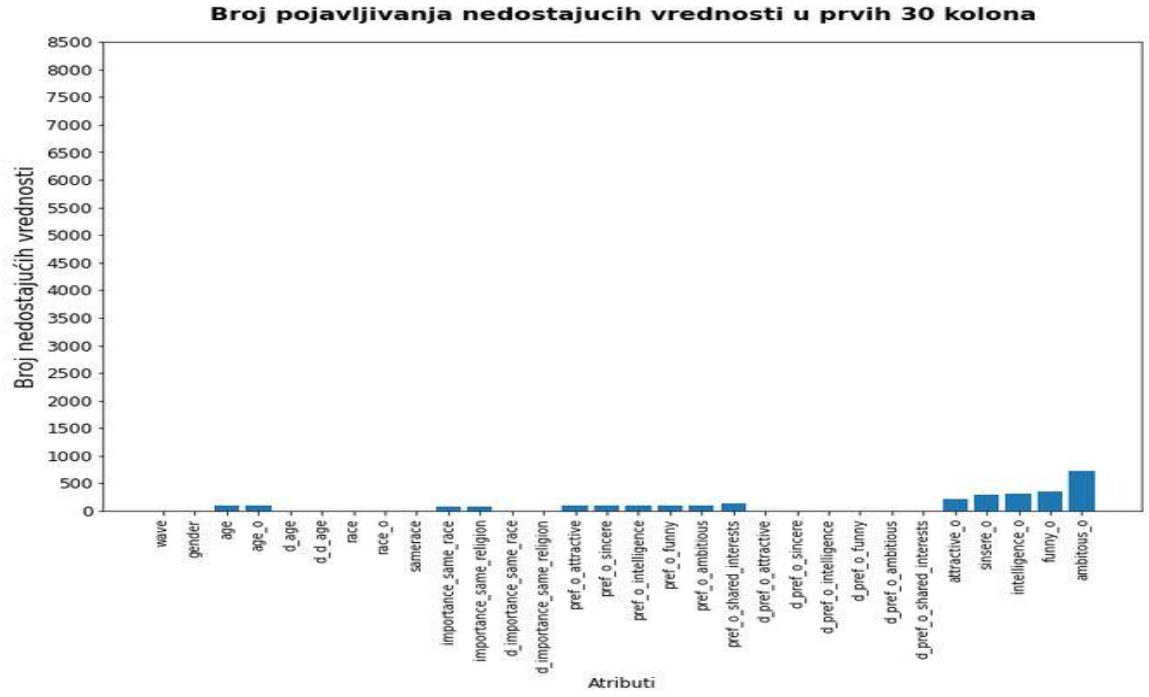
match: Da li ce se osobe opet sresti

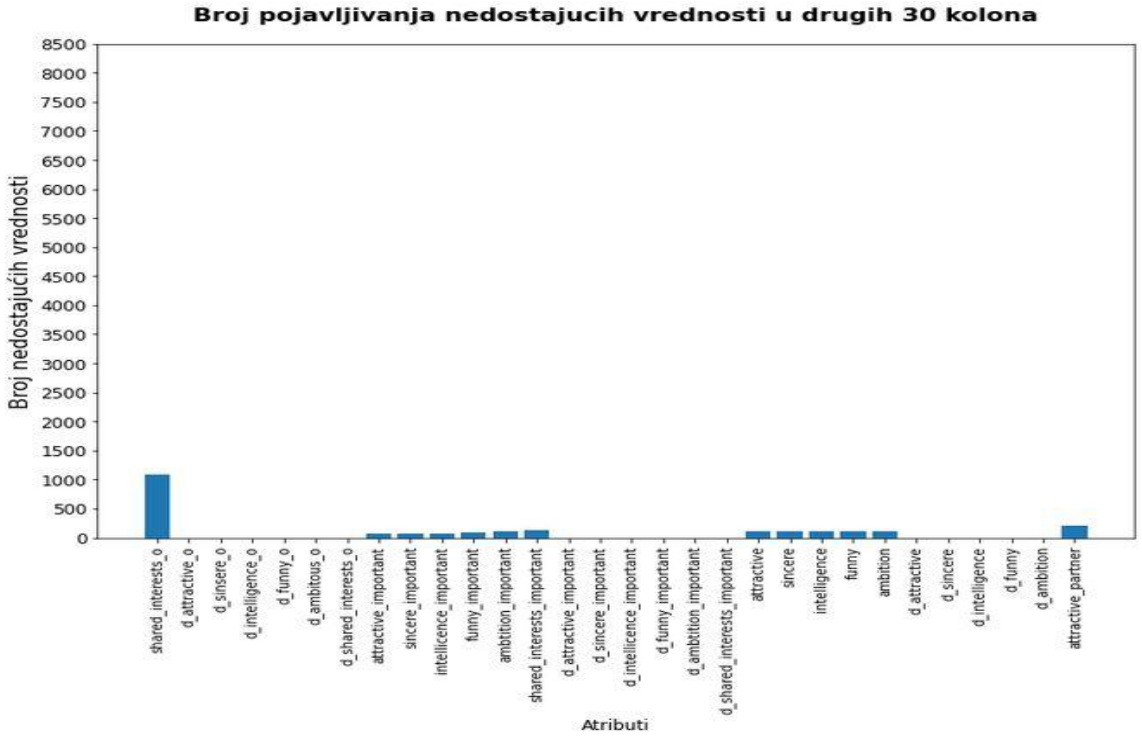
Na slikama ispod prikazan je histogram raspodele podatka nekih od atributa

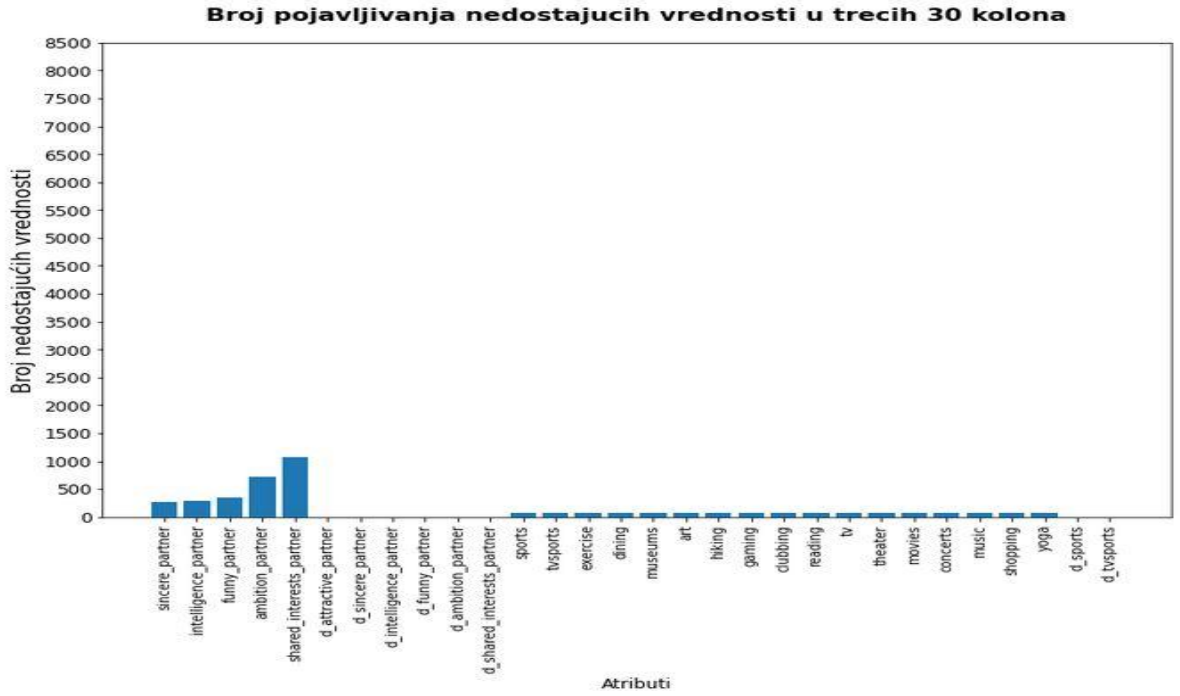


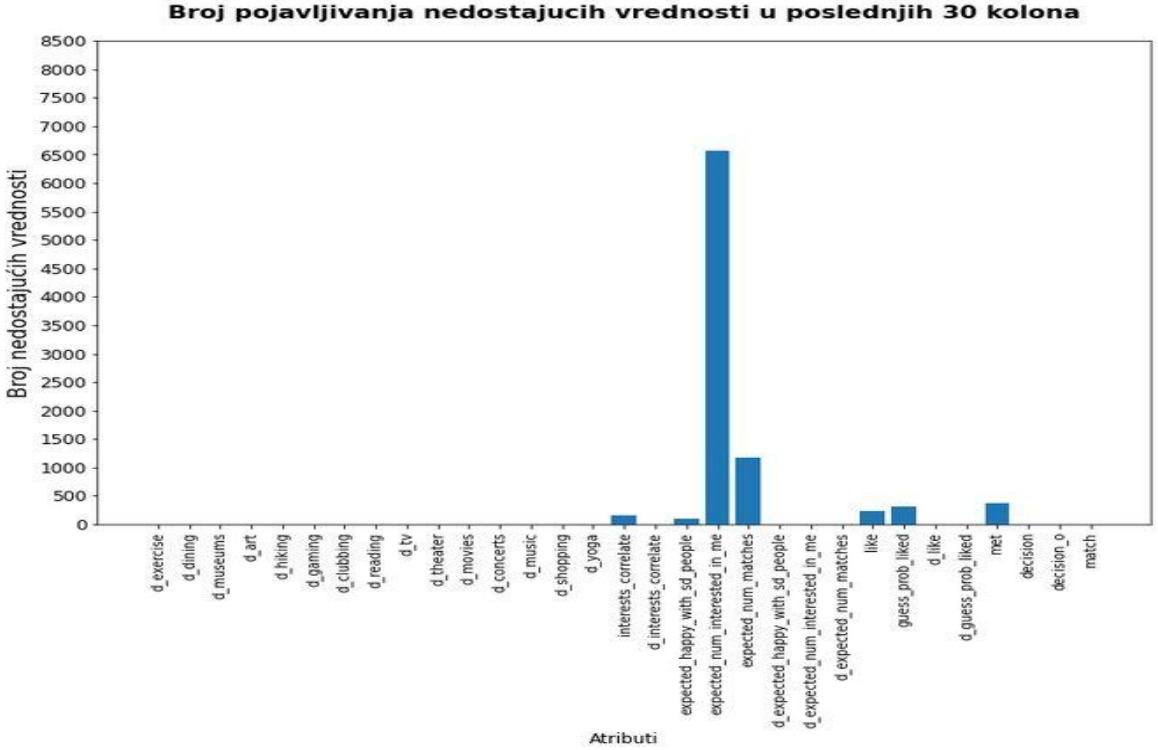


Baza podatka sa kojom radim, ima veliki broj nedostajucih vrednosti.Procenat instanci koje sadrze nedostajuce vrednosti je priblizno 87.5 %. Graficki prikaz nedostajucih vrednosti:





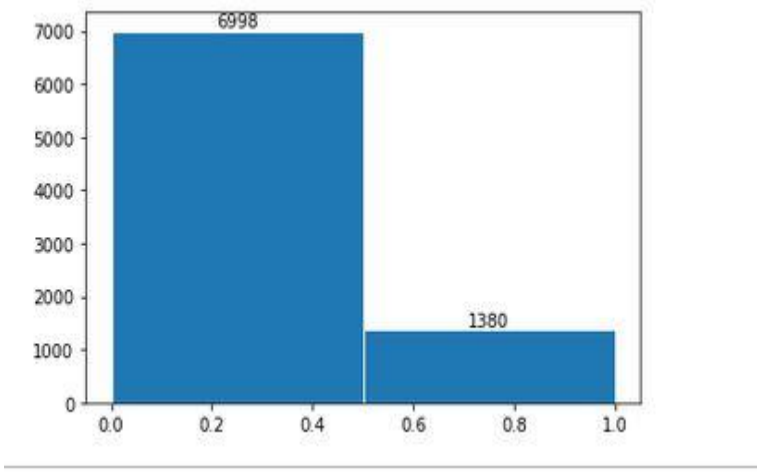




U koraku pretprocesiranja sam sve nedostajuce vrednosti zamenila medijanom za tu kolonu, a sve kategoricke (imenske) atribute sam prevela u numericke koriscenjem LabelEncoder i na taj nacin sam pripremila podatke za klasifikaciju.

3.Klasifikacija

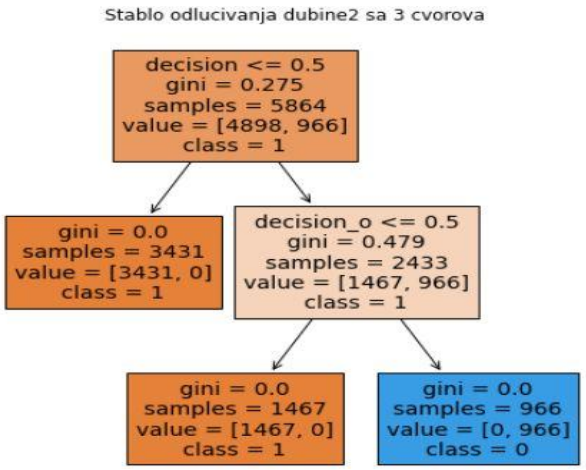
Zadatak klasifikacije u slucaju mog skupa podataka, jeste predvidjanje ciljne promenljive (‘match’), na osnovu ostalih atributa. Dakle, zadatak klasifikacije je predvidjanje da li ce se osobe ponovo vidjeti. Ciljna promenljiva ‘match’ ima 2 vrednosti (0 i 1), pa zakljucujem da se radi o binarnoj klasifikaciji. Na slici ispod je prikazana balansiranost izmedju klasa. Na osnovu slike zakljucujemo da postoji nebalansiranost izmedju klasa, medjutim, mislim da i u jednoj i u drugoj klasi ima dovoljno instanci, pa stoga necu koristiti neke od metoda za smanjenje razlike u dimenziji klasa, medjutim, zbog ovoga necu koristiti tacnost kao kriterijum provere modela, nego f1-meru.



Obradjeni algoritmi klasifikacije: Stabla odlucivanja,Slucajna suma,K najblizih suseda,Bagging Classifier i SVM.

3.1. Stabla odlucivanja

Primenom DecisionTreeClassifier() sa podrazumevanim parametrima, dobila sam sledece stablo odlucivanja:

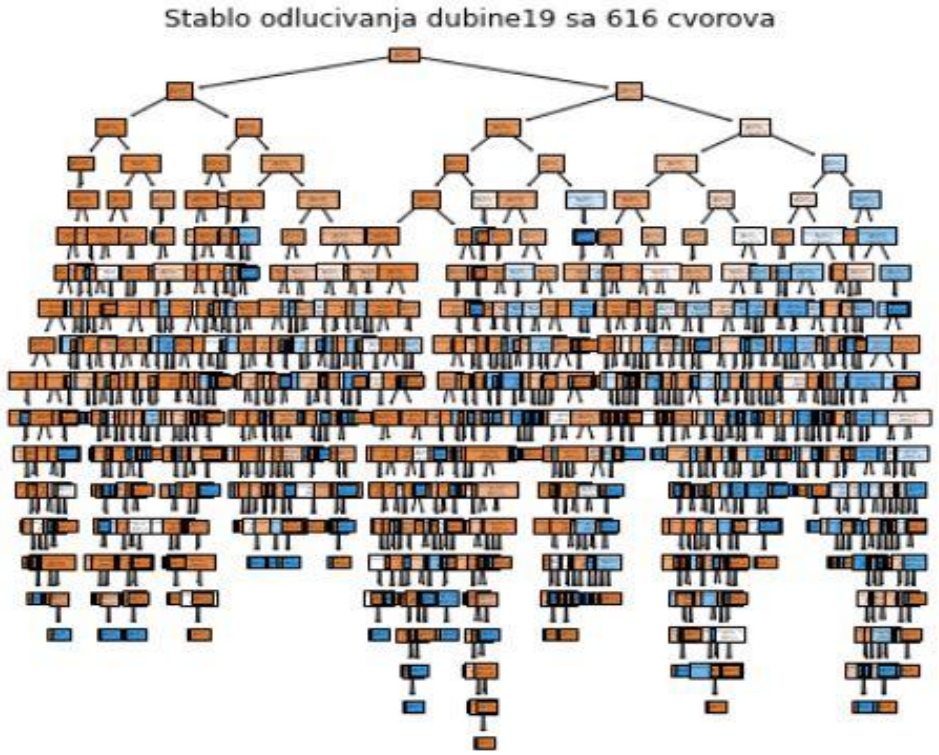


Na slici primecujemo, da se prlikom formiranja stabla koriste samo 2 atributa : ‘decision’ i ‘decision_o’. Uticaji ovih atributa na formiranje stabla su redom: 0.27812303 i 0.72187697, a uticaj svih ostalih jednak je nula. Koriscenjem ovog modela dobijen je f1-score od 1 i na trening i na test skupu, a odgovarajuce matrice konfuzije za trening i test skup redom su:

	1	0
1	4898	0
0	0	966

	1	0
1	2100	0
0	0	414

Posto sam videla da na formiranje stabla uticu samo 2 atributa, htela sam da vidim kako bi izgledalo stablo odlucivanja kada bih izbacila te atribute.. Dobila sam stablo dubine 19 sa 616 cvorova.



Ovaj model ne daje dobre rezultete, zato sto se previse prilagodio trening podacima i izgubio moc generalizacije. Matrice konfuzije na trening i test skupu:

	1	0
1	5248	0
0	0	1035

	1	0
1	1506	244
0	201	143

Model je komplikovaniji nego sto bi trebalo da bude.

3.1.2. Slucajna suma

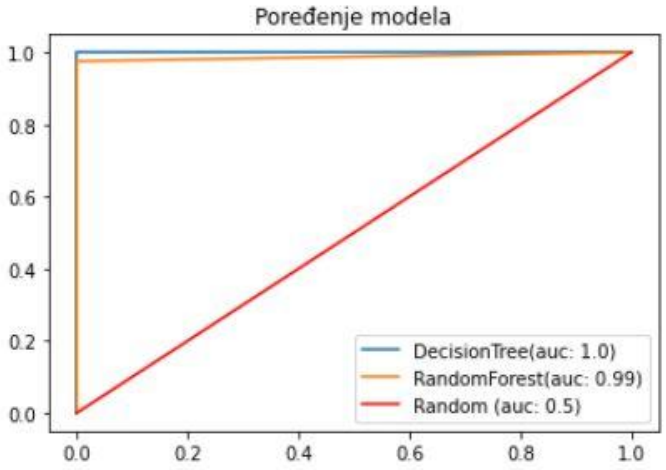
Pozivanjem GridSearchCV() za model RandomForestClassifer, dobila sam da su optimalni parametri ovog modela 'max_depth' =15, 'min_samples_split'= 2, 'n_estimators': 300. Proverom matrice konfuzije, vidimo da model ne pravi greske na trening skupu, ali pravi 10 gresaka na test skupu.

	1	0
1	4898	0
0	0	966

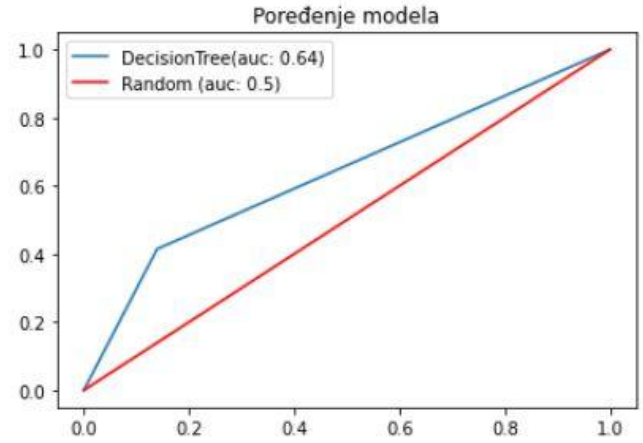
	1	0
1	2100	0
0	10	404

3.1.3. Poredjenje modela

Modele poredimo koristeci ROC krivu. Sa slike ispod primecujemo da su navedeni algoritmi dali jako dobre rezultate,zato sto je povrsina ispod krive blizu jedinice.



Poredim model koji ne sadrzi dva glava atributa sa random modelom. Ovaj model je malo bolji od random modela, ali je losiji u odnosu na gore navedene.



3.2.1 KNN

Pre primene KNN algoritma moramo da normalizujemo podatke. Za normalizaciju sam korstila StandardScaler. Na normalizovane podatke sam primenila KNeighborsClassifier() sa podrazumevanim parametrima. Model pravi jako veliki broj gresaka gde instance klase 1, klasifikuje kao instance klase 0. Odgovarajuce matrice konfuzije na trening i test skupu su:

	1	0
1	4780	118
0	516	450

	1	0
1	2024	76
0	307	107

Model je znacajno losiji u odnosu na stabla odlucivanja, pa sam htela da proverim da li postoje elementi van granica koji mozda kvare tacnost ovog modela. Za proveru elemenata van granica koristila sam IQR algoritam i videla da postoji veliki broj elemena koji se tretiraju kao outlieri(u jednoj koloni je procenat takvih elemenata je cak 49%). Zbog velikog broja outiliera nisam mogla da obrisem te podatke, zato sto bih izgubila mnogo podataka, pa sam odlucila da outliere zamenim medijanom i na taj nacin smanjim njihov broj. Ovim postupkom nisam u potpunosti izgubila elemente van granica, ali se procenat istih znacajno smanjio. Primenom KNeighborsClassifier() sa podrazumevanim parametrima nad ovim podacima dobila sam matrice konfuzije koje su skoro identicne. Zatim, probala sam da podesim hiper parametre. Zbog velike nebalansiranosti, za score za koristila f1 meru. Primenom GridSearchCV algoritma, dobila sam da su optimalni parametri KNeighborsClassifier() algoritma: 'n_neighbors': 10, 'p': 2, 'weights': 'distance', a najveca vrednost f1 scora je piblizno 0.33.Podesavanjem hiper parametara,dobili smo model koji se preprilagodio trening podacima.

	1	0
1	4898	0
0	0	966

	1	0
1	2044	56
0	331	83

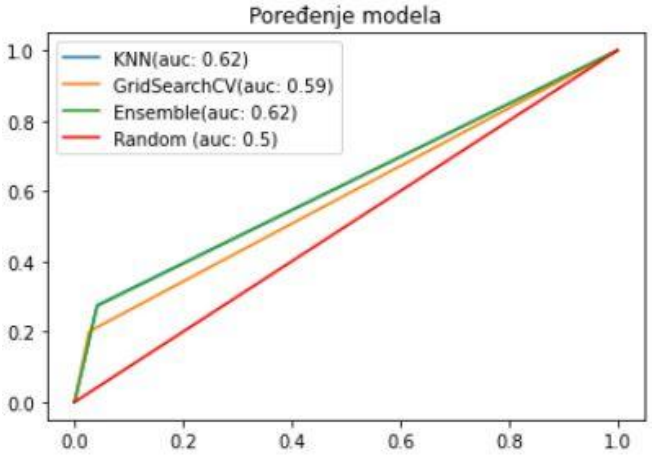
3.2.2.BaggingClassifier

Primenom ove ansambl metode dobila sam priblizno iste rezultate kao i primenom KNN algoritma.Matrice konfuzije na trening i test skupu:

	1	0
1	4812	86
0	537	429

	1	0
1	2014	86
0	301	113

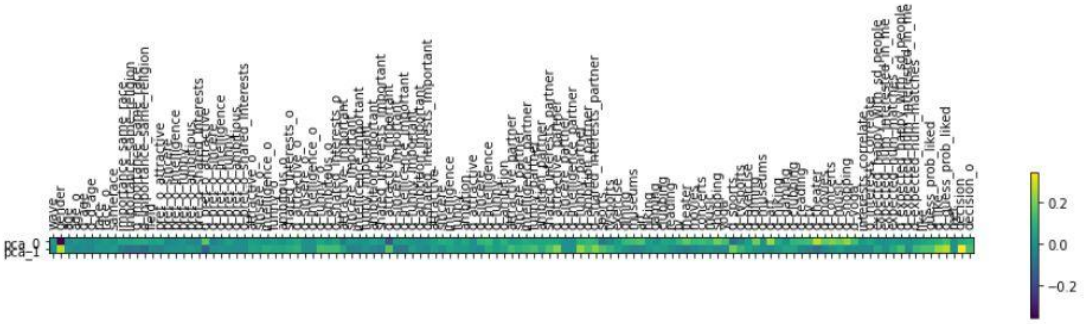
3.2.3 Poredjenje modela



Modeli su znatno losiji u odnosu na stabla odlucivanja. Zbog velike razlike izmedju broja instaci u klasama, mislim da K najblizih suseda nije najbolji izbor za klasifikaciju mog skupa podataka. Upravo zbog toga sto imamo mnogo manje istanci klase 1, vecina suseda ce pripadati klasi 0 i zbog toga model jako veliki broj FN gresaka.

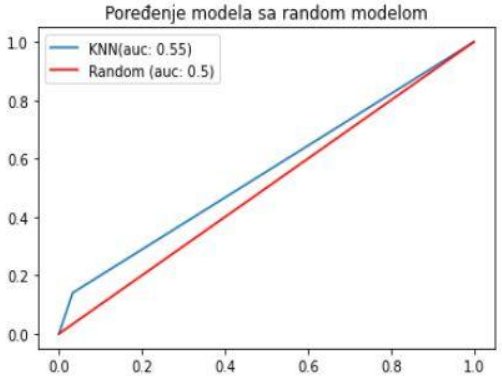
3.2.4 PCA - KNN

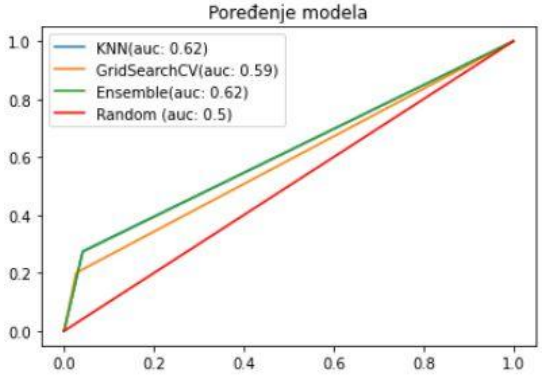
Moja baza podataka se sastoji od 121. atributa, sto je poprilicno puno. Htela sam da proverim da li primenom PCA algoritma mogu da smanjim broj atributa. Prvo sam probala da broj atributa smanjim na dva. Pokusala sam da vizuelno prikazem koliko je koji atribut uticao na pravljenje novih atributa, medjutim zbog velikog broja atributa vizualizacija najjasnija.



Sa slike vidimo da smo sa 2 glavne komponente uspeli da objasnimo samo 15% ukupne varijanse, pa zakljucujem da nije moguce smanjenje broja atributa na dva. Zatim sam probala da smanjim broj atributa, ali da udeo objasnjenje varijanse bude bar 90%. Uz ovoj uslov broj glavnih atirbuta je porastao na 56.Na slici ispod prikazana je kumulativna suma objasnjene varijanse:

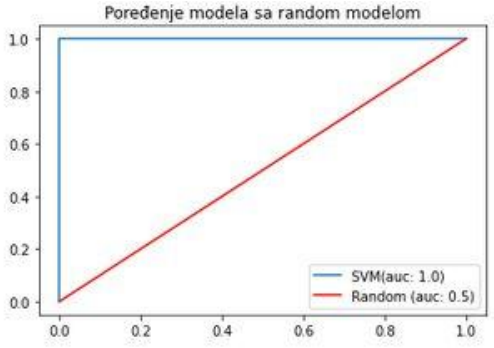
Na transformirane podatke primenila sam KNN algoritam. Na levoj slici ispod prikazano je porednje ovog modela sa random modelom. Dobijeni model je losiji u odnosu na model koji sam dobila primenom KNN algoritma na originalnim podacima (slika desno), ali ne puno losiji.





3.3.1. SVM

Poslednji algoritam klasifikacije koji sam obradila jeste SVM. Ideja je pronaci hiperravan koja razdvaja podatke iz 2 klase. Ako su podaci linearno razdvojeni koristimo marginu, odnosno imacemo uslove da se hipperavan nalazi na bezbednom rastojanju od podataka i da su podaci iz jedne ravni sa jedne strane hiperrvani, a drugi sa druge. Ukoliko podaci nisu linearno razdvojeni koristimo meku marginu, odnosno dopustamo da imamo neke pogresne instance. Primenom GridSearchCV arlogitma za model SVM dobila sam da su optimalmi parametri SVM modela {'C= 0.001, 'kernel'= 'linear'}.GridSearch algoritmu nisam naglasila da kao score koristi f1-meru, medjutim, mislim da nije problem sto je koristio accuracy, zato sto pogledom na ROC krivu i AUC vidimo da je model dobar i sa ovom merom.

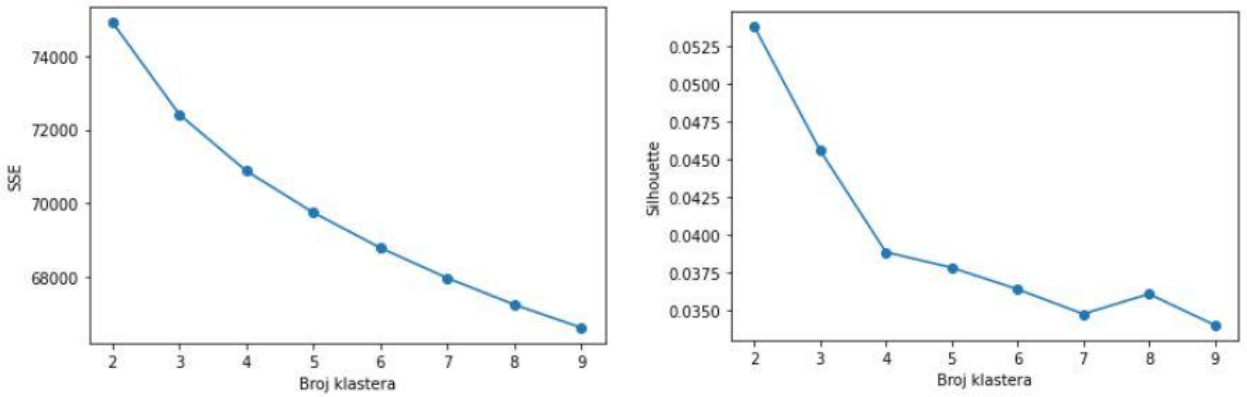


4.Klasterovanje

Za razliku od klasifikacije, gde smo imali ciljnu promenljivu koju je trebalo predvideti, klasterovanje spada u nenadledano ucenje, odnosno, bez ciljne promenljive potrebno je naci neke slicnosti izmedju podataka i grupisati podatke. Obradjeni algoritmi klasterovanja: K sredina, hijararhijsko klasterovanje i DBSCAN. Primenom ovih algoritama zaključila sam da moj skup podataka nije pogodan za klasterovanje.

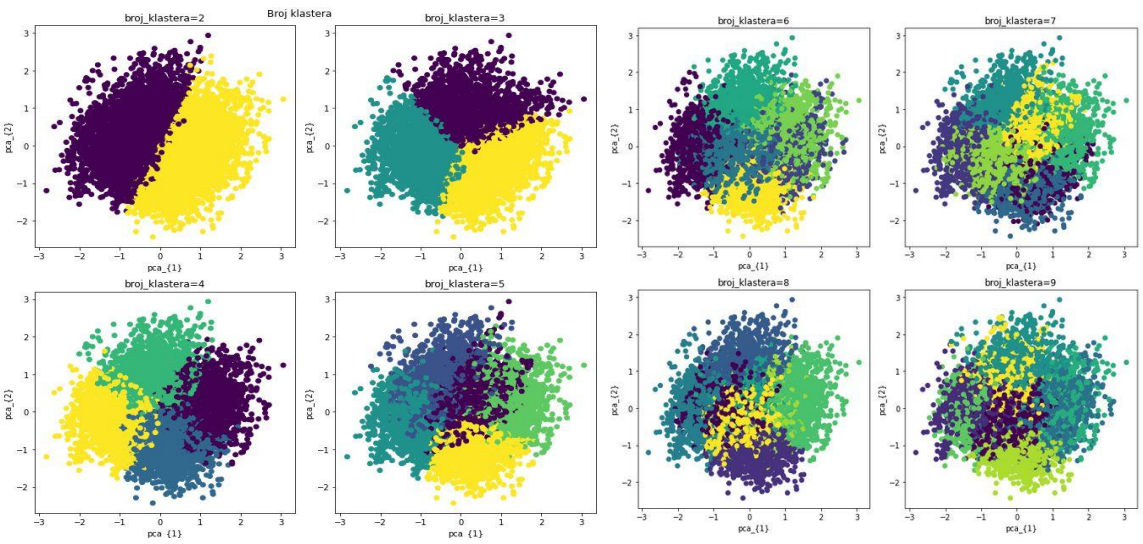
4.1. KMeans

KMeans predstavlja iterativni algoritam koji deli podatke u K klastera.Svaka tacka pripada tacno jednom klasteru. Ovaj algoritam zasniva se na reprezentativnim predstavnicima. Prvi korak prilikom primene ovog algoritma jeste odredjivanje optimalnog broja klastera.Na slici ispod prikazana je promena SSE tokom povecanja broja klastera. Promena SSE nije najbolja mera za izbor otimalnog broja klastera, zbog toga sto se povecanjem broja klastera SSE smanjuje, pa zato moramo da posmatramo i promenu Silhouette.Na slici ispod prikazana je ta promena.



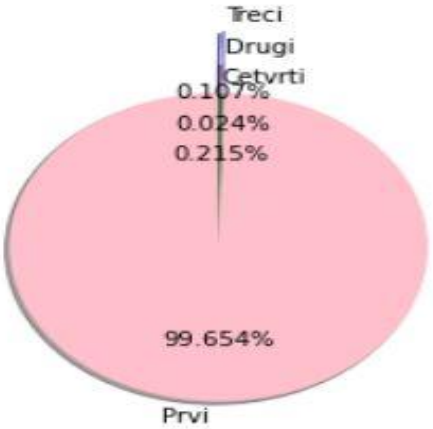
Primenom pravila lakta zakljucujemo da je optimalan broj klastera jednak 4(ili 7?). Silhouette scor za k=4 je priblizno jednak 0.0375, sto je jako lose.

Vec smo videli da primenom PCA algoritma i smanjem broja komponenti na 2 mozemo da objasnimo samo 15% ukupne varijanse podatka, medjutim, uradila sa smanjenje broja komponenti na 2 samo da bih probala da vizualizujem klastere.Na slikama ispod prikazana je promena sa povecanjem broja klastera:



4.2. Agglomerative

Primenom algoritma Agglomerative sa parametrima {n_clusters=4, linkage='average', compute_distances=True}, dobila sam da najveci broj instanci pripada prvom klasteru, cak 8349, drugom klasteru pripada samo 2 instance, trecem 9, a cetvrtom 18.



Vrednost silhouette scora za ove parametre jeste 0.076253, sto je i dalje jako malo. Proverila sam da li se bolji score dobija ako koristimo single ili complete vezu', ali u oba slucaja dobijamo manji silhouette score.

4.2.DBSCAN

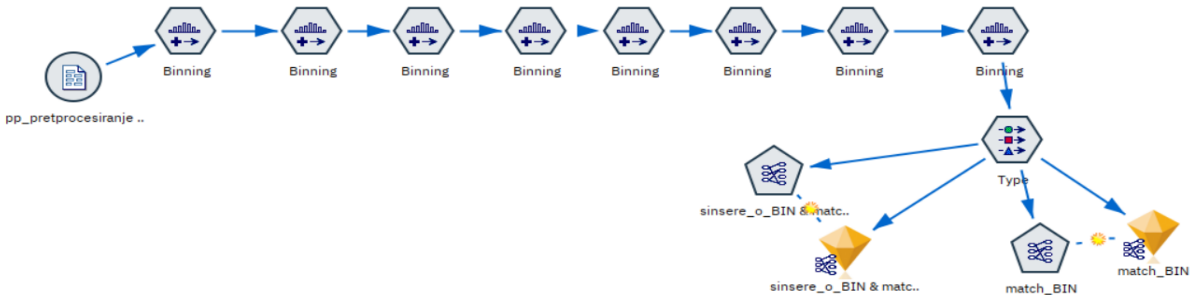
U petlji sam pokusala da odredim koji su dobri parametri ovog modela tako da imamo sto manje suma, odnosno, instanci koje ne pripadaju nijednom klasteru, medjutim, bilo kojom kombinacijom (min_samples_value = range(2,50) eps_values = [0.1, 0.15, 0.2, 0.25, 0.3]) sam dobila da su sve instance šumovi, odnosno ovaj algoritam nije uspeo da pronadje nijedan klaster. Na osnovu svega ovoga zakljucjujem da moji podaci nisu pogodni za klasterovanje.

5.Pravila pridruzivanja

Skup koji sam koristila za pravila pridruzivanja jeste onaj koji sam dobila pretprocesiranjem podataka na pocetku, odnosno zamenom null vrednosti medijanom i koriscenjem LabelEncodera. Zbog jako velikog broja atributa, iskoristila sam Feature Selection da izdvojim glavne atribute. Izdvojila sam 35 najvaznijih atributa, odnosno 36 sa atributom 'match'. Odlucila sam da cilj pravila pridruzivanja u slucaju mog skupa podataka bude nalazenje skupa osobina koje povlace vrednost atibuta 'match = 1', zato sto mi je to imalo najvise smisla.. Obradila sam algoritam Apriori.

5.1.Apriori

Svih 35 atributa koje sam dobila kao najvaznije za predvidjanje cilja bila su neprekidna, pa sam iskoristila cvor Binning..Nad ovakvim tipovima mogla sam da primenim Apriori algoritam.



Prvo sam primenila algoritam sa uslovima da se u glavi moze naci samo atribut ‘match’, a u telu se mogu naci svi ostali atributi. Pravila koja sam dobila nisu preterano zanimljiva, iako imaju visoku ocenu Lift mere, zato sto je dobijeni zakljucak isti kao i onaj koji sam dobila primenom stabla odlucivanja. Ono sto mora da vazi da bi odgovor za ponovni susret bio pozitivan, tj. da je vrednost promenljive ‘match’= 1, jeste da atributi ‘decision’ i ‘decision_o’ takodje moraju da budu True. Dakle, dobila sam da ukoliko je vrednost flag atributa ‘decision’ i ‘decision_o’ true, onda ce se osobe ponovo izadji, bez obzira na vrednost ostalih atributa. Neka od pravila:

Consequent	Antecedent	Support %	Confidence %	Lift
match_BIN	decision_o_BIN decision_BIN	16.472	100.0	6.071
match_BIN	decision_o_BIN decision_BIN like_BIN = 3	12.593	100.0	6.071
match_BIN	decision_o_BIN decision_BIN attractive_o_BIN = 3	11.9	100.0	6.071
match_BIN	decision_o_BIN decision_BIN attractive_partner_BIN = 3	11.9	100.0	6.071
match_BIN	decision_o_BIN decision_BIN funny_partner_BIN = 3	13.046	100.0	6.071
match_BIN	decision_o_BIN decision_BIN d_attractive_o_BIN = 2	11.017	100.0	6.071
match_BIN	decision_o_BIN decision_BIN d_attractive_partner_BIN = 2	11.017	100.0	6.071
match_BIN	decision_o_BIN decision_BIN d_ambitous_o_BIN = 2	10.098	100.0	6.071
match_BIN	decision_o_BIN decision_BIN d_ambition_partner_BIN = 2	10.098	100.0	6.071

Probala sam da izbacim atribute ‘decision’ i ‘decision_o’ i dobila sam sledeca pravila.Pravila imaju manju pouzdanost i lift meru od gore navedenih, ali nam daju neke nove informacije u odnosu na stabla odlucivanja.

Consequent	Antecedent	Support %	Confidence %	Lift
match_BIN match_BIN = 3	shared_interests_o_BIN = 3 funny_o_BIN = 3 like_BIN = 3 attractive_o_BIN = 3 attractive_partner_BIN = 3	5.908	65.051	3.949
match_BIN	d_guess_prob_liked_BIN = 3 funny_o_BIN = 3 like_BIN = 3 attractive_o_BIN = 3 funny_partner_BIN = 3	5.312	64.944	3.943
match_BIN	guess_prob_liked_BIN = 3 funny_o_BIN = 3 like_BIN = 3 attractive_o_BIN = 3 funny_partner_BIN = 3	5.3	64.865	3.938
match_BIN	d_guess_prob_liked_BIN = 3 funny_o_BIN = 3 attractive_o_BIN = 3 attractive_partner_BIN = 3 intelligence_o_BIN = 3	5.085	64.319	3.905
match_BIN	guess_prob_liked_BIN = 3 funny_o_BIN = 3 attractive_o_BIN = 3 attractive_partner_BIN = 3 intelligence_o_BIN = 3	5.073	64.235	3.9

6. Literatura

1. Materijali sa predavanja profesora Nenada Mitića

2. Materijali sa vezbi iz kursa Istrazivanje podataka:
https://github.com/MATF-istrazivanje-podataka-1/materijali_2022-2023

3.Scikit-Learn Machine Learning in Python: https://scikit-learn.org/stable/