Analiza skupa podataka

Loans Default

Dataset

Stefan Kerkoč

***Uvod***

U ovom izveštaju prikazan je proces istraživanja podataka, preuzetih sa Kaggle web strane, a koji sadrže informacije o bankovnim kreditima. Cilj istraživanja je izrada modela koji predviđaju da li ce neko biti u mogućnosti da otplati kredit na osnovu 32 atributa.

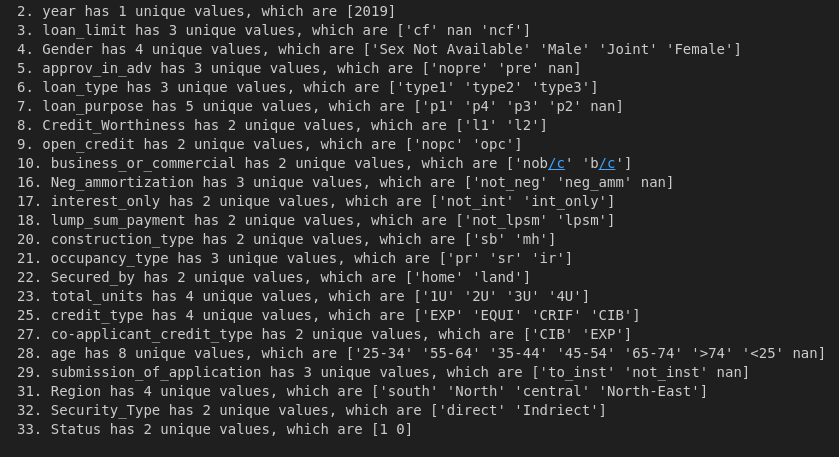
Podacima možete pristupiti klikom na [link](https://www.kaggle.com/datasets/yasserh/loan-default-dataset?resource=download)

***Ekspolativna analiza podataka***

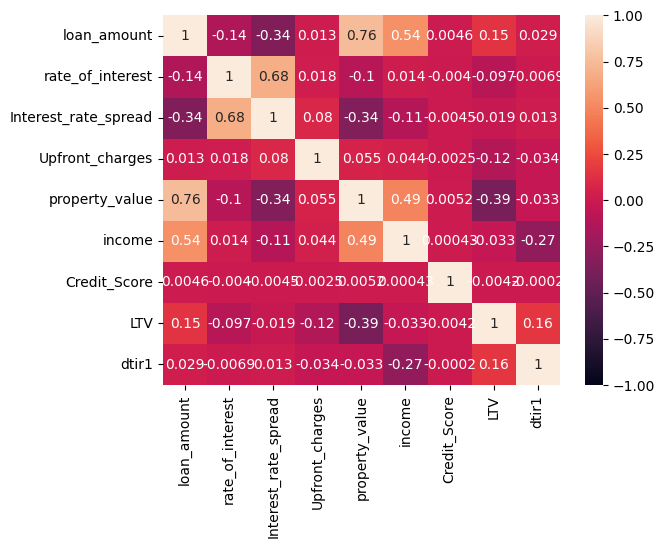
Pre pretprocesiranja, potrebno je da uradimo analizu podataka. Nažalost naši atributi dolaze bez pojašnjenja šta koji od njih znači. Ipak, za neke je moguće pretpostaviti, a kompletna lista atributa je sledeća:

* loan\_limit
* Gender
* approv\_in\_adv
* loan\_type
* loan\_purpose
* Credit\_worthiness
* open:credit
* business\_or\_commercial
* loan\_amount
* rate\_of\_interest
* Interest\_rate\_spread
* Upfront\_charges
* term
* Neg\_ammortization
* interest\_only
* lump\_sum\_payment
* property\_value
* construction\_type
* occupancy\_type
* Secured\_by
* total\_units
* income
* credit\_type
* Credit\_score
* co-applicant\_credit\_type
* age
* submission\_of\_application
* LTV
* Region
* Security\_Type
* dtir1
* Status

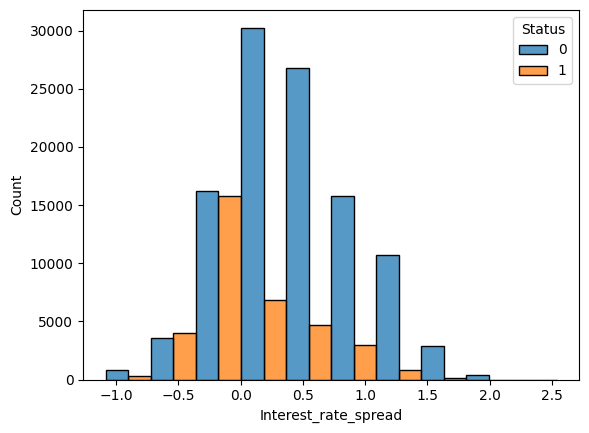
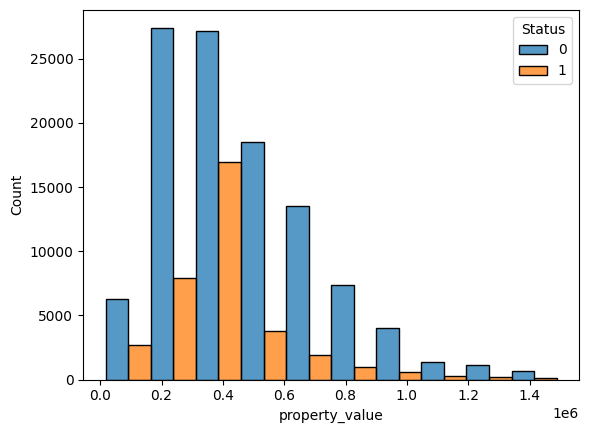
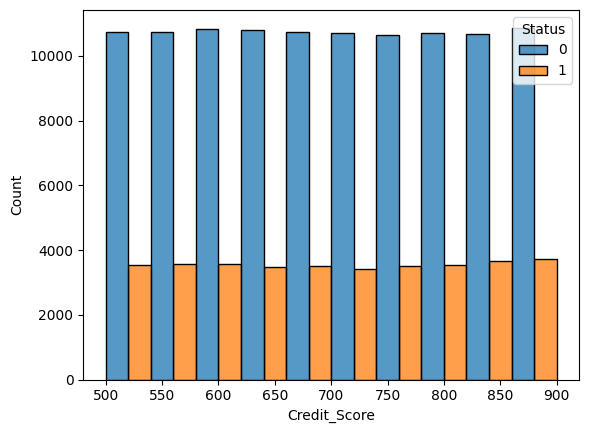
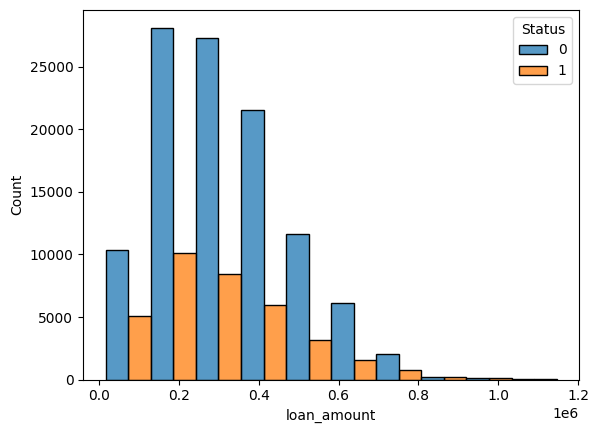
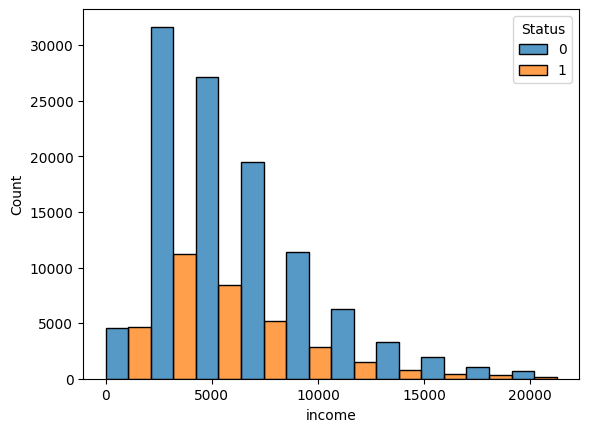
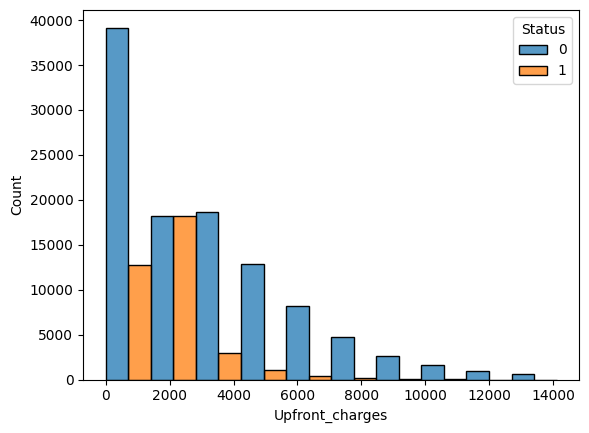
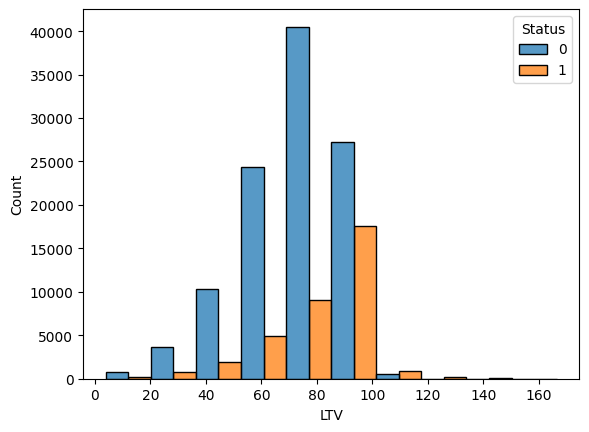
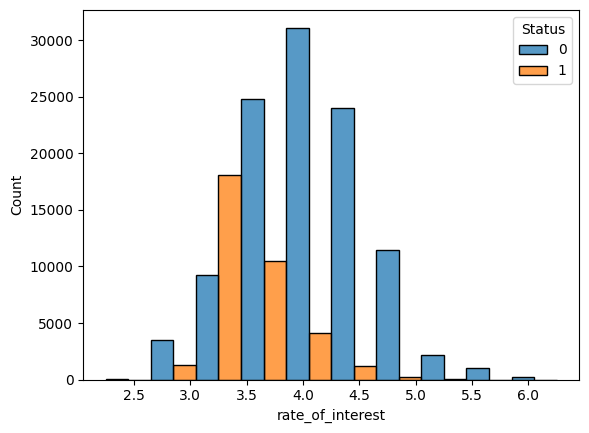
Prva stvar koju treba da uradimo je da odredimo tip atributa, a to možemo uraditi tako što pogledamo pod kojim tipom podataka se čuvaji i, značajnije, broj jednistvenih vrednosti.

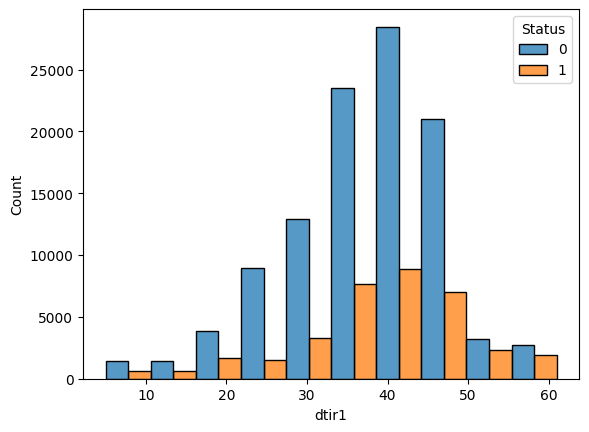
Slika 1: broj jedinstvenih vrednosti

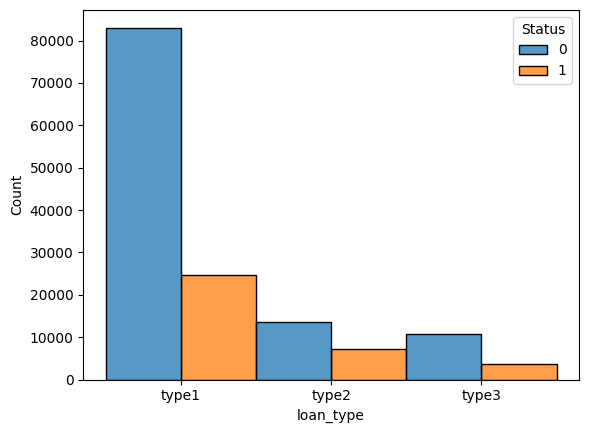
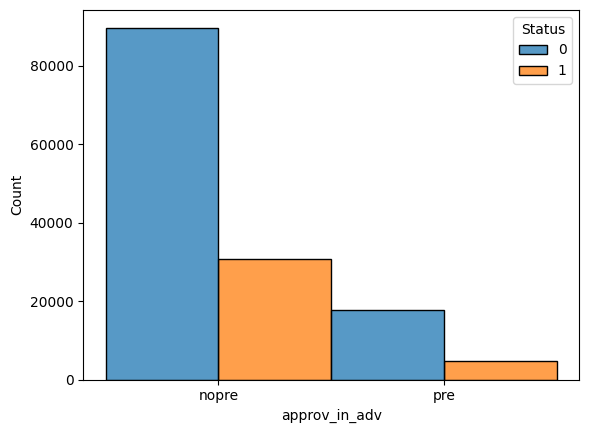
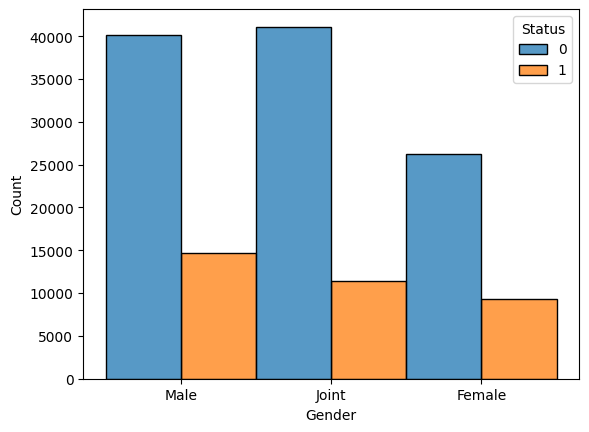
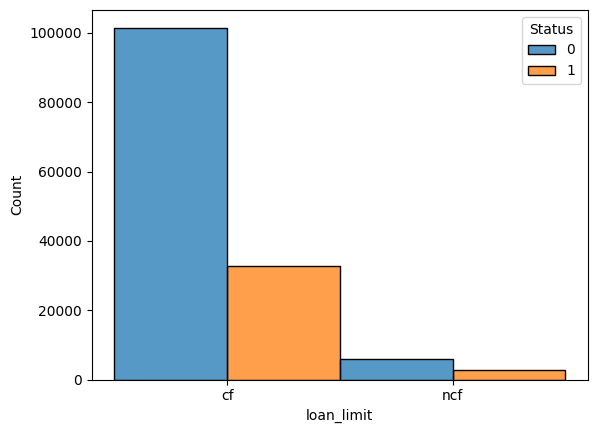
Nakon pretprocesiranja, o kojem će biti reči kasnije, možemo proveriti korelacije između ulaznih atributa

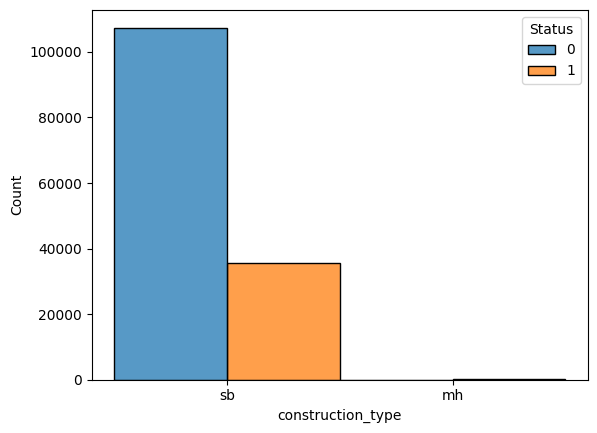
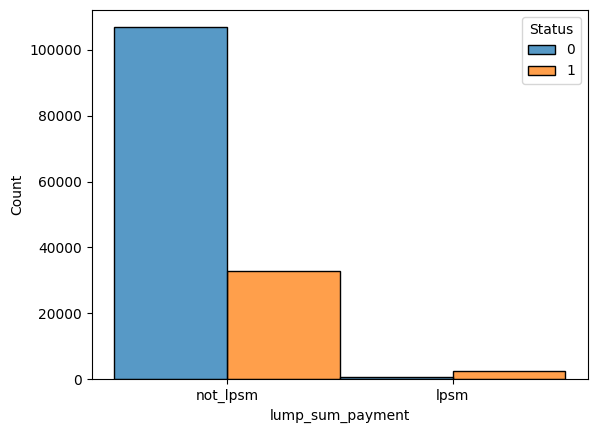
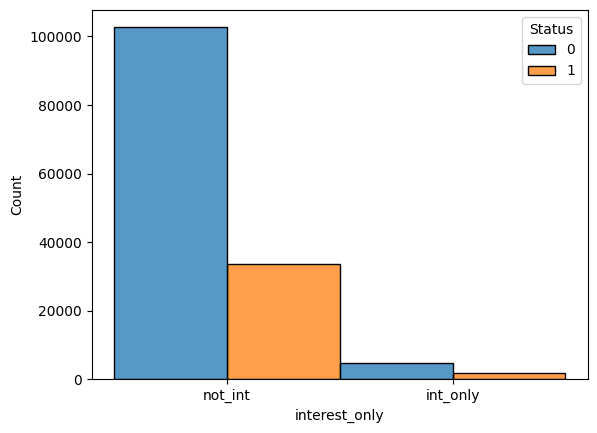
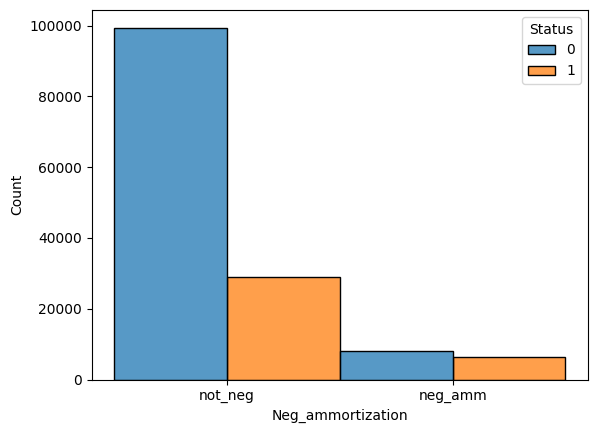
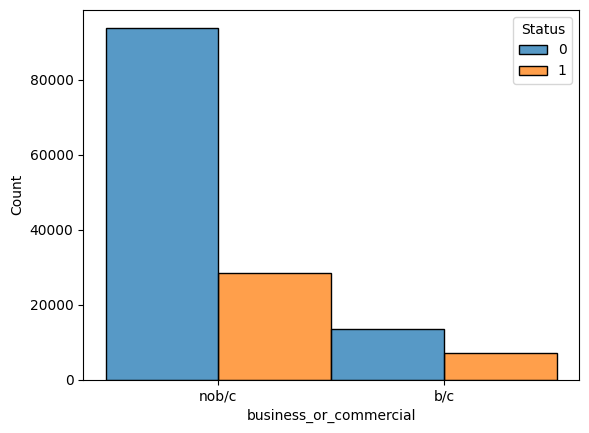
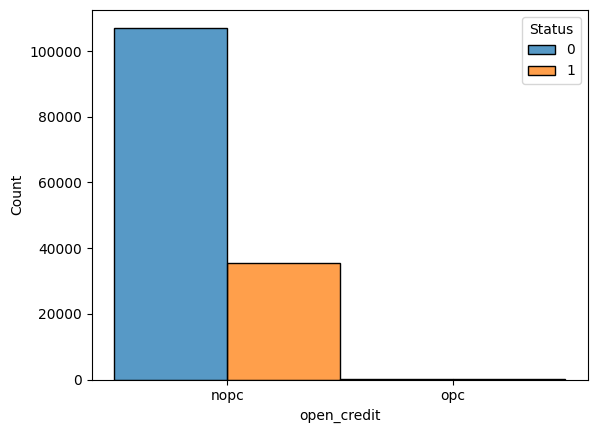
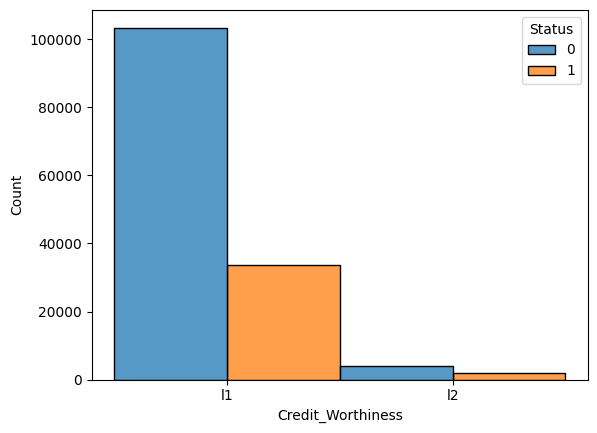
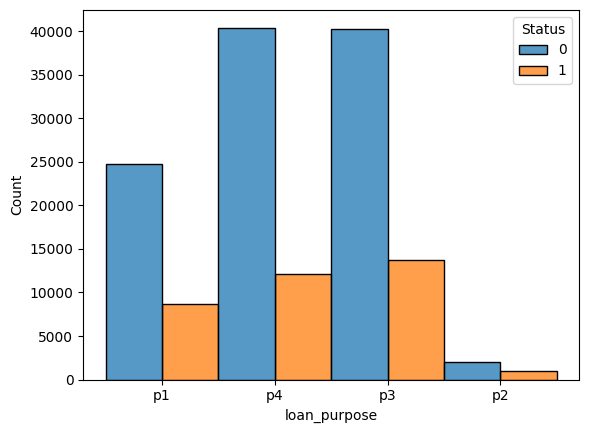
Slika 2: mapa korelacija

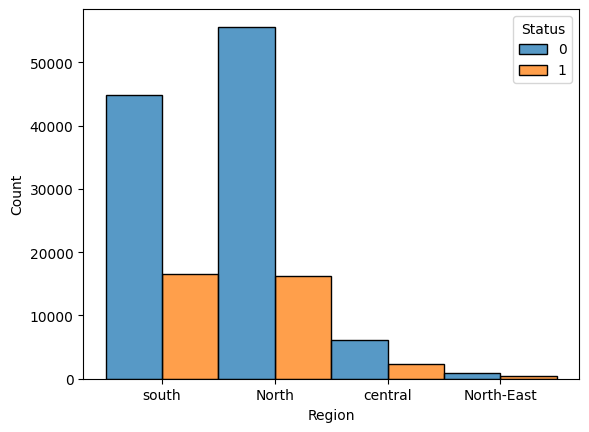
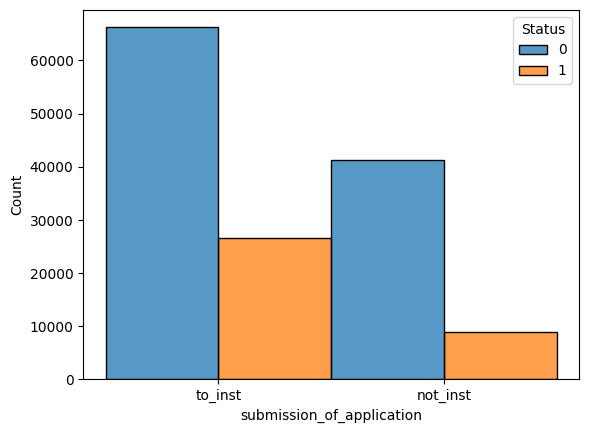
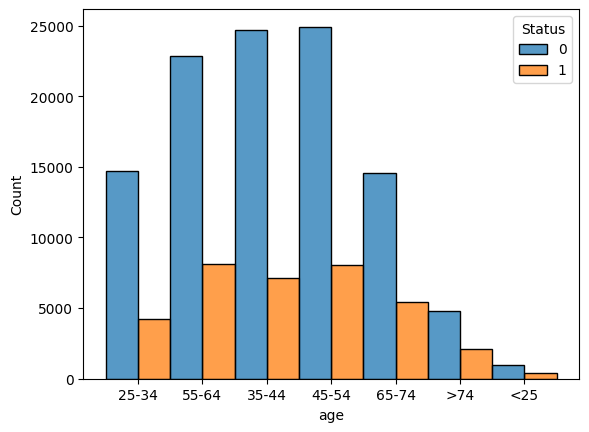
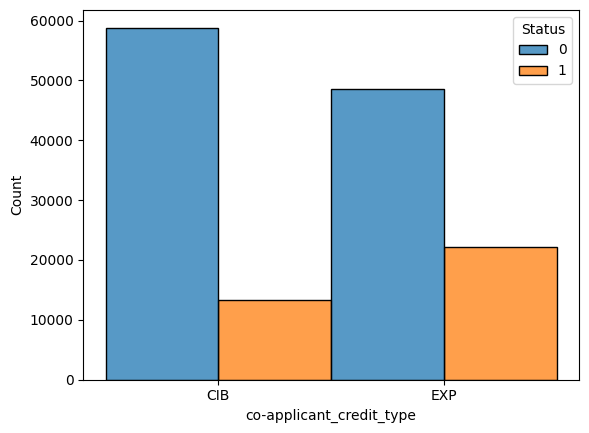
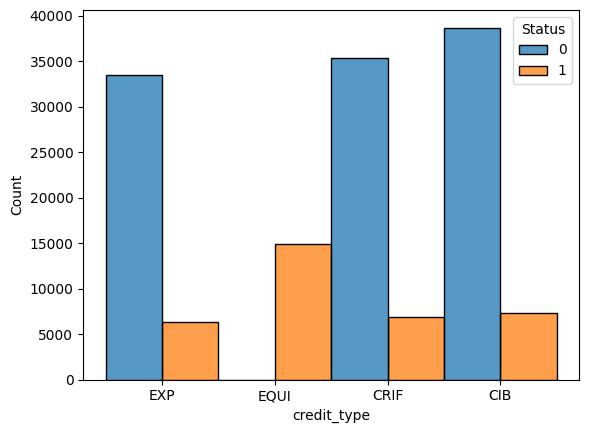
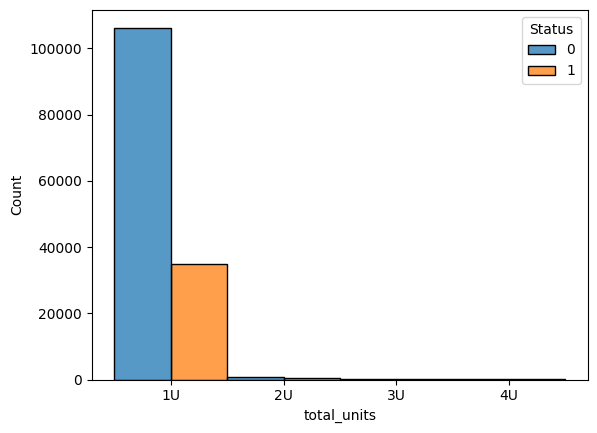
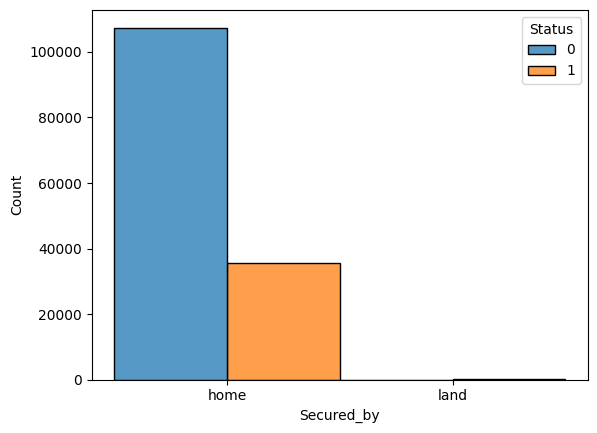
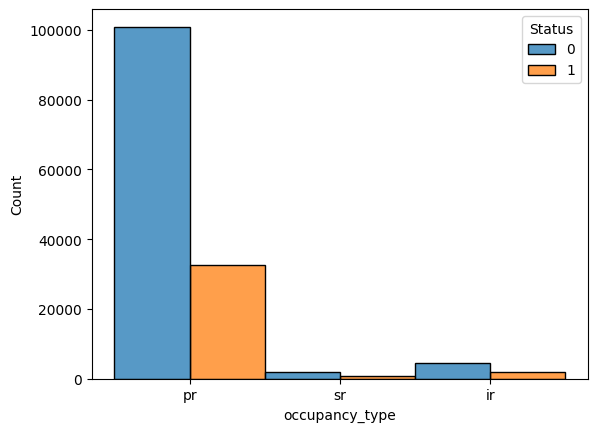
Vidimo da atributi property\_value, income i loan\_amount imaju jaku korelaciju, što važi i za atribute rate\_of\_interest i Interest\_rate\_spread. Ove činjenice su veoma logične i nisu iznenađujuće.

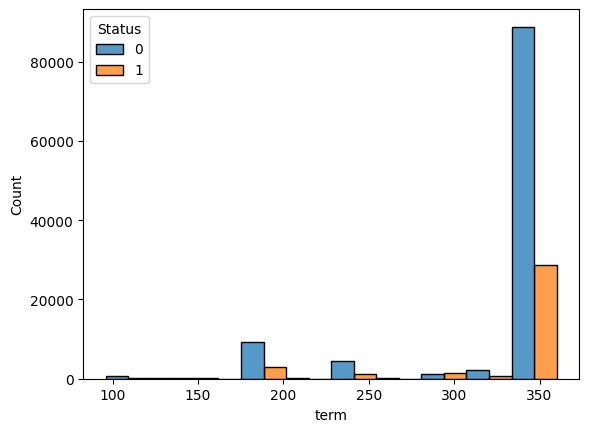
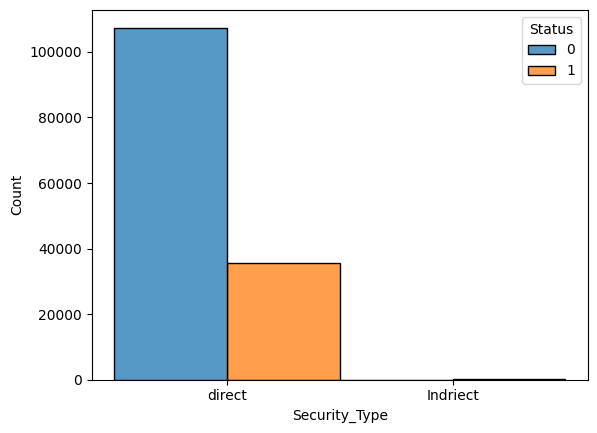
Dalje, želili bi da vidimo kako naš ciljni atribut zavisi od pojedinačnih ulaznih atributa.

Slika 3: Zavisnost statusa od numeričkih atributa



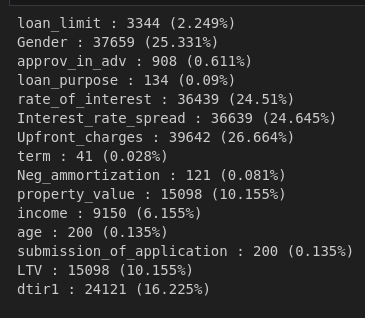




Slika 4: Zavisnost statusa od kategoričkih atributa

***Pretprocesiranje***

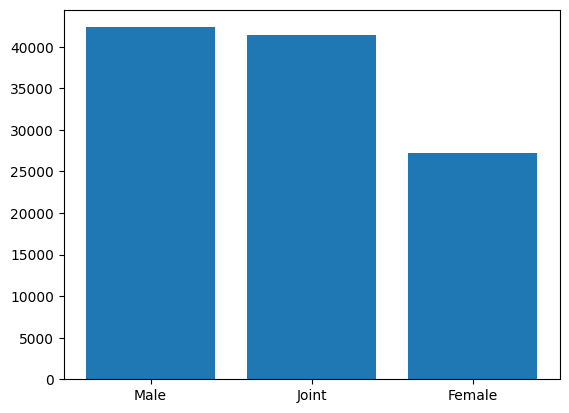
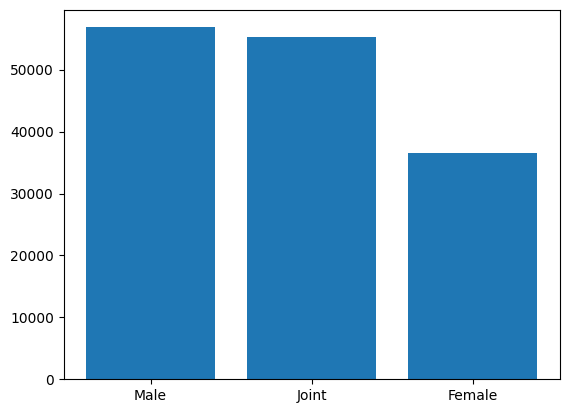
Prvi korak pri pretprocesiranju je da rešimo problem nedostajućih vrednosti. Prvo možemo da nađemo broj i procenat nedostajućih vrednosti po koloni. Nakon toga tražimo korelaciju između nedostajućih polja sa slicnim procentima. Dolazimo do zaključka da za atribute LTV i property\_value važi da ako jednog nema, nema ni drugog, a veoma jako povezanost u nedostatku podataka primećujemo u poljima Interest\_rate\_spread, Upfront\_charges i rate\_of\_interest, gde u svim redovima u kojima nedostaje poslednji, nedostaju i druga dva, koji fale u malo(<3%) većem broju redova. Kolona Gender koja ima sličan procenat nedostajućih vrednosti ne pokazuje jaku korelaciju u nedostajanju sa ova 3 atributa.

Slika 6: Broj nedostajućih vrednosti zajedničkih za LTV i property\_value, odnosno rate\_of\_interest, Interest\_rate\_spread i Upfront charges

, Slika 7: Broj nedostajućih vrednosti Izajedničkih zaGender i gorepomenuta tri atributa

Slika 5: Broj i procenat nedostajućih vrednosti

Gender je kateogrička promenljiva, pa nedostajuće vrednosti možemo popuniti koristeći bfill ili ffill. Iscrtavanjem grafa sa brojem vrednosti po kolonama pre i posle popunjavanja grubo proverimo da li je odnos ocuvan.



Slika 8: Broj vrenosti po kolonama pre i posle popunjavanja nedostajućih vrednosti za atribut Gender

Za ostale kategoričke atribute možemo koristiti metodu ffill i bfill zajedno za popunjavanje nedostajućih vrednosti, i ne moramo da proveravamo odnos nakon popunjavanja jer je njihov procenat nedostajućih vrednosti mali, pa bilo kakvo popunjavanje neće pokvariti odnos.

Za numeričke podatke koristimo KNNImputer. On radi tako što za red r1 koji ima nedostajuće vrednosti nađe k redova koji su mu najsličniji po kolonama koje nemaju nedostajuće vrednosti, a u njima ima nešto upisano u kolonama koje su prazne u redu r1, a zatim u prazna polja r1 upiše interpoliranu vrednost atributa iz tih k redova.

Nakon ovoga, izbacimo outlier-e.



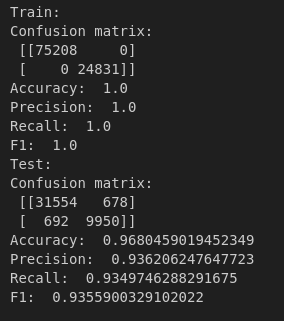
Slika 9: Broj i procenat izbašenih redova

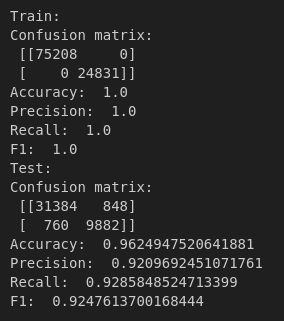
Sada vršimo kodiranje kategoričkih atributa, i to dummy kodiranje.

Nakon toga, spremni smo da za klasifikaciju podelimo podatke u train I test skup u odnosu 70/30, dok za klasterovanje skaliramo podatke koristecći StandardScaler.

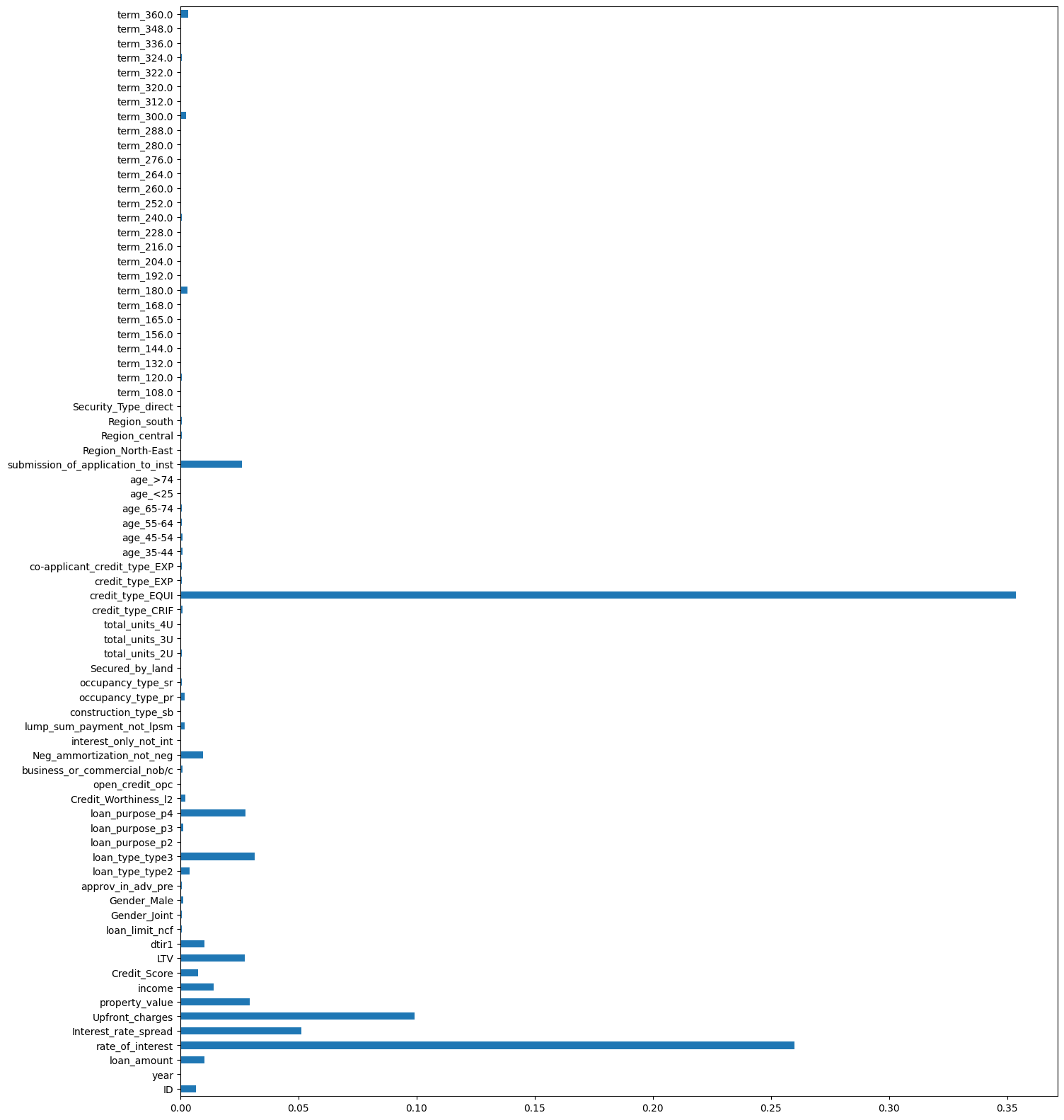
***Klasifikacija***

Prvi algoritam koji koristimo je Decision Tree. Iako se model prepirlagodjava nad trening podacima, i kada se ne podese hiper parametri, a i kada se podese koristeći GridSearchCV, daje odlične rezultate nad trening skupom. Sa ~96% tačnošću i ~92 f1 merom kada se ne podese hiper parametri i ~97% tačnošću i ~94% f1 merom kada se podese, ovo će biti naš najbolji model.

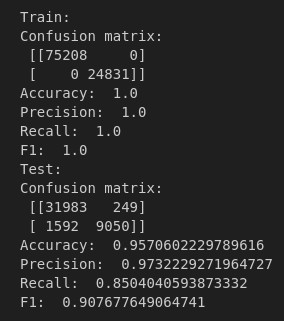


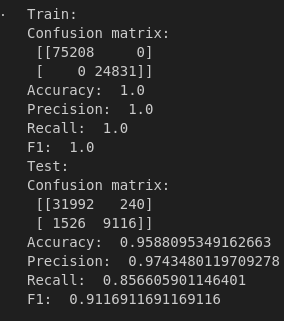


Slika 10: Model DecisionTree – mere

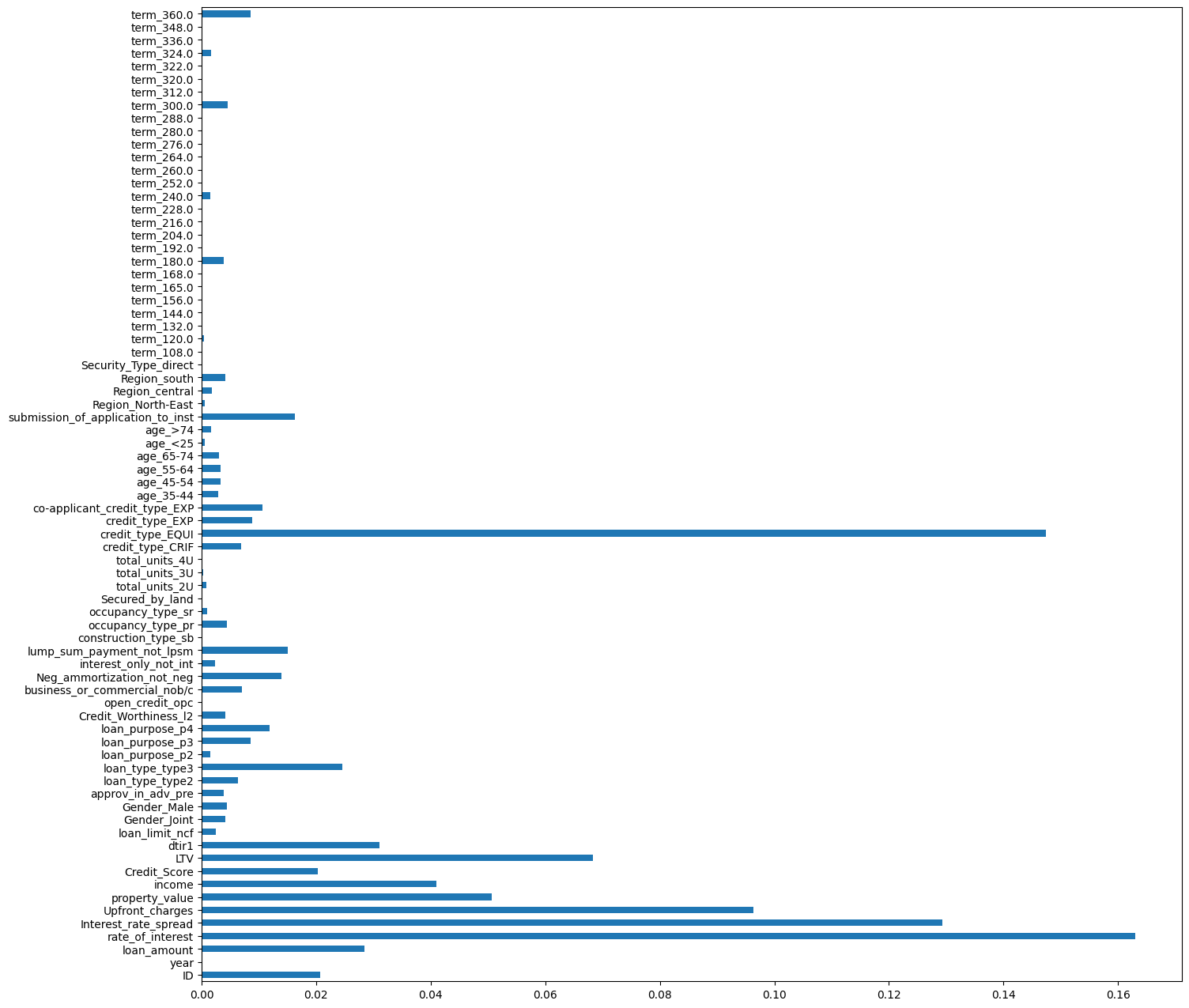


Slika 11: Graf feature\_importance-a za DecisionTree model

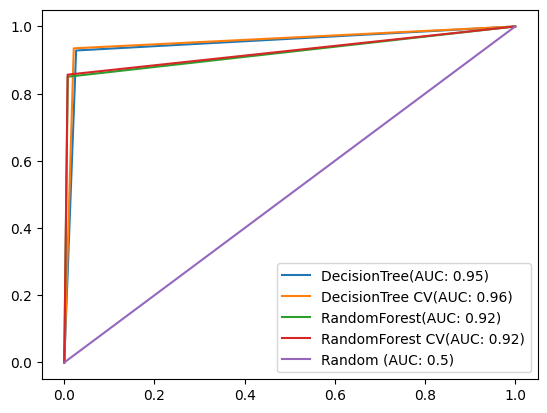
RandomForrest algoritam nam iznenađujuće daje gore rezultate u svemu osim u preciznošću od običnog DecisionTree algoritma.



Slika 12: Model Random Forrest – mere

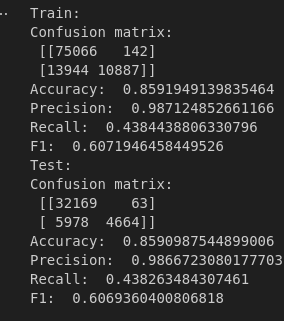


Slika 13: Graf feature\_importance-a za RandomForrest

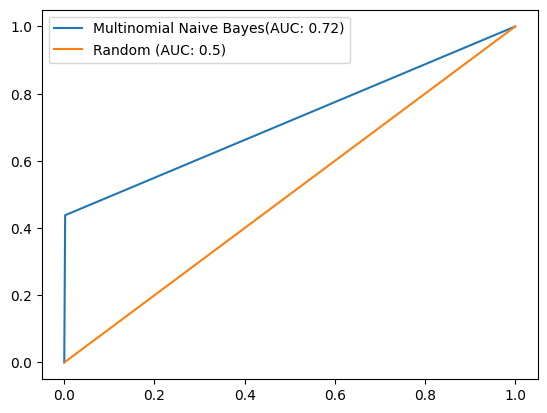


Slika 14: ROC krive i AUC za DecisionTree i RandomForrest

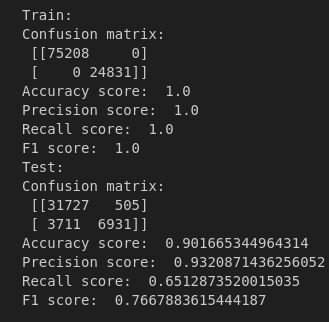
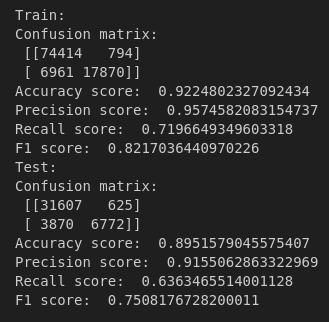
Multinomijalni naivni Bajesova algoritam nam ocekivano ne daje dobre rezultate, s obzirom na prirodu skupa podataka.



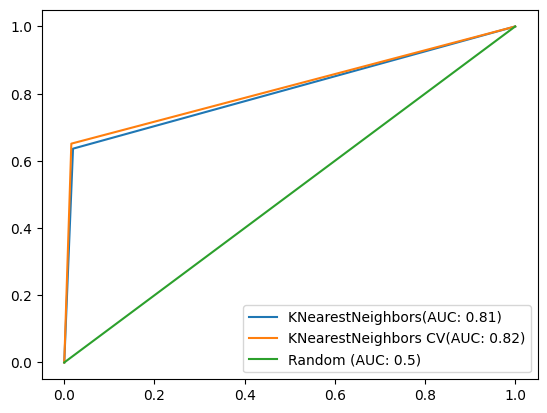
Slika 15: Model MultinomialNB – mere

 Slika 16: ROC kriva i AUC za MultinomialNB

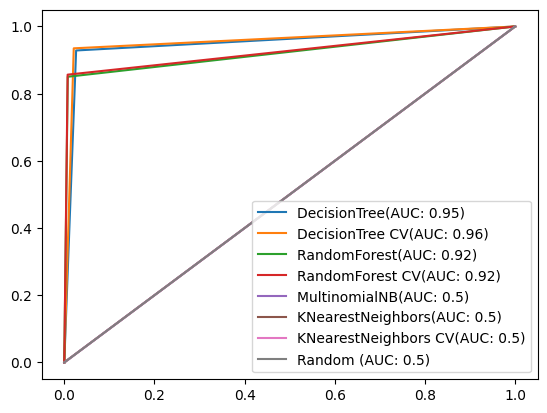
KNN algoritam nam daje dobru tačnost i preciznost međutim odziv je loš. Kako bi zadržali dobro, a poboljšali loše, kao za scoring hiperparametar postavljamo f1\_scorer, međutim ne vidimo preterano veliko poboljšanje



Slika 17: Model KNN – mere

Slika 18: ROC kriva i AUC za KNN

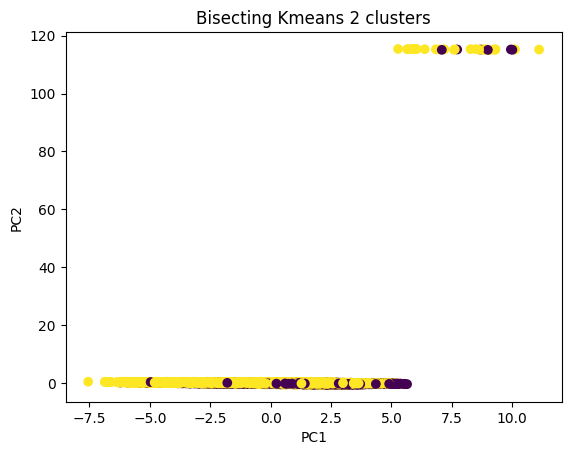
***Poređenje modela***

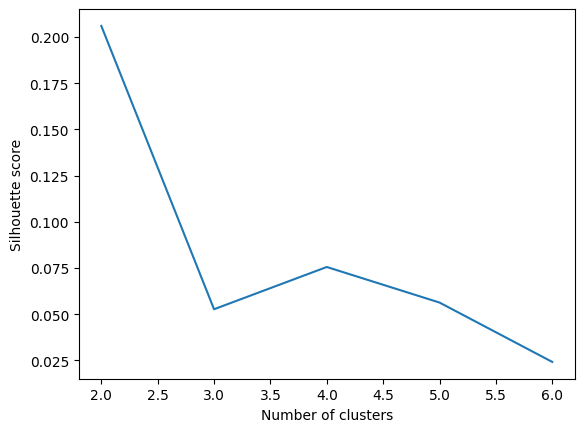
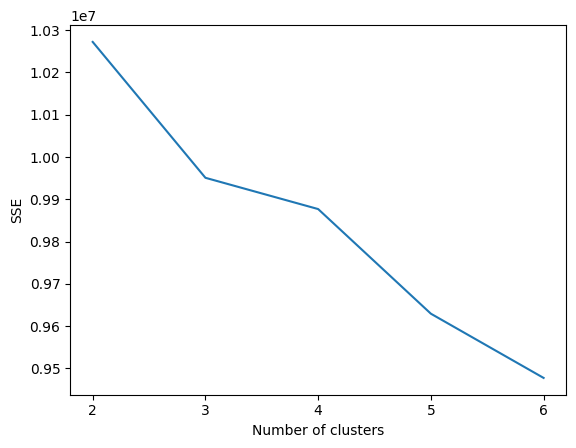


Slika 19: Poređenje modela klasifikacije

***Klasterovanje***

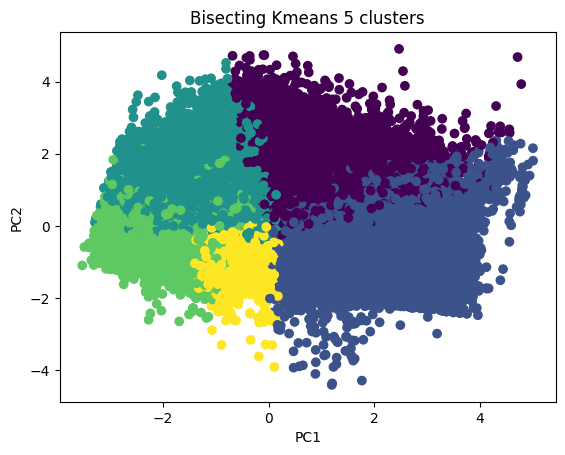
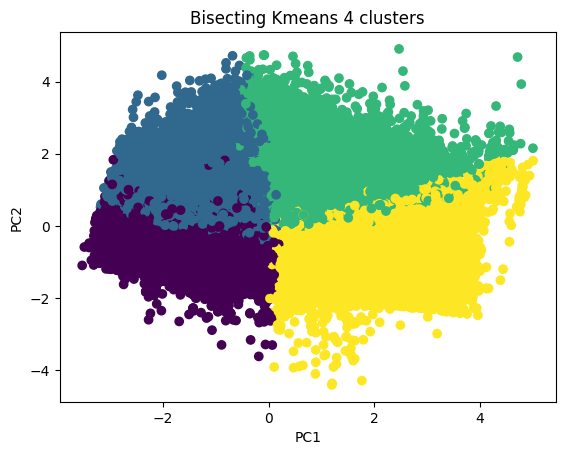
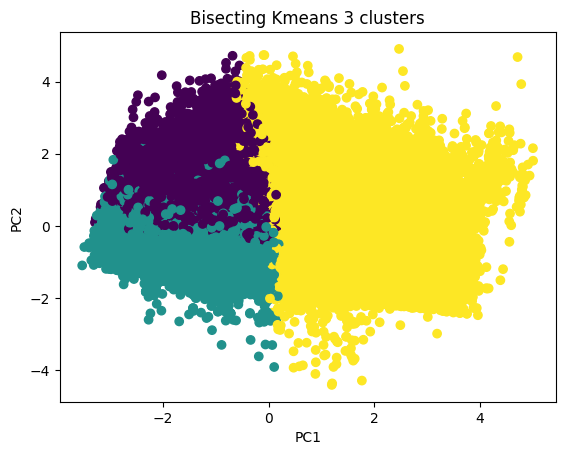
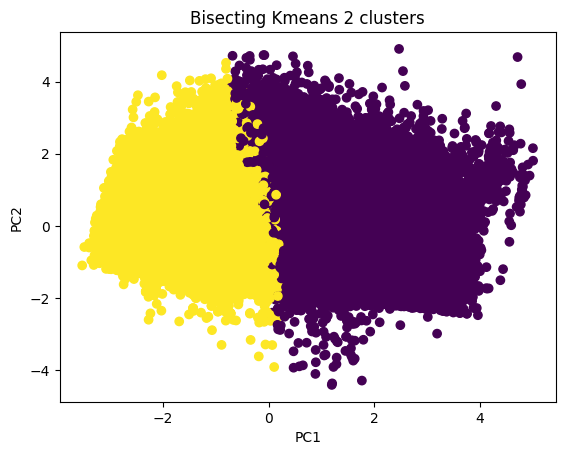
Kako bi mogli da vizualizujemo klasterovanje prvo ćemo primeniti PCA nad celim X skupom svesti skup na 2 atributa. Prvi algoritam za klasterovanje koji primenjujemo je Bisecting Kmeans, čije centre ćemo kasnije iskoristiti za običan Kmeans algoritam. Međutim imamo problem, jer nam grafici izgledaju ovako (Silhoutte score i SSE nam ne idu u korist takođe):

 Slika 20: Bisecting Kmeans greška

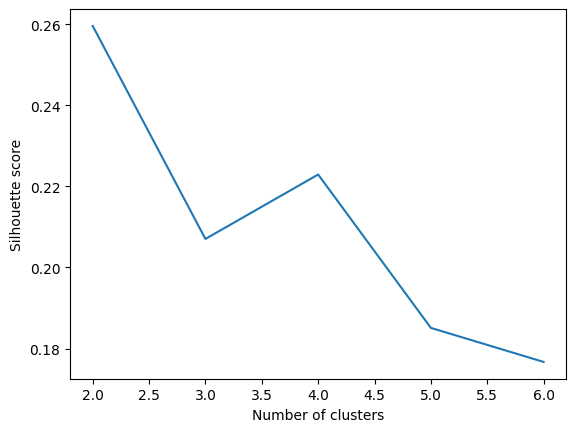
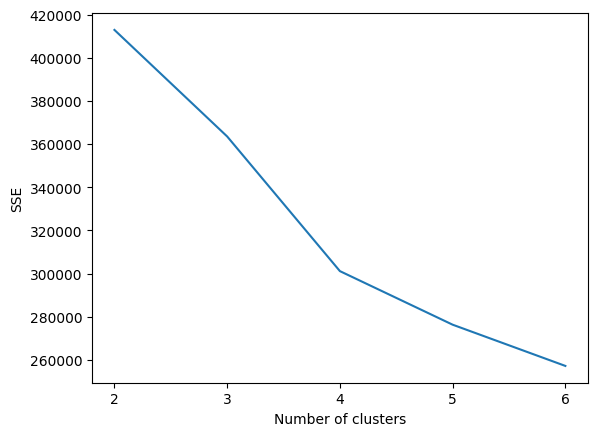


Slika 21: Bisecting Kmeans silhoutte score i SSE – greška

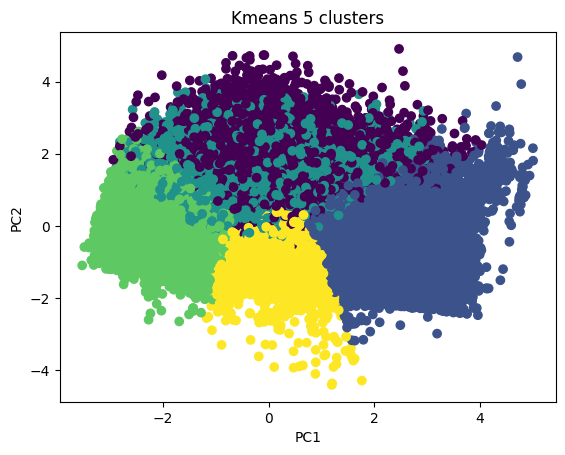
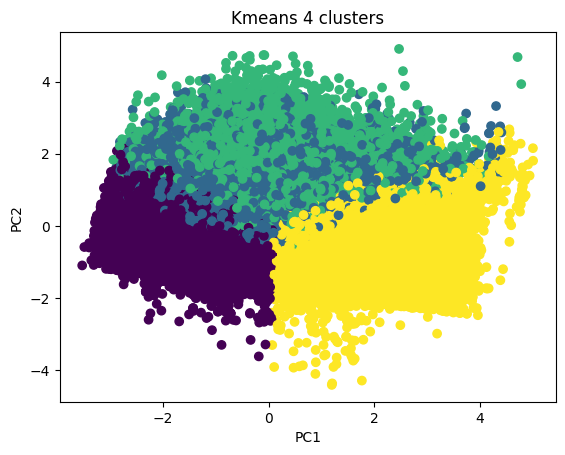
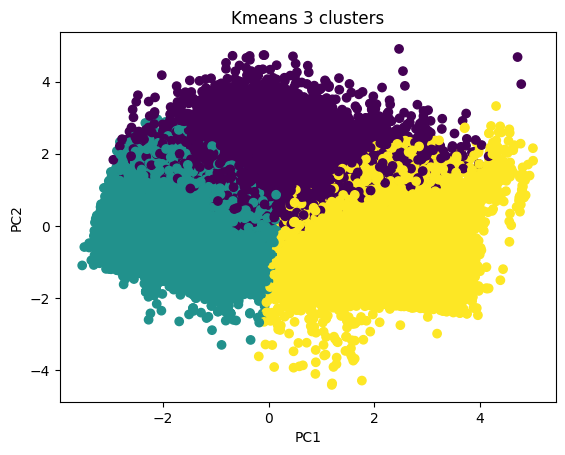
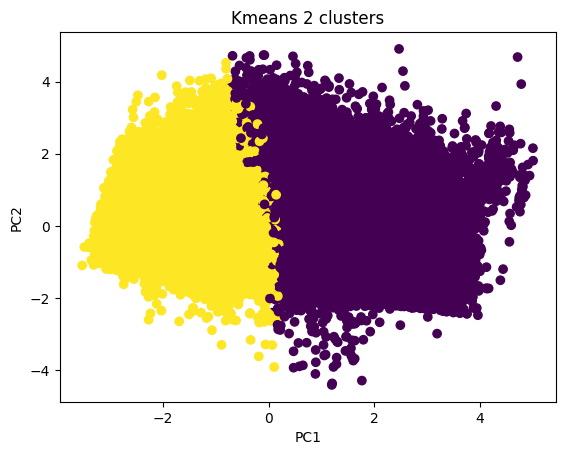
Kako bi ovo ispravili PCA primenjujemo samo na 4 kolone iz X skupa, i to 4 numeričke koje su imale najveći feature\_importance u RandomForrest algoritmu.

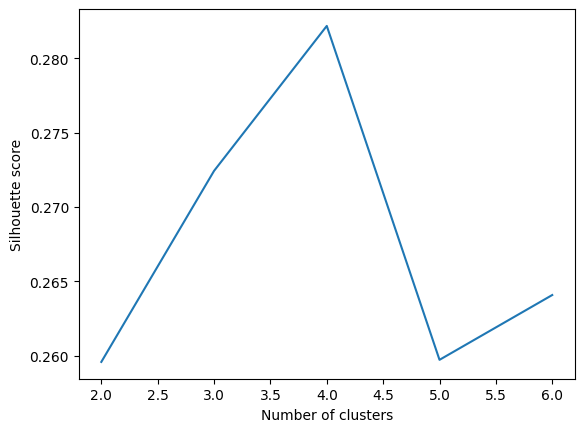
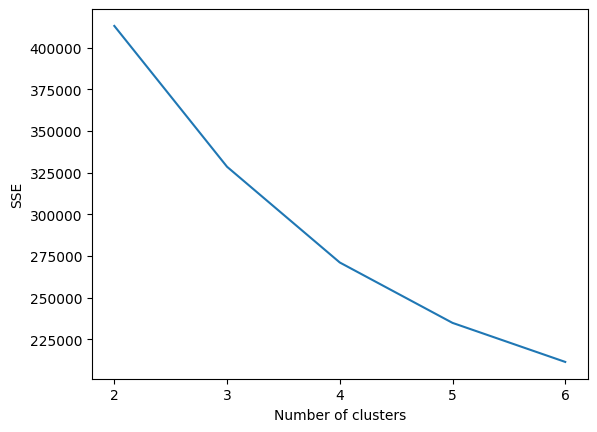
 Slika 22: Bisecting Kmeans

Sillhoute score i SSE nam sugerišu da je najbolje podeliti na 4 klastera, što je suprotno od onoga što mi znamo. Probajmo sad Kmeans.

 Slika 23: Bisecting Kmeans – Silhoutte score i SSE

Kmeans algoritam nam sugerise isto, još jasnije.

 Slika 24: Kmeans

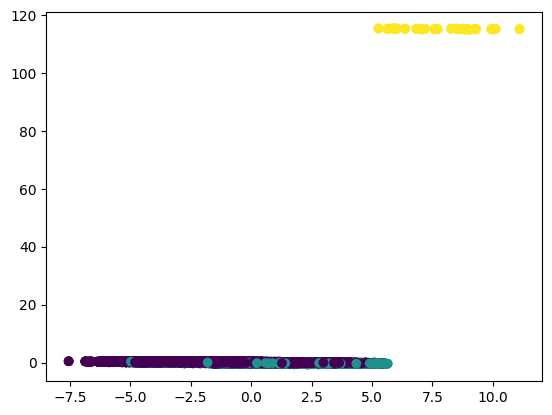
 Slika 25: Kmeans SSE i Silhoutte score

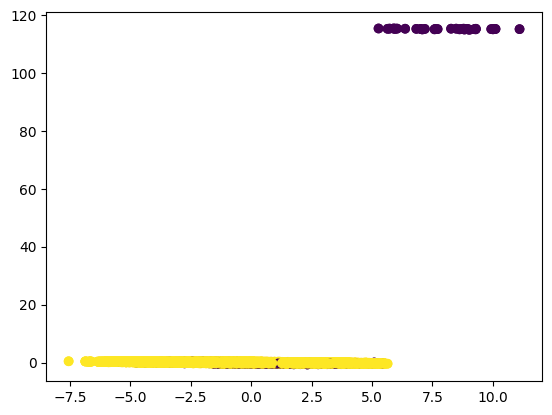
Dalje, koristimo Gaussian Mixture model. On koristi Gausovu raspodelu i parametar Pi za odredjivanje verovatnoce da odredjena tacka pripada odredjenom klasteru. Gausova raspodela je parametrizovana prosekom i matricom kovarijanse za dati broj klastera, pa se zavreme treniranja ti parametri, kao I Pi parametar optimizuju.

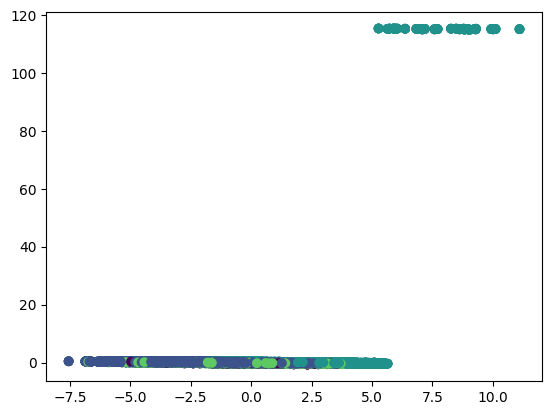


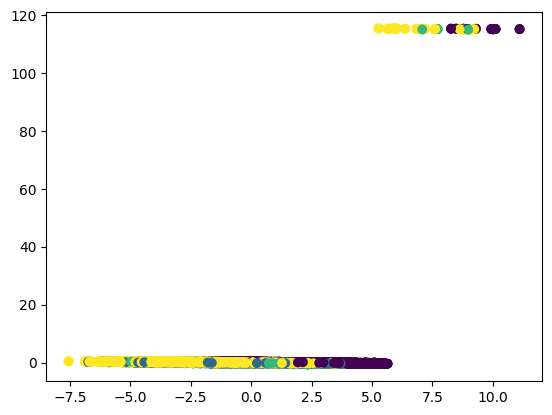
******

Slika 26: Formule za Gausovu distribuciju i Pi parametar

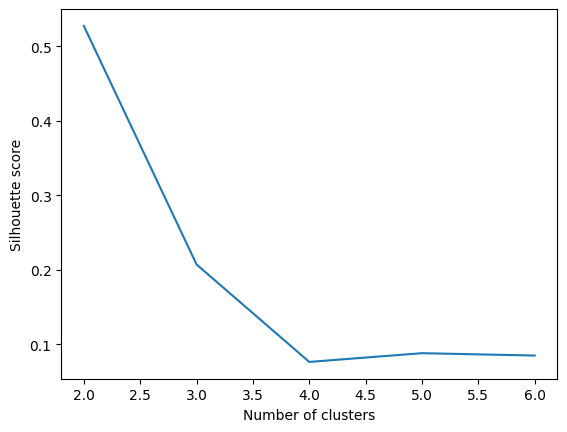




******

******

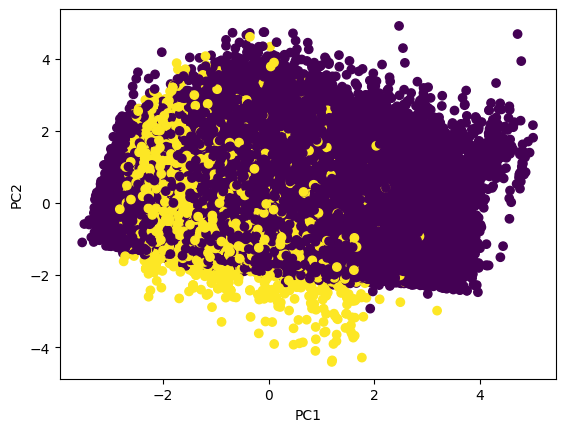
Slika 27: Gausian Mixture

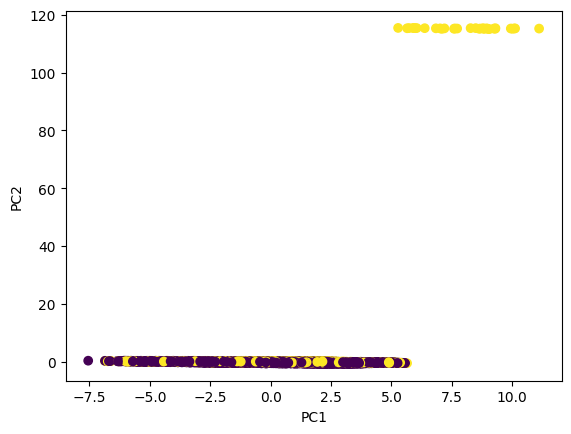
******

Slika 28: Silhoutte score za Gausian Mixture

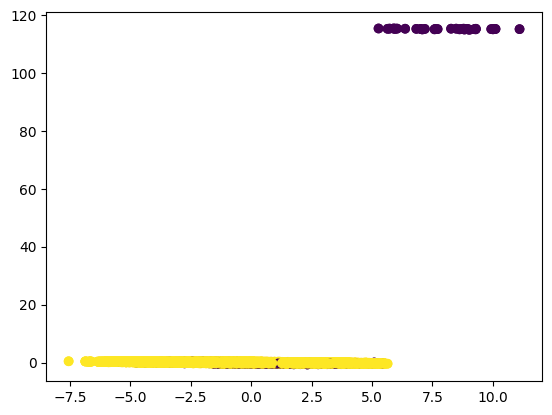
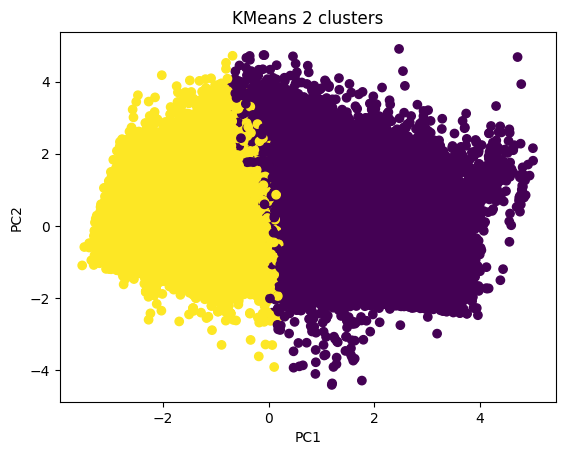
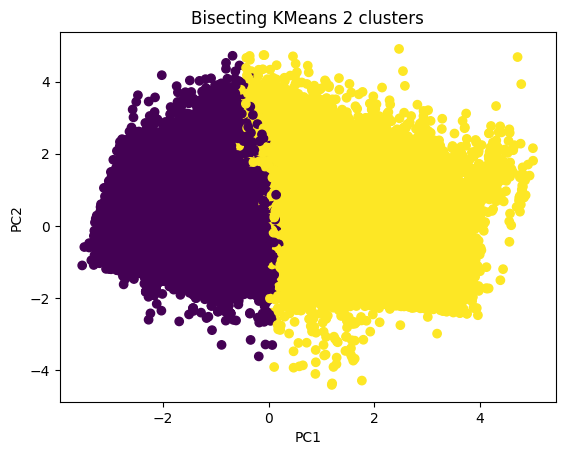
***Poređenje modela***

Kako klaster stvarnih podataka ne liči na naše klastere 2, ne možemo da očekujemo mnogo dobre rezultate. Dobijamo tačnost od ~65%, odnosno ~73%, što nam ne znači mnogo s obzirom da je jedna klasa otprilike duplo veća od druge. Očekivano je da klasterovanje ne daje dobre rezultate jer je u pitanju skup za klasifikaciju





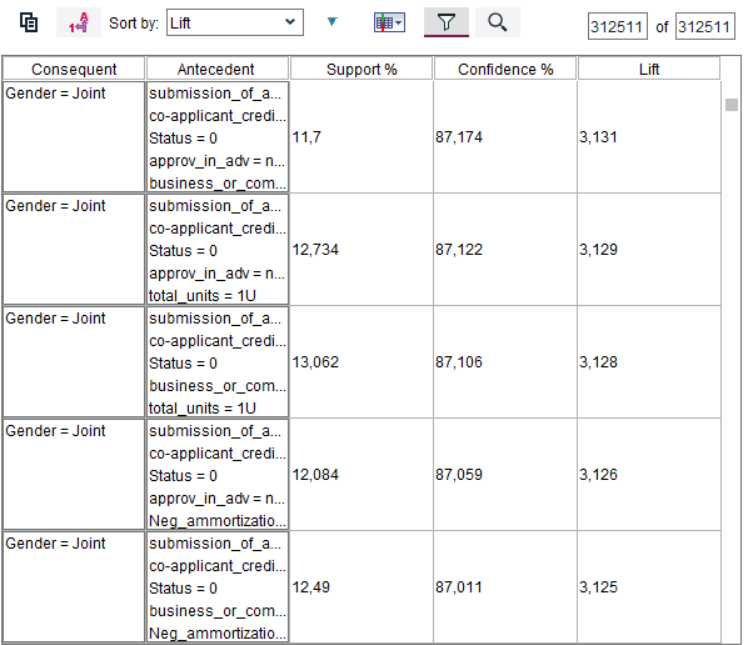
Slika 29: Stvarno podaci - klasteri

******

Slika 30: Pogadjanje kod klasterovanja

***Pravila pridruživanja***

Od pravila pridruživanja ne možemo mnogo očekivati, s obzirom da nam značenje kolona nije objašnjeno, a pritom je u pitanju skup za klasifikaciju. Najbolje što možemo dobiti je šta to sve povlači zajednički kredit. Koristli smo Apriori algoritam.

 Slika 31: Pravila pridruživanja

***Zaključak***

Predviđanje da li će neko moći da otplati kredit ne može biti perfektno, međutim možemo napraviti veoma dobro predviđanje. Ljudska priroda, koliko god nepredvidiva, otplaćivanje kerdita je prioritet gotovo svima, što nama omogućava pravljenje dobrog modela.