„Online Shoppers Purchasing Intention“

Zoran Vujičić

Avgust, 2023

**Seminarski rad u okviru kursa Istraživanje podataka 1 na**

**Matematičkom fakultetu**

**Sadržaj**

[​ 1 Uvod 2](#__RefHeading___Toc1448_3102450165)

[​ 1.1 Analiza skupa podataka 2](#__RefHeading___Toc1450_3102450165)

[​ 2 Preprocesiranje podataka 5](#__RefHeading___Toc1452_3102450165)

[​ 2.1 Rad sa nedostajućim vrednostima 5](#__RefHeading___Toc1454_3102450165)

[​ 2.2 Odabir atributa 5](#__RefHeading___Toc1605_3102450165)

[​ 2.3 Transformisanje kategoričkih atributa 6](#__RefHeading___Toc1456_3102450165)

[​ 2.4 Rad sa elementima izvan granica 7](#__RefHeading___Toc1458_3102450165)

[​ 2.5 Standardizacija 8](#__RefHeading___Toc1460_3102450165)

[​ 3 Klasifikacija 8](#__RefHeading___Toc1462_3102450165)

[​ 3.1 Nasumične šume 8](#__RefHeading___Toc1464_3102450165)

[​ 3.2 Stabla odlučivanja 11](#__RefHeading___Toc1466_3102450165)

[​ 3.3 Logistička regresija 12](#__RefHeading___Toc1468_3102450165)

[​ 3.4 Metoda potpornih vektora 13](#__RefHeading___Toc1470_3102450165)

[3.4.1 Linearni SVM 13](#__RefHeading___Toc1472_3102450165)

[3.4.2 SVM sa kernelom 14](#__RefHeading___Toc1474_3102450165)

[​ 3.5 Poređenje modela klasifikacije 15](#__RefHeading___Toc1476_3102450165)

[​ 4 Klasterovanje 16](#__RefHeading___Toc1478_3102450165)

[​ 4.1 Algoritam K-sredina 16](#__RefHeading___Toc1480_3102450165)

[​ 4.2 Sakupljajuće klasterovanje 18](#__RefHeading___Toc1482_3102450165)

[​ 4.3 Poređenje modela klasterovanja 20](#__RefHeading___Toc2391_1493345932)

[​ 5 Pravila pridruživanja 21](#__RefHeading___Toc1486_3102450165)

[​ 5.1 Apriori 21](#__RefHeading___Toc1488_3102450165)

[​ 6 Zaključak 24](#__RefHeading___Toc2393_1493345932)

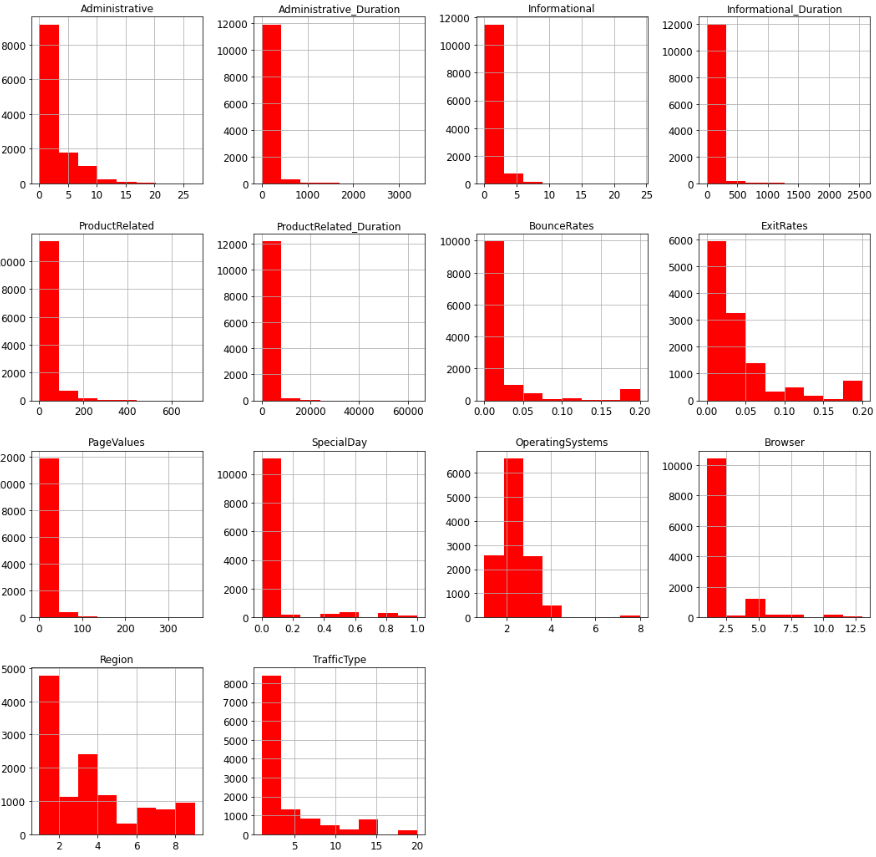
# 1 Uvod

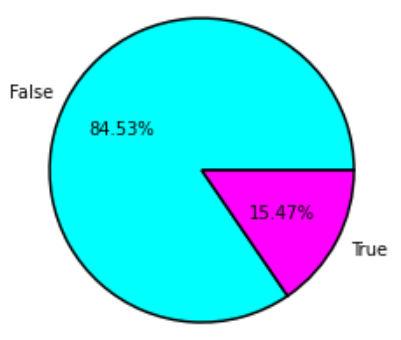
Skup podataka Online Shoppers Purchasing Intention sastavljen je od informacija o aktivnostima korisnika na internetu a podaci su prikupljeni sa sajta za maloprodaju. Konstruisan od strane *Google Analytics[1](https://www.techtarget.com/searchbusinessanalytics/definition/Google-Analytics" \l ":~:text=How does Google Analytics work,of Google's data collection servers.)* servisa za sakupljanje statističkih podataka o aktivnostima korisnika na internetu, skup se sastoji od 12330 pristupa korisnika. Skup je pažljivo formiran tako da svaki pristup odgovara različitom korisniku u periodu od jedne godine, kako bi se izbegla sklonost ka specifičnom profilu korisnika, danu ili periodu.

## 1.1 Analiza skupa podataka

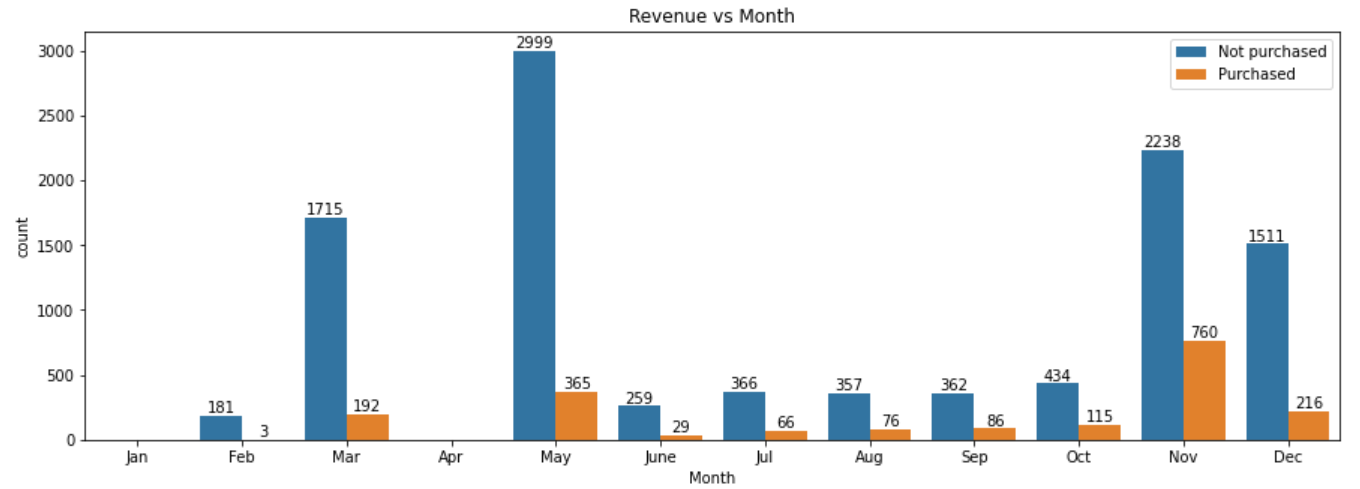
Naš skup podataka ima ukupno 18 atributa. U sledećoj tabeli si ukratko opisani svi atributi, prvih 10 su numerički a ostalih 8 su kategorički atributi. Na slici 1 su prikazani histogrami raspodele nekih atributa.

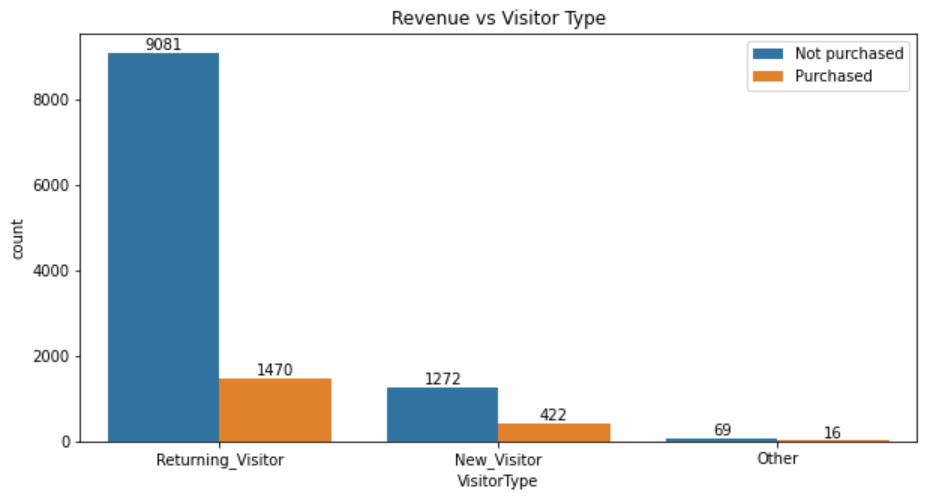
|  |  |
| --- | --- |
| Atributi | Opis |
| Administrative | Broj posećenih veb strana vezanih za upravljanje profilom |
| Administrative\_Duration | Vreme provedeno na veb stranama o upravljanju profilom u sekundama |
| Informational | Broj posećenih veb strana vezanih za informacije o sajtu |
| Informational\_Duration | Vreme provedeno na stranama za informacije u sekundama |
| ProductRelated | Broj posećenih veb strana vezanih za proizvode |
| ProductRelated\_Duration | Vreme provedeno na veb stranama vezanim za proizvode  u sekundama |
| BounceRates | Procenat korisnika koji nakon ulaska na veb sajt izađu bez pokretanja drugih zahteva ka serveru |
| ExitRates | Koliko je puta u procentima veb strana bila poslednja u jednom pristupu korisnika internetu, u odnosu na ukupan broj pregleda |
| PageValues | Predstvalja prosečnu vrednost veb stranica koje je korisnik posetio pre nego što je izvršio transakciju |
| SpecialDay | Pokazuje koliko je vreme posete veb sajtu blizu nekog specijalnog dana u godini (npr. 8. Mart), u kojima je veća verovatnoća da se uspešno izvrši transakcija |
| Month | Mesec u godini u kome je korisnik pristupio veb sajtu |
| OperatingSystems | Operativni sistem koji je koristio korisnik |
| Browser | Internet pregledač koji je koristio korisnik |
| Region | Geografski region iz kog se prijavio korisnik |
| TrafficType | Izvor, odakle je korisnik pristupio veb sajtu |
| VisitorType | Tip korisnika koji može biti *Novi Korisnik*, *Povratnik* i *Ostali* |
| Weekend | Pokazauje da li je datum posete vikend ili ne |
| Revenue | Pokazuje da li je korisnik pri poseti veb sajtu ivršio transakciju ili nije |

*Slika 1: Histogrami raspodele nekih atributa*

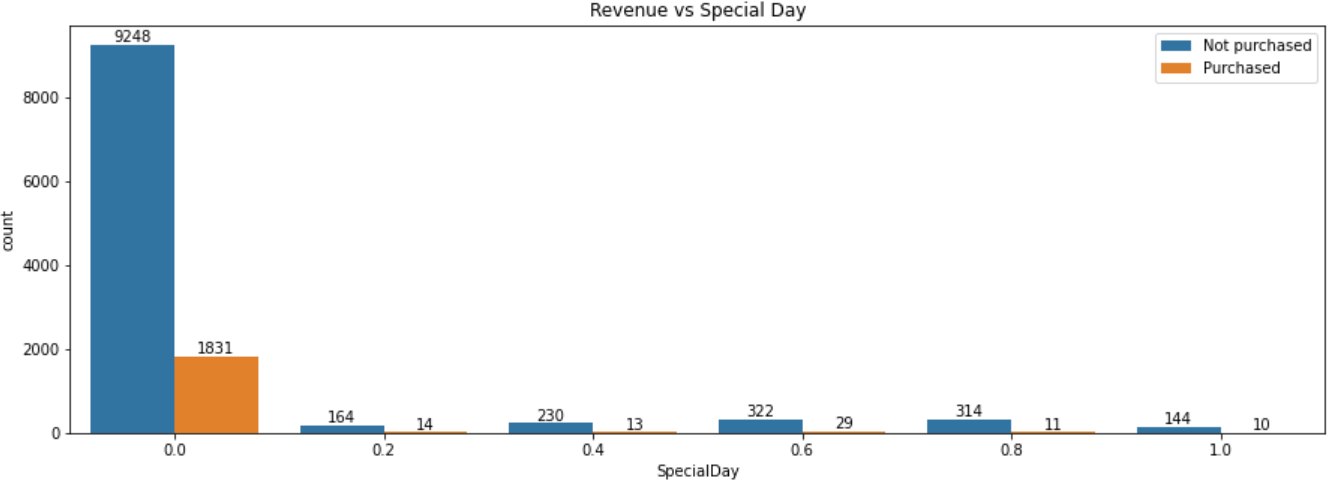
Ciljna promenljiva u procesu klasifikacije biće **Revenue**, odnosno da li je korisnik kupio neki proizvod ili nije, jer je to informacija koju želimo da dobijemo nakon istraživanja. Atrubut Revenue ima 2 klase a to su True i False. Broj instanci klase True je 1908 a klase False 10422. Dakle klase nisu balansirane što će predstavljati problem u procesu klasifikacije. To je i prikazano na slici 2. Odnos atributa Revenue sa nekim atributima prikazan je na slici 3.1, 3.2, 3.3 i 3.4.

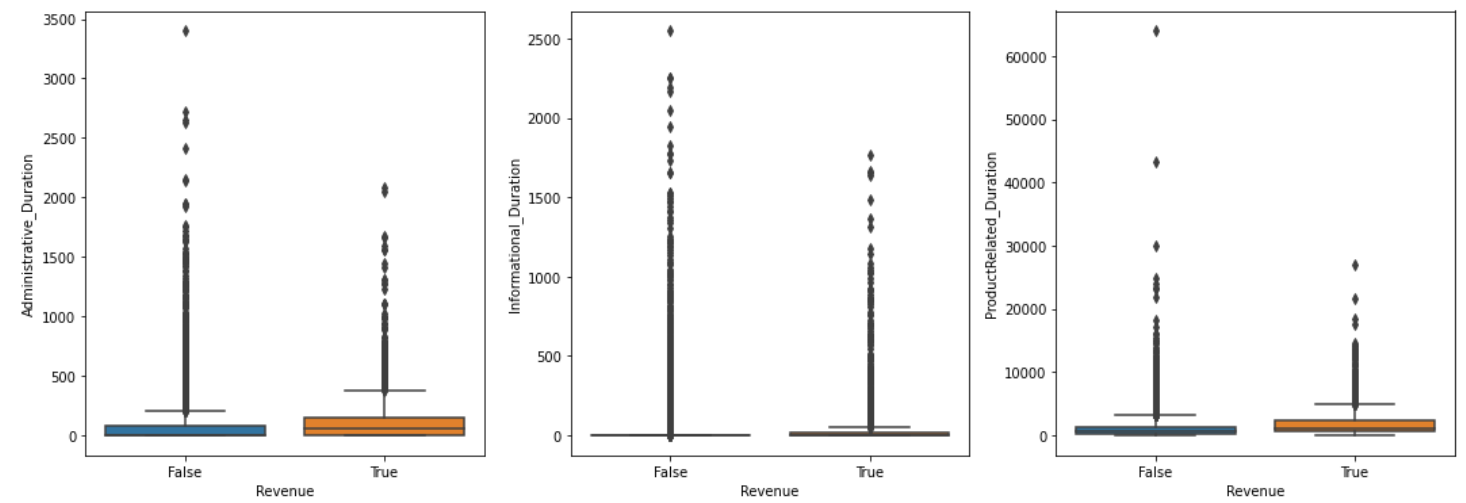
*Slika 2: Procentualni prikaz raspodele klasa*

 *Slika 3.1: Odnos atributa Revenue i Month*



*Slika 3.2: Odnos atributa Revenue i Visitor Type*

 *Slika 3.3: Odnos atributa Revenue i Special Day*



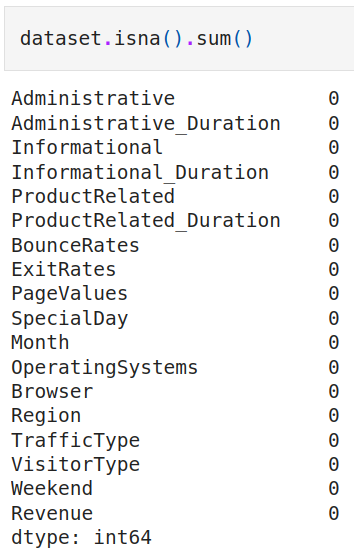
*Slika 3.4: Odnos atributa Revenue i atributa vezanih za vreme provedeno na stranicama*

Vidimo da ćemo imati i elemenata van granica *(outlier)*, što ćemo istražiti u predprocesiranju.

# 2 Preprocesiranje podataka

## 2.1 Rad sa nedostajućim vrednostima

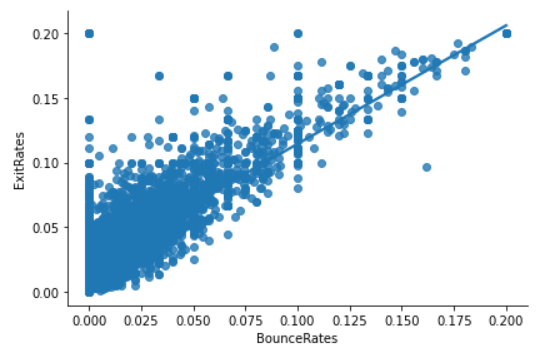
Proveravamo da li u našem skupu podataka postoje nedostajuće vrednosti i ako postoje treba ih na adekvatan način eliminisati ili zameniti.



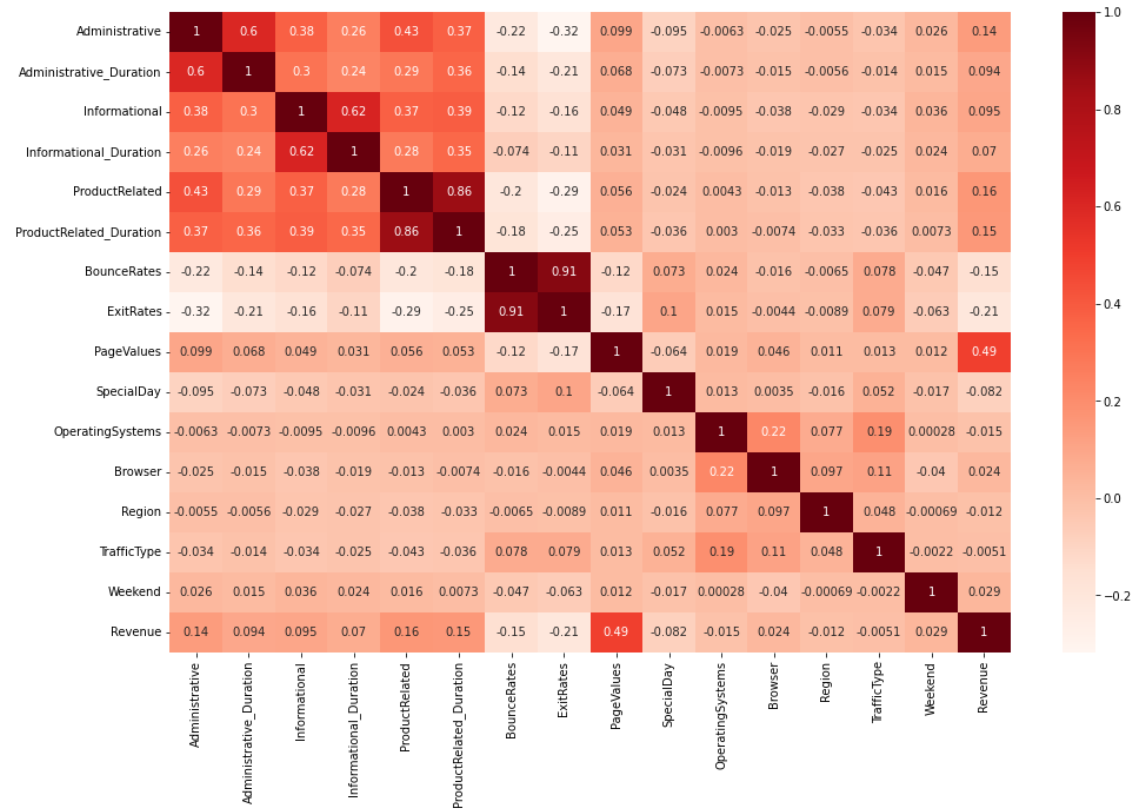
Dakle, iz rezultata izvršavanja možemo videti da nema nedostajućih vrednosti.

## 2.2 Odabir atributa

Koristićemo matricu korelacije pri odabiru atributa. Na osnovu matrice korelacije, koja je prikazana na slici 4 saznajemo koliko atributi utiču jedni na druge. Tamno crvenom bojom su označeni atributi sa visokom korelacijom. Na osnovu toga vidimo da su atributi *Product Related* i *Product Related Duration* vioko korelirani, tako da ćemo jedan izbaciti. Isto važi i za atribute *Exit Rates* i *Bounce Rates.*

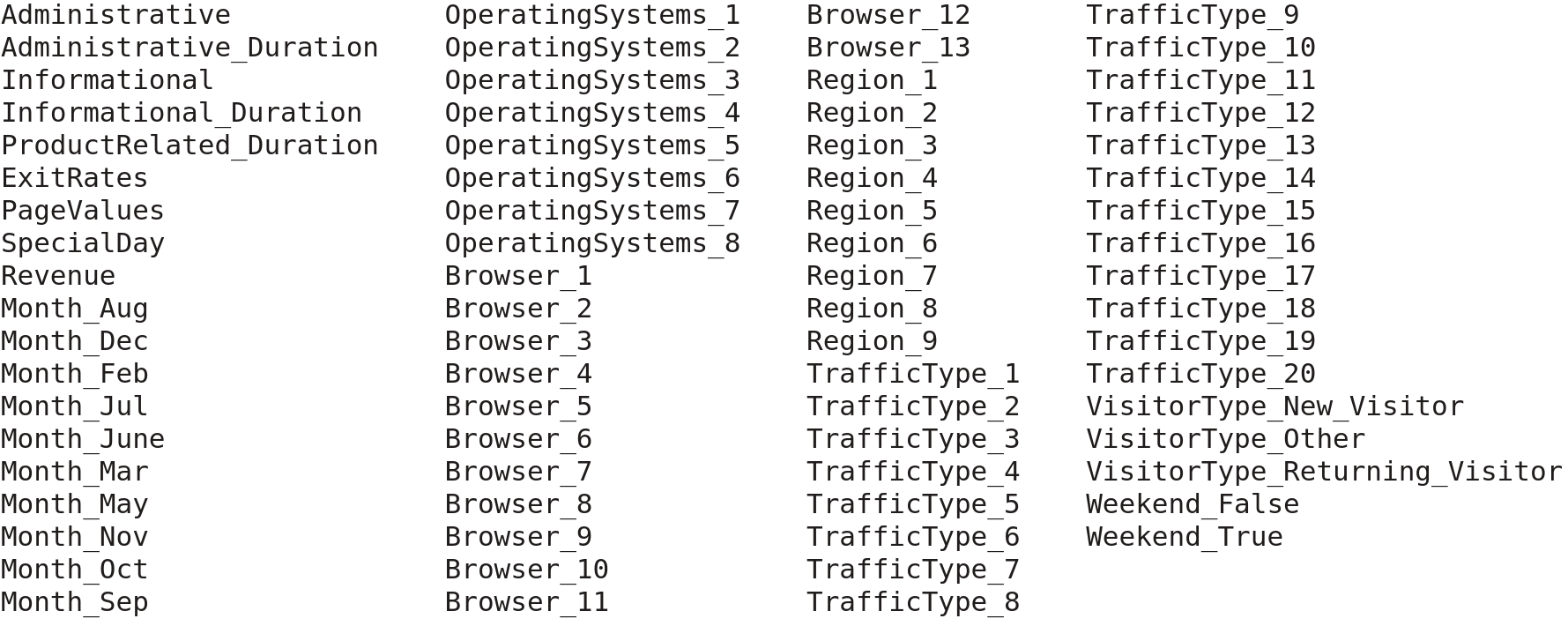


Slika 5: Odnos atrubuta *Exit Rates* i *Bounce Rates*

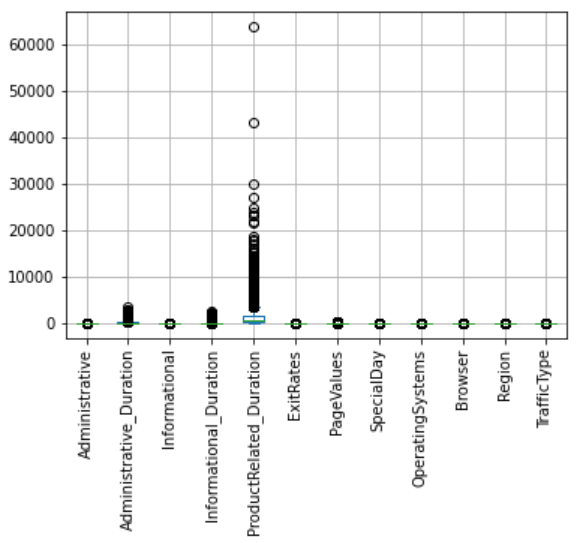


Slika 4: Matrica korelacije

## 2.3 Transformisanje kategoričkih atributa

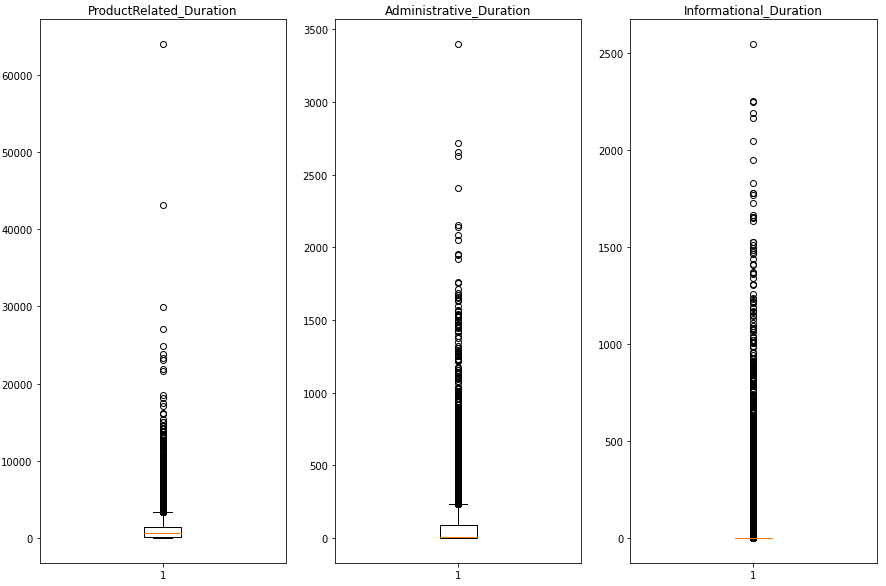
U našem skupu podataka postoje kategorički atributi koje ćemo transformisati zbog algoritama u procesu klasifikacije koja zahtevaju da oni budu numerički. Atributi mogu da uzimaju vrednosti iz diskretnog skupa, a to može da predstavlja problem pri izračunavanju ako je neki atribut označen većim brojem a nema nužno veću vrednost. Na primer atribut *Browser* kojiuzimavrednosti iz skupa {1, 2, … , 13}, dakle ne mora da znači da je neki veb pregledač „veći“ od nekog drugog. Zbog toga je potrebno izvršiti proces binarizacije *(binarization*[*2*](https://deepai.org/machine-learning-glossary-and-terms/binarization)*)*. Na osnovu kategoričkog atributa koji ima n različitih vrednosti stvoriće se n novih različitih binarnih atributa. Svaki binarni atribut odgovaraće jednoj mogućoj vrednosti kategoričkog atributa. Od n atributa koji se nalaze u jednom redu tačno jedan će imati vrednost 1, a ostali će imati vrednost 0. Binarni podaci su specijalni slučaj i numeričkih i kategoričkih atributa tako da binarni kategorički atributi koji imaju vrednosti *True* i*False* zamenjuju se sa 0 i 1. Ispod je spisak novonastalih atributa nakon transformacije.

## 2.4 Rad sa elementima izvan granica

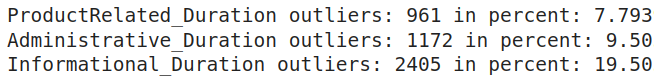


*Slika 6: Dijagram elemenata van granica*

Na slici 6 vidimo da atributi vezani za vreme provedeno na stranicama imaju izražene elemente izvan granica *(outliers*[*3*](https://www.javatpoint.com/what-is-outlier-in-data-mining)*)*. To su atributi *Administrative Duration, Informational Duration* i *Product Related Duration.* Izdvajamo ih na posban boxplot dijagram na slici broj 7 kako bi se jasnije i preciznije videli.

*Slika 7: Boxplot dijagram elemenata van granica*

Prvo pokušavamo sa metodom interkvantilnog opsega (*IQR*[*4*](https://www.geeksforgeeks.org/interquartile-range-to-detect-outliers-in-data/)). Međutim, ovom metodom bismo izgubili dosta infromacija što je i prikazano na slici 8.

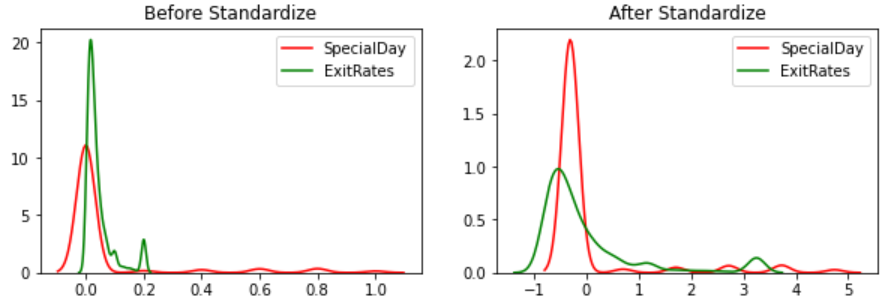


*Slika 8: Procentualni prikaz elemenata van granica koristeći IQR metodu*

Tako da će biti uklonjeni samo ekstremni elementi izvan granica. Granice bi bile 20000 za atribut *Product Related Duration,* 2400za *Administrative Duration* i 1900 za *Informational Duration.* Takođe negativne vredosti su primećene u kolonama veznaim za vreme provedeno na stranicama, pošto ove vrednosti ne mogu biti negativne zamenjene su sa 0.

## 2.5 Standardizacija

Pre standardizacije, skup se deli na skup atributa i na specijalni atribut koji će biti korišćen kao oznaka klase. Nakon toga se oba skupa dele na trening i test skup koji će biti korišćeni u procesu klasifikacije. Pošto su atributi različito skalirani, to znači da ih ne možemo međusobno upoređivati. Zbog toga se vrši standardizacija *(standardization*[*5*](https://builtin.com/data-science/when-and-why-standardize-your-data)*)* koja funkcioniše tako što se od atributa oduzme njegova srednja vrednost i to se podeli njegovom standardnom devijacijom. Prikaz raspodele atributa pre i posle standardizacije je na slici 9.



*Slika 9: Prikaz raspodele atributa pre i posle standardizacije*

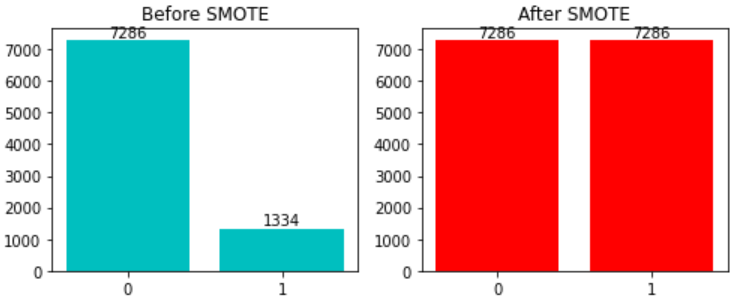
# 3 Klasifikacija

## 3.1 Nasumične šume

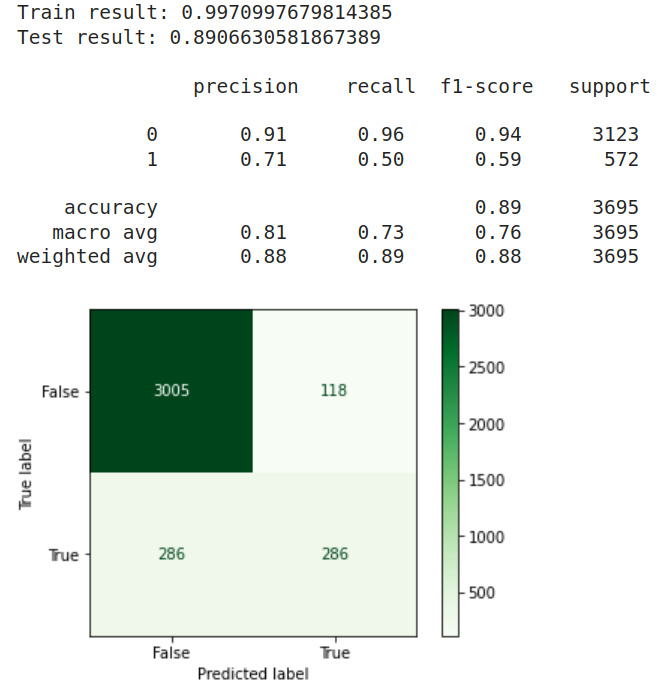
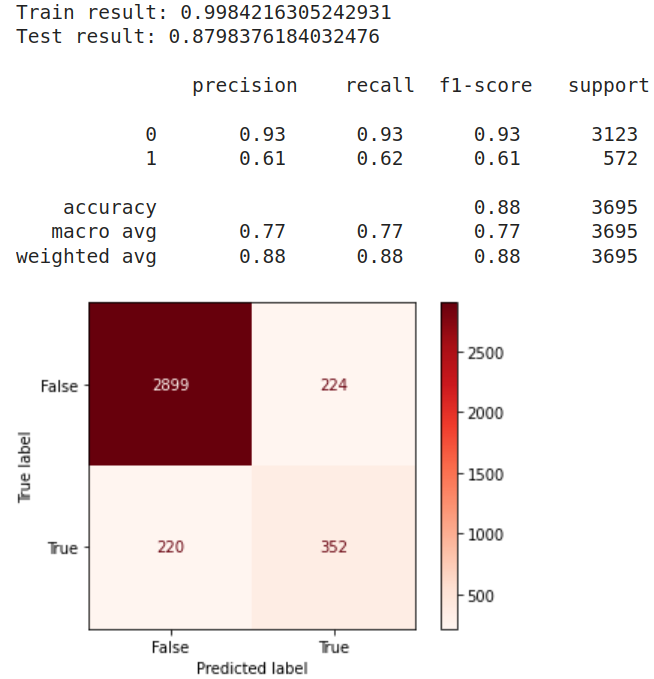
Nasumične šume *(Random Forest*[*6*](https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/)*)* algoritam radi tako što sagrađuje mnoštvo stabala odlučivanja pri treniranju i dodeljuje instanci onu klasu koja se najčešće pojavljivala. Ovo je algoritam koji spada u grupu ansabala koji koriste više algoritama za učenje kako bi postigli što bolje rezultate u predikciji klasa.

Što se tiče odabira parametara, broj stabala u šumi biće jednak 15 a za kriterijume podele biće korišćena Entropija. To su parametri koji su dobijeni pomoći *Grid Search[6](https://www.mygreatlearning.com/blog/gridsearchcv/" \l ":~:text=GridSearchCV is a technique for,parameter values%2C predictions are made.)* algoritma.

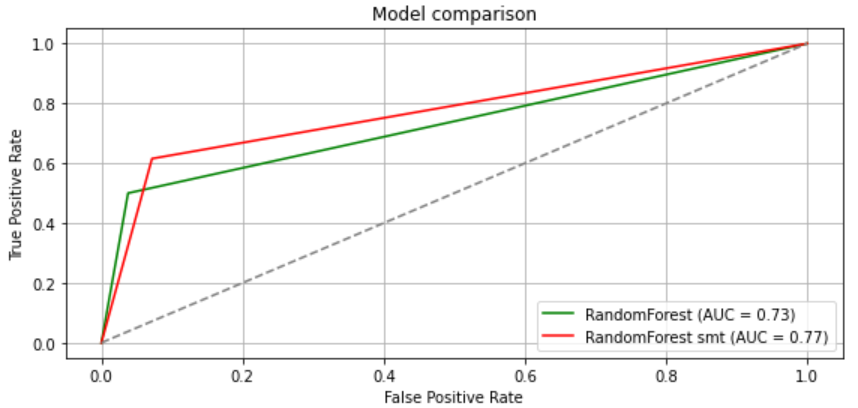
Prvo ćemo pokušati bez balansiranja klasa, a posle ćemo pokušati da popravimo performanse korišćenjem SMOTE *(Synthetic Minority Oversampling Technique*[*7*](https://towardsdatascience.com/smote-fdce2f605729)*)* tehnike. Ona radi tako što na slučajan način odabere tačku iz manje klase i računa k najbližih suseda za tu tačku. Tačke koje su sintetisane se dodaju između odabrane tačke i njenih suseda. Rezultati pre i posle balansiranja klasa prikazani su na slici 11, a odgovarajuće matrice konfizije sa rezultatima pre balansiranja (zelena) i posle balansiranja (crvena) na slici 12 i 13. Takođe je prikazano i poređenje ROC krive i AUC rezultata pre i posle balansiranja na slici 14. Na slici 15.1 prikazana je značajnost atribura.



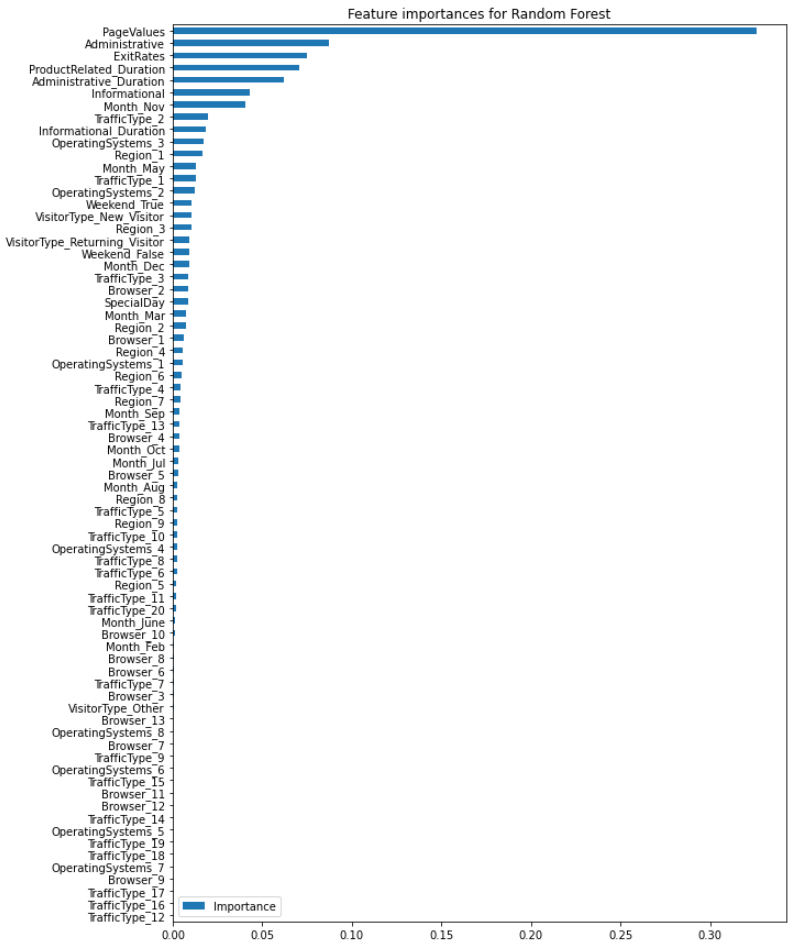
*Slika 11: Rezultati pre i posle balansiranja klasa korišćenjem SMOTE tehnike*



*Slika 12: Matrica konfuzije pre balansiranja Slika 13: Matrica konfuzije posle balansiranja*



*Slika 15: Poređenje ROC krive i AUC rezltata pre i posle balansiranja klasa*

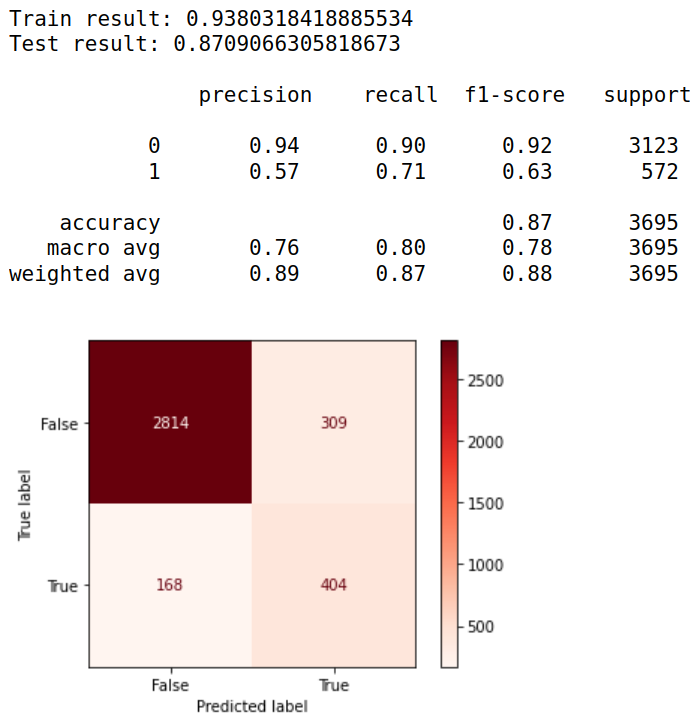
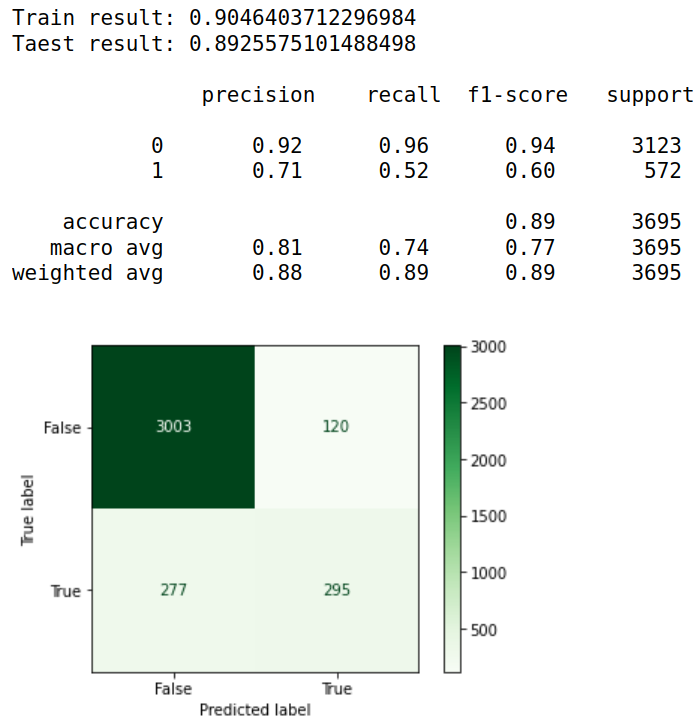
**

*Slika 15.1: Značajnost atributa*

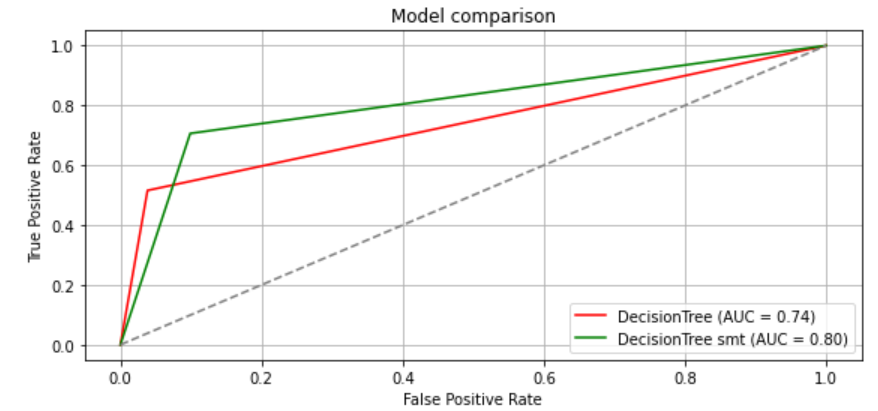
## 3.2 Stabla odlučivanja

Naredni algoritam koji ćemo primeniti na naš skup su Stabla odlučivanja *(Decision Trees*[*8*](https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm)*)*. To je algoritam u kome se proces klasifikacije modeluje pomoću skupa hijerarhijskih odluka koje su donete na osnovu atributa trening podataka čija je struktura uređena u obliku drveta.

Što se tiče parametara, maksimala dubina čvorova će biti 5 pre balansiranja i 10 posle balansiranja. Za kriterijume podele biće korišćena Entropija. Kao i kod nasumičnih šuma, ovi parametri su dobijeni pomoću *Grid Search* algoritma. Na slici 16 i 17 su prikazane odgovarajuće matrice konfuzije sa rezultatima pre i posle balansiranja, a na slici 18 poređenje ROC kirive i AUC rezultata pre i posle balansiranja.



*Slika 16: Matrica konfuzije pre balansiranja Slika 17: Matrica konfuzije posle balansiranja*

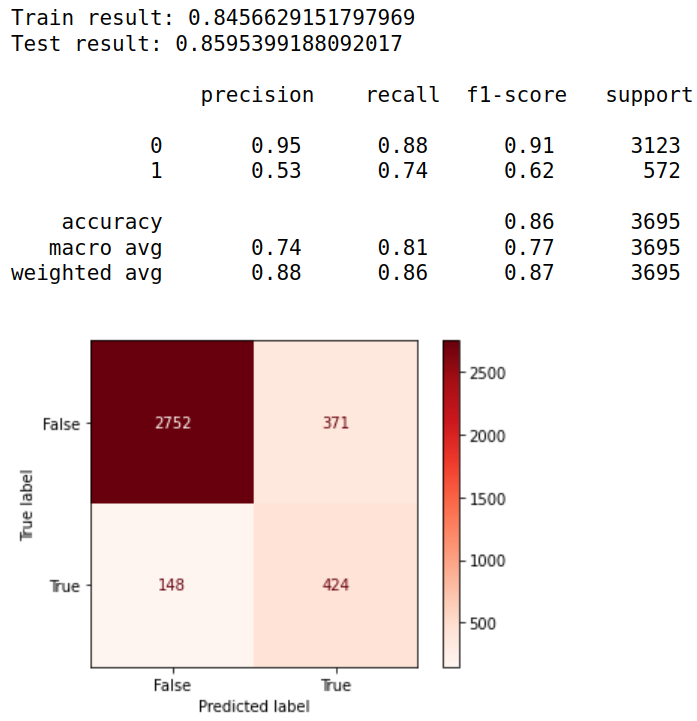
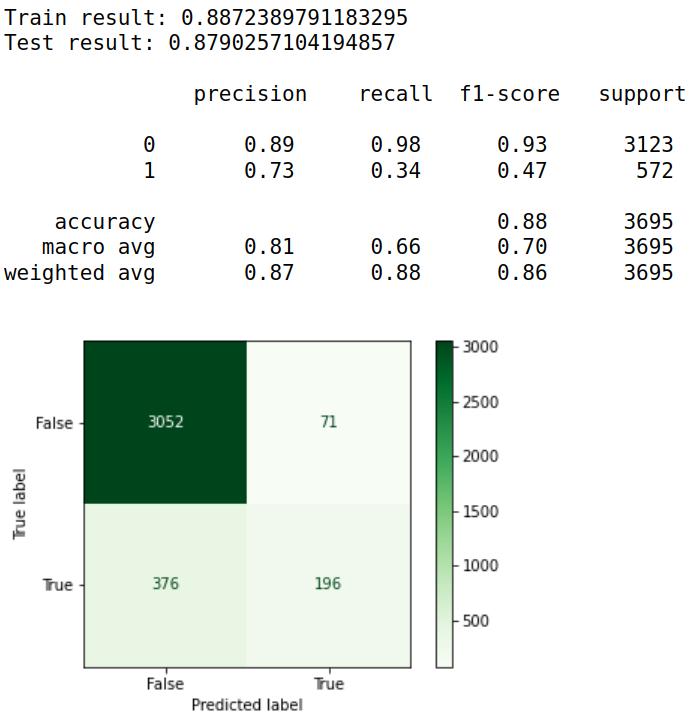


*Slika 18: Poređenje ROC krive i AUC rezltata pre i posle balansiranja klasa*

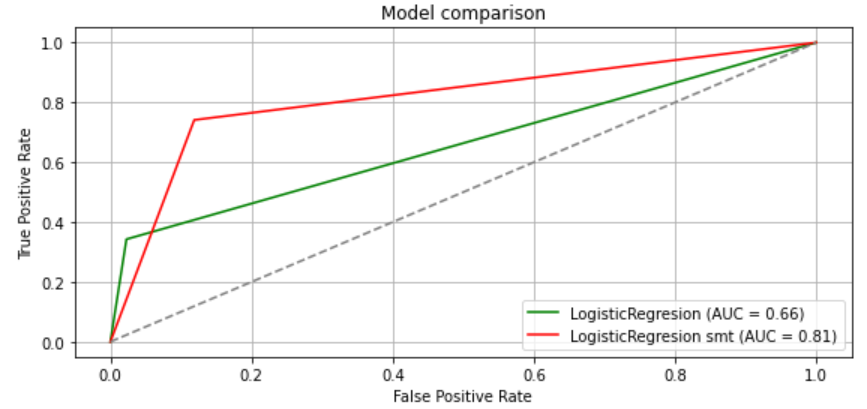
## 3.3 Logistička regresija

Sledeći što ćemo pokušati je Logistička regresija *(Logistic Regression*[*9*](https://www.geeksforgeeks.org/understanding-logistic-regression/)*)*. Ona je jedna od najkorišćenijih metoda te se zato odlučujemo za nju, vrlo je jednostavna i pruža efikasno treniranje. Takođe nam odgovara zato što je upotrebljiva samo na binarnu klasifikaciju, a to jeste naš slučaj.

Što se tiče parametara, kao i u prethodim metodama korišćen je *Grid Search* algoritam. Na slici 19 i 20 su prikazane odgovarajuće matrice konfuzije sa rezultatima pre i posle balansiranja, a na slici 21 poređenja ROC kirive i AUC rezultata pre i posle balansiranja.



*Slika 19: Matrica konfuzije pre balansiranja Slika 20: Matrica konfuzije posle balansiranja*



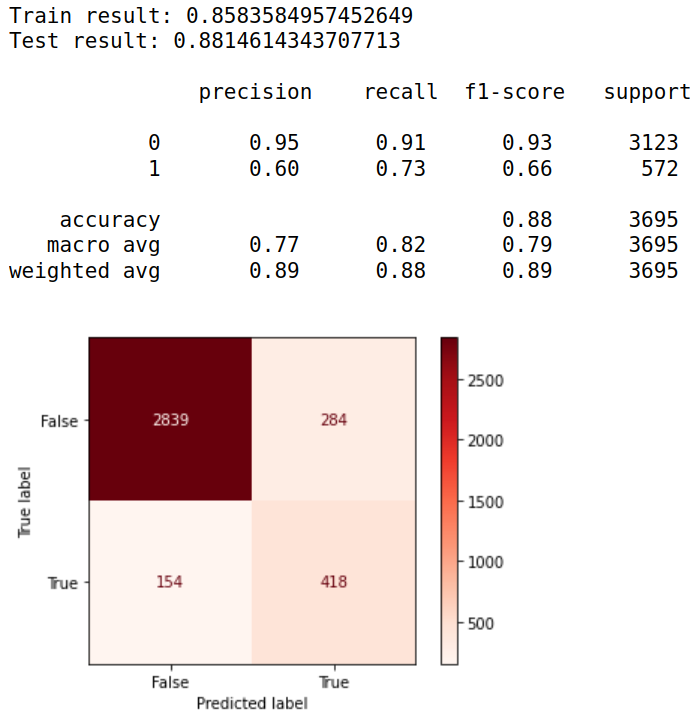
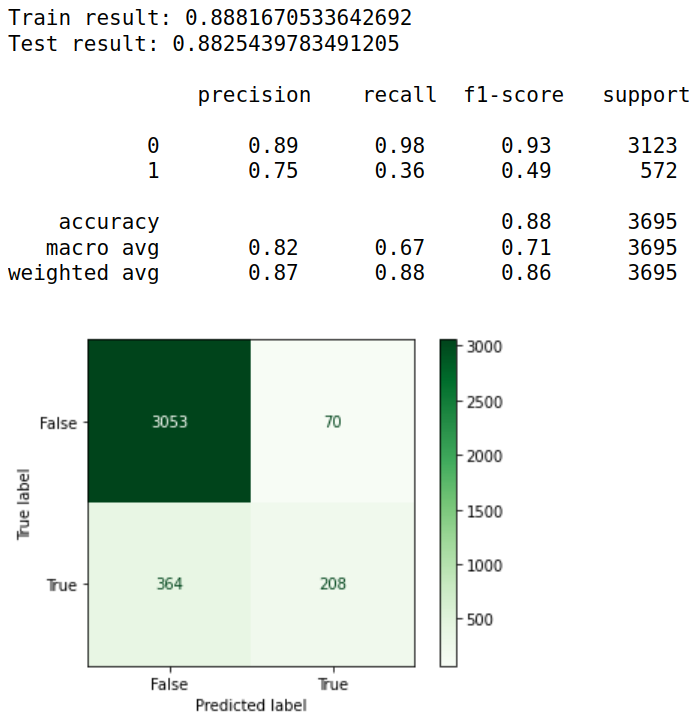
*Slika 21: Poređenje ROC krive i AUC rezltata pre i posle balansiranja klasa*

## 3.4 Metoda potpornih vektora

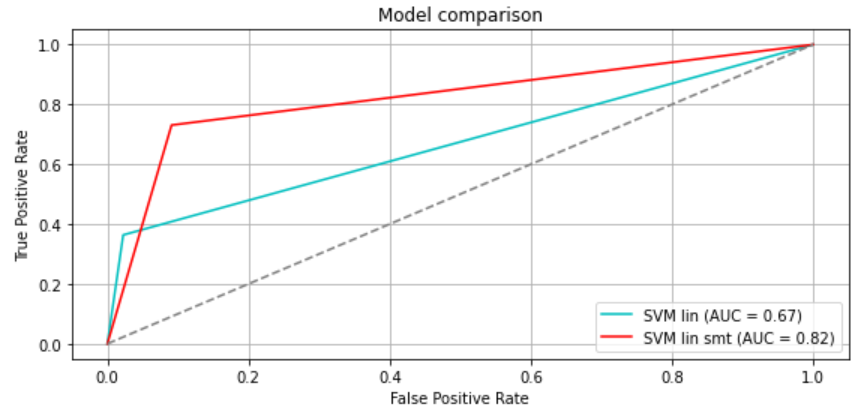
Poslednja metoda koju ćemo koristiti je SVM *(Support Vector Machine*[*10*](https://www.analyticsvidhya.com/blog/2021/10/support-vector-machinessvm-a-complete-guide-for-beginners/)*).* To je metoda koja je zasnovana na ideji vektorskih prostora. Prvo ćemo primeniti linearni SVM, a onda i SVM sa kernel funkcijom.

### 3.4.1 Linearni SVM

Što se tiče parametara biće izabrana vrednost 1.0 koja je dobijena na osnovu *Grid Search* algoritma. Na slici 22 i 23 su prikazane odgovarajuće matrice konfuzije sa rezultatima pre i posle balansiranja, a na slici 24 poređenja ROC kirive i AUC rezultata pre i posle balansiranja.



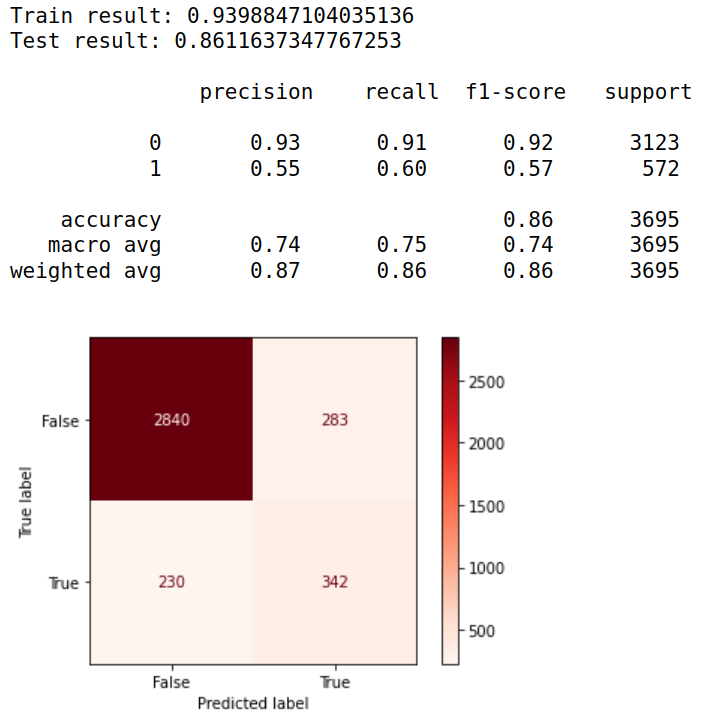
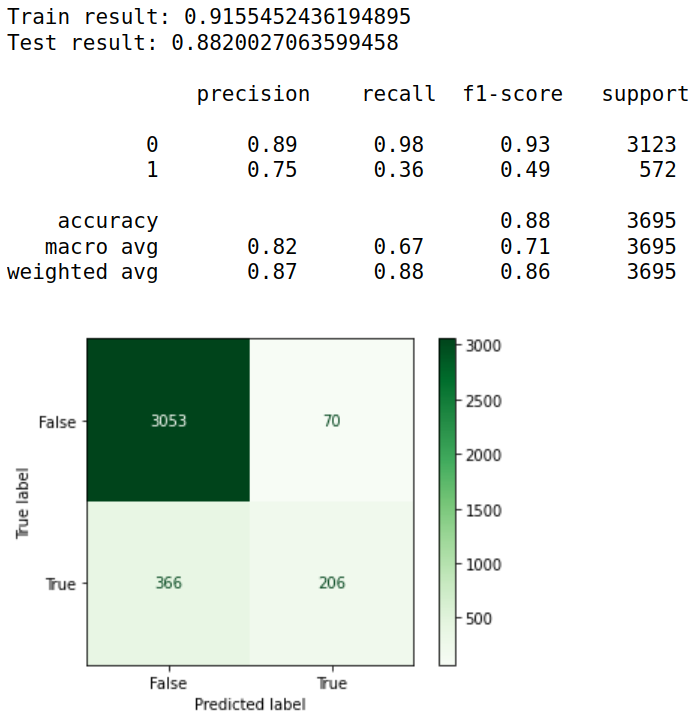
*Slika 22: Matrica konfuzije pre balansiranja Slika 23: Matrica konfuzije posle balansiranja*



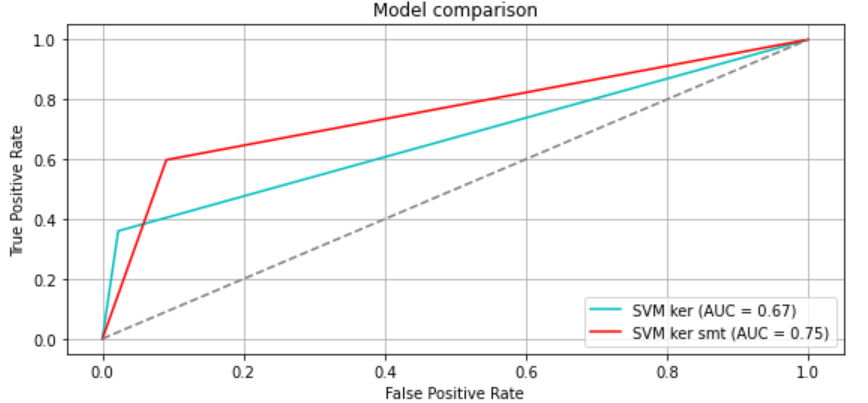
*Slika 24: Poređenje ROC krive i AUC rezltata pre i posle balansiranja klasa*

### 3.4.2 SVM sa kernelom

Za parametre biće izbarana vrednost 2.0, a za kernel je izabran rbf kernel. Oni su dobijeni na osnovu *Grid Search* algoritma. Na slici 25 i 26 su prikazane odgovarajuće matrice konfuzije sa rezultatima pre i posle balansiranja, a na slici 27 poređenja ROC kirive i AUC rezultata pre i posle balansiranja.



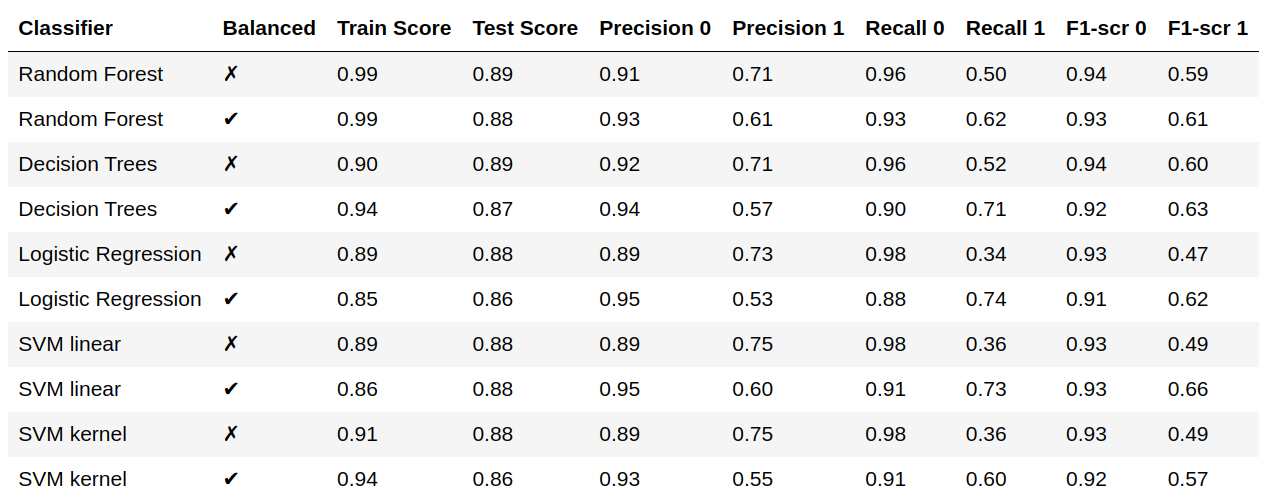
*Slika 25: Matrica konfuzije pre balansiranja Slika 26: Matrica konfuzije posle balansiranja*



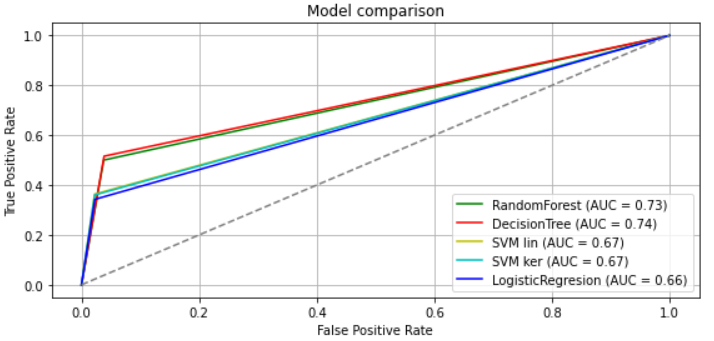
*Slika 27: Poređenje ROC krive i AUC rezltata pre i posle balansiranja klasa*

## 3.5 Poređenje modela klasifikacije

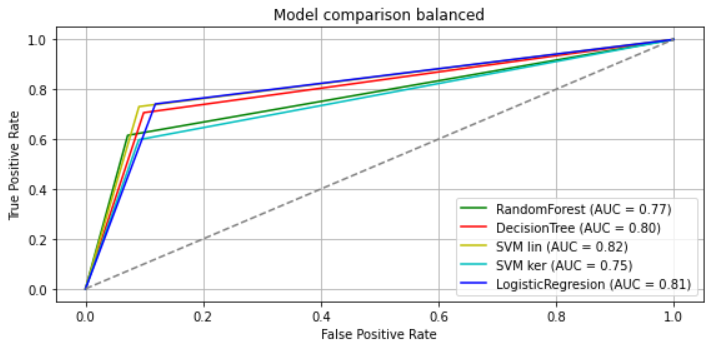
Svi korišćeni metodi se jako slično ponašaju. Na slici 25 je prikazano poređenje svih metoda. Na svaki metod balansirane je uticalo na isti način a to je da je se smanjila tačnost na test skupu ali se povećao odziv klase 1, što svakako znači da je model dobijen balansiranjem kalsa bolji. Takođe nakon balansiranja dobijamo bolje ROC i AUC rezultate. Na slici 26 prikazano je poređenje ROC krive i AUC rezultata pre balansiranja. Na njoj se vidi da najbolji rezultat daju stabla odlučivanja. Na slici 27 prikazano je poređenje ROC krive i AUC rezultata nakon balansiranja klasa, u tom slučaju iznenađujuće najbolji rezultat daje linearni SVM.



*Slika 28: Tablica poređenja rezultata svih metoda pre i posle balansiranja*



*Slika 29: Poređenje ROC krive i AUC rezultata svih metoda pre balansiranja klasa*



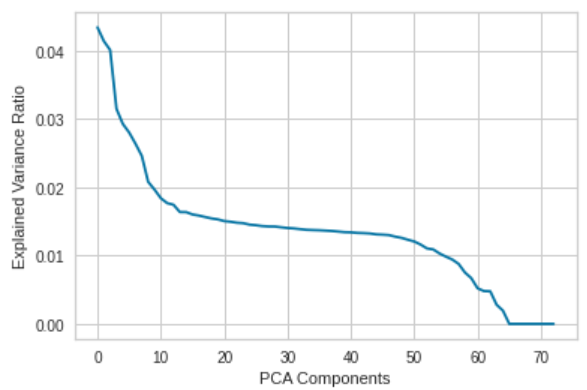
*Slika 30: Poređenje ROC krive i AUC rezultata svih metoda nakon balansiranja klasa*

# 4 Klasterovanje

## 4.1 Algoritam K-sredina

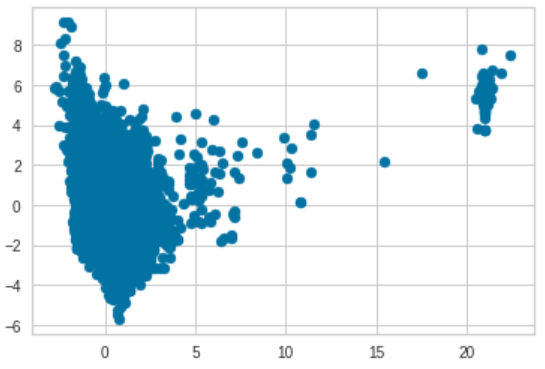
Algoritam K-sredina *(K-means[11](https://www.lifewire.com/k-means-clustering-1019648" \l ":~:text=The k-means clustering algorithm,prior knowledge of those relationships.))* korišćen je za grupisnje podataka u klastere na osnovu njihove

sličnosti. Na slici 31 je prikazana varijansa (*Explained Variance Ratio*[*12*](https://vitalflux.com/pca-explained-variance-concept-python-example/)*)* koju ćemo koristiti radi odabira broja PCA komponenti. U našem slučaju to će biti 2.



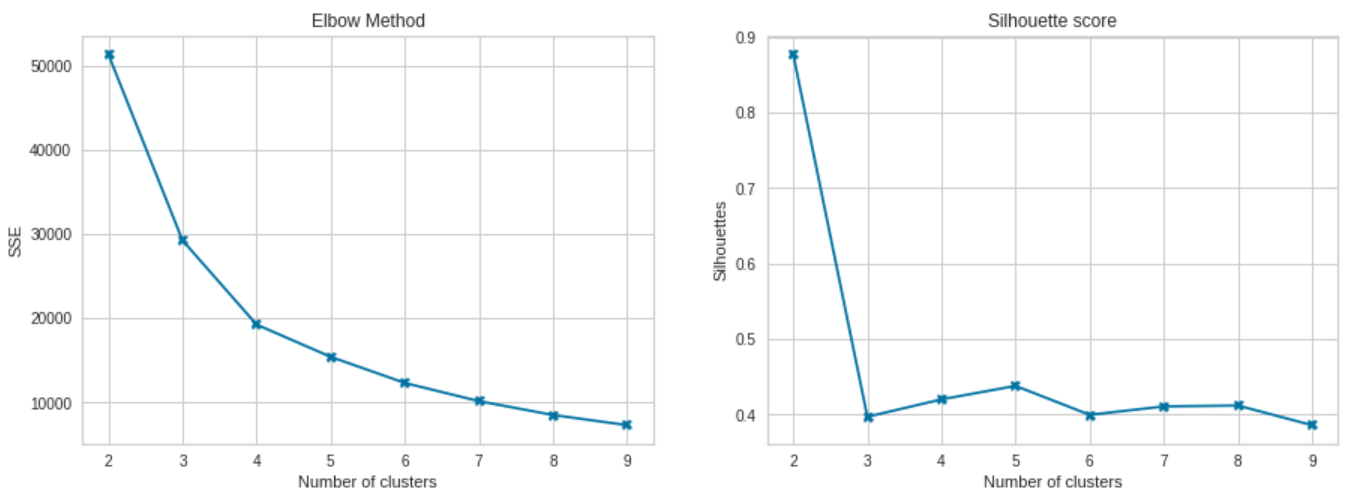
*Slika 31: Explained Variance Ratio*

Nakon toga na naš model primenjujemo tehniku analize glavnih komponenti *(PCA - Principal Component Analysis).* Model pre klasterovanja je prikazan na slici 32.



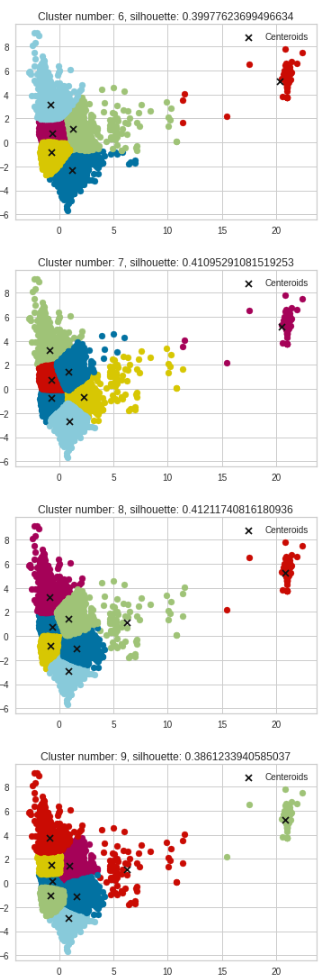
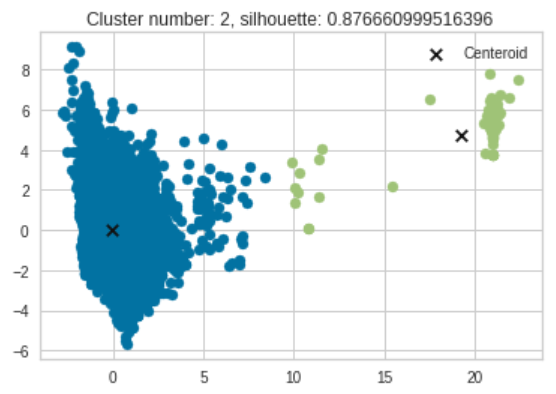
*Slika 32: Model pre klasterovanja*

Sada biramo optimalan broj klastera. Mere koje koristimo za to su inercija koja pri minimizaciji koristi „pravilo lakta” *(Elbow method)* i silueta, koja je najveća za 2 klastera, što će biti optimalno.

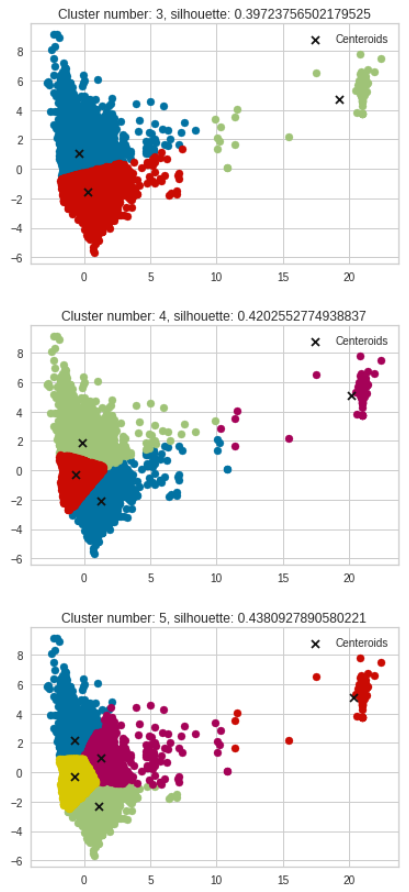


*Slika 33: Mere inercija i silueta*

Na slici 34 je prikazan model sa optimalnim brojem klastera tj. 2, centeroidima i siluetom. Za ostali broj klastera, dakle od 3 do 5 klastera prikazani su modeli na slici 35 a od 6 do 9 klastera na slici 36.



*Slika 34: Model sa 2 klastera*

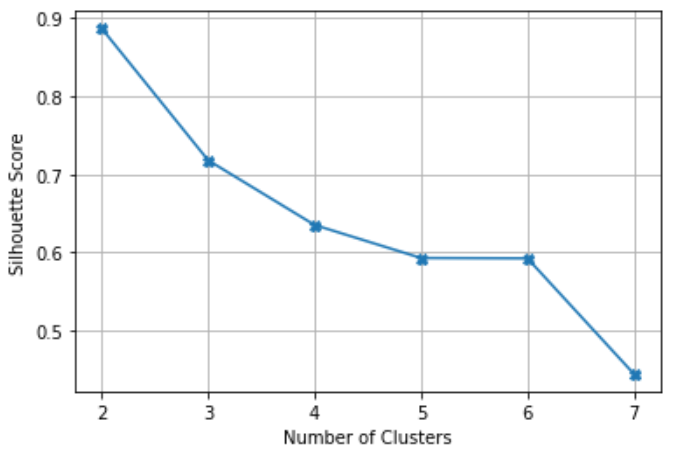
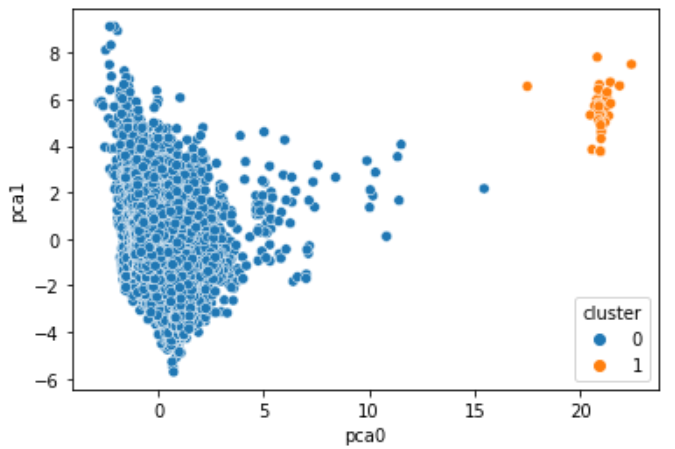


*Slika 35: Modeli sa 3, 4 i 5 klastera Slika 36: Modeli sa 6, 7, 8 i 9 klastera*

## 4.2 **Sakupljajuće klasterovanje**

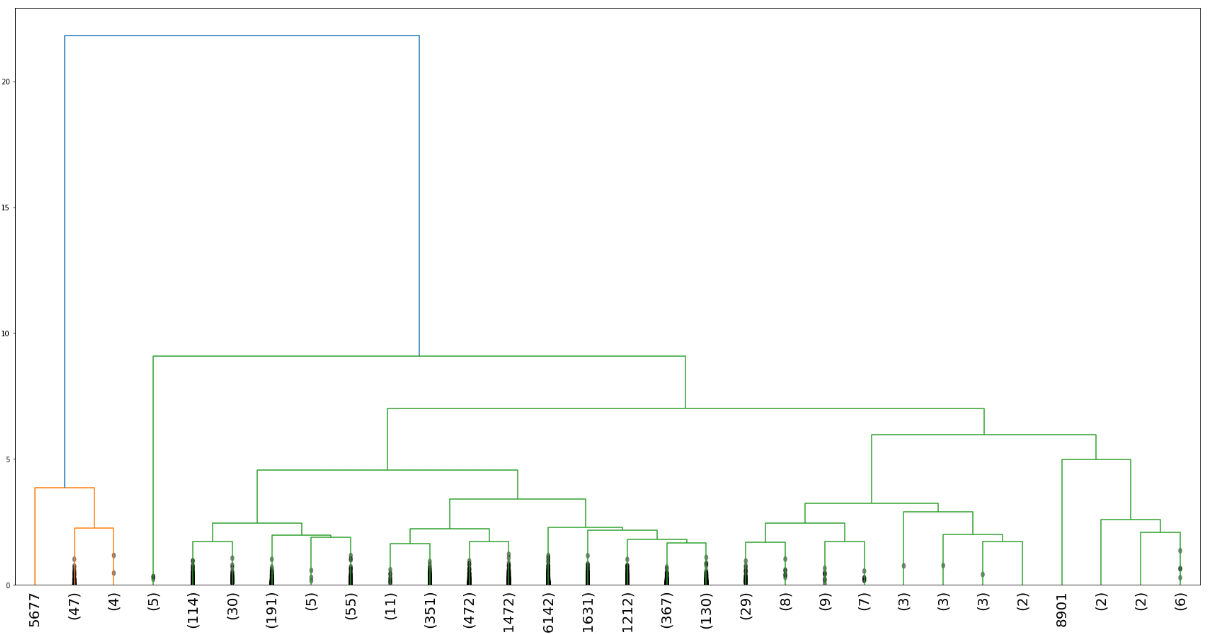
Sakupljajuće klasterovanje *(Agglomerative Clustering[13](https://towardsdatascience.com/agglomerative-clustering-and-dendrograms-explained-29fc12b85f23" \l ":~:text=Agglomerative Clustering is a type,in different clusters are dissimilar.))* je algoritam hijerarhijskog klasterovanja koji se koristi za grupisanje sličnih tačaka podataka u klasterima. Ovo je pristup odozdo prema gore (bottom-up), u kome svaka tačka podataka počinje kao sopstveni klaster, a nakon toga se iterativno spajaju klasteri na osnovu njihove sličnosti sve dok se ne dostigne željeni broj klastera.

Počinjemo tako što testiramo ponašanja algoritma za različite vrednosti parametara. Kao i ko K sreditna koristili smo PCA. Definišemo klastere u rasponu od 2 do 8, slično kao za algoritam K sredina. Isprobavamo sve moguće vrednosti za parametar vezivanja *(linkage)* a to su *average, ward, complete* i *single*. Na slici 37 izdvajamo model sa najboljom ocenom siluete, brojem klastera koji je jednak 2 i najboljim parametrom vezivanja a to je average. Ostali modeli biće prikazani na slici 40. Na slici 38 su prikazani rezultati siluete u odnosu na broj klastera.



*Slika 37: Najbolji model Slika 38: Rezultati siluete*

Da bismo dobili vizuelnu reprezentaciju hijerarhijskog klasterovanja koristimo biblioteku *scipy*, pomoću koje kreiramo *dendrogram*[*14*](https://www.displayr.com/what-is-dendrogram/). Na slici 39 biće prikazano poslednjih 30 klastera. Na osnovu ove slike ništa posebno značajno nismo mogli zaključiti.



*Slika 39: Dendogram poslednjih 30 klastera*

## 

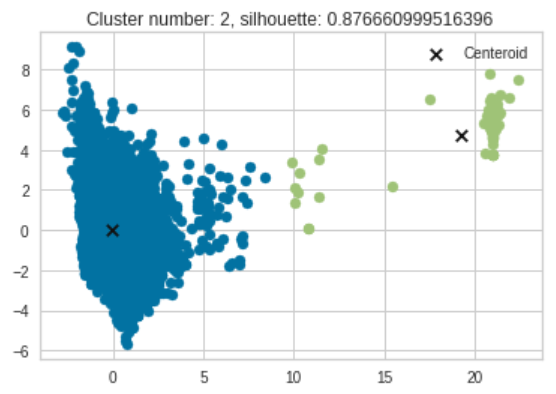
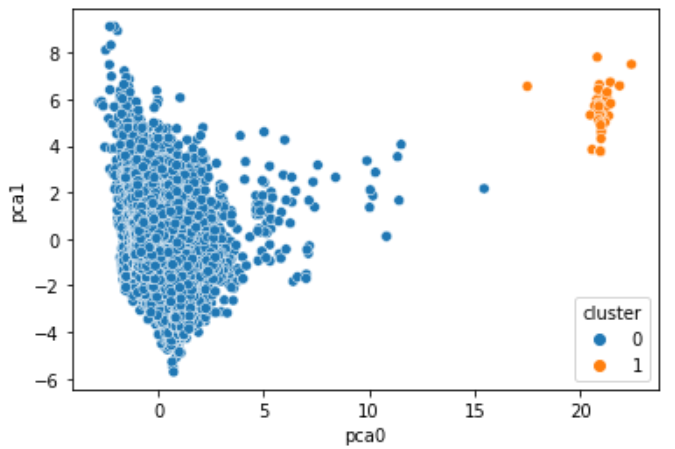
*Slika 40: Modeli za broj klastera od 2 do 8 i parametrima vezivanja average, ward, complete i single*

## 4.3 Poređenje modela klasterovanja

Na slici 40.1 i 40.2 prikazani su rezltati poređenja silueta za modele K sredina i Sakupljajućeg klasterovanja. Na slici 40.3 i 40.4 prikazano je poređenje najboljih modela za oba algoritma. Oba modela daju najbolje rezultate za 2 klastera. Rezultati siluete su takođe slični, 0.89 za algoritam K sredina i 0.88 za sakupljajuće klasterovanje.

# 

*Slika 40.1: Rezultati siluete za K sredina Slika 40.2: Rezultati siluete Sakupljajućeg klasterovanja*

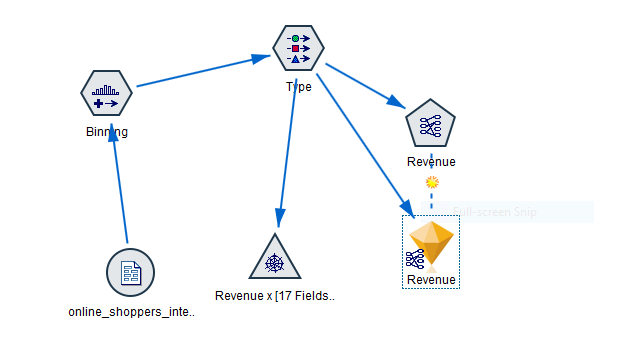
**

*Slika 40.3: Najbolji model K sredina klasterovanja Slika 40.3: Najbolji model Sakupljajućeg klasterovanja*

# 5 Pravila pridruživanja

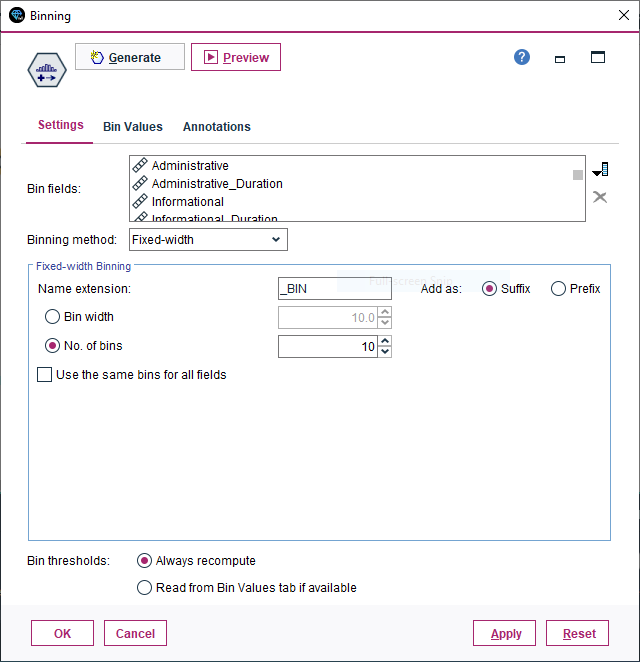
## 5.1 Apriori

*Apriori*[*15*](https://www.geeksforgeeks.org/apriori-algorithm/)algoritam se koristi za pronalaženje zanimljivih obrazaca koji postoje u skupovima podataka. Za primenu ovog algoritma korišćen je *IBM SPSS Modeler*[*16*](https://www.ibm.com/docs/en/spss-modeler/18.0.0?topic=guide-about-spss-modeler). Na slici 41 prikazan je dijagram toka.



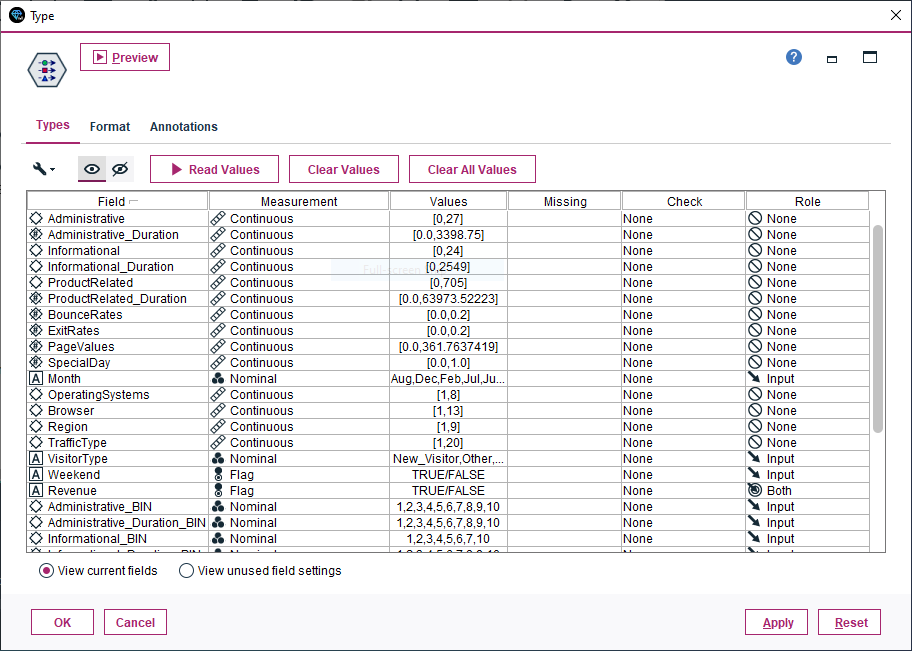
*Slika 41: Dijagram toka*

Pošto Apriori algoritam ne radi sa neprekidnim podacima nego samo sa diskretnim, morali smo neprekidne čvorove da diskretizujemo i to radimo pomoću *bining* opcije u SPSS-u. Binovanje *(Bining*[*16*](https://www.ibm.com/docs/en/cloud-paks/cp-data/4.7.x?topic=operations-binning-node)*)* se koristi pri radu sa neprekidnim promenljivim, konvertujući ih u „binove“ – intervale. Za metodu biramo fiksnu širinu. Na slici 42 je prikazan prozor.



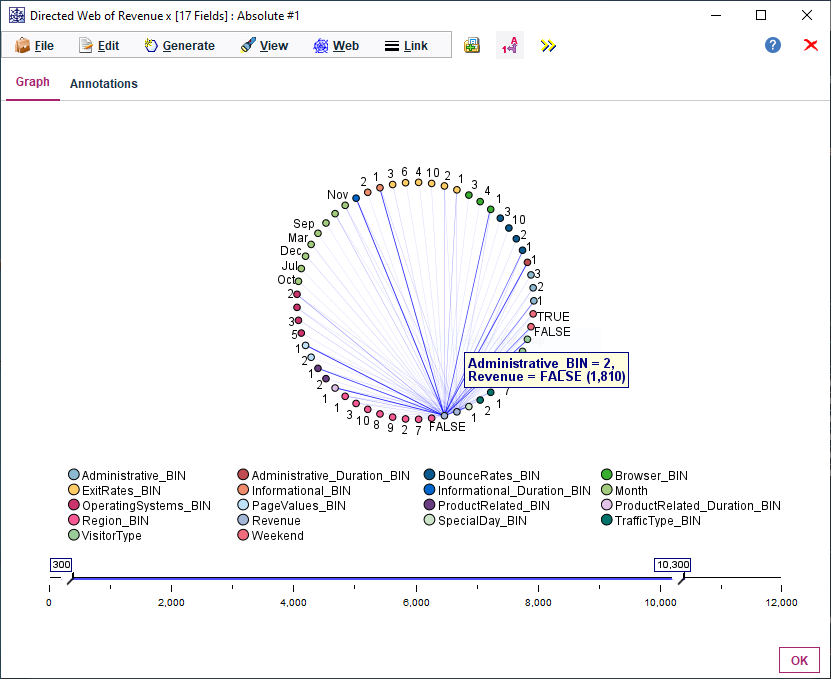
*Slika 42: Prozor bining opcije u SPSS-u*

Na slici 43 prikazan je *Type*[*17*](https://www.ibm.com/docs/en/cloud-paks/cp-data/4.7.x?topic=operations-type-node) prozor na kome smo podesili uloge naših atributa nakon binovanja.



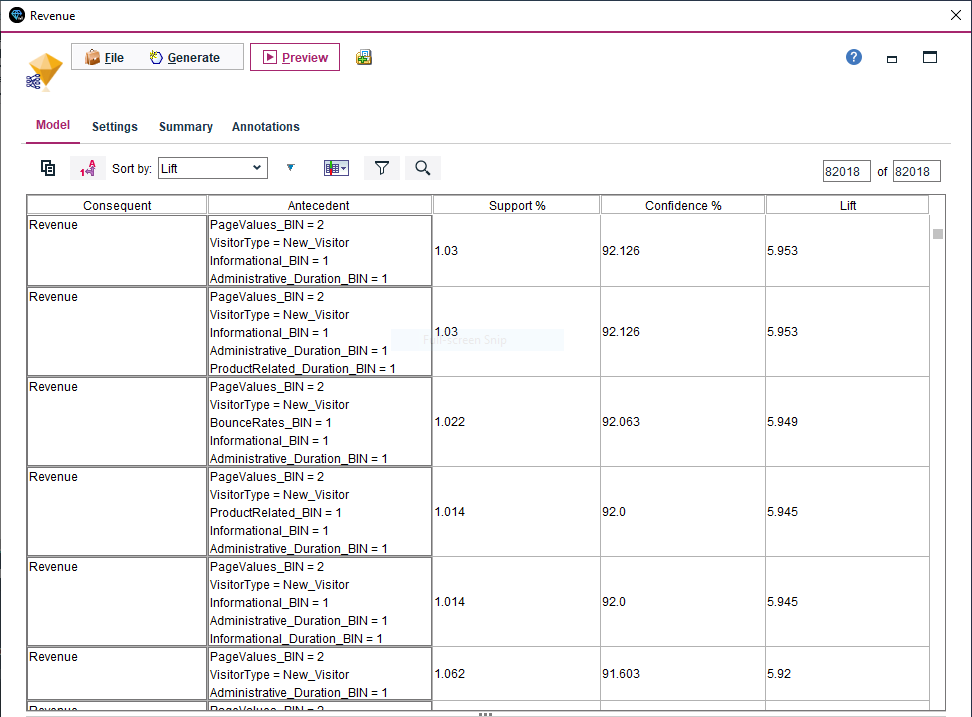
*Slika 43: Type prozor*

Na slici 44 prikazan je *Web*[*18*](https://www.ibm.com/docs/en/cloud-paks/cp-data/4.7.x?topic=graphs-web-node) prozor. „Deblje” linije predstavljaju jaču korelaciju. Na primer između našeg ključnog atributa Revenue (False) i Month (Nov).

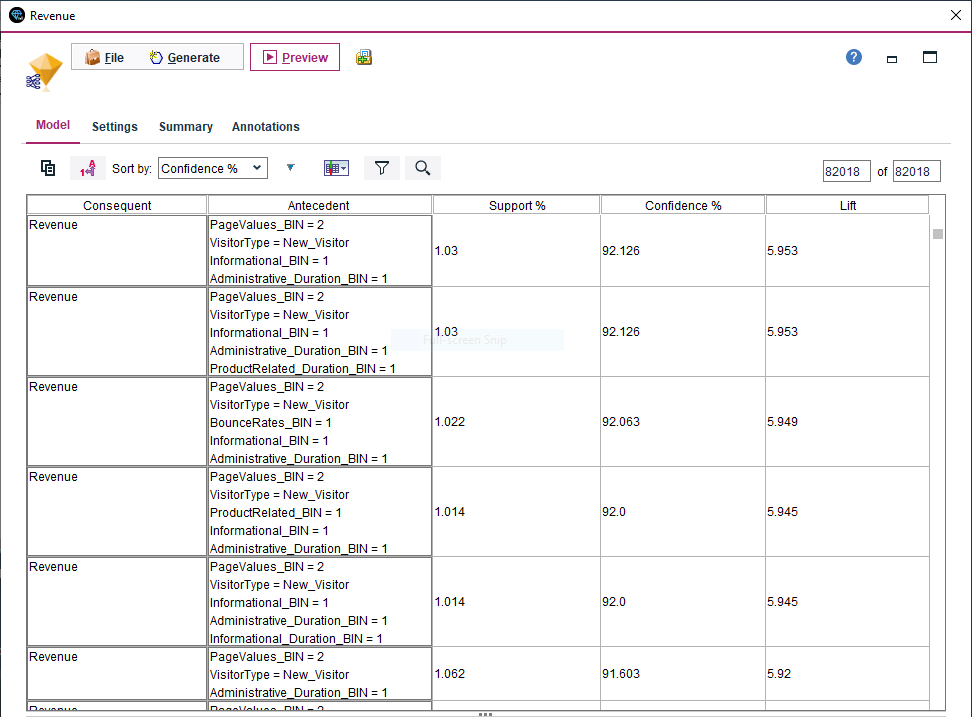


*Slika 44: Prozor na kome je prikazana mreža*

Na slici 45 prikazan je przor za *Lift* sortiranje. Lift vrednosti su jako visoke što ukazuje da su pravila zastupljenija nego očekivano. Na slici 46 prikazan je prozor za *Confidence (pouzdanost) s*ortiranje. Tu nam je atribut PageValues interesantan, kao i činjenica da je VisitorType jedank novom korisniku. *Support* (*podrška*) sortiranje nije prikazano jer iz njega nije nešto značajno moglo da se zaključi.



*Slika 45: Lift sortiranje*

**

*Slika 46: Confidence sortiranje*

# 6 Zaključak

Predviđanje da li će neko kupiti nešto preko interneta ne može biti perfektno, ali rekao bih da nije ni teško. Neki atributi koje smo primetili na početku još u eksplorativnoj analizi, koji bi intuitivno mogli da budu značajni, kao na primer *Special Day* atribut, nakon klasifikacije na tabeli značajnosti atributa nije imao neku visoku poziciju. Očekivano je da klasterovanje ne daje neke značajnije rezultate jer je ipak u pitanju skup za klasifikaciju. Model pravila pridruživanja uspeo je da pronađe neka određena pravila.