UNIVERZITET U BEOGRADU
MATEMATIČKI FAKULTET

Desa Marinković

Metode istraživanja podataka u proceni rizika u bankarstvu

Master rad

Beograd, 2010

1 Uvod

Globalna konkurencija, dinamično tržište i sve brža tehnološka inovacija stvaraju velike izazove u svim oblastima pa i u bankarstvu. Danas, u vreme globalizacije i konkurencije, pojedinci i ogranizacije koji dobiju pravu informaciju na vreme imaju veće šanse za uspeh.

Finansijski sektor je prinuđeni da se aktivnije i kreativnije bori da zadrži stare i osvoje nove korisnike. Danas, standarana bankarska ponuda proizvoda kao i konvencionalne metode marketinga nisu dovoljan uslov za opstanak. Banke su prinuđene da za svakog klijenta imaju pravu ponudu u odgovarajućem trenutku, da znaju koji korisnici najčešće uzimaju koje proizvode, koji korisnici uzimaju kredite, koji proizvodi se najčešće uzimaju zajedno itd. Veći broj proizvoda uvećava i profit ali i izloženost banke prema klijentima i tržištu što vodi povećanju rizika. Rukovođenje u datim institucijama postaje sve zahtevnije jer odluke moraju da budu donete brzo i realizovane efikasno, a svaka odluka mora da bude doneta sa minimalnim rizikom po instituciju. Upravljačkoj strukturi je potrebna podrška u odlučivanju. Međutim, današnji informacioni sistemi nisu konstruisani kao sistemi za podršku u odlučivanju.

Podaci koji su potrebni za podršku u odlučivanju se već nalaze u bazama podataka. Ovi podaci su prikupljani (automatski ili ručno) kao podrška dnevnim transakcijama i operacijama. U prethodnom periodu su ovi podaci prikupljani zbog internog ili eksternog izveštavanja, i radi zahteva centralne banke. Danas znamo da se u ovim podacima nalaze i mnogo vrednije informacije. Često su baze u kojima su podaci

uskladišteni jako velike i mere se terabajtima. Zbog veličine i složenosti izdvajanje informacija iz takvih baza je vrlo komplikovano. Upravo na takvim podacima se primenjuju metode istraživanja podataka jer imaju mehanizme koji omogućavaju uočavanje veza i zavisnosti među podacima koje nisu očigledne bilo zbog obima podataka ili zato što je zbog brzog generisanja onemogućena njihova analiza.

Jedna od prvih primena istraživanja podataka je istraživanje potrošačke korpe u supermarketu. Analizirani su podaci skupljeni na POS terminalu-u da bi se došlo do zaključka koji proizvodi se najčešće kupuju zajedno. Dato istraživanje dovelo je do nekih sasvim očekivanih rezultata, npr. da se hleb i mleko, ili vino i sir kupuju 'u paketu' ali i do nekih sasvim neočekivanih kao što je npr. kupovina pelena i piva petkom uveče. Rezultat ovakvih istraživanja je promena rasporeda proizvoda u supermarketu tako što se proizvodi koji se kupuju zajedno postavljaju da budu i fizički blizu. Rezultat ovakvih aktivnosti je bilo povećanja prometa i do 20 posto što je dalo novi podstrek istraživanju podataka.

Primene istraživanja podataka su neograničene u raznim oblasitima. Mogu se ispitivati sekvence i periodičnosti ponašanja klijenata i na osnovu toga ih svrstati u različite kategorije za koje se formiraju proizvodi specijalno predviđeni za tu kategoriju.

U ovom radu ćemo se baviti primenom istraživanja podataka u bankarstvu. Neka od pitanja na koje nam istraživanje podataka može dati odgovore u bankarstvu su :

1. Koje su uobičajene transakcije koje klijent obavi pre nego što promeni banku?
2. Koji je profil tipičnog ATM klijenta i koje proizvode on uobičajeno uzima?
3. Koji proizvodi se uobičajeno uzimaju zajedno i od strane koje grupe klijenata?
4. Koji je profil visokorizičnog klijenta?
5. Koje proizvode bi današnji klijenta još mogli da uzmu?

Najveći izazov u bankarstvu danas predstavlja konstruisanje sistema koji bi identifikovali, merili i upravljali rizicima. Merenje rizika predstavlja pokušaj kvantitativnog izražavanja rizika od gubitka pri svakoj odluci. U bankarstvu su najznačajniji kreditini i tržišni rizik ali su prisutne i druge vrste rizika (rizik likvidnosti, operacioni rizik, koncentracioni rizik itd.). Konstrukcija sistema koji bi pratili i merili ove rizike kao i njihovu međusobnu reakciju i interakciju sa tržištem je danas jedno od najznačajnijih polja istraživanja u ovoj oblasti.

U bankarstvu se čuva velika količina podataka o svakom klijentu, na osnovu tih podataka klijenti se mogu svrstati u nekoliko kategorija. Ovako grupisani podaci o klijentima se mogu koristiti za marketinške kampanje, analizu rizika, sprečavanje odliva klijenata itd. Za svakog klijenta se na osnovu kategorije kojoj pripada može reći koja je njegova buduća vrednost kao klijenta kao i kategorija rizika kojoj pripada.

U ovom radu bavićemo se profilom viskorizičnog bankarskog klijenta. Odnosno, napravićemo model koji klijente klasifikuje u jednu od dve kategorije *visokorizičan* i *niskorizičan*. Ulazni podaci za model su dvogodišnja istorija rada klijentovog tekućeg računa i demografski podaci a cilj pomenuta podela.

2 Priprema podataka

Priprema podataka je najzahtevnija faza u procesu istraživanja. Nepotpuni, nekonzistentini i podaci sa šumom vode ka lošim rezultatima istraživanja. Priprema ja ključni korak koji skraćuje sam proces istraživanja i povećava preciznost dobijenih rezultata.

U velikim sistemima baza podataka podaci često potiču iz različitih izvora i istorijski su dugo prikupljani, što za posledicu ima veliki broj grešaka, neočekivanih vrednosti i nepravilnosti u podacima. Primena algoritama istraživanja nad ovakvim podacima daje vrlo loše razultate jer nailazimo na podatke kojima nedostaju vrednosti nekih atributa, neki atributi izlaze iz očekivanog opsega vrednosti, itd.

Nepotpuni podaci (nedostatak vrednosti nekog atributa) se javljaju iz mnogo razloga: vrednosti atributa nisu uvek dostupne, neki od atributa mozda u vreme kad su evidentirani nisu smatrani bitnim bilo zbog nerazumevanja ili zbog ljudske ili mašinske greške.

Pod podacima sa šumom podrazumevamo podatke koji imaju vrednosti atributa koje nisu očekivane tj. izlaze iz očekivanog opsega vrednosti ili sadrže greške. Do ovakvih promena na podacima dolazi usled ljudske ili mašinske greške pri unosu podataka, tehnoloških ograničenja, grešaka u prenosu podataka itd.

Često u procesu istraživanja nisu dovoljni podaci iz jednog izvora nego je potrebno da uključimo podatke iz različitih baza, datoteka ili kocki podataka. Integracija

podataka iz više različitih izvora može da dovede do nekonzistentosti.Na primer, u dva izvorima jedan podataka ima različito ime tako da, nakon integracije dobijamo tabelu sa dva atributa različitog imena a istih vrednosti atributa. Nekonzistentnost podataka ne utiče fatalno na preciznost istraživanja ali značajno utiče na efikasnost primenjenih algoritama istraživanja.

Metode koje se koriste za otklanjanje prethodno navedenih nedostataka i pripremu podataka za istraživanje mogu se podeliti u četri grupe (slika 1):

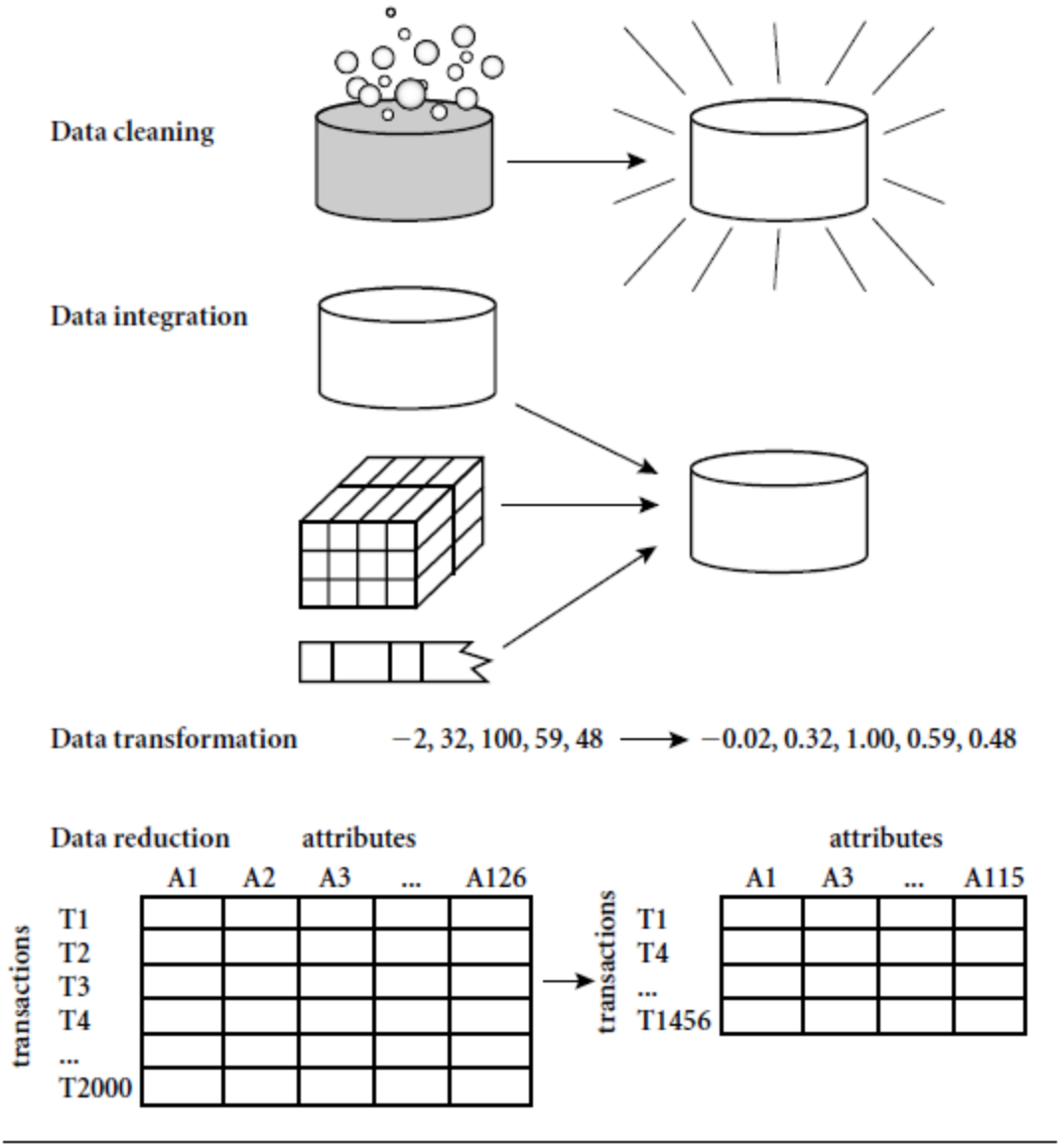
1. Metode za čišćenje podataka
2. Metode za integraciju podatka
3. Metode za transformacija podataka
4. Metode za redukciju podataka

Redukcija dimenzije podataka je proces koji je u izvesnom smislu suprotan integraciji. Redukcijom se smanjuje količina ili broj dimenzija podataka koji se koriste tako da rezultati istraživanja ostanu istog ili skoro istog kvaliteta.

Normalizacija je još jedna metoda za pripremu podataka, naročito je bitna kod algoritama koji se zasnivaju na rastojanju kao što su neuronske mreže, klasterovanje, itd. Normalizacijom se podaci skaliraju tako da pripadaju određenom malom opsegu vrednosti.

Bitno je naglasiti da ove tehnike nisu uzajamno isključive i da se ne koriste uvek u navedenom redosledu. Često primena jedne metode dovodi do promena koje zahtevaju ponovnu primenu neke prethodno primenjene metode što proces pripreme čini komplikovanijim i vremenski zahtevnijim. Prema nekim autorima priprema podataka oduzima 60 posto vremena celog procesa istraživanja.

Da bi preprocesiranje bilo uspešno neophodno je imati globalan pregled i detaljan opis podataka koji će biti korišćeni. Opis podataka nam pomaže da bolje shvatimo glavne i tipične karakteristike podataka i da lakše identifikujemo koje vrednosti treba tretirati kao greške i nepravilnosti, što će biti korisno za efikasnije izvršavanje faze čišćenja i integracije podataka.



Slika.1. Metode za pripremu podataka

2.1 Čišćenje podataka

Procedure za čišćenje podataka smanjuju nepravilnosti tako što popunjavaju vrednosti atributa koje nedostaju i identifikuju vrednosti atributa koje nisu u dozvoljenom opsegu . Čišćenje podataka je najčešće iterativni postupak koji se sastoji od uočavanja nepravilnosti i transformacije podatka čija je nepravilnost uočena. Najčešća situacija sa kojom se susrećemo je da neki atribut nema vrednost. Na primer, za klijenta

postoje svi podaci osim vrednosti atributa godišnji prihod. Ako se smatra da je taj atribut izuzetno bitan u procesu istraživanja, akcije koje mogu da se preduzmu u tom slučaju su:

1. Ignorisanje date instance – svi podaci za datog klijenta se uopšte ne razmatraju. Ovaj metod nije efikasan osim u slučaju kada instanca ima više atributa bez vrednosti.

2. Ručno popunjavanje nedostajuće vrednosti – ovaj metod zahteva dosta vremana i nije primenjiv na baze sa velikom količinom podataka.

3. Korišćenje konstante za popunjavanje nedostajućih vrednosti. U ovom slučaju se sve instance u kojima nedostaje vrednost popunjavaju istom vrednošću. Na primer, svuda gde nije evidentirana vrednost atributa godišnji prihod upisuje se vrednost 5000.

4. Korišćenje srednje vrednosti atributa za popunjavanje nedostajućih vrednosti. Na primer, ako je prosečan godišnji prihod 15.000 za sve instance gde vrednost atributa godišnji prihod nije evidentirana upisuje se 15.000.

5. Popunjavanje vrednosti atributa srednjom vrednošću atributa instanci koje pripadaju istoj klasi. Na primer, ako se razmatra kategorija rizika, zamenjuje se vrednost atributa srednjom vrednošću atributa klijenata koji pripadaju istoj kategoriji rizika.

U nekim slučajevima, nedostatak vrednosti atributa ne znači grešku. Na primer, kada se klijent prijavljuje za kreditnu karticu, očekuje se da unese i broj pasoša. Međutim, ukoliko klijent nema pasoš ovo polje će ostati prazno. Još u procesu projektovanja baze trebalo bi predvideti da li atribut mora da ima vrednosti ili ne.

Nepravilnosti i greške u podacima su izazvane raznim faktorima: loše projektovanim ulazom sa mnogo opcionih polja, ljudskom greškom pri unosu podataka, namernom greškom (npr. kada klijent ne želi da ostavi tačan podatak) i zastarelim podacima (npr. adrese koje više nisu važeće). Greške se takođe javljaju kad se podaci koriste u druge svrhe od onih za koje su predviđeni. Nepravilnosti se mogu takođe pojaviti i posle integracije podataka.

Kako god da je do grešaka i nepravilnosti došlo, prvi korak je njihovo identifikovanje. Najbolja polazna tačaka za identifikaciju grešaka je analiza detaljnog opisa podataka. Opis podataka bi trebalo da sadrži za svaki atribut domen i tip podatka, očekivane vrednosti za svaki atribut, podatak da li sve vrednosti moraju da pripadaju očekivanom opsegu i da li ima zavisnosti medju atributima.
Postoji više komercijalnih alata koji se bave otkrivanjem i ispravljanjem grešaka i nepravilnosti u podacima. Neki od njih koriste statističke funkcije da bi otkrili veze medju atributima, drugi klasterovanje da bi uočili vrednosti koje izlaze iz očekivanog opsega, dok neki, čak mogu da koriste i detaljan opis podataka. Mnoge od grešaka mogu biti ispravljene i ručno tj. upotrebom nekog upitnog jezika.

Nakon uočavanja grešaka i nepravilnosti drugi korak je transfomacija podataka. Neki alati dozvoljavaju jednostavne transformacije podataka ili dozvoljavaju da transformacije budu definisanje preko GUI-ja ali se zbog vrlo ograničenih mogućnosti najčešće pišu prilagođeni skriptovi.

Proces otkrivanja grešaka i transformacije podataka u cilju uklanjanja tih grešaka troši vreme i daje nikakvu povratnu informaciju. Zbog toga a i zbog činjenice da transformacija može da dovede do novih grešaka, ukoliko je došlo do bilo kakvih promena u podacima, nakon završetka procesa otkrivanja grešaka i transformacije podataka, korisnik treba da se vrati na početak i ponovo proveri greške.

2.2 Intergracija podataka

U procesu istraživanja podataka često je neophodno da koristimo podatke iz više baza, kocki podataka ili datoteka. U ovom slučaju moramo biti jako oprezni jer se često dešava da je isti podatak opisan različitim atributima u različitim izvorima podataka. Višestruko korišćenje istih podataka produžava proces čišćenja podataka i značajno utiče na efikasnost algoritama istraživanja. Da bi ovakve nepravilnosti sveli na minimum nepohodno je detaljno opisati sve podatke.

Druga bitna stvar o kojoj treba voditi računa pri integraciji podatak jeste: da li neki atribut može da bude izveden iz drugih atributa. Na primer, ako u istraživanju već

koristimo podatke o kvartalnim prihodima klijenata suvišno je koristiti i godišnji prihod jer se godišnji prihod dobija agregacijom iz kvartalnih. U većini slučajeva međusobna zavisnost atributa nije lako uočljiva kao u navedenom primeru. Za merenje međusobne zavisnosti atributa koristi se korelacijska analiza. Korelacijska analiza meri zavisnost atributa na osnovu vrednosti atributa. Za numeričke atribute najčešće korišćeni metod korelacijske analize je Pirsonov moment koeficijent. Vrednost ovog koeficijenta govori da li su atributi nezavisni ili pozitivno (ili negativno) povezani. U slučaju da su atributi pozitivno povezani vrednosti jednog aributa rastu sa porastom vrednosti drugog atributa. Za ispitivanje zavisnosti kategoričkih atributa se koristi χ^2 test koji pretpostavlja da su atributi nezavisni, a zatim statistički testira ovu hipotezu.

Još jedna konfliktna situacija sa kojom se možemo da sretnemo je različito evidentiranje istog podatka u različitim izvorima. Na primer, starost klijenta može u jednoj bazi da bude zabeležena pomoću broja godina (npr. kao brgodina =28) a u drugoj pomoću informacije o godini rođenja (npr. kao godrođenja=1982).

2.3 Transformacija podataka

U ovom koraku, podaci se transformišu u oblik koja proces istraživanja čini brzim i efikasnijim. Tehnike koje se koriste pri transformaciji su:

1. **Preoblikovanje.** Preoblikovanjem se otklanjaju šumovi u numeričkim atributima. Metode koje se najčešće koriste se mogu podeliti u tri grupe: metodom ograđivanja (eng. *binning*) se skup mogućih vrednosti atributa deli u nekoliko grupa, metodom izravnavanja (eng. *smooting*) se podaci preoblikuju na osnovu srednje vrednosti atributa ili graničnih vrednosti grupe i metodom regresije se vrednosti atributa aproksimiraju adekvatnom funkcijom. Može se koristiti i klasterovanje kojim se slične vrednosti atributa grupišu a ostale smatraju vrednostima van granica.

2. **Agregacija.** Agregacija se koristi kada su za analizu potrebni zbirni podaci. Tipična primena agregacije je konstrukcija kocki podataka. Na primer, na osnovu mesečnih

podataka o prodaji nekog proizvoda dobijamo godišnji total. Rezultujući podaci su manjih dimenzija i bez gubitka informacija neophodnih za analizu.

3. **Generalizacija.** Generalizacijom se podaci nižeg nivoa zamenjuju podacima višeg nivoa. Na primer, kategorički atribut ulica može da uzima veliki broj vrednosti. Uz činjenicu da se se imena ulica ponavljaju u raznim gradovima bilo kakva analiza po ovom atributu neće dati zadovoljavajuće razultate. Za većinu upotreba neophodno je zameniti vrednost atributa ulica podatkom višeg nivoa: grad, region i sl.

4. **Normalizacija.** Normalizacija atributa ubrzava fazu učenja u istraživanju podataka jer se atribut skalira tako da njegove vrednosti pripadaju malom opsegu vrednosti npr. od 0 do 1. Normalizacija je posebno značajna kod metoda koji se zasnivaju na rastojanju kao što su neuronske mreže, metode najbližeg suseda i klasterovanja jer sprečava da atributi sa velikim vrednostima (npr.prihod) u prosesu istraživanja budu smatrani bitnijim od atributa čiji je opseg vrednosti manji (npr. godine). Najčešće korišćene metode normalizacije su min-max normalizacija, normalizacija z vrednosti i normalizacija decimalnim skaliranjem.

   Min-max normalizacija je linearna transfomacija koja za sve vrednosti atributa A čiji su maksimum i minimum $\min_A$ i $\max_A$ vrši preslikavanje $[\min_A, \max_A] \rightarrow [novi\_\min_A, novi\_\max_A]$. Normalizacija z vrednosti se zasniva na srednjoj vrednosti i standardnoj devijaciji atributa i koristi se kada stvarni minimum i maksimum nisu poznati ili postoji mogućnost da su minimum i maksimum vrednosti koje izlaze iz domena vrednosti atributa. Normalizacija decimalnim skaliranjem je pomeranje decimalne tačke deobom vrednosti atributa odredjenom konstantom čija vrednost zavisi od maksimalne vrednosti atributa. Poslednja dva metoda menjaju neznatno podatke dok min-max normalizacija čuva odnos među vrednostima atributa.

5. **Konstrukcija atributa.** Dodatni atributi se konstruišu i dodaju na osnovu postojećih u cilju povećanja tačnosti i efikasnosti istraživanja. Novi atributi treba da daju nove informacije u otkrivanju novih veze među postojećim podacima. Koristi se nekoliko tehnika za konstrukciju novih atributa koje su zasnovane na poznavanju date oblasti. Dodavanjem novih atributa povećava se složenost podataka i usporava proces istraživanja. Princip koji bi trebalo primenjivati je da se novi atributi dodaju postojećim a da se zatim uklone svi oni atributi koji se posle tog dodavanja mogu smatrati irelevantnim.

2.4 Redukcija podataka

Metode za redukciju podataka se primenjuju za dobijanje podataka manjeg obima uz zadržavanje karakteristika orginalnog skupa podataka. Neke od tehnika redukcije podataka su izbor atributa, redukcija dimenzije podataka, redukcija brojnosti podataka, diskretizacija i hijerarhija koncepata. U odeljku 2.4.1 je dat detaljan opis tehnike izbora atributa, a u odeljku 2.4.2 diskretizacije i hijerarhije koncepata.

2.4.1 Izbor atributa

Rezultati istraživanja podataka zavise od kvaliteta ulaznih podataka. Ukoliko su podaci neadekvatni i/ili lošeg kvaliteta i rezultati će biti manje precizni. Baze podataka uobičajeno sadrže veliku količinu podataka koji nisu povezani sa specifičnim istaživanjem. Korišćenje atributa koji su zasnovani na ovim podacima dovodi do manje efikasnosti primenjenog algoritma i manje preciznosti. Cilj ove metode je da elimniše sve irelevantne i nađe najmanji podskup atributa tako da dobijena raspodela verovatnoća klasa podataka bude što bliža orginalnoj raspodeli koja se dobija kada se koriste svi atributi.

Za n atributa postoji 2^n mogućih podskupova,tako da broj podskupova raste eksponencijalno sa povećanjem broja atributa kao i broja klasa. *Najbolji* i *najgori* atributi se uobičajeno biraju statističkim metodama koje mere koliko su atributi međusobno nezavisni. Najboljim se smatraju oni koji su medjusobno nezavisni a imaju veliku zavisnost sa atributom koji određije pripadnost klasi.

Osnovne metode za izbor atributa uključuju neku od sledećih tehnika:

1. Izbor korakom napred: početni skup atributa je prazan. U svakom koraku se dodaje najbolji od preostalih atributa.
2. Eliminacija korakom unazad: početni skup atributa sadrži sve atributa. U svakom koraku se eliminiše najgori od preostalih atributa.

3. Kombinacija prethodno navedenih metoda: u svakom koraku se bira najbolji i eliminiše najgori od preostalih atributa.
4. Indukciju drvetom odlučivanja: koristeći sve atribute konstruiše se drvo odlučivanja. Posle konstrukcije, svi atributi koji se ne pojave u drvetu smatraju se nebitnim i eliminišu se.

Izbor što manjeg broja atributa je značajan i u analizi rezultata istraživanja jer na osnovu manjeg broja atributa lakše shvatamo zašto su neki podaci baš tako klasifikovani.

2.4.2 Diskretizacija i hijerarhija koncepata

Istraživanje na redukovanim podacima zahteva manje operacija i efikasnije je. Iako se redukcijom gubi na detaljnosti podataka dobijeni podaci imaju veću vrednost za primenjeni algoritam i lakši su za interpretaciju. Tehnike za diskretizaciju se dele na dve grupe na osnovu pravca u kome se vrši diskretizacija i da li se tom prilikom koriste atributi koji određuju pripadnost klasi. Diskretizacijom se opseg vrednosti numeričkog atributa deli u nekoliko intervala, pri čemu se graničnim vrednostima intervala zamenjuju stvarne vrednosti atributa. U zavisnosti da li se na početku biraju granične tačke na osnovu kojih se opseg vrednosti atributa deli na intervale, ili se sve vrednosti atributa posmatraju kao potencijalne granične tačke a zatim se vrši njihova eliminacija, za metodu diskretizacije se kaže da radi odozgo na niže ili odozdo na više.

Primena hijerarhije podataka podrazumeva diskretizaciju numeričkih atributa u smislu da se podaci nižeg nivoa zamenjuju podacima višeg novoa. Na primer, numerički atribut godine se zamenjuje podatkom višeg nivoa mlad, srednjih godina ili penzioner. Iako se ovakvom generalizacijom gubi na detaljnosti podatka,pokazalo se da primenom algoritama na ovakve podatke dobijaju bolji rezultati uz manji broj ulazno/izlaznih operacija.

3 Klasifikacija

Klasifikacija je jedna od najčešće korišćenih metoda za istraživanje podataka jer je u našoj prirodi da stvari oko sebe, kako bih ih bolje shvatili ili organizovali, konstantno klasifikujemo i kategorizujemo. Klasifikacija se koristi u :

- odabiru najbolje terapije za pacijenta od nekoliko mogućih
- klasifikaciji kreditinih zahteva kao visokorizičnih ili niskorizičnih
- proceni da li će i koji korisnici kupiti određeni proizvod
- izboru ciljne grupe klijenata za marketinške kampanje

Navedene primeri predstavljaju neke od tipičnih primena klasifikacije u medicini, bankarstu i marketingu. Primena klasifikacije je široka i u rešavanju problema u drugim oblastima. Da bi bolje objasnili klasifikaciju iskoristićemo problem klasifikacije kreditnih zahteva kao visoko ili nisko rizične. Banka za svakog svog klijenta poseduje informacije o mesečnom prihodu, godinama starosti, proizvodima koje koristi i sl. Nakon što klijent podnese zahtev za kredit svrstava se u jednu od dve kategorije (kao visoko ili nisko rizičan) na osnovu informacija koje banka poseduje o klijentu i sličnosti tih informacija sa klijentima koji su se ranije pokazali kao pripadnici odgovarajuće kategorije.

Klasifikacija nekog objekta se zasniva na pronalaženju sličnosti sa unapred određenim objektima koji su pripadnici različitih klasa. Sličnost dva objekta se određuje analizom njihovih karakteristika pri tome se svaki objekat svrstava u neku od klasa sa određenom tačnošću. Objekti koji treba da budu klasifikovani su najčešće predstavljeni kao n-torke u tabeli ili datoteci. Zadatak je, na osnovu karakteristika objekata čija klasifikacija je unapred poznata, napraviti model na osnovu koga će se vršiti klasifikacija novih objekata. Broj klasa je unapred poznat i ograničen. Redosled klasa nije bitan.

Proces klasifikacije se sastoji iz dve faze kao što je prikazano na slici 2. U prvoj fazi gradi se model na osnovu karakteristika objekata čija klasifikacija je poznata. Podaci za izgradnju modela se najčešće nalaze u tabelama gde je svaka instanca[1] objekta X predstavljena (n+1)-dimenzionim vektorom atributa $X = (x_1, x_2, ..., x_n, x)$. Pri tome svaka instanca uzima samo jednu vrednost atributa klase x. Atribut klase može da ima konačan broj diskretnih vrednosti koje nisu uređene. U ovoj fazi klasifikacioni algoritam uči na osnovu poznatih klasifikacija tj. na osnovu instanci objekata čija je klasifikacije poznata. Na osnovu vrednosti njihovih atributa i atributa klase, gradi se skup pravila na osnovu kojih će se kasnije vršiti klasifikacija. Metode klasifikacije su zasnovane na drvetima odlučivanja, pravilima klasifikacije, Bajesovim klasifikatorima, neuronskim mrežama, itd. U daljem tekstu će biti detaljnije opisane metode koje se koriste u radu i koje su zasnovane na drvetima odlučivanja (odeljak 3.1) i neuronskim mrežama (odeljak 3.2).

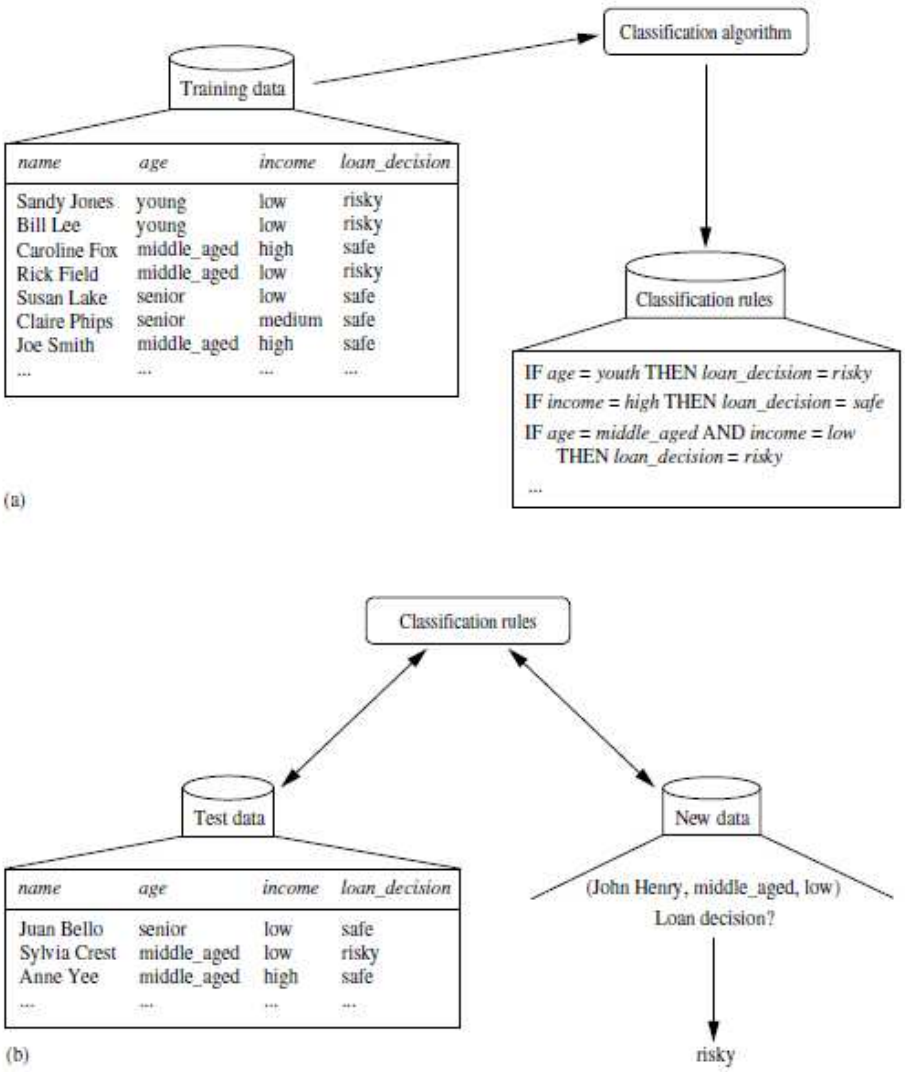
U drugoj fazi model se testira tj. procenjuje se njegova tačnost. Pod tačnošću podrazumevamo procenat instanci koje su tačno klasifikovane. Vrednost atributa klase svake testne instance poredi se sa vrednošću atributa klase koja je određena na osnovu modela. Za testiranje modela koriste se instance koje nisu korišćene u fazi učenja.

Testne instance se najčešće izdvajaju slučajnim izborom, pre faze učenja, od instanci čija je klasifikacija poznata. Ako je tačnost modela zadovoljavajuća onda se dalje koristi u klasfikaciji objekata čija vrednost atributa klase nije poznata.

Metode klasifikacije se porede i ocenjuju na osnovu sledećih kriterijuma:

- Tačnost – sposobnost klasifikatora da tačno klasifikuje instancu nepoznate vrednosti atributa klase
- Brzina – broj operacija koje se izvrše pri konstrukciji i primeni klasifikatora
- Robustnost – preciznost klasifikatora kada se primeni na podacima sa šumom ili podacima kojima nedostaju vrednosti nekih atributa
- Skalabilnost – efikasnost metode ako se primenjuje na velike količine podataka
- Interpretabilnost – jasan prikaz i razumevanje rezultata

[1] Pod instancom podrazumevamo n-torku $X = (x_1, x_2, ..., x_n, x)$, gde su $x_1, x_2, ..., x_n$ atributi a x klasni atribut

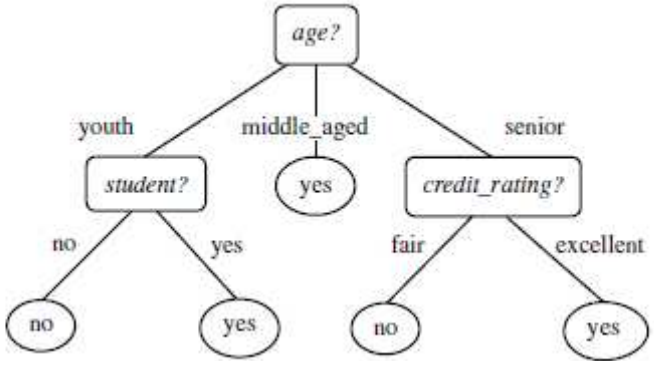


Slika 2. Klasifikacija klijenata prema kreditnom riziku

3.1 Klasifikacija pomoću drveta odlučivanja

Klasifikacija pomoću drveta odlučivanje je jednostavna i brza što je i razlog zbog čega se široko primenjuje u raznim oblastima kao što su medicina, finansije, marketing, astronomija, itd. Konstrukcija drveta odlučivanja ne podrazumeva nikakvo znanje iz

oblasti kojoj pripada problem koji se rešava. Druga dobra strana primene drveta odlučivanja je što reprezentacija rezultata u formi drveta intuitivana i lako shvatljiva.



Slika 3. Drvo odlučivanja

Na slici 3. je dat primer drveta odlučivanja koje na osnovu atributa godine, student i kreditne sposobnosti daje odgovor na pitanje da li će klijent kupiti određeni proizvod ili ne. Svaki unutrašni čvor drveta je testiranje nekog atributa, svaka grana rezlultat tog testa i svaki list jedna od mogućih vrednosti atributa klase . Za instancu čija vrednost atributa klase nije poznata, atributi se testiraju kroz drvo od korena ka listovima koji sadrže vrednost klasnog atributa za tu instancu.

Osnovni algoritam za konstruisanje drveta odlučivanja je:

Ulaz:

- D - skup instanci i njima pridruženih atributa klase predstavljenih n-torkama
- Lista atributa
- Procedura *izbor_atributa* koja bira najbolji kriterijum za podelu tj. bira atribut koji najbolje deli n-torke u klase

Izlaz:

- drvo odlučivanja

Algoritam:

(1) Konstruiši koren N

(2) Ako n-torke iz D pripadaju istoj klasi C onda vrati N kao list drveta sa klasom C

(3) Ako je lista atributa prazna onda vrati N kao list drveta sa najbrojnijom klasom u D

(4) Primeni proceduru *izbor_atributa* pri izboru najboljeg kriterijuma A za podelu u čvoru N

(5) Obeleži čvor N kriterijumom podele

(6)

a) Ako atribut A uzima diskretan skup vrednosti i algoritam nije ograničen samo na generisanje kreiranje binarnog drveta konstruiši granu za svaku od vrednosti a_j koje A uzima

b) Ako A ne uzima diskretan skup vrednosti procedura *izbor_atributa* vraća tačku_podele. Na osnovu nje se konstruišu dve grane gde jedna sadrži sve vrednosti za koje je $a_j \leq$ tačka_podele, a druga vrednosti za koje je $a_j >$ tačka_podele

c) Ako A uzima diskretan skup vrednosti i algoritam podržava samo generisanje binarnog drveta. Procedura *izbor_atributa* vraća podskup S_a vrednosti atributa A i formira dve grane $a_j \in S_a$ i $a_j \notin S_a$

(7) Ukloni atribut A iz liste atributa

(8) Za svaki podskup D_j n-torki iz D ako je D_j prazan pridruži list čvoru N sa najzastupljenijom klasom u D. Inače, pridruži drvo generisano za Dj i *lista_atributa*

(9) Vrati drvo

Ako je n broj atributa koji opisuju n-torke u D, $|D|$ broj n-torki u D onda je broj operacija potrebnih za konstrukciju drveta n * $|D|$ * log $|D|$.

Ovo je primer osnovnog algoritma za konstrukciju drveta. Generalno, algoritmi se međusobno razlikuju po mehanizmima koje koriste za potkresivanje drveta i po proceduri za izbor sledećeg atributa za podelu. Osnovni algoritam koji je prethodno opisan zahteva jedan prolaz kroz n-torke za svaki nivo drveta što ima za posledicu dugu fazu učenja i potencijalni manjak memorije u slučaju rada sa velikim bazama.

Tokom konstukcije drveta mera koja se primenjuje u izboru atributa određuje kriterijum podele odnosno atribut koji najbolje deli n-torke u klase. U idealnom slučaju svaka particija bi trebalo samo da sadrži n-torke koje pripadaju istoj klasi. U praksi najbolji kriterijum podele je onaj koji je najbliži idealnom. Primenjena mera rangira atribute i atribut koji je najbolje ocenjen se bira za sledeći kriterijum podele. Ako je

algoritam ograničen samo na konsturkciju binarnog drveta procedura vraća i tačku podele i podskupove podele.

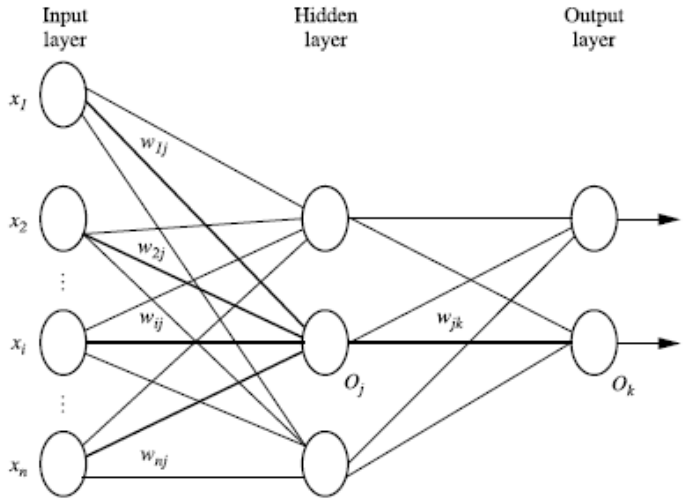
Najčešće korišćene mere su informaciona dobit, odnos dobiti i Gini indeks. Algoritam ID3 koristi entropiju kao meru za izbor atributa. Za svaki atribut A procenjuje se entropija i bira atribut čija je entropija minimalna. Nedostatak ID3 algoritma je to što preferira atribute sa velikim brojem vrednosti. Njegov naslednik je C4.5 algoritma koji kao meru koristi odnos dobiti. Za razliku od ID3 koji rangira atribute na osnovu količine informacija potrebne da se klasifikacija završi ova mera uzima u obzir i ukupan broj n-torki u D. Problem sa ovim algoritmom je to što je često jedna particija mnogo veća od druge što dovodi do nebalansiranosti.
CART algoritam koristi Ginijev indeks. Neka postoji K klasa i ako je verovatnoća i-te klase p_i, vrednost Ginijevog indeksa je $1-\sum_{i=1}^{K} p_i^2$. Najmanja vrednost koju ovaj zbir može da ima je 0, i u tom slučaju sve instance pripadaju istoj klasi. Kada su instance ravnomerno raspoređene u klase vrednost Ginijevog indeksa je najveća i iznosi $1-1/K$. U svakom koraku bira se atribut koji najviše umanjuje vrednost indeksa.

Svi navedeni algoritmi imaju svoje prednosti i mane tako da ne može da se kaže koji je najbolji u svim slučajevima. U zavisnosti od podataka sa kojima se radi, isti algoritam može da pokaže različite karakteristike.

Drugi problem sa kojim se srećemo pri konstukciji drveta je generisanje grana koje su rezultat podataka sa šumom ili vrednosti van granica. Odsecanjem ovakvih grana dobija se drvo koje je manje složeno, a uz to bolje i brže klasifikuje testne podatke. Metode za potkresivanje drveta uglavnom koriste statističke metode za procenu najmanje pouzdanih grana. Kod metode predpotkresivanja drvo se i ne podseca nego se u fazi konstukcije drveta procenjuje po kojim granama ne treba dalje vršiti deobu, a zatim se te grane zamenjuju listom. Drugi pristup, koji zahteva više izračunavanja, je da se drvo generiše u potpunosti a zatim se najmanje pouzdane grane podsecaju odnosno zamenjuju listom koji sadrži najzastupljeniju klasu.

3.2 Neuronske mreže

Neuronske mreže imaju široku primenu u u medicinskim istraživanjima, prepoznavanju rukopisa i govora, finansijama i marketingu. Koriste se i za klasifikaciju i za predviđanje U ovom tekstu se o neuronskim mrežama govori sa aspekta klasifikacije podataka. U odnosu na drvo odlučivanja neuronske mreže su se pokazale efikasnije u slučaju nepotpunih podataka, nedostajućih vrednosti atributa i podataka sa šumom. Dodatna prednost je preciznija klasifikacija podataka za koje nisu postojali slični podaci u fazi učenja. Neuronske mreže minimalizuju ljudski faktor jer mogu biti u velikoj meri automatizovane. Paralelizacija je još jedan od faktora zbog kojih su mreže popularne jer u radu sa velikim bazama podataka mogućnost paralelizacije daje veliku prednost u odnosu na pristupe koji ne podržavaju paralelizaciju obrade.



Slika 4. Višeslojna mreža sa tokom podataka unapred

Grubo govoreći, neuronska mreža je mreža ulazno/izlaznih jedinica povezanih težinskim koeficijentima. Tokom faze učenja mreža podešava se težinske koeficijente tako da greška bude minimalna a tačnost klasifikacije maksimalna. U testnoj fazi se proverava tačnost dobijene mreže i ukoliko su rezultati zadovoljavajući mreža se dalje primenjuje za klasifikovanje podataka. U upotrebi su razni tipovi mreža i algoritama a najpopularniji je algoritam sa propagacijom unazad. Ovaj algoritam je zasnovan na višeslojnoj mreži sa tokom podataka unapred (Slika 4.). Ovakav tip mreže sastoji se od

jednog ulaznog sloja, jednog ili više unutrašnih slojeva i izlaznog sloja. Podaci putuju samo u jednom pravcu od ulaznog ka izlaznom sloju. Ulazni sloj prihvata podatke od atributa n-torki, a unutrašnji slojevi i izlazni sloj od prethodnog sloja. Veze medju slojevima su iskazane težinskim koeficijentima w_{ij} koji se koriguju posle svakog prolaza poređenjem tačne i dodeljene vrednosti atributa klase. Nakon ocene greške koeficijenti se koriguju, pri čemu se korekcija vrši u suprotnom smeru, tj. od izlaznog sloja ka ulaznom.

Iako su se neuronske mreže pokazale superiornije u odnosu na druge metode dosta su kritikovane zbog apstraktnosti i male interpretabilnosti rezultata. Ljudskom umu je teško da uoči vezu između težinskih koeficijenata i skrivenih slojeva neuronske mreže. Međutim, zbog ostalih dobrih karakterisitka mreža, u poslednje vreme se radi na razvoju tehnika koje će generisati pravila od neuonskih mreža što će ih učiniti razumljivijim i manje apstraktim.

.

Drugi nedostatak neuronskih mreža je topologija mreže odnosno nepostojanje jasnih pravila u kojim sličajevima je koja topologija mreže najpogodnija. Za svaku primenu se metodom pokušaja i greške određuje najbolja topologija mreže. Pri tome se pod topologijom mreže podrazumeva broj jedinica ulaznog sloja, broj unutrašnjih slojeva i jedinica unutar njih i broj jedinica izlaznog sloja. Ako tačnost nije zadovoljavajuća, nakon faze testiranja, postupak se ponavlja.

Faza učenja je uobičajeno duga i u potpunosti je zavisna od količine i kvaliteta ulaznih podataka tako da su dobar odabir podataka, kvalitetna priprema i normalizacija atributa ključni za uspeh ove metode.

U nastavku je dat algoritam klasifikacije pomoću neuronskih mreža sa propagacijom unazad:

Ulaz:

- D skup objekata odnosno n-torki i njima pridruženih atributa klase
- l stepen učenja
- Topologija mreže

Izlaz:

- neuronska mreža

Algoritam:

(1) inicijalizovati sve težinske koeficijente w_{ij} i odstupanja θ_j
(2) sve dok uslov_zaustavljanja nije zadovoljen{
(3) za svaku n-torku X iz D {
(4) za svaku jedinicu ulaznog sloja j {
izjednači ulaz i izlaz
$O_j = I_j$
(5) za svaki jedinicu unutrašnjeg ili izlaznog sloja j{
izračunaj ulaz i izlaz u odnosu na prethodni sloj i
$I_j = \sum_i w_{ij} O_i + \theta_j$
$O_j = \frac{1}{1+e^{-I_j}}$ }
(6) za svaku jedinicu izlaznog sloja
izračunaj grešku
$E_{rr_j} = O_j(1-O_j)(T_j - O_j)$
(7) za svaku jedinicu j unutrašnjeg sloja(od izlaza ka ulazu)
izračunaj grešku u odnosu na viši nivo k
$E_{rr_j} = O_j(1-O_j)\sum_k E_{rr_k} w_{jk}$
(8) za svaki koeficijent u mreži w_{ij} {
izračunaj povećanje inkrement i koriguj
$\Delta w_{ij} = (l) E_{rr_j} O_j$
$w_{ij} = w_{ij} + \Delta w_{ij}$ }
(9) za svako odstupanje θ_j u mreži {
izračunaj povećanje i koriguj
$\Delta\theta_j = (l) E_{rr_j}$
$\theta_j = \theta_j + \Delta\theta_j$ }
} }

Kod algoritma sa propagacijom unazad u prvom koraku se težinski koeficijenti w_{ij} i odstupanja θ_j inicijalizuju na male slučajne vrednosti (uobičajeno izmedju -1 i 1). Zatim se iterativno obradjuje svaka n-torka i porede predviđene i tačne vrednosti klasnog atributa. Za svaku ulaznu jedinicu izlaz O_j je jednak ulazu I_j, dok se ulaz i izlaz za jedinice unutrašnjih i izlaznog sloja izračunavaju na sledeći način:

- ulaz kao linearna kombinacija izlaza iz prethodnog sloja pomnoženih sa težinskim koeficijentima $I_j = \sum_i w_{ij} O_i + \theta_j$
- izlaz pomoću funkcije $O_j = 1/(1+e^{-I_j})$. Za izlazni sloj ova funkcija daje pocenu vrednosti atributa klase.

Greška se računa od izlaznog ka ulaznom sloju, pri čemu se za izlazni sloj računa po formuli $E_{rr_j} = O_j(1-O_j)(T_j - O_j)$ gde je O_j pocenjena vrednost atributa klase a T_j tačna vrednost atributa klase. Faza učenja se završava kada razlika između težinskih koeficijenata prethodne i tekuće iteracije dostigne minimum. Stepen učenja l je konstanta izmedju 0 i 1 koja sprečava da se učenje završi kad je razlika koeficijenata dostigla lokalni minimum. Uobičajeno je da se l postavlja na 1/t,gde je t do tada izvršen broj iteracija.Uslov zaustavljanja je ili urađen predviđeni broj iteracija, ili ako je Δw_{ij} ispod određene granice, ili ukoliko je procenat nekorektno klasifikovanih n-torki ispod određene granice.

4 Procena rizika u bankarstvu

Prikazane metode za istraživanje podataka mogu da se primene u velikom broju različitih istraživanja u oblasti bankarstva [2]. Jedno od istraživanja sprovedeno je u Banci Poštanskoj štedionici sa ciljem da se identifikuje profil viskorizičnog klijenta u poslovanju sa dinarskim tekućim računima građana. Dobijeni rezultati bi se koristili pri obradi zahteva klijenata za odobravanjem sredstava banke sa ciljem da se smanji rizik u poslovanju banke eventualno količina odobrenih a kasnije nenaplativih sredstava svede na minimum. Formirani profil potencijalnog klijenta bi se, pre odobravanja sredstava banke klijentu na korišćenje, uporedio sa profilom visokorizičnih (i niskorizičnih) klijenata u bazi klijenata banke. Potencijalno visokorizičnim klijentima se ne bi odobrila sredstva banke (ili bi se odobrila pod vrlo restriktivnim uslovima), Profil klijenta bi bio napravljen na osnovu istorijskih podataka o klijentima sa kojima banka raspolaže. Podaci o klijentima obuhvataju podatke o stanju na tekućim računima i lične podatke o klijentu. Za formiranje željenog profila klijenta mogu da se iskoriste metode istraživanja podataka, od kojih se kao najpogodnija pokazala klasifikacija. U daljem tekstu je prikazan primer primene klasifikacije u proceni klijenta koji je rađen u Banci Poštanska štedionica A.D.

4.1 Primena klasifikacije u proceni klijenata

Istraživanje koje je rađeno je imalo za cilj da da klasifikuje potencijalnog klijenta kao niskorizičnog ili visokorizičnog u periodu od narednih 12 meseci. Posedovanje ovakve informacije bi omogućilo banci da pravovremeno reaguje i ne dozvoli klijentima sa visokorizičnim profilom da koriste sredstva banke (kredite, kreditne kartice, dozvoljeni

minus, čekove itd.) za koja se pretpostavlja da neće moći da budu vraćena u narednom periodu.

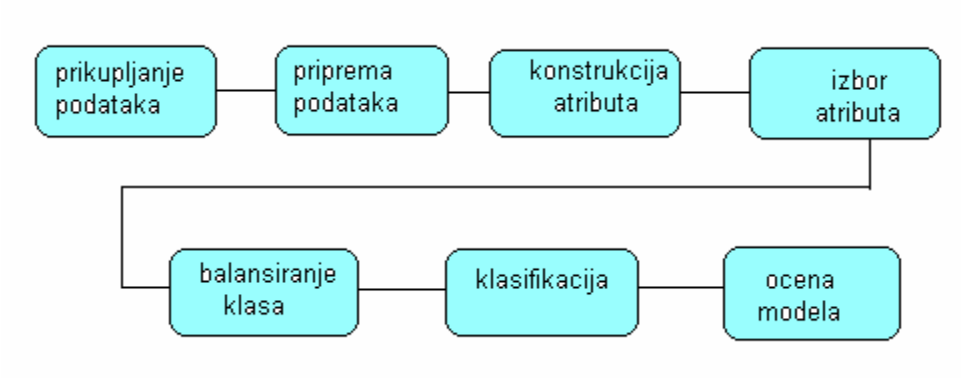
Model za ovu vrstu klasifikacije može da se napravi na osnovu raspoloživih podataka o ponašanju klijenata banke u prethodnom periodu. Banka PŠ ima 5 miliona otvorenih tekućih računa, od kojih je 1.5 miliona računa stalno aktivno. Podaci o tekućim računima se nalaze na *mainframe* računaru IBM z10 series, dok se kao RSUBP koristi IBM DB2. Postoji verzija baze koja sadrži tekuće stanje, dok se arhivski podaci izmeštaju u druge (takođe DB2 podržane) baze, ili se arhiviraju na magnetnim trakama ako su stariji od određenog broja godina.

Atributi koji su izabrani za opis klijenata izdvojeni su iz pet DB2 tabela. Odabir atributa je obavljen u skladu sa onim što se nalazi u podacima i saglasan je sa izborom atributa u sličnim istraživanjima. Izdvojeni su najčešće korišćeni atributi opisani u radovima [2] i [3] i još nekoliko dodatnih atributa sa kojima su vršeni eksperimenti.

Skup izdvojenih atributa bi trebalo da ubrza identifikovanje potencijalno rizičnih klijenata. Vektori atributa klijenata koji pripadaju istoj kategoriji rizika će imati slične ili identične vrednosti atributa.

4.2 Proces konstrukcije modela

Cilj modela je klasifikacija klijenata u klase *visokorizičan* i *niskorizičan* na osnovu istorijskih i demografskih podataka. Konstrukcija modela se sastoji iz nekoliko koraka koji uključuju prikupljanje i pripremu podataka, konstrukciju atributa, izbor atributa, balansiranje klasa, klasifikaciju i ocenu modela. Na slici 5. dat je grafički prikaz koraka koji su detaljnije opisani u narednim odeljcima.



Slika 5. Konstrukcija modela

4.3 Prikupljanje podataka

Prvi korak u konstrukciji modela je prikupljanje i konvertovanje podataka u oblik koji je prihvatljiv za korišćeni alat.

Izdvojeni su podaci o 3259 tekućih računa koji pripadaju visokorizičnim klijentima. Navedeni klijenti su u toku 2009. godine prestali da izmiruju svoja zaduženja. Politika Banke je da se klijenti koji kasne sa izmirivanjem dugovanja najpre opomenu (usmeno i pismeno). Ukoliko klijent nakon opomena ne izmiri zaduženja podaci o računu koji ima neizmirene obaveze se upućuju u pravnu službu, koja sudskim putem naplaćuje dugovanja od vlasnika pomenutih računa. Ovakav ishod je nepoželjan i broj ovakvih klijenata treba svesti na minimum tj. sprečiti ovakve klijente da se zadužuju van raspoloživih sredstava.

U drugoj grupi klijenata su svi ostali klijenti, odnosno, klijenti koji imaju trenutno aktivne tekući račune u štedionici. Za formiranje modela izdvojeno je 5000 ovakvih računa. Birani su računi koji su imali uplate ili isplate u toku 2007. ili 2008. godine. Analizom podataka klijenata koji su tuženi tokom 2009. godine u odnosu na trenutno aktivne možemo videti koje su akcije prethodile negativnom ishodu odnosno koje su karakteristike visokorizičnog bankarskog klijenta. U budućnosti, takva informacija će nam omogućiti da analiziramo podatke klijenta pre odobravanja sredstava van raspoloživih.

Odlučili smo se da 5000 aktivnih tekućih računa bude predstavik celog skupa niskorizičnih klijenata iz sledećih razloga . Udeo visokorizičnih klijenata procentualno je oko 1% u odnosu na ostale klijente. Ako bi sve ostale klijente smatrali niskorizičnim i nad takvim podacima primenili algoritme istraživanja podataka suočili bi se sa jednim od sledeća dva problema. Dobijeni model bi mogao da ima jako veliku tačnost (preko 99 procenata) ali bi mogao i da loše klasifikuje visokorizične klijente jer se tačnost modela ocenjuje kao procenata tačno klasifikovanih (visokorizičnih i niskorizičnih) klijenata u odnosu na ukupan broj klijenata. Drugi problem koji bi mogao da se javi je nedostatak memorije[2]. Primena algoritama klasifikacije na ovako velikoj bazi je izuzetno zahtevan proces u pogledu vremena i resursa. Ukoliko tačnost modela ne bude zadovoljavajuća, na osnovu [1], treba uzeti veći skup niskorizičnih klijenata.

Kako se u konstrukciji modela koriste istorijski podaci klijenata bitno pitanje je i kolika količina takvih podataka je dovoljna? Banke čuvaju veliku količinu podataka o klijentima duži niz godina. Uzimanjem svih tih podataka prepoteretio bi se model suvišnim podacima ili čak podacima koji nisu relevantni. Na primer, ako za klijenta koji je račun otvorio pre 10 godina razmatramo sve transakcije koje su se desile, vrlo je moguće da ćemo pogrešiti jer se u međuvremenu stanje na tržištu promenilio i podaci od pre 7 ili 10 godina više nisu korisni. Na osnovu [1] za sve klijentske aplikacije 2-3 godine istorije su dovoljne iako se i podaci o samom početku rada sa klijentom vrlo značajni. U konstrukciji ovog modela smo koristili dvogodišnju istoriju rada tekućih računa kao i podatke o otvaranju tih računa.

Generalno, kod izbora podataka ne bi trebalo da žurimo sa odbacivanjem tj. trebalo bi da koristimo što više podataka, uz uslov da se u toku istraživanja odbace nebitni podaci odnosno da se na osnovu rezultata primene algoritama odredi koji podaci jesu a koji nisu bitni. Krajni rezultati se zasnivaju na nekoliko ključnih podataka koji se uglavnom dobijaju kombinacijom iz drugih i koji nisu očigledni na početku rada. Model je zasnovan na podacima koji se uobičajeno koriste u bankarstvu za kategorizaciju klijenata i dele se u četri grupe:

1. demografske
2. relacijske

[2] Pri radu sa WEKA alatom koji je korišćen u ovom istraživanju.

3. transakcione
4. dodatne

Demografski podaci su lični podaci klijenta. To su npr.godine, pol, obrazovanje, adresa itd. Ovi podaci su stabilni tokom vremena ali mogu da budu promenljivog kvaliteta u zavisnosti od procedura i discipline evidentiranja.

Relacijski podaci se odnose na dužinu i trajanje veze sa bankom. Ovi podaci uključuju koliko dugo je data osoba klijent, koje proizvode koristi, statuse neki bitnih događaja (žalbi, prekoračenja, upozorenja itd.).

Transakcioni podaci se odnose na broj transakcija datog klijenta kao i promet pri tim transakcijama. Obično se uzimaju zbirni podaci za mesec, kvartal ili godinu.

Dodatni podaci sadrže sve što je potrebno ili što se smatra bitnim za bliže određivanje veze sa datim klijentom, kao na primer, reakciju klijenta na prethodnu marketinšku kampanju.

Kao što smo već naveli podaci se nalaze u pet tabela, čiju su kratki opisi dati u daljem tekstu.

TABELA 1- osnovna tabela tekućih računa ima oko 6,5 miliona slogova i oko 200 atributa koji opisuju svaki od tekućih računa. Sadrži podatke o broju tekućeg računa, trenutno stanje na računu, zadnju uplatu, itd. Takodje, sadrže i većinu demografskih podataka koji su korišćeni (matični broj, adresa, pošta i sl.), ali i neke relacijske podatke (datum otvaranja računa, statuse upozorenja itd.)

TABELA 2- sadrži podatke o firmama koji su uplatioci sredstava na tekuće račune Banke Postanske stedionice a.d.Tabela sadrži interne šifre firmi, naziv, adresu, poštanski broj, šifru delatnosti, broj zaposlenih itd.

TABELA 3- sadrži podatke o promenama na svim tekućim računima u toku 2007. godine. Tabela sadrži oko 90 miliona slogova, koji se sastoje od broja tekućeg računa, datuma i vrste promene (uplata i isplata), šifre promene i iznosa. Iz ove tabele izdvojeni su sledeći podaci za analizu klijenta:

- Uplate i isplate
- Broj obavljenih transakcija
- Minimalan i maksimalan iznos transakcija
- Ostvarni profit
- Zatezne kamate
- Deblokade
- Krediti
- Ovlašćena lica
- Posedovanje i upotreba debitnih i kreditnih platnih kartica
- Upotreba bankomata
- Internet plaćanja
- Upotreba šaltera i bankomata drugih banaka

TABELA 4- istorijska tabela za 2008.godinu, sa atributima koji su identični atributima u TABELI 3.

TABELA 5- sadrži podatke o svim poštama u Srbiji koji uključuju naziv, broj pošte, radno vreme, adresu, grad i region u kome se nalazi pošta.

4.3.1 Izbor atributa

U ovom odeljku su prikazani atributi koji su izdvojeni za konstruisanje modela. Izdvojeno je 55 atribut iz pet tabela i podeljeno u četri grupe. Kao što je spomenuto ranije, atributi izdvojeni za opis klijenta su korišćeni u radovima koji se bave istom problematikom. Uz to su izdvojeni i dodatni atributi na osnovu analize preostalih podataka i poznavanja date oblasti. Od izdvojenih atributa izabran je podskup koji najbolje opisuje klijenta i taj podskup je korišćen u fazama pravljenja modela ('treniranje') i njegovoj proveri ('testiranje').

Demografski podaci
Većinu demografski podataka samo dobili na osnovu jedinstvenog matičnog broja građana koji ima 13 cifara u formi DDMMGGGRRBBBK. Izdvojeni su sledeći atributi:

POL – u matičnom broju određuju pozicije od 10 do 12

BBB	pol
000-499	muški
500-999	ženski

RR - region rođenja određuju pozicije 8 i 9 matičnog broja

RR	region rođenja
00-99	stranci
10-19	BiH
20-29	Crna Gora
30-39	Hrvatska
41-49	Makedonija
50-59	Slovenija
70-79	Centralna Srbija
80-89	Vojvodina
90-99	Kosovo i Metohija

GGG	godina rođenja se nalazi na pozicijama od 5 do 7
RADNI_STATUS	s obzirom su penzioneri veliki procenata klijenata Banke Poštanske Štedionice a.d. na osnovu šifara uplatioca sredstava na tekući račun klijenti su podeljeni u dve grupe: penzionere i zaposlene.
SIFRADELAT	odnosi se na užu oblast poslovanje firme koje je uplatilac sredstava na tekući račun klijenta npr. sve firme iz oblasti trgovine imaju istu šifru delatnosti. Ukupno je izdvojeno 42 različite šifre.
REGION	na osnovu broja pošte šire smo odredili lokaciju gde klijent živi.

Relacijski podaci

BRDANODOTV	u TABELI 1 se evidentira datum otvaranja tekućeg računa. Broj dana od otvaranja do 01.01.2009 smo izračunali kao razliku dva datuma u danima.
NEDOZKAMUK	ukupna suma naplaćene kamate za nedozvoljeno prekoračenja tokom 2007. i 2008.godine.
NEDOZKAM08	ukupna suma naplaćene kamate za nedozvoljena prekoračena u 2008. godini.
DEBLOKADUK	broj deblokada tekućeg računa tokom 2007. i 2008. godine.
DEBLOKADE08	broj deblokada tekućeg računa tokom 2008 .godine.

IMADINUUK	indikator da li korisnik računa imao DINA karticu ili obavljao plaćanja DINA karticom tokom 2007. i 2008. godine.
IMADINU08	indikator da li korisnik računa imao DINA karticu ili obavljao plaćanja DINA karticom tokom 2008. godine.
IMAVISUUK	indikator da li korisnik računa imao VISA ili obavljao plaćanja VISA karticom tokom 2007. i 2008. godine.
IMAVISU08	indikator da li korisnik računa imao VISA ili obavljao plaćanja VISA karticom tokom 2008. godine.
IMAMASTERUK	indikator da li korisnik računa imao MASTER ili obavljao plaćanja MASTER karticom tokom 2007. i 2008. godine.
IMAMASTER08	indikator da li korisnik računa imao MASTER ili obavljao plaćanja MASTER karticom tokom 2008. godine.
IMAMAESTROUK	indikator da li korisnik računa imao *MAESTRO* ili obavljao plaćanja *MAESTRO* karticom tokom 2007. i 2008. godine.
IMAMAESTRO08	indikator da li korisnik računa imao *MAESTRO* ili obavljao plaćanja *MAESTRO* karticom tokom 2008. godine.
KREDIT UK	indikator da li je vlasnik tekućeg računa podigao ili otplaćivao kredit tokom 2007. i 2008. godine.
KREDIT08	indikator da li je vlasnik tekućeg računa podigao ili otplaćivao kredit tokom 2008. godine.

Transakcioni podaci

UPUK	zbir svi uplata na dati tekući račun u toku 2007. i 2008. godine, iz TABELA3 i TABELA4.
ISUK	zbir svih isplata sa datog tekućeg računa iz TABELA 3 i TABELA4.
UPUK08	ukupan iznos uplata u 2008. godini, iz TABELA4.
ISUK08	ukupan iznos isplata u 2008. godini.
AVGUK	prosečan iznos transakcije u toku 2007. i 2008. godine, dobijen deljenjem ukupnog iznosa sa brojem transakcija u ovom periodu.
AVG08	prosečan iznos transakcije u 2008. godini.
AVGUPLUK	prosečan iznos transakcije uplate u 2007. i 2008. godini.
AVGUPL08	posečan iznos transakcije uplate u 2008. godini.
AVGISPLUK	prosečan iznos transakcije isplate u 2007. i 2008. godini.
AVGISPL08	posečan iznos transakcije isplate u 2008. godini.
MAXUPLUK	maksimalna transakcija uplate u 2007. i 2008. godini.
MAXUPL08	maksimalna transakcija uplate u 2008.godini.
MAXISPUK	maksimalana transakcija isplate u 2007. i 2008. godini.
MAXISP08	maksimalana transakcija isplate u 2008. godini.
MINUPLUK	minimalna transakcija uplate u toku 2007. i 2008. godine.
MINUPL08	(najmanja)minimalna transakcija uplate u 2008. godini.
MINISPUK	minimalana transakcija isplate u 2007. i 2008. godini.
MINISP08	minimalna transakcija isplate u 2008. godini.

BRTRUK	ukupan broj transakcija u 2007. i 2008. godini.
BRTR08	broj transakcija u 2008. godini.
BRTRUPLUK	broj transakcija uplate u 2007. i 2008. godini.
BRTRUPL08	broj transakcija uplate u 2008. godini.
BRTRISPUK	broj transakcija isplate u 2007. i 2008. godini.
BRTRISP08	broj transakcija isplate u 2008. godini.

Dodatni podaci

PROFITUK	ostvarena zarada od datog klijenta tokom 2007. i 2008. godine. Zarada podrazumeva naplaćene naknade za održavanje računa, izdavanje novih kartica, obavljene transakcije, kamate dozvoljenog prekoračenja itd.
OVLLICEUK	indikator da li je klijent imao ovlašćena lica u 2007. i 2008. godini.
OVLLICE08	indikator da li je klijent imao ovlašćena lica u 2008. godini
BRTRBANUK	broj transakcija na bankomatima tokom 2007 i 2008. godine.
BRTRBAN08	broj transakcija na bankomatima tokom 2008. godine.
BRTRINTUK	indikator da li je korisnik tokom 2007. i 2008. plaćao preko Interneta.
BRTRINT08	indikator da li je korisnik tokom 2008. godine plaćao preko Interneta.
BRTRDRBANKUK	indikator da li je klijent koristio bankomate ili šaltere drugih banaka u toku 2007. i 2008. godine.
BRTRDRBANK08	indikator da li je klijent koristio bankomate ili šaltere drugih banaka u toku 2008. godine.

4.3.2 Konstrukcija atributa

Nakon što su definisani podaci koje ćemo koristiti u konstrukciji klasifikatora neophodno je da te podatke integrišemo u jednu tabelu. Kao što smo već naveli podaci se nalaze u pet DB2 tabela, međutim oblik u kome se nalaze nije prihvatljiv za algoritme istraživanja koji se koriste u alatu pomoći koga je istraživanje vršeno. TABELA1 sadrži osnovne podatke o klijentu mežu kojima je jako malo istorijskih podataka o ponašanju klijenta. Istorijske promene na računu su zabeležene u tabelama 3 i 4. Te promene su evidentirane kao uređena petorka (broj računa, datum promene, vrsta promene, sifra promene, iznos). Da bi dobili zbirne promene koje se odnose na promene po karticama,

profit, deblokade, ovlašćena lica, upotrebu bankomata i interneta tokom 2007. i 2008. godine bilo je neophodno analizirati šifre promena i na osnovu analize izvući željene podatke pomoću sql upita nad tabelama 3 i 4.

U tabeli 1 za svakog klijenta navedena je šifra firme u kojoj je klijent zaposlen.Kako je broj različitih šifara veliki, šifre su grupisane prema oblasti u kojoj firma posluje odnosno oblasti u kojoj je klijent zaposlen. Kako je šifra firme suviše specifičan podatak, umesto nje se koristi odgovarajuća oblast delatnosti. Ovaj podatak se nalazi u tabeli 2, tako da smo uparivanjem tabela 2 i 1 dobili podatak o oblasti delatnosti firme u kojoj klijent radi. Zatim, u tabeli 1 je data adresa klijenta, ulica i broj pošte, a za neke klijente je dat i adresni kod. S obzirom da su ovi podaci suviše specifični da bi u istraživanju bila uočena zavisnost, korišćen je samo podatak o broju pošte koji je zamenjen brojem regiona u kome se broj pošte nalazi. Sve pošte u Srbiji su grupisane u 55 regiona u zavisnosti od lokacije na kojoj se nalaze. Ovaj podatak smo dobili iz tabele 5.

Kod klasifikacije klijenata od značaja su zbirni podaci, tj.ukupne uplate/isplate u određenom periodu, broj transakcija itd. Sve promene na računima date su u tabelama 3 i 4. TABELA3 sadrži podatke za 2007 godinu a TABELA4 za 2008.godinu. Promene na računima od interesa su posmatrane kroz dva vremenska perioda.[2]: promene koje su se desile u 2008. godini i promene koje su se desile tokom 2007. i 2008. S obzirom, da nismo sigurni kolko istorije je potrebano za ovakvu analizu uzimamo dve godine a u slučaju da naš model postiže istu tačnost upotrebom samo 2008. godine, odbacićemo podatke koji se odnose na 2007. godinu.

4.4 Priprema podataka

Kao što je već rečeno svi podaci se nalaze u DB2 tabeli i da bi bili upotrebljivi za algoritme za istraživanja moraju biti adekvatno pripremljeni. Priprema je zatevala integraciju, čišćenje i transforamciju. Integracija podataka u jednu tabelu je izvršena njihovim uparivanjem po jedinstvenom broju računa. Čišćenje integrisanih podataka je bilo relativno jednostavno jer su podaci iz izvornih tabela (koje su i inače sadržavale konzistentne podatke) pažljivo birani. Čišćenje je uključivalo sledeće procese:

Kako su podaci ciljno odabrani, nisu zahtevali mnogo čišćenja, osim navedenog:

- brisanje svih računa koji su otvoreni posle 01.01.2007.
- brisanje svih računa koji nisu imali uplate ili isplate tokom 2007. i 2008. godine
- brisanje svih računa koji imaju status da su likvidirani
- brisanje svih računa čiji je vlasnik umro
- brisanje svih računa koji imaju status da je račun zatvoren

Na ovaj način su obrisani svi podaci koji nisu od značaja. Sledeći korak je bilo rešavanje problema nedostajućih vrednosti. Kao što smo naveli u uvodnom delu rada moguće akcije u tom slučaju popunjavanje vrednosti atributa sa srednjom vrednošću atributa n-torki koje pripadaju istoj klasi, upis srednje ili najčešće vrednosti kao vrednosti nedostajućeg atributa, brisanje date n-torke, ručno popunjavanje vrednosti atritbuta koje nedostaju, ili korišćenje konstante za popunjavanje nedostajućih vrednosti. Tabela nije velika tako da smo vrednosti koje nedostaju popunili najčešćom vrednošću atributa:

- Vrednosti atributa *SIFRADELAT* je popunjena sa vrednošću koja se najčešće javlja kao vrednost atributa *SIFRADELAT*.
- Nedostajuće vrednosti atribut RR koji se odnosi zemlju rođenja je popunjen najčešćom vrednošću 7 koja znači da je klijent rođen u Srbiji.

4.5 Normalizacija

Normalizacijom se atributi skaliraju da pripadaju određenom malom opsegu vrednosti. Kao što smo spomenuli u uvodnom delu kod konstrukcije klasifikatora koji se zasnivaju na distanci kao što su neuralne mreže naročito je bitno da podaci budu normalizovani jer se na ovaj način sprečava da algoritam smatra bitnijim atribute koji imaju značajno veće numeričke vrednosti.Normalizacija z vrednosti je najpogodnija kada nisu poznati stvarni minimum i maksimum atributa ili kada su te vrednosti posledica greške ili nekakvih vanrednih stanja. Neophodna je normalizacija svih atributa koji se odnose na iznose uplate, isplate, broj transakcija i broj dana. Analizom ovih atributa primećeno je da postoje vrednosti koje mnogo odstupaju od prosečne vrednosti datog atributa. Na ovim podacima je primenjena normalizacija z vrednosti koja se zasniva na

srednjoj vrednosti atributa i standardnoj devijaciji. Atribut A sa vrednostima $x_1, x_2, ..., x_n$ se normalizuje po formuli :

$$v' = \frac{v - \bar{A}}{\sigma_A}$$

gde je $\bar{A}$ srednja vrednost atributa A,a σ_A standardna devijacija.

$$\bar{A} = \frac{x_1 + x_2 + ... + x_n}{n}$$

$$\sigma_A^2 = \frac{1}{n}\left[\sum x_i^2 - \frac{1}{n}\left(\sum x_i^2\right)^2\right]$$

4.6 Alat korišćen za klasifikaciju

WEKA (*Waikato Environment for Knowledge Analysis*) je alat za pripremu i istraživanje podataka razvijen na Waikato Univerzitetu na Novom Zelandu. Alat poseduje podršku za ceo proces istraživanja počevši od pripreme podataka preko procene i pripreme različitih algoritama kao i grafičkog prikaza podataka i rezultata istraživanja. WEKA je napisana u Javi i distribuira se pod *GNU General Public Licence*. Radi na skoro svim platformama i tesitrana je na Linux, Windows i Macintosh operativnim sistemima. Poslednja stabilna verzija WEKA-e je 3.6 i može se preuzeti sa http://www.cs.waikato.ac.nz/ml/weka/index.html.

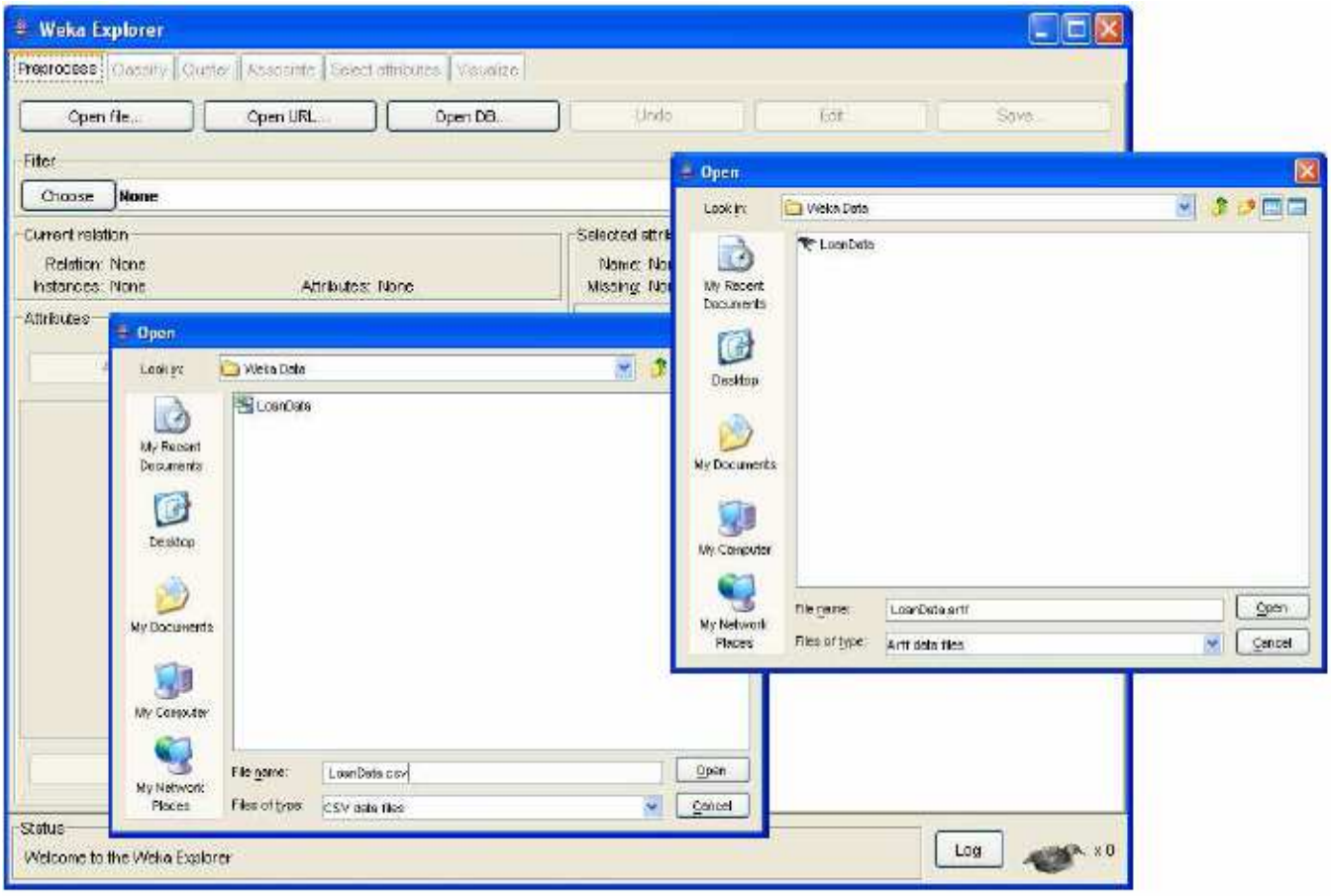
Implementirane su najčešće metode koje se javljaju u istraživanju podataka: regresija, klasifikacija, klasterovanje, asocijacija i izbor atributa. Mnoge od implementiranih metoda omogućavaju podešavanje parametara prema konkretnom problemu i podacima. Kako je upoznavanje sa podacima ključno u istraživanju *Weka* poseduje čak 49 metoda za pripremu podataka (eng. *filter*).

Podaci u WEKA-u mogu biti učitani u nekoliko različitih formata datoteka. Iako WEKA podržava razne formate datoteka prepručljivo je koristiti ARFF format datoteke koji je osnovni podržani format. Ostali podržani formati su: CVS, C4.5 ili binarni. Podaci takođe mogu da se preuzmu sa URL adrese ili iz SQL baze podatka (videti sliku 6).

Alat ima tri različita grafička korisnička okruženja (slika 7.): *Explorer*, *Knowledge Flow* i *Experimenter*.

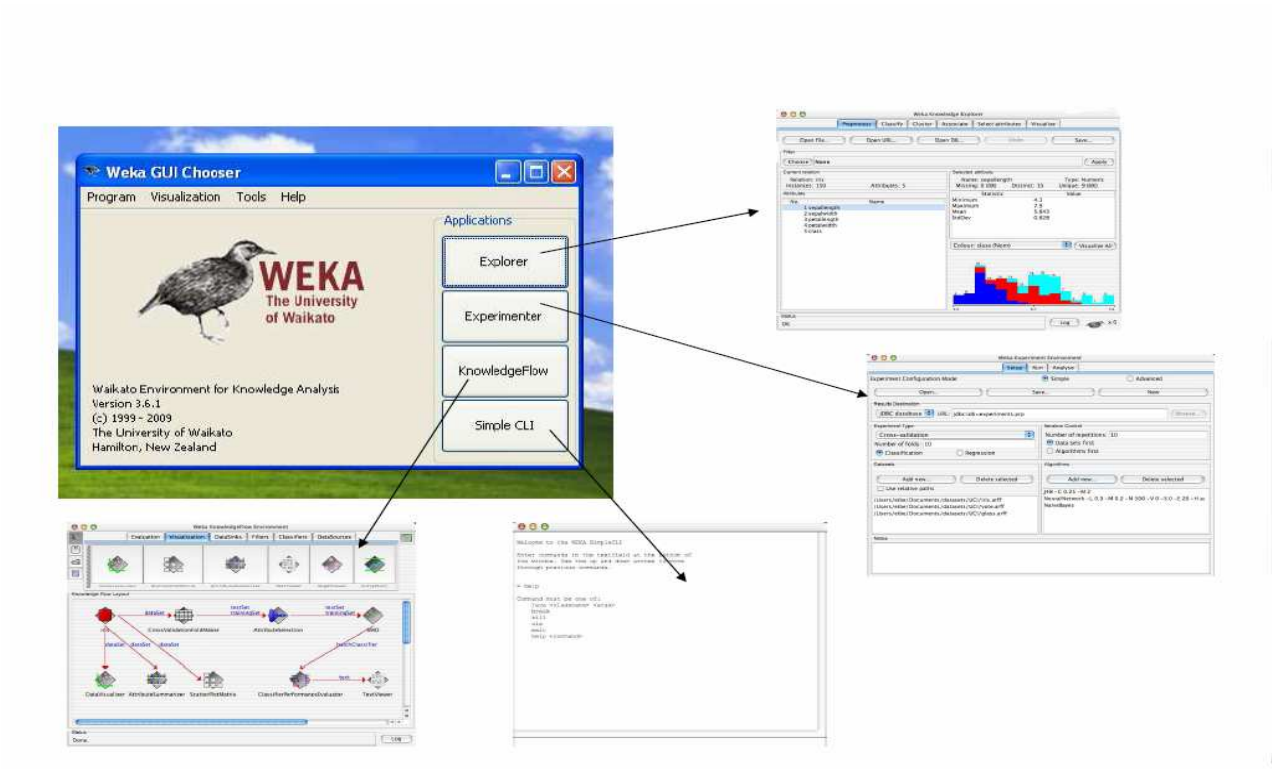
Najlakši način za korišćenje WEKA-e je *Explorer*. Ovo okruženje omogućava laku i efikasnu primenu svih funkcionalnosti alata putem izbornih menija. Nakon učitavanja podataka omogućena je priprema podataka korišćenjem *filter*-a, primena algoritama za klasifikaciju, klasterovanje ili asocijacije, kao i izbor atributa i grafički prikaz modela.

Drugo korisničko okruženje, *Knowledge Flow*, omogućava korisniku da samostalno



Slika 6. Učitavanje podataka u WEKA

definiše sekvencijalnu obradu podatka. Naime, glavni nedostatak *Explorer*-a je činjenica da sve podatke čuva u glavnoj memoriji – po otvaranju datoteke ili baze podataka ceo sadržaj se učitava odmah. Posledica je da je primena moguća samo na srednjim i malim bazama podatka. *Knowledge Flow* omogućava precizno definisanje obrade podatka povezivanjem komponenti koje predstavljaju izvore podataka, metode za pripremu, algoritme, metode evaulacije i grafički prikaz. Ukoliko algoritmi imaju mogućnosti inkrementalnog učenja, podaci će biti učitani i obrađivani sekvencijalno.



Slika 7. Prikaz grafičkih korisničkih okruženja WEKA-e

Treće okruženje, *Experimenter*, je osmišljeno da pomogne korisniku da odgovori na osnovno pitanje pri upotrebi klasifikacije i regresije: koje metode i parametri su najbolji za dati problem? Ovo okruženje omogućava poređenje različitih klasifikatora i filtera sa različitim parametrima. Naravno, poređenje može da se realiziju i kroz Explorer ali

Exprerimenter nudi automatizaciju celog procesa, testove valjanosti i performansi sistema.

Naravno, iza svih ovih grafičkih okruženja nalazi se osnovna funkcionalnost WEKA-e, kojima može da se pristupi iz komandne linije.

Prednosti WEKA-e su svakako širok spektar metoda za pripremu podataka, izbor atributa i algoritama integrisanih u jednom alatu. Zatim činjenica da bilo koja od metoda koja je implementirana u Weki može biti pozivana iz korisničkog koda. Posledica je olakšan razvoj novih aplikacija za istaživanje podataka uz minimum dodatnog kodiranja. Zatim, alat je potpuno besplatan, jednostavno i lako se instalira na svakoj platformi. GUI ga čini jednostavnim za korišćenje.

S druge strane veliki nedostatak WEKA-e je dokumentacija. WEKA konstantno raste i dokumentacija daje samo listu raspoloživih algoritama. Naročito je dokumentacija za grafičko korisničko okruženje ograničena. Drugi mogući problem pri radu sa WEKA-om je skalabilnost. Pri radu sa velikim količinama podataka vreme za obradu se drastično povećava. Treći nedostatak je činjenica da u GUI okruženju nisu implementirane sve funkcionalnosti WEKA-e pa je u radu neke opcije potrebno pozivati iz komandne linije.

4.7 Primenjene metode klasifikacije

Za formiranje testnog modela korišćena su dva algoritma klasifikacije - drvo odlučivanja i neuronske mreže. Za oba klasifikaciona algoritma razmotrili smo odgovarajuće implementacije u *WEKA*-i.

U praksi se pokazalo da se kombinovanjem više različitih metoda postižu bolji rezultati nego primenom samo jedne metode. Neke od metoda koje su se razvile kao varijacije na klasično drvo odlučivanja koriste *bagging*, *boosting* i *stacking*[11]. Algoritam *ADTree* je jedna od metoda razvijena tokom poslednje decenije koja je

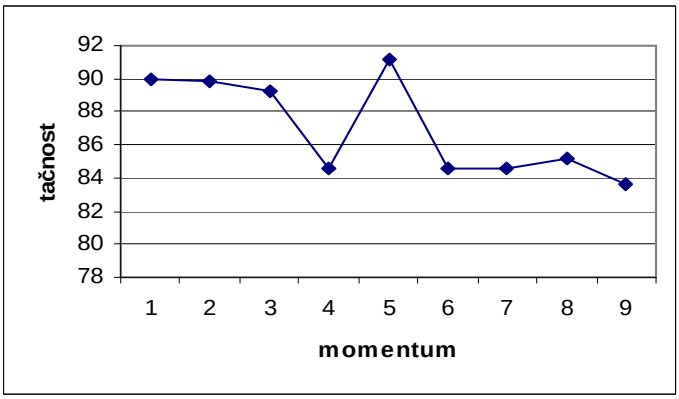
pokazala jako dobre rezultate i kod predviđanja i kod klasifikacije. *ADTree* se zasniva na isticanju (eng. *boosting*), možda najmoćnijoj od pomenute tri tehnike. Klasifikator *ADTree* se zasniva na konstrukciji alternirajućeg drveta odlučivanja isticanjem [11] i prilagođen je problemima u kojima se pojavljuju dve klase. Broj iteracija isticanja (eng. *NumOfBoostingIterations*) je parametar koji treba prilagoditi podacima i željenom odnosu tačnosti i vremena utrošenog na učenje. Više iteracija isticanja rezultovaće većim i potencijalno preciznijim drvetom odlučivanja ali će vreme učenja biti duže. Svaka iteracija dodaje tri čvora drvetu odlučivanja – jedan čvor podele i dva čvora predviđanja. Podrazumevani metod pretrage je detaljna pretraga (eng. *Expand all paths*). Ostale metode za pretragu su heurističke, ne garantuju optimalno rešenje ali su brže. Ovaj klasifikator nudi još opciju čuvanja podataka za grafički prikaz.

Klasifikator *MultilayerPerception* (*MLP*) je implementacija algoritma neuronske mreže sa propagacijom unazad. Za razliku od drugih funkcija implementiranih u *WEKA*-i *MLP* poseduje i grafičko korisničko okruženje. Kroz ovo okruženje je moguće interaktivno menjati topologiju mreže dodavanjem čvorova i veza. Takođe je moguće podešavanje stepena učenja, momentuma kao i broja prolazaka kroz podatke (eng. *epoch*). Stepen učenja je parametar koji određuje stepen korigovanja težinskih koeficijenata. Veće vrednosti dovode do većeg oscilovanja težinskih koeficijenata i proces treniranja je kraći. Momentum sprečava preveliku oscilaciju težinskih koeficijenata. Učenje se prekida nakon zahtevanog broja prolazaka kroz podatke, nakon čega je moguće prihvatiti rezultate ili povećati broj prolazaka. Svi parametri *MLP* mogu da se podešavaju i u editoru. Parametar *hiddenlayers* definiše unutrašnje slojeve mreže tj. broj čvorova u svakom od slojeva. Parametar *decay* uzrokuje smanjenje stepena učenja i određuje tekuću vrednost stepena učenja tako što početnu vrednost stepena učenja deli sa brojem prolaska. Na ovaj način se sprečava da mreža divergira. Broj prolazaka kroz podatke se zadaje parametorm *trainingTime* a početne vrednosti težinskih koeficijenata se podešavaju na male slučajne vrednosti parametrom *seed*. Klasifkator *MLP* izabran je zbog ranijih rezultata neuronskih mreža u klasifikovanju podataka iz oblasti finansija i bankarstva.

Experimentisali smo sa raspoloživim parametrima za oba klasifikatora. Pokazalo se da klasifkator *ADTree* postiže najveću tačnost kada je paramatar *searchPath* podešen na *Expand the heaviest path.* Ostali parametri su takođe ostali na podrazumevanim vrednostima jer nisu imali uticaja na tačnost klasifikacije.

Kao što je prikazano na slici 8, *MLP* postiže dobru tačnost za vrednost momentuma 0.1. Slična analiza je primenjena za procenu optimalnih vrednosti drugih parametara. Mreža ima jedan unutrašnji sloj a stepen učenja je 0.2. Ostali parametri su zadržali podrazumevane vrednosti.

Algoritme smo ocenili tako što smo ceo skup podataka od 4478 instanci podelili u 6 disjunktnih trening-test skupova. Za svaki trening-test skup dve trećine podataka su korišćeni za treniranje, a jedna trećina za testiranje dobijenog modela. U tabeli 1. su prikazani dobijeni rezultati.



Slika 8. Zavisnost tačnosti klasifikacije od vrednosti momentuma

Najveća tačnost postignuta upotrebom *ADTree* algoritma je 95,52%. Sve informacije o tačnosti modela mogu se dobiti iz matrice konfuzije. Matrica konfuzije sadrži informacije o tačnim i prognoziranim klasifikacijama. Kolone matrice su

	Prosečna tačnost	Standardna devijacija
MLP	90,05	3,01
ADTree	92,08	2,83

Tabela 1. Prosečna tačnost i standardna devijacija za *MLP* i *ADTree*

prognozirane klasifikacije, a vrste tačne klasifiacije. Tačnost modela je ukupan broj tačnih klasifiacija kroz ukupan broj instanci. Prosečna tačnost ovog algoritma je 92,08%

a standardna devijacija 2,83 (videti sliku 8). Primenom *MLP* u proseku je postignuta za 2% manja tačnost. Prosečna tačnost ovog algoritma iznosi 90,05% a standardna devijacija 3,01.

4.7.1 Povećanje tačnosti odabirom atributa

Odabir atributa je jedan od najvažnijih koraka u klasifikaciji i istraživanju podataka. Takođe, to je i vrlo efikasna tehnika za redukciju dimenzije podataka. Početni skup atributa najčešće nije optimalan i sadrži nepovezane i suvišne atribute.

Nepovezanim nazivamo atribute kod kojih se ne može uočiti veza sa atributom klase, dok su suvišni oni atributi koji nemaju pozitivan uticaj na tačnost klasifikacije. Obično ovakvi atributi čak imaju negativan uticaj na tačnost i/ili vreme potrebno za klasifikaciju. Posle inicijalne primene metoda klasifikacije i dobijanja radne verzije rezultata, primenom metoda odabira atributa iz početnog skupa atributa izdvojen je podskup koji daje veću tačnost klasifikacije.

Na osnovu [8], metod za odabir atributa se sastoji od procedure generisanja (za pretragu kroz prostor svih mogućih podskupova početnog skupa atributa), funkcije evaulacije (meri vrednosti tekućeg podskupa atributa) i procedure provere ispravnosti (potvrđuje tačnost rezultujućeg podskupa atributa). Osnovna ideja metoda za odabir atributa je analiza svih mogućih podskupova početnog skupa atributa u cilju nalaženja optimalnog rezultujućeg podskupa. Rezultujući podskup treba da da najveću tačnost i/ili brzinu klasifikacije. Teorijski, svi atributi koji nemaju uticaj na atribut klase treba da budu eliminisani iz rezultujućeg podskupa. U praksi, rezultujući podskup zavisi od primenjene funkcije evaulacije.

Metode za odabir atributa su podeljene u dve grupe: *filter* i *wrapper*. *Filter* metod je nezavisan od primenjenog algoritma. Primenom ovog metoda funkcija evaulacije ocenjuje vrednost tekućeg podskupa atributa na osnovu prediktivne sposobnosti svakog od atributa. Sa druge strane, *wrapper* metod kao funkciju evaulacije koristi neki od algoritama i uobičajeno postiže bolje rezultate jer uzima u obzir interakciju algoritma i podataka. Nedostatak *wrapper* metoda je što zahteva više izračunavanja u odnosu na *filter* metodu, zbog čega se ređe primenju na velikim skupovima atributa.

Optimizacija inicijalno dobijenog modela je pokušana odabirom atributa pomoću *filter* i *wrapper* metoda impemantiranih u *WEKA-i*. Od implementiranih *filter* metoda korišćena je metoda zasnovana na korelaciji (*CFS*, eng. *Corelation Based Filter*). *CFS* ocenjuje vrednost podskupa atributa na osnovu pojedinačene prediktivne sposobnosti svakog atributa i njihove međusobne zavisnosti. Poželjni su atributi koji imaju veliku zavisnost sa atributom klase a malu međusobnu zavisnost. Iako je *CFS* u osnovi *filter* metod funkcija evaulacije je drvo odlučivanja. Uporedna analiza prikazane u [8] pokazuje da *CFS* postiže slične rezultate kao i *wrapper* metode u slučaju klasifikacije. Za generisanje podskupova kod primene obe metode je korišćena *best-first* funkcija.

Algoritam *MLP* smo eliminisali iz daljeg razmatranja jer su rezultati *ADTree* algoritma dali za 2% veću tačnost. Na osnovu [9], izbor atributa ne utiče na početni rang klasifikatora. Odnosno, klasifikator koji daje najbolje razultate na početnom skupu atributa daće najbolje rezultate i na svakom podskupu.

Dobijeni rezultati nakon klasifikacije *ADTree* na redukovanom skupu atributa pokazali su *wrapper* metod daje bolje rezulate od *filter* metoda. Tačnost klafikacije upotrebom *wrapper* metode se povećala posečno 1%. Dok je filter metoda izdvojila veći broj atributa.

Atributi koji odbačeni su nepovezani ili suvišni. Analiziranjem nekih od atributa koji su odbačeni može se doći do ovog zaključka (videti tabelu 2.). Atributi koje su obe metode odbacile kao nebitne odnose se na indikatore posedovanja kredita i platnih kartica, upotrebu bankomata, internet plaćanja i šaltera i bankomata drugih banaka. Analizom ovih atributa dolazimo do zaključka da više od 90% klijenta nije koristilo ni jednu od ovih usluga.

Iznenađujuće je da su odbačeni atributi DEBOLOKADEUK i DEBOLOKADE08 koji su indikatori da je na klijentovom računu već naplaćena kazna za prekoračenje po dinarskom tekućem računu.

U poslednjoj fazi, sa ciljem dobijanja što boljih rezultata klasifikacije kombinovali smo rezultate oba metoda. Primetili smo da je *wrapper* metod izabrao demografske i transakcione atribute. S obzirom da je *wrapper* metod dao bolje rezultate zaključak je da demografski i transacioni podaci nose više informacija o klijentu. Sa druge strane, *filter* metod je postigao malo slabije rezultate i , takođe, je odabrao najveći

broj demografski i transakcionih atributa ali je izdvojeno i nekoliko dodatnih i relacijskih atributa.

	Wrapper metod	*Filter* metod
Zajednički Odabrani Atributi	*SIFRADELAT, RR, UPLUK08, AVGUPL08, AVGISPUK, AVGISP08, BRTRUPLUK, BRTRUPL08*	
Nezavisno Odabrani Atributi	*REGION, GGG , MAXUPL08, MAXISPUK, MAXISPL08, BRTRISPUK*	*RADNISTATUS, AVG08, MINUPLUK, MINISPUK,KREDIT08, OVLLICE08, MINISP08, BRDANAODOTV, AVGUPLUK, MINUPL08, POL, BRTR08, DEBLOKADEUK, BRTRDRBANKUK*
Zajednički Odbačeni Atributi	*NEDOZKAMUK, NEDOZKAM08, DEBLOKADE08, IMADINUUK, IMADINU08 , IMAVISUUK, IMAVISU08, IMAMASTERUK , IMAMASTER08, IMAMAESTROUK, IMAMAESTRO08, KREDITUK, UPUK , ISUK, ISUK08, AVGUK, MAXUPLUK, BRTRUK, BRTRISP08, PROFITUK , PROFIT08, OVLLICEUK, BRTRBANUK, BRTRBAN08, BRTRINTUK , BRTRINT08, BRTRDRBANK08*	
Nezavisno Odbačeni Atributi	*POL, RADNISTATUS, DEBLOKADUK , KREDIT08, AVG08, AVGUPLUK, BRTR08, OVLLICE08, BRTRDRBANKUK, BRDANAODOTV, KREDIT08, MINUPLUK, MINUPL08*	*GGG, REGION, MAXUPL08, BRTRISPUK, BRTRBANUK*

Tabela 2. Odabrani skupovi atributa *wrapper* i *filter* metodom

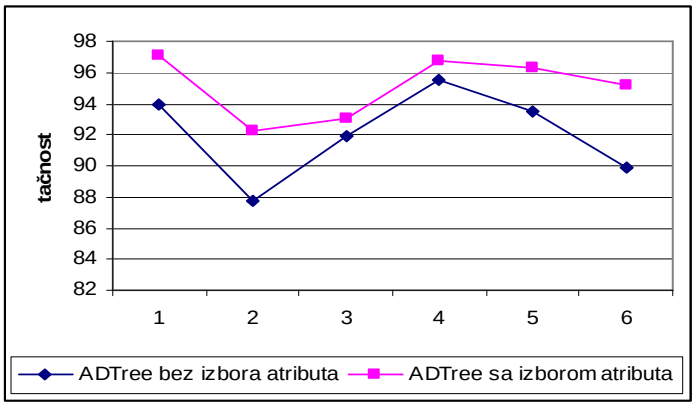
Počevši od unije atributa odabranih od strane oba metoda odbacivali smo po jedan atribut odabran od strane *filter* metoda a iz skupa relacijskih i dodatnih. Odbacivani su atributi iz skupa relacijskih i dodatnih jer *wrapper* metodom koja je postigla bolje

rezultate nije odabran ni jedan atribut iz ovih skupova. Odbacivanjem atributa KREDIT08, OVLLICE08, BRTRDRBANUK i DEBLOKADEUK prosečna tačnost je uvećana na 94,85%. Atribut BRDANAODOTV je zadržan.

4.8 Analiza krive učenja

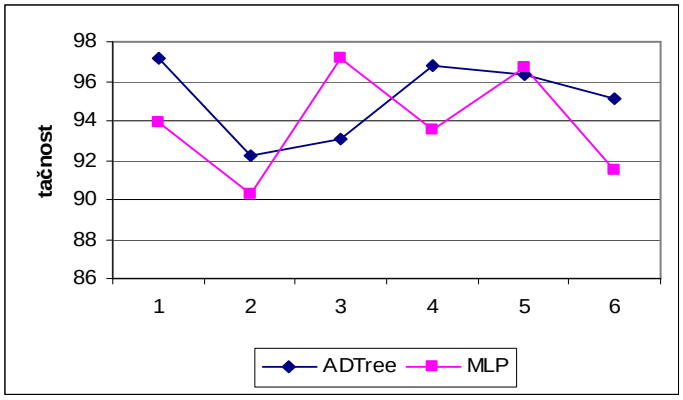
Ovaj odeljak sadrži prikaz analize krive učenja oba klasifikaciona algoritma. Ispitivali smo kakav uticaj ima izbor atributa na proces učenja odnosno, kakav uticaj ima izbor atributa na odnos klasifiktora . Takođe, ispitivali smo i kakav uticaj izbor atributa ima na bolji klasifikator *ADTree*.

Na slici 8 prikazana je kriva učenja algoritma *ADTree* sa i bez izbora atributa. Vidimo da ne samo da su postignuti bolji rezultati nakon izbora atributa nego je i kriva učenja stabilnija. Oscilacije u tačnosti između različitih trening-test skupova su ublažene. Tako da možemo očekivati i da rezultati na novim objektima (nepoznate klasifikacije) budu precizniji i stabilniji.



Slika 9. Kriva učenja *ADTree* sa i bez izbora atributa

Zatim, analizirali smo odnos tačnosti oba algoritma nakon izbora atributa. Rezultati su prikazani na slici 9. Kao što smo i očekivali *ADTree* je zadržao poziciju boljeg klasifkatora dok je *MLP* postigao prosečnu tačnost od 93,84% što je za procenat slabije od *ADTree*.



Slika 10. Krive učenja za *ADTree* i *MLP* nakon izbora atributa

5 Zaključak

U prvom delu rada prikazana je jedna od metoda za istraživanje podataka – klasifikacija. Klasifikacija je jedna od najčešće korišćenih metoda u ovoj oblasti. Prikazani su teorijske osnove i principi za dva algoritma klasifikacije: drvo odlučivanja i neurnoske mreže. Takođe su date teorijske osnove za pripremu podataka za proces istraživanja.

U drugom delu rada dat je opis modela konstruisanog za identifikovanje visokorizičnih klijenata u poslovanju sa dinarskim tekućim računima. Za ovu primenu, korišćeni su podaci Banke Poštanske Štedionice a.d. U fazi prikupljanja podataka razmotrili smo veliku količinu istorijskih podataka. Odabir atributa za konstrukciju modela saglasan je radovima iz sličnih oblasti. Takođe, određeni atributi dodati su nezavisno. Neki od ovako odabranih atributa su reprezentativni s obzirom da su odabrani od strane dva različita metoda za odbir atributa.

Koristili smo klasifikacione algoritme obrađene u prvom delu rada: drvo odlučivanja i neurnoske mreže tj. njihove implementacija u WEKA-i. Ovi algoritmi su korišćeni ranije u sličnim sistemima sa dobrim rezultatima. Iako su neuronske mreže pokazale dobre rezultate u rešavanju različitih problema , za problem klasifikacije klijenata bolje rezultate je dalo drvo odlučivanja. S obzirom, da nismo teorijski razmatrali nezavisnost atributa pri odabiru korisitili smo dva metoda za izbor atributa *filter* i *wrapper*. Pokazalo se da se najbolji razultati postižu kombinovanjem oba metoda. Na osnovu tekućih podataka algoritam drvo odlučivanja pokazao je bolje rezultate u identifikovanju visokorizičnih klijenata i nakon izbora atributa.

Ovo istraživanje je pokazalo da se istraživanje podataka može sa uspehom da se primeni na određivanje profila potencijalno rizičnog klijenta i na taj način olakša odlučivanje u slučaju odobravanja kredita ili drugih sredstava iz budzeta banke. S

obzirom, da u ovakvoj vrsti istraživanja veliku ulogu imaju raspoloživi podaci, još jedna od posledica sprovedenog istraživanja je potreba da se donese odluka o minimalnoj količina podataka koja mora da se vodi o klijentu, kao i o procedurama da se ti podaci održavaju ažurnim i proširuju u budućnosti.

Literatura

[1] M.Berry, G.S.Linoff, Data mining techniques for marketing sales and customer support, 2nd ed. Willey, 2004, ISBN: 978-0-471-47064-9

[2] C.Baragoin, C.M. Anderson, S.Bayerl, G.Bent, J.Lee, C. Schammer , Mining your own business in Banking, IBM RedBooks , 2001, SG24-6272-00

[3] S.H. Javaheri, Response Modeling in direct marketing: a data mining based approach for target selection, Lulea University of Technology, 2008:014 ISSN 1653-0187

[4] J. Han, M. Kamber, Data mining,concepts and techniques, Morgan Kaufmann, 2nd edition, 2006, ISBN: 1-55860-901-6

[5] R.Dass, Data mining in banking and finance: a note for bankers, Indian Institute of Management Abmedabad

[6] M. Bramer, Principles od Data Mining, Springer, 2007, ISBN 978-1-84628-765-7

[7] Web strana http://www.cs.waikato.ac.nz/ml/weka/

[8] M.A.Hall, Correlation-based feature selection for machine learning, April 1999

[9] C.V.Bratu, T. Muresan, R.Potolea, Improving Classification Accuracy trough Geature Selection, Proceedings of IEEE ICCP 2008, pp 25-32, **ISBN:** 978-1-4244-2673-7

[10] Portia A. Cerny, Aim Proximity, Data mining and Neural networks from a Commercial perspective, University of Technology Sydney, Australia

[11] I.W.Witten, E. Frank, Practical machine learning tools and techniques, 2005, ISBN 0-12-088407-0

Sadržaj

1 Uvod 1
2 Priprema podataka 4
2.1 Čišćenje podataka 6
2.2 Intergracija podataka 8
2.3 Transformacija podataka 9
2.4 Redukcija podataka 11
2.4.1 Izbor atributa 11
2.4.2 Diskretizacija i hijerarhija koncepata 12
3 Klasifikacija 13
3.1 Klasifikacija pomoću drveta odlučivanja 15
3.2 Neuronske mreže 19
4 Procena rizika u bankarstvu 23
4.1 Primena klasifikacije u proceni klijenata 23
4.2 Proces konstrukcije modela 24
4.3 Prikupljanje podataka 25
4.3.1 Izbor atributa 28
4.3.2 Konstrukcija atributa 31
4.4 Priprema podataka 32
4.5 Normalizacija 33
4.6 Alat korišćen za klasifikaciju 34
4.7 Primenjene metode klasifikacije 37
4.7.1 Povećanje tačnosti odabirom atributa 40
4.8 Analiza krive učenja 43
5 Zaključak 44
Literatura 47