Ispitivanje performansi genetskog algoritma za klasifikaciju podataka u poslovnim primjenama

Korman, Mateo

Sveučilište u Zagrebu

Ekonomski fakultet

Specijalistički diplomski stručni studij Elektroničko poslovanje u privatnom i javnom sektoru

Ispitivanje performansi genetskog algoritma za klasifikaciju podataka u poslovnim primjenama

Diplomski rad

Mateo Korman

Zagreb, rujan, 2023.

Sveučilište u Zagrebu

Ekonomski fakultet

Specijalistički diplomski stručni studij Elektroničko poslovanje u privatnom i javnom sektoru

Ispitivanje performansi genetskog algoritma za klasifikaciju podataka u poslovnim primjenama

Testing the performance of a genetic algorithm for data classification in business applications

Diplomski rad

Mateo Korman, JMBAG: 0067559131

Mentor: Prof. dr. sc. Mirjana Pejić Bach

Zagreb, rujan, 2023.

Sažetak i ključne riječi

U svijetu u kojem danas živimo, kada se svaki dan generiraju ogromne količine podataka, poduzeća su primorana naučiti kako iz njih iskoristiti maksimum. Jedan od načina su korištenje raznih metoda otkrivanja znanja iz podataka. Pristupom internetu, svima su dostupni razni alati i već napisani dijelovi programskog koda koji se koriste u algoritmima te je tako pristup otkrivanju znanja iz podataka uvelike olakšan. Jedan od takvih alata s knjižnicom mnoštva algoritama je Weka. Glavni fokus u ovom radu je klasifikacija podataka pomoću algoritama šuma, odnosno usporediti algoritme šuma s algoritmom temeljenim na genetskom algoritmu. Uz pomoć Weka-e prezentirani su rezultati četiri klasifikacijskih algoritama nad pet setova podataka iz različitih poslovnih područja te donesen zaključak o performansama genetskog algoritma.

Ključne riječi: otkrivanje znanja iz podataka, klasifikacija, algoritmi šuma, genetski algoritam, Weka, OptimizedForest, RandomForest, RandomTree, REPTree

Summary and keywords

In the world and time we live in today, when huge amounts of data are generated every day, companies are forced to learn how to make the most of it. One of the ways is the use of various methods of discovering knowledge from data. With access to the Internet, various tools and already written parts of program code that are used in algorithms are available to everyone, and thus access to discovering knowledge from data is greatly facilitated. One such tool with a library of many algorithms is Weka. The main focus of this paper is data classification using forest algorithms, respectively to compare forest algorithms with an algorithm based on a genetic algorithm. With the help of Weka, the results of four classification algorithms were presented over five sets of data from different business areas, and in the and conclusion was delivered in order to evaluate performance of genetic algorithm.

Keywords: data mining, classification, forest algorithms, genetic algorithm, Weka, OptimizedForest, RandomForest, RandomTree, REPTree

Sadržaj

1. Uvod

U današnjem digitalnom dobu, poslovne organizacije se suočavaju s ogromnim količinama podataka koje je potrebno analizirati kako bi se donosile kvalitetne i kompetentne odluke. Klasifikacija podataka je jedan od ključnih procesa u analizi podataka i rudarenju znanja. U tu svrhu, različite metode strojnog učenja i dubokog učenja su postale neprocjenjive, među kojima se ističe genetski algoritam kao moćan optimizacijski alat. Genetski algoritmi su algoritmi inspirirani procesima prirodne selekcije i evolucije, a koriste se za pronalaženje optimalnih rješenja u raznim problemima. U poslovnim primjenama, klasifikacija podataka igra ključnu ulogu u segmentaciji tržišta, analizi korisničkih preferencija, identifikaciji anomalija i mnogim drugim aspektima poslovanja. U ovom diplomskom radu, istražujemo performanse genetskog algoritma u kontekstu klasifikacije podataka u poslovnim primjenama.

1.1. Cilj rada

Cilj ovog istraživanja je procijeniti kako genetski algoritam može doprinijeti kvaliteti i točnosti klasifikacije podataka u poslovnom okruženju. Analizirat ćemo različite aspekte primjene genetskog algoritma, uključujući prilagodbu parametara i rezultate modela kreiranog algoritmima. Ispitivanje će se izvršiti na uzorku od pet setova podataka kroz četiri različita algoritma. Također, usporedit ćemo rezultate genetskog algoritma s drugim popularnim metodama strojnog učenja kako bismo utvrdili njegovu konkurentnost i prednosti ili nedostatke.

1.2. Metodologija rada i izvori podataka

Teorijski dio rada obrađen je istraživanjem sekundarnih izvora podataka, a to su razni znanstveni članci, stručni radovi, stručna literatura u obliku knjiga te ostali internetski izvori. Istraživački dio rada u kojem se želi prikazati rezultat genetskog algoritma u usporedbi s ostalim algoritmima je proveden s javno dostupnim setovima podataka preuzetih sa stranice www.kaggle.com. Za obradu podataka je korišten software Weka verzija 3.8.6. Korištene su deskriptivne metode prilikom analize rezultata.

1.3. Struktura diplomskog rada

Rad je podijeljen na sedam poglavlja u kojem se predstavljaju ključna područja za izradu rada. Predstavljeni su pojmovi otkrivanja znanja i podataka, ciljevi i proces otkrivanja

znanja, metode klasifikacije i algoritmi šuma, objašnjen je pojam genetskog algoritma te što on je, a na posljetku analizirani su rezultati dobiveni provedbom algoritama kroz setove podataka.

U prvom djelu rada nalazi se kratki uvod u temu, cilj rada, izvori podataka koji su korišteni za pisanje rada te strukturu samog diplomskog rada.

U drugom poglavlju rada predstavljen je pojam otkrivanja znanja iz podataka. Opisani su osnovni pojmovi otkrivanja znanja iz podataka te su predstavljeni ciljevi, procesi otkrivanja znanja te glavne metode koje se koriste u ovom području.

U trećem poglavlju rada je objašnjen pojam klasifikacija podataka koja je ključna za izradu rada s obzirom ta upravo svim algoritmi korišteni u radu pripadaju ovoj metodi otkrivanja znanja.

U četvrtom djelu rada približe je objašnjena tehnologija genetskog algoritma te kako on zapravo djeluje i na temelju kojih zakona.

U petom djelu rada su prikazani setovi podataka nad kojima su algoritmi djelovali. Setovi su objašnjeni svaki individualno, a u konačnici je obrađena kratka usporedba svih setova.

U šestom djelu rada analizirani su i predstavljeni rezultati koji su dobiveni analizom setova uz pomoć softwarea Weka. Prikazani su i algoritmi korišteni u radu te su pobliže objašnjeni.

2. Otkrivanje znanja iz baza podataka

Načinom na koji informatičke tehnologije napreduju, a i samom činjenicom da su te tehnologije dostupne širokoj populaciji, stvaraju se i generiraju iznimno velike količine podataka koje se spremaju na računalima, internim i eksternim mrežama, bazama podataka raznih trgovačkih društava (Kantardžić, 2003.). Pitanje koje se postavlja je što učiniti s tom količinom podataka i na koji način ih adekvatno i optimalno analizirati. Poduzeća, institucije i slični više fokusa usmjeravaju na načine skladištenja i prikupljanja što nužno neće rezultirati najkvalitetnijim podacima za analizu, a manifestira se potrošnjom velike količine resursa potrebnih za njihovo upravljanje. Iz ovih se problema, a i prilika razvila potreba za razumijevanjem i upravljanjem podataka u svim praktički svim poslovnim područjima. Dakle, načini i metode za rješavanje problema velike količine podataka, postale su od dragocjene važnosti kako bi se izvukle informacije i znanja korisna za donošenje kvalitetnih poslovnih odluka do razina strateškog odlučivanja unutar poslovnih organizacija. Jedan od načina upravljanja podacima je rudarenje podataka koje će biti detaljnije obrađene u ovom poglavlju rada (Žapčević i Butala, 2015.).

2.1. Osnovni pojmovi otkrivanja znanja iz baza podataka

Pod pojmom otkrivanje znanja iz baza podataka odnosno rudarenje podataka (engl. data mining) podrazumijevamo korištenje računalnih postupaka i alata uz pomoć kojih lakše analiziramo i obrađujemo podatke. Ovo područje smatra se relativno novim područjem znanosti koje se razvija enormnom brzinom te sadrži veliki broj tehnika i metoda za kvalitetnu obradu podataka. Razvija se brzinom da praktički sadašnja tehnologija i tehnika već sutra može postati starina te pristup obradama može biti unaprijeđen novim vještinama i znanjima. Zbog ovakvih činjenica ne postoji standardiziran pristup rješavanju problema analize podataka već svaki problem može biti promatran iz raznih perspektiva i rješavan mnoštvo tehnika sukladno tome. Svaki pristup se može smatrati kvalitetnim pristupom konkretnom problemu. Sve što može pomoći shvaćanju i razumijevanju podataka koji se prikupljaju može se smatrati dobro došlim te se jedinim ograničenjem može smatrati uključenost računala i nedovoljna znanja korištenja istoga (Žapčević i Butala, 2015.).

Termin rudarenje podataka najčešće koriste statističari, analitičari i stručnjaci zaduženi za upravljanje sustavima informacijskih tehnologija unutar organizacija. Definicija otkrivanje znanja u bazama podataka (engl. Knowledge Discovery in Databases – KDD)

predložena je 1989. godine. Otkrivanje znanja iz podataka termin je za sveukupni proces otkrivanja korisnog znanja iz podataka, a rudarenje podataka je jedan korak u tom procesu.

Rudarenje podataka je primjena specifičnih algoritama za izvlačenje uzoraka iz podataka. Dodatni koraci u procesu otkrivanja znanja su priprema, odabir i čišćenje podataka, korištenje odgovarajućeg znanja te pravilno i jasno tumačenje rezultata rudarenja kako bi se osigurala maksimalna iskoristivost analiziranih skupova podataka i kako bi pomogla organizacijama u odlučivanjima.

Rudarenje podataka se također može definirati kao znanost izdvajanja korisnih informacija iz velikih skupova podataka ili baza podataka. Disciplina je koja koristi statistiku, strojno učenje, upravljanje podacima i bazama podataka, prepoznavanje uzoraka, umjetnu inteligenciju i druga područja analize podataka.

Kako je iznad u radu navedeno, s obzirom na to da je svaki pristup obradi podataka jedinstven te praktički krivi pristup ne postoji, rudarenje podataka možemo smatrati i umjetnošću, a ne samo kao znanost (Pejić Bach, 2005.).

2.1.1. Baze podataka

Baze podataka možemo definirati kao središte informacijskog sustava, a one služe za pohranjivanje podataka poslovanja neke organizacije te rezultiraju velikom količinom podataka koje je potrebno dugoročno čuvati i skladištiti. Bazu podataka također možemo promatrati kao bazu koja za potrebe dobivanja poslovnog znanja unutar sebe sadrži posebno pripremljene podatke za pokretanje zahtjevnih analiza (Kramberger i suradnici, 2018.).

Baze podataka sadrži podatke o entitetima. Entiteti su bilo kakav stvarni ili apstraktni objekti koji imaju svrhu, a po svojim se značajka mogu razlikovati od ostalih objekata. Entiteti su bilo što o čemu mogu biti prikupljeni podatci.

Značajke entiteta koje ga razlikuju od ostalih entiteta nazivamo atributima. Odabir atributa za entitet ovisi o svrsi informacijskog sustava. Za entitet osobe koja je radnik u određenoj tvrtki bit će potrebni atributi kao što su adresa, iznos plaće, podatak o odjelu u kojem radi, dok će se za osobu koja je student na nekom fakultetu izraditi atributi kao što su JMBAG studenta, naziv studija na kojem studira, prethodno završena škola i slično (Kramberger i suradnici, 2018.).

2.2. Ciljevi otkrivanja znanja iz podataka

Primarni ciljevi rudarenja podataka obično su predviđanje i deskripcija. Predviđanje uključuje korištenje varijabli ili polja u skupu podataka za predviđanje nepoznatih ili budućih vrijednosti drugih varijabli od koje su korisniku podataka zanimljivi. Deskripcija, s druge, fokusira se na pronalaženje obrazaca koji opisuje podatke tako da bih ih korisnici mogli interpretirati. Stoga, rudarenja podataka se može svrstati u sljedeće kategorije

1. Prediktivno rudarenje podataka – predlaže budući model temeljem danog skupa podataka

2. Deskriptivno rudarenje podataka – kreirati novi skup informacija na temelju danog skupa podataka

Prediktivnom rudarenju je proizvesti model, izražen kao izvršni kod, koji se može koristiti za obavljanje klasifikacije, predviđanje, procjena i sličnih tehnika. Deskriptivnom rudarenju cilj je steći razumijevanje analiziranog sustava otkrivanjem obrazaca i odnosa u velikim skupovima podataka (Kantardžić, 2003.).

Ciljevi otkrivanja znanja iz podataka se mogu pojednostavniti i navesti kao sljedeći:

1. Provođenje analize kao pomoć pri donošenju ispravnih poslovnih odluka

2. Razvoj alata za pomoć kod upravljanja poslovnim pravilima

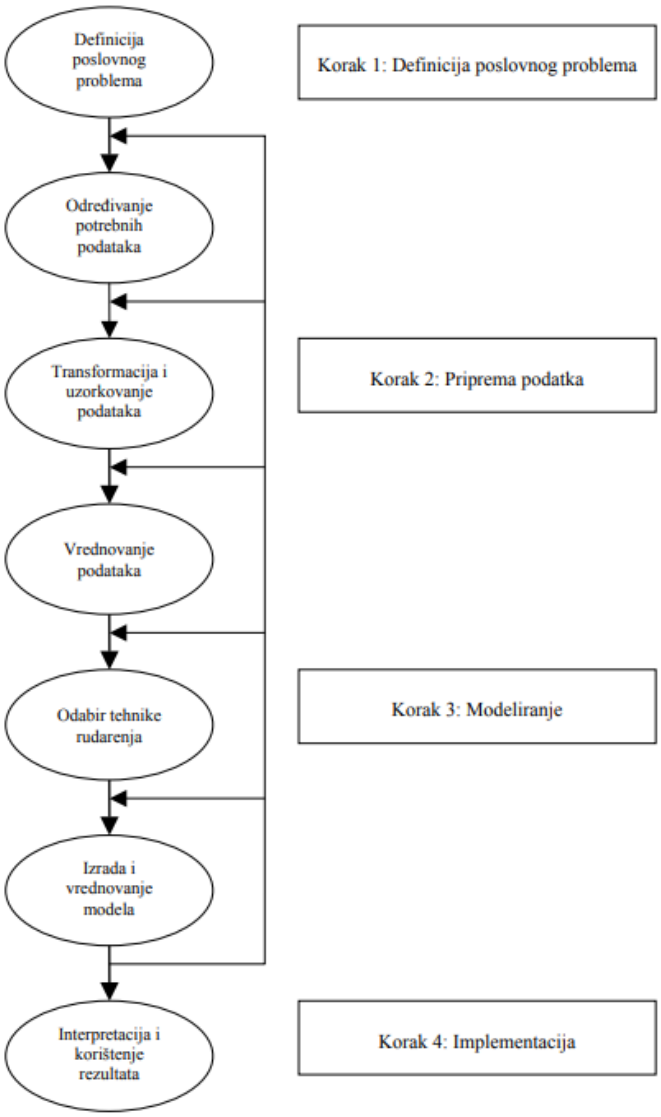
3. Razvoj aplikacija koje će se moći implementirati u informacijske sustave (Pejić Bach i Kerep, 2011).

2.3. Proces otkrivanja znanja iz podataka

Kako ne postoji standardizirani pristup analizi podataka, tako ni ne postoji određen način kojim će se garantirati da će rudarenje podataka biti uspješno te da će informacije dobivene rudarenjem biti vrijedne i korisne. Šanse da se takve informacije osiguraju, mogu se povećati sljedeći korake procesa rudarenja podataka. Prvi korak je definiranje poslovnog problema za čije se rješavanje mora odraditi rudarenje podataka. Drugi korak je vezan za pripremanje podataka. Uključuje određivanje podataka, transformaciju i uzrokovanje, te vrednovanje podataka. Treći korak u procesu je modeliranje podataka te odabir metode rudarenja i izradu modela. Četvrti korak je implementacija koja podrazumijeva interpretaciju i korištenje rezultata. Proces rudarenja se smatra iterativnim procesom što bi značilo da se u svakom od

koraka može vratiti na neki od prethodnih. Kao primjer se navodi vračanje na korak dva, što je i najčešća situacija uzeći u obzir da je podatke teško obraditi i pripremiti inicijalno. Priprema podataka često iziskuje dodatne dorade zbog određenih problema na koje će se sigurno naići, a oni nisu prepoznatljivi dok se podaci ne krenu modelirati. Također, osoba ili osobe koje rudare podatke upoznaju i poslovni problem sa samim procesom te se on ujedno i kroz proces revidira konstantno. Slika broj 1 prikazuje proces otkrivanja znanja iz podataka.

Slika 1.: Prikaz procesa otkrivanja znanja iz podataka



Izvor: Radni materijali MZOP, EFZG

2.3.1. Definicija poslovnog problema

Prvi korak u procesu rudarenja podataka je definicija poslovnog problema. Poslovni problem je potrebno formulirati u obliku pitanja na koje odgovor dolazi pri završetku procesa. Najbolji pristup definiranju poslovnog problema je analiziranje područja gdje je rudarenje podataka već uspješno korišteno. Nakon što se dobro upoznamo s uspješnim primjenama rudarenja podataka, možemo odabrati područje koje je najkritičnije za naše poduzeće (Pejić Bach, 2005.).

2.3.2. Priprema podataka

Drugi korak u procesu je priprema podataka. Onda obuhvaća određivanje potrebnih podataka, transformaciju i uzrokovanje te vrednovanje podataka. Vremenski je najzahtjevnija i obuhvaća između 60 i 90 posto vremena koje se treba utrošiti na rudarenje. Iako se podaci mogu nalaziti na raznim mjestima, najčešće su to relacijske baze podataka i skladišta

2.3.2.1. Određivanje potrebnih podataka

Odluku o setu podataka koji će se koristiti za izradu modela najčešće donose analitičar, stručnjak iz poduzeća te informatičar. Popis podataka koji će se koristiti u izradi modela konačan su rezultat ovog koraka. U ovom pod koraku procesa, cilj je i odrediti varijable za analizu. Neke se odbacuju, a određuje se koje varijable će biti zavisne, a koja će biti ciljna. Konačan rezultat određivanja potrebnih podataka je popis varijabli koje će se koristiti u izradi modela (Pejić Bach, 2005.).

2.3.2.2. Transformacija podataka

U ovom pod koraku, cilj je podatke transformirati u oblik koji će odgovarati odabranoj metodi za rudarenje podataka. Podaci se transformiraju tako da formiraju tabelu u kojoj će stupci sadržavati atribute entiteta, a u recima će postojati sami entiteti. Svaki redak mora opisivati podatak značajan za poslovni problem koji se pokušava riješiti rudarenjem podataka (Pejić Bach, 2005.).

2.3.2.3. Uzorkovanje podataka

Za izradu modela rudarenjem podataka nije potrebno koristiti sve podatke koji su dostupni. Uzorkovanjem podataka se odabire dovoljna količina podataka koji su potrebni. Pitanje i nedoumica koja se često postavlja je: Koliko podataka je dovoljno kako bi model bio uspješan? Ovdje se često postavlja pitanje: Koliko je podataka dovoljno? Odgovor na ovo pitanje je jedinstven za svaki set iz razlog što količina potrebna količina podataka ovisi

o algoritmu koji će se koristiti za rudarenje podataka. Česta je situacija da je količina podataka manja od optimalne. Nakon što se izabere uzorak za izradu modela, dijeli ga se u dva dijela, dio za izradu modela i dio za testiranje modela. Takav pristup je nužan jer služi za provjeru efikasnosti modela na podacima koji se ne koriste kako bi model bio izgrađen (Pejić Bach, 2005.).

2.3.2.4. Vrednovanje podataka

Cilj vrednovanja podataka je utvrđivanje postojana vrijednosti netipične za skup koji se analizira. Netipične vrijednost postoje u svakom skupu podataka. Primjerice, u skupu podataka koji sadrži informacije o broju narudžbi kupaca, takve vrijednosti bi bili kupci s jako velikim i jako malim brojem narudžbi. Kako te vrijednosti ne bi utjecale značajno na krajnji rezultat, potrebno je odlučiti što s njima napraviti. One se mogu ukloniti iz skupa, može ih se svesti na primjerice aritmetičku sredinu i tako dalje. uz netipične podatke, setovi podataka skloni su sadržavati i prljave podatke. To su naprimjer, nepostojeće vrijednosti. Kako bi se takve vrijednosti popunile, potrebno je probati doći do njih nekim drugim putem. Primjerice, u setu podataka banke, imamo neispunjen podatak o zadnjoj transakciji. Do zadnje transakcije bi mogli doći tako da uz pomoć trenutnog stanja računa i prethodno stanja računa izračunamo promjenu i nju označimo kao zadnju transakciju. Prljavi podaci su najčešće produkt ljudske pogreške prilikom unošenja podataka u sustav (Pejić Bach, 2005.).

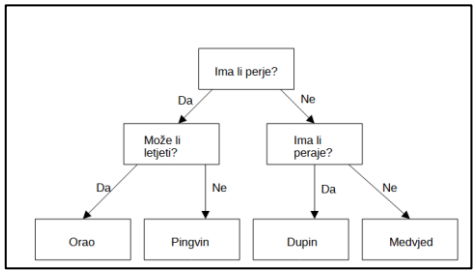
2.3.3. Modeliranje

U procesu se koriste razne metode: statistika, baze i skladišta podataka, umjetna inteligencija i vizualizacija. Tri su glavne metode koje se koriste za izradu modela: otkrivanje, klasifikacija i predviđanje.

Nakon što je u drugom koraku odabran set podataka, i definirano pitanje poslovnog problema, odabire se metoda kojom će model biti izgrađen.

Za predviđanje događaja koriste se metode za klasifikaciju – stabla odlučivanja, neuralne mreže i tako dalje. Na slici broj 2 možemo vidjeti jednostavni prikaz predviđanja događaja stablom odlučivanja.

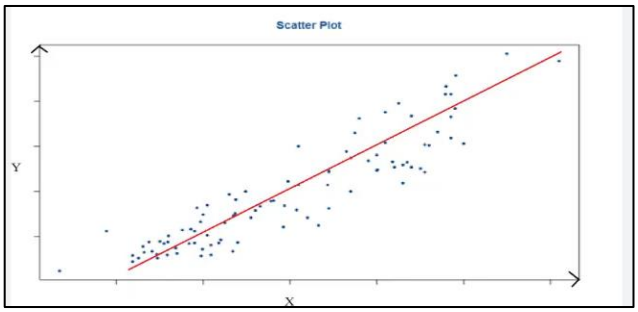
Slika 2.: Praktični primjer stabla odlučivanja



Izvor: Izrada autora

Za predviđanje numeričkih vrijednosti koriste se metode za predviđanje vrijednosti - linearna regresija, metode vremenskih serija te isto kao i kod metoda za klasifikaciju, neuronske mreže. Na slici broj 3 vidimo primjer analize linearnom regresijom.

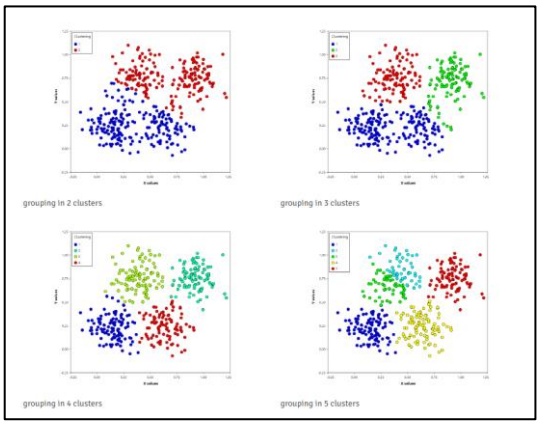
Slika 3.: Primjer linearne regresije



Izvor: Google Images

Za identifikaciju grupa unutar setova podataka koriste se metode grupiranja – klaster analiza. Slika broj 4 predstavlja prikaz analize podataka klaster analizom.

Slika 4.: Primjeri klaster analize



Izvor: Google Images

Metode koje se koriste kod modeliranja podataka će biti detaljnije objašnjenje u poglavlju 2.4. rada.

2.3.4. Implementacija

Jedna od glavnih uloga u procesu rudarenja podataka je uloga stručnjaka za poslovanje za čiji se problem rudarenje provodi. Stručnjaci su od značaja jer pomažu prilikom interpretacije rezultata te njihovoj primjeni u stvarnom poslovnom okruženju.

Nedostatak je što sami stručnjaci nemaju dovoljna znanja za rudarenje podataka stoga je krucijalno rezultate dostaviti u, čovjeku što razumljivijem formatu. Najčešće se u tu svrhu koriste grafikoni i vizualni prikazi. Što su rezultati bolje predstavljeni, to će se više koristiti i izglednije je da će model biti primijenjen u poslovanju.

Prednost bi bila i ako bi se modeli rudarenja implementirali u informacijske sustave poduzeća.

2.4. Metode otkrivanja znanja iz podataka

Metoda koja će se koristiti za otkrivanje znanja iz podataka ovisi o setu podataka koji će se koristiti. Također, bitno je i poslovno pitanje koje se postavlja prije nego što se krene u odlučivanje metode koja će se koristiti. Metode se koriste matematikom, statistikom, vizualizacijom podataka i tehnologijama za rudarenje podataka. Važno je napomenuti kako bez podrške softwarea nije moguće provoditi proces otkrivanja znanja. Tri su glavne metode: metoda otkrivanja grupa, metode za predviđanje događaja i metode za predviđanje vrijednosti te će one biti detaljnije objašnjene u ovom potpoglavlju rada (Pejić Bach, 2005.).

2.4.1. Metode otkrivanja grupa

Metoda otkrivanja grupa je metoda koja uključuje grupiranje podataka u grupe. Temeljni cilj je segmentirati promatrane instance u homogene skupine, tako da su objekti unutar jedne grupe što sličniji jedni drugima, a da su individualne grupe heterogene u skupu svih grupa, odnosno da su po svojim zajedničkim obilježjima što različitiji jedni od drugih. Najčešće korištena metoda je klaster analiza.

2.4.1.1. Klaster analiza

Klaster analiza je metoda koja se koristi za segmentiranje podataka, odnosno instanci seta koji se pokušava analizirati, s ciljem da se instance nađu u istom klasteru. Segmentacija se

vrši na osnovi rezultata koji se izračunava za svaki promatrani objekt skupa zasebno, ovisno o vrijednostima obilježja tog objekta po svim varijablama.

Na početku se broj klastera ne zna, a tako ni broj objekata koji pripada tom klasteru. Ne zna se ni koji će objekt pripasti kojem klasteru. Nekada se grupe mogu preklapati; primjer može istovremeno pripadati nekolicini grupa.

Neke od vrsta algoritama klaster analize su:

- samo-organizirajuće neuralne mreže (Kohonen-ove mreže)
- probabilističke metode (AutoClass algoritam)
- algoritam k-srednjih vrijednosti (k-means)
- Joining tree clustering

Metodologija klaster analize raspisana je u šest koraka:

- određivanje ciljeva klaster analize
- određivanje istraživačkog obrasca
- određivanje pretpostavki
- formiranje i procjena broja klastera
- interpretacija klastera
- procjena klaster analize i profiliranje klastera

Mnogi su razlozi za korištenje klaster analize. Primjerice prilikom istraživanja tržišta za potrebe marketinške kampanje, kupci se segmentiraju po raznim obilježjima. Tako ta obilježja mogu biti spol, dob, kupovne navike, proizvodi koje kupac najčešće kupuje. Nakon što se svi kupci koji su predmet istraživanja segmentiraju u homogene skupine, ciljano će se određivati marketinški pristup prema svakoj skupini kupaca.

Broj klastera koji će se koristiti za analizu određuje se najčešće preko Elbow metode. Elbow metoda je metoda koja pomaže u odabiru optimalnog broja klastera prilagođavanjem modela rasponom vrijednosti za k. Grafičkim prikazom ćemo pažljivo ispitati vrijednosti. U nekom trenutku će se grafikon smanjivati, odnosno počet će nalikovati ruci, tada je "lakat" (točka savijanja na krivulji) dobar pokazatelj da temeljni model najbolje odgovara u tom trenutku ((Ekonomski fakultet Subotica, predavanje, 2015.).

2.4.2. Metode za predviđanje događaja

Za predviđanje događaja su metode pomoću kojih se želi predvidjeti mogućnost da se neki događaja ostvari ili ne. Razlika između metoda za predviđanje vrijednosti je u tome što ne koristi statističke pristupe kojima bi se rezultati oslanjali na distribuciji unutar podataka. Ova metoda je vođena atributima podataka i oslanja se na odnose u podacima približnih značajki. Cilj joj je kreirati najbolji model za buduće ulazne podatke na koristeći pravila za odlučivanje temeljem podataka prošlosti (Akar i Gungor, 2012.). Najpoznatije metode su stabla odlučivanja, a često se koriste i ostale, a to su neuronske mreže i logit regresija. Tako je čest primjer upravo gdje telekomunikacijska društva koriste ove metode kako bi predvidjeli hoće li neki od klijenata prijeći kod konkurencije.

2.4.3. Metode za predviđanje vrijednosti

Metode za predviđanje događaja su metode bazirane na statističkim formulama koje se koriste za predviđanje nekih događaja i numeričku procjenu u budućnosti. Uključuje estimaciju i kao rezultat vraća vrijednosti koje se ne nalaze u setu podataka koji je odabran za rudarenje. Također uključuje analizu obrazaca i odnosa u podacima kako bi se razvio model koji se može koristiti za procjenu ili predviđanje vrijednosti ciljane varijable na temelju vrijednosti drugih ulaznih varijabli. Najčešće korištena metoda za predviđanje vrijednost je regresijska analiza. Ostale značajne metode koje se također koriste su neuronske mreže i metode vremenskih serija. Primjer gdje se može koristiti metoda predviđanja vrijednosti je predviđanje kretanja cijena dionica na tržištu kapitala (Provost i Fawcett, 2013.).

3. Klasifikacija podataka i algoritmi šuma

Klasifikacija je jedna od glavnih metoda za rudarenje podataka te se primjenjuje u mnogim prilikama i na različitim tipovima podataka koji su dostupni u gotovo svim sferama kako naših života, tako i života organizacija i njihovih poslovnih jedinica (Saritas i Yasar, 2019.).

Klasifikacija se može odnositi i na kategorizaciju. To je proces u kojemu se objekti prepoznaju, diferenciraju te na kraju interpretiraju kroz dobivene rezultate. Algoritmi koji karakteriziraju klasifikaciju i koji se koriste za izgradnju modela rudarenja podataka se nazivaju klasifikatori. Termin klasifikator se također može koristiti i za niz matematičkih funkcija koje su ugrađene u algoritme korištene u klasifikaciji podataka te služe za diferenciranje podataka određenim grupama. U terminologiji rudarenje podataka, klasifikacije se smatra i dijelom nadziranog strojnog učenja, odnosno strojnog učenja gdje je dostupan set podataka za trening modela s ispravno određenim pojavama unutar seta podataka. Klasifikacija je i metoda rudarenja podataka koja se koristi korak-po-korak metodologijom kako bi odredila rezultat novog podatka koji dolazi u informacijski sustav. Stabla koja su generirana klasifikacijom podataka su upravo to. Svako stablo se grana te podaci putuju kroz grane stabla te ovisno o njegovim atributima prolazi kroz krošnje te dolazi do krajnjeg cilja koji simbolizira rezultat (Kamegh, 2015.).

Na rezultate klasifikacije utječu svi promatrani atributi unutar seta podatka koji je odabran kao prigodan za rudarenje (Kraljević i Staničić, 2020.).

Klasifikacija podataka je proces u dva koraka, korak učenja i korak predviđanja. U koraku učenja model se razvija na temelju zadanih podataka. U koraku predviđanja, model se koristi za predviđanje odgovora za dane podatke. Stabla odlučivanja su jedan od najlakših i najpopularnijih klasifikacijskih algoritama za razumijevanje i tumačenje (Munandar i Winarko, 2015.).

3.1. Ciljevi klasifikacije podataka

Glavni cilj klasifikacije podataka je izrada modela korištenje algoritama tako da ono rezultira mogućnošću da se neki od događaja predvidi uz visoku vjerojatnost.

Klasifikacije će korištenjem pravila podatke pridruživati različitim skupinama, dok će pravilima predikcije na temelju atributa tih podataka pokušati što točnije predvidjeti buduće događaje na temelju povijesnih događaja (Kraljević i Staničić, 2020.).

Predviđanje kao cilj klasifikacije je segmentiranje novih podataka u klase. Primjerice, implementacijom takvog modela možete pacijente neke zdravstvene ustanove automatski naručiti na odjel liječenja ovisno o zdravstvenim kartonima koji su dostupni.

Jedan od ciljeva klasifikacije podataka bi bio otkrivanje obrazaca unutar nekog seta podataka. Na primjeru banke možemo klasificiranjem podataka određene transakcije svrstati u grupe i tako uočiti potencijalne prevare i spriječiti pranje novaca.

Deskripcija kao cilj podataka bi bila razumjeti podatke kroz skupine sličnih karakteristika. Na primjeru banke bi se tako mogle promatrati platne poruke u stalnom vremenu kako bi se potencijalne poruke zaustavile i plaćanje ne bi prošlo.

Jedan od bitnijih ciljeva koji je već navođen kroz rad bi trebala biti i težnja za implementacijom modela u informacijske sustave s rezultatom poboljšanja učinkovitosti donošenja poslovnih odluka unutar organizacija. Na primjeru banke to može biti implementacija modela koji utvrdio hoće li klijent uspješno vračati kredit ili ne.

3.2. Algoritmi šuma

Algoritmi šuma su skupovi stabala odlučivanja i koriste se za otkrivanje logičkih pravila i predviđanje događaja točnije od pojedinačnih stabla odlučivanja. Služe za izgradnju modela koji uključuju velike broj eve stabala odlučivanja te koriste veliku memoriju i računalne resurse s ciljem postizanja visoke točnosti modela. Međutim, veliki broj stabala ne znači nužno da će algoritam kreirati točne rezultate. Stoga su u njih uključeni algoritmi koji će velike šume skratiti, i pretvoriti ih u takozvane podšume. Tako model se temelji na stablima koja su imala najveću točnost unutar algoritma šuma. Tim načinom pospješujemo točnost modela od izvorne šume stabala.

Glavni cilj algoritma šuma je odabrati što manji broj stabala koji će je tvoriti, uz uvjet da taj broj stvara najveću točnost i reprezentativnost izgrađenog modela. Svako stablo

odlučivanja unutar šume djeluje kao zaseban klasifikator. Klasifikacija se izvodi na temelju predviđanja i točnosti svakog stabla u šumi. Ako u šumi postoje stabla koja su značajno različita jedna od drugih, tada neka stabla mogu otkriti logička pravila za jedan skup testnih podataka, a druga stabla mogu otkriti pravila koja vrijedi za drugi skup podataka što za rezultat daje bolju generalizaciju za šumu (Kumar i Pati, 2022.). Algoritam šuma se može smatrati i nadogradnjom algoritma stabla odlučivanja. Neki od najčešćih algoritama šuma korišteni za rudarenje podataka su OptimizedForest i RandomForest.

Glavne značajke algoritama šumi su sljedeće:

- Algoritam šuma može se koristiti za klasifikaciju i regresijske zadatke
- Daje visoke rezultate točnosti zbog korištenje velikog broja stabala
- Rješava probleme prljavih podataka zadržavajući visoku točnost
- Ne dozvoljava over-fitting nad podacima
- Pogodan je za iznimno velike skupove podataka (NewGenApps, 2018.).

3.3. Karakteristike uspješnog modela klasifikacije podataka

Uspješni model klasifikacije podataka karakteriziraju sljedeće značajke:

- Točnost klasifikacije – model uz što veću točnost mora klasificirati podatke u konačnoj uporabi. Problemi koji mogu nastati su over-fitting i under-fitting. Over-fitting je problem do kojega dolazi kad je model na treniranim podacima naučen na previše šuma u podacima te tako negativno utječe na generalizaciju novih podataka u stvarnim poslovnim situacijama. Under-fitting je problem koji nastaje kada model ni na testnim ni na stvarnim podacima ne može biti istreniran te ne rezultira generalizacijom podataka (Akar i Gungor, 2012.).
- Vrijeme – vrijeme za klasifikaciju kao i za izradu modela mora biti prihvatljivo. Poduzeća si ne mogu priuštiti velike količine vremena kod izrade modela za stvarne poslovne prilike
- Razumljiv – model dobiven klasifikacijom se mora moći jasno i jednostavno interpretirati
- Adaptivnost – model mora biti vjerodostojan i točan za razne setove podataka. Ne smije značajno reagirati u manjim odstupanjima i šumovima u novim podacima (Bowen i suradnici, 2021.).

4. Genetski algoritam

Genetički algoritam je metoda strojnog učenja iz kategorije evolucijskih algoritama. Ovaj algoritam oponaša evolucijski proces te kao takav traži optimalno rješenje na postavljeni problem. Funkcionira tako da koristi generacije genoma koje progresivno postaju prilagođenije okolini i cilju vezanom u problem, odnosno prilagođava se funkciji cilj (Fitness Function) koja predstavlja sposobnost prilagođavanja i reproduciranja. Kao metoda strojnog učenja prvi puta je javnosti predstavljena 1975. godine kada rezultate na ovom području prezentira John Holland, u literaturi često nazivan i kao izumitelj genetičkog algoritma (Kumar i suradnici, 2010.).

Glavne karakteristike genetičkog algoritma su:

- Početna populacija – najčešće se dobiva nasumičnim odabirom (Complete chaos); svaka jedinka dobiva vlastiti genom
- Funkcija cilja (engl. Fitness function) – vraća vrijednost „sposobnosti" određene jedinke
- Jedinke se poredaju po vrijednostima dobivenim funkcijom cilja te se odbacuju loše jedinke dok se dobre ostavljaju
- Križanje – kombiniraju se dva genoma preostalih jedinki te se tako tvore dva potpuno nove jedinke; Na ovaj način dolazimo do nove generacije gdje se opet provjerava funkcija cilja
- Generacija je iteracija petlje algoritma
- Mutacija – mijenja se vrijednost jednog gena, koristi se rijetko; vrši se slučajnim odabirom ili nad svim genima(potpuna mutacija)

Posao genetskog algoritama može se opisati jednom rečenicom: nakon što se generira početna populacija, genetski algoritam ciklički obavlja selekciju boljih jedinki koje potom sudjeluju u reprodukciji, sve dok nije zadovoljen uvjet završetka evolucijskog procesa (Golub, 1997.).

Genetski algoritam u široj upotrebi pojma, je svaki populacijski model koji koristi operatore selekcije i rekombinacije za generiranje novih generacija uzorka u okruženju u kojem se provodi (Whitley, 1994.).

4.1. Prirodni evolucijski proces

Charles Darwin dokazuje da unutar živućeg svijeta, jedinke koje ga čine stvaraju više potomaka od njihove populacije. Tim naukom, broj jedinki u populaciji trebao bi eksponencijalno rasti. Međutim, Darwin je također otkrio da suprotno tome, priroda procesom selekcije održava broj jedinki neke populacije te tako drži konstantu. Tako procesom selekcije uspijevaju preživljavati samo one jedinke s najboljim karakteristikama. Evolucija je neprekidan proces prilagođavanja živih bića na svoju okolinu, tj. na uvjete u kojima žive. U prirodi vlada nemilosrdna borba za opstanak u kojoj pobjeđuju najbolji, a loši umiru (Golub, 1997.).

S obzirom na to da se uvjeti u kojima živa bića egzistiraju konstantno mijenjaju, jedinke se moraju prilagođavati kako bi opstale. Svaka buduća generacija jedine preuzima najbolja svojstva prijašnje generacije te se kroz svoj vijek tak svojstva dodatno mijenjaju kako bi svaka generacija bila prilagođena uvjetima u kojima živi. Jedinke koje su naslijedile loša svojstva, najvjerojatnije neće opstati, a samim time u neće opstati ni svojstva koja ih karakteriziraju. Tako priroda osigurava da dobra svojstva opstaju i budu prenesena na sljedeću generaciju jedinki.

Svojstva neke jedinke se prenose kromosomima. Oni se nalaze u jezgrama stanica, a to znači da svaki organizam ima pohranjene sva svojstva koja posjeduje. Sve karakteristike nekog svojstva jedine, nalaze se u dijelu kromosoma koji se naziva gen. Skup informacija koje karakteriziraju jedno svojstvo zapisano je u jedan djelić kromosoma koji se naziva gen. Kromosomi se nasljeđuju od strane muškog roditelja i ženskog roditelja. Znači da za svako svojstvo neke jedine postoje dvije informacije. Takvi parovi gena nazivaju se aleli. U paru gena koji je naslijeđen, geni mogu biti ravnopravni ili neravnopravni. U ravnopravnom paru gena, jedan je superioran dok je jedan inferioran. Ako su geni ravnopravni, kod nasljeđivanja svojstvo će biti miješani gen oca i majke, dok u neravnopravnom genu se nasljeđuje onaj koji je superioran. Sva svojstva jedinke nasljeđuju se kroz više kromosoma.

Struktura kromosoma otkrivena je 1953. godine. Građen je od molekule DNK (deoksiribonukleinska kiselina). DNK je građena od spirala koji su spojeni nitima sačinjenih od dušičnih baza adenina, gvanina, timina i citozina. Tako su tim bazama dani akronimi A, G, T, C. Daljnjim istraživanjima otkriveno je da su upravo te baze elementi kromosoma koji nose informacije o svojstvima. Informacije se prenose tako da prilikom repliciranja stanice,

spirale molekule DNK razmotaju i vežu se na parove ostalih bazi DNK koje se nalaze u jezgrama stanica.

Metoda kojom se genetski materijal prenosi naziva se križanje. Kako se prenosi genetski materijal, tako se i prenose informacije o svojstvima neke jedinke. Ako u nekom od slučaja dođe do nenamjerne promjene naslijeđenog gena, takva promjena se naziva mutacija. S obzirom na to da neki geni mogu mutirati lakše, oni se nazivaju nestabilni geni, a geni koji mutiraju teže se nazivaju stabilni geni. Neka svojstva koja individualna jedinka stekne u životu nisu prenosiva.

4.2. Odabir jedinke za rješenje – Inicijalna populacija

Populacija se smatra kao set kromosoma. Točnije, inicijalna populacija se kreira tako što se kromosomi neke jedine zapisuju u binarnom znakovlju (Back i Schwefel, 1993.). Prilikom inicijalizacije generira se početna populacija jedinki. Obično se početna populacija generira slučajnim odabirom rješenja iz domene, iako je moguće početnu populaciju generirati uniformno. U tom slučaju, sve jedinke su iste pa u početku evolucijskog procesa genetski algoritam nije učinkovit pa se taj postupak ne preporučuje. Druga metoda je usaditi početno rješenje u početnu populaciju dobiveno nekom drugom optimizacijskom metodom (Golub, 1997.). Populaciju inicijalnog rješenja gledamo kao veličinu populacije binarnih vektora – VEL_POP(t), gdje t označava vrijeme ili redni broj generacije. Najčešća metoda je korištenje veličine inicijalne populacije kao konstante za genetski algoritam, odnosno da se veličina populacije kroz broj iteracija ne mijenja - VEL_POP(t)=konst=VEL_POP. Vrijednost koja je sadržava u VEL_POP je parametar algoritma. Proces početnog odabira populacije jednostavan je tako da se VEL_POP populira slučajnim brojevima u sljedećem intervalu - [0, 2n -1], odnosno u binarnim zapisima duljine n bitova. Evolucijski proces genetskog algoritma ponavlja se u iteracijama sve dok se ne postigne funkcija cilja. Često se broj iteracija definira prije početka izvedbe genetskog algoritma. Trajanje izvedbe genetskog algoritma od inicijalizacije početne populacije do zadovoljavanja funkcije cilja, može se smatrati znanom vrijednošću s obzirom na to da su vremena izvođenja iteracija konstante.

Veličina populacije ovisi o prirodi problema, ali obično sadrži nekoliko stotina ili tisuća mogućih rješenja. Tradicionalno, populacija se generira nasumično, pokrivajući cijeli niz mogućih rješenja. Povremeno se rješenja mogu "zasijati" u područjima gdje je vjerojatno da će se pronaći optimalna rješenja (Kumar i suradnici, 2010.).

Prilikom inicijalizacije generira se početna populacija s pomoću generatora slučajnih brojeva (Golub, 1997.).

Posao genetskog algoritma naviše ovisi o početnoj populaciji jer cijeli proces počinje upravo iz nje sve dok se ne zadovolji uvjet završetka evolucije. Slika broj 5 prikazuje odabir inicijalne populacije genetskim algoritmom.

Slika 5.: Primjer generirane prve populacije

rbr	kromosom	x	f(x)
0	10001010011100000111111010011100	8.1558	0.1381
1	11000001001000011101000011011110	50.8844	0.5127
2	01110010101111011111001011111000	-10.3578	0.3813
3	10011110101110000110010000001010	24.0002	0.3712
4	10101100101110100000010010110100	34.9427	0.5727
5	10001111110110110011100100010110	12.3878	0.8521
6	11001000110000000110001011010000	56.8371	0.5234
7	01010100111001011101001100000010	-33.6736	0.4785
8	11100001101100011101010001001100	76.3239	0.4970
9	11000110001011101110110011001110	54.8307	0.4702
10	00100011011101011111100000101000	-72.2962	0.5129
11	00100001110001110011100101111010	-73.6108	0.4890
12	10000011101100110000010101100100	2.8901	0.9308
13	00101110001101100010010000000110	-63.8973	0.4897
14	10001110101010111000101100000000	11.4610	0.2667
15	10000101011001001000111101110010	4.2131	0.2386

Izvor: Golub, 1997.

3 su glavne metode generiranje početne populacije:

- Nasumična inicijalizacija - U ovoj metodi, početna populacija se generira nasumično. Svaka jedinka u populaciji stvorena je nasumičnim dodjeljivanjem vrijednosti svojim genima u skladu s ograničenjima i zastupljenošću problema. Ovaj pristup je jednostavan i često se koristi kada nema prethodnog znanja o domeni problema.
- Uniformna inicijalizacija: Ako postoji određeno prethodno znanje ili stručnost u domeni, može se koristiti uniformna inicijalizacija za stvaranje početne populacije. To uključuje ravnomjerno uzorkovanje pojedinaca iz domene pretraživanja, čime se osigurava dobra pokrivenost prostora rješenja. Ovaj pristup može biti koristan kada postoji razumijevanje distribucije potencijalnih rješenja
- Heuristička inicijalizacija: Heuristika se može koristiti za generiranje početne populacije koja uključuje znanje specifično za domenu ili ograničenja specifična za problem. Na primjer, ako znate da su određene značajke ili karakteristike važne

za problem, možete dizajnirati heuristiku za stvaranje pojedinaca s tim značajkama. Ovaj pristup može pomoći algoritmu da brže konvergira prema dobrim rješenjima (Katoch i suradnici, 2021.).

4.3. Funkcija cilja

Funkcija cilja (engl. Fitness function) je funkcija genetskog algoritma koja vraća vrijednost „sposobnosti" određene jedinke. Funkcija cilja koju se još naziva i funkcijom sposobnosti, funkcijom dobrote ili eval u najjednostavnijoj interpretaciji ekvivalent je funkciji f koju treba optimizirati:

$$\text{dobrota}(v)=f(x) \tag{1}$$

gdje je binarni vektor v realan broj. Što neka jedina ostvari bolji rezultat tijekom evaluacije funkcijom cilja, njena šansa preživljavanje postaje veća. Funkcija cilja smatra se kao ključ tijekom cijelog procesa evolucije genetskog algoritma. Za problem koji je stavljen pred genetski algoritam, najveća poteškoća je definiranje funkcije cilja jer ona mora odražavati problem koji se njome želi riješiti. Kroz proces evolucije, dobro dizajniran genetski algoritam će iz generacije u generaciju generirati populaciju čiji će ukupni rezultati funkcije cilja (D) biti sve bolji te će tako i prosječna vrijednost rezultata funkcije cilja (D') biti sve bolja (Golun, 1997.). Na slici broj 6 vidimo formule za izračun ukupne dobrote i prosječne dobrote populacije.

Slika 6.: Formule ukupne dobrote populacije i prosječne dobrote populacije

$$D = \sum_{i=1}^{VEL\_POP} dobrota(v_i), \qquad \overline{D} = \frac{D}{VEL\_POP}.$$

Izvor: Golub, 1994.

Funkcija cilja je funkcija kojom se evaluiraju jedinke kroz sve iteracije genetskog algoritma i započinje odmah nakon kreiranja inicijalne populacije. Svaki kromosom jedinke se smatra potencijalnim kandidatom za dostizanje optimalne solucije te se kao takav svaki od njih mora provući kroz funkciju.

Genetski algoritam će raditi ispravno ako se funkcija cilja odredi na odgovarajući način imajući na umu da je odabir funkcije vrlo subjektivan i ovisan o problemu. Obično je potrebna funkcija cilja koja je pozitivna. Pozitivna funkcija cilja prilikom evaluacije jedinki

dodjeljuje veće vrijednosti onim jedinkama koje više teže k optimalnom rješenju. Drugi pristup je korištenje rangiranja članova u populaciji na temelju njihovog rezultata ostvarenog funkcijom cilja. Prednost ovog pristupa je u tome što funkcija cilja ne mora biti točna, sve dok može pružiti točne informacije o rangiranju. Treća metoda je turnirska selekcija. Ove metode bit će objašnjene dalje u radu.

4.4. Selekcija jedinki za preživljavanje

Selekcijom se u genetskom algoritmu osigurava prenošenje najboljih svojstava na sljedeću generaciju roditelja. Odabiru dobre jedinke za sljedeću iteraciju. Tako se dobar genetski materijal uspije sačuvati i prenosi se na sljedeću generaciju, dok se loše jedinke odbacuju, odnosno odumiru. Tako se dobri geni ili dobri genetski materijal sačuvaju i prenose na slijedeću populaciju, a loši odumiru.

Genetske algoritme, s obzirom na vrstu selekcije, dijelimo na generacijske i eliminacijske. Generacijski genetski algoritam u jednoj iteraciji raspolaže s dvije populacije (što je ujedno i nedostatak generacijskog GA), jer se odabiru dobre jedinke iz stare populacije koje čine novu populaciju i nakon selekcije sudjeluju u procesu reprodukcije. Karakteristične vrste selekcija koje koristi generacijski GA su: jednostavna selekcija, selekcija po rangu i turnirska selekcija. S druge strane eliminacijska selekcija je karakteristika eliminacijskog genetskog algoritma (Golub, 1997.)..

4.4.1. Jednostavna selekcija

Genetski algoritam koji koristi jednostavnu selekciju naziva se generacijski genetski algoritam (Golub, 1997.). Jednostavna selekcija koja se na engleskom naziva roulette wheel parent selection) služi za generiranje nove populacije iz prethodne populacije, a sadržat će isti broj jedinki kao i prijašnja populacija. Nova generacija se može prikazati na sljedeći način:

$$VEL\_POP(P'(t))=VEL\_POP(P(t-1)) \quad (2)$$

Jednostavna selekcija generira novu populaciju jedinki čije su vrijednosti selekcije proporcionalne rezultatima dobivenih funkcijom cilja.

Postupak jednostavne selekcije je sljedeći:

- Ako je rezultat funkcije cilja negativni, dodaje se unaprijed određena konstanta kako bi rezultat dobio pozitivni ishod

dobrota(v)=f(x)+C, tako da je (3)

dobrota(v)≥0

- izračunaju se sve funkcije vrijednosti jedinki u populaciji
- računanje ukupne funkcije dobrote po formuli na slici 7.
- računanje kumulativne funkcije cilja za svaki kromosom po formulama prikazano je na slici 7

Slika 7.: Formule za izračun kumulativnih funkcija cilja

$$q_k = \sum_{i=1}^{k} dobrota(v_i)\text{, gdje je } k=1,2,...VEL\_POP, \qquad p_k = \frac{dobrota(v_k)}{D}$$

Izvor: Golub, 1994.

Tako se osigurava da je selekcija proporcionalna funkciji cilja. Slika broj 8 prikazuje formule za izračun kumulativnih funkcija cilja.

- generira se slučajni realan broj r u intervalu (0,D) i potraži se i-ti kromosom za koji vrijedi da je r∈(qi-1, qi) i prenosi se u slijedeću populaciju.

Prateći ovaj postupak, jedan kromosom može više puta biti predstavljen u novoj generaciji. Što neka jedinka vraća bolji rezultat funkcije cilja, veća je vjerojatnost da će biti izabrana u svakoj novoj generaciji roditelja. Gotovo sigurno je da će s e neka jedinka pojaviti dva puta u generaciji ako za nju vrijedi da je njen rezultat funkcije cilja dvostruko veći od ukupne funkcije cilja cijele populacije.

4.4.2. Selekcija po rangu

Cilj metode selekcije po rangu je, umjesto rangiranja po samoj vrijednosti funkcije cilja neke jedinke, jedinke poredati po položaju u poretku rezultata funkcije cilja. Rangiranje, odnosno sortiranje jedinki može biti padajuće ili rastuće s obzirom na vrijednost funkcije cilja. Sortiranjem padajućom dobrotom, najbolja je ona jedinka čiji indeks iznosi i=1, dok je najgora ona čiji indeks iznosi i=N (N predstavlja ukupni broj jedinki). Za rangirajuću selekciju nije važan rezultat funkcije cilja, već njen odnos prema rezultatima funkcije cilja ostalih jedinki. Isto tako nije važno je li funkcija cilja poprimila negativne vrijednosti. Ova selekcija nema ograničenja i jednaka je funkciji cilja:

d(x)=f(x) (4)

Rješenje x1 je bolje od rješenja x2, ako je f(x1)>f(x2). Kod korištenja selekcije po rangu, bitno je znati da onda konzumira značajan dio vremena cjelokupnog procesa s obzirom na to da se ponavlja u svakoj iteraciji. Uzeći u obzir tu činjenicu i da je svako sortiranje posao koji zahtjeva mnogo vremena, selekcije po rangu se u pravilu rijetko koriste. Postupak sortiranja značajno usporava proces selekcije i ako je ikako moguće, treba ga izbjeći.

4.4.3. Turnirska selekcija

Turnirsku selekciju prvi je puta predstavio Brindle 1983. godine (Katoch i suradnici, 2021.). Genetski algoritam koji koristi turnirsku selekciju u svakom koraku stvara novu populaciju iz stare populacije tako da generira veličinu populacije s istom vjerojatnošću jedinki iz starije populacije međusobno ih uspoređuje te najbolju jedinku iz usporedbe prosljeđuje k reprodukciji.

Turnirska selekcija provodi se kao sličan oblik rangiranja. Sastoji se od dva koraka. Prvi korak je uzeti dvije jedinke i usporediti ih, odabrati najbolju jedinku, a potom u drugom koraku nad njima provesti modifikatore za stvaranje nove generacije (križanje i mutacija) kako bi se stvorila nova generacija. Ovi procesi se ponavljaju dok se ne stvori dovoljna količina jedinki u novoj generaciji (Whitely, 1994.).

Drugi oblik turnirske selekcije je eliminacijska selekcija koji eliminira najlošiju jedinku i nadomješta je potomkom nastalim dviju preživjelih jedinki (Golub, 1997.). Slika broj 8 prikazuje jednostavnu turnirsku selekciju genetskim algoritmom.

Slika 8.: Jednostavna turnirska selekcija

```
Genetski_algoritam_s_jednostavnom_turnirskom_selekcijom(){
   dok(nije_zadovoljen_uvjet_završetka_evolucijskog_procesa){
      selektiraj_slučajno_dvije_jedinke();
      Roditelj_A = bolja_jedinka_od_dvije_selektirane();
      Eliminirana_A = lošija_jedinka_od_dvije_selektirane();
      selektiraj_slučajno_dvije_jedinke_između_preostalih_jedinki();
      Roditelj_B = bolja_jedinka_od_dvije_selektirane();
      Eliminirana_B = lošija_jedinka_od_dvije_selektirane();
      križanjem_roditelja_nadomjesti_eliminirane_jedinke();
      mutiraj_djecu(pm);
      evaluiraj_djecu();
   }
}
```

Izvor: Golub, 1994.

4.5. Generiranje nove populacije jedinki

4.5.1. Križanje

Križanjem u genetskom algoritmu osigurava se nesmetan prijenos genetskog materijala, a njegovim prijenos uključuje i prijenos svojstva populaciju na iduću generaciju. Oba

roditelja sudjeluju u procesu križanja. Najbitnija karakteristika križanja je ta da nova generacija nasljeđuje karakteristike prijašnje. Vrijedi pravilo, ako je prijašnja generacija dobra, tada će i sljedeća biti dobra. Križanje se definira brojem prekidnih točaka. Najčešća i najjednostavnija metoda križanja u genetskom algoritmu je uniformno križanje. To je križanje kojoj se broj prekidnih točaka definira kao b-1, gdje b predstavlja broj bitova. Vjerojatnost nasljeđivanja gena jednog roditelja je 0,5 što znači da jednaka vjerojatnost da će dijete naslijediti gene oba roditelja. Ako postoje razlike u vjerojatnosti nasljeđivanja gena, tada dolazimo do p-uniformnog križanja. Ako primjerice vrijedi p=0.3, tada je vjerojatnost nasljeđivanja gena prvog roditelja 30%, a vjerojatnost nasljeđivanja gena drugog roditelja 70%.

Tablica 1.: Primjer vjerojatnosti odabira gena

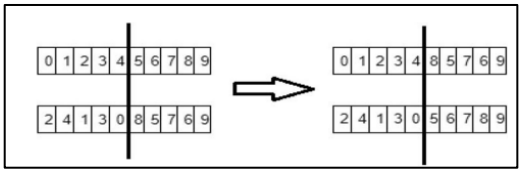
Gen	1	2	3	4	5	…	b-2	b-1	b
P	0,1	0,5	0,9	0,1	1	…	0,7	0,6	0,1

Izvor: Izrada autora prema Golub, 1994.

U tablici broj 1 vidimo primjer vjerojatnosti nasljeđivanja gena u genetskom algoritmu. Vjerojatnost da će 5. gen biti naslijeđen od prvog roditelja je 100%. Za gene 1. i 4. vrijedi da je vjerojatnost da će biti naslijeđeni od prvog roditelja 10%, a drugog 90% (Golub, 1997.).

Za broj točki prekida se najčešće koristi Single point crossover, križanje gena između točki križanja i mijenjanje individualnih nasumičnih gena. Na slici broj 9 vidimo kako izgleda križanje gena s korištenjem Single point crossover metode. Samo geni iza točke prekida će biti zamijenjeni jedni drugima.

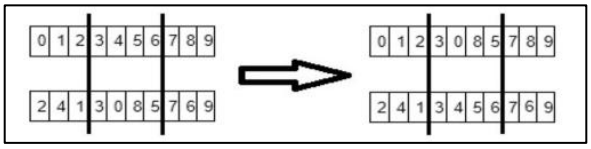
Slika 9.: Single point crossover



Izvor: Golub, 1994.

Na slici broj 10 je predstavljena metoda križanja između dvije točke prekida. Za nju vrijedi da će samo geni između dvije točke biti zamijenjeni i tako stvoriti novu generaciju populacije.

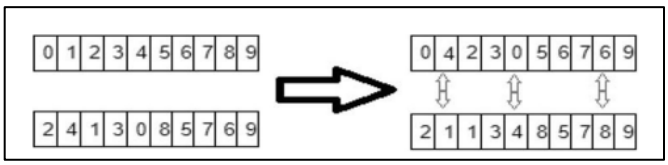
Slika 10.: Križanje između dvije točke prekida



Izvor: Golub, 1994.

Na slici broj 11 je prikazana metoda individualnog mijenjanja gena gdje se geni nasumice odabiru i mijenjaju tvoreći novu populaciju.

Slika 11.: Individualno mijenjanje gena



Izvor: Golub, 1994.

4.5.2. Mutacija

Mutacija u genetskom algoritmu predstavlja slučajna promjene na genima. Mutacija djeluje samo kod jedne jedinke i rezultira izmijenjenim svojstvima te jedinke. Vjerojatnost mutacije se određuje parametrima genetskog algoritma. Četiri su glavne vrste mutacije, a to su: jednostavna mutacija, miješajuća mutacija, potpuna miješajuća mutacija i invertirajuća miješajuća mutacija. Jednostavna mutacija mijena svaki kromosom koristeći jednaku vjerojatnost. Miješajuća mutacija koja koristeći slučajan odabir kromosoma, odabire kromosom i mijenja ga. Koristi dvije točke za mutaciju, te ih izmiješa i kao rezultat se dobiva invertirana miješajuća mutacija, a ako nove gene generira slučajno, tada se dobiva potpuna miješajuća mutacija.

4.5.3. Elitizam

Elitizam u genetskom algoritmu predložen je 1975. godine od strane K. D. Jonga kako bi performanse selekcije bile učinkovitije (Katoch i suradnici, 2021.). Zbog mnogo iteracije genetskog algoritma, postoji opasnost da se najbolje jedinke u nekoj od iteracije izgube zbog križanja i mutacije. Iz tog razloga je trebalo smisliti način kako zaštiti najbolje jedinke od izmjena i eliminacije. Stoga je smišljen mehanizam imena elitizam (Golub, 1997.). Elitzam

osigurava da se elitist, odnosno jedinka s najboljim karakteristikama koju se ne smije izgubiti, nalazi u sljedećim iteracijama genetskog algoritma. Ako pojedinac s najvišom vrijednošću fitnessa cilja nije prisutan u sljedećoj generacije nakon normalne selekcijske procedure, tada se u sljedeću generaciju automatski uključuje elitist (Katoch i suradnici, 2021.).

5. Izvori podataka i priprema za analizu

Za izradu analize i usporedbe uspješnosti algoritama za klasifikaciju podataka, u ovom radu će biti obrađeno 5 setova podataka iz poslovnih primjena. Podaci se odnose na područje bankarstva, financija, turizma, i zabave. Podaci su preuzeti sa stranice www.kaggle.com te su javno dostupni. Kako bi se podaci adekvatno analizirali i pripremili za otkrivanje znanja iz podataka, koristit će se alati Microsoft Excel i Weka.

5.1. Popis setova podataka

- Marketinška kampanja portugalske banke (1) - Podaci se odnose na izravnu marketinšku kampanju jedne portugalske bankarske institucije. Marketinška kampanja temeljila se na telefonskim pozivima. Često je bilo potrebno više od jednog kontakta s istim klijentom, kako bi se vidjelo hoće li klijente ili neće koristiti bankarske proizvode.

- Netflix pretplata (2) – set podataka s kojim se želi predvidjeti hoće li se osoba odlučiti preuzeti Netflix aplikaciju ili ne.

- Predviđanje bankrota kompanije (3) – Set sadrži podatke prikupljene između 1999 godine i 2009 godine. Podatke je prikupljao Tajvanski ekonomski časopis (engl. Taiwan Economic Journal) s ciljem predviđanja bankrota poduzeća.

- Predviđanje odustajanja od korištenja usluga turističke agencije (4) – Set sadrži podatke o klijentima turističke agencije s ciljem predviđanja odustanka od putovanja.

- Predviđanje klijenata koji će odustati od korištenja kreditne kartice (5) – Set sadrži podatke o klijentima banke s ciljem predviđanja odustanka od korištenja kreditne kartice.

Svi setovi podataka će u zadnjem poglavlju analize rezultata biti numerirani brojevima od 1 do 5, počevši od prvog seta u ovome poglavlju 'Marketinška kampanja portugalske banke kao broj 1, do zadnjeg seta navedenog u ovom poglavlju 'Predviđanje klijenata koji će odustati od korištenja kreditne kartice' kao broj 5.

5.2. Tumačenje varijabli i priprema za analizu

U ovom poglavlju rada će se definirati i objasniti varijable unutar setova podataka koji su korišteni za izradu rada. Za sve setove podataka koristit će se jednaki stil tablice za prikazivanje varijabli. U prvom stupcu se nalaze nazivi varijabli u setu. Drugi stupac objašnjava što pojedina varijabla predstavlja. U trećem stupcu su formati tj. tipovi koje modaliteta koje atribut može poprimiti. U zadnjem, četvrtom stupcu, vidimo modalitete atributa koji pobliže karakteriziraju instancu s obzirom na promatrani atribut.

Marketinška kampanja portugalske banke

Tablica 2.: Atributi seta marketinške kampanje portugalske banke

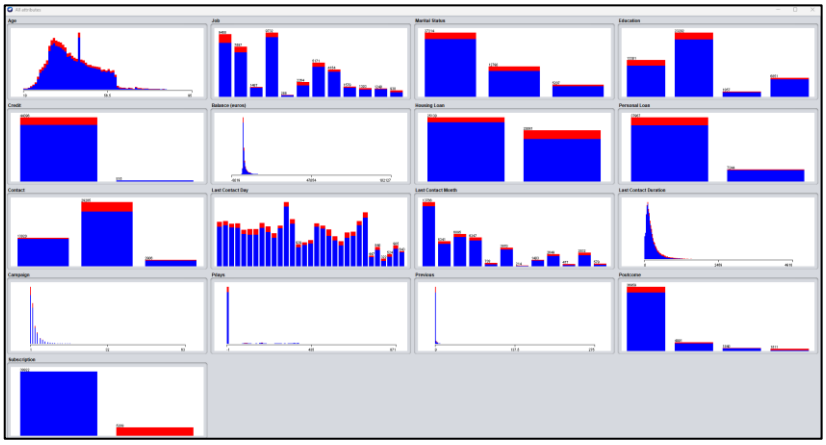
Atribut	**Objašnjenje atributa**	**Tip modaliteta**	**Modaliteti atributa**
Age	Dob ispitanika	Numerički	MIN: 18, MAX:95, MEAN: 40,94, STD.DEV: 10,62
Job	Vrsta zaposlenja	Nominalna	Menadžment, Tehničar, Poduzetnik, Građevinar, Nepoznato, Umirovljenik, Administracija, Uslužne djelatnosti, Samozaposlen, Nezaposlen, Kućanica, Student
Marital Status	Bračni status	Nominalna	U braku, Rastavljen, Slobodan
Education	Razina edukacije	Nominalna	Nepoznato, Osnovnoškolsko obrazovanje, Srednjoškolsko obrazovanje, Više obrazovanje
Credit	Atribut koji prikazuje ima li putnik kredita ili ne	Nominalna	Da, Ne
Balance (euros)	Prosječno godišnje stanje računa	Numerički	MIN: -8019, MAX: 102127, MEAN: 1362,27, STD.DEV: 3044,77
Housing Loan	Stambeni kredit	Nominalna	Da, Ne
Personal Loan	Nenamjenski kredit	Nominalna	Da, Ne
Contact	Vrsta komunikacije	Nominalna	Nepoznato, Mobilni telefon, Fiksni telefon
Last Contact Day	Datum (Dan) zadnje komunikacije	Nominalna	Od prvog do posljednjeg dana mjeseca
Last Contact Month	Mjesec zadnje komunikacije	Nominalna	Siječanj, Veljača, Ožujak, Travanj, Svibanj, Lipanj, Srpanj, Kolovoz, Rujan, Listopad, Studeni, Prosinac

Last Contact Duration	Vrijeme trajanja zadnje komunikacije	Numerički	MIN: 0, MAX: 4918, MEAN: 258,16, STD.DEV: 257,53
Campaign	Broj puta kontaktiranja klijenta u kampanju	Numerički	MIN: 1, MAX: 63, MEAN: 2,76, STD.DEV: 3,10
Pdays	Broj dana od kada je ispitanik zadnji puta kontaktiran	Numerički	MIN: -1, MAX: 871, MEAN: 40,20, STD.DEV: 100,13
Previous	Broj puta kontaktiranja klijenta od prošle kampanje	Numerički	MIN: 0, MAX: 275, MEAN: 0,58, STD.DEV: 2,303
Poutcome	Ishod prošle kampanje	Nominalna	Uspješna, Neuspješna, Nepoznato, Ostalo
Subscription	Pretplata na bankarske usluge (Ciljana varijbla)	Nominalna	1- Da, 2- Ne

Izvor: Izrada autora

U tablici broj 2 vidimo varijable seta podataka marketinške kampanje portugalske banke. Vidljivo je kako set sadrži 6 numerički tipova atributa, a 11 nominalnih tipova atributa. Za numeričke atribute koristili smo osnovna statistička obilježja. MIN predstavlja minimalnu vrijednost koju instanca može primiti (Primjerice, u atributu Last Contact Duration, minimalni broj sekundi komunikacije s klijentom je 0 sekundi ; ako klijent nije kontaktiran od strane banke). Max predstavlja maksimalnu vrijednost koju instanca može poprimiti (Primjerice, maksimalni broj sekundi komunikacije s klijentom je iznosio 4918 sekundi). MEAN predstavlja aritmetičku sredinu vrijednosti unutar atributa (Primjer, prosječno se s klijentima komuniciralo 258,16 sekundi). STD.DEV predstavlja prosječno kvadratno odstupanje neke vrijednosti atributa od aritmetičke sredine (Primjerice, prosječno odstupanje od aritmetičkog prosjeka trajanja komuniciranja s klijentom je 257,53).

Slika 12.: Raspodjela instanci po atributima – Kampanje portugalske banke



Izvor: Weka

Na slici broj 12, možemo vidjeti kako je alat Weka pomogla prilikom analize distribucije entiteta po atributima. Tako primjerice možemo vidjeti nominalnu varijablu 'Marital Status'. Najviše instanci pripada modalitetu osoba koje su u braku, točnije njih 27214 od kojih je većina odlučila koristiti pretplatu na proizvode banke. Najmanje instanci je u bračnom statusu 'Rastavljeno', točnije njih 5207 od kojih također većina koristi proizvode banke nakon kampanje. Za primjer objašnjenja druge vizualizacije odabran je atribut 'Last Call Duration'. Vidimo da najveći broj ljudi je komuniciralo između 100 do 126 sekundi, točnije 4123 osobe. Ciljni atribut u setu podataka je atribut 'Subscription' koji označava je li osoba stvarno odlučila koristiti proizvode banke ili ne.

Set je preuzet kao .csv file. Kako bi dobili podatke u tabličnom prikazu, koristili smo opciju alata Microsoft Excel 'Text to Columns'. Weka je varijablu Last Contact Day i ciljnu varijablu Subscription prepoznavala kao numeričke varijable pa se iz tog razloga na modalitete varijable Last Contact Day dodao string '-', a modaliteti varijable Subscription su zamijenjeni na sljedeći način: 1 u 'Yes', 0 u 'No'. Za izračun statističkih obilježja numeričkih varijabli, koristi smo opciju alata Excel 'Data Analysis'. Prazne vrijednosti numeričkih atributa zamijenjene su vrijednosti aritmetičke sredine za instance koje vrijednost imaju zapisanu. Nakon što je set pripremljen, spremljen je opet u .csv format kako bi bio čitljiv za alat Weka.

Netflix pretplata

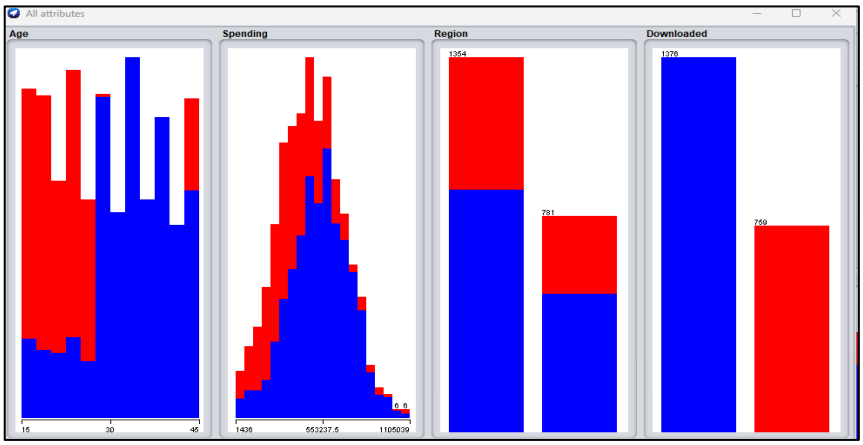
Tablica 3.: Atributi seta Netflix pretplate

Atribut	Objašnjenje atributa	Tip modaliteta	Modaliteti atributa
Age	Broj godina klijenta	Numerički	MIN: 15, MAX:45, MEAN: 29,76, STD.DEV: 8,88
Spending	Prosječna godišnja potrošnja klijenta	Numerički	MIN: 1436, MAX:1105038, MEAN: 477979, STD.DEV: 201834
Region	Mjesto stanovanja	Nominalni	Urban, Rural
Downloaded	Preuzeta Netflix aplikaciju ili ne (Ciljna varijabla)	Nominalni	Yes, No

Izvor: Izrada autora

U tablici broj 3 vidimo varijable seta podataka Netflix pretplate. Set sadrži 4 varijable od kojih su dvije numeričke, a dvije nominalne. Varijable Age i Spending su numeričkog tipa i za njih vidimo neka statistička obilježja. Tako primjerice za varijablu Age, najmanje vrijednost je 15 godina, dok je najv iša 45 godina. Prosjek godina ispitanih osoba je 29,76 godina, dok je prosječno kvadratno odstupanje od aritmetičke sredine 8,88 godina.

Slika 13.: Raspodjela instanci po atributima – Netflix pretplata



Izvor:Weka

Na slici broj 13, možemo vidjeti kako je alat Weka raspodijelila instance po atributima s obzirom na ciljnu varijablu. Primjerice, u atributu Region, 1354 promatrane instance dolaze

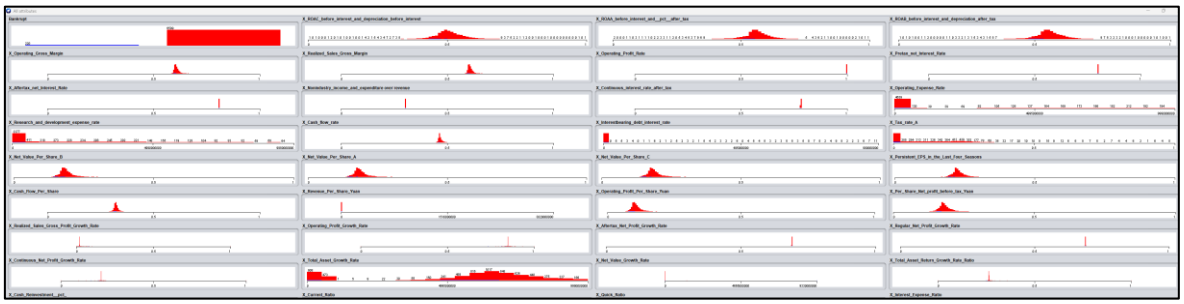
iz urbane sredine i većina se odlučila preuzeti aplikaciju Netflix, dok ih 781 dolazi iz ruralnih sredina te su također odlučili preuzeti Netflix aplikaciju. U atributu Spending vidimo da je najviše instanci u kategoriji prihoda između 442877 i 498057 te se također većina odlučila preuzet Netflix. Ciljni atribut ovog seta je atribut 'Downloaded' koji prikazuje je li osoba odlučila preuzeti aplikaciju ili ne.

Set je preuzet u .csv formatu te je opcijom alata Excel 'Text to Columns' pretvoren u tablični format kako bi se lakše rukovalo podacima. Instance koje nisu imale upisanu neku od vrijednosti nominalne varijable su uklonjene. Za numeričke varijable i izračun statističkih obilježja modaliteta koristila se opcija alata Excel 'Data Analysis'. Set je opet spremljen kao .csv file i tako je postao spreman za daljnju obradu i analizu kroz alat Weka.

Predviđanje bankrota kompanije

Set podataka Predviđanje bankrota kompanije je set podataka koji sadrži 94 varijable koje prikazuju vrijednosti izračuna financijskih pokazatelja za pojedinu instancu odnosno poduzeće koje se želi analizirati. Od svih varijabli u set, samo su dvije nominalnog tipa. To su ciljna varijabla Bankrupt, te varijabla Liability Assets Flag.

Slika 14.: Raspodjela instanci po atributima – Predviđanje bankrota kompanije



Izvor: Weka

Na slici broj 14 vidimo dio atributa seta podataka. Slika prikazuje raspodijele atributa s obzirom na ciljanu varijablu iz alata Weka.

Set je preuzet u .csv formatu. Alat Weka nije mogao učitati set podataka u tom formatu pa se uz pomoć programskog jezika R file spremio kao format .arff. Nakon što je pomoću koda set spremljen, bio je prikladan za daljnju analizu programom Weka.

Predviđanje odustajanja od korištenja usluga turističke agencije

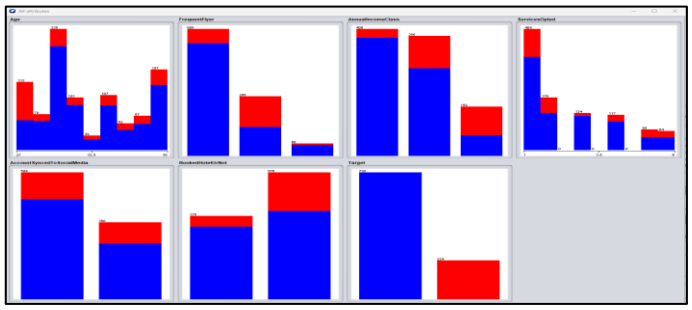
Tablica 4.: Atributi seta Predviđanje odustajanja od korištenja usluga turističke agencije

Atribut	**Objašnjenje atributa**	**Tip modaliteta**	**Modaliteti atributa**
Age	Dob klijenta	Numerički	MIN: 27, MAX: 38, MEAN: 32.11, STD.DEV: 3.34
FrequentFlyer	Redovan korisnik leta	Nominalni	Yes, No, No Record
AnnualIncomeClass	Klasa godišnjeg prihoda	Nominalni	Low Income, Middle Income, High Income
ServicesOpted	Broj puta koliko je klijent koristio usluge agencije	Numerički	MIN: 1, MAX: 6, MEAN: 2.44, STD.DEV: 1.61
AccountSyncedToSocialMedia	Je li račun tvrtke korisnika sinkroniziran s društvenim medijima	Nominalni	Yes, No
BookedHotelOrNot	Je li klijent koristio usluge agencije prilikom rezervacije smještaja	Nominalni	Yes, No
Target	Odustao ili nije odustao (Ciljna varijabla)	Nominalni	Yes, No

Izvor: Izrada autora

U tablici broj 4 vidimo atribute seta podataka Predviđanje odustajanja od korištenja usluga turističke agencije. Set se sastoji od 7 atributa koji definiraju instance, 5 je nominalnih, a 2 su numeričke. Nominalne varijable sadrže vrijednosti definirane kao opisne za instancu, dok numeričke imaju izračunata neka osnovna statistička obilježja. Tako primjerice za varijablu Age vidimo kako je najmlađi putnik osoba od 27 godina, najstariji putnik osoba od 38 godina. Prosječna starost putnika je 32,11 godina, a prosječno kvadratno odstupanje od aritmetičke sredine je 3,34 godine.

Slika 15.: Raspodjela instanci po atributima – Predviđanje odustajanja od korištenja usluga turističke agencije



Izvor: Weka

Na slici broj 15 vidimo kako je alat Weka raspodijelila instance unutar atributa s obzirom na ciljnu varijablu. Set sadrži 954 instance. Raspodjelu po atributu AnnualIncomeClass možemo objasniti tako da najveći broj instanci pripada srednjoj klasi primanja, a najmanji visokoj klasi primanja. Vidimo da u klasi srednjeg i niskog primanja većina ljudi ne otkazuje putovanja, dok su klasi visokih primanja ljudi u većini slučajeva otkazuju putovanja. Za objašnjenje numeričke varijable, odabrana je varijable ServicesOpted. Može se vidjeti kako je najviše instanci, njih 404 koristilo usluge agencije između 1 i 1,56 puta. Najmanje instanci je koristilo usluge više od 5,44 puta. U rasponima od 2,11 do 2,67, 3,22 do 3,78 i 4,33 do 4,89 nema nijedne instance. U svim rasponima većina putnika nije otkazala putovanje. Ciljni atribut ovog seta je atribut Target.

Set je preuzet u csv. Formatu te je opcijom alata Excel 'Text to Columns' pretvoren u tablični format. Za izračun statističkih obilježja numeričkih varijabli korištena je funkcija 'Data Analysis'. Ciljni atribut Target je Weka prepoznala kao numeričku varijablu pa se umjesto vrijednosti 0 postavila vrijednost 'No', a umjesto 1 'Yes'. Set je, nakon što je doveden u stanje za analizu, ponovno spremljen kao .csv format kako bi bio prikladan za upotrebu alatom Weka.

Predviđanje klijenata koji će odustati od korištenja kreditne kartice

Tablica 5.: Atributi seta Predviđanje klijenata koji će odustati od korištenja kreditne kartice

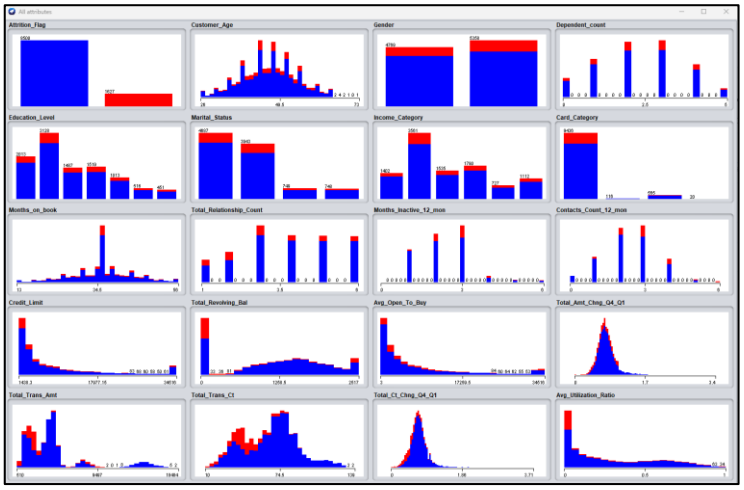
Atribut	**Objašnjenje atributa**	**Tip modaliteta**	**Modaliteti atributa**
Attrition_Flag	Odustanak od korištenja kreditne kartice (Ciljna varijabla)	Nominalni	Existing Customer, Attrited Customer
Customer_Age	Broj godina klijenta	Numerički	MIN: 26, MAX: 73, MEAN: 46.33, STD.DEV: 8.02
Gender	Spol klijenta	Nominalni	M, F
Dependent_count	Broj uzdržavanih članova	Numerički	MIN: 0, MAX: 5, MEAN: 2.35, STD.DEV: 1.30
Education_Level	Razina obrazovanja	Nominalni	High School, Graduate, Uneducated, Unknown, College, Post-Graduate, Doctorate
Marital_Status	Bračni status	Nominalni	Married, Single, Unknown, Divorced
Income_Category	Kategorija prihoda	Nominalni	$60K - $80K, Less than $40K, $80K - $120K, $40K - $60K, $120K +, Unknown
Card_Category	Kategorija kreditne kartice	Nominalni	Blue, Gold, Silver, Platinum
Months_on_book	Broj mjeseci od kada je osoba postala klijent	Numerički	MIN: 13, MAX: 56, MEAN: 35.93, STD.DEV: 7.97
Total_Relationship_Count	Broj proizvoda banke koje klijent koristi	Numerički	MIN: 1, MAX: 6, MEAN: 3.81, STD.DEV: 1.55
Months_Inactive_12_mon	Broj mjeseci nekorištenja kreditne kartice u posljednjih 12 mjeseci	Numerički	MIN: 0, MAX: 6, MEAN: 2.34, STD.DEV: 1.01
Contacts_Count_12_mon	Broj puta koliko je klijent kontaktiran u posljednjih 12 mjeseci	Numerički	MIN: 0, MAX: 6, MEAN: 2.46, STD.DEV: 1.11
Credit_Limit	Limit kreditne kartice	Numerički	MIN: 1438, MAX: 34516, MEAN: 8631.95, STD.DEV: 9088.78
Total_Revolving_Bal	Stanje revolving kredita	Numerički	MIN: 0, MAX: 2517, MEAN: 1162.81, STD.DEV: 814.99
Avg_Open_To_Buy	Prosječan preostali iznos potrošnje u posljednjih 12 mjeseci	Numerički	MIN: 3, MAX: 34516, MEAN: 7469.14, STD.DEV: 9090.69
Total_Amt_Chng_Q4_Q1	Postotna promjena u ukupnom potrošenom iznosu između prvog i četvrtog kvartala	Numerički	MIN: 0, MAX: 3.40, MEAN: 0.76, STD.DEV: 0.22
Total_Trans_Amt	Ukupan iznos potrošnje u posljednjih 12 mjeseci	Numerički	MIN: 510, MAX: 18484, MEAN: 4404.09, STD.DEV: 3397.13
Total_Trans_Ct	Ukupan broj transakcija u posljednjih 12 mjeseci	Numerički	MIN: 10, MAX: 139, MEAN: 64.86, STD.DEV: 23.47
Total_Ct_Chng_Q4_Q1	Postotna promjena u ukupnom broju transakcija između prvog i četvrtog kvartala	Numerički	MIN: 0, MAX: 3.71, MEAN: 0.71, STD.DEV: 0.24

Avg_Utilization_Ratio	Postotak iskorištenosti limita	Numerički	MIN: 0, MAX: 1, MEAN: 0.275, STD.DEV: 0.276

Izvor: Izrada autora

U tablici broj 5 su prikazani atributa seta Predviđanje klijenata koji će odustati od korištenja kreditne kartice. Set sadrži 21 varijablu, od kojih je 6 nominalnog tipa, a 15 numeričkog tipa. Za nominalne varijable su kao modaliteti ispisane vrijednosti koje instance mogu poprimiti, a za numeričke su ispisana statistička obilježja. Tako primjerice za varijablu Months_on_book vidimo kako je minimalni broj mjeseci od kad je neka osoba postala klijent banke 13 mjeseci, a maksimalni 56 mjeseci. Prosječan broj mjeseci je 35,93, a kvadratno odstupanje od prosjeka 7,97.

Slika 16.: Raspodjela instanci po atributima - Predviđanje klijenata koji će odustati od korištenja kreditne kartice



Izvor: Weka

Na slici broj 16 vidimo kako je softwarea Weka raspodijelila instance unutar atributa s obzirom na ciljnu varijablu. Set sadrži 10128 instanci. Za objašnjenje značenja modaliteta numeričke varijable, odabrana je varijabla Credit_Limit. Minimalni limit za mjesečnu potrošnju na kartici je 1438, a maksimalni 34516. Prosječan limit za potrošnju je 8631.95, a kvadratno odstupanje od prosjeka iznosi 9088.78. Vidljivo je kako najveći broj klijenata ima određen limit u iznosu od 1438,3 do 2876,46, a najmanji broj klijenata u rasponu od 30201,52 do 31639,68.

Ciljna varijabla ovog seta je Attrition_Flag.

Set je preuzet u .csv formatu te je kao takav bio spreman za korištenje u Weki i za daljnju analizu.

5.3. Usporedba setova

Tablica 6.: Usporedba setova podataka

Set	Broj Instanci	Broj varijabli	Broj Numeričkih varijabli	Broj Nominalnih varijabli	Ciljna varijabla
Marketinška kampanja portugalske banke	45211	17	6	11	Subscription
Netflix pretplata	2135	4	2	2	Downloaded
Predviđanje bankrota kompanije	6819	94	92	2	Bankrupt
Predviđanje odustajanja od korištenja usluga turističke agencije	954	7	2	5	Target
Predviđanje klijenata koji će odustati od korištenja kreditne kartice	10127	21	15	6	Attrition_Flag

Izvor: Izrada autora

U tablici broj 6 su uspoređene karakteristike setova podataka korištenih u izradi rada. Gledajući broj instanci, najveći set je Marketinška kampanja portugalske banke s 45211 instanci. Redom slijede set Predviđanje klijenata koji će odustati od korištenja kreditne kartice s 10127 instanci, Predviđanje bankrota kompanije sa 6819 instanci, Netflix pretplata s 2135 instanci i Predviđanje odustajanja od korištenja usluga turističke agencije s 954

instance. Najviše varijabli korištenih za izradu modela se nalazi u setu Predviđanje bankrota kompanije s 94 varijable, a set s najmanje varijabli je Netflix pretplata s 4 varijable. Najveći postotaka varijabli numeričkog tipa ima set Predviđanje bankrota kompanije s 97,87%, dok je set s najmanjim broj numeričkih varijabli, odnosno najvećim brojem nominalnih varijabli, set Predviđanje odustajanja od korištenja usluga turističke agencije s 28,57% numeričkih varijabli.

6. Analiza rezultata i usporedba genetskog algoritma s ostalim algoritmima šume

6.1. Prikaz koraka za analizu Weka softwareom

Weka je software razvijen na Sveučilištu Waikato u Novom Zelandu; naziv Weka je kratica za Waikato Environment for Knowledge Analysis. Program je napisan u Javi i distribuiran pod uvjetima GNU-ove licence opće javnosti. Radi na gotovo svim platformama i testiran je na operativnim sustavima Linux, Windows i Macintosh. Weka pruža jedinstveno sučelje za izvedbu mnogih različitih algoritama, zajedno s metodama za obradu i za vrednovan je rezultata shema strojnog učenja na bilo kojem skupu podataka (Kalmegh, 2015.). Drugim riječima, Weka je kolekcija otvorenog koda mnogih metoda rudarenja podataka i algoritama strojnog učenja, koja uključuje prethodnu obradu podataka, klasifikaciju, grupiranje, ekstrakciju asocijativnih pravila i tako dalje (Desai i Rai, 2013.).

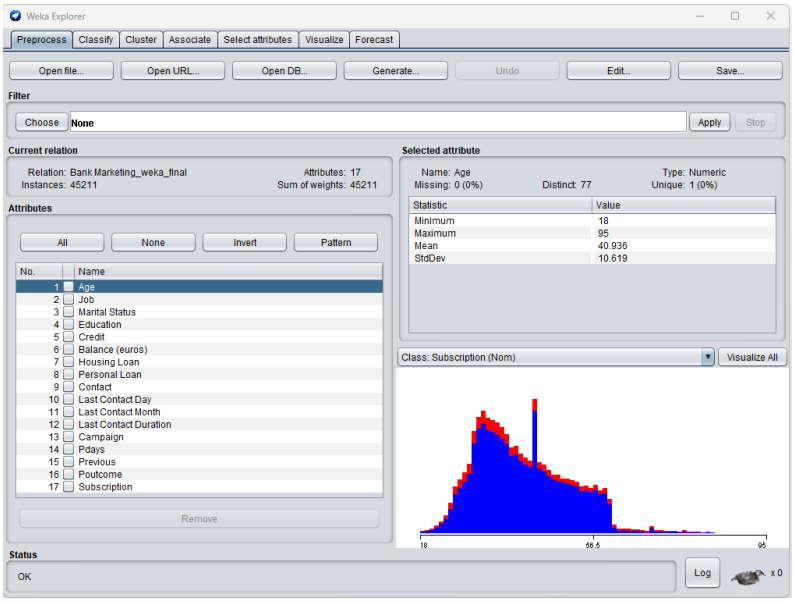
Podatke, koje želimo analizirati algoritmom, moramo učitati u Weku pomoću izbornika Preprocess do kojeg dolazimo kroz sučelje Explorer na početnom prozoru koji se otvara nakon što pokrenemo software. Na slici broj 17 je prikazana početni prozor nakon pokretanja softwarea.

Slika 17.: Početni prozor nakon pokretanja softwarea Weka

Izvor: Weka

Na slici broj 18 se nalazi prikaz već spomenutog izbornika Preprocess nakon što je učitan jedan od setova podataka koji će biti analizirani. Kroz izbornik učitavamo podatke opcijom 'Open File'.

Slika 18.: Prikaz izbornika Preprocess softwarea Weka



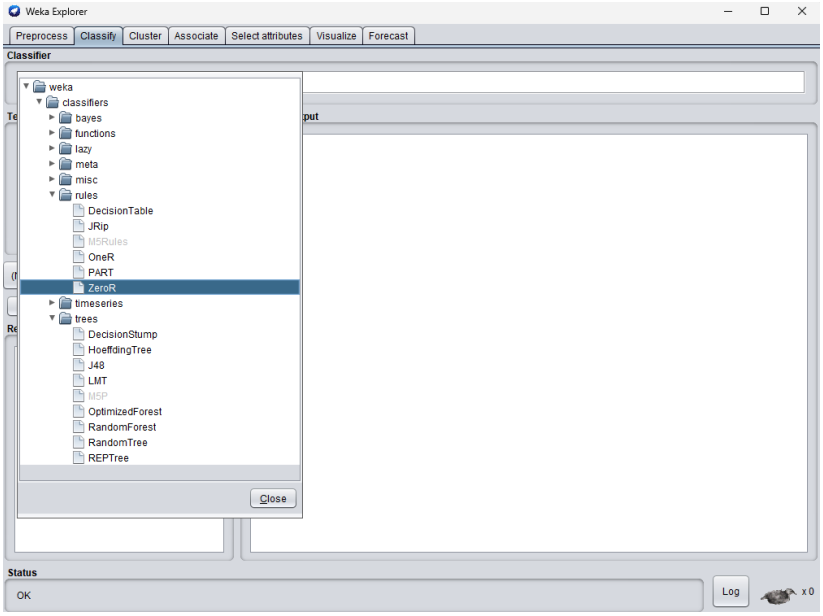
Izvor: Weka

Izbornik Preprocess služi za pregled podataka koje analiziramo. Za set tako možemo vidjeti njegove karakteristike – broj instanci unutar seta, atribute i vrijednosti koje mogu poprimiti (minimalnu vrijednost, maksimalnu vrijednost, aritmetičku sredinu i kvadratno odstupanje od aritmetičke sredine). Za potrebe analize podataka, u ovom izborniku se mogu ukloniti određene varijable ako osoba koja provodi proces rudarenja podataka smatra da će neki od atributa nepovoljno utjecati na rezultate ili ako ih se nema smisla analizirati jer ne doprinose reprezentativnosti rezultata.

Bitno je naglasiti kako Explorer softwarea Weka sadrži i druge izbornike, a to su izbornici koji služe za različite načine obrade podataka – klasifikaciju, klaster analizu, asocijativna pravila i tako dalje. Na slici broj 18 pri vrhu su vidljivi navedeni izbornici. Svi setovi podataka u ovom radu su analizirani metodama klasifikacije u izborniku Classify.

Ulaskom u izbornik odabiremo algoritam za analizu kroz prozor Classifier, pritiskom na dugme Choose. Na slici broj 19 su prikazani neki od algoritama klasifikacije koje Weka nudi.

Slika 19.: Popis dijela algoritama za klasifikaciju dostupnih u softwareu Weka



Izvor: Weka

U istom prozoru možemo konfigurirati sve parametre za analizu podataka. Na slici broj 20 vidimo kako Weka prikazuje rezultate rudarenja podataka određenog seta.

Slika 20.: Prikaz rezultata rudarenja podataka softwareom Weka

```
Size of the tree : 1602

Time taken to build model: 0.95 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances       40588               89.7746 %
Incorrectly Classified Instances      4623               10.2254 %
Kappa statistic                          0.4349
Mean absolute error                      0.133
Root mean squared error                  0.2732
Relative absolute error                 64.3674 %
Root relative squared error             84.9932 %
Total Number of Instances            45211

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0,961    0,580    0,926      0,961    0,943      0,443    0,868     0,967     Yes
                 0,420    0,039    0,588      0,420    0,490      0,443    0,868     0,499     No
Weighted Avg.    0,898    0,517    0,886      0,898    0,890      0,443    0,868     0,912

=== Confusion Matrix ===

     a     b   <-- classified as
 38367  1555 |     a = Yes
  3068  2221 |     b = No
```

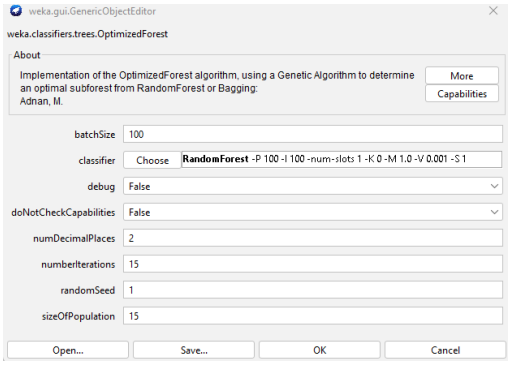
Izvor: Weka

6.2. Algoritmi korišteni u analizi i postavke algoritama

U ovom poglavlju rada će biti objašnjeni i opisani algoritmi kojima će se setovi podataka analizirati. Korišteni algoritmi su sljedeći:

- OptimizedForest - Optimized Forest je algoritam baziran na šumama odlučivanja koji koristi genetski algoritam za odabir onih šuma koje su najpreciznije i najraznolikije s ciljem da generalno poveća točnost algoritma. Glavni cilj algoritma je da stvori najkvalitetnija stabla koja će biti korištena kao inicijalna populacija genetskog algoritma. Genetski algoritam kreira populaciju u formi kromosoma. Optimized Forest koristi 20 kromosoma kako bi stvorio populaciju. 10 kromosoma je generirano koristeći metodu Stratified sampling koja služi za grupiranje podataka u homogene grupe ovisno o njihovim karakteristikama. Te grupe se nazivaju strata. Ostalih 10 kromosoma se ge nerira slučajno. Križanje i mutacija se u kromosomima obavlja metodom roulette wheel (Kumar i Pati, 2022.). Postupak elitizma se odvija nakon križanja i mutacije. Genetski algoritam kroz iteracije gradi i testira šume koje generira klasifikator RandomForest. Nakon 10 iteracija, odnosno 10 generiranih populacija, algoritam staje i rezultate prikazuje kao konačne. Za sve setove podataka su korištene iste postavke algoritma, koristeći Cross-validation = 10 kako bi se set raspodijelo na 10 dijelova. Na slici broj 21 su prikazane postavke algoritma OptimizedForest. Najvažniji parametri su numberIterations koja označava broj iteracija za stvaranje nove generacije te sizeOfPopulation koji označava početnu populacija algoritma odnosno broj šuma namijenjenih za izračun funkcije cilja i stvaranje nove generacije. Korišteno je 15 iteracija algoritma te početna populacija od 15 šuma.

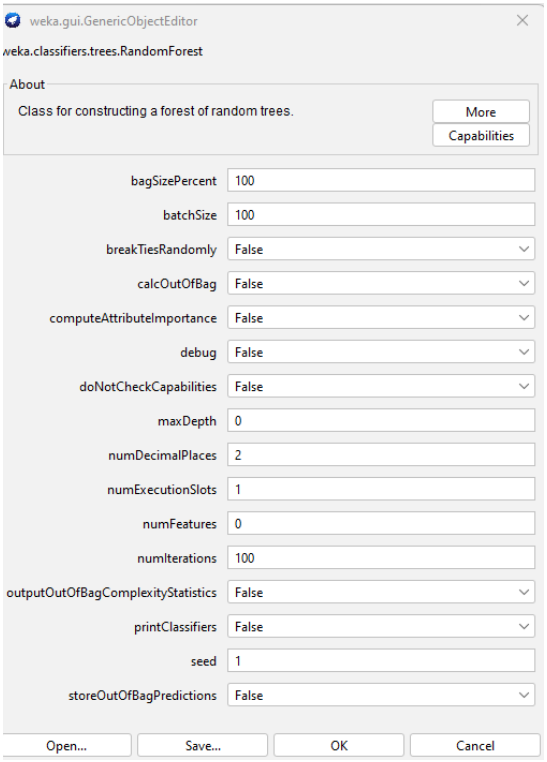
Slika 21.: Postavke algoritma OptimizedForest



Izvor: Weka

- RandomForest - Random forest algoritam je jedan od fleksibilnijih i moćnijih algoritama za klasifikaciju koji se temelju na šumama odlučivanja. Djeluje tako da kreira mnoštvo stabala bazirano na slučajnim podacima dobivenih bootstrap metodama (metode ponovnog uzorkovanja podataka korištene za procjenu pouzdanosti modela i procedura) (Naghibi i suradnici, 2017.). Osim što se služi slučajnim uzorkovanjem podataka, prilikom razdvajanja stabla, razmatra samo podskup podataka umjesto svih dostupnih. To pomaže u raznolikosti stabla (Kantardžić, 2003). Random forest je veoma brz i robustan u borbi sa over-fittingom podataka te je moguće kreirati onoliko stabla koliko osoba želi (Akar i Gungor, 2012.). Postavke algoritma za sve setove podataka su iste i korišten je cross-validation = 10. Na slici broj 22 su prikazane postavke algoritma RandomForest. Broj stabala koje će algoritam kreirati je namješten na 2.

Slika 22.: Postavke algoritma RandomForest

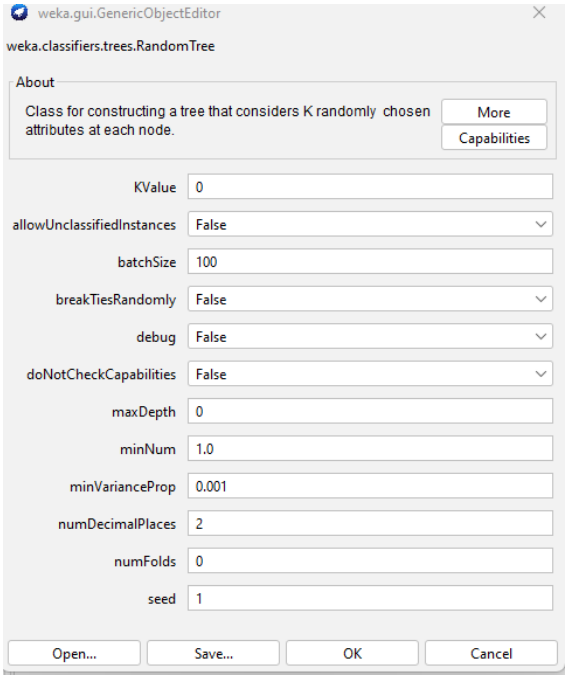


Izvor: Weka

- RandomTree - Random Tree je algoritam koji nasumično kreira nekoliko stabala odlučivanja. Koristiti bagging metode kako bi iznjedrio slučajan set podataka koji će služiti za kreiranje stabla. Može služiti za klasifikaciju nad podacima, ali i za regresiju. Prvi puta su ovaj algoritam predstavili Leo Breiman

i Adele Cutler (Kalmegh, 2015.). Algoritam kreira više slučajnih stabala i tako postiže veću točnost modela. Za razliku od Random Forest algoritma, Random Tree u konačnici rezultira samo jednim stablom. Za sve setove podataka su korištene jednake postavke algoritma s metodom Cross-validation = 10. Tako je set raspoređen u 10 manjih setova na kojima se model gradio. Na slici broj 23 su prikazane postavke algoritma RandomTree.

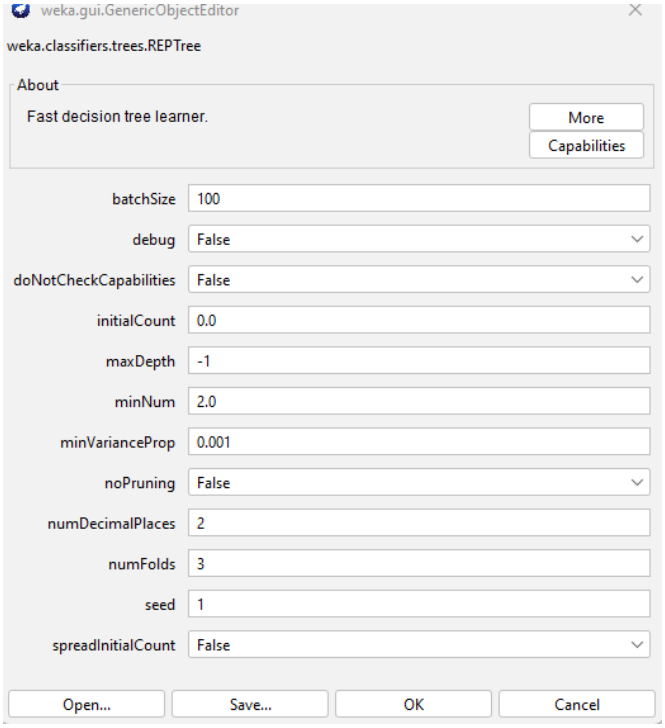
Slika 23.: Postavke algoritma RandomTree



Izvor: Weka

- REPTree - REPTree je algoritam koji koristi regresijsku logiku kako bi kreirao više stabala u više iteracija. Nakon toga, odabire najbolje stablo između svih kreiranih. To stablo se smatra najreprezentativnijim za izradu modela. Kod odabira najboljeg stabla, glavna značajka točnosti je mean square error (Kalmegh, 2015.). REPTree za cilj ima smanjenje greške varijance u podacima i rezultira smanjenom kompleksnošću stabla korištenog u modelu (Munandar i Winarko, 2015.). U numeričkim varijablama koristi sortiranje samo jednom, a za sve instance kojima vrijednosti nedostaju, koristi metodu C4.5's. Za sve setove podataka su korištene iste postavke algoritma s Cross-validation = 10. Na slici broj 24 su prikazane postavke algoritma REPTree.

Slika 24.: Postavke algoritma REPTree



Izvor: Weka

6.3. Mjere točnosti algoritama

Za analizu setova podataka koristit će se sljedeće vrijednosti te će one biti objašnjene u nastavku: Correctly Classified Instances, Incorrectly Classified Instances, TP Rate, FP Rate, Precision, Recall i F-Measure. Točnost algoritama može se mjeriti računanjem točno klasificiranih instanci (TP – True Positive i TN – True Negative) i instanci koje je algoritam netočno klasificirao (FP – False Positive i FN – False Negative). Te vrijednosti zajedno tvore konfuzijsku matricu koja će za sve setove biti prikazana u poglavlju 6.4. rada (Pejić Bach i suradnici, 2019.).

- Correctly Classified Instances – broj instanci koje je algoritam točno klasificirao
- Incorrectly Classified Instances – broj instanci koje je algoritam točno klasificirao
- TP Rate – broj koji označava u kojoj mjeri algoritam osigurava da će pozitivni slučajevi u algoritmu biti točno klasificirani. Računa se kao količnik točno klasificiranih pozitivnih instanci (TP) i ukupnog broja pozitivnih instanci (zbroj TP i FN). Što je TP veći, algoritam se smatra točnijim.

$$TP\ RATE = \frac{TP}{TP + FN} \tag{5}$$

- FP Rate – broj koji se računa kao količnik netočno klasificiranih pozitivnih instanci (FP) i ukupnog broja negativnih instanci (zbroj FP + TN).

$$FP\ RATE = \frac{FP}{FP + TN} \tag{6}$$

- Precision – Broj koji se računa kao količnik TP-a i broj instanci koje su u setu klasificirane kao pozitivne (zbroj TP i FP).

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

- Recall – Broj koji se računa kao količnik točno klasificiranih pozitivnih instanci i ukupnog broja pozitivnih instanci. Jednak je vrijednosti TP Rate (Vujović, 2021.).
- F-Measure – Broj koji se računa pomoću mjere Precision i Recall i predstavlja aproksimativni prosjek tih mjera (Pejić Bach i suradnici, 2019.).

$$F = 2\frac{Precision * Recall}{Precision + Recall} \tag{8}$$

6.4. Analiza rezultata

U tablici broj 7 na sljedećoj stranici rada su predstavljeni rezultati izabranih algoritama nad svim setovima podataka. Setovi podataka su označeni brojem kako bi tablica bila preglednija (vidi poglavlje 5.1. rada). Kolone u tablici označavaju mjere točnosti koje su korištene za evaluaciju i usporedbu algoritama.

Tablica 7.: Postotak točno klasificiranih instanci u modelu

Algoritam	Set	Correctly Classified Instances	Incorrectly Classified Instances
RandomForest	1	**90.34%**	9.66%
OptimizedForest	1	90.32%	9.68%
RepTree	1	89.77%	10.23%
RandomTree	1	87.18%	12.82%
RepTree	2	**88.67%**	11.33%
OptimizedForest	2	85.06%	14.94%
RandomForest	2	84.87%	15.13%
RandomTree	2	83.51%	16.49%
OptimizedForest	3	**97.11%**	2.89%
RandomForest	3	97.08%	2.92%
RepTree	3	96.70%	3.30%
RandomTree	3	95.31%	4.69%
OptimizedForest	4	**87.84%**	12.16%
RandomTree	4	87.53%	12.47%
RandomForest	4	87.00%	13.00%
RepTree	4	86.48%	13.52%
OptimizedForest	5	**96.16%**	3.84%
RandomForest	5	96.12%	3.88%
RepTree	5	94.54%	5.46%
RandomTree	5	92.52%	7.48%

Izvor: Izrada autora

U tablici broj 7 su vidljivi rezultati algoritama po točnosti klasificiranih instanci u setovima podataka. Za prvi set podataka Marketinška kampanja portugalske banke najtočniji je bio algoritam RandomForest s 90,34% točno klasificiranih instanci. Zatim slijede OptimizedForest s 90,32% točnosti, REPTree s 89,77% točnosti, a algoritam s najmanje točno klasificiranih instanci za prvi set je RandomTree s 87,18%.

Algoritam s najviše točno klasificiranih instanci za drugi set podataka Netflix pretplata je REPTree s 88,67%, zatim OptimizedForest s 85,06%, RandomForest s 84,87%, a najlošije algoritam je RandomTree s 83,51%.

U trećem setu podataka Predviđanje bankrota kompanije, najtočniji je algoritam OptimizedForest s 97,11%, zatim RandomForest s 97,08%, REPTree s 96,70%, a najmanje točno klasificiranih instanci u trećem setu ima algoritam RandomTree s 95,31%.

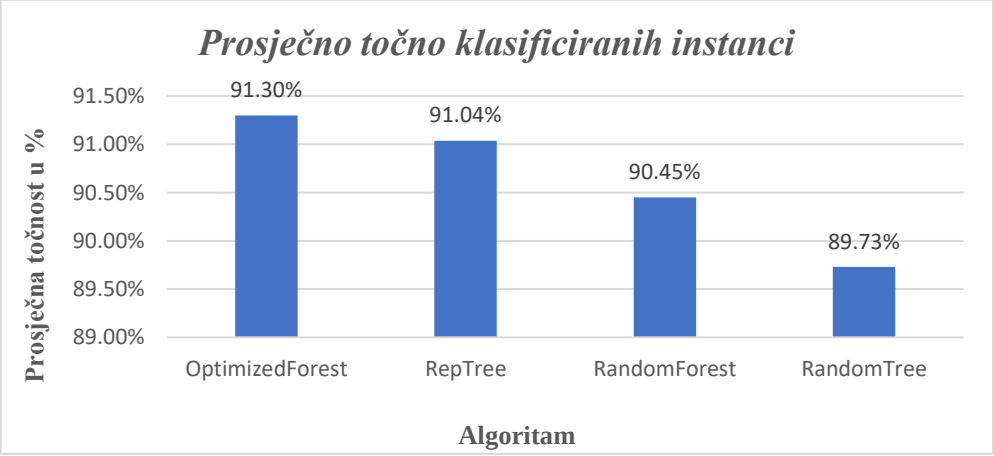
U četvrtom setu podataka Predviđanje odustajanja od korištenja usluga turističke agencije, najtočniji algoritam je OptimizedForest s 87,84% točno klasificiranih instanci, na drugom mjestu je algoritam RandomTree s 87,53% točnosti, zatim RandomForest s 87,00% točnosti, a algoritam s najmanje točno klasificiranih instanci je algoritam REPTree s 86,48% točno klasificiranih instanci.

Za peti set podataka Predviđanje klijenata koji će odustati od korištenja kreditne kartice najtočniji algoritam je bio OptimizedForest s 96,16% točno klasificiranih instanci, na drugom mjestu po broju točno klasificiranih instanci se nalazi algoritam RandomForest s 96,12%, na trećem mjestu algoritam REPTree s 94,54% točno klasificiranih instanci, a algoritam s najmanjim brojem točno klasificiranih instanci je algoritam RandomTree s 92,52% točno klasificiranih instanci.

OptimizedForest, temeljen na genetskom algoritmu je bio najtočniji u tri od pet setova podataka, dok u dva u kojima nije bio najbolji je bio relativno blizu vrhu.

Na grafikonu broj 1 je prikazana prosječna točnost algoritma gledano kroz točno klasificiranie instance kroz izvedbu algoritma na svih pet setova podataka. OptimizedForest je prosječno najtočniji algoritam s 91,30% točnosti. Slijedi REPTree s 91,04% točnosti, zatim RandomForest s 90,45% točnosti, a na posljednjem mjestu nalazi se algoritam RandomTree s 89,73% točnosti.

Grafikon 1.: Prosječno točno klasificiranih instanci u svim setovima



Izvor: Izrada autora uz pomoć alata Excel

Tablica 8.: Mjere točnosti algoritama

Algoritam	Set	TP Rate	FP Rate	Precision	Recall	F-Measure
OptimizedForest	1	**0.903**	0.568	0.889	**0.903**	**0.890**
RandomForest	1	**0.903**	0.554	**0.890**	**0.903**	0.892
RandomTree	1	0.872	0.520	0.868	0.872	0.870
RepTree	1	0.898	**0.517**	0.886	0.898	**0.890**
OptimizedForest	2	0.851	0.154	0.856	0.851	0.852
RandomForest	2	0.849	0.159	0.853	0.849	0.820
RandomTree	2	0.835	0.196	0.835	0.835	0.835
RepTree	2	**0.887**	**0.066**	**0.912**	**0.887**	**0.889**
OptimizedForest	3	**0.971**	0.783	**0.964**	**0.971**	**0.963**
RandomForest	3	**0.971**	0.783	**0.964**	**0.971**	**0.963**
RandomTree	3	0.953	**0.700**	0.953	0.953	0.953
RepTree	3	0.967	0.867	0.954	0.967	0.957
OptimizedForest	4	**0.878**	0.223	**0.878**	**0.878**	**0.878**
RandomForest	4	0.870	0.260	0.867	0.870	0.868
RandomTree	4	0.875	**0.218**	0.876	0.875	0.876
RepTree	4	0.865	0.264	0.862	0.865	0.863
OptimizedForest	5	**0.962**	**0.148**	**0.961**	**0.962**	**0.961**
RandomForest	5	0.961	0.149	0.960	0.961	0.960
RandomTree	5	0.925	0.205	0.925	0.925	0.925
RepTree	5	0.945	0.178	0.944	0.945	0.945

Izvor: Izrada Autora

U tablici broj 8 su prikazani rezultati algoritmima promatrajući ostale mjere. Mjera točnosti TP Rate za set Marketinška kampanja portugalske banke pokazuje kako su OptimizedForest i RandomForest jednaki. Isto vrijedi i za set Predviđanje bankrota kompanije. Za set Netflix pretplata, najtočniji je algoritam REPTree, dok je OptimizedForest najbolji za setove Predviđanje odustajanja od korištenja usluga turističke agencije i Predviđanje klijenata koji će odustati od korištenja kreditne kartice.

FP Rate pokazuje kako je REPTree bio najbolji u dva seta, RandomTree za dva seta, a OptimizedForest u jednom setu.

Najtočniji set sudeći mjerom Precision je OptimizedForest u 3 slučaja, dok u jednom, za set Predviđanje bankrota kompanije dijeli najbolji rezultat s algoritmom RandomForest. RandomForest je najtočniji tako u dva set jer je ostavio najbolji rezultat za set Marketinška kampanja portugalske banke, dok je za set Netflix Pretplata najtočniji algoritam REPTree.

Po mjeri Recall, algoritam OptimizedForest je najtočniji za 4 seta, odnosno za set Predviđanje bankrota kompanije dijeli najtočniji rezultat s algoritmom RandomForest. Za set Netflix pretplata je po mjeri Precision najtočniji algoritam RepTree.
Promatrajući mjeri F-Measure, OptimizedForest ostvaruje najbolji rezultat u 4 slučaja. U 2 slučaja, za set Marketinška kampanja portugalske banke i set Predviđanje bankrota kompanije dijeli najbolji rezultat s algoritmom REPTree i algoritmom RandomForest. Za set Netflix pretplata, najtočniji algoritam po mjeri F-Measure je REPTree.

Jedan od također bitnijih pokazatelja je klasifikacijska matrica koja pokazuje klasificirane instance s obzirom na ciljni atribut. Elementi u matrici na poziciji AA (True Positive) predstavljaju broj instanci koje su u modelu točno klasificirane kao pozitivne. Elementi u matrici na poziciji BB (True Negative) također prikazuju točno klasificirane instance, ali kao negativne. Elementi na poziciji AB (False Negative) prikazuje instance koje su u modelu klasificirane kao negativne, a u stvarnosti su pozitivne. Elementi na poziciji BA (False Positive) prikazuje instance koje su u modelu klasificirane kao pozitivne, a u stvarnosti su negativne.
Za primjer je objašnjena klasifikacijska matrica u tablici broj 7, generirana od strane algoritma OptimizedForest za set podataka marketinške kampanje portugalske banke. 86,09% instanci je u setu klasificirano kao da se pretplatilo na proizvod banke i stvarno se pretplatilo. 2,19% instanci je klasificirano kao da su odlučili da se ne žele pretplatiti na prozivod banke, a pretplatili su se u stvarnosti. 7,49% instanci je klasificirano kao da se pretplatilo na proizvod, a u stvarnosti nije. 4,21% instanci je točno klasificirano kao da se nije pretplatilo na proizvod banke. Tablica broj 7 prikazuje i ostale klasifikacijske matrice za setove podataka generirane od strane algoritma OptimizedForest. OptimizedForest je za sve setove najbolje klasificirao instance, odnosno najveći postotak instanci u klasifikacijskom matrici se odnosi na True Postive i True Negative instance. Klasifikacijske matrice za sve algoritme i sve setove podataka se nalaze u tablicama na sljedećoj stranici rada.

Tablica 9.: Klasifikacijska matrica - OptimizedForest

Set	Broj instanci	% Klasificirano kao A	% Klasificirano kao B	Ciljni atribut
1	45221	86.09%	2.19%	A = YES
		7.49%	4.21%	B = NO
2	2135	55.18%	9.27%	A = YES
		5.67%	29.88%	B = NO
3	6819	0.62%	2.61%	A = Y
		0.28%	96.50%	B = N
4	954	17.19%	6.29%	A = YES
		5.87%	70.65%	B = NO
5	10127	82.90%	1.04%	A = EXISTING CUSTOMER
		2.80%	13.26%	B = ATTRITED CUSTOMER

Izvor: Rad autora

U tablici broj 9 je prikazana klasifikacijska matrica algoritma OptimizedForest.

Tablica 10.: Klasifikacijska matrica - RandomForest

Set	Broj instanci	% Klasificirano kao A	% Klasificirano kao B	Ciljni atribut
1	45221	85.92%	2.37%	A = YES
		7.29%	4.40%	B = NO
2	2135	55.32%	9.13%	A = YES
		6.00%	29.56%	B = NO
3	6819	0.62%	2.61%	A = Y
		0.31%	96.60%	B = N
4	954	16.04%	7.44%	A = YES
		5.56%	70.96%	B = NO
5	10127	82.86%	1.08%	A = EXISTING CUSTOMER
		2.80%	13.26%	B = ATTRITED CUSTOMER

Izvor: Rad autora

U tablici broj 10 je prikazana klasifikacijska matrica generirana algoritmom RandomForest.

Tablica 11.: Klasifikacijska matrica - Random Tree

Set	Broj instanci	% Klasificirano kao A	% Klasificirano kao B	Ciljni atribut
1	45221	82.24%	6.04%	A = YES
		6.78%	4.92%	B = NO
2	2135	56.30%	8.15%	A = YES
		8.34%	27.21%	B = NO
3	6819	0.89%	2.33%	A = Y
		2.36%	94.41%	B = N
4	954	17.40%	6.08%	A = YES
		6.39%	70.13%	B = NO
5	10127	80.24%	3.69%	A = EXISTING CUSTOMER
		3.78%	12.28%	B = ATTRITED CUSTOMER

Izvor: Rad autora

U tablici broj 11 je prikazana klasifikacijska matrica generirana algoritmom RandomTree.

Tablica 12.: Klasifikacijska matrica - REPTree

Set	Broj instanci	% Klasificirano kao A	% Klasificirano kao B	Ciljni atribut
1	45221	84.84%	3.44%	A = YES
		6.78%	4.91%	B = NO
2	2135	53.40%	11.05%	A = YES
		0.28%	35.27%	B = NO
3	6819	0.34%	2.89%	A = Y
		0.41%	96.36%	B = N
4	954	15.93%	7.55%	A = YES
		5.97%	70.55%	B = NO
5	10127	81.80%	2.13%	A = EXISTING CUSTOMER
		3.33%	12.74%	B = ATTRITED CUSTOMER

Izvor: Rad autora

U tablici broj 12 je prikazana klasifikacijska matrica generirana algoritmom REPTree.

U tablici broj 13 su prikazana vremena izvedbe algoritama na određenim setom podataka. OptimizedForest, zbog kreiranja višestrukih šuma koristeći classifier RandomForest značajno više vremena provodi analizirajući setove podataka. Nakon njega, najviše vremena je potrebno algoritmu RandomForest, zatim algoritmu RandomTree,a a najbrži algoritam je REPTree.

Tablica 13.: Vrijeme izvedbe algoritama u sekundama

Algoritam	Set	Vrijeme u sekundama
OptimizedForest	1	1367
RandomForest	1	97
RandomTree	1	3
RepTree	1	8
OptimizedForest	2	112
RandomForest	2	2
RandomTree	2	1
RepTree	2	1
OptimizedForest	3	157
RandomForest	3	21
RandomTree	3	1
RepTree	3	2
OptimizedForest	4	43
RandomForest	4	1
RandomTree	4	1
RepTree	4	1
OptimizedForest	5	571
RandomForest	5	20
RandomTree	5	2
RepTree	5	1

Izvor: Izrada autora

7. Zaključak

Zaključak diplomskog rada ukazuje značajnu vrijednost genetskog algoritma u klasifikaciji podataka u poslovnim primjenama. Kroz istraživanje i interpretaciju rezultata na više setova podataka, dokazano je kako je algoritam temeljen na genetskom algoritmu iznimno uspješan i učinkovit u rješavanju problema koji zahtijevaju obradu klasifikacijom podataka. Genetski algoritam postiže visoke rezultate u usporedbi s ostalim algoritmima šuma te je on uistinu dominantni algoritam. Ovaj rad također istražuje izazove i ograničenja genetskog algoritma, kao što su potreba za pravilnim podešavanjem parametara, ovisnost o početnim rješenjima i vremenski zahtjevi za izvođenje. Razumijevanje tih aspekata u softwareima u kojim se genetski algoritam može iskoristiti pomaže u boljem upravljanju i primjeni genetskog algoritma u stvarnim poslovnim scenarijima. Primijećeno je kako genetski algoritam zahtjeva najviše vremena za obradu, no taj nedostatak kompenzira svojom točnošću nad, posebice u setovima s velikim brojem podataka i varijabli koji mogu utjecati na ishod klasifikacije. Nedostatak algoritma je taj što zbog generiranja velikog broja stabala u svakoj iteraciji algoritma, potrebno je softwareu za obradu dozvoliti pristup velikom djeluju resursa računala kojim se provodi analiza.

Genetski algoritam se može smatrati vrijednim pristupom otkrivanju znanja u podacima te može pomoći u donošenju kvalitetnih poslovnih odluka koji će rezultirati povećanjem konkurentske prednosti poduzeća koje se odluči na njegovu primjenu i implementaciju. Njegove performanse čine ga perspektivnim izborom za širok spektar poslovnih problema. Smatra se kao i vrijedan alat za rješenja poslovnih pitanja.

Ovaj rad smatra se kao doprinos istraživanjima genetskog algoritma te njegovu razumijevanju unutar metoda klasifikacije podataka i otkrivanja znanja u podacima.

Literatura

1. Adnan N. i Islam Z. (2016). *Optimizing the number of trees in a decision forest to discover a subforest with high ensemble accuracy using a genetic algorithm*. Knowledge-Based Systems, 110, str. 86-97.
2. Akar, Ö. i Gungor, O. (2012). *Classification of Multispectral Images Using Random Forest Algorithm*. Journal of Geodesy and Geoinformation, 1, str. 105-112.
3. Bäck, T. i Schwefel, H.P. (1993). *An Overview of Evolutionary Algorithms for Parameter Optimization*. Evolutionary Computation, 1 , str. 1-23.
4. Bottaci, L. (2001). *A Genetic Algorithm Fitness Function for Mutation Testing. Software engineering using metaheuristic inovative algorithms.*
5. Desai, A. i Rai, S. (2013). *Analysis of Machine Learning Algorithms using Weka*.
6. Fortin, F.A., De Rainville, F.M., Gardner, M.A., Parizeau, M. i Gagné, C. (2012). *DEAP: Evolutionary algorithms made easy*. Journal of Machine Learning Research, Machine Learning Open Source Software, 13, str. 2171-2175.
7. Gamberger, D. (2011) *Otkrivanje znanja dubinskom analizom podataka*. Zagreb. Institut R. Bošković.
8. Golub, M. (1997). *Genetski Algoritam: Prvi dio*. Nastavni materijal. Zavod za elektroniku, mikroelektroniku, računalne i inteligentne sustave, Fakultet elektrotehnike i računarstva.
9. Golub, M. (2004). *Genetski Algoritam: Drugi dio*. Nastavni materijal. Zavod za elektroniku, mikroelektroniku, računalne i inteligentne sustave, Fakultet elektrotehnike i računarstva.
10. Grundler, D., i Rolich, T. (2001). *Evolucijski algoritmi (II) Primjena*, Automatika, 42 (3-4), str. 133-142.
11. Hermawan, D. R., Fatihah, M. F.G., Kurniawati, L. i Helen, A. (2021). *Comparative Study of J48 Decision Tree Classification Algorithm, Random Tree, and Random Forest on In-Vehicle Coupon Recommendation Data*. 2021 International Conference on Artificial Intelligence and Big Data Analytics, str. 1-6.
12. Holmes, G.,Donkin, A. i Witten I.H. (1994) *WEKA: a machine learning workbench*. Proceedings of ANZIIS '94 - Australian New Zealnd Intelligent Information Systems Conference, str. 357-361.

13. Hossin M. i Sulaiman M.N. (2015) *A Review on Evaluation Metrics for Data Classification Evaluations*. International Journal of Data Mining & Knowledge Management Process, 5 (2), str. 01-11.
14. Kalmegh, S.R. (2015). *Analysis of WEKA Data Mining Algorithm REPTree, Simple Cart and RandomTree for Classification of Indian News*.
15. Katoch, S., Chauhan, S.S. i Kumar, V. (2021). *A review on genetic algorithm: past, present, and future*. Multimed Tools Appl, 80, str. 8091–8126
16. Kraljević, S., i Staničić, O. (2020). *Primjena algoritama dubinske analize podataka i strojnog učenja za klasifikaciju i predikciju u društvenom području*. Polytechnic and design, 8(1), str. 38-46.
17. Krotov, V. (2017). *A Quick Introduction to R and RStudio*. 10.13140/RG.2.2.10401.92009.
18. Kumar S. i Pati J. (2022) *Assessment of groundwater arsenic contamination using machine learning in Varanasi, Uttar Pradesh, India*. J Water Health, 20 (5), str. 829–848.
19. Kumar, M., Husain, M., Upreti, N. i Gupta, D. (2010). *Genetic Algorithm: Review and Application*
20. Markov, Z. i Russell, I. (2006). *An introduction to the WEKA data mining system*. ACM SIGCSE, 38, str. 367-368.
21. Munandar, T.A., i Winarko, E. (2015). *Regional Development Classification Model using Decision Tree Approach.*
22. Naghibi, S.A., Ahmadi, K. i Daneshi, A. (2017) *Application of Support Vector Machine, Random Forest, and Genetic Algorithm Optimized Random Forest Models in Groundwater Potential Mapping*. Water Resources Management, 31, str. 2761–2775.
23. NewGenApps (2018). *Random forest analysis in ML and when to use it*.
24. Žapčević S., Butala P., Otkrivanje znanja i metode rudarenja podataka u proizvodnim sistemimam. Univerzitet u Bihaću, Bihać
25. Pejić Bach, M., Kerep, I. *Weka - alat za otkrivanje znanja iz baza podataka*, mikrorad, Zagreb
26. Pejić Bach, M., Šarlija, N., Zoroja, J., Jaković, B. i Čosić, D. (2019). *Credit Risk Scoring in Entrepreneurship: Feature Selection. Managing Global Transitions*. 2019., vol. 17, issue 4, str.265-287

27. Saritas, M. i YASAR, A. (2019) *Performance Analysis of ANN and Naive Bayes 24. Classification Algorithm for Data Classification*. International Journal of Intelligent Systems and Applications, 7.
28. Vujović, Ž. (2021). *A case study of the application of WEKA software to solve the problem of liver inflamation.* PREPRINT (Version 1)
29. Whitley, D. A. (1994). *Genetic algorithm tutorial*. Stat Comput, 4, str. 65–85.
30. Witten, I.H., Frank, E., Trigg, L., Hall, M., Holmes, G. i Cunningham, S.J. (1999). *Weka: Practical machine learning tools and techniques with Java implementations*. Working paper (99/11). Hamilton, New Zealand: University of Waikato, Department of Computer Science.
31. Yuniati, D. i Sinaga, K. (2021). *Analytics-Based on Classification and Clustering Methods for Local Community Empowerment in Indonesia.* Soft Computing in Data Science, str. 133-145.

Prilozi

1. Portuguese Bank Marketing,
   Dostupno na: https://www.kaggle.com/datasets/aakashverma8900/portuguese-bank-marketing
2. Dummy Marketing Data for Classification,
   Dostupno na: https://www.kaggle.com/datasets/harrimansaragih/dummy-data-for-classification
3. Company Bankruptcy Prediction,
   Dostupno na: https://www.kaggle.com/datasets/fedesoriano/company-bankruptcy-prediction
4. Tour & Travels Customer Churn Prediction,
   Dostupno na: https://www.kaggle.com/datasets/tejashvi14/tour-travels-customer-churn-prediction
5. Credit Card Churn Prediction,
   Dostupno na: https://www.kaggle.com/datasets/anwarsan/credit-card-bank-churn

Popis slika

Popis tablica i grafikona

Životopis

Ime i Prezime: Mateo Korman

Datum rođenja: 30.10.1996.

Grad: Zagreb

Kontakt: 091/ 544-7113

E-mail: mateokorman1@net.hr

Vještine

- Napredni korisnik Microsoft Office paketa (Naglasak na Excel – VBA, Power Query)
- Google alati
- BPMN
- UIPATH
- SQL
- Python
- HTML, CSS
- Power BI, Tableau
- R for Data Science
- Microsoft Power Platfom
- Java
- JSON, XML, XSD

Iskustvo:

SIJEČANJ, 2022. – TRENUTNO

- AML Software Developer / Data Analyst / Intesa Sanpaolo International Value Services

SIJEČANJ, 2021. – SIJEČANJ, 2022

- Računovodstvo i financije / Meblo Trade

RUJAN, 2020.- STUDENI, 2021.

- Računovodstvo / Konto-Modus

LIPANJ, 2015. – RUJAN,2016.

- Voditelj studenata / Gebruder Weiss

Obrazovanje:

STUDENI, 2020. – TRENUTNO

- Ekonomski fakultet Zagreb - Specijalistički diplomski stručni studij „Elektroničko poslovanje u privatnom i javnom sektoru“

LIPANJ, 2016. – RUJAN, 2020.

- VŠS / Ekonomski fakultet Zagreb
  Bacc.oec. Računovodstva i financija

RUJAN, 2011. – SVIBANJ, 2015.

- SSS / Druga ekonomska škola Zagreb

Jezične sposobnosti:

Engleski- B2

Njemački – A1

Talijanski – A2

Aktivnosti i hobiji:

Planinarenje, Pub kvizovi, Sportovi, Putovanja

Ostalo:

Vozačka dozvola B kategorije

Izjava o akademskoj čestitosti

IZJAVA O AKADEMSKOJ ČESTITOSTI

Izjavljujem i svojim potpisom potvrđujem da je diplomski rad / prijava teme diplomskog rada isključivo rezultat mog vlastitog rada koji se temelji na mojim istraživanjima i oslanja se na objavljenu literaturu, a što pokazuju korištene bilješke i bibliografija.

Izjavljujem da nijedan dio rada / prijave teme nije napisan na nedozvoljen način, odnosno da je prepisan iz necitiranog izvora te da nijedan dio rada / prijave teme ne krši bilo čija autorska prava.

Izjavljujem, također, da nijedan dio rada / prijave teme nije iskorišten za bilo koji drugi rad u bilokojoj drugoj visokoškolskoj, znanstvenoj ili obrazovnoj ustanovi.

(vlastoručni potpis studenta)

Zagreb, rujan 2023.

(mjesto i datum)

STATEMENT ON THE ACADEMIC INTEGRITY

I hereby declare and confirm by my signature that the final thesis is the sole result of my own work based on my research and relies on the published literature, as shown in the listed notes and bibliography.

I declare that no part of the thesis has been written in an unauthorized manner, i.e., it is not transcribed from the non-cited work, and that no part of the thesis infringes any of the copyrights.

I also declare that no part of the thesis has been used for any other work in any other higher education, scientific or educational institution.

(Personal signature of the student)

Zagreb, September 2023

(place & date)