Otkrivanje znanja u bazama podataka

Vinković, Goran

Undergraduate thesis / Završni rad

2017

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Pula / Sveučilište Jurja Dobrile u Puli**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:137:763328

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-29**

Image not found or type unknown

Repository / Repozitorij:

Digital Repository Juraj Dobrila University of Pula

zir.nsk.hr

Image not found or type unknown

Sveučilište Jurja Dobrile u Puli

Fakultet ekonomije i turizma

«Dr. Mijo Mirković»

GORAN VINKOVIĆ

OTKRIVANJE ZNANJA U BAZAMA PODATAKA

Završni rad

Pula, kolovoz 2017.

Sveučilište Jurja Dobrile u Puli

Fakultet ekonomije i turizma

«Dr. Mijo Mirković»

GORAN VINKOVIĆ

OTKRIVANJE ZNANJA U BAZAMA PODATAKA

Završni rad

JMBAG: 0303038043, redoviti student

Studijski smjer: Informatika

Predmet: Upravljački sustavi

Znanstveno područje: Društvene znanosti

Znanstveno polje: Informacijske i komunikacijske znanosti

Znanstvena grana: Informacijski sustavi i tehnologija

Mentorica: Prof. dr. sc. Vanja Bevanda

Pula, kolovoz 2017.

IZJAVA O AKADEMSKOJ ČESTITOSTI

Ja, dolje potpisani _____, kandidat za prvostupnika _____ovime izjavljujem da je ovaj Završni rad rezultat isključivo mojega vlastitog rada, da se temelji na mojim istraživanjima te da se oslanja na objavljenu literaturu kao što to pokazuju korištene bilješke i bibliografija. Izjavljujem da niti jedan dio Završnog rada nije napisan na nedozvoljen način, odnosno da je prepisan iz kojega necitiranog rada, te da ikoji dio rada krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za koji drugi rad pri bilo kojoj drugoj visokoškolskoj, znanstvenoj ili radnoj ustanovi.

Student

_____

U Puli, _____, _____ godine

IZJAVA

o korištenju autorskog djela

Ja, _____ dajem odobrenje Sveučilištu Jurja Dobrile u Puli, kao nositelju prava iskorištavanja, da moj završni rad pod nazivom _____ _____ koristi na način da gore navedeno autorsko djelo, kao cjeloviti tekst trajno objavi u javnoj internetskoj bazi Sveučilišne knjižnice Sveučilišta Jurja Dobrile u Puli te kopira u javnu internetsku bazu završnih radova Nacionalne i sveučilišne knjižnice (stavljanje na raspolaganje javnosti), sve u skladu s Zakonom o autorskom pravu i drugim srodnim pravima i dobrom akademskom praksom, a radi promicanja otvorenoga, slobodnoga pristupa znanstvenim informacijama.

Za korištenje autorskog djela na gore navedeni način ne potražujem naknadu.

U Puli, _____ (datum)

Potpis

_____

Sadržaj

1. RUDARENJE PO PODATCIMA 2
1.1 Proces rudarenja po podatcima 4
1.2 Veliki skupovi podataka 7
2. PRIPREMA PODATAKA 11
2.1 Reprezentacija neobrađenih podataka 11
2.2 Karakteristike neobrađenih podataka 13
2.3 Transformacija podataka 15
2.3.1 Normalizacija 15
2.3.2 Uglađivanje podataka 16
2.3.3 Razlike i omjeri 16
2.4 Podatci koji nedostaju 17
3. ALATI ZA RUDARENJE PO PODATCIMA 19
3.1 Alati 19
3.1.2 Weka 20
3.2 Podatci 21
3.3 Regresija 22
3.4 Klasifikacija 24
3.4.1 Naive Bayes klasifikator 25
3.4.2 K- nearest neighbors (k-NN) algoritam 27
3.4.3 OneR 29
3.4.4 Stabla odlučivanja 31
3.4.5 J48 31
3.5 Klasteriranje 33
3.5.1 K-means klasteriranje 34
3.5.2 EM algoritam 36
3.5.3 Hijerarhijsko klasteriranje 37
4. VALIDACIJA 39
4.1 Cross validacija 40
4.2 Konfuzijska matrica 41
4.3 Split validacija 42
5. USPOREDBA 43
6. ZAKLJUČAK 45
7. LITERATURA 46
8. SAŽETAK 47

UVOD

Cilj ovog rada jest opisati i objasniti proces otkrivanja znanja u bazama podataka, odnosno rudarenja po podatcima, kao i proces pripreme i obrade podataka. Prije nego što se može započeti s rudarenjem po podatcima potrebno je nabaviti dobar skup podataka. Međutim skup podataka nije uvijek dobar te je te podatke ponekad potrebno obraditi kako bi se dobili željeni rezultati. Podatci se mogu obraditi na više načina koji su detaljnije opisani dalje u radu.

Rudarenje po podatcima može se definirati kao proces otkrivanja znanja pomoću raznih metoda koje se primjenjuju na skupove podataka. U ovome radu koristit će se nekoliko metoda i algoritama kako bi ih se bolje objasnilo, te vidjelo na primjerima kako se te metode i algoritmi primjenjuju.

Valja napomenuti kako se u ovom radu neće spomenuti neuronske mreže, deep learning i skladišta podataka.

Za potrebe rada koristit će se alat Weka za primjenu metoda i algoritama. Podatci s kojima će se raditi u Weki su skupovi podataka koji su opisani dalju u radu. Cilj je opisati metode i algoritme te vidjeti kako ih se primjenjuje.

Metode i algoritmi koji će se koristiti u radu su sljedeći; linearna regresija, Naive Bayes klasifikator, k-nearest neighbor, OneR, J48, k-means, EM algoritam i hijerarhijsko klasteriranje.

J48 spada u metode stabla odlučivanja. Stabla odlučivanja se koriste za predviđanje, razvrstavanje, grupiranje i vizualizaciju podataka. Ona su izgrađena od čvorova koji su povezani granama i listova.

K-means, EM algoritam i hijerarhijsko klasteriranje su metode klasteriranja. Klasteriranje je metoda čiji algoritmi traže sličnosti unutar zadanog skupa podataka te ih grupiraju na temelju zajedničkih karakteristika u klastere.

Na kraju rada se nalazi tablica u kojoj su korištene metode ukratko opisane i uspoređene.

1. RUDARENJE PO PODATCIMA

S napretkom informatičkih tehnologija i sve većom dostupnošću tih tehnologija široj populaciji, činjenica je da postoje ogromne količine podataka koje pune računala, mreže i živote. Vladine agencije, znanstvene institucije, i poduzeća su posvetili ogromnu količinu resursa za prikupljanje i pohranjivanje podataka, dok u stvarnosti vrlo mala količina tih podataka će se koristiti jer u mnogim slučajevima, količine su jednostavno prevelike za upravljanje, ili je sama struktura tih podataka prekomplicirana za analiziranje. Glavni je razlog taj što je često izvorni napor za stvaranje skupa podataka usmjeren na pitanja poput učinkovitosti skladištenja, ali to ne uključuje plan kako će se ti podatci koristiti i analizirati. (Kantardžić, 2011)

Potreba za razumijevanjem velikih, složenih, informacijom bogatih podataka je zajednička gotovo svim područjima poslovanja, znanosti i inženjeringa. U poslovnome svijetu podatci o tvrtki i klijentima su prepoznati kao strateška imovina. Sposobnost za izvlačenjem skrivenog korisnog znanja u tim podatcima i djelovanjem prema tome znanju je sve važnija u današnjem konkurentnom svijetu. Cijeli proces primjene računalne metodologije, uključujući nove tehnike, za otkrivanje znanja iz podataka se naziva rudarenje po podatcima. (Kantardžić, 2011., str. 2.)

Rudarenje po podatcima je računalni proces otkrivanja uzoraka u velikim skupovima podataka koji uključuju metode strojnog učenja, statistike i sustava baza podataka. To je interdisciplinarno područje računalnih znanosti. Cjelokupni cilj procesa prikupljanja podataka jest izvući podatke iz skupa podataka i pretvoriti ih u razumljivu strukturu za daljnju upotrebu. (Witten et al., 2016)

Osim koraka analize, rudarenje po podatcima uključuje aspekte upravljanja bazom podataka i podatcima, pred-procesiranje podataka, modeli i zaključna razmatranja, metrike zanimljivosti, razmatranje složenosti, vizualizacija, i online ažuriranje.

„Rudarenje po podatcima je analitički korak procesa otkrivanja znanja u bazama podataka (eng. Knowledge discovery in databases), poznatiji kao KDD.“ (Fayyad et al., 1996)

Najbolji rezultati se postižu balansiranjem znanja stručnjaka u opisivanju problema i ciljeva s mogućnostima pretraživanja kod računala. Dva primarna cilja rudarenja po

podatcima obično su predviđanje i opis. Predviđanje uključuje upotrebu varijabli i polja u skupu podataka za predviđanje nepoznatih ili budućih vrijednosti drugih varijabla od interesa. Opis se, s druge strane, usredotočuje na utvrđivanje uzoraka koji opisuju podatke koje ljudi mogu interpretirati. (Kantardžić, 2011., str. 3.)

Aktivnosti u rudarenju po podatcima je moguće svesti na dvije kategorije:

1. Prediktivno rudarenje – proizvodi model sustava opisanog od strane danog skupa podata.
2. Deskriptivno rudarenje – proizvodi nove informacije na temelju dostupnog seta podataka.

Ciljevi predviđanja i opisa se postižu korištenjem tehnika za rudarenje po podatcima za sljedeće primarne zadatke kod rudarenja: (Kantardžić, 2011., str. 3.)

1. Klasifikacija – Otkrivanje prediktivne funkcije učenja koja klasificira podatke u jednu od nekoliko definiranih klasa.
2. Regresija – Otkrivanje prediktivne funkcije učenja koja mapira podatak na varijablu predviđanja stvarne vrijednosti.
3. Grupiranje (klasteriranje) – Deskriptivni zadatak u kojem se nastoji identificirati konačan skup podataka ili klastera za opisivanje podataka.
4. Sažetak – Dodatni deskriptivni zadatak koji uključuje metoda za pronalaženje kompaktnog opisa za skup podataka.
5. Modeliranje ovisnosti – Pronalaženje lokalnog modela koji opisuje značajne zavisnosti.
6. Detekcija promjena i devijacija – Otkrivanje najznačajnijih promjena u skupu podataka.

Uspjeh rudarenja po podatcima uvelike ovisi o količini energije, znanja i kreativnosti koju rudar uloži. Rudarenje je poput rješavanja slagalice. Pojedini dijelovi nisu složene strukture, ali kao cjelina, oni mogu predstavljati vrlo razrađene i složene sisteme. Stoga, biti analitičar i dizajner u procesu rudarenja zahtijeva, osim temeljitog stručnog znanja, kreativno razmišljanje i mogućnost gledanja problema iz različitih perspektiva. (Kantardžić, 2011., str. 3.)

Rudarenje po podatcima je jedno od najbrže rastućih polja u računalnoj industriji. Jedna od najvećih prednosti rudarenja je široki raspon metodologije i tehnika koje se

mogu primijeniti na niz skupova problema. Budući da je rudarenje aktivnost koja se obavlja na velikim skupovima podataka, jedno od najvećih ciljnih tržišta je cijela zajednica za skladištenje podatka, data – mart, i podršku pri odlučivanju koja obuhvaća stručnjake iz industrije poput maloprodaje, proizvodnje, telekomunikacija, zdravstva, osiguranja i transportacije. U poslovnome svijetu, rudarenje po podatcima se može koristiti za otkrivanje novih trendova kupnje i planiranje strategija ulaganja. Može poboljšati marketinške kampanje, a ishodi se mogu koristiti kako bi se klijentima pružala više usmjerena podrška i pažnja. Tehnike rudarenja se mogu primijeniti i na probleme reinženjeringa poslovnog procesa gdje je cilj razumjeti interakcije i odnose između poslovnih praksi i organizacija. (Kantardžić, 2011., str. 4.)

1.1 Proces rudarenja po podatcima

Rudarenje po podatcima ima korijene u raznim disciplinama, od kojih su najvažnije statistika i strojno učenje.

Statistika ima svoje korijene u matematici, stoga postoji naglasak na matematičkoj strogosti i želja da se nešto utvrdi teorijski prije testiranja u praksi. Nasuprot tome, strojno učenje ima korijene u računalnoj praksi, što vodi do praktične orijentacije da se nešto testira da se vidi koliko dobro izvodi, bez čekanja formalnog dokaza učinkovitosti. Osnovni principi modeliranja u rudarenju po podatcima imaju korijene u teoriji kontrole, koja se primarno primjenjuje na inženjerske sustave i industrijske procese. Problem određivanja matematičkog modela za nepoznati sustav promatranjem njegovih ulaznih i izlaznih podataka se općenito naziva identifikacija sustava. Svrha identifikacije sustava je predvidjeti ponašanje sustava i objasniti interakciju i odnose između varijabli sustava. (Kantardžić, 2011., str. 5.)

Identifikacija sustava se sastoji od dva koraka:

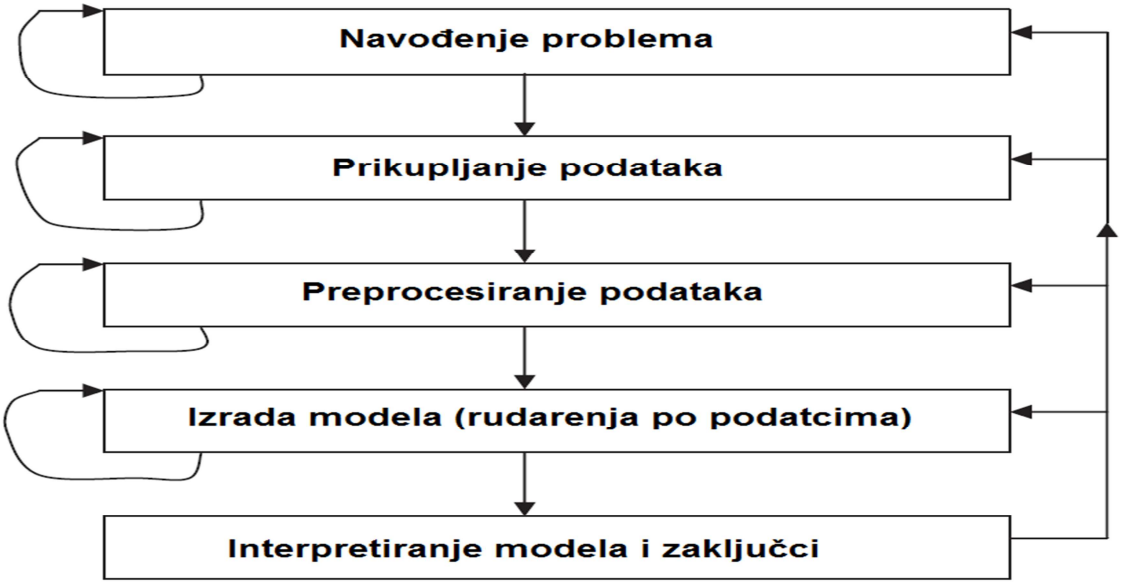
1. Identifikacije strukture – Primjenjuje se a priori znanje o ciljnom sustavu kako bi se odredile klase modela unutar kojih se vrši potraga za najboljim modelom. Često je ta klasa modela obilježena parametriziranom funkcijom $y = f(u, t)$, gdje je y izlaz modela, u je ulazni vektor i t je vektor parametra. Određivanje funkcije f je ovisno o problemu.

2. Identifikacija parametara - Kada je struktura modela poznata primjenjuju se tehnike optimizacije za utvrđivanje parametara vektora kako bi se odredio vektor parametar t tako da rezultat modela y* = f(u,t*) može opisati sustav na odgovarajući način.

Identifikacija sustava nije proces koji se odvija u jednom smjeru. I struktura i parametar identifikacije se moraju ponavljati dok nije nađen zadovoljavajući model. Koraci su sljedeći: (Kantardžić, 2011., str.5)

1. Odrediti i parametrizirati klasu matematičkih modela, y* = f(u, t*), koji predstavljaju sustav koji se identificira.
2. Provesti identifikaciju parametara kako bi se izabrali parametri koji najbolje pristaju dostupnom skupu podataka (razlika y – y* je minimalna).
3. Obaviti testove validacije kako bi se vidjelo da li model identificira odgovore točno na nevidljivim skupovima podataka.
4. Završiti proces kada je rezultat validacije testa zadovoljavajući.

Riječ proces je vrlo važna u definiciji rudarenja po podatcima.



Slika 1 Proces rudarenja po podatcima (Kantardžić, 2011., str. 9.)

Na slici 1 se nalazi pojednostavljeni grafički prikaz procesa rudarenja po podatcima.

U praksi rudarenje postaje iterativan proces. Istraživač proučava podatke, ispituje ih koristeći neku analitičku tehniku, odlučuje gledati ih na drugi način, možda ih mijenja, a zatim se vraća na početak i primjenjuje drugi alat za analizu podataka, dobivajući bolje ili drugačije rezultate. Taj se proces može ponoviti mnogo puta, svaki put s drugačijim pristupom prema podatcima. Proces uključuje sljedeće korake: (Kantardžić, 2011., str. 7.)

1. Navođenje problema i formuliranje hipoteze – Većina studija temeljena na modeliranju podataka se izvodi u određenoj domeni aplikacije. Potrebno je znanje za specifičnu domenu kako bi se formulirala izjava o problemu. U ovom koraku se određuje skup varijabli za nepoznatu ovisnost i opći oblik te ovisnosti kao početna hipoteza. Moguće je da postoji nekoliko hipoteza za jedan problem u ovome koraku.

2. Skupljanje podataka – U ovome koraku se generiraju i prikupljaju podatci. Postoje dva pristupa; osmišljen eksperiment, kod kojega je proces generiranja pod kontrolom stručnjaka, i opservacijski pristup, kod kojega stručnjak ne može utjecati na proces generiranja podataka. Opservacijski pristup, tj., generiranje nasumičnih podataka se provodi u većini aplikacija za rudarenje po podatcima.

3. Preprocesiranje podataka – U opservacijskom pristupu, podatci se obično prikupljaju iz postojećih baza podataka. Preprocesiranje uključuje dva zadatka:
   a) Otkrivanje i uklanjanje outliera – Outlieri su neuobičajene vrijednosti podataka koje nisu u skladu s većinom zapažanja, obično rezultat pogrešaka kod mjerenja, kodiranja ili nenormalnih vrijednosti. Rješavaju se detektiranjem i uklanjanjem kod preprocesiranja ili se razvijaju robusne metode modeliranja koje nisu osjetljive na outliere.
   b) Skaliranje, kodiranje i odabir značajki – preprocesiranje uključuje nekoliko koraka, kao što su skaliranje varijabli i različite vrste kodiranja. Na primjer, ako jedna značajka ima raspon [0, 1], a druga [-100, 1000], neće imati istu težinu u primijenjenoj tehnici, što će imati utjecaja na krajnje rezultate rudarenja. Stoga, potrebno ih je izmjeriti i postaviti na istu težinu za daljnju analizu.

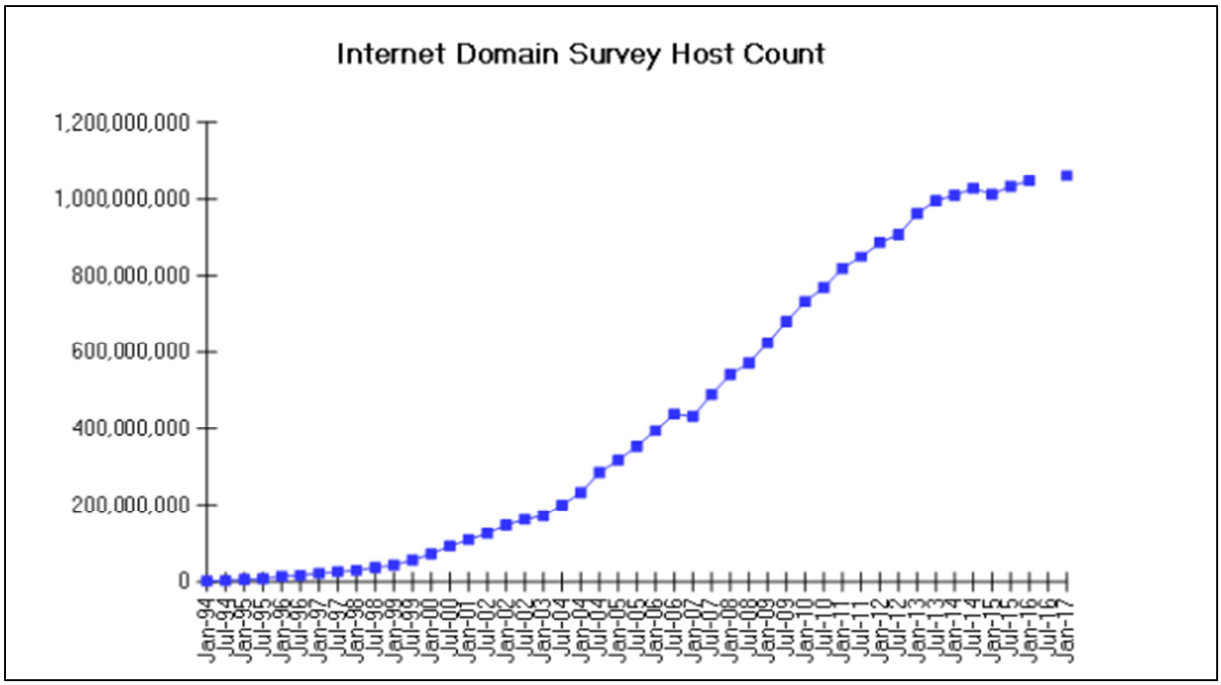
4. Procjena modela – U ovoj fazi se vrši odabir odgovarajuće tehnike za rudarenje po podatcima.

5. Tumačenje modela i zaključci – U većini slučajeva, modeli rudarenja po podatcima trebali bi pomoći u donošenju odluka. Takvi modeli moraju biti tumačivi kako bi bili korisni.

1.2 Veliki skupovi podataka

Informacijsko doba, s razvojem i rastom interneta, je uzrokovalo eksponencijalni rast izvora informacija i skladišnih jedinica za te informacije. Na slici 2 se može vidjeti rast Internet hostova od 1994. do 2017. godine

E-mailovi, blogovi, podatci o transakcijama i milijarde web stranica stvaraju terabajte novih podataka svaki dan i time se brzo povećava jaz između prikupljanja podataka, organizacije podataka i sposobnosti za analizom podataka. Trenutna tehnologija omogućava učinkovito, jeftino i pouzdano pohranjivanje podataka i pristup istima. U svome neobrađenome obliku, ti podatci imaju malo izravne vrijednosti. Ono što je vrijedno je što se može zaključiti iz podataka i staviti u uporabu. Na primjer, baza podataka marketinga neke potrošačke tvrtke može dati znanje o povezanosti između pojedinih stavki i određenih demografskih skupina. To se znanje može koristiti za nove ciljane marketinške kampanje s predvidljivim financijskim povratkom. U današnjem okruženju temeljenom na multimediji koja ima velike internetsku infrastrukturu, generiraju se različite vrste podataka i pohranjuju digitalno. Za pripremu odgovarajuće metode rudarenja po podatcima moraju se analizirati osnovne vrste i karakteristike skupova podataka. (Kantardžić, 2011., str. 10.)



Slika 2. Rast internet hostova od 1994-2017, pristupljeno 14.08.2017, 23:38, https://www.isc.org/network/survey/

Količina podataka kojom se raspolaže često je prevelika da bi se obradila ručnom analizom, čak i za računalne analize. Logično je da će neka osoba, odnosno voditelj raditi efektivnije ako raspolaže s velikom količinom podataka, nekoliko stotina ili tisuća podataka u arhivi. Poslovno društvo svjesno je s problemom opterećenosti informacijama, te jedna analiza pokazuje sljedeće: (Kantardžić, 2011., str. 11.)

1. 61% menadžera vjeruje kako je opterećenje informacija prisutno u njihovom radnom okruženju
2. 80% vjeruje kako će situacija postati još gora
3. Preko 50% menadžera ignoriraju podatke u trenutnom procesu donošenja odluka zbog preopterećenja informacija
4. 84% menadžera pohranjuje te podatke za budućnost, ne koriste ih u aktualnim analizama
5. 60% vjeruje da je trošak prikupljanja informacija veći od same vrijednosti

Najprihvatljivije rješenje na probleme ove analize jest zamjena klasičnih analiza i interpretacije podataka s novim tehnologijama rudarenja podataka. U teoriji, kod većine metoda koje se primjenjuju za rudarenje po podatcima trebalo bi se biti zadovoljno s velikim skupom podataka, iz toga razloga kad se raspolaže s većim

skupom podataka postoji veća mogućnost za prikupljanje vrijednih informacija. (Kantardžić, 2011., str. 11.)

Kako bi se odabrale odgovarajuće metode za rudarenje po podatcima, moraju se analizirati osnovni tipovi i karakteristike skupova podataka. Prvi korak u toj analizi je sistematizacija podataka u odnosu na kompjuterski prikaz i uporabu. Podatci koji su obično izvor za proces rudarenja po podatcima mogu biti klasificirani u strukturirane, polu strukturirane i nestrukturirane podatke. (Kantardžić, 2011., str. 12)

Većina poslovnih baza podataka sadrže strukturirane podatke koji se sastoje od definiranih polja s numeričkim ili alfanumeričkim vrijednostima, dok znanstvene baze podataka mogu sadržavati sve tri klase podataka.

Primjeri polu strukturiranih podataka su elektroničke slike dokumenata, medicinska izvješća, izvršni sažeci i slično. Većina web dokumenata također spada u ovu kategoriju. Primjer nestrukturiranih podataka je video snimljen nadzornom kamerom u nekom odjelu. Ova forma podataka generalno zahtjeva opsežnu analizu kako bi se izdvojila i strukturirala informacija koja je sadržana u njemu. Strukturirani podatci se obično odnose na tradicionalne podatke, dok polu strukturirani i nestrukturirani podatci spadaju u netradicionalne podatke (multimedija). Većina trenutnih metoda rudarenja po podatcima i komercijalnih alata se primjenjuje na tradicionalne podatke. (Kantardžić, 2011., str. 12.)

U literaturi rudarenja po podatcima koristi se pojam uzorak (eng. Sample) ili slučaj (eng. Case) za redove. Mnogo različitih tipova obilježja (atributi i varijable) polja u arhivi strukturiranih podataka su zajednička u rudarenju po podatcima. Sve metode rudarenja po podatcima nisu jednako dobre kada je riječ o različitim tipovima obilježja. Današnja računala i odgovarajući softverski alati podržavaju procesiranje skupova podataka koji imaju milijune uzoraka i stotine obilježja. Veliki skupovi podataka, uključujući one s miješanim tipovima podataka, su tipično inicijalno okruženje aplikacija za tehnike rudarenja po podatcima. Kada je velika količina podataka pohranjena u računalu ne mogu se odmah početi primjenjivati tehnike rudarenja po podatcima, zato što se prvo mora riješiti problem kvalitete podataka. Kvalitetna ručna analiza nije moguća u toj fazi. Iz tog razloga je vrlo važno pripremiti analizu kvalitete podataka u ranijim fazama procesa rudarenja po podatcima, obično u fazi preprocesiranja podataka. Kvaliteta podataka može ograničavati mogućnost

krajnjih korisnika za donošenje informiranih odluka. Postoji nekoliko indikatora kvalitete podataka, a to su: (Kantardžić, 2011., str. 13.)

1. Podatci moraju biti točni – analitičar mora provjeriti je li ime napisano ispravno, da je kod u datom rasponu, vrijednost je dovršena, itd.
2. Podatci moraju biti pohranjeni prema vrsti podataka
3. Podatci trebaju imati integritet – Ažuriranja se ne smiju izgubiti zbog sukoba među različitim korisnicima. Postupci sigurnosnog kopiranja i oporavka trebali bi biti implementirani u sustav upravljanja bazom podataka,tzv. DBMS (eng. Data Base Management System
4. Podatci trebaju biti konzistentni – oblik i sadržaj moraju biti isti nakon integracije skupova podataka iz različitih izvora
5. Podatci ne smiju biti suvišni – redundantni podatci trebaju biti svedeni na minimum, a duplicirani zapisi trebaju biti uklonjeni
6. Podatci trebaju biti pravodobni
7. Podatci trebaju biti razumljivi – standardi za imenovanje su neophodni, ali ne jedini uvjet da se podatci dobro razumiju.
8. Skup podataka treba biti potpun - podatci koji nedostaju trebaju biti minimalizirani. Podatci koji nedostaju mogu smanjiti kvalitetu globalnog modela.

2. PRIPREMA PODATAKA

Prije nego što se može započeti s rudarenjem po podatcima potrebno je nabaviti dobar skup podataka. Međutim skup podataka nije uvijek dobar te je te podatke ponekad potrebno obraditi kako bi se mogli dobiti željeni rezultati. Podatci se mogu obraditi na više načina.

2.1 Reprezentacija neobrađenih podataka

Dva najčešća tipa neobrađenih podataka su numerički i kategorijski. Numeričke vrijednosti uključuju varijable stvarne vrijednosti ili integer varijable kao što su dob, brzina ili duljina. Nasuprot tome, kategorijski (simbolički) podatci su varijable koje mogu biti jednake ili nejednake. One podržavaju samo odnos jednakosti (plava = plava ili crvena ≠ crna). Primjeri varijabla ove vrste su boja očiju, spol ili zemlja državljanstva. (Kantardžić, 2011., str. 27.)

Kategorijska varijabla s dvije vrijednosti može se pretvoriti u binarnu varijablu s dvije vrijednosti: 0 ili 1.

Vrijednost		Kod
Crna		1000
Plava		0100
Zelena		0010
Smeđa		0001

Slika 3. Primjer kategorijske vrijednosti (Kantardžić, 2011., str. 27.)

Kategorijska varijabla s n vrijednosti se može pretvoriti u n numeričku binarnu varijablu. Jedna binarna varijabla za svaku vrijednost. Ove kodirane kategorijske varijable u statistici su poznate kao „dummy“ varijable. (Kantardžić, 2011., str. 27.)

Drugi način klasificiranja varijable se temelji na njihovim vrijednostima, a one mogu biti kontinuirane ili diskretne varijable.

Kontinuirane varijable su poznate kao i kvantitativne ili metričke varijable. One se mjere pomoću intervalne ljestvice ili razmjera omjera. Obje mjere dopuštaju varijabli da bude precizno definirana ili mjerena. Razlika između ove dvije skale leži u tome kako je 0 definirana u skali. U intervalnoj ljestvici 0 se postavlja proizvoljno, i stoga ne ukazuje na potpunu odsutnost onoga što se mjeri. Na primjer, u intervalnoj skali

temperature gdje 0 stupnjeva Celzijusa ne znači potpunu odsutnost temperature. Također, 80 stupnjeva Celzijusa ne podrazumijeva duplo više topline od 40 stupnjeva Celzijusa. Zbog proizvoljnog položaja točke 0, omjer odnosa ne vrijedi. Nasuprot tome, skala razmjera omjera ima apsolutnu 0, a time odnos omjera vrijedi za izmjerene varijable pomoću te ljestvice. Veličine kao što su visina, dužina i plaća koriste ovu vrstu razmjera. Kontinuirane varijable su prikazane u velikim skupovima podataka s vrijednostima koje su stvarni ili cijeli brojevi. (Kantardžić, 2011., str. 28.)

Diskretne varijable se nazivaju i kvalitativnim varijablama. Takve varijable se mjere, odnosno njihove varijable su definirane, koristeći jednu od dvije nemetričkih skala – nominalna i ordinalna.

Nominalna skala je skala u kojoj se koriste simboli, znakovi i brojeva za prikaz različitih stanja (vrijednosti) varijable koja se mjeri. U ovoj skali brojevi i simboli ne prikazuju redoslijed. Na primjer, uslužni identifikator vrste klijenta s mogućim vrijednostima je stambeni, komercijalni i industrijski. Te se vrijednosti mogu kodirati abecednim redom kao A, B i C ili numerički kao 1, 2 i 3, ali nemaju metričke karakteristike kao drugi numerički podatci. Drugi primjer nominalne vrijednosti su poštanski brojevi koji se mogu pronaći u mnogim skupovima podataka. U oba primjera korišteni brojevi za različite vrijednosti nemaju poseban redoslijed niti odnos između jedan drugoga. (Kantardžić, 2011., str. 28.)

Tablica 1. Tipovi varijabli sa primjerima

Tip	Opis	Primjeri	Operacije
Nominalna	Upotrebljava oznaku ili naziv za razlikovanje objekata.	Poštanski broj, ID, spol	= ili ≠
Ordinalna	Upotrebljava vrijednosti za pružanje redoslijeda objekata.	Ocjene, pozicije	< ili >
Interval	Upotrebljava mjerne jedinice, ali je podrijetlo proizvoljno.	Celzijusi ili Fahrenheiti, datumi	+ ili -
Omjer	Upotrebljava mjerne jedinice, ali podrijetlo nije proizvoljno	Duljina, visina, dob, plaće	+, -, *, /

(Kantardžić, 2011., str. 28.)

Ordinalna skala se sastoji od uređenih, diskretnih gradacija, na primjer, ljestvica. Ordinalna varijabla je kategorijska varijabla za koju se definira redni odnos ali ne i udaljenost. Neki primjeri rednog atributa su pozicije na natjecanjima, zlato, srebro i bronca u sportskim natjecanjima ili nekakvi činovi (vojska). (Kantardžić, 2011., str. 28.)

Posebna klasa diskretnih varijabli su periodičke varijable. Periodna varijabla je značajka za koju postoji veza između udaljenosti, ali ne i redoslijeda. Primjeri su dani u tjednu, dani u mjesecu, ili dani u godini. Ponedjeljak i utorak su bliže od ponedjeljka i četvrtka, ali ponedjeljak može doći prije ili poslije petka. (Kantardžić, 2011., str. 28.)

Također klasifikacija podatka se temelji na njihovom ponašanju s obzirom na vrijeme. Neki se podatci ne mijenjaju s vremenom i ti podatci se smatraju statičnim podatcima. S druge strane postoje atributne vrijednosti koje se s vremenom mijenjaju i ti podatci se nazivaju dinamičkim podatcima. Većina metoda za rudarenje po podatcima je pogodnija za statičke podatke, tako da je potrebna posebna pažnja i preprocesiranja za rudarenje dinamičkih podataka. (Kantardžić, 2011., str. 28.)

2.2 Karakteristike neobrađenih podataka

Svi neobrađeni skupovi podataka koji su u početku pripremljeni za rudarenje su često preveliki. Mnogi su povezani s ljudskim bićima i imaju potencijal da budu neuredni. Analitičar treba očekivati vrijednosti koje nedostaju, izobličenja, pogrešno snimanje podataka, neodgovarajuće uzorkovanje i slične probleme u početnim skupovima podataka. Neobrađeni podatci koji ne prikazuju jedan od tih problema trebali bi odmah izazvati sumnju. Jedini pravi razlog za visoku kvalitetu podataka bi bio taj da su prezentirani podatci očišćeni i preprocesirani prije nego što ih analitičar vidi. Postoji mnogo razloga za neuredne podatke. Ponekad postoje greške u mjerenjima ili u snimkama, ali u mnogim slučajevima vrijednost nije dostupna. Kako bi se nosio s time u procesu rudarenja, analitičar mora biti sposoban napraviti model s podatcima koji su prezentirani, čak i ako njihove vrijednosti nedostaju. Ako je metoda dovoljno robusna, tada vrijednosti koje nedostaju nisu problem. U suprotnom, potrebno je riješiti problem nedostajućih vrijednosti prije primjene odabrane tehnike za rudarenje po podatcima. (Kantardžić, 2011., str. 31.)

Drugi uzrok neurednih podataka su pogrešno snimljeni podatci, što je tipično za velike količine podataka. Analitičar mora imati mehanizme za otkrivanje takvih neobičnih vrijednosti, a i u nekim slučajevima, i raditi s njima kako bi eliminirao njihov utjecaj na krajnji rezultat. Nadalje, podatci mogu i ne biti iz populacije s koje bi trebali biti. Outlieri su tipično primjeri i zahtijevaju pažljivu analizu prije nego što analitičar može odlučiti hoće li ih trebati izbaciti iz procesa obrade podataka kao anomalni ili uključeni kao neobični primjeri iz populacije koju se proučava. Vrlo je važno temeljito ispitati podatke prije poduzimanja bilo kakvih dodatnih koraka u formalnoj analizi. Tradicionalno, analitičari rudarenja po podatcima su se trebali upoznati s njihovim podatcima prije nego što bi ih počeli modelirati ili ih koristiti s nekim od algoritama za rudarenje. Međutim, s veličinom modernih skupova podataka to je manje moguće ili čak potpuno nemoguće u nekim slučajevima. Tu se analitičar oslanja na računalne programe za provjeru podataka. (Kantardžić, 2011., str. 32.)

„Izobličeni podatci, pogrešan izbor koraka u metodologiji, pogrešna primjena rudarskih alata, previše idealizirani model, model koji nadilazi različite izvore neizvjesnosti i dvosmislenosti u podatcima – sve to predstavlja mogućnost uzimanja pogrešnog smjera u procesu rudarenja po podatcima. Rudarenje nije samo primjena alata na određeni problem, već proces kritičnog procjenjivanja, istraživanja, testiranja i evaluacije.“ (Kantardžić, 2011., str. 32.)

Podatci trebaju biti dobro definirani i dosljedni, a količina podataka bi trebala biti dovoljno velika da podržava analizu podataka, upite, izvješćivanje i usporedbe povijesnih podataka putem dužeg perioda vremena.

Neobrađeni podatci nisu uvijek najbolji skup podataka za rudarenje. Mnoge transformacije mogu biti potrebne za izradu značajki korisnijih za odabrane metode rudarenja.

Postoje dva glavna zadatka za pripremu podataka: (Kantardžić, 2011., str. 33.)

1. Organiziranje podataka u standardni oblik koji je spreman za obradu rudarenjem.
2. Priprema skupova podataka koji vode do najboljih performansi rudarenja po podatcima.

2.3 Transformacija podataka

Odabir i upotreba tehnika u određenim aplikacijama ovisi o vrsti podataka, količini podataka i općim karakteristikama za prikupljanje podataka. Postoji nekoliko generalnih vrsta transformacije podataka.

2.3.1 Normalizacija

Neke metode rudarenja po podatcima, obično one koje se temelje na izračunu udaljenosti između točaka u n-dimenzionalnom prostoru, mogu trebati normalizirane podatke za najbolje rezultate. Mjerene vrijednosti mogu se skalirati na određeni raspon, na primjer, [-1, 1] ili [0, 1]. Ako se vrijednosti ne normaliziraju, mjere udaljenosti će pretežiti te značajke, koje u prosjeku imaju veće vrijednosti. Sljedeće tri tehnike su jednostavne i učinkovite tehnike normalizacije: (Kantardžić, 2011., str. 34.)

1. Decimalno skaliranje – Decimalna skala pomiče decimalnu točku, ali čuva većinu originalne vrijednosti. Tipična ljestvica održava vrijednosti u rasponu od -1 do 1.
2. Min – Max normalizacija – Na primjer, podatci za značajku v su u rasponu između 150 i 250. Tada će prethodna metoda normalizacije dati sve normalizirane podatke između .15 i .25, ali će akumulirati vrijednosti na malom subintervalu cijelog raspona. Da bi se postigla bolja raspodjela vrijednosti na normaliziranome intervalu, na primjer, [0, 1], može se koristiti min – max formula

   $$v'(i) = (v(i) - min[v(i)]) / (max[v(i)] - min[v(i)])$$

   gdje su minimalne i maksimalne vrijednosti za značajku v izračunate na skupu automatski, ili ih procjenjuje stručnjak na određenoj domeni. Računalno, postupak je vrlo jednostavan. (Kantardžić, 2011., str. 34.)

3. Normalizacija standardne devijacije – Normalizacija sa standardnom devijacijom radi dobro s mjerama udaljenosti, ali pretvara podatke u neprepoznatljiv oblik od izvornih podataka.

2.3.2 Uglađivanje podataka

Numerička značajka y, može biti u rasponu preko mnogih različitih vrijednosti. Za mnoge tehnike rudarenja po podatcima manje razlike između ove vrijednosti nisu značajne te mogu degradirati izvedbu metodu i krajnje rezultate. Ponekad je važno ugladiti vrijednosti varijable. Ako je skup vrijednosti za danu značajku F = {0.93, 1.01, 1.001, 3.02, 2,99, 5.03, 5.01, 4.98}, tada je očito da će glatke vrijednosti biti F smoothed = {1.0, 1.0, 1.0, 3.0, 3.0, 5.0, 5.0, 5.0}. Ova jednostavna transformacija se izvodi bez gubljenja kvalitete u skupu podataka, te u isto vrijeme smanjuje broj različitih vrijednosti na samo tri. (Kantardžić, 2011., str. 34.)

2.3.3 Razlike i omjeri

Najmanje promjene u podatcima mogu značajno poboljšati performansu rudarenja po podatcima. Učinci relativno manjih transformacija ulaznih ili izlaznih značajki posebno su važni u specifikaciji ciljeva rudarenja. Transformacija razlika i omjera, može značajno poboljšati cilj, osobito ako se primjenjuju na izlazne značajke. Ove transformacije ponekad proizvode bolje rezultate od početnog predviđanja broja.

Na primjer, u jednoj aplikaciji potrebno je postaviti kontrole proizvodnog procesa na optimalne postavke. Ali umjesto optimiziranja apsolutne specifikacije za output s(t+1), učinkovitije je postaviti cilj relativnog premještanja od trenutne vrijednosti do krajnje optimalne s(t+1) – s(t). Raspon vrijednosti za relativne poteze je općenito mnogo manji od raspona vrijednosti za postavku apsolutne kontrole. Stoga za mnoge metode rudarenja, manji broj alternativa će poboljšati efikasnost algoritma i često dati bolje rezultate. (Kantardžić, 2011., str. 35.)

Promjene razlike i omjera nisu samo korisne za outpute nego i za inpute. Mogu se koristiti kao promjene u vremenu za jednu značajku ili kao sastav različitih ulaznih

značajki. Na primjer, u mnogim skupovima medicinskih podataka postoje dvije značajke pacijenta (visina i težina) koje se uzimaju kao ulazni parametri za različite dijagnostičke analize. Mnoge aplikacije pokazuju da se dobivaju bolji rezultati kada se inicijalna transformacija provodi upotrebom nove osobine pod nazivom Index tjelesne mase (BMI), koji je omjer između težine i visine. (Kantardžić, 2011., str. 35.)

2.4 Podatci koji nedostaju

Za mnoge aplikacije rudarenja po podatcima, čak i kada postoje ogromne količine podataka, podskup slučajeva s potpunim podatcima može biti relativno mali. Dostupni uzorci i budući slučajevi mogu također imati vrijednosti koje nedostaju. Neke od metoda za rudarenje će prihvatiti vrijednosti koje nedostaju i zadovoljavajuće obrađuju podatke kako bi došli do zaključka. Druge metode zahtijevaju da sve vrijednosti budu dostupne. Najjednostavnije rješenje ovog problema je smanjenje skupa podataka i uklanjanje svih uzoraka s vrijednostima koje nedostaju. To je moguće kada su dostupni veliki skupovi podataka, a vrijednosti koje nedostaju pojavljuju se samo u malom postotku uzoraka. Ali, ako se ne odbace uzorci u kojima nedostaju vrijednosti, onda se moraju pronaći vrijednosti za njih. Postoji nekoliko praktičnih rješenja. (Kantardžić, 2011., str. 36.)

Prvo, analitičar može ručno pregledati uzorke koji nemaju vrijednosti i unijeti razumnu, vjerojatnu ili očekivanu vrijednost na temelju domene. Metoda je jednostavna za mali broj vrijednosti koje nedostaju i relativno male skupove podataka. Ali, ako nema očite ili vjerojatne vrijednosti za svaki slučaj, analitičar uvodi šum u skup podataka ručnim generiranjem vrijednosti. Te vrijednosti mogu biti: (Kantardžić, 2011., str. 36.)

1. Zamjena svih vrijednosti koje nedostaju s jednom globalnom konstantom
2. Zamjena vrijednosti koja nedostaje sa svojom značajkom
3. Zamjena vrijednosti koja nedostaje svojom značajkom srednje vrijednosti za danu klasu

Njihov glavni nedostatak je taj da ta supstituirana vrijednost nije točna vrijednost. Zamjenom vrijednosti koja nedostaje s konstantom ili mijenjanjem vrijednosti za nekoliko različitih značajki, podatci postaju pristrani. Jedno moguće tumačenje vrijednosti koje nedostaju je to da su one takozvane „nije nas briga“ vrijednosti.

Odnosno, pretpostavljamo da ove vrijednosti nemaju nikakav utjecaj na krajnji rezultat rudarenja. Općenito, često je varljivo i spekulativno zamijeniti vrijednosti koje nedostaju. Najbolje je generirati više rješenja rudarenja po podatcima s i bez značajki koje nedostaju te ih zatim analizirati i interpretirati. (Kantardžić, 2011., str. 37.)

3. ALATI ZA RUDARENJE PO PODATCIMA

Algoritam u rudarenju po podatcima je skup izračuna koji stvaraju model iz podataka. Da bi se izradio model, algoritam prvo analizira podatke koji su uneseni tražeći određene vrste uzoraka ili trendova. Algoritam koristi rezultate ove analize tijekom mnogih iteracija kako bi pronašao optimalne parametre za izradu rudarskog modela. Ti se parametri zatim primjenjuju na cijeli skup podataka radi izdvajanja izvedivih obrazaca i detaljnih statistika.

Model kojeg algoritam stvara iz podataka može imati različite oblike, uključujući:

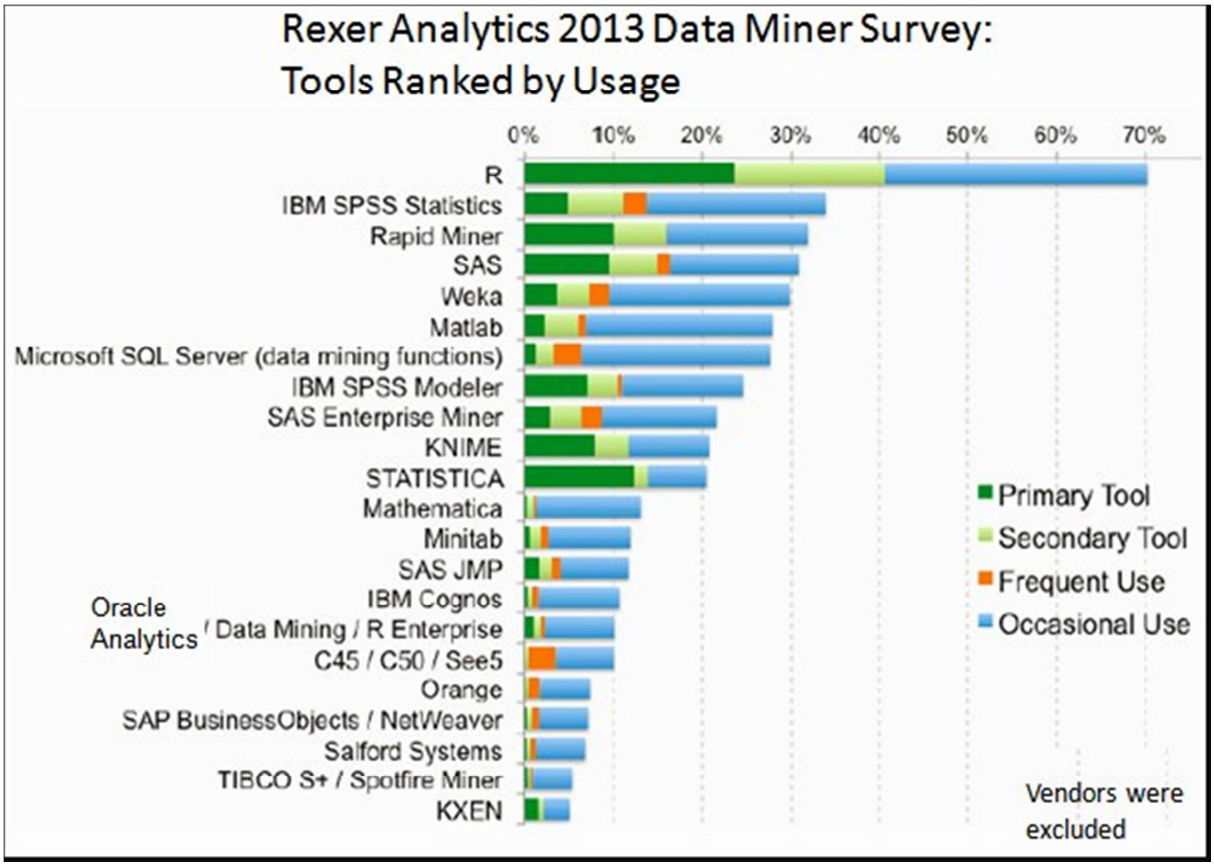
- Skup klastera koji opisuju kako se slučajevi u skupu podataka odnose.
- Stablo odluke koje predviđa ishod i opisuje kako različiti kriteriji utječu na taj ishod.
- Matematički model.
- Skup pravila koja opisuju kako su proizvodi grupirani zajedno u transakciji i vjerojatnosti da se proizvodi kupuju zajedno

Odabir najboljeg algoritma koji će se koristiti za određeni analitički zadatak može biti izazov. Iako se mogu koristiti različiti algoritmi za obavljanje istog zadatka, svaki algoritam proizvodi različite rezultate, a neki algoritmi mogu proizvesti više od jedne vrste rezultata.

3.1 Alati

Programi za rudarenje po podatcima omogućavaju rješavanje problema rudarenja podataka. Koristi ih se za rješavanja problema klasifikacije, klasteriranja, Bayesovih mreža, asocijativnih pravila i ostalih metoda rudarenja po podatcima. Postoje alati koji su besplatni za upotrebu te komercijalni alati. Neki od komercijalnih alata imaju trial verziju tako da ih se može besplatno probati na određeni period vremena.

Prema anketi koja je provedena 2013. godine od strane rexanalytics, a objavljena na stranici www.kdnuggets.com, top deset najkorištenijih alata za rudarenje po podatcima su, redom: R, STATISTICA, Rapid Miner, SAS, KNIME, IBM SPSS Modeler, IBM SPSS Statistics, Weka, SAS Enterprise Miner, Matlab.



Slika 4. Najkorišteniji alati za rudarenje podatcima 2013. godine, pristupljeno 04.09.2017, 23:01, http://www.kdnuggets.com/2013/10/rexer-analytics-2013-data-miner-survey-highlights.html

Postoje besplatni alati za rudarenje po podatcima, međutim postoje i oni koji su proizvedeni za prodaju te ne podržavaju besplatne verzije. Neki od takvih alata navedeni su na stranici www.kdnuggets.com, a neki od njih su AdvancedMiner, Alteryx, BayesiaLab, Civis, Data Miner SoftwareKit, SAS Enterprise Miner, Synapse i mnogi drugi.

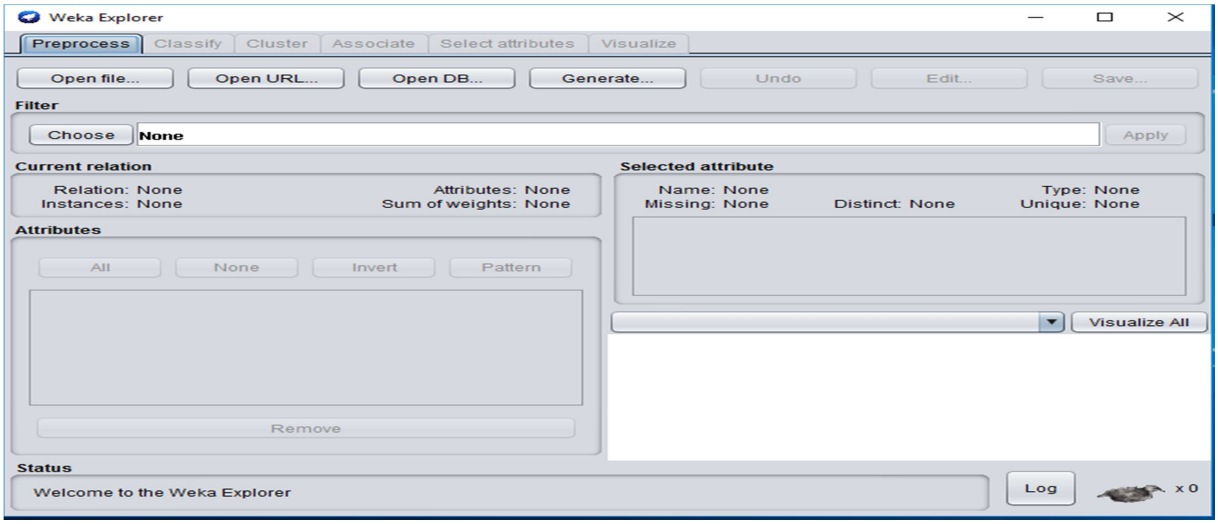
Postoji i mnogo besplatnih alata za rudarenje po podatcima a neki od njih su RapidMiner, Weka, Orange, KNIME, AlphaMiner, R, MiningMart, KEEL, TANGARA, OpenNN i tako dalje.

Za potrebe ovoga rada koristit će se alat Weka.

3.1.2 Weka

Weka (Waikato Environment for Knowledge Analysis) je poznat softver napisan u Javi, razvijen je na sveučilištu Waikato na Novom Zelandu. Ovaj alat sadrži kolekciju

vizualizacijskih alata i algoritama za analizu podataka i prediktivno modeliranje. Weka podržava nekoliko standardnih zadataka rudarenja podataka, kao što su pred obrada, klasteriranje, klasifikacija, regresija, vizualizacija te selekcija obilježja. Prednosti ovog alata su slobodno korištenje pod GNU (General Public Licence) licencom, prenosivost zato što je napisan u Javi te ga je moguće pokrenuti na gotovo svakoj modernoj platformi, opsežan skup pred obrade podataka, tehnika modeliranja te jednostavnost korištenja s obzirom na grafičko korisničko sučelje. Softver je dobio ime po znatiželjnoj ptici neletačici koja se može pronaći samo na Novom Zelandu.



Slika 5. Prikaz grafičkog sučelja u Weki

3.2 Podatci

Skup podataka koji se koristi prilikom analize regresije jest cpu.arff. Ovaj skup podataka se sastoji od 7 atributa, a to su MYCT (machine cycle time in nanoseconds (numeric)), MMIN (minimum main memory in kilobytes (numeric)), MMAX (maximum main memory in kilobytes (numeric)), CACH (cache memory in kilobytes (numeric)), CHMIN (minimum channels in units (numeric)), CHMAX (maximum channels in units (numeric)) i class. Klase su Minimum, Maximum, Mean i StdDev. Broj instanci iznosi 209.

Skup podataka koji se koristi prilikom analize Naive Bayes, k-NN, OneR i J48 klasifikatora jest weather.nominal.arff. Ovaj skup se sastoji od 5 atributa, a to su outlook, temperature, humidity, windy i play. U ovome skupu podataka svaki atribut

ima različite klase, redom; outlook sadrži klase sunny, overcast i rainy, temperature sadrži klase hot, mild i cold, humidity sadrži klase high i normal, windy sadrži klase TRUE i FALSE, dok play sadrži klase yes i no. Ovaj skup podataka sadrži 14 instanci.

Skup podataka koji se koristi prilikom k-means analize klastera, EM algoritma i hijerarhijskog klasteriranja jest glass.arff. Ovaj skup se sastoji od 10 atributa, a to su RI, Na, Mg, Al, Si, K, Ca, Ba, Fe i Type. Ovaj skup podataka sadrži 214 instanci.

3.3 Regresija

Regresija je funkcija rudarenja po podatcima koja predviđa broj. Dobit, prodaja, hipoteka, vrijednosti kuće, kvadratura, temperatura ili udaljenost mogu se predvidjeti pomoću tehnika regresije. Na primjer, regresijski model može se koristiti za predviđanje vrijednosti kuće na temelju lokacije, broja soba, veličine parcele i drugih čimbenika. (Witten et al., 2016)

Regresijski zadatak počinje sa skupom podataka u kojem su poznate ciljne vrijednosti. Na primjer, regresijski model koji predviđa vrijednosti kuće može se razviti na temelju promatranih podataka za mnoge kuće tijekom određenog vremenskog razdoblja. Osim vrijednosti, podatci bi mogli pratiti starost kuće, kvadrature, broj soba, poreze, blizinu škola i trgovačkih centara i tako dalje. Vrijednost kuće bila bi cilj, drugi atributi bili bi prediktori, a podatci za svaku kuću predstavljali bi slučaj. (Witten et al., 2016)

U procesu izgradnje modela, regresijski algoritam procjenjuje vrijednost cilja kao funkciju prediktora za svaki slučaj u podatcima izrade. Ovi odnosi između prediktora i cilja sažeti su u modelu koji se zatim može primijeniti na drugi skup podataka u kojem su ciljane vrijednosti nepoznate. (Witten et al., 2016)

Regresijski modeli se testiraju računanjem različitih statistika koje mjere razliku između predviđenih vrijednosti i očekivanih vrijednosti. Povijesni podatci za regresijski model obično su podijeljeni u dva skupa podataka: jedan za izgradnju modela, drugi za ispitivanje modela. (Witten et al., 2016)

Regresijsko modeliranje ima mnogo primjena u analizi trendova, planiranju poslovanja, marketingu, financijskim predviđanjima, predviđanju vremenskih serija, biomedicinskom i modeliranju odgovora na lijekove te modeliranju okoliša.

Uz pomoć alata Weke napravljena je analiza regresije nad skupom podataka cpu.arff, koji je opisan u poglavlju 4.2 Podatci. Prvi korak jest učitavanje skupa podataka. Nakon učitavanja mogu se vidjeti osnovni podatci o skupu podataka koji je učitan. Zatim se odabire metoda koja će se koristiti, u ovom slučaju odabire se Classify, a kao klasifikator (algoritam) odabire se LinearRegression. Zatim se taj algoritam primjenjuje na skup podataka, te se dobiju sljedeći rezultati:

```
Linear Regression Model

class =

      0.0491 * MYCT +
      0.0152 * MMIN +
      0.0056 * MMAX +
      0.6298 * CACH +
      1.4599 * CHMAX +
    -56.075

Time taken to build model: 0.06 seconds

=== Cross-validation ===
=== Summary ===

Correlation coefficient                  0.9012
Mean absolute error                     41.0886
Root mean squared error                 69.556
Relative absolute error                 42.6943 %
Root relative squared error             43.2421 %
Total Number of Instances              209
```

Slika 6 Rezultati algoritma regresije nad cpu.arff skupom podataka

Na slici 7 vidi se model koji je alat izgradio za ovaj skup podataka, a to je

*class = 0.0491 * MYCT + 0.0152*MMIN + 0.0056*MMAX + 0.6298*CACH + 1.4599*CHMAX + -56.075*

Kada se taj model primjeni dobije se da je Correlation coefficient 0.9012, što je vrlo pozitivna korelacija, što je vrlo dobar rezultat. Mean absolute error prikazuje prosječnu udaljenost predikcija modela nasuprot stvarnih podataka, što je u ovome slučaju 41.0886. Ostali rezultati su Root means squared error, koji iznosi 69.556, Relative absolute error, koji iznosi 43.6943%, Root relative square error koji iznosi 43.2421% i broj instanci koji iznosi 209. Rezultati su vrlo dobri za izgrađeni model.

3.4 Klasifikacija

Klasifikacija je funkcija rudarenja po podatcima koja dodjeljuje stavke u skupu podataka ciljanim kategorijama ili klasama. Cilj klasifikacije je precizno predvidjeti ciljnu klasu za svaki slučaj u podatcima. Na primjer, klasifikacijski model može se koristiti za identifikaciju podnositelja kredita kao niski, srednji ili visoki kreditni rizici. (Witten et al., 2016)

Zadatak klasificiranja počinje skupom podatka u kojem su poznati klasni zadatci. Na primjer, model klasifikacije koji predviđa kreditni rizik mogao bi se razviti na temelju promatranih podataka za mnoge podnositelje kredita tijekom određenog vremenskog razdoblja. Osim povijesnog kreditnog rejtinga, podatci mogu pratiti povijest zaposlenja, vlasništvo nad stanovima ili najam, godina zaposlenja, broj i vrsta ulaganja i tako dalje. Kreditna ocjena bi bila cilj, drugi atributi bi bili prediktori, a podatci za svakog klijenta predstavljaju slučaj. (Witten et al., 2016)

Klasifikacije su diskretne i ne označavaju redoslijed. Kontinuirane vrijednosti s pomičnim zarezom pokazuju numerički, a ne kategorični cilj. Prediktivni model s numeričkim ciljem koristi regresijski algoritam, a ne algoritam klasifikacije. (Witten et al., 2016)

Najjednostavniji tip klasifikacijskog problema je binarna klasifikacija. U binarnoj klasifikaciji, ciljni atribut ima samo dvije moguće vrijednosti: na primjer, visoki ili niski kreditni rejting. Višestruki ciljevi imaju više od dvije vrijednosti: na primjer, nizak, srednji, visok ili nepoznat kreditni rejting.

U procesu izgradnje modela, klasifikacijski algoritam pronalazi odnose između vrijednosti prediktora i vrijednosti cilja. Različiti algoritmi klasifikacije koriste različite tehnike za pronalaženje odnosa. Ti su odnosi sažeti u modelu koji se zatim može primijeniti na drugi skup podataka u koje su zadatci klase nepoznati. Modeli razvrstavanja testirani su usporedbom predviđenih vrijednosti s poznatim ciljnim vrijednostima u skupu testnih podataka. Podatci za klasifikacijski projekt obično su podijeljeni u dva skupa podataka: jedan za izgradnju modela, a drugi za ispitivanje modela. (Witten et al., 2016)

Bodovanje modela klasifikacije rezultira zadatcima klase i vjerojatnostima za svaki slučaj. Na primjer, model koji klasificira korisnike kao nisku, srednju ili visoku vrijednost također bi predvidio vjerojatnost svake klasifikacije za svakog korisnika. Klasifikacija ima mnoge primjene u segmentaciji kupaca, modeliranju poslovanja, marketingu, analizi kredita i tako dalje. (Witten et al., 2016)

3.4.1 Naive Bayes klasifikator

Ovaj algoritam pripada skupini klasifikacijskih algoritama. Pretpostavka ovog algoritma jest da atributi nisu ovisni jedan od drugog te da svi atributi imaju jednaku važnost. Osnovni koncept ovog algoritma počiva na uvjetnoj vjerojatnosti. Uvjetna vjerojatnost je definirana kao:

$$P(a|b) = m$$

Uvjetnu vjerojatnost iz ove jednadžbe definira se kao vjerojatnost događaja a uz uvjet b iznosi m. (Witten et al., 2016)

„Uvjetna vjerojatnost reducira polje slučajnih događaja, te nosi dodatnu informaciju reducirajući pri tome stupanj neizvjesnosti ishoda događaja.“ (Witten et al., 2016)

Temeljno pravilo vjerojatnosti događaja x i y glasi:

$$P(x, y) = P(x|y)P(y),$$

na temelju ovog pravila proizlazi:

$$P(x|y)P(y) = P(y|x)P(x)$$

Iz čega je izvedeno Bayesovo pravilo uvjetne vjerojatnosti:

$$P(y|x) = \frac{P(x|y)P(y)}{P(x)}$$

Naive Bayes klasifikator pretpostavlja da prisustvo ili odsustvo određenog obilježja klase nije povezano sa prisustvom ili odsustvom bilo koje drugog obilježja, odnosno atributi su neovisni. (Witten et al., 2016)

Koristeći Weku primijenit će se Naive Bayes klasifikator nad weather.nominal.arff skupom podataka, koji je opisan u poglavlju 4.2 Podatci. Učitavaju se podatci,

odabire se metodu Classify te zatim se odabire NaiveBayes za klasifikator koji će se primijeniti, te se dobivaju sljedeći rezultate:

Rezultati za klasu yes iznose 0.63, dok rezultati za klasu no iznose 0.33. Na temelju rezultata analize može se zaključiti da s obzirom na vremenske uvijete vjerojatnost da je pogodno vrijeme za igrati golf iznosi 63%, a da nije iznose 33%.

Iz konfuzijske matrice, koja se nalazi na slici 9, može se zaključiti da je algoritam točno klasificirao 8 instanci dok je pogrešno klasificiranih instanci 6.

```
Naive Bayes Classifier

                Class
Attribute         yes     no
               (0.63) (0.38)
==============================
outlook
  sunny            3.0    4.0
  overcast         5.0    1.0
  rainy            4.0    3.0
  [total]         12.0    8.0

temperature
  hot              3.0    3.0
  mild             5.0    3.0
  cool             4.0    2.0
  [total]         12.0    8.0

humidity
  high             4.0    5.0
  normal           7.0    2.0
  [total]         11.0    7.0

windy
  TRUE             4.0    4.0
  FALSE            7.0    3.0
  [total]         11.0    7.0



Time taken to build model: 0 seconds

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

Correctly Classified Instances           8               57.1429 %
Incorrectly Classified Instances         6               42.8571 %
Kappa statistic                         -0.0244
Mean absolute error                      0.4374
Root mean squared error                  0.4916
Relative absolute error                 91.8631 %
Root relative squared error             99.6492 %
Total Number of Instances               14

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0,778    0,800    0,636      0,778    0,700      -0,026   0,578     0,697     yes
               0,200    0,222    0,333      0,200    0,250      -0,026   0,578     0,557     no
Weighted Avg.  0,571    0,594    0,528      0,571    0,539      -0,026   0,578     0,647

=== Confusion Matrix ===

 a b   <-- classified as
 7 2 | a = yes
 4 1 | b = no
```

Slika 7. Rezultati Naive Bayes klasifikatora nad weather.nominal.arff skupom podataka

3.4.2 K- nearest neighbors (k-NN) algoritam

K-nearest neighbors (k-NN) je neparametarska metoda koja se koristi za klasifikaciju i regresiju. U oba slučaja ulaz se sastoji od najbližih primjera treninga u prostoru značajki. Izlaz ovisi o tome koristi li se k-NN za klasifikaciju ili regresiju. (Witten et al., 2016)

U k-NN klasifikaciji, output je klasno članstvo. Objekt se klasificira većinskim glasom svojih susjeda, a predmet se dodjeljuje najčešćim klasama među najbližim susjedima (k je pozitivan cijeli broj, obično mali). Ako je k=1, objekt je dodijeljen jednostavno klasi tog najbližeg susjeda . U k-NN regresiji, izlaz je vrijednost svojstva objekta. Ova vrijednost je prosjek vrijednosti njegovih najbližih susjeda. K-NN je vrsta učenja temeljenog na instancama, odnosno lijenog učenja (eng. Lazy learning), gdje je funkcija aproksimirana samo lokalno i sve se računanje odgađa do klasifikacije. K-NN algoritam je jedan od najjednostavnijih algoritama strojnog učenja. Susjedi su preuzeti iz skupa predmeta za koje je poznata klasa(klasifikacija) ili vrijednost objekta (regresija). To se može smatrati skupom vježbanja za algoritam, iako nije potreban korak treniranja. (Witten et al., 2016)

Pošto je k-NN lijen algoritam, to znači da ne koristi točke treninga za provođenje generalizacije. To znači da je trening faza vrlo brza. Nedostatak generalizacije znači da k-NN čuva sve podatke iz treninga. Većina lijenih algoritama, naročito k-NN, donosi odluku temeljenu na cjelokupnom skupu podataka. (Witten et al., 2016)

U Weki se koristi skup podataka weather.nominal.arff, opisan u poglavlju 4.2 Podatci, kako bi se testirao k-NN algoritam. Učitavaju se podatci, odabire se metodu Classify, a algoritam se odabire iz lazy datoteke. Algoritam koji se treba odabrati jest Ibk (instance-based learning), te se u prvom testiranju namješta da je parametar k=1. Dobivaju se sljedeći rezultati:

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

Correctly Classified Instances           8               57.1429 %
Incorrectly Classified Instances         6               42.8571 %
Kappa statistic                          0.0667
Mean absolute error                      0.4911
Root mean squared error                  0.5985
Relative absolute error                103.137  %
Root relative squared error            121.313  %
Total Number of Instances               14

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0,667    0,600    0,667      0,667    0,667      0,067    0,500     0,703     yes
               0,400    0,333    0,400      0,400    0,400      0,067    0,456     0,396     no
Weighted Avg.  0,571    0,505    0,571      0,571    0,571      0,067    0,484     0,593

=== Confusion Matrix ===

 a b   <-- classified as
 6 3 | a = yes
 3 2 | b = no
```

Slika 8. Rezultati k-NN algoritma sa parametrom k=1

Rezultat je 8 točno identificiranih instanci, odnosno njih 57.1%, dok ih je netočno klasificirano 6, odnosno njih 42.9%. Isti rezultat kao i kod Naive Bayes algoritma. Testira se još jednom, ali ovaj put će se staviti da je parametar k=3. Rezultat je sljedeći:

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

Correctly Classified Instances           9               64.2857 %
Incorrectly Classified Instances         5               35.7143 %
Kappa statistic                          0.1026
Mean absolute error                      0.4414
Root mean squared error                  0.4747
Relative absolute error                 92.699  %
Root relative squared error             96.2242 %
Total Number of Instances               14

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0,889    0,800    0,667      0,889    0,762      0,122    0,689     0,821     yes
               0,200    0,111    0,500      0,200    0,286      0,122    0,644     0,578     no
Weighted Avg.  0,643    0,554    0,607      0,643    0,592      0,122    0,673     0,734

=== Confusion Matrix ===

 a b   <-- classified as
 8 1 | a = yes
 4 1 | b = no
```

Slika 9. Rezultat k-NN algoritma sa parametrom k=3

Vidi se da je rezultat nešto drugačiji kada parametar k=3. U ovome slučaju broj točno klasificiranih instanci je 9, odnosno njih 64.3%, dok je netočno klasificiranih instanci 5, odnosno njih 35.7%. Valja napomenuti kako za ovaj skup podataka daljnje povećavanje parametra k neće utjecati na rezultat.

3.4.3 OneR

OneR, kratica za „Jedno pravilo“ (eng. One rule), je jednostavan ali precizan algoritam klasifikacije koji generira jedno pravilo za svaki prediktor u podatcima, a zatim odabire pravilo s najmanjom ukupnom pogreškom kao jedno pravilo. OneR proizvodi pravila koja nešto manje precizna od najsuvremenijih algoritama za klasifikaciju, ali stvara pravila koja su jednostavna za interpretaciju. OneR radi na sljedeći način: (Witten et al., 2016)

Za svaki prediktor, za svaku vrijednost prediktora, napravi pravilo;

- Broji koliko puta se svaka vrijednost od ciljane klase pojavljuje
- Pronađi najčešću klasu
- Neka pravilo dodijeli tu klasu na ovu vrijednost prediktora
- Izračunaj ukupnu pogrešku pravila svakog prediktora
- Odabir prediktora s najmanjom ukupnom pogreškom

U Weki se koristi weather.nominal.arff skup podataka, opisan u poglavlju 4.2 Podatci, kako bi se testirao OneR algoritam. Učitavaju se podatci, te se odabire metoda Classify i algoritam OneR iz rules datoteke. Dobivaju se sljedeći rezultati:

```
=== Classifier model (full training set) ===

outlook:
        sunny   -> no
        overcast        -> yes
        rainy   -> yes
(10/14 instances correct)


Time taken to build model: 0 seconds

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

Correctly Classified Instances           6               42.8571 %
Incorrectly Classified Instances         8               57.1429 %
Kappa statistic                         -0.1429
Mean absolute error                      0.5714
Root mean squared error                  0.7559
Relative absolute error                120      %
Root relative squared error            153.2194 %
Total Number of Instances               14

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0,444    0,600    0,571      0,444    0,500      -0,149   0,422     0,611     yes
                 0,400    0,556    0,286      0,400    0,333      -0,149   0,422     0,329     no
Weighted Avg.    0,429    0,584    0,469      0,429    0,440      -0,149   0,422     0,510

=== Confusion Matrix ===

 a b   <-- classified as
 4 5 | a = yes
 3 2 | b = no
```

Slika 10. Rezultat OneR algoritma nad weather.nominal.arff skupom podataka

U rezultatima se vidi da je 10 od 14 instanci točno po pravilu koje je OneR algoritam postavio. Međutim, vidi se da je točno klasificiranih instanci 6, odnosno njih 42.9%, dok je netočno klasificiranih instanci 8, odnosno njih 57.1%, što nije vrlo dobro.

3.4.4 Stabla odlučivanja

Stabla odlučivanja i pravila odlučivanja su metode za rudarenje po podacima koje se primjenjuju za rješavanje problema klasifikacije. Generiranje stabla odlučivanja jest vrlo učinkovita metoda stvaranja klasifikatora iz podataka te je ujedno i jedna od najkorištenijih logičkih metoda. Stablo odlučivanja predstavlja hijerarhijski model, a sastoji se od čvorova i grana. Atributi se testiraju u čvorovima, a grane predstavljaju sve moguće izlaze za testirani atribut u određenom čvoru. Algoritmi stabla odlučivanja spadaju u metode nadgledanog učenja. Postoji mnogo algoritama stabla odlučivanja, kao što su ID3, C4.5, CHAID, CR&T i QUEST. Uglavnom, većina algoritama ove metode koristi tako zvanu „od vrha prema dnu“ (eng. Top-down) metodu pretraživanja. ID3 (eng. Induction of Decision Trees) je jedan od najpoznatijih algoritama koji je razvio J. Ross Quinlan te je bio temelj za proširenje C4.5 algoritma. (Kantardžić, 2011., str. 172.)

ID3 algoritam započinje sa svim uzorcima za treniranje u početnom čvoru stabla. Odabran je neki atribut kako bi podijelio te uzorke. Za svaku vrijednost atributa kreira se grana, te je odgovarajući pod skup uzoraka koji imaju vrijednost atributa specificiranu od grane, pomaknut na novo kreirani čvor dijete (eng. Child node). Algoritam se primjenjuje rekurzivno na svaki čvor dijete sve dok svi uzorci na čvoru ne pripadnu nekoj klasi. Svaki put do lista (eng. Leaf) u stablu odlučivanja predstavlja klasifikacijsko pravilo. Odabir atributa kod ID3 i C4.5 algoritma bazira se na minimiziranju mjere entropije informacije koja se primjenjuje na primjerima u čvoru. Pristup baziran na entropiji informacije predstavlja minimiziranje broja testova koji će omogućiti uzorku klasifikaciju u bazi podataka. (Kantardžić, 2011., str. 172.)

3.4.5 J48

J48 je algoritam klasifikacijskog stabla odlučivanja koji se temelji na C4.5 algoritmu. U vrijeme kada je Weka napravljena najnovija verzija C4.5 algoritma je bila C4.8, pošto je Weka napisana u Javi, a algoritam se bazira na spomenutom C4.8, taj algoritam je dobio ime J48 koje je specifično za Weku.

Koristi se Weka, te se primjenjuje J48 algoritam na weather.nominal.arff skupu podataka. Učitavaju se podatci, te se odabire Classify kao metodu i J48 kao algoritam iz datoteke trees. Rezultat je sljedeći:

```
J48 pruned tree
------------------

outlook = sunny
|   humidity = high: no (3.0)
|   humidity = normal: yes (2.0)
outlook = overcast: yes (4.0)
outlook = rainy
|   windy = TRUE: no (2.0)
|   windy = FALSE: yes (3.0)

Number of Leaves  :     5

Size of the tree :      8


Time taken to build model: 0 seconds

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

Correctly Classified Instances           7               50      %
Incorrectly Classified Instances         7               50      %
Kappa statistic                          -0.0426
Mean absolute error                       0.4167
Root mean squared error                   0.5984
Relative absolute error                  87.5    %
Root relative squared error             121.2987 %
Total Number of Instances                14

=== Detailed Accuracy By Class ===

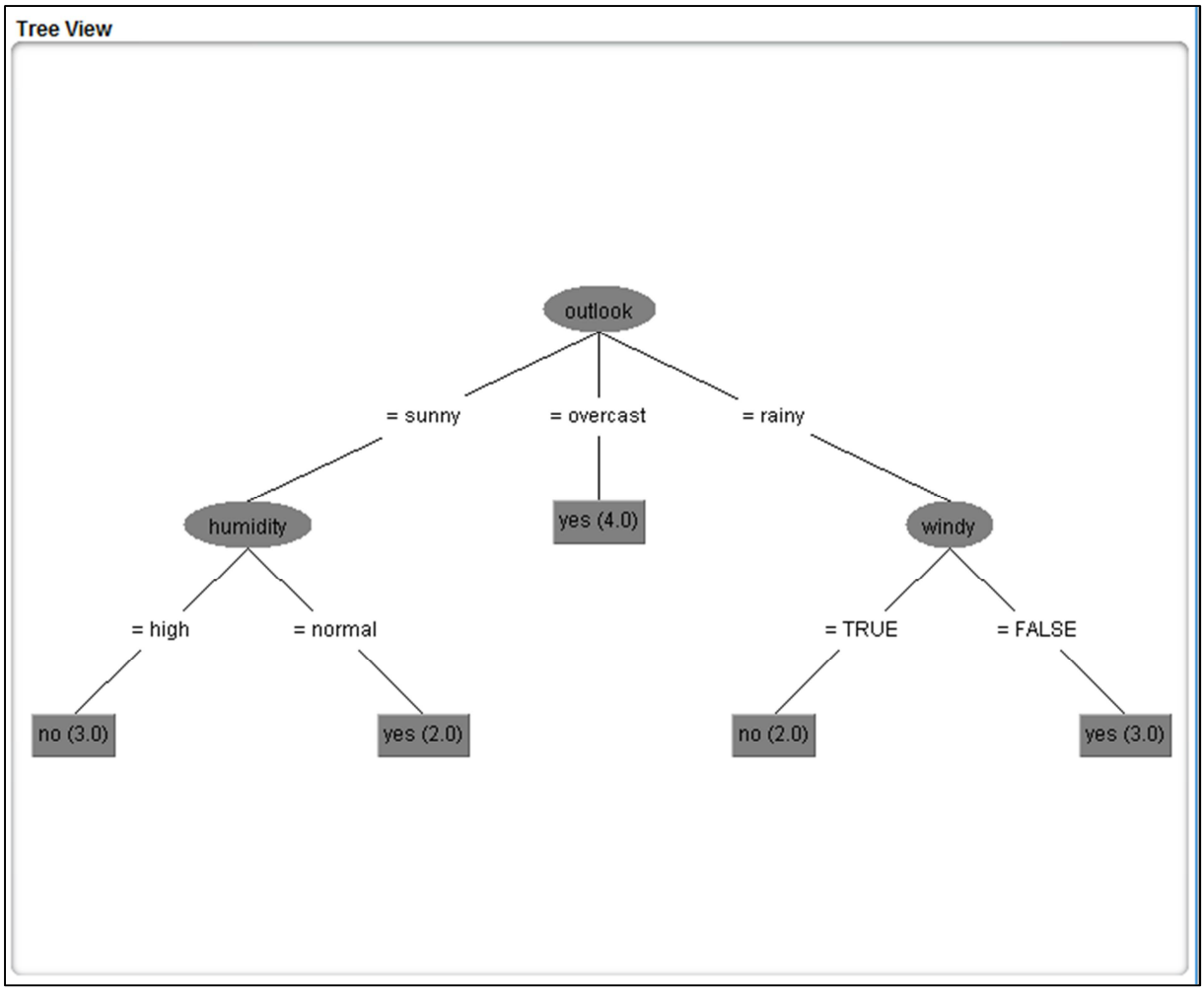
                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0,556    0,600    0,625      0,556    0,588      -0,043   0,633     0,758     yes
                 0,400    0,444    0,333      0,400    0,364      -0,043   0,633     0,457     no
Weighted Avg.    0,500    0,544    0,521      0,500    0,508      -0,043   0,633     0,650

=== Confusion Matrix ===

 a b   <-- classified as
 5 4 | a = yes
 3 2 | b = no
```

Slika 11. Rezultat primjene J48 algoritma nad weather.nominal.arff skupom podataka

U rezultatu se vidi da je u izgrađenom modelu, algoritam izabrao outlook kao početni čvor stabla. Taj čvor se onda grana u humidity, overcast i temperature čvorove koji se onda granaju u odgovarajuće listove (eng. Leaves). Kada se vizualizira ovaj rezultat dobiva se sljedeće stablo:



Slika 12. Vizualizirano stablo J48 algoritma

3.5 Klasteriranje

Klaster analiza ili klasteriranje je grupiranje skupa objekata na takav način da su objekti u istoj skupini (klaster) sličniji međusobno nego onima u ostalim skupinama. To je glavna zadaća eksplorativnog rudarenja po podatcima i zajednička tehnika za statističke analize podataka koja se koristi u mnogim područjima uključujući strojno učenje, prepoznavanje uzoraka, analizu slike, pronalaženje informacija, bioinformatiku, računalnu grafiku i tako dalje. (Witten et al., 2016)

Klaster analiza sama po sebi nije jedan specifičan algoritam, već opći zadatak koji treba riješiti. To se može postići različitim algoritmima koji se značajno razlikuju u njihovom pojmu što čini klaster i kako ih učinkovito pronaći. Koristi se Euklidska ili Manhattan udaljenost za računanje sličnosti između članova populacije nad kojom se

vrši analiza. Grupe se formiraju postupkom dijeljenja skupa podataka, pri čemu se pripadnost grupi definira na temelju značajki sličnih obilježja. Algoritmi za klasteriranje pokušavaju pronaći sličnosti unutar zadane populacije koristeći zadani skup atributa.

Postoji mnogo algoritama za klasteriranje međutim najpoznatiji je K – means algoritam, koji pomoću funkcija za procjenu distance i centroida, u iterativnom postupku kreira klastere, te aglomerativni hijerarhijski algoritam. (Witten et al., 2016)

3.5.1 K-means klasteriranje

Ova metoda funkcionira tako da dijeli osnovnu populaciju na k segmente. Svaki od segmenata sadrži n sličnih elemenata. Na temelju funkcije udaljenosti algoritam procjenjuje sličnost elemenata. Ova metoda se algoritamski može prikazati na sljedeći način: (Witten et al., 2016)

1. Izaberi proizvoljno k segmenata (klastera)
2. Odredi središte za svaki od k segmenata
3. Ponavljaj:
    - Pridruži pomoću funkcije udaljenosti sve elemente populacije njihovim najbližim klasterima (proračun se vrši na temelju centralnih vrijednosti).
    - Izračunaj novu vrijednost središta klastera za svaki klaster pojedinačno kao prosječnu vrijednost objekata sadržanih unutar svakog klastera.
    - Ponavljaj sve dok se mijenjaju vrijednosti središta klastera.

Obično se koristi kriterij kvadratne pogreške (eng. Square-error criterion).

Koristeći Weku primijeniti će se k-means metoda klasteriranja na glass.arff skup podataka, koji je opisan u poglavlju 4.2 Podatci. Učitavaju se podatci, odabire se metoda Cluster, te se zatim odabire metoda SimpleKMeans za klasteriranje. Rezultat je sljedeći:

```
Number of iterations: 13
Within cluster sum of squared errors: 34.13433421599164

Initial starting points (random):

Cluster 0: 1.52152,13.05,3.65,0.87,72.32,0.19,9.85,0,0.17
Cluster 1: 1.51618,13.53,3.55,1.54,72.99,0.39,7.78,0,0

Missing values globally replaced with mean/mode

Final cluster centroids:
                        Cluster#
Attribute    Full Data          0          1
               (214.0)    (162.0)     (52.0)
=============================================
RI              1.5184     1.5181     1.5191
Na             13.4079    13.2811    13.8027
Mg              2.6845     3.4541     0.2871
Al              1.4449     1.3104      1.864
Si             72.6509    72.6122    72.7715
K               0.4971     0.4957     0.5013
Ca               8.957     8.6236     9.9954
Ba               0.175      0.028     0.6333
Fe               0.057     0.0605     0.0462



Time taken to build model (full training data) : 0 seconds

=== Model and evaluation on training set ===

Clustered Instances

0      162 ( 76%)
1       52 ( 24%)


Class attribute: Type
Classes to Clusters:

  0  1  <-- assigned to cluster
 70  0 | build wind float
 65 11 | build wind non-float
 17  0 | vehic wind float
  0  0 | vehic wind non-float
  1 12 | containers
  5  4 | tableware
  4 25 | headlamps

Cluster 0 <-- build wind float
Cluster 1 <-- headlamps

Incorrectly clustered instances :       119.0    55.6075 %
```

Slika 13. Rezultat k-means klasteriranja

U analizi k vrijednost je postavljena na 2 što znači da je željeni broj klastera 2. Način klasteriranja postavljen je na Classes to Clusters, što znači da prvo se ignorira klasni atribut te se generira klasteriranje. Za vrijeme faze testiranja dodjeljuju se klase klasterima na osnovu većinske vrijednosti klasnog atributa unutar svakog klastera.

U rezultatima se vidi da kvadratna pogreška iznosi 34.13, što je relativno dobro s obzirom na to da je poželjno da taj broj bude što manji. Broj iteracija iznosi 13. Prvom klasteru dodijeljeno je 162 instance, odnosno 76%, dok je drugom dodijeljeno 52 instance, odnosno 24%. Broj pogrešno klasteriranih instanci iznosi 119, odnosno 55.6%.

Ako se poveća željeni broj klastera na 5, dobije se da kvadratna pogreška iznosi 23.61, odnosno smanjuje se u odnosu na prethodnu analizu. Broj iteracija iznosi 10. Generirano je 5 klastera. Prvi klaster sadrži 40 instanci, odnosno 19%, drugi klaster sadrži 45 instanci, odnosno 21%, treći klaster sadrži 2 instance, odnosno 1%, četvrti klaster sadrži 24 instance, odnosno 11% i peti klaster sadrži 103 instance, odnosno 48%. Broj pogrešno klasteriranih instanci je 126, odnosno 58.9 %.

3.5.2 EM algoritam

EM (Expectation-Maximization) algoritam je algoritam za maksimiziranje očekivanja. To je iterativna metoda za pronalaženje maksimalnih vjerojatnosti u statističkim modelima, gdje model ovisi o neprimijećenim latentnim varijablama. EM iteracija izmjenjuje se između izvođenja koraka očekivanja (E), koji stvara funkciju za očekivanje log-vjerojatnosti koja se procjenjuje korištenjem trenutne procjene parametra i koraka maksimizacije (M), koji izračunava parametre koji maksimiziraju log-vjerojatnost pronađenu na E koraku. Ove se procjene parametara zatim koriste za određivanje raspodjele latentnih varijabli u sljedećem koraku E. (Witten et al., 2016)

Koristeći Weku primijeniti će se EM metodu na glass.arff skup podataka. Učitavaju se podatci. Odabire se Cluster kao metoda, te se zatim odabire EM kao metoda klasteriranja. Rezultat je sljedeći:

```
=== Model and evaluation on training set ===

Clustered Instances

0       75 ( 35%)
1      139 ( 65%)


Log likelihood: 2.67064


Class attribute: Type
Classes to Clusters:

  0   1  <-- assigned to cluster
  4  66 | build wind float
 20  56 | build wind non-float
  1  16 | vehic wind float
  0   0 | vehic wind non-float
 13   0 | containers
  9   0 | tableware
 28   1 | headlamps

Cluster 0 <-- headlamps
Cluster 1 <-- build wind float

Incorrectly clustered instances :       120.0    56.0748 %
```

Slika 14. Rezultat EM metode klasteriranja

EM metoda sama odabire broj klastera za koji misli da je najbolji, te se u ovome slučaju vidi da je broj klastera 2. Prvi klaster sadrži 75 instanci, odnosno 35%, dok drugi klaster sadrži 139 instanci, odnosno 65%. Broj pogrešno klasteriranih instanci iznosi 120, odnosno 56%.

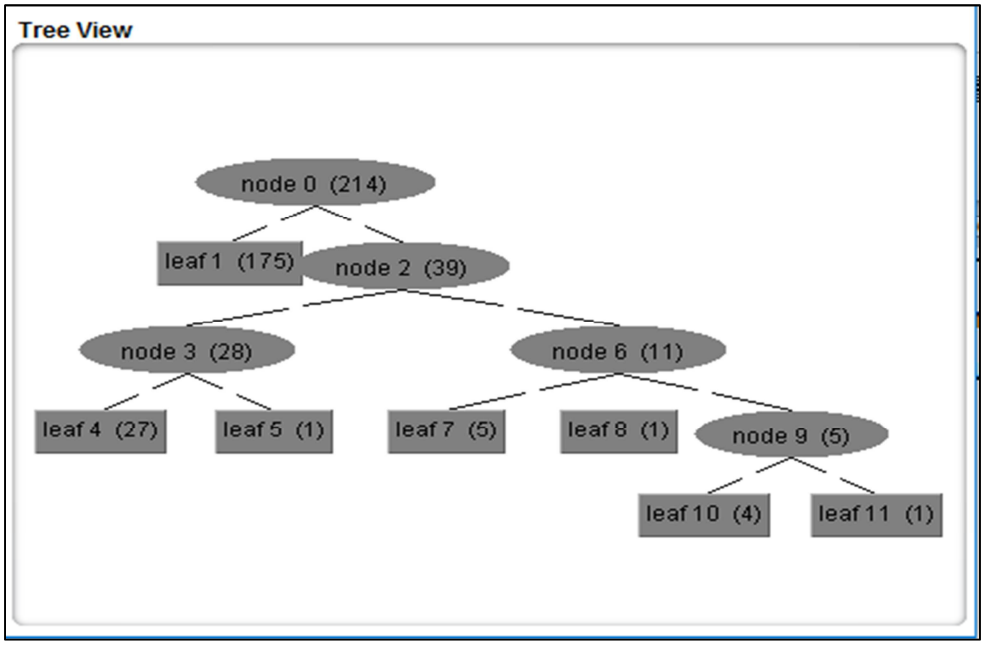
3.5.3 Hijerarhijsko klasteriranje

Hijerarhijsko klasteriranje polazi od grupiranja objekata u stablo klastera. Ova se vrsta klasteriranja klasificira na aglomerativno i divizijsko hijerarhijsko klasteriranje. To ovisi o smjeru particioniranja, koje može biti od dna prema vrhu ili obrnuto. Nakon što se jednom izvrši podjela populacije u klastere, nemoguće je ponavljanje procesa

klasifikacije na istoj razini stabla. To se smatra jednim od glavnih nedostataka hijerarhijskih algoritama za klasteriranje. (Witten et al., 2016)

Aglomerativno hijerarhijsko klasteriranje može se definirati kao klasteriranje metodologijom „od dna prema vrhu“ (eng. Bottom - up), svrstavanjem svakog pojedinačnog objekta u njegov vlastiti klaster. Sljedeći korak se sastoji od stvaranja novih klastera povezujući temeljne klastere u sve veće skupine, sve dok svi elementi u krajnjem koraku ne formiraju zajednički klaster, ili dok se ne ostvari uvjet prekida daljnjeg klasteriranja. Divizijsko klasteriranje se od aglomerativnog razlikuje jedino u smjeru klasteriranja koji je u ovom slučaju „od vrha prema dnu“ (eng.Top-down), pri čemu se temeljni, inicijalni jedinstveni klaster, koji sadrži sve elemente populacije, dijeli u manje klastere sve dok svaki od elemenata ne formira vlastiti klaster, ili dok se ne ispuni zadani uvjet prekida daljnjeg klasteriranja. (Witten et al., 2016)

Koristeći Weku primijeniti će se Cobweb metoda hijerarhijskog klasteriranja na glass.arff skup podataka. Učitavaju se podatci, zatim se odabire Cluster kao metoda te se odabire Cobweb kao metodu klasteriranja. Rezultat je sljedeći:



Slika 15. Rezultat Cobweb metode hijerarhijskog klasteriranja

Prema rezultatima klasterirane instance su 1 sa 175 instanci, odnosno 82%, 3 sa 7 instanci, odnosno 3%, 4 sa 24 instance, odnosno 11%, 7 sa 3 instance, odnosno 1%, 10 sa 4 instance, odnosno 2% i 11 sa 1 instancom, odnosno 0%. Broj pogrešno klasteriranih instanci iznosi 115, odnosno 53%.

4. VALIDACIJA

Validacija je proces procjene uspješnosti rudarskih modela na stvarnim podatcima. Važno je da se provjeri kvaliteta i karakteristika modela rudarstva prije nego što ih se uvede u proizvodno okruženje.

Postoje mnogi pristupi za procjenu kvalitete i karakteristika modela rudarenja po podatcima. Koriste se različite mjere statističke valjanosti kako bi se utvrdilo postoje li problemi u podatcima ili modelu, odvajaju se podatci u setove za treniranje i setove za testiranje kako bi se provjerila točnost predviđanja, pita se poslovne stručnjake da pregledaju rezultate modela kako bi utvrdili otkrivaju li otkriveni obrasci ciljani poslovni scenarij.

Sve su ove metode korisne u metodologiji rudarenja po podatcima i upotrebljavaju se iterativno prilikom izrade, testiranja i poboljšavanja modela za odgovaranje na određeni problem.

Definicije kriterija za provjeru modela podatka rudarenja općenito spadaju u kategorije točnosti, pouzdanosti i korisnosti.

Točnost je mjera koliko dobro model korelira ishod s atributima u podatcima koji su dobiveni. Postoje različite mjere točnosti, ali sve mjere ovise o korištenim podatcima. U stvarnosti, vrijednosti možda nedostaju ili su približne, ili su podatci možda promijenjeni pomoću više procesa. Posebno u fazi istraživanja i razvoja, može se odlučiti prihvatiti određenu količinu pogrešaka u podatcima, osobito ako su podatci u svojim karakteristikama prilično jednolični. Na primjer, model koji predviđa prodaju za određenu trgovinu na temelju prošlih prodaja može biti snažno koreliran i vrlo precizan, čak i ako ta trgovina dosljedno koristi pogrešnu računovodstvenu metodu. Stoga mjerenje točnosti mora biti uravnoteženo procjenom pouzdanosti. (Witten et al., 2016)

Pouzdanost procjenjuje način na koji model rudarenja po podatcima obavlja na različitim skupovima podataka. Model rudarenja pouzdan je ako generira isti tip predviđanja ili pronalazi iste opće vrste uzoraka bez obzira na isporučene testne podatke. Na primjer, model koji generira za trgovinu koja koristi pogrešnu

računovodstvenu metodu ne bi se dobro generalizirala u druge trgovine i stoga ne bi bila pouzdana. (Witten et al., 2016)

Korisnost uključuje različite mjerne podatke koji govore da li model daje korisne informacije. Na primjer, model rudarenja po podatcima koji povezuje lokaciju trgovine s prodajom može biti točan i pouzdan, ali možda neće biti koristan zato što se ne može generalizirati taj rezultat dodavanjem više trgovina na istoj lokaciji. Osim toga, ne odgovara na temeljno poslovno pitanje zašto određena mjesta imaju više prodaje. (Witten et al., 2016)

Neke od tehnika validacije su cross validacija, konfuzijska matrica i split validacija.

4.1 Cross validacija

Cross validacija je tehnika validacije koja procjenjuje kako će se rezultati rudarenja generalizirati na skup podataka. Uglavnom se upotrebljava u postavkama u kojima je cilj predviđanje, a želi se procijeniti koliko je prediktivni model točan u praksi. U problemu predviđanja, modelu se obično daje skup poznatih podataka na kojima se izvodi istraživanje (skup podataka za trening) i skup nepoznatih podataka protiv kojih se model ispituje (skup podataka za testiranje). (Witten et al., 2016)

Jedan krug cross validacije uključuje podjelu uzorka podataka u komplementarne podskupove, provođenje analize na jednom podskupu (validacijski skup ili skup za testiranje). Da bi se smanjila varijabilnost, izvršava se više krugova cross validacije pomoću različitih particija, a rezultati validacije se kombiniraju, odnosno nalazi se prosječna vrijednost, tijekom krugova kako bi se procijenio konačni prediktivni model. Na primjer, standardni broj krugova cross validacije u Weki je deset.

Jedan od glavnih razloga za upotrebu cross validacije umjesto korištenja konvencionalne provjere valjanosti (na primjer, particioniranje skupova podataka u dva seta od 70% za trening i 30% za test) je taj što nema dovoljno podataka za podjelu u posebne skupove za trening i testiranje bez gubljenja značajnih mogućnosti modeliranja ili testiranja. U tim slučajevima, načina da se pravilno procjenjuje izvedba predviđanja modela jest korištenje cross validacije.Ukratko, cross validacija kombinira prosječne vrijednosti pogreški predviđanja kako bi izvela precizniji model predviđanja.

4.2 Konfuzijska matrica

Konfuzijska matrica je tablica koja se često koristi za opisivanje izvedbe klasifikacijskog modela na skupu testnih podataka za kojeg su poznate stvarne vrijednosti.

U konfuzijskoj matrici su prikazane sljedeće vrijednosti:

- Pravi pozitivni (eng. True positives (TP)) : to su slučajevi u kojima je predviđeno da će se nešto dogoditi i da se to dogodilo
- Pravi negativni (eng. True negatives (TN)): to su slučajevi u kojima je predviđeno da se nešto neće dogoditi i da se to nije dogodilo
- Lažno pozitivni (eng. False positives (FP)): to su slučajevi u kojima je predviđeno da će se nešto dogoditi, ali se to nije dogodilo
- Lažno negativni (eng. False negatives (FN)): to su slučajevi u kojima je predviđeno da se nešto neće dogoditi, ali se to dogodilo

Na primjer, na slici 16 prikaza je konfuzijska matrica koja je dobivena provedbom Naive Bayes klasifikatora nad weather.nominal.arff skupom podataka je sljedeća:

```
=== Confusion Matrix ===

 a b   <-- classified as
 7 2 | a = yes
 4 1 | b = no
```

Slika 16 Primjer konfuzijske matrice

Na temelju matrice se zaključuje da je 8 instanci klasificirano točno, dok ih je 6 klasificirano netočno. Od tih 8 točnih instanci 7 ih se dogodilo kao što je to predviđeno, odnosno ta vrijednost predstavlja true positive slučaj, dok se jedna instanca nije dogodila kao što je to predviđeno, odnosno ta vrijednost predstavlja True negative slučaj. Od 6 netočnih instanci, 2 instance se nisu dogodile iako je predviđeno da će se dogoditi, odnosno one predstavljaju False positive slučaj, dok su se 4 instance dogodile iako je predviđeno da se neće dogoditi, odnosno one predstavljaju False negative slučaj.

4.3 Split validacija

Split validacija je validacija u kojoj se podatci odvajaju u skupove treninga i testiranja. Većinom kada se podatci odvajaju, većina podataka se koristi za trening, a manji dio za testiranje (na primjer, 70% za trening, 30% za testiranje). Korištenjem sličnih podataka za trening i testiranje može se smanjiti razlika u podatcima te se mogu bolje razumjeti značajke modela. Nakon što je model obrađen korištenjem trening skupa, testira se model predviđanja u odnosu na skup za testiranje. Budući da podatci u skupu za testiranje već sadrže poznate vrijednosti za atribut koji se želi predvidjeti, lako je utvrditi da li je pogodak modela točan. (Witten et al., 2016)

5. USPOREDBA

U ovome poglavlju se nalazi tablica u kojoj se ukratko opisuju i uspoređuju algoritmi korišteni u ovome radu.

Tablica 2 Opis i usporedba korištenih algoritama i metoda

Algoritam	Opis
Linearna regresija	-koristi se u regresiji -predviđa broj -ne koristiti za nominalne vrijednosti -mjeri razliku između predviđenih i očekivanih vrijednosti
Naive Bayes klasifikator	-koristi se za klasifikaciju -atributi su neovisni -osnovni koncept počiva na vjerojatnosti
k-nearest neighbors (k-NN)	-koristi se za klasifikaciju i regresiju -učenje temeljeno na instancama -odabire glavnu klasu s najvećim brojem glasova među susjedima (parametar k)
OneR	-koristi se za klasifikaciju -odabire pravilo za svaki prediktor -odabire pravilo s najmanjom ukupnom pogreškom kao glavno pravilo
J48	-stablo odlučivanja -koristi se za klasifikaciju -metoda od vrha prema dnu -generira čvorove i listove koji su povezani granama -svaki put do lista predstavlja klasifikacijsko pravilo
k-means	-metoda klasteriranja -dijeli populaciju na k segmenata -svaki segment sadrži n elemenata koji su slični
EM algoritam	-metoda klasteriranja -temelji se na vjerojatnosti -koristi maksimiziranje očekivanja

Hijerarhijsko klasteriranje	-metoda klasteriranja -kao rezultat daje stablo -može biti od dna prema vrhu ili od vrha prema dnu

U tablici 2 se nalazi popis metoda koje su korištene u radu. Metode su sljedeće; linearna regresija, Naive Bayes klasifikator, k-nearest neighbors (k-NN), OnerR, J48, k-means, EM algoritam i hijerarhijsko klasteriranja.

Za pronalaženje numeričkih vrijednosti, potrebno je koristiti linearnu regresiju, dok se za pronalaženje nominalnih vrijednosti mogu koristiti neke od metoda klasifikacije kao što su, Naive Bayes klasifikator, k-nearest neighbors i OneR. J48 se također primjenjuje u klasifikaciji, ali ova metoda spada u metode stabala odlučivanja.

K-means, EM algoritam i hijerarhijsko klasteriranje se koriste kada je cilj grupiranje skupa objekata na takav način da su objekti u istoj skupini (klaster) sličniji međusobno nego onima u ostalim skupinama.

Svaka od ovih metoda ima svoje specifičnosti koje ih razlikuju od drugih metoda te ih čine primjenjivijim za određene skupove podataka. Specifičnosti su ukratko prikazane u tablici 2.

6. ZAKLJUČAK

Rudarenje po podatcima se koristi kako bi se iz velike količine podataka došlo do zanimljivih i potrebnih informacija ili znanja. Kako bi se moglo uspješno rudariti po podatcima na raspolaganju se nalaze brojne metode i algoritmi pomoću kojih se dobivaju traženi rezultati. No prije nego što se počne rudariti po podatcima potrebno je pripremiti podatke. Vrlo je važno imati dobro pripremljene podatke kako bi proces rudarenja prošao glatko i bez većih problem. Postoji cijeli proces za pripremu podataka, kao i metode pomoću kojih se mogu transformirati, reducirati ili upotpuniti podatci kako bi željeni skup podataka dao što bolje rezultate.

U ovome radu su navedene i isprobane neke od metoda i algoritama. Koristili su se sljedeći algoritmi; linearna regresija, Naive Bayes klasifikator, k-nearest neighbor, OneR, J48, k-means, EM algoritam i hijerarhijsko klasteriranje. Kroz upotrebu tih algoritama može se zaključiti da svaki od njih ima svoje specifične primjene u kojima daje bolje rezultate nego drugi algoritmi. Na primjer, linearna regresija će se koristiti kada se želi dobiti neka numerička vrijednost, na primjer vrijednosti kuće ili vrijednost nekakvo cpu-a, dok će se algoritme poput Naive Bayesa, k-nearest neighbora i OneR-a koristiti kada se želi istražiti nekakva nominalna vrijednost. Doduše, svaki od tih neće dati isti rezultat na istome setu podataka, kao što se može vidjeti kroz primjenu tih algoritama na weather.nominal.arff set podataka, već se mora pronaći onaj koji daje najbolje rezultate.

Za prikaz metoda rudarenja podataka u ovome radu je korišten alat Weka. Weka je jedan od mnogih besplatnih alata koji su dostupni svima. Postoje i mnogi drugi alati, komercijalni i besplatni, i prema istraživanjima prikazanim u radu može se vidjeti da je najpopularniji alat trenutno R. Korištenje alata uvelike ubrzava proces rudarenja po podatcima. Također, na kraju rada je prezentirana tablica u kojoj se nalazi kratka usporedba i opis algoritama koji su korišteni u radu.

7. LITERATURA

Knjige:

1. KANTARDŽIĆ, M. (2011) *Data Mining concepts, models, methods, and algorithms*. John WIley & Sons, Inc.: Hoboken, New Jersey
2. WITTEN, I. H. et al; (2016) *Data Mining: Practical Machine Learning Tools and Techniques*. 4th Edition Morgan Kaufmann Publishers
3. FAYYAD, U. et al; (1996) *From Data Mining to Knowledge Discovery in Databases* [Online] 17 (3) str.37-54. Dostupno na: https://www.aaai.org/ [Pristupljeno: 23. Kolovoza 2017.]

Popis slika:

Slika 1 Proces rudarenja po podatcima (Kantardžić, 2011., str. 9.) 5
Slika 2. Rast internet hostova od 1994-2017 8
Slika 3. Primjer kategorijske vrijednosti (Kantardžić, 2011., str. 27.) 11
Slika 4. Najkorišteniji alati za rudarenje podatcima 2013. godine 20
Slika 5. Prikaz grafičkog sučelja u Weki 21
Slika 6 Rezultati algoritma regresije nad cpu.arff skupom podataka 23
Slika 7. Rezultati Naive Bayes klasifikatora nad weather.nominal.arff skupom podataka 26
Slika 8. Rezultati k-NN algoritma sa parametrom k=1 28
Slika 9. Rezultat k-NN algoritma sa parametrom k=3 28
Slika 10. Rezultat OneR algoritma nad weather.nominal.arff skupom podataka 30
Slika 11. Rezultat primjene J48 algoritma nad weather.nominal.arff skupom podataka 32
Slika 12. Vizualizirano stablo J48 algoritma 33
Slika 13. Rezultat k-means klasteriranja 35
Slika 14. Rezultat EM metode klasteriranja 37
Slika 15. Rezultat Cobweb metode hijerarhijskog klasteriranja 38
Slika 16 Primjer konfuzijske matrice 41

Popis tablica:

Tablica 1. Tipovi varijabli sa primjerima 12
Tablica 2 Opis i usporedba korištenih algoritama i metoda 43

8. SAŽETAK

Cilj rada jest opisati i objasniti metode i algoritme za rudarenje po podatcima, kao i proces pripreme i obrade podataka. Rudarenje po podatcima može se definirati kao proces otkrivanja znanja pomoću raznih metoda koje se primjenjuju na skupove podataka. Za potrebe rada koristi se alat Weka za primjenu metoda i algoritama. Metode i algoritmi koji se koriste u radu su sljedeći; linearna regresija, Naive Bayes klasifikator, k-nearest neighbor, OneR, J48, k-means, EM algoritam i hijerarhijsko klasteriranje. Rezultat usporedbe je prikazan tablicom u kojoj se nalaze kratke specifikacije metoda, te koja može služiti kao vodič pri odabiru metoda za rudarenja.

ABSTRACT

The aim of this paper is to describe and explain the data mining methods and algorithms as well as the process of preparation and processing of data. Data mining can be defined as a process of discovering knowledge by using the various methods we apply to data sets. For the purposes of the work, the Weka tool will be used for the application of methods and algorithms. The methods and algorithms to be used in the paper are as follows; Linear regression, Naive Bayes classifier, k-nearest neighbor, OneR, J48, k-means, EM algorithm and hierarchical clustering. The result of the comparison is shown in the table with short specification of the methods, which can serve as a guide when selecting the data mining method.

Ključne riječi:

Rudarenje po podatcima, Weka, algoritmi za rudarenje, metode za rudarenje, podatci, priprema podataka