MATEMATIČKI FAKULTET
UNIVERZITET U BEOGRADU

Istraživanje pravila pridruživanja nad bazom podataka mikroorganizama

Master rad

Autor:
Dragana Lazić

Mentor:
Dr Gordana Pavlović Lažetić

U Beogradu, 2014. godine

Matematički fakultet - Univerzitet u Beogradu

Master rad

Autor:	Dragana Lazić
Naslov:	Istraživanje pravila pridruživanja nad bazom podataka mikroorganizama
Mentor:	dr Gordana Pavlović Lažetić
Članovi komisije:	dr Nenad Mitić
	dr Miloš Beljanski
Datum:	02.10.2014

Sadržaj

Spisak slika

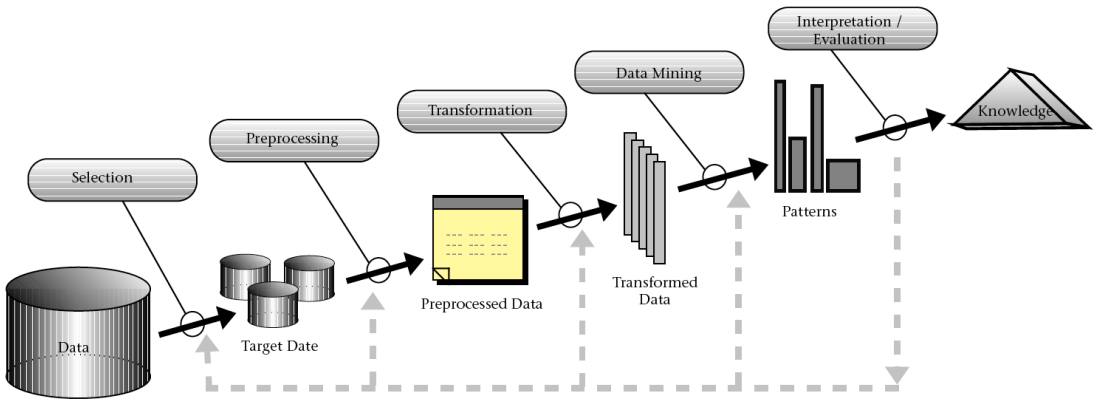
Spisak tabela

1 Uvod

1.1 Istraživanja podataka - Pravila pridruživanja

U informacionom dobu u kome živimo svakodnevno se prikupljaju ogromne količine podataka. Bilo da se radi o običnim transakcijama prilikom kupovine, medicinskim pregledima ili naučnim merenjima i istraživanjima, sa razvojem računara i njihovom širokim primenom, svaka obavljena transakcija se beleži u elektronskom obliku. Raspoložive količine podataka su toliko velike da tradicionalne statističke metode ne mogu da budu primenjene nad njima, ne samo zbog velikog broja entiteta već i velikog broja pridruženih atributa. Nastaje jaz između nove generacije metoda za prikupljanja i skladištenja podataka i tradicionalnih metoda za njihovu obradu. Javlja se potreba za novima alatima koji će omogućiti izdvajanje korisnih informacija (znanja) iz sirovih podataka neupotrebljivih za čoveka.

Interdisciplinarna oblast koja od devedesetih godina prošlog veka pa do danas pokušava da prevaziđe ovaj jaz je ***otkrivanje znanja u bazama podataka*** (eng. Knowledge Discovery in Database KDD). Otkrivanje znanja u bazama podataka je netrivijalan proces identifikovanja novih, validnih, korisnih i na kraju razumljivih obrazaca iz podataka. ***Istraživanje podataka*** (eng. Data Mining) predstavlja skup metoda i tehnika kojima se podacima daje smisao. Vrši se preslikavanje sirovih podataka koji su obično preobimni za lako razumevanje u kompraktniju i apstraktniju formu u vidu kratkog izveštaja, aproksimacije procesa koji generiše podatke ili modela kojim se mogu predvideti vrednosti podataka u nekim budućim slučajevima.



Slika 1: Koraci u otkrivanju znanja u bazi podataka

Iako se često smatra sinonimom, istraživanje podataka je samo jedan korak u otkrivanju znanja u bazi podataka. Kao što se vidi na slici (Slika 1.), ovom koraku prethodi

izdvajanje validnih podataka za domen istraživanja iz nekog šireg skupa podataka čija primarna svrha nije istraživanje (to mogu biti npr. log datoteke neke poslovne aplikacije), kao i transformacija podataka u oblik pogodan za primenu određenog algoritma radi ekstrakcije znanja. Data mining je termin nastao iz analogije između traženja znanja u ogromnoj količini podataka sa traženjem zlata u rudniku.

Algoritmi istraživanja podataka se na osnovu toga da li predviđaju vrednost za neki konkretan atribut ili samo otkrivaju obrasce koji važe na skupu podataka dele na prediktivne i deskriptivne. Algoritam ***pravila pridruživanja*** (eng. assoociation analysis) je deskriptivni algoritam kojim se otkrivaju korelacije među podacima. Uvodi ga Rakesh Agrawal 1993. godine na domenu kupovnih transakcija (eng. market basket data)[7]. Osim u marketinške svrhe algoritam se primenjuje i u drugim domenima kao što su: bioinformatika, medicinska dijagnostika, istraživanje veba i druga naučna istraživanja. Postoje različite implementacije ove metode. Neke od njih su Apriori, FP rast, Predictive Apriori, Tertius i sl.

1.2 Podaci

Istraživanje započinje obezbeđivanjem adekvatnih podataka. Pod adekvatnim podacima podrazumevaju se oni koji verno oslikavaju realan svet, odnosno domen istraživanja. Elemente skupa podataka obično nazivamo slogovima. Svaki slog oslikava jedan entitet iz domena istraživanja. Slogovima može biti pridružen veći broj atributa. Atributi odgovaraju svojstvima, odnosno karakteristikama entiteta iz realnog sveta. Postoje dve podele atributa:

1. Prema operacijama koje mogu biti primenjene nad njihovim vrednostima
2. Prema broju vrednosti

Tip atributa		Opis	Primer
Kategorički	Nominalni	Sadrže dovoljno informacija da bi se razlikovali među sobom ($=, \neq$)	boja očiju, pol, JMBG
	Ordinalni	Sadrze dovoljno informacija da bi se poredili ($<$,$>$)	razredi, brojevi ulica...
Numerički	Intervalni	Sledeće operacije se mogu vršiti (+,-)	datumi, temperature...
	Odnosni	Sledeće operacije se mogu vršiti (*,/)	masa, dužina...

Tabela 1: Podela podataka prema operacijama koje mogu biti primenjene nad njima

Definicija atributa prema operacijama koje mogu biti primenjene nad njima je kumulativna što znači da svaki atribut definisan u tabeli (Tabela 1) poseduje svojstva i svih prethodno definisanih atributa u njoj.

Tipovi atributa prema drugoj podeli mogu biti:

- Diskretni - imaju konačan broj vrednosti (npr. kategorički), obično se predstavljaju celobrojnim (eng. integer) promenljivama, mada u slučaju binarnih atributa (diskretni atributi koji imaju samo dve vrednosti) mogu se predstaviti i logičkom (eng. boolean) promenljivom
- Kontinualni - predstavljaju se realnim promenljivama (eng. float, double)

Skup podataka se u zavisnosti od toga da li svi slogovi imaju jednak broj atributa ili ne, dele na matrične i transakcione. Matrični podaci se obično smeštaju u obične tekstualne datoteke ili baze podataka. Smeštanjem u baze pretraga podataka je jednostavnija.

Za istraživanje se obično koriste podaci koji su ranije prikupljeni u druge svrhe. Pri izračunavanju (merenju), prikupljanju i unošenju podataka može doći do grešaka. Greške koje se mogu javiti su sledeće:

- Duplirani podaci
- Nedostajuće vrednosti
- Netačne vrednosti
- Elementi van granica

Zbog toga istraživanje obično počinje "čišćenjem" podataka. Pod čišćenjem se podrazumeva otkrivanje grešaka, njihovo ispravljanje ukoliko je to moguće ili uklanjanje loših slogova. Kako nemamo kontrolu nad prikupljanjem podataka, oni se obično nalaze u formatu nepodesnom za neposrednu primenu algoritma istraživanja. U fazi preprocesiranja, podaci odnosno vrednosti njihovih atributa konvertuju se u format razumljiv algoritmu koji se koristi u istraživanju.

1.3 Genomske karakteristike mikroorganizama

Genomika (engl. Genomics) je grana savremene biologije koja se bavi analizom strukture i funkcije genoma (celokupnog sadržaja DNK unutar pojedinačne ćelije nekog organizma). Smatra se da era genomike počinje 1995. godine kada je sekvencioniran prvi bakterijski genom - *Haemophilus influenzae* [15]. Razvojem novih tehnologija vreme potrebno za sekvencioniranje organizma se znatno smanjuje tako da vremenom veliki broj sekvencioniranih genoma postaje dostupan za razna uporedna istraživanja.

Rezultati sekvencioniranja genoma i izračunavanja genomskih karakteristika se smeštaju u elektronskom obliku u strukture podataka pogodne za računarsku obradu. S druge strane, informacije o fenotipskim svojstvima se i dalje uglavnom nalaze u raznim naučnim radovima i knjigama što ih čini nepodesnim za automatsku obradu. Genotip predstavlja celokupnu gensku konstrukciju nekog organizma dok fenotip (gr. phainein - pokazati + typos - tip, vrsta) predstavlja skup svih osobina jednog organizma nastalih zajedničkim delovanjem genotipa i uslova sredineu kojima se organizam razvija (genotip + sredina = fenotip).

Iako između genotipskih i fenotipskih svojstava organizama postoji veliki jaz u dostupnosti za automatsku obradu, izvršen je veliki broj istraživanja kojima se otkrivaju skrivena pravila među ovim svojstvima.

Genom prokariota (prokariotske (grč. pro - pre, karyon - jedro ili jezgro) ćelije čiji je genetički materijal organizovan kao nukleid i nije membranom odvojen od ostatka ćelije) je relativno mali, svega nekoliko miliona baznih parova. Čini ga jedan ili više najčešće kružnh lanaca molekula DNA i eventualni plazmidi. Njihova široka rasprostranjenost u svim staništima u kojima postoji život objašnjena je malim dimenzijama ovih organizama zbog čega ih vetar i voda lako raznose, ali i metaboličkom raznolikošću i fleksibilnošću [6]. Bakterije naseljavaju vrele izvore, kisela vulkanska jezera (niska pH vrednost), anaerobna staništa, pustinjsko zemljište i druga nutritivno siromašna staništa kao i staništa izložena radioaktivnom zračenju i temperaturama višim od 100°C. Još uvek nisu sa sigurnošću utvrđeni geni (gen je molekulska (fizička i funkcionalna) jedinica nasleđivanja svih organizama) niti genomske karakteristike koje obezbeđuju rezistentnost prokariotskih organizama na ekstremne uslove sredine koju naseljavaju.

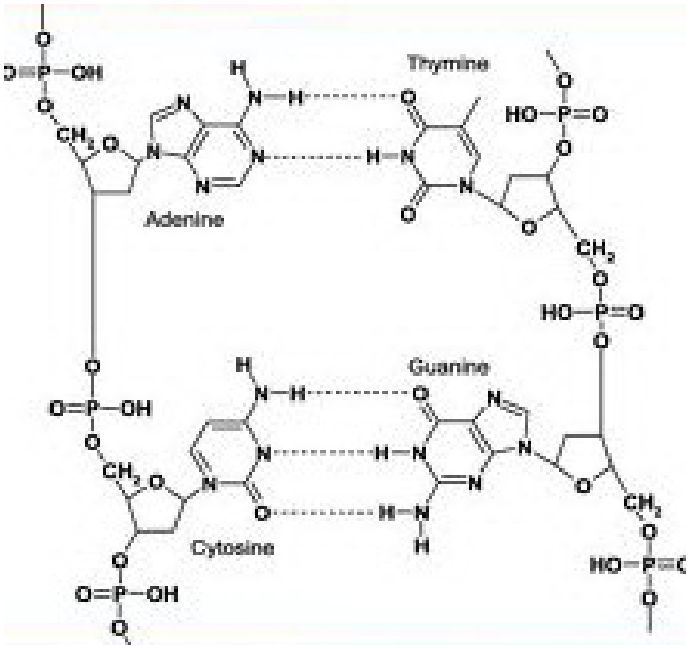
Otkrivanje povezanosti među specifičnim karakteristikama organizama (kao što su GC% (odnos baza koje čine lanac DNA), broj gena i proteina, procenat kodirajuće i nekodirajuće sekvence i sl.) omogućuje bolje razumevanje evolucije, ali takođe može omogućiti i predviđanje nekih mogućnosti [2].

Cilj ovog rada je da potvrdi ranije postavljene hipoteze o korelaciji među genomskim karakteristikama prokariotskih organizama ali i da otkrije neke nove pravilnosti koje važe među njima. Genomske karakteristike koje će biti uzete u razmatranje su sledeće:

Veličina genoma. Veličina genoma predstavlja dužinu celokupnog DNA lanca (lanaca) odnosno celukupan broj nukleotida. Merna jedinica za dužinu DNA odnosno genoma

jeste MB - milion baznih parova (eng. megabases - millions of base pairs). Iako složenije od virusa i jednostavnije od eukariota, među raspodelama dužina ovih organizama ne postoje velike razlike, tako na primer virus *mimivirus* ima dužinu genoma oko 1Mb što ga čini dužim od mnogih bakterisjkih genoma. S druge strane, eukariotski organizam *Encephalitozoon cuniculi* npr. je znatno manji od mnogih bakterijskih genoma [4]. Kraći genomi obično odgovaraju parazitskim bakterijama, odnosno bakterijama koje naseljavaju druge domaćine ili žive u simbiozi sa drugim organizmima. Genomi ovih organizama su kraći jer ne poseduju gene odgovorne za sintezu neophodnih proteina, zbog čega moraju da ulaze u simbioze ili naseljavaju druge domaćine kako bi se snabdevali neophodnim proteinima koje sami ne mogu da sintetišu.

Guanin-Citozin sastav (GC%). Šargafovim pravilom iz 1951. godine je utvrđen molarni odnos azotnih baza u molekulu DNA. Količina komplementarnih purinskih i pirimidinskih baza je jednaka, odnosno broj guanina jednak je broju citozina (isto važi i za adenin i timin) [5]. S druge strane, odnos parova guanin-citozin i adenin-timin varira od vrste do vrste organizma, ali i u okviru segmenata istog genoma. Azotne baze adenin i timin spojene su sa dve vodonične veze, a guanin i citozin sa tri (Slika 2.).



Slika 2: Hemijska struktura molekula DNA

Činjenica da su azotne baze guanin i citozin povezane jednom vodoničnom vezom više utiče na sledeće:

- Za izgradnju ovih veza potrebna je veća količina energije, što znači da organizmi sa većim GC% troše(proizvode) više energije. Zbog toga organizmi sa većim GC% obično žive kao slobodni organizmi dok organizmi bogatiji AT parovima često žive u domaćinima i drugim nutritivno siromašnim staništima.

- Dodatna vodonična veza povećava termostabilnost, delovi DNA lanca bogatiji guaninom i citozinom se sporije denaturališu tako da organizmi sa većim GC% podnose više temperature.

Procenat kodirajuće i nekodirajuće sekvence Pod nekodirajućom sekvencom podrazumevaju se oni delovi DNA lanca koji ne kodiraju proteine. Ranije su nekodirajuće sekvence nazivane genetičkim otpadom (eng Junk DNA) jer se smatralo da ovi delovi nemaju nikakvu biološku funkciju. Međutim, danas se zna da delovi nekodirajućih sekvenci pored ribozomalne i transportne RNA kodiraju i male (kratke) i duhačke RNA koje imaju regulatornu ulogu. Zanimljivo je da prokariotski organizmi imaju manje od 15% nekodirajuće sekvence dok humani organizmi imaju i preko 90 % nekodirajuće sekvence.

Broj proteina Proteini su veoma važni organski molekuli (grč. proteus - najvažniji). Imaju nekoliko uloga u organizmu: gradivnu (npr. kolagen), katalitičku (enzimi), regulatornu (hormoni), zaštitnu (anti tela) itd.

Broj gena Gen je osnovna jedinica nasleđivanja, predstavlja deo DNA lanca koji nosi informaciju o sintezi bilo koje RNA. Zanimljivo je da složenost organizma nije uvek u korelaciji sa brojem gena. Primer za to je najčešće korišćen eukariotski organizam, organizam u genetskom istraživanju - vinska mušica (Drosophila melanogaster) sa svega oko 13.000 gena.

Broj RNA Molekuli RNA nastaju u procesu transkripcije i predstavljaju kopije delova DNA lanca. Odlikuje ih velika raznolikost funkcija u organizmu. Pored tri osnovna tipa čija je funkcija poznata (informaciona, transportna i ribozomalna) postoje i tzv. mali (kratki) i dugački molekuli RNA čija funkcija nije do kraja poznata i smatra se da pored ostalog učestvuju u bakterijskom imunom sistemu tj. u prepoznavanju stranih nukleinskih kiselina.

2 Algoritmi pravila pridruživanja

2.1 Osnovni pojmovi

Da bi se na nekom skupu podataka primenio Apriori algoritam, podaci moraju zadovoljavati odgovarajući format. Entiteti tj. slogovi (redovi u bazi) se često označavaju transakcijama jer su osnovni pojmovi istraživanja pravila pridruživanja definisani na problemu potrošačke korpe. Atribute odnosno svojstva entiteta nazivamo stavkama ili artiklima. Predstavljanje podataka binarnom reprezentacijom znači da ukoliko entitet poseduje neko svojstvo, vrednost tog atrributa je 1, a 0 inače. Uvode se sledeći pojmovi:

Pojam	*Opis*	*Definicija*
I	Skup svih stavki	$I = \{i_1 ... i_d\}$
T	Skup svih transakcija	$T = \{t_1 ... t_N\}$
t_j	transakcija	$t_j \in T \wedge t_j \subseteq I$
K-itemset	Skup od k stavki	$\{i_1 ... i_k\}$
$\sigma(X)$	Podrška skupa stavki	$\sigma(X) = \lvert\{t_j \mid X \subseteq t_j, t_j \in T\}\rvert$
$[X \longrightarrow Y]$	Pravilo pridruživanja	Pravilo pridruživanja je implikacija oblika $X \longrightarrow Y$ gde su X i Y disjunktni skupovi stavki, $X \cap Y = \emptyset$. Jačina pravila se meri podrškom i poverenjem pravila.
$s[X \longrightarrow Y]$	Podrška pravila	$s[X \longrightarrow Y] = \frac{\sigma(X \cup Y)}{N}$,određuje koliko se često pravilo javlja na datom skupu podataka. Pravila sa malom podrškom nisu zanimljiva iz poslovne perspektive, neprofitabilna su.
$c[X \longrightarrow Y]$	Poverenje pravila	$c[X \longrightarrow Y] = \frac{\sigma(X \cup Y)}{\sigma(X)}$, pouzdanost zaključka da se Y pojavljuje u transakcijama koje sadrže X, procenu uslovne verovatnoće od Y kada je dato X

Tabela 2: Pravila priduživanja - osnovni pojmovi

Definicija 1 (Istraživanje pravila pridruživanja) *Za dati skup transakcija T nađi sva pravila koja imaju podršku $\geq$ minsup i poverenje $\geq$ minconf, gde su minsup i minconf unapred zadate vrednosti.*

Ako bi se ovaj problem rešavao metodom grube sile tj. nabrajanjem svih mogućih pravila i računanjem podrške i poverenja za svako od njih, postupak bi bio preskup jer ukupan broj pravila koja se mogu dobiti na skupu podataka sa d stavki je [1]:

$$R = 3^d - 2^{d+1} + 1$$

Dokaz

$$\sum_{i=2}^{d} \binom{d}{i} (2^i - 2)$$

$$= \sum_{i=2}^{d} \binom{d}{i} 2^i - \sum_{i=2}^{d} \binom{d}{i} 2$$

$$= \sum_{i=0}^{d} \binom{d}{i} 2^i - \binom{d}{0} 2^0 - \binom{d}{1} 2^1 - 2 \left(\sum_{i=2}^{d} \binom{d}{i} \right)$$

$$= \sum_{i=0}^{d} \binom{d}{i} 2^i 1^{d-i} - 1 - 2d - 2 \left(\sum_{i=0}^{d} \binom{d}{i} - \binom{d}{0} - \binom{d}{1} \right)$$

$$= (2+1)^d - 1 - 2d - 2\left(\sum_{i=0}^{d}\binom{d}{i}1^d 1^{d-i} - 1 - d\right)$$

$$= 3^d - 1 - 2d - 2\left((1+1)^d - 1 - d\right)$$

$$= 3^d - 1 - 2d - 22^d + 2 + 2d$$

$$= 3^d + 2^{d+1} + 1.\blacksquare$$

Imajući u vidu da podrška pravila $X \longrightarrow Y$ zavisi samo od podrške skupa stavki $X \cup Y$ što znači da će skup stavki male podrške dati i pravila male podrške dolazi se do ideje da se problem raščlani na dva zadatka:

1. **Generisanje čestih skupova** - skupove stavki čija je podrška veća od zadate donje granice *minsup* nazivamo čestim (eng. frequent), dok skupove čija podrška ne zadovoljava ovaj uslov nazivamo retkim ili nefrekventnim (eng. unfrequent).
2. **Generisanje pravila** - pronalaženje pravila čije je poverenje veće od zadate donje granice $minconf$ na osnovu čestih skupova stavki dobijenih u prethodnom koraku.

2.2 Apriori algoritam

Prvi korak je i dalje preskup da bi se rešavao metodom grube sile jer moguć broj skupova stavki nad skupom sa d stavki je $2^d - 1$. Ako bi skup podataka imao N transakcija prosečne širine w broj poređenja svakog kandidata sa svakom transakcijom bi bio $O(MNw)$. Moguća su sledeća dva unapređenja prvog koraka:

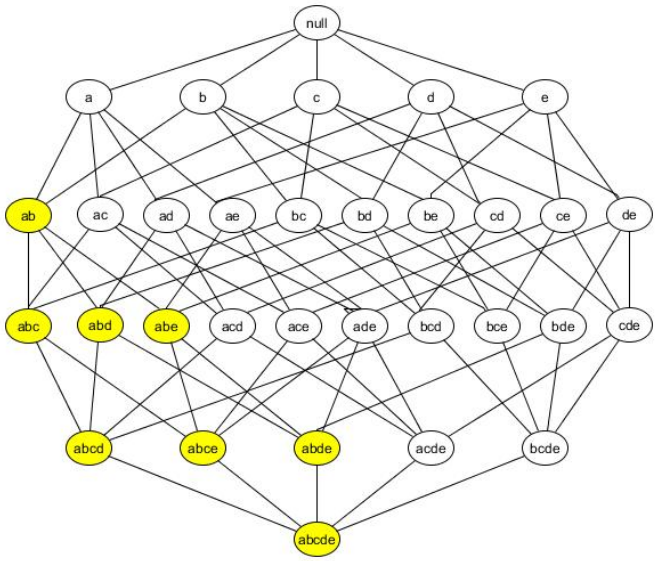
1. **Redukcija broja kandidata skupova stavki** - odbacivanje skupova stavki pre nego što se izračuna njihova podrška
2. **Redukcija broja poređenja** - smeštanjem transakcija u naprednije strukture podataka kao sto je heš drvo

Teorema 1 (Apriori princip) *Ako je neki skup stavki čest onda su svi njegovi podskupovi takođe česti.*

Direktna posledica ove teoreme je da je nadskup nefrekventnnog skupa, takođe, nefrekventan. Ovo svojstvo se naziva ***anti-monotonost***.

$$\forall X, Y \in D : X \subseteq Y \Rightarrow s(Y) \leq s(X)$$

Zahvaljujući ovom svojstvu podrške moguće je bezbedno eliminisati nadskupove nefrekventnog skupa bez računanja njihove podrške. Ovaj postupak se naziva potkresivanje na osnovu podrške (eng. support based pruning) (slika 3).



Slika 3: Potkresivanje na osnovu podrške

2.2.1 Generisanje čestih skupova

Uvode se sledeće oznake :

$$C_k - \text{ skup kandidatskih k-skupovi stavki}$$

$$F_k - \text{skup čestih k-skupova stavki}$$

Generisanje čestih skupova se vrši u nekoliko koraka:

- Inicijalno se prolazi kroz skup svih stavki (jednočlanih skupova stavki) i određuje podrška za svaki od njih. Eliminisanjem onih čija je podrška manja od *minsup* dobija se F_1 skup čestih jednočlanih skupova.
- Iterativno se na osnovu (k-1)-skupova stavki dobijenih u prethodnom koraku kreiraju kandidatski k-skupovi stavki. (korak 5. apriori-gen metoda). Stavke u kandidatskim skupovima podataka seređaju u leksikografskom poretku. Apriori-gen metodom se spajaju parovi (k-1)-skupova stavki koji imaju zajedničkih prvih (k-2) stavki a razlikuju se u poslednjoj stavki:

  $$A = \{a_1...a_{k-1}\}, B = \{b_1...b_{k-1}\} a_i = b_i, i = 1...k-2, a_{k-1} \neq b_{k-1}$$

  Ovom $(F_{k-1} \times F_{k-1})$ metodom se garantuje da se isti kandidat neće generisati više puta kao i potpunost tj. da će se svi kandidati dužine k generisati.
- Dobijenim kandidatima se određuje podrška poređenjem sa transakcijama (koraci 6-10). Radi efikasnosti se kandidatski skupovi stavki kao i skupovi stavki iz transakcije smeštaju u heš drvo koristeći istu heš funkciju tako da se umesto

poređenja jednog kandidatskog skupa stavki sa svim transakcijama vrši samo poređenje skupova stavki iz iste grane drveta.

- Eliminišu se kandidati sa nedovoljnom podrškom
- Algoritam se završava kada se više ne mogu generisati novi česti skupovi

Algoritam generisanja čestih skupova
1.$k = 1$
2. $F_k = \{i \mid i \in I \wedge \sigma(\{i\}) \geq N \times minsup\}$
3.**repeat**
4. $k = k + 1$
5. $C_k = apriori - gen(F_{k-1})$
6. **for each** transaction $t \in T$ do
7. $C_i = subset(C_k, t)$
8. **for each** kandidat $c \in C_i$ do
9. $\sigma(c) = \sigma(c) + 1$
10. **end for**
11. **end for**
12. $F_k = \{c \mid c \in C_k \wedge \sigma(c) \geq N \times minsup\}$
13.**until** $F_k = \emptyset$
14.**result**$= \cup F_k$

Tabela 3: Apriori algoritam generisanja čestih skupova

2.2.2 Generisanje pravila

Pravila se generišu na osnovu prethodno dobijenih čestih skupova. Za računanje poverenja pravila nisu potrebna dodatna izračunavanja jer su podrške za sve česte skupove već izračunate.

Teorema 2 *Ako pravilo $X \Rightarrow Y - X$ nema zadovoljavajuće poverenje tada nijedno pravilo $X' \Rightarrow Y - X'$ gde je $X' \subset X$ nema zadovoljavajuće poverenje.*

Dokaz Iz $X' \subset X$ sledi da je $s(X') \geq s(X)$ tako da je $c(X \Rightarrow Y - X) = \frac{s(X \cup Y - X)}{s(X)} = \frac{s(Y)}{s(X)} \geq \frac{s(Y)}{s(X')} = \frac{s(X' \cup Y - X')}{s(X')} = c(X' \Rightarrow Y - X')$ ■

Posledica ove teoreme je mogućnost sečenja skupa pravila po poverenju. Zbog ove posledice algoritam se izvršava nivo po nivo strategijom gde se u svakom koraku generišu pravila sa više elemenata sa desne strane na osnovu posledica (glava) prethodno dobijenih jakih pravila.

Algortiam generisanja pravila
1.**for each** čest k-skup stavki $f_k, k \geq 2$ **do**
2. $H_1 = \{i \mid i \in f_k\}$
3. pozovi ap-genrules (f_k, H_1)
4.**end for**

Tabela 4: Apriori - Algoritam generisanja pravila

Metoda ap-genrules (f_k, H_m)
1.$k = \|f_k\|$
2.$m = \|H_m\|$
3. **if** $k > m + 1$ **then**
4. $H_{m+1} = apriiori - gen\,(H_m)$
5. **for each** $h_{m+1} \in H_{m+1}$ **do**
6. $conf = \sigma\,(f_k)\,/\sigma\,(f_k - h_{m+1})$
7. **if** $conf \geq minconf$ **then**
8. Izlaz: pravilo $f_k - h_{m+1} \rightarrow h_{m+1}$
9. **else**
10. ukloni H_{m+1} iz H_{m+1}
11. **end if**
12. **end for**
13. pozovi ap-genrules (f_k, H_{m+1})
14.**end if**

Tabela 5: Apriori-rekurzivna ap-genrules metoda

2.3 Algoritam FP rasta (eng. FP Growth)

Za razliku od *Apriori* algoritma gde se testiraj i generiši strategijom generišu česti skupovi prolaženjem kroz sve transakcije pri svakom izračunavanju podrške skupa, *FP Growth* algoritam koristi kompaktnu strukturu podataka iz koje se direktno čitaju česti skupovi a za čiju konstrukciju su potrebna samo dva prolaska kroz bazu.

2.3.1 Kompaktna struktura podataka - FP stablo

Stablo čestih uzoraka ili FP stablo (eng. Frequent Pattern Tree) je struktura koja se sastoji od:

1. Stabla sa čvorovima. Koreni čvor stabla je obeležen sa null dok deca čvorovi odgovaraju stavkama iz transakcija i obeleženi su njihovim imenima. Decu čvorove čine:
   - brojač - broji transakcija koje dele isti prefiks koji sadrži datu stavku

- pokazivač na čvor sa istom stavkom u susednoj grani; na ovaj način su iste stavke u svim transakcijama tj. granama u stablu povezane u listu

2. Tabele zaglavlja čestih stavki (eng. frequent item header table). Svaki unos ove tabele odgovara jednoj stavki i sastoji se od dva polja. Prvo polje predstavlja ime stavke a drugo pokazivač na prvi čvor u stablu koji odgovara toj stavci. Ovaj pokazivač predstavlja glavu povezane liste za datu stavku.

2.3.2 Smeštanje podataka - konstrukcija FP stabla

Stablo se konstruiše u dva prolaza kroz bazu transakcija:

1. Računaju se podrške za sve stavke, stavke sa podrškom manjom od zadate se odbacuju iz transakcija, a preostale se sortiraju u opadajući poredak prema podrški

2. Iščitavaju se transakcije iz skupa i upisuju u stablo tako da svakoj transakciji odgovara putanja u stablu; putanje koje odgovaraju transakcijama sa istim prefiksom se preklapaju čime se i postiže kompresovanje podataka

Algoritam konstrukcije FP stabla
Ulaz Skup transakcija DB, minimalna podrška minsup **Izlaz** Skup čestih skupova 1.Prolazi se kroz skup transakcija **DB** i za svaku stavku izračunava njen broj pojavljivanja tj. podrška. Česte stavke i njihove podrške smeštaju se u skup F. Sortira se skup F na osnovu podrški u opadajućem poretku i dobija se skup sortiranih čestih stavki - L. Na osnovu skupa L popunjava se tabela zaglavlja čestih stavki; pokazivači pokazuju na null do pojave prvog čvora u stablu koji odgovara toj stavci 2. Kreira se koren FP stabla T i obelezi sa "null". 3.**For each** transakciju *trans* iz **DB do** Eliminiši retke stavke iz transakcije a preostale sortiraj prema listi L. Neka je nova ažurirana transakcija Trans=$[p \mid P]$ gde je p prva stavka a P skup preostalih. **Call** metodu $insert\_tree([p \mid P], null)$ **End for**

Tabela 6: FP rast - Algoritam konstruisanja FP stabla

Insert_tree metoda
Ulaz Putanja $[p \mid P]$, čvor T 1.**if** čvor T ima dete N gde je N.ime_stavke=p.ime_stavke 2. **then** povećaj N.brojač za 1 3. **else** dodaj novi čvor N tako da N.ime_stavke=p.ime_stavke i N.brojač=1 i otac od N neka je T 4.**if** P neprazno **call** insert_tree(P,N)

Tabela 7: FP rast - insert_tree metoda

Što se više putanja preklapa tj. sto više transakcija ima zajedničke prefikse kompresija je veća, međutim, ako bi sve transakcije bile bez zajedničkih prefiksa smeštanje podataka na ovaj način bii bilo skuplje od klasičnog zbog dodatne memorije potrebne za smeštanje pokazivača.

2.3.3 Išč̌itavanje čestih skupova iz stabla

Konstruisano FP stablo ima dva važna svojstva koja omogućuju lako izdvajanje čestih skupova, to su:

Svojstvo povezanosti čvorova (eng. node-link property) Za svaku čestu stavku a_i, svi mogući česti uzorci koji sadrže a_i mogu biti dohvaćeni praćenjem povezane liste pridružene tom čvoru počevši od glave u tabeli čestih uzoraka;

Svojstvo prefiksne putanje (eng. prefix path property) Da bi se dohvatili svi česti skupovi koji sadrže stavku a_i u datoj putanji P potrebno je analizirati samo prefksne podputanje za čvor a_i, takođe, sve brojače duž ove podputanje treba ažurirati tako da pokazuju istu vrednost kao pokazivač čvora a_i;

Na osnovu ovog svojstva definiše se transformisana prefiksna putanja za datu putanju P i čvor a_i. Transformisana prefiksna putanja predstavlja prefiks čvora a_i gde su brojači ažurirani tako da sadrže istu vrednost kao i brojač a_i. Skup svih transformisanih putanja za čvor a_i i sve putanje P koje ga sadrze naziva se **kondicionalna baza uzoraka**. Drvo konstruisano na osnovu ove baze naziva se kondicionalno drvo i obeležava sa $FPtree \mid a_i$.

Lema 1 *Neka je α skup stavki u DB, i neka je B kondicionalna baza od α, i β jedan skup stavki iz B, tada je podrška $\alpha \cup \beta$ u DB jednaka podršci od β u B.*

Lema 2 *Neka je α skup stavki u DB, i neka je B kondicionalna baza od α, i β jedan skup stavki iz B, tada je $\alpha \cup \beta$ čest skup u DB ako i samo ako je β čest skup u B.*

Lema 3 *Ako FP stablo T ima samo jednu granu tada se svi česti skupovi takvog stabla mogu dobiti nabrajanjem svih mogućih podputanja od putanje P, gde je podrška ovako dobijenog skupa (podputanje) jednaka minimalnoj podršci čvorova u podputanji.*

Algoritam pretrage FP stabla
Ulaz FP stablo fp_tree i minimalna podrška minsup
Izlaz Skup čestih skupova
1.Pozovi metodu $fp\_growth(fp\_tree, null)$

Tabela 8: FP rast - Algoritam pretrazivanja FP stabla

Algortiam pretrage FP stabla
Ulaz FP stablo fp_tree i čvor α **Izlaz** Skup čestih skupova **if** tree sadrži samo jednu putanju P **then for each** kombinaciju β čvorova iz P **DO** generiši uzorak $\beta \cup \alpha$ sa podrškom jednakom minimalnoj podrški svih čvorova u β. **else for each** a_i iz tabele čestih uzoraka **DO** generiši uzorak $\beta = \alpha_i \cup \alpha$ sa podrškom jednakom $\alpha_i.support$ konstruiši kondicionalnu bazu uzoraka za β a zatim i kondicionalno drvo fp_tree_β **if** $fp\_tree_\beta \neq \emptyset$ **then** call $fp\_growth(fp\_tree_\beta, \beta)$

Tabela 9: FP rast - Algoritam fp_growth

Primer (primer preuzet iz [14]) Neka su date transakcije kao na slici (Slika 4) i $minsup = 3$.

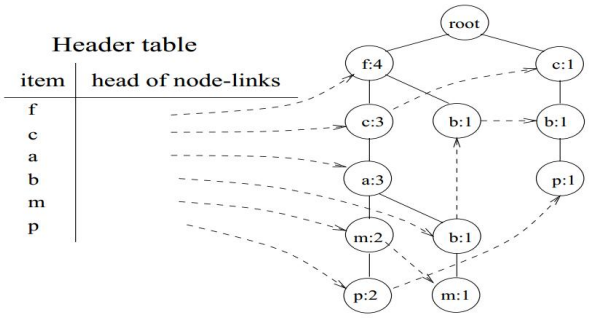
Transaction ID	Items Bought	(Ordered) Frequent Items
100	f, a, c, d, g, i, m, p	f, c, a, m, p
200	a, b, c, f, l, m, o	f, c, a, b, m
300	b, f, h, j, o	f, b
400	b, c, k, s, p	c, b, p
500	a, f, c, e, l, p, m, n	f, c, a, m, p

Slika 4: FP Rast - primer skupa transakcija

Izračunavanjem podrški za sve stavke i odbacivanjem retkih dobija se skup L:

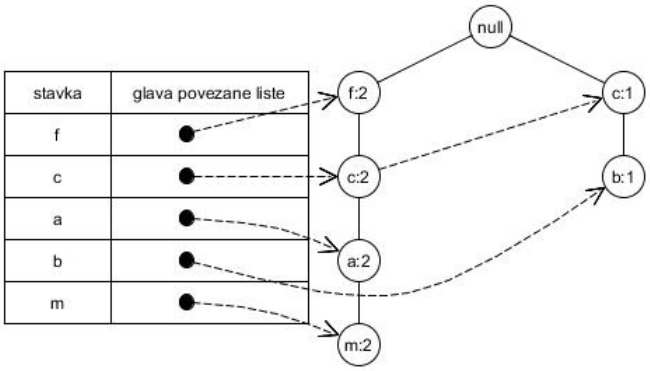
L= (f : 4); (c : 4); (a : 3); (b : 3); (m : 3); (p : 3)

Na osnovu novog skupa transakcija konstruiše se FP stablo :



Slika 5: FP stablo

Pronalaženje čestih skupova počinje sa pronalaženjem skupova sa sufiksom p (p se nalazi u dnu tabele). Kreira se kondicionalana baza uzoraka za sufiks p (f : 2; c : 2; a : 2; m : 2), (c : 1; b : 1) i na osnovu nje se konstruise FP stablo:



Slika 6: FP kondicionalno stablo za sufiks p

Kondicionalno stablo za sufiks p se najpre očisti od retkih stavki a zatim se rekurzivno pretražuje na isti način. Posle uklanjanja stavki sa manjom podrškom od *minsup* dobija se stablo sa jednom granom:

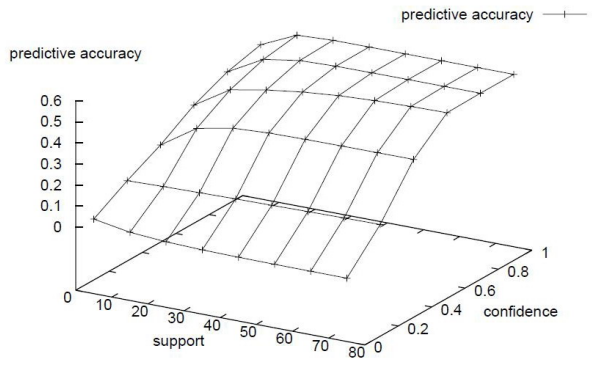
Slika 7: FP kondicionalno stablo za sufiks p nakon uklanjanja retkih stavki

Tako dobijamo da je jedini čest skup sa sufiksom p : cp ■.

2.4 Predictive apriori

Koja pravila su zanimljiva za korisnika zavisi od toga koji problem on rešava tj. koju hipotezu želi da dokaže istraživanjem. Pravila koja potvrđuju njegovu hipotezu su zanimljiva na osnovu subjektivne mere. Objektivno, najzanimljivija praviila su ona koja osim na skupu za učenje takođe važe i u realnom životu na podacima koji nisu unapred poznati.

Međutim, otkrivena pravila iz skupa za učenje nisu savršeno tačna i ne moraju oslikavati realnost, ona predstavljaju samo ***procenu*** korelacije koja važi u realnosti. Poverenje predstavlja relativnu frekvenciju na uzorku (skupu za učenje) verovatnoće važenja pravila u realnosti (čitavoj populaciji), tako da što je veća podrška relativna frekvencija će biti



Slika 8: Odnos tačnosti pravila i podrške i povrenja

bliža stvarnoj verovatnoći pravila. Kako verovovatnoća pravila zavisi od podrške i poverenja potrebno je pogodno izabrati ove dve vrednosti tako da maksimizuju verovatnoću. Pronalaženje balansa između ove dve veličine nije jednostavno zbog sledećeg:

1. veliko poverenje i velika podrška obično rezultiraju praznim skupom pravila
2. veliko poverenje i mala podrška ukazuje na to da je poverenje optimistična procena stvarne verovatnoće
3. malo poverenje i velika podrška garantuju da je poverenje dobra procena verovatnoće tj. da će verovatnoća pravila biti mala
4. malo poverenje i mala podrška znače da pravilo nije zanimljivo tj. da se slučajno javlja u skupu za učenje

Zbog toga algoritmi u kojima korisnik mora da izabere minimalne granice za poverenje i podršku nisu intuitivno laki za upotrebu.

Za razliku od prethodnih algoritama ***predictive apriori*** određuje kompromis između poverenja i podrške u cilju nalaženja određenog broja pravila koja sa velikom verovatnoćom važe i van skupa za učenje [13].

2.4.1 Očekivana verovatnoća pravila

Predictive apriori algoritam se zasniva na teoriji verovatnoće. U statistici se razlika između relativne frekvencije i verovatnoće koristi za proveru da li empirijsko zapažanje oslikava realnost. Tako i ovde uvodimo sledeće oznake:

$\hat{c}[x \to y]$- relativna frekvencija važenja pravila na skupu za učenje $D = \{a_1, ...a_k\}$

$c[x \to y]$- verovatnoća važenja (tačnosti) pravila u stvarnosti

Formalno :

$$\hat{c}[x \to y] = \frac{s(x \cup y)}{s(x)}$$

Kako u statistici broj eksperimenata igra važnu ulogu tako i u istraživanju podataka važi da što je veća podrška pravila, možemo biti sigurniji da je izračunato poverenje bliže stvarnom. To je jedan od razloga zašto se pravila sa malom podrškom smatraju nezanimljivim. Očekivana tačnost pravila se računa primenom Bajesove teoreme uzimajući da je apriori verovatnoća da proizvoljno pravilo $x \to y$ ima tačnost c:

$$\pi(c) = \frac{|\{[x \to y] \mid c([x \to y]) = c\}|}{|\{[x \to y]\}|}$$

Očekivana tačnost pravila je:

$$E(c([x \to y]) \mid \hat{c}([x \to y]), s(x))$$

$$= \int cP(c([x \to y]) = c \mid \hat{c}([x \to y]), s(x))\, dc$$

$$= \int c \frac{P(\hat{c}([x \to y]) \mid \hat{c}([x \to y]) = c, s(x))\, \pi(c)}{P(\hat{c}[x \to y] \mid s(c))} dc$$

$$\int P(c([x \to y]) = c \mid \hat{c}([x \to y]), s(x))\, dc = 1$$

$$\Leftrightarrow \int \frac{P(\hat{c}([x \to y]) \mid \hat{c}([x \to y]) = c, s(x))\, \pi(c)}{P(\hat{c}[x \to y] \mid s(c))} dc = 1$$

$$\Leftrightarrow P(\hat{c}[x \to y] \mid s(c)) = \int P(\hat{c}([x \to y]) \mid \hat{c}([x \to y]) = c, s(x))\, \pi(c)\, dc$$

Dobija se očekivana tačnost pravila:

$$E(c([x \to y]) \mid \hat{c}([x \to y]), s(x)) = \frac{\int cB[c, s(x)]\, (\hat{c}([x \to y]))\, \pi(c)\, dc}{\int B[c, s(x)]\, (\hat{c}([x \to y]))\, \pi(c)\, dc} \qquad (1)$$

2.4.2 Nabrajanje skupova sa dinamički određenom granicom podrške

1. **Input:** n željeni broj pravila i D skup za učenje sa binarnim atributima $a_1, ...a_k$.
2. **Let:** $\tau = 1$.
3. **for** i=1....k **do:** skiciraj određen broj pravila $[x \to y]$ određene dužine i, izračunaj poverenje za svako od njih. Neka je $\pi_i(c)$ funkcija raspodela poverenja

4. **for** svako c neka je $\pi(c) = \frac{\sum_{i=1}^{k} \pi_i(c)\binom{k}{i}\left(2^i-1\right)}{\sum_{i=1}^{k}\binom{k}{i}(2^i-1)}$
5. **Let:** $X_0 = \{\emptyset\}$; **Let:** $X_1 = \{\{a_1\}, ..., \{a_k\}\}$ skup svih jednočlanih skupova
6. **For** i=1...k-1 **While** ($i = 1$ ili $X_{i-1} = \emptyset$)
   - **If** $i > 1$ **Then** odredi sve skupove stavki dužine i na osnovu skupova stavki dobijenih u prethodnom koraku $X_i = \{x \cup x' \mid x, x' \in X_{i-1}, |x \cup x'| = i\}$, kao i u apriori algoritmu razmatraju se samo skupovi stavki x, x' koji se razlikuju u elementu sa najvećim indeksom. U ovom koraku se, takođe, eliminišu eventualni duplikati.
   - Jednim prolaskom kroz bazu izračunati podršku svih skupova stavki iz X_i i eliminisati one koji imaju manju podršku od τ.
   - $\forall x \in X_i$ **Call: GenRules** (x).
   - **If** *best* se izmenio **Then:** povećaj τ tako da bude najmanji mogući broj koji zadovoljava sledeće:

     $$E(c \mid 1, \tau) > E(c(best[n]) \mid \hat{c}(best[n]), s(best[n])).$$

     If $\tau > |D|$ **Then Exit**.
   - **If** τ je povećano u prethodnom koraku **Then** izbaci iz X_i sve nefrekventne skupove stavki
7. **Output** $best[1]....best[n]$ n najboljih pravila.

2.4.3 Generisanje pravila sa datim telom x

1. **Let** γ je najmanju broj koji zadovoljava sledeći uslov:

   $$E\left(c \mid \frac{\gamma}{s(x)}, s(x)\right) > E(c(best[n]) \mid \hat{c}(best[n]), s(best[n]))$$

2. **For** $j = 1...k - |x|$
   (a) **If** $j = 1$ **Then** $Y_1 = \{a_1, ...a_k\} \backslash x$
   (b) **Else** $Y_j = \{y \cup y' \mid y, y' \in Y_{j-1}, |y \cup y'| = j\}$
   (c) **For** $\forall y \in Y_j$ **Do:**
      i. Izračunaj podršku $s(x \cup y)$ **If** $s(x \cup y) < \gamma$ **Then** eliminiši y iz Y_j. **Continue** (nastavi sa sledećim y).
      ii. Izračunaj očekivanu tačnost pravila $E\left(c(x \to y) \mid \frac{s(x \cup y)}{s(x)}, s(x)\right)$ na osnovu formule (1).

iii. Ako je izračunata tačnost veća od tačnosti pravila iz niza *best*, ažuriraj niz *best*, eliminiši pravila koja su podrazumevana drugim i povećaj γ tako da bude najmanji broj za koji važi

$$E\left(c \mid \frac{\gamma}{s(x)}, s(x)\right) > E\left(c\left(best\left[n\right]\right) \mid \hat{c}\left(best\left[n\right]\right), s\left(best\left[n\right]\right)\right)$$

3. **If** neko pravilo eliminisano iz niza *best* **Then** krenuti od koraka 1.

Procena apriori verovatnoće. $\pi(c)$ Ako bismo birali slučajna pravila uniformnom raspodelom, uglavnom bismo dobijali dugačka pravila pa bi vrednost za kratka pravila od $\pi(c)$ bila mala. Da bismo to izbegli, za datu dužinu skiciramo određen broj pravila a za svako pravilo određujemo poverenje i to beležimo u histogramu (aproksimacija funkcije raspodele verovatnoće da pravilo ima tačnost c). Verovatnoća da pravilo ima dužinu i je:

$$P\left[dužina = i\right] = \frac{\binom{k}{i}\left(2^i - 1\right)}{\sum_{j=1}^{k}\binom{k}{j}\left(2^j - 1\right)}$$

Pa je apriori verovatnoća da pravilo ima tačnost c:

$$\begin{aligned}\pi(c) &= \sum_{i=1}^{k}\pi_i(c)\,P\left(dužina = i\right)\\ &= \frac{\sum_{i=1}^{k}\pi_i(c)\binom{k}{i}\left(2^i-1\right)}{\sum_{i=1}^{k}\binom{k}{i}\left(2^i-1\right)}.\end{aligned}$$

Uklanjanje redundantnih pravila. Prilikom generisanja pravila može se desiti da neko pravilo podrazumeva druga pravila. Kao npr. $[a \to b, c]$ podrazumeva važenje i pravila :$[a \to b]$, $[a \to b]$. Formalna definicija podrazumevanosti pravila je :

Definicija 2 *Neka su* $[x \to y]$ *i* $[x' \to y']$ *pravila pridruživanja i D skup podataka za učenje.*$[x \to y] \models [x' \to y'] \Leftrightarrow (\forall r \in D : (x \in r \to y \in r) \Rightarrow (x' \in r \to y' \in r))$.

Pravila koja su podrazumevana nekim drugim pravilom treba ukloniti. Sledeća teorema govori kako pronaći redundantna pravila.

Teorema 3 *Da li pravilo* $[x \to y]$ *podrazumeva pravilo* $[x' \to y']$ *može se proveriti sledećim testom* $[x \to y] \models [x' \to y'] \Leftrightarrow x \subseteq x' \wedge y \supseteq y'$.

2.5 Tertius

2.5.1 Osnovni pojmovi

Predstavljanje domena istraživanja logikom prvog reda je mnogo fleksibilnije od individualne reprezentacije gde se svaki entitet predstavlja jednim slogom. Tertius algoritam istražuje pravila pridruživanja u domenima predstavljenim na oba načina. Pravilo pridruživanja se posmatra kao logička formula impllikacije a zadatak pronalaženja pravila se svodi na otkrivanje hipoteza koje su najbolje **potvrđene** datim dokazima (primerima iz skupa za učenje) [8]. Relacija potvrđenosti se definiše na sledeći način:

Definicija 3 (Relacija potvrđenosti) *Za dati jezik prvog reda L, relacija potvrđenosti je binarna relacija* $|< \subseteq 2^L \times L$*; Ako* E $|<$ H *kažemo da dokaz* E *potvrđuje hipotezu* H*. Funkcija potvrđenosti je parcijalna funkcija definisana sa* $c : 2^L \times L \to [0, 1]$*; kažemo da dokaz* E *potvrđuje hipotezu u stepenu* $c(E, H)$ *ako je* $c(E, h)$ *definisano. Relacija potvrđenosti je kategorička ako važi da su sve hipoteze potvrđene istim dokazom međusobno konzistentne [8].*

Za dati skup dokaza E (zatvorenih formula) kategorička relacija potvrđenosti može biti definisana na sledeći način: neka je m(E) model tada E $|<$ H akko je H tačno u m(E). Kako bismo odredili najbolja pravila odnosno najbolje potvrđene hipoteze, mera potvrđenosti se definiše tako da se hipoteze mogu rangirati:

$$E \mid< H \text{ akko je } c\,(E, H) \geq c_0, \text{ gde je } c_0 \text{ data donja granica}$$

Zavisnost dva binarna atributa B i H može se odrediti Pirsonovom statistikom uzimajući njihovu nezavisnost za nultu hipotezu (μ_{ij} očekivana frekvencija istovremenog pojavljivanja promenljivih i i j,a n_{ij} uočena frekvencija istovremenog pojavljivanja i i j na skupu podataka za učenje):

$$\chi^2 = \sum_{ij} \frac{(n_{ij} - \mu_{ij})^2}{\mu_{ij}}, \tag{2}$$

gde je $\mu_{ij} = \frac{n_{i*} n_{*j}}{N}$. Na osnovu (1) dobija se mera zavisnosti dve promenljive Φ^2:

$$\Phi^2 = \frac{\chi^2}{N} = \sum_{ij} \frac{(n_{ij} - \mu_{ij})^2}{N \mu_{ij}} = \frac{(n_{11} n_{22} - n_{12} n_{21})^2}{n_{1*} n_{2*} n_{*1} n_{*2}} \tag{3}$$

Φ^2 uzima vrednosti između 0 (totalna nezavisnost promenljivih) i 1 (totalna zavisnost promenljivih).

$$B \longrightarrow H$$

	B	$\bar{B}$	total
H	n_{HB}	$n_{H\bar{B}}$	n_{H*}
$\bar{H}$	$n_{\bar{B}H}$	$n_{\bar{H}\bar{B}}$	$n_{\bar{H}*}$
total	n_{*H}	$n_{*\bar{H}}$	N

Tabela 10: Tabela kontigencije pridružena pravilu $B \rightarrow H$

2.5.2 Tertius mera potvrđenosti

Kako se pravilo posmatra kao logička formula implikacije čija je istinitosna vrednost $\perp$ u slučaju $\top \rightarrow \perp$, ideja tertius algoritma je da pronađe zanimljiva pravila razmatranjem kontraprimera (eng. counter-instane). Kontraprimer je onaj dokaz koji zadovoljava telo ali ne i glavu pravila.

Da bi pravilo bilo zanimljivo mora biti neočekivano i zadovoljeno. Formalne definicije neočekivanosti i zadovoljenosti pravila su date (π_{HB}- relativna očekivana frekvencija istovremenog pojavljivanja atributa B i H, a p_{HB}- relativna uočena frekvencija istovremenog pojavljivanja atributa H i B na skupu podataka za učenje):

Definicija 4 (Neočekivanost pravila (eng. novelty of a rule)) *Neočekivanost pravila se definiše na sledeći način:* $\Delta_{\bar{H}B} = \pi_{\bar{H}B} - p_{\bar{H}B}$.

Definicija 5 (Zadovoljenost pravila (eng. satifaction of a rule)) *Zadovoljenost pravila se definiše na sledeći način:* $\sigma_{\bar{H}B} = \frac{\pi_{\bar{H}B} - p_{\bar{H}B}}{\pi_{\bar{H}B}}$.

Kombinovanjem ove dve mere u jednu dobija se

$$\frac{\left(\pi_{\bar{H}B} - p_{\bar{H}B}\right)^2}{\pi_{\bar{H}B}} \tag{4}$$

što predstavlja vrednost od Φ^2 za polje tabele kontigencije (Tabela 10.) pridruženo kontraprimerima. Kako se ova vrednost minimizuje govori sledeća teorema:

Teorema 4 *Za svaku tabelu kontigencije sa datim* $\pi_{\bar{H}B}$ *i* $p_{\bar{H}B}$ *važi:*

$$\Phi^2 \geq \Phi_{\bar{H}B} = \left(\frac{\pi_{\bar{H}B} - p_{\bar{H}B}}{\sqrt{\pi_{\bar{H}B}} - \pi_{\bar{H}B}}\right)^2,$$

minimum se postiže u sledećoj tabeli:

	B	$\bar{B}$	total
H	$\sqrt{\pi_{\bar{H}B}} - p_{\bar{H}B}$	$1 + p_{\bar{H}B} - 2\sqrt{\pi_{\bar{H}B}}$	$1 - \sqrt{\pi_{\bar{H}B}}$
$\bar{H}$	$p_{\bar{H}B}$	$\sqrt{\pi_{\bar{H}B}} - p_{\bar{H}B}$	$\sqrt{\pi_{\bar{H}B}}$
total	$\sqrt{\pi_{\bar{H}B}}$	$1 - \sqrt{\pi_{\bar{H}B}}$	1

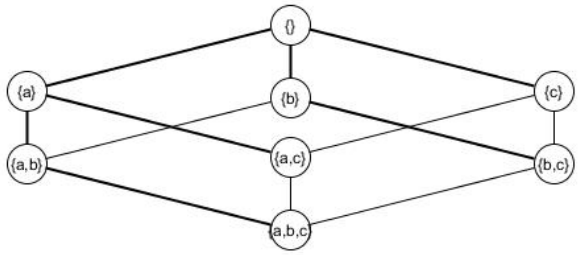
Definicija 6 *Stepen potvrđenosti pravila* $\forall(H \leftarrow B)$ *je definisan kao:*

$$\Phi_{\bar{H}B} = \pm\sqrt{\Phi^2_{\bar{H}B}} = \frac{\pi_{\bar{H}B} - p_{\bar{H}B}}{\sqrt{\pi_{\bar{H}B}} - \pi_{\bar{H}B}}$$

2.5.3 Generisanje i prečišćavanje pravila

Pretraga pravila se vrši strategijom od vrha ka dnu, počevši od praznog pravila tako da se prvo dobijaju kraća tj. generalnija pravila koja se dalje procesom pročišćavanja (eng. refinement) specijalizuju. Pročišćavanje se vrši dodavanjem literala, unifikacijom dve promenljive i instanciranjem promenljive konstantom iz domena. Redosled ovih operacija je strogo definisan kako bi se izbeglo generisanje istog pravila više puta.

Na ovaj način se prostor pretrage predstavljen rešetkom kroz koju se od praznog pravila do bilo koje hipoteze može doći na više načina pokriva stablom u kome se do bilo kog čvora može doći tačno jednom putanjom (Slika 9).



Slika 9: Tertius - Pokrivanje resteke stablom

Sledeća teorema govori kako odrediti da li specijalizacija datog pravila ima zadovoljavajuću potvrđenost:

Teorema 5 *Pravilo $B' \rightarrow H'$ je prihvatljiva specijalizacija pravila $B \rightarrow H$ ako važi:*

$$\Phi_{\bar{H}'B'} = \frac{1 - p_{H\bar{B}}}{1 + p_{H\bar{B}}}$$

3 Priprema i obrada podataka

3.1 Priprema podataka za obradu

3.1.1 Preuzimanje podataka

Podaci za istraživanje su preuzeti sa sajta Nacionalnog Instituta za Biotehnološke Informacije (NCBI). NCBI je deo američke nacionalne biblioteke za medicinu (NLM) i poseduje resurse potrebne za biotehnološka istraživanja. Tu spadaju mnogobrojni alati poput BLAST-a, COBALT-a, Cn3D-a i dr. kao i baze podataka stručne literature, objavljenih naučnih članaka, nukleotida, proteina, gena, genoma.... Sve podatke dostupne u NCBI bazama moguće je preuzeti FTP-om.

Za ovo istraživanje preuzeti su genomi 2753 mikroorganizama sa sledeće adrese: $ftp://ftp.ncbi.nlm.nih.gov/genomes/Bacteria/$. Među organizmima se nalaze bakterije i arhee. Podaci su organizovani tako da svakom organizmu odgovara jedan direktorijum u kome su smeštene datoteke sa svojstvima svih njegovih hromozoma odnosno plazmida. Svakom hromozomu odnosno plazmidu pridružene su datoteke različitih ekstenzija gde su grupisana odgovarajuća svojstva na sledeći način:

- gff (General Feature Format) - informacije o genima i proteinima (lokacija, lanac, produkt...)
- fna (Fasta Format) - sekvenca nukleotida gde je svaki nukleotid prikazan jednim slovom
- rnt (Rna Table) - informacije o ribonukleinskim kiselinama lokacija, lanac, dužina, tip...
- rpt (Report Format) - sumirana svojstva organizma (dužina DNA lanca, broj proteina, gena, RNA)

Datoteke se ne moraju preuzimati jedna po jedna za svaki organizam, umesto toga moguće je preuzeti odjednom datoteke datog tipa za sve organizme. Ovo su url-ovi za preuzimanje .gff, .fna, .rnt i .rpt datoteka za sve organizme:

- ftp://ftp.ncbi.nlm.nih.gov/genomes/Bacteria/all.gff.tar.gz
- ftp://ftp.ncbi.nlm.nih.gov/genomes/Bacteria/all.fna.tar.gz
- ftp://ftp.ncbi.nlm.nih.gov/genomes/Bacteria/all.rnt.tar.gz
- ftp://ftp.ncbi.nlm.nih.gov/genomes/Bacteria/all.rpt.tar.gz

3.1.2 Izračunavanje genomskih karakteristika

Preuzete datoteke potrebno je analizirati kako bi se dohvatile genomske karakteristike organizama odnosno podaci potrebni za njihovo izračunavanje. Genomske karakteristike nad kojima se istražuju pravila pridruživanja su:

- Dužina DNA lanca (broj nukleotida)
- Broj gena
- Broj proteina (kodirajućih sekvenci)
- Ukupna dužina kodirajućih sekvenci
- Procenat kodirajućih sekvenci
- Procenat nekodirajućih sekvenci
- GC procenat
- Broj enzima
- Procenat enzima u odnosu na ukupan broj proteina
- Broj RNA
- Broj tRNA
- Broj rRNA

Parsiranjem **rpt** datoteke dobijaju se dužina DNA lanca, broj gena, broj proteina i kodirajućih sekvenci. Primer **rpt** datoteke dat je na slici (Slika 10.)

```
Accession: NC_021879.1
GI: 526461592
DNA  length = 1477581
Taxname: Anaplasma phagocytophilum str. HZ2
Taxid: 1184253
Genetic Code: 11
Protein count: 1247
CDS count: 1247
Pseudo CDS count: 0
RNA count: 40
Gene count: 1295
Pseudo gene count: 8
Others: 1295
Total: 2588
```

Slika 10: Primer rpt datoteke

Parsiranjem **rnt** datoteke dobijaju se ukupan broj i broj pojedinačnih tipova RNA. Primer **rnt** datoteke dat je na slici (Slika 11.)

Parsiranjem **fna** datoteke i prebrojavanjem guanin i citozin baza i poređenjem dobijenog broja sa ukupnim brojem nukleotida dobija se GC%. Primer **fna** datoteke dat je na slici (Slika 12.)

Parsiranjem **gff** datoteke dobija se ukupna dužina kodirajuće sekvence kao i nazivi proteina kodirani njima. Imajući u vidu da se nazivi enzima uvek završavaju sufiksom "ase" dobija se i broj enzima. Primer **gff** datoteke dat je na slici (Slika 13.)

Acidaminococcus intestini RyC-MR95 chromosome, complete genome - 1..2487765
61 RNAs

Location	Strand	Length	PID	Gene	Synonym	Code	COG	Product
15620..17108	+	1489	352683411	-	Acin_3100	-	-	16S ribosomal RNA
18059..20311	+	2253	352683411	-	Acin_3101	-	-	23S ribosomal RNA
20421..20500	+	80	352683411	-	Acin_3102	-	-	5S ribosomal RNA
74062..74136	+	75	352683411	-	Acin_3000	-	-	Gly tRNA
120458..121946	+	1489	352683411	-	Acin_3103	-	-	16S ribosomal RNA
122897..125149	+	2253	352683411	-	Acin_3104	-	-	23S ribosomal RNA
125259..125338	+	80	352683411	-	Acin_3105	-	-	5S ribosomal RNA
270863..270939	-	77	352683411	-	Acin_3001	-	-	Arg tRNA
391435..392923	+	1489	352683411	-	Acin_3106	-	-	16S ribosomal RNA
393874..396126	+	2253	352683411	-	Acin_3107	-	-	23S ribosomal RNA

Slika 11: Primer rnt datoteke

```
>gi|384061216|ref|NC_017130.1| Acetobacter pasteurianus IFO 3283-26 plasmid pAPA26-013, complete sequence
CGCAGGTTGAGTTCCTGTTCCCGATAGATCCGATAAACCCGCTTATGATTCCAGAGCTGTCCCTGCACAT
TGCGCAGATACAGGAAACACAGACCAAATCCCCATCTCCTGTGAGCCTGGGTCAGTCCCACCAGAAGAGC
GGCAATCCTGTCGTTCTCCGCTGCCAGTCGCGGACGATAGCGAAAGCAGGTCTCGGATATCCCAAAAATC
CGACAGGCCAGCGCAATGCTGACCCCATGATGCGCCACAGCTTGTGCGGCCAGTTCCCGGGCGCTGGGCTG
GCCGCTTCATTTTTTTCCAAGGGCTTCCTTCAGGATATCCGTCTGCATGCTCAAATCCGCATACATGCGC
TTCAGCCGACGGTTCTCCTCTTCCAAAGCCTTCATCTGACTGATCATCGAAGCATCCATGCCGCCATATT
TCGCGCGCCACCGGTAAAACGTGGCGTTGCTGATCCCATGCTCCCGACACAGGTCAGGAACCGGGACACC
```

Slika 12: Primer fna datoteke

Obrada podataka je urađena u programskom jeziku JAVA. Za svaki tip gore pomenute datoteke (rpt,rnt,fna,gff) implementirana su četiri analizatora čiji prefiks naziva odgovara tipu datoteke koju obrađuje (biće obeležen sa *):

- *LineParser - parsira liniju date datoteke i dohvata bitne vrednosti iz nje
- *FileParser - koristi *LineParser nad svim linijama datoteke i sumiranjem dobijenih rezultata računa informacije o hromozomu (ili plazmidu)
- *OrganismAnalyser - poziva *FileParser nad svim datotekama, sumiranjem rezultata svih hromozoma (plazmida) računa informacije o celom organizmu
- *BacteriaAnalyser - poziva *OrganismAnalyser nad svim organizmima (tj. direktorijumima) i na taj način se dobijaju informacije o svim organizmima i smeštaju u odgovarajuću *Info SQL tabelu. (Slika 15) Proces pripreme podataka prikazan je na slici (Slika 14).

```
##gff-version 3
#!gff-spec-version 1.20
#!processor NCBI annotwriter
##sequence-region NC_021879.1 1 1477581
##species http://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi?id=1184253
NC_021879.1     RefSeq  region  1       1477581 .       +       .
ID=id0;Name=ANONYMOUS;Dbxref=taxon:1184253;Is_circular=true;gbkey=Src;genome=chromosome;mol_type=genomic DNA;old-
lineage=Bacteria%3B Proteobacteria%3B Alphaproteobacteria%3B Rickettsiales%3B Anaplasmataceae%3B Anaplasma%3B
phagocytophilum group%3B Anaplasma phagocytophilum%3B Anaplasma phagocytophilum H22;old-name=Anaplasma phagocytophilum
H22;strain=H22
NC_021879.1     RefSeq  gene    122     2683    .       +       .
ID=gene0;Name=YYU_00005;Dbxref=GeneID:16839547;gbkey=Gene;locus_tag=YYU_00005
NC_021879.1     Protein Homology        CDS     122     2683    .       +       0
ID=cds0;Name=YP_008332169.1;Parent=gene0;Dbxref=Genbank:YP_008332169.1,GeneID:16839547;gbkey=CDS;product=DNA polymerase
I;protein_id=YP_008332169.1;transl_table=11
```

Slika 13: Primer gff datoteke

Slika 14: Tok procesiranja podataka

3.1.3 Smeštanje podataka

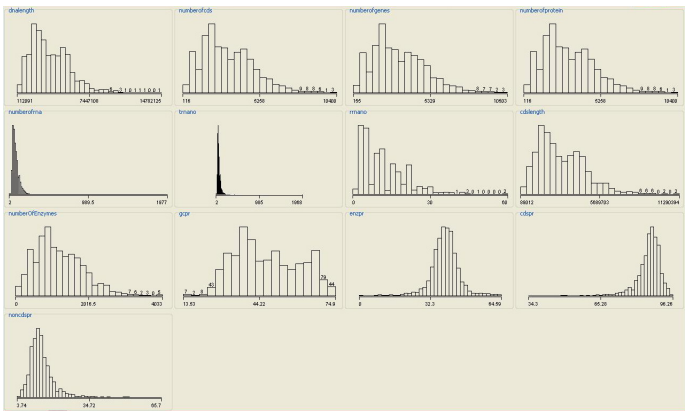
Nakon završene analize sirovih podataka, rezultati se smeštaju u sledeće tabele (tabele se nalaze u bazi podataka na MySQL serveru):

```
CREATE TABLE rntinfo(
    refseqid        integer      not null,
    trnano       integer              ,
    rrnano       integer              ,
    primary      key(refseqid)
);
CREATE TABLE fnainfo(
    refseqid        integer      not null,
    gcpr            real                 ,
    primary      key(refseqid)
);
CREATE TABLE gffinfo(
    refseqid        integer      not null,
    cdslength       integer              ,
    numberOfEnzymes integer              ,
    primary      key(refseqid)
);
CREATE TABLE rptinfo(
    refseqid        integer      not null,
    dnalength       integer              ,
    numberofcds     integer              ,
    numberofgenes   integer              ,
    numberofprotein integer              ,
    primary      key(refseqid)
);
```

Slika 15: Tabele u koje se smeštaju dobijeni rezultati na osnovu ulaznih podataka sa NCBI-a

3.2 Diskretizacija numeričkih vrednosti

Algoritmi pravila pridruživanja ne mogu biti primenjeni na kontinualne numeričke vrednosti. Na slici (Slika 16.) prikazana je raspodela numeričkih vrednosti genomskih karakteristika mikroorganizama na kojima će biti izvršeno istraživanje. Minimalna, maksimalna, srednja vrednost kao i standardna devijacija za svaki od atributa date su u tabeli (Tabela 11.). Za primenu Weka implementacije **apriori**, **predictive apriori** i **tertius** algoritma dovoljno je da podaci budu diskretni dok za algoritam **FP rasta** moraju biti binarni.



Slika 16: Raspodela vrednosti genomskih karakteristika

ATRIBUT	Minimum	Maximum	Mean	Standard deviation
DNA length	112091	14782125	3452154.679	1956387.604
Number of CDS	116	10400	3136.133	1728.331
Number of genes	155	10503	3256.48	1766.782
Number of proteins	116	10400	3135.846	1728.342
Number of RNA	2	1977	84.325	53.49
Number of tRNA	2	1968	72.336	48.438
Number of rRNA	0	60	11.989	8.442
CDS length	89012	11290394	2961560.423	1692045.369
Number of enzymes	0	4033	1192.55	646.352
GC%	13.53	74.9	47.364	12.873
Enzyme%	0	64.59	38.865	6.772
CDS%	34.303	96.259	85.76	5.51
Non CDS%	3.741	65.697	14.24	5.51

Tabela 11: Minimum, maksimum, srednja vrednost i standardna devijacija vrednosti genomskih karakteristika

Diskretizacija. Neka atribut A uzima vrednosti iz domena $D = [d_{min}, d_{max}]$. Interval D se deli na određen broj podintervala k $[d_{min}, d_1), [d_1, d_2), [d_2, d_3) ... [d_{k-1}, d_{max}]$. Svaka vrednost domena D se zamenjuje intervalom kom pripada.

Binarizacija. Binarizacija numeičkih vrednosti se vrši najpre diskretizacijom, a zatim se svakom intervalu dodeljuje novi binarni atribut: $A \in \{d_{min}, d_1\}, A \in \{d_1, d_2\}, A \in \{d_2, d_3\}, A \in \{d_{k-1}, d_{max}\}$, tako da se od jednog kontinualnog atributa dobija k binarnih.

Diskretizacija se može vršiti tako da dobijeni intervali budu:

- jednakih frekvencija
- jednakih širina

- proizvoljnih širina

Diskretizacija pomoću alata Weka vrši se filterima:

1. weka.filters.unsupervised.attribute.Discretize (intervali jednake frekvencije ili širine)
2. weka.filters.unsupervised.attribute.MathExpression (proizvoljni intervali)

Diskretizacijom podataka genomskih karakteristika na 2 intervala dobijaju se podaci prikazani u tabeli (Tabela 12.) (u pravilima će se umesto granica intervala pominjati reči SMALL i LARGE). U tabeli (Tabela 13.) nalaze se podaci posle diskretizacije podataka na 3 intervala jednake frekvencije (u pravilima će se umesto granica intervala pominjati reči SMALL, MEDIUM i LARGE kako bi pravila bila čitljivija).

ATRIBUT	Interval 1 SMALL	Interval 2 LARGE
DNA length	[112091-3052491],1378	(3052491-14782125],1378
Number of CDS	[116-2841],1378	(2841-10400],1378
Number of genes	[155-2948.5],1378	(2948.5-10503],1378
Number of proteins	[116-2841],1378	(2841-10400],1378
Number of RNA	[2-73.5],1378	(73.5-1977],1378
Number of tRNA	[2-63.5],1366	(63.5-1968],1390
Number of rRNA	[0-9.5],1393	(9.5-60],1363
CDS length	[89012-2611611.5],1378	(2611611.5-11290394],1378
Number of enzymes	[0-1090.5],1378	(1090.5-4033],1378
GC%	[13.53-45.435],1378	(45.435-74.9],1378
Enzyme%	[0-39.05448],1378	(39.05448-64.59],1378
CDS%	[34.303-86.773845],1378	(86.773845-96.259],1378
Non CDS%	[3.741-13.226155],1378	(13.226155-65.697],1378

Tabela 12: Diskretizovane vrednosti genomskih karakteristika podelom na dva intervala jednake frekvencije

ATRIBUT	Interval 1 SMALL	Interval 2 MEDIUM	Interval 3 LARGE
Dna length	[112091-2240087.5],919	(2240087.5-4215608],919	(4215608-14782125],918
Number of CDS	[116-2081.5],919	(2081.5-3848],919	(3848-10400],918
Number of genes	[155-2192.5],919	(2192.5-3956.5],919	(3956.5-10503],918
Number of proteins	[116-2081.5],919	(2081.5-3848],919	(3848-inf],918
Number of Rna	[2-60.5],934	(60.5-90.5],906	(90.5-1977],916
Number of tRNA	[2-53.5],900	(53.5-75.5],915	(75.5-1968],941
Number of rRNA	[0-6.5],1053	(6.5-15.5],904	(15.5-60],799
CDS length	[89012-1916342.5],919	(1916342.5-3608937.5],919	(3608937.5-11290394],918
Number of enzymes	[0-835.5],917	(835.5-1434],920	(1434-4033],919
GC%	[13.53-39.085],920	(39.085-53.135],919	(53.135-74.9],917
Enzyme%	[0-36.86934],919	(36.86934-41.119756],919	(41.119756-64.59),918
CDS%	[34.303-85.166008],919	(85.166008-88.206842],919	(88.206842-96.259),918
Non CDS%	[3.741-11.796556],919	(11.796556-14.840267],919	(14.840267-65.697),918

Tabela 13: Diskretizovane vrednosti genomskih karakteristika podelom na tri intervala jednake frekvencije

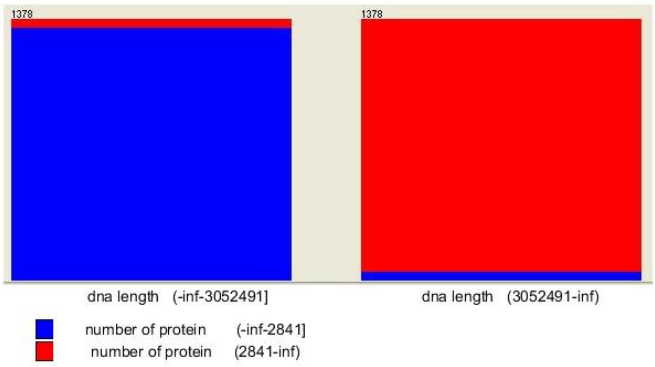
3.3 Rezultati i diskusija

3.3.1 Rezultati dobijeni istraživanjem nad diskretizovanim podacima na dva intervala jednake frekvencije

Posle primene Apriori algoritma sa sledećim parametrima: **s=0.2, c=0.6**, izdvajaju se zanimljiva pravila:

Organizmi sa dužim DNA lancem imaju i veći broj proteina (c=0.97) (Slika 17.)

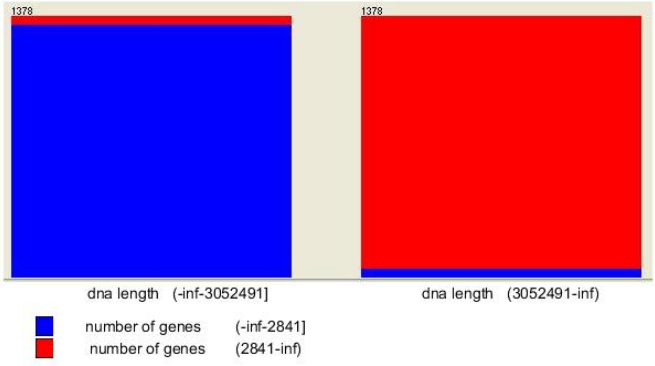
- $dnalength =' SMALL'$ 1387 $\Longrightarrow numberofprotein =' SMALL'$ 1340
- $numberofprotein =' LARGE'$ 1386 $\Longrightarrow dnalength =' LARGE'$ 1339
- $numberofprotein =' SMALL'$ 1388 $\Longrightarrow dnalength =' SMALL'$ 1340
- $dnalength =' LARGE'$ 1387 $\Longrightarrow numberofprotein =' LARGE'$ 1339



Slika 17: Odnos dužine DNA lanca i broja proteina

Organizmi sa dužim DNA lancem imaju i veći broj gena (c=0.96)(Slika 18.)

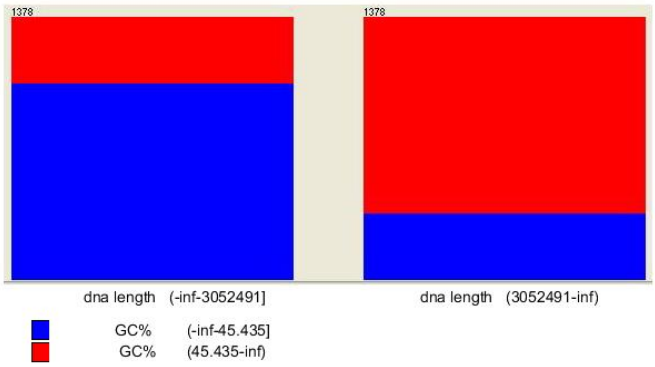
- $numberofgenes =' SMALL'1387 ==> dnalength =' SMALL'1337$
- $dnalength =' SMALL'1387 ==> numberofgenes =' SMALL'1337$
- $numberofgenes =' LARGE'1387 ==> dnalength =' LARGE'1337$
- $dnalength =' LARGE'1387 ==> numberofgenes =' LARGE'1337$



Slika 18: Odnos dužine DNA lanca i broja gena

Organizmi sa dužim DNA lancem imaju i veći GC%, kao što se navodi u [3] (c=0.75)

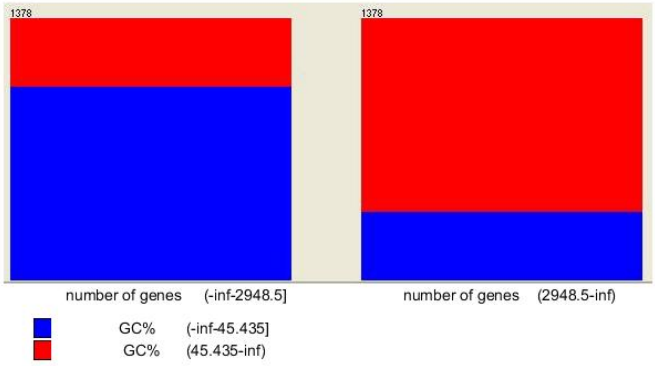
- $dnalength =' SMALL'1387 ==> gcpr =' SMALL'1037$
- $gcpr =' LARGE'1386 ==> dnalength =' LARGE'1036$
- $gcpr =' SMALL'1388 ==> dnalength =' SMALL'1037$
- $dnalength =' LARGE'1387 ==> gcpr =' LARGE'1036$



Slika 19: Odnos dužine DNA lanca i GC%

Organizmi sa većim brojem gena imaju i veći GC%,važi i obrnuto (c=0.74) (Slika 20.)

- $numberofgenes =' SMALL'1387 ==> gcpr =' SMALL'1028$
- $gcpr =' LARGE'1386 ==> numberofgenes =' LARGE'1027$
- $gcpr =' SMALL'1388 ==> numberofgenes =' SMALL'1028$
- $numberofgenes =' LARGE'1387 ==> gcpr =' LARGE'1027$

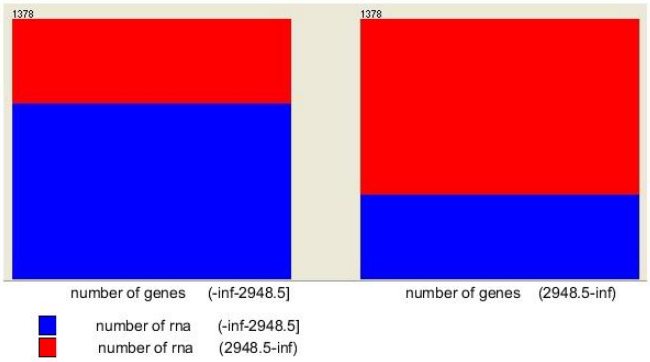


Slika 20: Odnos broja gena i GC%

Organizmi sa dužim DNA lancem imaju i veći broj RNA (c=0.68) (Slika 21.)

- $numberofgenes =' LARGE'1387 ==> numberofrna =' LARGE'949$
- $nmberofrna =' SMALL'1385 ==> numberofgenes =' SMALL'947$
- $numberofrna =' LARGE'1389 ==> numberofgenes =' LARGE'949$

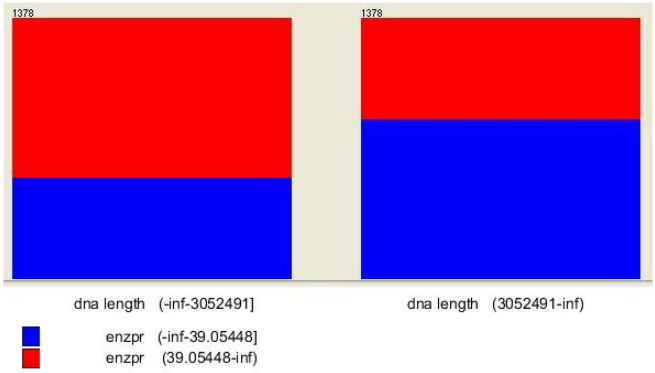
- $numberofgenes =' SMALL' 1387 ==> numberofrna =' SMALL' 947$



Slika 21: Odnos broja gena i broja RNA

Organizmima sa dužim lancem DNA odgovara manji procenat enzima, važi i obrnuto (Slika 22.)

- $enzpr =' SMALL' 1382 ==> dnalength =' LARGE' 842$
- $enzpr =' LARGE' 1381 ==> dnalength =' SMALL' 841$
- $dnalength =' LARGE' 1387 ==> enzpr =' SMALL' 842$
- $dnalength =' SMALL' 1387 ==> enzpr =' LARGE' 841$



Slika 22: Odnos dužine DNA lanca i procenta enzima

3.3.2 Rezultati dobijeni istraživanjem nad diskretizovanim podacima na tri intervala jednake frekvencije

Pravila koja povezuju dužinu DNA lanca i broj gena i proteina potvrđena su i rezultatima dobijenim istraživanjem nad podacima diskretizovanim na tri intervala, primenom

algoritma FP rasta:

Organizmima sa dužim DNA lancem odgovara i veći broj gena

- $[numberofgenes =' LARGE' = t] : 918 ==> [dnalength =' LARGE' = t] : 865 < conf : (0.94) > lift : (2.83) lev : (0.2) conv : (11.34)$
- $[dnalength =' LARGE' = t] : 918 ==> [numberofgenes =' LARGE' = t] : 865 < conf : (0.94) > lift : (2.83) lev : (0.2) conv : (11.34)$

Organizmima sa kratkim DNA lancem odgovara i mali broj gena

- $[numberofgenes =' SMALL' = t] : 919 ==> [dnalength =' SMALL' = t] : 857 < conf : (0.93) > lift : (2.8) lev : (0.2) conv : (9.72)$
- $[dnalength =' SMALL' = t] : 919 ==> [numberofgenes =' SMALL' = t] : 857 < conf : (0.93) > lift : (2.8) lev : (0.2) conv : (9.72)$

Organizmima sa dugačkim DNA lancem odgovara i veliki broj proteina

- $[numberofprotein =' LARGE' = t] : 918 ==> [dnalength =' LARGE' = t] : 862 < conf : (0.94) > lift : (2.82) lev : (0.2) conv : (10.74)$
- $[dnalength =' LARGE' = t] : 918 ==> [numberofprotein =' LARGE' = t] : 862 < conf : (0.94) > lift : (2.82) lev : (0.2) conv : (10.74)$

Organizmima sa kratkim DNA lancem odgovara i mali broj proteina

- $[numberofprotein =' SMALL' = t] : 919 ==> [dnalength =' SMALL' = t] : 856 < conf : (0.93) > lift : (2.79) lev : (0.2) conv : (9.57)$
- $[dnalength =' SMALL' = t] : 919 ==> [numberofprotein =' SMALL' = t] : 856 < conf : (0.93) > lift : (2.79) lev : (0.2) conv : (9.57)$

Naredna pravila dobijena su primenom Tertius algoritma i ukazuju na to da organizmi sa malim lancem DNA (malim brojem gena, RNA i tRNA) imaju mali procenat nekodirajuće odnosno veliki procenat kodirajuće sekvence:

- $./ * 0, 2148210, 091074 * /dnalength =' SMALL'$ and $rrnano =' SMALL' ==> noncdspr =' SMALL'$
- $/*0, 2139040, 099419*/dnalength =' SMALL'$ and $numberofrna =' SMALL' ==> noncdspr =' SMALL'$
- $/*0, 2035600, 098694*/numberofgenes =' SMALL'$ and $numberofrna =' SMALL' ==> noncdspr =' SMALL'$
- $/*0, 2001930, 091800*/numberofgenes =' SMALL'$ and $rrnano =' SMALL' ==> noncdspr =' SMALL'$
- $/ * 0, 2151260, 091074 * /dnalength =' SMALL'$ and $rrnano =' SMALL' ==> cdspr =' LARGE'$

- $/*0, 2142270, 099419*/dnalength =' SMALL'$ and $numberofrna =' SMALL' ==> cdspr =' LARGE'$
- $/*0, 2038780, 098694*/numberofgenes =' SMALL'$ and $numberofrna =' SMALL' ==> cdspr =' LARGE'$
- $/*0, 2004930, 091800*/numberofgenes =' SMALL'$ and $rrnano =' SMALL' ==> cdspr =' LARGE'$

Organizmima sa malim lancem DNA (malim brojem gena, kodirajućih sekvenci i proteina) i malim brojem tRNA odgovara i mali procenat GC%:

- $/*0, 3030430, 075109*/numberofgenes =' (SMALL]'$ and $trnano =' (SMALL]' ==> gcpr =' (SMALL]'$
- $/*0, 3005790, 075472*/numberofcds =' (SMALL]'$ and $trnano =' (SMALL]' ==> gcpr =' (SMALL]'$
- $/*0, 3005790, 075472*/numberofprotein =' (SMALL]'$ and $trnano =' (SMALL]' ==> gcpr =' (SMALL]'$
- $/*0, 2997970, 078737*/dnalength =' (SMALL]'$ and $trnano =' (SMALL]' ==> gcpr =' (SMALL]'$

Organizmima sa dugačkim lancem DNA (velikim brojem gena, kodirajućih sekvenci i proteina) i srednjim brojem rRNA odgovara i veliki procenat GC%:

- $/*0, 2578550, 018505*/numberofcds =' (LARGE)'andrrnano =' (MEDIUM]' ==> gcpr =' (LARGE)'$
- $/*0, 2578550, 018505*/numberofprotein =' (LARGE)'andrrnano =' (MEDIUM]' ==> gcpr =' (LARGE)'$
- $/*0, 2540510, 018868*/numberofgenes =' (LARGE)'andrrnano =' (MEDIUM]' ==> gcpr =' (LARGE)'$
- $/*0, 2536310, 022134*/dnalength =' (LARGE)'andrrnano =' (MEDIUM]' ==> gcpr =' (LARGE)'$

3.3.3 Vreme izvršavanja

Pokretanjem sva četiri algoritma na istom skupu podataka, sa istim atributima diskretizovanim na isti način, mogu se uporediti njihove brzine izvršavanja. Iz tabele (Tabela 14.) se vidi da je za pronalaženje hiljadu pravila, najviše vremena potrebno Predictive Apriori algoritmu dok se izvršavanje algoritma FP rasta najbrže završava .

	Apriori	Predictive Apriori	Fp rast	Tertius
Diskretizacija na dva intervala	5s	29min 1s	manje od 1s	1min 4s
Diskretizacija na 3 intervala	6s	17min 3s	manje od 1s	2min 20s

Tabela 14: Poređenje vremena izvršavanja algoritama potrebnog za pronalaženje 1000 pravila

3.3.4 Argumenti komandne linije

Zadavanjem opcija iz komandne linije može se uticati na izvršavanje algoritama. Obzirom na to da istraživanje pravila pridruživanja može biti nadgledano i nenadgledano (nadgledano znači da je unapred zadata glava očekivanog pravila), svi algoritmi imaju opciju zadavanja atributa klase. Izuzetak je algoritam FP Rasta gde se može specificirati koji atributi će se pojavljivati na nivou celog pravila, ne samo glave. Najmanje opcija se može zadati Predictive Apriori algoritmu jer je nejegova osnovna ideja da se izvršava bez zadatih donjih ograničenja kako bi korisniku bio lakši za upotrebu. Kako ne postoji mogućnost specificiranja broja stavki u pravilu, rezultati ovog algoritma su obično teški za interpretiranje usled velikog broja stavki u pravilu. U tabelama: (Tabela 15.), (Tabela 16.), (Tabela 17.) i (Tabela 18.) date su opcije svih pomenutih algoritama za istraživanje pravila pridruživanja Weka alatom.

Apriori opcije		
opcija	značenje	podrazumevana vrednost
car	predefinisati glavu pravila	false
classIndex	indeks atributa klase	-1
delta	Iterativno umanjuj podršku za ovaj faktor	0.05
lowerBoundMinSupport	minimalna podrška	0.1
metricType	mera valjanosti pravila	confidence
removeAllMissingCols	ukloni kolone sa svim nedostajućim vrednostima	false
numRules	broj pravila koje treba naći	10
minMetric	prikaži samo pravila sa većom vrednosću od ove za izabranu metriku	0.9
outputItemSets	sem pravila ispisuje i sve skupove stavki	false
significanceLevel	mera značajnosti	-1.0
upperBoundMinSupport	gornja granica za minimalnu podršku	1.0
verbose	opširan ispis tokom izvršavanja	false

Tabela 15: Opcije apriori algoritma

Predictiveapriori opcije		
opcija	značenje	podrazumevana vrednost
car	predefinisati glavu pravila	false
classIndex	indeks atributa klase	-1
numRules	broj pravila	100

Tabela 16: Opcije predictiveapriori algoritma

FpGrowth opcije		
opcija	značenje	podrazumevana vrednost
delta	iterativno umanjuj podršku, za ovaj faktor	0.05
findAllRulesForSupportLevel	pronađi sva pravila sa minimalnom podrškom i poverenjem	false
lowerBoundMinSupport	donja granica podrške	0.1
maxNumberOfItems	maksimalan broj stavki u pravilu	-1
metricType	mera valjanosti pravila	Confidence
minMetric	prikaži samo pravila sa vrednošću većom od ove za izabranu metriku	0.9
numRulesToFind	broj pravila	10
positiveIndex	koja vrednost binarnog atributa se tretira kao,pozitivna vrednost	2
rulesMustContain	lista stavki koju dobijena pravila moraju sadržati	
transactionsMustContain	lista stavki koju transakcija mora sadržati da bi bila uzeta u obzir	
upperBoundMinSupport	lista stavki koju dobijena pravila moraju sadržati	1.0
useORForMustContainList	gornja granica za minimalnu podršsku	false

Tabela 17: Opcije algoritma FP rasta

Tertius opcije		
Opcija	Značenje	Podrazumevana vrednost
classIndex	indeks atributa klase	0
classification	predefinisati glavu pravila	false
confirmationTrashhold	minimalna potvrđenost pravila	0.0
confirmationValues	broj pravila	10
frequencyTrashhold	donja granica frekvencije	1.0
hornClauses	koristiti Hornove klauzule	false
missingValues	nedostajuće vrednosti	matches all
negation	dozvoljena negacija u pravilu	none
noiseTrashhold	procenat kontraprimera	1.0
numberLiterals	broj stavki u pravilu	4
repeatLiterals	dozvoljeno ponavljanje literala u pravilima	false
rocAnalysis	ROC analiza	false
valuesOutput	ispis tokom izvršavanja	no

Tabela 18: Opcije Tertius algoritma

4 Zaključak

Cilj ovog rada bilo je otkrivanje skrivenih pravila među genomskim karakteristikama prokariotskih organizama kao što su: dužina genoma, procenat i dužina kodirajuće i nekodirajuće sekvence, broj proteina, enzima, molekula transportnih i ribozomalnih RNA i GC%.

Sprovedenim istraživanjem pravila pridruživanja nad bazom prokariota potvrđena je hipoteza o korelaciji između dužine genoma i GC% [2],[3]. Takođe, dobijena pravila govore i o nekim očigledim i očekivanim vezama kao što su korelacija dužine DNA i broja gena i proteina. Uočena su i manje očekivana pravila kao npr. veza između dužine DNA lanca i procenta kodirajuće sekvence i procenta enzima u organizmu.

Dobijeni rezultati koji su u skladu sa ranijim znanjima kvalifikuju primenjene metode kao pogodne za dalja istraživanja u bioinformatici.

Proširivanjem skupa proučavanih atributa fenotipskim svojstvima i uočavanjem pravila među njima, mogla bi se bolje razumeti široka rasprostranjenost bakterija kao i njihov fleksibilan metabolizam koji im omogućuje opstanak u ekstremnim uslovima.

Primenom opisanih metoda na podatke ma kojih organizama sa atributima koji predstavljaju gene i fenotipska svojstva može se odrediti veza između većeg broja gena i nekog fenotipskog svojstva. Uočavanje veza između prisustva (ili odsustva) gena i oboljenja bilo bi od velikog značaja u medicini, pre svega u prenatalnoj dijagnostici i onkogenetici.

5 Literatura

[1] Pang-Ning Tan, Michigan State University, Michael Steinbach, University of Minnesota Vipin Kumar, University of Minnesota, Introduction to Data Mining

[2] Gordana Pavlović Lažetić, Vesna Pajić, Nenad Mitić, Jovana Kovačević, Miloš Beljanski, Mining Associations for Organism Characteristics in Prokaryotes - an Integrative Approach 2nd International Work-Conference on Bioinformatics and Biomedical Engineering, Granada, Spain, April 7-9 2014,

[3] Bacterial genomic G+C composition-eliciting environmental adaptation,Scott Mann,Yi-Ping Phoebe Chen,Faculty of Science and Technology, Deakin University, Australia,ARC Centre of Excellence in Bioinformatics, Australia

[4] Genomics of bacteria and archaea: the emerging dynamic view of the prokaryotic world

[5] Radivoje Papović, Ljiljana Luković, Humana genetika, Medicinski fakultetUniverzitet u Beogradu, Beograd 2011.

[6] The Prokaryotes: A Handbook on the Biology of Bacteria, Martin Dworkin, Stanley Falkow,2006 Springer Science & Business Media, Inc.

[7] R. Agrawal, T. Imielinski, A. Swami (1993), "Mining Associations between Sets of Items in Massive Databases"

[8] PETER A. FLACH, NICOLAS LACHICHE, Confirmation-Guided Discovery of First-Order Rules with Tertius

[9] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth, From Data Mining to Knowledge Discovery in Databases

[10] William J. Frawley, Gregory Piatetsky-Shapiro, and Christopher J. Matheus, Knowledge Discovery in Databases: An Overview

[11] Rakesh Agrawal Tomasz Imielinski Arun Swami, Mining Association Rules between Sets of Items in Large Databases

[12] Microbial genotypephenotype mapping by class association rule miningMakio Tamura, Patrik D'haeseleer, Bioinformatics Volume 24, Issue 13, Pp. 1523-1529

[13] Finding association rules that trade support optimally against confidence by Tobias Scheffer Humboldt-Universitat zu Berlin, Department of Computer Science, Unter den Linden 6, 10099 Berlin, Germany. E-mail: scheffer@informatik.hu-berlin.de

[14] Mining Frequent Patterns without Candidate Generation Jiawei Han, Jian Pei, and Yiwen Yin School of Computing Science Simon Fraser University

[15] Fleischmann RD, Adams MD, White O, Clayton RA, Kirkness EF, Kerlavage AR, Bult CJ, Tomb JF, Dougherty BA, Merrick JM, et al. Whole-genome random sequencing and assembly of Haemophilus influenzae Rd Science. 1995.