## Asocijativna analiza

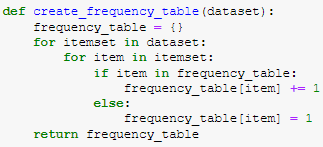
### Objašnjenje načina implementacije algoritma

Asocijativna analiza sastoji se u identifikovanju jakih asocijativnih pravila u datom skupu transakcija. Za određivanje asocijativnih pravila može se koristiti više različitih algoritama, od kojih je jedan Apriori algoritam. Ovaj algoritam koristi “frekventne skupove” da bi generisao asocijativna pravila. Zasniva se na konceptu da se podskup frekventnog skupa takođe frekventan skup. Cilj je iz početnog skupa transakcija odrediti frekventne podskupove odnosno skupove elemenata koji se javljaju zajedno minimalan broj puta(“minimum support count”), pri čemu se taj minimalan broj određuje posebno, u zavisnosti od skupa transakcija, a zatim generisati pravila asocijacije i na osnovu minimuma “minimum confidence” odrediti koja pravila asocijacije zaista važe u skupu transakcija.

Ovaj algoritam sam implementirala u nekoliko koraka:

1. Korak – Kreiranje tabele učestalosti

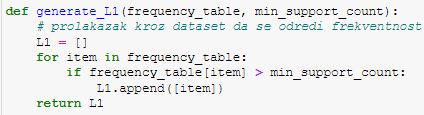
Kreirala sam tabelu učestalosti svih stavki koje se javljaju u svim transakciama pomoću funkcije create\_frequency\_table(dataset) čiji je parametar skup transakcija. Cilj je pronaći broj pojavljivanja pojedinačnih stavki, a zatim filtrirati one koje ne ispunjavaju minimalan „support count“.



1. Korak – Generisanje L1, liste svih kandidata skupova stavki dužine 1

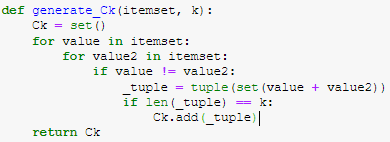
Generisanje kiste L1 je urađeno pomoću funkcije generate\_L1. Ova funkcija ima dva parametra:

* 1. frequency\_table - tabelu učestalosti koja se generiše pozivom funkcije create\_frequency\_table
  2. min\_support\_count – minimalni broj javaljanja elementa u svim transakcijama da bi se smatrao važnim za razmatranje u asocijativnoj analizi



1. Korak – Generisanje Ck, liste svih kandidata skupova stavki dužine k  
   Generisanje liste Ck je urađeno pomoću funkcije generate\_Ck koja ima dva parametra
   1. itemset – Skup stavki iz kog je potrebno pronaći sve moguće parove koji zajedno imaju k elemenata
   2. k – broj koji određuje sužinu podskupa koji tražimo

Za tip podatka za Ck sam odabrala set, zato što set sprečava duplikate.



1. Korak – Generisanje Lk, liste svih čestih skupova stavki dužine k

Za ovaj korak sam najpre kreirala pomoćnu funkciju frequency\_of\_candidate koja određuje frekventnost kandidatskog seta u početnom skupu transakcija. Ova funkcija ima dva parametra:

* 1. datasest – skup transakcija
  2. candidate – skup kandidat za koga se određuje učestalost

Zatim je implementirana funkcija generate\_Lk koja određuje setove veličine k koji su frekventni. Ova funkcija ima tri parametra:

1. dataset – originalni skup transakcija koji služi da bi pomoću funkcije frequency\_of\_candidate mogla da se odredi frekventnost kandidata iz skupa Ck
2. Ck – skup kandidata iz kog je potrebno isfiltrirati one koji nemaju dovoljnu učestalost
3. min\_support\_count – parametar koji određuje minimalnu učestalost za smatranje kandidata frekventnim skupom

Najpre se za sve kandidate iz seta kandidata Ck određuje učestalost i dodaju se u dictionary. U prolazu kroz dictionary se određuju ključevi onih kandidata koji će biti obrisani. Zatim se na osnovu liste ključeva brišu određeni kandidati iz dictionary-ja i konačno, od ključeva, koji su zapravo frekventni kandidati se formira lista koja predstavlja povratni parametar funkcije.

1. Generisanje svih frekventnih setova počevši od dužine 2.

Da bi se implementirao ovaj korak potrebno je nasumično pozivati funcije iz koraka 3 i 4 sve dok skup kandidata Ck ne postane prazan.

Za ovaj korak je zadužena funkcija generate\_candidate\_sets koja ima dva parametra:

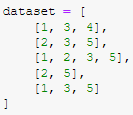
* 1. dataset – originalni skup transakcija iz kog će se odrediti svi frekventni skupovi
  2. min\_support\_count – granica koja oynačava koliko minimalno puta treba skup da se javi u originalno skupu transakcija da bi se smatrao učestalim

1. Generisanje asocijativnih pravila

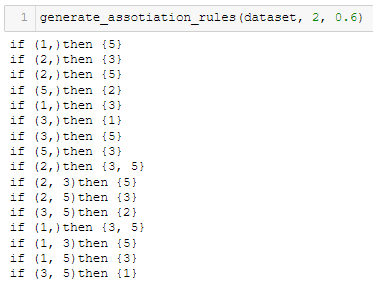
Ostvareno je implementacijom funcije generate\_association\_rules. Prvo se poziva funkcija za generisanje svih frekventnih skupova, a onda se za svaki od njih pronalaze svi podsetovi, i utvrđuju asocijativna pravila formata „if A then B“ za svako A, B gde su A i B podskupovi svakog frekventnog skupa. Ako pravilo zadovoljava „min\_confidence“ uslov onda se uvrštava u skup asocijativnih pravila za taj skup podataka.

### Skup podataka

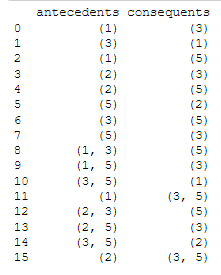
Skup podataka koji sam koristila sam sama izgenerisala. Korišćen je jednostavan skup podataka kako bih utvrdila da li sam ispravno implementirala algoritam upoređivanjem skupa asocijativnih pravila generisanih pomoću funkcije iz biblioteke **mlxtend** sa skupom asocijativnih pravila generisanih algoritmom koji sam ja implementirala.



### Zaključak



1 Asocijativna pravila koja se dobijaju sa mojom implementacijom algoritma



2 Asocijativna pravila koja se dobijaju korišćenjem gotove funkcije iz biblioteke mlxtend

Zakjučak je da se generišu ista pravila na oba načina što je ukazatelj na to da je implementacija ispravna.