Pēteris Račinskis
pr20015

Mājas darbs 2: Asociācijas, lēmumu koki

Uzdevums 1

a. *Itemsets un minimum support izmaiņas atkarībā no delta un numRules*

Eksperimentējot ar WEKA Apriori algoritma implementāciju var secināt, ka:

1. Asociācija (“rule”) tiek ieskaitīta, ja tās metrika (šajā gadījumā - “confidence”) ir vismaz *minMetric*;
2. Elementu kopa Ln tiek saglabāta, ja tās atbalsts ir vismaz minimum support;
3. Algortims darbojas iteratīvi, par *delta* katrā iterācijā samazinot *minimum support*;
4. Algoritms beidz darbu, ja sasniegts viens no diviem nosacījumiem: atrastas vismaz *numRules* ieskaitītas asociācijas vai sasniegts *lowerBoundMinSupport* un vairs nevar samazināt *minimum support*.

Attiecīgi izriet, ka:

1. Samazinot *minimum suppor*t, katrā kopas izmērā n atrasto elementu kopu (*itemsets*) Ln skaits palielinās;
2. Samazinot *lowerBoundMinSupport*, tiek veikts vairāk iterāciju, atrasts vairāk *itemsets* un vairāk asociāciju (*rules*), ja vispirms netiek sastapts *numRules*;
3. Palielinot numRules, tiek veikts vairāk iterāciju, atrasts vairāk *itemsets* un vairak asociāciju (*rules*), ja vispirms netiek sastapts *lowerBoundMinSupport*;
4. Samazinot *delta*, tiek veikts vairāk iterāciju lai sasniegtu to pašu *minimumSupport*, bet ir iespējams precīzāk atrast tieši nepieciešamo *itemsets*/asociāciju skaitu un tuvāku *minimumSupport* tieši prasītajam *numRules* (piemēram, minMetric<conf> = 0.75, N = 10 gadījumā, delta = 0.05 atrod minimum support 0.35 ar 13 iterācijām, taču delta = 0.01 - 0.38 ar 62 iterācijām; otrajā gadījumā *itemsets* skaits ir mazāks).

b. Metriku salīdzinājums

Veicot asociāciju meklēšanu, mainot tikai metrikas tipu un robežu (n=10 d=0.05), novērotas sekojošās sakarības:

1. Ar *confidence*, absolūti dominē sakarības formā x => bread and cake, kas jau tika novērots iepriekšējā mājas darbā. Pieņemot f=>g, gan f, gan g skaits ir samērā neliels;
2. Pie konservatīviem parametriem, metrikas *lift* un *leverage* sniedz ļoti līdzīgus rezultātus kur pieņemot f=>g, gan f, gan g skaits ir lielāks. Pie lift=1.1/lev=0.05 rezultāti ir visi tie paši, taču atšķiras to secība. Veikto iterāciju skaits vienāds (13);
3. Uzstādot agresīvākus parametrus (lift=2.52, leverage=0.0785), lai limitējošais faktors būtu *minSupport* (0.1), iegūst atšķirīgus rezultātus. Manāma sakarība: ar *lift* iegūtajās asociācijās dominē tās, kam (pieņemot f=>g) g skaits ir mazāks; ar *leverage* iegūtajās asociācijās dominē tās, kurām g skaits ir lielāks. Šī sakariba ļoti izteikti saglabājas, arī palielinot *numRules*;
4. *Conviction* gadījumā rezultāti ir ļoti līdzīgi kā pie *Confidence*, taču parādās vairākas asociācijas, kas nav pie *Confidence* (formā x=>vegetables).

Confidence metrika darbojas tīri ar nosacītajām varbūtībām - P(g|f); Ja ir kāds atribūts, ko ļoti precīzi var paredzēt ar daudziem citiem, tas varēs pilnībā dominēt rezultātos (atbildot uz autora paša sūdzībām pirmajā mājas darbā) - neatkarīgi no tā, cik nozīmīga uz kopējā sakarību skaita ir katra atrastā. Nosacītai varbūtībai nav simetrijas, tāpēc f=>g situācijā P(g) vai count(g) nav tiešas ietekmes uz rezultātu. *Bread and cake* ir ļoti lielas nosacītās varbūtības pret daudziem citiem atribūtiem, bet katras daļa no kopējā *bread and cake* ierakstu skaita var būt samērā neliela.

Conviction parādās sakarība pret 1-P(g) vai 1-(count(g)/total), kas nozīmē, ka atribūti, kas parādās daudzās rindās, saņem zemāku novērtējumu - līdz ar to starp neskaitāmajiem *Bread and cake* var iespiesties arī kāda rinda ar *vegetables*.

Lift un *Leverage* ļoti liela nozīme ir P(f) un P(g) jeb count(f) un count(g) - tāpēc biežāki atribūti saņem mazākus novērtējumus, augsts novērtējums ir tādiem, kuros paredzēto rindu skaits ir liels attiecībā pret kopējo rindu skaitu ar šo atribūtu. Galvenā atšķirība ir tā, ka *Lift* gadījumā salīdzinoši augstāku novērtējumu saņem attiecības (dalīšana, relatīvās vērtības), bet *Leverage* - starpības (atņemšana, absolūtās vērtības).

Pēteris Račinskis
pr20015

Uzdevums 2

a. Lēmumu koku salīdzinājums dažādām datu kopām

Salīdzinot trīs dotās datu kopas pie J48 ar noklusējuma parametriem *training set* režīmā, redzams, ka *Ionosphere* un *Hyperthyroid* var ļoti labi klasificēt koki ar attiecīgi 18 lapām, 25 virsotnēm, 99.72% un 15 lapām, 25 virsotnēm, 99.58%. Datu kopai *German-credit* koks ar 103 virsotnēm un 140 lapām sasniedz tikai 85.5% precizitāti.

Dažas atšķirības, kas visticamāk apgrūtina koka izveidi pēdējā gadījumā:

1. Daudzi atribūti ir nevis skaitliskas vērtības, bet gan kategoriski/diskrēti lielumi, kas kokā tiek plaši paralēli izplesti. To vērtībām ne vienmēr pastāv skaidras sakarības ar good/bad klasēm, un skaitliskie lielumi tiek jaukti ar kategoriskiem, kas neļauj veidot matemātiskas attiecības un, iespējams, mulsina algoritmu ar troksni;
2. Datu kopa, šķiet, ir par mazu, lai varētu efektīvi atrisināt doto uzdevumu (kam ir ļoti daudzas brīvības pakāpes). Abos pārējos gadījumos pastāv samērā neliels skaits vienkāršu, skaitlisku sakarību, kas skaidri nodala klases pie dotā datu apjoma.