Data mining - assignment 5

Balaganesh Mohan
Bartlomiej Bitner

October 2019

1 problem 1

1.1 Run the Apriori algorithm to generate all frequent itemsets from the 'T10I4D100K' data set at a support thresholds of 0.01%, 0.02% and 0.03%, and report the number of frequent item sets so produced.

Threshold 0.01 - Number of frequent item sets produced : 411365
Threshold 0.02 - Number of frequent item sets produced : 129875
Threshold 0.03 - Number of frequent item sets produced : 112398

1.2 Use the -ts option with Apriori to generate frequent itemsets. Compare the performance of the algorithm in terms of the time taken to produce the results at these thresholds

Threshold 0.01

Reading : 0.10s
Filtering, sorting, and recoding items: 0.01s
Sorting and reducing transactions: 0.04s
Building transaction tree: 0.03s
Checking subsets of size: 2.31s
Writing out: 0.08s

Threshold 0.02

reading: [0.10s].
filtering, sorting and recoding items [0.01s].

sorting and reducing transactions [0.04s].
building transaction tree [0.03s].
checking subsets of size [2.31s].
writing [0.08s].

Threshold 0.03

reading: [0.10s].
filtering, sorting and recoding items [0.01s].
sorting and reducing transactions [0.04s].
building transaction tree [0.03s].
checking subsets of size [1.01s].
writing [0.02s].

1.3 Comment on the possible reason(s) for this difference in performance. You may estimate the amount of time spent by adding up the time displayed by the program when it is executed. Include all the times displayed by the program

As expected, the number of frequent item sets dropped considerably and the time that it took to produce the results was shorter as the support threshold increased. The reason behind it is that when support threshold increases, the frequent item sets which satisfy support threshold are decreasing. Based on the apriori principle, if one set is infrequent, then all its supersets are infrequent and will be pruned.

1.4 Run Apriori (using the -ts option) on the 'mushroom' data set to generate frequent itemsets of sizes 2 through 15 at support thresholds of 5%, 10% and 20%. Generate three plots, one for each threshold, showing the number of frequent sets obtained of size 2 through size 15.

	5%	10%	20%
2	1329	763	376
3	10618	4593	1472
4	48226	16150	3559
5	144928	38800	6267
6	315873	69835	8802
7	527176	98846	10151
8	692740	111786	9488
9	723735	100660	7010
10	600196	71342	4004
11	391578	39171	1729
12	197889	16292	546
13	75624	4956	119
14	21041	1039	16
15	4000	134	1

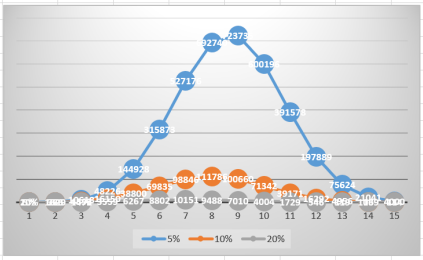

Figure 1: mushroom.dat

1.5 Comment on how the plots vary between the three thresholds.

Looking at the graphs, it can be observed that at the three support thresholds the shape of the graph is shaped, which means that there is a peak in every plot. In fact at any support threshold, there must be one peak value of number of frequent itemsets. The frequent itemsets become greater until size reached around 9 and decreases after 9. This is because apriori algorithm states that if an itemset is either frequent or not frequent, the subset of it will follow the same. Thus all curves are bell curves.

1.6 Use the Apriori algorithm to generate closed (using the -tc option) and maximal (using the -tm option) frequent itemsets from the 'T10I4D100K' and 'mushroom' data sets. Use a support threshold of 5% for the 'mushroom' data set and 0.01% for the 'T10I4D100K' data set. Compare the total number of closed and maximal frequent itemsets obtained for each dataset individually.

	Mushroom	T10104D100k
Closed	12843	283397
Maximal	1442	127264

Figure 2: dat

1.7 How do these numbers compare with the number of frequent itemsets obtained from these data sets using the same threshold? What relationship among closed, maximal and frequent itemsets is revealed by this comparison?

Looking at the table above, the closed frequent itemset is larger than maximal frequent itemset and subsequently the number of frequent itemset is larger than closed frequent itemset. This is because apriori uses and extra constraint when finding the closed frequent itemset and again an extra constraint when finding the maximal frequent itemsets.

2 Problem 2

2.1 Find the probability of Team A winning against Team B (Pr[Won_By_A—Team_B]). Also find the probability of Team A winning against Team C (Pr[Won_By_A—Team_C]). Compare these probabilities.

Pr[Won_By_A—Team_B]: 0.400
Pr[Won_By_A—Team_C]: 0.455

2.2 compare the probabilities of Team A winning against Team B and C at a home venue (Pr[Won_by_A—Team_B,Home] and Pr[Won_by_A—Team_C,Home] respectively). State the rules that you based your comparison upon

Pr[Won_by_A—Team_B,Home]: 0.700 Pr[Won_by_A—Team_C,Home]: 0.677

2.3 Compare the probabilities of Team A winning against Team B and C when the games are played away from A's home (Pr[Won_by_A—Team_B,Away] and Pr[Won_by_A—Team_C,Away] respectively). State the rules that you based your comparison upon.

Pr[Won_by_A—Team_B,Away]: 0.367 Pr[Won_by_A—Team_C,Away]: 0.200

2.4 Are the results in (b) and (c) consistent with those in (a)? Explain why or why not?

The results in b and c are not consistent with those of a because team A plays better when they are at home than when they are playing away.