Nick Holleran
Day Lecture
ID: 5222260

Project 2

Problem 1:

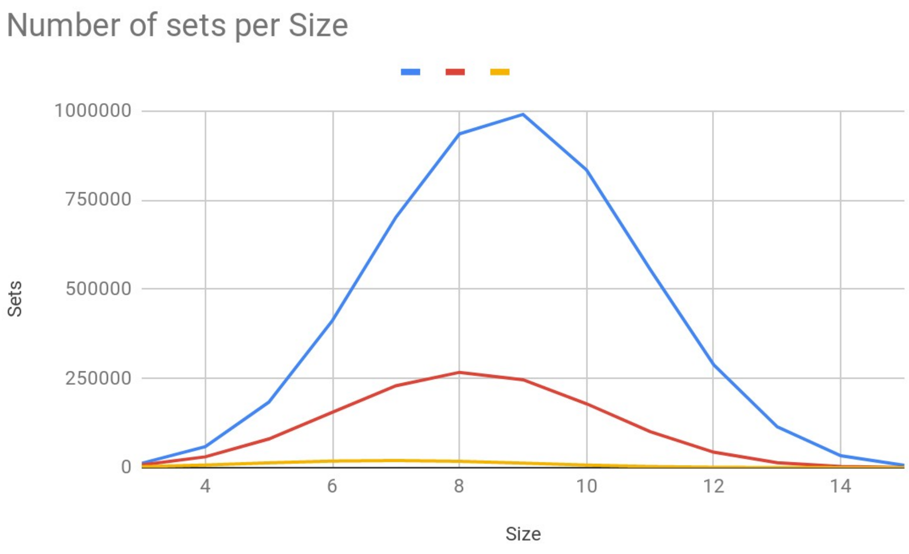
a) .0075 produced 665460 sets, .0275 produced 87323 sets, and .045 produced 57471 sets. The total run-time for each program was 2.28 seconds for .0075, .79 seconds for .0275, and .62 seconds for .045. The run-time for each individual section of each program is listed below:

```
.0075:
        reading T10I4D100K ... [870 item(s), 99936 transaction(s)] done [0.06s].
        filtering, sorting and recoding items ... [867 item(s)] done [0.00s].
        sorting and reducing transactions ... [89081/99936 transaction(s)] done [0.02s].
        building transaction tree ... [112305 node(s)] done [0.01s].
        checking subsets of size 1 2 3 4 5 6 7 8 9 10 11 done [2.02s].
        writing .0075 ... [665460 set(s)] done [0.17s].
.0275:
        reading T10I4D100K ... [870 item(s), 99936 transaction(s)] done [0.06s].
        filtering, sorting and recoding items ... [855 item(s)] done [0.00s].
        sorting and reducing transactions ... [89080/99936 transaction(s)] done [0.02s].
        building transaction tree ... [112323 node(s)] done [0.02s].
        checking subsets of size 1 2 3 4 5 6 7 8 9 10 done [0.67s].
        writing .0275 ... [87323 set(s)] done [0.02s].
.045:
        reading T10I4D100K ... [870 item(s), 99936 transaction(s)] done [0.06s].
        filtering, sorting and recoding items ... [843 item(s)] done [0.00s].
        sorting and reducing transactions ... [89078/99936 transaction(s)] done [0.02s].
        building transaction tree ... [112366 node(s)] done [0.02s].
        checking subsets of size 1 2 3 4 5 6 7 8 9 10 done [0.51s].
        writing .045 ... [57471 set(s)] done [0.01s].
```

The difference in performance for each algorithm comes from the fact that minsup is changing for each one. The lower the minsup is, the more frequent itemsets there are, and the more frequent itemsets there are, the longer it will take to create them all. This is why we see the difference in runtime between a minsup of .0075% and .045%

b)

	4% (Blue)	7% (Red)	14% (Yellow)
3	12241	7562	2652
4	58573	30305	7290
5	183576	80387	13421
6	412314	155016	18376
7	702040	229380	19982
8	935641	267175	17671
9	990279	246437	12617
10	834152	178955	7130
11	555539	100933	3105



12	288484	43269	1001
13	114204	13617	224
14	33249	2965	31
15	6700	399	2

The data above shows that when the minsup is 4%, the number of itemsets based on the size of the itemsets varies a lot, while the number of itemsets for 7% doesn’t vary as much, and 14% barely varies in number of itemsets at all. The 4% minsup has the highest number of itemsets compared to 7% and 14%, with 7% having the second most itemsets between the three of them. This is all because the lower the minsup is, the more frequent itemsets you will have, so since 4% is the lowest of the three minsups, it will have the most itemsets. We also see a gaussian distribution for each minsup, with around size 8 being the maximum. This is because as you increase the width of an itemset you the amount of itemsets will increase until the width is too large and the itemsets become less frequent because the width is so big.

c)

The mushroom example has 936016 frequent itemsets, 7463 closed itemsets, and 912 maximal itemsets. The other file has 96896 frequent itemsets, 84590 closed itemsets, and 36633 maximal itemsets.

Across both data sets we see that there are always more frequent itemsets than closed, and always more closed itemsets than maximal. This is because all closed and maximal itemsets must be frequent, but not all frequent itemsets are closed or maximal. And all maximal itemsets are closed, but not all closed itemsets are maximal. So if we look at a data set, first we prune to get the frequent itemsets, then we prune to get the closed itemsets, and then we prune to get the maximal itemsets. There will never be more maximal itemsets than closed, and the same is true for closed and frequent itemsets.

Problem 2:

a)
P(Won_By_A|Team_B) = 5 <- 1 (47.619, 40)
P(Won_By_A|Team_C) = 5 <- 2 (52.381, 45.4545)

The confidence is listed on the right of the ordered pair, and the confidence in this data is also the probability. This means that the probability of Team A winning against Team B is 40%. The probability of Team A winning against Team C is 545.4545%. This means that Team A is more likely to win against team C than team B but team A kinda sucks since they have a losing record against both teams.

b)
5 <- 3 1 (4.7619, 70)
5 <- 3 2 (28.5714, 66.6667)

Again the confidence for these rules is also the probability so:
P(Won_By_A|Team_B,Home) = 70%
P(Won_By_A|Team_C,Home) = 66.6667%

c)
5 <- 1 4 (42.8571, 36.6667)
5 <- 2 4 (23.8095, 20)

Again the confidence for these rules is also the probability so:
P(Won_By_A|Team_B,Away) = 36.6667%
P(Won_By_A|Team_C,Away) = 20%

d) Given the results in b) and c), Team B is more likely to lose since Team A has a higher win percentage against them in both Home and Away games, this is not consistent with the results found in 1. This is because the support isn't high enough for some of the Home and Away cases which means that the more we define won_by_A, the more skewed the probability is with such a low support.

Problem 3:

a)

Support (in %):
4 (0.466667)
3 (0.466667)
5 (0.466667)
6 (0.666667)
2 (35.3333)

b)

At a minsup of 15%, the only item set that appears is: 2 1 (35.3333). And there are no frequent item sets that appear at a minsup of 40%.

c)

The minsup that gets all the items in at least 1 itemset is .4 (or .46) which can be found if we use the solution in a). If we look at the support for each item in a), then we see if the minsup is any larger than .46 than we will not find any item sets of size 2 with those numbers since they don't even appear that often in the entire data set. The itemsets that of size 2 that appear with a minsup of .46 are:

4 1 (0.466667)
3 1 (0.466667)
5 1 6 (0.466667)
5 1 (0.466667)
5 6 (0.466667)
6 1 (0.666667)
2 1 (35.3333)

d) The results from hyperclique are

4 3 (0.4%, 92.9%)
5 6 (0.5%, 70.0%)
2 1 (35.3%, 35.3%)

These results show that item set 4 3 and 5 6 are frequent as well as 2 1. The itemset 4 3 doesn't appear in the apriori example since it has a support of .43, while the other supports are higher. This example shows only 3 itemsets and does not include the itemset of size 3. Regardless this shows that 4 and 3 items appear frequently together, as do 5 and 6. But 2 and 1 appear very very frequently compared to the others.

e)

1) There are no item overlaps in hyperclique (hc) while in apriori (ap) overlaps frequently with item 1, since item 1 appears in all the itemsets. Ap also includes subsets of the size 3 itemset and hc doesn't even include the size 3 itemset. This shows that 'headline' appears in every article, and so it will appear frequently with every item, and it just so happens that 'writer' appears the second most often. Since Hong Kong is the name of a city it makes sense that they would appear together frequently, the reason this has a lower support than 'hong' or 'kong' paired with headline may be because there is an article that mentions someone named 'kong' or something similar. Since the support has such a low difference the two words likely appeared on their own very rarely. Puerto Rico is also a name of a place, and so it would make sense that the two are paired together frequently, and it would also make sense that they don't appear with 'hong' or 'kong' since there is no such thing as Puerto Kong or Hong Rico.

2) Hc has the frequency for 5 and 6 at .5, which is most likely from rounding up, which makes the frequency the same in both ap and hc. The frequency for 3 4, however, is around .43 which means that the minsup in ap won’t include it. 1 appears in every set so the only thing that affects the frequency in 1 2 is the frequency of 2, which is the same across both hc and ap.

3) I went over this a little in 1) but basically ‘Headline’ will appear with every other word since its in every article. We don’t see ‘Hong’ and ‘Kong’ paired with ‘Puerto’ and ‘Rico since they are different places, but since they are places when paired with each other (3 4 and 5 6) they appear frequently. ‘Writer’ is also a common phrase in articles so it appears in many item sets, it just does not appear in enough item sets to be as frequent with each word as ‘headline’ is, so it only appears in 1 frequent item set/ hyperclique. The hyperclique only includes one item set with headline however, and it appears in every set so it would be a frequent itemset with all other items too as we see with the frequent itemsets.