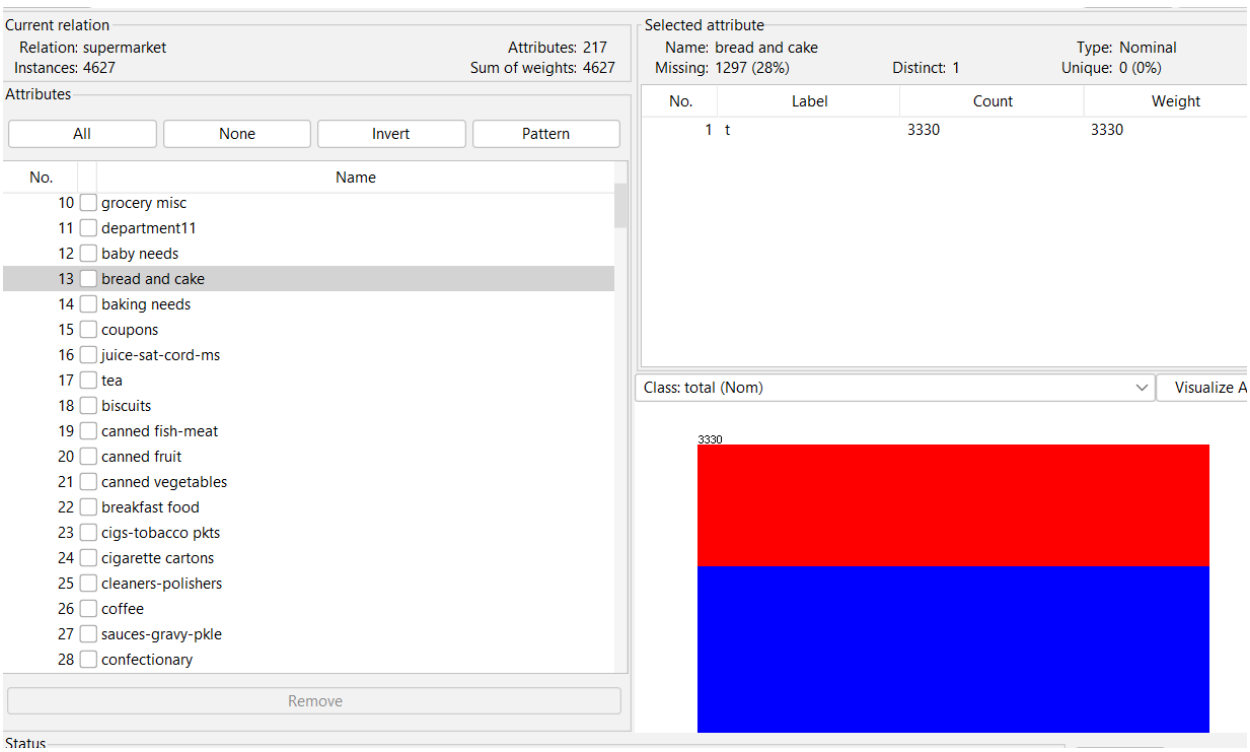
CO544 – Machine Learning and Data Mining

Lab 06 – Part 4

E/17/407

WIJESOORIYA H.D

(01) transactions and their respective attribute values



(03)

LowerBoundMiniSupport : this is the lower bound for minimum support. This parameter is used to exclude rules in the result that have a support lower than the minimum support.

UpperBoundMiniSupport : this is the upper bound for minimum support. Start reducing the minimum support value from this value iteratively.

Delta: Iteratively decrease support by this factor. Reduces support until the needed number of rules are created or the minimum support is attained.

NumRules: the number of rules to find.

metricType: this can be “Confidence”, “Lift”, “Leverage” and “Conviction”. This tells us how we rank the association rules. Generally, Confidence is chosen.

(04) Apriori with default values

```
Minimum support: 0.15 (694 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 17

Generated sets of large itemsets:

Size of set of large itemsets L(1): 44

Size of set of large itemsets L(2): 380

Size of set of large itemsets L(3): 910

Size of set of large itemsets L(4): 633

Size of set of large itemsets L(5): 105

Size of set of large itemsets L(6): 1
```

```
Best rules found:

 1. biscuits=t frozen foods=t fruit=t total=high 788 ==> bread and cake=t 723    <conf:(0.92)> lift:(1.27) lev:(0.03) [155] conv:(3.35)
 2. baking needs=t biscuits=t fruit=t total=high 760 ==> bread and cake=t 696    <conf:(0.92)> lift:(1.27) lev:(0.03) [149] conv:(3.28)
 3. baking needs=t frozen foods=t fruit=t total=high 770 ==> bread and cake=t 705    <conf:(0.92)> lift:(1.27) lev:(0.03) [150] conv:(3.27)
 4. biscuits=t fruit=t vegetables=t total=high 815 ==> bread and cake=t 746    <conf:(0.92)> lift:(1.27) lev:(0.03) [159] conv:(3.26)
 5. party snack foods=t fruit=t total=high 854 ==> bread and cake=t 779    <conf:(0.91)> lift:(1.27) lev:(0.04) [164] conv:(3.15)
 6. biscuits=t frozen foods=t vegetables=t total=high 797 ==> bread and cake=t 725    <conf:(0.91)> lift:(1.26) lev:(0.03) [151] conv:(3.06)
 7. baking needs=t biscuits=t vegetables=t total=high 772 ==> bread and cake=t 701    <conf:(0.91)> lift:(1.26) lev:(0.03) [145] conv:(3.01)
 8. biscuits=t fruit=t total=high 954 ==> bread and cake=t 866    <conf:(0.91)> lift:(1.26) lev:(0.04) [179] conv:(3)
 9. frozen foods=t fruit=t vegetables=t total=high 834 ==> bread and cake=t 757    <conf:(0.91)> lift:(1.26) lev:(0.03) [156] conv:(3)
10. frozen foods=t fruit=t total=high 969 ==> bread and cake=t 877    <conf:(0.91)> lift:(1.26) lev:(0.04) [179] conv:(2.92)
```

Rule 1: If the customer buys biscuits, frozen foods, fruit, and the total price is high then he or she will definitely buy bread and cake. (confidence=92%)

Rule 2: If the customer buys baking needs, biscuits, fruit, and the total price is high then he or she will definitely buy bread and cake. (confidence=92%)

Rule 3: If the customer buys baking needs, frozen foods, fruit, and the total price is high then he or she will definitely buy bread and cake. (confidence=92%)

Rule 4: If the customer buys biscuits, fruit, vegetables and the total price is high then he or she will definitely buy bread and cake. (confidence=92%)

Rule 5: If the customer buys party snack foods, fruits and the total price is high then he or she will definitely buy bread and cake. (confidence=91%)

Rule 6: If the customer buys biscuits, frozen foods, vegetables and the total price is high then he or she will definitely buy bread and cake. (confidence=91%)

Rule7: If the customer buys baking needs, biscuits, vegetables, and the total price is high then he or she will definitely buy bread and cake. (confidence=91%)

Rule 8: If the customer buys biscuits, fruit, and the total price is high then he or she will definitely buy bread and cake. (confidence=91%)

Rule 9: If the customer buys frozen foods, fruit and the total price is high then he or she will definitely buy bread and cake. (confidence=91%)

Rule 10: If the customer buys baking needs, biscuits, fruit, and the total price is high then he or she will definitely buy bread and cake. (confidence=91%)

(05) Apriori with one less and one grater than the default number of rules.

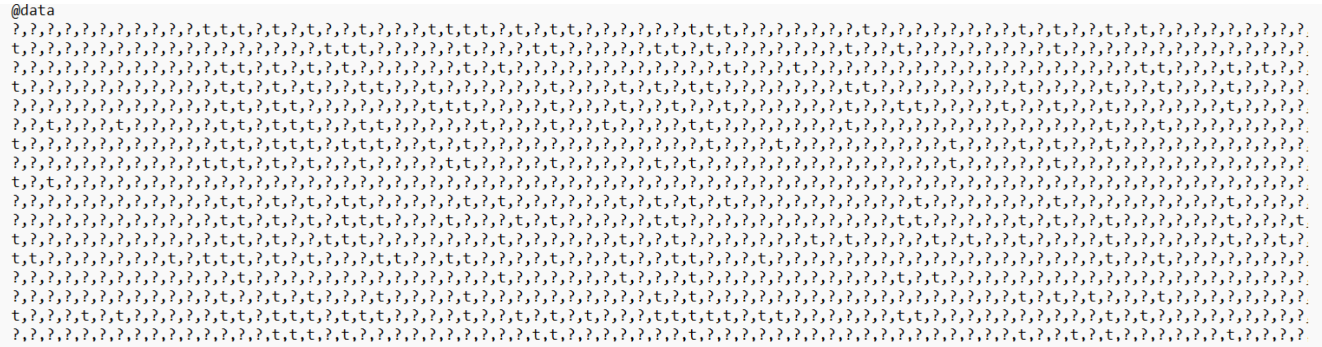
```
Apriori
=======

Minimum support: 0.15 (694 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 17

Generated sets of large itemsets:

Size of set of large itemsets L(1): 44

Size of set of large itemsets L(2): 380

Size of set of large itemsets L(3): 910

Size of set of large itemsets L(4): 633

Size of set of large itemsets L(5): 105

Size of set of large itemsets L(6): 1

Best rules found:

1. biscuits=t frozen foods=t fruit=t total=high 788 ==> bread and cake=t 723    <conf:(0.92)> lift:(1.27) lev:(0.03) [155] conv:(3.35)
2. baking needs=t biscuits=t fruit=t total=high 760 ==> bread and cake=t 696    <conf:(0.92)> lift:(1.27) lev:(0.03) [149] conv:(3.28)
3. baking needs=t frozen foods=t fruit=t total=high 770 ==> bread and cake=t 705    <conf:(0.92)> lift:(1.27) lev:(0.03) [150] conv:(3.27)
4. biscuits=t fruit=t vegetables=t total=high 815 ==> bread and cake=t 746    <conf:(0.92)> lift:(1.27) lev:(0.03) [159] conv:(3.26)
5. party snack foods=t fruit=t total=high 854 ==> bread and cake=t 779    <conf:(0.91)> lift:(1.27) lev:(0.04) [164] conv:(3.15)
6. biscuits=t frozen foods=t vegetables=t total=high 797 ==> bread and cake=t 725    <conf:(0.91)> lift:(1.26) lev:(0.03) [151] conv:(3.06)
7. baking needs=t biscuits=t vegetables=t total=high 772 ==> bread and cake=t 701    <conf:(0.91)> lift:(1.26) lev:(0.03) [145] conv:(3.01)
8. biscuits=t fruit=t total=high 954 ==> bread and cake=t 866    <conf:(0.91)> lift:(1.26) lev:(0.04) [179] conv:(3)
9. frozen foods=t fruit=t vegetables=t total=high 834 ==> bread and cake=t 757    <conf:(0.91)> lift:(1.26) lev:(0.03) [156] conv:(3)
```

```
Apriori
=======

Minimum support: 0.15 (694 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 17

Generated sets of large itemsets:

Size of set of large itemsets L(1): 44

Size of set of large itemsets L(2): 380

Size of set of large itemsets L(3): 910

Size of set of large itemsets L(4): 633

Size of set of large itemsets L(5): 105

Size of set of large itemsets L(6): 1

Best rules found:

 1. biscuits=t frozen foods=t fruit=t total=high 788 ==> bread and cake=t 723    <conf:(0.92)> lift:(1.27) lev:(0.03) [155] conv:(3.35)
 2. baking needs=t biscuits=t fruit=t total=high 760 ==> bread and cake=t 696    <conf:(0.92)> lift:(1.27) lev:(0.03) [149] conv:(3.28)
 3. baking needs=t frozen foods=t fruit=t total=high 770 ==> bread and cake=t 705    <conf:(0.92)> lift:(1.27) lev:(0.03) [150] conv:(3.27)
 4. biscuits=t fruit=t vegetables=t total=high 815 ==> bread and cake=t 746    <conf:(0.92)> lift:(1.27) lev:(0.03) [159] conv:(3.26)
 5. party snack foods=t fruit=t total=high 854 ==> bread and cake=t 779    <conf:(0.91)> lift:(1.27) lev:(0.04) [164] conv:(3.15)
 6. biscuits=t frozen foods=t vegetables=t total=high 797 ==> bread and cake=t 725    <conf:(0.91)> lift:(1.26) lev:(0.03) [151] conv:(3.06)
 7. baking needs=t biscuits=t vegetables=t total=high 772 ==> bread and cake=t 701    <conf:(0.91)> lift:(1.26) lev:(0.03) [145] conv:(3.01)
 8. biscuits=t fruit=t total=high 954 ==> bread and cake=t 866    <conf:(0.91)> lift:(1.26) lev:(0.04) [179] conv:(3)
 9. frozen foods=t fruit=t vegetables=t total=high 834 ==> bread and cake=t 757    <conf:(0.91)> lift:(1.26) lev:(0.03) [156] conv:(3)
10. frozen foods=t fruit=t total=high 969 ==> bread and cake=t 877    <conf:(0.91)> lift:(1.26) lev:(0.04) [179] conv:(2.92)
11. baking needs=t fruit=t vegetables=t total=high 831 ==> bread and cake=t 752    <conf:(0.9)> lift:(1.26) lev:(0.03) [153] conv:(2.91)
```

In the last rule set we can see a new rule related to the above task. That is if the customer buys a baking need, fruit, vegetable and the total price is high then he or she will definitely buy bread and cake. That rule has a 90% confidence level.

(07) supermarket.arff

```
@relation supermarket
@attribute 'department1' { t}
@attribute 'department2' { t}
@attribute 'department3' { t}
@attribute 'department4' { t}
@attribute 'department5' { t}
@attribute 'department6' { t}
@attribute 'department7' { t}
@attribute 'department8' { t}
@attribute 'department9' { t}
@attribute 'grocery misc' { t}
@attribute 'department11' { t}
@attribute 'baby needs' { t}
@attribute 'bread and cake' { t}
@attribute 'baking needs' { t}
@attribute 'coupons' { t}
@attribute 'juice-sat-cord-ms' { t}
@attribute 'tea' { t}
@attribute 'biscuits' { t}
@attribute 'canned fish-meat' { t}
@attribute 'canned fruit' { t}
@attribute 'canned vegetables' { t}
@attribute 'breakfast food' { t}
@attribute 'cigs-tobacco pkts' { t}
```



```
@data
?,?,?,?,?,?,?,?,?,?,?,t,t,t,?,t,?,t,?,?,t,?,?,?,t,t,t,t,?,t,?,t,t,?,?,?,?,?,?,t,t,t,?,?,?,?,?,?,?,t,?,?,?,?,?,?,?,?,t,?,t,?,?,t,?,t,?,?,?,?,?,?,?,?,?
t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,t,t,t,?,?,?,?,?,t,?,?,?,t,t,?,?,?,?,?,t,t,?,t,?,?,?,?,?,?,?,t,?,?,t,?,?,?,?,?,?,?,?,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?
?,?,?,?,?,?,?,?,?,?,?,?,t,t,?,t,?,t,?,t,?,?,?,?,?,?,t,?,t,?,?,?,?,?,?,?,?,?,?,?,?,t,?,?,?,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,t,t,?,?,?,t,?,t,?,?
t,?,?,?,?,?,?,?,?,?,?,?,t,t,?,t,?,t,?,?,t,t,?,?,t,?,?,?,?,?,?,t,?,?,?,t,?,t,?,t,t,?,?,?,?,?,?,?,t,t,?,?,?,?,?,?,?,?,t,?,?,?,?,t,?,?,t,?,?,?,t,?,?,?,?
?,?,?,?,?,?,?,?,?,?,?,?,t,t,?,t,t,?,?,?,?,?,?,?,t,t,t,?,?,?,?,t,?,?,?,t,?,?,t,?,?,t,?,?,?,?,?,?,t,?,?,t,t,?,?,?,?,t,?,?,t,?,?,t,?,?,?,?,?,?,t,?,?,?,?
?,?,t,?,?,?,t,?,?,?,?,?,t,t,?,t,t,t,?,?,t,t,?,?,?,?,?,t,?,?,?,t,?,?,t,?,?,?,?,t,t,?,?,?,?,?,?,?,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,t,?,?,t,?,?,?,?,?,?,?,?
t,?,?,?,?,?,?,?,?,?,?,?,t,t,?,t,t,t,?,t,t,t,?,?,t,?,t,?,?,?,?,?,?,?,?,?,?,?,?,?,t,?,?,?,t,?,?,?,?,?,?,?,?,?,t,?,?,?,t,?,t,?,?,t,?,?,?,?,?,?,?,?,?,?,?
?,?,?,?,?,?,?,?,?,?,?,t,t,t,?,t,?,t,?,?,t,?,?,?,?,t,t,?,?,?,?,t,?,?,?,?,?,t,?,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,t,?,?,?,?,?,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?
t,?,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?
?,?,?,?,?,?,?,?,?,?,?,?,t,t,?,t,?,t,?,t,t,?,?,?,?,?,t,?,t,?,?,?,?,?,?,t,?,t,?,t,?,t,?,?,?,?,?,?,?,?,?,?,t,?,?,?,?,?,?,?,t,?,?,?,?,?,?,?,?,?,t,?,?,?,?
?,?,?,?,?,?,?,?,?,?,?,?,t,t,?,t,?,t,?,t,t,t,?,?,?,t,?,?,?,t,?,t,?,?,?,?,?,t,t,?,?,?,?,?,?,?,?,?,?,?,?,t,t,?,?,?,?,?,t,?,t,?,?,t,?,?,?,?,?,?,t,?,?,?,t
t,?,?,?,?,?,?,?,?,?,?,?,t,t,?,t,?,?,t,t,t,?,?,?,?,?,?,?,t,?,?,?,?,?,?,t,?,?,t,?,?,?,?,?,?,?,t,?,t,?,?,?,?,t,?,t,?,?,t,?,?,?,?,t,?,?,?,?,?,?,t,?,?,t,?
t,t,?,?,?,?,?,?,?,t,?,t,t,t,?,t,?,t,?,?,?,t,t,?,?,t,t,?,?,?,?,t,?,?,?,t,?,?,t,t,?,?,?,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,t,?,?,t,?,?,?,?,?,?,?,?
?,?,?,?,?,?,?,?,?,?,?,?,?,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,t,?,?,?,?,?,?,t,?,?,?,t,?,?,?,?,?,?,?,?,?,?,?,t,?,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?
?,?,?,?,?,?,?,?,?,?,?,?,t,?,?,t,?,t,?,?,?,t,?,?,?,?,t,?,?,?,?,?,?,?,?,?,?,t,?,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,t,?,t,?,t,?,?,?,t,?,?,?,?,?,?,?,?
t,?,?,?,t,?,t,?,?,?,?,?,t,t,?,t,t,t,?,t,t,t,?,?,?,?,t,?,?,t,?,t,?,t,?,?,?,t,t,t,t,t,?,t,t,?,?,?,?,?,?,t,t,?,?,?,?,?,?,?,?,?,?,t,?,t,?,?,?,?,?,?,?,?,?
?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,t,t,t,?,t,?,?,?,?,?,?,?,?,?,?,t,t,?,?,?,?,?,?,?,t,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,t,?,?,t,?,t,?,?,?,?,?,?,t,?,?,?,?
```

In the ‘supermarket.arff’ file we can see the attributes and their data types as shown in figure 01. . Each attribute is binary and either has a value (“t” for true) or no value (“?” for missing).Apart from that we can also see the data records in the dataset as shown in figure 02.