Sardar Patel Institute of Technology, Mumbai
Department of Electronics and Telecommunication Engineering
B.E. Sem-VII (2022-2023)
OEIT6 - Data Analytics

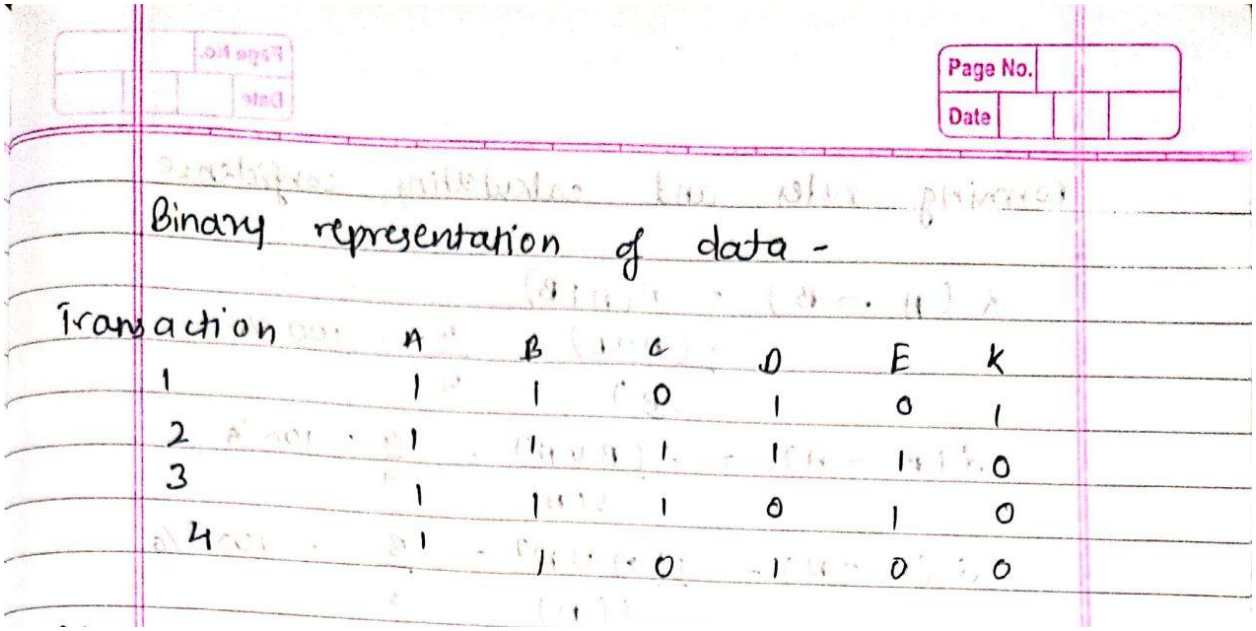
Experiment: Apriori Algorithm

Name: Pushkar Sutar **Roll No. 2019110060**

Objective : Apply Apriori Algorithm to given dataset.

Code and Output:

Exercise 1: Basic association rule creation manually
The 'database' below has four transactions. What association rules can be found in this set, if the minimum support (i.e coverage) is 60% and the minimum confidence (i.e. accuracy) is 80% ?
Trans_id Itemlist
T1 {K, A, D, B}
T2 {D, A C, E, B}
T3 {C, A, B, E}
T4 {B, A, D}
Hint: Make a tabular and binary representation of the data in order to better see the relationship between Items. First generate all item sets with minimum support of 60%. Then form rules and calculate their confidence base on the conditional probability P(B|A) = |B∩A| / |A|. Remember to only take the item sets from the previous phase whose support is 60% or more.



Given → minimum support = 60%

L1 →

Support counts for the itemsets

Itemset	A	B	C	D	E	K
Support count	{3}	{3}	{2}	{3}	{2}	{1}
Support	4/4 = 100%	100%	50%	75%	50%	25%

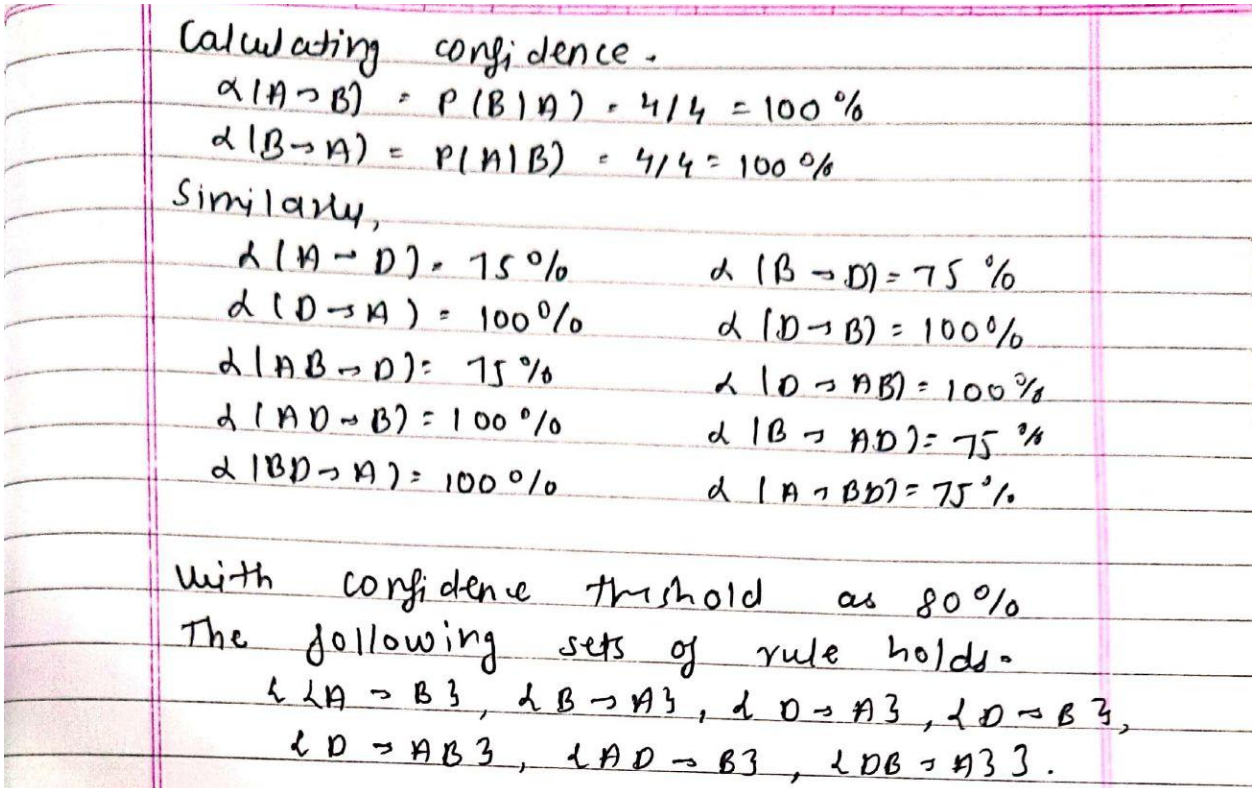
Itemsets with support > min. support are A, B & D.

L2 →

Itemset	A B	A D	B D
Support count	{3}	{3}	{3}
Support	100%	75%	75%

L3 →

Itemset	A B D
Support count	{3}
Support	75%

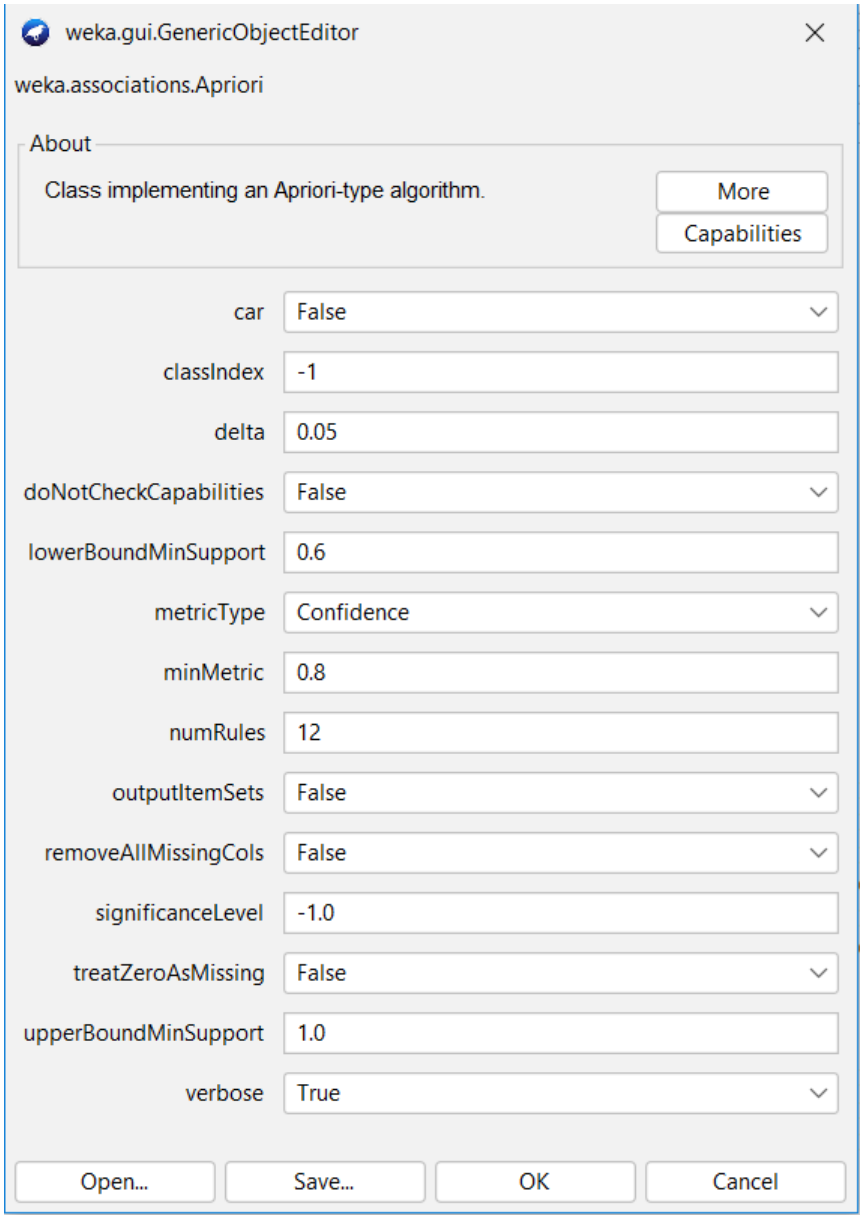


Exercise 2 : Input file generation and Initial experiments with Weka's association rule discovery.
1. Launch Weka and try to do the calculations you performed manually in the previous exercise.Use
the apriori algorithm for generating the association rules.
The file may be given to Weka in e.g. two different formats. They are called ARFF (attribute-relation
file format) and CSV (comma separated values). Both are given below:
ARFF:
@relation exercise
@attribute exista {TRUE, FALSE}
…
@data
TRUE,TRUE,FALSE,TRUE,FALSE,TRUE
… …
CSV:
exista,existb,existc,existd,existe,existk
TRUE,TRUE,FALSE,TRUE,FALSE,TRUE
… …

Relation: data

No.	1: Trans_id Nominal	2: exista Nominal	3: existb Nominal	4: existc Nominal	5: existd Nominal	6: existe Nominal	7: **existk** Nominal
1	T1	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE
2	T2	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE
3	T3	TRUE	TRUE	TRUE	FALSE	TRUE	FALSE
4	T4	TRUE	TRUE	FALSE	TRUE	FALSE	FALSE

2. Once Data is loaded Click Associate Tab on top of the window.



3. Left click the field of Associator, choose Show Property from the drop down list. The property

the window of Apriori opens.
4. Weka runs an Apriori-type algorithm to find association rules, but this algorithm is not exact the same one as we discussed in class.
a. The min. support is not fixed. This algorithm starts with min. support as upperBoundMinSupport (default 1.0 = 100%), iteratively decrease it by delta (default 0.05 = 5%). Note that upperBoundMinSupport is decreased by delta before the basic Apriori algorithm is run for the first time.
b. The algorithm stops when lowerBoundMinSupport (default 0.1 = 10%) is reached, or required number of rules – numRules (default value 10) have been generated.
c. c. Rules generated are ranked by metricType (default Confidence). Only rules with score higher than minMetric (default 0.9 for Confidence) are considered and delivered as the output.
d. If you choose to show the all frequent itemsets found, outputItemSets should be set as True.
5. Click the Start button on the left of the window, the algorithm begins to run. The output is showing in the right window.
Output -

```
Apriori
=======

Minimum support: 0.85 (3 instances)
Minimum metric <confidence>: 0.8
Number of cycles performed: 3

Generated sets of large itemsets:

Size of set of large itemsets L(1): 4

Size of set of large itemsets L(2): 5

Size of set of large itemsets L(3): 2

Best rules found:

 1. existb=TRUE 4 ==> exista=TRUE 4    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
 2. exista=TRUE 4 ==> existb=TRUE 4    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
 3. existd=TRUE 3 ==> exista=TRUE 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
 4. existk=FALSE 3 ==> exista=TRUE 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
 5. existd=TRUE 3 ==> existb=TRUE 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
 6. existk=FALSE 3 ==> existb=TRUE 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
 7. existb=TRUE existd=TRUE 3 ==> exista=TRUE 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
 8. exista=TRUE existd=TRUE 3 ==> existb=TRUE 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
 9. existd=TRUE 3 ==> exista=TRUE existb=TRUE 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
10. existb=TRUE existk=FALSE 3 ==> exista=TRUE 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
11. exista=TRUE existk=FALSE 3 ==> existb=TRUE 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
12. existk=FALSE 3 ==> exista=TRUE existb=TRUE 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
```

Did you succeed? Are the results the same as in your calculations? What kind of file did you use as input?
Yes I succeeded. The results are the same as my calculations. I used a csv file as an input.

Exercise 3: Mining Association Rule with WEKA Explorer – Weather dataset

1. To get a feel for how to apply Apriori to prepared data set, start by mining association rules from the weather.nominal.arff data set of Lab One. Note that Apriori algorithm expects data that is purely nominal: If present, numeric attributes must be discretized first.
2. Like in the previous example choose Associate and Click Start button on the left of the window, the algorithm begins to run. The output is showing in the right window.
3. You could re-run Apriori algorithm by selecting different parameters, such as lowerBoundMinSupport, minMetric (min. confidence level), and different evaluation metric (confidence vs. lift), and so on.

Default parameters:
Minimum Support 0.15, Confidence 0.9

```
Apriori
=======

Minimum support: 0.15 (2 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 17

Generated sets of large itemsets:

Size of set of large itemsets L(1): 12

Size of set of large itemsets L(2): 47

Size of set of large itemsets L(3): 39

Size of set of large itemsets L(4): 6

Best rules found:

 1. outlook=overcast 4 ==> play=yes 4    <conf:(1)> lift:(1.56) lev:(0.1) [1] conv:(1.43)
 2. temperature=cool 4 ==> humidity=normal 4    <conf:(1)> lift:(2) lev:(0.14) [2] conv:(2)
 3. humidity=normal windy=FALSE 4 ==> play=yes 4    <conf:(1)> lift:(1.56) lev:(0.1) [1] conv:(1.43)
 4. outlook=sunny play=no 3 ==> humidity=high 3    <conf:(1)> lift:(2) lev:(0.11) [1] conv:(1.5)
 5. outlook=sunny humidity=high 3 ==> play=no 3    <conf:(1)> lift:(2.8) lev:(0.14) [1] conv:(1.93)
 6. outlook=rainy play=yes 3 ==> windy=FALSE 3    <conf:(1)> lift:(1.75) lev:(0.09) [1] conv:(1.29)
 7. outlook=rainy windy=FALSE 3 ==> play=yes 3    <conf:(1)> lift:(1.56) lev:(0.08) [1] conv:(1.07)
 8. temperature=cool play=yes 3 ==> humidity=normal 3    <conf:(1)> lift:(2) lev:(0.11) [1] conv:(1.5)
 9. outlook=sunny temperature=hot 2 ==> humidity=high 2    <conf:(1)> lift:(2) lev:(0.07) [1] conv:(1)
10. temperature=hot play=no 2 ==> outlook=sunny 2    <conf:(1)> lift:(2.8) lev:(0.09) [1] conv:(1.29)
```

Decreasing the confidence to 0.6-

```
Apriori
=======

Minimum support: 0.3 (4 instances)
Minimum metric <confidence>: 0.6
Number of cycles performed: 14

Generated sets of large itemsets:

Size of set of large itemsets L(1): 12

Size of set of large itemsets L(2): 9

Size of set of large itemsets L(3): 1

Best rules found:

 1. outlook=overcast 4 ==> play=yes 4    <conf:(1)> lift:(1.56) lev:(0.1) [1] conv:(1.43)
 2. temperature=cool 4 ==> humidity=normal 4    <conf:(1)> lift:(2) lev:(0.14) [2] conv:(2)
 3. humidity=normal windy=FALSE 4 ==> play=yes 4    <conf:(1)> lift:(1.56) lev:(0.1) [1] conv:(1.43)
 4. humidity=normal 7 ==> play=yes 6    <conf:(0.86)> lift:(1.33) lev:(0.11) [1] conv:(1.25)
 5. play=no 5 ==> humidity=high 4    <conf:(0.8)> lift:(1.6) lev:(0.11) [1] conv:(1.25)
 6. windy=FALSE 8 ==> play=yes 6    <conf:(0.75)> lift:(1.17) lev:(0.06) [0] conv:(0.95)
 7. play=yes 9 ==> humidity=normal 6    <conf:(0.67)> lift:(1.33) lev:(0.11) [1] conv:(1.13)
 8. play=yes 9 ==> windy=FALSE 6    <conf:(0.67)> lift:(1.17) lev:(0.06) [0] conv:(0.96)
 9. temperature=mild 6 ==> humidity=high 4    <conf:(0.67)> lift:(1.33) lev:(0.07) [1] conv:(1)
10. temperature=mild 6 ==> play=yes 4    <conf:(0.67)> lift:(1.04) lev:(0.01) [0] conv:(0.71)
```

The rules with confidence level 0.67 are also included here. Instances such as temperature mild and humidity high also suggest play as yes.

Changing metric to lift -

```
Apriori
=======

Minimum support: 0.3 (4 instances)
Minimum metric <lift>: 1.1
Number of cycles performed: 14

Generated sets of large itemsets:

Size of set of large itemsets L(1): 12

Size of set of large itemsets L(2): 9

Size of set of large itemsets L(3): 1

Best rules found:

 1. temperature=cool 4 ==> humidity=normal 4    conf:(1) < lift:(2)> lev:(0.14) [2] conv:(2)
 2. humidity=normal 7 ==> temperature=cool 4    conf:(0.57) < lift:(2)> lev:(0.14) [2] conv:(1.25)
 3. play=no 5 ==> humidity=high 4    conf:(0.8) < lift:(1.6)> lev:(0.11) [1] conv:(1.25)
 4. humidity=high 7 ==> play=no 4    conf:(0.57) < lift:(1.6)> lev:(0.11) [1] conv:(1.13)
 5. play=yes 9 ==> outlook=overcast 4    conf:(0.44) < lift:(1.56)> lev:(0.1) [1] conv:(1.07)
 6. play=yes 9 ==> humidity=normal windy=FALSE 4    conf:(0.44) < lift:(1.56)> lev:(0.1) [1] conv:(1.07)
 7. outlook=overcast 4 ==> play=yes 4    conf:(1) < lift:(1.56)> lev:(0.1) [1] conv:(1.43)
 8. humidity=normal windy=FALSE 4 ==> play=yes 4    conf:(1) < lift:(1.56)> lev:(0.1) [1] conv:(1.43)
 9. humidity=normal 7 ==> play=yes 6    conf:(0.86) < lift:(1.33)> lev:(0.11) [1] conv:(1.25)
10. play=yes 9 ==> humidity=normal 6    conf:(0.67) < lift:(1.33)> lev:(0.11) [1] conv:(1.13)
```

We can observe that different sets of rules are accepted here.

Exercise 4: Mining Association Rule with WEKA Explorer – Vote

Now consider a real-world dataset, vote.arff, which gives the votes of 435 U.S. congressmen on 16 key issues gathered in the mid-1980s, and also includes their party affiliation as a binary attribute.
Association-rule mining can also be applied to this data to seek interesting associations.
Load data at the Preprocess tab. Click the Open file button to bring up a standard dialog through which you can select a file. Choose the vote.arff file. To see the original dataset, click the Edit button, a viewer window opens with the dataset loaded. This is a purely nominal dataset with some missing values (corresponding to abstentions).

Task 1. Run Apriori on this data with default settings. Comment on the rules that are generated. Several of them are quite similar. How are their support and confidence values related?

```
Apriori
=======

Minimum support: 0.45 (196 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 11

Generated sets of large itemsets:

Size of set of large itemsets L(1): 20

Size of set of large itemsets L(2): 17

Size of set of large itemsets L(3): 6

Size of set of large itemsets L(4): 1

Best rules found:

 1. adoption-of-the-budget-resolution=y physician-fee-freeze=n 219 ==> Class=democrat 219    <conf:(1)> lift:(1.63) lev:(0.19) [84] conv:(84.58)
 2. adoption-of-the-budget-resolution=y physician-fee-freeze=n aid-to-nicaraguan-contras=y 198 ==> Class=democrat 198    <conf:(1)> lift:(1.63) lev:(0.18) [76] conv:(76.47)
 3. physician-fee-freeze=n aid-to-nicaraguan-contras=y 211 ==> Class=democrat 210    <conf:(1)> lift:(1.62) lev:(0.19) [80] conv:(40.74)
 4. physician-fee-freeze=n education-spending=n 202 ==> Class=democrat 201    <conf:(1)> lift:(1.62) lev:(0.18) [77] conv:(39.01)
 5. physician-fee-freeze=n 247 ==> Class=democrat 245    <conf:(0.99)> lift:(1.62) lev:(0.21) [93] conv:(31.8)
 6. el-salvador-aid=n Class=democrat 200 ==> aid-to-nicaraguan-contras=y 197    <conf:(0.98)> lift:(1.77) lev:(0.2) [85] conv:(22.18)
 7. el-salvador-aid=n 208 ==> aid-to-nicaraguan-contras=y 204    <conf:(0.98)> lift:(1.76) lev:(0.2) [88] conv:(18.46)
 8. adoption-of-the-budget-resolution=y aid-to-nicaraguan-contras=y Class=democrat 203 ==> physician-fee-freeze=n 198    <conf:(0.98)> lift:(1.72) lev:(0.19) [82] conv:(14.62)
 9. el-salvador-aid=n aid-to-nicaraguan-contras=y 204 ==> Class=democrat 197    <conf:(0.97)> lift:(1.57) lev:(0.17) [71] conv:(9.85)
10. aid-to-nicaraguan-contras=y Class=democrat 218 ==> physician-fee-freeze=n 210    <conf:(0.96)> lift:(1.7) lev:(0.2) [86] conv:(10.47)
```

It is seen that most of the candidates are Democrats with as low as 0.9 confidence level.
In the rules the candidate is a Democrat. We can analyse the relation between support and confidence using the parameter lift. The class is Democratic when the budget resolution is adopted, doctor fees are frozen, and no education funding is done. When aid is provided to the Nicaraguan contras but not to El Salvador, the class is a democrat's. Given the following circumstances, the data indicates that the class will be a Democrat with a minimum 98% confidence.

Task 2. It is interesting to see that none of the rules in the default output involve Class = republican. Why do you think that is?
Apriori algorithm bases its rule-making on the frequency of each incident. 267 cases in the provided dataset belong to Democrats, whereas 168 instances belong to Republicans. The class is more likely to be predicted as a Democrat than a Republican due to the bias in the data and the higher frequency of Democrats.

Exercise 5: Let’s run Apriori on another real-world dataset.

Load data at Preprocess tab. Click the Open file button to bring up a standard dialog through which you can select a file. Choose the supermarket.arff file. To see the original dataset, click the Edit button, a viewer window opens with dataset loaded.
To do market basket analysis in Weka, each transaction is coded as an instance of which the attributes represent the items in the store. Each attribute has only one value: If a particular transaction does not contain it (i.e., the customer did not buy that item), this is coded as a missing value.

Task 1. Experiment with Apriori and investigate the effect of the various parameters described before. Prepare a brief oral presentation on the main findings of your investigation.

```
Apriori
=======

Minimum support: 0.3 (1388 instances)
Minimum metric <confidence>: 0.8
Number of cycles performed: 14

Generated sets of large itemsets:

Size of set of large itemsets L(1): 25

Size of set of large itemsets L(2): 69

Size of set of large itemsets L(3): 20

Best rules found:

 1. biscuits=t vegetables=t 1764 ==> bread and cake=t 1487    <conf:(0.84)> lift:(1.17) lev:(0.05) [217] conv:(1.78)
 2. total=high 1679 ==> bread and cake=t 1413    <conf:(0.84)> lift:(1.17) lev:(0.04) [204] conv:(1.76)
 3. biscuits=t milk-cream=t 1767 ==> bread and cake=t 1485    <conf:(0.84)> lift:(1.17) lev:(0.05) [213] conv:(1.75)
 4. biscuits=t fruit=t 1837 ==> bread and cake=t 1541    <conf:(0.84)> lift:(1.17) lev:(0.05) [218] conv:(1.73)
 5. biscuits=t frozen foods=t 1810 ==> bread and cake=t 1510    <conf:(0.83)> lift:(1.16) lev:(0.04) [207] conv:(1.69)
 6. frozen foods=t fruit=t 1861 ==> bread and cake=t 1548    <conf:(0.83)> lift:(1.16) lev:(0.05) [208] conv:(1.66)
 7. frozen foods=t milk-cream=t 1826 ==> bread and cake=t 1516    <conf:(0.83)> lift:(1.15) lev:(0.04) [201] conv:(1.65)
 8. baking needs=t milk-cream=t 1907 ==> bread and cake=t 1580    <conf:(0.83)> lift:(1.15) lev:(0.04) [207] conv:(1.63)
 9. milk-cream=t fruit=t 2038 ==> bread and cake=t 1684    <conf:(0.83)> lift:(1.15) lev:(0.05) [217] conv:(1.61)
10. baking needs=t biscuits=t 1764 ==> bread and cake=t 1456    <conf:(0.83)> lift:(1.15) lev:(0.04) [186] conv:(1.6)
```

Conclusions :

- Apriori is a straightforward algorithm that quickly learns association rules between items (data points).
- While working with very large datasets we should carefully consider the value of minimum support threshold as the algorithm will produce lots of rules for a lower value of minimum support threshold.
- Apriori algorithm is an expensive method to find support since the calculation has to pass through the whole database.
- The outcome of any rule depends on the frequency of the item. Hence we can sometimes get a biassed rule.
- The two factors considered for association rules generation are Minimum Support Threshold and Minimum Confidence Threshold.