# Association Rule Mining: Exercises and Answers

**Contains both theoretical and practical exercises to be done using Weka. The exercises are part of the DBTech Virtual Workshop on KDD and BI.**

## 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}

Read the separate article by Lili Aunimo on association rule generation. You may also read the pages 112 - 117 in Witen, Ian: Practical tools for Data Mining or the articles on Wikipedia on "Association rules" http://en.wikipedia.org/wiki/Association_rules and "Apriori algorithm" http://en.wikipedia.org/wiki/Apriori_algorithm.

**The solution:**

Let's first make a tabular and binary representation of the data:

| Transaction | A | B | C | D | E | K |
|---|---|---|---|---|---|---|
| T1 | 1 | 1 | 0 | 1 | 0 | 1 |
| T2 | 1 | 1 | 1 | 1 | 1 | 0 |
| T3 | 1 | 1 | 1 | 0 | 1 | 0 |
| T4 | 1 | 1 | 0 | 1 | 0 | 0 |

STEP 1. Form the item sets. Let's start by forming the item set containing one item. The number of occurrences and the support of each item set is given after it. In order to reach a minimum support of 60%, the item has to occur in at least 3 transactions.
A 4, 100%
B 4, 100%
C 2, 50%
D 3, 75%
E 2, 50%
K 1, 25%

STEP 2. Now let's form the item sets containing 2 items. We only take the item sets from the previous phase whose support is 60% or more.
A B 4, 100%
A D 3, 75%
B D 3, 75%

STEP 3. The item sets containing 3 items. We only take the item sets from the previous phase whose support is 60% or more.
A B D 3

STEP4. Lets now form the rules and calculate their confidence (c). We only take the item sets from the previous phases whose support is 60% or more.

Rules:
A -> B P(B|A) = |B∩A| / |A| = 4/4, |c: 100%
B -> A c: 100%
A -> D c: 75%
D -> A c: 100%
B -> D c: 75%
D -> B c: 100%
AB -> D c: 75%
D -> AB c: 100%
AD -> B c: 100%
B - > AD c: 75%
BD -> A c: 100%
A -> BD c: 75%

The rules with a confidence measure of 75% are pruned, and we are left with the following rule set:

A -> B
B -> A
D -> A
D -> B
D -> AB
AD-> B
DB-> A

## Exercise 2. Initial experiments with Weka's assiociation rule generation tool.

Launch Weka and try to do with it the calculations you performed manually in the previous exercise. Use the apriori algorithm for generating the association rules.
Did you succeed? Are the results the same as in your calculations? What kind of file did you use as input?

**The Solution:**

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}
@attribute existb {TRUE, FALSE}
@attribute existc {TRUE, FALSE}
@attribute existd {TRUE, FALSE}
@attribute existe {TRUE, FALSE}
@attribute existk {TRUE, FALSE}

@data
TRUE,TRUE,FALSE,TRUE,FALSE,TRUE
TRUE,TRUE,TRUE,TRUE,TRUE,FALSE
TRUE,TRUE,TRUE,FALSE,TRUE,FALSE
TRUE,TRUE,FALSE,TRUE,FALSE,FALSE

CSV format:

exista,existb,existc,existd,existe,existk
TRUE,TRUE,FALSE,TRUE,FALSE,TRUE
TRUE,TRUE,TRUE,TRUE,TRUE,FALSE
TRUE,TRUE,TRUE,FALSE,TRUE,FALSE
TRUE,TRUE,FALSE,TRUE,FALSE,FALSE

The following shows how to launch Weka and what the initial user interface looks like.. In the directory where Weka is installed, type java –jar weka.jar as shown in the Figure 1. Use the Explorer in order to load the file and to try the association rule generator. As you can observe, Weka creates also negative association rules.

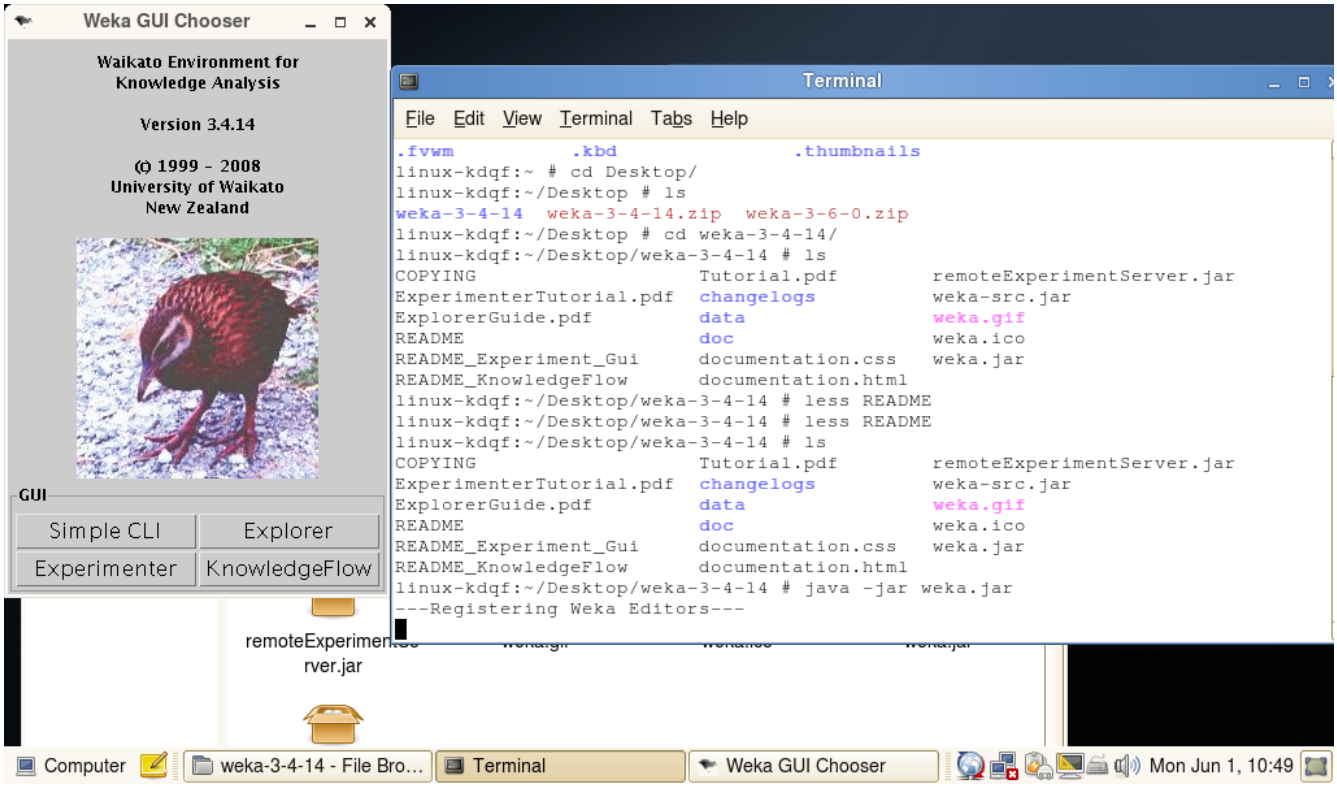


**Figure 1: The first screen of the user interface of Weka.**

## Exercise 3. Weka and the command line parameters of the apriori algorithm.

The apriori algorithm for generating association rules has many command line options. How do you modify these? What do the options mean? Can you modify the options in such a way that you get the same rules as in Exercise 1?

**The Solution**
The options offered are as follows:

Apriori -I N(umRules) 100 -T 0 (metric type is confidence) -C(onfidence) 0.8 -D(elta) 0.5 -U (upperBoundMinSupport) -M (lowerBoundMinSupport) -S (significance level) -1.0 -V(verbose)

delta - iteratively decreases support by this factor. Reduces support until minimum support has been reached or the required number of rules has been generated.
The above presented parameters produce the same results as the one we calculated manually. When the significance level is -1.0, the parameter is not used

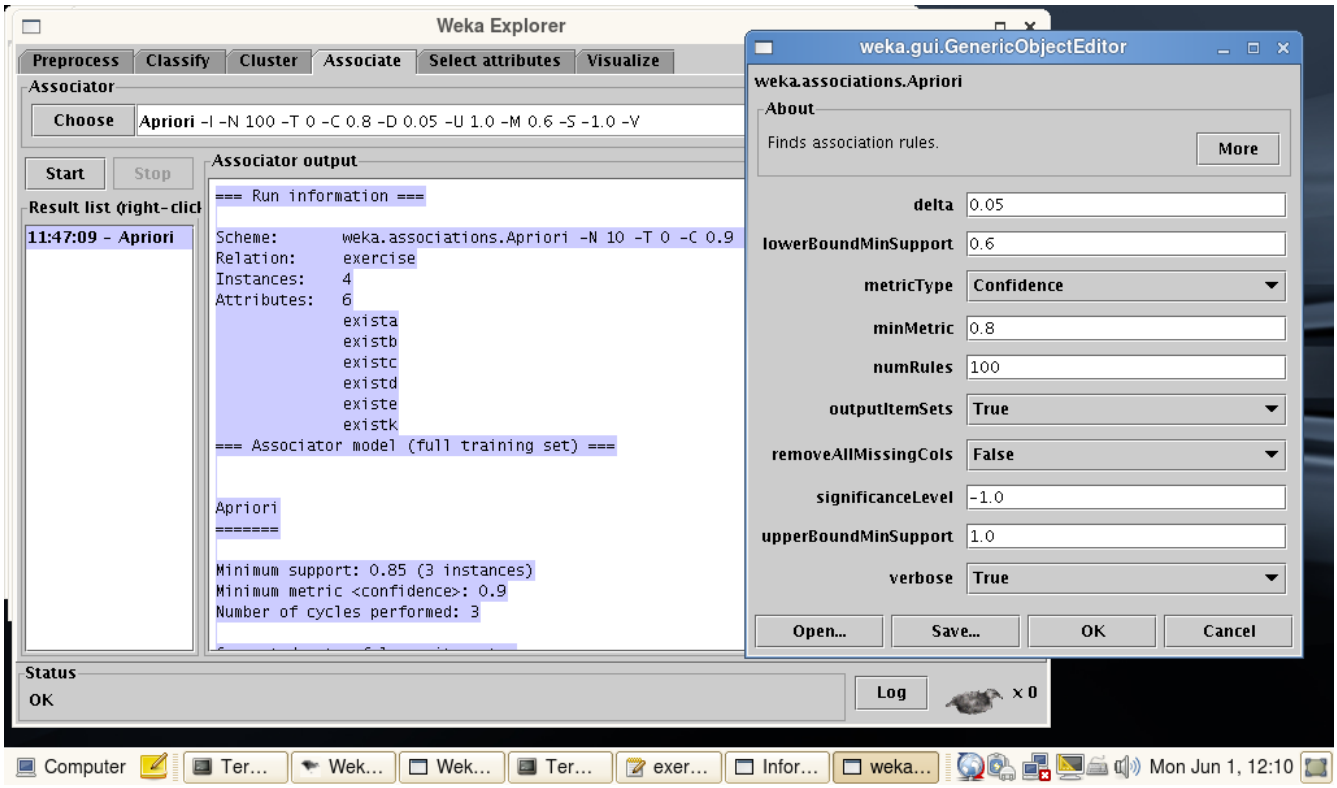


**Figure 2: Running the apriori algorithm in Weka.**

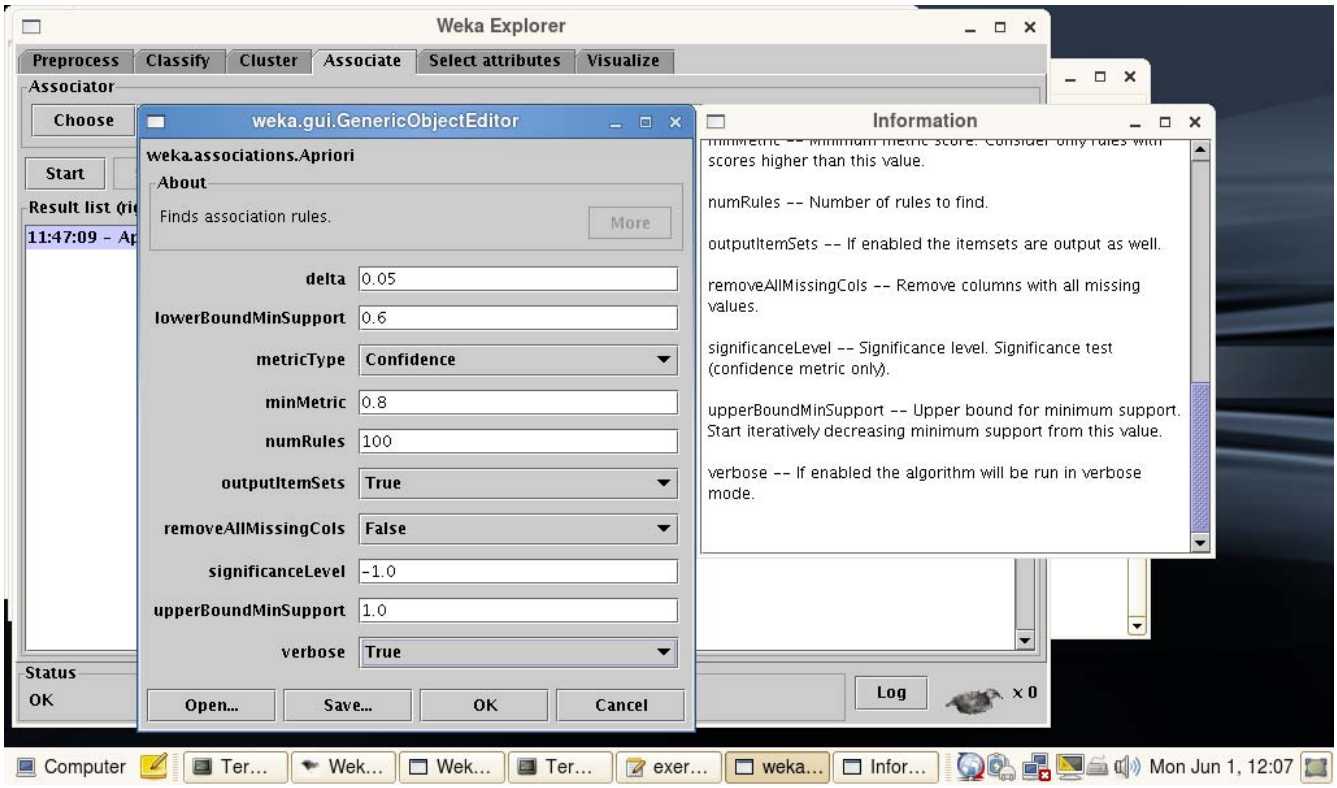


**Figure 3: Setting the parameters of the apriori algorithm. Information about the contents of the parameters may also be found here.**

## Exercise 4. Measures for describing the interestingness of association rules.

In addition to confidence and support, some other measures used to describe association rules are: lift, leverage and conviction. What are these and what do they measure? Calculate these measures for the rules you found in Excercise 1.

## The Solution

Lift. Confidence divided by the proportion of all examples that are covered by the consequence. L(A, B) = c(A,B)/P(B). If this value is 1, then A and B are independent. The higher this value, the more likely that the existence of A and B together in a transaction is not just a random occurrence, but because of some relationship between them.

Leverage. The proportion of additional examples covered by both the premise and the consequence above those expected if the premise and consequence were independent of each other. The equation is: Leverage(A=> B) = P(*A*,*B*) / P(*A*)P(*B*).

Conviction(A=>B) = P(A)P(negation(B))/P(A, negation(B)). It was ntroduced by Brin et al., 1997. Conviction takes the value 1 when A and B have no items in common and it is undefined when the rule A => B always holds.

The vales for the association rules are as follows:

A => B, conf: 1, lift 1/1, conviction: 1*0/0, undefined, Leverage 1/1
B => A, conf: 1, lift 1/1, conviction: undefined, leverage 1
D => A, conf: 1, lift 1/1, conviction: undefined, leverage 0.75/0.75= 1
D => B, conf: 1, lift 1/1, conviction: undefined, leverage 0.75/0.75
AD => B, conf: 1, lift 1/1, conviction: undefined, leverage 0.75/0.75
DB => A, conf: 1, lift 1/1, conviction: undefined, leverage 1
D => AB, conf: 1, lift 1/1, conviction: undefined, leverage 0.75/0.75*1 = 1

**Exercise 5. Data discretization for association rule discovery in Weka.**

Import *the banking dataset* into Weka. The dataset is given in the virtual server environmet. The name of the file is: bank_data.csv. Inspect the data in the preprocessing window of Weka. You can inspect each data field separately by clicking on it. Perform different visualizations on the fields. How is the information given on categorical fields different from that given on continuous fields?

Association rule mining can only be performed on categorical data. Therefore, we have to discretize the continuous data fields. After discretization, perform association rule mining on the dataset. Do you find the rules interesting? Explain why.

**Exercise 6. Data sets for association rule mining.**

Association rule mining suits data sets that have no single category that needs to be predicted. Rather, the technique suits best very large datasets from which unexpected associations between any fields of the data are looked for. Thus, the task is exploratory data analysis. To what kind of datasets are association rules typically applied to? Find such a dataset and perform association rule generation to it. You may consider the datasets that come with the virtual server.

Alternatively, you may think of a dataset of your own, create it, and perform association rule mining on it. In this case the dataset does not have to be very large. The idea is to illustrate a dataset that in real life would be very large.

**Additional resources**

Witten, Ian: Data Mining: Practical Tools and Techniques
KDNuggets, http://www.kdnuggets.com/
McNicholas, P. D. and Zhao, Y. C. (2009), Association rules: An overview, *in* Y. Zhao, C. Zhang & L. Cao, eds, 'Post-Mining of Association Rules: Techniques for Effective Knowledge Extraction', IGI Global, pp. 1-10. Available at https://irma-international.org/downloads/excerpts/33406.pdf
http://maya.cs.depaul.edu/~Classes/Ect584/Weka/preprocess.html