Assignment - 5

[**Association Rule Mining and Application**]

Rahul Midha
B00766975

Niravsinh Jadeja
B00789139

1. Application Scenario

Application Name: Tennis Forecasting Application

This application helps us in deciding whether it is right time to play Tennis, based on the various weather conditions. This application reduces manual calculations and saves us a lot of time. In this application, we have made use of WEKA *[1]* to select the dataset and generated rules in R *[2]* and WEKA. In general context, this dataset has different weather conditions and these conditions specify the probability of playing tennis. The Dataset has 5 columns named “Outlook”, ”Temperature”, ”Humidity”, ”Windy”, ”PlayTennis”. The first column “outlook” has 3 unique elements - sunny, overcast, rain. Other 2 columns - Temperature and Humidity has “hot, mild, and cool” and “high and normal” unique elements respectively. The remaining 2 columns - Windy and PlayTennis has “False and True” and “P (positive/yes) and N (negative/no)” unique element respectively.

2. Data Preprocess

We have taken our dataset from sample data provided in WEKA which is entitled as “weather.nominal” and converted into CSV format for the ease access. The data was cleaned and noise free so we did not have to preprocess it before utilizing for our application.

outlook	temperatu	Humidity	Windy	PlayTennis
sunny	hot	high	FALSE	N
sunny	hot	high	TRUE	N
overcast	hot	high	FALSE	P
rain	mild	high	FALSE	P
rain	cool	normal	FALSE	P
rain	cool	normal	TRUE	N
overcast	cool	normal	TRUE	P
sunny	mild	high	FALSE	N
sunny	cool	normal	FALSE	P
rain	mild	normal	FALSE	P
sunny	mild	normal	TRUE	P
overcast	mild	high	TRUE	P
overcast	hot	normal	FALSE	P
rain	mild	high	TRUE	N

Figure 1: Weather Dataset

3. Association Rule Mining

3.1 Algorithm Description

Association mining *[3]* is related to finding frequent patterns, associations, and regularities between itemsets. In order to do so, one has to find rules which identify these patterns. Here, we have utilized Apriori algorithm *[4]* for association rules generation. It is an algorithm for frequent item set mining and association rule learning over transactional databases. This task is achieved by characterizing the frequent individual items within the dataset and lengthening them to vast item sets as long as those items seem sufficiently within the information set.

These periodic item sets give us association rules which shows common trends in the database. This is one of many application of association mining such as market analysis. It follows bottom-up approach for itemsets generation. It uses breadth-first search and a Hash tree structure to count candidate itemsets efficiently.

The Support is the number which indicates how frequent itemsets appear in the database. The Rules which has support rate greater than a user-defined support is said to have minimum support. The confidence means how often rule has been found true. The rules which have a confidence value greater than a user-defined confidence value is said to have minimum confidence *[5]*.

3.2 Steps for performing association rule mining using WEKA and R

(i) First, we loaded the data in WEKA in CSV format, in which we chose the type of algorithm.

outlook	temperatu	Humidity	Windy	PlayTennis
sunny	hot	high	FALSE	N
sunny	hot	high	TRUE	N
overcast	hot	high	FALSE	P
rain	mild	high	FALSE	P
rain	cool	normal	FALSE	P
rain	cool	normal	TRUE	N
overcast	cool	normal	TRUE	P
sunny	mild	high	FALSE	N
sunny	cool	normal	FALSE	P
rain	mild	normal	FALSE	P
sunny	mild	normal	TRUE	P
overcast	mild	high	TRUE	P
overcast	hot	normal	FALSE	P
rain	mild	high	TRUE	N

Figure 2: Dataset Preview

(ii) Then we have provided the confidence level and minimum support based on the requirements. These two parameters determine the number of rules which we will have at the end of algorithm's execution.

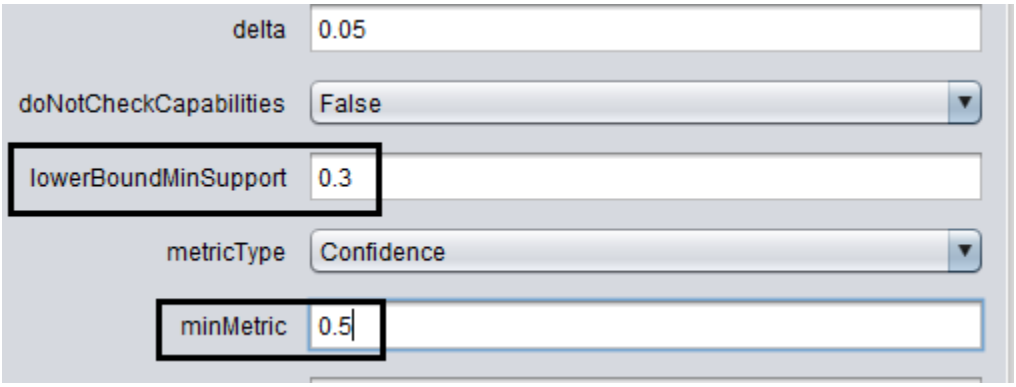

Figure 3: Minimum Support and Confidence Window

(iii) The results we got, which consists of the rules from the dataset were saved in the text format.

```
=== Run information ===

Scheme:       weka.associations.Apriori -N 10 -T 0 -C 0.5 -D 0.05 -U 1.0 -M 0.3 -S -1.0 -c -1
Relation:     1
Instances:    14
Attributes:   5
              outlook
              temperature
              Humidity
              Windy
              PlayTennis
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.3 (4 instances)
Minimum metric <confidence>: 0.5
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 ==> PlayTennis=P 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 ==> PlayTennis=P 4    <conf:(1)> lift:(1.56) lev:(0.1) [1] conv:(1.43)
 4. Humidity=normal 7 ==> PlayTennis=P 6    <conf:(0.86)> lift:(1.33) lev:(0.11) [1] conv:(1.25)
 5. PlayTennis=N 5 ==> Humidity=high 4    <conf:(0.8)> lift:(1.6) lev:(0.11) [1] conv:(1.25)
 6. Windy=FALSE 8 ==> PlayTennis=P 6    <conf:(0.75)> lift:(1.17) lev:(0.06) [0] conv:(0.95)
 7. PlayTennis=P 9 ==> Humidity=normal 6    <conf:(0.67)> lift:(1.33) lev:(0.11) [1] conv:(1.13)
 8. PlayTennis=P 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 ==> PlayTennis=P 4    <conf:(0.67)> lift:(1.04) lev:(0.01) [0] conv:(0.71)
```

Figure 4: Result Preview (WEKA)

(iv) Then we uploaded the .csv file in the R Studio, where we wrote the R Script and gave the file as an input and got the desired results.

```
startTime <- proc.time()
X2 <- read.csv("1.csv")
library(arules)
rules <- apriori(X2,parameter = list(supp = 0.3, conf = 0.5, target = "rules"))
summary(rules)
inspect(rules)
proc.time() - startTime
write(rules, file = "output.csv", sep = ",", quote = TRUE, row.names = FALSE)
```

Figure 5: R Script

(v) The result so obtained are saved in the CSV format.

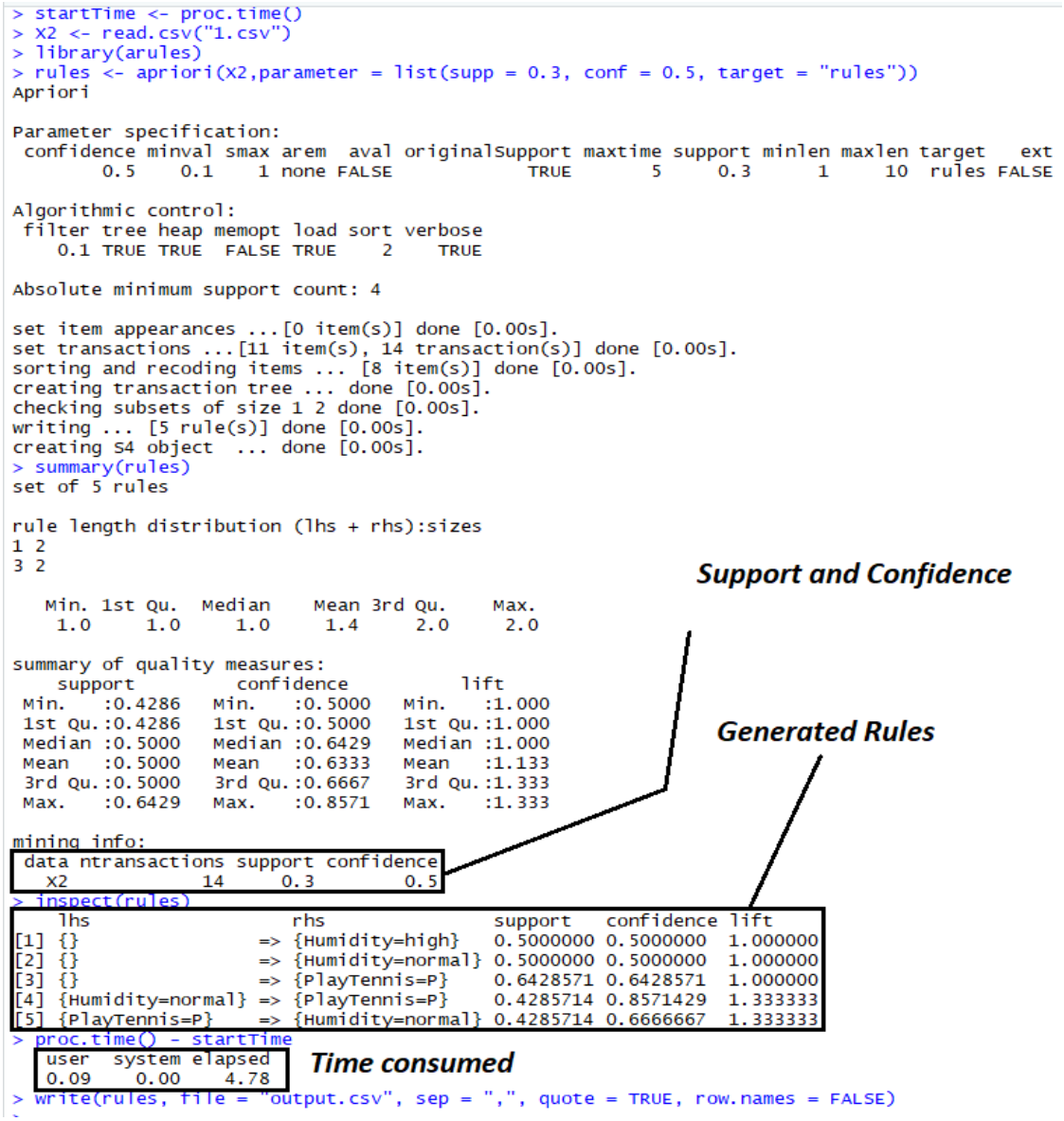

Figure 6 Result Preview (R Studio)

4. Experiment Results (Rules Comparison)

Rules Comparison (WEKA vs R)

```
 1. outlook=overcast 4 ==> PlayTennis=P 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 ==> PlayTennis=P 4    <conf:(1)> lift:(1.56) lev:(0.1) [1] conv:(1.43)
 4. Humidity=normal 7 ==> PlayTennis=P 6    <conf:(0.86)> lift:(1.33) lev:(0.11) [1] conv:(1.25)
 5. PlayTennis=N 5 ==> Humidity=high 4    <conf:(0.8)> lift:(1.6) lev:(0.11) [1] conv:(1.25)
 6. Windy=FALSE 8 ==> PlayTennis=P 6    <conf:(0.75)> lift:(1.17) lev:(0.06) [0] conv:(0.95)
 7. PlayTennis=P 9 ==> Humidity=normal 6    <conf:(0.67)> lift:(1.33) lev:(0.11) [1] conv:(1.13)
 8. PlayTennis=P 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 ==> PlayTennis=P 4    <conf:(0.67)> lift:(1.04) lev:(0.01) [0] conv:(0.71)
```

Figure 7: Generated Rules in WEKA

In WEKA, we have found 10 rules after the execution of algorithms. We can say that if outlook is overcast then chances of playing tennis will be high (P) from the first rule. Adding further, if Humidity is normal and windy condition is false then also chance of playing tennis is high (P) from rule 3. We can also conclude that if the temperature is mild then also playing tennis probability is high (P) from rule 10.

rules	support	confidence	lift
{} => {Humidity=high}	0.5	0.5	1
{} => {Humidity=normal}	0.5	0.5	1
{} => {PlayTennis=P}	0.642857143	0.642857143	1
{Humidity=normal} => {PlayTennis=P}	0.428571429	0.857142857	1.333333333
{PlayTennis=P} => {Humidity=normal}	0.428571429	0.666666667	1.333333333

Figure 8: Generated Rules in R

With the help of R, we have found 5 rules as seen above. We can deduce further that if humidity is normal then the probability of playing tennis will be high and vice versa.

These rules generation depends on support and confidence value, the higher value it is, a lesser number of rules will be generated. In Weka, we have found more rules compared to R, even the support and confidence value was taken in both the cases was same.

Execution Time:

1. WEKA timing : 0.002 Seconds

```
=== Evaluation ===

Elapsed time: 0.002s
```

Figure 9: Execution Time (WEKA)

2. R Timing: 0.00 Seconds (System time), 4.78 Seconds (total elapsed time)

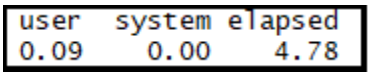

Figure 10: Execution Time (R)

5. Conclusion

In this assignment, we have created a Tennis Forecasting Application. Weka is a very powerful software with which we can load the dataset and generate required results in a short span of time. The loading of data is very easy and minimalistic user actions are required to get the task done.

Similarly, the experience with R was also good. It was easy and user-friendly at the same time. With the minimum number of steps, we can get the desired output. One difficulty we faced while working the application is to find a dataset according to the algorithm. We had to search a lot before finalizing the dataset.

We strongly recommend these software tools for future assignments as they do not require any kind of purchase or registration. These are open source and can be used by anyone without any cost. These softwares are very good for data mining and testing purposes. Alside, utilization of python *[6]* for the data mining and other data science related operations would be a great practice as it has strong market share along with R.

Dataset Link: Link

6. References

[1]W. Learning, "Weka – Graphical User Interference Way To Learn Machine Learning", *Analytics Vidhya*, 2018. [Online]. Available: https://www.analyticsvidhya.com/learning-paths-data-science-business-analytics-business-intelligence-big-data/weka-gui-learn-machine-learning/. [Accessed: 29- Mar- 2018].

[2]"R (programming language)", *En.wikipedia.org*, 2018. [Online]. Available: https://en.wikipedia.org/wiki/R_(programming_language). [Accessed: 29- Mar- 2018].

[3]D. Borne, "Association Rule Mining – Not Your Typical Data Science Algorithm | MapR", *Mapr.com*, 2018. [Online]. Available: https://mapr.com/blog/association-rule-mining-not-your-typical-data-science-algorithm/. [Accessed: 29- Mar- 2018].

[4]R. Jain, "A beginner's tutorial on the apriori algorithm in data mining with R implementation | Machine Learning | HackerEarth Blog", *Hackerearth.com*, 2018. [Online]. Available:

https://www.hackerearth.com/blog/machine-learning/beginners-tutorial-apriori-algorithm-data-mining-r-implementation/. [Accessed: 29- Mar- 2018].

[5] “Ass3 Tutorial: Apriori Algorithm Implementation” Revised by Serikzhan Kazi

[6]"Welcome to Python.org", *Python.org*, 2018. [Online]. Available: https://www.python.org/. [Accessed: 29- Mar- 2018].