Learning outcomes

After solving these exercises, you should be able to understand the following:

1. Reading transaction data and exploring the data and items
2. Implementing association rule mining in R
3. Understanding the computation of support, confidence and lift
4. Interpreting rules and results

I. Using the following store's purchase data and answer the following.

Table 1

Trans ID	Item Purchased		Trans ID	Item Purchased
1	A		4	B
1	B		5	E
2	B		6	A
2	C		6	E
2	D		6	B
4	A		7	D

1. What is the Confidence of {B => C}?
2. What is the support of {A, B}?
3. What is the Confidence of {B => A}?
4. What is the Lift of {B => A}?

II. Association Rules for transaction data :

Steps to follow:

a. Install and load 'arules' package

```
install.packages("arules")
```

b. Read transaction 'Transactions.csv' data in the way arules package should treats the transaction data

```
trans = read.transactions(file="Transactions.csv", rm.duplicates= FALSE,
format="single",sep=",",cols =c(1,2))
```

c. Check the data read format

```
inspect(trans)
```

d. Explore and understand the data and items of transaction data

```
trans
itemFrequency(trans)
itemFrequencyPlot(trans)
```

e. Implementing association mining using 'Apriori' algorithm to extract rules

```
rules <- apriori(trans,parameter = list(sup = 0.5, conf = 0.6,target="rules"),control =
list(verbose=F))
```

f. Understanding the rules

```
inspect(rules)
```

III. Assignment :

Practice above analysis on 'Groceries' data set (in-built data set in R) which has 9835 transactions and 169 items.

```
#To load data
data("Groceries")
Groceries
```

IV. Association Rules for Flight Delays data:

Use file 'flight_delays' data to generate the rules and identify the patterns.

Steps to follow:

1. Read the data into R

```
flight_Delays = read.csv("FlightDelays.csv", header=T)
```

2. Look at the summary of all the variables and convert the following variables as factors
   a. CARRIER
   b. DEST
   c. ORIGIN
   d. Weather
   e. DAY_WEEK
   f. Flight Status

```
cat_Data <- subset(flight_Delays, select=-c(1))
cat_Data <- data.frame(sapply(cat_Data, function(x){as.factor(x)}))
```

3. Bin the numeric variable 'CRS_DEP_TIME' into four bins based on equal frequency or equal width. Or, you may bin them as based on the following criterion :
   If time is less than 6 AM then code it as 1 and if the time is less than 12PM then code it as 2 …

```
time_Bins <- ifelse(flight_Delays$CRS_DEP_TIME < 600, 1,
            ifelse(flight_Delays$CRS_DEP_TIME < 1200, 2,
            ifelse(flight_Delays$CRS_DEP_TIME < 1800, 3, 4)))
```

4. Merge the data from step 2,3.

```
data <- data.frame(time_Bins, cat_Data)
```

5. Convert the data frame in a transactions object. Look at the first 6 transactions to understand

```
flight <- as(data, "transactions")
```

6. Apply 'arules' algorithm and play with various support, lift and confidence values

```
rules <- apriori(flight,
    parameter = list(support = 0.06, confidence = 0.6),control = list(verbose=F))
```

7. Inspect all the rules

8. Filter the rules with specific LHS and RHS conditions

   a) Filter the rules with Flighstatus=0.

```
rules.classfilter1 <- as(subset(rules, subset = rhs %in% "Flight.Status=0"),
                         "data.frame")
```

   b) Filter the rules with Flighstatus=0 and support >.8

9. Sort the rules based upon “lift”,select top 20 of them and then plot them.

```
rules.sorted <- sort(rules,by="lift")
rulesImp <- rules.sorted[1:20]
library(arulesViz)
plot(rulesImp,method="graph",interactive=TRUE,control=list(type="items"))
```