Academic Year: 2025-26 **Semester: V**
Class/Branch: T.E. DS **Subject: DWMLab**

EXPERIMENT NO. 9

1. **Aim:** To apply preprocessing & visualization techniques in R for association rule mining.

2. **Objectives:** From this experiment, the student will be able to

- Learn about R Programming.
- Learn about the association mining algorithm and frequent pattern.
- Learn about Apriori algorithm.

3. **Theory:**

R Programming:

R is a powerful open-source programming language and environment specifically designed for **statistical computing, data analysis, and visualization**. It provides a wide range of packages and functions to handle data preprocessing, perform statistical modeling, and generate graphical representations of data. R is widely used in data mining, machine learning, and exploratory data analysis due to its flexibility and rich ecosystem of libraries.

Association rule mining is a popular data mining technique used to discover interesting patterns, relationships, or correlations between items in large datasets (e.g., market basket analysis).
Preprocessing ensures that the data is clean, formatted, and suitable for analysis, while visualization helps in understanding patterns and interpreting the mined rules effectively. Preprocessing involves cleaning the data, handling missing values, transforming data types, and structuring the dataset so that it is suitable for analysis. Visualization helps in exploring the data, identifying patterns, and understanding relationships between variables, which aids in interpreting the results of association rule mining more effectively.

4. Code:

1. Importing required packages.

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

2. Load libraries.

```
library(arules)
library(arulesViz)
```

3. Create a small transaction dataset.

```
groceries <- list(
  c("milk", "bread", "eggs"),
  c("milk", "bread"),
  c("milk", "soda", "beer"),
  c("bread", "butter"),
  c("milk", "bread", "soda", "beer"),
  c("bread", "butter", "jam")
)
```

4. Convert list into transactions.

```
trans <- as(groceries, "transactions")
```

5. View first 5 transactions to check data.

```
# View first 5 transactions to check data
inspect(trans[1:5])
```

6. Preprocessing:

Step 1: Rename in the list before transactions.

```
groceries <- lapply(groceries, function(x) gsub("soda", "soft_drink", x))
```

```
Step 1: Convert to transactions
trans_clean <- as(groceries, "transactions")
```

Step 2:

```
# Step 2: Filter sparse items (>= 2 occurrences)
trans_up <- trans_clean[, itemFrequency(trans_clean) >=
(2/length(trans_clean))]
```

Step 3:

```
#Inspect

cat("Dimensions after cleaning:\n")
print(dim(trans_up))
inspect(trans_up)
```

Step 4:

```
# Step 4: Item frequencies
itemFrequency(trans_up)
```

7. Apply Apriori algorithm

```
rules <- apriori(
  trans_up,
  parameter = list(supp = 0.2, conf = 0.6, minlen = 2)
)
```

Inspect the rules

```
inspect(rules)
```

8. Visualization

```
itemFrequencyPlot(trans_up, topN = 5, col = "skyblue",
                  main = "Top 5 Items", type = "absolute")
plot(rules, method = "grouped matrix", measure = "lift")
```

5. Conclusion: