**Big Data Spring 2018**

**Project 6**

Yifan Chen

# Databricks notebook source

library(sparklyr)

library(dplyr)

library(arules)

library(arulesViz)

# COMMAND ----------

# open a spark connection with method = databricks

sc <- spark\_connect(method="databricks")

# import csv file into Spark Dataframe

receipt\_list <- spark\_read\_csv(sc, "tbl", "/FileStore/tables/75000i.csv", header = F)

# name the columns in csv file

names(receipt\_list) <- c("Receipt\_Number","Quantity","Food")

# COMMAND ----------

# convert receipt\_list from SparkDataFrame into an R DataFrame

receipt\_df <- data.frame(receipt\_list)

# select columns

receipt\_df\_filter <- receipt\_df[c("Receipt\_Number","Food")]

# coerce the R data.frame to a transactions class

trans <- as(split(receipt\_df\_filter$Food, receipt\_df\_filter$Receipt\_Number), "transactions")

# COMMAND ----------

# question 1

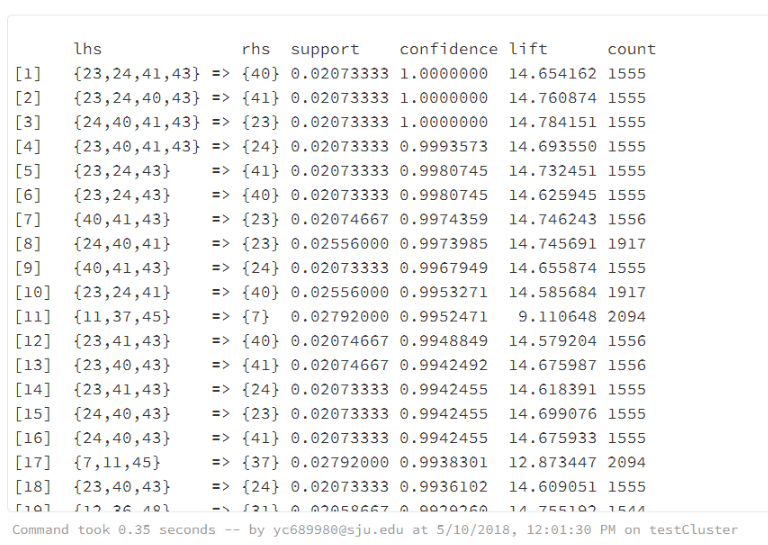
# use aprior() algo to generate ARs

rules<-apriori(trans, control=list(verbose=F), parameter=list(supp=0.015,conf=0.3))

# sort by confidence and limit 100

rules %>% sort(by="confidence") %>% head(100) %>% inspect()

***Result:***



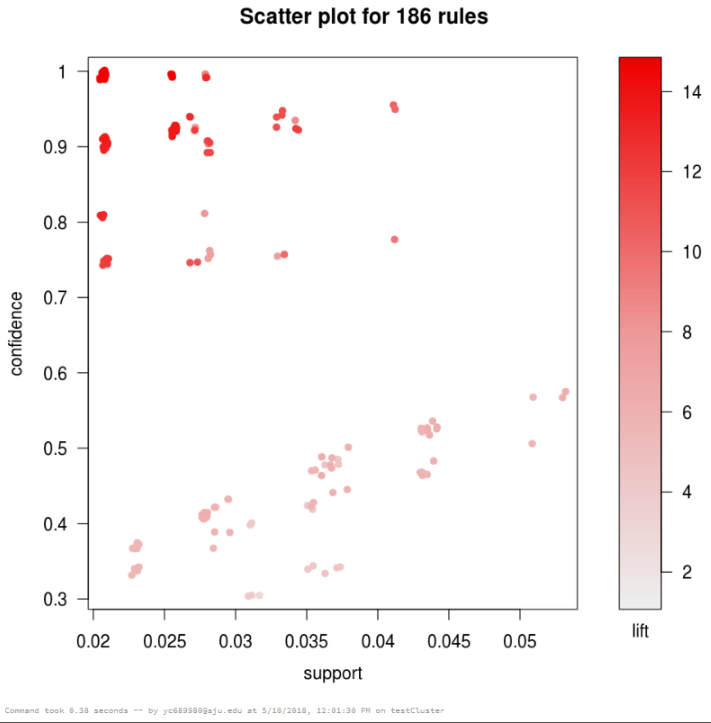
# COMMAND ----------

# question 2

# visualize each arule's support, confidence and lift

plot(rules, measure=c("support","confidence"), shading="lift")

***Result:***



# COMMAND ----------

# question 3

# arules\_number stores number of ARs generated by different confidence settings

arules\_number <- vector(mode="integer")

cnt <- 1

# loop 10 times to get number of ARs with different confidence value

for(i in seq(from=0.1, to=1, by=0.1)){

rules<-apriori(trans,

control=list(verbose=F),

parameter=list(supp=0.001,conf=i))

arules\_number[cnt]=length(rules)

cnt=cnt+1

}

# Graph the number of ARs using blue points overlayed by a line

plot(arules\_number, type="o", col="blue", axes=FALSE)

# specify x,y axes

axis(1, at=1:10, lab = c("0.1","0.2","0.3","0.4","0.5","0.6","0.7","0.8","0.9","1.0"))

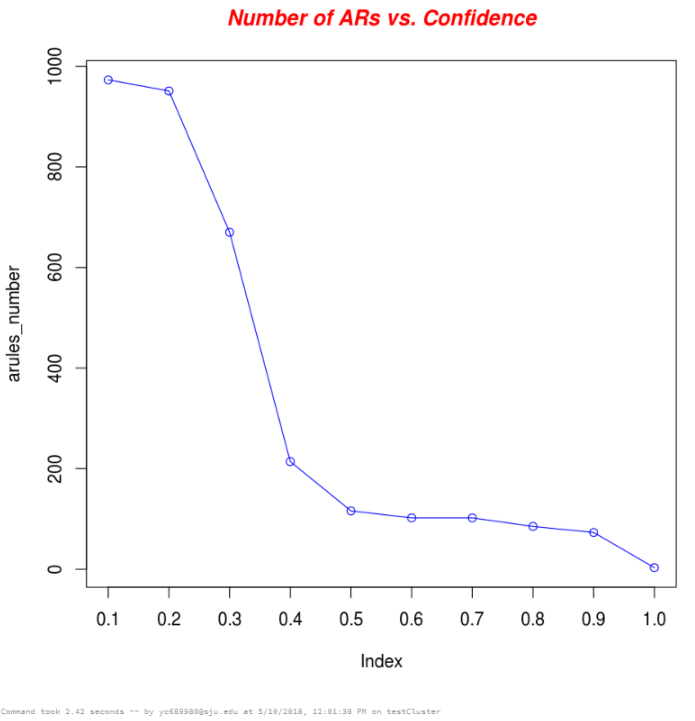
axis(2)

# draw the box of diagram

box()

# create a title with a red, bold/italic font

title(main="Number of ARs vs. Confidence", col.main="red", font.main=4)



# COMMAND ----------

# question 4

# generate 3 frequent itemsets(large itemsets) with different support settings 0.0001, 0.0002 and 0.0003

freq\_items\_1 <- apriori(trans, parameter = list(support = 0.001, target="frequent itemsets"))

freq\_items\_2 <- apriori(trans, parameter = list(support = 0.002, target="frequent itemsets"))

freq\_items\_3 <- apriori(trans, parameter = list(support = 0.003, target="frequent itemsets"))

# Demo

inspect(freq\_items\_1)

# COMMAND ----------

# store the number of the specific itemset number

cnt1 <- rep(0, 50)

cnt2 <- rep(0, 50)

cnt3 <- rep(0, 50)

# the max number of items in itemset

maxLength1 = 0

maxLength2 = 0

maxLength3 = 0

# calculate the number of the specific itemset scan

# freq\_items\_1

for (i in 1:length(freq\_items\_1)) {

iterItemset = items(freq\_items\_1[i])

list\_iterItemset = as(iterItemset, "list")

index = as.integer(lengths(list\_iterItemset))

cnt1[index] = cnt1[index] +1

}

# freq\_items\_2

for (i in 1:length(freq\_items\_2)) {

iterItemset = items(freq\_items\_2[i])

list\_iterItemset = as(iterItemset, "list")

index = as.integer(lengths(list\_iterItemset))

cnt2[index] = cnt2[index] +1

}

# freq\_items\_3

for (i in 1:length(freq\_items\_3)) {

iterItemset = items(freq\_items\_3[i])

list\_iterItemset = as(iterItemset, "list")

index = as.integer(lengths(list\_iterItemset))

cnt3[index] = cnt3[index] +1

}

# Demo

cnt1

# COMMAND ----------

# calculate the largest element number of the large itemsets

size1 = 0

size2 = 0

size3 = 0

for (i in 1:length(cnt1))

if (cnt1[i]!=0) size1=size1+1

else break;

for (i in 1:length(cnt2))

if (cnt2[i]!=0) size2=size2+1

else break;

for (i in 1:length(cnt3))

if (cnt3[i]!=0) size3=size3+1

else break;

# get the large itemsets number for each scan

for (i in 2:size1)

cnt1[i]=cnt1[i-1]+cnt1[i]

for (i in 2:size2)

cnt2[i]=cnt2[i-1]+cnt2[i]

for (i in 2:size3)

cnt3[i]=cnt3[i-1]+cnt3[i]

# COMMAND ----------

new\_cnt1 = rep(0, size1)

new\_cnt2 = rep(0, size2)

new\_cnt3 = rep(0, size3)

for (i in 1:size1)

new\_cnt1[i]=cnt1[i]

for (i in 1:size2)

new\_cnt2[i]=cnt2[i]

for (i in 1:size3)

new\_cnt3[i]=cnt3[i]

# Graph cars using blue points overlayed by a line

plot(new\_cnt1, type="o", col="blue")

lines(new\_cnt2, type="o", col="red")

lines(new\_cnt3, type="o", col="green")

# Create a title with a red, bold/italic font

title(main="Large ItemSets", col.main="red", font.main=4)

legend("topleft", legend=c("minsup=0.1%", "minsup=0.2%","minsup=0.3%"), col=c("blue", "red", "green"), lty=1)

***Result:***

