Association Rules Project

# Load libraries

library(dplyr)

## Warning: package 'dplyr' was built under R version 3.5.2

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(arules)

## Warning: package 'arules' was built under R version 3.5.2

## Loading required package: Matrix

##   
## Attaching package: 'arules'

## The following object is masked from 'package:dplyr':  
##   
## recode

## The following objects are masked from 'package:base':  
##   
## abbreviate, write

library(arulesViz)

## Warning: package 'arulesViz' was built under R version 3.5.2

## Loading required package: grid

# Set the working directory

setwd("C:/Users/Ashley Rainbow/Desktop/Data Fellowship/data-fellowship/Association Rules/Association Rules Project")

# Import the data

orders <- read.csv('Orders.csv', nrows = 35000)

# Check the data

head(orders)

## Order.Number  
## 1 208065104\_20181221134250  
## 2 211995719\_20181224134001  
## 3 212147781\_20181202154311  
## 4 212378740\_20181121114943  
## 5 212630336\_20181120173713  
## 6 212656834\_20181115153649  
## Item.Name Item.SKU  
## 1 TONGUE AND GROOVED LEDGED AND TGLBG  
## 2 Servis DC60W Ceramic Double Ov 10651659  
## 3 Russell Hobbs 60CM G Dishwasher - White RHDW3  
## 4 WHITE KNIGHT 5KG 1000RPM WASH 2276100  
## 5 INDSIT BWE91483KWUK 9KG WM BLK 6699226  
## 6 Hotpoint FFFL2012P Tall Fridge Freezer - White/Ins/Del/Rec. 1704398

# Change item name from a character string to factor.

orders$Item.Name <- as.factor(orders$Item.Name)

# Change item name from factor to character

orders$Order.Number <- as.character(orders$Order.Number)

# Group products by order number

order\_baskets = orders %>%   
 group\_by(Order.Number) %>% summarise(order\_basket = as.vector(list(Item.Name)))

## Warning: package 'bindrcpp' was built under R version 3.5.2

# View a sample order\_basket

head(order\_baskets)

## # A tibble: 6 x 2  
## Order.Number order\_basket  
## <chr> <list>   
## 1 208065104\_20181221134250 <fct [1]>   
## 2 211995719\_20181224134001 <fct [1]>   
## 3 212147781\_20181202154311 <fct [1]>   
## 4 212378740\_20181121114943 <fct [1]>   
## 5 212630336\_20181120173713 <fct [1]>   
## 6 212656834\_20181115153649 <fct [1]>

# Convert to sparse matrix

sparse\_data <- as(order\_baskets$order\_basket, "transactions")

## Warning in asMethod(object): removing duplicated items in transactions

# Inspect the sparse matrix

summary(sparse\_data)

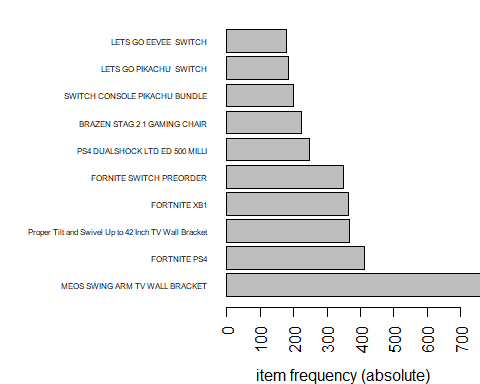
## transactions as itemMatrix in sparse format with  
## 32204 rows (elements/itemsets/transactions) and  
## 7139 columns (items) and a density of 0.0001499406   
##   
## most frequent items:  
## MEOS SWING ARM TV WALL BRACKET   
## 772   
## FORTNITE PS4   
## 413   
## Proper Tilt and Swivel Up to 42 Inch TV Wall Bracket   
## 366   
## FORTNITE XB1   
## 365   
## FORNITE SWITCH PREORDER   
## 350   
## (Other)   
## 32206   
##   
## element (itemset/transaction) length distribution:  
## sizes  
## 1 2 3 4 5 6 7 9   
## 30292 1665 174 52 14 3 2 2   
##   
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1.00 1.00 1.00 1.07 1.00 9.00   
##   
## includes extended item information - examples:  
## labels  
## 1 1 2 M FIBRE OPTIC TREE TREE TO  
## 2 1 2 M FIBREOPTIC BURSTTREE 125  
## 3 1 2 M HINGED REDWOOD BASKET TR

inspect(sparse\_data[1:20])

## items   
## [1] {TONGUE AND GROOVED LEDGED AND}   
## [2] {Servis DC60W Ceramic Double Ov}   
## [3] {Russell Hobbs 60CM G Dishwasher - White}   
## [4] {WHITE KNIGHT 5KG 1000RPM WASH}   
## [5] {INDSIT BWE91483KWUK 9KG WM BLK}   
## [6] {Hotpoint FFFL2012P Tall Fridge Freezer - White/Ins/Del/Rec.}  
## [7] {Hotpoint FFFL2012P Tall Fridge Freezer - White/Ins/Del/Rec.}  
## [8] {HOTPOINT HAE51K ELEC COOKER BL}   
## [9] {WHITE 60CM G DISHWASHER}   
## [10] {MIGHTY NO 9 3DS PRE ORDER GAME}   
## [11] {MIGHTY NO 9 3DS PRE ORDER GAME}   
## [12] {MIGHTY NO 9 3DS PRE ORDER GAME}   
## [13] {MIGHTY NO 9 3DS PRE ORDER GAME}   
## [14] {Mighty No.9 Nintendo 3DS Pre-order Game.}   
## [15] {RUSSELL HOBBS BUILT IN 20L SS}   
## [16] {RUSSELL HOBBS BUILT IN 20L SS}   
## [17] {R HOBBS BLK 7KG VENTED TUMBLE}   
## [18] {HOTPOINT HUG61P GAS COOKER BLK}   
## [19] {HOTPOINT CANNON CH60DHKF BLACK,   
## HOTPOINT HUG61P GAS COOKER BLK}   
## [20] {9IE DORMEO MEM PLUS MATT 5FT}

# Plot the most popular products

par(mar = c(4, 5,1, 1))  
itemFrequencyPlot(sparse\_data, topN = 10, type = 'absolute', cex.names = 0.5, horiz = TRUE, xlim = c(0, max(700)))



# Create a rule for the most frequent itemset

rules = apriori(data = sparse\_data, parameter = list(target = "frequent itemsets", support = 0.00015, minlen = 2), control = list(verbose = FALSE))  
rules = sort(rules, by = 'support', decreasing = TRUE)  
inspect(rules[1])

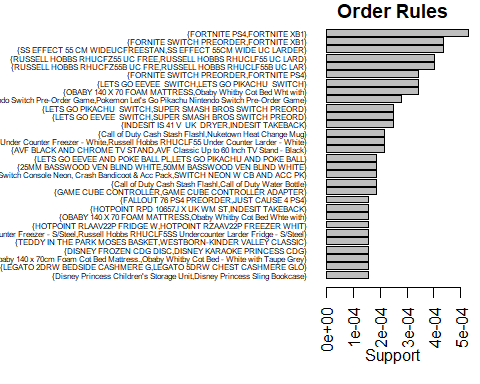
## items support count  
## [1] {FORTNITE PS4,FORTNITE XB1} 0.0005278847 17

# Create a dataframe base on rules data.

order\_rules\_df <- DATAFRAME(sort(rules, by = "support", decreasing = FALSE))

# Create a barplot from the orders dataframe.

par(mar = c(5,17, 1, 1))  
barplot(order\_rules\_df$support, main = "Order Rules", xlab = "Support", names.arg = order\_rules\_df$items, horiz = TRUE, las = 2, cex.names = 0.5, xlim=c(0, max = 0.0005))



# Finding the most common itemsets

rules = apriori(data = sparse\_data, parameter = list(support = 0.0001, confidence = 0.4, minlen = 2))

## Apriori  
##   
## Parameter specification:  
## confidence minval smax arem aval originalSupport maxtime support minlen  
## 0.4 0.1 1 none FALSE TRUE 5 1e-04 2  
## maxlen target ext  
## 10 rules FALSE  
##   
## Algorithmic control:  
## filter tree heap memopt load sort verbose  
## 0.1 TRUE TRUE FALSE TRUE 2 TRUE  
##   
## Absolute minimum support count: 3   
##   
## set item appearances ...[0 item(s)] done [0.00s].  
## set transactions ...[7139 item(s), 32204 transaction(s)] done [0.04s].  
## sorting and recoding items ... [1992 item(s)] done [0.00s].  
## creating transaction tree ... done [0.02s].  
## checking subsets of size 1 2 3 done [0.03s].  
## writing ... [12 rule(s)] done [0.01s].  
## creating S4 object ... done [0.00s].

summary(rules)

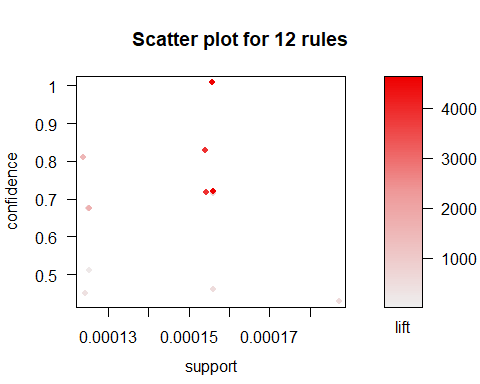
## set of 12 rules  
##   
## rule length distribution (lhs + rhs):sizes  
## 2   
## 12   
##   
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2 2 2 2 2 2   
##   
## summary of quality measures:  
## support confidence lift count   
## Min. :0.0001242 Min. :0.4286 Min. : 164.3 Min. :4.000   
## 1st Qu.:0.0001242 1st Qu.:0.4886 1st Qu.: 530.9 1st Qu.:4.000   
## Median :0.0001553 Median :0.6905 Median :1141.1 Median :5.000   
## Mean :0.0001449 Mean :0.6614 Mean :1909.0 Mean :4.667   
## 3rd Qu.:0.0001553 3rd Qu.:0.7357 3rd Qu.:3833.8 3rd Qu.:5.000   
## Max. :0.0001863 Max. :1.0000 Max. :4600.6 Max. :6.000   
##   
## mining info:  
## data ntransactions support confidence  
## sparse\_data 32204 1e-04 0.4

inspect(rules[1:10])

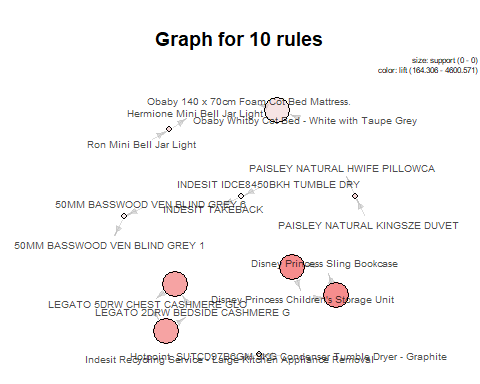
## lhs rhs support confidence lift count  
## [1] {Disney Princess Children's Storage Unit} => {Disney Princess Sling Bookcase} 0.0001552602 1.0000000 4600.5714 5  
## [2] {Disney Princess Sling Bookcase} => {Disney Princess Children's Storage Unit} 0.0001552602 0.7142857 4600.5714 5  
## [3] {Hotpoint SUTCD97B6GM 9KG Condenser Tumble Dryer - Graphite} => {Indesit Recycling Service - Large Kitchen Appliance Removal} 0.0001242082 0.6666667 550.4957 4  
## [4] {LEGATO 2DRW BEDSIDE CASHMERE G} => {LEGATO 5DRW CHEST CASHMERE GLO} 0.0001552602 0.8333333 3833.8095 5  
## [5] {LEGATO 5DRW CHEST CASHMERE GLO} => {LEGATO 2DRW BEDSIDE CASHMERE G} 0.0001552602 0.7142857 3833.8095 5  
## [6] {PAISLEY NATURAL KINGSZE DUVET} => {PAISLEY NATURAL HWIFE PILLOWCA} 0.0001242082 0.6666667 1651.4872 4  
## [7] {Ron Mini Bell Jar Light} => {Hermione Mini Bell Jar Light} 0.0001242082 0.8000000 1515.4824 4  
## [8] {INDESIT IDCE8450BKH TUMBLE DRY} => {INDESIT TAKEBACK} 0.0001242082 0.5000000 164.3061 4  
## [9] {Obaby Whitby Cot Bed - White with Taupe Grey} => {Obaby 140 x 70cm Foam Cot Bed Mattress.} 0.0001552602 0.7142857 766.7619 5  
## [10] {50MM BASSWOOD VEN BLIND GREY 6} => {50MM BASSWOOD VEN BLIND GREY 1} 0.0001242082 0.4444444 366.9972 4

# Plotting most common sets of items being bought together.

plot(rules, method = 'scatterplot', measure = c("support", "confidence"), shading = "lift", jitter = 1)



plot(rules[1:10], method = 'graph', control = list(cex = 0.65))



# Viewing rule sorted by sorted by confidence.

inspect(sort(rules, by = 'confidence', decreasing = TRUE)[1:10])

## lhs rhs support confidence lift count  
## [1] {Disney Princess Children's Storage Unit} => {Disney Princess Sling Bookcase} 0.0001552602 1.0000000 4600.5714 5  
## [2] {LEGATO 2DRW BEDSIDE CASHMERE G} => {LEGATO 5DRW CHEST CASHMERE GLO} 0.0001552602 0.8333333 3833.8095 5  
## [3] {Ron Mini Bell Jar Light} => {Hermione Mini Bell Jar Light} 0.0001242082 0.8000000 1515.4824 4  
## [4] {Disney Princess Sling Bookcase} => {Disney Princess Children's Storage Unit} 0.0001552602 0.7142857 4600.5714 5  
## [5] {LEGATO 5DRW CHEST CASHMERE GLO} => {LEGATO 2DRW BEDSIDE CASHMERE G} 0.0001552602 0.7142857 3833.8095 5  
## [6] {Obaby Whitby Cot Bed - White with Taupe Grey} => {Obaby 140 x 70cm Foam Cot Bed Mattress.} 0.0001552602 0.7142857 766.7619 5  
## [7] {Hotpoint SUTCD97B6GM 9KG Condenser Tumble Dryer - Graphite} => {Indesit Recycling Service - Large Kitchen Appliance Removal} 0.0001242082 0.6666667 550.4957 4  
## [8] {PAISLEY NATURAL KINGSZE DUVET} => {PAISLEY NATURAL HWIFE PILLOWCA} 0.0001242082 0.6666667 1651.4872 4  
## [9] {INDESIT IDCE8450BKH TUMBLE DRY} => {INDESIT TAKEBACK} 0.0001242082 0.5000000 164.3061 4  
## [10] {WESTBORN-KINDER VALLEY CLASSIC} => {TEDDY IN THE PARK MOSES BASKET} 0.0001552602 0.4545455 472.1994 5

# Viewing rule sorted by sorted by support.

inspect(sort(rules, by = 'support', decreasing = TRUE)[1:10])

## lhs rhs support confidence lift count  
## [1] {GAME CUBE CONTROLLER ADAPTER} => {GAME CUBE CONTROLLER} 0.0001863123 0.4285714 552.0686 6  
## [2] {Disney Princess Children's Storage Unit} => {Disney Princess Sling Bookcase} 0.0001552602 1.0000000 4600.5714 5  
## [3] {Disney Princess Sling Bookcase} => {Disney Princess Children's Storage Unit} 0.0001552602 0.7142857 4600.5714 5  
## [4] {LEGATO 2DRW BEDSIDE CASHMERE G} => {LEGATO 5DRW CHEST CASHMERE GLO} 0.0001552602 0.8333333 3833.8095 5  
## [5] {LEGATO 5DRW CHEST CASHMERE GLO} => {LEGATO 2DRW BEDSIDE CASHMERE G} 0.0001552602 0.7142857 3833.8095 5  
## [6] {Obaby Whitby Cot Bed - White with Taupe Grey} => {Obaby 140 x 70cm Foam Cot Bed Mattress.} 0.0001552602 0.7142857 766.7619 5  
## [7] {WESTBORN-KINDER VALLEY CLASSIC} => {TEDDY IN THE PARK MOSES BASKET} 0.0001552602 0.4545455 472.1994 5  
## [8] {Hotpoint SUTCD97B6GM 9KG Condenser Tumble Dryer - Graphite} => {Indesit Recycling Service - Large Kitchen Appliance Removal} 0.0001242082 0.6666667 550.4957 4  
## [9] {PAISLEY NATURAL KINGSZE DUVET} => {PAISLEY NATURAL HWIFE PILLOWCA} 0.0001242082 0.6666667 1651.4872 4  
## [10] {Ron Mini Bell Jar Light} => {Hermione Mini Bell Jar Light} 0.0001242082 0.8000000 1515.4824 4

# Viewing the rule sorted by lift.

inspect(sort(rules, by = 'lift', decreasing = TRUE)[1:10])

## lhs rhs support confidence lift count  
## [1] {Disney Princess Sling Bookcase} => {Disney Princess Children's Storage Unit} 0.0001552602 0.7142857 4600.5714 5  
## [2] {Disney Princess Children's Storage Unit} => {Disney Princess Sling Bookcase} 0.0001552602 1.0000000 4600.5714 5  
## [3] {LEGATO 5DRW CHEST CASHMERE GLO} => {LEGATO 2DRW BEDSIDE CASHMERE G} 0.0001552602 0.7142857 3833.8095 5  
## [4] {LEGATO 2DRW BEDSIDE CASHMERE G} => {LEGATO 5DRW CHEST CASHMERE GLO} 0.0001552602 0.8333333 3833.8095 5  
## [5] {PAISLEY NATURAL KINGSZE DUVET} => {PAISLEY NATURAL HWIFE PILLOWCA} 0.0001242082 0.6666667 1651.4872 4  
## [6] {Ron Mini Bell Jar Light} => {Hermione Mini Bell Jar Light} 0.0001242082 0.8000000 1515.4824 4  
## [7] {Obaby Whitby Cot Bed - White with Taupe Grey} => {Obaby 140 x 70cm Foam Cot Bed Mattress.} 0.0001552602 0.7142857 766.7619 5  
## [8] {GAME CUBE CONTROLLER ADAPTER} => {GAME CUBE CONTROLLER} 0.0001863123 0.4285714 552.0686 6  
## [9] {Hotpoint SUTCD97B6GM 9KG Condenser Tumble Dryer - Graphite} => {Indesit Recycling Service - Large Kitchen Appliance Removal} 0.0001242082 0.6666667 550.4957 4  
## [10] {WESTBORN-KINDER VALLEY CLASSIC} => {TEDDY IN THE PARK MOSES BASKET} 0.0001552602 0.4545455 472.1994 5

# Finding the most common set of items bought with higher support but lower confidence.

rules = apriori(data = sparse\_data, parameter = list(support = 0.0003 , confidence = 0.1 , minlen = 2, maxlen = 3))

## Apriori  
##   
## Parameter specification:  
## confidence minval smax arem aval originalSupport maxtime support minlen  
## 0.1 0.1 1 none FALSE TRUE 5 3e-04 2  
## maxlen target ext  
## 3 rules FALSE  
##   
## Algorithmic control:  
## filter tree heap memopt load sort verbose  
## 0.1 TRUE TRUE FALSE TRUE 2 TRUE  
##   
## Absolute minimum support count: 9   
##   
## set item appearances ...[0 item(s)] done [0.00s].  
## set transactions ...[7139 item(s), 32204 transaction(s)] done [0.04s].  
## sorting and recoding items ... [599 item(s)] done [0.00s].  
## creating transaction tree ... done [0.01s].  
## checking subsets of size 1 2 3

## Warning in apriori(data = sparse\_data, parameter = list(support = 3e-04, :  
## Mining stopped (maxlen reached). Only patterns up to a length of 3  
## returned!

## done [0.00s].  
## writing ... [7 rule(s)] done [0.00s].  
## creating S4 object ... done [0.00s].

summary(rules)

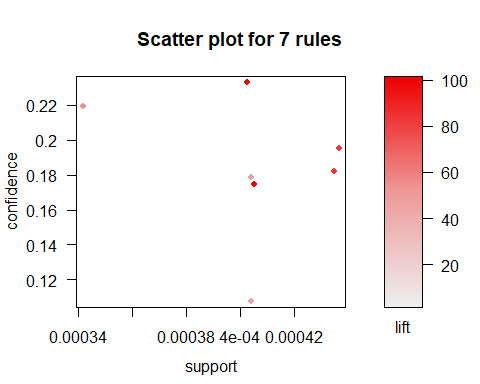
## set of 7 rules  
##   
## rule length distribution (lhs + rhs):sizes  
## 2   
## 7   
##   
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2 2 2 2 2 2   
##   
## summary of quality measures:  
## support confidence lift count   
## Min. :0.0003416 Min. :0.1066 Min. : 47.66 Min. :11.0   
## 1st Qu.:0.0004037 1st Qu.:0.1781 1st Qu.: 51.29 1st Qu.:13.0   
## Median :0.0004037 Median :0.1818 Median : 82.47 Median :13.0   
## Mean :0.0004037 Mean :0.1848 Mean : 73.89 Mean :13.0   
## 3rd Qu.:0.0004192 3rd Qu.:0.2086 3rd Qu.: 91.75 3rd Qu.:13.5   
## Max. :0.0004347 Max. :0.2321 Max. :101.03 Max. :14.0   
##   
## mining info:  
## data ntransactions support confidence  
## sparse\_data 32204 3e-04 0.1

inspect(rules)

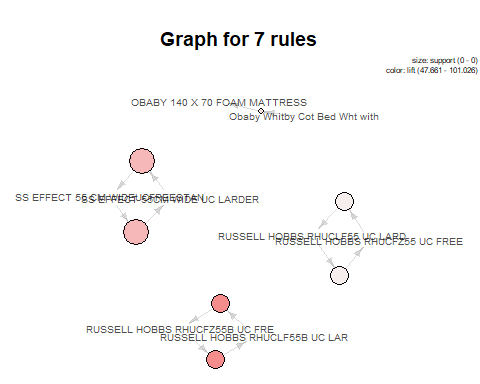
## lhs rhs support confidence lift count  
## [1] {Obaby Whitby Cot Bed Wht with} => {OBABY 140 X 70 FOAM MATTRESS} 0.0003415725 0.2200000 54.92155 11  
## [2] {RUSSELL HOBBS RHUCFZ55B UC FRE} => {RUSSELL HOBBS RHUCLF55B UC LAR} 0.0004036766 0.2321429 101.02606 13  
## [3] {RUSSELL HOBBS RHUCLF55B UC LAR} => {RUSSELL HOBBS RHUCFZ55B UC FRE} 0.0004036766 0.1756757 101.02606 13  
## [4] {RUSSELL HOBBS RHUCFZ55 UC FREE} => {RUSSELL HOBBS RHUCLF55 UC LARD} 0.0004036766 0.1805556 47.66075 13  
## [5] {RUSSELL HOBBS RHUCLF55 UC LARD} => {RUSSELL HOBBS RHUCFZ55 UC FREE} 0.0004036766 0.1065574 47.66075 13  
## [6] {SS EFFECT 55CM WIDE UC LARDER} => {SS EFFECT 55 CM WIDEUCFREESTAN} 0.0004347286 0.1971831 82.46863 14  
## [7] {SS EFFECT 55 CM WIDEUCFREESTAN} => {SS EFFECT 55CM WIDE UC LARDER} 0.0004347286 0.1818182 82.46863 14

# Plotting most common sets of items being bought together with low confidence but high support.

plot(rules, method = 'scatterplot', measure = c("support", "confidence"), shading = "lift", jitter = 1)



plot(rules, method = "graph", control = list (cex = 0.65))



# Creating Fortnite PS4 subset

rules = apriori(data = sparse\_data, parameter = list(support = 0.00001, confidence = 0.01, minlen = 2))

## Apriori  
##   
## Parameter specification:  
## confidence minval smax arem aval originalSupport maxtime support minlen  
## 0.01 0.1 1 none FALSE TRUE 5 1e-05 2  
## maxlen target ext  
## 10 rules FALSE  
##   
## Algorithmic control:  
## filter tree heap memopt load sort verbose  
## 0.1 TRUE TRUE FALSE TRUE 2 TRUE  
##   
## Absolute minimum support count: 0   
##   
## set item appearances ...[0 item(s)] done [0.00s].  
## set transactions ...[7139 item(s), 32204 transaction(s)] done [0.02s].  
## sorting and recoding items ... [7139 item(s)] done [0.00s].  
## creating transaction tree ... done [0.01s].  
## checking subsets of size 1 2 3 4 5 6 7 8 9 done [0.32s].  
## writing ... [11904 rule(s)] done [0.10s].  
## creating S4 object ... done [0.02s].

fnps4\_rules = sort(  
 subset(rules, subset = rhs %in% 'FORTNITE PS4'),  
 by = 'lift', decreasing = T  
)  
inspect(fnps4\_rules)

## lhs rhs support confidence lift count  
## [1] {SKULL AND BONES PS4} => {FORTNITE PS4} 3.105204e-05 1.00000000 77.9757869 1  
## [2] {FORNITE SWITCH PREORDER,   
## SKULL AND BONES PS4} => {FORTNITE PS4} 3.105204e-05 1.00000000 77.9757869 1  
## [3] {FALLOUT 76 PS4 PREORDER,   
## FORNITE SWITCH PREORDER} => {FORTNITE PS4} 3.105204e-05 1.00000000 77.9757869 1  
## [4] {FORTNITE XB1,   
## PS4 DUALSHOCK LTD ED 500 MILLI} => {FORTNITE PS4} 3.105204e-05 1.00000000 77.9757869 1  
## [5] {ACE COMBAT 7 SKIES UNKNOWN PS4} => {FORTNITE PS4} 3.105204e-05 0.33333333 25.9919290 1  
## [6] {GRIP COMBAT RACING PS4 PREORDE} => {FORTNITE PS4} 3.105204e-05 0.33333333 25.9919290 1  
## [7] {FORNITE SWITCH PREORDER,   
## FORTNITE XB1} => {FORTNITE PS4} 3.105204e-05 0.07142857 5.5696991 1  
## [8] {JUST CAUSE 4 PS4} => {FORTNITE PS4} 3.105204e-05 0.04761905 3.7131327 1  
## [9] {FORTNITE XB1} => {FORTNITE PS4} 5.278847e-04 0.04657534 3.6317490 17  
## [10] {HITMAN 2 PS4 PREORDER} => {FORTNITE PS4} 3.105204e-05 0.04545455 3.5443540 1  
## [11] {OFFICIAL PREMIUM STORAGE TOWER} => {FORTNITE PS4} 3.105204e-05 0.03333333 2.5991929 1  
## [12] {FORNITE SWITCH PREORDER} => {FORTNITE PS4} 3.415725e-04 0.03142857 2.4506676 11  
## [13] {FALLOUT 76 PS4 PREORDER} => {FORTNITE PS4} 6.210409e-05 0.01869159 1.4574913 2  
## [14] {SUPER SMASH BROS ULT LIM ED S} => {FORTNITE PS4} 3.105204e-05 0.01515152 1.1814513 1  
## [15] {LETS GO PIKACHU SWITCH} => {FORTNITE PS4} 6.210409e-05 0.01081081 0.8429815 2

lift\_fnps4 = sort(fnps4\_rules, by = 'lift', decreasing = TRUE)  
paste("Sorted by lift. ----------------------------------")

## [1] "Sorted by lift. ----------------------------------"

inspect(lift\_fnps4)

## lhs rhs support confidence lift count  
## [1] {SKULL AND BONES PS4} => {FORTNITE PS4} 3.105204e-05 1.00000000 77.9757869 1  
## [2] {FORNITE SWITCH PREORDER,   
## SKULL AND BONES PS4} => {FORTNITE PS4} 3.105204e-05 1.00000000 77.9757869 1  
## [3] {FALLOUT 76 PS4 PREORDER,   
## FORNITE SWITCH PREORDER} => {FORTNITE PS4} 3.105204e-05 1.00000000 77.9757869 1  
## [4] {FORTNITE XB1,   
## PS4 DUALSHOCK LTD ED 500 MILLI} => {FORTNITE PS4} 3.105204e-05 1.00000000 77.9757869 1  
## [5] {ACE COMBAT 7 SKIES UNKNOWN PS4} => {FORTNITE PS4} 3.105204e-05 0.33333333 25.9919290 1  
## [6] {GRIP COMBAT RACING PS4 PREORDE} => {FORTNITE PS4} 3.105204e-05 0.33333333 25.9919290 1  
## [7] {FORNITE SWITCH PREORDER,   
## FORTNITE XB1} => {FORTNITE PS4} 3.105204e-05 0.07142857 5.5696991 1  
## [8] {JUST CAUSE 4 PS4} => {FORTNITE PS4} 3.105204e-05 0.04761905 3.7131327 1  
## [9] {FORTNITE XB1} => {FORTNITE PS4} 5.278847e-04 0.04657534 3.6317490 17  
## [10] {HITMAN 2 PS4 PREORDER} => {FORTNITE PS4} 3.105204e-05 0.04545455 3.5443540 1  
## [11] {OFFICIAL PREMIUM STORAGE TOWER} => {FORTNITE PS4} 3.105204e-05 0.03333333 2.5991929 1  
## [12] {FORNITE SWITCH PREORDER} => {FORTNITE PS4} 3.415725e-04 0.03142857 2.4506676 11  
## [13] {FALLOUT 76 PS4 PREORDER} => {FORTNITE PS4} 6.210409e-05 0.01869159 1.4574913 2  
## [14] {SUPER SMASH BROS ULT LIM ED S} => {FORTNITE PS4} 3.105204e-05 0.01515152 1.1814513 1  
## [15] {LETS GO PIKACHU SWITCH} => {FORTNITE PS4} 6.210409e-05 0.01081081 0.8429815 2

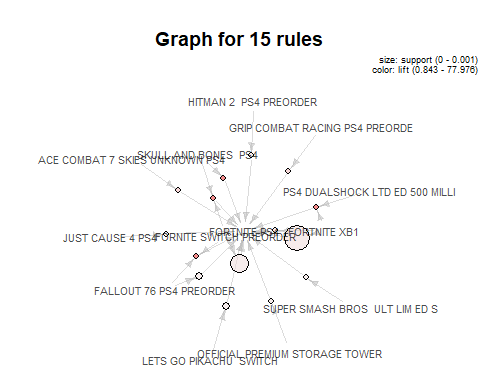
confidence\_fnps4 = sort(fnps4\_rules, by = 'confidence', decreasing = TRUE)  
paste("Sorted by confidence. ----------------------------------")

## [1] "Sorted by confidence. ----------------------------------"

inspect(confidence\_fnps4)

## lhs rhs support confidence lift count  
## [1] {SKULL AND BONES PS4} => {FORTNITE PS4} 3.105204e-05 1.00000000 77.9757869 1  
## [2] {FORNITE SWITCH PREORDER,   
## SKULL AND BONES PS4} => {FORTNITE PS4} 3.105204e-05 1.00000000 77.9757869 1  
## [3] {FALLOUT 76 PS4 PREORDER,   
## FORNITE SWITCH PREORDER} => {FORTNITE PS4} 3.105204e-05 1.00000000 77.9757869 1  
## [4] {FORTNITE XB1,   
## PS4 DUALSHOCK LTD ED 500 MILLI} => {FORTNITE PS4} 3.105204e-05 1.00000000 77.9757869 1  
## [5] {ACE COMBAT 7 SKIES UNKNOWN PS4} => {FORTNITE PS4} 3.105204e-05 0.33333333 25.9919290 1  
## [6] {GRIP COMBAT RACING PS4 PREORDE} => {FORTNITE PS4} 3.105204e-05 0.33333333 25.9919290 1  
## [7] {FORNITE SWITCH PREORDER,   
## FORTNITE XB1} => {FORTNITE PS4} 3.105204e-05 0.07142857 5.5696991 1  
## [8] {JUST CAUSE 4 PS4} => {FORTNITE PS4} 3.105204e-05 0.04761905 3.7131327 1  
## [9] {FORTNITE XB1} => {FORTNITE PS4} 5.278847e-04 0.04657534 3.6317490 17  
## [10] {HITMAN 2 PS4 PREORDER} => {FORTNITE PS4} 3.105204e-05 0.04545455 3.5443540 1  
## [11] {OFFICIAL PREMIUM STORAGE TOWER} => {FORTNITE PS4} 3.105204e-05 0.03333333 2.5991929 1  
## [12] {FORNITE SWITCH PREORDER} => {FORTNITE PS4} 3.415725e-04 0.03142857 2.4506676 11  
## [13] {FALLOUT 76 PS4 PREORDER} => {FORTNITE PS4} 6.210409e-05 0.01869159 1.4574913 2  
## [14] {SUPER SMASH BROS ULT LIM ED S} => {FORTNITE PS4} 3.105204e-05 0.01515152 1.1814513 1  
## [15] {LETS GO PIKACHU SWITCH} => {FORTNITE PS4} 6.210409e-05 0.01081081 0.8429815 2

plot(fnps4\_rules, method = 'graph', control = list(cex = 0.7))



# Generate summary reports for summary and lift rules.

paste("Rules Report")

## [1] "Rules Report"

paste("Support")

## [1] "Support"

summary(rules@quality$support)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 3.105e-05 3.105e-05 3.105e-05 3.452e-05 3.105e-05 5.279e-04

paste("Lift")

## [1] "Lift"

summary(rules@quality$lift)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.46 2300.29 10734.67 10763.97 16102.00 32204.00

paste("Fortnite PS4 Rules Report")

## [1] "Fortnite PS4 Rules Report"

paste("Support")

## [1] "Support"

summary(rules@quality$support)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 3.105e-05 3.105e-05 3.105e-05 3.452e-05 3.105e-05 5.279e-04

paste("Lift")

## [1] "Lift"

summary(rules@quality$lift)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.46 2300.29 10734.67 10763.97 16102.00 32204.00

# Comparing Fornite PS4 rule results to the other rules.

support\_list <- rules@quality$support  
mean\_fnps4\_support <- mean(fnps4\_rules@quality$support)  
lower\_than\_mean\_rules <- support\_list[support\_list>c(mean\_fnps4\_support)]  
support\_rank <- length(lower\_than\_mean\_rules)  
support\_rank

## [1] 220

paste("On average, Fortnite PS4 has the", support\_rank, "largest support of all the", length(rules), "rules")

## [1] "On average, Fortnite PS4 has the 220 largest support of all the 11904 rules"

confidence\_list <- rules@quality$confidence  
mean\_fnps4\_confidence <- mean(fnps4\_rules@quality$confidence)  
lower\_than\_mean\_rules\_confidence <- confidence\_list[confidence\_list>c(mean\_fnps4\_confidence)]  
confidence\_rank <- length(lower\_than\_mean\_rules\_confidence)  
confidence\_rank

## [1] 9329

paste("On average, Fortnite PS4 has the", confidence\_rank, "largest confidence of all the", length(rules), "rules")

## [1] "On average, Fortnite PS4 has the 9329 largest confidence of all the 11904 rules"

# Summary

Skull and Bones PS4 could sell well as part of a bundle with Fortnite PS4 as could Fallout 76 PS4 Preorder these are shown by the high lift, although this is with low support (Hard to tell if this is due to the sample size or if this just has low support).