FinalProject

Shivangi Vashi

6/22/2020

Association Rules Mining

For the Market Basket Analysis, only the order_products__prior and product are utilized and the same are joined on basis of product id.

```
library(plyr)
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.0 ──
```

```
## ✓ ggplot2 3.3.2     ✓ purrr   0.3.4
## ✓ tibble  3.0.1     ✓ dplyr   1.0.0
## ✓ tidyr   1.1.0     ✓ stringr 1.4.0
## ✓ readr   1.3.1     ✓ forcats 0.5.0
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## x dplyr::arrange()   masks plyr::arrange()
## x purrr::compact()   masks plyr::compact()
## x dplyr::count()     masks plyr::count()
## x dplyr::failwith()  masks plyr::failwith()
## x dplyr::filter()    masks stats::filter()
## x dplyr::id()        masks plyr::id()
## x dplyr::lag()       masks stats::lag()
## x dplyr::mutate()    masks plyr::mutate()
## x dplyr::rename()    masks plyr::rename()
## x dplyr::summarise() masks plyr::summarise()
## x dplyr::summarize() masks plyr::summarize()
```

```
library(data.table)
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
## The following object is masked from 'package:purrr':
##
##     transpose
```

```
library(dplyr)
library(arules)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
##
## Attaching package: 'arules'
```

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

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

```
library(arulesViz)
```

```
## Loading required package: grid
```

```
## Registered S3 method overwritten by 'seriation':
##   method         from
##   reorder.hclust gclus
```

```
order_products_prior<-fread("instacart-market-basket-analysis/order_products__prior.csv")
order_products_train<-fread("instacart-market-basket-analysis/order_products__train.csv")
orders<-fread("instacart-market-basket-analysis/orders.csv")
products<-fread("instacart-market-basket-analysis/products.csv")

set.seed(123)
user_fraction <- 0.1
users <- unique(orders$user_id)
sample_users <- sample(users, round(user_fraction * length(users)))

cat('Number of orders (before): ', nrow(orders))
```

```
## Number of orders (before):  3421083
```

```
orders <- orders[user_id %in% sample_users]
cat('Number of orders (after): ', nrow(orders))
```

```
## Number of orders (after):  346739
```

```
# Training dataset
OrderProductPrior<-orders%>%inner_join(order_products_prior)
```

```
## Joining, by = "order_id"
```

```
OrderProductPrior<-drop_na(OrderProductPrior)


#Testing dataset
OrderProductTrain<-orders%>%inner_join(order_products_train)
```

```
## Joining, by = "order_id"
```

```
OrderProductTrain<-drop_na(OrderProductTrain)


dim(OrderProductPrior)
```

```
## [1] 3061150      10
```

```
dim(OrderProductTrain)
```

```
## [1] 137284     10
```

```
head(OrderProductPrior)
```

```
##    order_id user_id eval_set order_number order_dow order_hour_of_day
## 1:  3106101      18    prior            2         0                17
## 2:  3106101      18    prior            2         0                17
## 3:  3106101      18    prior            2         0                17
## 4:  3106101      18    prior            2         0                17
## 5:  3106101      18    prior            2         0                17
## 6:  1860960      18    prior            3         1                19
##    days_since_prior_order product_id add_to_cart_order reordered
## 1:                      1      36216                 1         1
## 2:                      1       4461                 2         0
## 3:                      1       5876                 3         0
## 4:                      1        810                 4         0
## 5:                      1      31717                 5         0
## 6:                      8      36216                 1         1
```

```
head(OrderProductTrain)
```

```
##    order_id user_id eval_set order_number order_dow order_hour_of_day
## 1:  2461523      18    train            7         6                 9
## 2:  2461523      18    train            7         6                 9
## 3:  2461523      18    train            7         6                 9
## 4:  2461523      18    train            7         6                 9
## 5:  2461523      18    train            7         6                 9
## 6:  2461523      18    train            7         6                 9
##    days_since_prior_order product_id add_to_cart_order reordered
## 1:                      7      36216                 1         1
## 2:                      7      47546                 2         1
## 3:                      7      21137                 3         1
## 4:                      7       5450                 4         0
## 5:                      7       8518                 5         0
## 6:                      7      22031                 6         1
```

Preparing data for apriori

```
basket_data <- left_join(OrderProductPrior, products, by='product_id')
basket_data <-  group_by(basket_data, order_id)
basket_data <- summarise(basket_data,items=as.vector(list(product_name)))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
head(basket_data)
```

```
## # A tibble: 6 x 2
##   order_id items
##      <int> <list>
## 1        3 <chr [8]>
## 2       42 <chr [3]>
## 3       60 <chr [18]>
## 4       63 <chr [10]>
## 5       66 <chr [6]>
## 6       74 <chr [11]>
```

Since the eclat and apriori functions are only valid on transaction format, henceforth the format is converted to the transaction type with the help of below code snippet.

```
transactions=as(basket_data$items, 'transactions')
head(transactions)
```

```
## transactions in sparse format with
##  6 transactions (rows) and
##  40840 items (columns)
```

```
length(transactions)
```

```
## [1] 305497
```

Setting the Support and Confidence Intervals

Understanding support

A value of Support =0.02 means that an item will be considered as frequent if at least 2 percent of all the baskets contain it.

Understanding Confidence

Confidence is a measure of the strength of an association rule. It is the frequency of occurrence of the right-hand items in the rule from among those baskets that contain the items on the left-hand side of the rule.

So now we have created a function that will plot the number of rules we can generate depending on different support levels and varied upon different confidence levels.

```
# Support and confidence values
supportLevels <- c(0.1, 0.05, 0.01, 0.005)
confidenceLevels <- c(0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1)

# Empty integers
rules_sup10 <- integer(length=9)
rules_sup5 <- integer(length=9)
rules_sup1 <- integer(length=9)
rules_sup0.5 <- integer(length=9)

# Apriori algorithm with a support level of 10%
for (i in 1:length(confidenceLevels)) {

  rules_sup10[i] <- length(apriori(transactions, parameter=list(sup=supportLevels[1],
                                   conf=confidenceLevels[i], target="rules")))

}
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.9    0.1    1 none FALSE            TRUE       5     0.1      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 30549
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.31s].
## sorting and recoding items ... [2 item(s)] done [0.01s].
## creating transaction tree ... done [0.02s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.05s].
## Apriori
```

```
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.8    0.1    1 none FALSE            TRUE       5     0.1      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 30549
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.22s].
## sorting and recoding items ... [2 item(s)] done [0.01s].
## creating transaction tree ... done [0.02s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.05s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.7    0.1    1 none FALSE            TRUE       5     0.1      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 30549
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.22s].
## sorting and recoding items ... [2 item(s)] done [0.01s].
## creating transaction tree ... done [0.03s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.05s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.6    0.1    1 none FALSE            TRUE       5     0.1      1
##  maxlen target  ext
```

```
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 30549
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
## sorting and recoding items ... [2 item(s)] done [0.01s].
## creating transaction tree ... done [0.03s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.05s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.5    0.1    1 none FALSE            TRUE       5     0.1      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 30549
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.19s].
## sorting and recoding items ... [2 item(s)] done [0.01s].
## creating transaction tree ... done [0.02s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.05s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.4    0.1    1 none FALSE            TRUE       5     0.1      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
```

```
##
## Absolute minimum support count: 30549
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
## sorting and recoding items ... [2 item(s)] done [0.01s].
## creating transaction tree ... done [0.02s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.05s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.3    0.1    1 none FALSE            TRUE       5     0.1      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 30549
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
## sorting and recoding items ... [2 item(s)] done [0.01s].
## creating transaction tree ... done [0.03s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.05s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.2    0.1    1 none FALSE            TRUE       5     0.1      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 30549
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
```

```
## sorting and recoding items ... [2 item(s)] done [0.01s].
## creating transaction tree ... done [0.02s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.05s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5     0.1      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 30549
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
## sorting and recoding items ... [2 item(s)] done [0.01s].
## creating transaction tree ... done [0.02s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [2 rule(s)] done [0.00s].
## creating S4 object  ... done [0.05s].
```

```
# Apriori algorithm with a support level of 5%
for (i in 1:length(confidenceLevels)){

  rules_sup5[i] <- length(apriori(transactions, parameter=list(sup=supportLevels[2],
                                  conf=confidenceLevels[i], target="rules")))

}
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.9    0.1    1 none FALSE            TRUE       5    0.05      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
```

```
##
## Absolute minimum support count: 15274
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
## sorting and recoding items ... [6 item(s)] done [0.01s].
## creating transaction tree ... done [0.04s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.06s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.8    0.1    1 none FALSE            TRUE       5    0.05      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 15274
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.18s].
## sorting and recoding items ... [6 item(s)] done [0.01s].
## creating transaction tree ... done [0.04s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.06s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.7    0.1    1 none FALSE            TRUE       5    0.05      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 15274
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
```

```
## sorting and recoding items ... [6 item(s)] done [0.01s].
## creating transaction tree ... done [0.04s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.06s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.6    0.1    1 none FALSE            TRUE       5    0.05      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 15274
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.19s].
## sorting and recoding items ... [6 item(s)] done [0.01s].
## creating transaction tree ... done [0.04s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.07s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.5    0.1    1 none FALSE            TRUE       5    0.05      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 15274
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.19s].
## sorting and recoding items ... [6 item(s)] done [0.01s].
## creating transaction tree ... done [0.04s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.06s].
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.4    0.1    1 none FALSE            TRUE       5    0.05      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 15274
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.21s].
## sorting and recoding items ... [6 item(s)] done [0.01s].
## creating transaction tree ... done [0.04s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.06s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.3    0.1    1 none FALSE            TRUE       5    0.05      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 15274
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
## sorting and recoding items ... [6 item(s)] done [0.01s].
## creating transaction tree ... done [0.04s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.06s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.2    0.1    1 none FALSE            TRUE       5    0.05      1
```

```
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 15274
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.19s].
## sorting and recoding items ... [6 item(s)] done [0.01s].
## creating transaction tree ... done [0.05s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.06s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5    0.05      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 15274
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
## sorting and recoding items ... [6 item(s)] done [0.01s].
## creating transaction tree ... done [0.04s].
## checking subsets of size 1 2 done [0.00s].
## writing ... [2 rule(s)] done [0.00s].
## creating S4 object  ... done [0.06s].
```

```
# Apriori algorithm with a support level of 1%
for (i in 1:length(confidenceLevels)){

  rules_sup1[i] <- length(apriori(transactions, parameter=list(sup=supportLevels[3],
                                  conf=confidenceLevels[i], target="rules")))

}
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.9    0.1    1 none FALSE            TRUE       5    0.01      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 3054
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.18s].
## sorting and recoding items ... [103 item(s)] done [0.02s].
## creating transaction tree ... done [0.14s].
## checking subsets of size 1 2 3 done [0.02s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.09s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.8    0.1    1 none FALSE            TRUE       5    0.01      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 3054
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
## sorting and recoding items ... [103 item(s)] done [0.02s].
## creating transaction tree ... done [0.15s].
## checking subsets of size 1 2 3 done [0.02s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.09s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.7    0.1    1 none FALSE            TRUE       5    0.01      1
```

```
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 3054
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
## sorting and recoding items ... [103 item(s)] done [0.02s].
## creating transaction tree ... done [0.15s].
## checking subsets of size 1 2 3 done [0.02s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.09s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.6    0.1    1 none FALSE            TRUE       5    0.01      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 3054
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.19s].
## sorting and recoding items ... [103 item(s)] done [0.02s].
## creating transaction tree ... done [0.15s].
## checking subsets of size 1 2 3 done [0.02s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.09s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.5    0.1    1 none FALSE            TRUE       5    0.01      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
```

```
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 3054
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
## sorting and recoding items ... [103 item(s)] done [0.02s].
## creating transaction tree ... done [0.15s].
## checking subsets of size 1 2 3 done [0.02s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.08s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.4    0.1    1 none FALSE            TRUE       5    0.01      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 3054
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
## sorting and recoding items ... [103 item(s)] done [0.02s].
## creating transaction tree ... done [0.14s].
## checking subsets of size 1 2 3 done [0.02s].
## writing ... [1 rule(s)] done [0.00s].
## creating S4 object  ... done [0.08s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.3    0.1    1 none FALSE            TRUE       5    0.01      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 3054
##
## set item appearances ...[0 item(s)] done [0.00s].
```

```
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
## sorting and recoding items ... [103 item(s)] done [0.02s].
## creating transaction tree ... done [0.15s].
## checking subsets of size 1 2 3 done [0.02s].
## writing ... [2 rule(s)] done [0.00s].
## creating S4 object  ... done [0.09s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.2    0.1    1 none FALSE            TRUE       5    0.01      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 3054
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.21s].
## sorting and recoding items ... [103 item(s)] done [0.02s].
## creating transaction tree ... done [0.15s].
## checking subsets of size 1 2 3 done [0.02s].
## writing ... [12 rule(s)] done [0.00s].
## creating S4 object  ... done [0.09s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5    0.01      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 3054
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.22s].
## sorting and recoding items ... [103 item(s)] done [0.02s].
## creating transaction tree ... done [0.15s].
## checking subsets of size 1 2 3 done [0.02s].
## writing ... [28 rule(s)] done [0.00s].
```

```
## creating S4 object  ... done [0.09s].
```

```
# Apriori algorithm with a support level of 0.5%
for (i in 1:length(confidenceLevels)){

  rules_sup0.5[i] <- length(apriori(transactions, parameter=list(sup=supportLevels[4]
,
                                    conf=confidenceLevels[i], target="rules")))

}
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.9    0.1    1 none FALSE            TRUE       5   0.005      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 1527
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
## sorting and recoding items ... [249 item(s)] done [0.03s].
## creating transaction tree ... done [0.16s].
## checking subsets of size 1 2 3 done [0.03s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.09s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.8    0.1    1 none FALSE            TRUE       5   0.005      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 1527
##
```

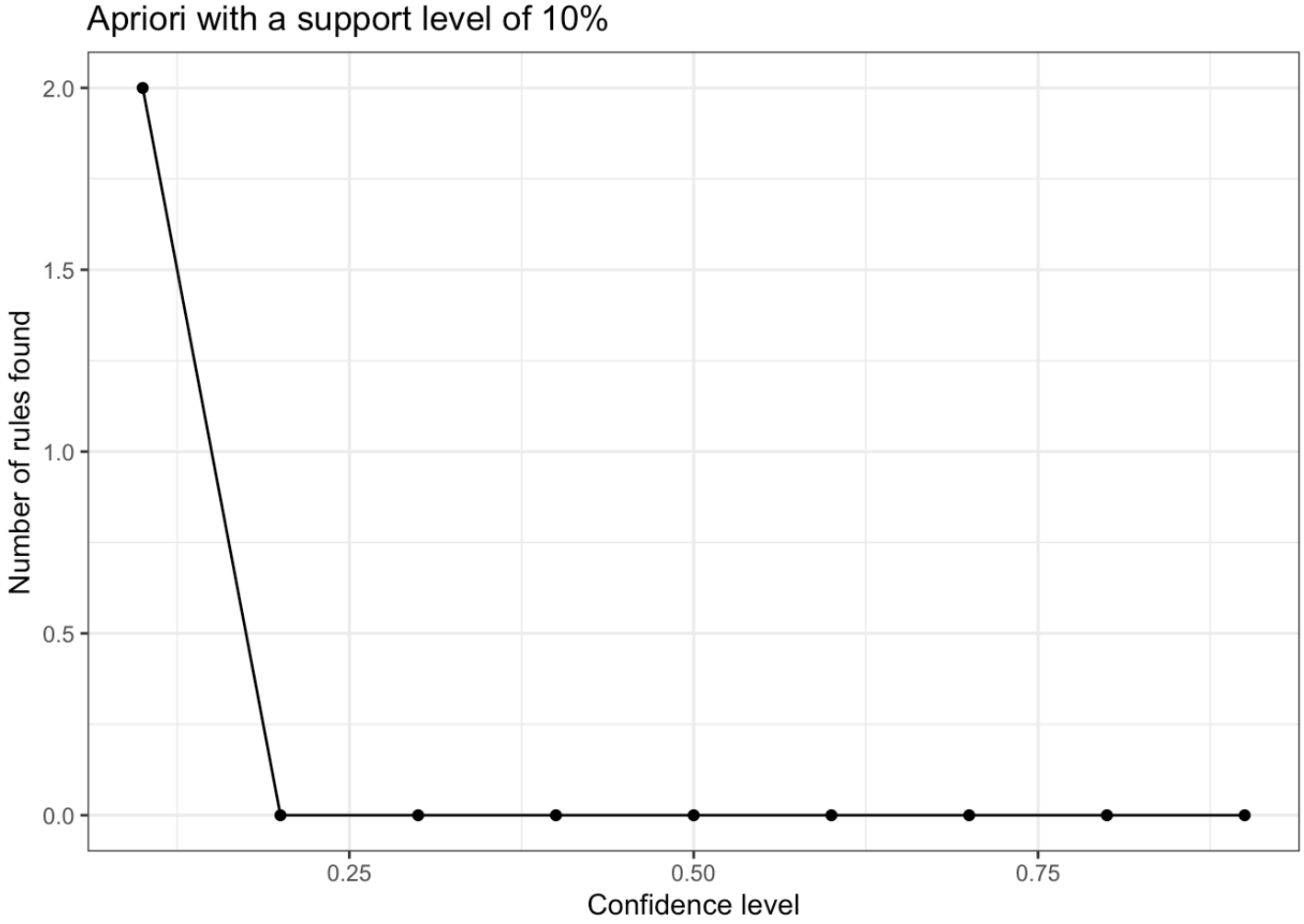
```
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
## sorting and recoding items ... [249 item(s)] done [0.03s].
## creating transaction tree ... done [0.17s].
## checking subsets of size 1 2 3 done [0.03s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.09s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.7    0.1    1 none FALSE            TRUE       5   0.005      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 1527
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.18s].
## sorting and recoding items ... [249 item(s)] done [0.03s].
## creating transaction tree ... done [0.16s].
## checking subsets of size 1 2 3 done [0.03s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.09s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.6    0.1    1 none FALSE            TRUE       5   0.005      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 1527
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.20s].
## sorting and recoding items ... [249 item(s)] done [0.03s].
## creating transaction tree ... done [0.16s].
## checking subsets of size 1 2 3 done [0.03s].
```

```
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.09s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.5    0.1    1 none FALSE            TRUE       5   0.005      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 1527
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.22s].
## sorting and recoding items ... [249 item(s)] done [0.03s].
## creating transaction tree ... done [0.16s].
## checking subsets of size 1 2 3 done [0.03s].
## writing ... [0 rule(s)] done [0.00s].
## creating S4 object  ... done [0.09s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.4    0.1    1 none FALSE            TRUE       5   0.005      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 1527
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.21s].
## sorting and recoding items ... [249 item(s)] done [0.03s].
## creating transaction tree ... done [0.17s].
## checking subsets of size 1 2 3 done [0.03s].
## writing ... [1 rule(s)] done [0.00s].
## creating S4 object  ... done [0.09s].
## Apriori
##
## Parameter specification:
```

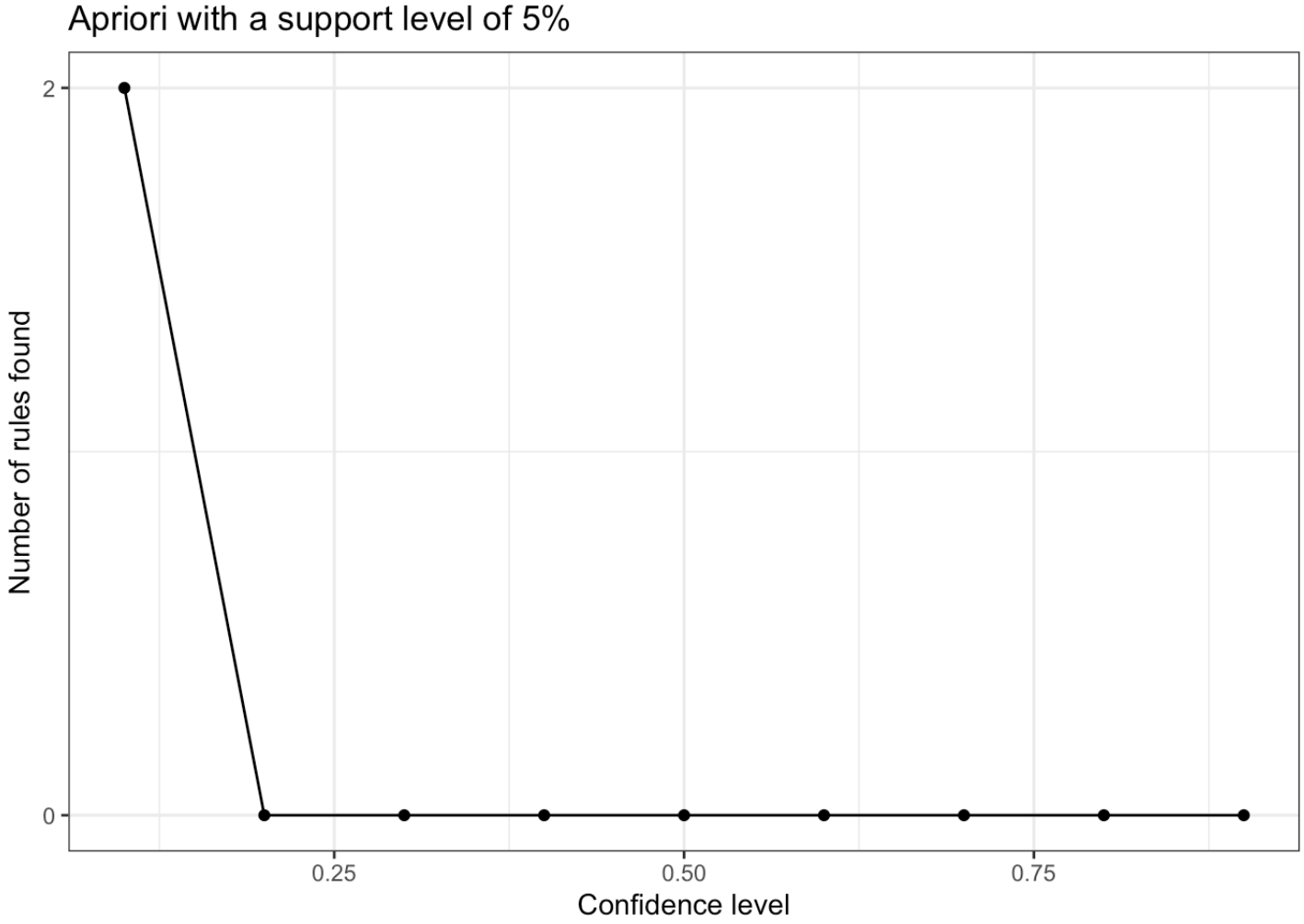
```
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.3    0.1    1 none FALSE            TRUE       5   0.005      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 1527
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.21s].
## sorting and recoding items ... [249 item(s)] done [0.03s].
## creating transaction tree ... done [0.17s].
## checking subsets of size 1 2 3 done [0.03s].
## writing ... [5 rule(s)] done [0.00s].
## creating S4 object  ... done [0.10s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.2    0.1    1 none FALSE            TRUE       5   0.005      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 1527
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.24s].
## sorting and recoding items ... [249 item(s)] done [0.02s].
## creating transaction tree ... done [0.17s].
## checking subsets of size 1 2 3 done [0.03s].
## writing ... [42 rule(s)] done [0.00s].
## creating S4 object  ... done [0.10s].
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5   0.005      1
##  maxlen target  ext
##      10  rules TRUE
##
```

```
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 1527
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[40840 item(s), 305497 transaction(s)] done [1.25s].
## sorting and recoding items ... [249 item(s)] done [0.02s].
## creating transaction tree ... done [0.17s].
## checking subsets of size 1 2 3 done [0.03s].
## writing ... [103 rule(s)] done [0.00s].
## creating S4 object  ... done [0.09s].
```

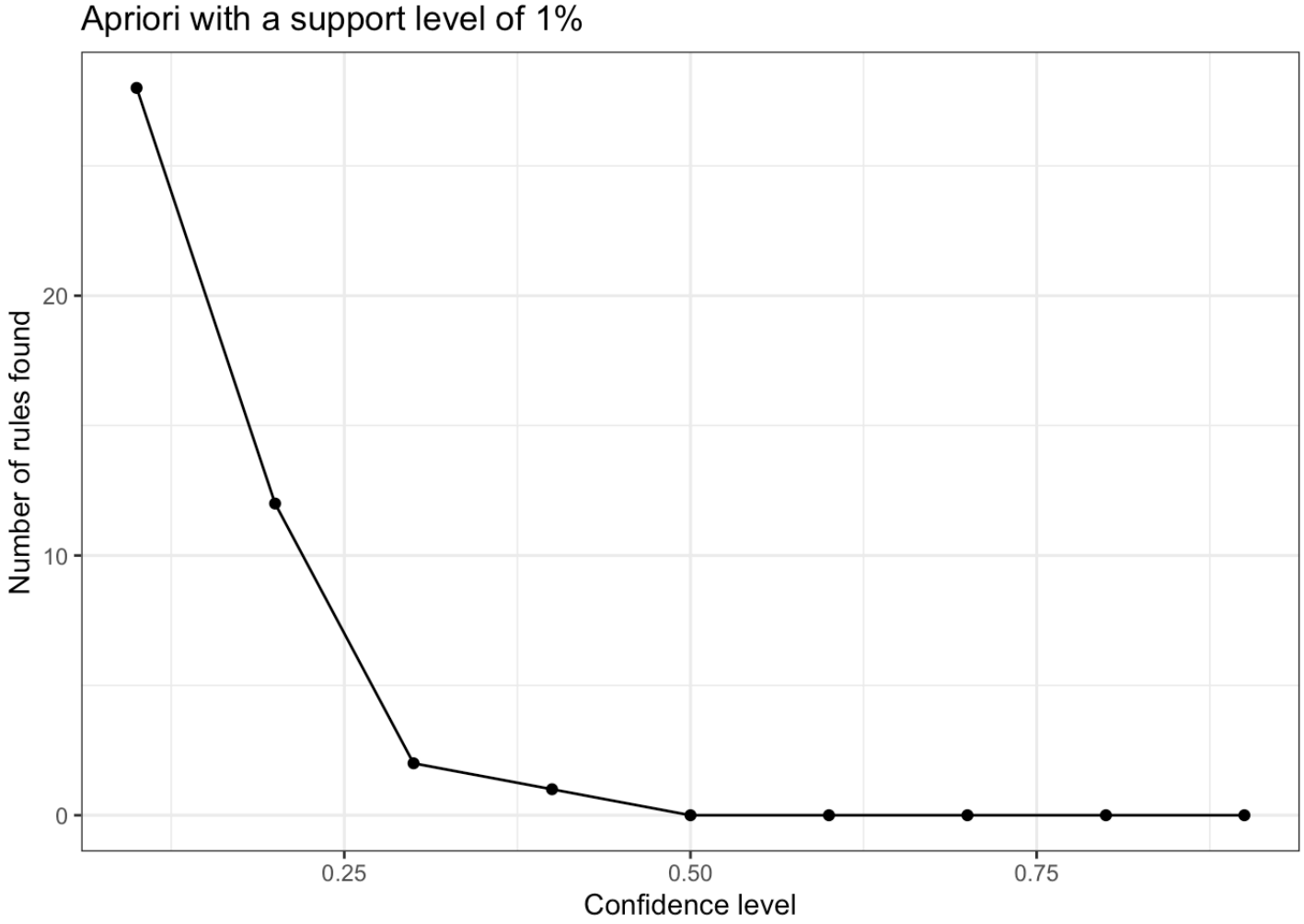
```
# Number of rules found with a support level of 10%
plot1 <- qplot(confidenceLevels, rules_sup10, geom=c("point", "line"),
               xlab="Confidence level", ylab="Number of rules found",
               main="Apriori with a support level of 10%") +
  theme_bw()

# Number of rules found with a support level of 5%
plot2 <- qplot(confidenceLevels, rules_sup5, geom=c("point", "line"),
               xlab="Confidence level", ylab="Number of rules found",
               main="Apriori with a support level of 5%") +
  scale_y_continuous(breaks=seq(0, 10, 2)) +
  theme_bw()

# Number of rules found with a support level of 1%
plot3 <- qplot(confidenceLevels, rules_sup1, geom=c("point", "line"),
               xlab="Confidence level", ylab="Number of rules found",
               main="Apriori with a support level of 1%") +
  scale_y_continuous(breaks=seq(0, 50, 10)) +
  theme_bw()

# Number of rules found with a support level of 0.5%
plot4 <- qplot(confidenceLevels, rules_sup0.5, geom=c("point", "line"),
               xlab="Confidence level", ylab="Number of rules found",
               main="Apriori with a support level of 0.5%") +
  scale_y_continuous(breaks=seq(0, 130, 20)) +
  theme_bw()

# Subplot
plot1
```


Apriori with a support level of 10%
Number of rules found
2.0
1.5
1.0
0.5
0.0
0.25
0.50
0.75
Confidence level

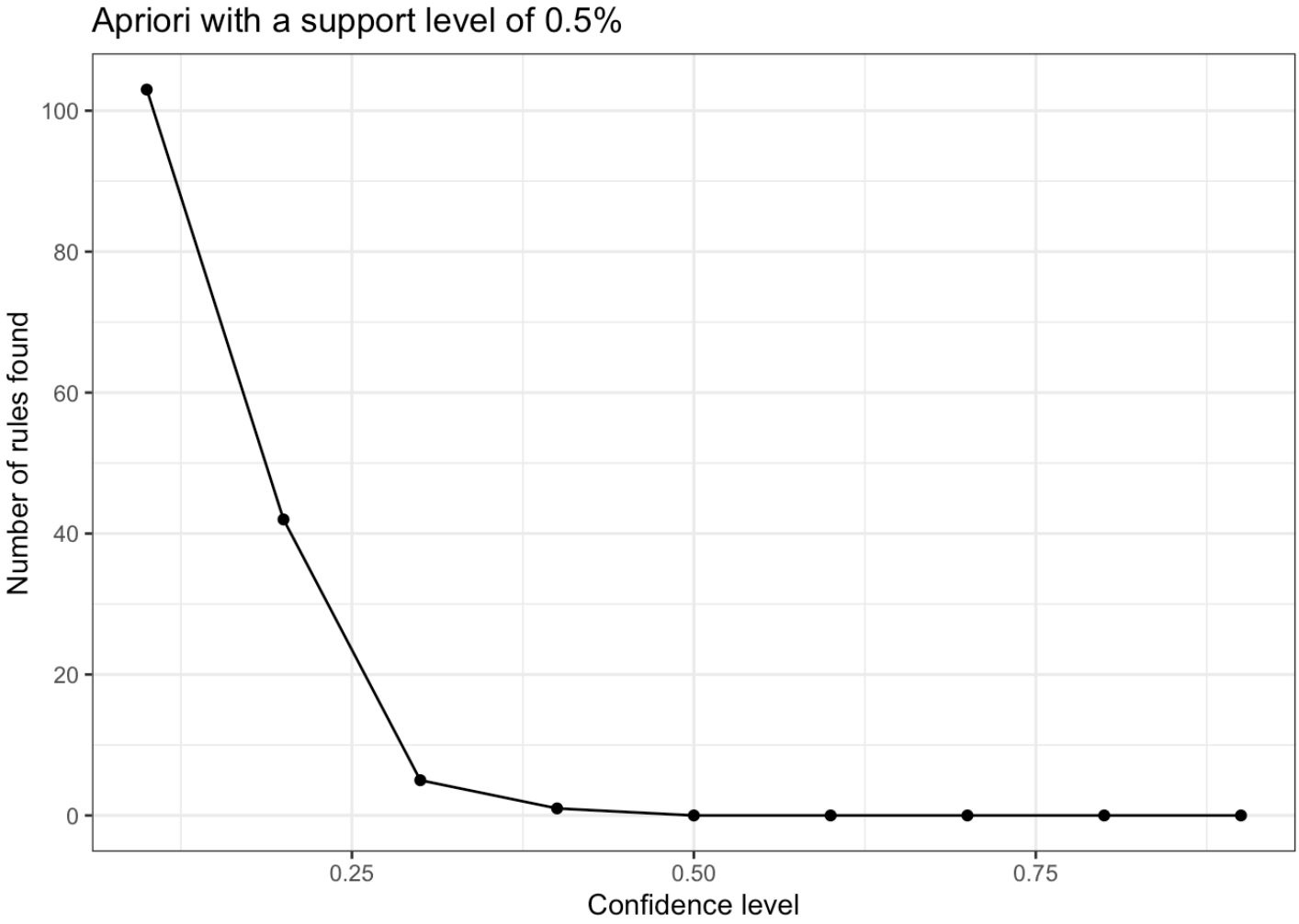
```
plot2
```


Apriori with a support level of 5%
Number of rules found
2
0
0.25
0.50
0.75
Confidence level

```
plot3
```


Apriori with a support level of 1%
Number of rules found
20
10
0
0.25
0.50
0.75
Confidence level

```
plot4
```




```
rules1 <- apriori(transactions, parameter = list(supp = 0.1, conf = 0.5, maxlen=3), c
ontrol = list(verbose = FALSE))
as(rules1,"data.frame")
```

```
## data frame with 0 columns and 0 rows
```

```
rules2 <- apriori(transactions, parameter = list(supp = 0.001, conf = 0.4, maxlen=3),
control = list(verbose = FALSE))
as(rules2,"data.frame")
```

```
##
rules
## 1
{Zero Calorie Cola} => {Soda}
## 2                          {Non Fat Acai & Mixed Berries Yogurt} => {Icelandic St
yle Skyr Blueberry Non-fat Yogurt}
## 3                      {Nonfat Icelandic Style Strawberry Yogurt} => {Icelandic St
```

```
yle Skyr Blueberry Non-fat Yogurt}
## 4                                                                    {Organic Ye
llow Squash} => {Organic Zucchini}
## 5                                      {Non Fat Raspberry Yogurt} => {Icelandic St
yle Skyr Blueberry Non-fat Yogurt}
## 6  {Total 2% Lowfat Greek Strained Yogurt With Blueberry} => {Total 2% with Strawb
erry Lowfat Greek Strained Yogurt}
## 7
{Organic Fuji Apple} => {Banana}
## 8           {Non Fat Raspberry Yogurt,Vanilla Skyr Nonfat Yogurt} => {Icelandic St
yle Skyr Blueberry Non-fat Yogurt}
## 9           {Icelandic Style Skyr Blueberry Non-fat Yogurt,Non Fat Raspberry Yogur
t} => {Vanilla Skyr Nonfat Yogurt}
## 10          {Icelandic Style Skyr Blueberry Non-fat Yogurt,Vanilla Skyr Nonfat Yog
urt} => {Non Fat Raspberry Yogurt}
## 11                                    {Lime Sparkling Water,Sparkling Lemon Wate
r} => {Sparkling Water Grapefruit}
## 12                                    {Sparkling Lemon Water,Sparkling Water Gra
pefruit} => {Lime Sparkling Water}
## 13                                                                  {Broccoli
Crown,Organic Avocado} => {Banana}
## 14                                        {Organic Hass Avocado,Organic Navel O
range} => {Bag of Organic Bananas}
## 15                                {Organic Hass Avocado,Organic Lacinato (Dinosaur)
Kale} => {Bag of Organic Bananas}
## 16                                                    {Organic Kiwi,Organic Raspbe
rries} => {Bag of Organic Bananas}
## 17                                                   {Organic Hass Avocado,Organic
Kiwi} => {Bag of Organic Bananas}
## 18                                 {Organic Hass Avocado,Organic Unsweetened Almond
Milk} => {Bag of Organic Bananas}
## 19                                                                  {Organic Av
ocado,Red Vine Tomato} => {Banana}
## 20                                      {Organic D'Anjou Pears,Organic Hass Av
ocado} => {Bag of Organic Bananas}
## 21                                   {Organic Hass Avocado,Organic Whole String C
heese} => {Bag of Organic Bananas}
## 22                                                                    {Organic
Avocado,Yellow Onions} => {Banana}
## 23                                       {Organic Gala Apples,Organic Hass Av
ocado} => {Bag of Organic Bananas}
## 24                                                                  {Honeycri
sp Apple,Strawberries} => {Banana}
## 25                                                                   {Honeycr
isp Apple,Large Lemon} => {Banana}
## 26                                                                {Honeycrisp
Apple,Organic Avocado} => {Banana}
```

```
## 27                            {Organic Large Extra Fancy Fuji Apple,Organic Raspbe
rries} => {Bag of Organic Bananas}
## 28                           {Organic Hass Avocado,Organic Large Extra Fancy Fuji
Apple} => {Bag of Organic Bananas}
## 29                                                                {Organic Fuji Appl
e,Seedless Red Grapes} => {Banana}
## 30                                                                   {Organic Avocad
o,Seedless Red Grapes} => {Banana}
## 31                                                                       {Organic Fu
ji Apple,Strawberries} => {Banana}
## 32                                                                        {Large Lem
on,Organic Fuji Apple} => {Banana}
## 33                                                                     {Organic Avoca
do,Organic Fuji Apple} => {Banana}
## 34                                          {Apple Honeycrisp Organic,Organic Hass Av
ocado} => {Bag of Organic Bananas}
## 35                                             {Organic Cucumber,Organic Raspbe
rries} => {Bag of Organic Bananas}
## 36                                            {Organic Cucumber,Organic Hass Av
ocado} => {Bag of Organic Bananas}
## 37                                               {Organic Lemon,Organic Raspbe
rries} => {Bag of Organic Bananas}
## 38                                                                          {Cucumb
er Kirby,Strawberries} => {Banana}
## 39                                                                       {Cucumber
Kirby,Organic Avocado} => {Banana}
## 40                                                                {Cucumber Kirby
,Organic Strawberries} => {Banana}
## 41                                                                   {Organic Wh
ole Milk,Strawberries} => {Banana}
## 42                                                                    {Organic Avoca
do,Organic Whole Milk} => {Banana}
## 43                                                                          {Lar
ge Lemon,Strawberries} => {Banana}
## 44                                                                       {Organic
Avocado,Strawberries} => {Banana}
## 45                                          {Organic Hass Avocado,Organic Raspbe
rries} => {Bag of Organic Bananas}
##        support confidence    coverage      lift count
## 1  0.001407542  0.4663774 0.003018033 44.247487   430
## 2  0.001044200  0.4455307 0.002343722 70.231321   319
## 3  0.001603944  0.4618285 0.003473029 72.800418   490
## 4  0.001702144  0.4873477 0.003492669 13.870250   520
## 5  0.002343722  0.4789298 0.004893665 75.496185   716
## 6  0.002360089  0.4064262 0.005806931 48.748320   721
## 7  0.011224333  0.4014752 0.027957721  2.714205  3429
## 8  0.001073660  0.6319846 0.001698871 99.623011   328
```

```
## 9  0.001073660  0.4581006 0.002343722 76.768155   328
## 10 0.001073660  0.4568245 0.002350269 93.350179   328
## 11 0.001286428  0.4930991 0.002608864 21.800333   393
## 12 0.001286428  0.4673008 0.002752891 34.103918   393
## 13 0.001018013  0.4748092 0.002144047  3.209984   311
## 14 0.001001646  0.4580838 0.002186601  3.864127   306
## 15 0.001044200  0.4236388 0.002464836  3.573569   319
## 16 0.001054020  0.4417010 0.002386275  3.725931   322
## 17 0.001243875  0.4352806 0.002857639  3.671773   380
## 18 0.001214415  0.4279123 0.002837998  3.609618   371
## 19 0.001024560  0.4190094 0.002445196  2.832746   313
## 20 0.001453369  0.4758842 0.003054040  4.014281   444
## 21 0.001528657  0.4364486 0.003502489  3.681625   467
## 22 0.001122761  0.4122596 0.002723431  2.787113   343
## 23 0.001705418  0.4422750 0.003856012  3.730774   521
## 24 0.001103120  0.5121581 0.002153867  3.462484   337
## 25 0.001227508  0.4040948 0.003037673  2.731915   375
## 26 0.001489376  0.4430380 0.003361735  2.995193   455
## 27 0.001358442  0.4424307 0.003070407  3.732087   415
## 28 0.002036027  0.4624535 0.004402662  3.900988   622
## 29 0.001011467  0.5065574 0.001996746  3.424621   309
## 30 0.001004920  0.4336158 0.002317535  2.931494   307
## 31 0.001191501  0.4932249 0.002415736  3.334486   364
## 32 0.001083480  0.4332461 0.002500843  2.928994   331
## 33 0.001306068  0.4623407 0.002824905  3.125690   399
## 34 0.001967286  0.4082880 0.004818378  3.444079   601
## 35 0.001204594  0.4269142 0.002821632  3.601198   368
## 36 0.002216061  0.4204969 0.005270101  3.547066   677
## 37 0.001364989  0.4052478 0.003368282  3.418434   417
## 38 0.001243875  0.4871795 0.002553217  3.293615   380
## 39 0.001865812  0.4209749 0.004432122  2.846034   570
## 40 0.001328982  0.4076305 0.003260261  2.755818   406
## 41 0.001201321  0.4503067 0.002667784  3.044334   367
## 42 0.001541750  0.4081456 0.003777451  2.759300   471
## 43 0.001623584  0.4563017 0.003558136  3.084864   496
## 44 0.001545023  0.4334252 0.003564683  2.930205   472
## 45 0.003623604  0.4483597 0.008081912  3.782100  1107
```

```
rules3 <- apriori(transactions, parameter = list(supp = 0.005, conf = 0.1, maxlen=3),
control = list(verbose = FALSE))
as(rules3,"data.frame")
```

```
##                                                    rules
## 1                                {} => {Bag of Organic Bananas}
## 2                                                {} => {Banana}
## 3                                 {Red Vine Tomato} => {Banana}
```

```
## 4                                          {Red Peppers} => {Banana}
## 5                                          {Half & Half} => {Banana}
## 6                              {Sparkling Water Grapefruit} => {Banana}
## 7                                            {Yellow Onions} => {Banana}
## 8                                         {Organic Cilantro} => {Limes}
## 9                                         {Limes} => {Organic Cilantro}
## 10                    {Organic Gala Apples} => {Bag of Organic Bananas}
## 11                                  {Organic Gala Apples} => {Banana}
## 12                                     {Honeycrisp Apple} => {Banana}
## 13                                      {Original Hummus} => {Banana}
## 14  {Organic Large Extra Fancy Fuji Apple} => {Bag of Organic Bananas}
## 15                                  {Organic Baby Carrots} => {Banana}
## 16                                   {Seedless Red Grapes} => {Banana}
## 17                                    {Organic Fuji Apple} => {Banana}
## 18                {Apple Honeycrisp Organic} => {Organic Strawberries}
## 19              {Apple Honeycrisp Organic} => {Bag of Organic Bananas}
## 20                            {Apple Honeycrisp Organic} => {Banana}
## 21                       {Organic Cucumber} => {Organic Hass Avocado}
## 22                       {Organic Cucumber} => {Organic Baby Spinach}
## 23                       {Organic Cucumber} => {Organic Strawberries}
## 24                     {Organic Cucumber} => {Bag of Organic Bananas}
## 25                          {Organic Lemon} => {Organic Hass Avocado}
## 26                        {Organic Lemon} => {Bag of Organic Bananas}
## 27                  {Organic Grape Tomatoes} => {Organic Baby Spinach}
## 28                               {Organic Grape Tomatoes} => {Banana}
## 29                                        {Cucumber Kirby} => {Banana}
## 30                     {Organic Blueberries} => {Organic Strawberries}
## 31                   {Organic Blueberries} => {Bag of Organic Bananas}
## 32                                {Organic Blueberries} => {Banana}
## 33                       {Organic Zucchini} => {Organic Hass Avocado}
## 34                       {Organic Zucchini} => {Organic Baby Spinach}
## 35                       {Organic Zucchini} => {Organic Strawberries}
## 36                     {Organic Zucchini} => {Bag of Organic Bananas}
## 37                                   {Organic Zucchini} => {Banana}
## 38                         {Organic Garlic} => {Organic Yellow Onion}
## 39                         {Organic Yellow Onion} => {Organic Garlic}
## 40                                 {Organic Garlic} => {Large Lemon}
## 41                                 {Large Lemon} => {Organic Garlic}
## 42                         {Organic Garlic} => {Organic Hass Avocado}
## 43                         {Organic Garlic} => {Organic Baby Spinach}
## 44                       {Organic Garlic} => {Bag of Organic Bananas}
## 45                                    {Organic Garlic} => {Banana}
## 46                    {Organic Yellow Onion} => {Organic Hass Avocado}
## 47                    {Organic Yellow Onion} => {Organic Baby Spinach}
## 48                    {Organic Yellow Onion} => {Organic Strawberries}
## 49                  {Organic Yellow Onion} => {Bag of Organic Bananas}
## 50                                {Organic Yellow Onion} => {Banana}
```

```
## 51                    {Organic Whole Milk} => {Organic Baby Spinach}
## 52                    {Organic Whole Milk} => {Organic Strawberries}
## 53                  {Organic Whole Milk} => {Bag of Organic Bananas}
## 54                                {Organic Whole Milk} => {Banana}
## 55                      {Strawberries} => {Bag of Organic Bananas}
## 56                                   {Strawberries} => {Banana}
## 57                   {Organic Raspberries} => {Organic Hass Avocado}
## 58                   {Organic Hass Avocado} => {Organic Raspberries}
## 59                   {Organic Raspberries} => {Organic Baby Spinach}
## 60                   {Organic Raspberries} => {Organic Strawberries}
## 61                   {Organic Strawberries} => {Organic Raspberries}
## 62                 {Organic Raspberries} => {Bag of Organic Bananas}
## 63                 {Bag of Organic Bananas} => {Organic Raspberries}
## 64                               {Organic Raspberries} => {Banana}
## 65                                        {Limes} => {Large Lemon}
## 66                                        {Large Lemon} => {Limes}
## 67                                    {Limes} => {Organic Avocado}
## 68                                    {Organic Avocado} => {Limes}
## 69                                  {Limes} => {Organic Hass Avocado}
## 70                                  {Organic Hass Avocado} => {Limes}
## 71                                  {Limes} => {Organic Baby Spinach}
## 72                                  {Limes} => {Organic Strawberries}
## 73                                {Limes} => {Bag of Organic Bananas}
## 74                                               {Limes} => {Banana}
## 75                                {Large Lemon} => {Organic Avocado}
## 76                                {Organic Avocado} => {Large Lemon}
## 77                              {Large Lemon} => {Organic Baby Spinach}
## 78                              {Large Lemon} => {Organic Strawberries}
## 79                            {Large Lemon} => {Bag of Organic Bananas}
## 80                                           {Large Lemon} => {Banana}
## 81                          {Organic Avocado} => {Organic Baby Spinach}
## 82                          {Organic Baby Spinach} => {Organic Avocado}
## 83                          {Organic Avocado} => {Organic Strawberries}
## 84                        {Organic Avocado} => {Bag of Organic Bananas}
## 85                                       {Organic Avocado} => {Banana}
## 86                                       {Banana} => {Organic Avocado}
## 87                    {Organic Hass Avocado} => {Organic Baby Spinach}
## 88                    {Organic Baby Spinach} => {Organic Hass Avocado}
## 89                    {Organic Hass Avocado} => {Organic Strawberries}
## 90                    {Organic Strawberries} => {Organic Hass Avocado}
## 91                  {Organic Hass Avocado} => {Bag of Organic Bananas}
## 92                  {Bag of Organic Bananas} => {Organic Hass Avocado}
## 93                               {Organic Hass Avocado} => {Banana}
## 94                    {Organic Baby Spinach} => {Organic Strawberries}
## 95                    {Organic Strawberries} => {Organic Baby Spinach}
## 96                  {Organic Baby Spinach} => {Bag of Organic Bananas}
## 97                  {Bag of Organic Bananas} => {Organic Baby Spinach}
```

```
## 98                                  {Organic Baby Spinach} => {Banana}
## 99                                  {Banana} => {Organic Baby Spinach}
## 100                {Organic Strawberries} => {Bag of Organic Bananas}
## 101                {Bag of Organic Bananas} => {Organic Strawberries}
## 102                                 {Organic Strawberries} => {Banana}
## 103                                 {Banana} => {Organic Strawberries}
##         support confidence   coverage      lift count
## 1   0.118547809  0.1185478 1.00000000 1.0000000 36216
## 2   0.147916346  0.1479163 1.00000000 1.0000000 45188
## 3   0.005001686  0.2940158 0.01701162 1.9877166  1528
## 4   0.005070426  0.2834919 0.01788561 1.9165694  1549
## 5   0.005260281  0.2525141 0.02083163 1.7071416  1607
## 6   0.005044239  0.2230101 0.02261888 1.5076774  1541
## 7   0.006392861  0.2920156 0.02189220 1.9741939  1953
## 8   0.005437042  0.2570014 0.02115569 5.7472480  1661
## 9   0.005437042  0.1215870 0.04471730 5.7472480  1661
## 10  0.005856031  0.2607492 0.02245849 2.1995275  1789
## 11  0.005420675  0.2413642 0.02245849 1.6317617  1656
## 12  0.008487808  0.3524055 0.02408534 2.3824652  2593
## 13  0.006016426  0.2676179 0.02248140 1.8092520  1838
## 14  0.007689110  0.3218690 0.02388894 2.7150987  2349
## 15  0.005885492  0.2332641 0.02523102 1.5770004  1798
## 16  0.007662923  0.2968552 0.02581367 2.0069127  2341
## 17  0.011224333  0.4014752 0.02795772 2.7142047  3429
## 18  0.005397762  0.1986508 0.02717212 2.3872866  1649
## 19  0.007024619  0.2585231 0.02717212 2.1807495  2146
## 20  0.005574523  0.2051560 0.02717212 1.3869732  1703
## 21  0.005270101  0.2170104 0.02428502 3.2231037  1610
## 22  0.005021326  0.2067664 0.02428502 2.7479235  1534
## 23  0.005371575  0.2211888 0.02428502 2.6581380  1641
## 24  0.006363401  0.2620299 0.02428502 2.2103312  1944
## 25  0.006576169  0.2447016 0.02687424 3.6343818  2009
## 26  0.006900231  0.2567600 0.02687424 2.1658776  2108
## 27  0.005103160  0.1920424 0.02657309 2.5522412  1559
## 28  0.006153907  0.2315841 0.02657309 1.5656426  1880
## 29  0.009672763  0.3362922 0.02876297 2.2735300  2955
## 30  0.007456702  0.2305201 0.03234729 2.7702770  2278
## 31  0.007014799  0.2168589 0.03234729 1.8292952  2143
## 32  0.005810204  0.1796195 0.03234729 1.2143317  1775
## 33  0.005230821  0.1488727 0.03513619 2.2111029  1598
## 34  0.006946058  0.1976896 0.03513619 2.6272926  2122
## 35  0.005329021  0.1516676 0.03513619 1.8226662  1628
## 36  0.007525442  0.2141792 0.03513619 1.8066909  2299
## 37  0.006798757  0.1934973 0.03513619 1.3081536  2077
## 38  0.007090086  0.2037246 0.03480231 5.6749572  2166
## 39  0.007090086  0.1975016 0.03589888 5.6749572  2166
## 40  0.005116253  0.1470090 0.03480231 3.0253161  1563
```

```
## 41  0.005116253  0.1052880 0.04859295 3.0253161  1563
## 42  0.005659630  0.1626223 0.03480231 2.4153151  1729
## 43  0.006870771  0.1974229 0.03480231 2.6237480  2099
## 44  0.007044259  0.2024078 0.03480231 1.7073941  2152
## 45  0.005623623  0.1615877 0.03480231 1.0924260  1718
## 46  0.006369948  0.1774414 0.03589888 2.6354135  1946
## 47  0.005901858  0.1644023 0.03589888 2.1849049  1803
## 48  0.005541789  0.1543722 0.03589888 1.8551688  1693
## 49  0.007898605  0.2200237 0.03589888 1.8559913  2413
## 50  0.005211180  0.1451628 0.03589888 0.9813842  1592
## 51  0.005659630  0.1333385 0.04244559 1.7720670  1729
## 52  0.007450155  0.1755225 0.04244559 2.1093423  2276
## 53  0.007859324  0.1851623 0.04244559 1.5619212  2401
## 54  0.010458368  0.2463947 0.04244559 1.6657706  3195
## 55  0.005319201  0.1221345 0.04355198 1.0302555  1625
## 56  0.012716982  0.2919955 0.04355198 1.9740583  3885
## 57  0.008081912  0.1796290 0.04499226 2.6679035  2469
## 58  0.008081912  0.1200350 0.06732963 2.6679035  2469
## 59  0.006016426  0.1337214 0.04499226 1.7771554  1838
## 60  0.011532028  0.2563114 0.04499226 3.0802234  3523
## 61  0.011532028  0.1385862 0.08321195 3.0802234  3523
## 62  0.013473127  0.2994543 0.04499226 2.5260218  4116
## 63  0.013473127  0.1136514 0.11854781 2.5260218  4116
## 64  0.007770944  0.1727174 0.04499226 1.1676691  2374
## 65  0.008851151  0.1979357 0.04471730 4.0733426  2704
## 66  0.008851151  0.1821489 0.04859295 4.0733426  2704
## 67  0.006500882  0.1453774 0.04471730 2.8363996  1986
## 68  0.006500882  0.1268361 0.05125419 2.8363996  1986
## 69  0.006926418  0.1548935 0.04471730 2.3005249  2116
## 70  0.006926418  0.1028733 0.06732963 2.3005249  2116
## 71  0.006913325  0.1546007 0.04471730 2.0546416  2112
## 72  0.006160453  0.1377644 0.04471730 1.6555849  1882
## 73  0.007554902  0.1689481 0.04471730 1.4251474  2308
## 74  0.009679309  0.2164556 0.04471730 1.4633650  2957
## 75  0.007446882  0.1532503 0.04859295 2.9900046  2275
## 76  0.007446882  0.1452931 0.05125419 2.9900046  2275
## 77  0.007342134  0.1510946 0.04859295 2.0080463  2243
## 78  0.005037693  0.1036713 0.04859295 1.2458700  1539
## 79  0.006055706  0.1246211 0.04859295 1.0512306  1850
## 80  0.013129425  0.2701920 0.04859295 1.8266540  4011
## 81  0.009142479  0.1783753 0.05125419 2.3706056  2793
## 82  0.009142479  0.1215035 0.07524460 2.3706056  2793
## 83  0.006919872  0.1350109 0.05125419 1.6224937  2114
## 84  0.006651456  0.1297739 0.05125419 1.0946969  2032
## 85  0.015908503  0.3103845 0.05125419 2.0983784  4860
## 86  0.015908503  0.1075507 0.14791635 2.0983784  4860
## 87  0.011276706  0.1674851 0.06732963 2.2258746  3445
```

```
## 88  0.011276706  0.1498673 0.07524460 2.2258746  3445
## 89  0.013361833  0.1984540 0.06732963 2.3849218  4082
## 90  0.013361833  0.1605759 0.08321195 2.3849218  4082
## 91  0.019823435  0.2944236 0.06732963 2.4835857  6056
## 92  0.019823435  0.1672189 0.11854781 2.4835857  6056
## 93  0.009613842  0.1427877 0.06732963 0.9653273  2937
## 94  0.012556588  0.1668769 0.07524460 2.0054444  3836
## 95  0.012556588  0.1508989 0.08321195 2.0054444  3836
## 96  0.016284939  0.2164267 0.07524460 1.8256489  4975
## 97  0.016284939  0.1373702 0.11854781 1.8256489  4975
## 98  0.015365126  0.2042024 0.07524460 1.3805261  4694
## 99  0.015365126  0.1038771 0.14791635 1.3805261  4694
## 100 0.018442080  0.2216278 0.08321195 1.8695224  5634
## 101 0.018442080  0.1555666 0.11854781 1.8695224  5634
## 102 0.017558274  0.2110066 0.08321195 1.4265269  5364
## 103 0.017558274  0.1187041 0.14791635 1.4265269  5364
```

```
plot(rules2, method="paracoord",  control=list(alpha=.5, reorder=TRUE))
```

Parallel coordinates plot for 45 rules