Would You Like a Cookie With That Coffee? A Basket Analisys on Data From a Bakery

Michał Szałański

20 marca 2019

```
library(tibble)
library(dplyr)
library(tidyr)
library(ggplot2)
library(gridExtra)
library(knitr)
library(arules)
library(arulesViz)
library(arulesCBA)
library(arulesSequences)
library(knitr)
library(reshape)
options(scipen=999)

# Nice colors
cYellow = '#FADA5E'
cBlue = '#378CC7'
```

```
# Read the data
dataImport <- read.csv('data/teaBasket_DMS.csv')

# Remove transactions with NONE and Adjustment
dataExport <- dataImport %>% filter(dataImport$Item != "NONE" & dataImport$Item != "Adjustment")

# Remove two most frequent items
dataExportNoCoffee <- dataImport %>% filter(!(dataImport$Item %in% c("NONE", "Adjustment", "Coffee", "B

# Write only the relevant columns
dataExport[, c(3:4)] %>% write.csv( './data/transactions.csv')
dataExportNoCoffee[, c(3:4)] %>% write.csv( './data/transactionsNoCoffee.csv')
```

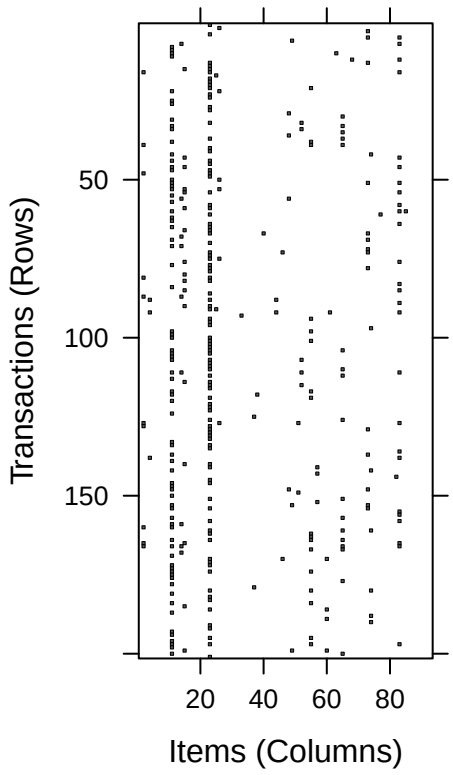
```
# Read the data as transactions
coffee <- read.transactions('./data/transactions.csv',
                            format="single",
                            cols = c('Transaction', 'Item'),
                            sep = ',',
                            header = TRUE)

tea <- read.transactions('./data/transactionsNoCoffee.csv',
                             format="single",
                             cols = c('Transaction', 'Item'),
                             sep = ',',
                             header = TRUE)
```

With coffee

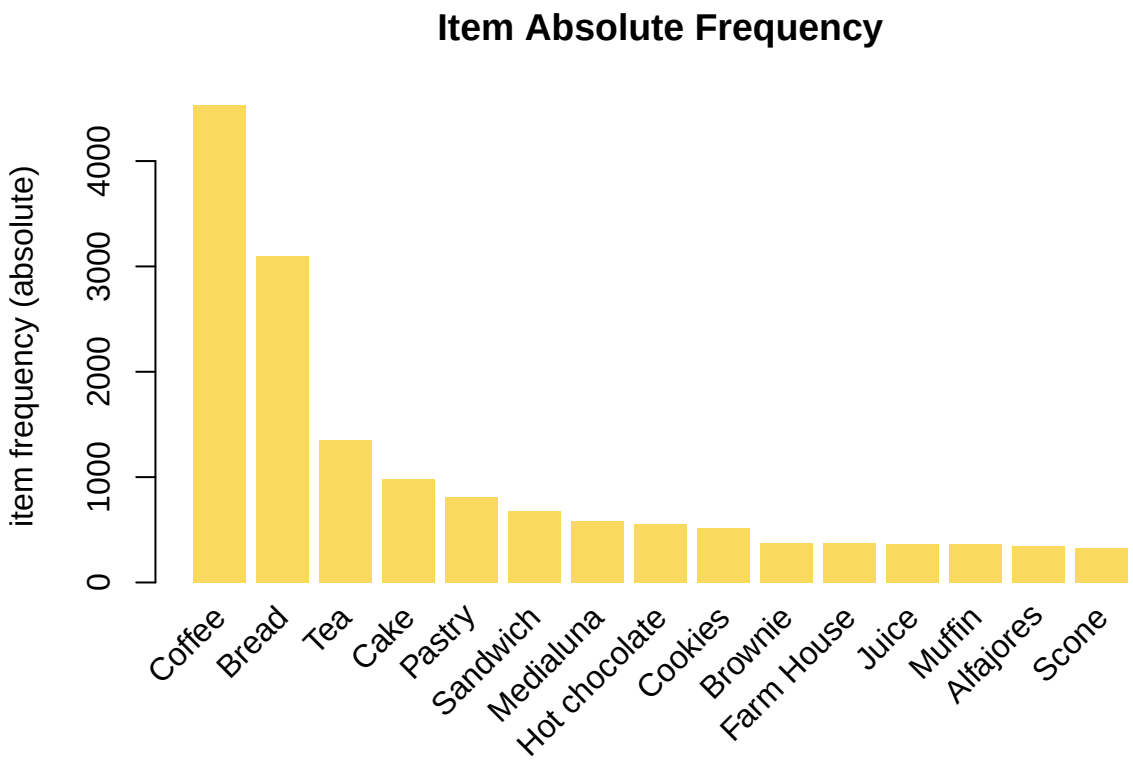
```
coffee
```

```
transactions in sparse format with
 9464 transactions (rows) and
 93 items (columns)
```

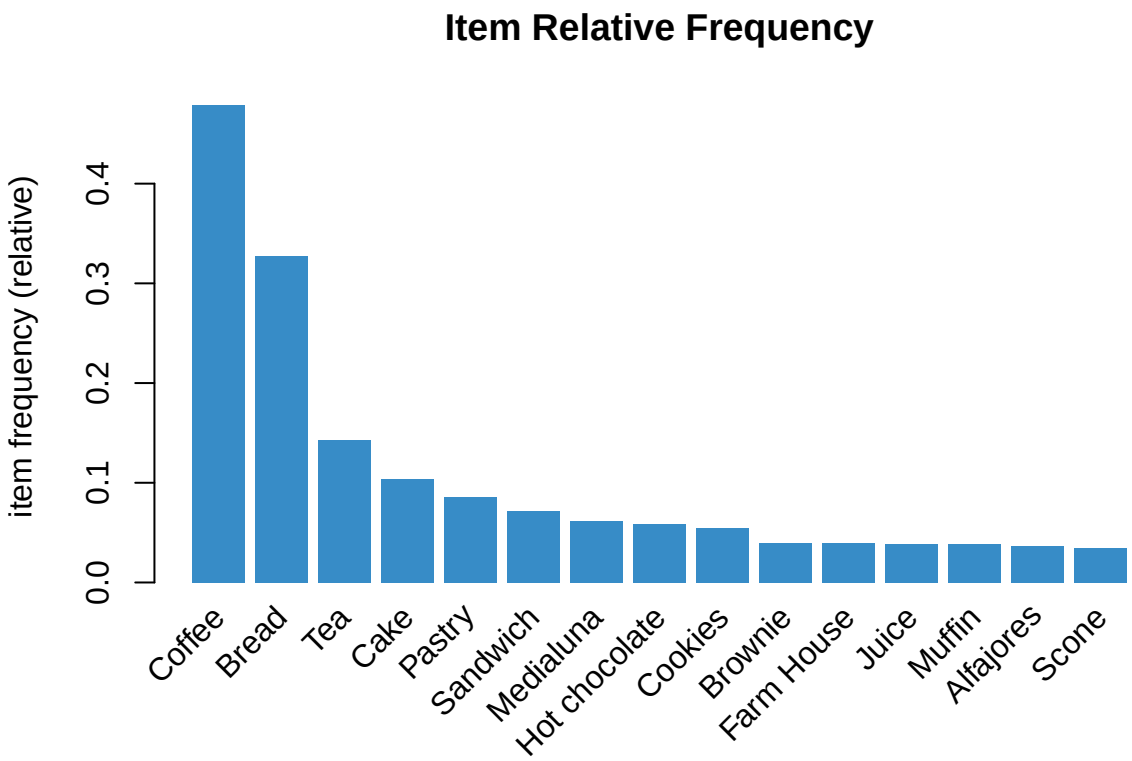
```
image(coffee[1000:1200])
```




```
coffee %>% itemFrequencyPlot(topN=15,
                             type="absolute",
                             main="Item Absolute Frequency",
                             col = cYellow,
                             border = NA)
```




```
coffee %>% itemFrequencyPlot(topN=15,
                             type="relative",
                             main="Item Relative Frequency",
                             col = cBlue,
                             border = NA)
```

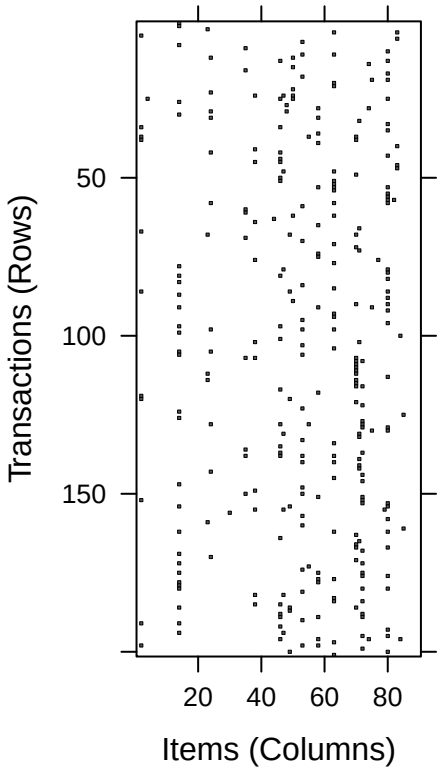


Without coffee and bread

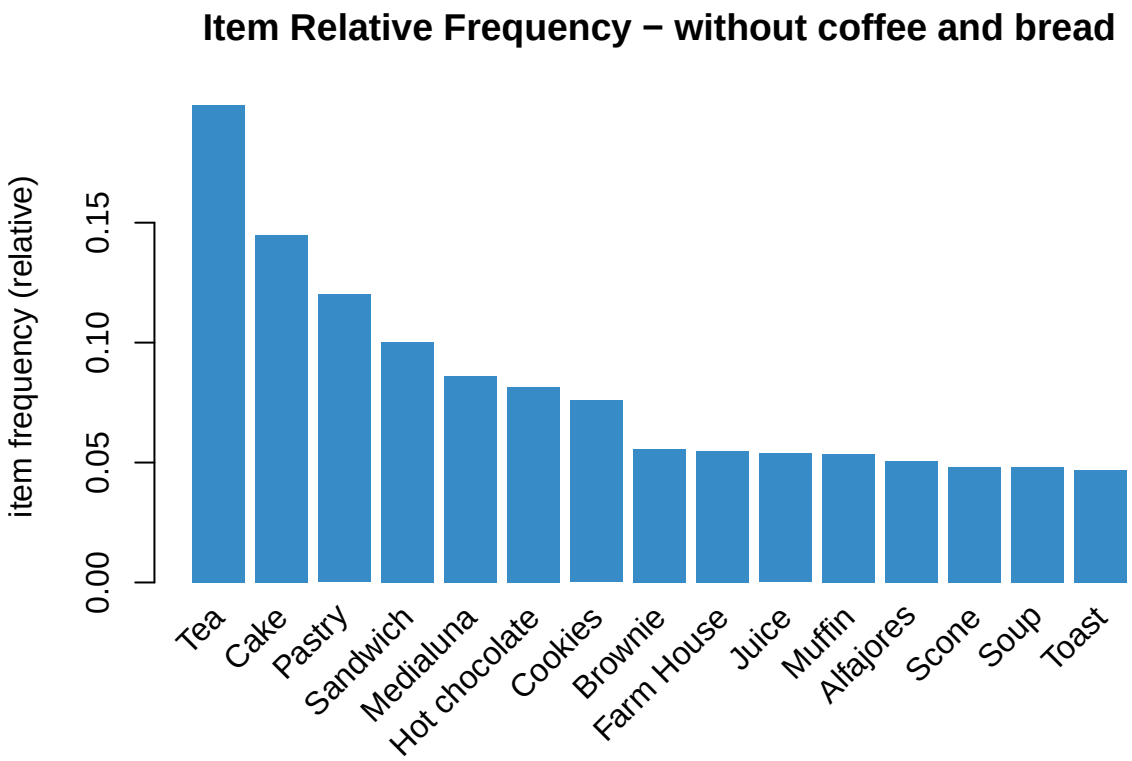
```
tea
```

```
transactions in sparse format with
 6788 transactions (rows) and
 90 items (columns)
```

```
image(tea[1000:1200])
```




```
tea %>% itemFrequencyPlot(topN=15,
                          type="relative",
                          main="Item Relative Frequency - without coffee and bread",
                          col = cBlue,
                          border = NA)
```



Cross tables

```
coffeeCount <- coffee %>% crossTable(measure="count", sort=TRUE)
teaCount <- tea %>% crossTable(measure="count", sort=TRUE)
```

```
coffeeCount[1:8,1:6] %>% kable()
```

	Coffee	Bread	Tea	Cake	Pastry	Sandwich
Coffee	4528	852	472	518	450	362
Bread	852	3097	266	221	276	161
Tea	472	266	1350	225	91	136
Cake	518	221	225	983	49	65
Pastry	450	276	91	49	815	11
Sandwich	362	161	136	65	11	680
Medialuna	333	160	77	35	87	20
Hot chocolate	280	127	76	108	54	42

```
coffeeCount[1,1]
```

```
[1] 4528
```

```
coffeeCount[2,2]
```

```
[1] 3097
```

```
teaSupport <- tea %>% crossTable(measure="support", sort=TRUE)
teaLift <- tea %>% crossTable(measure="lift", sort=TRUE)
```

```
teaSupport[1:8,1:6] %>% kable()
```

	Tea	Cake	Pastry	Sandwich	Medialuna	Hot chocolate
Tea	0.1988804	0.0331467	0.0134060	0.0200354	0.0113435	0.0111962
Cake	0.0331467	0.1448144	0.0072186	0.0095757	0.0051562	0.0159104
Pastry	0.0134060	0.0072186	0.1200648	0.0016205	0.0128167	0.0079552
Sandwich	0.0200354	0.0095757	0.0016205	0.1001768	0.0029464	0.0061874
Medialuna	0.0113435	0.0051562	0.0128167	0.0029464	0.0861815	0.0066293
Hot chocolate	0.0111962	0.0159104	0.0079552	0.0061874	0.0066293	0.0813200
Cookies	0.0137006	0.0101650	0.0041249	0.0039776	0.0035357	0.0083972
Brownie	0.0094284	0.0061874	0.0033883	0.0032410	0.0027991	0.0057454

```
teaLift[1:8,1:6] %>% kable()
```

	Tea	Cake	Pastry	Sandwich	Medialuna	Hot chocolate
Tea	NA	1.1508986	0.5614251	1.0056296	0.6618246	0.6922813
Cake	1.1508986	NA	0.4151708	0.6600742	0.4131431	1.3510549
Pastry	0.5614251	0.4151708	NA	0.1347311	1.2386472	0.8147773
Sandwich	1.0056296	0.6600742	0.1347311	NA	0.3412770	0.7595269
Medialuna	0.6618246	0.4131431	1.2386472	0.3412770	NA	0.9459309
Hot chocolate	0.6922813	1.3510549	0.8147773	0.7595269	0.9459309	NA
Cookies	0.9079957	0.9251884	0.4528298	0.5233467	0.5407418	1.3610384
Brownie	0.8490804	0.7652413	0.5054438	0.5794506	0.5817017	1.2654009

Eclat

Low support still gives items that have at least ~ 8 occurences

```
teaFreqItems <- tea %>% eclat(list(supp=0.0003, maxlen=4))
```

```
Eclat

parameter specification:
 tidLists support minlen maxlen            target   ext
    FALSE  0.0003      1      4 frequent itemsets FALSE

algorithmic control:
 sparse sort verbose
      7   -2    TRUE

Absolute minimum support count: 2

create itemset ...
set transactions ...[90 item(s), 6788 transaction(s)] done [0.00s].
sorting and recoding items ... [77 item(s)] done [0.00s].
creating sparse bit matrix ... [77 row(s), 6788 column(s)] done [0.00s].
writing  ... [751 set(s)] done [0.00s].
Creating S4 object  ... done [0.00s].
```

```
median(teaFreqItems@quality$count)
```

```
[1] 6
```

```
teaFreqRules <- teaFreqItems %>% ruleInduction(tea, confidence=0.4)
teaFreqRules
```

```
set of 60 rules
```

```
a <- teaFreqRules %>%
  head(10) %>%
  inspect(ruleSep = ">>>", itemSep = " + ", setStart = "", setEnd ="") %>%
  as.data.frame()
```

```
kable(a)
```

	lhs		rhs	support	confidence	lift	itemset
[1]	Victorian Sponge	>>>	Tea	0.0005893	0.5714286	2.873228	1
[2]	Chocolates + Juice	>>>	Hot chocolate	0.0004420	0.7500000	9.222826	8
[3]	Chocolates + Hot chocolate	>>>	Juice	0.0004420	0.7500000	13.947945	8
[4]	Chocolates	>>>	Hot chocolate	0.0005893	0.4444444	5.465378	10
[5]	Chocolates	>>>	Juice	0.0005893	0.4444444	8.265449	11
[6]	Mineral water + Pick and Mix Bowls	>>>	Juice	0.0004420	1.0000000	18.597260	13
[7]	Juice + Pick and Mix Bowls	>>>	Mineral water	0.0004420	0.7500000	37.992537	13
[8]	Duck egg	>>>	Tea	0.0008839	0.5000000	2.514074	18
[9]	Duck egg	>>>	Spanish Brunch	0.0008839	0.5000000	19.732558	19
[10]	Crisps	>>>	Sandwich	0.0008839	0.4285714	4.278151	20

Apriori

```
b <- tea %>%
  apriori(list(supp=0.0004, conf=0.3), control=list(verbose=F)) %>%
  sort(by="lift", decreasing=TRUE) %>%
  head(10) %>%
  inspect(ruleSep = ">>>", itemSep = " + ", setStart = "", setEnd ="") %>%
  as.data.frame()
```

```
kable(b)
```

	lhs		rhs	support	confidence	lift	count
[1]	Juice + Salad	>>>	Extra Salami or Feta	0.0004420	0.3000000	53.58947	3
[2]	Extra Salami or Feta + Juice	>>>	Salad	0.0004420	0.7500000	51.42424	3
[3]	Juice + Pick and Mix Bowls	>>>	Mineral water	0.0004420	0.7500000	37.99254	3
[4]	Extra Salami or Feta + Sandwich	>>>	Salad	0.0004420	0.5000000	34.28283	3
[5]	Cake + Extra Salami or Feta	>>>	Salad	0.0004420	0.4285714	29.38528	3
[6]	Extra Salami or Feta	>>>	Salad	0.0023571	0.4210526	28.86975	16
[7]	Chicken Stew + Salad	>>>	Truffles	0.0004420	0.7500000	26.51562	3
[8]	Extra Salami or Feta + Spanish Brunch	>>>	Salad	0.0004420	0.3750000	25.71212	3
[9]	Scone + Truffles	>>>	Mineral water	0.0005893	0.5000000	25.32836	4
[10]	Salad + Truffles	>>>	Chicken Stew	0.0004420	0.4285714	23.65157	3

LHS

Coffee

```
c <- coffee %>%
  apriori(list(supp=0.001,conf = 0.10),
          appearance=list(default="lhs", rhs="Coffee"),
          control=list(verbose=F)) %>%
  sort(by="confidence", decreasing=TRUE) %>%
  head(10) %>%
  inspect(ruleSep = ">>>", itemSep = " + ", setStart = "", setEnd ="") %>%
  as.data.frame()
```

```
kable(c)
```

	lhs		rhs	support	confidence	lift	count
[1]	Extra Salami or Feta + Salad	»>	Coffee	0.0014793	0.8750000	1.828843	14
[2]	Pastry + Toast	»>	Coffee	0.0013736	0.8666667	1.811425	13
[3]	Hearty & Seasonal + Sandwich	»>	Coffee	0.0012680	0.8571429	1.791519	12
[4]	Cake + Vegan mincepie	»>	Coffee	0.0010566	0.8333333	1.741755	10
[5]	Salad + Sandwich	»>	Coffee	0.0015850	0.8333333	1.741755	15
[6]	Extra Salami or Feta	»>	Coffee	0.0032756	0.8157895	1.705086	31
[7]	Keeping It Local	»>	Coffee	0.0053888	0.8095238	1.691991	51
[8]	Cookies + Scone	»>	Coffee	0.0015850	0.7894737	1.650084	15
[9]	Juice + Pastry	»>	Coffee	0.0017963	0.7727273	1.615082	17
[10]	Cake + Salad	»>	Coffee	0.0010566	0.7692308	1.607774	10

Bread

```
d <- coffee %>%
  apriori(list(supp=0.001,conf = 0.10),
          appearance=list(default="lhs", rhs="Bread"),
          control=list(verbose=F)) %>%
  sort(by="confidence", decreasing=TRUE) %>%
  head(10) %>%
  inspect(ruleSep = ">>>", itemSep = " + ", setStart = "", setEnd ="") %>%
  as.data.frame()
```

```
kable(d)
```

	lhs		rhs	support	confidence	lift	count
[1]	Cake + Jammie Dodgers	»>	Bread	0.0015850	0.5172414	1.580618	15
[2]	Eggs	»>	Bread	0.0014793	0.5000000	1.527930	14
[3]	Jammie Dodgers + Tea	»>	Bread	0.0010566	0.4166667	1.273275	10
[4]	Hot chocolate + Scone	»>	Bread	0.0011623	0.3928571	1.200517	11
[5]	Alfajores + Brownie	»>	Bread	0.0010566	0.3703704	1.131800	10
[6]	Alfajores + Medialuna	»>	Bread	0.0011623	0.3666667	1.120482	11
[7]	Tea + Tiffin	»>	Bread	0.0012680	0.3636364	1.111222	12
[8]	Jammie Dodgers	»>	Bread	0.0046492	0.3520000	1.075663	44
[9]	Focaccia	»>	Bread	0.0020076	0.3518519	1.075210	19
[10]	Pastry	»>	Bread	0.0291631	0.3386503	1.034868	276

RHS

Coffee

```
e <- coffee %>%
  apriori(list(supp=0.001,conf = 0.05),
          appearance=list(default="rhs", lhs="Coffee"),
          control=list(verbose=F)) %>%
  sort(by="confidence", decreasing=TRUE) %>%
  head(10) %>%
  inspect(ruleSep = ">>>", itemSep = " + ", setStart = "", setEnd ="") %>%
  as.data.frame()
```

```
kable(e)
```

	lhs		rhs	support	confidence	lift	count
[1]		»>	Bread	0.3272401	0.3272401	1.0000000	3097
[2]	Coffee	»>	Bread	0.0900254	0.1881625	0.5749985	852
[3]		»>	Tea	0.1426458	0.1426458	1.0000000	1350
[4]	Coffee	»>	Cake	0.0547337	0.1143993	1.1013987	518
[5]	Coffee	»>	Tea	0.0498732	0.1042403	0.7307630	472
[6]		»>	Cake	0.1038673	0.1038673	1.0000000	983
[7]	Coffee	»>	Pastry	0.0475486	0.0993816	1.1540463	450
[8]		»>	Pastry	0.0861158	0.0861158	1.0000000	815
[9]	Coffee	»>	Sandwich	0.0382502	0.0799470	1.1126741	362
[10]	Coffee	»>	Medialuna	0.0351860	0.0735424	1.1897527	333

Bread

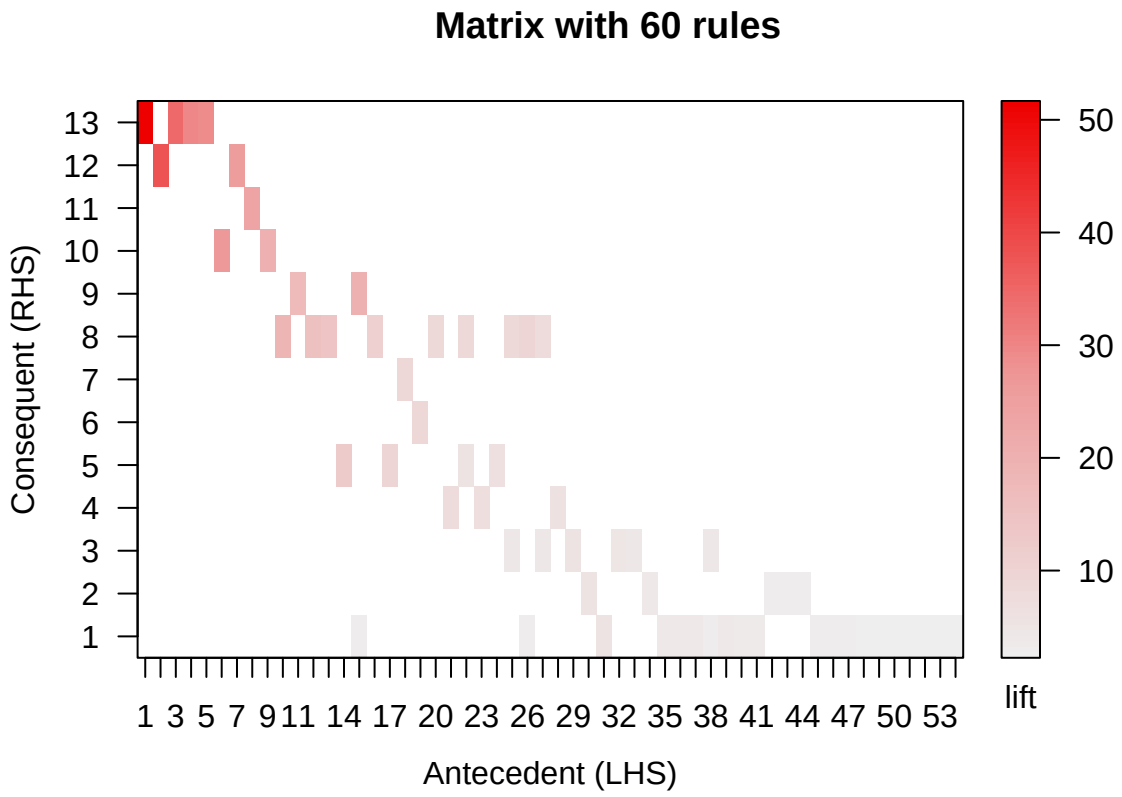
```
f <- coffee %>%
  apriori(list(supp=0.001,conf = 0.05),
          appearance=list(default="rhs", lhs="Bread"),
          control=list(verbose=F)) %>%
  sort(by="confidence", decreasing=TRUE) %>%
  head(10) %>%
  inspect(ruleSep = ">>>", itemSep = " + ", setStart = "", setEnd ="") %>%
  as.data.frame()
```

```
kable(f)
```

	lhs		rhs	support	confidence	lift	count
[1]		»>	Coffee	0.4784446	0.4784446	1.0000000	4528
[2]	Bread	»>	Coffee	0.0900254	0.2751049	0.5749985	852
[3]		»>	Tea	0.1426458	0.1426458	1.0000000	1350
[4]		»>	Cake	0.1038673	0.1038673	1.0000000	983
[5]	Bread	»>	Pastry	0.0291631	0.0891185	1.0348681	276
[6]		»>	Pastry	0.0861158	0.0861158	1.0000000	815
[7]	Bread	»>	Tea	0.0281065	0.0858896	0.6021177	266
[8]		»>	Sandwich	0.0718512	0.0718512	1.0000000	680
[9]	Bread	»>	Cake	0.0233516	0.0713594	0.6870246	221
[10]		»>	Medialuna	0.0618132	0.0618132	1.0000000	585

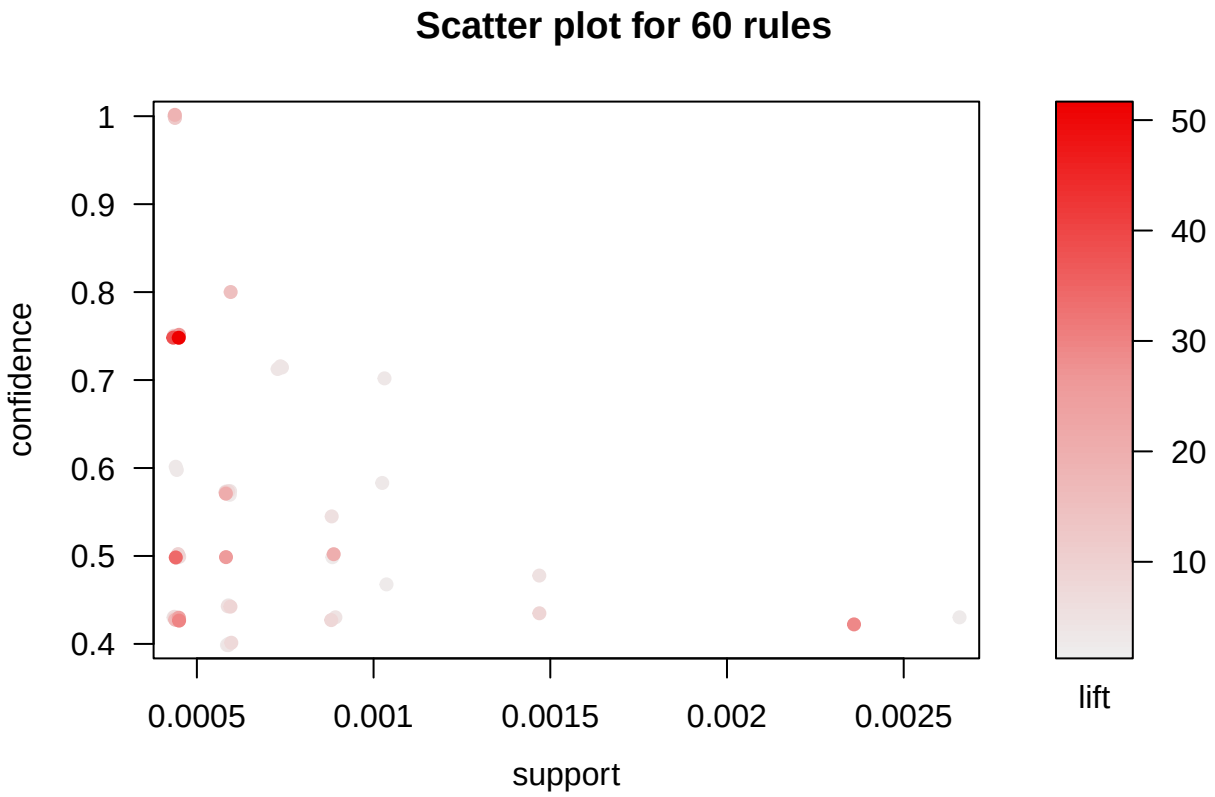
Visulalization

```
plot(teaFreqRules, method="matrix", measure="confidence")
```



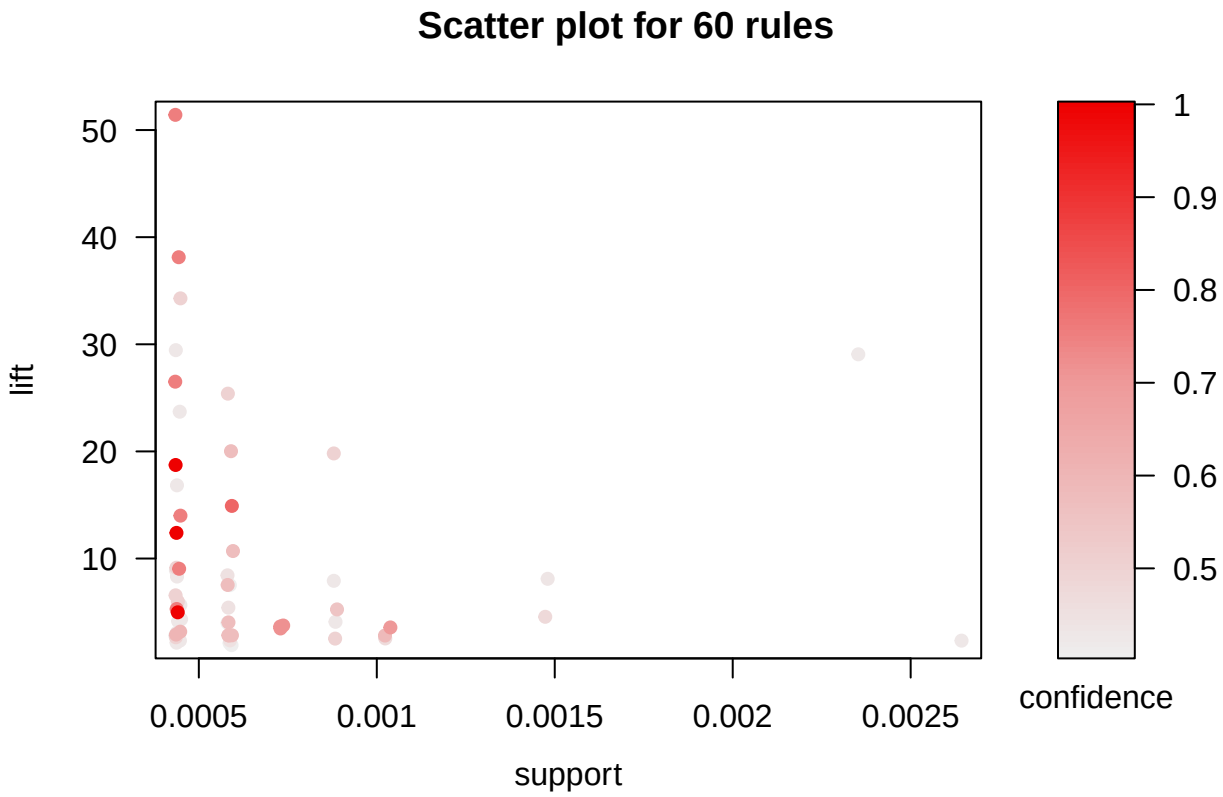

```
plot(teaFreqRules)
```

```
To reduce overplotting, jitter is added! Use jitter = 0 to prevent jitter.
```



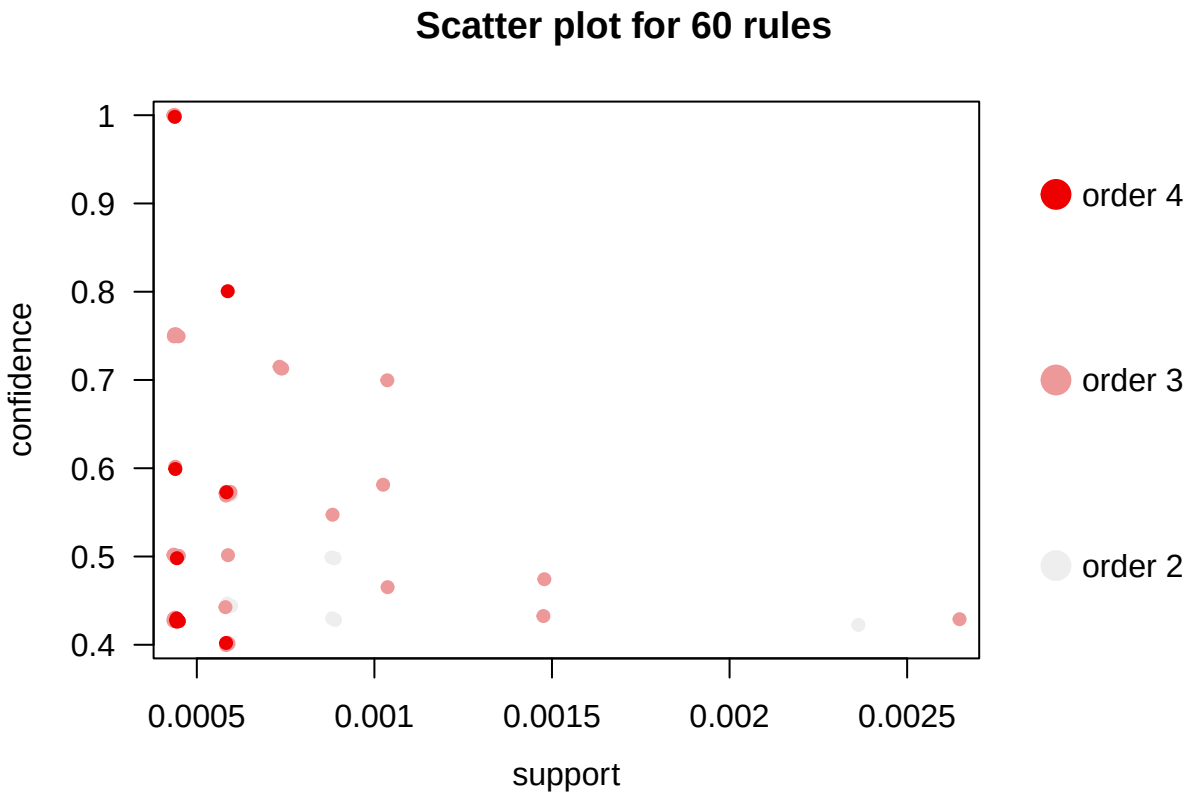

```
plot(teaFreqRules, measure=c("support","lift"), shading="confidence")
```

```
To reduce overplotting, jitter is added! Use jitter = 0 to prevent jitter.
```

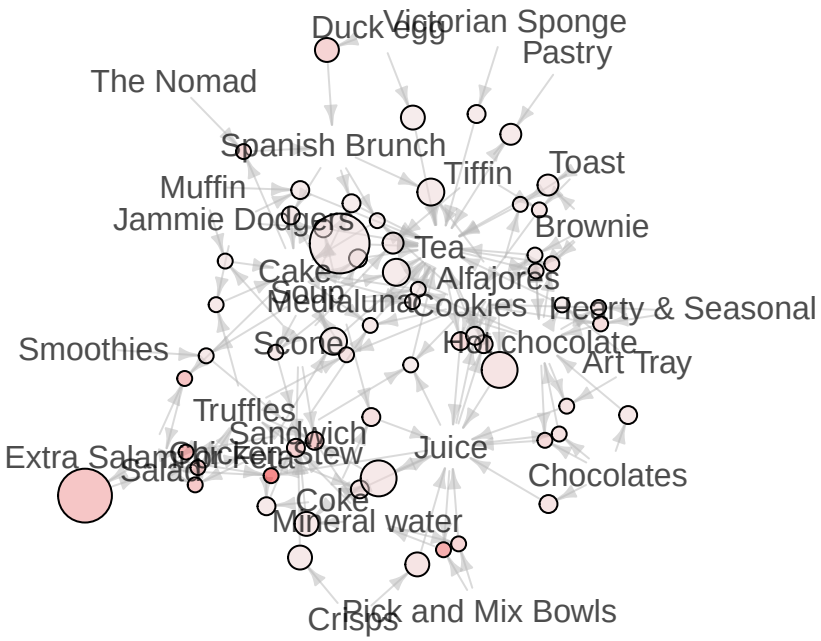



```
plot(teaFreqRules, shading="order")
```

```
To reduce overplotting, jitter is added! Use jitter = 0 to prevent jitter.
```




```
plot(teaFreqRules, method="graph")
```



Dissimilarity

most are dissimilar - 90%

```
coffeeDiss <- coffee[,itemFrequency(coffee)>0.05] %>%
  dissimilarity(which="items") %>%
  round(2) %>%
  as.matrix()
```

```
kable(coffeeDiss)
```

	Bread	Cake	Coffee	Cookies	Hot chocolate	Medialuna	Pastry	Sandwich	Tea
Bread	0.00	0.94	0.87	0.96	0.96	0.95	0.92	0.96	0.94
Cake	0.94	0.00	0.90	0.95	0.92	0.98	0.97	0.96	0.89
Coffee	0.87	0.90	0.00	0.94	0.94	0.93	0.91	0.93	0.91
Cookies	0.96	0.95	0.94	0.00	0.94	0.98	0.98	0.98	0.95
Hot chocolate	0.96	0.92	0.94	0.94	0.00	0.96	0.96	0.96	0.96
Medialuna	0.95	0.98	0.93	0.98	0.96	0.00	0.93	0.98	0.96
Pastry	0.92	0.97	0.91	0.98	0.96	0.93	0.00	0.99	0.96
Sandwich	0.96	0.96	0.93	0.98	0.96	0.98	0.99	0.00	0.93
Tea	0.94	0.89	0.91	0.95	0.96	0.96	0.96	0.93	0.00

```
coffeeDiss %>%
  melt() %>%
  ggplot(aes(X1, X2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = cBlue,  high = cYellow)
```