groceries association rules

November 25, 2020

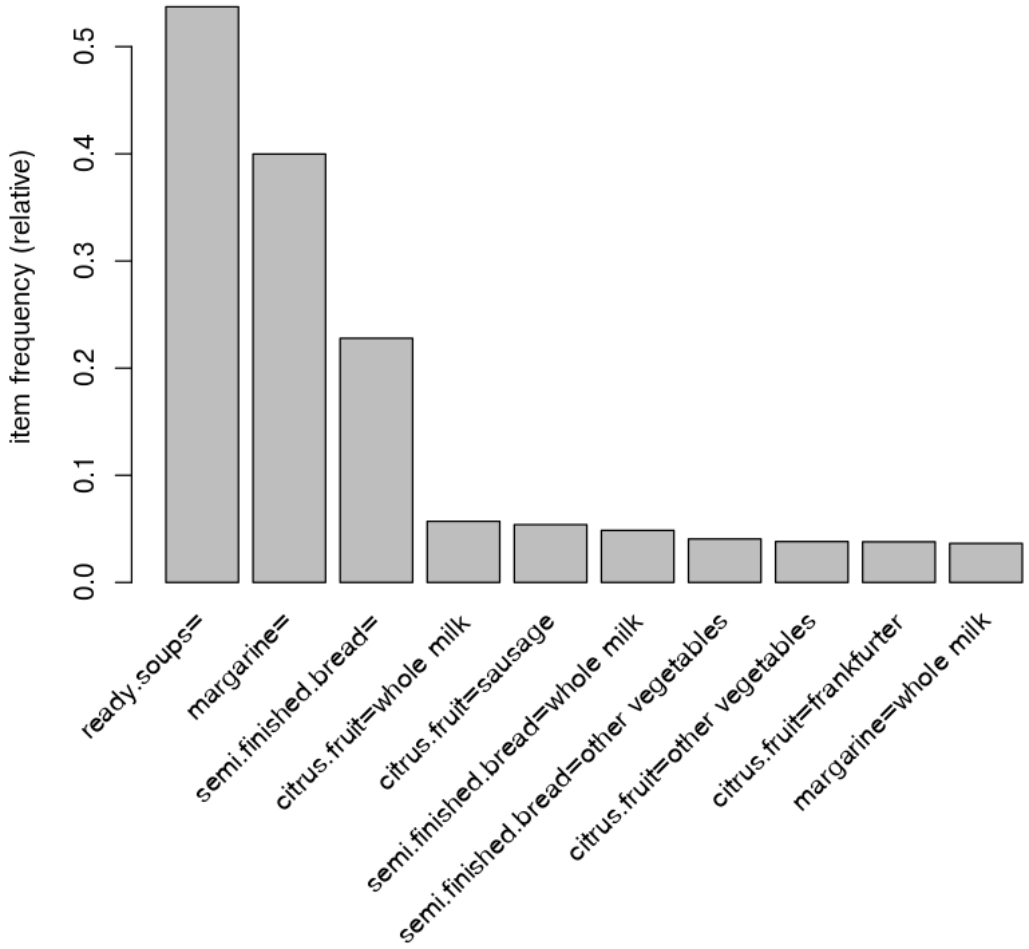
[2]:
```
library(stringr)
library(arules)
library(arulesViz)
library(mvinfluence)
library(MASS)
library(caret)
```

[3]:
```
groceries_data <- read.csv("groceries.csv")

str(groceries_data)
```

```
'data.frame':   15295 obs. of  4 variables:
 $ citrus.fruit       : chr  "tropical fruit" "whole milk" "pip fruit" "other
vegetables" …
 $ semi.finished.bread: chr  "yogurt" "" "yogurt" "whole milk" …
 $ margarine          : chr  "coffee" "" "cream cheese " "condensed milk" …
 $ ready.soups        : chr  "" "" "meat spreads" "long life bakery product" …
```

[4]:
```
groceries <- as(groceries_data, "transactions")

itemFrequencyPlot(groceries, topN=10)
```

```
Warning message:
"Column(s) 1, 2, 3, 4 not logical or factor. Applying default discretization
(see '? discretizeDF')."
```



[5]:
```
grocires_apriori <- apriori(groceries, parameter = list(supp=0.03, conf=0.4,␣
↪minlen=2, maxlen=4))
```

```
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen
        0.4    0.1    1 none FALSE            TRUE       5    0.03      2
 maxlen target  ext
      4  rules TRUE

Algorithmic control:
 filter tree heap memopt load sort verbose
```

```
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 458

set item appearances …[0 item(s)] done [0.00s].
set transactions …[655 item(s), 15295 transaction(s)] done [0.01s].
sorting and recoding items … [15 item(s)] done [0.00s].
creating transaction tree … done [0.00s].
checking subsets of size 1 2 3 done [0.00s].
writing … [10 rule(s)] done [0.00s].
creating S4 object  … done [0.00s].
```

[6]:
```
grocires_apriori
```

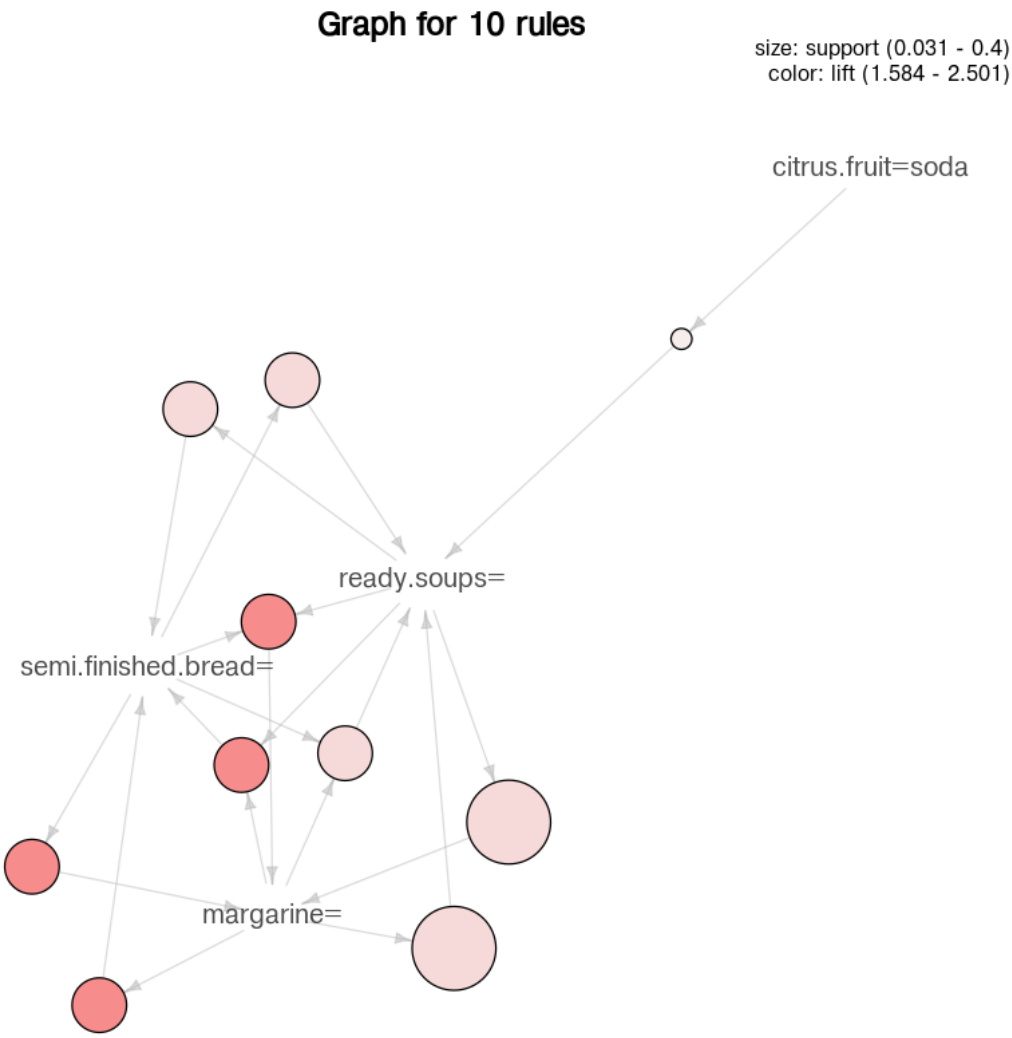
```
set of 10 rules
```

[7]:
```
inspect(grocires_apriori)
```

```
     lhs                                    rhs                     support
[1]  {citrus.fruit=soda}                 => {ready.soups=}          0.03059823
[2]  {semi.finished.bread=}              => {margarine=}            0.22785224
[3]  {margarine=}                        => {semi.finished.bread=}  0.22785224
[4]  {semi.finished.bread=}              => {ready.soups=}          0.22785224
[5]  {ready.soups=}                      => {semi.finished.bread=}  0.22785224
[6]  {margarine=}                        => {ready.soups=}          0.39980386
[7]  {ready.soups=}                      => {margarine=}            0.39980386
[8]  {semi.finished.bread=,margarine=}   => {ready.soups=}          0.22785224
[9]  {semi.finished.bread=,ready.soups=} => {margarine=}            0.22785224
[10] {margarine=,ready.soups=}           => {semi.finished.bread=}  0.22785224
     confidence coverage   lift     count
[1]  0.8509091  0.03595946 1.583869  468
[2]  1.0000000  0.22785224 2.501226 3485
[3]  0.5699101  0.39980386 2.501226 3485
[4]  1.0000000  0.22785224 1.861385 3485
[5]  0.4241207  0.53723439 1.861385 3485
[6]  1.0000000  0.39980386 1.861385 6115
[7]  0.7441889  0.53723439 1.861385 6115
[8]  1.0000000  0.22785224 1.861385 3485
[9]  1.0000000  0.22785224 2.501226 3485
[10] 0.5699101  0.39980386 2.501226 3485
```

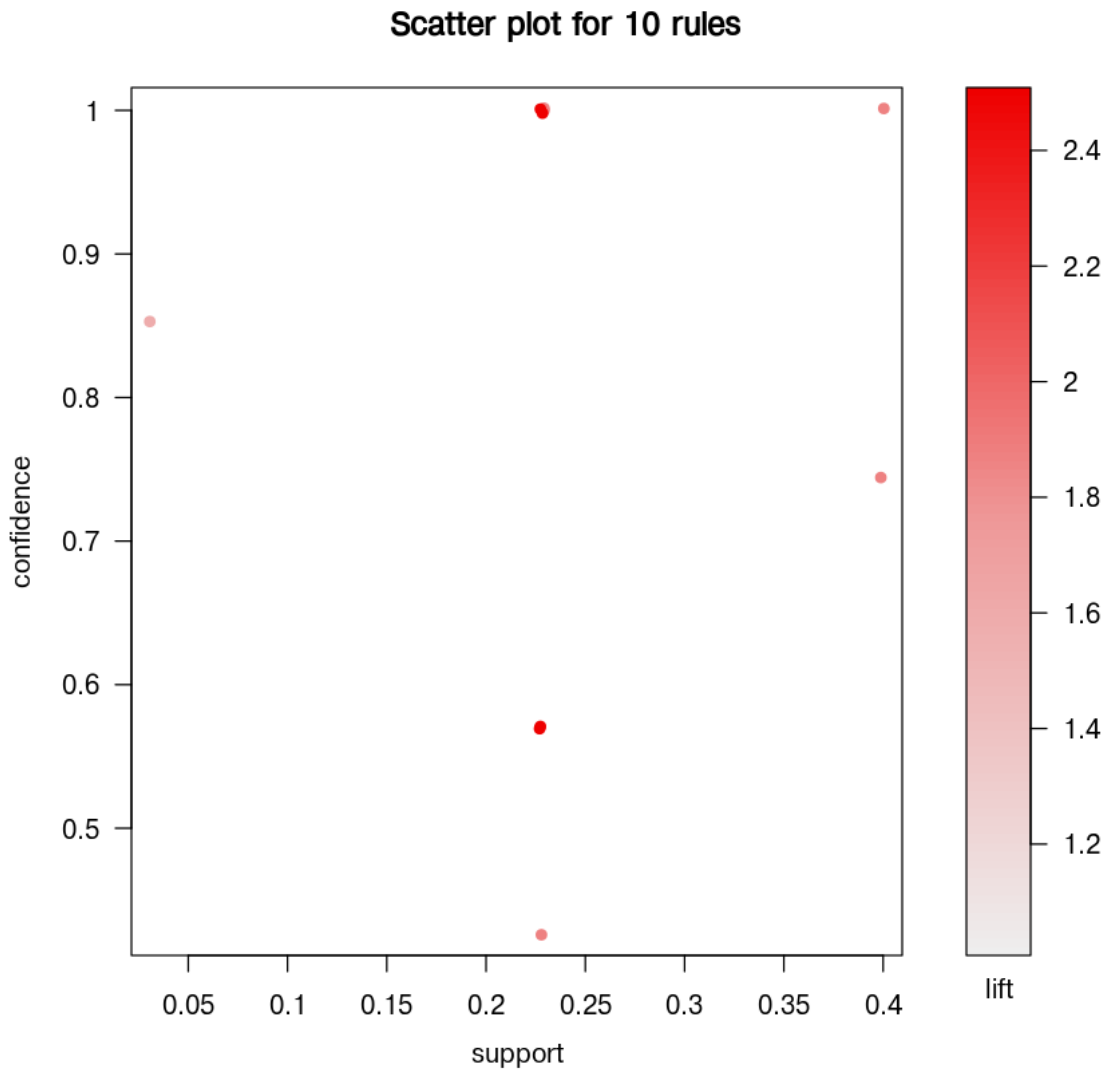
[8]:
```
write(grocires_apriori,
      file = "rules.csv",
      sep = ",",
      quote = TRUE,
      row.names = FALSE)
```

[9]:
```
plot(grocires_apriori, method="graph", control=list())
```

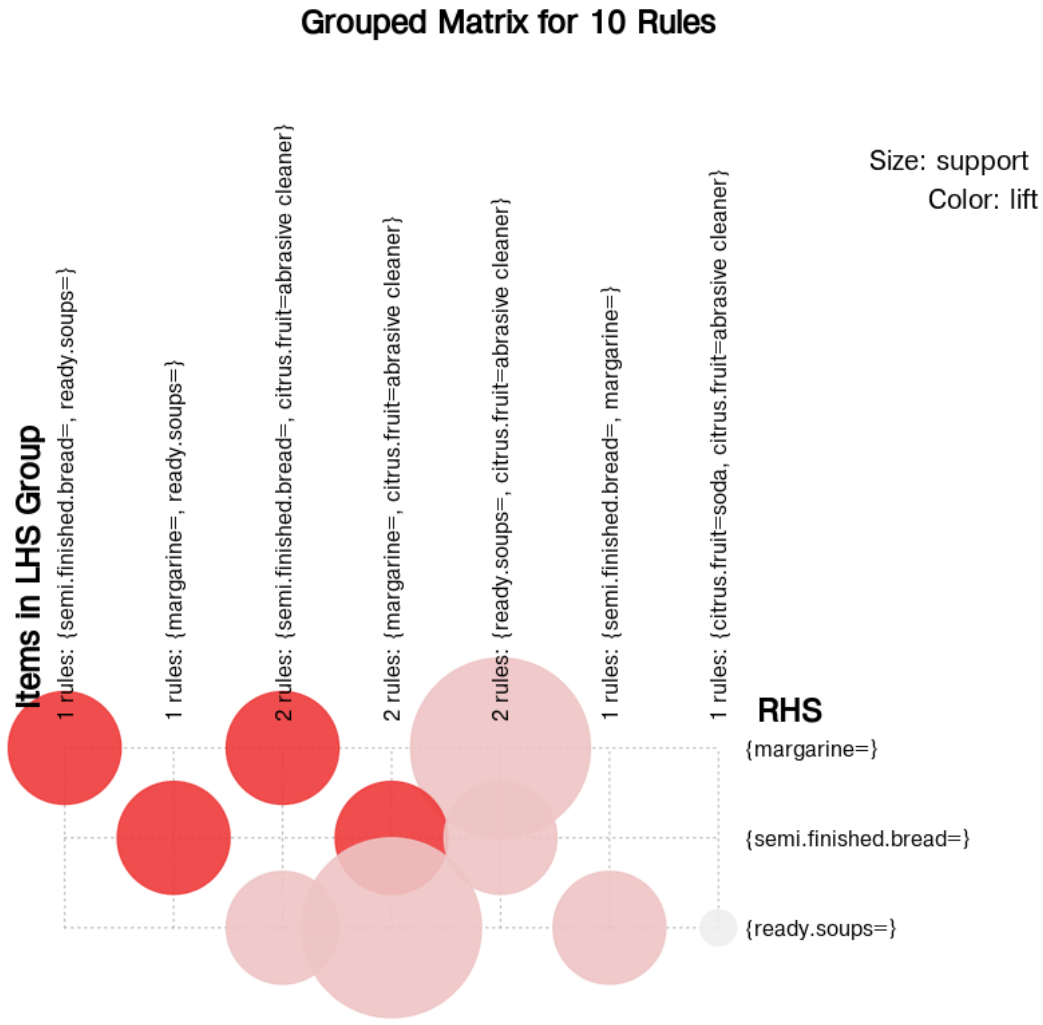


[10]:
```
plot(grocires_apriori)
```

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



[11]:
```
plot(grocires_apriori, method="grouped", control=list())
```


Grouped Matrix for 10 Rules
Size: support
Color: lift
Items in LHS Group
1 rules: {semi.finished.bread=, ready.soups=}
1 rules: {margarine=, ready.soups=}
2 rules: {semi.finished.bread=, citrus.fruit=abrasive cleaner}
2 rules: {margarine=, citrus.fruit=abrasive cleaner}
2 rules: {ready.soups=, citrus.fruit=abrasive cleaner}
1 rules: {semi.finished.bread=, margarine=}
1 rules: {citrus.fruit=soda, citrus.fruit=abrasive cleaner}
RHS
{margarine=}
{semi.finished.bread=}
{ready.soups=}