**Assignment 2 – Groceries**

Using Apriori algorithm: Below are the 3 steps to be accomplished for the assignment and necessary solution is being provided in the subsequent sections

**Problem Statement:**

Prepare rules for the all the data sets

1) Try different values of support and confidence. Observe the change in number of rules for different support, confidence values

2) Change the minimum length in apriori algorithm

3) Visualize the obtained rules using different plots

**Solution:**

Step 1: Try different values of support and confidence. Observe the change in number of rules for different support, confidence values

Groceries\_rules <- apriori(Groceries\_data,parameter = list(support = 0.05,confidence = 0.4,minlen=3))

|  |
| --- |
| > Groceries\_rules <- apriori(Groceries\_data,parameter = list(support = 0.05,confidence = 0.4,minlen=3))  Apriori  Parameter specification:  confidence minval smax arem aval originalSupport maxtime support minlen maxlen  0.4 0.1 1 none FALSE TRUE 5 0.05 3 10  target ext  rules FALSE  Algorithmic control:  filter tree heap memopt load sort verbose  0.1 TRUE TRUE FALSE TRUE 2 TRUE  Absolute minimum support count: 764  set item appearances ...[0 item(s)] done [0.00s].  set transactions ...[655 item(s), 15295 transaction(s)] done [0.02s].  sorting and recoding items ... [5 item(s)] done [0.00s].  creating transaction tree ... done [0.01s].  checking subsets of size 1 2 3 done [0.00s].  writing ... [3 rule(s)] done [0.00s].  creating S4 object ... done [0.00s]. |
|  |
| |  | | --- | | > | |

> inspect(Groceries\_rules)

lhs rhs support

[1] {semi.finished.bread=,margarine=} => {ready.soups=} 0.2278522

[2] {semi.finished.bread=,ready.soups=} => {margarine=} 0.2278522

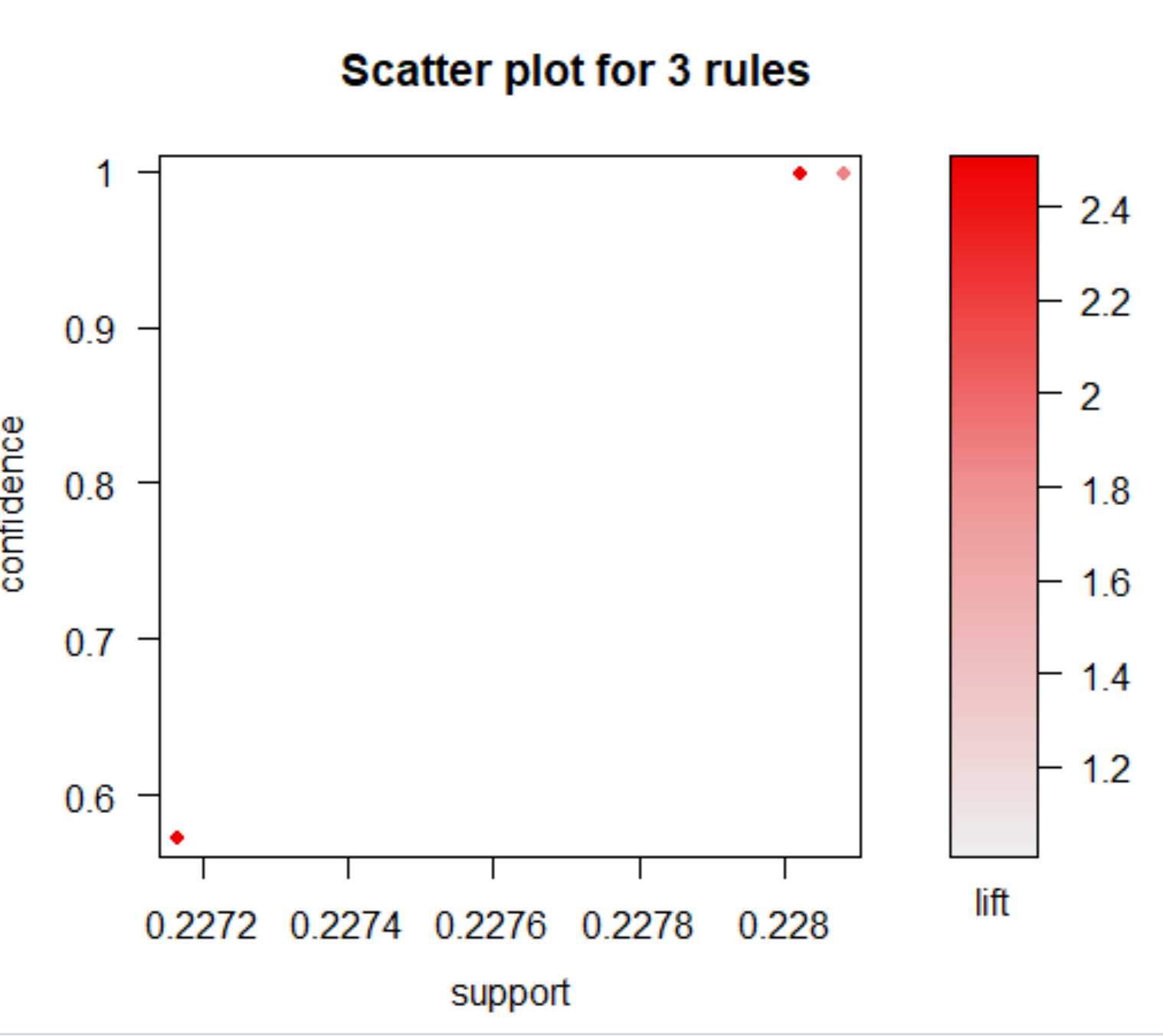
[3] {margarine=,ready.soups=} => {semi.finished.bread=} 0.2278522

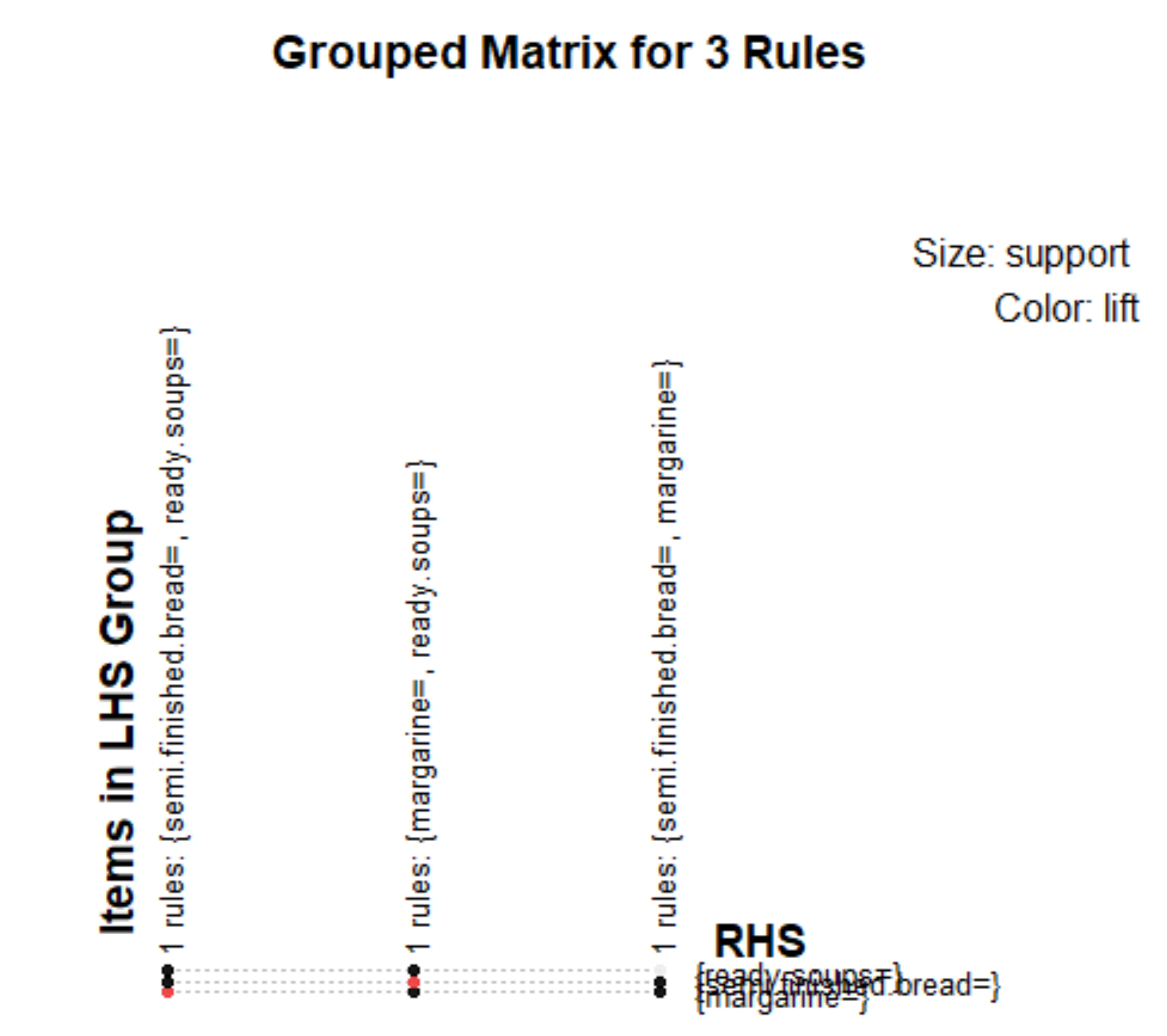
confidence lift count

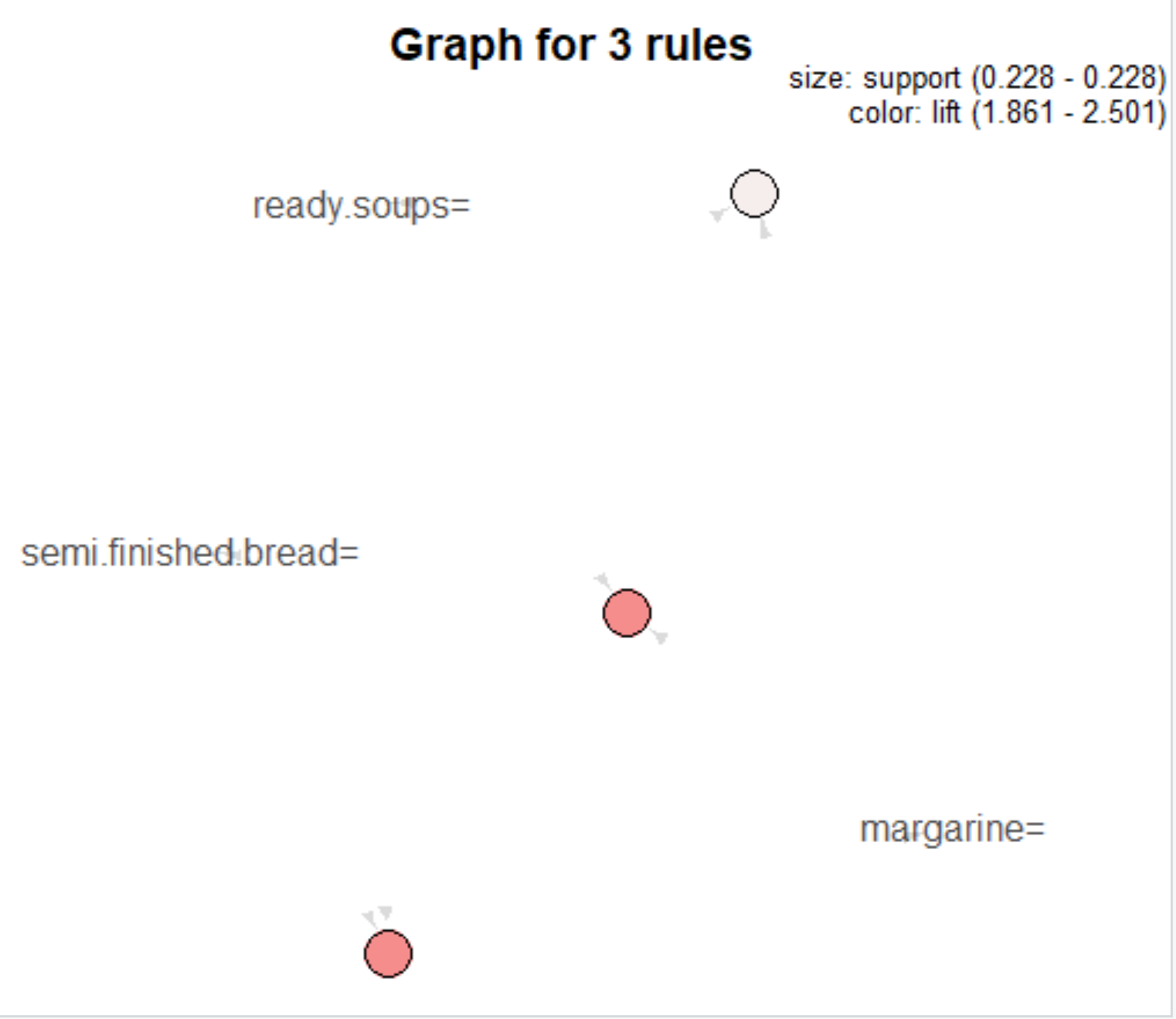
[1] 1.0000000 1.861385 3485

[2] 1.0000000 2.501226 3485

[3] 0.5699101 2.501226 3485







#2nd rule : Groceries\_rules <- apriori(Groceries\_data,parameter = list(support = 0.02,confidence = 0.5,minlen=2))

> Groceries\_rules <- apriori(Groceries\_data,parameter = list(support = 0.02,confidence = 0.5,minlen=2))

Apriori

Parameter specification:

confidence minval smax arem aval originalSupport maxtime support minlen maxlen

0.5 0.1 1 none FALSE TRUE 5 0.02 2 10

target ext

rules FALSE

Algorithmic control:

filter tree heap memopt load sort verbose

0.1 TRUE TRUE FALSE TRUE 2 TRUE

Absolute minimum support count: 305

set item appearances ...[0 item(s)] done [0.00s].

set transactions ...[655 item(s), 15295 transaction(s)] done [0.02s].

sorting and recoding items ... [31 item(s)] done [0.00s].

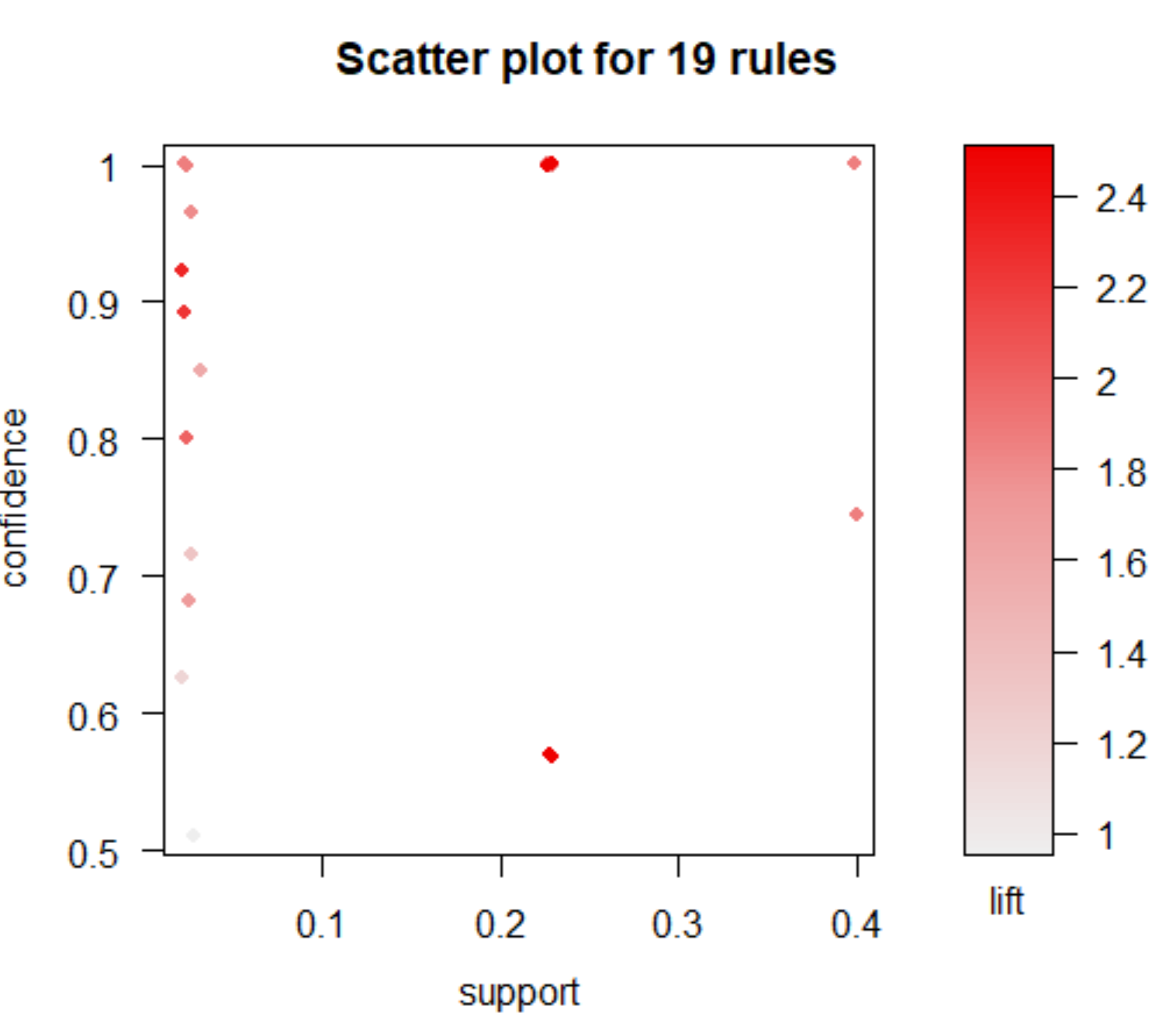
creating transaction tree ... done [0.01s].

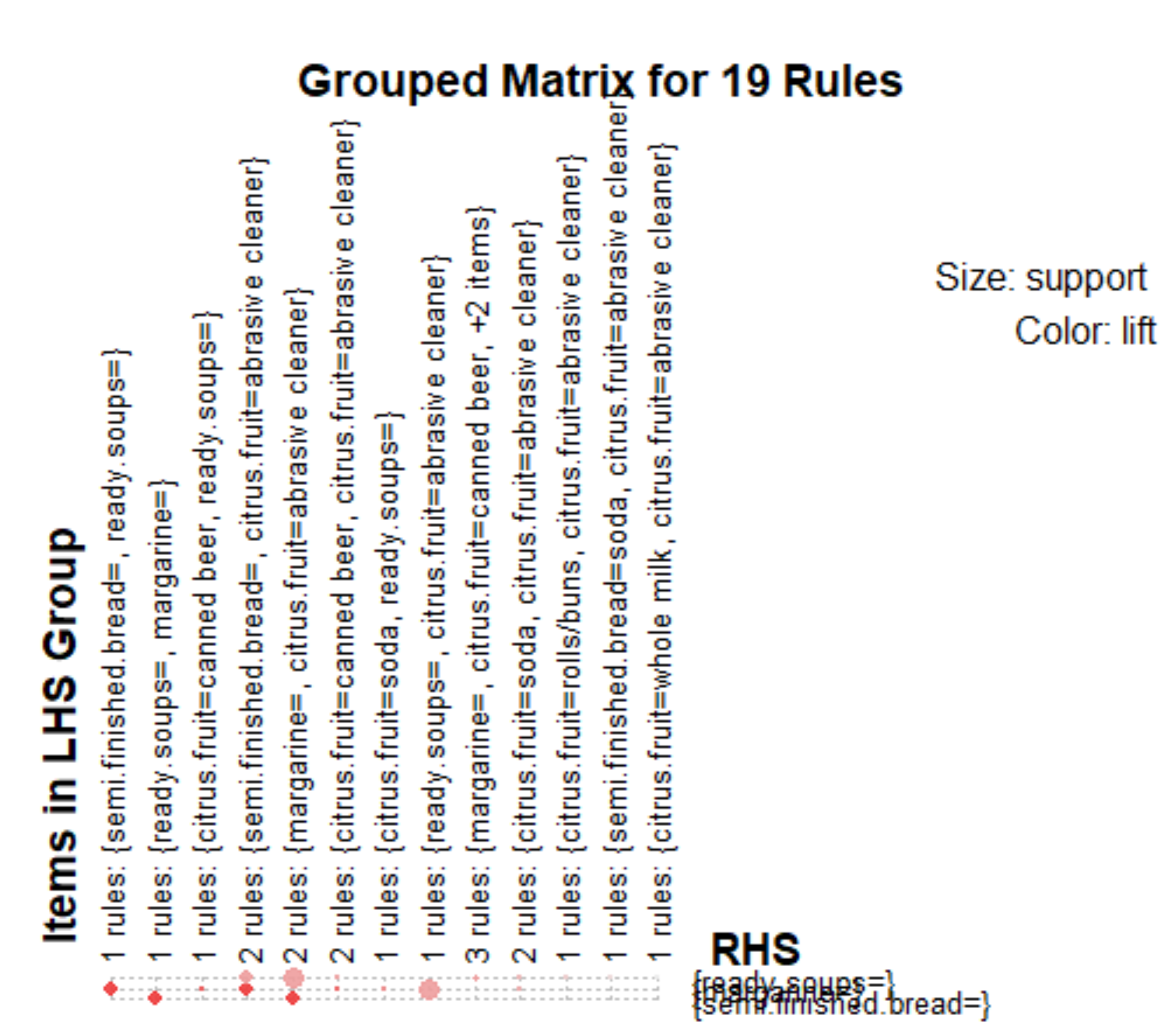
checking subsets of size 1 2 3 done [0.00s].

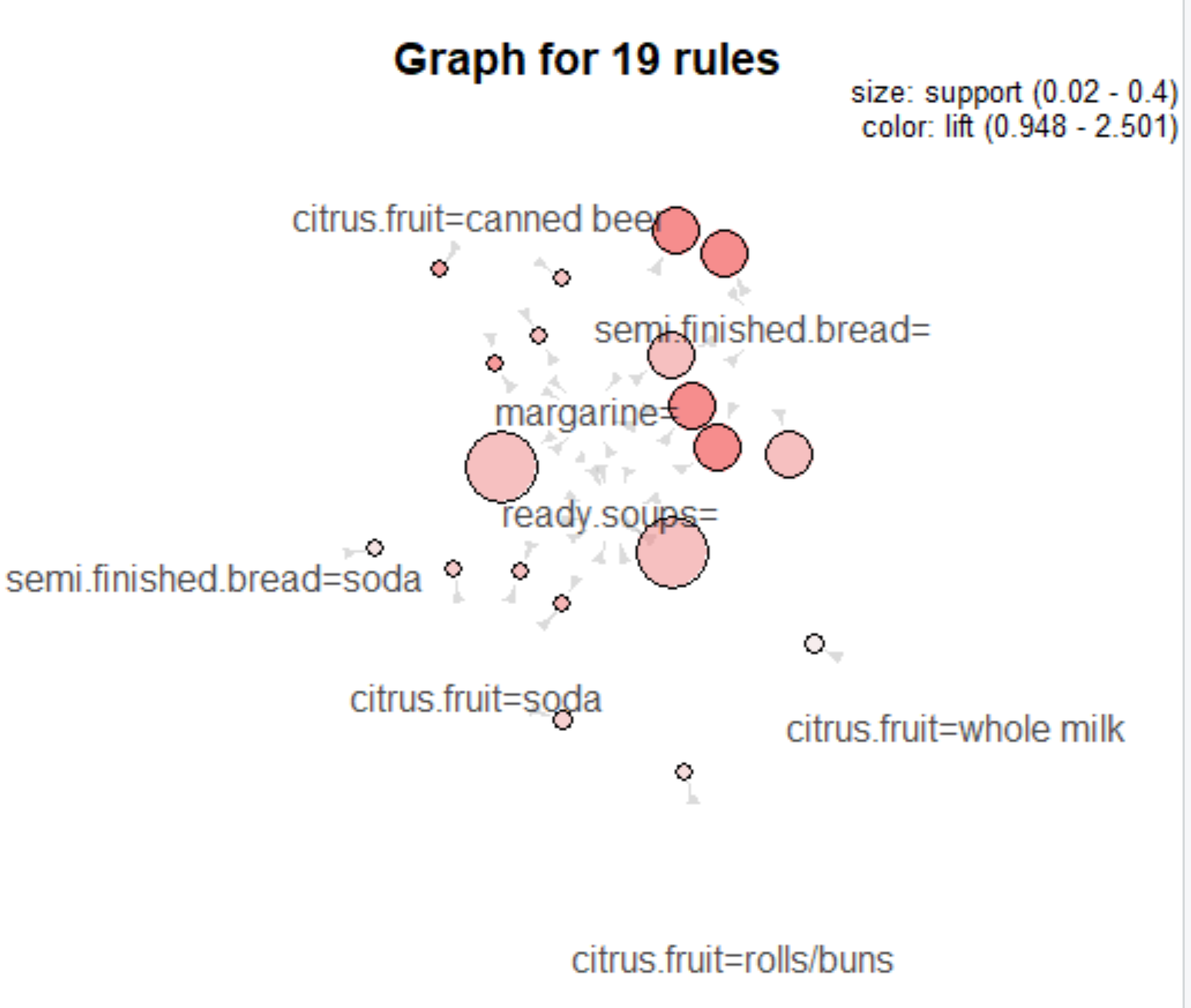
writing ... [19 rule(s)] done [0.00s].

creating S4 object ... done [0.00s].

|  |
| --- |
| > inspect(head(sort(Groceries\_rules, by = "lift")))  lhs rhs support  [1] {semi.finished.bread=} => {margarine=} 0.22785224  [2] {margarine=} => {semi.finished.bread=} 0.22785224  [3] {semi.finished.bread=,ready.soups=} => {margarine=} 0.22785224  [4] {margarine=,ready.soups=} => {semi.finished.bread=} 0.22785224  [5] {citrus.fruit=canned beer,ready.soups=} => {margarine=} 0.02262177  [6] {citrus.fruit=canned beer} => {margarine=} 0.02262177  confidence lift count  [1] 1.0000000 2.501226 3485  [2] 0.5699101 2.501226 3485  [3] 1.0000000 2.501226 3485  [4] 0.5699101 2.501226 3485  [5] 0.9226667 2.307798 346  [6] 0.8917526 2.230475 346 |
|  |
| |  | | --- | | > | |







#3rd rule: Groceries\_rules <- apriori(Groceries\_data,parameter = list(support = 0.04,confidence = 0.6,minlen=3))

> Groceries\_rules <- apriori(Groceries\_data,parameter = list(support = 0.04,confidence = 0.6,minlen=3))

Apriori

Parameter specification:

confidence minval smax arem aval originalSupport maxtime support minlen maxlen

0.6 0.1 1 none FALSE TRUE 5 0.04 3 10

target ext

rules FALSE

Algorithmic control:

filter tree heap memopt load sort verbose

0.1 TRUE TRUE FALSE TRUE 2 TRUE

Absolute minimum support count: 611

set item appearances ...[0 item(s)] done [0.00s].

set transactions ...[655 item(s), 15295 transaction(s)] done [0.02s].

sorting and recoding items ... [7 item(s)] done [0.00s].

creating transaction tree ... done [0.01s].

checking subsets of size 1 2 3 done [0.00s].

writing ... [2 rule(s)] done [0.00s].

creating S4 object ... done [0.00s].

> inspect(Groceries\_rules)

lhs rhs support confidence

[1] {semi.finished.bread=,margarine=} => {ready.soups=} 0.2278522 1

[2] {semi.finished.bread=,ready.soups=} => {margarine=} 0.2278522 1

lift count

[1] 1.861385 3485

[2] 2.501226 3485

> inspect(head(sort(Groceries\_rules, by = "lift")))

lhs rhs support confidence

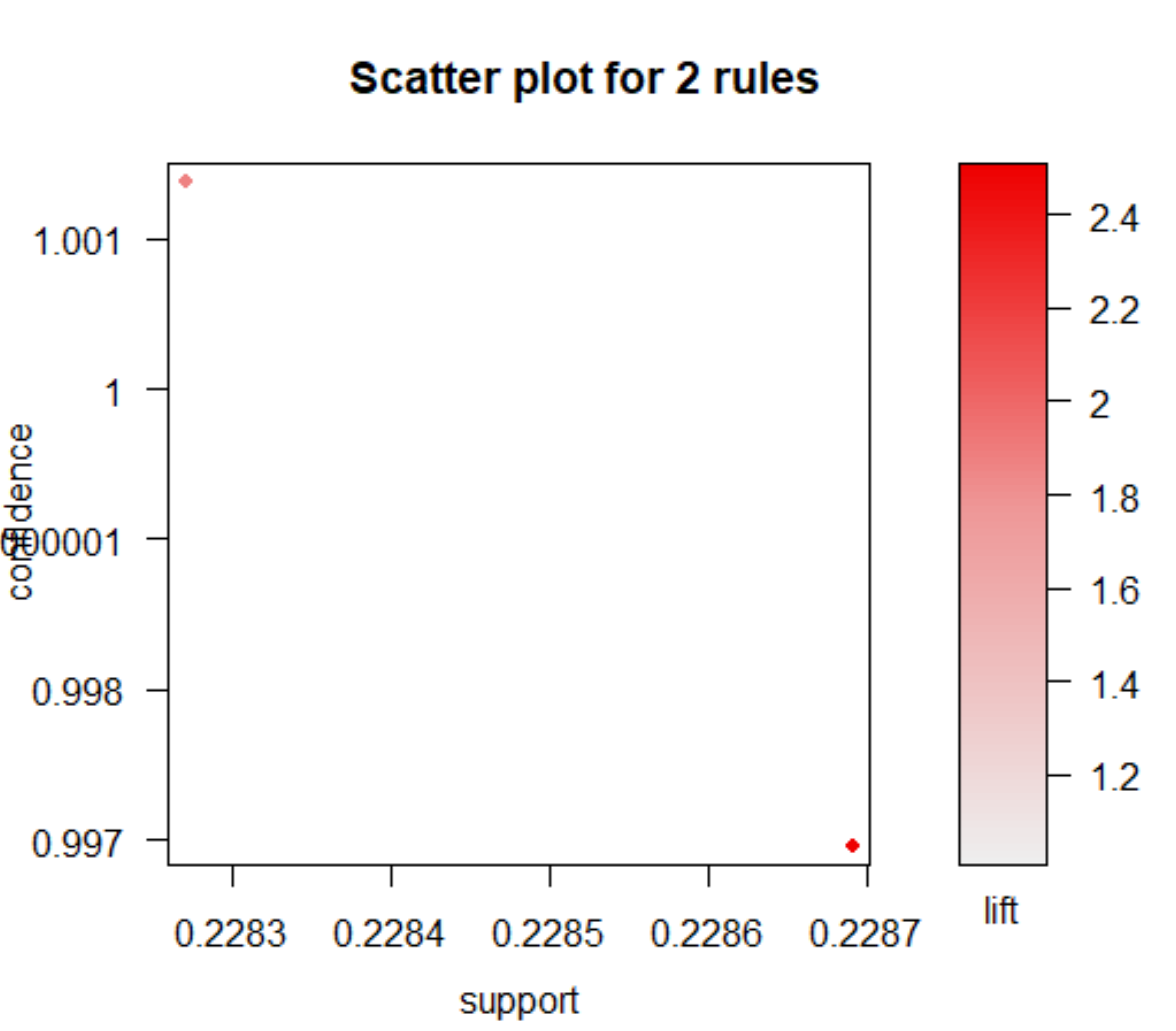
[1] {semi.finished.bread=,ready.soups=} => {margarine=} 0.2278522 1

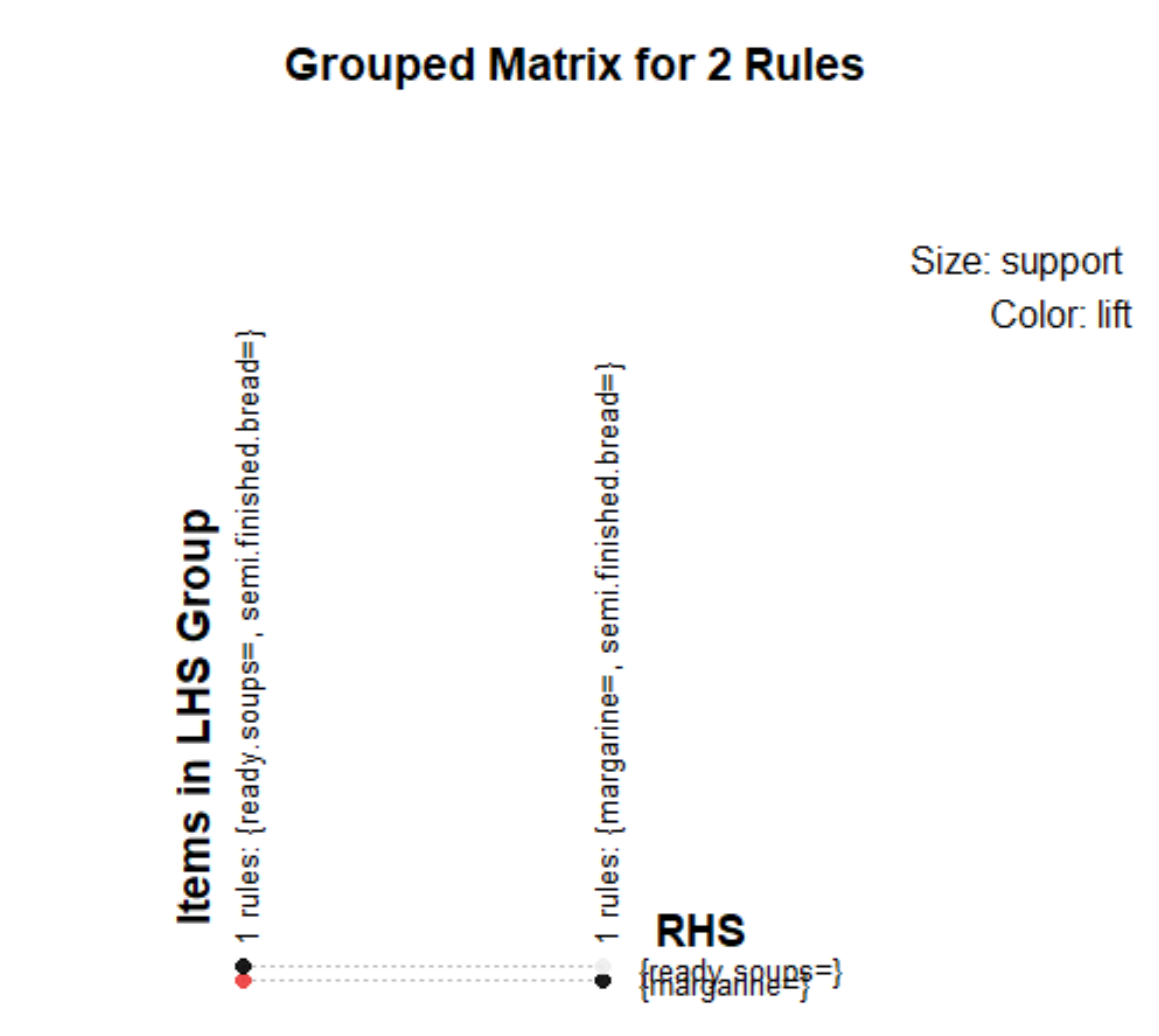
[2] {semi.finished.bread=,margarine=} => {ready.soups=} 0.2278522 1

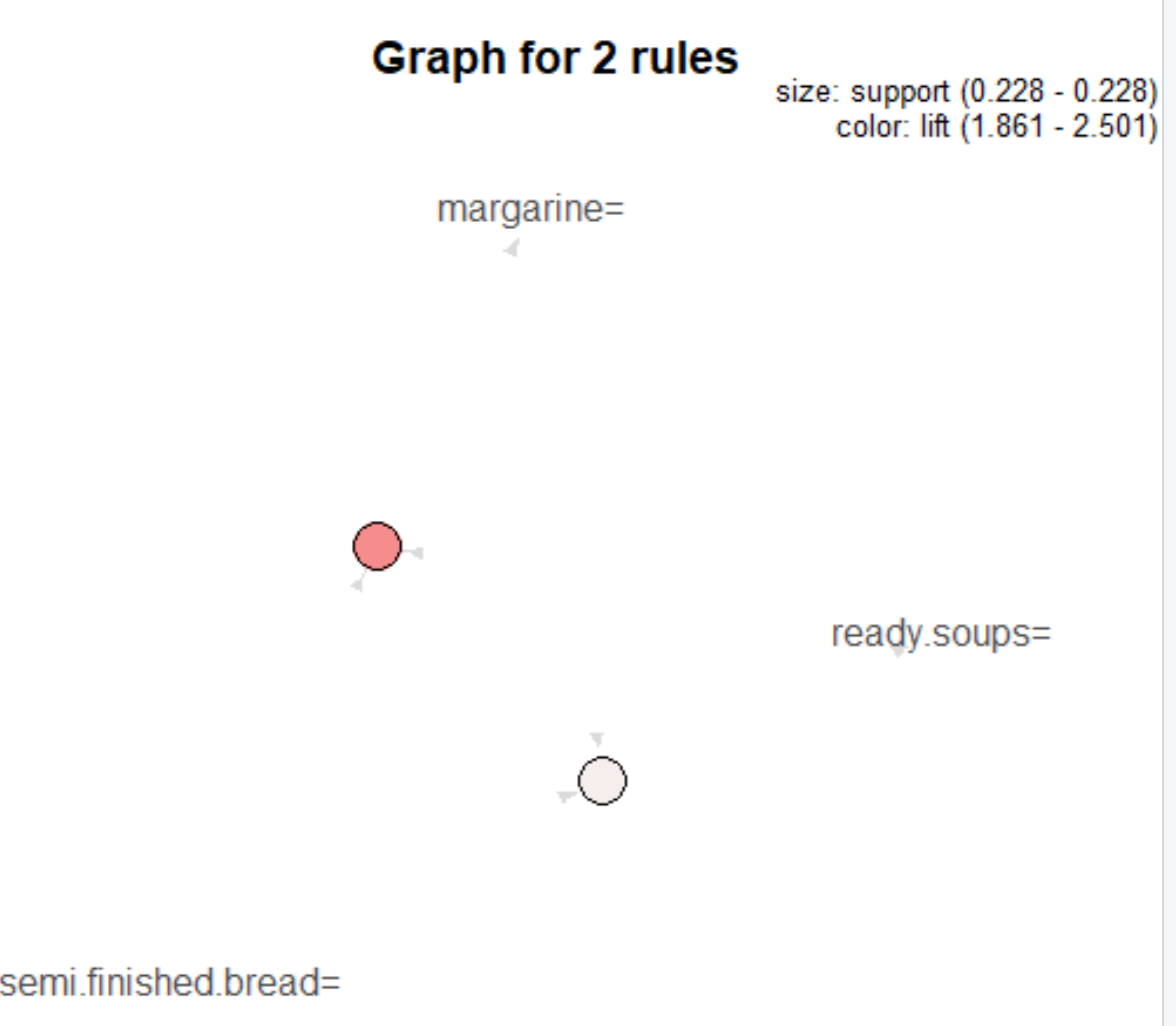
lift count

[1] 2.501226 3485

[2] 1.861385 3485







#4th rule: Groceries\_rules <- apriori(Groceries\_data,parameter = list(support = 0.03,confidence = 0.5,minlen=3))

|  |
| --- |
| > Groceries\_rules <- apriori(Groceries\_data,parameter = list(support = 0.03,confidence = 0.5,minlen=3))  Apriori  Parameter specification:  confidence minval smax arem aval originalSupport maxtime support minlen maxlen  0.5 0.1 1 none FALSE TRUE 5 0.03 3 10  target ext  rules FALSE  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 [0.01s].  checking subsets of size 1 2 3 done [0.00s].  writing ... [3 rule(s)] done [0.00s].  creating S4 object ... done [0.00s]. |
|  |
| |  | | --- | | > | |

> inspect(head(sort(Groceries\_rules, by = "lift")))

lhs rhs support

[1] {semi.finished.bread=,ready.soups=} => {margarine=} 0.2278522

[2] {margarine=,ready.soups=} => {semi.finished.bread=} 0.2278522

[3] {semi.finished.bread=,margarine=} => {ready.soups=} 0.2278522

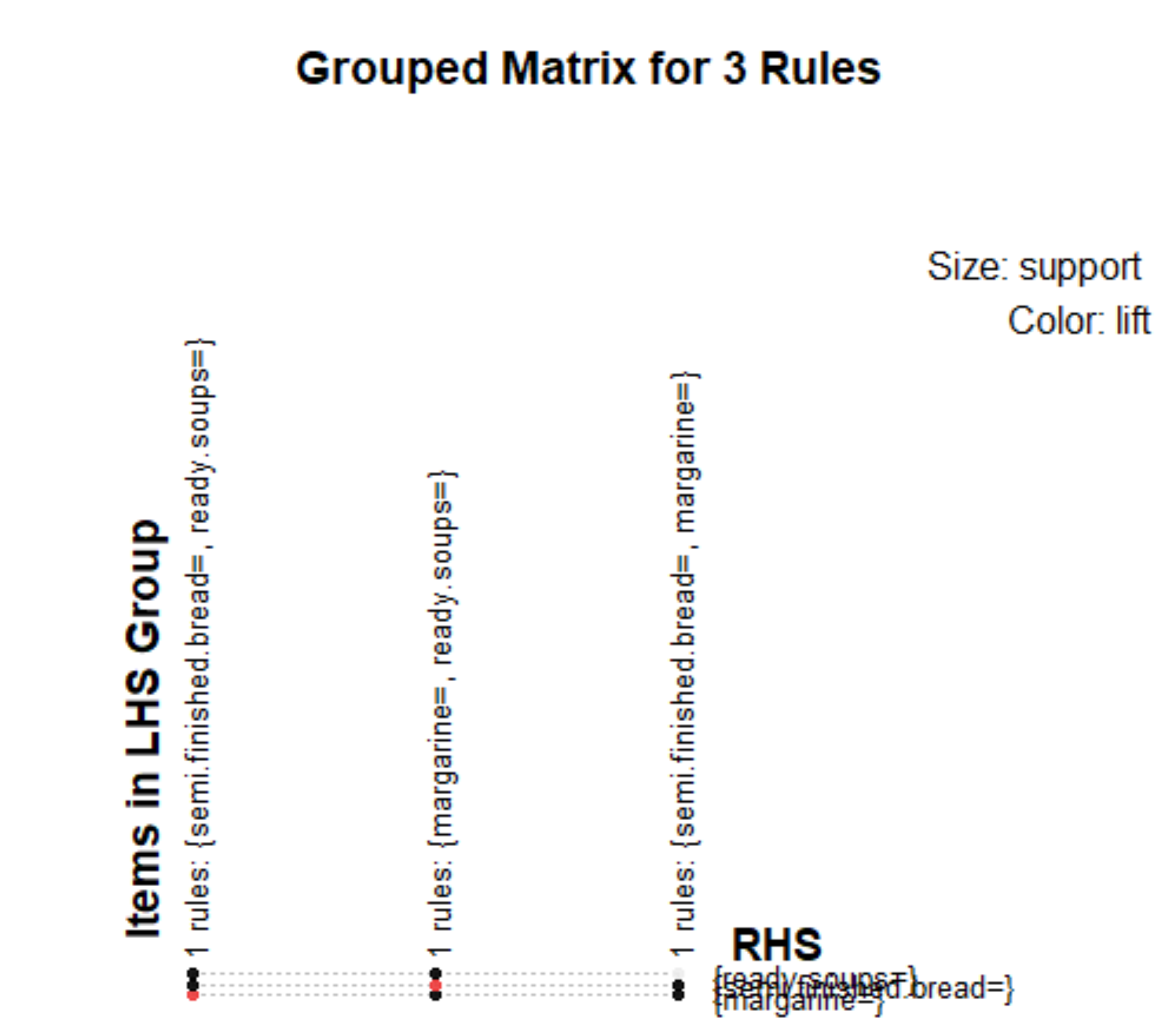
confidence lift count

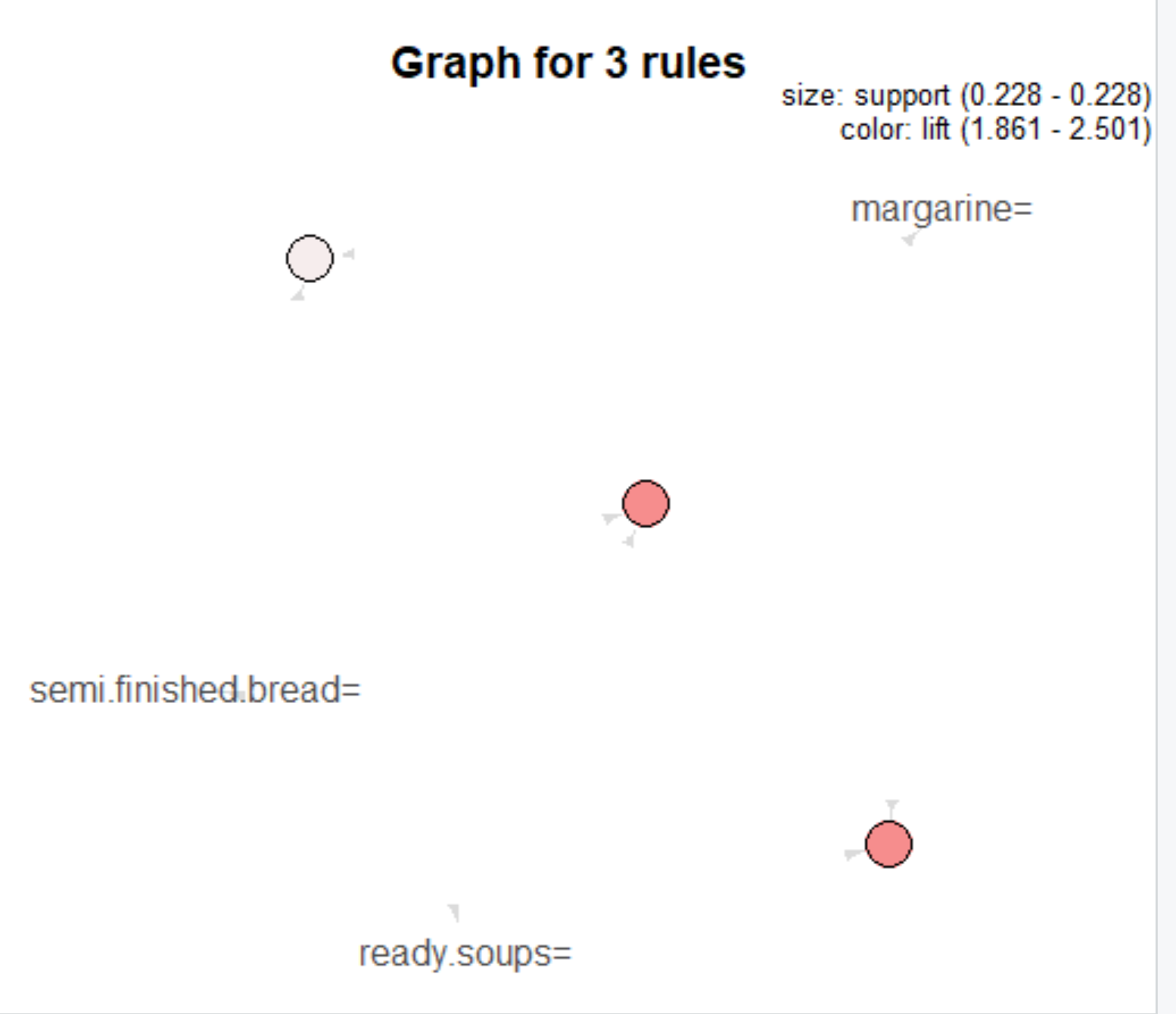
[1] 1.0000000 2.501226 3485

[2] 0.5699101 2.501226 3485

[3] 1.0000000 1.861385 3485







#5th rule : Groceries\_rules <- apriori(Groceries\_data,parameter = list(support = 0.01,confidence = 0.3,minlen=3))

> Groceries\_rules <- apriori(Groceries\_data,parameter = list(support = 0.01,confidence = 0.3,minlen=3))

Apriori

Parameter specification:

confidence minval smax arem aval originalSupport maxtime support minlen maxlen

0.3 0.1 1 none FALSE TRUE 5 0.01 3 10

target ext

rules FALSE

Algorithmic control:

filter tree heap memopt load sort verbose

0.1 TRUE TRUE FALSE TRUE 2 TRUE

Absolute minimum support count: 152

set item appearances ...[0 item(s)] done [0.00s].

set transactions ...[655 item(s), 15295 transaction(s)] done [0.02s].

sorting and recoding items ... [75 item(s)] done [0.00s].

creating transaction tree ... done [0.01s].

checking subsets of size 1 2 3 4 done [0.00s].

writing ... [60 rule(s)] done [0.00s].

creating S4 object ... done [0.00s].

> inspect(head(sort(Groceries\_rules, by = "lift")))

lhs rhs support confidence lift count

[1] {citrus.fruit=shopping bags,

margarine=} => {semi.finished.bread=} 0.01765283 1.0000000 4.388809 270

[2] {citrus.fruit=shopping bags,

ready.soups=} => {semi.finished.bread=} 0.01765283 1.0000000 4.388809 270

[3] {citrus.fruit=shopping bags,

margarine=,

ready.soups=} => {semi.finished.bread=} 0.01765283 1.0000000 4.388809 270

[4] {citrus.fruit=newspapers,

margarine=} => {semi.finished.bread=} 0.01203008 0.8932039 3.920101 184

[5] {citrus.fruit=newspapers,

ready.soups=} => {semi.finished.bread=} 0.01203008 0.8932039 3.920101 184

[6] {citrus.fruit=newspapers,

margarine=,

ready.soups=} => {semi.finished.bread=} 0.01203008 0.8932039 3.920101 184

