

**Marketing Analytics**  
**Professor Swagato Chatterjee**  
**Vinod Gupta School of Management**  
**Indian Institute of Technology, Kharagpur**  
**Lecture 44**  
**RFM and Market Basket Analysis (Contd.)**

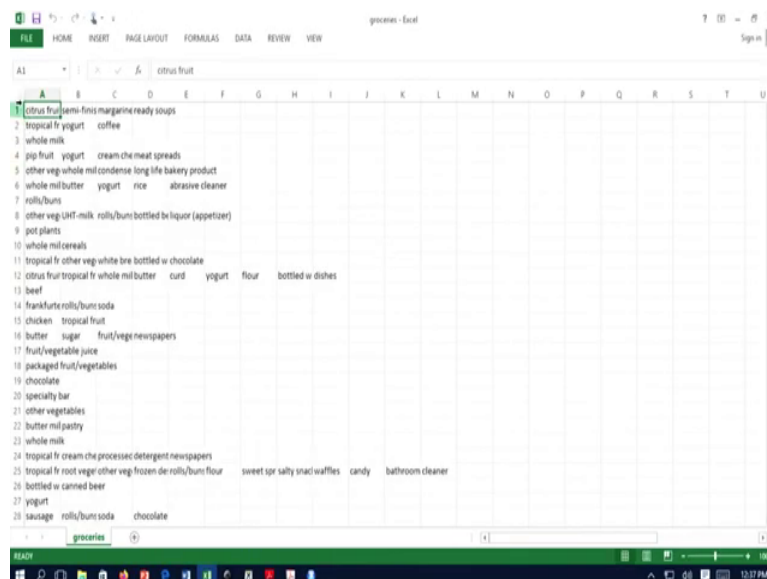
Hello everybody, welcome to marketing analytics class, we are in week 8, session 5 and in this particular session we will discuss about, how to do a market basket analysis in a hands on way?

(Refer Slide Time: 00:27)



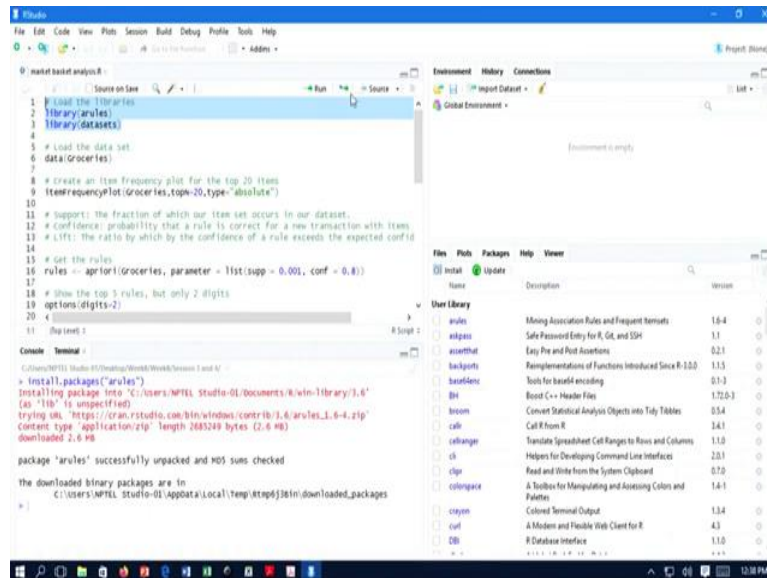
So, if you go to the files section of week 8, you will find a file called Market Basket analysis dot r and you will find also our grocery stores data.

(Refer Slide Time: 00:39)



So, if I just opened this data, these each of these particular rows is a transaction, so if I just. So, you see that like whole milk, butter, yoghurt, rice and abrasive cleaner has been bought together. And let say pip fruit, yogurt, cream cheese and meat spades have been bought together. So, this is a publicly available data we have taken it and then we will be using that use in Market Basket Analysis.

(Refer Slide Time: 01:02)



```
market basket analysis.R
1 # Load the libraries
2 library(arules)
3 library(datasets)
4
5 # Load the data set
6 data(Groceries)
7
8 # Create an Item Frequency plot for the top 20 items
9 itemFrequencyPlot(Groceries,top=20,type="absolute")
10
11 # Support: the fraction of which our item set occurs in our dataset.
12 # Confidence: probability that a rule is correct for a new transaction with items
13 # Lift: the ratio by which the confidence of a rule exceeds the expected confid
14
15 # Get the rules
16 rules = apriori(Groceries, parameter = list(supp = 0.001, conf = 0.8))
17
18 # Show the top 5 rules, but only 2 digits
19 options(digits=2)
20
21 # Items 1
```

Console Terminal

```
C:\Users\NPTEL> RStudio\bin\Windows\bin\Shell\Terminal 1 and 4
> install.packages("arules")
Installing package into 'C:\Users\NPTEL Studio-01\Documents\R\win-library\3.6'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/arules_1.6-4.zip'
Content type 'application/zip' length 2683249 bytes (2.6 MB)
downloaded 2.6 MB

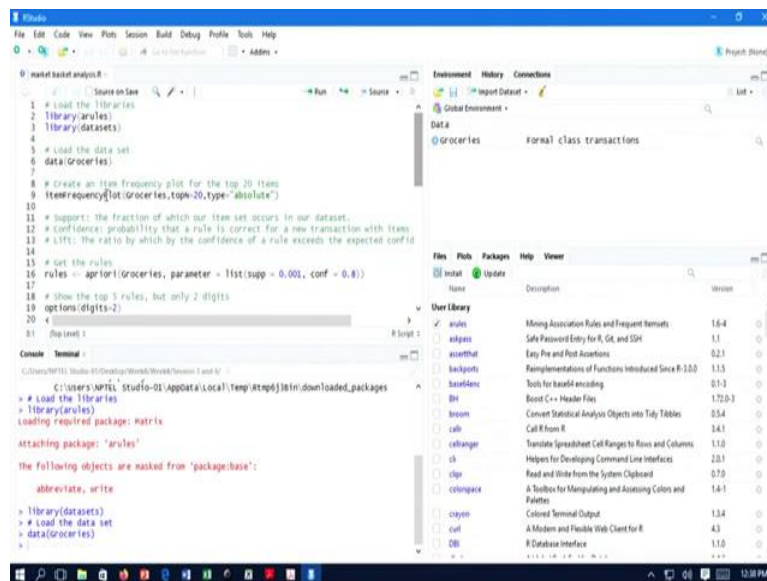
package 'arules' successfully unpacked and MD5 sums checked
The downloaded binary packages are in
C:\Users\NPTEL Studio-01\AppData\Local\Temp\itxepj36in\downloaded_packages
```

Name	Description	Version
arules	Mining Association Rules and Frequent Itemsets	1.6-4
askpass	Safe Password Entry for R, Git, and SSH	1.1
assertthat	Easy Pre and Post Assertions	0.2.1
backports	Implementations of Functions Introduced Since R 3.0.0	1.1.5
base64enc	Tools for base64 encoding	0.1-3
bit	Bitmap C++ Header Files	1.1-2.3
bsom	Convert Statistical Analysis Objects into Tidy Tables	0.5.4
call	Call R from R	14.1
colorspace	Translate Spreadsheet Cell Ranges to Rows and Columns	1.1.0
cli	Helpers for Developing Command Line Interfaces	2.0.1
clipr	Read and Write from the System Clipboard	0.7.0
colorspace	A Toolbox for Manipulating and Assessing Colors and Palettes	14-1
crayon	Colored Terminal Output	1.3.4
curl	A Modern and Flexible Web Client for R	4.3
DBI	R Database Interface	1.1.0

So, for that there are two libraries one is a-rules and one is data sets that I will use, so, if I do not have as usual I have to install them. So, a-rules stands for association rules. So, I will install them and then I will also install the data sets library which will be use. So, a-rules has been installed easy and data sets library.

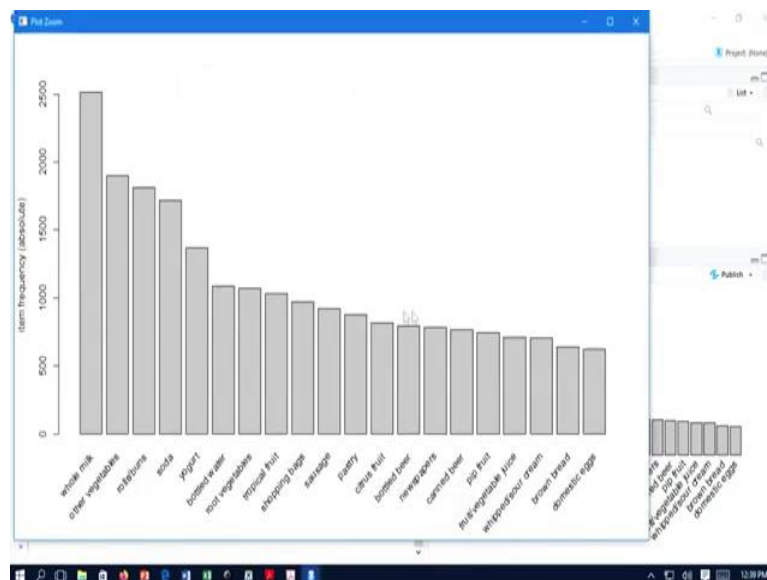
So, data sets is not there I think right now for the new one data sets is not there, so let us see whether we can do it without that. So, a-rules library I am calling or data sets library might already we installed here it is there in a base package, so a-rules has been installed.

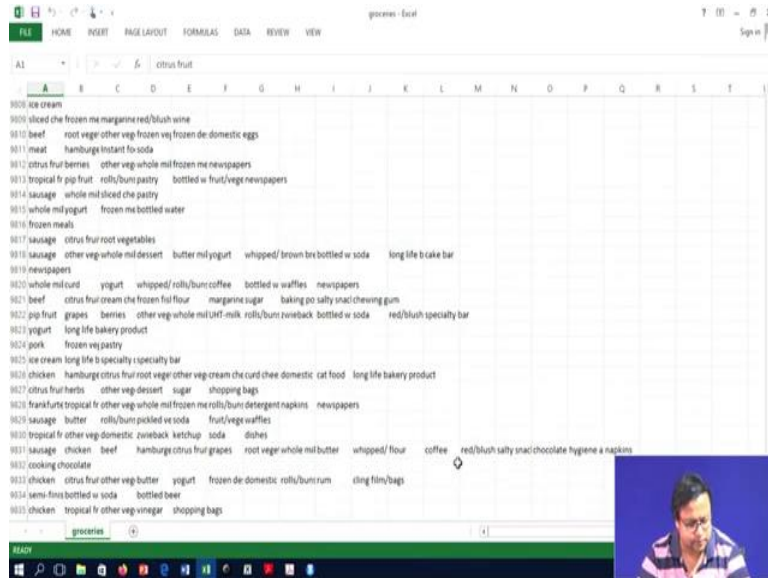
(Refer Slide Time: 02:00)



Now, the data is data groceries, it is a inbuilt data, so the same one that I have written here I have there is inbuilt and it is written here all the data classes and etc. So, I have used that. Now, in I will find out the first thing is to find out the one items which items are more prominent. So, item frequency plot is something that I am creating with top 20.

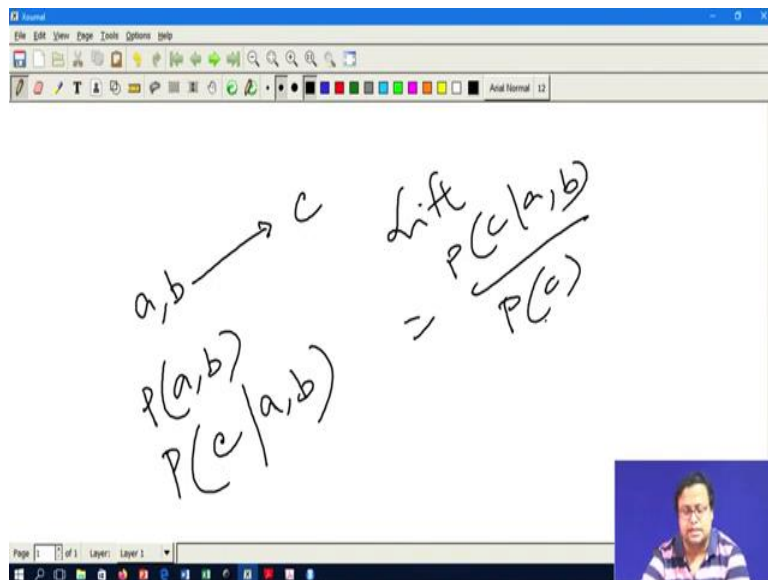
(Refer Slide Time: 02:27)





So, this are my top 20 products, you can see that whole milk is the most common product, then other vegetables, then comes rolls and buns, then comes soda, yoghurt, bottled water and etc. So, I think I have around 10,000 observations and out of them 2500 cases whole milk is occurring and the other ones is occurring like that. So, that is something which is important to understand. Just one minute, let me check how many rows I have, yeah around 10,000 it is there.

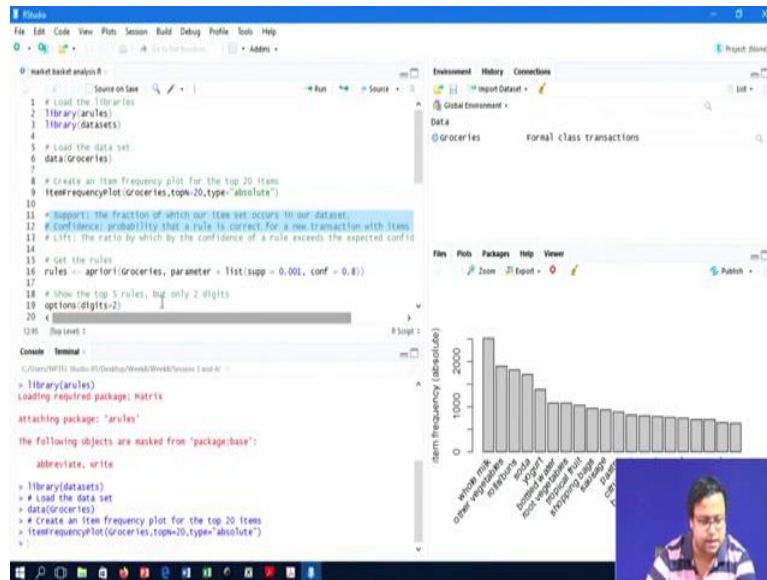
(Refer Slide Time: 03:20)



So, now there are three definitions support and confidence we are already known. What is support? Support is basically the fraction of which our item set occurs, means? If I say that item a and b, given that if I if I asked you that what is the probability, what is the case that a and b occurs, c will also occur? So, if you buy a and b, c will also occur. So,

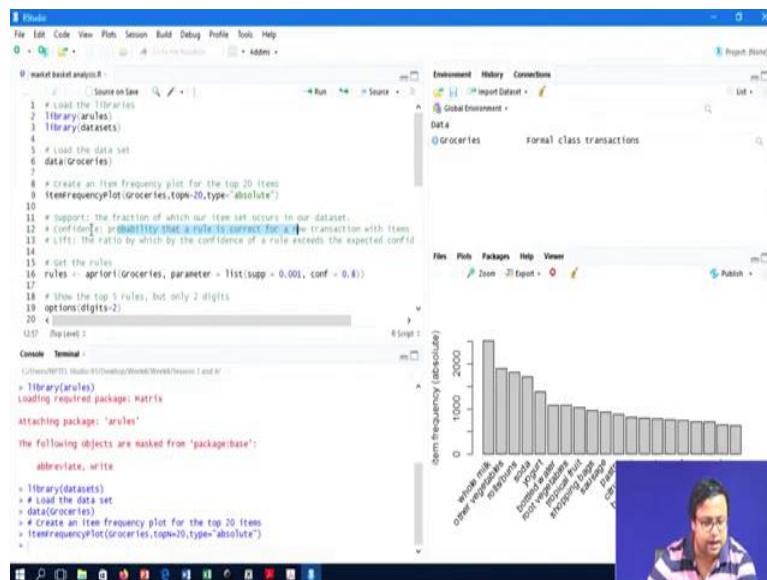
support is basically probability of a comma b and confidence is basically probability of c given a comma b, so that is basic thing.

(Refer Slide Time: 03:38)



So, out of this case, how when the fraction of which our items set occurs in our data set? That is support. Confidence is probability that the rule is correct, probability that this rule is correct. This rule correct means? Probability of this thing, so that is called your confidence.

(Refer Slide Time: 03:52)



And lift, the ratio by which the by which the confidence of a rule exceeds the expected confidence.

(Refer Slide Time: 04:03)

The slide shows handwritten mathematical formulas. On the left, there is a diagram where 'a, b' has an arrow pointing to 'c'. Below this, the formulas  $P(a, b)$  and  $P(c | a, b)$  are written. To the right, the formula for Lift is given as 
$$\text{Lift} = \frac{P(c | a, b)}{P(c)}$$

So, what is the Lift is basically in other words, probability of c given a comma b, by probability of c, so that is lift. So, this is something these three things is important for marketing. The higher the lift, the better is your rule, the more stronger, more applicable, more actionable is your rule. Because if our rule is increasing the chances of being bought very high, then you might want to focus on that rule other than some other rules.

(Refer Slide Time: 04:20)

The screenshot shows RStudio with the following R code in the console:

```
1 # Load the libraries
2 library(arules)
3 library(datasets)
4
5 # Load the data set
6 data(Groceries)
7
8 # Create an Item Frequency plot for the top 20 items
9 itemFrequencyPlot(Groceries, top=20, type="absolute")
10
11 # Support: the fraction of which our item set occurs in our dataset.
12 # Confidence: probability that a rule is correct for a new transaction with items
13 # Lift: the ratio by which the confidence of a rule exceeds the expected confid
14
15 # Get the rules
16 rules = apriori(Groceries, parameter = list(supp = 0.005, conf = 0.8))
17
18 # Show the top 5 rules, but only 2 digits
19 options(digits=2)
20
```

The bar chart on the right shows item frequency (absolute) for the top 20 items. The y-axis ranges from 0 to 2000. The x-axis lists items: whole milk, strawberries, apples, bananas, bread, rice, potatoes, coffee, instant noodle, instant noodle, instant noodle, instant noodle, instant noodle, instant noodle, instant noodle, instant noodle, instant noodle, instant noodle, instant noodle, instant noodle.

So, rules in the previous presentation I have told that that will be analysed based on, so rules has to be a has to have a minimum support. Higher confidence means rule will be accurate and lift high means rule will be useful, simplest form. So, it has to be it has to be supportive that means you the rule is not a bleep, not a statistical error that he has enough support for the rule, you have to have confidence on the rule and the rule has to be in useful. So, that is the

three things based on which we judge which rule we will act on which will discard, so these are the three things.

So, rules to find out the rules, we use the Apriori algorithm with the groceries data set, so the syntax is Apriori within bracket data. And then what is the parameter? I am saying that the support cut off has to be 0.001 that means, 0.1 percentage times it has to occur, as I told that, that is also very rare. And the confidence has to be 0.8 at least. So, at least 80 percent times it has to be correct.

(Refer Slide Time: 05:35)

The top screenshot shows the RStudio interface with the following code in the editor:

```

1 library(datasets)
2
3 # Load the data set
4 data(Groceries)
5
6 # Create an item frequency plot for the top 20 items
7
8 itemFrequencyPlot(Groceries,top=20,type="absolute")
9
10
11 # Support: the fraction of which our item set occurs in our dataset.
12 # Confidence: probability that a rule is correct for a new transaction with items
13 # Lift: the ratio by which the confidence of a rule exceeds the expected confid
14
15 # Get the rules
16 rules = apriori(Groceries, parameter = list(supp = 0.001, conf = 0.8))
17
18 # Show the top 5 rules, but only 2 digits
19 options(digits=2)
20 inspect(rules[1:5])
21
22

```

The console output for the Apriori function is as follows:

```

> rules = apriori(Groceries, parameter = list(supp = 0.001, conf = 0.8))
apriori

Parameter specification:
confidence minval smax area aval originalSupport maxtime support minlen maxlen
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: 9

```

The bottom screenshot shows the code with the following additions:

```

21
22 summary(rules) #summary info of the rules generated
23
24 #the most likely rules
25 rules = sort(rules, by="confidence", decreasing=TRUE)
26 options(digits=2)
27 inspect(rules[1:5])
28
29 rules = apriori(Groceries, parameter = list(supp = 0.001, conf = 0.8, maxlen=3))
30
31

```

The console output for the top 5 rules is:

```

> inspect(rules[1:5])
  lhs              rhs      support confidence lift count
(1) {liquor,red/blush wine} => {bottled beer} 0.0019 0.90 11.2 19
(2) {curd,cereals}         => {whole milk} 0.0010 0.91  3.6 10
(3) {yogurt,cereals}       => {whole milk} 0.0017 0.81  3.2 17
(4) {butter,jam}           => {whole milk} 0.0010 0.83  3.2 10
(5) {soups,bottled beer}   => {whole milk} 0.0011 0.92  3.6 11

```

If I just run this, so there are lots of rules that has been created. And if I just want to see a summary of those rules. So, let us say inspect the top 5 first 5 rules. These are the first 5 rules that has been created. Now, these 5 rules are not based on anything they are randomly

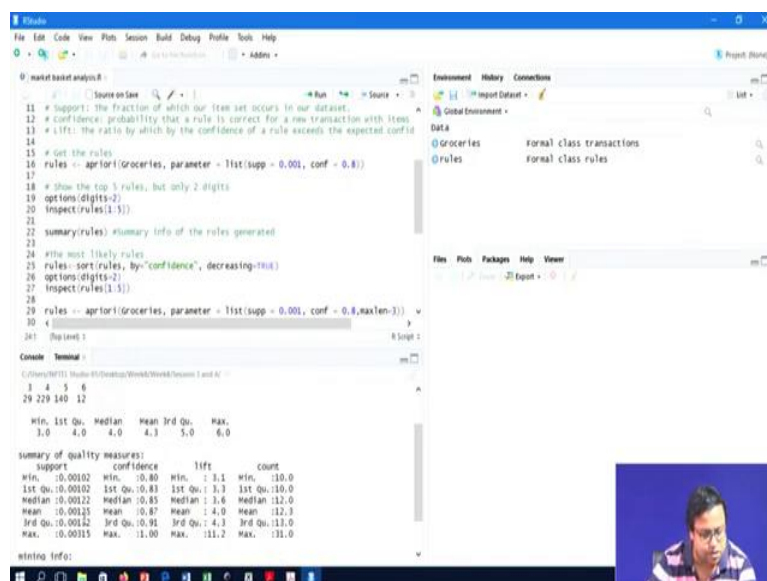
created. Like the first rule says in the left hand side, we have got liquor, and then we all we have got liquor and red blush wine and the right hand side we have bottled water, bottled beer that means, given that liquor and wine occurs, what is the probability that the bottle beer will also occur.

So, corresponding support is 0.0019 that means, one up 0.19 percent cases that left-hand side occurs which is small. But when the left-hand side occurs, right hand side occurs with 90 percent probability 0.9 is my confidence, the lift is also very high 11.2. So, otherwise it will not occur, but the moment the left-hand side occurs, what will be a any way people not buy, but if left hand side occurs then the parts probability of buying is 11.2 times normally buying, so, that is a huge lift.

Similarly, curd and cereals, this has a support which is low, but when that occurs purchasing of whole milk is very high, confidence is 0.9 and so on. But the lift is only 3.6, why lift and 6? Because anyway people buy whole milk, I have shown you 2500 times out of around 10,000 times at least 40 percent times people anyway by whole milk, almost 40 percent times.

So, given that this occurs the percentage is 0.91 that is not very high, the lift is 3.6 that means actually, normally people buy whole milk 0.91 divided by 3.6 this is the natural probability of buying a whole milk. Similarly, 0.81 divided by 3.2, it is also will come 0.25, so that is the natural probability of buying whole milk. So, 25 percent times people anyway buy whole milk, so, that is why the lift is not so high.

(Refer Slide Time: 08:02)



```
## market basket analysis ##
11 # support: the fraction of which our item set occurs in our dataset.
12 # confidence: probability that a rule is correct for a new transaction with items.
13 # lift: the ratio by which the confidence of a rule exceeds the expected confid
14
15 # Get the rules
16 rules = apriori(groceries, parameter = list(supp = 0.001, conf = 0.8))
17
18 # Show the top 5 rules, but only 2 digits
19 options(digits=2)
20 inspect(rules[1:5])
21
22 summary(rules) #summary info of the rules generated
23
24 #the most likely rules
25 rules=sort(rules, by="confidence", decreasing=TRUE)
26 options(digits=2)
27 inspect(rules[1:5])
28
29 rules = apriori(groceries, parameter = list(supp = 0.001, maxlen=3))
30 4
31 ## (base) >
32
33 Console Terminal
34 C:\Users\NPT11\Desktop\Work\Work\Topic 3 and 4 >
35 3 4 5 6
36 29 229 140 12
37
38 Min. 1st Qu. median Mean 3rd Qu. Max.
39 3.0 4.0 4.0 4.3 5.0 6.0
40
41 summary of quality measures:
42 support confidence lift count
43 Min. :0.00102 Min. :0.80 Min. : 3.1 Min. :10.0
44 1st Qu.:0.00102 1st Qu.:0.83 1st Qu.: 3.3 1st Qu.:10.0
45 Median :0.00122 median :0.85 median : 3.6 Median :12.0
46 Mean :0.00135 Mean :0.87 Mean : 4.0 Mean :12.3
47 3rd Qu.:0.00132 3rd Qu.:0.91 3rd Qu.: 4.3 3rd Qu.:13.0
48 Max. :0.00115 Max. :1.00 Max. :11.2 Max. :13.0
49
50 writing info:
```



```

marketbasketanalysis.R
# Home on file
# # support: the fraction of which our item set occurs in our dataset.
# # confidence: probability that a rule is correct for a new transaction with items
# # lift: the ratio by which the confidence of a rule exceeds the expected confid
# # Get the rules.
# rules = apriori(groceries, parameter = list(supp = 0.001, conf = 0.8))
# # Show the top 5 rules, but only 2 digits
# options(digits=2)
# inspect(rules(1:5))
# summary(rules) #summary info of the rules generated
# #the most likely rules
# rules = sort(rules, by="confidence", decreasing=TRUE)
# options(digits=2)
# inspect(rules(1:5))
# rules = apriori(groceries, parameter = list(supp = 0.001, conf = 0.8, maxlen=3))
# # the level
# # Script

Console Terminal
C:\Users\NPTU\OneDrive\Work\Work\Week1\Week1 and 2
> summary(rules) #Summary info of the rules generated
set of 410 rules

rule length distribution (lhs + rhs):sizes
  3  4  5  6
29 229 140 12

summary of quality measures:
      support      confidence      lift      count
min. :0.00102 min. :0.80 min. : 3.1 min. :10.0
1st Qu.:0.00102 1st Qu.:0.83 1st Qu.: 3.3 1st Qu.:10.0
Median :0.00122 Median :0.85 Median : 3.6 Median :12.0
Mean :0.00125 Mean :0.87 Mean : 4.0 Mean :12.3

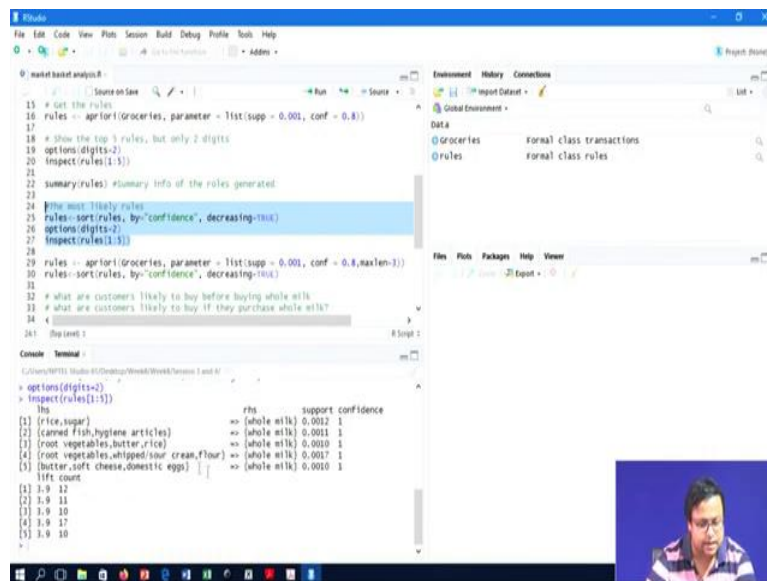
```

Now, but will I do? I will find out a summary of this rules. So, the average support is 0.00125 the maximum support is 0.003 and the minimum is 0.001 that that is where the cut of that we have taken. But confidence is 0.8 to 1, the average confidence is 0.87, lift is 0.31 to 11.2 which is good and so is the rules.

So, we get some basic rules, the rules are distributed based on the sizes also. So, three size left hand side and right hand side together three items rule that 29 such rules. Two items, four items rules there are 229 such rules, 5 items 140 and six items 12, so that is the distribution that has been given that total 410 rules based on the cut off that we have created.

Now, which rules are the most likely that means most likely means? Most accurate. So, I will this thing I will sort them based on confidence. So, rules is, I sort the rules by confidence in a decreasing order.

(Refer Slide Time: 09:17)



```
0: market basket analysis.R
15 # Get the rules
16 rules = apriori(groceries, parameter = list(supp = 0.001, conf = 0.8))
17
18 # Show the top 3 rules, but only 2 digits
19 options(digits=2)
20 inspect(rules[1:3])
21
22 summary(rules) #Summary info of the rules generated
23
24 #We want "highly" rules
25 rules=sort(rules, by="confidence", decreasing=TRUE)
26 options(digits=2)
27 inspect(rules[1:5])
28
29 rules = apriori(groceries, parameter = list(supp = 0.001, conf = 0.8, maxlen=3))
30 rules=sort(rules, by="confidence", decreasing=TRUE)
31
32 # what are customers likely to buy before buying whole milk?
33 # what are customers likely to buy if they purchase whole milk?
34
351 > inspect(rules[1:5])

```

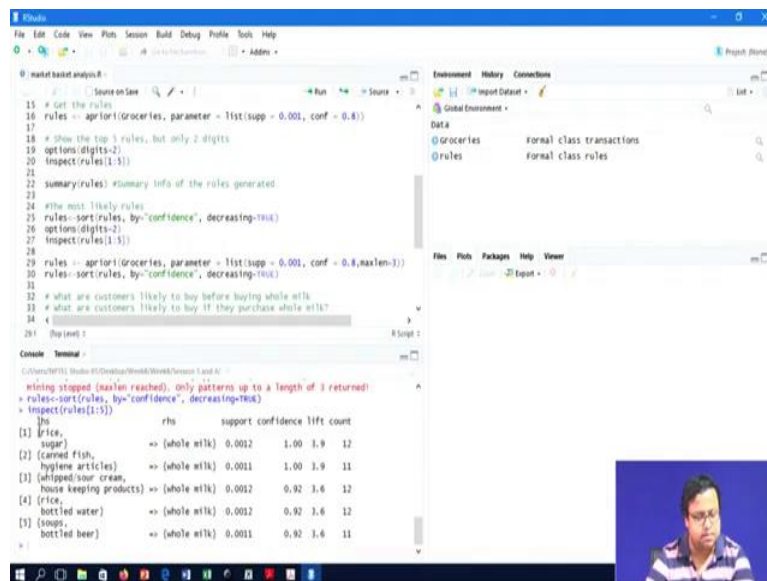
lhs	rhs	support	confidence
{1} {rice,sugar}	=> {whole milk}	0.0012	1
{2} {canned fish,hygiene articles}	=> {whole milk}	0.0011	1
{3} {root vegetables,butter,rice}	=> {whole milk}	0.0010	1
{4} {root vegetables,whipped/sour cream,flour}	=> {whole milk}	0.0017	1
{5} {butter,soft cheese,domestic eggs}	=> {whole milk}	0.0010	1

So, if I now, just plot them, these are the rules which have confidence is 1, they are most. So, whenever people buy these, they always buy whole milk, so that is thing. But you see their count is very small first of all and the support is also very small. Now, I can also see that which of these guys have a maximum length of 3 that means, there are lots combinations has to happen then only this happens.

For example, this one, you will see that root vegetables, whipped sour cream, flower, happens, then whole milk happens, this rule number 4 has 4 items which is big, or let us say this one butter, soft cheese, domestic eggs, 3 items happening in the left hand side, then only 1 item happening in the right hand side that is also very big rule.

So, if I just want to limit my size of the rule, I can do maxlen is equal to 3, maxlen is equal to 3 means? The maximum length both side included, right hand and then right hand side and left hand side, the length of the rules will be 3 items at max.

(Refer Slide Time: 10:28)

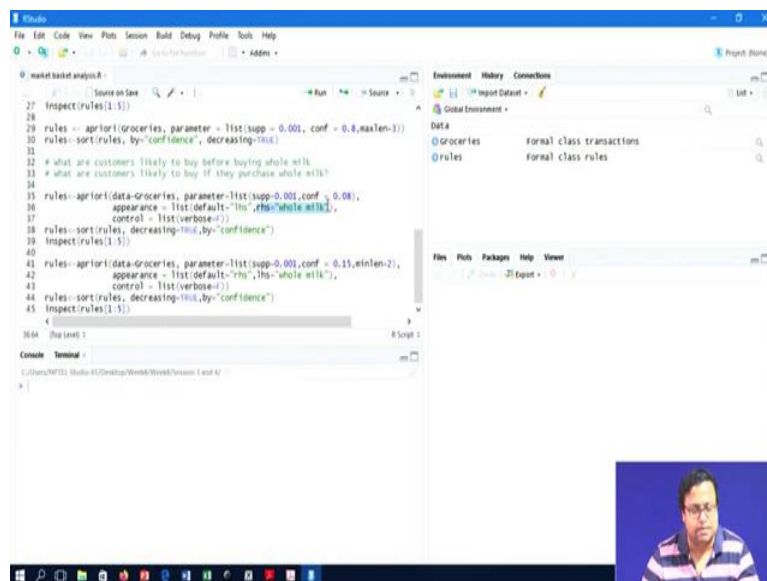


```
market basket analysis.R
# Get the rules
rules = apriori(groceries, parameter = list(supp = 0.001, conf = 0.8))
# Show the top 3 rules, but only 2 digits
options(digits=2)
inspect(rules(1:3))
summary(rules) #Summary info of the rules generated
# the most likely rules
rules=sort(rules, by="confidence", decreasing=TRUE)
options(digits=2)
inspect(rules(1:5))
rules = apriori(groceries, parameter = list(supp = 0.001, conf = 0.8, maxlen=3))
rules=sort(rules, by="confidence", decreasing=TRUE)
# what are customers likely to buy before buying whole milk?
# what are customers likely to buy if they purchase whole milk?
# By level:
# Script:
C:\Users\NPT11\Desktop\Week4\Week4\Source 1 and 4/
# mining stopped (maxlen reached). Only patterns up to a length of 3 returned!
# rules=sort(rules, by="confidence", decreasing=TRUE)
# inspect(rules(1:5))
  rhs      support confidence lift count
[1] rice,    sugar => (whole milk) 0.0012 1.00 1.9 12
[2] (canned fish, hygiene articles) => (whole milk) 0.0011 1.00 1.9 11
[3] (whipped/soor cream, house keeping products) => (whole milk) 0.0012 0.92 3.6 12
[4] (rice, bottled water) => (whole milk) 0.0012 0.92 3.6 12
[5] (soups, bottled beer) => (whole milk) 0.0011 0.92 3.6 11
```

So, now if I do that, and then I sort it by confidence, what will I look like? So, I have created them, and now if I just inspect them, see the maximum items are 3 in all the cases not more than that, again the confidence is very high, but the support is small. So, this kind of this and that you can do, this kind of changes, this kind of playing you can do.

I will do two more playing and then I will close. For example, I can also say that, given this particular item is in the left hand side, what is the chances or given hold thing in the right hand side, what is what are the rules? So, that I can say.

(Refer Slide Time: 11:02)



```
market basket analysis.R
# Get the rules
rules = apriori(groceries, parameter = list(supp = 0.001, conf = 0.8, maxlen=3))
inspect(rules(1:5))
rules = apriori(data=groceries, parameter=list(supp=0.001, conf = 0.11, maxlen=2),
  appearance = list(default="rhs", lhs="whole milk"),
  control = list(verbose=1))
rules=sort(rules, decreasing=TRUE, by="confidence")
inspect(rules(1:5))
# By level:
# Script:
C:\Users\NPT11\Desktop\Week4\Week4\Source 1 and 4/
```

For example, here what I am trying? I am putting up appearances is equal to list, left hand side is default and right-hand side has to have whole milk. So, the moment I am saying right hand side has to have whole milk and then I am expecting.

(Refer Slide Time: 11:21)

```

27 inspect(rules[1:5])
28
29 rules = apriori(Groceries, parameter = list(supp = 0.001, conf = 0.8, maxlen = 3))
30 rules = sort(rules, by = "confidence", decreasing = TRUE)
31
32 # what are customers likely to buy before buying whole milk?
33 # what are customers likely to buy if they purchase whole milk?
34
35 rules = apriori(data = Groceries, parameter = list(supp = 0.001, conf = 0.08),
36             appearance = list(default = "rhs", rhs = "whole milk"),
37             control = list(verbose = TRUE))
38 rules = sort(rules, decreasing = TRUE, by = "confidence")
39 inspect(rules[1:5])
40
41 rules = apriori(data = Groceries, parameter = list(supp = 0.001, conf = 0.11, minlen = 2),
42             appearance = list(default = "rhs", rhs = "whole milk"),
43             control = list(verbose = TRUE))
44 rules = sort(rules, decreasing = TRUE, by = "confidence")
45 inspect(rules[1:5])

```

```

> rules = sort(rules, decreasing = TRUE, by = "confidence")
> inspect(rules[1:5])
  lhs      rhs      support confidence
(1) {rice,sugar} => {whole milk} 0.0012 1
(2) {canned fish,hygiene articles} => {whole milk} 0.0011 1
(3) {root vegetables,butter,rice} => {whole milk} 0.0010 1
(4) {root vegetables,whipped sour cream,flour} => {whole milk} 0.0017 1
(5) {butter,soft cheese,domestic eggs} => {whole milk} 0.0010 1
lift count
(1) 3.9 12
(2) 3.9 11
(3) 3.9 10
(4) 3.9 17
(5) 3.9 10

```

So, then I am saying the right-hand side has to, it must that it will have whole milk and corresponding then I am sorting it based on confidence and that values I am getting. Similarly, I can say left-hand side also must have whole milk.

(Refer Slide Time: 11:35)

```

41 rules = apriori(data = Groceries, parameter = list(supp = 0.001, conf = 0.11, minlen = 2),
42             appearance = list(default = "rhs", lhs = "whole milk"),
43             control = list(verbose = TRUE))
44 rules = sort(rules, decreasing = TRUE, by = "confidence")
45 inspect(rules[1:5])

```

```

> rules = apriori(data = Groceries, parameter = list(supp = 0.001, conf = 0.11, minlen = 2),
+             appearance = list(default = "rhs", lhs = "whole milk"),
+             control = list(verbose = TRUE))
+ rules = sort(rules, decreasing = TRUE, by = "confidence")
+ inspect(rules[1:5])
  lhs      rhs      support confidence lift count
(1) {whole milk} => {other vegetables} 0.0715 0.29 1.5 736
(2) {whole milk} => {rolls,buns} 0.0517 0.22 1.2 557
(3) {whole milk} => {yogurt} 0.0316 0.22 1.6 551
(4) {whole milk} => {root vegetables} 0.0409 0.19 1.8 482
(5) {whole milk} => {tropical fruit} 0.042 0.17 1.6 416

```

So, that will create this kind of rules, for left-hand side must have whole milk there is no other choice. And you say I have written min-length is equal to 2, min-length is equal to 2 means? The minimum length of this particular thing has to be 2. So, minimum length is

always 2, there will be rules which are more than 2. If I do max-length is equal to 2 then there will be absolutely 1 item in the left-hand side 1 has in the right-hand side. So, that is how we create lots of rules, you can sort by confidence, you can also sort by lift, you can sort by support or you can create a combination of lift and support.

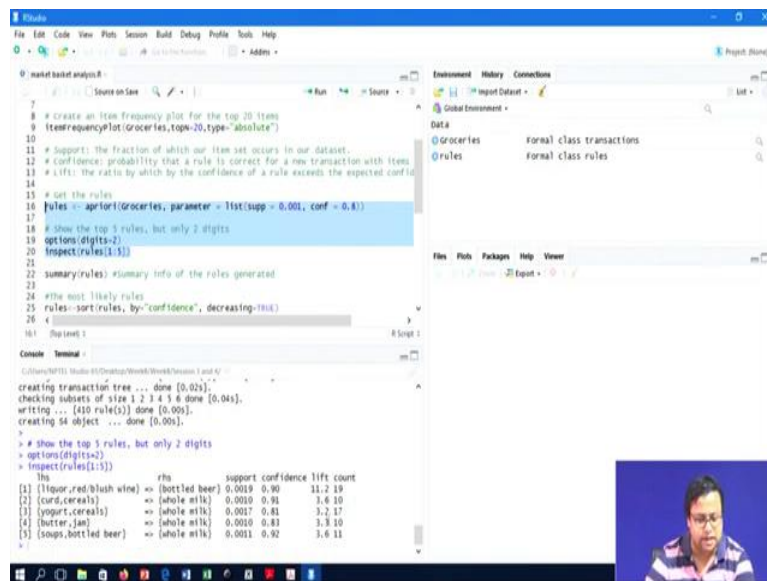
(Refer Slide Time: 12:15)

```

27 inspect(rules[1:5])
28
29 rules = apriori(groceries, parameter = list(supp = 0.001, conf = 0.8, maxlen = 3))
30 rules = sort(rules, by = "confidence", decreasing = TRUE)
31
32 # what are customers likely to buy before buying whole milk?
33 # what are customers likely to buy if they purchase whole milk?
34
35 rules = apriori(data = groceries, parameter = list(supp = 0.001, conf = 0.08),
36 appearance = list(default = "lhs", rhs = "whole milk"),
37 control = list(verbose = F))
38 rules = sort(rules, decreasing = TRUE, by = "confidence")
39 inspect(rules[1:5])
40
41 rules = apriori(data = groceries, parameter = list(supp = 0.001, conf = 0.13, minlen = 2),
42 appearance = list(default = "rhs", lhs = "whole milk"),
43 control = list(verbose = F))
44 rules = sort(rules, decreasing = TRUE, by = "confidence")
45 inspect(rules[1:5])
46
47 # Script
48
49 Console Terminal
50 C:\Users\NPT11\Studio 01\Desktop\Work\Work\Session 3 and 4\
51 [1] 1.9 10
52 [4] 1.9 17
53 [5] 1.9 10
54 + rules = apriori(data = groceries, parameter = list(supp = 0.001, conf = 0.13, minlen = 2),
55 + appearance = list(default = "rhs", lhs = "whole milk"),
56 + control = list(verbose = F))
57 + rules = sort(rules, decreasing = TRUE, by = "confidence")
58 + inspect(rules[1:5])
59
60 support confidence lift count
61 [1] whole milk => {other vegetables} 0.075 0.29 1.5 736
62 [2] whole milk => {rolls;buns} 0.057 0.22 1.2 595
63 [3] whole milk => {yogurt} 0.056 0.22 1.6 553
64 [4] whole milk => {root vegetables} 0.049 0.19 1.8 481
65 [5] whole milk => {tropical fruit} 0.042 0.17 1.6 416
  
```

And so, because the rules are all stored here, so, if I if we want to find out that for support, I will give x percentage weightage, confidence I will give y percentage weightage, you can do multiple combinations to find out which rules are acceptable and based on that you can take a call. So, here we are just creating the rules, playing with the rules left-hand, right-hand side I am sticking up and doing something, but later the marketing decision is part will I do with rules?

(Refer Slide Time: 12:47)



```
0 market basket analysis.R
7
8 # Create an item frequency plot for the top 20 items
9 itemfrequencyplot(groceries,top=20,type="absolute")
10
11 # Support: the fraction of which our item set occurs in our dataset.
12 # Confidence: probability that a rule is correct for a new transaction with items
13 # Lift: the ratio by which the confidence of a rule exceeds the expected confid
14
15 # Get the rules
16 rules = apriori(groceries, parameter = list(supp = 0.001, conf = 0.8))
17
18 # Show the top 5 rules, but only 2 digits
19 options(digits=2)
20 inspect(rules[1:5])
21
22 summary(rules) # Summary info of the rules generated
23
24 #the most likely rules
25 rules=sort(rules, by="confidence", decreasing=TRUE)
26 <
27 <
28 <
29 <
30 <
31 <
32 <
33 <
34 <
35 <
36 <
37 <
38 <
39 <
40 <
41 <
42 <
43 <
44 <
45 <
46 <
47 <
48 <
49 <
50 <
51 <
52 <
53 <
54 <
55 <
56 <
57 <
58 <
59 <
60 <
61 <
62 <
63 <
64 <
65 <
66 <
67 <
68 <
69 <
70 <
71 <
72 <
73 <
74 <
75 <
76 <
77 <
78 <
79 <
80 <
81 <
82 <
83 <
84 <
85 <
86 <
87 <
88 <
89 <
90 <
91 <
92 <
93 <
94 <
95 <
96 <
97 <
98 <
99 <
100 <
```

```
creating transaction tree ... done [0.02s].
checking subsets of size 1 2 3 4 5 6 done [0.04s].
writing ... [150 rules] done [0.00s].
creating 54 object ... done [0.00s].
>
> # Show the top 5 rules, but only 2 digits
> options(digits=2)
> inspect(rules[1:5])
  lhs              rhs      support confidence lift count
(1) {liquor,red/blush wine} => {bottled beer} 0.0019 0.90 11.2 19
(2) {curd,cereals}         => {whole milk} 0.0010 0.91 3.6 10
(3) {yogurt,cereals}       => {whole milk} 0.0017 0.81 3.2 17
(4) {butter,jam}           => {whole milk} 0.0010 0.83 3.3 10
(5) {soups,bottled beer}  => {whole milk} 0.0011 0.92 3.6 11
```

For example, what will I do if I know that rice and sugar along with that whole milk is also being sold? Or if you remember the initial rules that we have created and the very first one just one minute. Yeah. So, now if I just inspect the rules, the very first one, yeah. So, this is also 3.9, not this one sorry.

So, very first one was this. So, if I just checked, so what will I do with this? If I know the lift is very high, confidence is very high, support is not bad, what will I do with this rule? Liquor and wine, red wine and along with that bottled beer is also being sold, then I have to put them together or I have to create a bundle or I can put them into different ends, so that people can purchase.

So, these kind of decisions you have to take as a marketing manager. This part is only a data science to create hypothesis that hypothesis can be tested, can be used, once it is tested true to create marketing insights. So, that is what all about Market Basket Analysis. Analysis but is very easy, its understanding is very easy, and you have to use that in your marketing context. So, that is all for week 8. We will meet you in week 9. And we will discuss about further aspects of franchises and later text mining. Thank you for being with me and I will see you in the next video.