

**Predictive Analytics - Regression and Classification**  
**Prof. Sourish Das**  
**Department of Mathematics**  
**Chennai Mathematical Institute**

**Lecture - 59**  
**Hands on with R: Dynamic Pricing with Cheese Data**

Hello all, in this video I am going to use show how can we use regression analysis in dynamic pricing. What is dynamic pricing? Sometimes you will see from if you want to buy air ticket from say Chennai to Bombay it might cost you maybe 7000 rupees in one way, in another day it might cost you 9000 rupees and another day it might cost you 6000 rupees. Why airline charge different price for different days or different time slot for different, though it is travelling the it is in the same route.

The air travel the price of these cases depends on the availability of the seat and the demand for the seat. If the demand for the seat is high then airline charges higher fees or higher fare for the same ticket. If the sub demand for the seat is low then airline charges a bare minimum base price for the air fare.

So, this kind of discriminatory pricing often called in industry dynamic pricing based on the demand and supply of the product or the service that are available. So, in this video I am going to use cheese data set from bayesm package to demonstrate how can you use regression analysis for dynamic pricing.

(Refer Slide Time: 02:15)

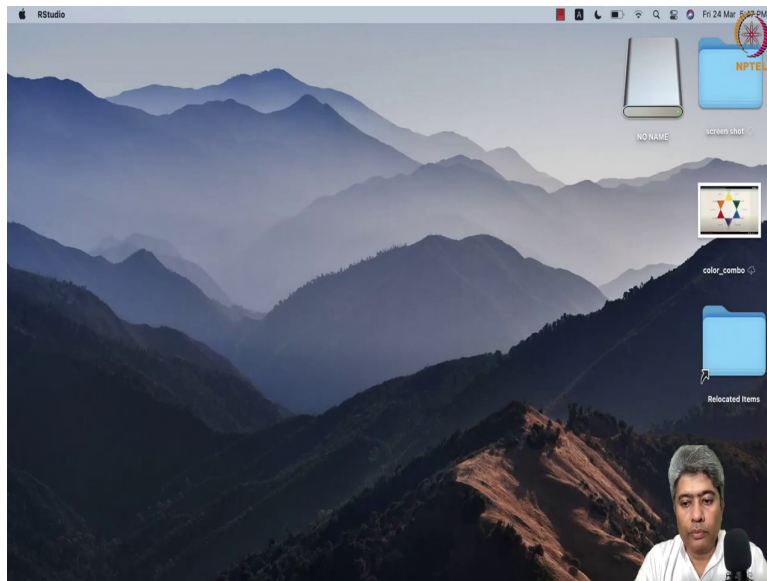


So, first I will open my R.

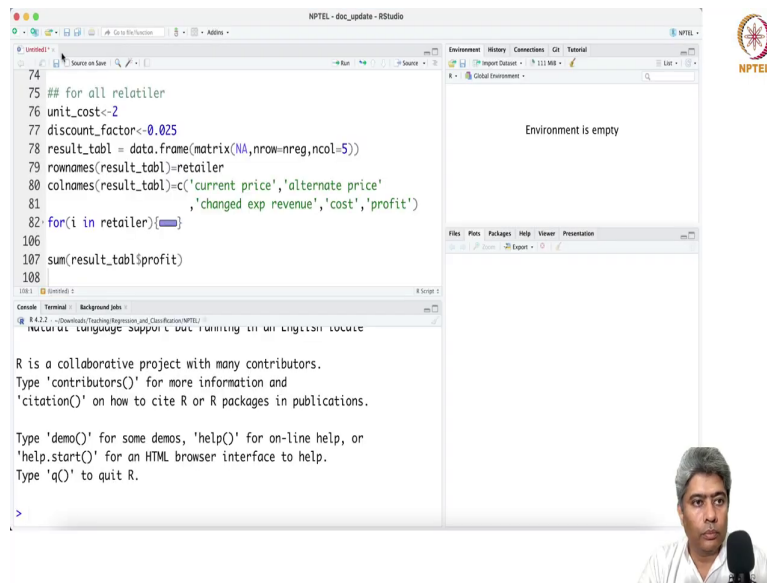
(Refer Slide Time: 02:19)



(Refer Slide Time: 02:21)



(Refer Slide Time: 02:25)



The image shows a screenshot of the RStudio interface. The main editor window contains the following R code:

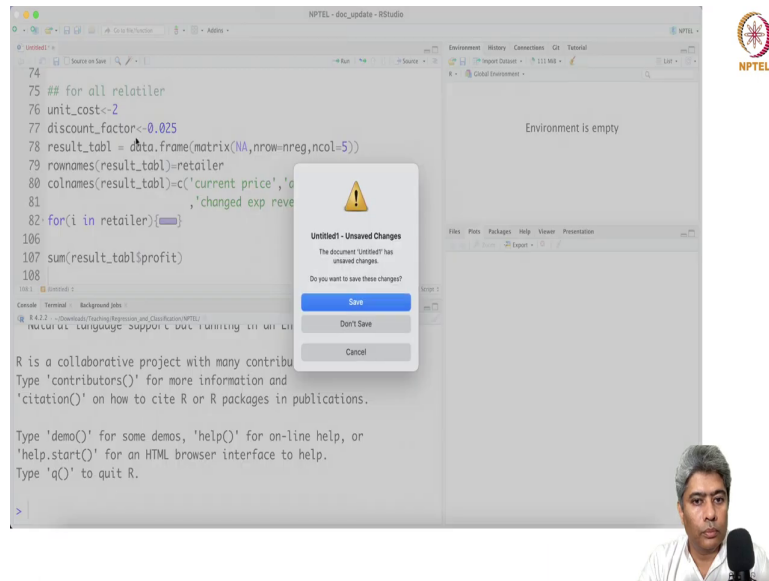
```
74  
75 ## for all retailer  
76 unit_cost<-2  
77 discount_factor<-0.025  
78 result_tabl = data.frame(matrix(NA,nrow=nreg,ncol=5))  
79 rownames(result_tabl)=retailer  
80 colnames(result_tabl)=c('current price','alternate price'  
81 , 'changed exp revenue','cost','profit')  
82 for(i in retailer){  
106  
107 sum(result_tabl$profit)  
108  
>
```

The Environment pane on the right shows "Environment is empty". The Console pane at the bottom displays the following text:

```
108.1 | Evented | R Script |  
R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.  
  
Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.  
>
```

In the bottom right corner, there is a small video inset showing a man speaking into a microphone. The NPTEL logo is visible in the top right corner of the RStudio window.

(Refer Slide Time: 02:23)



The image shows a screenshot of the RStudio interface. The main window displays a script with the following R code:

```
74  
75 ## for all retailer  
76 unit_cost<-2  
77 discount_factor<-0.025  
78 result_tabl = data.frame(matrix(NA,nrow=nreg,ncol=5))  
79 rownames(result_tabl)=retailer  
80 colnames(result_tabl)=c('current price','d  
81 , 'changed exp reve  
82 for(i in retailer){  
106  
107 sum(result_tabl$profit)  
108  
109 }  
110  
111
```

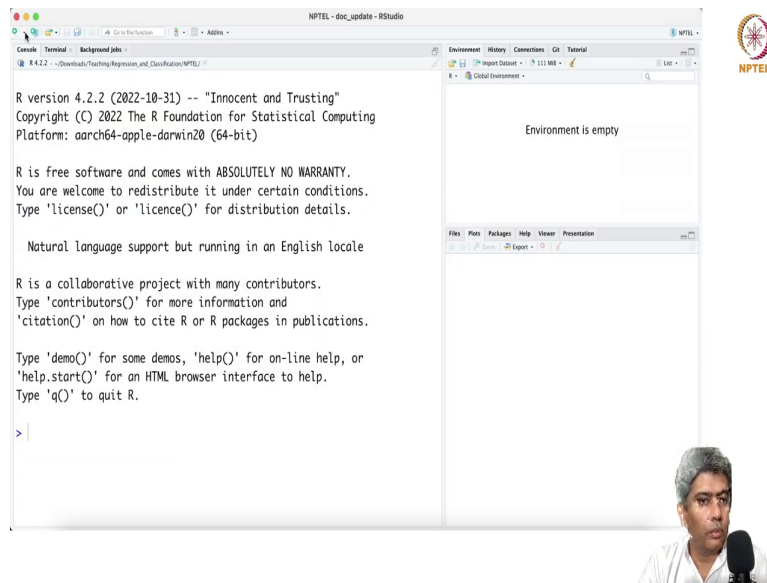
The console window at the bottom shows the following text:

```
R is a collaborative project with many contribu  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.  
  
Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.  
>
```

A dialog box titled "Untitled1 - Unsaved Changes" is overlaid on the script, asking "Do you want to save these changes?" with buttons for "Save", "Don't Save", and "Cancel".

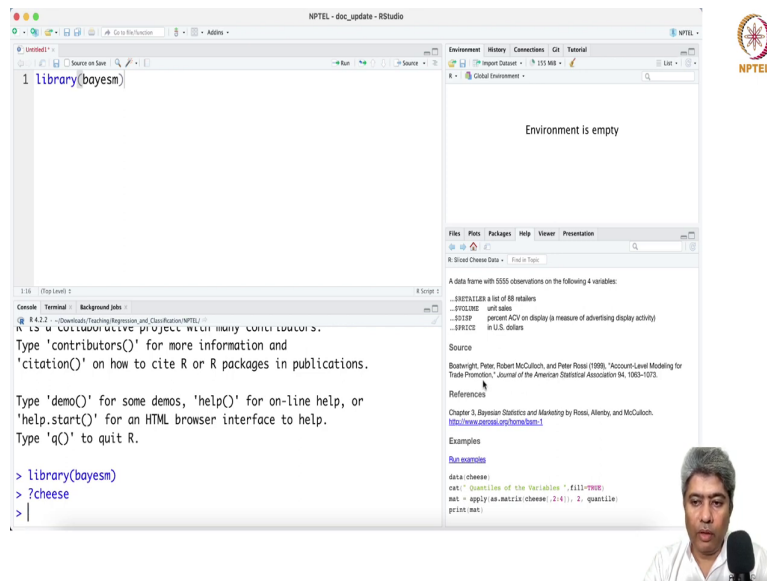
The NPTEL logo is visible in the top right corner of the RStudio window. A small inset image of a man speaking into a microphone is located in the bottom right corner of the overall image.

(Refer Slide Time: 02:27)



Ok.

(Refer Slide Time: 02:31)



The screenshot shows the RStudio interface with the following content:

- Source Editor:** Contains the code `library(bayesm)`.
- Environment:** Shows "Environment is empty".
- Files:** Shows "R Sliced Cheese Data".
- Terminal:** Contains the following text:

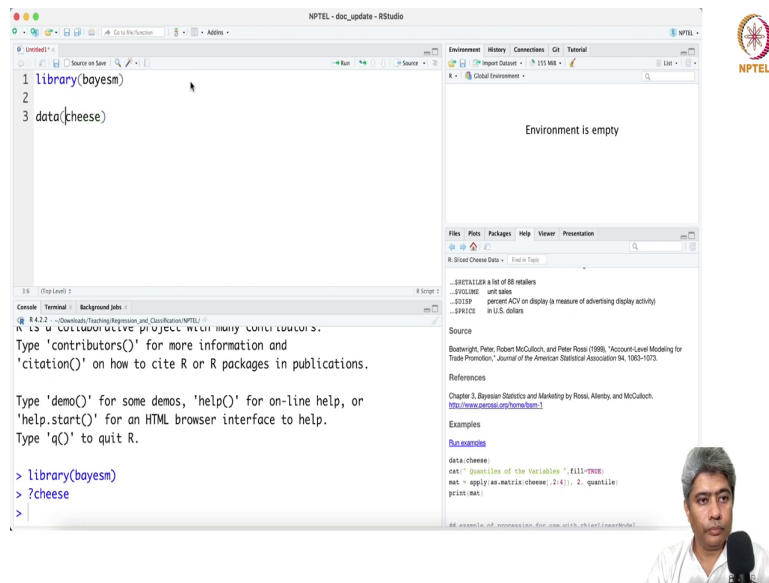
```
> library(bayesm)
> ?cheese
> |
```
- Help Panel:** Displays the documentation for the "R Sliced Cheese Data" dataset, including a description of the data frame (5555 observations, 4 variables: RETAILERS, STORES, PLOTLINE, and SPREAD) and a reference to the book "Account-Level Modeling for Trade Promotion" by Robert McCulloch and Peter Rossi.

An NPTEL logo is visible in the top right corner, and a small video feed of a man is in the bottom right corner.

So, first I am going to call library bayesm. In the If you do not have this library just, please install this library and in this bayesm package there is a called data set available called sliced cheese data set ok. The reference is also available in McCulloch and you know Peter McCulloch and Peter Rossis Robert McCulloch and Peter Rossis book on Account-Level Modeling for Trade Promotion.



(Refer Slide Time: 03:17)



The image shows a screenshot of the RStudio interface. The main editor window contains the following R code:

```
1 library(bayesm)
2
3 data(cheese)
```

The console window displays the following text:

```
R 4.2.2 - Overview: Training Regression and Classification (NPTEL)
R is a COLLABORATIVE project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> library(bayesm)
> ?cheese
>
```

The help window for the 'cheese' dataset is open, showing the following information:

**Environment is empty**

**Files** **Plots** **Packages** **Help** **Viewer** **Presentation**

R: **cheese** Data - Find in Type

**DESCRIPTION** A list of 88 retailers  
...**SYNOPSIS** unit sales  
...**DETAILS** percent ACP on display (a measure of advertising display activity)  
...**FILES** in U.S. dollars

**Source**  
Boothwright, Peter, Robert McCulloch, and Peter Rossi (1998), "Account-Level Modeling for Trade Promotions," *Journal of the American Statistical Association* 94, 1003-1073.


**References**  
Chapter 3, *Bayesian Statistics and Marketing* by Rossi, Allenby, and McCulloch.  
<http://www.person.umd.edu/~rossi/rlsart/learn08de>

**Examples**

```
data(cheese)
cat("quantiles of the variables ", fill=TRUE)
mat = apply(as.matrix(cheese), 2, FUN=function(x) {
  print(x)
})
```

# An example of a regression function for use with rpart (non-robust)

NPTEL



And in this paper the this data set first appeared. So, first I am going to call this data set data cheese.

(Refer Slide Time: 03:43)

The screenshot displays the RStudio interface with the 'cheese' dataset loaded. The Environment pane shows 'cheese' with 5555 observations and 4 variables. The console shows the following R code:

```
> library(bayesm)
> ?cheese
> data(cheese)
> View(cheese)
>
```

The source pane shows the following code:

```
## RETAILER a list of 88 retailers
## VOLUME unit sales
## DISPLAY percent A/C on display (a measure of advertising display activity)
## PRICE in U.S. dollars
```

The console also shows the following text:

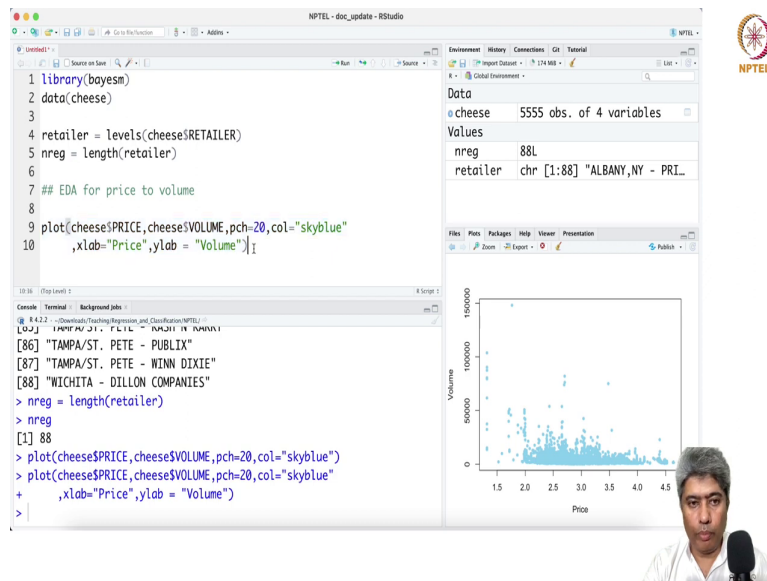
```
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

R 4.2.2 > download.packages(repositories=getOption("repos"),
                           which.packages=getOption("packages"))
```

So, here you have the data set. So, you have four columns retailer name of the retailer there are about 88 it says 88 retailers LOS ANGELES-LUCKY, LOS ANGELES- RALPHS, LOS ANGELES-VONS. So, Lucky, Ralph, Vons, Dominick these are the retailer chain and this is the city name.

So, city name and retailer chain they are different and; obviously, and then along with that the VOLUME, VOLUME or the unit sales the number of volume sales that happened in that week. This typically called display activity that has happened some marketing activity they measured in the scale of 0 to 1 and the PRICE per unit price in US dollar is given.

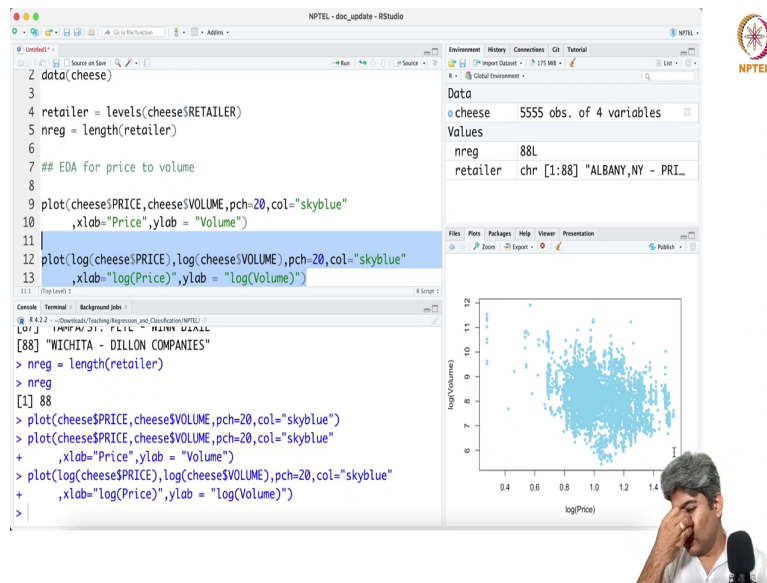
(Refer Slide Time: 04:40)



So, so, the first thing I am going to do I am going to take the retailer equal to check what are the levels that are available in cheese data set. So, you can see there are 88 retailer different retailer are available. So, I am going to say there reg. So, these are the number of leng retailer is 88, ok. Now, I am going to do some Exploratory Data Analysis EDA for price to volume data to volume.

So, the first thing is plot cheese dollar PRICE and cheese dollar VOLUME for each price what are the cells that you see. And say ch equal to say 20 and color equal to “skyblue”, ok. So, it looks like this how that is how the price and so, xlab equal to “Price” and ylab equal to “Volume” or unit cell number of unit cells that has happened ok. Let me just run it.

(Refer Slide Time: 07:08)



And now, as I told you in the in a previous video that, if it is any economic variable one should use log transformation; so, I am going to use log transformation over PRICE and VOLUME and. So, it will be “log Price” versus “log Volume”. So, looks like log Price versus log Volume has a sort of a you know expected slope, which is sort of fitting one line kind of thing you see.

And now so; that means, most likely Log Price to Log Volume modeling Log Price and Log Volume will be much more you know effective than you know trying to do something different. So, next what I am going to do is say if you say retailer.

(Refer Slide Time: 08:26)

The screenshot displays an RStudio interface with the following components:

- Source Editor:** Contains R code for data analysis:

```
3  
4 retailer = levels(cheese$RETAILER)  
5 nreg = length(retailer)  
6  
7 ## EDA for price to volume  
8  
9 plot(cheese$PRICE,cheese$VOLUME,pch=20,col="skyblue"  
10 ,xlab="Price",ylab = "Volume")  
11  
12 plot(log(cheese$PRICE),log(cheese$VOLUME),pch=20,col="skyblue"  
13 ,xlab="log(Price)",ylab = "log(Volume)")  
14
```
- Environment:** Shows the 'Data' environment with 'cheese' (5555 obs. of 4 variables) and 'Values' for 'nreg' (88L) and 'retailer' (chr [1:88] "ALBANY,NY - PRI...").
- Console:** Shows the output of the 'retailer' variable:

```
> retailer  
[1] "ALBANY,NY - PRICE CHOPPER"  
[2] "ATLANTA - KROGER CO"  
[3] "ATLANTA - WINN DIXIE"  
[4] "BALTI/WASH - GIANT FOOD INC"  
[5] "BALTI/WASH - SAFEWAY"  
[6] "BALTI/WASH - SUPER FRESH"  
[7] "BIRMINGHAM/MONTGOM - BRUNOS"  
[8] "BIRMINGHAM/MONTGOM - KROGER"  
[9] "BIRMINGHAM/MONTGOM - WINN DIXIE"  
[10] "BOSTON - SHAWNS"
```
- Plots:** A scatter plot titled 'log(Volume)' vs 'log(Price)' showing a positive correlation between the two variables. The plot uses blue circles (pch=20) for data points.
- Speaker:** A small video inset of a man speaking into a microphone is visible in the bottom right corner of the RStudio window.

So, ALBANY PRICE CHOPPER and so, Albany is a small place and I we can have a look like you know maybe Los Angeles is a big place “LOS ANGELES – LUCKY” and they are also bit different places. So, let me what I am going to do is something like this.

(Refer Slide Time: 08:54)

The screenshot shows the RStudio interface with the following components:

- Source Editor:** Contains R code for data filtering and plotting:

```
6  
7 ## EDA for price to volume  
8  
9 plot(cheese$PRICE,cheese$VOLUME,pch=20,col="skyblue"  
10 ,xlab="Price",ylab = "Volume")  
11  
12 plot(log(cheese$PRICE),log(cheese$VOLUME),pch=20,col="skyblue"  
13 ,xlab="log(Price)",ylab = "log(Volume)")  
14 cheese_albany=cheese[cheese$RETAILER=="ALBANY,NY - PRICE CHOPPER",]  
15  
16  
17
```
- Environment:** Shows the data objects: 'cheese' (5555 obs. of 4 variables) and 'cheese\_al...' (61 obs. of 4 variables).
- Values:** Shows the structure of the 'cheese\_albany' object: 'nreg' (88L) and 'retailer' (chr [1:88] "ALBANY,NY - PRI...").
- Console:** Shows the execution of the R code, listing the 88 observations for the 'cheese\_albany' subset:

```
14:46 (Stop Level) | R Script |  
[80] "ST. LOUIS - SCHNUCK MARKETS"  
[81] "ST. LOUIS - SCHNUCK MARKETS"  
[82] "SYRACUSE - P & C FOOD MARKE"  
[83] "SYRACUSE - PRICE CHOPPER"  
[84] "SYRACUSE - WEGMANS"  
[85] "TAMPA/ST. PETE - KASH N KARRY"  
[86] "TAMPA/ST. PETE - PUBLIX"  
[87] "TAMPA/ST. PETE - WINN DIXIE"  
[88] "WICHITA - DILLON COMPANIES"  
> cheese_albany=cheese[cheese$RETAILER=="ALBANY,NY - PRICE CHOPPER",]  
>
```
- Plots:** A scatter plot titled 'log(Volume) vs log(Price)' showing a positive correlation between the log of price and the log of volume for the selected subset of data. The x-axis ranges from 0.4 to 1.4, and the y-axis ranges from 6 to 12.
- Background Jobs:** Shows the execution of the R script.

So, first I am going to make a subplot or you know subset of the data. So, cheese, but its only for the albany ok. So, the cheese dataset dollar RETAILER equal to. So, let me just copy this and comma if I just do that.

(Refer Slide Time: 09:42)

The screenshot displays the RStudio interface. The main window shows a data table with columns: RETAILER, VOLUME, DSP, and PRICE. The data is filtered to show only 'ALBANY, NY - PRICE CHOPPER' observations. The console shows the following commands and output:

```
> cheese_albany=cheese[cheese$RETAILER=="ALBANY, NY - PRICE CHOPPER",]  
> View(cheese_albany)  
>
```

The scatter plot on the right shows the relationship between log(Price) on the x-axis and log(Volume) on the y-axis. The data points are clustered around log(Price) = 1.0 and log(Volume) = 8.0. The NPTEL logo is visible in the top right corner of the RStudio window.

So, these are the values of all the 61 observations all belongs to ALBANY PRICE CHOPPER, ok.

(Refer Slide Time: 09:53)

The screenshot displays the RStudio interface with the following components:

- Source Editor:** Contains R code for data visualization:

```
7 ## EDA for price to volume
9 plot(cheese$PRICE,cheese$VOLUME,pch=20,col="skyblue"
10      ,xlab="Price",ylab = "Volume")
12 plot(log(cheese$PRICE),log(cheese$VOLUME),pch=20,col="skyblue"
13      ,xlab="log(Price)",ylab = "log(Volume)")
14 cheese_albany=cheese[cheese$RETAILER=="ALBANY,NY - PRICE CHOPPER",]
15 poi
16 points
17 points.default
18 poisson
19 poisson.test
```
- Environment:** Shows data objects: 'cheese' (5555 obs. of 4 variables) and 'cheese\_albany' (61 obs. of 4 variables).
- Values:** Lists 'nreg' as 88L and 'retailer' as 'chr [1:88] "ALBANY,NY - PRI...".
- Console:** Displays a list of 88 retailers, including "SYRACUSE - P & C FOOD MARKE", "TAMPA/ST. PETE - KASH N KARRY", and "WICHITA - DILLON COMPANIES". The final command is `> View(cheese_albany)`.
- Plots:** A scatter plot titled 'log(Volume)' vs 'log(Price)' showing a positive correlation between the two variables. The x-axis ranges from 0.4 to 1.4, and the y-axis ranges from 6 to 12.
- NPTEL Logo:** Located in the top right corner.
- Video Feed:** A small inset in the bottom right shows a man speaking into a microphone.

And now what I am going to do.



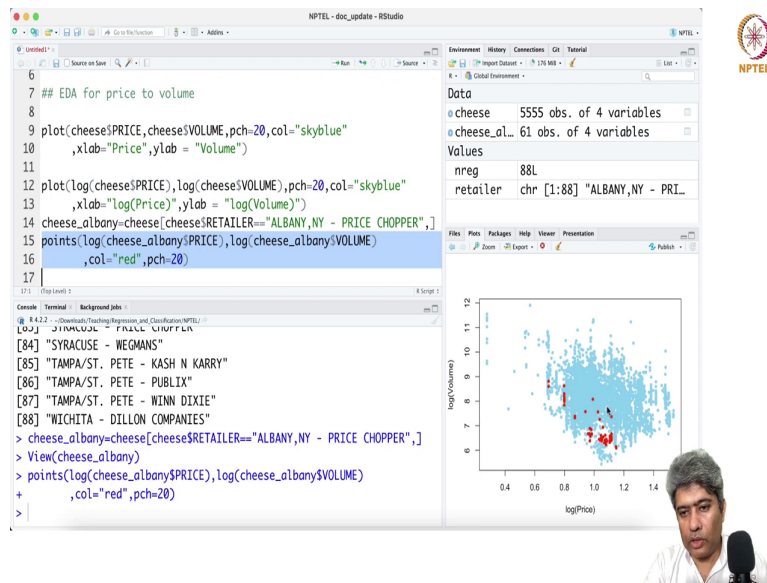
(Refer Slide Time: 10:07)

The screenshot shows the RStudio interface with the following components:

- Source Editor:** Contains R code for data analysis and plotting. Lines 12-13 show a scatter plot of log(Volume) vs log(Price) for all data points. Lines 14-16 show a subset of data for 'ALBANY, NY - PRICE CHOPPER'.
- Environment:** Shows the 'Data' environment with 'cheese' (5555 obs. of 4 variables) and 'cheese\_alb...' (61 obs. of 4 variables).
- Values:** Shows the structure of the 'cheese' data frame: nreg (88L) and retailer (chr [1:88] "ALBANY, NY - PRI...").
- Console:** Shows the execution of the R code, including the output of the 'view' function for the selected subset of data.
- Scatter Plot:** A plot of log(Volume) vs log(Price) showing a positive correlation between the two variables. The x-axis is labeled 'log(Price)' and the y-axis is labeled 'log(Volume)'. The data points are colored cyan.
- NPTEL Logo:** Located in the top right corner of the RStudio window.
- Speaker:** A small video feed of a person speaking is visible in the bottom right corner of the RStudio window.

I am going to point plot the points of only log cheese albany dollar PRICE comma log cheese albany dollar VOLUME and color equal to “red”.

(Refer Slide Time: 10:27)



The screenshot displays the RStudio interface. The script editor on the left contains the following R code:

```
6  
7 ## EDA for price to volume  
8  
9 plot(cheese$PRICE,cheese$VOLUME,pch=20,col="skyblue"  
10 ,xlab="Price",ylab = "Volume")  
11  
12 plot(log(cheese$PRICE),log(cheese$VOLUME),pch=20,col="skyblue"  
13 ,xlab="log(Price)",ylab = "log(Volume)")  
14 cheese_albany=cheese[cheese$RETAILER=="ALBANY,NY - PRICE CHOPPER",]  
15 points(log(cheese_albany$PRICE),log(cheese_albany$VOLUME)  
16 ,col="red",pch=20)  
17
```

The console on the left shows the execution of the code, listing the names of the retailers in the 'cheese\_albany' dataset:

```
## [84] "SYRACUSE - WEGMANS"  
## [85] "TAMPA/ST. PETE - KASH N KARRY"  
## [86] "TAMPA/ST. PETE - PUBLIX"  
## [87] "TAMPA/ST. PETE - WINN DIXIE"  
## [88] "WICHITA - DILLON COMPANIES"  
> cheese_albany=cheese[cheese$RETAILER=="ALBANY,NY - PRICE CHOPPER",]  
> View(cheese_albany)  
> points(log(cheese_albany$PRICE),log(cheese_albany$VOLUME)  
+ ,col="red",pch=20)  
>
```

The environment pane on the right shows the 'Data' tab with the following information:

Object	Class	Attributes
cheese	data.frame	5555 obs. of 4 variables
cheese_albany	data.frame	61 obs. of 4 variables
nreg	numeric	88L
retailer	character	[1:88] "ALBANY,NY - PRI..."

The plot pane on the right shows a scatter plot of log(Volume) versus log(Price). The plot contains a large cloud of light blue points (pch=20) representing the entire 'cheese' dataset. A smaller cluster of red points (pch=20) is overlaid, representing the 'cheese\_albany' dataset. The x-axis is labeled 'log(Price)' and ranges from 0.4 to 1.4. The y-axis is labeled 'log(Volume)' and ranges from 6 to 12.

The NPTEL logo is visible in the top right corner of the RStudio window.

And pch equal to 20. So, you can see the price is these are the points, which all belongs to price chopper. So, chopper Albany price chopper in this range. So, you can give a different for different price at different price level the number of sales behaves like this, ok.

(Refer Slide Time: 10:58)

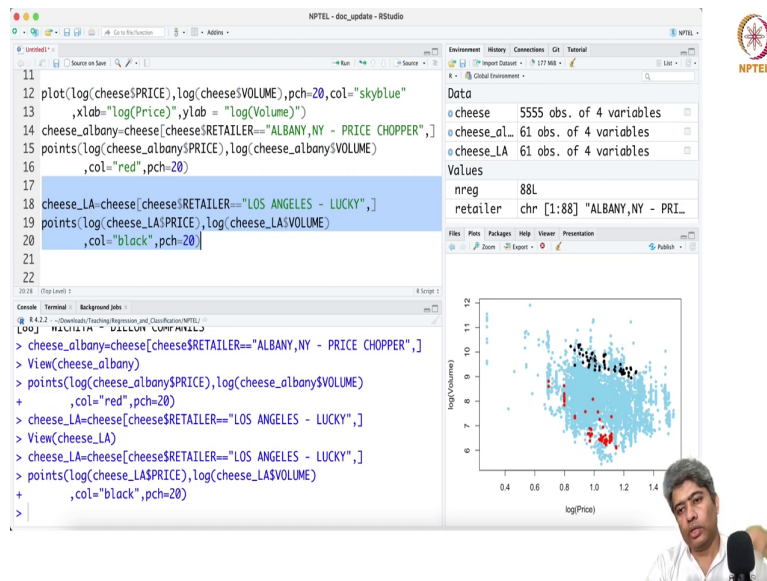
The screenshot displays an RStudio interface with the following components:

- Source Editor:** Contains R code for plotting log(price) vs log(volume) for different cheese retailers. The code includes:

```
10 ,xlab="Price",ylab = "Volume")
11
12 plot(log(cheese$PRICE),log(cheese$VOLUME),pch=20,col="skyblue"
13 ,xlab="log(Price)",ylab = "log(Volume)")
14 cheese_albany=cheese[cheese$RETAILER=="ALBANY,NY - PRICE CHOPPER",]
15 points(log(cheese_albany$PRICE),log(cheese_albany$VOLUME)
16 ,col="red",pch=20)
17
18 cheese_LA=cheese[cheese$RETAILER=="LOS ANGELES - LUCKY",]
19
20 points(log(cheese_albany$PRICE),log(cheese_albany$VOLUME)
21 ,col="red",pch=20)
```
- Environment:** Shows data objects: 'cheese' (5555 obs. of 4 variables) and 'cheese\_al...' (61 obs. of 4 variables).
- Values:** Shows values for 'nreg' (88L) and 'retailer' (chr [1:88] "ALBANY,NY - PRI...").
- Console:** Displays the output of the R code, listing retailers such as "JACKSONVILLE, FL - PUBLIX", "LOS ANGELES - LUCKY", "LOS ANGELES - RALPHS", etc.
- Plot:** A scatter plot showing log(Volume) on the y-axis (ranging from 6 to 12) and log(Price) on the x-axis (ranging from 0.4 to 1.4). The plot shows a positive correlation between log(price) and log(volume) for various retailers, with 'LOS ANGELES - LUCKY' highlighted in red.
- Video Inset:** A small video inset in the bottom right corner shows a man speaking into a microphone.

Now, I am going to curve compare it with “LOS ANGELES – LUCKY” let me see same thing, but instead of I am going to do with “LOS ANGELES - LUCKY” ok cheese LA.

(Refer Slide Time: 11:31)



Now, if you see it also has 61 observations VOLUME and these are the PRICE and VOLUMES. Now, if I just do instead of cheese LA, LA and instead of “red” I will use “black” right. So, all these are just different cluster altogether. So, now, if I fit a model here based on using all the models then the line will pass through somewhere here.

And if I use this line right because if I use all these points to fit a model that line will pass through the middle of this and that line will neither fit price chopper in Albany nor it will fit Los Angeles Lucky. So, what we want? We want a particular for a for each retailer we want to fit a model for each model or we want to fit a separate model. So, that is the idea. So, what we will do so; that means, there are how many retailer?

(Refer Slide Time: 12:54)

The screenshot shows an RStudio session with the following code in the script editor:

```
1 library(bayesm)
2 data(cheese)
3
4 retailer = levels(cheese$RETAILER)
5 nreg = length(retailer)
6
7 ## EDA for price to volume
8
9 plot(cheese$PRICE,cheese$VOLUME,pch=20,col="skyblue"
10      ,xlab="Price",ylab = "Volume")
11
12 plot(log(cheese$PRICE).log(cheese$VOLUME).pch=20,col="skvblue"
```

The console shows the execution of the following commands:

```
> cheese_albany=cheese[cheese$RETAILER=="ALBANY, NY - PRICE CHOPPER",]
> View(cheese_albany)
> points(log(cheese_albany$PRICE),log(cheese_albany$VOLUME)
+        ,col="red",pch=20)
> cheese_LA=cheese[cheese$RETAILER=="LOS ANGELES - LUCKY",]
> View(cheese_LA)
> cheese_LA=cheese[cheese$RETAILER=="LOS ANGELES - LUCKY",]
> points(log(cheese_LA$PRICE),log(cheese_LA$VOLUME)
+        ,col="black",pch=20)
>
```



The environment pane shows the following data objects:

- cheese: 5555 obs. of 4 variables
- cheese\_albany: 61 obs. of 4 variables
- cheese\_LA: 61 obs. of 4 variables

The values pane shows:

```
nreg      88L
retailer  chr [1:88] "ALBANY, NY - PRI..."
```

The plot pane shows a scatter plot of log(Volume) vs log(Price) with points colored by retailer. The x-axis is labeled log(Price) and ranges from 0.4 to 1.4. The y-axis is labeled log(Volume) and ranges from 6 to 12. The plot shows a positive correlation between log(Price) and log(Volume). Points are colored by retailer: skyblue (main cluster), red (Albany, NY - Price Chopper), and black (Los Angeles - Lucky).



88 retailer, we want to fit 88 regression model.

(Refer Slide Time: 13:00)

The screenshot displays the RStudio interface. The script editor contains the following R code:

```
15 points(log(cheese_albany$PRICE), log(cheese_albany$VOLUME))
16 ,col="red",pch=20)
17
18 cheese_LA=cheese[cheese$RETAILER=="LOS ANGELES - LUCKY",]
19 points(log(cheese_LA$PRICE), log(cheese_LA$VOLUME))
20 ,col="black",pch=20)
21
22 ## We will fit 88 separate regression model for each retailer separately
23
24 beta.ls = matrix(NA, nrow=nreg, ncol = 2)
25
26
```

The console shows the execution of the code for the 'LOS ANGELES - LUCKY' retailer:

```
> cheese_LA=cheese[cheese$RETAILER=="LOS ANGELES - LUCKY",]
> View(cheese_LA)
> cheese_LA=cheese[cheese$RETAILER=="LOS ANGELES - LUCKY",]
> points(log(cheese_LA$PRICE), log(cheese_LA$VOLUME))
+ ,col="black",pch=20)
> nreg
[1] 88
> beta.ls = matrix(NA, nrow=nreg, ncol = 2)
> View(beta.ls)
>
```

The Environment pane shows the following data objects:

- beta.ls: logi [1:88, 1:2] NA
- cheese: 5555 obs. of 4 variab...
- cheese\_: 61 obs. of 4 variab...
- cheese\_: 61 obs. of 4 variab...

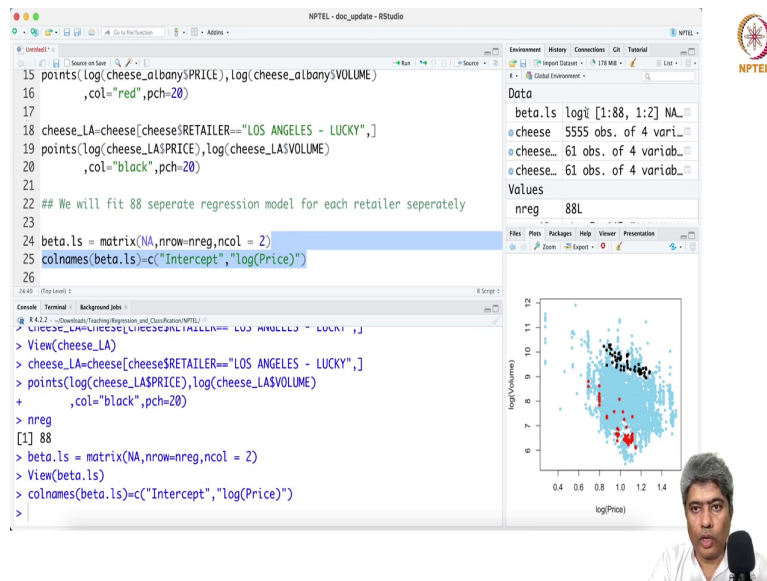
The Values pane shows:

```
nreg 88L
```

The Plots pane displays a scatter plot of log(Volume) versus log(Price). The plot shows a positive correlation between log(Price) and log(Volume). The data points are colored by retailer, with blue points representing the 'LOS ANGELES - LUCKY' retailer and red points representing other retailers. The x-axis (log(Price)) ranges from 0.4 to 1.4, and the y-axis (log(Volume)) ranges from 6 to 12.

So, let us fit 88 regression model, we will fit we will fit 88 separate regression model for each retailer separately, correct. Now, that is my goal that is my goal. So, first what I am going to do beta dot ls equal to matrix NA, nrow equal to nreg and ncol equal to 2. So, this is my beta I am just creating a place holder this is just a place holder.

(Refer Slide Time: 14:04)



The screenshot displays the RStudio interface with the following R code in the editor:

```
15 points(log(cheese_albany$PRICE), log(cheese_albany$VOLUME))
16     ,col="red",pch=20)
17
18 cheese_LA=cheese[cheese$RETAILER=="LOS ANGELES - LUCKY",]
19 points(log(cheese_LA$PRICE), log(cheese_LA$VOLUME))
20     ,col="black",pch=20)
21
22 ## We will fit 88 separate regression model for each retailer separately
23
24 beta.ls = matrix(NA,nrow=nreg,ncol = 2)
25 colnames(beta.ls)=c("Intercept", "log(Price)")
26
```


The console shows the execution of the code:

```
> View(cheese_LA)
> cheese_LA=cheese[cheese$RETAILER=="LOS ANGELES - LUCKY",]
> points(log(cheese_LA$PRICE), log(cheese_LA$VOLUME))
+
+     ,col="black",pch=20)
> nreg
[1] 88
> beta.ls = matrix(NA,nrow=nreg,ncol = 2)
> View(beta.ls)
> colnames(beta.ls)=c("Intercept", "log(Price)")
>
```

The Environment pane shows the following data:

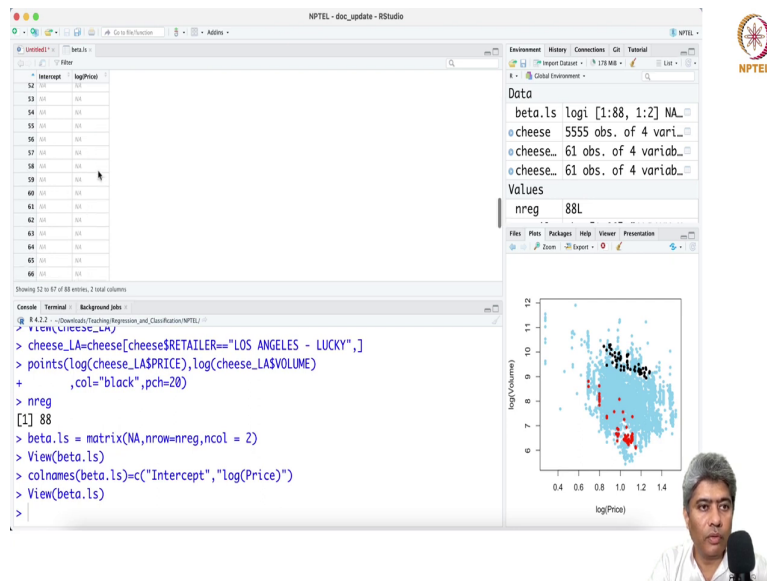
```
beta.ls logi [1:88, 1:2] NA_
cheese 5555 obs. of 4 vari_
cheese_ 61 obs. of 4 variab_
cheese_ 61 obs. of 4 variab_
Values
nreg 88L
```

A scatter plot of log(Volume) vs log(Price) is shown in the bottom right, with points colored by retailer (red and black).



And colnames will be c first is “Intercept”, ok and the second is “log Price” correct.

(Refer Slide Time: 14:23)



The screenshot displays the RStudio interface. The top-left pane shows a data table with columns 'Intercept' and 'logPrice'. The bottom-left pane shows the following R code in the console:

```
view(cheese_CA)
> cheese_LA=cheese[cheese$RETAILER=="LOS ANGELES - LUCKY",]
> points(log(cheese_LASPRICE), log(cheese_LASVOLUME)
+ ,col="black",pch=20)
> nreg
[1] 88
> beta.ls = matrix(NA,nrow=nreg,ncol = 2)
> View(beta.ls)
> colnames(beta.ls)=c("Intercept", "log(Price)")
> View(beta.ls)
>
```

The top-right pane shows the environment with 'beta.ls' and 'logi' objects. The bottom-right pane shows a scatter plot of log(Volume) versus log(Price) with points colored by retailer.

So, what I am going to do these are the two things. So, intercept and the coefficient of log Price these two will be stored will calculate and store for all 88 retailers separately. So, I am going to fit 88 separate model for this. This is Los Angeles Lucky I will have a separate line this is Albany you know price chopper I will have a separate line. So, we will fit a separate line for each of them. So, what we are going to do now is for i in 1 is to nreg.



(Refer Slide Time: 15:24)

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

```
22 # We will fit 88 separate regression model for each retailer separately
23
24 beta.ls = matrix(NA, nrow=nreg, ncol = 2)
25 colnames(beta.ls)=c("Intercept", "log(Price)")
26
27 for(i in 1:nreg){
28   filter = cheese$RETAILER==retailer[i]
29   y = log(cheese$VOLUME[filter])
30   X = cbind(1, ## intercept or placeholder
31           log(cheese$PRICE[filter]))
32   beta.ls[i,]=lm(y~X-1)$coefficients
33 }
```



The console shows the execution of the code:

```
> colnames(beta.ls)=c("Intercept", "log(Price)")
> View(beta.ls)
> for(i in 1:nreg){
+   filter = cheese$RETAILER==retailer[i]
+   y = log(cheese$VOLUME[filter])
+   X = cbind(1, ## intercept or placeholder
+           log(cheese$PRICE[filter]))
+   beta.ls[i,]=lm(y~X-1)$coefficients
+ }
>
```

The Environment pane on the right shows the objects created:

- beta.ls: num [1:88, 1:2] 12...
- cheese: 5555 obs. of 4 vari...
- cheese\_: 61 obs. of 4 variab...
- cheese\_: 61 obs. of 4 variab...
- X: num [1:68, 1:2] 1 1...

The bottom right pane shows a scatter plot of log(Volume) vs log(Price) with data points colored by retailer.



Filter equal to cheese RETAILER retailer i and then what I am going to do y equal to log of cheese dollar VOLUME filter. And X equal to cbind 1 comma this is essentially intercept or placeholder. And then this is log cheese dollar PRICE filter, correct, and then beta dot ls i comma lm y tilde X minus 1 dollar coefficients.

(Refer Slide Time: 17:19)

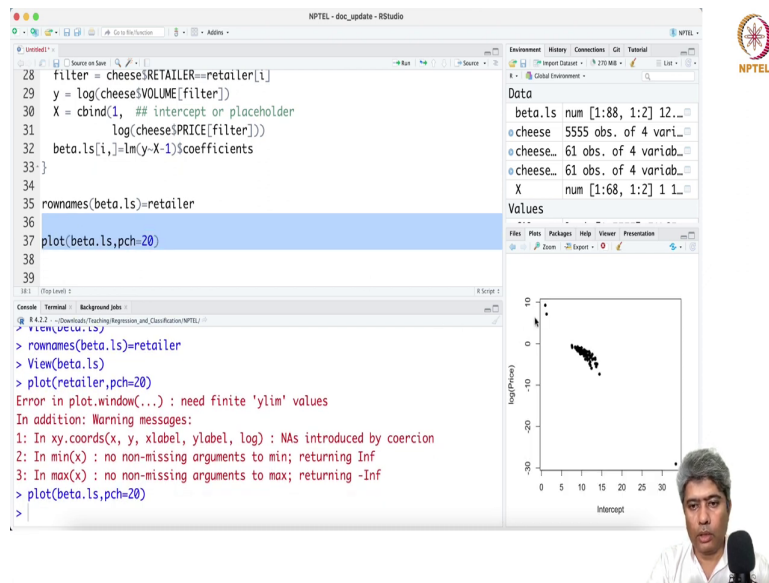
The screenshot displays the RStudio interface. The top-left pane shows a data frame with columns 'Intercept' and 'logPrice'. The top-right pane shows the environment with 'beta.ls' and 'Data'. The bottom-left pane shows the R console with the following code:

```
filter = cheese$RETAILER==retailer[i]
y = log(cheese$VOLUME[filter])
X = cbind(1, # intercept or placeholder
          log(cheese$PRICE[filter]))
beta.ls[i,]=lm(y~X-1)$coefficients
}
> View(beta.ls)
> rownames(beta.ls)=retailer
> View(beta.ls)
>
```

The bottom-right pane shows a scatter plot of log(Volume) versus log(Price). The plot shows a positive correlation between the two variables, with data points colored in shades of blue and red. A small inset image of a man speaking into a microphone is visible in the bottom right corner of the RStudio window.

Now, if you just say beta dot ls these are the coefficient values of intercept and log price coefficient of log price. So, rownames equal to beta dot ls equals to you can put retailer let me put retailer. So, these are the names that we have for each guys you have the retailer names.

(Refer Slide Time: 17:55)



The screenshot shows the RStudio interface with the following components:

- Source Editor:** Contains R code for fitting a linear regression model and plotting the intercept coefficient.
- Environment:** Shows the objects created in the environment, including the fitted model and the intercept coefficient.
- Console:** Displays the execution of the code and the resulting error and warning messages.
- Plot Window:** Shows a scatter plot of the intercept coefficient.

```
28 filter = cheese$RETAILER==retailer[i]
29 y = log(cheese$VOLUME[filter])
30 X = cbind(1, ## intercept or placeholder
31           log(cheese$PRICE[filter]))
32 beta.ls[i,]=lm(y-X-1)$coefficients
33 }
34
35 rownames(beta.ls)=retailer
36
37 plot(beta.ls,pch=20)
38
39
```

**Environment:**

Object	Class	Attributes
beta.ls	num	[1:68, 1:2] 12...
cheese	data.frame	5555 obs. of 4 vari...
cheese..	data.frame	61 obs. of 4 variab...
cheese..	data.frame	61 obs. of 4 variab...
X	num	[1:68, 1:2] 1 1...

**Console:**

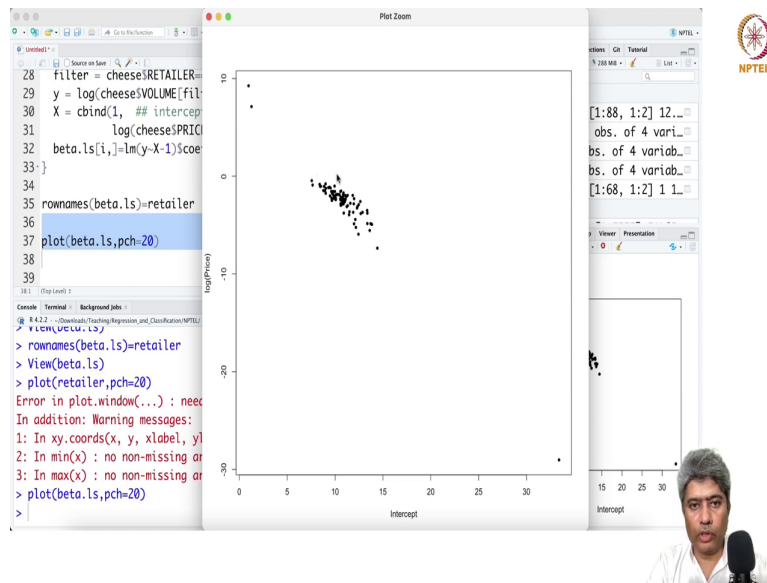
```
> rownames(beta.ls)=retailer
> View(beta.ls)
> plot(retailer,pch=20)
Error in plot.window(...): need finite 'ylim' values
In addition: Warning messages:
1: In xy.coords(x, y, xlabel, ylabel, log) : NAs introduced by coercion
2: In min(x) : no non-missing arguments to min; returning Inf
3: In max(x) : no non-missing arguments to max; returning -Inf
> plot(beta.ls,pch=20)
>
```

**Plot Window:**

A scatter plot showing the intercept coefficient. The x-axis is labeled 'Intercept' and ranges from 0 to 30. The y-axis is labeled 'log(Price)' and ranges from -30 to 10. The plot shows a positive correlation between the intercept and the log of the price.

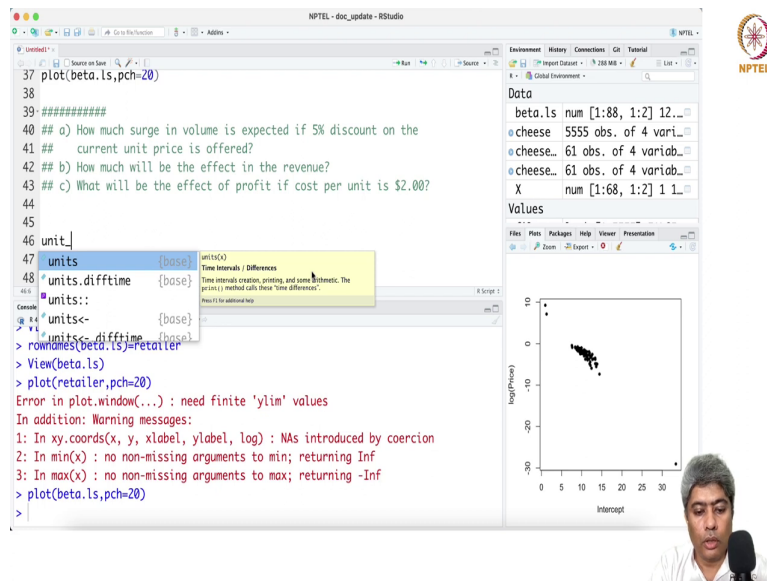
Now, if I just put plot beta dot ls pch equal to 20 sorry this is beta dot ls, yeah.

(Refer Slide Time: 18:16)



So, if you just so, these are the typical coefficients there are two outlier and there is one outlier here. There are 3 two different outliers. So, we have to do a little bit of outlier analysis later.

(Refer Slide Time: 18:34)



The screenshot shows an RStudio window titled "NPTEL - doc\_update - RStudio". The console displays the following code and output:

```
37 plot(beta.ls,pch=20)
38
39 #####
40 ## a) How much surge in volume is expected if 5% discount on the
41 ## current unit price is offered?
42 ## b) How much will be the effect in the revenue?
43 ## c) What will be the effect of profit if cost per unit is $2.00?
44
45
46 unit_|
47 units {base} units()
48 # units.difftime {base} Time intervals / Differences
49 # units:: units:: Time intervals creation, printing, and some Rithmetic. The
50 # units:: units:: print() method calls these "time differences".
51 # units<- units<- {base}
52 # units<- difftime {base}
53 > rownames(beta.ls)=retailer
54 > View(beta.ls)
55 > plot(retailer,pch=20)
Error in plot.window(...): need finite 'ylim' values
In addition: Warning messages:
1: In xy.coords(x, y, xlabel, ylabel, log) : NAs introduced by coercion
2: In min(x) : no non-missing arguments to min; returning Inf
3: In max(x) : no non-missing arguments to max; returning -Inf
> plot(beta.ls,pch=20)
>
```

The Environment pane on the right shows the following data objects:

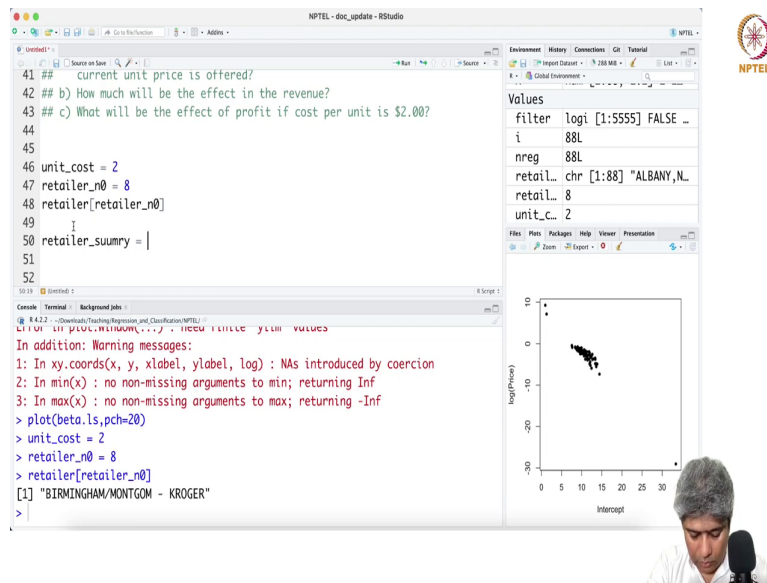
Object	Class	Dimensions
beta.ls	num	[1:88, 1:2] 12...
cheese	5555 obs. of 4 vari...	
cheese	61 obs. of 4 variab...	
X	num	[1:68, 1:2] 1 1...

The plot shows a scatter plot of log(Phone) versus Intercept. The x-axis is labeled "Intercept" and ranges from 0 to 30. The y-axis is labeled "log(Phone)" and ranges from -30 to 10. The data points are scattered around a central point, with a slight downward trend.

So, now what I am going to do? I am going to try to answer following questions, ok. How much; first question how much surge in volume surge in volume is expected if 5 percent discount on the current unit price on the current unit price is offered? Ok, so, this is the if I give 5 percent discount on the current unit price then how much surge I would in terms of volume sales I can expect.

Then how much will be the effect in the revenue how much will be the effect in the revenue? And what will be the effect of cost profit if cost per unit is dollar 2? So, these are the three questions we are going to discuss, ok. So, the first thing is unit cost is 2, right, unit cost is 2.

(Refer Slide Time: 20:59)



The screenshot shows an RStudio window titled "NPTEL - doc\_update - RStudio". The editor pane contains the following R code:

```
41 ## current unit price is offered?  
42 ## b) How much will be the effect in the revenue?  
43 ## c) What will be the effect of profit if cost per unit is $2.00?  
44  
45  
46 unit_cost = 2  
47 retailer_n0 = 8  
48 retailer[retailer_n0]  
49  
50 retailer_summary = |  
51  
52
```

The console pane shows the following output and warnings:

```
## 3.4.2.2 - Download: Training Regression and Classification (NPTEL)  
Error in plot.minimum(...): new title 'ymin values'  
In addition: Warning messages:  
1: In xy.coords(x, y, xlabel, ylabel, log) : NAs introduced by coercion  
2: In min(x) : no non-missing arguments to min; returning Inf  
3: In max(x) : no non-missing arguments to max; returning -Inf  
> plot(beta.ls, pch=20)  
> unit_cost = 2  
> retailer_n0 = 8  
> retailer[retailer_n0]  
[1] "BIRMINGHAM/MONTGOM - KROGER"  
>
```

The Environment pane shows the following values:

```
filter logi [1:5555] FALSE _  
i 88L  
nreg 88L  
retail_ chr [1:88] "ALBANY,NL  
retail_ 8  
unit_c_ 2
```

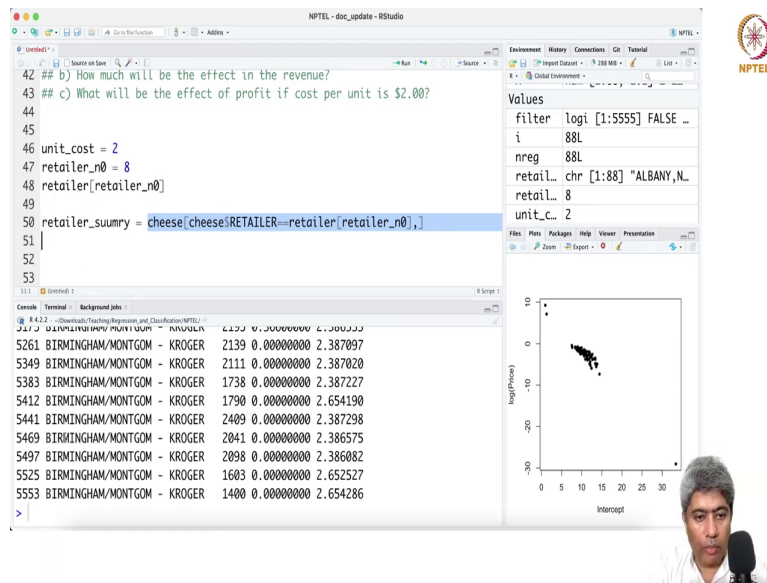
The plot pane shows a scatter plot of log(Price) versus Intercept. The plot shows a positive correlation between the intercept and the log price. The x-axis is labeled "Intercept" and ranges from 0 to 30. The y-axis is labeled "log(Price)" and ranges from -20 to 10. The plot shows a cluster of points around (10, 0) and a few points at higher intercept values.

The NPTEL logo is visible in the top right corner of the RStudio window.

And let us take a retailer number is some number maybe 8, ok. Now, retailer number 8 who is this guy, ok? Birmingham Kroger, Kroger is a big chain and in Birmingham Montgomery district or county Kroger retailer this is the retailer they are talking about.

So, let us have a look into the retailer summary typical what is the summary statistics of this particular retailer ok. So, the first thing I am going to do is I am going to take this guys whoever this guy is retailer with retailer number, right.

(Refer Slide Time: 22:27)



The screenshot displays the RStudio interface. The script editor contains the following R code:

```
42 ## b) How much will be the effect in the revenue?  
43 ## c) What will be the effect of profit if cost per unit is $2.00?  
44  
45  
46 unit_cost = 2  
47 retailer_n0 = 8  
48 retailer[retailer_n0]  
49  
50 retailer_summy = cheese[cheese:RETAILER==retailer[retailer_n0],]  
51  
52  
53
```

The console shows the following output:

```
5261 BIRMINGHAM/MONTGOM - KROGER 2139 0.00000000 2.387097  
5349 BIRMINGHAM/MONTGOM - KROGER 2111 0.00000000 2.387020  
5383 BIRMINGHAM/MONTGOM - KROGER 1738 0.00000000 2.387227  
5412 BIRMINGHAM/MONTGOM - KROGER 1790 0.00000000 2.654190  
5441 BIRMINGHAM/MONTGOM - KROGER 2409 0.00000000 2.387298  
5469 BIRMINGHAM/MONTGOM - KROGER 2041 0.00000000 2.386575  
5497 BIRMINGHAM/MONTGOM - KROGER 2098 0.00000000 2.386082  
5525 BIRMINGHAM/MONTGOM - KROGER 1603 0.00000000 2.652527  
5553 BIRMINGHAM/MONTGOM - KROGER 1400 0.00000000 2.654286
```

The Environment pane shows the following values:

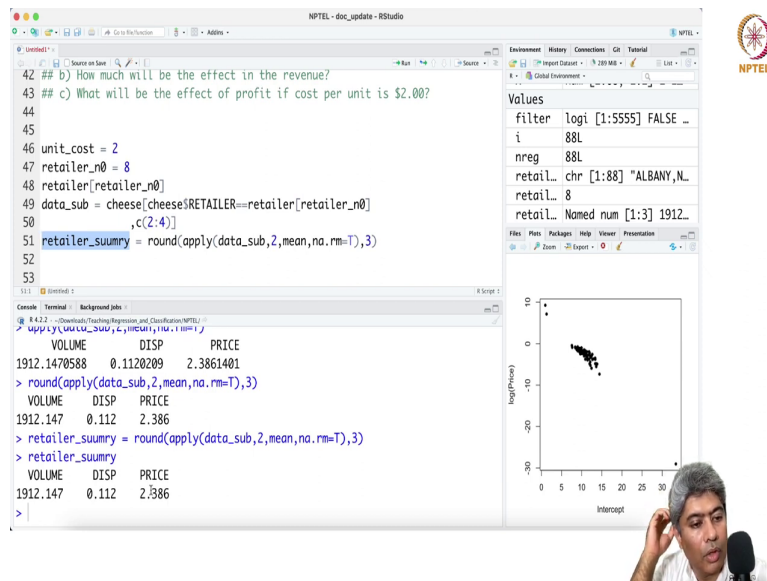
Variable	Value
filter	logi [1:5555] FALSE
i	88L
nreg	88L
retail_chr	[1:88] "ALBANY,NL"
retail_n	8
unit_c	2

The plot pane shows a scatter plot of log(Price) versus Intercept. The y-axis ranges from -30 to 10, and the x-axis ranges from 0 to 30. The data points are clustered around a positive correlation.

The NPTEL logo is visible in the top right corner of the RStudio window.

And in that particularly I am interested in 2 and 4 the volume and price. Particularly I am interested in 2 is to 4, volume, display and price; why not I apply? So, this is the so, first let me just take a data sub if I just take data sub data set. So, this is the data set.

(Refer Slide Time: 23:25)



The screenshot shows an RStudio session with the following code in the editor:

```
42 ## b) How much will be the effect in the revenue?  
43 ## c) What will be the effect of profit if cost per unit is $2.00?  
44  
45  
46 unit_cost = 2  
47 retailer_n0 = 8  
48 retailer[retailer_n0]  
49 data_sub = cheese[cheese$RETAILER==retailer[retailer_n0]  
50 ,c(2:4)]  
51 retailer_suumry = round(apply(data_sub,2,mean,na.rm=T),3)  
52  
53
```

The console shows the execution of the code:

```
## 8.4.2.2 - Quantile Training Regression and Classification (NPTEL)  
> apply(data_sub, 2, mean, na.rm=T)  
  VOLUME    DISP    PRICE  
1912.1470588 0.1120209 2.3861401  
> round(apply(data_sub, 2, mean, na.rm=T), 3)  
  VOLUME    DISP    PRICE  
1912.147  0.112  2.386  
> retailer_suumry = round(apply(data_sub, 2, mean, na.rm=T), 3)  
> retailer_suumry  
  VOLUME    DISP    PRICE  
1912.147  0.112  2.386  
>
```

The Environment pane shows the following values:

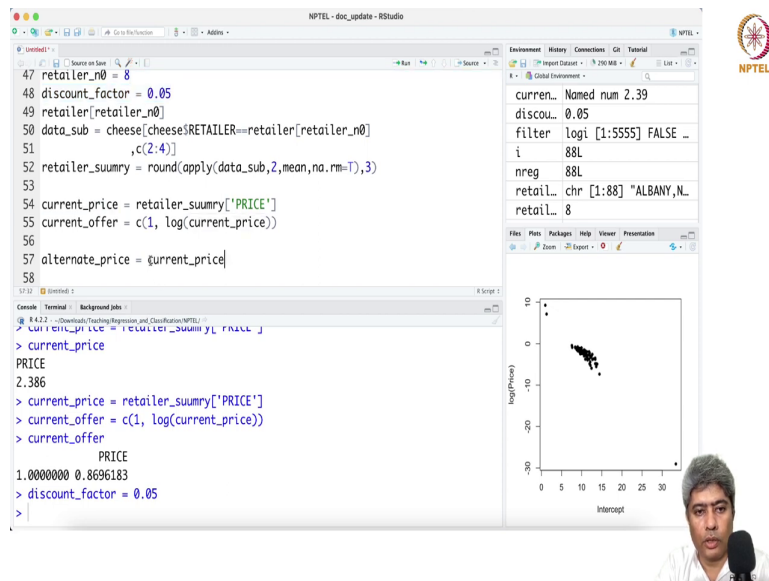
filter	logi	[1:5555]	FALSE	_
i	88L			
nreg	88L			
retail_	chr	[1:88]	"ALBANY,N	
retail_		8		
retail_	Named num	[1:3]	1912_	

The plot shows a scatter plot of log(Price) versus Intercept. The y-axis ranges from -20 to 10, and the x-axis ranges from 0 to 30. The data points are clustered around a log(Price) of 0 for an Intercept between 10 and 20.

And data set comma data sub comma 2 comma mean na dot rm equal to true, ok. And then if I just round up to 3 decimal places. So, that gives me these values. So, that means, this particular retailer is selling at 2.386 on an average the price that they are selling and volume is they selling about 1900; 1900 1912 many units per week.



(Refer Slide Time: 24:31)



The screenshot displays the RStudio interface. The main editor window contains the following R code:

```
47 retailer_n0 = 8
48 discount_factor = 0.05
49 retailer[retailer_n0]
50 data_sub = cheese[cheese$RETAILER==retailer[retailer_n0]
51 ,c(2:4)]
52 retailer_summy = round(apply(data_sub,2,mean,na.rm=T),3)
53
54 current_price = retailer_summy['PRICE']
55 current_offer = c(1, log(current_price))
56
57 alternate_price = current_price|
58
```

The console window shows the execution of the code:

```
> current_price = retailer_summy['PRICE']
> current_price
PRICE
2.386
> current_price = retailer_summy['PRICE']
> current_offer = c(1, log(current_price))
> current_offer
PRICE
1.0000000 0.8696183
> discount_factor = 0.05
>
```

The Environment pane on the right shows the following variables:

current_	Named num	2.39
discount_		0.05
filter	logi [1:5555]	FALSE
i		88L
nreg		88L
retail_	chr [1:88]	"ALBANY,N
retail_		8

A scatter plot is visible in the bottom right corner, with the y-axis labeled 'log(Price)' and the x-axis labeled 'Intercept'. The plot shows a positive correlation between the intercept and the log of the price.

The NPTEL logo is visible in the top right corner of the RStudio window.

So, let me take the current price is this price the average price as the current price. So, this is suppose current price, ok. If this is the current price and then I will call say current offer equal to 1 comma log equal to log of current price. So, this is my current offer. Similarly, we have to offer make a alternate price, which is say discount we have to define a discount factor discount factor. So, maybe 0.05 5 percent discount I am giving.

(Refer Slide Time: 26:02)

The screenshot displays the RStudio interface with the following R code in the editor:

```
47 retailer_n0 = 8
48 discount_factor = 0.05
49 retailer[retailer_n0]
50 data_sub = cheese[cheese$RETAILER==retailer[retailer_n0]
51 ,c(2:4)]
52 retailer_summary = round(apply(data_sub,2,mean,na.rm=T),3)
53
54 current_price = retailer_summary['PRICE']
55 current_offer = c(1, log(current_price))
56
57 alternate_price = current_price*(1-discount_factor)
58 alternate_offer = c(1,alternate_price)
```

The console output shows the results of the calculations:

```
> current_price = retailer_summary['PRICE']
PRICE
2.2667
> current_price
PRICE
2.386
> alternate_price = current_price*(1-discount_factor)
> alternate_price
PRICE
2.2667
>
```

The environment pane on the right shows the values of the objects created:

Object	Value
altern.	Named num 2.27
current.	Named num [1:2] 1 0...
current.	Named num 2.39
discount.	0.05
filter	logi [1:5555] FALSE ...
i	88L

A scatter plot is visible in the bottom right corner, showing the relationship between the Intercept and log(Price).

So, whatever the current price times 1 minus whatever the discount factor. So, to so, the current price is 2.386 after giving say 5 percent discount alternate price is 2.267. Now, so, my alternate offer alternate offer is 1 comma alternate price, ok.

(Refer Slide Time: 26:53)

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

```
53  
54 current_price = retailer_summary['PRICE']  
55 current_offer = c(1, log(current_price))  
56  
57 alternate_price = current_price*(1-discount_factor)  
58 alternate_offer = c(1, log(alternate_price))  
59  
60 beta_hat = beta.ls[retailer[retailer_n0],]  
61  
62 volume_current = exp(current_offer**beta_hat)  
63  
64
```



The console output shows the results of the `beta.ls` function:

```
8.439343 -1.033322  
> beta_hat = beta.ls[retailer[retailer_n0],]  
> current_offer**beta_hat  
[1,] 7.540747  
> volume_current = exp(current_offer**beta_hat)  
> volume_current  
[1,] 1883.236  
>
```

The Environment pane on the right shows the following objects:

Object	Class	Attributes	Value
volume...	num	[1, 1]	1883
X	num	[1:68, 1:2]	1 1...
Values			
altern..	Named num	[1:2]	1 2...
altern..	Named num		2.27
beta_h..	Named num	[1:2]	8.44...
current..	Named num	[1:2]	1 0...

A scatter plot in the bottom right shows `log(Price)` on the y-axis (ranging from -30 to 10) versus `Intercept` on the x-axis (ranging from 0 to 30). The plot shows a positive linear relationship between the intercept and the log price.



So, the first thing now I have to do is beta hat equal to beta dash beta dot ls retailer equal to retailer number. So, if I just take so, this is my coefficient for this particular retailer, ok. Now, volume if for this current offer and alternate offer what would be the volume expected sales of the volume, that is, clearly current offer beta hat percentage star percentage beta hat.

And then you have to make expected because remember that we are fitting log volume versus log price. So, volume current sales, ok. So, this is 8 you are expected to make a sale of 1883.

(Refer Slide Time: 28:25)

The screenshot displays the RStudio interface with the following components:

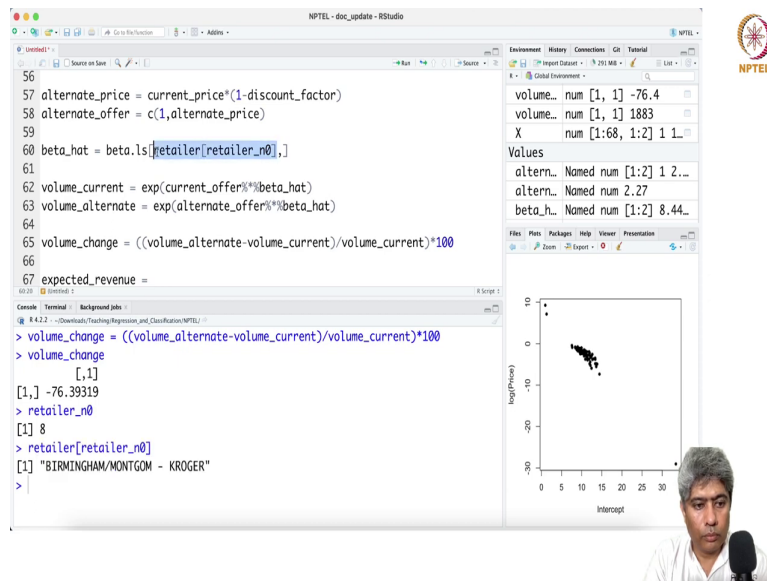
- Source Editor:** Lines 54-65 of R code:

```
54 current_price = retailer_summary['PRICE']
55 current_offer = c(1, log(current_price))
56
57 alternate_price = current_price*(1-discount_factor)
58 alternate_offer = c(1, alternate_price)
59
60 beta_hat = beta.ls[retailer[retailer_n0],]
61
62 volume_current = exp(current_offer%*%beta_hat)
63 volume_alternate = exp(alternate_offer%*%beta_hat)
64
65 (volume_alternate-volume_current)/volume_current*100
```
- Environment Pane:** Lists variables: volume.. num [1, 1] 445, volume.. num [1, 1] 1883, X num [1:68, 1:2] 1 1..
- Values Pane:** Lists values: altern.. Named num [1:2] 1 2..., altern.. Named num 2.27, beta\_h.. Named num [1:2] 8.44..
- Console:** Shows the execution of the code:

```
> volume_alternate
[1,]
[1,] 444.5721
> volume_current
[1,]
[1,] 1883.236
> (volume_alternate-volume_current)/volume_current*100
[1,]
[1,] -76.39319
>
```
- Plot:** A scatter plot with 'Intercept' on the x-axis (0 to 30) and 'log(price)' on the y-axis (-30 to 10). The data points show a negative correlation.
- NPTEL Logo:** Located in the top right corner.
- Speaker:** A small video inset of a man speaking is visible in the bottom right corner.

Now, similarly I can also make prediction for the alternate offer so, alternate offer, ok. So, this is my current offer this is my alternate offer, ok. So, alternate offer minus current offer divided by volume of current offer times 100, ok.

(Refer Slide Time: 29:48)



The screenshot shows the RStudio interface with the following components:

- Source Editor:** Contains R code for calculating alternate price, volume change, and expected revenue based on current price, discount factor, and beta parameters.
- Environment:** Lists variables: volume.. num [1, 1] -76.4, volume.. num [1, 1] 1883, X num [1:68, 1:2] 1 1., and Values for altern., altern., and beta\_h.
- Console:** Shows the execution of the volume\_change calculation, resulting in a vector of values for each retailer.
- Plots:** A scatter plot of log(Phone) vs Intercept, showing a negative correlation.

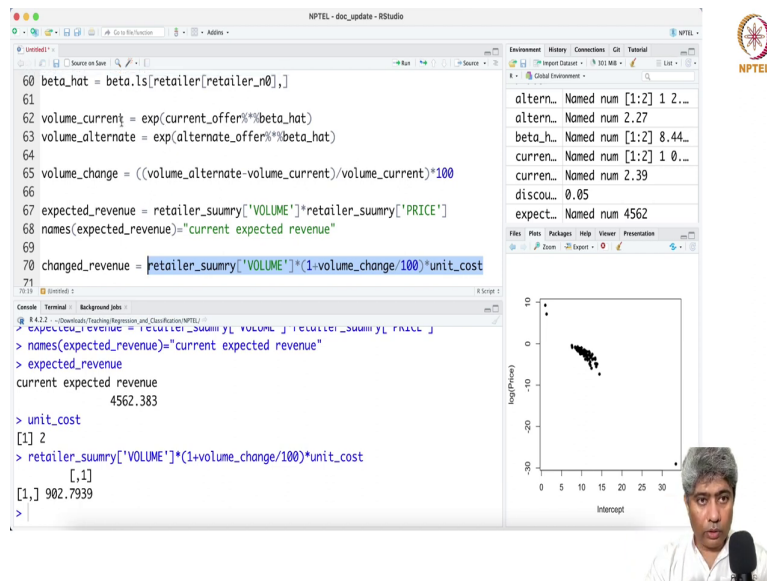
```
56
57 alternate_price = current_price*(1-discount_factor)
58 alternate_offer = c(1,alternate_price)
59
60 beta_hat = beta.ls[retailer[retailer_n0],]
61
62 volume_current = exp(current_offer**beta_hat)
63 volume_alternate = exp(alternate_offer**beta_hat)
64
65 volume_change = ((volume_alternate-volume_current)/volume_current)*100
66
67 expected_revenue =
```

```
> volume_change = ((volume_alternate-volume_current)/volume_current)*100
> volume_change
      [,1]
[1,] -76.39319
> retailer_n0
[1] 8
> retailer[retailer_n0]
[1] "BIRMINGHAM/MONTGOM - KROGER"
>
```

Variable	Value
volume..	num [1, 1] -76.4
volume..	num [1, 1] 1883
X	num [1:68, 1:2] 1 1.
altern.	Named num [1:2] 1 2...
altern.	Named num 2.27
beta_h.	Named num [1:2] 8.44...

Let me just do one thing. Percentage star percentage I think I am making 7.75, right. And 2.67 this seems very different alternate minus volume current, ok. So, let me just take this volume change. Maybe there will be a serious drop in the volume. I think that we can take it as at face value. And we can compute the expected revenue expected revenue, yeah, ok.

(Refer Slide Time: 31:26)



The screenshot displays the RStudio interface. The script editor contains the following R code:

```
60 beta_hat = beta.ls[retailer[retailer_n0],]
61
62 volume_current = exp(current_offer%*beta_hat)
63 volume_alterate = exp(alternate_offer%*beta_hat)
64
65 volume_change = ((volume_alterate-volume_current)/volume_current)*100
66
67 expected_revenue = retailer_summy["VOLUME"]*retailer_summy["PRICE"]
68 names(expected_revenue)="current expected revenue"
69
70 changed_revenue = retailer_summy["VOLUME"]*(1+volume_change/100)*unit_cost
71
```

The console shows the following output:

```
> expected_revenue = retailer_summy["VOLUME"]*retailer_summy["PRICE"]
> names(expected_revenue)="current expected revenue"
> expected_revenue
current expected revenue
4562.383
> unit_cost
[1] 2
> retailer_summy["VOLUME"]*(1+volume_change/100)*unit_cost
      [,1]
[1,] 902.7939
>
```

The Environment pane on the right lists several objects:

altern_	Named num [1:2]	1 2...
altern_	Named num	2.27
beta_h_	Named num [1:2]	8.44...
current_	Named num [1:2]	1 0...
current_	Named num	2.39
discou_		0.05
expect_	Named num	4562

A scatter plot titled "log(Phone)" vs "Intercept" is visible in the bottom right corner, showing a positive correlation between the intercept and the log of phone volume.

Expected revenue is whatever be the retailer summary and that is the “VOLUME” V o l u m e Times retailer summary of the PRICE, right. So, this is the volume that you expect, ok. So, we can call it current expected revenue, ok, ok. Now, what we will do is we will check what would be the changed revenue. The changed revenue will be first we have to take the this times 1 plus whatever the volume change divided by 100 times unit cost right. That will be the sorry that will be not unit cost that will be alternate price.

(Refer Slide Time: 33:33)

The screenshot shows an RStudio session with the following code and output:

```
64 volume_change = ((volume_alternate - volume_current) / volume_current) * 100
65
66 expected_revenue = retailer_summary["VOLUME"] * retailer_summary["PRICE"]
67 res(expected_revenue) = "current expected revenue"
68
69 updated_revenue = retailer_summary["VOLUME"] * (1 + volume_change / 100) * alternate_price
70 res(changed_revenue) = "updated expected revenue"
71
72 total_cost = retailer_summary["VOLUME"] * (1 + volume_change / 100) * unit_cost
73
74
75
76
```

Environment pane:

- change.. num [1, 1] 1023
- cheese.. 5555 obs. of 4 variab...
- cheese.. 61 obs. of 4 variab...
- cheese.. 61 obs. of 4 variab...
- data\_s... 68 obs. of 3 variab...
- volume.. num [1, 1] 445
- volume.. num [1, 1] -76.4

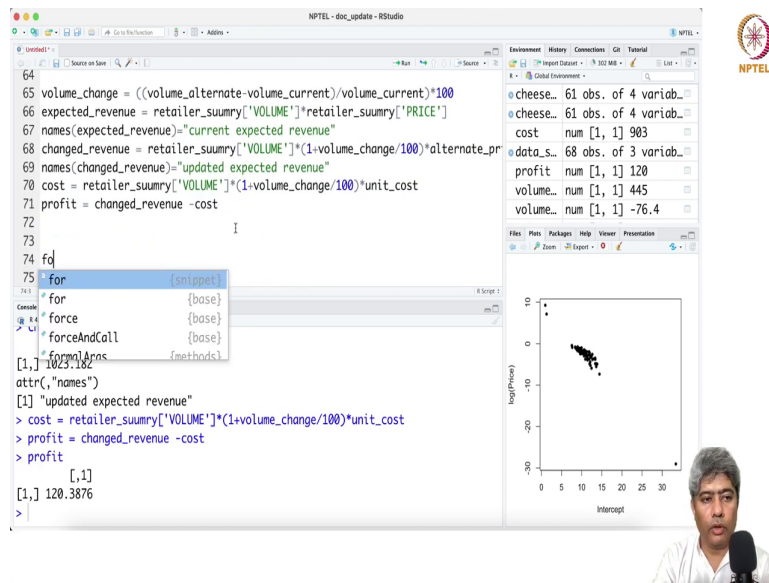
Console output:

```
> changed_revenue
[1,] 1023.182
> names(changed_revenue) = "updated expected revenue"
> changed_revenue
[1,] 1023.182
attr(,"names")
[1] "updated expected revenue"
>
```

A scatter plot titled 'log(price)' vs 'Intercept' is visible in the bottom right corner of the RStudio window.

That is the X that will be the so, the expected revenue is 4562 and the change revenue is sorry 1000 something 1023 right. So, this is sort of a names of changed revenue is “updated expected revenue”, ok. Now, total cost will be cost will be whatever the “VOLUME” times this changed percentage times unit cost. So, this will be the total cost.

(Refer Slide Time: 35:05)



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

```
64
65 volume_change = ((volume_alternate-volume_current)/volume_current)*100
66 expected_revenue = retailer_suumry["VOLUME"]*retailer_suumry["PRICE"]
67 names(expected_revenue)="current expected revenue"
68 changed_revenue = retailer_suumry["VOLUME"]*(1+volume_change/100)*alternate_pr
69 names(changed_revenue)="updated expected revenue"
70 cost = retailer_suumry["VOLUME"]*(1+volume_change/100)*unit_cost
71 profit = changed_revenue -cost
72
73
74 fo
75 for {snippet}
76 for {base}
77 force {base}
78 forceAndCall {base}
79 forceAndCall {base}
80 attr(,"names")
81 [1] "updated expected revenue"
82 > cost = retailer_suumry["VOLUME"]*(1+volume_change/100)*unit_cost
83 > profit = changed_revenue -cost
84 > profit
85
86 [,1]
87 [1,] 120.3876
88 >
```

The Environment pane on the right shows the following objects:

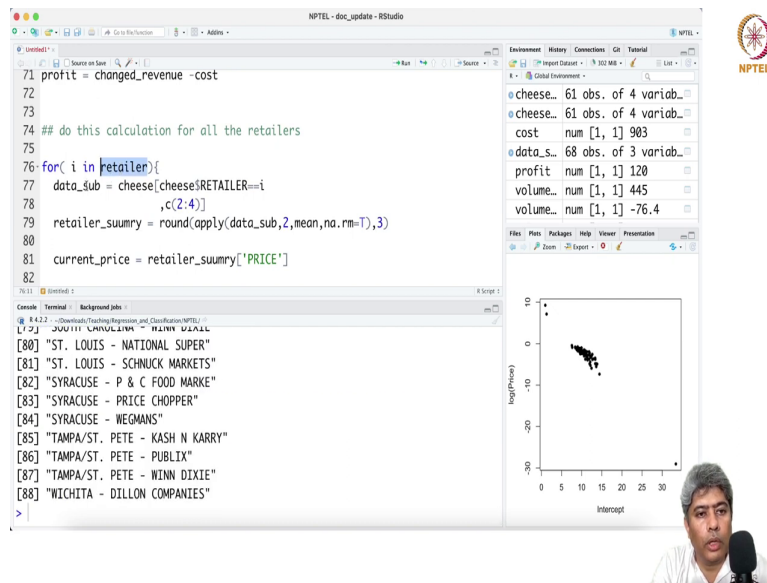
- cheese.. 61 obs. of 4 variab...
- cheese.. 61 obs. of 4 variab...
- cost num [1, 1] 903
- data\_s.. 68 obs. of 3 variab...
- profit num [1, 1] 120
- volume.. num [1, 1] 445
- volume.. num [1, 1] -76.4

The console shows the execution of the code, resulting in the profit value of 120.3876. A scatter plot is visible in the bottom right corner, with the y-axis labeled log(Price) and the x-axis labeled Intercept.

And changed revenue is this. So, this will be the total cost and profit would be profit would be changed revenue minus cost. So, the total profit will be 120 dollar, ok; from this one particular guy. So, let me now what we will do we will put it into sort of a we will do have to do this calculation for all the retailer.



(Refer Slide Time: 35:52)



The screenshot displays the RStudio interface. The main editor window contains the following R code:

```
71 profit = changed_revenue - cost
72
73
74 ## do this calculation for all the retailers
75
76 for( i in retailer){
77   data_sub = cheese[cheese$RETAILER==i
78     ,c(2:4)]
79   retailer_summy = round(apply(data_sub,2,mean,na.rm=T),3)
80
81   current_price = retailer_summy['PRICE']
82
```

The Environment pane on the right shows the following objects:

- cheese.. 61 obs. of 4 variab...
- cost num [1, 1] 903
- data\_s\_ 68 obs. of 3 variab...
- profit num [1, 1] 120
- volume\_ num [1, 1] 445
- volume\_ num [1, 1] -76.4

The Console window shows the output of the loop, listing various retailers:

```
[80] "ST. LOUIS - NATIONAL SUPER"
[81] "ST. LOUIS - SCHNUCK MARKETS"
[82] "SYRACUSE - P & C FOOD MARKE"
[83] "SYRACUSE - PRICE CHOPPER"
[84] "SYRACUSE - WEGMANS"
[85] "TAMPA/ST. PETE - KASH N KARRY"
[86] "TAMPA/ST. PETE - PUBLIX"
[87] "TAMPA/ST. PETE - WINN DIXIE"
[88] "WICHITA - DILLON COMPANIES"
```

A scatter plot is visible in the bottom right corner, with the y-axis labeled 'log(Price)' and the x-axis labeled 'Intercept'. The plot shows a positive correlation between the intercept and the log of the price.

The NPTEL logo is visible in the top right corner of the RStudio window.

For do this calculation for all the retailer. For i in retailer ok. So, the first thing I have to do is perhaps data sub retailer sub i right, ok.

(Refer Slide Time: 37:06)

The screenshot displays the RStudio interface. The main editor window contains the following R code:

```
68 changed_revenue = retailer_summary["VOLUME"]*(1+volume_change/100)*alternate_pr
69 names(changed_revenue)="updated expected revenue"
70 cost = retailer_summary["VOLUME"]*(1+volume_change/100)*unit_cost
71 profit = changed_revenue - cost
72
73
74 # do this calculation for all the retailers
75
76 for( i in retailer){
92
93
94
```

The Environment pane on the right shows the following objects:

- expect... Named num 4562
- filter logi [1:5555] FALSE \_
- i "WICHITA - DILLON CO.
- nreg 88L
- retail\_ chr [1:88] "ALBANY,N
- retail\_ 8
- retail\_ Named num [1:3] 1283\_

The Console pane at the bottom shows the output of the R script:

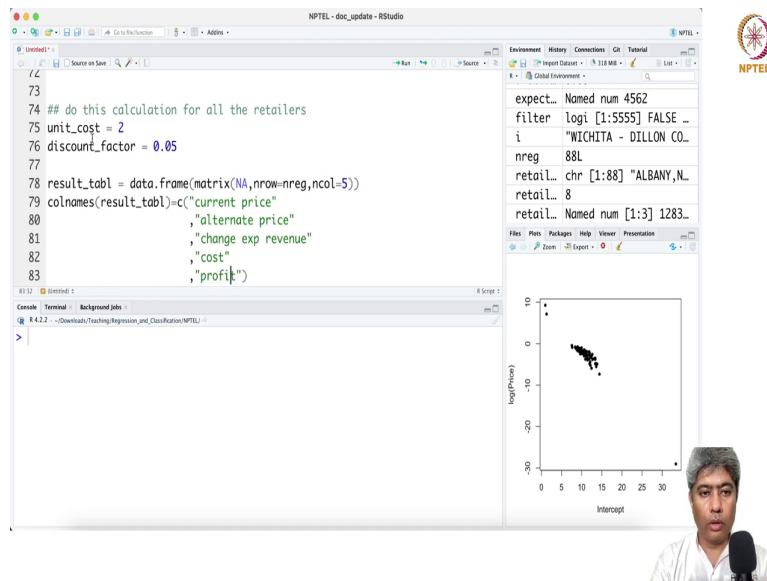
```
ST. LOUIS - NATIONAL SUPER 13.84359 -4.928126
ST. LOUIS - SCHNUCK MARKETS 12.29257 -3.254894
SYRACUSE - P & C FOOD MARKE 9.666966 -2.477946
SYRACUSE - PRICE CHOPPER 12.44068 -5.934727
SYRACUSE - WEGMANS 12.11473 -4.425017
TAMPA/ST. PETE - KASH N KARRY 9.420322 -1.1215
TAMPA/ST. PETE - PUBLIX 10.93023 -2.331113
TAMPA/ST. PETE - WINN DIXIE 10.67047 -2.206763
WICHITA - DILLON COMPANIES 8.945987 -1.491804
```

A scatter plot is visible in the bottom right corner, with the y-axis labeled 'log(Price)' and the x-axis labeled 'Intercept'. The plot shows a positive correlation between the two variables.

The NPTEL logo is visible in the top right corner of the RStudio window.

So, retailer summary and let me just run this part. Let me see if this is coming properly, alright i comma, yeah. So, it is alright that is happening. That is fine. So, now what we have to do, we have to just take the current price; retailer summary is done, right. And then we have to just take current price. I so, let me just take if it is doing well, yeah ok, fine.

(Refer Slide Time: 38:30)



The screenshot displays the RStudio interface. The main editor window contains the following R code:

```
73
74 ## do this calculation for all the retailers
75 unit_cost = 2
76 discount_factor = 0.05
77
78 result_tabl = data.frame(matrix(NA,nrow=nreg,ncol=5))
79 colnames(result_tabl)=c("current price"
80                        ,"alternate price"
81                        ,"change exp revenue"
82                        ,"cost"
83                        ,"profit")
```

The Environment pane on the right shows the following objects:

- expect\_ Named num 4562
- filter logi [1:5555] FALSE \_
- i "WICHITA - DILLON CO."
- nreg 88L
- retail\_ chr [1:88] "ALBANY,NL"
- retail\_ 8
- retail\_ Named num [1:3] 1283\_

The Console pane at the bottom left is empty. The bottom right pane shows a scatter plot with 'Intercept' on the x-axis (ranging from 0 to 30) and 'log(Phone)' on the y-axis (ranging from -30 to 10). The plot shows a positive correlation between the intercept and the log of phone volume. An NPTEL logo is visible in the top right corner of the RStudio window.

Now, what I am going to do is I am going to calculate the current volume and the expected volume. This, these things will be exactly same actually. No change. So, this will be, there will be no change. But so, now, what we need, we need essentially, we I need a discount factor. So, discount factor is here actually, you know.

So, let me just take these things, two things. So, I do not need this ha. Unit cost is 2, retail. And then essentially, I need result table, ok. Result table data dot frame matrix NA nrow equal to nreg and ncol equal to 5, ok. And colnames equal to result table equal to c. First is "current price e". Second is "alternate price" price. Third is changed in "change in expected revenue". How much change we are expecting total "cost" and final "profit", ok, alright.

(Refer Slide Time: 40:47)



The screenshot displays the RStudio interface. The top-left pane shows a data frame with columns: current price, alternate price, change exp revenue, cost, and profit. The rows list various retailers such as SOUTH CAROLINA - BI LO, SOUTH CAROLINA - FOOD LION, etc. The bottom-left pane shows the R console with the following code:

```
> result_tabl = data.frame(matrix(NA,nrow=nreg,ncol=5))
> colnames(result_tabl)=c("current price"
+ "alternate price"
+ "change exp revenue"
+ "cost"
+ "profit")
> rownames(result_tabl)=retailer
> View(result_tabl)
```

The top-right pane shows the Environment window with the following output:

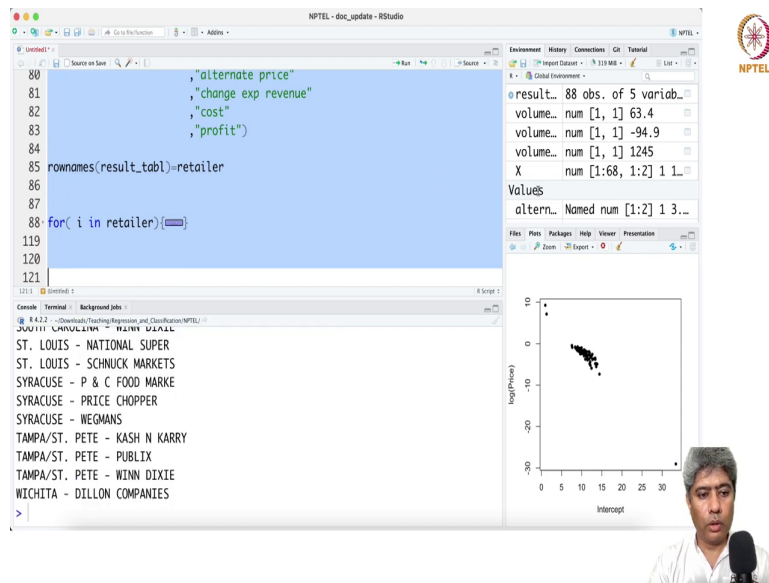
```
result. 88 obs. of 5 variab...
volume.. num [1, 1] 445
volume.. num [1, 1] -76.4
volume.. num [1, 1] 1883
X num [1:68, 1:2] 1 1...
Values
altern.. Named num [1:2] 1 3...
```

The bottom-right pane shows a scatter plot of log(Price) versus Intercept, with data points clustered around the origin.



So, now if I just run it, and result table equals to, I have to do one more thing. We have to do that is col names. I have to do row names also, rownames. Result table equal to retailer, right, ok. So, if I just now define it, you can see all these names are there. I have just created a place holder for that.

(Refer Slide Time: 41:35)



The screenshot shows the RStudio interface. The main editor contains the following R code:

```
80     , "alternate price"  
81     , "change exp revenue"  
82     , "cost"  
83     , "profit")  
84  
85 rownames(result_tabl)=retailer  
86  
87  
88 for( i in retailer){  
119  
120  
121
```

The Environment pane on the right shows the following objects:

- result.. 88 obs. of 5 variab...
- volume.. num [1, 1] 63.4
- volume.. num [1, 1] -94.9
- volume.. num [1, 1] 1245
- X num [1:68, 1:2] 1 1...
- Values  
altern.. Named num [1:2] 1 3...

The Console pane at the bottom shows the following output:

```
ST. LOUIS - NATIONAL SUPER  
ST. LOUIS - SCHNUCK MARKETS  
SYRACUSE - P & C FOOD MARKE  
SYRACUSE - PRICE CHOPPER  
SYRACUSE - WEGMANS  
TAMPA/ST. PETE - KASH N KARRY  
TAMPA/ST. PETE - PUBLIX  
TAMPA/ST. PETE - WINN DIXIE  
WICHITA - DILLON COMPANIES
```

A scatter plot is visible in the bottom right corner, with the y-axis labeled 'log(Price)' and the x-axis labeled 'Intercept'. The plot shows a positive correlation between the intercept and the log price.

The NPTEL logo is visible in the top right corner of the RStudio window.

Now, result table equals to i comma c. So, what I have to now fit is current price. What is current price? This is the current price. Then alternate price. Then current price, alternate price, change in expected revenue; so, changed revenue, then cost because of the change offer and what is the final profit ok. So, let me see alright. Let me just run it, ok.

(Refer Slide Time: 42:38)

The screenshot displays the RStudio interface. The main window shows a data table with columns: name, current price, alternate price, change exp revenue, cost, and profit. The table lists various grocery stores and their associated values. The Environment pane on the right shows the result of a linear regression model, including coefficients for volume and X, and the values of the alternate and named variables. A scatter plot is visible in the bottom right, showing the relationship between log(Price) and Intercept.

name	current price	alternate price	change exp revenue	cost	profit
ALBANY - PRICE CHOPPER	2.758	2.62020	7.97038e+01	6.08388e+01	1.88650e+01
ATLANTA - KROGER CO	2.828	2.88860	6.36962e+02	4.59288e+02	1.57673e+02
ATLANTA - WINN DIXIE	2.688	2.53480	3.04272e+02	2.40050e+02	6.42364e+01
BALTIMORE - GANT FOOD INC	3.682	3.49790	4.26279e+00	2.47302e+00	1.82249e+00
BALTIMORE - SAFeway	3.882	3.68880	2.26038e+01	1.62124e+01	5.79642e+01
BALTIMORE - SUPER FRESH	3.682	3.53740	5.68810e+00	3.19805e+00	2.43837e+00
BIRMINGHAM - BRUNOS	2.362	2.24920	2.63730e+01	2.33380e+01	2.83488e+02
BIRMINGHAM - KROGER	2.386	2.28870	1.62318e+01	9.02793e+00	1.20387e+02
BIRMINGHAM - WINN DIXIE	2.338	2.20350	3.29417e+01	2.99950e+01	3.63816e+01
BOSTON - SHARS	2.440	2.33880	3.66225e+01	3.45144e+01	2.30779e+02
BOSTON - STAR MARKET	3.048	2.89020	4.86804e+00	3.02124e+00	1.83307e+00
BOSTON - STOP & SHOP	2.482	2.35850	2.27175e+01	1.92770e+01	3.46427e+00
BUFFALO - TOPS MARKETS	3.006	2.85570	4.83308e+00	2.87424e+00	1.22827e+00
BUFFALO - WEGMANS	3.112	2.95730	3.59867e+00	1.40744e+00	8.17419e+01

Environment pane output:

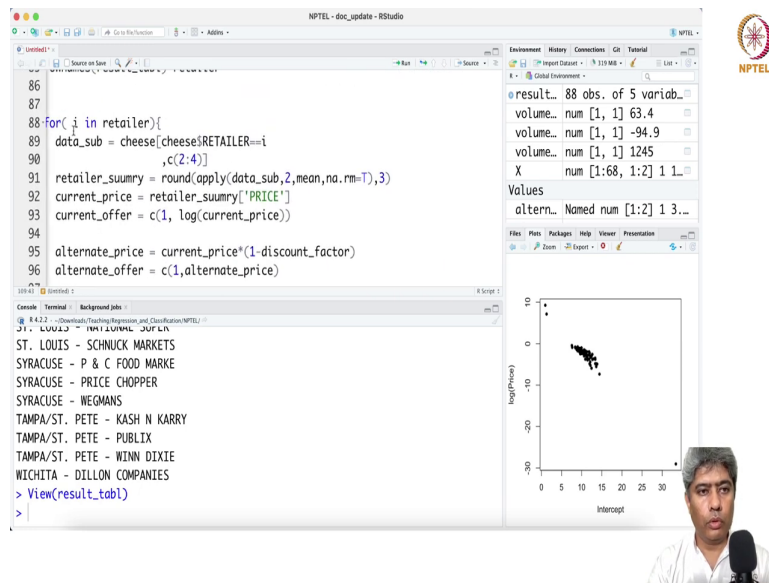
```
result. 88 obs. of 5 variab...  
volume.. num [1, 1] 63.4  
volume.. num [1, 1] -94.9  
volume.. num [1, 1] 1245  
X num [1:68, 1:2] 1 1...  
Values  
altern.. Named num [1:2] 1 3...
```

Terminal output:

```
> View(result.tbl)  
>
```

Now, here is the things, ok.

(Refer Slide Time: 42:53)



The screenshot shows an RStudio window with the following code in the editor:

```
86  
87  
88 for(i in retailer){  
89   data_sub = cheese[cheese$RETAILER==i  
90     ,c(2:4)]  
91   retailer_summy = round(apply(data_sub,2,mean,na.rm=T),3)  
92   current_price = retailer_summy["PRICE"]  
93   current_offer = c(1, log(current_price))  
94  
95   alternate_price = current_price*(1-discount_factor)  
96   alternate_offer = c(1,alternate_price)
```



The Environment pane on the right shows the following objects:

```
result. 88 obs. of 5 variab...  
volume. num [1, 1] 63.4  
volume. num [1, 1] -94.9  
volume. num [1, 1] 1245  
X num [1:68, 1:2] 1 1...  
Values  
altern. Named num [1:2] 1 3...
```

The Console pane shows the following output:

```
ST. LOUIS - SCHNUCK MARKETS  
SYRACUSE - P & C FOOD MARKE  
SYRACUSE - PRICE CHOPPER  
SYRACUSE - WEGMANS  
TAMPA/ST. PETE - KASH N KARRY  
TAMPA/ST. PETE - PUBLIX  
TAMPA/ST. PETE - WINN DIXIE  
WICHITA - DILLON COMPANIES  
> View(result_tab1)  
>
```

A scatter plot is displayed in the bottom right corner, with the y-axis labeled "log(Price)" and the x-axis labeled "Intercept". The plot shows a positive correlation between the two variables.



Do one thing boss what I am going to do is round it off to cost I am going to make it round it off to 2 and profit also comma 2 and changed revenue equal to change round, round it off to 2, ok.

(Refer Slide Time: 43:42)

The screenshot shows the RStudio interface with the following components:

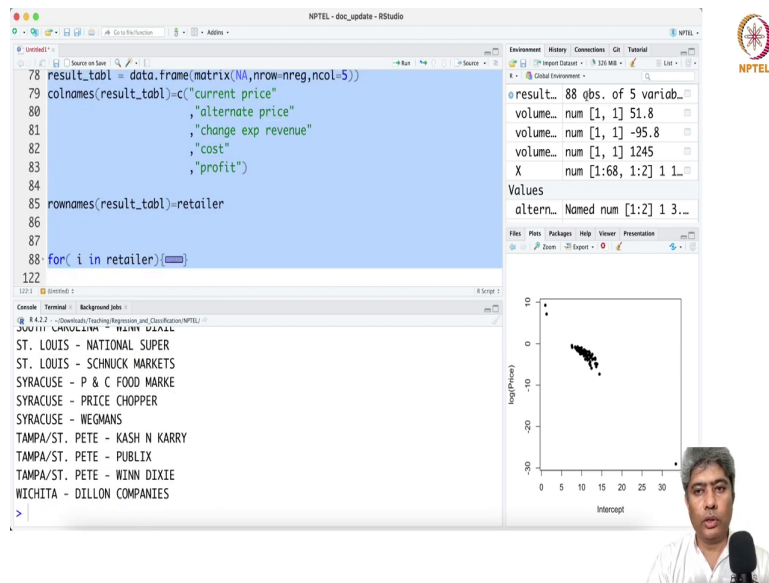
- Environment Pane:** Shows the result of a linear regression model with 88 observations and 5 variables. The variables are: `volume.. num [1, 1] 63.4`, `volume.. num [1, 1] -94.9`, `volume.. num [1, 1] 1245`, `X num [1:68, 1:2] 1 1...`, and `Values altern. Named num [1:2] 1 3...`
- Scatter Plot:** A plot of `log(Price)` versus `Intercept`. The y-axis ranges from -20 to 10, and the x-axis ranges from 0 to 30. The data points form a downward-sloping curve.
- Terminal:** Shows the command `> View(result_tab1)` and the output listing various store names and their corresponding profit values.

Store	profit
ORLANDOUL - FOOD LION	90.89
ORLANDOUL - PUBLIX	149.33
ORLANDOUL - WINN DIXIE	399.37
PHILADELPHIA - ACME MARKET	3.87
PHOENIX - FRY'S FOOD STORE	8.30
PHOENIX - SAFEWAY	9.30
PHOENIX - SMITHS FOOD	244.66
PITTSBURGH - GIANT EAGLE	366.20
RALIGH/GREENBORO - FOOD LION	149.33
RALIGH/GREENBORO - WINN DIXIE	32.81
RICHMOND/NORFOLK - FOOD LION	89.20
RICHMOND/NORFOLK - NEW FARM FRESH	281.62
RIANWICK INPA - FOOD LION	107.52
RIANWICK INPA - KROGER CO	46.81

So, let me just run it once more. Looks like fine ok. Here is the for CHARLOTTE - BI LO. There is a huge profit that are coming up. BI LO is also giving us quite a bit of huge profit.



(Refer Slide Time: 44:00)



The screenshot shows the RStudio interface. The script editor contains the following R code:

```
78 result_tabl = data.frame(matrix(NA,nrow=nreg,ncol=5))
79 colnames(result_tabl)=c("current price"
80                        ,"alternate price"
81                        ,"change exp revenue"
82                        ,"cost"
83                        ,"profit")
84
85 rownames(result_tabl)=retailer
86
87
88 for( i in retailer){
122
```

The Environment pane on the right shows the following objects:

- result: 88 obs. of 5 variab...
- volume.. num [1, 1] 51.8
- volume.. num [1, 1] -95.8
- volume.. num [1, 1] 1245
- X num [1:68, 1:2] 1 1...
- Values altern.. Named num [1:2] 1 3...

The Console pane shows the output of the `retailer` variable:

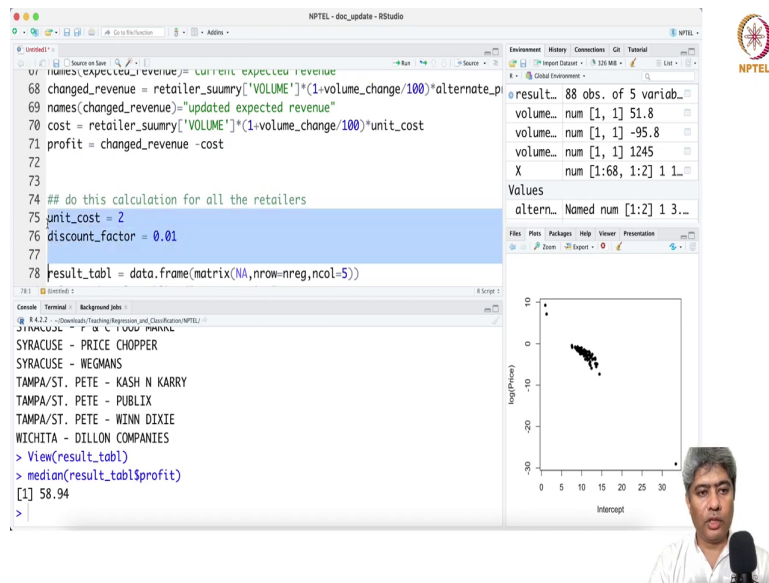
```
ST. LOUIS - NATIONAL SUPER
ST. LOUIS - SCHNUCK MARKETS
SYRACUSE - P & C FOOD MARKE
SYRACUSE - PRICE CHOPPER
SYRACUSE - WEGMANS
TAMPA/ST. PETE - KASH N KARRY
TAMPA/ST. PETE - PUBLIX
TAMPA/ST. PETE - WINN DIXIE
WICHITA - DILLON COMPANIES
```

A scatter plot is visible in the bottom right corner, with the y-axis labeled `log(Price)` and the x-axis labeled `Intercept`. The plot shows a positive correlation between the intercept and the log price, with a few outliers at high intercept values.

The NPTEL logo is visible in the top right corner of the RStudio window.

So, there is something to do with the Bi Lo Charlotte and South Carolina. Why it is, I do not know, maybe there is must there could be some issue with the coefficient. Remember that maybe these are the these one of the outlier coefficient that is getting reflected. But if you now, instead of 0.05, if I just give 1 discount, well, let us see what happened.

(Refer Slide Time: 44:38)



The screenshot shows an RStudio window titled "NPTEL - doc\_update - RStudio". The script editor contains the following R code:

```
68 changed_revenue = retailer_suumry["VOLUME"]*(1+volume_change/100)*alternate.p
69 names(changed_revenue)="updated expected revenue"
70 cost = retailer_suumry["VOLUME"]*(1-volume_change/100)*unit_cost
71 profit = changed_revenue - cost
72
73
74 ## do this calculation for all the retailers
75 unit_cost = 2
76 discount_factor = 0.01
77
78 result_tabl = data.frame(matrix(NA,nrow=nreg,ncol=5))
```

The console output shows the following:

```
> View(result_tabl)
> median(result_tabl$profit)
[1] 58.94
>
```

The Environment pane shows the following variables:

Variable	Class	Attributes
result	num	[1, 1] 51.8
volume	num	[1, 1] -95.8
volume	num	[1, 1] 1245
X	num	[1:68, 1:2] 1 1

The Values pane shows:

```
altern. Named num [1:2] 1 3...
```

The console also displays a list of retailer names:

```
SYRACUSE - PRICE CHOPPER
SYRACUSE - WEGMANS
TAMPA/ST. PETE - KASH N KARRY
TAMPA/ST. PETE - PUBLIX
TAMPA/ST. PETE - WINN DIXIE
WICHITA - DILLON COMPANIES
```

A scatter plot is visible in the bottom right corner, with the y-axis labeled "log(Profit)" and the x-axis labeled "Intercept". The plot shows a positive correlation between the intercept and the log of profit.

You see ok. So, what I am going to do is result table is dollar profit kind of median profit. What is my median profit from a 58 dollar? Ok so, that makes sense perhaps if I just give 1 percent discount, my median profit is 58 dollar.

(Refer Slide Time: 45:12)

The screenshot shows the RStudio interface. The script editor on the left contains the following R code:

```
81     , "change exp revenue"  
82     , "cost"  
83     , "profit")  
84  
85 rownames(result_tbl)=retailer  
86  
87  
88 for( i in retailer){  
122  
123 median(result_tbl$profit)  
124 max(result_tbl$profit)  
125 max  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000
```

The console output shows:

```
> median(result_tbl$profit)  
[1] 58.94  
> sum(result_tbl$profit)  
[1] 2867163171  
>
```

The environment pane on the right shows the following variables:

```
result. 88 obs. of 5 variab...  
volume. num [1, 1] 51.8  
volume. num [1, 1] -95.8  
volume. num [1, 1] 1245  
X num [1:68, 1:2] 1 1...  
Values  
altern. Named num [1:2] 1 3...
```

The scatter plot on the right shows a positive correlation between 'Intercept' (x-axis, 0 to 30) and 'log(Profit)' (y-axis, -20 to 10). The data points are clustered around a diagonal line, indicating a strong positive relationship.

The NPTEL logo is visible in the top right corner of the RStudio window.

And total profit of if I give this to everybody, everybody, then this is my huge profit or maybe I am just getting I can give, you know, max profit that can I get is this is a big number. This is definitely not right, something wrong. And then if I give say 10 percent profit, 10 percent discount, then what happens is median profit is 64 and maximum profit.

(Refer Slide Time: 45:52)

The screenshot shows the RStudio interface with the following components:

- Environment:** Shows a data frame with 88 observations and 5 variables.
- Console:** Contains the following R commands and their outputs:

```
> median(result_tbl$profit)
[1] 64.69
> max(result_tbl$profit)
[1] 156325188
> View(result_tbl)
> beta.ls["CHARLOTTE - BI LO*"]
Intercept log(Price)
1.270048 7.129214
```
- Table:** A table with columns: current price, alternate price, change exp revenue, cost, profit. Rows include various store locations like Buffalo-Rochester - Wegmans, Charlotte - Bi Lo, etc.
- Scatter Plot:** A plot of log(Price) vs Intercept, showing a positive correlation.

So, I think I am pretty much. So, this particular guy, whoever this is, this is Bi Lo Charlotte Bi Lo. Let me just take copy and beta dot ls, yeah.

(Refer Slide Time: 46:17)

The screenshot shows the RStudio interface with the following components:

- Environment:** Shows a data frame with 88 observations and 5 variables.
- Console:** Contains the following R code and output:

```
> max(result.tbl$profit)
[1] 156325188
> View(result.tbl)
> beta.ls["CHARLOTTE - BI LO",]
Intercept log(Price)
1.270048 7.129214
> beta.ls["SOUTH CAROLINA - BI LO",]
Intercept log(Price)
0.9619538 9.2589297
>
```
- Scatter Plot:** A plot of log(Price) versus Intercept, showing a positive correlation between the two variables.

So, 1.27 is somewhere here, no, yeah. And 7 point something on the log price. So, this is definitely one these two guys. Here is, I am pretty sure these two guys here is giving us the problem. So, if I just put it there, yeah, 0.96 and 9.5. So, my log price, the coefficient is 9.25.

So, most of my coefficient are negative, you see, most of the coefficient, but this guy is giving me very large positive coefficient. That means and intercept is 0.96, which is near 1, whereas, most of the intercept is near 10, at least above 5. So, this tells us this two estimates for South Carolina Bi Lo may be. What I will do, I will do a plot.

(Refer Slide Time: 47:32)

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

```
14 cheese_albany=cheese[cheese$RETAILER=="ALBANY, NY - PRICE CHOPPER",]  
15 points(log(cheese_albany$PRICE), log(cheese_albany$VOLUME)  
16         ,col="red",pch=20)  
17  
18 cheese_LA=cheese[cheese$RETAILER=="LOS ANGELES - LUCKY",]  
19 points(log(cheese_LA$PRICE), log(cheese_LA$VOLUME)  
20         ,col="black",pch=20)  
21  
22 cheese_SC=cheese[cheese$RETAILER=="SOUTH CAROLINA - BI LO",]  
23 points(log(cheese_LA$PRICE), log(cheese_LA$VOLUME)  
24         ,col="black",pch=20)  
25
```

The console shows the execution of the following commands:

```
> plot(log(cheese$PRICE), log(cheese$VOLUME), pch=20, col="skyblue"  
+       , xlab="log(Price)", ylab = "log(Volume)")  
> cheese_albany=cheese[cheese$RETAILER=="ALBANY, NY - PRICE CHOPPER",]  
> points(log(cheese_albany$PRICE), log(cheese_albany$VOLUME)  
+        , col="red", pch=20)  
> cheese_LA=cheese[cheese$RETAILER=="LOS ANGELES - LUCKY",]  
> points(log(cheese_LA$PRICE), log(cheese_LA$VOLUME)  
+        , col="black", pch=20)  
> cheese_SC=cheese[cheese$RETAILER=="SOUTH CAROLINA - BI LO",]  
>
```

The environment pane on the right lists objects: cheese (61 obs. of 4 variab...), cost (num [1, 1] 168), data\_s (68 obs. of 3 variab...), profit (num [1, 1] 88), result (88 obs. of 5 variab...), and volume (num [1, 1] 81.5). The plot shows log(Volume) on the y-axis (ranging from 6 to 12) and log(Price) on the x-axis (ranging from 0.4 to 1.4). The plot contains a dense cloud of light blue points, with a few red points (Albany Chopper) and black points (Los Angeles Lucky and South Carolina Bi Lo) scattered within the cloud.

So, in this, let me just draw a plot. So, here is the plot. Here is the Chinese cheese data set Albany Chopper. This is our LA Los Angeles. This is SC Bi Lo South Carolina. And this is “SOUTH CAROLINA - BI LO”, right. And then if I just plot that guy, cheese SC versus cheese SC, and instead of “black”, maybe “brown” ok.

Let us see where this guy is. You see, this is the problem. All points are here and that is why. So, there is no diversity in the points. All points are plotted here. That is the issue. So, all I can just say “blue”. Let me see if I can show you guys, yeah.

See, all points are in one line. And as a result, probably the line is very sharp. So, these kind of weird point, set of points, if you get, you cannot do much effectively, you cannot do much you have, yeah, one possibility is you can try some Bayesian; hierarchical Bayesian models, but that is bit out of the question here. But we will see if we can fix it. But I hope you

understood what is happening. So, if we even if we decide to, ok, let us drop these two guy, ok. We have this, if we drop these two guy for the time being from the.

(Refer Slide Time: 49:44)

```
89 names(result_tabl)=retailer
90
91
92 i in retailer)()
126
127 an(result_tabl$profit)
128 result_tabl$profit
129
130 lt_tabl = result_tabl[-c("CHARLOTTE - BI LO", "SOUTH CAROLINA - BI LO"),]
131
132
133
```

```
## 8.4.2.2 - Overview: Training Regression and Classification
log(cheese_SCSPRICE)~log(cheese_SCSVOLUME) - BI LO ,]
> points(log(cheese_SCSPRICE),log(cheese_SCSVOLUME)
+ ,col="brown",pch=20)
> points(log(cheese_SCSPRICE),log(cheese_SCSVOLUME)
+ ,col="blue",pch=20)
> View(result_tabl)
> result_tabl = result_tabl[-c("CHARLOTTE - BI LO", "SOUTH CAROLINA - BI LO"),]
Error in -c("CHARLOTTE - BI LO", "SOUTH CAROLINA - BI LO") :
  invalid argument to unary operator
> View(result_tabl)
>
```

Environment: Global Environment

- cheese... 61 obs. of 4 variab...
- cheese... 61 obs. of 4 variab...
- cost num [1, 1] 168
- data\_s... 68 obs. of 3 variab...
- profit num [1, 1] 88
- result... 88 obs. of 5 variab...
- volume... num [1, 1] 81.5

log(Volume)

log(Price)

So, so, what are those two names? Let us see. One is CHARLOTTE will BI LO. And the other guy is other person, other one where we got a terrible estimate is South Carolina Bi Lo. So, this, two retailer I think have some issues. So, what we are going to do? We are going to result table, minus these two person result table.

Now, if I just run this, I am not sure if we can drop it in this way, alright. So, it is not looks like it is not mutable ok. If I just drop this guy like this, invalid line argument, ok. What I can do is. So, I just, why not I do this? Just for the time being, I can come up with a much better index, i n d x of (Refer Time: 51:58) tab retailer.

(Refer Slide Time: 51:51)

The screenshot shows the RStudio interface. The script editor contains the following code:

```
89 rownames(result_tabl)=retailer
90
91 for( i in retailer){
92   median(result_tabl$profit)
93   max(result_tabl$profit)
94 }
95 result_tabl = result_tabl[, ]
96
97 which(retailer=="CHARLOTTE - BI LO")
```

The console shows the following error messages:

```
> View(result_tabl)
> result_tabl = result_tabl[~"CHARLOTTE - BI LO",]
Error in ~"CHARLOTTE - BI LO" : invalid argument to unary operator
> View(result_tabl)
> index(retailer=="CHARLOTTE - BI LO")
Error in index(retailer == "CHARLOTTE - BI LO") :
could not find function "index"
> which(retailer=="CHARLOTTE - BI LO")
[1] 15
>
```

The Environment pane on the right shows several objects:

- cheese... 61 obs. of 4 variab...
- cheese... 61 obs. of 4 variab...
- cost num [1, 1] 168
- data\_s... 68 obs. of 3 variab...
- profit num [1, 1] 88
- result... 88 obs. of 5 variab...
- volume... num [1, 1] 81.5

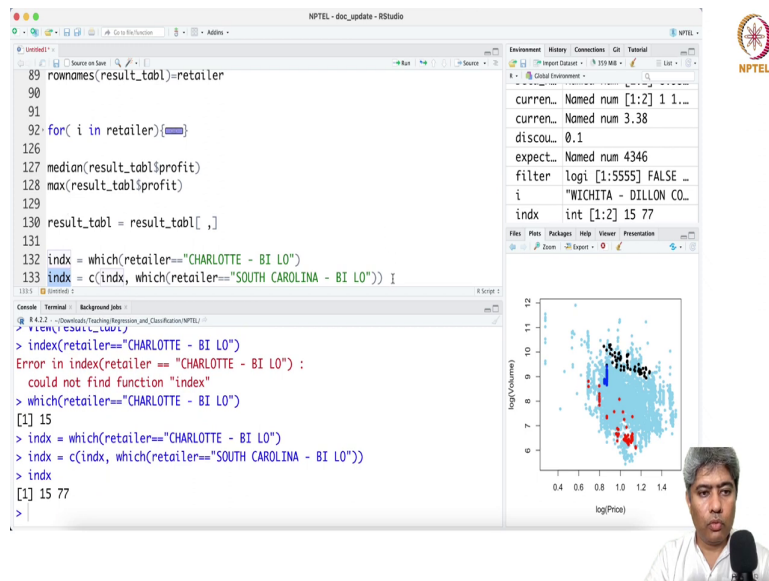
A scatter plot is displayed in the bottom right, with log(Volume) on the y-axis (ranging from 6 to 12) and log(Price) on the x-axis (ranging from 0.4 to 1.4). The plot shows a dense cloud of blue points with a few red points scattered throughout.

The NPTEL logo is visible in the top right corner of the RStudio window.

Equals to, CHARLOTTE will, “CHARLOTTE - BI LO”, which 15.



(Refer Slide Time: 52:32)



The screenshot shows an RStudio window titled "NPTEL - doc\_update - RStudio". The editor pane contains the following R code:

```
89 rownames(result_tabl)=retailer
90
91 for( i in retailer){
92   median(result_tabl$profit)
93   max(result_tabl$profit)
94 }
95 result_tabl = result_tabl[, ]
96
97 indx = which(retailer=="CHARLOTTE - BI LO")
98 indx = c(indx, which(retailer=="SOUTH CAROLINA - BI LO"))
```

The console shows the execution of the code, with an error message for the first `index` call:

```
> index(retailer=="CHARLOTTE - BI LO")
Error in index(retailer == "CHARLOTTE - BI LO") :
  could not find function "index"
> which(retailer=="CHARLOTTE - BI LO")
[1] 15
> indx = which(retailer=="CHARLOTTE - BI LO")
> indx = c(indx, which(retailer=="SOUTH CAROLINA - BI LO"))
> indx
[1] 15 77
>
```

The Environment pane shows the following objects:

current_	Named num [1:2] 1 1...
current_	Named num 3.38
discou_	0.1
expect_	Named num 4346
filter	logi [1:5555] FALSE ...
i	"WICHITA - DILLON CO_
indx	int [1:2] 15 77

The Plots pane shows a scatter plot of `log(Volume)` versus `log(Price)`. The plot contains a large number of blue points, with a few red points and a vertical line at `log(Price) ≈ 1.0`. A small inset image of a man speaking is visible in the bottom right corner of the RStudio window.

And so, x and then equal to c index comma "SOUTH CAROLINA - BI LO" so, these are the two cases and then result table minus index if I just say that. And now if you see I have dropped these two guys kind of.

(Refer Slide Time: 53:27)

The screenshot shows the RStudio interface. The script editor contains the following R code:

```
91  
92 for( i in retailer){  
126  
127  
128 indx = which(retailer=="CHARLOTTE - BI LO")  
129 indx = c(indx, which(retailer=="SOUTH CAROLINA - BI LO"))  
130 result_tabl = result_tabl[-indx ,]  
131  
132 median(result_tabl$profit)  
133 max(result_tabl$profit)  
134 sum(result_tabl$profit)  
135
```

The console shows the output of the code:

```
> indx = which(retailer=="CHARLOTTE - BI LO")  
> indx = c(indx, which(retailer=="SOUTH CAROLINA - BI LO"))  
> result_tabl = result_tabl[-indx ,]  
>  
> median(result_tabl$profit)  
[1] 561395  
> max(result_tabl$profit)  
[1] 820.05  
> sum(result_tabl$profit)  
[1] 9822.57  
>
```

The Environment pane shows the following objects:

- result: 86 obs. of 5 variab...
- volume: num [1, 1] 51.8
- volume: num [1, 1] -95.8
- volume: num [1, 1] 1245
- X: num [1:68, 1:2] 1 1...
- Values: altern. Named num [1:2] 1 3...

The scatter plot shows log(Volume) on the y-axis (ranging from 0 to 12) and log(Price) on the x-axis (ranging from 0.4 to 1.4). The plot contains numerous data points, with a cluster of points around log(Price) = 1.0 and log(Volume) = 10.0.

Now, what is happening is if I now median profit and this is the max profit and sum if the result table and dollar profit is this, right. Now, if I just go and say 0.01 percent if I give 1 percent discount then median profit is 56. Let me just start with no discount. So, I will not give any discount.

Current price, my kind of median profit is 55; max I can earn 841 from one particular retailer. And total profit can be expected to 9,014 dollar. Now, if I give 5 percent discount then it is median profit sort of increased 59.717. Actually, let me just do one thing. Let me not print this line otherwise it is creating lot of (Refer Time: 55:05).

(Refer Slide Time: 55:11)

The screenshot displays the RStudio interface. The main editor window contains the following R code:

```
91  
92 for( i in retailer){  
126  
127  
128 indx = which(retailer=="CHARLOTTE - BI LO")  
129 indx = c(indx, which(retailer=="SOUTH CAROLINA - BI LO"))  
130 result_tabl = result_tabl[-indx,]  
131  
132 median(result_tabl$profit)  
133 max(result_tabl$profit)  
134 sum(result_tabl$profit)  
135  
|
```

The console window shows the execution of the code for the 'SOUTH CAROLINA - BI LO' retailer:

```
> indx = which(retailer=="SOUTH CAROLINA - BI LO")  
> result_tabl = result_tabl[-indx,]  
>  
> median(result_tabl$profit)  
[1] 55.46  
> max(result_tabl$profit)  
[1] 841.47  
> sum(result_tabl$profit)  
[1] 9814.6  
>
```

The Environment pane on the right shows the following objects:

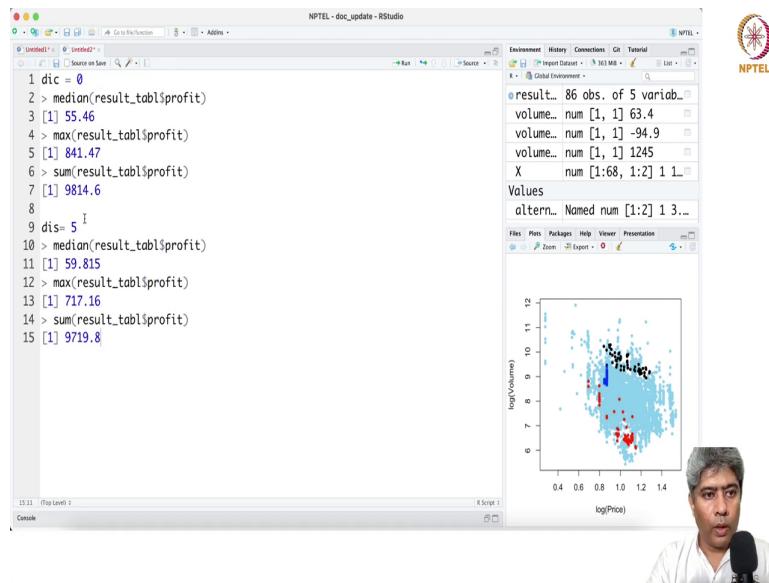
- result: 86 obs. of 5 variab...
- volume: num [1, 1] 49.2
- volume: num [1, 1] -96
- volume: num [1, 1] 1245
- X: num [1:68, 1:2] 1 1...
- Values: altern. Named num [1:2] 1 3...

The plot pane shows a scatter plot of log(Volume) on the y-axis (ranging from 6 to 12) versus log(Price) on the x-axis (ranging from 0.4 to 1.4). The plot contains numerous data points, with a cluster of points at higher log(Price) and log(Volume) values.

The NPTEL logo is visible in the top right corner of the RStudio window.

So, so, let me just create another thing, ok. If I do not give discount then that is what happen so, discount 0, discount.

(Refer Slide Time: 55:24)



The screenshot shows the RStudio interface. The console on the left contains the following R code and its output:

```
1 dic = 0
2 > median(result_tabl$profit)
3 [1] 55.46
4 > max(result_tabl$profit)
5 [1] 841.47
6 > sum(result_tabl$profit)
7 [1] 9814.6
8
9 dis = 5
10 > median(result_tabl$profit)
11 [1] 59.815
12 > max(result_tabl$profit)
13 [1] 717.16
14 > sum(result_tabl$profit)
15 [1] 9719.8
```

The Environment pane on the right shows the following variables:

- result: 86 obs. of 5 variab...
- volume.. num [1, 1] 63.4
- volume.. num [1, 1] -94.9
- volume.. num [1, 1] 1245
- X num [1:68, 1:2] 1 1...

The Values pane shows:

```
altern. Named num [1:2] 1 3...
```

The plot pane shows a scatter plot of log(Volume) versus log(Price). The x-axis is labeled 'log(Price)' and ranges from 0.4 to 1.4. The y-axis is labeled 'log(Volume)' and ranges from 6 to 12. The plot shows a dense cloud of blue points with a few red points scattered throughout.

The NPTEL logo is visible in the top right corner of the RStudio window. A small inset image of a man speaking is visible in the bottom right corner of the RStudio window.

Discount equal to 0 I get median profit of these guys. If I give 5 percent discount if I give 5 percent discount then I get this. Discount is 5 percent. Now, my total profit reduce a bit.

(Refer Slide Time: 56:07)

The image shows an RStudio window with the following R code in the editor:

```
91  
92 for( i in retailer){  
126  
127  
128 indx = which(retailer=="CHARLOTTE - BI LO")  
129 indx = c(indx, which(retailer=="SOUTH CAROLINA - BI LO"))  
130 result_tabl = result_tabl[-indx,]  
131  
132 median(result_tabl$profit)  
133 max(result_tabl$profit)  
134 sum(result_tabl$profit)  
135
```

The console output shows the results of the code:

```
> indx = which(retailer=="CHARLOTTE - BI LO")  
> indx = c(indx, which(retailer=="SOUTH CAROLINA - BI LO"))  
> result_tabl = result_tabl[-indx,]  
>  
> median(result_tabl$profit)  
[1] 62.16  
> max(result_tabl$profit)  
[1] 544.17  
> sum(result_tabl$profit)  
[1] 9178.08  
>
```

The Environment pane on the right shows the following variables:

```
result... 86 obs. of 5 variab...  
volume... num [1, 1] 81.5  
volume... num [1, 1] -93.5  
volume... num [1, 1] 1245  
X num [1:68, 1:2] 1 1...
```

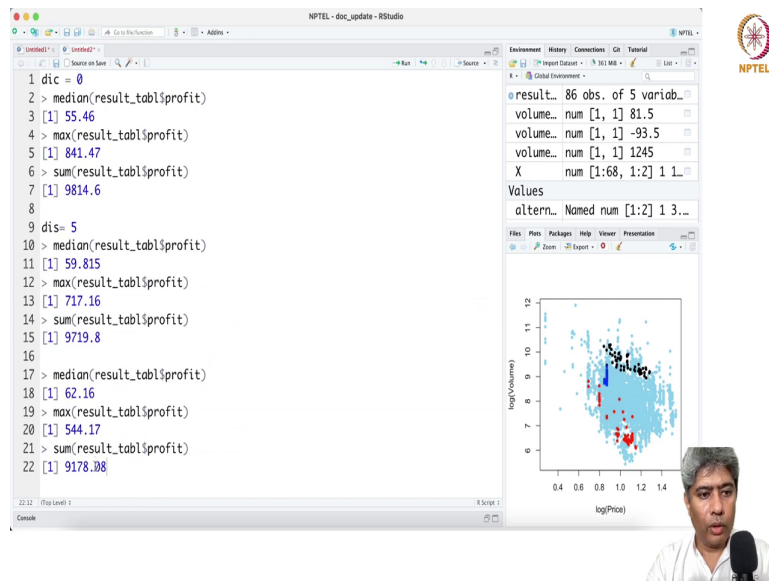
The Values pane shows:

```
altern... Named num [1:2] 1 3...
```

The scatter plot shows log(Volume) on the y-axis (ranging from 0 to 12) and log(Price) on the x-axis (ranging from 0.4 to 1.4). The plot contains a dense cloud of blue points, with a few red points scattered throughout. A vertical line is drawn at approximately log(Price) = 0.8.

And then if I give 10 percent discount my median profit increases probably 62.

(Refer Slide Time: 56:28)



The screenshot shows the RStudio interface. The console on the left contains the following R code and its output:

```
1 dic = 0
2 > median(result_tabl$profit)
3 [1] 55.46
4 > max(result_tabl$profit)
5 [1] 841.47
6 > sum(result_tabl$profit)
7 [1] 9814.6
8
9 dis = 5
10 > median(result_tabl$profit)
11 [1] 59.815
12 > max(result_tabl$profit)
13 [1] 717.16
14 > sum(result_tabl$profit)
15 [1] 9719.8
16
17 > median(result_tabl$profit)
18 [1] 62.16
19 > max(result_tabl$profit)
20 [1] 544.17
21 > sum(result_tabl$profit)
22 [1] 9178.38
```

The Environment pane on the right shows the following variables:

- result: 86 obs. of 5 variab...
- volume: num [1, 1] 81.5
- volume: num [1, 1] -93.5
- volume: num [1, 1] 1245
- X: num [1:68, 1:2] 1 1...

The Values section shows: altern.: Named num [1:2] 1 3...

A scatter plot is displayed in the bottom right, with log(Volume) on the y-axis (ranging from 6 to 12) and log(Price) on the x-axis (ranging from 0.4 to 1.4). The plot shows a dense cloud of blue points with several red points scattered throughout.

The NPTEL logo is visible in the top right corner of the RStudio window. A small inset video of a man speaking is visible in the bottom right corner of the overall image.

But total profit drops. So, total profit drops, max profit drops, but overall drops.

(Refer Slide Time: 56:49)

The screenshot displays the RStudio interface. The source editor contains R code for data manipulation and analysis. The console shows the execution of these commands, resulting in a data frame and summary statistics. The environment pane shows the current objects in memory. A scatter plot of log(Volume) versus log(Price) is visible in the bottom right, with data points colored by retailer. A small inset video of a man speaking is located in the bottom right corner of the RStudio window.

```
91
92 for( i in retailer){
126
127
128 indx = which(retailer=="CHARLOTTE - BI LO")
129 indx = c(indx, which(retailer=="SOUTH CAROLINA - BI LO"))
130 result_tabl = result_tabl[-indx ,]
131
132 median(result_tabl$profit)
133 max(result_tabl$profit)
134 sum(result_tabl$profit)
135 sum(res)
```

Environment: 86 obs. of 5 variab...  
volume.. num [1, 1] 81.5  
volume.. num [1, 1] -93.5  
volume.. num [1, 1] 1245  
X num [1:68, 1:2] 1 1...  
Values  
altern.. Named num [1:2] 1 3...

Console:  
> indx =  
[1] 62.16  
> max(result\_tabl\$profit)  
[1] 544.17  
> sum(result\_tabl\$profit)  
[1] 9178.08  
>

And then if I just say sum of result table dollar expected revenue “change of expected revenue”. So, at 0 discount what happens is my total revenue is 39,000 dollar when I am giving 5 percent discount, when I am giving 5 percent discount my total revenue is 46,000 dollar. My total profit, my total revenue goes up from 39,000 to 46000.

(Refer Slide Time: 57:49)

The screenshot displays the RStudio interface with the following R code in the console:

```
17 > sum(result_tabl$profit)
18 [1] 9719.8
19 > sum(result_tabl$`change exp revenue`)
20 [1] 46348.74
21
22
23 > median(result_tabl$profit)
24 [1] 62.16
25 > max(result_tabl$profit)
26 [1] 544.17
27 > sum(result_tabl$profit)
28 [1] 9178.08
```

The Environment pane shows the following variables:

- result: 86 obs. of 5 variab...
- volume: num [1, 1] 63.4
- volume: num [1, 1] -94.9
- volume: num [1, 1] 1245
- X: num [1:68, 1:2] 1 1...

The Values pane shows:

```
altern. Named num [1:2] 1 3...
```

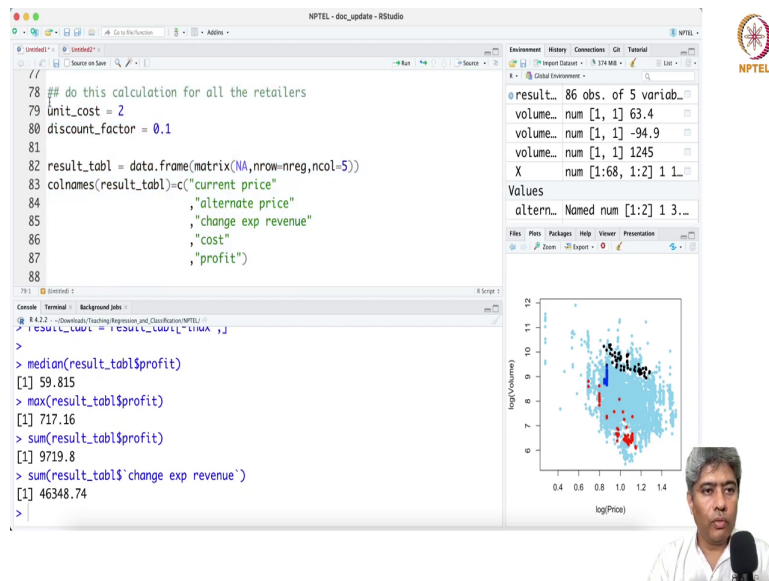
The scatter plot shows log(Volume) on the y-axis (ranging from 6 to 12) and log(Price) on the x-axis (ranging from 0.4 to 1.4). The plot contains numerous blue data points, with a few red points and a vertical line at approximately log(Price) = 1.0.

NPTEL

But total profit came down from 9,800 to 9,700, but median profit raised by 4 dollar, ok. Median is the 50 percent at a lower level overall profit goes up and then what happened at when I give 10 percent discount at 10 percent discount.



(Refer Slide Time: 58:16)



The screenshot shows an RStudio window titled "NPTEL - doc\_update - RStudio". The editor pane contains the following R code:

```
//  
78 ## do this calculation for all the retailers  
79 unit_cost = 2  
80 discount_factor = 0.1  
81  
82 result_tabl = data.frame(matrix(NA,nrow=nreg,ncol=5))  
83 colnames(result_tabl)=c("current price"  
84                        ,"alternate price"  
85                        ,"change exp revenue"  
86                        ,"cost"  
87                        ,"profit")  
88
```

The console shows the following output:

```
>  
> median(result_tabl$profit)  
[1] 59.815  
> max(result_tabl$profit)  
[1] 717.16  
> sum(result_tabl$profit)  
[1] 9719.8  
> sum(result_tabl$`change exp revenue`)  
[1] 46348.74  
>
```

The Environment pane shows the following objects:

```
result... 86 obs. of 5 variab...  
volume... num [1, 1] 63.4  
volume... num [1, 1] -94.9  
volume... num [1, 1] 1245  
X         num [1:68, 1:2] 1 1...  
Values  
altern... Named num [1:2] 1 3...
```

The Plots pane shows a scatter plot of log(Volume) versus log(Price). The plot shows a positive correlation between log(Price) and log(Volume). The x-axis ranges from 0.4 to 1.4, and the y-axis ranges from 6 to 12. The data points are colored in shades of blue and red.

The NPTEL logo is visible in the top right corner of the RStudio window.

So, revenue I expect will go up further, but profit may not go up. So, the median profit again goes up, but total profit goes down, but the revenue goes up quite significantly; discount is 10 percent.

(Refer Slide Time: 58:36)

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

```
27 > sum(result_tabl$profit)
28 [1] 9178.08
29
30 disc=10
31 > median(result_tabl$profit)
32 [1] 62.16
33 > max(result_tabl$profit)
34 [1] 544.17
35 > sum(result_tabl$profit)
36 [1] 9178.08
37 > sum(result_tabl$change exp revenue)
38 [1] 54163.7
```

The Environment pane shows the following variables:

- result: 86 obs. of 5 variab...
- volume: num [1, 1] 81.5
- volume: num [1, 1] -93.5
- volume: num [1, 1] 1245
- X: num [1:68, 1:2] 1 1...

The Values pane shows:

```
altern. Named num [1:2] 1 3...
```

The scatter plot shows log(Volume) on the y-axis (ranging from 0 to 12) and log(Price) on the x-axis (ranging from 0.4 to 1.4). The plot contains numerous data points, with a cluster of points around log(Price) = 1.0 and log(Volume) = 10.0.

So, total profit going up to 46,000 to 54,000. So, total revenue stands up, but total profit goes down marginally. So, sometimes the company wants to show the, you know, grab the market share and in order to, if they see that if my total profit is not taking hit, but my overall market share is going up, why not I give some kind of promotional activity by giving discount.

Now, based on the, this is like giving overall, but at the same time what people can do they can check where overall profit going up and where the profit is not going up. The where the profit is not going up they can like.

(Refer Slide Time: 59:31)

The screenshot shows an RStudio interface with a data table on the left and a scatter plot on the right. The data table has columns for 'current price', 'alternate price', 'change esp. increase', 'cost', and 'profit'. The scatter plot shows 'log(Volume)' on the y-axis (ranging from 6 to 12) and 'log(Price)' on the x-axis (ranging from 0.4 to 1.4). A small inset video of a man speaking is visible in the bottom right corner of the RStudio window.

	current price	alternate price	change esp. increase	cost	profit
HARRISBURG-KANTY - GIANT FOOD STD	2.616	2.354	326.60	276.91	49.67
HARTFORD - STOP & SHOP	3.362	2.958	15.75	11.07	4.68
HOUSTON - KROGER CO	2.644	2.376	2792.77	2318.85	443.92
HOUSTON - KRAMALLS	1.145	2.835	446.90	331.78	115.12
INDIANAPOLIS - KROGER CO	2.610	2.340	696.13	594.20	101.93
JACKSONVILLE - FOOD LOW	2.462	2.218	1587.27	1413.05	174.22
JACKSONVILLE - PUBLIX	2.287	2.168	218.25	174.61	43.64
JACKSONVILLE - WINN DIXIE	2.535	2.265	656.48	575.48	81.00
LITTLE ROCK - HARVEST FOODS	2.676	2.484	197.23	163.79	33.44
LOS ANGELES - LUCKY	2.536	2.624	1061.12	861.15	257.97
LOS ANGELES - KALMIS	3.043	2.787	856.74	625.67	231.09
LOS ANGELES - VONS	3.366	3.420	111.86	86.61	45.85
LOUISVILLE - KROGER CO	2.588	2.526	51.68	44.37	7.31
LOUISVILLE - WINN DIXIE	2.504	2.258	223.86	198.67	25.19
MIAMI - PUBLIX	2.675	2.475	2148.83	1781.28	363.75
MIAMI - WINN DIXIE	2.503	2.257	4858.87	4386.80	544.17
MILWAUKEE - KOHL'S FOOD STORE	3.087	2.778	128.47	92.48	35.99
MONTREAL - KROGER CO	2.399	2.195	5142.18	4958.94	84.24
NEW ENGLAND NORTH SHOP N	2.339	1.625	-82.54	453.51	-13.81
NEW ORLEANS - WINN DIXIE	2.620	2.350	338.49	287.10	51.39
NEW YORK (NEW) - A & P	3.895	3.555	417.48	238.39	179.38
NEW YORK (NEW) - PATHMARK	4.064	3.676	11.54	6.31	5.23
NEW YORK (NEW) - WALDMANS	4.229	3.891	48.77	25.63	23.14
OKLAHOMA CITY - HOMELAND	2.679	2.411	209.60	173.31	35.75
ORLANDO - FOOD LOW	2.477	2.280	641.31	575.51	65.88
ORLANDO - PUBLIX	2.276	2.454	806.39	657.36	149.03
ORLANDO - WINN DIXIE	2.513	2.267	2099.22	1856.32	242.90
PHILADELPHIA - ACME MARKET	3.429	3.881	17.89	11.59	6.30
PHOENIX - FREY'S FOOD STORE	3.330	2.975	34.45	22.99	11.46

Here in this NEW ENGLAND NORTH SHOP N if the profit is negative then they should not give the discount here, ok. Or anywhere if the profit is less than certain value they will not get profit discount; otherwise, they will get a discount. So, this kind of discriminatory pricing practice sometimes helps companies to improve their profitability.

So, I will stop here. I hope you enjoyed that how simple predictive model and regression analysis can help companies and industry to implement dynamic pricing and improve their profitability, improve the total revenue and maintain the cost and pay. We will continue with such more data analysis in the next few videos.

Thank you very much. See you in the next video.

