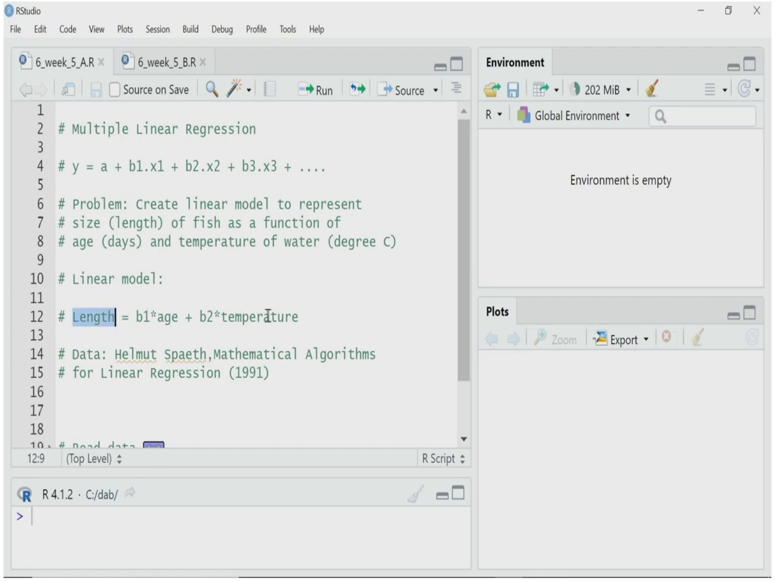
Data Analysis for Biologists
Professor Biplab Bose
Department of Bioscience & Bioengineering
Mehta Family School of Data Science & Artificial Intelligence
Indian Institute of Technology, Guwahati
Lecture 31
Multiple Linear Regression Using R

Welcome back. In this lecture, I will show how to perform Multiple Linear Regression Using R. In simple linear regression, I have only one independent variable, and the linear model that we use is of the form y equal to a plus bx. Whereas, in case of a multiple linear regression, I have a linear model, where I have only 1 dependent variable or response variable, but I have more than one independent variable or independent predictors.

So, the formulation of this multiple linear regression will have a equation of the form y equal to a which is intercept plus b1 into x1 plus b2 into x2 plus b3 into x3 and so on. Here, x1, x2, x3 are my 3 independent variable or 3 predictors. So, in this particular example that I will use for this lecture is a data where we have measured the length of fish as the fish are growing in a tank at different days at different intervals after hatching.

And we have performed experiments with different water temperature. So, we believe the growth of the fish has a relationship, linear relationship with two variable, two predictor one is the age obviously, with age the size of the a fish increase, and also the temperature of the water has a effect on the growth of the fish. So, we will create a linear model here with 2 variables, 2 independent variables, the age of the fish and the temperature of the tank water, whereas the dependent or the response variable will be the length of the fish. I will perform this using R studio.

(Refer Slide Time: 02:31)

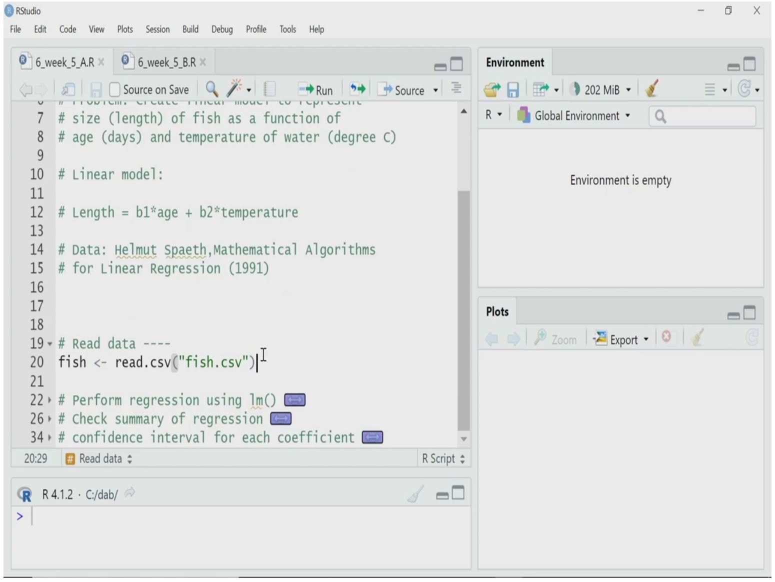


```
# Multiple Linear Regression

# y = a + b1.x1 + b2.x2 + b3.x3 + ....

# Problem: Create linear model to represent
# size (length) of fish as a function of
# age (days) and temperature of water (degree C)

# Linear model:

# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)



# Read data
```

```
>
```

So, as I said, the linear model I want to create is of this form length equal to b1 into age and plus b2 into temperature. So, b1 and b2 are the 2 unknown coefficients that I have to estimate from the data. Age and temperature are 2 independent variables or predictor, length is the dependent variable. And note, I do not have any intercept here.

(Refer Slide Time: 03:03)



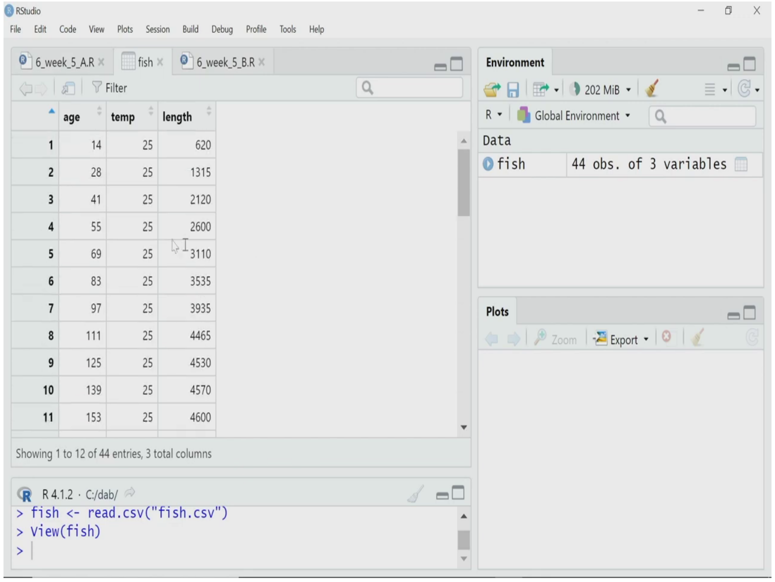
```
# Problem: Create linear model to represent
# size (length) of fish as a function of
# age (days) and temperature of water (degree C)

# Linear model:

# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)



# Read data ----
fish <- read.csv("fish.csv")

# Perform regression using lm()
# Check summary of regression
# confidence interval for each coefficient
```

```
>
```

fish ← read.csv("fish.csv”)

So, I will start with reading the data. I already have the data in a csv file format, so, I will use the read dot csv, the name of the file is fish dot csv. So, I will read that data and I will store that data in a variable fish.

(Refer Slide Time: 03:16)



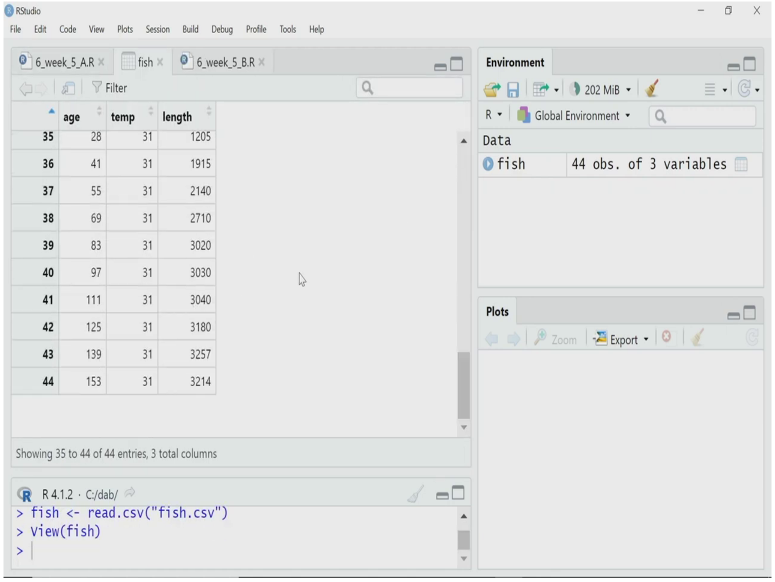
	age	temp	length
1	14	25	620
2	28	25	1315
3	41	25	2120
4	55	25	2600
5	69	25	3110
6	83	25	3535
7	97	25	3935
8	111	25	4465
9	125	25	4530
10	139	25	4570
11	153	25	4600

Showing 1 to 12 of 44 entries, 3 total columns

Data

fish 44 obs. of 3 variables

```
R 4.1.2 · C:/dab/
> fish <- read.csv("fish.csv")
> View(fish)
>
```



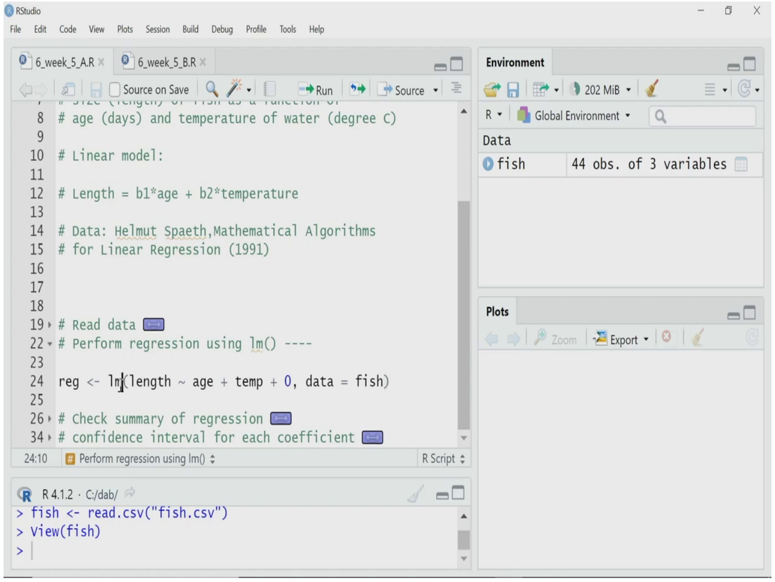
	age	temp	length
35	28	31	1205
36	41	31	1915
37	55	31	2140
38	69	31	2710
39	83	31	3020
40	97	31	3030
41	111	31	3040
42	125	31	3180
43	139	31	3257
44	153	31	3214

Showing 35 to 44 of 44 entries, 3 total columns

Data

fish 44 obs. of 3 variables

```
R 4.1.2 · C:/dab/
> fish <- read.csv("fish.csv")
> View(fish)
>
```



```
# age (days) and temperature of water (degree C)

# Linear model:

# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)



# Read data
# Perform regression using lm() ----

reg <- lm(length ~ age + temp + 0, data = fish)

# Check summary of regression
# confidence interval for each coefficient
```

```
> fish <- read.csv("fish.csv")
> View(fish)
> 
```

Data

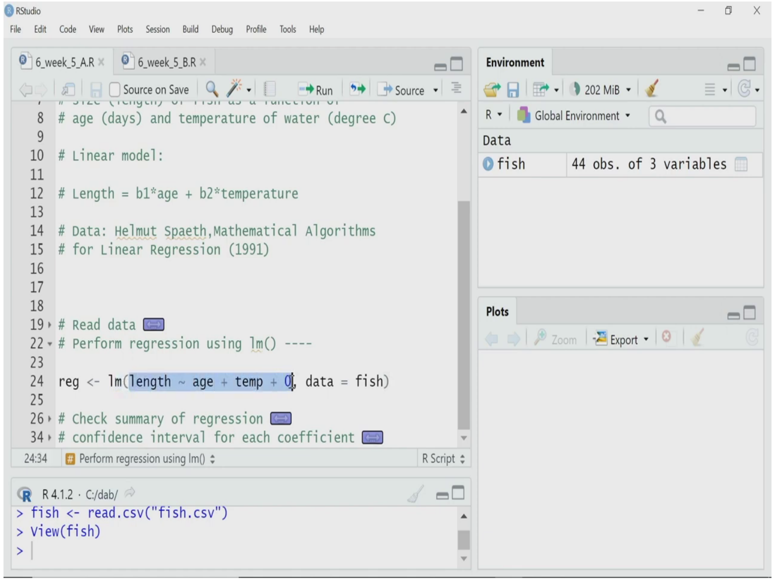
fish 44 obs. of 3 variables

reg ← lm(length ~ age + temp + 0, data = fish)

So, check that data, actually it is a 3 column data and it has around 44 data point observations. The first column, the first variable as the header says is age, age of the fish it is in days and I have the temperature as the second variable of the second column it is centigrade, this the temperature of the water of the tank and length is the last variable which is the dependent variable or the response variable.

So, I want to perform a linear regression multiple linear regression for this data and I will use the same lm function that I have used to create a simple linear regression model that is the advantage of using lm function is a linear model function, it is not specific for just simple linear regression, it can actually handle more than one independent variable. So, how should I specify to lm that, I have more than one independent variable.

(Refer Slide Time: 04:13)

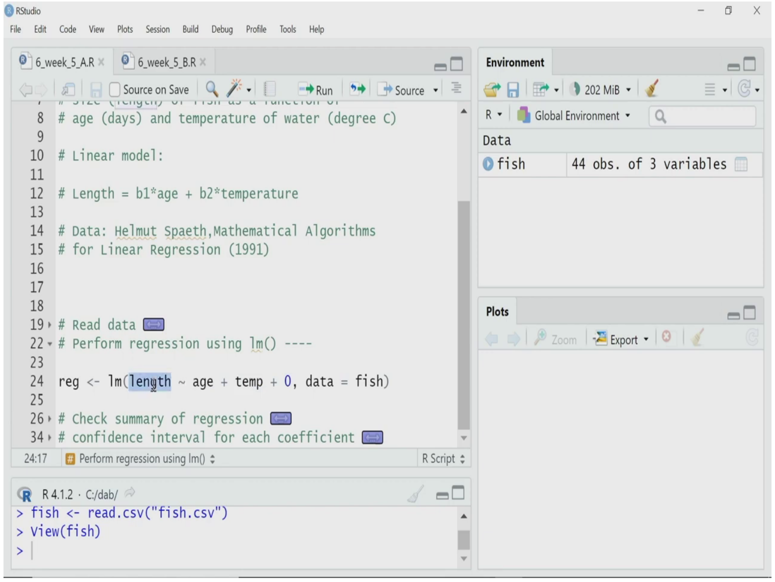


```r
# size (length) of fish as a function of
# age (days) and temperature of water (degree C)

# Linear model:

# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)



# Read data
# Perform regression using lm() ----

reg <- lm(length ~ age + temp + 0, data = fish)

# Check summary of regression
# confidence interval for each coefficient
```

Data

fish	44 obs. of 3 variables

```
> fish <- read.csv("fish.csv")
> View(fish)
>
```

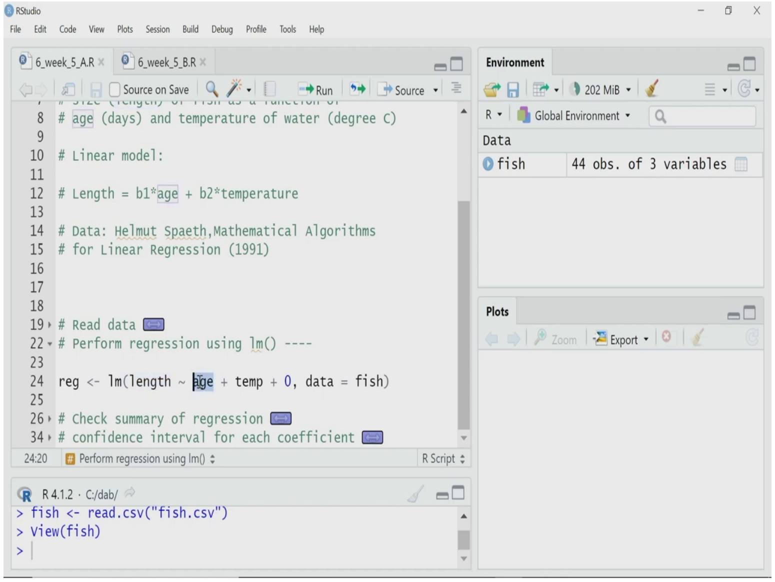


```r
# size (length) of fish as a function of
# age (days) and temperature of water (degree C)

# Linear model:

# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)



# Read data
# Perform regression using lm() ----

reg <- lm(length ~ age + temp + 0, data = fish)

# Check summary of regression
# confidence interval for each coefficient
```

Data

fish	44 obs. of 3 variables

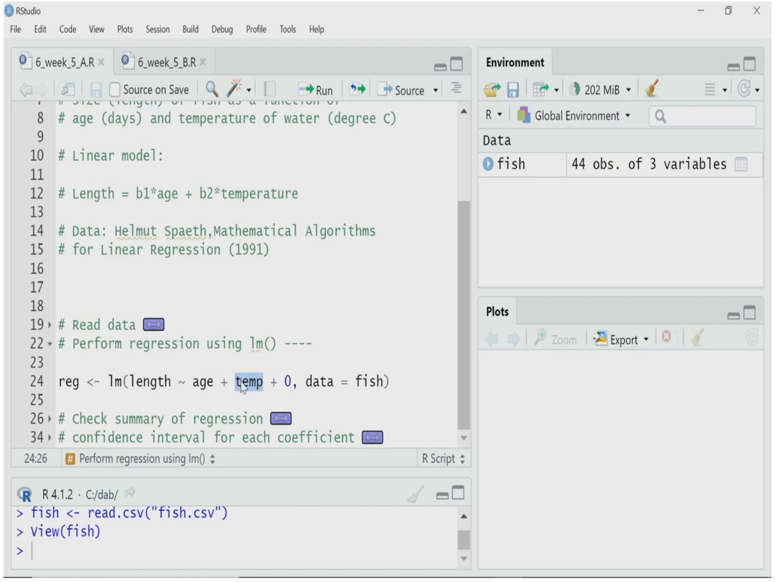
```
> fish <- read.csv("fish.csv")
> View(fish)
>
```



```
# age (days) and temperature of water (degree C)

# Linear model:

# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)



# Read data
# Perform regression using lm() ----

reg <- lm(length ~ age + temp + 0, data = fish)

# Check summary of regression
# confidence interval for each coefficient
```

```
> fish <- read.csv("fish.csv")
> View(fish)
>
```

Environment: fish 44 obs. of 3 variables



```
# age (days) and temperature of water (degree C)

# Linear model:

# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)



# Read data
# Perform regression using lm() ----

reg <- lm(length ~ age + temp + 0, data = fish)

# Check summary of regression
# confidence interval for each coefficient
```

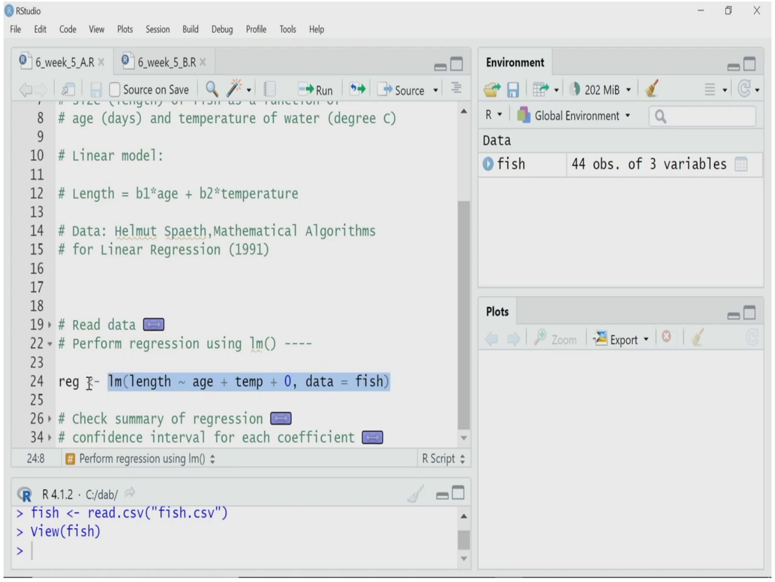
```
> fish <- read.csv("fish.csv")
> View(fish)
>
```

Environment: fish 44 obs. of 3 variables

reg ← lm(length ~ age + temp + 0, data = fish)

This is how I do that. So, I am calling lm function. As an argument the first argument is I am defining the model. So, I am writing length tilde age plus temperature plus 0. So, by defining this way, I am telling the lm function that see length is my dependent variable. Age is one of the independent variable or the predictor. Temperature is another independent variable or predictor, and I do not have any intercept in my model, I am asking you to set the intercept as equal to 0. And what is the data for this model? That data is my fish data. So, data equal to fish. The first variable where I have stored all the data after reading the csv file.

(Refer Slide Time: 05:05)

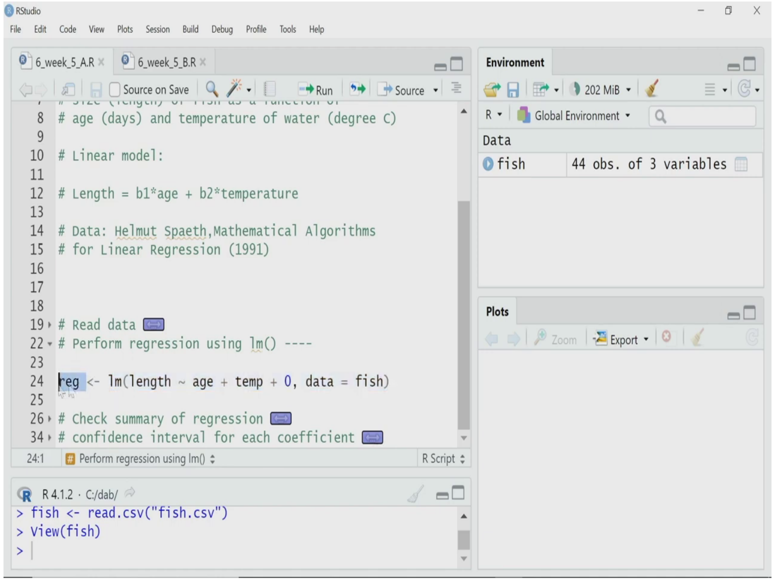


```
# age (days) and temperature of water (degree C)

# Linear model:

# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)



# Read data
# Perform regression using lm() ----

reg <- lm(length ~ age + temp + 0, data = fish)

# Check summary of regression
# confidence interval for each coefficient
```

```
> fish <- read.csv("fish.csv")
> View(fish)
>
```

Data

fish 44 obs. of 3 variables



```
# age (days) and temperature of water (degree C)

# Linear model:

# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)



# Read data
# Perform regression using lm() ----

reg <- lm(length ~ age + temp + 0, data = fish)

# Check summary of regression
# confidence interval for each coefficient
```

```
> fish <- read.csv("fish.csv")
> View(fish)
>
```

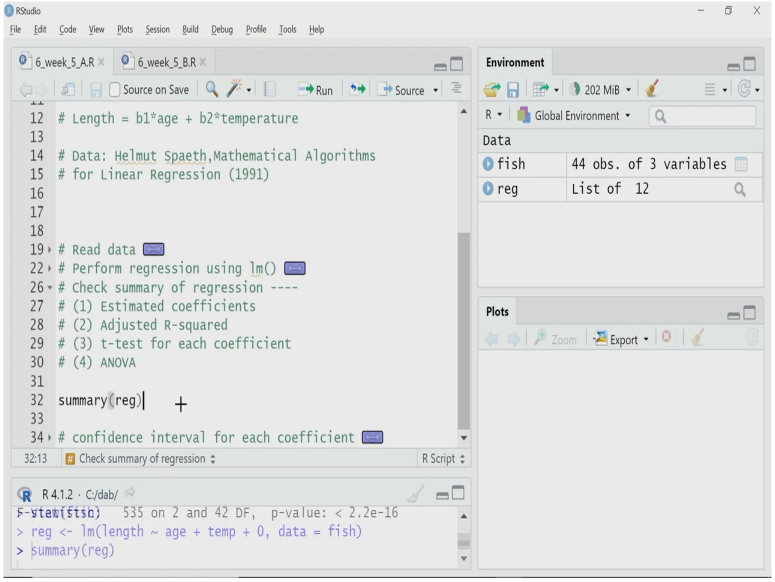
Data

fish 44 obs. of 3 variables

reg ← lm(length ~ age + temp + 0, data = fish)

And all the regression result will be assigned to a variable reg or r e g. I have performed the regression. Now, let us use the summary function to check the summary of that regression result. As we know, by calling the summary, I can look into the value of the coefficient estimated values of the coefficient, I can check the t test data, ANOVA data as well as the R square data all these things which I require to understand how good is my model.

(Refer Slide Time: 05:38)



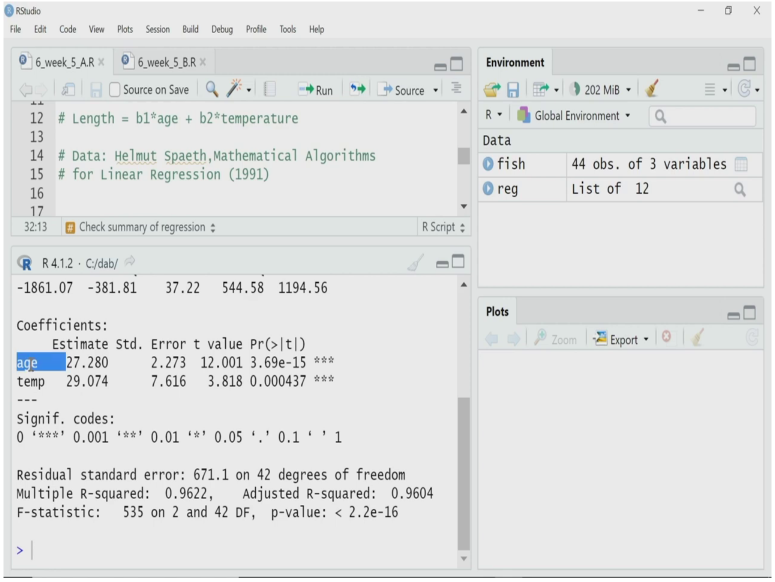
```
# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)



# Read data
# Perform regression using lm()
# Check summary of regression ----
# (1) Estimated coefficients
# (2) Adjusted R-squared
# (3) t-test for each coefficient
# (4) ANOVA

summary(reg)

# confidence interval for each coefficient
```

```
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16
> reg <- lm(length ~ age + temp + 0, data = fish)
> summary(reg)
```

Data	
fish	44 obs. of 3 variables
reg	List of 12

So, let me check the summary. So, I will call the summary function and reg the variable which is storing all the regression data right now will be used as an argument.

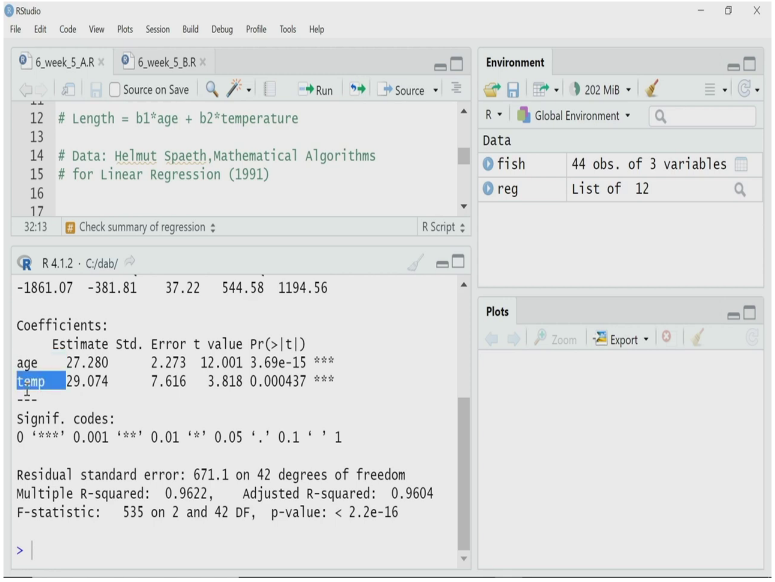
(Refer Slide Time: 05:51)



```
# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)
```

```
-1861.07  -381.81    37.22   544.58  1194.56

Coefficients:
     Estimate Std. Error t value Pr(>|t|)    
age    27.280      2.273  12.001 3.69e-15 ***
temp   29.074      7.616   3.818 0.000437 ***
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 671.1 on 42 degrees of freedom
Multiple R-squared:  0.9622,    Adjusted R-squared:  0.9604 
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16

> 
```

Data	
fish	44 obs. of 3 variables
reg	List of 12

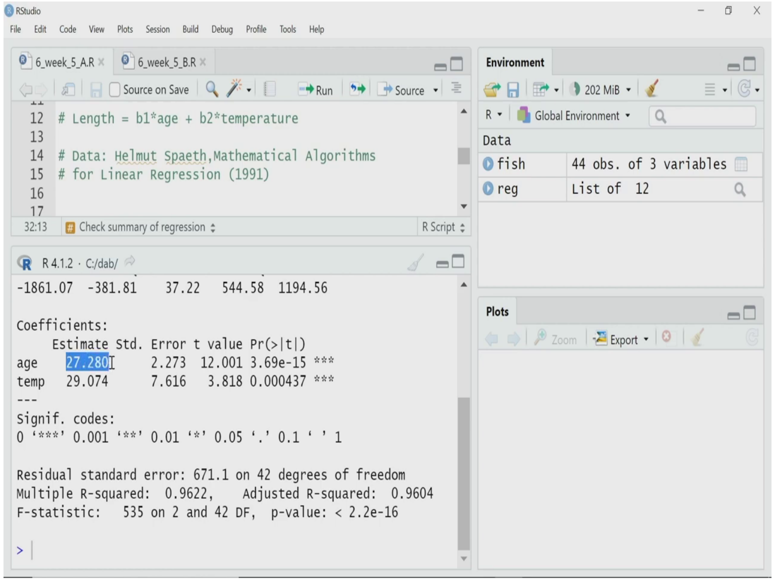


```
# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)
```

```
-1861.07  -381.81    37.22   544.58  1194.56

Coefficients:
     Estimate Std. Error t value Pr(>|t|)
age    27.280      2.273  12.001 3.69e-15 ***
temp   29.074      7.616   3.818 0.000437 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 671.1 on 42 degrees of freedom
Multiple R-squared:  0.9622,    Adjusted R-squared:  0.9604
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16

>
```

Data

fish	44 obs. of 3 variables
reg	List of 12

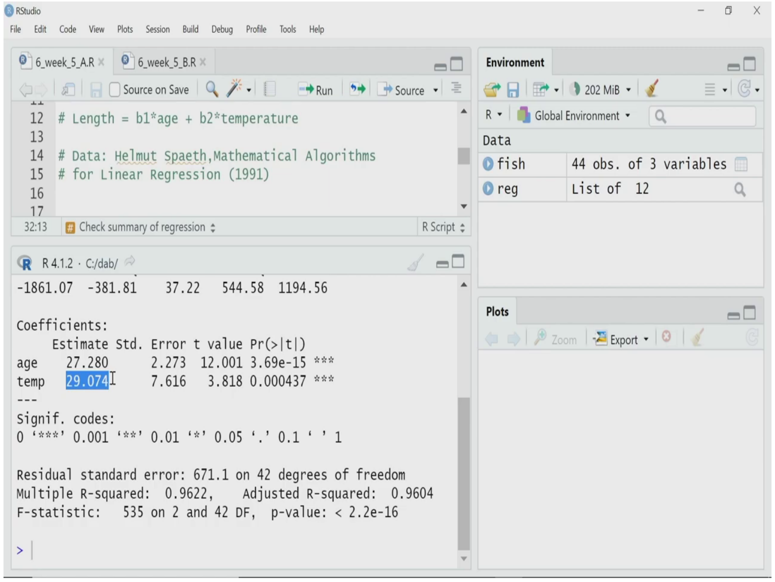


```
# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)
```

```
-1861.07  -381.81    37.22   544.58  1194.56

Coefficients:
     Estimate Std. Error t value Pr(>|t|)
age    27.280      2.273  12.001 3.69e-15 ***
temp   29.074      7.616   3.818 0.000437 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 671.1 on 42 degrees of freedom
Multiple R-squared:  0.9622,    Adjusted R-squared:  0.9604
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16

>
```

Data

fish	44 obs. of 3 variables
reg	List of 12



```
# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)
```

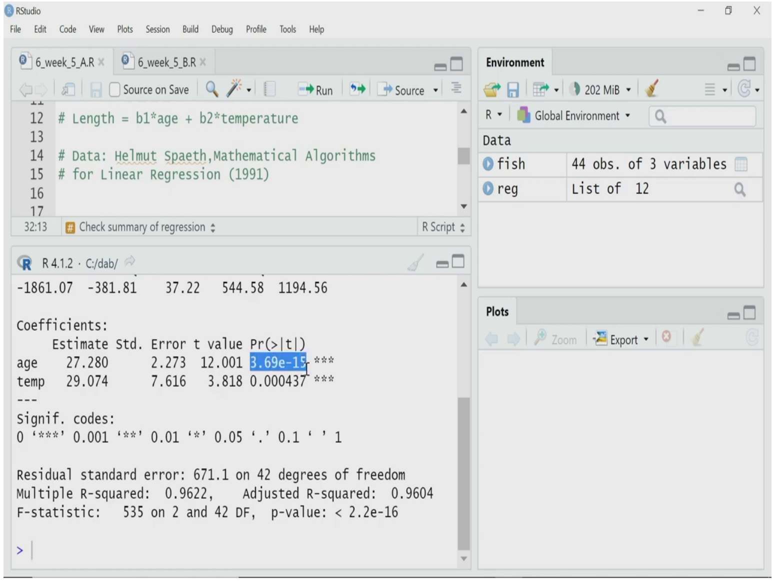
```
-1861.07  -381.81    37.22   544.58  1194.56

Coefficients:
     Estimate Std. Error t value Pr(>|t|)    
age    27.280      2.273  12.001 3.69e-15 ***
temp   29.074      7.616   3.818 0.000437 ***
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 671.1 on 42 degrees of freedom
Multiple R-squared:  0.9622,	Adjusted R-squared:  0.9604 
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16

> 
```

Data
fish 44 obs. of 3 variables
reg List of 12

summary(reg)

Let me, expand that console fine. Remember, in this model, I have specified that there is no intercept I have said intercept equal to 0. So, lm has not calculated intercept, it has calculated only 2 coefficients, 1 coefficient for age the other coefficient for temperature. For age the estimated value of the coefficient is 27.28. Whereas for temperature is 29.074. So, my model is the length of the fish is equal to 27.28 into its age plus 29 into the temperature of the water.

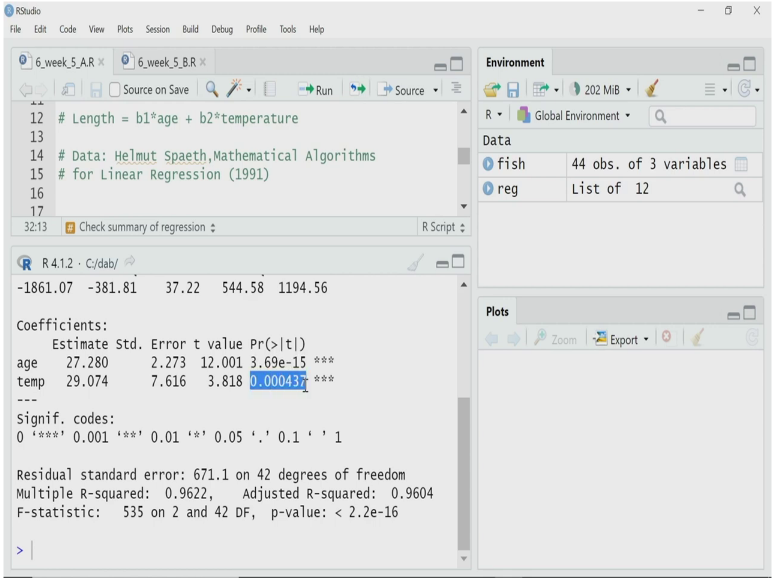
(Refer Slide Time: 06:31)



```
# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)
```

```
-1861.07  -381.81    37.22   544.58  1194.56

Coefficients:
     Estimate Std. Error t value Pr(>|t|)    
age    27.280      2.273  12.001 3.69e-15 ***
temp   29.074      7.616   3.818 0.000437 ***
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 671.1 on 42 degrees of freedom
Multiple R-squared:  0.9622,	Adjusted R-squared:  0.9604 
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16

> 
```

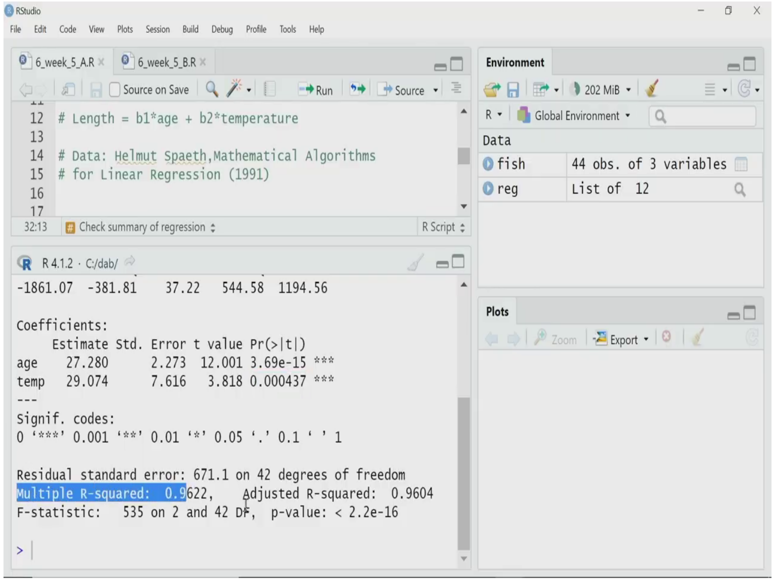
Data
fish 44 obs. of 3 variables
reg List of 12



```
# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)
```

```
Data
fish     44 obs. of 3 variables
reg      List of  12
```

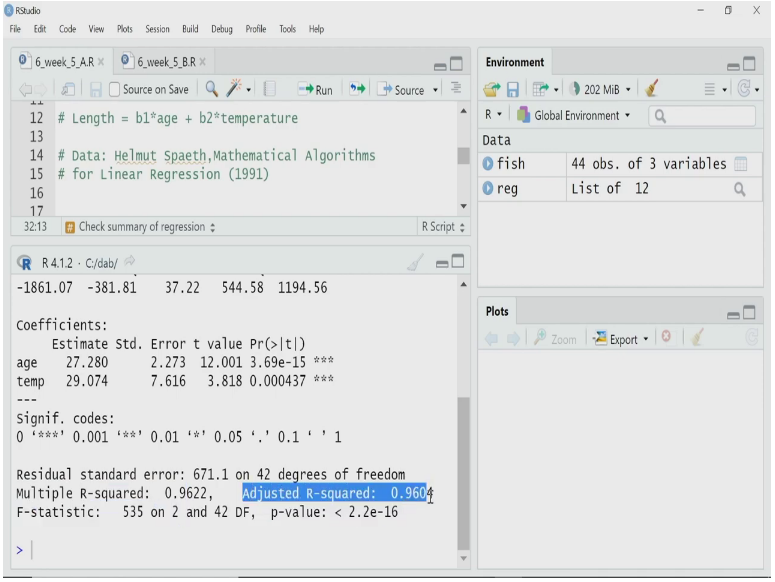
```
-1861.07  -381.81    37.22   544.58  1194.56

Coefficients:
     Estimate Std. Error t value Pr(>|t|)    
age    27.280      2.273  12.001 3.69e-15 ***
temp   29.074      7.616   3.818 0.000437 ***
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 671.1 on 42 degrees of freedom
Multiple R-squared:  0.9622,	Adjusted R-squared:  0.9604 
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16

> 
```



```
# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)
```

```
Data
fish     44 obs. of 3 variables
reg      List of  12
```

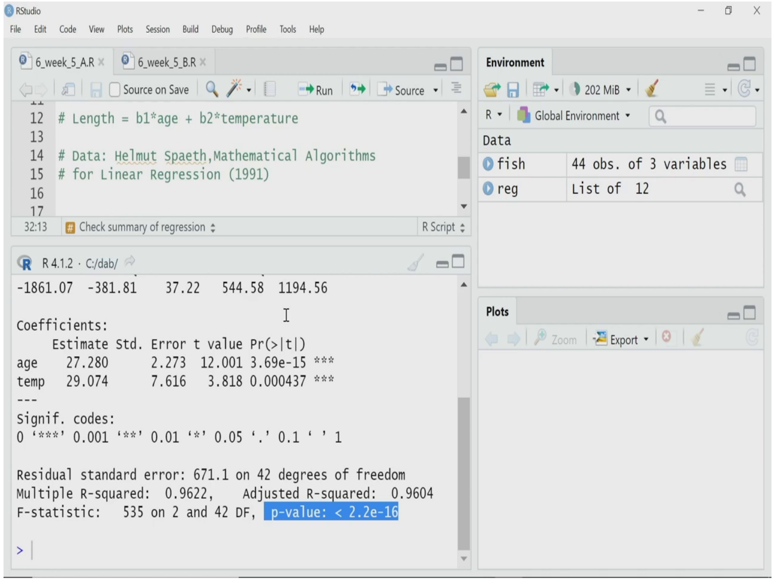
```
-1861.07  -381.81    37.22   544.58  1194.56

Coefficients:
     Estimate Std. Error t value Pr(>|t|)    
age    27.280      2.273  12.001 3.69e-15 ***
temp   29.074      7.616   3.818 0.000437 ***
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 671.1 on 42 degrees of freedom
Multiple R-squared:  0.9622,	Adjusted R-squared:  0.9604 
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16

> 
```



```
# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)
```

Data
fish	44 obs. of 3 variables
reg	List of 12

```
-1861.07  -381.81    37.22   544.58  1194.56

Coefficients:
     Estimate Std. Error t value Pr(>|t|)
age    27.280      2.273  12.001 3.69e-15 ***
temp   29.074      7.616   3.818 0.000437 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 671.1 on 42 degrees of freedom
Multiple R-squared:  0.9622,	Adjusted R-squared:  0.9604
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16

>
```



```
# Length = b1*age + b2*temperature

# Data: Helmut Spaeth,Mathematical Algorithms
# for Linear Regression (1991)
```

Data
fish	44 obs. of 3 variables
reg	List of 12

```
-1861.07  -381.81    37.22   544.58  1194.56

Coefficients:
     Estimate Std. Error t value Pr(>|t|)
age    27.280      2.273  12.001 3.69e-15 ***
temp   29.074      7.616   3.818 0.000437 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 671.1 on 42 degrees of freedom
Multiple R-squared:  0.9622,	Adjusted R-squared:  0.9604
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16

>
```

Now, it has also performed the t test as usual, and you can see the p values for these both the coefficients are very small. So, that means, I can reject the null hypothesis, and that in other words, it means that both these coefficients are statistically significant. It is also calculated the R square both the multiple R squared as well as the adjusted R square. This is a multiple linear regression I have more than one independent variable.

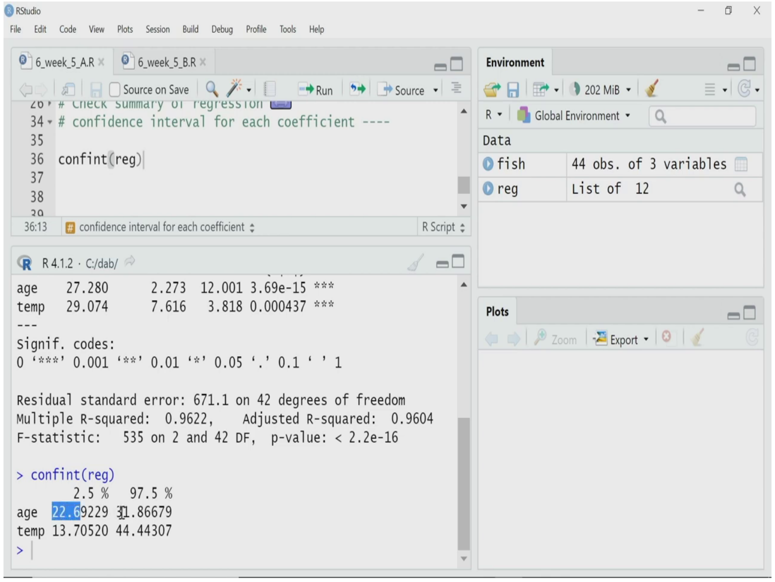
And as we know, as I keep on increasing the number of independent variable or predictors in the model, you know, always the normal R squared value will actually not be right reflection of the goodness of fit, I have to use the adjusted R square. And so, in this case, I will consider the adjusted R square which is 0.96 still it is quite good close to 1 that means, my linear model for this data is very good, very good.

Now, if you remember when we are discussing the multiple linear regression in another video, we have to discuss about the ANOVA or F test. When you have multiple independent variable you perform an ANOVA while with the regression with a particular null hypothesis, what is the null hypothesis? The null hypothesis is that all the coefficients for the independent variable, all the coefficients for the independent variables are equal to 0. Whereas, your alternate hypothesis is at least one of those coefficients is not equal to 0.

So, lm function has already performed the ANOVA the F test for this particular data set for this linear model and it has calculated the p value and it is saying the p value is less than 2.2 into 10 to the power minus 16. That means, it is a very small p value that means, I can reject the null hypothesis that means, I can say okay the null hypothesis that all the coefficients for the independent variables are equal to 0 is rejected.

That means, at least one of the coefficient must be not equal to 0 and I have already got the coefficient value and individual t test has also said that these coefficients as statistically significant that means, my model will should retain both the coefficient both the independent variable age as well as the temperature of the water. So, I have performed a regression, I have checked the statistics of that also.

(Refer Slide Time: 09:06)

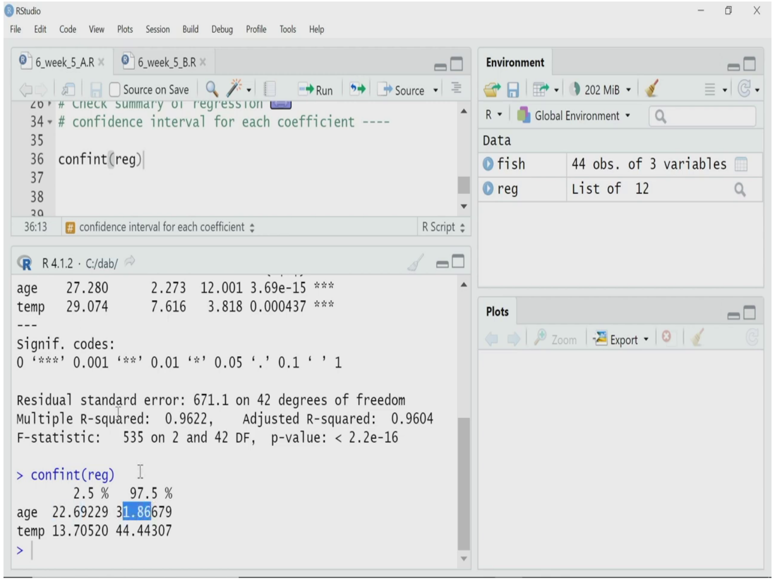


```
# check summary of regression
# confidence interval for each coefficient ----

confint(reg)
```

Data

fish	44 obs. of 3 variables
reg	List of 12

```
age     27.280       2.273  12.001 3.69e-15 ***
temp    29.074       7.616   3.818 0.000437 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 671.1 on 42 degrees of freedom
Multiple R-squared:  0.9622,    Adjusted R-squared:  0.9604
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16

> confint(reg)
        2.5 %   97.5 %
age  22.69229 31.86679
temp 13.70520 44.44307
>
```

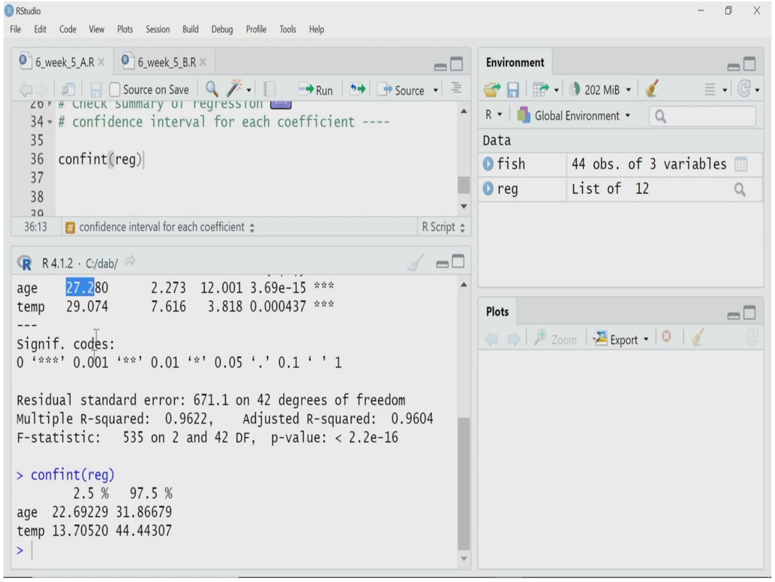


```
# check summary of regression
# confidence interval for each coefficient ----

confint(reg)
```

```
age    27.280      2.273  12.001 3.69e-15 ***
temp   29.074      7.616   3.818 0.000437 ***
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 671.1 on 42 degrees of freedom
Multiple R-squared:  0.9622,    Adjusted R-squared:  0.9604
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16

> confint(reg)
        2.5 %   97.5 %
age  22.69229 31.86679
temp 13.70520 44.44307
>
```

Data

fish	44 obs. of 3 variables
reg	List of 12

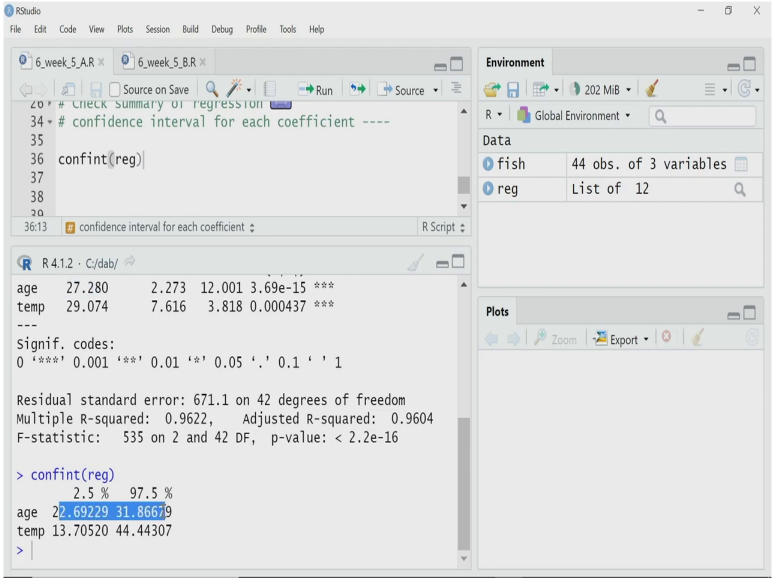


```
# check summary of regression
# confidence interval for each coefficient ----

confint(reg)
```

```
age    27.280      2.273  12.001 3.69e-15 ***
temp   29.074      7.616   3.818 0.000437 ***
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 671.1 on 42 degrees of freedom
Multiple R-squared:  0.9622,    Adjusted R-squared:  0.9604
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16

> confint(reg)
        2.5 %   97.5 %
age  22.69229 31.86679
temp 13.70520 44.44307
>
```

Data

fish	44 obs. of 3 variables
reg	List of 12



```
# check summary of regression
# confidence interval for each coefficient ----

confint(reg)
```

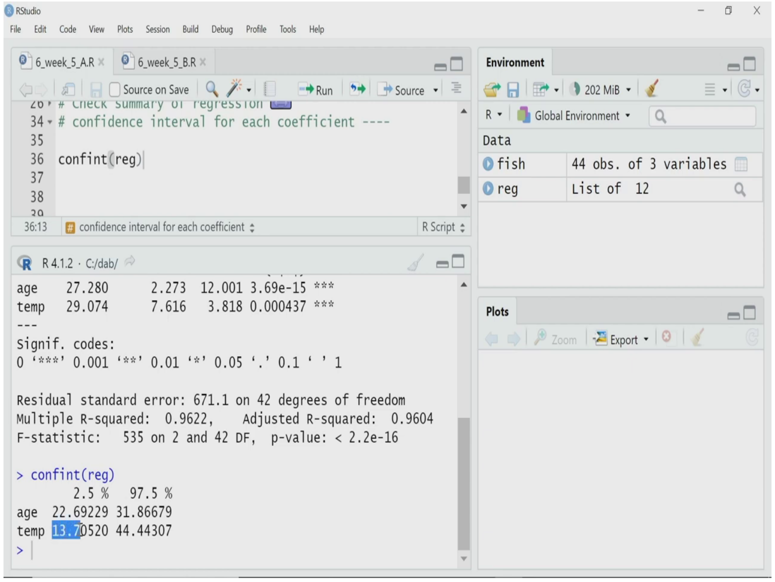
```
age     27.280      2.273  12.001 3.69e-15 ***
temp    29.074      7.616   3.818 0.000437 ***
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 671.1 on 42 degrees of freedom
Multiple R-squared:  0.9622,    Adjusted R-squared:  0.9604
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16

> confint(reg)
        2.5 %   97.5 %
age  22.69229 31.86679
temp 13.70520 44.44307
>
```

Environment: fish — 44 obs. of 3 variables; reg — List of 12

confint(reg)

Now, I want to calculate the confidence interval for each of these coefficients. So, let me calculate the confidence interval, I will use the confint function again and the reg which is storing the all the data for my regression should be my argument. And if I calculate I can say at the 95 percent level of confidence interval, the value, estimated value of the coefficient for age should vary from 22.69 to 31.86, the estimated value is 27.28. So, it should lie in between these 2, the true value should lie in between 22 to 31.

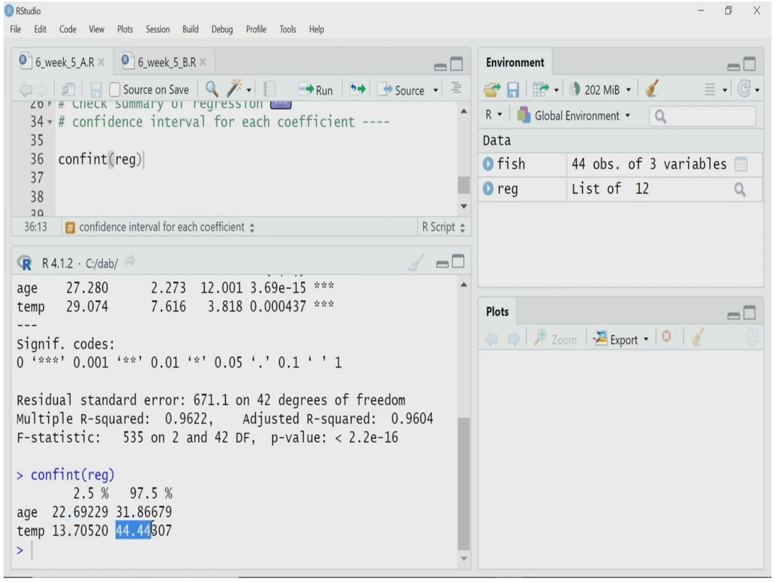
(Refer Slide Time: 09:45)



```
# check summary of regression
# confidence interval for each coefficient ----

confint(reg)
```

```
age     27.280      2.273  12.001 3.69e-15 ***
temp    29.074      7.616   3.818 0.000437 ***
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 671.1 on 42 degrees of freedom
Multiple R-squared:  0.9622,    Adjusted R-squared:  0.9604
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16

> confint(reg)
        2.5 %   97.5 %
age  22.69229 31.86679
temp 13.70520 44.44307
>
```

Environment: fish — 44 obs. of 3 variables; reg — List of 12



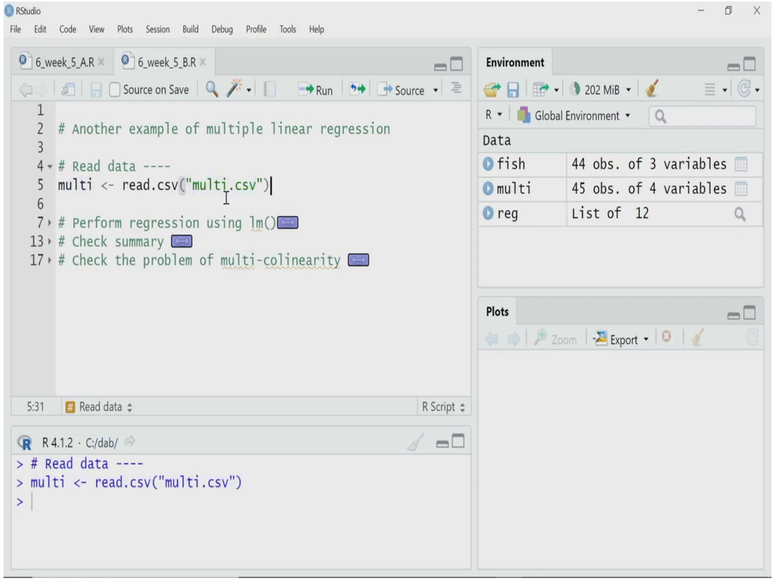
```
# check summary of regression
# confidence interval for each coefficient ----

confint(reg)
```

```
age    27.280      2.273  12.001 3.69e-15 ***
temp   29.074      7.616   3.818 0.000437 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 671.1 on 42 degrees of freedom
Multiple R-squared:  0.9622,    Adjusted R-squared:  0.9604
F-statistic:   535 on 2 and 42 DF,  p-value: < 2.2e-16

> confint(reg)
        2.5 %   97.5 %
age  22.69229 31.86679
temp 13.70520 44.44307
>
```

```
Data
fish    44 obs. of 3 variables
reg     List of  12
```

Whereas, the true value in the population level for the coefficient of temperature should lie between 13.7 to 44.44. That is all for performing a simple multiple linear regression. What I have done I have just used the lm function and specified the model I have specified which is dependent variable which are the independent variable, and in this particular model, as we are considering there is no intercept. Because the growth of the fish cannot be independent without age. If a fish has age 0 it cannot have a size. So, that is why you have set the intercept equal to 0, and then perform the linear multiple linear regression.

(Refer Slide Time: 10:43)



```

# Another example of multiple linear regression

# Read data ----
multi <- read.csv("multi.csv")

# Perform regression using lm()
# Check summary
# Check the problem of multi-colinearity
```

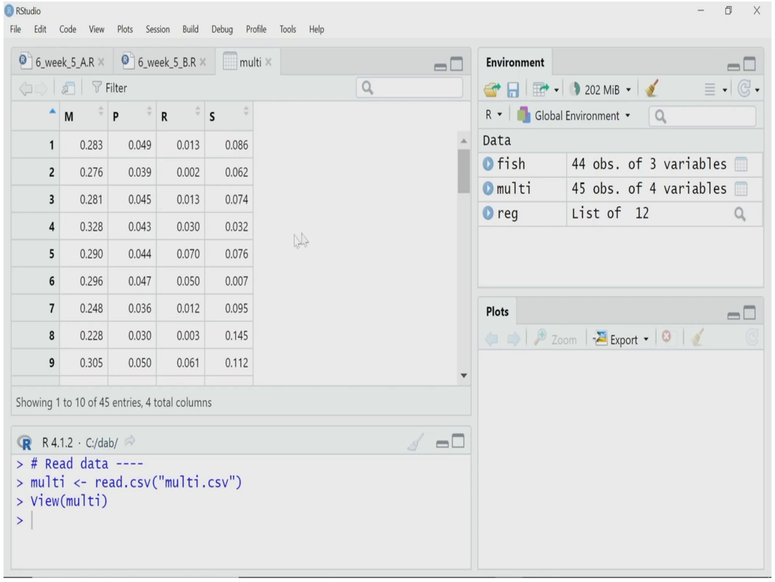
```
> # Read data ----
> multi <- read.csv("multi.csv")
>
```

```
Data
fish    44 obs. of 3 variables
multi   45 obs. of 4 variables
reg     List of  12
```

multi ← read.csv("multi.csv”)

In this lecture, I have another example where I have more than 2 independent variable, and we have to perform multiple linear regression. Let us, check that. So, I will start by reading the data. And then I will check what we have in the data and then perform the multiple linear regression using the lm function again. So, the name of this data file is multi dot csv, it is in my current working directory. So, I call the read dot csv function to read that data and assign that data to a variable called multi.

(Refer Slide Time: 10:56)



	M	P	R	S
1	0.283	0.049	0.013	0.086
2	0.276	0.039	0.002	0.062
3	0.281	0.045	0.013	0.074
4	0.328	0.043	0.030	0.032
5	0.290	0.044	0.070	0.076
6	0.296	0.047	0.050	0.007
7	0.248	0.036	0.012	0.095
8	0.228	0.030	0.003	0.145
9	0.305	0.050	0.061	0.112

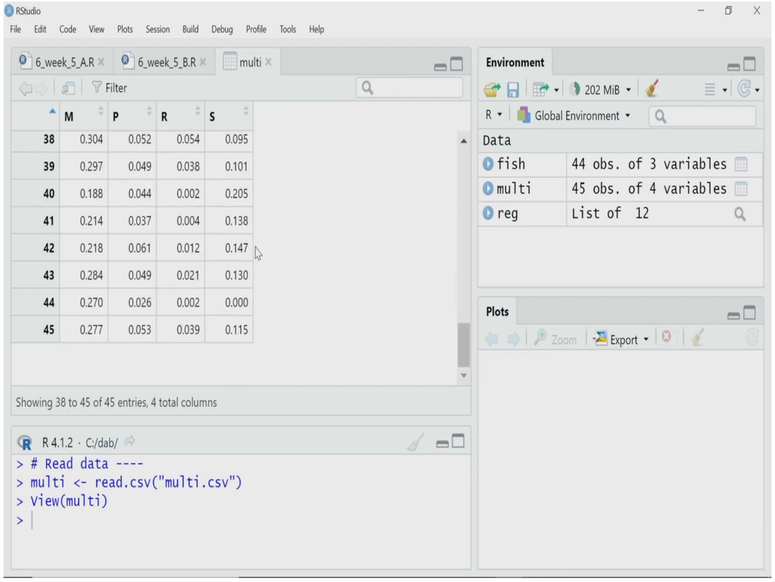
Showing 1 to 10 of 45 entries, 4 total columns

Data

fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12

R 4.1.2 · C:/dab/

```
> # Read data ----
> multi <- read.csv("multi.csv")
> View(multi)
> 
```



	M	P	R	S
38	0.304	0.052	0.054	0.095
39	0.297	0.049	0.038	0.101
40	0.188	0.044	0.002	0.205
41	0.214	0.037	0.004	0.138
42	0.218	0.061	0.012	0.147
43	0.284	0.049	0.021	0.130
44	0.270	0.026	0.002	0.000
45	0.277	0.053	0.039	0.115

Showing 38 to 45 of 45 entries, 4 total columns

Data

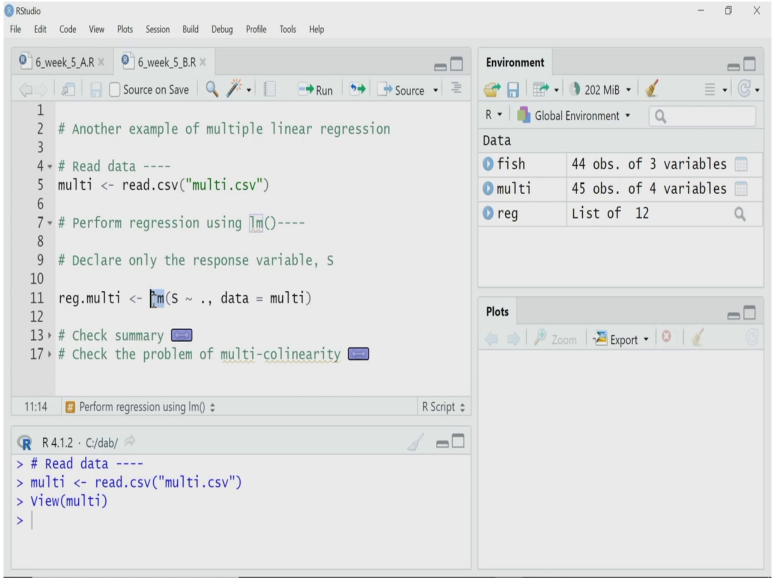
fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12

R 4.1.2 · C:/dab/

```
> # Read data ----
> multi <- read.csv("multi.csv")
> View(multi)
> 
```

Let me check the data. So, you can see I have four variable M, P, R and S and I have 45 observation 45 data points, 45 rows, and what I will do I will consider S as the dependent variable or response variable, and M, P and R as the independent variable, and I will perform multiple linear regression.

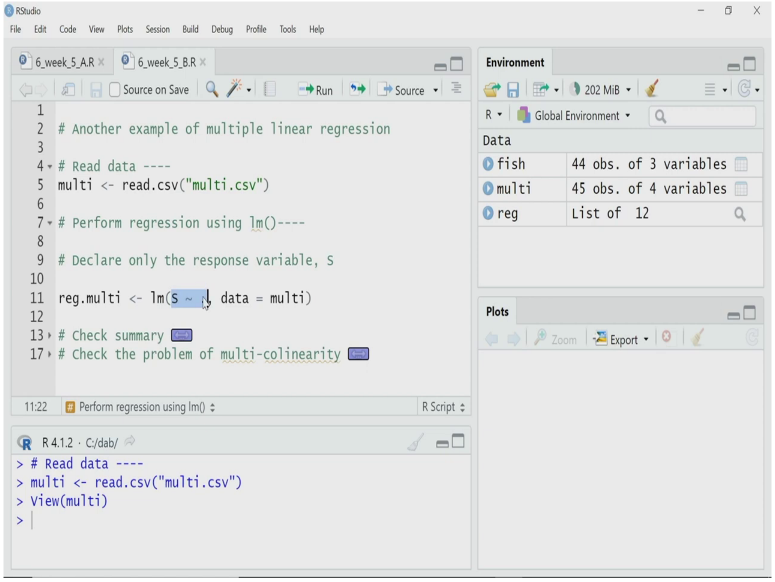
(Refer Slide Time: 11:19)



```r
# Another example of multiple linear regression

# Read data ----
multi <- read.csv("multi.csv")

# Perform regression using lm()----

# Declare only the response variable, S

reg.multi <- lm(S ~ ., data = multi)

# Check summary
# Check the problem of multi-colinearity
```

Data	
fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12

```
> # Read data ----
> multi <- read.csv("multi.csv")
> View(multi)
>
```



```r
# Another example of multiple linear regression

# Read data ----
multi <- read.csv("multi.csv")

# Perform regression using lm()----

# Declare only the response variable, S

reg.multi <- lm(S ~ ., data = multi)

# Check summary
# Check the problem of multi-colinearity
```

Data	
fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12

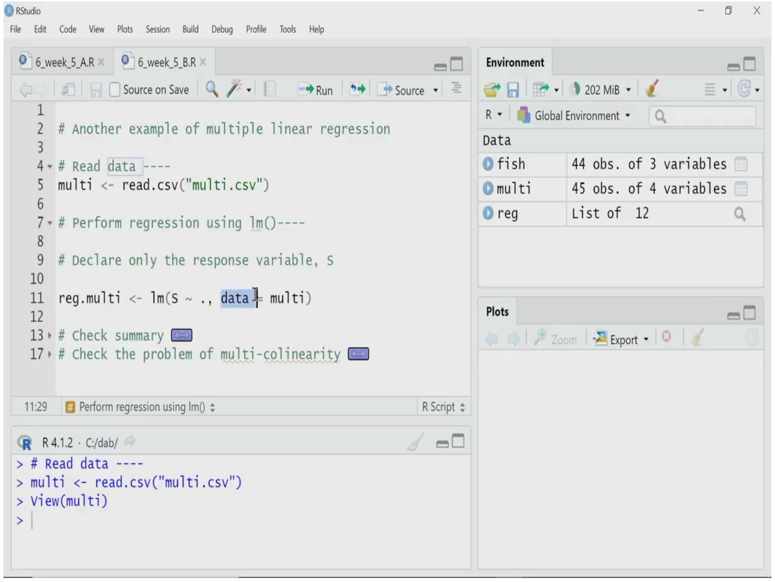
```
> # Read data ----
> multi <- read.csv("multi.csv")
> View(multi)
>
```

reg.multi ← lm(S ~ ., data = multi)

So, to do that, I will use the lm function. Now in this case, we will consider intercept and also, I will play a trick the way I call lm. See, when I am using these lm function, to perform multiple linear regression, if I have 2 dependent variable, it is very easy to write them, I can spell out for example, previous example I said, age plus temp. But if I have suppose 10, 20 independent variables, then you have to type all those that does not make sense. So, there is a shortcut to do that. And that is what I am showing here.

I am calling the lm function and the first argument is this s tilde and dot, what do I am telling here? I am telling to lm function that I am specifying that see s is my dependent variable, and after tilde I have not written anything but dot, by dot I am saying consider all the variable present, other variable present in this data set, consider all other variable present in the data set, and the intercept also, as on the right hand side, by putting that dot, I am assigning all of those, I have not to spell them out separately, that is a easy shortcut.

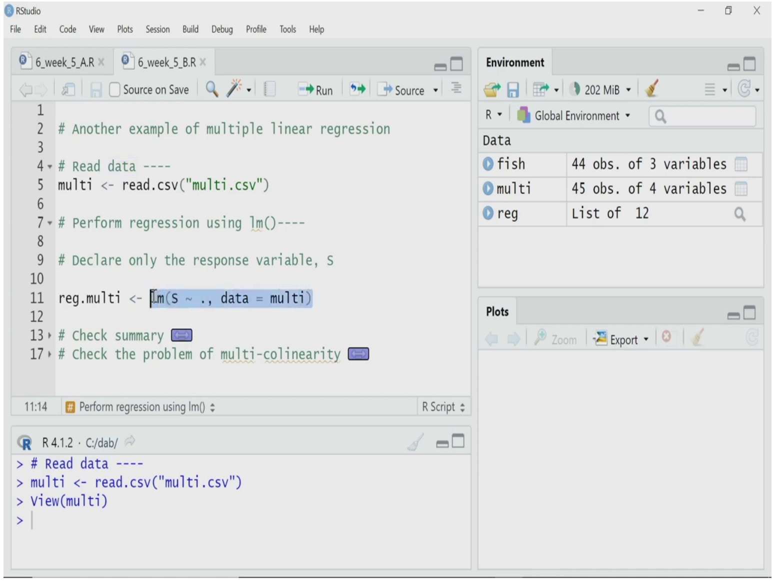
(Refer Slide Time: 12:36)



```
# Another example of multiple linear regression

# Read data ----
multi <- read.csv("multi.csv")

# Perform regression using lm()----

# Declare only the response variable, S

reg.multi <- lm(S ~ ., data = multi)

# Check summary
# Check the problem of multi-colinearity
```

```
> # Read data ----
> multi <- read.csv("multi.csv")
> View(multi)
> 
```

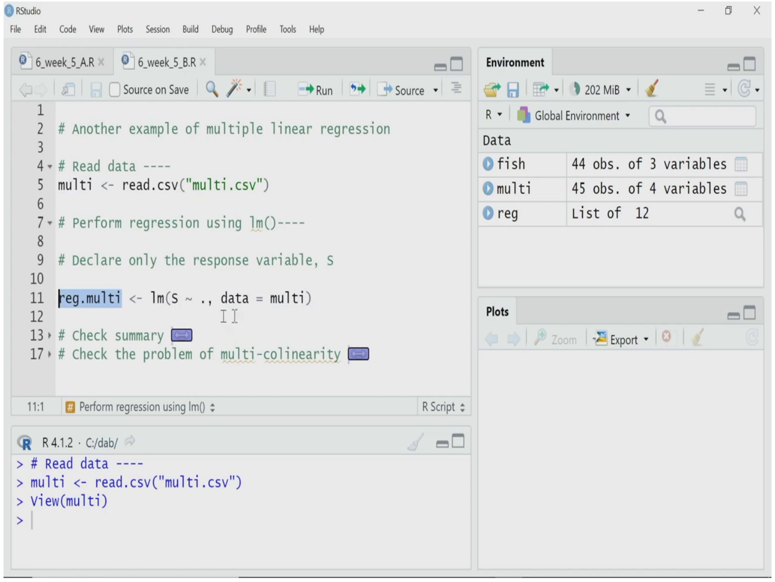
Data	
fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12



```
# Another example of multiple linear regression

# Read data ----
multi <- read.csv("multi.csv")

# Perform regression using lm()----

# Declare only the response variable, S

reg.multi <- lm(S ~ ., data = multi)

# Check summary
# Check the problem of multi-colinearity
```

```
> # Read data ----
> multi <- read.csv("multi.csv")
> View(multi)
>
```

Data	
fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12



```
# Another example of multiple linear regression

# Read data ----
multi <- read.csv("multi.csv")

# Perform regression using lm()----

# Declare only the response variable, S

reg.multi <- lm(S ~ ., data = multi)

# Check summary
# Check the problem of multi-colinearity
```

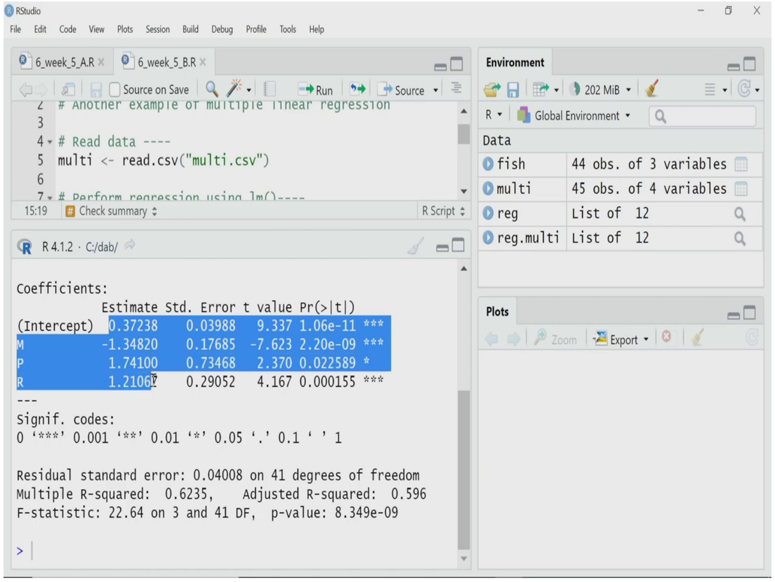
```
> # Read data ----
> multi <- read.csv("multi.csv")
> View(multi)
>
```

Data	
fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12

reg.multi ← lm(S ~ ., data = multi)

And then I am specifying the data, data equal to multi and I want to store or assign all these data regression to a variable called reg dot multi. I perform that and now I will check the summary of my regression using the summary function.

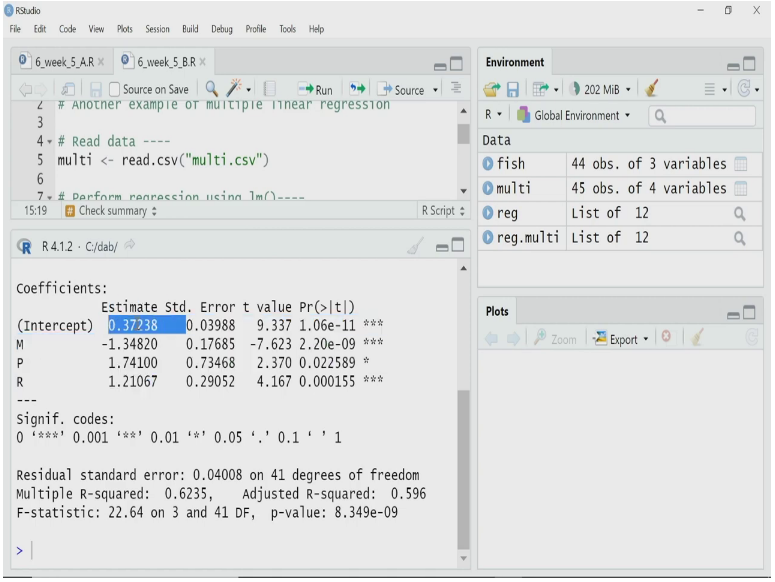
(Refer Slide Time: 12:54)



```
# Another example of multiple linear regression

# Read data ----
multi <- read.csv("multi.csv")

# Perform regression using lm()----
```

```
Data
fish        44 obs. of 3 variables
multi       45 obs. of 4 variables
reg         List of  12
reg.multi   List of  12
```

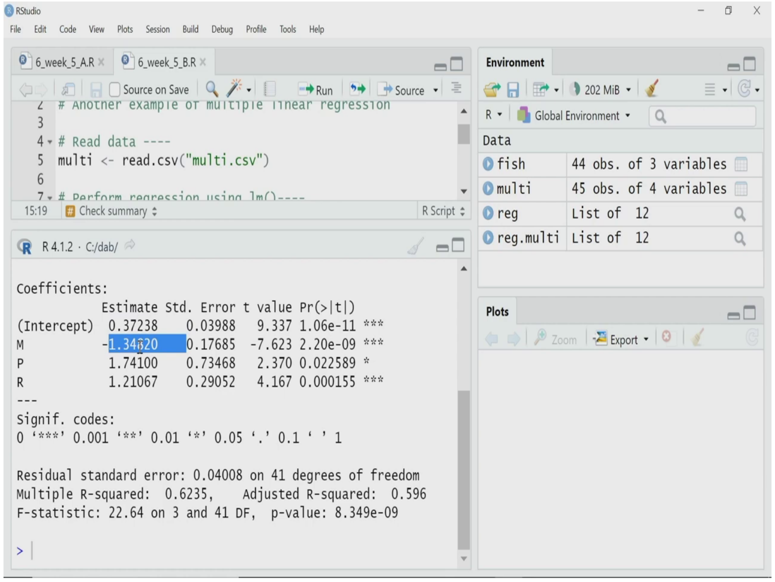
```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.37238    0.03988   9.337 1.06e-11 ***
M           -1.34820    0.17685  -7.623 2.20e-09 ***
P            1.74100    0.73468   2.370 0.022589 *  
R            1.21067    0.29052   4.167 0.000155 ***
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.04008 on 41 degrees of freedom
Multiple R-squared:  0.6235,	Adjusted R-squared:  0.596 
F-statistic: 22.64 on 3 and 41 DF,  p-value: 8.349e-09

> 
```



```
# Another example of multiple linear regression

# Read data ----
multi <- read.csv("multi.csv")

# Perform regression using lm()----
```

```
Data
fish        44 obs. of 3 variables
multi       45 obs. of 4 variables
reg         List of  12
reg.multi   List of  12
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.37238    0.03988   9.337 1.06e-11 ***
M           -1.34820    0.17685  -7.623 2.20e-09 ***
P            1.74100    0.73468   2.370 0.022589 *  
R            1.21067    0.29052   4.167 0.000155 ***
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.04008 on 41 degrees of freedom
Multiple R-squared:  0.6235,	Adjusted R-squared:  0.596 
F-statistic: 22.64 on 3 and 41 DF,  p-value: 8.349e-09

> 
```

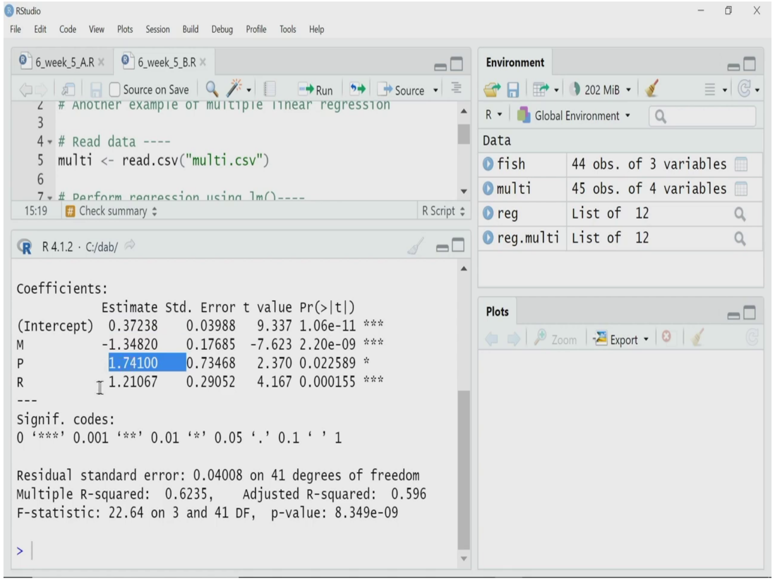


```r
# Another example of multiple linear regression

# Read data ----
multi <- read.csv("multi.csv")

# Perform regression using lm()----
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.37238    0.03988   9.337 1.06e-11 ***
M           -1.34820    0.17685  -7.623 2.20e-09 ***
P            1.74100    0.73468   2.370 0.022589 *  
R            1.21067    0.29052   4.167 0.000155 ***
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.04008 on 41 degrees of freedom
Multiple R-squared:  0.6235,	Adjusted R-squared:  0.596 
F-statistic: 22.64 on 3 and 41 DF,  p-value: 8.349e-09

> 
```

Data

fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12
reg.multi	List of 12

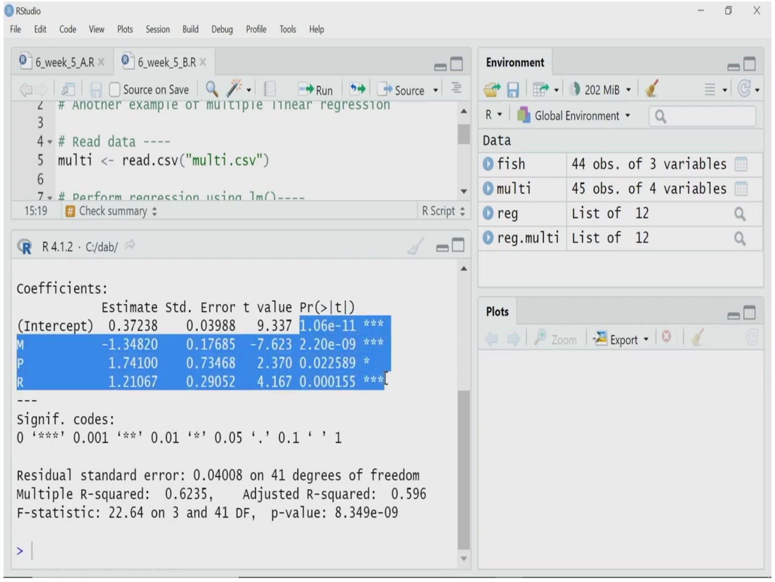


```r
# Another example of multiple linear regression

# Read data ----
multi <- read.csv("multi.csv")

# Perform regression using lm()----
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.37238    0.03988   9.337 1.06e-11 ***
M           -1.34820    0.17685  -7.623 2.20e-09 ***
P            1.74100    0.73468   2.370 0.022589 *  
R            1.21067    0.29052   4.167 0.000155 ***
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.04008 on 41 degrees of freedom
Multiple R-squared:  0.6235,	Adjusted R-squared:  0.596 
F-statistic: 22.64 on 3 and 41 DF,  p-value: 8.349e-09

> 
```

Data

fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12
reg.multi	List of 12



```
# Another example of multiple linear regression

# Read data ----
multi <- read.csv("multi.csv")

# Perform regression using lm()----
```

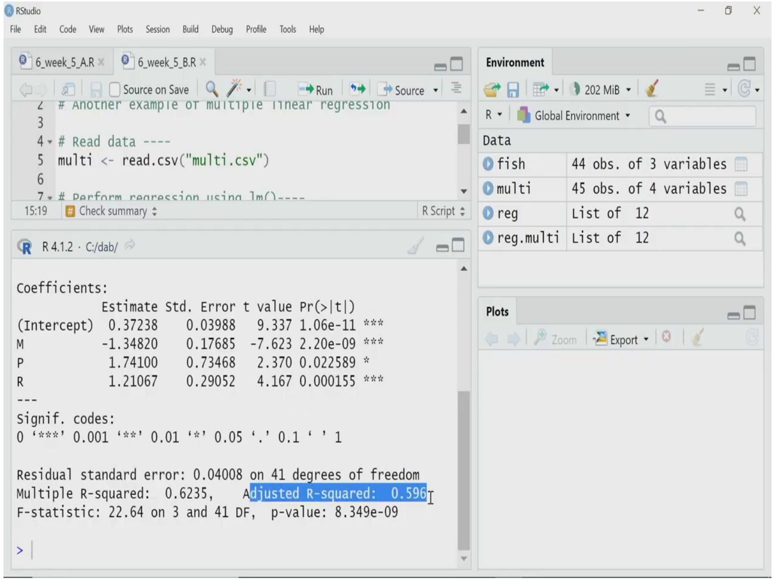
Data	
fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12
reg.multi	List of 12

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.37238    0.03988   9.337 1.06e-11 ***
M           -1.34820    0.17685  -7.623 2.20e-09 ***
P            1.74100    0.73468   2.370 0.022589 *
R            1.21067    0.29052   4.167 0.000155 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.04008 on 41 degrees of freedom
Multiple R-squared:  0.6235,	Adjusted R-squared:  0.596
F-statistic: 22.64 on 3 and 41 DF,  p-value: 8.349e-09

>
```

summary(reg.multi)

So, the first column here are the values, estimated value the coefficient the intercept is 0.37, the M is, coefficient for M is minus 1.34, the coefficient for P is 1.7 and the coefficient for R is 1.2. And if you see the t test data, all of them has low P value. So that means all of them at certain level of significance are statistically significant. That means each of these independent variable has effect on the response variable that is S.

(Refer Slide Time: 13:31)

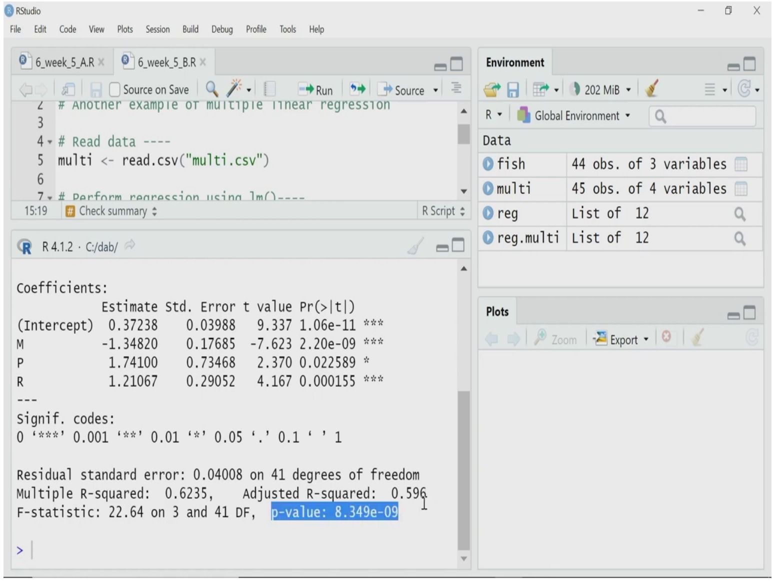


```
# Another example of multiple linear regression

# Read data ----
multi <- read.csv("multi.csv")

# Perform regression using lm()----
```

Data

fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12
reg.multi	List of 12

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.37238    0.03988   9.337 1.06e-11 ***
M           -1.34820    0.17685  -7.623 2.20e-09 ***
P            1.74100    0.73468   2.370 0.022589 *
R            1.21067    0.29052   4.167 0.000155 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.04008 on 41 degrees of freedom
Multiple R-squared:  0.6235,    Adjusted R-squared:  0.596
F-statistic: 22.64 on 3 and 41 DF,  p-value: 8.349e-09

>
```

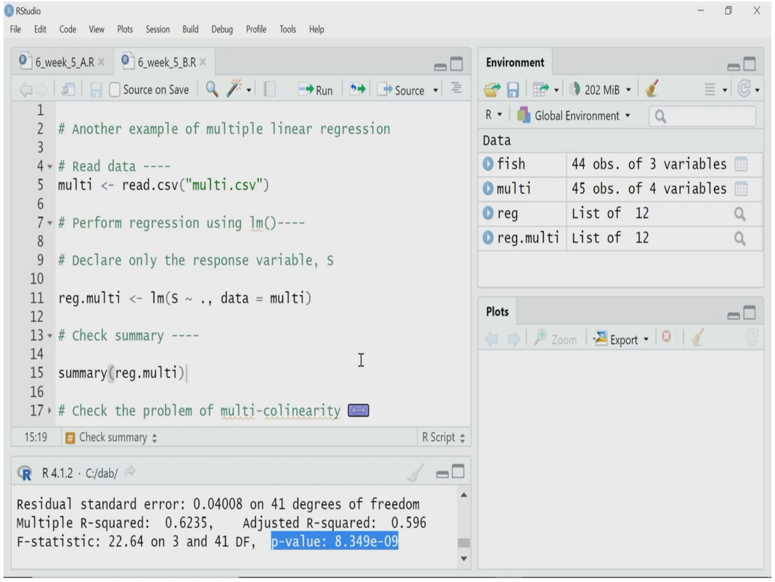


```
# Another example of multiple linear regression

# Read data ----
multi <- read.csv("multi.csv")

# Perform regression using lm()----
```

Data

fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12
reg.multi	List of 12

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.37238    0.03988   9.337 1.06e-11 ***
M           -1.34820    0.17685  -7.623 2.20e-09 ***
P            1.74100    0.73468   2.370 0.022589 *
R            1.21067    0.29052   4.167 0.000155 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.04008 on 41 degrees of freedom
Multiple R-squared:  0.6235,    Adjusted R-squared:  0.596
F-statistic: 22.64 on 3 and 41 DF,  p-value: 8.349e-09

>
```



```
# Another example of multiple linear regression

# Read data ----
multi <- read.csv("multi.csv")

# Perform regression using lm()----

# Declare only the response variable, S

reg.multi <- lm(S ~ ., data = multi)

# Check summary ----

summary(reg.multi)

# Check the problem of multi-colinearity
```

Data

fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12
reg.multi	List of 12

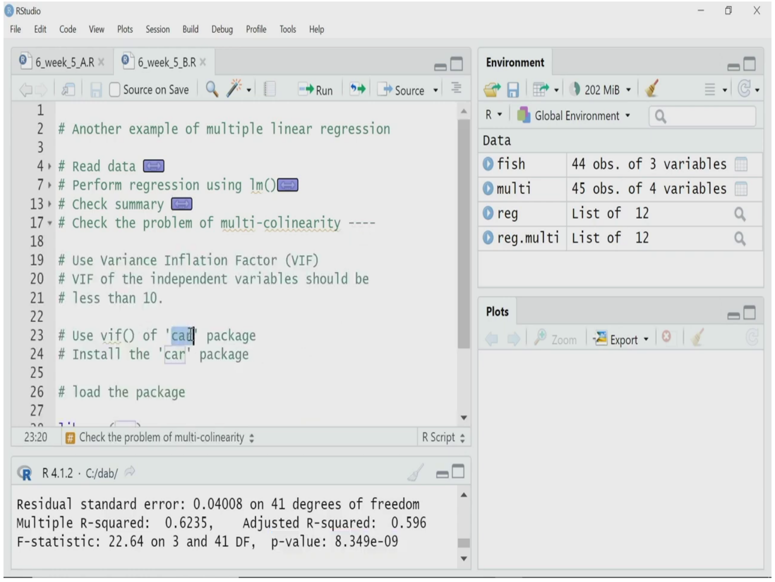
```
Residual standard error: 0.04008 on 41 degrees of freedom
Multiple R-squared:  0.6235,    Adjusted R-squared:  0.596
F-statistic: 22.64 on 3 and 41 DF,  p-value: 8.349e-09
```

And also, it has calculated the adjusted R square that is quite decent 0.59 and 0.6 and the P value for ANOVA is also very small. That means I can reject the null hypothesis for ANOVA that all these coefficients are equal to 0. No, they are not equal to 0 and the t test has already set individually they are statistically significant.

Now, if you remember the lecture of our multiple linear regression, in that lecture, we have discussed a very important point that is the problem of multicollinearity. You may have that in your data set that some of the independent variables may have relationship among themselves. So, maybe one of the independent variable may be represented as a linear combination of one or more other independent variables that can happen.

So, we have discussed at length about this multicollinearity problem in this lecture, if you do not remember please go back and check that and when you are performing a multiple linear regression, you are supposed to check whether you are facing this multicollinearity problem or not. There are many way to do that and in that lecture, we have discussed about it one way to do that, to check the multicollinearity problem whether you have that problem in your data set or not, is to use the VIF Variance Inflation Factor. So, I have discussed the definition of variance inflation factor in that lecture, what I will do here, I will show how in R I can calculate the VIF for this particular data set and check whether I have multicollinearity problem in this data set or not.

(Refer Slide Time: 15:17)

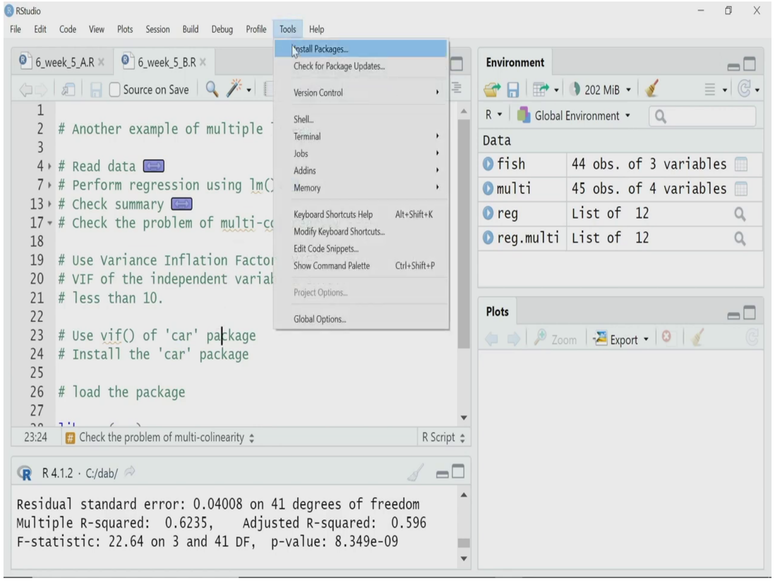


```
# Another example of multiple linear regression

# Read data
# Perform regression using lm()
# Check summary
# Check the problem of multi-colinearity ----

# Use Variance Inflation Factor (VIF)
# VIF of the independent variables should be
# less than 10.

# Use vif() of 'car' package
# Install the 'car' package

# load the package
```

```
Residual standard error: 0.04008 on 41 degrees of freedom
Multiple R-squared:  0.6235,	Adjusted R-squared:  0.596
F-statistic: 22.64 on 3 and 41 DF,  p-value: 8.349e-09
```

Data

fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12
reg.multi	List of 12

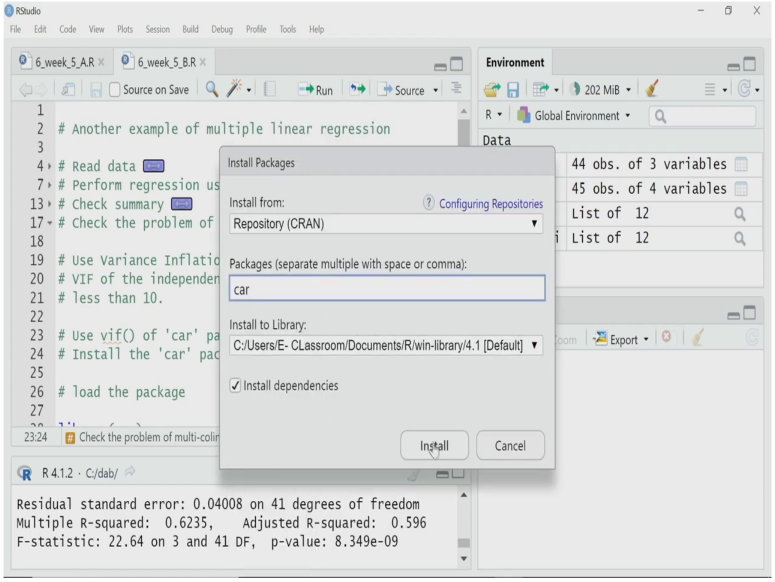


```
# Another example of multiple

# Read data
# Perform regression using lm()
# Check summary
# Check the problem of multi-co
# Use Variance Inflation Factor
# VIF of the independent varial
# less than 10.

# Use vif() of 'car' package
# Install the 'car' package

# load the package
```

```
Residual standard error: 0.04008 on 41 degrees of freedom
Multiple R-squared:  0.6235,	Adjusted R-squared:  0.596
F-statistic: 22.64 on 3 and 41 DF,  p-value: 8.349e-09
```

Data

fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12
reg.multi	List of 12



```
# Another example of multiple linear regression

# Read data
# Perform regression us
# Check summary
# Check the problem of

# Use Variance Inflatio
# VIF of the independen
# less than 10.

# Use vif() of 'car' pa
# Install the 'car' pac

# load the package
```

Install Packages

Install from: Repository (CRAN)

Packages (separate multiple with space or comma): car

Install to Library: C:/Users/E- CLassroom/Documents/R/win-library/4.1 [Default]

☑ Install dependencies

Install | Cancel

Data

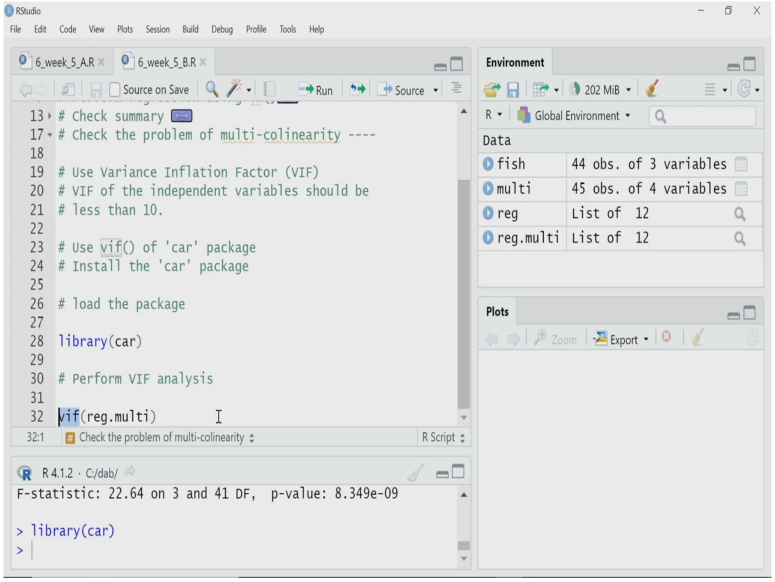
44 obs. of 3 variables
45 obs. of 4 variables
List of 12
List of 12

R 4.1.2 · C:/dab/

```
Residual standard error: 0.04008 on 41 degrees of freedom
Multiple R-squared:  0.6235,    Adjusted R-squared:  0.596
F-statistic: 22.64 on 3 and 41 DF,  p-value: 8.349e-09
```

To do that I require a particular package, that package is called car package, it has lots of useful tool, I have already installed that you can easily install that in R studio by going to tools and then clicking install package and writing here as car and then select and then you install. I have already installed so, will not install it right now. What I have to do? I have already installed so I have to call that package and load it in my this working space here. So, I will use the library function library car to load that library, I have loaded it.

(Refer Slide Time: 15:50)



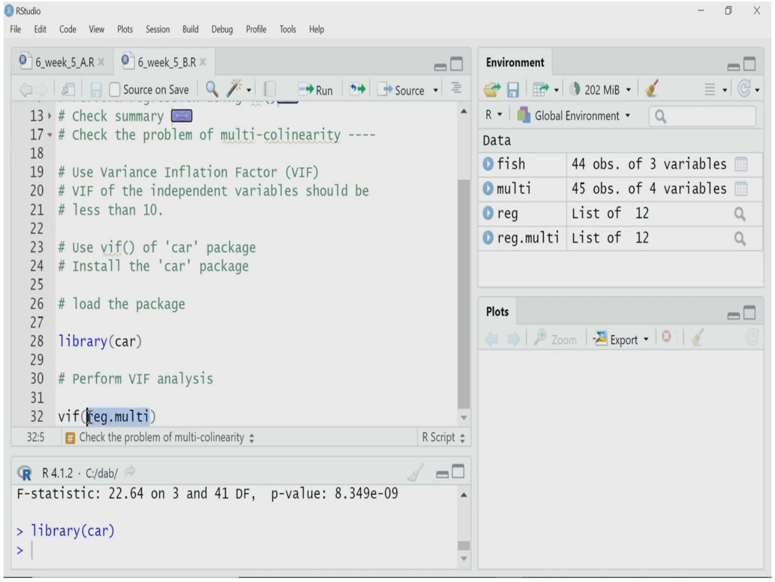
```
# Check summary
# Check the problem of multi-colinearity ----

# Use Variance Inflation Factor (VIF)
# VIF of the independent variables should be
# less than 10.

# Use vif() of 'car' package
# Install the 'car' package

# load the package

library(car)

# Perform VIF analysis

vif(reg.multi)
```

Data

fish	44 obs. of 3 variables
multi	45 obs. of 4 variables
reg	List of 12
reg.multi	List of 12

R 4.1.2 · C:/dab/

```
F-statistic: 22.64 on 3 and 41 DF,  p-value: 8.349e-09

> library(car)
>
```

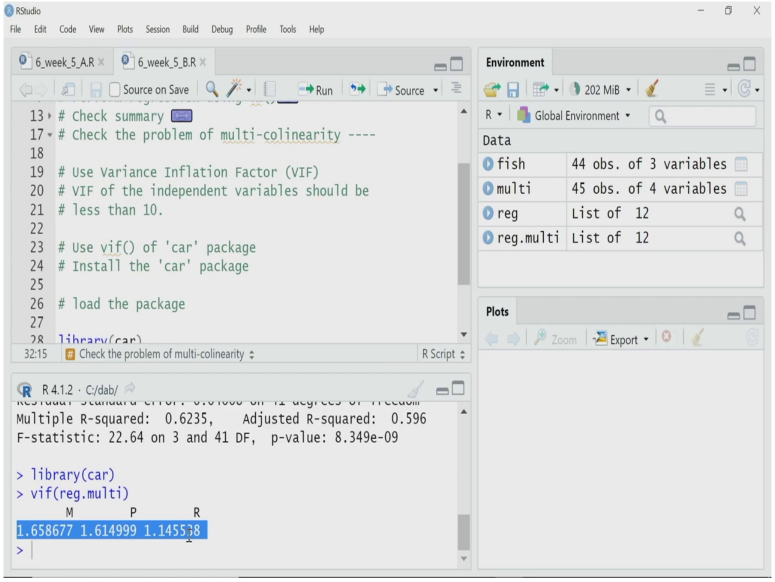


```r
# Check summary
# Check the problem of multi-colinearity ----

# Use Variance Inflation Factor (VIF)
# VIF of the independent variables should be
# less than 10.

# Use vif() of 'car' package
# Install the 'car' package

# load the package

library(car)

# Perform VIF analysis

vif(reg.multi)
```

```
32:5   Check the problem of multi-colinearity   R Script
```

```
R 4.1.2 · C:/dab/
F-statistic: 22.64 on 3 and 41 DF,  p-value: 8.349e-09

> library(car)
>
```

```
Data
fish        44 obs. of 3 variables
multi       45 obs. of 4 variables
reg         List of  12
reg.multi   List of  12
```



```r
# Check summary
# Check the problem of multi-colinearity ----

# Use Variance Inflation Factor (VIF)
# VIF of the independent variables should be
# less than 10.

# Use vif() of 'car' package
# Install the 'car' package

# load the package

library(car)
```

```
32:15   Check the problem of multi-colinearity   R Script
```

```
R 4.1.2 · C:/dab/
Multiple R-squared:  0.6235,    Adjusted R-squared:  0.596
F-statistic: 22.64 on 3 and 41 DF,  p-value: 8.349e-09

> library(car)
> vif(reg.multi)
       M        P        R
1.658677 1.614999 1.145538
>
```

```
Data
fish        44 obs. of 3 variables
multi       45 obs. of 4 variables
reg         List of  12
reg.multi   List of  12
```

library(car)

vif(reg.multi)

Now, this car has a function called VIF shorthand for Variance Inflation Factor, and I will call that function and I will use this data of my regression reg dot multi as an argument. So, it has calculated the variance inflation factor for each of these independent variable M, P and R. If you remember, if the variance inflation factor of any of this variable is greater than 10, then we have trouble or multicollinearity in our data set. But in this case, all of them are close to 1 or 2. So that means my data set does not have multicollinearity problem.

So, I am happy I have not to get rid of any of these independent variable and my multiple linear regression model that I have created just now is good enough. So, what we have learned in this lecture? We have learned that I can use the lm function to perform multiple linear regression, just like we have done simple linear regression only thing when I am specifying the model, I have to specify who are the independent variable.

And I have also shown you a shortcut, when to write that model in the lm function, when you have large number of independent variable where you can simply skip all the writing all those variables explicitly and just put a dot to make lm understand that you want to consider all the independent variable and intercept in your model. And also discussed about the statistics that you get from the summary of this regression. And we have also discussed about how we can calculate the variance inflation factor. That is all for this video. Thank you for learning with me today.