

Lecture - 04

Calculation with R Software - Built-in Commands and missing data handling

Welcome to the, next lecture on descriptive statistics with R software. You may recall that in the earlier lectures, we had discuss, that how R is going to be used for different types of

computation. when we are trying to use the scalars as well as data vectors, in this lecture we will continue, with the same topic and I would try to show you, that in R, there are two types of computation, one using simple operation, where you have to define and beside those there are some built-in functions and those built-in functions can be used, directly over the data or the data vector, to obtain the required outcome. In this lecture I will also show you that, how you are going to handle the R software, when some values are one value in our data vector is missing, which is not available due to any reason. So, let us start our, lecture here. Now, whatever you do, I will try to take some examples and through that example, I will try to show you, an illustrator, that how to use the built-in function, there are many, many functions available, so it is not practically possible for me to cover all the built-in function. But, I will take sufficient number of examples, so that you are comfortable in using all other function.

Refer slide time :(1:54)

Maximum

```
> max(4.2, 3.7, -2.3, 4)
[1] 4.2
```

R Console

```
> max(4.2, 3.7, -2.3, 4)
[1] 4.2
>
```

↓

```
> max(c(4.2, 3.7, -2.3, 4))
[1] 4.2
```

R Console

```
> max(c(4.2, 3.7, -2.3, 4))
[1] 4.2
>
```

So, let me take here, the first example, that suppose I want to find out the, maximum among some values, four example, I am take taking here a Decatur vector, which contains four values, four point two, three point seven, minus two point three and four. And I really want to know, out of these four values, whichever is the maximum out of them. So, you can see here means, obviously Because, there are only four values, so you can see here, that this 4.2 is the maximum value and once you enter it here ,it will give you, this value for 0.2. and if you try to, do it over the R console, here is the screenshot, well I will also be showing you later on, that how to do this thing and one thing what you have to observe here, that earlier I had told you, that whenever you are trying to give a data, you have to give it, in the form of a data vector using the C command. But, there are some built-in functions, which can be used, without using the C command. So, I am giving you here, an example of this, and max function. Here, you can see here, in this case and in the second case, I am finding the maximum among the same set of values, but here I have used, here the C command. so here, I am trying to combine that data using the C command, for these four values, 4 point two, three point seven, minus two point three and four and here also I get the same outcome and here is

the screenshot of this thing, but here, I would like to give you one advice, well it is more difficult to keep a track that, which built-in function is going to use with the C command and what are the built-in function? Which are used without the C command? And so I will say, the simple rule of thumb is this always uses the C command. So, whenever I have to give a data vector, without creating any confusion, without creating any problem, I will simply try to give the data vector using the C command. Right? Okay? Similarly, in case if you want to find out the minimum, so minimum again, I am trying to take the same value.

Refer slide time: (4:15)

Minimum

```

> min(4.2, 3.7, -2.3, 4)
[1] -2.3

```

not using c

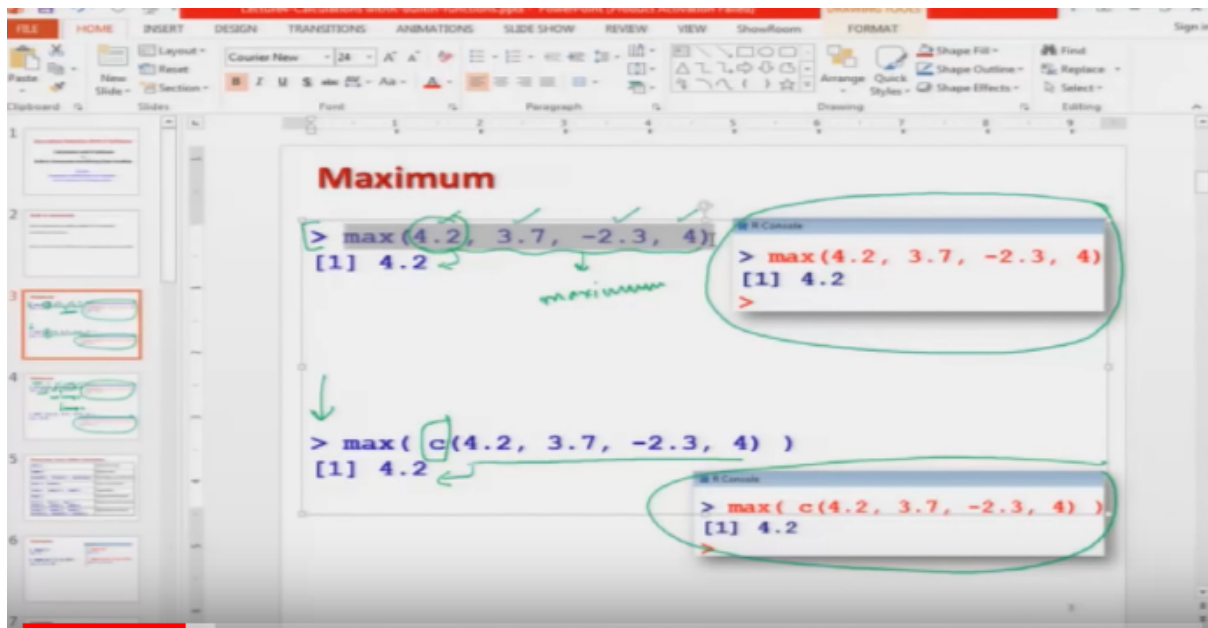
```

> min( c(4.2, 3.7, -2.3, 4) )
[1] -2.3

```

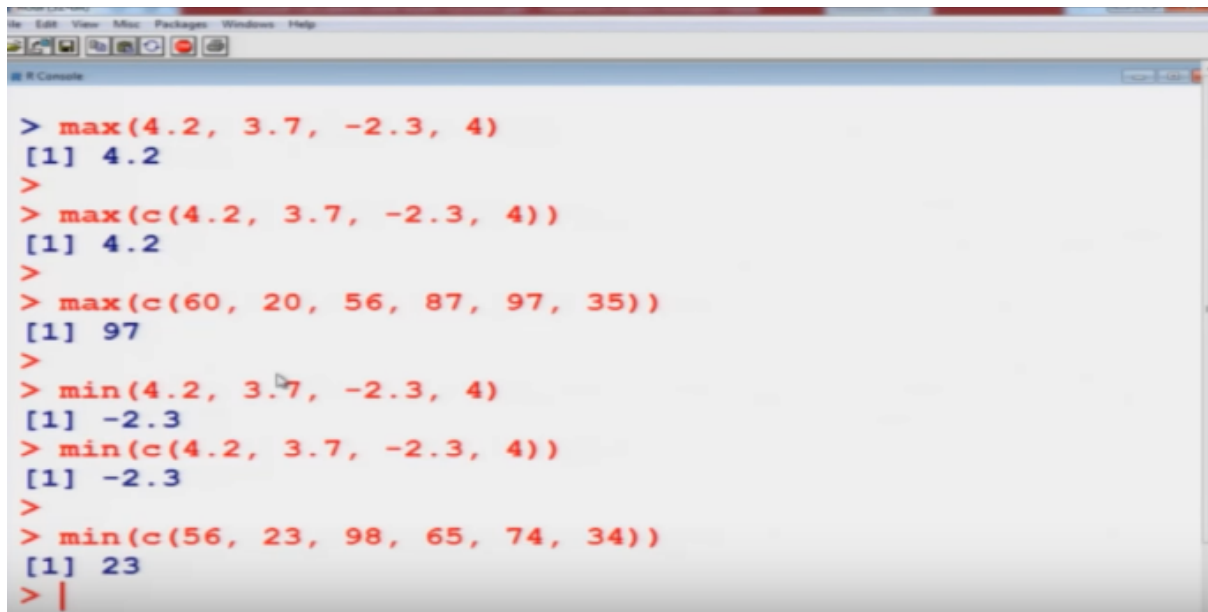
But here, in this case, the command here is M I N and after the command in the case of maximum, it was M A X. Right? so now, in case if I try to write down here mi n, inside the bracket, if I try to give her all the data values and if, if I try to press your enter, on the R console, then I get here, the value minus two point three and you can see yourself here, that out of four point two, three point seven, minus two point three and 4, this minus two point three is the minimum value. Okay? So, minimum and maximum are the functions, where you need not to write the entire program yourself. But, somebody has already done the programming, to find out the minimum and maximum value and that program has been renamed, as say M I N and says ma M A X and you simply need to use it. But, definitely as I told you that r is not a black box, so in case, if you really want to know that, what logic has been used? What programming has been used? It is possible to look into the steps, which are used in finding these values. Okay? Here, is the same operation and Again, I'm trying to show you here, that here I am, not using this C command and Here, I am using C command. I mean, say the data is combined, using the C command, where as in the first case the data is not combined using the C command but, in both the cases the outcome is the same. So, again I will advise, you that always use the C command, to combine the data vector and here is the output or the screenshot. Now, I will try to show you, that how these things are happening on the R console. Right?

Refer slide time: (6:09)



So, let me, go to here, this thing first I try to copy this command and I come to here,

Refer slide time: (6:19)



R console and I try to paste this here. So, you can see here, this will give you, four point two and yeah. In case, if I want to write down, here the C command, you can see here yeah. so now, the same the data, data vector has been combined using the C command and if you see the answer is going to be the same and similarly in case if you try to take here, another example here, suppose here, between 60, 20, say 56, 87, 97, 35 and so on. So, you can see here, the maximum value, here is 97. Right? Okay? now, similarly in case if you try to find out the minimum, then I can do here, I can use here the command, M I N, over the same data

set, 4 point two, three point seven, minus two point three and four and you can see here, now the minimum value is coming out to be, minus 2 point 3 and similarly in case if I try to use here this C commend on here, even then ,you will get the same outcome . Right? And similarly if you try to take it, another value here and if I try to say here, minimum of the data vector between, 56, 23, 98, 65, 74, 34 and so on. So, it comes out to be 23. So, you see here, I am trying to take a very small data set, where you can verify yourself that whether it is giving you a minimum value or a maximum value, but this built-in function are very useful, when you are trying to deal with a huge Gator said where there may be five thousand values, ten thousand values or twenty thousand values or so on .hey, there you cannot find out these values manually, so these functions help you there. Now, if you try to understand I have taken here, two examples, M I N the minimum and M A X, maximum and I have illustrated how you can operate it over a data vector. now ,similar to minimum and maximum ,there are some other built-in functions, which are available in R , there is a long list, but here, in this slide I am trying to give you an overview of those built-in function and how to use those built-in function? The process and the procedure is exactly the same what you have learnt in the case of minimum and maximum.

Refer slide time: (8:55)

Overview Over Other Functions	
<code>abs()</code> <i>abs(c(data vector))</i>	<u>Absolute value</u>
<code>sqrt()</code>	<u>Square root</u>
<code>round()</code> , <code>floor()</code> , <code>ceiling()</code>	Rounding, up and down
<code>sum()</code> , <code>prod()</code>	Sum and product
<code>log()</code> , <code>log10()</code> , <code>log2()</code>	Logarithms
<code>exp()</code>	Exponential function
<code>sin()</code> , <code>cos()</code> , <code>tan()</code> , <code>asin()</code> , <code>acos()</code> , <code>atan()</code> }	Trigonometric functions
<code>sinh()</code> , <code>cosh()</code> , <code>tanh()</code> , <code>asinh()</code>, <code>acosh()</code>, <code>atanh()</code> }	Hyperbolic functions

For example, in case if you are interested in finding out an absolute value. Right? Then you simply have to write here, ABS inside the brackets, you simply have to give the here the data vector, even a single value or a vector value, both are acceptable. similarly ,in case if you want to find out ,the square root ,of any value or any data vector, the function is SQRT and similarly, in case if you want to find out the rounding of value, the function is round, if you want to floor, the function is floor, if you want to ceiling , then the function is Ceiling, if you want to find out the sum of two numbers or say sum of two vectors, then the function here is SUM and for put it the function is PROD and similarly if you want to make different types of logs, these functions are there for exponential, functions are there for trigonometric

functions ,sine, cost and etcetera, are there and hyperbolic function says sine X cos a standard etcetera, are also there and, and there is a long list. But now, I will try to take here, some example and I will try to show you, How you are going to use them.

Refer slide time: (10:16)

Examples

```
> sqrt(4)
[1] 2
```

$\sqrt{4}$
 $\sqrt{4}$

```
> sqrt(c(4,9,16,25))
[1] 2 3 4 5
```

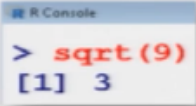
```
> sqrt(4)
[1] 2
> sqrt(c(4,9,16,25))
[1] 2 3 4 5
```

$\sqrt{4}$ $\sqrt{9}$ $\sqrt{16}$ $\sqrt{25}$
 ↓ ↓ ↓ ↓
 2 3 4 5

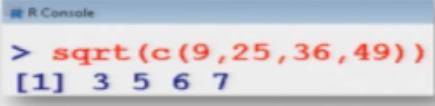
For example, in case if you want to find out, the square root of any value, suppose I want to find out, the square root of 4. Right? So, in this case I simply have to write it SQRT and inside here, single value 4. And if you try to see it here, this valuable, value will come out to be 2. Now, I will try to illustrate you, that if you want to find out the square root of a data vector, then exactly on the same way, as addition, subtraction, multiplication, division, they are operated, over each and every element in the data vector, similarly this square root operator will also be executed over all the elements of the data vector. if for example, in case if I try to take here at cricket a vector C, say four, nine, sixteen and twenty five and suppose if I try to find out, the square root of this guitar vector, then this will be operated like this, say here,C a square root of four, square root of nine, square root of sixteen, square root of twenty-five, and this answer will come out to be here, two, three, four and here five. So, you can see here, in this case, I'm trying to do the same thing and the outcome is coming out to be here, two, three, four and five. And here is the screenshot of the same operation. Right? Okay?
 Refer slide time: (11:48)

Examples

```
> sqrt(9)
[1] 3
```




```
> sqrt(c(9,25,36,49))
[1] 3 5 6 7
```



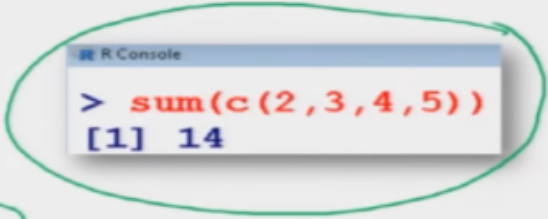
Now, and similarly means, if you try to click another example, that you can try yourself try to find out a square root of nine and then you square root of this another data vector ,contained containing the values nine, 25, 36, 49. So, I'm just leaving it for your practice. Okay?

Refer slide time: (12:05)

Examples

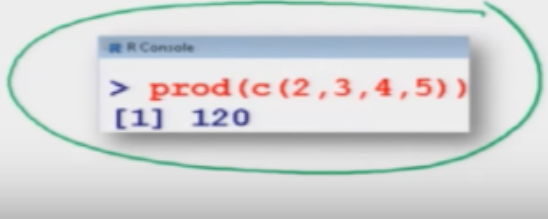
```
> sum(c(2,3,4,5))
[1] 14
```

Sum
 $2 + 3 + 4 + 5$
 $\text{sum}(c(2,3,4,5))$




```
> prod(c(2,3,4,5))
[1] 120
```

$2 \times 3 \times 4 \times 5$
 $\text{prod}(c(2,3,4,5))$



Now, another important aspect, sum and product, there is a built-in function here s um and this functions, find out the sum of all the values inside the data vector, for example, in case if I want to find out the values of 2 plus, 3 plus, 4 plus, 5. then I can give these values in the form of a data vector, consisting of 4 values, 2, 3, 4, 5 and then I simply have to operate it here and write s um of this data vector and this will, give us the value of 2 plus, 3 plus, 4 plus, 5. that is the summation of all the values inside the data vector and this value will come out to be here ,14 and this is the screenshot here, similarly in case if I say, I want to find out the product of say 2, 3, 4, 5, that is 2, into 3, into 4, into 5, that means, I can combine this data, data in the form of a data vector, C, two, three, four, five and then if I use here the built-in

function PROD, then this will give us the multiplication of all the values inside the data vector. Right? So, this PROD function, this is used to find out the product of all the values inside the data vector. So, for example here, I try to use, this command over the R console and this value comes out to be 120. 2, 3's are 6, 4's are 24, 5's are 120. Right? And this is the screenshot of the same operation. Now, I try to come to the R console, so that I can show you, these operations. Right? So I will try to show you here, the square root sum and product operation.

Refer slide time: (14:04)

```
> sqrt(9)
[1] 3
>
> sqrt(c(4, 9, 16, 25))
[1] 2 3 4 5
>
> sqrt(c(-4, 9, 16, 25))
[1] NaN 3 4 5
Warning message:
In sqrt(c(-4, 9, 16, 25)) : NaNs produced
> |
```

For example, if I try to find out here, square root of here, see here nine, you can see here this comes out to be here three and if I try to find out the square root of our theta vector, C here, 4, 9, 16, 25. So, this will give us the value of square root of 4, a square root of 9, is square root of 16, square root of 25. Which is 2, 3, 4 and 5? Ok? Now, look at me they take here some negative value, let us see, what happens minus 4, say here nine, sixteen and here 25. So, this will give here, what is here NaN? this is something, new for us, so I will try to show you, the use of this NaN and etc, all these things after a little couple of slides in the same lecture. so, the quickest is, trying to show you, in very simple word that there is some issue, some problem and it's showing that it is not possible to compute it. Right? So, that is what you have to be careful that? For this minus 4, it is giving us the value and a something like, not available type of thing and for all other value it is giving you the correct values. Okay?

Refer slide time: (15:31)


```

> sum(c(5,6,7,8,9))
[1] 35
> sum(c(-3,-4,-5,-6))
[1] -18
>
> prod(c(2,3,4,5))
[1] 120
>
> prod(c(-2,3,4,5))
[1] -120
>
> abs(-9)
[1] 9
> abs(c(-9, -6, 7, 8))
[1] 9 6 7 8
> |

```

So now, let me give you an idea of the, sum function. So, if I try to find out the here, sum of C here, 5 values, 5, 6, 7, 8, 9, combine in the data vector. because ,this comes out to be here 35 and this sum operator is also valid for negative values, also for example, some of - 3, - 4, - 5 and say - 6, this will again come out to be minus 18, so there is no issue and in case if I try to take here, two data vectors also, that also can be done, that we already have seen in the earlier lecture and similarly if I try to show you here, now the product function, C if I want to make it here, product affair C of here 2, 3 ,4, 5 . Right? So, the quickest comes out to be hundred twenty 2, 3's are 6 and 6, 4's are 24, 24, 5's are, 120. So, it is trying to give us the product of all the values present inside the data vectors. Now, if I try to take here one value to be here negative, then let me see whether this operates or not. So, you can see here the unset is coming out to be minus 120 because, this sum and product function they are the valid mathematical functions and they are operative or positive, as well as, negative values. Right? and similarly if you want to C the ,use of your absolute function. So, I can show you here, absolute off here, say this minus nine is actually nine, so you can see here, which is happening and similarly if I try to take care, C a data vector consisting of C here, - 9, - 6 and 7 and here 8. So, you can see here, that there are two negative values and two positive values in this theta vector. so now, once you try to operate the absolute function, it will give you the minus 9, will become 9 ,minus 6 will become say 6 & 7 & 8 they are ,they already are the positive values, so their absolute function remains the same ,so you can see here, that these operations are not difficult to do in the R software.

Refer slide time: (17:43)

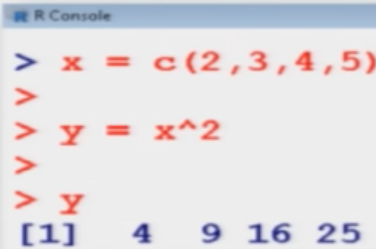
Assignments

An assignment can also be used to save values in variables:

```
> x = c(2,3,4,5)
```

```
> y = x^2
```

```
> y  
[1] 4 9 16 25  
    22 32 42 52
```



```
R Console  
> x = c(2,3,4,5)  
>  
> y = x^2  
>  
> y  
[1] 4 9 16 25
```

And they will help us with hitting in solving many complicated functions, very easily and beside this thing I would also try to show you, that because for a quick revision, that whenever we are trying to assign a value to a vector, then it is also possible to assign our new variable to our new vector, for example, in case if I try to say, take here, data Walter of here's, two, three, four, five and suppose I want to obtain the square of all the values say, 2 squared, 3 squared, 4 squares, 5 B Square, then I can simply denoted here, by here X hat 2 and this value can be stored into a new variable here Y. Right? and once you try to see the value of here Y, this will come out to be 4, 9, 16, 25, which is, 2 squared, 3 squared, 4 squared, + 5 squared and this is here the screen shot, so the cursor pretty simple operation, so that will help us in assigning, the outcome of an operation into a new variable al. Right? if for example, in statistics cooker will see at many, many places we need to find out the sum of squares.

Refer slide time: (19:04)

Examples

To find sum of squares:

```
> x = c(2,3,4,5)
```

$$z = \sum_{i=1}^n x_i^2$$

$$x = c(2, 3, 4, 5)$$

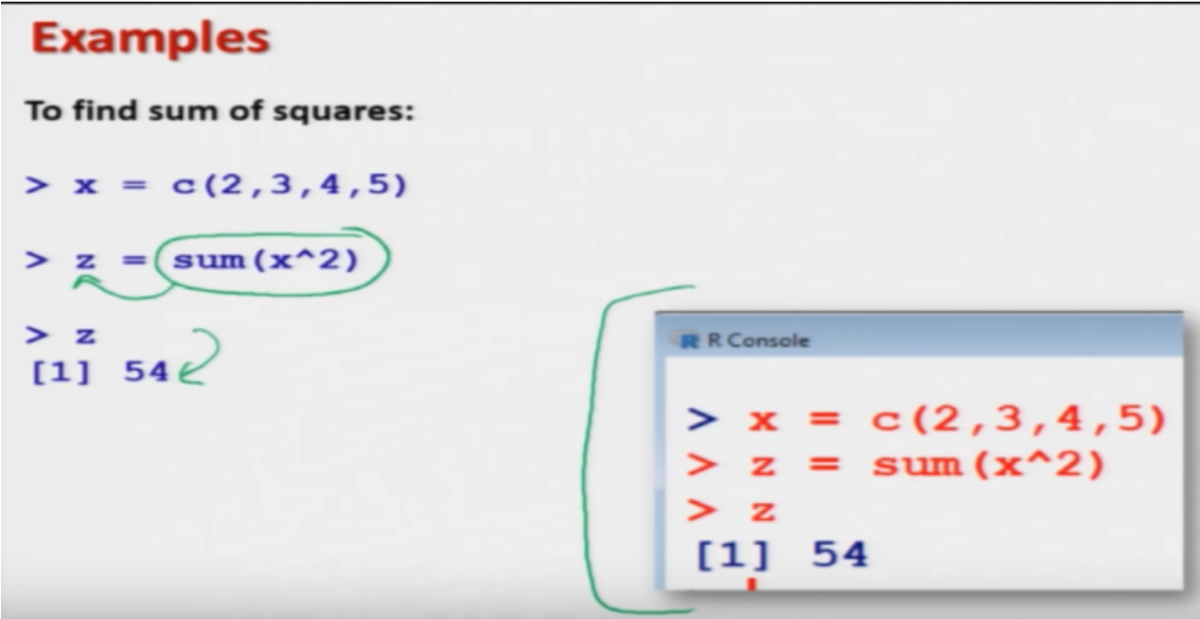
$$x^2 \text{ or } x * x = c(2^2, 3^2, 4^2, 5^2)$$

$$\text{sum}(x^2) \text{ or } \text{sum}(x * x) = 2^2 + 3^2 + 4^2 + 5^2$$

$$x = c(x_1, x_2, \dots, x_n)$$
$$\sum_{i=1}^n x_i^2$$

So basically, if I want to find out the value of suppose I have, some data vector here consisting of some values X_1, X_2 , say here, X_N and suppose I want to find out the sum of squares of these value, that means, you first try to square the value and then find out the sum. so now ,using this built-in function ,you can see here, what I can do? So, what I have to do? Here, that first I need to define here all the values inside a data vector, so I am trying to define here, a data vector here $C, 2, 3, 4, 5$. Right? And now, this is asking me to find out the square, so what I can do here? That, I need to find out the square, so I can write down here X^2 or I can write down even here X multiplication X and this is going to give me the value of here, say two square, three square, 4 ay Square + v square and now, I need to find out the value of sum of two square, three square, 4 a square,5 B Square. So now, I can use here, the built-in function here, $\text{sum}(X^2)$ or see here, some $X * X$ and this is going to give me the value of 2 square plus, 3 squared plus, 4 squared plus, 5 square. So, you can see here, that in case, if I want to find out the value of this function here Z , which is the sum of squares of different values

Refer slide time: (20:42)



Examples

To find sum of squares:

```
> x = c(2,3,4,5)
> z = sum(x^2)
> z
[1] 54
```

R Console

```
> x = c(2,3,4,5)
> z = sum(x^2)
> z
[1] 54
```

Then I can write down, it here simply like here, $\text{sum}(X^2)$ and this value can be stored in a variable, say here Z and some of 2 square plus, 3 square plus, 4 square plus, 5 b square, comes out to be here, 54 and this is the outcome here, so let me try to show you, it over the R console also so that you get more confidence.

Refer slide time: (21:10)

```

> x=c(2,3,4,5)
> x^2
[1] 4 9 16 25
> z=sum(x^2)
> z
[1] 54
>
> prod(x^2)
[1] 14400
> |

```

So, you try to see here, I try to define here, a vector here, 2, 3, 4 & 5 and I try to find out here, X square . Right? And suppose if I try to find out here, Z is equal to hear, some of say X square, so this comes out to be and if you try to C. What is the value of Z? This comes out to be 54 . Right? And similarly in case , if you want to find out the product of 2 is square , 3 square, 4 square ,v square, I can write down here, Again, here is some instead of some ,I can write down here ,product of X square and this will come out to be 14400.so, what is this value this is the product offered 2 squared, into 3 square, into 4 a square ,into 5 u square or simply the product of, 2, three square, four square, v square . Right? So, these types of operations are possible and with this example I will try to illustrate here,

Refer slide time: (22:19)

Examples

To find sum of squares of deviation from mean

$x = c(2, 3, 4, 5)$

$$z = \sum_{i=1}^n (x_i - \bar{x})^2 = \sum_{i=1}^n x_i^2 - n\bar{x}^2$$

$x = c(x_1, x_2, \dots, x_n)$
 $\bar{x} = A.M = \frac{1}{n} \sum_{i=1}^n x_i$
 ↓
 mean(x)

$\sum_{i=1}^n (x_i - \bar{x})^2 = \sum_{i=1}^n (x_i^2 + \bar{x}^2 - 2x_i\bar{x})$
 $= \sum_{i=1}^n x_i^2 + n\bar{x}^2 - 2\bar{x} \sum_{i=1}^n x_i$
 $= \sum_{i=1}^n x_i^2 + n\bar{x}^2 - 2\bar{x}(n\bar{x})$
 $= \sum_{i=1}^n x_i^2 - n\bar{x}^2$

length(x)
 ↓
 # of data points
 in x

~~$\sum_{i=1}^n x_i^2 - \text{length}(x) * \text{mean}(x)^2$~~

The computation of a very important or very popular function any statistic. Suppose, I want to find out the value of this function here, summation I goes from 1 to n xi minus X bar whole square, what is here X bar? X bar here is the automatic mean, it is simply here 1 over n ,summation I goes from 1 to n, say here X I for a data vector here, see X 1, X 2 see here X n. So, what I am trying to show you here, that if I have got the values, I can find out it's automatic mean and for that I have some built-in function, what is called as here mean MEAN? so now, what I want here ,first that I want that each and every value X I, should be subtracted from the automatic mean and then, this value has to be squared and then all these values are a square and then I want to find out the sum of all those values .now, this value can be further simplified, as I goes from 1 to n, X I square plus, X bar square minus, twice of X I into X bar and if you try to open it this becomes a summation, I goes from 1 to n, X I square plus ,n times X bar square, minus twice of X bar, summation I goes from 1 to n X I and this becomes a summation I goes from 1 to n X I squared plus NX bar square minus twice of X bar and the summation X I is n times X bar. So, this quantity comes out to be, the same as, I goes from 1 to n, summation X I square, minus n times X bar square ,which I have written here, so now, in order to compute, this value, what I can do? that I already have computed ,this value, this I already had written I say here sum of say X hat 2 and what about this value in X value ? What is your n? n is the number of elements in the data vector, so for that there is a built-in command in R ,which is called here as a length ,so if I try to say here length of here X ,so this is going to give me the length of the data vector or the total number of data points, in the Decatur vector X ,so instead of here and I can use here the command here length of here X and x for X bar I already have a command built in command here, so you here mean of here X so I can write down here ,mean of here X and say here square well don't worry, all these commands like as mean L length, etcetera we are going to discuss in the for the slides ,but here I wanted to give you an example, to show you that how this built-in functions are going to be useful in computing a very important quantity in statistics.

Refer slide time: (25:56)

Examples

To find sum of squares of deviation from mean

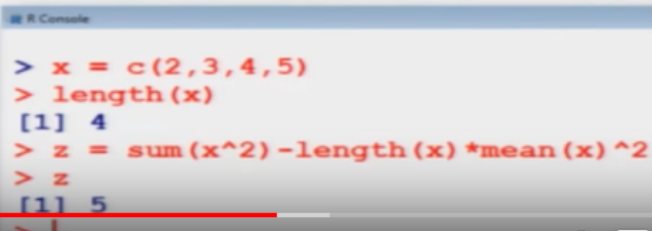
```
> x = c(2,3,4,5)
```

$$z = \sum_{i=1}^n (x_i - \bar{x})^2 = \sum_{i=1}^n x_i^2 - n\bar{x}^2$$

```
> z = sum(x^2) - length(x) * mean(x)^2
```

```
> length(x)
[1] 4
```

```
> z
[1] 5
```



So now, in case if you try to do it on the R console here. So, suppose I try to take here the cricketer said, said 2, 3, 4, 5 and this is the same value, so I have written here sum of X square, which is corresponding to this thing, length of here X which is corresponding to here N and this here mean of X square here which is corresponding to X bar square and once you try to store, the value of this function, into a new variable here Z. So, once you try to execute it, first I try to show you here, what is the P length of here X? Which, which is coming out to be here, 4 and what is the value of here Z, this comes out to be at 5 and this is the screen short of the same operation. Right? Okay?

Refer slide time: (26:44)

Examples

To find sum of cross product:

```
> x1 = c(2, 3, 4, 5)
> x2 = c(6, 7, 8, 9)
```

$$2 \times 6 + 3 \times 7 + 4 \times 8 + 5 \times 9$$

```
> z = sum(x1 * x2)
```

```
> z
[1] 110
```

And similarly if I try to take it another example on the same lines, suppose I try to take care 2, the gate our vectors say here, X 1 and X 2, consisting of 4 values 2, 3, 4, 5 & 6, 7, 8, 9 and suppose I want to find out the sum of the product. So, this is something like this, 2 into 6 plus, 3 into 7 plus, 4 into 8 plus, 5 into 9. So, I need to find out the value of this thing. so now, I can do it very easily using the this built-in operator, first I need to find out the multiplication, so for that I can simply use here the operator X 1 star X 2, that means, I'm trying to multiply the components of two vectors X 1 and X 2 and whatever is the outcome, that I am going to sum. So, this value will come out to be 110. Right? So, before I go further I will try to show you, on the R console that how these things are happening

Refer slide time: (27:44)

```

> x
[1] 2 3 4 5
> length(x)
[1] 4
>
> mean(x)
[1] 3.5
>
> sum(x^2) - length(x) * mean(x)^2
[1] 5
> y=c(6,7,8,9)
> sum(x*y)
[1] 110
> |

```

So, suppose if I try to, to take it here the same data vector ,that we have taken it here is here earlier ,2, 3, 4 ,5 .so, first if you try to see here ,the pin length of here X ,there are 4 elements .so, it should come out to be here 4 and similarly if you want to find out the help of, find out the here, mean of here x mean of x will come out to be a 3.5 ,that is 2 plus, 3 plus, 4 plus, 5 divided by 4. Which is equal to 3 point 5? Now, in case if you want to find out the same command here, some of her X square ,minus length of X into mean of X a square. So, you can see here, this function is coming out to be here, like this and the value is coming out to be here 5, and similarly if I try to take here another data vector here, say 6, 7, 8 and here 9 and if I try to find out the sum of the, product of X and Y, data vector so, what this is coming out to be hundred ten, what is this value? This is trying to first multiply the corresponding elements of two data vector x and y and whatever is the product, it is trying to find out the sum. So, with this illustration you can see here very clearly, that this built-in function will help us in many type of operation that we are going to learn in future lectures. Right? Now, after this I would like to address, another small topic, which is very, very useful, suppose someone is asked to collect, suppose five thousand data values and after the values are collected, they are manually entered, on a computer and suppose there are various possibilities, that the number of values present in the data vector, are not really all and some data is missing. So, in case if a value is missing in any data vector, then how to handle it? this type of situation may occur ,for example ,someone is asked to collect the data from say five houses and suppose he goes to the third house and the house is locked, so he will try to indicate that this data is not available, so he will use some symbol ,so in R, in case if the data is missing, there are some standard symbols, which are used and there are some functions and commands, which helps us in modifying the statistical tool and the mathematical tools, to handle the missing data. So, this is what we are going to learn now? in the next couple, of minutes.

Refer slide time: (30:48)

Missing data

TRUE and **FALSE** are logical operators that are used to compare expressions.

TRUE and **FALSE** are reserved words.

T can also be used in place of **TRUE**. $T \rightarrow \text{TRUE}$
F can also be used in place of **FALSE**. $F \rightarrow \text{FALSE}$

TRUE and **FALSE** are not the same as ~~true~~ and ~~false~~ respectively.

So, before I go to discuss the handling of missing data, first let me inform you few small things, there are two letters, true and false, which are written in capital letters, capital t, capital r, capital u, capital e and all capital letters in false, these two are the logical operators and these operators are used to compare, different expressions, we know, that in mathematics, there are two types of operator which are the mathematical operator and, and say another are logical operator logical operator, for example, say less than, for example, if I say pipe is say more than three, so this is true this is false, this is true. But, if I say five, is smaller than three, then it is false. So, here I just want to know the answer in terms of true and false, I am NOT interested in, how much it is larger? And how much it is, is smaller? So, these are our logical operators, so this capital letters true and capital returns false, they are the logical operators and they are the reserved word, means, as soon as you Right here. Right? Capital T, capital u, capital R, capital u, capital E, the R will automatically assume it, that you are going to use a logical operator. So, you cannot use it to define a new variable or, or any variable from your site, that will be acceptable and also in case of here the entire word, true or false, you can also use here, the first letter, capital T or say capital F, to denote the words true and false respectively. So, T can be replaced for say here true and say F can be replaced for here false, . Right? and remember one thing these true and false, have to be written only in that capital letters, in case if you are writing there many small letters or even if a single letter in these two words is a small, this will not remain as a logical operator and R will not consider it as a logical operator. So, this small true and small false, this is not possible, and they are not the same as the true and false in capital letters

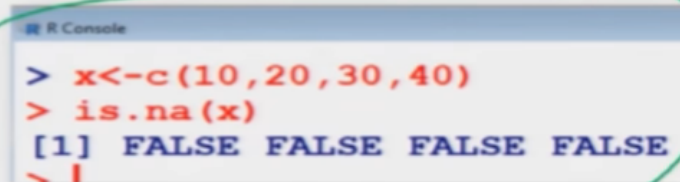
Refer slide time: (33:04)

Missing data

How to know if any value is missing in a data vector?

```
> x <- c(10,20,30,40)
> is.na(x)
[1] FALSE FALSE FALSE FALSE
```

is.na (data vector)



```
R Console
> x<-c(10,20,30,40)
> is.na(x)
[1] FALSE FALSE FALSE FALSE
> |
```

As soon as, we get any data, then I try to input the data, first option is this, I can input the data in the form of a data vector, using the command C. Now, I would like to know, is there any value which is missing in the data. So, first question is how to know whether any data value is missing inside the data vector? So, in order to note this thing there is a built in command here, what we call here? Say, is dot and a and inside the bracket, you have to give the data vector, in which you want to find, is any value missing. So, for example, I try to take here later the vector here, consisting of 4 values 10, 20, 30, 40, here you can see ,all the values are present and no values is missing, so I try to execute the command ,is dot NA and inside the bracket say X and you will get here an outcome like this, false, false, false, false, that means, this 10 value is not missing, so saying that 10 is missing or is 10 missing, this is false, similarly this false corresponding to this 20. So, so I'm asking with this command is 20 missing, answer is false, no it is not mean, what you think it is present? Now, both the 30 and 40 these values are available ,so this command is dot n it is giving me a false statement, so this is the screenshot, of the same operation, I will try to show you over the r console also,

Refer slide time: (34:53)

Missing data

How to know if any value is missing in a data vector?

```
> x <- c(10, NA, 30, 40)
> is.na(x)
[1] FALSE TRUE FALSE FALSE
```

NA

```
R Console
> x<-c(10,NA,30,40)
> is.na(x)
[1] FALSE TRUE FALSE FALSE
```

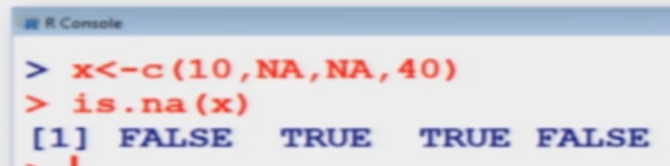
And now let me take here another example, Here I am trying to replace the value 20 from the earlier data vector by here n a. Right? you can see here I am writing here, capital N and capital A, this is also a reserved word, I will try to discuss it after a couple of slides but, this is also a result word and this is used in R, to indicate that the value is not available. Right? so now, in this case, whosoever is entering the data he has to be told, that in case if the data is missing, he or she has to write an, A in place of the missing data, so now, the data vector comes out to be consisting of here, 4 values, 10, NA, 30 and 40. So, now when I try to operate the command here is dot na, this is giving me the outcome here false, true, false, false, this will share what? this false corresponding to distance, so I'm trying to ask here is 10 missing, answer says, no it is available, hence my, mic command is false. So, it is giving me the value here false. Now, I come to the second value here, na and I ask is dot na and this value, is say yes. this value is missing and hence the answer is true, my statement is true and similarly for 30 and 40, these two values are available, so this is trying to say, that these values are not missing, they are available,

Refer slide time: (36:28)

Missing data

How to know if any value is missing in a data vector?

```
> x <- c(10, NA, NA, 40)
> is.na(x)
[1] FALSE TRUE TRUE FALSE
```



```
R Console
> x<-c(10,NA,NA,40)
> is.na(x)
[1] FALSE TRUE TRUE FALSE
```

And after this, Suppose, if there are more than one values, even then there is no problem at all, for example, in the same data vector if I try to miss two values 20 and 30 here, in this new X, so once I try to operate the command here, is dot NA, then it is giving me here, false and false for the values, for those values which are available and it is giving me the answer, true and true, for the values which are not available, so by this operation I can always find out whether the values are missing or not, so as long as, I am getting here, all false, that means all the values are available and if I am getting even a single true, that means value is missing in the data vector and now, I will try to show you that in case if the value is not available it is missing in the data vector, then what happens in case if I try to operate any built-in function. Right? Ok. But, before that let me come back to here.

Refer slide time: (37:33)

```
> x=c(20,30,40,50)
> x
[1] 20 30 40 50
> is.na(x)
[1] FALSE FALSE FALSE FALSE
>
> x=c(20,NA,40,50)
> x
[1] 20 NA 40 50
> is.na(x)
[1] FALSE TRUE FALSE FALSE
> x=c(20,NA,40,NA)
> is.na(x)
[1] FALSE TRUE FALSE TRUE
> |
```

R Console, so that I can show you, that is it really happening or not. So, if I try to take care for example here, the X greater vector here, C here, 20, 30, 40 and here fifty. Right? So, the Chris is my here X, so I try to do hit is dot na and inside here X and it is giving me false, false, false, all that means, all the data values are available. Now, in the same data set I try to replace, the second value by NA and now, you can see here, my this X becomes here, 20 NA, forty ,fifty and if I try to repeat here, the same command is dot NA, can it is giving me here for the missing value, it is giving me here true . Right? And similarly if I try to make it here, more than two values, to be missing. So, and if I try to operate with the same operator, I get here ,false for those values ,which are available and the true for those values, which are Epson, which are missing .so, for the twenty and a 40 and a I am getting the outcome false ,true, false, true . Okay?

Refer slide time: (38:40)

Example : How to work with missing data

```
> x <- c(10, 20, NA, 40) # data vector
```

$\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$

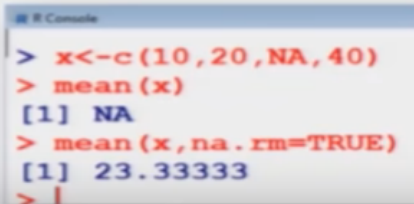
```
> mean(x)
[1] NA
```

$\frac{10+20+NA+40}{4}$

```
> mean(x, na.rm = TRUE) # NAs can be removed
[1] 23.33333
```

$\frac{10+20+40}{3} = 23.33$

$mean(x, na.rm = T)$



```
> x<-c(10,20,NA,40)
> mean(x)
[1] NA
> mean(x,na.rm=TRUE)
[1] 23.33333
> |
```

So now, let me come back to our slide and let me try to show you here, that how the things are going to happen, when some value is missing. So, suppose I try to consider the data set here, in which one value is missing, twenty, thirty, Na and forty. Now, suppose I want to find out the mean, mean of X ,this sample mean is defined as here sum of all the values x_1 plus, x_2 plus, x_n divided by the total number of observations here X . Right? So now, in case if you try to find out the mean of 10 plus, 20 and na and 40, this will become here, 10 plus, 20 plus, na, plus 40, more of elements in the data vector which is here 4 and you will see here, it is not really mathematically possible to add value, na in any numerical values. So, this answer will come out to be here Na. Right? whereas,in case if you try to use this command ,see here n a dot, R M is equal to true, allowed this command, I am going to explain you in later on, whenever we are going to deal with the statically function but, here I want to give you an idea that that how you are going to modify the same command, when there is a missing value in the data vector al. Right? So, in this case suppose I know ,that there is one value, which is missing in the data set ,I have to modify my command ,mean of XS ,mean of X .so, I can

write down here, mean of hex here X and I'm trying to write down n a dot RM ,that means, NA value has to be removed and this option is true or false, this option is true. by writing T, that means all the any value has have to be removed and the automatic mean has to be calculated on the basis of available numerical values .so, in case if I try to write down, this command here, then this automatic min, will be found as 10 plus, 20 plus, 40 divided by 3 not 4 . Right? And then I will get here a value here 23.3 3 and so on. So, this is how we try to work when some values are missing, but let me show you this thing over the R console.

Refer slide time: (41:10)

```
> x=c(10,20,NA,40)
> x
[1] 10 20 NA 40
> mean(x)
[1] NA
>
> mean(x, na.rm=TRUE)
[1] 23.33333
>
> mean(x, na.rm=T)
[1] 23.33333
>
> mean(x, na.rm=True)
Error in mean.default(x, na.rm = True) : object 'True' no$
> mean(x, na.rm=t)
Error in if (na.rm) x <- x[!is.na(x)] :
  argument is not interpretable as logical
> |
```

So, let me take here, the data here, X to be here 10, 20 say here, n a, and here 40. Right? so you can see here, this is my data here and if I try to write down here mean of X, this is giving you me, me, me here NA and ,and if I try to find out here mean of X, n a dot R M, is equal to true, true, then you get here ,the value twenty three point three three .so, here means again just for the sake of illustration, I will try to show you that instead of here using the entire word TR u, I can also use here, capital T and the answer is coming out to be same, but on the other hand, in case if I try to make it here, say small letters ,say are you in small letters quickest, will give me an error and even if I try to find out this with only n a dot R M ,is equal to a small T, that is not capital T, this is going to be an error .so ,this is how we try to handle missing values .

Refer slide time: (42:13)

Example : How to work with missing data

The null object, called **NULL**, is returned by some functions and expressions.

Note that **NA** and **NULL** are not the same.

Note that **NA** and **na** are not the same.

```
R Console
> x<-c(10,20,na,40)
Error: object 'na' not found
```

NA is a placeholder for something that exists but is missing.

NULL stands for something that never existed at all.

But, before concluding the Lecture, I would try to inform you, that in R, in some places, you will be getting an outcome, like here and you Double L in Capitol Visitor's ,this is going to be the outcome, which is retained by some of the function, remember one thing, na and null they are not the same thing and even this capital n a and is small NA, they're also not the same thing capital N, capital a is a result word and this is case sensitive, the difference between NA and na panel is the following na is a place holder, place holder means, yes. Inside the class, a student has been assigned a seat, but the student is absent today, it does not mean that a student does not exist, so in this case the student is missing, so that is going to be given by NA, not available, at that point, but in case if I am trying to use the word, here and you double L, this null stands for something, which never existed .so, that is the difference between the use of NA, null, which we have to be careful in the while we try to use it, so here I would like to stop and I would request, you that you please, try to attempt your assignment, try to take some exercises from the books, or even you can create your own exercises, just try to take ,take our data set ,do small manipulations and verify them with their manual calculation, that are you getting the same thing, which you used to get manually, try to replace some data set in that data vector by NA and try to C, what is really happening? If one value is missing, two values missing and even if no value is missing, how you are going to obtain or how you are going to interpret the value of this logical operator to end false. Right? So, you practice and we will see you in the next lecture, till then good bye.