

**Essentials of Data Science with R Software – 2**  
**Sampling Theory and Linear Regression Analysis**  
**Prof. Shalabh**  
**Department of Mathematics and Statistics**  
**Indian Institute of Technology, Kanpur**

**Sampling Theory with R Software**  
**Lecture – 16**  
**Simple Random Sampling**  
**SRSWR and SRSWOR with R with “sample” Package**

Hello, welcome to the course Essentials of Data Science with R software-2, where we are trying to learn the topics of Sampling Theory and Linear Regression Analysis. And in this module, we are going to continue to learn the topics of Simple Random Sampling.

And as you know we are handling the topic of simple random sampling. And in the last couple of lectures, we have understood the basic fundamentals and more importantly, we have understood how we are going to select a sample using the simple random sampling with and without replacement.

And we also have computed their probability. So, all those things are now clear to us. And today after learning this much, we are in a position that we have understood how are the samples drawn manually.

But now once you come to data science things are not going to be manual, they are based on software, they are based on different type of programming language. So, the question today is this, whatever we have done up to now means we have drawn, essentially the samples using simple random sampling methodology under with and without replacement?

So, how to draw the samples using R software? Now, there are two questions. One question which I would like to address in this lecture and say another question in the next lecture. The questions are- first question is this suppose I have got simply some numbers say 1 to N. And I simply have to select a sample of size n from those numbers. So, essentially I have to select only the unit number, that is the first question. So, this I am going to do it now.

The next question which I would, which I will try to address in the next lecture is that suppose you have got a data set, the data set has various columns, say for example, students roll number first column, second column is their marks in mathematics, third column is marks in physics, fourth column is marks in chemistry, and they have suppose total. This is your data set.

And suppose there are 500 students, and you want to select a sample of size say 20. So, what do you want? You do not want only the student number or the student roll number. Suppose, the roll numbers are from 1 to 500.

You do not want only the 20 roll numbers, but you also want the information which is in the rows for those selected 20 roll numbers. So, how to select the information corresponding to a particular unit id, not only unit id, but also the information on that unit id will be a topic which I will try to address in the next lecture.

So, today we start and we are going to use here a package- sample. Actually, there are two popular packages which are useful for us while dealing with the simple random sampling. So, first I will try to show you all this illustration using the sample package. And another package is sampling. So, after that I will continue and I will try to show you how to do the same thing under the sampling packages.

And in R when you are trying to work to achieve certain objective, there is a good possibility that different people have contributed different software to do the same thing. But they may also have different syntaxes, they have different commands, so that is what I want to illustrate you today and different packages may have different types of facilities also.

So, once you have an objective, once you know that what you really want to do, first you have to spend some time and you have to explore that what are the different packages available, and what is more convenient to you, and which will give you the required information.

And this process is a continuous process. Why? Because R is continuously developing and that is the advantage that as soon as there is an updated version you can have the

most recent version. And it is possible that in the recent in the most recent version something may change.

So, you have to look for those thing and you have to be careful. So, let us start our lecture that how to use the package whose name is sample - s a m p l e for drawing different types of samples in SRSWR and WOR, ok.

(Refer Slide Time: 05:35)

**Using R software: How to draw simple random sample**

**sample** takes a sample of the specified size from the elements of **x** using either with or without replacement.

**Usage**

```
sample(x, size, replace = FALSE, prob = NULL)
sample.int(n, size = n, replace = FALSE, prob = NULL)
```

Handwritten annotations: 'Propr' with an arrow pointing to 'size', 'n' with an arrow pointing to 'size', 'FALSE: WOR' with an arrow pointing to 'replace = FALSE', and 'TRUE: WR' with an arrow pointing to 'replace = FALSE'. In the second line, 'n' has an arrow pointing to 'size = n' and '1, n' has an arrow pointing to 'size = n'.

**Arguments**

- x** Either a vector of one or more elements from which to choose, or a positive integer.
- n** a positive number, the number of items to choose from.

So, this sample - s a m p l e is a R package which helps in taking a sample of the specified size from a population which is stored under a variable say x. And this sample package helps us in drawing the sample with and without replacement both. The rule or the command or the syntax to draw the sample is first you have to write down sample - s a m p l e. Then inside the parenthesis, there are few options which are optional, and few options which are compulsory.

So, first you have to compulsorily give the data vector x which is your actually population, that means, you have to specify from which of the population you want to draw the sample. Then followed by comma, there is another parameter- size. Size will give you an idea about n, so that means, how many units or what is the sample size which you would like to draw from the data vector x which is of size N. Then there is another option replace.

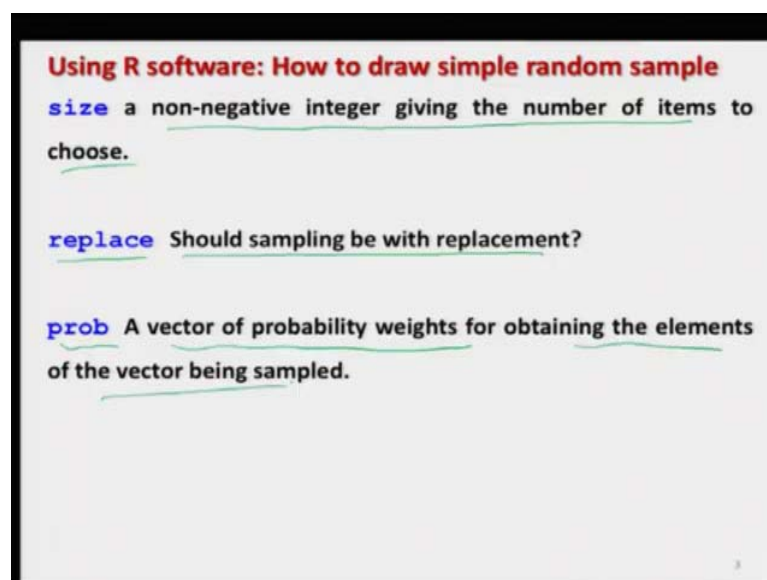
This replace is a logical parameter which can take here two possible values TRUE and FALSE. And TRUE and FALSE, they are going to indicate whether the sampling scheme is SRSWR that is with replacement or without replacement.

So, that goes by the logic of the word, if you try to read it here this is replace is equal to FALSE, that means, it is trying to say please do not replace. So, that means, when is you are trying to use here the command say option FALSE that mean this is corresponding to do not replace that is without replacement.

And when you try to use replace is equal to TRUE, that means, yes, you want replacement. So, this is w R with replacement. And another option here is sample dot int. This is actually only a sort of interface and where this n, and this size equal to n has to be integers. And obviously, I mean the sample size can be between 1 to n that cannot exceed the population size in case of SRSWOR, right.

So, similarly I will give you the details of all these options through example in the next couple of slides, right. So, now, let us try to move to the next slide. So, I already have explained you here what is here x, and what is here actually n which is here the sample size in these cases, and what is here the size, right.

(Refer Slide Time: 08:53)



So, size is once again I am trying to give you a brief details for your records that size in is a non-negative integer which gives the number of item to be chosen which is your

sample size. Replace is a command which ask you should sampling be without replacement or not, so that I already have explained you. Probability is something like which I am not considering here because in your case the probabilities are always going to be the same.

But definitely if you go to the some more sampling scheme there you have to give, where you have to use this function or where you have to use this parameter `prob` where you have to assign a vector of probability weights for obtaining the elements of the vector being sampled ok. But anyway, we are not going to use this. So, I am not giving you more detail.

(Refer Slide Time: 09:40)

**Using R software : How to draw simple random sample**

For `sample` the default for `size` is the number of items inferred from the first argument, so that `sample(x)` generates a random permutation of the elements of `x` (or `1:x`). *sample(x)*

**Values** *x = 1, 2, 3, 4, 5*  
*sample(x) = \_\_\_\_\_*

`sample.int` is a bare interface in which both `n` and `size` must be supplied as integers.

For `sample` a vector of length `size` with elements drawn from either `x` or from the integers `1:x`. *→ 1, 2, ..., x.*

For `sample.int`, an integer vector of length `size` with elements from `1:n`. *← 1, 2, ..., n*

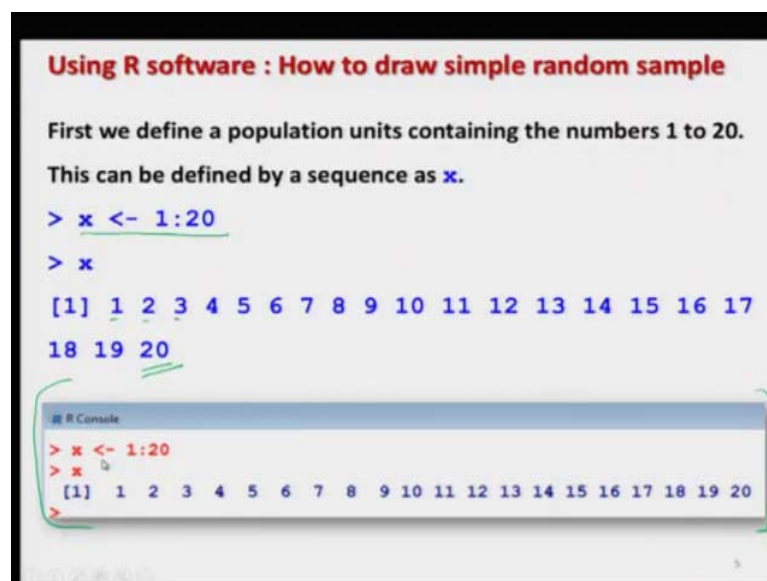
And in case, if you try to write only one parameter, do not write anything, but you simply write here `sample` and inside the parenthesis just write small `x`. So, actually this command will simply generate a random permutation of the elements of `x`, that means, suppose `x` is here 1, 2, 3, 4, 5, this is of size five. So, once you say here only `sample x` and do not give any other parameter, this will generate a sample of size five using this any of this 1, 2, 3, 4, 5 numbers, right.

For example, that can be 1, 2, 3, 4, 5; 5, 4, 3, 2, 1; 4, 3, 2, 1, 3 and so on. And as I said earlier `sample.int` is simply an interface in which both `n` and `size` must be supplied as integer. And the only difference is that when you are using the command `sample`, in

that case a vector of length which was actually parameter size are drawn from the data vector  $x$  or they are drawn from the integer 1 to  $x$ ;  $1$  colon  $x$  means that means 1, 2, 3, 4, ...,  $x$ .

And when you are trying to use the sample dot i n t , then what you have to do? You have to look into the parameter size. And this size is an integer vector of length, and whose elements are 1, 2, 3, 4 , ...,  $n$  which are indicated by  $1 : n$ . So, that is the very simple difference between the two.

(Refer Slide Time: 11:29)



```
Using R software : How to draw simple random sample

First we define a population units containing the numbers 1 to 20.
This can be defined by a sequence as x.

> x <- 1:20
> x
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
    18 19 20

R Console
> x <- 1:20
> x
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
```

So, now let me try to take here some example. First I will try to show you on the slides. And I already have executed these examples, and I have pasted here the screenshots also. But there will be one problem. What problem? That I will try to show you in the next slide right ok. But now first let us create our population.

So, just for the sake of simplicity in understanding I am defining a sequence 1 to 20, that means, all integer 1, 2, 3, , ..., 20; and I take it as my data vector here  $x$  that is my population, right. So, you can see here  $x$  is equal to 1, 2, 3, 4, ..., 20. And this is here the screenshot. And now I will try to draw different types of samples using this population to explain you the different types of application.

(Refer Slide Time: 12:27)

**Using R software : How to draw simple random sample**  
 Let us draw the sample of size 5 from population  $x$  by SRSWOR .  
 This is controlled by the statement **replace = FALSE** inside  
 the argument.

$N = 20, n = 5$  } SRSWOR  
} SRSWOR

```
> x
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
    18 19 20
```

**SRSWOR command**

```
sample(x, size=5, replace = FALSE)
```

So, suppose we decide that we want to draw a sample of size 5. So, here in this case  $N$  is 20, and  $n$  is here 5. Now, I have here two options; I can use here SRSWOR or I can use here SRSWR. So, as already discussed in order to differentiate between the SRSWR and WOR, we have here a parameter `replace` which can take two possible values logical TRUE and logical FALSE.

So, in case of SRSWR, you do not want to replace. So, the `replace` will be FALSE. So, the entire command to choose a sample of size five from the population of size 20 given in the data vector  $x$ , the command is `sample(x, size=5, replace=FALSE)`, ok.

(Refer Slide Time: 13:29)

**Using R software : How to draw simple random sample**

**SRSWOR**

```
> sample(x, size=5, replace = FALSE)
[1] 15 1 10 11 5 ✓✓
```

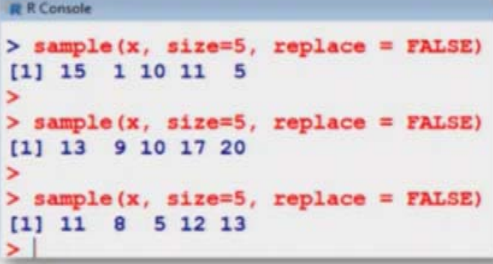
$P\left(\frac{n=r}{N=20}\right) = \frac{1}{\binom{20}{5}}$

```
> sample(x, size=5, replace = FALSE)
[1] 13 9 10 17 20 ✓✓
```

```
> sample(x, size=5, replace = FALSE)
[1] 11 8 5 12 13 ✓✓
```

(Refer Slide Time: 13:36)

**Using R software : How to draw simple random sample**



```
# R Console
> sample(x, size=5, replace = FALSE)
[1] 15 1 10 11 5
>
> sample(x, size=5, replace = FALSE)
[1] 13 9 10 17 20
>
> sample(x, size=5, replace = FALSE)
[1] 11 8 5 12 13
>
```

Now, if you try to see I am trying to execute this command on the R console and ok, in the next slide, you can see here I have given the screenshot, but those values are here. But let us try to first understand. So, if you try to observe here when I execute this command on the R console, I get here an outcome 15, 1, 10, 11, 5. And if I try to repeat this command, then I get here another sample here 13 9 10 17 and 20. And if I try to repeat it once again this is 11, 8, 5, 12, 13.



Now, you can see here this sample in the first case, sample in the second case and sample in the third case, they are all together different. So obviously, you are trying to choose 5 numbers out of 20 numbers at random. So, the possibility or the probability that the two samples are going to be the same that is very very less. Why? Because if you try to see the probability of selection of a sample of size 5 from the population of size 20, this was  $1/\binom{20}{5}$  which is going to be a very small quantity.

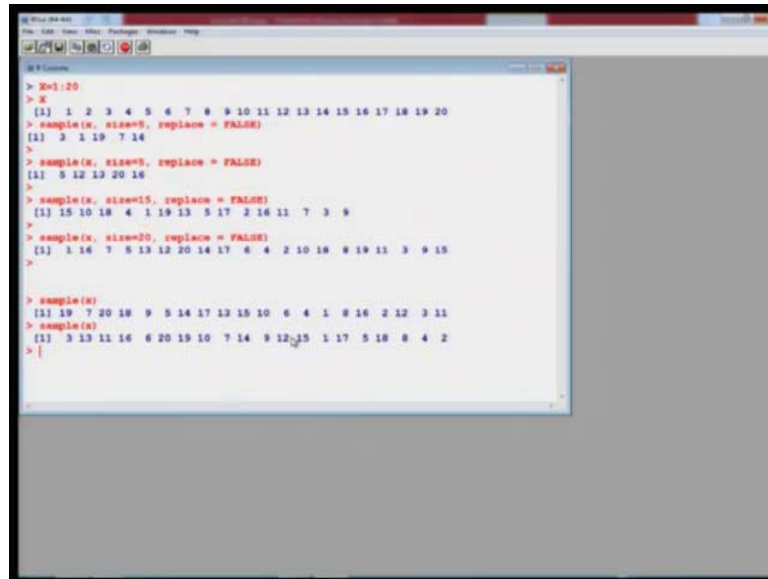
And, and if you are usually the population size will be very very large and the sample size will be reasonably large. So, this probability is going to be very close to 0 and that is why we say that the probability of repetition of a sample is extremely less. So, you can see here that using the same command, I have generated three possible samples. And these are here the screenshot.

Now, what is my trouble? My trouble is the following, that up till now whatever data I have taken while explaining you the R commands, whatever I had done at the time of preparation of the slides, the same outcome was available when I was doing it during the lecture. But when you are trying to draw a random sample, this is practically not possible that if I try to repeat the same command here during the lecture, I will get the same outcome the possibility is extremely less.

So, that is what you have to keep in mind that when you will try to look into the lectures. Whatever results I have given you on the slide they may not really match with the results which I will be showing you on the R console.

So, please do not get confused. These are random samples, and means every time you repeat them they will give you a different value, and that is basically the concept, ok. So, now, if you try to see first let me try to show you this thing on the R console over here.

(Refer Slide Time: 16:54)



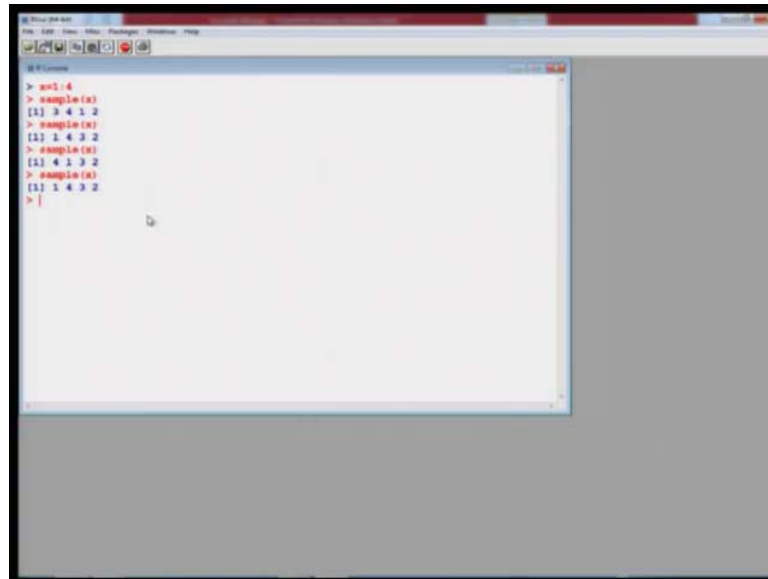
```
> X=1:20
> X
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
> sample(x, size=5, replace = FALSE)
[1] 3 1 19 7 14
>
> sample(x, size=5, replace = FALSE)
[1] 8 12 13 20 16
>
> sample(x, size=15, replace = FALSE)
[1] 15 10 18 4 1 19 13 5 17 2 16 11 7 3 9
>
> sample(x, size=20, replace = FALSE)
[1] 1 14 7 5 13 12 20 14 17 4 4 2 10 18 8 19 11 3 9 15
>
> sample(x)
[1] 19 7 20 18 9 5 14 17 13 15 10 6 4 1 8 16 2 12 3 11
> sample(x)
[1] 3 13 11 16 6 20 19 10 7 14 9 12 15 1 17 5 18 8 4 2
> |
```

So, first I try to define here my x, I come on the R console. And suppose I try to define here x is equal to 1 to 20. So, you can see here this is my here x, and I try to define here say sample here x. And I try to sample capital X, I have taken. So, this is my here x. So, you can see here this will come out to be like this. So, it is trying to choose unit number 3, 1, 19, 7 and 14.

And if I try to repeat it here, you can see here, you will get a sample of size 5, but which is different from the first sample. And similarly if you want to have a sample of size see here more, for example, size of 15, you can see here, you are getting a sample of size 15 the 5 units are not there.

But if you try to take let me show you, if you try to take here a sample of size 20, that means, you are going for complete enumeration or senses you can see here that you have simply got a random permutation of the numbers 1 to 20. And the same result I can obtain when I try to give here sample x see here sample of here x. So, you can see here that this is only a random permutation just like what I did here in this command. And if you try to repeat it, this will give you 20 values, but they are once again different.

(Refer Slide Time: 18:43)



```
> x=1:4
> sample(x)
[1] 3 4 1 2
> sample(x)
[1] 1 4 3 2
> sample(x)
[1] 4 1 3 2
> sample(x)
[1] 1 4 3 2
> |
```

And I can show you this example here more clearly just by taking  $x$  is equal to here one to say here, suppose, I will let me take it a 4. Now, you can see here sample of here  $x$ . This is 3, 4, 1, 2. And if you try to repeat it, this is 1, 4, 3, 2; if you try to repeat it, 4, 1, 3, 2; if you try to repeat it, 1, 4, 3, 2. So, now, you can see here this, and this they are repeated. But you are just taking a population of size 4, so not many combinations are possible. So, this is happening.

So, this is what I meant when I say that if you try to use the command- sample with the data vector  $x$  without giving any other option, then you will get a random permutation of the numbers in the  $x$  data vector. So, now let us come back to our slides and now I will show you that how you can draw or you can use the same command to draw sample using SRSWR, ok.

(Refer Slide Time: 19:42)

**Using R software : How to draw simple random sample**

Let us draw the sample of size 10 from population  $x$  by SRSWR .

This is controlled by the statement `replace = TRUE` inside the argument.

```
> x
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
18 19 20
```

$N = 20$   
 $n = 10$

**SRSWR Command**

```
sample(x, size=10, replace = TRUE)
```

So, now you understand that is pretty simple that if you want to use the command `sample` to draw the SRSWR sample, then you simply have to use the option here, `replace` equal to `TRUE` instead of `replace` equal to `FALSE`. So, now suppose I consider here the same population where I have numbers from 1 to 20, and here I try to draw a sample of size 10 ok.

Well, you can ask me why I am not taking here a sample of size 5, because my objective is this I want to show you that how the values are repeated. And if I try to take such a small sample size 5, possibly I will have to take many more sample to show you that the values are repeated that is the simple reason. So, in this case my here  $N$  is 20, and  $n$  is my here 10.

And so the command becomes here `sample`, the data vector  $x$ , sample size is equal to 10, and `replace` is equal to `TRUE`. And so now, in the next slide, I have taken some example which I already have executed on the R console.

(Refer Slide Time: 20:50)

**Using R software : How to draw simple random sample SRSWR**

```
> sample(x, size=10, replace = TRUE)
```

[1] 4 17 6 3 20 14 13 2 15 2 ✓

Value 2 is repeated.

```
> sample(x, size=10, replace = TRUE)
```

[1] 5 1 7 4 18 2 12 1 3 7 ✓  $\frac{1}{20^{10}}$

Values 12 and 7 are repeated.

```
> sample(x, size=10, replace = TRUE)
```

[1] 15 11 19 10 4 3 11 17 9 3 ✓

Value 11 is repeated.

And I have copied and pasted their outcome. But as I said earlier I will try to show you that how are you going to use this command on the R console, but that may possibly not result in the same outcome which is reported in my slides, because these are random sample ok. So, if I try to use the command here sample on size x and for a n that is size equal to 10 and replace equal to TRUE that is SRSWR.

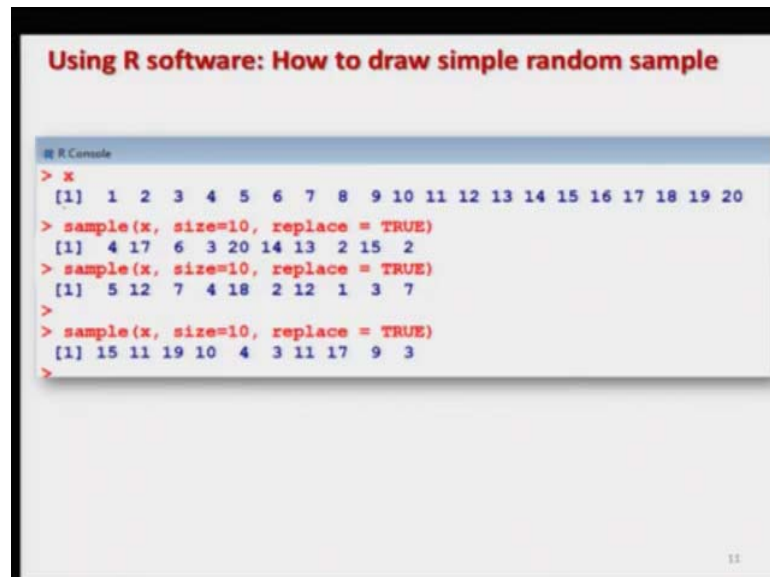
In that case, I get here a here an outcome like this one, I have got here 10 values which are chosen from the data vector x consisting of the values 1 to 20. And here you can see that the value 2 here is repeated 2 times. And similarly, if I try to repeat the same command, I obtain here another data vector which is a sample. And in this sample, there are two values which are repeated; one is here 12 and another here is 7. So, 12 and 7, they are repeated.

So, it is possible as I said that this is going to be SRSWR. So, more than one values may also repeat in the sample. And an extreme case will be that the same value is repeated 10 times. But again the probability of such an event will be very very less. Why?

Because in this case if you try to see the probability of selecting a sample is  $1/N$  that is  $20$  raised to the power of here n which is 10. Now, you can compute this probability and you can see that this is very very small. So, the probability of repetition of a sample is extremely small.

And similarly, if I try to repeat the same command here then I get here one more sample where you can see that this 11 unit is repeated. And you can see that every time I repeat the command, I get here this sample, this sample and sample number 3 which are all together different.

(Refer Slide Time: 23:07)

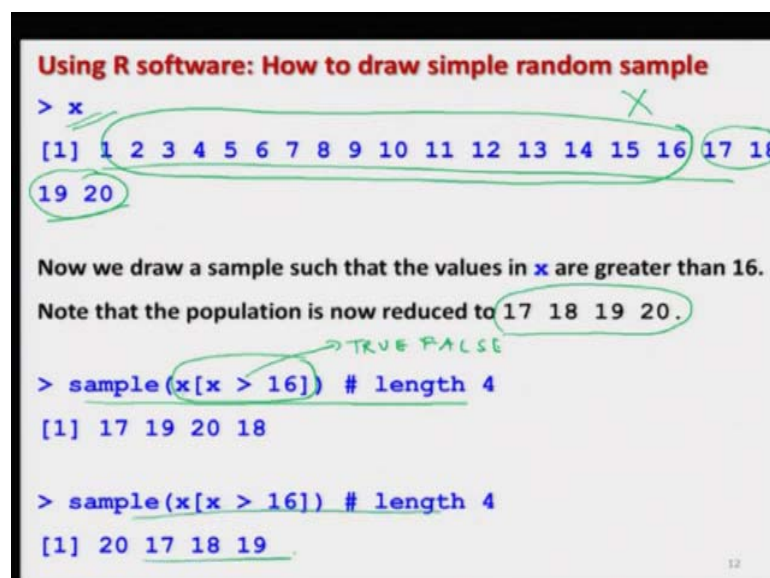


```
Using R software: How to draw simple random sample

R Console
> x
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
> sample(x, size=10, replace = TRUE)
[1] 4 17 6 3 20 14 13 2 15 2
> sample(x, size=10, replace = TRUE)
[1] 5 12 7 4 18 2 12 1 3 7
> sample(x, size=10, replace = TRUE)
[1] 15 11 19 10 4 3 11 17 9 3
```

And you can see here this is the screenshot. So, you can believe on me, yes that I had executed these things.

(Refer Slide Time: 23:14)



```
Using R software: How to draw simple random sample

> x
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
19 20

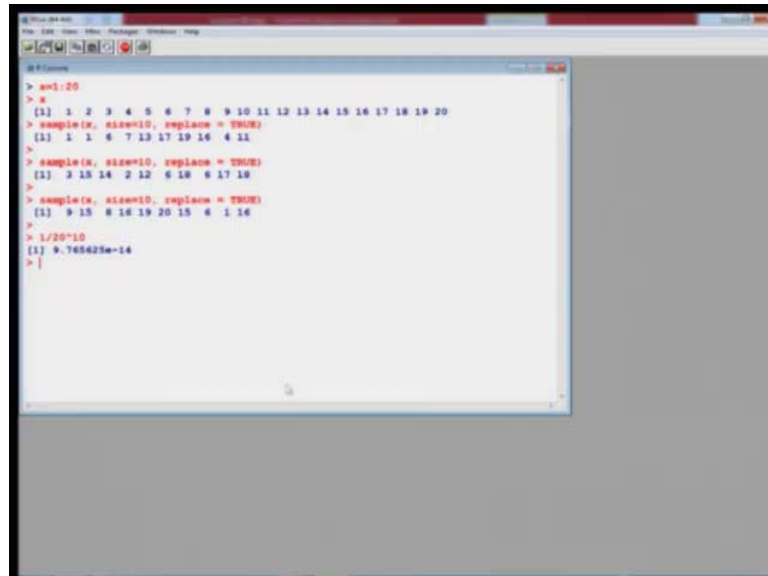
Now we draw a sample such that the values in x are greater than 16.
Note that the population is now reduced to 17 18 19 20.

> sample(x[x > 16], # length 4
[1] 17 19 20 18

> sample(x[x > 16], # length 4
[1] 20 17 18 19
```

And before I go further let me try to show you these all these things on the R console first, and then I will try to move further.

(Refer Slide Time: 23:31)



```
> n=1:20
> x
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
> sample(x, size=10, replace = TRUE)
[1] 1 1 6 7 13 17 19 16 4 11
>
> sample(x, size=10, replace = TRUE)
[1] 3 13 14 2 12 6 18 6 17 18
>
> sample(x, size=10, replace = TRUE)
[1] 9 15 8 16 19 20 15 6 1 16
>
> 1/20^10
[1] 9.765625e-14
> |
```

So, let me try to copy the command, and I come to R console. And I need to define my here x once again because last time I had changed it. So, x is my here 1 to 20, you can see here. Now, I try to say here please draw a sample of size 10 from the population of x by SRSWR. And you can see here that in this case unit number 1 is being repeated 2 times. And similarly, if I try to repeat this command here, so you can see here is there any unit which is repeated? Yes, this is here 18; 18 is repeated here two times.

And similarly, if you try to take here one more sample over here, you can see here which of the unit is repeated, for example, here you can see here 16 is repeated, and 15 is also repeated here. So, this is how you can draw different types of samples, and every time you will get a new sample. And well, if you want to compute here the probability of drawing a sample that is  $1/20^{10}$ . So, you can see here, this is  $9.765 \times 10^{-14}$ .

One thing you have to learn that in R these numbers are coming as e minus 14, that means, the number multiplied by  $10^{-14}$ . So, this is the probability of choosing one sample which is very very small that is what I was explaining you in the slides right.

Now, I try to address here one more issue that is a very simple thing, but that is possibly the need in the data sciences, right.

Suppose, you have got a big data set, and you want to select some specific values. For example, suppose you have got a data of say this say 20 million people whose ages are reported. And you want to draw a sample of required size from those people whose ages are more than 30, 30 years, or I can reframe this question once again. Suppose, you have got the data of people staying in a city; and you want to draw a sample of the young people from that city.

And suppose you decide that I would like to have a sample of size 500 people whose ages are between 20 and 25. That means what you need to do? You have to do two steps. First step is this, you have got a huge data set mean you have got the ages of everybody staying in that city or a state. First step is this you have to select all those values in that population where the ages are between say 20 years and 25 years. Now, this becomes your population.

And then you have to select a sample from that population. Well, I can give you several example from our real life. For example, now a days many people are doing shopping on online mode through some various websites, popular websites. And the persons who are owing the website or that is their actually shop, they would like to know that who is visiting their site more. And based on that, they can put more things on their site.

Suppose, they take an example of clothings. Suppose, they find out that on their website younger people are visiting more than the elder people. But the first question is how they will try to find it out? For that suppose, for a given day or for a given week, they have to select who have visited their websites. They have their all the information because that because that information is transferred to them when we try to do any online shopping.

First we have to create a login id. So, in that login id, we try to give our all information. And we believe that this is correct like as name, age, address, etc. So, now, they will so from all the logins, they will try to see how many people are there in the age group of say up to 10 years, 10 years to 20 years, 20 years to 30 years, 30 years to 40 years and so on, and then they will try to see there are lots of people who are between the ages 20 and 25, and they are interested in a particular type of clothing.



We all understand that as a younger we always try to wear more fashionable clothing, more fashionable dresses, but when you become elder like me possibly you will always be wearing a sky blue shirt or a white shirt, so that is the reason. So, now they want to know if younger people are striking their website more, then they will try to put more fashionable clothes that will increase their sale.

So, now, they have two options. They have got the population which is the total number of visitors in that site on a given day on a given week that will be in millions and billions, and then they will try to select only those people whose ages are between say 20 years to 25 years or 20 years to 30 years. Now, the definition of population has changed, now this becomes your population.

And from these selected people between the age of 20 and 25, they will try to choose a random sample. And then they can have information, they can contact them, they can have feedback or they can give some discount coupons to them. Once they give the discount coupon towards collected people, then the news will spread in the market or among the consumer that there are some lucky people who are getting the discounts coupon. So, more number of people will start visiting their website. And when they are visiting the website, there is a good possibility that they may buy something. So, their, their sales will increase, so that is the role of data sciences means earlier the sampling theory was mainly concentrated to agricultural experiment or smaller experiments that is the change. But without learning this sampling techniques or this or this sampling methodologies, you cannot execute it.

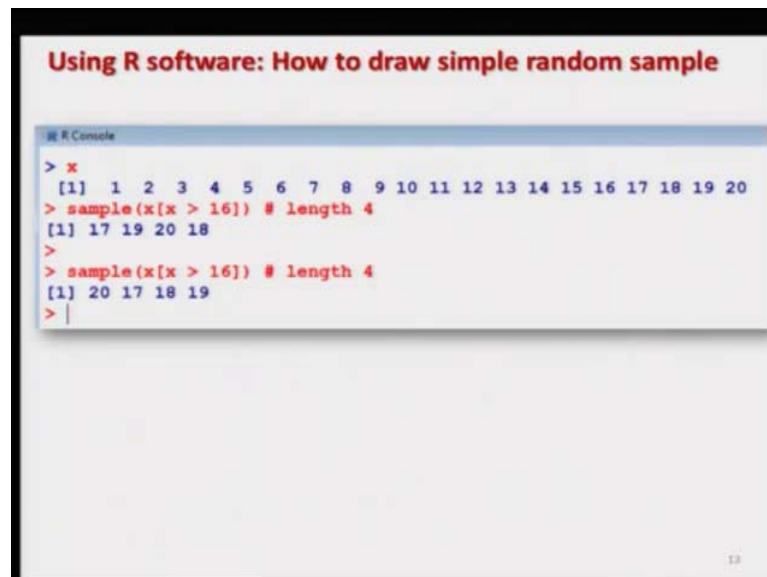
So, that is the same example in a very small way in a small miniature I am trying to give it here, right, ok. So, let us try to do it. So, now, I am taking the same population  $x$  which are the numbers between 1 to 20. And suppose I want to draw a sample of size say here 4 in which the values are at least more than or greater than 16. So, for example, in this sample you can see here from 1 to 16, this part will not remain as a size as a part of the population. And my sample will contain only here 4 values.

So, now, yes that is not a very relevant question, but I will try to show you by example that how you can modify it. So, suppose I want to choose here a sample of size four that then obviously I have only here four numbers, 17, 18, 19, and 20. So, in case if I try to

use here the command sample, the sample will give me only a random permutation of the value 17, 18, 19, 20.

For example, you can see here when I execute this command over here. So, that we already have learnt during the R commands that if I try to use this command over here as say x greater than 16 will give us the value in terms of TRUE and FALSE. And when I want to count that what is the number of TRUE, then I have to use here the command x. So, this will become x, inside the square bracket x greater than 16, right. And if you try to repeat it here, you will get the same thing over here, ok.

(Refer Slide Time: 32:56)

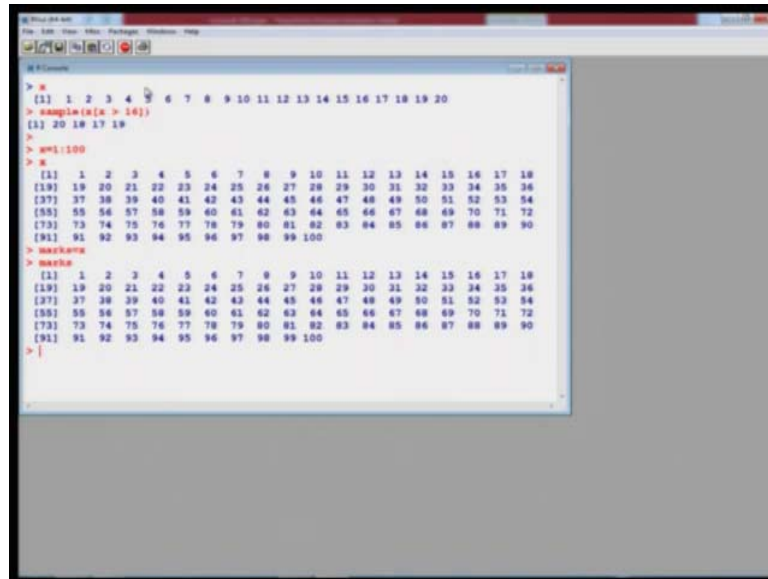


```
Using R software: How to draw simple random sample

R Console
> x
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
> sample(x[x > 16]) # length 4
[1] 17 19 20 18
>
> sample(x[x > 16]) # length 4
[1] 20 17 18 19
> |
```

So, now and you can see here this is the screenshot. But before that I will try to show you it on the R console also. And I will try to show you here something more also.

(Refer Slide Time: 33:11)

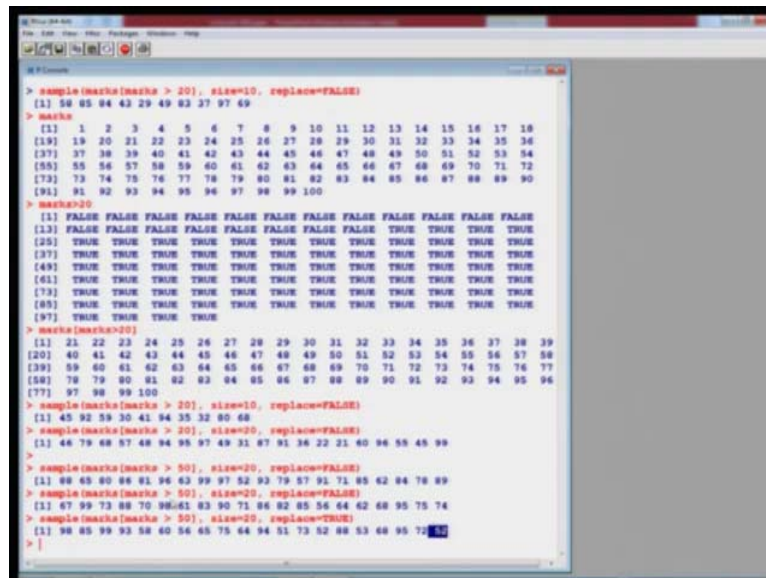


```
> x
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
> x[x > 16]
[1] 17 18 19
> marks
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
[19] 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
[37] 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54
[55] 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72
[73] 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90
[91] 91 92 93 94 95 96 97 98 99 100
> marks[marks > 70]
[1] 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90
[91] 91 92 93 94 95 96 97 98 99 100
> |
```

So, we come to R console. And you can see here that we already had defined x equal to 1 to 20. And now I can use here this thing. So, you can see here these are the values which are which are being obtained from the number which are more than 16. Now, I try to do it here a better example that will give you.

For example, if I try to take here x equal to 1 to 100, ok. Now, you can see here this is my here 100. Now, I try to find out suppose these are the marks of students in the examination. So, I can give it here a name here say marks is equal to x. So, now, these are the marks of 100 students which have been obtained in different in an examination from 1 to 100.

(Refer Slide Time: 34:11)



```
> sample(marks[marks > 20], size=10, replace=FALSE)
[1] 58 85 84 43 29 49 83 37 97 69
> marks
 [1] 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18
[19] 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
[37] 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54
[55] 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72
[73] 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90
[91] 91 92 93 94 95 96 97 98 99 100
> marks>20
 [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE TRUE TRUE
[25] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
[37] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
[49] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
[61] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
[73] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
[85] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
[97] TRUE TRUE TRUE TRUE
> marks[marks>20]
 [1] 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39
[20] 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58
[39] 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77
[58] 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96
[77] 97 98 99 100
> sample(marks[marks > 20], size=10, replace=FALSE)
[1] 45 92 59 38 41 94 35 32 80 68
> sample(marks[marks > 20], size=20, replace=FALSE)
[1] 46 79 68 57 48 84 95 97 49 31 87 91 36 22 21 40 96 55 45 99
> sample(marks[marks > 50], size=20, replace=FALSE)
[1] 88 89 90 86 81 84 82 99 97 92 92 79 87 92 71 85 82 84 78 89
> sample(marks[marks > 50], size=20, replace=FALSE)
[1] 67 99 73 88 70 98 61 83 90 71 86 82 85 56 64 62 68 95 75 74
> sample(marks[marks > 50], size=20, replace=TRUE)
[1] 98 85 99 93 58 60 56 65 75 64 84 51 73 52 88 53 68 95 72
```

Now, I want to draw here a sample here of say size 20 from those students who have got say here marks say more than 20. And I want to have a sample of size say equal to here say 10. And whether I want replace or not that is now it is your choice, so replace I am saying no, please do not replace it. So, this is equal to FALSE. So, you can see here this is my outcome.

So, what it has done? Initial population was 1 to 100. And once I try to take the command marks greater than 20, it has gave me the answer in terms of TRUE and FALSE. You can see here first 20 values, they are here FALSE; and remaining values which are more than 20 they are here TRUE. So, now once I try to use here the command here, see here, marks which are greater than 20, this will come out to be like this you can see here.

So, now, this becomes my new population. And now I am trying to sample 10 observation from this population using the command this one. And suppose if you want to sample here say more observation say 20 observation, you will get here like this. And suppose if I want to say no, I want 20 observation from those people who have got more than 50 marks, so you can see here this is the sample of size 20 from the population of those students who have got who have secured more than 50 marks.

And if you try to repeat it, you will again get the different sample. And if you want to obtain this sample by SRSWR, you simply have to replace, replace by TRUE. So, you can see here. And you can see here, for example, I can see very quickly that the value 52 has been repeated at least twice, rest you can find. So, this is the application of sampling theory in the current marketing strategies which are being used by different people over here, right.

(Refer Slide Time: 36:34)

```

Using R software: How to draw simple random sample
> x
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
18 19 20

For sample the default for size is the number of items inferred
from the first argument, so that sample(x) generates a
random permutation of the elements of x (or 1:x).

> sample(x)
[1] 19 2 1 7 12 15 4 14 13 5 10 17 6 16
18 9 20 3 8 11

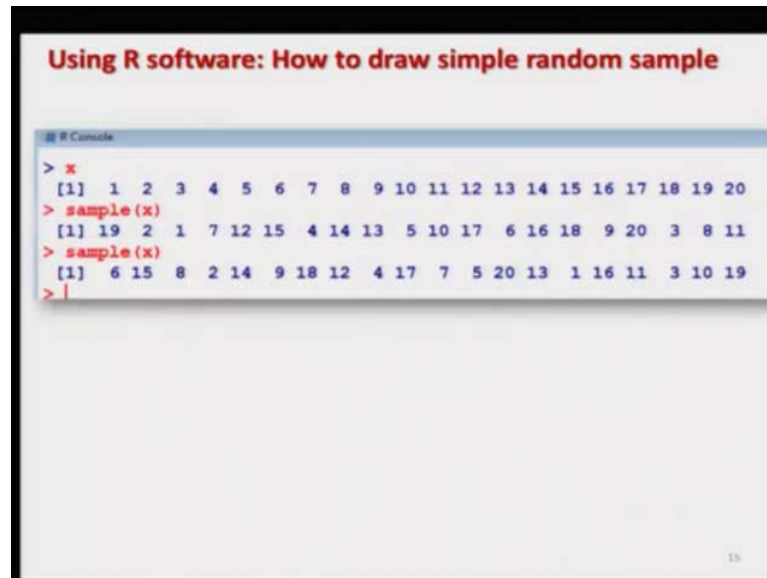
> sample(x)
[1] 6 15 8 2 14 9 18 12 4 17 7 5 20 13
1 16 11 3 10 19
  
```

Handwritten annotations in the image include: a green circle around the `x` variable; a green circle around the `sample(x)` command in the first example; a green circle around the `sample(x)` command in the second example; and a green circle around the second resulting vector. A small diagram shows `1, 2, 3 → 1 3 2` and `2 5 1`.

And as I said here you simply want to have here a sample. Then you can see here if you try to use here the sample command here, sample only here x, here like this, suppose I take the same say the population x going from 1 to 20. And if you simply try to take here the command here sample x, that means, you have to just obtain a random permutation of the values 1 to 20.

So, you can see here you have got here 20 values, but their order is different from 1 to 20. And if you try to repeat the same command, you again get here these 20 values, but their order is entirely different. What is called the random permutation? The random permutation means if you have the values 1, 2, 3 means I can arrange it here 1, 3, 2, 2, 3 1 and so on, right. So, that you already have done in your, I think class-10 or class-12, this combinations and permutation. All those examples you already have done ok.

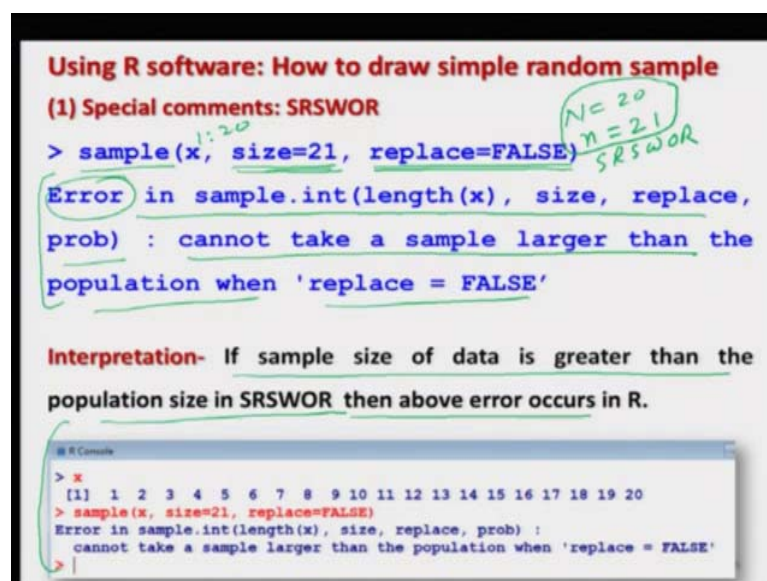
(Refer Slide Time: 37:42)



```
Using R software: How to draw simple random sample

R Console
> x
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
> sample(x)
[1] 19 2 1 7 12 15 4 14 13 5 10 17 6 16 18 9 20 3 8 11
> sample(x)
[1] 6 15 8 2 14 9 18 12 4 17 7 5 20 13 1 16 11 3 10 19
> |
```

(Refer Slide Time: 37:47)



```
Using R software: How to draw simple random sample
(1) Special comments: SRSWOR

> sample(x, size=21, replace=FALSE)
Error in sample.int(length(x), size, replace, prob) :
cannot take a sample larger than the population when 'replace = FALSE'

Interpretation- If sample size of data is greater than the
population size in SRSWOR then above error occurs in R.

R Console
> x
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
> sample(x, size=21, replace=FALSE)
Error in sample.int(length(x), size, replace, prob) :
cannot take a sample larger than the population when 'replace = FALSE'
> |
```

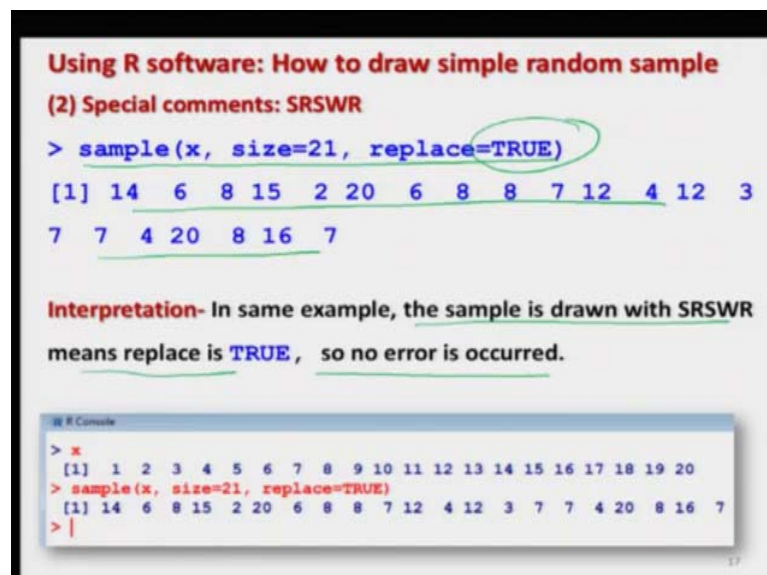
Now, this is the screenshot of the same command which I explained you here. And now I will try to show you two possible mistakes. The mistakes are like here like this. If you try to see here I am trying to take here the command sample, my population is the same x, x is from 1 to 20. And I am drawing a sample of size here 21, but I am making here replace equal to FALSE.

And if you try to do this thing on the R console, this will give you this type of message which is an error. And you can see here the screenshot also here. What it is trying to say? Error in sample dot int length x size etc., etc., etc., cannot take a sample larger than the population when replace equal to FALSE. So, what are we trying to do? My population size here is 20. And my sample size what I am asking it is 21.

And I want to use here SRSWOR. So, how you can draw a sample of size 21 from a population of size 20 using without replacement technique? It is not possible. So, that is why it is trying to give you an error over here. So, the interpretation goes very simple that if the sample size of data is greater than the population size in SRSWOR, then this type of error will occur.

But now my next question is to you is what will happen if instead of using SRSWOR, I use SRSWR - Simple Random Sampling With Replacement. What do you think? Is it possible to draw a sample of size 21 from a population of size 20? Please pause the video, try to think about 1 minute, and then start the video again. My answer is this. Yes, because in the case of SRSWR, you are always trying to replace your units back and you can reuse them.

(Refer Slide Time: 39:50)

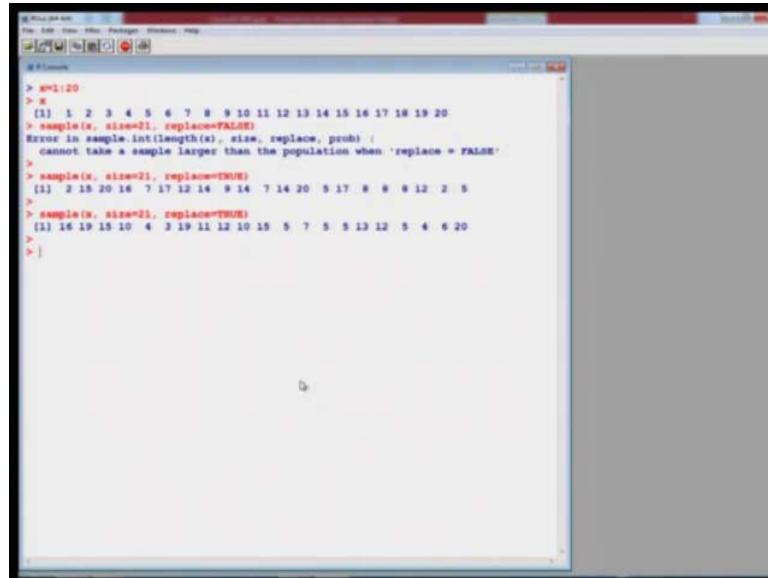


The image shows a slide titled "Using R software: How to draw simple random sample" with a subtitle "(2) Special comments: SRSWR". It displays R console output for the command `sample(x, size=21, replace=TRUE)`, resulting in a vector of 21 numbers: `[1] 14 6 8 15 2 20 6 8 8 7 12 4 12 3 7 7 4 20 8 16 7`. Below the output, an interpretation states: "In same example, the sample is drawn with SRSWR means replace is TRUE, so no error is occurred." At the bottom, a smaller screenshot of the R console shows the full context: `> x` followed by `[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20`, then `> sample(x, size=21, replace=TRUE)` followed by the same 21-number vector, and a prompt `> |`.

For example, if you try to see here in the same example, if I try to say here say sample x size equal to 21, but replace is now equal to TRUE. So, we are getting an getting a

sample over here that is possible, because there will always be population of size 20 that will always be available for sampling. So, the interpretation goes very simple that the sample is drawn with SRSWR. So, there is no error which is occurring in this case.

(Refer Slide Time: 40:31)



```
> n=20
> x
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
> sample(x, size=21, replace=FALSE)
Error in sample.int(length(x), size, replace, prob) :
  cannot take a sample larger than the population when 'replace = FALSE'
>
> sample(x, size=21, replace=TRUE)
[1] 2 15 20 16 7 17 12 14 9 14 7 14 20 5 17 8 8 8 12 2 5
>
> sample(x, size=21, replace=TRUE)
[1] 16 19 15 10 4 3 19 11 12 10 15 5 7 5 5 13 12 5 4 6 20
>
> }
```

So, I will try to show you this thing on the R console also, so that you can believe on me. So, I try to I need to define here x once again. So, they can see here x is 1 to 20. Now, if I try to take here x sample size is equal to 21, it is giving me an error. But if I make it here TRUE, then it is giving me a sample. And if you try to repeat this command, you will get a different sample right ok.

So, now I would like to stop here. And I have given you a fair idea that how you can draw the simple random sample with and without replacement using the R software, but that was the answer to the first question which I raised in the beginning. The answer to the second question I will take up in the next lecture. But before leaving it is very important for me to once again remind you that now this is your turn.

You please take your computers, and then just try to take hypothetical data set, try to draw different type of samples, try to vary the size of the sample and size of the population, and try to see whether your software or data works.

Well, every software, every computer will have a limitation to handle the big data, so that will raise to different types of question whether you have to modify your computer,



your computer architecture or you have to learn some programming technique, so that you can handle such a big data set.

For example, one possible option is parallel programming. Well, I am not aware that whether parallel programming is possible in R or not, but definitely I am not saying that R is the only way out. But you can always call R in different types of program, those possibilities exist and that is what you have to explore. So, you try to think, try to practice, try to take some examples, and try to make yourself more confident. And I will see you in the next lecture.

Till then good bye.