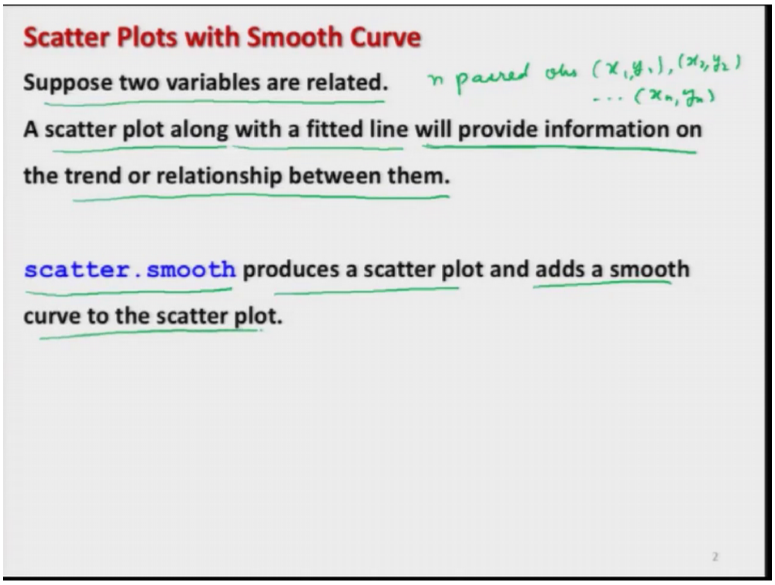
Descriptive Statistics with R Software
Prof. Shalabh
Department of Mathematics and Statistics
Indian Institute of Technology, Kanpur

Lecture – 26
Association of Variables – Smooth Scatter Plots

Welcome to the lecture on the course Descriptive Statistic with R Software. You may recall that in the earlier lecture, we started a discussion on the association of variables. And, we had considered how to construct the bivariate plots using the command plot and we had used different types of option, to create a more interactive plots. Now, when we are trying to use the plot option to create the scatter plot, then we would like to have two types of information; one is the direction or the trend and second is the magnitude and the magnitude was decided on the basis whether the strength of the relationship is strong or say moderate or weak and so on.

So now, the question is this by looking at the scatter diagram how you would know that whether the strength is more or less. So, in order to do so we had created a line manually, but now that can be done on the basis of software also. So now, we are going to consider the plots where we will create the scatter plot and we will also add a smooth line and that line will give us a sort of fit; and by comparing the observation with that fit we can compare whether the strength is less or more or this is weak or strong and so on. So, in this lecture we are going to consider the scatter smooth plots.

(Refer Slide Time: 01:56)



Scatter Plots with Smooth Curve

Suppose two variables are related. n paired obs $(x_1,y_1), (x_2,y_2) \ldots (x_n,y_n)$

A scatter plot along with a fitted line will provide information on the trend or relationship between them.

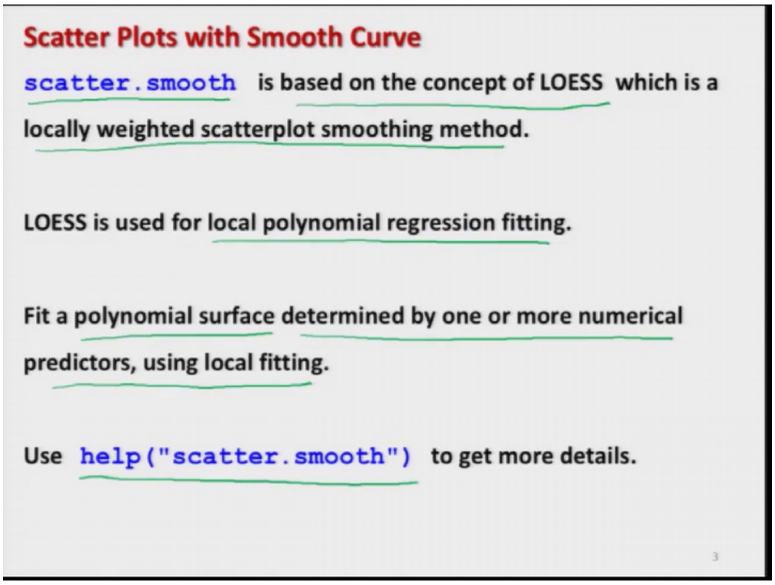
`scatter.smooth` produces a scatter plot and adds a smooth curve to the scatter plot.

2

So, now we assume that there are two variables and those variables are related and we have obtained and paired observations say x 1 y 1 x 2 y 2 up to here x n y n and so on. Now, the objective is that we want to create a scatter plot and inside the scatter plot we want to have a line which is called as fitted line why this is called a fitted line that will be clear to you after some lectures. And when I try to do so, this type of graphics will provide us the information on the trend or relationship between them.

And in order to construct such a graphic in R software we have a command scatter dot smooth s c a double t e r dot s m double o t h. And this command produces a scatter plot and it also adds a smooth curve to the scatter plot.

(Refer Slide Time: 03:09)



Scatter Plots with Smooth Curve

`scatter.smooth` is based on the concept of LOESS which is a locally weighted scatterplot smoothing method.

LOESS is used for local polynomial regression fitting.

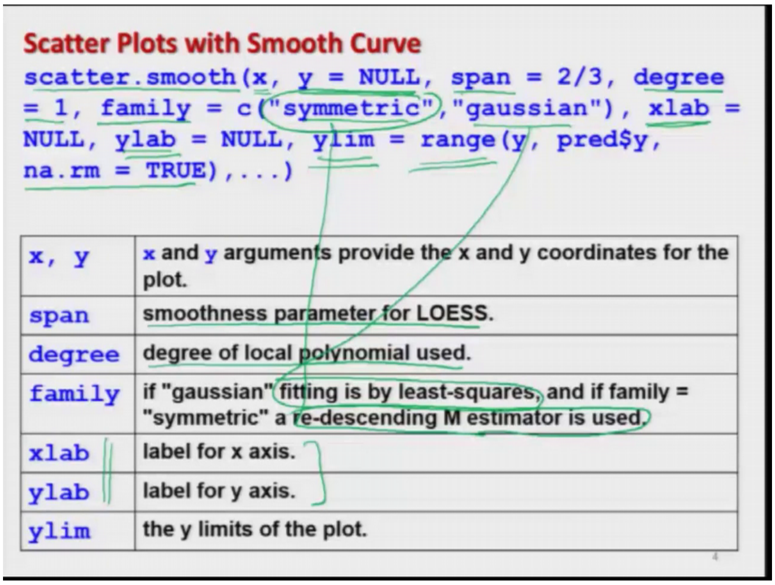
Fit a polynomial surface determined by one or more numerical predictors, using local fitting.

Use `help("scatter.smooth")` to get more details.

So, now how to do it and what are the details? So, this function actually this command is scatter dot smooth, this is based on the concept of loess l o e double s actually is locally weighted scatter plot smoothing method. And this is used for local polynomial regression fitting and in this case it fits a polynomial surface which is determined by one of the one or more numerical predictors using the local fitting.

Definitely I am not going to discuss about the details about the lowest and so on, but we are simply going to use it. And I thought that because you will see that later on there are some options the details are written in terms of l o e double s loess. So, I thought that I should tell you. So, that you do not get confused at a later stage well; if you want to have more details about this is scatter dot smooth function please say look into the help of this command and you will get more details right.

(Refer Slide Time: 04:11)



Scatter Plots with Smooth Curve

```
scatter.smooth(x, y = NULL, span = 2/3, degree
= 1, family = c("symmetric","gaussian"), xlab =
NULL, ylab = NULL, ylim = range(y, pred$y,
na.rm = TRUE),...)
```

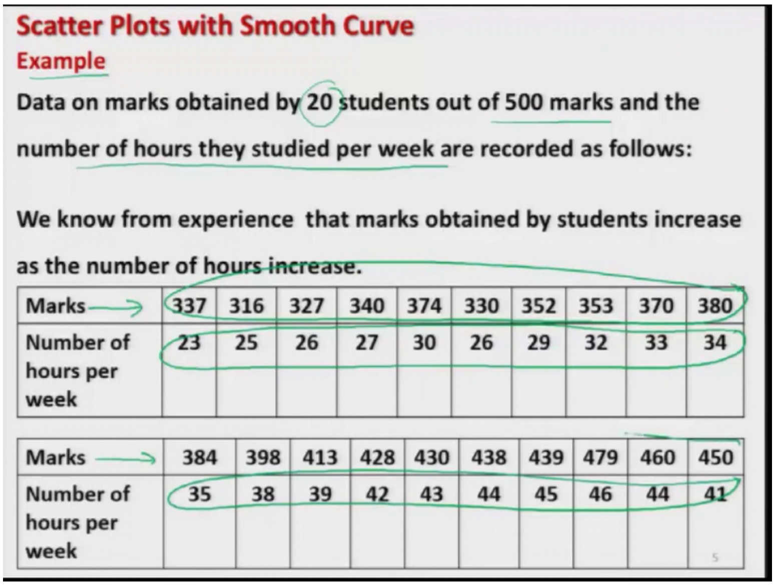
x, y	x and y arguments provide the x and y coordinates for the plot.
span	smoothness parameter for LOESS.
degree	degree of local polynomial used.
family	if "gaussian" fitting is by least-squares, and if family = "symmetric" a re-descending M estimator is used.
xlab	label for x axis.
ylab	label for y axis.
ylim	the y limits of the plot.

So, now the more detailed command of a scatter dot smooth which will give you a scatter plot and a smooth curve is the following; you can see here the command is the same scattered dot smooth. Now I am trying to give here the data here x and here y, y is here actually null because we are trying to make it only with the one variable, but if you want to make it bivariate plot you can use both x and y data. And then there are different option here as say span, span controls the smoothness for this loess and this then there is a degree here is given as here one.

This degree is the degree of the local polynomial which is used for fitting and then it is asking for family this family can be symmetric or Gaussian and well there are different methods for fitting the data. So, in case if we are using the Gaussian, Gaussian is indicating that the fitting has been done on the basis of least square method, he will consider the least square method at a later stage. And, in case if the option of symmetric is used then this is indicating that co sending m estimator is being used to get the polynomial.

And, similarly if you want to give the labels on x axis or say y axis then these are the command x lab and y lab here. And, similarly if you want to give the limits for example, if you want to provide the limits, on the y axis then this is given by y lim and then yeah means this is the range. If you remember range gives you the two values minimum and maximum that, we had discuss and after that if you are handling with the missing data then you have to use the command na dot r m is equal true or false. So, there are some more option available, but I would request you that you please try to look in to the help menu and try to understand how to use them.

(Refer Slide Time: 06:13)



Scatter Plots with Smooth Curve

Example

Data on marks obtained by 20 students out of 500 marks and the number of hours they studied per week are recorded as follows:

We know from experience that marks obtained by students increase as the number of hours increase.

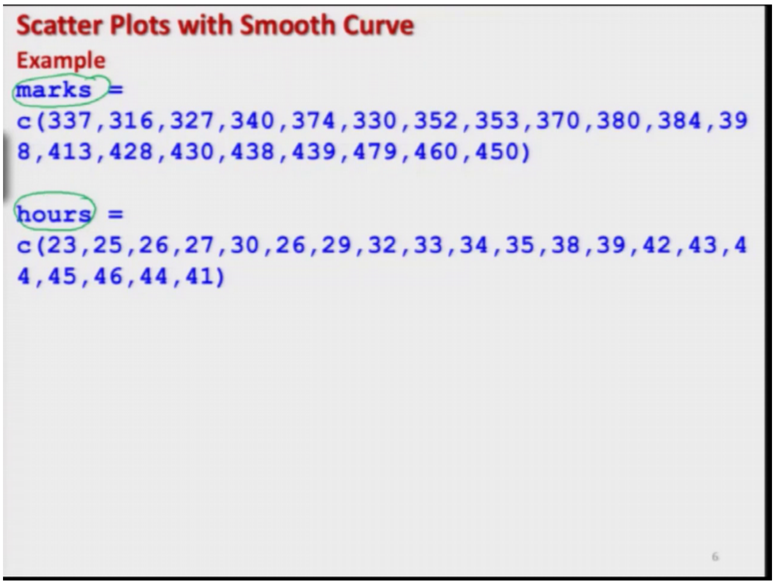
Marks →	337	316	327	340	374	330	352	353	370	380
Number of hours per week	23	25	26	27	30	26	29	32	33	34

Marks →	384	398	413	428	430	438	439	479	460	450
Number of hours per week	35	38	39	42	43	44	45	46	44	41

Now, I will try to take an example and would show you that how to plot such graphics. So, I am going to consider the same example which I discussed in the last lecture, where we have obtained the data on the marks obtained by the students out of 500. And, the number of hours they studied in a week; and this data was obtained for 20 students and this data here is given like this, the first way here in this case is the marks of the students out of 500 similarly here also.

So, these are the marks student have obtained and the second row is giving the information on number of hours they have studied in a week like this here ok. And this data I already have stored, in two variables when I am calling it here marks and hours exactly in the same way as I did in the last lecture.

(Refer Slide Time: 07:01)



Scatter Plots with Smooth Curve

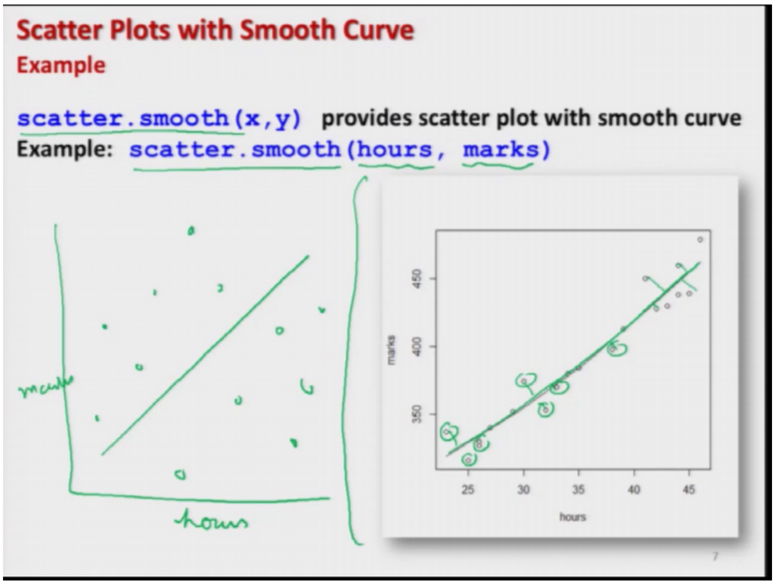
Example

```
marks =
c(337,316,327,340,374,330,352,353,370,380,384,39
8,413,428,430,438,439,479,460,450)

hours =
c(23,25,26,27,30,26,29,32,33,34,35,38,39,42,43,4
4,45,46,44,41)
```

Now, after this I would like to create a scatter plot with a smooth curve using this data.

(Refer Slide Time: 07:14)



Scatter Plots with Smooth Curve

Example

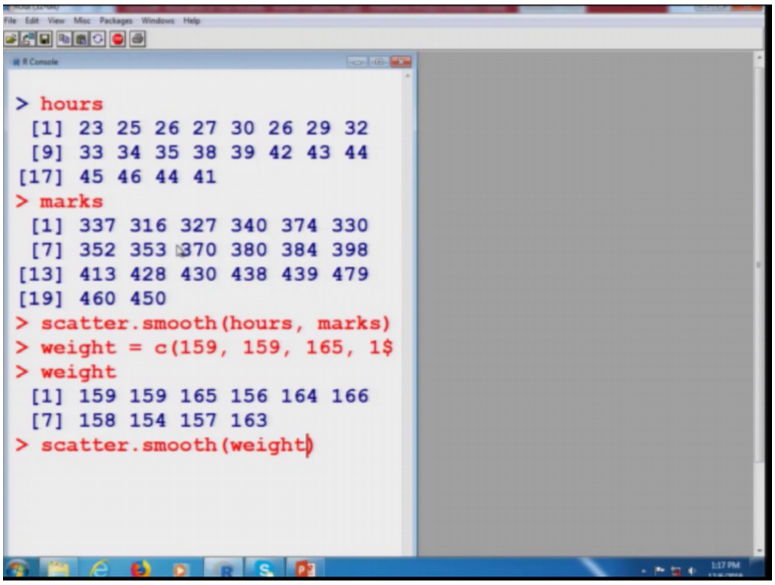
`scatter.smooth(x,y)` provides scatter plot with smooth curve

Example: `scatter.smooth(hours, marks)`

And for that I use the command scatter dot smooth and inside the argument then I try to give the name of the variable say hours and here marks; and if you try to see here this is the graphic that we are going to obtain right. You can see here now these points are the same point which you occurring only in the scatter plot that we had constructed in the last lecture.

But now there is a line which is added to this thing and this line is helping us in knowing that, how much is the deviation of this individual observation from these points. And, if these deviations are less or if these points are lying very close to the line I can say that the strength is quite high. Suppose, if you had got the same data with this line and, but the points are lying here and there and so on right yeah, this may happen then in this case I would say that the strength is weak or the degree of the linear relationship is weak in this case.

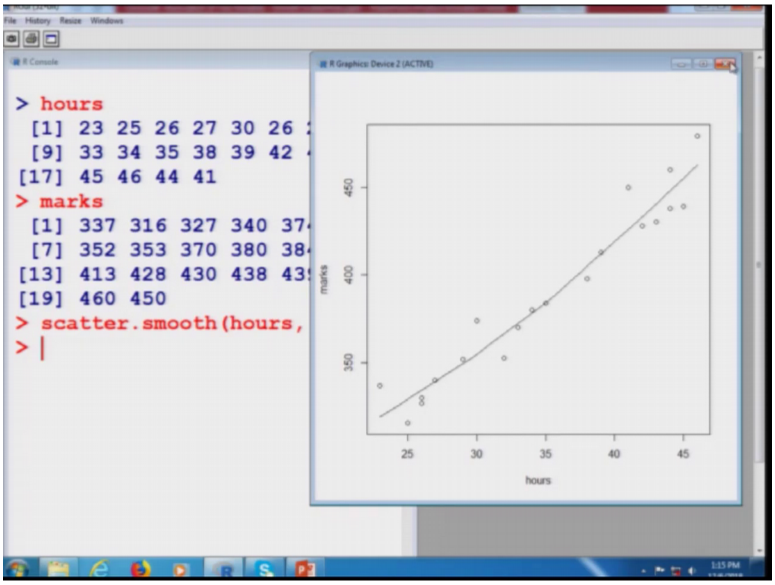
(Refer Slide Time: 08:35)



```
> hours
 [1] 23 25 26 27 30 26 29 32
 [9] 33 34 35 38 39 42 43 44
[17] 45 46 44 41
> marks
 [1] 337 316 327 340 374 330
 [7] 352 353 370 380 384 398
[13] 413 428 430 438 439 479
[19] 460 450
> scatter.smooth(hours, marks)
> weight = c(159, 159, 165, 1$
> weight
 [1] 159 159 165 156 164 166
 [7] 158 154 157 163
> scatter.smooth(weight)
```

So, this is how we try to do it and now I will try to show you on the R console also that how to operate it. So, I already have entered the data on say hours and here marks you can see here.

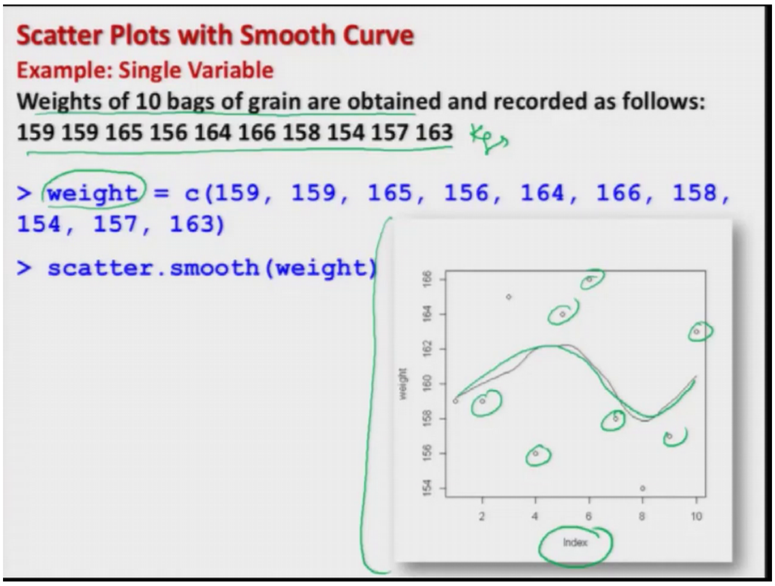
(Refer Slide Time: 08:43)



```
> hours
 [1] 23 25 26 27 30 26 
 [9] 33 34 35 38 39 42 
[17] 45 46 44 41
> marks
 [1] 337 316 327 340 37
 [7] 352 353 370 380 38
[13] 413 428 430 438 43
[19] 460 450
> scatter.smooth(hours,
> |
```

And I try to make it here a plot scatter a smooth by this command and you can see here we are getting the same curve what we have shown here right ok. So, now let us come back towards slide.

(Refer Slide Time: 09:03)



Scatter Plots with Smooth Curve

Example: Single Variable

Weights of 10 bags of grain are obtained and recorded as follows:

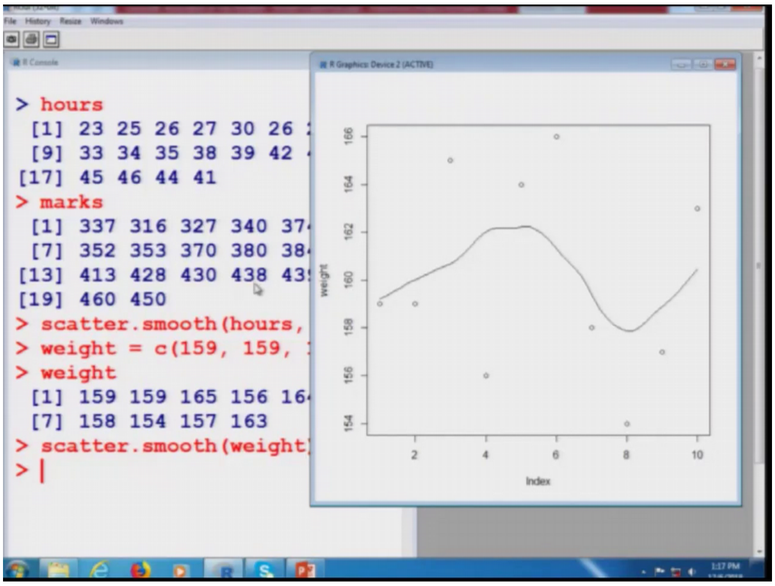
159 159 165 156 164 166 158 154 157 163 Kgs

```
> weight = c(159, 159, 165, 156, 164, 166, 158,
154, 157, 163)
> scatter.smooth(weight)
```

And now if you try to see I am taking a very small data set here. And, I am just trying to consider 10 values and these 10 values are indicating the weight of 10 bags of grains. And this data which is recorded here for 10 bags in say kilograms, this is stored here inside the variable weight right. And I am trying to plot the scatter smooth graph for this data set and this is plotted here and the graph comes out to be like this.

So, you can see here these are the points and this is indicating that possibly the relationship is not actually linear, but it is a sort of non-linear relationship. And please keep this figure in mind because I will try to give you some more information look at here on because in the same data set ok. And, if you want to plot it here on the R console, I can copy this data here and right this comes out to be like this and if you try to see what I am trying to do here.

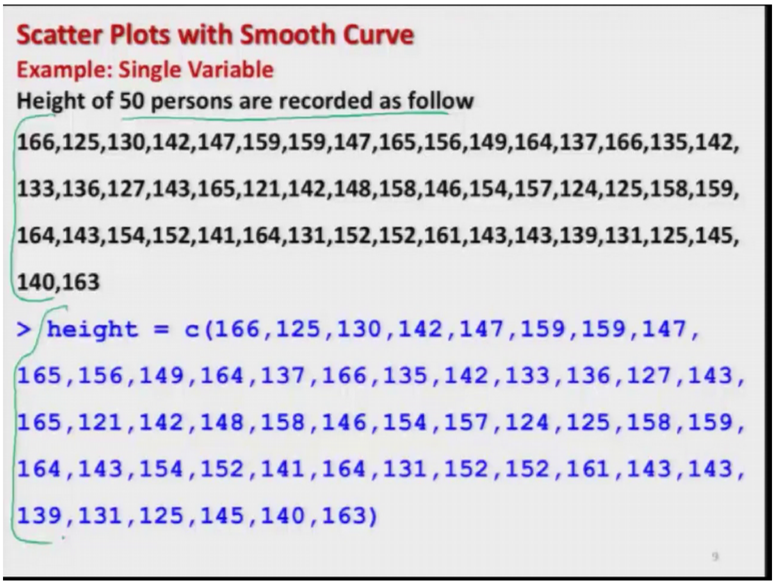
(Refer Slide Time: 10:28)



```
> hours
 [1] 23 25 26 27 30 26
 [9] 33 34 35 38 39 42
[17] 45 46 44 41
> marks
 [1] 337 316 327 340 37
 [7] 352 353 370 380 38
[13] 413 428 430 438 43
[19] 460 450
> scatter.smooth(hours,
> weight = c(159, 159,
> weight
 [1] 159 159 165 156 16
 [7] 158 154 157 163
> scatter.smooth(weight
> |
```

I am simply trying to make a scatter smooth plot of this thing and this comes out to be like this; one thing what you have to notice here that in this example I have used only here one variable. So, you can see in this graphic that the value on the x axis this is only an index. So, what we are trying to learn here that is scattered smooth command can also be used for getting the curve only for univariate data when we have only one variable right.

(Refer Slide Time: 11:10)



Scatter Plots with Smooth Curve

Example: Single Variable

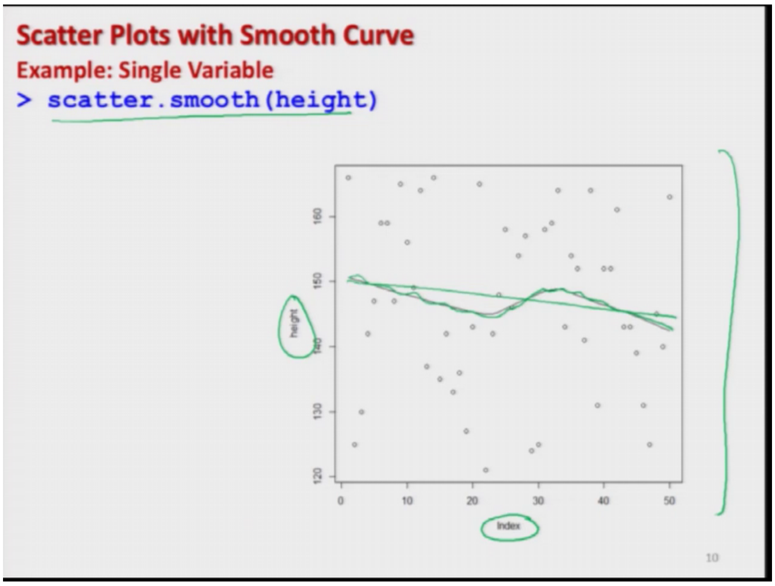
Height of 50 persons are recorded as follow

166,125,130,142,147,159,159,147,165,156,149,164,137,166,135,142,
133,136,127,143,165,121,142,148,158,146,154,157,124,125,158,159,
164,143,154,152,141,164,131,152,152,161,143,143,139,131,125,145,
140,163

```
> height = c(166,125,130,142,147,159,159,147,
165,156,149,164,137,166,135,142,133,136,127,143,
165,121,142,148,158,146,154,157,124,125,158,159,
164,143,154,152,141,164,131,152,152,161,143,143,
139,131,125,145,140,163)
```

And now I will take an example where there are some more number of data set and I will try to show you how this curve look in the univariate case single variable. So, I have collected the heights of 50 percents like this, it is the same example that we have considered earlier and this data has been stored inside a variable here height like this; and based on that I will try to create a scatter a smooth plot of this data here in height.

(Refer Slide Time: 11:36)



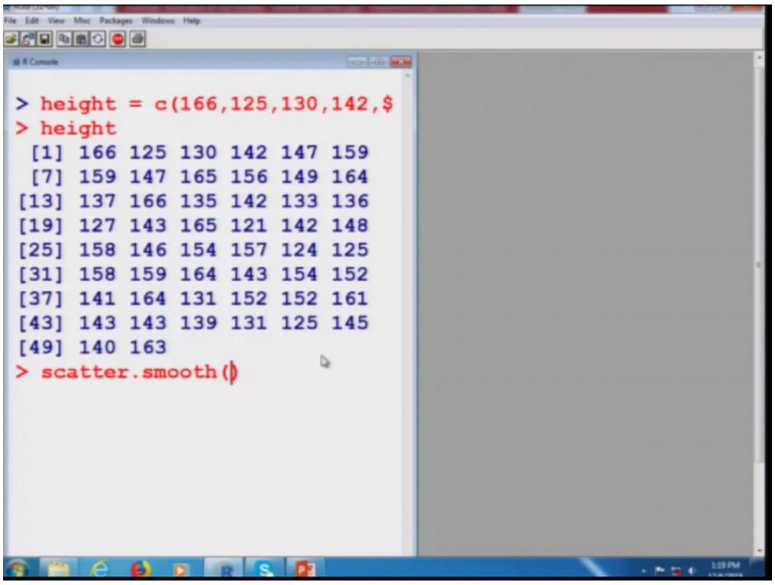
Scatter Plots with Smooth Curve

Example: Single Variable

```
> scatter.smooth(height)
```

So, you will see here this looks like this and it is indicating well, this can approximately be a sort of linear relationship, but it is indicating that this relationship here is can be linear. On the x axis this is only the index of the observation and on the y axis these are the values of heights. The reason why I am taking here this graphic is that that you will get on I want to create another graphic on the same data set and I would like to compare both of them together, but if you want to see the same graphic on the R console.

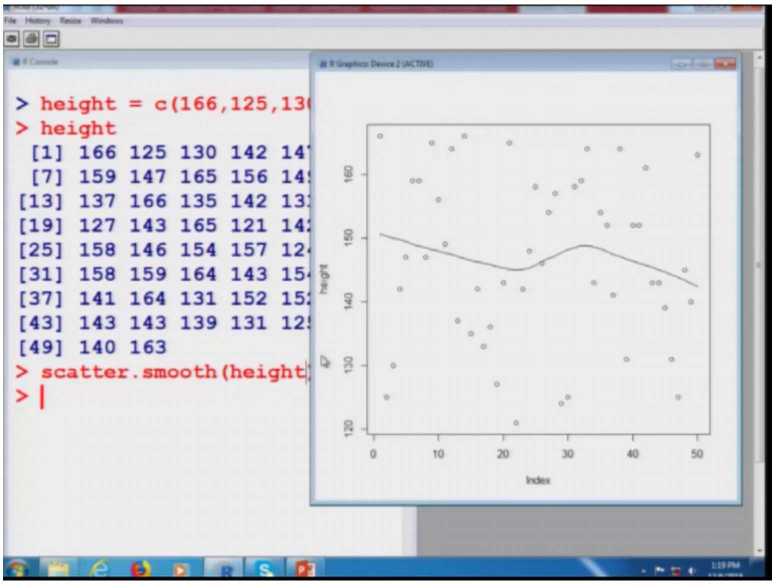
(Refer Slide Time: 12:24)



```
> height = c(166,125,130,142,$
> height
 [1] 166 125 130 142 147 159
 [7] 159 147 165 156 149 164
[13] 137 166 135 142 133 136
[19] 127 143 165 121 142 148
[25] 158 146 154 157 124 125
[31] 158 159 164 143 154 152
[37] 141 164 131 152 152 161
[43] 143 143 139 131 125 145
[49] 140 163
> scatter.smooth()
```

You can see here that I am trying to store the data say here height this is the data vector height and I try to create here it is quite a smooth graph of height this will come like this.

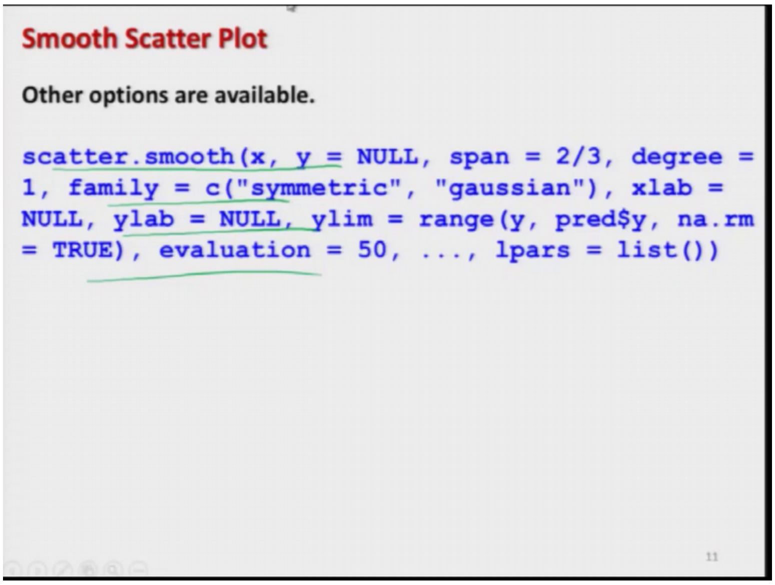
(Refer Slide Time: 12:35)



```
> height = c(166,125,13(
> height
 [1] 166 125 130 142 14
 [7] 159 147 165 156 14
[13] 137 166 135 142 13
[19] 127 143 165 121 14
[25] 158 146 154 157 12
[31] 158 159 164 143 15
[37] 141 164 131 152 15
[43] 143 143 139 131 12
[49] 140 163
> scatter.smooth(height
> |
```

So, this is the same one which we have shown here on this slide.

(Refer Slide Time: 12:41)



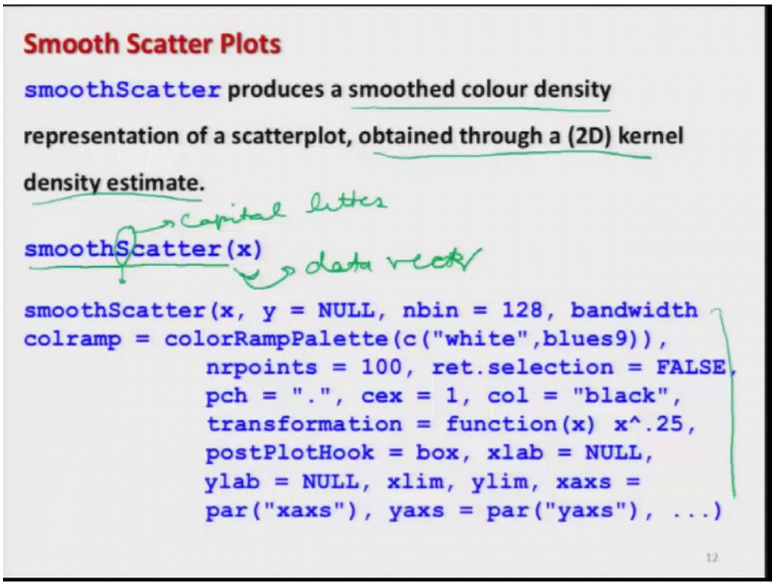
Smooth Scatter Plot

Other options are available.

```
scatter.smooth(x, y = NULL, span = 2/3, degree =
1, family = c("symmetric", "gaussian"), xlab =
NULL, ylab = NULL, ylim = range(y, pred$y, na.rm
= TRUE), evaluation = 50, ..., lpars = list())
```

Now, there are some more options which are available on this scatter a smooth curve, which will give you more interactive things and that we already have discussed here. So, you can just look into the help and try to experiment with various options.

(Refer Slide Time: 12:59)



Smooth Scatter Plots

smoothScatter produces a smoothed colour density representation of a scatterplot, obtained through a (2D) kernel density estimate.

```
smoothScatter(x)
```

Capital letter

data vector

```
smoothScatter(x, y = NULL, nbin = 128, bandwidth
colramp = colorRampPalette(c("white",blues9)),
          nrpoints = 100, ret.selection = FALSE,
          pch = ".", cex = 1, col = "black",
          transformation = function(x) x^.25,
          postPlotHook = box, xlab = NULL,
          ylab = NULL, xlim, ylim, xaxs =
          par("xaxs"), yaxs = par("yaxs"), ...)
```

Now, I would like to discuss another type of smooth scatter plot. There is another command in R software, which produces a smooth a scatter plot, but this plot is little bit different than what we have obtained earlier this is essentially a smooth and colored density plot. And, this is obtained through a two dimensional kernel density estimate;

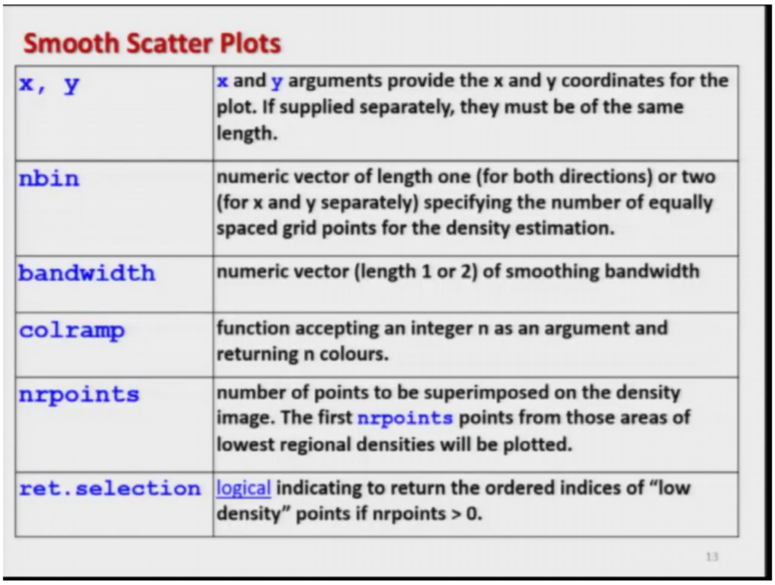
you may recall that when we were discussing the graphics in the univariate case then we had created the frequency density curve or say density plots using the kernel estimates. So, there we have defined the kernel functions those kernel functions were having some nice statistical properties simile to the properties having in the probability density function. So, that kernel function was for one variable. So it so that was a univariate kernel function.

Similarly, when we have more than one variables, then in statistics it is possible to handle them through the probability density functions, which are functions for more than one variable. Suppose you have two variables x and y or three variables xy and z then it is possible to define the joint probability density function of x and y or joint probability density function of x y and z. And similarly the kernel functions can also be defined in a multivariate setup. So, in this case this plot is going to use the concept of kernel density estimates in two variates and that is why this is called a two dimensional kernel density estimate? So, well we are not going into the detail that what are those kernel density estimates in two dimension, but definitely you should know that what are we going to do and how the outcome has been obtained?

And definitely if you want to know the more details, we already have understood that one of the biggest advantage of R software is that it is not a black box you can go to the site of R software. And there you will see the help menu and there will be all the details that how this scatter plot has been constructed ok. So, let us Now come back to our slides and in order to create this type of a smooth scatter plot, the command is smooth scatter, but you try to see the difference, here one letter capital S of a scatter is in capital letter that you have to keep in mind capital letter; and the spelling is a smooth scatter s m double o t h any small letters then S in capital letters and then a double t e r in small letters and inside the arguments you have to give the data vector right.

And similarly if you want to have more options, you can see here they are given here, but definitely I am not going to discuss them.

(Refer Slide Time: 16:09)

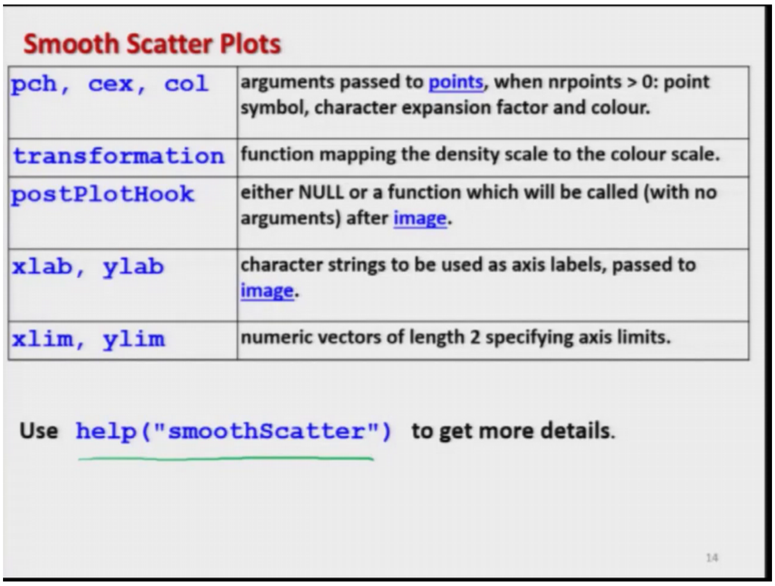


Smooth Scatter Plots

`x, y`	x and y arguments provide the x and y coordinates for the plot. If supplied separately, they must be of the same length.
`nbin`	numeric vector of length one (for both directions) or two (for x and y separately) specifying the number of equally spaced grid points for the density estimation.
`bandwidth`	numeric vector (length 1 or 2) of smoothing bandwidth
`colramp`	function accepting an integer n as an argument and returning n colours.
`nrpoints`	number of points to be superimposed on the density image. The first `nrpoints` points from those areas of lowest regional densities will be plotted.
`ret.selection`	logical indicating to return the ordered indices of "low density" points if nrpoints > 0.

But I have given them on the slides those lights are with you and, you can have a look and if you try to use them, you will get a more informative and better graphics.

(Refer Slide Time: 16:18)



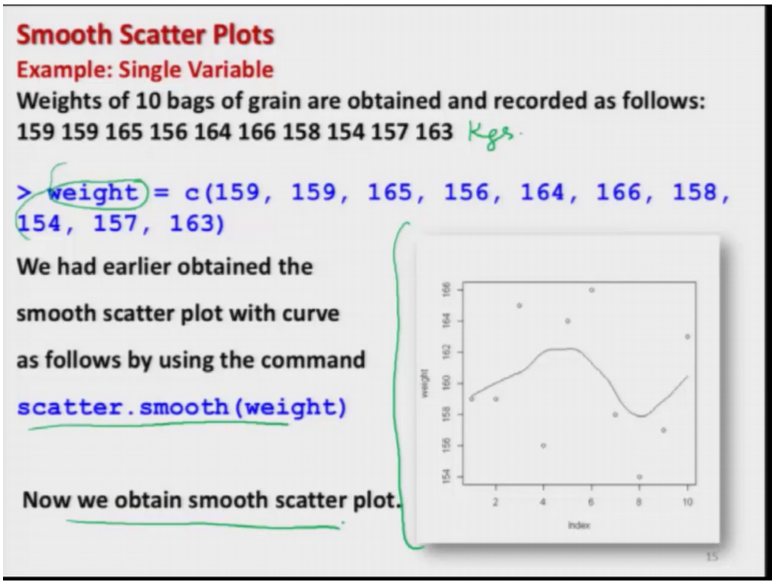
Smooth Scatter Plots

`pch, cex, col`	arguments passed to points, when nrpoints > 0: point symbol, character expansion factor and colour.
`transformation`	function mapping the density scale to the colour scale.
`postPlotHook`	either NULL or a function which will be called (with no arguments) after image.
`xlab, ylab`	character strings to be used as axis labels, passed to image.
`xlim, ylim`	numeric vectors of length 2 specifying axis limits.

Use `help("smoothScatter")` to get more details.

So, just try to have a look on the help on this smooth scatter and you will get all these information right.

(Refer Slide Time: 16:31)



Smooth Scatter Plots

Example: Single Variable

Weights of 10 bags of grain are obtained and recorded as follows:
159 159 165 156 164 166 158 154 157 163 Kgs.

```
> weight = c(159, 159, 165, 156, 164, 166, 158,
154, 157, 163)
```

We had earlier obtained the smooth scatter plot with curve as follows by using the command

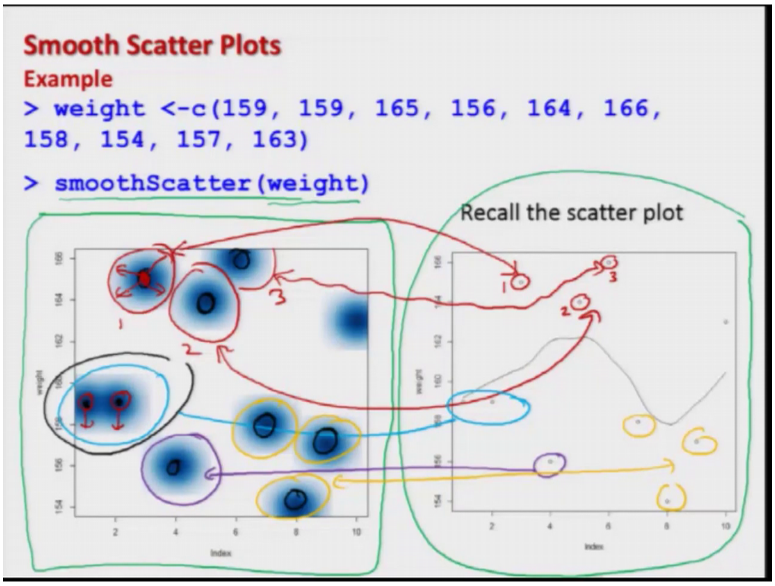
```
scatter.smooth(weight)
```

Now we obtain smooth scatter plot.

So, Now I would try to first take the same example which I just took in the case of univariate data; and I would try to take the data on say here weight where we had collected the weight of 10 bags of grains in kilograms right, and this data has been given inside the data vector here weight. So, earlier we had obtained the smooth scatter plot using the command scatter dot is smooth and this curve was looking like this.

Now, I will use this new command a smooth scatter on the same data set and we will obtain the new graphic and I will try to compare it with that too.

(Refer Slide Time: 17:14)



Smooth Scatter Plots

Example

```
> weight <-c(159, 159, 165, 156, 164, 166,
158, 154, 157, 163)
> smoothScatter(weight)
```

Recall the scatter plot

So, you can see here when I am trying to use the command a smooth scatter for the data on weight here I get here a graphic like this one. What are these points and what is this showing? Now if you try to recall, the earlier plot that was obtained was like here this, now if you try to put both these points side by side you can see here the similarity. You can see here this is here a point this is here a point this is here the point on the earlier plot let me call this points a point number 1, 2 and 3 and they were created as a dots.

Now, the same points have been created with this new command is smooth scatter which are here like this point number 1, point number 2 and point number 3. So, this point is here, this point is here and this point is here right. And, similarly if you try to see here now I will use a different color, so that you can observe the movement of my pen these two points they are here. And similarly if you try to hear this point this is here and similar to this if I try to see here this point 1, 2 and 3 here these are the points here.

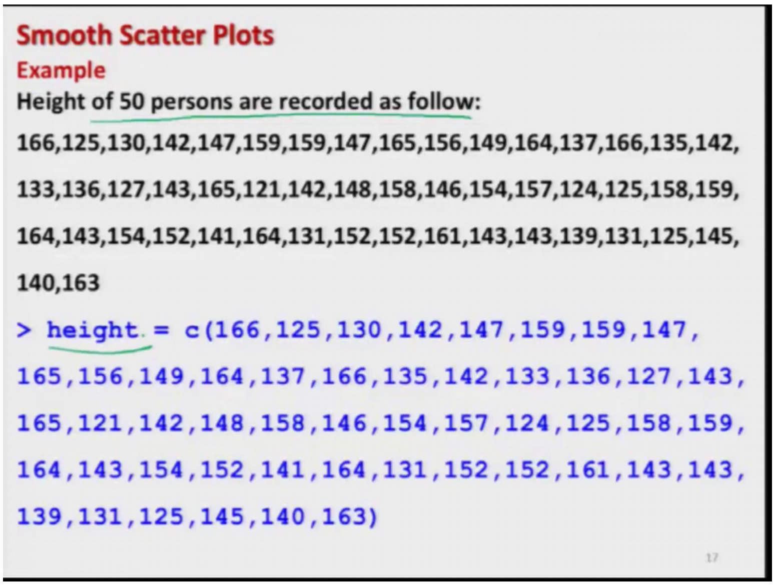
So, they are here. So, you can see here both these plots are going to give us the similar information, but their structures are different. And now it depends on the experimenter or the statistician or on new people, you have to decide that under the given circumstances which graph is going to give you much better picture. One situation where this smooth scatter type of plot will be more useful is that, that when you are trying to obtain the observation and you are not 100 percent confident about the values whether this value is 20 or 20.01 or say 19.08 then in that case this type of graphics that we have obtained.

Now using the smooth scatter command they will be more useful; because they will also try to show you the uncertainty involved in the point. But definitely, when I am trying to say the value is 20 then definitely the margin of error should be as small as possible and definitely if the value is 20 then there is no error and. This part is indicated that the values are 20 or 19.08 or 19.05 or 19 or the value 20 is 20.02, 20.5, 21 or 22 this is indicated by the decreasing tint of the color. Now if you try to see in this a graphic you can see here, suppose if you try to observe in this black color one you can see here the values in the center here are most dark.

Similarly if you try to take it another say anything any point over here, the middle part has the darkest color. And as we are moving from the center like this here if you try to see my pen infrared color, we are moving towards outside or in this point number 1 this is the center part and when we are trying to move from the center you can see that the color is now decreasing and the color is becoming lighter. So, this is what is indicated by this type of curve or this type of graphic, as the color is becoming lighter, that is showing the level of uncertainty. If you are confident, if you are is strong that your data value is correct that is indicated by a stronger color. But if you are weak and you are not confident about the data, then that variation or that uncertainty is indicated by lower tint of the same color or similar colors.

So, this is how you have to take a decision that which of the graphic you want you would like to use.

(Refer Slide Time: 22:21)



Smooth Scatter Plots

Example

Height of 50 persons are recorded as follow:

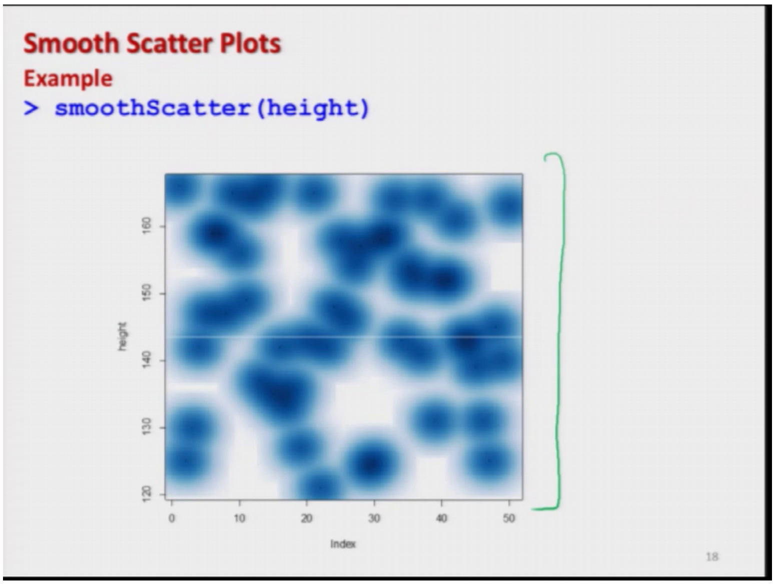
166,125,130,142,147,159,159,147,165,156,149,164,137,166,135,142,133,136,127,143,165,121,142,148,158,146,154,157,124,125,158,159,164,143,154,152,141,164,131,152,152,161,143,143,139,131,125,145,140,163

```
> height = c(166,125,130,142,147,159,159,147,
165,156,149,164,137,166,135,142,133,136,127,143,
165,121,142,148,158,146,154,157,124,125,158,159,
164,143,154,152,141,164,131,152,152,161,143,143,
139,131,125,145,140,163)
```

17

And now if I try to take the earlier example, where we have collected the height of 50 percents and this data was recorded inside of the vector height.

(Refer Slide Time: 22:33)



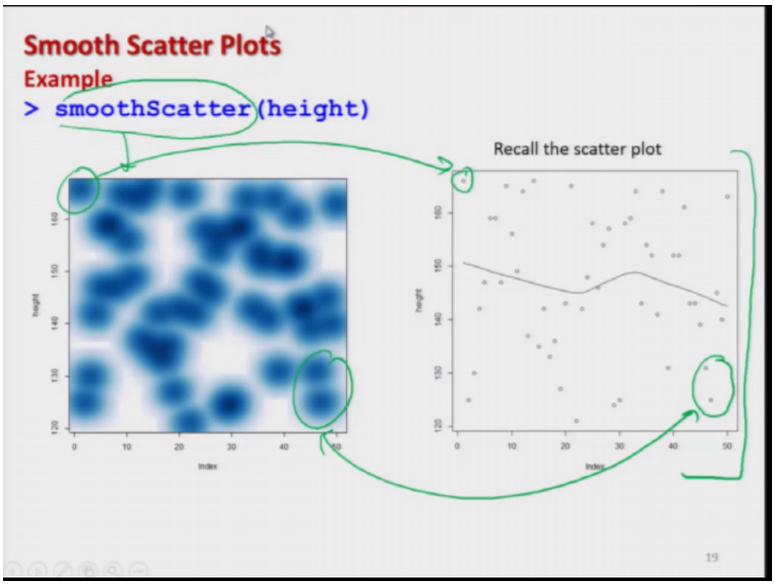
Smooth Scatter Plots

Example

```
> smoothScatter(height)
```

Then if I try to create the smooth a scatter plot it will look like this you can see this because the number of data points are quite large, so this is more concentrated right.

(Refer Slide Time: 22:45)



Smooth Scatter Plots

Example

```
> smoothScatter(height)
```

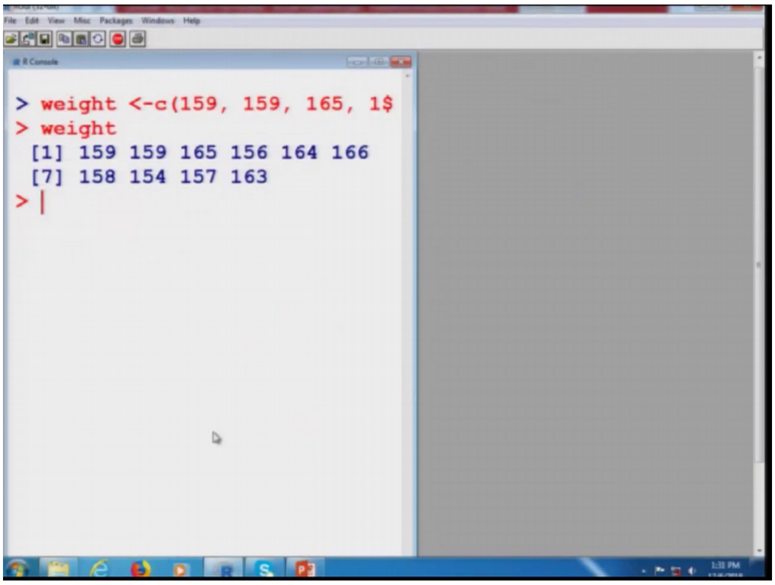
Recall the scatter plot

And, if you try to compare it with the with the earlier a scatter plot, you can see here earlier we had obtained this thing; and now excluding the new command is smooth a scatter we have obtained this plot.

So, you can see here this point and this point they are here the similar. And similarly here on the right hand side corner these two points and these two point they are the similar.

So, you can see here that both these graphics are trying to give different similar type of information, but in a different way. So, now, before I go further let me try to plot these things on the R console. So, first let me copy here the data here weight and you can see here.

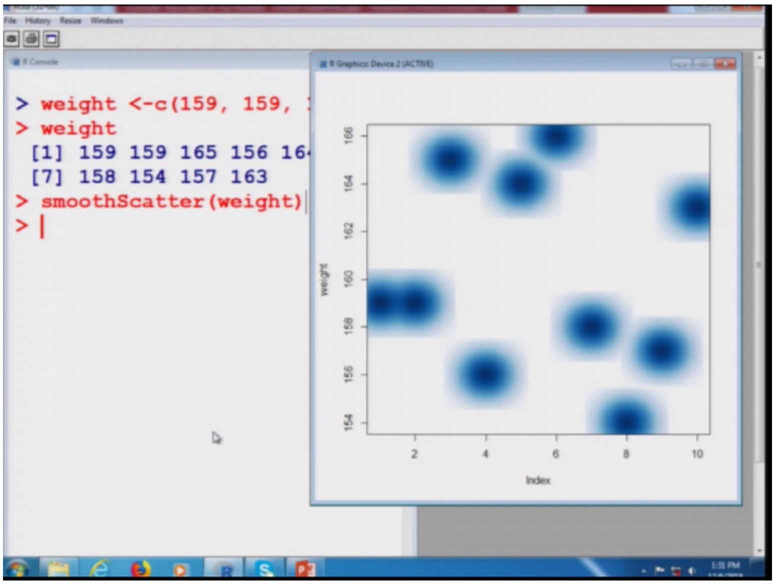
(Refer Slide Time: 23:27)



```
> weight <-c(159, 159, 165, 1$
> weight
 [1] 159 159 165 156 164 166
 [7] 158 154 157 163
> |
```

So, this data here is given as here weight and now I try to copy the same command is smooth scatter on the data set weight.

(Refer Slide Time: 23:40)

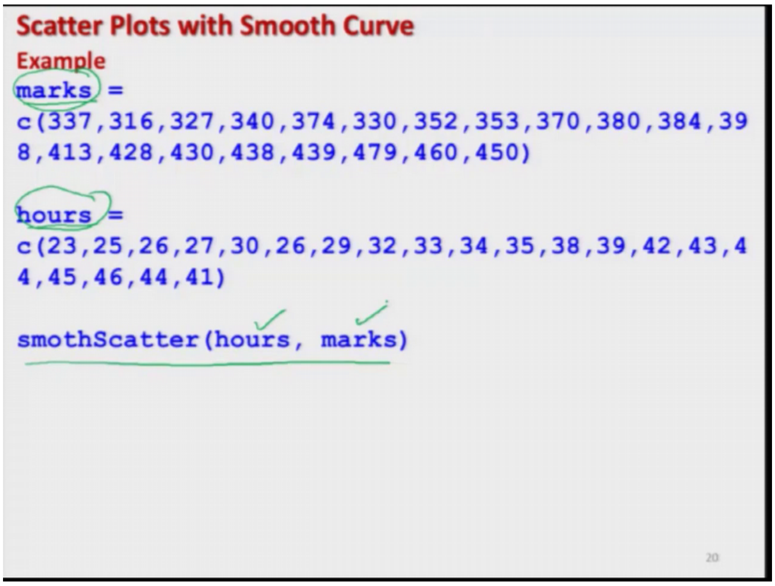


```
> weight <-c(159, 159,
> weight
 [1] 159 159 165 156 16
 [7] 158 154 157 163
> smoothScatter(weight)
> |
```

And you will see that, as soon as I executed it gives me this type of information. Now, up to now you could see I have taken two example, where I have considered the univariate data and I have created this plot.

Now, I would try to take a bivariate data and I would try to show you that how the information can be retrieved and how the information is present in this scattered a smooth curve?

(Refer Slide Time: 24:06)



Scatter Plots with Smooth Curve

Example

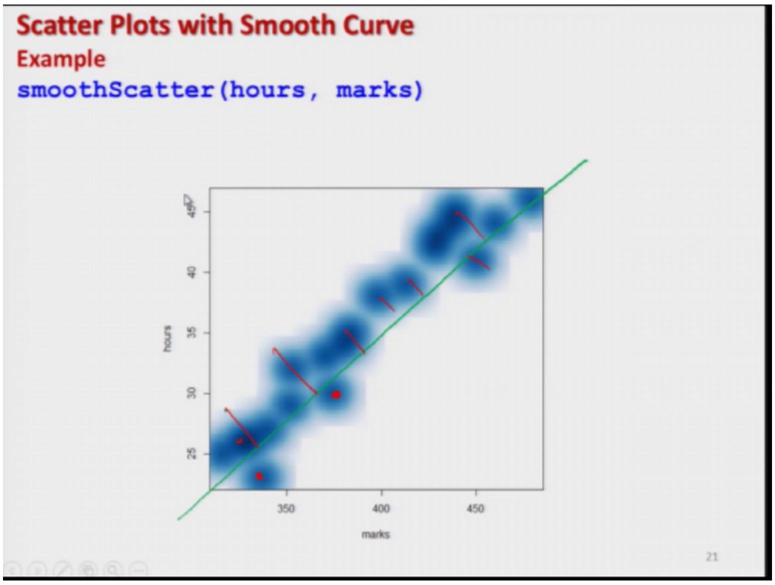
```
marks =
c(337,316,327,340,374,330,352,353,370,380,384,39
8,413,428,430,438,439,479,460,450)

hours =
c(23,25,26,27,30,26,29,32,33,34,35,38,39,42,43,4
4,45,46,44,41)

smothScatter(hours, marks)
```

So, you may recall that, we had consider an example where we have collected the data on the marks obtained by the students and the number of hours they studied every week. And this data was stored in the variables marks and hours and based on that, I would try to make the smooth scatter plot using the command is smooth scatter between hours and marks.

(Refer Slide Time: 24:29)



Scatter Plots with Smooth Curve

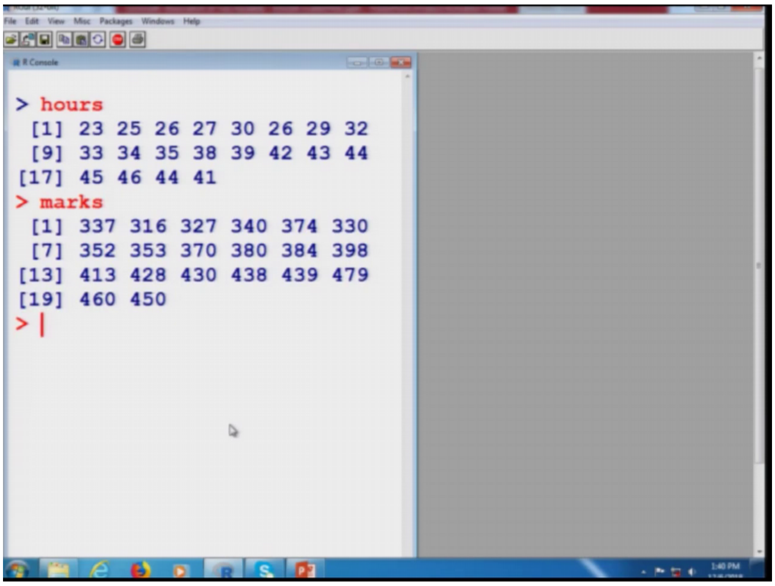
Example

smoothScatter(hours, marks)

And if you try to execute it you will see here you will get with this type of air curve. So, this is indicating that there is a sort of here linear trend and this is also giving you a sort of that what can be the maximum deviation in this data for example, like this.

You can simply have to visualize that what is the difference between the green line and the darkest part of the data darkest color of the data for them darkest colors is here in this center. So, this will give us information that how the things are happening. So, I would try to show you that how this is educated on the R console.

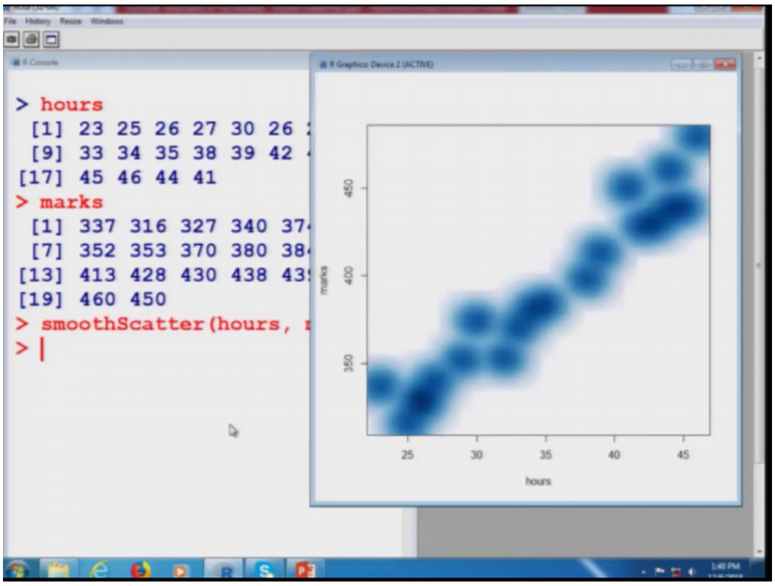
(Refer Slide Time: 25:08)



```
> hours
 [1] 23 25 26 27 30 26 29 32
 [9] 33 34 35 38 39 42 43 44
[17] 45 46 44 41
> marks
 [1] 337 316 327 340 374 330
 [7] 352 353 370 380 384 398
[13] 413 428 430 438 439 479
[19] 460 450
> |
```

So, you can see here I already have stored the data on hours and marks because we have just used it.

(Refer Slide Time: 25:16)



```
> hours
 [1] 23 25 26 27 30 26
 [9] 33 34 35 38 39 42
[17] 45 46 44 41
> marks
 [1] 337 316 327 340 37
 [7] 352 353 370 380 38
[13] 413 428 430 438 43
[19] 460 450
> smoothScatter(hours,
> |
```

And now I try to give here this smooth a scatter plot command and you get here a data like this one. So, now, in this lecture I would like to stop and you have seen that we have discussed that how to construct the smooth scatter plots. We have discussed two types of plots and in every type of graphic there are different commands, although I am not discussing it because they are exactly on the same lines as we have done several times in the past.

So, my request is that you please try to experiment with them. You can take even the same data set and try to see how you can add or change the labels on axis colors of these dots and how you can gave different types of titles and how you can incorporate more type of information by using different options available with the command? So, you practice it and we will see in the next lecture with some more graphics till then goodbye.