

Foundations of R Software

Lecture 52

Graphics

⋮

Bivariate and Three Dimensional Scatter Plots

Shalabh

Department of Mathematics and Statistics

Indian Institute of Technology Kanpur

Association of two variables:

Example

- Number of hours of study affect the marks obtained in an examination.
- Electricity/power consumption increases when the weather temperature increases.
- Weight of infants and small children increases as their height increases under normal circumstances.

Association of two variables:

The observations on both the variables are related to each other.

How to know the variables are related?

How to know the degree of relationship between the two variables?

Graphical procedures – Two dimensional plots, three dimensional plots etc.

Quantitative procedures – Correlation coefficients, contingency tables, Chi-square statistic, linear regression, nonlinear regression etc.

Association of two variables:

How to judge or graphically summarize the association of two variables?

X, Y : Two variables

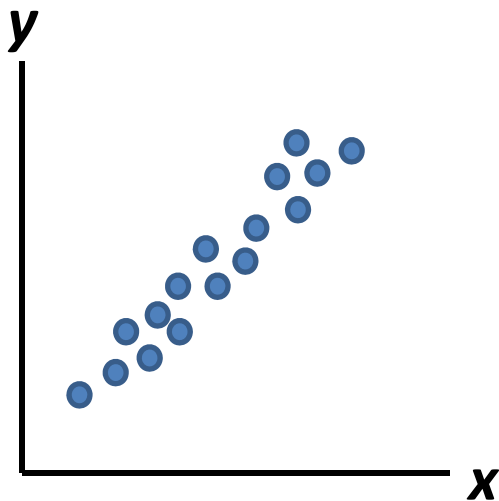
n pairs of observations are available as $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$

Scatter plots:

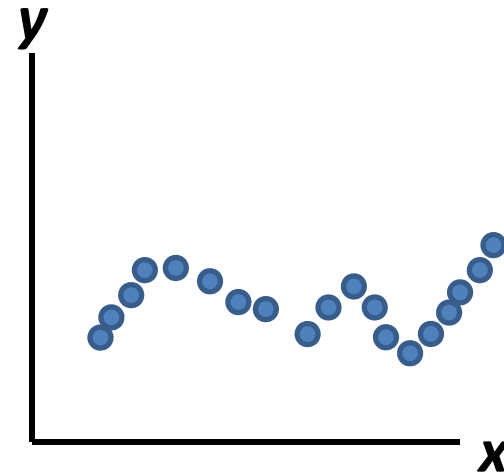
Plot the paired observations in a single graph, called as scatter plot.

Scatter plot reveals the nature and trend of possible relationship.

Relationships : Linear or nonlinear.



Linear relationship between X and Y



Nonlinear relationship between X and Y

Strength and trend of relationships:

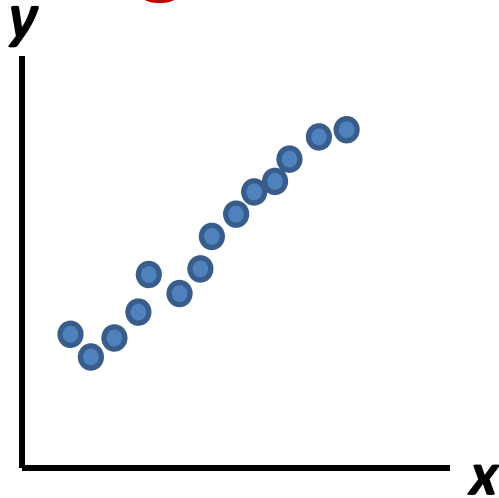


Fig. 1: Strong positive linear relationship

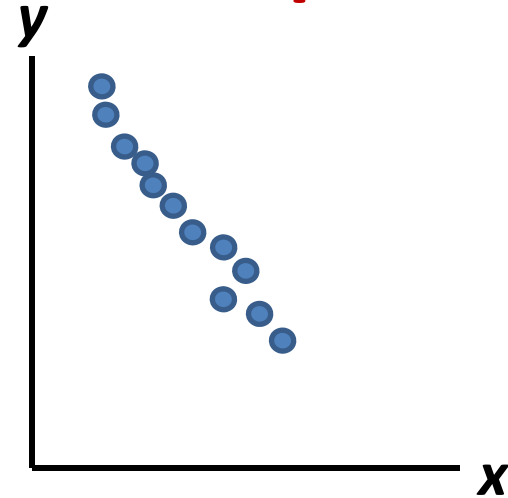


Fig. 2: Strong negative linear relationship

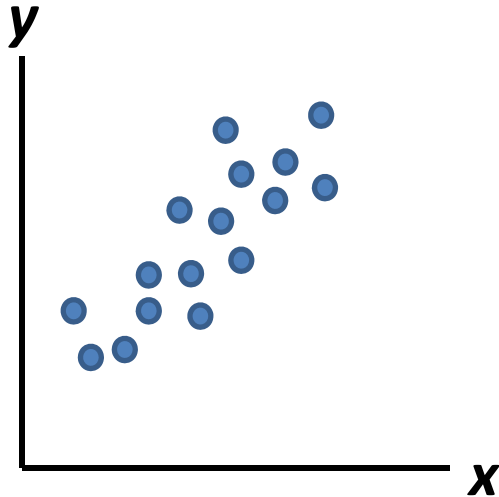


Fig. 3: Moderate positive linear relationship

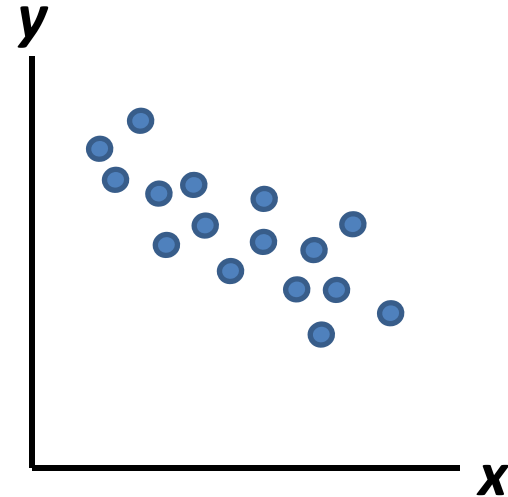


Fig. 4: Moderate negative linear relationship

Strength and trend of relationships:

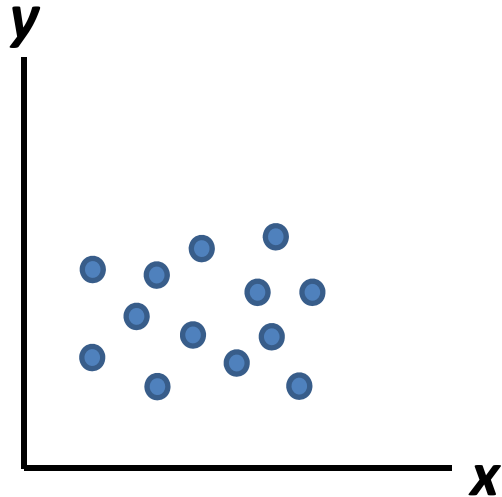


Fig. 5: No clear relationship

We will study about the direction and degree of linear relationships.

Two aspects – graphical and quantitative

Bivariate plots:

Provide first hand visual information about the nature and degree of relationship between two variables.

Relationship can be linear or nonlinear.

We discuss several types of plots through examples.

Bivariate scatter plots:

Plot command:

x, y: Two data vectors

plot(x, y)

plot(x, y, type)

type	
"p" for <u>p</u> oints	"l" for <u>l</u> ines
"b" for <u>b</u> oth	"c" for the lines part alone of "b"
"o" for both ' <u>o</u> verplotted'	"s" for stair <u>s</u> teps.
"h" for ' <u>h</u> istogram' like (or ' <u>h</u> igh-density') vertical lines	

Bivariate scatter plots:

Plot command

x, y: Two data vectors

plot(x, y)

plot(x, y, type)

Get more details from help: **help("type")**

Other options:

main an overall title for the plot.

suba sub title for the plot.

xlab title for the x axis.

ylab title for the y axis.

aspthe y/x aspect ratio.

Bivariate scatter plots:

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

Bivariate scatter plots:

Example

marks =

```
c(337, 316, 327, 340, 374, 330, 352, 353, 370, 380, 384, 398, 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, 44, 45, 46, 44, 41)
```

Bivariate scatter plots:

Example

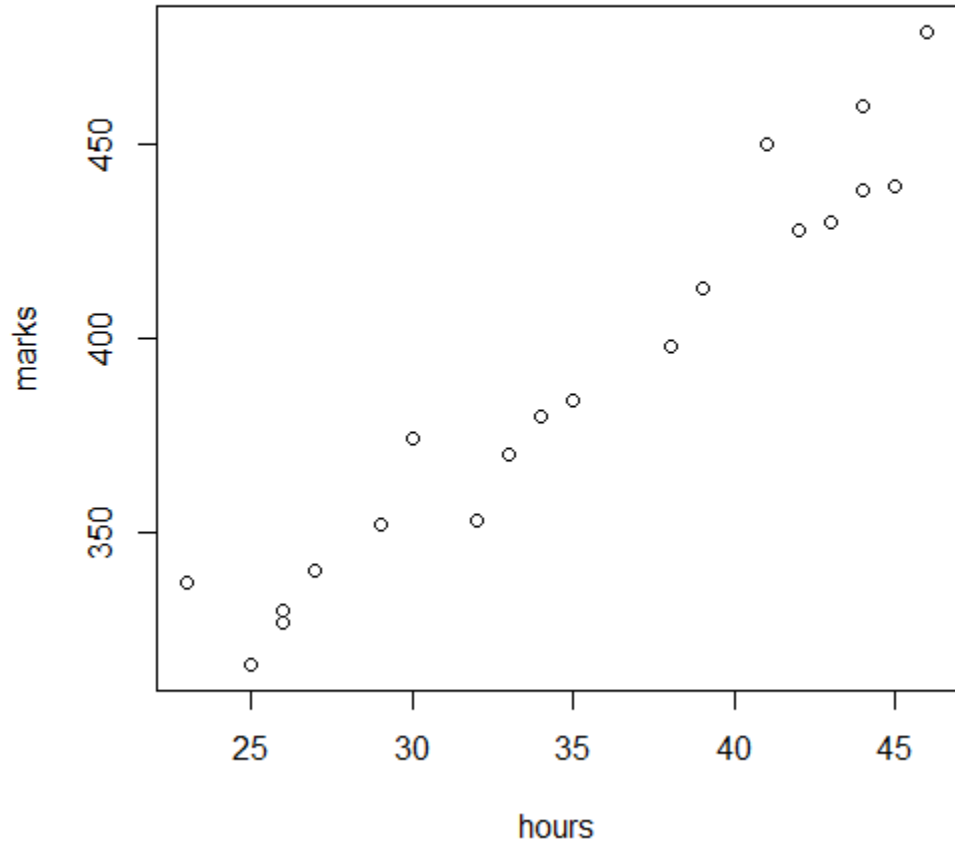
plot command:

x, y: Two data vectors

Various type of plots are possible to draw.

plot(x, y)

plot(hours, marks)

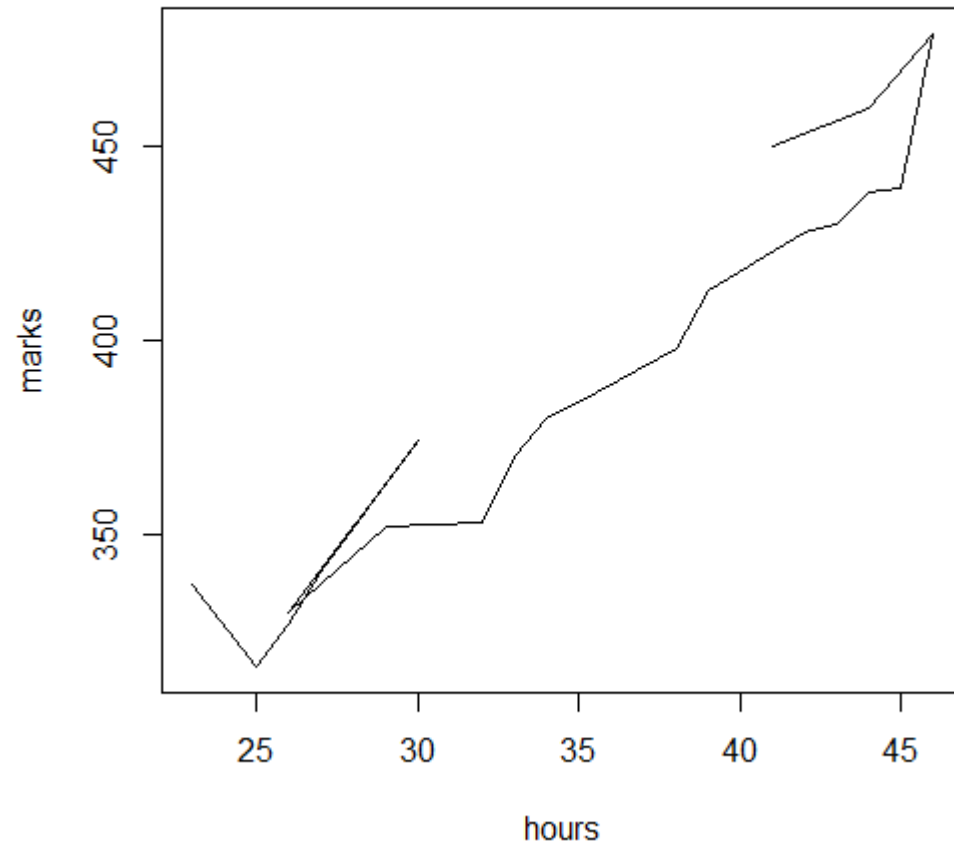


Bivariate scatter plots:

Example

```
plot(hours, marks, "l")
```

“l” for lines,

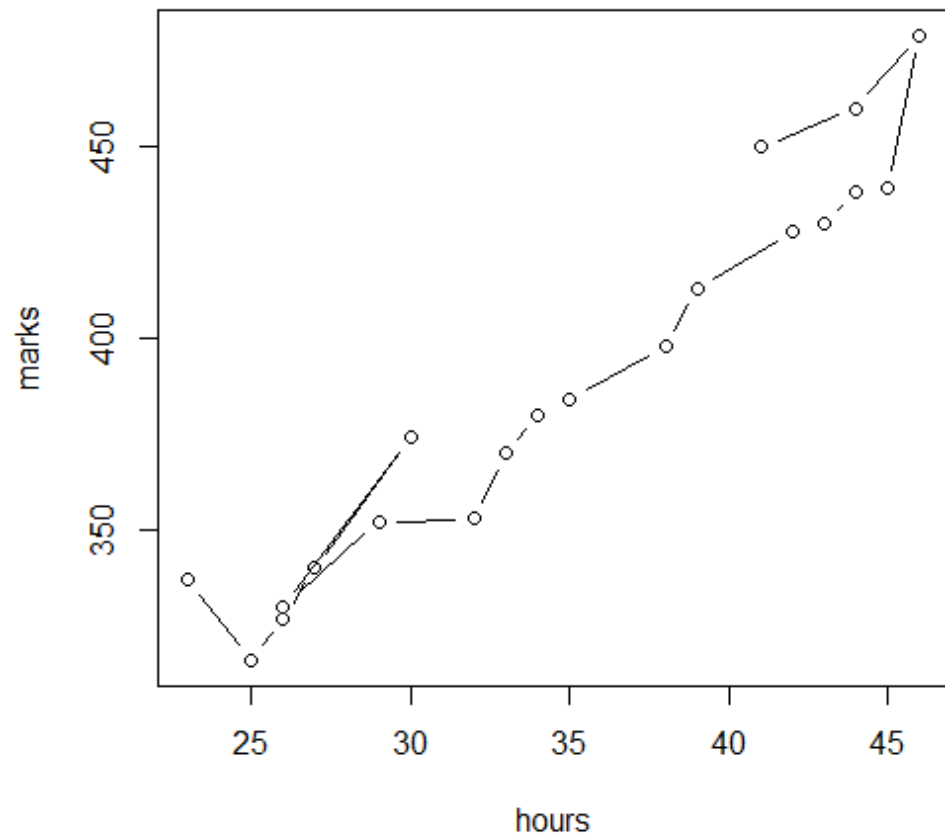


Bivariate scatter plots:

Example

`plot(hours, marks, "b")`

“b” for both – line and point

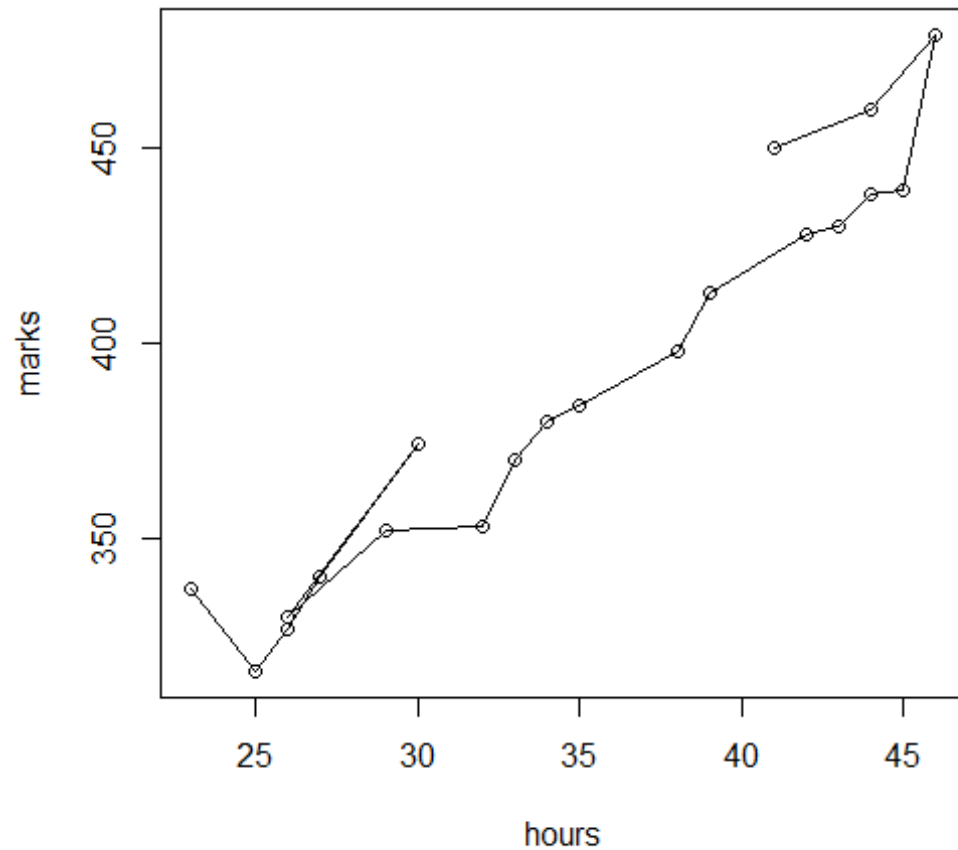


Bivariate scatter plots:

Example

```
plot(hours, marks, "o")
```

“o” for both ‘overplotted’

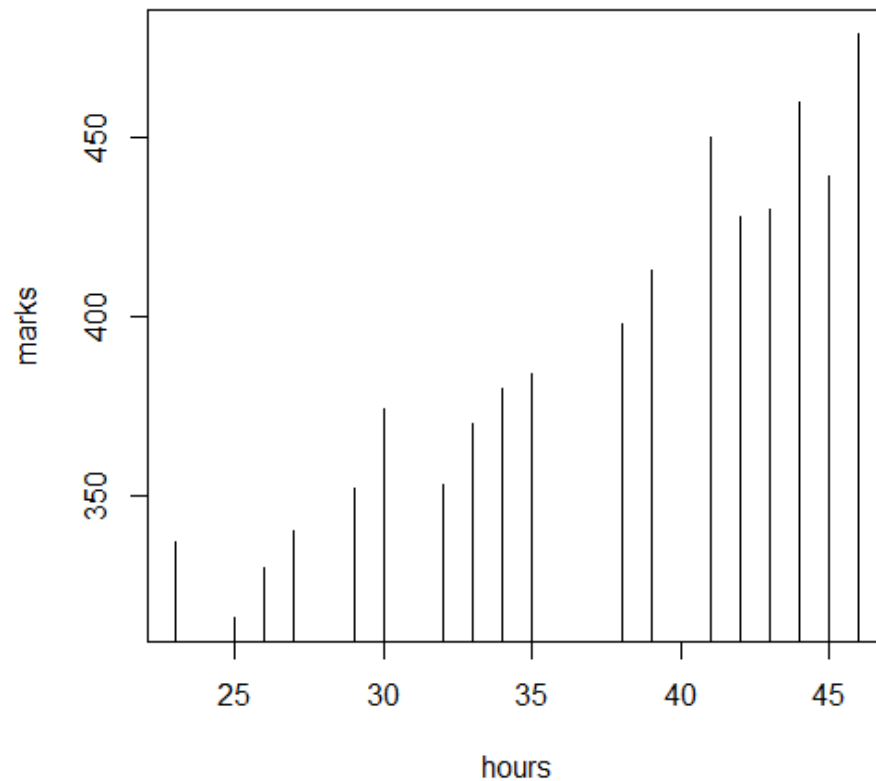


Bivariate scatter plots:

Example

```
plot(hours, marks, "h")
```

"h" for 'histogram' like (or 'high-density') vertical lines

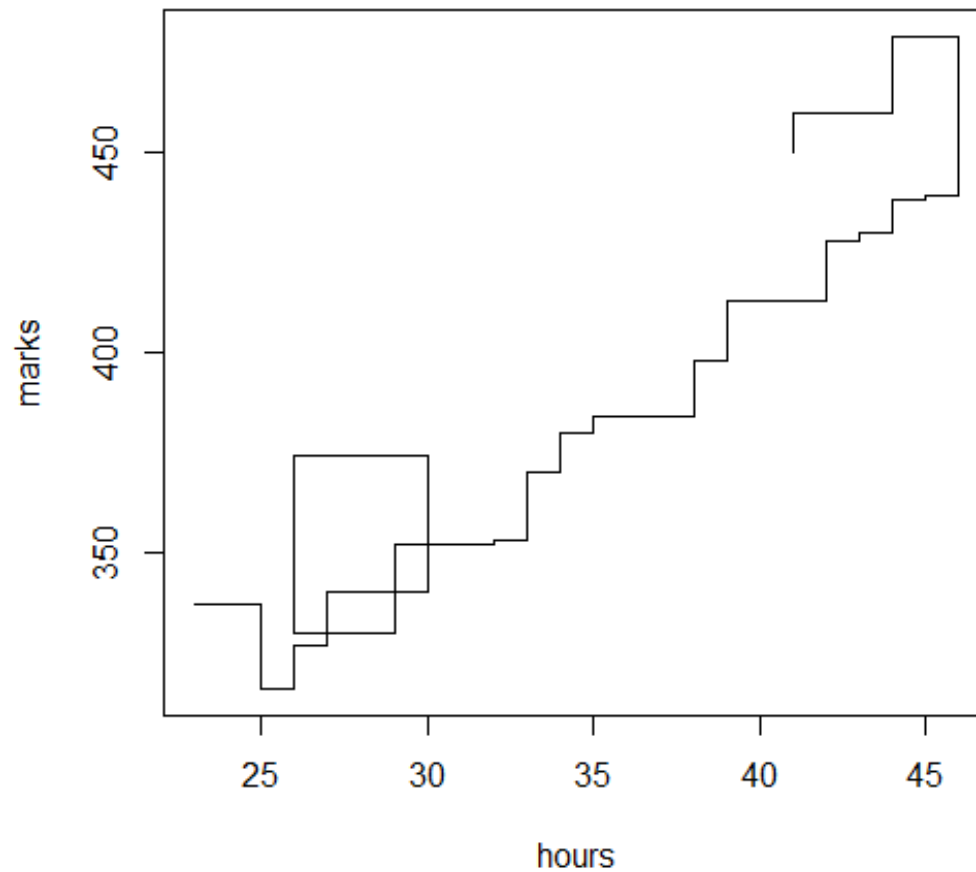


Bivariate scatter plots:

Example

`plot(hours, marks, "s")`

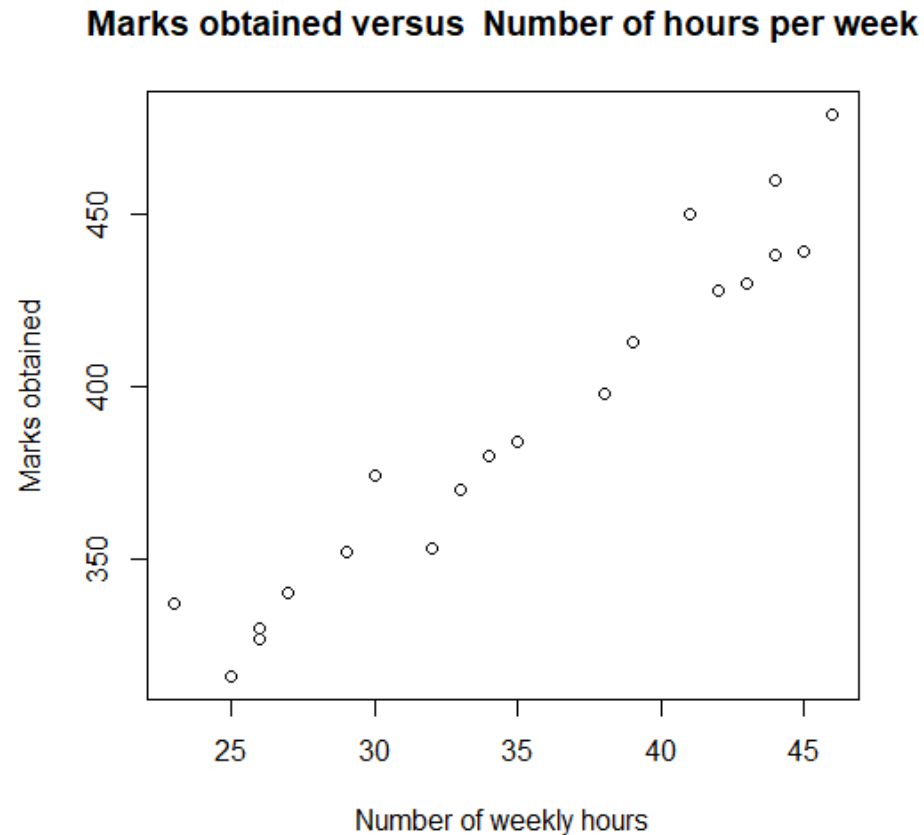
“s” for stair steps.



Bivariate scatter plots:

Example

```
plot(hours, marks, xlab="Number of weekly  
hours", ylab="Marks obtained", main="Marks  
obtained versus Number of hours per week")
```

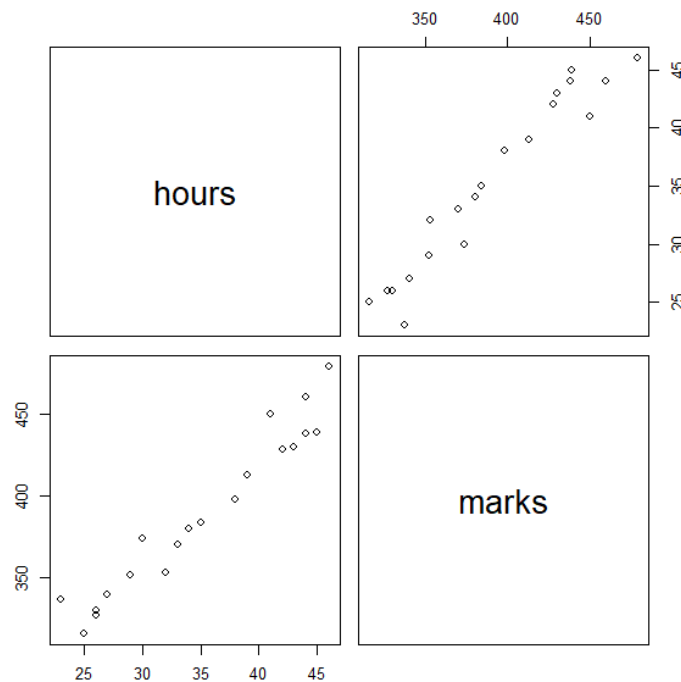


Matrix Scatter plot:

The command `pairs()` allows the simple creation of a matrix of scatter plots.

Example

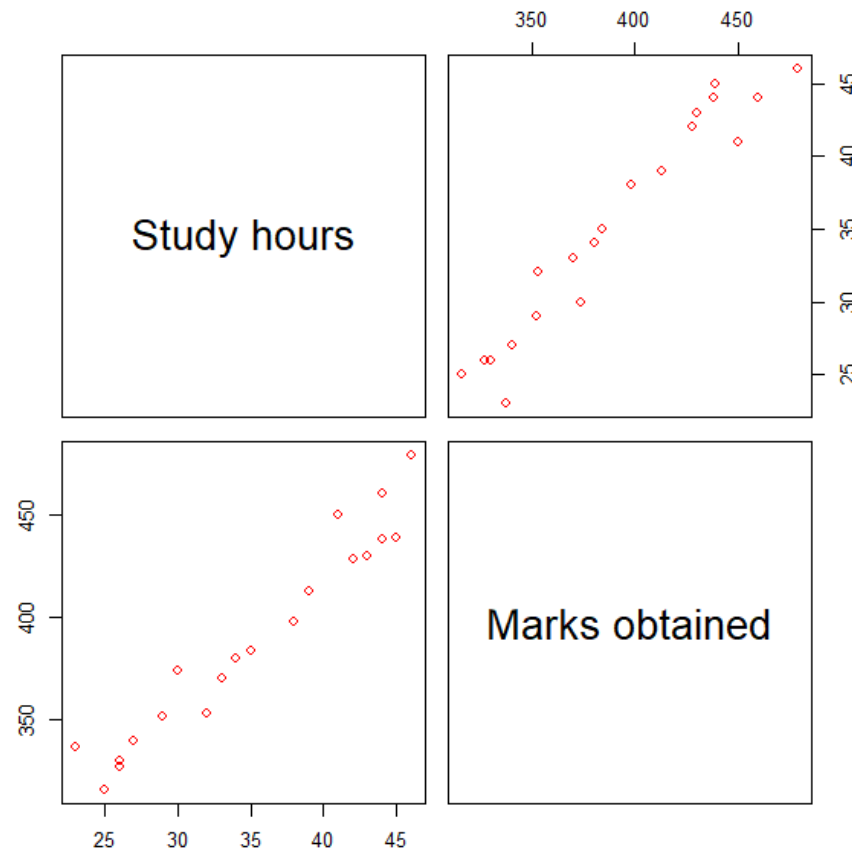
```
pairs(cbind(hours, marks))
```



Matrix Scatter plot

Example: Adding labels and colour

```
pairs(cbind(hours, marks), labels=c("Study  
hours", "Marks obtained"), col="red" )
```



Scatter plots with smooth curve:

Suppose two variables are related.

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.

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.

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.

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

Scatter plots with smooth curve:

Example

marks =

```
c(337, 316, 327, 340, 374, 330, 352, 353, 370, 380, 384, 398, 413, 428, 430, 438, 439, 479, 460, 450)
```

hours =

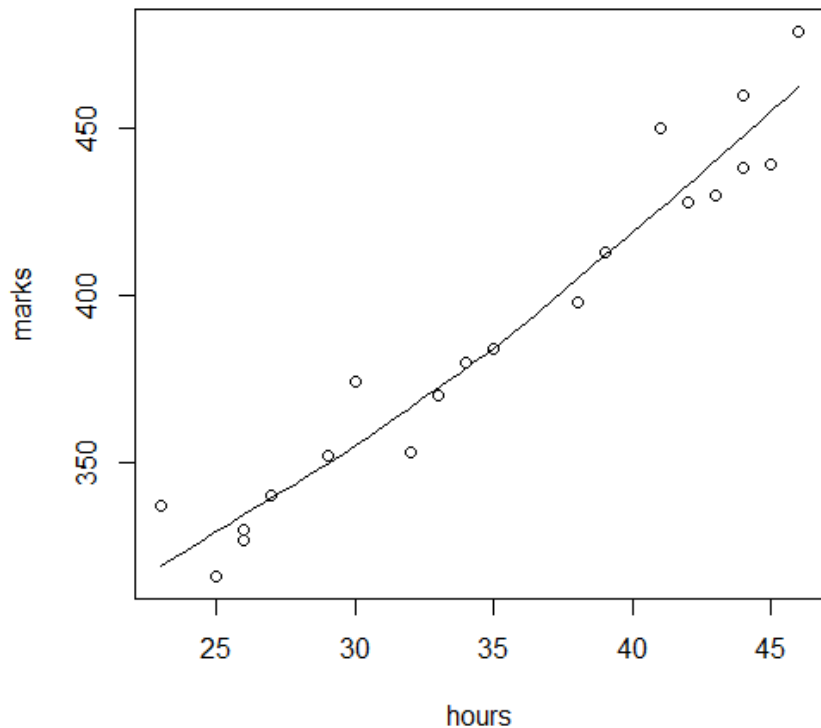
```
c(23, 25, 26, 27, 30, 26, 29, 32, 33, 34, 35, 38, 39, 42, 43, 44, 45, 46, 44, 41)
```

Scatter plots with smooth curve:

Example

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

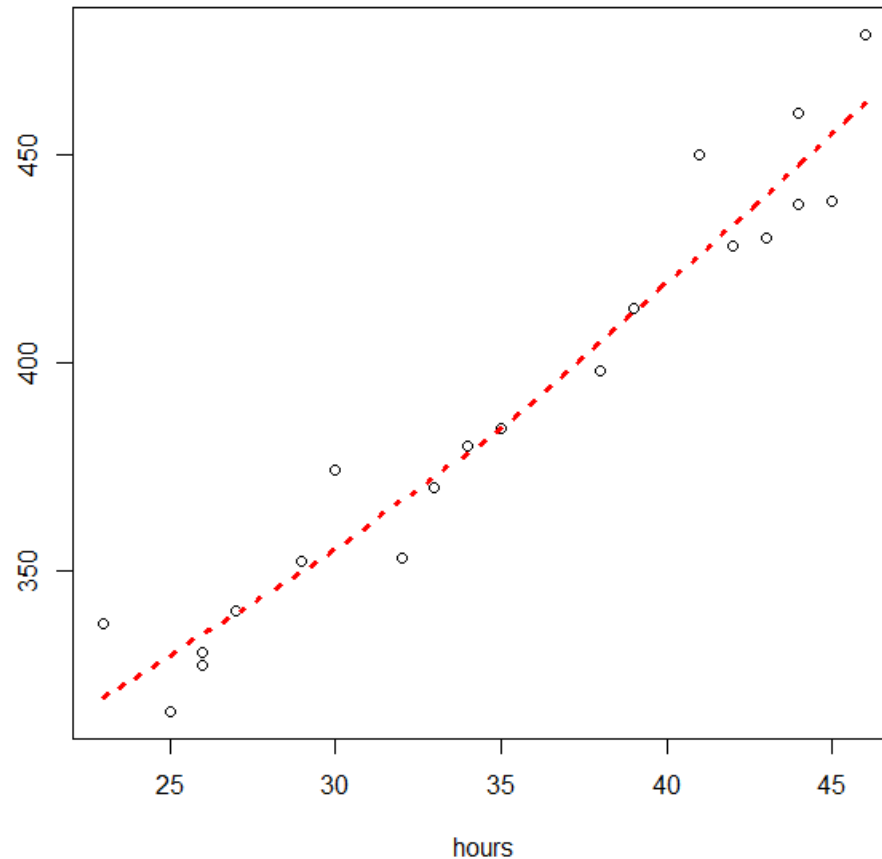
Example: `scatter.smooth(hours, marks)`



Scatter plots with smooth curve:

Example: Add more options

```
scatter.smooth(hours, marks, lpar = list(col =  
"red", lwd = 3, lty = 3))
```



Three dimensional scatter plot:

```
scatterplot3d(x, y, z)
```

Plots a three dimensional (3D) point cloud of the data in **x**, **y** and **z**

Need a package **scatterplot3d**

```
install.packages("scatterplot3d")
```

```
library(scatterplot3d)
```

Three dimensional scatter plot:

Example

The data on height (in cms.), weight (in kg.) and age (in years) of 5 persons are recorded as follows. We would like to create a 3 dimensional plot for this data.

Person No.	Height (Cms.)	Weight (Kg.)	Age (Years)
1	100	30	10
2	125	35	15
3	145	50	20
4	160	65	30
5	170	70	35

Three dimensional scatter plot:

`scatterplot3d()` Plots a three dimensional (3D) point cloud

```
install.packages("scatterplot3d")
```

```
library(scatterplot3d)
```

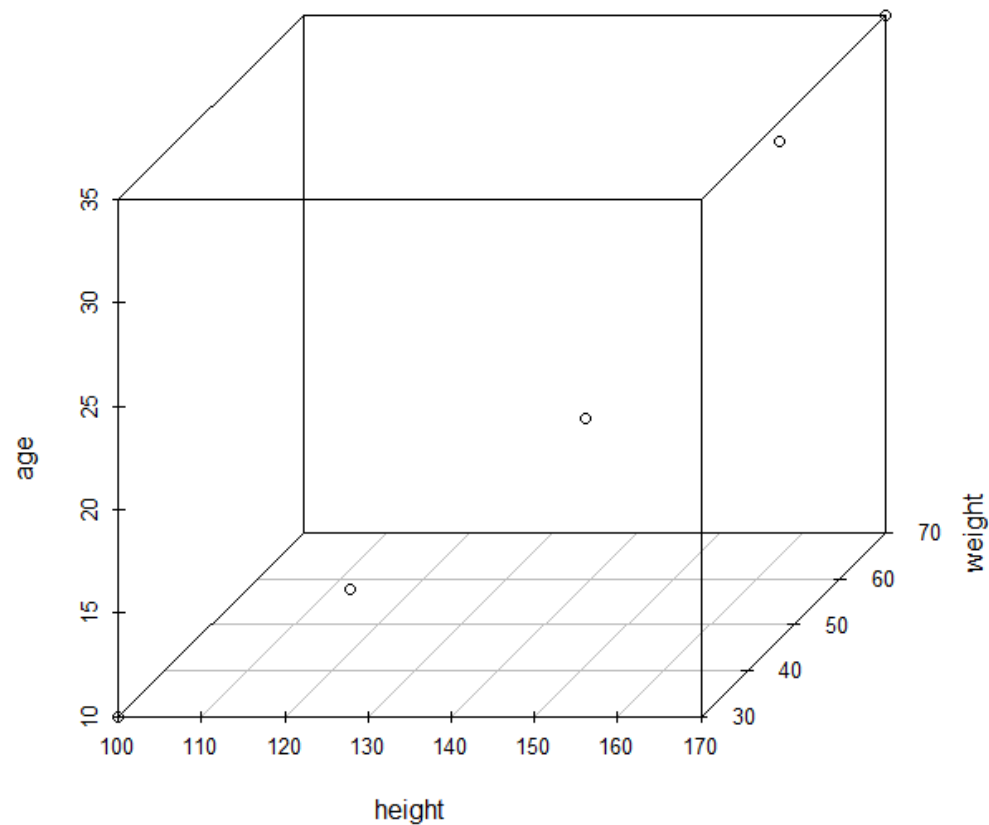
```
height = c(100, 125, 145, 160, 170)
```

```
weight = c(30, 35, 50, 65, 70)
```

```
age = c(10, 15, 20, 30, 35)
```

Three dimensional scatter plot:

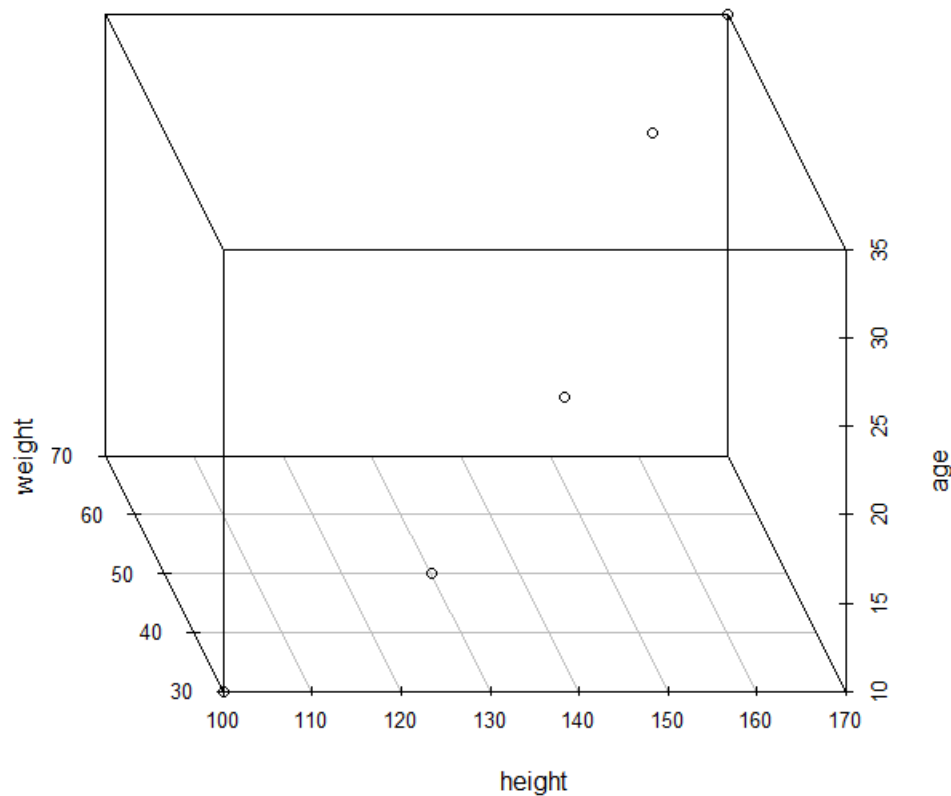
`scatterplot3d(height, weight, age)`



Three dimensional scatter plot:

Direction of the figure can be changed.

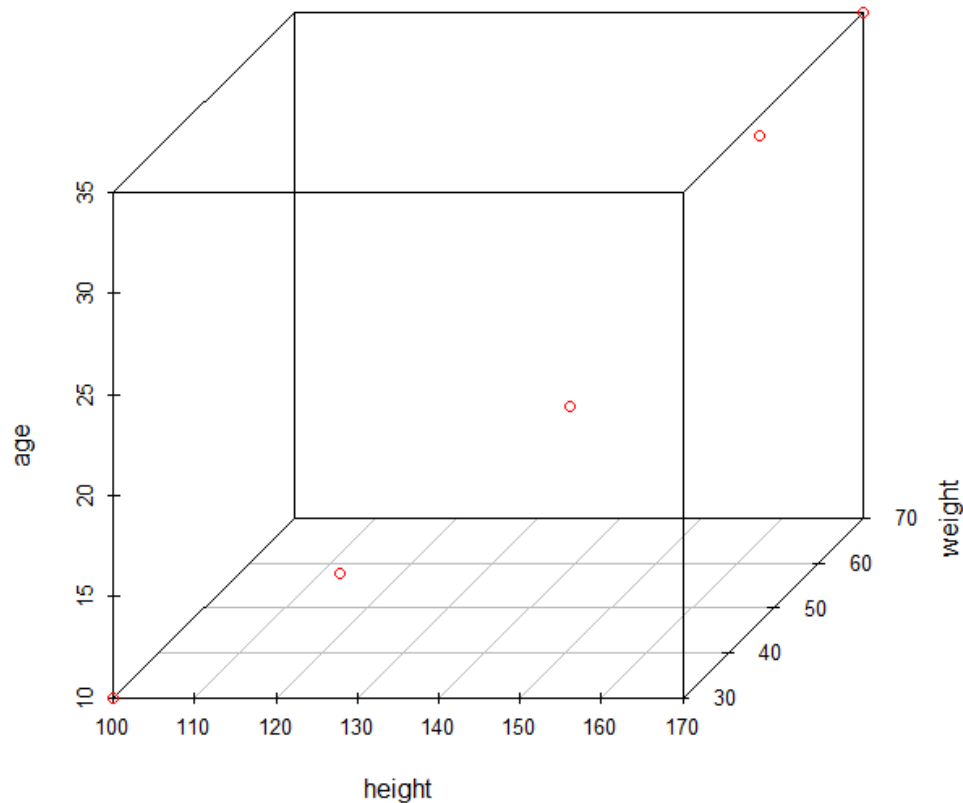
```
scatterplot3d(height, weight, age, angle = 120)
```



Three dimensional scatter plot:

Colours of points can be changed.

```
scatterplot3d(height, weight, age, color="red")
```



More functions:

- `contour()` for contour lines
- `dotchart()` for dot charts (replacement for bar charts)
- `image()` pictures with colors as third dimension
- `mosaicplot()` mosaic plot for (multidimensional) diagrams of categorical variables (contingency tables)
- `persp()` perspective surfaces over the x–y plane

More functions:

Example of perspective plot

persp() perspective surfaces over the x–y plane

```
x = seq(-10, 10, length= 30)
```

```
y = x
```

```
f =function(x,y){r = sqrt(x^2+y^2);10*sin(r)/r}
```

```
z = outer(x, y, f)
```

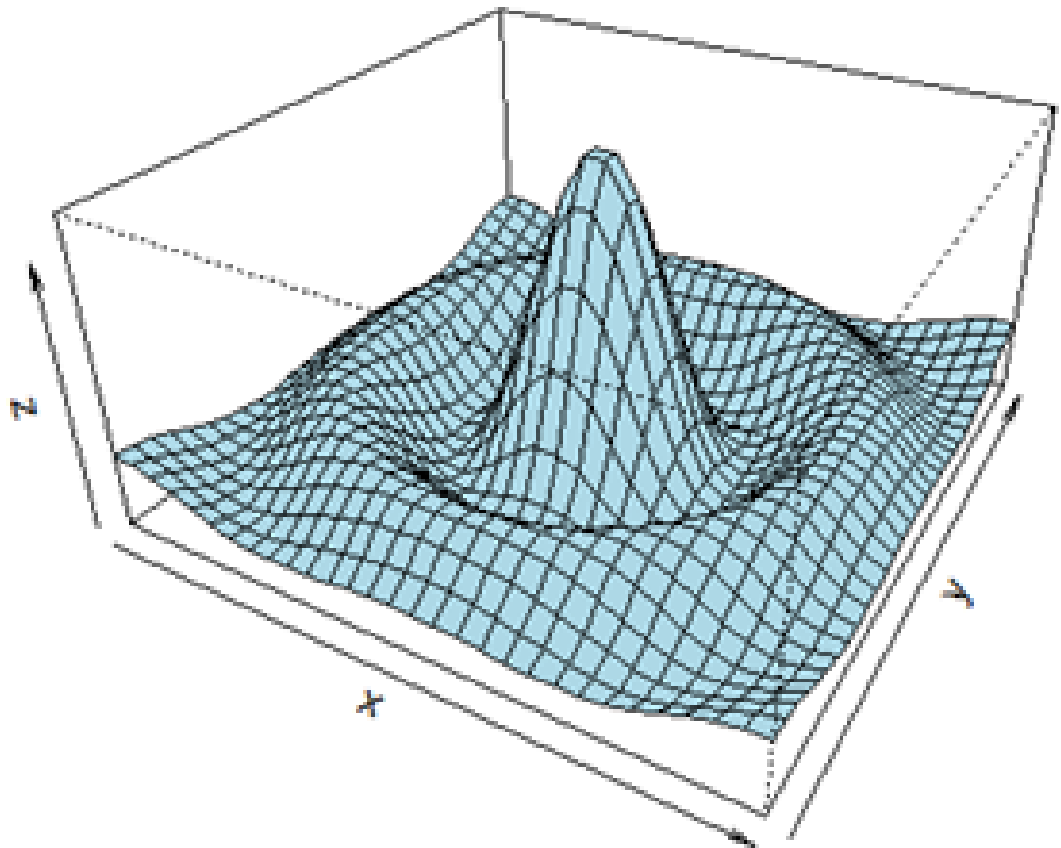
```
z[is.na(z)] = 1
```

```
op = par(bg = "white")
```

More functions:

Example of perspective plot

```
persp(x,y,z, theta=30, phi=30, expand=0.5, col=  
"lightblue")
```



More functions:

Example of perspective plot

```
persp(x, y, z, theta = 30, phi = 30, expand =  
0.5, col = "lightblue", ltheta = 120, shade =  
0.75, ticktype = "detailed", xlab = "X", ylab =  
"Y", zlab = "Sinc( r )")
```

