Statistical analysis using R

Written by:

عواطف المطيري
سناء أبو نصرة
نوره المسعود
خلود باسالم
العنود الزغيبي
زهراء الكعبي

Presented by:

نوره المسعود
أمل المحيسن

Outline

1. Introduction. What & Why R
2. Brief Navigation of R.
3. Types of R objects and Data.
4. Basic Command in R.
5. Matrices.
6. Some Statistical Distributions.
7. Graphics.
8. Simple Linear Regression

Statistical analysis

Once you have collected quantitative data, you will have a lot of numbers.

It's now time to carry out some statistical analysis to make sense of, and draw some inferences from, your data.

There is a wide range of possible techniques that you can use.

We will provides a brief summary of some of the most common techniques for summarizing

your data, and explains when you would use each one by using **R**

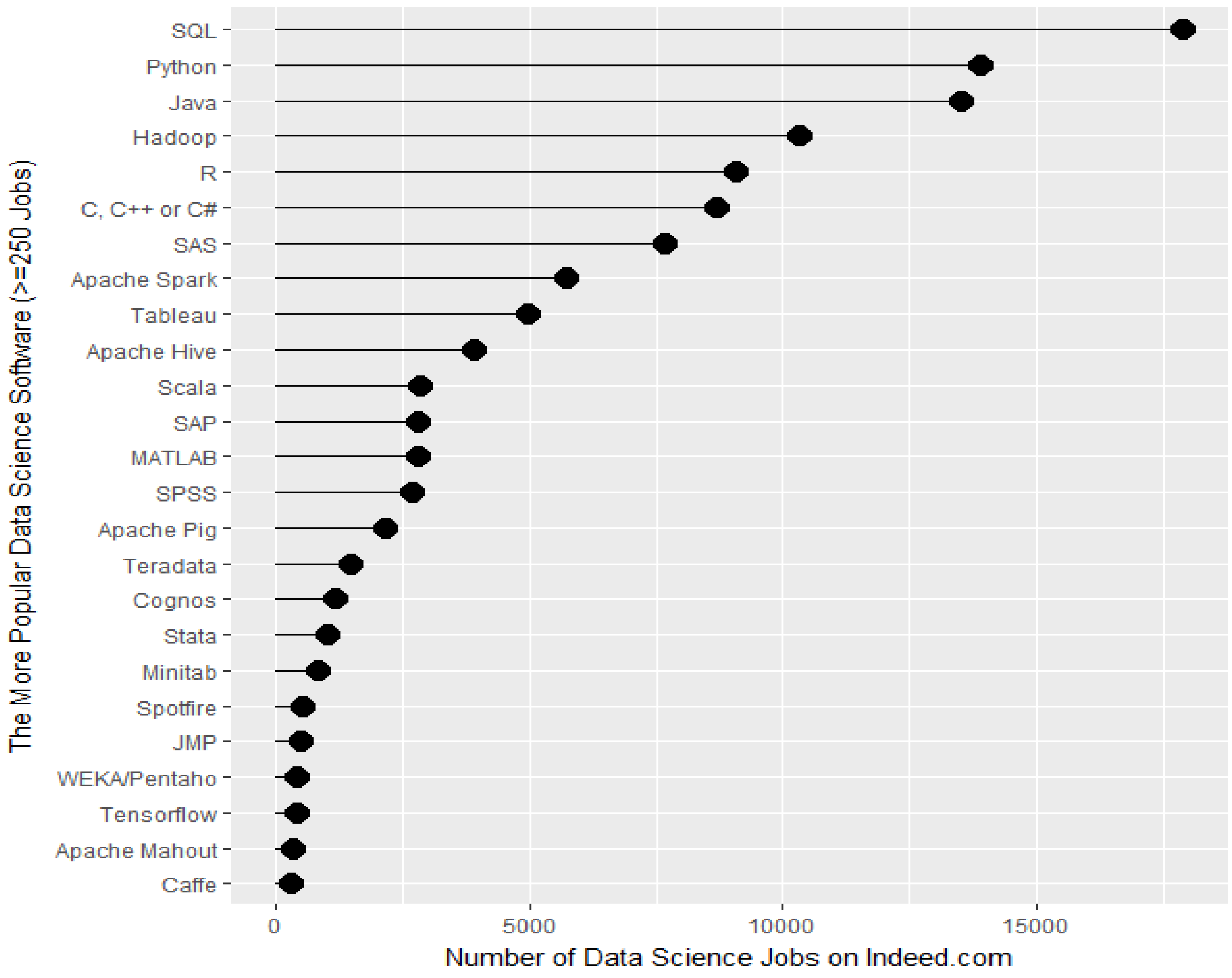
What is R ?

- R is a programming language.
- R is an open-source software environment for statistical computing and graphics .
- R works with a command-line interface, meaning you type in commands telling R what to do .
- For more information and to download R, visit Cran.r-project.org

Why learn R ?

- It supports larger data sets. Excel ~ 1 Million , R~2 Billion vector index limit
- Faster. i.e. 100K in excel ~15 mins vs 1M ~30 second
- It reads any type of data.
- Availability of instant access to over 7800 packages customized for various computation tasks.
- Get high performance computing experience.
- Advanced Statistics capabilities.
- The community support is overwhelming. There are numerous forums to help you out. For example:
    1. Numerous Discipline Specific R Groups
    2. Numerous Local R User Groups (including R-Ladies Groups)
    3. Stack Overflow
- Learning Resources (quantity and quality)
    1. R books
    2. (Free Online) R Books
    3. https://www.datasciencecentral.com/profiles/blogs/600-websites-about-r

Reproducibility (important for detecting errors)


The More Popular Data Science Software (>=250 Jobs)
SQL
Python
Java
Hadoop
R
C, C++ or C#
SAS
Apache Spark
Tableau
Apache Hive
Scala
SAP
MATLAB
SPSS
Apache Pig
Teradata
Cognos
Stata
Minitab
Spotfire
JMP
WEKA/Pentaho
Tensorflow
Apache Mahout
Caffe
0
5000
10000
15000
Number of Data Science Jobs on Indeed.com

Difference Between R & RStudio

R: Engine	RStudio: Dashboard
	

R: Do not open this	RStudio: Open this

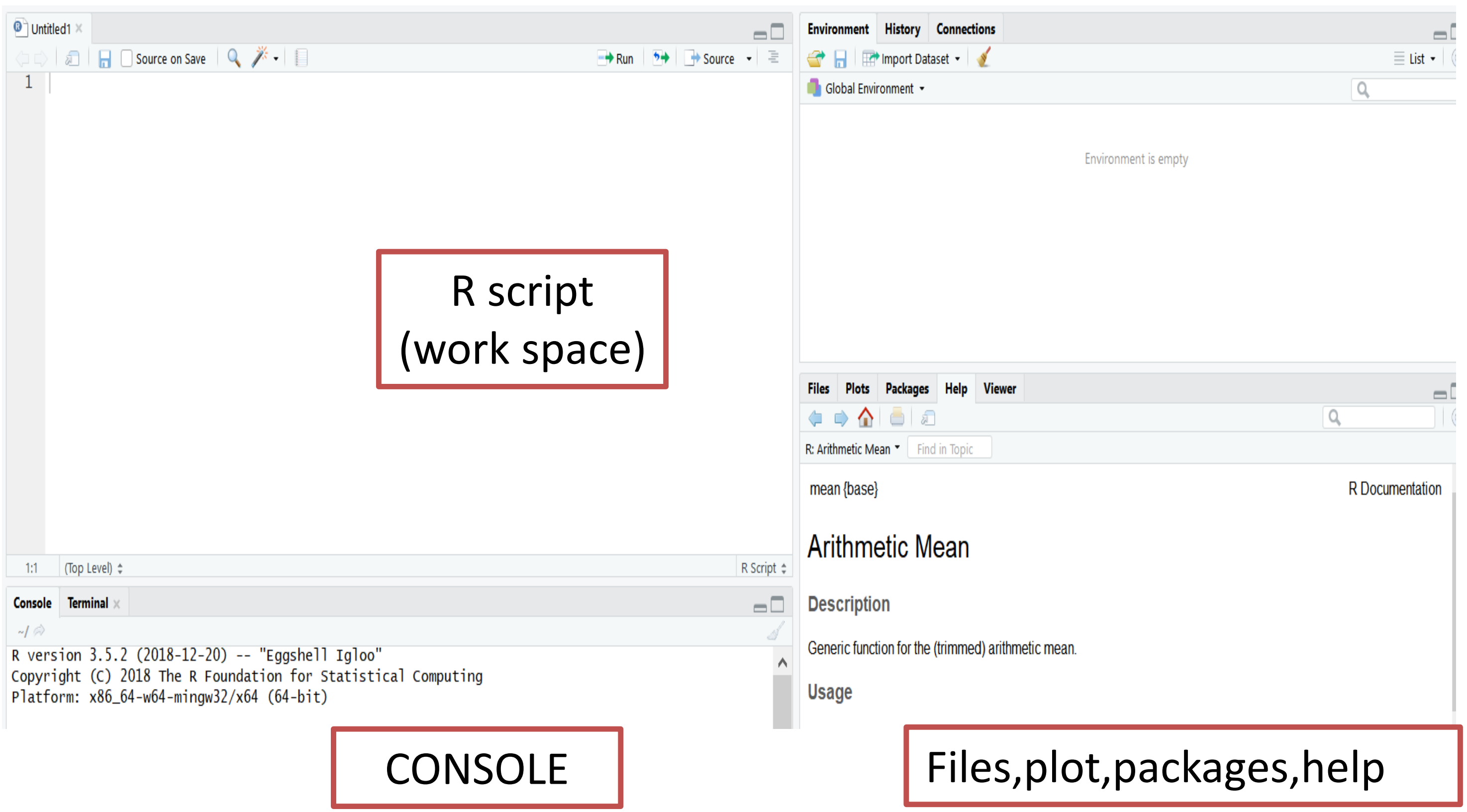
Getting started with R

How to install R and R Studio ?

https://www.youtube.com/watch?v=Ohnk9hcxf9M&feature=youtu.be

R studio (nice editor and features)

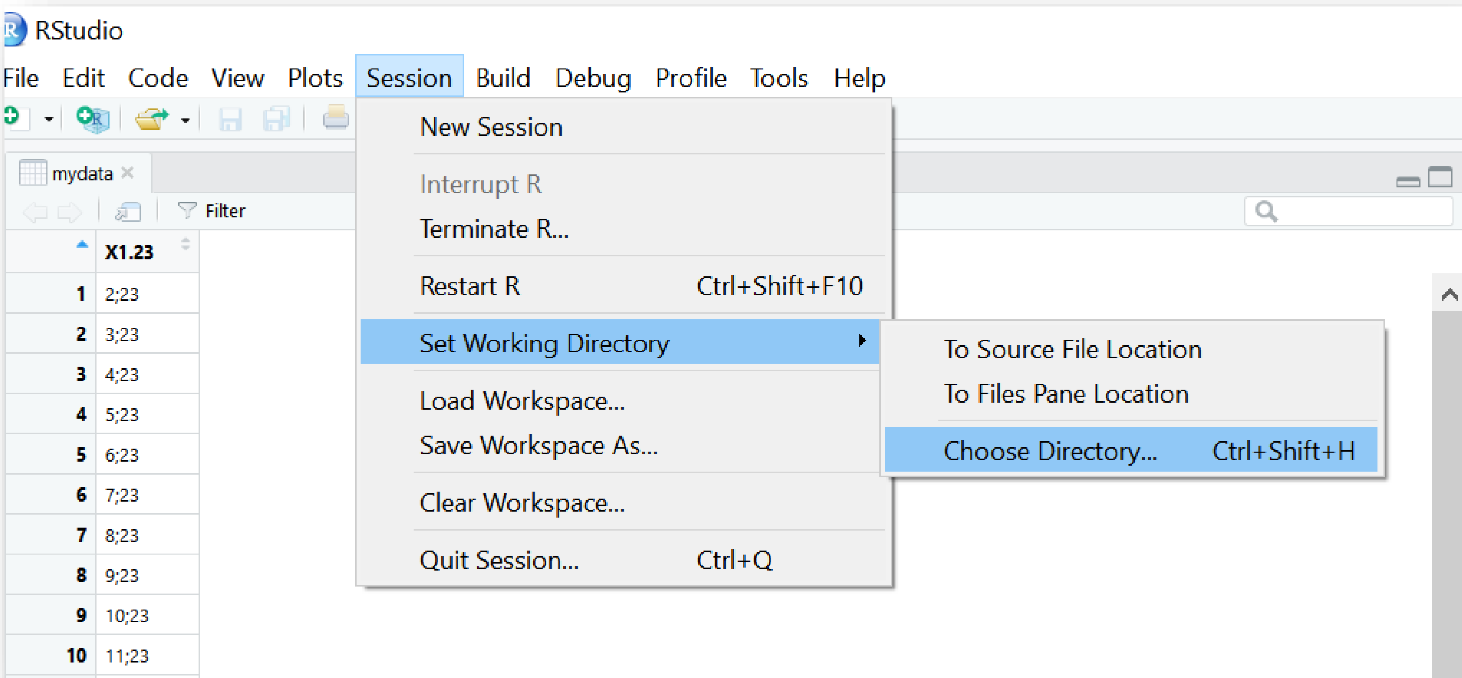
R studio


Untitled1
Source on Save
Run
Source
R script
(work space)
(Top Level)
R Script
Console
Terminal
R version 3.5.2 (2018-12-20) -- "Eggshell Igloo"
Copyright (C) 2018 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)
Environment
History
Connections
Import Dataset
List
Global Environment
Environment is empty
Files
Plots
Packages
Help
Viewer
R: Arithmetic Mean
Find in Topic
mean {base}
R Documentation
Arithmetic Mean
Description
Generic function for the (trimmed) arithmetic mean.
Usage
CONSOLE
Files,plot,packages,help

Working directory

If you need to check the current working directory use :getwd()

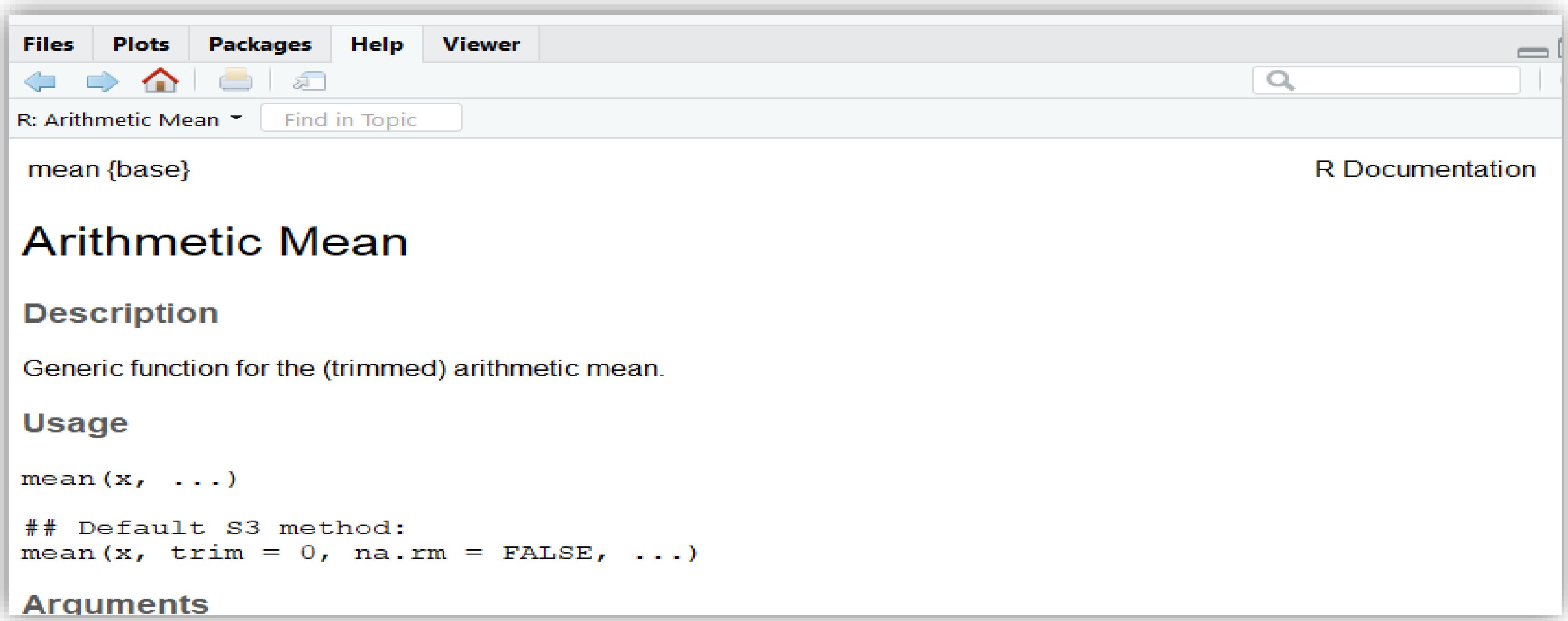
If you need to change the current working directory use the following step :



Help in R

To get more information on any built-in R commands, simply type the following and this will bring up a separate help page.

```
> help("mean")
> ?mean
```



Loading Data

From your spreadsheet editing program (Excel, Google Docs, etc) save your spreadsheet as a csv. File on your computer.

In R, decide on a name for your dataset. Usually a short name relevant to the particular dataset is best. Lets assume you picked the name (mydata).

Type

```
> mydata = read.csv("mydata.csv")
> 
```

Importing and exporting data

There are many ways to get data into R and out of R.

Most programs (e.g. Excel), as well as humans, know how to deal with rectangular tables in the form of tab-delimited text files.

```
> x = read.delim("filename.txt")
```

also: read.table, read.csv

```
> write.table(x, file="x.txt", sep="\t")
```

R packages

1. Installation

   install.packages(“packagename")

2. Loading

   Library(packagename)

3. Use

   ?packagename

Saved files

- You can scroll back to previous commands typed by using the `up' arrow key and `down' to scroll back again.
- You can also `copy' and `paste' using standard windows editor techniques (for example, using the `copy' and `paste' dialog buttons) .
- If at any point you want to save the transcript of your session, click on `File' and then `Save', which will enable you to save a copy of the commands you have used for later use.
- As an alternative you might copy and paste commands manually into a note pad editor or something similar. You finish an R session by typing
- <q ( ).

Five basic classes of objects

What is an object?

- **Numeric :** (Real Numbers)
- **Integer :** (Whole Numbers)
- **character**
- **Logical** (True / False)
- **complex**

Example

- <- c(1.8, 4.5)   #numeric
- b <- c(1 + 2i, 3 - 6i) #complex
- d <- c(23, 44)   #integer
- e <- rep(c("Male”,Female"),each=5)

Data Types in R

- **Vector**: a vector contains object of same class.

  Ex: bar <- 0:5

- **Matrices**: When a vector is introduced with row and column i.e. a dimension attribute, it becomes a matrix. A matrix is represented by set of rows and columns. It is a 2 dimensional data structure. It consist of elements of same class.

  Ex: my_matrix <- matrix(1:6, nrow=3, ncol=2)

- **Data Frame**: This is the most commonly used member of data types family. It is used to store tabular data. It is different from matrix. In a matrix, every element must have same class. But, in a data frame, you can put list of vectors containing different classes.
  - Ex: df <- data.frame(name = c("Sara","Wafa","Norah","Reem"), score = c(67,56,87,91))
- **List:** A list is a special type of vector which contain elements of different data types.
  - Ex: my_list <- list(22, "ab", TRUE, 1 + 2i)

Basic Commands

Arithmetic operations:

Operator	Description
+	addition
-	subtraction
*	multiplication
/	division
^ or **	exponentiation
x %% y	modulus (x mod y) 5%%2 is 1
x %/% y	integer division 5%/%2 is 2

```
> 2+10
[1] 12
> 11-10
[1] 1
> 3*5
[1] 15
> 2^3
[1] 8
> 2**3
[1] 8
> 6/2
[1] 3
> 6%/%4
[1] 1
> 5%%2
[1] 1
```

Mathematical functions :

```
> log(5)
[1] 1.609438
> exp(-2)
[1] 0.1353353
> log10(10)
[1] 1
> sqrt(16)
[1] 4
> factorial(3)
[1] 6
> choose(4,2)
[1] 6
> gamma(5)
[1] 24

> floor(3.66)
[1] 3
```

```
> cos(0)
[1] 1
> sin(0)
[1] 0
> tan(45)
[1] 1.619775
> acos(1)
[1] 0
> acosh(60)
[1] 4.787422
> abs(-9)
[1] 9
> pi
[1] 3.141593
```

Mathematical functions used in R.

Function	Meaning
log(x)	log to base e of x
exp(x)	antilog of x (e^x)
log(x,n)	log to base n of x
log10(x)	log to base 10 of x
sqrt(x)	square root of x
factorial(x)	$x!$
choose(n,x)	binomial coefficients $n!/(x!\,(n-x)!)$
gamma(x)	$\Gamma(x)$, for real x $(x-1)!$, for integer x
lgamma(x)	natural log of $\Gamma(x)$
floor(x)	greatest integer $< x$
ceiling(x)	smallest integer $> x$
trunc(x)	closest integer to x between x and 0 trunc(1.5) $= 1$, trunc(-1.5) $= -1$ trunc is like floor for positive values and like ceiling for negative values
round(x, digits=0)	round the value of x to an integer
signif(x, digits=6)	give x to 6 digits in scientific notation
runif(n)	generates n random numbers between 0 and 1 from a uniform distribution
cos(x)	cosine of x in radians
sin(x)	sine of x in radians
tan(x)	tangent of x in radians
acos(x), asin(x), atan(x)	inverse trigonometric transformations of real or complex numbers
acosh(x), asinh(x), atanh(x)	inverse hyperbolic trigonometric transformations of real or complex numbers
abs(x)	the absolute value of x, ignoring the minus sign if there is one

Vector :

```
> x<-c(1,6,4,100)
> x
[1]   1   6   4 100
> y<-c(1:4)
> y
[1] 1 2 3 4
>
```

Vector Functions:

```
> x<-c(1,6,4,100)
> y<-(1:4)
> x+y
[1]   2   8   7 104
> x*y
[1]   1  12  12 400
> sum(x)
[1] 111
> min(y)
[1] 1
> max(y)
[1] 4
> mean(x)
[1] 27.75
> median(x)
[1] 5
> range(x)
[1]   1 100
> var(x)
[1] 2324.25
> sd(x)
[1] 48.21048
> cor(x,y)
[1] 0.7899598
```

```
> sort(x)
[1]   1   4   6 100
> rank(x)
[1] 1 3 2 4
> summary(x)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1.00    3.25    5.00   27.75   29.50  100.00
> cumsum(x)
[1]   1   7  11 111
> prod(y)
[1] 24
```

Vector functions used in R.

Operation	Meaning
max(x)	maximum value in x
min(x)	minimum value in x
sum(x)	total of all the values in x
mean(x)	arithmetic average of the values in x
median(x)	median value in x
range(x)	vector of min(x) and max(x)
var(x)	sample variance of x
cor(x,y)	correlation between vectors x and y
sort(x)	a sorted version of x
rank(x)	vector of the ranks of the values in x
order(x)	an integer vector containing the permutation to sort x into ascending order
quantile(x)	vector containing the minimum, lower quartile, median, upper quartile, and maximum of x
cumsum(x)	vector containing the sum of all of the elements up to that point
cumprod(x)	vector containing the product of all of the elements up to that point
cummax(x)	vector of non-decreasing numbers which are the cumulative maxima of the values in x up to that point
cummin(x)	vector of non-increasing numbers which are the cumulative minima of the values in x up to that point
pmax(x,y,z)	vector, of length equal to the longest of x, y or z, containing the maximum of x, y or z for the ith position in each

Logical Operators

Operator	Description
<	less than
<=	less than or equal to
>	greater than
>=	greater than or equal to
==	exactly equal to
!=	not equal to
!x	Not x
x | y	x OR y
x & y	x AND y

```
> xx<-c(1:10)
> xx[x<=5|x>8]
[1] 1 3 4 5 7 8 9
> 5==3
[1] FALSE
> 3==3
[1] TRUE
> 3!=5
[1] TRUE
>
```

Matrices

Write matrix and Dimensions

```
> A<-matrix(c(2,3,4,5),nrow=2,ncol=2)
> A
     [,1] [,2]
[1,]    2    4
[2,]    3    5
> B<-matrix(c(1,0,0,8),nrow=2,ncol=2)
> B
     [,1] [,2]
[1,]    1    0
[2,]    0    8
```

```
> dim(A)
[1] 2 2
> dim(B)
[1] 2 2
> A[2,1]
[1] 3
```

Multiplication

```
> A+B
     [,1] [,2]
[1,]    3    4
[2,]    3   13
> B-A
     [,1] [,2]
[1,]   -1   -4
[2,]   -3    3

> B/A
     [,1] [,2]
[1,]  0.5  0.0
[2,]  0.0  1.6
> A%*%B
     [,1] [,2]
[1,]    2   32
[2,]    3   40
> 2*A
     [,1] [,2]
[1,]    4    8
[2,]    6   10
```

Transpose

```
> t(A)
     [,1] [,2]
[1,]    2    3
[2,]    4    5
```

Inverse

```
> solve(A)
     [,1] [,2]
[1,] -2.5    2
[2,]  1.5   -1
> 
```

Statistical distributions in R

Common Probability Distribution Functions in R

- **Probability functions**
  - f(x) P(X=x)
  - d
- **Cumulative distribution function**
  - F(x)=p(X≤x)
  - p
- **Inverse distribution function**
  - X=$F(x)^{-1}$
  - q
- **Random sample**
  - r

R function	Distribution	Parameters
beta	beta	shape1, shape2
binom	binomial	sample size, probability
cauchy	Cauchy	location, scale
exp	exponential	rate (optional)
chisq	chi-squared	degrees of freedom
f	Fisher's F	df1, df2
gamma	gamma	shape
geom	geometric	probability
hyper	hypergeometric	m, n, k
lnorm	lognormal	mean, standard deviation
logis	logistic	location, scale
nbinom	negative binomial	size, probability
norm	normal	mean, standard deviation
pois	Poisson	mean
signrank	Wilcoxon signed rank statistic	sample size n
t	Student's t	degrees of freedom
unif	uniform	minimum, maximum (opt.)
weibull	Weibull	shape
wilcox	Wilcoxon rank sum	m, n

Discrete Probability Distributions

The binomial distribution:

```
> #p(x=3)
>
> dbinom(3,20,1/6)
[1] 0.2378866
>
> #p(x<=3)
>
> pbinom(3,20,1/6)
[1] 0.5665456
>
> #random sample
>
> rbind(3,20,1/6)
           [,1]
[1,]  3.0000000
[2,] 20.0000000
[3,]  0.1666667
>
> #invers
> qbinom(0.105,20,1/6)
[1] 1
>
```

X is Binomial Distribution with n=20 trials and p=1/6 probability of success

X~BIN(n=20 ,p=1/6)

```
dbinom(x, size, prob)
pbinom(x, size, prob)
qbinom(p, size, prob)
rbinom(n, size, prob)
```

Discrete Probability Distributions

The Poisson distribution:

```
> #P(X=1)
>
> dpois(1,2)
[1] 0.2706706
>
> #P(x<=4)
>
> ppois(4,2)
[1] 0.947347
>
> #random sample
>
> rpois(3,2)
[1] 1 0 3
>
> #invers
>
> qpois(0.432,2)
[1] 2
>
```

X is Poisson Distribution with □=2
X~Pois(2)

```
dpois(x, lambda)
ppois(q, lambda)
qpois(p, lambda)
rpois(n, lambda)
```

Continuous Probability Distributions

The Normal distribution:

```
> dnorm(5,3,4)
[1] 0.08801633
> pnorm(1.69,3,4)
[1] 0.3716449
> qnorm(0.3716,3,4)
[1] 1.689525
> rnorm(2,3,4)
[1] 0.9303246 8.6888187
>
```

X is Normal Distribution with μ=3 and □=4 ,
X~N(3,4)

```
dnorm(x, mean, sd)
pnorm(x, mean, sd)
qnorm(p, mean, sd)
rnorm(n, mean, sd)
```

Continuous Probability Distributions

The Exponential distribution:

```
> #P(X=2)
>
> dexp(2,2)
[1] 0.03663128
>
> #P(X<=4)
>
> pexp(4,2)
[1] 0.9996645
>
> #random sample
> rexp(4,2)
[1] 0.1368625 0.1511670 0.2151919 0.5941433
>
> #invers
> qexp(0.42,2)
[1] 0.2723636
```

X is Exponential Distribution with
λ=2
X~Exp(2)

```
dexp(x,lambda)
pexp(q,lambda)
qexp(p,lambda)
rexp(n,lambda)
```

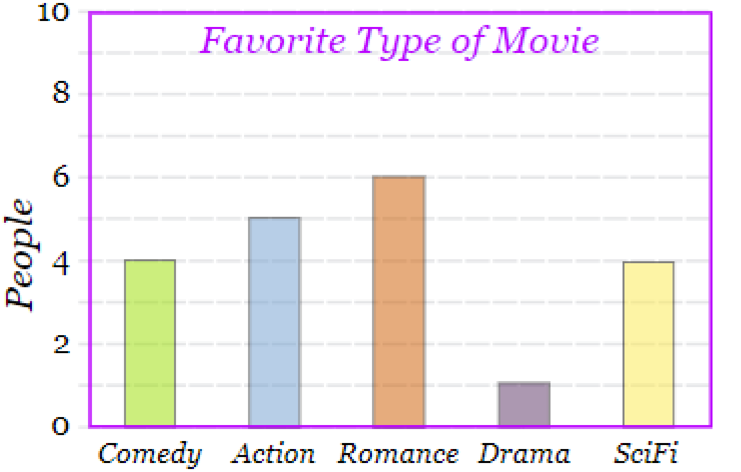
Basic Graphics

1- Bar Graph

is a graphical display of data using bars of different heights.

EXAMPLE:

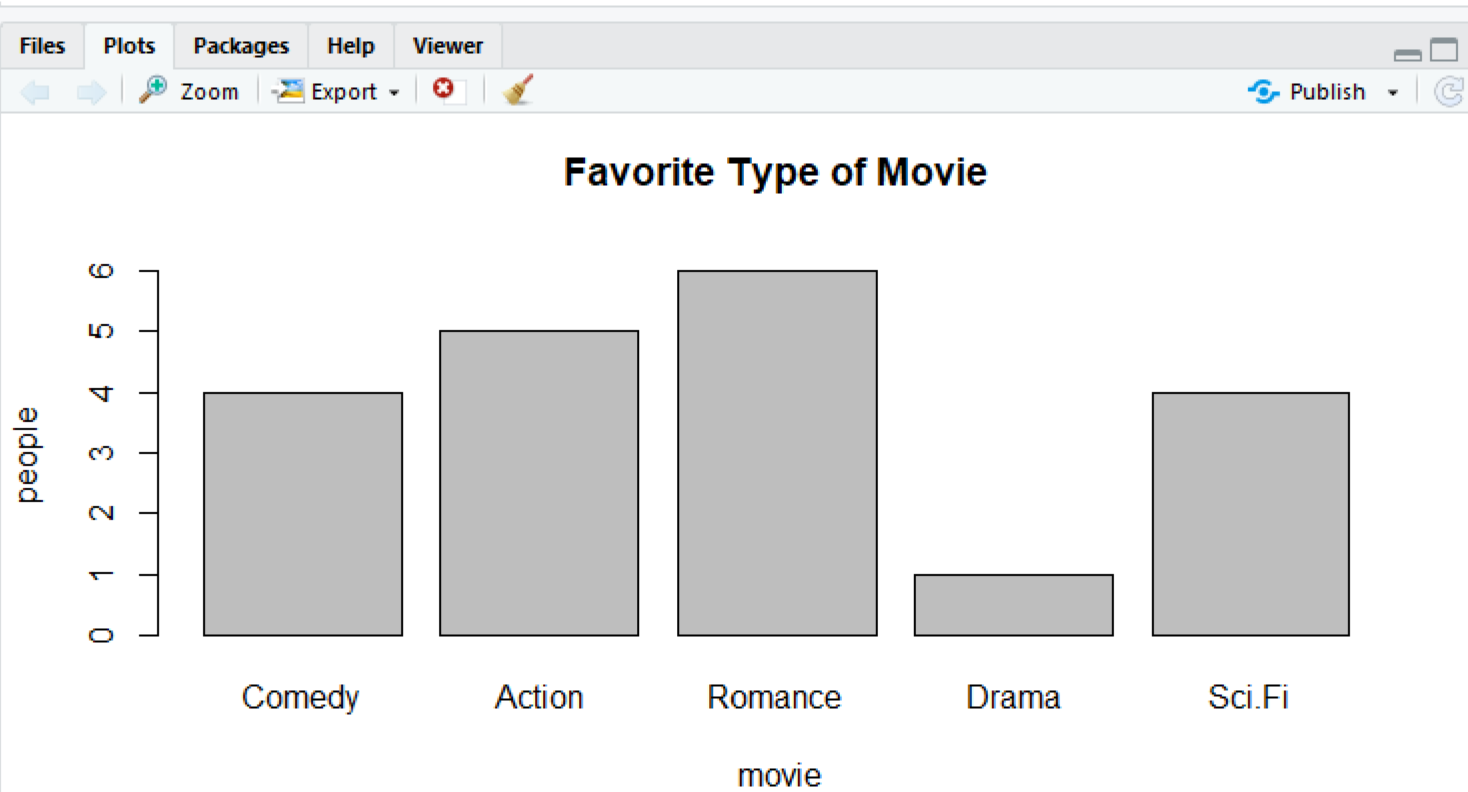
Favorite Type of Movie				
Comedy	**Action**	**Romance**	**Drama**	**Science fiction**
4	**5**	**6**	**1**	**4**



```
barplot(H , xlab = , ylab= , main = ,names.arg  )
```

- **H** is a vector containing numeric values used in bar chart.
- **xlab** is the label for x axis.
- **ylab** is the label for y axis.
- **main** is the title of the bar chart.
- **names.arg** is a vector of names appearing under each bar.

```
H<-c(4,5,6,1,4)

barplot(H,xlab=" movie", ylab="people",main="Favorite Type of Movie
",names.arg =c("Comedy","Action","Romance","Drama","Sci.Fi") )

```

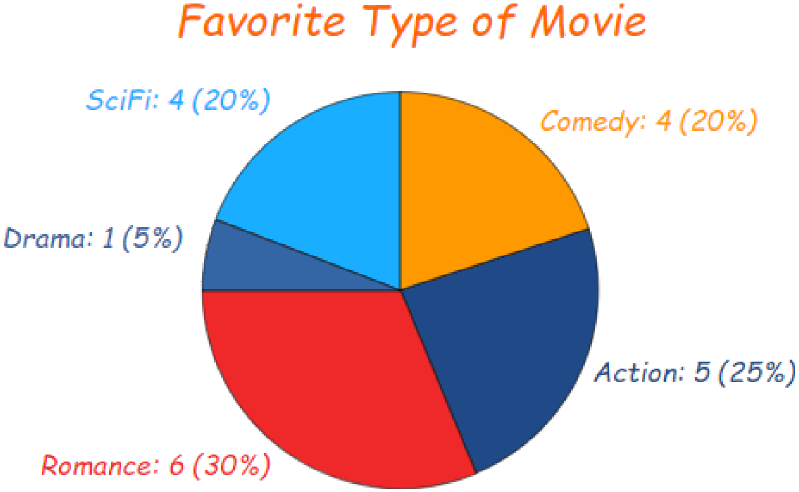


2- pie Graph

a special chart that uses "pie slices" to show relative sizes of data.

EXAMPLE:

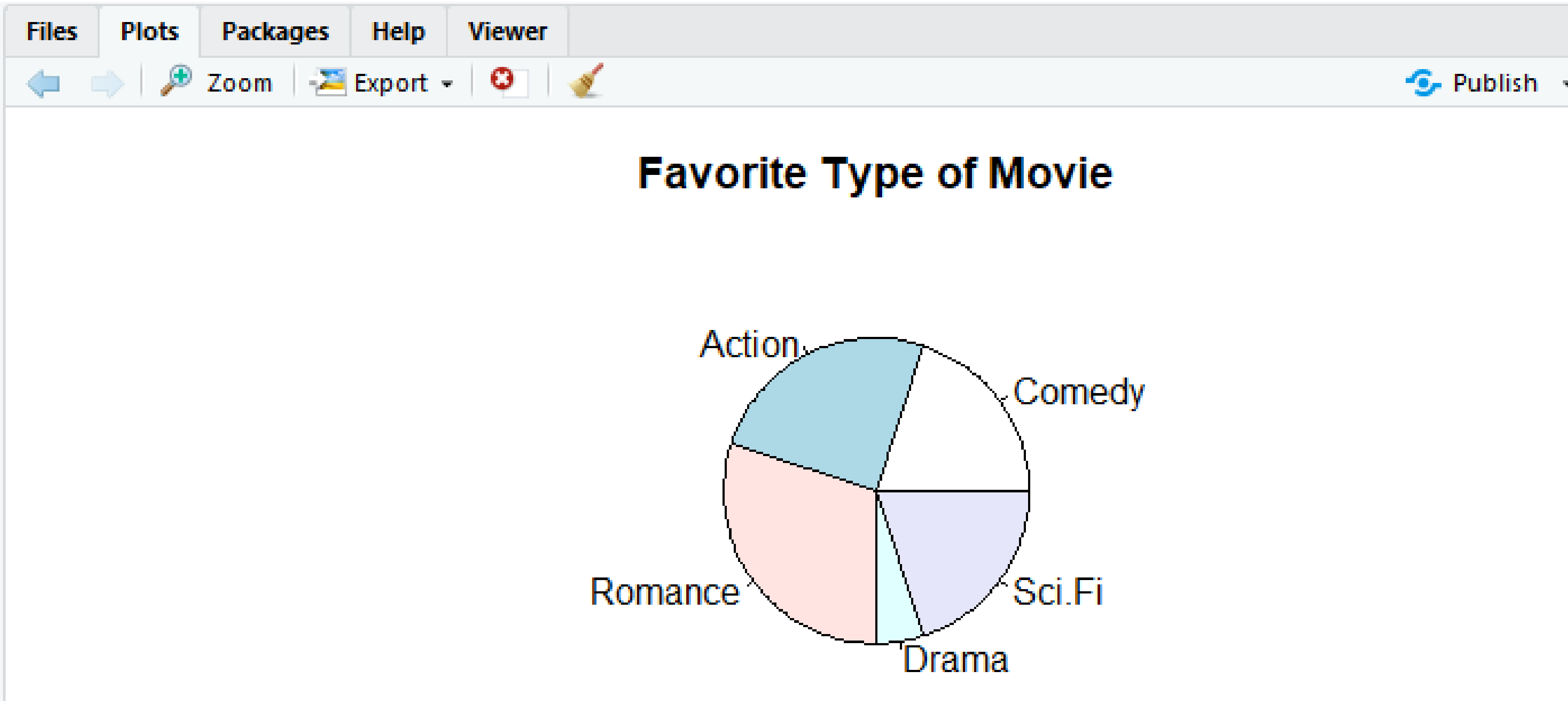
Favorite Type of Movie				
Comedy	**Action**	**Romance**	**Drama**	**Science fiction**
4	**5**	**6**	**1**	**4**



```
pie(x ,main= ,labels= )
```

- **x** is a vector containing the numeric values used in the pie chart.
- **main** indicates the title of the chart.
- **labels** is used to give description to the slices.

```r
x<-c(4,5,6,1,4)

pie(x,main="Favorite Type of Movie
          ",labels =c("Comedy","Action","Romance","Drama","Sci.Fi"))
```

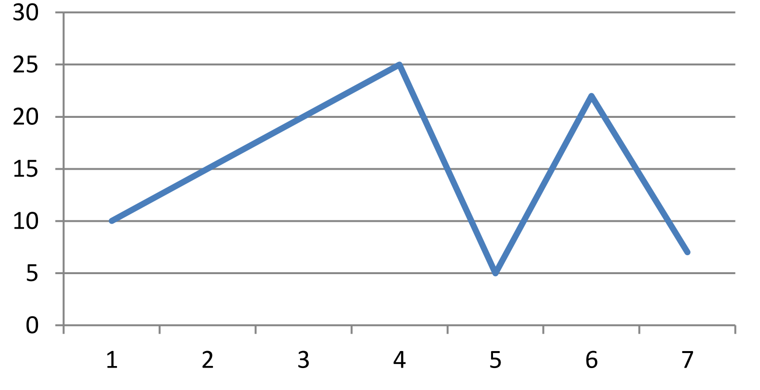


3- Line Graph:

a graph that shows information that is connected in some way (such as change over time)

EXAMPLE:

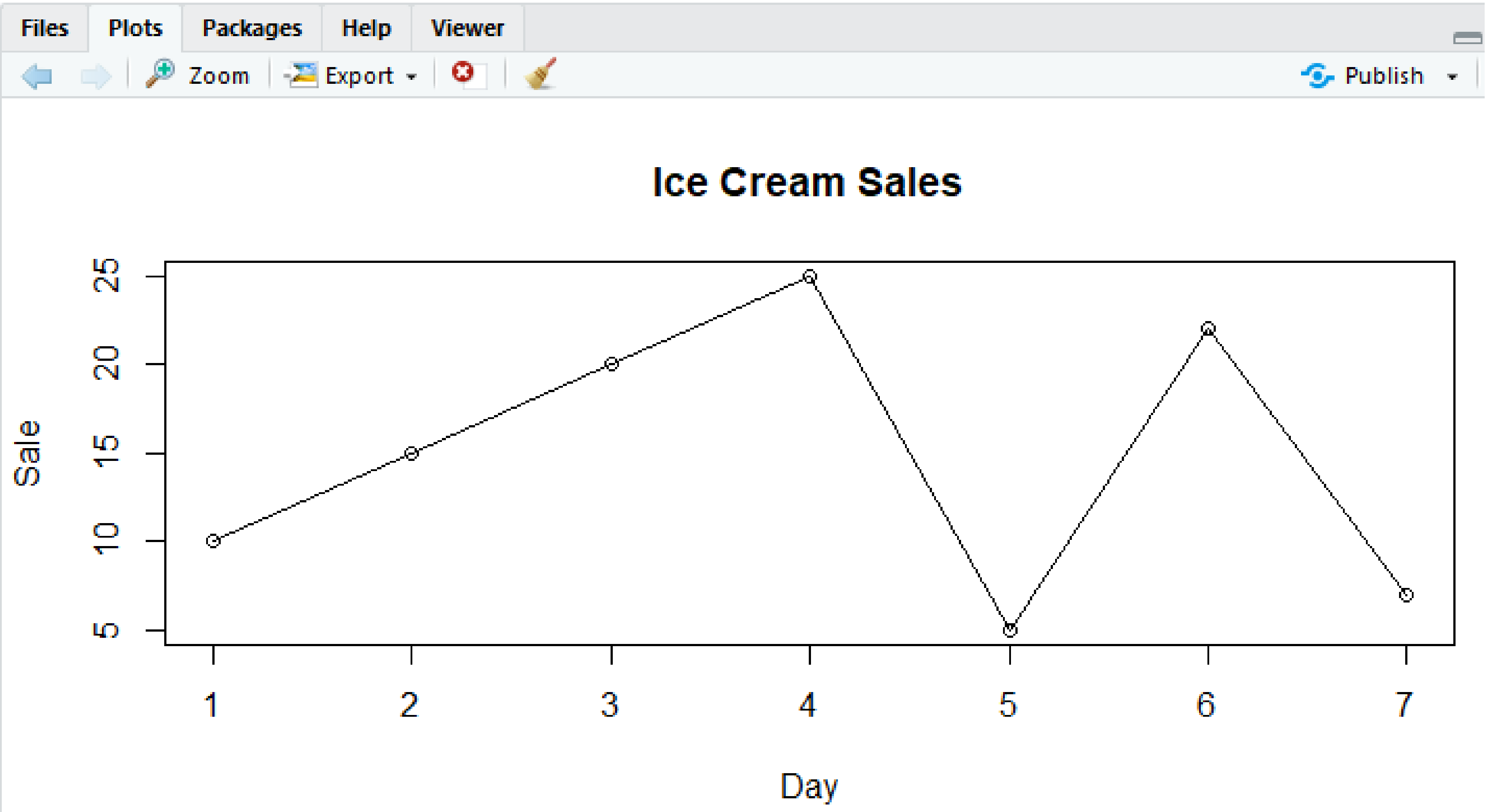
Ice Cream Sales						
Mon	Tue	Wed	Thu	Fri	Sat	Sun
10	15	20	25	5	22	7



```
plot(y ,type= ,xlab= ,ylab= ,main=  )
```

- **y** is a vector containing the numeric values.
- **type** takes the value "p" to draw only the points, "l" to draw only the lines and "o" to draw both points and lines.
- **xlab** is the label for x axis.
- **ylab** is the label for y axis.
- **main** is the Title of the chart.

```r
y<-c(10,15,20,25,5,22,7)
plot(y,type = "o",xlab="Day" ,ylab="Sale"
     ,main="Ice Cream Sales")
```

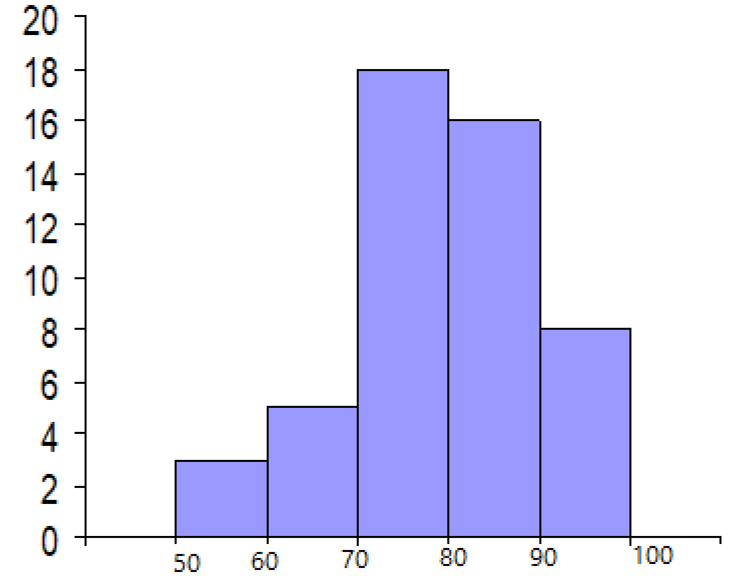


4- Histogram Graph

graphical display of data using bars of different heights.

EXAMPLE:

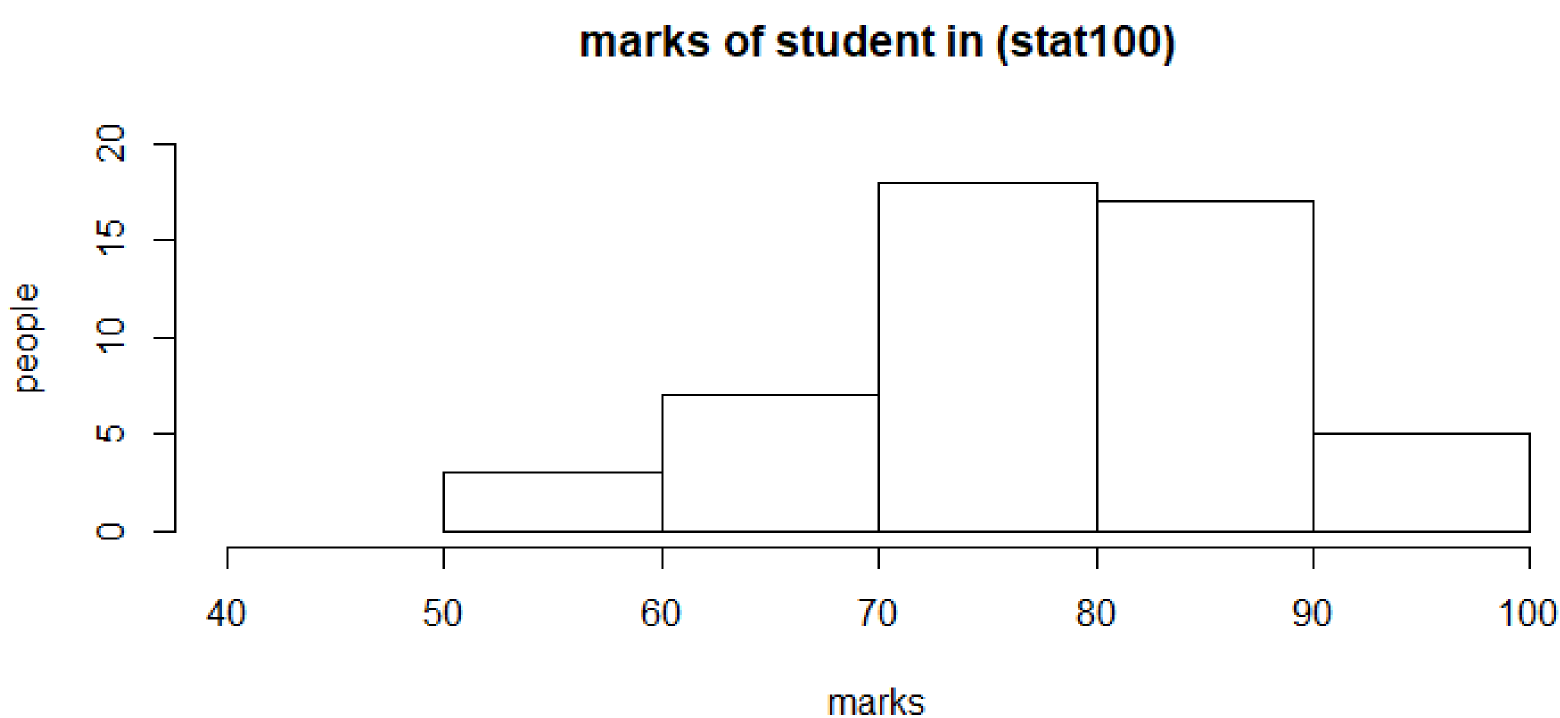
51	95	70	74	73	90	71	74	90	67
91	72	83	89	50	80	72	84	85	69
62	82	87	76	91	76	87	75	78	79
71	96	81	88	64	82	73	57	86	70
80	81	75	85	74	90	83	66	77	91



```
hist(v ,main= ,xlab= ,ylab= ,xlim= ,ylim= , breaks = )
```

- **v** is a vector containing numeric values used in histogram.
- **main** indicates title of the chart.
- **xlab** is used to give description of x-axis.
- **xlim** is used to specify the range of values on the x-axis.
- **ylim** is used to specify the range of values on the y-axis.
- **breaks** is used to mention the width of each bar.

```r
xx<-c(51,95,70,74,73,90,71,74,90,67,91,72,83,89,50,80,72,84,85,69,
      62,82,87,76,91,76,87,75,78,79,71,96,81,88,64,82,73,57,86,70,
      80,81,75,85,74,90,83,66,77,91)

xx
hist(xx,main="marks of student in (stat100)",xlab="marks",
     ylab="people", xlim=c(40,100),ylim=c(0,20),breaks =5)
```

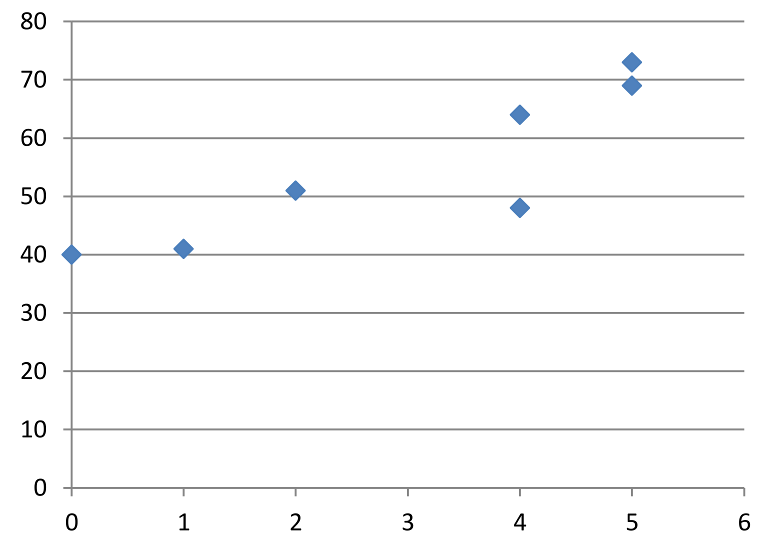


5- Scatter plot

Each point represents the values of two variables. One variable is chosen in the horizontal axis and another in the vertical axis.

EXAMPLE:

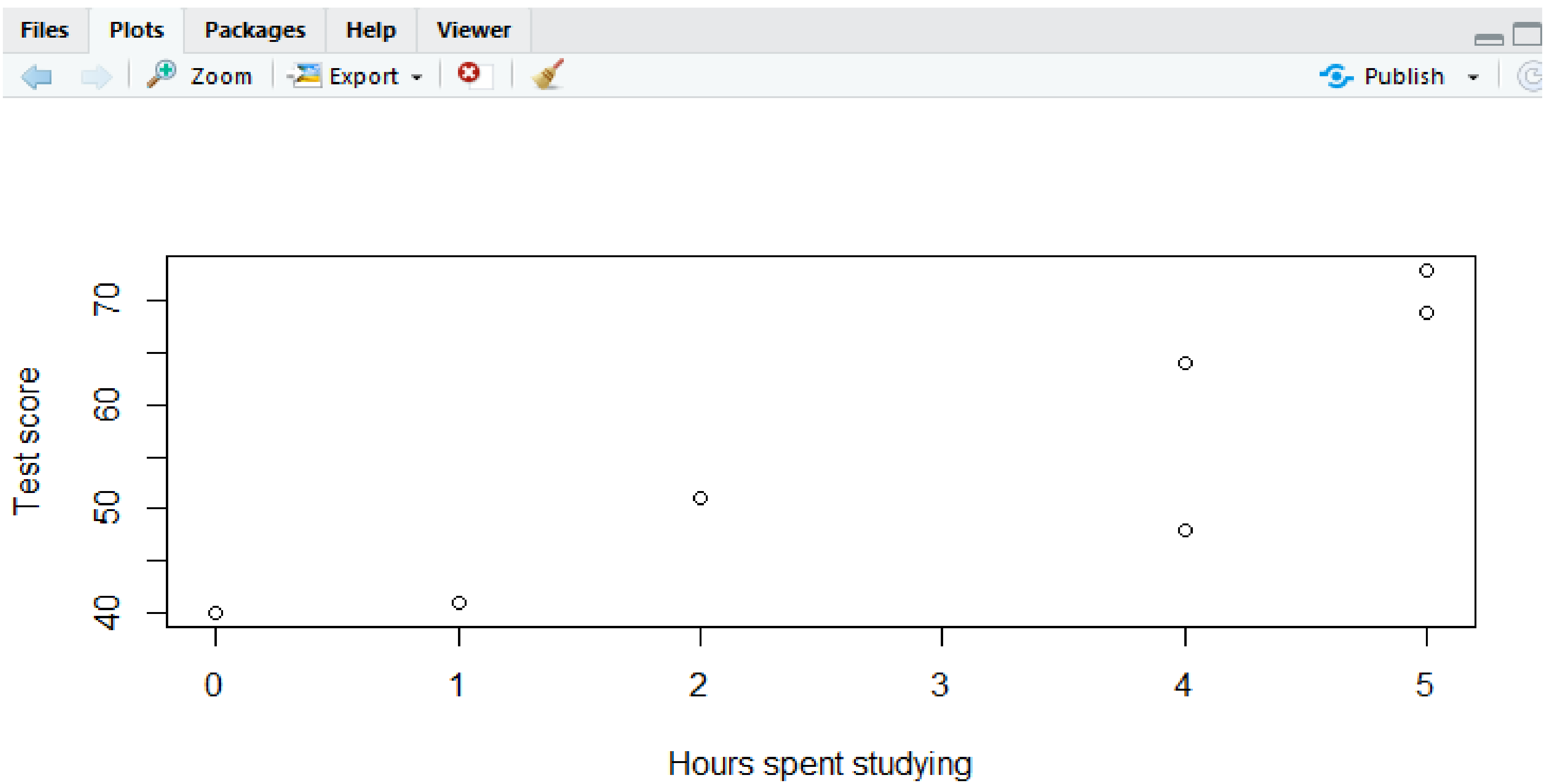
Hours spent studying, *x*	Test score, *y*
0	40
1	41
2	51
4	48
4	64
5	69
5	73



```
plot(x , y , main= ,xlab=  ,ylab= )
```

- **x** is the data set whose values are the horizontal coordinates.
- **y** is the data set whose values are the vertical coordinates.
- **main** is the tile of the graph.
- **xlab** is the label in the horizontal axis.
- **ylab** is the label in the vertical axis.

```r

<-c(0,1,2,4,4,5,5)
y<-c(40,41,51,48,64,69,73)

plot(x,y,xlab="Hours spent studying",ylab="Test score")

```



Simple Linear Regression

- The results shown below were obtained in a small-scale experiment to study the relation between 0C of storage temperature (X) and number of weeks before flavor deterioration of a food product begins to occur (Y).

i	1	2	3	4	5
X_i	8	4	0	-4	-8
Y_i	7.8	9.8	10.2	11.0	11.7

Assume that first-order regression model ($Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$) is applicable.

the command:

```
X = matrix( c(8, 4, 0, -4,  -8),
nrow=5,
ncol=1,
byrow = TRUE)

Y = matrix( c(7.8,9.0,10.2,11.0,11.7),
nrow=5,
ncol=1,
byrow = TRUE)
```

scatter plot

the command:

```
plot(X,Y)
```

correlation

The command:

cor(X,Y)

linear regression model

The command:

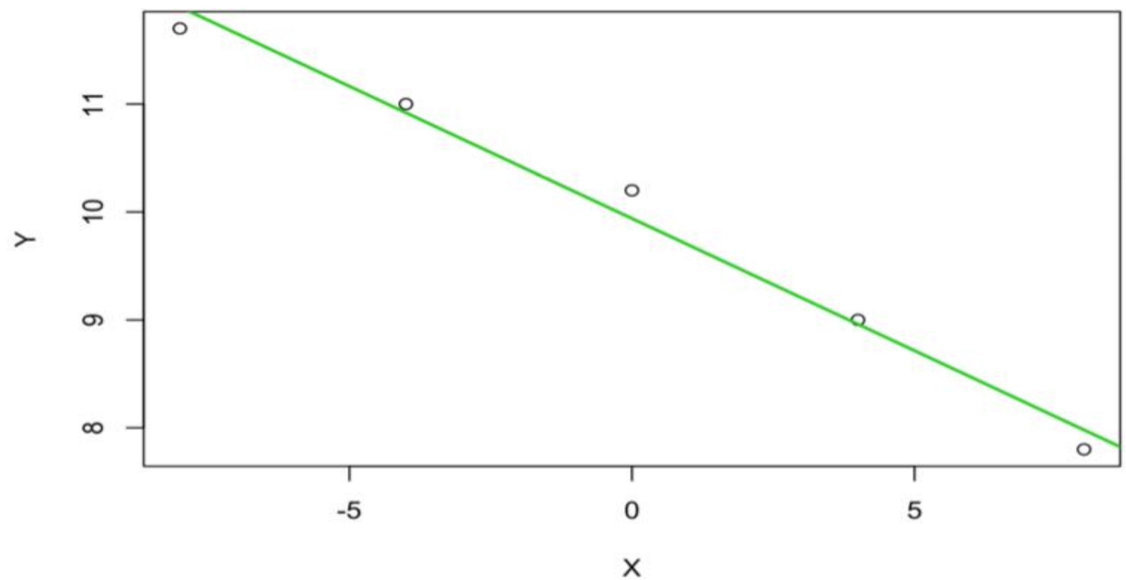
```
model<-lm(Y~X)

summary(model)

anova(model)
```

```r
model<-lm(Y~X)
summary(model)
```

```
##
## Call:
## lm(formula = Y ~ X)
##
## Residuals:
##     1     2     3     4     5
## -0.18  0.04  0.26  0.08 -0.20
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  9.94000    0.09933  100.07  2.2e-06 ***
## X           -0.24500    0.01756  -13.95 0.000797 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2221 on 3 degrees of freedom
## Multiple R-squared:  0.9848, Adjusted R-squared:  0.9798
## F-statistic: 194.7 on 1 and 3 DF,  p-value: 0.0007971
```

Add a regression line to the plot and change color and line width

```
plot(X,Y)
abline(model)
abline(model, col=3,lwd=2)
```



Import data from excel to R

- **1- Open** the Excel file containing your data: **select** and **copy the data** (ctrl + c)
- 2- Type the R code below to import the copied data from the **clipboard** into R and store the data in a data frame (Table):

D	C	B	A	
salary	experanice	edulevel	gender	1
500	1	1	1	2
450	2	1	2	3
440	1	1	1	4
500	3	1	2	5
570	2	2	1	6
550	3	2	2	7
490	2	2	2	8
540	3	2	2	9
600	2	2	1	10
650	3	2	1	11

```
> Table<-read.delim('clipboard')
> Table
   gender edulevel experanice salary
1       1        1          1    500
2       2        1          2    450
3       1        1          1    440
4       2        1          3    500
5       1        2          2    570
6       2        2          3    550
7       2        2          2    490
8       2        2          3    540
9       1        2          2    600
10      1        2          3    650
>
```

Hypothesis testing

A statistical method that uses sample data to evaluate a hypothesis a bout a population parameter

Testing about population mean

Goal	Test
Compare one group to hypothetical value	One sample t-test
Compare two paired group	Paired t-test
Compare two unpaired group	Two sample t-test
Compare three or more sample	ANOVA

One sample t test

Use the 1-sample t-test to estimate the mean of a population and compare it to a target or reference value when you do not know the standard deviation of the population, assuming the population to be approximately normal. Using this test, you can:

- Determine whether the mean of a group differs from a specified value.
- Calculate a range of values that is likely to include the population mean.

- Example: Six students get scores of 62, 92, 75, 68, 83, and 95. Can the professor have 90 percent confidence that the mean score for the class on the test would be above 70

```
> t.test(x,mu=70,alternative = "two.sided",conf.level = 0.90)

        One Sample t-test

data:  x
t = 1.7053, df = 5, p-value = 0.1489
alternative hypothesis: true mean is not equal to 70
90 percent confidence interval:
 68.33507 89.99827
sample estimates:
mean of x
 79.16667
```

For CI
Choose
“two.sided”
for alternative

Two sample t-test

- Use the 2-sample t-test to two compare between two population means, when the variances are unknowns assuming the both population are independent and approximately normal

There are two cases

1- population variances are unknown but equal.

2- population variances are unknown but unequal.

- Example: Below you can find the study hours of 6 female students and 5 male students.

Female	26	25	43	34	18	52
Male	23	30	18	25	28	

Is there a difference in average number of a study hours between male and female students.

```
> Female<-c(26,25,43,34,18,52)
> Male<-c(23,30,18,25,28)
> t.test(Female,Male,alternative = "two.sided",mu=0,paired =FALSE,var.equal
 = FALSE, conf.level = 0.95)

        Welch Two Sample t-test

data:  Female and Male
t = 1.4726, df = 6.5433, p-value = 0.1873
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -5.155702 21.555702
sample estimates:
mean of x mean of y
     33.0      24.8
```

Paired Sample t Test

- In paired sample hypothesis testing, a sample from the population is chosen and two measurements for each element in the sample are taken. Each set of measurements is considered a sample. the two samples are not independent of one another. Paired samples are also called matched samples or repeated measures.
- Use the Paired-sample t-test to compare between the means of paired observations taken from the same population. This can be very useful to see the effectiveness of a treatment on some objects.

- Example:

A clinic provides a program to help their clients lose weight and asks a consumer agency to investigate the effectiveness of the program. The agency takes a sample of 15 people, weighing each person in the sample before the program begins and 3 months later to produce the table below

Determine whether the program is effective?

Weight before	**210**	**205**	**193**	**182**	**259**	**239**	**164**	**197**	**222**	**211**	**187**	**175**	**186**	**243**	**246**
Weight after	197	195	191	174	236	226	157	196	201	196	181	164	181	229	231

```
> x1<-c(210,205,193,182,259,239,164,197,222,211,187,175,186,243,246)
> y1<-c(197,195,191,174,236,226,157,196,201,196,181,164,181,229,231)
> t.test(x1,y1,alternative = "greater",mu=0,paired =TRUE, conf.level = 0.95
)

        Paired t-test

data:  x1 and y1
t = 6.6897, df = 14, p-value = 5.138e-06
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 8.05473      Inf
sample estimates:
mean of the differences
               10.93333
```

Analysis of variance (ANOVA)

- The one-way analysis of variance (ANOVA) is used to determine whether there are any statistically significant differences between the means of three or more independent (unrelated) groups.
- Treatment population are normally distributed equal variances

- Example: Suppose the National Transportation Safety Board (NTSB) wants to examine the safety of compact cars, midsize cars, and full-size cars. It collects a sample of three for each of the treatments (cars types). Using the hypothetical data provided below, test whether the mean pressure applied to the driver’s head during a crash test is equal for each types of car. Use $\alpha = 5\%$.

Compact cars	Midsize cars	Full size cars
643	469	484
655	427	456
702	525	402

```
> z1<-c(643,655,702)
> z2<-c(469,427,525)
> z3<-c(484,456,402)
> y=c(z1,z2,z3)
> n=rep(3,3)
> group=rep(1:3,n)
> data=data.frame(y=y,group=factor(group))
> fit=lm(y~group,data)
> anova(fit)
Analysis of Variance Table

Response: y
          Df Sum Sq Mean Sq F value   Pr(>F)
group      2  86050   43025  25.175 0.001207 **
Residuals  6  10254    1709
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> 
```