R Markdown

It allows you to embed R code and R output directly into documents, pdf, HTML ,word etc…

<ls> command：show everything stored in workspace memory

<rm> command：remove object from workspace

<“ “> will be treated as character , not number

by showing <+> sign , R tells you that your command is incomplete

by showing <#> sign , R will ignore everything follow below

Vectors

x1 ← c( 1,3,5 )

x2 ← c( “male” , “female” )

x3 ← 2:7

seq( from= , to= , by= )

rep( , times= )

rep(seq( from= , to= , by= ) , times= )

x1… may be calculated +, -, \*, /

if two vectors have same length , we may calculate +, -, \*, /

y ← c( 1,3,5,7,9 )

y[3] > 5

y[-3] > 1 3 7 9

y[1:3] > 1 3 5

y[y<6] > 1 3 5

matrix(c( 1,2,3,4,5,6,7,8,9) , nrow=3 , byrow=true )

1 2 3

4 5 6

7 8 9

byrow=false , elements will be entered in column-wise

mat ← matrix(c( 1,2,3,4,5,6,7,8,9) , nrow=3 , byrow=true )

mat [1,2] > 2 ( row,column )

mat [c(1,3) , 2 ] > 2 8 ( row1&row3 , column2 )

mat [ 2, ] > 4 5 6 ( all row 2 )

mat [ 2,3 ] ← “apple” ( replace 2,3 with “apple” , and transform all elements into characters )

length(mat) ( amounts of elements in matrix )

mat… may be calculated +, -, \*, /

data.frame ( nickname=c( “Tom” , “Jack” ) , height=170:180 )

nickname height

1 Tom 170

2 Jack 180

\*\* data frame 的每個行（column）可以儲存不同變數類型的資料，甚至非狀巢結構的列表亦可。

data1 ← read.csv( file,choose(), header=T )

T = the first row of our dataset are variables or headers

data2 ← read.table( file,choose(), header=T , sep= ” , “ )

sep= ” , “ ( csv file )

sep= ” \t “ ( tab-delimited file )

write.table( DataToExport , file=”...url.../ExportedFileName.csv” , sep= ”, “ )

row.names=false ( no column header )

data <- read.table ( file=”path”, header=TRUE , sep=”\t” )

header = first row

sep =” \t “ >> tab delimited

sep = “, ” >> comma separated

sep = “ “ >> blank

data <- read.table( file.choose() , header=TRUE , sep=”\t” )

rm(data)

dim(data)

head(data) >>> first six rows

tail(data) >>> last six rows

names(data) >>> the headers

dim(data) >>> #row #column

attach(data) >>> put memory in every column

data$Age

$ helps to extract data

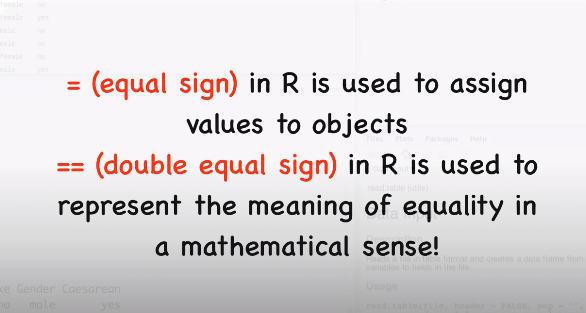
levels(factor)

> “male” “female”

summary(data)

> 敘述統計 ( if data is numeric )

> frequency ( if data is factor )



mean(Age[Gender==”male”])

Maleover15 <- data[ Gender==”male” & Age>15 , ]

as numeric => 0 or 1

：as numeric ( condition )

* 0 > false
* 1 > true

equal sign > “ == ”

bind a new column to data > cbind

Newdata <- cbind ( data , newcolumn )