**國企三 陳湘雲 B08704024**

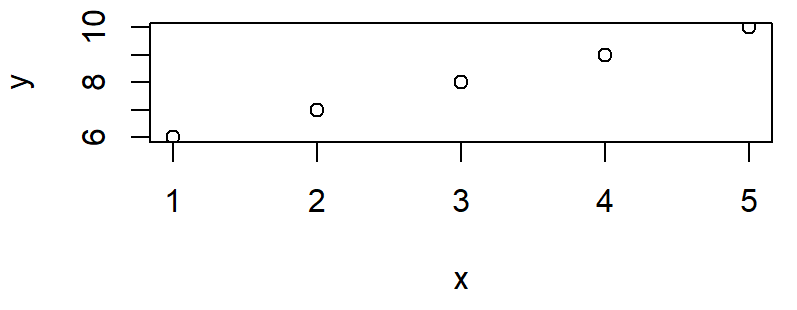
**Week 1**

**Video1.1-1.3**

> x <- 1:5

> y <- 6:10

> plot(x,y)



\*Plots裡的Export可以把圖表輸出並調整大小

R Studio 的好處

1. 可顯示結果在同頁面，如上面那個圖
2. 圖可存各種檔案類型並調整點距
3. 能匯入其他已完成檔案
4. 可寫草稿再丟進程式並儲存

[1] "x" "y"

> z=3:4

> sum(x,y,z)

[1] 62

> x\*y

[1] 6 14 24 36 50

> print(x)

[1] 1 2 3 4 5

> rm(y)-------移除

> ls()-----------看已建立的未知數

[1] "x"

> z=x+3

> z

[1] 4 5 6 7 8

> r=9

> sqrt(r)-------平方根

[1] 3

> exp(r)----------er

[1] 8103.084

> getwd()-------------存檔的位置

[1] "C:/Users/w4566/Desktop/台大/大三/數量方法與分析決策"

save

**week 2**

**Video1.4-1.6**

> x1=c(1,3,5,7,9)

> x1

[1] 1 3 5 7 9

> gender=c("male","female")

> gender

[1] "male" "female"

> 2:7

[1] 2 3 4 5 6 7

> seq(from=1,to=7,by=2)

[1] 1 3 5 7

> seq(1,7,2)

[1] 1 3 5 7

> rep(x,times=5)

[1] 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5

> y[c(1,5)]

[1] 6 10

> y[c(y<8)]

[1] 6 7

> matrix(c(1:9),nrow=3,byrow=TRUE)

[,1] [,2] [,3]

[1,] 1 2 3

[2,] 4 5 6

[3,] 7 8 9

> matrix(c(1:10),nrow=5,byrow=FALSE)

[,1] [,2]

[1,] 1 6

[2,] 2 7

[3,] 3 8

[4,] 4 9

[5,] 5 10

> mat=matrix(c(1:10),nrow=5,byrow=FALSE)

> mat[1,2]

[1] 6

> mat[c(1,3),2]

[1] 6 8

> mat[2,]

[1] 2 7

> data1=read.table(file.choose(),header=T,sep=",")

> data1

X Stock1 Stock2 Stock3 Stock4

1 Day1 185.74 1.47 1605 95.05

2 Day2 184.26 1.56 1580 97.49

3 Day3 162.21 1.39 1490 88.57

4 Day4 159.04 1.43 1520 85.55

5 Day5 164.87 1.42 1550 92.04

6 Day6 162.72 1.36 1525 91.70

7 Day7 157.89 NA 1495 89.88

8 Day8 159.49 1.43 1485 93.17

9 Day9 150.22 1.57 1470 90.12

10 Day10 151.02 1.54 1510 92.14

**Week3**

**Video1.7-1.9**

> dim(matrix(c(1,2,3,4,5,6,7,8,9),nrow=3,byrow=T))--------dimension 幾行幾列

[1] 3 3

> matrixOne <- matrix(1:100, nrow=10, ncol=10, byrow=FALSE)

> head(matrixOne)---------=matrixOne[1:6,]

[,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]

[1,] 1 11 21 31 41 51 61 71 81 91

[2,] 2 12 22 32 42 52 62 72 82 92

[3,] 3 13 23 33 43 53 63 73 83 93

[4,] 4 14 24 34 44 54 64 74 84 94

[5,] 5 15 25 35 45 55 65 75 85 95

[6,] 6 16 26 36 46 56 66 76 86 96

> tail(matrixOne)

[,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]

[5,] 5 15 25 35 45 55 65 75 85 95

[6,] 6 16 26 36 46 56 66 76 86 96

[7,] 7 17 27 37 47 57 67 77 87 97

[8,] 8 18 28 38 48 58 68 78 88 98

[9,] 9 19 29 39 49 59 69 79 89 99

[10,] 10 20 30 40 50 60 70 80 90 100

> attach(data1)

> mean(Stock1)

[1] 163.746

> names(data1)

[1] "X" "Stock1" "Stock2" "Stock3" "Stock4"

> summary(data1)

X Stock1 Stock2

Length:10 Min. :150.2 Min. :1.360

Class :character 1st Qu.:158.2 1st Qu.:1.420

Mode :character Median :160.8 Median :1.430

Mean :163.7 Mean :1.463

3rd Qu.:164.3 3rd Qu.:1.540

Max. :185.7 Max. :1.570

NA's :1

Stock3 Stock4

Min. :1470 Min. :85.55

1st Qu.:1491 1st Qu.:89.94

Median :1515 Median :91.87

Mean :1523 Mean :91.57

3rd Qu.:1544 3rd Qu.:92.91

Max. :1605 Max. :97.49

> mean(aa$weight)-----------$表示你要表裡面的什麼 mean就是平均

[1] 61

attach(aa)--------只是舉例，如果有從外匯進來的表格，要讓R認識裡面的資訊就用attach

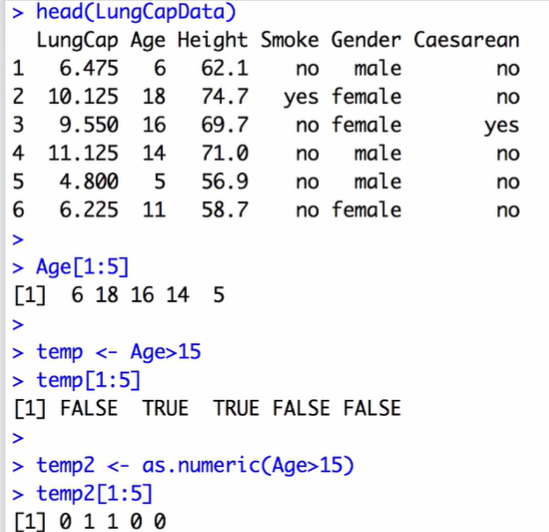
detach(aa)----------刪掉匯進來的

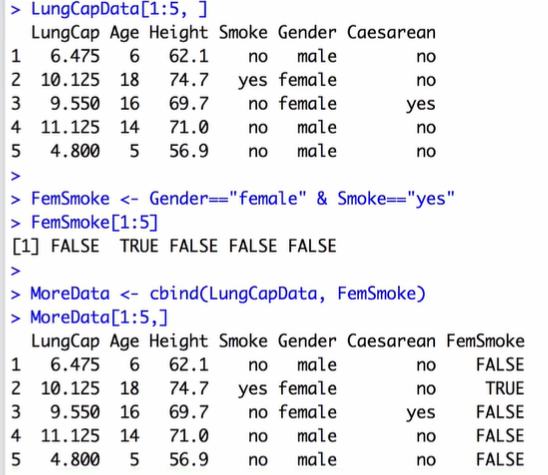
> mean(Age[Gender=="female"])

[1] 12.44972

**Week 4**

**Video** 1.10-1.12

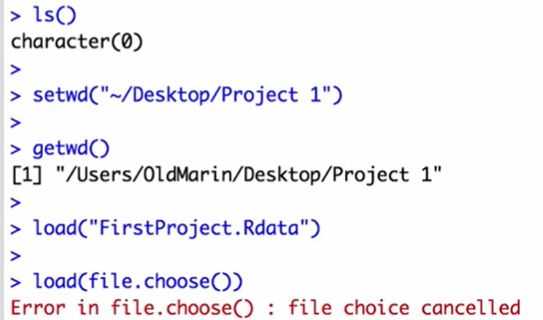




> rm(list=ls())-------------environment被清空

(開了一個空白R studio然後下載剛剛的environment)



**Class**

Youtube影片要找五個指令 要露臉

**Week 5**

**Video** 1.13-1.15

install.packages("epiR")

library(epiR)

help(packages=epiR)

rm.packages(epiR)

**Week 6**

**Video** 1.16

> getwd()

[1] "C:/Users/w4566/Desktop/台大/大三/數量方法與分析決策"

> StockData <- read.table(file="C:/Users/w4566/Desktop/台大/大三/數量方法與分析決策/StockExample.csv",sep=",", header=T, row.names=1)

> View(StockData)

> View(StockData)

> StockData

Stock1 Stock2 Stock3 Stock4

Day1 185.74 1.47 1605 95.05

Day2 184.26 1.56 1580 97.49

Day3 162.21 1.39 1490 88.57

Day4 159.04 1.43 1520 85.55

Day5 164.87 1.42 1550 92.04

Day6 162.72 1.36 1525 91.70

Day7 157.89 NA 1495 89.88

Day8 159.49 1.43 1485 93.17

Day9 150.22 1.57 1470 90.12

Day10 151.02 1.54 1510 92.14

> apply(X=StockData, MARGIN=2, FUN=mean)

Stock1 Stock2 Stock3 Stock4

163.746 NA 1523.000 91.571

> apply(X=StockData, MARGIN=2, FUN=mean, na.rm=TRUE)

Stock1 Stock2 Stock3 Stock4

163.746000 1.463333 1523.000000 91.571000

> AVG <- apply(X=StockData, MARGIN=2, FUN=mean, na.rm=TRUE)

> AVG

Stock1 Stock2 Stock3 Stock4

163.746000 1.463333 1523.000000 91.571000

> apply(StockData, 2, mean, na.rm=TRUE)

Stock1 Stock2 Stock3 Stock4

163.746000 1.463333 1523.000000 91.571000

> colMeans(StockData, na.rm=TRUE)

Stock1 Stock2 Stock3 Stock4

163.746000 1.463333 1523.000000 91.571000

> # notice that we don't need to include "MARGIN", etc, as long

> # as we enter info in the specified order

> apply(StockData, 2, mean, na.rm=TRUE)

Stock1 Stock2 Stock3 Stock4

163.746000 1.463333 1523.000000 91.571000

> colMeans(StockData, na.rm=TRUE)

Stock1 Stock2 Stock3 Stock4

163.746000 1.463333 1523.000000 91.571000

> apply(X=StockData, MARGIN=2, FUN=max, na.rm=TRUE)

Stock1 Stock2 Stock3 Stock4

185.74 1.57 1605.00 97.49

> apply(X=StockData, MARGIN=2, FUN=quantile, probs=c(0.2, .80),na.rm=TRUE)

Stock1 Stock2 Stock3 Stock4

20% 156.516 1.408 1489 89.618

80% 168.748 1.548 1556 93.546

> apply(X=StockData, MARGIN=2, FUN=plot, type="l")

NULL

> apply(X=StockData, MARGIN=2, FUN=plot, type="l", main="stock", ylab="Price", xlab="Day")

NULL

> apply(X=StockData, MARGIN=1, FUN=sum, na.rm=TRUE)

Day1 Day2 Day3 Day4 Day5 Day6 Day7

1887.26 1863.31 1742.17 1766.02 1808.33 1780.78 1742.77

Day8 Day9 Day10

1739.09 1711.91 1754.70

> rowSums(StockData, na.rm=TRUE)

Day1 Day2 Day3 Day4 Day5 Day6 Day7

1887.26 1863.31 1742.17 1766.02 1808.33 1780.78 1742.77

Day8 Day9 Day10

1739.09 1711.91 1754.70

> plot(apply(X=StockData, MARGIN=1, FUN=sum, na.rm=TRUE), type="l",ylab="Total Market Value", xlab="Day", main="Market Trend")

> X=StockData, MARGIN=1, FUN=sum, na.rm=TRUE),

Error: unexpected ',' in "X=StockData,"

> pch=16, col="blue")

Error: unexpected ',' in " pch=16,"

> points(apply(X=StockData, MARGIN=1, FUN=sum, na.rm=TRUE), pch=16, col="blue")