大數據資料處理與分析基礎

大數據簡介、特徵、應用案例

主講人:
臺北城市科技大學資管系
林慶昌博士

2017.09.05

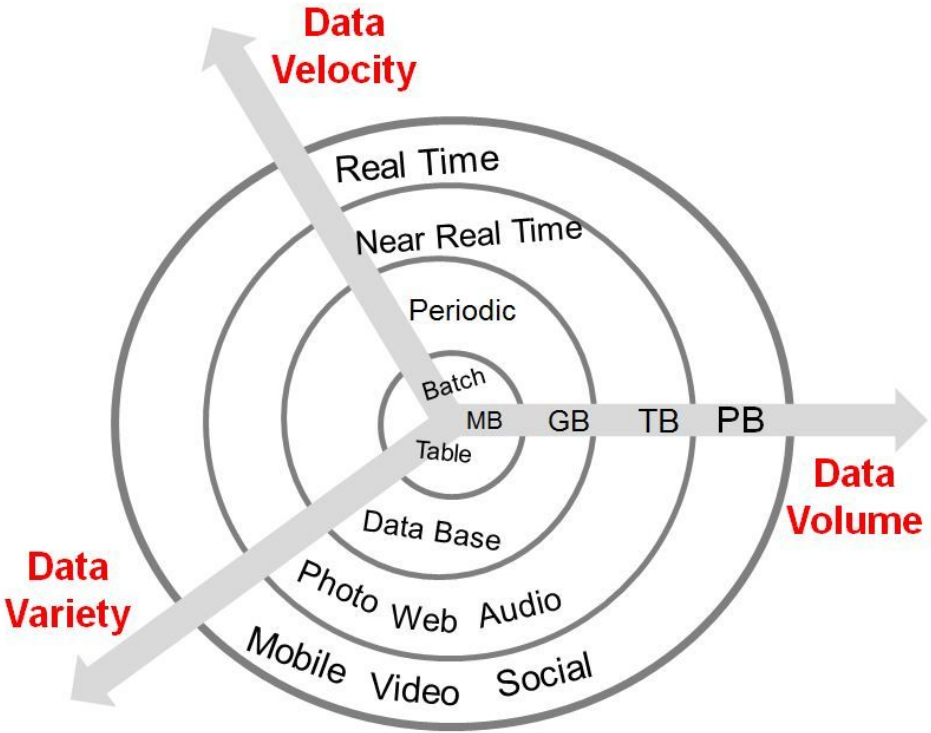
大數據簡介、特徵、應用案例

Big Data 大數據、巨量資料

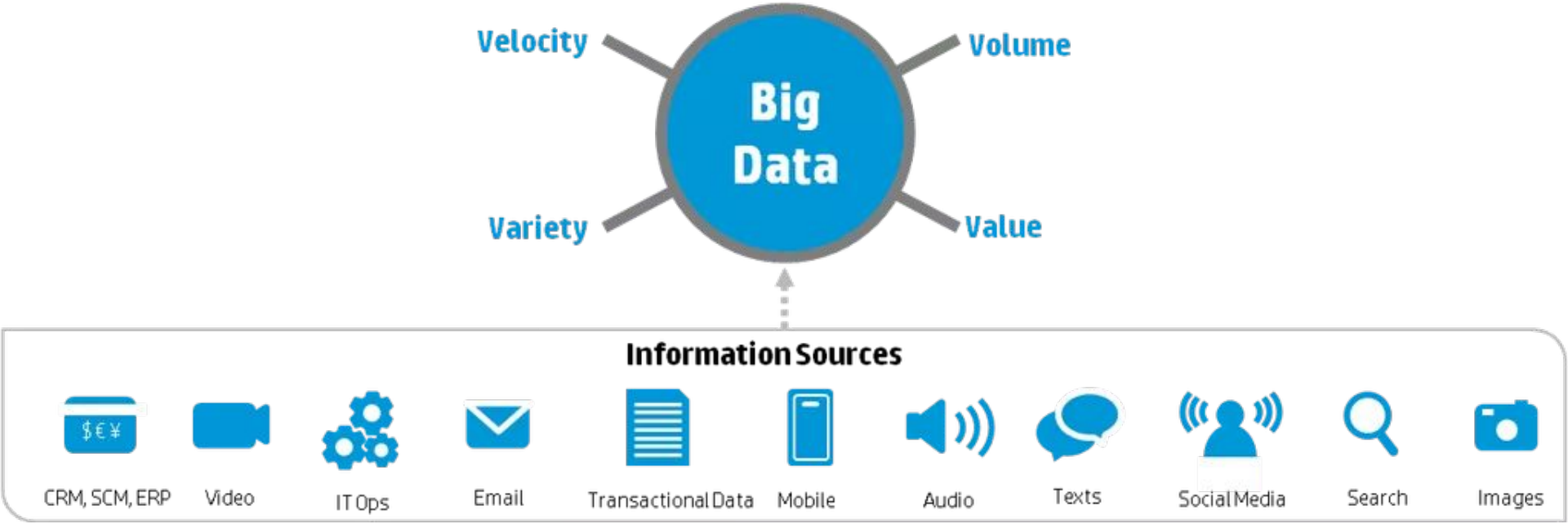
- 顧名思義，是指大量的資訊，當資料量龐大到資料庫系統無法在合理時間內進行儲存、運算、處理，分析成能解讀的資訊時，就稱為大數據
- Big data is data that exceeds the processing capacity of conventional database systems.
- 大數據就是規模非常、非常龐大的數位資訊，這些資料量巨大到無法藉人工和現有科技來儲存、傳送和分析，進而促使人們研發出更高階的資料儲存設備和科技

大數據簡介、特徵、應用案例

- Volumn
  - 資料量龐「大」
- Velocity
  - 變化飛「快」
- Variety
  - 種類繁「雜」
- Veracity
  - 真偽存「疑」



大數據簡介、特徵、應用案例



大數據簡介、特徵、應用案例

大數據特性

四字箴言：「大、快、雜、疑」

大數據資料量龐「大」(Volume)、變化飛「快」(Velocity)，種類繁「雜」(Variety)，以及真偽存「疑」(Veracity)

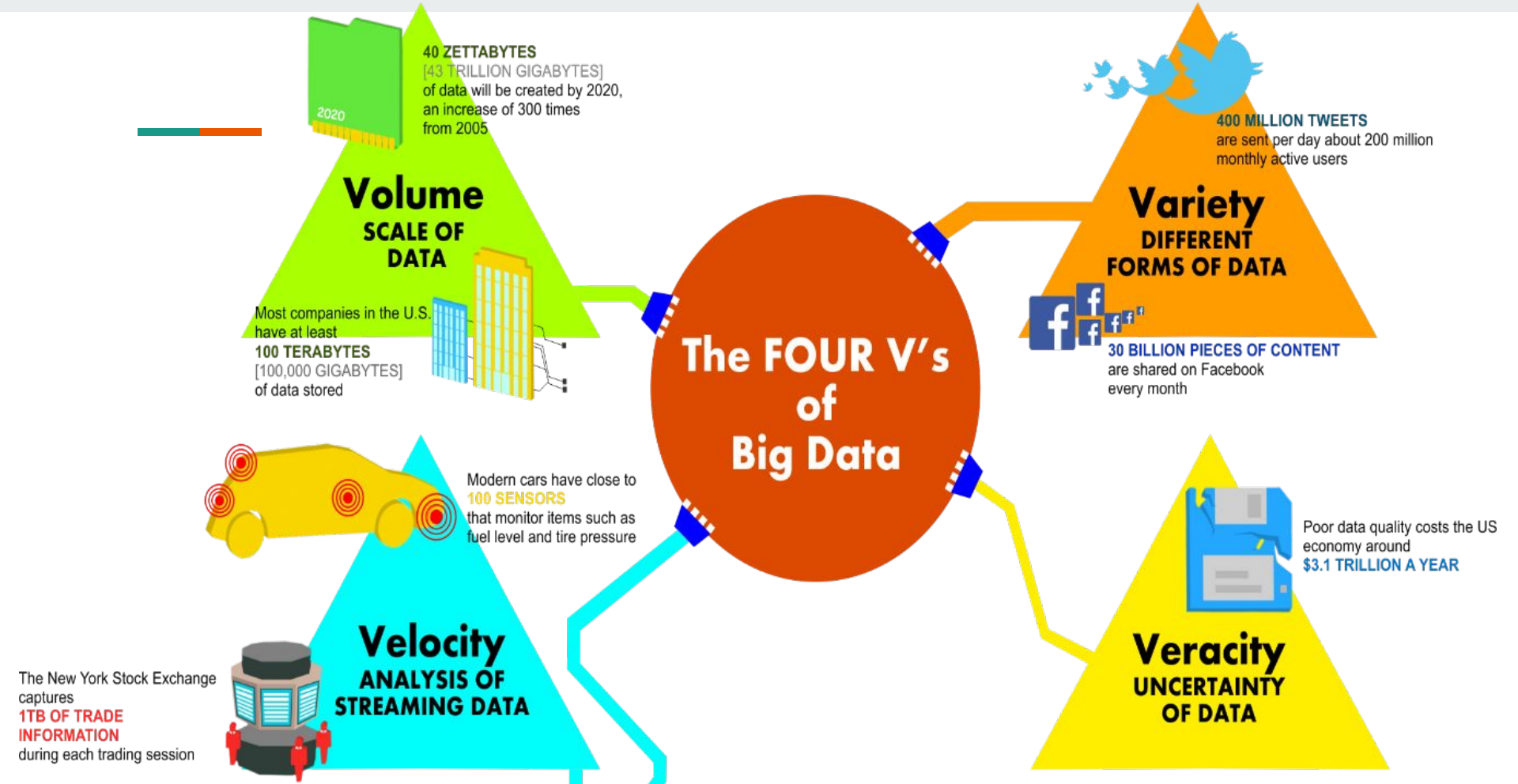
簡而言之，資料量**又大**、傳輸速度**又快**、內容**又雜**、真實性**又真偽難分**

[Big Data - Tim Smith (Youtube)]

大數據簡介、特徵、應用案例

FAQ:

- 大數據應用案例之電視媒體
- 大數據應用案例之社交網路
- 大數據應用案例之醫療行業
- 大數據應用案例之保險行業
- 大數據應用案例之能源行業
- 大數據應用案例之公路交通
- 大數據應用案例之零售業


2020
40 ZETTABYTES
[43 TRILLION GIGABYTES]
of data will be created by 2020,
an increase of 300 times
from 2005
Volume
SCALE OF DATA
Most companies in the U.S.
have at least
100 TERABYTES
[100,000 GIGABYTES]
of data stored
400 MILLION TWEETS
are sent per day about 200 million
monthly active users
Variety
DIFFERENT FORMS OF DATA
30 BILLION PIECES OF CONTENT
are shared on Facebook
every month
The FOUR V's of Big Data
Modern cars have close to
100 SENSORS
that monitor items such as
fuel level and tire pressure
Velocity
ANALYSIS OF STREAMING DATA
The New York Stock Exchange
captures
1TB OF TRADE
INFORMATION
during each trading session
Poor data quality costs the US
economy around
$3.1 TRILLION A YEAR
Veracity
UNCERTAINTY OF DATA

大數據簡介、特徵、應用案例

What is big data?

Volume - Scale of data

Variety - Different forms of data

Velocity - Analysis of streaming data

Veracity - Uncertainty of data

大數據簡介、特徵、應用案例

大數據分析，皆需考量四個特性：

Volume：累積的巨大的資料量

Variety：資料形式的多樣性，包括文字、影像、社群訊息、搜尋行為等等

Velocity：快速的傳輸速度

Veracity：為確保資料的真實性與正確性，分析過程相當重要

大數據簡介、特徵、應用案例

FAQ:

- 大數據應用案例
  - BigData【世界翻轉中】丹麥版U bike 結合平板電腦+導航
  - BigData【世界翻轉中】不跟上你就輸了！　進入大數據新世界
  - Airbnb 旅行，不再需要住旅館[中文字幕 - 廣告裁判]
- 隨堂測驗
  - 大數據資料處理與分析 隨堂測驗(1)

大數據簡介、特徵、應用案例

隨堂測驗:

根據 IBM 的定義，下列哪一項不是巨量資料（Big Data）的特性？

A. Volume
B. Visualization
C. Variety
D. Veracity

大數據簡介、特徵、應用案例

隨堂測驗:

下列哪個軟體不常用來處理 Big data？

A. R
B. Python
C. Julia
D. Excel

大數據簡介、特徵、應用案例

隨堂測驗:

GB, PB, TB, ZB 為四種電腦記憶體容量的單位,請問它們的大小排序為何?

A. ZB > PB > TB > GB
B. PB > TB > ZB > GB
C. TB > ZB > GB > PB
D. PB > ZB > GB > TB

大數據簡介、特徵、應用案例

隨堂測驗:

Big Data 顧名思義形容資料很 "大", 以下哪一項不是描述其"大" 的主要特徵?

A. 資料總量很大
B. 資料計算的時間很長
C. 資料產生的速度很快
D. 資料的來源很多

大數據簡介、特徵、應用案例

隨堂測驗:

下列哪些是資料科學家所需具備的條件？(複選)

A. 精通資料結構、運算邏輯、物件導向程式設計、自然語言 與影像處理等資訊技術
B. 具備純熟的數學運算能力
C. 運用統計模型、機器學習與作業研究等建模技巧
D. 熟捻會計、財務、行銷與管理等商業語言

大數據簡介、特徵、應用案例

中央氣象局 觀測資料查詢系統

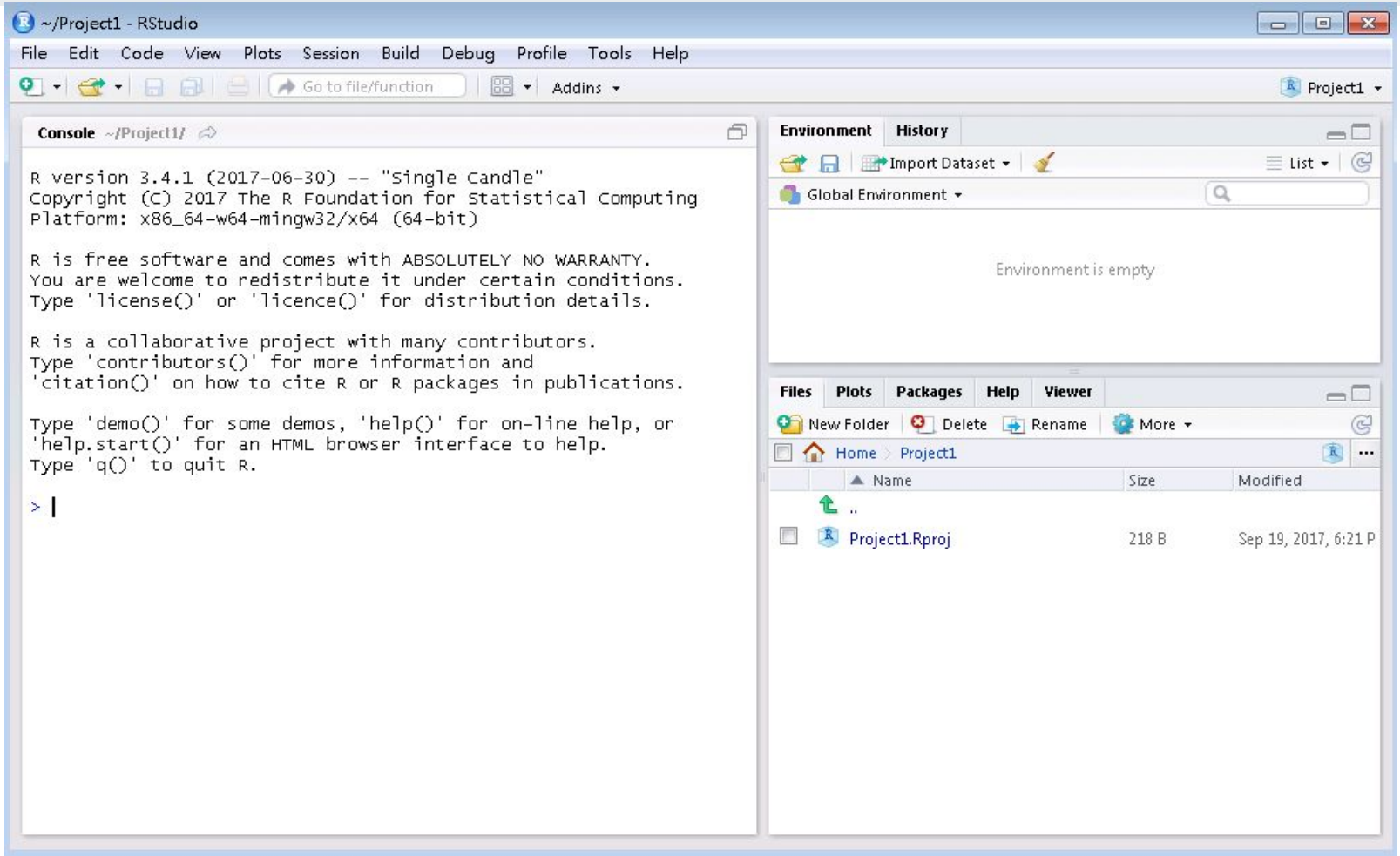
- www.cwb.gov.tw
  - 氣候 -> 氣候統計 -> 觀測資料查詢系統
- 台北市 2017 七月 平均溫度
- 台北市 2017 八月 平均溫度

大數據簡介、特徵、應用案例

隨堂測驗:

讓學期成績比較好看,老師決定幫每個人的考試成績都加 10 分,請問這個數值樣本中的哪個統計量不會因為調分而有差別?

A. 平均值
B. 標準差
C. 中位數
D. 第一四分位數


~/Project1 - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project1
Console ~/Project1/
R version 3.4.1 (2017-06-30) -- "Single Candle"
Copyright (C) 2017 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
>
Environment History
Import Dataset
List
Global Environment
Environment is empty
Files Plots Packages Help Viewer
New Folder
Delete
Rename
More
Home Project1
Name
Size
Modified
..
Project1.Rproj
218 B
Sep 19, 2017, 6:21 P

RStudio IDE Cheat Sheet

RStudio IDE Cheat Sheet

資料的分類

一般的資料分類

1. 定量資料 Quantitative Data (定量變數 = 量變數 = 數值變數)
    - 日常接觸最多，直觀上最容易接受
    - 連續型
        - 身高、體重、成績、價格、薪資
    - 離散型
        - 網站瀏覽人次、班上人數
2. 定性資料 Qualitative Data (定性變數 = 質變數 = 類別變數)
    - 變數本身沒辦法以數量、數值呈現僅能代表不同的類別
        - 例：血型、性別、年級、職稱

R 的變數分類

讓我瞧瞧你是誰！ - mode

物件	object	範例 ex. 形式 mode
數字	age[1]	numeric
數字向量	age	numeric
字串	name[1]	character
字串向量	name	character
因子	factor(country)	numeric
列表	list(name)	list
資料框架	data.frame(Age=age, Name=name)	list
函數	print	function

R 的資料屬性

R 的基本資料屬性包含以下五種，可用 class 函數判斷資料屬性

1. character：字元，用 "" 包起來，ex："test"
2. numeric：實數
3. integer：整數
4. complex：複數
5. logical：True 或 False

R 的資料屬性

numeric 數值型

```
> x=c(84,90,99)
> x
[1] 84 90 99

> class(x)
[1] "numeric"

>is. integer(1)
[1] FALSE
```

integer 整數型

```
> y=1:10
> y
[1]  1  2  3  4  5  6  7  8  9 10

> class(y)
[1] "integer"

>is.integer(1L)
[1] TRUE
```

R 的資料屬性

試試看輸入

.Machine$integer.max

.Machine$double.xmax

a <- 100; b <- 100L
is.numeric(a); is.integer(a)
is.numeric(b); is.integer(b)

R 的資料屬性

logical 邏輯型 (TRUE,FALSE)

```
a = 53>60
> a
[1] FALSE

> is.logical(a)
[1] TRUE
```

logical 邏輯型 (TRUE,FALSE)

```
> x=c(50,70,58,80)
> x
[1] 50 70 58 80

> x < 60
[1]  TRUE FALSE  TRUE FALSE

> which(x<60)
[1] 1 3
```

R 的資料屬性

character字元

```
> x="Taipei City"
> nchar(x)
[1] 11
> nchar(x,type="byte")
[1] 11
> nchar(x,type="width")
[1] 11
```

character 字元

認識中文

```
> x="臺北城市科大"
> nchar(x)
[1] 6
> nchar(x,type="byte")
[1] 18
> nchar(x,type="width")
[1] 12
```

R 的資料屬性

factor 因子型

R 的因子(factor)變數是專門用來儲存類別資料的變數，它同時具有字串與整數的特性。

```
> drink=factor(c("紅茶","可樂","咖啡","紅茶","可樂"))
> drink
[1] 紅茶 可樂 咖啡 紅茶 可樂
Levels: 可樂 咖啡 紅茶
```

```
> levels(drink)
[1] "可樂" "咖啡" "紅茶"
> nlevels(drink)
[1] 3
> length(drink[which(drink=="紅茶")])
[1] 2
> summary(drink)
可樂 咖啡 紅茶
  2   1   2
```

R 的資料屬性

```
> score=40:99
> x=sample(score, 20, replace=TRUE)
> summary(cut(x,breaks=c(0,60,100.1),right=FALSE))
 [0,60) [60,100)
      5       15
> table(x)
```

R 的資料屬性

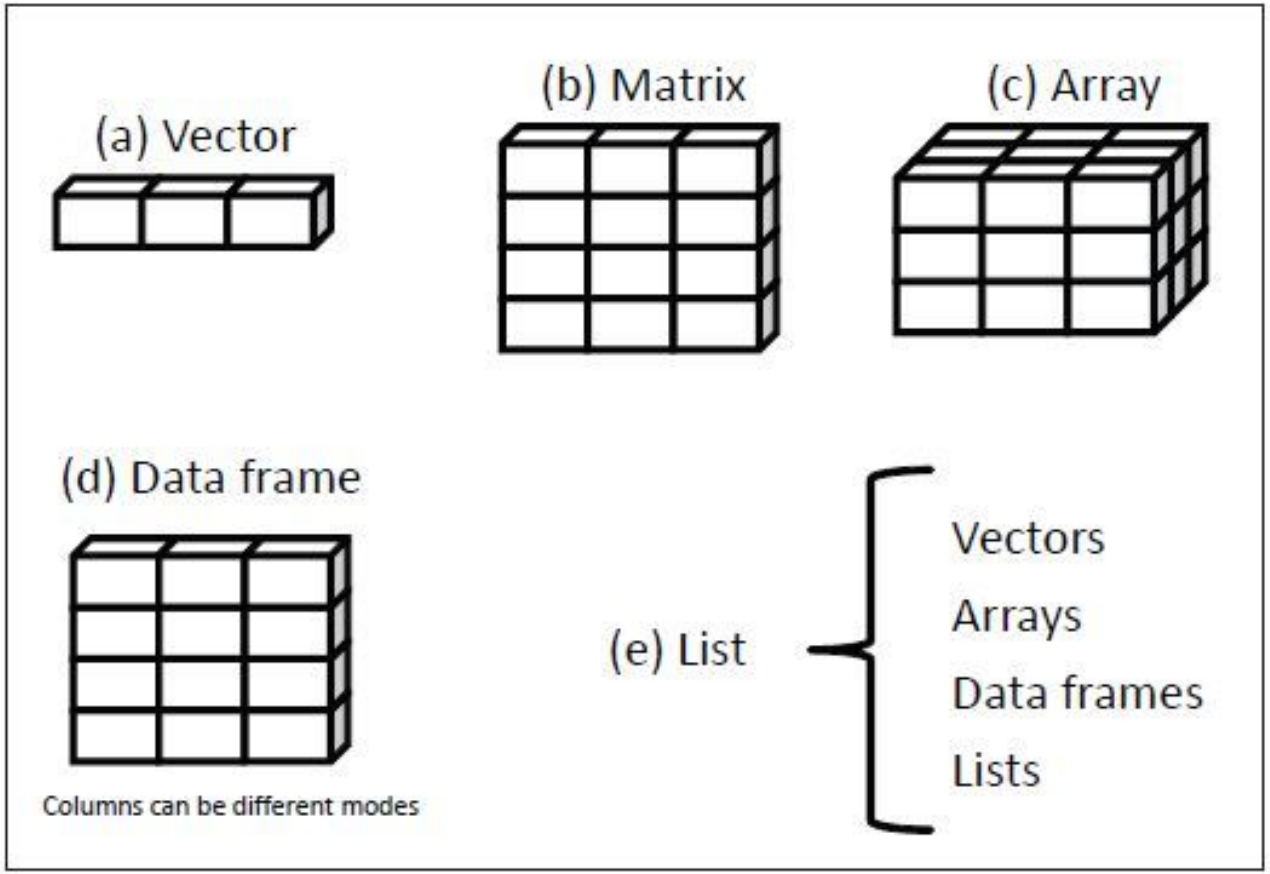
函數
c(), rep(), paste()

撲克牌發牌

```
number = c(1:13,1:13,1:13,1:13)
suit = c(rep("黑桃",13),rep("紅心",13),rep("方塊",13),rep("梅花",13))
cards = paste(suit,number)
sample(cards,5)
[1] "方塊 4"  "紅心 9"  "紅心 4"  "方塊 8"  "梅花 12"
```

R 的資料結構

(a) 向量
(b) 矩陣
(c) 陣列組
(d) 資料框架
(e) 串列



R 的資料結構

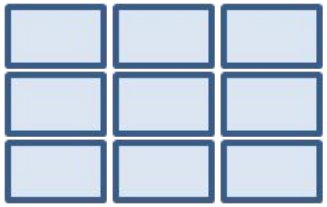
(a) 向量

Vector

- 1 column or row of data
- 1 type (numeric or text)

(b) 矩陣

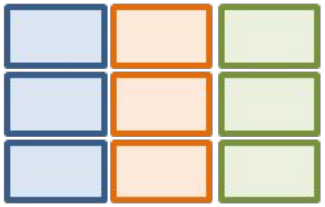
Matrix



- multiple columns and/or rows of data
- 1 type (numeric or text)

(c) 陣列組

Data Frame



- multiple columns and/or rows of data
- multiple types

R 的向量(vector)、矩陣(matrix)、陣列組(Array)

數據資料中,一維資料稱向量(Vector)、二維資料稱矩陣(Matrix),超過二維的資料稱陣列組(Array)

向量(Vector)是一種**「線性的概念」**，相當於是 Microsoft Excel 表格的一列(row)或一行(column),同時存放著**相同類型的資料**。但在真實的世界裡,這是不夠的,我們常碰上需要處理不同類型的資料。

矩陣是一種「平面的概念」，例如表格、教室的座位

R 的向量(vector)、矩陣(matrix)、陣列組(Array)

探索物件的結構:
str( ) 函數可用於探索物件的結構。對於向量而言,可由此瞭解物件的資料類型, 長度和元素內容

```
a=1:10
b=1:10
c=11:20
str(a)
# int [1:10] 1 2 3 4 5 6 7 8 9 10
```

用 identical 函數比較一下 a,b,c 變數

```
identical(a,b)  # TRUE
identical(a,c)  # FALSE
```

R 的向量(vector)、矩陣(matrix)、陣列組(Array)

a=1:10
b=1:10
c=11:20
用 identical 函數比較一下 a,b,c 變數
identical(a,b)
identical(a,c)

R 的資料框架

Most data sets consist of more than just one variable, so to store a complete data set we need a different data structure. In R, several variables can be stored together in an object called a data frame. However, all columns of a data frame must have the same length.

```
name <- c("Joe", "Bob", "Vicky")
age <- c(28, 26, 34)
gender <- c("Male","Male","Female")
data <- data.frame(name, age, gender)
View(data) # 自動點選 data 變數就會開啟資料的畫面。
```

R 的資料框架

明明是字串向量,為何在建立數據框時卻成了因子變數? 這是 R 語言的預設狀況, 如果不想要如此, 在使用 data.frame( ) 函數建立數據框時, 可以增加參數 "stringsAsFactors = FALSE"

```
name <- c("Joe", "Bob", "Vicky")
age <- c(28, 26, 34)
gender <- c("Male","Male","Female")
data1 <- data.frame(name, age, gender, stringsAsFactors = FALSE)
View(data1) # 自動點選 data 變數就會開啟資料的畫面。
```

練習

```
> str(data)
'data.frame':       3 obs. of  3 variables:
 $ name  : Factor w/ 3 levels "Bob","Joe","Vicky": 2 1 3
 $ age   : num  28 26 34
 $ gender: Factor w/ 2 levels "Female","Male": 2 2 1
> str(data1)
'data.frame':       3 obs. of  3 variables:
 $ name  : chr  "Joe" "Bob" "Vicky"
 $ age   : num  28 26 34
 $ gender: chr  "Male" "Male" "Female"
```

練習

str( ) 函數可以查看矩陣物件的結構。
比較
>str(data)
>str(data1)
> identical(data,data1)
[1] FALSE

R 的變數分類

```
a=array(1:24,dim=c(6,4))
b=matrix(1:24,nrow=6)
identical(a,b)
[1] TRUE
```

擷取 dataframe 資料

Taking a Subset of a Data Frame

The ‘ChickWeight’ data frame has 578 rows and 4 columns from an experiment on the effect of diet on early growth of chicks.

The body weights of the chicks were measured at birth and every second day thereafter until day 20. They were also measured on day 21. There were four groups on chicks on different protein diets.

This contains the weights of little chickens at 12 different times throughout their lives. The chickens are on different diets, numbered 1, 2, 3, and 4

擷取 dataframe 資料

Get One Column

```
ChickWeight[,1]              # get all rows, but only the first column
ChickWeight[,c("weight")]    # get all rows, and only the column named “weight”
ChickWeight$weight           # get all rows, but only the “weight” column
```

Get Multiple Columns

```
ChickWeight[,c(1,4)]              # get all rows, but only 1st and 4th columns
ChickWeight[,c("weight","Diet")]      # get all rows, only “weight” & “Diet” columns
ChickWeight[,c(1:3)]
```

擷取 dataframe 資料

Get One Row

```
ChickWeight[1,]          # get first row, and all columns
ChickWeight[95,]         # get 95nd row, and all columns
```

Get Multiple Rows Where One Variable Has a Certain Value

```
ChickWeight[ChickWeight$Diet == 1,]          # get all rows where Diet is 1
ChickWeight[ChickWeight$Chick == 20,]          # get all rows for Chick #20
```

練習 擷取 dataframe 資料

擷取 Chick 編號為1號的所有資料

```
subset(ChickWeight,Chick==1)
subset(ChickWeight,Chick==1,select=c(Time,weight))
```

擷取 Chick 編號為18號的所有資料, 並顯示符合的筆數

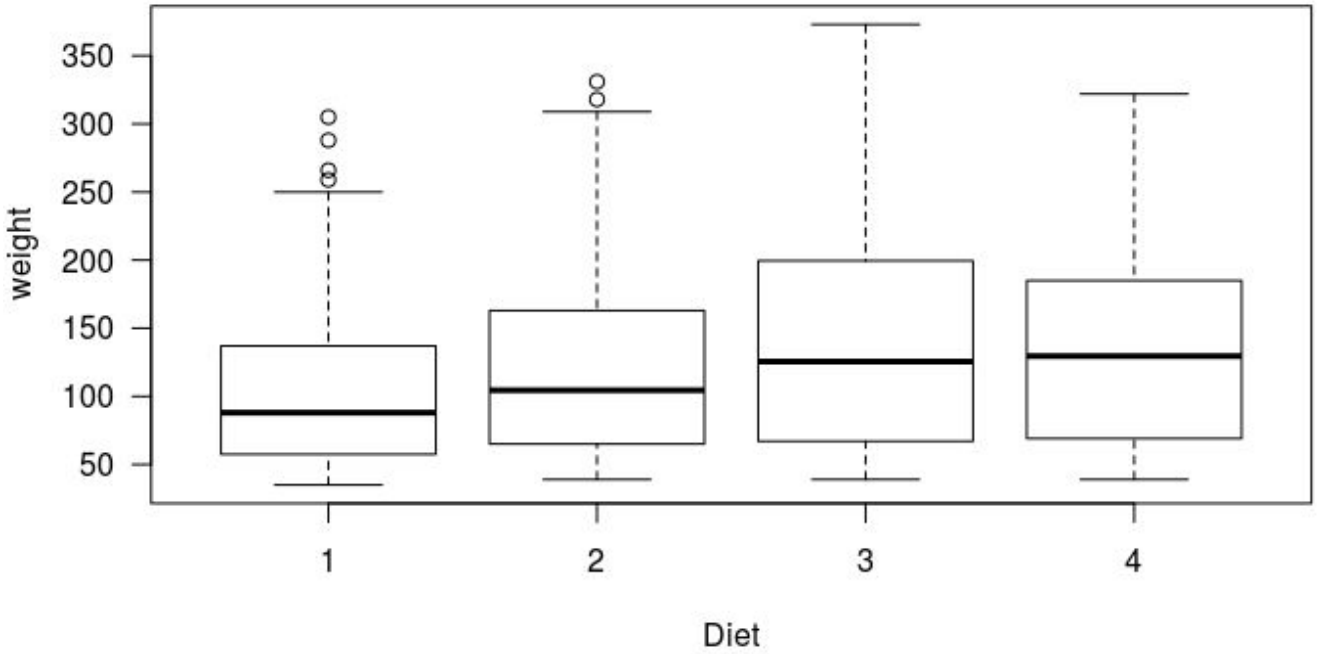
```
subset(ChickWeight,Chick==18)
nrow(subset(ChickWeight,Chick==18))
```

擷取 Diet 為3的所有資料

```
subset(ChickWeight,Diet==3)
```

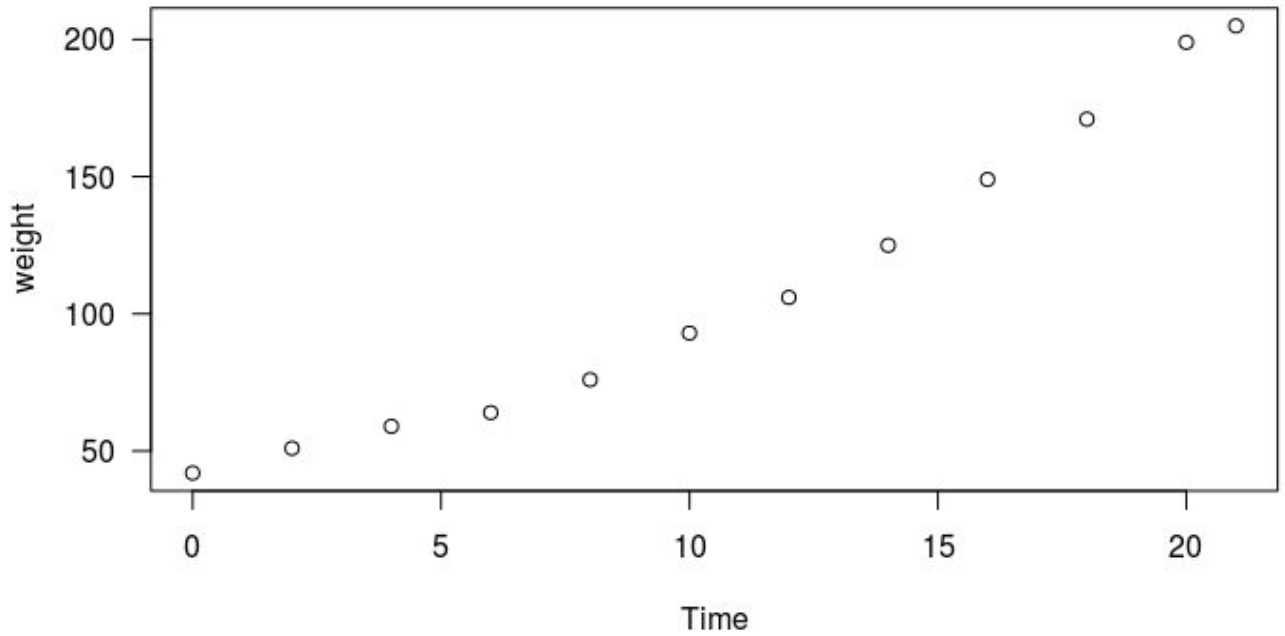
練習 ChickWeight

```
# 繪製 boxplot 圖示四種食譜的效果
boxplot(ChickWeight$weight ~ ChickWeight$Diet, las=1)
```



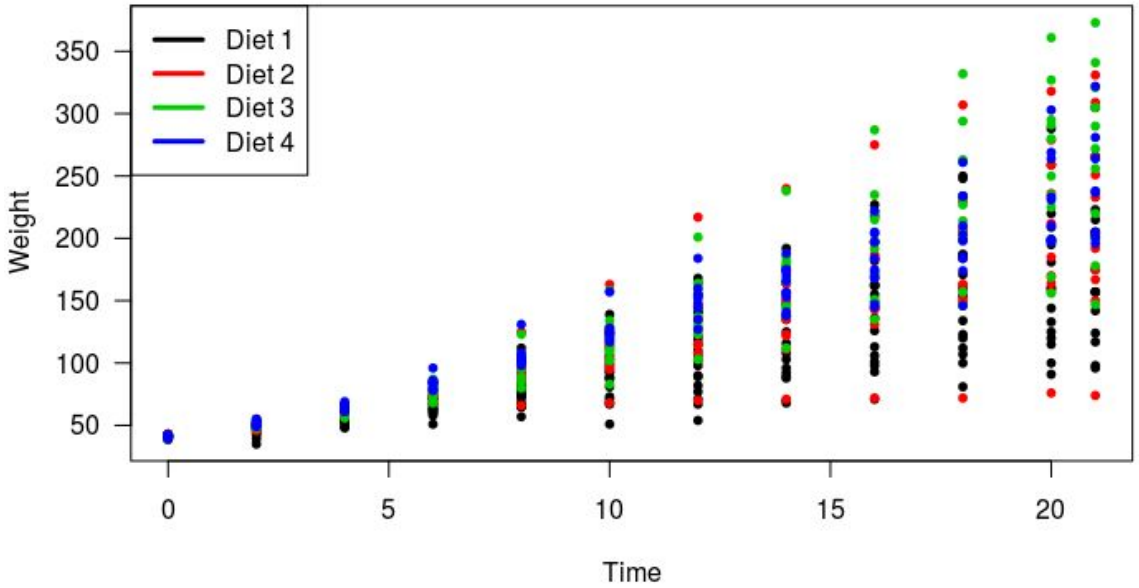
練習 ChickWeight

```
# 繪製 plot 圖示 1號 Chick 的成長體重
plot(subset(ChickWeight,Chick==1,select=c(Time,weight)),type='p',las=1)
```



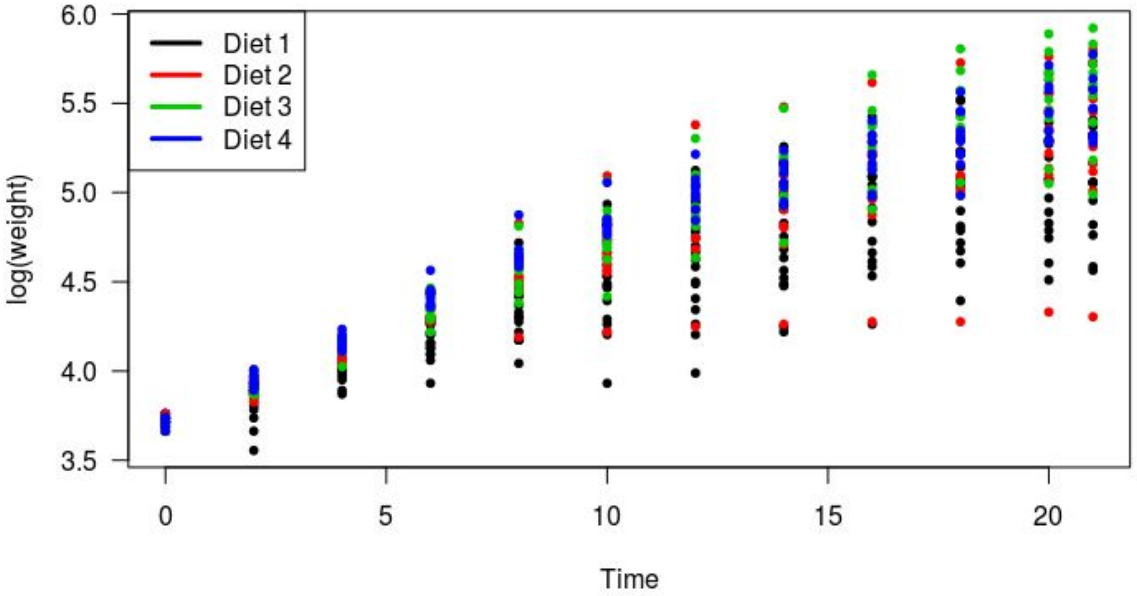
練習 ChickWeight

```
# 繪製 plot 圖示 Chick的成長體重
with(ChickWeight, plot(Time, weight, col = Diet,pch=20,ylab='Weight'))
legend('topleft',legend=paste("Diet",levels(ChickWeight$Diet)),col=1:4,lwd=3)
```



練習 ChickWeight

```
# 繪製 plot 圖示 Chick的成長體重 (取 log)
with(ChickWeight, plot(Time, log(weight), col = Diet,pch=20))
legend('topleft',legend=paste("Diet",levels(ChickWeight$Diet)),col=1:4,lwd=3)
```



九九乘法表運算

```
for (i in 1:9) {
 for (j in 1:9) {
  k = i * j
  cat(i)
  cat("x")
  cat(j)
  cat("=")
  cat(k)
  cat(" ")
 }
 cat("\n")
}
```

```
1x1=1 1x2=2 1x3=3 1x4=4 1x5=5 1x6=6 1x7=7 1x8=8 1x9=9
2x1=2 2x2=4 2x3=6 2x4=8 2x5=10 2x6=12 2x7=14 2x8=16 2x9=18
3x1=3 3x2=6 3x3=9 3x4=12 3x5=15 3x6=18 3x7=21 3x8=24 3x9=27
4x1=4 4x2=8 4x3=12 4x4=16 4x5=20 4x6=24 4x7=28 4x8=32 4x9=36
5x1=5 5x2=10 5x3=15 5x4=20 5x5=25 5x6=30 5x7=35 5x8=40 5x9=45
6x1=6 6x2=12 6x3=18 6x4=24 6x5=30 6x6=36 6x7=42 6x8=48 6x9=54
7x1=7 7x2=14 7x3=21 7x4=28 7x5=35 7x6=42 7x7=49 7x8=56 7x9=63
8x1=8 8x2=16 8x3=24 8x4=32 8x5=40 8x6=48 8x7=56 8x8=64 8x9=72
9x1=9 9x2=18 9x3=27 9x4=36 9x5=45 9x6=54 9x7=63 9x8=72 9x9=81
```

R 的基本數學運算

四則運算

- R 的四則運算是指加(+)、減(-)、乘(*)和除(/)。

餘數和整除

- 餘數(mod)所使用的符號是 %% 可計算出除法運算中的餘數
  - 16%%3
- 整除所使用的符號是 %/% 是指除法運算中只保留整數部分
  - 16%/%3

R 的基本數學運算

次方或平方根

- 次方的符號是 ** 或 ^
  - 2**3
  - 2^3
- 平方根的符號是使用函數 sqrt( )
  - sqrt(16)
  - 16**0.5

R 的基本數學運算

四捨五入函數

- round(x, digits = k) 相當於將實數 x 以四捨五入方式，計算至第 k 位小數。另外 round( ) 函數中的第 2 個參數 "digits =" 也可以省略，直接在第 2 個參數位置輸入數字
  - round(87.4567, digits=2)
  - round(87.4567, 2)
- signif(x, digits = k),也是一個四捨五入的函數, k 是實數 x的「有效數字」個數
  - signif(79843.597, digits = 6),代表取 6 個數字,左邊算來第 7 個數字以四捨五入方式處理

R 的基本數學運算

R 語言有 6 個近似函數

- round(x): round(23456.789, 2) 得到23456.79四捨五入
- signif(x): signif(23456.789, 3) 得到 23500 四捨五入結果
- floor(x): 可得到小於等於x的最近整數
  - floor(5.6)
- ceiling(x): 可得到大於等於x的最近整數
  - ceiling(5.6)
- trunc(x):可直接取整數
  - trunc(5.6)

R 的基本數學運算

階乘

- factorial(x) 可以計算 x 的階乘
  - factorial(5)
    - 120
- prod(x):計算所有元素的積。
  - x=1:5
  - prod(x) 相當於 fractorial(5)

R 的基本數學運算 因數分解

Writing a function to calculate divisors in R

```
divisors <- function(x){
  y <- seq_len(x) # 產生 x 的除數
  y[ x%%y == 0 ] # 判斷 x/y 的餘數是否為 0
}

divisors(200)
[1]  1  2  4  5  8 10 20 25 40 50 100 200
```

R 的基本數學運算

無限大 Infinity

- R 語言可以處理無限大的值,使用代號值 **Inf**，如果是負無限大則是 **-Inf**
  - 某數減去 **Inf**,可以獲得負無限大 **-Inf**
    - 50-Inf
  - 某數除以 0 就可獲得無限大
    - 50/0
- 如果某一個數字除以**無限大 Inf** 或**負無限大 -Inf** 是多少?
  - 答案是 0
- 使用 is.infinite( ) 測試 Inf 和 -Inf 是否為正或負無限大

R 的基本數學運算

Not a Number(NaN)

- Not a Number(NaN) 可以解釋為非數字或稱無定義數字
- 數字除以 0 是無限大，任一數字除以無限大是 0
- **Inf / Inf** 可以獲得 **NaN** 代表 Not a Number
- 使用 **NaN** 做四則運算,所得結果皆是 **NaN**
  - NaN +18
- is.nan(x) 函數,可檢測 x 值是否為 NaN,如果是則傳回 TRUE,否則傳回 FALSE

R 的基本數學運算

Not Available(NA)

- Not Available 也可稱缺失值或是遺漏值英文簡稱 **NA**
- 任何數與 NA 做四則運算中，計算結果是 NA
  - 5+NA
  - Inf+NA
  - is.nan(Inf+NaN)
    - TRUE

R 的基本數學運算

練習 哪一國的 iPhone 8 比較便宜

- i8.64G.us = c(699,799,999)
- i8.64G.tw = c(25500,28900,35900)
- i8.64G.jp = c(78800,89800,112800)
- 台灣的 iPhone 8 比較貴? 貴多少? 貴幾成? 請四捨五入到小數點2位
  - i8.64G.tw - i8.64G.us * 30
  - i8.64G.tw / (i8.64G.us * 30)
  - round(i8.64G.tw / (i8.64G.us * 30),2)

R 的基本數學運算

練習

- x = 366.379 將 x 放入 floor( )、ceil( ) 和 trunc( ),使用預設值測試,並列出結果。
- 重複上一習題,將 x 改為負值 -366.379 並列出結果。
- s = sample(30:99, 100, replace=TRUE)
- sum(x<60)
- which(x<60)
- x[which(x<60)]

R 的基本數學運算

計算下列執行結果

- Inf + 100
- Inf – Inf + 10
- NaN + Inf
- Inf - NaN
- NA + Inf
- Inf – NA
- NaN + NA

練習

列出當月有 31 天的月份, 系統內建資料集變數

- month.name
- month.abb
- month.date=c(31,28,31,30,31,30,31,31,30,31,30,31)
- names(month.date) = month.name
- names(month.date) = NULL
- 求 1 月到 9 月共計幾天?
  - sum(month.date[1:10])
- 求 31 天的月份
  - names(month.days[month.days == 31])

系統內建的資料集 letters 和 LETTERS

R 語言系統內建的數據集 letters 和 LETTERS 為例, 可以取得英文字母的小寫與大寫字母

如何取得向量的部分元素或稱取得子集(Subsetting)

- letters 物件索引值是 10 和 18
- letter[c(10,18)]
- LETTER[18:22]

使用 tail(LETTER, 8) 函數可取得 LETTERS 物件最後 8 筆元素。
如果省略第 2 個參數, 系統自動返回最後的 6 個元素

邏輯向量(Logical Vector) TRUE 和 FALSE

x == y 如果 x 等於 y,則傳回 TRUE
x ! = y 如果 x 不等於 y,則傳回 TRUE
x > y 如果 x 大於 y,則傳回 TRUE
x > = y 如果 x 大於或等於 y,則傳回 TRUE
x < y 如果 x 小於 y,則傳回 TRUE
x < = y 如果 x 小於或等於 y,則傳回 TRUE
x & y 相當於 AND 運算,如果 x 和 y 皆是 TRUE 則傳回 TRUE
x | y 相當於 OR 運算,如果 x 或 y 是 TRUE 則傳回 TRUE
! x 相當於 NOT 運算,傳回非 x
xor(x, y) 相當於 XOR 運算,如果 x 和 y 不同,則傳回 TRUE

邏輯向量(Logical Vector) TRUE 和 FALSE

Count number of vector values in range with R

```
v <- c(1,2,3,4,5)
sum(v >= 2)
sum(v <= 4)
sum(v >= 2 & v <=4)
```

向量物件的元素名稱

建立簡單含元素名稱的向量

雖然我們可以使用索引很方便取得向量物件的元素,R 語言有一個強大的功能是為向量的每一個元素命名,未來我們也可以利用物件的元素名稱引用元素內容。

下列是建立向量物件,同時物件元素含名稱的方法。

object = c(name1= data1, name2 = data2, ...)
fruit = c(apple=40,banana=20,orange=50)
fruit = c(蘋果=40,香蕉=20,橘子=50)

練習

使用系統內建資料集變數 islands,列出排名最小與第3小的島名稱和面積

- islands
- x = sort(islands) ; head(names(x),10)

取得最大的 10 個島嶼的名稱,只列出名稱

- big10.islands = head(sort(islands,decreasing=TRUE),10)
- names(big10.islands)

練習

i8.64G.tw = c(25500,28900,35900)
names(i8.64G.tw) = c('iPhone8', 'iPhone8 Plus', 'iPhone X')
round(i8.64G.tw/30) - c(699,799,999)

練習

```
FILE="201710.csv"
DATA = read.csv(FILE,skip=0,header=FALSE,stringsAsFactors=FALSE)
names(DATA) = c("觀測時間","測站氣壓","海面氣壓","測站最高氣壓",
"測站最高氣壓時間","測站最低氣壓","測站最低氣壓時間","氣溫","最高氣溫",
"最高氣溫時間","最低氣溫","最低氣溫時間","露點溫度","相對溼度","最小相對溼度",
"最小相對溼度時間","風速","風向","最大陣風","最大陣風風向","最大陣風風速時間",
"降水量","降水時數","觀測10分鐘最大降水量","觀測10分鐘最大降水起始時間",
"一小時最大降水量","一小時最大降水量起始時間","日照時數","日照率",
"全天空日射量","能見度","A型蒸發量")

plot(DATA$最低氣溫,xlab="日期",ylab="最低溫度",main="2017 Oct")
```

關於向量 vector

Indexing Vectors

If you have a vector and want the i’th element of that vector, you can index the vector to get it like this:

```
v <- 1:5
[1] 1 2 3 4 5

v[1]
[1] 1

v[3]
[1] 3
```

關於向量 vector

You can even use a vector of Boolean values to pick out those values that are "true":

```
v[c(TRUE, FALSE, TRUE, FALSE, TRUE)]
[1] 1 3 5
```

It is also possible to give vector indices names and, if you do, you can use those to index the vector. You can set the names of a vector when constructing it or use the names() function.

```
v <- c("A" = 1, "B" = 2, "C" = 3)
v

A B C
1 2 3
```

```
v["A"]
A
1
```

```
names(v) <- c("x", "y", "z")
v
x y z
1 2 3
v["x"]
x
1
```

練習

尋找 因數

```
x=200
y=1:x
x%%y
x%%y==0
y[x%%y==0]
```

練習

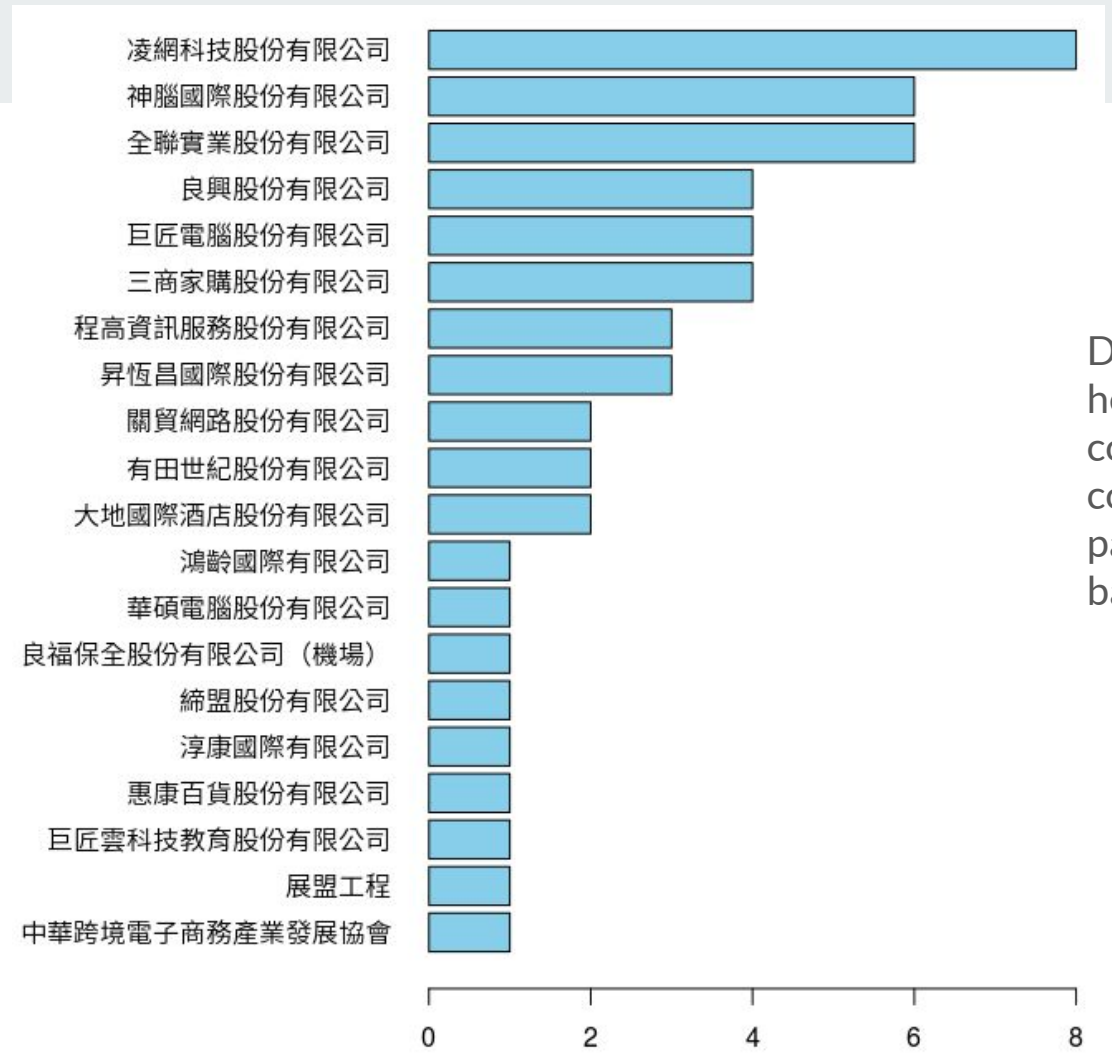
尋找 因數

```
divisors = function(x) {
  #  Vector of numberes to test against
  y <- seq_len(x)
  #  Modulo division. If remainder is 0 that number is a divisor of x so return it
  y[ x%%y == 0 ]
}

divisors(200)
```

練習

撲克牌發牌 5 張, 請新增 deal 函數來做為發牌的函數

```
suit=c('黑桃','紅心','方塊','梅花')
n=1:13
a=sample(suit,5,replace=TRUE)
b=sample(n,5)
paste(a,b)
```




```
D <- read.csv("106_0911.csv",
header=TRUE,stringsAsFactors=TRUE)
company = table(D)
company = sort(company)
par(mar=c(5,13,1,2)) # 設定繪圖範圍
barplot(company,horiz=T,las=1,col="skyblue")
```

來玩大數 Big Number 商數

安裝gmp
install.packages('gmp')
library(gmp)

x= 2^2000 # 出現 Inf 無限大
> 2^2000 / 2^1000
[1] Inf
x= as.bigz(2)^2000

練習
求 2^2000 / 2^1000 的結果是?
x = as.bigz(2)^2000
y = as.bigz(2)^1000
div.bigz(x,y)

來玩大數 Big Number 階乘

安裝gmp
install.packages('gmp')
library(gmp)

階乘函數
factorial(5)
factorial(23) # 就超過範圍了
factorialZ(23) # 23! 就超過範圍了
cumprod(1:23) # 23! 就超過範圍了

練習
求 100! 的結果是?
factorialZ(100)

來玩大數 Big Number 求位數

安裝gmp
install.packages('gmp')
library(gmp)

x= 2^2000 # 出現 Inf 無限大
x= as.bigz(2)^2000

練習
求 2^2000 的結果是幾個位數？
as.character: Create or test for objects of type ‘"character"’
nchar: ‘nchar’ takes a character vector as an argument and returns a vector whose elements contain the sizes of the corresponding elements of ‘x’.

來玩大數 Big Number 質數

安裝gmp
```
install.packages('gmp')
library(gmp)
```

練習: 求 1到100 之間的質數
```
n = 1:100
p = isprime(n)
n[p==2]
```

四分位數 Quartile

Example:

3, 4, 4, 5, 6, 8, 8

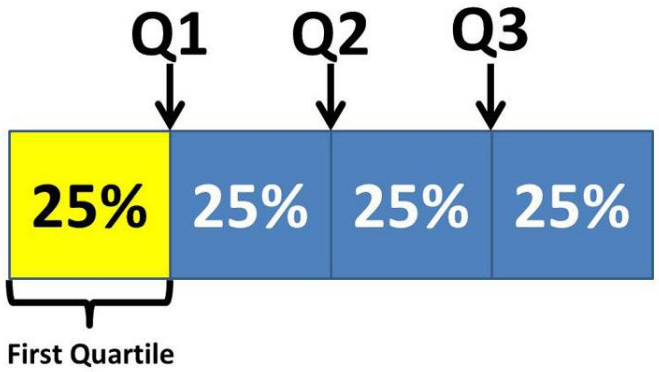
Q1 lower quartile

Q2 middle quartile (median)

Q3 upper quartile

The **Interquartile Range** is:

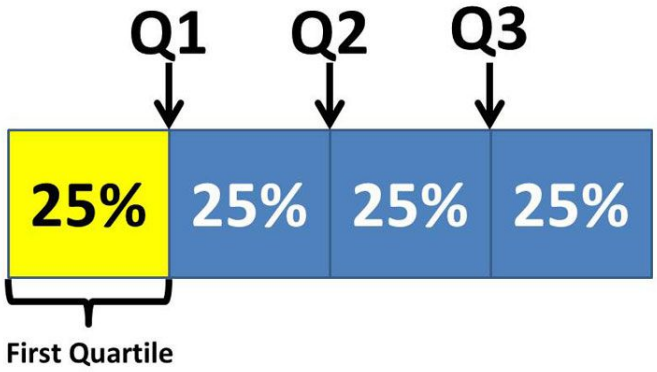
Q3 - Q1 = 8 - 4 = **4**

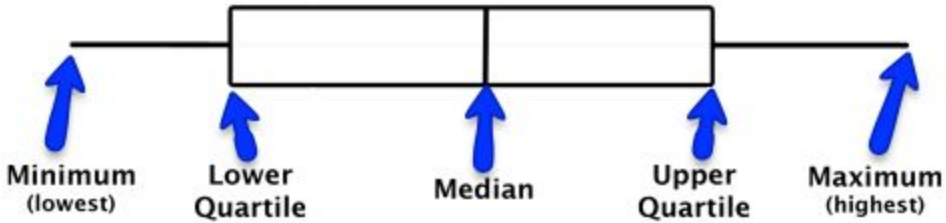



```
x=c(3,4,4,5,6,8,8)
quantile(x,type=1)
  0%  25%  50%  75% 100%
   3    4    5    8    8
```

可能不是很合理！

四分位數 Quartile





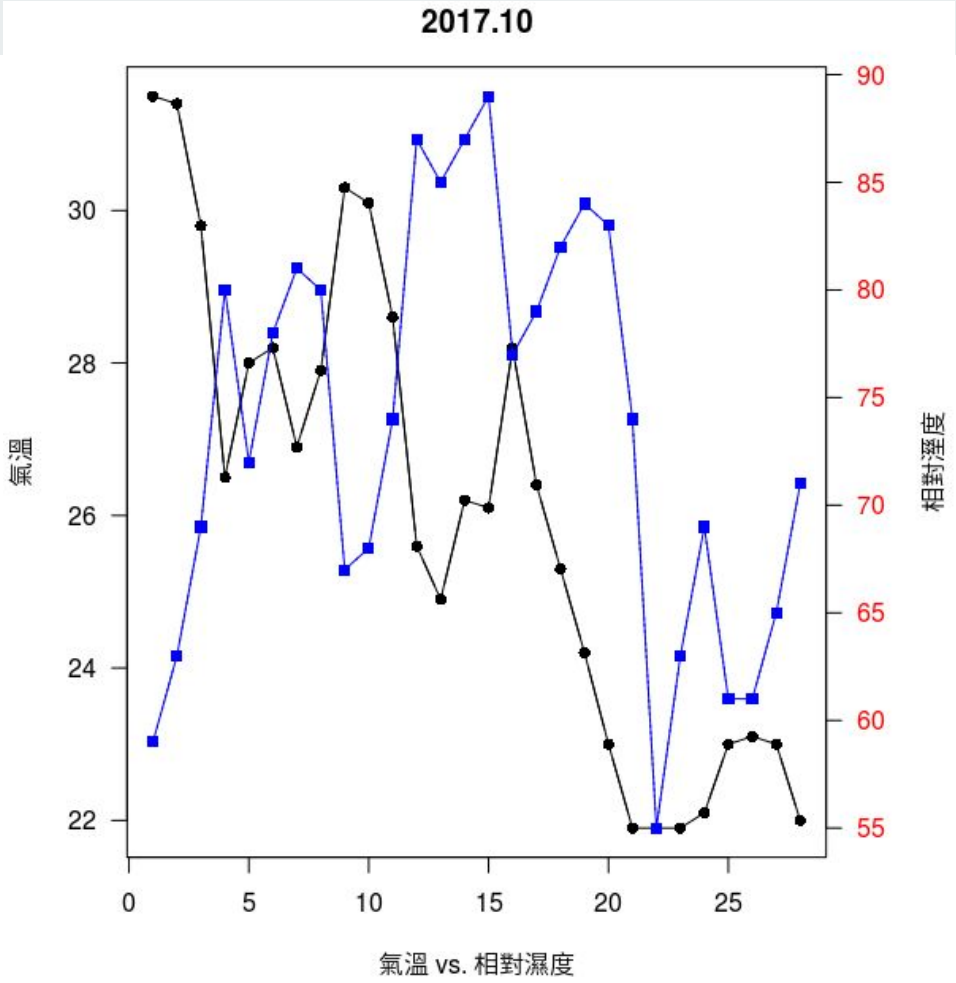
n 表示有多少個數字
Q1 位置 = 1+(n-1)/4*1
Q2 位置 = 1+(n-1)/4*2
Q3 位置 = 1+(n-1)/4*3

Q1 的值 = 取 floor(Q1 位置)的值+Q1 位置相鄰的兩數之差乘上 Q1的小數值

Q2 的值 = 取 floor(Q2 位置)的值+Q2 位置相鄰的兩數之差乘上 Q2的小數值

Q3 的值 = 取 floor(Q3 位置)的值+Q3 位置相鄰的兩數之差乘上 Q3的小數值

```
quantile(c(3,4,4,5,6,8,8))
  0%  25%  50%  75% 100%
   3    4    5    7    8
```




```
MAIN="2017.10"
par(new=FALSE)
plot(DATA$氣溫,type="b", pch=16, col="black",las=1,
xlab="氣溫 vs. 相對濕度" ,ylab="氣溫", axes=TRUE)
lines(DATA$氣溫)
par(new=TRUE)
plot(DATA$相對溼度,type="b",main=MAIN, pch=15,
col="blue", lty=3, xlab="", ylab="",axes=FALSE)
lines(DATA$相對溼度,col="blue")
axis(4,col.axis="red",las=1)
mtext(side = 4, line = 2.5, '相對溼度',las=0)
```

練習

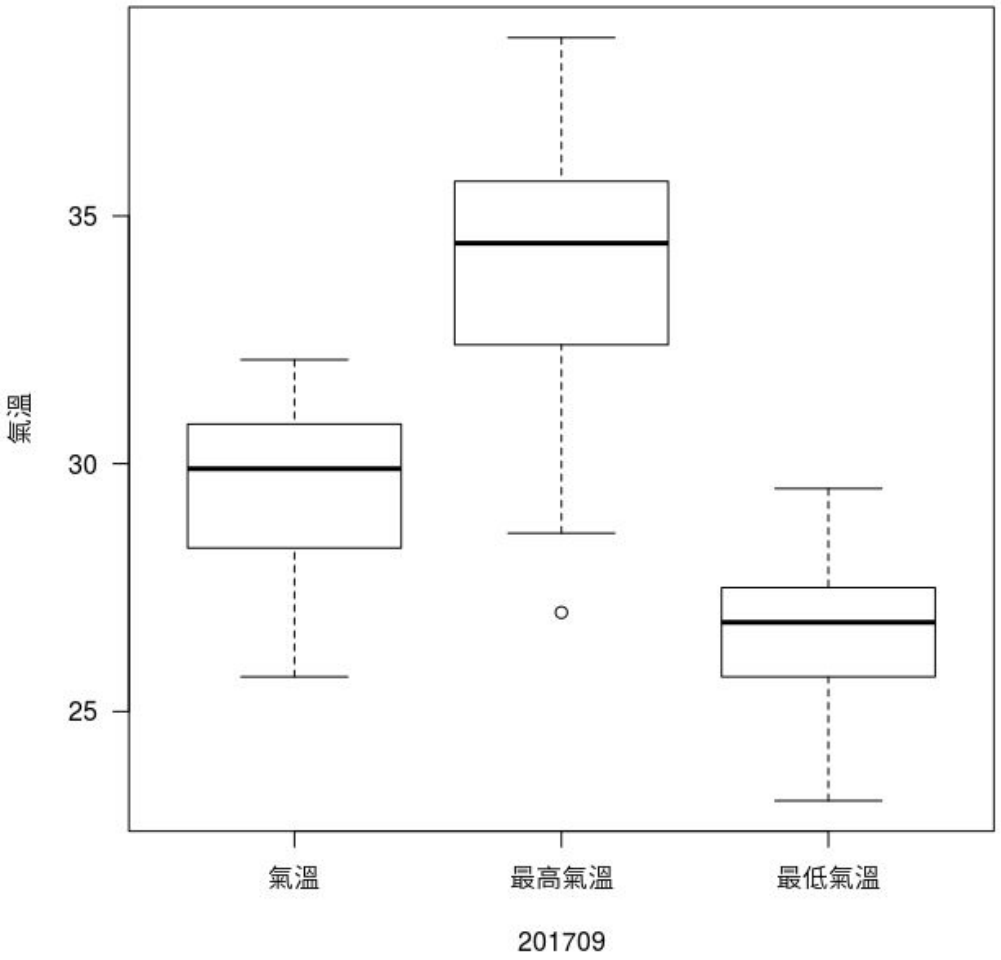
```
title=c("觀測時間","測站氣壓","海面氣壓",
"測站最高氣壓","測站最高氣壓時間","測站最低氣壓",
"測站最低氣壓時間","氣溫","最高氣溫","最高氣溫時間",
"最低氣溫","最低氣溫時間","露點溫度","相對溼度",
"最小相對溼度","最小相對溼度時間","風速","風向",
"最大陣風","最大陣風風向","最大陣風風速時間",
"降水量","降水時數","觀測10分鐘最大降水量",
"觀測10分鐘最大降水起始時間","一小時最大降水量",
"一小時最大降水量起始時間","日照時數","日照率",
"全天空日射量","能見度","A型蒸發量")
x=read.csv("201709.csv",skip=1,header=F)
names(x) = title
y=x[,c(1,2,8,9,11,14)];
y['label']='201709'
par(mar=c(5,4,4,2)+1)
boxplot(y[,c(3,4,5)],ylab="氣溫" ,xlab="201709", las=1)
```

如何取得 "最高氣溫" 變數?

直接輸入 "最高氣溫" 是不行的,因為它是在 y 裡面的欄位。

```
y$最高氣溫 #意思是取出 y 裡的最高氣溫資料
y[ , 3] #意思是「 y 裡的第 3 欄資料」
y[c(3,6), ] #意思是「 y 裡的第 3 、6列資料」

subset(y,最高氣溫>36)

計算總列數
nrow(subset(y,最高氣溫>36))
```

Q: 如何計算 2017 的 7,8,9,10 月最高氣溫超過 36 度的天數？




```
get_weather=function(FILE,LABEL) {
title=c("觀測時間","測站氣壓","海面氣壓",
"測站最高氣壓","測站最高氣壓時間","測站最低氣壓",
"測站最低氣壓時間","氣溫","最高氣溫","最高氣溫時間",
"最低氣溫","最低氣溫時間","露點溫度","相對溼度",
"最小相對溼度","最小相對溼度時間","風速","風向",
"最大陣風","最大陣風風向","最大陣風風速時間",
"降水量","降水時數","觀測10分鐘最大降水量",
"觀測10分鐘最大降水起始時間","一小時最大降水量",
"一小時最大降水量起始時間","日照時數","日照率",
"全天空日射量","能見度","A型蒸發量")
x=read.csv(FILE,skip=1,header=F)
names(x) = title; y=x[,c(1,2,8,9,11,14)];
y['label']=LABEL;  return (y) ;
}
sep = get_weather('201709.csv', '201709')
par(mar=c(5,4,4,2)+1)
boxplot(sep[,c(3,4,5)],ylab="氣溫" ,xlab="201709", las=1)
```

練習

Q: 請計算出 2017 的 7,8,9,10,11 月最高氣溫超過 36 度的天數？

Q: 請計算出 2017 的 7,8,9,10,11 月最低氣溫低於 20 度的天數？

Q: 請計算出 2017 的 7,8,9,10,11 月濕度超過70%的天數？

Q: 請計算出 2017 的 10 月每日最高氣溫與最低溫度的溫差？

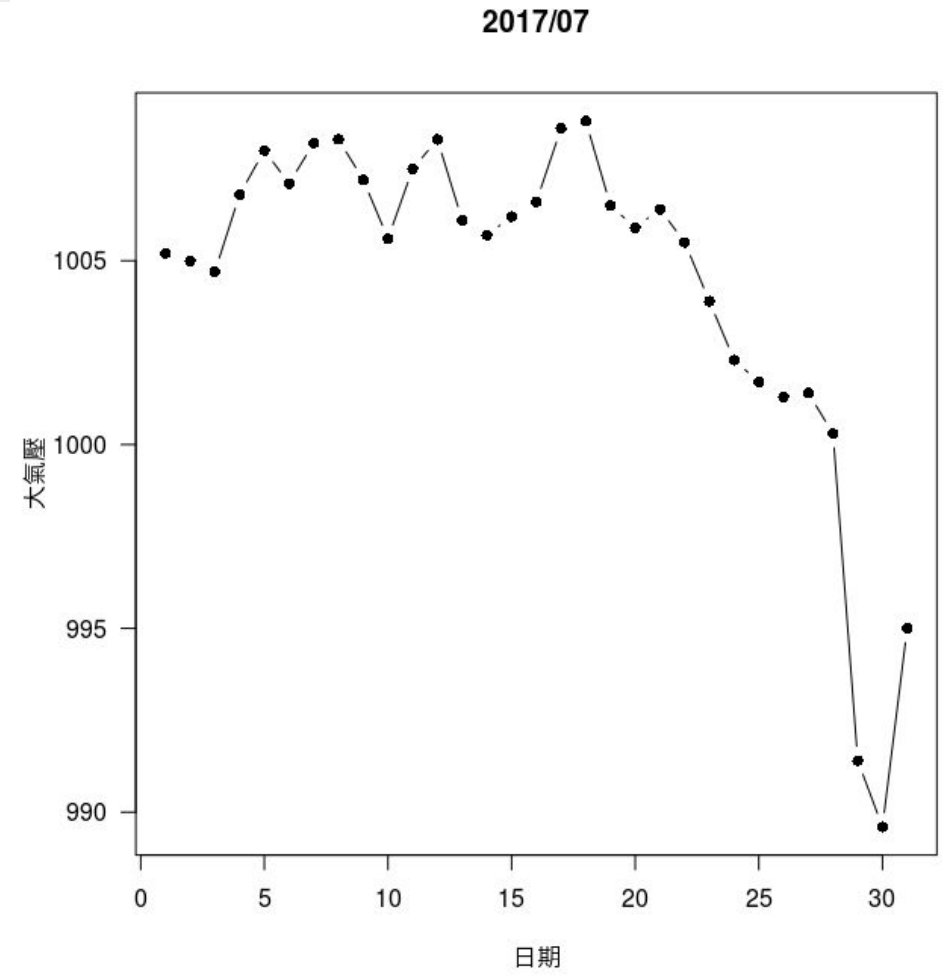
Q: 請計算出 2017 的 11 月每日最高氣溫與最低溫度的溫差？

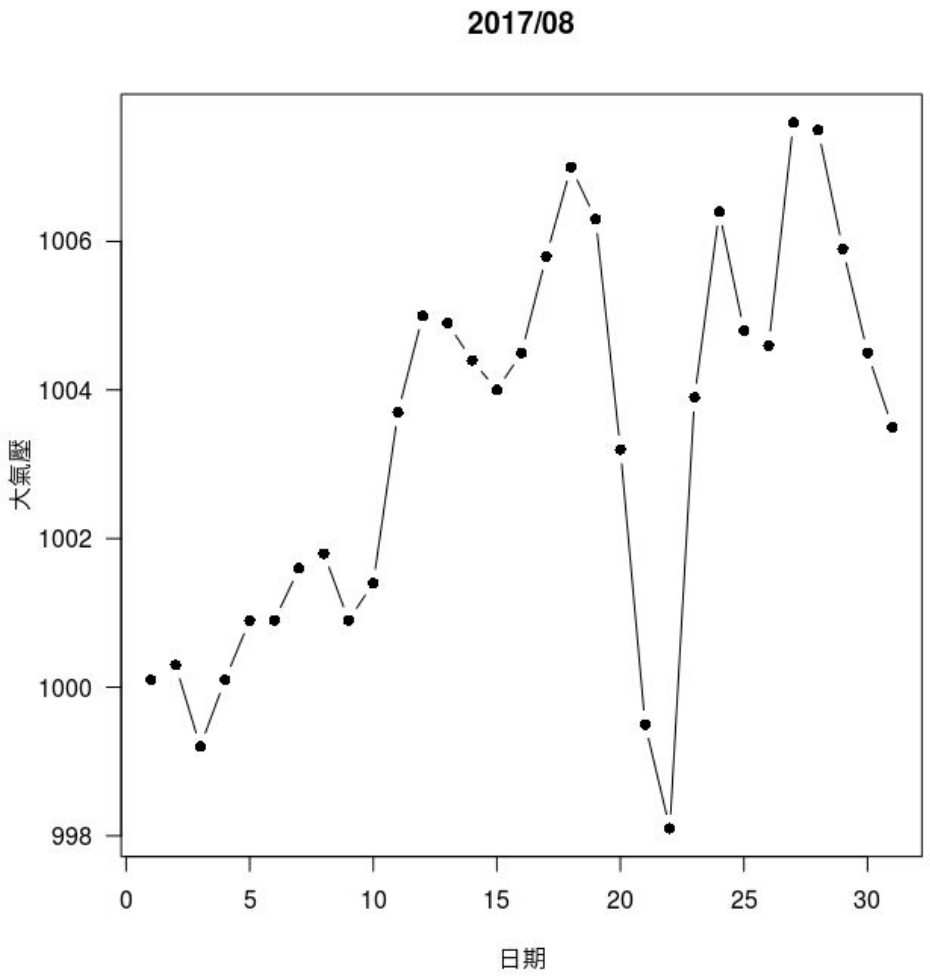
練習

Q: 請計算出 2017/ 7 有沒有颱風? 請用 plot 嘗試畫圖

```
plot(jul$測站氣壓,type="b", pch=16, col="black",las=1,xlab="日期" ,
ylab="大氣壓", main="2017/07", axes=TRUE)
```

```
plot(jul$測站氣壓,type="o", pch=16, col="black",las=1,xlab="日期" ,
ylab="大氣壓", main="2017/07", axes=TRUE)
```


2017/07
大氣壓
1005
1000
995
990
0
5
10
15
20
25
30
日期


2017/08
大氣壓
1006
1004
1002
1000
998
0
5
10
15
20
25
30
日期

練習 Orange

data(Orange)

The ‘Orange’ data frame has 35 rows and 3 columns of records of the growth of orange trees.

plot

subset(Orange,Tree==1)

plot(subset(Orange,Tree==1))

練習 summary

```
> summary(jul$相對溼度)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  54.00   58.00   62.00   63.55   68.50   79.00
```

```
> summary(aug$相對溼度)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  55.00   60.50   64.00   64.26   66.00   79.00
```

```
> summary(sep$相對溼度)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  54.00   58.00   62.50   65.10   69.75   85.00
```

```
> summary(oct$相對溼度)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  55.00   65.00   73.00   73.00   80.75   89.00
```

```
> summary(nov$相對溼度)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   68.0    73.0    76.0    76.6    80.0    88.0
```

練習 抓取 2010 年份氣象資料

```
fields=c(1,2,8,9,10,11,14)

w2010=x[grep("2010-",x$最高氣溫時間), fields]
w2010[,"month"]=as.numeric(format(as.Date(w2010$最高氣溫時間),"%m"))
table(w2010$month)
```

table 函數可以自動統計因子所有元素在各個levels 出現的次數統計

練習 敘述統計函式

敘述統計是進行資料分析時，有效瀏覽(explore)、了解(recognize)資料狀態的步驟。
作為一個資料分析軟體，R裡面自然有許多可以協助我們進行敘述統計分析的函式：

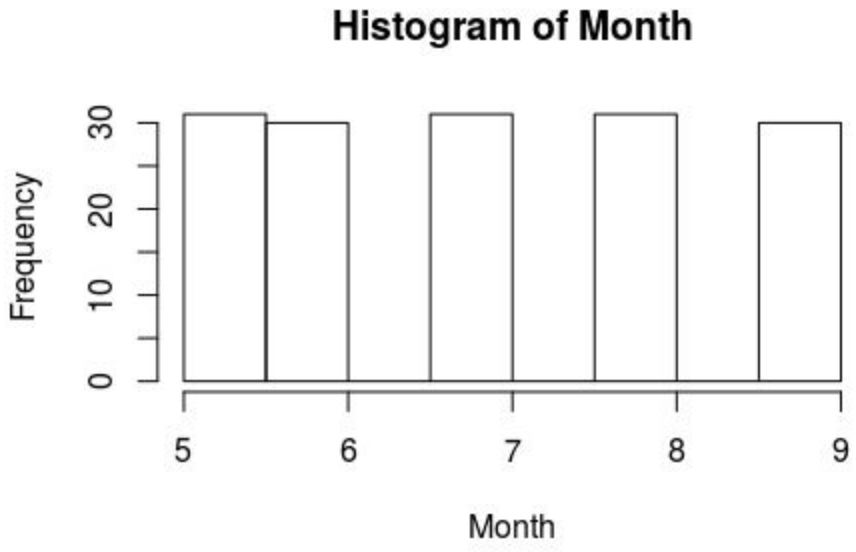
mean()：平均值
var()：變異數
sd()：標準差
median()：中位數
max()：最大值
min()：最小值
sum()：綜合相加
quantile()：分位數
range()：全距

練習 一個關於空氣品質的資料集airquality

```
require(datasets)
head(airquality)
table(airquality$Month) # 查看每月有幾筆
hist(x=airquality$Month, main="Histogram of Month", xlab="Month", ylab="Frequency")
```

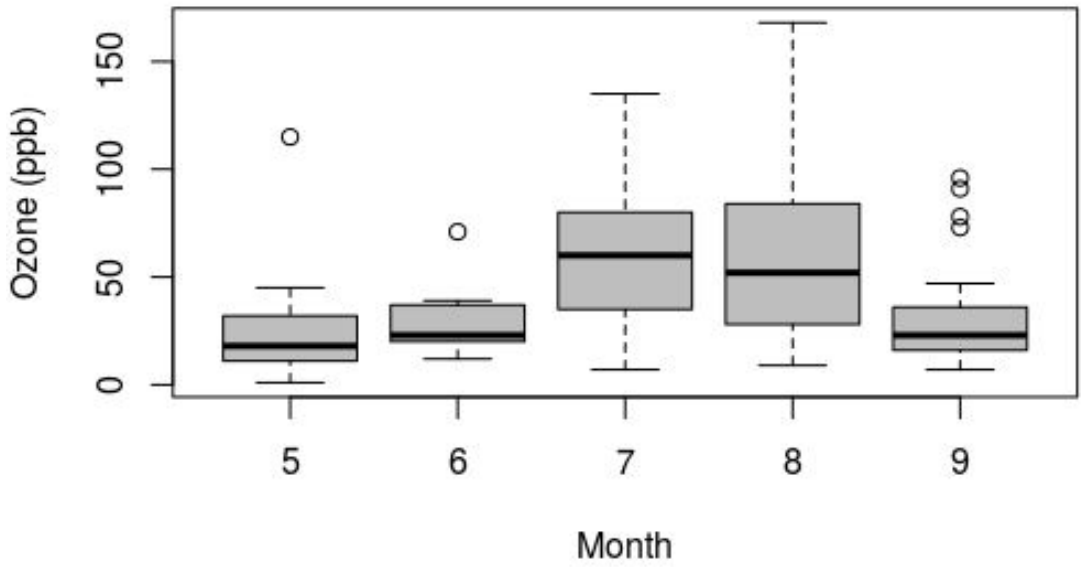
```
> table(airquality$Month)

 5  6  7  8  9
31 30 31 31 30
```

```
> head(airquality)
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
```



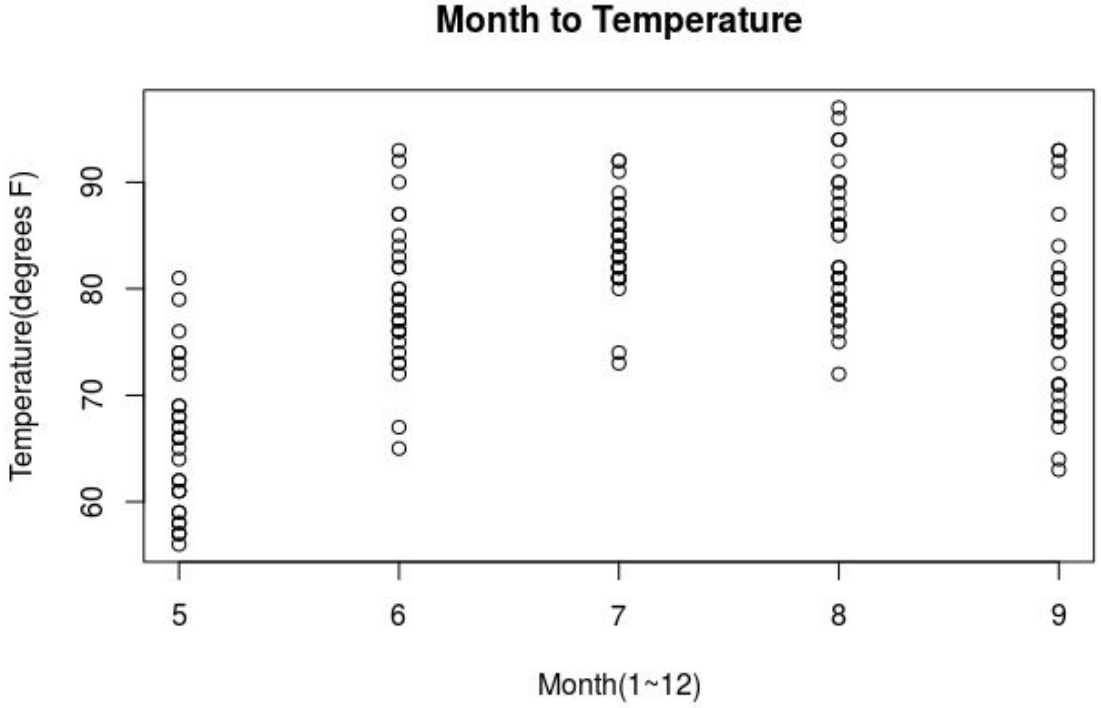
練習 一個關於空氣品質的資料集airquality

```
boxplot(formula = Ozone ~ Month, # Y ~ X (代表X和Y軸要放的數值)
data = airquality,     # 資料
xlab = "Month",        # X軸名稱
ylab = "Ozone (ppb)",  # Y軸名稱
 col ="gray")          # 顏色
```



練習 一個關於空氣品質的資料集airquality

```
plot(x=airquality$Month,
    y=airquality$Temp,
    main="Month to Temperature",
    xlab="Month(1~12)",
    ylab="Temperature(degrees F)")
```



Q: 能不能繪製攝氏溫度呢？

攝氏= (華氏-32)*5/9

```
airquality$CTemp=(airquality$Temp-32)*5/9 # 四捨五入至小數位數一位
```

練習 汽車油耗資料集

汽車油耗資料集(mtcars)是由Motor Trend US 雜誌所提供，總共有32部介於1973-74年出產的汽車其性能的資訊。

data(mtcars)

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

練習 汽車油耗資料集

汽車油耗資料集(mtcars)是由Motor Trend US 雜誌所提供，總共有32部介於1973-74年出產的汽車其性能的資訊。

? mtcars

```
A data frame with 32 observations on 11 variables.

  [, 1]  mpg   Miles/(US) gallon
  [, 2]  cyl   Number of cylinders
  [, 3]  disp  Displacement (cu.in.)
  [, 4]  hp    Gross horsepower
  [, 5]  drat  Rear axle ratio
  [, 6]  wt    Weight (1000 lbs)
  [, 7]  qsec  1/4 mile time
  [, 8]  vs    V/S
  [, 9]  am    Transmission (0 = automatic, 1 = manual)
  [,10]  gear  Number of forward gears
  [,11]  carb  Number of carburetors
```

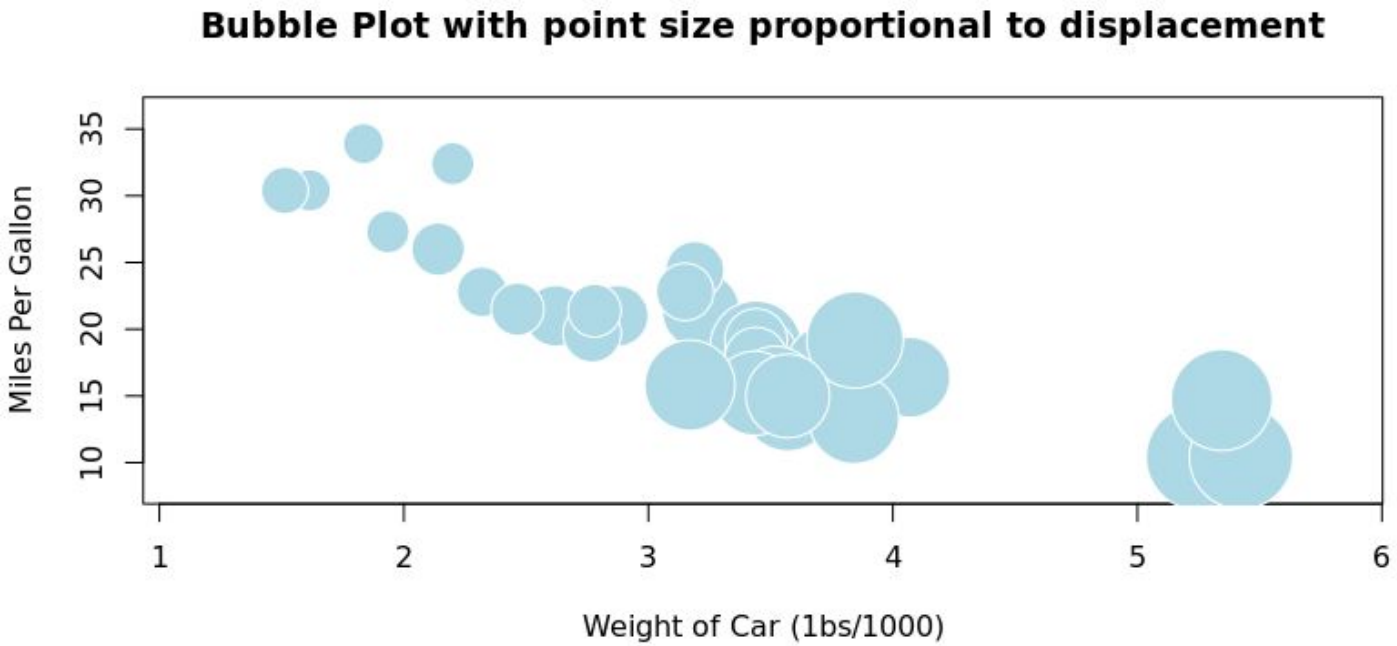
練習 汽車油耗資料集

data(mtcars)

mpg: Miles / (US) gallon 英哩/每加侖 (每加侖英里數)
cyl: Number of cylinders 汽缸數
disp: Displacement(cu.in.) 單汽缸排氣量
hp: Gross horsepower 馬力
wt: Weight(lb / 1000) 車重
am: Transmission(0 = automatic, 1 = manual) 自手排
gear: 檔數
carb: 化油器

練習 汽車油耗資料集

x軸代表車重，y軸代表每加侖英里數，氣泡大小代表引擎排氣量



練習

如何將鐵達尼號資料轉出成為 csv 檔案?

```
load("titanic.raw.rdata")
ls()
write.csv (titanic.raw,"titanic.csv", row.names=TRUE)
```

如何計算鐵達尼號所有搭船在船上的人數？船員人數？乘客人數?

```
t=read.csv('titanic.csv',skip=0,header=TRUE)
str(t)
summary(t)
table(t)
install.packages('arules','arulesViz')
```

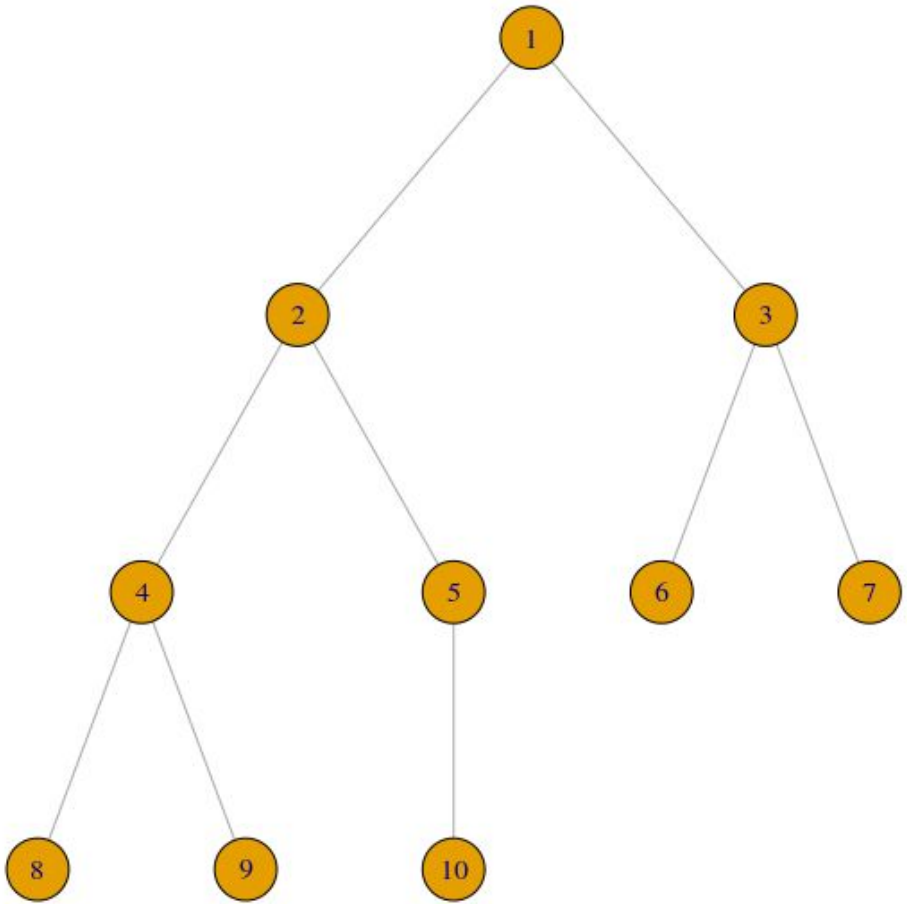
練習

如何如何繪製文字雲

```
#install.packages('wordcloud2')
#library(wordcloud2)
require('wordcloud2')
x=rev(c('中華跨境電商', '展盟工程', '巨匠雲科技', '惠康百貨', '淳康國際', '締盟',
'良福保全', '華碩電腦', '鴻齡國際', '大地國際', '有田世紀', '關貿網路', '昇恆昌',
'程高資訊', '三商家購', '巨匠電腦', '良興', '全聯實業', '神腦國際', '凌網科技'))
y=rev(c(1,1,1,1,1,1,1,1,1,2,2,2,3,3,4,4,4,6,6,8))
z=data.frame(x,y)
wordcloud2(z, size = 1,shape = 'circle')
```

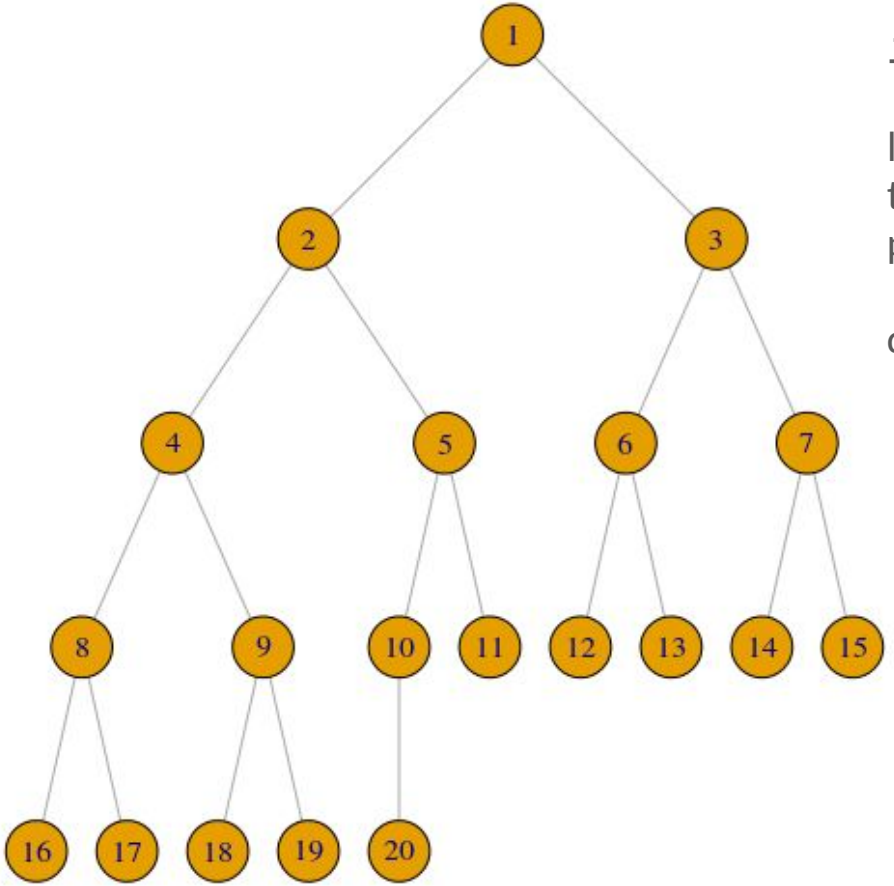
練習


有田世紀
大地國際
神腦國際
凌網科技
鴻齡國際
締盟
程高資訊
關貿網路
華碩電腦
淳康國際
良福保全
展盟工程
巨匠電腦
良興
昇恆昌
巨匠雲科技
惠康百貨
中華跨境電商



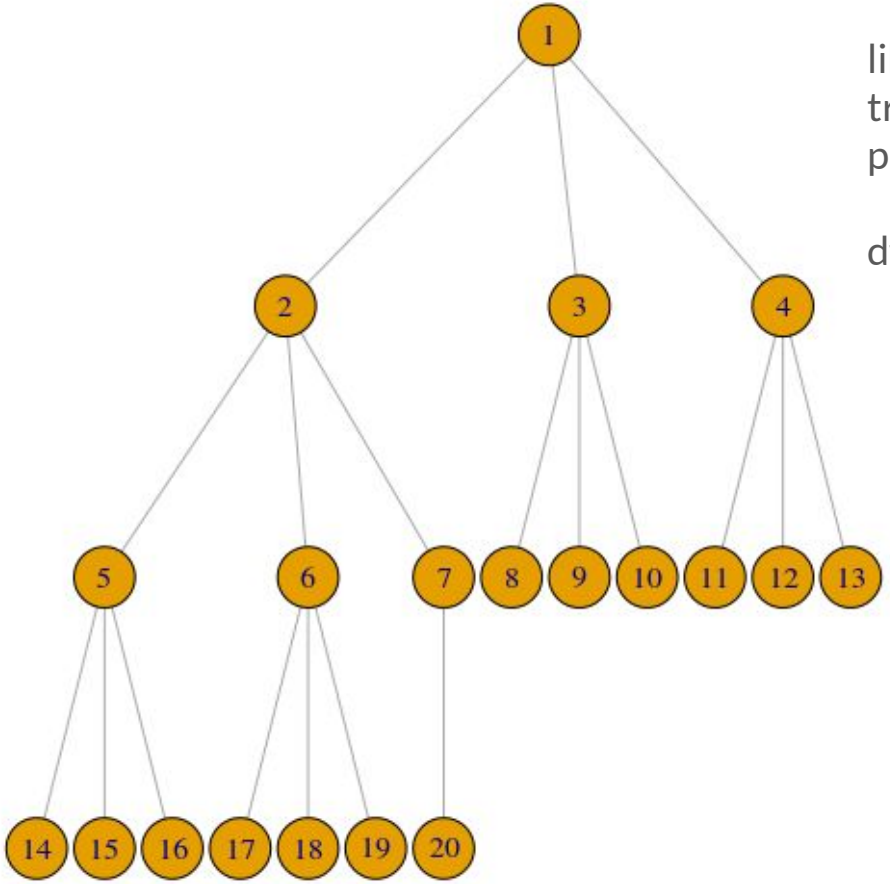
二元樹 (10個節點)

```
library(igraph)
tree=make_tree(n=10, children=2, mod=c('undirected'))
plot(tree, layout=layout_as_tree(tree,root=1))

dfs(tree,root=1,"out",TRUE, TRUE, TRUE, TRUE,TRUE)
$order
+ 10/10 vertices, from bf9baa4:
 [1]  1  2  4  8  9  5 10  3  6  7

$order.out
+ 10/10 vertices, from bf9baa4:
 [1]  8  9  4 10  5  2  6  7  3  1

$father
+ 10/10 vertices, from bf9baa4:
 [1] NA  1  1  2  2  3  3  4  4  5

$dist
 [1] 0 1 1 2 2 2 2 3 3 3
```



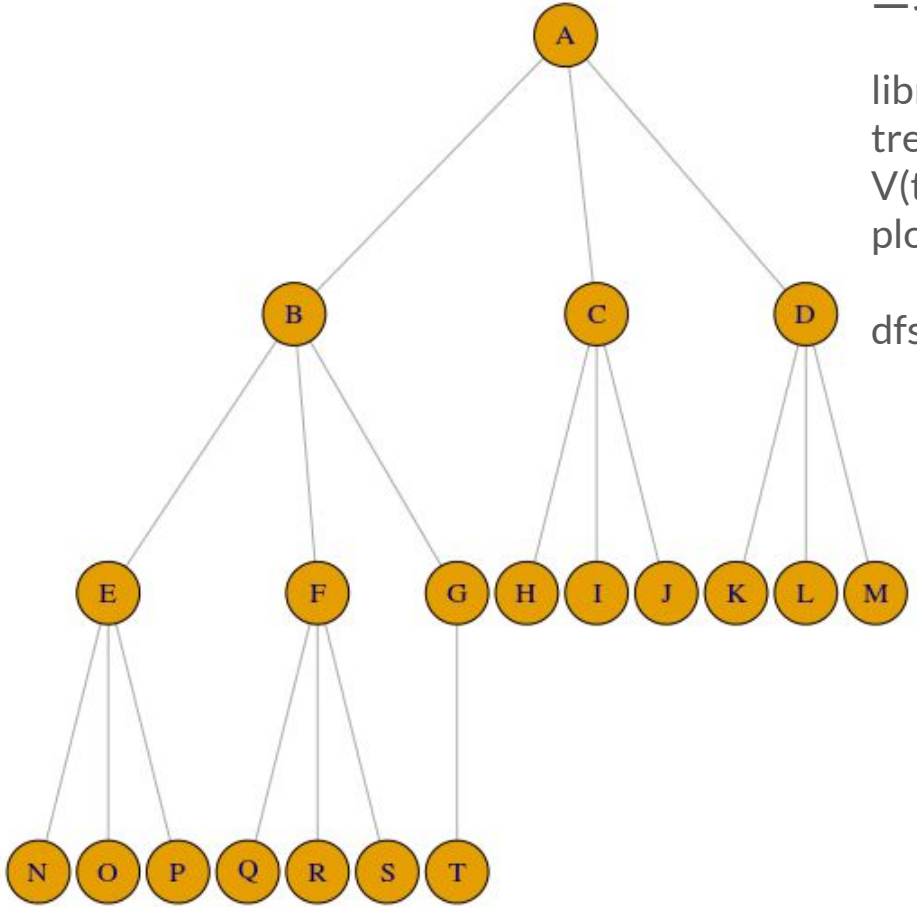
二元樹 (10個節點)

```
library(igraph)
tree=make_tree(n=20, children=2, mod=c('undirected'))
plot(tree, layout=layout_as_tree(tree,root=1))

dfs(tree,root=1,"out",TRUE, TRUE, TRUE, TRUE,TRUE)
```



三元樹 (20個節點)

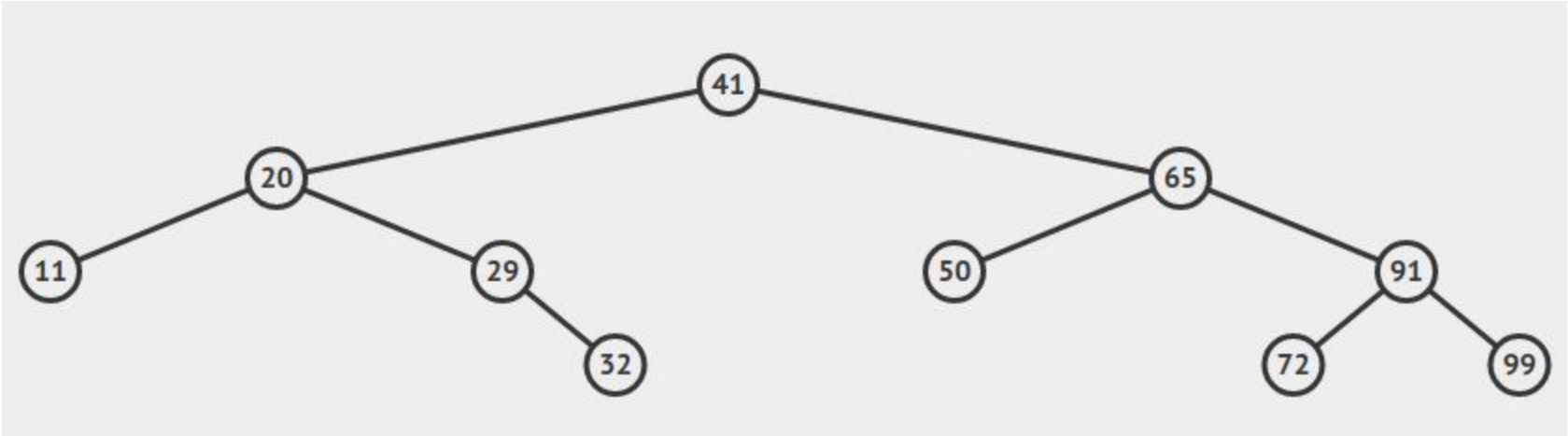
```
library(igraph)
tree=make_tree(n=20, children=3, mode=c('undirected'))
plot(tree, layout=layout_as_tree(tree,root=1))

dfs(tree,root=1,"out",TRUE, TRUE, TRUE, TRUE,TRUE)
```

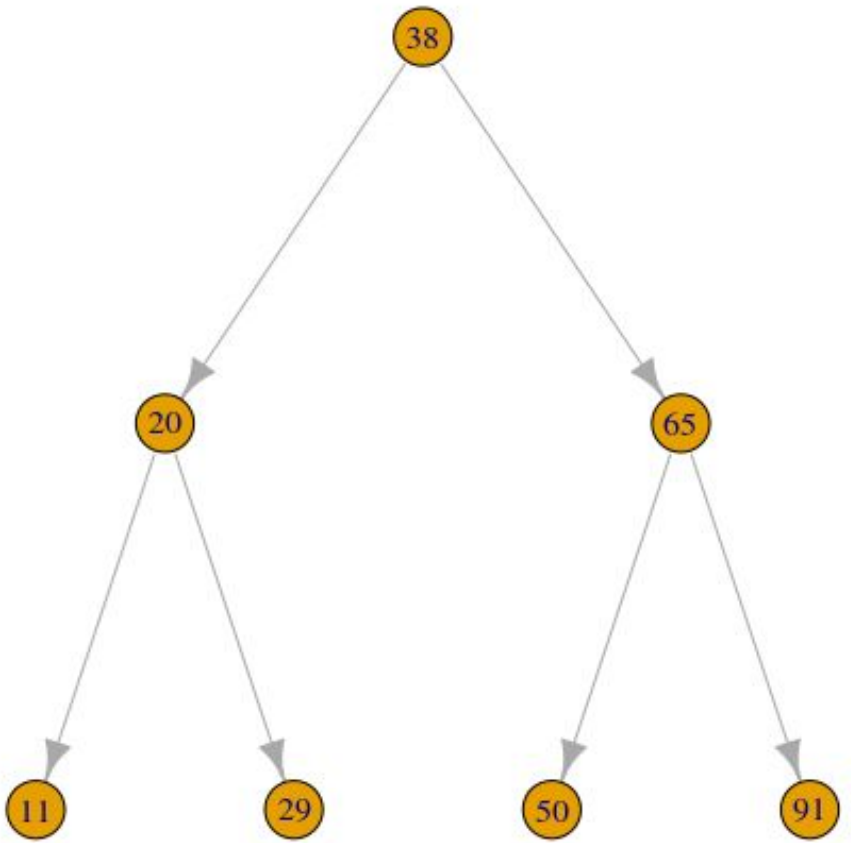


三元樹 (20個節點)

```
library(igraph)
tree=make_tree(n=20, children=3, mode=c('undirected'))
V(tree)$name=LETTERS[1:20]
plot(tree, layout=layout_as_tree(tree,root=1))

dfs(tree,root=1,"out",TRUE, TRUE, TRUE, TRUE,TRUE)
```

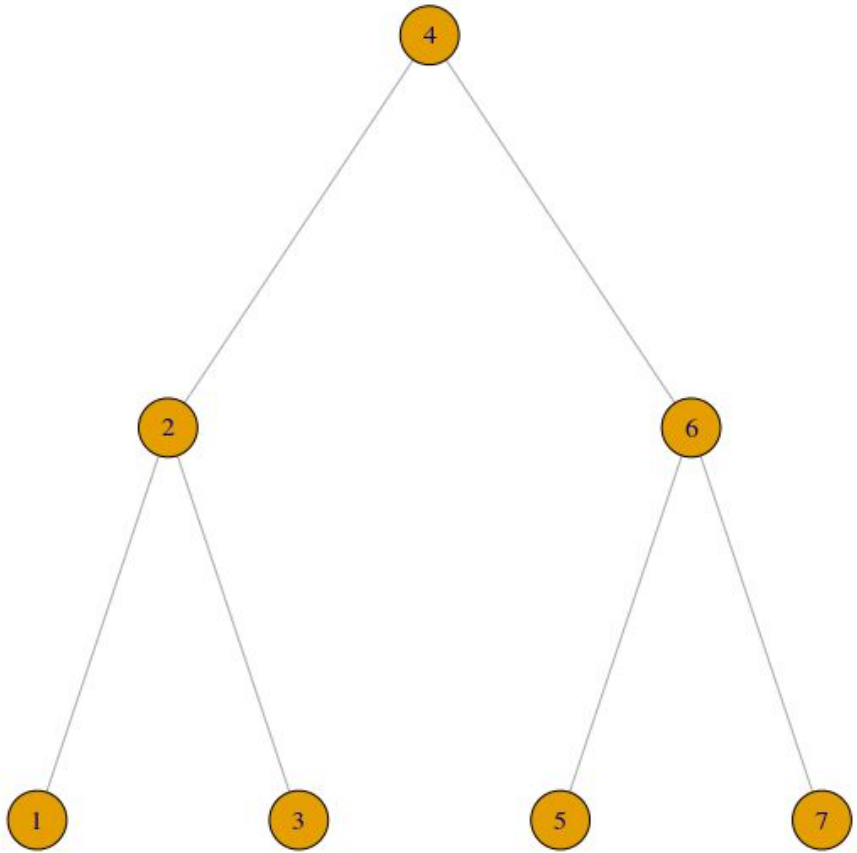
Binary Search Tree, AVL





二元搜尋樹 (7個節點)

```
library(igraph)
tree=make_tree(n=7, children=2, mod=c('undirected'))
V(tree)$name=c(38,20,65,11,29,50,91) # BST
plot(tree, layout=layout_as_tree(tree,root=1))

# 深度優先
dfs(tree,root=1,"out",TRUE, TRUE, TRUE, TRUE,TRUE)
$order
+ 7/7 vertices, named, from ea05e4e:
[1] 38 20 11 29 65 50 91
$order.out
+ 7/7 vertices, named, from ea05e4e:
[1] 11 29 20 50 91 65 38
$father
+ 7/7 vertices, named, from ea05e4e:
[1] 38 20 65 11 29 50 91
$dist
38 20 65 11 29 50 91
 0  1  1  2  2  2  2
```



二元搜尋樹 (7個節點)

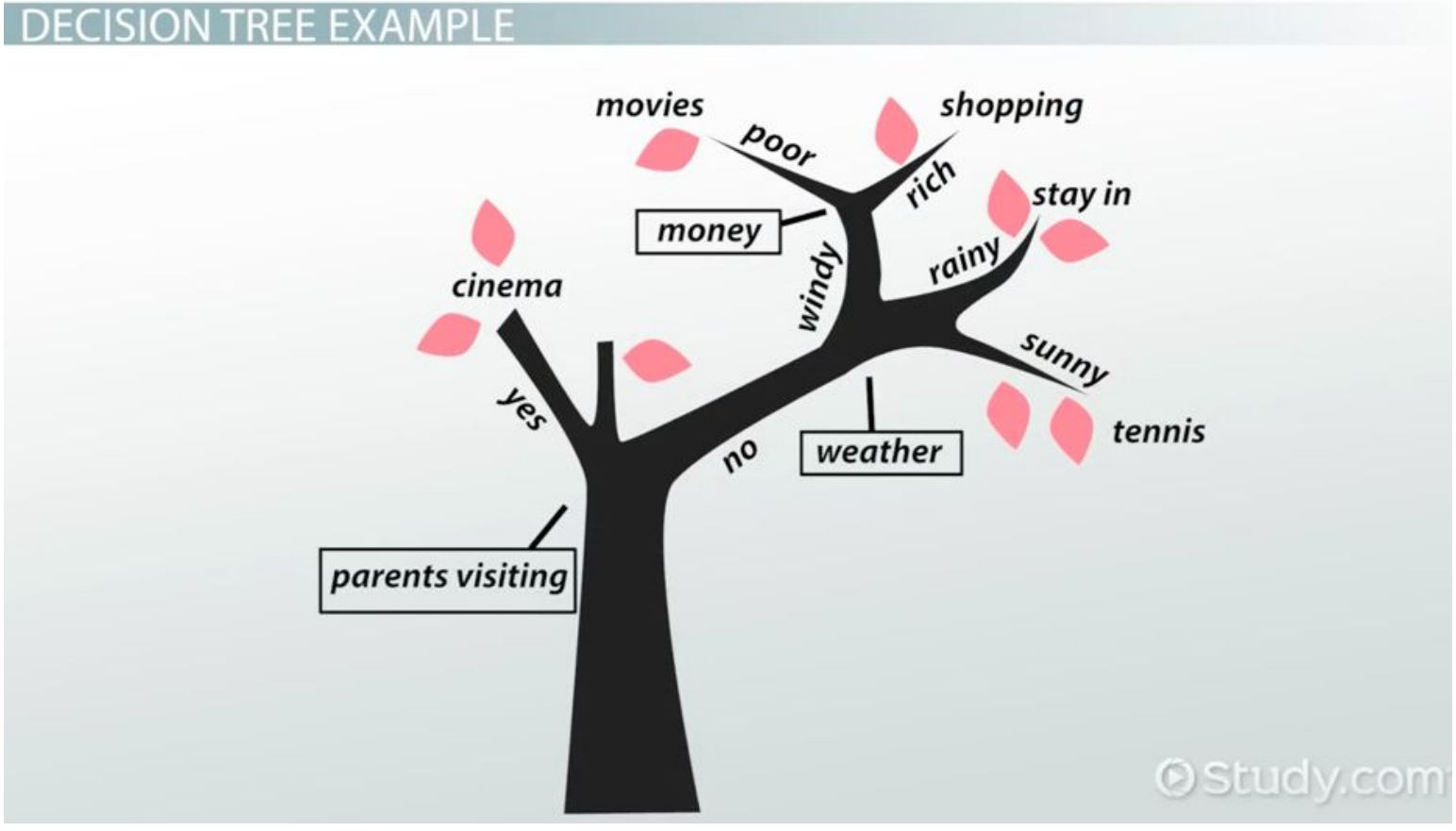
```
library(igraph)
tree=make_tree(n=7, children=2, mod=c('undirected'))
V(tree)$name=c(4,2,6,1,3,5,7) # BST
plot(tree, layout=layout_as_tree(tree,root=1))

# 深度優先
dfs(tree,root=1,"out",TRUE, TRUE, TRUE, TRUE,TRUE)
$order
+ 7/7 vertices, named, from 758c93e::
[1] 4 2 1 3 6 5 7
$order.out
+ 7/7 vertices, named, from 758c93e:
[1] 1 3 2 5 7 6 4
$father
+ 7/7 vertices, named, from 758c93e:
[1] 4 2 6 1 3 5 7
$dist
4 2 6 1 3 5 7
0 1 1 2 2 2 2
```

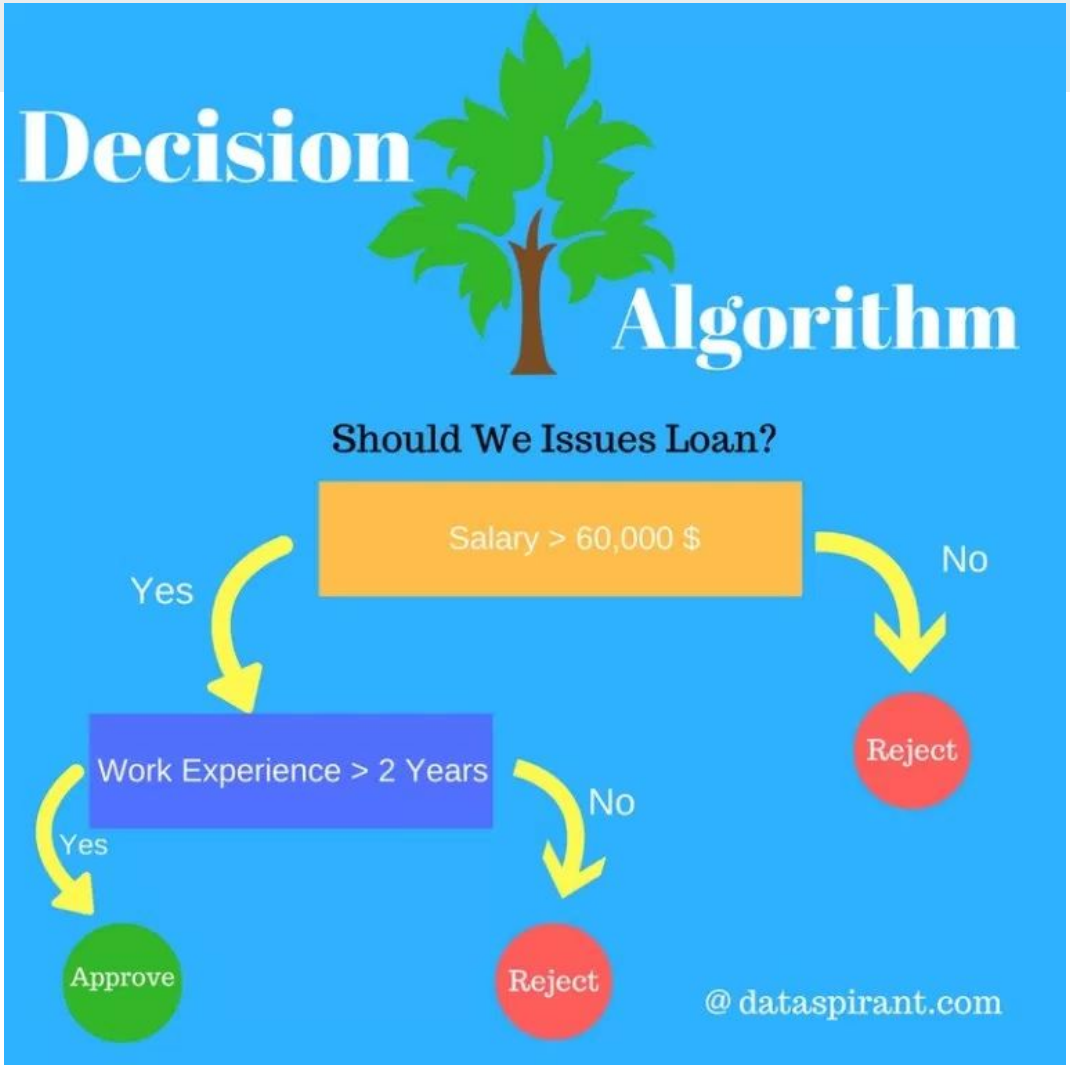
參考網站

- Introduction DataScience+ R Programming
- Introduction to data.tree
- Binary Search Tree, AVL
- AVL Tree Visualzation

決策樹 (Decision Tree)

What is a decision tree?

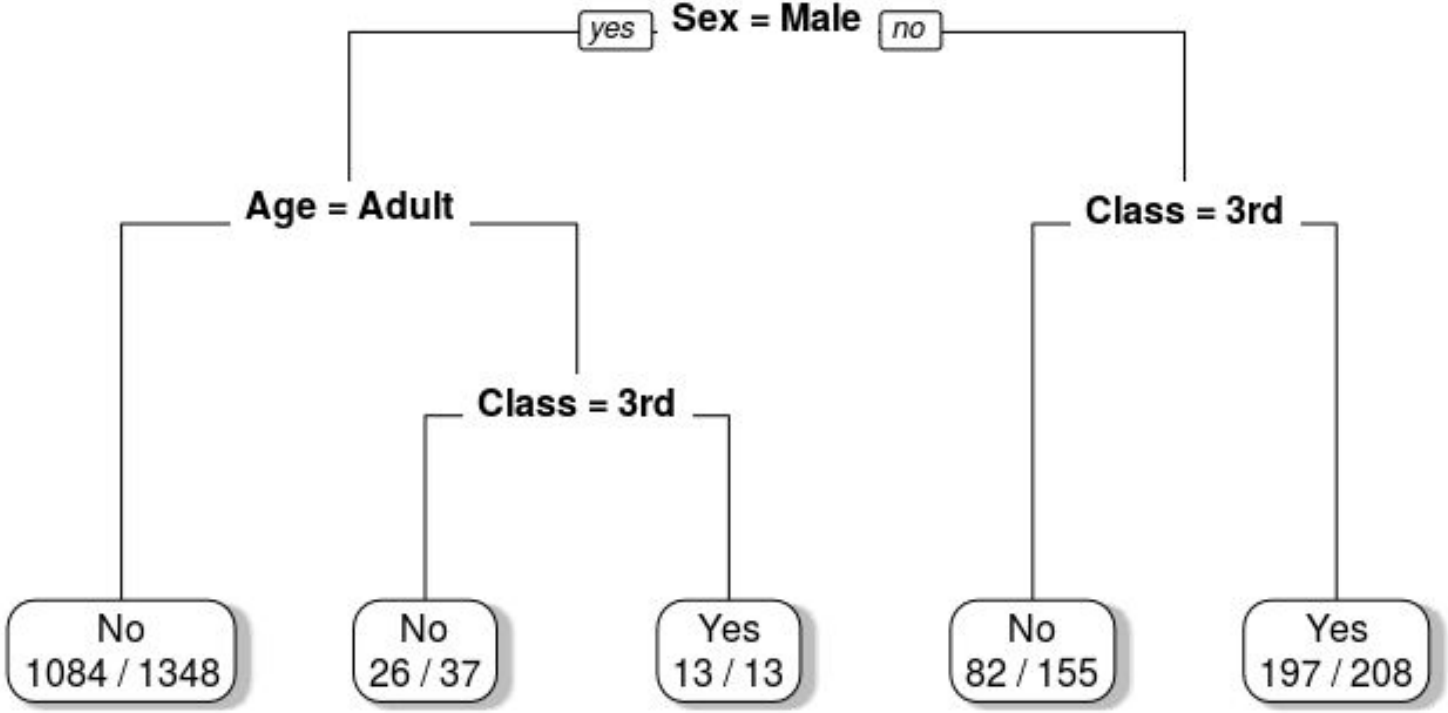




Decision Algorithm

鐵達尼號沈船事件分析

Class	Sex	Age	Survived
1st :325	Female: 470	Adult:2092	No :1490
2nd :285	Male :1731	Child: 109	Yes: 711
3rd :706			
Crew:885		乘客+船員 : 2201	



subset 子集合篩選

```
subset(titanic.raw,Class=='Crew' )
subset(titanic.raw,Class=='Crew' & Survived=='No' & Sex=='Male')
```

END