

109-1: EE4052

通識課程：

計算機程式設計

之旅

Computer Programming

Unit 01: 課程介紹

連 豐 力

臺大電機系

Sep 2020 - Jan 2020

大綱

- 為什麼要設計程式？
- 問題探索分析：程式設計的角色
- 為什麼選擇 R 語言，
作為本課程之程式設計語言？
- 課程討論主題
- 作業、報告與評分
- 常用的數據資料庫與參考書籍
- 一個簡單的範例

為什麼要設計程式？

■ 七個人的資料

	A	B	C	D
1	Name	Chinese	English	Math
2				
3	John	85	81	74
4	Mary	70	87	82
5	Henry	65	79	85
6	Alan	73	90	82
7	Frank	88	84	83
8	Helen	84	86	89
9	Angela	80	82	76

■ 如何計算平均值？

■ 用卡西歐計算機！

- $85 + 70 + 65 + \dots + 80 = 545$

- $545 / 7 = 77.86$

- ...



為什麼要設計程式？

■ 七個人的資料

	A	B	C	D
1	Name	Chinese	English	Math
2				
3	John	85	81	74
4	Mary	70	87	82
5	Henry	65	79	85
6	Alan	73	90	82
7	Frank	88	84	83
8	Helen	84	86	89
9	Angela	80	82	76

■ 如何計算平均值？

■ 用卡西歐計算機！

- $85 + 70 + 65 + \dots + 80 = 545$

- $545 / 7 = 77.86$

- ...

■ 100人的資料！

	A	B	C	D
1	Name	Chinese	English	Math
2	Agnes	70	19	65
3	Aiolos	84	71	56
4	Alan	39	79	19
5	Alexis	32	76	33
6	Alice	60	90	82
7	Alina	31	81	71
8	Ambrose	78	10	84
9	Amelia	77	82	76
10	Angela	61	48	76
11	Ann	79	60	46
12	Anna	81	29	79
13	April	33	60	65
14	Arabella	76	60	38
15	Arnold	69	33	83
16	Arthur	71	39	83
17	Aveza	70	56	33
18	Avis	66	21	78
19	Bethany	83	57	19
20	Bill	67	13	69
21	Bridget	75	39	70
22	Betsy	39	81	83
23	Charlie	0	0	59
24	Caroline	87	69	71
25	Chancellor	82	44	57
26	Cindy	19	28	39
27	Christopher	62	10	45
28	Daniel	71	46	69
29	Debbie	60	36	84
30	David	83	78	30

為什麼要設計程式？

- 100人的資料
- 如何計算平均值？
- 用 Excel 來計算：

D2		=AVERAGE(D3:D102)					
	A	B	C	D	E	F	
1	Name	Chinese	English	Math			
2	Average	57.98	51.86	59.38			
3	Agnes	70	19	65			
4	Aiolos	84	71	56			
5	Alan	39	79	19			
6	Alexis	32	76	33			
7	Alice	60	90	82			
8	Alina	31	81	71			
9	Ambrose	78	10	84			
10	Amelia	77	82	76			
11	Angela	61	48	76			
12	Ann	79	60	46			
13	Anna	81	29	79			
14	April	33	60	65			
15	Arabella	76	60	38			
16	Arnold	69	33	83			

	A	B	C	D	
1	Name	Chinese	English	Math	
2	Agnes	70	19	65	
3	Aiolos	84	71	56	
4	Alan	39	79	19	
5	Alexis	32	76	33	
6	Alice	60	90	82	
7	Alina	31	81	71	
8	Ambrose	78	10	84	
9	Amelia	77	82	76	
10	Angela	61	48	76	
11	Ann	79	60	46	
12	Anna	81	29	79	
13	April	33	60	65	
14	Arabella	76	60	38	
15	Arnold	69	33	83	
16	Arthur	71	39	83	
17	Aveza	70	56	33	
18	Avis	66	21	78	
19	Bethany	83	57	19	
20	Bill	67	13	69	
21	Bridget	75	39	70	
22	Betsy	39	81	83	
23	Charlie	0	0	59	
24	Caroline	87	69	71	
25	Chancellor	82	44	57	
26	Cindy	19	28	39	
27	Christopher	62	10	45	
28	Daniel	71	46	69	
29	Debbie	60	36	84	
30	David	83	78	30	

為什麼要設計程式？

- 100人的資料
- 如何計算平均值？

- 寫一個程式依序計算：

- sum <- 0
- for(k in 1:100)
 - { sum <- sum + Math[k] }
- avg <- sum/100
- mean(Math)
- mean(Math[1:100])
- mean(Math[11:20])

	A	B	C	D
1	Name	Chinese	English	Math
2	Agnes	70	19	65
3	Aiolos	84	71	56
4	Alan	39	79	19
5	Alexis	32	76	33
6	Alice	60	90	82
7	Alina	31	81	71
8	Ambrose	78	10	84
9	Amelia	77	82	76
10	Angela	61	48	76
11	Ann	79	60	46
12	Anna	81	29	79
13	April	33	60	65
14	Arabella	76	60	38
15	Arnold	69	33	83
16	Arthur	71	39	83
17	Aveza	70	56	33
18	Avis	66	21	78
19	Bethany	83	57	19
20	Bill	67	13	69
21	Bridget	75	39	70
22	Betsy	39	81	83
23	Charlie	0	0	59
24	Caroline	87	69	71
25	Chancellor	82	44	57
26	Cindy	19	28	39
27	Christopher	62	10	45
28	Daniel	71	46	69
29	Debbie	60	36	84
30	David	83	78	30

為什麼要設計程式？

- 100人的資料
- 如何計算平均值？

- 寫一個程式依序計算：

- `sum <- 0`
- `for( k in 1:100)`
- `{ sum <- sum + Math[k] }`
- `avg <- sum/100`

- 複雜的，繁瑣的計算或決策工作

- 大量的，重複性的計算或決策工作

	A	B	C	D
1	Name	Chinese	English	Math
2	Agnes	70	19	65
3	Aiols	84	71	56
4	Alan	39	79	19
5	Alexis	32	76	33
6	Alice	60	90	82
7	Alina	31	81	71
8	Ambrose	78	10	84
9	Amelia	77	82	76
10	Angela	61	48	76
11	Ann	79	60	46
12	Anna	81	29	79
13	April	33	60	65
14	Arabella	76	60	38
15	Arnold	69	33	83
16	Arthur	71	39	83
17	Aveza	70	56	33
18	Avis	66	21	78
19	Bethany	83	57	19
20	Bill	67	13	69
21	Bridget	75	39	70
22	Betsy	39	81	83
23	Charlie	0	0	59
24	Caroline	87	69	71
25	Chancellor	82	44	57
26	Cindy	19	28	39
27	Christopher	62	10	45
28	Daniel	71	46	69
29	Debbie	60	36	84
30	David	83	78	30

為什麼要設計程式？

■ 100人的資料

	A	B	C	D
1	Name	Chinese	English	Math
2	Elinor	60	86	89
3	Matthew	65	28	87
4	Elizabeth	7	79	85
5	Ambrose	78	10	84
6	Debbie	60	36	84
7	Arnold	69	33	83
8	Arthur	71	39	83
9	Betsy	39	81	83
10	Dionys	60	84	83
11	Erro	78	38	83
12	John	80	71	83
13	Joshua	67	76	83
14	Olive	79	33	83
15	Alice	60	90	82
16	Humphrey	60	87	82
17	Zachariah	68	52	82
18	Edgar	10	44	81
19	Issac	57	32	81
20	Richard	22	77	81
21	Anna	81	29	79
22	Judy	68	19	79
23	Avis	66	21	78
24	Harry	31	52	78
25	Leonardo	47	19	78
26	Grace	78	66	77
27	Amelia	77	82	76
28	Angela	61	48	76
29	Mark	52	30	76
30	Florence	60	81	74

■ 如何找出 最大值 或 最小值？

	A	B	C	D
1	Name	Chinese	English	Math
2	Caroline	87	69	71
3	Myles	87	71	57
4	Sandy	87	71	57
5	Westwood	86	60	39
6	Gilbert	84	46	67
7	Aiolos	84	71	56
8	Vallen	84	66	29
9	Roger	83	74	71
10	Sunny	83	22	52
11	Edward	83	0	47
12	Jay	83	22	32
13	David	83	78	30
14	Bethany	83	57	19
15	Chancellor	82	44	57
16	Zipporah	82	71	52
17	Joyce	82	57	19
18	Anna	81	29	79
19	Emma	81	67	60
20	John	80	71	83
21	Olive	79	33	83
22	Francis	79	52	70
23	William	79	81	56
24	Ann	79	60	46
25	Ambrose	78	10	84
26	Erro	78	38	83
27	Grace	78	66	77
28	Henry	78	71	10
29	Amelia	77	82	76
30	Marcy	77	22	31

	A	B	C	D
1	Name	Chinese	English	Math
2	Agnes	70	19	65
3	Aiolos	84	71	56
4	Alan	39	79	19
5	Alexis	32	76	33
6	Alice	60	90	82
7	Alina	31	81	71
8	Ambrose	78	10	84
9	Amelia	77	82	76
10	Angela	61	48	76
11	Ann	79	60	46
12	Anna	81	29	79
13	April	33	60	65
14	Arabella	76	60	38
15	Arnold	69	33	83
16	Arthur	71	39	83
17	Aveza	70	56	33
18	Avis	66	21	78
19	Bethany	83	57	19
20	Bill	67	13	69
21	Bridget	75	39	70
22	Betsy	39	81	83
23	Charlie	0	0	59
24	Caroline	87	69	71
25	Chancellor	82	44	57
26	Cindy	19	28	39
27	Christopher	62	10	45
28	Daniel	71	46	69
29	Debbie	60	36	84
30	David	83	78	30

為什麼要設計程式？

- 100人的資料
- 如何找出最大值或最小值？
- 寫一個程式，先把資料依照大小排序：


```

bubble_sort = function( array ) {
  count = 0
  while(1) {
    count_swaps = 0
    for ( j in 1 : (length(array) - 1 - count) ) {
      if (array[j] > array[j + 1]) {
        s = array[j]
        array[j] = array[j+1]
        array[j+1] = s
        count_swaps = count_swaps + 1
      }
    }
    count = count + 1
    if(count_swaps == 0) break
  }
  array
}

```
- Math_sort <- sort(Math)

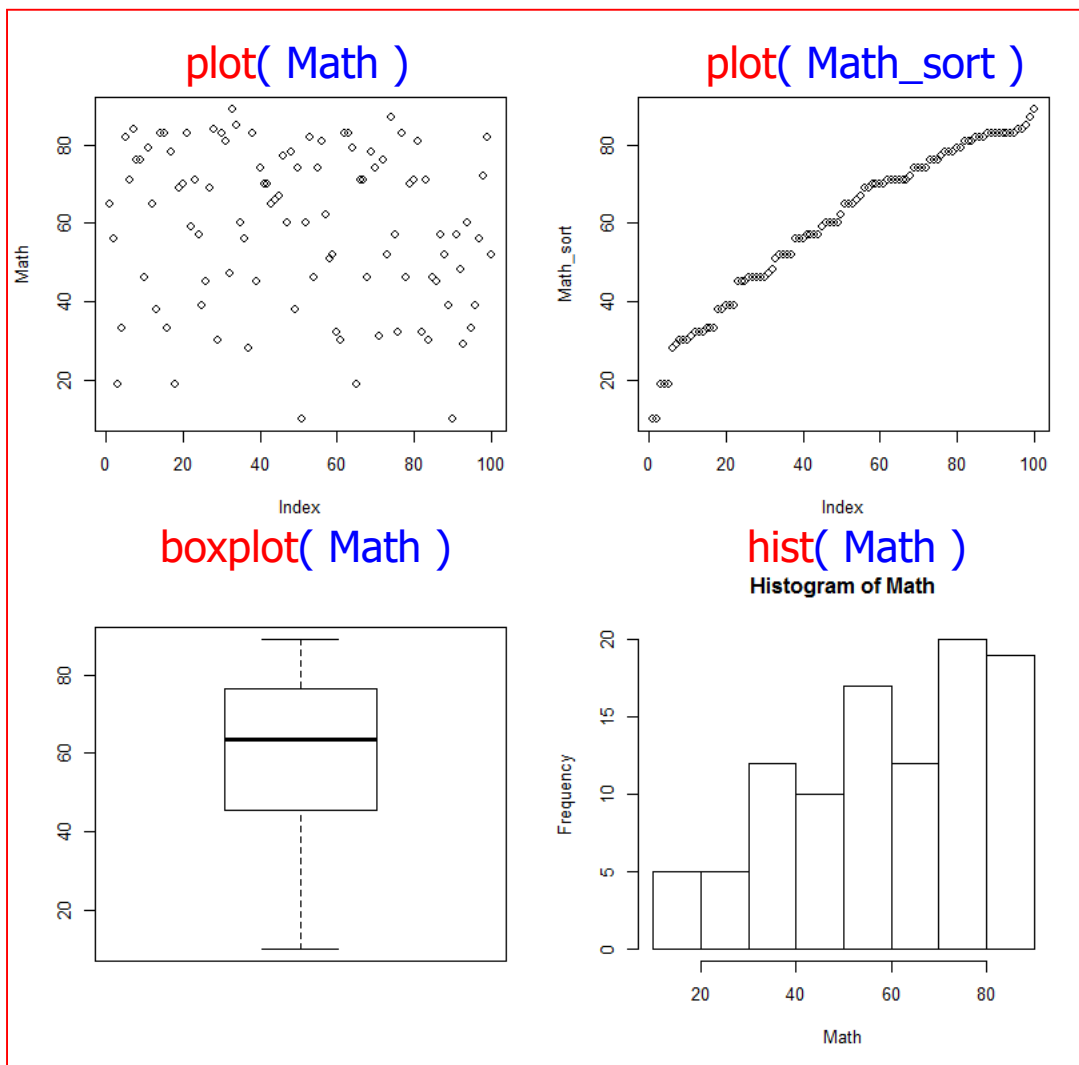
	A	B	C	D
1	Name	Chinese	English	Math
2	Agnes	70	19	65
3	Aiolos	84	71	56
4	Alan	39	79	19
5	Alexis	32	76	33
6	Alice	60	90	82
7	Alina	31	81	71
8	Ambrose	78	10	84
9	Amelia	77	82	76
10	Angela	61	48	76
11	Ann	79	60	46
12	Anna	81	29	79
13	April	33	60	65
14	Arabella	76	60	38
15	Arnold	69	33	83
16	Arthur	71	39	83
17	Aveza	70	56	33
18	Avis	66	21	78
19	Bethany	83	57	19
20	Bill	67	13	69
21	Bridget	75	39	70
22	Betsy	39	81	83
23	Charlie	0	0	59
24	Caroline	87	69	71
25	Chancellor	82	44	57
26	Cindy	19	28	39
27	Christopher	62	10	45
28	Daniel	71	46	69
29	Debbie	60	36	84
30	David	83	78	30

為什麼要設計程式？

■ 100人的資料

- `plot( Math )`
- `plot( Math_sort )`
- `boxplot( Math )`
- `hist( Math )`

■ 如何找出 最大值 或 最小值？



為什麼要設計程式？

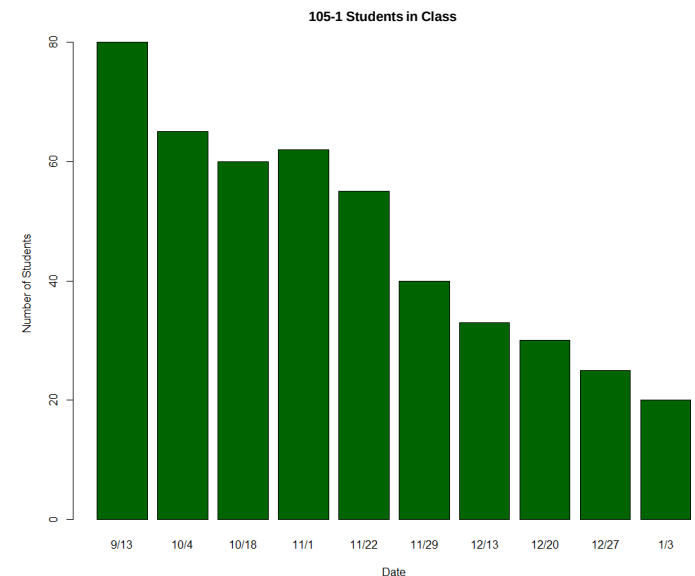
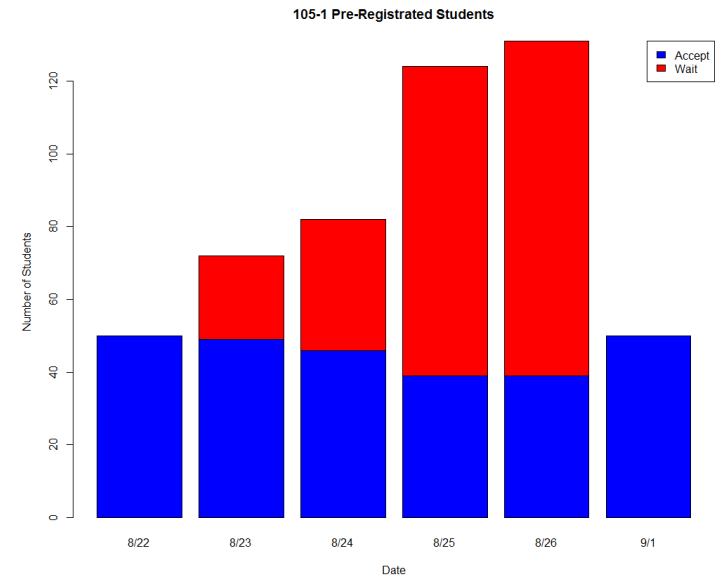
計算機程式設計 – 2020F

Unit 01: 課程介紹

Feng-Li Lian at NTU-EE

■ 105-1 的上課人數分析

Date	Accept	Wait
8/22	50	0
8/23	49	23
8/24	46	36
8/25	39	85
8/26	39	92
9/1	50	0
9/20	65	0
9/22	70	0
9/23	69	0
9/26	64	0
10/9	65	0
9/13	80	0
10/4	60	0
10/18	60	0
11/1	62	0
11/22	55	0
11/29	40	0
12/13	30	0
12/20	30	0
12/27	25	0
1/3	20	0



為什麼要設計程式？

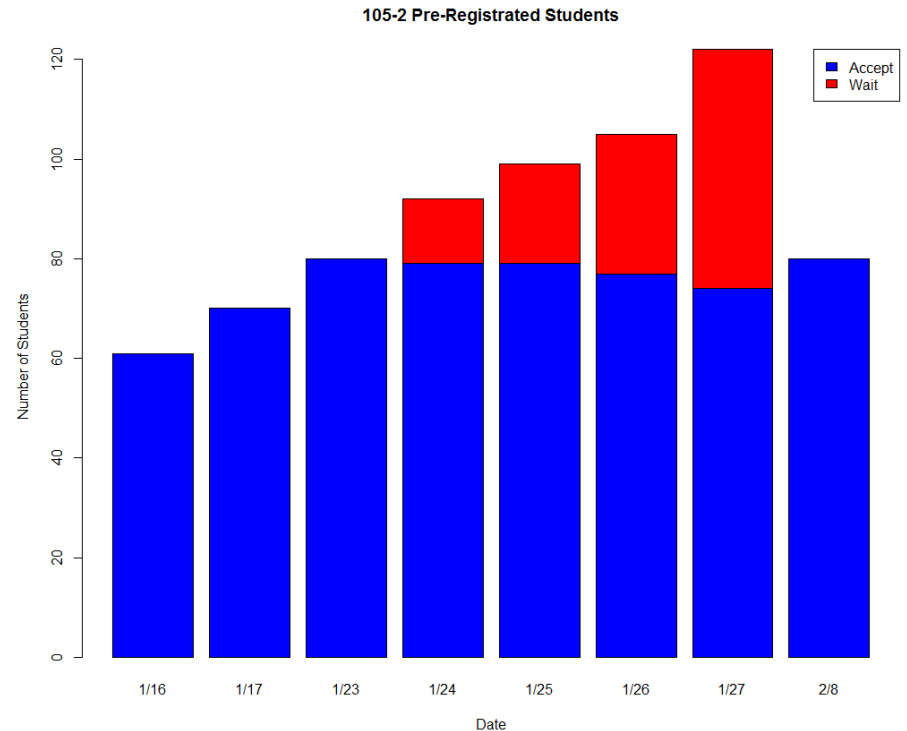
計算機程式設計 – 2020F

Unit 01: 課程介紹

Feng-Li Lian at NTU-EE

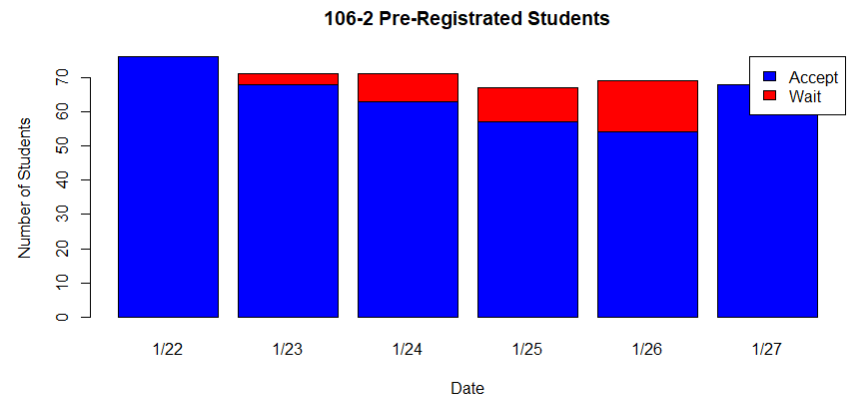
■ 105-2 的初選人數分析

Date	Accept	Wait
1/16	61	0
1/17	70	0
1/23	80	0
1/24	79	13
1/25	79	20
1/26	77	28
1/27	74	48
2/8	80	0



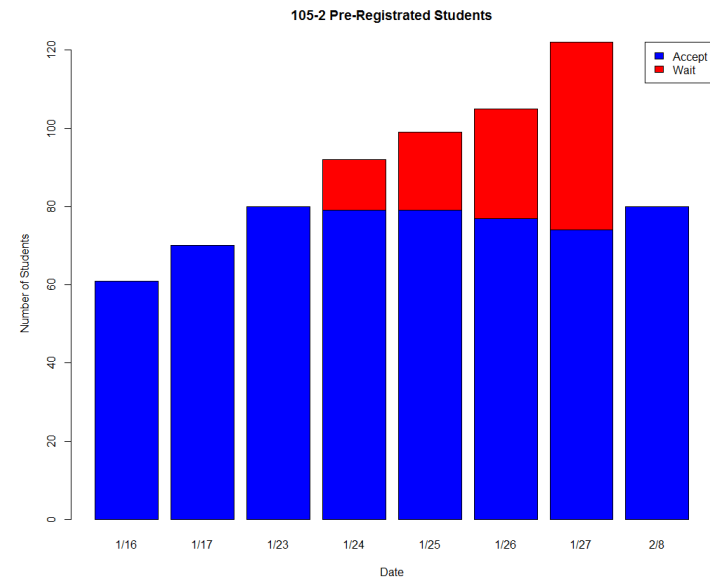
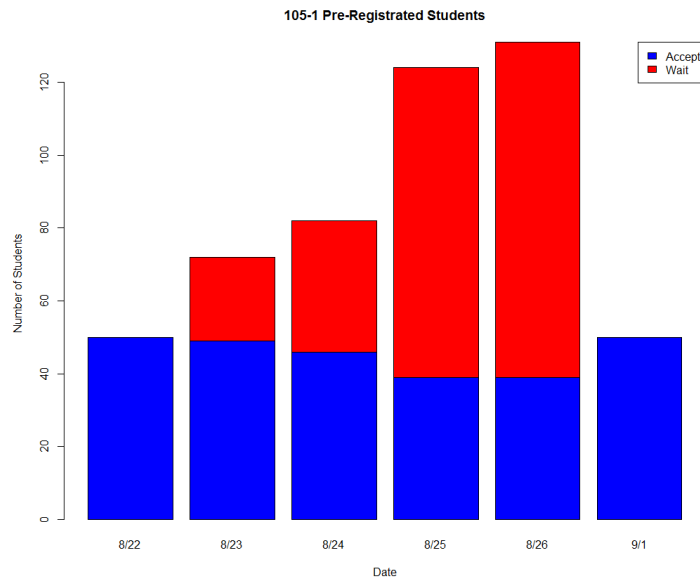
■ 106-2 的初選人數分析

Date	Accept	Wait
1/22	76	0
1/23	68	3
1/24	63	8
1/25	57	10
1/26	54	15
1/27	68	0



為什麼要設計程式？

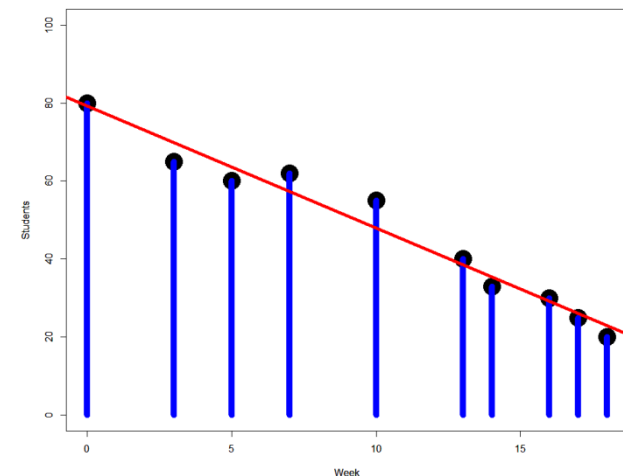
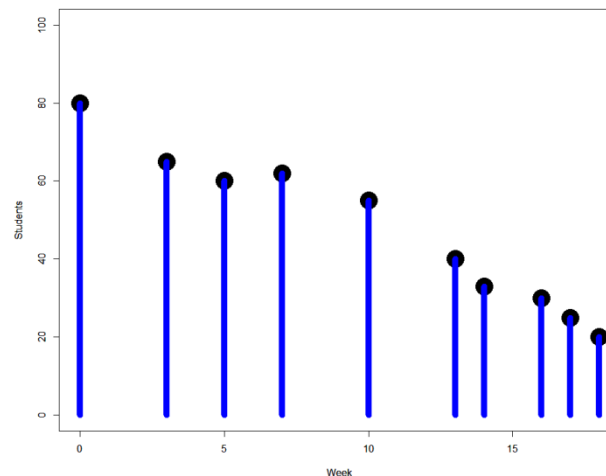
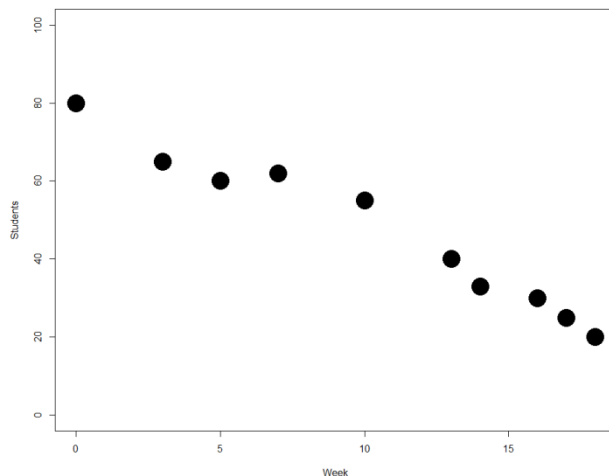
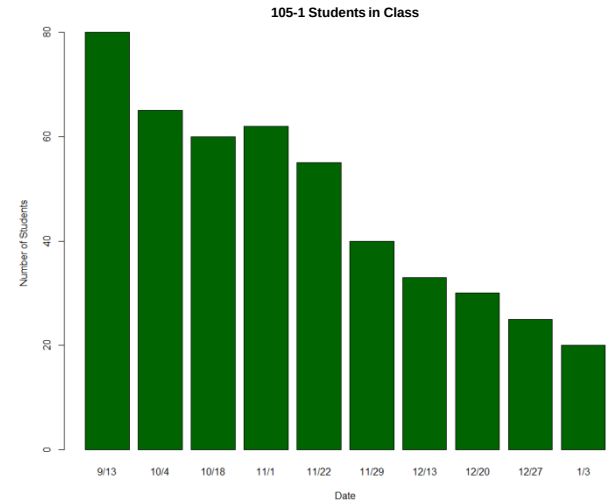
- mydata <- read.csv("StudentNumber.csv")
- From <- 1; To <- 6;
- barplot(rbind(mydata[From:To, 2], mydata[From:To, 3]), names.arg = mydata[From:To,1], xlab = "Date", ylab = "Number of Students", col = c("blue", "red"), main = "105-1 Pre-Registrated Students")
- legend(x = "topright", (c("Accept", "Wait")), fill = c("blue", "red"))
- From <- 20; To <- 27;
- barplot(rbind(mydata[From:To, 2], mydata[From:To, 3]), names.arg = mydata[From:To,1], xlab = "Date", ylab = "Number of Students", col = c("blue", "red"), main = "105-2 Pre-Registrated Students")
- legend(x = "topright", (c("Accept", "Wait")), fill = c("blue", "red"))



為什麼要設計程式？

■ 105-1 的上課人數分析

- `d1 <- mydata[ From:To, 2]`
- `d0 <- c( 0, 3, 5, 7, 10, 13, 14, 16, 17, 18)`
- `plot( d0, d1, type = "p", cex = 4 , ylim = c(0,100), pch = 16,`
`xlab="Week", ylab="Students" )` # 黑色點
- `plot( d0, d1, type = "h", lwd=10 , ylim = c(0,100), col = "blue",`
`xlab="Week", ylab="Students" )` # 藍色線段
- `fit1 <- lm( d1 ~ d0 )`
- `abline( fit1, col = "red", lwd = 5)` # 紅色曲線



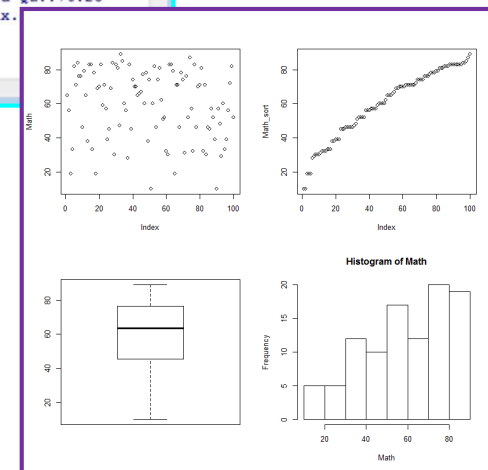
問題

```
R Console
> summary(data)
      Name      Chinese      English      Math
Agnes : 1      Min.   : 0.00      Min.   : 0.00      Min.   :10.00
Aiolos : 1     1st Qu.:38.75     1st Qu.:32.00     1st Qu.:45.75
Alan   : 1     Median :66.00     Median :57.00     Median :63.50
Alexis : 1     Mean   :57.98     Mean   :51.86     Mean   :59.38
Alice  : 1     3rd Qu.:78.00     3rd Qu.:71.00     3rd Qu.:76.25
Alina  : 1     Max.   :87.00     Max.   :90.00     Max.   :
(Other):94
```

算術平均數 $M(\bar{X}) = \frac{1}{n} (x_1 + x_2 + \dots + x_n) = \frac{1}{n} \sum_{i=1}^n x_i$

標準差 $S = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{X})^2} = \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2 - \bar{X}^2}$

- `summary( Math )`
- `sort ( Math )`
- `boxplot( Math )`
- `hist( Math )`



理論
(公式)

計算機
程式設計

分析：數據化

分析：視覺化

關於這門課

■ 課程概述：

- 在日常生活與工作當中，
- 我們常常面對：
要**花多少錢**去買書，買衣服，旅遊娛樂休閒，
- 需要瞭解：
花多少時間在交通，上課，與遊玩，
- 或者是擔心：
這個學期修課過程中，**作業分數**，**考試分數**等，
- 以及，如何**影響**最後的**學期成績**，
- 有時候，還會想要知道**別人的分數**，
- 以及，這些分數對**我的分數**的**影響**。

關於這門課

■ 課程概述：

- 在這些許許多多的過程中，
- 我們經常面對許多的數據，
- 有時候，這些數據的數量是**很多**或者**繁雜**，
- 需要進行一些**計算**，以便於瞭解這些數據所代表的**意義**。
- 有時候，就是一些**重複性**出現的數據，
- 因而需要一次又一次的進行**相同的計算**等工作。

關於這門課

■ 課程概述：

- 面對這些繁多複雜或者重複性產生的的數據，
- 最好的方法是：
能有一個電腦程式可以自動產生我們所需要的資訊。
- 所以，在這門課之中，
我們將討論一些基本的程式設計概念與作法，
- 讓你們可以重複地，自動地，處理數據，
以便於產生有意義的資訊。
- 本課程所使用的程式語言是 R，
這是一個比較直覺式的簡單程式設計語言。

關於這門課

■ 課程規劃：

1. 首先，我們會介紹幾個代表性的數據，

例如：

1000人的體重與身高數據，

150筆鳶尾花(iris)的：

花萼長度，花萼寬度，花瓣長度，花瓣寬度等數據。

786筆超市購買產品與個人基本資料等。

關於這門課

■ 課程規劃：

2. 接著，我們會針對這些**數據資料**進行**分析**，
以瞭解這些數據背後所代表的**資訊**。
這個過程需要數據分析的**理論公式**，
以及**設計程式**的基本工作。

3. 第三階段，
則是由修課的同學提供一些跟您們**有關的數據**，
然後，我們共同來討論分析這些數據所需的**理論公式**，
最後，我們在一起**設計程式**
來處理這些數據產生有意義的資訊。

關於這門課

- 所以，在這門課程之中，所討論的內容，：
 - 會著重於使用一個簡單易用且免費的電腦軟體
 - 設計一些簡單的程式
 - 計算數據的工作
 - 分析數據的工作
- 但是，在這門課程之中，我們不會強調：
 - 討論嚴謹的程式設計流程，撰寫複雜的程式碼
 - 使用許多統計分析的工具
 - 討論所使用的計算公式背後的理論與原理

- 課程網站：
 - NTU-COOL
 - <http://homepage.ntu.edu.tw/~fengli/Teaching/Computer/>
- 所有課程訊息與講義等，會同步公告於所有課程網站。
- 過去的課程資料：
 - 105-1 課程資料：
http://homepage.ntu.edu.tw/~fengli/Teaching/Computer/index_2016F.html
 - 105-2 課程資料：
http://homepage.ntu.edu.tw/~fengli/Teaching/Computer/index_2017S.html
 - 106-2 課程資料：
http://homepage.ntu.edu.tw/~fengli/Teaching/Computer/index_2018S.html

課程討論主題

■ 案例討論與分析

- 鳶尾花 (iris) 資料集
- 第三季 車險 投保人資料 (Insurance)
- 植物 對 二氧化碳之攝取的數據 (CO2)

■ 把數據變成有用的資料與資訊的分析

- 平均值，中位數，標準差，範圍，四分位數

■ 探索資料內涵的圖形分析

- 長條圖，累積分布圖，箱形圖 盒鬚圖，橫條圖，點陣圖，圓形圖

■ 程式設計基本邏輯

- 資料結構，流程圖，輸入與輸出

■ R 語言基本語法

- 資料表示法，運算與函數
- 陣列，向量與矩陣
- 數據輸入輸出與處理

課程目標

■ 課程目標：

- 瞭解可以 "分析數據" 與 "繪製圖形" 的程式
- 設計可以 "分析數據" 與 "繪製圖形" 的程式

■ 給只會用基本電腦操作的同學：

- 一點點邏輯分析的概念
- 一點點設計程式的概念
- 提供自主學習的資源

■ 預期成果：

- 讓您有能力與資源，可以設計一些程式，
來幫助您課業的學習或簡化您的生活上的工作。

課程要求

- 請儘量帶電腦來，
(沒帶電腦來，其實也沒關係，可以回去再練習)

- 請自己練習打一些程式碼，
自己整理一下寫了什麼東西，
產生的結果為何？
(也就是，不要叫別人幫您打程式碼，不要叫別人幫您寫報告)

- 請找一些您想用程式來幫來解決的問題，
(課業上的，生活上的都可以)

- 請準時繳交作業與報告

課程工作與評分

■ 作業：

- 每週一次，約 10 次，佔 60%

■ 專題報告：

- 期末，繳交一份專題報告，佔 40%

■ 評分標準：

- 最低數量目標：都有參與課程，以及都有交作業
- A, B, C 的差別在於：作業與報告的品質，是否符合課程的精髓

A

- A+：所有目標皆達成且超越期望 (All goals achieved beyond expectation)
- A：所有目標皆達成 (All goals achieved)
- A-：所有目標皆達成，但需一些精進 (All goals achieved, but need some polish)
- B+：達成部分目標，且品質佳 (Some goals well achieved)

B

- B：達成部分目標，但品質普通 (Some goals adequately achieved)
- B-：達成部分目標，但有些缺失 (Some goals achieved with minor flaws)

C

- C+：達成最低目標 (Minimum goals achieved)
- C：達成最低目標，但有些缺失 (Minimum goals achieved with minor flaws)
- C-：達成最低目標但有重大缺失 (Minimum goals achieved with major flaws)
- F：未達成最低目標 (Minimum goals not achieved)
- X：因故不核予成績 (Not graded due to unexcused absences or other reasons)

■ 評分標準：（範例）

■ 報告 (.pdf) 部分評分方式：

- **9分:** 詳細說明如何使用 R 程式，來進行數據擷取，計算，繪圖等呈現數據的功能
 - 例如：使用了哪一個指令，以及設定的參數，獲得了哪一些數據或圖形
 - 也就是，每一個地方的說明，都要有三個部分：
 - (1) 程式碼，(2) 使用此程式碼用途說明，以及(3) 呈現所產生的數據或圖的結果
- **7分:** 大概說明如何使用 R 程式，來進行數據擷取，計算，繪圖等呈現數據的功能
 - 例如：僅呈現最後數據或圖形結果，但是，不清楚如何使用 R 程式來獲得這
- **5分:** 大部分都是文字描述問題與處理過程，並未說明如何使用 R 程式

■ 程式 (.R) 部分評分方式：

- **9分:** 所有程式都可以執行成功，大部分主要程式碼，都有註解
- **7分:** 部分程式都可以執行成功，僅註解部分程式碼
- **5分:** 大部分程式無法執行成功，程式碼沒有什麼註解

每週作業與期末專題報告

■ 作業：60%

- 每週一次，
- 針對每週討論內容，
練習一些相關的計算與繪圖等程式設計功能，
預計需要花費時間，約 30-60分鐘
- 最後，e-mail 繳交：一份程式碼檔 (.R)，以及成果說明檔(.pdf)

■ 專題報告：40%

- 主題自選，可以與自己的課業或興趣有關，或者從資料庫中尋找
- 進行工作，包含：
搜尋問題，取得數據，瞭解理論基礎，確認分析方法，
設計與撰寫程式，成果分析與撰寫說明報告
- 期末，繳交專題成果報告說明檔 (.pdf)，與程式碼檔 (.R)，
以及其他數據檔

每週作業與期末專題報告

計算機程式設計 – 2020F

Unit 01: 課程介紹

Feng-Li Lian at NTU-EE

■ 統計學習方式：

■ 請註明此次的學習方式所花的時間，例如：

作業編號	現場上課	同步觀看	事後觀看	閱讀講義	編纂程式	整理作業	
HW01	40	60	40	25	40	20	(分鐘)

<https://forms.gle/TGYXj2uLoL4HwqLHA>

學習時間登錄

請輸入此次的學習方式所花的時間（單位為分鐘）

*必填

基本資料

學號（例如：B02921033）*

您的回答

系級（例如：電機三）*

您的回答

作業編號*

☐ HW01

☐ HW02

☐ HW03

現場上課

現場上課*

(1: 10 min左右, 2: 20 min左右, 3: 30 min左右, ..., 10: 100 min or above)

0 1 2 3 4 5 6 7 8 9 10

0 min ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ 10: >= 100 min

同步觀看

同步觀看*

(1: 10 min左右, 2: 20 min左右, 3: 30 min左右, ..., 10: 100 min or above)

0 1 2 3 4 5 6 7 8 9 10

0 min ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ 10: >= 100 min

事後觀看

- The R Project for Statistical Computing:
 - <https://www.r-project.org/>
- The R Datasets Package:
 - <https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/00Index.html>
- UC Irvine Machine Learning Repository:
 - <https://archive.ics.uci.edu/ml/datasets.html>
- The Free Datasets at r-dir.com:
 - <https://r-dir.com/reference/datasets.html>
- Rdatasets: An archive of datasets distributed with R:
 - <http://vincentarelbundock.github.io/Rdatasets>
- Datasets in R packages (IAState):
 - http://www.public.iastate.edu/~hofmann/data_in_r_sortable.html

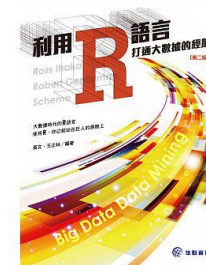
■ 實用R程式設計

- 郭英勝, 鄭志宏, 謝哲光, 龔志銘, 松崗, 2014/06/30



■ 利用R語言打通大數據的經脈(第2版)

- 黃文, 王正林, 佳魁資訊, 2016/01/22



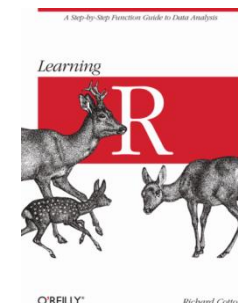
■ R 軟體資料分析基礎與應用

- Jared P. Lander 著、鍾振蔚 譯, 旗標 2015/7/31



■ Learning R

- Richard Cotton, O'Reilly, 2013
- http://kek.ksu.ru/EOS/DataMining/1449357105_LearR.pdf

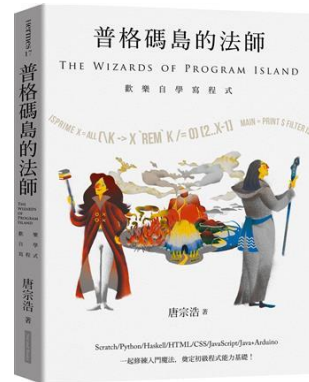


■ An Introduction to R

- W. N. Venables, D. M. Smith and the R Core Team, V 3.3.1 (06-21-16)
- <https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf>

參考書籍

- 普格碼島的法師：歡樂自學寫程式
 - 唐宗浩 著，網路與書出版，2017/07/31
- 改變世界的九大演算法：
讓今日電腦無所不能的最強概念
 - John MacCormick 著，陳正芬 譯，經濟新潮社 2014/08/07



■ The 2020-2017 Top Programming Languages by IEEE Spectrum:

2020

Rank	Language	Score
1	Python ▾	100.0
2	Java ▾	95.3
3	C ▾	94.6
4	C++ ▾	87.0
5	JavaScript	79.5
6	R ▾	78.6
7	Arduino ▾	73.2
8	Go ▾	73.1
9	Swift ▾	70.5
10	Matlab ▾	68.4

2019

Rank	Language	Score
1	Python	100.0
2	Java	96.3
3	C	94.4
4	C++	87.5
5	R	81.5
6	JavaScript	79.4
7	C#	74.5
8	Matlab	70.6
9	Swift	69.1
10	Go	68.0

2018

1.	Python	100.0
2.	C++	99.7
3.	Java	97.5
4.	C	96.7
5.	C#	89.4
6.	PHP	84.9
7.	R	82.9
8.	JavaScript	82.6
9.	Go	76.4
10.	Assembly	74.1

2017

1.	Python	100.0
2.	C	99.7
3.	Java	99.5
4.	C++	97.1
5.	C#	87.7
6.	R	87.7
7.	JavaScript	85.6
8.	PHP	81.2
9.	Go	75.1
10.	Swift	73.7

- ◆ **R** is a free software environment for statistical computing and graphics.
- ◆ **R** is widely used among statisticians and data miners for developing statistical software and data analysis.

為什麼選擇 R 語言？

- The 2014, 2015, 2016 & 2017 Top Programming Languages (IEEE Spectrum, Jul 2015, Jul 2016, Jul 2017) :

2017		2016		2015		2014
Language Rank		Language Rank		Language Rank		Spectrum Ranking
1. Python	100.0	1. C	100.0	1. Java	100.0	100.0
2. C	99.7	2. Java	98.1	2. C	99.9	99.3
3. Java	99.5	3. Python	98.0	3. C++	99.4	95.5
4. C++	97.1	4. C++	95.9	4. Python	96.5	93.5
5. C#	87.7	5. R	87.9	5. C#	91.3	92.4
6. R	87.7	6. C#	86.7	6. R	84.8	84.8
7. JavaScript	85.6	7. PHP	82.8	7. PHP	84.5	84.5
8. PHP	81.2	8. JavaScript	82.2	8. JavaScript	83.0	78.9
9. Go	75.1	9. Ruby	74.5	9. Ruby	76.2	74.3
10. Swift	73.7	10. Go	71.9	10. Matlab	72.4	72.8

- ◆ R is following its momentum from previous years, as part of a positive trend in general for modern big-data languages

Sources: <https://spectrum.ieee.org/computing/software/the-2015-top-ten-programming-languages>

Sources: <https://spectrum.ieee.org/computing/software/the-2016-top-programming-languages>

Sources: <https://spectrum.ieee.org/computing/software/the-2017-top-programming-languages>

範例：鳶尾花 (iris) 資料集

■ 鳶尾花(iris)資料集

- 非常著名的生物資訊資料集之一
- 取自美國加州大學歐文分校的機械學習資料庫

■ 資料的筆數為150筆，

■ 共有五個欄位：

1. 花萼長度(Sepal Length)：計算單位是公分。
2. 花萼寬度(Sepal Width)：計算單位是公分。
3. 花瓣長度(Petal Length)：計算單位是公分。
4. 花瓣寬度(Petal Width)：計算單位是公分。
5. 類別(Class)：可分為 Setosa，Versicolor 和 Virginica 三個品種。

範例：鳶尾花 (iris) 資料集

R Console

```
> iris
```

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4		
2	4.9	3.0	1.4		
3	4.7	3.2	1.3		
4	4.6	3.1	1.5		
5	5.0	3.6	1.4		
6	5.4	3.9	1.7		
7	4.6	3.4	1.4		
8	5.0	3.4	1.5		
9	4.4	2.9	1.4		
10	4.9	3.1	1.5		
11	5.4	3.7	1.5		
12	4.8	3.4	1.6		
13	4.8	3.0	1.4		
14	4.3	3.0	1.1		
15	5.8	4.0	1.2		
16	5.7	4.4	1.5		
17	5.4	3.9	1.3		
18	5.1	3.5	1.4		
19	5.7	3.8	1.7		
20	5.1	3.8	1.5		
21	5.4	3.4	1.7		
22	5.1	3.7	1.5		
23	4.6	3.6	1.0		
24	5.1	3.3	1.7		
25	4.8	3.4	1.9		
26	5.0	3.0	1.6		
27	5.0	3.4	1.6		
28	5.2	3.5	1.5		
29	5.2	3.4	1.4		
30	4.7	3.2	1.6		
31	4.8	3.1	1.6		
32	5.4	3.4	1.5		
33	5.2	4.1	1.5		

R Console

41	5.0	3.5	1.3	0.3	setosa
42	4.5	2.3	1.3	0.3	setosa
43	4.4	3.2	1.3	0.2	setosa
44	5.0	3.5	1.6	0.6	setosa
45	5.1	3.8	1.9	0.4	setosa
46	4.8	3.0	1.4	0.3	setosa
47	5.1	3.8	1.6	0.2	setosa
48	4.6	3.2	1.4	0.2	setosa
49	5.3	3.7	1.5	0.2	setosa
50	5.0	3.3	1.4	0.2	setosa
51	7.0	3.2	4.7	1.4	versicolor
52	6.4	3.2	4.5	1.5	versicolor
53	6.9	3.1	4.9	1.5	versicolor
54	5.5	2.3	4.0	1.3	versicolor
55	6.5	2.8	4.6	1.5	versicolor
56	5.7	2.8	4.5	1.3	versicolor
57	6.3	3.3	4.7	1.6	versicolor
58	4.9	2.4	3.3	1.0	versicolor
59	6.6	2.9	4.6	1.3	versicolor
60	5.2	2.7	3.9	1.4	versicolor
61	5.0	2.0	3.5	1.0	versicolor
62	5.9	3.0	4.2	1.5	versicolor
63	6.0	2.2	4.0	1.0	versicolor
64	6.1	2.9	4.7	1.4	versicolor
65	5.6	2.9	3.6	1.3	versicolor
66	6.7	3.1	4.4	1.4	versicolor
67	5.6	3.0	4.5	1.5	versicolor
68	5.8	2.7	4.1	1.0	versicolor
69	6.2	2.2	4.5	1.5	versicolor
70	5.6	2.5	3.9	1.1	versicolor
71	5.9	3.2	4.8	1.8	versicolor
72	6.1	2.8	4.0	1.3	versicolor
73	6.3	2.5	4.9	1.5	versicolor
74	6.1	2.8	4.7	1.2	versicolor
75	6.4	2.9	4.3	1.3	versicolor

範例：鳶尾花 (iris) 資料集

```
R Console

> summary(iris)

  Sepal.Length    Sepal.Width    Petal.Length    Petal.Width
Min.      :4.300    Min.      :2.000    Min.      :1.000    Min.      :0.100
1st Qu.:5.100    1st Qu.:2.800    1st Qu.:1.600    1st Qu.:0.300
Median :5.800    Median :3.000    Median :4.350    Median :1.300
Mean   :5.843    Mean   :3.057    Mean   :3.758    Mean   :1.199
3rd Qu.:6.400    3rd Qu.:3.300    3rd Qu.:5.100    3rd Qu.:1.800
Max.   :7.900    Max.   :4.400    Max.   :6.900    Max.   :2.500

   Species
setosa   :50
versicolor:50
virginica :50

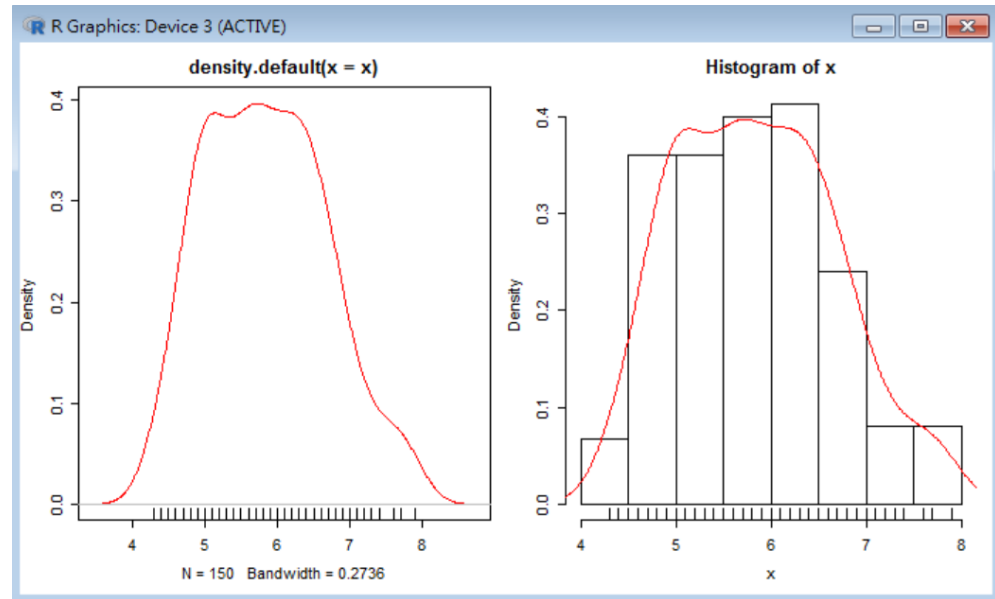
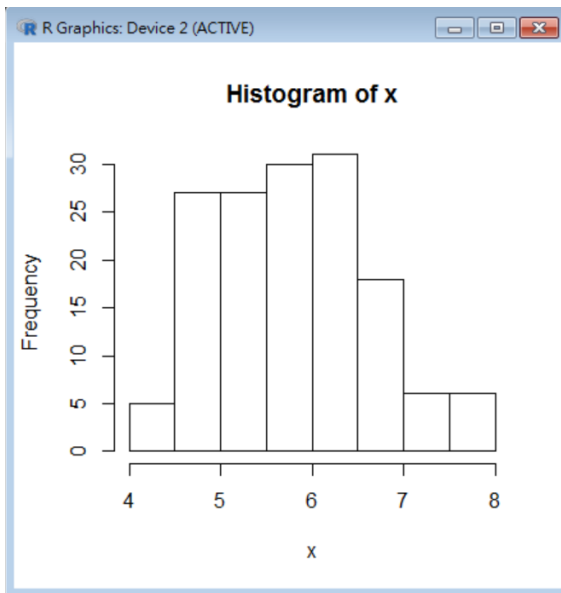
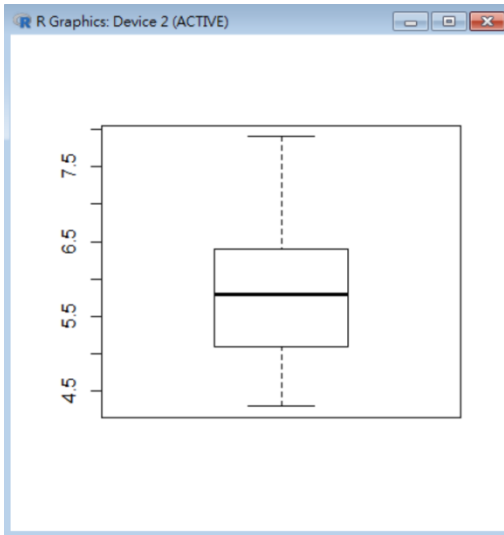
> |
```

範例：鳶尾花 (iris) 資料集

計算機程式設計 – 2020F

Unit 01: 課程介紹

Feng-Li Lian at NTU-EE

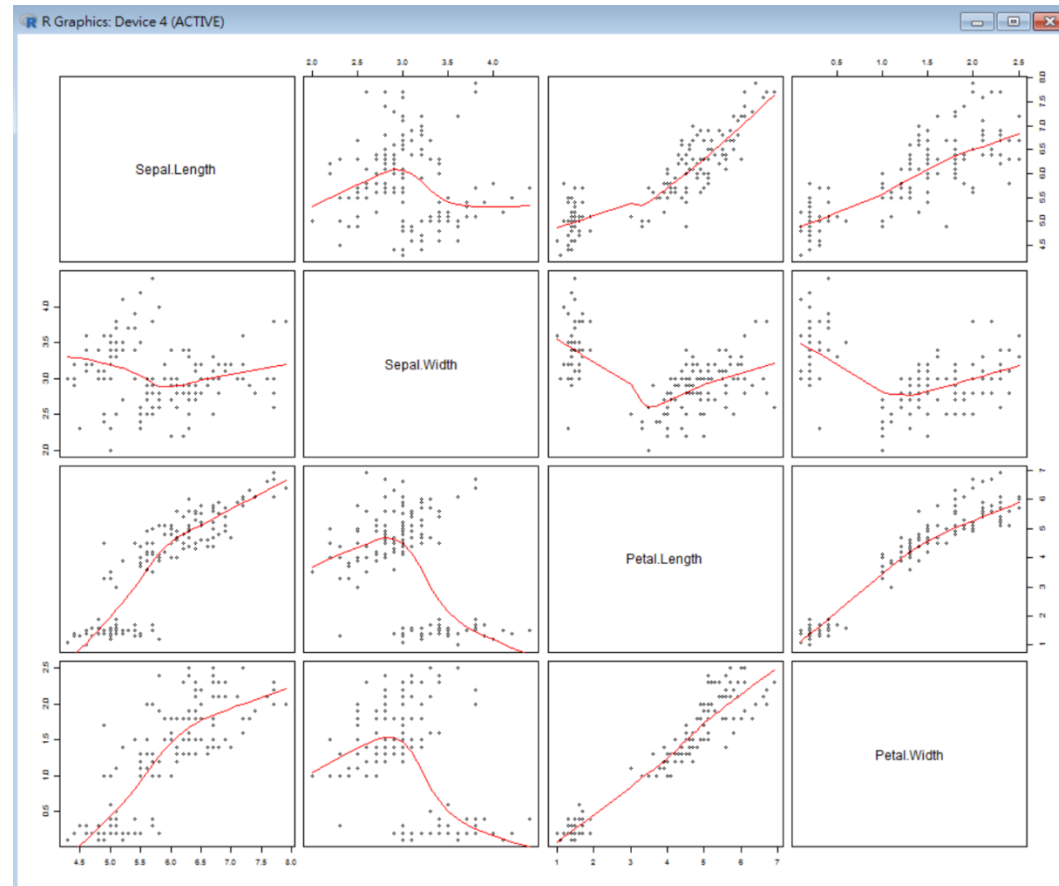
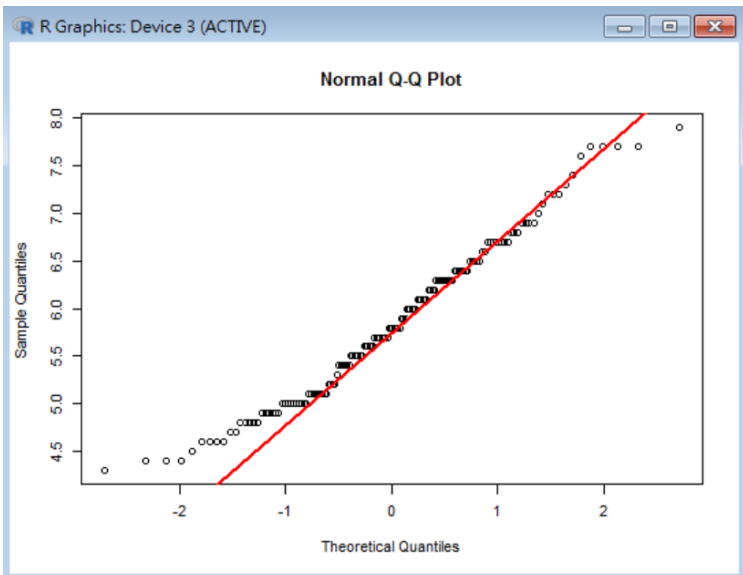
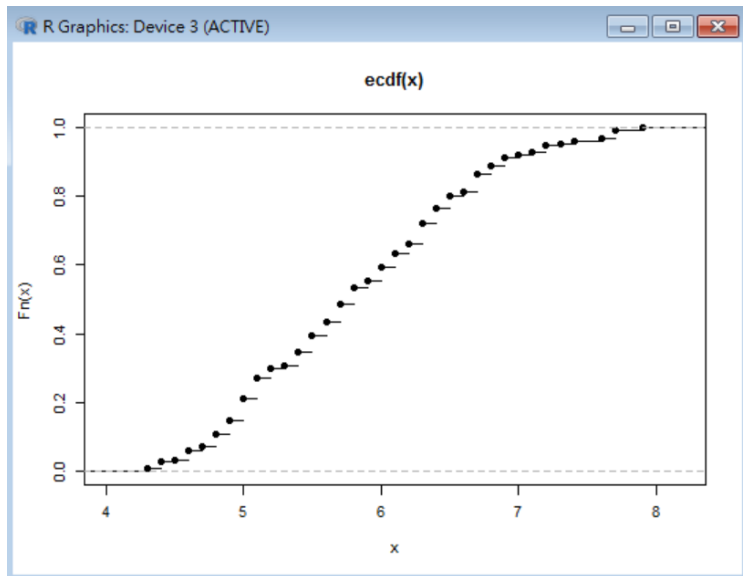


範例：鳶尾花 (iris) 資料集

計算機程式設計 – 2020F

Unit 01: 課程介紹

Feng-Li Lian at NTU-EE



討論事項

- 加選授權碼：
 - 總人數：36人
 - 總人數：36-42人
 - 總人數：>42人
- 教室與座位：
 - 自由入座
- 上課時間與電腦教室：
 - 2-4pm: 講解，實作
 - 1:30-2pm, 4-4:20pm: 自由討論
- 作業與報告：
 - 一人一組
- 自備電腦：
 - 需要自備電腦嗎？
- 課程內容與主題：
 - 使用基本資料庫，練習計算與分析資料庫
 - 提供您的資料庫與預計討論的問題
- 成績計算方式：
 - 標準方式：依照 作業 與 報告分數
 - 創新方式：依照 參與課程互動進行的程度