(3) 資料的輸入與輸出

吳漢銘
淡江大學 數學系
資料科學與數理統計組

http://www.hmwu.idv.tw
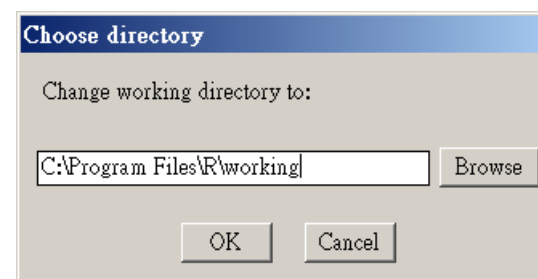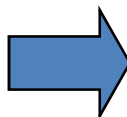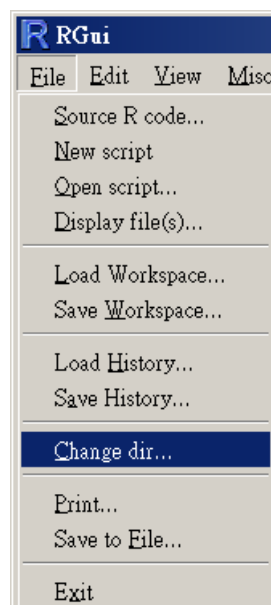
2/29

本章大綱&學習目標

- 標準輸入及輸出 (Standard Input/Output)
- 讀取外部資料檔 (Reading Data From Files)
- 匯入內建資料 (Load Built-in Data)
- 匯出文字檔 (Export to Text Files)
- 讀取部份資料
- R環境的記憶體設置

Setting Working Directory

■ 工作第一步: 設定工作目錄

```
> getwd()
[1] "C:/Documents and Settings/user/My Documents"
> setwd("C:\\Program Files\\R\\working")
> getwd()
[1] "C:/Program Files/R/working"
```

標準輸出: cat

Usage

```
cat(... , file = "", sep = " ", fill = FALSE, labels = NULL, append = FALSE)
```

```
> stdout()
description        class        mode          text         opened      can read     can write
   "stdout"    "terminal"        "w"        "text"      "opened"          "no"          "yes"
> ?stdout()
>
> cat("Hello R users!\n")
Hello R users!
>
> a <- c(1,2,3)
> cat("Here is a list: ", a, "\n")
Here is a list:  1 2 3
>
> cat("3 + 5 =", 3+ 5, "\n" )
3 + 5 = 8
>
> cat("A test list: ", paste("Test", 1:3, sep="-"), "\n")
A test list:  Test-1 Test-2 Test-3
```

標準輸出: cat

```
C:\Program Files\R\working\Example2.R - R Editor
a1 <- 1.2123344
a2 <- 23.3
a3 <- 10/3

cat("iteration", "\t", "mathod-1","\t","method-2", "\t", "method-3\n")
for (i in 1:3){
    cat(i,"\t", round(a1, 3),"\t",  round(a2, 3), "\t", round(a3, 3),"\n")
    a1 <- a1+i
    a2 <- a2*i
    a3 <- a3/i
}
```

```
> a1 <- 1.2123344
> a2 <- 23.3
> a3 <- 10/3
>
> cat("iteration", "\t", "mathod-1","\t","method-2", "\t", "method-3\n")
iteration        mathod-1        method-2        method-3
> for (i in 1:3){
+     cat(i,"\t", round(a1, 3),"\t",  round(a2, 3), "\t", round(a3, 3),"\n")
+     a1 <- a1+i
+     a2 <- a2*i
+     a3 <- a3/i
+ }
1        1.212   23.3    3.333
2        2.212   23.3    3.333
3        4.212   46.6    1.667
>
```

```
> source("Example2.R")
iteration mathod-1   method-2 method-3
1           1.212       23.3     3.333
2           2.212       23.3     3.333
3           4.212       46.6     1.667
```

標準輸出: cat

```
> cat("this is my output","\n", "2 3 5 7","\n", "11 13 17 19", file="test1.txt")
> cat("this is my output", "2 3 5 7", "11 13 17 19", file="test2.txt", sep="\n")
```
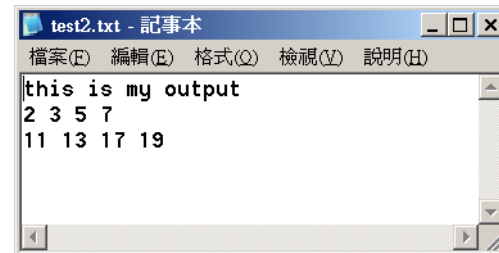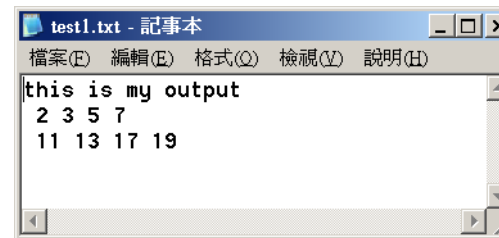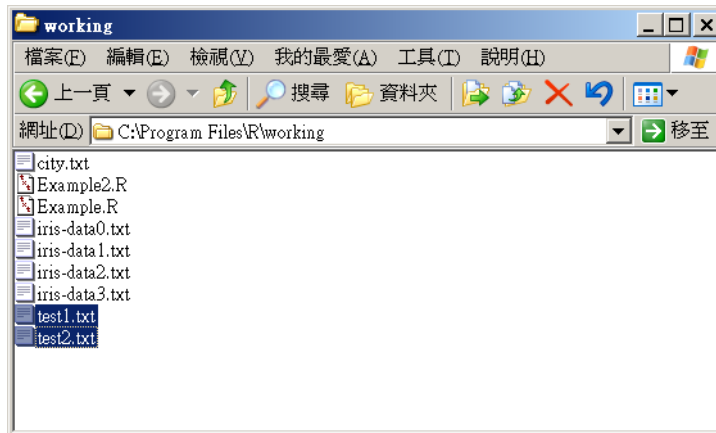

See also:
* **print**
* **sprintf**
* **print.data.frame**
* **paste**

標準輸入 (Standard Input)

```
> stdin()
description          class           mode            text         opened      can read    can write
    "stdin"    "terminal"            "r"          "text"       "opened"         "yes"         "no"
```

```
> a <- scan()
1: 1 2
3: 3
4:
Read 3 items
> a
[1] 1 2 3
> b <- scan(nmax=1)
1: 5
Read 1 item
> b
[1] 5
> b <- scan(nmax=1,
quiet=TRUE)
1: 5
> b
[1] 5
```

logical, integer, numeric, complex, character, raw and list

```
> c <- scan(what="character", quiet=TRUE)
1: this is a test
5:
> c
[1] "this" "is"    "a"     "test"
```

標準輸入 (Standard Input)

```
> d <- scan(what=list(name="character", age="numeric", isboy="logical"))
1: john 28 true
2: mary 11 false
3:
Read 2 records
> d
$name
[1] "john" "mary"

$age
[1] "28" "11"

$isboy
[1] "true"  "false"
```
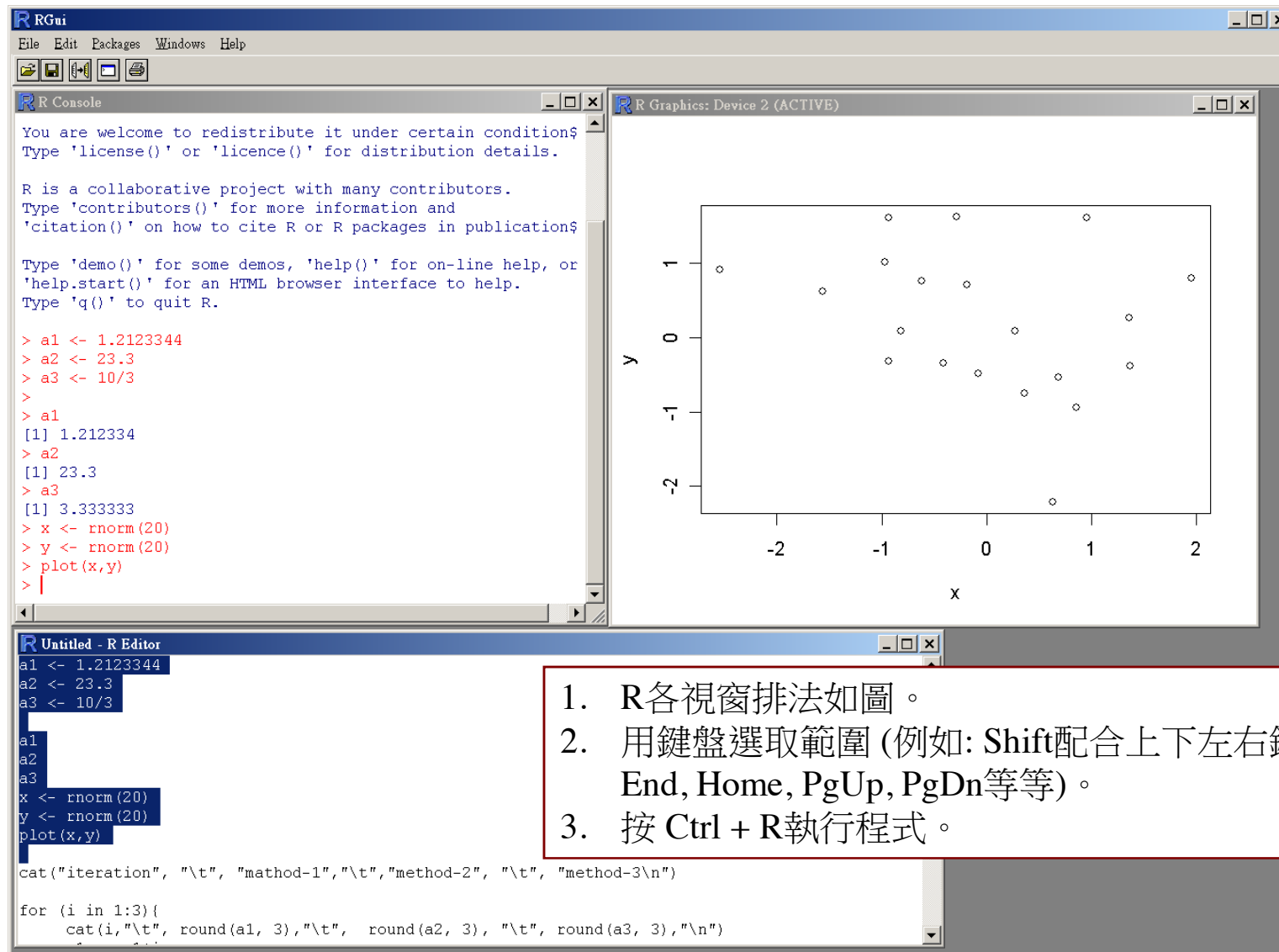
```
> e <- scan()
1: 1 2 3
4: 4 5 6
7: 7 8 9
10:
Read 9 items
> e.mat <- matrix(e, ncol=3, byrow=TRUE)
> e.mat
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
[3,]    7    8    9
```

RGui 小技巧

1. R各視窗排法如圖。
2. 用鍵盤選取範圍 (例如: Shift配合上下左右鍵，End, Home, PgUp, PgDn等等)。
3. 按 Ctrl + R執行程式。

課堂練習1
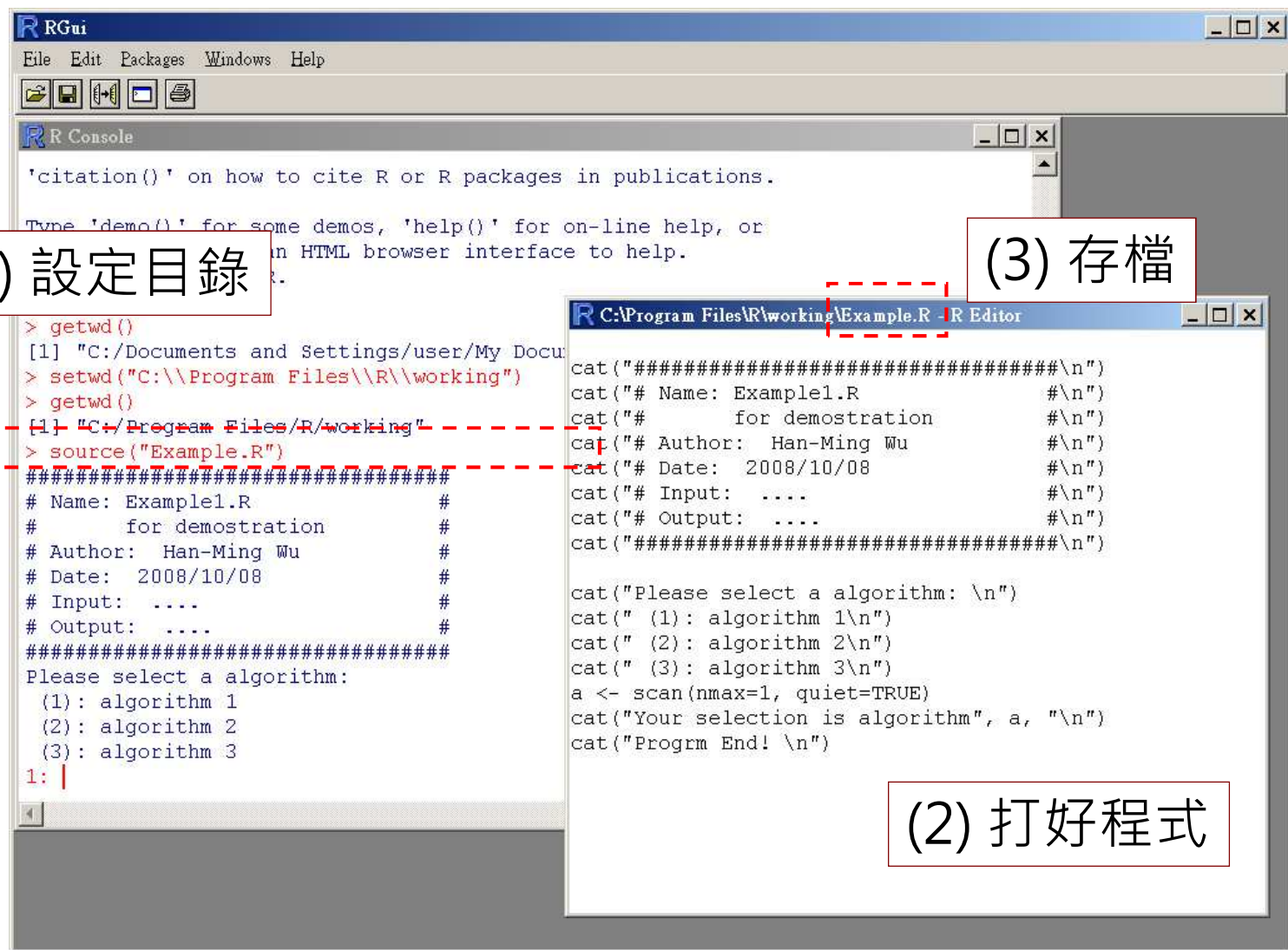
(1) 設定目錄

(3) 存檔

(4) 執行程式

(2) 打好程式

(5) 請用Rstudio建立一個專案，並實作課堂練習1

讀取外部資料檔: `read.table()`

```
Reads a file in table format and creates a data frame from it, with cases
corresponding to lines and variables to fields in the file.

read.table(file, header = FALSE, sep = "", quote = "\"'",
           dec = ".", row.names, col.names,
           as.is = !stringsAsFactors,
           na.strings = "NA", colClasses = NA, nrows = -1,
           skip = 0, check.names = TRUE, fill = !blank.lines.skip,
           strip.white = FALSE, blank.lines.skip = TRUE,
           comment.char = "#",
           allowEscapes = FALSE, flush = FALSE,
           stringsAsFactors = default.stringsAsFactors(),
           fileEncoding = "", encoding = "unknown", text, skipNul = FALSE)
```

`read.table()`

- read in a rectangular grid of data.
- 文字檔.txt, 以空白(" ")或Tab("\t")做區隔。
- **`read.table()`** is an inefficient way to read in very large numerical matrices. (use **`scan()`**)

`read.csv()`

- 格式檔.csv，以","做區隔

讀取外部資料檔(Reading Data From Files)

！注意資料是否有「欄位名稱」。

first line: a name for each variable
`header=TRUE`

iris-data1.txt

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa
10	4.9	3.1	1.5	0.1	setosa
11	5.4	3.7	1.5	0.2	setosa
12			1.6		
13			1.4		
14			1.1		
15			1.2		
16	5.7	4.4	1.5	0.4	setosa
17	5.4	3.9	1.3	0.4	setosa
18	5.1	3.5	1.4	0.3	setosa
19	5.7	3.8	1.7	0.3	setosa
20	5.1	3.8	1.5	0.3	setosa
21	5.4	3.4	1.7	0.2	setosa
22	5.1	3.7	1.5	0.4	setosa
23	4.6	3.6	1	0.2	setosa

row label

values factors

iris-data2.txt

no	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa
10	4.9	3.1	1.5	0.1	setosa
11	5.4	3.7	1.5	0.2	setosa
12	4.8	3.4	1.6	0.2	setosa
13	4.8	3	1.4	0.1	setosa
14	4.3	3	1.1	0.1	setosa
15	5.8	4	1.2	0.2	setosa
16	5.7	4.4	1.5	0.4	setosa
17	5.4	3.9	1.3	0.4	setosa
18	5.1	3.5	1.4	0.3	setosa
19	5.7	3.8	1.7	0.3	setosa
20	5.1	3.8	1.5	0.3	setosa
21	5.4	3.4	1.7	0.2	setosa
22	5.1	3.7	1.5	0.4	setosa
23	4.6	3.6	1	0.2	setosa

iris-data3.txt

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3	1.4	0.1	setosa
4.3	3	1.1	0.1	setosa
5.8	4	1.2	0.2	setosa
5.7	4.4	1.5	0.4	setosa
5.4	3.9	1.3	0.4	setosa
5.1	3.5	1.4	0.3	setosa
5.7	3.8	1.7	0.3	setosa
5.1	3.8	1.5	0.3	setosa
5.4	3.4	1.7	0.2	setosa
5.1	3.7	1.5	0.4	setosa
4.6	3.6	1	0.2	setosa

```
my.data <- read.table("iris-data1.txt")

my.data <- read.table("iris-data2.txt", header=TRUE, row.names=1)

my.data <- read.table("iris-data3.txt", header=TRUE, sep="\t")
```

課堂練習2.1

```
> my.data <- read.table("iris-data0.txt", header=FALSE)
> dim(my.data)
[1] 150    5
> my.data[1:3,]
   V1  V2  V3  V4      V5
1 5.1 3.5 1.4 0.2 setosa
2 4.9 3.0 1.4 0.2 setosa
3 4.7 3.2 1.3 0.2 setosa
> attributes(my.data)
$names
[1] "V1" "V2" "V3" "V4" "V5"

$class
[1] "data.frame"

$row.names
  [1]   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18
...
[145] 145 146 147 148 149 150

> row.names(my.data)
  [1] "1"   "2"   "3"   "4"   "5"   "6"   "7"   "8"   "9"   "10"  "11"  "12"
..
[145] "145" "146" "147" "148" "149" "150"
> names(my.data)
[1] "V1" "V2" "V3" "V4" "V5"
> colnames(my.data)
[1] "V1" "V2" "V3" "V4" "V5"
>
```

iris-data0.txt

5.1	3.5	1.4	0.2	setosa
4.9	3	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3	1.4	0.1	setosa
4.3	3	1.1	0.1	setosa
5.8	4	1.2	0.2	setosa
5.7	4.4	1.5	0.4	setosa
5.4	3.9	1.3	0.4	setosa
5.1	3.5	1.4	0.3	setosa
5.7	3.8	1.7	0.3	setosa
5.1	3.8	1.5	0.3	setosa
5.4	3.4	1.7	0.2	setosa
5.1	3.7	1.5	0.4	setosa
4.6	3.6	1	0.2	setosa

課堂練習2.2

```
> str(my.data)
'data.frame':    150 obs. of  5 variables:
 $ V1: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ V2: num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ V3: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
 $ V4: num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
 $ V5: Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

iris-data0.txt

5.1	3.5	1.4	0.2	setosa
4.9	3	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3	1.4	0.1	setosa
4.3	3	1.1	0.1	setosa
5.8	4	1.2	0.2	setosa
5.7	4.4	1.5	0.4	setosa
5.4	3.9	1.3	0.4	setosa
5.1	3.5	1.4	0.3	setosa
5.7	3.8	1.7	0.3	setosa
5.1	3.8	1.5	0.3	setosa
5.4	3.4	1.7	0.2	setosa
5.1	3.7	1.5	0.4	setosa
4.6	3.6	1	0.2	setosa

注意事項 (Notes)

- **Missing values:**
  - code **"NA"** in the files.
  - **na.strings="any words".**
  - Numeric columns: **NaN, Inf, -Inf**

- **Blank lines:**
  - **read.table()** ignores empty lines.

- **Fixed-width-format file**
  - **read.fwf()**
  - **read.fortran()**

讀取外部資料檔: `scan()`

```
scan(file = "", what = double(0), nmax = -1, n = -1, sep = "",
     quote = if(identical(sep, "\n")) "" else "'\"", dec = ".",
     skip = 0, nlines = 0, na.strings = "NA",
     flush = FALSE, fill = FALSE, strip.white = FALSE,
     quiet = FALSE, blank.lines.skip = TRUE, multi.line = TRUE,
     comment.char = "", allowEscapes = FALSE,
     encoding = "unknown")
```

sep

by default, scan expects to read white-space delimited input fields. Alternatively, sep can be used to specify a character which delimits fields. A field is always delimited by an end-of-line marker unless it is quoted.

skip

the number of lines of the input file to skip before beginning to read data values.

nlines

if positive, the maximum number of lines of data to be read.

讀取外部資料檔: scan()

```
my.data <- scan(file="iris-data0.txt", what=list(w=numeric(0), x=numeric(0),
y=numeric(0), z=numeric(0), name="character"))


my.mat <- as.data.frame(my.data)
```

```
my.data <- scan(file="iris-data1.txt", what=list(n=integer(0), w=numeric(0),
x=numeric(0), y=numeric(0), z=numeric(0), name="character"), skip=1)
my.data$n
```

iris-data0.txt

5.1	3.5	1.4	0.2	setosa
4.9	3	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3	1.4	0.1	setosa
4.3	3	1.1	0.1	setosa
5.8	4	1.2	0.2	setosa
5.7	4.4	1.5	0.4	setosa
5.4	3.9	1.3	0.4	setosa
5.1	3.5	1.4	0.3	setosa
5.7	3.8	1.7	0.3	setosa
5.1	3.8	1.5	0.3	setosa
5.4	3.4	1.7	0.2	setosa
5.1	3.7	1.5	0.4	setosa
4.6	3.6	1	0.2	setosa

iris-data1.txt

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa
10	4.9	3.1	1.5	0.1	setosa
11	5.4	3.7	1.5	0.2	setosa
12	4.8	3.4	1.6	0.2	setosa
13	4.8	3	1.4	0.1	setosa
14	4.3	3	1.1	0.1	setosa
15	5.8	4	1.2	0.2	setosa
16	5.7	4.4	1.5	0.4	setosa
17	5.4	3.9	1.3	0.4	setosa
18	5.1	3.5	1.4	0.3	setosa
19	5.7	3.8	1.7	0.3	setosa
20	5.1	3.8	1.5	0.3	setosa
21	5.4	3.4	1.7	0.2	setosa
22	5.1	3.7	1.5	0.4	setosa
23	4.6	3.6	1	0.2	setosa

課堂練習3

```
> getwd()
[1] "C:/Documents and Settings/user/My Documents"
> cat("1 2 3", "11 12 13", "21 22 23", "31 32 33", "41 42 43",
file="ex.txt", sep="\n")
> scan(file="ex.txt", what=list(x=0, y="", z=0))
Read 5 records
$x
[1]  1 11 21 31 41


$y
[1] "2"  "12" "22" "32" "42"


$z
[1]  3 13 23 33 43
```

```
1 2 3
11 12 13
21 22 23
31 32 33
41 42 43
```

Read in a large matrix

```
A <- matrix(scan("matrix.txt", n=200*2000), 200, 2000, byrow=TRUE)
```

readLines()

```
readLines(con = stdin(), n = -1, ok = TRUE, warn = TRUE,
          encoding = "unknown")
```
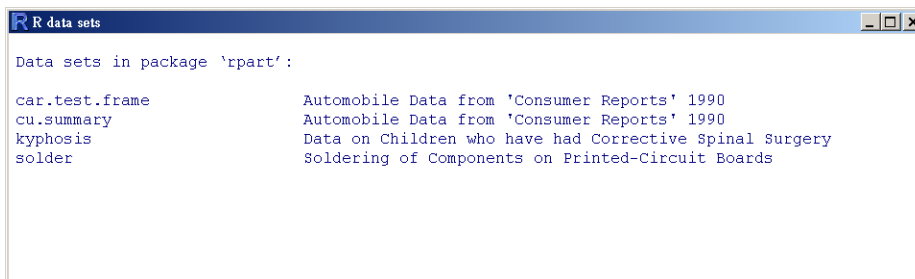
```
> data()

> data(Puromycin, package="datasets")
> Puromycin

> data(package="rpart")
```

讀取R的rda檔案：

```
> load("test.rda")
```

```
R data sets                                                    _ □ ✕

Data sets in package 'datasets':

AirPassengers              Monthly Airline Passenger Numbers 194$
BJsales                    Sales Data with Leading Indicator
BJsales.lead (BJsales)     Sales Data with Leading Indicator
BOD                        Biochemical Oxygen Demand
CO2                        Carbon Dioxide uptake in grass plants
ChickWeight                Weight versus age of chicks on differ$
DNase                      Elisa assay of DNase
EuStockMarkets             Daily Closing Prices of Major Europea$
Formaldehyde               Determination of Formaldehyde
HairEyeColor               Hair and Eye Color of Statistics Stud$
Harman23.cor               Harman Example 2.3
Harman74.cor               Harman Example 7.4
Indometh                   Pharmacokinetics of Indomethicin
InsectSprays               Effectiveness of Insect Sprays
JohnsonJohnson             Quarterly Earnings per Johnson & John$
LakeHuron                  Level of Lake Huron 1875-1972
LifeCycleSavings           Intercountry Life-Cycle Savings Data
Loblolly                   Growth of Loblolly pine trees
Nile                       Flow of the River Nile
Orange                     Growth of Orange Trees
OrchardSprays              Potency of Orchard Sprays
PlantGrowth                Results from an Experiment on Plant G$
Puromycin                  Reaction velocity of an enzymatic rea$
Seatbelts                  Road Casualties in Great Britain 1969$
Theoph                     Pharmacokinetics of theophylline
Titanic                    Survival of passengers on the Titanic
ToothGrowth                The Effect of Vitamin C on Tooth Grow$
UCBAdmissions              Student Admissions at UC Berkeley
UKDriverDeaths             Road Casualties in Great Britain 1969$
UKgas                      UK Quarterly Gas Consumption
```

```
R data sets                                                    _ □ ✕

Data sets in package 'rpart':

car.test.frame          Automobile Data from 'Consumer Reports' 1990
cu.summary              Automobile Data from 'Consumer Reports' 1990
kyphosis                Data on Children who have had Corrective Spinal Surgery
solder                  Soldering of Components on Printed-Circuit Boards
```

```
> library(MASS)

> class(crabs)

> dim(crabs)

> colnames(crabs)

> str(crabs)
```

編輯資料 (Editing Data)

	sp	sex	index	FL	RW	CL	CW	BD
1	B	M	1	8.1	6.7	16.1	19	7
2	B	M	2	8.8	7.7	18.1	20.8	7.4
3	B	M	3	9.2	7.8	19	22.4	7.7
4	B	M	4	9.6	7.9	20.1	23.1	8.2
5	B	M	5	9.8	8	20.3	23	8.2
6	B	M	6	10.8	9	23	26.5	9.8
7	B	M	7	11.1	9.9	23.8	27.1	9.8
8	B	M	8	11.6	9.1	24.5	28.4	10.4
9	B	M	9	11.8	9.6	24.2	27.8	9.7
10	B	M	10	11.8	10.5	25.2	29.3	10.3
11	B	M	11	12.2	10.8	27.3	31.6	10.9
12	B	M	12	12.3	11	26.8	31.5	11.4
13	B	M	13	12.6	10	27.7	31.7	11.4
14	B	M	14	12.8	10.2	27.2	31.8	10.9
15	B	M	15	12.8	10.9	27.4	31.5	11
16	B	M	16	12.9	11	26.8	30.9	11.4
17	B	M	17	13.1	10.6	28.2	32.3	11
18	B	M	18	13.1	10.9	28.3	32.4	11.2
19	B	M	19	13.3	11.1	27.8	32.3	11.3
20	B	M	20	13.9	11.1	29.2	33.3	12.1
21	B	M	21	14.3	11.6	31.3	35.5	12.7
22	B	M	22	14.6	11.3	31.9	36.4	13.7
23	B	M	23	15	10.9	31.4	36.4	13.2

Data Editor

```
library(MASS)
class(crabs)
dim(crabs)
colnames(crabs)
str(crabs)

edit(data.name)
edit(crabs)

new.data <- edit(data.name)
crabs.new <- edit(crabs)
fix(crabs.new)

new.data <- edit(data.frame())
new.data <- edit(matrix(0, ncol=2, nrow=3))
```

```
write.table(x, file = "", append = FALSE, quote = TRUE, sep = " ",
            eol = "\n", na = "NA", dec = ".", row.names = TRUE,
            col.names = TRUE, qmethod = c("escape", "double"))
```

header line

```
> write.csv(iris, "myNewData.csv", sep=",", col.names=TRUE)


> write.table(iris, "myNewData.txt", quote=FALSE, sep="\t")
```

```
> library(MASS)
> hills
> hills10 <- hills[1:10, 1:2]
> hills10

> write.table(hills10, "hill10.txt", sep="\t", quote=F, row.names=TRUE)

> write.table(hills[11:15,1:2], "hill10.txt", append=TRUE, sep="\t",
row.names=TRUE, col.names=FALSE)
```

Note: 在既有的資料檔案中，加入資料時，需要有相同的欄位名稱。

讀取其它軟體檔案

- **This is often best avoided!**

```
> read.xport()  #SAS XPORT
> read.ssd() #SAS dataset
> read.S() #S-plus binary object
> read.spss() #SPSS
> read.xls() #R package(xlsReadWrite)
```

- **Browsing to find files**

```
> Data <- read.table(file.choose(), header=TRUE)
```

- **Checking files from the command line**

```
> File.exists("c:\\temp\\data.txt")
```

```
print(x, ...)

## S3 method for class 'factor':
print(x, quote = FALSE, max.levels = NULL,
      width = getOption("width"), ...)

## S3 method for class 'table':
print(x, digits = getOption("digits"), quote = FALSE,
      na.print = "", zero.print = "0", justify = "none", ...)
```

課堂練習4

```r
>  zz <- file("output.txt", "w")
>  cat("Title line", "2 3 5 7", " ", "11 13 17", file=zz, sep="\n")
>  cat("One more line \n", file=zz)
>  close(zz)


>  zz <- textConnection("output.obj", "w")
>  sink(zz)
>  example(lm)
>  sink()
>  close(zz)
>  cat(output.obj, sep="\n")
>  write(output.obj, file="result.txt")
```

課堂練習5

```
> iris[1:10, ]
> write.table(iris, "iris-data0.txt", sep="\t", quote=F, row.names=FALSE,
col.names = FALSE)
> write.table(iris, "iris-data1.txt", sep="\t", quote=F, row.names=TRUE,
col.names = TRUE)


> write.table(hills[11:15,1:2], "iris-data2.txt", append=TRUE, sep="\t",
row.names=TRUE, col.names=FALSE)


> write.table(hills[11:15,1:2], "iris-data3.txt", append=TRUE, sep="\t",
row.names=TRUE, col.names=FALSE)
```

iris-data0.txt

5.1	3.5	1.4	0.2	setosa
4.9	3	1.4	0.2	setosa
4.7	3.2			
4.6	3.1			
5	3.6			
5.4	3.9	1.7	0.4	setosa
4.6				
5				
4.4				
4.9				
5.4				
4.8				
4.8				
4.3				
5.8				
5.7				
5.4				
5.1				
5.7				
5.1				
5.4				
5.1				
4.6				
5.1				

iris-data1.txt

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa
10	4.9	3.1	1.5	0.1	setosa
11	5.4	3.7	1.5	0.2	setosa
12	4.8	3.4	1.6	0.2	setosa
13	4.8	3	1.4	0.1	setosa
14	4.3	3	1.1	0.1	setosa
15	5.8	4	1.2	0.2	setosa
16	5.7	4.4	1.5	0.4	setosa
17	5.4	3.9	1.3	0.4	setosa
18	5.1	3.5	1.4	0.3	setosa
19	5.7	3.8	1.7	0.3	setosa
20	5.1	3.8	1.7	0.3	setosa
21	5.4	3.4	1.7	0.2	setosa
22	5.1	3.7	1.5	0.4	setosa
23	4.6	3.6	1	0.2	setosa

iris-data2.txt

no	Sepal.Length	Sepal.Width	Petal.Length	P
1	5.1	3.5	1.4	
2	4.9	3	1.4	
3	4.7	3.2		
4	4.6	3.1		
5	5	3.6		
6	5.4	3.9		
7	4.6	3.4		
8	5	3.4		
9	4.4	2.9		
10	4.9	3.1		
11	5.4	3.7		
12	4.8	3.4		
13	4.8	3		
14	4.3	3		
15	5.8	4		
16	5.7	4.4		
17	5.4	3.9		
18	5.1	3.5		
19	5.7	3.8		
20	5.1	3.8		
21	5.4	3.4		
22	5.1	3.7		
23	4.6	3.6		

iris-data3.txt

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3	1.4	0.1	setosa
4.3	3	1.1	0.1	setosa
5.8	4	1.2	0.2	setosa
5.7	4.4	1.5	0.4	setosa
5.4	3.9	1.3	0.4	setosa
5.1	3.5	1.4	0.3	setosa
5.7	3.8	1.7	0.3	setosa
5.1	3.8	1.5	0.3	setosa
5.4	3.4	1.7	0.2	setosa
5.1	3.7	1.5	0.4	setosa
4.6	3.6	1	0.2	setosa

課堂練習5

```
> my.data0 <- read.table("iris-data0.txt")
> my.data0[1:5, ]

> my.data1 <- read.table("iris-data1.txt")
> my.data1[1:5, ]

> my.data2 <- read.table("iris-data2.txt", header=TRUE, row.names=1)
> my.data2[1:5, ]

> my.data3 <- read.table("iris-data3.txt", header=TRUE, sep="\t")
> my.data3[1:5, ]
```

```
> my.sdata0 <- scan(file="iris-data0.txt", what=list(w=numeric(0),
x=numeric(0), y=numeric(0), z=numeric(0), name="character"))
> my.sdata0
> my.mat <- as.data.frame(my.data)
> my.mat[1:5, ]
```

```
> my.sdata1 <- scan(file="iris-data1.txt", what=list(n=integer(0),
w=numeric(0), x=numeric(0), y=numeric(0), z=numeric(0), name="character"),
skip=1)
> str(my.sdata1)
> my.sdata1$n
```

讀取部份資料

■ 僅輸入所需要的部份資料，而不是全部。

```
Variables <- c("NULL", "NULL", "factor", "numeric")
myData <- read.table("fileName", colClasses = Variables)
```

■ 用適合的函式或演算法: O(N) vs O(N²)

```
x <- 1:10000; s <- sample(x, 10)
a1 <- which(x %in% s)
a2 <- intersect(x, s)
a3 <- which(is.element(x, s))

for(i in 1:10000){
    for(j in 1:10){
        if(all.equal(x[i], s[j]){
            ...
        }
    }
}
```

```
> n <- 10000
> p <- 1000
> Mat <- matrix(rnorm(n*p), nrow=n, ncol=p)
> system.time(apply(Mat, 1, sum))
   user   system elapsed
   0.61     0.19    2.56
> system.time(rowSums(Mat))
   user   system elapsed
   0.05     0.00    0.08
```

See also: CRAN Task View: High-Performance and Parallel Computing with R

二進位儲存資料

- 資料儲存以二進位檔(binary)為優先:
  - 讀寫文字檔比壓縮二進位檔慢。
  - 壓縮二進位檔又比二進位慢。

```
> n <- 1000
> p <- 1000
> Mat <- matrix(rnorm(n*p),
nrow=n, ncol=p)
```

```
> system.time(write.table(Mat, file="myData.txt"))
   user   system elapsed
   8.89    0.09    12.14
> system.time(read.table("myData.txt"))
   user   system elapsed
  10.85    0.06    11.98
```

```
> system.time(save(Mat, file="myData.gz"))
   user   system elapsed
   1.11    0.01    2.52
> system.time(load("myData.gz"))
   user   system elapsed
   0.36    0.02    3.56
```

```
> system.time(save(Mat, file="myData.Rdata", compress=FALSE))
   user   system elapsed
   0.24    0.00    0.23
> system.time(load("myData.Rdata"))
   user   system elapsed
   0.23    0.00    0.24
```

R環境的記憶體設置

■ 當R啟動時，設定最大可穫得的記憶體:

"C:\Program Files\R\R-3.1.0\bin\Rgui.exe" --max-mem-size=2040M

□ 最小需求是32MB.

□ 32-bit Windows: 無法超過3G ，通常是2GB。

□ R啟動後僅可設定更高值，不能再用**memory.limit**設定較低的值。

■ `memory.size(max = FALSE)`
目前使用的記憶體量

■ `memory.size(max = TRUE)`
從作業系統可得到的最大量記憶體

■ `memory.limit(size = NA)`
列出目前記憶體的限制

■ `memory.limit(size = 1024)`
設定新的記憶體限制為 1024 MB

■ R與Windows作業系統
最大可穫得的記憶體

- 32-bit R + 32-bit Windows: 2GB.

- 32-bit R + 64-bit Windows: 4GB.

- 64-bit R + 64-bit Windows: 8TB.

R物件所佔用的記憶體

■ 儲存R物件所佔用的記憶體估計。

```
object.size(x)
print(object.size(x), units = "Mb")
```

```
> n <- 10000
> p <- 200
> myData <- as.data.frame(matrix(rnorm(n*p), ncol = p, nrow=n))
> print(object.size(myData), units = "Mb")
15.3 Mb

> write.table(myData, "myData.txt") ## 約 34.7 MB

> InData <- read.table("myData.txt")
> print(object.size(InData), units = "Mb")
15.6 Mb
```