Quick R Introduction

Li Cheng En(leslie@dsp.im)

Ch01：為何要學 R 語言？

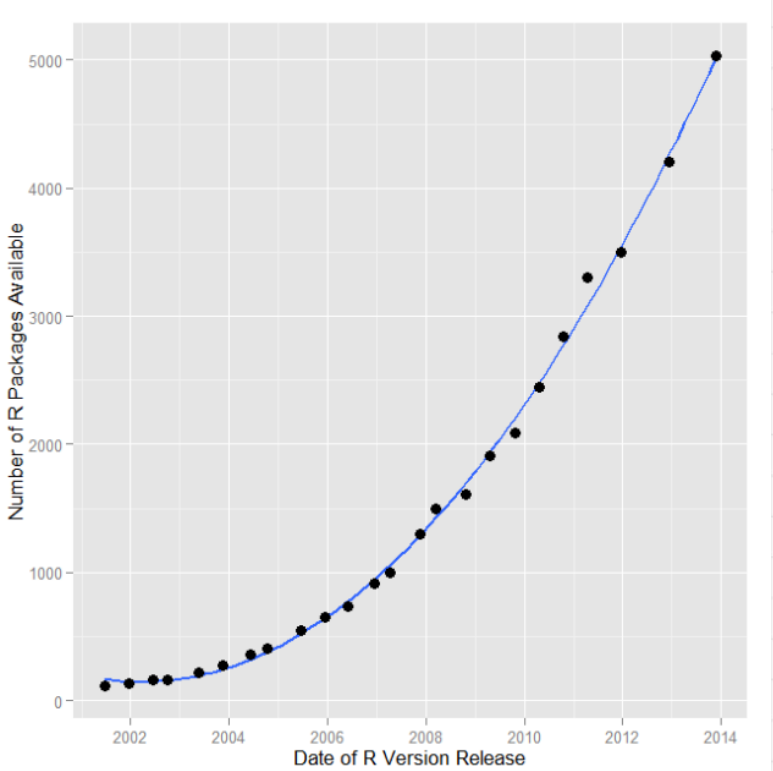
R 是專門為資料分析而設計的程式語言

R 可以執行大多數的統計計算、機器學習、資料採礦方法

取自 http://goo.gl/CkXlvY

dsp.im

免費、開源、豐沛的社群資源

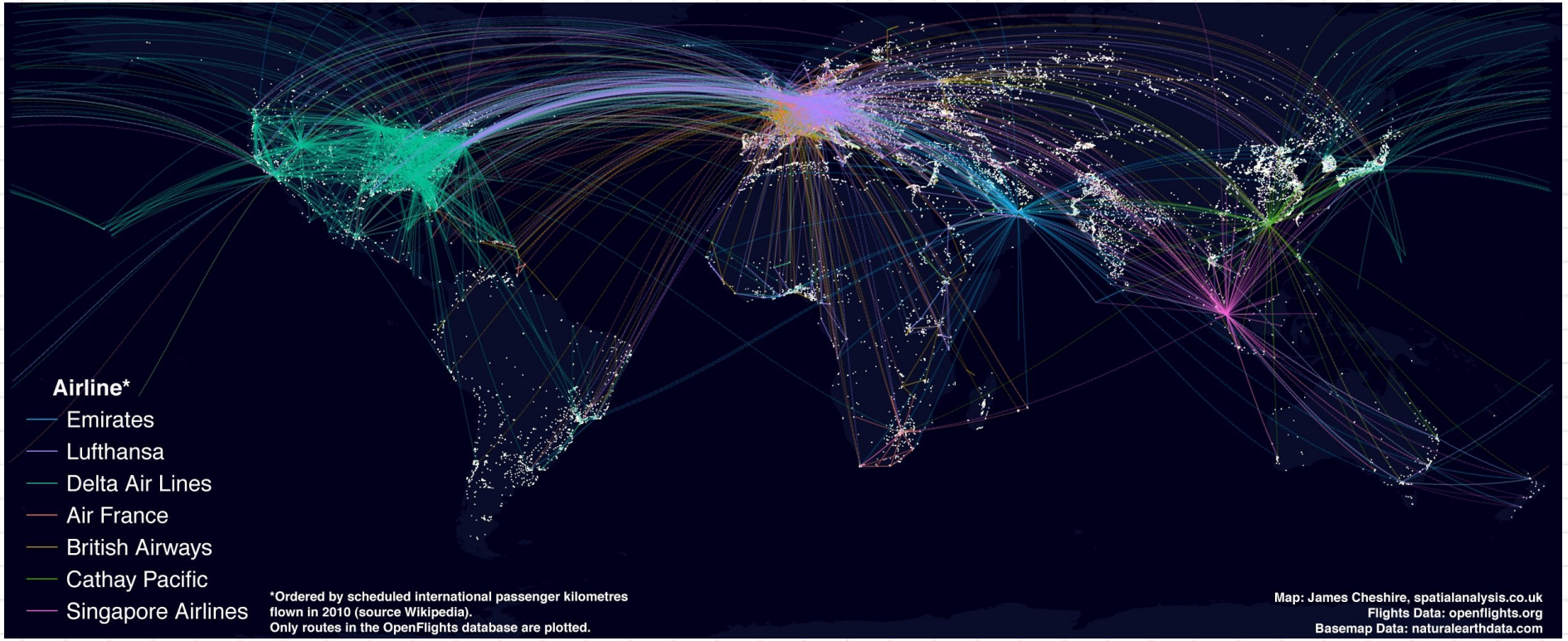


很容易擴充和客製化

取自 http://goo.gl/AwfHLx

dsp.im

R 可以輸出高品質的視覺化



取自 http://goo.gl/q1NX26

課程目標

- 建立 R 的使用環境
- 熟悉 R 語言基礎操作
- 了解 R 語言的物件的結構
- 體會 R 語言的流程控制
- 學習 R 語言的資料整理
- 確立 R 語言的資料爬析概念

dsp.im

Ch02：建立 R 的使用環境

傢俬準備好

環境安裝

- 主程式：R（R-3.2.2 以上版本）
- 編輯界面：RStudio IDE（0.99.473 以上版本）
- 疑難排解指南

RStudio 界面說明

- 程式碼編輯區
- 命令列區
- 環境資訊區
- 檔案系統區

熟悉 RStudio 的 命令列 界面

程式的**輸入**、**輸出**、**中斷**

- 左下角的當符號 > 表示可以輸入指令
- 輸入 1 + 1 後按下 Enter，檢查螢幕輸出
- 輸入 1 + 後按下 Enter，檢查螢幕輸出
  最左下角的開頭變成 + 表示尚未輸入完成，應繼續輸入
- 按下 ESC，會中斷執行中的程式 (左下角回復成 > 開頭)

dsp.im

熟悉 RStudio 的 程式碼編輯 界面

停留時間最多的區域

- New File -> R Script -> Untiled1.R
- 在程式碼編輯區中輸入 `1 + 1` 後按下 Control + Enter，檢查 命令列區
- 在程式碼編輯區中輸入 `1 +` 後按下 Control + Enter，檢查 命令列區
- 在命令列區按下 ESC 中斷程式

Ch03 今天學完，你一定會用R做資料整理

先別想，複製貼上就是了

```
dat1 <- data.frame(date=c("11/29","11/30","12/01","12/02","12/03","12/04","12/05"),
                   weekday=c("Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat"))
dat1 # print R object
```

```
   date weekday
1 11/29     Sun
2 11/30     Mon
3 12/01     Tue
4 12/02     Wed
5 12/03     Thu
6 12/04     Fri
7 12/05     Sat
```

```
dat1[1, 1]  # 取值
dat1[, 1]
dat1[1:2,]
```

先別想，複製貼上就是了

```
dat2 <- data.frame(date=c("11/29","11/30","12/01","12/02","12/03","12/04","12/05"),
                   temp=c(17, 18, 24, 20, 21, 22, 24))
dat2 # print R object
```

```
   date temp
1 11/29   17
2 11/30   18
3 12/01   24
4 12/02   20
5 12/03   21
6 12/04   22
7 12/05   24
```

```
dat2[, 2]
dat2[, 2] * 2
dat2[, 2] + c(1, 0, -0.5, 2, 3, -2, 0.5)
```

先別想，複製貼上就是了

```
dat2[dat2$temp<20,]
```

```
   date temp
1 11/29   17
2 11/30   18
```

```
dat2[grepl("11", dat2$date), ]
```

```
   date temp
1 11/29   17
2 11/30   18
```

```
cbind(dat1,temp=dat2$temp)[dat2$temp<20,]
```

```
   date weekday temp
1 11/29     Sun   17
2 11/30     Mon   18
```

Ch04：基礎教學 - 敘述句與數列

敘述句

```
1
```

```
[1] 1
```

```
2
```

```
[1] 2
```

```
1; 2
```

```
[1] 1
[1] 2
```

筆記

16/59 灰底的區塊為程式碼 (輸入)，[1] 為運算結果 (輸出)

dsp.im

敘述句2

```
# 基礎運算
1 + 2 + 3
```

```
[1] 6
```

```
1 +       2 + 3
```

```
[1] 6
```

```
x <- 10
y <- 4
(x + y) / 2 # 簡單的公式運算
```

```
[1] 7
```

最基礎的物件：數值型向量 (數列)

```
# basic expression of integer vector
c(1, 2, 3, 4)
```

```
[1] 1 2 3 4
```

```
# simple expression
1:4
```

```
[1] 1 2 3 4
```

```
4:1
```

```
[1] 4 3 2 1
```

筆記

- 以 `c(...)` 表示 (c 取自combine之意), 整個函數內容以 `()` 包括起來，元素以逗號

如何生成有序的數值向量

```
seq(1, 4)
```

```
[1] 1 2 3 4
```

```
seq(1, 9, by = 2) # 間隔為2
```

```
[1] 1 3 5 7 9
```

```
seq(1, 9, length.out = 5) # 分割長度為5
```

```
[1] 1 3 5 7 9
```

筆記

- 除了冒號簡記法外，可以透過`seq`函數生成有規則的數值向量(序列)
- 在`seq()`函數中按 tab 鍵觀察有哪些參數可以使用
- `by` 表示數列間隔，預設為1

小挑戰

- 利用簡記法列出 1 ~ 10的數列
- 利用 `seq` 函數列出偶數數列: 2, 4, 6, 8, 10
- 觀察 `seq(1, 10, length.out=5)` 的輸出結果

參考解答

```
1:10
```

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

```
seq(2, 10, by = 2)
```

```
[1]  2  4  6  8 10
```

```
seq(2, 10, length.out = 5)
```

```
[1]  2  4  6  8 10
```

```
seq(1, 10, length.out=5)
```

```
[1]  1.00  3.25  5.50  7.75 10.00
```

Ch05：基礎教學 - 數列的運算

數列的運算

R的數列運算滿足 recycling properties

```
c(1, 2, 3) * c(2, 2, 2)
```

```
[1] 2 4 6
```

```
# shorter arguments are recycled
1:3 * 2
```

```
[1] 2 4 6
```

```
c(0.5, 1.5, 2.5, 3.5) * c(2, 1)
```

```
[1] 1.0 1.5 5.0 3.5
```

小挑戰

向量的四則運算，請計算以下五位女藝人的BMI

向量的取值

- 在[ ] (中括號) 中輸入元素的位置進行取值
- 使負號 (-) 移除給定位置元素 (反向選取)

```
x <- c(174, 158, 160, 168, 173)
x[1]            # 選取第1個位置的元素
```

```
[1] 174
```

```
x[c(1, 3)]      # 選取第1, 3個位置的元素
```

```
[1] 174 160
```

```
x[c(2, 3, 1)]  # 依序取值
```

```
[1] 158 160 174
```

```
# 在[ ]中使用負號 (-) 做反向選取
x[-1]
```

向量的取值2

給定條件進行取值

- 比較運算子(>, <, >=, <=, ==, !=)
- 邏輯運算子 (&, |)

```
x > 160
```

```
[1]  TRUE FALSE FALSE  TRUE  TRUE
```

```
# 使用比較運算子 加上 `which` 函數進行取值
index <- which(x > 160)
index
```

```
[1] 1 4 5
```

```
x[index]
```

```
[1] 174 168 173
```

向量的取代

- 利用 [ ] (中括號) 與 <- (箭號) 進行取代與新增元素

```
x <- c(174, 158, 160, 168, 173)

# 取代特定位置的元素
x[2] <- 158.5 # 取代x物件的第二個元素
x
```

```
[1] 174.0 158.5 160.0 168.0 173.0
```

```
x[c(1, 5)] <- 175
x
```

```
[1] 175.0 158.5 160.0 168.0 175.0
```

```
# 也可以用條件篩選做取代
x[x > 160] <- 170 # 取代大於160的值為170
x
```

向量新增

- 可用 `[ ]` (中括號) 與 `<-` (箭號) 進行新增元素
- `NA` 為系統保留字，表示Not Available / Missing Values

```
x <- c(174, 158, 160, 168, 173)
# 在 [ ] 中新增元素
x[6] <- 168
x
```

```
[1] 174 158 160 168 173 168
```

```
x[8] <- 147
x # 未指定的元素值預設為NA
```

```
[1] 174 158 160 168 173 168  NA 147
```

```
length(x) # 查看向量物件的長度
```

```
[1] 8
```

Ch06：查詢說明檔

在 R 中查詢說明文件

各種自救措施

```
help.start()
ab # 輸入`ab`後 按下tab
?abs   # 等同於 help(abs)
??abs
vignette()
vignette("Introduction", "Matrix")
```

筆記

dsp.im

Ch07：資料測量的尺度

R 的資料形態分類

資料衡量尺度	R變數形態	特性	範例
連續資料	numeric	數值	身高
比例資料	numeric	比值	流失率
區間資料	numeric	大小距離	溫度
順序資料	factor	優先順序	名次
名目資料	factor	類別	國家
布林資料	logical	邏輯值	性別
文字資料	character	文字	電話號碼

判斷 Logical

產生自比較，或是使用T、TRUE、F或FALSE輸入

```
x <- 1 # 賦值
x < 2
```

```
[1] TRUE
```

```
x <= 1
```

```
[1] TRUE
```

字串 (Character)

- 輸入的時候利用"或'來包覆要輸入的文字
- 常用的Character處理函數

字串的剪接：paste

```
x <- "bubble"
y <- "bobble"
paste(x, y, sep=",")
```

```
[1] "bubble,bobble"
```

字串的切割：strsplit

```
strsplit(x, "b")
```

```
[[1]]
[1] ""   "u"  ""   "le"
```

截取子字串：substring

小挑戰

- 取出金城武的姓
- 取出字串 `a <- "2015-12-14"` 的月份

dsp.im

解答

```
# 取出金城武的
name2<-"金城武"
substring(name2, 1, 2)
```

```
[1] "金城"
```

```
# 取出字串 "2015-12-14" 的月份
a <- "2015-12-14"
substring(a, 6, 7)
```

```
[1] "12"
```

```
tmp <- strsplit(a, "-")
tmp[[1]][2]
```

```
[1] "12"
```

Factor (類別)

如何處理名目變數?

```
x <- c("F","M","F","F")
x
```

```
[1] "F" "M" "F" "F"
```

```
x <- factor(c("F","M","F","F"), levels=c("F","M"))
x
```

```
[1] F M F F
Levels: F M
```

```
x <- factor(c("F","M","F","F"), levels=c("F"))
levels(x)
```

```
[1] "F"
```

```
as.integer(x)
```

如何處理順序資料?

```
#農業社會 男尊女卑
Argri <- factor(c("F","M","F","F"), order=TRUE, levels=c("F","M"))

#阿美族 女尊男卑
Amis <- factor(c("F","M","F","F"), order=FALSE, levels=c("F","M"))

#應該要用
Amis <- factor(c("F","M","F","F"), order=TRUE, levels=c("M","F"))
```

換個例子

```
#舉一個認真的例子 - 班上一號到六號分別拿到A, B, C的級別
rank=factor(c("C","A","B","B","C","C"), order=TRUE, level=c("C","B","A"))
rank
```

```
[1] C A B B C C
Levels: C < B < A
```

```
rank[1] < rank[2]
```

```
[1] TRUE
```

Ch08: 資料型態的轉換

向量有同質性 Vector

Character > Numeric > Integer > Logical

```
x <- c(1, 2.0, "3")
x
```

```
[1] "1" "2" "3"
```

資料型態的轉換

- 利用以下函數自行轉換向量型態： `as.character`, `as.numeric`, `as.logical`。

```
as.numeric("2")
```

```
[1] 2
```

```
x <- c(1, 2.0, "3")
as.numeric(x)
```

```
[1] 1 2 3
```

```
y <- c("1", "2", "3", "2", "a")
as.numeric(y)
```

```
Warning: NAs introduced by coercion
```

```
[1]  1  2  3  2 NA
```

- NA代表Not available，代表著missing value

dsp.im

資料型態的轉換2

```
# 字串轉數字
a1 <- c("89", "91", "102")
as.numeric(a1)
```

```
[1]  89  91 102
```

```
# 布林轉數字
a2 <- c(TRUE, TRUE, FALSE)
as.numeric(a2)
```

```
[1] 1 1 0
```

```
# 數字轉布林
a3 <- c(-2, -1, 0, 1, 2) # 只有0會被轉成FALSE
as.logical(a3)
```

```
[1]  TRUE  TRUE FALSE  TRUE  TRUE
```

```
# 數字轉字串
as.character(a3)
```


Ch09: List 存放異質性資料的容器

List

```
x1 <- c("林志玲", 174, 52, TRUE)   # 所有元素都被轉換成字串
x1
```

```
[1] "林志玲" "174"    "52"     "TRUE"
```

```
x2 <- list("林志玲", 174, 52, TRUE) # 保留資料型態
str(x2)
```

```
List of 4
 $ : chr "林志玲"
 $ : num 174
 $ : num 52
 $ : logi TRUE
```

List 賦值/取值

```
x3 <- list(name=c("林志玲", "隋棠", "蔡依林"),
           height=c(174, 173, 158),
           weight=c(52, 48, 39),
           model=c(TRUE, TRUE, FALSE))

x3[[1]]
```

```
[1] "林志玲" "隋棠"   "蔡依林"
```

```
x3$name
```

```
[1] "林志玲" "隋棠"   "蔡依林"
```

```
x3[["name"]]
```

```
[1] "林志玲" "隋棠"   "蔡依林"
```

```
names(x3)
```

```
[1] "name"   "height" "weight" "model"
```


dsp.im

Ch10：DataFrame 資料表

資料表 data.frame

- `data.frame` 是資料分析時最基本的物件
- `R` 提供將外部資料轉成 `data.frame` 的功能
- 透過`data.frame` 可以進行以下功能：
  - 資料的整理
  - 圖形的繪製
  - 模型的配適與預測

世界上最常見的範例資料 iris

```
data("iris")
head(iris) # 列出前幾筆資料, 預設6筆
```

```
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
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
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
```

```
dim(iris)  # 列出資料表的 rows and columns
```

```
[1] 150   5
```

表格的取值 - 座標

- 類似於向量取值，在中括號 `[i, j]` 中進行取值
- 逗號的前後分別表示資料表的 row and column

```
iris[2, 3]
```

```
[1] 1.4
```

```
iris[1:6, 1:3]
```

```
  Sepal.Length Sepal.Width Petal.Length
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
```

表格的取值 - 列

欲選取第i筆觀察資料時，使用 `[i, ]` 在column欄位留白

```
iris[2, ]
```

```
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
2          4.9           3          1.4         0.2  setosa
```

```
iris[c(1, 51, 101),]
```

```
    Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
1            5.1         3.5          1.4         0.2     setosa
51           7.0         3.2          4.7         1.4 versicolor
101          6.3         3.3          6.0         2.5  virginica
```

表格的取值 - 欄

欲選取整欄資料時，有三種常用方法

```
iris[,1] # 欄位名稱未知
iris$Sepal.Length # 已知欄位名稱
iris[["Sepal.Length"]] # 已知欄位名稱
```

```
  [1] 5.1 4.9 4.7 4.6 5.0 5.4 4.6 5.0 4.4 4.9 5.4 4.8 4.8 4.3 5.8 5.7 5.4
 [18] 5.1 5.7 5.1 5.4 5.1 4.6 5.1 4.8 5.0 5.0 5.2 5.2 4.7 4.8 5.4 5.2 5.5
 [35] 4.9 5.0 5.5 4.9 4.4 5.1 5.0 4.5 4.4 5.0 5.1 4.8 5.1 4.6 5.3 5.0 7.0
 [52] 6.4 6.9 5.5 6.5 5.7 6.3 4.9 6.6 5.2 5.0 5.9 6.0 6.1 5.6 6.7 5.6 5.8
 [69] 6.2 5.6 5.9 6.1 6.3 6.1 6.4 6.6 6.8 6.7 6.0 5.7 5.5 5.5 5.8 6.0 5.4
 [86] 6.0 6.7 6.3 5.6 5.5 5.5 6.1 5.8 5.0 5.6 5.7 5.7 6.2 5.1 5.7 6.3 5.8
[103] 7.1 6.3 6.5 7.6 4.9 7.3 6.7 7.2 6.5 6.4 6.8 5.7 5.8 6.4 6.5 7.7 7.7
[120] 6.0 6.9 5.6 7.7 6.3 6.7 7.2 6.2 6.1 6.4 7.2 7.4 7.9 6.4 6.3 6.1 7.7
[137] 6.3 6.4 6.0 6.9 6.7 6.9 5.8 6.8 6.7 6.7 6.3 6.5 6.2 5.9
```

表格的取值 - 篩選

利用條件式做篩選

```
iris[iris$Sepal.Length > 5.5 & iris$Species=="setosa", ]
```

```
   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
15          5.8         4.0          1.2         0.2  setosa
16          5.7         4.4          1.5         0.4  setosa
19          5.7         3.8          1.7         0.3  setosa
```

Ch11: Pattern Matching and Replacement

關鍵字的取代與查找 - gsub

```
gsub(pattern, replacement, x)
```

```
year <- c("民國99", "民國100", "民國101")
gsub("民國", "", year)
```

```
[1] "99"  "100" "101"
```

```
as.numeric(gsub("民國", "", year)) + 1911
```

```
[1] 2010 2011 2012
```

關鍵字的取代與查找 - grep

- `grep(pattern, x, value=FALSE)`
- `grepl(pattern, x)`
- `grep(pattern, x, value=TRUE)`

```
title <- c("馬習會前交涉祕辛曝光", "馬說明馬習會：公布逐字稿不可思議的透明", "談22K政策朱立倫：不幸被企業濫用")
grep("馬習會", title)
```

```
[1] 1 2
```

```
grepl("馬習會", title)
```

```
[1]  TRUE  TRUE FALSE
```

```
grep("馬習會", title, value = TRUE)
```

```
[1] "馬習會前交涉祕辛曝光"
[2] "馬說明馬習會：公布逐字稿不可思議的透明"
```

關鍵字的取代與查找 - gregexpr

```
txt <- c("名模林志玲身高有174公分，體重52公斤", "女神蔡依林身高158公分，體重只有39公斤")
matches <- gregexpr("[0-9]+", txt)
regmatches(txt, matches)
```

```
[[1]]
[1] "174" "52"

[[2]]
[1] "158" "39"
```

Recape

- 取代：`gsub`
- 查找位置：`grep(value=FALSE)`, `grep(value=TRUE)`, `grepl`
- 查找結果：`grepexpr`

dsp.im

補充資料

- Learn R in R (Swirls)
- Text Processing (wikibooks)
- Introduction to R (around 4 hours)
- Cookbook for R

繼續學習之路

- 了解自己的需求，詢問關鍵字與函數
- Taiwan R User Group，mailing list: Taiwan-useR-Group-list@meetup.com
- ptt R_Language版
- R軟體使用者論壇
- StackOverflow
- 歡迎來信 leslie@dsp.im 或其他DSP優秀教師多多交流