



R Basic

第五組 讀取資料

黃昱璇、老賴、Vivian Chan、謝包妹

分配

1. 基本讀檔 read&write Vivian
2. 批次上傳 昱璇
3. 不同套件比較 昱璇
4. 編碼環境(台灣,美國) 包妹
5. 特殊資料上傳 包妹
6. 資料庫串連 包妹

1 基本讀檔 read&write

R 內建的資料

`data()` #列出所有可用的 dataset

`head(rock)` #直接叫出想要使用資料

讀取外部資料

`getwd()` #可以先確認目前的位置（工作目錄），若資料讀不進來也可能是編碼問題

`setwd()` #重新設定位置

1 基本讀檔 read&write

```
read.table( "SampleData.csv" , header = TRUE, sep = "\t" )
```

#header：資料是否包含欄位名稱；sep：設定字元的分隔

```
read.csv( ) read.csv2( )      #分隔符號不同
```

```
readLines( )                  #逐行讀取
```

補充1：資料量較大（ex:數值超過一百萬筆）時，使用 `fread()` 效果較佳

1 基本讀檔 read&write

```
write.csv(c, file = "SampleData2.csv" )
```

#用R把資料輸出成csv檔案，參數1：要輸出的物件，參數2：檔名

補充1：excel的預設編碼為big-5；R為 UTF-8

1 基本讀檔 read&write (讀Excel檔案) 建議直接轉成csv較方便

1. 確認資料路徑

2. 讀取檔案名稱

3. 指定檔案內的工作表(可用順序編號或者工作表名稱)

情境: 我要讀201707資料夾內的「進店問題統計.xlsx」的subset_Q2_1工作表裡的資料

```
1
2 library(gdata)      →xls檔案才能讀!!且Windows系統多用
3 library(xlsx)        RODBC::odbcConnectExcel (僅支持32-bit)
4 library(readxl)
5
6 # 讀 Excel
7 dir<-setwd("D:/Dana/Others/RBasic/201707")
8 ##xlsx|
9 df<-read.xlsx("進店問題統計.xlsx",sheetIndex=3)
10 ##readxl
11 my_data <- read_excel("進店問題統計.xlsx", sheet = "subset_Q2_1")
12 my_data <- read_excel("進店問題統計.xlsx", sheet = 2)
13
```

2 批次上傳

情境: 我要上傳五個csv檔，並命名欄位名稱，最後將五個檔案接續一起

```
D1<-read.csv("201702.csv")
D2<-read.csv("201703.csv")
D3<-read.csv("201704.csv")
D4<-read.csv("201705.csv")
D5<-read.csv("201706.csv")

#重新命名欄位
colnames(D1) <- c("Subscriber_No","MINING_DW_SUBSCR_NO","Date","Time","Store_No","St
colnames(D2) <- c("Subscriber_No","MINING_DW_SUBSCR_NO","Date","Time","Store_No","St
colnames(D3) <- c("Subscriber_No","MINING_DW_SUBSCR_NO","Date","Time","Store_No","St
colnames(D4) <- c("Subscriber_No","MINING_DW_SUBSCR_NO","Date","Time","Store_No","St
colnames(D5) <- c("Subscriber_No","MINING_DW_SUBSCR_NO","Date","Time","Store_No","St
|
#整併
All<-rbind(D1,D2,D3,D4,D5)
```

<--原始作法<一行行寫>

思考: 若要100個以上檔案，我要寫100列???

使用迴圈-->

改以for迴圈讓程式自行讀檔，步驟:

- 1)先列出檔案清單dir()
- 2)檔案關鍵字共通性，2017開頭
- 3)針對符合條件的檔案，執行讀檔、建欄位與合併列

```
#匯入每月進店資料
dir_files<-dir(path = dir)
All<-data.frame()
for( i in (1:length(dir_files))){
  if (regexpr("2017",dir_files[i])[1]--1){
    D<-read.csv(dir_files[i])
    colnames(D) <- c("Subscriber_No","MINING_DW_SUBSCR_NO","Date","Time","Store_No"
    All<-rbind(All,D) )
  }
  else{
    next
  }
}
```

3 不同套件比較

base::read.table/read.csv 基本

data.table::fread

readr::read_csv

情境: 我想讀All.csv這個檔案(筆數約60幾萬)

```
52 # save as csv file
53 write.csv(All,"All.csv")
54
55 # 不同套件比較
56
57 read_All<-read.csv("All.csv")
58
59 library(data.table)
60 library(readr)
61
62 read_All2<-fread("All.csv")
63 read_All3<-read_csv("All.csv")
```

勝

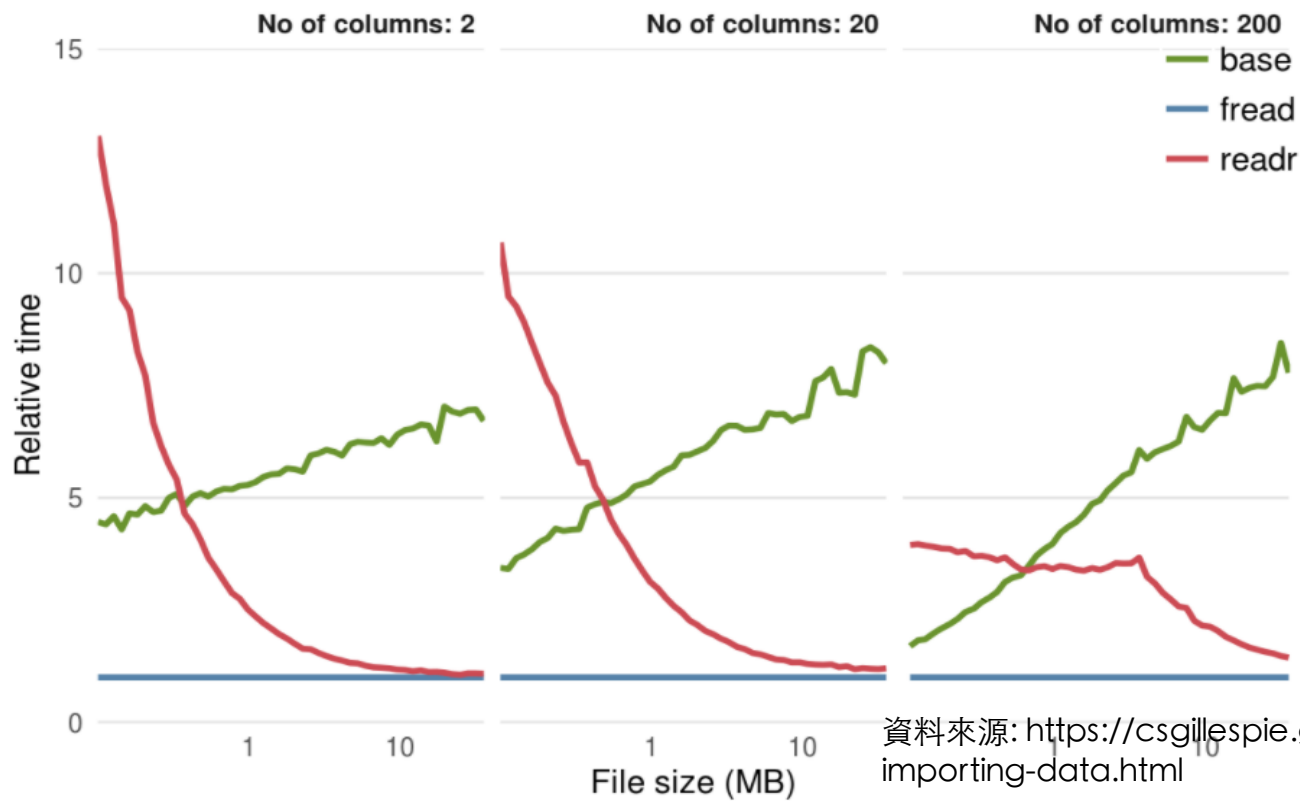
比較:

read.table 14秒

fread 2秒

read_csv 4秒

3 不同套件比較



資料來源: <https://csgillespie.github.io/efficientR/5-3-importing-data.html>

4 編碼環境

不管怎麼圖檔都這樣

	X.Á..	°p.î.Ĭ	Á..p°Ê.ú...Æ
1	104!~1#ē!Ü3#ē	Xp	12
2	104!~1#ē!Ü3#ē	²Ä#@Äp°b°î°Ĭ	2
3	104!~1#ē!Ü3#ē	²Ä#GÄp°b°î°Ĭ	4
4	104!~1#ē!Ü3#ē	²Ä#TÄp°b°î°Ĭ	4
5	104!~1#ē!Ü3#ē	²Ä# Äp°b°î°Ĭ	2

希望變成這樣

	期間	管制區	總計監測站數
1	104年1月至3月	合計	12
2	104年1月至3月	第一類管制區	2
3	104年1月至3月	第二類管制區	4
4	104年1月至3月	第三類管制區	4
5	104年1月至3月	第四類管制區	2

Rstudio作業系統編碼不同，Windows 的中文編碼是 big5，而 Linux / Mac 都是 UTF-8

4 編碼環境

已經加上encoding = 'UTF-8' 怎麼會這樣？

	X.U.00B4..U.0076..U.00A1.	X.U.00BA..U.07A8..ee..U.00B0..cf.
1	104<U+00A6>~1<U+00A4><eb><U+00A6><dc>3<U+00...	<U+00A6>X<U+00AD>p
2	104<U+00A6>~1<U+00A4><eb><U+00A6><dc>3<U+00...	<U+00B2>H@<c3><fe><U+00BA><U+07A8><ee><U+00...
3	104<U+00A6>~1<U+00A4><eb><U+00A6><dc>3<U+00...	<U+00B2>HG<c3><fe><U+00BA><U+07A8><ee><U+00...
4	104<U+00A6>~1<U+00A4><eb><U+00A6><dc>3<U+00...	<U+00B2>HT<c3><fe><U+00BA><U+07A8><ee><U+00B...
5	104<U+00A6>~1<U+00A4><eb><U+00A6><dc>3<U+00...	<U+00B2>h <c3><fe><U+00BA><U+07A8><ee><U+00B...

很正常，因為你的Windows 的中文編碼還是 big5

4 編碼環境

方法一 善用notepad++



方法二 `read.csv('./data.csv' , encoding= 'UTF-8')`

方法三

`Sys.getlocale(category = "LC_ALL")` # 看你的RStudio環境設定

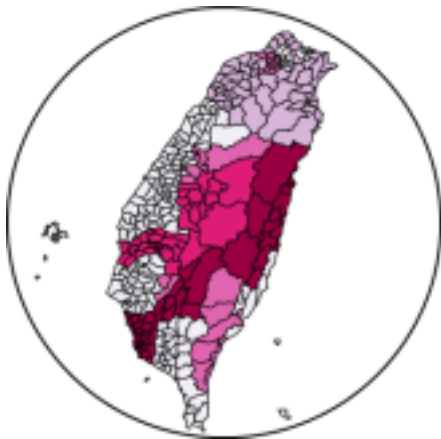
`Sys.setlocale(category = "LC_ALL", locale = "cht")` # 英文換中文

`Sys.setlocale(category = "LC_ALL", locale = "zh_TW.UTF-8")` # Mac 使用者英文換中文

`Sys.setlocale(category = "LC_ALL", locale = "English_United States.1252")` # 中文換英文

```
> Sys.getlocale(category="LC_ALL")
[1] "LC_COLLATE=English_United States.1252;LC_CTYPE=English_United States.1252;
LC_MONETARY=English_United States.1252;LC_NUMERIC=C;LC_TIME=English_United States.1252"
> Sys.setlocale(category = "LC_ALL", locale = "cht")
[1] "LC_COLLATE=Chinese (Traditional)_Taiwan.950;LC_CTYPE=Chinese (Traditional)_Taiwan.950;LC_MONETARY=Chinese (Traditional)_Taiwan.950;LC_NUMERIC=C;LC_TIME=Chinese (Traditional)_Taiwan.950"
```

5 特殊資料讀取



Shape file



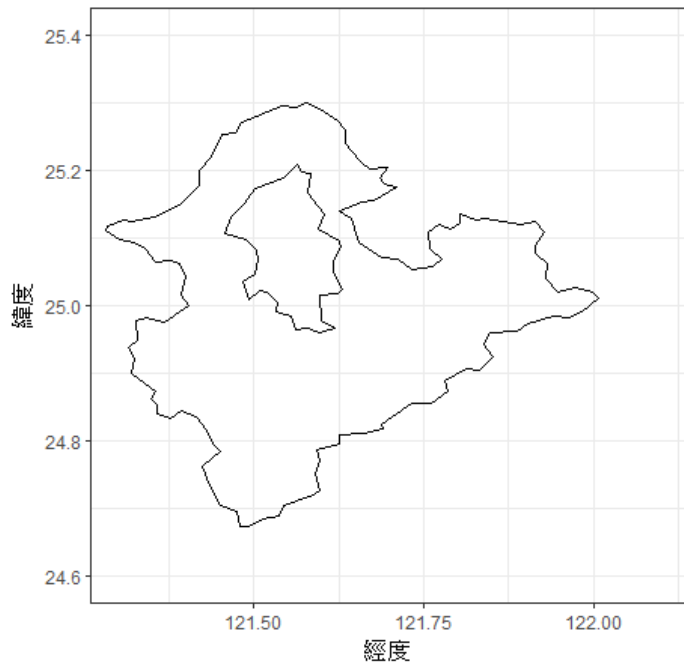
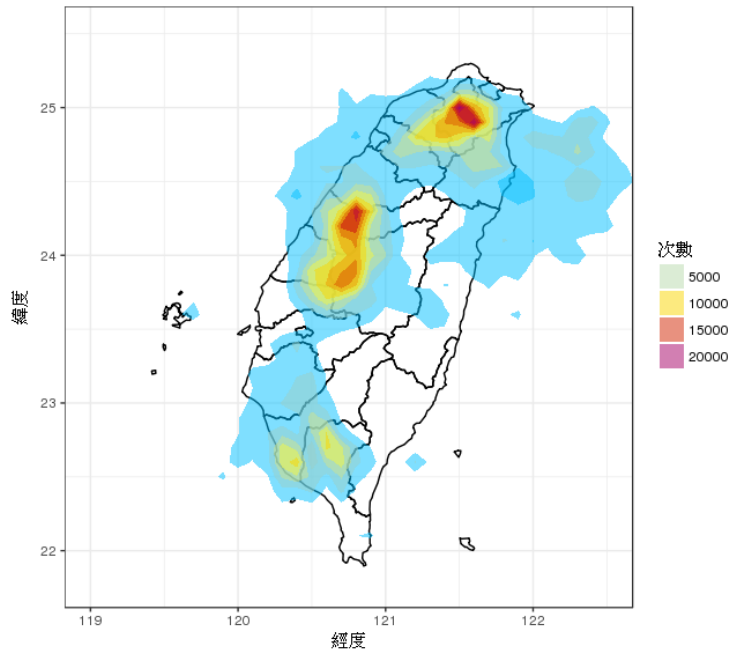
JSON { }



MongoDB

5 特殊資料讀取 shape file

過去十年台灣周圍所有閃電分佈

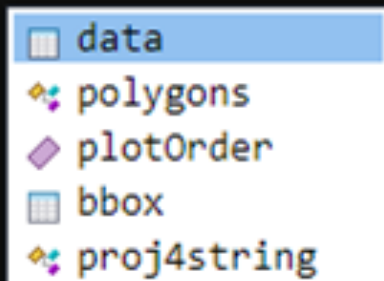


5 特殊資料讀取 shape file

```
library(maptools)
#### 縣市邊界 ####
county.shp = readShapePoly("./shp/County_MOI_1041215.shp", IDvar = "County_ID")
```

```
class(county.shp)
[1] "SpatialPolygonsDataFrame"
attr(,"package")
[1] "sp"
```

```
county.shp@
```



```
data
polygons
plotOrder
bbox
proj4string
```

必備檔案

- .shp — 圖形格式。
- .shx — 圖形索引格式。
- .dbf — 屬性資料格式。

Name	Size
..	
County_MOI_1041215.dbf	12.3 KB
County_MOI_1041215.prj	145 B
County_MOI_1041215.shp	31.5 KB
County_MOI_1041215.shx	276 B

5 特殊資料讀取 JSON

```
#### JSON ####  
library(jsonlite)  
url <- "./apiIn.json"  
json_data <- fromJSON(url)
```

	Station	Destination	UpdateTime
1	大安站	南港展覽館站	2017-07-26T18:13:23
2	大安站	動物園站	2017-07-26T18:13:23
3	大安站	象山站	2017-07-26T18:13:16.277
4	大直站	南港展覽館站	2017-07-26T18:13:23
5	大湖公園站	動物園站	2017-07-26T18:13:23

5 特殊資料讀取 MongoDB

rmongodb::mongo.create

mongolite::mongo

```
#### MongoDB ####
# 3.0 版以下
library(rmongodb)

mongo <- mongo.create(host="xx.xxx.xx.xxx", db="DB 名稱",
                      username="帳號", password="密碼")
res1 <- mongo.find.all(mongo, "Collection 名稱",
                      fields='{"_id":0,"SiteId":1}', limit = 100)

# 3.0 版以上
library(mongolite)
con1 <- mongolite::mongo(collection = "DB 名稱",
                          url = "mongodb://帳號:密碼@xx.xxx.xx.xxx/Collection 名稱")
gis <- con1$find(query = '{"name" : { "$regex" : ".*桃園市*." },
                      "types" : {"$in" : ["weather station", "rain station"]}}',
                  fields = '{"name" : 1, "types" : 1, "location" : 1}')
```

6 資料庫串連

MySQL

```
67 library(RMySQL)
68 mydb = dbConnect(MySQL(), user='xxuser', password='xxr', dbname='xxprod', host='xx.xx.xx.xx')
69
```

詳情參考: <https://cran.r-project.org/web/packages/RMySQL/RMySQL.pdf>

Teradata

```
2 library("RODBC")
3 ch <- odbcConnect(dsn="10.68.64.138",uid="u_daXXXXX",pwd="DJXXXX")
4 NET_ALL<-sqlQuery(ch,paste("select MINING_DW_SUBSCR_NO, DATA_MONTH,
5 VOICE_RC_AMT , DATA_RC_AMT, ONNET_AMT, OFFNETM_AMT, ONNET_AMT, OFFNETM_AMT,
6 PSTN_AMT, INT_ROAM_AMT
7
8 From MDS_MART.MDS_ACTIVE_MLY_ORIG
9 WHERE MINING_DW_SUBSCR_NO=8132924013"))
10
```