R軟體簡介及其在 Data Mining 之應用

淡江大學統計系　陳景祥

steve@stat.tku.edu.tw

「R軟體：應用統計方法」作者

R-Web 雲端計算軟體團隊

國際技能競賽「資訊與網路技術」類組國際裁判

Steve Chen實務經驗

- 大型健保醫療資料處理與分析
- 銀行信用卡客戶評分表系統
- 金融機構土地質押貸款估價系統
- NetStat 線上統計計算網站

  (http://netstat.stat.tku.edu.tw)
- R-Web 雲端資料運算系統(含 Data Mining)

  (http://www.r-web.com.tw)
- SPSS+R：醫學存活分析外加模組 8 個
- API 鼎利電通：媒體(廣告)投資模擬器軟體
- Upcoming: 協助台灣某大研究機構建置數百至上千台 R 平行運算系統

大綱

- R軟體簡介
- 資料探勘簡介
- R：分群技術
- R：分類技術
- R：類神經網路(ANN)
- R：關聯規則分析(Association Analysis)
- R and Big Data

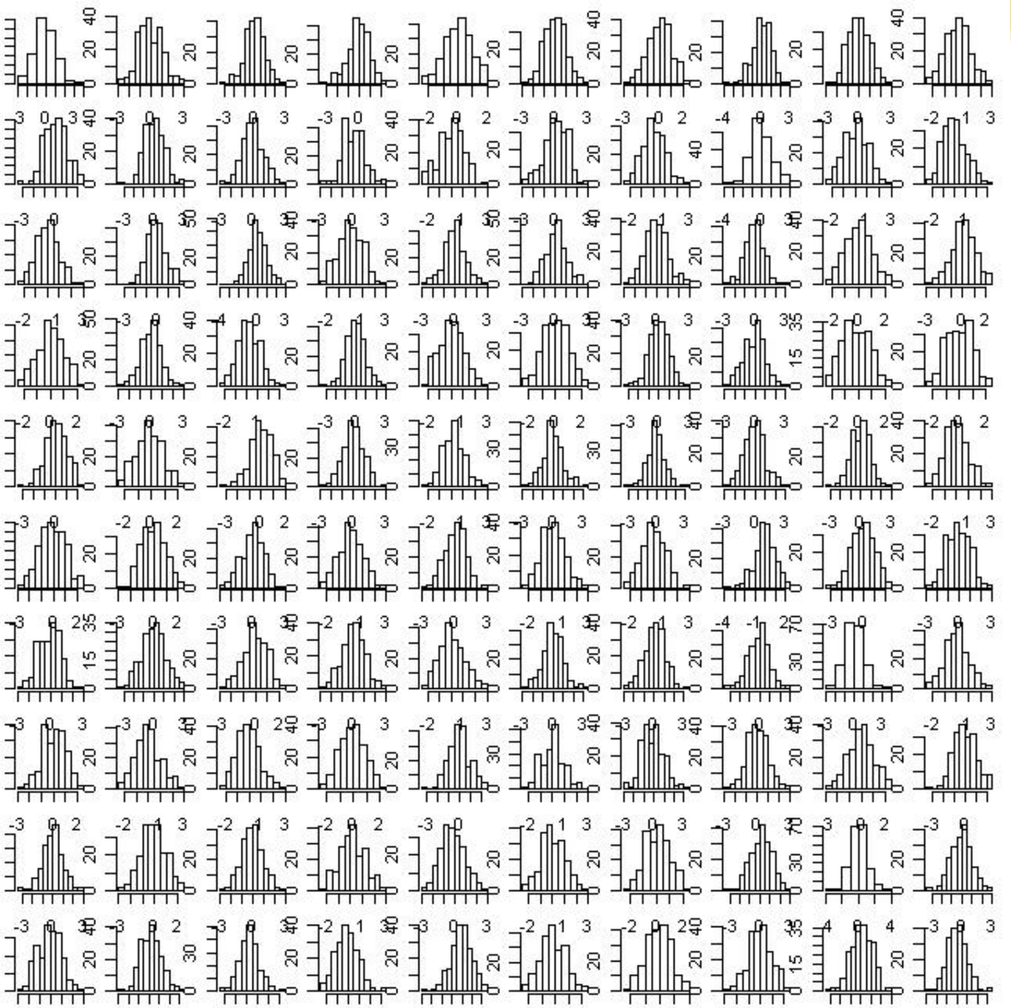
R 軟體的特色

- 完整的程式語言功能
- Vector 與 Array 運算導向
- 與統計領域直接對應的變數型態
- 函數(function)與套件(package)為主要單元
- 強大的繪圖功能
- 活躍且龐大的套件(package)發展與更新
- R程式可以使用 C, Fortran, Java 等程式
- 平行運算(Parallel Computing)
- 樂高玩具特質：可打照出自己的 R 環境

R 與 SAS、SPSS 的不同

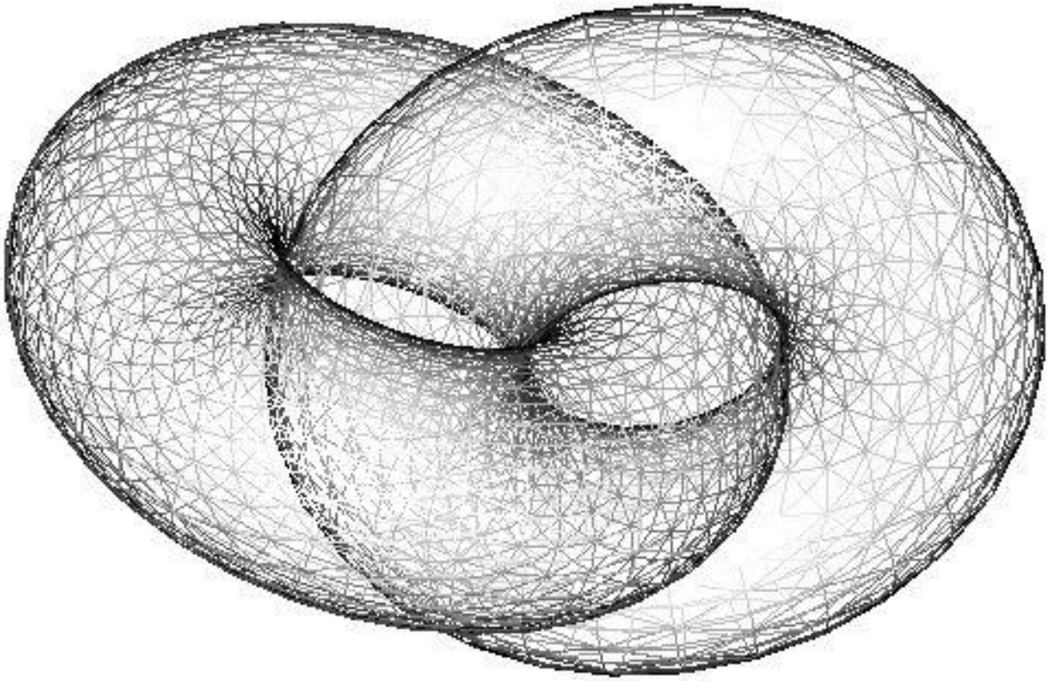
功能	R	SAS	SPSS
程式語言功能	完整	不完整	不完整
繪圖功能	強悍	普通	普通
應用最新研究結果	快	慢	慢
分析模組數目	非常多(5900個)	多	普通
使用介面客製化	容易	麻煩	麻煩
輸出成果客製化	容易	很難	很難
搭配使用其他語言或軟體的功能	強	稀少	稀少
價格	0	昂貴/每年版權費	昂貴

R軟體強大的繪圖功能(1)

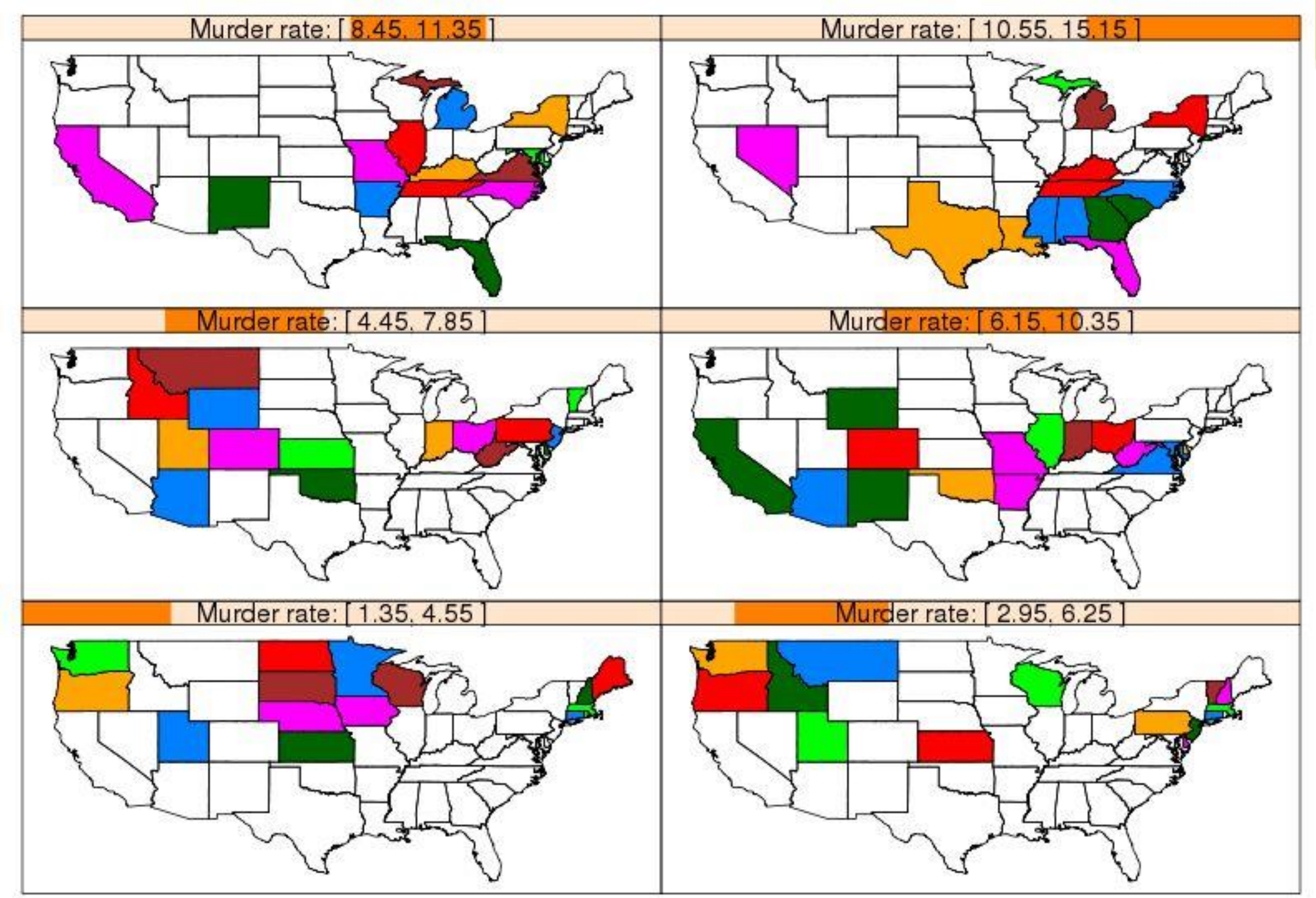


R軟體強大的繪圖功能(2)

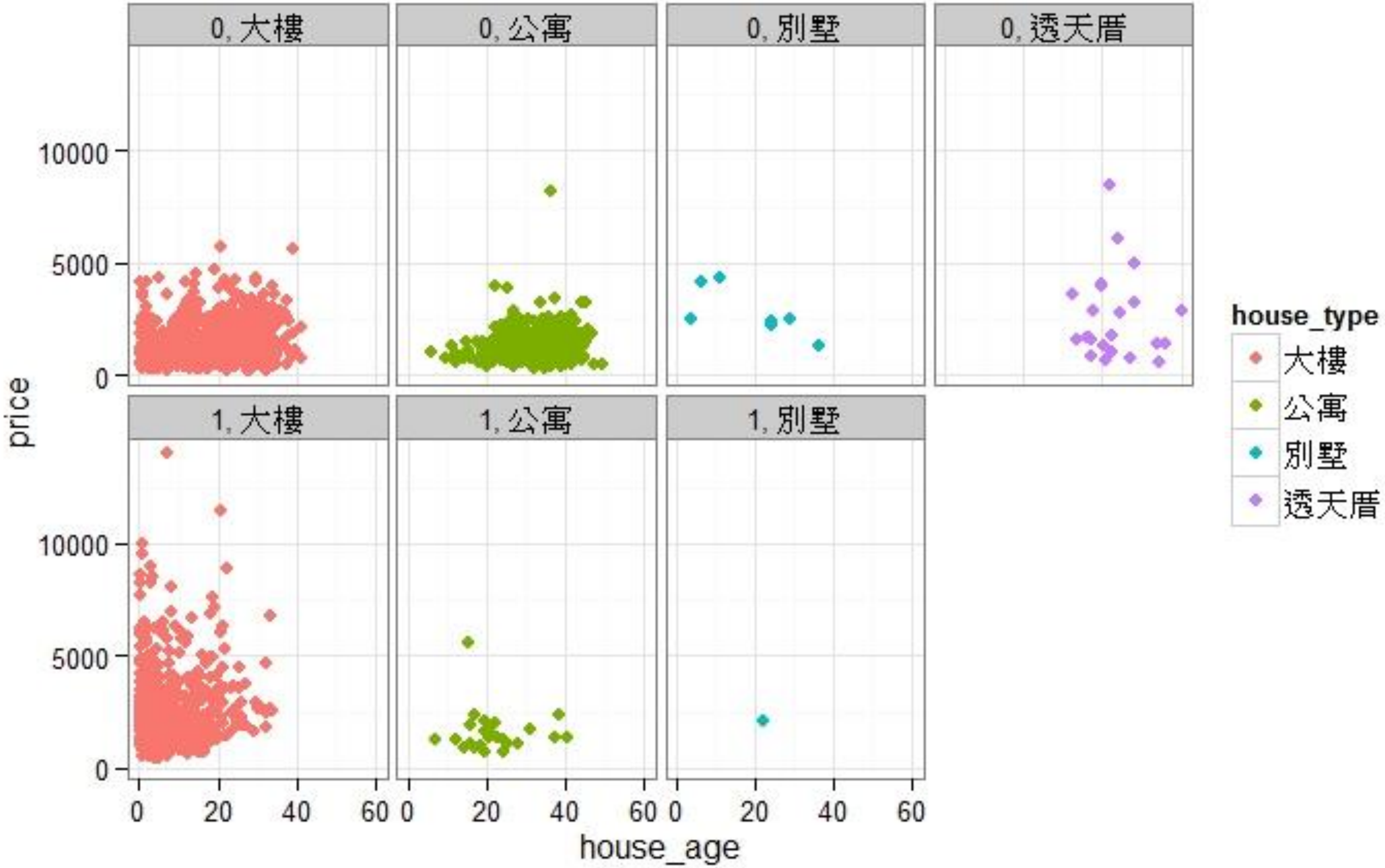
3D 即時互動圖形：可用滑鼠控制3D旋轉



R軟體強大的繪圖功能(3)

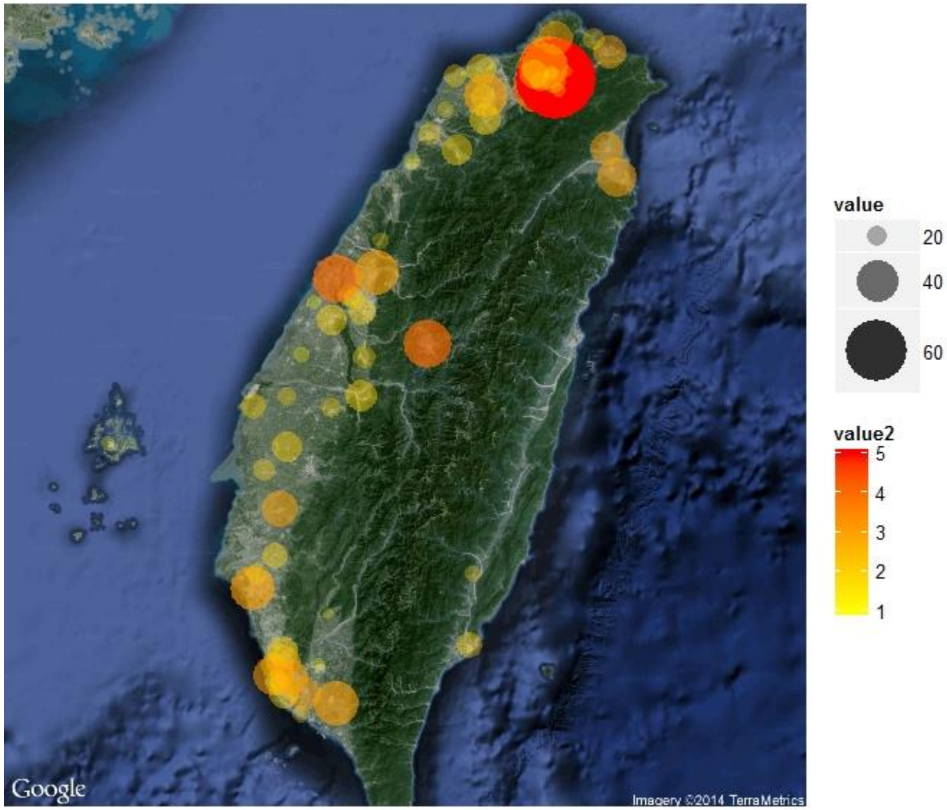

Murder rate: [8.45, 11.35]
Murder rate: [10.55, 15.15]
Murder rate: [4.45, 7.85]
Murder rate: [6.15, 10.35]
Murder rate: [1.35, 4.55]
Murder rate: [2.95, 6.25]

R軟體強大的繪圖功能(4)



R軟體強大的繪圖功能(5)

台灣懸浮微粒濃度(PM10)資料



R程式範例：迴歸分析

```
# 以 IQ 預測 成績
students = read.csv("d:/mydir/students.csv")
result = lm(scores ~ IQ, data=students)
summary(result )
```

```
Call:
lm(formula = scores ~ IQ)
Residuals:
      1       2       3       4       5       6 
 2.4883 -1.0897  0.6060 -0.7132 -0.4453 -0.8461 
Coefficients:
             Estimate   Std. Error   t value    Pr(>|t|)
(Intercept)  -2.69628    5.05144     -0.534     0.621786
IQ            0.67207    0.04476     15.014     0.000115 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.514 on 4 degrees of freedom
Multiple R-squared: 0.9826,     Adjusted R-squared: 0.9782
F-statistic: 225.4 on 1 and 4 DF,  p-value: 0.0001147
```

R軟體應用的領域 (1)

- Bayesian Inference 貝氏統計方法
- Chemometrics and Computational Physics 化學與物理
- Clinical Trial Design, Monitoring, and Analysis 臨床實驗分析
- Cluster Analysis & Finite Mixture Models 集群分析
- Probability Distributions 機率分配
- Computational Econometrics 計量經濟
- Analysis of Ecological and Environmental Data 生態與環境分析
- Design of Experiments (DoE) & Analysis of Experimental Data 實驗設計
- Empirical Finance 財政實務分析

R軟體應用的領域 (2)

- Statistical Genetics 基因統計
- Graphic Displays & Dynamic Graphics & Graphic Devices & Visualization 圖形分析
- gRaphical Models in R 圖形模組
- **High-Performance and Parallel Computing**

  高效率運算與平行運算
- **Machine Learning & Statistical Learning**

  機器學習、資料探勘
- Medical Image Analysis 醫學影像分析
- Multivariate Statistics 多變量分析
- Natural Language Processing 自然語言分析

R軟體應用的領域 (3)

- Official Statistics & Survey Methodology 政府統計調查
- Optimization and Mathematical Programming 函數最佳化
- Analysis of Pharmacokinetic Data 藥物動力學分析
- Phylogenetics 系統發生學
- Psychometric Models and Methods 心理學測量分析
- Reproducible Research 實驗複製分析
- Robust Statistical Methods 強韌統計方法
- Statistics for the Social Sciences 社會科學統計
- Analysis of Spatial Data 空間統計
- Survival Analysis 存活分析、可靠度分析
- Time Series Analysis 時間數列

Example:舊金山購物商場客戶

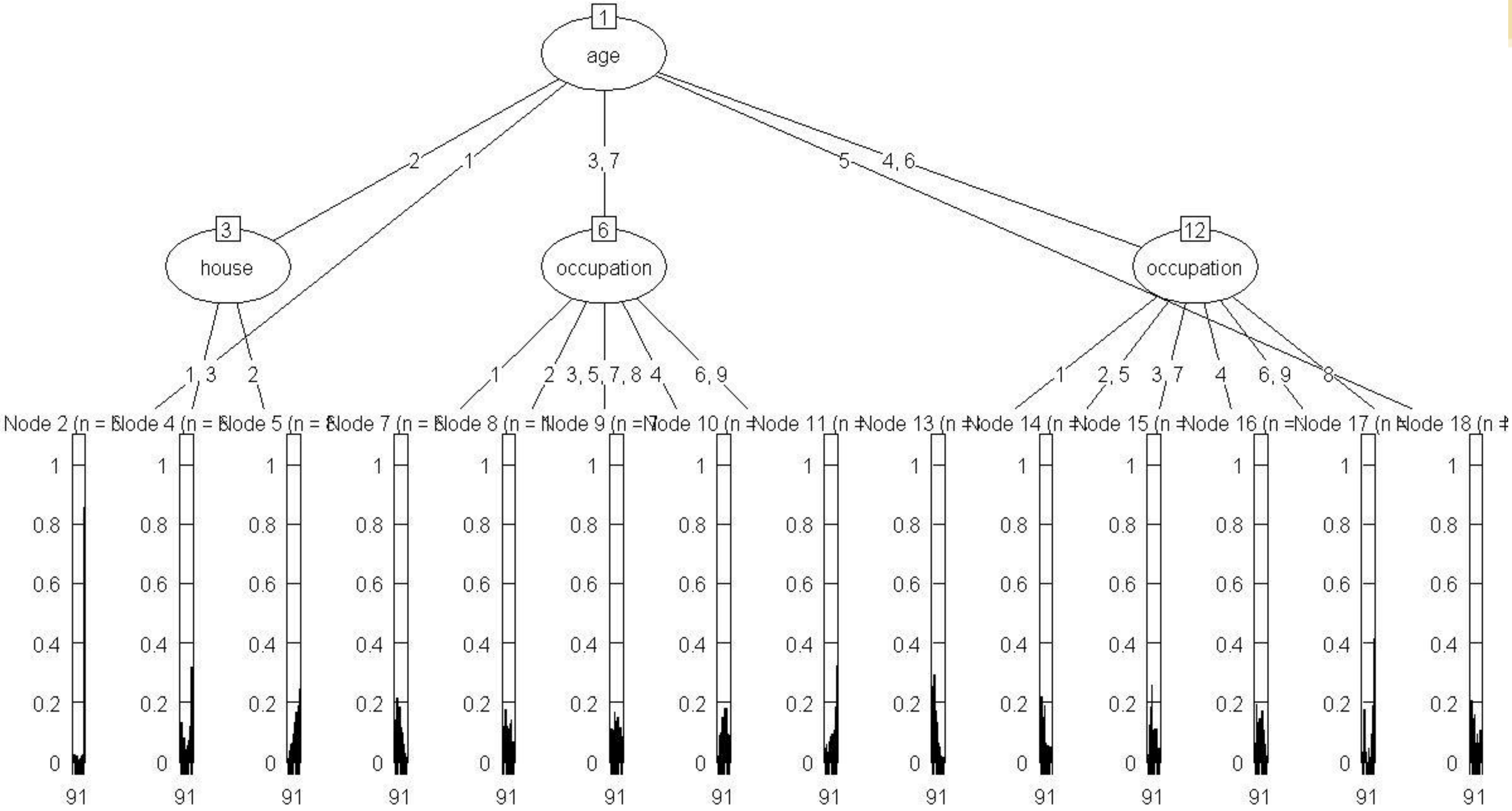
舊金山海灣區(Bay Area) Shopping Mall
客戶問卷資料

9409 個受訪者，14 個人口統計變數

目標變數：income

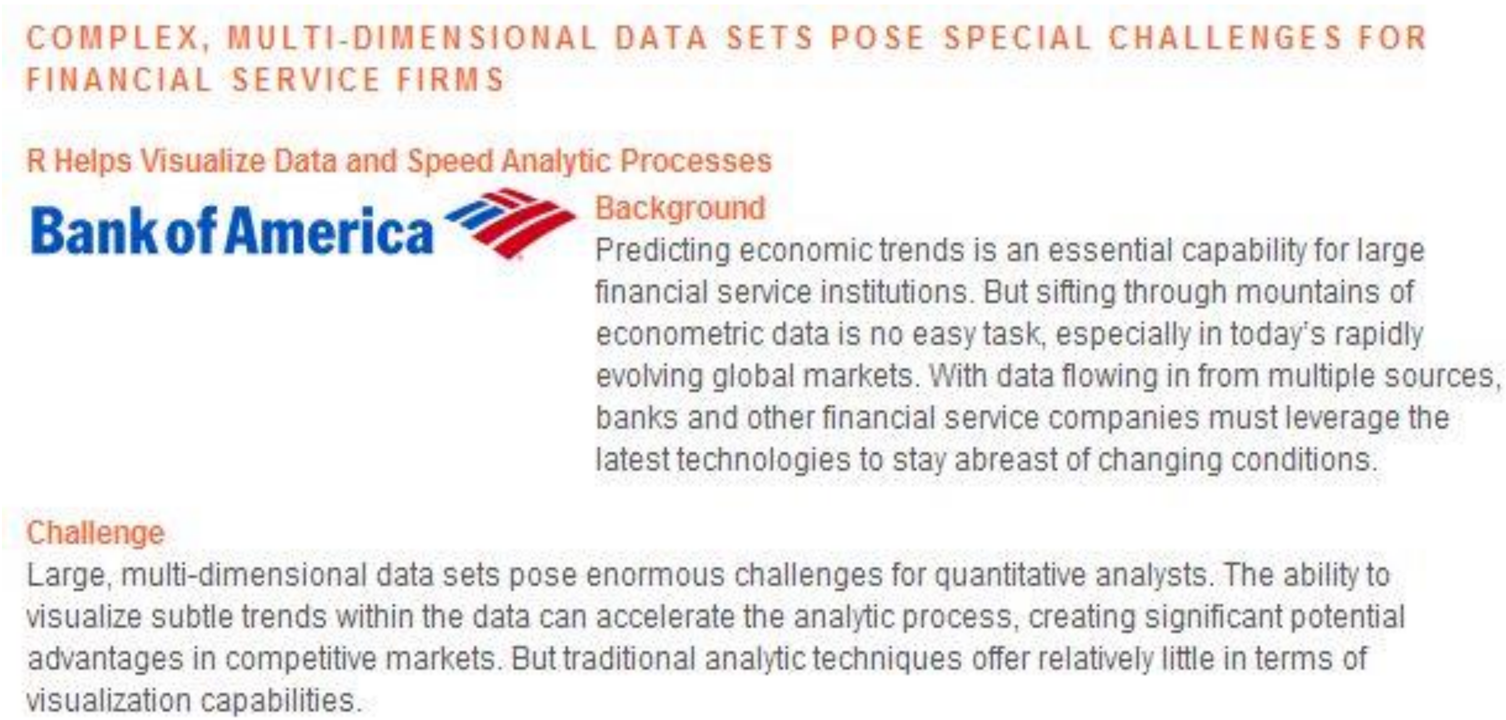
解釋變數：sex, marital, age, education, occupation, livetime, dualincome, persons, young, house, hometype, ethnic, language

CHAID 決策樹分析收入因素



應用：Bank of America

高維度資料圖形顯示、模型分析



COMPLEX, MULTI-DIMENSIONAL DATA SETS POSE SPECIAL CHALLENGES FOR FINANCIAL SERVICE FIRMS

R Helps Visualize Data and Speed Analytic Processes

Bank of America

Background

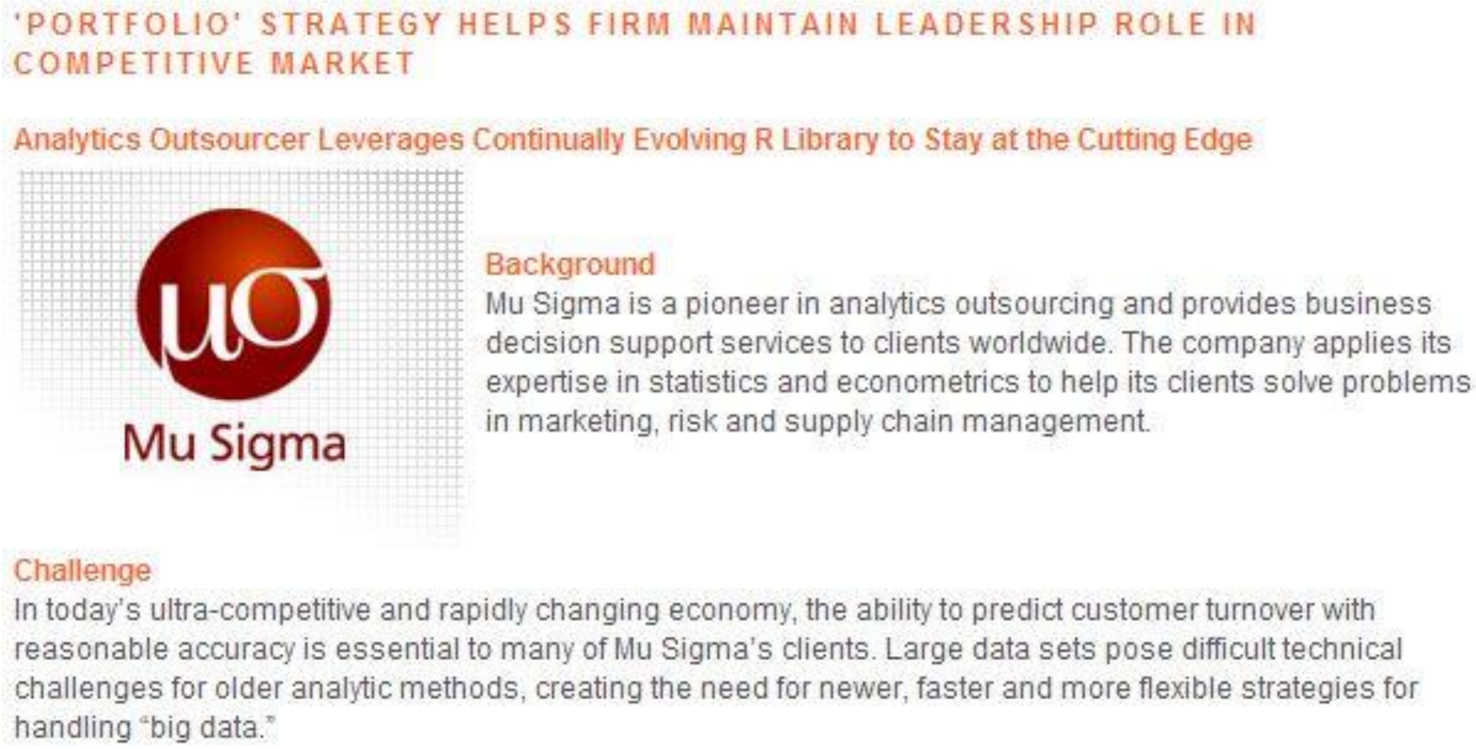
Predicting economic trends is an essential capability for large financial service institutions. But sifting through mountains of econometric data is no easy task, especially in today's rapidly evolving global markets. With data flowing in from multiple sources, banks and other financial service companies must leverage the latest technologies to stay abreast of changing conditions.

Challenge

Large, multi-dimensional data sets pose enormous challenges for quantitative analysts. The ability to visualize subtle trends within the data can accelerate the analytic process, creating significant potential advantages in competitive markets. But traditional analytic techniques offer relatively little in terms of visualization capabilities.

應用：Mu Sigma 決策顧問公司

最佳決策組合與客戶轉向預測分析



'PORTFOLIO' STRATEGY HELPS FIRM MAINTAIN LEADERSHIP ROLE IN COMPETITIVE MARKET

Analytics Outsourcer Leverages Continually Evolving R Library to Stay at the Cutting Edge

Background

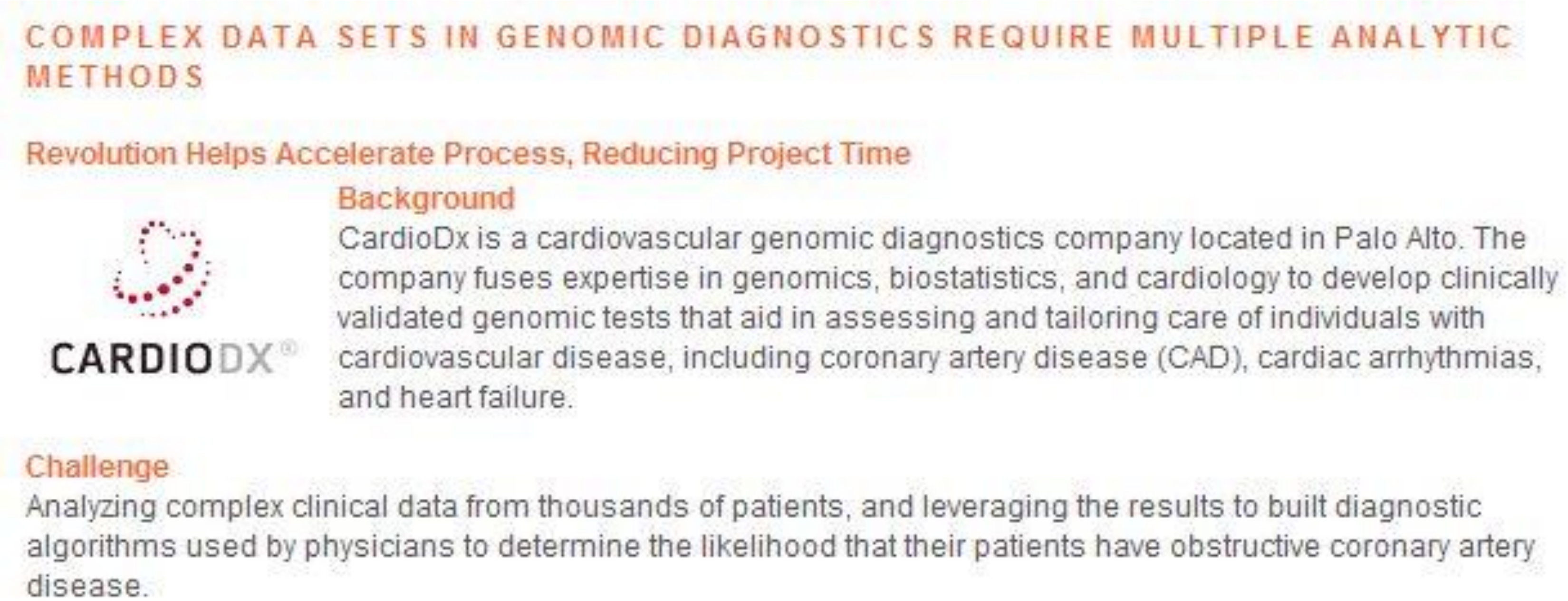
Mu Sigma is a pioneer in analytics outsourcing and provides business decision support services to clients worldwide. The company applies its expertise in statistics and econometrics to help its clients solve problems in marketing, risk and supply chain management.

Challenge

In today's ultra-competitive and rapidly changing economy, the ability to predict customer turnover with reasonable accuracy is essential to many of Mu Sigma's clients. Large data sets pose difficult technical challenges for older analytic methods, creating the need for newer, faster and more flexible strategies for handling "big data."

應用：CardioDx基因檢測公司

心血管疾病相關的基因檢測研究



COMPLEX DATA SETS IN GENOMIC DIAGNOSTICS REQUIRE MULTIPLE ANALYTIC METHODS

Revolution Helps Accelerate Process, Reducing Project Time

Background

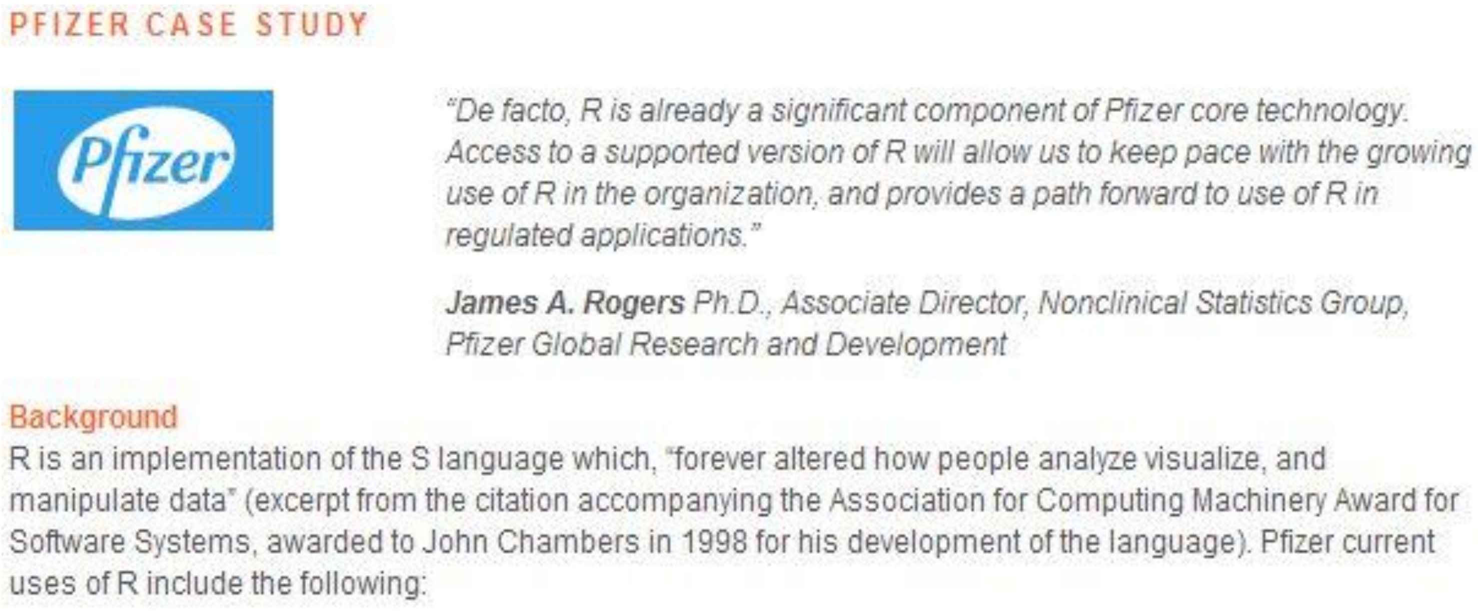
CardioDx is a cardiovascular genomic diagnostics company located in Palo Alto. The company fuses expertise in genomics, biostatistics, and cardiology to develop clinically validated genomic tests that aid in assessing and tailoring care of individuals with cardiovascular disease, including coronary artery disease (CAD), cardiac arrhythmias, and heart failure.

Challenge

Analyzing complex clinical data from thousands of patients, and leveraging the results to built diagnostic algorithms used by physicians to determine the likelihood that their patients have obstructive coronary artery disease.

應用：Pfizer (輝瑞)研究中心

基因資料分析、MicroArray 資料分析



PFIZER CASE STUDY

"De facto, R is already a significant component of Pfizer core technology. Access to a supported version of R will allow us to keep pace with the growing use of R in the organization, and provides a path forward to use of R in regulated applications."

***James A. Rogers** Ph.D., Associate Director, Nonclinical Statistics Group, Pfizer Global Research and Development*

Background

R is an implementation of the S language which, "forever altered how people analyze visualize, and manipulate data" (excerpt from the citation accompanying the Association for Computing Machinery Award for Software Systems, awarded to John Chambers in 1998 for his development of the language). Pfizer current uses of R include the following:

應用：澳洲國稅局(1)

- Australian Taxation Office — Case Study
- 全澳洲共有 22,000 員工
- Revenue Collection and Refund Management
- Compliance and Risk Modelling
- 12M Individuals, $450B Income, $100B Tax
- 2M Companies..., $1800B Income, $40B Tax

2005 年改用 R 軟體分析資料！

應用：澳洲國稅局(2)

主要任務：

- High Risk Refunds
- Required to Lodge ($110M)
- Assessing Levels of Debt
- Propensity to Pay
- Capacity to Pay
- Determining Optimal Treatment Strategies
- Identity Theft — eTax and International
- Project Wickenby Text Mining

德國 Fraunhofer 財經顧問公司

- 60 家分支機構、80 個研究單位
- 18000 個員工，年預算 1.65 億歐元
- http://www.fraunhofer.org

A case study on using generalized additive models to fit credit rating Scores (客戶信用評分卡系統)

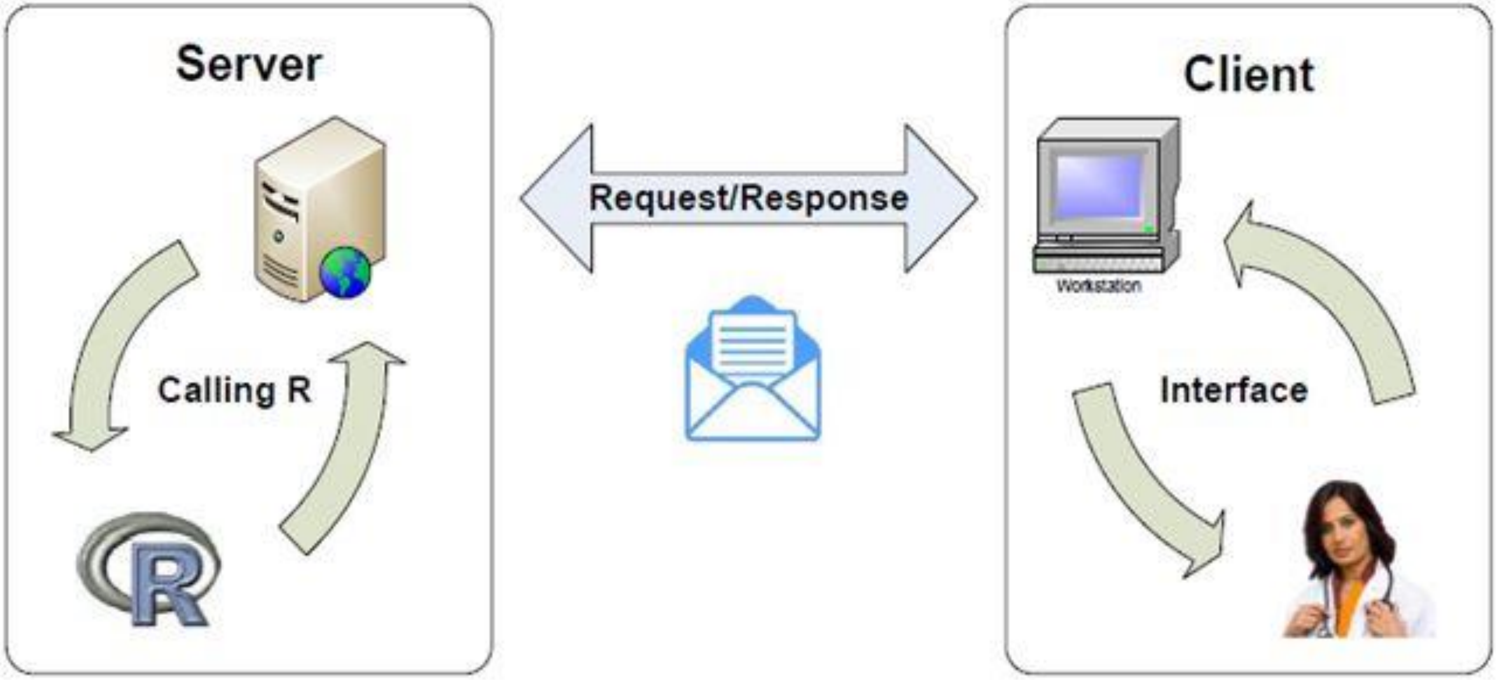
(Marlene Müller,
marlene.mueller@itwm.fraunhofer.de)

R在台灣的應用

- 官方健保資料庫(NHIP)分析與彙整
- 醫學統計，公共衛生分析
- 科學領域、平行運算
- 工程領域運算、晶圓廠品管控制
- 經濟與金融資料的時間數列分析
- 客戶關係管理、客戶信用評估
- 雲端資料運算與資料庫管理
- 任何涉及大量資料分析的領域或行業

應用：台灣健保門診資料庫

提供醫師與研究者遠端資料分析服務



Short Summary about R

優點：

- 免費、自由
- 程式語言功能完整
- 彈性：可搭配其他程式語言或函數庫
- 擴充性：有多個平行處理模組可用
- 與最新統計理論技術或其他技術接軌

缺點：

- 中文文件或教科書較少、初學者學習曲線陡峭
- 基本的R軟體運算受限於記憶體的大小
- 尚無完整的圖形使用者介面(GUI)
- 還沒有完整齊全的Big Data Solutions.

資料探勘(Data Mining)

使用自動或半自動化的技術，從大量資料中挖掘出有用的潛在規則或模式

Key Points:

- 自動或半自動化
- 大量資料？其實少量資料一樣可以分析
- 有用(useful)？需要人類主觀判斷

資料探勘：常用的技術分類

- 預測(Prediction)或估計(Estimation)
- 集群(Clustering)
- 分類(Classification)
- 規則(Association)或序列(Sequence)

預測技術(Prediction)

- 一般統計估計與檢定
- 迴歸分析
- 時間數列
- 迴歸樹 (Regression Trees)
- 具有預測功能的類神經網路(ANN)技術

集群技術(Cluster Analysis)

- K-Means 集群分析
- Hierarchical 集群分析
- Fuzzy 集群分析
- SOM (Self-Organizing Map)網路

其他: e.g.變數分群(因素分析,Factor Analysis)

規則/序列挖掘

- 關聯分析(Association Analysis)

  {尿布}->{啤酒}

  {廚房用品、美容用品}->{Sony相機}

- 序列探索(Sequence Discovery)

iris (鳶尾花)範例資料檔

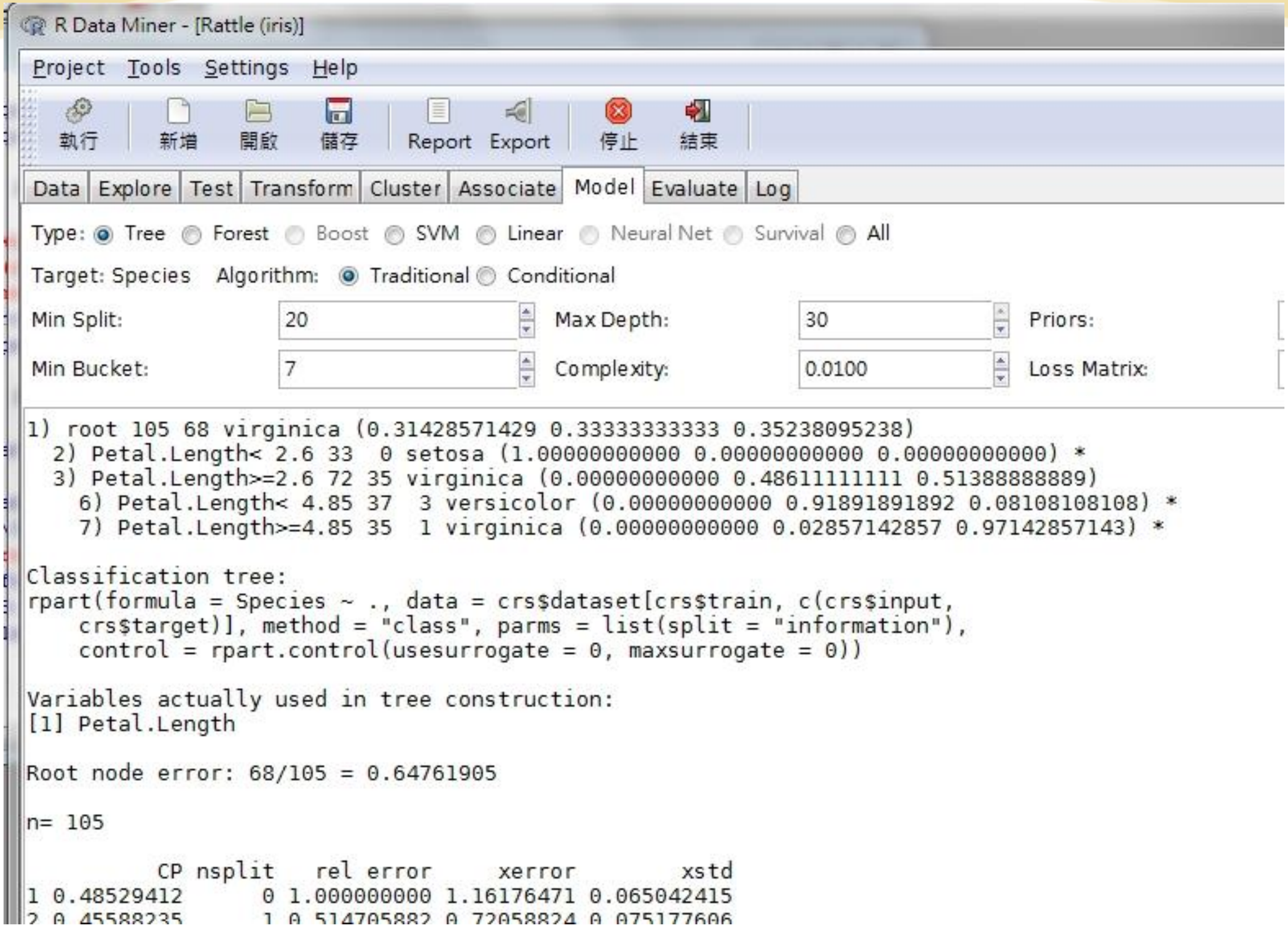
150 朵鳶尾花，每一朵花有花萼長度、花萼寬度、花瓣長度、花瓣寬度、品種共 5 個變數

```
> iris
        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              ..........................................................
149           6.2           3.4           5.4           2.3        virginica
150           5.9           3.0           5.1           1.8        virginica
```

```
> iris$Species
  [1] setosa    setosa    setosa    setosa    setosa    setosa
  ...................................................................
[145] virginica  virginica  virginica  virginica  virginica  virginica

Levels: setosa versicolor virginica
```

R: rattle 套件

library(rattle) ; rattle()



R Data Miner - [Rattle (iris)]
Project Tools Settings Help
執行
新增
開啟
儲存
Report
Export
停止
結束
Data Explore Test Transform Cluster Associate Model Evaluate Log

Type: ● Tree ○ Forest ○ Boost ○ SVM ○ Linear ○ Neural Net ○ Survival ○ All

Target: Species Algorithm: ● Traditional ○ Conditional

Min Split: 20 Max Depth: 30 Priors:

Min Bucket: 7 Complexity: 0.0100 Loss Matrix:

```
1) root 105 68 virginica (0.31428571429 0.33333333333 0.35238095238)
  2) Petal.Length< 2.6 33  0 setosa (1.00000000000 0.00000000000 0.00000000000) *
  3) Petal.Length>=2.6 72 35 virginica (0.00000000000 0.48611111111 0.51388888889)
    6) Petal.Length< 4.85 37  3 versicolor (0.00000000000 0.91891891892 0.08108108108) *
    7) Petal.Length>=4.85 35  1 virginica (0.00000000000 0.02857142857 0.97142857143) *

Classification tree:
rpart(formula = Species ~ ., data = crs$dataset[crs$train, c(crs$input,
    crs$target)], method = "class", parms = list(split = "information"),
    control = rpart.control(usesurrogate = 0, maxsurrogate = 0))

Variables actually used in tree construction:
[1] Petal.Length

Root node error: 68/105 = 0.64761905

n= 105

          CP nsplit   rel error     xerror        xstd
1 0.48529412      0 1.000000000 1.16176471 0.065042415
2 0.45588235      1 0.514705882 0.72058824 0.075177606
```

訓練樣本 vs. 測試樣本

```
# 把資料檔隨機切割為訓練樣本及測試樣本
Splitdata = function(data,p=0.9)  {
    #p = 訓練樣本佔全部觀察值的比例
    index = sample(2, nrow(data), replace = TRUE,
           prob=c(p,1-p))
    train = data[index == 1,]
    test = data[index == 2,]
    out = list(train=train,test=test)
    return(out)
}
# 找出資料檔中，不是 Y 變數的所有其他變數
notY = function(data,Yname) {
    return(data[ ,-which(names(data) == Yname)])  }
```

訓練樣本 vs. 測試樣本

```
out = Splitdata(iris, 0.9) # 訓練樣本 佔90%, 測試樣本佔10%
Dtrain = out$train           # 訓練樣本
Dtest= out$test              # 測試樣本

Xtrain = notY(Dtrain,"Species")]
Ytrain = Dtrain$Species

Xtest = notY(Dtest,"Species")
Ytest = Dtest$Species
```

R 的集群分析 Packages/Functions

Algorithm	Package	Function
K-Means	base	kmeans
Hierarchical	base	hclust
Hierarchical	cluster	agnes
Fuzzy Clustering	cluster	fanny
決定最佳分群數目	fpc	pamk

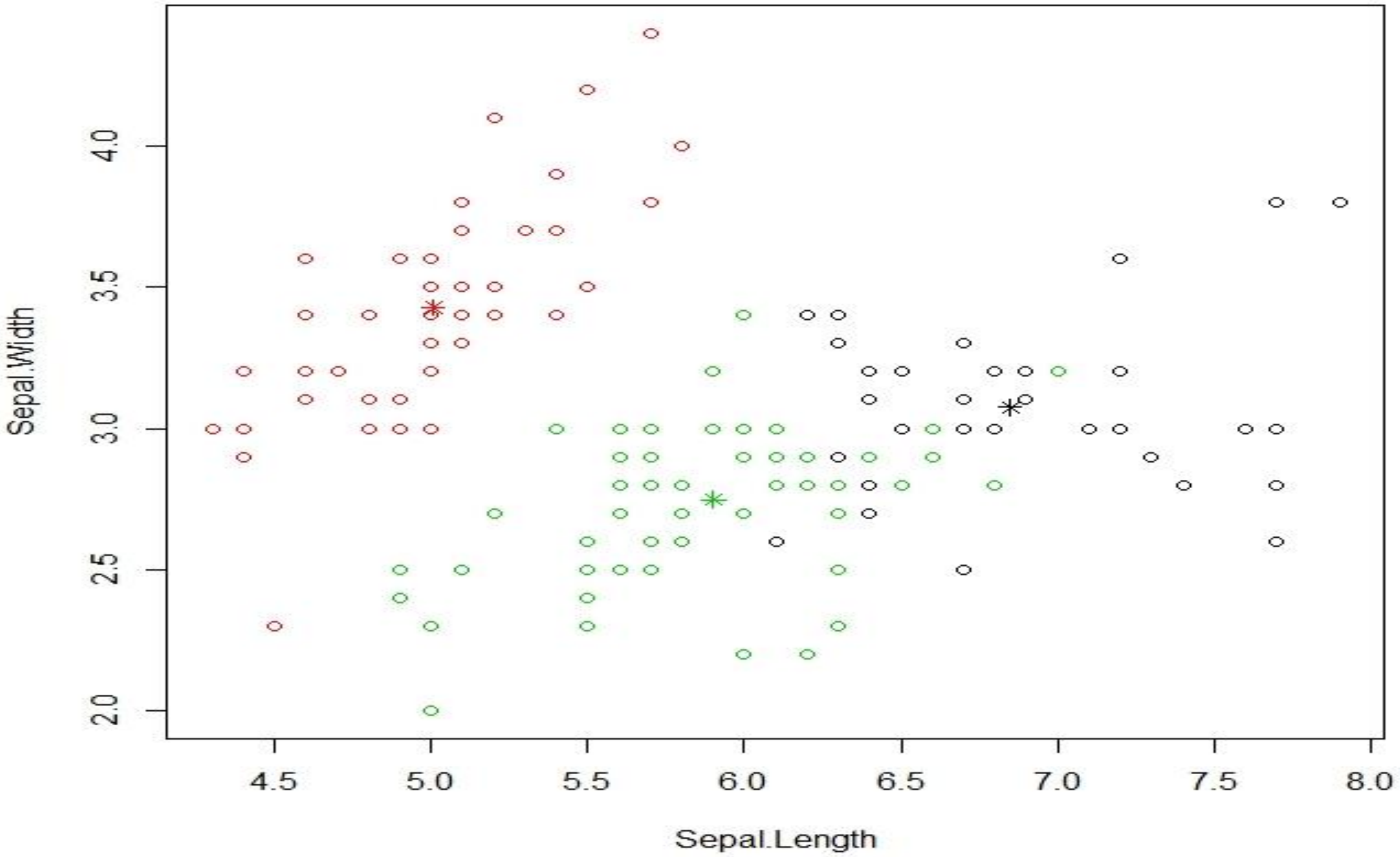
集群分析

```
# 抓取資料檔中的數值變數
NumVars = function(data) {
    nc = ncol(data)
    keep = numeric(nc)
    j = 0
    for (i in 1:nc)  {
        if (is.numeric(data[,i]))  {
            j = j + 1
            keep[j] = i
        }
    }
    return(as.matrix(data[,keep]))
}
```

K-Means 集群分析

```
# 基本安裝包含 kmeans 函數

iris2 = NumVars(iris)
ncluster = 3
result = kmeans(iris2,centers=ncluster,nstart=10)
result
result$cluster  # 顯示各觀察值的分群
table(result$cluster)   # 計算各群的數目
plot(iris2, col = result$cluster)
points(result$centers, col = 1:5, pch = 8)
```

K-Means 集群分析



決定最佳分群數目

```
library(fpc)
# 試探 2 ~ 6 群
pamk(B2,2:6)
```

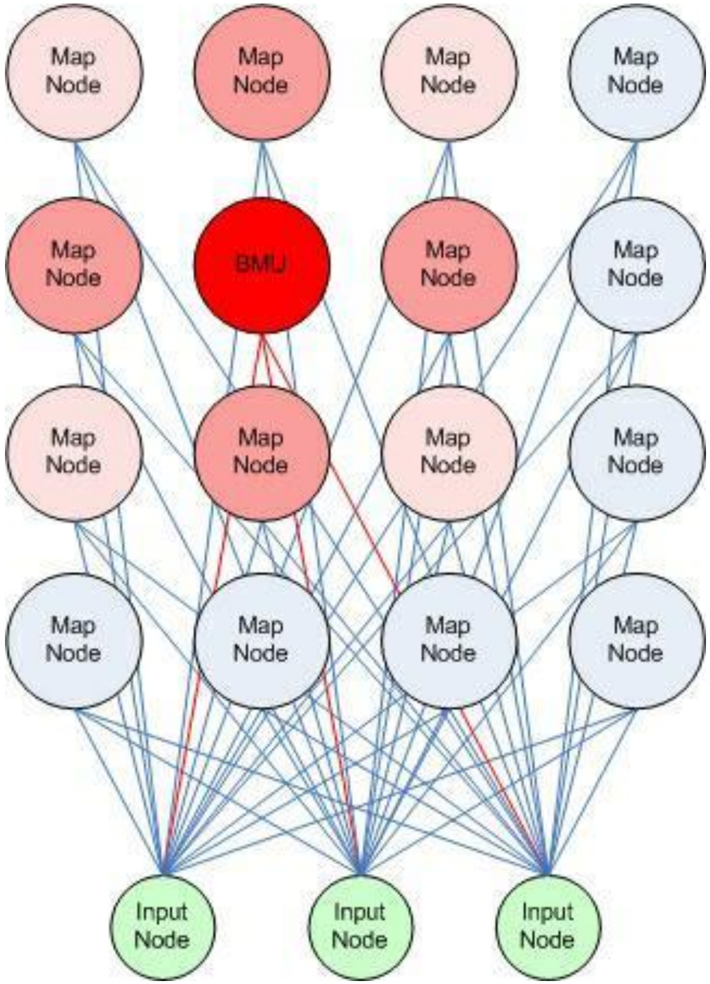
```
Output:

$pamobject
Medoids:
      ID tract       lon       lat medv cmedv     crim
     zn  indus    nox     rm   age
468 468   1101 -71.066 42.1780 19.1   19.1 4.42228
     0  18.1 0.584 6.003 94.5
223 223   3733 -71.125 42.2134 27.5   27.5 0.62356
     0    6.2 0.507 6.879 77.7
       dis  rad  tax ptratio      b lstat
468 2.5403   24  666      20.2 331.29 21.32
223 3.2721    8  307      17.4 390.39  9.93
```

```
Clustering vector:
  1    2    3    4    5    6    7    8    9   10   11   12
     13   14   15   16   17   18   19
  1    1    1    1    1    1    1    1    1    1    1    1
      1    1    1    1    1    1    1
 20   21   22   23   24   25   26   27   28   29   30   31
     32   33   34   35   36   37   38
  1    1    1    1    1    1    1    1    1    1    1    1
      1    1    1    1    1    1    1
........................
495  496  497  498  499  500  501  502  503  504  505  506
  1    1    1    1    1    1    1    1    1    1    1    1


 ..........................
```

```
$nc
    [1]  2
```

# 最佳 cluster 數目 ：2

SOM: Self-Organizing Map



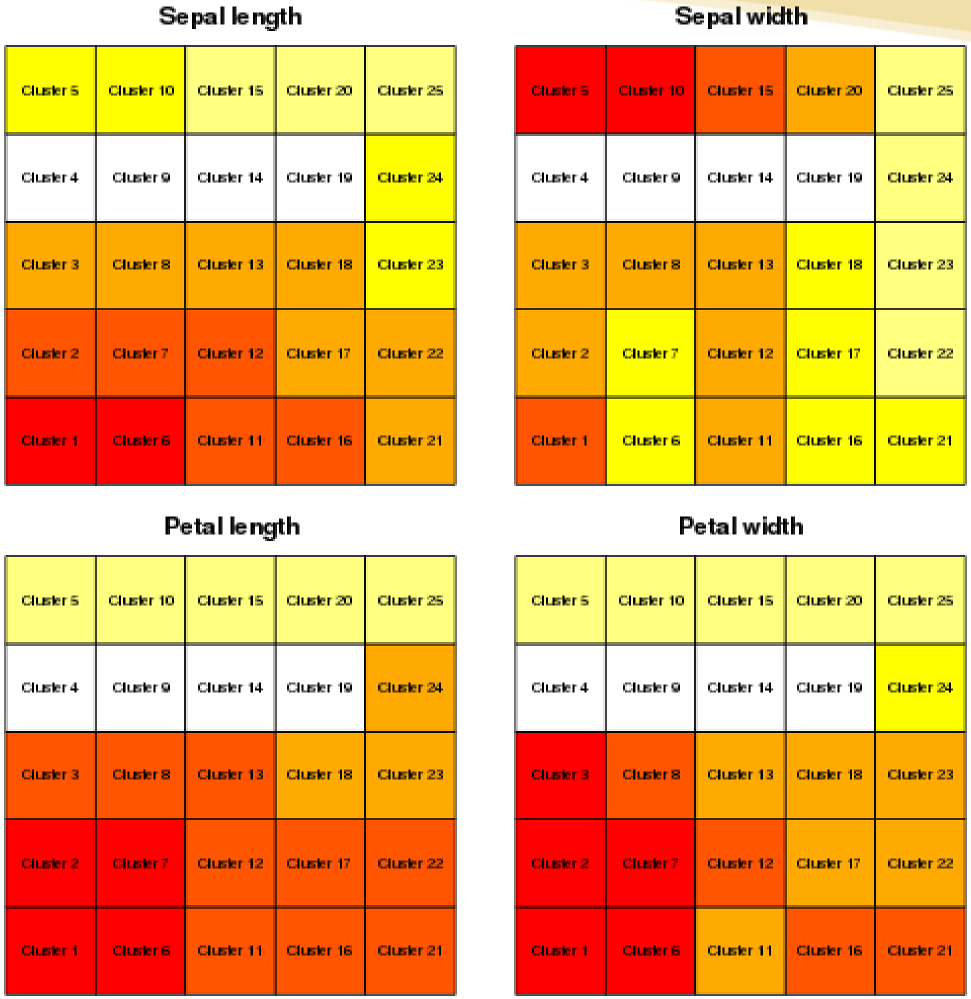
SOMbrero 套件：iris 資料檔

```
# 先安裝 slam, knitr, igraph 套件
install.packages("SOMbrero",
                       repos="http://R-Forge.R-project.org")
library(SOMbrero)
set.seed(4031730)
result = trainSOM(x.data = iris[, 1:4], verbose = TRUE,
                              nb.save = 5)
result$clustering
table(result$clustering)
summary(result)
```

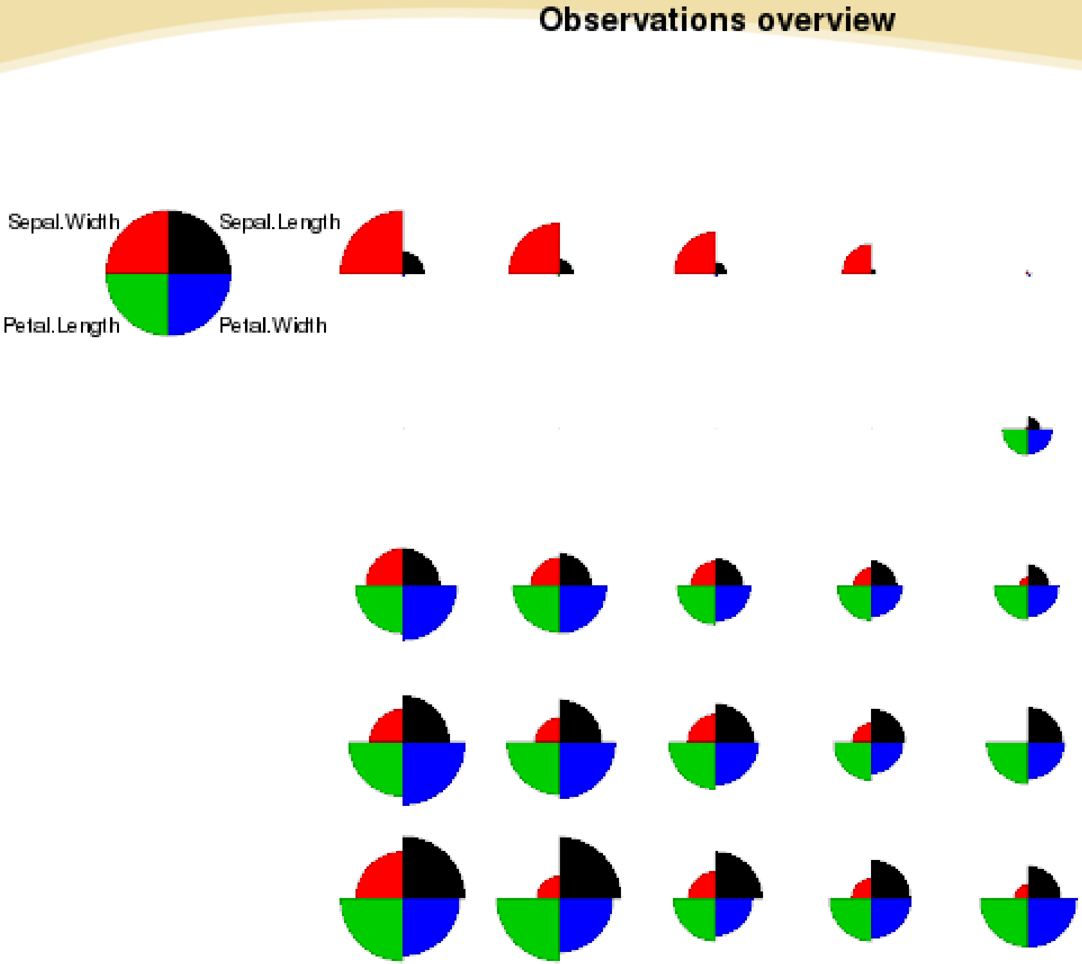
SOMbrero 範例

```
oldpar=par()  ;  par(mfrow = c(2, 2))
plot(result, what = "obs", type = "color", variable = 1, print.title = TRUE,
          main = "Sepal length")
plot(result, what = "obs", type = "color", variable = 2, print.title = TRUE,
          main = "Sepal width")
plot(result, what = "obs", type = "color", variable = 3, print.title = TRUE,
          main = "Petal length")
plot(result, what = "obs", type = "color", variable = 4, print.title = TRUE,
          main = "Petal width")
par(oldpar)
plot(result, what = "obs", type = "boxplot", print.title = TRUE)
plot(result, what = "obs", type = "names", print.title = TRUE)
predict(result, iris[1, 1:4])
result$clustering[1]
```

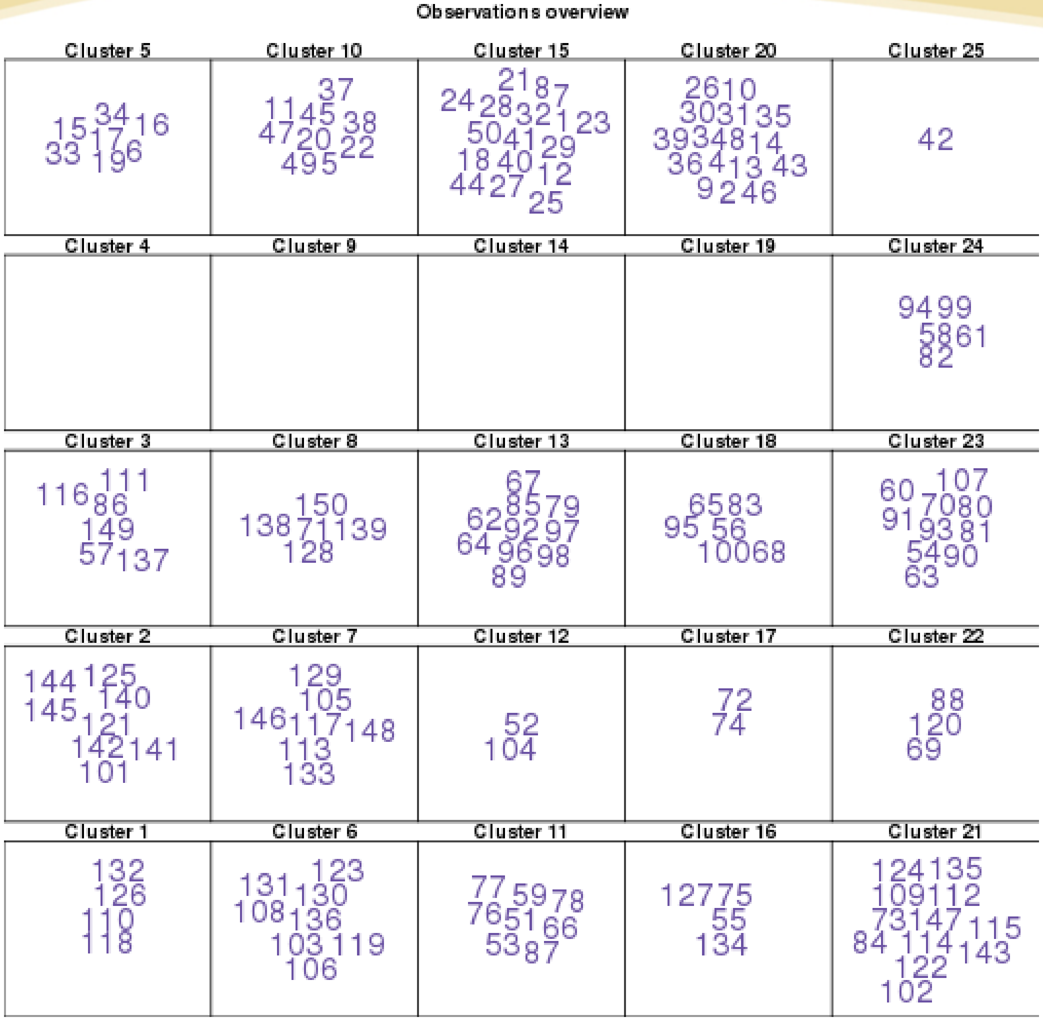
SOMbrero Plot (1)


Sepal length
Sepal width
Petal length
Petal width

SOMbrero Plot (2)



SOMbrero Plot (3)



分類：Confusion Matrix(混淆矩陣)

```
confmatrix = function(Y,Ypred) {
   t1 = table(Y,Ypredict=Ypred)
   print(t1)
   p = sum(diag(t1))/sum(t1)*100
   cat("\n\n預測正確率 = ",p,"% \n")
}

 # Example:
library(tree)
 result = tree(Species ~ . , data=iris)
 p1 = predict(result,type="class")
confmatrix(iris$Species, p1)
```

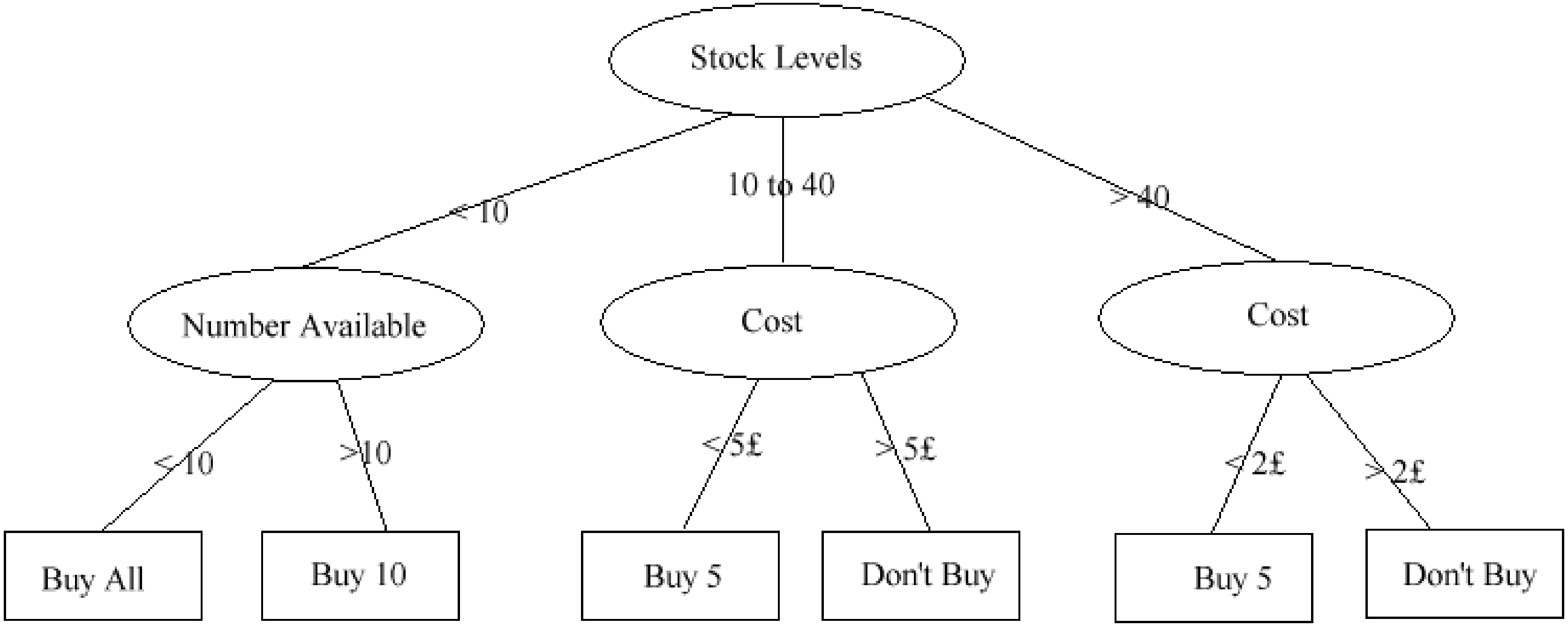
# 計算結果：

```
            Ypredict
Y            setosa versicolor virginica
  setosa         50          0         0
  versicolor      0         47         3
  virginica       0          1        49


預測正確率 =  97.33333 %
```

分類技術(Classification)

- Logistic 迴歸模式
- 判別分析(Discriminant Analysis)
- **分類樹(Classification Trees)**
- **隨機森林(Random Forest)**
- **類神經網路(Artificial Neural Network)**
- **支持向量機(SVM:Support Vector Machine)**
- 貝式分類器(Bayesian Classifier)

決策樹(Decision Trees)



決策樹建構流程

從上而下、從左而右，任意一個節點

Step 1. 找出此節點的最佳分割「變數」

Step 2. 找出此最佳變數的最佳分割點

Step 3. 產生左右兩個分支或更多分支

Step 4. 若滿足停止條件，則停止分支動作

Step 5. 必要時，刪剪決策樹以求最佳化

決策樹(Decision Trees)

依照功能區分：

1. 分類樹(Classification Tree): 目標變數為分類變數
2. 迴歸樹(Regression Tree)：目標變數為數值變數

常用的決策樹 Algorithm：

1. CART （分類/迴歸）
2. CHAID （分類）
3. QUEST(分類/迴歸）
4. C4.5 （分類/迴歸）
5. Random Forest （分類/迴歸）
6. Mob，Cubist （迴歸): Model based trees

四種常用的決策樹比較

特色	QUEST	CART	CHAID	C4.5
變數型態	連續/分類	連續/分類	分類	連續/分類
分支數目	2	2	2 以上	連續: 2 以上 分類: 2
分支變數	單/多變數	單/多變數	單變數	單變數
分割規則	卡方/F 檢定	Gain ratio	卡方檢定	Gain Ratio
可設定分類先驗機率	O	O	X	X
樹的修剪	測試樣本 或 交叉驗證	測試樣本 或 交叉驗證	Stopping Rules	同時分支與刪減
遺失值	內插法 或代理變數	代理變數	分出遺失值的支幹	使用機率加權

R 的決策樹函數

Algorithm	Package	Function
CART	rpart , tree	rpart , tree
CHAID	CHAID (in R-Forge site)	chaid
C4.5	RWeka	J48
C5.0	C50	C5.0
Random Forest	randomForest	randomForest
Model based Tree	party	mob
Model based Tree	Cubist	cubist

R 分類樹函數通用語法(1)

result = 函數名稱(Y ~ X_1+X_2+…+X_k , 其他選項)

# Y 若為分類變數，需為 Factor 型態

# "Y ~ . " 代表除 Y 之外的所有變數均為解釋變數

e.g result = tree(score ~ IQ + gender, data=students)

result = tree(score ~ . , data=students)

result 或 summary(result)   # 查看計算結果

names(result)   # 查看細項資訊

plot(result) 或 plot(result) ; text(result)

R 分類函數通用語法(2)

計算預測值

(1) 計算**訓練樣本**的預測分類 或 預測值

```
Ypred = predict(result , type="class")
Ypred = predict(result)
```

(2) 計算**測試樣本**的預測分類或預測值

```
Ypred = predict(result , new_data,
                                  type=“class”)
Ypred = predict(result , new_data)
```

R 分類函數通用語法(3)

分類樹: 混淆矩陣 (confusion matrix)

```
ctable = table(Y 變數名稱, Ypred)
```

分類樹：計算預測正確率

```
sum(diag(ctable))/sum(ctable)
```

迴歸樹: 計算 MAPE (Mean Absolute Percentage Error)

```
MAPE = function(Y, Ypred) mean(abs((Y - Ypred)/Y))
MAPE(Y, Ypred)
```

$$MAPE = \frac{1}{n}\sum\left|\frac{\hat{y}_t - y_t}{y_t}\right|$$

Example: CART Tree(1)

```
library(rpart)
result = rpart(Species ~ . ,data=iris); plot(result2); text(result2)
Ypred = predict(result,type="class")
confmatrix(iris$Species, Ypred)
```

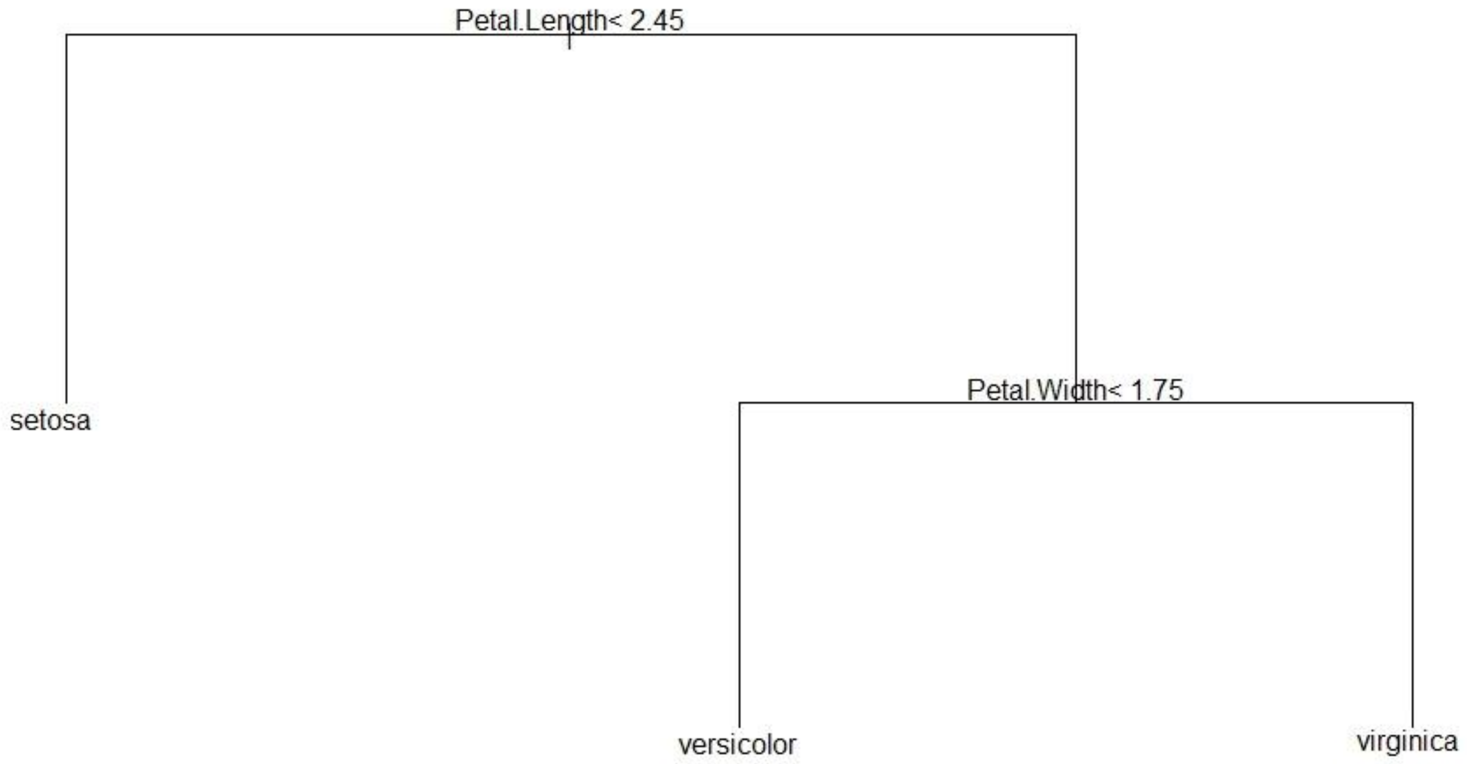
```
               Ypredict
Y               setosa versicolor virginica
  setosa            50          0         0
  versicolor         0         49         1
  virginica          0          5        45
```

預測正確率 = 96 %

```
result$variable.importance
```

```
 Petal.Width Petal.Length Sepal.Length  Sepal.Width
    88.96940     81.34496     54.09606     36.01309
```

Example: CART Tree(2)



Example:C4.5 (J48 in Rweka)

#J48: 需先安裝 partykit 與 Rweka 套件

```
library(RWeka)
result = J48(Species ~ . , data = iris)
result
summary(result)
plot(result)
Ypred = predict(result)
confmatrix(iris$Species, Ypred)
```

Example:C4.5 (continued)

```
J48 pruned tree
------------------

Petal.Width <= 0.6: setosa (50.0)
Petal.Width > 0.6
|   Petal.Width <= 1.7
|   |   Petal.Length <= 4.9: versicolor (48.0/1.0)
|   |   Petal.Length > 4.9
|   |   |   Petal.Width <= 1.5: virginica (3.0)
|   |   |   Petal.Width > 1.5: versicolor (3.0/1.0)
|   Petal.Width > 1.7: virginica (46.0/1.0)

Number of Leaves  :     5

Size of the tree :      9
```

Example:C4.5 (continued)

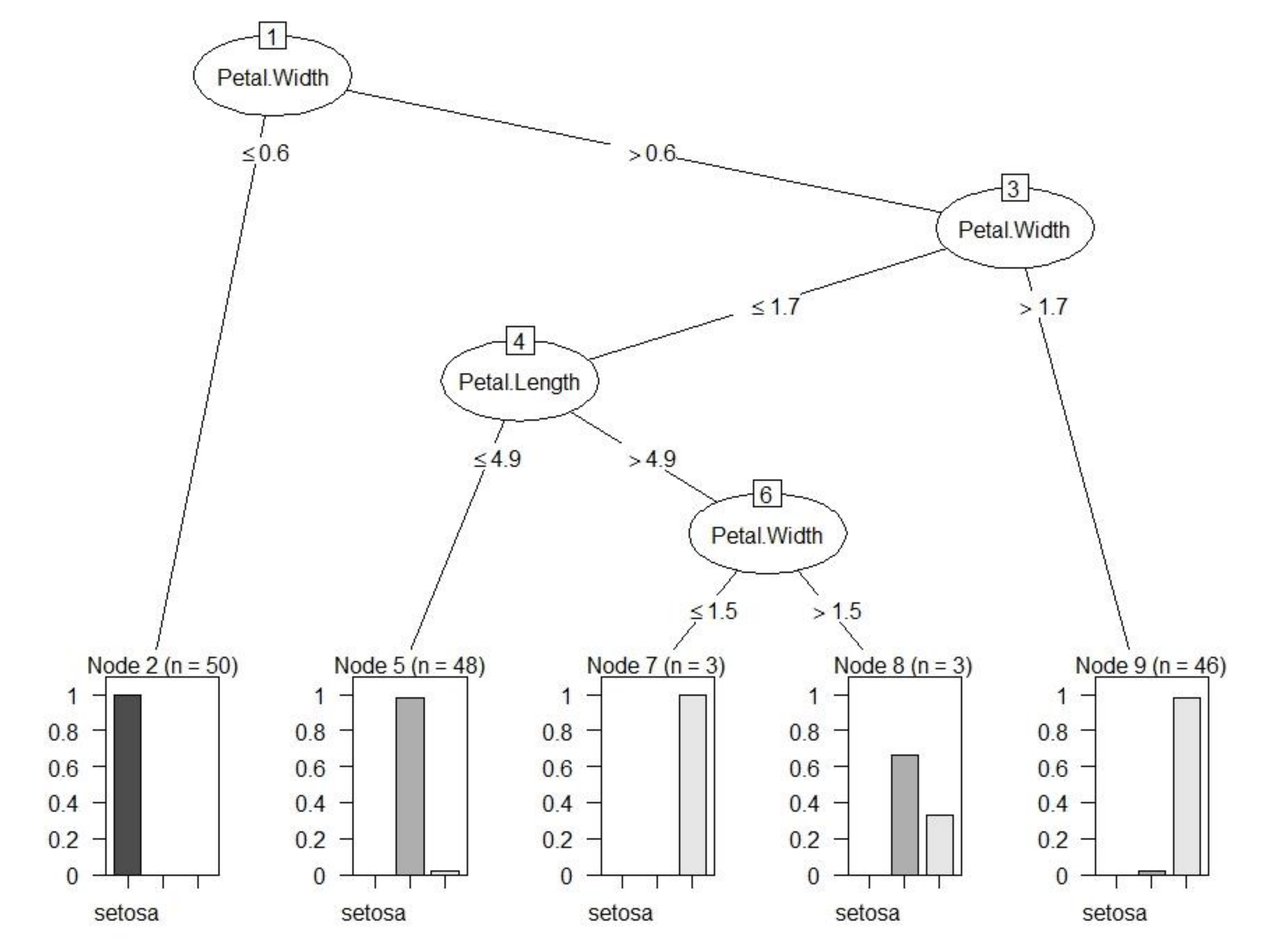
```
=== Summary ===

Correctly Classified Instances          147
    98      %
Incorrectly Classified Instances          3
    2      %
Kappa statistic
    0.97
Mean absolute error
    0.0233
Root mean squared error
    0.108
Relative absolute error
    5.2482 %
Root relative squared error
    22.9089 %
```

```
Coverage of cases (0.95 level)
    98.6667 %
Mean rel. region size (0.95 level)        34
    %
Total Number of Instances                150
```

```
=== Confusion Matrix ===

  a  b  c   <-- classified as
 50  0  0 |  a = setosa
  0 49  1 |  b = versicolor
  0  2 48 |  c = virginica
```

預測正確率：98%

Example:C4.5 (continued)



Example: Random Forest

```
library(randomForest)
set.seed(71)
result = randomForest(Species ~ . , data=iris ,
             importance=TRUE, proximity=TRUE)
print(result)
round(importance(result), 2)
names(result)
(   t = result$confusion   )
t = t[,1:3]
sum(diag(t))/sum(t)
```

Example: Random Forest (2)

```
Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 2

        OOB estimate of  error rate: 5.33%

Confusion matrix:
           setosa versicolor virginica class.error
setosa         50          0         0        0.00
versicolor      0         46         4        0.08
virginica       0          4        46        0.08
```

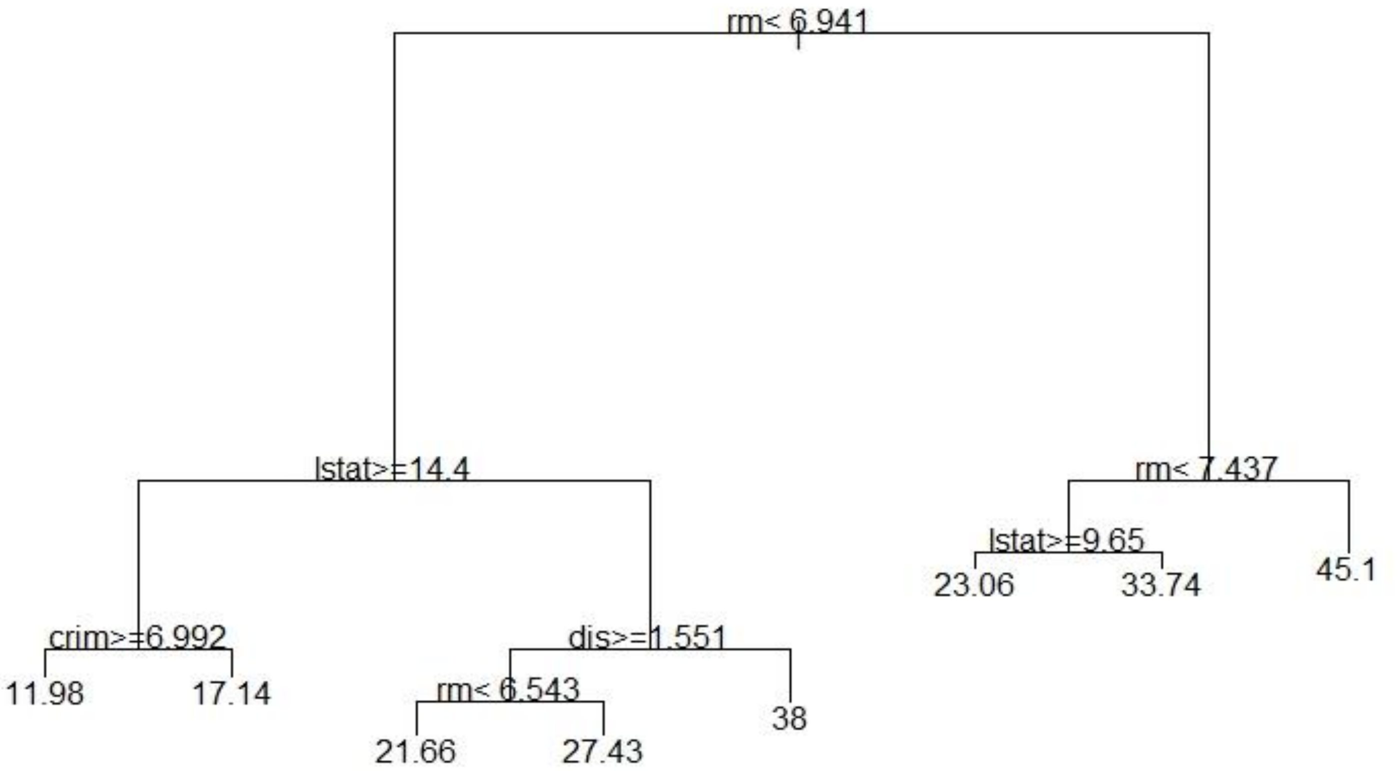
Example: Random Forest (3)

```
# 解釋變數相對重要性
             setosa versicolor virginica   MeanDecreaseAccuracy
Sepal.Length   6.04       7.85      7.93                  11.51
Sepal.Width    4.40       1.03      5.44                   5.40
Petal.Length  21.76      31.33     29.64                  32.94
Petal.Width   22.84      32.67     31.68                  34.50
             MeanDecreaseGini
Sepal.Length             8.77
Sepal.Width              2.19
Petal.Length            42.54
Petal.Width             45.77

#sum(diag(t))/sum(t)   # 預測正確率
[1] 0.9466667
```

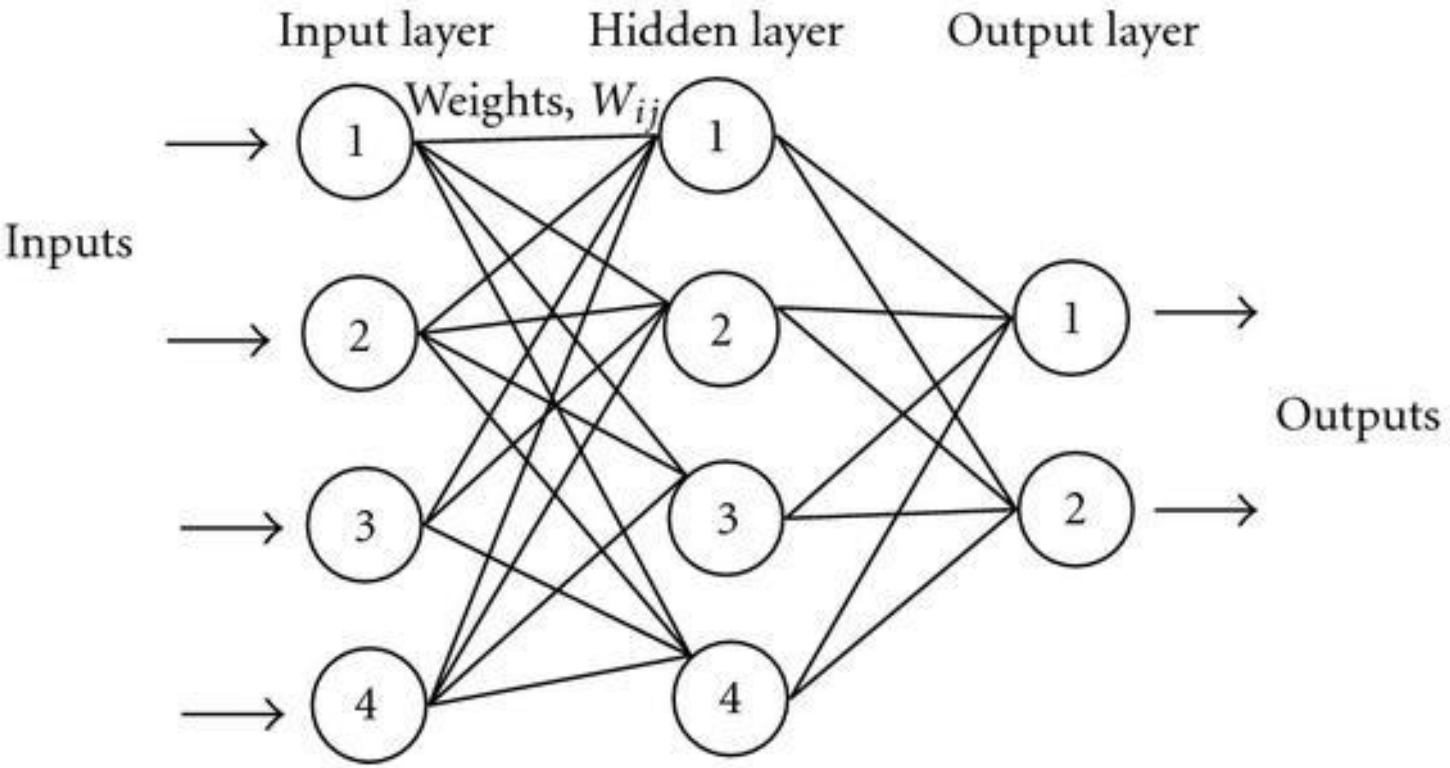
迴歸樹: BostonHousing 資料

```
# Boston Housing: 506 個納稅行政區資料,每一區有 14 個變數
# medv  : 該區自購房屋價格的中位數 (median). 單位: 1000 美元
# rm    : 該區房屋平均房間數目(rooms)
# dis   : 該區距離波士頓 5 個商業區的加權距離指標
# lstat : 該區低收入戶比例, crim : 該區犯罪率

library(mlbench)  ; data(BostonHousing)
library(rpart)
result = rpart(mdev ~ . ,data=BostonHousing)
result  ; summary(result)
plot(result);text(result)
Ypred = predict(result)
MAPE(BostonHousing$medv, Ypred)
   [1] 0.1545698
```

迴歸樹: BostonHousing 資料



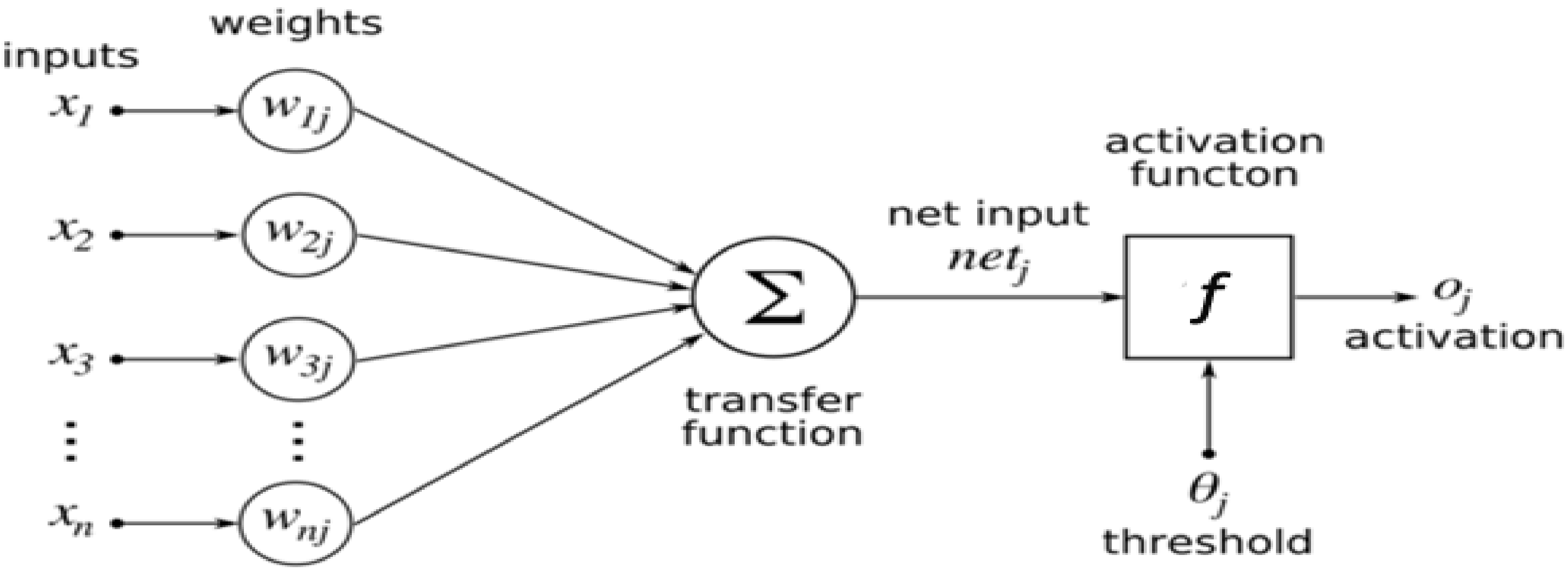
類神經網路(ANN)


Input layer
Hidden layer
Output layer
Weights, W_{ij}
Inputs
1
2
3
4
1
2
3
4
1
2
Outputs

類神經網路(ANN)

- Input: X_1, X_2, ..., X_k ，Output: Y
- 一個 ANN 具有輸入層、隱藏層、輸出層
- Y 的估計量 = $f(\sum w_i X_i - \theta)$

  其中 w_i 為各 X_i 的權重, θ 為閥值 (threshold), $f(\,.\,)$ 為某個非線性函數



R 的 ANN 套件

- **nnet**: 單一隱藏層 ANN
- **neuralnet**: 倒傳遞 ANN
- **RSNNS**: 包含 MLP, RBF, SOM, DLVQ 網路
- **pnn**: 機率類神經網路
- **popsom, som, SOMbrero**: SOM 網路

單一隱藏層網路：nnet

```
library(nnet)
result = nnet(Species ~ . , data = iris, size=3)
```

```
  a 4-3-3 network with 27 weights
```

```
Ypred = predict(result, type="class")
confmatrix(iris$Species, Ypred)
```

```
                    Ypredict
      Y          setosa versicolor virginica
   setosa          50         0         0
   versicolor       0        49         1
   virginica        0         1        49

  預測正確率 =  98.66667 %
```

倒傳遞網路：neuralnet

```
library(neuralnet)
y=as.numeric(iris$Species)
iris2=data.frame(iris[,1:4],y)
result = neuralnet(y~Sepal.Length+Sepal.Width+Petal.Length+Petal.Width,
                                    hidden=c(3,2),data=iris2)
result
cf=compute(result,iris2[,1:4])
Ypred = round(cf$net.result)
confmatrix(iris$Species, Ypred)
```

```
                Ypredict
Y                1  2  3
   setosa       50  0  0
   versicolor    0 49  1
   virginica     0  0 50

預測正確率 =  99.33333333 %
```

SVM: Support Vector Machine

```
library(e1071)
out = Splitdata(iris)  ;  iris.Train = out$train  ; iris.Test = out$test
result = svm(Species ~ . ,data=iris.Train)
print(result)
summary(result)
# 訓練樣本預測正確率
Ypred = predict(result, iris.Train)
confmatrix(iris.Train$Species,Ypred)
# 測試樣本預測正確率
Ypred = predict(result, iris.Test)
confmatrix(iris.Test$Species,Ypred)
```

SVM（continued）

```
# 訓練樣本
              Ypredict
Y            setosa versicolor virginica
  setosa         45          0         0
  versicolor      0         42         2
  virginica       0          2        45
預測正確率 =  97.05882353 %

# 測試樣本
             Ypredict
Y            setosa versicolor virginica
  setosa          5          0         0
  versicolor      0          6         0
  virginica       0          0         3
預測正確率 =  100 %
```

關聯規則分析

原理：根據 support 跟 confidence 挑選規則

資料格式：

	牛奶	餅乾	可樂	蛋	啤酒
t1	1	1	0	0	1
t2	1	0	1	1	1

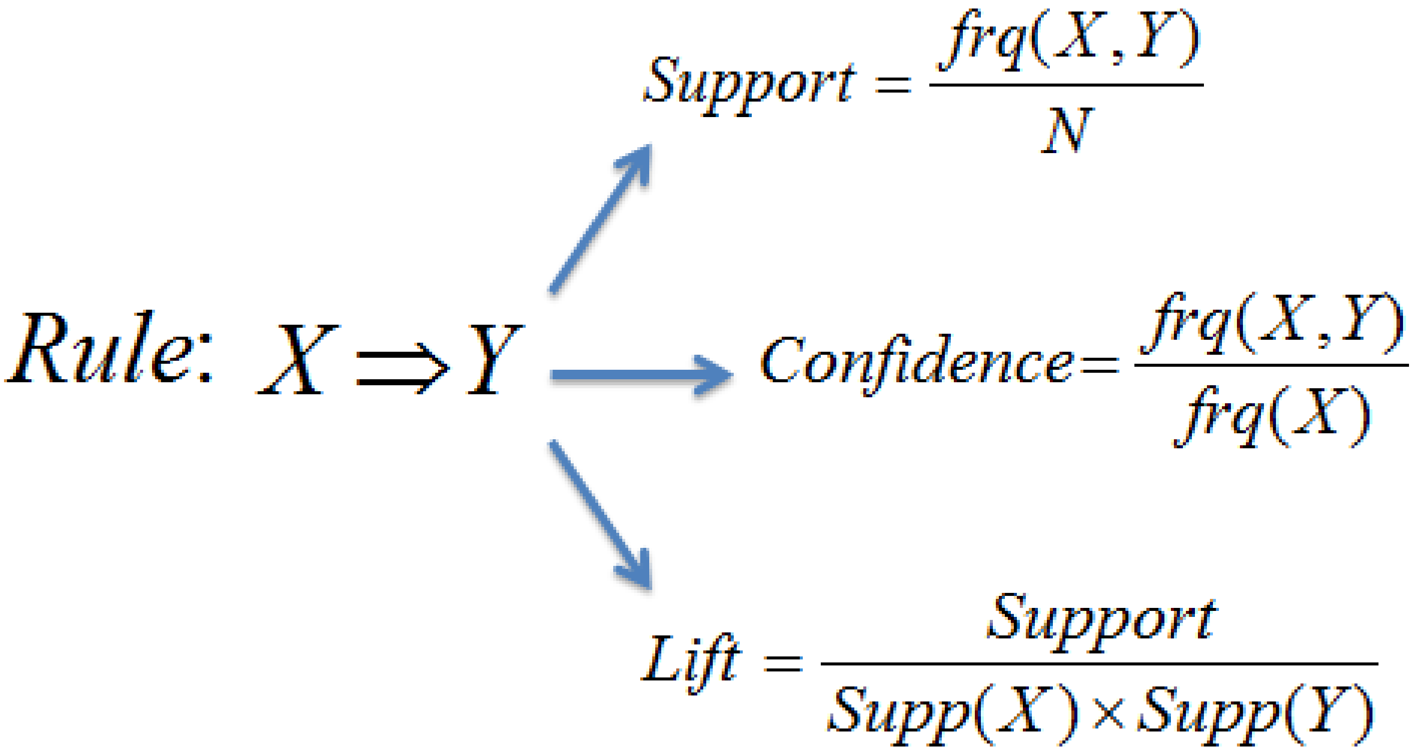
……………………………………………………………………………

- 規則：X -> Y （其中 X 與 Y 為物件的集合)

  例如：**{ 牛奶, 餅乾 } -> { 啤酒 }**

- **support** = X 與 Y 同時出現的次數 / 所有交易數
- **confidence** = X 與 Y 同時出現的次數 / X 出現次數
- **lift** = 實際 Confidence/預期的Confidence

Support,Confidence,Lift



$$Rule:\ X \Rightarrow Y$$

$$Support = \frac{frq(X,Y)}{N}$$

$$Confidence = \frac{frq(X,Y)}{frq(X)}$$

$$Lift = \frac{Support}{Supp(X) \times Supp(Y)}$$

關聯分析： arules 套件

```
library(arules)
# 使用iris資料檔
# 所有數值變數需先轉成分類變數(factor或ordered factor)
SepL = ordered(cut(iris$Sepal.Length,breaks = 4))
SepW = ordered(cut(iris$Sepal.Width, breaks = 4))
PetL = ordered(cut(iris$Petal.Length,breaks = 4))
PetW = ordered(cut(iris$Petal.Width, breaks = 4))

iris2=data.frame(SepL,SepW,PetL,PetW,Species=iris$Species)
iris3 = as(iris2, "transactions")

rules = apriori(iris3,parameter = list(supp = 0.2, conf = 0.6, target = "rules"))
summary(rules)
inspect(head(sort(rules, by = "support"), n = 100))
```

arules: iris 資料檔 Output

```
lhs                              rhs                       support    confidence     lift
1  {Species=setosa}         => {PetW=(0.0976,0.7]}  0.3333333333 1.0000000000 3.000000000

2  {PetW=(0.0976,0.7]}  => {Species=setosa}         0.3333333333 1.0000000000 3.000000000

3  {Species=setosa}         => {PetL=(0.994,2.48]}  0.3333333333 1.0000000000 3.000000000

4  {PetL=(0.994,2.48]}  => {Species=setosa}         0.3333333333 1.0000000000 3.000000000

5  {PetW=(0.0976,0.7]}  => {PetL=(0.994,2.48]}  0.3333333333 1.0000000000 3.000000000

6  {PetL=(0.994,2.48]}  => {PetW=(0.0976,0.7]}  0.3333333333 1.0000000000 3.000000000

 。 。 。 。

# 若 Lift 值超過 1.0，則 X 與 Y 關聯性越強
```

Big Data？Hadoop?

Don't use Hadoop - your data isn't that big

作者 Chris Stucchio 認為：

「5 TB 以上就需要 Hadoop.」

訪客 Rama Ramasamy 回應：

「Considering I've managed hadoop from 0 to 40+ PB in a single cluster , my magic number for hadoop and datasize would be 20 TB」

Big Data vs.我是誰？

「Google Ads 認為我喜歡政治、亞洲實務、香水、名人八卦、動漫电影、犯罪新聞，但是對書籍、文學、社會人文沒有興趣」

「Yahoo 認為我喜歡曲棍球(hockey)、Rap、搖滾樂、食譜、親子關係、服裝、化妝品，使用 Mac 電腦，住在紐約」

「BlueKai 認為我是 18 ~ 19 歲的年輕女性，常租跑車代步」

2011/03/10, 時代雜誌(Time)主編 Joel Stein

-- Data Mining: How Companies Now Know Everything About You

Big Data 的迷思

有了 Big Data，資料探勘跟統計都落伍了？

Comment:

這是典型的張飛打岳飛的半吊子論述。

Big Data 的概念只牽涉到資料量的大小與儲存。但是當我們要分析 Big Data 時，仍然需要用到傳統的統計技術與資料探勘技術、或是從這些技術改寫的新 algorithms。

統計 vs. Big Data

Q：Big Data之所以能被迅速紀錄，是否可能因為資料與變數**並非**那麼直接與重要？

Example: 哪一類型消費者可能會買白色手機？

(1)**間接資訊**：網站紀錄數十萬筆瀏覽資料。

e.g. 點選至白色手機網頁的瀏覽紀錄

(2)**直接資訊**：隨機抽選 1000 人所得的問卷

Big Data 分析的考量

1. 大量資料的儲存/資料倉儲

   特殊格式資料檔(e.g. SAS)？資料庫軟體？In-Database 計算？

2. 資料是否能完全載入電腦記憶體中？

3. 分析方法是否能順利完成計算？

4. 如果能完成計算，計算速度快不快？

Big Data 分析類別

1. **資料處理**：排序、合併、比對串聯等
2. **彙整統計**：平均數、counting、比例
3. **一般統計計算**：迴歸、ANOVA等
4. **進階分析**：資料探勘、存活分析等

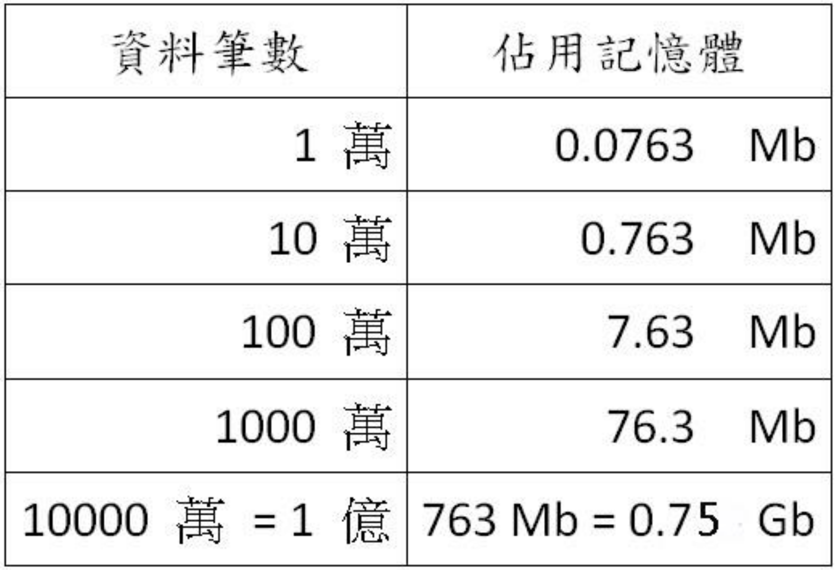
分析技術 vs. Big Data

分析計算法則(Algorithm)的考量：

1. 不需改寫：電腦記憶體足夠
2. 部分改寫：e.g. 循序切割(Chunking)彙整
3. 大幅度改寫：e.g. MLE 估計量
4. 大幅度改寫 + 多核心運算/平行化處理
5. 重新定義：e.g. Streaming Data

R 軟體資料大小 vs. 電腦記憶體

R 軟體物件的記憶體大小：1個變數



資料筆數	佔用記憶體
1 萬	0.0763 Mb
10 萬	0.763 Mb
100 萬	7.63 Mb
1000 萬	76.3 Mb
10000 萬 = 1 億	763 Mb = 0.75 Gb

1000萬筆資料、100 變數

=> 0.075x100 = 7.5 Gb

=> 加上計算需求 7.5 Gb x 3 = 22.5 Gb < 32 Gb

R and Big Data

- 8 Gb 記憶體：應可順利處理 100萬筆資料
- 16 Gb 記憶體：應可順利處理 500萬筆資料
- 32 Gb 記憶體：應可順利處理 1000萬筆資料
- 64 Gb 記憶體：應可順利處理 2000萬筆資料
- 1億筆以上或TB/PB等級：搭配特殊 R套件 +/- Database

- 大量資料已經近似於母體，不需要作統計推論(點估計、檢定、信賴區間）=> 隨機樣本才需要作統計推論

R 的 Big Data 套件

- R + 各類商業/免費資料庫軟體
- bigmemory
- ff
- DatABEL
- pdbBASE (多機平行化處理)
- RMOA = R + MOA (Massive Online Analysis)
- Resolution R Enterprise: xdf 檔案格式
- Oracle R Enterprise: in-database 運算
- R + Hadoop +/- Mahout