數據分析師假日精修班
Lab6

David Chiu
2016/07/31

文字探勘步驟

文字處理

- 斷詞
- 斷句

資料量化

- 詞頻計算
- 文字矩陣
- 計算TF-IDF

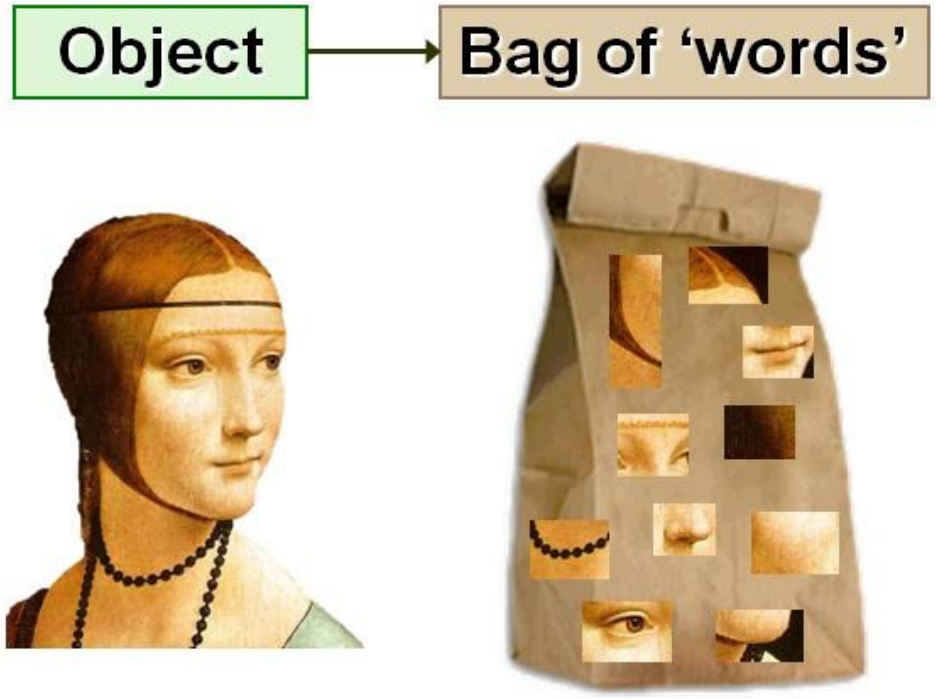
探勘分析

- 文字雲
- 文章分群
- 文章分類
- 關聯分析

詞頻矩陣

詞袋模型 (Bag of Words)

- 將文章斷詞以後，可以用向量表示文字。這種表示方式如同將文字變成在袋子中零散且獨立的物件。



詞袋範例

```
s = "大巨蛋案對市府同仁下封口令？柯P否認"
mixseg = worker()
segment(code= s , jiebar = mixseg)
```

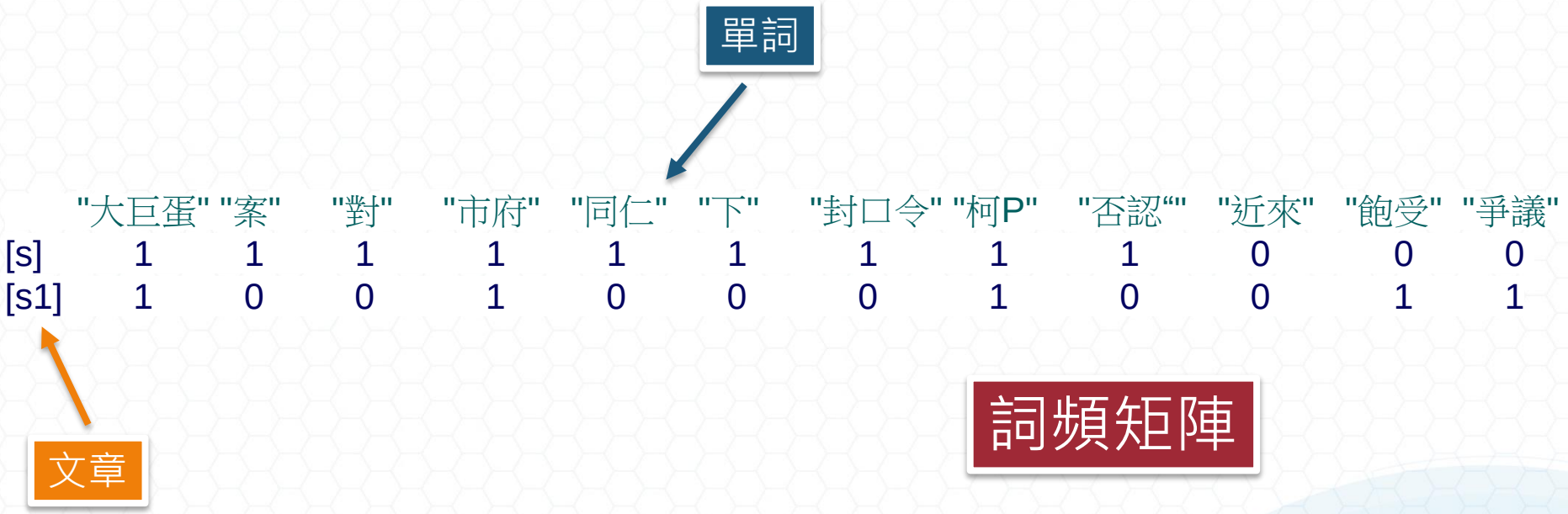
```
[1] "大巨蛋" "案"     "對"     "市府"   "同仁"   "下"     "封口令" "柯P"    "否認"
```

	"大巨蛋"	"案"	"對"	"市府"	"同仁"	"下"	"封口令"	"柯P"	"否認"
[1]	1	1	1	1	1	1	1	1	1

詞頻矩陣 (document-term matrix)

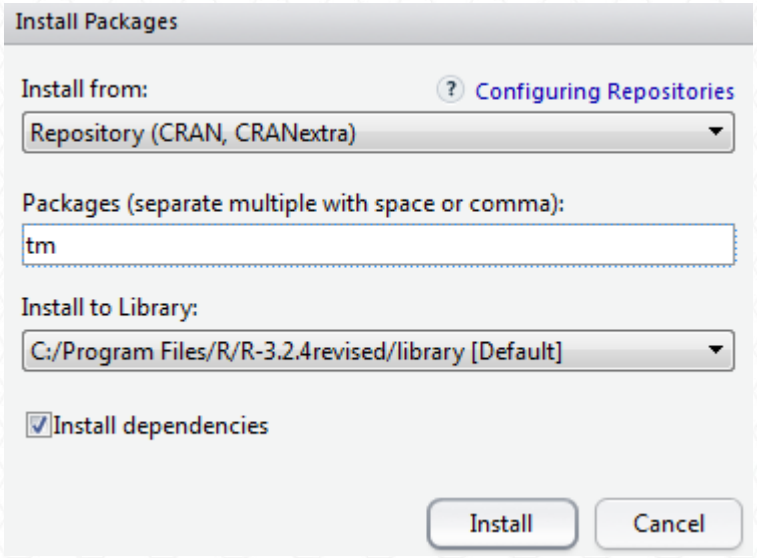
s = "大巨蛋案對市府同仁下封口令？柯P否認"

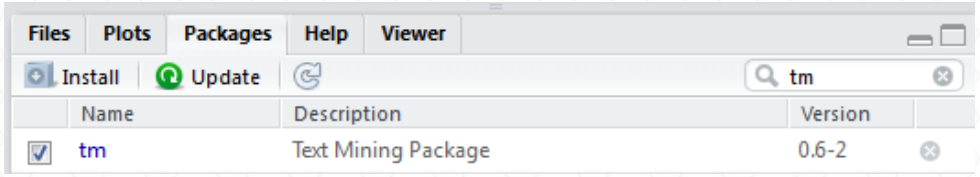
s1 = "柯P市府近來飽受大巨蛋爭議"



使用 tm 套件建立詞頻矩陣

```
install.packages("tm")
library(tm)
```





建立英文詞頻矩陣

建立詞頻向量

```
e3 = 'Hello, I am David. I have taken over 100 courses ~~~'
e3.vec = strsplit(e3, ' ')[[1]]
e3.corpus = Corpus(VectorSource(list(e3.vec)))
e3.dtm = DocumentTermMatrix(e3.corpus)
inspect(e3.dtm)
```

```
     Terms
Docs ~~~ 100 courses david. have hello, over taken
   1   1   1       1      1    1      1    1     1
```

於control 處可以設定蒐集字詞長度

```
dtm = DocumentTermMatrix(e3.corpus,
control=list(wordLengths=c(1, 20)))
inspect(dtm)
```

```
    Terms
Docs ~~~ 100 am courses david. have hello, i over taken
   1   1   1  1       1      1    1      1 2    1     1
```

使用Transformer 可以做字詞轉換

```
> getTransformations()
[1] "removeNumbers"     "removePunctuation" "removeWords"
[4] "stemDocument"      "stripWhitespace"
```

移除數字、標點符號

```
doc = tm_map(e3.corpus, removeNumbers)
doc = tm_map(doc, removePunctuation)
dtm = DocumentTermMatrix(doc)
inspect(dtm)
```

```
    Terms
Docs courses david have hello over taken
   1       1     1    1     1    1     1
```

或可以自製Transformer

```
removetilde <- content_transformer(function(x, pattern) {return (gsub("~", "", x))})
doc = tm_map(e3.corpus, removetilde)
dtm = DocumentTermMatrix(doc)
inspect(dtm)
```

移除 ~

```
     Terms
Docs 100 courses david. have hello, over taken
   1   1       1      1    1      1    1     1
```

建立詞頻矩陣

```
e1 = 'this is a book'
e2 = 'this is my car'
e1.vec = strsplit(e1, ' ')[[1]]
e2.vec = strsplit(e2, ' ')[[1]]
e.vec = list(e1.vec, e2.vec)
e.corpus = Corpus(VectorSource(e.vec))
e.dtm = DocumentTermMatrix(e.corpus)
```

```
      Terms
Docs book car this
   1    1   0    1
   2    0   1    1
```

建立中文詞頻矩陣

建立中文的詞頻矩陣

```
library(jiebaR)
mixseg = worker()
s = "大巨蛋案對市府同仁下封口令？柯P否認"
s1 = "柯P市府近來飽受大巨蛋爭議"

s.vec <- segment(code= s , jiebar = mixseg)
s1.vec <- segment(code= s1 , jiebar = mixseg)
s.corpus = Corpus(VectorSource(list(s.vec, s1.vec)))

s.dtm <- DocumentTermMatrix(s.corpus)
inspect(s.dtm)
```

```
    Terms
Docs 下\n封口令\n柯p 大巨蛋\n爭議 大巨蛋\n案\n對\n市府
   1                1             0
   2                0             1
```

產生錯誤詞頻

產生正確中文詞頻矩陣

```
source('https://raw.githubusercontent.com/ywchiu/rtibame/master/Lib/CNCorpus.R')
library(jiebaR)
mixseg = worker()
s = "大巨蛋案對市府同仁下封口令？柯P否認"
s1 = "柯P市府近來飽受大巨蛋爭議"
s.vec <- segment(code= s , jiebar = mixseg)
s1.vec <- segment(code= s1 , jiebar = mixseg)
s.corpus = CNCorpus(list(s.vec, s1.vec))
control.list=list(wordLengths=c(1,Inf),tokenize=space_tokenizer)
s.dtm <- DocumentTermMatrix(s.corpus, control=control.list)
inspect(s.dtm)
```

```
        Terms
Docs 下 大巨蛋 市府 同仁 否認 爭議 近來 封口令 柯p 案 飽受 對
   1 1      1    1    1    1    0    0      1   1 1    0  1
   2 0      1    1    0    0    1    1      0   1 0    1  0
```

詞頻矩陣的應用

產生1,500篇文章詞頻矩陣

```
library(jiebaR)
mixseg = worker()
apple.seg =lapply(applenews$content,
function(e)segment(code=e, jiebar=mixseg))

s.corpus <- CNCorpus(apple.seg)
control.list=list(wordLengths=c(2,Inf),tokenize=space_tok
enizer)
s.dtm <- DocumentTermMatrix(s.corpus,
control=control.list)
dim(s.dtm )
```

詞頻矩陣操作

- 尋找詞頻介於200~ 300 的詞

```
findFreqTerms(dtm, 200,300)
```

- 尋找與”大巨蛋”相關係數大於0.7的詞

```
findAssocs(dtm, "大巨蛋", 0.7)
$大巨蛋
遠雄 解約 市府 展延
0.88 0.78 0.74 0.72
```

刪除稀疏條目

```
dim(dtm)
[1]  1500 41855

dtm.remove = removeSparseTerms(dtm, 0.9)
dim(dtm.remove)
[1] 1500   66

dtm.remove$dimnames$Terms
```

詞頻矩陣的進階應用

- 文章相似度計算，產生推薦文章
- 文章分群，群聚類似主題
- 文章自動分類
- 建立主題模型

機器學習

機器學習

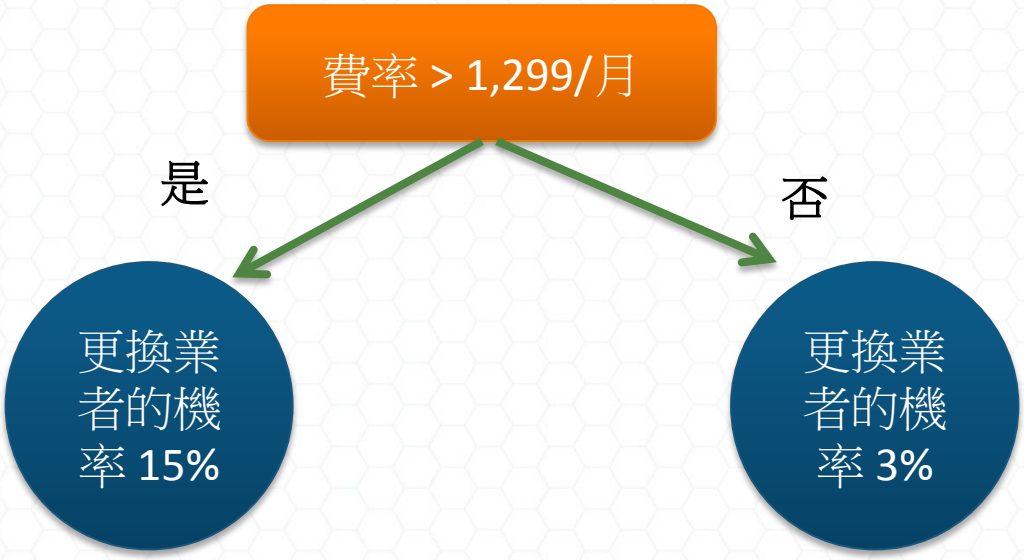
- 機器學習的目的是：歸納（Induction）
  - 從詳細事實到一般通論

A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E

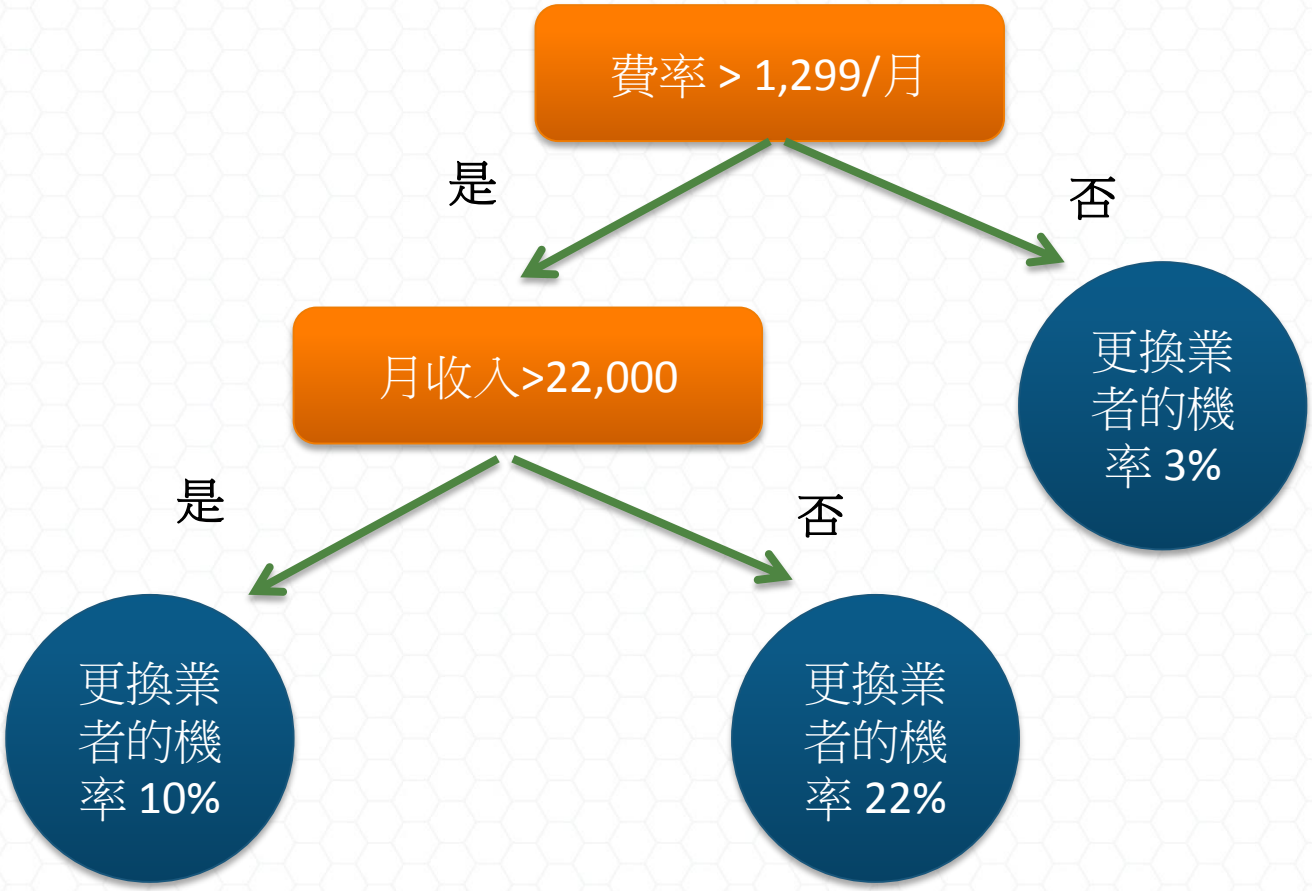
-- Tom Mitchell (1998)

- 找出有效的預測模型
  - 一開始都從一個簡單的模型開始
  - 藉由不斷餵入訓練資料，修改模型
  - 不斷提升預測績效

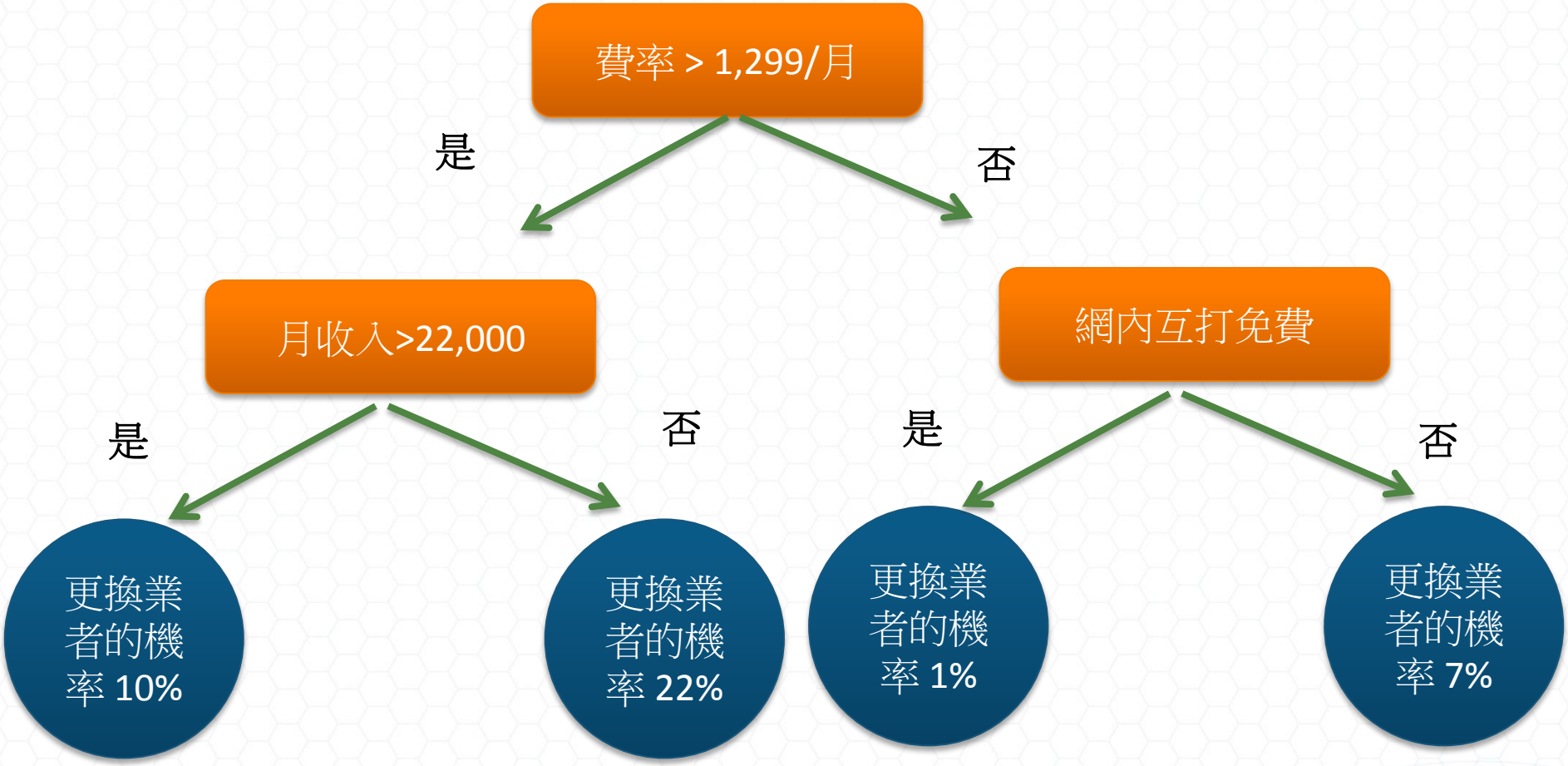
簡單的分類問題(決策樹)



簡單的分類問題(決策樹)



簡單的分類問題(決策樹)

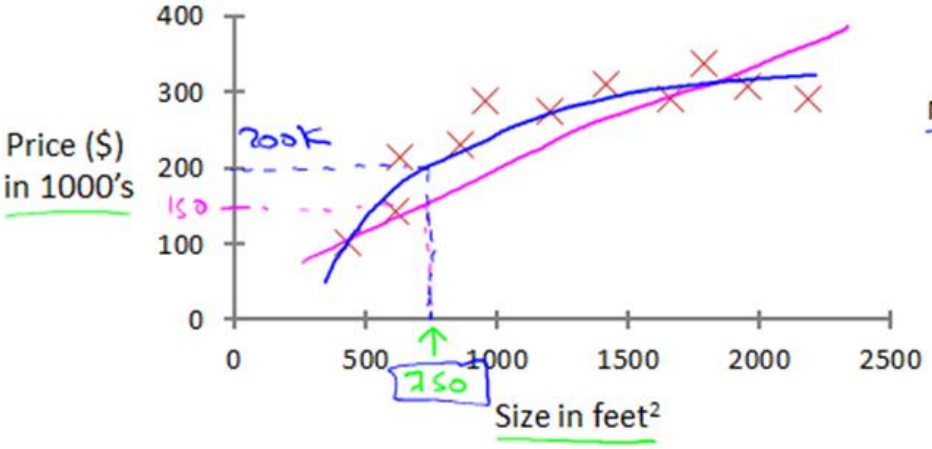

費率 > 1,299/月
是
否
月收入>22,000
網內互打免費
是
否
是
否
更換業者的機率 10%
更換業者的機率 22%
更換業者的機率 1%
更換業者的機率 7%

機器學習問題分類

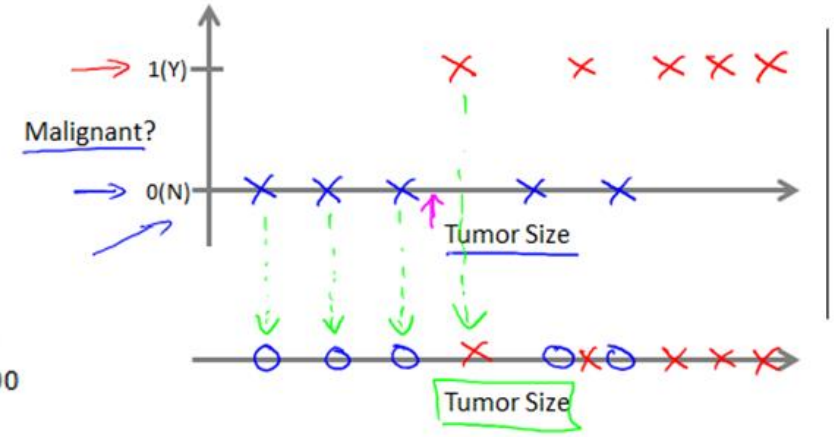
- 監督式學習 (Supervised Learning)
  - 迴歸分析 (Regression)
  - 分類問題 (Classification)
- 非監督式學習　(Unsupervised Learning)
  - 降低維度 (Dimension Reduction)
  - 分群問題 (Clustering)

監督式學習

Housing price prediction.



Breast cancer (malignant, benign)



監督式學習

- 分類問題
  - 根據已知標籤的訓練資料集(Training Set)，產生一個新模型，用以預測測試資料集(Testing Set)的標籤。
  - e.g. 股市漲跌預測
- 迴歸分析
  - 使用一組**已知對應值**的數據產生的模型，預測新數據的對應值
  - e.g. 股價預測

非監督式學習

- 降低維度
  - 產生一有最大變異數的欄位線性組合，可用來降低原本問題的維度與複雜度
  - e.g. 濃縮用到的特徵，編纂成一個新指標
- 分群問題
  - 物以類聚 (近朱者赤、近墨者黑)
  - e.g.將客戶分層

分群方法簡介

分群應用

- 市場分析
  - 將客戶依行為跟特徵做不同區隔
  - 產品定位
  - 區分市場
- 產品搭配銷售
  - 將同類型的產品組合成紅綠標組合
- 社會網路分析
  - 找出相似的朋友群
- 搜尋結果分組
  - 找出類似文章或主題

分群問題

- 特色
  - 沒有正確答案 (標籤)
  - 依靠自身屬性相似度，物以類聚

- 如何判斷相似度
  - 以『距離』作為分類的依據，『相對距離』愈近的，『相似程度』愈高，歸類成同一群組。

各種距離公式

- **歐氏距離**
  - 二維平面上兩點直線距離

$$d_{12} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

- **曼哈頓距離**
  - 城市街區距離(City Block distance)

$$d_{12} = |x_1 - x_2| + |y_1 - y_2|$$

各種距離公式 (續)

- **切比雪夫距離 (Chebyshev Distance)**
  - 象棋中國王走一步能夠移動到相鄰的8個方格中的任意一個。那麼國王從格子(x1,y1)走到格子(x2,y2)最少需的步數

$$d_{12} = \max(|x_1 - x_2|, |y_1 - y_2|)$$

- **閔可夫斯基距離(Minkowski Distance)**
  - 閔氏距離不是一種距離，而是一組距離的定義。
  - 其中p是一個變參數。
  - 當p=1時，就是曼哈頓距離
  - 當p=2時，就是歐氏距離
  - 當p→∞時，就是切比雪夫距離

$$d_{12} = \sqrt[p]{\sum_{k=1}^{n} |x_{1k} - x_{2k}|^p}$$

使用R 計算距離

```
?dist
x = c(0, 0, 1, 1, 1, 1)
y = c(1, 0, 1, 1, 0, 1)
```

■ 歐氏距離

```
dist(rbind(x,y), method =  "euclidean")
dist(rbind(x,y), method ="minkowski", p=2)
```

■ 曼哈頓距離

```
dist(rbind(x,y), method =  "manhattan")
dist(rbind(x,y), method ="minkowski", p=1)
```

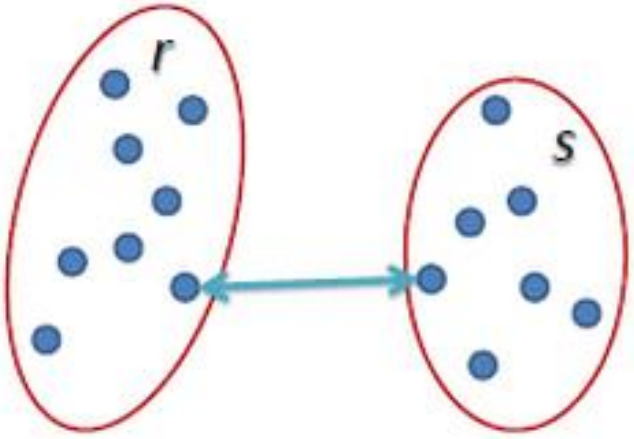
點間距離與群間距離

- 點間距離能衡量兩點間的距離
  - 相近的點可被視為類似樣本點
- 但如何去計算群與群之間的相似度
  - 單一連結聚合演算法
  - 完整連結聚合演算法
  - 完整連結聚合演算法
  - 沃德法

單一連結聚合演算法

- 單一連結聚合演算法（single-linkage）：群聚與群聚間的距離可以定義為不同群聚中最接近兩點間的距離

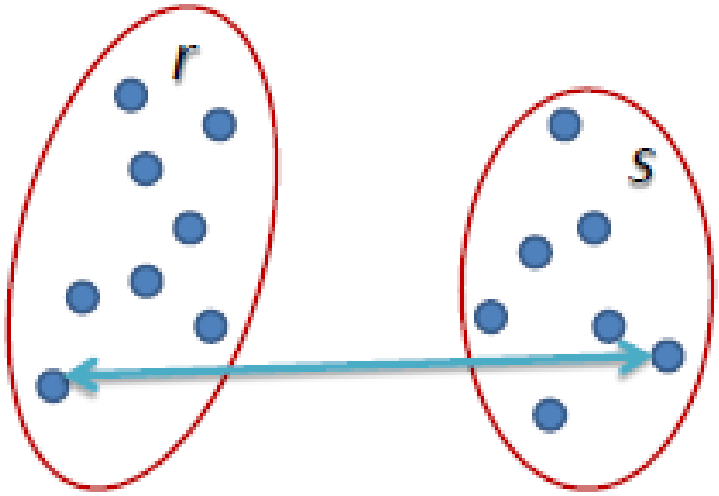
$$d(C_i, C_j) = \min_{\mathbf{a} \in C_i, \mathbf{b} \in C_j} d(\mathbf{a}, \mathbf{b})$$



完整連結聚合演算法

- 完整連結聚合演算法（complete-linkage）：群聚間的距離定義為不同群聚中最遠兩點間的距離

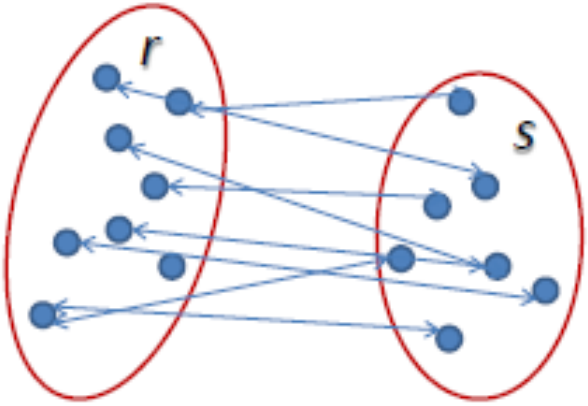
$$d(C_i, C_j) = \max_{\mathbf{a} \in C_i, \mathbf{b} \in C_j} d(\mathbf{a}, \mathbf{b})$$



平均連結聚合演算法

- 平均連結聚合演算法（average-linkage）：群聚間的距離則定義為不同群聚間各點與各點間距離總和的平均

$$d(C_i, C_j) = \sum_{\mathbf{a} \in C_i, \mathbf{b} \in C_j} \frac{d(\mathbf{a}, \mathbf{b})}{|C_i||C_j|},$$



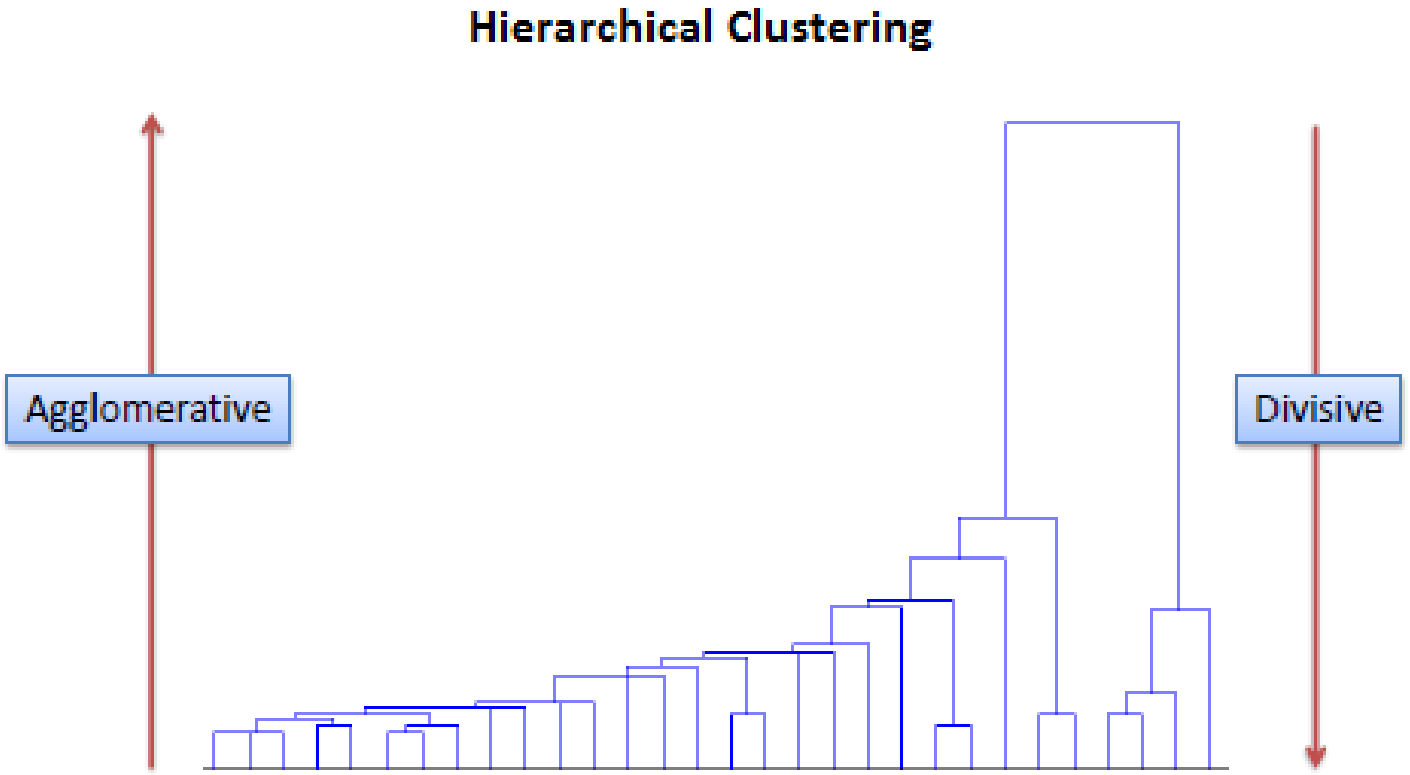
沃德法（Ward's method）：

- 沃德法（Ward's method）：群聚間的距離定義為在將兩群合併後，各點到合併後的群中心的距離平方和（**m** 表示 $C_i \cup C_j$ 的平均值）

$$d(C_i, C_j) = \sum_{\mathbf{a} \in C_i \cup C_j} \|\mathbf{a} - \mu\|,$$

階層式分群

- 聚合式、分裂式



聚合式分群

- 聚合式分群
  - 階層式分群法可由樹狀結構的底部開始，將資料或群聚逐次合併
  - 最終合併為一個大的群組
  - 使用hclust

Given:

A set X of objects $\{x_1,\ldots,x_n\}$

A distance function $dist(c_1,c_2)$

for $i = 1$ to n

  $c_i = \{x_i\}$

end for

$C = \{c_1,\ldots,c_n\}$

$l = n+1$

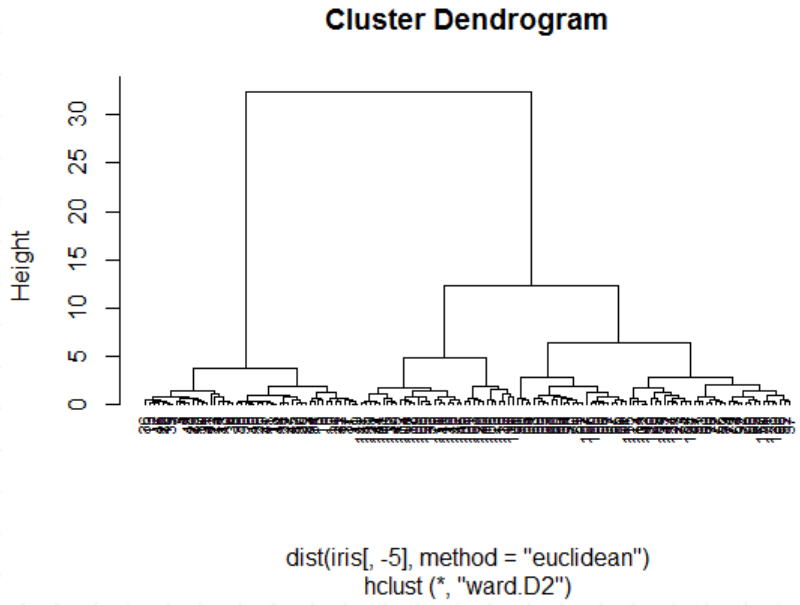
while $C.size > 1$ **do**

- (c_{min1},c_{min2}) = minimum $dist(c_i,c_j)$ for all c_i,c_j in C
- remove c_{min1} and c_{min2} from C
- add $\{c_{min1},c_{min2}\}$ to C
- $l = l + 1$

end while

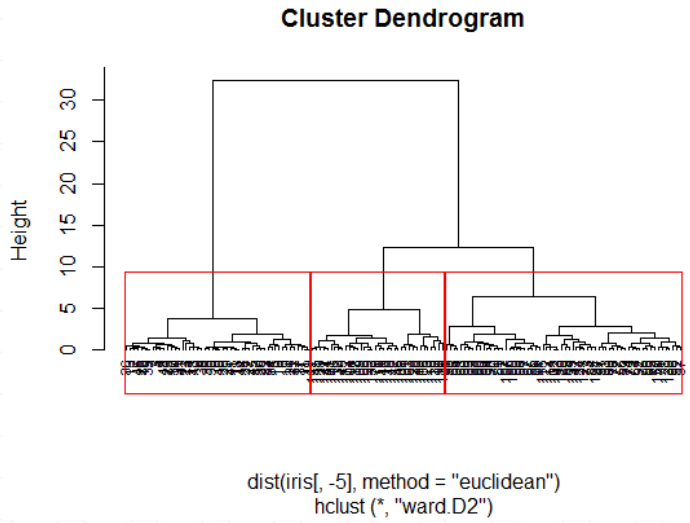
使用hclust 做iris 分群

```
data(iris)
hc = hclust(dist(iris[,-5], method="euclidean"), method="ward.D2")
plot(hc, hang = -0.01, cex = 0.7)
```



使用cutree樹做分群

```
fit = cutree(hc, k = 3)
table(fit)
plot(hc, hang = -0.01, cex = 0.7)
rect.hclust(hc, k = 3 , border="red")
```

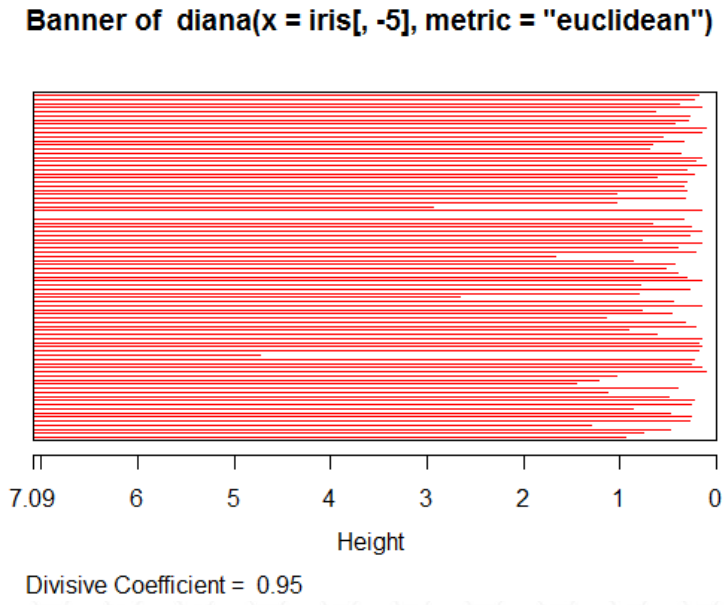


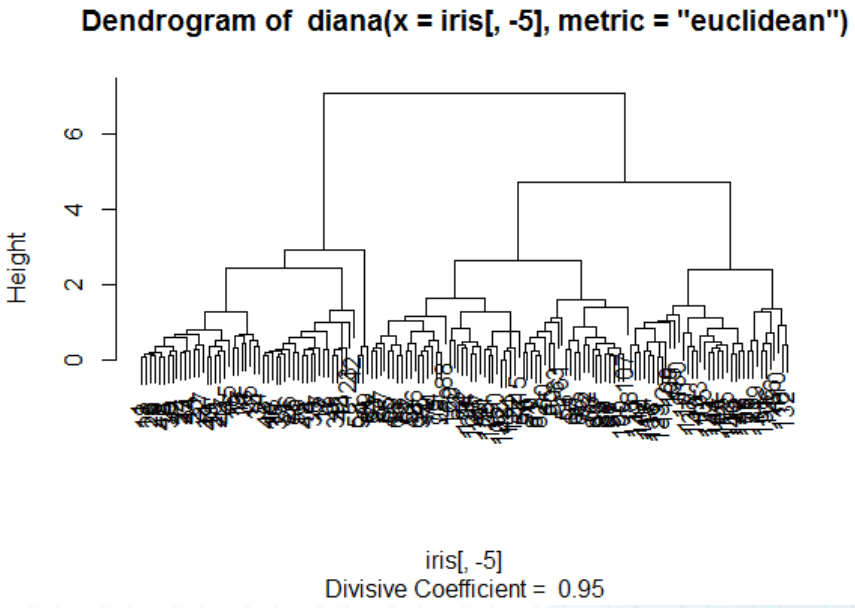
分裂式分群

- 分裂式分群
  - 由樹狀結構的頂端開始，將群聚逐次分裂，直到所以觀測值都被分到屬於自己單一個群組
  - diana {cluster}

分裂式分群法

```
library(cluster)
dv = diana(iris[,-5], metric = "euclidean")
summary(dv)
plot(dv)
```





階層式分群的優點/缺點

- 優點
  - 可以產生視覺化分群結果 (使用plot)
  - 可以等結構產生後，再使用cutree進行分群
  - 不用一開始決定要分多少群
- 缺點
  - 計算速度緩慢(採用遞迴式聚合或分裂)

文章分群

文章相關的相似度

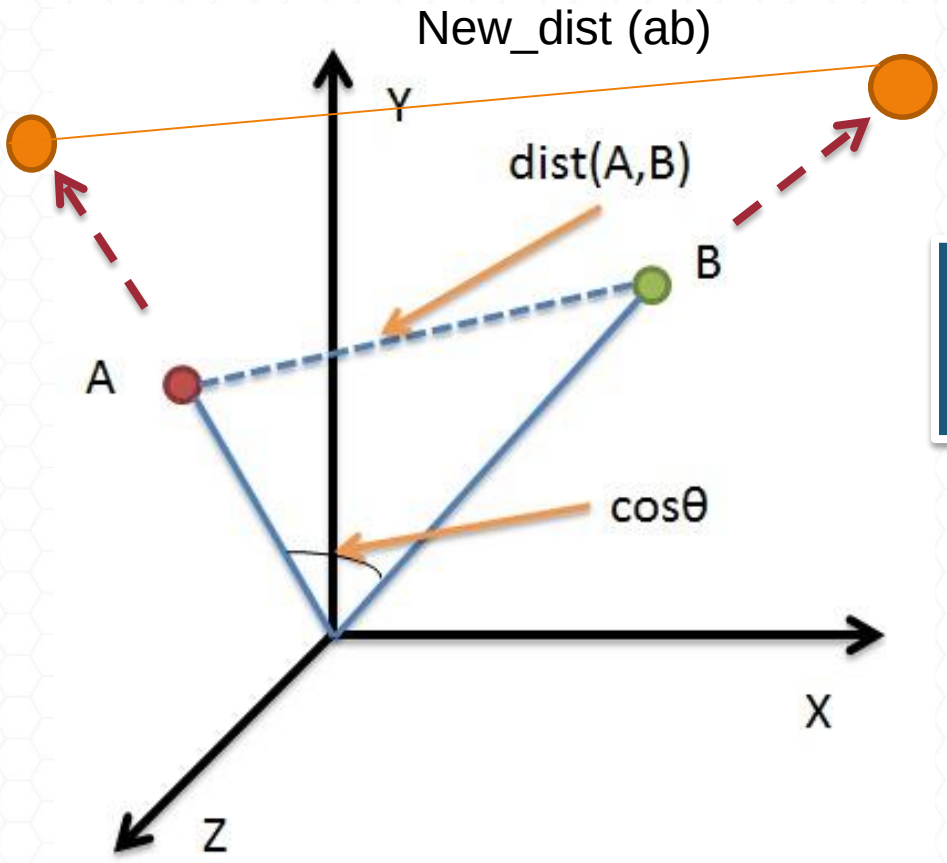
1. 使用TF-IDF演算法，找出兩篇文章的關鍵字

2. 每篇文章各取出若干個關鍵字（比如20個），合併成一個集合，計算每篇文章對於這個集合中的詞的詞頻

3. 生成兩篇文章各自的詞頻向量

4. 計算兩個向量的余弦相似度，值越大就表示越相似。

Euclidean Distance v.s. Cosine Distance



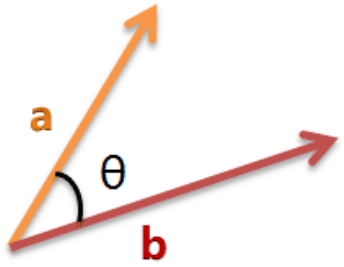
計算相對向量距離
而非絕對距離

實際範例

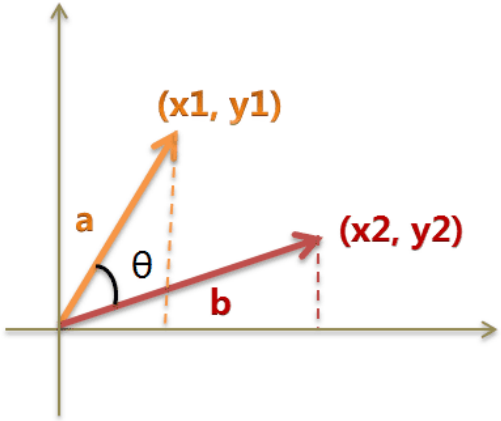
句子A：我 1，喜欢 2，看 2，电视 1，电影 1，不 1，也 0。
句子B：我 1，喜欢 2，看 2，电视 1，电影 1，不 2，也 1。

句子A：[1, 2, 2, 1, 1, 1, 0]
句子B：[1, 2, 2, 1, 1, 2, 1]

計算 Cosine Distance



$$\cos\theta = \frac{a^2 + b^2 - c^2}{2ab}$$



$$\cos\theta = \frac{x_1 x_2 + y_1 y_2}{\sqrt{x_1^2 + y_1^2} \times \sqrt{x_2^2 + y_2^2}}$$

計算Cosine Similarity

$$\cos\theta = \frac{\sum_{i=1}^{n}(A_i \times B_i)}{\sqrt{\sum_{i=1}^{n}(A_i)^2} \times \sqrt{\sum_{i=1}^{n}(B_i)^2}}$$

$$= \frac{A \cdot B}{|A| \times |B|}$$

句子A：[1, 2, 2, 1, 1, 1, 0]
句子B：[1, 2, 2, 1, 1, 2, 1]

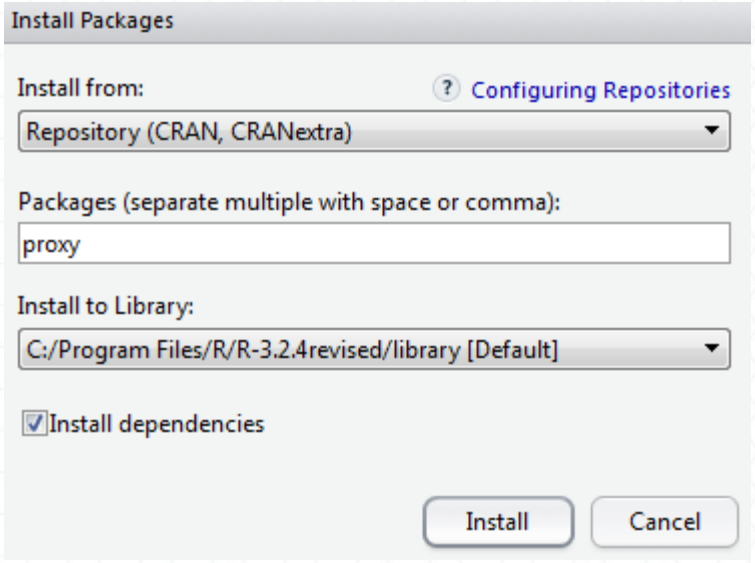
$$\cos\theta = \frac{1\times1 + 2\times2 + 2\times2 + 1\times1 + 1\times1 + 1\times2 + 0\times1}{\sqrt{1^2+2^2+2^2+1^2+1^2+1^2+0^2} \times \sqrt{1^2+2^2+2^2+1^2+1^2+2^2+1^2}}$$

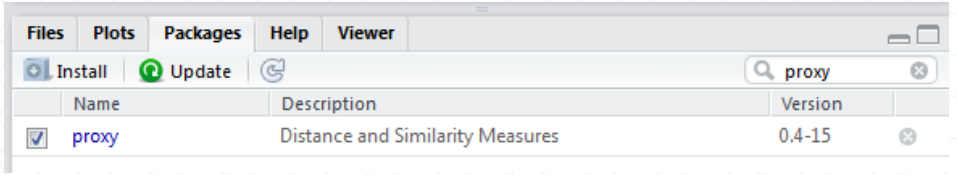
$$= \frac{13}{\sqrt{12} \times \sqrt{16}}$$

$$= 0.938$$

使用 proxy 套件計算cosine similarity

```
install.packages("proxy")
library(proxy)
```





1. 文章斷詞

```
load("applenews.RData")
library(jiebaR)
mixseg = worker()
apple.seg =lapply(applenews$content,
function(e)segment(code=e, jiebar=mixseg))
```

2. 建立詞頻矩陣

```
library(jiebaR)
mixseg = worker()
apple.seg =lapply(applenews$content,
       function(e)segment(code=e, jiebar=mixseg))

s.corpus <- CNCorpus(apple.seg)
control.list=list(wordLengths=c(2,Inf),
        tokenize=space_tokenizer)
s.corpus = tm_map(s.corpus, removeNumbers)
s.corpus = tm_map(s.corpus, removePunctuation)
dtm <- DocumentTermMatrix(s.corpus,
        control=control.list)
dim(dtm )
dtm.remove = removeSparseTerms(dtm, 0.99)
```

3. 計算文字間的cosine distance

```
dtm.dist = proxy::dist(as.matrix(dtm.remove),
method = "cosine")
dtm.mat = as.matrix(dtm.dist)
```

查詢最相似文章

```
applenews$title[order(dtm.mat[7,])[1:10]]
```

```
##  [1] "【央廣RTI】每318秒就有1人罹癌　大腸癌名列第一"
##  [2] "保持窈窕5個祕密　最後一個猜不到！"
##  [3] "十大癌症總發生人數上升　女罹乳癌飆增最兇"
##  [4] "【央廣RTI】台灣廉航快速成長　虎航市佔奪第一"
##  [5] "十大癌症大腸癌最兇猛　連續8年蟬聯冠軍"
##  [6] "癌症時鐘轉更快　每5分18秒就有1人罹癌"
##  [7] "【央廣RTI】以色列批准屯墾區新建住宅"
##  [8] "【台灣英文新聞】詐騙案讓台灣無光"
##  [9] "1年5黑熊斷掌　「黑熊媽媽」譴責補獸鋏"
## [10] "【央廣RTI】一起來「品東風」吧！文博會20日登場 "
```

查詢文章

根據相似度查詢

```
applenews$title[as.integer(names(sort(dtm.mat[18,
which(dtm.mat[18,] < 0.8)])))]
```

```
##  [1] "陸委會跨部會議確認　下周登陸展開肯亞案協商"
##  [2] "【法廣RFI】肯亞案:北京高調遣送　學者稱短期難返台"
##  [3] "【法廣RFI】肯亞案45台灣人均被拘北京海淀"
##  [4] "【法廣RFI】肯亞強遣台嫌回陸　目的何在?"
##  [5] "【法廣RFI】國台辦：堅決法辦肯亞詐騙台嫌犯"
##  [6] "四月十四日各報頭條搶先報"
##  [7] "【肯亞案】45台人遣中　陸委會：尚無掌握詐欺相關犯罪資訊"
##  [8] "【肯亞案】跨部會專案會議明召開　討論赴中交涉事宜"
##  [9] "【法廣RFI】印尼步肯亞後塵 台緊急行動防遣送陸"
## [10] "詐欺刑責太輕？　張善政指示研議修法"
## [11] "【更新】中國公布受害者數字　夏立言：拿出證據證明不是說說"
## [12] "肯亞將台灣人遣送中國　美國最新回應"
## [13] "肯亞案確定組團赴中　羅瑩雪：最快下周一出發"
## [14] "【更新】大馬50台人再被遣中？　陸委會：協商中"
## [15] "【肯亞案】邱太三自爆：中方曾發簡訊　不要在台上吵"
```

查詢文章

文章查詢函式

```
article.query = function(idx){
  applenews$title[as.integer(names(sort(dtm.mat[idx,
which(dtm.mat[idx,] < 0.8)])))]
}
article.query(18)[1:10]
```

使用cosine 距離分群

```
dtm.cluster = hclust(dtm.dist)
fit = cutree(dtm.cluster, k = 20)
applenews$title[fit == 16]
```

```
##  [1] "<U+200B>想看勇士季後賽　最少要花6700元　"
##  [2] "哈潑百轟出爐是支滿貫砲　助國民擊敗勇士"
##  [3] "【影片】勇士73勝　打破NBA單季最多勝紀錄"
##  [4] "中信兄弟最新喊聲　駒擊(跳兩下)！"
##  [5] "【體育動新聞】Curry神準三分球"
##  [6] "MLB美國職棒今日戰果"
##  [7] "衛冕大方祝賀勇士打破公牛的紀錄 "
##  [8] "NBA瘋狂夜日本也有感　愛勇士多過Kobe"
##  [9] "NBA今日戰績　勇士73勝達標"
## [10] "勇士隊與柯瑞創2大NBA紀錄　網友卻表示..."
## [11] "柯瑞單季402記3分球　創恐怖的柯瑞障礙"
## [12] "勇士破公牛紀錄　公牛迷歐巴馬認了"
## [13] "勇士、柯神創神蹟　PTT鄉民搶神串留名見證歷史"
## [14] "勇士本季的破紀錄之旅"
```

THANK YOU