

數據分析師假日精修班

Lab5

David Chiu
2016/12/17

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

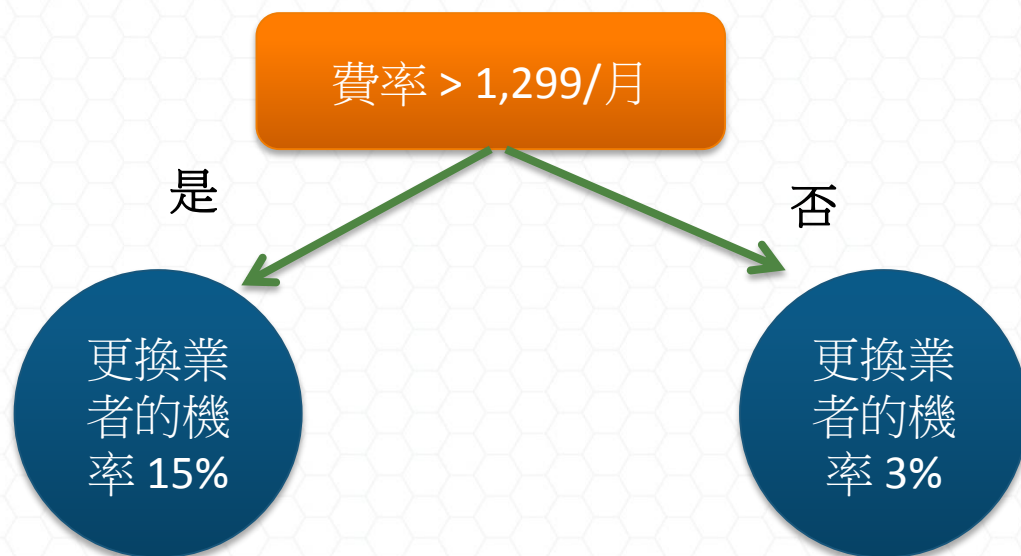
A computer program is said to learn from experience E with respect to some task T and some performance measure P , if its performance on T , as measured by P , improves with experience E

-- Tom Mitchell (1998)

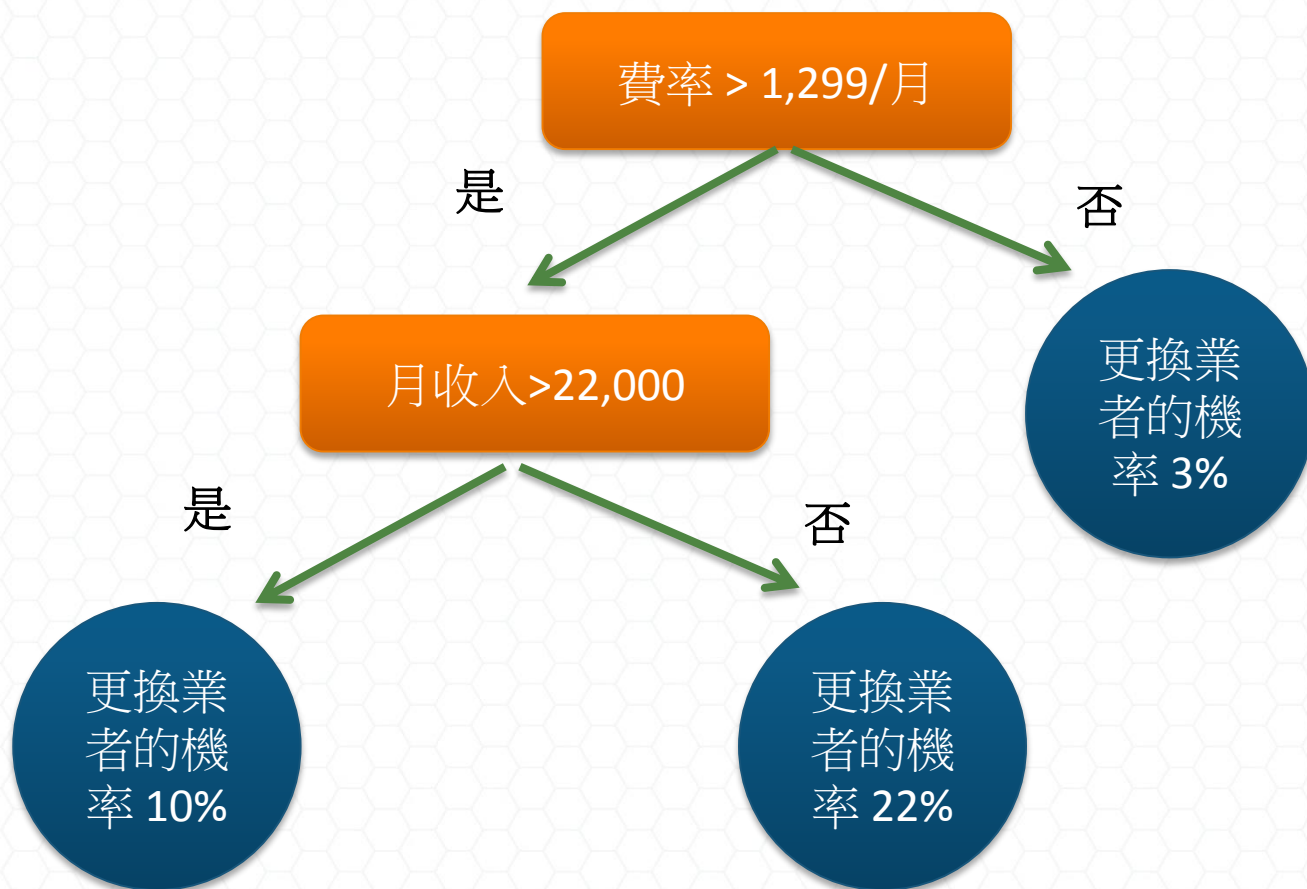
■ 找出有效的預測模型

- 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

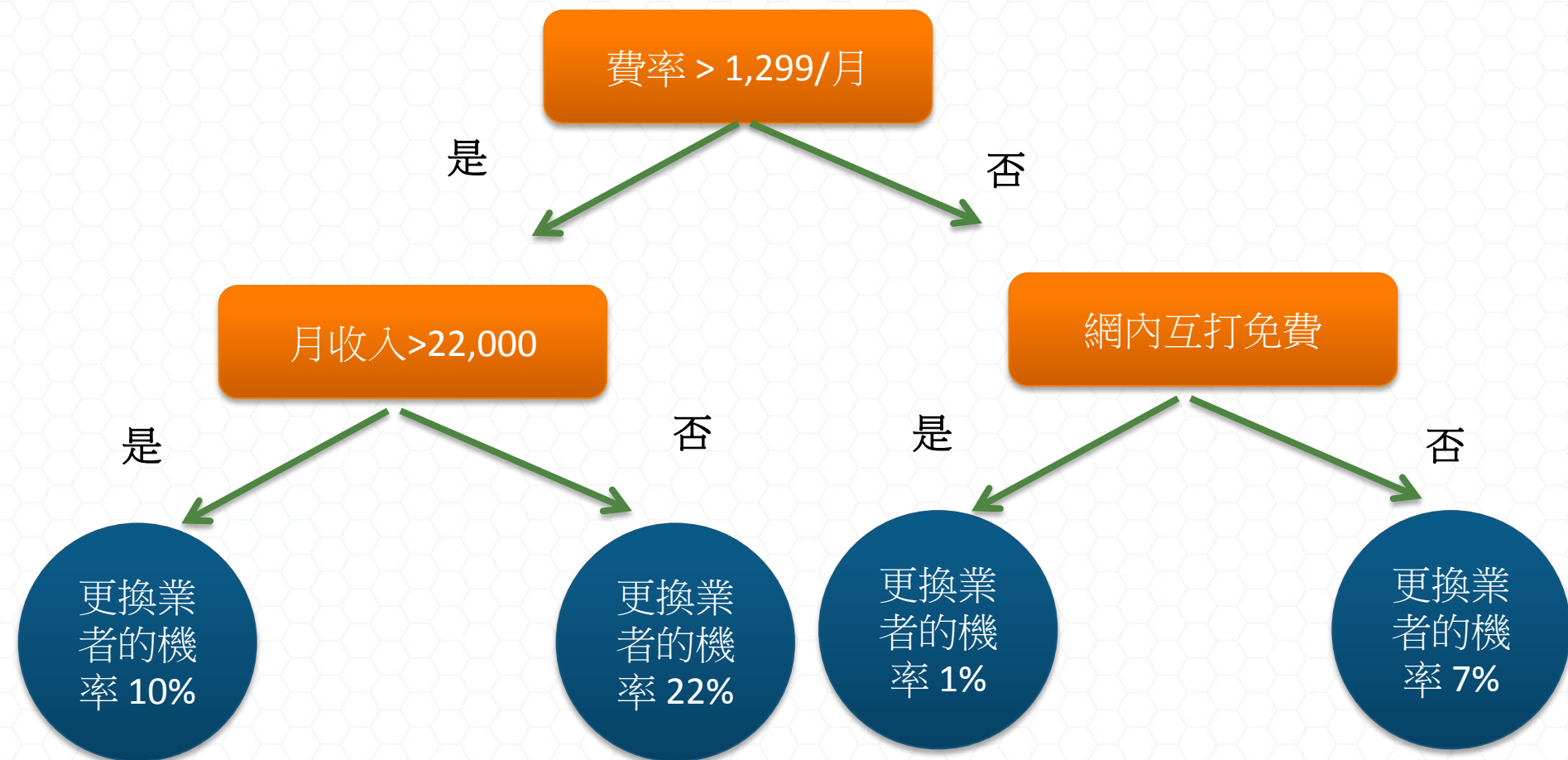
簡單的分類問題(決策樹)



簡單的分類問題(決策樹)



簡單的分類問題(決策樹)

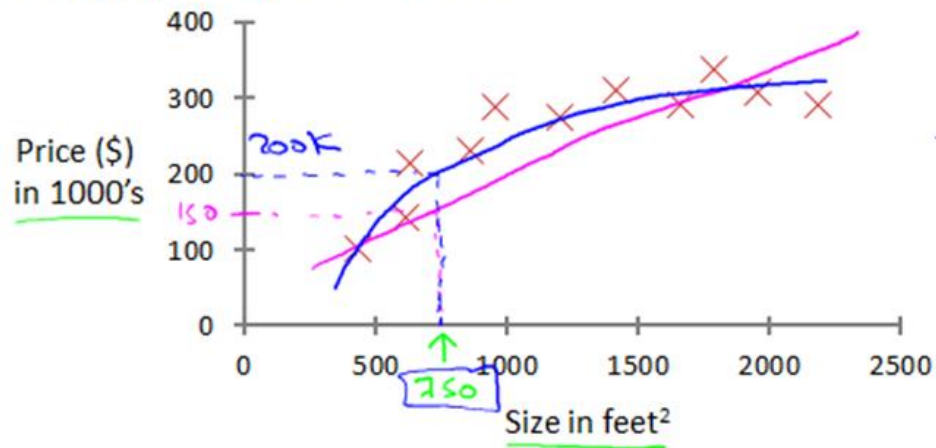


機器學習問題分類

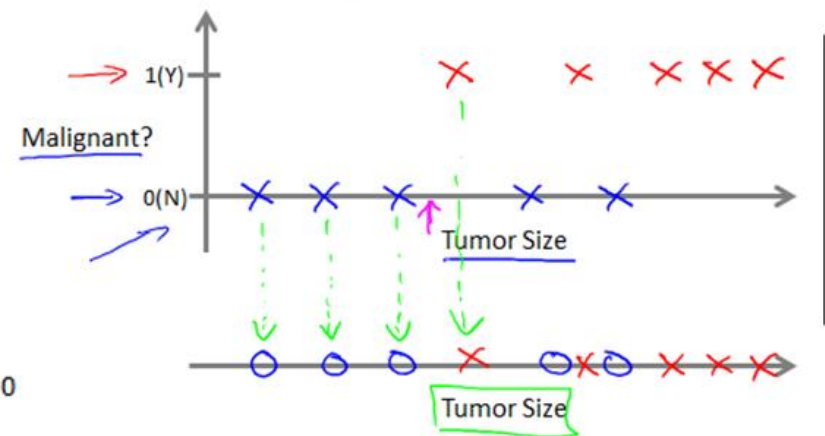
- 監督式學習 (Supervised Learning)
 - 迴歸分析 (Regression)
 - 分類問題 (Classification)
- 非監督式學習 (Unsupervised Learning)
 - 降低維度 (Dimension Reduction)
 - 分群問題 (Clustering)

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

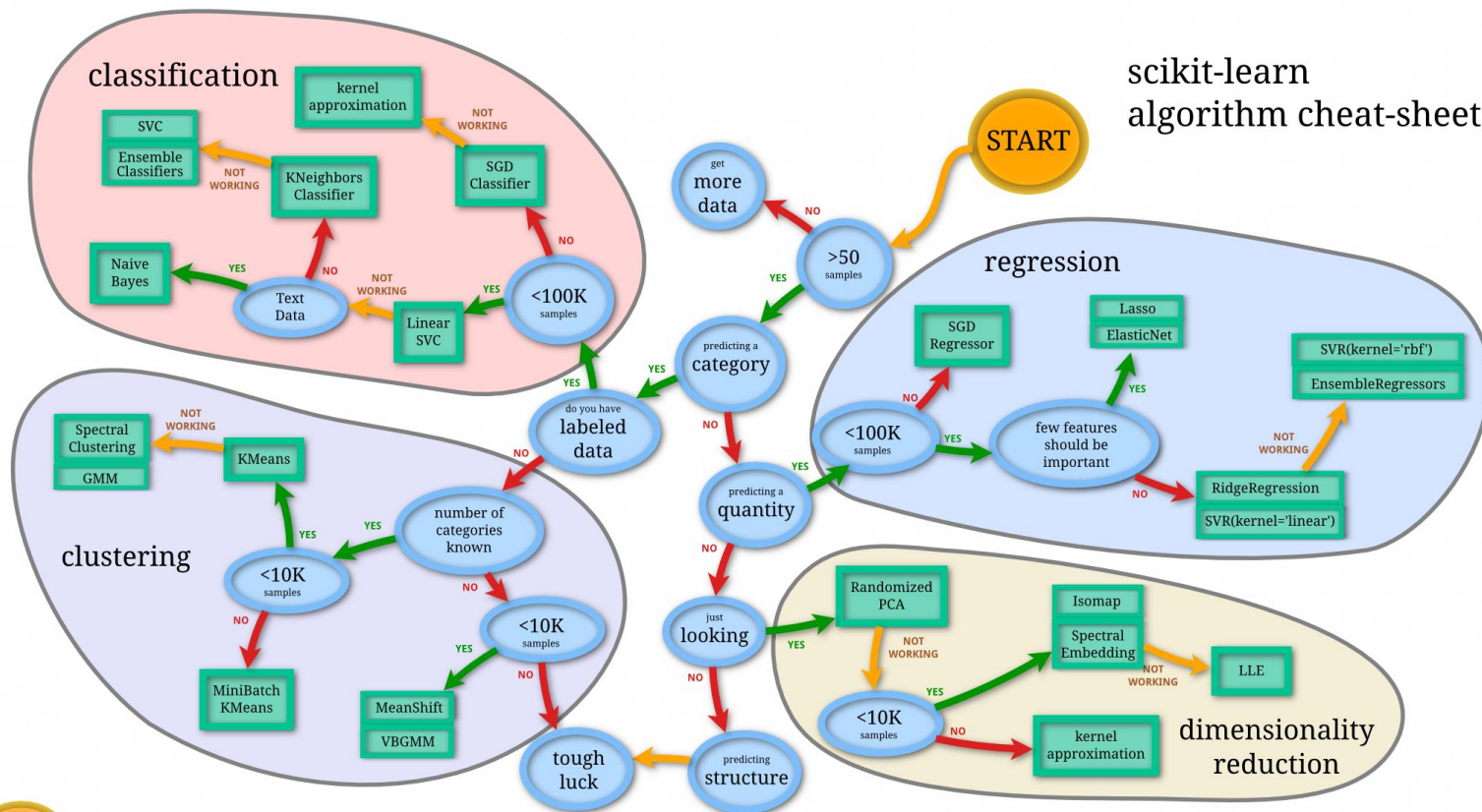
- 產生一有最大變異數的欄位線性組合，可用來降低原本問題的維度與複雜度
- e.g. 濃縮用到的特徵，編纂成一個新指標

■ 分群問題

- 物以類聚 (近朱者赤、近墨者黑)
- e.g. 將客戶分層

機器學習地圖

http://scikit-learn.org/stable/_static/ml_map.png



分類方法簡介

如何分類鳶尾花 (iris)

■ https://en.wikipedia.org/wiki/Iris_flower_data_set



Iris setosa



Iris versicolor



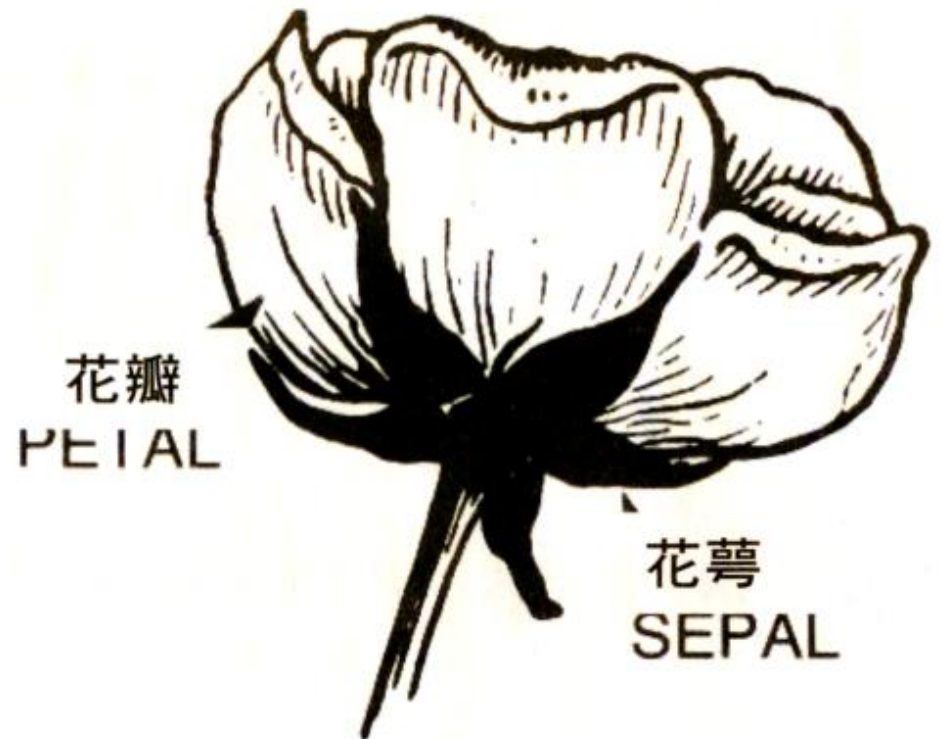
Iris virginica

Sepal? Petal?

■ 如何從花萼與花瓣長寬分辨花種？

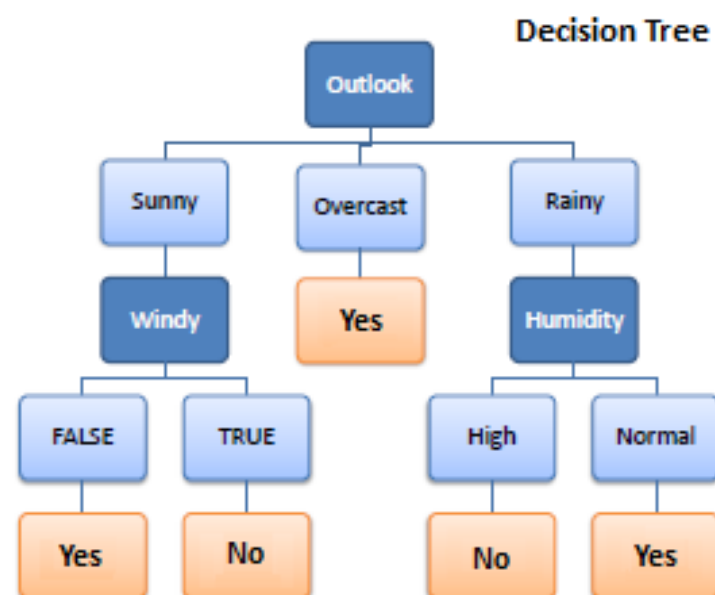
Fisher's Iris Data

Sepal length ⇄	Sepal width ⇄	Petal length ⇄	Petal width ⇄	Species ⇄
5.1	3.5	1.4	0.2	<i>I. setosa</i>
4.9	3.0	1.4	0.2	<i>I. setosa</i>
4.7	3.2	1.3	0.2	<i>I. setosa</i>
4.6	3.1	1.5	0.2	<i>I. setosa</i>
5.0	3.6	1.4	0.2	<i>I. setosa</i>
5.4	3.9	1.7	0.4	<i>I. setosa</i>
4.6	3.4	1.4	0.3	<i>I. setosa</i>
5.0	3.4	1.5	0.2	<i>I. setosa</i>



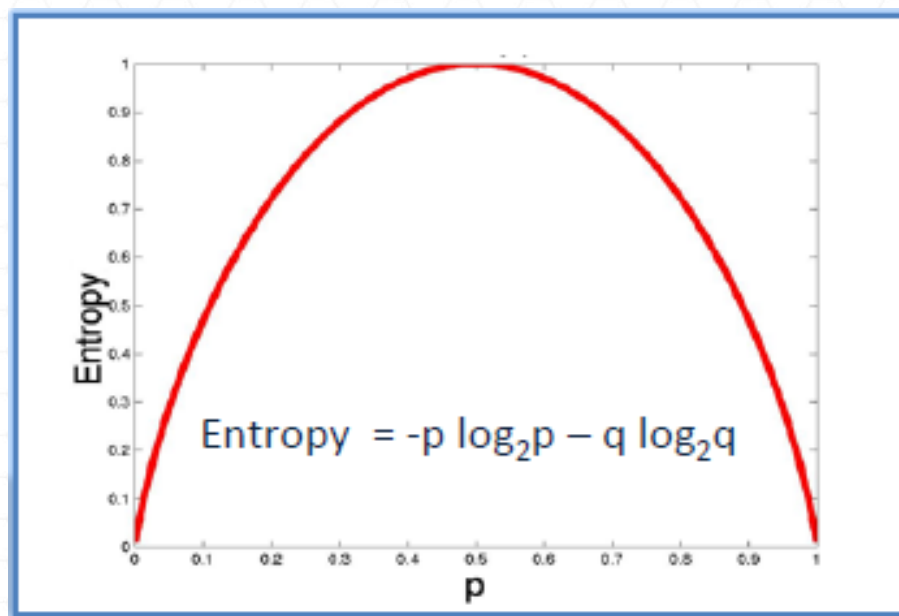
決策樹

Predictors				Target
Outlook	Temp.	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No



Entropy

- 用於計算一個系統中的失序現象，也就是計算該系統混亂的程度



$$\text{Entropy} = -0.5 \log_2 0.5 - 0.5 \log_2 0.5 = 1$$

單一變數的計算

$$E(S) = \sum_{i=1}^c -p_i \log_2 p_i$$

Play Golf	
Yes	No
9	5



Entropy(PlayGolf) = Entropy (5,9)
= Entropy (0.36, 0.64)
= - (0.36 \log_2 0.36) - (0.64 \log_2 0.64)
= 0.94

多變數的計算

$$E(T, X) = \sum_{c \in X} P(c)E(c)$$

		Play Golf		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5
				14



$$\begin{aligned} E(\text{PlayGolf}, \text{Outlook}) &= P(\text{Sunny}) * E(3,2) + P(\text{Overcast}) * E(4,0) + P(\text{Rainy}) * E(2,3) \\ &= (5/14) * 0.971 + (4/14) * 0.0 + (5/14) * 0.971 \\ &= 0.693 \end{aligned}$$

Information Gain

- 根據分割(Split)後，所減少的Entropy
- 因此做分割時，會尋找最大的Information Gain

1. 計算Entropy

$$\begin{aligned}\text{Entropy}(\text{PlayGolf}) &= \text{Entropy}(5,9) \\ &= \text{Entropy}(0.36, 0.64) \\ &= - (0.36 \log_2 0.36) - (0.64 \log_2 0.64) \\ &= 0.94\end{aligned}$$

計算Information Gain

		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3
Gain = 0.247			

		Play Golf	
		Yes	No
Temp.	Hot	2	2
	Mild	4	2
	Cool	3	1
Gain = 0.029			


		Play Golf	
		Yes	No
Humidity	High	3	4
	Normal	6	1
Gain = 0.152			

		Play Golf	
		Yes	No
Windy	False	6	2
	True	3	3
Gain = 0.048			

$$Gain(T, X) = Entropy(T) - Entropy(T, X)$$

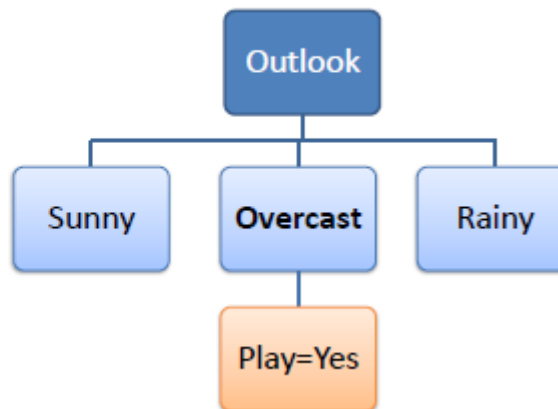
$$\begin{aligned} G(\text{PlayGolf}, \text{Outlook}) &= E(\text{PlayGolf}) - E(\text{PlayGolf}, \text{Outlook}) \\ &= 0.940 - 0.693 = 0.247 \end{aligned}$$

選擇有最大Information Gain的屬性

		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3
Gain = 0.247			

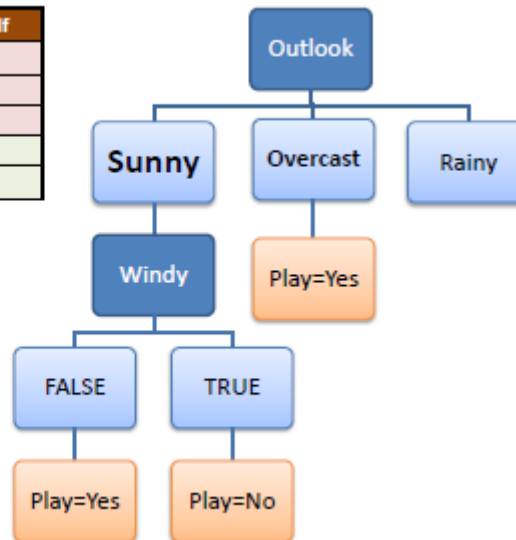
選擇子節點與分割節點

Temp	Humidity	Windy	Play Golf
Hot	High	FALSE	Yes
Cool	Normal	TRUE	Yes
Mild	High	TRUE	Yes
Hot	Normal	FALSE	Yes
Hot	High	FALSE	Yes



Entropy 為 0
則為子節點

Temp	Humidity	Windy	Play Golf
Mild	High	FALSE	Yes
Cool	Normal	FALSE	Yes
Mild	Normal	FALSE	Yes
Cool	Normal	TRUE	No
Mild	High	TRUE	No



Entropy 非 0
則為分割節點

決策樹如同IF...ELSE

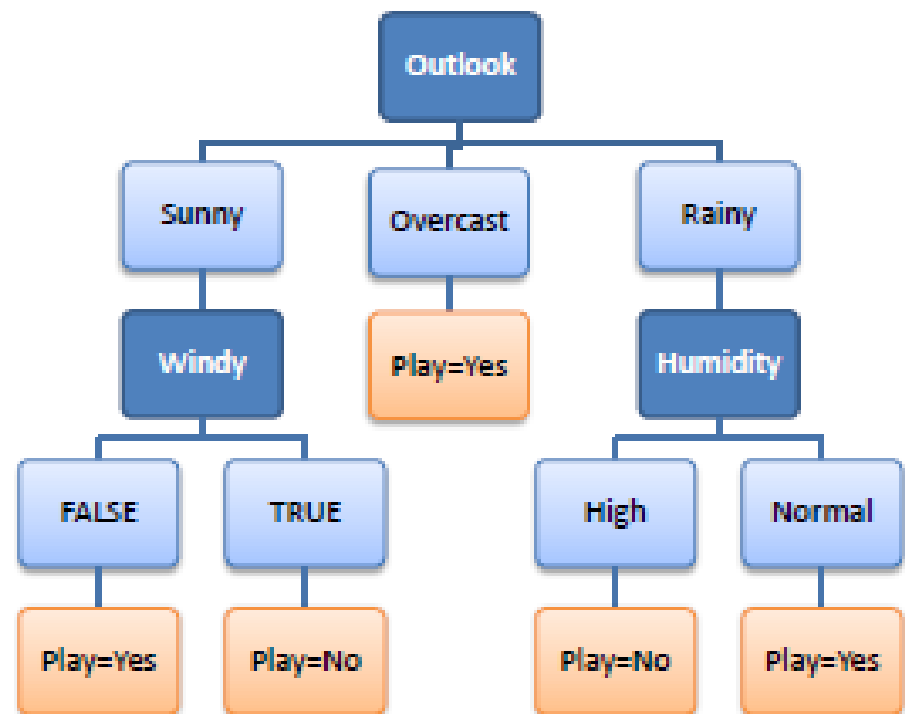
R_1 : IF (Outlook=Sunny) AND
(Windy=FALSE) THEN Play=Yes

R_2 : IF (Outlook=Sunny) AND
(Windy=TRUE) THEN Play=No

R_3 : IF (Outlook=Overcast) THEN
Play=Yes

R_4 : IF (Outlook=Rainy) AND
(Humidity=High) THEN Play=No

R_5 : IF (Outlook=Rain) AND
(Humidity=Normal) THEN
Play=Yes



rpart 與遞迴分割法

■ rpart

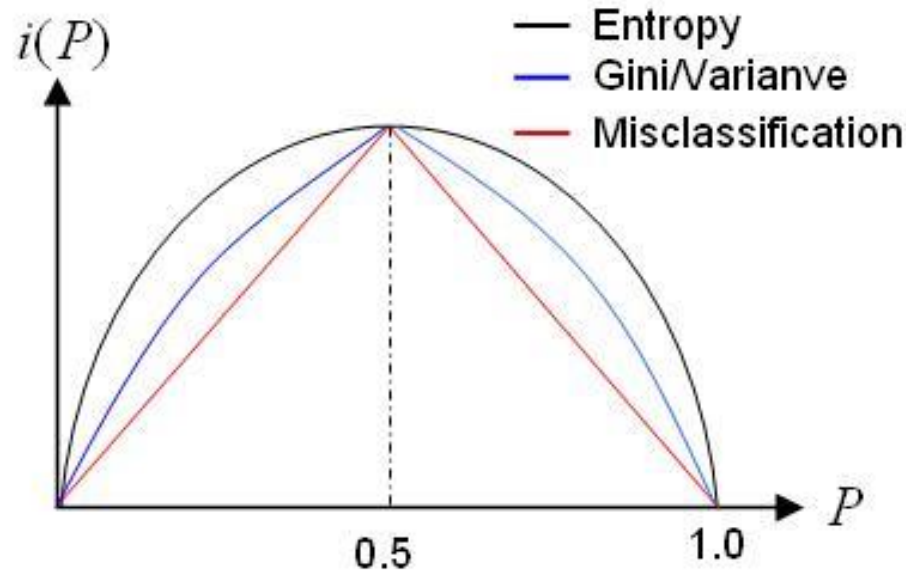
- 對所有參數和所有分割點進行評估
- 最佳的選擇是使分割後組內的資料更為“一致”(pure)
 - “一致”是指組內資料的因變數取值變異較小
- 使用Gini 值量測“一致”性
- 遞迴分割法 (Recursive Partitioning Tree)
- 使用“剪枝” (prune) 方法
 - 先建立一個劃分較細較為複雜的樹模型
 - 根據交叉檢驗(Cross-Validation)的方法來估計不同“剪枝”條件下
 - 選擇誤差最小的樹模型

Ctree 與條件推斷決策樹

■ Party

- 根據統計檢驗來確定參數和分割點的選擇
 - 先假設所有參數與因變數均獨立
 - 對它們進行卡方獨立檢驗
 - 檢驗P值小於閾值的引數加入模型
 - 相關性最強的引數作為第一次分割的引數
- 參數選擇好後，用置換檢驗來選擇分割點
- 用party建立的決策樹不需要剪枝(Prune)
 - 因為閾值就決定了模型的複雜程度。

Gini Impurity



範例:

Prob (晴天) = 0.4

Prob (陰天) = 0.3

Prob (雨天) = 0.3,

$$\text{Gini Index} = 1 - \sum_j p_j^2$$

$$\text{Gini Index} = 1 - (0.4^2 + 0.3^2 + 0.3^2) = 0.660$$

使用rpart 做出分類結果

```
library(rpart)
```

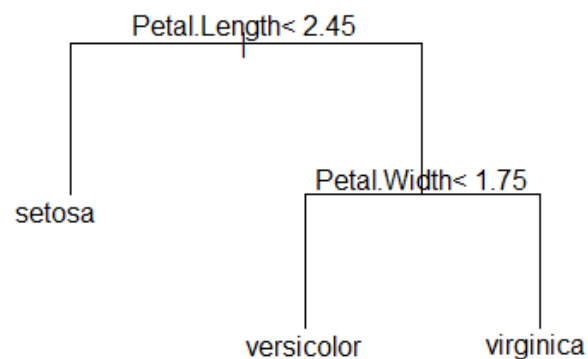
```
data(iris)
```

```
fit <- rpart(Species ~ Sepal.Length + Sepal.Width +  
Petal.Length + Petal.Width, data=iris)
```

```
summary(fit)
```

```
plot(fit, margin = 0.1)
```

```
text(fit)
```

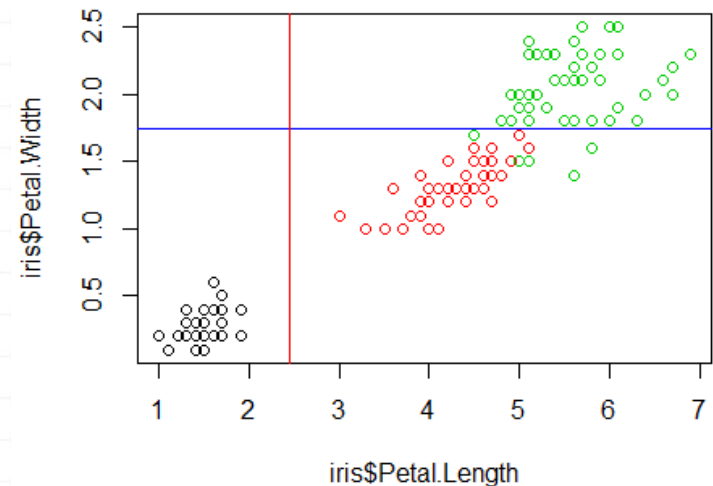


將分類結果顯示在圖上

```
plot(iris$Petal.Length, iris$Petal.Width,  
col=iris$Species)
```

```
abline(h = 1.75, col="blue")
```

```
abline(v = 2.45, col="red")
```



觀看分類結果

```
table(predict(fit, iris[,1:4], type="class"), iris[,5])
```

	setosa	versicolor	virginica
setosa	50	0	0
versicolor	0	49	5
virginica	0	1	45

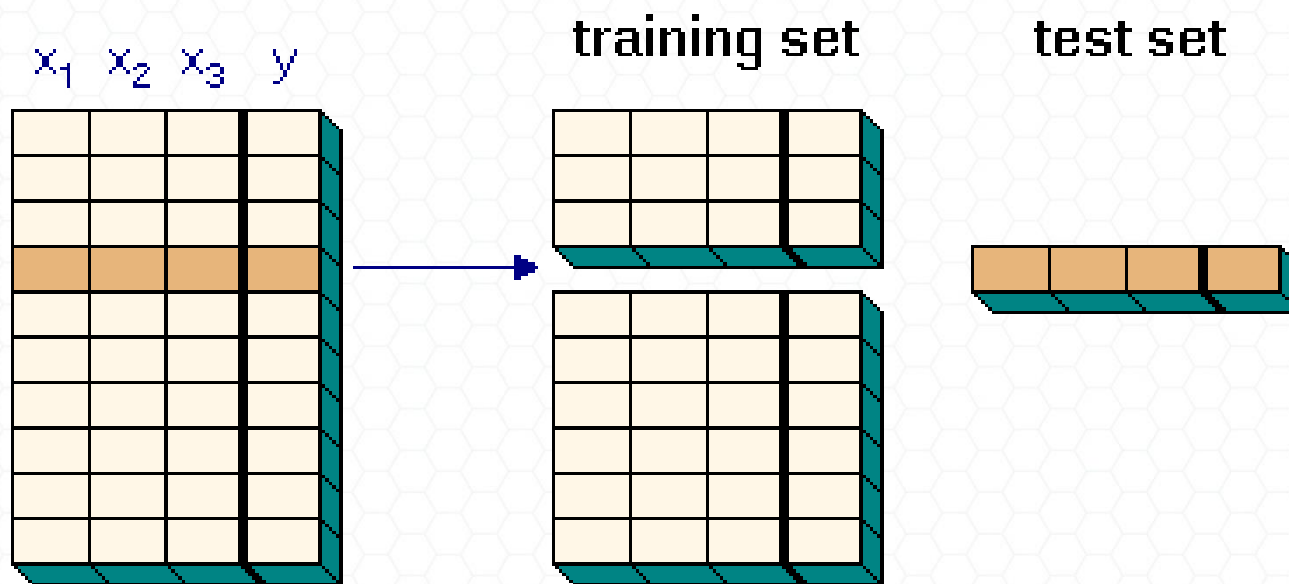
要驗證是否有過度學習?

使用caret 套件找出準確率

```
library(caret)  
cm <- table(predict(fit, iris[,1:4], type="class"),  
iris[,5])  
confusionMatrix(cm)
```

測試模型

- 使用外部資料或是一部分的內部資料來測試資料



訓練模型與測試模型都為同一份
有球員兼裁判的嫌疑

將資料分為訓練與測試資料集

固定產生亂數

```
set.seed(123)
```

```
idx <- sample.int(2, nrow(iris), replace=TRUE,  
prob=c(0.7,0.3))
```

70% 分為訓練資料集
30% 分為測試資料集

```
trainset <- iris[idx==1, ]
```

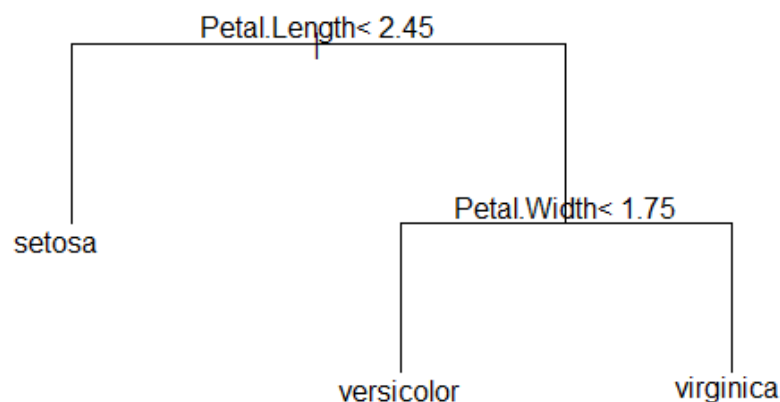
```
testset <- iris[idx==2, ]
```

```
dim(trainset)
```

```
dim(testset)
```


使用訓練資料集建立模型

```
fit2 <- rpart(Species ~., data=trainset)
plot(fit2, margin = 0.1)
text(fit2)
```



套用在測試資料集測試模型

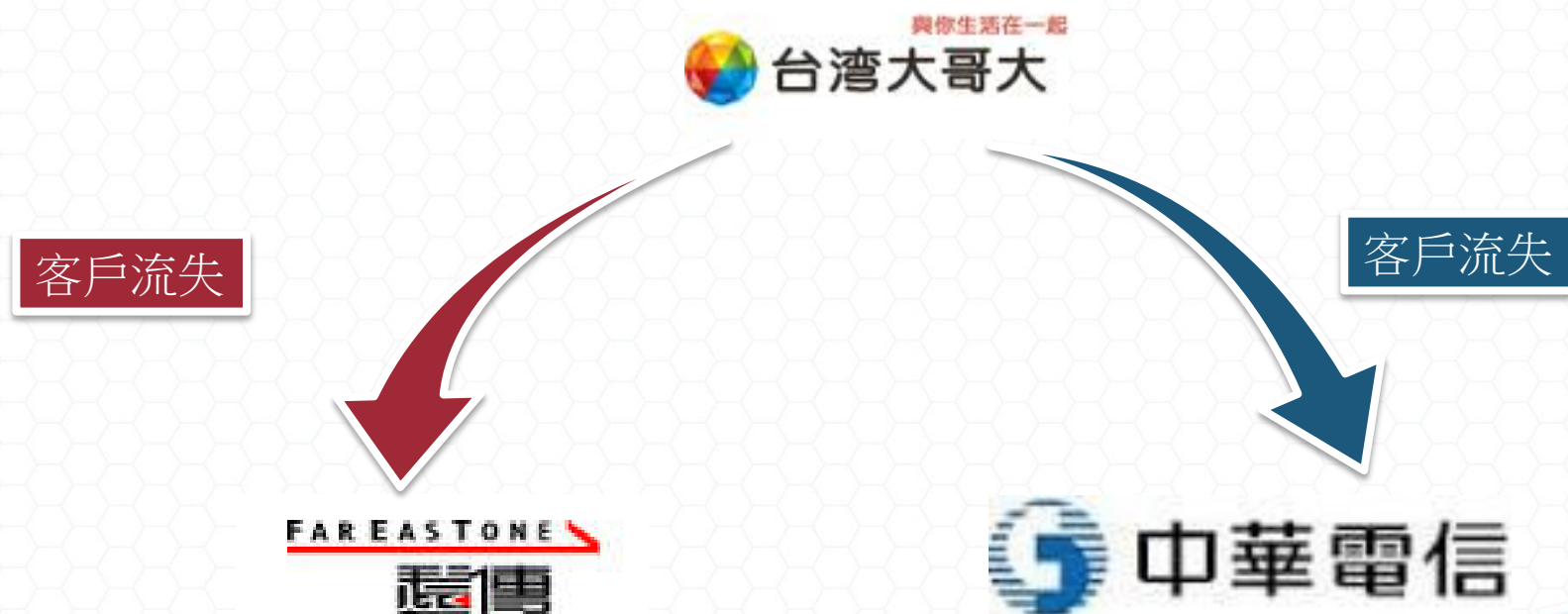
```
pred <- predict(fit2, testset[,-5], type= "class")  
cm <- table(pred, testset[,5])  
confusionMatrix(cm)
```

Accuracy : 0.9762
整體效果不錯

顧客流失分析

顧客流失分析

- 從顧客的通聯記錄預測哪些客戶容易更換電信業者？



把資料分成訓練與測試集

```
install.packages("C50")  
library(C50)  
data(churn)  
str(churnTrain)  
churnTrain = churnTrain[,! names(churnTrain) %in% c("state",  
"area_code", "account_length") ]  
set.seed(2)  
ind <- sample(2, nrow(churnTrain), replace = TRUE, prob=c(0.7, 0.3))  
trainset = churnTrain[ind == 1,]  
testset = churnTrain[ind == 2,]
```

分成70% 為訓練資料集
30%為測試資料集

資料敘述

■ 顧客基本資訊

- state
- account length.
- area code
- phone number

■ 使用者行為

- international plan
- voice mail plan, number vmail messages
- total day minutes, total day calls, total day charge
- total eve minutes, total eve calls, total eve charge
- total night minutes, total night calls, total night charge
- total intl minutes, total intl calls, total intl charge
- number customer service calls

■ 預測標的

- Churn (Yes/No)

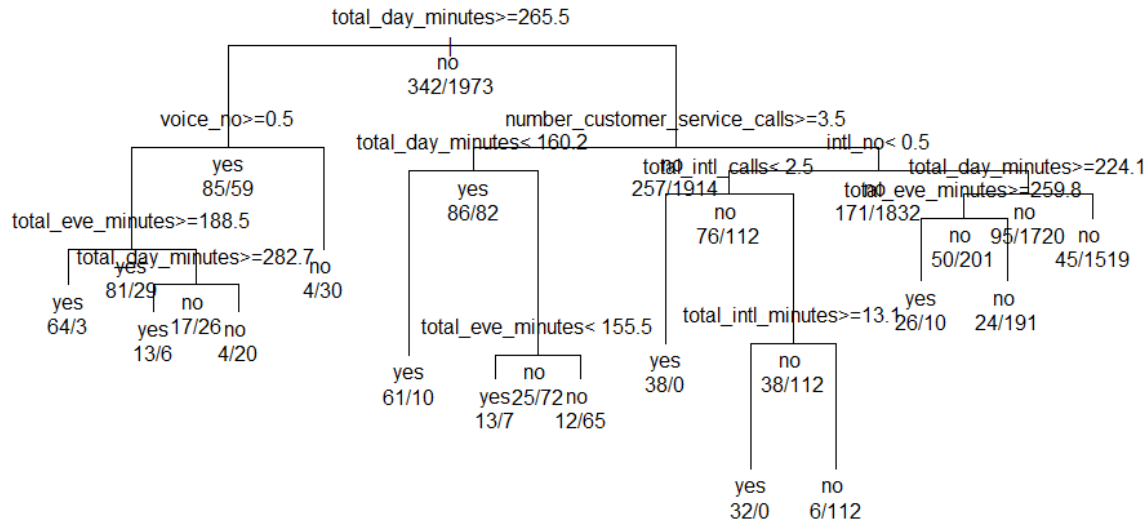
建立分類樹

```
churn.rp <- rpart(churn ~ ., data=trainset)
```

```
plot(churn.rp, margin= 0.1)
```

```
text(churn.rp, all=TRUE, use.n = TRUE)
```

機器學習



預測結果

```
predictions <- predict(churn.rp, testset, type="class")  
table(testset$churn, predictions)
```

pred	no	yes
no	859	18
yes	41	100

兩種分類結果(Yes/No)的
confusion Matrix 可以根據
真實類別跟預測結果分為
四種類別

評估結果

- True positive：代表檢測出有，且實際上有的狀況
- False positive：代表檢測出有，而實際上沒有的狀況
- True negative：代表檢測出無，且實際上無的狀況
- False negative：代表檢測出無，而實際上有的狀況

		真實狀況	
		真	假
檢測結果	有	檢測有，且為真 TP 真陽性	檢測有，但為假 FP 假陽性
	無	檢測無，但為真 FN 假陰性	檢測無，且為假 TN 真陰性

使用confusionMatrix

```
> confusionMatrix(table(predictions, testset$churn))
```

Confusion Matrix and Statistics

predictions yes no

yes 100 18

no 41 859

Accuracy : 0.942

95% CI : (0.9259, 0.9556)

No Information Rate : 0.8615

P-Value [Acc > NIR] : < 2.2e-16

Kappa : 0.7393

Mcnemar's Test P-Value : 0.004181

Sensitivity : 0.70922

Specificity : 0.97948

Pos Pred Value : 0.84746

Neg Pred Value : 0.95444

Prevalence : 0.13851

Detection Rate : 0.09823

Detection Prevalence : 0.11591

Balanced Accuracy : 0.84435

'Positive' Class : yes

K-fold cross-validation

■ Holdout 驗證

隨機從最初的樣本中選出部分，形成交叉驗證數據，而剩餘的就當做訓練數據。通常少於原本樣本三分之一的數據被選做驗證數據

■ K-fold cross-validation

K次交叉驗證，初始採樣分割成K個子樣本，一個單獨的子樣本被保留作為驗證模型的數據，其他K-1個樣本用來訓練，交叉驗證重複K次

■ 留一驗證

只使用樣本中的一項來當做驗證資料，而剩餘的則留下來當做訓練資料

如何進行 *K*-fold cross-validation

```
library(caret)
```

```
control = trainControl(method="repeatedcv",  
number=10, repeats=3)
```

```
model = train(churn~., data=trainset,  
method="rpart", preProcess="scale",  
trControl=control)
```

```
model
```

做三次10-Fold 交叉驗證

評估結果(續)

- True positive rate：代表所有陽性樣本中，得以正確檢測出陽性結果的機率，以 $TP/(TP+FN)$ 計算，又稱為靈敏度(sensitivity)。
- True negative rate，代表所有陰性樣本中，得以正確檢測出陰性結果的機率，以 $TN/(FP+TN)$ 計算，又稱為特異性(specificity)。
- False positive rate：代表所有陰性樣本中，檢測出假陽性的機率，以 $FP/(TN+FP)$ 計算，常以 $(1-SPC)$ 的方式呈現。

		真實狀況	
		真	假
檢測結果	有	檢測有，且為真 TP 真陽性 A	檢測有，但為假 FP 假陽性 B
	無	檢測無，但為真 FN 假陰性 C	檢測無，且為假 TN 真陰性 D

$TPR = \frac{A}{A+C}$ 真陽性率

$SPC = \frac{D}{B+D}$ 真陰性率

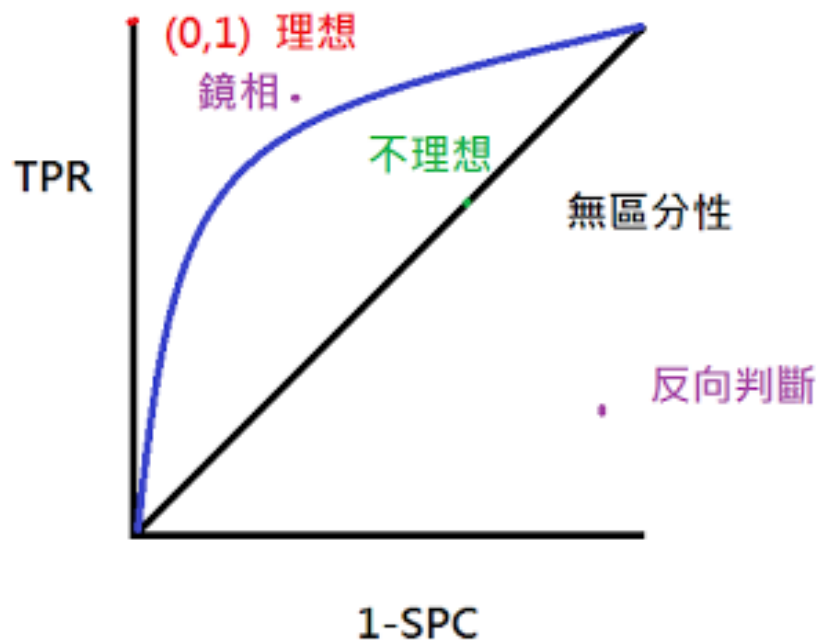
$1-SPC=FPR$

Confusion Matrix
會隨著限制條件不同而改變
該怎麼更客觀評估分類器的能力?

ROC 曲線

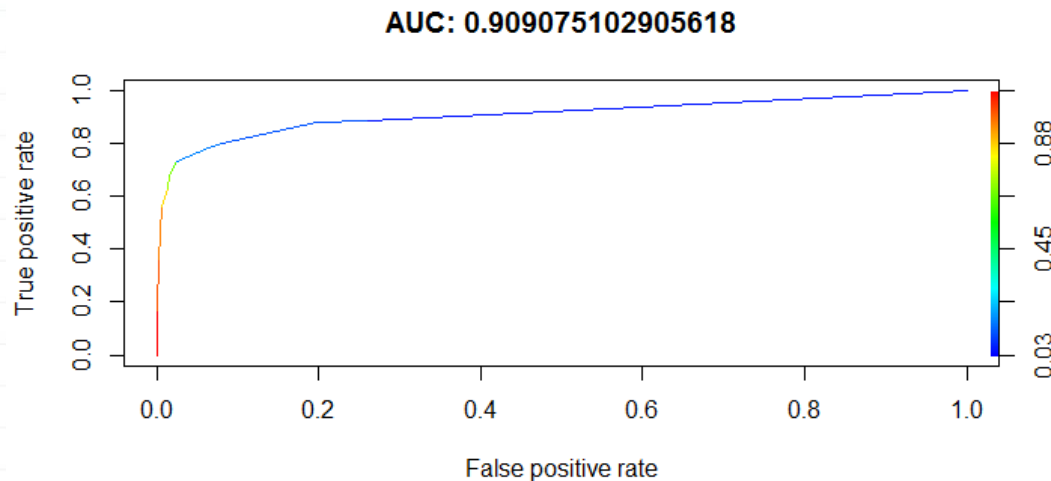
■ 接收者操作特徵(receiver operating characteristic, ROC curve)

- 1.以假陽性率(False Positive Rate, FPR)為X軸，代表在所有陰性相本中，被判斷為陽性(假陽性)的機率，又寫為(1-特異性)。
- 2.以真陽性率(True Positive Rate, TPR)為Y軸，代表在所有陽性樣本中，被判斷為陽性(真陽性)的機率，又稱為敏感性



使用測試資料集驗證預測能力

```
predictions <- predict(churn.rp, testset, type="prob")
pred.to.roc <- predictions[, 1]
pred.rocr <- prediction(pred.to.roc, as.factor(testset[, (dim(testset)[[2]])]))
perf.rocr <- performance(pred.rocr, measure = "auc", x.measure = "cutoff")
perf.tpr.rocr <- performance(pred.rocr, "tpr", "fpr")
plot(perf.tpr.rocr, colorize=T, main=paste("AUC:", (perf.rocr@y.values)))
```



AUC

曲線下面積(Area Under Curve, AUC)為此篩檢方式性能優劣之指標，AUC越接近1，代表此篩檢方式效能越佳。指標可參考以下條件。

AUC數值	解釋
1	完美分類器，無論cut-off point如何設定都可正確預測。通常不存在
$0.5 < \text{AUC} < 1$	優於隨機，妥善設定可有預測價值
0.5	同隨機，預測訊息沒有價值

如何找出最重要的變數

```
install.packages("rminer")
```

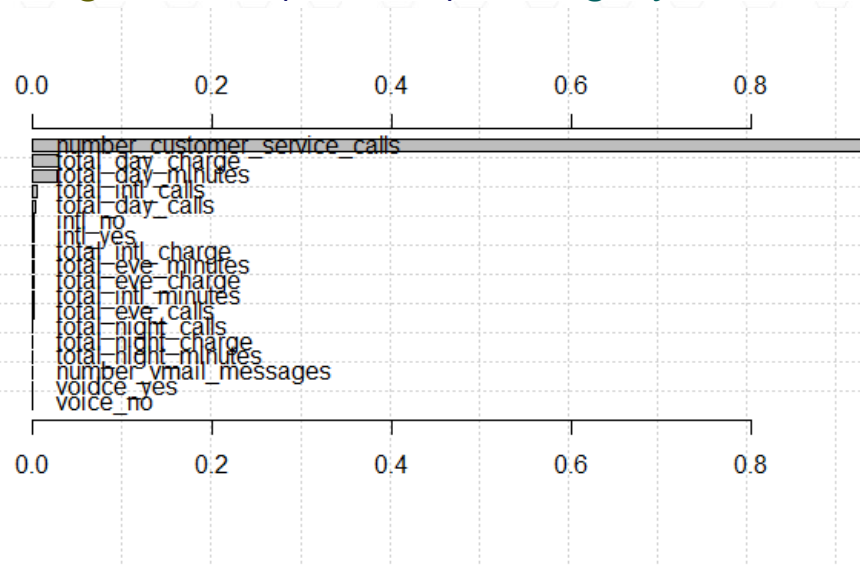
```
library(rminer)
```

```
model=fit(churn~.,trainset,model="rpart")
```

```
VariableImportance=Importance(model,trainset,method="sensv")
```

```
L=list(runs=1,sen=t(VariableImportance$imp),sresponses=VariableImportance$sresponses)
```

```
mgraph(L,graph="IMP",leg=names(trainset),col="gray",Grid=10)
```



可採取行動

■ 如果發現有客戶想要更換業者時? (查費率)

- 主動降低費率
- 提出更好的續約方案
- 送手機

■ PDCA 循環

- 用資料分析來擬訂策略



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)
```

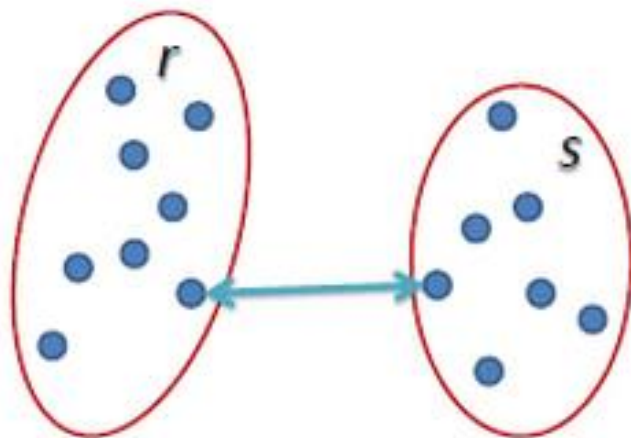

點間距離與群間距離

- 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

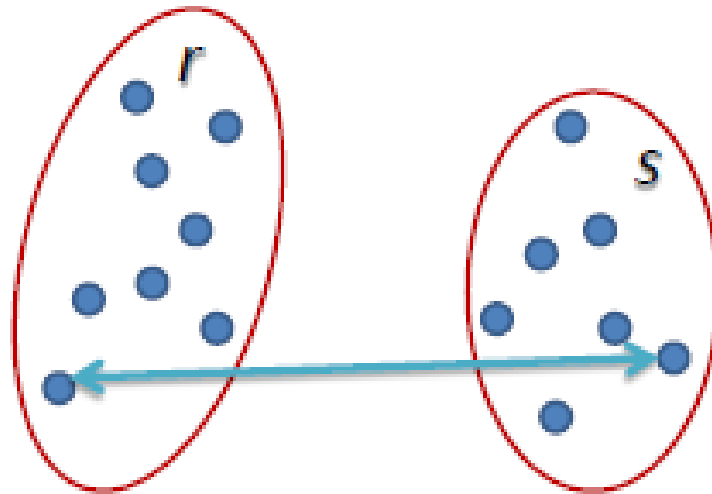
$$d(C_i, C_j) = \min_{a \in C_i, b \in C_j} d(a, b)$$



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

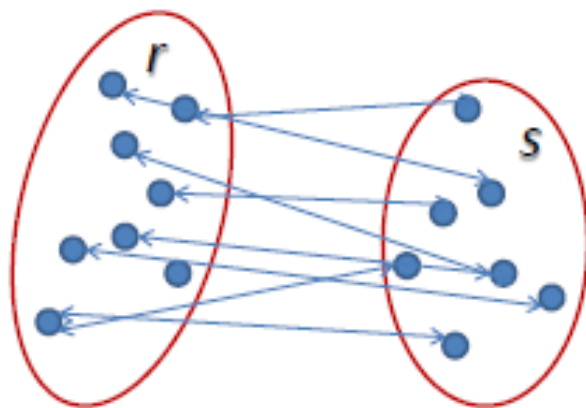
$$d(C_i, C_j) = \max_{a \in C_i, b \in C_j} d(a, b)$$



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) :

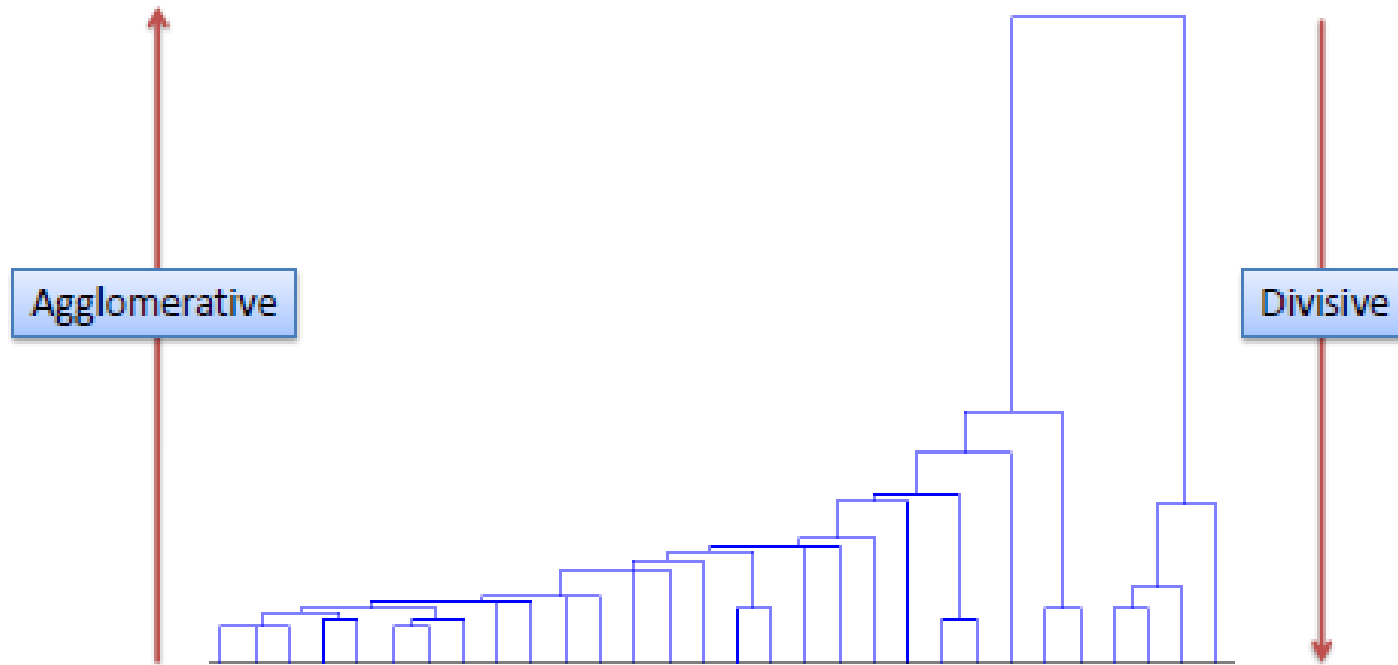
- 沃德法 (Ward's method) : 群聚間的距離定義為在將兩群合併後，各點到合併後的群中心的距離平方和 (\mathbf{m} 表示 $C_i \cup C_j$ 的平均值)

$$d(C_i, C_j) = \sum_{\mathbf{a} \in C_i \cup C_j} \|\mathbf{a} - \mu\|^2,$$

階層式分群

■ 聚合式、分裂式

Hierarchical Clustering



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 $\{x_1, \dots, x_n\}$

A distance function $dist(c_1, c_2)$

for $i = 1$ to n

$c_i = \{x_i\}$

end for

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

$l = n+1$

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

– $(c_{min1}, c_{min2}) = \text{minimum } dist(c_i, c_j) \text{ for all } c_i, c_j \text{ in } C$

– remove c_{min1} and c_{min2} from C

– add $\{c_{min1}, c_{min2}\}$ to C

– $l = l + 1$

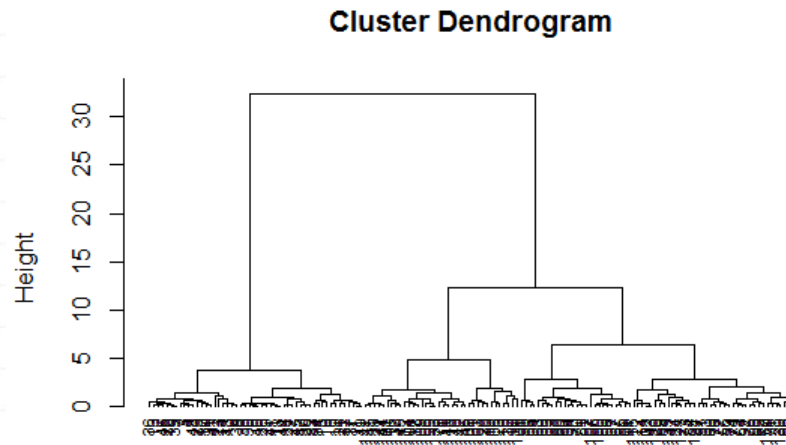
end while

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)
```



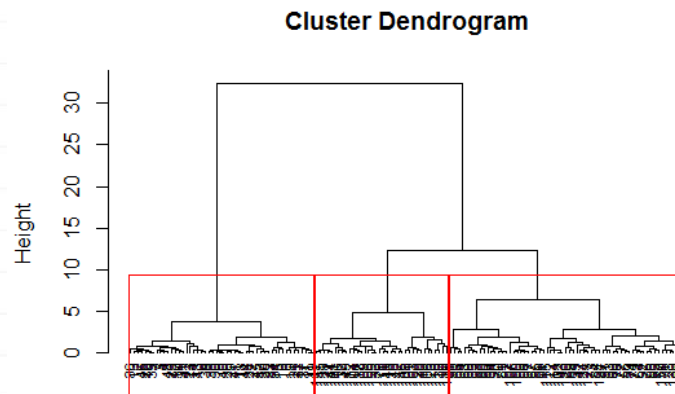
使用cutree樹做分群

```
fit = cutree(hc, k = 3)
```

```
table(fit)
```

```
plot(hc, hang = -0.01, cex = 0.7)
```

```
rect.hclust(hc, k = 3, border="red")
```



```
dist(iris[, -5], method = "euclidean")  
hclust (*, "ward.D2")
```

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

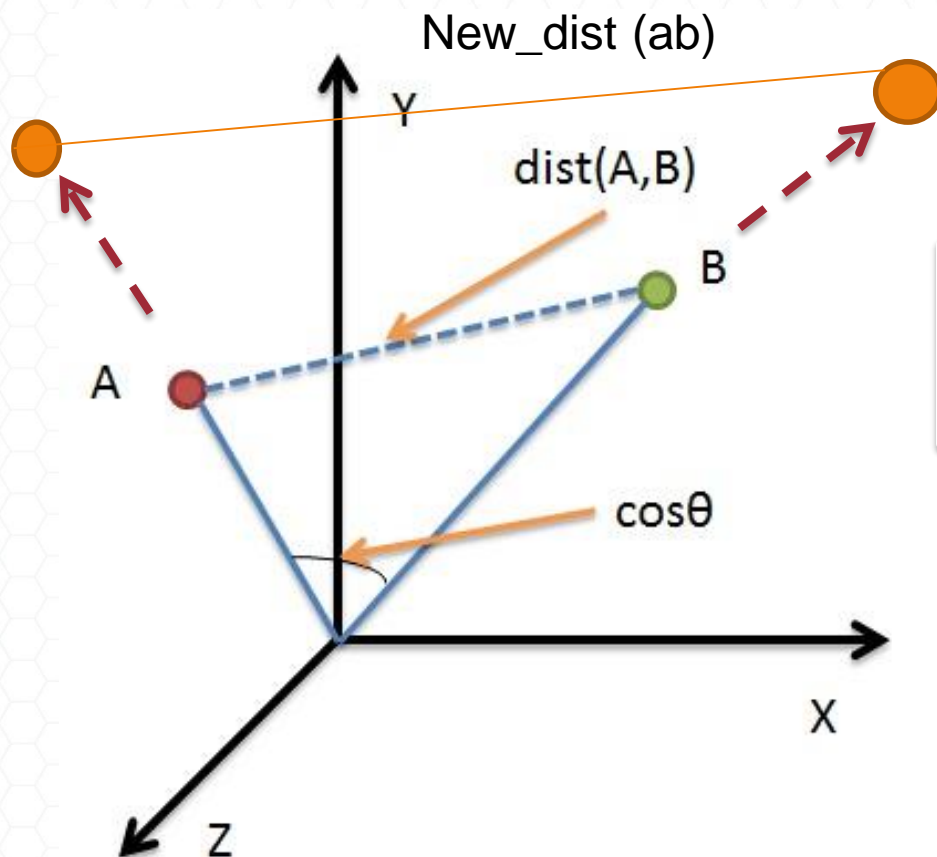
- 計算速度緩慢(採用遞迴式聚合或分裂)

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

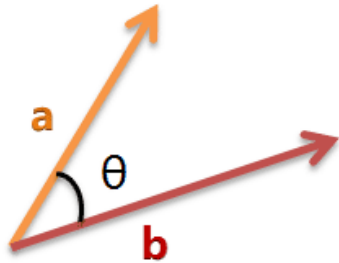
句子A：我 1，喜欢 2，看 2，电视 1，电影 1，不 1，也 0。

句子B：我 1，喜欢 2，看 2，电视 1，电影 1，不 2，也 1。

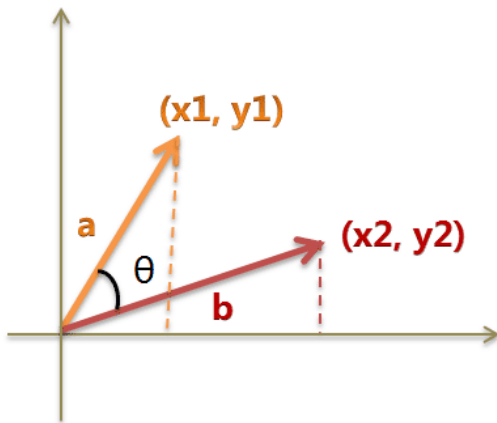
句子A：[1, 2, 2, 1, 1, 1, 0]

句子B：[1, 2, 2, 1, 1, 2, 1]

計算 Cosine Distance



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



$$\cos\theta = \frac{x_1x_2 + y_1y_2}{\sqrt{x_1^2 + y_1^2} \times \sqrt{x_2^2 + y_2^2}}$$

計算Cosine Similarity

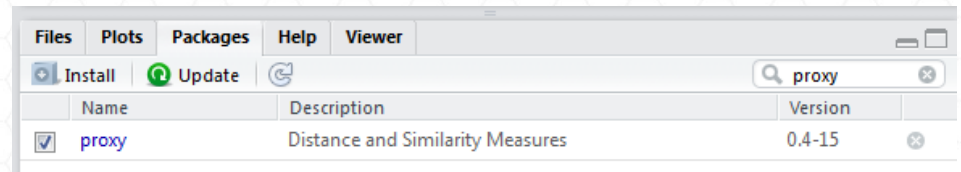
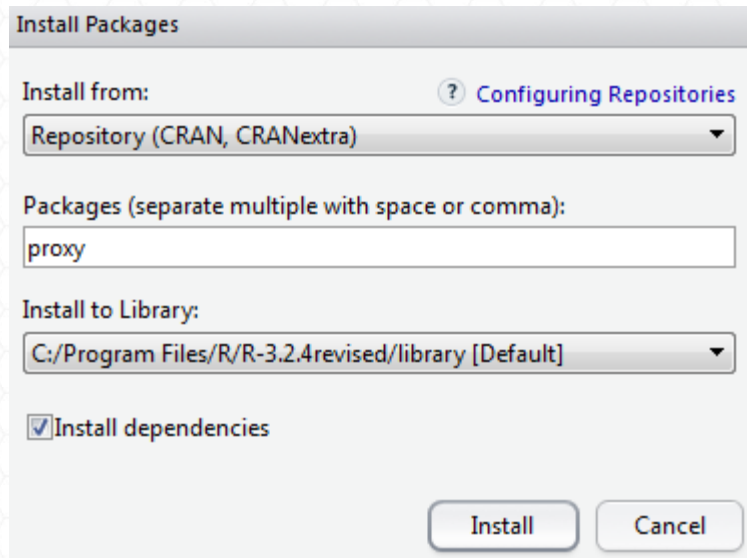
$$\begin{aligned}\cos\theta &= \frac{\sum_{i=1}^n (A_i \times B_i)}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}} \\ &= \frac{A \cdot B}{|A| \times |B|}\end{aligned}$$

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

$$\begin{aligned}\cos\theta &= \frac{1 \times 1 + 2 \times 2 + 2 \times 2 + 1 \times 1 + 1 \times 1 + 1 \times 2 + 0 \times 1}{\sqrt{1^2 + 2^2 + 2^2 + 1^2 + 1^2 + 1^2 + 0^2} \times \sqrt{1^2 + 2^2 + 2^2 + 1^2 + 1^2 + 2^2 + 1^2}} \\ &= \frac{13}{\sqrt{12} \times \sqrt{16}} \\ &= 0.938\end{aligned}$$

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」譴責補獸缺"
- ## [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】肯亞強遭台嫌回陸 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台人遭中 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] "喬丹大方祝賀勇士打破公牛的紀錄 "
```

```
## [8] "NBA癡狂夜日本也有感 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"
```


The background features a light blue hexagonal grid pattern. Overlaid on this is a large, faint, light blue circular graphic composed of concentric rings and radial lines, resembling a stylized spiral or a target. A solid dark blue horizontal bar runs across the top of the image, and a similar but slightly textured dark blue bar runs across the bottom.

THANK YOU