

數據分析師假日精修班

Lab7

David Chiu
2016/08/14

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

分類方法簡介

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

■ 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

如何分類鳶尾花 (iris)

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



Iris setosa



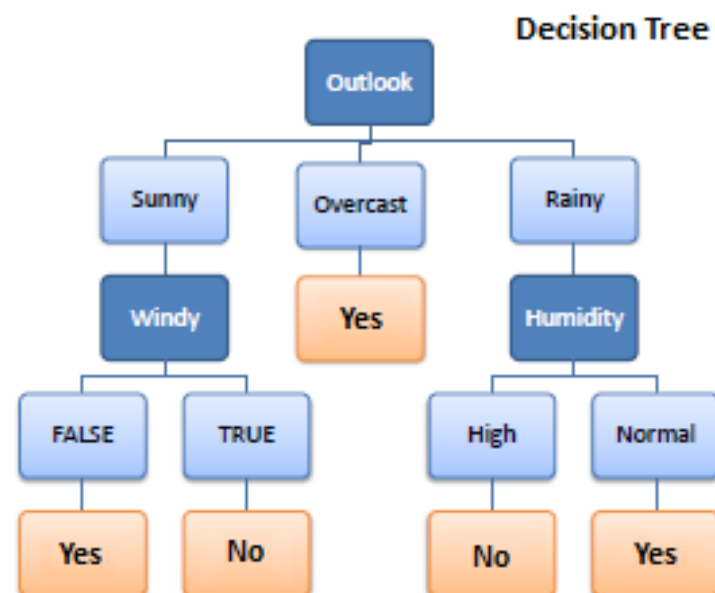
Iris versicolor



Iris virginica

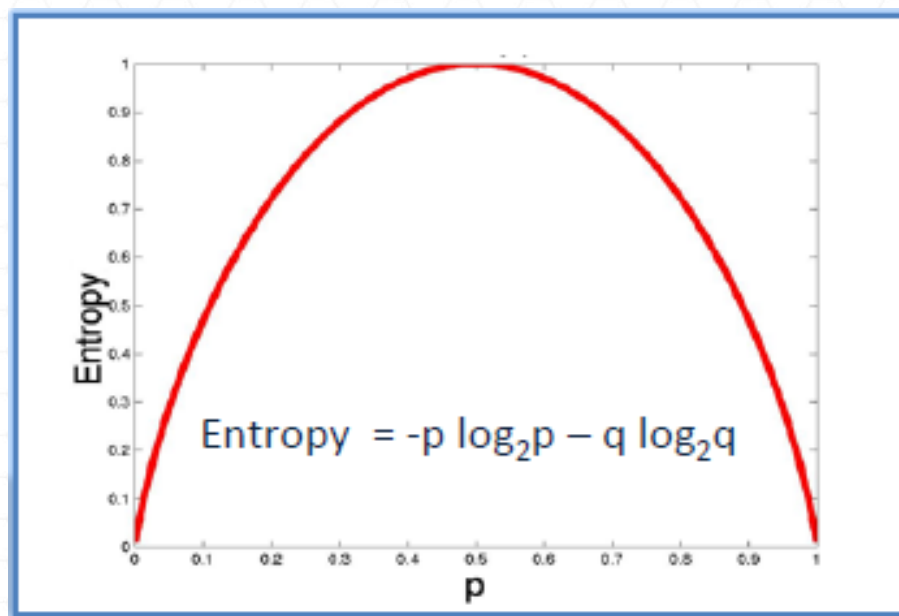
決策樹

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			

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

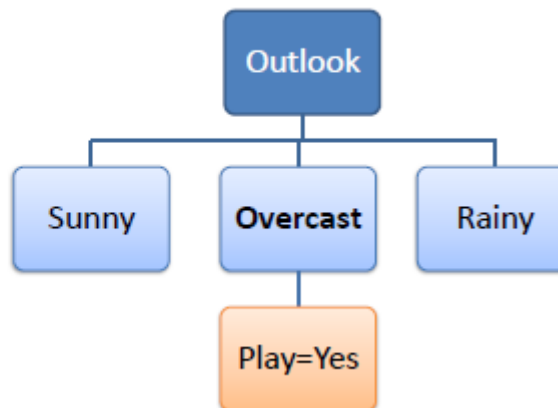
$$\begin{aligned} G(\text{PlayGolf}, \text{Outlook}) &= E(\text{PlayGolf}) - E(\text{PlayGolf}, \text{Outlook}) \\ &= 0.940 - 0.693 = 0.247 \end{aligned}$$

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			

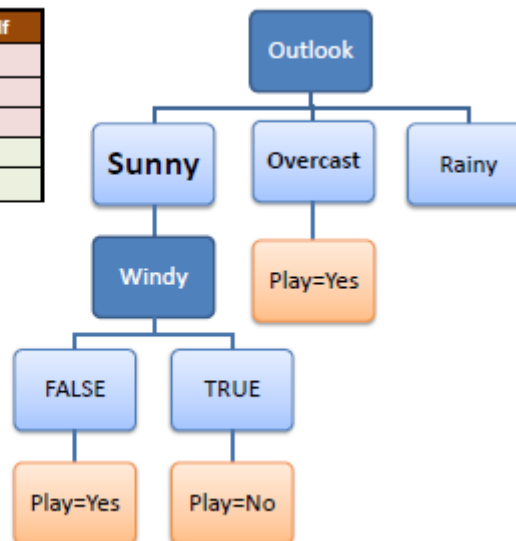
選擇子節點與分割節點

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

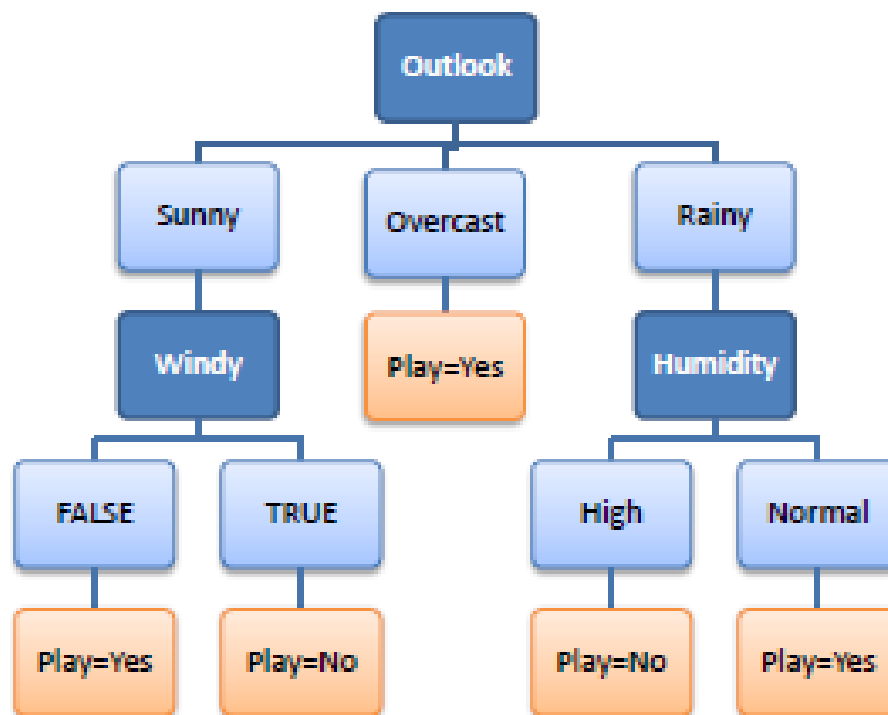
R_1 : IF (Outlook=Sunny) AND (Windy=FALSE) THEN Play=Yes

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

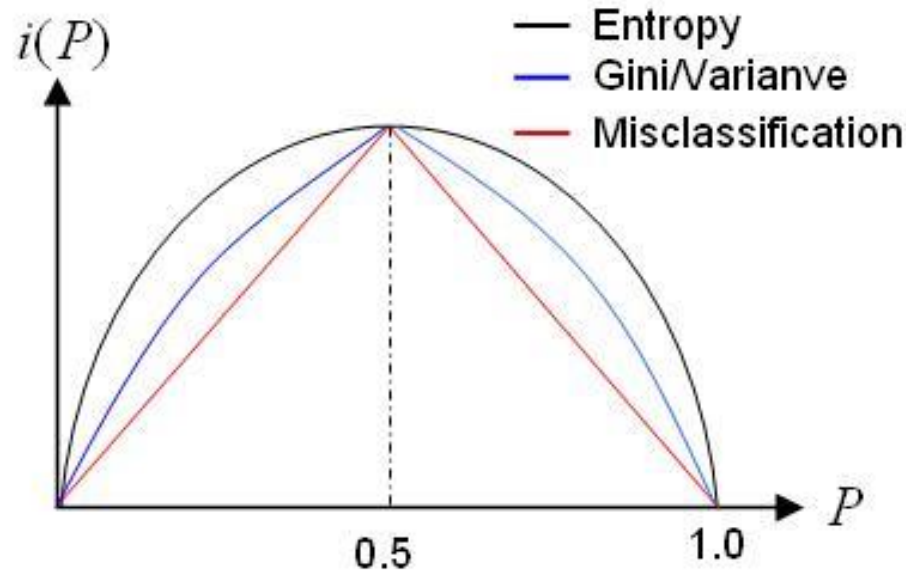


rpart 與遞迴分割法

■ 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

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)
```

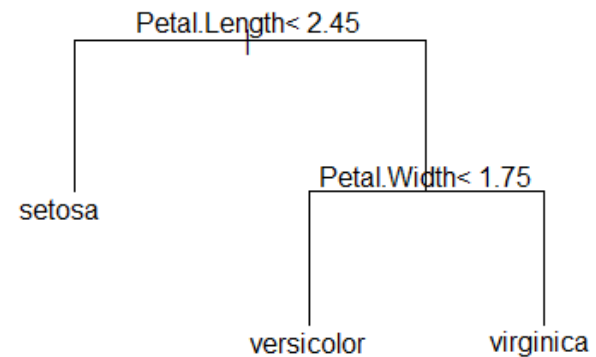
```
data(iris)
```

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

```
summary(fit)
```

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

```
text(fit)
```

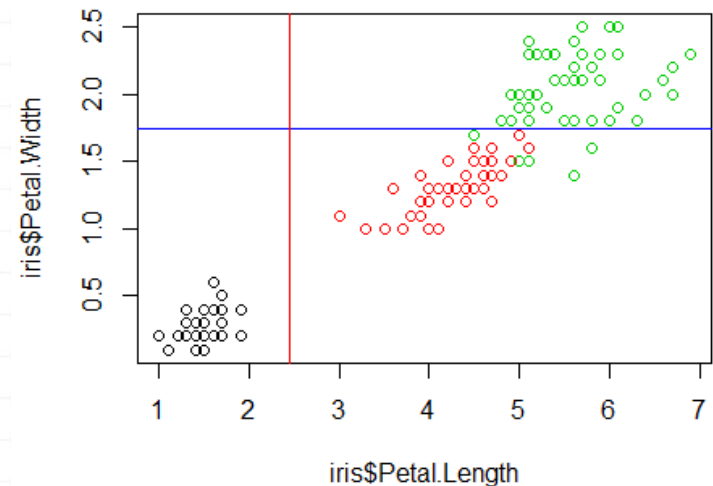


將分類結果顯示在圖上

```
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

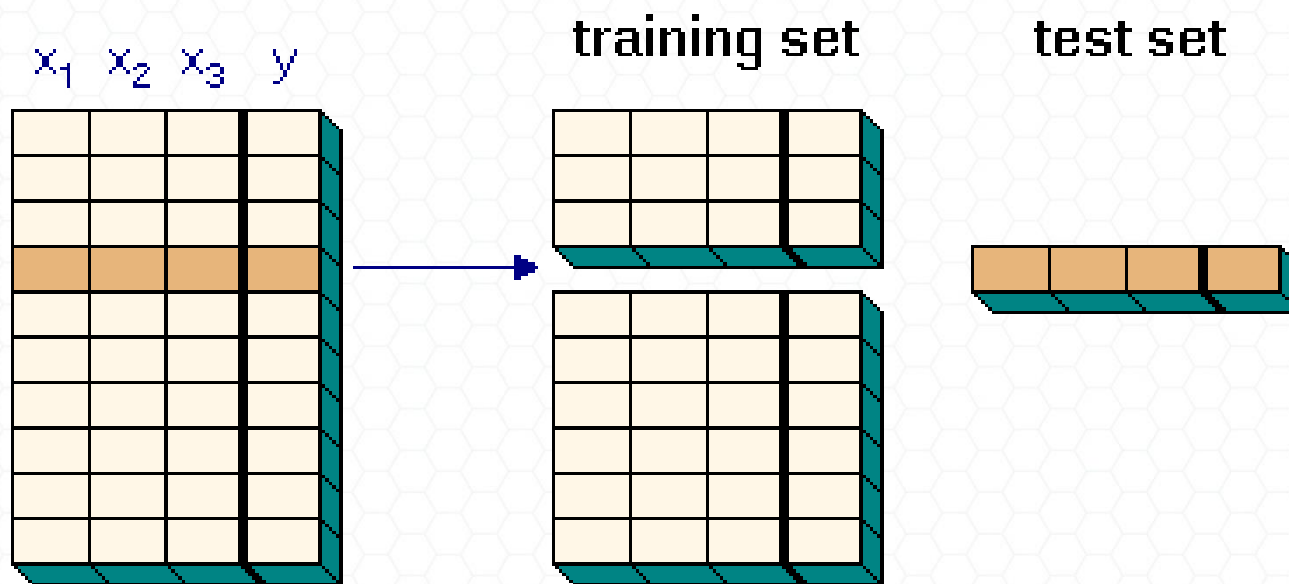
要驗證是否有過度學習?

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, ]
```

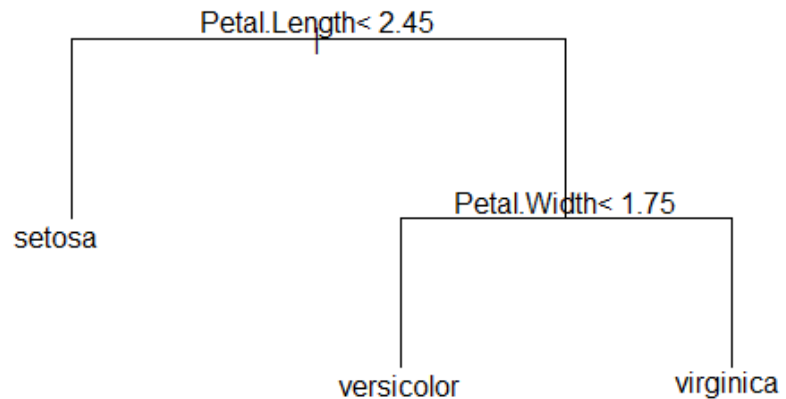
```
testset <- iris[idx==2, ]
```

```
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

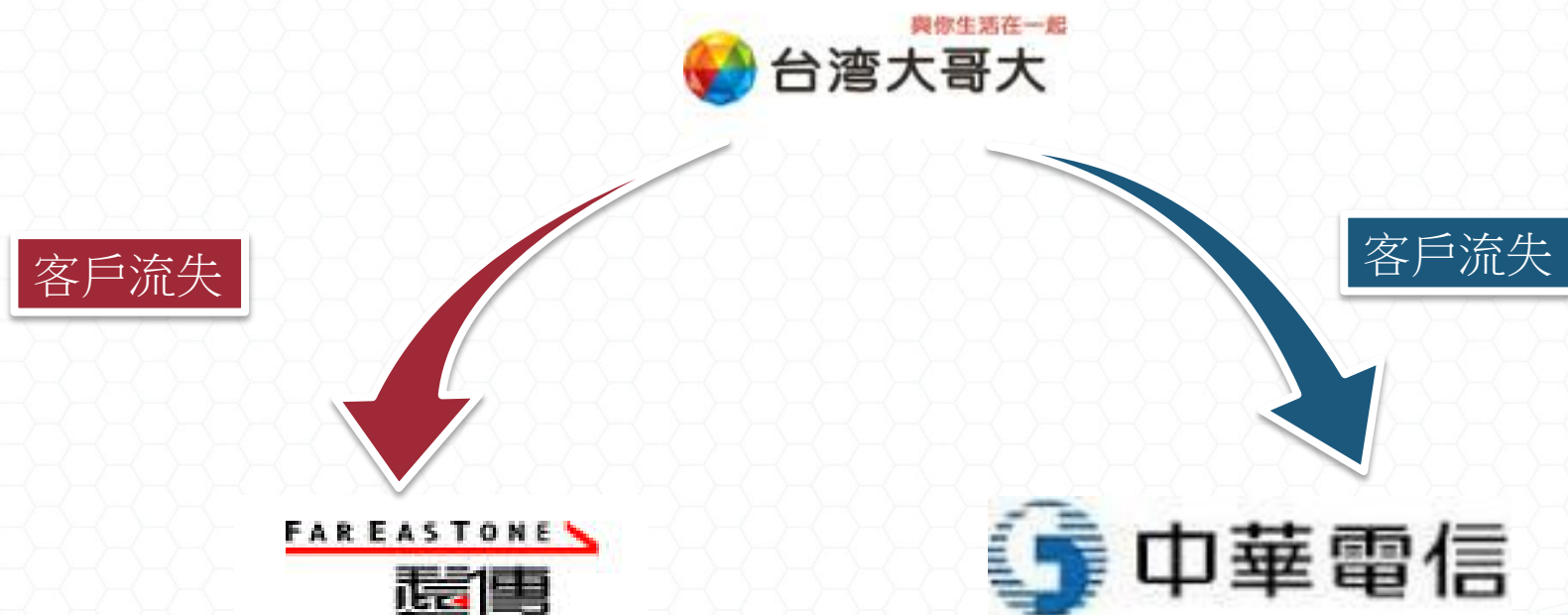
```
pred <- predict(fit2, testset[,-5], type= "class")  
cm <- table(pred, testset[,5])  
confusionMatrix(cm)
```

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)

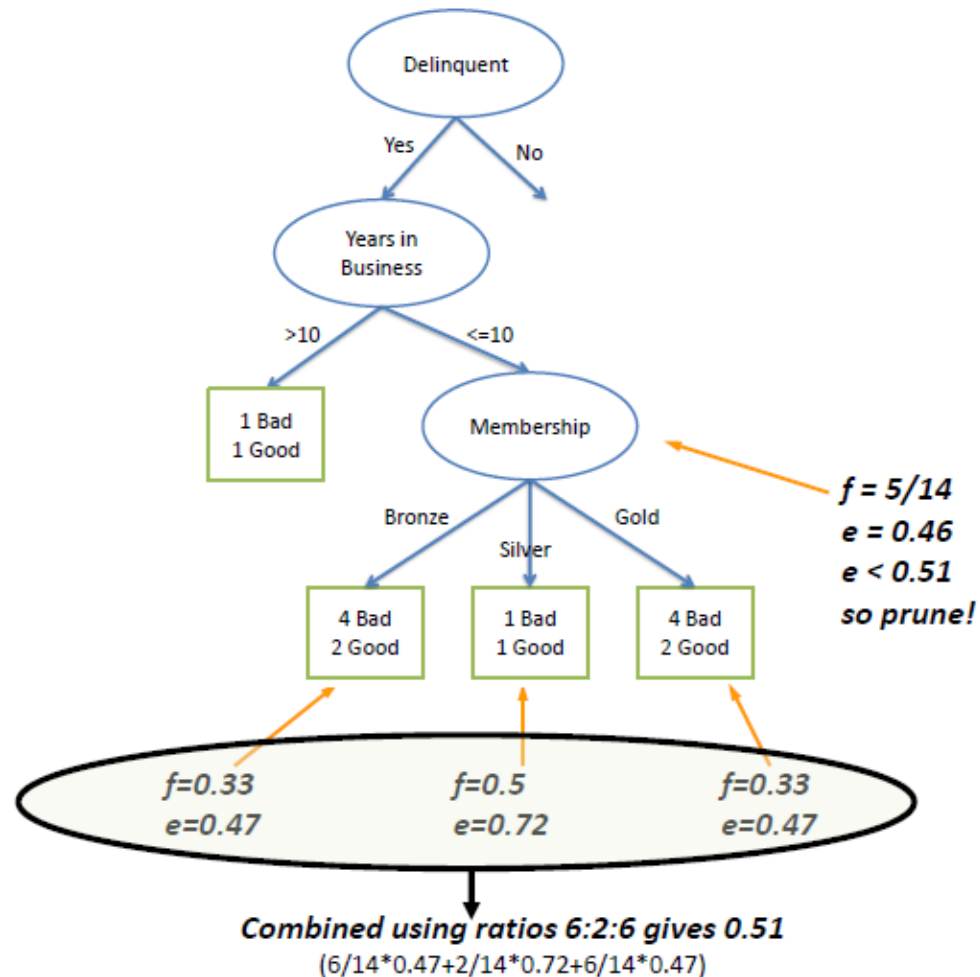
```
churn.rp <- rpart(churn ~ ., data=trainset)
plot(churn.rp, margin= 0.1)
text(churn.rp, all=TRUE, use.n = TRUE)
```

```
churn.rp <- rpart(churn ~ ., data=trainset)
plot(churn.rp, margin= 0.1)
text(churn.rp, all=TRUE, use.n = TRUE)
```



避免過度學習

- 預先剪枝(Pre-pruning)
：設定條件，當條件到達時，樹就停止生長
- 後剪枝(Post-pruning)
：等樹發展完全以後，再行剪枝



進行剪枝 (pruning)

```
min(churn.rp$cpstable[, "xerror"])
which.min(churn.rp$cpstable[, "xerror"])
churn.cp = churn.rp$cpstable[7, "CP"]
prune.tree = prune(churn.rp, cp = churn.cp)
plot(prune.tree, margin = 0.1)
text(prune.tree, all = TRUE, use.n = TRUE)
```


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

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

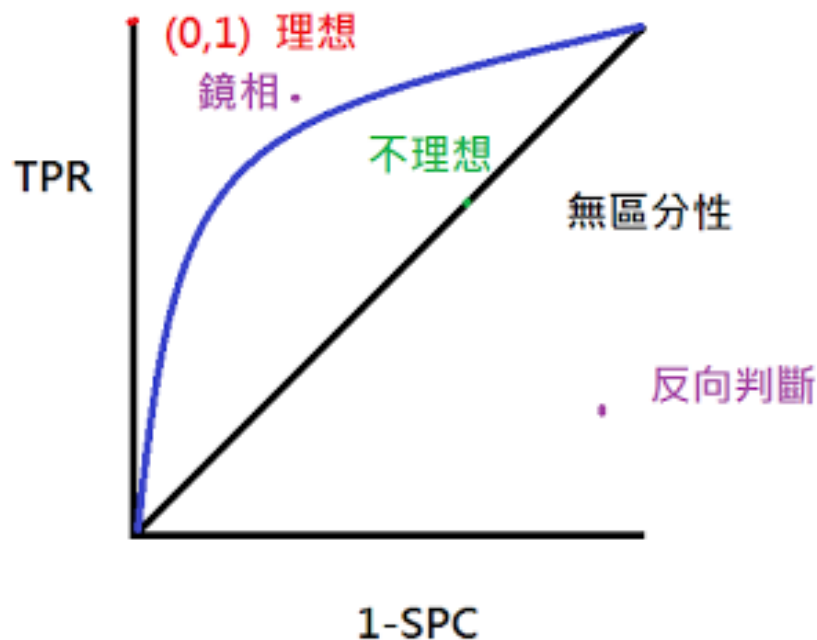
$1-SPC=FPR$

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

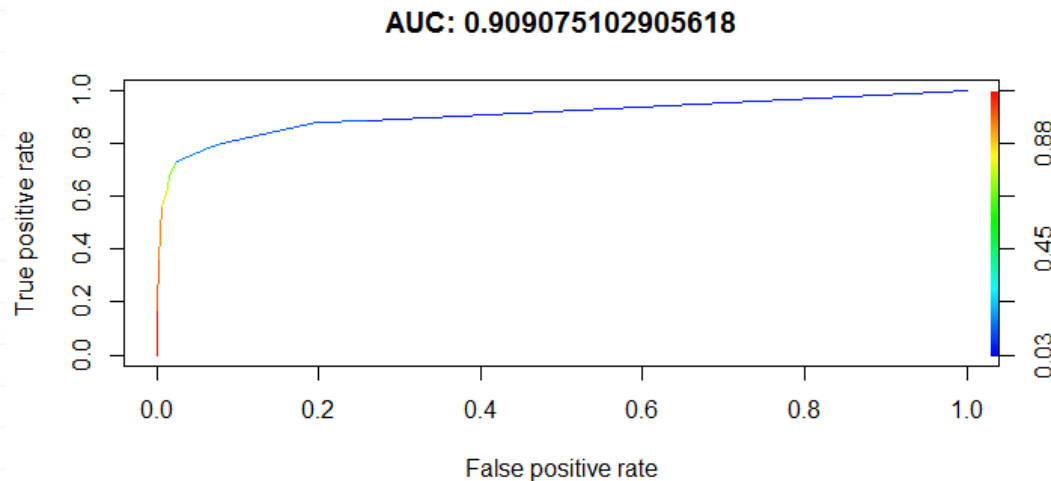
■ 接收者操作特徵(receiver operating characteristic, ROC curve)

- 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

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



文章分類

文章分類步驟

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

人怎麼分類新聞？



【狗仔偷拍】
陳柏霖「分
手」宋智孝
事實竟是...

【動新聞】
接見外賓唸
稿卡住 蔡
英文：稿子
借你唸唸看

下列三篇新聞該怎麼分類？

- 鴻海收購夏普正式簽約郭台銘：全球高科技產業最棒的一天
- 嘉玲採果郭台銘美人柑到手
- 憶起「馬習會」 郭台銘爆氣飆罵：Stupid！

使用Naïve Bayes 分類器

■ Bayes 分類器源自Bayes 理論



Drew Barrymore



Drew Carey

What is the probability of being called
“drew” given that you are a **male**?

What is the probability
of being a **male**?

$$p(\text{male} | \text{drew}) = \frac{p(\text{drew} | \text{male}) p(\text{male})}{p(\text{drew})}$$

What is the probability of
being named “drew”?

(actually irrelevant, since it is

Naïve Bayes 分類器

- 假設每個特徵(Feature)都為獨立

$$p(d|c_j) = p(d_1|c_j) * p(d_2|c_j) * \dots * p(d_n|c_j)$$

根據Feature d 所產生類別c 的機率

在文章中每個詞可以視為獨立
因此使用貝氏分類法即可以做文章分類

Naïve Bayes 分類器 (續)

■ 假設用Naïve Bayes 預測性別

$$p(\text{officer drew}|c_j) = p(\text{over_170}_{\text{cm}} = \text{yes}|c_j) * p(\text{eye} = \text{blue}|c_j) * \dots$$



Officer Drew
is blue-eyed,
over 170_{cm}
tall, and has
long hair

$$p(\text{officer drew} | \text{Female}) = 2/5 * 3/5 * \dots$$

$$p(\text{officer drew} | \text{Male}) = 2/3 * 2/3 * \dots$$

新聞分類問題

1. 篩選分類新聞
2. 建立詞頻矩陣
3. 篩選詞頻矩陣
4. 分為訓練與測試資料集
5. 使用訓練資料集建立模型
6. 使用建立模型預測測試資料集的準確度

對娛樂與財經新聞斷詞

```
download.file('https://github.com/ywchiu/rtibame/raw/master/appliedaily2.RData', destfile="appliedaily2.RData")
load("appliedaily2.RData")
apple.subset = appliedaily[appliedaily$category %in% c('財經', '娛樂'),]
library(jiebaR)
mixseg = worker()
apple.seg =lapply(apple.subset$content,
function(e)segment(code=e, jiebar=mixseg))
```

產生詞頻矩陣

```
library(tm)
jieba_tokenizer=function(d){
  unlist(segment(d[[1]],mixseg))
}
space_tokenizer=function(x){
  unlist(strsplit(as.character(x[[1]]),'[:space:])+'))
}
doc=VCorpus(VectorSource(apple.seg))
doc=unlist(tm_map(doc,jieba_tokenizer),recursive=F)
doc=lapply(doc,function(d)paste(d,collapse=' '))
control.list=list(wordLengths=c(2,Inf),tokenize=space_tokenizer)
dtm=DocumentTermMatrix(Corpus(VectorSource(doc)),control=control.list)
dim(dtm)
```

會產生相當高維的矩陣

挑選詞頻大於五的

```
ft <- findFreqTerms(dtm, 5)
```

```
control.list=list(wordLengths=c(2,Inf),tokenize=space_tokenizer,dictionary =ft)
```

```
new.dtm=DocumentTermMatrix(Corpus(VectorSource(doc)),control=control.list)
```

建立新的詞頻矩陣

只列出是否有對到該詞的

```
convert_counts <- function(x) {
```

```
  x <- ifelse(x > 0, 1, 0)
```

```
  x <- factor(x, levels = c(0, 1), labels = c("No",  
  "Yes"))
```

```
  return(x)
```

```
}
```

有出現該詞列為yes/no

```
dtm.count <- apply(new.dtm, MARGIN = 2,  
convert_counts)
```


將資料列為訓練跟測試資料集

```
library(e1071)
m <- as.data.frame(dtm.count)
idx <- sample.int(2, nrow(m), replace=TRUE,
prob=c(0.7,0.3))
trainset <- m[idx==1,]
testset <- m[idx==2,]
traintag <- apple.subset[idx==1,"category"]
testtag <- apple.subset[idx==2,"category"]
```

建立confusionMatrix

```
model <- naiveBayes(trainset,as.factor(traintag) )  
pred <- predict(model, testset)  
tb <- table(pred, testtag)
```

```
library(caret)  
confusionMatrix(tb)
```

Accuracy : 0.9851

	testtag	
pred	娛樂	財經
娛樂	35	1
財經	0	31

聯絡方式

- Website:

- ywchiu.com

- Email:

- david@largitdata.com

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

分類技巧補充

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

如何找出最重要的變數

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

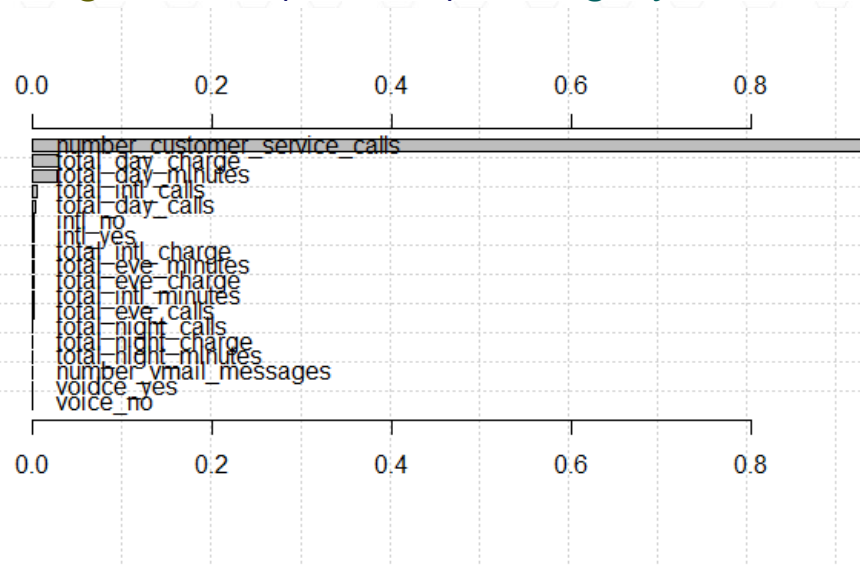
```
library(rminer)
```

```
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)
```

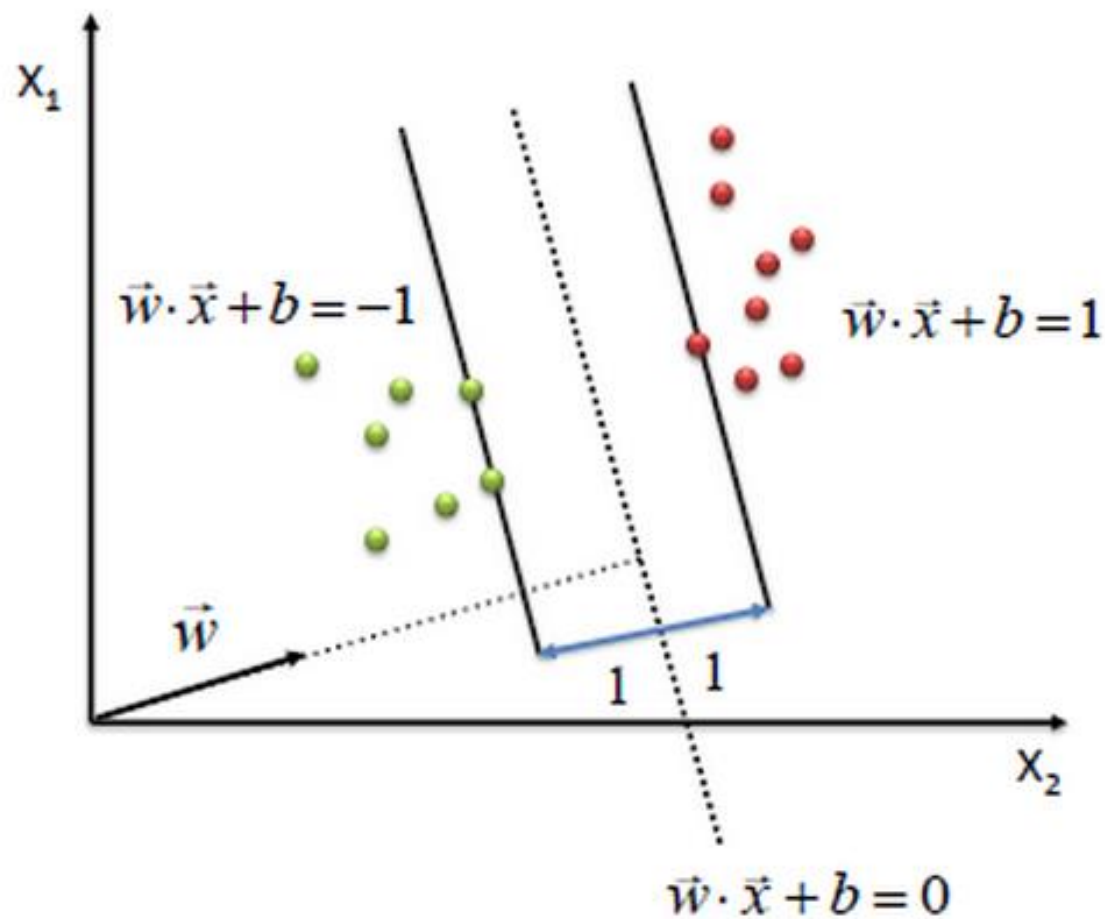


Ctree 與條件推斷決策樹

■ 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。

支持向量機



$$\max \frac{2}{\|\vec{w}\|}$$

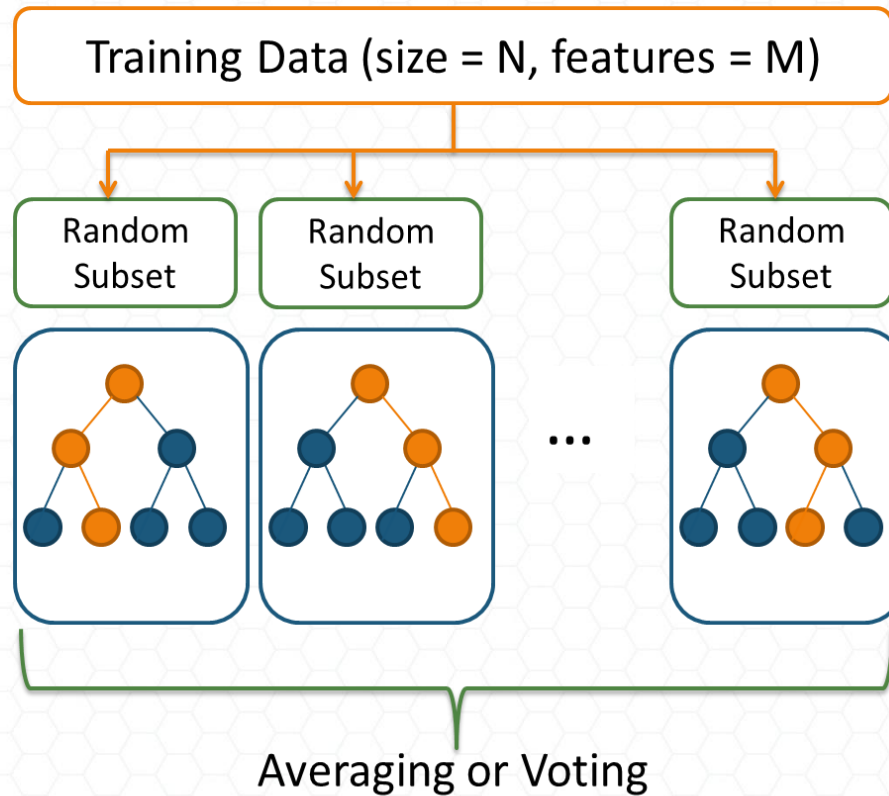
s.t.

$$(\vec{w} \cdot \vec{x} + b) \geq 1, \forall \vec{x} \text{ of class 1}$$

$$(\vec{w} \cdot \vec{x} + b) \leq -1, \forall \vec{x} \text{ of class 2}$$

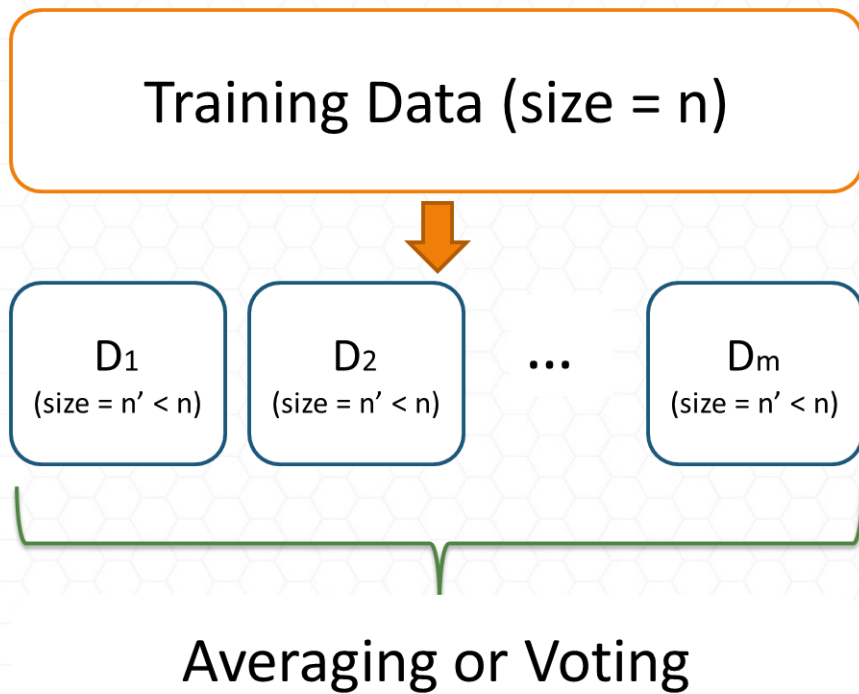
隨機森林 (Random Forest)

■ N 多少樹, M 多少個特徵

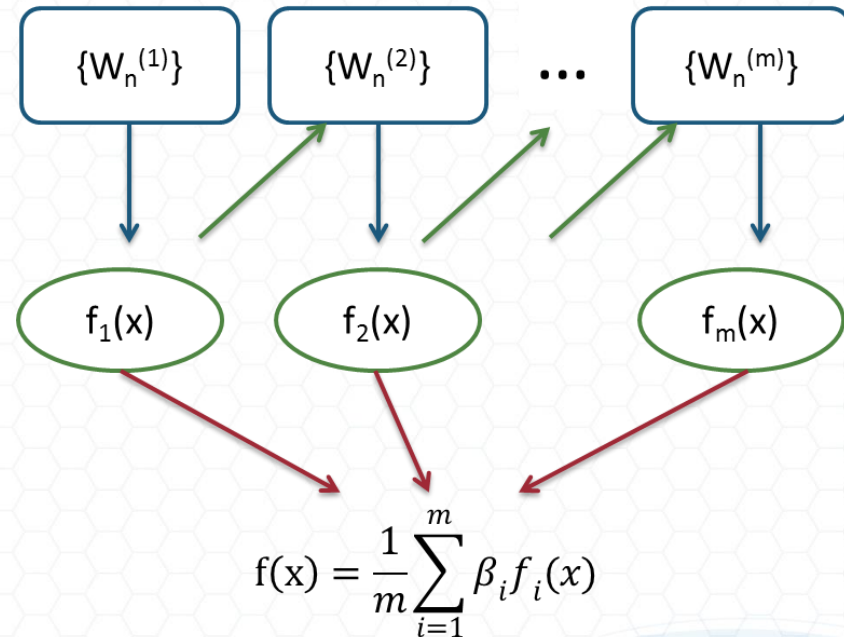


Bagging & Boosting

Bagging



Boosting



Naïve Bayes

The diagram shows the Naïve Bayes formula with arrows pointing from descriptive labels to the corresponding parts of the equation:

$$P(c | x) = \frac{P(x | c) P(c)}{P(x)}$$

Labels and their corresponding terms in the formula:

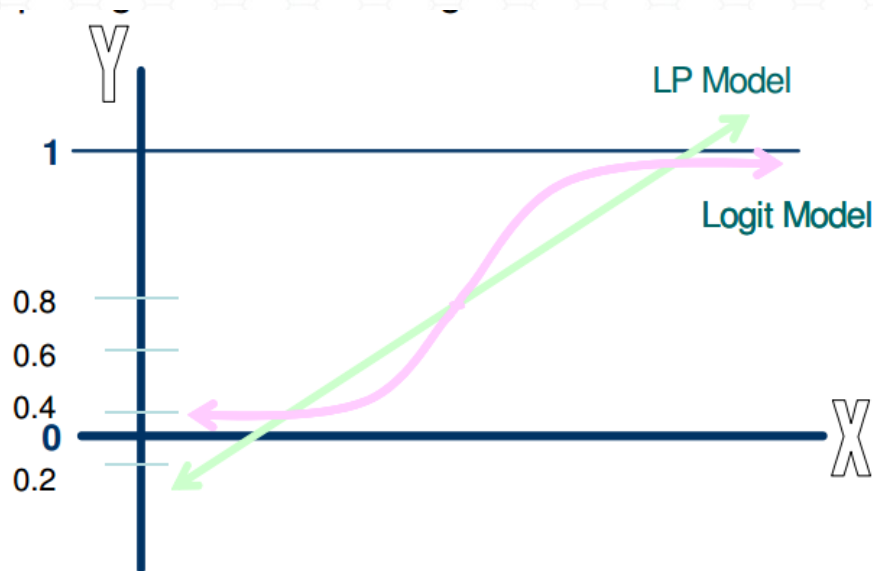
- Likelihood** points to $P(x | c)$
- Class Prior Probability** points to $P(c)$
- Posterior Probability** points to $P(c | x)$
- Predictor Prior Probability** points to $P(x)$

$$P(c | X) = P(x_1 | c) \times P(x_2 | c) \times \cdots \times P(x_n | c) \times P(c)$$

邏輯迴歸分析 (Logistic Regression)

■ 從對連續依變數的預測轉變為二元的結果(是/否)

- ▣ 客戶是否流失?
- ▣ 客戶是否買單?
- ▣ 腫瘤為良性還惡性?



如果是線性迴歸
會不會X值越大會得到
>100% 的預測結果?

邏輯迴歸分析 (Logistic Regression)

Logistic Regression

■ 定義

$$\log it(y) = \ln(odds) = b_0 + b_1 X_1 + \varepsilon$$

■ Odds

□ Odds

= Probability of event for success (PE)/ failure

= PE/(1-PE)

■ 推導

$$e^{\ln(odds)} = odds = e^{(b_0 + b_1 X_1 + \varepsilon_i)}$$

$$PE = odds/(1+Odds) = e^{(b_0 + b_1 X_1 + \varepsilon_i)} * \frac{1}{1 + e^{(b_0 + b_1 X_1 + \varepsilon_i)}}$$

單純代表獲勝/失敗的機率