

## 2.1 Logistic Regression



“

## 是與否的選擇

假設你是影音串流平台的數據工程師  
客戶會不會從免費觀看變成訂閱觀看呢？

什麼因素會影響他會不會付錢？

—••—

# Abstract

1. 是與否的選擇
2. 邏輯迴歸模型原理
3. 衡量分類表現的方法



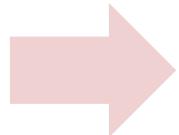
# 是與否的選擇

要不要買Mac呢？

當工程師當然要用Mac，買！

年終下來了，\$ \$夠多，買！

有台Mac才潮，買



# 分類問題：影響因素(X)與是/否(Y)的關係

X

工作是否需要？

- 1, 當工程師當然要用Mac
- 0, 當打工旅遊族，不需要

預算充足嗎？

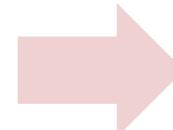
年終作為購買預算

夠不夠潮

- 1, 潮
- 0, 不care

Y

- 1, 要買Mac
- 0, 不要買Mac



# 透過條件機率影響 X 與 Y ( 是/否 )

工程師



買Mac的條件機率為：

$$P(Y_1=1 \mid X_{11}=1, X_{12}=300,000, X_{13}=0)$$

(工作需要)

(年終)

(不潮)

打工旅遊族



$$P(Y_1=0 \mid X_{21}=0, X_{22}=0, X_{23}=1)$$

(工作不需要)

(沒年終)

(潮)

# 透過大量資料分析所有因素如何影響是否決策

	工作是否需要 ( $X_1$ )	年終多少 ( $X_2$ )	是否夠潮 ( $X_3$ )	是否購買 ( $Y$ )
1	1	300,000	1	1
2	1	300,000	1	0
3	0	0	1	0
4	0	50,000	1	0
5	1	50,000	1	1
6	1	100,000	1	1
7	1	100,000	0	0
.....				

完全相同的情境  
可能會有不同結果

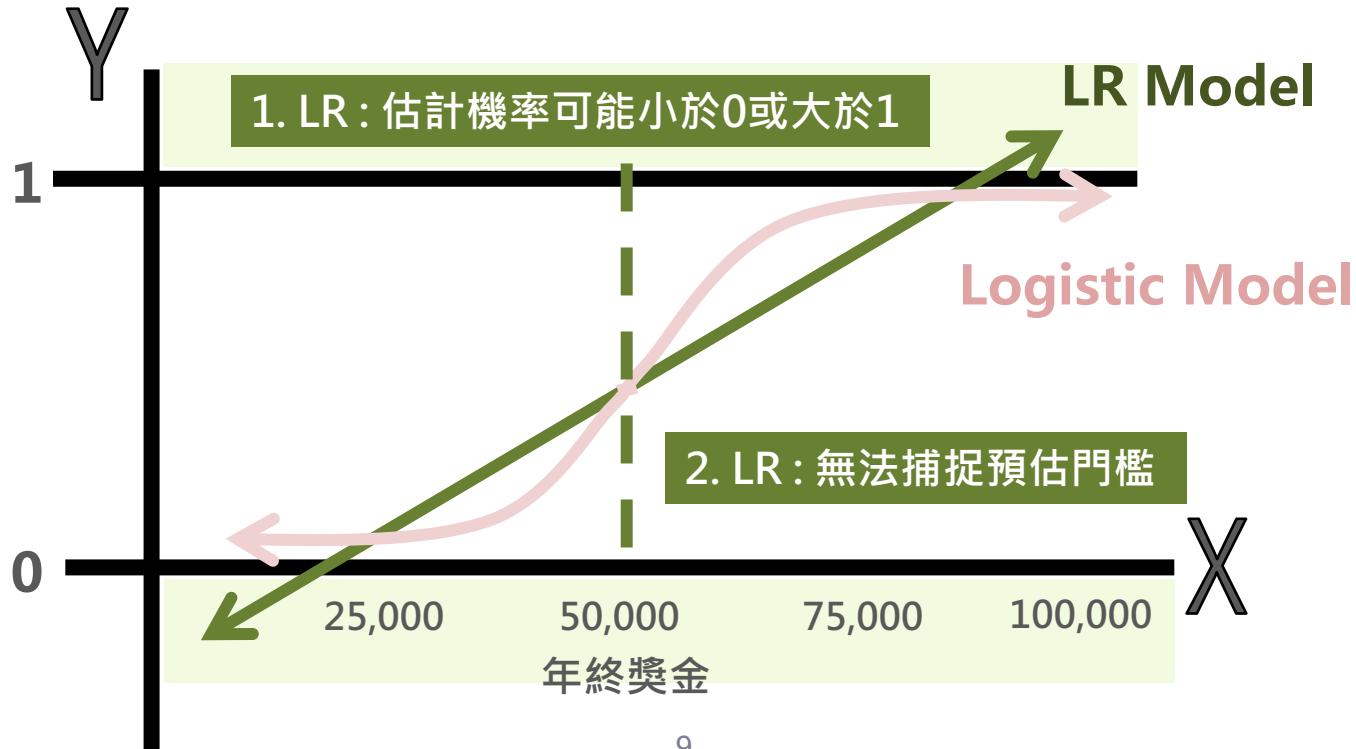
- 4,5比較  $X_1$
- 5,6比較  $X_2$
- 6,7比較  $X_3$
- 所有人比較所有因素的強弱

“

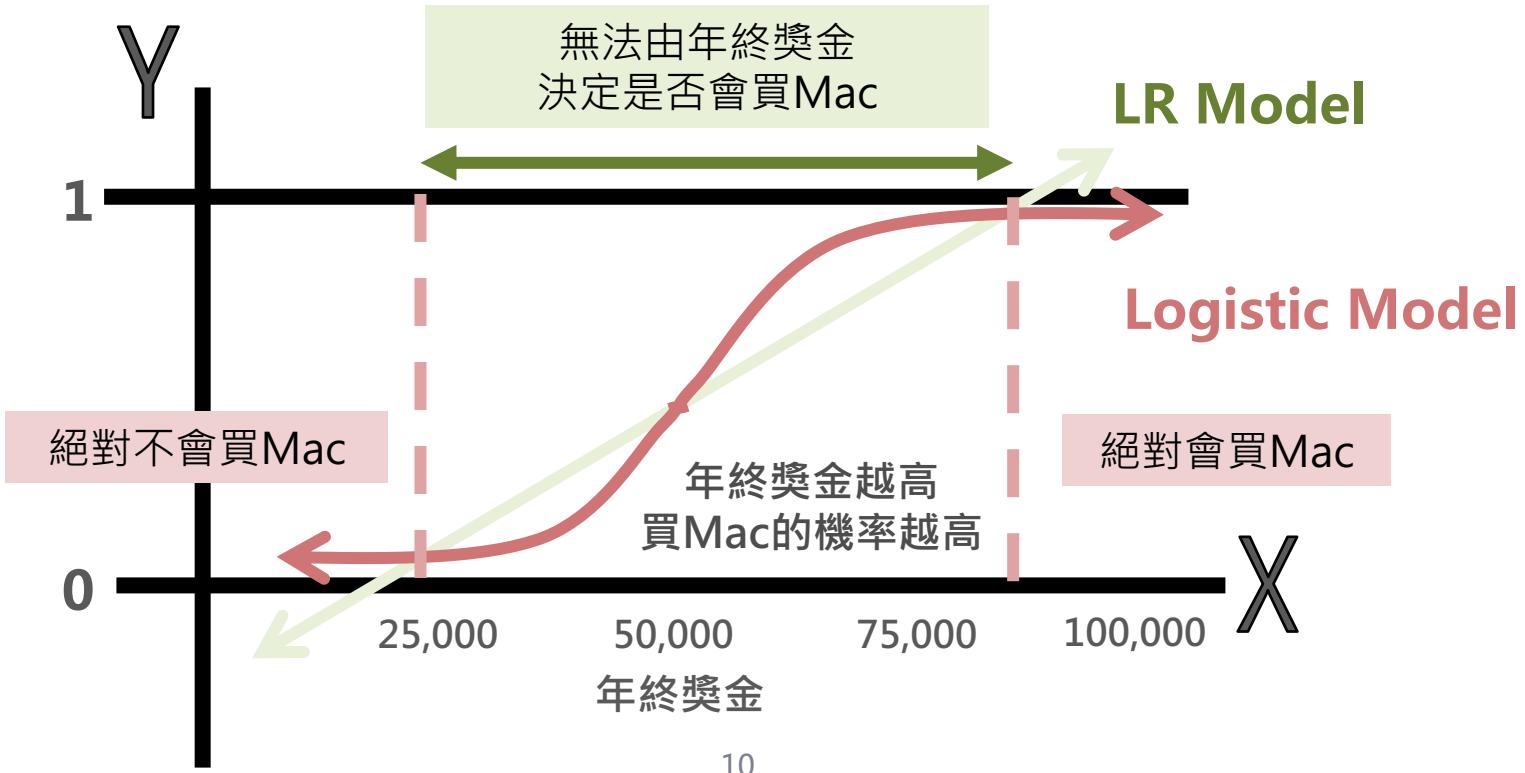
# 邏輯迴歸模型原理

—••—

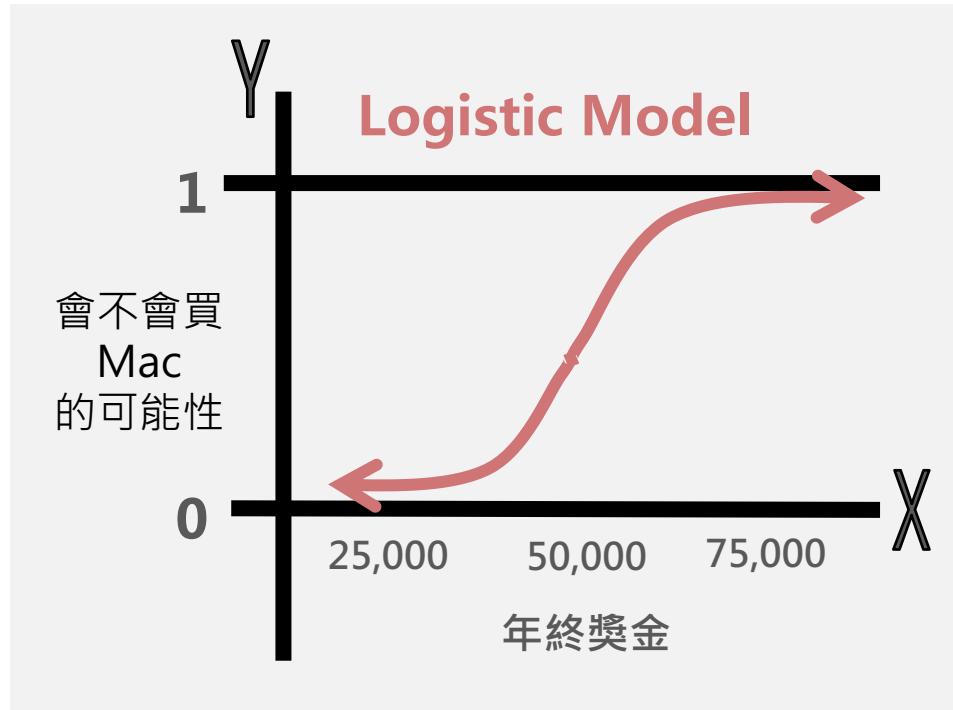
# Comparing LR and Logistic Models



# Comparing LR and Logistic Models



# 使用sigmoid函數解釋X與條件機率的關係



條件機率

$$P(Y = 1|X) = \frac{\exp(\beta_0 + \beta_1 X)}{1 + \exp(\beta_0 + \beta_1 X)}$$

sigmoid函數

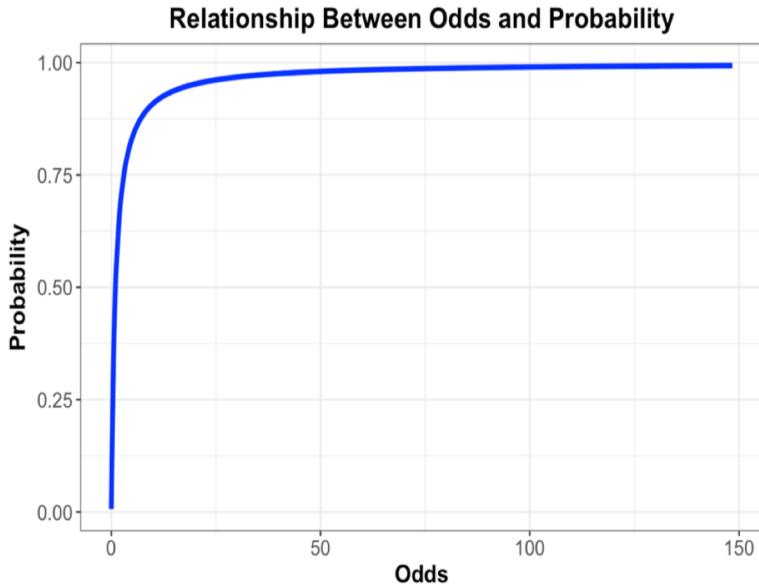
$$\log \left( \frac{P(Y = 1|X)}{1 - P(Y = 1|X)} \right) = \beta_0 + \beta_1 X$$

線性迴歸

對數勝率  
Log Odds

# 勝率(Odds)代表<成功與失敗機率>的比值

勝率 = 成功機率 / 失敗機率

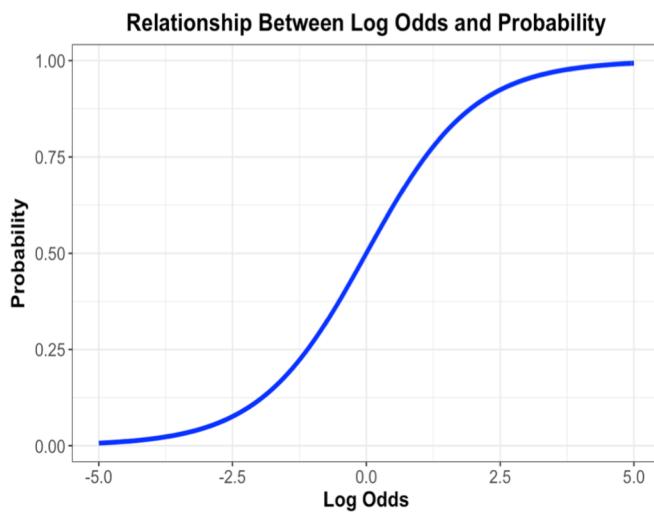


$$\text{Odds} = \frac{P(Y = 1|X)}{1 - P(Y = 1|X)}$$

勝率是5  
代表一個人買Mac和不買Mac  
的機率比為5:1

# 對數勝率(Log Odds / Logit)與機率呈現Sigmoid關係

Log Odds 與機率呈現Sigmoid關係



為什麼要取Log Odds(Logit)?

- $X$ 與條件機率呈現Sigmoid關係  
( 邏輯迴歸的基本假設 )
- 對數勝率與條件機率呈現Sigmoid關係  
( 對數勝率的基本性質 )

# 用「最大概似法」估計邏輯迴歸的參數

	年終 (X)	是否購買 (Y)
1	300,000	1
2	20,000	0
3	0	0
...	...	...
n	X <sub>n</sub>	Y <sub>n</sub>



$$p_1 \equiv P(Y = 1|X_1) = \frac{\exp(\beta_0 + \beta_1 X_1)}{1 + \exp(\beta_0 + \beta_1 X_1)}$$

若  $Y_1 = 1$ ，則得到  $p_1$  分數；  
 若  $Y_1 = 0$ ，則得到  $1 - p_1$  分數。  
 該個體獲得分數 =  $(p_1)^{y_1}(1 - p_1)^{1 - y_1}$



$$p_n \equiv P(Y = 1|X_n) = \frac{\exp(\beta_0 + \beta_1 X_n)}{1 + \exp(\beta_0 + \beta_1 X_n)}$$

找到參數  $(\hat{\beta}_0, \hat{\beta}_1)$  最大大化概似函數

$$\mathcal{L}(\beta_0, \beta_1) = \prod_{i=1}^n p_i^{y_i} (1 - p_i)^{1 - y_i}$$

分數乘積

# 多元邏輯迴歸模型

## 變數和估計參數增加

$$P(Y = 1|X_1, \dots, X_p) = \frac{\exp(\beta_0 + \beta_1 X_1 + \dots + \beta_p X_p)}{1 + \exp(\beta_0 + \beta_1 X_1 + \dots + \beta_p X_p)}$$

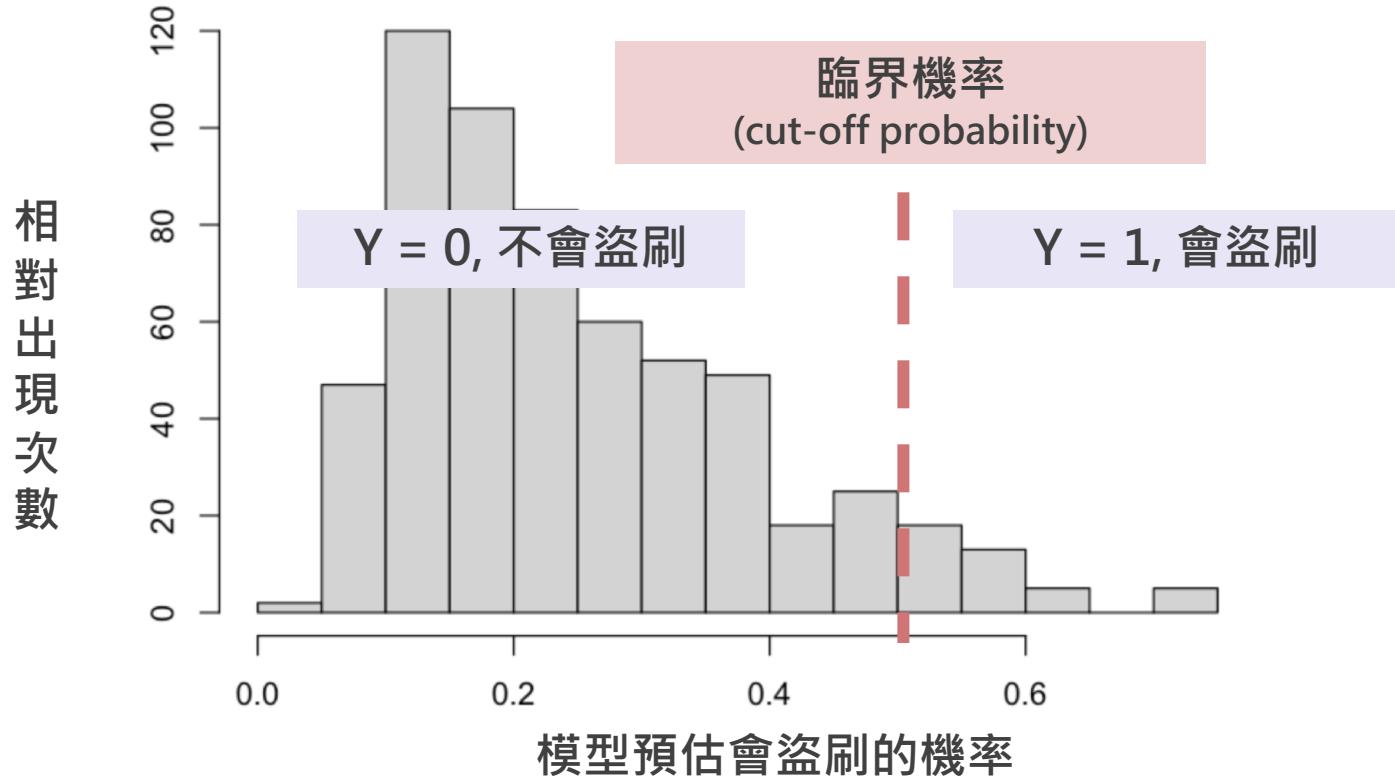
條件機率 **p+1** 個待估計參數

“

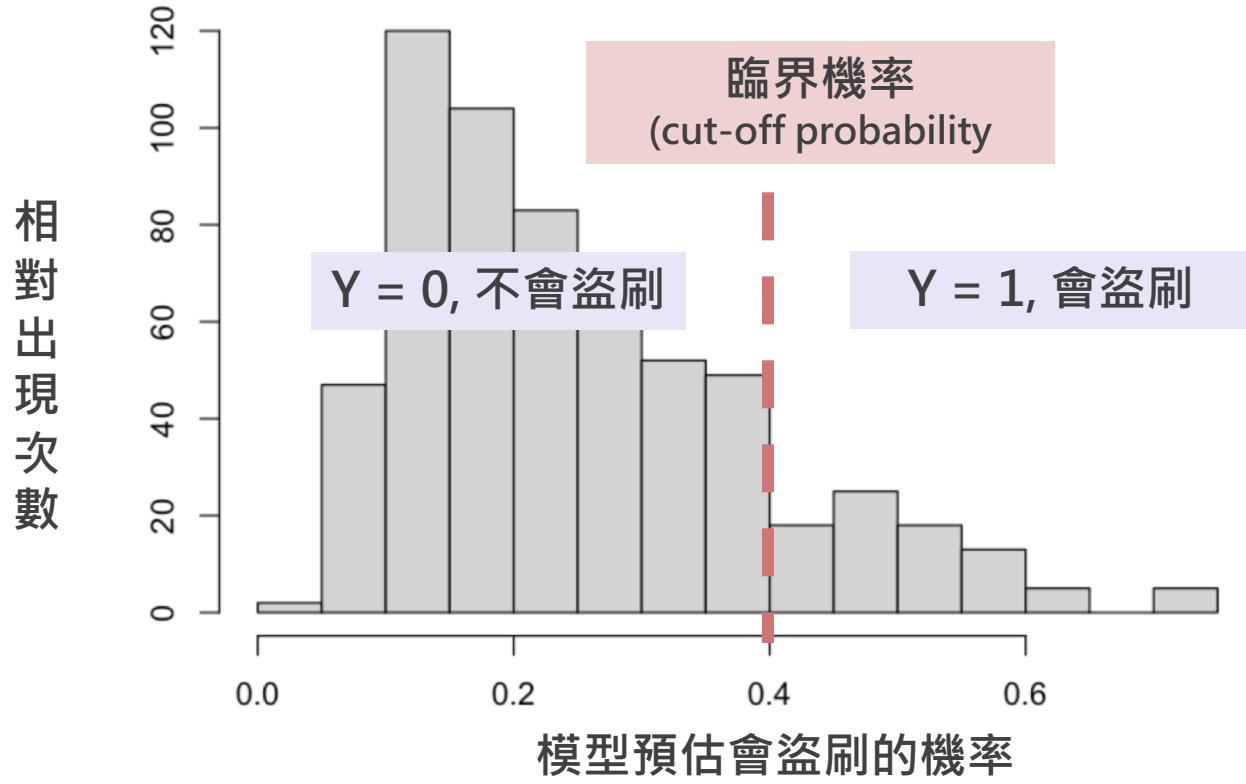
# 衡量分類表現的方法

—••—

# 設定臨界機率



## 調整臨界機率



# 模糊矩陣(Confusion Matrix)

		預測 NO	預測 YES
		沒盜刷	有盜刷
實際 NO	沒盜刷	TN (True Negative)	FP (False Positive) Type I Error
實際 YES	有盜刷	FN (False Negative) Type II Error	TP (True Positive)

# 模糊矩陣(Confusion Matrix) – 準確率(Accuracy)

## 衡量模型分類正確率

- 指標定義：  
所有個體中，有多少比率的個體被分類正確
- 計算公式：  
$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{FP} + \text{TN} + \text{FN})$$



	預測 NO	預測 YES
實際 NO	TN	FP
實際 YES	FN	TP

# 模糊矩陣(Confusion Matrix) – 精確度(Precision)

## 避免模型誤判個體為陽性

- 指標定義：  
被分類為陽性的個體，有多少比率是真陽性

- 計算公式：  
 $Precision = TP / (TP + FP)$

所有被預測為盜刷的人，真正盜刷的人  
寧可錯殺一千，不可放過一人

	預測 NO	預測 YES
實際 NO	TN	FP
實際 YES	FN	TP

# 模糊矩陣(Confusion Matrix)

## – 召回度/敏感度(Recall/Sensitivity)

### 避免錯過任何一個陽性個體

- 指標定義：

真正為陽性的個體中，有多少比率被預測是真陽性

- 計算公式：

$$\text{Recall/Sensitivity} = \text{TP} / (\text{TP} + \text{FN})$$

實際上盜刷的人裡，有多少被預測為真正被盜刷的人

	預測 NO	預測 YES
實際 NO	TN	FP
實際 YES	FN	TP

明明盜刷卻  
預測沒盜刷

# 模糊矩陣(Confusion Matrix) – 明確度 (Specificity)

能明確分辨出陰性個體

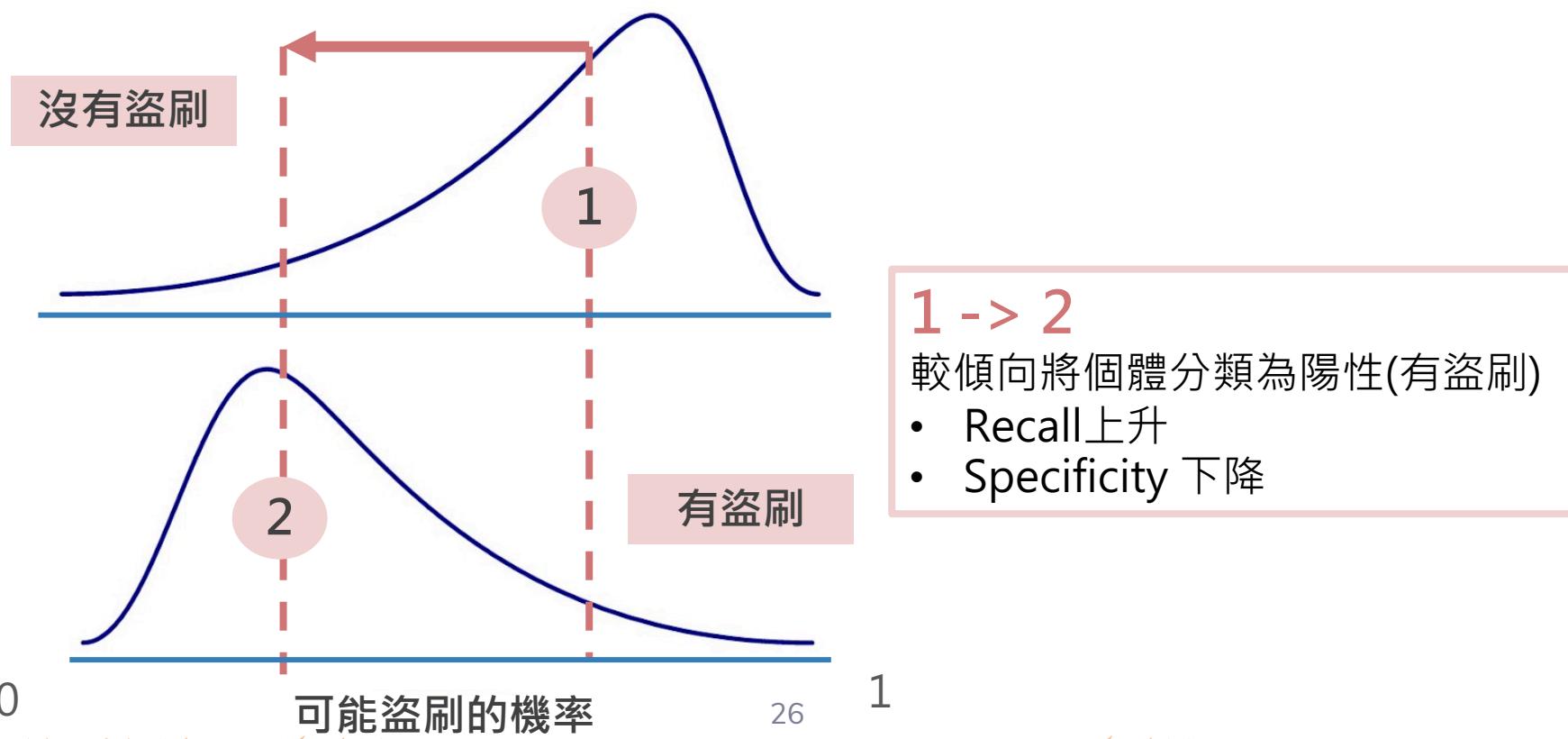
- 指標定義：  
被實際為陰性的個體中，有多少比率是真陰性

- 計算公式：  
 $\text{Specificity} = \text{TN} / (\text{TN} + \text{FP})$

在真實沒有盜刷的人裡，真的沒有盜刷的人

	預測 NO	預測 YES
實際 NO	TN	FP
實際 YES	FN	TP

# Recall 與 Specificity 存在抵換關係



# F1 Score

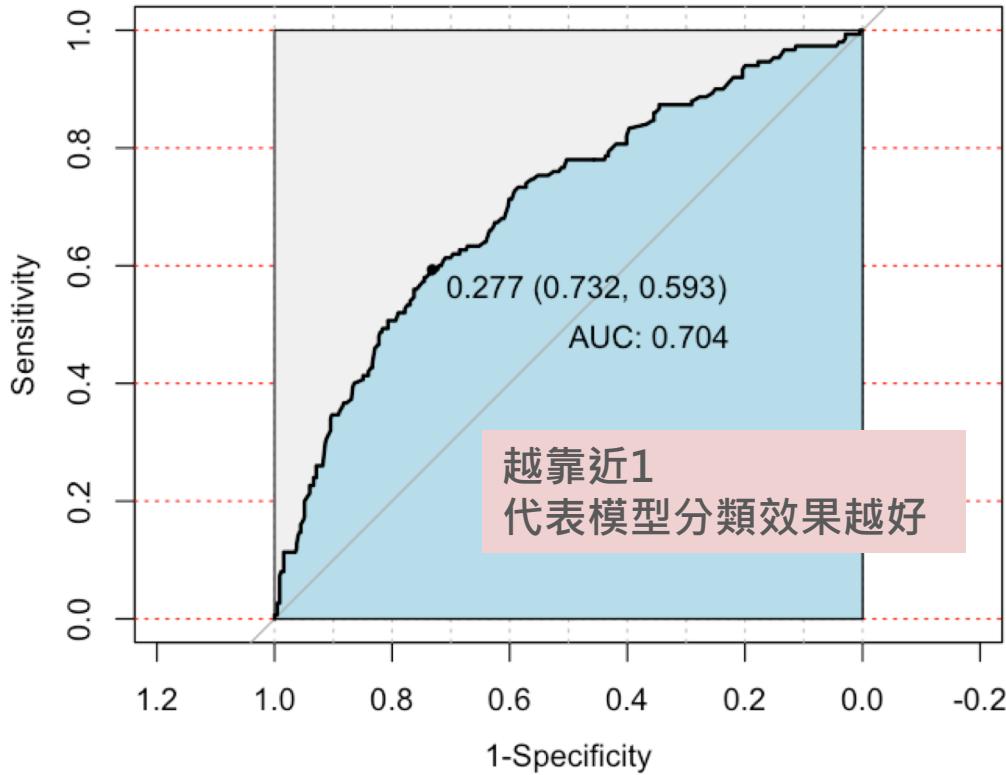
如果覺得Precision和Recall都同等重要  
可用F1 Score來代表

F1 Score的取值範圍從0到1的  
- 1代表模型的輸出最好  
- 0代表模型的輸出結果最差

$$\text{F1-score} = \frac{2 \times (\text{precision} \times \text{recall})}{(\text{precision} + \text{recall})}$$

# ROC曲線

## (Receiver Operating Characteristics)





# THANKS!