

# CH02

## 資料表示、處理及分析

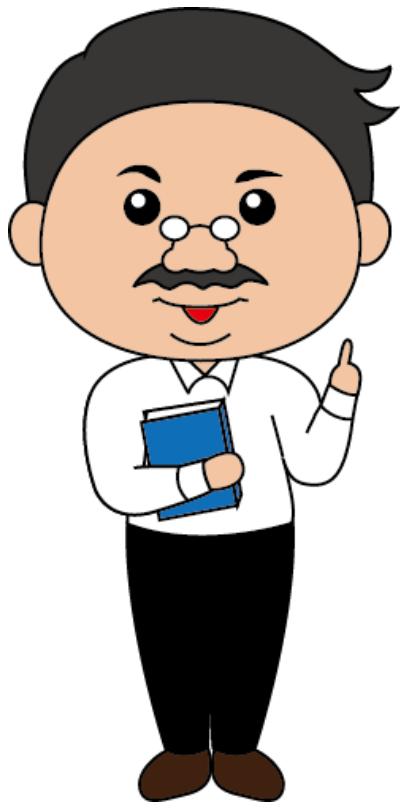
2-1 資料分析之基本概念

2-2 資料處理之常用演算法

2-3 資料處理之軟體工具—試算表

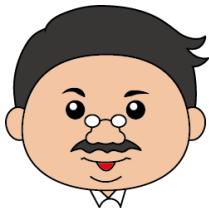
2-4 資料處理之軟體工具—Python





## 2-1 資料分析之基本概念

- 2-1-1 設定研究目標
- 2-1-2 撷取資料
- 2-1-3 資料準備
- 2-1-4 資料探索
- 2-1-5 資料建模
- 2-1-6 呈現結果



## 2-1 資料分析之基本概念

- 資料科學分析是使用方法分析大量的資料，以擷取出當中所蘊含的知識，通常有六種常見的目的：

異常偵測

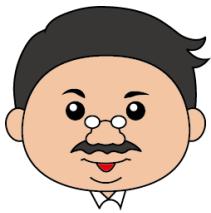
關聯規則

聚類

分類

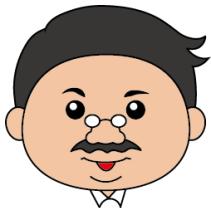
回歸

匯總



## 2-1-1 設定研究目標

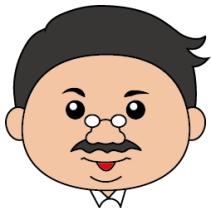
- 一個資料分析的專案從了解專案3個W開始，也就是：
  - 期待做出什麼(What) ?
  - 為什麼要做這樣的分析(Why) ?
  - 專案的規模大小(How) ?



## 2-1-1 設定研究目標

- 這三個問題的答案就是研究目標。
- 研究目標應該是明確的，內容描述是能被理解，且有一個具有時間表的行動計畫。
- 在專案形成的初期階段，經常會需要經驗豐富的人員進行指導，以設定研究目標。





## 2-1-1 設定研究目標

- 設定研究目標是要花費很多時間來確認其重要性，一個好的目標通常會定義出下列項目：

一個明確的研究目標

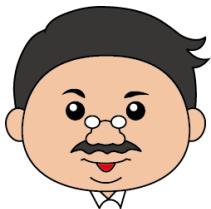
具體的成果和陳述

如何呈現資料分析

預期用到哪些資源

證明這是一個可達成的專案

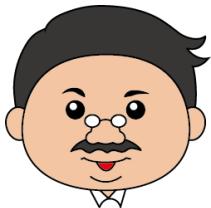




## 2-1-2 摳取資料

- 摳取相關資料，理論上必須到該領域環境中，設計一套資料擳取的流程。
- 但大部分情況是已經有單位將資料大量擳取，並且儲存下來可供使用；或者能從其他地方，協助我們取得資料。

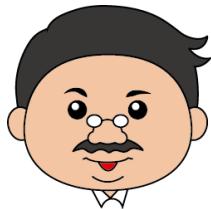




## 2-1-2 摳取資料

- 隨著網際網路的發達，越來越多的資料是可以公開取得，可能是簡單的文字檔案，或是複雜的資料庫形式。
- 在資料擷取階段，會檢查取得的資料是否與資料說明文件所描述的一致，也會檢查其資料形式是否正確。



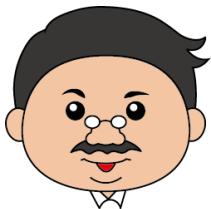


## 2-1-2 摶取資料

### ■ 開放資料來源

提供資料的網站	網址
中華民國政府的資料開放平臺	<a href="http://data.gov.tw">http://data.gov.tw</a>
交通部觀光統計資料庫	<a href="http://stat.taiwan.net.tw">http://stat.taiwan.net.tw</a>
內政部資料開放平台	<a href="http://data.moi.gov.tw">http://data.moi.gov.tw</a>
國家發展委員會統計資料庫	<a href="https://www.ndc.gov.tw">https://www.ndc.gov.tw</a>
疾病管制署資料開放平台	<a href="https://data.cdc.gov.tw">https://data.cdc.gov.tw</a>

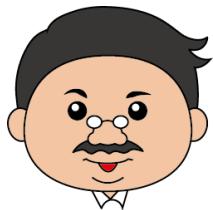




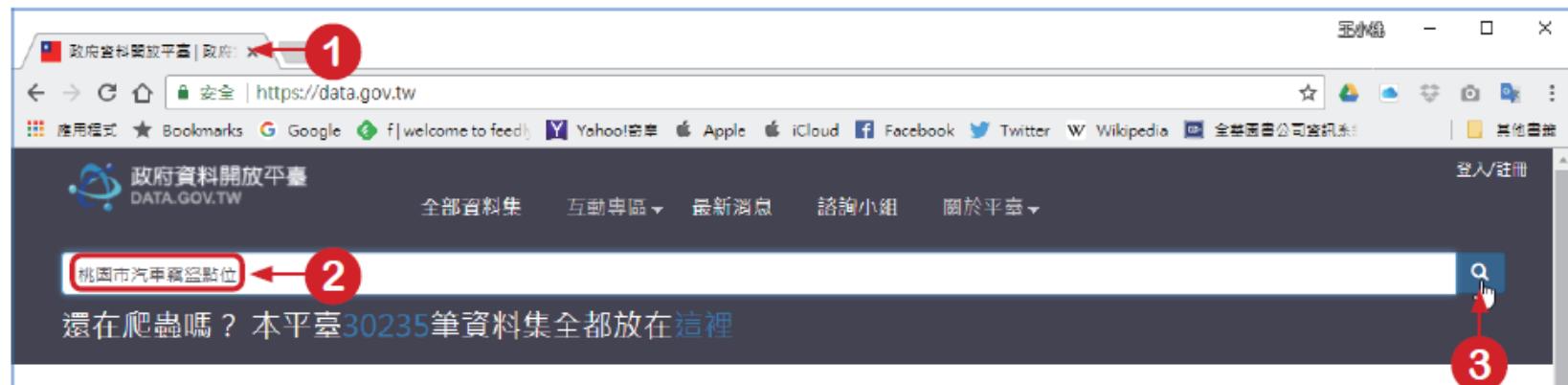
## 2-1-2 摳取資料

### 擳取資料

- 01 進入政府資料開放平台網站中(<http://data.gov.tw>)。
- 02 於搜尋欄位中輸入「桃園市汽車竊盜點位」關鍵字。
- 03 輸入好後按下「**搜尋**」按鈕，或是**Enter**鍵。



## 2-1-2 摶取資料

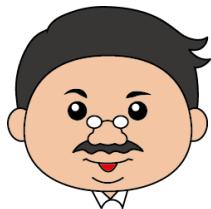


2-1

2-2

2-3

2-4



## 2-1-2 摶取資料

04 搜尋出相關資料後，點選進入該資料頁面中。

The screenshot shows a search results page for "TaoYuan City Car Theft Locations".  
Left sidebar (Place of Origin):

- 地方機關:
  - 桃園市 1
- 主題分類:
  - 其他 1
- 提供機關:
  - 桃園市政府警察局 1
- 服務分類:
  - 生活安全及品質 1
- 檔案格式

Top right (Search & Sort):

- 篩選條件: 桃園市汽車竊盜點位
- 排序方式: 上架日期 新至舊

Main content:

- 共1筆，本頁顯示1-1筆
- 桃園市汽車竊盜點位** (highlighted with a red box and a red arrow labeled '4')
- 桃園市汽車竊盜點位，提供犯罪發生的時間、管轄單位及鄰近經緯度。
- 主要欄位說明:
- type(犯罪發生的種類)、time(犯罪發生的年、月、日)、year(犯罪發生的年度)、month(犯罪發生的月份)、date(犯罪發生日期)、bureau(該案管轄的分局)、station(該案管轄的派出所)、Latitude(犯罪發生的緯度)、Longitude(犯罪發生的經度)
- 桃園市政府警察局 / 註釋資料更新時間：2017/08/11 01:31
- CSV
- 1236 ± 506 0

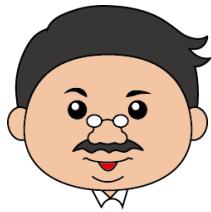


2-1

2-2

2-3

2-4



## 2-1-2 摶取資料

### 05 點選要下載的檔案連結。

桃園市汽車竊盜點位

資料集評分: ★☆☆☆☆  
No votes yet

資料集描述: 桃園市汽車竊盜點位，提供犯罪發生的時間、管轄單位及鄰近經緯度。

主要欄位說明: type(犯罪發生的種類)、time(犯罪發生的年、月、日)、year(犯罪發生的年度)、month(犯罪發生的月份)、date(犯罪發生日期)、bureau(該案管轄的分局)、station(該案管轄的派出所)、Latitude(犯罪發生的緯度)、Longitude(犯罪發生的經度)

資料資源: [CSV](#) 5 料 10606\_2.csv

提供機關: 桃園市政府警察局

提供機關聯絡人: 林先生 (033363488)

更新頻率: 每月

授權方式: 政府資料開放授權條款-第1版

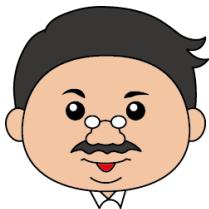


2-1

2-2

2-3

2-4



## 2-1-2 摷取資料

06 開啟**另存新檔**對話方塊，設定檔案要存放的位置。

07 按下**存檔**按鈕，進行下載的動作。

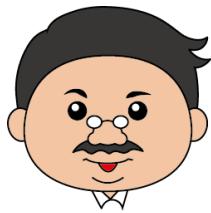


2-1

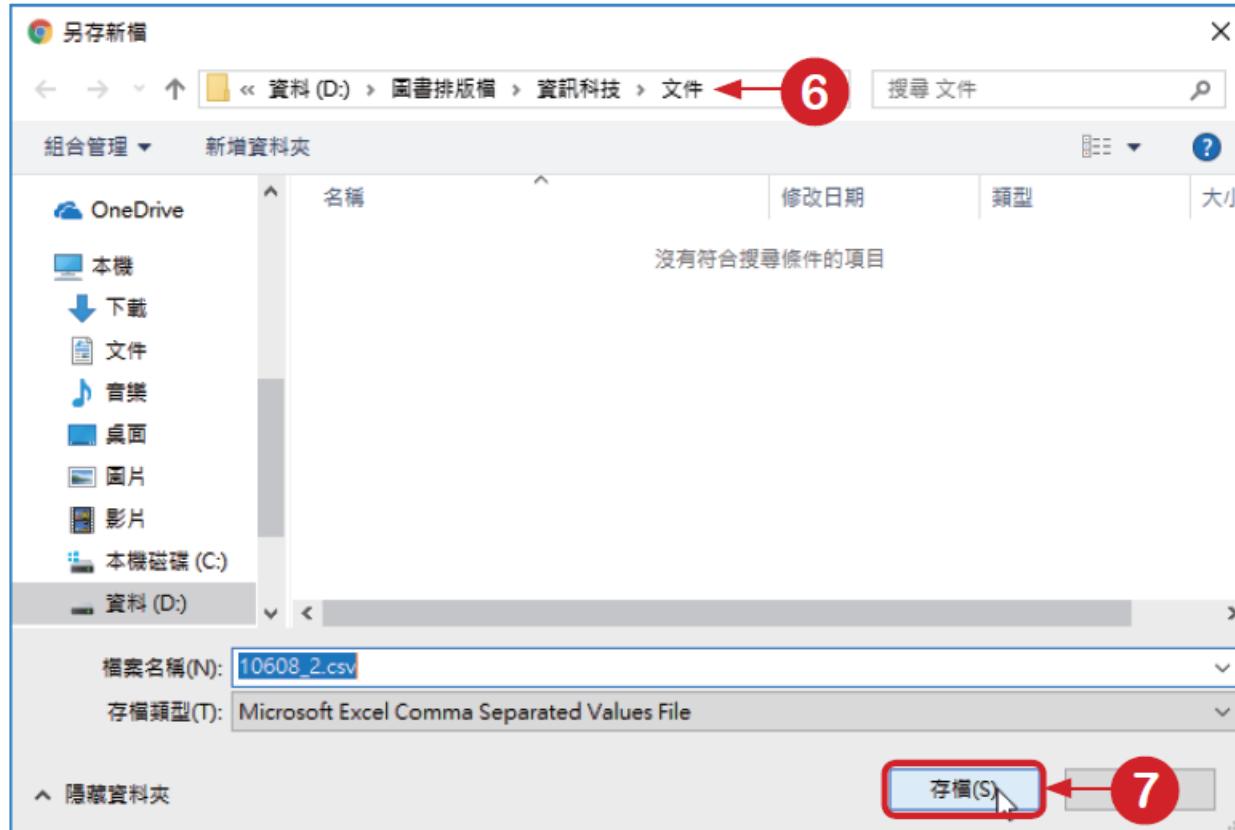
2-2

2-3

2-4



## 2-1-2 摄取資料

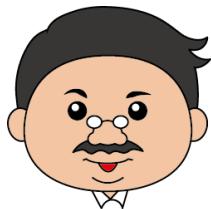


2-1

2-2

2-3

2-4



## 2-1-2 摶取資料

08 下載完成後，即可點選該檔案選單鈕，於選單中選擇要如何檢視該檔案。

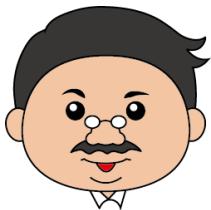


2-1

2-2

2-3

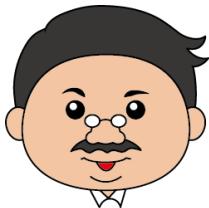
2-4



## 2-1-3 資料準備

- 在資料準備這個階段，主要是將資料進行清理及準備。
- 清理工作是清除一些不相關資料，以避免在後面的步驟中花太多時間處理奇怪的輸出，而得到較好的模型。

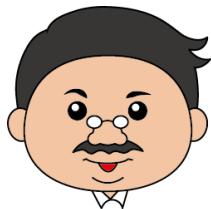




## 2-1-3 資料準備

- 準備工作則是將資料轉換成模型所需的特定格式表示，將一些不用的資料或不用的欄位清除，甚至合併一些相關資料，以簡化或完整化我們的資料。
- 取得到的資料會有不同的形態，不同形態的資料通常也會使用到不同的工具或技術。





## 2-1-3 資料準備

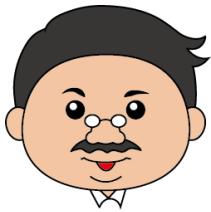
■ 資料形態主要會有下列幾種類別：

結構化

非結構化

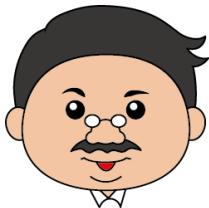
機器產生

資料流



## 2-1-3 資料準備

- 第3列的部分資料是空的，應予以清除。
- B欄位又與C、D、及E欄位重複，以予以合併。
- 第20列的資料與其他資料差異過大，應是局外資料，應予以忽略。



## 2-1-3 資料準備

桃園市汽車竊盜事件統計表

檔案 編輯 檢視

式 說明 所有變更都已儲存到雲端硬碟

註解 共用

B欄位與C、D及E欄位重複，應予以合併

第三列的部分資料是空的，應予以清除

第20列的資料與其他資料差異過大，應是局外資料，應予以忽略

	A	B	C	D	E	F	G	H	I	J	K
1	type	time	year	month	date	breau	station	lat	lon		
2	汽車竊盜	1060626	106	6	26	大園分局	新坡所	NA	NA		
3	汽車竊盜	1060625	106	6	25	楊梅分局	楊梅所	24.9374587	121.1428692		
4	汽車竊盜	1060623	106	6	23	平鎮分局	北勢所	24.9403508	121.2187712		
5	汽車竊盜	1060623	106	6	23	龜山分局	大華所	25.0510817	121.3636367		
6	汽車竊盜	1060620	106	6	20	楊梅分局	楊梅所	24.903438	121.127201		
7	汽車竊盜	1060620	106	6	20	桃園分局	大樹所	24.9845663	121.3128774		
8	汽車竊盜	1060618	106	6	18	龍潭分局	石門所	24.8386295	121.2341451		
9	汽車竊盜	1060617	106	6	17	龍潭分局	聖亭所	24.8834369	121.211422		
10	汽車竊盜	1060617	106	6	17	中壢分局	普仁所	24.9506092	121.2371469		
11	汽車竊盜	1060613	106	6	13	平鎮分局	北勢所	24.947511	121.217197		
12	汽車竊盜	1060612	106	6	12	中壢分局	普仁所	24.9438941	121.247631		
13	汽車竊盜	1060612	106	6	12	大園分局	草漯所	25.0498546	121.1172268		
14	汽車竊盜	1060609	106	6	9	龍潭分局	中興所	24.8671686	121.2310062		
15	汽車竊盜	1060608	106	6	8	大溪分局	圳頂所	24.8974614	121.2760056		
16	汽車竊盜	1060607	106	6	7	楊梅分局	楊梅所	24.8962875	121.1382351		
17	汽車竊盜	1060607	106	6	7	平鎮分局	宋屋所	24.9483437	121.2102846		
18	汽車竊盜	1060606	106	6	6	中壢分局	龍興所	24.9381149	121.23147		
19	汽車竊盜	1060606	106	6	6	中壢分局	仁愛所	24.94314	121.2499464		
20	汽車竊盜	1060605	106	6	5	平鎮分局	建安所	24.9270682	121.22219991		

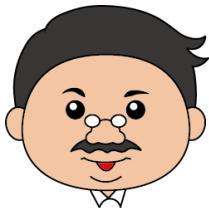


2-1

2-2

2-3

2-4

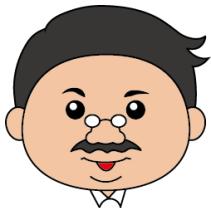


## 2-1-3 資料準備

### ■ 常見的錯誤資料及處理方式

錯誤型態描述	可能的處理方式
消失的空資料	移除該筆資料
多餘的空白字元	刪除多餘空白字元
數值過大或過小	修正值或移除該筆資料
資料內容誤植	修正資料

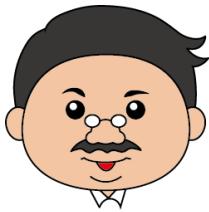




## 2-1-4 資料探索

- 在資料探索階段，通常會將資料用統計圖表表示出來，以視覺化方式理解當中所透露的訊息。
- 在這個階段中不專注在資料的清理，但通常透過圖表的顯示能讓我們發現在前一階段所漏失的處理，此時就得回到前一步驟清理資料。





## 2-1-4 資料探索

■ 使用何種圖表來幫助資料探索並沒有一定的標準。

折線圖及區域圖

- 變數範圍被切割成離散的分類範圍，顯示每個範圍內的資料總數關係。

柱狀圖及長條圖

- 協助資料的分布形態、集中趨勢及變異性。

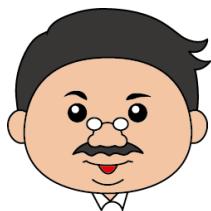
圓餅圖

- 能顯示比率關係。

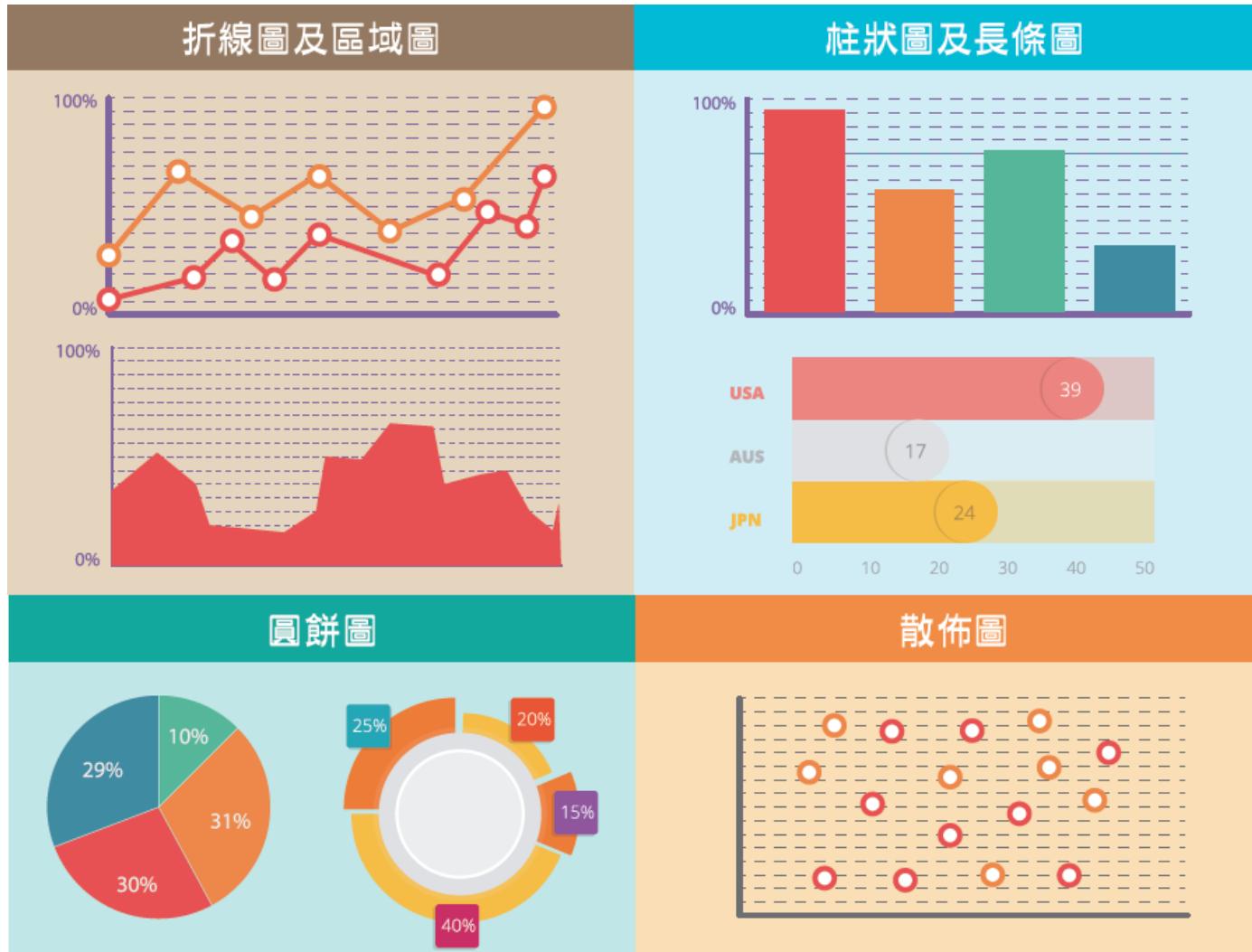
散佈圖

- 能顯示兩變數間之相關性或因果關係。





## 2-1-4 資料探索

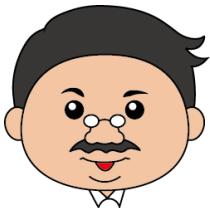


2-1

2-2

2-3

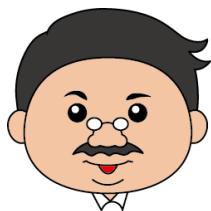
2-4



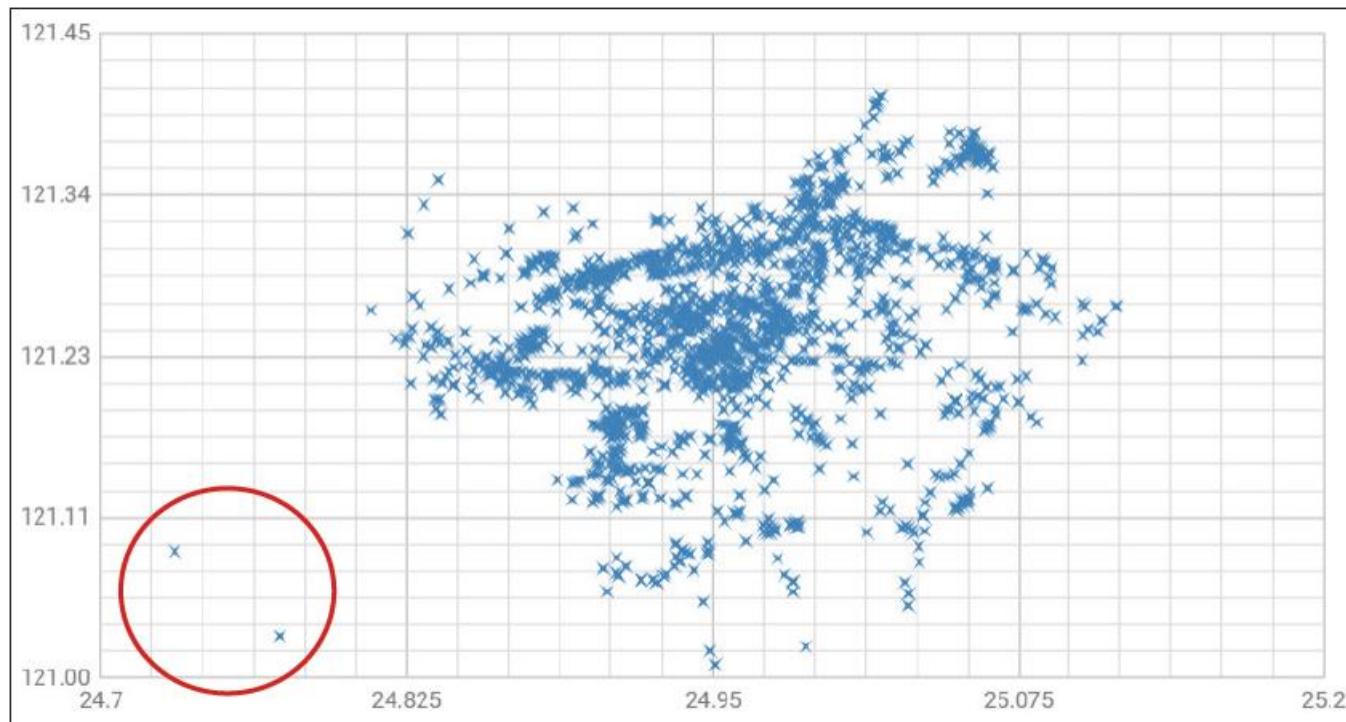
## 2-1-4 資料探索

- 若依經度及緯度在散佈圖上表示，就會形成類似座標平面的資料顯示。
- 透過圖表，可以很輕易的發現一些在前一階段應清除的資料，如左下角的局外資料。此時，就得回到上一步驟，再將資料清理。





## 2-1-4 資料探索

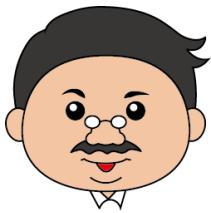


2-1

2-2

2-3

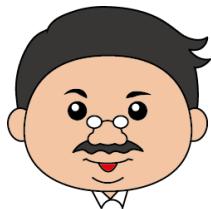
2-4



## 2-1-5 資料建模

- 在這個階段所要做的，就是在機器學習 (Machine Learning)、資料探勘 (Data Mining)、人工智慧或統計等領域中，找到適合的資料分析方法來建立模型。





## 2-1-5 資料建模

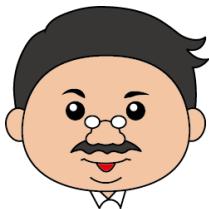
- 大部分的模型在建立時，會分為三個主要的步驟：

選擇分析模型以及將資料變數輸入到模型中

啟動模型運算

診斷運算結果及模型比較





## 2-1-5 資料建模

### 資料建模

- 桃園市汽車竊盜點位資料中，選擇用lat及lon變數輸入模型中。
- 在此挑選線性迴歸模型，來分析竊盜點位的分佈情況。

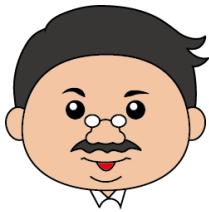


2-1

2-2

2-3

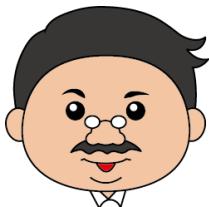
2-4



## 2-1-5 資料建模

■ 一般試算表軟體可以啟動線性迴歸模型的運算，基本步驟是：

- 01 • 選取lat及lon欄位。
- 02 • 插入散佈圖表。
- 03 • 點選其中一資料點進行編輯。



## 2-1-5 資料建模

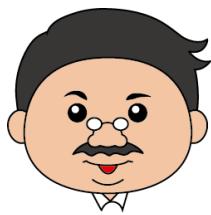
04

- 勾選圖表編輯器中的趨勢線選項，就會畫出線性迴歸模型。

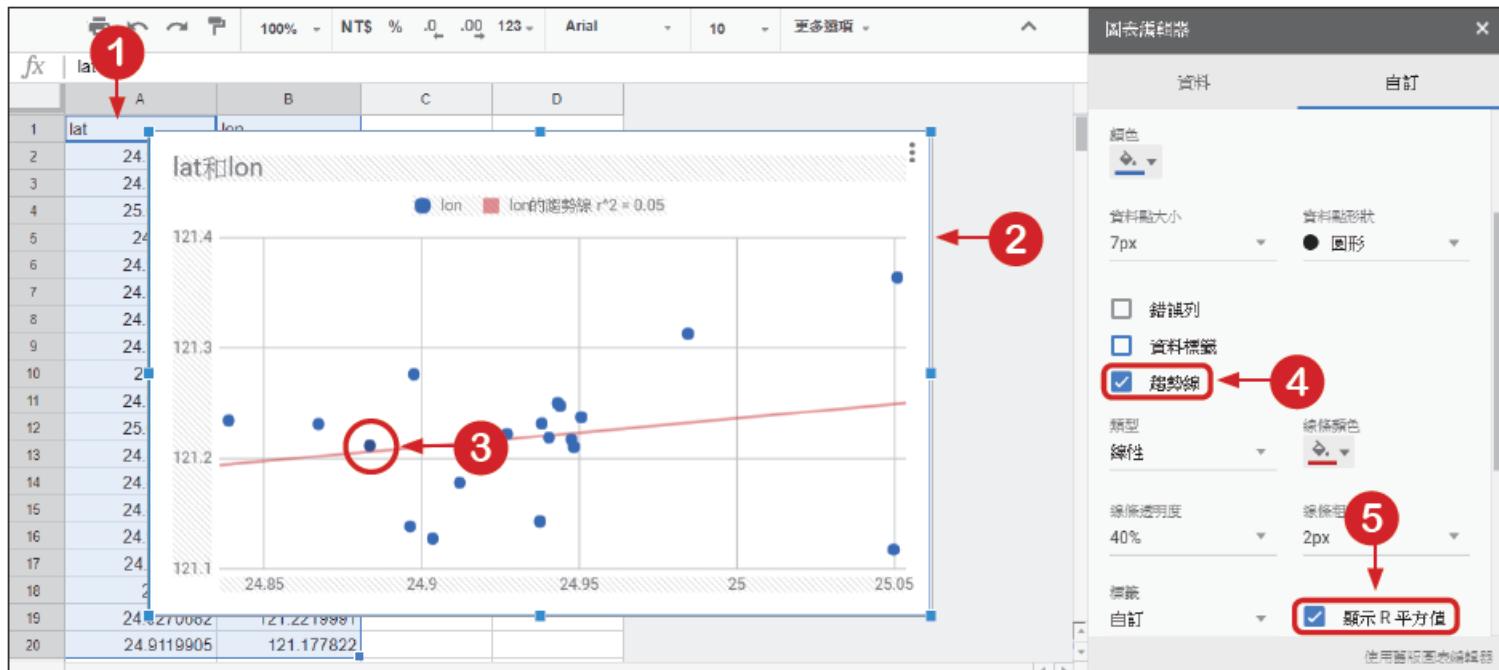
05

- 若再勾選顯示R平方值，就會在圖表中顯示R平方值，R平方計算出資料偏移該線性迴歸模型的程度，有助於診斷及進行模型比較。





## 2-1-5 資料建模

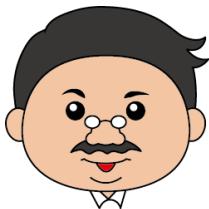


2-1

2-2

2-3

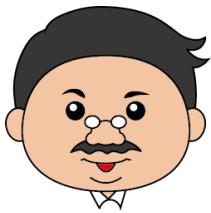
2-4



## 2-1-6 呈現結果

- 在成功分析好資料後，就可以將發現的成果公諸於世。呈現結果的方式通常有兩種：
- 將計算結果在投影片上表示出來
- 將模型提出供人使用





## 2-1-6 呈現結果

- 若是前者，就有賴平時的報告訓練，讓人了解工作成果；若是後者，則是將結果開發成應用程式供人使用。



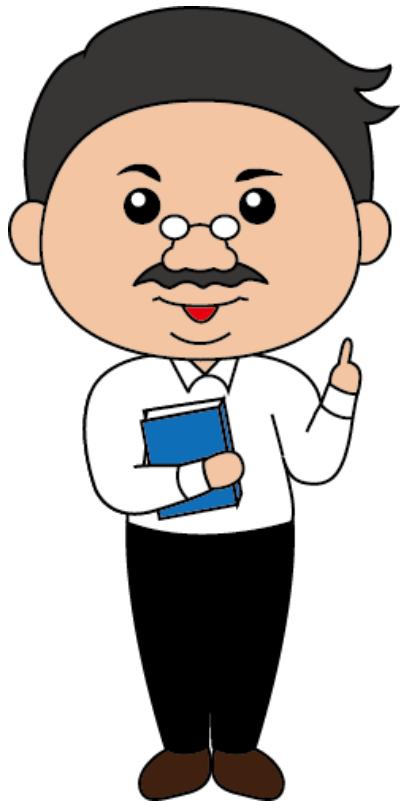
2-1

2-2

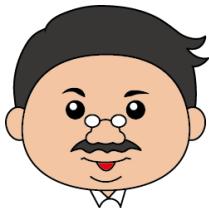
2-3

2-4

## 2-2 資料處理之常用演算法



- 2-2-1 監督式學習
- 2-2-2 非監督式學習
- 2-2-3 半監督式學習



## 2-2 資料處理之常用演算法

■ 機器學習就是資料分析時建立模型的過程，以便洞察資料所表現出來的現象，主要分為四個步驟：

STEP01

- 選擇模型

STEP02

- 訓練模型

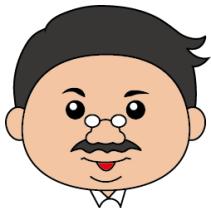
STEP03

- 驗證模型

STEP04

- 應用模型

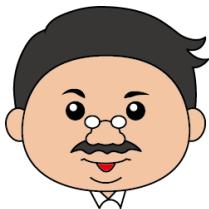




## 2-2 資料處理之常用演算法

- 選擇模型是依據輸入的資料及預期達到的目標來決定使用何種模型。
- 訓練及驗證時會採用已知的樣本，訓練出符合資料表現的模型，當模型能符合預期結果時，便可應用於預測或檢驗新的觀測資料了。





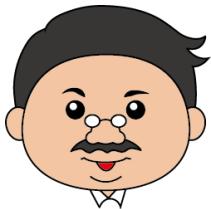
## 2-2 資料處理之常用演算法

■ 訓練及驗證過程時，依據人為介入的程度，可分為三種學習類型：

監督式(Supervised)

非監督式(Unsupervised)

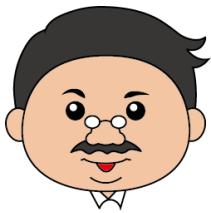
半監督式(Semi-supervised)



## 2-2-1 監督式學習

- 監督式學習只能應用於經過人為標註過的資料，主要用於資料科學分析中的分類、估計或預測等應用。
- 例如在桃園市汽車竊盜點位中，我們已經知道點位是屬於哪個分局的資料，便可以為這些點位標註資訊。

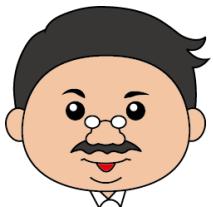




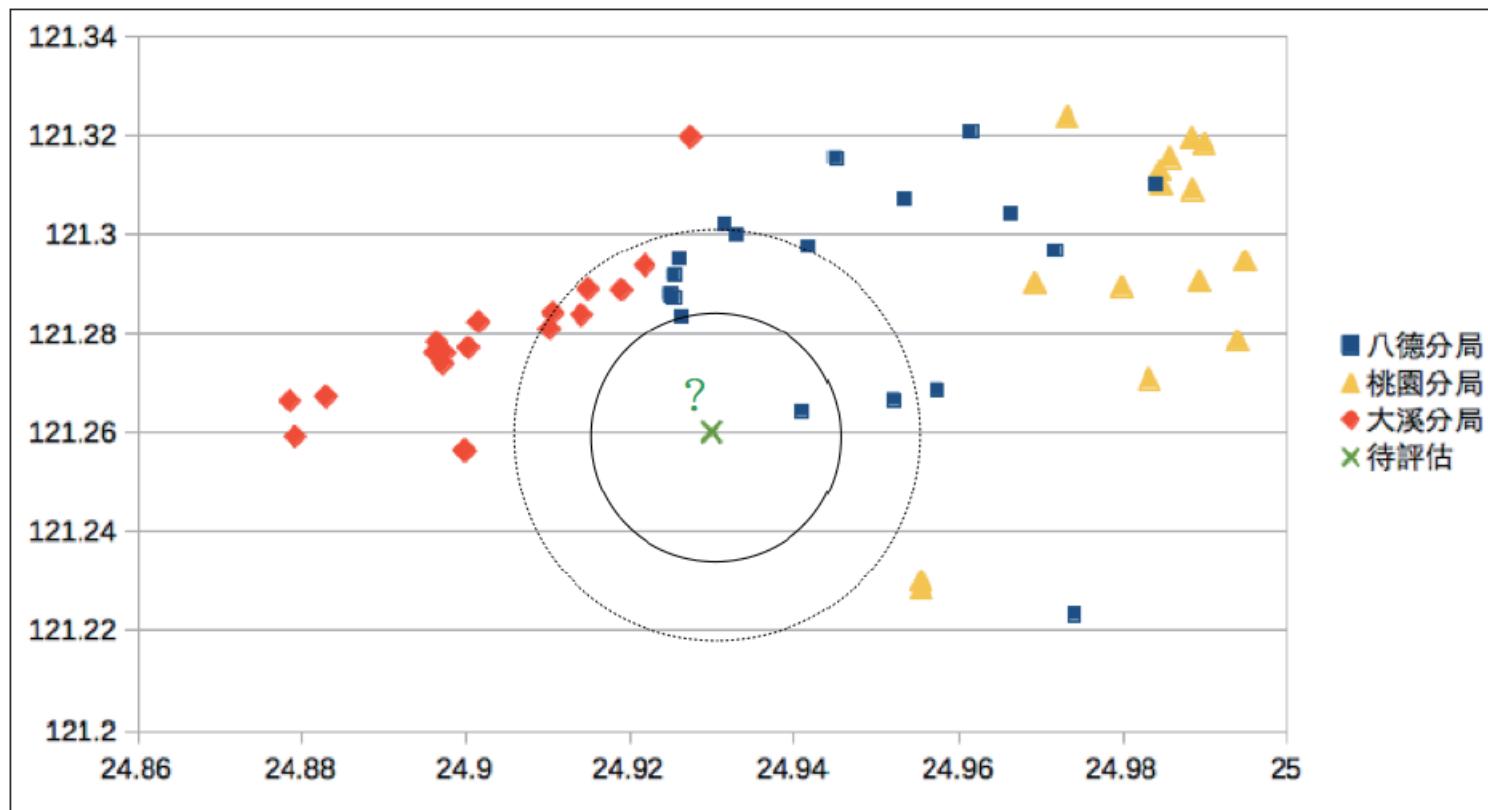
## 2-2-1 監督式學習

- 假設今天發生了一件汽車竊盜案件，我們希望知道哪個分局可以提供較佳的協助，就可以找尋竊盜點位與該案件最接近的幾個點位，請求相關分局協助。





## 2-2-1 監督式學習

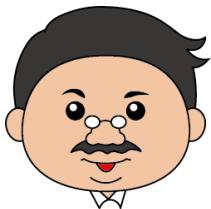


2-1

2-2

2-3

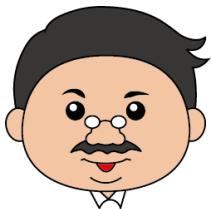
2-4



## 2-2-1 監督式學習

- 這種利用鄰近資料來找相關性的問題就是**近鄰問題(k-Nearest Neighbor)**，藉由計算與已知類別案件之相似度，來評估未知類別案件的可能分類。





## 2-2-1 監督式學習

- 在實務上最常見的應用，就是線上購物時，系統會推薦相關產品供消費者選擇，就是利用近鄰演算法所產生的結果。

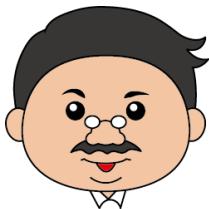


2-1

2-2

2-3

2-4



## 2-2-1 監督式學習

### 近鄰問題

#### 輸入

- 待評估的案件V及n筆案件資料，資料儲存在陣列A中，每一案件i包括所屬分局(類別)、發生地x座標、及發生地y座標，分別以A[i].c、A[i].x、及A[i].y表示。

#### 輸出

- 待評估的案件V，屬於哪一類別，亦即我們可以由哪一分局提供協助。

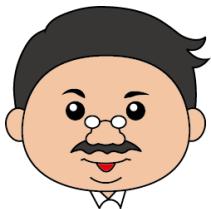


2-1

2-2

2-3

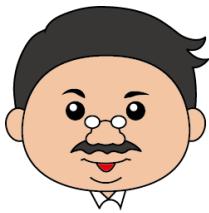
2-4



## 2-2-1 監督式學習

- 有一種解決近鄰問題的近鄰演算法，是先定義近鄰的數量  $k$ ，從  $k$  個最近的鄰居中，看看哪一類的鄰居最多，來決定其類別，因此又稱為  **$k$ -近鄰演算法**。





## 2-2-1 監督式學習

- 在實務上最常見的應用，就是線上購物時，系統會推薦相關產品供消費者選擇，就是利用近鄰演算法所產生的結果。

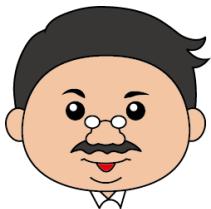


2-1

2-2

2-3

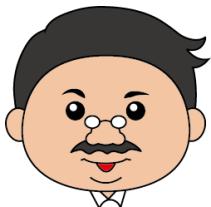
2-4



## 2-2-1 監督式學習

1. 對於所有  $i = 1..k$ ，重覆執行下列步驟
  - 1.1.  $D[i].c \leftarrow ""$ ， $D[i].d \leftarrow \infty$ ， $C[i] \leftarrow 0$
2. 對於所有  $i = 1..n$ ，重覆執行下列步驟
  - 2.1.  $d \leftarrow \text{dist}(A[i], V)$  //計算  $A[i]$  及  $V$  的距離
  - 2.2. 對於所有  $i = 1..k$ ，重覆執行下列步驟 //選擇排序
    - 2.2.1. 如果  $d \leq D[i].d$  執行
      - 2.2.1.1. 對於所有  $j = i..k-1$ ，重覆執行下列步驟
        - 2.2.1.1.1.  $D[j+1] \leftarrow D[j]$
        - 2.2.1.1.2.  $D[i].c \leftarrow A[i].c$ ， $D[i].d \leftarrow d$





## 2-2-1 監督式學習

3.  $\max \leftarrow -1$  ,  $\maxc \leftarrow 0$

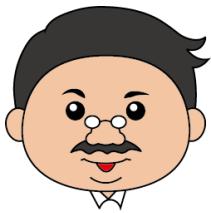
4. 對於所有  $i = 1..k$  , 重覆執行下列步驟

4.1. 對於所有  $j = 1..k$  , 重覆執行下列步驟

4.1.1. 如 果  $D[i].c = D[j].c$  , 執 行  
 $C[i] \leftarrow C[i] + 1$

4.2. 如 果  $C[i] > \max$  , 執 行  $\max \leftarrow C[i]$  ,  
 $\maxc \leftarrow D[i].c$

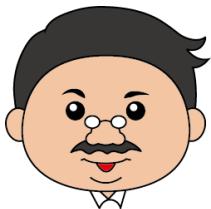
5. 輸出  $\max$



## 2-2-1 監督式學習

- 近鄰演算法的發展歷史相當悠久其優點是方法簡單，容易實作，幾乎沒有什麼參數需要設定或訓練，且可以用於多類別的情況。

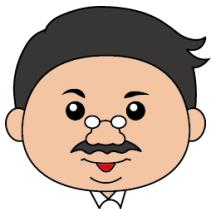




## 2-2-1 監督式學習

- 在實務應用上則有一些常見的問題及解決方法：第一個問題是若已知類別的案件(樣本)與待評估案件距離過遠，卻仍用於考慮的話，容易產生分類缺陷；為避免這種情況，可以設定距離排除樣本，或依距離 $d$ 設定樣本的權重(如 $1/d$ )，來解決此問題。

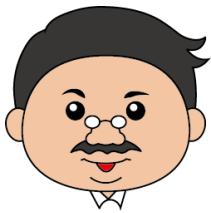




## 2-2-1 監督式學習

- 這種利用鄰近資料來找相關性的問題就是**近鄰問題(k-Nearest Neighbor)**，藉由計算與已知類別案件之相似度，來評估未知類別案件的可能分類。

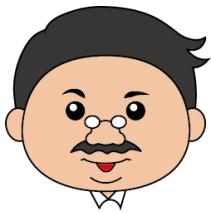




## 2-2-1 監督式學習

- 在實務上最常見的應用，就是線上購物時，系統會推薦相關產品供消費者選擇，就是利用近鄰演算法所產生的結果。

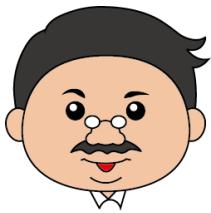




## 2-2-1 監督式學習

- 第二個問題是其計算量比其他模型計算複雜度多很多；解決方式可以先將樣本群聚，以幾個抽象的樣本代表各群，供待評估案件計算。

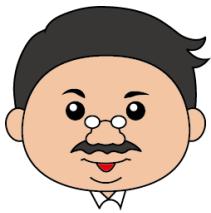




## 2-2-1 監督式學習

■ 除了近鄰演算法，還有幾種比較常見的監督式學習，像是

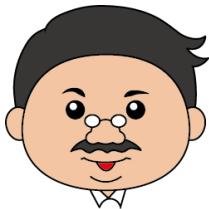
- 支持向量機
- 線性回歸
- 類神經網路
- 決策樹學習
- 隨機森林演算法



## 2-2-1 監督式學習

- 監督式學習的模型表現，與資料的特性非常相關，這使得挑選模型被視為一種藝術，而不是單純科學評估，有時得經過長久的測試及經驗判斷。

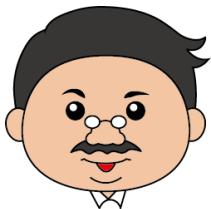




## 2-2-1 監督式學習

演算法名稱	說明
支持向量機	使用線性方程式，透過線性組合將資料點分為兩類，以便將特徵空間分為兩部分，成為分類模型。
線性回歸	以線性回歸方程式對變數之間的關係進行建模的一種分析，方程式可以是一個或多個變數的線性組合。
類神經網路	以數學函數模擬神經的反應，以建立模型模仿生物神經網路的結構和處理功能。
決策樹學習	訓練時，主要是建立一連串的判斷指標，對已知的資料點進行正確分類。然後，就可以根據每個節點的判斷條件將測試資料進行分類。
隨機森林演算法	透過隨機的方式選擇決策樹，由多決策樹決定測試資料的分類。

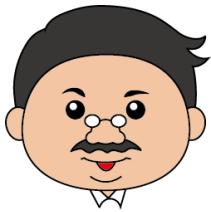




## 2-2-2 非監督式學習

- 在大部分的情況下，資料來源是觀測來的，而觀測資料通常是自動產生而不會有標註。
- 因此，除非我們先進行人為標註，否則監督式學習方法是無法進行應用的。

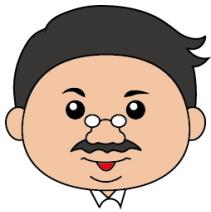




## 2-2-2 非監督式學習

- 取而代之的，非監督式學習不需要人為標註觀測資料，就可以將資料進行聚類及關聯規則分析。
- 聚類的目的是將資料分開成幾群，資料在同一群內的會彼此相似，和不同群的資料則會有極大的差異。

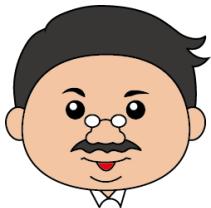




## 2-2-2 非監督式學習

- 以k-平均聚類為例，將資料劃分到k個類別中，使得每筆資料都與其所屬類別內的資料屬性平均值最近，亦即使得所有資料與其同一類別內的平均值差異量之平方和最小。

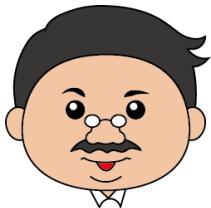




## 2-2-2 非監督式學習

### k-平均聚類問題

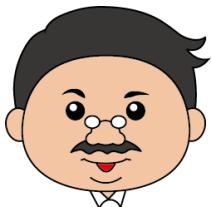
- 輸入：n筆資料儲存在陣列A中，每一筆資料i包含x及y屬性，分別以 $A[i].x$ 及 $A[i].y$ 表示。
- 輸出：每筆資料 $A[i]$ 的類別，使得 $dist^2(A[1],\mu_1)+dist^2(A[2],\mu_2)+...+dist^2(A[n],\mu_n)$ 最小，其中 $\mu_i$ 為 $A[i]$ 所屬類別內的資料屬性平均值。



## 2-2-2 非監督式學習

- 解決k-平均聚類問題的演算法計算複雜度通常相當高，一般情況下，會使用效率比較高的啟發式演算法，以快速得到一個可接受的類別分配結果。
- 其中最有名的，是一個被稱為Lloyd演算法的k-平均演算法。





## 2-2-2 非監督式學習

1. 對於所有  $i = 1..k$ ，重覆執行下列步驟

1.1.  $M[i].x \leftarrow \text{RANDOM}()$  ,  $M[i].y \leftarrow \text{RANDOM}()$

2. 重覆執行下列步驟  $t$  次

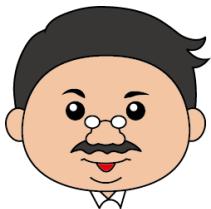
2.1. 對於所有  $i = 1..n$ ，重覆執行下列步驟

2.1.1.  $\min \leftarrow \infty$  ,  $\text{best} \leftarrow 0$

2.1.2. 對於所有  $i = 1..k$ ，重覆執行下列步驟

2.1.2.1.  $d \leftarrow \text{dist}(A[i], M[k])$

2.1.2.2. 如果  $d < \min$  ，執行  $\min \leftarrow d$  ,  
 $\text{best} \leftarrow k$



## 2-2-2 非監督式學習

2.1.3.  $A[i].c \leftarrow \text{best}$

2.2. 對於所有  $i = 1..k$ ，重覆執行下列步驟

2.2.1.  $M[i].x \leftarrow 0$ ， $M[i].y \leftarrow 0$ ， $M[i].n \leftarrow 0$

2.3. 對於所有  $i = 1..n$ ，重覆執行下列步驟

2.3.1.  $c \leftarrow A[i].c$

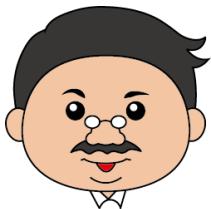
2.3.2.  $M[c].x \leftarrow M[c].x + A[i].x$

2.3.3.  $M[c].y \leftarrow M[c].y + A[i].y$

2.3.4.  $M[c].n \leftarrow M[c].n + 1$

2.4. 對於所有  $i = 1..k$ ，重覆執行下列步驟

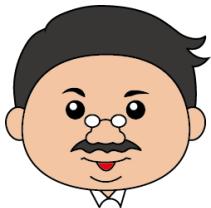
2.4.1.  $M[i].x \leftarrow M[i].x/M[i].n$ ， $M[i].y \leftarrow M[i].y/M[i].n$



## 2-2-2 非監督式學習

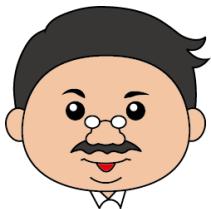
- 在這個演算法中，主要分為分配(2.1)及更新(2.3)兩大步驟。
- 隨機產生的初始化的類別資料之屬性平均值  $M[i]$ (步驟1)，接著該筆資料與  $M[k]$  的距離  $dist(A[i], M[k])$ ，分配該筆資料在距離最小的類別(2.1.3)。





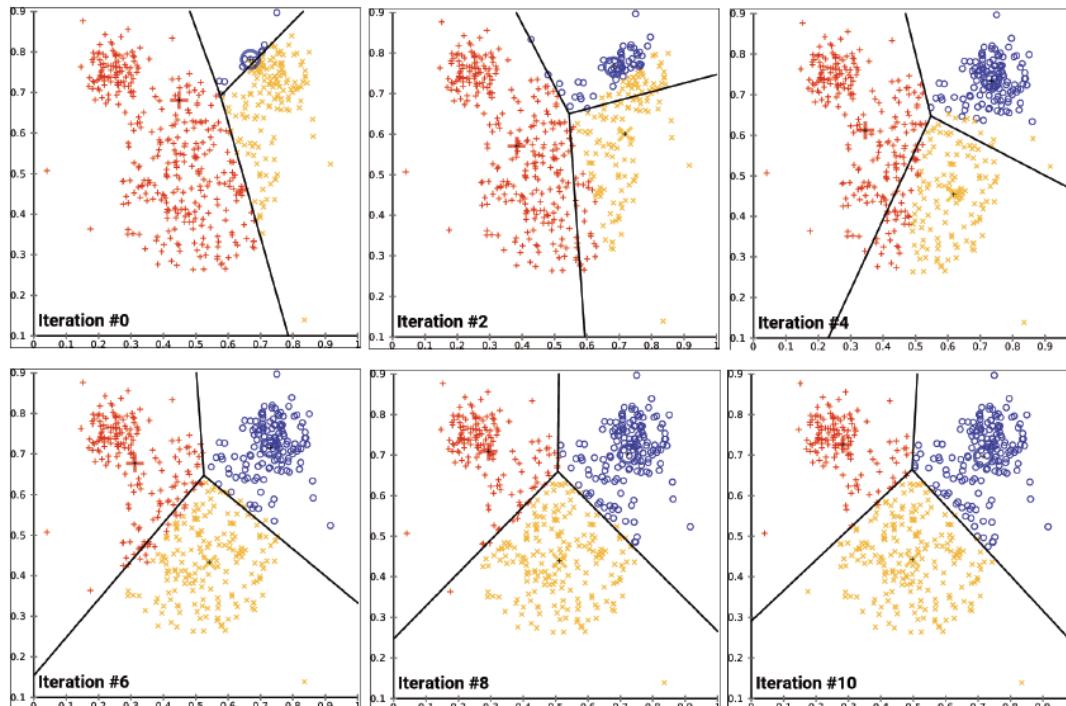
## 2-2-2 非監督式學習

- 將所有資料分配好類別後，將同一類別的資料重新計算出新的屬性平均值，更新 $M[i](2.4)$ 。
- 利用新的屬性平均值 $M[i]$ 去重新分配及再更新的疊代動作  $t$  次。



## 2-2-2 非監督式學習

- 而  $A[i]$  的分類  $A[i].c$  則會隨著疊代次數的增加而漸趨穩定，所以通常可以等分類穩定後輸出結果。

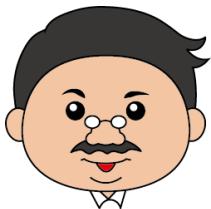


2-1

2-2

2-3

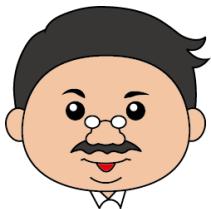
2-4



## 2-2-2 非監督式學習

- 影響k-平均聚類法效率最主要的关键參數k，也是影響分類結果的重要參數。
- 過小的k值當然會使分類結果不正確，過大的k值卻容易產生非最佳的結果，也就是  $\text{dist}^2(\mathbf{A}[1], \mu_1) + \text{dist}^2(\mathbf{A}[2], \mu_2) + \dots + \text{dist}^2(\mathbf{A}[n], \mu_n)$  並非最小。

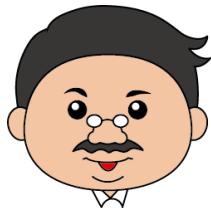




## 2-2-2 非監督式學習

- 因此，在實務上， $k$ 值會經過使用者輸入或經過不同值測試來產生較佳的結果。
- 另外，由於資料與類別平均值的計算多半採用歐幾里得距離，使得同類資料分佈非圓形時，分類結果較不正確。



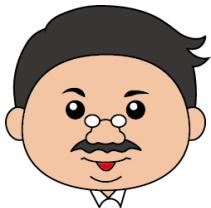


## 2-2-2 非監督式學習

### ■ 常見的非監督式學習演算法

演算法名稱	說明
最大期望演算法	先假設資料的分佈函數，透過已知資料與分佈函數的期望值計算，調整分佈函數，然後不斷重複以得到正確的資料分佈函數。
高斯混合模型	將一般常態分佈資料的假設，延伸為多個常態分佈的組合，以估算資料的分佈狀況。
主成分分析	在資料的多個特徵中找出主要成分，以簡化特徵數。主要成份往往是多個特徵的組合，反而能找出資料的重要面向。
奇異值分解	對資料點進行拆解，以找出含重要資訊的資料要素，然後以簡化後的資料來近似原先的資料分佈。

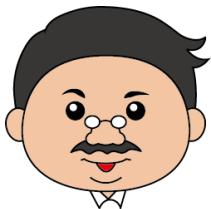




## 2-2-3 半監督式學習

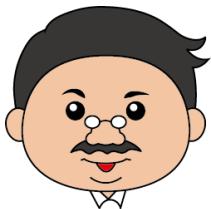
- 透過已知標註的資料來進行監督式學習，顯然能比非監督學習得到更好的模型。
- 然而，因為大部分得到的資料卻是未標註的，為了得到監督式學習的好處，便可以用半監督式學習來混合監督式及非監督式學習方法。





## 2-2-3 半監督式學習

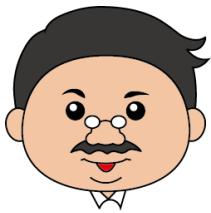
- 透過人為標註部分資料，使用監督式學習方式得到初始模型；接著利用未標註的資料，透過非監督式學習的方式來更新模型。
- 這種整合多種學習技術的方法，就稱為半監督式學習。



## 2-2-3 半監督式學習

- 許多研究已經證實，未標註的資料與少數經過標註的資料一起使用時，可以顯著提高學習準確性。
- 一種常見的半監督式學習方法為**標註傳播**。

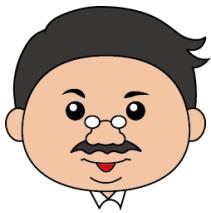




## 2-2-3 半監督式學習

- 大部分資料點是未知類別，當中有三筆資料點則是已知類別的。
- 標註傳播法會將最接近已知類別的資料點標註為同類別，重複執行到所有資料點都被標註類別為止。

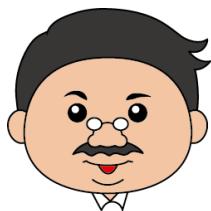




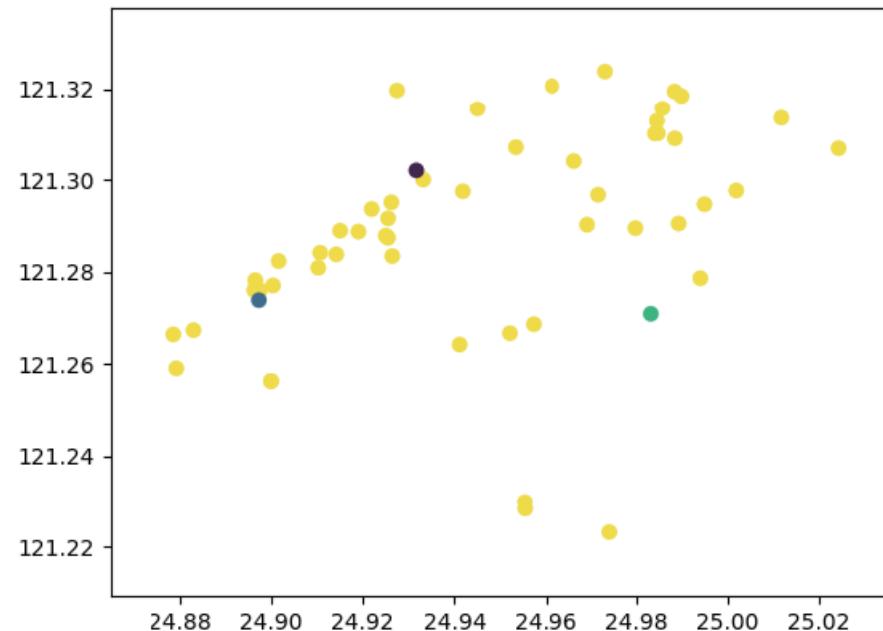
## 2-2-3 半監督式學習

- 在實務的應用上，可以讓使用者手動修正機器學習後的結果，再讓機器學習出更佳的模型出來，因此又稱為主動式學習。



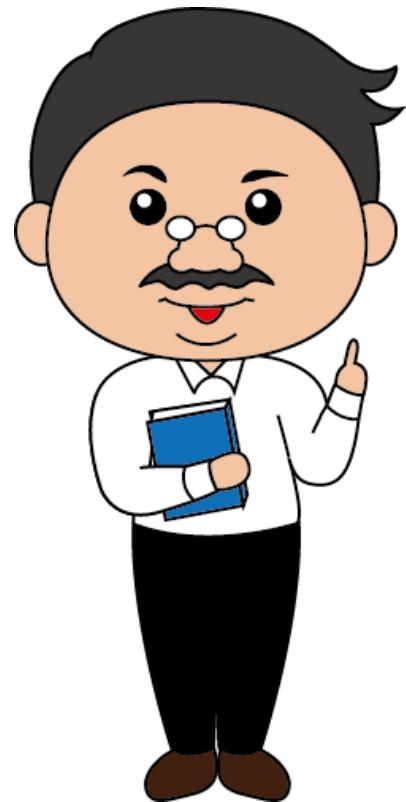


## 2-2-3 半監督式學習



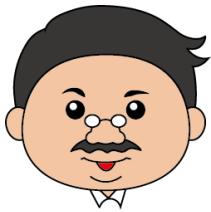
未標註類別資料點為黃色，其他顏色的資料點為已知類別，共三類





## 2-3 資料處理之軟體工具— 試算表

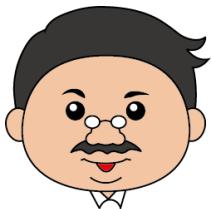
- 2-3-1 資料表示
- 2-3-2 資料處理
- 2-3-3 資料分析
- 2-3-4 資料篩選
- 2-3-5 資料透視表



## 2-3-1 資料表示

- 早期多是使用人工慢慢計算，電腦的出現後，使得計算工作能交由電腦處理完成。
- 但大型電腦的工作處理通常須要排隊等候，而電子試算表的出現，使得人們可以利用個人電腦迅速得到處理結果。





## 2-3-1 資料表示

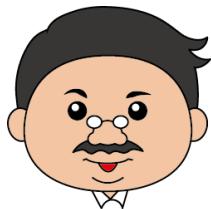
- 由於電子試算表發展的歷史相當悠久，其功能上大致已相當穩定，使得各家電子試算表使用上皆大同小異，目前比較常見有：

Microsoft Excel

LibreOfficeCalc

Google試算表





## 2-3-1 資料表示

The screenshot shows a Google Sheets interface with the following details:

- Title Bar:** Google 試算表 (Google Sheets)
- Address Bar:** 無標題的試算表 - Google Sheets | https://docs.google.com/spreadsheets/d/1o0Yn0s9QJGftE1fH92jJrQ4w02TRemt\_FSG5cqrvtIc/edit#gid=0
- Toolbar:** Includes icons for back, forward, search, and various document functions.
- User Info:** momoco.wang@gmail.com
- Menu Bar:** 檔案 (File), 編輯 (Edit), 檢視 (View), 插入 (Insert), 格式 (Format), 資料 (Data), 工具 (Tools), 外掛程式 (Add-ons), 說明 (Help).
- Tool Buttons:** Print, Refresh, Undo, Redo, Font Size (10), Bold (B), Italic (I), Underline (U), Alignment, Cell Style, More Options.
- Spreadsheet Area:** A 17x10 grid of cells labeled A through J and 1 through 17. Cell E4 is highlighted with a blue border.
- Bottom Navigation:** Includes a new sheet icon, a three-dot menu, and the current sheet name "工作表1" (Sheet1).

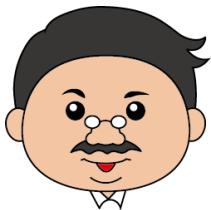


2-1

2-2

2-3

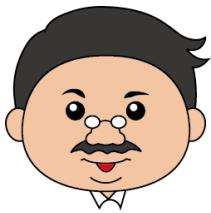
2-4



## 2-3-1 資料表示

- 試算表會顯示由一系列欄與列構成的表格，欄號以A、B、C等英文字母表示，列號以1、2、3等數字表示，表格內的每一儲存格都以欄號與列號表示其位置，如A1表示第A欄第1列。





## 2-3-1 資料表示

- 若要使用試算表中一群連續的儲存格時，可以使用A1:A10來表示A欄的前10個儲存格，或用A1:C10來表示A、B、C三欄的前10個儲存格。

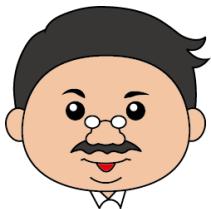


2-1

2-2

2-3

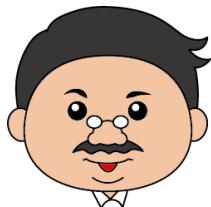
2-4



## 2-3-1 資料表示

- 每一儲存格內可以存放數值、計算式、或文字，而試算表主要的功能便是透過計算式與其他儲存格運算，以得到我們要的資料分析結果。
- 一個檔案內通常包含了數個工作表而形成活頁簿的形式。





## 2-3-1 資料表示

A screenshot of a Google Sheets document titled "無標題的試算表". The spreadsheet has 17 rows and 10 columns labeled A through J. Row 1 contains a single cell in column A which is currently selected and highlighted with a blue border. A speech bubble in the center-right of the sheet area contains the text: "一個活頁簿可以包含多個工作表". At the bottom of the screen, there is a navigation bar with five tabs labeled "工作表1" through "工作表5", with "工作表1" being the active tab. The entire row of tabs is highlighted with a red box.

一個活頁簿可以  
包含多個工作表

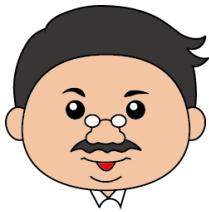


2-1

2-2

2-3

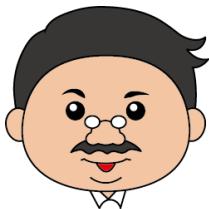
2-4



## 2-3-1 資料表示

- 當使用計算式存取的儲存格是在不同工作表時，該儲存格就在前面加上工作表名稱，例如：工作表1!A1，即表示在工作表1內的A1儲存格。
- 甚至有些試算表軟體(如 Google 試算表)，延伸了這種表示法，足以用來表示在不同活頁簿內的儲存格。

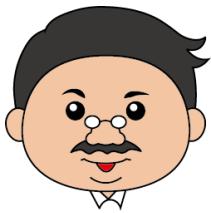




## 2-3-2 資料處理

- 我們將前一節中所使用的桃園市汽車竊盜點位讀取至Google試算表中，數值及文字資料可以直接輸入到儲存格中，例如在O2及P2欄位中輸入待評估的資料點座標。

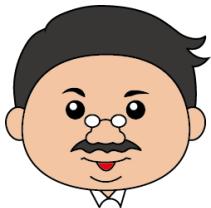




## 2-3-2 資料處理

- 若是要輸入計算式，則是先輸入等於「=」的符號，如： $=3*8$ ，但此時儲存格上顯示的就是計算式的計算結果24，而不是顯示計算式。

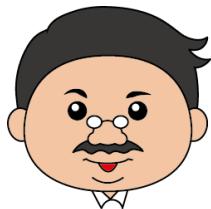




## 2-3-2 資料處理

- 計算式最大的特色就是可以使用儲存格中的資料進行運算，比如我們在K2儲存格輸入=H2-O2，則K2儲存格會顯示H2及O2兩儲存格內數值24.9264036及24.93相減的結果，而上方的公式欄位則會顯示原計算式。





## 2-3-2 資料處理

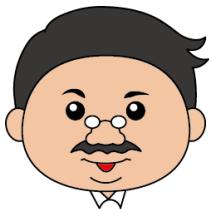
無標題的試算表

公式欄會顯示原計算式

儲存格顯示相減的結果

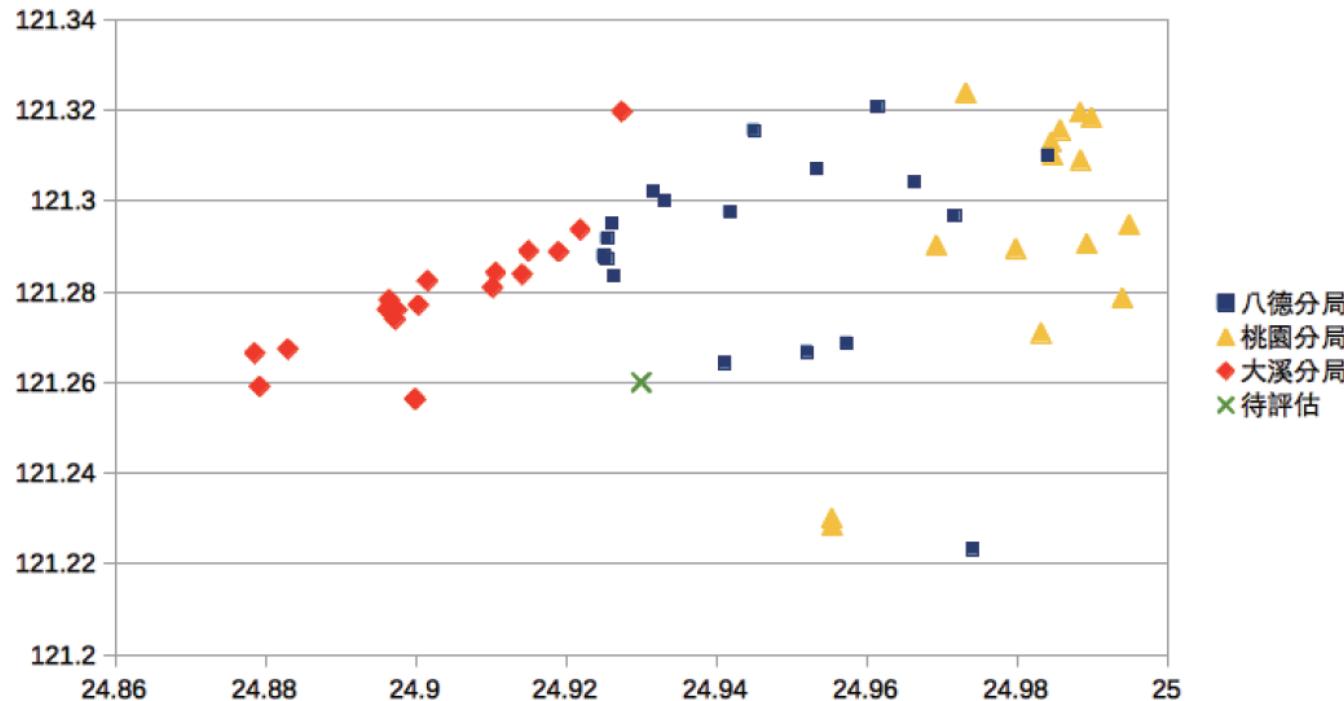
	H	I	J	K	L	M	N	O	P
1	lat	lon	breau	dis				lat	lon
2	24.9264036	121.2834587	八德分局	-0.0035964		待評估	24.93	121.26	
3	24.9534034	121.3071043	八德分局						
4	24.9418178	121.2975227	八德分局						
5	24.9254825	121.2916526	八德分局						
6	24.92546	121.2874246	八德分局						
7	24.926193	121.2951256	八德分局						
8	24.9717124	121.2967754	八德分局						

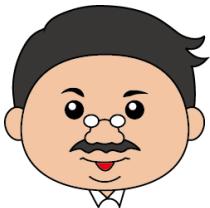




## 2-3-2 資料處理

- 各資料點依座標(lat, lon)在平面圖上表示。

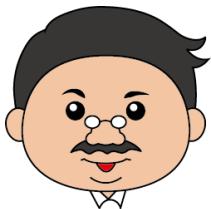




## 2-3-2 資料處理

- 很多問題都可以透過公式的組合，來發揮試算表的威力，若要計算待評估點(24.93, 121.26)與第一個點(24.9264036, 121.2834587)的直線距離，可以利用直線距離的公式  $dis = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$  來計算。

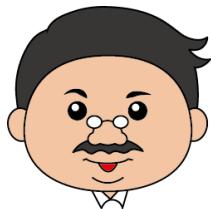




## 2-3-2 資料處理

- 開根號這種複雜的運算，試算表軟體會提供相關的函式供運用。
- 在這個例子中可以使用**sqrt()**這個開根號的函式寫出計算距離的計算式  
 $=\text{sqrt}((O2-H2)^*(O2-H2)+(P2-I2)^*(P2-I2))$ 。





## 2-3-2 資料處理

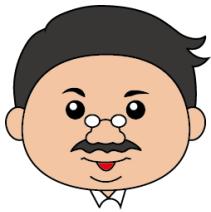
無標題的試算表

檔案 編輯 檢視 插入 格式 資料 工具 外掛程式 說明 所有

fx =sqrt((O2-H2)\*(O2-H2)+(P2-I2)\*(P2-I2))

	H	I	J	K	L
1	lat	lon	breau	dis	
2	24.9264036	121.2834587	八德分局	0.02373277688	
3	24.9534034	121.3071043	八德分局		
4	24.9418178	121.2975227	八德分局		
5	24.9254825	121.2916526	八德分局		
6	24.92546	121.2874246	八德分局		





## 2-3-2 資料處理

■ 一般試算表所提供的函式，可以在函式清單中找到，使用者通常利用搜尋的功能找到所需的相關函式。

Google 試算表函式清單

Google 試算表與市面上大多數桌上型試算表套裝軟體一樣，都能支援儲存格公式。您可以使用這些公式來建立各種函式，藉此管理資料或計算字串長度或數字。

在下面的表格中，我們根據不同的類別，為您一一列出所有支援的函式。使用這些函式時請注意，如果函式元件是由英文字元組成，但並未參照儲存格或資料欄，請務必在前後加上引號。

您可以將 Google 試算表的函式語言變為英文或其他 21 種語言。

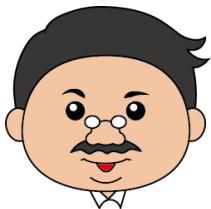
使用關鍵字篩選 ...

類型	名稱	句法	描述
Google	GOOGLEFINANCE	GOOGLEFINANCE(代號, 屬性, 開始日期, 結束日期, 天數, 開窗)	從 Google 財經服務擷取有價證券的最新或過往資訊。 <a href="#">瞭解詳情</a>
Google	SPARKLINE	SPARKLINE(資料, 選項)	在單一儲存格中建立迷你圖表。 <a href="#">瞭解詳情</a>
Google	IMPORTXML	IMPORTXML(網址, XPath_查詢)	匯入多種結構化資料類型的資料，包括 XML、HTML、CSV、TSV 和 RSS 以及 ATOM XML 資訊提供。 <a href="#">瞭解詳情</a>
Google	IMPORTRANGE	IMPORTRANGE(試算表索引, 範圍字串)	匯入指定試算表中特定儲存格的範圍。 <a href="#">瞭解詳情</a>
Google	IMPORTHTML	IMPORTHTML(網址, 查詢, 索引)	將表格或清單中的資料匯入 HTML 網頁。 <a href="#">瞭解詳情</a>
Google	IMPORTFEED	IMPORTFEED(網址, 查詢, 標題, 項數)	匯入 RSS 或 ATOM 資訊提供。 <a href="#">瞭解詳情</a>

函式和公式

在試算表中加入公式和函式  
在試算表中查看加總和平均值  
參閱其他工作表的資料  
[Google 試算表函式清單](#)

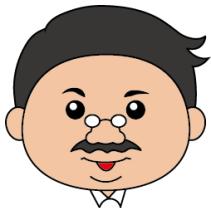




## 2-3-2 資料處理

- 在選定的儲存格右下角有一個小黑點，稱為控制點，可以拖曳填滿週邊的儲存格。
- 利用控制點將計算式複製到其他儲存格時，計算式內的儲存格會自動依相對位置改變。





## 2-3-2 資料處理

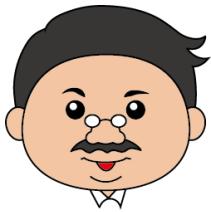
- 以向下拖曳為例，原本的  $=\sqrt{((O2-H2)^2+(P2-I2)^2)}$  會變成  $=\sqrt{((O3-H3)^2+(P3-I3)^2)}$ 。

The screenshot shows a Microsoft Excel spreadsheet titled "無標題的試算表". The formula  $=\sqrt{((O3-H3)^2+(P3-I3)^2)}$  is entered in cell K1. A red arrow points from the formula bar to the formula in cell K1. Another red arrow points from the formula in K1 to the formula in cell K2. A third red arrow points from the formula in K2 down to the formula in cell K3. A fourth red arrow points from the formula in K3 down to the formula in cell K4. A fifth red arrow points from the formula in K4 down to the formula in cell K5. A sixth red arrow points from the formula in K5 down to the formula in cell K6. A seventh red arrow points from the formula in K6 down to the formula in cell K7. A eighth red arrow points from the formula in K7 down to the formula in cell K8. A ninth red arrow points from the formula in K8 down to the formula in cell K9. A tenth red arrow points from the formula in K9 down to the formula in cell K10. A eleventh red arrow points from the formula in K10 down to the formula in cell K11. A twelfth red arrow points from the formula in K11 down to the formula in cell K12. A thirteenth red arrow points from the formula in K12 down to the formula in cell K13. A fourteenth red arrow points from the formula in K13 down to the formula in cell K14. A fifteenth red arrow points from the formula in K14 down to the formula in cell K15. A sixteenth red arrow points from the formula in K15 down to the formula in cell K16. A seventeenth red arrow points from the formula in K16 down to the formula in cell K17. A eighteenth red arrow points from the formula in K17 down to the formula in cell K18. A nineteenth red arrow points from the formula in K18 down to the formula in cell K19. A twentieth red arrow points from the formula in K19 down to the formula in cell K20. A twenty-first red arrow points from the formula in K20 down to the formula in cell K21. A twenty-second red arrow points from the formula in K21 down to the formula in cell K22. A twenty-third red arrow points from the formula in K22 down to the formula in cell K23. A twenty-fourth red arrow points from the formula in K23 down to the formula in cell K24. A twenty-fifth red arrow points from the formula in K24 down to the formula in cell K25. A twenty-sixth red arrow points from the formula in K25 down to the formula in cell K26. A twenty-seventh red arrow points from the formula in K26 down to the formula in cell K27. A twenty-eighth red arrow points from the formula in K27 down to the formula in cell K28. A twenty-ninth red arrow points from the formula in K28 down to the formula in cell K29. A thirtieth red arrow points from the formula in K29 down to the formula in cell K30. A thirty-first red arrow points from the formula in K30 down to the formula in cell K31. A thirty-second red arrow points from the formula in K31 down to the formula in cell K32. A thirty-third red arrow points from the formula in K32 down to the formula in cell K33. A thirty-fourth red arrow points from the formula in K33 down to the formula in cell K34. A thirty-fifth red arrow points from the formula in K34 down to the formula in cell K35. A thirty-sixth red arrow points from the formula in K35 down to the formula in cell K36. A thirty-seventh red arrow points from the formula in K36 down to the formula in cell K37. A thirty-eighth red arrow points from the formula in K37 down to the formula in cell K38. A thirty-ninth red arrow points from the formula in K38 down to the formula in cell K39. A forty-red arrow points from the formula in K39 down to the formula in cell K40. A forty-one-red arrow points from the formula in K40 down to the formula in cell K41. A forty-two-red arrow points from the formula in K41 down to the formula in cell K42. A forty-three-red arrow points from the formula in K42 down to the formula in cell K43. A forty-four-red arrow points from the formula in K43 down to the formula in cell K44. A forty-five-red arrow points from the formula in K44 down to the formula in cell K45. A forty-six-red arrow points from the formula in K45 down to the formula in cell K46. A forty-seven-red arrow points from the formula in K46 down to the formula in cell K47. A forty-eight-red arrow points from the formula in K47 down to the formula in cell K48. A forty-nine-red arrow points from the formula in K48 down to the formula in cell K49. A fifty-red arrow points from the formula in K49 down to the formula in cell K50. A fifty-one-red arrow points from the formula in K50 down to the formula in cell K51. A fifty-two-red arrow points from the formula in K51 down to the formula in cell K52. A fifty-three-red arrow points from the formula in K52 down to the formula in cell K53. A fifty-four-red arrow points from the formula in K53 down to the formula in cell K54. A fifty-five-red arrow points from the formula in K54 down to the formula in cell K55. A fifty-six-red arrow points from the formula in K55 down to the formula in cell K56. A fifty-seven-red arrow points from the formula in K56 down to the formula in cell K57. A fifty-eight-red arrow points from the formula in K57 down to the formula in cell K58. A fifty-nine-red arrow points from the formula in K58 down to the formula in cell K59. A六十-red arrow points from the formula in K59 down to the formula in cell K60. A六十-one-red arrow points from the formula in K60 down to the formula in cell K61. A六十-two-red arrow points from the formula in K61 down to the formula in cell K62. A六十-three-red arrow points from the formula in K62 down to the formula in cell K63. A六十-four-red arrow points from the formula in K63 down to the formula in cell K64. A六十-five-red arrow points from the formula in K64 down to the formula in cell K65. A六十六-red arrow points from the formula in K65 down to the formula in cell K66. A六十七-red arrow points from the formula in K66 down to the formula in cell K67. A六十八-red arrow points from the formula in K67 down to the formula in cell K68. A六十九-red arrow points from the formula in K68 down to the formula in cell K69. A七十-red arrow points from the formula in K69 down to the formula in cell K70. A七十-one-red arrow points from the formula in K70 down to the formula in cell K71. A七十-two-red arrow points from the formula in K71 down to the formula in cell K72. A七十三-red arrow points from the formula in K72 down to the formula in cell K73. A七十四-red arrow points from the formula in K73 down to the formula in cell K74. A七十五-red arrow points from the formula in K74 down to the formula in cell K75. A七十六-red arrow points from the formula in K75 down to the formula in cell K76. A七十七-red arrow points from the formula in K76 down to the formula in cell K77. A七十八-red arrow points from the formula in K77 down to the formula in cell K78. A七十九-red arrow points from the formula in K78 down to the formula in cell K79. A八十-red arrow points from the formula in K79 down to the formula in cell K80. A八十-one-red arrow points from the formula in K80 down to the formula in cell K81. A八十二-red arrow points from the formula in K81 down to the formula in cell K82. A八十三-red arrow points from the formula in K82 down to the formula in cell K83. A八十四-red arrow points from the formula in K83 down to the formula in cell K84. A八十五-red arrow points from the formula in K84 down to the formula in cell K85. A八十六-red arrow points from the formula in K85 down to the formula in cell K86. A八十七-red arrow points from the formula in K86 down to the formula in cell K87. A八十八-red arrow points from the formula in K87 down to the formula in cell K88. A八十九-red arrow points from the formula in K88 down to the formula in cell K89. A九十-red arrow points from the formula in K89 down to the formula in cell K90. A九十-one-red arrow points from the formula in K90 down to the formula in cell K91. A九十二-red arrow points from the formula in K91 down to the formula in cell K92. A九十三-red arrow points from the formula in K92 down to the formula in cell K93. A九十四-red arrow points from the formula in K93 down to the formula in cell K94. A九十五-red arrow points from the formula in K94 down to the formula in cell K95. A九十六-red arrow points from the formula in K95 down to the formula in cell K96. A九十七-red arrow points from the formula in K96 down to the formula in cell K97. A九十八-red arrow points from the formula in K97 down to the formula in cell K98. A九十九-red arrow points from the formula in K98 down to the formula in cell K99. A八十九-red arrow points from the formula in K99 down to the formula in cell K100.

K	dis
	0.02373277688
	123.8470262
	123.8353071
	123.8262681
	123.8221221
	123.829813
	123.8405997
	123.8619234
	123.807219

K	lat	lon	breau	dis
1	24.9264036	121.2834587	八德分局	0.02373277688
2	24.9534034	121.3071043	八德分局	123.8470262
3	24.9418178	121.2975227	八德分局	123.8353071
4				

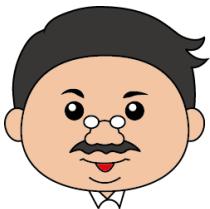




## 2-3-2 資料處理

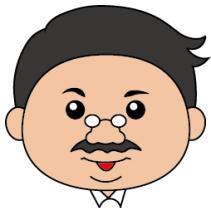
- 然而，在本例中，我們並不希望O2及P2儲存格的計算位置被自動調整，因此計算式中的O2及P2應該是固定不變的。
- 為了讓試算表知道拖曳控制點填滿時，不要自動調整某些儲存格位置，我們會在不調整的欄號或列號加上錢字號\$。





## 2-3-2 資料處理

- 例如：在本例向下填滿時，我們希望列號不要自動調整，便會在列號2前面加上錢字號\$，形成 =sqrt((O\$2-H2)\*(O\$2-H2)+(P\$2-I2)\*(P\$2-I2))。
- 此時，再向下填滿，就不會自動調整列號了。



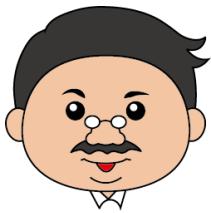
## 2-3-2 資料處理

	H	I	J	K
1	lat	lon	breau	dis
2	24.9264036	121.2834587	八德分局	0.02373277688
3	24.9534034	121.3071043	八德分局	0.05259785366
4	24.9418178	121.2975227	八德	
5	24.9254825	121.2916526	八德	
6	24.92546	121.2874246	八德	
7	24.926193	121.2951256	八德	
8	24.9717124	121.2967754	八德	
9	24.9614635	121.3206553	八德	
10	24.9521032	121.2667309	八德	

在儲存格位址前面加上「\$」，公式就不會根據相對位置調整參照，這種加上「\$」的儲存格位址，就稱作「絕對參照」，絕對參照可以只固定欄或只固定列，沒有固定的部分，仍然會依據相對位置調整參照

	H	I	J	K
1	lat	lon	breau	dis
2	24.9264036	121.2834587	八德分局	0.02373277688
3	24.9534034	121.3071043	八德分局	0.05259785366
4	24.9418178	121.2975227	八德分局	0.03933971800
5	24.9254825	121.2916526	八德分局	0.03197334660
6	24.92546	121.2874246	八德分局	0.02779784677

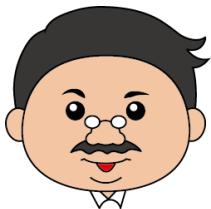




## 2-3-3 資料分析

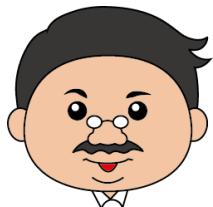
- 在資料分析基本演算法中介紹了k-近鄰演算法，在此可以使用試算表計算出待評估點最多的近鄰。
- 首先，我們在dis欄位的右邊新增一個rank的欄位，用於排名。





## 2-3-3 資料分析

- 第一筆資料對應的儲存格輸入「=RANK(K2,K:K,1)」公式，即K2在K欄位中的遞增排名，計算出距離排名，在本例中即0.02373278在dis欄位中的名次。
- 使用控制點往下自動填入，就可以得到所有資料依距離的排名。



## 2-3-3 資料分析

	H	I	J	K	L	M
1	lat	lon	breau	dis	rank	
2	24.9264036	121.2834587	八德分局	0.023732776	=rank(K2,K:K,1)	
3	24.9534034	121.3071043	八德分局	0.052597853		

	F	G	H	I	J	K	L
1	breau	station	lat	lon	breau	dis	rank
2	八德分局	八德所	24.9264036	121.2834587	八德分局	0.023732776	3
3	八德分局	四維所	24.9534034	121.3071043	八德分局	0.052597853	33
4	八德分局	八德所	24.9418178	121.2975227	八德分局	0.039339718	25
5	八德分局	八德所	24.92546	121.2874246	八德分局	0.027797846	4
6	八德分局	八德所	24.9254825	121.2916526	八德分局	0.031973346	13
7	八德分局	八德所	24.926193	121.2951256	八德分局	0.035331303	17
8	八德分局	四維所	24.9717124	121.2967754	八德分局	0.055608941	35
9	八德分局	大安所	24.9614635	121.3206553	八德分局	0.068330207	43
10	八德分局	高明所	24.9521032	121.2667309	八德分局	0.023105334	2
11	八德分局	四維所			八德分局	123.7961732	58
12	八德分局	八德所	24.9249914	121.2878889	八德分局	0.028335081	5
13	八德分局	廣興所	24.9741694	121.2232581	八德分局	0.057453486	37
14	八德分局	四維所	24.9841331	121.3100923	八德分局	0.073753854	45

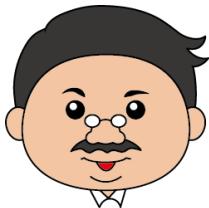


2-1

2-2

2-3

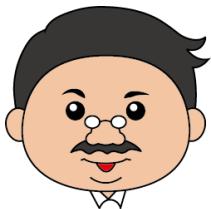
2-4



## 2-3-3 資料分析

### RANK函數

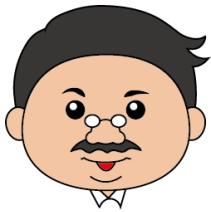
- 功能：傳回資料集中指定值的排名。
- 語法：**RANK(值, 資料, [遞增])**
  - 值：系統會計算此值的排名。
  - 資料：要列入計算的資料集所屬陣列或範圍。
  - 遷增(選用)：要以遞減還是遞增順序計算「資料」中的值。
- 範例：
  - **RANK(A2,A2:A100)**
  - **RANK(4,A2:A100,1)**



## 2-3-3 資料分析

- 利用COUNTIFS函式來計算各類別距離排名10名內的資料點數量，亦即 $k=10$ 的k-近鄰方法。
- 我們在N5到N7中分別填入三個類群的名稱，亦即八德分局、大溪分局及桃園分局。

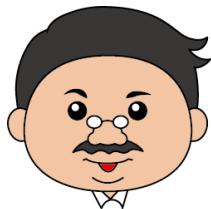




## 2-3-3 資料分析

- 在八德分局的左方M5儲存格中，使用  
 $=COUNTIFS(J:J,"="&N5, L:L,"<10")$ 計算式，計算「在J欄位中與N5儲存格相同」而且「在L欄位中其值小於10」的儲存格數量。
- 利用控制點將M5往下自動填入到M7，就可以計算出各類群排名10名內的資料點數量了。

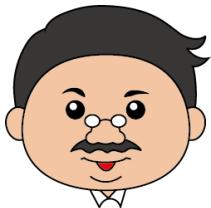




## 2-3-3 資料分析

	J	K	L	M	N
1	breau	dis	rank		
2	八德分局	0.0237327763			待評估
3	八德分局	0.05259785333			
4	八德分局	0.03933971825			
5	八德分局	0.0277978464		6	八德分局
6	八德分局	0.03197334613		3	大溪分局
7	八德分局	0.03533130317		0	桃園分局
8	八德分局	0.05560894135			
9	八德分局	0.06833020743			最大值
10	八德分局	0.0231053342		6	八德分局

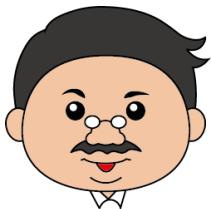




## 2-3-3 資料分析

### COUNTIFS函數

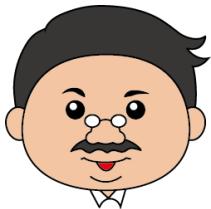
- 功能：傳回根據多個條件而得出的範圍大小。
- 語法：**COUNTIFS(條件範圍1, 條件1, [條件範圍2, 條件2, ...])**
  - 條件範圍1：核對條件1的範圍。
  - 條件1：要套用到條件範圍的模式或測試。
  - 條件範圍2：選用，其他要比對的範圍，可重複出現。
  - 條件2：選用，其他要比對的條件，可重複出現。



## 2-3-3 資料分析

### ■ 範例：

- **COUNTIFS(A1:A10, ">20", B1:B10, "<30")**
- **COUNTIFS(A7:A24, ">6", B7:B24, "<"&DATE(1969,7,20))**
- **COUNTIFS(B8:B27, ">" & B12, C8:C27, "<" & C13, D8:D27, “<>10”)**

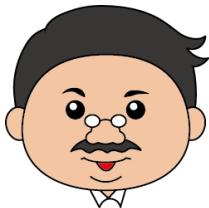


## 2-3-3 資料分析

- 使用 $=MAX(M5:M7)$ 計算式，找出M5至M7三個儲存格中的最大值，亦即各類群中最多的數量。

	J	K	L	M	N
1	breau	dis	rank		
2	八德分局	0.0237327763			待評估
3	八德分局	0.05259785333			
4	八德分局	0.03933971825			計數
5	八德分局	0.0277978464		6	八德分局
6	八德分局	0.03197334613		3	大溪分局
7	八德分局	0.03533130317		0	桃園分局
8	八德分局	0.05560894135			
9	八德分局	0.06833020743			
10	八德分局	0.0231053342		6	八德分局

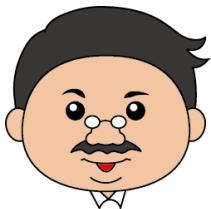




## 2-3-3 資料分析

### MAX函數

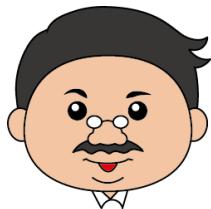
- 功能：傳回數字資料集中的最大值。
- 語法：**MAX(值1, [值2, ...])**
  - 值1：計算眾數時要列入計算的第一個值或範圍。
  - 值2：計算最大值時要列入計算的其他值或範圍。
- 範例：
  - **MAX(A2:A100,B2:B100,4,26)**
  - **MAX(1,2,3,4,5,C6:C20)**



## 2-3-3 資料分析

- 使用 `=VLOOKUP(M10, M5:N7, 2, FALSE)`，找出對應最大值的類群名稱。
- 利用此函式，可以協助找出對應M10儲存格的值6，右方所對應的群組名，在本例中為八德分局。



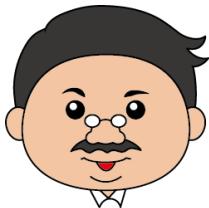


## 2-3-3 資料分析

The screenshot shows a Microsoft Excel spreadsheet with data in columns J through P. The formula bar displays the formula `=vlookup(M10, M5:N7, 2, false)`. The cell M10 contains the value 6, which is highlighted with a red dashed border. A tooltip for this cell shows the value "八德分局". The cell N10 contains the formula `=vlookup(M10, M5:N7, 2, false)`, which is highlighted with an orange dashed border. A tooltip for this cell shows the value "八德分局". The data in columns J and K includes the word "breau" and the letters "dis" respectively. Column L contains the word "rank". Columns M and N contain numerical values. Column O contains the word "lat". Column P contains the word "lon". Row 2 contains the value "待評估". Row 4 contains the word "計數". Rows 5 through 9 show the results of the VLOOKUP function for the value 6 in column M, returning the corresponding values from column N: 八德分局, 大溪分局, and 桃園分局.

	J	K	L	M	N	O	P
1	breau	dis	rank			lat	lon
2	八德分局	0.0237327763			待評估	24.93	121.26
3	八德分局	0.05259785333					
4	八德分局	0.03933971625		計數			
5	八德分局	0.0277978464		6	八德分局		
6	八德分局	0.03197334613		3	大溪分局		
7	八德分局	0.03533130317		0	桃園分局		
8	八德分局	0.05560894135		最大值	八德分局		
9	八德分局	0.06833020743					
10	八德分局	0.0231053342		6	=vlookup(M10, M5:N7, 2, false)		



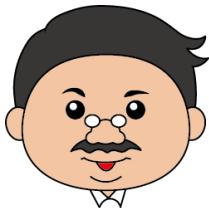


## 2-3-3 資料分析

### VLOOKUP函數

- 功能：垂直查詢。縱向搜尋特定範圍中的第一欄是否有指定準則，並將找到的資料列中指定儲存格的值傳回。
- 語法：**VLOOKUP(搜尋詞, 範圍, 索引, [已排序])**
  - 搜尋詞：要搜尋的值。
  - 範圍：要搜尋的範圍。
  - 索引：代表要傳回的值所屬的欄索引。指定範圍中的第一欄編號為1。
  - 如果索引值並非介於 1 和範圍指定的欄數之間，系統會傳回 #VALUE!。
  - 已排序：預設為TRUE，代表是否對搜尋的欄進行排序。
- 範例：
  - **VLOOKUP(10003, A2:B26, 2, FALSE)**

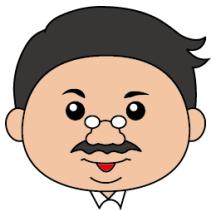




## 2-3-3 資料分析

- 利用這樣的試算表，只要修改待評估的座標值，就能依據k-近鄰方法重新計算其所屬的類群。
- 修改待評估點為(24.91, 121.29)，就會重新計算出最大的近鄰是大溪分局了。





## 2-3-3 資料分析

- 最後，若要在試算表上以統計圖顯示資料點，可以選取bread，lat，及lon欄位繪製泡泡圖。

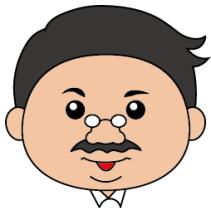


2-1

2-2

2-3

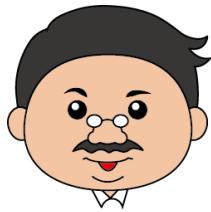
2-4



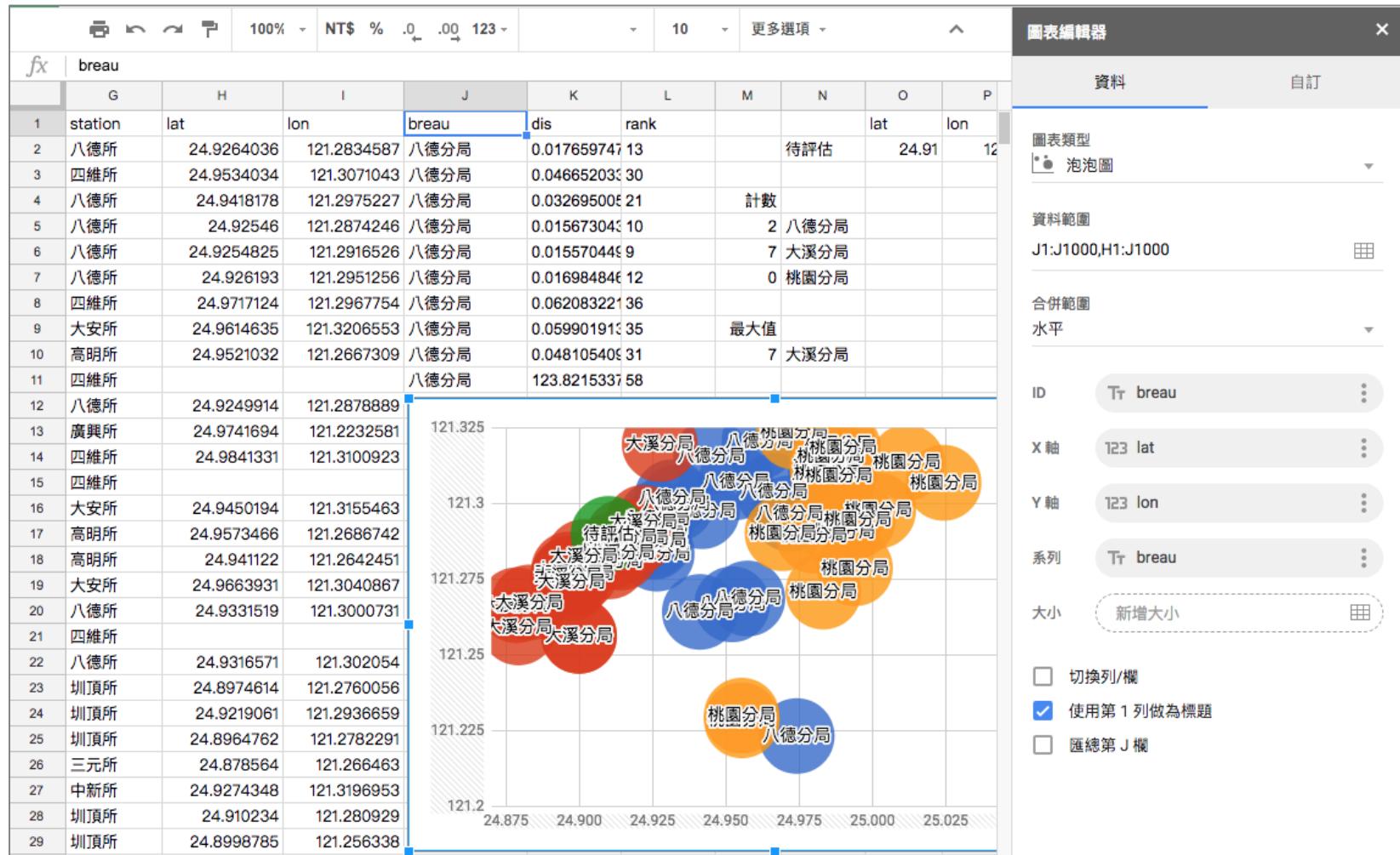
## 2-3-3 資料分析

- 剛開始的繪製結果可能不符合需求，可以在圖表編輯器中，將「系列」設定資料點的類別欄位(在本例中是bread欄位)，就可以在圖表中將每一類別的資料點以不同顏色標示出來。
- 再透過「大小」的欄位設定，可以設定每一個資料點的顯示大小。





## 2-3-3 資料分析

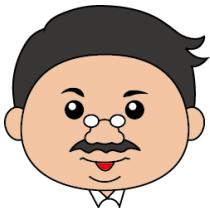


2-1

2-2

2-3

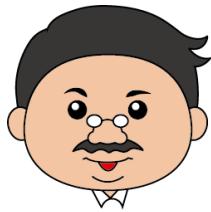
2-4



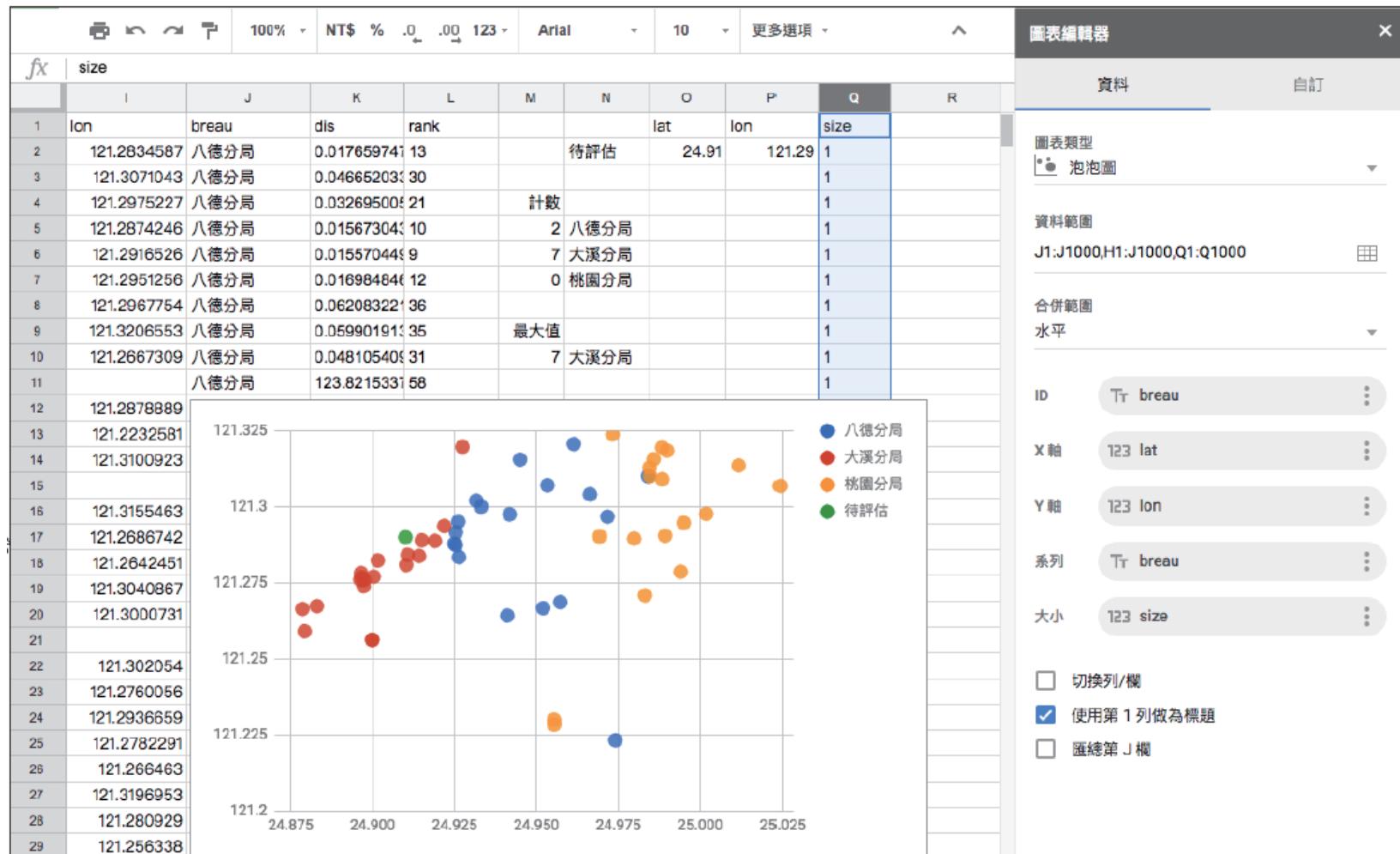
## 2-3-3 資料分析

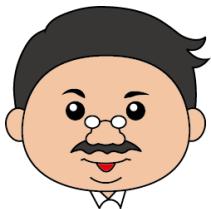
- 在資料的最右邊，我們可以新增一個size欄位用來設定資料點的大小。
- 在本例中，size欄位的內容值為1，亦即資料點的大小為1。然後將圖表編輯器的「大小」設定為size欄位。





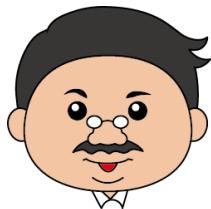
## 2-3-3 資料分析





## 2-3-4 資料篩選

- 試算表軟體內建的資料篩選器功能，使得試算表除了可以處理及分析資料外，亦能扮演資料庫的角色，進行資料擷取。
- 點選功能表單上的「資料」功能，就可以找到「篩選器」選項，點選後便可啟動篩選器。



## 2-3-4 資料篩選

The screenshot shows a Microsoft Excel spreadsheet with the following data:

	A	B	C	D
1	type	time	year	month
2	汽車竊盜	1060519	106	
3	汽車竊盜	1060519	106	
4	汽車竊盜	1060425	106	
5	汽車竊盜	1060420	106	
6	汽車竊盜	1060420	106	
7	汽車竊盜	1060418	106	
8	汽車竊盜	1060416	106	
9	汽車竊盜	1060410	106	
10	汽車竊盜	1060403	106	
11	汽車竊盜	1060316	106	

The 'Data' tab is selected in the ribbon. A context menu is open over the first row, listing options: '依 G 欄排序工作表 (A → Z)', '依 G 欄排序工作表 (Z → A)', '排序範圍...', '隨機範圍', '已命名範圍...', '受保護的工作表和範圍...', '將文字分隔成不同欄...', '筛选器' (highlighted with a red box), and '筛选器檢視畫面...'. The '筛选器' option is the fourth item from the top.

	H	I
lat	24.9264036	121.28349
	24.9534034	121.30710
	24.9418178	121.29752
	24.92546	121.28742
	24.9254825	121.29165
	24.926193	121.29512
	24.9717124	121.29671
	24.9614635	121.32065
	24.9521032	121.26673

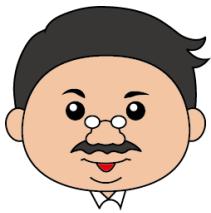


2-1

2-2

2-3

2-4



## 2-3-4 資料篩選

- 篩選器啟動後，名稱欄位中就會出現「」下拉式選單符號，要根據哪個欄位篩選資料，就點選欄位中的該符號即可。

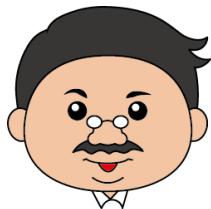


2-1

2-2

2-3

2-4



## 2-3-4 資料篩選

	A	B	C	D	E	F	G	H
1	type	time	yea	mo	dat	brea	station	lat
2	汽車竊盜	1060519	106	106	排序 (A → Z)			24.9264036
3	汽車竊盜	1060519	106	106	排序 (Z → A)			24.9534034
4	汽車竊盜	1060425	106	106	依條件篩選...			24.9418178
5	汽車竊盜	1060420	106	106	依值篩選...			24.92546
6	汽車竊盜	1060420	106	106	全部選取 - 清除			24.9254825
7	汽車竊盜	1060418	106	106				24.926193
8	汽車竊盜	1060416	106	106				24.9717124
9	汽車竊盜	1060410	106	106				24.9614635
10	汽車竊盜	1060403	106	106				24.9521032
11	汽車竊盜	1060316	106	106				
12	汽車竊盜	1060309	106	106				24.9249914
13	汽車竊盜	1060304	106	106				24.9741694
14	汽車竊盜	1060228	106	106				24.9841331
15	汽車竊盜	1060221	106	106				
16	汽車竊盜	1060208	106	106				24.9450194
17	汽車竊盜	1060206	106	106				24.9573466
18	汽車竊盜	1060204	106	106				24.941122
19	汽車竊盜	1060106	106	106				24.9663931
20	汽車竊盜	1060104	106	106				24.9331519
21	汽車竊盜	1060103	106	106				

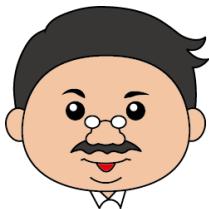


2-1

2-2

2-3

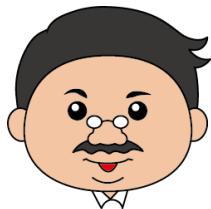
2-4



## 2-3-4 資料篩選

- 篩選器提供了排序、依條件篩選及依值篩選等功能，在本例中，勾選特定值（如：八德所），然後依值篩選，則試算表便會將欄位中具有該值的各列顯示出來。

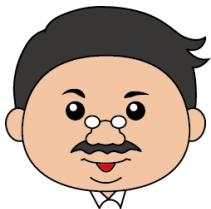




## 2-3-4 資料篩選

	A	B	C	D	E	F	G	H
1	type	time	yea	mo	dat	breau	station	lat
2	汽車竊盜	1060519	106	5	19	八德分局	八德所	24.9264036
4	汽車竊盜	1060425	106	4	25	八德分局	八德所	24.9418178
5	汽車竊盜	1060420	106	4	20	八德分局	八德所	24.92546
6	汽車竊盜	1060420	106	4	20	八德分局	八德所	24.9254825
7	汽車竊盜	1060418	106	4	18	八德分局	八德所	24.926193
12	汽車竊盜	1060309	106	3	9	八德分局	八德所	24.9249914
20	汽車竊盜	1060104	106	1	4	八德分局	八德所	24.9331519
22	汽車竊盜	1051206	105	12	6	八德分局	八德所	24.9316571
..								





## 2-3-5 資料透視表

- 資料透視表可以用來將龐大的資料精簡顯示，或是分析各資料點之間的關係。
- 點選功能表中的「資料→資料透視表」選項，會在原活頁簿後新增一個樞紐分析表活頁簿。

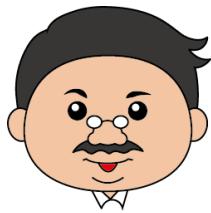


2-1

2-2

2-3

2-4



## 2-3-5 資料透視表

Screenshot of the Microsoft Excel PivotTable Editor dialog box.

The dialog shows the formula bar: '工作表1'!A1:Q63

The main area displays the PivotTable structure:

	A	B	C	D	E
1	欄				
2	列	值			
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					

The right pane shows recommendations and new items:

- 建議使用
  - 各「station」的「time」平均
  - 各「breau」的「date」總和
  - 各「breau」的「lat」平均
- 新增列
- 新增欄
- 新增值
- 新增篩選器

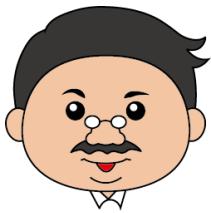


2-1

2-2

2-3

2-4



## 2-3-5 資料透視表

- 我們要計算各分局的所有點位之lon及lat平均值，可以先點選「列→新增」，篩選出分局名稱。

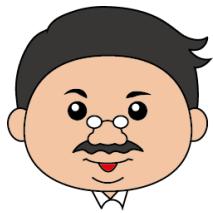


2-1

2-2

2-3

2-4



## 2-3-5 資料透視表

The screenshot shows a Microsoft Excel spreadsheet and its corresponding PivotTable Editor dialog box.

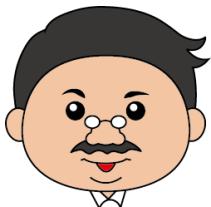
**工作表数据:**

	A	B	C
1	bureau		
2	八德分局		
3	大溪分局		
4	桃園分局		
5	總和		
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			

**資料透視表編輯器 (PivotTable Editor) 对话框:**

- 工作表:** '工作表1'!A1:Q63
- 建議使用:** (下拉菜单)
- 列 (Column):** 新增  
- 属性: bureau  
  - 排序: 递增  
  - 排序依据: bureau  
  - 显示总计 (显示总计):
- 欄 (Row):** 新增
- 值 (Value):** 新增
- 篩選器 (Filter):** 新增

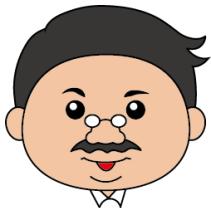




## 2-3-5 資料透視表

- 點選「值→新增」將lon及lat的資料匯總出來，預設的匯總資料為總和(SUM)。
- 若點選「匯總依據」可以修改為平均(AVERAGE)，完成我們預計要分析的平均值資訊。





# 2-3-5 資料透視表

	A	B	C
1	breau	lat的SUM	lon的SUM
2	八德分局	449.0420234	2183.222369
3	大溪分局	497.9705376	2425.596981
4	桃園分局	474.6931613	2304.428121
5	總和	1421.705722	6913.24747
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			
21			
22			
23			
24			
25			
26			
27			

資料透視表編輯器

'工作表1'!A1:Q63

建議使用

列 新增

breau

排序：排序依據：遞增 breau

顯示總計

欄 新增

值 方向：欄數 新增

顯示 lat

匯總依據：顯示方式：SUM 預設

顯示 lon

匯總依據：顯示方式：SUM 預設

	A	B	C
1	breau	lat的AVERAGE	lon的AVERAGE
2	八德分局	24.94677908	121.2901316
3	大溪分局	24.89852688	121.279849
4	桃園分局	24.98385059	121.2856906
5	總和	24.94220565	121.2850433
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			
21			
22			
23			
24			
25			
26			
27			

資料透視表編輯器

'工作表1'!A1:Q63

建議使用

列 新增

breau

排序：排序依據：遞增 breau

顯示總計

欄 新增

值 方向：欄數 新增

顯示 lat

匯總依據：顯示方式：AVERAGE 預設

顯示 lon

匯總依據：顯示方式：AVERAGE 預設

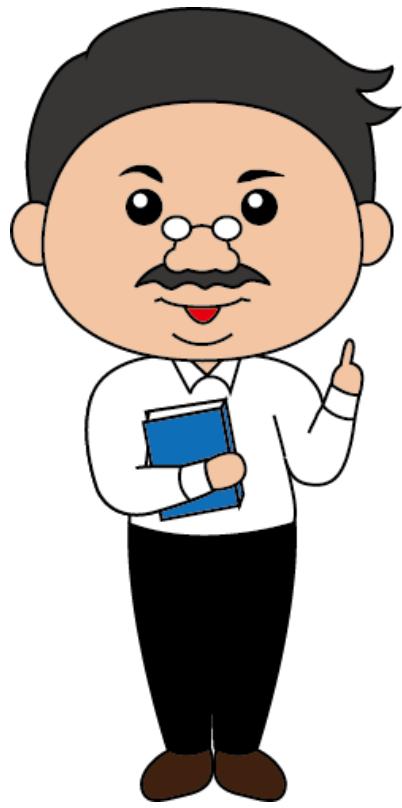


2-1

2-2

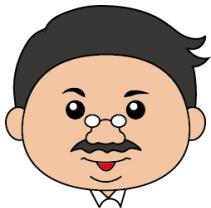
2-3

2-4



## 2-4 資料處理之軟體工具— Python

- 2-4-1 Python程式語言
- 2-4-2 線性回歸
- 2-4-3 k-近鄰分類
- 2-4-4 k-平均聚類
- 2-4-5 標註傳播法

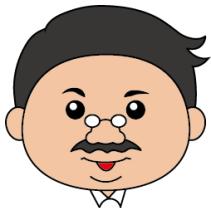


## 2-4-1 Python程式語言

### 簡介

- Python程式語言在1990年代初由Guido van Rossum創造。
- 依循GPL相容的條款進行授權使用，允許使用者在修改了Python的原始碼後重新散佈。

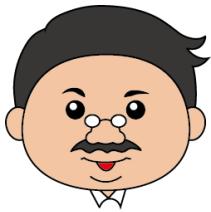




## 2-4-1 Python程式語言

- 發展期間累積了相當完整的標準套件，更有無以計數的非標準套件，能應用於系統管理、網路管理、網路傳輸程式、網頁程式開發、數值分析程式、圖形介面應用程式等方面。





## 2-4-1 Python程式語言

- 主要特點是使用方式簡單友善、幾乎可以無縫讓程式跨平台、而且是基於GPL協議下免費開放的。

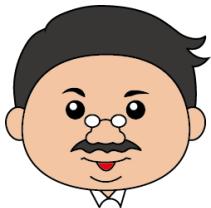


2-1

2-2

2-3

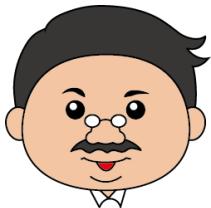
2-4



## 2-4-1 Python程式語言

### 安裝

- 在 Python 的 官 方 網 站 (<https://www.python.org>) 可以下載各種平台版本，建議安裝Python 3以上的版本。除了Python以外，有些作業系統還必須安裝pip，以便安裝其他套件。



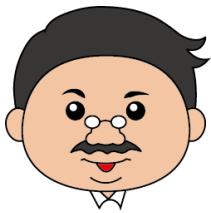
## 2-4-1 Python程式語言

- 安裝好Python，在作業系統的命令列模式下輸入python3，執行後，就會顯示Python的操作介面：

```
$ python3
Python 3.6.3 (default, Nov 16 2017, 23:15:59)
[GCC 4.2.1 Compatible Apple LLVM 8.0.0 (clang-800.0.38)] on
darwin
Type "help", "copyright", "credits" or "license" for more
information.

>>>
```



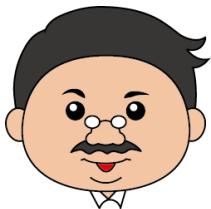


## 2-4-1 Python程式語言

- 輸入 `print("Hello, World!")`，就可以顯示其執行結果：

```
>>> print("Hello, World!")
Hello, World!
>>>
```



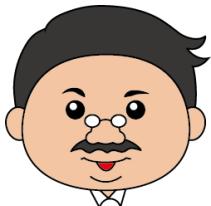


## 2-4-1 Python程式語言

- 若要離開Python，輸入exit()，就會回到作業系統的命令列提示字元：

```
>>> exit()  
$
```



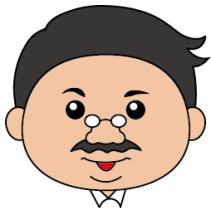


## 2-4-1 Python程式語言

- 安裝套件則使用**pip**指令，在作業系統的命令列模式下，輸入**pip3 install <套件名稱>**，就可以安裝所需的套件。
- 例如：安裝**statsmodels**套件的指令為  
**pip3 install statsmodels**。

```
$ pip3 install statsmodels
Collecting statsmodels
  Downloading statsmodels-0.8.0-cp36-cp36m-
    macosx_10_6_intel.macosx_10_9_intel.macosx_10_9_x86_
      64.macosx_10_10_intel.macosx_10_10_x86_64.whl (5.4MB)
  100% |████████████████████████████████| 5.4MB 216kB/s
```



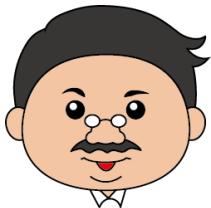


## 2-4-1 Python程式語言

- 在後面的課程內容中，還會用到**numpy**、**matplotlib**、**sklearn**等套件，可以事先利用**pip3**指令先安裝起來。
- 我們也可以將Python程式存成一般文字檔，然後在命令列模式下執行，例如：要執行**ols.py**這個程式檔，指令就是：

```
$ python3 ols.py
```



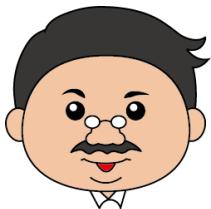


## 2-4-1 Python程式語言

### 讀取資料

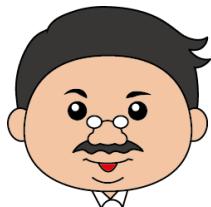
- 目前網路上常見的結構化資料格式檔是 **CSV(Comma-Separated Values)**，是一種通用的交換資料格式，以逗號將每筆資料作分隔，每筆資料為一行。





## 2-4-1 Python程式語言

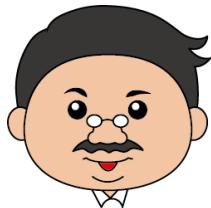
- 除了大多數的結構化資料為以CSV檔儲存外，一般的資料庫及試算表都可以讓使用者把資料用CSV格式匯出。
- 例如：以CSV檔案10606\_2.csv作為讀取目標：



## 2-4-1 Python程式語言

lat, lon	24.8792239, 121.2590988
24.8974614, 121.2760056	24.8829867, 121.2673747
24.8964762, 121.2782291	24.9003435, 121.2770788
24.9219061, 121.2936659	24.8966187, 121.2768273
24.878564, 121.266463	24.8962408, 121.2760791
24.9274348, 121.3196953	24.8972648, 121.2738993
24.910234, 121.280929	24.9077478, 121.2785517
24.9106574, 121.2841842	24.8831799, 121.2670356
24.9190081, 121.2887719	24.8866892, 121.2701116
24.9150083, 121.2889617	24.8836848, 121.2720818
24.9141703, 121.2838311	24.884841, 121.26729
24.9015767, 121.2823814	24.9025937, 121.2733802
24.8999549, 121.256327	



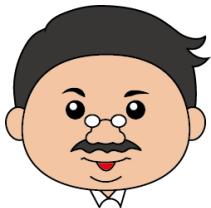


## 2-4-1 Python程式語言

- 若將numpy套件輸入Python中，可以輕易讀取CSV檔。

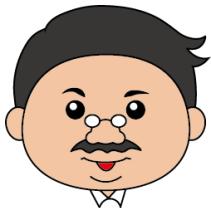
```
>>> import numpy as np  
>>> data = np.loadtxt('10606_2.csv', delimiter=",", skiprows=1)  
>>> lat_list = data[:, 0]  
>>> lon_list = data[:, 1]  
>>> lon_list  
array([ 121.2760056, 121.2782291, 121.2936659, 121.266463 ,  
       121.3196953, 121.280929 , 121.256338 , 121.2841842,  
       121.2887719, 121.2889617, 121.3108394, 121.2838311,  
       121.2823814, 121.256327 , 121.2590988, 121.2673747,  
       121.2770788, 121.2768273, 121.2760791, 121.2738993,  
       121.2785517, 121.2670356, 121.2701116, 121.2720818,  
       121.26729 , 121.2733802])  
>>>
```





## 2-4-1 Python程式語言

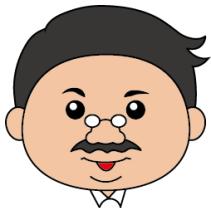
- 第一行指令將numpy套件輸入Python中，然後使用numpy所提供的loadtxt函式讀取10606\_2.csv檔案，skiprows=1參數是為了忽略第一列的標題。
- 接著lat\_list = data[:, 0]及lon\_list = data[:, 1]則分別將data陣列的第一欄及第二欄資料存進lat\_list及lon\_list的序列中。



## 2-4-1 Python程式語言

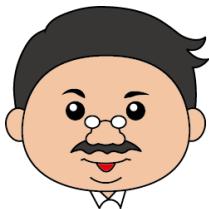
- 最後，若只輸入變數lon\_list名稱，則顯示lon\_list的內容，可以看到這是一個陣列(array)的型式及其資料。
- 當資料在平面座標上表示，然後以一直線表示資料的分佈趨勢時，就稱為線性回歸。





## 2-4-2 線性回歸

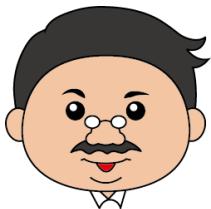
- 主要是利用線性回歸方程式對變數之間的關係進行建模的一種分析，方程式可以是一個或多個變數的線性組合。
- 只有一個變數的情況稱為簡單線性回歸，大於一個變數的情況的叫做多元線性回歸。
- 若使用非直線方程式為回歸方程式的情況，則稱為非線性回歸。



## 2-4-2 線性回歸

### 簡單線性回歸

- 簡單線性回歸的線性方程式為  $y = b_0 + b_1 x_1$ ，這是假設資料  $y_i$  與  $x_i$  有線性函數關係。



## 2-4-2 線性回歸

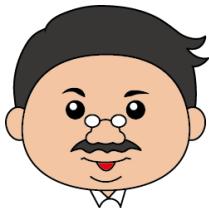
- 以矩陣方程式表示所有資料的關係為  $Y=XB+\varepsilon$ ，其中  $\varepsilon$  為誤差量，若資料有  $n$  筆，則

$$Y = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix} \quad (2-1)$$

$$X = \begin{bmatrix} 1 & x_{I0} \\ 1 & x_{I1} \\ \vdots & \vdots \\ 1 & x_{In} \end{bmatrix} \quad (2-2)$$

$$B = \begin{bmatrix} b_0 \\ b_I \end{bmatrix} \quad (2-3)$$



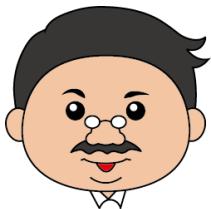


## 2-4-2 線性回歸

- 在Python中，有**StatsModels**套件設計進行統計上的資料分析，只要在程式中輸入**statsmodels.api**，就可以利用該套件進行簡單線性回歸：

```
1 import statsmodels.api as sm
2 import numpy as np
3 data = np.loadtxt('10606_2.csv', delimiter=",", skiprows=1)
4 lat_list = data[:, 0]
5 lon_list = data[:, 1]
6 X = sm.add_constant(lon_list)
7 Y = lat_list
8 lmRegModel = sm.OLS(Y, X)
9 result = lmRegModel.fit()
10 print(result.summary())
```



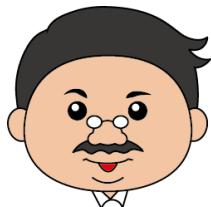


## 2-4-2 線性回歸

### 程式說明

- 第6行指令在lat\_list陣列中新增一欄常數值1形成新的兩欄陣列X，以符合陣列表示。
- 第8行指令及第9行指令根據Y矩陣及X矩陣，進行線性回歸。
- 第10行指令將結果顯出來。

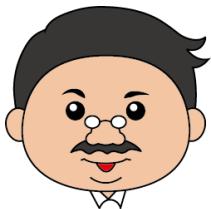




## 2-4-2 線性回歸

OLS Regression Results						
Dep. Variable:	y	R-squared:	0.694			
Model:	OLS	Adj. R-squared:	0.680			
Method:	Least Squares	F-statistic:	49.86			
Date:	Tue, 21 Nov 2017	Prob (F-statistic):	4.39e-07			
Time:	12:54:21	Log-Likelihood:	83.145			
No. Observations:	24	AIC:	-162.3			
Df Residuals:	22	BIC:	-159.9			
Df Model:	1					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
const	-85.4654	15.629	-5.468	0.000	-117.879	-53.052
x1	0.9100	0.129	7.061	0.000	0.643	1.177
Omnibus:	1.247	Durbin-Watson:	1.869			
Prob(Omnibus):	0.536	Jarque-Bera (JB):	0.906			
Skew:	0.463	Prob(JB):	0.636			
Kurtosis:	2.777	Cond. No.	1.17e+06			

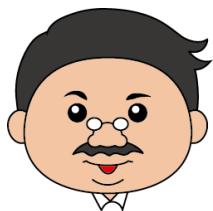




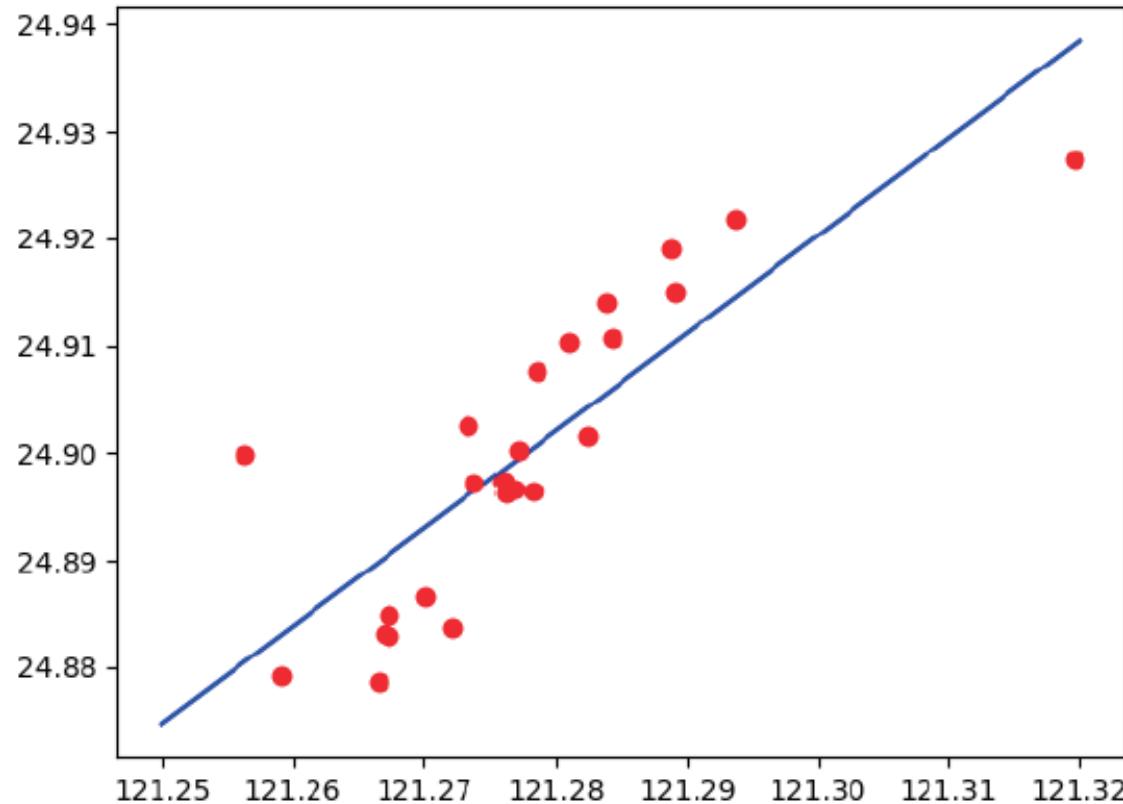
## 2-4-2 線性回歸

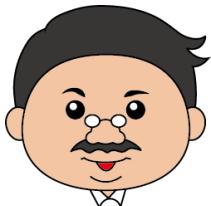
- 其中**const**對應了常數項 $b_0$ 值，而**x<sub>1</sub>**對應了方程式的**x<sub>1</sub>**係數 $b_1$ 值，所以本例中的線性回歸方程式為  $y = -85.46 + 0.91 * x_1$ ，其中**R**平方值(**R-squared**)0.69，即顯示了此迴歸方程式與原資料的差異量。





## 2-4-2 線性回歸





## 2-4-2 線性回歸

### 非線性回歸

- 非線性回歸使用非線性回歸方程式，是簡單線性回歸的擴展。方程式的選擇應
- 符合變數間相對關係，且函數形式儘可能簡單，以下列出幾種常見的非線性回歸方程式：

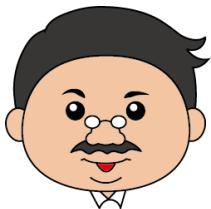
↳ 二次函數： $y=b_0+b_1 \times x_1 + b_2 \times x_1^2$

↳ 倒數函數： $y=b_0+b_1 \times \frac{1}{x_1}$

↳ 指數函數： $y=b_0+b_1^{x_1}$

↳ 對數函數： $y=b_0+b_1 \times \ln(x_1)$

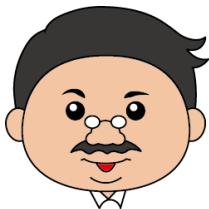




## 2-4-2 線性回歸

- 假設要利用二次函數進行非線性回歸，可以將簡單線性回歸方程式的X矩陣擴充為：

$$X = \begin{bmatrix} 1 & x_{I0} & {x_{I0}}^2 \\ 1 & x_{II} & {x_{II}}^2 \\ \vdots & \vdots & \vdots \\ 1 & x_{In} & {x_{In}}^2 \end{bmatrix} \quad (2-4)$$

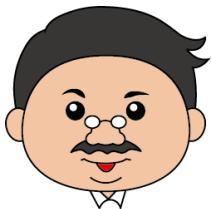


## 2-4-2 線性回歸

■ 利用Python進行非線性回歸的程式內容為：

```
1 import statsmodels.api as sm  
2 import numpy as np  
3 data = np.loadtxt('10606_2.csv', delimiter=',', skiprows=1)  
4 lat_list = data[:, 0]  
5 lon_list = data[:, 1]  
6 X = sm.add_constant(lon_list)  
7 X = np.c_[X, lon_list**2]  
8 Y = lat_list  
9 lmRegModel = sm.OLS(Y, X)  
10 result = lmRegModel.fit()  
11 print(result.params)
```





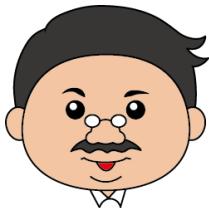
## 2-4-2 線性回歸

### 程式說明

- 第7行指令為X矩陣新增一欄，內容為lat\_list中各項的平方，以符合方程式2-4的型式。
- 第11行指令將係數 $b_0$ ， $b_1$ ，及 $b_2$ 列印出來。

```
[ -8.43085449e+04 1.38974702e+03 -5.72547388e+00]
```

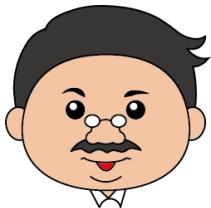




## 2-4-2 線性回歸

- Matplotlib套件提供了繪圖程式庫，可以協助將原始資料及線性回歸方程式以圖形顯示出來：

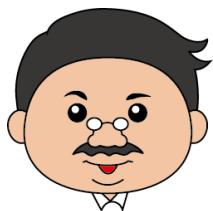
```
12 import matplotlib.pyplot as plt  
13 B = result.params  
14 px = np.linspace(121.25, 121.32, 300)  
15 X = np.c_[px, px**2]  
16 X = sm.add_constant(X)  
17 Y = X.dot(B)  
18 plt.plot(px, Y, 'b')  
19 plt.plot(lon_list, lat_list, 'ro')  
20 plt.savefig("10606_2.png")  
21 plt.show()
```



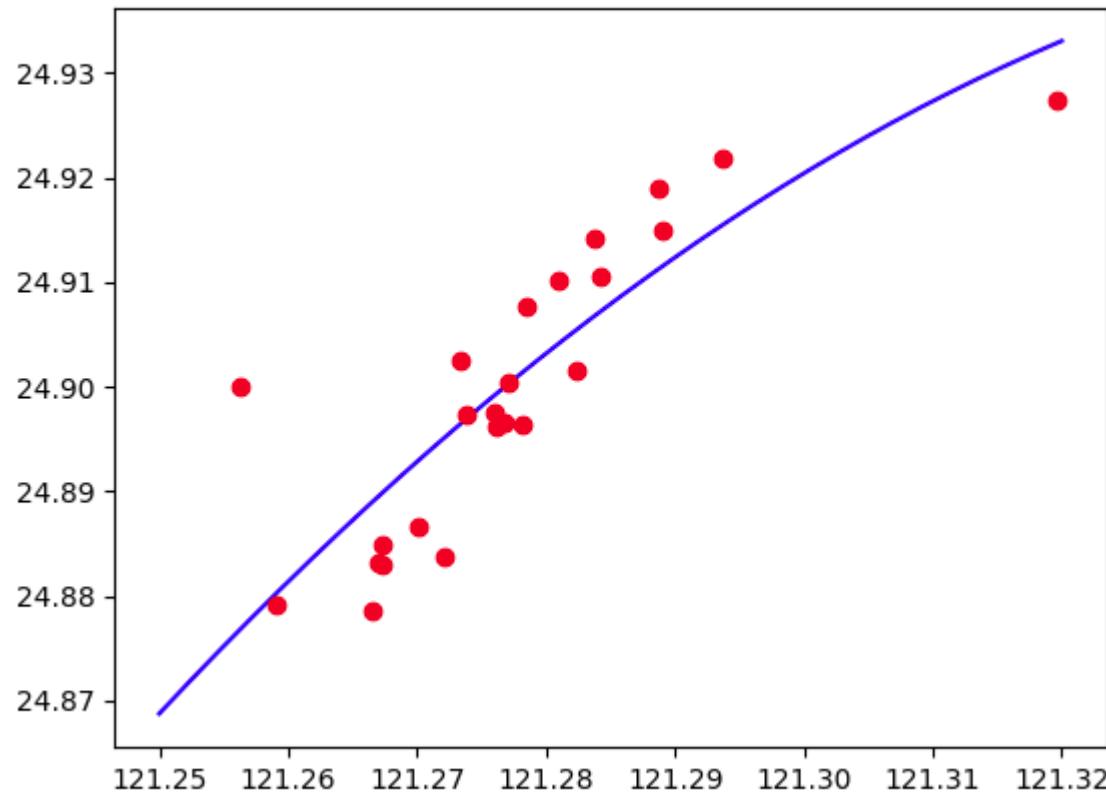
## 2-4-2 線性回歸

### 程式說明

- 第14行程式在121.25到121,32中產生了300個數字，放在px序列中。
- 第15及16行矩陣X，然後以px為輸入值產生300個py值。
- 第18行畫出回歸方程式。
- 第19行畫出原資料點。
- 第20行將圖形輸出成檔案。



## 2-4-2 線性回歸

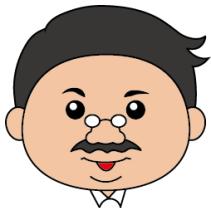


2-1

2-2

2-3

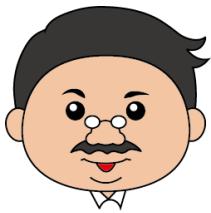
2-4



## 2-4-2 線性回歸

### 多元線性回歸

- 建立多元線性回歸模型是假設變數對結果的影響是線性的，且變數之間有一定程度的獨立不相關。
- 多元線性回歸方程式是一個多變數的線性組合方程式，若變數有m個，其形式就會是：



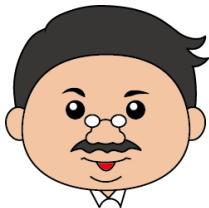
## 2-4-2 線性回歸

■ 矩陣方程式  $Y = XB + \epsilon$  中的  $X$  及  $B$  則變成：

$$X = \begin{bmatrix} 1 & x_{10} \dots x_{m0} \\ 1 & x_{11} \dots x_{m1} \\ \vdots & \vdots \ddots \vdots \\ 1 & x_{1n} \dots x_{mn} \end{bmatrix} \quad (2-5)$$

$$B = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_n \end{bmatrix} \quad (2-6)$$



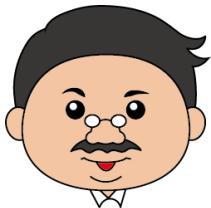


## 2-4-2 線性回歸

- 在此我們利用lat\_list及lon\_list線性合成一個已知的方程式，再使用多元線性回歸分析驗證：

```
1 import statsmodels.api as sm  
2 import numpy as np  
3 data = np.loadtxt('10606_2.csv', delimiter=',', skiprows=1)  
4 lat_list = data[:, 0]  
5 lon_list = data[:, 1]  
6 X = np.c_[lon_list, lat_list]  
7 X = sm.add_constant(X)  
8 num = lon_list.size  
9 Y = X.dot(np.array([1, 0.2, 0.8])) + np.random.random(num)/150  
10 lmRegModel = sm.OLS(Y, X)  
11 result = lmRegModel.fit()  
12 print(result.params)
```





## 2-4-2 線性回歸

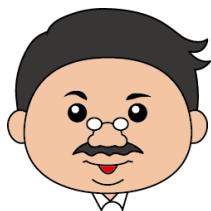
- 程式中的第9行將矩陣B設定為：

$$B = \begin{bmatrix} 1 \\ 0.2 \\ 0.8 \end{bmatrix}$$

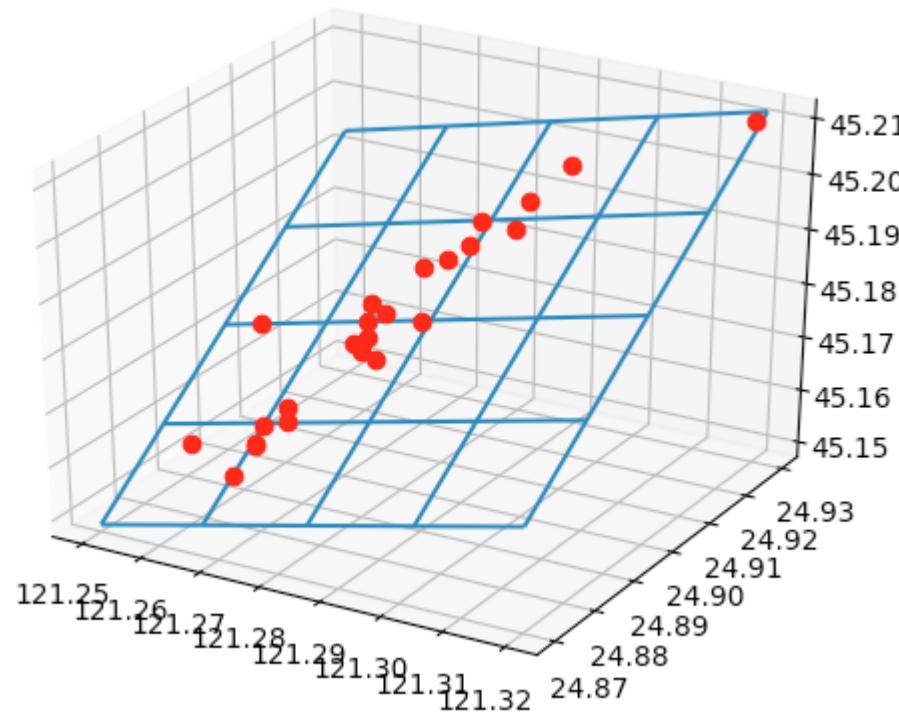
- 然後利用numpy套件所提供的矩陣運算，再加上一些亂數值產生Y矩陣。程式執行線性回歸的結果會有些許不同，但其值會接近原先設定的B值。

```
[-5.33227142 0.2617001 0.75390209]
```





## 2-4-2 線性回歸

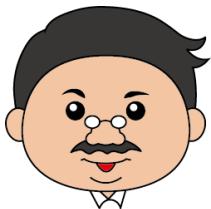


2-1

2-2

2-3

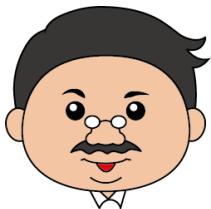
2-4



## 2-4-3 k-近鄰分類

- Python程式語言中，也有相當方便的套件Scikitlearn機器學習程式，可以用來執行k-近鄰分類。
- 首先，我們從桃園市汽車竊盜點資料中，擷取出三個分類，建立knn\_test.csv檔(…為省略顯示的資料)。

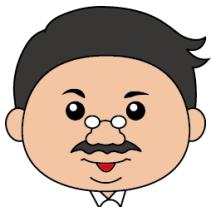




## 2-4-3 k-近鄰分類

- 在這個檔案中，第一個欄位代表資料點的已知類別，第二個欄位及第三個欄位代表資料屬性，在本例中是座標的經緯度值。





## 2-4-3

# k-近鄰分類

breau,lat,lon

1,24.9264036,121.2834587

1,24.9534034,121.3071043

...

1,24.9316571,121.302054

2,24.8974614,121.2760056

2,24.9219061,121.2936659

...

2,24.8972648,121.2738993

3,24.9845663,121.3128774

3,24.9893136,121.2905215

...

3,24.9832358,121.2709382

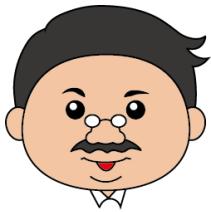


2-1

2-2

2-3

2-4

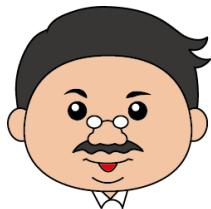


## 2-4-3 k-近鄰分類

■ 利用 Python 進行 k 近鄰計算的程式如下：

```
1 from sklearn import neighbors, datasets  
2 import numpy as np  
3  
4 #輸入已知資料  
5 data = np.loadtxt('knn_test.csv', delimiter=',', skiprows=1)  
6 X = np.c_[data[:,1], data[:,2]]  
7 Y = np.array(data[:,0])  
8 Y.astype(int)  
9 knn = neighbors.KNeighborsClassifier(5)  
10 knn.fit(X, Y)  
11  
12 #待評估資料
```

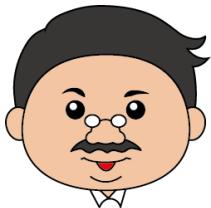




## 2-4-3 k-近鄰分類

```
13 px = np.linspace(24.87, 25.03, 20)
14 py = np.linspace(121.26, 121.26, 20)
15 target = np.c_[px, py]
16 result = knn.predict(target)
17
18 #繪圖顯示結果
19 import matplotlib.pyplot as plt
20 plt.scatter(data[:,1], data[:,2], c=Y)
21 plt.scatter(target[:,0], target[:,1], c=result, marker='x')
22 plt.savefig("knn.png")
23 plt.show()
```



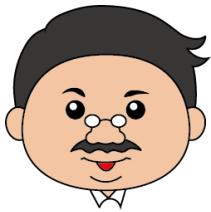


## 2-4-3 k-近鄰分類

### 程式說明

- 第6行及第7行建立了已標註的資料點，Y為標註值，也就是輸入資料的第一個欄位；X為特徵值，在本例中是輸入資料的第二及第三個欄位。
- 第9行建立k近鄰分類模型，輸入參數k = 5。

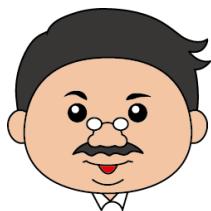




## 2-4-3 k-近鄰分類

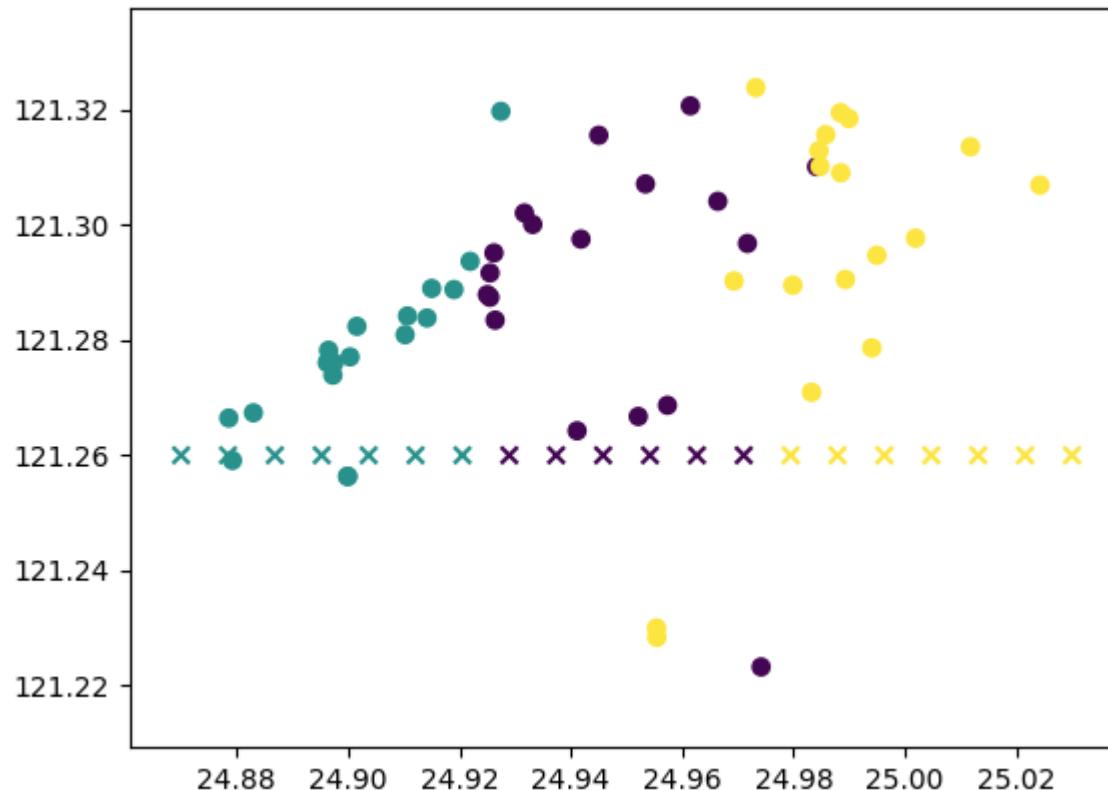
- 第13-15行產生20筆待評估資料，然後在第16行呼叫predict函數進行評估。
- 第18-19行則將結果以圖2-39顯示，圖中圓形點為原始輸入已知道類別的資料點，叉形點為產生的20筆待評估資料，其顏色顯示計算出來的近鄰類別。





2-4-3

## k-近鄰分類

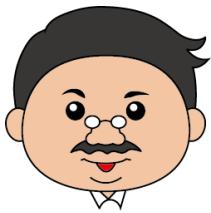


2-1

2-2

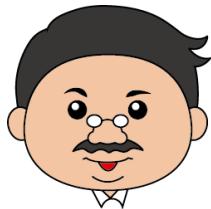
2-3

2-4



## 2-4-4 k-平均聚類

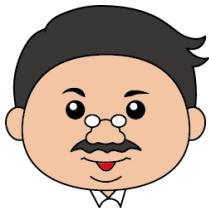
- 使用同樣的knn\_test.csv資料，若忽略原本的已知的類別，就可以利用來進行k-平均聚類，也可以看看與原本類別之差別。



## 2-4-4 k-平均聚類

```
1 from sklearn import cluster  
2 import numpy as np  
3  
4 #輸入已知資料  
5 data = np.genfromtxt('knn_test.csv', delimiter=',', skip_header=1)  
6 X = np.c_[data[:,1], data[:,2]]  
7  
8 #進行k-平均聚類  
9 kmeans_fit = cluster.KMeans(n_clusters = 3).fit(X)  
10 cluster_labels = kmeans_fit.labels_  
11  
12 #繪圖顯示結果  
13 import matplotlib.pyplot as plt  
14 plt.scatter(X[:,0], X[:,1], c=cluster_labels)  
15 plt.savefig("kmean.png")  
16 plt.show()
```



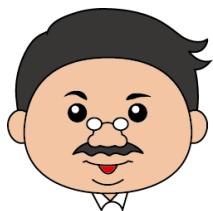


## 2-4-4 k-平均聚類

### 程式說明

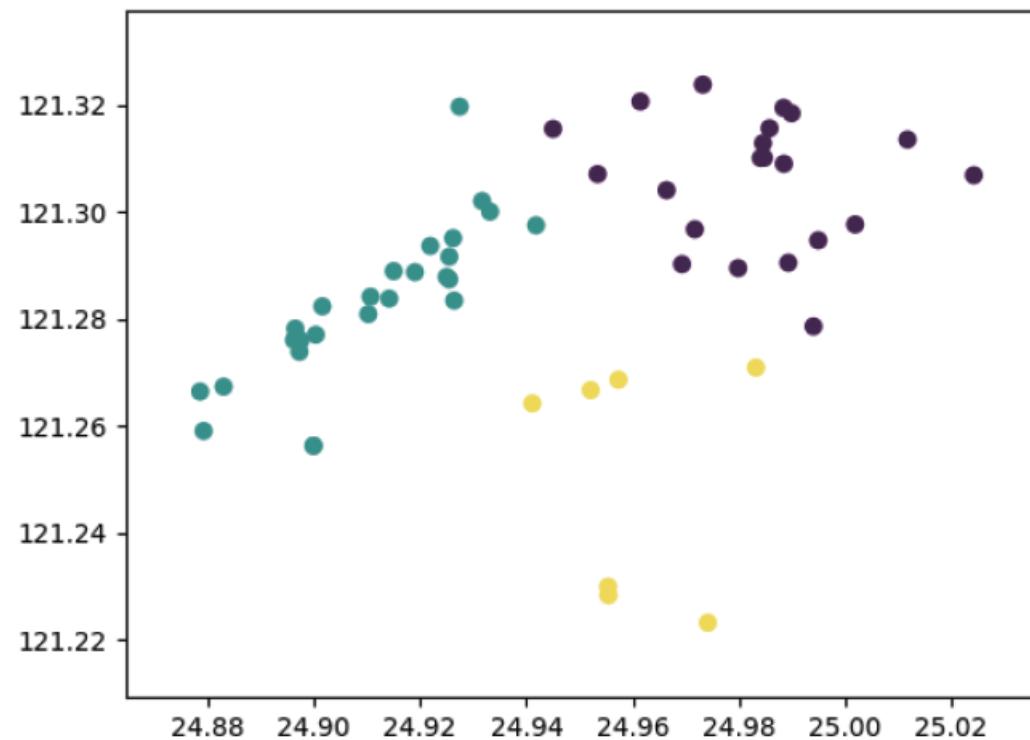
- 第9 -10行為k平均聚類，k平均聚類中所需的k值設定在n\_clusters = 3這個變數中。
- 第10行是擷取每一資料點的類別標示，其結果是一個一維序列。





2-4-4

## k-平均聚類

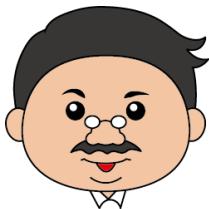


2-1

2-2

2-3

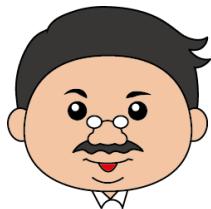
2-4



## 2-4-5 標註傳播法

■ 利用Python程式來實作標註傳播方法如下：

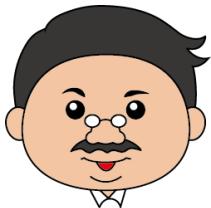
```
1 from sklearn.semi_supervised import label_propagation  
2 import scipy.sparse.csgraph  
3 import numpy as np  
4  
5 #輸入已知資料  
6 data = np.genfromtxt('knn_test.csv', delimiter=',', skip_header=1)  
7 X = np.c_[data[:,1], data[:,2]]  
8 labels = np.full(data.shape[0], -1)  
9 labels[17] = 0  
10 labels[36] = 1  
11 labels[54] = 2  
12
```



## 2-4-5 標註傳播法

```
13 #進行Label Propagation  
14 label_prop_model =  
    label_propagation.LabelSpreading()  
15 label_prop_model.fit(X, labels)  
16 cluster_labels = label_prop_model.transduction_  
17  
18 #繪圖顯示結果  
19 import matplotlib.pyplot as plt  
20 plt.scatter(X[:,0], X[:,1], c=cluster_labels)  
21 plt.savefig("labelpro.png")  
22 plt.show()
```



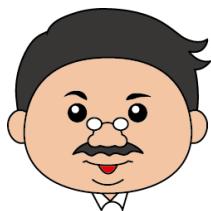


## 2-4-5 標註傳播法

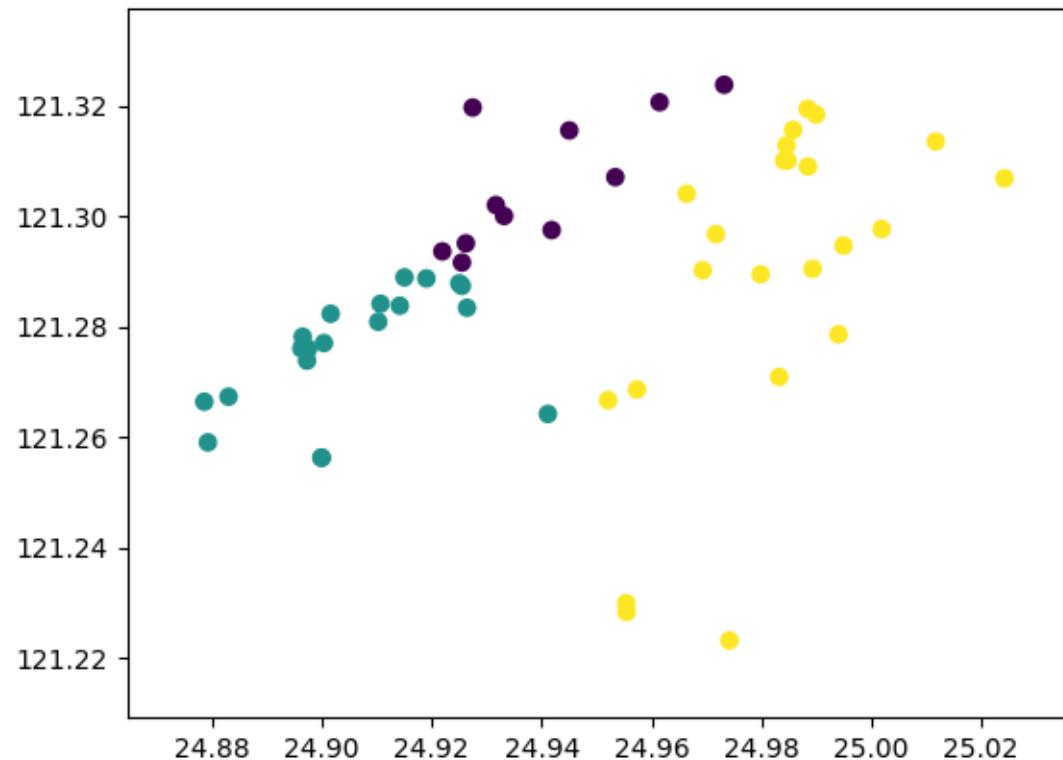
### 程式說明

- 第8行先設定所有的資料點都是未知標註，也就是設為-1。
- 第9-11行將其中三筆資料點分別給多不同的標註，也就是0，1，和2。
- 第14和15行進行標註傳播，然後取出結果放在cluster\_labels陣列中。





## 2-4-5 標註傳播法



2-1

2-2

2-3

2-4