財政部關務署工作坊

> *資料預處理 | 2024-07-19*

數聚點 | 郭耀仁 yaojenkuo@ntu.edu.tw

講義所用資料可以在 Google Colab 以下列指令下載至工作目錄

```
!wget --no-check-certificate
https://raw.githubusercontent.com/datainpoint/workshop-customs-gov-tw-2024/main/data.zip
```

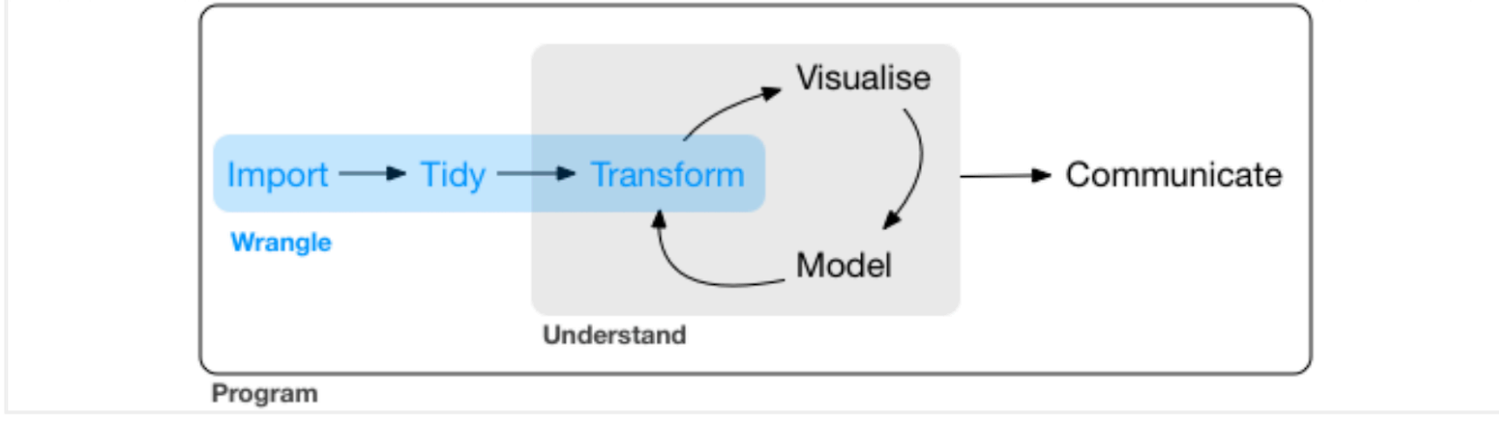
```
import zipfile

local_zip = "/content/data.zip"
zip_ref = zipfile.ZipFile(local_zip, "r")
zip_ref.extractall("/content")
zip_ref.close()
```

In [1]:

```
from string import ascii_uppercase
import os
import re
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
#import cv2
```

關於資料清理

現代資料科學：以程式設計做資料科學的應用



來源：R for Data Science

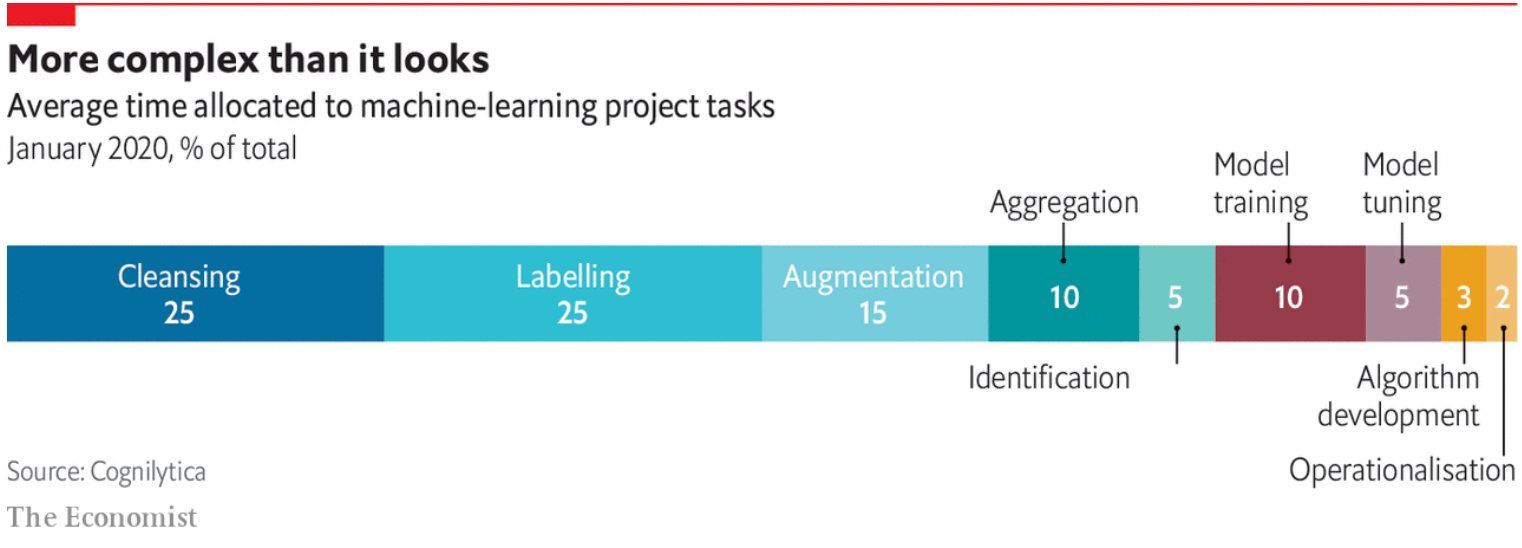
以程式設計做資料科學的應用場景

- **Import 資料的載入**。
- **Tidy 資料清理**。
- **Transform 資料外型與類別的轉換**。
- Visualise 探索性分析。
- Model 分析與預測模型。
- Communicate 溝通分享。

（沒什麼用的冷知識）Wrangle

來源：https://media.giphy.com/media/MnlZWRFHR4xruE4N2Z/giphy.gif

機器學習專案花費 50% 的時間處理 Data Wrangling 的相關任務



來源：https://www.economist.com/technology-quarterly/2020/06/11/for-ai-data-are-harder-to-come-by-than-you-think

多數的資料清理、資料外型與類別的轉換是面對 `DataFrame`

入門 Pandas 的第一步就是掌握 `Index`、`ndarray`、`Series` 與 `DataFrame` 四個資料結構類別彼此之間的關係。

- `Series` 由 `Index` 與 `ndarray` 組合而成。
- `DataFrame` 由數個共享同一個 `Index` 的 `Series` 組合而成。

`DataFrame` 是有兩個維度的資料結構

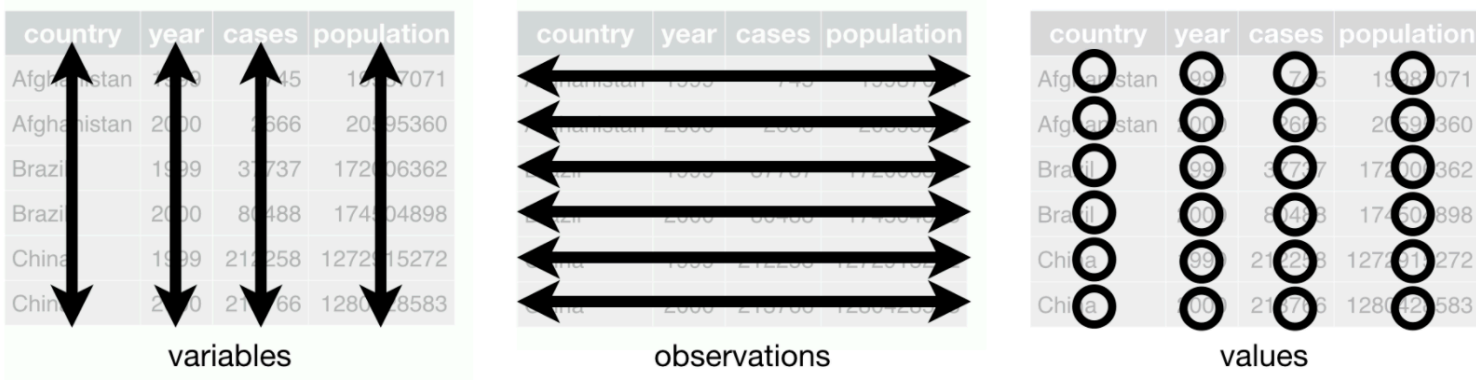
- 第一個維度稱為觀測值（Observations），有時亦稱為列（Rows）
- 第二個維度稱為變數（Variables），有時亦稱為欄（Columns）
- 我們習慣以 `（m, n）` 或者 `m x n` 來描述一個具有 `m` 列觀測值、`n` 欄變數的 `DataFrame`

`DataFrame` 與二維 `ndarray` 不同的地方

- `DataFrame` 的每個變數可以是異質的。
- `DataFrame` 的觀測值具有列標籤（row-label）、變數具有欄標籤（column-label）

什麼是乾淨資料

1. 每個變數有自己的欄位。
2. 每個觀測值有自己的資料列。
3. 每個列、欄標籤與值的對應有自己的儲存格。



來源：https://r4ds.had.co.nz/tidy-data.html

不乾淨資料有著各自的樣態

> *Tidy datasets are all alike, but every messy dataset is messy in its own way.*
>
> *Hadley Wickham*

來源：https://r4ds.had.co.nz/tidy-data.html

資料清理的對象

- 資料框。
- 文字。
- 圖片。

資料框

資料來源為中選會選舉資料庫

https://db.cec.gov.tw/ElecTable/Election

原始資料格式為試算表

我們可以使用 `pd.read_excel()` 函數載入資料。

以臺北市的資料為例

In [2]:
```
file_name = "總統-A05-4-候選人得票數一覽表-各投開票所(臺北市).xlsx"
spreadsheet_path = "data/總統-各投票所得票明細及概況(Excel檔)/{}".format(fi
xl = pd.ExcelFile(spreadsheet_path)
print(xl.sheet_names)
```

```
['臺北市']
```

```
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's de
fault")
```

In [3]:
```
df = pd.read_excel(spreadsheet_path)
df.head()
```

Out[3]:

	第16任總統副總統選舉候選人在臺北市各投開票所得票數一覽表	Unnamed: 1	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5	Unnamed:
0	鄉(鎮、市、區)別	村里別	投開票所別	各組候選人得票情形	NaN	NaN	有效票\nA\nA=1+2+...+
1	NaN	NaN	NaN	(1)\n柯文哲\n吳欣盈	(2)\n賴清德\n蕭美琴	(3)\n侯友宜\n趙少康	Na
2	NaN	NaN	NaN	NaN	NaN	NaN	Na
3	NaN	NaN	NaN	NaN	NaN	NaN	Na
4	總計	NaN	NaN	366854	587899	587258	15420

哪些因素使得原本的試算表不是「乾淨資料」？

- 合併儲存格。
- 未定義值、遺漏值。
- 在觀測值中參雜了「小計」與「總計」。
- 資料值（候選人、政黨）記錄在變數名稱中。

載入試算表時使用 `skiprows` 參數略過合併儲存格

In [4]:
```
df = pd.read_excel(spreadsheet_path, skiprows=[0, 1, 3, 4], thousands='
```

```
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's de
fault")
```

In [5]:
```
df.head()
```

Out[5]:

	Unnamed: 0	Unnamed: 1	Unnamed: 2	(1)\n柯文哲\n吳欣盈	(2)\n賴清德\n蕭美琴	(3)\n侯友宜\n趙少康	Unnamed: 6	Unn
0	總　計	NaN	NaN	366854	587899	587258	1542011	
1	北投區	NaN	NaN	35975	61151	51657	148783	
2	NaN	建民里	1.0	208	401	311	920	
3	NaN	建民里	2.0	209	455	272	936	
4	NaN	建民里	3.0	221	439	306	966	

更新資料框的 `columns` 屬性

In [6]:
```
n_cols = df.columns.size
n_candidates = n_cols - 11
id_vars = ['town', 'village', 'office']
candidates = list(df.columns[3:(3 + n_candidates)])
office_cols = list(ascii_uppercase[:8])
col_names = id_vars + candidates + office_cols
df.columns = col_names
```

In [7]:
```
print(n_candidates)
print(candidates)
print(office_cols)
print(col_names)
```

```
3
['(1)\n柯文哲\n吳欣盈', '(2)\n賴清德\n蕭美琴', '(3)\n侯友宜\n趙少康']
['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H']
['town', 'village', 'office', '(1)\n柯文哲\n吳欣盈', '(2)\n賴清德
\n蕭美琴', '(3)\n侯友宜\n趙少康', 'A', 'B', 'C', 'D', 'E', 'F',
'G', 'H']
```

使用 `df.fillna()` 方法前向填補 `town` 欄位中的未定義值

`ffill` 參數：利用前一個有效值填補未定義值，直到下一個有效值。

In [8]:
```
filled_towns = df['town'].fillna(method='ffill')
df = df.assign(town=filled_towns)
df.head()
```

Out[8]:

	town	village	office	(1)\n柯文哲\n吳欣盈	(2)\n賴清德\n蕭美琴	(3)\n侯友宜\n趙少康	A	B	C	
0	總計	NaN	NaN	366854	587899	587258	1542011	10581	1552592	9
1	北投區	NaN	NaN	35975	61151	51657	148783	1091	149874	
2	北投區	建民里	1.0	208	401	311	920	6	926	
3	北投區	建民里	2.0	209	455	272	936	3	939	
4	北投區	建民里	3.0	221	439	306	966	11	977	

使用 `df.dropna()` 移除「小計」與「總計」

- 遵循「乾淨資料」法則。
- 避免錯誤的加總。

In [9]:
```
df = df.dropna()
df.head()
```

Out[9]:

	town	village	office	(1)\n柯文哲\n吳欣盈	(2)\n賴清德\n蕭美琴	(3)\n侯友宜\n趙少康	A	B	C	D	E	F	G
2	北投區	建民里	1.0	208	401	311	920	6	926	0	926	283	1209
3	北投區	建民里	2.0	209	455	272	936	3	939	0	939	279	1218
4	北投區	建民里	3.0	221	439	306	966	11	977	0	977	263	1240
5	北投區	文林里	4.0	181	396	282	859	7	866	0	866	255	1121

	town	village	office	(1)\n柯文哲\n吳欣盈	(2)\n賴清德\n蕭美琴	(3)\n侯友宜\n趙少康	A	B	C	D	E	F	G
6	北投區	文林里	5.0	206	445	299	950	4	954	0	954	296	1250

使用 `str.replace()` 方法取代多餘的特殊文字 `"\u3000"`

In [10]:

```
print(df['town'].unique())
stripped_strict = df['town'].str.replace("\u3000", "")
df = df.assign(town=stripped_strict)
print(df['town'].unique())
```

```
['\u3000北投區' '\u3000士林區' '\u3000大同區' '\u3000中山區' '\u300
0松山區' '\u3000內湖區'
 '\u3000南港區' '\u3000萬華區' '\u3000中正區' '\u3000大安區' '\u300
0信義區' '\u3000文山區']
['北投區' '士林區' '大同區' '中山區' '松山區' '內湖區' '南港區' '萬華
區' '中正區' '大安區' '信義區' '文山區']
```

使用 `pd.melt()` 函數轉置資料框

In [11]:
```
df = df.drop(labels=office_cols, axis=1)
df_long = pd.melt(df,
                  id_vars=id_vars,
                  var_name='candidate_info',
                  value_name='votes'
                 )
df_long.head()
```

Out[11]:

	town	village	office	candidate_info	votes
0	北投區	建民里	1.0	(1)\n柯文哲\n吳欣盈	208
1	北投區	建民里	2.0	(1)\n柯文哲\n吳欣盈	209
2	北投區	建民里	3.0	(1)\n柯文哲\n吳欣盈	221
3	北投區	文林里	4.0	(1)\n柯文哲\n吳欣盈	181
4	北投區	文林里	5.0	(1)\n柯文哲\n吳欣盈	206

定義函數 `tidy_dataframe()` 將前述的資料操作組織起來

In [12]:

```
def tidy_dataframe(df):
    # updating columns attributes
    n_cols = df.columns.size
    n_candidates = n_cols - 11
    id_vars = ['town', 'village', 'office']
    candidates = list(df.columns[3:(3 + n_candidates)])
    office_cols = list(ascii_uppercase[:8])
    col_names = id_vars + candidates + office_cols
    df.columns = col_names
    # forward-fill district values
    filled_towns = df['town'].fillna(method='ffill')
    df = df.assign(town=filled_towns)
    # removing summations
    df = df.dropna()
    # removing extra spaces
    stripped_towns = df['town'].str.replace("\u3000", "")
    df = df.assign(town=stripped_towns)
    # pivoting
    df = df.drop(labels=office_cols, axis=1)
    tidy_df = pd.melt(df,
                      id_vars=id_vars,
                      var_name='candidate_info',
                      value_name='votes'
```

```
                    )
    return tidy_df
```

將縣市名稱從檔名中取出

In [13]:
```
files = [f for f in os.listdir("data/總統-各投票所得票明細及概況(Excel檔)/"
counties = [re.split("\(|\)", f)[1] for f in files]
print(counties)
```

```
['連江縣', '屏東縣', '臺南市', '雲林縣', '基隆市', '新北市', '新竹市',
'宜蘭縣', '嘉義縣', '臺東縣', '臺北市', '彰化縣', '嘉義市', '新竹縣',
'金門縣', '苗栗縣', '南投縣', '臺中市', '花蓮縣', '高雄市', '澎湖縣',
'桃園市']
```

應用 `tidy_dataframe()` 函數

In [14]:
```
presidential = pd.DataFrame()
for county in counties:
    file_name = "總統-A05-4-候選人得票數一覽表-各投開票所({}).xlsx".format(c
    spreadsheet_path = "data/總統-各投票所得票明細及概況(Excel檔)/{}".forma
    # skip those combined cells
    df = pd.read_excel(spreadsheet_path, skiprows=[0, 1, 3, 4], thousar
    tidy_df = tidy_dataframe(df)
    # appending dataframe of each city/county
    tidy_df['county'] = county
    presidential = pd.concat([presidential, tidy_df])
    print("Tidying {}".format(file_name))
presidential = presidential.reset_index(drop=True) # reset index for th
```

```
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(連江縣).xlsx
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's de
fault")
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's de
fault")
```

```
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(屏東縣).xlsx
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpyxl/styles/stylesheet.py:226: UserWarning: Workbook contains no default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's default")
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(臺南市).xlsx
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(雲林縣).xlsx
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpyxl/styles/stylesheet.py:226: UserWarning: Workbook contains no default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's default")
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpyxl/styles/stylesheet.py:226: UserWarning: Workbook contains no default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's default")
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(基隆市).xlsx
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpyxl/styles/stylesheet.py:226: UserWarning: Workbook contains no default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's default")
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(新北市).xlsx
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(新竹市).xlsx
```

```
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's de
fault")
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's de
fault")
```

```
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(宜蘭縣).xlsx
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(嘉義縣).xlsx
```

```
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's de
fault")
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's de
fault")
```

```
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(臺東縣).xlsx
```

```
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
```

```
  warn("Workbook contains no default style, apply openpyxl's de
fault")
```

```
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(臺北市).xlsx
```

```
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's de
fault")
```

```
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(彰化縣).xlsx
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(嘉義市).xlsx
```

```
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's de
fault")
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's de
fault")
```

```
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(新竹縣).xlsx
```

```
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's de
fault")
```

```
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(金門縣).xlsx
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpyxl/styles/stylesheet.py:226: UserWarning: Workbook contains no default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's default")
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(苗栗縣).xlsx
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpyxl/styles/stylesheet.py:226: UserWarning: Workbook contains no default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's default")
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(南投縣).xlsx
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpyxl/styles/stylesheet.py:226: UserWarning: Workbook contains no default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's default")
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(臺中市).xlsx
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpyxl/styles/stylesheet.py:226: UserWarning: Workbook contains no default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's default")
Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(花蓮縣).xlsx
```

```
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's de
fault")
```

Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(高雄市).xlsx

```
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's de
fault")
```

Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(澎湖縣).xlsx

```
/Users/kuoyaojen/miniconda3/lib/python3.11/site-packages/openpy
xl/styles/stylesheet.py:226: UserWarning: Workbook contains no
default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's de
fault")
```

Tidying 總統-A05-4-候選人得票數一覽表-各投開票所(桃園市).xlsx

In [15]:
```
presidential.head()
```

Out[15]:

	town	village	office	candidate_info	votes	county
0	南竿鄉	介壽村	1.0	(1)\n柯文哲\n吳欣盈	146	連江縣
1	南竿鄉	介壽村	2.0	(1)\n柯文哲\n吳欣盈	128	連江縣
2	南竿鄉	復興村、福沃村	3.0	(1)\n柯文哲\n吳欣盈	239	連江縣
3	南竿鄉	清水村、珠螺村	4.0	(1)\n柯文哲\n吳欣盈	208	連江縣
4	南竿鄉	仁愛村、津沙村、馬祖村、四維村	5.0	(1)\n柯文哲\n吳欣盈	210	連江縣

In [16]:
```
presidential.tail()
```

Out[16]:

	town	village	office	candidate_info	votes	county
53380	復興區	長興里	1348.0	(3)\n侯友宜\n趙少康	192	桃園市
53381	復興區	奎輝里	1349.0	(3)\n侯友宜\n趙少康	268	桃園市
53382	復興區	高義里	1350.0	(3)\n侯友宜\n趙少康	224	桃園市
53383	復興區	三光里	1351.0	(3)\n侯友宜\n趙少康	238	桃園市
53384	復興區	華陵里	1352.0	(3)\n侯友宜\n趙少康	364	桃園市

定義函數 `adjust_presidential()` 調整

In [17]:

```
def adjust_presidential(df):
    # split candidate information into 2 columns
    candidate_info_df = df['candidate_info'].str.split("\n", expand=Tru
    numbers = candidate_info_df[0].str.replace("\(|\)", "", regex=True)
    candidates = candidate_info_df[1].str.cat(candidate_info_df[2], sep
    # re-arrange columns
    df = df.drop(labels='candidate_info', axis=1)
    df['number'] = numbers
    df['candidate'] = candidates
    df['office'] = df['office'].astype(int)
    df = df[['county', 'town', 'village', 'office', 'number', 'candidat
    return df
```

應用 `adjust_presidential()` 函數

In [18]:
```
presidential = adjust_presidential(presidential)
```

In [19]:
```
presidential.head()
```

Out[19]:

	county	town	village	office	number	candidate	votes
0	連江縣	南竿鄉	介壽村	1	1	柯文哲/吳欣盈	146
1	連江縣	南竿鄉	介壽村	2	1	柯文哲/吳欣盈	128
2	連江縣	南竿鄉	復興村、福沃村	3	1	柯文哲/吳欣盈	239
3	連江縣	南竿鄉	清水村、珠螺村	4	1	柯文哲/吳欣盈	208
4	連江縣	南竿鄉	仁愛村、津沙村、馬祖村、四維村	5	1	柯文哲/吳欣盈	210

In [20]:
```
presidential.tail()
```

Out[20]:

	county	town	village	office	number	candidate	votes
53380	桃園市	復興區	長興里	1348	3	侯友宜/趙少康	192
53381	桃園市	復興區	奎輝里	1349	3	侯友宜/趙少康	268
53382	桃園市	復興區	高義里	1350	3	侯友宜/趙少康	224
53383	桃園市	復興區	三光里	1351	3	侯友宜/趙少康	238
53384	桃園市	復興區	華陵里	1352	3	侯友宜/趙少康	364

完成了`總統–各投票所得票明細及概況`!

來源：https://media.giphy.com/media/1sjwSoZLcENCE/giphy.gif

文字

從 `DataFrame` 中擷取特徵矩陣與目標陣列

來源：財政部北區國稅局工作坊

In [21]:
```
file_path = "111年新竹分局電子來文.xls" # upload before importing
electronic_official_doc = pd.read_excel(file_path) # import data
print(type(electronic_official_doc))
print(electronic_official_doc.shape)
```

```
<class 'pandas.core.frame.DataFrame'>
(107, 15)
```

In [22]:
```
electronic_official_doc.head()
```

Out[22]:

	機關	收創文日期	來文方式	收創文文號	來文機關	來文字	來文號	主旨	公文性質	公文文別	簽呈方式
0	O44	1110103	電子來文	1112210002	財政部臺北國稅局	財北國稅內湖營業一	1101607997	貴公司110年12月31日至111年1月2日於新竹尼尼生活館(新竹市東區新安路2-1號)舉辦...	一般公文	函	空白
1	O44	1110103	電子來文	1112210003	財政部中區國稅局	中區國稅竹南銷售	1103355878	貴公司110年12月30日依加值型及非加值型營業稅法(以下簡稱營業稅法)第30條規定，申請變...	一般公文	函	空白

	收創機關	收創日期	來文方式	收創文文號	來文機關	來文字	來文號	主旨	公文性質	公文文別	簽呈方式
2	O44	1110103	電子來文	1112210004	財政部北區國稅局竹北分局	北區國稅竹北綜	1100310056	貴轄納稅義務人張〇勇君對其109年度源自源明科技工程(有)公司之薪資所得持有疑義一案，復如說...	一般公文	函	空白
3	O44	1110104	電子來文	1112210006	財政部北區國稅局	北區國稅審二	1100015568	檢送109年度投資國外或大陸地區且持有投資公司股份比例為100%之公司、外國或大陸公司在臺分...	一般公文	函	空白

	收創機關	收創文日期	來文方式	收創文文號	來文機關	來文字	來文號	主旨	公文性質	公文文別	簽呈方式
4	O44	1110104	電子來文	1112210007	新竹市政府	府產商	1100197212	本府110年12月22日府產商字第1100004569號函核准貴商業合夥人變更登記	一般公文	函	空白

In [23]:
```
X = electronic_official_doc[["來文機關", "主旨", "來文字"]].values
y = electronic_official_doc["承辦科室"].values
print(X.shape)
print(y.shape)
```

```
(107, 3)
(107,)
```

In [24]:
```
print(X[:5])
print(y[:5])
```

```
[['財政部臺北國稅局'
  '貴公司110年12月31日至111年1月2日於新竹尼尼生活館(新竹市東區新安路2-1號)舉辦短期商品展售活動乙案，同意備查，請查照。'
  '財北國稅內湖營業一']
 ['財政部中區國稅局'
  '貴公司110年12月30日依加值型及非加值型營業稅法(以下簡稱營業稅法)第30條規定，申請變更營業所在地址登記一案，稅務部分准予辦理，請查照。'
  '中區國稅竹南銷售']
 ['財政部北區國稅局竹北分局' '貴轄納稅義務人張○勇君對其109年度源自源明科技工程(有)公司之薪資所得持有疑義一案，復如說明二，請查照。'
  '北區國稅竹北綜']
 ['財政部北區國稅局'
  '檢送109年度投資國外或大陸地區且持有投資公司股份比例為100%之公司、外國或大陸公司在臺分公司清冊各1份，請依說明事項辦理，請查照。'
  '北區國稅審二']
 ['新竹市政府'
  '本府110年12月22日府產商字第1100004569號函核准貴商業合夥人變更登記案，因合夥人誤繕，特此更正，茲附登記抄本，請查照。'
  '府產商']]
['新竹分局銷售稅課' '新竹分局銷售稅課' '新竹分局綜所稅課' '新竹分局綜所稅課' '新竹分局銷售稅課']
```

承辦科室有 6 個不同的類別

In [25]:
```
print(electronic_official_doc["承辦科室"].unique())
print(electronic_official_doc["承辦科室"].nunique())
print(electronic_official_doc["承辦科室"].value_counts())
```

```
['新竹分局銷售稅課' '新竹分局綜所稅課' '新竹分局服務管理課' '新竹分局營所
遺贈稅課' '新竹分局人事室' '新竹分局政風室']
6
承辦科室
新竹分局銷售稅課        39
新竹分局綜所稅課        30
新竹分局服務管理課       23
新竹分局營所遺贈稅課       7
新竹分局政風室           5
新竹分局人事室           3
Name: count, dtype: int64
```

什麼是自然語言處理

自然語言處理（Natural Language Processing, NLP）的目標是使得電腦能夠理解自然語言，進而完成一些特定任務，例如：拼字檢查、解析資訊以及語意分析等。

常見的自然語言處理任務

- 關鍵字搜尋。
- 同義字詞搜尋。
- 機器翻譯。
- 語意分析。
- 問答系統。
- ...等。

自然語言處理任務的基礎

- 分詞分句。
- 詞性標注。
- 關鍵詞擷取。
- 命名實體辨識（Named-Entity Recognition, NER）。

依據語言選擇 Python 自然語言處理模組

- 英文：`nltk`
- 中文：`jieba`
- 多語系：`polyglot`

關於 `nltk`

Natural Language Toolkit 是 Python 最好的英文自然語言處理模組，功能涵蓋了分類、分詞、詞幹提取與詞性標註等。

來源：https://www.nltk.org

關於 `jieba`

「结巴」中文分詞：做最好的 Python 中文分詞模組。

來源：https://github.com/fxsjy/jieba

關於 `polyglot`

支援多語系的 Python 分詞、命名實體辨識模組。

來源：https://polyglot.readthedocs.io/en/latest

In [26]:
```
for title in X[:, 1].ravel()[:5]:
    print(title)
```

貴公司110年12月31日至111年1月2日於新竹尼尼生活館(新竹市東區新安路2-1號)舉辦短期商品展售活動乙案，同意備查，請查照。
貴公司110年12月30日依加值型及非加值型營業稅法(以下簡稱營業稅法)第30條規定，申請變更營業所在地址登記一案，稅務部分准予辦理，請查照。
貴轄納稅義務人張○勇君對其109年度源自源明科技工程(有)公司之薪資所得持有疑義一案，復如說明二，請查照。
檢送109年度投資國外或大陸地區且持有投資公司股份比例為100%之公司、外國或大陸公司在臺分公司清冊各1份，請依說明事項辦理，請查照。
本府110年12月22日府產商字第1100004569號函核准貴商業合夥人變更登記案，因合夥人誤繕，特此更正，茲附登記抄本，請查照。

使用 `jieba` 進行特徵工程：分詞

In [27]:

```
import jieba

for title in X[:, 1].ravel()[:5]:
    list_cut = jieba.lcut(title)
    print(list_cut)
```

```
Building prefix dict from the default dictionary ...
Loading model from cache /var/folders/0b/r__z5mpn6ldgb_w2j7_y_n
tr0000gn/T/jieba.cache
Loading model cost 4.914 seconds.
Prefix dict has been built successfully.
['貴', '公司', '110', '年', '12', '月', '31', '日至', '111', '年',
'1', '月', '2', '日', '於', '新竹', '尼尼', '生活', '館', '(', '新
竹市', '東區', '新安', '路', '2', '-', '1', '號', ')', '舉辦', '短
期', '商品', '展售', '活動', '乙案', ',', '同意', '備查', ',',
'請', '查照', '。']
['貴', '公司', '110', '年', '12', '月', '30', '日依', '加值', '型',
'及', '非', '加值', '型', '營業', '稅法', '(', '以下', '簡稱', '營
業', '稅法', ')', '第', '30', '條規定', ',', '申請', '變', '更',
'營業', '所在', '地址', '登記', '一案', ',', '稅務', '部分', '准予',
'辦理', ',', '請', '查照', '。']
['貴轄納', '稅義務人', '張', '○', '勇君', '對', '其', '109', '年度',
'源自', '源明', '科技', '工程', '(', '有', ')', '公司', '之薪資',
'所得', '持有', '疑義', '一案', ',', '復', '如', '說', '明二',
```

', ', '請', '查照', '。']
['檢送', '109', '年度', '投資國外', '或', '大陸', '地區且', '持有', '投資', '公司', '股份', '比例', '為', '100%', '之', '公司', '、', '外國', '或', '大陸', '公司', '在', '臺', '分公司', '清冊', '各', '1', '份', '，', '請', '依說', '明事項', '辦理', '，', '請', '查照', '。']
['本府', '110', '年', '12', '月', '22', '日府', '產商字', '第', '11100004569', '號函', '核准', '貴商業', '合夥人', '變', '更', '登記案', '，', '因', '合夥人', '誤繕', '，', '特此', '更正', '，', '茲', '附登記', '抄本', '，', '請', '查照', '。']

使用 `jieba` 進行特徵工程：設定詞典並分詞

In [28]:

```
jieba.set_dictionary("data/dict.txt.big") # 設定詞典
for title in X[:, 1].ravel()[:5]:
    list_cut = jieba.lcut(title)
    print(list_cut)
```

```
Building prefix dict from /Users/kuoyaojen/workshop-customs-gov
-tw-2024/data/dict.txt.big ...
Loading model from cache /var/folders/0b/r__z5mpn6ldgb_w2j7_y_n
tr0000gn/T/jieba.uf58e43bf70006972920ed3b7ad7e2c46.cache
Loading model cost 5.641 seconds.
Prefix dict has been built successfully.
['貴', '公司', '110', '年', '12', '月', '31', '日至', '111', '年',
'1', '月', '2', '日', '於', '新竹', '尼尼', '生活館', '(', '新竹市',
'東區', '新安', '路', '2', '-', '1', '號', ')', '舉辦', '短期', '商
品', '展售', '活動', '乙案', ',', '同意', '備查', ',', '請', '查
照', '。']
['貴', '公司', '110', '年', '12', '月', '30', '日依', '加值', '型',
'及', '非', '加值', '型', '營業稅', '法', '(', '以下', '簡稱', '營業
稅', '法', ')', '第', '30', '條規', '定', ',', '申請', '變更', '營
業所', '在', '地址', '登記', '一案', ',', '稅務', '部分', '准予',
'辦理', ',', '請', '查照', '。']
['貴轄', '納稅', '義務人', '張', '○', '勇君', '對', '其', '109', '年
度', '源自', '源明', '科技', '工程', '(', '有', ')', '公司', '之',
'薪資', '所得', '持有', '疑義', '一案', ',', '復', '如', '說明',
'二', ',', '請', '查照', '。']
['檢送', '109', '年度', '投資', '國外', '或', '大陸', '地區', '且',
'持有', '投資', '公司', '股份', '比例', '為', '100%', '之', '公司',
'、', '外國', '或', '大陸', '公司', '在', '臺', '分公司', '清冊',
'各', '1', '份', ',', '請', '依', '說明', '事項', '辦理', ',',
```

'請', '查照', '。']
['本府', '110', '年', '12', '月', '22', '日府', '產商字', '第', '11
00004569', '號函', '核准', '貴', '商業', '合夥人', '變更', '登記',
'案', '，', '因', '合夥人', '誤繕', '，', '特此', '更正', '，',
'茲', '附', '登記', '抄本', '，', '請', '查照', '。']

使用 `jieba` 進行特徵工程：設定詞典、新增詞典並分詞

`userdict.txt`

```
人事
政風
服務管理
營業所得
遺產贈與
綜合所得
銷售
尼尼生活館
```

In [29]:

```
jieba.load_userdict("data/userdict.txt")  # 新增詞典
for title in X[:, 1].ravel()[:5]:
    list_cut = jieba.lcut(title)
    print(list_cut)
```

```
['貴', '公司', '110', '年', '12', '月', '31', '日至', '111', '年',
'1', '月', '2', '日', '於', '新竹', '尼尼生活館', '(', '新竹市', '東
區', '新安', '路', '2', '-', '1', '號', ')', '舉辦', '短期', '商
品', '展售', '活動', '乙案', ',', '同意', '備查', ',', '請', '查
照', '。']
['貴', '公司', '110', '年', '12', '月', '30', '日依', '加值', '型',
'及', '非', '加值', '型', '營業稅', '法', '(', '以下', '簡稱', '營業
稅', '法', ')', '第', '30', '條規', '定', ',', '申請', '變更', '營
業所', '在', '地址', '登記', '一案', ',', '稅務', '部分', '准予',
'辦理', ',', '請', '查照', '。']
['貴轄', '納稅', '義務人', '張', '○', '勇君', '對', '其', '109', '年
度', '源自', '源明', '科技', '工程', '(', '有', ')', '公司', '之',
'薪資', '所得', '持有', '疑義', '一案', ',', '復', '如', '說明',
'二', ',', '請', '查照', '。']
['檢送', '109', '年度', '投資', '國外', '或', '大陸', '地區', '且',
'持有', '投資', '公司', '股份', '比例', '為', '100%', '之', '公司',
'、', '外國', '或', '大陸', '公司', '在', '臺', '分公司', '清冊',
'各', '1', '份', ',', '請', '依', '說明', '事項', '辦理', ',',
'請', '查照', '。']
['本府', '110', '年', '12', '月', '22', '日府', '產商字', '第', '11
00004569', '號函', '核准', '貴', '商業', '合夥人', '變更', '登記',
'案', ',', '因', '合夥人', '誤繕', ',', '特此', '更正', ',',
'茲', '附', '登記', '抄本', ',', '請', '查照', '。']
```

使用 jieba 進行特徵工程：詞性標注

In [30]:
```
import jieba.posseg as pseg

words = pseg.cut(X[0, 1])
for word, flag in words:
    print({flag: word})
```

```
{'a': '貴'}
{'n': '公司'}
{'m': '110'}
{'m': '年'}
{'m': '12'}
{'m': '月'}
{'m': '31'}
{'m': '日'}
{'p': '至'}
{'m': '111'}
{'m': '年'}
{'m': '1'}
{'m': '月'}
{'m': '2'}
{'m': '日'}
{'nr': '於'}
{'ns': '新竹'}
{'x': '尼尼生活館'}
{'x': '('}
```

{'ns': '新竹市'}
{'ns': '東區'}
{'ns': '新安'}
{'n': '路'}
{'m': '2'}
{'x': '-'}
{'m': '1'}
{'m': '號'}
{'x': ')'}
{'v': '舉辦'}
{'b': '短期'}
{'n': '商品'}
{'v': '展售'}
{'vn': '活動'}
{'n': '乙案'}
{'x': ', '}
{'d': '同意'}
{'vn': '備查'}
{'x': ', '}
{'zg': '請'}
{'v': '查照'}
{'x': '。'}

詞性列表

專有名詞類別標籤：

- PER 人名
- LOC 地名
- ORG 機構
- TIME 時間

來源：https://github.com/fxsjy/jieba

詞性列表（續）

一般名詞類別標籤：

來源：https://github.com/fxsjy/jieba

關鍵詞擷取

- TF-IDF 關鍵詞演算法（Term Frequency-Inverse Document Frequency Algorithm）。
- 某個詞在一篇文章中出現的頻率高，且在其他文章中很少出現，那麼該詞為具代表性的關鍵詞。

In [31]:
```
import jieba.analyse

for title in X[:, 1].ravel()[:5]:
    tags = jieba.analyse.extract_tags(title, 10)
    print(list(tags))
```

```
['查照', '110', '12', '31', '111', '尼尼生活館', '東區', '舉辦',
'展售', '活動']
['30', '加值', '營業稅', '查照', '110', '12', '日依', '簡稱', '條
規', '申請']
['查照', '貴轄', '納稅', '義務人', '勇君', '109', '源明', '薪資', '疑
義', '說明']
['投資', '大陸', '查照', '檢送', '109', '國外', '地區', '100%', '外
國', '清冊']
['合夥人', '登記', '查照', '110', '12', '22', '日府', '產商字', '11
00004569', '號函']
```

依據詞性標注結果移除數量詞（m）之後再進行關鍵詞擷取

- TF-IDF 關鍵詞演算法（Term Frequency-Inverse Document Frequency Algorithm）。
- 某個詞在一篇文章中出現的頻率高，且在其他文章中很少出現，那麼該詞為具代表性的關鍵詞。

In [32]:
```
for title in X[:, 1].ravel()[:5]:
    words = pseg.cut(title)
    title_removed_m = ""
    for word, flag in words:
        if flag != "m":
            title_removed_m += word
    tags = jieba.analyse.extract_tags(title_removed_m, 10)
    print(list(tags))
```

```
['查照', '至於', '尼尼生活館', '東區', '舉辦', '展售', '活動', '乙案',
'備查', '新竹市']
['加值', '營業稅', '查照', '簡稱', '條規', '申請', '變更', '營業所',
'登記', '稅務']
['查照', '貴轄', '納稅', '義務人', '勇君', '源明', '薪資', '疑義',
'說明', '源自']
['投資', '大陸', '查照', '檢送', '國外', '地區', '外國', '清冊', '各
份', '說明']
```

```
['合夥人', '登記', '查照', '日府', '產商字', '商業', '變更', '誤繕',
'本府', '更正']
```

In [33]:

```
titles_in_tags = []
for title in X[:, 1].ravel():
    words = pseg.cut(title)
    title_removed_m = ""
    for word, flag in words:
        if flag != "m":
            title_removed_m += word
    tags = jieba.analyse.extract_tags(title_removed_m, 10)
    join_tags = " ".join(tags)
    titles_in_tags.append(join_tags)
print(titles_in_tags[:5])
```

```
['查照 至於 尼尼生活館 東區 舉辦 展售 活動 乙案 備查 新竹市', '加值 營業
稅 查照 簡稱 條規 申請 變更 營業所 登記 稅務', '查照 貴轄 納稅 義務人 勇
君 源明 薪資 疑義 說明 源自', '投資 大陸 查照 檢送 國外 地區 外國 清冊 各
份 說明', '合夥人 登記 查照 日府 產商字 商業 變更 誤繕 本府 更正']
```

關鍵詞再加入來文機關、來文字

In [34]:

```
orgs_titles_in_tags = []
orgs = electronic_official_doc["來文機關"].values
org_abbs = electronic_official_doc["來文字"].values
for title, org, org_abb in zip(X[:, 1].ravel(), orgs, org_abbs):
    words = pseg.cut(title)
    title_removed_m = ""
    for word, flag in words:
        if flag != "m":
            title_removed_m += word
    tags = jieba.analyse.extract_tags(title_removed_m, 10)
    join_tags = " ".join(tags)
    org_str = f"{org} {org_abb} "
    org_join_tags = org_str + join_tags
    orgs_titles_in_tags.append(org_join_tags)
print(orgs_titles_in_tags[:5])
```

```
['財政部臺北國稅局 財北國稅內湖營業一 查照 至於 尼尼生活館 東區 舉辦 展售
活動 乙案 備查 新竹市', '財政部中區國稅局 中區國稅竹南銷售 加值 營業稅 查
照 簡稱 條規 申請 變更 營業所 登記 稅務', '財政部北區國稅局竹北分局 北區國
稅竹北綜 查照 貴轄 納稅 義務人 勇君 源明 薪資 疑義 說明 源自', '財政部北
區國稅局 北區國稅審二 投資 大陸 查照 檢送 國外 地區 外國 清冊 各份 說明',
'新竹市政府 府產商 合夥人 登記 查照 日府 產商字 商業 變更 誤繕 本府 更
正']
```

對特徵矩陣 X 進行文字編碼：文字無法計算，數值才能計算

- Bag of Words
- TF-IDF（Term Frequency Inverse Document Frequency）
- Word2Vector
- BERT

使用 Scikit-Learn 模組的轉換器進行 Bag of Words 文字編碼

In [35]:

```
from sklearn.feature_extraction.text import CountVectorizer

cv = CountVectorizer(stop_words="english")
X_org_title = cv.fit_transform(orgs_titles_in_tags)
print(X_org_title.toarray().shape)
print(X_org_title.toarray()[:5, :])
```

```
(107, 516)
[[0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]]
```

使用 Scikit-Learn 模組的轉換器進行 TF-IDF 文字編碼

In [36]:

```
from sklearn.feature_extraction.text import TfidfVectorizer

tfidf = TfidfVectorizer(stop_words="english")
X_org_title = tfidf.fit_transform(orgs_titles_in_tags)
print(X_org_title.toarray().shape)
print(X_org_title.toarray()[:5, :])
```

```
(107, 516)
[[0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]]
```

圖片

關於圖片資料預處理

- 圖片資料在電腦視覺中是以「像素強度」所組成的陣列來表示。
- 圖片資料預處理目的是為了增強資料的品質或者資料中指定的特徵。
- 因應運用目的性，預處理技巧可以大致分為四種：
    - 像素亮度的調整。
    - 圖片的幾何轉換（旋轉、正規化等）。
    - 運用鄰近像素特性進行預處理。
    - 基於對全圖理解所進行的修復。

常見的圖片資料預處理技巧

- 影像降噪（Noise reduction）。
- 影像強化（Contrast enhancement）。
- 調整大小。
- 色彩校正。
- 圖像分割（Segmentation）。
- 特徵提取。

Python 圖片資料預處理模組

- `numpy`
- `matplotlib`
- `opencv-python`

關於 `opencv-python`

OpenCV-Python 是 OpenCV(Open Computer Vision)的 Python API，能夠讓 Python 使用者在處理電腦視覺問題時獲得和 OpenCV C++ 依樣良好的體驗。

來源：https://opencv.org

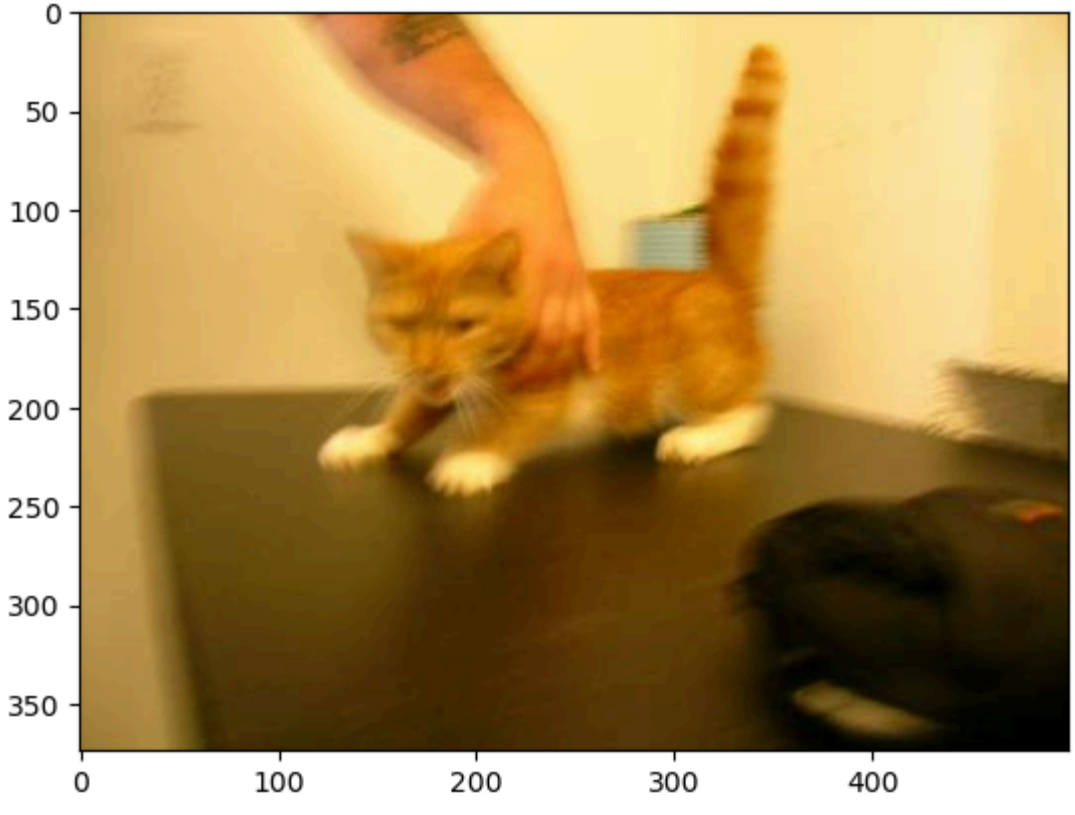
載入圖片為 `ndarray`

來源：https://www.kaggle.com/competitions/dogs-vs-cats

In [37]:
```
cat0 = plt.imread("data/cat.0.jpg")
print(type(cat0))
print(cat0.shape)
```

```
<class 'numpy.ndarray'>
(374, 500, 3)
```

In [38]:
```
fig, ax = plt.subplots()
ax.imshow(cat0)
plt.show()
```



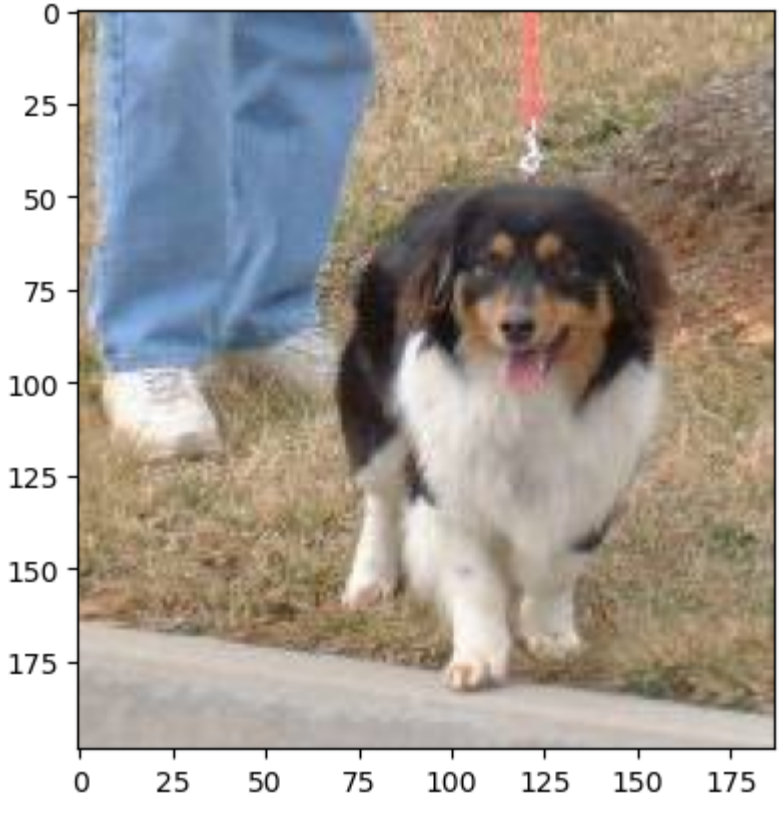
In [39]:

```
dog2 = plt.imread("data/dog.2.jpg")
print(type(dog2))
print(dog2.shape)
```

```
<class 'numpy.ndarray'>
(199, 187, 3)
```

In [40]:

```
fig, ax = plt.subplots()
ax.imshow(dog2)
plt.show()
```



灰階化 grayscaling

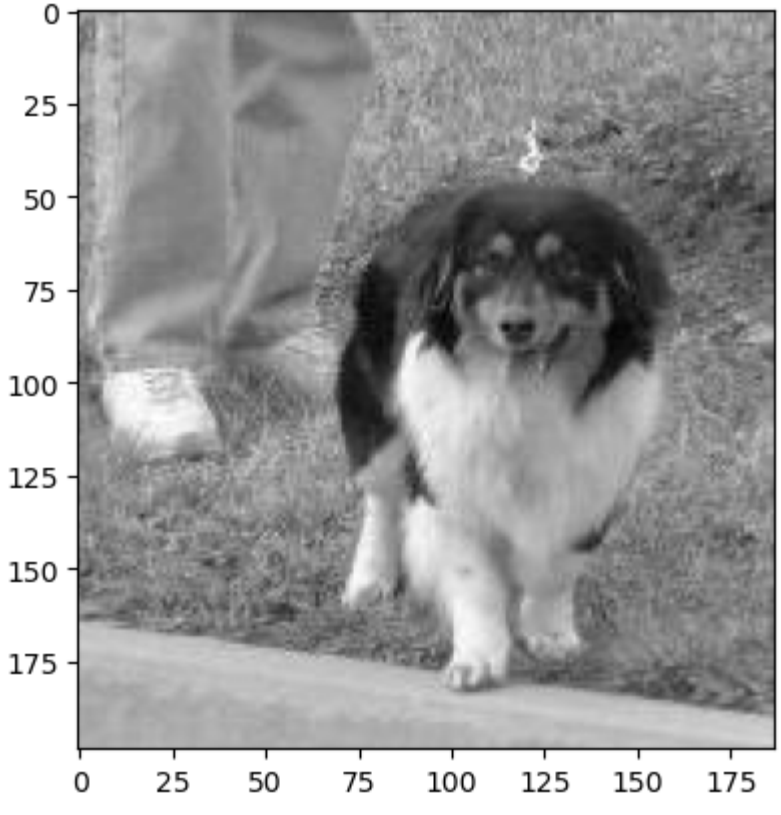
- 降低圖片的複雜度，將三維陣列降維成二維陣列。
- 進而減少模型運算需求。

In [41]:
```
import cv2

grayscaled_dog2 = cv2.cvtColor(dog2, cv2.COLOR_BGR2GRAY)
print(dog2.shape)
print(grayscaled_dog2.shape)
```

```
(199, 187, 3)
(199, 187)
```

In [42]:

```
fig, ax = plt.subplots()
ax.imshow(grayscaled_dog2, cmap='gray')
plt.show()
```



閾值化 thresholding

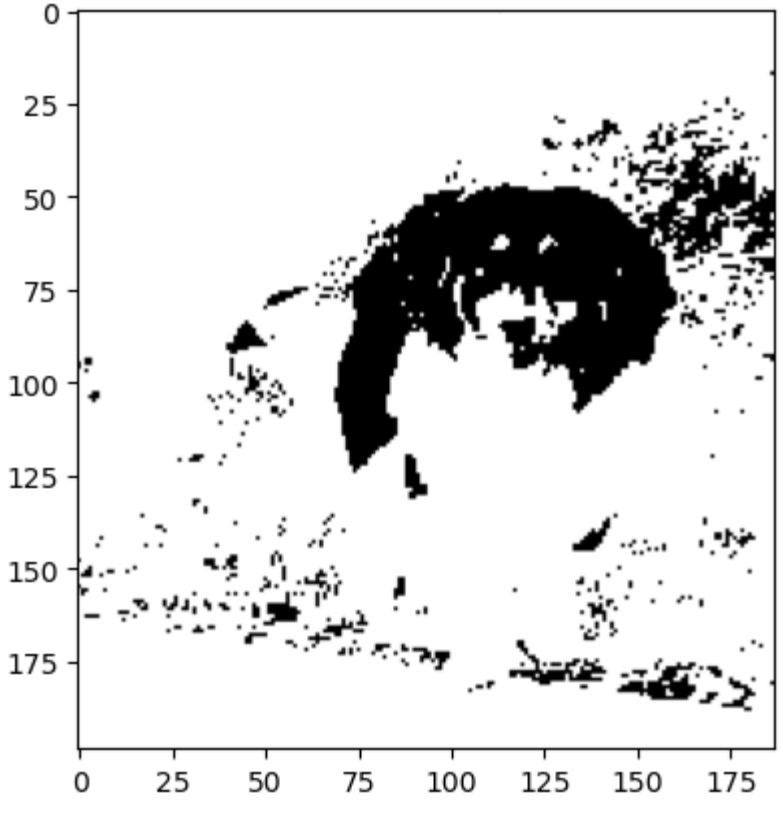
- 進一步將灰階圖片轉換為黑白圖片。
- 低於閾值的給予白色（強度 0）、高於閾值的給予黑色（強度 255）。

In [43]:
```
retval, threshold_dog2 = cv2.threshold(grayscaled_dog2, thresh=100, ma
print(retval)
print(grayscaled_dog2)
print(threshold_dog2)
```

```
100.0
[[148 170 178 ... 148 159 170]
 [138 169 170 ... 142 153 167]
 [147 166 147 ... 143 153 165]
 ...
 [174 175 176 ... 214 212 210]
 [177 177 176 ... 215 215 214]
 [176 176 175 ... 193 194 194]]
[[255 255 255 ... 255 255 255]
 [255 255 255 ... 255 255 255]
 [255 255 255 ... 255 255 255]
 ...
 [255 255 255 ... 255 255 255]
 [255 255 255 ... 255 255 255]
 [255 255 255 ... 255 255 255]]
```

In [44]:

```
fig, ax = plt.subplots()
ax.imshow(threshold_dog2, cmap='gray')
plt.show()
```



大津閾值化 Otsu's Threshold

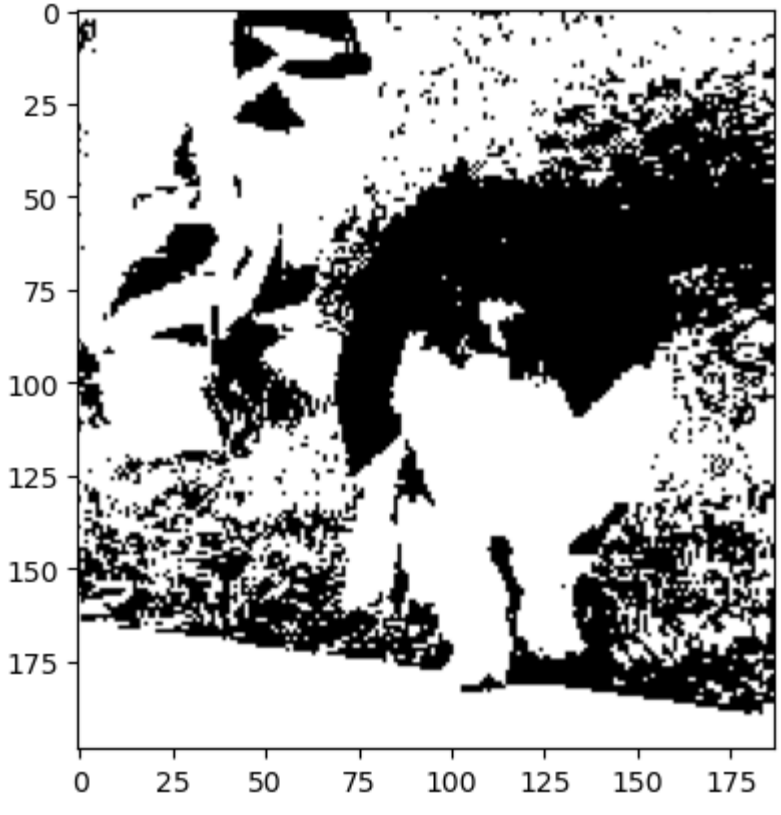
使用大津演算法決定閾值。

In [45]:
```
retval, threshold_dog2 = cv2.threshold(grayscaled_dog2, thresh=0, maxva
print(retval)
```

```
132.0
```

In [46]:

```
fig, ax = plt.subplots()
ax.imshow(threshold_dog2, cmap='gray')
plt.show()
```

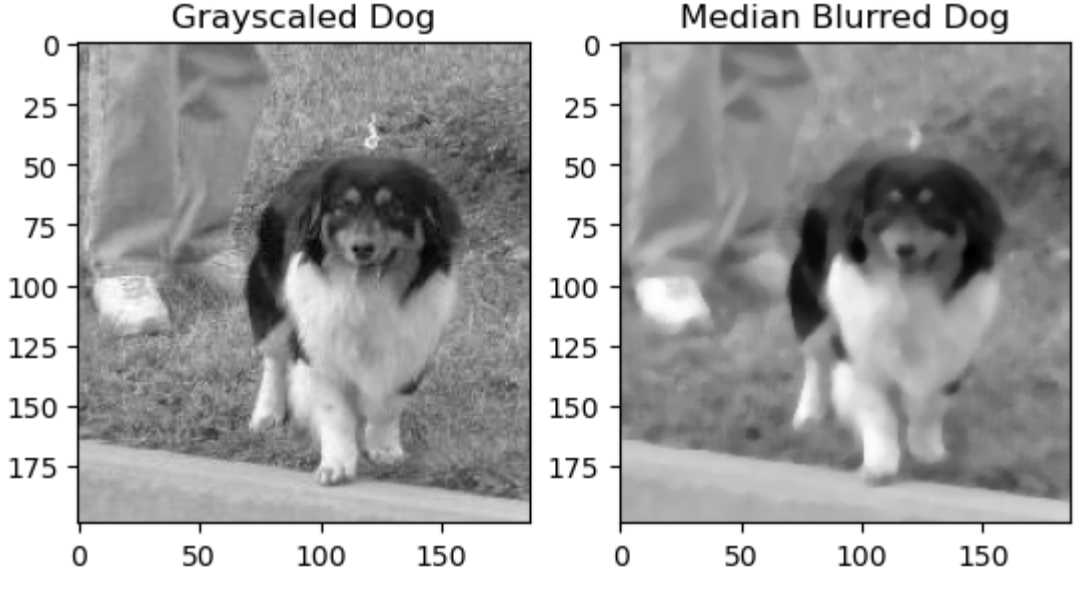


影像降噪 Noise reduction

- 移除影像中不必要的雜訊，保留影像中較為重要的細節，進而使得到的圖像清晰。
- 影像降噪的方法稱為濾波器（Filter），有許多不同的濾波器可以採用，例如均值濾波或高斯平滑濾波。

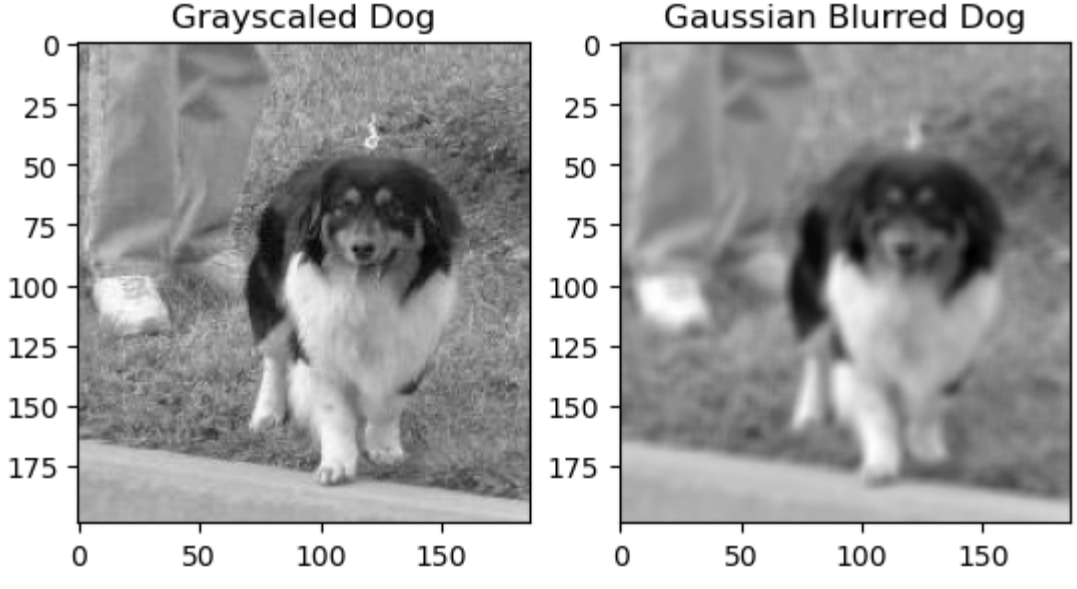
In [47]:

```
median_blur_dog2 = cv2.medianBlur(grayscaled_dog2, ksize=5)
fig, axes = plt.subplots(1, 2)
axes[0].imshow(grayscaled_dog2, cmap='gray')
axes[0].set_title("Grayscaled Dog")
axes[1].imshow(median_blur_dog2, cmap='gray')
axes[1].set_title("Median Blurred Dog")
plt.show()
```



In [48]:

```
gaussian_blur_dog2 = cv2.GaussianBlur(grayscaled_dog2, ksize=(5, 5), s
fig, axes = plt.subplots(1, 2)
axes[0].imshow(grayscaled_dog2, cmap='gray')
axes[0].set_title("Grayscaled Dog")
axes[1].imshow(gaussian_blur_dog2, cmap='gray')
axes[1].set_title("Gaussian Blurred Dog")
plt.show()
```



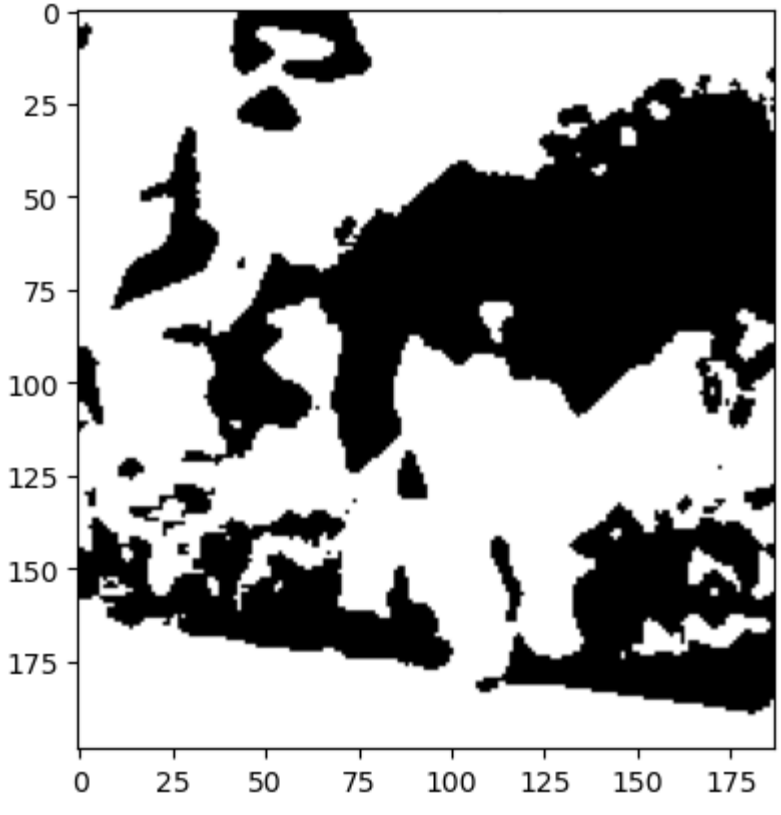
影像降噪後再進行閾值化

In [49]:
```
median_blur_dog2 = cv2.medianBlur(grayscaled_dog2, ksize=5)
retval, threshold_dog2 = cv2.threshold(median_blur_dog2, thresh=0, maxv
print(retval)
```

```
134.0
```

In [50]:

```
fig, ax = plt.subplots()
ax.imshow(threshold_dog2, cmap='gray')
plt.show()
```



邊緣偵測 Edge detection

標記影像中強度變化明顯的點。

In [51]:
```
edges_dog2 = cv2.Canny(grayscaled_dog2, 0, 255)
```

In [52]:

```
fig, ax = plt.subplots()
ax.imshow(edges_dog2, cmap='gray')
plt.show()
```