

# 資料處理 Data Processing

國立東華大學電機工程學系 楊哲旻

# Outline




1 純文字文件 (.txt)

2 常見傳遞資料的格式  
(.csv .xlsx .xml .json)

3 影像 (.jpg .png ...)

4 Homework

 在數據分析與人工智慧模型訓練前的第一步，就是要了解資料與蒐集資料，目前常見的資料格式分為**結構化**、**半結構化**與**非結構化**資料：

	結構化資料 (Structured Data)	半結構化資料 (Semi-Structured Data)	非結構化資料 (Unstructured Data)
定義	嚴謹定義為資料可以被呈現在資料庫表格的行與欄，即已被整理過的資料	便於資料交換，其特性同時具備欄位概念與欄位可拓展性，可透過欄位查詢資料，並可根據使用者需求來增減欄位	形式自由且不遵循標準的格式規範，一團沒有組織的數據，即未經整理過的資料
優缺點	查詢資料快速，佔用存儲空間少；缺點是拓展新的欄位比較麻煩，在資料交換上的規定也比較嚴格	利於資料交換與傳輸，並可以增減欄位；缺點每筆資料的結構可能會不一致	佔用更多存儲空間，無法直接用於數據分析、未規則性的資料很難處理與整理
範例	關聯式資料庫(MySQL, Oracle等)的資料、Excel	CSV、JSON與XML	文字、圖片、音樂、影片、PDF、網頁等

※ 先有結構，再有資料

# 1. 純文字文件 (.txt)

## 步驟：開啟 — 寫入/讀取 — 關閉

1. 開啟 `file_obj = open(file, mode="r")`

r 開啟檔案只供讀取，為預設值

w 開啟檔案供寫入，如果原先檔案有內容，其內容將被覆蓋

a 開啟檔案供寫入，如果原先檔案有內容，新寫入的資料將附加在後面

x 開啟一個新的檔案供寫入，如果所開啟的檔案已經存在則會產生錯誤

2. 讀檔 `file_obj.readlines()` 一次讀一行，以列表呈現

`file_obj.read()` 一次讀全部

3. 寫檔 `file_obj.write(str)`

`print(str, file=file_obj)`

4. 關閉 `file_obj.close()` 每次開啟檔案，請必要執行關閉

若使用with as 開啟檔案，以下程式執行完會自動關閉 `with open(file, mode="r") as file_obj:`

## 2. 常見傳遞資料的格式

 Pandas 是一個資料處理與資料分析常用的開源套件 (<https://pandas.pydata.org/docs/index.html>)

1. 給予資料與欄位字串，建立一DataFrame `df = pd.DataFrame(data, columns = [str])`
2. 呈現DataFrame資料：  
前n筆資料(n預設為5) `df.head(n=5)`  
後n筆資料(n預設為5) `df.tail(n=5)`
3. 從DataFrame中取得一欄位的資料：  
`data_column = df[str]`
4. 儲存DataFrame資料：
  - csv `df.to_csv(path, index = bool)` index為第一欄位編號是否存取
  - excel `df.to_excel(path, index = bool, sheet_name = str)`
  - json `df.to_json(path)` sheet\_name 為工作表的名稱
  - xml `df.to_xml(path)` 備註：pandas 版本為 1.3.0 以上才能使用

## 2. 常見傳遞資料的格式

5. 讀取資料： ■ csv `df = pd.read_csv(path)` openpyxl 支持較新的試算表格式

■ excel `df = pd.read_excel(path, engine='openpyxl')`

■ json `df = pd.read_json(path)`

■ xml `df = pd.read_xml(path)`

6. 新建欄位 `df[str] = data`

單欄新增，若str是以存在的欄位，其資料內容則會被取代

`df.insert(index, str, data)` 單欄新增，index為插入的欄位位置

`df = df.assign(str1 = data1, str2 = data2, ...)` 多欄新增

7. 新建資料(列) `df2 = df2.append(df1)`

多列新增，df1的欄位名稱要與df2相同，此方法較多限制

# 3. 影像

5



處理影像的套件常見的如下五個：

	OpenCV (cv2)	Matplotlib	Scipy
讀取資料	<code>cv2.imread(path)</code>	<code>matplotlib.image.imread(path)</code>	<code>scipy.misc.imread(path)</code>
資料型別	<code>numpy.ndarray</code>	<code>numpy.ndarray</code>	<code>numpy.ndarray</code>
顯示影像	<code>cv2.imshow(Title, img)</code>	<code>matplotlib.pyplot.imshow(img)</code> <code>matplotlib.pyplot.matshow(img)</code> <code>matplotlib.pyplot.show()</code>	<code>scipy.misc.imshow(img)</code>
儲存影像	<code>cv2.imwrite(path, img)</code>	<code>matplotlib.pyplot.imsave(path, img)</code>	<code>scipy.misc.imsave(path, img)</code>

	PIL	Tensorflow (tf), Keras
讀取資料	<code>PIL.Image.open(path)</code>	<code>tf.keras.preprocessing.image.load_img(path)</code>
資料型別	<code>PIL</code>	<code>PIL</code>
顯示影像	<code>img.show()</code>	<code>img.show()</code>
儲存影像	<code>img.save(path)</code>	<code>tf.keras.preprocessing.image.save_img(path, img)</code>

PIL轉為陣列，可用兩種方法：`tf.keras.preprocessing.image.img_to_array(img)`，`numpy.array(img)`



# Homework



# Homework

6

1. 打印三角形聖誕樹（右圖），使用for迴圈打印，並儲存至純文字文件：

- 樹葉為底十個 \* 字號，依序減二，最高為兩個 \* 字號且皆置中
- 樹幹為高寬兩個 \*

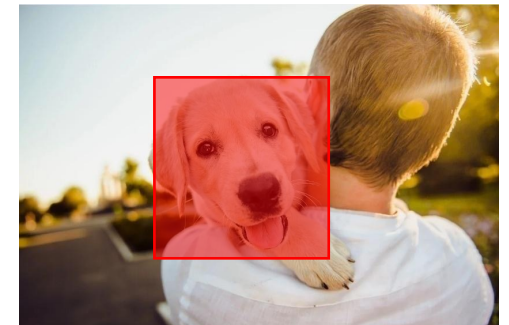
```
  **
 ****
*****
*****
*****
  **
  **
```

2. 表格（左圖）再新增欄位為 BMI，其數值為身高與體重所計算的，型別為浮點數取小數點兩位（右圖）

	age	city	height	weight	sex	SBP	DBP
0	23	Japan	175	68	M	120	85
1	18	Taiwan	168	55	F	114	90
2	30	USA	173	75	M	145	75
3	25	Taiwan	158	50	F	110	78



	age	city	height	weight	sex	SBP	DBP	BMI
0	23	Japan	175	68	M	120	85	22.20
1	18	Taiwan	168	55	F	114	90	19.49
2	30	USA	173	75	M	145	75	25.06
3	25	Taiwan	158	50	F	110	78	20.03



3. 右圖影像轉為陣列裁減至左上角(140, 220)至右下角(450, 520)的矩形，並用Matplotlib顯示其裁減影像並儲存