COVID-19 vs Netfiex 進行一場資料科學研究

簡報閱讀 範例與作業 問題討論 學習心得(完成)

Python資料科學期末專題目標

利用機器學習(Machine Learning)進行資料挖掘(Data Ming)

- 如何使用特徵工程找出目標
- 串聯資料分析在分析的過程中

了解資料視覺化應用

- 如何處理 CSV data
- 比方說用熱點圖來看你的 Deep learning 的 model 是對圖片中哪一部分的看得較重要，
- 可以降維之後將資料視覺化去看資料在空間中的分佈

Numpy, Pandas, Seaborn, Bokeh, Basemap

Python資料科學期末專題知識點

了解如何使用 Pandas 處理資料集

- 導入資料集，瞭解有關資料集屬性
- 針對特徵，視覺化的處理流程與效果

視覺化的前置工作，利用 Matplotlib, Seaborn, Basemao

- 制定好題目先有清楚目的才有好視覺化
- 釐清測量尺度
- 資料處理與探勘

如何達到正確的視覺化效果

期末專題實作提示

- 您可以在 windows，Mac 或 Linux 的 command prompt/terminal 的 Python 環境上執行習題的程式。也可以在 Jupyter 的環境上執行，我是在 Windows 10 上裝 Anaconda 然後在 command prompt 上執行。至於編輯，任何的純文字編輯器皆可。
- 每個習題對應到課程的某一或某些學習點，除了題目和說明外，我們也提供基礎 Python 程式碼給您參考，以及當作延伸的起點。
- 期待同學在原有基礎上以不同的繪圖套件，並使用不同的 ML 甚至 DL 的演算法把原始資料做了正確繪製。

專題習題實作進階提示

- 用 Pandas 讀入時有時需要注意 encoding 參數
  - 利用 dataframe 去 Creating new feature "Active_case"
  - Active_case = Confirmed - Deaths – Recovered

- 需要注意異常值與缺失值的處理
- 注意資料區間，評估值區間差異過大的問題
- 資料的分類，以期可以分別繪製比對圖形
- 可以利用 BOKEH，Basemap 繪製出交互作用的地圖

特徵工程到底是什麼
網站：知乎

主要是希望同學大致知道特徵工程大致**包含哪些部分**，若對細節有興趣，還可以從這篇中了解一些概念

使用preproccessing庫的StandardScaler類對數據進行標準化的代碼如下：

$$x' = \frac{x - \bar{X}}{S}$$

```
from sklearn.preprocessing import StandardScaler

#标准化，返回值为标准化后的数据
StandardScaler().fit_transform(iris.data)
```

2.1.2 區間縮放法

區間縮放法的思路有多種，常見的一種為利用兩個最值進行縮放，公式表達為：

使用preproccessing庫的MinMaxScaler類對數據進行區間縮放的代碼如下：

$$x' = \frac{x - Min}{Max - Min}$$

專題習題實作相關使用套件

繪圖套件

- plotly
- matplotlib
- seaborn
- Basemap
- Wordcloud

資料處理套件

- Numpy
- Pandas
- Pandas.Profiling
- Sklearn.preprocess

期末專題主題題目 - Covid-19-可視化和比較

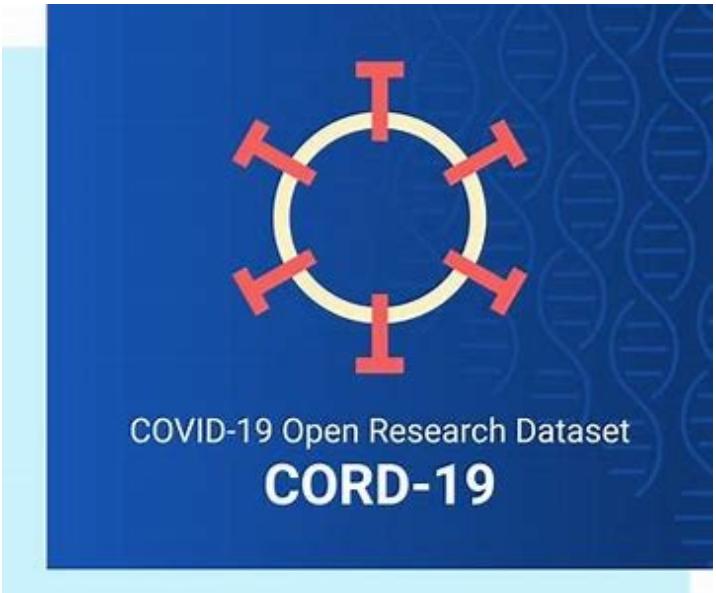
Covid-19 - 可視化和比較

網站：[COVID-19 Open Research Dataset Challenge (CORD-19)]

為了應對 COVID-19 大流行，白宮和主要研究小組的聯盟已經準備好了 COVID-19 開放研究數據集（CORD-19）。該免費可用的數據集已提供給全球研究社區，以應用自然語言處理和其他 AI 技術的最新進展來產生新見解，以支持正在進行的抵抗這種傳染病的鬥爭。

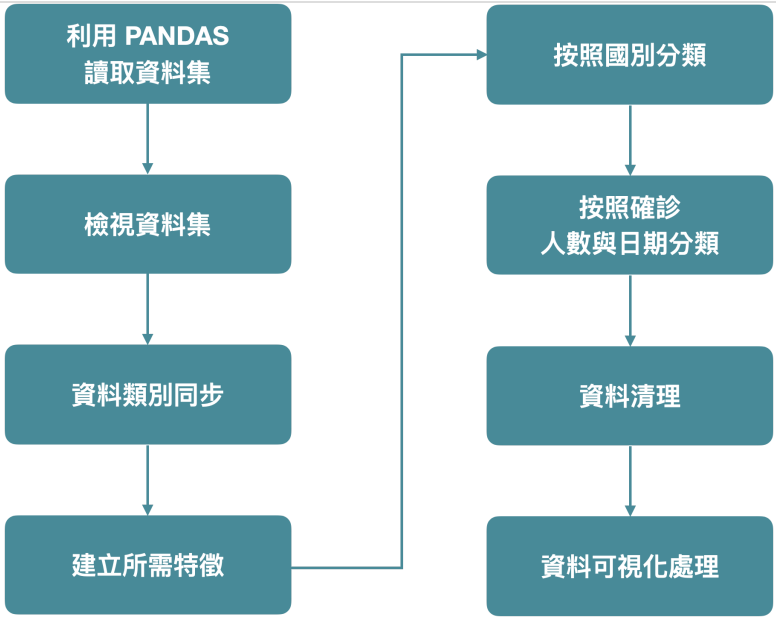
可以對此數據集執行的一些有趣的問題（任務）

1. 了解不同國家/地區數據及分析病毒的分布與感染的區域
2. 病毒治癒人數比例 - 是否與天氣, 環境, 體質相關
3. 病毒分布區域可以提供規避路徑規劃



重點欄位說明

	日期	區域	國家	最後更新時間	確診	死亡	治癒
	SNo	ObservationDate	Province/State	Country/Region	Last Update	Confirmed	Deaths	Recovered
0	1	01/22/2020	Anhui	Mainland China	1/22/2020 17:00	1.0	0.0	0.0
1	2	01/22/2020	Beijing	Mainland China	1/22/2020 17:00	14.0	0.0	0.0
2	3	01/22/2020	Chongqing	Mainland China	1/22/2020 17:00	6.0	0.0	0.0
3	4	01/22/2020	Fujian	Mainland China	1/22/2020 17:00	1.0	0.0	0.0
4	5	01/22/2020	Gansu	Mainland China	1/22/2020 17:00	0.0	0.0	0.0



- Creating new feature "Active_case"
- Active_case = Confirmed - Deaths - Recovered

Coronavirus in the word :

最後更新日期	確診人數	追蹤	治癒	死亡
Last Update	Confirmed	Active cases	Recovered	Deaths
09/13/2020	28902753	8432593	19547423	922737

目標一：繪製出全球的感染與康復人數，以國家別區分

目標二：利用地理資訊圖表繪製出全球的感染與康復人數

目標三：隔離前與隔離後的影響分布

目標四：旅遊業與運輸業的影響

期末專題主題題目 - Netflix 電影資料集

Netflix 上列出的電視節目和電影

擎 Flixable。

在 2018 年，他們發布了一份有趣的報告，顯示 Netflix 上的電視節目數量自 2010 年以來幾乎翻了三倍。流媒體服務的電影數量自 2010 年以來已減少了 2,000 多部電影，而其電視節目數量卻幾乎翻了三倍。探索可以從同一數據集中獲得哪些其他所有見解將很有趣。可以對此數據集執行的一些有趣的問題（任務）

1. 了解不同國家/地區提供的內容
2. 通過匹配基於文本的功能來識別相似內容
3. 演員/導演的網絡分析，並找到有趣的見解
4. 近年來，Netflix 是否越來越關注電視而不是電影。

Pandas Profiling

- pandas-profiling 能夠使用 DataFrame 自動生成數據的詳細報告
- 前 pandas-profiling 目前只支持導出 html 格式的文件。如果想要生成圖片，先生成的 html 文件，使用 Chrome 的內建截屏功能來生成圖片

以下統計信息（如果與列類型相關）將顯示在交互式 HTML 報告中：

- **導入 pandas-profiling**
- import pandas_profiling

型別

- pandas_profiling.ProfileReport(data)
- 導出報告
- pfr = pandas_profiling.ProfileReport(data)
  pfr.to_file('report.html')

[提示]

用 Pandas 讀入時有時需要注意encoding參數

1.1 引入 Pandas Profiling

1.2 需要注意異常值與缺失值的處理，注意資料區間，評估值區間差異過大的問題

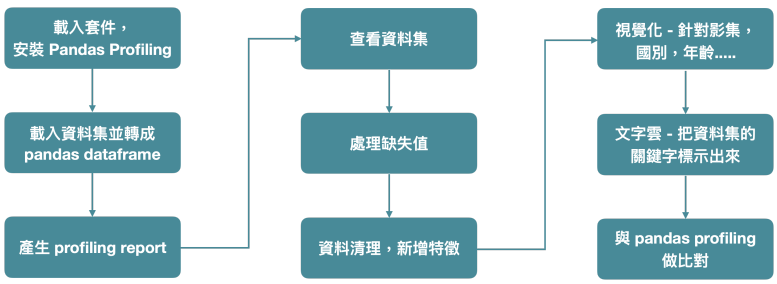
1.3 資料的分類，以期可以分別繪製比對圖形；

[基本目標]

把 Netflix 的資訊分門別列出來

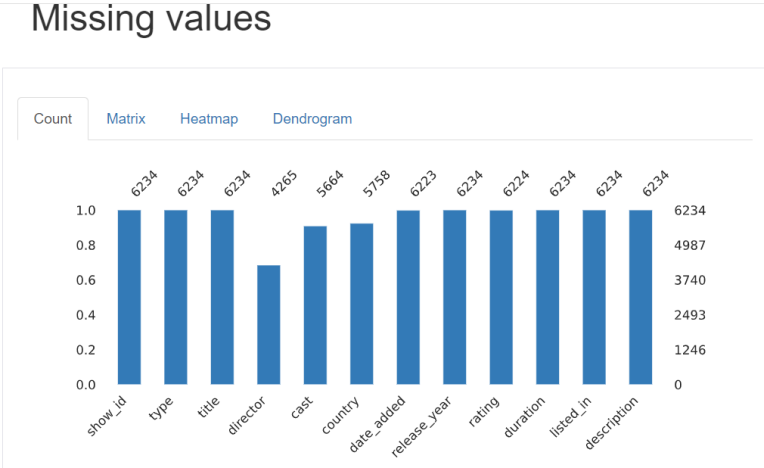
[進階目標]

畫出 Heatmap 與 文字雲



	影片ID show_id	type	片名 title	編劇 director	cast	country	上映時間 date_added	release_year	rating	長度 duration	listed_in	簡述 description
0	81145628	Movie	Norm of the North: King Sized Adventure	Richard Finn, Tim Maltby	Alan Marriott, Andrew Toth, Brian Dobson, Cole...	United States, India, South Korea, China	September 9, 2019	2019	TV-PG	90 min	Children & Family Movies, Comedies	Before planning an awesome wedding for his gra...
1	80117401	Movie	Jandino: Whatever it Takes	NaN	Jandino Asporaat	United Kingdom	September 9, 2016	2016	TV-MA	94 min	Stand-Up Comedy	Jandino Asporaat riffs on the challenges of ra...
2	70234439	TV Show	Transformers Prime	NaN	Peter Cullen, Sumalee Montano, Frank Welker, J...	United States	September 8, 2018	2013	TV-Y7-FV	1 Season	Kids' TV	With the help of three human allies, the Autob...
3	80058654	TV Show	Transformers: Robots in Disguise	NaN	Will Friedle, Darren Criss, Constance Zimmer, ...	United States	September 8, 2018	2016	TV-Y7	1 Season	Kids' TV	When a prison ship crash unleashes hundreds of...
4	80125979	Movie	#realityhigh	Fernando Lebrija	Nesta Cooper, Kate Walsh, John Michael Higgins...	United States	September 8, 2017	2017	TV-14	99 min	Comedies	When nerdy high schooler Dani finally attracts...

缺失值的統計與分類



目標一：# TV shows 跟 Movies 的比例；年齡層的分類

目標二：# 製作國別與發佈內容

目標三：建立內容類型的數據框的直方圖與 Heatmap

目標四：建立文字雲分析 - 找出最多人看的影片

延伸閱讀

[資料分析套件 - pandas-profiling]

緣起 - 每拿到新資料時，總用 pandas 做一些重複性的探勘工作，今天發現一個好套件：**pandas-profiling**，套件作者覺得 describe 實在是太陽春了，用這個一鍵幫你完成以下初步的資料分析。

- **Essentials**：type, unique values, missing values
- **Quantile statistics**：minimum, Q1, median, Q3, maximum, range, interquartile range

- Most frequent values
- Histogram
- Correlations heatmap(Pearman and Pearson)

本文

安裝(擇一)

```
pip install pandas-profiling
conda install pandas-profiling
```

需求
目前是連網版，需要網路連線下載一些Bootstrap跟JQuery。

準備好資料

```
from sklearn.datasets import load_boston

data = load_boston()["data"]
cols = load_boston()["feature_names"]
df = pd.DataFrame(data=data, columns=cols)
```

丟進去分析

```
profile = pandas_profiling.ProfileReport(df)
profile.to_file(outputfile="output.html")  #支援輸出html
```

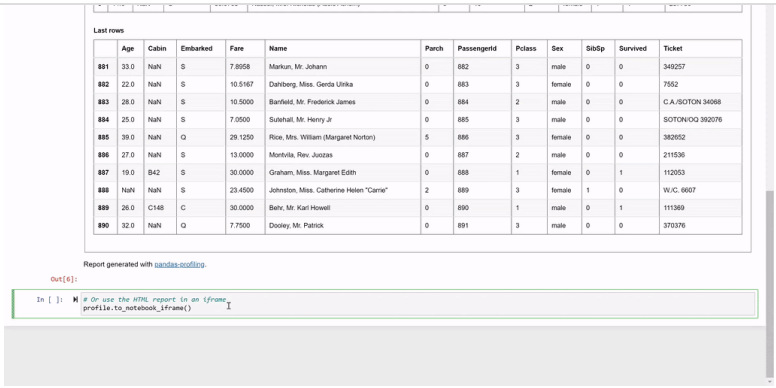
This is achieved by simply displaying the report. In the Jupyter Notebook, run：

profile.to_widgets()

The HTML report can be included in a Jupyter notebook：

Run the following code：

profile.to_notebook_iframe()



Last rows

	Age	Cabin	Embarked	Fare	Name	Parch	PassengerId	Pclass	Sex	SibSp	Survived	Ticket
881	33.0	NaN	S	7.8958	Markun, Mr. Johann	0	882	3	male	0	0	349257
882	22.0	NaN	S	10.5167	Dahlberg, Miss. Gerda Ulrika	0	883	3	female	0	0	7552
883	28.0	NaN	S	10.5000	Banfield, Mr. Frederick James	0	884	2	male	0	0	C.A./SOTON 34068
884	25.0	NaN	S	7.0500	Sutehall, Mr. Henry Jr	0	885	3	male	0	0	SOTON/OQ 392076
885	39.0	NaN	Q	29.1250	Rice, Mrs. William (Margaret Norton)	5	886	3	female	0	0	382652
886	27.0	NaN	S	13.0000	Montvila, Rev. Juozas	0	887	2	male	0	0	211536
887	19.0	B42	S	30.0000	Graham, Miss. Margaret Edith	0	888	1	female	0	1	112053
888	NaN	NaN	S	23.4500	Johnston, Miss. Catherine Helen "Carrie"	2	889	3	female	1	0	W./C. 6607
889	26.0	C148	C	30.0000	Behr, Mr. Karl Howell	0	890	1	male	0	1	111369
890	32.0	NaN	Q	7.7500	Dooley, Mr. Patrick	0	891	3	male	0	0	370376

Report generated with pandas-profiling.

Out[6]:

In [ ]:
```
# Or use the HTML report in an iframe
profile.to_notebook_iframe()
```

Python輕鬆實現地圖可視化(附詳細源碼)

pyecharts

首先，必須說說強大的 pyecharts 庫，簡單易用又酷炫，幾乎可以製作任何圖表。pyecharts 有 v0.5 和 v1 兩個版本，兩者不兼容，最新的 v1 版本開始支持鏈式調用，採用 options 配置圖表。pyecharts 在製作地圖方面，包含 Map、Geo 和 Bmap 三類，使用 Map 類支持世界、國家、省市和區縣四級地圖，使用前需獨立安裝。so，pip 它們！

pip install pyecharts

pip install echarts-countries-pypkg

pip install echarts-china-provinces-pypkg

pip install echarts-china-cities-pypkg

pip install echarts-china-counties-pypkg

```
    .set_global_opts(
        title_opts=opts.TitleOpts(title="2019年各省GDP分布圖
單位:億元"),  #配置標題
        visualmap_opts=opts.VisualMapOpts(
            type_ = "scatter"    #散點類型
        )
    )
    .add("GDP",list,maptype="china")   #將list傳入，地圖類型為中
國地圖
    .render("Map1.html")
)
```

運行以上代碼，用瀏覽器打開生成的Map1.html，效果如下：

利用python構建一個推薦系統，這個技術是出了名的強大！

推薦系統的目的是通過發現數據集中的模式，為用戶提供與之最為相關的信息。當你訪問 Netflix 的時候，它也會為你推薦電影。音樂軟體如 Spotify 及 Deezer 也使用推薦系統進行音樂推薦。

舉個簡單的例子，如果要向個用戶推薦一部電影，那麼一定是基於他/她的朋友對這部電影的喜愛。基於協同過濾的推薦又可以分為兩類：啟發式推薦算法（Memory-based algorithms）及基於模型的推薦算法（Model-based algorithms）。啟發式推薦算法易於實現，並且推薦結果的可解釋性強。啟發式推薦算法又可以分為兩類：

1. 基於用戶的協同過濾（User-based collaborative filtering）

推薦系統構建

我們將使用movielens構建一個基於項目相似度的推薦系統，首先導入pandas和numpy。

```
import pandas as pd
import numpy as np
import warnings
warnings.filterwarnings('ignore')
```

接下來利用pandas中的read_csv()對數據進行加載。數據集中的數據以tab進行分隔，我們需要設置sep = t來指定字符的分隔符號，然後通過names參數傳入列名。

```
df = pd.read_csv('u.data', sep='\t',names=
['user_id','item_id','rating','titmestamp'])
```

接下來，檢查正在處理的數據。

```
df.head()
```

相比只知道電影的ID，能看到它們的標題更為方便。接下來，下載電影的標題並將它們整合到數據集中。

下一步：閱讀範例與完成作業