CSS 是 HTML 文件擁有顏色、特色及風格的原因！ CSS 是讓不同類型的 HTML 內容披上特定外觀的語言，可以將 CSS 看作一組視覺呈現規則，告訴瀏覽器對每個 HTML 元素在網頁要長成什麼樣子。

例如，透過 CSS 可以將圖 1-4 的內容變成圖 1-5 的樣子（文字顏色變淡）。

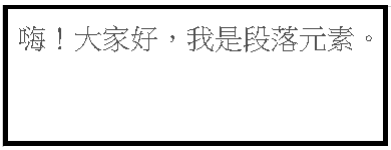

圖 1-5：在瀏覽器中使用 CSS 樣式渲染 div 和段落元素

在社交平台的資料中，CSS 通常用於確保相同類型的元素有一致的風格，例如，在推文時間軸（timeline）上，每條推文的時間戳記需要以相同的字體、顏色和大小顯示。

有好幾種方法可以為 HTML 標籤指定 CSS 樣式，其中一種是內聯（inline）的 CSS，它是在建立 HTML 標籤的同一列程式中指定 CSS，可以參考清單 1-1 的例子。

```
<div ❶style="❷color: #727272;">
    <p> 嗨！大家好，我是段落元素。</p>
</div>
```

清單 1-1：使用內聯 CSS 設定 標籤的樣式

以這個範例來說，是將一組屬性（attribute）加到 div 元素的起始標籤裡。屬性是指與該 HTML 標籤相關聯的其他資訊，屬性名稱跟隨在左角括號（<）和標籤名稱（本例 div）之後、右角括號（>）之前，屬性名稱後面跟著一個等號（=），接著是屬性值，該值由雙引號（也可使用單引號）括住，屬性通常是代表其所在的標籤之特性，而屬性的影響會向下傳遞給被嵌套的 HTML 元素。以此例而言，div 元素擁有內聯 CSS 的 style 屬性 ❶，表示 標籤內的所有內容都必須遵循此 style 屬性所定義的 CSS 樣式規則，由於段落元素被嵌套在 div 元素內，因此，段落元素及其內容會繼承此 div 的所有樣式。

想體會上述說明，可以參閱 BuzzFeed 的新聞故事《Inside the Partisan Fight for Your News Feed》（https://buzzfeed.com/craigsilverman/inside-the-partisan-fight-for-your-news-feed），該專案由筆者及 Craig Silverman、Jane Lytvynenko、Jeremy Singer-Vine 等人，藉由臉書的 Graph API 從 452 個不同頁面收集 400 萬條貼文，面對數百萬個資料點，無法輕易地分析所有資料，始終無法找到任何有意義的模式或趨勢，讓我們不知如何是好，想要進行分析，必須先縮小資訊範圍。

由於越來越多的新聞機構依靠像臉書這類第三方媒體來吸引觀眾，所以，此專案深入研究這些機構（無論新舊）在臉書上的較量，我們決定依照追蹤者數量以及每頁獲得的參與度（反應和評論），分析左派和右派新聞受歡迎的程度，一旦將資料縮小為兩類，就能隨時間繪製如圖 2-3 的資訊走勢圖。

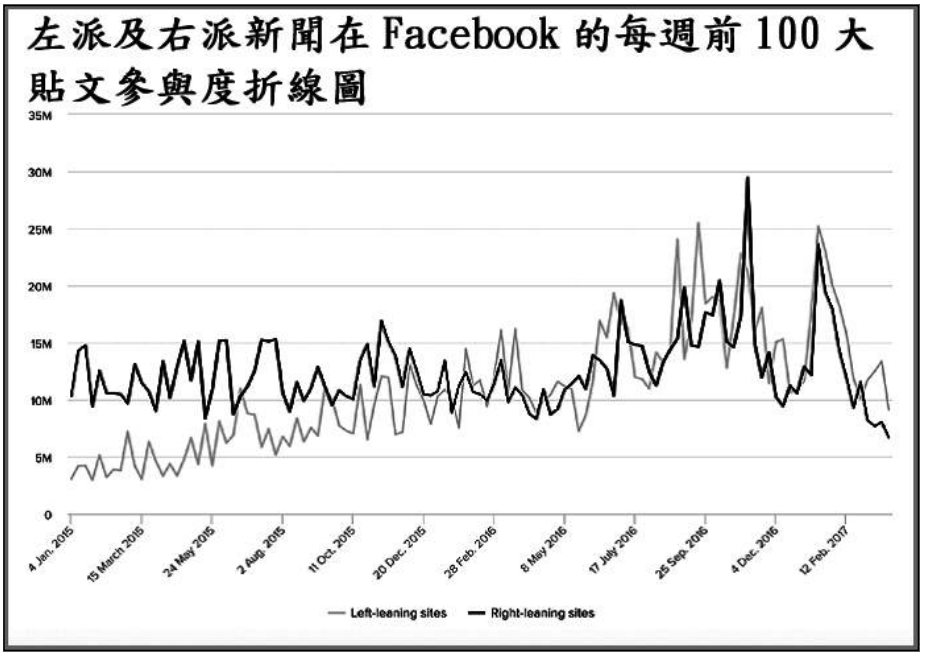

圖 2-3：顯示 BuzzFeed News 分析左派及右派新聞在臉書頁面參與度的圖表

可以看到隨著時間推移，左派新聞的臉書頁面參與度增加了。

好的，要如何取得其中的資料呢？這需要閱讀 API 文件才能瞭解有哪些資料可用，像 Google 這種大公司會為 API 準備各種說明文件，我們關心的是 YouTube 的資料 API，該文件位於以下網址：

https://developers.google.com/youtube/

每個 API 文件的編寫風格不盡相同，使用前，需要先閱讀簡介或導覽。在介紹如何使用此文件之前，先複習一下 Google YouTube API 的一些基礎知識。

提升 API 回傳結果

有許多參數可以進一步縮小範圍或指定待收集的資訊類型，前進到 YouTube API 文件[1]，將畫面向下捲動至參數表（Parameters），該表左欄按名稱列出參數，右欄則提供參數說明和使用指南。查找資料時，請閱讀每個參數的說明，找出與欲取得的資料類型相匹配的參數，假設想將 API 的回傳結果縮小到僅涉及「cake」單字的影片，為了提升 API 的處理結果，使用參數 q（query 的縮寫）攜帶要搜尋的單字，也就是在瀏覽器輸入：https://www.googleapis.com/youtube/v3/search?channelId=UCJFp8uSYCjXOMnkUyb3CQ3Q&part=snippet&key=<*YOUR_API_KEY*>&q=cake。（記得將 <*YOUR_API_KEY*> 換成你自己的 Youtube API 金鑰）

這裡針對此 URL 稍作解說，前半部分與本章第一個 API 呼叫方式雷同，是使用 API 的 search（搜尋）功能，並經由參數 channelId 限制搜尋範圍為 BuzzFeed Tasty 頻道的影片，接下來和之前一樣指定 API 金鑰，後面再跟著輸入「&」字元，並增加參數「q」，在 q 之後使用等號（=）及指定 API 要搜尋的單字「cake」。把此呼叫 API 的 URL 輸入瀏覽器，應該得到一組 JSON 回應，該回傳文件只包含說明文字或標題帶有單字「cake」的影片。

很好！相信讀者現在已經瞭解如何使用參數來客製 API 資料請求了。

1. https://developers.google.com/youtube/v3/docs/search/list

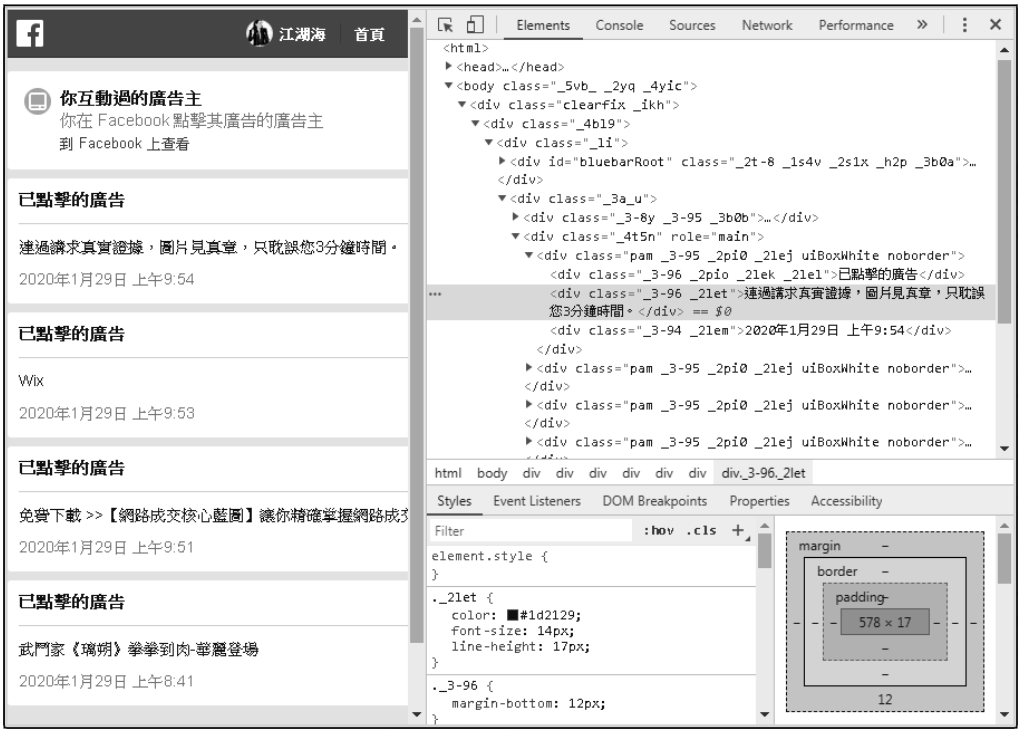

圖 4-2：Chrome 的網頁檢視器範例

網頁是一種 HTML 檔案，它將資料包在 HTML 標籤裡，並利用 CSS 的 ID 和類別（class）名稱為其設置外觀風格，當頁面呈現重複性內容時，可能為每條資料套用相同模式的 HTML 標籤和 CSS 類別，例如精選的新聞貼文或資料副本中的廣告清單。為了收集這些包含在 HTML 標籤裡的資料，必須要能識別及理解這種編碼模式。

將網頁轉化成為可處理的資料

以臉書資料副本裡已點擊的廣告為例，每個廣告被包在 標籤內，此標籤的 class 屬性值為「_4t5n」和 role 屬性值為「main」，其 HTML 原始碼範例如清單 4-1 所示。

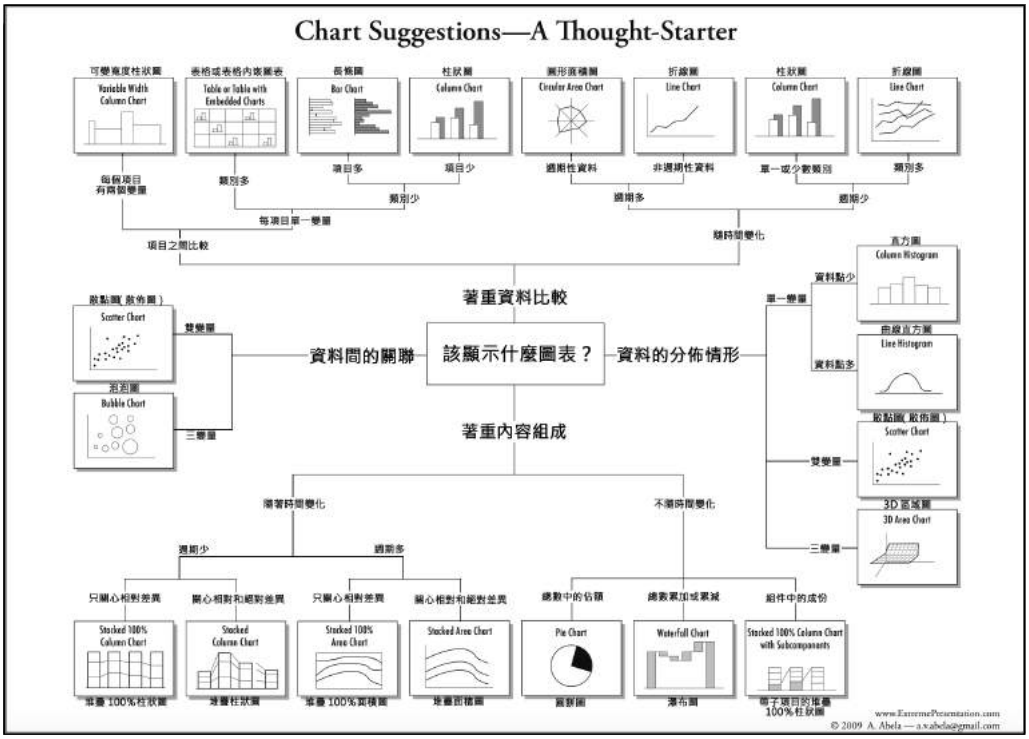

圖 7-2：Andrew Abela 於 2012 年提出的圖表選用建議指引

現在要說明的是不同的圖表類型及其使用方式。首先，比較型圖表（comparison）用在比較資料集的差異，如前一章比較了機器人和自然人的資料集。

圖 7-3 的柱狀圖（column chart）是常見的比較型圖表，它繪製了圖 7-1 合併後的資料透視表之圖表（本章稍後將介紹如何製作此類圖表）。

使用 Jupyter Notebook

之前章節是透過命令列界面（CLI）執行 Python 腳本程式，對我們來說，這是一種認識程式語言的便捷好方法。

但為了增進我們的 Python 技能，並開始使用更複雜的腳本，應該研究如何讓這些項目更易於管理、結構化和共享的工具，因為，當腳本變得又長又複雜時，要追蹤資料分析的每個步驟就更加困難。

此時，學習使用 Jupyter Notebook 將有所助益，Jupyter 是一套在本機電腦執行的開源 Web 應用程式，而它的畫面則顯示在 Chrome 之類的瀏覽器上，筆記紙（notebook）可以一次執行含有幾列程式碼的區塊，方便使用者漸進地調整、改進部分程式碼，這套 Jupyter Notebook Web 程式是由 IPython Notebooks 演變而來，建構之初是為了調和 Julia、Python 和 R 三種程式語言，故取名為 Jupyter(**Ju-Pyt-R**），但此後已發展成支援數十種程式語言的執行環境。

各領域的資料科學家都有使用 Jupyter Notebook，包括為提高網站性能而努力工作者、研究人口統計學的社會學家，以及應用資訊自由法（FOIA）探索所取得資料的趨勢和異常信號之新聞從業人員。使用 Jupyter Notebook 有一個絕佳好處，就是許多資料科學家和研究人員將他們的筆記紙（通常伴有註解和詳細分析說明）放在 GitHub 之類的程式碼分享平台，使初學者更容易重作他們的研究。

建置虛擬環境

為了使用 Jupyter Notebook，需要藉由學習三個重要概念，將程式撰寫技能提高到更上一層水準。

首先，需要具備建立和應用虛擬環境的能力。要搞清楚虛擬環境可能不容易，且以大方向來審視其目的。

就像過去幾章所學到的，要使用函式庫時，就必須在 CLI 以命令進行安裝，每個函式庫都安裝到電腦上的預設目錄，除非將它移除安裝，不然會一直保留在該目錄。

簡述研究方法

分析步驟如下：

1. **篩選資料並將它分為兩個資料框：**第一個資料框是提交內容帶有 vaccine、vaccinate 或 vaccination 的所有貼文；第二個資料框是內容未提及前述單字的貼文，作為第一個資料框的對照組。
2. **在每個資料框上進行簡單的計算：**藉由計算參與度計數（engagement counts）的平均數及中位數取得比較基值，可以比較容易理解 r/askscience 資料的每個子集，及為研究的問題制訂答案標準。此分析的參與度計數由評論貼文的筆數和按讚次數加總而得。

在此說明一下本節會講到的術語。

平均數：取得資料集裡的所有值，將這些值加總，再將加總的結果除於值的個數；

中位數：在整個資料集裡，出現在中間位置的數值，為了找出中位數，需要先排序資料集裡的值（由小至大或由大到小），排序後，中位數就是恰好位於中間位置的值，若有偶數個值，則取中間兩個位置的值之平均。

平均數和中位數均為集中趨勢量數，可以藉由檢驗這些指標來評估資料集趨勢，對於離群值（outlier）不多的大型資料集，平均數是衡量集中趨勢的好方法，反之，若離群值比例較高的資料集，利用中位數可以得到更好的量測結果，本章的分析將同時使用這兩種量測方法。

縮小資料範圍

就算只查看 Reddit 單個看板的貼文數，資料量仍然非常多，雖然盡可能從完整資料集下手是很重要，但根據專案目標篩選資料，可以提供更好、更整潔的資料概覽，還可以減少每次運算所花的時間。

本章也說明母體和取樣資料的概念。**母體：**即整個群組的資料集，本章的群組即指 2014 至 2017 年之間在 r/askscience 看板發布的所有貼文；**取樣資料：**顧名思義，就是資料集的子集或樣本。本習作將有兩個子集，一個是與疫苗接種有關的貼文組成（後面會定義），另一個由其他貼文組成，我們將對這兩個資料子集進行分析與比較。

這裡會使用第 8 章建置的虛擬環境和 Jupyter Notebook 專案，接下來的練習打算由現有的筆記紙繼續擴充，將繼續延用該筆記紙已建立的變數。

選擇特定欄位的資料

要為此任務篩選資料，首先是將資料集縮減到只與分析相關的欄位，接著濾掉不當的樣本，例如含有 null（空值）的紀錄。

先從選擇所需的欄位開始，我們在意兩種不同類型的資料：貼文標題（提交到 r/askscience 的文字），以及對此貼文的回應。如前一章所介紹的，利用下列程式即可得到欄位名稱的清單：

```
ask_science_data.columns
```

應該看到如圖 10-1 所示的前五筆資料。

```
tweets.head()
```

	tweetid	userid	user_display_name	user_screen_name	user_reported_location	user_profile_description	user_profile_url	follower_count	follo
0	533622371429543936	299148448	Maria Luis	marialuis91	Nantes, France	journaliste indépendante/un vrai journaliste e...	NaN	8012	
1	527205814906654721	299148448	Maria Luis	marialuis91	Nantes, France	journaliste indépendante/un vrai journaliste e...	NaN	8012	
2	545166827350134784	299148448	Maria Luis	marialuis91	Nantes, France	journaliste indépendante/un vrai journaliste e...	NaN	8012	
3	538045437316321280	299148448	Maria Luis	marialuis91	Nantes, France	journaliste indépendante/un vrai journaliste e...	NaN	8012	
4	530053681668841472	299148448	Maria Luis	marialuis91	Nantes, France	journaliste indépendante/un vrai journaliste e...	NaN	8012	

5 rows × 31 columns

圖 10-1：所載入的資料框

誠如所見，此資料副本保存與推文相關的大量詮釋資料，每一列代表一條推文，裡頭包含有關推文本身的內容，以及發布推文的使用者資訊。還記得第 8 章以清單方式查看每個欄位名稱的作法吧！可以使用下列程式：

```
tweets.columns
```

執行該單元格後，會看到如下清單：

```
Index(['tweetid', 'userid', 'user_display_name', 'user_screen_name',
       'user_reported_location', 'user_profile_description',
       'user_profile_url', 'follower_count', 'following_count',
       'account_creation_date', 'account_language', 'tweet_language',
       'tweet_text', 'tweet_time', 'tweet_client_name', 'in_reply_to_tweetid',
       'in_reply_to_userid', 'quoted_tweet_tweetid', 'is_retweet',
       'retweet_userid', 'retweet_tweetid', 'latitude', 'longitude',
       'quote_count', 'reply_count', 'like_count', 'retweet_count', 'hashtags',
       'urls', 'user_mentions', 'poll_choices'],
      dtype='object')
```

就我們的目的，重要的欄位是「hashtags」和「tweet_time」，hashtags 欄將每條推文使用的所有主題標籤以文字清單方式呈顯，中括號裡的標籤文字彼此以逗號（,）分隔。雖然它們遵循資料清單的格式，但 Python 會將它視為一個長字串，圖 10-2 的範例是資料集第 359 筆推文，使用的主題標籤是「Impeachment」和「MuellerMonday」，並以長字串「[Impeachment, MuellerMonday]」格式儲存。請注意，並非每條推文都有用到主題標籤，而我們只會分析那些有標籤的部分。

```
tweets.iloc[358]

tweetid                                                  1026465539835785216
userid                   fa345559085c3eefd96303a1378c1a6164a036b0e24472...
user_display_name        fa345559085c3eefd96303a1378c1a6164a036b0e24472...
user_screen_name         fa345559085c3eefd96303a1378c1a6164a036b0e24472...
user_reported_location                                          Delaware, USA
user_profile_description Progress is impossible without change, and tho...
user_profile_url                                    https://t.co/i2omiuAU7S
follower_count                                                         1341
following_count                                                        1774
account_creation_date                                            2018-01-13
account_language                                                         en
tweet_language                                                           en
tweet_text               #Impeachment: Last episode, arrestment of Trum...
tweet_time                                                 2018-08-06 13:50
tweet_client_name                                        Twitter Web Client
in_reply_to_tweetid                                                    NaN
in_reply_to_userid                                                     NaN
quoted_tweet_tweetid                                          1.02495e+18
is_retweet                                                           False
retweet_userid                                                         NaN
retweet_tweetid                                                        NaN
latitude                                                               NaN
longitude                                                              NaN
quote_count                                                              0
reply_count                                                              0
like_count                                                               1
retweet_count                                                            0
hashtags                                      [Impeachment, MuellerMonday]
urls                     [https://twitter.com/i/status/1024946380857458...
user_mentions                                                          NaN
poll_choices                                                           NaN
includes_trump_or_clinton                                            False
Name: 358, dtype: object
```

圖 10-2：使用 .iloc [] 方法顯示 tweet 資料框的第 359 列的內容

將資料轉換成 datetime 格式

利用篩選後的資料集計算特定時段內有關川普或柯林頓的推文之數量，這種計數通常稱為時間序列。為此，需要將資料格式轉換為時間戳記，並使用 pandas 函式根據這些時間戳記進行統計。

如同第 6 章使用 Google 試算表進行探險時所看到的，為程式碼指定要處理的資料型別是重要的，儘管 Google 試算表和 pandas 可以自動偵測整數、浮點數和字串等資料型別，但也可能出錯，最好明確告知資料型別，不要跟它賭運氣，其中一種方法是為每一條推文選擇具時間戳記的欄位，並告訴 Python 以 datetime 的資料型別處理這類資料。

先來看看 pandas 如何詮釋資料欄位，這裡使用第 8 章看過 dtypes 屬性來檢視資料集的某些特性，就是檢查每一欄的資料型別：

```
tweets_subset.dtypes
```

如果在單元格執行此段程式，Jupyter 筆記紙會顯示資料框的欄位及其資料型別：

```
tweetid                      int64
userid                      object
user_display_name           object
user_screen_name            object
user_reported_location      object
user_profile_description    object
user_profile_url            object
follower_count               int64
following_count              int64
account_creation_date       object
account_language            object
tweet_language              object
tweet_text                  object
tweet_time                  object
tweet_client_name           object
in_reply_to_tweetid        float64
in_reply_to_userid          object
quoted_tweet_tweetid       float64
```

將資料繪成圖表

為了更全面地瞭解這些資料，使用本章前面安裝及匯入的 matplotlib 函式庫為 pandas 資料框繪製圖表，使本專案更趨完美，因為視覺化的時間序列會讓結果更加清晰及易於解讀。

在本專案開始的時候，已經匯入 matplotlib 函式庫的 pyplot 功能模組，並以簡寫的 plt 代表。要存取其功能，請在 plt 之後接上要使用的函式，如下所示：

```
plt.plot(monthly_tweet_count)
```

上式，plot() 函式以 monthly_tweet_count 資料框作為參數，在 x 軸繪製資料框的日期，y 軸上繪製每月的推文計數，如圖 10-5 所示。

NOTE 在 matplotlib 裡有許多種自定繪圖的方法，想要瞭解進一步資訊，請瀏覽 https://matplotlib.org/。

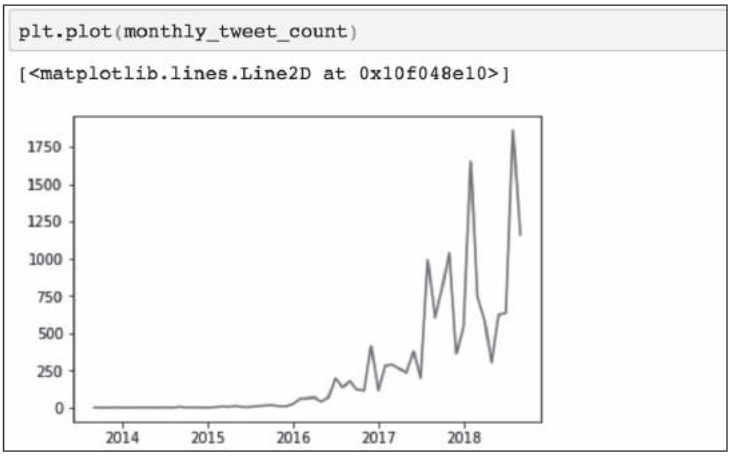

圖 10-5：在 Jupyter Notebook 利用 matplotlib 建立的圖表