StreetVoice
排行榜
爬蟲與資料分析

Collaborator：李季澄

涂譽寶

陳奕勛

陳昶安

李佳恩

賀智暉

動機：

之所以會選擇這項主題作為專案，是因為我們當中，有一個人的興趣和音樂相關，也在去年夏天成立了自己的樂團。

而在成立樂團初期，團員們曾討論過 2 種不同的發歌策略。**第一種是**，把手上有的 3、4 首 Demo，在短期內都發布到音樂串流平台上。**第二種則是**，把每一首歌都修到最好，處理完細節、混完音，再每幾個月發布一首歌曲。

兩種方式各有優缺。第一種方法，雖然歌曲品質上會較為粗糙，但短時間內持續地發布歌曲確實有機會為樂團帶來很大的流量、累積粉絲。第二種方法，雖然可能每發不完一首歌，就得沈寂一段時間，但歌曲品質好，每一首都比較耐聽，也是累積粉絲的另一種方式。

這樣的情況下，其實再怎麼開會討論，都很難知道哪一個方案是最好的。即便當下團員們還是達成了共識，但沒有人敢肯定這麼做是對的，也促成了我們組員提出做音樂排行榜資料分析的提案。

問題剖析：

有了問題之後，我們便開始發想要如何用程式來解決這個問題。一開始是希望能畫出一張圖明確地，讓人一看就知道，到底應該要高頻率地發 Demo，或是採用量少質精的發佈策略。

然而，經過討論後，我們發現我們其實很難直接得到結論，而最大的原因就是**單位的不同**。頻率有很多計算方式，像是一年發佈幾首歌、一個月發佈幾首歌等。品質也有不同的標準，像是歌曲愛心數、用播放次數除以愛心數粗估的循環播放次數以及音樂平台頒布的獎項等都可以做為品質的衡量指標。而當頻率與品質沒有一個衡量的基準時，我們就很難得出一個準確的答案。

後來我們覺得其實我們不一定要有一個結論是能很明確的告訴創作者怎麼做會比較好，但我們可以**讓創作者知道目前能登上排行榜的歌手們，他們的發佈頻率、歌曲品質分別和粉絲人數呈現什麼樣的關係**。

程式設計：

首先是發布頻率的部分，我們以**每年平均發布的歌曲數量為 X 軸**（以現在這個時間減去某創作者第一首歌發佈的時間，換算成年數後作為分母），**粉絲人數為 Y 軸**，去除**離群值**(outlier)**後**，畫出簡單迴歸的圖。

再來是歌曲品質，我們則畫了兩張圖。第一張圖是以**某創作者所有歌曲的人均循環播放次數為 X 軸**（這裡假設所有聽過的人都會按愛心，以播放次數除以愛心數算出人均循環播放次數），以**粉絲人數為 Y 軸**，去除離群值後，畫出簡單迴歸圖。第二張圖則是以**某創作者每首歌曲的平均獲獎數為 X 軸**（StreetVoice 官方會針對好的歌曲頒布獎項，有些歌曲沒有得獎，有些歌曲可以包辦 3、4 個獎項），**粉絲人數為 Y 軸**，一樣去除離群值後，畫出簡單迴歸圖。最後我們也**以發布頻率為 X1，人均循環播放次數為 X2，獲獎數為 X3，粉絲人數為 Y**，畫出三維的複迴歸的模型，希望能透過這個模型再推論出一些端倪。

為了完成以上這些任務，我們主要要學習的就是**網路爬蟲**以及**繪製統計圖**的能力。爬蟲的部分，我們使用了不少套件。像是用 request 取得網頁的原始碼，**用 lxml 找出片段原始碼的** xpath 以取得我們真正要用到的資料，**用 re 來在較雜亂的** xpath **中使用正則表達來準確地取得某幾項數據**，最後再**用** pandas **來把這些資料存成** csv **檔**，讓下一階段的組員能用這些資料來畫圖。

而繪圖的部分，我們則是使**用** csv **來開啟檔案，用** datetime **來處理爬蟲取得的日期資料**，再使**用** matplotlib **畫出這些資料的迴歸圖**。

執行結果：

看到最終結果執行結果時，大家都吃了一驚。原本我們都覺得這些變數的關係都是成正比，只是想知道具體的圖形會長什麼樣子。但沒想到的是，**發布頻率和粉絲人數的關係竟然是成反比**。算是讓我們得以推測出**量少質精的策略是比持續曝光還來得好的**。

除了劃出圖形外，我們也計算出 SSR、SST、**判定係數**等統計數據，協助我們了解這些模型的解釋強度。知道解釋強度後，我們也能進一步思考這樣的模型

是否有足以解答我們的問題，又應該要怎麼去修正統計過程來獲得更接近真實情況的結果。

這次的結果可說是讓我們了解了程式設計搭配數據分析是多麼強大的事情，在做專案之前，樂團會為了兩個看似都很合理的策略不斷地討論。然而，兩個策略會產生的結果可說是天差地別，若是走錯了這步，對於樂團的發展會造成相當大地傷害。但執行專案後，樂團就能在開會討論部分節省不少時間，也會有更好的發展。

心得：

其實在決定要做專案之後，我便常在想專案可以做什麼。過了一、兩個月，因為聽了幾場企業說明會，發現處理及分析大量資料會是未來找工作必須具備的重要能力，又有在別堂課聽到教授提到爬蟲這個名詞，我便燃起了對資料分析的興趣。而我又想到，暑假樂團在開會時碰到的問題，因此決定向組員提出分析音樂排行榜的建議。

雖然分工上，每個人被分配到要寫的程式並不多，但為了能在期末報告時能順利地把專案講述給教授及同學聽，我參與了大部分程式撰寫及 debug 的過程。學習爬蟲的過程中，真的發現“自學”是我很缺乏的一項能力，常常在一個地方卡關後，就不知道下一步該如何是好。

但經過這次經驗，我有比較知道該怎麼搜關鍵字、怎麼去吸收網路上五花八門的資源，未來也希望自己能利用時間繼續寫專案，持續精進程式以及自學的能力。另外，這次專案比較可惜的，應該是後續資料分析的部分。因為我們都對統計不太擅長的關係，雖然有達到我們心中的效果，但最後的分析方式真的蠻粗糙的。這也讓我有了去修其他統計相關課程的動力，希望未來也能用更精確的分析方式，來找到問題最好的答案。

而繪圖的部分，我們在聽完助教意見後，也有試著把後續過程弄得更精細一些，這也會是我們以後想再努力的部分。總之，能完成這份專案我要非常感謝教授、助教以及和我一同成長的組員，我們每個人都是初學者，大家都很努

力，雖然期末報告的結果未臻理想，但能完成這些專案我還是很有成就感，也因為互相討論，讓我從他們身上學到了很多。