R語言與資料科學導論書面報告
RECOMMEND THE COCKTAILS TO YOU

第十八組 無名南瓜走開
財金一 蘇以恩
經濟一 吳京
圖資二 許青純
圖資二 羅俊欣

2019-01-12

目錄

1.簡介

(1) 構想

我們想要利用Ｒ能夠快速調閱資料庫，方便處理dataframe格式物件的優點，做出一個一個能夠在使用者輸入一段文字之後，根據分析文字情緒的結果向使用者推薦調酒，或因應使用者個人喜好，協助他/她選出符合自己需求的調酒的程式，為什麼想要做出這麼樣一個程式呢？原因是這樣的：

酒在我們的生活中扮演著重要角色，有時是在社交場合中重要的催化劑、有時作為三五好友聚會的最佳配角，又或者在某個夜深人靜時分，酒精在心上撕開一個口子，任情緒傾洩而出。而不同於普通的酒，調酒的滋味隨其五花八門的基底與調料改變，簡單一杯調酒，調的卻是人生的百態紛呈。

不過雖然調酒的世界博大精深，卻容易受限於場地與知識，在酒吧以外的地方難以好好品嘗和真正了解眾多調酒的真實樣態，因此，如果有一款能向使用者推薦調酒的程式，對調酒初心者或選擇困難的人來說是有幫助的。

(2) 說明

使用者透過程式可以根據當下心情或自身偏好，自行選擇出想喝的調酒，或者由程式隨機給使用者推薦五款調酒，使用者再自行從這上天決定的五款酒裡挑出自己想喝的。

2.方法

2.1 資料取得

首先，我們要建立調酒的資料庫。我們選擇的是 Social and Cocktail 這個調酒網站，這個網站上儲存了約1000種不同種類的調酒，而且已經依不同種類分類過（排名前100、使用的玻璃杯、不同的風味等），這在日後可以讓我們可以比較好對於資料庫裡面的調酒再進行地更細部的分類。同時，因為此網頁設計的關係，我們可以比較輕易的從這個網站抓下每個調酒的成分及容量，所以最終決定使用這個網站作為我們資料的來源。

2.2 原始碼運作說明

R中使用到的套件：httr、tidyr、dplyr、shiny、readr、DT、shinythemes

(1) 爬蟲與固定格式 (python)

我們利用python從網站上爬取所有調酒的名稱及配方，配方包含基酒種類與其添加的容量，準備依據這些成分的資料建立一個調酒資料庫。

但是，因為不同的液體（基酒或其他添加物，如果汁等）所使用的容量單位可能不同，所以我們需要先把所有的容量單位分類[1]，轉成固定格式[2]，方便等等統一容量單位。

(2) 基酒酒精濃度表 (R 、人工)

接下來，因為我們認為資料庫的其中一個變數應該要依酒精濃度分類，但此網站並沒有提供酒精濃度，卻剛剛好有依據基酒種類分類的列表（左手邊的列表），我們便只好把這些基酒（共258種）抓下來，並運用「人工智慧」，手動上網查詢他們各自的酒精濃度，依據各個基酒的名稱與其酒精濃度做成一個csv檔。

(3) 計算調酒總容量與酒精濃度 (python)

現在有了基酒酒精濃度表，我們只差統一容量單位，理論上就可以分類出所有調酒的各個變數了（酒精濃度、酸或甜的程度等）。我們使用 ml 作為統一單位，因為他在爬取的資料中最為常見。最後運用基酒酒精濃度表，用 python 算出所有調酒的總容量與酒精濃度[3]。

---

[1] 根據我們的觀察，網站上溶液的容量單位共有11種，除了 ml 以外，還有dash, teaspoon, splash, bottle, cup, bar spoon, shot, tablespoon, can, drizzle。之後將我們依據其換算成 ml 的比例，將其轉換成 ml，藉以統一容量單位。

[2] [成分名稱]:[容量值]([單位名稱]) 。如：Orange Bitters:2.0(dash)，盡量保存原始資訊；若單位名稱是 ml，則為[成分名稱]:[容量值]。如：Tequila:45.0，因為 ml 為出現最多次的容量單位，且以後會用 ml 作為統一單位之基準。

[3] 使用 python 的原因，是因為我們目前運用使用多層迴圈計算，在使用 R 使造成電腦當機，之後優化程式時應該嘗試解決這個問題。

(4) 最後清理與資料庫最終搭建 (人工、R)

接下來用 R 合併成同一個 dataframe，然後根據調酒中特別的成分（例如檸漿等）將調酒依味道分類，使用 shiny 產出 GUI。另外，因為 Social and Cocktail 這個網站上仍舊有填寫疏漏[4]，我們最後採取人工方式刪除那些在成分上有填寫錯誤的調酒[5]。

	cocktails_name	X1	ingredient_1	ingredient_2	ingredient_3	ingredient_4	ingr
1	"Blue Dog" Zombie	1	Plantation Overproof Rum:12.5	Goslings Bermudan Dark Rum:12.5	Chairman's Reserve Spiced Rum:12.5	Chairmans Reserve White Rum:12.5	Gab
2	20th Century	2	Gin:37.5	Lillet Blanc:20.0	Crème de Cacao:20.0	Fresh Lemon Juice:20.0	0
3	50 - 50	3	Gin:50.0	Dry Vermouth:50.0	1 Olive	0	0
4	Abaci Batida	4	Cachaca:50.0	Pineapple Juice:75.0	Lemon Juice:12.5	Sugar Syrup:5.0	0
5	Abbey Cocktail	5	Gin:50.0	Orange Juice:37.5	Orange Bitters:2.0(dash)	Maraschino Cherry	0
6	Adios M.F	6	Vodka:12.5	Tequila:12.5	Gin:12.5	Rum:12.5	Blue
7	Admiral Benbow	7	Plymouth Gin:50.0	Dry Vermouth:25.0	Lime Juice:12.5	Maraschino Cherry	0
8	Admiral Perry	8	Pear Vodka:50.0	Cinnamon Schnapps:25.0	Dry Vermouth:25.0	White Crème de Cacao:5.0	Pear
9	Adonis	9	Dry Sherry:25.0	Sweet Vermouth:12.5	Dry Vermouth:12.5	Orange Bitters:2.0(dash)	0
10	Adult Hot Chocolate	10	Peppermint Schnapps:50.0	Hot Chocolate	Whipped Cream	Chocolate Shavings	0

(5) 分類（R）

1. 從資料清理完搭建出來的資料庫，利用dplyr與正則表達式的配合，根據每款調酒中所使用到的材料，分成酸（檸檬/酸橙）、甜（糖漿/砂糖/蜂蜜/巧克力）、苦（苦精）和薄荷四個data frame並標記好；
2. 根據調酒的容量，把120ml以下的分為短飲（快快喝，快快醉），120ml以上的分為長飲（慢慢喝，還是會醉）；
3. 資料庫中的調酒的酒精濃度跨度為0-56，按無酒精和每20度劃分，分成四個層級如下：
    - （0度無酒精）偷偷告訴你，我不含酒精呦～
    - （1～20度）啊啊啊啊明天還有早八Orz
    - （21～40度）喝到微醺最快樂！
    - （41～56度）人生偶爾就是要來大醉一場

最後把它們都合併起來：

cocktails_name <chr>	abv <dbl>	volume <dbl>	classified_by_abv <chr>	long_and_short_drink <chr>	sour <dbl>	sweet <dbl>	bitter <dbl>	mint <dbl>
"Blue Dog" Zombie	21.4	125	喝到微醺最快樂！	慢慢喝，還是會醉	1	1	0	0
20th Century	23.6	98	喝到微醺最快樂！	快快喝，快快醉	1	0	0	0
50 - 50	39.6	100	喝到微醺最快樂！	快快喝，快快醉	0	0	0	0
Abaci Batida	16.7	142	啊啊啊啊明天還有早八Orz	慢慢喝，還是會醉	1	1	0	0
Abbey Cocktail	22.3	89	喝到微醺最快樂！	快快喝，快快醉	0	0	1	0
Adios M.F	39.4	62	喝到微醺最快樂！	快快喝，快快醉	0	0	0	0
Admiral Benbow	34.6	88	喝到微醺最快樂！	快快喝，快快醉	1	0	0	0
Admiral Perry	39.7	105	喝到微醺最快樂！	快快喝，快快醉	0	0	0	0
Adonis	22.6	52	喝到微醺最快樂！	快快喝，快快醉	0	0	1	0
Adult Hot Chocolate	49.0	50	人生偶爾就是要來大醉一場	快快喝，快快醉	0	1	0	0

1-10 of 908 rows　Previous 1 2 3 4 5 6 ... 91 Next

[4] 我們有將那些疏漏記錄下來，供未來擴通資料庫之用。錯誤如：有成分名成卻沒有寫容量，如：只寫 hot water 卻沒有寫要幾 ml，或是成分名為空白。另外，還有一種型態被我們刪除了，這種形態的寫法是Top-with[成分名稱] 或是 [成分名稱]to-fill，要依據酒杯容量減去杯內已有成分之容量來計算，現階段我們還不處理，未來優化時將一並解決，詳情請見「未來展望」。

[5] 目前共刪除153列（種調酒），總調酒數量從1061減至908種。

(6) 隨機推薦

利用sample和data frame的資料結構從分類好的資料庫中，抽出五款存放到ramdom的data frame裡，每跑一次就會抽出新的五款酒

(7) 視覺化 (R shiny)

利用shiny建立使用者互動介面，先用navbarPage把介面可以分成三個頁面供使用者選擇，並設置好介面呈現的樣式（如背景顏色、字體大小等）。

a. 頁面一（我要按喜好選酒！）

將版面劃分成上下兩塊，上方設置了兩個下拉選單，連結到調酒資料庫中的濃度與容量，下方為output的調酒表格，會根據下拉選單選擇的篩選條件即時顯示出對應的調酒有哪些。

b.頁面二（我要按味道選酒！）

將版面劃分成左右兩塊，左方設置了欄位顯示選單，連結到調酒資料庫中的酸、甜、苦、薄荷，右方為output的調酒表格，會根據欄位選單和調酒表格上方的數值選項所選擇的的篩選條件即時顯示出對應的調酒有哪些。

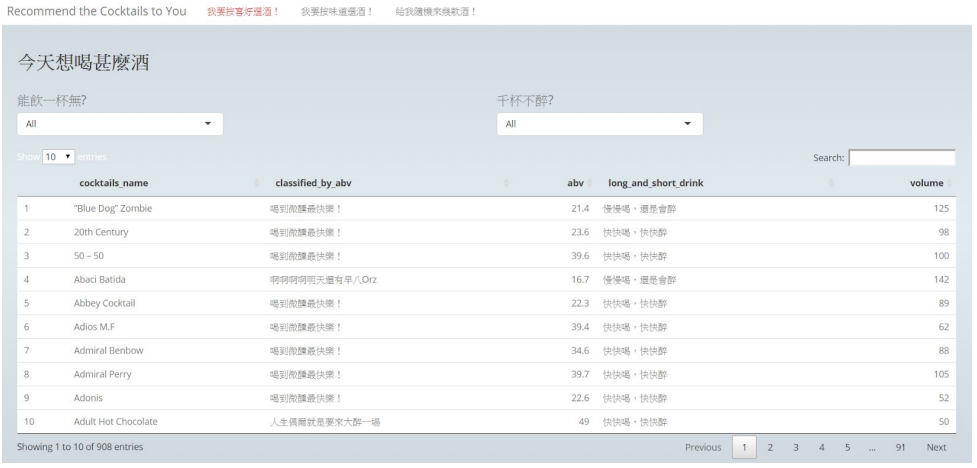
c.頁面三（給我隨機來幾款酒！）

output表格連結到ramdom的data frame，顯示出程式隨機抽出的五款調酒的詳細資料。

3.結果

透過shiny 讓資料視覺化，讓使用者可以透過介面操作，依據喜好和口味選擇符合心目中條件的雞尾酒，以下分三個頁面進行操作說明。

(1)頁面一（我要按喜好選酒！）

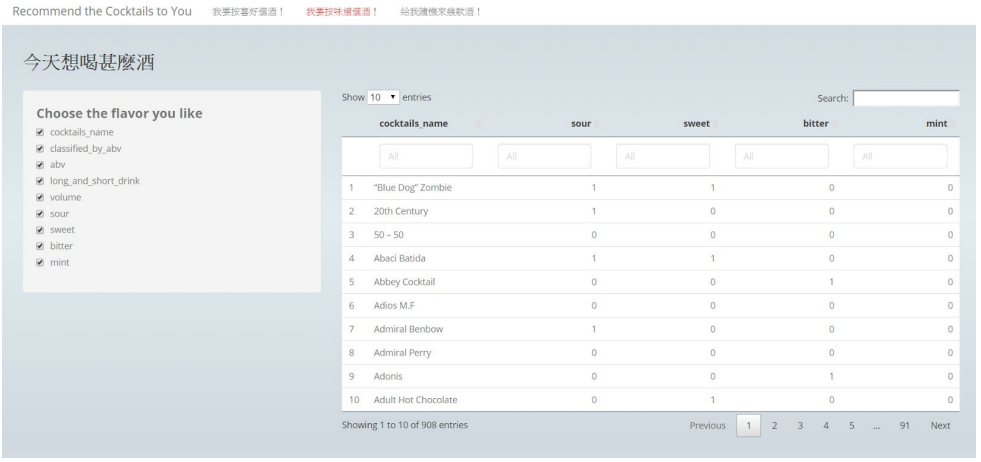


a. 此頁面讓使用者可依據喜好進行選擇，上面有兩個下拉選項，分別為「能飲一杯無？」(依據酒精濃度區分)和「千杯不醉？」(依據120ml為界區分長短飲)，詳細分類在2-2(5)有進行說明，預設為All，也就是所有種類的雞尾酒連同abv和volume都呈現在底下的datatable內。

b. 挑選好喜好後，底下的datatable即會依據條件進行篩選，呈現符合使用者需求的雞尾酒。

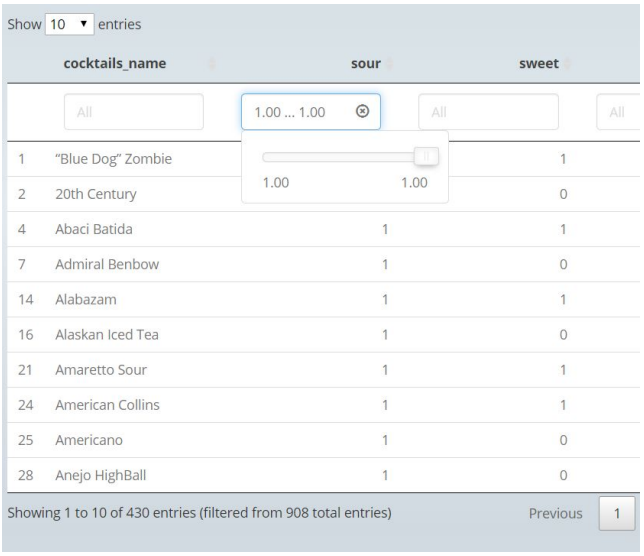
c. 可再透過search框綜合兩種條件進行名稱的篩選。

(2)頁面二（我要按味道選酒！）



a. 此頁面讓使用者可依據口味進行選擇，同樣的資料庫，依據酸甜苦薄荷將雞尾酒分類。

b. 依據datatable的filter功能，可以將口味進行1(有)、0(無)的篩選，讓使用者可以自由選擇，要酸要苦，或是不要酸也不要薄荷等。



c. 利用search框，可以搜尋想要的雞尾酒的口味。

d. 點選口味的欄位，會有灰底框出該欄位，箭頭向下依據大至小排列，也就1~0，箭頭向上，由小至大排列，也就是0~1。

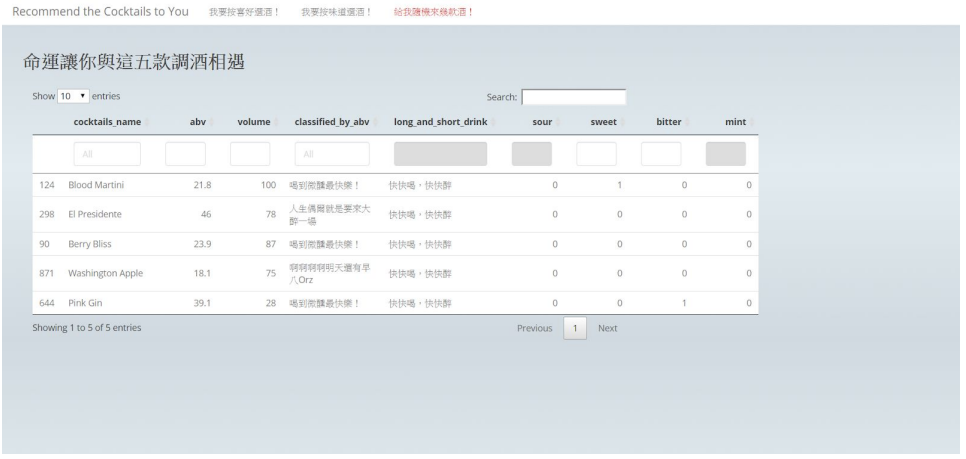
	cocktails_name	sour ▼	sweet
	All	All	All
1	"Blue Dog" Zombie	1	1
2	20th Century	1	0
4	Abaci Batida	1	1
7	Admiral Benbow	1	0
14	Alabazam	1	1
16	Alaskan Iced Tea	1	0
21	Amaretto Sour	1	1
24	American Collins	1	1
25	Americano	1	0
28	Anejo HighBall	1	0

Showing 1 to 10 of 908 entries Previous

	cocktails_name	sour ▲	sweet
	All	All	All
3	50 – 50	0	0
5	Abbey Cocktail	0	0
6	Adios M.F	0	0
8	Admiral Perry	0	0
9	Adonis	0	0
10	Adult Hot Chocolate	0	1
11	Affinity	0	0
12	Agave Kiss	0	1
13	Agent Orange	0	0
15	Alaska Cocktail	0	0

Showing 1 to 10 of 908 entries Previous

(3)頁面三（給我隨機來幾款酒！）



a. 隨機推薦五種雞尾酒給使用者，並呈現所有分類資料，重新開啟shiny會再隨機新挑選另五種雞尾酒。

4.討論與貢獻

4.1困難

(1) 調酒網站資料不全

我們需要計算各種調酒的酒精濃度，但是酒網站的基酒列表並不齊全，也沒有提供各基酒（例如琴酒、伏特加等）的酒精濃度，所以我們必須先從各種調酒中整理出所有原料，再靠人工建立三百多種原料的酒精濃度資料，才能計算各種調酒的酒精濃度。

同樣的情況也發生在成分資料填寫上，網站建立時資料填寫的疏漏同樣使我們必須耗費時間挑去那些不完整的調酒資料。

(2) 情緒分析

原先的程式中，我們希望能讓使用者在 GUI 上一篇心情小文，程式分析出文章中的情緒，根據情緒的強弱來幫使用者決定酒精濃度的強弱[6]。當時決定要沿用一位同學在 python 課程中的所使用的情緒分析工具 SnowNLP，但是在實作時遭遇到了兩個問題：

a. 因為 SnowNLP 是寫給 python 的套件，因此要在 R 上運行，必須嘗試將其接至 R 上，轉換的過程很繁瑣而且還不確定有效（不一定學得會哈哈哈），所以那時就有在直接用 R 寫一個新的。
b. 在尋求 python 與 R 的轉換時，我們也順便更深入的研究了一下情緒文本分析，發現 SnowNLP 這方面其實做得有點粗糙[7]，於是‘我們直接決定改用quanteda 進行情緒分析。但其實在發現這個問題時，距離報告截止的時間已所剩不多，所以最後沒有達到原本先進行情緒分析再推薦調酒的目標，對教授與助教非常抱歉。這（做出情緒分析）將成為我們未來優化的第一個目標。

4.2未來展望

(1) 文本情緒分析優化[8]

(2) 篩選條件優化

目前的口味篩選條件，我們僅以我們認為該成分為甜的，如蜂蜜等，酸的，如檸檬等，條件進行區分，而實質是否如此，我們也無法保證，只能確定它是否有甜味、酸味、苦味、薄荷味的條件存在，其中因為我們對成分的不熟悉，可能還會遺漏了許多成分的判別，造成篩選不準確，且在程度的區分上目前也只有1(有)、0(無)的區分，在實際上很難發會作用。未來可能可以去找是否有成分說明的網站或是自行建立出口味程度對比的資料庫去換算，讓使用者可以有更深入的條件選項，可以做選擇的依據。

---

[6] 越接近兩極，越極端的情緒，就給酒精濃度較高的酒，反之則給出酒精濃度較低的酒種。
[7] 並沒有一個正負面的詞庫，而是把一個句子分出的所有詞都參與計算，包括各種各樣的無意義詞、符號等等，而事實上這些詞並不能反映任何情感。
[8] 敘述請見4.1.(2)

(3) 介面優化

目前的介面主要以datatable的表格方式呈現，對於使用者來說，不夠親近，因為直接看到數字，很難想像到底是多少，如酒精濃度、容量等，用文字難以顯示一般人喝多少會醉等等，未來可以加上動態圖片呈現，，比較各酒類之間的差別等，增加介面的互動性。

5.附錄

5.1 組員分工

	分工
蘇以恩	資料爬取、資料清理、情緒分析
吳京	資料清理、書面報告
許青純	shiny介面、PPT
羅俊欣	資料分類、shiny介面