Data Mining Hw1 P76074046 方鈞麒

環境:

系統: Ubuntu 18.04

語言: Python 2.7

執行方式:

Apriori: python hw1\_apriori.py 資料路徑 min\_support min\_confidence

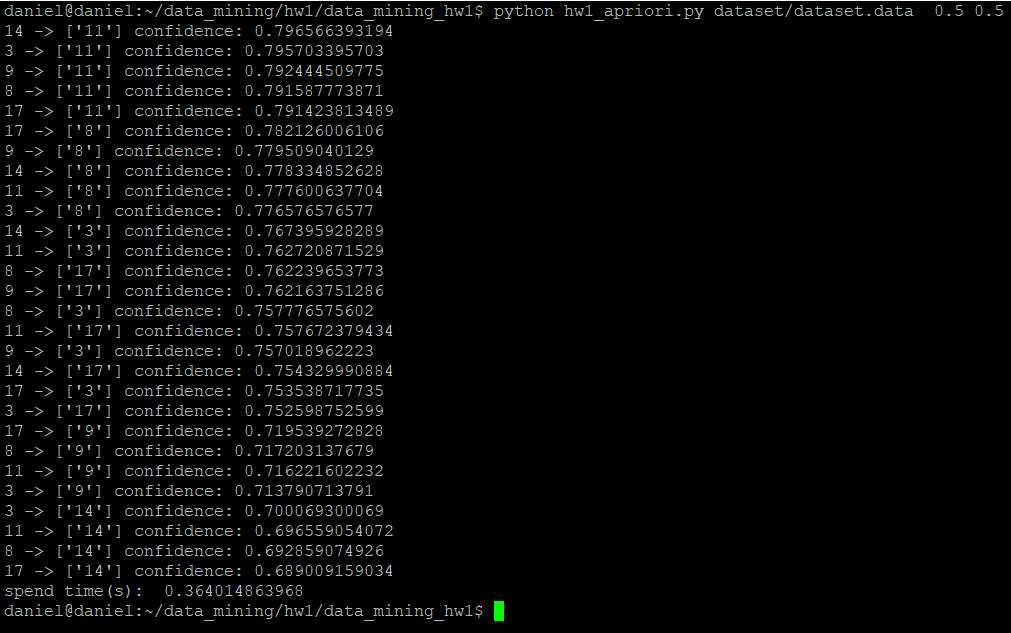
Fp-growth: python hw1\_fp\_growth.py 資料路徑 min\_support min\_confidence

資料檔案都放在dataset資料夾內，總共有五個dataset，物品和交易數量分別為

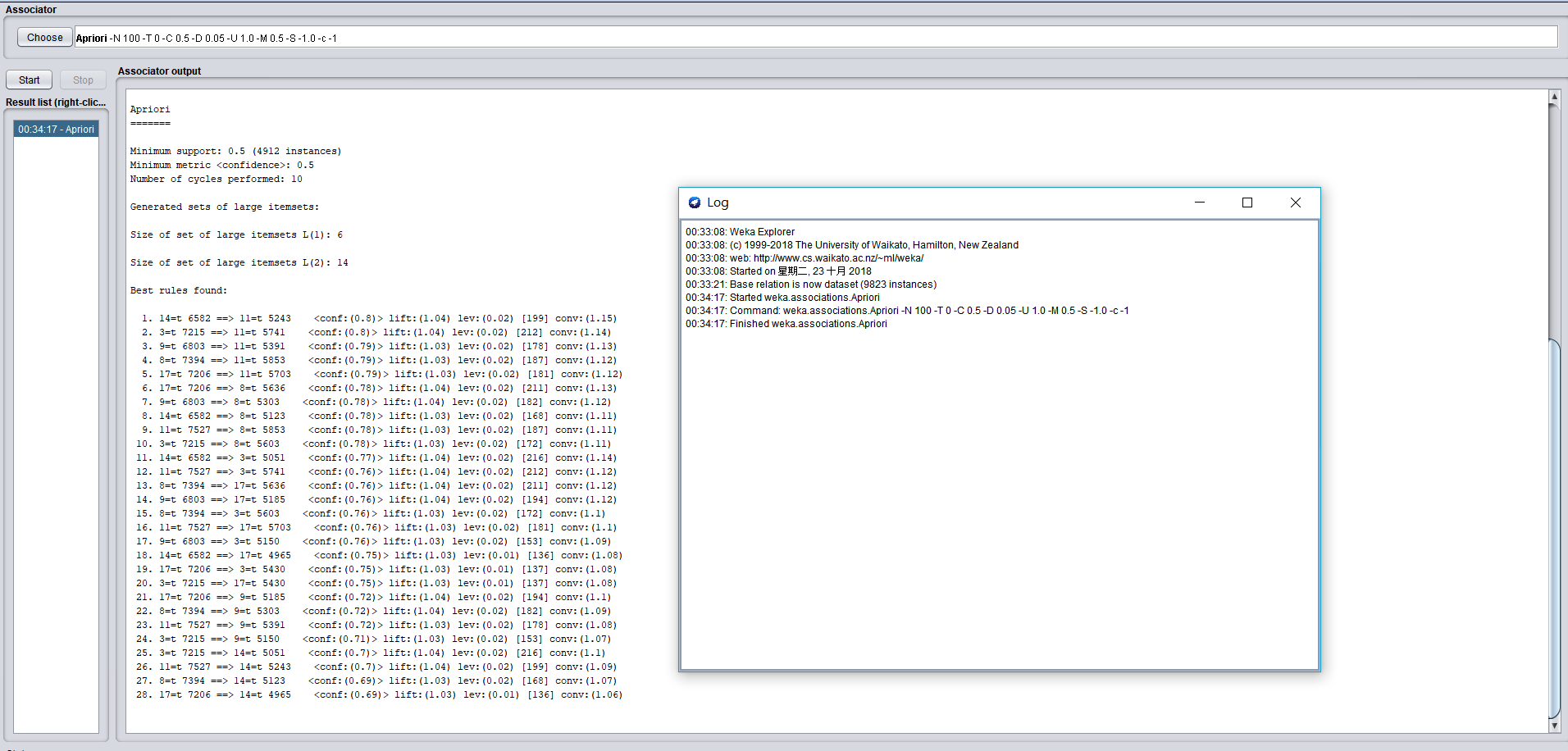
1. #Items: 20, #Transactions: 10000
2. #Items: 20, #Transactions: 50000
3. #Items: 20, #Transactions: 100000
4. #Items: 30, #Transactions: 10000
5. #Items: 40, #Transactions: 10000

執行程式時使用 .data 檔案，.pat檔案是data的資訊，而因為weka不吃.data檔案，所以將.data透過github上此專案https://github.com/mhwong2007/IBM-Quest-Data-Converter 轉成 .csv 檔案。

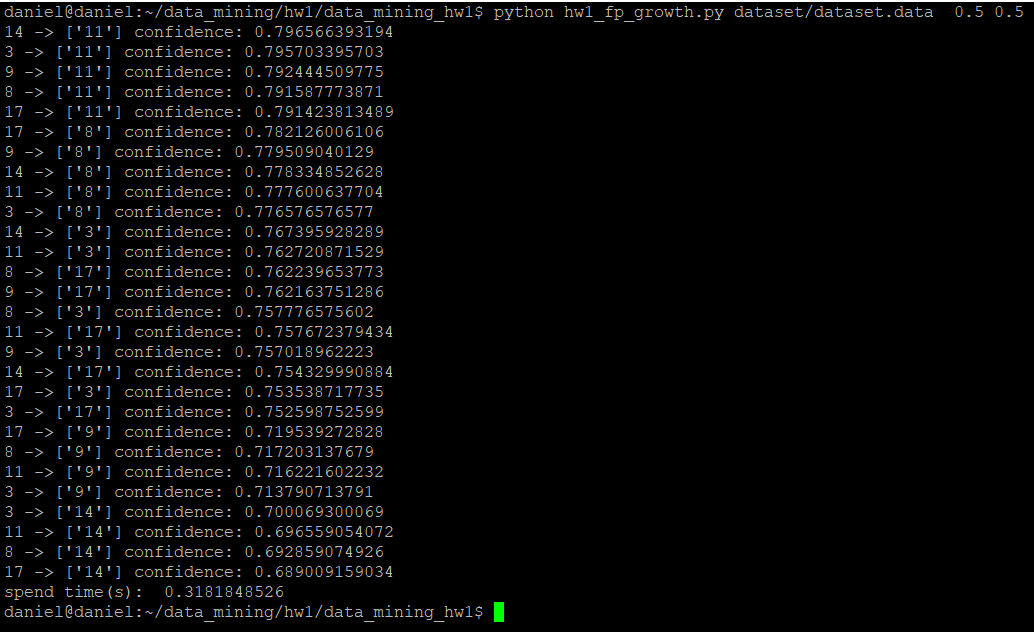
下面分別為自己和weka使用apriori & fp-growth的結果圖，min\_support和min\_confidence皆設為0.5



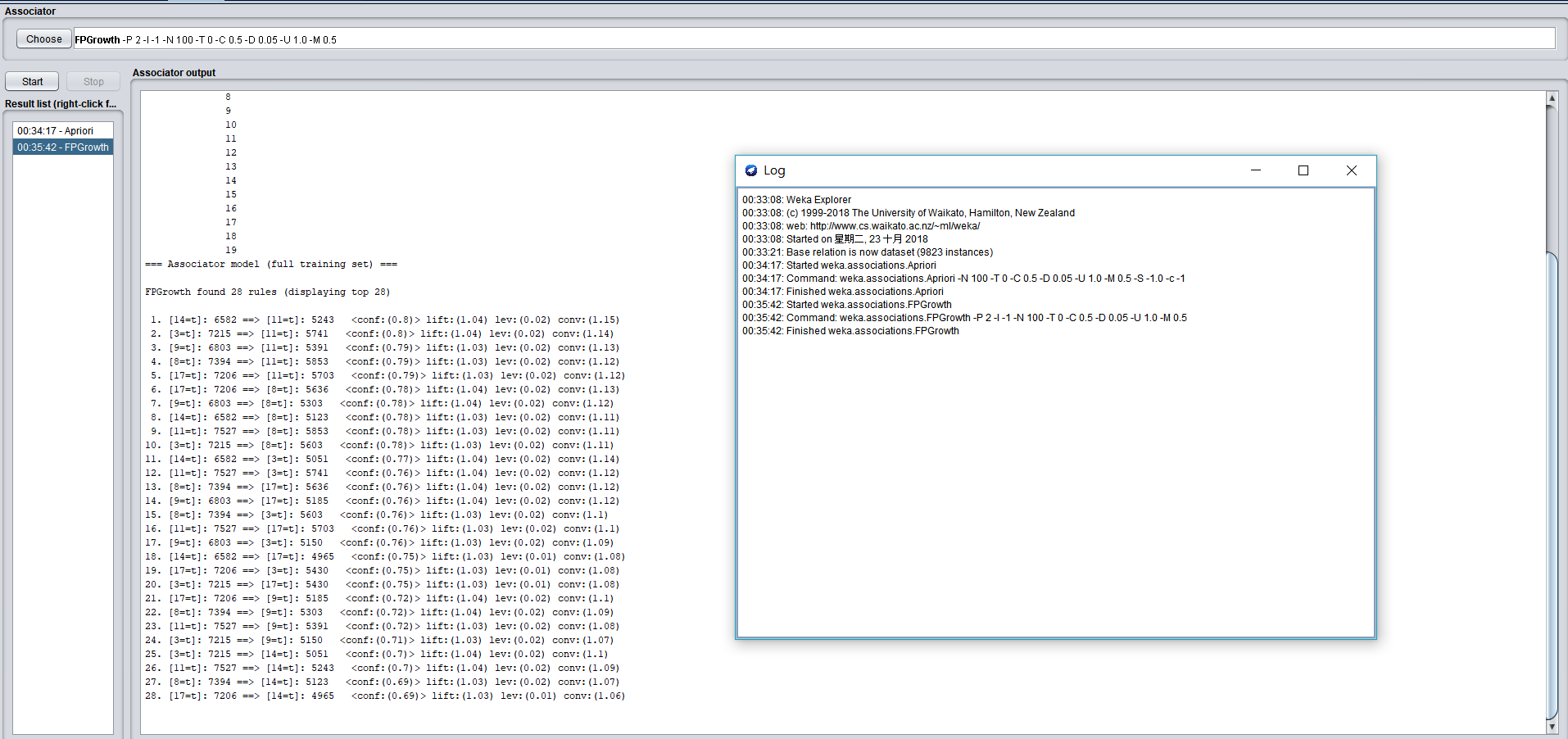
圖一、作業apriori結果與時間



圖二、weka使用apriori結果



圖三、作業fp-growth結果與時間



圖四、weka使用fp-growth結果

由於weka沒有輸出他計算的時間，只能從log去大概看過了幾秒，這五個資料集在weka都只需不到1秒即可算完。下表為自己實作程式的執行時間。從dataset1~3的變化可以看出物件數量和執行時間成倍數關係，而且當物件數量越多時Fp-growth減少的時間越明顯。而從dataset1、4、5，可以發現當交易數量變多但是物件數量不變時，Fp-growth就會花比Apriori算法還要多時間，猜測可能是因為物件多導致建樹時生成很多枝系，像是一條transaction就獨立一條根枝，導致在執行時間上沒有省比較多，而且還多出建樹的時間。

|  |  |  |
| --- | --- | --- |
|  | Apriori | Fp-growth |
| Dataset1 | 0.37 | 0.32 |
| Dataset2 | 1.61 | 1.3 |
| Dataset3 | 3.18 | 2.51 |
| Dataset4 | 0.24 | 0.37 |
| Dataset5 | 0.13 | 0.16 |

表五、作業使用兩種算法所需時間