**Features description**

1. Administrative: the number of Administrative type of pages visited by the visitor
2. Administrative\_Duration: total time spent in Administrative page categories
3. Informational: the number of Informational type of pages visited by the visitor
4. Informational\_Duration: total time spent in Informational page categories
5. ProductRelated: the number of ProductRelated type of pages visited by the visitor
6. ProductRelated\_Duration: total time spent in ProductRelatede page categories
7. BounceRates: the percentage of visitors who enter the site from that page and then leave (bounce) without triggering any other requests to the analytics server
8. ExitRates: the percentage that were the last in the session
9. PageValues: the average value for a web page that a user visited before completing an e-commerce transaction
10. SpecialDay: the closeness of the site visiting time to a specific special day in which the sessions are more likely to be finalized with transaction
11. Month: month of the year
12. OperatingSystems: which Operating ystem
13. Browser: which Browser
14. Region: which Region
15. TrafficType: which Traffic Type
16. VisitorType: which Visitor Type
17. Weekend: the date of the visit is weekend
18. Revenue: be used as the class label(True:1, False:0)

**Preprocessing**

1. 在Preprocessing.ipynb裡

2. 將 'Month' 的資料換為數字

3. 將 'VisitorTyp' 的資料換為數字

'Returning\_Visitor' --> 0

'New\_Visitor' --> 1

'Other' --> 2

4. 將 'Weekend' 的資料換為數字

False --> 0

True --> 1

5. 將 'Revenue' 的資料換為數字

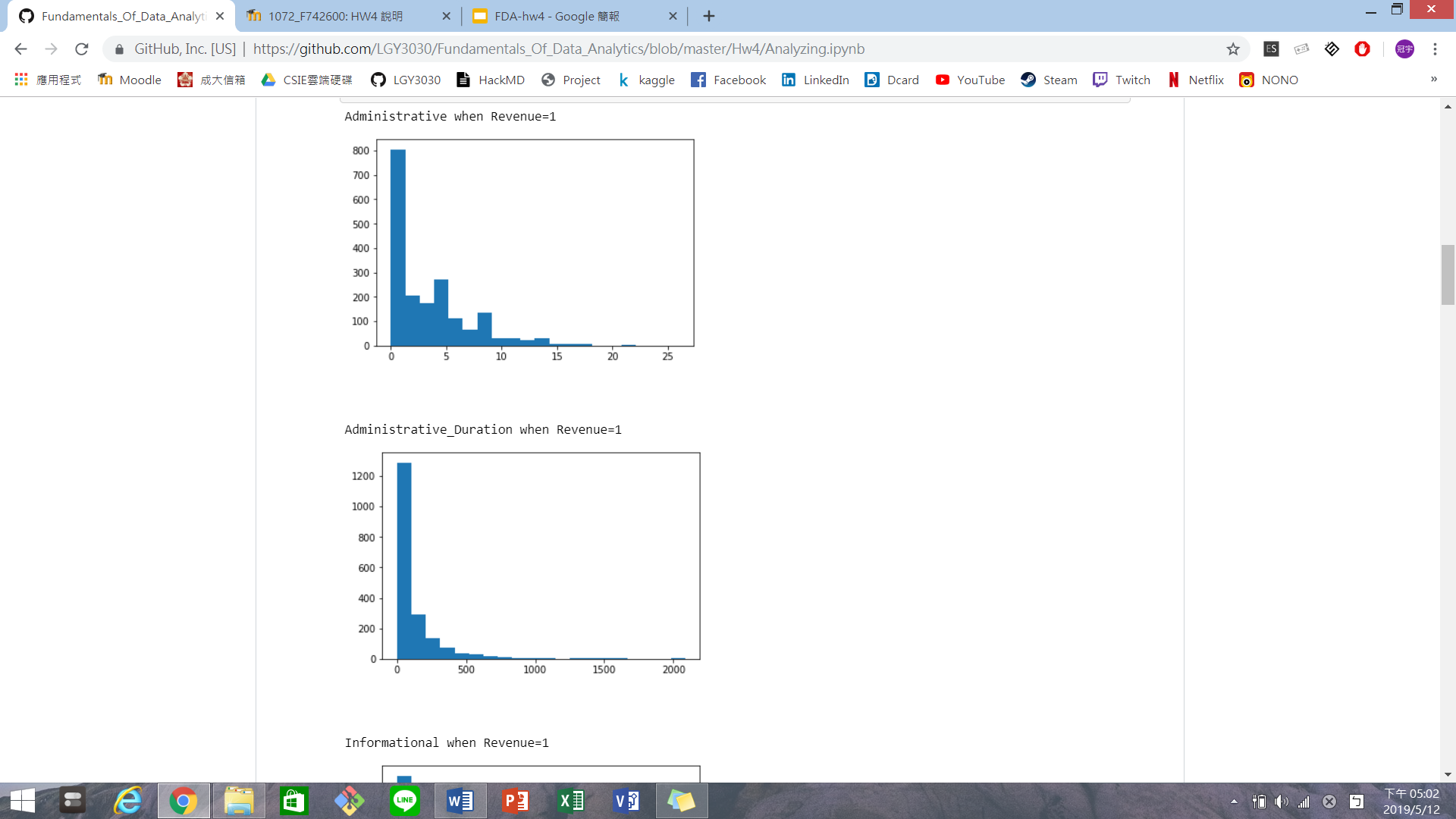
False --> 0

True --> 1

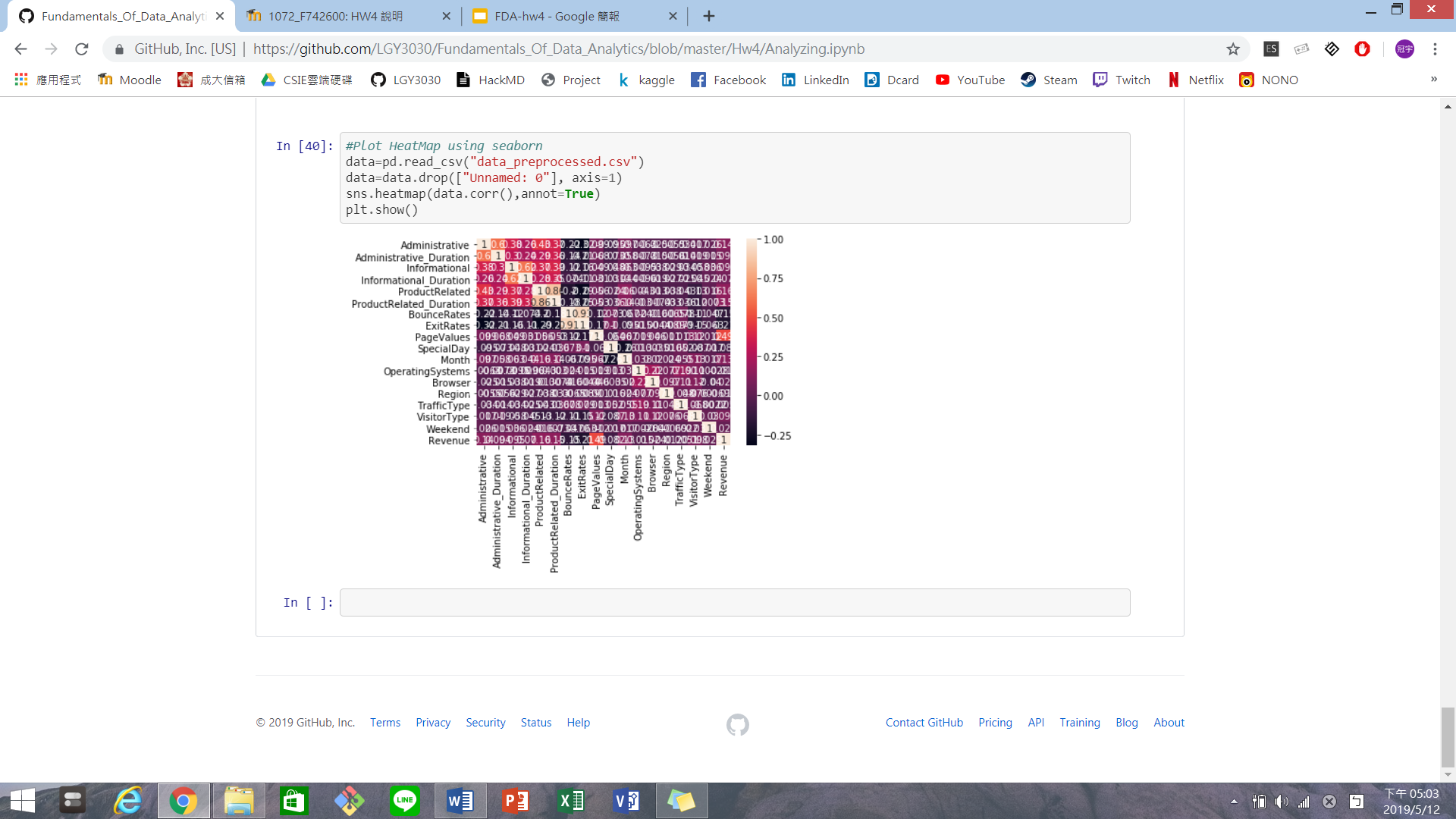
6. 將資料儲存為data\_preprocessed.csv

**Analyzing**

1. 在Analyzing.ipynb裡

2. 取得Revenue為True=1的資料, 觀察各features和Revenue的關係

3. Plot HeatMap using seaborn

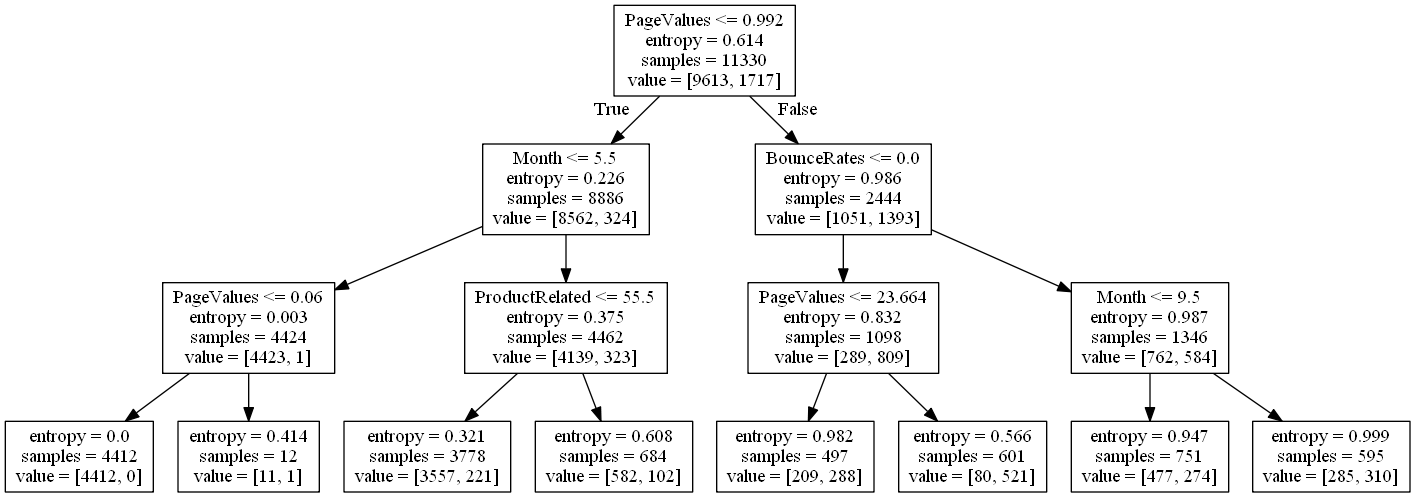


**Define a problem**

1. 將問題定義為分類問題
2. 將 'Revenue' 的資料 (False --> 0, True --> 1) 作為label
3. 其他資料作為輸入的features

**Decision Tree Classifier**

1. 在Decision\_Tree\_Classifier.ipynb裡
2. 讀取data\_preprocessed.csv
3. 將資料分為train\_x,train\_y,test\_x,test\_y
4. 建立Decision Tree分類器
5. 預測測試資料
6. 準確率為84.39%
7. Decision Tree圖



1. ['Administrative', 'Administrative\_Duration', 'Informational', 'Informational\_Duration', 'ProductRelated', 'ProductRelated\_Duration', 'BounceRates', 'ExitRates', 'PageValues', 'SpecialDay', 'Month', 'OperatingSystems', 'Browser', 'Region', 'TrafficType', 'VisitorType', 'Weekend']分別的feature importance為[0, 0, 0, 0, 0.01324909,0, 0.0525466 , 0, 0.82597418, 0.,0.10823013, 0, 0, 0, 0, 0, 0]
2. PageValues的影響力最大,Month第二大

**Neural network**

1. 在Neural\_Network.ipynb
2. 讀取data\_preprocessed.csv
3. 將資料分為train\_x,train\_y,test\_x,test\_y
4. create the model
5. add layers
6. show the summary
7. compile the model
8. fit the model
9. 準確率為84.5%

**Improvement 1**

1. 在Improvement.ipynb裡
2. Decision\_Tree\_Classifier的準確率為84.39% , 而Neural\_Network的準確率為84.5%
3. 此外, 在Decision\_Tree\_Classifier的部分可以看出PageValues的影響力最大,Month第二大
4. 所以我將資料的PageValues都乘上5, Month都乘上2,希望以此增加他們的比重,因為每項資料的重要性都不一樣
5. 將乘上數字的新資料丟入Neural\_Network的model裡
6. 準確率提升為84.8%

**Improvement 2**

1. 在Improvement.ipynb裡
2. 以Improvement 1的改變為底,再增加一個Dense層,以及改變學習率(0.001🡪0.0005)
3. 準確率提升為85.1%