P76074282林品磊 Project2\_report

1. Dataset：
   * 輸出.csv：  
     我用爬蟲爬全台灣已上市公司的基本資料

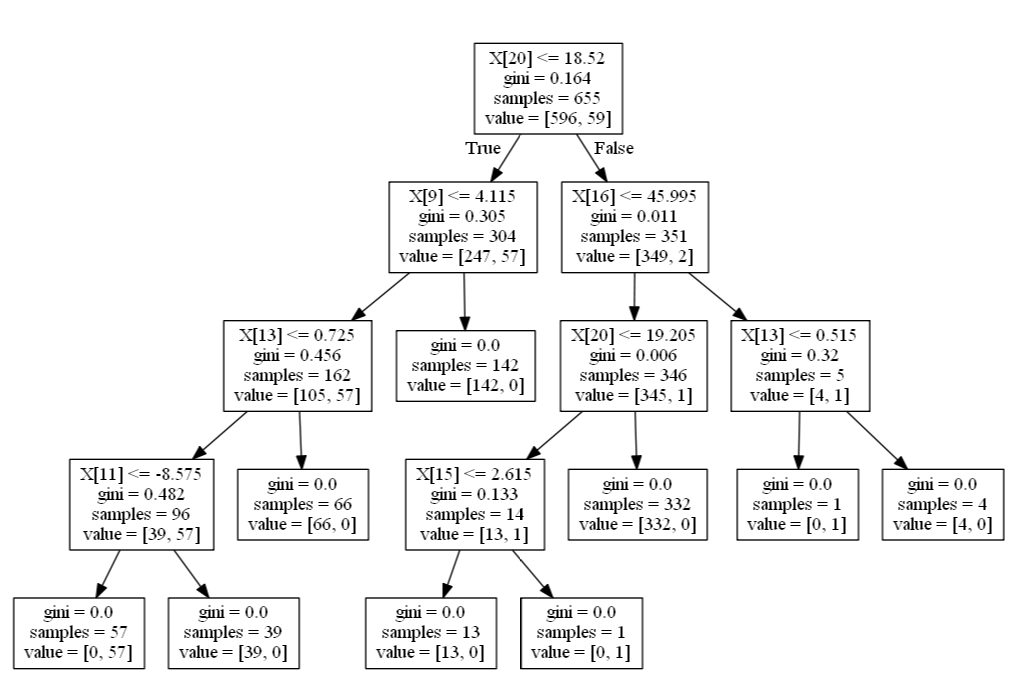
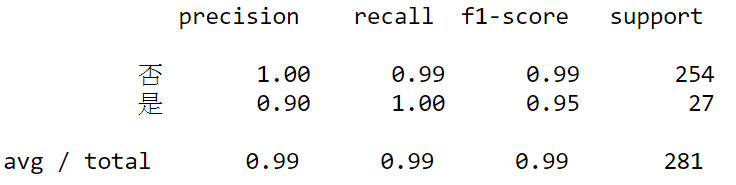
Ex. http://jsjustweb.jihsun.com.tw/z/zc/zca/zca\_1103.djhtm

* + Attribute：

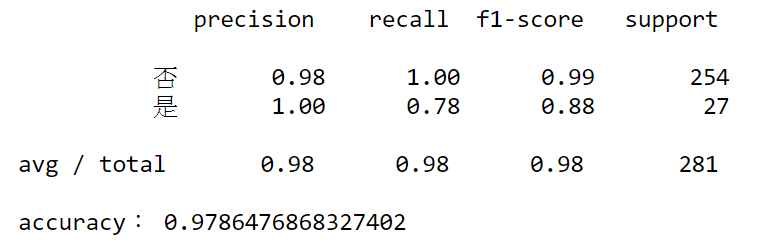
1. 公司代碼
2. 開盤價
3. 最高價
4. 最低價
5. 收盤價
6. 漲跌
7. 本益比
8. 成交量
9. 同業平均本益比
10. 殖利率
11. 總市值
12. 今年以來(投資報酬率)
13. 每股淨值(元)
14. 貝他值
15. 每人營收(仟元)
16. 標準差
17. 每股營收(元)
18. 負債比例
19. 股價淨值比
20. 股本(億, 台幣)
21. 營業毛利率
22. 現金股利(元)
23. 營業利益率
24. 稅前淨利率
25. 資產報酬率
26. 股東權益報酬率
    * Rule：
      1. 本益比<平均
      2. 貝他值<平均
      3. 殖利率>平均
      4. 今年以來(投資報酬率)>平均
      5. 營業毛利率>平均

🡺符合上述五點：可投資。否則，不可投資

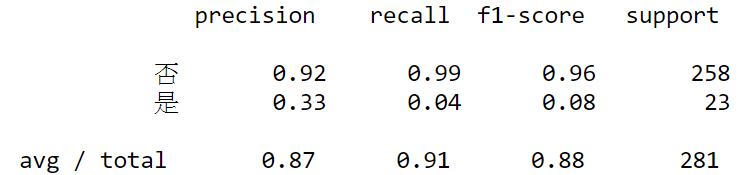
1. 結果：
   * Decision tree：  
     p.s因為中文顯示會變亂碼，所以要去對照data的欄位了><



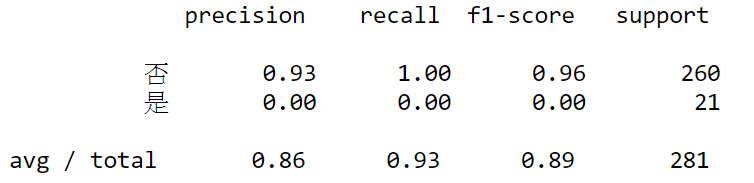
* + Random forest：



* + kNN：



* + SVM：



1. 心得：  
   現在要使用分類的model真的很方便，很多套件都已經寫好了  
   沒想到原來decision tree跟 random forest的準確度竟然如此高  
   因為不想只是為了寫作業而寫，所以我就自己爬了真實的資料  
   之後還可以真的應用在選股上☺
2. Comparison：

經過實驗後發現，decision tree跟random forest是這四種model裡最準的兩個，準確率都在0.95以上，有可能是因為當初在決定可投資股票的時候的假設比較強硬，只要符合假設就百分之百是可投資。其實random forest只是多個decision tree組合而成的，有點像是三個臭皮匠勝過一個諸葛亮個感覺。  
kNN跟SVM的準確率就差了一點，不過也算不錯了。這兩個模型也很相像，都是在多為的空間上用一個平面來區分兩類的資料。不過準確率差了一點我猜可能是因為有太多多餘的feature以致維度很高影響到無法切出很好的區分。

1. Discussion：

我覺得每個model都有每個model適合應用的地方。只不過剛好這題用decision tree這類的model來分類最準確。在寫作業的時候發現，現在好像也很流行不用一個超強的model來解決問題，因為怕如果訓練的方向錯誤的話跑出來的結果也就掛了，反而是用Ensemble learning：結合多個弱模型來找出一個最好的結果，有點像眾人投票的感覺。用這種學習法跑出來的準確率有時候還比一個超強模型來的高，蠻神奇的。

這次作業也讓我對分類模型更認識，也知道怎麼去使用現有的模型function，呼叫起來真的很方便。很好用！