Data Mining Project 2

資工碩一P76084300 施逢怡

# 一、簡介

1. 專案目標:分析學生的各種資料來分類學生(好學生、表現沒這麼好的學生)

2. 資料: Student Performance Data Set

(<https://archive.ics.uci.edu/ml/datasets/student%2Bperformance>)

有將attributes做篩選，最後選擇20個attributes(student\_all.csv)

3. 方法: **Decision tree、Random Forest**

# 二、Design a set of rules to classify data

我認為”好學生”應該要有三種特質，分別是

1:每天溫習功課的時間至少要有2小時

2.不可以有被當的科目

3.成績要中上以上

Dataset裡面相對應的Attribute就是”studytime”, “failures”, “grade”

* studytime裡面的數值就是讀書時間(**1** - <2 hours, **2** - 2 to 5 hours, **3** - 5 to 10 hours, or **4** - >10 hours)
* failures則是被當科目的數目，n if 1<=n<3, else 4
* grade則是分為”good”和”bad”，是以他們的成績最分類

所以最後的三個基本**Rules(後面有再增加Feature相對應的Rule)**:

1. **studytime >= 2**
2. **failures == 0**
3. **grade == ”good”**

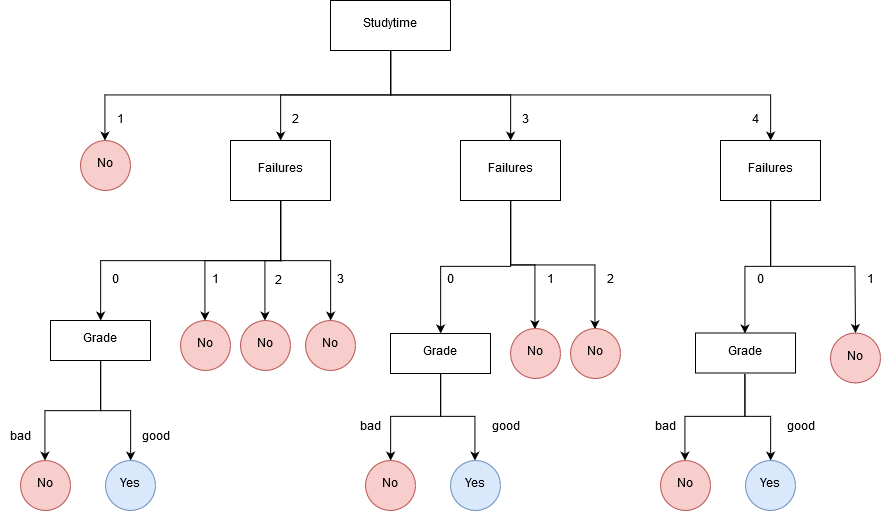
# 結果

* 我自己實作了Decision tree(decisiontree.py)，並得到圖一的結果，圖二和圖三則是用

DecisionTreeClassifier(Scikit-learn\_decisiontree.ipynb)跑出來的結果。

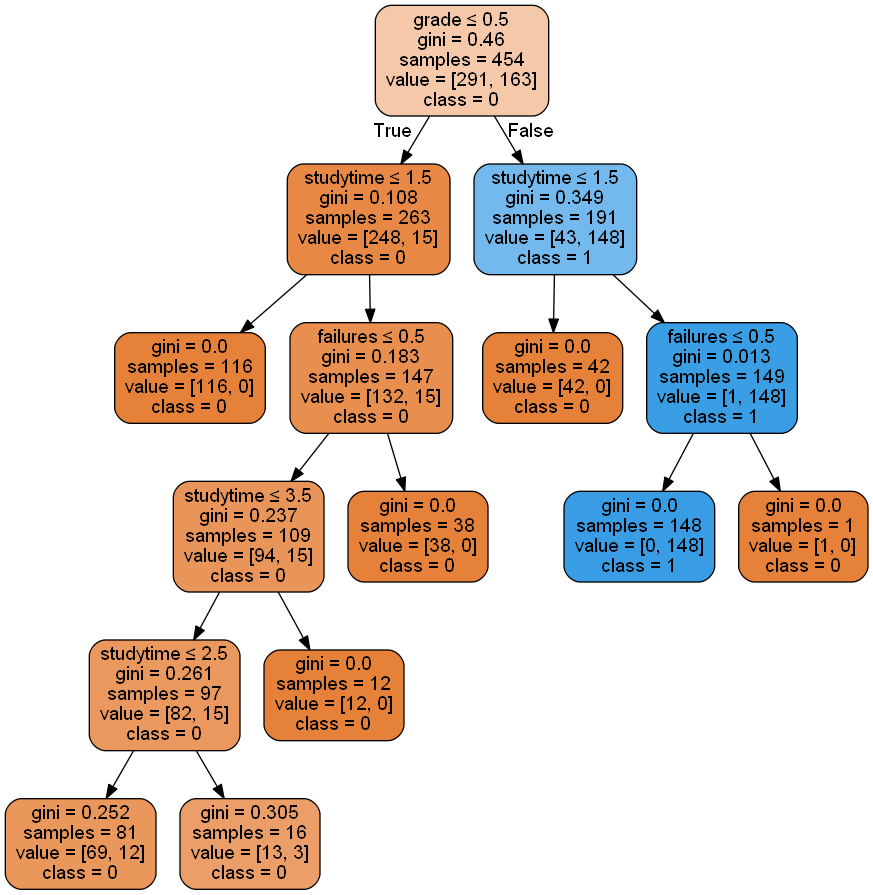
* 圖一和圖二為3個features的結果
* 圖三是10個features的Decision Tree的結果，增加了freetime(課後自由時間)、goout(出門和朋友玩的頻率)、Medu(母親教育程度)、Fedu(父親教育程度)、Dalc(平日喝酒頻率)、health(健康程度)，各個結果如下圖:

**圖一**



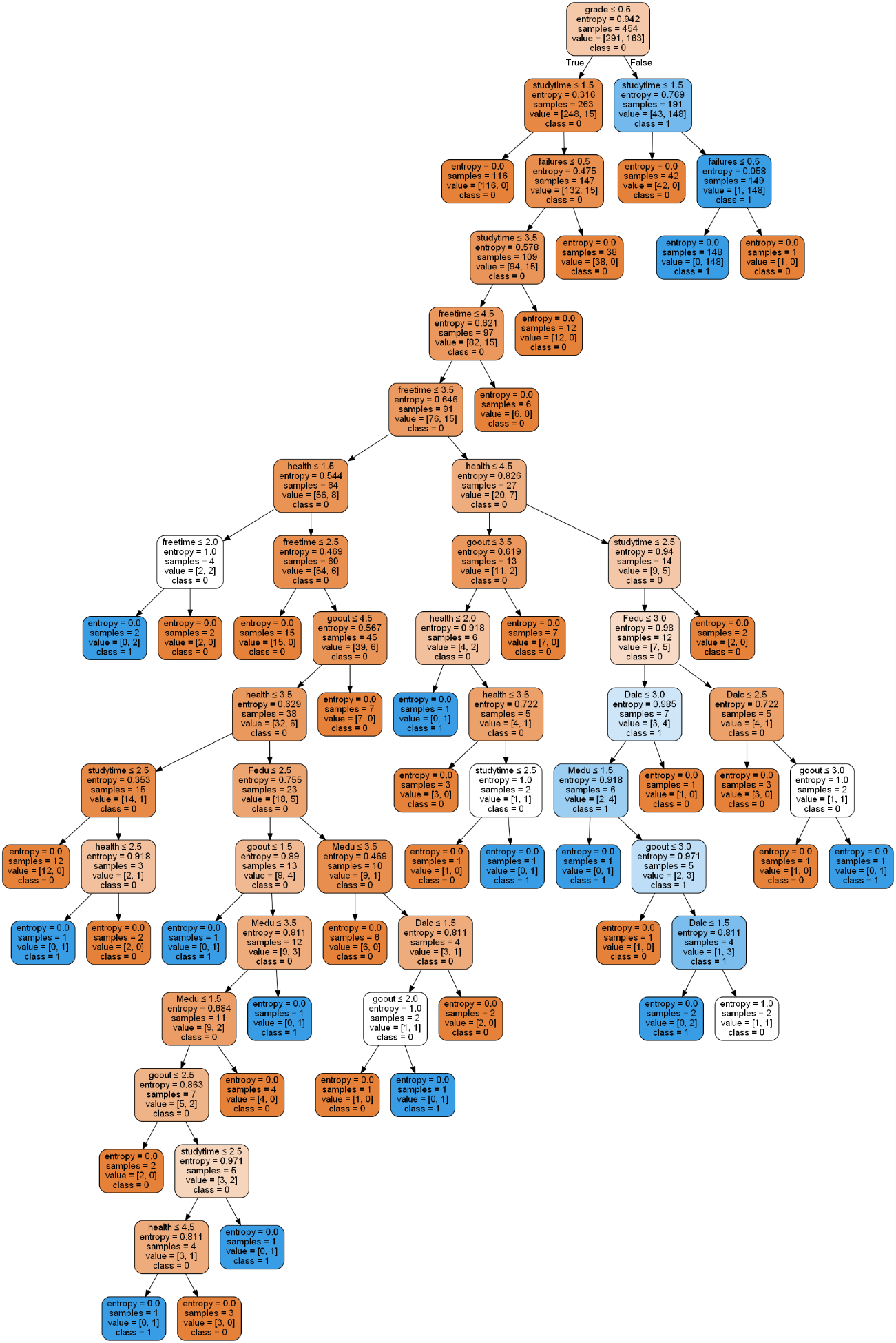
**圖二**

**(Accuracy:0.958974358974359)**



**圖三**

**(Accuracy: 0.9076923076923077)**



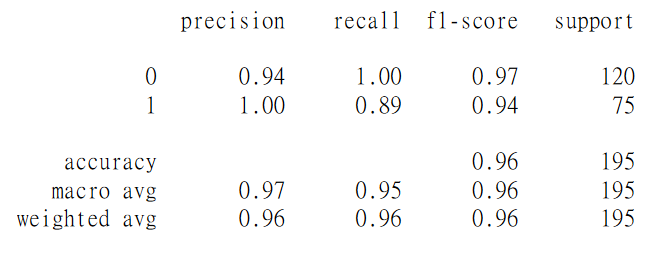
# Random Forest:

隨機森林裡面的每棵樹的產生的過程中，都已經考慮了避免共線性，避免過擬合，剩下的每棵樹需要做的就是盡可能的在自己所對應的數據(特徵)集情況下盡可能的做到最好的預測結果。

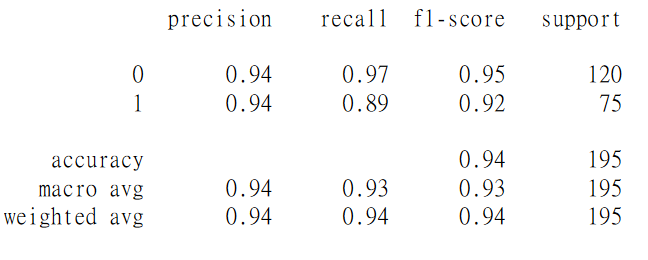
用隨機森林對一個新的對象進行分類判別時，隨機森林中的每一棵樹都會給出自己的分類選擇，並由此進行「投票」，森林整體的輸出結果將會是票數最多的分類選項；而在回歸問題中，隨機森林的輸出將會是所有決策樹輸出的平均值。

# 結果

**Feature**: studytime, failures, grade



**Feature**：studytime, failures, grade, goout, Medu, Fedu, freetime, Dalc, health



# 專案結論

這次的專案主要是Decision Tree的實作，我有自己實作(檔案:decisiontree.py)，裡面用entropy來做分類依據，在結果可以發現出來的結果並沒有和我一開始的Rule相衝突。

但不一樣的點是:我是依照平行的方式來Label我的資料，Decision tree出來的結果是以Tree的方式做分類，所以會有階層的先後順序。順序是Studytime -> Failure -> Grade。

除此之外，可以發現在讀書時間3和4中，被當科目只有相對應的(0,1,2)、(0,1)，代表在Training data裡面，讀書時間越久的學生，並不會被當太多的科目。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 讀書時間(studytime) | 1 | Bad student | | | | |
|  | 2 | 被當科目(Failure) | 0 | 成績 | 好 | Good student |
|  | |  | | 壞 | Bad student |
| 1 | Bad student | | |
| 2 | Bad student | | |
| 3 | Bad student | | |
| **3** | 被當科目(Failure) | 0 | 成績 | 好 | Good student |
|  | |  | | 壞 | Bad student |
| 1 | Bad student | | |
| 2 | Bad student | | |
| **4** | 被當科目(Failure) | 0 | 成績 | 好 | Good student |
|  | |  | | 壞 | Bad student |
| 1 | Bad student | | |

用DecisionTreeClassifier得出的結果，因為它將各個feature視為數值，所以切得更細，像是studytime就被切為1.5、2.5、3.5三個區段，和我原先的Rule的數值就不太一樣(比較圖如下)。並且也會導致tree較為deep。

|  |  |  |
| --- | --- | --- |
|  | 原本Rule | Classifier |
| Studytime | >=2 | >=1.5 |
| Failure | ==0 | <=0.5 |
| Grade | ==1(good) | >=0.5 |