

## Attributes setting

```
# Define the attributes
max_depth = 4
depth = 0
min_samples_split = 35

n_trees = 551
n_features = 8
sample_size = 0.1
n_samples = 500
```

Number of Trees: 551

當樹的數量成長到一定的值的時候，浮蕩的結果會趨於穩定，故我從原本幾十棵樹調整成目前 551 棵並 train 了好幾個小時得到最終結果

Number of Features: 8

根據樹的最深深度 4，取了兩倍的數當做 feature

Number of instance: 500

因為 tree 數量很多，故 instance 取小一點讓 random 的多樣性更多

## Difficulty :

- Panda 函式庫與 DataFrame 的不熟悉：
- 對 Decision Tree 的觀念不清晰
- Hyperparameters 的取值
- Training 過程蠻長且不能保證結果好壞

## Summarize :

一開始在實作 basic 的時候，因為此次作業多使用了 panda 函式庫的 DataFrame，導致在操作上遇到了不小的困難，它的使用方式與 numpy 沒有那麼相似，所以在查找文件和實際測試裡來回反復上花了不少時間，但在實作的過程中也逐漸掌握了該如何使用。

隨後在 basic part 裡，雖然有 template 的指示該往哪個方向去做，但初次接觸仍有很多觀念和想法不清晰，在這部分也是搭配著講義和網上資料將自己不足的知識都一一補全，再將這些想法逐個實現在 code 裡

就這麼跌跌撞撞總算完成了所有的 baseline，而後就是漫無止境的 training 過程，因為使用 random forest，每次 train 出來的結果有好有壞，隨後再調整參數再做 training 使得花費了很多時間。在多次嘗試不同的 hyperparameters 之後，我發現當 tree 的數目成長到一定數量的時候，其 train 出來的結果會逐漸穩定，而 tree 的深度和所使用的 feature 有需要有一定的關係數量才可以達到比較好的結果。

完成了此次 decision tree 的作業，基本的 machine learning 的觀念想法已經大致掌握，接下來便逐漸要踏入 neural network 的知識領域，難度會逐漸提升，但我相信我會做好準備的。