機器學習 ID3 以及 Random forest

一、ID3 流程

1. 首先先將資料由 txt 檔切成 float 放入 class flower 的 sepal 和 petal 長寬，並將花名放入 class flower 的 string，將資料放進一個 class 是因為較好排序，一筆資料都被綁在一起，sort 時不會亂掉。

2. 第二部就可以開始建立樹，首先先將剛剛讀入的 data 洗亂並切成 training data 跟 test data，然後用 training data 去建立樹。首先先將 training data 放在 root 並尋找 information gain 最高的切點，也就是 entrophy 最低的切點，在尋找時要先依照 column 做排序，數值由小到大，並由小掃到大觀察花的品種是否有改變，如果有則可能是切點，所以要計算其 information gain。

3. 找到切點後，因為資料為連續，因此將切點兩側的數值相加除 2 作為判斷切在左子樹或是右子樹的標準，並開始依照 column 跟判斷值將資料分別分到左邊以及右邊，然後繼續算 information gain 最高的繼續切，然後一直遞迴下去。直到 node 中的分類為一樣，或是 node 中資料量少於一定數目，則由該 node 中較多的作為分類(label)。

4. 之後就開始做 test 並且計算 accuracy、precision 及 recall，首先先將資料一筆一筆從 root 開始根據 node 的切值做判斷，資料沿著 node 走到 leaf 時，則以 leaf 的分類作為推測，猜測花的種類，並跟資料的 flower_name 做核對，並統計資料，看是否精準。

5. 我最後實做的是 k-fold，一顆樹做好後，將原本 data 資料洗亂並切成五份，每一次都使用不同的那一份當作 test data，剩下其他的當作 training data。以此來驗證較公平的精確度。

二、Random forest

1. random forest 建樹的過程與前面都相同，不同的點在於前面方法只建立一棵樹、做一次 test，如此循環五次。而我在 random forest 先將資料切成 training data 跟 test data 並每一次用部份的 training data 建立樹，一共九棵每一棵的 training data 數量都一樣，每建一棵就將所以有 training data 洗亂，再選出下一個樹的 training data。
2. 而 test data 放入時，則是根據每一個樹投票共同決定這筆 test data 分類為何，由多數的作為此筆 data 的分類。
3. k-fold 實做為每次建立九棵樹，並對他們做 test，如此循環五次。

三、shell script

1. 這是我第一次自己撰寫 shell script，之前大部分都是直接手打，或是複製貼上，使用 shell script 之後覺得方便許多，只要執行.sh 就可以完成所有動作，shell script 內大容至上就是，進入目標目錄，編譯檔案，產生可執行檔案，執行檔案輸出結果，相當的簡單好用。

四、遭遇的困難以及自我檢討

1. 原本剛開始想使用 python 來撰寫，但是由於作業限制諸多，不能使用模板，且對於 python 不算太熟悉，所以放棄 python，改由最熟悉的 c++做撰寫，也因為使用 c++做撰寫，讓我的觀念比較清楚，知道每一步要如何處理，使用自己所撰寫的 function。

2. 一剛開始撰寫第一個遭遇的困難是資料的切割，不知道該使用何種方法將資料除存，後來經過討論決定將資料寫成一個 class，也方便日後做管理。

3. 架構真的很重要，一個好的架構能讓觀念清楚也讓 debug 時輕鬆許多，這是我還要多多學習的地方，在寫 code 前一定要先想好架構，絕對不能邊寫邊想，否則將會造成 code 凌亂且很難 debug。

4. 剛開始在 windows 上撰寫，結果發現放到 ubuntu16.04 上居然無法執行，請教朋友研究了很久發現是讀資料的問題，因此把資料處理的部份改成可以讀取的方法，學到寶貴的一課，要先看測試環境，並一開始就直接用測試環境做為開發環境。

結果

```
joseph@joseph-G56JR: ~/0316323
1 1
0.95 0.907533
0.916739 0.96
joseph@joseph-G56JR:~/0316323$ chmod +x run.sh; ./run.sh
0.96
1 1
0.943636 0.95
0.941414 0.952381
joseph@joseph-G56JR:~/0316323$ chmod +x run.sh; ./run.sh
0.953333
1 1
0.94127 0.916667
0.933846 0.943636
joseph@joseph-G56JR:~/0316323$ chmod +x run.sh; ./run.sh
0.96
1 1
0.951049 0.940659
0.952381 0.93
joseph@joseph-G56JR:~/0316323$ chmod +x RF.sh; ./RF.sh
0.966667
1 1
0.961111 0.944444
0.944444 0.960714
joseph@joseph-G56JR:~/0316323$
```

環境為 ubuntu 16.04 使用語言為 c++

run.sh:

```
終端機
#!/bin/bash
cd ../0316323
g++ -std=c++11 0316323_ID3.cpp -o aaa
./aaa
```

RF.sh:

```
終端機
#!/bin/bash
cd ../0316323
g++ -std=c++11 0316323_RF.cpp -o fff
./fff
~
```

使用 library

```
#include<iostream>
#include<fstream>
#include<string>
#include<vector>
#include<math.h>
#include<algorithm>
#include<time.h>
```