作業五

404410039 李維哲

一、方法與流程

最後成果:

主要方法:

使用 RandomForest

並使用 RandomOverSampler 來平衡資料量的差距

流程:

1.讀入.csv 資料庫(使用 pandas)

2. preprocess:

對 train(TraData.csv)跟 test(input.csv)作 labelencoder

目的是因為 RandomForest 的工具只支援 int->需要 encode

3.將 train 切成 train 跟 vaild

(vaild 是後面用來測試 train 中的準確度的)

4.preprocessing:RandomOverSampler

5.建立 RandomForest 的 classfier，並利用 train data 作 fit

(使用 sklearn 工具)

6.利用 fit 出來的 RandomForest 驗證在 train 的準確度，並印在螢幕上

6.利用 fit 出來的 RandomForest 預測 test data 的結果(click)

5.將結果輸出成.csv 檔(使用 pandas)

結果:

Accuracy	0.997
Precision	0.092
Recall	0.093
F1	0.092

自己原本的方法:

主要方法:

使用 RandomForest

流程:

1.同時讀入.csv 資料庫(使用 pandas)

同時讀入的目的是為了等下 encode 時統一一種格式

把 train 資料的 click 數切開，分成了 fullData 跟 fullTarget

2.同時對 train 跟 test 作 labelencoder

目的是因為 RandomForest 的工具只支援 int->需要 encode

3.建立 RandomForest 的 classfier，並利用 train data 作 fit

(使用 sklearn 工具)

為了檢查正確度，有從 fullData 中切出 testdata(來自 traindata 的)，以此來計算在 trainData 中的準確度

4.利用 fit 出來的 RandomForest 預測 test data 的結果(click)

5.將結果輸出成.csv 檔(使用 pandas)

結果:

Accuracy	0.998
Precision	0.154
Recall	0.031
F1	0.052

方法說明:RandomForest

優點:

1.對於很多種資料，它可以產生高準確度的分類器。

2.它可以處理大量的輸入變數。

3.它可以在決定類別時，評估變數的重要性。

4.在建造森林時，它可以在內部對於一般化後的誤差產生不偏差的估計。

5.它包含一個好方法可以估計遺失的資料，並且，如果有很大一部分的資料遺失，仍可以維持準確度。

6.它提供一個實驗方法，可以去偵測 variable interactions。

7.對於不平衡的分類資料集來說，它可以平衡誤差。

8.它計算各例中的親近度，對於數據挖掘、偵測離群點（outlier）和將資料視覺化非常有用。

9.使用上述。它可被延伸應用在未標記的資料上，這類資料通常是使用非監督式聚類。也可偵測偏離者和觀看資料。

10 學習過程是很快速的。

(來源:wiki)

在這其中，最主要使用 RandomForest 的原因在於其可以**平衡誤差，花費時間不是太高**

實作:

RandomForest:

隨機作出許多 DecisionTree，再根據這些 DecisionTree 做 ensemble&boosting

boosting:

針對每個 DecisionTree 的錯誤增加權重來加強訓練

ensemble:

在此用法為各 DecisionTree 的加權總和

RandomOverSampler:

把比較少的資料多複製幾份，讓訓練資料可以平衡(缺點:可能 overfit 不過這

字的資料有不平衡，所以此方法可行)

二、組員評分

林佑荃:3 分

人有出現，稍微有說話，不過都是說自己寫不出來，有試著提問自己卡著的地方出了什麼錯，不過也沒有提出做出來的成果，幾乎不知道進度到哪了。

林政賢:7 分

有嘗試過很多 classfier 的參數設定，還蠻投入的，不過可惜的是好像沒能趕上 test 的時間，並沒有跟我們討論他做出來的 F1score 如何。

李淼晶:1 分

人有出現，但沒貢獻，只有在剛開群組時出現，打個招呼，然後就沒有然後了，中間一度失蹤，直到最後一刻最高 score 的人開源時才出現

廖翊凱:10 分

有幫助隊員 debug，還蠻投入的，除了基本的 RandomForest 之外，還有提出使用 RandomOverSampler 的方法來平衡資料差異，因為結果 F1score 比較高，所以最後採用他的版本