Kaggle 競賽

Ghouls, Goblins, and Ghosts... Boo!

訓練資料集說明

共 371 筆資料、7 個特徵(含應變量: type)

id	bone_length	rotting_flesh	hair_length	has_soul	color	type
0	0.3545122	0.3508390	0.4657609	0.781141666	clear	Ghoul
1	0.5755599	0.4258684	0.5314014	0.439898877	green	Goblin
2	0.4678755	0.3543304	0.8116161	0.791224973	black	Ghoul
4	0.7766525	0.5087225	0.6367656	0.884463692	black	Ghoul
5	0.5661166	0.8758618	0.4185937	0.636437819	green	Ghost
7	0.4056797	0.2532775	0.4414197	0.280323820	green	Goblin
8	0.3993309	0.5689518	0.6183910	0.467900835	white	Goblin
11	0.5162239	0.5364287	0.6127761	0.468048270	clear	Ghoul
12	0.3142953	0.6712797	0.4172669	0.227547575	blue	Ghost

1. 資料清洗 (R-code 3~22)
   - 觀察 2~4 欄(數值型特徵)的離群值，如圖 A，發現 2、3 欄共有五個離群值，剔除 (樣本數變為 366)

```
> par(mfrow=c(1,4))
> boxplot(data$bone_length)$out
[1] 0.81700143 0.06103182
> boxplot(data$rotting_flesh)$out
[1] 0.93246609 0.09568665 0.92943959
> boxplot(data$hair_length)$out
numeric(0)
> boxplot(data$has_soul)$out
numeric(0)
```

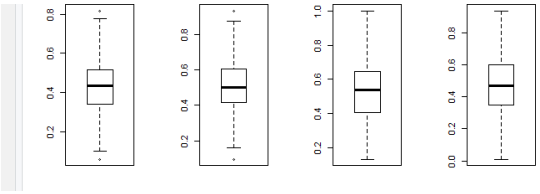

圖 A: 盒狀圖找離群值

   - 透過 shapiro.test 發現 2~4 欄皆為常態分佈 >> 考慮迴歸(但成效不佳)

2. 類別特徵處理 (R-code 24~34)
   透過單熱變數轉換，將資料自變量變為 11 個(剔除 id)，應變量(type)不變

3. 集群分析 (python-code)
   將上步的 11 個數值型資料各自標準化，依據 silhouette 指標(如圖 B)，決定分 6 群

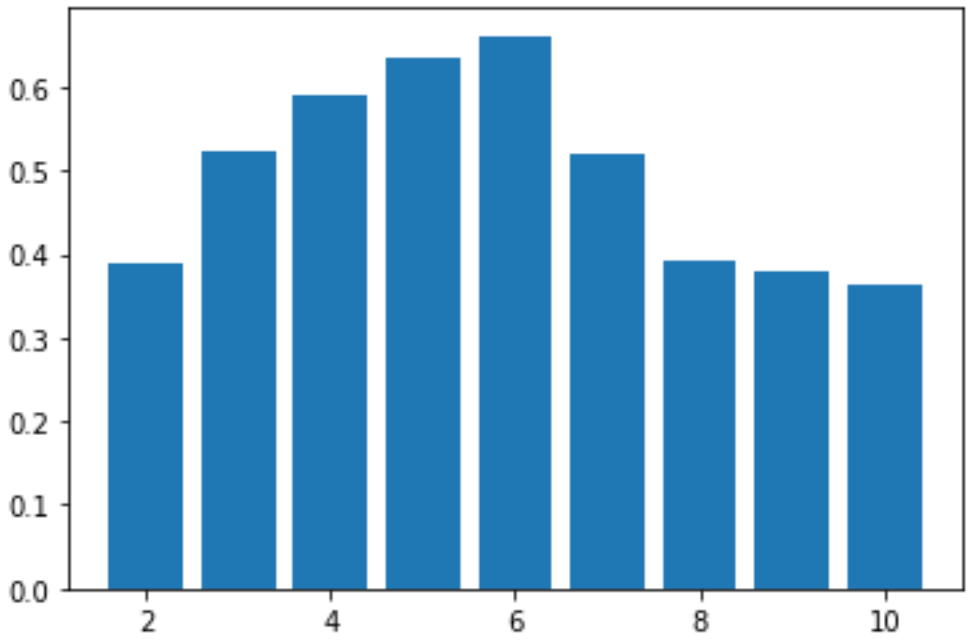

圖 B: silhouette 指標

透過 k-means 配合歐式距離與 k-means++找中心，最後分成 6 組

PS: 因集群分析的計算量常常比較大，習慣使用 python 做

4. 集群應用 (R-code 38~44)
   - 將組別標籤(0~5)視為新特徵 (預測效果提升)
   - 4、5 組樣本數合計 30，視為離群值，剔除兩組 (預測效果提升)
   - 依照組別分組建模 (預測效果下降，可能是樣本數太少)

5. 本次嘗試過隨機森林、GB、SVM 三種方法來建模，成效差異不大，皆為 7 成，故選用模型解釋力較高的 SVM。

6. 特徵篩選 (R-code 45~53)
   透過 AAD、variance 指標來觀察 11 個特徵在 svm 下的重要性，如圖 C，經多次建模測試，最終以 bone_length、has_soul、hair_length、rotting_flesh、group 為自變量的預測效果最佳。

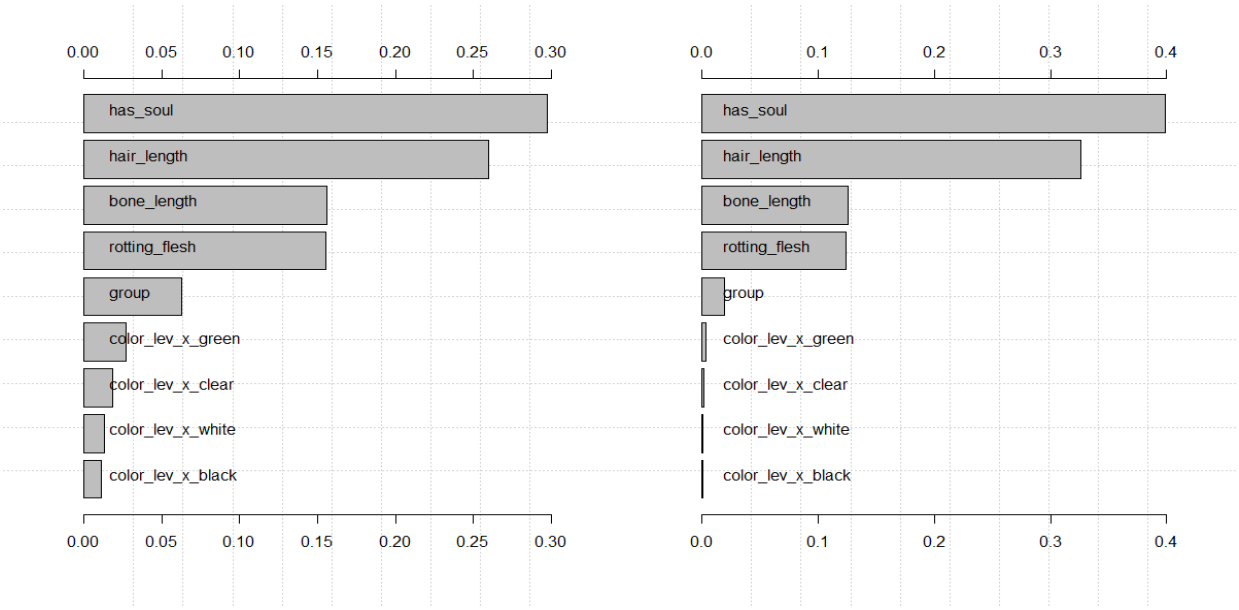

圖 C: 左邊為 AAD 指標，右邊為 variance 指標

7. 建模 (R-code 55~67)
   - 調整參數
     在 10-fold 下，最佳參數為 {cost=1, gamma=0.0625}
   - 建立 svm
     因為本次為類別行預測，故選用 C-classification 型態下的 svm，且又為了有效率的處理非線性模型，故選用 radial 為核函數

8. 預測結果 (R-code 68~87)
   雖然做了一連串的前置作業，但比起直接將所有資料丟入模型建模(都用預測參數)，準確度只提升了 4%，模型最終表現如圖 D。

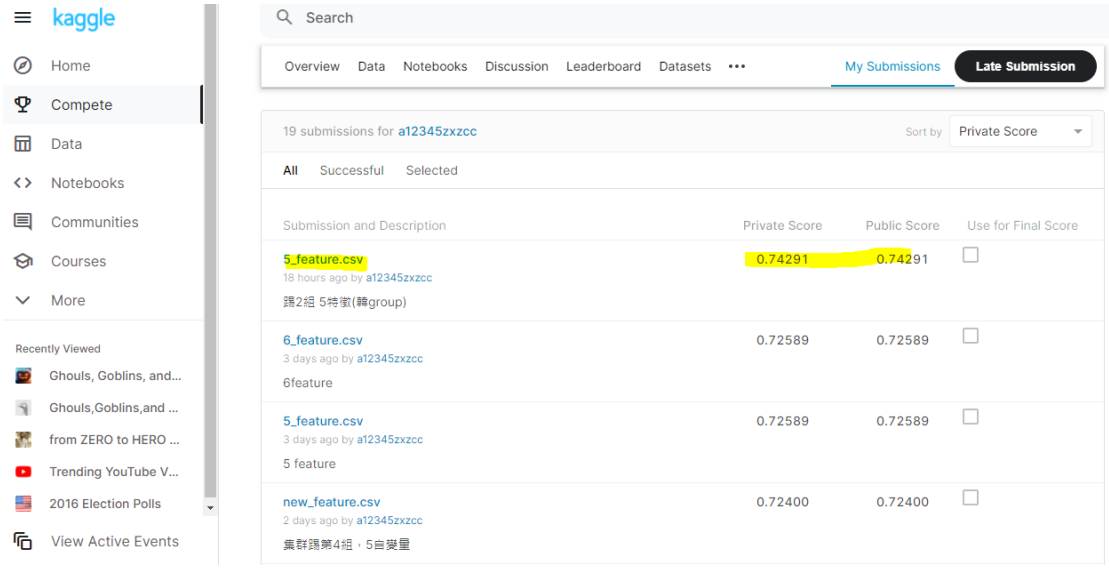

圖 D: 預測結果