Machine Learning – Assignment I

楊子萱
Institute of Data Science
National Cheng Kung University
Tainan, Taiwan
re6114056@gs.ncku.edu.tw

***Abstract*—利用 crx 與 data 這兩個簡易的資料集去測試實作的 model，並比較不同 model 間預測準確性的差異，最終觀察到幾乎所有的分類器都能夠在 data 這個資料集取得 0.99 的 Accuracy，而在於 crx 資料集中則出現較大的分類正確性差異，結果最好的為 linear classifier 的 0.85。**

Keywords—linear classifier, voted perception, SVM

I. INTRODUCTION

利用 crx 與 data 這兩個資料集去測試實作的 linear classifier, linear classier with least-squared manner, vote perception, Basic SVM 以及 Soft SVM，並且再取用 Sklearn 的 SVM，去比較實作的 SVM 與 package 中的 SVM 在於預測上正確性的差異，最終選出最適的 model。

II. DATA PREPROCESSING

A. *crx.csv*

crx.csv 中的 target feature 為 label，將 label 中的+/-，轉化成為 0/1 以去做後續的分類任務，扣除掉 target feature 後總共有 15 個欄位，將數值型欄位做 Normalize，類別型欄位因為是做 classifier，因此值的大小不影響分類，因此將該類別轉成數字去代表該類別，最終將資料集儲存成 tidy_crx.csv，以去做後續的分類任務。

B. *data.csv*

data.csv 中的 target feature 為 Diagnosis，將 label 中的 B/M，轉化成為 0/1 以去做後續的分類任務，將認為不相關的欄位去除(ex:ID, unnamed column)，並扣除掉 target feature 後總共有 30 個欄位，因為所有的欄位皆為數值型欄位，因此將所有的欄位做 Normalize，最終將資料集儲存成 tidy_data.csv，以去做後續的分類任務。

III. METHOD OF PROBLEMS

A. *Linear classifier*

線性分類器透過簡單的權重與該特徵的加總，去得出最終預測的結果，預測公式如下：

$$Prediction = \sum_{i=1}^{n} w_i f_i \quad (1)$$

在於分類錯誤時，會針對預測的結果，將特徵往結果的反方向做權重的更新，更新公式如下：

$$\text{For each } w_i,\ w_i = f_i * label \quad (2)$$

- 步驟一：初始化權重矩陣W。
- 步驟二：根據權重矩陣與 Input X去計算出預測值
- 步驟三：當預測值與 Ground truth 不同時，即預測錯誤，則更新權重W
- 步驟四：重複二、三步驟直到訓練完所有的 Input

完整的程式碼能夠於 linearclassifier.py 中查看，準確性的部分 crx 最佳的為 0.85，而 data 中最佳的為 0.98，在於設計上另外設計了加入 bias 的版本，但是在測試後發現在於兩個資料集上加入 bias 反而導致正確性下降，導致 overfitting 的問題。

B. *linear classier with least-squared manner*

相較於 Q1 較簡單，不需要透過單一資料去更新參數，能夠透過最小平方法的方式，透過公式解去求得最終的預測權重，公式如下：

$$W = (X^T X)^{-1} X^T Y \quad (3)$$

- 步驟一：透過最小平方法得出權重矩陣W。
- 步驟二：將權重矩陣與 Input X去計算出預測值
- 步驟三：當預測值與 Ground truth 相同時，則預測正確計數加一，最終計算預測準確率

	Accuracy	
	crx	data
linear classifier	0.85	0.99
linear classifier with bias	0.66	0.99
linear classifier with least-squared	0.84	0.96

由上面的表格中可以看出，least-squared 更新方式的正確性比起普通的線性分類器更新方式來的差一點點，但是可以簡化持續不斷的更新導致的時間消耗。

C. *Voted perception*

Voted perception 是一種會記錄該權重，以及該權重下預測正確的次數，最終在於預測時，使用所有的權重去預測該 test data，並透過該權重的正確預測次數給予該權重的預測結果進行加權平均得出最終的結果。

- 步驟一：初始化權重矩陣W。
- 步驟二：根據權重矩陣與 Input X 去計算出預測值
- 步驟三：當預測正確時，C_m會加一，但當預測錯誤時，更新權重矩陣W，C_m重頭由一開始計數
- 步驟四：重複二、三步驟直到訓練完所有的 Input

W為權重矩陣，C_m該權重矩陣預測正確的次數，m 為權重矩陣的 index。

詳細的程式碼可以在 voted_perception.py 中查看。

	Accuracy	
	crx	data
linear classifier	0.85	0.99
linear classifier with least-squared	0.84	0.96
voted_perception	0.85	0.99

在於最終的預測正確性的部分，幾乎與前幾個模型沒有差異，但是 voted_perception 在於計算上因為需要去將所有的權重以及計數都記錄下來，因此在訓練時，會相較於前兩個方法來的需要更大的空間複雜度，而且在於預測上，因為在於每一個 test data 預測時，會需要去預測於所有的權重後做加權平均，因此會需要比起其他方法來的更大的時間複雜度。

D. *SVM*

SVM 是透過尋找支持向量(support vector)的方式去最大化決策邊界的 margin，使得在線性不可分的資料中，透過映射空間的轉化達成線性可分的一種機器學習模型。

在於預測結果上 crx 的最佳準確性為 0.64，而 data 的最佳準確性為 0.92，皆沒有前幾個方法的準確性來的高。

E. *Soft margin SVM*

Soft margin SVM 即是在上題中的 SVM 加入一個容忍項 slack variable，在幾何意義上則表示容忍誤差的存在，將在 S 範圍內的點視為不確定性因子，因此當 S 設為 0 時，即等於上題的 Basic SVM(i.e. Hard margin SVM)，而當 S 設的過大時，則會無法正確地去將目標分類，因此 S 的設置需要多方嘗試。

在於預測結果上 crx 的最佳準確性為 0.69，而 data 的最佳準確性為 0.88，皆沒有前幾個方法的準確性來的高

F. *SVM by sklearn*

套用 sklearn 中已撰寫好的 svm 套件做分類器，以此為對照組，比較自行撰寫的分類器效能.

網址：https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html

	Accuracy	
	crx	data
Basic SVM	0.65	0.93
Soft margin SVM	0.69	0.88
Sklearn SVM	0.88	0.94

可以由上述表格中看出，在於兩個資料集中，皆是 Sklearn 的 SVM 在於預測上效果較好，可能 sklearn 上的 SVM 在於設計上有許多 robust 的方法用於預測上，且也能夠看出在於 crx 的資料集上的正確性高於本次作業中所實做的所有分類器。

IV. CONCLUSION

在於本次實作結果上，與現存的分類方法準確性部分並無明顯差異，推測可能的原因是因為本次的資料集不同類別中差異較明顯，因此透過簡易的分類器就能夠將類別分類出來，加上更多的誤差項反而會去影響預測結果。

期望未來能夠再將此分類器，去實際使用於其他資料集中在去細部觀察其中的差異性，或者是優化模型增加更多可以調控的參數。

GitHub 網址：https://github.com/tenyang1999/ML-assignment---1