105061254 林士平 數位訊號處理實驗報告 Lab7

1. Abstract/Introduction

本次的實驗主要介紹兩個重要的觀念：**SVM** 和 **Cross Validation**。SVM 是一種 supervised learning，可以用作 classification；Cross Validation 則是評斷機器學習模型好壞的重要方式。

第一個部分使用 SVM(Support Vector Machine)來分類用 make_moons 這個 function 做出來的 dataset。此 dataset 為 **nonlinearly seperable**，所以分別使用 linear 和 non-linear kernels 做 training 可以看出兩者 performance 的差異，並且把 decision boundary 和 data 的分布畫出來，可以更清楚地看到分類的狀況。

第二個部分則是用機器學習相當經典的 iris dataset，首先使用 SVM 做分類，然後利用 **Cross Validation** 來評估這個模型的好壞，最後把 **confusion matrix** 畫出來。

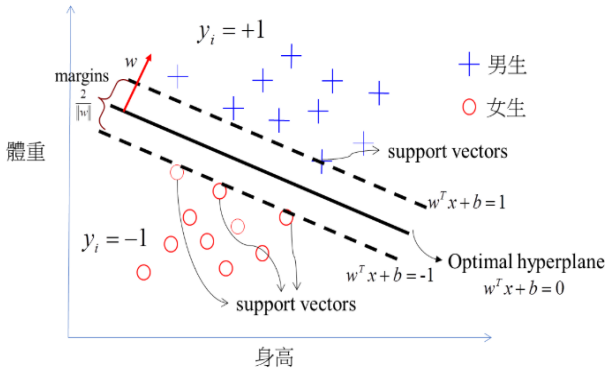
本實驗藉由實際操作 SVM 和 Cross Validation 來學習基本的機器學習技巧。

2. Goals of this lab

Demo 1 (Moon_SVM):
(1) Use both linear and non-linear kernels
(2) Plot decision boundary of SVM
Demo 2 (IRIS_CrossValidation):
(1) Implement K-fold cross-validation
(2) Plot confusion matrix

3. Method

SVM(Support vector machine)的概念非常簡單，就是找到一個決策邊界(decision boundary)**讓兩類之間的邊界(margins)最大化**，使其完美區隔開來。



由上頁的示意圖可以看到男生和女生之間有一條明顯的界線，SVM 的目標就是讓 margin 越大越好。

其中 SVM 有個重要的參數 C，**C 越大 model 會有 high accuracy 但 poor generalization，相對的 C 越小 model 會有 low accuracy 但 good generalization**。

在做 training 前我們會把 Data 分成 **training set 和 testing set**。Training set 用來訓練出模型的參數，然後用 testing set 來確認這個模型的好壞。Cross-validation 即是重複這個過程，其中 K-fold 是比較常用的交叉驗證方法。做法是將資料**隨機平均分成 k 個集合**，然後將某一個集合當做「測試資料(Testing data)」，剩下的 k-1 個集合做為「訓練資料(Training data)」，如此**重複進行直到每一個集合都被當做「測試資料(Testing data)」**為止。最後的結果(Predication results)再和真實答案(ground truth)進行成效比對(Performance Comparison)。

4. Pseudo code

以下是 Demo 1(Moon_SVM)程式碼重點：

(0)	*# choose linear kernel here* *model = SVC(kernel='linear', random_state=0)* *# choose a nonlinear kernel here* *model = SVC(kernel='rbf', random_state=0, C = 10)* **說明：整個程式碼大部分已經打好，僅需更改要使用哪種 kernal 以及 SVC 的參數例如 C、gamma 等。我 C 選用 10，如此一來可以讓 accuracy 達到 100%。**

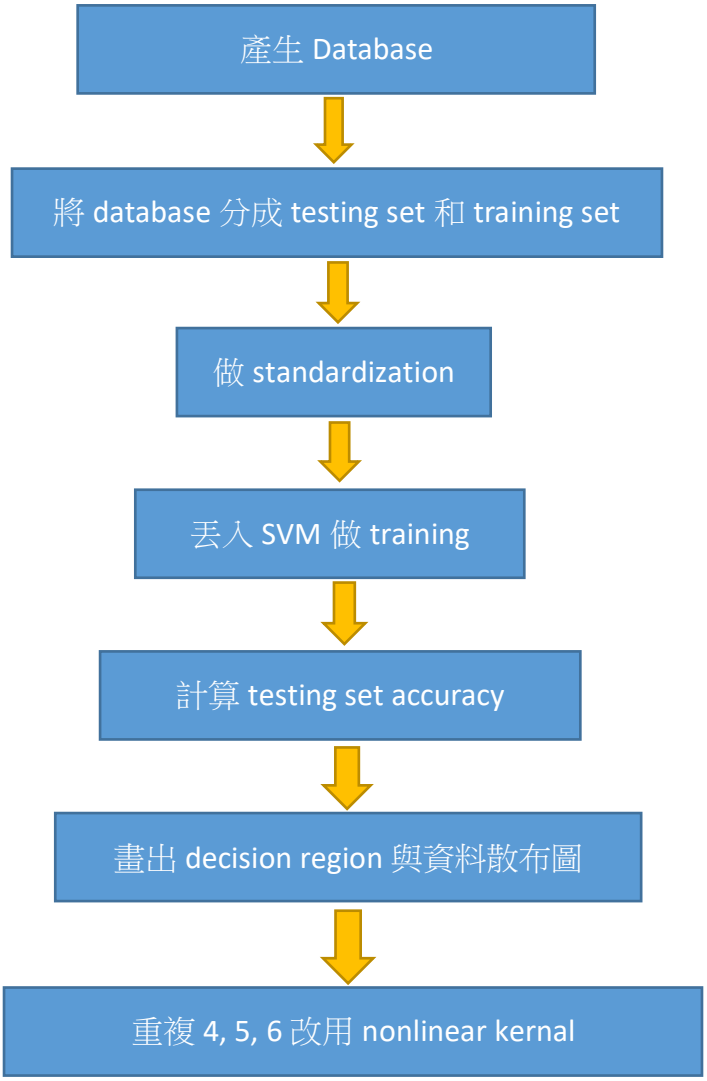
以下是 Demo 2(IRIS_CrossValidation)程式碼重點：

(0)	*# Parameters* *RANDSEED = 1 # setupt random seed for repeatness* *CVFOLD = 5 # number of folds of cross validation* *c = 10* **說明：設定參數，使用 5-fold Cross Validation，並且設定 c = 10。**
(1)	*# cross validation* *Kf = KFold(n_splits=CVFOLD, shuffle=True, random_state=RANDSEED)* *y_test_cv = []* *y_predict_cv = []* *for cvIdx, (trainIdx, testIdx) in enumerate(Kf.split(range(len(X)))):* *# split data into Train & Test* *X_train, X_test = X[trainIdx], X[testIdx]*

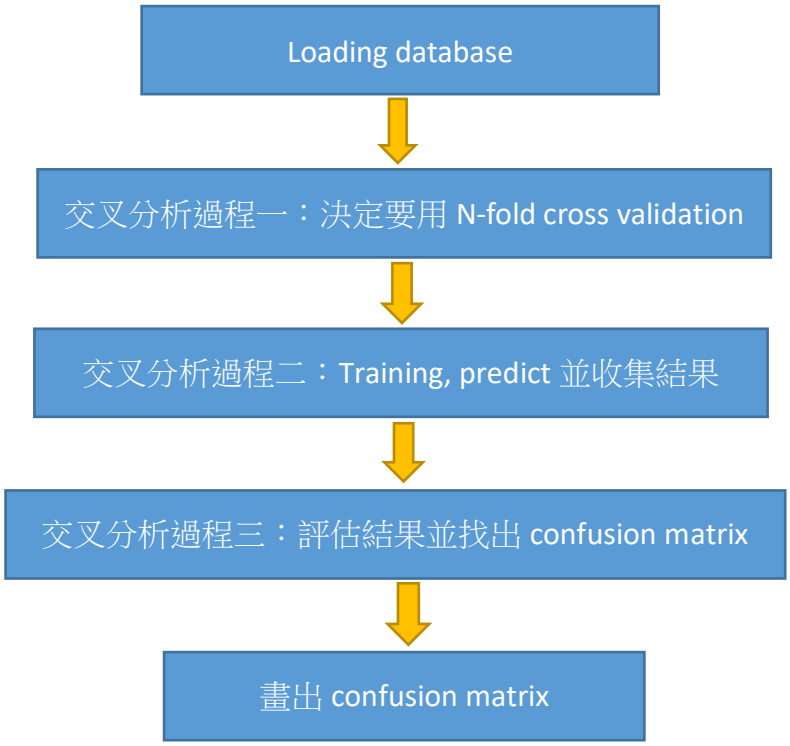
	y_train, y_test = y[trainIdx], y[testIdx] *# perform the same train / testing process in IRIS-TrainingTest* *clf = LinearSVC(C=c , class_weight='balanced')* *# Note that u have to build a new classifier too!* *clf.fit(X_train, y_train)* *y_predict= clf.predict(X_test)* *# collect the predict results and ground truths from each folds* *y_test_cv.extend(y_test)* *y_predict_cv.extend(y_predict)* 說明：本部分做交叉驗證。每次先將資料分為 **training set** 和 **testing set**，然後做 **training**，最後將結果收集起來，再重複此過程直到完成 **N-fold cross validation**。

5. Flow chart

以下是 Demo 1(Moon_SVM)的 flow chart：



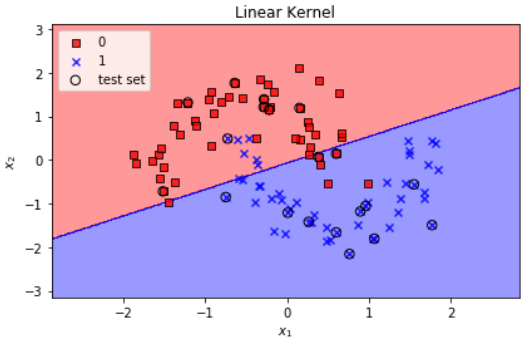
以下是 Demo 2(IRIS_CrossValidation)的 flow chart：



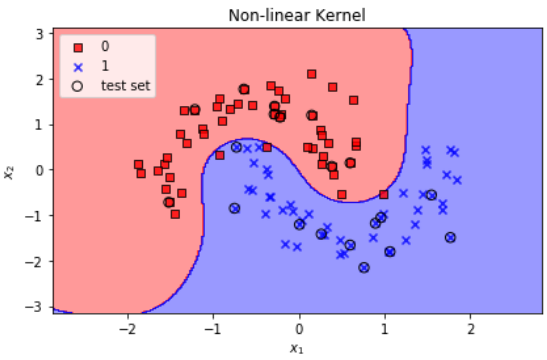
6. Results

Demo 1 :

```
[Linear SVC]
Misclassified samples: 3
Accuracy: 0.85
```

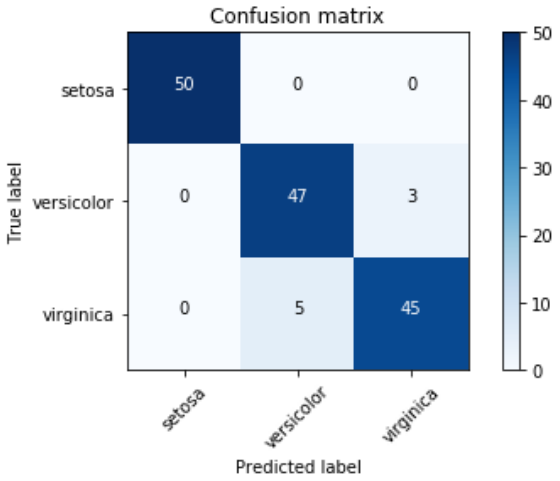



```
[Nonlinear SVC]
Misclassified samples: 0
Accuracy: 1.00
```



Demo 2 :

```
Confusion matrix
[[50  0  0]
 [ 0 47  3]
 [ 0  5 45]]
UAR =  0.9466666666666667
```



7. Reference

(1) sklearn.datasets.make_moons：
https://scikitlearn.org/stable/modules/generated/sklearn.datasets.make_moons.html

(2) 交叉驗證(Cross-Validation)：
https://medium.com/@chih.sheng.huang821/%E4%BA%A4%E5%8F%89%E9%A9%97%E8%AD%89-cross-validation-cv-3b2c714b18db

(3) 機器學習 – 支撐向量機(support vector machine, SVM)詳細推導：
https://medium.com/@chih.sheng.huang821/%E6%A9%9F%E5%99%A8%E5%AD%B8%E7%BF%92-%E6%94%AF%E6%92%90%E5%90%91%E9%87%8F%E6%A9%9F-support-vector-machine-svm-%E8%A9%B3%E7%B4%B0%E6%8E%A8%E5%B0%8E-c320098a3d2e