HW3 – Cross validation

一、 程式碼截圖與講解：

1. **資料前處理：**

```
# 1 Read file
data = pd.read_csv('E:/DM/HW3/data.csv')

# 2-1 replace all tha space in head and tail
data = data.replace('^\s+','',regex=True).replace('\s+$','',regex=True)

# 2-2 drop data with ?
data = data.replace('?',np.nan).dropna()

# 2-3 set label:income>50 as 1
data = pd.get_dummies(data).drop(['income_>50K'],axis=1)
```

a、 1 讀檔案

b、 2-1 使用正規表示法，將檔案中文字格字首和字尾的空格刪除

c、 2-2 drop 掉資料當中存在’?’ 的資料

d、 2-3 使用 get_dummies，並刪除資料最後一欄，以 income_<=50 為 y

2. **撰寫 function 進行 k-fold cross-validation 並計算 Accuracy：**

```
# 2-4 set trainning set and testing set
x = data.values[:,:-1]
y = data.values[:,-1].reshape(-1,1)
```

a、 2-4 設定好 x 和 y 的部分

```
# 3 Define function
def Cross_validation(k, data):
    #設定subset size 即data長度/k
    group = np.mod(np.arange(len(data)), k).reshape(-1,1)
    #設定Accuracy初始值
    Accuracy = 0
    for i in range(k):
        #設定testing set與training set的資料起始點與結束點
        trainidx = np.where(group!=i)[0]
        testidx = np.where(group==i)[0]
        #例如資料有100筆，testing set在本次iteration取第1到10筆，則training set為
        train_x, train_y = x[trainidx], y[trainidx]
        test_x, test_y =  x[testidx], y[testidx]
        # GradientBoostingClassifier
        clf = GradientBoostingClassifier().fit(train_x, train_y)
        print(clf.score(test_x, test_y))
        #利用training set建立模型，testing set計算出Accuracy累加
        Accuracy = Accuracy + clf.score(test_x, test_y)
    return Accuracy/k
```

b、 **3 function 撰寫：**

i. 以 mod 分出 k 群

ii. 經過 for 迭代跑完 k 群資料的 Gradient Boosting Classifier

iii. 最後將 Accuracy 累積後取平均，return

3. **呼叫 function** `Ans = Cross_validation(10,data)`

二、 程式實作結果：

a. **十次 Accuracy 如下：**

```
0.8544912164401723
0.8680808750414318
0.860079575596817
0.8680371352785146
0.8677055702917772
0.8580901856763926
0.866710875331565
0.8514588859416445
0.8541114058355438
0.863395225464191
```

b. **最終結果如下：**

Ans	float64	1	0.8612160950898049

三、 作業心得：

因為智慧型作業當中實作過 Cross-validation，且在 Data mining 第一次作業時做過類似的資料處理，因此在這次作業上做起來比較順手一點。比較不一樣的是跑 Gradient Boosting Classifier，每次跑起來都蠻費時的，常常以為電腦壞了哈哈哈哈。