機器學習作業 Lab03 報告

資訊工程學系 黃右萱(0416323)

問題一：

做法：

概念上，利用高斯分布的假設訓練 naïve 貝式模型；

實作上利用 sklearn 下之 GaussianNB 進行訓練

首先，對資料進行 moving average 平滑化：

```
window = 2
f = 6
for k in range(6):
    for i in range(len(data)-window):
        avg = 0.0
        for j in(i,i+window):
            avg+=data[j][k]
        avg/=window
        data[i][k]=avg
```

然後再丟入模型進行訓練：

```
model = GaussianNB().fit(x,y)
print codeStatus(model.predict([query])[0])
```

預測結果如下圖所示：

```
settler
```

Library:

1. sklearn
2. numpy

```
import numpy as np
import sklearn
from sklearn.naive_bayes import GaussianNB
```

問題二：

作法：

首先利用 re 模組之正則表達式去除無用資料與篩選出合適資料：

```
[train, test] = [i for i in (re.split(r'(--- training ---)|(--- testing ---)',dataFile.read())) if i!=None and len(i)>20]
train = [[j for j in re.split(r',|\s',i) if len(j)>0] for i in re.split('\n', train) if '?' not in i and len(i)>2]
test = [[j for j in re.split(r',|\s',i) if len(j)>0] for i in re.split('\n', test) if '?' not in i and len(i)>2]
train = np.array(train)
test = np.array(test)
trainY, trainX = train[:,0], train[:,1:].astype(float)
testY, testX = test[:,0], test[:,1:].astype(float)
trainY = [[int(j) for j in re.split('\D*',i) if len(j)>0] for i in trainY]
testY = [[int(j) for j in re.split('\D*',i) if len(j)>0] for i in testY]
attr = re.split(r'-- Class\s\d:.*\r\n',attrFile.read())[1:]
attr = [[term for term in re.split(r'[,\s]',block) if len(term)>0] for block in attr]
```

接著建立雜湊表、解析字串，將有效資料進一步解析成數值資料：

```
for i,block in enumerate(attr):
    for j,term in enumerate(block):
        if(term == 'to'):
            block[j-1:j+2] = [''.join(block[j-1:j+2])]
    attr[i] = [[int(digit) for digit in re.split(r'\D*',term) if len(digit)>0] for term in block if len(term)>0]
def getDate(lis):
    return lis[0]+lis[1]*100+lis[2]*10000
attr = [[[getDate(dates)] if len(dates)==3 else
    range(getDate(dates[0:3]), getDate(dates[3:6])+1)
    for dates in block] for block in attr]
trainX = list(trainX)
dClass = {}
for i, block in enumerate(attr):
    for sub in block:
        for term in sub:
            dClass.update({term: i+1})
for i,date in enumerate(trainY):
    temp = date
    date = getDate(date)
    if (date not in dClass):
        trainY[i] = None
        trainX[i] = None
        continue
    trainY[i] = dClass[date]
trainX = [i for i in trainX if i is not None]
trainY = [i for i in trainY if i is not None]
```

然後放入 GaussianNB 模型中進行預測，15-fold、resubstitution 準確率分別約為 89%與 73%左右，預測結果如下圖所示(以 Class 的編號表示)：

```
[1 5 3 1 1 1 5 3 5 5 1 1 3 5 3 1 1 1 1]
```

使用方法：在程式碼最下方對 model 物件呼叫 predict 方法，參數填入 feature，方法回傳值即為 prediction。

Library:

1. sklearn
2. numpy
3. string
4. re
5. sys

```
import numpy as np
import sklearn, re, sys
import string
```

```
from sklearn.naive_bayes import GaussianNB
from sklearn.model_selection import KFold
from sklearn import preprocessing
```

環境：

Ubuntu 14.04.5 LTS

Python 2.7.6