

人工智慧理論與實作 - HW3

311707006 汪文豪

題目：

1. 如上課教學影片所述，請把 ID3 decision tree 二元分類方法 (ID3DTtrain 函式) 實作完成。

已完成：

```
24 def ID3DTtrain(feature, target):
25     node = dict()
26     node['data'] = range(len(target)) # root initialization
27     tree = []
28     tree.append(node)
29     t = 0
30     while(t < len(tree)):
31         idx = tree[t]['data']
32         if(sum(target[idx]) == 0):
33             tree[t]['leaf'] = 1
34             tree[t]['decision'] = 0
35         elif(sum(target[idx]) == len(idx)): # 都是1 加起來就是長度
36             tree[t]['leaf'] = 1
37             tree[t]['decision'] = 1
38         else:
39             bestIG = 0
40             for i in range(feature.shape[1]): # TRY每個特徵切區塊
41                 pool = list(set(feature[idx, i]))
42                 pool.sort()
43                 for j in range(len(pool)-1):
44                     thres = (pool[j] + pool[j+1])/2
45                     G1 = []
46                     G2 = []
47                     for k in idx:
48                         if(feature[k][i] < thres):
49                             G1.append(k)
50                         else:
51                             G2.append(k)
52                     p1 = sum(target[G1] == 1)
53                     n1 = sum(target[G1] == 0)
54                     p2 = sum(target[G2] == 1)
55                     n2 = sum(target[G2] == 0)
56                     thisIG = InformationGain(p1, n1, p2, n2)
57                     if(thisIG > bestIG):
58                         bestIG = thisIG
59                         bestG1 = G1
60                         bestG2 = G2
61                         bestThres = thres
62                         bestFeature = i
63             if(bestIG > 0):
64                 tree[t]['leaf'] = 0
65                 tree[t]['selectf'] = bestFeature
66                 tree[t]['threshold'] = bestThres
67                 tree[t]['child'] = [len(tree), len(tree) + 1]
```

2. 完成單一樣本測試函式 (ID3DTtest 函式)，輸入為訓練好的 tree 跟單一
樣本的特徵，輸出為該樣本的預測分類。

已完成：

```
def ID3DTtest(tree, feature1):
    now = 0
    while(tree[now]['leaf'] == 0):
        bestf = tree[now]['selectf']
        thres = tree[now]['threshold']
        if(feature1[bestf] < thres):
            now = tree[now]['child'][0]
        else:
            now = tree[now]['child'][1]
    return tree[now]['decision']
```

3. IRIS 資料庫共有 150 筆資料，共有三類花各 50 筆，今僅取後 100 筆
(即 target 為 1 及 2 兩類) 來實驗，請用這 100 筆資料建樹，再用相同
資料測試，試算其預測正確率。

101 行：用後 100 筆資料建樹

102 - 106 行：使用 feature 資料預測結果並結果與真實值(target)比較，計
算錯誤數量

107 行：正確率印出

```
100 # 第三題
101 T = ID3DTtrain(feature[50:150, :], target[50:150]-1)
102 error = 0
103 for i in range(50, 150):
104     if(ID3DTtest(T, feature[i, :]) != (target[i]-1)):
105         print("Predit: ", ID3DTtest(T, feature[i, :]), "\nReal: ", target[i])
106         error += 1
107 print("正確率: ", (100-error)/100)
108
```

正確率: 1.0

PS: C:\Users\Hson\

4. 承上，若每類改用前 30 筆建樹，後 20 筆測試，試算其測試資料的預測正確率。

112 行 - 116 行：建立訓練與測試資料集

118 行：用訓練資料 train_data 與 train_target 建立樹

120 行 - 125 行：使用測試資料 test_data 預測結果並將結果與真實值 (test_target) 比較，計算錯誤數量

126 行：正確率印出

```
110 # 第四題
111 #####
112 train_data = np.concatenate((feature[50:80, :], feature[100:130, :])) #
113 train_target = np.concatenate((target[50:80] - 1, target[100:130] - 1))
114 #####
115 test_data = np.concatenate((feature[80:100, :], feature[130:150, :])) #
116 test_target = np.concatenate((target[80:100] - 1, target[130:150] - 1))
117 #####
118 T4 = ID3DTtrain(train_data, train_target)
119
120 error_4 = 0
121 for i in range(0, 40):
122     if(ID3DTtest(T4, test_data[i, :]) != (test_target[i])):
123         print("Predit: ", ID3DTtest(
124             T4, test_data[i, :]), "\nReal: ", test_target[i])
125         error_4 += 1
126 print("正確率: ", (40-error_4)/40)
127
```

```
Predit: 1
Real: 0
Predit: 1
Real: 0
正確率: 0.95
```