

2.1

500个正例中选取350个随机组合，反例中同理

$$\binom{350}{500}^2$$

2.3

$BEP = P = R$ ，是当查准率 = 查全率时的取值

$$F1 = \frac{2 * P * R}{P + R}$$

当 $P = R$ 时 $F1 = BEP = P = R$ ，在此情况下若 $F1_A > F1_B$ ，则 $BEP_A > BEP_B$

但 $F1$ 的大小和 BEP 之间没有关系， $F1$ 的值由 P, R 的值决定，而 BEP 仅仅是当 $P=R$ 时的一种特殊情况

2.6

ROC曲线的横坐标为假正例率 ($FPR = \frac{FP}{TN+FP} = \frac{FP}{m^-}$)，纵坐标为真正例率 ($TPR = \frac{TP}{TP+FN} = \frac{TP}{m^+}$)

$$\text{错误率} = \frac{FN+FP}{\text{样本总数}} = \frac{FP+m^+-TP}{m^++m^-} = \frac{m^- * FPR + m^+ (1 - TPR)}{m^++m^-}$$

附加1

$$l_{rank} = \frac{1}{m^+ m^-} \sum_{x^+ \in D^+} \sum_{x^- \in D^-} (\mathbb{I}(f(x^+) < f(x^-)) + \frac{1}{2} \mathbb{I}(f(x^+) = f(x^-)))$$

由书中内容可知，AUC就是ROC曲线下的面积并且

$$AUC = 1 - l_{rank}$$

当正例的预测值大于反例时，所设权重为1，即坐标图中的一个单位面积。

$$\text{总面积} = \sum_{x^+ \in D^+} \sum_{x^- \in D^-} \mathbb{I}(f(x^+) > f(x^-))$$

当正例的预测值等于反例时，所设权重为0.5，即坐标图中的半个单位面积。

$$\text{总面积} = \sum_{x^+ \in D^+} \sum_{x^- \in D^-} \mathbb{I}(f(x^+) = f(x^-))$$

最后二者相加，除以缩放的倍数 $m^+ * m^-$ 得

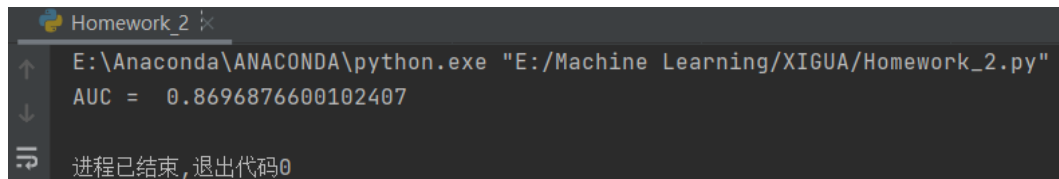
$$AUC = \frac{1}{m^+ m^-} \sum_{x^+ \in D^+} \sum_{x^- \in D^-} (\mathbb{I}(f(x^+) > f(x^-)) + \frac{1}{2} \mathbb{I}(f(x^+) = f(x^-)))$$

附加2

```
1 import matplotlib.pyplot as plt
2 import csv
3
4 # 初始化各个变量
5 label = []
6 recall_list = []
7 precision_list = []
8 FPR = []
9 TP = 0.0
10 FP = 0.0
11
12 # 读取csv文件并将其中的数据保存在对应的list中
13 f = csv.reader(open('data.csv', 'r'))
14 l = [] # l列表经降序排序处理后, 包含 第一关键字为label, 第二关键字为output 的表中数据
15 for i in f:
16     label.append(i[1])
17     l.append([i[1], i[2]])
18
19 # 删去不必要的表头信息
20 del label[0], l[0]
21
22 # 计算出真实情况中的正例和反例数量
23 M = label.count('1')
24 N = label.count('0')
25
26 # 对l中的内容进行第二关键字的排序 降序
27 l.sort(key=lambda x: float(x[1]), reverse=True)
28
29 # 对list进行遍历, 从中计算出TP,FP的值, 此处用了双重循环的, 可以采用dp化简
30 for i in range(500):
31     for k in range(i):
32         if float(l[k][0]) == 1.0:
33             TP += 1.0
34         else:
35             FP += 1.0
36     # 分母不能为0
37     if (TP + FP) != 0.0:
38         precision = TP / (TP + FP)
39         recall = TP / M
40         fpr = FP / N
41         precision_list.append(precision)
42         recall_list.append(recall)
43         FPR.append(fpr)
44     TP = 0.0
45     FP = 0.0
46 # 采用PPT上的方法 计算 AUC 的值
47 AUC = 0.0
48 for i in range(498):
49     AUC = AUC + ((FPR[i + 1] - FPR[i]) * (recall_list[i] + recall_list[i + 1]))
50 AUC = 0.5 * AUC
51 print(AUC) # AUC = 0.8696876600102407
52
53 # 调用matplotlib 输出P-R图 和 ROC图
54 plt.plot(recall_list, precision_list)
```

```
55 plt.title("P-R")
56 plt.ylabel("precision")
57 plt.xlabel("recall")
58 plt.xlim([-0.001, 1.01])
59 plt.ylim([-0.001, 1.01])
60 plt.savefig("P-R.png")
61 plt.show()
62
63 plt.plot(FPR, recall_list)
64 plt.title("ROC")
65 plt.ylabel("TPR")
66 plt.xlabel("FPR")
67 plt.xlim([-0.001, 1.01])
68 plt.ylim([-0.001, 1.01])
69 plt.savefig("ROC.png")
70 plt.show()
71
```

结果如下:

A terminal window titled 'Homework_2' showing the execution of a Python script. The command is 'E:\Anaconda\ANACONDA\python.exe "E:/Machine Learning/XIGUA/Homework_2.py"'. The output is 'AUC = 0.8696876600102407'. The terminal also shows standard navigation icons on the left and a status bar at the bottom indicating '进程已结束,退出代码0'.

```
Homework_2
E:\Anaconda\ANACONDA\python.exe "E:/Machine Learning/XIGUA/Homework_2.py"
AUC = 0.8696876600102407
进程已结束,退出代码0
```

