PR 曲线

【参考】

- PR曲线和F1、ROC曲线和AUC

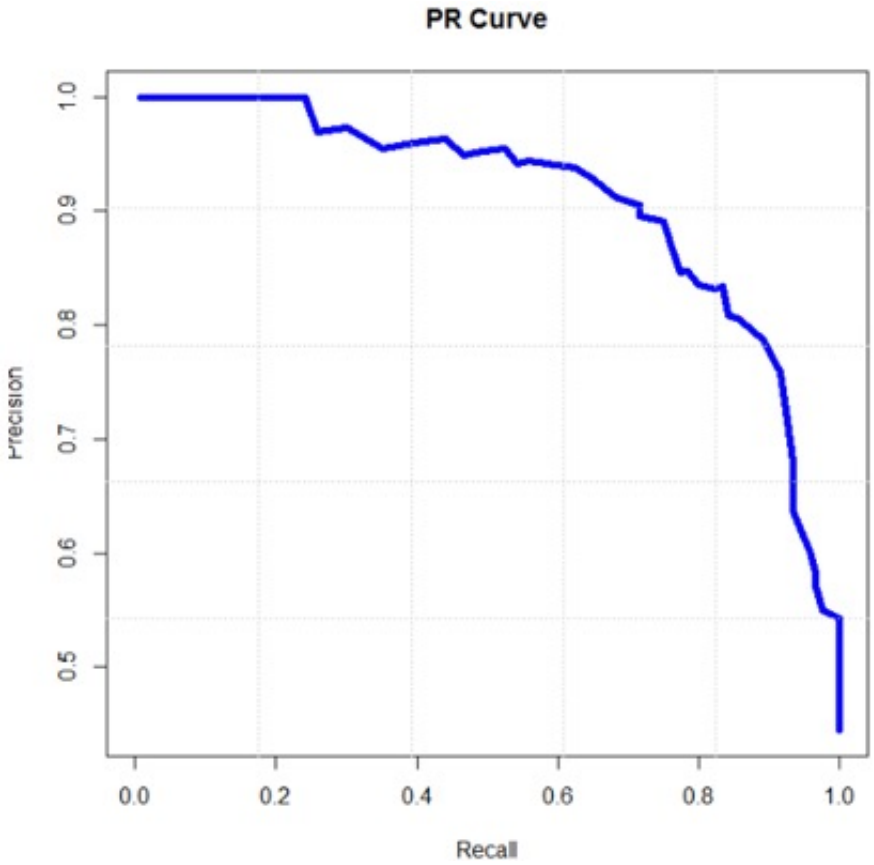
在机器学习中分类器往往输出的不是类别标号，而是属于某个类别的概率值，根据分类器的预测结果从大到小对样例进行排序，逐个把样例加入正例进行预测，算出此时的P、R值。 如下图：

Instance	Class	Score	Instance	Class	Score
1	P	0.9	11	P	0.4
2	P	0.8	12	N	0.39
3	N	0.7	13	P	0.38
4	P	0.6	14	N	0.37
5	P	0.55	15	N	0.36
6	P	0.54	16	N	0.35
7	N	0.53	17	P	0.34
8	N	0.52	18	N	0.33
9	P	0.51	19	P	0.3
10	N	0.505	20	N	0.1

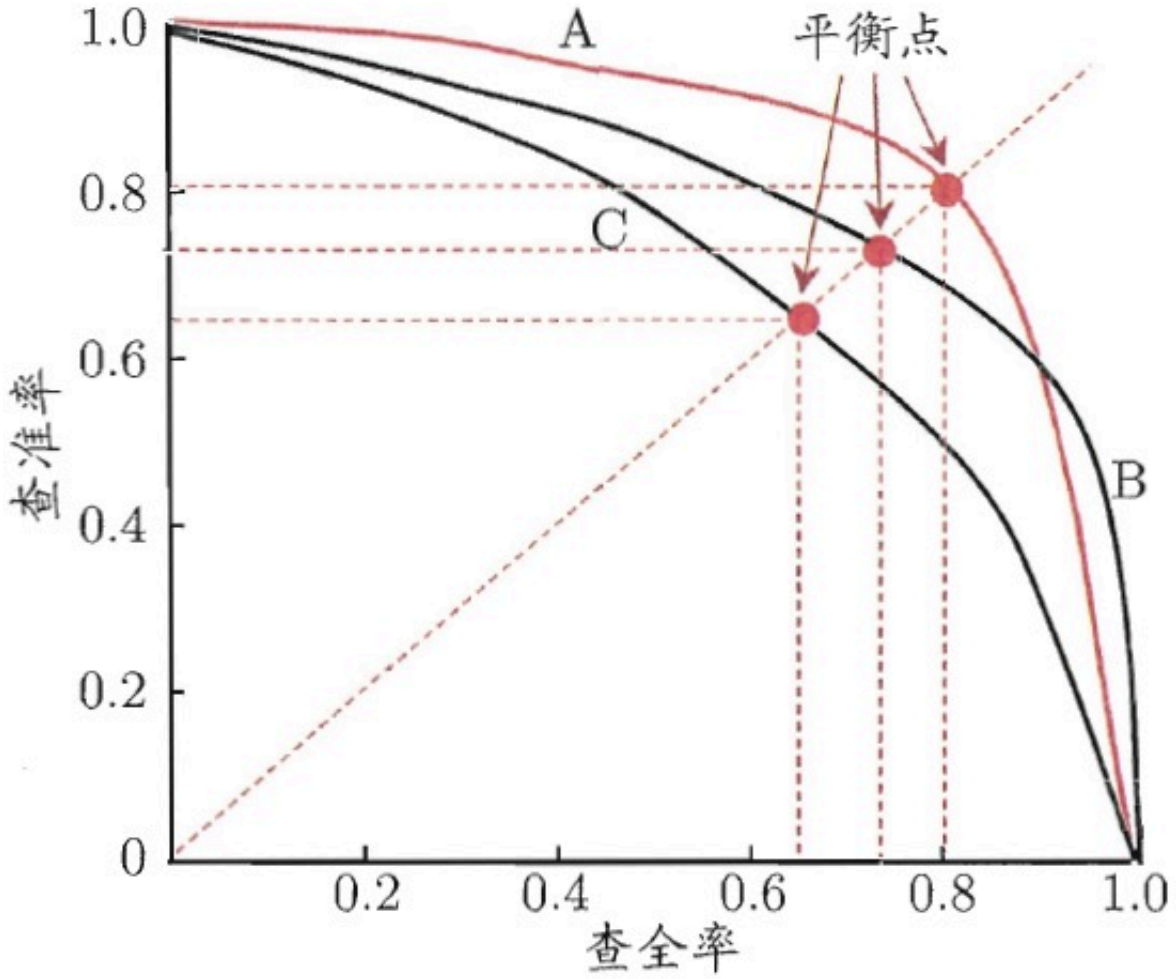
真实情况正例反例各有10个。 先用分数（score）： 0.9作为阈值（大于等于它为正例，小于为反例），此时TP=1，FP=0，FN=9，故P=1，R=0.1。

- 用0.8作为阈值，P=1，R=0.2。
- 用0.7作为阈值，P=0.67，R=0.2。
- 用0.6作为阈值，P=0.75，R=0.3。 依次类推，最后得到一系列P、R值序列，就画出类似如下的P-

R曲线：



P-R曲线越靠近右上角越好。进行比较时，若一个学习器的PR曲线被另一个学习器的PR曲线完全“包住”，我们就可以断言后者的性能优于前者，像下图中学习器A的性能就优于学习器C；若是两个学习器的PR曲线发生交叉，像A和B，就比较难断言孰优孰劣，只能是在具体的查准率或查全率条件进行比较。



但也有很多时候我们仍希望A和B分个高低，这个时候我们会选择比较PR曲线下面积的大小，这是一个比较合理的判据，它在一定程度上表征了学习器在查准率和查全率取得相对“双高”的比例。但这个值又不是那么好算，因此，人们又设计了一些综合考虑查准率、查全率的性能度量：平衡点（Break-Even Point,BEP）。

`平衡点`就是那么一个综合考虑查准率和查全率的性能度量，它是“查准率=查全率”时的取值，如上图中C的BEP就是0.64，而基于BEP进行比较，我们可以认为学习器A要比B好。

ROC 与 AUC

【参考】

- csdn - PR曲线和F1、ROC曲线和AUC
- github.io - 机器学习 第十四章 模型评估

ROC

在上面的操作过程中，我们每次在序列中选择一个`截断点(cut point)`将样本即划分为两部分，排在前面的为正例，排在后面的为反例。

这样子，我们就可以根据不同的任务需求来采用不同的截断点，假设我们更重视查准率，那就可以让截断点靠近序列的前面；若是重视查全率，则可以让截断点排在序列的靠后方。因此，**排序本身的质量好坏，就体现了综合考虑学习器在不同任务下期望泛化性能的好坏**，或者说是“一般情况下”泛化性能的好坏。

ROC曲线就是从这个角度出发来研究**学习器泛化性能**的工具。ROC全称**受试者工作特征（Receiver Operating Characteristic）曲线**，它的绘制过程与PR曲线类似，我们首先根据学习器的预测结果对样本进行排序，然后从前往后逐个将样本作为正例进行预测，每多预测一个样本，就对已预测的所有样本计算两个重要的值。这两个值就是它与PR曲线的区别，这两个值分别是`真正例率（True Positive Rate ,TPR）`，作为ROC曲线的纵轴；`假正例率（False Positive Rate,FPR）`，作为横轴，两者的定义分别为：

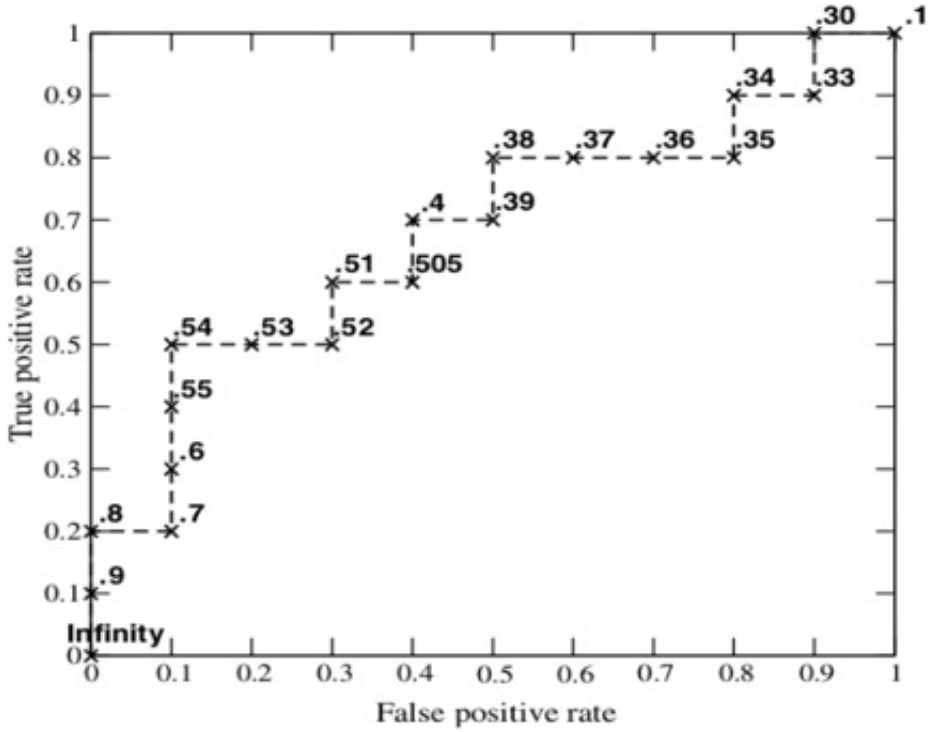
$$TPR = \frac{TP}{TP + FN}$$
$$FPR = \frac{FP}{TN + FP}$$

TPR即预测正确的正例占预测正确的正例和预测错误的反例中比例，FPR即预测错误的正例占预测正确的反例和预测错误的正例的比例。

同样用上面的数据：

- 用0.9作为阈值，此时TP=1，FP=0，FN=9，TN=10，故TPR=0.1，FPR=0。
- 用0.8作为阈值，此时TP=2，FP=0，FN=8，TN=10，故TPR=0.2，FPR=0。
- 用0.7作为阈值，此时TP=2，FP=1，FN=8，TN=9，故TPR=0.2，FPR=0.1。
- 用0.6作为阈值，此时TP=3，FP=1，FN=7，TN=9，故TPR=0.3，FPR=0.1。

依次类推，最后的有限样本ROC曲线如下图：

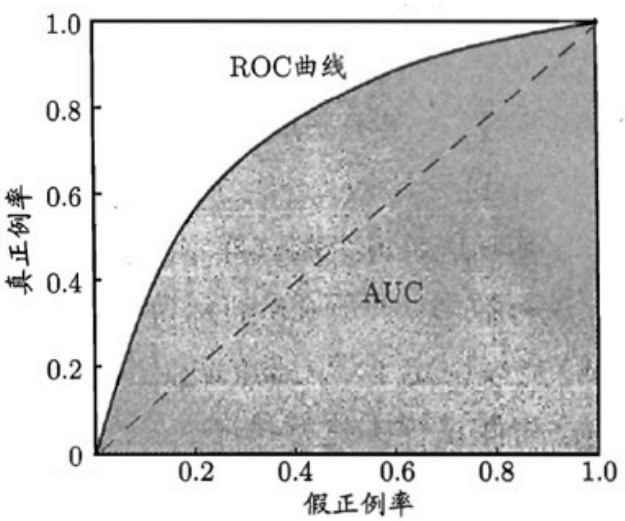


AUC

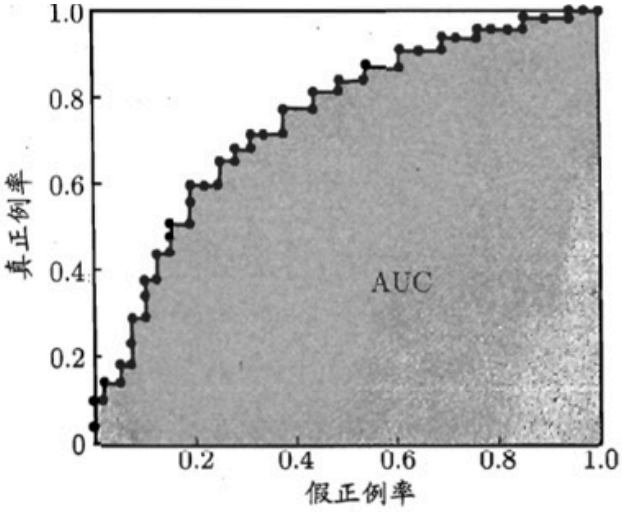
与PR图相同，若一个学习器的ROC曲线被另一个学习器包住，我们就断言后者性能优于前者；若两者发生交叉，则判断ROC曲线下的面积`AUC（Area Under ROC Curve）`.

假设ROC曲线的坐标集合为 $\{(x_1,y_1),\dots,(x_m,y_m)\}$，其中 $(x_1=0,x_m=1)$ 这些点连成曲线，如下图（b），则AUC可估算为：

$$AUC=\frac{1}{2}\sum_{i=1}^{m-1}(x_{i+1}-x_i)\cdot(y_i+y_{i+1})$$



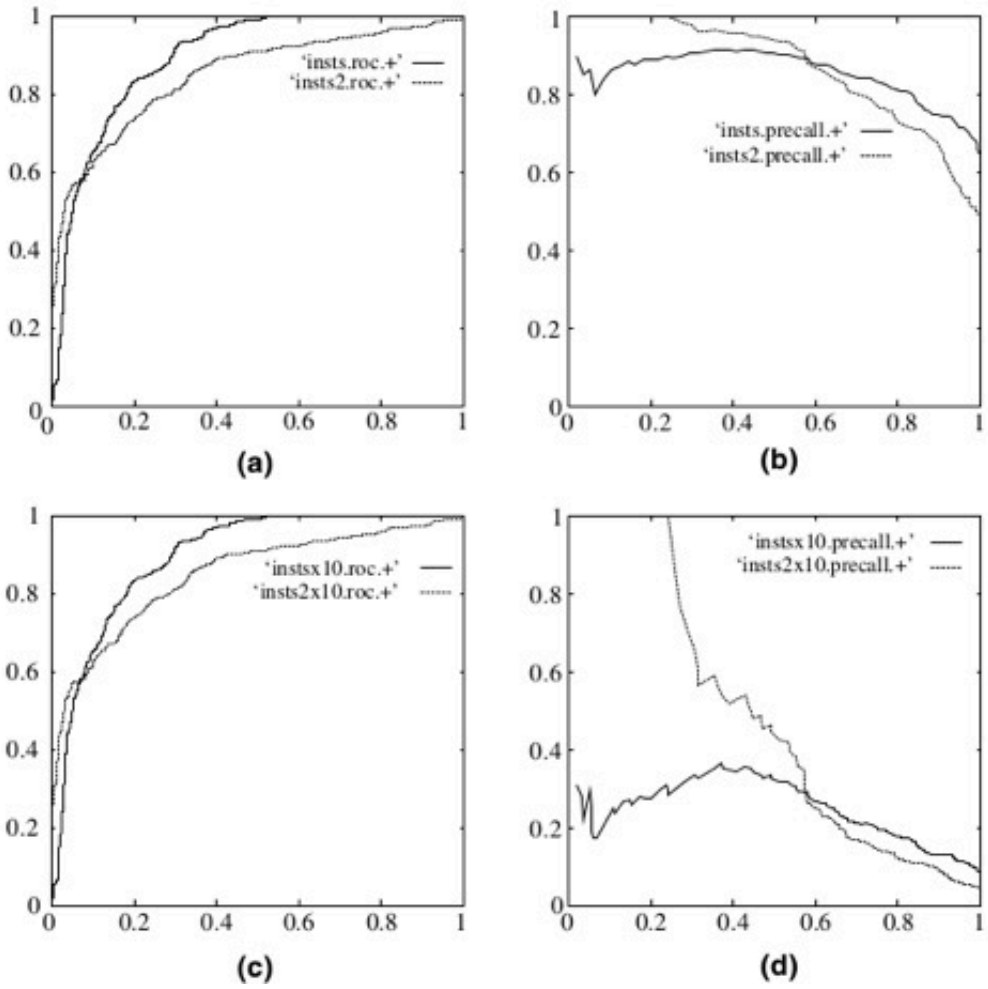
(a) ROC 曲线与 AUC



(b) 基于有限样例绘制的 ROC 曲线与 AUC

PR 曲线与 ROC 比较

从定义上PR曲线的R值是等于ROC曲线中的TPR值，都是用来评价分类器的性能的。 正负样本的分布失衡的时候，ROC曲线保持不变，而PR曲线会产生很大的变化。



- （a）（b）分别是正反例相等的时候的ROC曲线和PR曲线
- （c）（d）分别是十倍反例一倍正例的ROC曲线和PR曲线

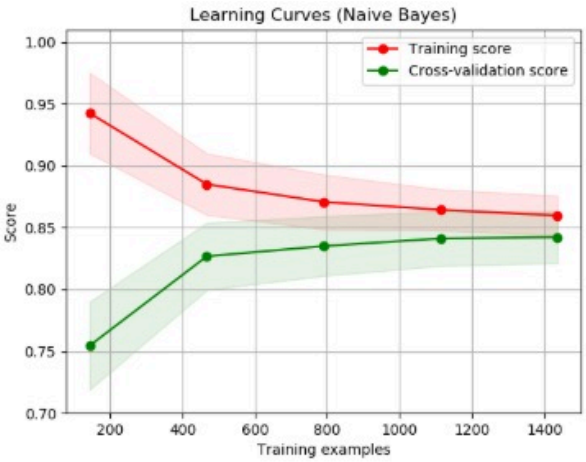
可以看出，在正负失衡的情况下，从ROC曲线看分类器的表现仍然较好（图c），然而从PR曲线来看，分类器就表现的很差。 事实情况是分类器确实表现的不好（分析过程见知乎 qian lv 的回答），是ROC曲线欺骗了我们。

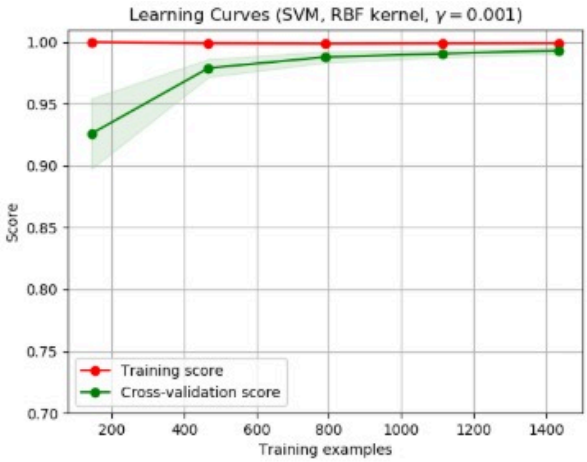
学习曲线

【参考】

- sklearn - Plotting Learning Curves

学习曲线展示了在不同的训练集上训练分与验证分的变化。从下图可以看到，朴素贝叶斯方法两者收敛的分数都很低，有再多的样本也不会改善得分。而带有 RBF 核的 SVM 训练样本越多最终收敛的的得分也越高。


Learning Curves (Naive Bayes)
Training score
Cross-validation score
Score
Training examples
1.00
0.95
0.90
0.85
0.80
0.75
0.70
200
400
600
800
1000
1200
1400


Learning Curves (SVM, RBF kernel, $\gamma = 0.001$)
Training score
Cross-validation score
Score
Training examples
1.00
0.95
0.90
0.85
0.80
0.75
0.70
200
400
600
800
1000
1200
1400