**樊子鹏学习笔记**

第二章主要讲了如何进行模型评估和选择。模型评估与选择分两步：选择合适的学习算法、设定算法参数。

选择学习算法分三步：

1. 划分数据集（训练集和测试集）；
2. 用测试误差作为泛化误差的近似，衡量模型的泛化能力，其评价标准是性能度量；
3. 根据性能度量进行比较检验

**一、划分数据集**

划分数据集有三种方法：留出法、交叉验证法、自助法。

1. **留出法**

将数据集D划分为两个互斥的集合，一个是训练集S，一个是测试集T。

需要注意的是，划分时要保持数据的一致性，就是保持数据类别比例相似。而且单次使用留出法得到的估计结果不稳定可靠，所以要多次进行留出法划分数据，最后取结果的平均值；

1. **交叉验证法**

将数据集D划分为k个大小相似的互斥子集，每次用k-1个子集作为训练集，一个子集作为测试集，进行k此训练和测试，最后取均值作为最终结果。

需要注意的是，划分时要保持数据一致性，还要重复p次划分进行验证；

1. **自助法**

从含有m个样本的数据集D中每次随机抽取一个样本，将其拷贝放进训练集D`中，然后将抽取的样本再放回数据集中，重复m次。这样训练集D`中不可能包含所有的样本，会有36.8%的样本一次都没有被抽中，将D\D`用作测试集。

1. **性能度量**
2. **回归任务**

均误方差：

对于数据分布D和概率密度函数，均误方差为

1. **分类任务**
2. 错误率和精度

错误率是分类错误的样本占样本总数的比例。对样例集D，分类错误率为

。

精度是分类正确的样本数占样本总数的比例。对样例集D，分类错误率为

。

对于数据分布D和概率密度函数，错误率和精度为





1. 查准率、查全率、F1

对于二分类问题，可以将样例根据其真实类别与学习其预测类别的组合划分为真正例（TP）、假正例（FP）、真反例（TN）、假反例（FN），分类结果的混淆矩阵为

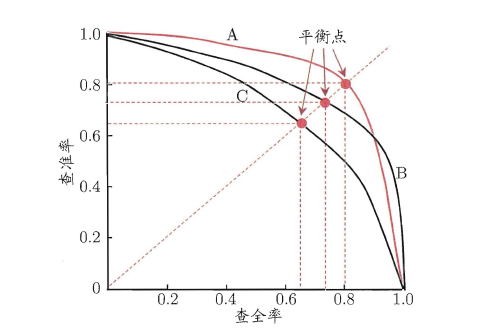
|  |  |  |
| --- | --- | --- |
| 真实情况 | 预测结果 | |
| 正例 | 反例 |
| 正例 | TP | FN |
| 反例 | FP | TN |

查准率P是在预测为正例的结果中有多少是真正的正例，查全率R是所有的正例中有多少被预测出来，它们的公式为





查准率和查全率是矛盾的，查准率高时，一般查全率低，查准率低时，一般查全率要高.在利用查准率查全率来比较学习器时，一般会绘制PR图，PR图越向右上角凸起，说明学习器性能越好



当一条学习器的曲线完全包住了另一条学习器的曲线时，可以说前者的性能要更好。当两条曲线有交点时就不好直接说哪个学习器性能更好了，这时要用F1度量：



当对查准率和查全率的重视程度有所不同时，会用到F1度量的一般形式：



当时查全率影响更大；时为标准F1度量；时查准率影响更大。

当有多个二分类混淆矩阵时，考察查准率查全率有两种方法，一种是先在各混淆矩阵上算出查准率查全率，再计算平均值，得到宏查准率、宏查全率、宏F1。另一种是先对各混淆矩阵上的TP、FP、TN、FN求平均值，再根据平均值算出微查准率、微查全率、微F1；

1. ROC和AUC

ROC曲线也是用来衡量学习器性能的一种曲线，它的纵轴为真正例率TPR，含义是正确预测到的正例占真正正例的比例，横轴为假正例率FPR，它的含义是预测为正例的反例占真正反例的比例。它们的公式为

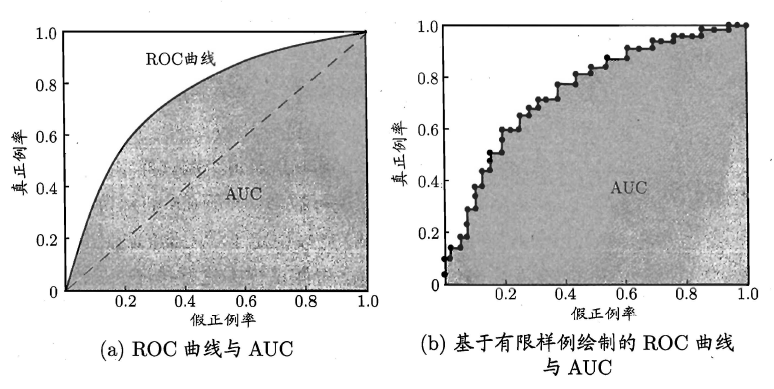




很多的学习器是为样本产生一个实值或概率预测，然后将这个值与一个阈值进行比较，大于阈值的为正例，小于阈值的为反例。

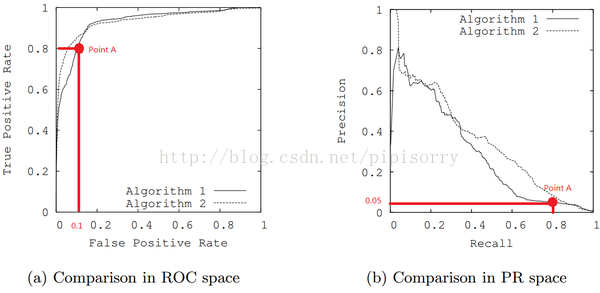
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 样例类型 | 预测结果 |  | 样例类型 | 预测结果 |
| 1 | P | 0.9 | 11 | P | 0.4 |
| 2 | P | 0.8 | 12 | N | 0.39 |
| 3 | N | 0.7 | 13 | P | 0.38 |
| 4 | P | 0.6 | 14 | N | 0.37 |
| 5 | P | 0.55 | 15 | N | 0.36 |
| 6 | P | 0.54 | 16 | N | 0.35 |
| 7 | N | 0.53 | 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 |

假设一个学习器对一组样本进行预测，结果如上图所示，将所有样例按预测结果降序排序，依次将阈值设为各个样例的预测值，比阈值大的看为正例，比阈值小的看为反例，以4为例，它的预测值为0.6，所以1、2、3、4为正例，其余的为反例，然后计算FPR、TPR，将点绘在ROC曲线图中，曲线与横轴围成的面积是AUC，可以根据AUC的面积大小来比较学习器性能。ROC曲线图如下所示



1. ROC与PR

当正反样例的数量相差不大时，ROC与PR趋势差不多。但是当正反样例的数量相差悬殊时，ROC的曲线变化不大，但是PR的曲线会发生大变化。



上图是ROC和PR图，两个图的A点是同一个点，这是因为ROC的TPR和PR的R公式是一样的，所以TPR=0.8和R=0.8是同一个点。ROC图显示的学习器性能很好，因为曲线向左上角凸起很明显，但再PR中，学习器性能并不是很好。那要根据哪个图来判断学习器性能呢？假设样例中有100个正例，通过计算可以得出反例有15200，所以这个数据集是极不平衡的，在预测中，正例有1600个，其中真正例只有80个，显而易见这个学习器性能并不好，可以得出结论，当样例类型分布不平衡时，正样本数非常少时，要用PR来衡量学习器的好坏；

1. 代价敏感错误率与代价曲线

现实任务中不同类型的错误造成的后果会有所不同，单纯的追求错误数最小化不能满足要求，要最小化总体代价。代价敏感错误率为



时将第i类样本预测为第j类样本的代价，分别表示样例集D的正例子集和反例子集。

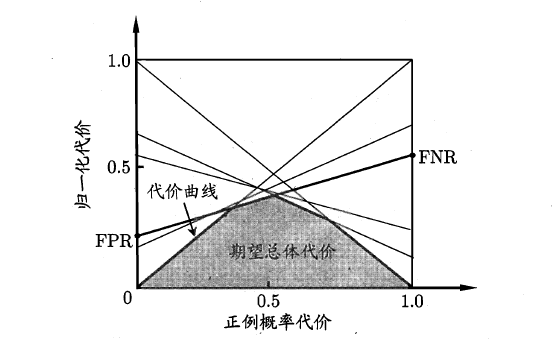
在这种非均等代价下，ROC曲线不能直接反应出学习器的期望总体代价，需要画出代价曲线，其横轴是正例概率代价

（p是样例为正例的概率）

纵轴是归一化代价



ROC图中的每一个点对应了代价曲线上的一条线段，所有的曲线和横轴围成的面积即是在所有条件下学习器的期望总体代价



1. **比较检验**

对学习器的性能度量进行比较，不是单纯的比数值大小，统计假设检验为比较学习器性能提供了重要依据。

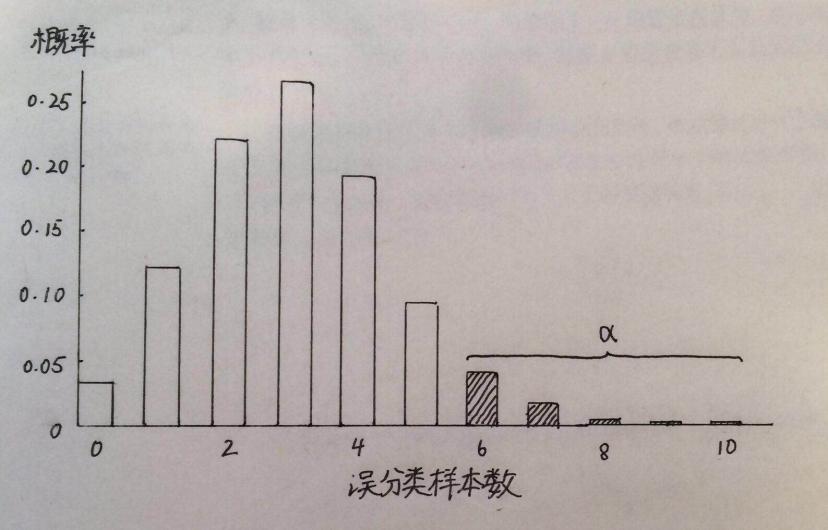
1. **单个学习器泛化性能的假设检验**

根据测试错误率估推出泛化错误率。

假设有个测试样本，其中有个被误分类，将个样本误分类的概率为



解得出在时取得最大值，增大时，减小，这符合二项分布。如图，时，10个样本中测得3个被误分类的概率最大



1. 二项检验（做一次留出法估计，得出一个测试错误率）

对这样的假设进行检验，在的概率内所能观测到的最大错误率如下式计算

计算出后，与测试错误率进行比较，若，则得出结论：在的显著度下，假设不能被拒绝，即能以的置信度认为，学习起的泛化错误率不大于。

置信度是区间估计的可靠性。显著度是原假设正确，人们却把它拒绝了的概率或风险。

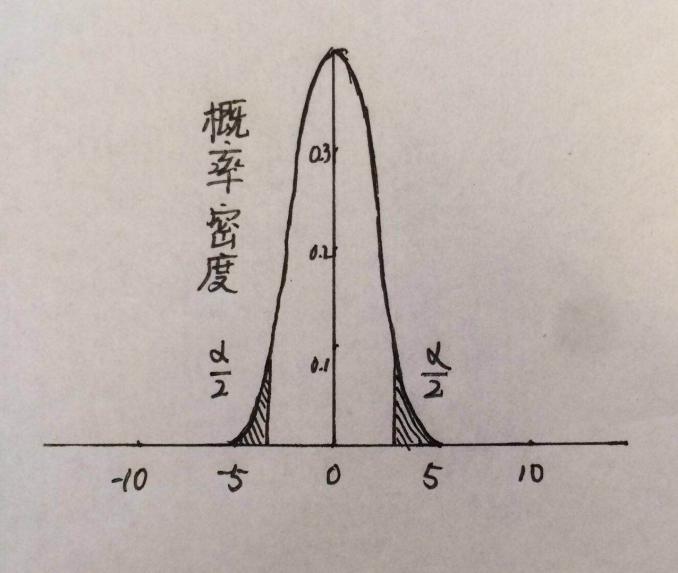
1. 双边t检验（多次留出法或交叉验证法，得出多个测试错误率）

假设得到了个测试错误率，，...，，计算平均测试错误率和方差





这个测试错误率可看作泛化错误率的独立采样，则变量服从自由度为的分布



假设和显著度，可以计算出当测试错误率均值为时，在概率内能观测到的最大错误率，即临界值（可以根据双边t检验常用临界值表得出）。若平均错误率与与之差位于临界值范围内，则不能拒绝假设，即可认为泛化错误率为

1. **两个学习器泛化性能的假设检验**
2. 交叉验证t检验

对两个学习器A和B同时使用k折交叉验证，得出k对测试错误率，对每对测试错误率求差，如果两个学习器的性能相同，则差值为零。根据差值作假设“学习器A与B性能相同”，计算出均值方差，在显著度下，若变量小于临界值，则假设不能被拒绝，否则可以认为两个学习器性能有显著差别，且平均错误率较小的那个学习器性能较优。

通常样本有限，使用交叉验证法时，不同轮次训练集会有一定程度的重叠，使测试错误率并不独立。解决这一问题，可以采用5×2交叉验证法。

做五次两折交叉验证，每次2折交叉验证之前随机打乱数据，对每次2折交叉验证产生的测试错误率进行求差，仅计算第一次2折交叉验证的两个结果的平均值，对每次2折交叉验证的结果都计算出方差变量服从自由度为5的t分布，双边检验临界值为当=0.05时为2.5706，当=0.1时为2.0150。然后比较方法同上。

1. McNemar检验

假设“两个学习器性能相同”，那么两个学习器测试正确的样本数相同，而且测试错误的样本数也相同，绘制两学习器分类差别列联表如下

|  |  |  |
| --- | --- | --- |
| 算法B | 算法A | |
| 正确 | 错误 |
| 正确 | e00 | e01 |
| 错误 | e10 | e11 |

其中e01=e10，变量服从自由度为1的分布。给定显著度，当小于临界值时，不能拒绝假设，否则拒绝假设，即两者性能有明显差别，且平均错误率较小的那个学习器性能较优。自由度为1的检验的临界值当=0.05时为3.8415，=0.1时为2.7055。

1. **多个学习器泛化性能的假设检验**

当有多个学习器需要比较时，要用到Friedman检验，当假设被拒绝时，需要进行Nemenyi后续检验。

假定有D1、D2、D3、D4四个数据集对算法A、B、C进行比较。

首先，用留出法或交叉验证法将三个算法在不同的数据集上进行测试，将结果由好到坏排序，并赋值1、2...，如果有几个算法的测试结果一样，则平分序值。然后计算每个算法的平均序值。以下是算法比较序值表

|  |  |  |  |
| --- | --- | --- | --- |
| 数据集 | 算法A | 算法B | 算法C |
| D1 | 1 | 2 | 3 |
| D2 | 1 | 2.5 | 2.5 |
| D3 | 1 | 2 | 3 |
| D4 | 1 | 2 | 3 |
| 平均序值 | 1 | 2.125 | 2.875 |

然后，假设“所有的算法的性能相同”，令表示第i个算法的平均序值，变量

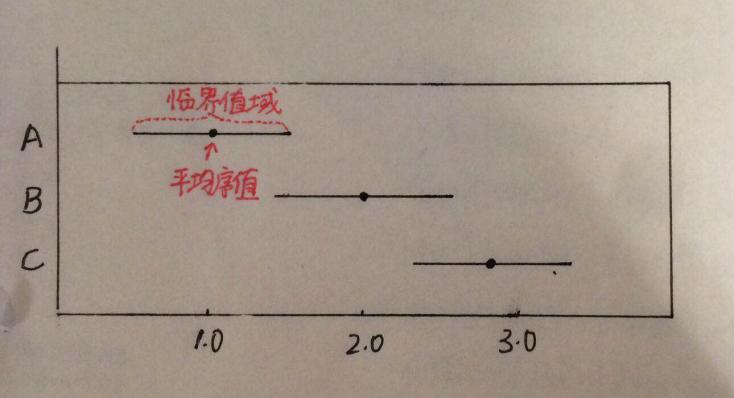




根据F检验常用临界值表来比较，若假设被拒绝，说明各算法的性能显著不同，需要后续检验。

Nemenyi后续检验计算出平均序值差别的临界值域，的值可以根据常用值表得出。若两个算法的平均序值之差超出了临界值域，则两个算法的性能不同。

可以根据Friedman检查图直观的看出算法的性质是否相同，若两个算法的图像有重叠的部分，则两个算法性能相同，没有重叠的地方，则两个算法的性能不同。



1. **调参数**

学习器有两类参数，一是算法的参数，即超参数，一是模型的参数。

1. 超参数是在开始学习过程之前设置值的参数，不能从数据估计得到，使用经验法则来探寻其最优值，需要手动设置。
2. 参数是模型内部的配置变量，可以用数据估计它的值。不需要手动配置，可以定义模型功能。
3. **偏差与方差**

偏差-方差分解是解释算法泛化性能的一种重要工具，要了解为什么学习器具有这样的泛化能力，就要用到偏差-方差分解。

偏差：度量学习算法的期望预测与真实结果的偏离程度，刻画学习算法本身的拟合能力；

方差：度量同样大小的训练集的变动所导致的学习性能的变化，刻画数据扰动所造成的影响；

噪声：表达当前任务上任何学习算法所能达到的期望泛化误差的下界，刻画学习问题本身的难度。

因为泛化误差=偏差+方差+噪声，所以为了取得更好的泛化能力，要降低方差和误差。但是方差和误差相矛盾，当训练不足时，数据扰动不足使学习器发生显著变化，此时偏差主导泛化错误率，当训练过足时，稍微的数据扰动都会导致学习器发生显著变化，这时方差主导泛化错误率，发生过拟合。