机器学习

2024 Summer

yingziyu-llt

最初写作于：**2024 年 08 月 01 日**

最后更新于：**2024 年 08 月 23 日**

目录

章节 1. 模型评估和选择

1.1. 衡量模型的好坏

我们用错误率、精度、误差等来衡量模型预测的好坏。衡量方法定义如下

Def 衡量模型预测的方法 **定义 1.1.1**

- 错误率:$E = \frac{a}{m}$,即错误样本数在总样本数中的占比
- 精度: 精度=1-错误率
- 误差: 实际预测输出和真实结果直接的差异
- 训练误差/经验误差: 在训练集上的误差
- 泛化误差: 在测试集上的误差

我们一般来说，希望模型能够在新样本上表现的很好，能找到样本间的潜在规律。当训练的过于好了的时候，可能模型会将样本的个别规律当成共同潜在规律，在遇到新样本的时候表现就会不好。这种情况叫做**过拟合**(overfitting),其对面为**欠拟合**(underfitting)。

1.2. 模型的评估

为了测试模型性能,我们要合理划分数据集，将其变成互斥的训练集和测试集。关于分割，也有很多种方法，下面介绍几种。

1.2.1. 留出法

对数据进行随机划分，可以将数据划分为两个部分。但是，随机划分可能会改变数据分布情况，因此我们要对数据集进行分层随机抽样，保证正例和反例接近 1:1。同时，单次划分可能也会对模型性能的衡量带来误差，所以要做多组实验取平均值。

1.2.2. 交叉验证法

将数据集划分为多个大小接近的子集，每次选择一个子集作为测试数据，其他作为训练数据，这样就可以做多次实验。这种方法一般叫做 k-fold cross validation。

如果分成的份数和数据集大小一致，那么称作留一法(Leave-One-Out)。这种方法的好处在于准确度不和划分方法挂钩，使得这种方法准确率很高，但是计算量很大，在数据量很大的情况下不适用。

1.2.3. 自助法

使用有放回的取样，反复多次取样，将取出的元素放入集合D'，最终D'中将有很多元素重复，其覆盖面应当为$1 - \frac{1}{e}$。将其作为训练集，D/D'为测试集。 这个方法特别适合在数据集很小，难以划分的时候使用，同时由于其可以产生很多种数据集，适合在集成学习中使用。

1.3. 性能衡量

对于分类问题，我们最常用的衡量模型性能的方法是错误率$E(f;D)=\frac{1}{m}\sum_{\{i=1\}}^{m}II(y_i\neq f(x_i))$，即错误率。对于回归问题，我们最常用的衡量模型性能的方法是$E(f;D)=\frac{1}{m}\sum_{\{i=1\}}^{m}(y_i-f(x_i))^2)$，即均方误差(mean squared error/MSE)。

然而，对于很多情况下，我们更加关心错误到底是怎么的错误，是将正确的识别成错误的还是将错误的识别成正确的。这时就需要用混淆矩阵来详细衡量。

1.3.1. P-R 曲线

Def 查准率和查全率 **定义 1.3.1**

对于一个二分类问题，设一类是有，一类是无。那么查准率(precision)就是在认为是有的部分中多少是真的有，查全率(recall)就是在真的有的部分里面有多少识别为有。

	真	假
真	TP(真正例)	FN(假反例)
假	FP(假正例)	TN(真反例)

表 1.1 混淆矩阵，列为判定，行为真实

那么查准率

$$P=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{1.1}$$

，查全率

$$R=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{1.2}$$

将查全率和查准率作为横、纵坐标绘图，形成 P-R 曲线。如果一个模型可以完全包住另一个，那么这个一定比另一个优。如果有交点，那么就不能这么说了。

对于有交点的两个曲线，有很多种衡量方法。第一种是看两个的线下面积，面积大的就好。然而，面积很难计算，所以要换一种量度。

最简单的一种是计算平衡点(BEP)即 P=R 的那个点。这个点越大，可以认为这个模型更好。

还有相对比较复杂的一种量度是$F1$度量，即 P 和 R 的调和平均值。$\mathrm{F1}=2*P*\frac{R}{P+R}=2*\frac{\mathrm{TP}}{\text{样例总数}+\mathrm{TP}-\mathrm{TN}}$

而对于查错和查全两个后果不同的情况，我们需要修正这个公式，转化为加权的调和平均值$\frac{1}{F_\beta}=\frac{1}{1+\beta^2}*\left(\frac{1}{P}+\frac{\beta}{R}\right)$,$F_\beta=(1+\beta^2)*P*\frac{R}{\beta^2*P+R}$，对于$\beta$，越大越倾向于查全。

对于在多次训练中获得多组测试结果，得到多组混淆矩阵，那么怎么求 P 和 R 呢?

我们有 marco-P 和 marco-R 度量法，就是分别求出各组的 P 和 R，然后求平均值。还有 micro-P 和 micro-R 度量法，就是将各组的 TP,TN,FP,FN 分别求平均，用均值算出 P 和 R。

1.3.2. ROC 曲线和 AUC

考虑到实际使用中更常见的是返回一个[0,1]的值，如果大于某个阈值就为 1,否则为 0. 那么分类过程就是对每个例子的的可能性排序，最终找到某个分点，使得小于该分点的都判断为负，而大于该分点的都判断为正。对于不同的实际任务，相当重要的一点是要衡量这个排序的质量。那么，更加重要的一点是要去衡量这个排序的质量。

为了衡量这个指标，我们画出 ROC 曲线。其横轴为假正例率，纵轴为真正例率。对于理想情况，我们可以求出每个点，然后描点画图。 但在实际中，我们只能做有限个实验，不能够描出所有点。做一组数据并进行如下操作：

1. 估算出每个数据是真的置信度
2. 用置信度从小到大排序
3. 先将阈值设为 0,描点(0,0)
4. 逐步提升阈值，使得阈值触碰每个预测的置信度。
5. 设前一个描点为(x,y)，如果碰上的是正例，那么描点$\left(x, y+\frac{1}{m^{+}}\right)$，否则描点$\left(x+\frac{1}{m^{-}}, y\right)$

线下的面积被称为 AUC,是一个很好的度量方法。当 AUC=1 时，说明存在一个阈值使得该分类器是一个理想分类器，当$0.5 < \mathrm{AUC} < 1$时，说明它是一个可用的分类器，其越大证明效果越好。当 $\mathrm{AUC} = 0.5$时，说明其和随机决定没有区别，是没用的。当其小于 0.5 时，说明其劣于随机分类，将其分类结果反过来也是可用的。

1.4. 统计检验

看不懂，跳了

1.5. 偏差与方差

我们先假设$f(x;D)$是在训练集D学到的预测输出，y是其真实标记，y_D是受到噪声干扰后的标记。

我们设其预测的期望$\overline{f}(x) = \mathbb{E}_D[f(x;D)]$

用不同训练集训练产生的方差为 $\mathrm{var}(x) = \mathbb{E}_D\left[\left(f(x;D) - \overline{f}(x)\right)^2\right]$，噪声为 $\varepsilon^2 = \mathbb{E}_D\left[(y - y_D)^2\right]$,期望数据和真实数据的偏差$\mathrm{bias}^2 = \left(y - \overline{f}(x)\right)^2$

假定噪声的期望为 0,那么 $E(f;D) = \mathbb{E}_D\left[(f(x;D) - y)^2\right]$ MSE LOSS $= \ldots = \mathrm{bias}^2(x) + \mathrm{var}(x) + \varepsilon^2$

这就是偏差-方差分解。bias 衡量了模型的拟合能力，而 var 衡量了模型对不同训练数据的敏感性，ε衡量了问题本身的难度。

章节 2. 线性模型

2.1. 基本形式

实际上就是找到一种对于各个属性的线性组合，使得其最接近真实的 lable。

整体的形式是$f(x) = w^T x + b$. w,b 两者学习到之后，模型就被确定。通过 w，我们可以看出每个元素的重要性，从而有一定的可解释性。

2.2. 线性回归

对于各种属性，怎么将其转换为数字呢？

1. 数字元素 直接使用就可以
2. 有序的属性 如高、中、低，可以分别用 1,0.5,0 来表示
3. 无序的属性 如红、黄、蓝，进行 one-hot 编码

如何确定两个参数呢？这取决于如何定义最接近。最常见的做法是用最小二乘法，即使得预测结果和目标的欧式距离最短。即$(w^*, b^*) = \arg\min_{\{a,b\}} \left(w^T x + b - y\right)^2$

对于更加一般的数据集 D,我们要试图学到$f(x_i) = w^T x_i + b$，使得均方误差最小。为了便于处理，我们让$\hat{w} = (w; b)$,将数据集表示为一个$(m + 1) \times b$的矩阵X. 对于每一个数据x_i,我们让X的第 i 行为$(x_i^T 1)$,于是有$\hat{w} = \arg\min_{\{w\}} (y - Xw)^{T(y-Xw)}$,对$w$求偏导，得$\frac{\partial(E)}{\partial(w)} = 2X^{T(Xw-y)}$

如果$2X^T X$是正定的/满秩的，那么便有一个 close form 的解。

实际上大概率是非满秩的。所以我们常引入一个正则项(regularization)。

如果要拟合的函数不是一个线性的怎么办呢？假设是f性质的函数，那么令$f^-1(y) = w^t x + b$，先处理 y，然后拟合就可以了。

2.3. 对数几率回归

对于一个分类问题，我们更需要知道是概率如何，应当是一个$[0, 1]$的数。为了将线性回归出来的数映射到$[0, 1]$上，我们有很多种方法。

最为显然的一个方法是，建立分段函数

$$f(x) = \begin{cases} 1x > 0 \\ 0.5x = 0 \\ 0x < 0 \end{cases} \tag{2.1}$$

显然可以描述。

但显然这个玩意不可逆，于是改变方法，用 Sigmoid 函数($\frac{1}{1+e^-x}$)来映射。

这玩意 loss 推导那块很无聊，反正最后得出来

$$l(\beta) = \sum_{i=1}^{m} (-y_i \beta^T x_i + \log\left(1 + e^{\beta^T x_i}\right) \tag{2.2}$$

是个凸函数，可以用凸优化的常规方法求出最小值。

2.4. 线性判别分析

整体的思路就是将一堆点映射到某个直线上，让类内点距离最近的同时类间点距离最远。

参考文献