Tracholar的博客 关于我 wiki 智子

深入理解AUC

Jan 26, 2018

在机器学习的评估指标中，AUC是一个最常见也是最常用的指标之一。 AUC本身的定义是基于几何的，但是其意义十分重要，应用十分广泛。 本文作者深入理解AUC，并总结于下。

AUC是什么

在统计和机器学习中，常常用AUC来评估二分类模型的性能。AUC的全称是 area under the curve，即曲线下的面积。 通常这里的曲线指的是受试者操作曲线(Receiver operating characteristic, ROC)。 相比于准确率、召回率、F1值等依赖于判决阈值的评估指标，AUC则没有这个问题。

ROC曲线早在第二次世界大战期间就被使用在电子工程和雷达工程当中，被用于军事目标检测。 后来，ROC曲线也被应用到心理学、医学、机器学习和数据挖掘等领域的模型性能评估。

对于二分类问题，预测模型会对每一个样本预测一个得分s或者一个概率p。 然后，可以选取一个阈值t，让得分s>t的样本预测为正，而得分s<t的样本预测为负。 这样一来，根据预测的结果和实际的标签可以把样本分为4类：

	正样本	**负样本**
预测为正	TP(真正例)	FP(假正例)
预测为负	FN(假负例)	TN(真负例)

随着阈值t选取的不同，这四类样本的比例各不相同。定义真正例率TPR和假正例率FPR为：

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

对于真正例率TPR，分子是得分>t里面正样本的数目，分母是总的正样本数目。 而对于假正例率FPR，分子是得分>t里面负样本的数目，分母是总的负样本数目。 因此，如果定义$N_+(t), N_-(t)$分别为得分大于t的样本中正负样本数目，N_+, N_-为总的正负样本数目，那么TPR和FPR可以表达为阈值t的函数

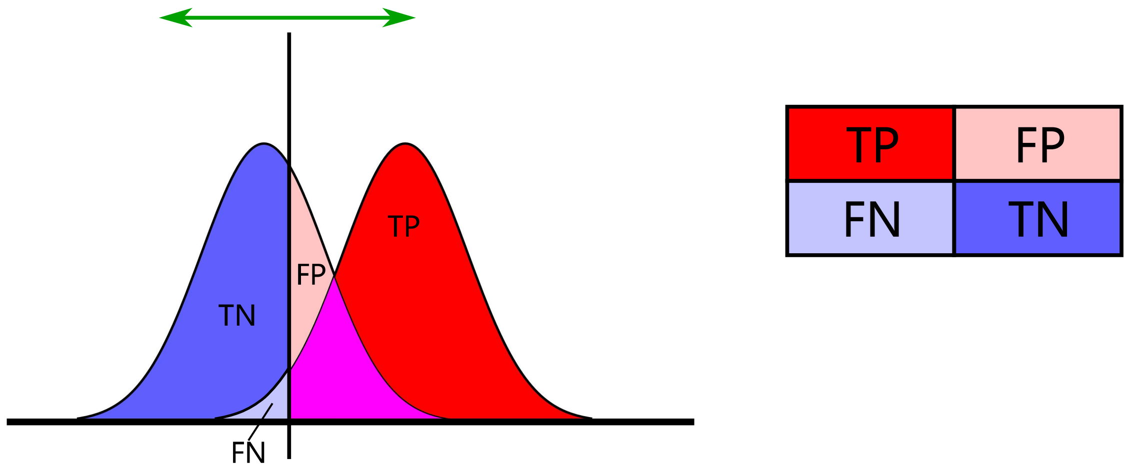
$$\mathrm{TPR}(t)=\frac{N_+(t)}{N_+}$$
$$\mathrm{FPR}(t)=\frac{N_-(t)}{N_-}$$

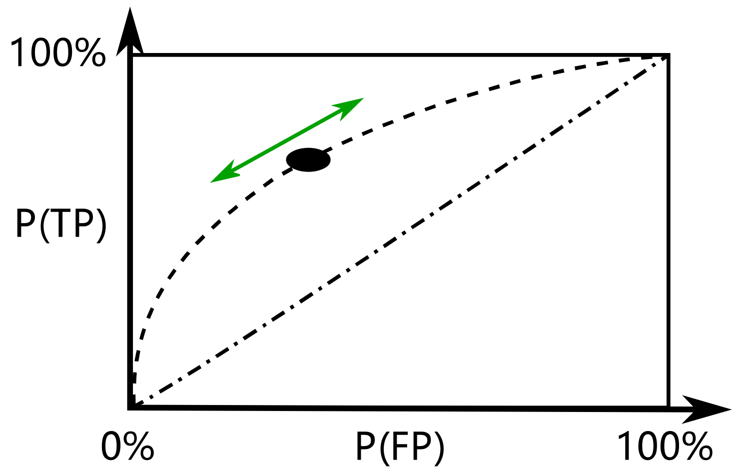
随着阈值t的变化，TPR和FPR在坐标图上形成一条曲线，这条曲线就是ROC曲线。 显然，如果模型是随机的，模型得分对正负样本没有区分性，那么得分大于t的样本中，正负样本比例和总体的正负样本比例应该基本一致。 也就是说

$$\frac{N_+(t)}{N_-(t)}=\frac{N_+}{N_-}$$

结合上面的式子可知TPR和FPR相等，对应的ROC曲线是一条直线！

反之，如果模型的区分性非常理想，也就是说正负样本的得分可以完全分开，所有的正样本都比负样本得分高，此时ROC曲线表现为「 字形。 因为正例得分都比负例搞，所以要么TPR=0要么FPR=0！





实际的模型的ROC曲线则是一条上凸的曲线，介于随机和理想的ROC曲线之间。而ROC曲线下的面积，即为AUC！

$$\mathrm{AUC} = \int_{t=\infty}^{-\infty} y(t)dx(t)$$

这里的x和y分别对应TPR和FPR，也是ROC曲线的横纵坐标。

AUC的概率解释

概率解释的证明

AUC常常被用来作为模型排序好坏的指标，原因在于AUC可以看做随机从正负样本中选取一对正负样本，其中正样本的得分大于负样本的概率！ 这个结论很容易证明，考虑随机取得这对正负样本中，负样本得分在$[t, t+\Delta t]$之间的概率为

$$\begin{aligned}
& P(t \leq s_- < t + \Delta t) \\
=& P(s_- > t) - P(s_- > t + \Delta t) \\
=& \frac{N_-(t) - N_-(t+\Delta t)}{N_-} \\
=& x(t) - x(t+\Delta t) = -\Delta x(t)
\end{aligned}$$

如果Δt很小，那么该正样本得分大于该负样本的概率为

$$
\begin{gathered}
P(s_+ > s_- | t \le s_- < t + \Delta t) \\
\approx P(s_+ > t) = \frac{N_+(t)}{N_+} = y(t)
\end{gathered}
$$

所以，

$$
\begin{aligned}
& P(s_+ > s_-) \\
=& \sum P(t \le s_- < t + \Delta t) P(s_+ > s_- | t \le s_- < t + \Delta t) \\
=& -\sum y(t) \Delta x(t) \\
=& -\int_{t=-\infty}^{\infty} y(t) dx(t) \\
=& \int_{t=\infty}^{-\infty} y(t) dx(t)
\end{aligned}
$$

注意积分区间，$t = -\infty$对应ROC图像最右上角的点，而$t = \infty$对应ROC图像最左下角的点。所以，计算面积是$\int_{t=\infty}^{-\infty}$。可以看出，积分项里面实际上是这样一个事件的概率：**随机取一对正负样本，负样本得分为t且正样本大于t！** 因此，对这个概率微元积分就可以到正样本得分大于负样本的概率！

AUC的排序特性

根据上述概率解释，AUC实际上在说一个模型把正样本排在负样本前面的概率！ 所以，AUC常用在排序场景的模型评估，比如搜索和推荐等场景！ 这个解释还表明，如果将所有的样本的得分都加上一个额外的常数，并不改变这个概率，因此AUC不变！ 因此，在广告等需要绝对的点击率场景下，AUC并不适合作为评估指标，而是用logloss等指标。

AUC对正负样本比例不敏感

利用概率解释，还可以得到AUC另外一个性质，对正负样本比例不敏感。 在训练模型的时候，如果正负比例差异比较大，例如正负比例为1:1000，训练模型的时候通常要对负样本进行下采样。当一个模型训练完了之后，用负样本下采样后的测试集计算出来的AUC和未采样的测试集计算的AUC基本一致，或者说前者是后者的无偏估计！ 如果采样是随机的，对于给定的正样本，假定得分为s_+，那么得分小于s_+的负样本比例不会因为采样而改变！ 例如，假设采样前负样本里面得分小于s_+的样本占比为70%，如果采样是均匀的，即$> s_+$的负样本和$< s_+$的负样本留下的概率是相同的，那么显然采样后这个比例仍然是70%！ 这表明，该正样本得分大于选取的负样本的概率不会因为采样而改变，也就是$y(t)dx(t)$是不变的，因此，AUC也不变！

相比于其他评估指标，例如准确率、召回率和F1值，负样本下采样相当于只将一部分真实的负例排除掉了，然而模型并不能准确地识别出这些负例，所以用下采样后的样本来评估会高估

准确率；因为采样只对负样本采样，正样本都在，所以采样对召回率并没什么影响。这两者结合起来，最终导致高估F1值！

AUC的计算

AUC可以直接根据ROC曲线，利用梯形积分进行计算。此外，还有一个比较有意思的是，可以 利用AUC与Wilcoxon-Mann-Whitney测试的U统计量的关系，来计算AUC。这可以从AUC的概率意义推导而来。

假设我们将测试集的正负样本按照模型预测得分 **从小到大** 排序，对于第j个正样本，假设它的排序为 r_j， 那么说明排在这个正样本前面的总样本有 r_j-1个，其中正样本有 $j-1$个（因为这个正样本在所有的正样本里面排第j）， 所以排在第j个正样本前面(得分比它小)的负样本个数为 r_j-j个。也就是说，对于第j个正样本来说，其得分比随机取的一个负样本大(排序比它靠后)的概率是 $(r_j-j)/N_-$,其中N_-是总的负样本数目。所以，平均下来，随机取的正样本得分比负样本大的概率为

$$\frac{1}{N_+}\sum_{j=1}^{N_+}(r_j-j)/N_- = \frac{\sum_{j=1}^{N_+} r_j - N_+(N_++1)/2}{N_+N_-}$$

所以

$$AUC = \frac{\sum_{j=1}^{N_+} r_j - N_+(N_++1)/2}{N_+N_-}$$

因此，很容易写出计算AUC的SQL代码

```
select
        (ry - 0.5*n1*(n1+1))/n0/n1 as auc
from(
        select
                sum(if(y=0, 1, 0)) as n0,
                sum(if(y=1, 1, 0)) as n1,
                sum(if(y=1, r, 0)) as ry
        from(
                select y, row_number() over(order by score asc) as r
                from(
                        select y, score
                        from some.table
                )A
        )B
)C
```

AUC的优化

采用极大似然估计对应的损失函数是logloss，因此极大似然估计的优化目标并不是AUC。 在一些排序场景下，AUC比logloss更贴近目标，因此直接优化AUC可以达到比极大似然估计更好的效果。 实际上，pairwise的目标函数就可以看做一种对AUC的近似。因为损失函数都是作用与正负样本得分差之上！ 例如，

rank-SVM	$\max(0, -s_+ + s_- + \Delta)$
rank-net	$\log(1+\exp(-(s_+ - s_-)))$
指数损失	$\exp(-(s_+ - s_-))$
TOP 损失	$\sum_s \max(0, -s_c + s + \Delta)$

显然，这些损失函数都是对$s_+ < s_-$的正负样本对进行惩罚！ 此外，也有一些其它对AUC近似度更好的损失函数，例如

$$
\begin{aligned}
&\mathbf{E}\left[(1 - w^T(s_+ - s_-))^2\right] \\
&= \frac{1}{n_+ n_-} \sum_{i=1}^{n_+} \sum_{j=1}^{n_-} (1 - w^T(s_i^+ - s_j^-))^2
\end{aligned}
$$

s_i^+, s_j^-分别表示正例和负例的得分。 这解释了为什么某些问题中，利用排序损失函数比logloss效果更好，**因为在这些问题中排序比概率更重要**！

AUC要到多少才算好的模型

AUC越大表示模型区分正例和负例的能力越强，那么AUC要达到多少才表示模型拟合的比较好呢？在实际建模中发现，预测点击的模型比预测下单的模型AUC要低很多，在月活用户里面预测下单和日活用户里面预测下单的AUC差异也很明显，预测用户未来1小时下单和预测未来1天的下单模型AUC差异也很大。这表明，AUC非常依赖于具体任务。

以预测点击和预测下单为例，下单通常决策成本比点击高很多，这使得点击行为比下单显得更加随意，也更加难以预测，所以导致点击率模型的AUC通常比下单率模型低很多。

那么月活用户和日活用户那个更容易区分下单与不下单用户呢？显然月活用户要容易一些，因为里面包含很多最近不活跃的用户，所以前者的AUC通常要高一些。

对于预测1小时和预测1天的模型，哪一个更加困难？因为时间越长，用户可能发生的意料之外的事情越多，也越难预测。举个极端的例子，预测用户下一秒中内会干啥，直接预测他会做正在干的事情即可，这个模型的准确率就会很高，但是预测长期会干啥就很困难了。所以对于这两个模型，后者更加困难，所以AUC也越低。

Like Issue Page

Error: API rate limit exceeded for 101.198.192.11. (But here's the good news: Authenticated requests get a higher rate limit. Check out the documentation for more details.)

Write Preview Login with GitHub

Leave a comment

Styling with Markdown is supported Comment

Powered by Gitment

tracholar
tracholar

记录每天的心情和收获，不积跬步无以至千里!