第三章 评估方法

一、为什么要评估

- 分类或预测有多种方法，对于每种方法有许多选择
- 为了选择出来最佳的模型，需要对每个模型进行评估
- 当数据集中有不均衡的分布例如偏态分布时候，用混淆矩阵评估模型相比于基础的准确率矩阵更有效
- 基于上述情况，我们也许能够接受一定的偏差，来换取在重要类别的更好的准确性

二、混淆矩阵

1. **混淆矩阵的概念：**
   混淆矩阵是一种在机器学习中评估总结模型在测试集上的性能的方法

Actual	Predicted	
	Yes	No
Yes	TURE+	FALSE-
No	FALSE+	TRUE-

   TP=真正类：真实结果为正，且预测结果也为正。
   TN=真负类：真实结果为负，且预测结果也为负。
   FP=假正类：真实结果为负，但预测结果为正。
   FN=假负类：真实结果为正，但预测结果为负。
2. 如何读混淆矩阵：
   前面带有真的说明预测结果与真实结果一致，带有假的说明预测结果与真实结果相反，例如说假正类含义是预测结果为正，但是实际上为负。例如说FN表示假负类，代表这个数据真实结果是正，预测为负

三、混淆矩阵

1. **准确率Accuracy(ACC)**
   正确实例占总实例的比例

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

   这里计算的是正确判定正例和负例的比例，有的时候我们并不关心整体的正确率，我们只

关心局部的正确率，例如说分类垃圾邮件，我们其实更关心的是正常的邮件有多少比例被分类正确，而不是正常邮件和垃圾邮件有多少被分类正确

2. **精确度Precision(PPV)**
   实际正类占预测正类的比例

$$Precision = \frac{TP}{TP + FP}$$

   这里体现了对正类分类的准确性，例如说对于垃圾邮件，正确分类为垃圾邮件，占所有被我们分类为垃圾邮件数量的比例

3. **召回率Recall&灵敏度Sensitivity(TPR)**
   实际正类中被正确预测的比例

$$Recall = \frac{TP}{TP + FN}$$

   这里表示在所有正类中，我们实际上识别出来了多少，这里说的是正确被分类为垃圾邮件的占实际上为垃圾邮件的比例

4. 特异度Specificity(TNR)
   实际负类中被正确分类为负类的比例

$$Specificity = \frac{TN}{TN + FP}$$

5. 剩余指标
   - FPR：$FPR = \frac{FP}{FP+TN}$在真实值为负类的所有结果中，被错误分类为正类的比例，该指标衡量的是负类中被错误分类的比例，FPR越大，表示正类中实际上的负类越多
   - FNR：$FNR = \frac{FN}{TP+FN}$表示真实值为正类的所有结果中，被错误分类为负类的比例
   - RPP：$RPP = \frac{TP+FP}{TP+TN+FP+FN}$表示所有判断为正类的占总观测数的比例
6. 比较：
   - 精确度Precision需要最小化FP，这里强调预测正类的准确性，我们要尽可能减少对误判负类为正类的情况发生，增加判别的准确性
   - 召回率Recall需要最小化FN，这里强调的是对实际正类预测的完备性，我们要尽可能识别出所有的正类，减少对实际正类的错误分类
   - 精确度更重要的情景：
     垃圾邮件检测：不希望正常的邮件被预测为垃圾邮件
     医疗诊断：假阳性（实际上是负，但是预测为正）可能会导致不必要的治疗
     金融欺诈检测：可能会给客户带来不便
   - 召回率更重要的场景：
     疾病筛查：遗漏实际上的病例可能会危及生命
     安全风险检测：未能识别出真实的风险可能会有危险
     质量监控：未检测到的残次品可能会损害企业声誉
7. F分数（F1值）
   - **F1指数**
     在精确度和召回率之间存在反向关系，经常此消彼长，一个增加了另外一个就趋向减少，这样就会导致无法合理对其进行评估，我们使用二者的调和平均数来平衡两者之

间的差异，即F1值

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

F1 Score是最正常的一种，它将精确率和召回率的权重设置相等

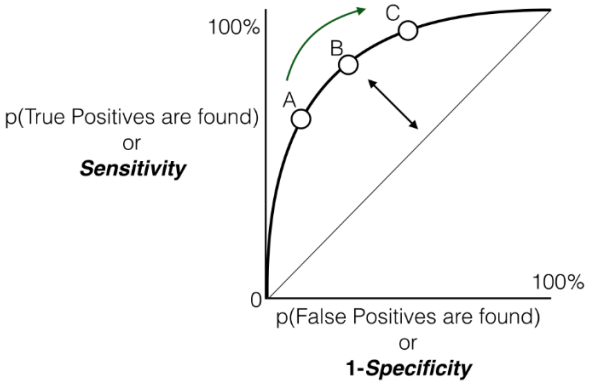
- F-beta Score
  此外还有F-beta Score它是用一个beta权重来控制两者之间的均衡关系

$$F_\beta = (1 + \beta^2)\frac{Precision \times Recall}{\beta^2 \times Precision + Recall}$$

当$\beta > 1$时，召回率的权重更大，当$\beta < 1$的时候精确度的权重更大
当$\beta \to 0$，$F - Score$更加趋近精确度
当$\beta \leftarrow \infty$，$F - Score$更加趋近于召回率
当$\beta = 1$，$F - Score$是精确度和召回率的调和平均值
- 在以下情况下使用 F1 分数：
  希望在假阳性结果和假阴性结果之间取得一种平衡的判断。
  正在处理数据不均衡的情况。
  单独来看，准确率和召回率都无法全面反映实际情况。
- F1分数的取值
  一般来说F1分数落在0到1之间
  当其为0时候，模型完全失效，无法做出任何有用的预测
  当其在0.5的时候，模型的精确度和召回率都很低
  当其在0.7的时候，模型对实际应用可行
  当其大于0.8的时候，模型是比较好的，有比较均衡的表现
  当其等于1的时候，模型实现完美预测

四、ROC曲线

1. **ROC 曲线**
   ROC曲线关注的是模型区分正负总体的能力



由于在比较两种不同的模型时，通常使用混淆矩阵中的单一指标会比使用多个指标更为方便：
真阳性率（TPR），又称灵敏度、命中率或召回率，其定义为 $\frac{TP}{(TP+FN)}$。它表示的是相对于所有正样本而言，被正确判定为正样本的正样本数据点所占的比例。 换句话说，TPR

值越高，我们就会遗漏的正数据点就越少。
假阳性率（FPR），又称误报率、漏检率或 1 - 特异性，其定义为 $\frac{FP}{(FP+TN)}$。直观地看，这个指标相当于所有负样本中被错误地判定为正样本的比例，相对于所有负样本而言。换句话说，这个数值越高 FPR（误报率）方面，那些更具负面特征的数据点将会被错误分类。
在ROC曲线中纵轴表示真阳性率，横轴表示假阳性率

2. ROC曲线性能判断标准
ROC曲线与x轴围成的面积越大(AUC)，分类器的性能越好
ROC图中的45度线代表TPR=FPR，它代表模型把正样本识别为正的概率，等于它把负样本误判为正的概率，因此就表示模型没有什么判别能力，模型的性能跟随机猜测是一样的
综上所述ROC曲线越接近左上角，性能越好，ROC曲线越接近45度线，性能越差
一般来说ROC曲线位于0.9到1是优秀模型，0.8到0.9是一个好的模型，0.7到0.8是一个一般的模型，0.6到0.7是一个差的模型，小于0.6则表示模型失败，无判别能力
3. **PR曲线**
PR曲线关注的是模型区分正类的能力
纵轴是精确度，横轴是召回率
$Precision = \frac{TP}{TP+FP}$
$Recall = \frac{TP}{TP+FN}$
精确度和召回率都关注的是对正类的分类，不关心真负类的分类
随着召回率的增加，精确度可能会减少
一个好的PR曲线有更大的AUC(曲线下面积)

五、偏差与方差

1. 模型误差的组成
认为$Y = f(x) + \epsilon, \epsilon\ N(0, \sigma_e)$，模型预测为$\hat{f}(x)$
误差$Error(x) = E(Y - \hat{f}(x))^2 = (E(\hat{f}(x)) - f(x))^2 + E(\hat{f}(x) - E(\hat{f}(x)))^2 + \sigma_e^2$
其中偏差为$(E(\hat{f}(x)) - f(x))^2$，
方差是$E(\hat{f}(x) - E(\hat{f}(x)))^2$，
误差是σ_e^2，代表的是模型中不可消除的噪声导致的误差
2. 误差类别
在任何机器学习模型中，存在两种主要类型的误差：
可约误差：是指那些其数值能够进一步降低从而有助于改进模型的那些错误。偏差与方差
不可约误差：在机器学习模型中，总会存在一些无法避免的错误，这是因为存在未知变量，而且这些错误的数值无法减少。例如随机噪声。
偏差被定义为由于模型存在的误差导致的预测值与真实值之间存在的误差，偏差代表的是模型的拟合能力
3. **机器学习中降低偏差的方法**
   - 用一个更加复杂的模型，对于高偏差最主要的原因就是模型非常简化，它会无法捕捉到数据的复杂程度
   - 增加特征数量，通过增加更多的特征来训练数据集将会增加模型的复杂程度，改进模型识别数据潜在模式的能力

- 增加训练数据的大小
- 降低正则化手段，降低正则化强度或者完全删除它，都有助于提升模型表现

4. 方差的概念
   方差是指模型使用不同的训练数据子集来训练的时候预测模型性能的改变程度，方差是模型变异程度，代表它对另外一个训练子集的敏感程度，以及它在新训练数据子集的调整程度。代表一种稳定性
5. **降低方差的方法：**
   - 交叉验证：通过将数据划分为训练集和测试集，交叉验证可以帮助识别模型是否过拟合或者欠拟合
   - 特征选择：通过仅仅选择相关的特征将会帮助降低模型的复杂程度
   - 正则化：我们可以用L1正则化或者L2正则化来降低方差
   - 集成方法：结合多种模型来增强集成后的表现，随机森林和Boosting都是常规的集成方法可以降低方差并且增强整体的表现
   - 简化模型：减少模型复杂度，例如减少神经网络中参数和层数，这样也可以降低方差，并且增强整体表现
   - 早停策略，早停是一种技巧防止过拟合，当模型在测试集的性能不再增加时候，停止深度学习模型的训练
   - 增加训练集的大小
6. 方差与偏差对于模型性能的关系
   综上偏差与方差的问题实际上就是解决过拟合和欠拟合的问题，当模型参数越来越大的时候，模型的偏差越来越小，方差越来越大，这时候可能会造成过拟合，相反则可能造成欠拟合