模型评估

评估指标的局限性

1. **准确率的局限性：**
   - 定义：分类正确的样本占总样本个数的比例
   - 优点：
     - 直观，简单
   - 缺点：
     - 当不同类别样本不平衡时，占比大的类别往往会成为影响准确率的主要原因
     - 比如负样本占99%，即使模型把所有样本都预测为负，也能有99%的准确率，但是却无法分辨出任何正样本
   - 可以用类别平均准确率来作为一个更好的评估指标
2. **精确率与召回率的权衡**：
   - 精确率：
     - 分类正确的正样本个数占分类器判定为正样本的样本个数的比例（质量）
   - 召回率：
     - 分类正确的正样本个数占真正的正样本个数的比例（数量）
   - 矛盾：
     - 为了提高精确率，分类器需要尽量在“更有把握”时才把样本预测为正样本，但因此往往会过于保守，而漏掉很多“没有把握”的正样本，导致召回率降低
   - 解决办法：
     - 可以通过测试多个阈值的精确率和召回率来绘制不同模型的P-R曲线，全面衡量模型的性能
     - 计算F1-score，精确率和召回率的调和平均值，2*P*R/(P+R)

3. **平方根误差RMSE**：
    - RMSE失效的原因：
        - 数据中存在离群点/异常值，来源可以结合业务背景分析，比如某个新功能上线等
    - 解决方案：
        - 如果确定离群点为噪音，那么在数据预处理的时候就需要过滤掉
        - 如果不是噪音，那可以在模型中加入先验知识，把离群点产生的原因加入进去
        - 换一个鲁棒性更好的评估指标，例如平均绝对百分比误差MAPE（每个数据点都被归一化了，降低了离群点带来的绝对误差影响）
        - 其实在数据处理的时候就应该进行归一化，有离群点最好采用零均值归一化的方法

ROC曲线

1. **什么是ROC曲线**：
    - 受试者工作特征曲线 Receiver Operatoing Characteristic Curve
    - 横坐标：假阳性率，False Positive / Negative
    - 纵坐标：真阳性率，True Positive / Positive
    - 一条曲线代表了一个模型在不同的阈值下的性能
2. **如何绘制ROC曲线**：
    - 通过不断移动分类器的截断点/阈值，来生产曲线上的每一个点
        - 将横轴的刻度间隔设置为1/N，纵轴的刻度间隔设置为1/P，根据模型输出的预测概率对样本进行降序排序，从(0,0)开始每遇到一个正样本就沿纵轴方向绘制一个刻度的曲线，遇到负样本则向横轴方向绘制，最终抵达(1,1)的位置
3. **什么是AUC，如何计算**：
    - ROC曲线下的面积大小，量化地反映基于ROC曲线衡量出的模型性能
    - 计算方法是沿着ROC横轴做积分
    - AUC越大，说明分类器越可能把真正的正样本排在前面，分类效果更好

4. **ROC曲线和P-R曲线的区别**：
    - 当正负样本分布发生较大变化是，ROC曲线的形状比较稳定，P-R曲线的形状则一般会发生剧烈变化
    - 这一特性使ROC曲线更能够降低不同测试集对模型评估带来的干扰，比较适合数据集不平衡的时候使用
    - 当希望反映模型在特定数据集上的效果时，P-R曲线会更加的直观

余弦距离的应用

1. **什么是余弦相似度 (cosine similarity)?**
    - $cos(A,B) = \frac{A.dot(B)}{norm(A)*norm(B)}$
    - 取值范围是[-1,1]，相同时为1，正交时为0，相反时为-1
    - 余弦距离=1-cos(A,B)，取值范围是[0,2]，数值越大距离越大
2. **为什么在一些场景中要使用余弦相似度而不是欧氏距离?**
    - 欧氏距离体现数值上的绝对差异，余弦距离体现方向上的相对差异
    - 例如，衡量指标不同时：
        1. 统计两部剧的用户观看行为，用户A的观看向量为(0,1)，用户B为(1,0)，此时二者的余弦距离很大，而欧氏距离很小，我们分析两个用户对于不同视频的偏好，更关注相对差异，显然应当使用余弦距离
        2. 当我们分析用户活跃度，以登陆次数和平均观看时长作为特征时，余弦距离会认为(1,10)、(10,100)两个用户距离很近，但显然这两个用户活跃度是有着极大差异的，此时我们更关注数值绝对差异，应当使用欧氏距离
    - 文本数据：当使用词频或者词向量作为特征时，会出现长度差距大（词库量级不同）但是内容相似的情况，此时欧氏距离会很大，然是余弦距离会很小
    - 当特征向量的模长经过归一化处理后，欧氏距离和余弦距离给的结果就会是相同的，存在关系：$D(A,B) = \sqrt{2(1-cos(A,B))}$
    - 无论特征维度有多高，余弦相似度始终保持：相同时为1，正交时为0，相反时为-1，而欧氏距离则受维度影响，范围不固定，对相似度的定义也相对模糊

3. **余弦距离是否是一个严格定义的距离?**
    - 不是，因为不满足三角不等式的定义
    - 需要满足三条距离公理：
        1. 正定性：距离永远是大于0的，满足
        2. 对称性： $1-cos(A,B)=1-cos(B,A)$，满足
        3. 三角不等式：A=(1,0)，B=(1,1)，C=(0,1)是一个反例，不满足

A/B测试的陷阱

1. **什么是A/B测试?**
    - 互联网：验证新模块、新特征、新功能、新产品等是否有效，新算法、新模型的实际效果是否有提升，新设计是否更受欢迎等
    - 机器学习：验证模型在实际应用时的效果的主要手段
2. **在对模型进行过充分的离线评估之后，为什么还要进行在线A/B测试?**
    - 离线评估无法完全消除模型过拟合的影响，无法取代线上评估
    - 离线评估无法完全还原线上的真实环境，比如延迟、数据丢失、标签缺失等情况
    - 离线评估只能对模型本身进行评估，但是一些商业指标就无法计算，比如新上线的推荐算法在离线评估时可能会关注ROC曲线，但是线上评估可以全面了解新算法带来的用户点击率、留存时长、页面访问量等
3. **如何进行线上A/B测试?**
    - 主要手段是用户分桶：实验组和对照组
    - 分组要注意样本的独立性和采样方法的无偏性，记得控制变量（比如同为年轻用户）
    - 实验组用户使用新模型，对照组用户使用旧模型

模型评估的方法

1. **Holdout检验**：
    - 将原始数据集划分成训练集和验证集两部份，验证集用来获取评估指标
    - 优点：简单直接
    - 缺点：在验证集上计算出来的评估指标与原始分组相关性很高

2. k-fold**交叉验证**：
    - 将样本划分成k个大小相等的样本子集，依此遍历子集作为验证集，剩下的子集作为训练集
    - 总共训练和评估模型k次，以k个指标的平均值来作为最终的评估指标
    - 优点：通过平均多个指标值，消除了原始分组的随机性
3. **留一验证**：
    - 每次留下一个样本作为验证集，其余所有样本作为训练集
    - 样本总数为n的时候，依次对n个样本进行遍历，总共进行n次训练和验证，最后平均结果
    - 属于k-fold交叉验证的特例，当k=n的时候就是留一验证
    - 属于留p验证的特例，因为从n个样本中留下p个有很多种选择方式，所以时间开销很大，极少被实际应用
4. **自助法（Bootstrap）**：
    - 当样本量较小的时候，划分样本集会让训练集更小，影响模型的训练效果
    - 通过自助采样法，从n个样本中有放回的采样n个样本作为训练集（其中会包含重复的样本），没被抽取的样本就作为验证集
    - 当n趋于无穷大的时候，大约有$1/e = 36.8\%$的样本会从未被采样到作为训练集，这是1个样本在n次采样中都没有被选择到的可能性

超参数调优

1. **超参数搜索算法的要素：**
    - 目标函数
    - 搜索范围
    - 算法的其他参数，例如：搜索步长等
2. **网格搜索（Grid Search）：**
    - 查找搜索范围内的所有点来确定最优值
    - 当采取较大的搜索范围和较小的搜索步长，很大概率能找到全局最优值
    - 当超参数比较多的时候，这个方法十分消耗计算资源和时间，因此在实际应用中会使用较广的搜索范围和较大的步长来寻找全局最优值的可能位置，然后逐渐缩小搜索范围和步长
    - 目标函数一般是非凸的，所以上述方法很有可能会错过全局最优值

3. **随即搜索（Randomized Search）：**
    - 在搜索范围中随机选取样本点进行验证
    - 如果样本点集够大，那么通过随机采样也能大概率地找到全局最优值，或者是近似值
    - 比网格搜索要更快，但还是结果是无法保证的
4. **贝叶斯优化算法（Bayes Optimization）：**
    - 网格搜索和随机搜索在的测试一个新样本点时，会忽略前一个点提供的信息
    - 贝叶斯优化算法会利用之前测试过的样本点提供的信息，通过对目标函数（样本点和验证指标）的形状进行学习，找到使目标函数向全局最优值提升的参数
    - 首先根据先验分布，假设一个搜集函数，然后每次使用新样本点测试目标函数的时候，利用测试结果（目标函数的反馈）来更新先验分布，不断学习目标函数可能存在的形状，最后由后验分布给出全局最优值可能出现的位置
    - 缺点：到找到一个局部最优值的时候，算法可能会在附近区域不断采样，陷入局部困境
    - 解决办法：找到探索（exploration）和深挖（exploitation）之间的平衡点

过拟合与欠拟合

1. **过拟合和欠拟合具体是指什么现象?**
    - 过拟合指模型过于复杂导致对于训练数据拟合过当，评估时在训练集上表现非常好但是在测试集和新数据上表现较差（泛化能力差）
    - 欠拟合指的是模型在训练和预测时的表现都不好的情况，没有很好地捕捉到数据的特征
2. **降低过拟合风险的方法：**
    - 获得更多的训练数据，减小噪音对训练的影响。直接获取更多数据一般是比较困难的，可以通过一些方法来扩充训练数据，例如：图像数据的平移/旋转/缩放等，或者可以使用生成式对抗网络来合成大量的新训练数据。
    - 降低模型的复杂度，避免模型过度拟合噪音，比如通过剪枝来降低决策树的深度，减少神经网络的层数和神经元个数等。
    - 正则化，给线性模型的参数加上一定的正则约束，避免参数太大带来的过拟合（放大噪音）的风险

3. **降低过欠合风险的方法：**
    - 通过特征工程来添加新特征，解决特征不足或者与样本标签相关性不强导致的欠拟合问题，常用方法：组合特征，领域知识，聚类分箱等。
    - 增加模型的复杂度，加强拟合能力，例如：给线性模型添加高次项，给神经网络增加网络层数和神经元个数，给决策树模型增加枝叶等。
    - 减小正则化系数