

机器学习算法对比				
	决策树	随机森林	gbdt	adBoost
1、算法原理	通过将数据集递归地分割为更小的子集来构建树形结构，每个节点根据某个特征选择最佳分裂点，直到达到停止条件。	基于决策树的集成学习方法，在训练过程中通过自助采样和随机选择特征进行多个决策树的构建，并通过投票或平均预测结果来进行分类或回归。	通过迭代地拟合残差来训练一系列的弱学习器（通常是决策树），每个学习器都试图修正前一个学习器的预测误差，最后将所有学习器的预测结果加权求和得到最终结果。	通过改变样本权重来递归地训练多个弱学习器（如决策树），每个学习器都试图纠正前一个学习器的错误，最后将所有学习器的预测结果加权求和得到最终结果。
2、缺省值处理	可以选择忽略该样本或使用其他策略进行处理。	不直接处理缺失值，而是通过投票或平均多个决策树的预测结果来获得最终结果。	使用插补方法（比如均值、中位数）来处理缺失值，以便在每次迭代中计算残差。	需要进行适当的插补处理，以确保正确训练弱学习器。
3、并行计算	构建单棵决策树时，可以通过并行计算来加快训练速度。	更加注重内存优化，使用GOSS方法选择具有较大梯度的样本进行训练，减少内存使用和加快训练速度。	由于每个弱学习器的训练是顺序进行的，无法直接进行并行计算。	与梯度提升类似，每个弱学习器的训练也是顺序进行的，无法直接进行并行计算。
4、内存使用	内存使用较低，因为只需存储树结构和少量的特征值。	由于需要存储多个决策树，随机森林的内存占用相对较高。	需要存储多个决策树，因此内存占用较高。	需要存储多个弱学习器，内存占用较高。
5、处理大规模数据	可以处理大规模数据，但可能会受限于内存和计算资源。	通过并行计算和随机子采样，随机森林能够有效地处理大规模数据集。	可以处理大规模数据，但较大的数据集可能导致训练时间较长。	可以处理大规模数据，但可能受限于计算资源。
6、样本不平衡	通过调整类别权重或设置不平衡相关参数来处理样本不平衡问题。	通过调整类别权重或使用平衡子采样等策略来处理样本不平衡问题。	通过调整样本权重或使用其他技术（如SMOTE）来处理样本不平衡问题。	使用样本权重来处理样本不平衡，通过递归地学习和纠正错误来提高对少数类的预测性能
7、GPU加速	不支持	不支持	XGBoost、LightGBM支持	不支持
8、优点	<div>1. 直观易解释，树形结构可以清晰地表示特征的重要性和决策过程。</div> <div>2. 能够处理数值型和类别型特征，并且对缺失数据和异常值具有较好的容忍性。</div> <div>3. 可以处理高维度数据集和大规模数据。</div> <div>4. 训练速度相对较快。</div>	<div>1. 随机森林通过随机选择特征子集和自助采样可以降低过拟合风险，提高泛化能力。</div> <div>2. 在处理高维度数据和大规模数据时效果较好。</div> <div>3. 能够评估特征的重要性，可以用于特征选择。</div> <div>4. 具有并行计算的能力，可以加快训练速度。</div>	<div>1. 梯度提升通过迭代地拟合残差来提高模型性能，能够捕捉数据中的复杂关系。</div> <div>2. 相对于随机森林，梯度提升更加灵活且通常具有更好的预测性能。</div> <div>3. 能够处理各种类型的特征。</div>	<div>• AdaBoost通过递归地学习和纠正错误来提高模型性能，能够获取更高的准确度。</div> <div>• 可以处理各种类型的特征。</div> <div>• 相对于单个弱学习器，AdaBoost能够更好地处理样本不平衡问题。</div>
9、缺点	<div>1. 容易过拟合训练数据，导致在未见过的数据上表现不佳。可以通过剪枝和限制树深度等方法来减轻过拟合。</div> <div>2. 对于连续型特征，需要进行离散化处理。</div> <div>3. 决策树的划分是贪心的，可能导致局部最优解而非全局最优解。</div>	<div>1. 对于噪声较大或包含冗余特征的数据集，可能会导致模型性能下降。</div> <div>2. 由于需要构建多个决策树，并行计算和存储开销较高。</div> <div>3. 模型的预测结果不容易解释。</div>	<div>1. 对于大规模数据集，训练时间较长。但可以通过使用近似算法（如Histogram-based Gradient Boosting）来加快速度。</div> <div>2. 容易过拟合训练数据，需要调节参数来控制模型的复杂度。</div> <div>3. 模型的预测结果不容易解释。</div>	<div>• 对于噪声较大的数据集，AdaBoost可能过拟合训练数据。</div> <div>• 对异常值和高群点敏感，可能导致模型性能下降。</div> <div>• 训练时间较长。</div>