机器学习算法对比			
	xgboost	lightgbm	catboost
1、算法原理	使用贪心算法选择最佳分裂点，遍历所有可能分裂点并计算增益，选择增益最大的分裂点。	采用基于直方图的算法，在构建直方图时对特征进行离散化，并选择一个或多个具有最大增益的分裂点。	使用基于梯度的决策树算法，通过对类别特征进行编码和使用对称树来处理类别特征的组合。
2、缺省值处理	使用缺省值将样本分配给左侧或右侧子节点，并在训练时学习缺省值是分裂的哪一侧。	在构建直方图时，将缺省值作为特殊的bin来处理，并根据增益选择最佳分裂。	使用基于统计的方法来处理缺省值，如指定一个默认的分裂方向或通过迭代训练该特征的分布情况。
3、并行计算	支持并行计算，可以利用多个核心进行特征分裂和节点分裂，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。	支持并行计算，在默认设置下会自动利用所有可用核心。
4、内存使用	采用按列存储的方式，对于稀疏数据和大规模特征空间，内存开销较高。	在构建直方图时使用离散化的特征表示，减少内存占用。同时支持将数据存储在压缩格式中，进一步减少内存需求。	使用特定的数据结构和压缩技术来降低内存需求，尤其针对高基数类别特征。
5、处理大规模数据	适用于处理大规模数据集，但对于高维度数据可能存在内存限制。	通过优化技术，在训练速度和内存利用方面更加出色，特别适合处理具有大量特征和高维度的数据集	通过使用基于梯度的决策树算法和内存优化技术，能够处理大规模数据集。
6、样本不平衡	权重调整+调整损失函数	样本权重+采样策略+调整损失函数	内置了对类别不平衡的自动处理设置参数+特定的损失函数
7、GPU加速	支持在GPU上加速计算，通过使用CUDA库实现并行计算，能够显著提升训练和预测的速度。	从版本2.2.1开始支持GPU加速，可以利用GPU进行特征分裂和节点分裂。	从版本0.15开始支持GPU加速，可以在GPU上并行计算，并且适用于大规模数据集。
8、优点	1. 可扩展性：XGBoost具有较好的可扩展性，能够处理大规模数据集和高维度特征，并针对稀疏数据进行了优化。 2. 准确性高：XGBoost通过贪心算法选择最佳分裂点，提供了较高的预测准确率，而且能够自动进行特征选择，减少了特征工程的需求。 3. 灵活性：XGBoost提供了丰富的参数调节选项，使得用户可以根据具体问题进行调整，从而获得更好的性能。	1. 高效性：LightGBM在训练速度上相对较快，通过采用基于直方图的算法、并行计算和内存优化技术，能够高效地处理大规模数据和高维度特征。 2. 低内存占用：LightGBM使用离散化的特征表示和压缩格式存储数据，减少了内存的需求，尤其适合处理内存受限的情况。 3. 准确性高：LightGBM通过构建直方图并选择具有最大增益的分裂点，可以更好地捕获特征之间的关系，从而提高模型的准确率。 4. 适应大规模数据集：LightGBM通过GOSS方法选择具有较大梯度的样本进行训练，能够处理大规模数据集，并且对于高维度问题表现出色。	1. 处理类别特征：CatBoost能够自动处理类别特征，无需进行手动编码。它使用对称树来处理组合特征，提供了更好的泛化能力。 2. 鲁棒性：CatBoost具有较强的鲁棒性，可以处理噪声和离群值，并且对于样本不平衡问题表现良好。 3. 可扩展性：CatBoost支持多线程训练，并且自适应地利用所有可用核心，从而加快训练速度。
9、缺点	1. 内存占用较高：XGBoost采用按列存储方式，在处理大规模特征空间时可能会消耗较多的内存。 2. 训练时间较长：相对于LightGBM和CatBoost，XGBoost的训练时间可能会稍长一些。	1. 对噪声敏感：LightGBM在个别样本或特征上可能对噪声敏感，这可能导致过拟合。因此，在一些噪声较多的数据集上，需要更加小心地调整参数来避免过拟合。 2. 不支持GPU加速：目前的LightGBM版本并不直接支持GPU加速，这意味着在需要使用GPU进行计算加速时，可能需要考虑其他算法或框架。	1. 内存占用较高：CatBoost在处理高基数类别特征时，会产生大量的哑变量，导致内存占用较高。 2. 训练时间较长：相对于LightGBM和XGBoost，CatBoost的训练时间可能会稍长一些。