XGBoost 阅读报告

201822150261 仲世超 人工智能 18

一、集成学习的两种方法

- Bagging：利用多个过拟合的弱学习器
- Boosting：利用多个欠拟合的弱学习器

二、XGBoost 的特点

Boosting 是一种常用的统计学习方法，在训练过程中，通过改变训练样本的权重，学习多个分类器，最终获得最优分类器。在每一轮训练结束之后，降低被正确分类的训练样本权重，增大分类错误的样本权重，多次训练之后，一些被错误分类的训练样本会获得更多关注，而正确的训练样本权重趋近于 0，得到多个简单的分类器，通过对这些分类器进行组合，得到一个最终模型。其典型算法有:GBDT/GBRT，Adaboost，XGBoost 和 LightGBM。

XGBoost 在传统 Boosting 的基础上，利用 CPU 的多线程，引入正则化项，加入剪辑，一定程度上降低了复杂度。

XGBoost 对损失函数做了二阶的泰勒展开，并在目标函数之外加入了正则项对整体求最优解，用以权衡目标函数的下降和模型的复杂程度，避免过拟合。

当样本存在缺失值时，XGBoost 能自动学习分裂方向。

XGBoost 在每次迭代之后，为叶子结点分配学习速率，降低每棵树的权重，减少每棵树的影响，为后面提供更好的学习空间。

三、XGBoost 的参数

XGBoost 的参数一共分为 3 类：
- 通用参数
- Booster 参数
- 学习目标参数

1、通用参数：

- booster：有两种参数选择 gbtree（树）和 gbliner（线性）。
- silent：静默模式，为 1 时模型运行不输出
- nthread：使用线程数，一般设置为-1，即使用所有线程

2、Booster 参数：

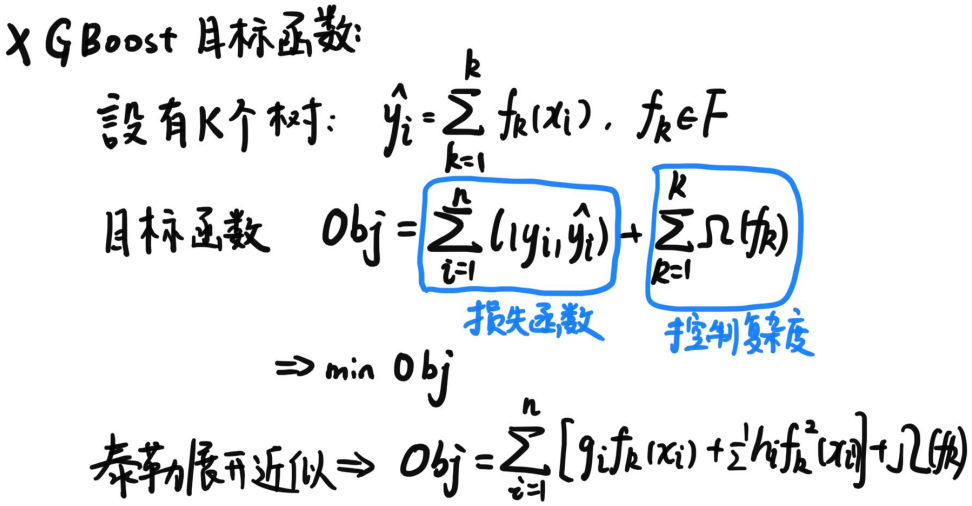
- n_estimator：生成的最大数的数目，也是最大迭代次数

- learning_rate：系统默认值为 0.3，每一步迭代的步长，太大了运行准确率不高，太小了运行速度慢，一般使用比默认值小一点，0.1 左右
- gamma：系统默认为 0，一般无需修改，在节点分裂时，只有分裂后损失函数的值下降了，才会分裂这个节点。gamma 指定了这个节点分裂所需的最小损失函数。这个参数值越大，算法越保守，因为 gamma 值越大的时候，损失函数下降更多才可以分裂节点，所以树生成的时候更不容易分裂节点
- subsample：系统默认为 1，这个参数控制对于每棵树，随机采样的比例，减少这个参数的值，算法会更加保守，避免过拟合。典型值 0.5-1，0.5 代表平均采样，防止过拟合
- colsample_bytree：系统默认值为 1，一般设置成 0.8 左右。用来控制每棵树随机采样的列数的占比（每一列是一个特征），典型值 0.5-1
- colsample_bylevel：默认值 1，一般也设置为 1，比前一个更细致
- max_depth：默认值 6，常用 3-10，数的最大深度
- max_delta_step：默认为 0，常用 0，这个参数限制了每棵树权重改变的最大步长，如果这个参数为 0，则意味着没有约束
- lambda：默认值 0，权重的 L2 正则化项（也称为 reg_lambda）
- alpha：默认值 0，权重的 L1 正则化项（也称为 reg_alpha）
- scale_pos_weight：默认值 1，在各类别样本十分不平衡时，把这个参数设定为一个正值，可以使算法更快收敛。通常可以将其设置为负样本的数目与正样本数目的比值

3、学习目标参数

- objective【缺省值=reg：linear】
  - reg: linear 线性回归
  - reg: logistic 逻辑回归
  - binary: logistic 二分类逻辑回归，输出为概率
  - binary: logitraw 二分类逻辑回归，输出是 WTX
  - count: poisson 计数问题的 possion 回归，输出结果是 possion 分布。在回归中，max_delta_step 的缺省值为 0.7
  - multi: softmax 设置 XGBoost 使用 softmax 目标函数做多分类，需要设置参数 num_class（类别个数）
  - multi: softprob 如同 softmax，但是输出结果为 ndata*nclass 的向量，其中的值是每个数据分为每个类的概率
- eval_metric 【缺省值=通过目标函数选择】
  - rmse 均方根误差
  - mae 平均绝对值误差
  - logloss negativelog-likelihood
  - error 二分类错误率
  - merror 多分类错误率
  - mlogloss 多分类 log 损失
  - auc 曲面下面积
  - ndcg Normalized Discounted Cumulative Gain
  - map 平均正确率

四、XGBoost 的目标函数



五、XGBoost 的缺点

1）采用预排序，在迭代之前，对结点的特征做预排序，遍历选择最优分割点，数据量大时，贪心法耗时占用的内存高，数据分割的复杂度高；

2）采用 level-wise 生成决策树，同时分裂同一层的叶子，从而进行多线程优化，不容易过拟合，但很多叶子节点的分裂增益较低，没必要进行跟进一步的分裂，这就带来了不必要的开销。

六、参考文献

1、https://blog.csdn.net/qq_33880788/article/details/79463534
2、周志华，机器学习, no. 84-85，清华大学出版社, 2016.
3、邱锡鹏，神经网络与深度学习, no. 74, Github, 2020.
4、https://www.cnblogs.com/willnote/p/6801496.html
5、