1.Bagging之随机森林

随机森林改变了决策树容易过拟合的问题，这主要是由两个操作所优化的：

（1）Boostrap从袋内有放回的抽取样本值

（2）每次随机抽取一定数量的特征（通常为sqr(n)）

分类问题：采用Bagging投票的方式选择类别频次最高的

回归问题：直接取每棵树结果的平均值

常见参数：树最大深度、树的个数、节点上的最小样本数、特征数

误差分析：将每个树的未采样样本作为预测样本统计误差作为误分率

优点：可以并行计算；不需要特征选择；可以总结出特征重要性；可以处理缺失数据；不需要额外设计测试集

缺点：在回归上不能输出连续结果

2.Boosting之Adaboost

Boosting的本质实际上是一个加法模型，通过改变训练样本权重学习多个分类器进行一些线性组合。而Adaboost就是加法模型+指数损失函数+前项分布算法。Adaboost就是从弱分类器出发反复训练，在其中不断调整数据权重或概率分布，同时提高前一轮被弱分类器误分的样本的权值。最后用分类器进行投票表决（但是分类器的重要性不同）。

3.Boosting之GBDT

将基分类器变成二叉树，回归用回归二叉树，分类用分类二叉树。和上面的Adaboost相比，回归树的损失函数为平方损失，可以用指数损失函数定义分类问题。但是对于一般损失函数怎么计算呢？GBDT（梯度

提升决策树）是为了解决一般损失函数的优化问题，方法是用损失函数的负梯度在当前模型的值来模拟回归问题中残差的近似值。

注：由于GBDT很容易出现过拟合问题，所以推荐GBDT深度不要超过6，而随机森林可以在15以上。

4.Boosting之Xgboost

这个工具主要有以下几个特点：

支持线性分类器；可以自定义损失函数，并且可以用于二阶偏导；加入了正则化项：叶节点数、每个叶节点输出score的L2-norm；支持特征抽样；在一定情况下支持并行，只有在建树的阶段才会用到，每个节点可以并行的寻找分裂特征。