

【干货】集成学习原理总结

原创 石头 机器学习算法那些事 2018-11-27

前言

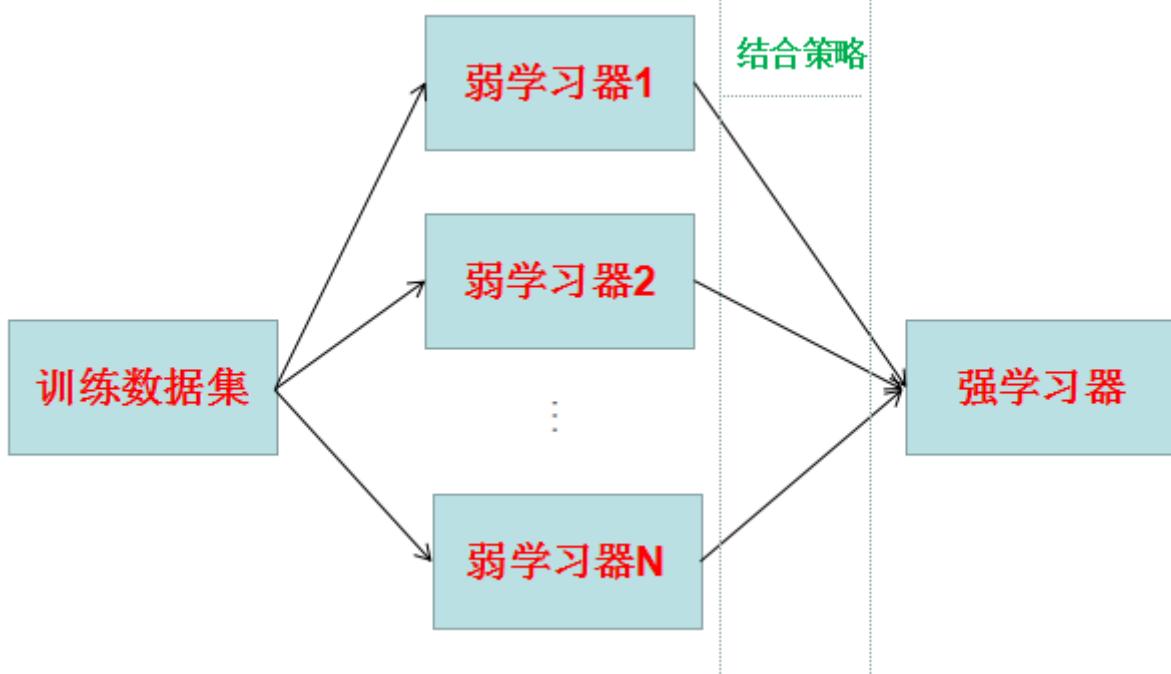
集成学习是目前比较火的机器学习方法，也是面试官考察的一个重点方向。集成学习不是一种机器学习方法，它是通过结合多个机器学习模型来给出学习结果，集成学习很好的避免了单一学习模型带来的过拟合问题，本文简明扼要的对集成学习原理做一个总结。

目录

1. 集成学习概述
2. 集成学习之bagging
3. 集成学习之boosting
4. 集成学习之结合策略
5. bagging和boosting两者之间的区别
6. bagging和boosting的方差和偏差问题讨论
7. 总结

集成学习概述

集成学习能够通过训练数据集产生多个学习模型，然后通过一定的结合策略生成强学习模型，如下图：



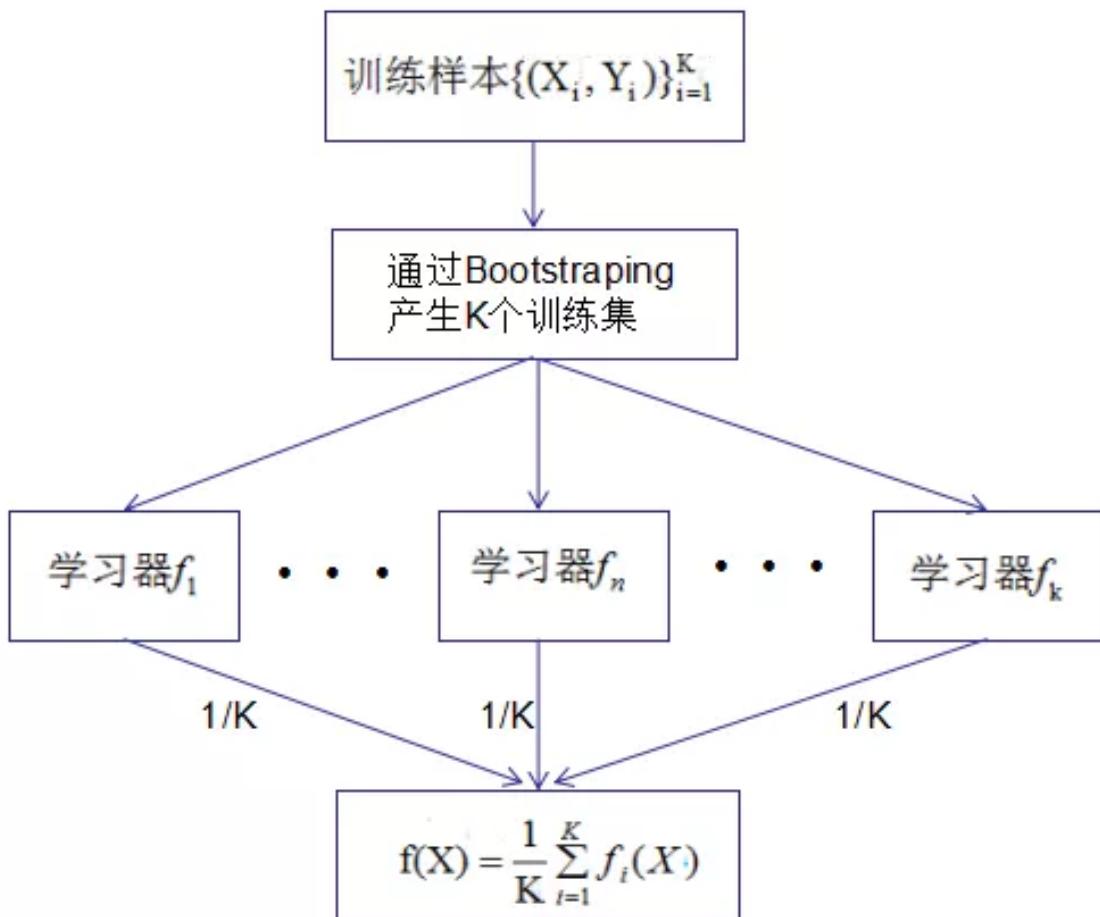
集成学习包括Bagging方法和Boosting方法，下面详细分析这两种方法。

集成学习之Bagging

Bagging即套袋法, 算法过程如下:

- (1) 从训练样本集中随机可放回抽样 (Bootstrapping)N次, 得到与训练集相同大小的训练集, 重复抽样K次, 得到K个训练集。
- (2) 每个训练集得到一个最优模型, K个训练集得到K个最优模型。
- (3) 分类问题: 对K个模型采用投票的方式得到分类结果; 回归问题: 对K个模型的值求平均得到分类结果。

Bagging算法图如下:



Bagging法假设训练样本集服从均匀分布, 即 $1/N$ 。

集成学习之Boosting

Boosting算法中, 每一个样本数据是有权重的, 每一个学习器是有先后顺序的。在PAC (概率近似正确) 的学习框架下, 一定可以将弱分类器组装成一个强分类器。

关于Boosting的两个核心问题

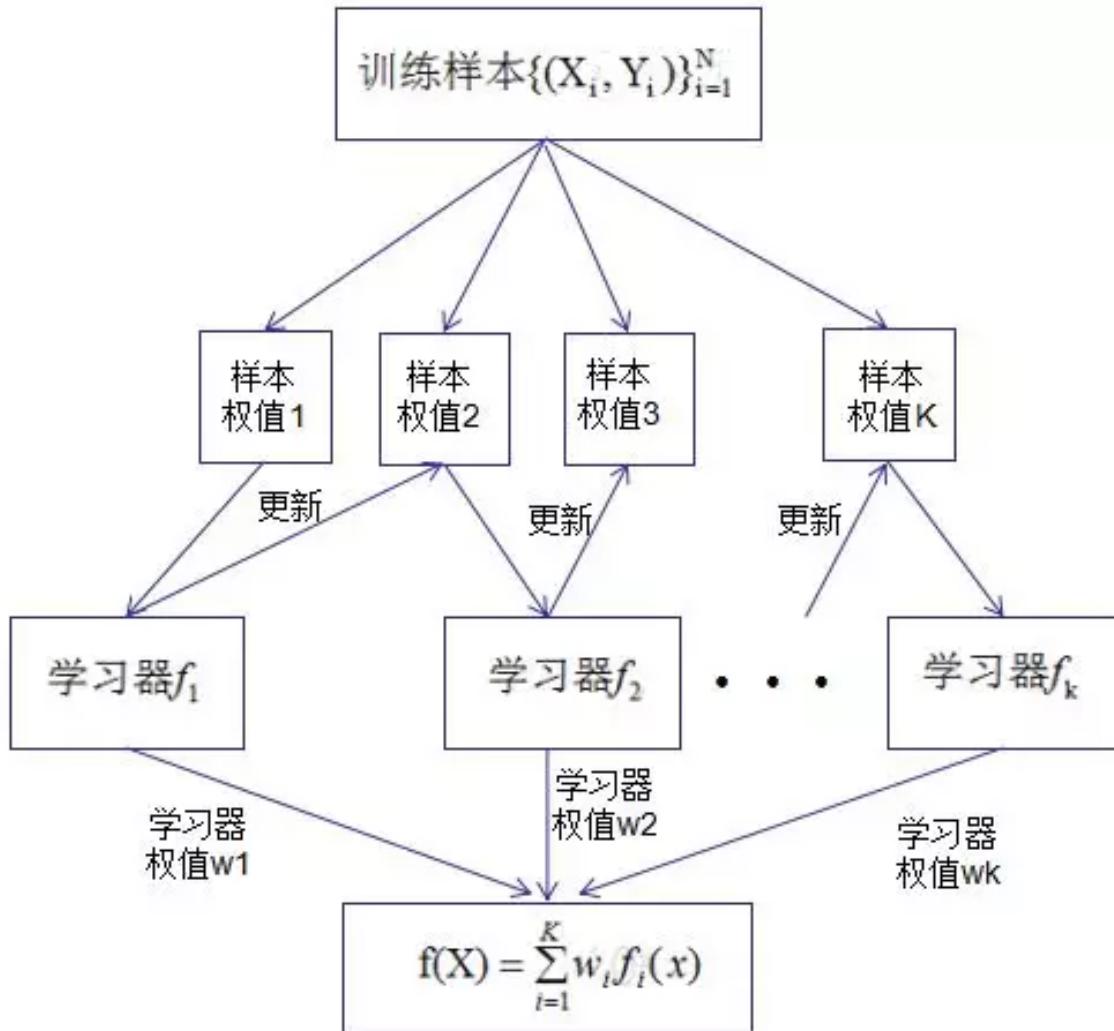
- (1) 每一轮如何改变训练数据的权值和概率分布?

通过提高那些在前一轮被弱学习器分错样例的权值, 减小前一轮正确样例的权值, 使**学习器重点学习分错的样本, 提高学习器的性能**。

(2) 通过什么方式来组合弱学习器?

通过加法模型将弱学习器进行线性组合，学习器准确率大，则相应的学习器权值大；反之，则学习器的权值小。即给学习器好的模型一个较大的确信度，提高学习器的性能。

Boosting算法如下图：



其中，学习器性能越好，对应的权值也越大。样本权值1初始化为 $1/N$ ，即初始样本集服从均匀分布，后面随着前一个学习器的结果更新样本权值。

集成学习之结合策略

集成学习得到多个学习器后，结合策略得到最终的结果。通常用到最多的是平均法，投票法和学习法。

1. 平均法

对于**数值类的回归预测**，通常使用的结合策略是平均法，即对K个学习器的学习结果求平均，得到最终的预测结果。

2. 投票法

对于**分类问题的预测**，通常使用的结合策略是投票法，也就是我们常说的少数服从多数。即对K个学习器的分类结果作一个统计，出现次数最多的类作为预测类。

3. 学习法

上面两种结合策略方法比较简单，可能学习误差较大。因此，我们尝试用学习法去预测结果，**学习法**是将K个学习器的分类结果再次作为输入，将训练集的输出作为输出，重新训练一个学习器来得到最终结果。

Bagging和Boosting两者之间的区别

1) 训练样本集

Bagging：训练集是有放回抽样，从原始集中选出的K组训练集是相互独立的。

Boosting：每一次迭代的训练集不变。

2) 训练样本权重

Bagging：每个训练样本的权重相等，即1/N。

Boosting：根据学习器的错误率不断调整样例的权值，错误率越大，权值越大。

3) 预测函数的权重：

Bagging：K组学习器的权重相等，即1/K。

Boosting：学习器性能好的分配较大的权重，学习器性能差的分配较小的权重。

4) 并行计算

Bagging：K组学习器模型可以并行生成。

Boosting：K组学习器只能顺序生成，因为后一个模型的样本权值需要前一个学习器模型的结果。

Bagging和Boosting的方差和偏差问题讨论

其实这一节内容也属于上一节部分，这是个容易忽视且比较抽象的问题，我把Bagging和Boosting的方差和偏差问题作一个新的小节，希望引起大家的注意。

1. Bagging减小模型的方差

bagging对样本进行有放回的重采样，学习结果是各个学习模型的平均值。由于重采样的样本集具有相似性以及使用相同的学习器模型，因此，各学习模型的结果相近，即模型有近似相等的偏差和方差。

假设 X_i 为*i*组训练样本集，各组训练样本集是相互独立的，不同的训练样本集代表不同的模型，由概率论可知：

$$E\left[\frac{\sum X_i}{n}\right] = E[X_i]$$

$$Var\left[\frac{\sum X_i}{n}\right] = \frac{1}{n^2} Var[X_i]$$

$$\frac{1}{n^2}$$

大家发现了没有，均值没变，但是方差却减小到只有原来方差的 $\frac{1}{n^2}$ 。因此，Bagging法是显著的减小了学习器的方差。

2. Boosting是减小模型的偏差

Boosting是从学习模型的优化角度去更新样本权值和分配学习器权值，因此，学习模型随着迭代次数的增加，**模型偏差越来越小。** Boosting是通过迭代的方式去更新模型，因此各个子模型的之间是强相关的，强相关的意思是各模型的相似度很多，且训练集一直是不变的。**因此模型之和并不能降低方差。**

总结

Bagging和Boosting方法都是把若干个学习器整合为一个学习器的方法，Bagging方法可以降低模型的方差，Boosting方法可以降低模型的偏差，在实际工作中，因情况需要选择集成方法。

下面是决策树与这些算法框架进行结合所得到的新的算法：

- 1) Bagging + 决策树 = 随机森林
- 2) AdaBoost + 决策树 = 提升树
- 3) **Gradient Boosting + 决策树 = GBDT**

参考：

<https://www.cnblogs.com/liuwu265/p/4690486.html>

<https://www.cnblogs.com/earendil/p/8872001.html>

<https://blog.csdn.net/google19890102/article/details/51746402/>

<http://www.cnblogs.com/pinard/p/6131423.html>

推荐阅读资料

[决策树算法总结](#)

[机器学习模型评估方法](#)

[支持向量机：人脸识别](#)

[浅谈频率学派和贝叶斯学派](#)



长按二维码关注

机器学习算法那些事
微信: beautifulife244

砥砺前行 不忘初心