# 原理阶段

1. **什么是集成方法？**

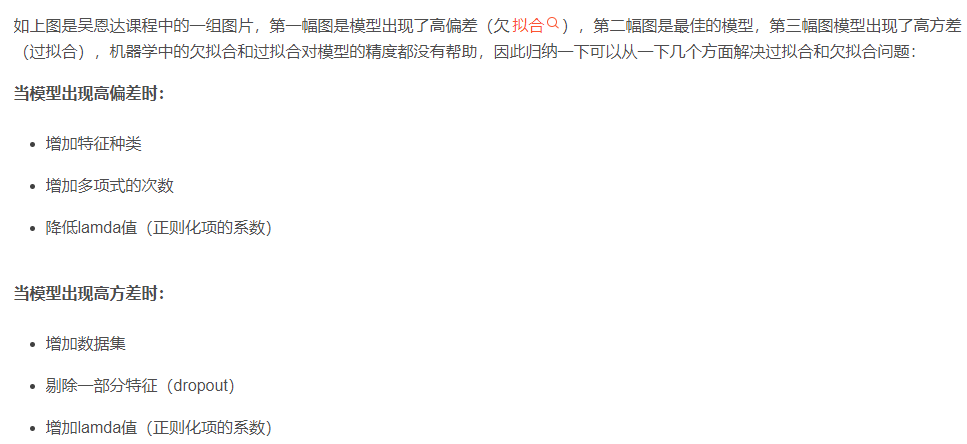
集成方法的思想是通过将这些弱学习器的偏置和/或方差结合起来，从而创建一个「强学习器」（或「集成模型」），从而获得更好的性能。

1. **如何组合弱学习器**

为了建立一个集成学习方法，我们首先要选择待聚合的基础模型。在大多数情况下（包括在众所周知的 bagging 和 boosting 方法中），我们会使用单一的基础学习算法，这样一来我们就有了以不同方式训练的同质弱学习器。这样得到的集成模型被称为「同质的」。然而，也有一些方法使用不同种类的基础学习算法：将一些异质的弱学习器组合成「异质集成模型」。

很重要的一点是：我们对弱学习器的选择应该和我们聚合这些模型的方式相一致。如果我们选择具有低偏置高方差的基础模型，我们应该使用一种倾向于减小方差的聚合方法；而如果我们选择具有低方差高偏置的基础模型，我们应该使用一种倾向于减小偏置的聚合方法。

1. **高偏差和高方差**



1. **三种集成方法**

bagging，该方法通常考虑的是同质弱学习器，相互独立地并行学习这些弱学习器，并按照某种确定性的平均过程将它们组合起来。

boosting，该方法通常考虑的也是同质弱学习器。它以一种高度自适应的方法顺序地学习这些弱学习器（每个基础模型都依赖于前面的模型），并按照某种确定性的策略将它们组合起来。

stacking，该方法通常考虑的是异质弱学习器，并行地学习它们，并通过训练一个「元模型」将它们组合起来，根据不同弱模型的预测结果输出一个最终的预测结果。

非常粗略地说，我们可以说 bagging 的重点在于获得一个方差比其组成部分更小的集成模型，而 boosting 和 stacking 则将主要生成偏置比其组成部分更低的强模型（即使方差也可以被减小）。

1. **Bagging算法**

（1）bagging算法可以解决回归问题和分类问题。

（2）它从原始数据中随机抽取n个样本，重复s次，于是就有个s个训练集，每个训练集都可以训练出一个弱分类器，最终生成s个弱分类器，预测结果将有这些分类器投票决定（选择分类器投票结果中最多的类别作为最后预测结果）。代表的有随机森林。

1. **Bagging投票算法**

(1): 一票否决：对训练出来的多个方法树计算RMSE(AUC)，RMSE最小的话(AUC最大的话)，使用该方法树。

(2): 少数服从多数（可以加权）

绝对多数投票法：假设有一半以上的基学习器都预测了类别c，那么此时集成后的学习器给出的预测结果就是c，否则拒绝预测。

相对投票法：这个是选择票数最多的类别，如果最多票数的类别个数大于1，则随机从中选择一个。

(3): 阈值表决

简单平均法

加权平均法

1. **Bagging算法代表**

随机森林

1. **什么是随机森林**

随机森林是一种多功能的机器学习算法，能够执行回归和分类的任务。同时，它也是一种数据降维手段，用于处理缺失值、异常值以及其他数据探索中的重要步骤，并取得了不错的成效。另外，它还担任了集成学习中的重要方法，在将几个低效模型整合为一个高效模型时大显身手。

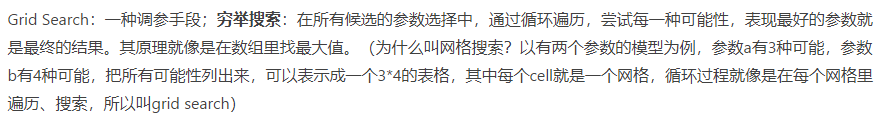
在随机森林中，我们将生成很多的决策树，并不像在CART模型里一样只生成唯一的树。（1）当在基于某些属性对一个新的对象进行分类判别时，随机森林中的每一棵树都会给出自己的分类选择，并由此进行“投票”，森林整体的输出结果将会是票数最多的分类选项；（2）而在回归问题中，随机森林的输出将会是所有决策树输出的平均值。

1. **Bagging和随机森林的区别？**

随机森林（Random Forset，简称RF）由于是bagging算法的一个扩展变体，RF在此基础上在决策树训练中引入了随机属性选择。假定属性一共有d个，那么RF则在树结点上在随机选择一个包含k个属性的子集。再从其中选择最优。k值一般推荐k=log2d。

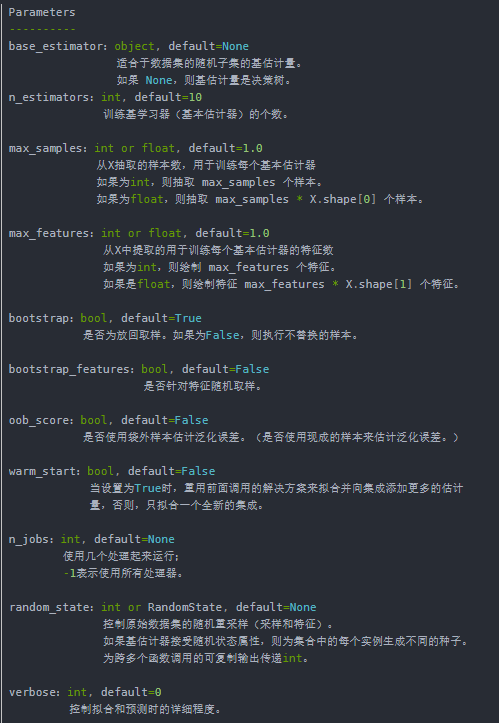
随机森林简单、容易实现、计算开销小。在个体学习器之间的差异度增加后泛化性能会进一步提升。但是RF在基学习器较少时，泛化性能通常较差，随着个体学习器的增加，泛化性能会有所改善。

1. **什么是GridSearch网格搜索**

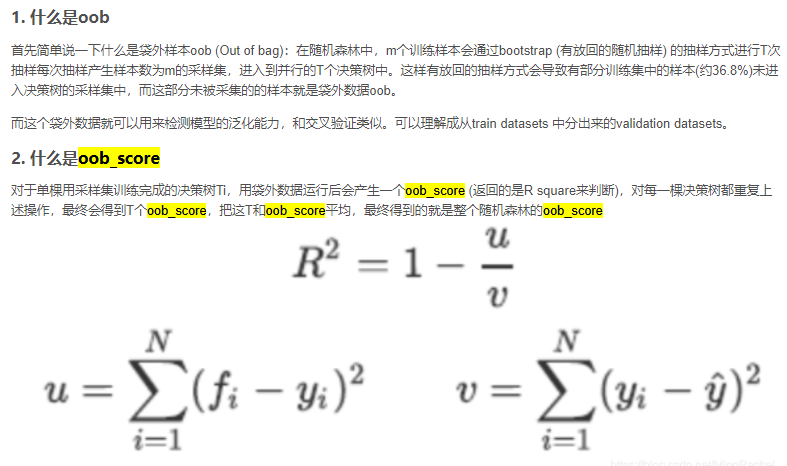


# 代码阶段

1. **BaggingClassifier使用**



**2. oob\_score啥意思**



思考题