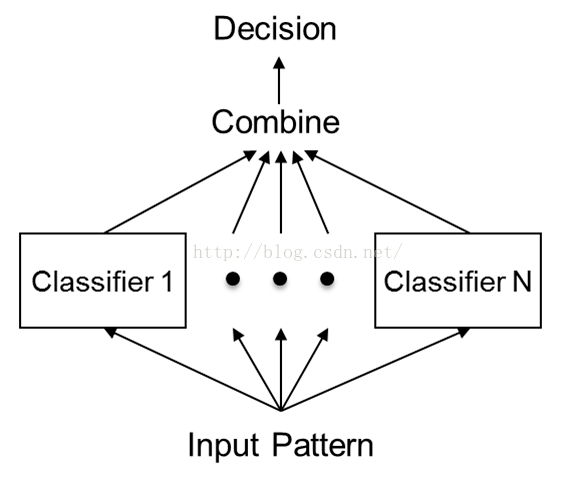
# 随机森林算法实现

## 1.概念

(1) 集成学习

集成学习是使用一系列学习器进行学习，并使用某种规则把各个学习结果进行整合从而获得比单个学习器更好的学习效果的一种机器学习方法。集成学习的方法都是指的同质个体学习器。而同质个体学习器使用最多的模型是CART决策树和神经网络。同质个体学习器按照个体学习器之间是否存在依赖关系可以分为两类，第一个是个体学习器之间存在强依赖关系，一系列个体学习器基本都需要串行生成，代表算法是boosting系列算法，第二个是个体学习器之间不存在强依赖关系，一系列个体学习器可以并行生成，代表算法是bagging和随机森林（Random Forest）系列算法。如下图所示



(2) 随机森林

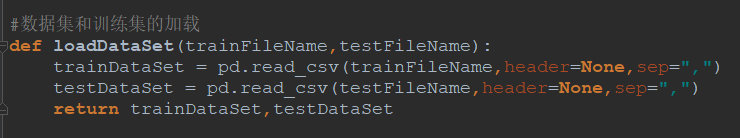
是用随机的方式建立一个森林，森林里面有很多的决策树组成，随机森林的每一棵决策树之间是没有关联的。在得到森林之后，当有一个新的输入样本进入的时候，就让森林中的每一棵决策树分别进行一下判断，看看这个样本应该属于哪一类（对于分类算法），然后看看哪一类被选择最多，就预测这个样本为那一类。随机森林在决策树的训练过程中引入了随机属性选择，随机森林基学习器的多样性不仅来自样本扰动，还来自属性扰动，这就使的最终形成的集成学习模型的泛化性能较强。

## 2.原理

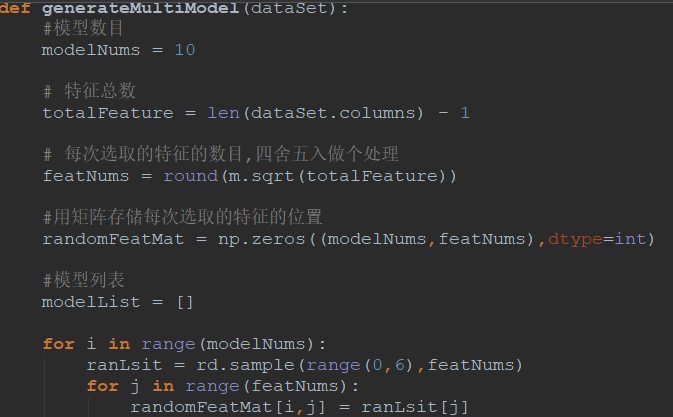
假设有k个特征，每次随机选取sqrt(k)个特征来训练模型，当随机选取10次之后，就有10个模型生成，然后用10个模型分别对测试集进行测试，如果是回归问题，就对10个模型的预测值取个平均值，如果是分类问题，就对10个模型的预测值进行投票，选取票数最多的那个作为预测值。

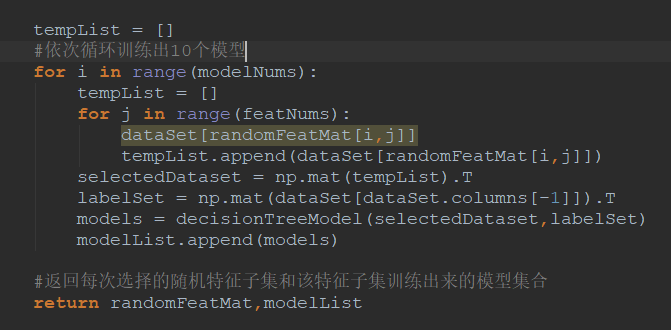
## 3.程序源码

(1) 数据集的加载:

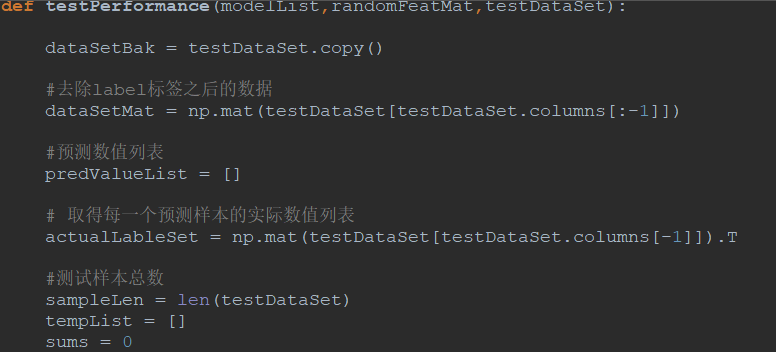


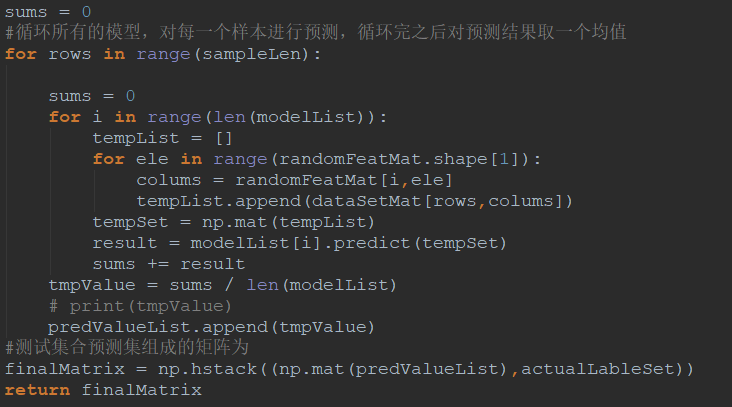
(2) 随机选取特征，训练多个模型



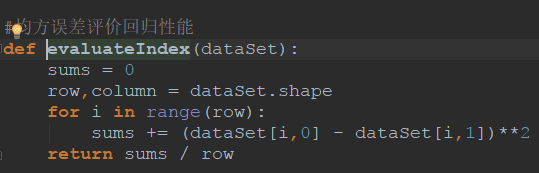


(3) 用训练生成的模型对测试集进行测试，并且返回测试结果

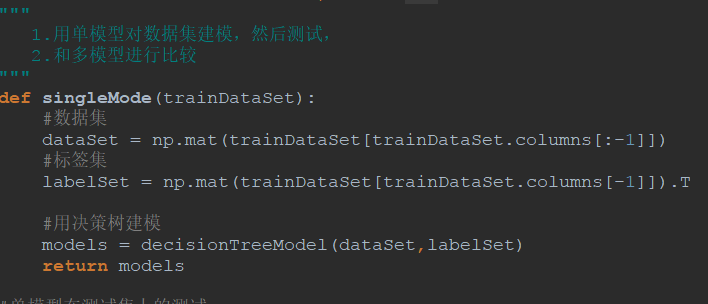




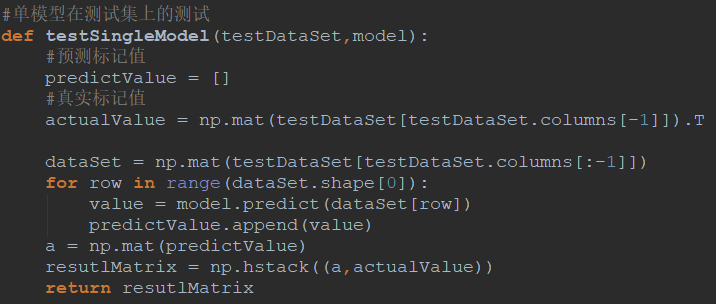
(4) 求解随机森林的均方误差(因为是回归问题)



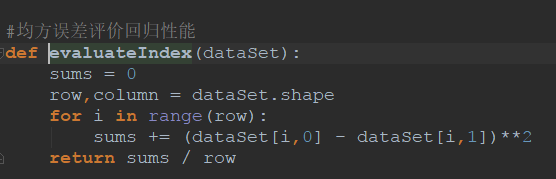
(5) 训练单个模型(决策树)



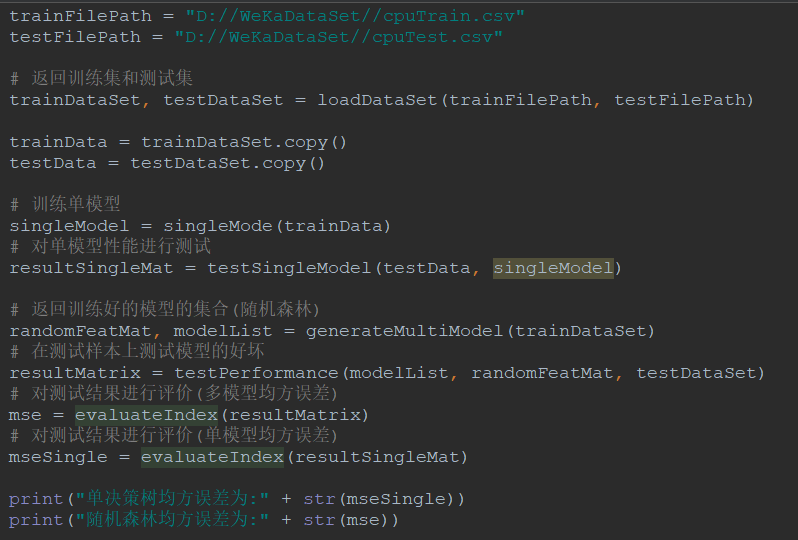
(6) 测试单决策树的性能



(7) 求解单模型均方误差

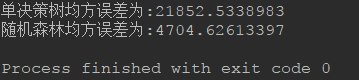


(8) 程序主函数如下



## 4.程序执行结果

程序最终执行结果如下图所示



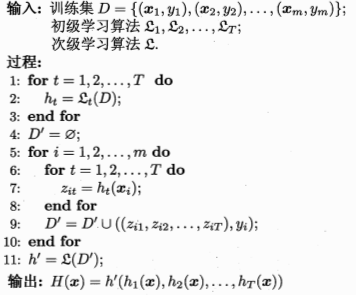
## 5.总结

运用多模型对数据进行预测,性能比单模型提高不少,但是由于随机森林每次都是随机选取sqrt(k)个特征来训练模型，导致最终程序的均方误差起伏较大。运用随机森林可以降低方差。

# 二. 基于线性回归的融合模型算法实现

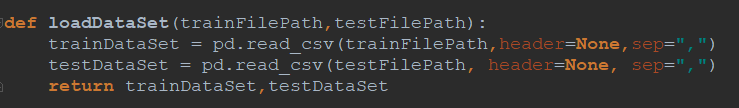
## 1.原理

stacking 模型融合，首先从初始训练集训练出初级学习器，把初级学习器的输出当做次级训练集的输入特征，来训练次级学习器,初始样本的标记仍被当做样例标记。Stacking算法描述如下图所示:

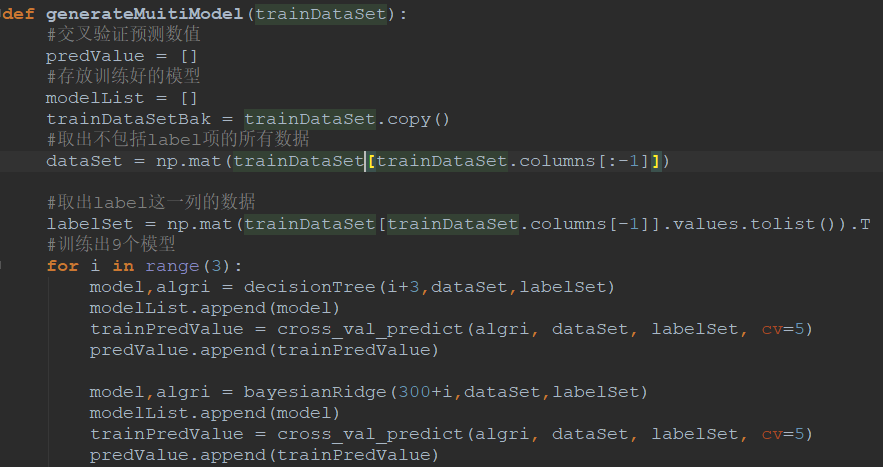


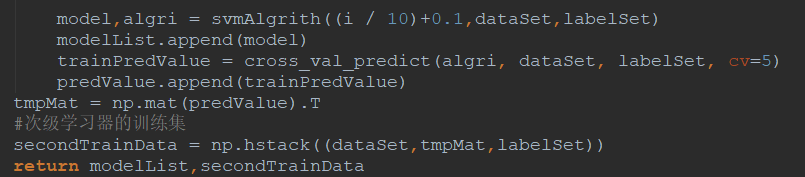
## 2.程序源码

(1) 数据集的加载

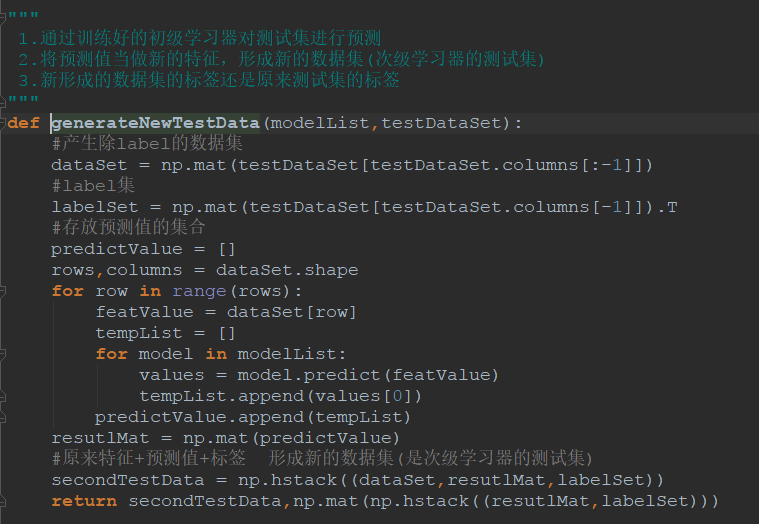


(2) 训练多个模型(通过改变算法参数和数据) 并且对训练集进行交叉预测以产生次级训练集。

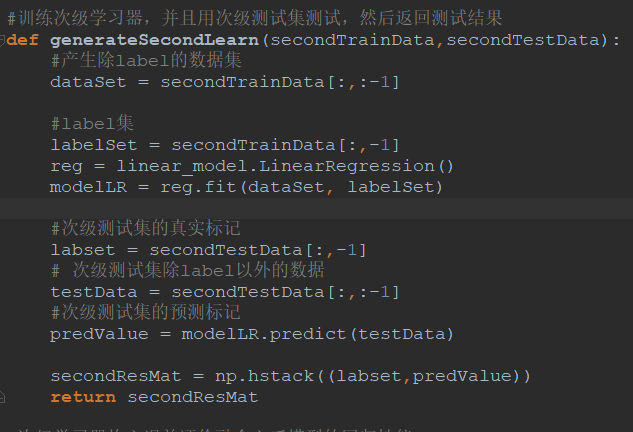




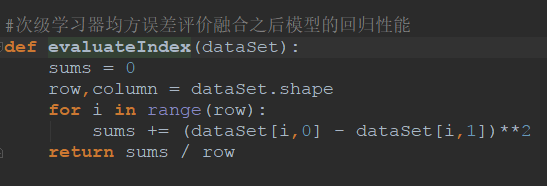
(3) 形成次级学习器的测试集



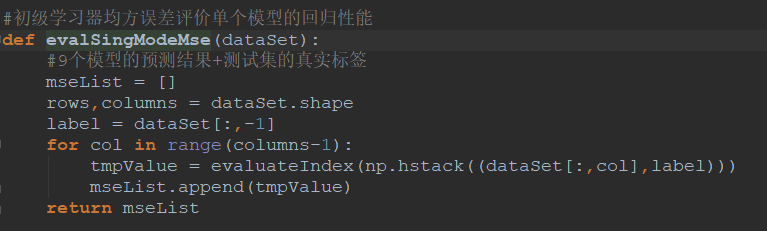
(4) 训练次级学习器，并且用次级测试集测试，然后返回测试结果



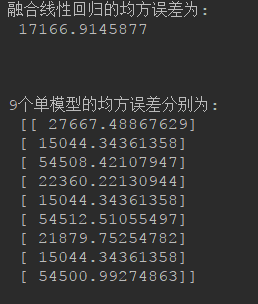
(5) 计算次级学习器(融合模型)均方误差



(6) 计算9个单模型的均方误差



## 3.程序最终执行结果(总共9个单模型)



## 4.总结

当生成次级训练集的时候，不能直接用训练初级学习器的集合直接生成次级训练集，必须对初级训练集进行交叉预测，以防止过拟合现象的出现。由于stack模型性能可能不是太好，所以有可能单个学习的性能要优于融合之后的学习器，但是若BaseModel属于弱学习器，则融合之后的学习器的性能要好于单个学习器的性能。