Week7

一、模型集成

1.简介

通过训练多个弱学习器模型(模型)解决相同的问题，将弱学习器的偏差和(或)方差结合起来，从而创建一个强学习器(集成模型),获得更好的性能。
在基于 SQuAD 的阅读理解任务中，集成模型效果排在前三位。

2.模型集成的方法

a.基于投票思想的多数票机制：即少数服从多数，对训练的多个弱分类器的输出结果进行投票，该样本的最终分类预测取投票数最多的那个预测。

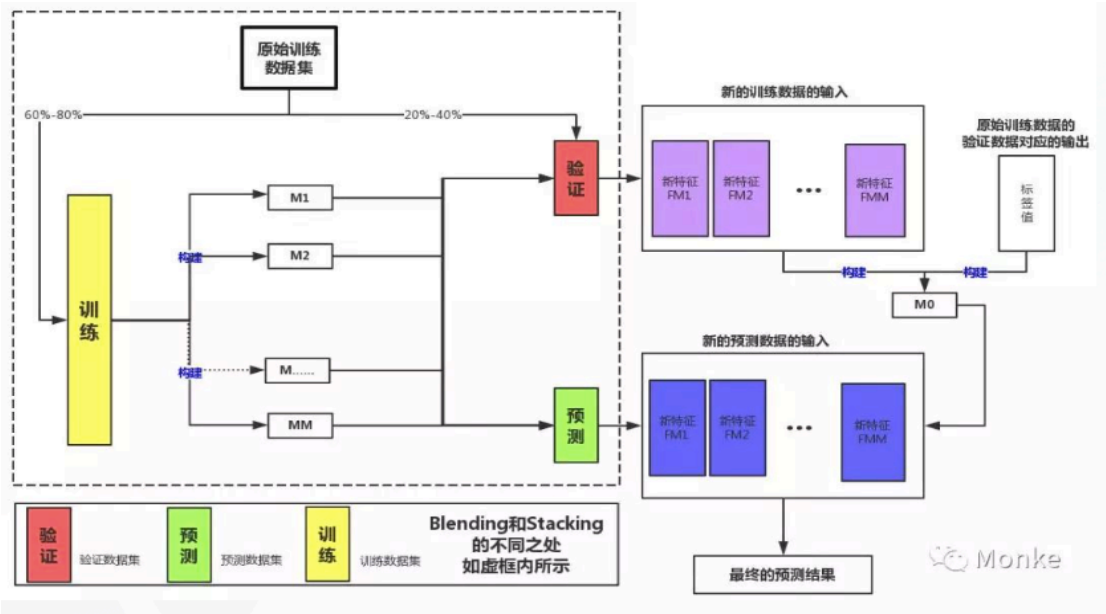
b.基于 bagging 思想的套袋集成技术：相比于 a 方法在训练每个分类器的时候均使用相同的全部样本，bagging 方法是通过随机采样的方法(如有放回采样)，对每个分类器使用不同的样本进行训练，生成众多并行式分类器，最终通过‘少数服从多数’的原则(回归的结果取其平均值)来确定最终结果；

c.基于 boosting 思想的自适应增强方法：通过将弱学器提升为强学习器的集成方法(按顺序学习，串行)来提高预测精度，即降低偏差，提高弱分类器的性能，如 Adaboost 和 GBDT。
GBDT：由多颗 CART 回归树组成，将所有树的结果累加起来，作为最终结果，每棵树学的是之前所有树结果和的残差。每次迭代都在减少残差的梯度方向新建立一颗决策树，迭代多少次就会生成多少颗决策树；
Adaboost：自适应增强方法，学的是错误样本权重，通过加大错误样本的权重(对错误样本加大惩罚力度)来更新权重；

d.分层模型集成框架 stacking(叠加算法)：可以理解为一个两层的集成，第一层含有一个分类器，把预测的结果(元特征)提供给第二层，而第二层把第一层分类器的结果当作特征做拟合输出预测结果。如推荐系统中的 GBDT+LR，先通过 GBDT 进行新特征构造(特征提取),然后把新特征作为第二层 LR 模型的输入，进行回归预测，得到预测结果。集成框架 stacking 见图一；

3.Blending

为了解决 Stacking 在交叉验证阶段出现的数据泄露，训练集不是通过 K-Fold 的 CV 策略来获得预测值从而生成第二阶段模型的特征，而是建立一个 Holdout 集，如 70%的训练数据，第二阶段的 stacker 模型就基于第一阶段模型对这 70%训练数据的预测值进行拟合。结构如下：



图一(集成框架 stacking)