



TAIJI LABORATORY
FOR GRAVITATIONAL WAVE UNIVERSE



ICTP-AP
International Centre
for Theoretical Physics Asia-Pacific
国际理论物理中心-亚太地区



中国科学院大学
University of Chinese Academy of Sciences

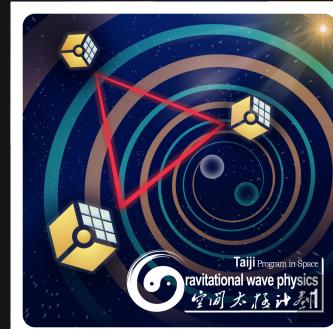
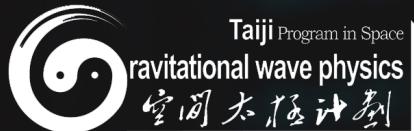
引力波数据探索：编程与分析实战训练营

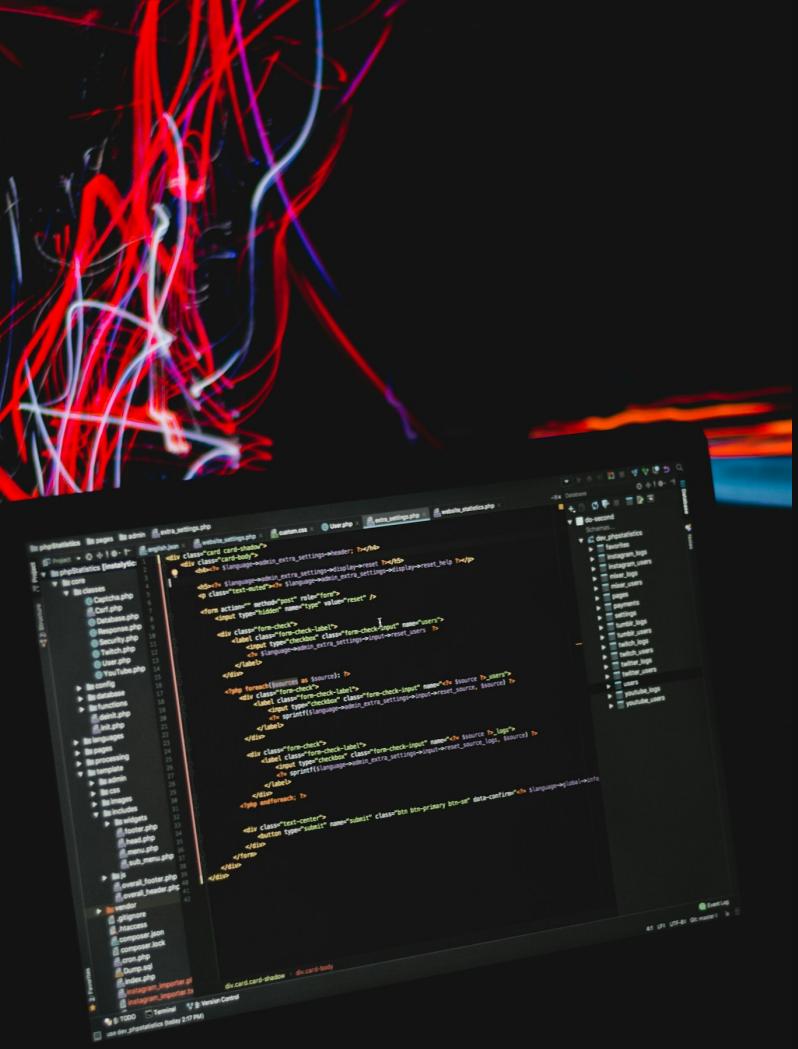
第3部分 机器学习基础 机器学习算法之应用进阶

主讲老师：王赫

ICTP-AP, UCAS

2023/12/24





引力波数据分析与机器学习

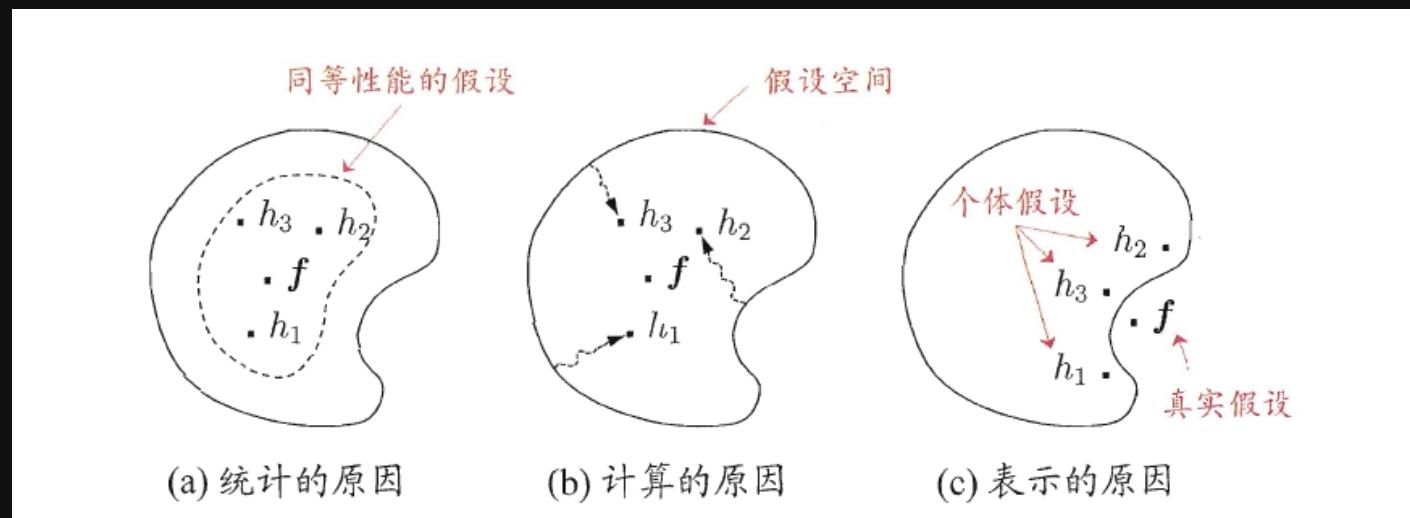
- 机器学习中的模型调优与模型融合
 - 交叉验证
 - 网格搜索
 - 集成学习
 - 实战项目：基于 LIGO 的 Glitch 元数据完成多分类任务
 - 实战项目：基于 LIGO 的 Glitch 时频图数据实现聚类分析



机器学习中的模型融合

- **模型融合 (Model Ensemble)** 是一种机器学习策略，它结合了多个模型的预测结果，以获得比单个模型更好的预测性能。模型融合的基本思想是通过**集成学习 (Ensemble Learning)** 的方式，利用一组模型的多样性，减少模型的偏差 (Bias) 和方差 (Variance)，从而提高模型的泛化能力。

1. 从**统计**方面看，由于学习任务的假设空间往往很大，可能有多个假设在训练集上达到同样性能，此时若使用单学习器可能因误选而导致泛化性能不佳，结合多个学习器则会减小这一风险
2. 从**计算**的方面来看，学习算法往往会陷入局部极小，有的局部极小点对应的泛化性能可能很糟糕，而通过多次运行之后进行结合，可降低陷入糟糕局部极小点的风险
3. 从**表示**的方面来看，某些学习任务的真实假设可能不在当前学习算法所考虑的假设空间中，此时若使用单学习器则肯定无效，而通过结合多个学习器，由于相应的假设空间有所扩大，有可能学得更好的近似



- 模型融合中的模型可以是同质的，也可以是异质的：
 1. **同质**模型 (Base Learner)：这些模型都是同一类型的，例如都是决策树或者都是神经网络。同质模型通常通过引入随机性来增加多样性，例如在随机森林中，我们通过自助采样 (Bootstrap Sampling) 和随机特征选择来生成多个不同的决策树
 2. **异质**模型 (Component Learner)：这些模型是不同类型的，例如一部分是决策树，一部分是神经网络。异质模型通常通过结合不同类型模型的优点来增加多样性，例如在堆叠泛化 (Stacking) 中，我们可以使用不同类型的模型作为基模型，然后使用另一个模型（元模型）来结合这些基模型的预测结果。



模型融合有什么要求？

模型融合若想取得较好的效果，需要个体学习器“**好而不同**”。

“好”指的是个体学习器需要有一定的准确性，“不同”指的是学习器间具有差异。模型的不同可以体现在：

1. **不同训练数据**: 数据集使用比例 (e.g. bootstrap、不同特征) 、预处理方法 (缺失值填补、特征工程)
2. **不同模型结构**: RF、XGBoost、LightGBM、CNN、LSTM等
3. **不同超参数**: 随机种子数、权重初始化、收敛相关参数 (如学习率、batch size、epoch、early stop) 、损失函数、子采样比例等

下图中西瓜书中的实例对该问题进行了说明，采用简单投票的方法确定融合模型的结果，图 (a) 子模型满足“好”和“不同”两个条件，图 (b) 中子模型不满足“不同”的条件，图 (c) 子模型不满足“好”的条件。

测试例1 测试例2 测试例3			测试例1 测试例2 测试例3			测试例1 测试例2 测试例3					
h_1	√	√	×	h_1	√	√	×	h_1	√	×	×
h_2	×	√	√	h_2	√	√	×	h_2	×	√	×
h_3	√	×	√	h_3	√	√	×	h_3	×	×	√
集成	√	√	√	集成	√	√	×	集成	×	×	×

(a) 集成提升性能 (b) 集成不起作用 (c) 集成起负作用

图 8.2 集成个体应“好而不同” (h_i 表示第 i 个分类器)

简单说来，我们信奉几个信条：

1. **群众的力量是伟大的，集体智慧是惊人的**

- 投票 (Voting)
- 装袋 (Bagging)
- 随机森林 (Random forest)

2. **站在巨人的肩膀上，能看得更远**

- 堆叠 (Stacking)
- Blending

3. **一万小时定律**

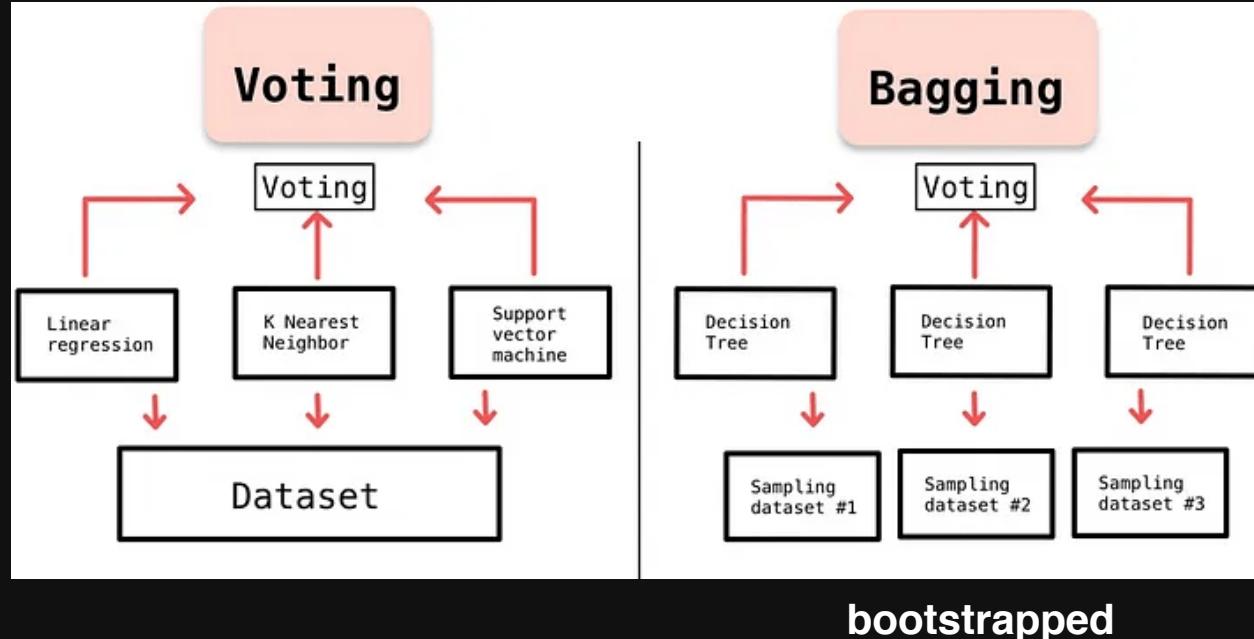
- Boosting

- 模型融合的常见方法包括投票 (Voting) 、装袋 (Bagging) 、提升 (Boosting) 和堆叠 (Stacking) 。这些方法都有各自的优点和适用场景，例如投票和装袋可以减少模型的方差，提升可以减少模型的偏差，堆叠泛化可以结合多个模型的优点。
- 总的来说，模型融合是一种强大的机器学习策略，它可以有效地提高模型的预测性能，特别是在处理复杂的非线性问题时。



Voting 与 Bagging (并行)

<https://medium.com/@chyun55555/ensemble-learning-voting-and-bagging-with-python-40de683b8ff0>



Voting: 单个模型很难控制过拟合。

- 那就多数表决

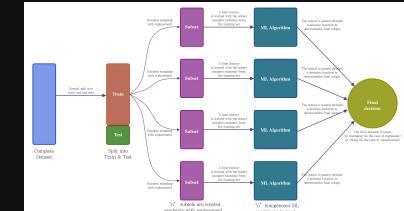
Bagging: 模型效果不好的原因是过拟合?

- 那就少给点题, 别让它直接把所有题目的答案背下来
- 多找几个同学来分配题目做题, 综合一下他们的答案

简单说来, 我们信奉几个信条:

1. 群众的力量是伟大的, 集体智慧是惊人的
 - 投票 (Voting)
 - 装袋 (Bagging)
 - 随机森林 (Random forest)
2. 站在巨人的肩膀上, 能看得更远
 - 堆叠 (Stacking)
 - Blending
3. 一万小时定律
 - Boosting

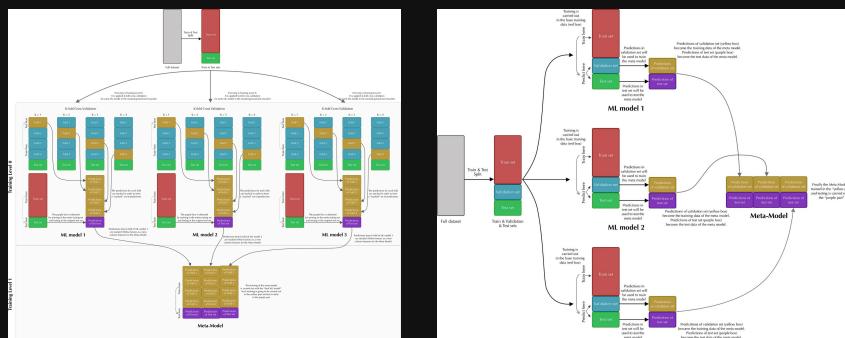
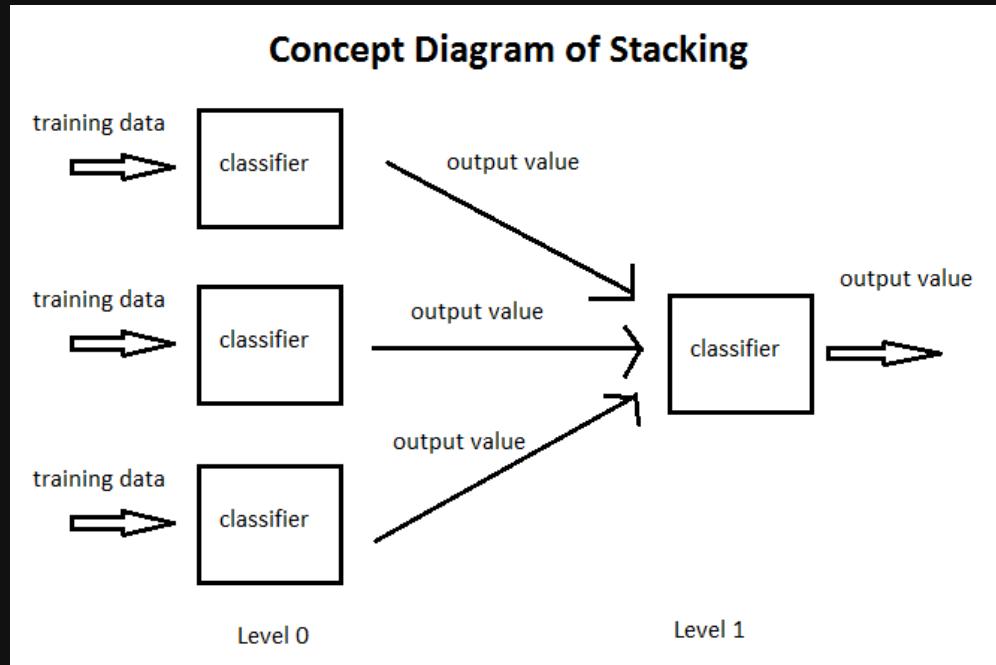
<https://towardsdatascience.com/ensemble-learning-bagging-boosting-3098079e5422>



- 随机森林 (Random forest) 是一种基于树模型的Bagging的优化版本。对每棵树构建的时候, 特征也会做采样处理。



Stacking 与 Blending (串行)



<https://www.kaggle.com/code/jeonghojae/blending-and-stacking/notebook>

<https://towardsdatascience.com/ensemble-learning-stacking-blending-voting-b37737c4f483>

简单说来，我们信奉几个信条：

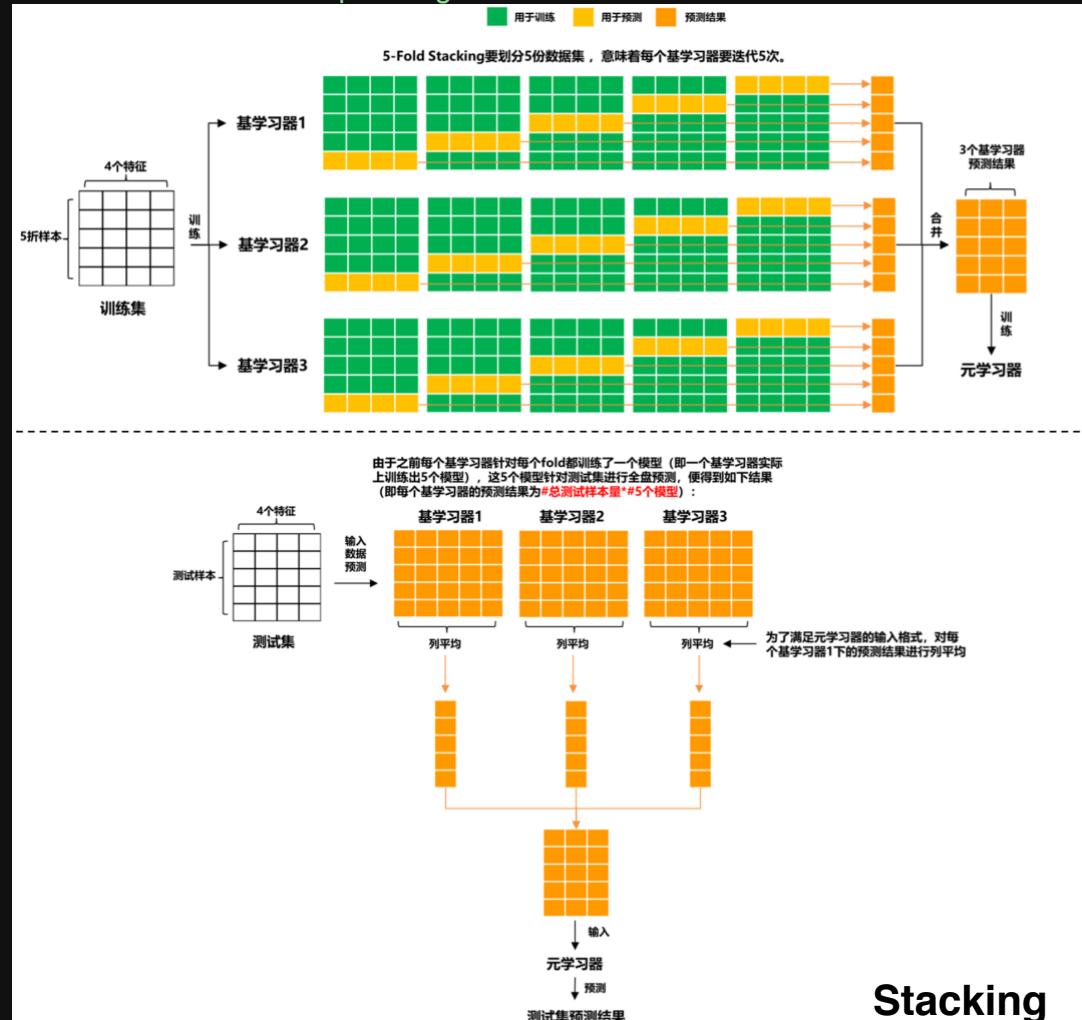
1. 群众的力量是伟大的，集体智慧是惊人的
 - 投票 (Voting)
 - 装袋 (Bagging)
 - 随机森林 (Random forest)
 2. 站在巨人的肩膀上，能看得更远
 - 堆叠 (Stacking)
 - Blending
 3. 一万小时定律
 - Boosting

- Stacking的思路是，基于原始数据，训练多个基学习器，然后将基学习器的预测结果组合成新的训练集，去训练新的学习器。



Stacking 与 Blending (串行)

<https://blog.csdn.net/datawhale/article/details/120108280>

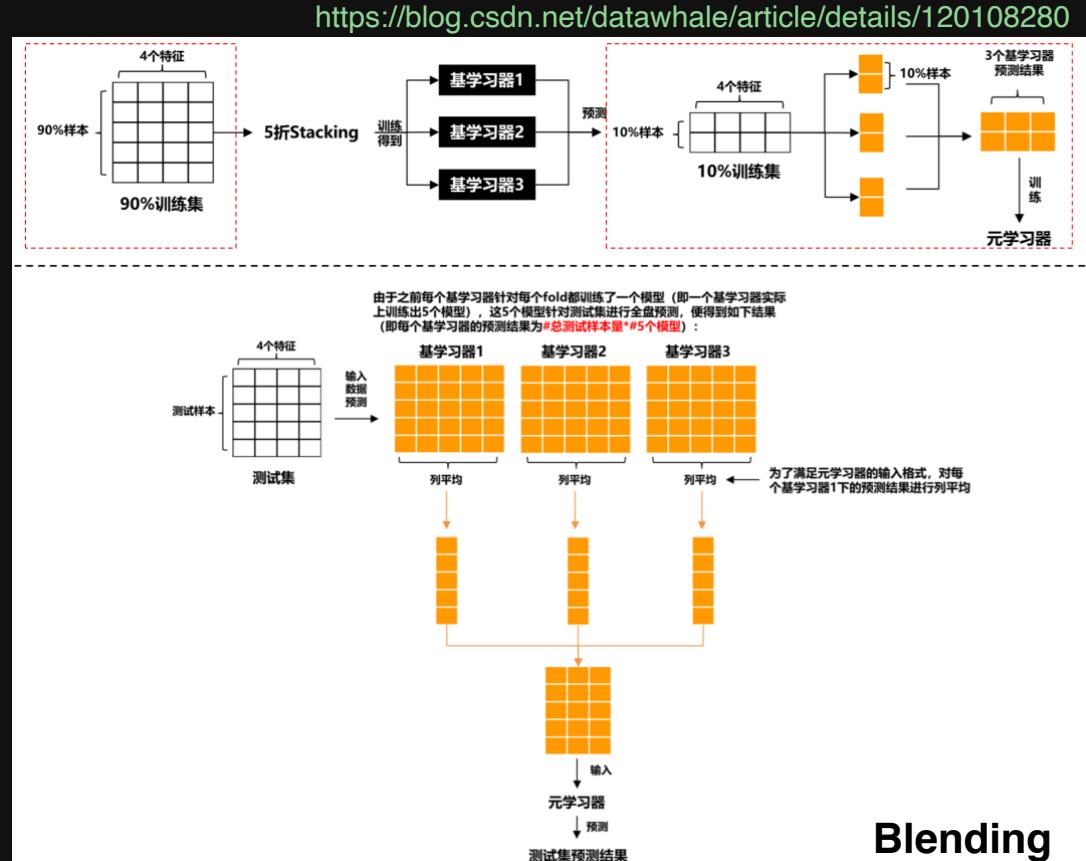


简单说来，我们信奉几个信条：

1. 群众的力量是伟大的，集体智慧是惊人的
 - 投票 (Voting)
 - 装袋 (Bagging)
 - 随机森林 (Random forest)
 2. 站在巨人的肩膀上，能看得更远
 - 堆叠 (Stacking)
 - Blending
 3. 一万小时定律
 - Boosting
- Stacking的思路是，基于原始数据，训练多个基学习器，然后将基学习器的预测结果组合成新的训练集，去训练新的学习器。



Stacking 与 Blending (串行)



简单说来，我们信奉几个信条：

1. 群众的力量是伟大的，集体智慧是惊人的

- 投票 (Voting)
- 装袋 (Bagging)
- 随机森林 (Random forest)

2. 站在巨人的肩膀上，能看得更远

- 堆叠 (Stacking)
- Blending

3. 一万小时定律

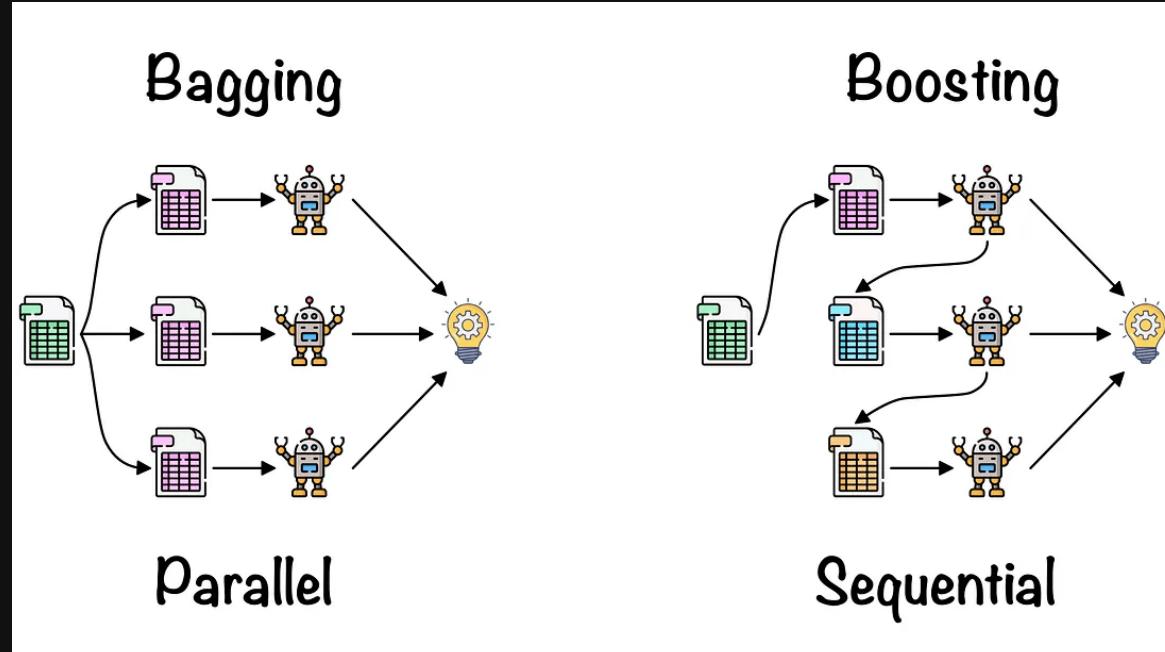
- Boosting

- Blending的想法是：对原始数据集，先划分出一个较小的留出集，比如10%训练集被当作留出集，那么Blending用90%的数据做基学习器的训练，而10%留出集用于训练元学习器，这样基学习器和元学习器是使用不同的数据来训练的。
- 相较于Stacking (虽然输入的x不一样，但标签y一样)，Blending能更有效地防止信息泄露，但也正因为如此，元学习器只用了较小部分的数据集进行训练，且容易对留出集过拟合。

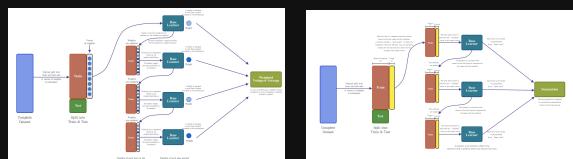
- 上图的红框是区分Stacking的关键。



Boosting (串行)



<https://towardsdatascience.com/ensemble-learning-bagging-boosting-3098079e5422>



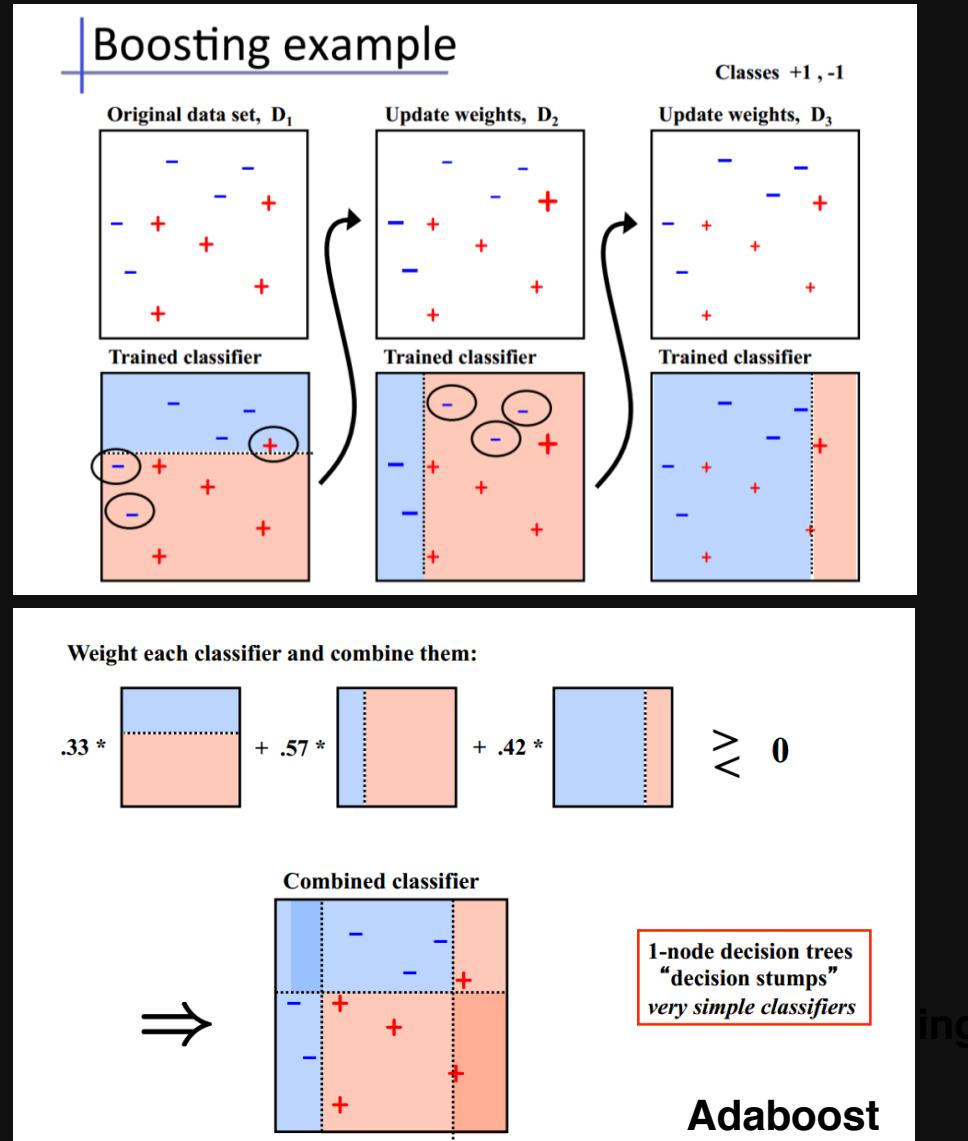
简单说来，我们信奉几个信条：

1. 群众的力量是伟大的，集体智慧是惊人的
 - 投票 (Voting)
 - 装袋 (Bagging)
 - 随机森林 (Random forest)
2. 站在巨人的肩膀上，能看得更远
 - 堆叠 (Stacking)
 - Blending
3. 一万小时定律
 - Boosting

- 考得不好的原因是什么？
 - 还不够努力，练习题要多次学习
 - 重复迭代和训练
 - 时间分配要合理，要多练习之前做错的题
 - 每次分配给预测错的样本更高的权重
 - 我不聪明，但是脚踏实地，用最简单的知识不断积累，称为专家
 - 最简单的分类器的叠加



Boosting (串行)



简单说来，我们信奉几个信条：

1. 群众的力量是伟大的，集体智慧是惊人的

- 投票 (Voting)
- 装袋 (Bagging)
- 随机森林 (Random forest)

2. 站在巨人的肩膀上，能看得更远

- 堆叠 (Stacking)
- Blending

3. 一万小时定律

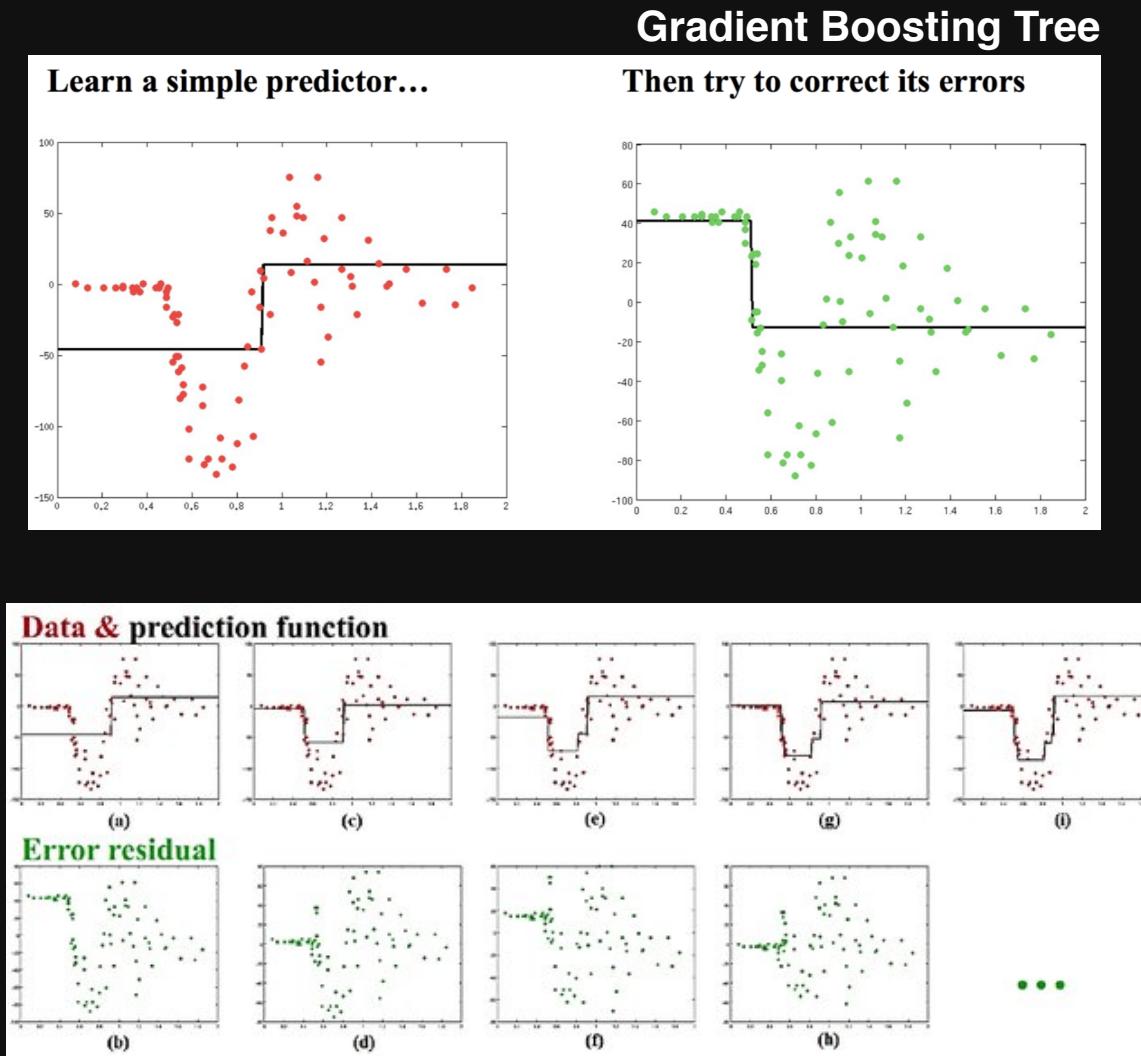
- Boosting

• 基础思想：Boosting是一种串行的工作机制，即个体学习器的训练存在依赖关系，必须一步一步序列化进行。Boosting是一个序列化的过程，后续模型会矫正之前模型的预测结果。也就是说，之后的模型依赖于之前的模型。

• 其基本思想是：增加前一个基学习器在训练过程中预测错误样本的权重，使得后续基学习器更加关注这些打标错误的训练样本，尽可能纠正这些错误，一直向下串行直至产生需要的T个基学习器，Boosting最终对这T个学习器进行加权结合，产生学习器委员会。



Boosting (串行)



简单说来，我们信奉几个信条：

1. 群众的力量是伟大的，集体智慧是惊人的
 - 投票 (Voting)
 - 装袋 (Bagging)
 - 随机森林 (Random forest)
 2. 站在巨人的肩膀上，能看得更远
 - 堆叠 (Stacking)
 - Blending
 3. 一万小时定律
 - Boosting
- 和 Adaboost 思路类似，解决回归问题



集成学习：Bagging vs Boosting

	learner弱依赖Methods eg.Bagging	learner强依赖Methods eg.Boosting
方法	1.部分数据/部分参数/1或N个算法训练model 2.上述多个model的组合	1.训练基础算法，后续算法利用前面算法结果重点处理错误case 2.上述多个stage的组合
流程	<p>$\{w_n^{(1)}\}$ $\{w_n^{(2)}\}$... $\{w_n^{(M)}\}$</p> <p>$y_1(\mathbf{x})$ $y_2(\mathbf{x})$... $y_M(\mathbf{x})$</p> $Y_M(\mathbf{x}) = \text{sign} \left(\sum_m \alpha_m y_m(\mathbf{x}) \right)$	<p>$\{w_n^{(1)}\}$ $\{w_n^{(2)}\}$... $\{w_n^{(M)}\}$</p> <p>$y_1(\mathbf{x})$ $y_2(\mathbf{x})$... $y_M(\mathbf{x})$</p> $Y_M(\mathbf{x}) = \text{sign} \left(\sum_m \alpha_m y_m(\mathbf{x}) \right)$
偏差-方差分析	Bagging主要关注降低方差 因此在不剪枝DT, Neural Network等易受样本扰动影响learner效果更明显	Boosting主要关注降低偏差 因此Boosting基于泛化能力相当弱的learner构建很强的集成
适用范围	高噪声	低噪声
串行并行	并行 Bagging的各个预测函数没有权重,各个预测函数可以并行生成	串行 Boosting是有权重的,各个预测函数只能顺序生成
样例	Random Forest	AdaBoost GDBT

Blending

- Ensemble Methods: Foundations and Algorithms by Ralf Herbrich and Thore Graepel
 - <https://tjzhifei.github.io/links/EMFA.pdf>

Repo of the course: <https://github.com/iphysresearch/GWData-Bootcamp>

Homework

- 基础及拓展作业:
 - 一起来打怪之 Credit Scoring 练习:
[homework_credit_scoring_finetune_ensemble.ipynb](#)
 - 在 homework 分支上 PR。

通向自我实现之路: Kaggle

- Kaggle是一个由Google所有的数据科学和机器学习竞赛平台。它为数据科学家和机器学习工程师提供了一个可以共享和协作的环境，用户可以在平台上找到并发布数据集，探索和构建模型，运行数据科学工作流，以及参加各种机器学习竞赛。<https://www.kaggle.com>
- Kaggle的主要特点包括：
 1. 竞赛：Kaggle举办了许多由企业和研究机构赞助的机器学习竞赛。这些竞赛涵盖了各种问题，从图像分类到自然语言处理，参赛者可以通过解决实际问题来提升自己的技能。
 2. 数据集：Kaggle拥有一个庞大的公开数据集库，用户可以在这里找到各种类型的数据集，并可以上传自己的数据集供他人使用。
 3. Kernels：Kaggle的Kernels是一种共享代码的方式，用户可以在Kernels中编写代码，进行数据分析和建模，并将其分享给其他用户。
 4. 社区：Kaggle有一个活跃的社区，用户可以在论坛上讨论问题，分享想法和经验，以及学习新的技术和方法。
 5. 学习：Kaggle还提供了一系列的数据科学和机器学习教程，帮助用户学习新的技能和知识。
- 总的来说，无论你是数据科学的新手，还是经验丰富的专家，Kaggle都是一个学习，实践，和分享知识的好地方。

强劲算力 触手可达

ICTP-AP引力波 x Sugon-但易

邀请您免费试用 90 天

1000 卡时

华东一区【昆山】

7185-32C-128G-4卡

加速计算

处理器 1*7185 32C 2.0GHz

内存 128GB

计算网络 200Gb

立即试用

咨询

QR code

• 赞助单位

• 中科曙光

曙光智算

计算服务 | Sugon

13