

Part
03

梯度下降树

- 有没有更加聪明的方法
- AdaBoost
- 梯度下降树



从现实世界说起

- 融合算法本质上是将数据给模型重新学习
- 就像我们小时候写作业那样
- 但是只会写作业的小朋友好像总不是学习最好的那个

进行，反应室之间有无摩擦、可滑动的密封隔板，反应开始和达到平衡时如图所示：

平衡(I) 平衡(II)

反应：强与反应开始时体系的压强之比为 14:15
 浓度为 $\frac{5}{11}$
 中 M 的体积分数相等

已知：左右两室后体积相等 $PV=nRT$
 $P_1V_1=P_2V_2$ 不可知压强是否相等
 $=\frac{11}{14}$
 $=P_2$

对左室而言：
 左右两室 P 相同
 $\frac{P_1}{P_2} = \frac{n_1}{n_2} = \frac{14}{15}$
 平衡时 $\frac{2.2}{2.2} = \frac{11}{15}$
 $n_2 = \frac{2.2}{1.1} = 2$
 $\therefore n_1 = 2.2$

对右室而言：
 $\frac{P_1}{P_2} = \frac{n_1}{n_2} = \frac{14}{15}$
 $\frac{P_1}{P_2} = \frac{n_1}{n_2} = \frac{14}{15}$
 $\therefore n_1 = 2.2$

24. 在一定条件下，可逆反应： $2X(g) \rightleftharpoons 2Y(g) + Z(g)$ ； $\Delta H < 0$ ，在 t_1 时刻开始加热，到一定温度后停止加热并保温，到 t_2 时刻又降温，变化的是 ()

① 加热 平衡向何方移动
 ② 从加入 X 开始反应，各物质浓度变化 B
 ③ 从加入 Y、Z 开始反应，各物质浓度变化 D

(2010 年四川理科) 反应 $aM(g) + bN(g) \rightleftharpoons cP(g) + dQ(g)$ 达到平衡时与反应条件的关系如图所示，其中 x 表示反应开始时 N 的物质的量与下列说法正确的是 ()

平衡时 P 的体积分数与反应条件的关系图

A. 同温同压时，加入催化剂，平衡时 Q 的体积分数增加
 B. 同压同温时，升高温度，平衡时 Q 的体积分数增加

Algorithm 10.1 *AdaBoost.M1.*

1. Initialize the observation weights $w_i = 1/N$, $i = 1, 2, \dots, N$.

2. For $m = 1$ to M :

(a) Fit a classifier $G_m(x)$ to the training data using weights w_i .

(b) Compute

$$\text{err}_m = \frac{\sum_{i=1}^N w_i I(y_i \neq G_m(x_i))}{\sum_{i=1}^N w_i}.$$

(c) Compute $\alpha_m = \log((1 - \text{err}_m)/\text{err}_m)$.

(d) Set $w_i \leftarrow w_i \cdot \exp[\alpha_m \cdot I(y_i \neq G_m(x_i))]$, $i = 1, 2, \dots, N$.

3. Output $G(x) = \text{sign} \left[\sum_{m=1}^M \alpha_m G_m(x) \right]$.

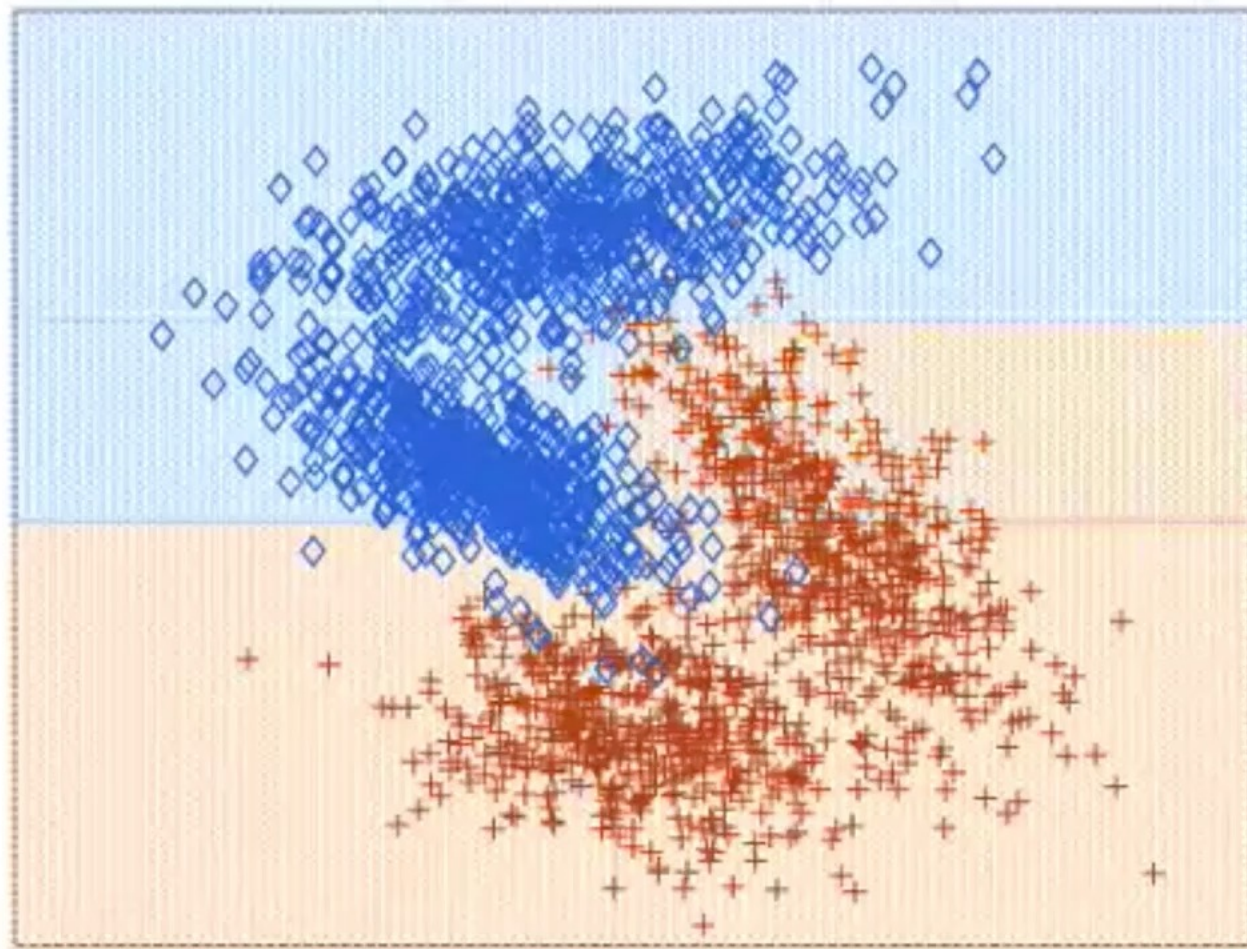
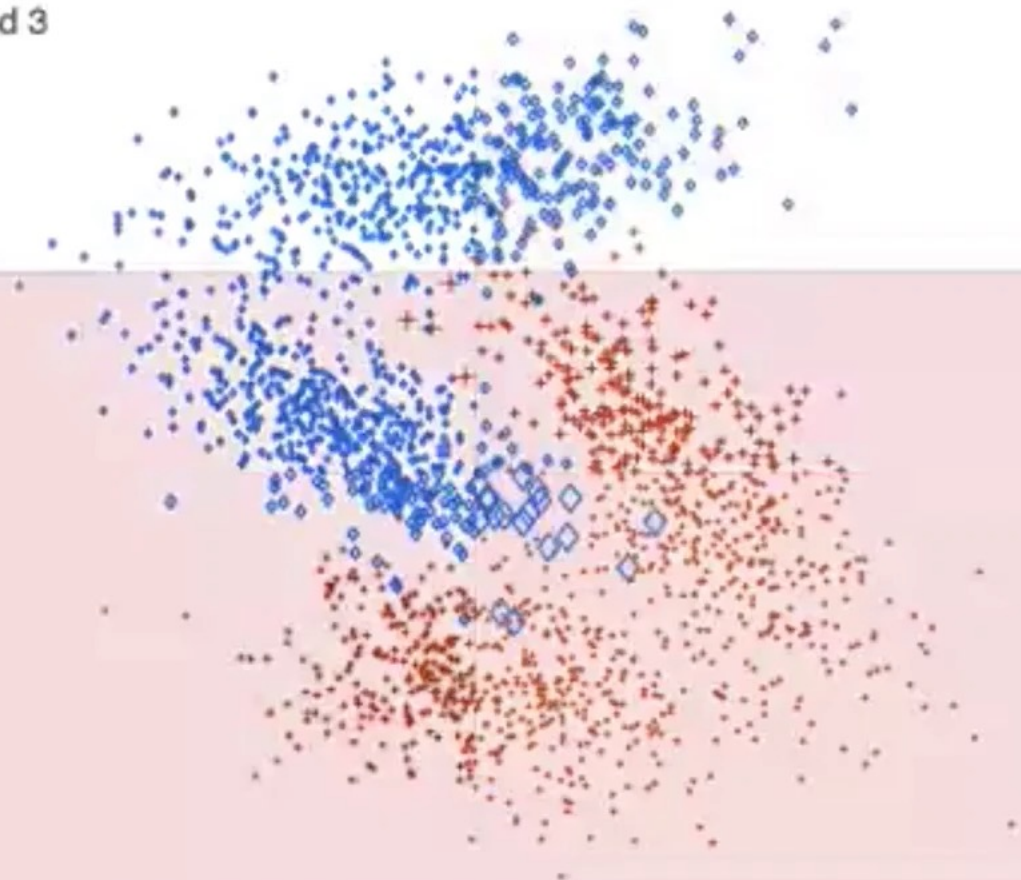
- AdaBoost算法 (ESL)
- i 为数据标号, w 为权重
- M 为总轮次, m 为某轮
- err_m 为某轮的错误率
- α_m 为权重更新参数
 - err_m 要大于0.5
 - 模型做错的越少, 参数越大
- 只对错误的样本进行更新
- 最后按照正确率加权输出



3.1

从随机的个体，到有迹可循的过程

Round 3

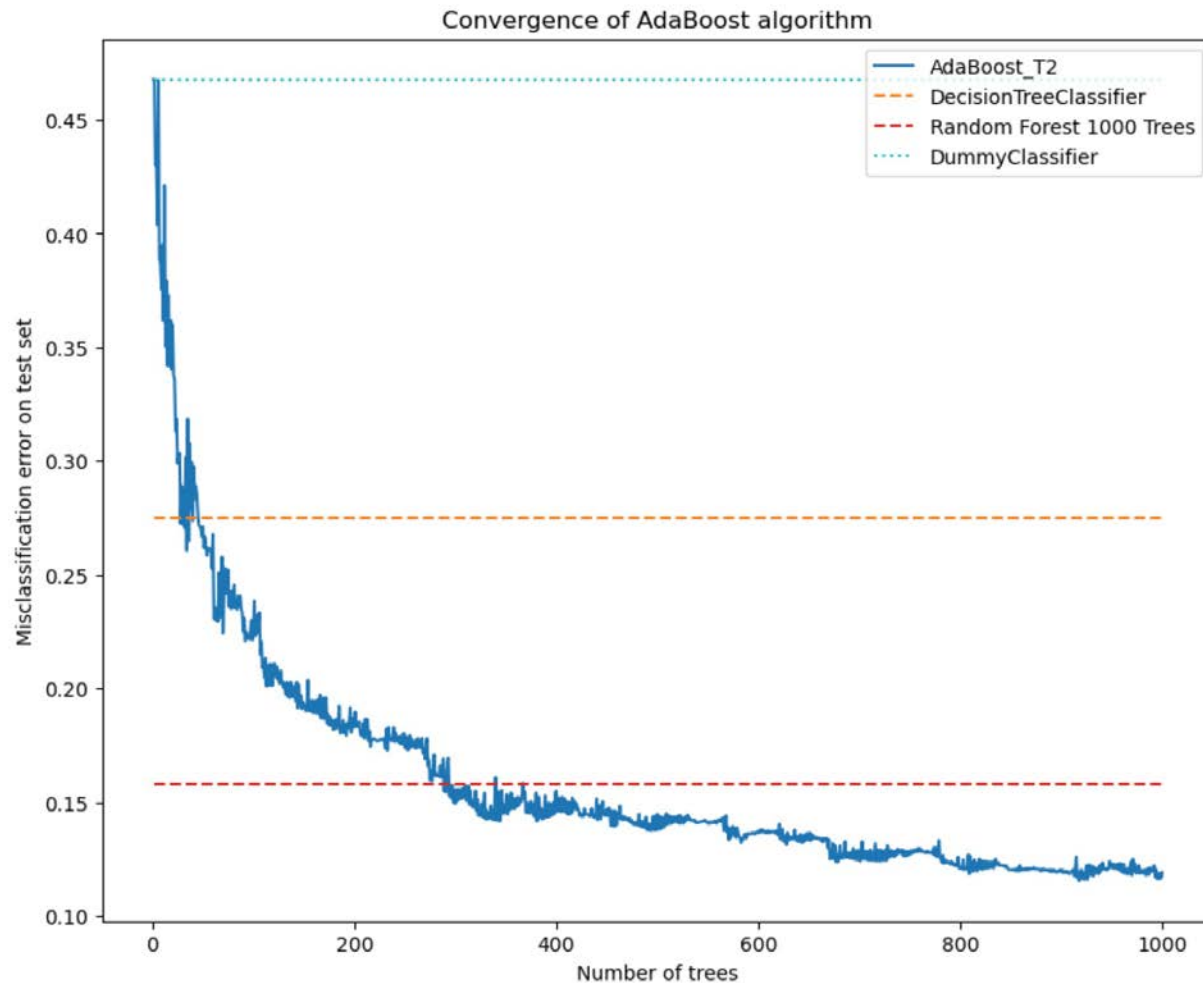


AdaBoost例子：ESL 10.2 扩展

- X : 十个高斯分布的随机变量
- X_1, X_2, \dots, X_{10}

$$Y = \begin{cases} 1 & \text{if } \sum_{j=1}^{10} X_j^2 > \chi_{10}^2(0.5), \\ -1 & \text{otherwise.} \end{cases}$$

- 4000条数据, $\text{len}(y[y==1]) = 1986$
- 四种算法:
 - 单层决策树
 - 无限制决策树
 - 单层决策树迭代1000次构成AdaBoost
 - 无限制决策树1000棵构成Random Forest



AdaBoost优缺点

- 优点：
 - 对于弱分类器要求很低
 - 不需要知道其能力上限，下限 >0.5 就好
 - 不容易过拟合
 - 是一种算法思想而非特指某种算法
 - 可以并列多种弱分类器
- 缺点：
 - 噪音样本的轮次累积
 - 序列训练&弱分类器“不弱”



【报告】Boosting 25年（2014周志华）（up主推荐）

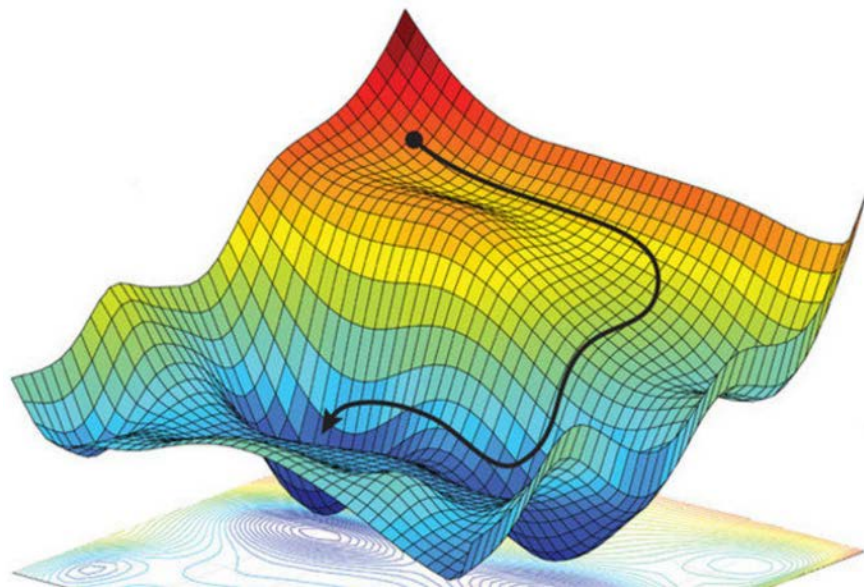
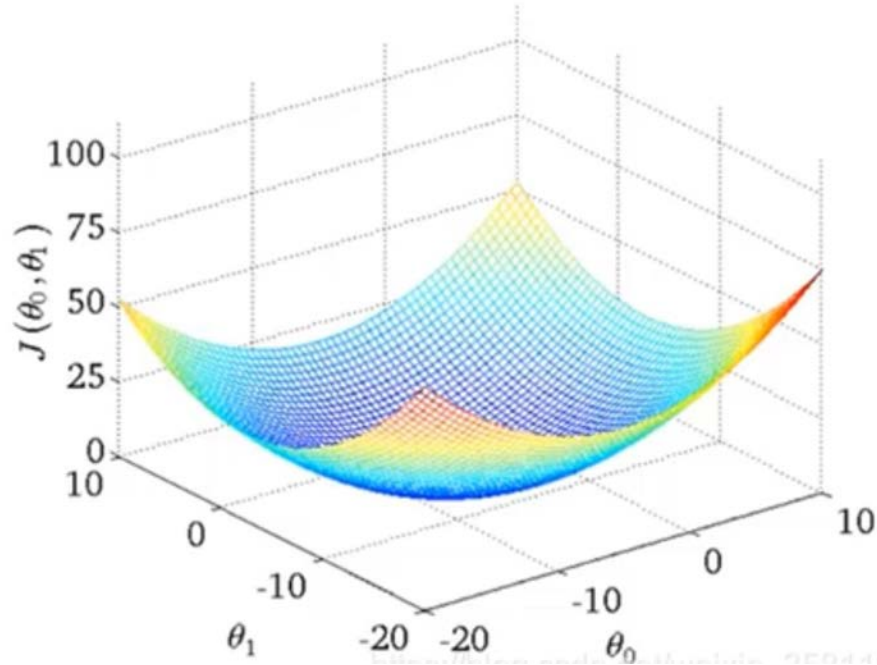
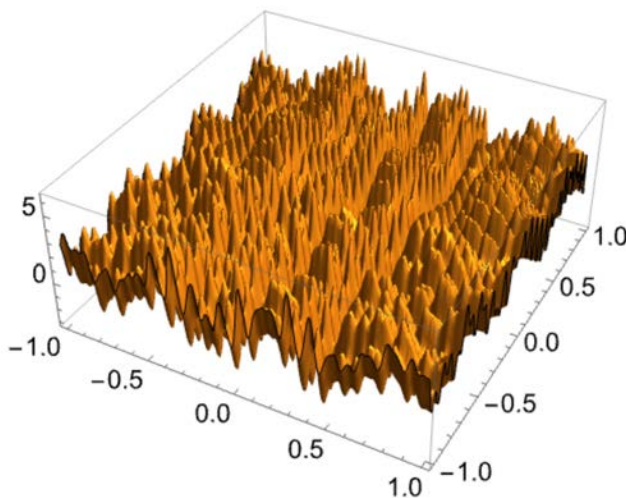
更一般形态的Boosting

- 在每一步的模型记为 $H_t(x)$, 我们定义 $H_0(x) = 0$
- 我们总共训练 T 轮, 对于 $t = 1, 2, \dots, T$
- 我们训练一个模型 h_t , 但是对于**残差建模** $\{(x_i, y_i - H_{t-1}(x_i))\}_{i=1, \dots, N}$
- 更新模型 $H_t = H_{t-1} + \eta h_t$ η 我们称之为**学习率**, (通常) 小于1 (0.1)
- $H_T(x)$ 即为模型的最终输出
- 不难理解, 若记 L_{t-1} 为模型 H_{t-1} 的损失函数, 则 h_t 与 $-\frac{\partial L_{t-1}}{\partial H_{t-1}}$ 平行
- 一般将 $\frac{\partial L_{t-1}}{\partial H_{t-1}}$ 称为梯度, 本方法即称为Gradient Boosting (梯度下降、梯度提升)



梯度下降法 Gradient descent

- 问题回到线性回归 $y = wx + b$,
- $L(w, b) = \sum_{i=1}^N (y_i - (wx_i + b))^2 = f(w, b)$
- 在线性回归中, 我们要求 X 满足一定性质, 从而使得 L 确定唯一的最小值
- 如果 L 不存在数学上的显示解, 如何呢?
- 找一条下山的路:
 - 方向?
 - 步幅?
 - 结束?
- 古圣先哲:
 - 沿着**梯度**的方向
 - 迈小步子
 - **学习率**: 一步迈梯度的一小点

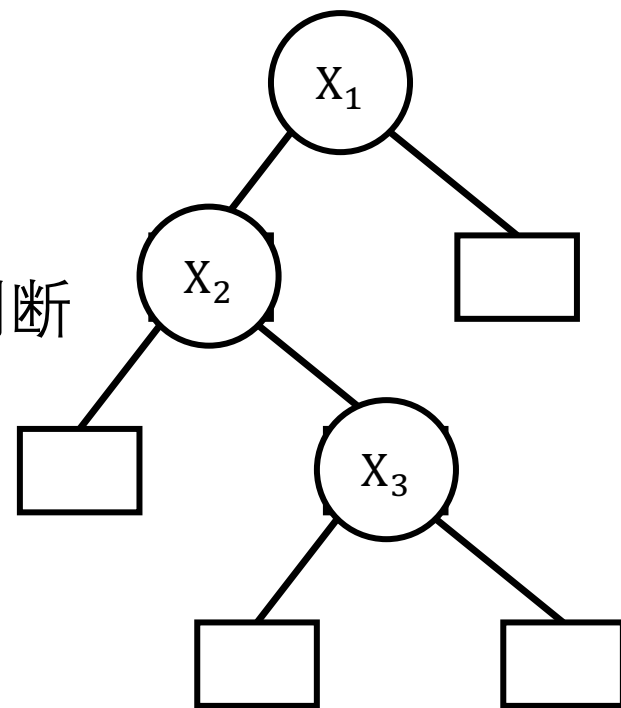


树的规模

- 梯度下降树是许许多多棵小树加和的结果
- 叶子节点数量 (`max_leaf_nodes`, `J`) 意味着什么?
- 对于 $J=2$, 所有的树都是一分为二的, 只针对某1个变量做判断
- 对于 $J=3$, 至多可以将2个变量组合判断
- 对于 $J=4$, 至多可以将3个变量进行组合判断
- 则, 对于 $J=n$, 至多可以将 $n-1$ 个变量进行组合判断

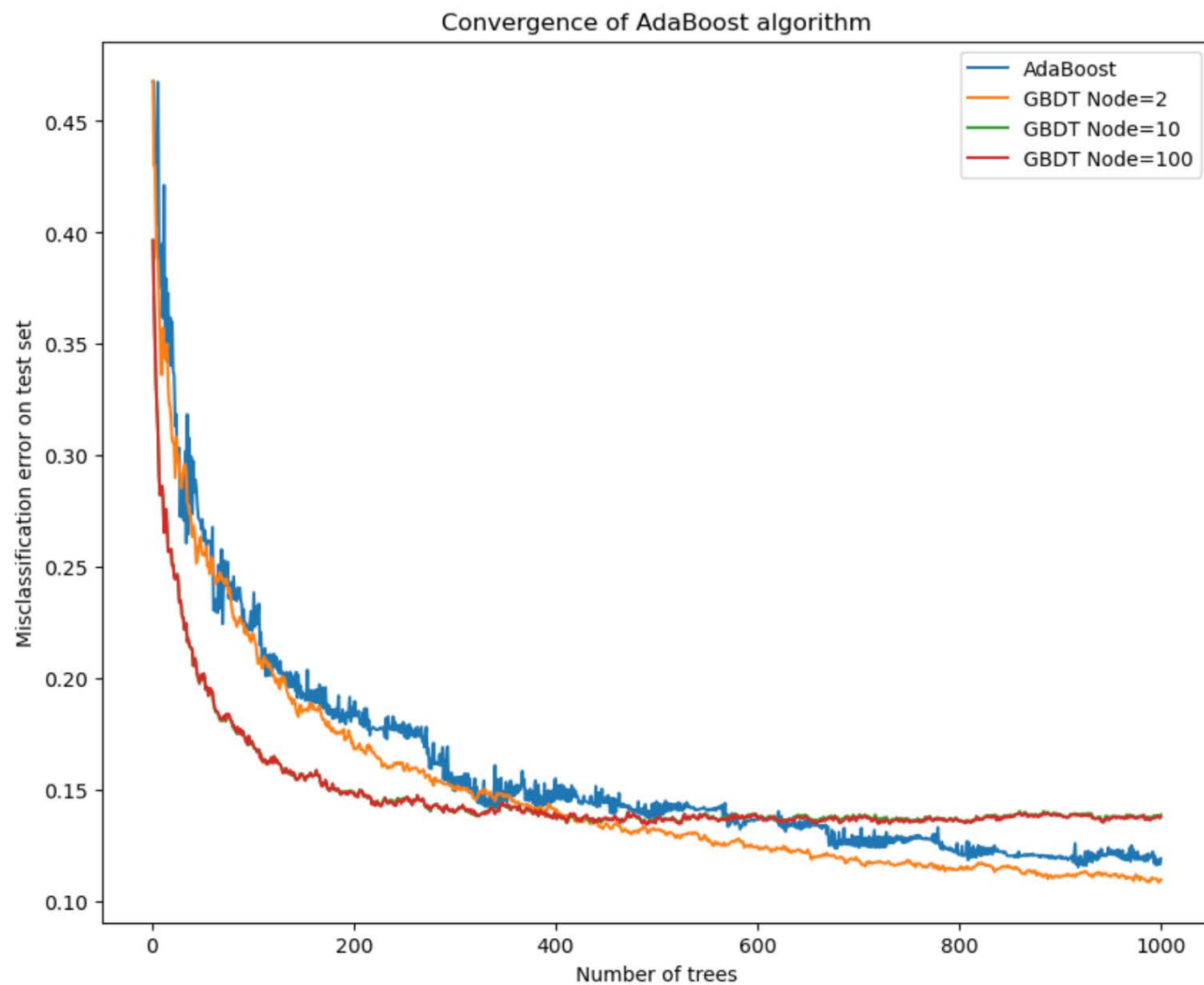
$$\eta(X) = \sum_j \eta_j(X_j) + \sum_{jk} \eta_{jk}(X_j, X_k) + \sum_{jkl} \eta_{jkl}(X_j, X_k, X_l) + \cdots$$

- 一般来说, $4 \leq J \leq 8$, 罕见 $J > 6$ 后仍有提升
- 古圣先哲!



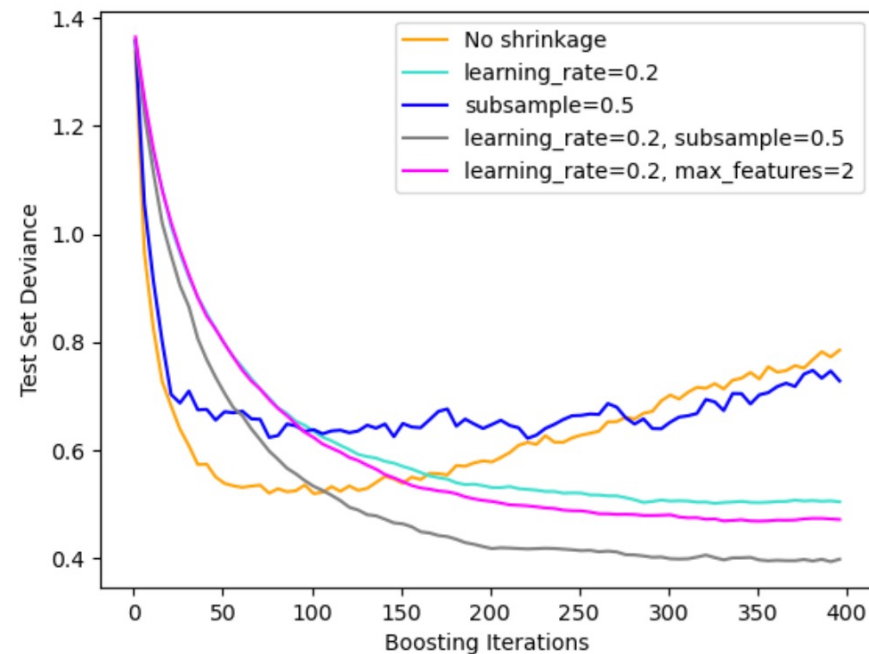
3.2

树的规模



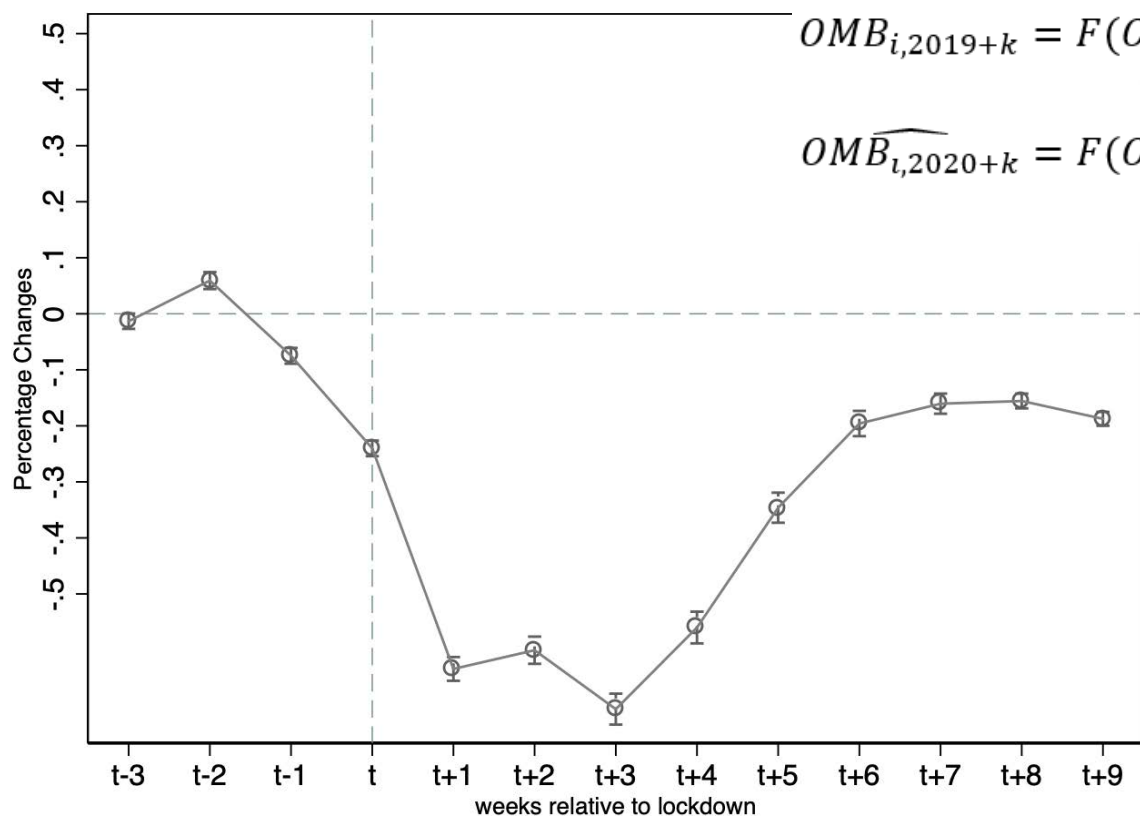
梯度下降树的正则化

- 树的高度：
 - 叶子节点数量 $\text{max_leaf_nodes} = J$
 - $\text{Max_depth} = J-1$
- 树的数量 ($n_estimators$) 要不要限制?
- 为什么要限制? 因为强。为什么强? 因为梯度!
 - 较低的学习率匹配较多的树
- 每一步弱一点
$$H_t = H_{t-1} + \eta h_t$$
 - $\text{max_features}=\text{None}$
 - $\text{Subsample}=1.0$ Friedman, J.H. (2002). [Stochastic gradient boosting.](#)
 - 预测 **Bias** 与 预测 **Variance** 间的权衡
- Early stopping 早点停下
 - 当持续若干轮次 $n_iter_no_change=\text{None}$
 - 在留作 $\text{validation_fraction}=0.1$ 验证的数据上
 - 提升小于 $\text{tol}=1e-4$ 则停止
 - 数据量大于10000, 则自动使用early stopping策略



如何估计2020年疫情中线下微型商户的受损情况

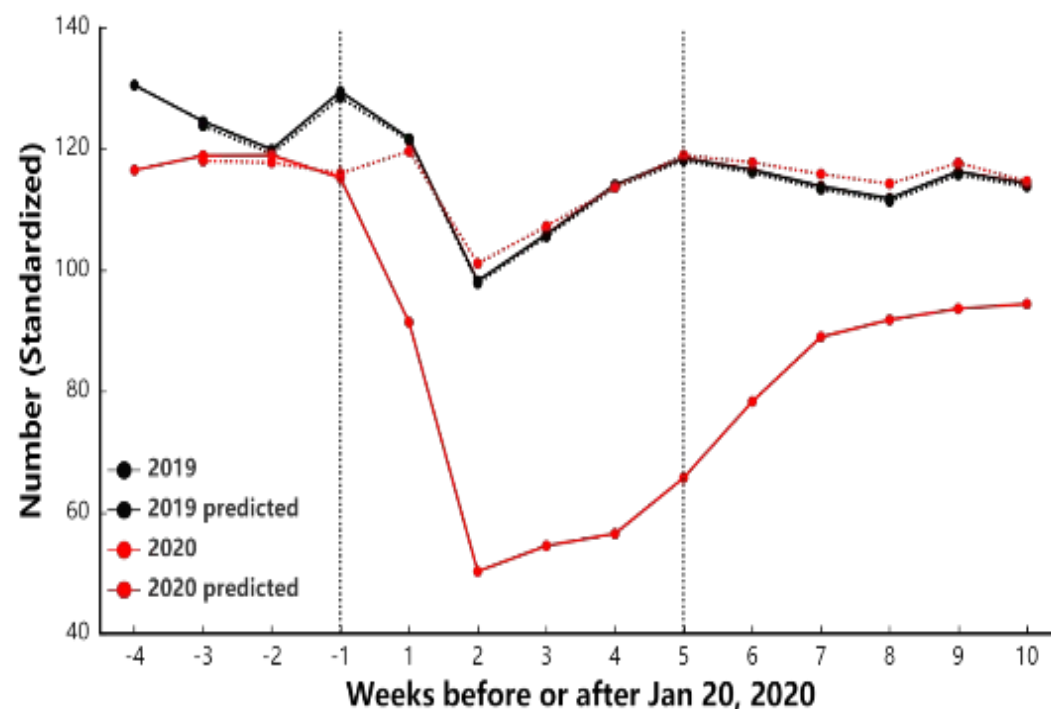
Guo, Feng, et al. "The informal economy at times of COVID-19 pandemic." *China Economic Review* 71 (2022): 101722.



$$OMB_{i,2019+k} = F(OMB_{i,2018+k}, OMB_{i,2018+(k-1)}, OMB_{i,2018-h}, OMB_{i,2019-h}, X_{i,2019+k}, Z_i) \quad (1)$$

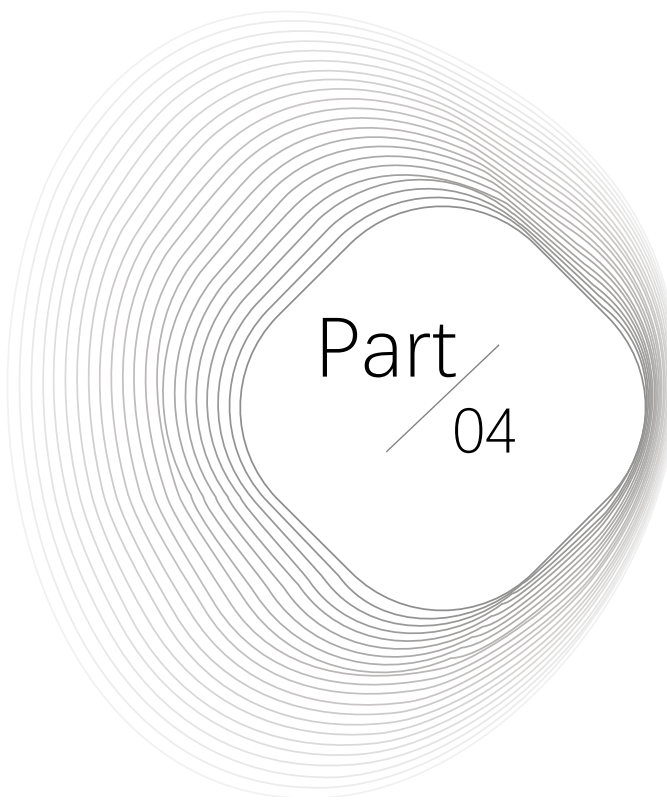
$$\widehat{OMB}_{i,2020+k} = F(OMB_{i,2019+k}, OMB_{i,2019+(k-1)}, OMB_{i,2019-h}, OMB_{i,2020-h}, X_{i,2020+k}, Z_i) \quad (2)$$

Number of Active OMBs



2019-2020活跃码商数渐进did结果





Part
04

总结对比

- 树、森林、随机树
- 多种方法对比



树、森林、梯度树

- 决策树
- 最美的形式
- 高度的灵活性与表示力
- 容易过拟合、容易过敏
- 随机森林
- 鲁棒性很强的算法
- 良好的能力、难过拟合
- 能力有时不够尤其回归
- 简单融合+强个体能力
- Bagging 算法
- 将一个个小的**强**算法
- 通过**简单方式**进行融合
- 当发现一个灵活算法容易过拟合时
- 降低**Variance**
- 梯度下降树
- 极其敏锐的算法
- 担当底牌的能力
- 过拟合、难训练
- 复杂融合+弱个体能力
- Boosting 算法
- 将一个个小的**弱**算法
- 通过**复杂方式**进行融合
- 当发现一个问题难求解时
- 降低**Bias**



	Ridge	Lasso	SVM	RF	GBDT
解析解	存在	不存在 可近似	存在	随机	随机
算法透明度	高	高	高	较低	较低
算法开销	低	低	较低	较高	高
变量数敏感	是	否	否	是， 可降维	是， 可降维
变量选择	否	是， 线性	是	是	是， 但不用
数据缺失值、分类	否	否	否	是	是
算法灵活性	差	差	适中	高	高
变量个数	一般	很高	很高	高	高
特色	简单有效	有效易懂	有效时的首选 大量稀疏变量 理解数据 核的选择很重要	特征选择 高度灵活的关系 好训但上限低	高度灵活 结构化数据的 State-of-art 但难训