第十章 Boosting 算法

AdaBoost 算法

- 一个弱学习算法能否被改造成一个强学习算法？— Michael Kearns
- Schapire 和 Freund 发明了 AdaBoost 算法 (Freund et al., 1999), 它可以对任一做分类的弱学习算法 A 的效果进行增强
- AdaBoost 的解决思路: 对训练集的每个样本用算法 A 产生一系列分类结果，然后巧妙地结合这些输出结果，降低出错率
  - 每次产生新的分类结果时，AdaBoost 会调整训练集的样本权重：**提高**前一轮分类**错误**的样本权重，**降低**前一轮分类**正确**的样本权重

AdaBoost 算法

- Notation
  - $d_{t,i}$：第 t 轮样本 $(\boldsymbol{x}_i, y_i)$ 的权重
  - $h^{(t)}(\boldsymbol{x}_i)$: 第 t 轮算法 A 对样本 $\boldsymbol{x}_i$ 的分类结果. 规定 $y_i,\ h^{(t)}(\boldsymbol{x}_i) \in \{-1, 1\},\ \forall i$
- AdaBoost 对 $d_{t,i}$ 的更新方式为:

$$d_{1,i} = \frac{1}{n},\ \forall i$$

$$d_{t+1,i} = \frac{d_{t,i}}{Z_t} \times \begin{cases} e^{-\alpha_t} & \text{if } y_i = h^{(t)}(\boldsymbol{x}_i) \\ e^{\alpha_t} & \text{if } y_i \neq h^{(t)}(\boldsymbol{x}_i) \end{cases} = \frac{d_{t,i}}{Z_t} e^{-\alpha_t y_i h^{(t)}(\boldsymbol{x}_i)}$$

其中

- Z_t 是归一化常数

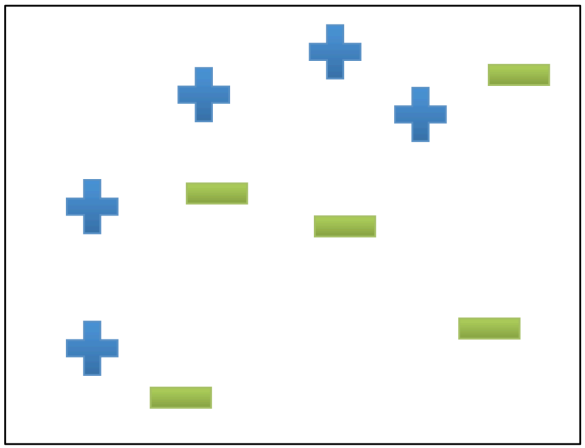
$$\alpha_t = \frac{1}{2} \ln\left(\frac{1-\epsilon_t}{\epsilon_t}\right),\ \epsilon_t = P_{i \sim \boldsymbol{d}_t}\left[h^{(t)}(\boldsymbol{x}_i) \neq y_i\right] = \sum_i d_{t,i} \mathbf{1}_{[h^{(t)}(\boldsymbol{x}_i) \neq y_i]} \tag{1}$$

假设每一轮算法 A 总可以保证 $\epsilon_t < 1/2$, 因此 $\alpha_t > 0$

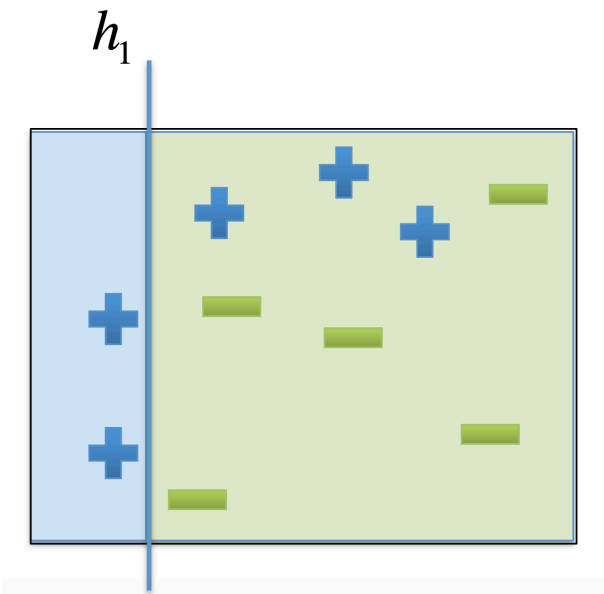
- AdaBoost 最终输出的结果是每一轮分类结果的线性组合:

$$f(\boldsymbol{x}_i) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h^{(t)}(\boldsymbol{x}_i)\right) \tag{2}$$

AdaBoost 工作流程演示



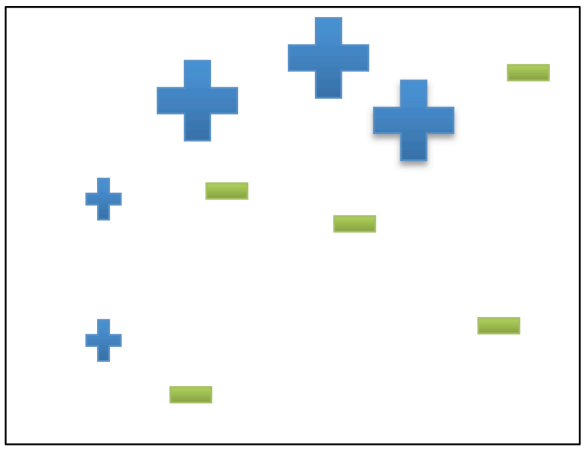
开始的时候所有样本权重相等



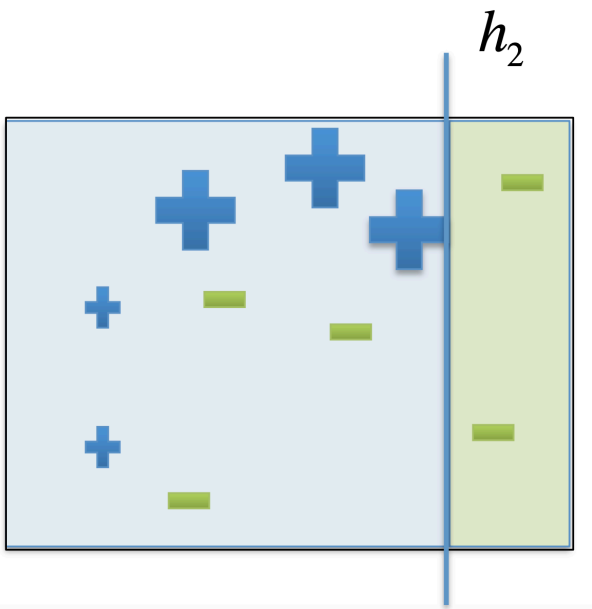
运行算法 A 将每个样本的分类结果记为 $h_1(x_i)$

计算得 $\alpha_1 = 0.42$

AdaBoost 工作流程演示



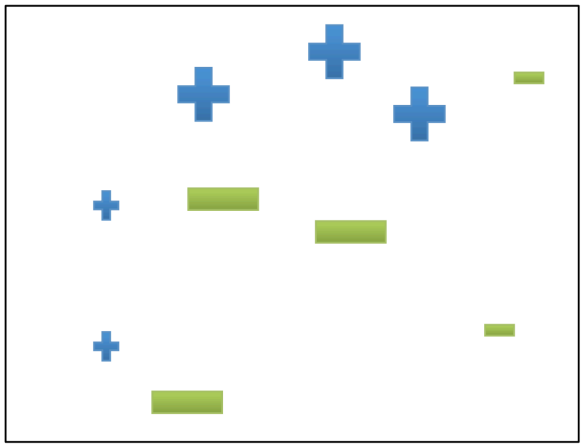
增大错误分类的样本权重，减小正确分类的样本权重



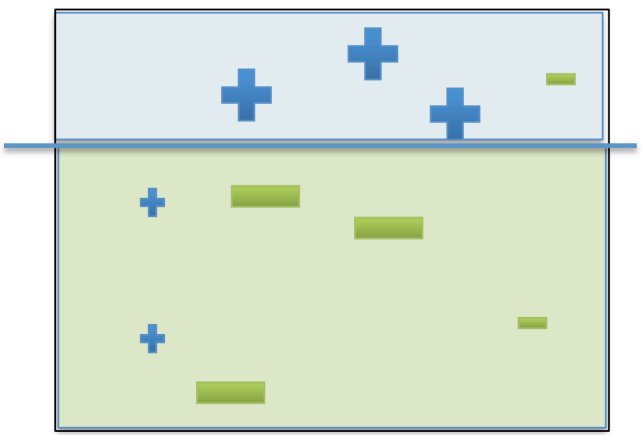
将调整权重后的样本输入算法 A 得到新的分类结果 h_2

此时 $\alpha_2 = 0.66$

AdaBoost 工作流程演示



增大错误分类的样本权重，减小正确分类的样本权重

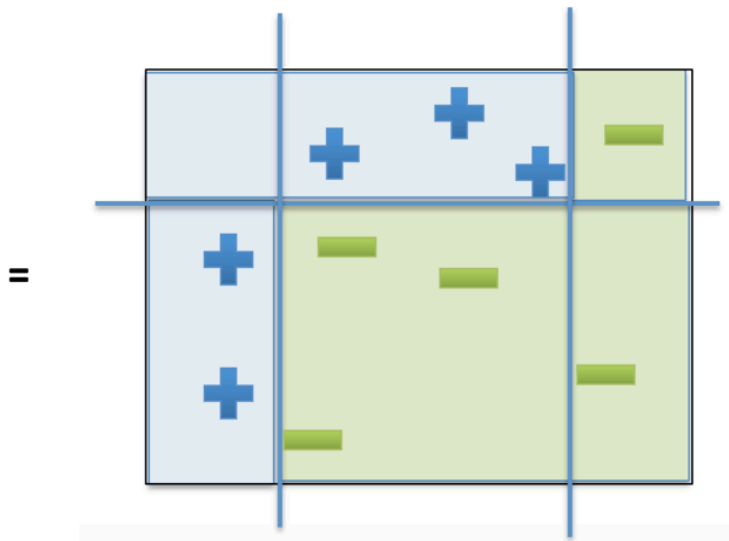


将调整权重后的样本再次输入算法 A 得到新的分类结果 h_3

计算得 $\alpha_3 = 0.93$

AdaBoost 工作流程演示

AdaBoost 最终输出的结果是每一轮分类结果的线性组合：



AdaBoost 统计解释

- AdaBoost 最早由 Freund 和 Schapire 提出，之后有 5 个研究团队几乎同时给出了 AdaBoost 的统计解释 (Breiman, 1997; Friedman et al., 2000; Rätsch et al., 2001; Duffy and Helmbold, 1999; Mason et al., 2000)
- 从统计角度理解 AdaBoost 会发现，它等价于用坐标下降法最小化一个指数损失函数
- 假设有 p 个弱分类器 $\{h_j : h_j(x) \in \{-1, 1\}\}_{j=1}^{p}$，考虑用这些弱分类器的线性组合构造一个新的分类算法

$$f(x) = \sum_{j=1}^{p} \lambda_j h_j(x) \tag{3}$$

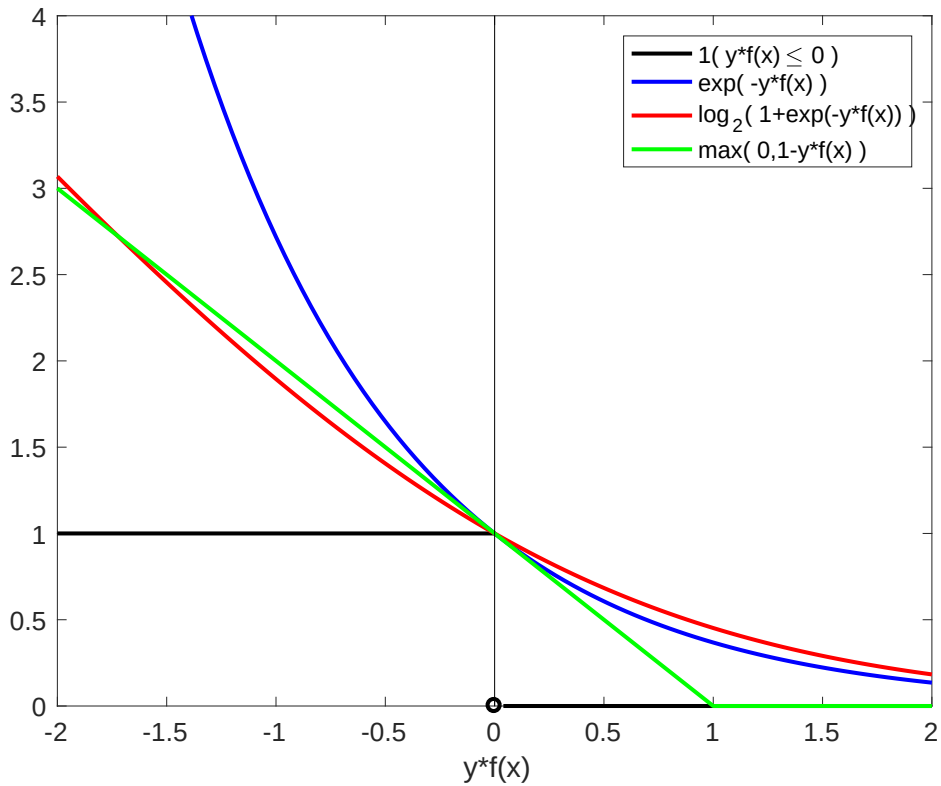
f 在训练集上的错误率定义为：

$$\text{Mis. err} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{[y_i f(x_i) \leq 0]} \tag{4}$$

AdaBoost 统计解释

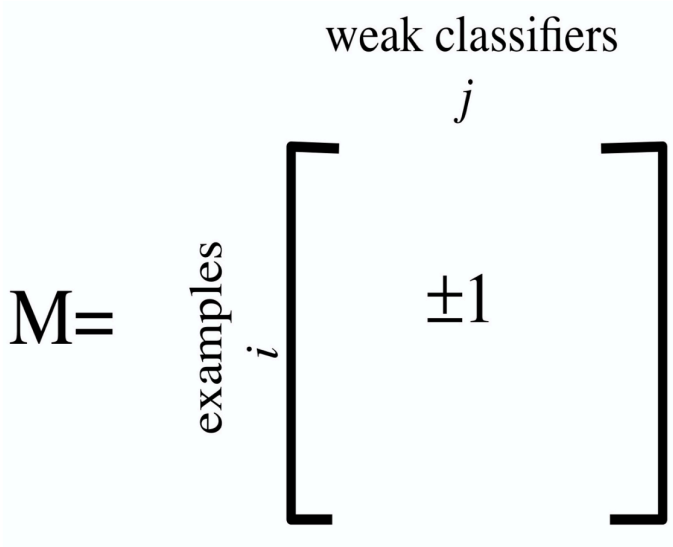
- 最小化(4)寻找最优的 f 比较困难，通常选择最小化(4)的一个凸上界函数，比如指数损失函数:

$$\frac{1}{n}\sum_{i=1}^{n} e^{-y_i f(x_i)} \tag{5}$$



AdaBoost 统计解释

- 如何选择(3)中的 $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_p)^\top$ 使 f 的指数损失函数(5)最小?
  - 定义 $n \times p$ 矩阵 M, 其元素为 $M_{ij} = y_i h_j(x_i)$



  - 则 f 在 $\boldsymbol{\lambda}$ 下的指数损失为

$$L(\boldsymbol{\lambda}) = \frac{1}{n}\sum_{i=1}^{n} e^{-y_i f(x_i)} = \frac{1}{n}\sum_{i=1}^{n} e^{-(M\boldsymbol{\lambda})_i} \tag{6}$$

AdaBoost 统计解释

使用 "坐标下降法" 最小化(6)的基本流程: 在每步迭代 t, 选择使 $L(\boldsymbol{\lambda}_t)$ 下降最快的坐标方向 j_t, 沿该方向移动最优步长 α_t，即只更新 $\boldsymbol{\lambda}_t$ 的第 j_t 分量

- 计算 $L(\boldsymbol{\lambda}_t)$ 关于各分量的偏导数

$$\frac{\partial L(\boldsymbol{\lambda}_t)}{\partial \lambda_j} = -\frac{1}{n}\sum_{i=1}^{n} M_{ij} e^{-(M\boldsymbol{\lambda}_t)_i},\ j = 1, \ldots, p$$

- 选取坐标方向 j_t 使 $L(\boldsymbol{\lambda}_t)$ 在该方向下降最快，即偏导数最小

$$j_t \in \text{argmin}_j \frac{\partial L(\boldsymbol{\lambda}_t)}{\partial \lambda_j} \in \text{argmax}_j \left[\frac{1}{n}\sum_{i=1}^{n} M_{ij} e^{-(M\boldsymbol{\lambda}_t)_i}\right]$$

  - 为了计算方便，将样本 i 经过归一化的指数损失记为:

$$d_{t,i} = e^{-(M\boldsymbol{\lambda}_t)_i}/Z_t,\ \text{其中} Z_t = \sum_{i=1}^{n} e^{-(M\boldsymbol{\lambda}_t)_i} \tag{7}$$

  则有

$$j_t \in \text{argmax}_j \left[\frac{Z_t}{n}\sum_{i=1}^{n} M_{ij} d_{t,i}\right] = \text{argmax}_j\ (\boldsymbol{d}_t^\top M)_j \tag{8}$$

AdaBoost 统计解释

- 选定方向 j_t 后，沿该方向移动的最优步长是多少?
  - 根据(6), $L(\boldsymbol{\lambda}_t + \alpha \boldsymbol{e}_{j_t})$ 是 α 的凸函数，因此只需找到使 $\frac{\partial L(\boldsymbol{\lambda}_t + \alpha \boldsymbol{e}_{j_t})}{\partial \alpha} = 0$ 对应的步长 α_t
  - 定义 $d_+ \triangleq \sum_{i:M_{ij_t}=1} d_{t,i}$, $d_- \triangleq \sum_{i:M_{ij_t}=-1} d_{t,i}$, 解得

$$\alpha_t = \frac{1}{2} \ln \frac{d_+}{d_-} = \frac{1}{2} \ln \frac{1 - d_-}{d_-} \tag{9}$$

Algorithm 1 最小化指数损失函数 (6) 的坐标下降算法

$\boldsymbol{\lambda}_1 = \mathbf{0}$
$d_{1,i} = 1/n$, $i = 1, \ldots, n$
for $t = 1 : T$ **do**
  $j_t \in \operatorname{argmax}_j (\boldsymbol{d}_t^\top M)_j$
  $d_- = \sum_{i:M_{ij_t}=-1} d_{t,i}$
  $\alpha_t = \frac{1}{2} \ln \left(\frac{1 - d_-}{d_-} \right)$
  $\boldsymbol{\lambda}_{t+1} = \boldsymbol{\lambda}_t + \alpha_t \boldsymbol{e}_{j_t}$
  $d_{t+1,i} = e^{-(M\boldsymbol{\lambda}_{t+1})_i} / Z_{t+1}$, 其中 $Z_{t+1} = \sum_{i=1}^n e^{-(M\boldsymbol{\lambda}_{t+1})_i}$

AdaBoost 统计解释

证明：Algorithm 1与 AdaBoost 是等价的

- 注意到 Algorithm 1输出的 $\lambda_{T+1,j}$ 是在 j 方向上移动的总步长，即 $\lambda_{T+1,j} = \sum_{t=1}^{T} \alpha_t \mathbf{1}_{[j_t=j]}$，则有

$$f(x) = \sum_{j=1}^{p} \lambda_{T+1,j} h_j(x) = \sum_{t=1}^{T} \alpha_t h_{j_t}(x) \tag{10}$$

  如果 AdaBoost 中的 $h^{(t)} = h_{j_t}$ 且两者的 $\{\alpha_t\}$ 相同，则 AdaBoost 输出的函数(2)与(10)等价

- 首先检查 AdaBoost 每轮使用的弱分类器 $h^{(t)}$ 与 Algorithm 1每步选择的分类器 h_{j_t} 是否相同?
  - 一个合理的假设是 AdaBoost 每轮在 p 个弱分类器中选择使(1)中定义的出错率 ϵ_t 最小的分类器，即

$$j_t \in \text{argmin}_j \sum_i d_{t,i} \mathbf{1}_{[h_j(x_i) \neq y_i]} = \text{argmax}_j \left(\boldsymbol{d}_t^\top M \right)_j \tag{11}$$

    比较(8)和(11)发现，如果 AdaBoost 和 Algorithm 1每步使用的 $\boldsymbol{d}_t$ 相同，则 AdaBoost 每步选择的分类器与 Algorithm 1相同

AdaBoost 统计解释

- 检查 AdaBoost 每步的权重向量 $\boldsymbol{d}_t$ 与 Algorithm 1是否相同?
  - 当 AdaBoost 每步选择的分类器及使用的 α_t 与 Algorithm 1相同, AdaBoost 的权重向量 $\boldsymbol{d}_{t+1}$ 和 Algorithm 1的 $\boldsymbol{d}_{t+1}$ 是一样的
- 如果 AdaBoost 与 Algorithm 1每步选择的分类器和 $\boldsymbol{d}_t$ 都相同，那么 AdaBoost 每步使用的 α_t 与 Algorithm 1每步移动的步长 α_t 相等
- Adaboost 和 Algorithm 1每步迭代涉及三个要素：权重向量 $\boldsymbol{d}_t$, 分类器 $h^{(t)}$ 和参数 α_t. 以上我们证明了固定其中任意两个要素相等，则第三个要素在两个算法中也相等
  - 注意到两个算法使用的初始值 $\boldsymbol{d}_1$ 相同，由(11)得 $h^{(1)} = h_{j_1}$, 则两个算法得到的 α_1 必然相等，因此权重向量 $\boldsymbol{d}_2$ 也相同，以此类推，两个算法每轮迭代的三要素都相同

AdaBoost 统计解释

定理

如果存在 $\gamma_A > 0$ 使得 *AdaBoost* 每轮出错样本的权重和

$$\epsilon_t = \sum_i d_{t,i} \mathbf{1}_{[h_{j_t}(x_i) \neq y_i]} = \frac{1}{2} - \gamma_t, \text{ 且} \gamma_t > \gamma_A, \forall t. \tag{12}$$

则 *AdaBoost* 在训练集上的错误率(4)以指数速率下降：

$$\frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{[y_i f(x_i) \leq 0]} \leq e^{-2\gamma_A^2 T}. \tag{13}$$

证明：证明的思路是利用与 AdaBoost 等价的 Algorithm 1 找到指数损失函数 $L(\boldsymbol{\lambda}_{t+1})$ 和 $L(\boldsymbol{\lambda}_t)$ 的递归关系，即找出每步迭代减小的训练集误差，然后把这些误差累加起来得出总误差的上界

AdaBoost 概率解释

- 在一些分类问题中，我们不仅希望对 Y 做出准确预测，还希望计算出条件概率 $P(Y=1 \mid x)$

定理 (Friedman et al., 2000)

使指数损失函数的期望

$$E_Y\left[e^{-Yf(x)}\right]$$

最小的 $f(x)$ 为

$$f(x)=\frac{1}{2}\ln\frac{P(Y=1 \mid x)}{P(Y=-1 \mid x)}.$$

- 根据上述定理，可以如下从 AdaBoost 输出的函数 f 中计算 $P(Y=1 \mid x)$

$$P(Y=1 \mid x)=\frac{e^{2f(x)}}{1+e^{2f(x)}}$$