

## Adaboost算法 (分类)

$$\left. \begin{array}{l} h_1(x) \in \{-1, +1\} \\ h_2(x) \in \{-1, +1\} \\ \vdots \\ h_T(x) \in \{-1, +1\} \end{array} \right\} \quad H_T(x) = \text{sign} \left( \sum_{t=1}^T \alpha_t h_t(x) \right)$$

Weak classifiers                      strong classifier

slightly better than random                      知乎 @Eureka

输入：训练数据集

$$T = \{(x_1, y_1), (x_2, y_2), (x_N, y_N)\}$$

其中

$$x_i \in \chi \subseteq R^n, y_i \in \gamma = \{+1, -1\}, i = 1, 2, \dots, N$$

弱学习算法输出为分类器 $G(x)$ 。

### 1. 初始化训练数据的权重 (等权重)

$$D_1 = (w_{11}, w_{12}, \dots, w_{1N}), w_{1i} = \frac{1}{N}, i = 1, 2, \dots, N$$

### 2. 使用具有权重分布 $D_M$ 的训练数据集学习，得到基本分类器

$$G_m(x) : \chi \rightarrow \{-1, +1\}$$

计算 $G_m(x)$ 在训练数据集上的分类误差率

$$e_m = \sum_{i=1}^N P(G_m(x_i)) \neq y_i = \sum_{i=1}^N w_{mi} I(G_m(x_i)) \neq y_i$$

计算 $G_m(x)$ 的系数

$$\alpha_m = \frac{1}{2} \log \frac{1 - e_m}{e_m}$$

更新训练数据集的权值分布

$$D_{m+1} = (w_{m+1,1}, \dots, w_{m+1,i}, \dots, w_{m+1,N})$$

$$w_{m+1,i} = \frac{w_{mi}}{Z_m} \exp(-\alpha_m y_i G_m(x_i)),$$

$$= \begin{cases} \frac{w_{mi}}{Z_m} \exp(-\alpha_m), & G_m(x_i) = y_i \\ \frac{w_{mi}}{Z_m} \exp(\alpha_m), & G_m(x_i) \neq y_i \end{cases}$$

其中， $Z_m$ 是规范化因子

$$Z_m = \sum_{i=1}^N w_{mi} \exp(-\alpha_m y_i G_m(x_i))$$

构建基本分类器的线性组合

$$f(x) = \sum_{m=1}^N \alpha_m G_m(x)$$

得到了最终的分类器！

$$G(x) = \text{sign}(f(x)) = \text{sign}\left(\sum_{m=1}^M \alpha_m G_m(x)\right)$$

**注1：以上为二分类的建模，对于多元变量差别只是在弱分类器的系数上：**

$$\alpha_m = \frac{1}{2} \log \frac{1 - e_m}{e_m} + \log(R - 1)$$

R为分类别数。如果是二元分类，则和二元分类算法中的系数一致。

**注2：** $\alpha$  随着 $e_m$ 的减少而增大，所以误分类误差率越小的基本分类器在最终分类器的作用越大，符合预期。

**注3：**这里的 $\alpha_m$ 表示了基本分类器 $G_M$ 的重要性，所以和不为1。

## AdaBoost算法（回归）

### 1. 初始化训练数据的权重（等权重）

$$D_1 = (w_{11}, w_{12}, \dots, w_{1N}), w_{1i} = \frac{1}{N}, i = 1, 2, \dots, N$$

## 2.使用具有权重分布 $D_M$ 的训练数据集学习，得到基本分类器 $G_m(x)$

计算训练集上的最大误差

$$E_M = \max |y_i - G_m(x_i)|$$

计算每个样本的相对误差：

$$\left\{ \begin{array}{l} \text{如果是线性的误差, 则 } e_{mi} = \frac{|y_i - G_m(x_i)|}{E_m} \\ \text{如果是平方误差, 则 } e_{mi} = \frac{(y_i - G_m(x_i))^2}{E_m^2} \\ \text{如果是指数误差, 则 } e_{mi} = 1 - \exp\left(\frac{-y_i + G_m(x_i)}{E_m}\right) \end{array} \right.$$

计算回归误差率：

$$e_m = \sum_{i=1}^N w_{mi} e_{mi}$$

并计算弱学习器的系数：

$$\alpha_m = \frac{e_m}{1 - e_m}$$

最终更新样本集的权重分布为：

$$w_{m+1,i} = \frac{w_{mi}}{Z_m} \alpha_m^{1-e_{mi}}$$

其中

$$Z_m = \sum_{i=1}^N w_{mi} \alpha_m^{1-e_{mi}}$$

## 3.构建基本分类器的线性组合

$$f(x) = \sum_{m=1}^M \left( \ln \frac{1}{\alpha_m} \right) g(x)$$

其中,  $g(x)$ 是所有 $\alpha_m G_m(x)$ ,  $m = 1, 2 \dots M$ 的中位数。

从另一个角度，AdaBoost还有一种解释，即可认为AdaBoost算法是模型为加法模型、损失函数为指数函数、学习算法为前向分布算法时的二类分类学习方法。

###

## Adaboost的优缺点

### 优点:

Adaboost提供一种框架，在框架内可以使用各种方法构建子分类器。可以使用简单的弱分类器，不用对特征进行筛选，也不存在过拟合的现象。

Adaboost可以根据弱分类器的反馈，自适应地调整假定的错误率，执行的效率高。

### 缺点:

在Adaboost训练过程中，Adaboost会使得难于分类样本的权值呈指数增长，训练将会过于偏向这类困难的样本，导致Adaboost算法易受噪声干扰。此外，Adaboost依赖于弱分类器，而弱分类器的训练时间往往很长。