Adaboost

输入: 训练集 $D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)\}$;
基学习算法 $\mathfrak{L}$;
训练轮数 T.

过程:

1: $\mathcal{D}_1(\boldsymbol{x}) = 1/m$.
2: **for** $t = 1, 2, \ldots, T$ **do**
3: $\quad h_t = \mathfrak{L}(D, \mathcal{D}_t)$;
4: $\quad \epsilon_t = P_{\boldsymbol{x} \sim \mathcal{D}_t}(h_t(\boldsymbol{x}) \neq f(\boldsymbol{x}))$;
5: $\quad$ **if** $\epsilon_t > 0.5$ **then break**
6: $\quad \alpha_t = \frac{1}{2} \ln \left(\frac{1-\epsilon_t}{\epsilon_t} \right)$;
7: $$\mathcal{D}_{t+1}(\boldsymbol{x}) = \frac{\mathcal{D}_t(\boldsymbol{x})}{Z_t} \times \begin{cases} \exp(-\alpha_t), & \text{if } h_t(\boldsymbol{x}) = f(\boldsymbol{x}) \\ \exp(\alpha_t), & \text{if } h_t(\boldsymbol{x}) \neq f(\boldsymbol{x}) \end{cases}$$
$$= \frac{\mathcal{D}_t(\boldsymbol{x})\exp(-\alpha_t f(\boldsymbol{x}) h_t(\boldsymbol{x}))}{Z_t}$$
8: **end for**

输出: $H(\boldsymbol{x}) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(\boldsymbol{x}) \right)$

Adaboost

例 给定如下表所示训练数据。假设个体学习器由x（输入）和y（输出）产生，其阈值v（判定正反例的分界线）使该分类器在训练数据集上分类误差率最低。（y=1为正例，y=-1为反例）

序号	1	2	3	4	5	6	7	8	9	10
x	0	1	2	3	4	5	6	7	8	9
y	1	1	1	-1	-1	-1	1	1	1	-1

第一个个体学习器：

我们首先认为 x_i (i=1,2,...,10)的权重是一样的，即每一个数据同等重要。（权重是用来计算误差的）

D_1

x	0	1	2	3	4	5	6	7	8	9
w	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
y	1	1	1	-1	-1	-1	1	1	1	-1

（a）在权值分布为 D_1 的训练数据上，阈值v取2.5（红线）时分类误差率最低（此时x=6,7,8的数据被错分为反例，误差为它们的权重之和 e_1 =0.1+0.1+0.1=0.3，误差率小于 $\frac{1}{2}$ 才有意义），故

Adaboost

个体学习器为 $G_1(x)=\begin{cases}1 & x<2.5,\\ -1 & x>2.5.\end{cases}$

(b）根据误差 e_1 计算系数 α_1 =0.4236（公式：$\alpha_i=\frac{1}{2}log\frac{1-e_i}{e_i}$，可以发现只有当 $e_i<\frac{1}{2}$ 时，α_i >0，这样个体学习器才是有意义的）

(c）更新训练数据的权值分布（公式：

$$w_{m+1,i}=\frac{w_{m,i}}{Z_m}exp\left(-\alpha_m y_i G_m\left(x_i\right)\right),i=1,2,\cdots,N,$$

$$Z_m=\sum_{i=1}^{N}w_{m,i}exp\left(-\alpha_m y_i G_m\left(x_i\right)\right),$$ Z_m 是为了保证每次权值总和为1）

(通过指数损失函数 $exp\left(x\right)$ 调整权重，分类正确的降低权重(y_i 和 $G_m\left(x_i\right)$ 同号则 $y_i G_m\left(x_i\right)>0$，$-\alpha_m y_i G_m\left(x_i\right)<0$)，分类错误的增加权重）：

Adaboost


```
>>> import math
>>> a = 0.5*math.log((1-0.3)/0.3,math.e)
>>> a
0.4236489301936018 4
```

```
>>> zm=0.1*math.e**(-0.4236)*7+0.1*math.e**(0.4236)*3
>>> zm
0.9165151400883117
>>> wi1 =  0.1/zm*math.e**(-0.4236)
>>> wi2 =  0.1/zm*math.e**(0.4236)
>>> wi1,wi2
(0.0714320664423973 4, 0.1666585116344062)
```

Adaboost

D_2 （权重之和始终为1）

x	0	1	2	3	4	5	6	7	8	9
w	0.07143	0.07143	0.07143	0.07143	0.07143	0.07143	0.16667	0.16667	0.16667	0.07143
y	1	1	1	-1	-1	-1	1	1	1	-1

可以看到x=6,7,8的数据的权重变大了，而其他数据的权重降低了，这是希望能把之前经常分类错误（经常分类错误会出现权重不断变大）的数据能在下一个个体学习器分类正确（记住：权重是用来计算误差的，为了降低误差，选择阈值时会倾向把权重大的分类正确）

$$f_1(x) = \alpha_1 G_1(x) = 0.4236 G_1(x)$$

$$0.4236 G_1(x) = \begin{cases} 0.4236 * 1 & x < 2.5, \\ 0.4236 * (-1) & x > 2.5. \end{cases}$$

$$sign[f_1(x)] = \begin{cases} 1 & x < 2.5, \\ -1 & x > 2.5. \end{cases}$$

集成学习器 $sign[f_1(x)]$ （第一次集成，只有一个个体学习器）在训练数据集上有3个误分类点

Adaboost

第二个个体学习器：

D_2

```
>>> a = 0.5*math.log((1-0.2143)/0.2143,math.e)
>>> a
0.6495990688511224
```

x	0	1	2	3	4	5	6	7	8	9
w	0.07143	0.07143	0.07143	0.07143	0.07143	0.07143	0.16667	0.16667	0.16667	0.07143
y	1	1	1	-1	-1	-1	1	1	1	-1

（a）在权值分布为 D_2 的训练数据上，阈值v取8.5时分类误差率最低（此时x=3,4,5的数据被错分为正例，误差为它们的权重之和 e_2 =0.07143+0.07143+0.07143=0.2143，误差率降低了！），

故个体学习器为 $G_2(x) = \begin{cases} 1 & x < 8.5, \\ -1 & x > 8.5. \end{cases}$

（b）根据误差 e_2 计算系数 $\alpha_2 = 0.6496$

（c）更新训练数据的权值分布（在 D_2 的基础上调整 D_3 ，分类正确的降低权重，分类错误的增加权重）：

Adaboost

D_3

x	0	1	2	3	4	5	6	7	8	9
w	0.0455	0.0455	0.0455	0.16667	0.16667	0.16667	0.1060	0.1060	0.1060	0.0455
y	1	1	1	-1	-1	-1	1	1	1	-1

对比 D_2 可以看到x=3,4,5的数据的权重变大了，而其他权重降低了。

$$0.4236G_1(x) = \begin{cases} 0.4236 * 1 & x < 2.5, \\ 0.4236 * (-1) & x > 2.5. \end{cases}$$

$$0.6496G_2(x) = \begin{cases} 0.6496 * 1 & x < 8.5, \\ 0.6496 * (-1) & x > 8.5. \end{cases}$$

$$f_2(x) = \alpha_1 G_1(x) + \alpha_2 G_2(x) = 0.4236G_1(x) + 0.6496G_2(x)$$

$$f_2(x) = \begin{cases} 0.4236 * 1 + 0.6496 * 1 = 1.0732 & x < 2.5, \\ 0.4236 * (-1) + 0.6496 * 1 = 0.226 & 2.5 < x < 8.5, \\ 0.6496 * (-1) = -0.6496 & x > 8.5, \end{cases}$$ （注意：x<2.5时，也<8.5）

$$sign[f_2(x)] = \begin{cases} 1 & x < 8.5, \\ -1 & x > 8.5. \end{cases}$$

Adaboost

D_3

x	0	1	2	3	4	5	6	7	8	9
w	0.0455	0.0455	0.0455	0.16667	0.16667	0.16667	0.1060	0.1060	0.1060	0.0455
y	1	1	1	-1	-1	-1	1	1	1	-1

（a）在权值分布为 D_3 的训练数据上，阈值v取5.5时分类误差率最低（e_3 =0.1820，误差率又降低了！x=0,1,2,9被分类错误），故个体学习器为 $G_3(x) = \begin{cases} -1 & x < 5.5, \\ 1 & x > 5.5. \end{cases}$

（b）根据误差 e_3 计算系数 $\alpha_3 = 0.7514$

（c）更新训练数据的权值分布：

Adaboost

D_4

x	0	1	2	3	4	5	6	7	8	9
w	0.125	0.125	0.125	0.102	0.102	0.102	0.065	0.065	0.065	0.125
y	1	1	1	-1	-1	-1	1	1	1	-1

$$f_3(x) = \alpha_1 G_1(x) + \alpha_2 G_2(x) + \alpha_3 G_3(x)$$

$$f_3(x) = \begin{cases} 0.3218 & x < 2.5, \\ -0.5254 & 2.5 < x < 5.5, \\ 0.9774 & 5.5 < x < 8.5, \\ -0.3218 & x > 8.5. \end{cases}$$

$$sign[f_3(x)] = \begin{cases} 1 & x < 2.5, \\ -1 & 2.5 < x < 5.5, \\ 1 & 5.5 < x < 8.5, \\ -1 & x > 8.5. \end{cases}$$

最终结果：

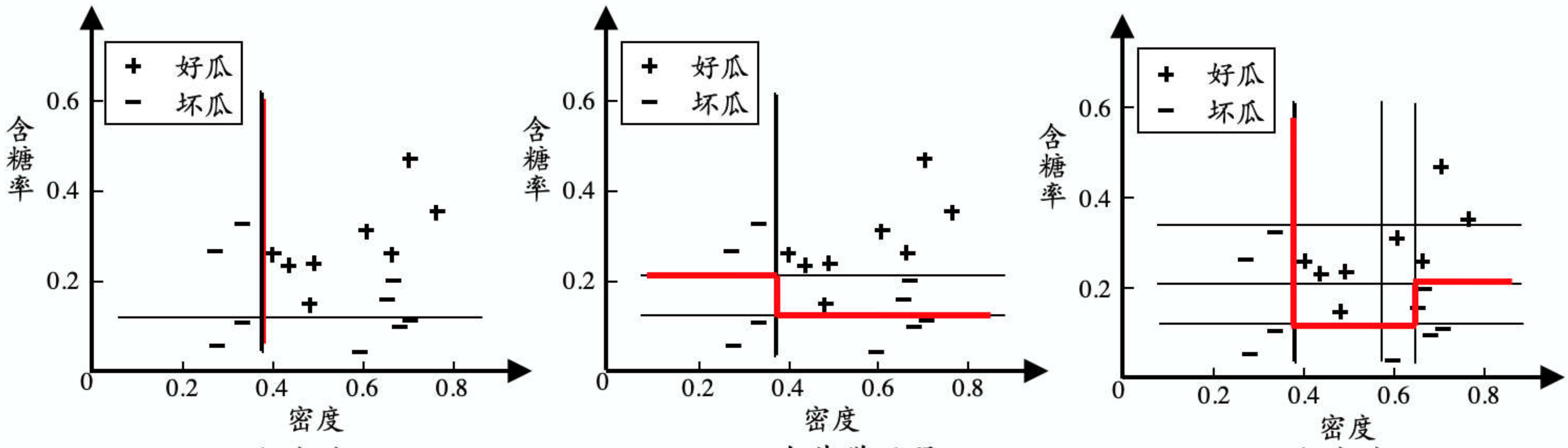
x	0	1	2	3	4	5	6	7	8	9
w	0.125	0.125	0.125	0.102	0.102	0.102	0.065	0.065	0.065	0.125
y	1	1	1	-1	-1	-1	1	1	1	-1

分类器 $sign[f_3(x)]$ 在训练数据集上有0个误分类点（amazing！）

Adaboost

- 对比三个个体学习器可以发现，权值很低的数据从侧面说明，它们在前面的学习器经常被分类正确，也就是说它们被分类正确的票数就比较多（a相当于每个分类器的票数），那么之后的个体学习器把它们分类错也影响不大，反正总票数是分类正确的票数多就可以了。例如x=1，前面两次分类对了，获得正确票数0.4236+ 0.6496 =1.0732，第三次错了，获得错误票数0.7514，正确票数多，最终还是分类正确了。
- 为了想办法让分类错误的数据变为分类正确的，后面的个体学习器也在努力。如x=6,第一次分类错误的票数为0.4236，第二次分类正确的票数0.6496，可以看到为了让前面分类错误的数据变为分类正确的，后面个体学习器的重要性（a）需要比前面的大。

Adaboost



(a) 3个基学习器 (b) 5个基学习器 (c) 11个基学习器