提纲

一、机器学习基本概念

二、回归分析

三、决策树

四、线性判别分析

五、Ada Boosting

六、支持向量机

七、生成学习模型

集成算法

集成算法通常有两种方式，分别是套袋法和提升法。

1.Bagging（套袋法）

- 从原始样本集中使用Bootstraping方法(自助法，是一种有放回的抽样方法)随机抽取n个训练样本，共进行k轮抽取，得到k个训练集。（k个训练集之间相互独立，元素可以有重复）
- 对于k个训练集，我们训练k个模型（这k个模型可以根据具体问题而定，比如决策树，knn等）
- 对于分类问题：由投票表决产生分类结果；对于回归问题：由k个模型预测结果的均值作为最后预测结果。（所有模型的重要性相同）

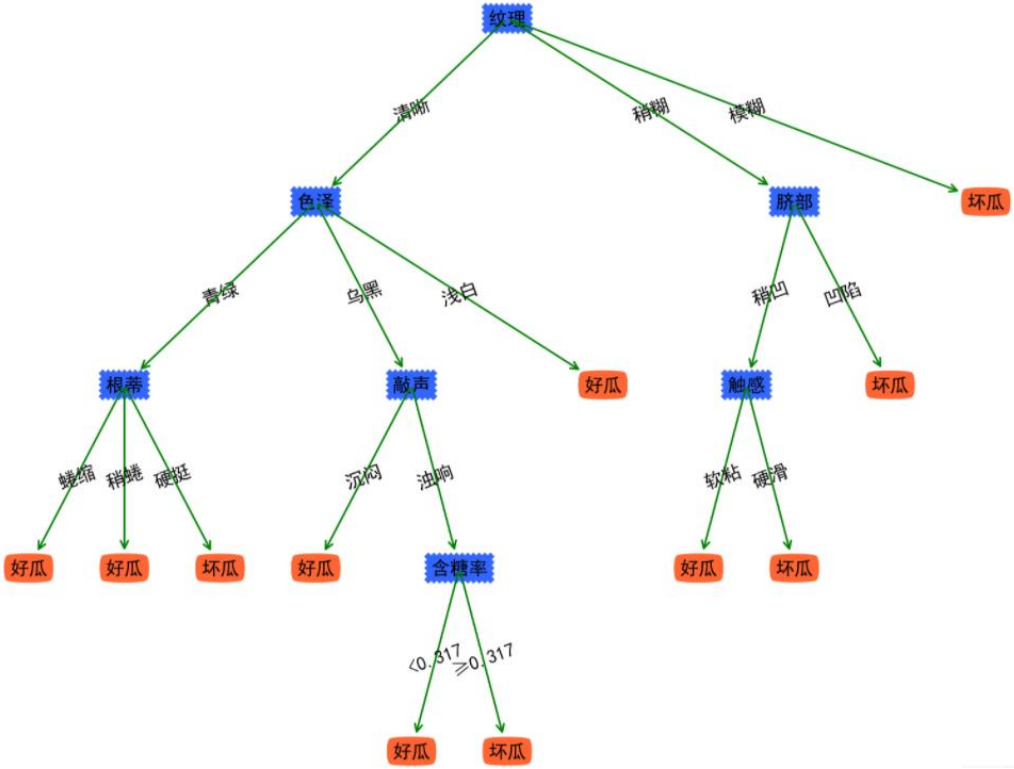
2.Boosting（提升法）

- 对于训练集中的每个样本建立权值wi，表示对每个样本的关注度。当某个样本被误分类的概率很高时，需要加大对该样本的权值。
- 进行迭代的过程中，每一步迭代都是一个弱分类器。我们需要用某种策略将其组合，作为最终模型。（例如AdaBoost给每个弱分类器一个权值，将其线性组合最为最终分类器。误差越小的弱分类器，权值越大）

算法组合案例

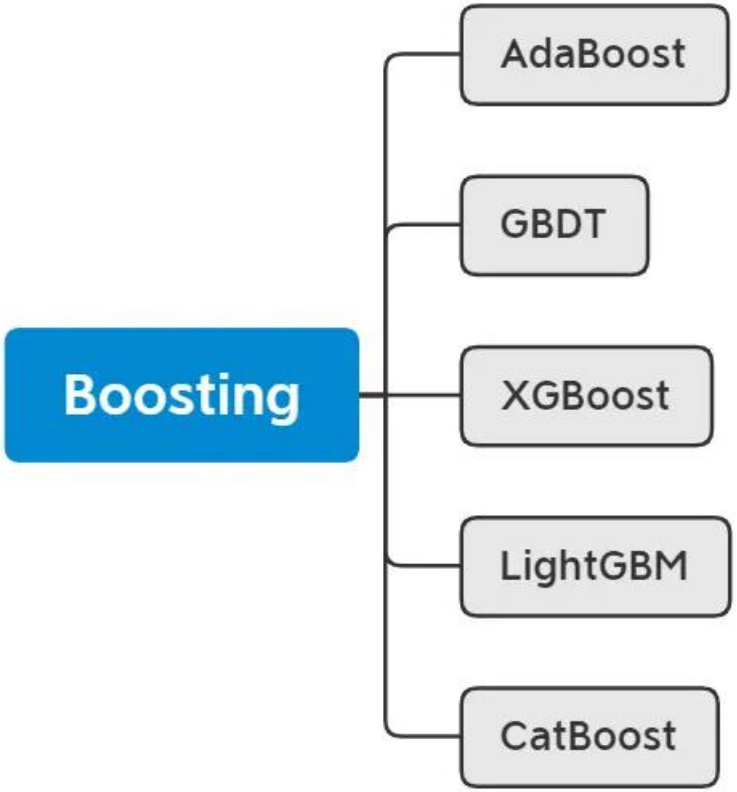
将决策树与这些算法框架进行结合所得到的新的算法：

- Bagging + 决策树 = **随机森林**
- AdaBoost + 决策树 = **自适应提升树**
- Gradient Boosting + 决策树 = **GBDT（梯度提升树）**



Boosting算法种类

我们可以将Boosting理解为一类将弱分类器提升为强分类器的算法，所以有时候Boosting算法也叫提升算法。下列的这些算法就是常见的Boosting算法：**AdaBoost、GBDT、XGBoost、LightGBM和CatBoost**。



Adaptive Boosting
自适应提升算法

gradient boosting decision tree
梯度提升决策树

eXtreme Gradient Boosting
极度梯度提升树

light gradient boosting machine
轻量的梯度提升机

Categorical+Boosting

Boosting (Adaptive Boosting, 自适应提升)

From Adaptive Computation and Machine Learning

Boosting

Foundations and Algorithms

By Robert E. Schapire and Yoav Freund

Overview

Boosting is an approach to machine learning based on the idea of creating a highly accurate predictor by combining many weak and inaccurate "rules of thumb." A remarkably rich theory has evolved around boosting, with connections to a range of topics, including statistics, game theory, convex optimization, and information geometry. Boosting algorithms have also enjoyed practical success in such fields as biology, vision, and speech processing. At various times in its history, boosting has been perceived as mysterious, controversial, even paradoxical.

This book, written by the inventors of the method, brings together, organizes, simplifies, and substantially extends two decades of research on boosting, presenting both theory and applications in a way that is accessible to readers from diverse backgrounds while also providing an authoritative reference for advanced researchers. With its introductory treatment of all material and its inclusion of exercises in every chapter, the book is appropriate for course use as well.

The book begins with a general introduction to machine learning algorithms and their analysis; then explores the core theory of boosting, especially its ability to generalize; examines some of the myriad other theoretical viewpoints that help to explain and understand boosting; provides practical extensions of boosting for more complex learning problems; and finally presents a number of advanced theoretical topics. Numerous applications and practical illustrations are offered throughout.

- 对于一个复杂的分类任务，可以将其分解为若干子任务，然后将若干子任务完成方法综合，最终完成该复杂任务。
- 将若干个弱分类器(weak classifiers)组合起来，形成一个强分类器(strong classifier)。
- 能用众力，则无敌于天下矣；**能用众智，则无畏于圣人矣**(语出《三国志·吴志·孙权传》)
- 三个臭皮匠顶个诸葛亮。

Freund, Yoav; Schapire, Robert E (1997), A decision-theoretic generalization of on-line learning and an application to boosting, Journal of Computer and System Sciences (original paper of Yoav Freund and Robert E.Schapire where AdaBoost is first introduced.)

计算学习理论：霍夫丁不等式(Hoeffding's inequality)

- 学习任务：统计某个电视节目在全国的收视率。
- 方法：不可能去统计整个国家中每个人是否观看电视节目、进而算出收视率。只能抽样一部分人口，然后将抽样人口中观看该电视节目的比例作为该电视节目的全国收视率。
- 霍夫丁不等式：全国人口中看该电视节目的人口比例（记作x）与抽样人口中观看该电视节目的人口比例（记作y）满足如下关系：

$$P(|x-y| \geq \epsilon) \leq 2e^{-2N\epsilon^2}$$ (N是采样人口总数、$\epsilon \in (0,1)$是所设定的可容忍误差范围)

当N足够大时，“全国人口中电视节目收视率”与“样本人口中电视节目收视率”差值超过误差范围ϵ的概率非常小。

概率近似正确

计算学习理论：概率近似正确 (probably approximately correct, PAC)

- 对于统计电视节目收视率这样的任务，可以通过不同的采样方法（即不同模型，比如根据人的不同年龄分布、文化程度分布、区域分布）来计算收视率。
- 每个模型会产生不同的误差。
- 问题：如果得到完成该任务的若干“弱模型”，是否可以将这些弱模型组合起来，形成一个“强模型”。该“强模型”产生误差很小呢？这就是概率近似正确（PAC）要回答的问题。(probably approxiamtely correct. PAC)

计算学习理论：概率近似正确 (probably approximately correct, PAC)

在概率近似正确背景下，有“强可学习模型”和“弱可学习模型”

强可学习 (strongly learnable)	● 学习模型能够以较高精度对绝大多数样本完成识别分类任务。
弱可学习 (weakly learnable)	学习模型仅能完成若干部分样本识别与分类，其精度略高于随机猜测。(不可学习指的是学习模型所获得精度仅为50%，即相当于随机猜测)
强可学习和弱可学习是等价的，也就是说，如果已经发现了“弱学习算法”，可将其提升（Boosting）为“强学习算法”。Ada Boosting算法就是这样的方法。具体而言，Ada Boosting将一系列弱分类器组合起来，构成一个强分类器。	

Ada Boosting：思路描述

- **Ada Boosting算法中两个核心问题：**
  - 在每个弱分类器学习过程中，如何改变训练数据的权重：提高在上一轮中分类错误样本的权重。（做错的题要重点复习-单个分类器）
  - 如何将一系列弱分类器组合成强分类器：通过加权多数表决方法来提高分类误差小的弱分类器的权重，让其在最终分类中起到更大作用。同时减少分类误差大的弱分类器的权重，让其在最终分类中仅起到较小作用。（能者多劳-多个分类器）

Ada Boosting：思路描述

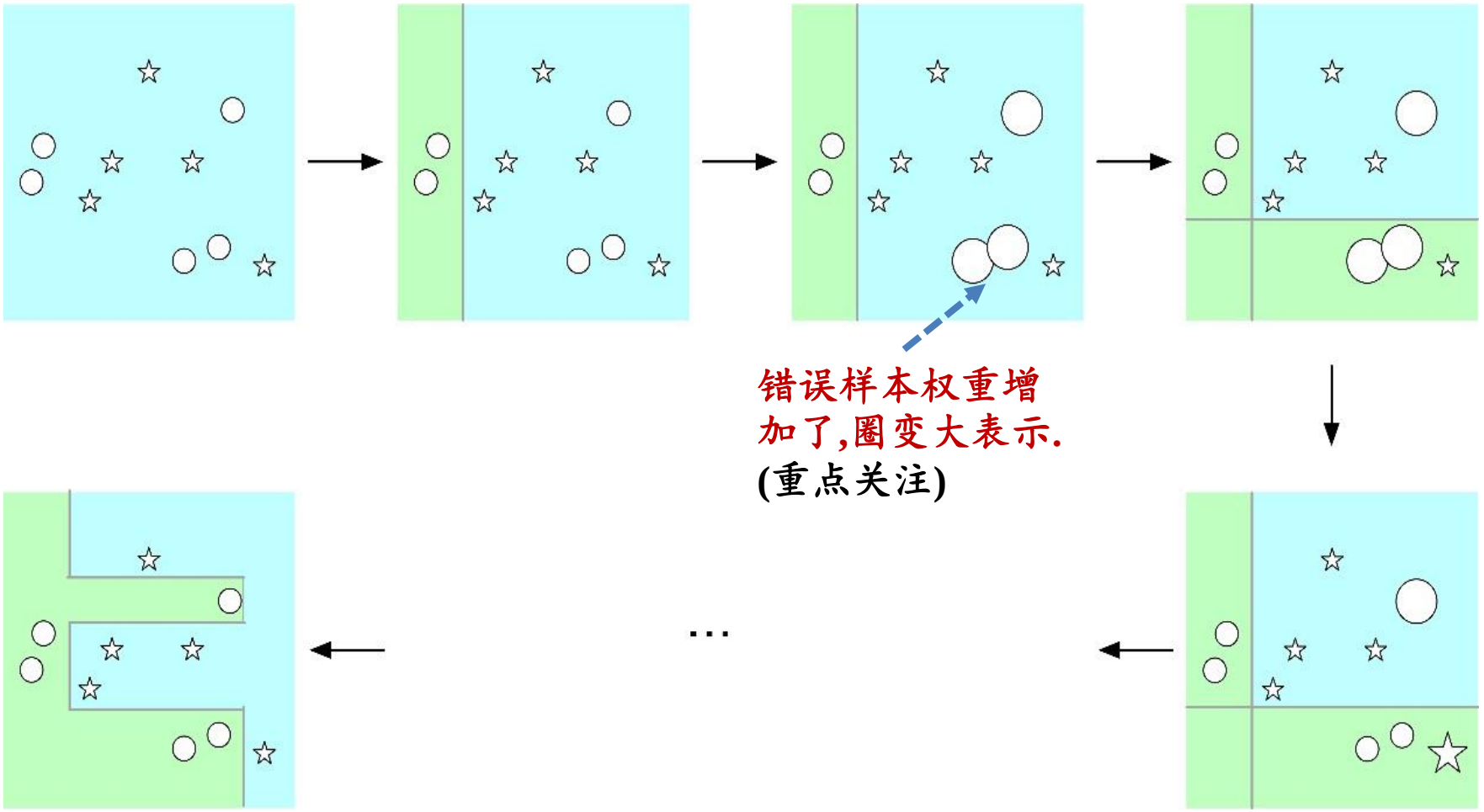


图4.9 Ada Boosting算法学习过程示意图　　错误样本权重增加了

Ada Boosting：算法描述---数据样本权重初始化

- 给定包含N个标注数据（样本）的训练集合Γ，$\Gamma = \{(x_1, y_1), \dots, (x_N, y_N)\}$。$x_i (1 \le i \le N) \in X \subseteq R^n, y_i \in Y = \{-1, 1\}$
- Ada Boosting算法将从这些标注数据出发，训练得到一系列弱分类器(如使用逻辑斯蒂回归,决策树等做而分类)，并将这些弱分类器线性组合得到一个强分类器。

1. 初始化每个训练样本的权重(每个样本重要性一样)

$D_1 = (w_{11}, \dots, w_{1i}, \dots, w_{1N})$，其中 $w_{1i} = \frac{1}{N} (1 \le i \le N)$（每个初始权重一样大，且都位于[0 1]区间，类似于概率分布。）

Ada Boosting：算法描述---第m个弱分类器训练

2. 对$m = 1, 2, \ldots, M$（在第m次迭代中，Ada Boosting总是趋向于将具有最小误差（err_m）的学习模型选做本轮生成的弱分类器G_m，使得累积误差快速下降。）

a) 使用具有分布权重D_m的训练数据来学习得到第m个基分类器（弱分类器）G_m：

$$G_m(x): X \to Y \quad \text{其中} X \subseteq R^n, Y = \{-1, 1\}$$

b) 计算$G_m(x)$在训练数据集上的分类误差

$err_m = \sum_{i=1}^{N} w_{mi} I(G_m(x_i) \neq y_i)$ 这里：$I(\cdot) = 1$，如果$G_m(x_i) \neq y_i$；否则为0。

c) 计算弱分类器$G_m(x)$的权重：$\alpha_m = \frac{1}{2} \ln \frac{1 - err_m}{err_m} > 0$，误差越低，分类器权重越大。

Ada Boosting：算法描述---第m个弱分类器训练

d) 更新训练样本数据的分布权重：

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

其中Z_m是归一化因子以使得D_{m+1}为概率分布(值在0和1之间)，

$$Z_m = \sum_{i=1}^{N} w_{m,i}\, e^{-\alpha_m y_i G_m(x_i)}$$

◆如果预测正确，则$y_i G_m(x_i)$=1，$w_{m+1,i} = \frac{w_{m,i} e^{-\alpha_m}}{\sum_{i=1}^{N} w_{m,i} e^{-\alpha_m}}$,

$e^{-\alpha_m}$<1(因为α_m>0),即$w_{m+1,i} < w_{m,i}$,权重降低.

◆如果预测错误，$G_m(x_i)$=-1,权重增加。

错误的样本权重更大----错误样本被重点关注。

Ada Boosting：算法描述---弱分类器组合成强分类器

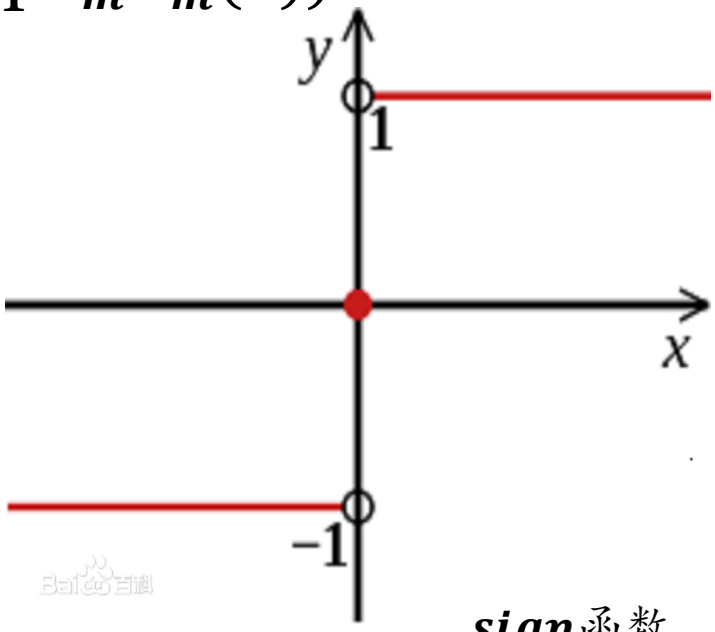
3. 以线性加权形式来组合弱分类器$f(x)$,得到强分类器$G(x)$

$$f(x)=\sum_{i=1}^{M}\alpha_m G_m(x)$$

能者多劳

α_m为$G_m(x)$的权重,错误率越低,权重越高,反之亦然.

$$G(x)=sign(f(x))=sign(\sum_{i=1}^{M}\alpha_m G_m(x))$$



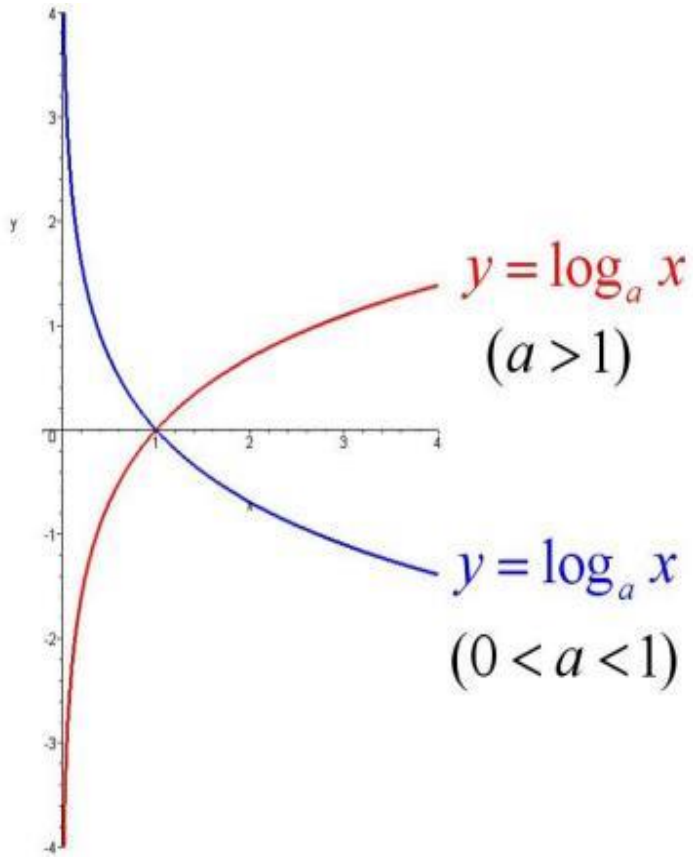
sign函数

Ada Boosting：算法解释

第m个弱分类器$G_m(x)$在训练数据集上产生的分类误差：

该误差为被错误分类的样本所具有权重的累加：

$err_m = \sum_{i=1}^{N} w_{m,i} I(G_m(x_i) \neq y_i)$ **这里：**$I(\cdot) = 1$，**如果**$G_m(x_i) \neq y_i$；**否则为0。**



Ada Boosting：算法解释

计算第m个弱分类器$G_m(x)$的权重α_m： $\alpha_m = \frac{1}{2}\ln\frac{1-err_m}{err_m}$

(a) 当第m个弱分类器$G_m(x)$错误率err_m为1，即$err_m = \sum_{i=1}^{N} w_{mi} I(G_m(x_i) \neq y_i) = 1$，意味每个样本分类出错，则$\alpha_m = \frac{1}{2}\ln\frac{1-err_m}{err_m} \to -\infty$，给予第$m$个弱分类器$G_m(x)$很低权重。即分类器的误差越大，权重越小。

(b) 当第m个弱分类器$G_m(x)$错误率err_m为$\frac{1}{2}$，$\alpha_m = \frac{1}{2}\ln\frac{1-err_m}{err_m} = 0$。如果错误率$err_m$小于$\frac{1}{2}$，权重$\alpha_m$为正($err_m < \frac{1}{2}$、$\alpha_m > 0$)。可知权重$\alpha_m$随$err_m$减少而增大，即误差越小的弱分类器会赋予更大权重。

(c) 如果一个弱分类器的分类错误率为$\frac{1}{2}$，可视为其性能仅相当于随机分类效果。

Ada Boosting：算法解释

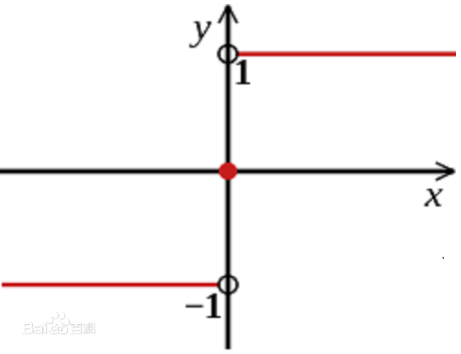
在开始训练第$m+1$个弱分类器$G_{m+1}(x)$之前对训练数据集中数据权重进行调整

$$w_{m+1,i}=\begin{cases}\dfrac{w_{m,i}}{Z_m}e^{-\alpha_m}, & G_m(x_i)=y_i\\ \dfrac{w_{m,i}}{Z_m}e^{\alpha_m}, & G_m(x_i)\neq y_i\end{cases}$$

- 可见，如果某个样本无法被第m个弱分类器$G_m(x)$分类成功，则需要增大该样本权重，否则减少该样本权重。这样，被错误分类样本会在训练第$m+1$个弱分类器$G_{m+1}(x)$时会被“重点关注”。即错误的样本权重更大。
- 在每一轮学习过程中，Ada Boosting算法均在划重点（重视当前尚未被正确分类的样本）。

Ada Boosting：算法解释

弱分类器构造强分类器



$$f(x) = \sum_{i=1}^{M} \alpha_m G_m(x)$$

$$G(x) = sign(f(x)) = sign(\sum_{i=1}^{M} \alpha_m G_m(x))$$

- $f(x)$是M个弱分类器的加权线性累加。分类能力越强的弱分类器具有更大权重。
- α_m累加之和并不等于1。
- $f(x)$符号决定样本x分类为1或-1。如果$\sum_{i=1}^{M} \alpha_m G_m(x)$为正，则强分类器$G(x)$将样本$x$分类为1；否则为-1。

Ada Boosting：回看霍夫丁不等式

假设有M个弱分类器$G_m(1 \le m \le M)$，则M个弱分类器线性组合所产生误差满足条件：

$$P\{\sum_{i=1}^{M} G_m(x) \neq \zeta(x)\} \le e^{-\frac{1}{2}M(1-2\varepsilon)^2}$$

- $\zeta(x)$ 是真实分类函数、$\varepsilon \in (0,1)$ 。上式表明：如果所“组合” 弱分类器越多，则学习分类误差呈指数级下降，直至为零。
- 上述不等式成立有两个前提条件：1) 每个弱分类器产生的误差相互独立； 2) 每个弱分类器的误差率小于50%。因为每个弱分类器均是在同一个训练集上产生，条件1）难以满足。也就说，“准确性（对分类结果而言）”和“差异性（对每个弱分类器而言）”难以同时满足。
- Ada Boosting 采取了序列化学习机制。

Ada Boosting：优化目标

Ada Boost实际在最小化如下指数损失函数(Minimization of Exponential Loss)：

$$\sum_i e^{-y_i f(x_i)} = \sum_i e^{-y_i \sum_{m=1}^{M} \alpha_m G_m(x_i)}$$

Ada Boost的分类误差上界如下所示：

$$\frac{1}{N}\sum_{i=1}^{N} I(G(x_i) \neq y_i) \leq \frac{1}{N}\sum_i e^{-y_i f(x_i)} = \prod_m Z_m$$

- 在第m次迭代中，Ada Boosting总是趋向于将具有最小误差（err_m）的学习模型选做本轮生成的弱分类器 G_m，使得累积误差快速下降。

Ada Boosting：例子

通过一个简单两类分类例子来介绍Ada Boosting算法过程。表4.8给出了10个数据点$x_i(i \in \{1,2,\ldots,10\})$取值及其所对应的类别标签$y_i \in \{1,-1\}(i \in \{1,2,\ldots,10\})$。

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

表4.8 两类分类问题数据

根据表4.8所给出的数据，要构造若干个弱分类器，然后将这些弱分类器组合为一个强分类器，完成表4.8所示数据的分类任务。

Ada Boosting：例子

这里定义每个弱分类器G为一种分段函数，由一个阈值ε构成，形式如下：

$$G(x_i)=\begin{cases}-1 & x_i<\varepsilon\\ 1 & x_i>\varepsilon\end{cases}\quad 或\quad G(x_i)=\begin{cases}1 & x_i<\varepsilon\\ -1 & x_i>\varepsilon\end{cases}$$

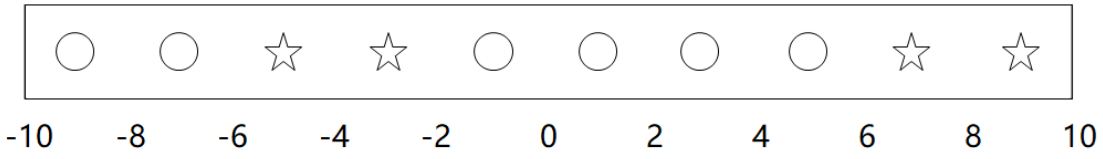
当然，在实际中，可根据需要使用其他的弱分类器。

Ada Boosting：例子

（1） 数据样本权重初始化

$$D_1 = (w_{1,1}, \ldots, w_{1,i}, \ldots, w_{1,10}),\ \text{其中} w_{1,i} = \frac{1}{10}(1 \le i \le 10)$$

下面用图来辅助说明算法流程。图中**圆圈**所代表数据点标签为-1、**五角星**所代表数据点标签为1，每个形状的颜色深浅代表这些数据被当前所学习得到（组合）分类器给出标签预测值大小，颜色越深表示越接近标签值-1、颜色越浅表示越接近标签值1。



Ada Boosting：例子

（2） 分别训练M个基分类器（弱分类器）

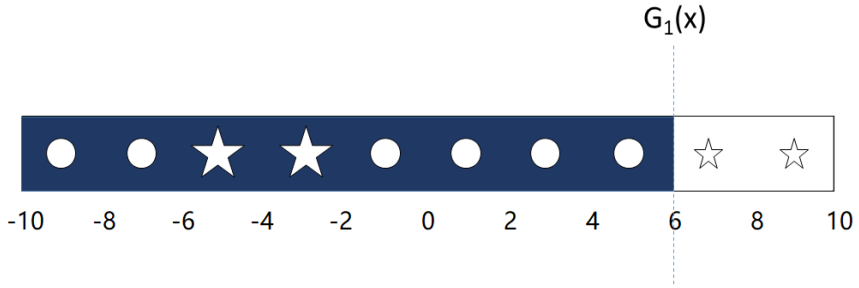
对$m=1$

- 使用具有分布权重D_1的训练数据来学习得到第$m=1$个基分类器G_1。不难看出，当阈值$\varepsilon=6$时，基分类器G_1具有最小错误率（err_m）。G_1分类器如下表示：
- $G_1(\mathrm{x_i})=\begin{cases}-1 & x_i<6\\ 1 & x_i>6\end{cases}$
- 计算$G_1(\mathrm{x})$在训练数据集上的分类误差，样例3、4被错误分类，因此G_1的分类误差为$err_1=\sum_{i=1}^{N} w_{1,i} I(G_1(x_i)\neq y_i)=0.1+0.1=0.2$
- 根据分类误差计算弱分类器$G_1(\mathrm{x})$的权重： $\alpha_1=\frac{1}{2}\ln\frac{1-err_1}{err_1}=\mathbf{0.6931}$
- 更新下一轮第$m=2$个分类器训练时第i个训练样本的权重： $D_2=\{w_{2,i}\}_0^{10}, w_{2,i}=\frac{w_{1,i}}{Z_1}e^{-\alpha_1 y_i G_1(x_i)}$，可得到数据样本新的权重：

  $D_2=(0.0625, 0.0625, 0.25, 0.25, 0.0625, 0.0625, 0.0625, 0.0625, 0.0625, 0.0625)$

- 通过加权线性组合得到当前的分类器$\boldsymbol{f_1(\mathrm{x})=\sum_{i=1}^{M}\alpha_m G_m(x)=0.6931 G_1(x)}$

在下图中，被分类错误数据样本的形状尺寸比其它数据样本形状稍大，以表示被分类错误的样本数据权重增大。



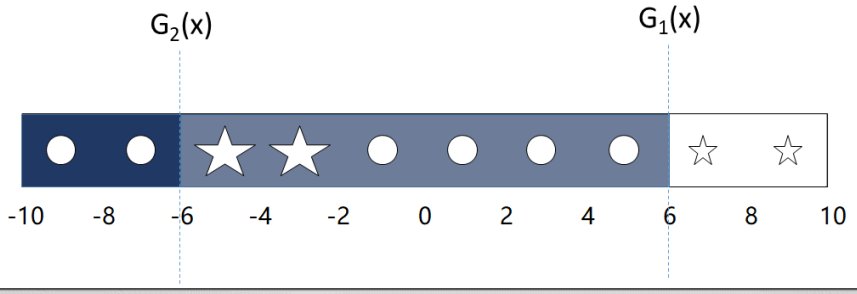
Ada Boosting：例子

对$m=2$

- 对于具有分布权重为D_2的训练数据，当阈值$\varepsilon=-6$时，基分类器G_2具有最小的错误率（err_m）。G_2分类器如下表示：

- $G_2(x_i)=\begin{cases}-1 & x_i<-6\\ 1 & x_i>-6\end{cases}$

- 分类误差$err_2=\sum_{i=1}^{N}w_{2,i}I(G_2(x_i)\neq y_i)=0.25$
- 弱分类器$G_2(x)$的权重$\alpha_2=\frac{1}{2}\ln\frac{1-err_2}{err_2}=\mathbf{0.5439}$
- 当进行下一轮分类器训练时，样本权重更新如下：$D_3=$ $(\mathbf{0.04166667, 0.04166667, 0.16666667, 0.16666667, 0.125, 0.125}$ $\mathbf{, 0.125, 0.125, 0.04166667, 0.04166667})$ 通过加权线性组合得到当前的分类器$\boldsymbol{f_2(x)=}$ $\mathbf{0.6931G_1(x)+0.5439G_2(x)}$



Ada Boosting：例子

对$m=3$

- 对于具有分布权重为D_3的训练样本数据，当阈值$\varepsilon=-2$时，基分类器G_3具有最小的错误率（err_m）。G_3分类器表示如下：

$$G_3(x_i)=\begin{cases}-1 & x_i>-2\\ 1 & x_i<-2\end{cases}$$

- 分类误差$err_3=\sum_{i=1}^{N}w_{3,i}I(G_3(x_i)\neq y_i)=0.1667$
- 弱分类器$G_3(x)$的权重$\alpha_3=\frac{1}{2}\ln\frac{1-err_3}{err_3}=\mathbf{0.8047}$
- 下一轮弱分类器训练时，训练数据样本的权重更新如下：$D_4=$ $(\mathbf{0.125},\mathbf{0.125},\mathbf{0.1},\mathbf{0.1},0.075,0.075,0.075,0.075,\mathbf{0.125},\mathbf{0.125})$
- 通过加权线性组合得到当前的分类器$\boldsymbol{f_3(x)=0.6931G_1(x)+0.5439G_2(x)+0.8047G_3(x)}$
- 在$f_3(x)$的基础上，构造强分类器$\boldsymbol{G(x)=sign\big(f_3(x)\big)=sign(0.6931G_1(x)+0.5439G_2(x)+0.8047G_3(x))}$。
- 这里$sign(\cdot)$是符号函数，其输入值大于0时，符号函数输出为1，反之为-1。由于$G(x)$在训练样本上分类错误率为0，算法终止，得到最终的强分类器。