比较全面的Adaboost算法总结（二）

原创 石头 机器学习算法那些事 2019-06-28

推荐阅读：比较全面的Adaboost算法总结（一）

目录

本文详细总结了AdaBoost算法的相关理论，第一篇文章相当于是入门AdaBoost算法，本文是第二篇文章，该文详细推导了AdaBoost算法的参数求解过程以及讨论了模型的过拟合问题。

AdaBoost算法的解释

AdaBoost算法是一种迭代算法，样本权重和学习器权重根据一定的公式进行更新，第一篇文章给出了更新公式，但是并没有解释原因，本节用前向分布算法去推导样本权重和学习器权重的更新公式。

1. 前向分布算法

考虑加法模型：

$$f(x)=\sum_{m=1}^{M}\beta_m b(x;\gamma_m) \qquad (1)$$

其中，$b(x;\gamma_m)$为基函数，γ_m为基函数的参数，β_m为基函数的系数

f(x)为基函数的线性组合

给定训练数据和损失函数L(y,f(x))的条件下，构建最优加法模型f(x)的问题等价于损失函数最小化：

$$\min_{\beta,\gamma}\sum_{i=1}^{N}L(y_i,\sum_{m=1}^{M}\beta_m b(x;\gamma_m)) \qquad (2)$$

我们利用前向分布算法来求解（2）式的最优参数，前向分布算法的核心是从前向后，每一步计算一个基函数及其系数，逐步逼近优化目标函数式（2），那么就可以简化优化的复杂度。

算法思路如下：

M-1个基函数的加法模型：

$$f_{M-1}(x)=\sum_{m=1}^{M-1}\beta_m b(x;\gamma_m) \qquad (3)$$

M个基函数的加法模型：

$$f_M(x)=\sum_{m=1}^{M}\beta_m b(x;\gamma_m) \qquad (4)$$

由（3）（4）得：

$$f_M(x)=f_{M-1}(x)+\beta_M b(x;\gamma_M) \qquad (5)$$

因此，极小化M个基函数的损失函数等价于：

$$\min_{\beta,\gamma}\sum_{i=1}^{N}L(y_i,f_{M-1}(x)+\beta_M b(x;\gamma_M)) \qquad (6)$$

前向分布算法的思想是从前向后计算，当我们已知$f_0(x)$的值时，可通过（6）式递归来计算第 i 个基函数$b(x,\gamma_i)$及其系数β_m，i = 1,2,...M。

结论：通过前向分布算法来求解加法模型的参数。

2. AdaBoost损失函数最小化

AdaBoost算法的强分类器是一系列弱分类器的线性组合：

$$f(x)=\sum_{m=1}^{M}\alpha_m G_m(x) \qquad (7)$$

其中f(x)为强分类器，共M个弱分类器$G_m(x)$，α_m是对应的弱分类器权重。

由（7）式易知，f(x)是一个加法模型。

AdaBoost的损失函数L(y,f(x))为指数函数：

$$L(y,f(x))=\exp(-y\cdot f(x)) \qquad (8)$$

利用前向分布算法最小化（8）式，可得到每一轮的弱学习器和弱学习器权值。第m轮的弱学习器和权值求解过程：

$$(\alpha_m,G_m(x))=\arg\min_{\alpha,G}\sum_{i=1}^{N}\exp(-y_i f_m(x_i))$$

$$\Rightarrow \arg\min_{\alpha,G}\sum_{i=1}^{N}\exp(-y_i(f_{m-1}(x)+\alpha_m G_m(x_i))$$

$$\Rightarrow \arg\min_{\alpha,G}\sum_{i=1}^{N}\overline{w}_{mi}\exp(-y_i\alpha_m G_m(x)) \qquad (9)$$

$$其中,\overline{w}_{mi}=\exp(-y_i(f_{m-1}(x))$$

首先根据（9）式来求解弱学习器，权值α看作常数：

(9)式展开得：

$$\arg\min_G\left(\sum_{i=1}^{N}\left[\exp(-\alpha_m)\cdot I(G_m(x_i)=y_i)\cdot\bar{w}_{mi}+\exp(\alpha_m)\cdot I(G_m(x_i)\neq y_i)\cdot\bar{w}_{mi}\right]\right) \quad (10)$$

∵权重精度和权重错误率的和等于1，即：

$$I(G_m(x_i)=y_i)\cdot\bar{w}_{mi}+I(G_m(x_i)\neq y_i)\cdot\bar{w}_{mi}=1 \quad (11)$$

由(10)(11)得：

$$\arg\min_G\left(\sum_{i=1}^{N}\left[I(G_m(x_i)\neq y_i)\cdot\bar{w}_{mi}[\exp(\alpha_m)-\exp(-\alpha_m)]+\exp(-\alpha_m)\right]\right) \quad (12)$$

α_m看成常数，最小化(12)式对应的弱学习器$G^*_m(x)$：

$$G^*_m(x)=\arg\min_G\sum_{i=1}^{N}\bar{w}_{mi}\cdot I(G_m(x_i)\neq y_i) \quad (13)$$

求解弱学习器$G_m^*(x)$后，（9）式对α求导并使导数为0，得：

$$\alpha_m^*=\frac{1}{2}\log\frac{1-e_m}{e_m} \quad (14)$$

其中，α是弱学习器权值，e为分类误差率：

$$e_m=\frac{\sum_{i=1}^{N}I(G_m(x_i)\neq y_i)\cdot\bar{w}_{mi}}{\sum_{i=1}^{N}\bar{w}_{mi}}=\sum_{i=1}^{N}I(G_m(x_i)\neq y_i)\cdot w_{mi} \quad (15)$$

因为AdaBoost是加法迭代模型：

$$f_m(x)=f_{m-1}(x)+\alpha_m G_m(x)$$

以及$\bar{w}_{mi}=\exp(-y_i f_{m-1}(x_i))$，得：

$$\bar{w}_{m+1,i}=\bar{w}_{mi}\cdot\exp(-y_i\alpha_m G_m(x)) \quad (16)$$

结论：式(14)(15)(16)与第一篇文章介绍AdaBoost算法的权重更新完全一致，即AdaBoost算法的权重更新与AdaBoost损失函数最优化是等价的，每次更新都是模型最优化的结果，（13）式的含义是每一轮弱学习器是最小化训练集权值误差率的结果。一句话，AdaBoost的参数更新和弱学习器模型构建都是模型最优化的结果。

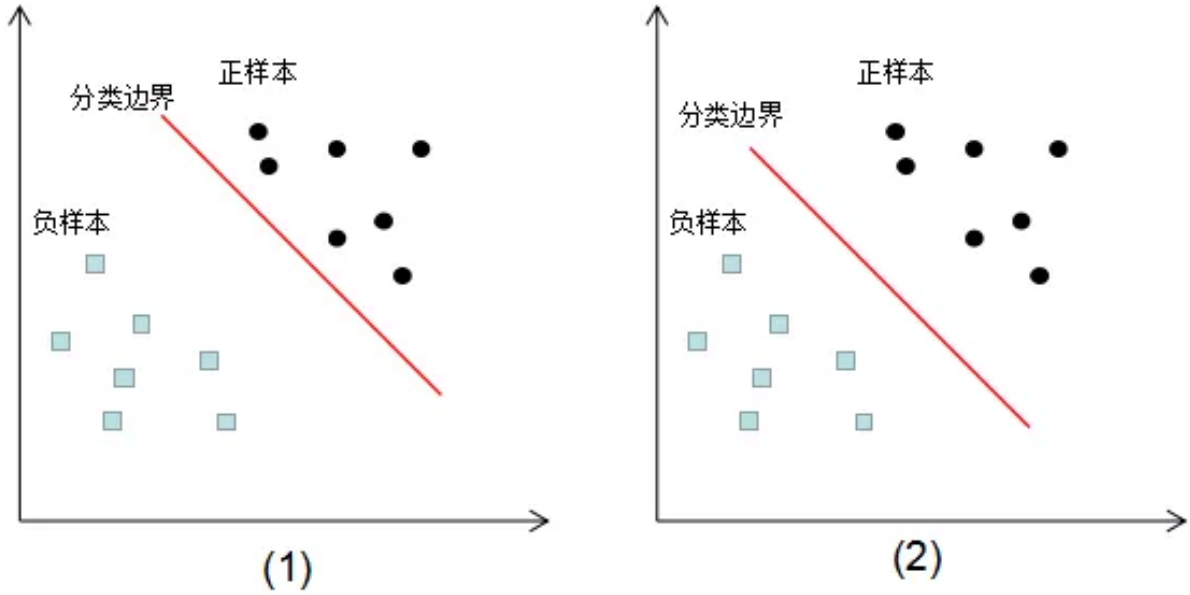
AdaBoost算法的过拟合问题讨论

1. 何时该讨论过拟合问题

模型的泛化误差可分解为偏差、方差与噪声之和。当模型的拟合能力不够强时，泛化误差由偏差主导；当模型的拟合能力足够强时，泛化误差由方差主导。因此，当模型的训练程度足够深时，我们才考虑模型的过拟合问题。

2. 问题的提出

如下图为同一份训练数据的不同模型分类情况：



图（1）（2）的训练误差都为0，那么这两种分类模型的泛化能力孰优孰劣？在回答这个问题，我想首先介绍下**边界理论**（Margin Theory）。

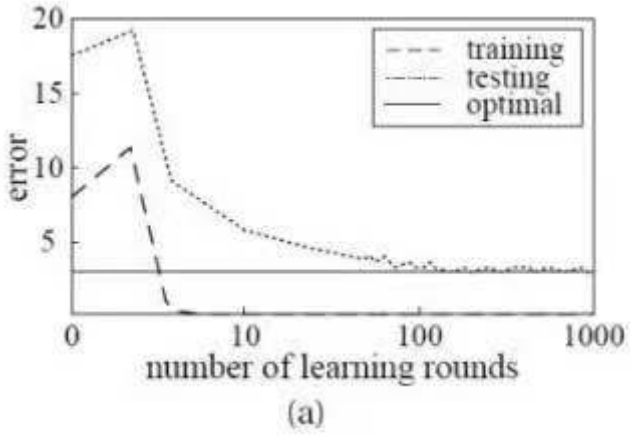
3. 边界理论

周志华教授在《集成学习方法基础与算法》证明了：

$$\varepsilon_D \le 2^T \prod_{t=1}^{T} \sqrt{\varepsilon_t^{1-\theta}(1-\varepsilon_t)^{1+\theta}} + \tilde{O}\left(\sqrt{\frac{d}{m\theta^2} + \ln\frac{1}{\delta}}\right) \qquad (17)$$

其中，ε_D 为泛化误差率，θ 为边界阈值。

由上式可知，泛化误差收敛于某个上界，训练集的边界（Margin）越大，泛化误差越小，防止模型处于过拟合情况。如下图：



结论：增加集成学习的弱学习器数目，边界变大，泛化误差减小。

4. 不同模型的边界评估

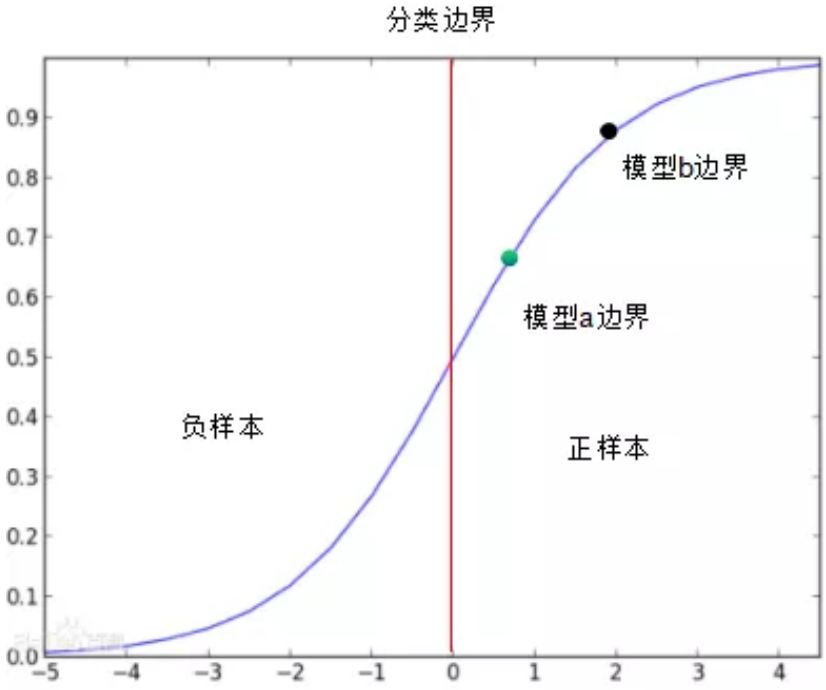
1） 线性分类模型的边界评估

用边界理论回答第一小节的问题

线性分类模型的边界定义为所有样本点到分类边界距离的最小值，第一小节的图（b）的边界值较大，因此图（b）的泛化能力较好。

2） logistic分类模型的边界评估

logistic分类模型的边界定义为所有输入样本特征绝对值的最小值，由下图可知，模型b边界大于模型a边界，因此，模型b的泛化能力强于模型a 。



3）AdaBoost分类模型边界评估

AdaBoost的强分类器：

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

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

AdaBoost的边界定义为f(x)的绝对值，边界越大，泛化误差越好。

当训练程度足够深时，弱学习器数目增加，f(x)绝对值增加，则泛化能力增强。

结论：AdaBoost算法随着弱学习器数目的增加，边界变大，泛化能力增强。

AdaBoost算法的正则化

为了防止AdaBoost过拟合，我们通常也会加入正则化项。AdaBoost的正则化项可以理解为学习率（learning rate）。

AdaBoost的弱学习器迭代：

$$f_k(x)=f_{k-1}(x)+\alpha_k G_k(x)$$

加入正则化项：

$$f_k(x) = f_{k-1}(x) + v\alpha_k G_k(x)$$

v的取值范围为：0 < v < 1。因此，要达到同样的训练集效果，加入正则化项的弱学习器迭代次数增加，由上节可知，迭代次数增加可以提高模型的泛化能力。

总结

> AdaBoost的核心思想在于样本权重的更新和弱分类器权值的生成，样本权重的更新保证了前面的弱分类器重点处理普遍情况，后续的分类器重点处理疑难杂症。最终，弱分类器加权组合保证了前面的弱分类器会有更大的权重，这其实有先抓总体，再抓特例的分而治之思想。

关于AdaBoost算法的过拟合问题，上两节描述当弱学习器迭代数增加时，泛化能力增强。AdaBoost算法不容易出现过拟合问题，但不是绝对的，模型可能会处于过拟合的情况：

（1）弱学习器的复杂度很大，因此选择较小复杂度模型可以避免过拟合问题，如选择决策树桩。adaboost + 决策树 = 提升树模型。

（2）训练数据含有较大的噪声，随着迭代次数的增加，可能出现过拟合情况。

希望这两篇文章能够打开你深入理解AdaBoost算法的大门。

参考

比较全面的AdaBoost算法总结（一）

【干货】集成学习原理总结

李航《统计学习方法》

周志华 《机器学习》

http://blog.sina.com.cn/s/blog_6ae183910101chcg.html

https://www.zhihu.com/question/41047671