Comprehensive Review: The AdaBoost Algorithm

Master's Level Data Science

Contents

1 Introduction

This review synthesizes the lecture slides (`ensemble-2.pdf`) and audio transcript (`AdaBoostAlgorithm.txt`) on the AdaBoost algorithm. We cover the motivation for boosting weak learners, the AdaBoost procedure, theoretical guarantees, a worked example, and practical considerations.

2 Motivation: Weak Learners and Boosting

A *weak learner* is an algorithm that, on any distribution over examples (x_i, y_i) with labels $y_i \in \{-1, +1\}$, returns a hypothesis h with error

$$\Pr\big(h(X) \neq Y\big) \;\leq\; \tfrac{1}{2} - \epsilon,$$

for some $\epsilon > 0$. Boosting is a method to convert such weak learners into a *strong learner* with arbitrarily low training error by combining multiple hypotheses in a weighted vote.

3 The AdaBoost Algorithm

Given a training set $\{(x_i, y_i)\}_{i=1}^n$, initialize weights

$$D_1(i) \;=\; \frac{1}{n}, \quad i = 1, 2, \ldots, n.$$

For rounds $t = 1, \ldots, T$:

1. Train weak learner on weighted sample D_t to obtain $h_t : X \to \{-1, +1\}$.
2. Compute weighted error
$$\varepsilon_t = \sum_{i=1}^{n} D_t(i)\,\mathbf{1}[h_t(x_i) \neq y_i].$$
3. Compute classifier weight
$$\alpha_t = \frac{1}{2}\ln\Big(\frac{1-\varepsilon_t}{\varepsilon_t}\Big).$$
4. Update weights for all i:
$$D_{t+1}(i) = \frac{D_t(i)\;\exp\big(-\alpha_t\, y_i\, h_t(x_i)\big)}{\sum_{j=1}^{n} D_t(j)\;\exp\big(-\alpha_t\, y_j\, h_t(x_j)\big)}.$$

The final strong classifier is

$$H(x) \;=\; sign\Big(\sum_{t=1}^{T} \alpha_t\, h_t(x)\Big).$$

4 Theoretical Guarantee

If each weak hypothesis has edge $\gamma_t = \frac{1}{2} - \varepsilon_t > 0$, then the training error of H satisfies

$$\frac{1}{n}\sum_{i=1}^{n} \mathbf{1}[H(x_i) \neq y_i] \;\leq\; \exp\Big(-2\sum_{t=1}^{T} \gamma_t^2\Big) \;\leq\; \exp\Big(-2T\gamma^2\Big),$$

where $\gamma = \min_t \gamma_t$. Thus training error decays exponentially in T.

5 Worked Example (Freund–Schapire)

We illustrate on a toy set with decision stumps as weak learners.

5.1 Example Data and Results

	D_1	D_2	D_3
h_1	$\varepsilon_1 = 0.40,\ \alpha_1 = 0.42$		
h_2		$\varepsilon_2 = 0.42,\ \alpha_2 = 0.37$	
h_3			$\varepsilon_3 = 0.30,\ \alpha_3 = 0.42$

After three rounds, the combined classifier is

$$H(x) = sign\big(0.42\, h_1(x) \;+\; 0.37\, h_2(x) \;+\; 0.42\, h_3(x)\big),$$

which can perfectly separate the training set.

6 Geometric Illustration

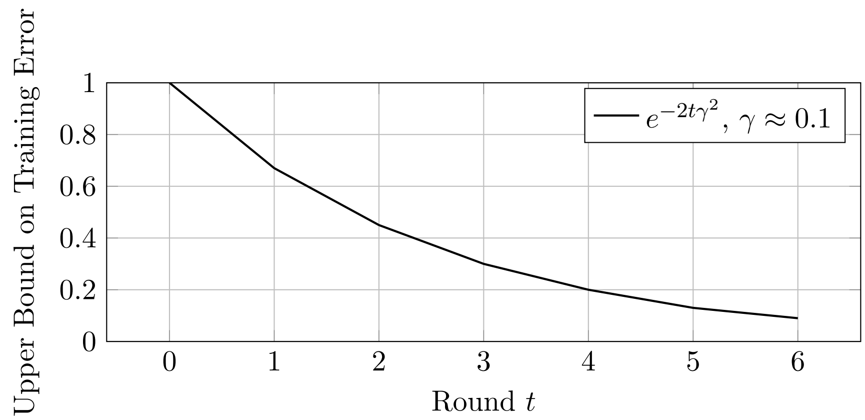

Figure 1: Exponential decay of the training error bound with rounds of boosting.

7 Algorithm Summary

1. Initialize uniform weights on examples.
2. For each round:
   (a) Train weak learner on weighted data.
   (b) Compute error and hypothesis weight.
   (c) Reweight examples to emphasize mistakes.
3. Output sign of weighted vote of hypotheses.

8 Practical Considerations

- **Choice of Weak Learner**: Decision stumps are common; deeper trees can be used.
- **Overfitting**: Monitor validation error; limiting T or adding shrinkage can help.
- **Computational Cost**: Each round requires retraining on weighted data.
- **Extensions**: Gradient boosting generalizes to arbitrary losses.

9 Future Directions

- **Gradient Boosting Machines**: Use differentiable loss and regression trees.
- **Regularization Schemes**: Subsample data (*bagging*), shrinkage.
- **Multi-class Boosting**: SAMME and variants for multiple labels.
- **Applications**: Ranking, regression, anomaly detection.