Introduction

Boosting combines many slightly-better-than-a-guess rules into a much more accurate rule. Boosting can be applied to any weak-learner.

- AdaBoost
- Relation to SVMs
- Experiments
- Pros and Cons
- Multi-class Classification
Ada-Boost

Start with $m$ training examples ($X$) and their outcomes ($Y = \{-1, +1\}$).
Let $D_1(i) = 1/m$

For $T$ training rounds, in the $t$th round we want to:

1. Train the weak learner using distribution $D_t$
2. Find the error $\epsilon_t$.
3. Update the coefficient $\alpha_t$.
4. Update the distribution $D_{t+1}$.

Adapted from “A Short Introduction to Boosting” - Freund, Schapire – p. 3/1
In each round we:

- Find the error for each hypothesis
  \[ \epsilon_t = Pr_{i \sim D_t}[h_t(x_i) \neq y_i] \]

- Give more weight to better performing learners:
  \[ \alpha_t = \frac{1}{2} \ln \left( \frac{1 - \epsilon_t}{\epsilon_t} \right) \]

- Update the distribution to prefer “harder” examples.
  \[ D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha y_i h_t(x_i))}{Z_t} \]
The end result is a weighted sum of the outputs of all hypotheses:

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$$
Generalization of AdaBoost

The generalization error of AdaBoost was shown to be at most:

\[
\hat{P}r[H(x) \neq y] + \tilde{O}\left(\sqrt{\frac{Td}{m}}\right)
\]

with high probability.

However, in early experiments AdaBoost would often not overfit, even after thousands of rounds. (Note that \(d\) is the VC Dimension of the weak hypothesis space.)

Adapted from “A Short Introduction to Boosting” - Freund, Schapire – p. 6/1
Furthermore, continuing to run AdaBoost after the training error had reached zero would sometimes continue to reduce the generalization error.

Figure 1: # rounds vs. error

Adapted from “A Short Introduction to Boosting” - Freund, Schapire – p. 7/1
Schapire et al. took a different approach and defined the margin of a training example to be:

$$\frac{y \sum_t \alpha_t h_t(x)}{\sum \alpha_t}$$

The magnitude of the margin can be a measure of confidence in the answer.
From this definition of margin the maximum generalization error is, with high probability:

$$\hat{P}r[\text{margin}(x, y) \leq \theta] + \tilde{O}\left(\sqrt{\frac{d}{m\theta^2}}\right), \quad \theta > 0$$

Adapted from "A Short Introduction to Boosting" - Freund, Schapire – p. 9/17
Relation to SVMs

In boosting, our goal is to maximize the minimum margin:

$$\max_{\alpha} \min_i \frac{(\alpha \cdot h(x_i)) y_i}{\|\alpha_i\| \|h(x_i)\|}$$

Which is the goal of SVMs, but using different distance metrics.

Adapted from “A Short Introduction to Boosting” - Freund, Schapire – p. 10/1
The norms for boosting are:

$$
\|\alpha\|_1 = \sum_t |\alpha_t| \quad \text{and} \quad \|h(x)\|_\infty = \max_t |h_t(x)|
$$

While the SVM norms are:

$$
\|\alpha\|_2 = \sqrt{\sum_t \alpha_t^2}, \quad \|h(x)\|_2 = \sqrt{\sum_t h_t(x)^2}
$$

Adapted from “A Short Introduction to Boosting” - Freund, Schapire – p. 11/17
The results is very different computational requirements, a different high dimensional search approach, and potentially very different margins.
Experiments

AdaBoost on decision trees:

boosting stumps

boosting C4.5

Adapted from “A Short Introduction to Boosting” - Freund, Schapire – p. 13/17
Experiments contd.

AdaBoost on text classification:

Adapted from “A Short Introduction to Boosting” - Freund, Schapire – p. 14/17
Advantages of AdaBoost

- AdaBoost is “Adaptive”, meaning it requires no prior knowledge about the learner to be boosted.
- Weak learners only need to be better than random.
- AdaBoost can identify outliers, mislabeled and ambiguous data points.

Adapted from “A Short Introduction to Boosting” - Freund, Schapire – p. 15/17
Pitfalls

- A large number of outliers (or noise) can be detrimental to performance.

- Performance is dependent on the weak learner chosen and the data trained on.
Multi-class Classification

- AdaBoost.M1, M2, MR, MH

Some methods formulate the problems in the form “Is the correct class $y$ or some other class?” Another considers the problem slightly differently, asking “Is the correct class $y$ or is it this other class $y'$?” Others make use of “error correcting output codes.”

Adapted from “A Short Introduction to Boosting” - Freund, Schapire – p. 17/1