**COS 424: Interacting with Data**

Lecturer: Dave Blei
Scribes: Parker Seidel & Peter J. DiFiore

Lecture #7
February 26, 2008

# 1 Boosting

- Easy to come up with rough rules of thumb for classifying data
  *- e.g., Doesn't contain "!!!" then HAM, contains "mortgage" then SPAM.*
- Rules are not great, but better than random
- Boosting converts these rough rules of thumb into an accurate classifier
- a class of ensemble methods

## 1.1 Sketch of an Boosting algorithm

- devise a weak learner that finds a weak hypothesis
- run on training data
- obtain weak hypothesis ($h_t$)
- reweight examples according to accuracy: Upweight misclassified data points and downweight correctly classified data points
- repeat T times to get 2nd, 3rd, ... Tth weak hypotheses ($h_2, h_2 \dots h_T$)
- at end, combine the weak hypthoeses into a strong classifier $H$.

## 1.2 The power of Boosting

- Boosting can drive the error down to an arbitrarily small positive number $\epsilon, \epsilon > 0$.
- That is, we can do as well as random or **we can obtain a training error 0**.
- Empirically, also does well at test time.

## 1.3 Illustrating the Boosting process - Toy example

For example, weak classifiers are horizontal and vertical lines. Two classes (+) and (-).
$D_t$: distribution over the training set.
$D_1$ $b_1$

### 1.4 Mathematical Description

We have Data: {$x_n$,$y_n$}; where $x_n$ is a document and $y_n$ = {-1,1} is a class
Ada.Boost Algorithm (Freund and Schapire, 1997)

- pronounced: Add-da-booost short for adaptive boosting
- Let $D_t$ be a distribution of weights over all data points at iteration $t$.
- $D_0(i) = \frac{1}{N}$ [- all data are equally likely]
- for $t = 1 \dots T$
  - Run weak learner on the training data weighted by $D_t$ to obtain hypothesis $h_t$
  - Reweight the data distribution based on the error rate of the hypothesis $h_t$
- Define $\epsilon_t = P_{D_t}(h_t(x_i) \neq y_i)$, the probability that under the current distribution, $D_t$, the predicted class,$h_t(x_i)$, does not equal the true class, $y_i$. When the weights are uniform this probability is simply $\frac{\#\text{ of misclassifications}}{\#\text{ total data points}}$. For non uniform weighting, this probability is sum of the weighted distribution over those data points that were misclassified.
- $\alpha = \frac{1}{2}\log(\frac{1-\epsilon_t}{\epsilon_t})$, note that $\alpha_t$ is big because $\epsilon \in \{0, 0.5\}$
- The update step in the algorithm takes the following form:
  $D_{t+1}(i) = \frac{1}{Z_t} D_t(i) * \begin{cases} \epsilon^{\alpha_t} & \text{if } h_t(x_i) \neq y_i \\ \epsilon^{-\alpha_t} & \text{if } h_t(x_i) = y_i \end{cases}$
- $Z_t$ is a constant for R.H.S. to sum to 1: $\sum_{i=1}^{N} D_{t+1}(i) = 1$.
- After T rounds, the predicted class ($H(x)$)is a weighted combination of the predicted classes of each weak hypothesis. This known as a **weighted majority vote**.
- H(x) = $\text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x_i)\right)$

### 1.5 Example on text data

usenet newsgroups - discussion forum from beginning of the internet. Benchmark data set in text classification. Goal: predict the forum of a given message.

Binary classification - we consider 1000 × 1000 matrix, M. $m_{ij}$ is # $j^{th}$ word occurs in $i^{th}$ document. Adaboost w/ weak learners being presence of a single word. Classify articles from alt.atheism vs. NOT from alt.atheism. A plot of the training accuracy and test accuracy of the classifier using $0 - 100$ rounds. As the number of rounds increased to 100, the training accuracy went from 0.92 to 1.0 (perfect classification). The training accuracy performed as well about 0.95.

There were several Important things to take from this example:

- The training accuracy almost always increases with more rounds.
- The test accuracy doesn't necessarily increase with more rounds.
- Choosing the number of rounds is an important part of Boosting.

## 2 Theoretical Error: how well does Boosting do?

One of the strong points of the Boosting algorithm is that one can prove an upper limit of the training error under the assumptions of the weak hypothesis ("rules of thumb").

**Theorem 2.1. An Upper Bound on the Training Error***:*
*The training error of the final, combined classifier* $H(x)$ *is at most* $exp\{-2 * \sum_{t=1}^{T} \gamma_t^2\}$, *where* $\gamma_t = \frac{1}{2} - \epsilon_t$.

*Proof.* The proof of this theorem requires three steps:

1. Show that
$$D_{T+1}(i) = \frac{1}{N} \frac{-y_i F(x_i)}{\prod_t Z_t}$$
$$\text{where, } F(x_i) = \sum_t \alpha_t h_t(x_i)$$

2. Show that the training error of the final classifier $H$ is at most
$$\prod_{t=1}^{T} Z_t$$

3. Combine step 1 and step 2 and rewrite $Z_t$ to show
$$\text{error} \leq \exp\{-2 \sum_{t=1}^{T} \gamma_t^2\}$$
where $\gamma_t = \frac{1}{2} - \epsilon_t$.

□

1. *Proof.* Show that $D_{T+1}(i) = \frac{1}{N} \frac{-y_i F(x_i)}{\prod_t Z_t}$
Recall that our update step took the following form:
$$D_{t+1}(i) = \frac{1}{Z_t} D_t(i) * \begin{cases} \epsilon^{\alpha_t} & \text{if } h_t(x_i) \neq y_i \\ \epsilon^{-\alpha_t} & \text{if } h_t(x_i) = y_i \end{cases}$$
where $h_t(x_i)$ and $y_i \in \{-1, 1\}$. We thus can recognize that
$$h_t(x_i) * y_i = \begin{cases} 1 & \text{if } h_t(x_i) = y_i \\ -1 & \text{if } h_t(x_i) \neq y_i \end{cases}$$
Recognizing this equivalency allows us substitute the product $h_t(x_i)y_i$ into our original update equation:
$$D_{t+1} = \frac{D_t(i) \cdot \exp\{-\alpha_t h_t(x_i) y_i\}}{Z_t}$$

Above we defined $D_0(i) = \frac{1}{N}, \forall\, i = \{1, \cdots, N\}$ so to equally weight each data point. Using our definition for $D_0(i)$ we can recursively apply the update step so to arrive at a formula for $D_{T+1}$:

$$D_{T+1}(i) = \underbrace{D_0(i) \cdot \frac{\exp\{-\alpha_1 h_1(x_i) y_i\}}{Z_1}}_{D_1(i)} \cdots \frac{\exp\{-\alpha_T h_T(x_i) y_i\}}{Z_T}$$

We can define $F(x_i) = \sum_{t=1}^{T} \alpha_t h_t(x_i)$, substitute in $D_0(i) = \frac{1}{N}$ and utilize the face that exponents of the same base sum when multiplying in order to simply our formula:

$$D_{T+1}(i) = \frac{1}{N} \frac{\exp\{-y_i \sum_{t=1}^{T} \alpha_t h_t(x_i)\}}{\prod_{t=1}^{T} Z_t} = \frac{1}{N} \frac{\exp\{-y_i F(x_i)\}}{\prod_{t=1}^{T} Z_t} \tag{1}$$

□

2. *Proof.* Show that the training error of the final classifier $H$ is at most $\prod_{t=1}^{T} Z_t$.

   Let us define the fractional training error as the number of incorrect classifications divided by the total number of training data points:

$$\text{error} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}[F(x_i) y_i \leq 0]$$

   Remember that $F(x_i)$ is the predicted class of $x_I$ using the "weighted majority vote" of all the individual weak hypotheses $h_t(x_i)$. Therefore $\mathbf{1}[F(x_i) y_i \leq 0]$ is an indicator variable that $x_i$ was misclassified. We can easily write an upper bound on this error:

$$\text{error} \leq \frac{1}{N} \sum_{i=1}^{N} \exp\{-y_i F(x_i)\} \tag{2}$$

   This bound comes from the property of the exponential. Let $a = y_i F(x_i)$ and consider the two cases:

$$a \leq 0 \longrightarrow e^{-a} \geq 1 = \mathbf{1}[a \leq 0]$$
$$a > 0 \longrightarrow \mathbf{1}[a \leq 0] = 0 < e^{-a}$$

   Thus in both cases, for an individual $x_i$, the indicator variable is upper-bounded and thus the sum is also upper-bounded.

   If we rewrite equation 1 as:

$$D_{T+1}(i) \prod_{t=1}^{T} Z_t = \frac{1}{N} \exp\{-y_i F(x_i)\} \tag{3}$$

   If substitute this into equation 2 we can rewrite the upper bound as:

$$\begin{aligned} \text{error} &\leq \sum_{i=1}^{N} \left( D_{T+1}(i) \prod_{t=1}^{T} Z_t \right) \\ &\leq \prod_{t=1}^{T} Z_t \cdot \underbrace{\sum_{i=1}^{N} D_{T+1}(i)}_{\text{by definition normalized to 1}} \end{aligned}$$

□

Through out definition of $Z_t$, we normalized $D_{t+1}$ to sum to one.

Therefore, we have proved an upper bound on the test error of final classifier using T weak hypotheses:

$$\text{error} \leq \prod_{t=1}^{T} Z_t \tag{4}$$

3. *Proof.* Let us examine how we can use the definition of $Z_t$ as a normalization constant to decompose it into two terms. Recall that $Z_t$ is defined such that $\sum_{i=1}^{N} D_{t+1}(i) = 1$, where $D_{t+1}$ was defined as:

$$D_{t+1}(i) = \frac{1}{Z_t} D_t(i) * \begin{cases} \epsilon^{\alpha_t} & \text{if } h_t(x_i) \neq y_i \\ \epsilon^{-\alpha_t} & \text{if } h_t(x_i) = y_i \end{cases} \tag{5}$$

Therefore, in order for $D_{t+1}$ to sum to one, $Z_t$ must be defined as:

$$Z_t = \underbrace{\sum_{i:h_t(xi)=y_i} D_t(i)e^{\alpha_t}}_{\text{correct classification}} + \underbrace{\sum_{i:h_t(xi)\neq y_i} D_t(i)e^{-\alpha_t}}_{\text{incorrect classification}} \tag{6}$$

Notice that the exponential is independent of $i$ and can therefore be removed from the sum:

$$Z_t = e^{\alpha_t} \underbrace{\sum_{i:h_t(xi)=y_i} D_t(i)}_{=(1-\epsilon_t)\text{weighted accuracy}} + e^{-\alpha_t} \underbrace{\sum_{i:h_t(xi)\neq y_i} D_t(i)}_{=\epsilon_t \text{ weighted error}} \tag{7}$$

where $\epsilon = P_{D_t}(h_t(x_i) \neq y_i)$: the probability under the current weights (distribution) that the predicted class is incorrect. If we think of $D_t$ as a probability distribution, then the summation of all the incorrect classifications is exactly $\epsilon_t$.

$$Z_t = e^{\alpha_t}(1 - \epsilon_t) + e^{-\alpha_t}(\epsilon_t) \tag{8}$$

The definition of $\alpha_t = \frac{1}{2}\log\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$ was chosen to minimize $Z_t$ and therefore minimize the training error in the final classifier. Using this definition of $\alpha_t$, the error can written more succinctly as:

$$\begin{aligned} Z_t &= 2\sqrt{\epsilon_t(1-\epsilon_t)} \\ &= \sqrt{1 - 4\gamma_t^2}, \text{ where } \gamma_t = \frac{1}{2} - \epsilon_t \end{aligned}$$

Using the property that $1 + x \leq e^x$, where $x = -2\gamma_t^2$, we can further upper bound the error:

$$Z_t \leq e^{-2\gamma_t^2} \tag{9}$$

□

4. *Proof.* By combining equations 4 and 9 from steps 2 and 3 we can arrive at the final upper bound of the classifier $H(x)$ using T weak hypotheses:

$$\text{error} \leq \exp\{-2\sum_{t=1}^{T}\gamma_t^2\} \tag{10}$$

where $\gamma_t = \frac{1}{2} - \epsilon_t$. □