Motivation
It is usually hard to learn an accurate predictor directly. Main Idea

1. Use weak learning algorithm to build a strong learner (an accurate PAC-learning algorithm)
2. Ensemble method: combine different base classifiers returned by a weak learner to create a more accurate predictor
Outline

1. Weak Learner

2. Adaboost
   - Algorithm
   - Bound on the Empirical Error

3. Boosting as Gradient Descent
   - Generalization of AdaBoost
Definition 6.1 Weak learning

A concept class \( C \) is said to be weakly PAC-learnable if there exists an algorithm \( A \), \( \gamma > 0 \), and a polynomial function \( \text{poly}(\cdot, \cdot, \cdot, \cdot) \) such that for any \( \varepsilon > 0 \) and \( \delta > 0 \), for all distributions \( D \) on \( X \) and for any target concept \( c \in C \), the following holds for any sample size \( m \geq \text{poly}(1/\delta, n, \text{size} (c)) \):

\[
\Pr_{S \sim D^m} [R(h_S) \leq \frac{1}{2} - \gamma] \geq 1 - \delta.
\]

When such an algorithm \( A \) exists, it is called a weak learning algorithm for \( C \) or a weak learner. The hypotheses returned by a weak learning algorithm are called base classifiers.
Outline

1. Weak Learner

2. Adaboost
   - Algorithm
   - Bound on the Empirical Error

3. Boosting as Gradient Descent
   - Generalization of AdaBoost
Algorithm 1 AdaBoost

1: for $i = 1$ to $m$ do
2:   $D_1(i) \leftarrow \frac{1}{m}$
3: end for
4: for $t = 1$ to $T$ do
5:   $h_t \leftarrow \arg\min_{h \in H} \Pr_i D_t[h_t(x_i) \neq y_i]$ 
6:   $\alpha_t \leftarrow \frac{1}{2} \log \frac{1-\epsilon_t}{\epsilon_t}$
7:   $Z_t \leftarrow 2[\epsilon_t(1-\epsilon_t)]^{\frac{1}{2}}$ (normalization factor)
8:   for $i=1$ to $m$ do
9:     $D_{t+1}(i) \leftarrow \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$
10:   end for
11: end for
12: $g \leftarrow \sum_{t=1}^{T} \alpha_t h_t$
Illustrative Example

(a) $\alpha_1 + \alpha_2 + \alpha_3 = \text{updated weights}$

(b) $\text{decision boundary}$
**Thm** The empirical error of the classifier returned by AdaBoost verifies

\[
\hat{R}(h) \leq \exp\left[-2 \sum_{t=1}^{T} T\left(\frac{1}{2} - \epsilon_t\right)^2\right]
\]

If for all \( t \in [1, T] \), exists \( \gamma \leq \frac{1}{2} - \epsilon_t \), then

\[
\hat{R}(h) \leq \exp(-2\gamma^2 T)
\]
Proof

\[ D_{t+1}(i) = \frac{D_t(i) e^{-\alpha_t y_i h_t(x_i)}}{Z_t} = e^{-y_i \sum_{s=1}^{t} \alpha_s h_s(x_i)} \]

(1)

\[ Z_t = \sum_{i=1}^{m} D_t(i) e^{-\alpha_t y_i h_t(x_i)} \]

(2)

\[ = \sum_{i: y_i h_t(x_i) = +1} D_t(i) e^{-\alpha_t} + \sum_{i: y_i h_t(x_i) = -1} D_t(i) e^{\alpha_t} \]

(3)

\[ = (1 - \epsilon_t) e^{-\alpha_t} + \epsilon_t e^{\alpha_t} = 2 \sqrt{\epsilon_t (1 - \epsilon_t)} \]

(4)

\[ \prod_{t=1}^{T} Z_t = \prod_{t=1}^{T} 2 \sqrt{\epsilon_t (1 - \epsilon_t)} = \prod_{t=1}^{T} \sqrt{1 - 4 \left(\frac{1}{2} - \epsilon_t\right)^2} \]

(5)

\[ \leq \prod_{t=1}^{T} \exp[-2 \left(\frac{1}{2} - \epsilon_t\right)^2] = \exp[-2 \sum_{t=1}^{T} \left(\frac{1}{2} - \epsilon_t\right)^2] \]
\[
\hat{R}(h) = \frac{1}{m} \sum_{i=1}^{m} 1_{y_i g(x_i) \leq 0} \\
\leq \frac{1}{m} \sum_{i=1}^{m} e^{-y_i g(x_i)} \\
= \frac{1}{m} \sum_{i=1}^{m} [m \prod_{t=1}^{T} Z_t] D_{t+1}(i) \\
= \prod_{t=1}^{T} Z_t \\
\leq \exp[-2 \sum_{t=1}^{T} \left(\frac{1}{2} - \epsilon_t\right)^2]
\]
Weak Learner
Adaboost
Boosting as Gradient Descent
Algorithm
Bound on the Empirical Error

Yipei Wang
Boosting
Outline

1. Weak Learner

2. Adaboost
   - Algorithm
   - Bound on the Empirical Error

3. Boosting as Gradient Descent
   - Generalization of AdaBoost
Algorithm 1: AnyBoost

Require:
- An inner product space $(\mathcal{X}, \langle, \rangle)$ containing functions mapping from $X$ to some set $Y$.
- A class of base classifiers $\mathcal{F} \subseteq \mathcal{X}$.
- A differentiable cost functional $C: \text{lin} (\mathcal{F}) \to \mathbb{R}$.
- A weak learner $L(F)$ that accepts $F \in \text{lin} (\mathcal{F})$ and returns $f \in \mathcal{F}$ with a large value of $-\langle \nabla C(F), f \rangle$.

Let $F_0(x) := 0$.
for $t := 0$ to $T$ do
  Let $f_{t+1} := L(F_t)$.
  if $-\langle \nabla C(F_t), f_{t+1} \rangle \leq 0$ then
    return $F_t$.
  end if
  Choose $w_{t+1}$.
  Let $F_{t+1} := F_t + w_{t+1} f_{t+1}$
end for
return $F_{T+1}$. 
\[
\frac{dF(\alpha_{t-1} + \eta e_t)}{d\eta} \bigg|_{\eta=0} = -\sum_{i=1}^{m} y_i h_t(x_i) \exp \left[-y_i \sum_{s=1}^{t-1} \alpha_s h_s(x_i)\right] \\
= -\sum_{i=1}^{m} y_i h_t(x_i) D_t(i) \left[m \prod_{s=1}^{t-1} Z_s\right] \\
= -\left[\sum_{i:y_i h_t(x_i)=+1} D_t(i) - \sum_{i:y_i h_t(x_i)=-1} D_t(i)\right] \left[m \prod_{s=1}^{t-1} Z_s\right] \\
= -[(1 - \epsilon_t) - \epsilon_t] \left[m \prod_{s=1}^{t-1} Z_s\right] = [2\epsilon_t - 1] \left[m \prod_{s=1}^{t-1} Z_s\right].
\]
\[
\frac{dF(\alpha_{t-1} + \eta e_t)}{d\eta} = 0 \iff - \sum_{i=1}^{m} y_i h_t(x_i) \exp \left[ - y_i \sum_{s=1}^{t-1} \alpha_s h_s(x_i) \right] e^{-\eta y_i h_t(x_i)} = 0
\]

\[
\iff - \sum_{i=1}^{m} y_i h_t(x_i) D_t(i) \left[ m \prod_{s=1}^{t-1} Z_s \right] e^{-y_i h_t(x_i) \eta} = 0
\]

\[
\iff - \sum_{i=1}^{m} y_i h_t(x_i) D_t(i) e^{-y_i h_t(x_i) \eta} = 0
\]

\[
\iff - (1 - \epsilon_t) e^{-\eta} - \epsilon_t e^\eta = 0
\]

\[
\iff \eta = \frac{1}{2} \log \frac{1 - \epsilon_t}{\epsilon_t}.
\]
Existing voting methods viewed as AnyBoost on margin cost functions.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Cost function</th>
<th>Step size</th>
</tr>
</thead>
<tbody>
<tr>
<td>AdaBoost [9]</td>
<td>$e^{-yF(x)}$</td>
<td>Line search</td>
</tr>
<tr>
<td>ARC-X4 [2]</td>
<td>$(1 - yF(x))^5$</td>
<td>$1/t$</td>
</tr>
<tr>
<td>ConfidenceBoost [19]</td>
<td>$e^{-yF(x)}$</td>
<td>Line search</td>
</tr>
<tr>
<td>LogitBoost [12]</td>
<td>$\ln(1 + e^{-yF(x)})$</td>
<td>Newton-Raphson</td>
</tr>
</tbody>
</table>