Nick Ketz
CSCI 5622
Homework #5 Boosting
03/13/2015

Problems 6.3 and 6.6 from the Foundations of Machine Learning textbook

6.3 Update guarantee. Assume that the main weak learner assumption of AdaBoost holds. Let h_t be the base learner selected at round t. Show that the base learner h_{t+1} selected at round $t+1$ must be different from h_t.

from the algorithm in Figure 6.1:
$D_0 = \frac{1}{m}$
$h_t \in \underset{h \in H}{\operatorname{argmin}} D_t(i) 1_{h(x_i) \neq y_i}$
h_t is base learner in H, i.e. $\epsilon_t = \Pr_{i \sim D_t}[h_t(x_i) \neq y_i] < 1/2$
$\alpha_t = \frac{1}{2} \log(\frac{1-\epsilon_t}{\epsilon_t})$
$D_{t+1}(i) = D_t(i) \cdot \exp(-\alpha_t y_i h_t(x_i)) \cdot Z_t^{-1}$

assume:
$h_t = h_{t+1}$

implies:
$\underset{h \in H}{\operatorname{argmin}} D_t \cdot 1_{h(X) \neq Y} = \underset{h \in H}{\operatorname{argmin}} D_{t+1} \cdot 1_{h(X) \neq Y}$
$D_t \cdot 1_{h_t(X) \neq Y} = D_{t+1} \cdot 1_{h_t(X) \neq Y}$
$D_t = D_{t+1}$

ignoring normalization factor Z, which rescales D after its calculated
$D_t(i) = (1/m) \cdot \exp(-\alpha_t y_i h_t(x_i)) = D_t(i) \cdot \exp(-\alpha_{t+1} y_i h_t(x_i))$
$1 = \exp(-\alpha_{t+1} y_i h_t(x_i))$
$0 = -\alpha_{t+1} y_i h_t(x_i)$
$0 = \frac{1}{2} \log(\frac{1-\epsilon_{t+1}}{\epsilon_{t+1}})$
$e^0 = \frac{1-\epsilon_{t+1}}{\epsilon_{t+1}}$
$\epsilon_{t+1} = 1 - \epsilon_{t+1}$
$\epsilon_{t+1} = 1/2$
$\therefore$ $h_t = h_{t+1}$ Contradicts base learner assumption

$\epsilon_t < \frac{1}{2}$, based on the weak learner assumption, forces α to reweigh the sample distribution through D_{t+1}, forcing $h_{t+1} \neq h_t$ ■

6.6 Fix $\epsilon \in \{0, \frac{1}{2}\}$. Let the training sample be defined by m points in the plane with $m/4$ negative points all at coordinate $(1,1)$, another $m/4$ negative points all at coordinate $(-1,-1)$, $\frac{m(1-\epsilon)}{4}$ positive points all at coordinate $(1,-1)$, and $\frac{m(1+\epsilon)}{4}$ positive points all at coordinate $(-1,+1)$. Describe the behavior of AdaBoost when run on this sample using boosting stumps. What solution does the algorithm return after T rounds?

consider four sets each with $m/4$ points:
$A : y = -1$ at $(-1,-1), B : y = -1$ at $(1,1)$
$C : y = +1$ at $(1,-1), D : y = +1$ at $(-1,1)$

$D_0 = 1/m$, equally weighting of all points
$h_0 =$ horizontal line through origin implying positive for $x_1 > 0$
$\epsilon_0 = \frac{1}{m}(B+C) = \frac{1}{m}(\frac{m}{4} + \frac{m}{4}) = \frac{1}{2}$
$\alpha_0 = \frac{1}{2}\log\frac{1-\epsilon_0}{\epsilon_0} = 0$
$Z_0 = 2[\epsilon_0(1-\epsilon_0)]^{1/2} = 2[(\frac{1}{2}^2)^{1/2}] = 1$
$D_1 = D_0 \cdot \exp(-\alpha_0 Y h_0(X)) \cdot Z_0^{-1} = \frac{1}{m} \cdot 1 \cdot 1^{-1} = 1/m$

h_1 = vertical line through origin, gives the same ϵ:
$\epsilon_1 = \frac{1}{m}(A+D) = \frac{1}{m}(\frac{m}{4} + \frac{m}{4}) = \frac{1}{2}$
implies $D_2 = D_1 = D_0 = 1/m$

There can be no change through T iterations as ϵ will always be $1/2$.

Final solution:
$h =$sgn(g_T)
$g_T = \sum_{i=1}^{T} \alpha_i h_i = 1/2(h_0 + h_1)$

$h(A) = 1/2(-m/4 + m/4) = 0$
$h(B) = 1/2(m/4 - m/4) = 0$
$h(C) = 1/2(m/4 - m/4) = 0$
$h(D) = 1/2(-m/4 + m/4) = 0$

■