Mehryar Mohri
Foundations of Machine Learning
Courant Institute of Mathematical Sciences
Homework assignment 4 - solution
Due: April 18th, 2008
Credit: Ashish Rastogi, Afshin Rostamizadeh
Ameet Talwalkar, and Eugene Weinstein.

1. **Problem 1:** Consider the following formulation of Adaboost. As in class, we start with a training set of labelled examples: $\{(\mathbf{x}_i, y_i)\}_{i=1,\ldots,m}$, where $(\mathbf{x_i}, y_i) \in \chi \times \{-1, 1\}$. Let $\mathcal{H} = \{h_1, \ldots, h_n\}$ be the set of weak classifiers where $h_j : \chi \to \{1, -1\}$ (note: we assume a finite number n of weak classifiers, where $m \ll n$). We define an $m \times n$ matrix $\mathbf{M}$ where $M_{ij} = y_i h_j(\mathbf{x}_i)$, i.e., $M_{ij} = +1$ if training example i is classified correctly by weak classifier h_j, and -1 otherwise. Let $d_t, \lambda_t \in \mathbb{R}^n$, $\| d_t \|_1 = 1$ and $d_{t,i}$ ($\lambda_{t,i}$) equal i^{th} component of d_t (λ_t). Now we define the following algorithm:

   (a) Input: Matrix $\mathbf{M}$, Number of iterations t_{max}
   (b) Initialize: $\lambda_{1,j} = 0$ for $j = 1, \ldots, n$
   (c) Loop for $t = 1, \ldots, t_{max}$
      i. $d_{t,i} = \frac{\exp(-(\mathbf{M}\boldsymbol{\lambda}_t)_i)}{\sum_{k=1}^{m} \exp(-(\mathbf{M}\boldsymbol{\lambda}_t)_k)}$ for $i = 1, \ldots, m$
      ii. $j_t \in \text{argmax}_j (\mathbf{d}_t^T \mathbf{M})_j$
      iii. $r_t = (\mathbf{d}_t^T \mathbf{M})_{j_t}$
      iv. $\alpha_t = \frac{1}{2} \ln \left(\frac{1+r_t}{1-r_t} \right)$
      v. $\boldsymbol{\lambda}_{t+1} = \boldsymbol{\lambda}_t + \alpha_t \mathbf{e}_{j_t}$, where $\mathbf{e}_{j_t}$ is 1 in position j_t and 0 elsewhere.
   (d) Output: $\frac{\boldsymbol{\lambda}_{tmax}}{\|\boldsymbol{\lambda}_{tmax}\|_1}$

5 points Is this approach of explicitly using M practical? Why/Why not?
Solution: Since n is large, it is not practical to store $\mathbf{M}$.

5 points What does $d_{1,i}$ equal for $t = 1$ for each value of i?
Solution: $d_{1,i} = \frac{1}{m}$.

5 points In one sentence, explain what is happening in step 3a.
Solution: The distribution over the training samples is being updated and normalized based on the results from the weak classifier chosen in the previous round of boosting.

5 points $(\mathbf{d}_t^T\mathbf{M})_j$ is called the "edge" of weak classifier j at time t w.r.t. the training examples. What are the max and min values for the edge of a weak classifier at time t?

Solution: min $= -1$; max $= 1$.

5 points What do large and small values of r_t tell us about the classifier?

Solution: r_t is the edge of the "best" weak classifier over distribution d_t. A larger (smaller) edge indicates a lower (higher) probability of error for this "best" weak classifier on the training set over d_t.

5 points How would you write the combined classifier $H(x)$ as defined in lecture in terms of $\boldsymbol{\lambda}_{tmax}$?

Solution: Let $f = \sum_i^n \left(\frac{\boldsymbol{\lambda}_{tmax}}{\|\boldsymbol{\lambda}_{tmax}\|_1}\right)_i h_i$. Then $H(x) = \text{sign}(f(x))$.

Problem 2

The explicit mapping between d_t and D_{t+1} for the algorithm presented in Problem 1 can defined as follows:

1. $j_t \in \text{argmax}_j(\mathbf{d}_t^T\mathbf{M})_j$
2. $r_t = (\mathbf{d}_t^T\mathbf{M})_{j_t}$
3. $d_{t+1,i} = \frac{d_{t,i}}{1+M_{ij_t}r_t}$ for $i = 1, \ldots, m$

5 points Let d_- be the probability of error of weak classifier h_{j_t} at iteration t. Define d_- as a summation over entries in $\mathbf{M}$.

Solution: $d_- = \sum_{i:M_{ij_t=-1}} d_{t,i}$.

5 points Write an expression for edge r_t in terms of d_-.

Solution:
$r_t = \sum_{i:M_{ij_t=1}} d_{t,i} - \sum_{i:M_{ij_t=-1}} d_{t,i} = (1 - d_-) - d_- = 1 - 2d_-$.

10 points Assuming d_t is normalized, show that d_{t+1} remains normalized, i.e., $\sum_i^m d_{t+1,i} = 1$.

Solution: When $M_{ij_t} = 1, d_{t+1,i} = \frac{d_{t,i}}{1+r_t}$ and when $M_{ij_t} = -1, d_{t+1,i} = \frac{d_{t,i}}{1-r_t}$. Defining $d_+ = (1 - d_-)$ we have:

$$\sum_i^m d_{t+1,i} = \frac{1}{1+r_t}d_+ + \frac{1}{1-r_t}d_- \tag{1}$$

Rearranging expression for r_t from previous question, we have:

$$\sum_i^m d_{t+1,i} = \frac{(1+r_t)}{2(1+r_t)} + \frac{(1-r_t)}{2(1-r_t)} = 1 \tag{2}$$

10 points Show that Adaboost sets the edge of the previous weak classifier to 0, i.e., $(\mathbf{d}_{t+1}^T\mathbf{M})_{j_t} = 0$.

Solution: Based on the definition of an edge we have:

$$(\mathbf{d}_{t+1}^T\mathbf{M})_{j_t} = \sum_{i:M_{ij_t=1}} d_{t+1,i} - \sum_{i:M_{ij_t=-1}} d_{t+1,i} \tag{3}$$

Using the mapping defined at the beginning of this question, we get:

$$= \sum_{i:M_{ij_t=1}} d_{t,i}\frac{1}{1+r_t} - \sum_{i:M_{ij_t=-1}} d_{t,i}\frac{1}{1-r_t} \tag{4}$$

Using the definitions of d_+ and d_- and the expression for r_t in terms of d_- we can simplify:

$$= d_+\frac{1}{1+r_t} - d_-\frac{1}{1-r_t} = \frac{1+r_t}{2}\frac{1}{1+r_t} - \frac{1-r_t}{2}\frac{1}{1-r_t} = 0 \tag{5}$$

Problem 3

5 points Observe the $\mathbf{M}$ defined below, with 8 training points and 8 weak classifiers. As defined in Problem 1, the i^{th} column of $\mathbf{M}$ represents weak classifier i applied to the training points.

$$\mathbf{M} = \begin{pmatrix} -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 \\ -1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 & 1 & -1 & 1 & 1 \\ 1 & -1 & 1 & 1 & -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 \\ 1 & 1 & -1 & 1 & 1 & 1 & 1 & -1 \\ 1 & 1 & -1 & 1 & 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 \end{pmatrix}$$

Assume that we start with the following initial distribution over the datapoints:

$$\mathbf{d}_1 = \left(\frac{3-\sqrt{5}}{8}, \frac{3-\sqrt{5}}{8}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{\sqrt{5}-1}{8}, \frac{\sqrt{5}-1}{8}, 0\right)^T$$

Perform Adaboost using the algorithm defined in Problem 2 using $\mathbf{M}$, $\mathbf{d}_1$, and $t_{max} = 7$. What weak classifier is picked at each round of boosting? Do you notice any pattern?

Solution: at $t = 1$ we have:

$$\mathbf{d}_1^T\mathbf{M} = \left(\frac{\sqrt{5}-1}{2}, 0, \frac{3-\sqrt{5}}{2}, \frac{3\sqrt{5}-1}{12}, \frac{3\sqrt{5}-1}{12}, \frac{3\sqrt{5}-1}{12}, \frac{1}{2}, \frac{11-3\sqrt{5}}{12}\right)$$

so we pick weak classifier 1. Now, the distribution at round two is:

$$\mathbf{d}_2 = \left(\frac{1}{4}, \frac{1}{4}, \frac{\sqrt{5}-1}{12}, \frac{\sqrt{5}-1}{12}, \frac{\sqrt{5}-1}{12}, \frac{3-\sqrt{5}}{8}, \frac{3-\sqrt{5}}{8}, 0\right)^T$$

and the edges at round 2 are:

$$\mathbf{d}_2^T\mathbf{M} = \left(0, \frac{3-\sqrt{5}}{2}, \frac{\sqrt{5}-1}{2}, \frac{4-\sqrt{5}}{6}, \frac{4-\sqrt{5}}{6}, \frac{4-\sqrt{5}}{6}, \frac{\sqrt{5}-1}{4}, \frac{5+\sqrt{5}}{12}\right)$$

so we pick weak classifier 3. Continuing this process, we then pick weak classifier 2 in round 3. However, now we observer that $\mathbf{d}_4 = \mathbf{d}_1$, hence we have found a cycle, in which we repeatedly select classifiers 1, 3, 2, 1, 3, 2, ...

5 points What is the norm-1 margin produced by Adaboost for this example?

Solution: $r_t = \frac{\sqrt{5}-1}{2}, t \in 1, 2, 3$. Thus, the coefficients used to combine classifiers in our example are: $[\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0, 0, 0, 0, 0]$ and the margin equals the minimum value in the following vector: $\mathbf{M} \times [\frac{1}{3}, \frac{1}{3}, \frac{1}{3}, 0, 0, 0, 0, 0]^T$, which is $\frac{1}{3}$.

10 points Instead of using Adaboost, imagine we combined our classifiers using the following coefficients: $[2, 3, 4, 1, 2, 2, 1, 1] \times \frac{1}{16}$. What is the margin in this case? Does Adaboost maximize the margin?

Solution: $\mathbf{M} \times [2, 3, 4, 1, 2, 2, 1, 1]^T \times \frac{1}{16} = \frac{3}{8}$ for all training points. This margin is greater than the one generated by Adaboost. Therefore Adaboost does NOT always maximize the norm-1 margin.