Optimization and Machine Learning, Spring 2020
Homework 4
(Due Tuesday, May 12 at 11:59pm (CST))

1. Given a training dataset $S = \{(x_i, y_i)\}_{i=1}^n$, in which $x_i \in \mathbb{R}^d$ and $y_i \in \{-1, 1\}$ denote the i-th sample and the i-th label, respectively. Suppose that we use S to train a machine learning model based on Adaboost. At the end of the t-th iteration ($t = 1, 2, ..., T$), the importance of the i-th ($i = 1, 2, ..., n$) sample x_i is reweighted as

$$D_i^{(t+1)} = D_i^{(t)} \exp\left(-\alpha_t y_i h_t(x_i)\right),$$

where α_t is the weight of the t-th weakly binary classifier h_t, i.e.,

$$\alpha_t = \frac{1}{2}\ln(\frac{1-\epsilon_t}{\epsilon_t}), \text{ with } \epsilon_t = \sum_{i=1}^{n} D_i^{(t)} \mathbb{1}(y_i \neq h_t(x_i)).$$

To classify an arbitrary test sample x, we calculate $f(x) = \sum_{t=1}^{T} \alpha_t h_t(x)$ and then return its sign. Now let's show that if every learner h_t ($\forall t$) achieves 51% classification accuracy (that is, only slightly better than random guessing), AdaBoost will converge to zero training error.

   (a) Let's change the update rule so that the weights of each iteration are normalized, that is, $\sum_{i=1}^{n} D_i^{(t)} = 1$ ($\forall t$). In this sense, we can treat the weights as a discrete probability distribution over the sample points. Hence we rewrite the update rule by

$$D_i^{(t+1)} = \frac{D_i^{(t)} \exp(-\alpha_t y_i h_t(x_i))}{Z_t},$$

   where Z_t is the normalization factor, $\forall t$. Please show that the following formula is satisfied,

$$Z_t = 2\sqrt{\epsilon_t(1-\epsilon_t)}.$$

   (5 points)

   (b) Assume that the initial weights follow uniform distribution, i.e.,

$$D_1^{(1)} = D_2^{(1)} = \cdots = D_n^{(1)} = \frac{1}{n}.$$

   Please show that

$$D_i^{(T+1)} = \frac{1}{n\prod_{t=1}^{T} Z_t} e^{-y_i f(x_i)},$$

   where $i = 1, 2, ..., n$ and $t = 2, 3, ..., T$. (5 points)

   (c) Let m be the number of sample points that Adaboost classifies incorrectly. Please show that

$$\sum_{i=1}^{n} e^{-y_i f(x_i)} \geq m.$$

   (5 points)

   (d) Based on the results in (a), (b), and (c), please show that once $\epsilon_t \leq 0.49$ is satisfied for every learner h_t ($\forall t$), then we have $m \to 0$ as $T \to \infty$. (5 points)

2. Suppose that we are interested in learning a classifier, such that at any turn of a game we can pose a question, like "should I attack this ant hill now?", and get an answer. That is, we want to build a classifier which we can feed some features on the current game state, and get the output "attack" or "don't attack". There are many possible ways to define what the action "attack" means, but for now let's define it as sending all friendly ants that can see the ant hill under consideration towards it.

Let's recall the AdaBoost algorithm described in class. Its input is a dataset $\{(x_i, y_i)\}_{i=1}^n$, with x_i being the i-th sample, and $y_i \in \{-1, 1\}$ denoting the i-th label, $i = 1, 2, ..., n$. The features might be composed of a count of the number of friendly ants that can see the ant hill under consideration, and a count of the number of enemy ants these friendly ants can see. For example, if there were 10 friendly ants that could see a particular ant hill, and 5 enemy ants that the friendly ants could see, we would have:

$$x_1 = \begin{bmatrix} 10 \\ 5 \end{bmatrix}.$$

The label of the example x_1 is $y_1 = 1$, once the friendly ants were successful in razing the enemy ant hill, and $y_1 = 0$ otherwise. We could generate such examples by running a greedy bot (or any other opponent bot) against a bot that we make periodically try to attack an enemy ant hill. Each time this bot tries the attack, we record (say, after 20 turns or some other significant amount of time) whether the attack was successful or not.

(a) Let ϵ_t denote the error of a weak classifier h_t:

$$\epsilon_t = \sum_{i=1}^{n} D_i^{(t)} \mathbb{1}(y_i \neq h_t(x_i)).$$

In the simple "attack" / "don't attack" scenario, suppose that we have implemented the following six weak classifiers:

$$h^{(1)}(x_i) = 2 * \mathbb{1}(x_{i1} \geq 2) - 1, \qquad h^{(4)}(x_i) = 2 * \mathbb{1}(x_{i2} \leq 2) - 1,$$
$$h^{(2)}(x_i) = 2 * \mathbb{1}(x_{i1} \geq 5) - 1, \qquad h^{(5)}(x_i) = 2 * \mathbb{1}(x_{i2} \leq 5) - 1,$$
$$h^{(3)}(x_i) = 2 * \mathbb{1}(x_{i1} \geq 10) - 1, \qquad h^{(6)}(x_i) = 2 * \mathbb{1}(x_{i2} \leq 10) - 1.$$

Given ten training data points ($n = 10$) as shown in Fig. 1, please show that what is the minimum value

i	x_{i1}	x_{i2}	y_i
1	3.5	0.5	1
2	2.5	1.5	1
3	5.5	5.5	1
4	7.5	7.5	1
5	6.5	9.5	1
6	1.5	2.5	-1
7	4.5	4.5	-1
8	3.5	7.5	-1
9	1.5	11.5	-1
10	6.5	11.5	-1

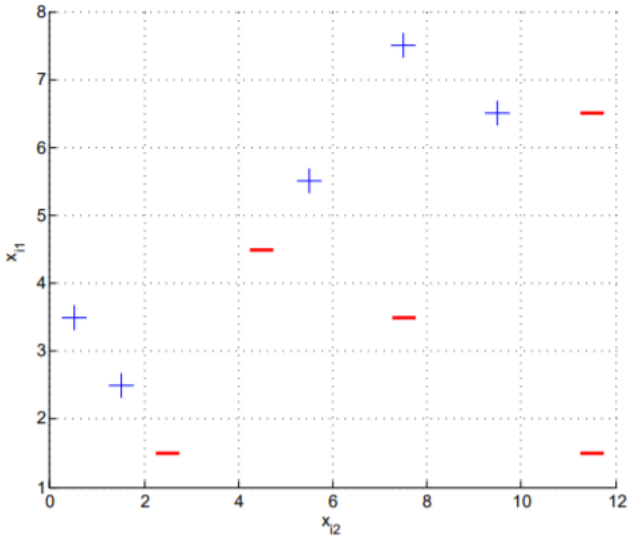

Figure 1: The training data in (a).

of ϵ_1 and which of $h^{(1)}, ..., h^{(6)}$ achieve this value? Note that there may be multiple classifiers that all have the same ϵ_1. You should list all classifiers that achieve the minimum ϵ_1 value. (5 points)

(b) For all the questions in the remainder of this section, let h_1 denote $h^{(1)}$ chosen in the first round of boosting. (That is, $h^{(1)}$ was the classifier that achieved the minimum ϵ_1.)

(1) What is the value of α_1 (the weight of this first classifier h_1)? Keep in mind that the log in the formula for α_t is a natural log (base e). (5 points)

(2) What should Z_t be in order to make sure the distribution $D^{(t+1)}$ is normalized correctly? That is, derive the formula of Z_t in terms of $D^{(t)}$, α_t, h_t, and $\{(x_i, y_i)\}_{i=1}^n$, that will ensure $\sum_{i=1}^n D_i^{(t+1)} = 1$. (5 points)

(3) Which points will increase in significance in the second round of boosting? That is, for which points will we have $D_i^{(1)} < D_i^{(2)}$? What are the values of $D^{(2)}$ for these points? (5 points)

(4) In the second round of boosting, the weights on the points will be different, and thus the error ϵ_2 will also be different. Which of $h^{(1)}, ..., h^{(6)}$ will minimize ϵ_2? (Which classifier will be selected as the second weak classifier h_2?) What is its value of ϵ_2? (5 points)

(5) What will the average error of the final classifier H be, if we stop after these two rounds of boosting? That is, if $H(x) = \text{sign}(\alpha_1 h_1(x) + \alpha_2 h_2(x))$, what will the training error $\epsilon = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}(y_i \neq h(x_i))$ be? Is this more, less, or the same as the error we would get, if we just used one of the weak classifiers instead of this final classifier H? (5 points)

3. Given valid kernels $k_1(\mathbf{x}, \mathbf{x}')$ and $k_2(\mathbf{x}, \mathbf{x}')$, please verify the following new kernels will also be valid:

   (a) $k(\mathbf{x}, \mathbf{x}') = f(\mathbf{x}) k_1(\mathbf{x}, \mathbf{x}') f(\mathbf{x}')$, where $f(\cdot)$ is any function. (2 points)

   (b) $k(\mathbf{x}, \mathbf{x}') = q(k_1(\mathbf{x}, \mathbf{x}'))$, where $q(\cdot)$ is a polynomial with nonnegative coefficients. (3 points)

   (c) $k(\mathbf{x}, \mathbf{x}') = \exp(k_1(\mathbf{x}, \mathbf{x}'))$. (5 points)

   (d) $k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^\top \mathbf{A} \mathbf{x}'$, where $\mathbf{A}$ is a symmetric positive semi-definite matrix. (5 points)

4. Consider the space of all possible subsets A of a given fixed set D. Show that the kernel function $k(A_1, A_2) = 2^{|A_1 \cap A_2|}$ corresponds to an inner product in a feature space of dimensionality $2^{|D|}$ defined by the mapping $\phi(A)$ where A is a subset of D and the element $\phi_U(A)$, indexed by the subset U, is given by

$$\phi_U(A) = \begin{cases} 1, & \text{if } U \subseteq A; \\ 0, & \text{otherwise.} \end{cases}$$

   Here $U \subseteq A$ denotes that U is either a subset of A of is equal to A. (10 points)

5. Suppose we have a data set of input vectors $\{\mathbf{x}_n\}$ with corresponding target values $t_n \in \{-1, 1\}$, and suppose that we model the density of input vectors within each class separately using a Parzen kernel density estimator which is defined as follows

$$p(\mathbf{x}|t) = \frac{1}{N_t} \sum_{n=1}^{N} \frac{1}{Z_k} k(\mathbf{x}, \mathbf{x}_n) \delta(t, t_n)$$

   where $k(\mathbf{x}, \mathbf{x}')$ is a valid kernel, Z_k is the normalization constant for the kernel and $\delta(t, t_n)$ equals 1 if $t = t_n$ and 0 otherwise.

   (a) Write down the minimum misclassification-rate decision rule assuming the two classes have equal prior probability. (3 points)

   (b) Show that, if the kernel is chosen to be $k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^\top \mathbf{x}'$, then the classification rule reduces to simply assigning a new input vector to the class having the closest mean. (4 points)

   (c) Show that, if the kernel takes the form $k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^\top \phi(\mathbf{x}')$, that the classification is based on the closest mean in the feature space $\phi(\mathbf{x})$. (4 points)

6. The problem of maximizing margin can be converted into an following equivalent problem

$$\begin{aligned} &\underset{\mathbf{w}, b}{\text{argmin}} \ \frac{1}{2}\|\mathbf{w}\|^2 \\ &\text{subject to } t_n(\mathbf{w}^\top \phi(\mathbf{x}_n) + b) \geq 1, \quad n = 1, \cdots, N. \end{aligned}$$

   where $\phi(\mathbf{x})$ is a fixed feature-space transformation.

   (a) By introducing Lagrange multipliers $\{a_n\}$, please give the Lagrangian function and the dual representation of the maximum margin problem. (8 points)

   (b) Please show that the value ρ of the margin for the maximum-margin hyperplane is given by

$$\frac{1}{\rho^2} = \sum_{n=1}^{N} a_n.$$

   (Hint: $\{a_n\}$ can be obtained by solving the dual representation of the maximum margin problem.) (6 points)