

# CSC 588: Homework 2

Chicheng Zhang

March 21, 2021

- This homework is due on Mar 23 on gradescope.
- This homework is intentionally made short (and due on a late date) to give you more time to think about your project, which is due on Mar 16.
- If you feel unable to make progress on any of the questions, feel free to post your questions on Piazza.
- You are encouraged to discuss the homework questions with your classmates, but the discussions should only be at a high level, and you should write your solutions in your own words. For every question you have had discussions on, please mention explicitly whom you have discussed with; otherwise it may be counted as academic integrity violation.
- Feel free to use existing theorems from the course notes / the textbook.

## Problem 1

In this exercise, we conduct experiments on AdaBoost using a simple benchmark dataset `diabetes` in `openml.org`. You may use any programming languages you like. Please submit your source code by emailing to `chichengz@cs.arizona.edu`. Some preparations:

1. Go to <https://www.openml.org/d/37> and download the dataset.
2. The last column of the dataset gives the classes of the examples - use `+1` to denote class 'tested\_positive' and `-1` to denote class 'tested\_negative'.
3. Choose a random subset of size 100 as the training set, and use the remaining 668 examples as the test set.

Answer the following questions:

1. Define base hypothesis class  $\mathcal{B} = \{\sigma \cdot (2I(x_i \leq t) - 1) : \sigma \in \{\pm 1\}, i \in \{1, \dots, d\}, t \in \mathbb{R}\}$  as the set of bi-directional decision stumps. Let the weak learner  $\mathcal{A}$  be: given a weighted dataset, return the classifier  $h \in \mathcal{B}$  that has the smallest weighted error. Implement AdaBoost with  $\mathcal{A}$ , and run it for 3000 iterations. At time  $t$ , suppose the following cumulative voting classifier

$$H_t(x) = \text{sign}(f_t(x)), \quad f_t(x) = \sum_{s=1}^t \alpha_s h_s(x)$$

is produced. Plot AdaBoost's learning curves: the training error of  $H_t$ , the test error of  $H_t$ , and the training exponential loss of  $f_t$ , as functions of iteration  $t$ . What do you see?

2. Given voting classifier  $f_t$ , define its normalization as

$$\bar{f}_t(x) = \frac{f_t(x)}{\sum_{s=1}^t \alpha_s} = \frac{\sum_{s=1}^t \alpha_s h_s(x)}{\sum_{s=1}^t \alpha_s}$$

Now, given an example  $(x, y)$ , define its normalized margin at time step  $t$  as  $y \bar{f}_t(x)$ . At iterations 3, 10, 30, 100, 300, 1000, 3000, plot histograms of normalized margins of training examples. Do you see any trend as  $t$  increases?

## Problem 2

Show that for AdaBoost, at iteration  $t$ , the updated distribution  $D_{t+1}$  satisfies that

$$\sum_{i=1}^m D_{t+1}(i) I(h_t(x_i) \neq y_i) = \frac{1}{2}.$$

Intuitively, why is this formula reasonable?

## Problem 3

Most of the problems we have seen in class so far are about classification. Consider instead a regression problem, where we have a distribution over  $\mathcal{X} \times \mathcal{Y}$ , where the feature space  $\mathcal{X} = \{x \in \mathbb{R}^d : \|x\|_\infty \leq R\}$  and the label space  $\mathcal{Y} = [-Y, Y]$ . Consider the hypothesis class  $\mathcal{H} = \{h_w(x) := \langle w, x \rangle : \|w\|_1 \leq B\}$ , and define the loss function to be the square loss  $\ell_{\text{sq}}(\hat{y}, y) = (\hat{y} - y)^2$ . For any predictor  $h : \mathbb{R}^d \rightarrow \mathbb{R}$ , define  $L_D(h) = \mathbb{E}_{(x,y) \sim D} \ell_{\text{sq}}(h(x), y)$  its generalization loss. Now, given a set of examples  $S = ((x_1, y_1), \dots, (x_m, y_m))$  drawn iid from  $D$ , define the ERM  $\hat{h} = \arg \min_{h \in \mathcal{H}} \mathbb{E}_S \ell_{\text{sq}}(h(x), y)$ . For any  $\delta > 0$ , can you show a tight upper bound on

$$L_D(\hat{h}) - \min_{h' \in \mathcal{H}} L_D(h')$$

that holds with probability  $1 - \delta$ ? (You might want to use the contraction inequality of Rademacher complexity to solve this problem.)

## Problem 4

Consider a set of examples  $S = (x_1, \dots, x_m) \subset \mathbb{R}^d$ , where for each  $i$ ,  $\|x_i\|_\infty \leq X_\infty$ . Define the class of  $\ell_1$ -regularized  $n$ -layer ReLU network as

$$\mathcal{F}_n = \left\{ h_{W_1, \dots, W_n} : \forall i, W_i \in \mathbb{R}_+^{N_i \times N_{i-1}}, \forall j, \|W_i^j\|_1 \leq B_i \right\},$$

where  $N_0, \dots, N_n$  are fixed numbers such that  $N_0 = d$ ,  $N_n = 1$ ,  $W_i^j$  denotes the  $j$ -th row of  $W_i$ , and

$$h_{W_1, \dots, W_n}(x) = \sigma(W_n \sigma(W_{n-1} \cdots \sigma(W_2 \sigma(W_1 x))));$$

here  $\sigma(z) = \max(z, 0)$  is the ReLU activation function, and when  $v = (v_1, \dots, v_l)$  is a vector, we denote by  $\sigma(v) = (\max(v_1, 0), \dots, \max(v_l, 0))$  the result of element-wise application of  $\sigma$  on  $v$ . Can you use the contraction inequality of Rademacher complexity to give a tight bound on  $\text{Rad}_S(\mathcal{F}_n)$ ? How would your bound change if  $\sigma$  is instead the sigmoid activation function  $\sigma(z) = \frac{1}{1+e^{-z}}$ ? (Perhaps start with something easier, say  $n = 1$  or  $2$ , then try to generalize.)

## Problem 5

How much time did it take you to complete this homework?