CS 260: Machine Learning **Spring 2020**

Homework 3

Hand Out: April.29 *Due: May.18*

1. **Exercise 9.1**

   Define a vector of auxiliary variables $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_m]^\top \in \mathbb{R}^m$. According to the hint, minimizing the empirical risk is equivalent to minimizing the following optimization problem:

$$\min_{\mathbf{w}} \sum_{i=1}^{m} \lambda_i$$
$$\mathbf{w}^\top \mathbf{x}_i - \lambda_i \leq y_i,$$
$$-\mathbf{w}^\top \mathbf{x}_i - \lambda_i \leq -y_i,$$

   then we can define the matrix $\mathbf{A}$ as follows,

$$\mathbf{A} = \begin{bmatrix} \mathbf{X} & -\mathbf{I}_m \\ -\mathbf{X} & \mathbf{I}_m \end{bmatrix},$$

   where $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m]^\top \in \mathbb{R}^{m \times d}$, and we further define

$$\mathbf{v} = [w_1, \cdots, w_d, \lambda_1, \cdots, \lambda_m]^\top = [\mathbf{w}^\top, \boldsymbol{\lambda}^\top]^\top \in \mathbb{R}^{d+m}.$$

   Then let $\mathbf{b} = [y_1, \cdots, y_m, -y_1, \cdots, -y_m]^\top \in \mathbb{R}^{2m}$, $= [\mathbf{0}_d, \mathbf{1}_m]$. Finally we are able to transform the original problem into the following problem

$$\min \quad {}^\top \mathbf{v}$$
$$\mathbf{A}\mathbf{v} \leq \mathbf{b}$$

   ■

2. **Exercise 9.3**

   Based on the hint, let $d = m$, for each $i \in [m]$, define $\mathbf{x}_i = \mathbf{e}_i \in \mathbb{R}^d$.
   Then we define sign function on 0 as $\text{sign}(0) = -1$. For $i \in [d]$, let the label of $\mathbf{x}_i$ be 1, i.e., $y_i = 1$. Denote by $\mathbf{w}^{(t)} \in \mathbb{R}^d$ the weight vector which is maintained by the Perceptron. Then we can derive that

$$\mathbf{w}^{(i)} = \sum_{j<i} \mathbf{e}_j, \quad i \in [d]$$

   We can see that for each $i \in [d]$,

$$\langle \mathbf{w}^{(i)}, \mathbf{x}_i \rangle = 0.$$

   We also derive that the vector $\mathbf{w}^\star = [1, ..., 1]^\top$ satisfies the requirements listed in the question. ■

3. **Exercise 9.5** Since for each $\mathbf{w} \in \mathbb{R}^d$, $x \in \mathbb{R}^d$, we have

$$\mathrm{sign}(\langle \mathbf{w}, \mathbf{x} \rangle) = \mathrm{sign}(\langle \eta\mathbf{w}, \mathbf{x} \rangle),$$

for any $\eta > 0$. Then we can get that the modified Perceptron and the Perceptron produce the same predictions. Therefore, both algorithms perform the same number of iterations. ■

4. **Exercise 18.1**

(1). Given h, construct a full binary tree, where the root note is $(x_1 = 0?)$, and all the nodes at depth i are of the form $(x_{i+1} = 0?)$. This tree has 2^d leaves, and the path from each root to the leaf is composed of the nodes $(x_1 = 0?)$, $(x_2 = 0?)$, $\cdots$, $(x_d = 0?)$. We can see that we can allocate one leaf to any possible combination of values for $\mathbf{x} = [x_1, x_2, \cdots, x_d]^\top \in \mathbb{R}^d$.
(2). Based on the previous result, we can derive that we can shatter the domain $\{0, 1\}^d$. Thus, the VC dimension is 2^d. ■

5. **Exercise 18.2**

(1). Here we use H to denote the binary entropy. Then the algorithm first picks the root node through searching for the feature that maximizes the information gain. We can get the information gain for x_1 (namely, if we choose $x_1 = 0?$ as the root) is

$$H(1/2) - (3/4H(2/3) + 1/4H(0)) \approx 0.22,$$

and the information gain for x_2 as weel as x_3, is

$$H(1/2) - (1/2H(1/2) + 1/2H(1/2)) = 0,$$

So the algorithm will pick $x_1 = 0?$ as the root. As a result, the three examples $((1,1,0),0)$, $((1,1,1),1)$, and $((1,0,0),1)$ will go to one subtree. Therefore, no matter what question will be asked later, we are not able to classify all three examples perfectly. We can find that at least one example will be mislabeled, which indicates that the training error is at least 1/4.

(2). One possible such decision tree is

6. **Exercise 19.1**

We follow the hints for proving that for any $k \geq 2$, we have

$$\mathbb{E}_{S \sim \mathcal{D}^m}\left[\sum_{i:|C_i \cap S| < k} P[C_i]\right] \leq \frac{2rk}{m}$$

The claim in the first hint follows easily from the linearity of the expectation and the fact that $\sum_{i:|C_i \cap S| \leq k} \mathbb{P}[C_i] = \sum_{i=1}^r 1_{[|C_i| \leq k]} \mathbb{P}[C_i]$. The claim in the second hint follows directly from Chernoff's bounds. The next hint leaves nothing to prove. Finally, combining the fourth hint with the previous hints,

$$\mathbb{E}_{S\sim\mathcal{D}^m}\left[\sum_{i:|C_i\cap S|<k}\mathbb{P}[C_i]\right]\leq\sum_{i=1}^{r}\max\{8/(me),2k/m\}$$

Since $k \geq 2$, our proof is completed.

7. (1) The testing error obtained by Perceptron model implemented in scikit-learn is 0.

   (2) No solution required.

   (3) The testing error obtained by my perceptron model is 0. It tooks 75 iterations ($\leq$ 2 epochs) to converge. (The number of iteration may vary from different implementation.)

8. (1) The mean-spuared testing error obtained by LinearRegression model implemented in scikit-learn is 27.18.

   (2) No solution required.

   (3) The mean-spuared testing error obtained by my linear regression model is 27.18.