

Assignment I

Big Data Analytics 2020-2021 - UMONS
Souhaib Ben Taieb

1

Solve Problem 1.5 in LFD..

Problem 1.5 The perceptron learning algorithm works like this: In each iteration t , pick a random $(\mathbf{x}(t), y(t))$ and compute the 'signal' $s(t) = \mathbf{w}^T(t)\mathbf{x}(t)$. If $y(t) \cdot s(t) \leq 0$, update \mathbf{w} by

$$\mathbf{w}(t+1) \leftarrow \mathbf{w}(t) + y(t) \cdot \mathbf{x}(t) ;$$

One may argue that this algorithm does not take the 'closeness' between $s(t)$ and $y(t)$ into consideration. Let's look at another perceptron learning algorithm: In each iteration, pick a random $(\mathbf{x}(t), y(t))$ and compute $s(t)$. If $y(t) \cdot s(t) \leq 1$, update \mathbf{w} by

$$\mathbf{w}(t+1) \leftarrow \mathbf{w}(t) + \eta \cdot (y(t) - s(t)) \cdot \mathbf{x}(t) ,$$

where η is a constant. That is, if $s(t)$ agrees with $y(t)$ well (their product is > 1), the algorithm does nothing. On the other hand, if $s(t)$ is further from $y(t)$, the algorithm changes $\mathbf{w}(t)$ more. In this problem, you are asked to implement this algorithm and study its performance.

- (a) Generate a training data set of size 100 similar to that used in Exercise 1.4. Generate a test data set of size 10,000 from the same process. To get g , run the algorithm above with $\eta = 100$ on the training data set, until a maximum of 1,000 updates has been reached. Plot the training data set, the target function f , and the final hypothesis g on the same figure. Report the error on the test set.
- (b) Use the data set in (a) and redo everything with $\eta = 1$.
- (c) Use the data set in (a) and redo everything with $\eta = 0.01$.
- (d) Use the data set in (a) and redo everything with $\eta = 0.0001$.
- (e) Compare the results that you get from (a) to (d).

The algorithm above is a variant of the so called Adaline (*Adaptive Linear Neuron*) algorithm for perceptron learning.

Figure 1: Source: Abu-Mostafa et al. Learning from data. AMLbook.

Solve Problem 1.7 in LFD..

Problem 1.7 A sample of heads and tails is created by tossing a coin a number of times independently. Assume we have a number of coins that generate different samples independently. For a given coin, let the probability of heads (probability of error) be μ . The probability of obtaining k heads in N tosses of this coin is given by the binomial distribution:

$$P[k | N, \mu] = \binom{N}{k} \mu^k (1 - \mu)^{N-k}.$$

Remember that the training error ν is $\frac{k}{N}$.

- (a) Assume the sample size (N) is 10. If all the coins have $\mu = 0.05$ compute the probability that at least one coin will have $\nu = 0$ for the case of 1 coin, 1,000 coins, 1,000,000 coins. Repeat for $\mu = 0.8$.
- (b) For the case $N = 6$ and 2 coins with $\mu = 0.5$ for both coins, plot the probability

$$P[\max_i |\nu_i - \mu_i| > \epsilon]$$

for ϵ in the range $[0, 1]$ (the max is over coins). On the same plot show the bound that would be obtained using the Hoeffding Inequality. Remember that for a single coin, the Hoeffding bound is

$$P[|\nu - \mu| > \epsilon] \leq 2e^{-2N\epsilon^2}.$$

[Hint: Use $P[A \text{ or } B] = P[A] + P[B]$ $P[A \text{ and } B] = P[A] + P[B] - P[A]P[B]$, where the last equality follows by independence, to evaluate $P[\max \dots]$]

Figure 2: Source: Abu-Mostafa et al. Learning from data. AMLbook.

3

Solve Problem 2.6 in LFD.

Problem 2.6 Prove that for $N \geq d$,

$$\sum_{i=0}^d \binom{N}{i} \leq \left(\frac{eN}{d} \right)^d.$$

We suggest you first show the following intermediate steps.

$$(a) \quad \sum_{i=0}^d \binom{N}{i} \leq \sum_{i=0}^d \binom{N}{i} \left(\frac{N}{d} \right)^{d-i} \leq \left(\frac{N}{d} \right)^d \sum_{i=0}^N \binom{N}{i} \left(\frac{d}{N} \right)^i.$$

$$(b) \quad \sum_{i=0}^N \binom{N}{i} \left(\frac{d}{N} \right)^i \leq e^d. \text{ [Hints: Binomial theorem; } (1 + \frac{1}{x})^x \leq e \text{ for } x > 0.]$$

Hence, argue that $m_{\mathcal{H}}(N) \leq \left(\frac{eN}{d_{VC}} \right)^{d_{VC}}$.

Figure 3: Source: Abu-Mostafa et al. Learning from data. AMLbook.

4

Solve Problem 2.18 in LFD.

Problem 2.18 The VC dimension of the perceptron hypothesis set corresponds to the number of parameters (w_0, w_1, \dots, w_d) of the set, and this observation is ‘usually’ true for other hypothesis sets. However, we will present a counter example here. Prove that the following hypothesis set for $x \in \mathbb{R}$ has an infinite VC dimension:

$$\mathcal{H} = \left\{ h_\alpha \mid h_\alpha(x) = (-1)^{\lfloor \alpha x \rfloor}, \text{ where } \alpha \in \mathbb{R} \right\},$$

where $\lfloor A \rfloor$ is the biggest integer $\leq A$ (the floor function). This hypothesis has only one parameter α but ‘enjoys’ an infinite VC dimension. *[Hint: Consider x_1, \dots, x_N , where $x_n = 10^n$, and show how to implement an arbitrary dichotomy y_1, \dots, y_N .]*

Figure 4: Source: Abu-Mostafa et al. Learning from data. AMLbook.

TURN IN

- A pdf file with your solutions.
- Your (complete) source code to (easily) reproduce the pdf file.
- **DUE:** March 28, 11:55pm (late submissions not allowed), loaded into Moodle.