Machine Learning Assignment 1

B08611035

September 2020

1 The Learning Problem

(1) **d.**

taking the mango images as the input x and the quality score as predicted output y. Then once a mango image input to the model, it could be classification in to a quality level or just a score (with regression).

(2) **a,b,c.**

[a]: the probability of coin flapping is not relate to spam email.

[b]: no training during the process, and it's more like a domain knowledge of human to label the email as spam email or not.

[c]: no training during the classification process, and not verify the spam email in more deeper than facial features.

2 Perceptron Learning Algorithm

(3) **d. unchanged**

define that:

$$R^2 = \max_n ||x_n||^2, \quad \rho = \min_n y_n \frac{\mathbf{w}_f^\top}{||\mathbf{w}_f||} x_n, \quad \frac{\mathbf{w}_f}{||\mathbf{w}_f||} \cdot \frac{\mathbf{w}_T}{||\mathbf{w}_T||} \geq \sqrt{T} \cdot \frac{\rho}{R}$$

$$T \leq \frac{\rho^2}{R^2} = \frac{\min_n |\mathbf{w}_f \cdot x_n|^2}{\max_n ||x_n||^2 \cdot ||\mathbf{w}_f||^2}$$

as the equation defined above, we can know that the upper bound of time-spent (T) is defined by the ratio ρ^2/R^2, where the ratio is determined by the maximum length of x_n. Thus, the scale down on value of all x_n (include x_0 within) does not have influential impact on time cost.

(4) **c. 2**

The original definition is like:

$$T \leq \frac{\rho^2}{R^2} = \frac{\min_n |w_f \cdot x_n|^2}{\max_n ||x_n||^2 \cdot ||w_f||^2}$$

and w_t changed under mistake:

$$||w_{t+1}||^2 = ||w_t + y_{n(t)}\frac{x_{n(t)}}{||x_{n(t)||}}||^2 \leq ||w_t||^2 + ||y_{n(t)}\frac{x_{n(t)}}{||x_{n(t)||}}||^2$$
$$\leq ||w_t||^2 + \frac{1}{16}\max_n ||x_n||^2 = ||w_t||^2 + \frac{1}{16}R^2$$
$$\frac{w_f^T w_T}{||w_f||} \geq \frac{1}{4}T\rho, \quad ||w_T|| \leq \frac{\sqrt{T}}{4}R$$

Thus,

$$\frac{\frac{1}{4}T\rho}{\frac{1}{4}\sqrt{T}R} \leq \frac{w_f^T w_T}{||w_f^T||||w_T||} \leq 1, \quad T \leq \hat{\rho^{-2}}$$

(5) **d.**

$$y_{n(t)}w_t^T x_{n(t)} \leq 0$$
$$w_{t+1} = w_t + A_{n(t)}y_{n(t)}x_{n(t)}$$
so it's going to find a $A_{n(t)}$ so that $y_{n(t)}w_t^T x_{n(t)} > 0$
$$y_{n(t)}w_{t+1}^T = y_{n(t)}w_t + A_{n(t)}y_{n(t)}y_{n(t)}x_{n(t)} = y_{n(t)}w_t + A_{n(t)}x_{n(t)}$$
$$y_{n(t)}w_{t+1}^T x_{n(t)} = y_{n(t)}w_t x_{n(t)} + A_{n(t)}x_{n(t)}x_{n(t)} > 0$$
$$A_{n(t)} > \frac{-y_{n(t)}w_{t+1}^T x_{n(t)}}{||x_{n(t)}||^2}$$
Thus, here we take $A_{n(t)}$ as:
$$A_{n(t)} = \Big\lfloor \frac{-y_{n(t)}w_{t+1}^T x_{n(t)}}{||x_{n(t)}||^2} + 1 \Big\rfloor$$
as the update could be:
$$w_{t+1} \leftarrow w_t + y_{n(t)}x_{n(t)} \cdot \Big\lfloor \frac{-y_{n(t)}w_{t+1}^T x_{n(t)}}{||x_{n(t)}||^2} + 1 \Big\rfloor$$

(6) **c. 3**

3 Types of Learning

(7) **e. reinforcement learning**
Just as the question mentions: learning to play the game by practicing with itself and getting the feedback from the "judge" environment. Here the machine is learning from the feed back from environment with its action. Thus, it's kind of reinforcement learning.

(8) **b.**
First, the learning problem is more like classification without explicit class definition, so it could be structure learning. Also, we just input whole dataset with some labeled and others unlabeled, so it could be batch and semi-supervised learning.

4 Off-Training-Set Error

(9) **e. (0, 1)**

The small $E_{ost}(g)$ is 0 because the dataset is linear separable, like the line $g = f(y) = 2.5$ can simply divide the dataset. And we can also assume there are a hypothesis that take six data as $+1$, if the in-sample data is the three with $+1$, the $E_{in}(g) = 0$, however, for the out-training set:

$$E_{ost}(g) = \frac{1}{|\mathcal{U} \backslash \mathcal{D}|} \sum_{(x,y) \in \mathcal{U} \backslash \mathcal{D}} [[g(x) \neq f]] = \frac{1}{3} \cdot 3 = 1$$

5 Hoeffding Inequality

(10) **b.**

Hoeffding's inequality is defined as:

$$\mathbf{P}[|\nu - \mu| > \epsilon] \leq 2 \exp(-2\epsilon^2 N)$$

Here, $1 - \delta$ is defined as the probability of find out the more probable side, thus we have δ probability with fail to discern (as the probability to think it's similar). Thus the Hoeffding's inequality can be rewrote as:

$$\mathbf{P}[|\nu - \mu| > \epsilon] \leq \delta = 2 \exp(-2\epsilon^2 N)$$

and we can just simplify the equation here, and where N is the times we toss the coin.

$$\delta = 2 \exp(-2\epsilon^2 N) \Rightarrow \log \frac{\delta}{2} = -2\epsilon^2 N$$

$$N = \frac{\log \frac{\delta}{2}}{-2\epsilon^2} = \frac{\log \frac{2}{\delta}}{2\epsilon^2}$$

6 Bad Data

(11) **c.** 1/32

The question asking the probability of 5 examples $(x_n, f(x_n))$ such that $E_{in}(h_2) = 0$. Thus, we can take it as the conditional probability that in the condition that $E_{in}(h_2)$ what is the probability of correct $f(x_n)$ and assuming that in 5 independent times:

$$\mathbf{P}[f(x_n)|E_{in}(h_2)] = \frac{\mathbf{P}[f(x_n) \cdot E_{in}(h_2)]}{\mathbf{P}[E_{in}(h_2)]} = \frac{1/4}{1/2} = \frac{1}{2}$$

$$\mathbf{P}[f(x_n)|E_{in}(h_2)]^5 = (\frac{1}{2})^5 = \frac{1}{32}$$

(12) **d.** 3843/32768

A target function $f(x) = \text{sign}(x_1)$, a hypothesis $h1(x) = \text{sign}(2x_1 - x_2)$ and another is $h_2(x) = \text{sign}(x_2)$. The results could be in the three case:

$$\mathbf{P}[E_{in}(h_2) = E_{in}(h_1)]$$

$$= (2 \times \frac{1}{4} \times \frac{3}{4})^5 + (2 \times \frac{1}{4} \times \frac{3}{4})^3 \times (2 \times \frac{1}{4}) \times (2 \times \frac{1}{4} \times \frac{1}{4})\frac{5!}{3!1!1!}$$
$$+(2 \times \frac{1}{4} \times \frac{3}{4}) \times (2 \times \frac{1}{4})^2 \times (2 \times \frac{1}{4} \times \frac{1}{4})^2 \frac{5!}{1!2!2!} = \frac{3843}{32768}$$

(13) **b. d**

$$\mathbf{P}[\text{Bad } \mathcal{D} \text{ for } \mathcal{H}] \le C \cdot 2\exp(-2\epsilon^2 N)$$

from the uniform bound, we can know that:

$$\mathbf{P}[\text{Bad } \mathcal{D} \text{ for } \mathcal{H}] = \mathbf{P}[\text{Bad } \mathcal{D} \text{ for } h_1 \text{ or Bad } \mathcal{D} \text{ for } h_2 \text{ or } ...]$$
$$\le \mathbf{P}[\text{Bad } \mathcal{D} \text{ for } h_1] + \mathbf{P}[\text{Bad } \mathcal{D} \text{ for } h_2]...$$

and from the hypotheses of the question, we can find that for the $h_1(x) = \text{sign}(x_1)$ and $h_{d+1}(x_i) = -\text{sign}(x_1)$ they both relate to x_1 mean influenced by same bad data, and so on for $i = 1, ..., d$ and $i = d+1, ..., 2d$. Thus, $c = d$

7 Multiple-Bin Sampling

(14) **c. five green 4's**

The probability of each draw to get green 3 is $\frac{1}{12}$, so we just find out the options with 1/12 probability in each draw, and the answer is green 4.

(15) **c.** 274/1024

Four dices for 5 time with replacement, and the combination that get some of number that is purely green.

$$\frac{\binom{3}{1}^2 + \binom{2}{1}^5 - 1}{\binom{4}{1}^5} = \frac{274}{1024}$$

8 Programming Part*

(16) **b. 11**

(17) **b. -7**

(18) **c. 15**

(19) **d. 17**

(20) **d. 17**