Machine learning, 2024 Spring

Homework 1

Name: Zhou Shouchen
Student ID: 2021533042

Due 23:59 (CST), Mar. 15, 2024

1 Exercise 1.8

Let X_i be the result for the i-th sample, and $X_i = 1$ be the i-th sample is a red marble, $X_i = 0$ be the i-th sample is a green marble.

Since the bin of marbles has that $\mu = 0.9$ for the probability of red marbles, so $X_i \overset{i.i.d}{\sim} \text{Bern}(0.9)$, $i = 1, 2, \cdots, 10$.

So $P(X_i = 1) = 0.9$, $P(X_i = 0) = 0.1$, $i = 1, 2, \cdots, 10$.

Since there are totally 10 independent samples, let $X = \sum_{i=1}^{10} X_i$, then $X \sim Bin(10, 0.9)$, so

$$P(X = k) = C_{10}^{k} \cdot (0.9)^k \cdot (0.1)^{10-k}, \text{ where } k = 0, 1, 2, \cdots, 10$$

Since $\nu = \dfrac{X}{10}$, so

$$\begin{aligned} P(\nu \leq 0.1) &= P(\frac{X}{10} \leq 0.1) \\ &= P(X \leq 1) \\ &= P(X = 0) + P(X = 1) \\ &= 0.1^{10} + 10 \times 0.9 \times 0.1^9 \\ &= 9.1 \times 10^{-9} \end{aligned}$$

So above all, the probability that a sample of 10 marbles will have $\nu \leq 0.1$ is 9.1×10^{-9}.

2 Exercise 1.9

Since the settings are the same with Exercise 1.8, so we have $X_i \overset{i.i.d}{\sim} \text{Bern}(0.9)$, $i = 1, 2, \cdots, 10$. And according to the Hoeffding's inequality, we have

$$P(|\nu - \mu| \geq \epsilon) \leq 2e^{-2N\epsilon^2}$$

Since $\mu = 0.9$, and there are total $N = 10$ samples, so we have

$$\begin{aligned} P(\nu \leq 0.1) &= P(\nu - \mu \leq -0.8) \\ &\leq P(|\nu - \mu| \geq 0.8) \\ &\leq 2e^{-2 \times 10 \times 0.8^2} \\ &= 2e^{-12.8} \\ &\approx 5.5215 \times 10^{-6} \\ &< 9.1 \times 10^{-9} \end{aligned}$$

So above all, the upper bound of the probability that a sample of 10 marbles will have $\nu \leq 0.1$ is 5.5215×10^{-6}, which is much bigger than the exact value 9.1×10^{-9}.

Comparing to the exact result in Exercise 1.8, we could find that the bound is not tight, this is because the Hoeffding's inequality is a general bound for all the probability, and it is not tight for this specific case.

3 Problem 1.10

(a) Since $f(x) = +1$, so only the odd number k could make the $\boldsymbol{x}_k$ be predicted as $+1$ to be the correct predict. So for even k, we have $h(\boldsymbol{x}_k) \neq f(\boldsymbol{x}_k)$.
So the number of the $h(\boldsymbol{x}_k) \neq f(\boldsymbol{x}_k)$ is the number of even integers in the range $[N+1, N+M]$, which is equal to the even number in range $[1, N+M]$ minus the even number in range $[1, N]$.
So the number of the $h(\boldsymbol{x}_k) \neq f(\boldsymbol{x}_k)$ is $\left\lfloor \frac{N+M}{2} \right\rfloor - \left\lfloor \frac{N}{2} \right\rfloor$, where $\lfloor x \rfloor$ is the largest integer not greater than x.
So

$$\begin{aligned} E_{\text{off}}(h, f) &= \frac{1}{M} \sum_{m=1}^{M} [\![h(\boldsymbol{x}_{N+m}) \neq f(\boldsymbol{x}_{N+m})]\!] \\ &= \frac{1}{M} (\#\text{ even integers in range } [N+1, N+M]) \\ &= \frac{1}{M} \left\lfloor \frac{N+M}{2} \right\rfloor - \frac{1}{M} \left\lfloor \frac{N}{2} \right\rfloor \end{aligned}$$

So above all, $E_{\text{off}}(h, f) = \frac{1}{M} \left\lfloor \frac{N+M}{2} \right\rfloor - \frac{1}{M} \left\lfloor \frac{N}{2} \right\rfloor$.

(b) Since there is no training error on the training data, so all possibly f represent to a different values for $\boldsymbol{x}_{N+1}, \boldsymbol{x}_{N+2}, \cdots, \boldsymbol{x}_{N+M}$.
Since each $\boldsymbol{x}_k$ has $\{+1, -1\}$ two selections, so there are totally 2^M different possible f for the test data.

So above all, there are 2^M possible f can generate $\mathcal{D}$ in a noiseless setting.

(c) Similarly to the settings of (b), since we have $E_{\text{off}}(h, f) = \frac{1}{M} \sum_{m=1}^{M} [\![h(\boldsymbol{x}_{N+m}) \neq f(\boldsymbol{x}_{N+m})]\!] = \frac{k}{M}$, so we can get that for the total M testing data $\boldsymbol{x}_{N+1}, \boldsymbol{x}_{N+2}, \cdots, \boldsymbol{x}_{N+M}$, there are exactly k of them are predicted wrong.
So the total number of the possible f is the number of the ways to choose k data from M total testing data, which is C_M^k.

So above all, the number of the possible f can make $E_{\text{off}} = \frac{k}{M}$ is C_M^k.

(d) From (b), we have known that there are total 2^M possible f can generate $\mathcal{D}$ in a noiseless setting. And from (c), we have known that there are C_M^k possible f can make $E_{\text{off}} = \frac{k}{M}$.
Combine them, we can get that $P(E_{\text{off}}(h, f) = \frac{k}{M}) = \frac{C_M^k}{2^M}$.

So from the definition, we can get that

$$\begin{aligned} \mathbb{E}_f[E_{\text{off}}(h, f)] &= \sum_{k=0}^{M} P(E_{\text{off}}(h, f) = \frac{k}{M}) \cdot (\frac{k}{M}) \\ &= \sum_{k=0}^{M} \cdot (\frac{C_M^k}{2^M} \frac{k}{M}) \\ &= \frac{1}{M \cdot 2^M} \sum_{k=0}^{M} k \cdot C_M^k \end{aligned}$$

since

$$k \cdot C_M^k = \frac{k \cdot M!}{(M-k)!k!} = \frac{M!}{(M-k)!(k-1)!} = \frac{M \cdot (M-1)}{[(M-1)-(k-1)]!(M-k)!} = M \cdot C_{M-1}^{k-1}$$

So

$$\begin{aligned}
\mathbb{E}_f[E_{\text{off}}(h,f)] &= \frac{1}{M \cdot 2^M} \sum_{k=0}^{M} k \cdot C_M^k \\
&= \frac{1}{M \cdot 2^M} \sum_{k=1}^{M} k \cdot C_M^k \qquad \text{(Since } k = 0 \text{ has no contribution to the sum)} \\
&= \frac{1}{M \cdot 2^M} \sum_{k=1}^{M} M \cdot C_{M-1}^{k-1} \\
&= \frac{1}{2^M} \sum_{k=1}^{M} C_{M-1}^{k-1} \\
&= \frac{1}{2^M} \cdot 2^{M-1} \\
&= \frac{1}{2}
\end{aligned}$$

So above all, $\mathbb{E}_f(E_{\text{off}}(h,f)) = \frac{1}{2}$.

(e) From deduction process during (d), we have known that the result of $\mathbb{E}_f[E_{\text{off}}(h,f)]$ only depends on the number of the testing data M, and has nothing to do with the hypothesis h. This is because although the number of difference between h and f in the testing data is different, it will be the same after taking the expectation. So for deterministic algorithms A_1 and A_2, the expectations are the same.

So for the more general settings, the number of testing data M is fixed, so the expected off-training-set error is always the same.

So above all, $\mathbb{E}_f[E_{\text{off}}(A_1(D),f)] = \mathbb{E}_f[E_{\text{off}}(A_2(D),f)]$ always holds for more general settings.

4 Problem 1.12

(a) $E_{\text{in}}(h) = \sum_{n=1}^{N} (h - y_n)^2$

$\frac{\partial E_{\text{in}}(h)}{\partial h} = \sum_{n=1}^{N} 2(h - y_n) = 2(Nh - \sum_{n=1}^{N} y_n)$

$\frac{\partial^2 E_{\text{in}}(h)}{\partial h^2} = 2N > 0$

So the function $E_{\text{in}}(h)$ is convex.

So to get the minimum value of $E_{\text{in}}(h)$, we need to make $\frac{\partial E_{\text{in}}(h)}{\partial h} = 0$.

i.e. $2(Nh - \sum_{n=1}^{N} y_n) = 0 \Rightarrow h = \frac{1}{N} \sum_{n=1}^{N} y_n$.

Which is same as h_{mean}.

So above all, we have the estimition to make $E_{\text{in}}(h) = \sum_{n=1}^{N} (h - y_n)^2$ minimum is in-sample mean $h_{\text{mean}} = \frac{1}{N} \sum_{n=1}^{N} y_n$.

(b) $E_{\text{in}}(h) = \sum_{n=1}^{N} |h - y_n|$

$$\frac{\partial E_{\text{in}}(h)}{\partial h} = \sum_{n=1}^{N} \text{sign}(h - y_n), \text{ where } \text{sign}(x) = \begin{cases} 1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0 \end{cases}$$

Since $E_{\text{in}}(h)$ could be seen as the L_1-norm of vector $[(h, h, \cdots, h) - (y_1, y_2, \cdots, y_N)]$, and we have known that all valid norm functions are convex.

So to minimize $E_{\text{in}}(h)$, we just need to let $\frac{\partial E_{\text{in}}(h)}{\partial h} = \sum_{n=1}^{N} \text{sign}(h - y_n) = 0$.

Let h^* to be the median of $y_1 \le y_2 \le \cdots \le y_N$, so we could know that

$$y_1 \le \cdots \le y_{\lfloor \frac{N}{2} \rfloor} \le h^* \le y_{\lceil \frac{N}{2} \rceil + 1} \le \cdots \le y_N$$

For $y_k \le h^*$, we have $\text{sign}(h^* - y_k) = 1$, and for $y_k \ge h^*$, we have $\text{sign}(h^* - y_k) = -1$.

And if N is odd, we have $y_{\lceil \frac{N}{2} \rceil} = h^*$, i.e. $\text{sign}(h^* - y_{\lceil \frac{N}{2} \rceil}) = 0$, and if N is even, then we have $\lfloor \frac{N}{2} \rfloor = \lceil \frac{N}{2} \rceil$.

And $\forall N$, the number of the $\text{sign}(h^* - y_k) = 1$ is equal to the number of the $\text{sign}(h^* - y_k) = -1$.

So

$$\sum_{n=1}^{N} \text{sign}(h^* - y_n) = 0$$

So above all, we have the estimition to make $E_{\text{in}}(h) = \sum_{n=1}^{N} |h - y_n|$ minimum is in-sample median h_{med}.

(c) As y_N is perturbed to $y_N + \epsilon, \epsilon \to \infty$, $h_{\text{mean}} = \frac{1}{N} \sum_{n=1}^{N} y_n$ will also $\to \infty$. But y_N is the largest value in the set $\{y_1, y_2, \cdots, y_N\}$, and other y_i remain, and y_N is not the median, so h_{med} remains.

So above all, the h_{mean} will also increase to ∞, but the h_{med} will remain the same.