ML HW2

R13922146 葉富銘

Problem 5

It first say that if the degree is N, we need exactly N terms to uniquely determine the coefficients. So given only the first N - 1 terms, I think it may not able to determine the coefficients. So I wonder what it will illustrate to me next, but it then diretly says ”Since you have the first N - 1 terms, you can set up a system of linear equations by substituting P(x) for each known term, solving for the coefficients.” without any other explaination, and I can’t figure out the relationship but only find there seems to be a contradiction between these statements, and I get more confused. Next it shows an example which has polynomial of degree N = 2, but it has the first 3 terms of an integer sequence, violating the assumption that given the first (N - 1) = terms, so this example is also unconvincing. Therefore, I don’t agree with it.

Problem 6

Let’s first Group 16 numbers into 4 groups :

G1 = {1, 3, 5, 7}, G2 = {2, 4, 6, 8}, G3 = {9, 11, 13, 15}, G4 = {10, 12, 14, 16}.

For each group, it’s easy to identify the type of tickets needed to make every number in the group be green:

$$\begin{aligned} G1 &: \text{ type A, D} \\ G2 &: \text{ type B, D} \\ G3 &: \text{ type A, C} \\ G4 &: \text{ type B, C} \end{aligned}$$

According to this finding, to let at least one number on the five tickets to be purely green, the composition of the tickets can only be either one of the four listed above.

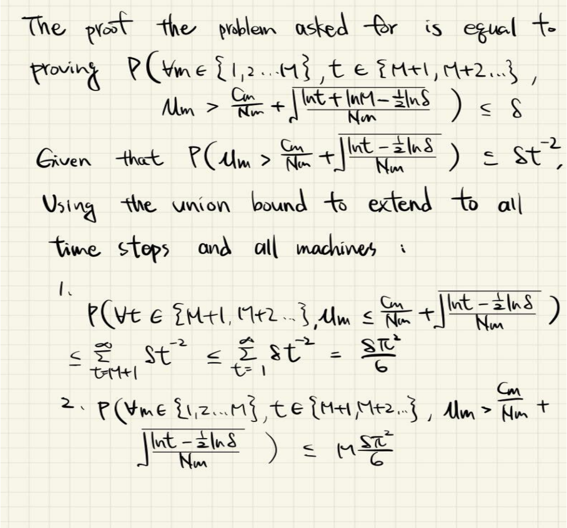
P(tickets $\in$ type A or D) = $\frac{1}{32}$, since the probability of drawing one ticket whose type is either A or D is $\frac{1}{2}$ beacuse of the same (super large) quantity each kind has, so the probability of drawing five tickets whose type are either A or D is $\left(\frac{1}{2}\right)^5 = \frac{1}{32}$. Using the same method above, we can derive P(tickets $\in$ type B or D) = P(tickets $\in$ type A or C) = P(tickets $\in$ type B or C) = $\frac{1}{32}$, P(tickets $\in$ type A) = P(tickets $\in$ type B) = P(tickets $\in$ type C)= P(tickets $\in$ type D) = $\left(\frac{1}{4}\right)^5 = \frac{1}{1024}$

P(the five tickets contain “some number” that is purely green) = P(tickets $\in$ type A or D) + P(tickets $\in$ type B or D) + P(tickets $\in$ type A or C) + P(tickets $\in$ type B or C) - P(tickets $\in$ type A) - P(tickets $\in$ type B) - P(tickets $\in$ type C) - P(tickets $\in$ type D) = $4\left(\frac{1}{32}\right)$ - $4\left(\frac{1}{1024}\right) = \frac{31}{256}$

Problem 7

Bad sample happens when the five tickets all contain green 5's. In such case, $E_{in} = 0$, and given $E_{out} = \frac{1}{2}$, set $\epsilon = 0.5$, we can calculate the bad probability $P[|E_{\text{in}}(h) - E_{\text{out}}(h)| > \epsilon] \leq 2e^{-\frac{N}{2}} = 0.16417$, which is the upperbound calculated by using Hoeffding' s Inequality.
The actual probability is $\frac{1}{32}$ since only tickets of Type A and Type D have a green number 5, so we can only drawn tickets from Type A and Type D. P(draw a ticket with green 5) = P(draw a ticket from type A or D) = $\frac{1}{2}$, so P(five green 5' s) = $\left(\frac{1}{2}\right)^5 = \frac{1}{32}$.

Problem 8



The proof the problem asked for is equal to proving $P\left(\forall m \in \{1,2 \ldots M\}, t \in \{M+1, M+2 \ldots\}, \mu_m > \frac{C_m}{N_m} + \sqrt{\frac{\ln t + \ln M - \frac{1}{2}\ln\delta}{N_m}}\right) \leq \delta$

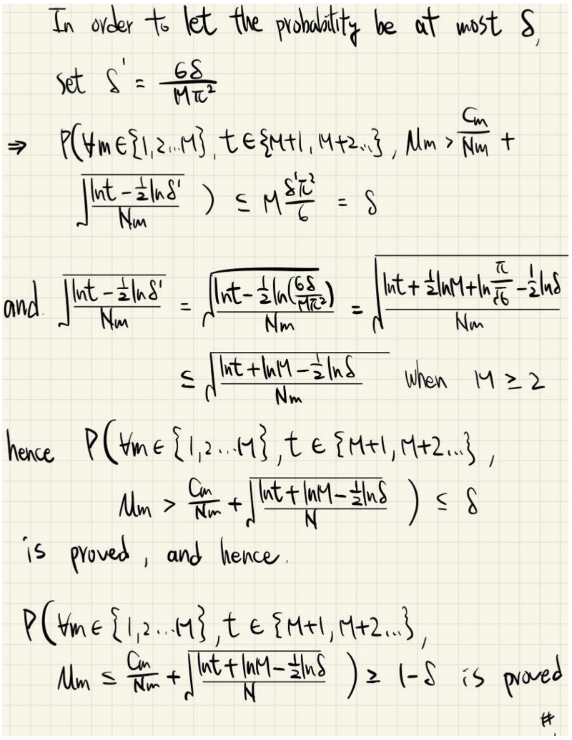
Given that $P\left(\mu_m > \frac{C_m}{N_m} + \sqrt{\frac{\ln t - \frac{1}{2}\ln\delta}{N_m}}\right) \leq \delta t^{-2}$,

Using the union bound to extend to all time stops and all machines :

1. $P\left(\forall t \in \{M+1, M+2 \ldots\}, \mu_m \leq \frac{C_m}{N_m} + \sqrt{\frac{\ln t - \frac{1}{2}\ln\delta}{N_m}}\right)$
$\leq \sum_{t=M+1}^{\infty} \delta t^{-2} \leq \sum_{t=1}^{\infty} \delta t^{-2} = \frac{\delta\pi^2}{6}$

2. $P\left(\forall m \in \{1,2 \ldots M\}, t \in \{M+1, M+2 \ldots\}, \mu_m > \frac{C_m}{N_m} + \sqrt{\frac{\ln t - \frac{1}{2}\ln\delta}{N_m}}\right) \leq M\frac{\delta\pi^2}{6}$

(a)



In order to let the probability be at most δ, set $\delta' = \frac{6\delta}{M\pi^2}$

$\Rightarrow P\left(\forall m \in \{1,2 \ldots M\}, t \in \{M+1, M+2 \ldots\}, \mu_m > \frac{C_m}{N_m} + \sqrt{\frac{\ln t - \frac{1}{2}\ln\delta'}{N_m}}\right) \leq M\frac{\delta'\pi^2}{6} = \delta$

and $\sqrt{\frac{\ln t - \frac{1}{2}\ln\delta'}{N_m}} = \sqrt{\frac{\ln t - \frac{1}{2}\ln\left(\frac{6\delta}{M\pi^2}\right)}{N_m}} = \sqrt{\frac{\ln t + \frac{1}{2}\ln M + \ln\frac{\pi}{\sqrt{6}} - \frac{1}{2}\ln\delta}{N_m}}$

$\leq \sqrt{\frac{\ln t + \ln M - \frac{1}{2}\ln\delta}{N_m}}$ when $M \geq 2$

hence $P\left(\forall m \in \{1,2 \ldots M\}, t \in \{M+1, M+2 \ldots\}, \mu_m > \frac{C_m}{N_m} + \sqrt{\frac{\ln t + \ln M - \frac{1}{2}\ln\delta}{N}}\right) \leq \delta$

is proved, and hence.

$P\left(\forall m \in \{1,2 \ldots M\}, t \in \{M+1, M+2 \ldots\}, \mu_m \leq \frac{C_m}{N_m} + \sqrt{\frac{\ln t + \ln M - \frac{1}{2}\ln\delta}{N}}\right) \geq 1-\delta$ is proved

#

(b)

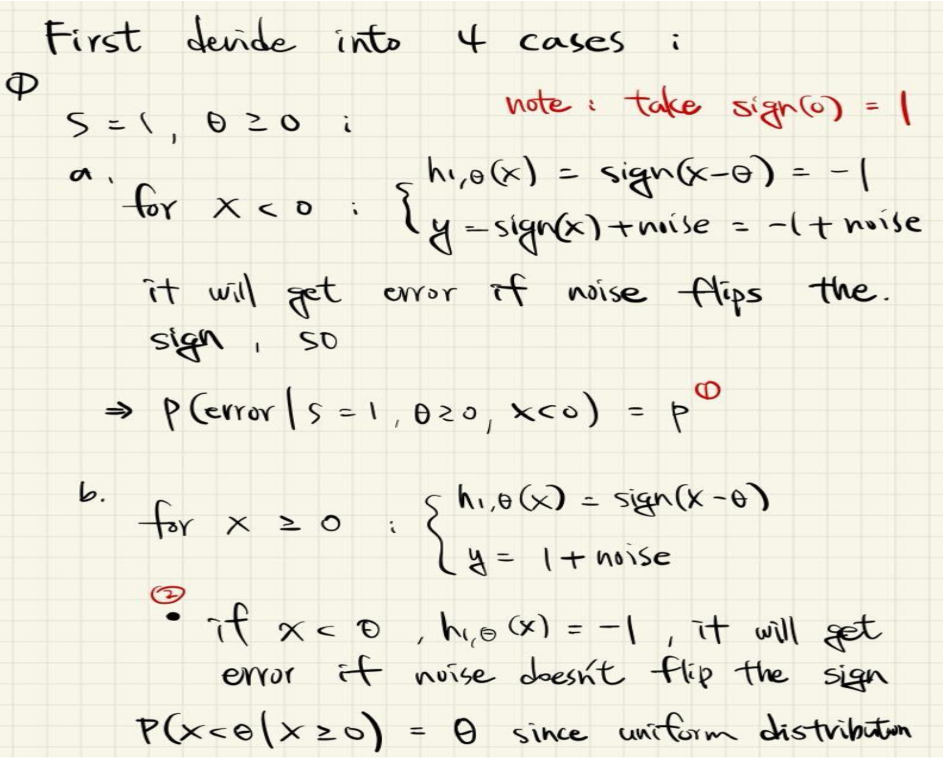
Problem 9

1. N = k + 1:
   We can construct a set of k inputs $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_k\}$, each with different number of ones, range from 0 to k. Hence each $\mathbf{x}_i$ can be assign either 1 or -1 independently. result in total 2^N dichotomies, hence it can be shattered.

2. N = k + 2:
   Since we only have k + 1 possible situations (0, 1, 2, ..., k) of number of ones in each input, but we have k + 2 inputs, it must have two inputs $\mathbf{x}_i$, $\mathbf{x}_j$, $1 \leq$ i, j $\leq$ k + 2 , i $\neq$ j, that has the same number of ones, and the corresponding y_i, y_j must be the same due to the contraint of symmetric property, leading to total number of dichotomies $< 2^N$, hence it can not be shattered.

According to the above, the VC dimension is k + 1

Problem 10



First devide into 4 cases :

①

$S=1,\ \theta \geq 0$:

note : take sign(0) = 1

a. for $x<0$: $\begin{cases} h_{1,\theta}(x) = \text{sign}(x-\theta) = -1 \\ y = \text{sign}(x) + \text{noise} = -1 + \text{noise} \end{cases}$

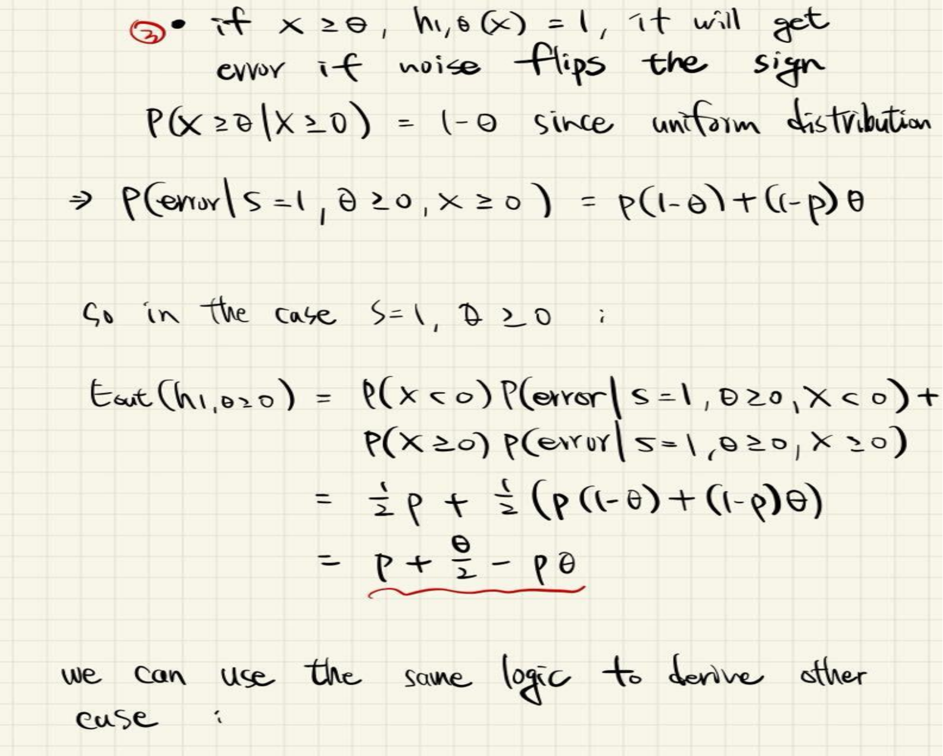
it will get error if noise flips the sign, so

$\Rightarrow P(\text{error} \mid s=1, \theta \geq 0, x<0) = p$ ①

b. for $x \geq 0$: $\begin{cases} h_{1,\theta}(x) = \text{sign}(x-\theta) \\ y = 1 + \text{noise} \end{cases}$

② • if $x<\theta$, $h_{1,\theta}(x) = -1$, it will get error if noise doesn't flip the sign

$P(x<\theta \mid x \geq 0) = \theta$ since uniform distribution



③ • if $x \geq \theta$, $h_{1,\theta}(x) = 1$, it will get error if noise flips the sign

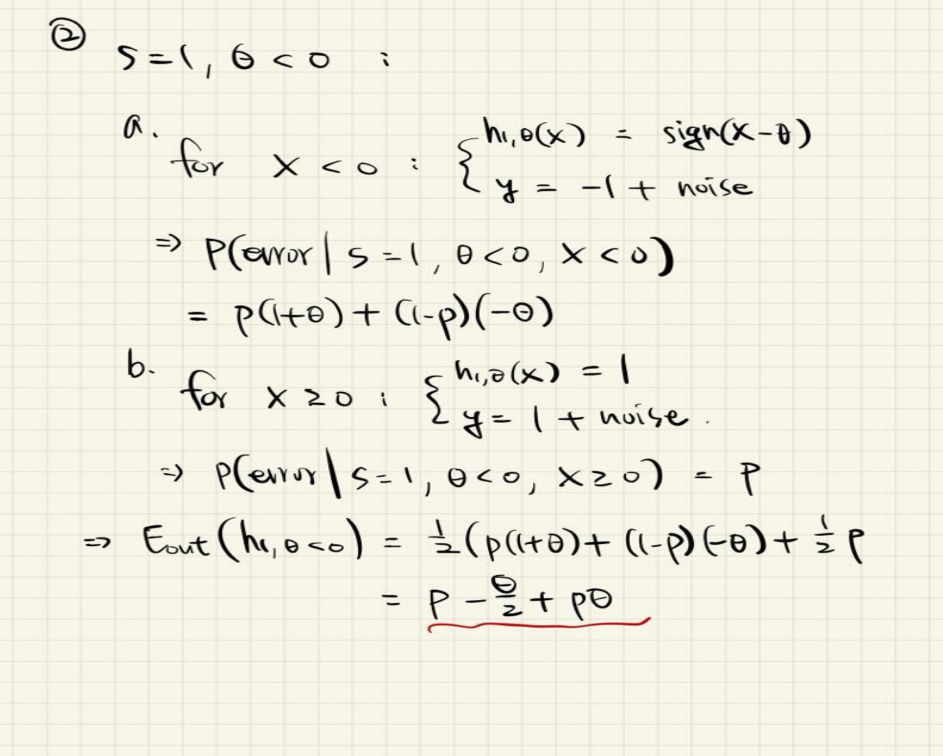
$P(x \geq \theta \mid x \geq 0) = 1-\theta$ since uniform distribution

$\Rightarrow P(\text{error} \mid s=1, \theta \geq 0, x \geq 0) = p(1-\theta) + (1-p)\theta$

So in the case $S=1,\ \theta \geq 0$:

$$\begin{aligned} E_{out}(h_{1,\theta\geq 0}) &= P(x<0)P(\text{error} \mid s=1, \theta \geq 0, x<0) + \\ &\quad P(x \geq 0)P(\text{error} \mid s=1, \theta \geq 0, x \geq 0) \\ &= \tfrac{1}{2}p + \tfrac{1}{2}(p(1-\theta) + (1-p)\theta) \\ &= \underline{p + \tfrac{\theta}{2} - p\theta} \end{aligned}$$

we can use the same logic to derive other cuse :



② $s=1,\ \theta<0$:

a. for $x<0$: $\begin{cases} h_{1,\theta}(x) = \mathrm{sign}(x-\theta) \\ y = -1 + \text{noise} \end{cases}$

$\Rightarrow P(\text{error} \mid s=1,\ \theta<0,\ x<0)$

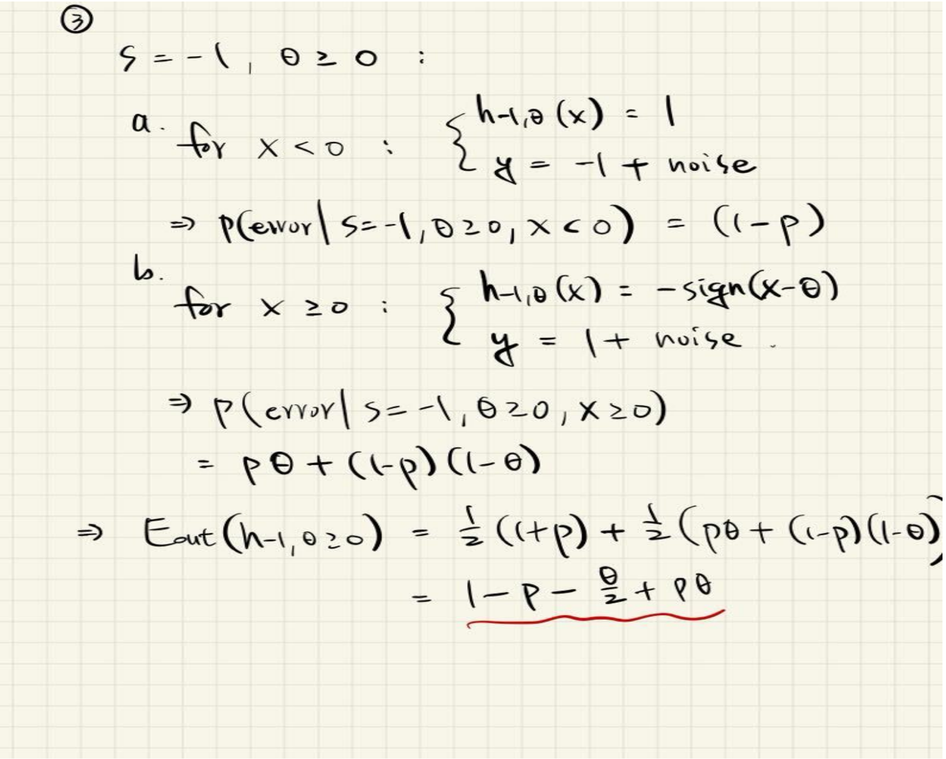
$= p(1+\theta) + (1-p)(-\theta)$

b. for $x \geq 0$: $\begin{cases} h_{1,\theta}(x) = 1 \\ y = 1 + \text{noise}. \end{cases}$

$\Rightarrow p(\text{error} \mid s=1,\ \theta<0,\ x\geq 0) = p$

$\Rightarrow E_{out}(h_{1,\theta<0}) = \frac{1}{2}(p(1+\theta) + (1-p)(-\theta) + \frac{1}{2}p$

$= \underline{p - \frac{\theta}{2} + p\theta}$



③ $s=-1,\ \theta \geq 0$:

a. for $x<0$: $\begin{cases} h_{-1,\theta}(x) = 1 \\ y = -1 + \text{noise} \end{cases}$

$\Rightarrow p(\text{error} \mid s=-1,\ \theta\geq 0,\ x<0) = (1-p)$

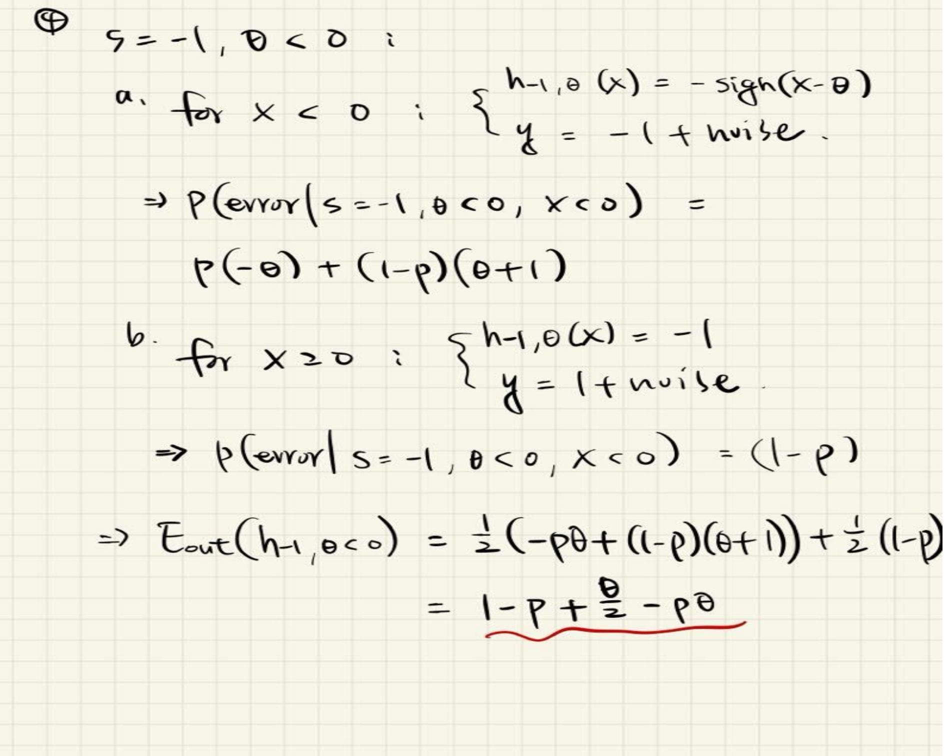
b. for $x \geq 0$: $\begin{cases} h_{-1,\theta}(x) = -\mathrm{sign}(x-\theta) \\ y = 1 + \text{noise}. \end{cases}$

$\Rightarrow p(\text{error} \mid s=-1,\ \theta\geq 0,\ x\geq 0)$

$= p\theta + (1-p)(1-\theta)$

$\Rightarrow E_{out}(h_{-1,\theta\geq 0}) = \frac{1}{2}(1+p) + \frac{1}{2}(p\theta + (1-p)(1-\theta))$

$= \underline{1 - p - \frac{\theta}{2} + p\theta}$



④ $s = -1, \theta < 0$:

a. for $x < 0$: $\begin{cases} h_{-1,\theta}(x) = -\text{sign}(x-\theta) \\ y = -1 + \text{noise}. \end{cases}$

$\Rightarrow p(\text{error} \mid s = -1, \theta < 0, x < 0) =$

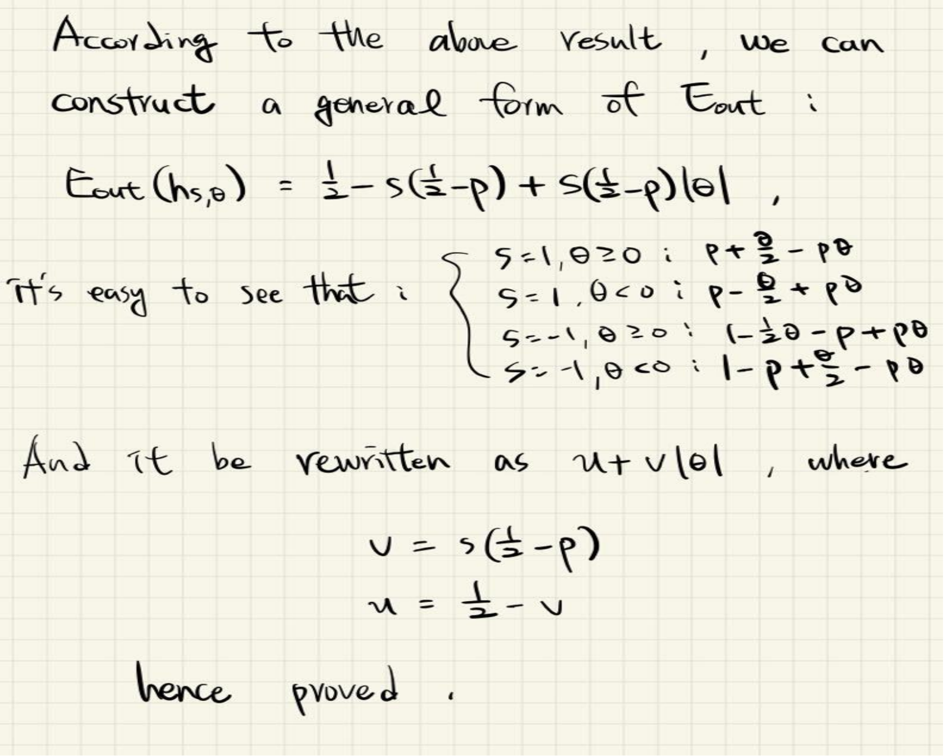
$p(-\theta) + (1-p)(\theta+1)$

b. for $x \geq 0$: $\begin{cases} h_{-1,\theta}(x) = -1 \\ y = 1 + \text{noise}. \end{cases}$

$\Rightarrow p(\text{error} \mid s = -1, \theta < 0, x < 0) = (1-p)$

$$\Rightarrow E_{out}(h_{-1,\theta<0}) = \frac{1}{2}(-p\theta + (1-p)(\theta+1)) + \frac{1}{2}(1-p)$$

$$= \underline{1 - p + \frac{\theta}{2} - p\theta}$$



According to the above result, we can construct a general form of E_{out} :

$$E_{out}(h_{s,\theta}) = \frac{1}{2} - s(\frac{1}{2} - p) + s(\frac{1}{2} - p)|\theta| ,$$

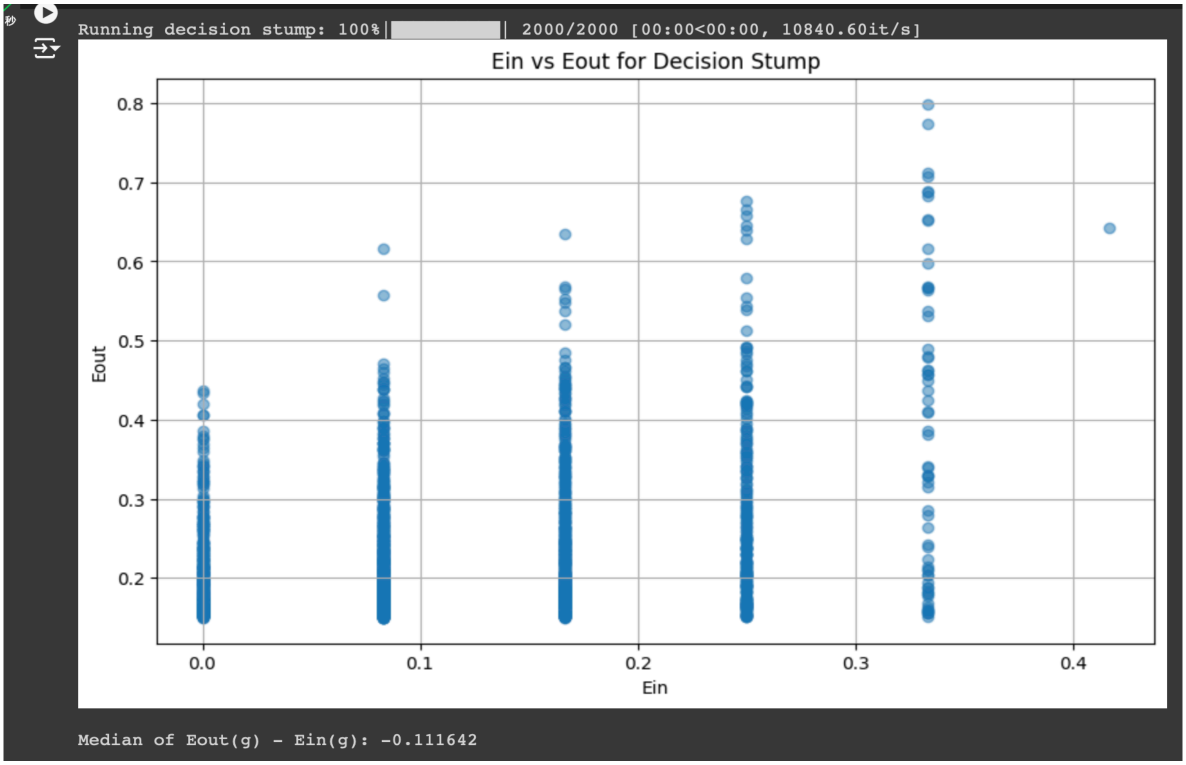
It's easy to see that : $\begin{cases} s = 1, \theta \geq 0 : p + \frac{\theta}{2} - p\theta \\ s = 1, \theta < 0 : p - \frac{\theta}{2} + p\theta \\ s = -1, \theta \geq 0 : 1 - \frac{1}{2}\theta - p + p\theta \\ s = -1, \theta < 0 : 1 - p + \frac{\theta}{2} - p\theta \end{cases}$

And it be rewritten as $u + v|\theta|$, where

$$v = s(\frac{1}{2} - p)$$
$$u = \frac{1}{2} - v$$

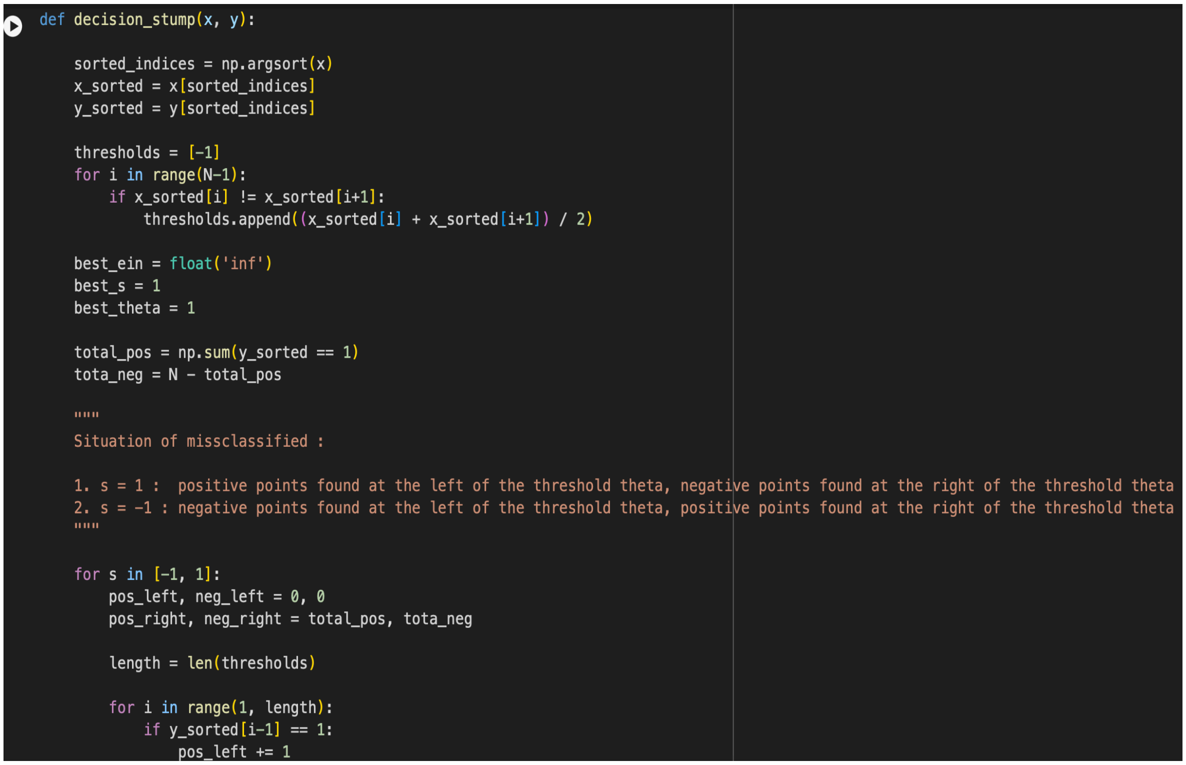
hence proved .

Problem 11

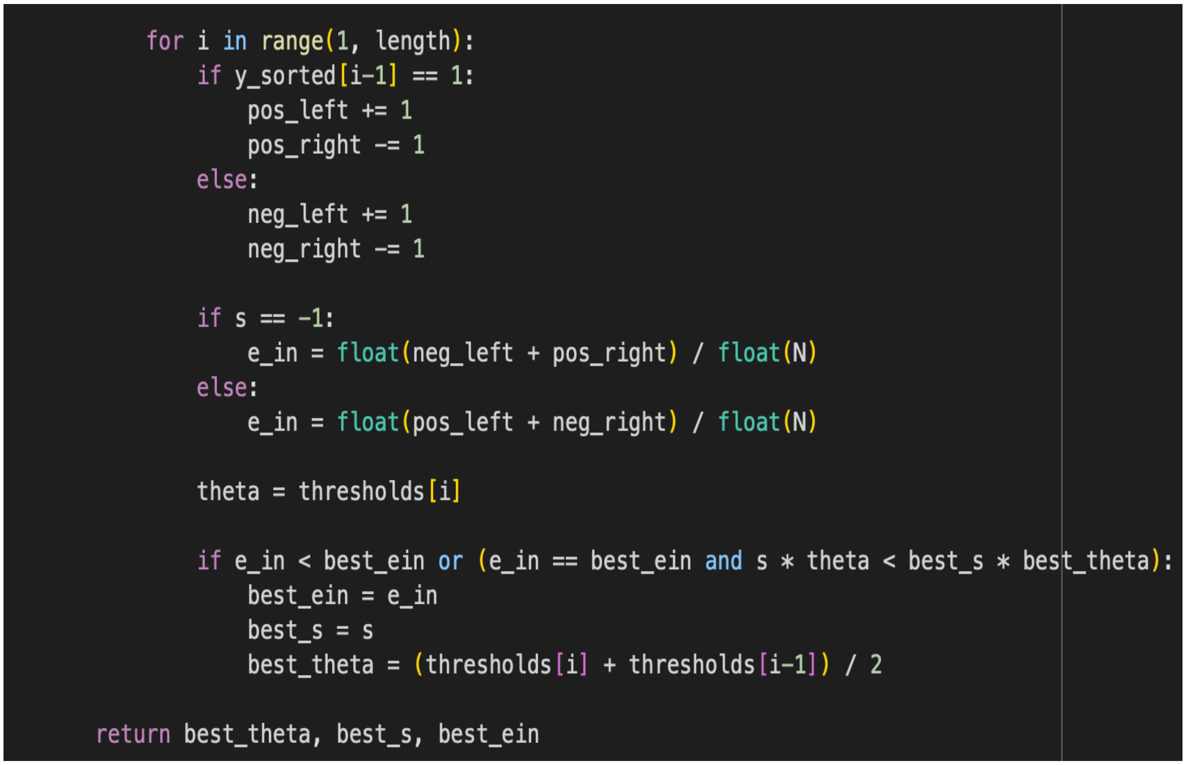


```
Running decision stump: 100%|██████████| 2000/2000 [00:00<00:00, 10840.60it/s]
```

```
Median of Eout(g) - Ein(g): -0.111642
```

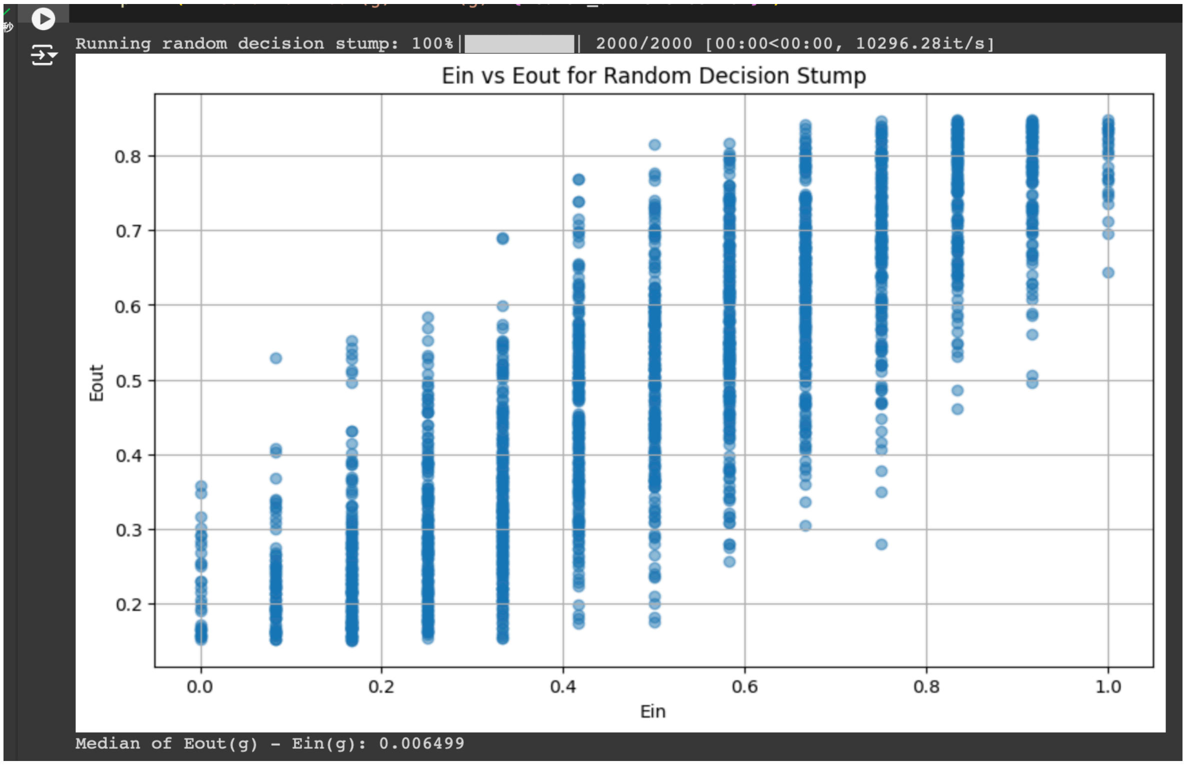


```
def decision_stump(x, y):

    sorted_indices = np.argsort(x)
    x_sorted = x[sorted_indices]
    y_sorted = y[sorted_indices]

    thresholds = [-1]
    for i in range(N-1):
        if x_sorted[i] != x_sorted[i+1]:
            thresholds.append((x_sorted[i] + x_sorted[i+1]) / 2)

    best_ein = float('inf')
    best_s = 1
    best_theta = 1

    total_pos = np.sum(y_sorted == 1)
    tota_neg = N - total_pos

    """
    Situation of missclassified :

    1. s = 1 :  positive points found at the left of the threshold theta, negative points found at the right of the threshold theta
    2. s = -1 : negative points found at the left of the threshold theta, positive points found at the right of the threshold theta
    """

    for s in [-1, 1]:
        pos_left, neg_left = 0, 0
        pos_right, neg_right = total_pos, tota_neg

        length = len(thresholds)

        for i in range(1, length):
            if y_sorted[i-1] == 1:
                pos_left += 1
```



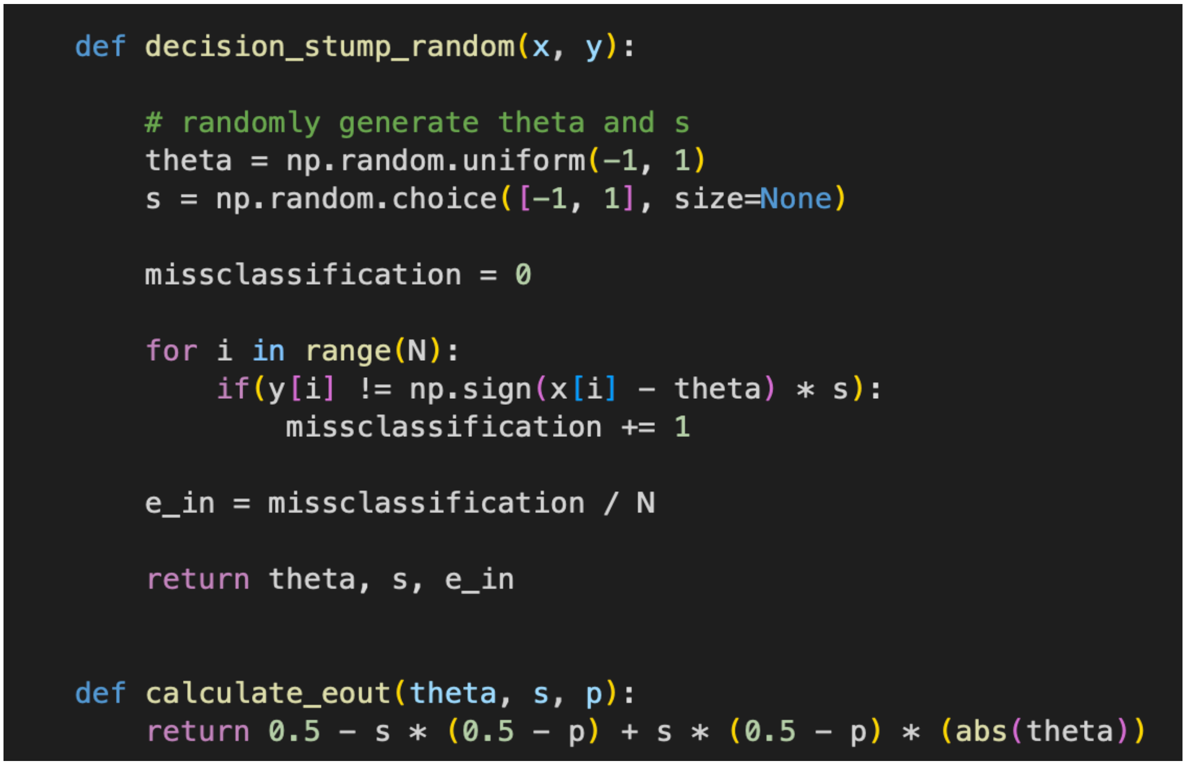
```
        for i in range(1, length):
            if y_sorted[i-1] == 1:
                pos_left += 1
                pos_right -= 1
            else:
                neg_left += 1
                neg_right -= 1

            if s == -1:
                e_in = float(neg_left + pos_right) / float(N)
            else:
                e_in = float(pos_left + neg_right) / float(N)

            theta = thresholds[i]

            if e_in < best_ein or (e_in == best_ein and s * theta < best_s * best_theta):
                best_ein = e_in
                best_s = s
                best_theta = (thresholds[i] + thresholds[i-1]) / 2

    return best_theta, best_s, best_ein
```

Problem 12



```
Running random decision stump: 100%|██████████| 2000/2000 [00:00<00:00, 10296.28it/s]
```

```
Median of Eout(g) - Ein(g): 0.006499
```



```python
def decision_stump_random(x, y):

    # randomly generate theta and s
    theta = np.random.uniform(-1, 1)
    s = np.random.choice([-1, 1], size=None)

    missclassification = 0

    for i in range(N):
        if(y[i] != np.sign(x[i] - theta) * s):
            missclassification += 1

    e_in = missclassification / N

    return theta, s, e_in


def calculate_eout(theta, s, p):
    return 0.5 - s * (0.5 - p) + s * (0.5 - p) * (abs(theta))
```

According to the lecture, the keypoint to successfull learning needs to consider two things :

1. small E_{in}
2. $E_{in} \approx E_{out}$

In problem 11, it selects the minimum E_{in} in each experiment, but it has the higher ϵ ($|E_{in} - E_{out}|$); In problem 12, it just randomly generates s and $theta$ to compute E_{in}, so it has more larger E_{in}, but it has lower ϵ. It indicates that it's hard to get both small E_{in} and ϵ, instead, we should strike a balance between them.