MLTech hw4

Problem 1

$$P(not\ be\ sampled) = (1 - \frac{1}{N})^{pN}$$

Since $N \to \infty$

$$\lim_{N\to\infty}(1 - \frac{1}{N})^{pN} = e^{-p}$$

期望值即$e^{-p}N$

Problem 2

可能的範圍 [0,0.35]

0是E_{out}彼此間沒有交集，0.35是前二者互斥但聯集恰好為第三種。

Problem 3

Maximum 的情況，我們可以想成$\frac{k+1}{2}$個model同樣產生錯誤，是最有效率增加總E_{out}的方式，因此maximum

$$E_{out} = \frac{\sum_{k=1}^{K} e_k}{\frac{K+1}{2}} = \frac{2}{K+1}\sum_{k=1}^{K} e_k$$

Problem 4

$$\frac{\sum_{i=1}^{N} y_i}{2N} * 2 = \frac{\sum_{i=1}^{N} y_i}{N} = 2$$

因為g是用decision tree，可以完全fitting，根據結果y=2

Problem 5

$$\frac{\partial \sum_{n=1}^{N}((y_n - s_n) - \alpha_t g_t(x_n))^2}{\partial \alpha_t}$$

$$= -\sum_{n=1}^{N} 2(y_n - s_n - \alpha_t g_t(x_n))g_t(x_n)$$

$= 0$ since α_t minimize the value

by the way $s_n \leftarrow s_n + \alpha_t g_t(x_n)$

So we can rewrite the equation as

$$\sum_{n=1}^{N} 2(y_n - s_n)g_t(x_n) = 0$$

=> $\sum_{n=1}^{N} s_n g_t(x_n) = \sum_{n=1}^{N} y_n g_t(x_n)$

Ans: $\sum_{n=1}^{N} y_n g_t(x_n)$

Problem 6

Since $g_1(x)$ is the function that minimized

$$(g(x_n) - y_n)^2$$

and the target function of α

$$(y_n - \alpha g(x_n))^2 = (\alpha g(x_n)^2 - y_n)^2$$

First of all, since multiplying a scalar with a general polynomial function is also a general polynomial function, we could claim that $\alpha = 1$.

Proof

If there exists $\alpha \neq 1$, that means we could find another minimum in the first equation. It contradicts with the definition of $g_1(x)$. Done

Problem 7

$g_2(x)$ is the function that minimized

$$(g_2(x_n) + g_1(x_n) - y_n)^2$$

Similar to the proof of the **Problem 6**.

Proof

First of all, since adding a general polynomial to another general polynomial, we could get another general polynomial. If there exists $g_2(x) \neq 0$, that means the function $(g_1(x_n) - y_n)^2$ isn't minimized. Thus, contradicts. Done

Problem 8

$$b = d - 1$$

$$w_1 = w_2 = \ldots = w_d = 1$$

只有當所有的都為-1時，才會為false，其餘都為True。

Problem 9

we assume sign(0) = 1

In the first layer we implement the following condition (Let the bias node be 1)

$Sum = 5$: weight (-5,1,1,1,1,1)

$Sum > -3$: weight (2,1,1,1,1,1)

$Sum < -3$: weight (-4,-1,-1,-1,-1,-1)

$Sum > 1$: weight (-2,1,1,1,1,1)

$Sum < 1$: weight (0,-1,-1,-1,-1,-1)

For the second layer, we want to utilize the condition. The goal is $Sum = -3, 1, 5$. With observing that $Sum < -3 \ and \ Sum > -3$, we find that it would only have the total sum 0 or 2 after sign. Thus we could implement the last layer as weight (-1,1,1,1,1,1) .

Problem 10

For all $\frac{\partial e_n}{\partial w_{ij}^{(l)}}$ would be zero

$\frac{\partial e_n}{\partial w_{i1}^{(L)}}$ would be zero

Since

$$\frac{\partial e_n}{\partial w_{i1}^{(L)}}$$

$$= \frac{\partial e_n}{\partial s_1^{(L)}} \frac{\partial s_1^{(L)}}{\partial w_{i1}^{(L)}}$$

$$= -2(y_n - s_1^{(L)})(x_i^{(L-1)})$$

For the output layer $x_i^{(L-1)} = 0$, since tanh(0) = 0.

$s_1^{(L)} = 0$, since initial w = 0

$\frac{\partial e_n}{\partial w_{ij}^{(l)}}$ would be zero

Since

$$\frac{\partial e_n}{\partial w_{ij}^{(l)}}$$

$$= \delta_j^{(l)}(x_i^{(l-1)})$$

$$= (\sum_k (\delta_k^{l+1})(w_{jk}^{(l+1)})(tanh'(s_j^{(l)})))(x_i^{(l-1)})$$

$$= (\sum_k (\delta_k^{l+1})(0)(tanh'(s_j^{(l)})))(x_i^{(l-1)})$$

$$= 0$$

Problem 11

First of all for

$$\delta_i^{(1)} = \sum_k (\delta_1^{(2)})(w_{ij})(tanh'(s_j^{(1)}))$$

$$= \sum_k (\delta_1^{(2)})(1)(tanh'(s_j^{(1)}))$$

$$\frac{\partial e_n}{\partial w_{ij}^{(1)}}$$

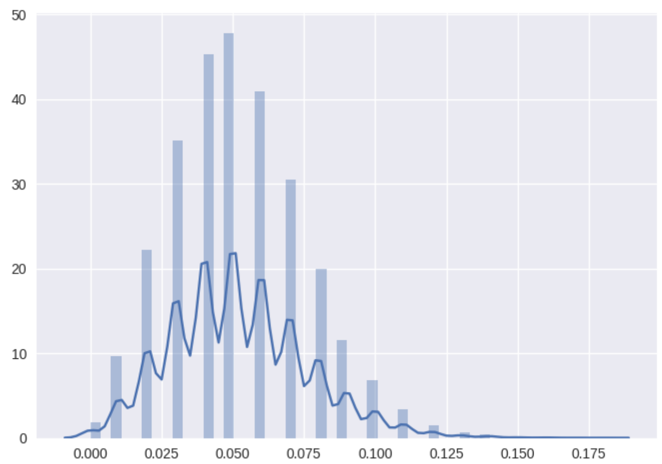
$$= \delta_j^{(1)}(x_i^{(0)})$$

$$= (x_i^{(0)}) \sum_k (\delta_1^{(2)})(tanh'(s_j^{(1)}))$$

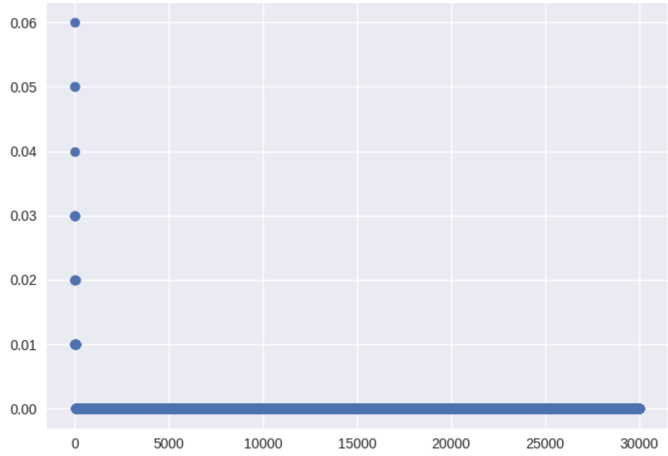
Since $s_j^{(1)} = \sum x_i^{(0)}$ which is independent to j

thus all $w_{ij}^{(1)}$ with the same i would be the same

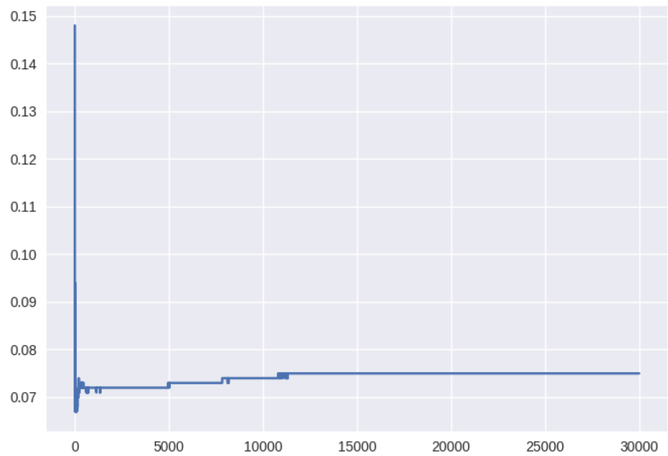
Problem 12



Problem 13

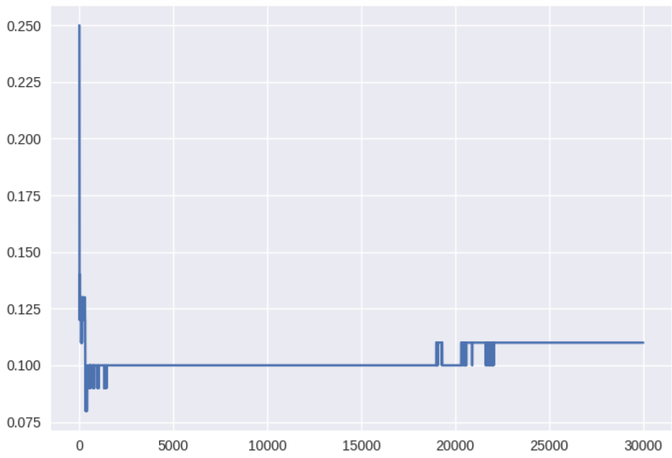


Problem 14

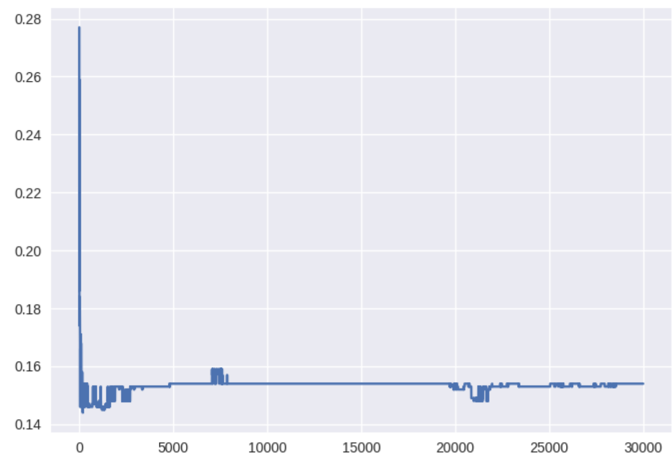


跟problem 13的圖相比，Eout並沒有像Ein泛化能力強烈，此外在更多棵樹時，Eout有些微的上升，這是因為Model對於Ein有些微的overfitting。至於最前面二者同樣的快速下降，可以視為Decision Tree Model強烈fitting的能力。

Problem 15



Problem 16



跟Ein相比，Eout同樣的增高許多。與Problem 13, 14類似，兩者在前期也呈現快速下降。但是在後期，Ein有上升的趨勢，Eout卻較為穩定。在這裡，因為base learner是一個decision stump 而已，而單一Decision stump 並沒有像Decision Tree有很強的fitting能力，因此Variance會較大，造成後期Ein還有可能上升，但相較之下，Eout也因為較弱的fitting能力，在後期並沒有太多overfit的情況。