1.

由柯西可得：

$$(\mu_1^2+\mu_2^2+\cdots+\mu_k^2)(1^2+1^2+\cdots+1^2)\geq(\mu_1+\mu_2+\cdots+\mu_n)^2$$

又由題目條件$\sum_{k=1}^{K}\mu_k=1$，可得：

$$\mathrm{K}\sum_{k=1}^{K}\mu_k^2\geq 1$$

故 Gini impurity 會滿足以下：

$$1-\sum_{k=1}^{K}\mu_k^2\leq 1-\frac{1}{K}=\frac{K-1}{K}$$

2.

題目給定$\mu_+ + \mu_- = 1$，故原式可化成：

$$\mu_+\big(1-(\mu_+-\mu_-)\big)^2+\mu_-\big(-1-(\mu_+-\mu_-)\big)^2$$

$$=(\mu_++\mu_-)-2(\mu_+-\mu_-)^2+(\mu_++\mu_-)(\mu_+-\mu_-)^2$$
$$=1-(\mu_+-\mu_-)^2=1-\mu_+^2-\mu_-^2+2\mu_+\mu_-$$
$$=1-\mu_+^2-\mu_-^2-\mu_+^2-\mu_-^2+\mu_+^2+\mu_-^2+2\mu_+\mu_-$$
$$=1-2\mu_+^2-2\mu_-^2+(\mu_++\mu_-)^2=2(1-\mu_+^2-\mu_-^2)$$

故 squared regression error 即為 Gini impurity 的 2 倍。

3.

易知對於每個 data 沒有被選取到的機率為$\left(1-\frac{1}{\mathrm{N}}\right)^{\mathrm{pN}}$，故不會被選到的 data 量之期望值為：

$$N*\left(1-\frac{1}{\mathrm{N}}\right)^{\mathrm{pN}}$$

又上式可轉換成：

$$N*\left(\left(\frac{1}{1+\frac{1}{N-1}}\right)^{N}\right)^{p}=N*\left(\frac{1}{\left(1+\frac{1}{N-1}\right)^{N}}\right)^{\mathrm{p}}$$

又因為題目說 N 極大，故上式可以化成：

$$\mathrm{N}*\left(\frac{1}{e}\right)^{\mathrm{p}}=N*e^{-p}$$

即得證題目之所求。

4.

在這 K 個 classification tree 中，總共犯錯的次數為$\sum_{k=1}^{K} e_k$。而對於每個 data 而言，若要 random forest 判斷錯誤，則需要在這 K 個 classification tree 中預測錯誤至少$\frac{\mathrm{K}+1}{2}$次。故最多會犯錯的 data 數量為：

$$\frac{\sum_{k=1}^{K} e_k}{\frac{\mathrm{K}+1}{2}} = \frac{2}{K+1}\sum\nolimits_{k=1}^{K} e_k$$

以上即為會犯錯的 data 數量的最大值，即為$\mathrm{E_{out}}(G)$的 upper bound。

7.

由上題加上初始條件$\mathrm{s_i} = 0$ 知，α_1可表示成：

$$\alpha_1 = \frac{\sum_{n=1}^{N} y_n g_1(x_n)}{\sum_{n=1}^{N} g_1^2(x_n)}$$

又因為找g(x)時要去 minimize $\sum_{n=1}^{N}\left(y_n - g_1(x_n)\right)^2$，在極值發生時，微分值要為 0；又因g(x)使用 regression，可用$\mathrm{x^T w}$表示。故可得：

$$2\mathrm{x^T}(\mathrm{x^T w} - \mathrm{y}) = 0 \ (\text{這邊的 } \mathrm{x, y, w} \text{ 均為向量})$$

乘上$\mathrm{w^T}$亦會為零：$\mathrm{w^T x}(\mathrm{w^T x} - \mathrm{y}) = \mathrm{w^T} 0 = 0$

故可得：

$$\mathrm{w^T x w^T x} = \mathrm{w^T x y}$$

$$(\mathrm{x^T} w)^2 = \mathrm{y}(\mathrm{x^T w})$$

$$\sum_{n=1}^{N} g_1^2(x_n) = \sum_{n=1}^{N} y_n g_1(x_n)$$

故$\alpha_1 = 1$。

8.

想要構造出$\mathrm{OR}(\mathrm{x_1} \dots \mathrm{x_d})$，目標要讓$\mathrm{x_1} \dots x_d$為$-1$時其值小於 0，其他則大於 0。故使：

$$\mathrm{w_i} = \begin{cases} d-1 & if\ i = 0 \\ 1 & if\ i = 1 \dots d \end{cases}$$

如此一來，若$\mathrm{x_1} \dots x_d$有任一為$+1$，則其值會> 0；全為-1時其值為-1。

9.

由 Backpropagation 的更新，我們可得偏微為：(取自課堂 slide)

$$\frac{\partial \mathrm{e_n}}{\partial w_{ij}^{(l)}} = \delta_j^{(l)}(x_i^{(l-1)})$$

而 δ_j 可表成：(取自課堂 slide)

$$\begin{cases}\delta_j^{(l)} = \sum_k \delta_k^{(l+1)} w_{jk}^{(l+1)} \tanh'(\mathrm{s_j^{(l)}}) \\ \delta_j^{(L)} = -2\left(y_n - s_j^{(L)}\right)\tanh'(\mathrm{s_j^{(l)}})\end{cases}$$

而因題目說初始化$w_{jk}^{(l)} = 0$，故可得$\delta_j^{(l)} = \sum_k 0 = 0$。

又因為初始化全部的$w_{jk}^{(l)} = 0$，故除了$x_0^{(L)}$外，$x_i^{(L)}$均為 0。

故滿足$\frac{\partial \mathrm{e_n}}{\partial w_{ij}^{(l)}} = \delta_j^{(l)}\left(x_i^{(l-1)}\right) = 0$的為除了$w_{0j}^{(L)}, j \in [1, output\ neuron\ number]$的所有其他 w。

10.

由 chain rule 可得：

$$\frac{\partial \mathrm{e}}{\partial s_k^{(L)}} = \sum\nolimits_{i=1}^{K} \frac{\partial \mathrm{e}}{\partial q_i} \frac{\partial q_i}{\partial s_k^{(L)}} = \sum\nolimits_{i=1}^{K} -\frac{v_i}{q_i} \frac{\partial q_i}{\partial s_k^{(L)}}$$

求$\frac{\partial q_i}{\partial s_k^{(L)}}$之值：$\Big($這邊避免版面過於凌亂，$s_k^{(L)}$簡化成用$s_k$表示$\Big)$

(1) $\mathrm{i = k}$：

$$\frac{\partial q_k}{\partial s_k^{(L)}} = \frac{e^{s_k}\left(\sum_{j=0}^{K} e^{s_j}\right) - e^{s_k} e^{s_k}}{\left(\sum_{j=0}^{K} e^{s_j}\right)^2} = \frac{e^{s_k}}{\left(\sum_{j=0}^{K} e^{s_j}\right)} - \left(\frac{e^{s_k}}{\left(\sum_{j=0}^{K} e^{s_j}\right)}\right)^2 = q_k - q_k^2$$

(2) $\mathrm{i \neq k}$：

$$\frac{\partial q_i}{\partial s_k^{(L)}} = \frac{-e^{s_i} e^{s_k}}{\left(\sum_{j=0}^{K} e^{s_j}\right)^2} = -q_i q_k$$

帶回原式可得：

$$\frac{\partial \mathrm{e}}{\partial s_k^{(L)}} = \sum\nolimits_{i=1}^{K} -\frac{v_i}{q_i} \frac{\partial q_i}{\partial s_k^{(L)}} = -\frac{v_k}{q_k}(q_k - q_k^2) + \sum_{i \in [1,K], i \neq k} \frac{v_i}{q_i} q_i q_k$$

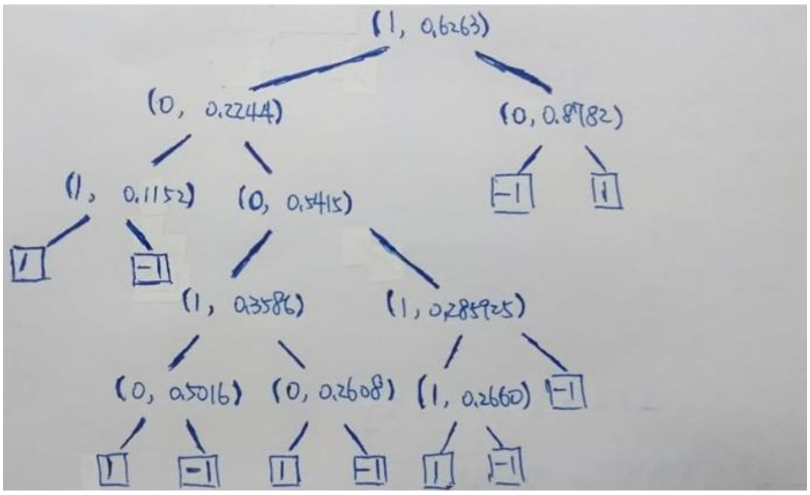
故$\frac{\partial \mathrm{e}}{\partial s_k^{(L)}}$為：

(1) $\mathrm{v}_k = 0$：$\frac{\partial \mathrm{e}}{\partial s_k^{(L)}} = q_k = q_k - 0 = q_k - v_k$

(2) $\mathrm{v_k} = 1$：$\frac{\partial \mathrm{e}}{\partial s_k^{(L)}} = -\frac{v_k}{q_k}(q_k - q_k^2) = v_k q_k - v_k = q_k - v_k$

故$\frac{\partial \mathrm{e}}{\partial s_k^{(L)}} = q_k - v_k$，得證。

11.



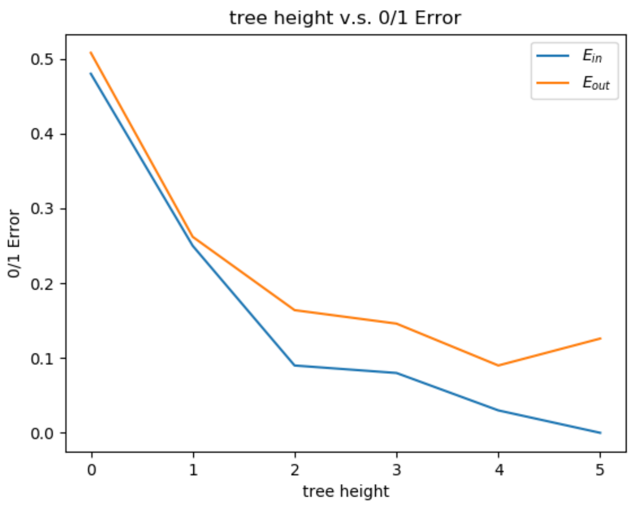
圖中括號內表示切開所使用之 n 和 θ ，以(n, θ)表示。(e.g. root 上之(1, 0.6263) 即為以用第 1 維來切，切的點為 0.6263)。

而有用方形框起來之位置為 leaf，裡面的值代表跑到該 leaf 時，decision tree 會將其判成甚麼值。

12.

Result: $E_{in} = 0, E_{out} = 0.126$

13.

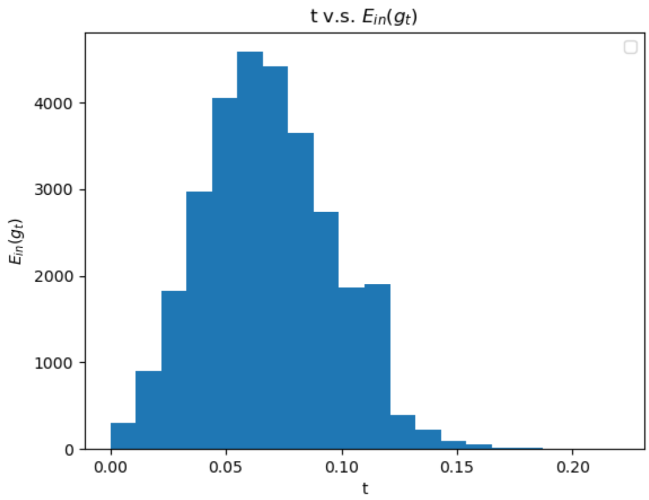


可以看出，隨著 Height 越高，E_{in}就越小。當到一定值後(相當於沒有 pruning)，E_{in}便等於 0。這是非常合理的事情，因為在 full-growing tree 時，$E_{in} = 0$，且隨著 purning 的程度越來越高，在 training data 的表現也會越來越差。

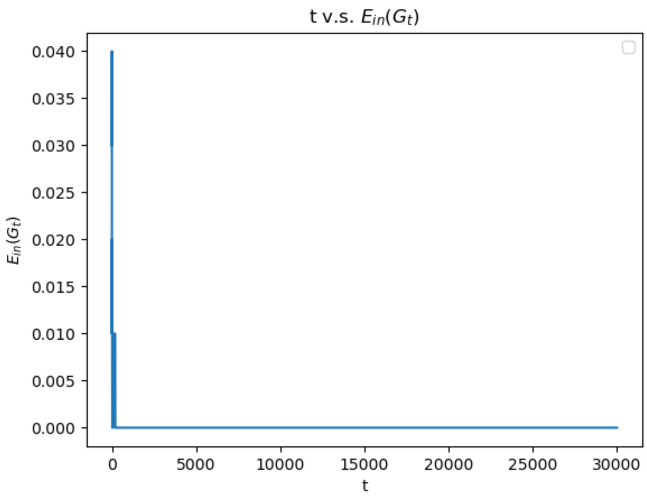
而E_{out}的趨勢為先降後升，也是相當合理。因為 Height 太深的時候雖然會使E_{in}降低，但已經 overfitting 了，故 testing data 的表現會變差。

(註：計算層數時，我並未將 root 視為一層)

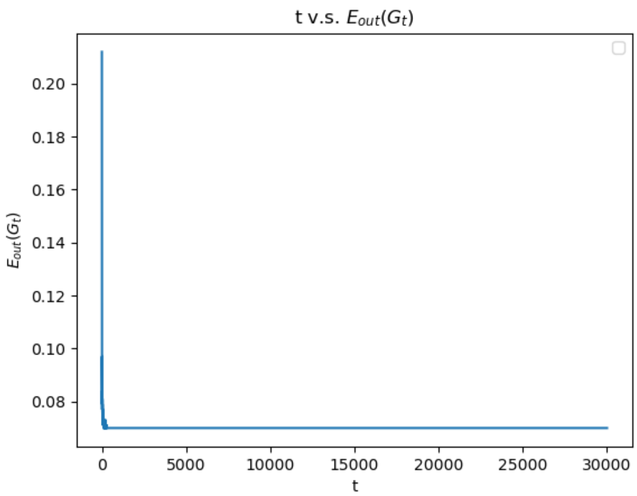
14.



15.



16.



可以看出，基本上$\mathrm{E_{in}}(G_t)$和$\mathrm{E_{out}}(G_t)$的走勢相近。而 t 夠大的時候，$\mathrm{E_{in}}(G_t)$和$\mathrm{E_{out}}(G_t)$都趨於定值，且$\mathrm{E_{in}}(G_t)$較$\mathrm{E_{out}}(G_t)$小，這是相當合理的。

17.

想法：判斷出有幾個+1(True)

故在第一層中，令除了連接x_0的所有 weight 為 1，而連接x_0的 weight 為

$$\{-d+1, -d+3, -d+5, \dots, -d+(2d-1)\}$$

(註：依序為判斷+1 的個數：$\{\geq d, \geq d-1, \dots, \geq 1\}$)

以上用數學式表達即為：

$$w_{ij}^{(1)} = \begin{cases} 1 & , for\ i\ = 1, 2, \dots, d \\ 2j-d-1 & , for\ i = 0 \end{cases}$$

接著，將$w_{j1}^{(2)}$從 $j = d$ 到 $j = 1$ 依序設成：$\{+1, -1, +1, -1, \dots\}$

(1) d 為偶數：則在有偶數個+1 時，總和為 0；有奇數個 + 1 時，總和為 2。

(2) d 為奇數：則在有偶數個+1 時，總和為 − 1；有奇數個 + 1 時，總和為 1。

故要使 output 正確，須使$w_{01}^{(2)}$的值為：

(1) d 為偶數：$w_{01}^{(2)} = 0$

(2) d 為奇數：$w_{01}^{(2)} = 1$

結合以上，可得：

$$w_{j1}^{(2)} = \begin{cases} [\![d \in 奇數]\!] & , for\ j = 0 \\ (-1)^{j+1} & , for\ j\ = 1, 2, \dots, d \end{cases}$$

18.

(1) 首先，當$d = 2$時：

易知$2-1-1$ 的 NN 為 Linear，又因 XOR 並非線性可分，故我們可知道$2-1-1$是不可能構出 XOR 的。

(2) 假設$d = k$成立，即$k-(k-1)-1$無法構成 XOR。

(3) 在$d = k+1$時：

因為我們可以知道$\mathrm{XOR}(x_1, x_2, x_3, \dots x_{k+1}) = \mathrm{XOR}(\mathrm{XOR}(x_1, x_2, x_3, \dots, x_k), x_{k+1})$。

(i) 若前 k 個連接 k-1 個 neuron，由(2)知無法成功構造出 XOR

(ii) 若前 k 個連接 k 個 neuron，其可簡化成：$\mathrm{XOR}(x', x_{k+1})$。又從(1)來看，我們可知其也無法構造出 XOR。

由數歸可得證，$d-(d-1)-1$無法構成 XOR。