Machine Learning Techniques HW8

- b03902089 林良翰

Random Forest

1.

- Every example has $\left(1-\frac{1}{N}\right)^{N'}=\left(1-\frac{1}{N}\right)^{Np}$ probability not to be sampled at all.
- $\lim_{N\to\infty}\left(1-\frac{1}{N}\right)^{Np}=e^{-p}$
- Thus, there are approximately $e^{-p}N$ examples will not be sampled at all.

2.

- If all the examples predicted wrongly by at most one classification tree (one of g_1, g_2, g_3), $E_{out}\left(G\right)$ has its minimum 0.
- The maximum ratio of examples which is predicted wrongly by at least two classification tree (two of g_1, g_2, g_3, or all of them) is 0.375 (e.g. g_1 and g_2 have 0.35 same error with g_3, g_1 and g_2 have more 0.025 same error but different with g_3).
- The range of $E_{out}\left(G\right)$ is $[0, 0.375]$

3. ?

- Assume that there are K (K is odd) binary classification trees $g_1, \dots, g_k$ with E_{out} $e_1, \dots, e_k$, if there is an error on an example, it's necessary that there are at least $\frac{K+1}{2}$ (more than half) trees make the mistake.
- To obtain the maximum $E_{out}\left(G\right)$, we can consider that we divide all classification trees into $\frac{K+1}{2}=M$ groups $G_1, \dots, G_M$, while each group contains at least one tree, and errors of every tree in the same group are mutual exclusive (maximize the error).
- With the assumption above, we can know

$$E_{out}\left(G\right)=\min_i E_{out}\left(G_i\right)$$

  where i is integer and $1\le i\le M$.
- "Minimum is always smaller than mean", by this simple theory, we could derive

$$\min_i E_{out}\left(G_i\right)\le\frac{1}{M}\sum_{i=1}^{M}E_{out}\left(G_i\right)$$

- The error of each group $G_1 \dots G_M$ is not necessary mutual exclusive, thus if all trees $g_1 \dots g_k$ have mutual exclusive errors, then

$$\frac{1}{M}\sum_{i=1}^{M} E_{out}\left(G_i\right) \leq \frac{1}{M}\sum_{k=1}^{K} E_{out}\left(g_k\right)$$

$$\frac{1}{M}\sum_{j=1}^{N} E_{out}\left(g_j\right) = \frac{2}{K+1}\sum_{k=1}^{K} e_k$$

- $\frac{2}{K+1}\sum_{k=1}^{K} e_k$ upper bounds $E_{out}\left(G\right)$

Gradient Boosting

- The algroithm of gradient boosting
  - Initialize $s_1 = s_2 = \cdots = s_N = 0$.
  - For $t = [1, 2, \ldots, N]$:
    - Obtain g_t by $\mathcal{A}\{(x_n, y_n - s_n)\}$, where $\mathcal{A}$ is a squared-error regression algorithm.
    - Compute $\alpha_t = \text{OneVariableLinearRegression}\{(g_t\left(x_n\right), y_n - s_n)\}$
    - Update $s_n \leftarrow s_n + \alpha_t g_t\left(x_n\right)$
  - Return $G\left(x\right) = \sum_{t=1}^{T} \alpha_t g_t\left(x\right)$

4.

- By computing the gradient of squared-error of $\{(g_t\left(x_n\right), y_n - s_n)\}$ over α_t

  $\frac{\partial \sum_{n=1}^{N}\left(\left(y_n - s_n\right) - \alpha_1 g_1\left(x_n\right)\right)^2}{\partial \alpha_1} = 0$

  $\Rightarrow 2\sum_{n=1}^{N}\left(\left(y_n - s_n\right) - \alpha_1 g_1\left(x_n\right)\right)\left(-g_1\left(x_n\right)\right) = 0$

  $\Rightarrow 2\sum_{n=1}^{N}\left(y_n - 0 - 2\alpha_1\right)\left(-2\right) = 0$

  $\Rightarrow { \alpha }_{ 1 }=\frac { 1 }{ 2N } \sum\limits_{ n=1 }^{ N }{ \left({ y }_{ n } \right) } $
- By the update function

  $s_n \leftarrow s_n + \alpha_1 g_1\left(x_n\right) = 0 + \frac{1}{2N}\sum_{n=1}^{N}\left(y_n\right)\cdot 2 = \frac{1}{N}\sum_{n=1}^{N}\left(y_n\right)$

5.

- The steepest η has the smallest gradient 0.

  $\frac{\partial \sum_{n=1}^{N}\left(\left(y_n - s_n\right) - \eta g_t\left(x_n\right)\right)^2}{\partial \eta} = 0$

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

  $\Rightarrow \sum_{n=1}^{N}\left(-y_n g_t\left(x_n\right) + s_n g_t\left(x_n\right) + \eta\left(g_t\left(x_n\right)\right)^2\right) = 0$

  $\Rightarrow \sum_{n=1}^{N} s_n g_t\left(x_n\right) = \sum_{n=1}^{N}\left(y_n g_t\left(x_n\right) - \eta\left(g_t\left(x_n\right)\right)^2\right) = \sum_{n=1}^{N} g_t\left(x_n\right)\left(y_n - \eta g_t\left(x_n\right)\right)$

6.

- $\mathcal{A}$ is an algorithm of linear regression, and we found the result $g_1(x) = w_1 x + b_1$
  w_1, b_1 are the optimal solution from linear regression, and must conform to the following equations

$$\begin{cases} \frac{\partial \sum_{n=1}^{N} \left(g_1(x_n) - \left(y_n - s_n\right)\right)^2}{\partial w_1} = \frac{\partial \sum_{n=1}^{N} \left(w_1 x_n + b_1 - \left(y_n - s_n\right)\right)^2}{\partial w_1} = 0 \\ \frac{\partial \sum_{n=1}^{N} \left(g_1(x_n) - \left(y_n - s_n\right)\right)^2}{\partial b_1} = \frac{\partial \sum_{n=1}^{N} \left(w_1 x_n + b_1 - \left(y_n - s_n\right)\right)^2}{\partial b_1} = 0 \end{cases}$$

  And obtain the following result

$$\begin{cases} w_1 = \frac{\sum_{n=1}^{N} x_n \left(y_n - s_n\right) - b_1 \sum_{n=1}^{N} x_n}{\sum_{n=1}^{N} (x_n)^2} \\ b_1 = \sum_{n=1}^{N} x_n \left(y_n - s_n\right) - w_1 \sum_{n=1}^{N} (x_n)^2 \end{cases}$$

- Suppose $\alpha_1 \neq 1$, then we can derive

$$\begin{cases} \frac{\partial \sum_{n=1}^{N} \left(\alpha_1 g_1(x_n) - \left(y_n - s_n\right)\right)^2}{\partial w_1} = \frac{\partial \sum_{n=1}^{N} \left(\alpha_1 w_1 x_n + \alpha_1 b_1 - \left(y_n - s_n\right)\right)^2}{\partial w_1} = 0 \\ \frac{\partial \sum_{n=1}^{N} \left(\alpha_1 g_1(x_n) - \left(y_n - s_n\right)\right)^2}{\partial b_1} = \frac{\partial \sum_{n=1}^{N} \left(\alpha_1 w_1 x_n + \alpha_1 b_1 - \left(y_n - s_n\right)\right)^2}{\partial b_1} = 0 \end{cases}$$

  And obtain

$$\begin{cases} w_1 = \frac{\sum_{n=1}^{N} x_n \left(y_n - s_n\right) - \alpha_1 b_1 \sum_{n=1}^{N} x_n}{\alpha_1 \sum_{n=1}^{N} (x_n)^2} \neq \frac{\sum_{n=1}^{N} x_n \left(y_n - s_n\right) - b_1 \sum_{n=1}^{N} x_n}{\sum_{n=1}^{N} (x_n)^2} \\ b_1 = \frac{1}{\alpha_1} \sum_{n=1}^{N} x_n \left(y_n - s_n\right) - w_1 \sum_{n=1}^{N} (x_n)^2 \neq \sum_{n=1}^{N} x_n \left(y_n - s_n\right) - w_1 \sum_{n=1}^{N} (x_n)^2 \end{cases}$$

  We found a more optimal w_1 and b_1

  $\Rightarrow$ Linear Regression $\mathcal{A}$ doesn't obtain the optimal solution.

  $\Rightarrow$ Contradiction.

- α_1 must be 1.

7.

- From 6., we have the result of first iteration

$$\begin{cases} w_1 = \frac{\sum_{n=1}^{N} x_n \left(y_n - s_n\right) - b_1 \sum_{n=1}^{N} x_n}{\sum_{n=1}^{N} (x_n)^2} = \frac{\sum_{n=1}^{N} x_n y_n - b_1 \sum_{n=1}^{N} x_n}{\sum_{n=1}^{N} (x_n)^2} \\ b_1 = \sum_{n=1}^{N} x_n \left(y_n - s_n\right) - w_1 \sum_{n=1}^{N} (x_n)^2 = \sum_{n=1}^{N} x_n y_n - w_1 \sum_{n=1}^{N} (x_n)^2 \end{cases}$$

where $s_n = s_n^1 = 0$, and w_1, b_1 are the optimal solution from $\mathcal{A}$.

And we get the new $s_n^2 = s_n + \alpha_1 g_1(x_n) = w_1 x_n + b_1$ at second iteration.

- Suppose we find $g_2(x_n) = w_2 x_n + b_2 \neq 0$.
  By the linear regression algorithm $\mathcal{A}$, we want the gradient of the error function to be 0.

$$\frac{\partial \sum_{n=1}^{N} \left(g_2\left(x_n\right) - \left(y_n - s_n^1\right)\right)^2}{\partial w_2} = \frac{\partial \sum_{n=1}^{N} \left(g_2\left(x_n\right) - \left(y_n - g_1\left(x_n\right)\right)\right)^2}{\partial w_2} = 0$$

$$\Rightarrow w_1 + w_2 = \frac{\sum_{n=1}^{N} y_n x_n - (b_1 + b_2) \sum_{n=1}^{N} x_n}{\sum_{n=1}^{N} (x_n)^2}$$

- Let $w' = w_1 + w_2$ and $b' = b_1 + b_2$.

$$w_2 x_n + b_2 \neq 0 \Rightarrow \begin{cases} w_2 = 0, b_2 \neq 0 \\ w_2 \neq 0, b_2 = 0 \\ w_2 \neq 0, b_2 \neq 0 \end{cases} \Rightarrow w' \neq w_1 \vee b' \neq b_1$$

  We found a more optimal solution w', b' from $\mathcal{A}$, which is different from w_1, b_1 $\Rightarrow$ Contradiction.

- $g_2(x_n)$ must be 0.

Neural Network

8.

- To implement $OR\left(x_1, x_2, \ldots, x_d\right)$, we need to make $w_0 = d - 1$, and $w_1 = w_2 = \cdots = w_d = +1$, meaning that the output of $\operatorname{sign}\left(\sum_{i=0}^{d} w_i x_i\right)$ will be positive if there is at least one x_i which is positive.
- $\left(w_0, w_1, \ldots, w_d\right) = \left(d - 1, \underbrace{+1, \ldots, +1}_{d}\right)$

9.

- The meaning of XOR is testing if there is odd number of positive examples.
- Thus we can use the formula of combinations

$$\binom{5}{5} + \binom{5}{3} + \binom{5}{1} = 1 + 10 + 5 = 16$$

- $D \geq 16$

10. ?

- The Error function

$$e_n = \left(y_n - x_1^{(L)}\right)^2 = \left(y_n - \tanh\left(s_1^{(L)}\right)\right)^2$$

$s_1^{(L)}$ is the linear combination from $(L-1)$ layer

$$s_1^{(L)} = \sum_{i=0}^{d^{(L-1)}} w_{i1}^{(L)} x_i^{(L-1)}$$

$x_1^{(L)}$ is the final output

$$x_1^{(L)} = \tanh\left(s_1^{(L)}\right) = \tanh\left(\sum_{i=0}^{d^{(L-1)}} w_{i1}^{(L)} x_i^{(L-1)}\right)$$

- Gradient of weights before output layer

$$\Rightarrow \quad \frac{\partial e_n}{\partial w_{i1}^{(L)}} = \frac{\partial\left(y_n - x_1^{(L)}\right)^2}{\partial x_1^{(L)}} \frac{\partial x_1^{(L)}}{\partial s_1^{(L)}} \frac{\partial s_1^{(L)}}{\partial w_{i1}^{(L)}}$$

$$= \quad \frac{\partial\left(y_n - \tanh\left(s_1^{(L)}\right)\right)^2}{\partial \tanh\left(s_1^{(L)}\right)} \frac{\partial \tanh\left(s_1^{(L)}\right)}{\partial s_1^{(L)}} \frac{\partial s_1^{(L)}}{\partial w_{i1}^{(L)}}$$

$$= \quad -2\left(y_n - \tanh\left(s_1^{(L)}\right)\right) \cdot \left(1 - \tanh^2\left(s_1^{(L)}\right)\right) \cdot x_i^{(L-1)}$$

$$= \quad \delta_1^{(L)} \cdot x_i^{(L-1)}$$

  - The notation $\delta_1^{(L)} = \frac{\partial e_n}{\partial s_1^{(L)}} = -2\left(y_n - \tanh\left(s_1^{(L)}\right)\right) \cdot \left(1 - \tanh^2\left(s_1^{(L)}\right)\right)$
  - Because all the weights are 0.

    $\Rightarrow x_i^{(L-1)} = \tanh\left(\sum_{j=0}^{d^{(L-2)}} w_{ji}^{(L-1)} x_j^{(L-2)}\right) = \tanh(0) = 0$
  - $\Rightarrow \frac{\partial e_n}{\partial w_{i1}^{(L)}} = 0$

- Gradient of other layers

$$\Rightarrow \quad \frac{\partial e_n}{\partial w_{ij}^{(\ell)}} = \frac{\partial e_n}{\partial s_j^{(\ell)}} \frac{\partial s_j^{(\ell)}}{\partial w_{ij}^{(\ell)}}$$

$$= \quad \left(\sum_{k=1}^{d^{(\ell+1)}} \frac{\partial e_n}{\partial s_k^{(\ell+1)}} \cdot \frac{\partial s_k^{(\ell+1)}}{\partial x_j^{(\ell)}} \cdot \frac{\partial x_j^{(\ell)}}{\partial s_j^{(\ell)}}\right) \cdot x_i^{(\ell-1)}$$

$$= \quad \left(\sum_{k=1}^{d^{(\ell+1)}} \frac{\partial e_n}{\partial s_k^{(\ell+1)}} \cdot \frac{\partial\left(\sum_{j=0}^{d^{(\ell)}} w_{jk}^{(\ell+1)} x_j^{(\ell)}\right)}{\partial x_j^{(\ell)}} \cdot \frac{\partial \tanh\left(s_j^{(\ell)}\right)}{\partial s_j^{(\ell)}}\right) \cdot x_i^{(\ell-1)}$$

$$= \quad \left(\sum_{k=1}^{d^{(\ell+1)}} \frac{\partial e_n}{\partial s_k^{(\ell+1)}} \cdot w_{jk}^{(\ell+1)} \cdot \left(1 - \tanh^2\left(s_j^{(\ell)}\right)\right)\right) \cdot x_i^{(\ell-1)}$$

$$= \quad \left(\sum_{k=1}^{d^{(\ell+1)}} \delta_k^{(\ell+1)} \cdot w_{jk}^{(\ell+1)} \cdot \left(1 - \tanh^2\left(s_j^{(\ell)}\right)\right)\right) \cdot x_i^{(\ell-1)}$$

- Because all the weights are 0.
  $\Rightarrow \frac{\partial e_n}{\partial w_{ij}^{(\ell)}} = 0$

- All the gradient components are 0.

11.

- From 10.

$$\frac{\partial e_n}{\partial w_{ij}^{(\ell)}} = \left(\sum_{k=1}^{d^{(\ell+1)}} \delta_k^{(\ell+1)} \cdot w_{jk}^{(\ell+1)} \cdot \left(1 - \tanh^2\left(s_j^{(\ell)}\right)\right) \right) \cdot x_i^{(\ell-1)}$$

- While $\ell = 1$

$$\frac{\partial e_n}{\partial w_{ij}^{(1)}} = \left(\sum_{k=1}^{d^{(2)}} \delta_k^{(2)} \cdot w_{jk}^{(2)} \cdot \left(1 - \tanh^2\left(s_j^{(1)}\right)\right) \right) \cdot x_i^{(0)}$$

$$\Rightarrow w_{jk}^{(2)} = w_{(j+1)k}^{(2)} = 1$$

- Thus we have to determine whether $s_j^{(1)}$ and $s_{j+1}^{(1)}$ are the same.

  After one iteration from initial weights with all ones

$$s_j^{(1)} = \sum_{i=0}^{d^{(0)}} w_{ij}^{(1)} x_i^{(0)}$$

$$s_{j+1}^{(1)} = \sum_{i=0}^{d^{(0)}} w_{i(j+1)}^{(1)} x_i^{(0)}$$

$$w_{ij}^{(1)} = w_{i(j+1)}^{(1)} = 1$$

  $\Rightarrow s_j^{(1)}$and $s_{j+1}^{(1)}$ are the same because the previous $w_{ij}^{(1)}$ and $w_{i(j+1)}^{(1)}$ are the same.

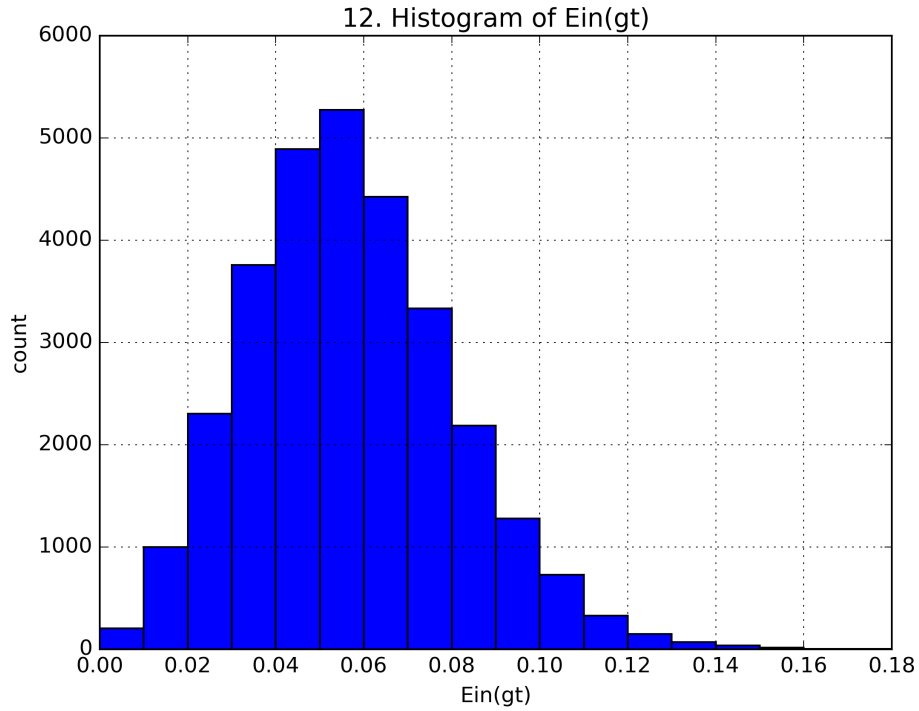
  $\Rightarrow$ The gradient of $w_{ij}^{(1)}$ and $w_{i(j+1)}^{(1)}$ are the same.

- After one iteration, $w_{ij}^{(1)}$ and $w_{i(j+1)}^{(1)}$ were updated to the same value.

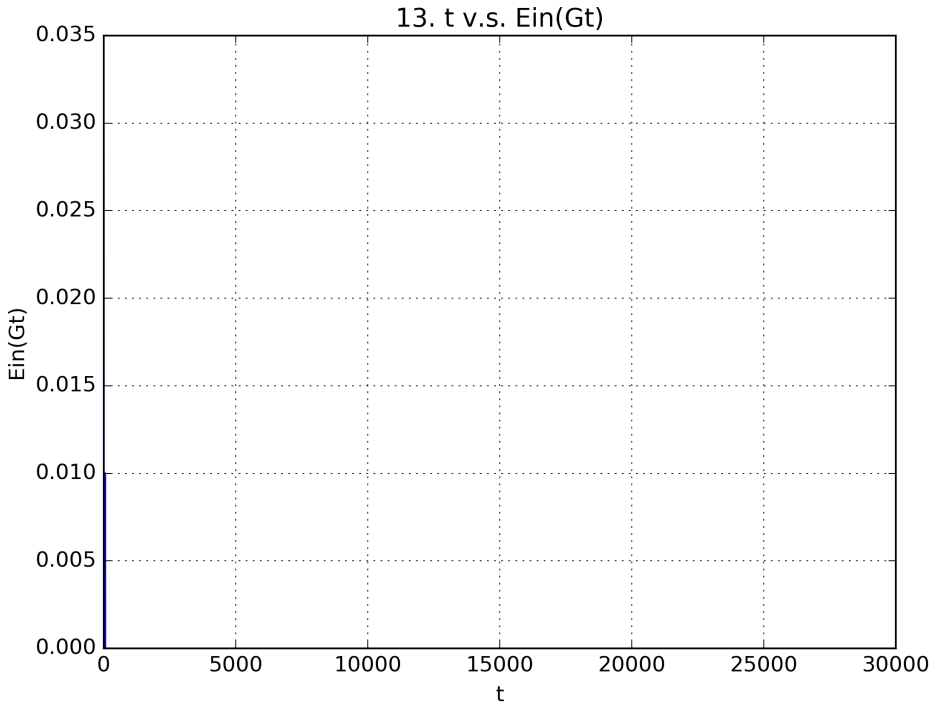
  $\Rightarrow$ Throughout the training preocess, the weights of $w_{ij}^{(1)}$ and $w_{i(j+1)}^{(1)}$ are always the same.

Experiments with Random Forest

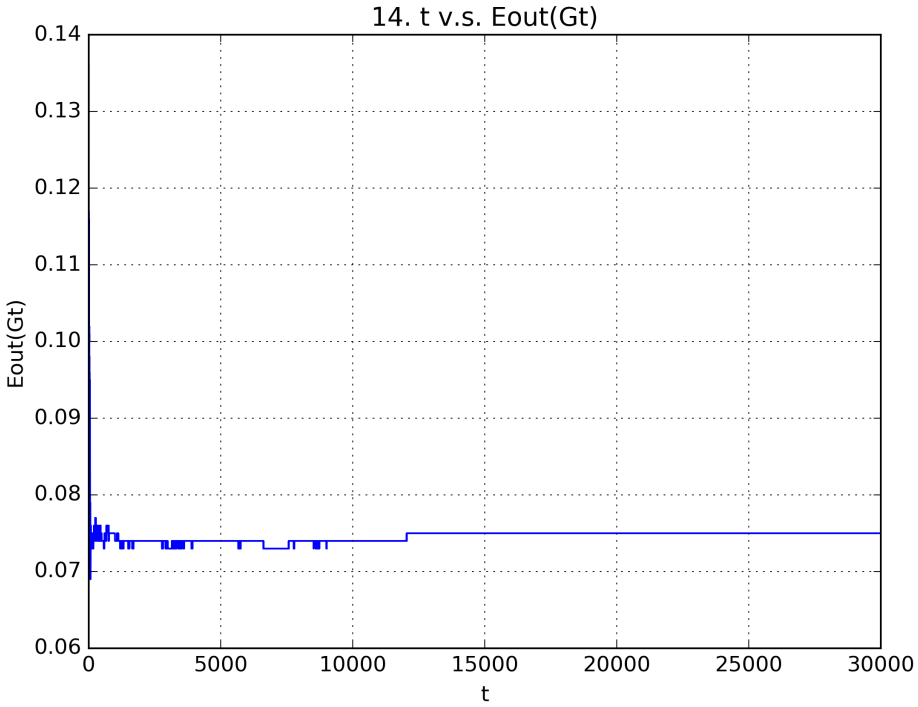
12.



13.

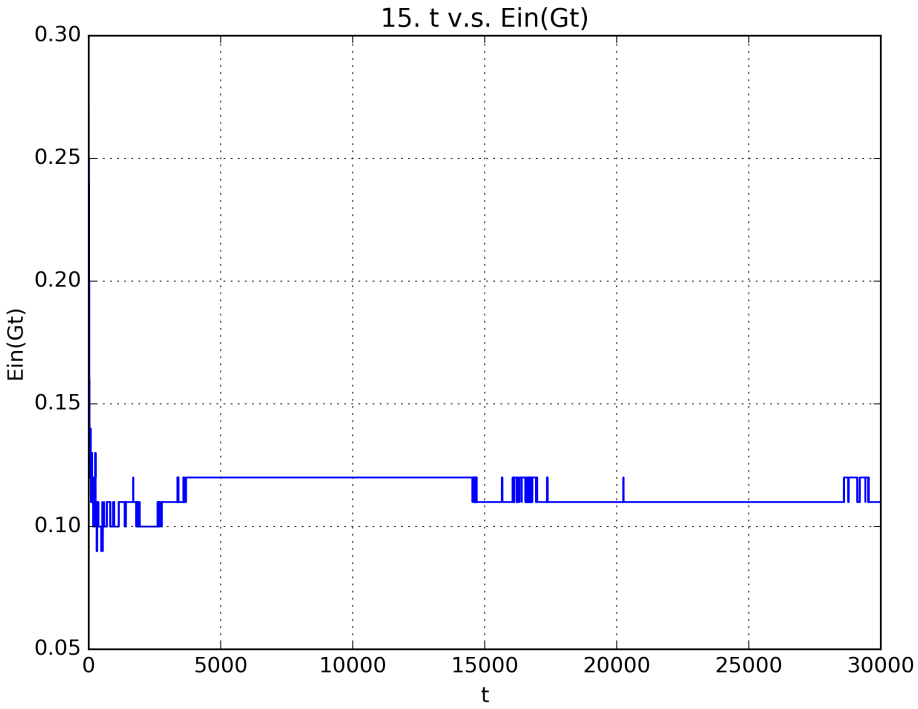


14.

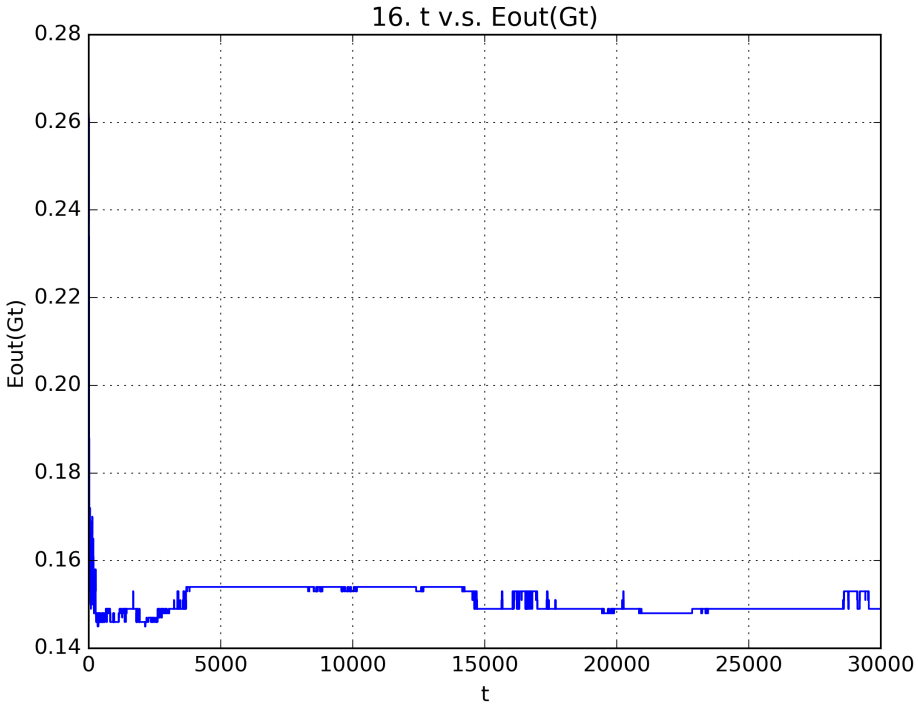


- The E_{in} goes down to 0 immediately, but E_{out} converges to a value instead.

15.



16.



- The $E_{out}(G_t)$ is higher than $E_{in}(G_t)$, and both converge to some values.