Homework 2 Answer

Zhao Wang u0905676

February 27, 2019

1 Paper Problems

1. (a) i. I prefer H_1 because H_1 uses less attributes that meets the principal of Occam's Razor: One should not make more assumptions than the minimum needed.

   ii. In the PAC, the learner receives samples and must select a generalization function (called the hypothesis) from a certain class of possible functions. In this case, H_1 is a smaller hypothesis so the learner will require only less examples in comparison to H_2.

   (b) From the inequality,

$$m > \frac{1}{\epsilon}\big(\log(|H|) + \log\frac{1}{\delta}\big)$$

$$m > \frac{1}{0.9}\big(\log_2(3^{10}) + \log_2\frac{1}{(1-0.95)}\big) \approx 22.413$$

   Therefore, at least 23 training examples are needed for L_1.

2. (a) This is PAC learnable because hypothesis space and the concept class is the same.
   (b) This is not PAC learnable because m is a fixed constant with m-of-n rules so the amount of samples are not polynomial.
   (c) This is PAC learnable because the amount of samples are polynomial.
   (d) This is not PAC learnable because the amount of samples are not polynomial.
   (e) This is PAC learnable because ID3's time complexity is polynomial.

3. We know that,

$$\epsilon_t = \frac{1}{2} - \frac{1}{2}\big(\sum D_t(i)y_i h_t(x_i)\big) = \frac{1}{2}\big(1 - \sum D_t(i)y_i h_t(x_i)\big)$$

$$= \frac{1}{2}\big(1 - \sum_{y_i = h_t(x_i)} D_t(i) + \sum_{y_i \neq h_t(x_i)} D_t(i)\big)$$

$$= \frac{1}{2}\big(1 - \sum_{y_i = h_t(x_i)} D_t(i) + \sum_{y_i \neq h_t(x_i)} D_t(i) + (\sum_{y_i = h_t(x_i)} D_t(i) + \sum_{y_i \neq h_t(x_i)} D_t(i) - 1)\big)$$

$$= \frac{1}{2}\big(1 - 1 + \sum_{y_i = h_t(x_i)} D_t(i) - \sum_{y_i = h_t(x_i)} D_t(i) + \sum_{y_i \neq h_t(x_i)} D_t(i) + \sum_{y_i \neq h_t(x_i)} D_t(i)\big)$$

$$\frac{1}{2}\big(2 \sum_{y_i \neq h_t(x_i)} D_t(i)\big)$$

Therefore,

$$\epsilon_t = \sum_{y_i \neq h_t(x_i)} D_t(i).$$

4. (a) $f(x_1, x_2, x_3) = x_1 \vee x_2 \vee x_3$

We have:

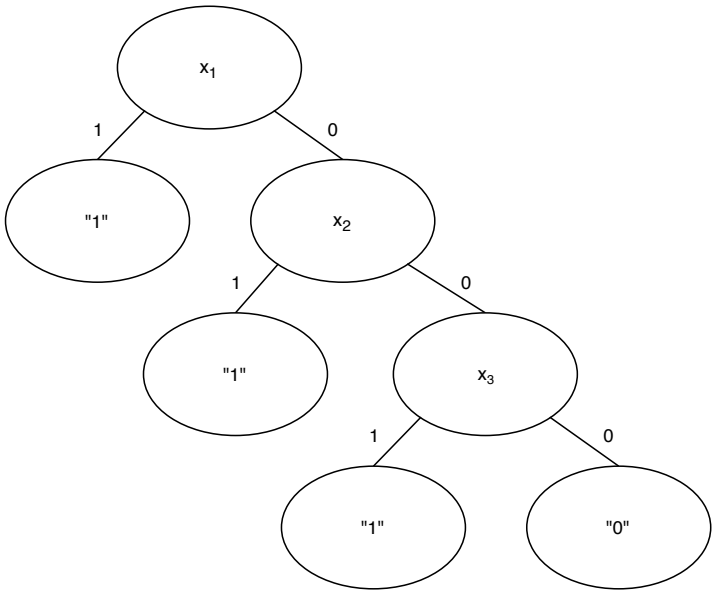
$$y = 1 \quad \text{when} \quad x_1 + x_2 + x_3 \geq 1$$

$$w = [1, 1, 1]$$

$$b = -1$$

and we get the hyper-plane and graph:

$$x_1 + x_2 + x_3 = 1$$



(b) $f(x_1, x_2, x_3) = x_1 \wedge \neg x_2 \wedge \neg x_3$

We have:

$$y = 1 \quad \text{when} \quad x_1 - x_2 - x_3 \geq 1$$

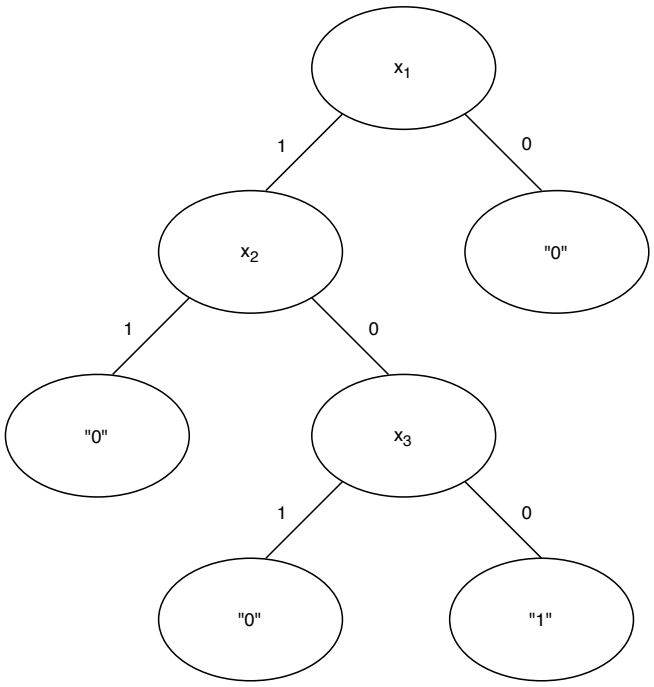
$$w = [1, -1, -1]$$

$$b = -1$$

and we get the hyper-plane:

$$x_1 - x_2 - x_3 = 1$$

and graph:



(c) $f(x_1, x_2, x_3) = \neg x_1 \vee \neg x_2 \vee \neg x_3$

We have:

$$y = 1 \quad \text{when} \quad -x_1 - x_2 - x_3 \geq -2$$

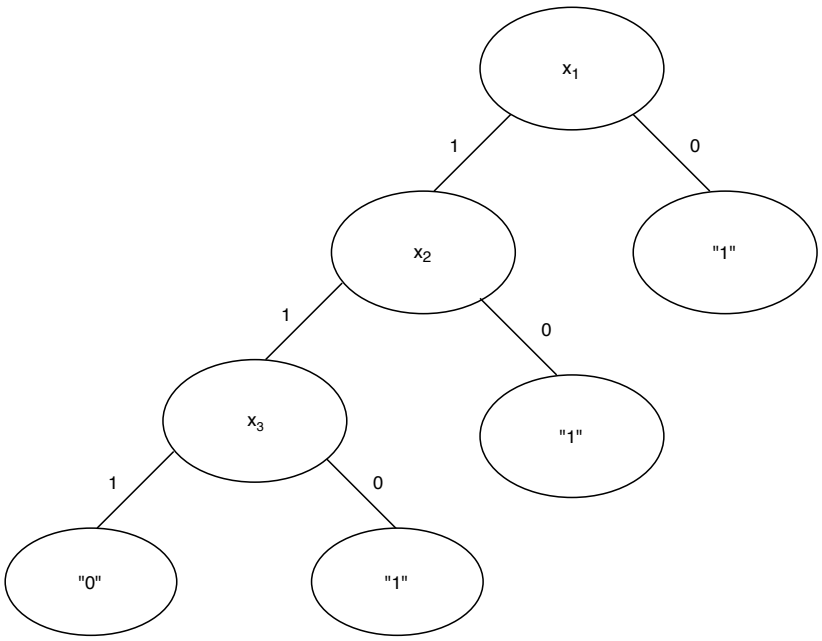
$$w = [-1, -1, -1]$$

$$b = 2$$

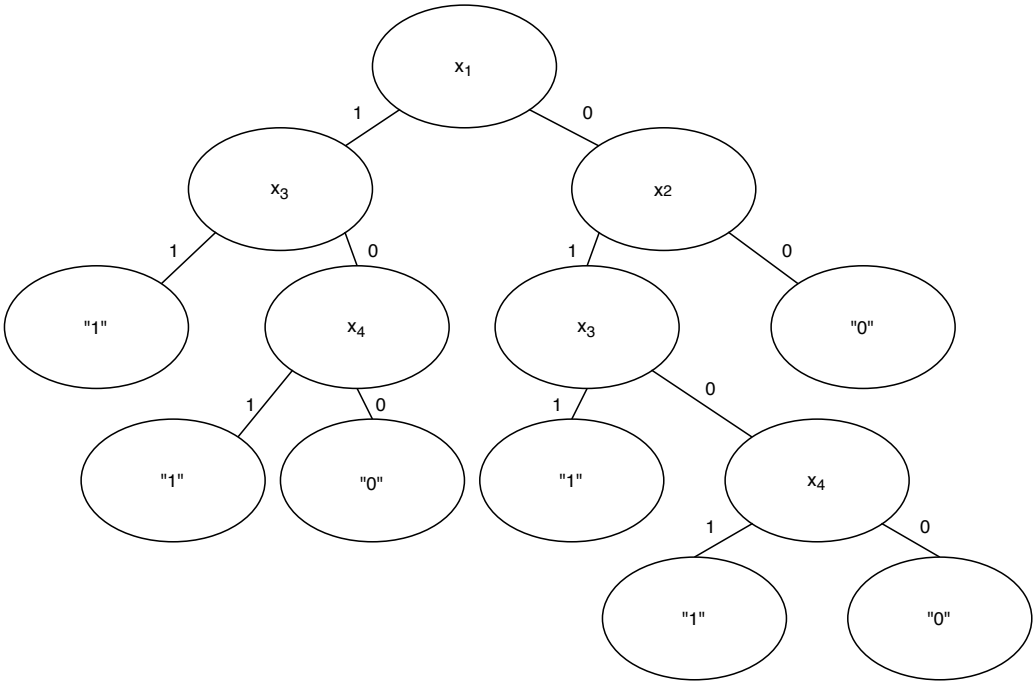
and we get the hyper-plane:

$$-x_1 - x_2 - x_3 = -2$$

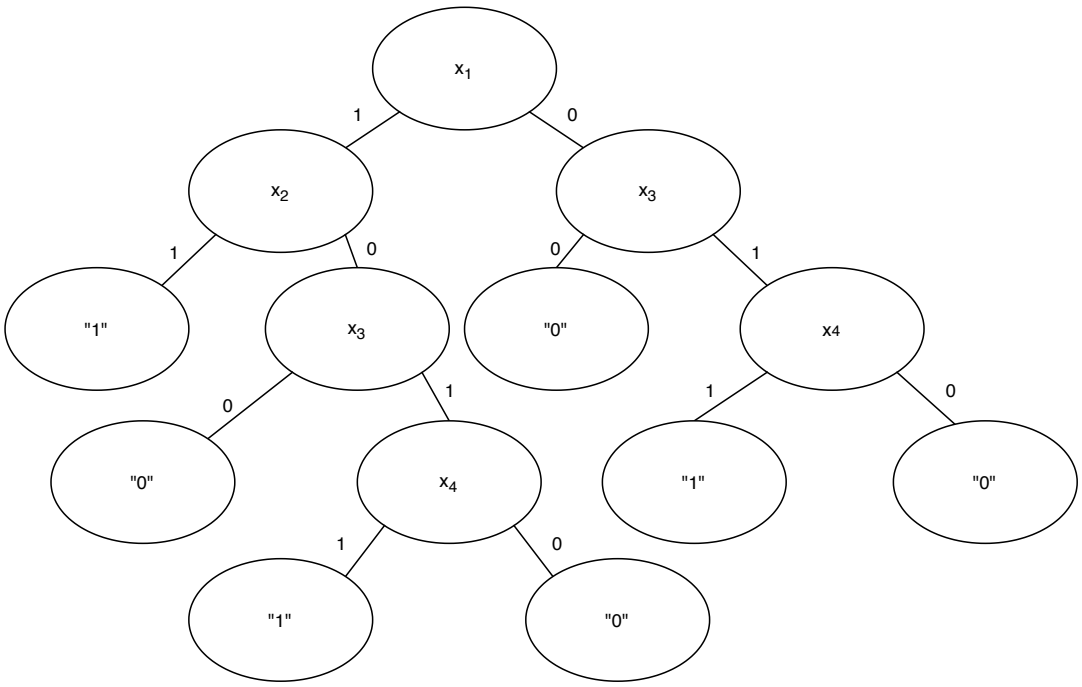
and graph:



(d) $f(x_1, x_2, x_3, x_4) = (x_1 \vee x_2) \wedge (x_3 \vee x_4)$
This function is not linearly separable.
Graph:



(e) $f(x_1, x_2, x_3, x_4) = (x_1 \wedge x_2) \vee (x_3 \wedge x_4)$
This function is not linearly separable.
Graph:



(f) All boolean functions can be drawn as the decision tree graphs. However, not every function has their own equivalent linear classier and hyperplane because some of them are not linear.

5. (a) $f(x_1, x_2) = (x_1 \wedge \neg x_2) \vee (\neg x_1 \wedge x_2)$
Let $x_3 = x_1 \wedge x_2$ and the new hyperplane will be $x_1 + x_2 + x_3 = 1$.

(b) $f(x_1, x_2) = (x_1 \wedge x_2) \vee (\neg x_1 \wedge \neg x_2)$
Let $x_3 = (x_1 \wedge \neg x_2) \vee (\neg x_1 \wedge x_2)$ and the new hyperplane will be $x_1 - x_2 - x_3 = 0$.

(c) $f(x_1, x_2, x_3)$ is listed in the following table

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	0
0	0	1	1
0	1	0	1
1	0	0	1
0	1	1	0
1	1	0	0
1	0	1	0
1	1	1	1

Let a 4_{th} dimension $x_4 = (x_1 \wedge \neg x_2 \wedge \neg x_3) \vee (\neg x_1 \wedge x_2 \wedge \neg x_3) \vee (\neg x_1 \wedge \neg x_2 \wedge x_3) \vee (x_1 \wedge x_2 \wedge x_3)$ then get $x_4 = 0.5$.

6. (a) $(\mathbf{x}^\top \mathbf{y})^2$
$\phi(\mathbf{x}) = [x_1^2, \sqrt{2}x_1x_2, x_2^2]$
$\phi(\mathbf{y}) = [y_1^2, \sqrt{2}y_1y_2, y_2^2]$

(b) $(\mathbf{x}^\top \mathbf{y})^3$
$\phi(\mathbf{x}) = [x_1^3, \sqrt{3}x_1^2 x_2, \sqrt{3}x_1 x_2^2, x_2^2]$
$\phi(\mathbf{y}) = [y_1^3, \sqrt{3}y_1^2 y_2, \sqrt{3}y_1 y_2^2, y_2^3]$

(c) $(\mathbf{x}^\top \mathbf{y})^k$ where k is any positive integer.
$\phi(\mathbf{x})_i = \sqrt{\frac{k}{i}} x_1^{k-i} x 2_i^2, i = 0, ..., k$
$\phi(\mathbf{y})_i = \sqrt{\frac{k}{i}} y_1^{k-i} y 2_i^2, i = 0, ..., k$

7. (a)

$$J(\mathbf{w}, b) = \frac{1}{2} \sum_{i=1}^{m} (y_i - \mathbf{w}^T \mathbf{x}_i - b)^2$$

(b) i. $\frac{\nabla J}{\nabla \mathbf{w}} = [-4, 2, -22]$, $\frac{\nabla J}{\nabla b} = -10$

ii. $\frac{\nabla J}{\nabla \mathbf{w}} = [-22, 16, -56]$, $\frac{\nabla J}{\nabla b} = -10$

iii. $\frac{\nabla J}{\nabla \mathbf{w}} = [7.5, -4, -5]$, $\frac{\nabla J}{\nabla b} = 4.5$

(c) The optimal $\mathbf{w}$ and b that minimize the cost function is

$$\mathbf{w} = [1, 1, 1]^\top$$
$$b = -1$$

(d) Step1: $stch_1$ = [1, -1, 2], $\frac{\nabla J}{\nabla \mathbf{w}} = [-1, 1, -2]$, $\frac{\nabla J}{\nabla b} = -1$
Therefore,

$$\mathbf{w} = [0.1, -0.1, 0.2]$$
$$b = 0.1$$

Step2: $stch_2$ = [1, 1, 3], $\frac{\nabla J}{\nabla \mathbf{w}} = [-3.3, -3.3, -9.9]$, $\frac{\nabla J}{\nabla b} = -3.3$
Therefore,

$$\mathbf{w} = [0.43, 0.23, 1.19]$$
$$b = 0.43$$

Step3: $stch_3$ = [-1, 1, 0], $\frac{\nabla J}{\nabla \mathbf{w}} = [-1.23, 1.23, 0.0]$, $\frac{\nabla J}{\nabla b} = 1.23$
Therefore,

$$\mathbf{w} = [0.553, 0.107, 1.19]$$
$$b = 0.307$$

Step4: $stch_4$ = [1, 2, -4], $\frac{\nabla J}{\nabla \mathbf{w}} = [-1.686, -3.372, 6.744]$, $\frac{\nabla J}{\nabla b} = -1.686$
Therefore,

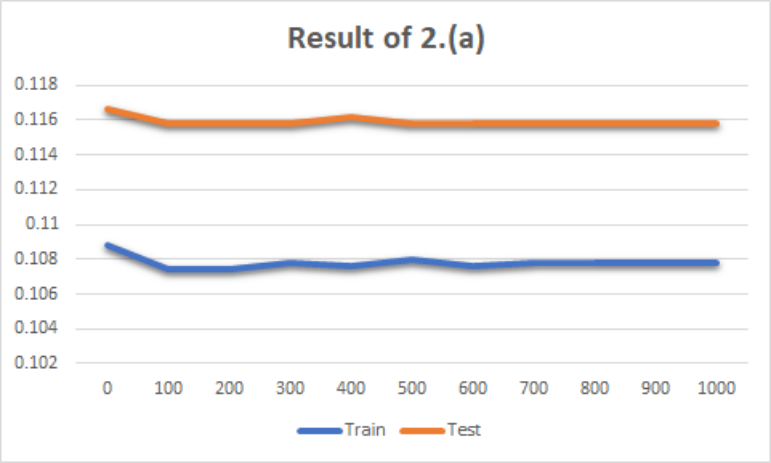
$$\mathbf{w} = [0.722, 0.444, 0.516]$$
$$b = 0.476$$

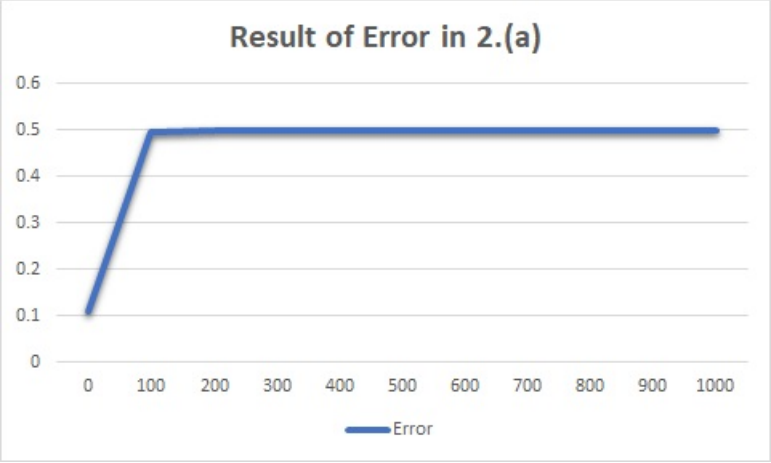
Step5: $stch_5$ s = [3, -1, -1], $\frac{\nabla J}{\nabla \mathbf{w}} = [5.042, -1.681, -1.681]$, $\frac{\nabla J}{\nabla b} = 1.681$
Therefore,

$$\mathbf{w} = [0.217, 0.612, 0.684]$$
$$b = 0.308$$

2 Practice

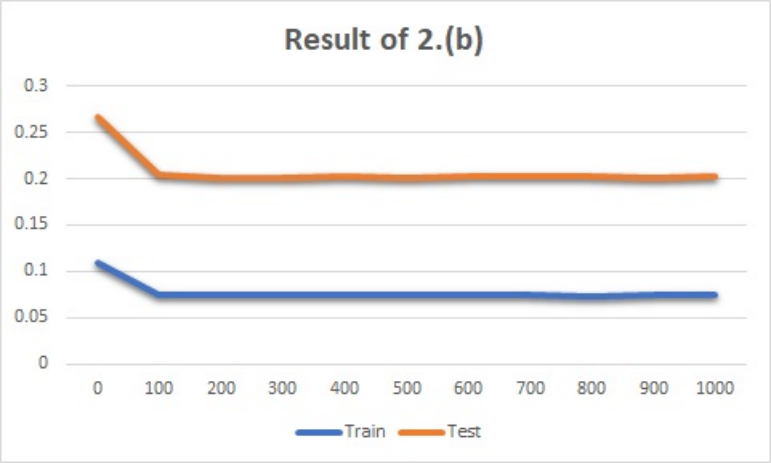
1. See the Github repo.
2. (a) The data is shown in the graphs below.





The data of testing error of AdaBoost is nearly to 0.5 but is better than the result in HW1.

(b) The data is shown in the graph below.



From the graph above, the data of bagging trees has higher train error and lower test error than single tree but lower train error and higher test error than AdaBoost.

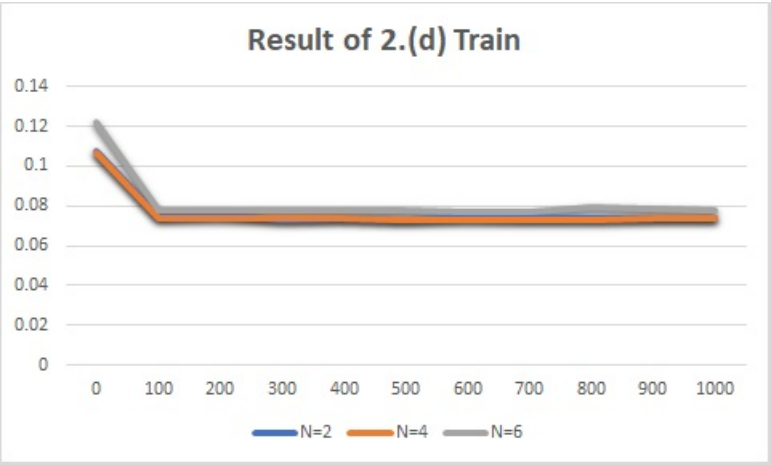
(c) From the data we get from testing result, we have:
Single Tree:

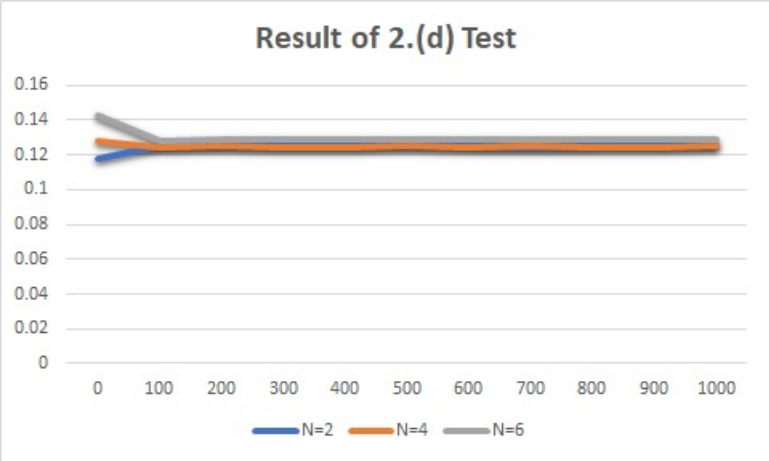
Bias: 0.376
Variance: 0.363
General Error: 0.738

Compare with overall:

Bias: 0.373
Variance: 0.063
General Error: 0.422

(d) We compare the data of both train and test in different size of the subset (2, 4, 6) and get result:





From the graphs, we can conclude that as the size of subset increases, both train error and test error will increase. Therefore, the performance becomes low.

(e) From the result of data collected, we have
Single Tree:

Bias: 23
Variance: 14
General Error: 39

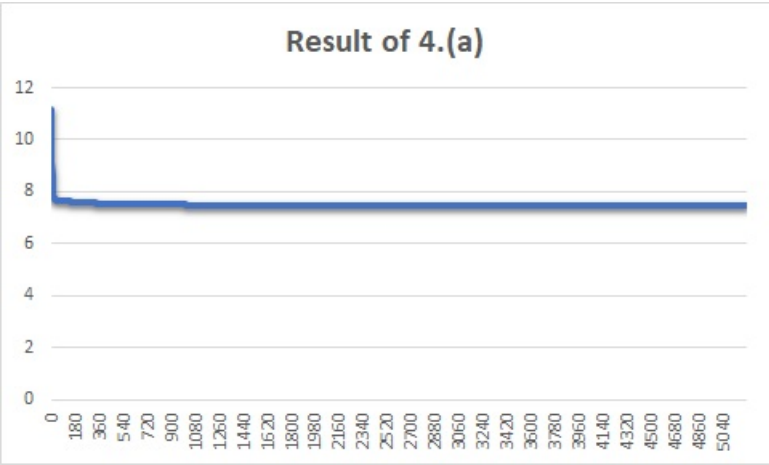
Compare with overall:

Bias: 19
Variance: 0.8
General Error: 20

In conclusion, the random trees have higher bias and variance.

3. Blank

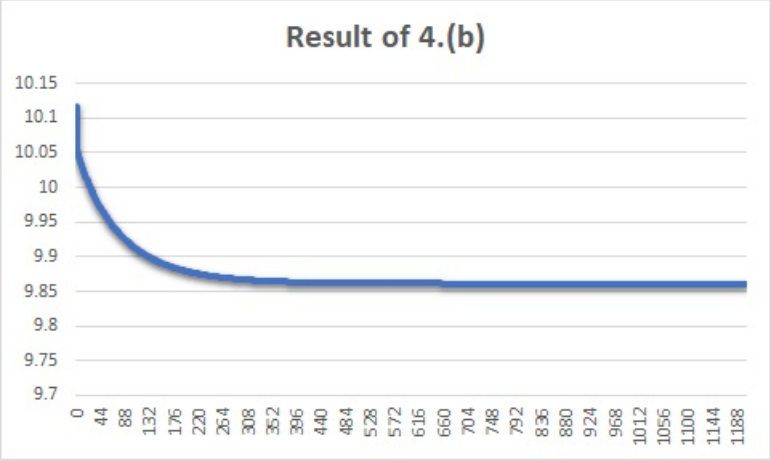
4. (a) **w**: [0.92110784, 0.80783938, 0.87348232, 1.31393419, 0.13385026, 1.5984533, 1.01984135]
Test cost of batch is: 23.360679.
r: 0.015



(b) **w**: [0.50947676, 0.02569702, 0.52711343, 0.84408679, 0.19446188, 0.58406311, 0.46172502]

Test cost of Stoch is: 20.444923.

r: 0.103



(c) AFter the calculation we get:

$\mathbf{w}^* = [0.92148954, 0.80803584, 0.87369845, 1.31403265, 0.13399563, 1.59889654, 1.02002369]$

The data looks very familiar with the weight vector of (a).