COMS W4701 HW6

Dawei He (dh3027)

May 4, 2022

Problem 1

(a)

$P(y=0)=0.5$, $P(y=1)=0.5$

The parameters of $P(x_i|y)$ are following:

```
          x1=-1   x1=0   x1=1
array([[[0.  , 0.5 , 0.5 ],   y=0
        [0.25, 0.5 , 0.25]], y=1
         x2=-1   x2=0   x2=1
       [[0.5 , 0.5 , 0.  ],   y=0
        [0.  , 0.5 , 0.5 ]], y=1
         x3=-1   x3=0   x3=1
       [[0.5 , 0.  , 0.5 ],   y=0
        [0.  , 0.25, 0.75]]]) y=1
```

Figure 1: $P(x_i|y)$

$P(x1=-1|y=0)=N(y=0,x1=-1)/N(y=0)=0$
$P(x2=1|y=0)=N(y=0,x2=1)/N(y=0)=0$
$P(x2=-1|y=1)=N(y=1,x2=-1)/N(y=1)=0$
$P(x3=0|y=0)=N(y=0,x3=0)/N(y=0)=0$
$P(x3=-1|y=1)=N(y=1,x3=-1)/N(y=1)=0$

(b)

x1	x2	x3
-1	-1	-1
-1	-1	0
-1	-1	1
-1	0	-1
-1	1	-1
0	1	-1
1	1	-1
0	-1	0
1	-1	0

(c)

```
   y=0      y=1
[[0.0625, 0.0],          sample1
 [0.0625, 0.046875],
 [0.0625, 0.0],
 [0.0625, 0.0],
 [0.0, 0.046875],
 [0.0, 0.03125],
 [0.0625, 0.09375],
 [0.0, 0.046875]]        sample 8
```

Figure 2: $P(Y|x_1, x_2, x_3)$ for each training sample

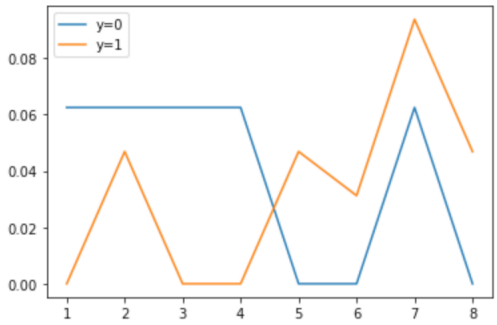


Figure 3: Plot of $P(Y|x_1, x_2, x_3)$ for each training sample

(d)

```
   y=0                  y=1
[[0.0625, 0.004629629629629629],                    sample 1
 [0.0625, 0.05555555555555555],
 [0.0625, 0.020833333333333332],
 [0.0625, 0.027777777777777776],
 [0.020833333333333332, 0.05555555555555555],
 [0.006944444444444444, 0.04166666666666664],
 [0.0625, 0.08333333333333333],
 [0.020833333333333332, 0.05555555555555555]]       sample 8
```

Figure 4: $P(Y|x_1, x_2, x_3)$ for each training sample with laplace smoothing $\alpha = 1$

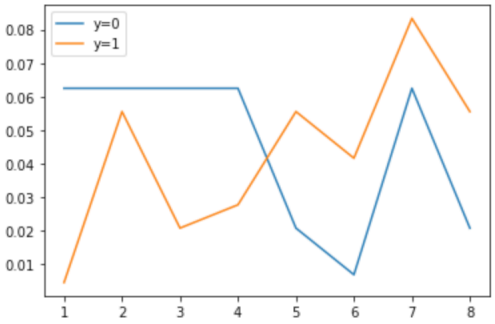


Figure 5: Plot of $P(Y|x_1, x_2, x_3)$ for each training sample with laplace smoothing $\alpha = 1$

There are some samples having probability of 0 without smoothing, but they have probability slightly greater than 0 with smoothing. So if a specific class/feature does not appear in training data, smoothing can make sure the the joint probability will not be zero out by these 0 probability.

(e)

We are uncertain with sample 2 and sample 7. Because they both have small difference of joint probability for $y = 0$ and $y = 1$.

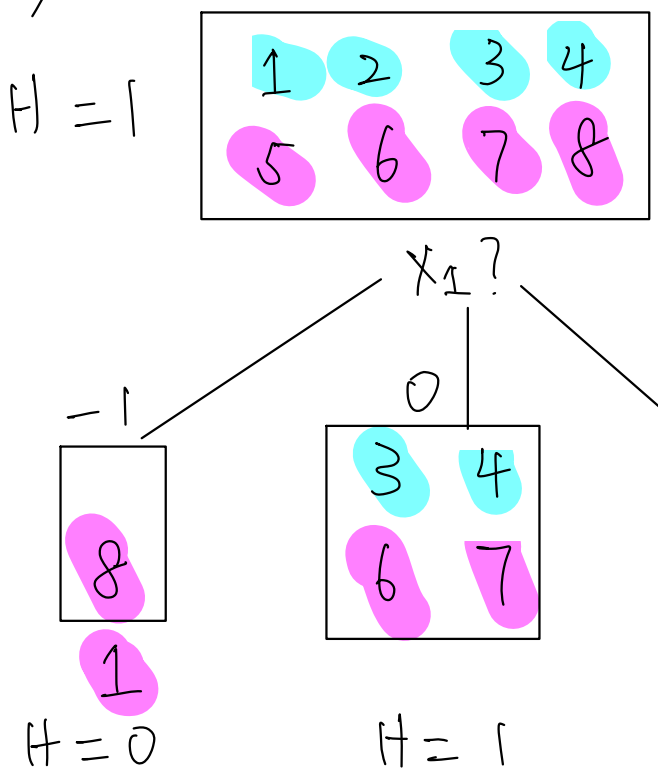
For sample 2, its variable values are (+1,0,+1) and true label is 0. We can see that $x_1 = +1, x_2 = 0$ and $x_3 = +1$ are also frequently appear in the training data whose true label is 1. So it also has a high probability of predicted as label 1.

For sample 7, its variable values are (0,0,+1) and true label is 1. We can see that $x_1 = 0, x_2 = 0$ and $x_3 = +1$ are also frequently appear in the training data whose true label is 0. Also the only difference of sample 2 and sample 7 is the value of x_1, but they have different label. So it also has a high probability of predicted as label 0.

Therefore we are uncertain about these two samples.

problem 2

(a)



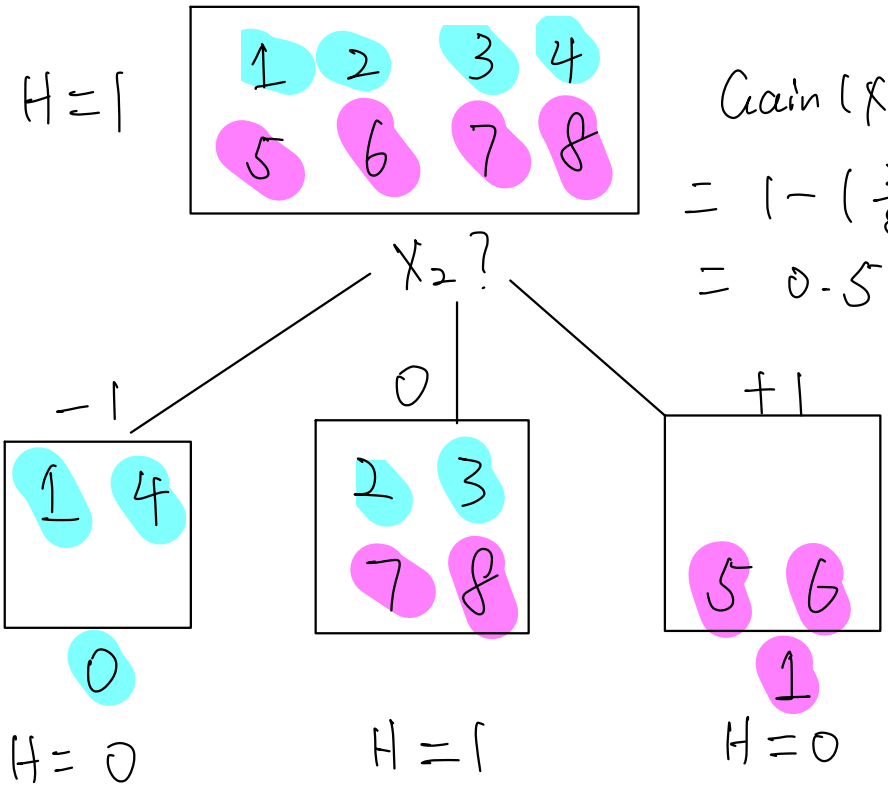
Gain (X_1)

$= 1 - (\frac{1}{8}(0) + \frac{4}{8}(1) + \frac{3}{8}(0.918))$

$= 0.156$

$H = -\frac{2}{3} \cdot \log_2 \frac{2}{3} - \frac{1}{3} \cdot \log_2 \frac{1}{3}$

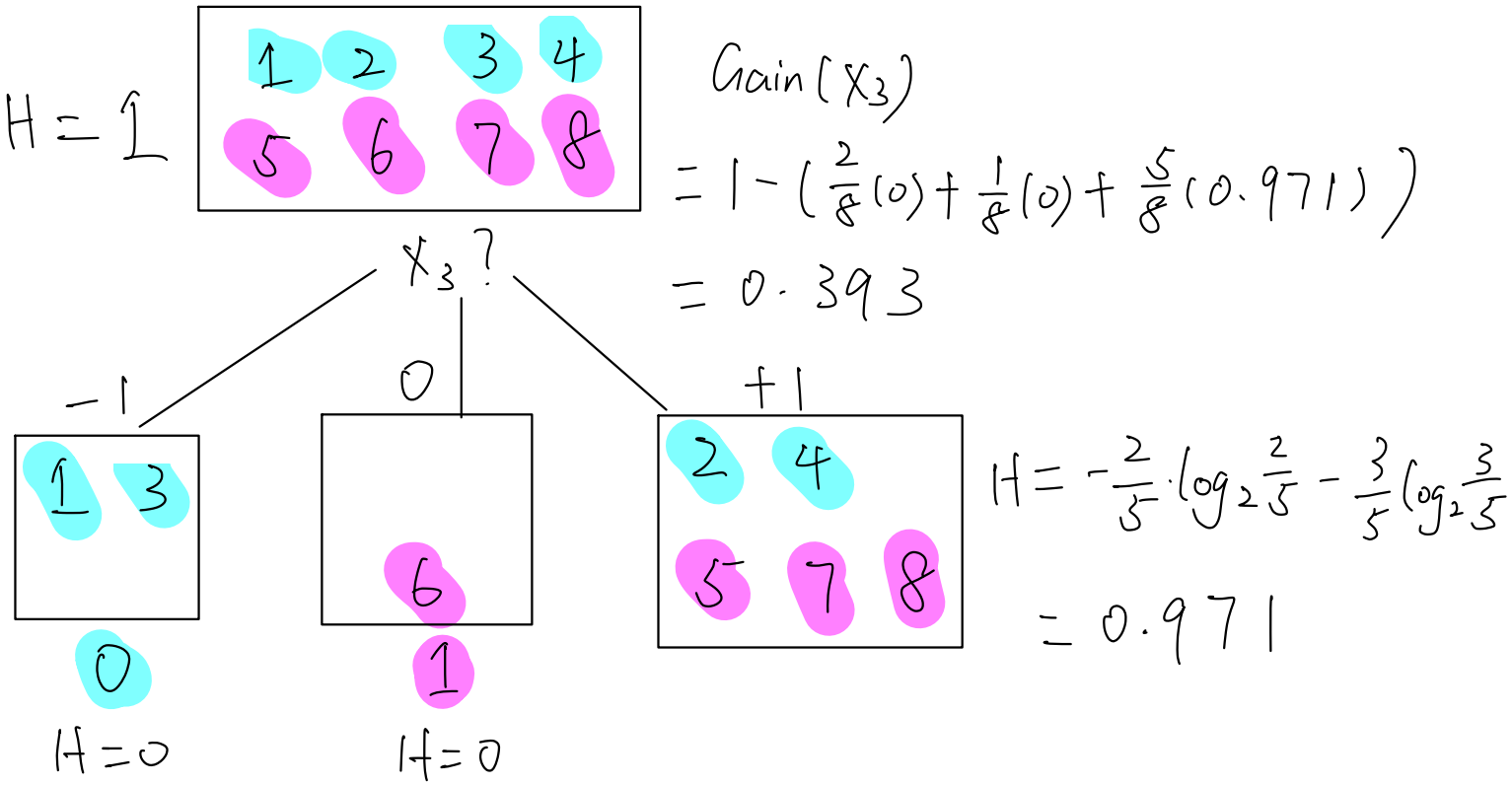
$= 0.918$



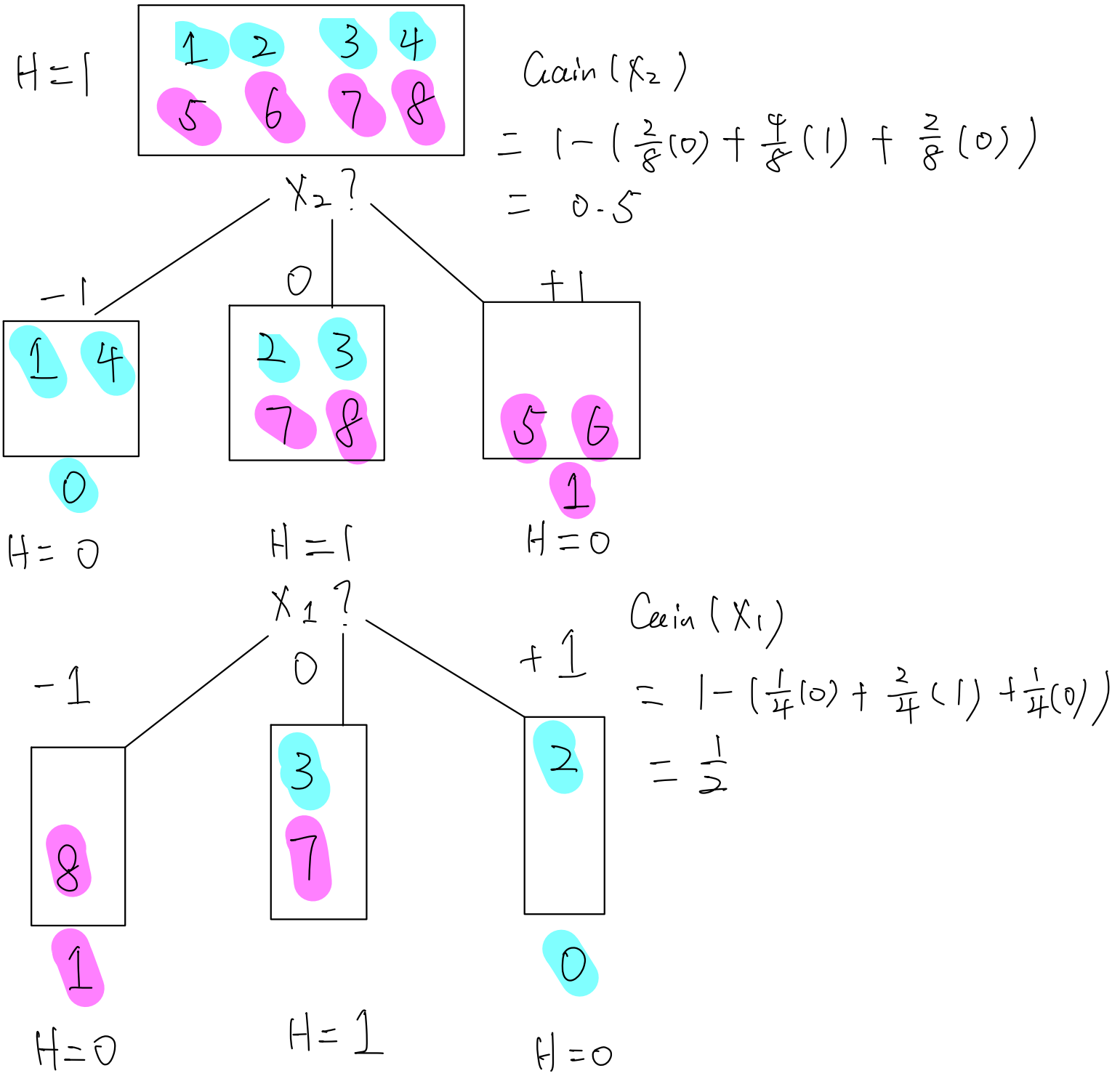
Gain (X_2)

$= 1 - (\frac{2}{8}(0) + \frac{4}{8}(1) + \frac{2}{8}(0))$

$= 0.5$

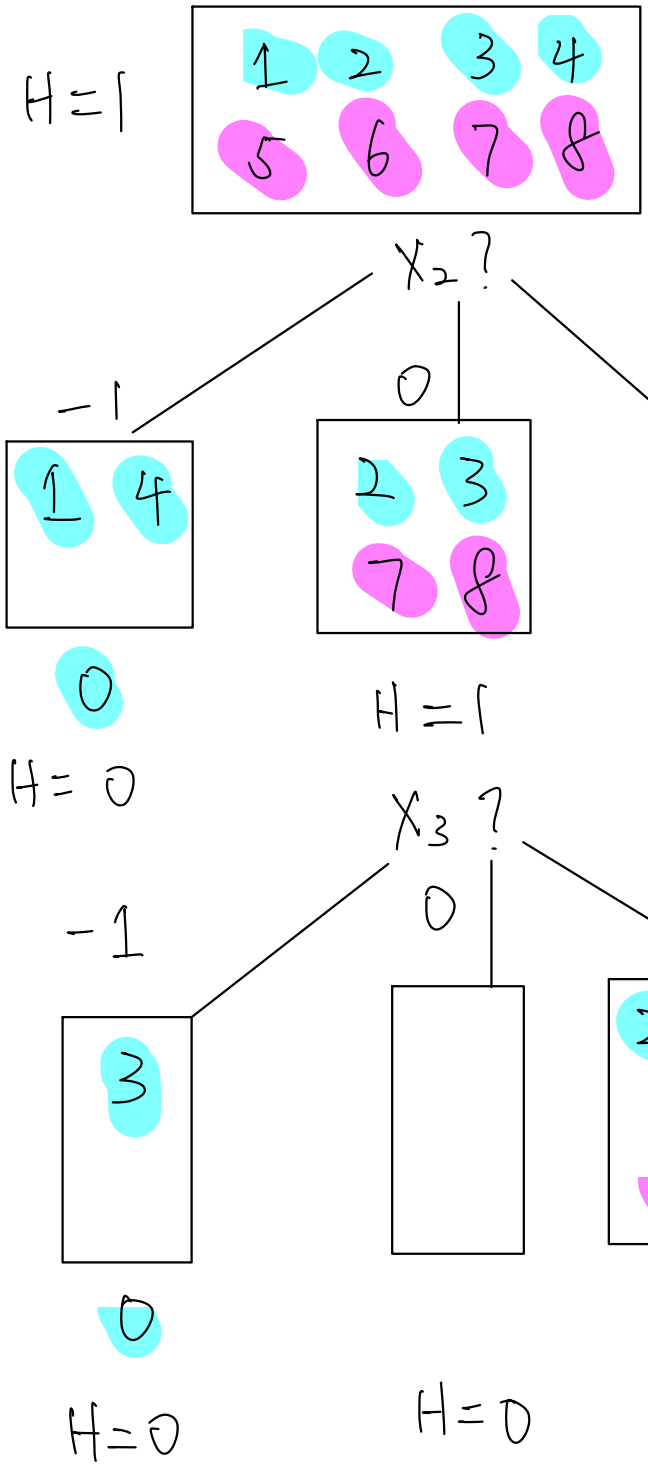


So X_2 has the largest information gain.



$Gain(X_2) = 1 - \left(\frac{2}{8}(0) + \frac{4}{8}(1) + \frac{2}{8}(0)\right) = 0.5$

$Gain(X_1) = 1 - \left(\frac{1}{4}(0) + \frac{2}{4}(1) + \frac{1}{4}(0)\right) = \frac{1}{2}$



Gain (X_2)

$= 1 - \left(\frac{2}{8}(0) + \frac{4}{8}(1) + \frac{2}{8}(0)\right)$

$= 0.5$

Gain (X_3)

$= 1 - \left(\frac{1}{4}(0) + \frac{0}{4}(0) + \frac{3}{4}(0.918)\right)$

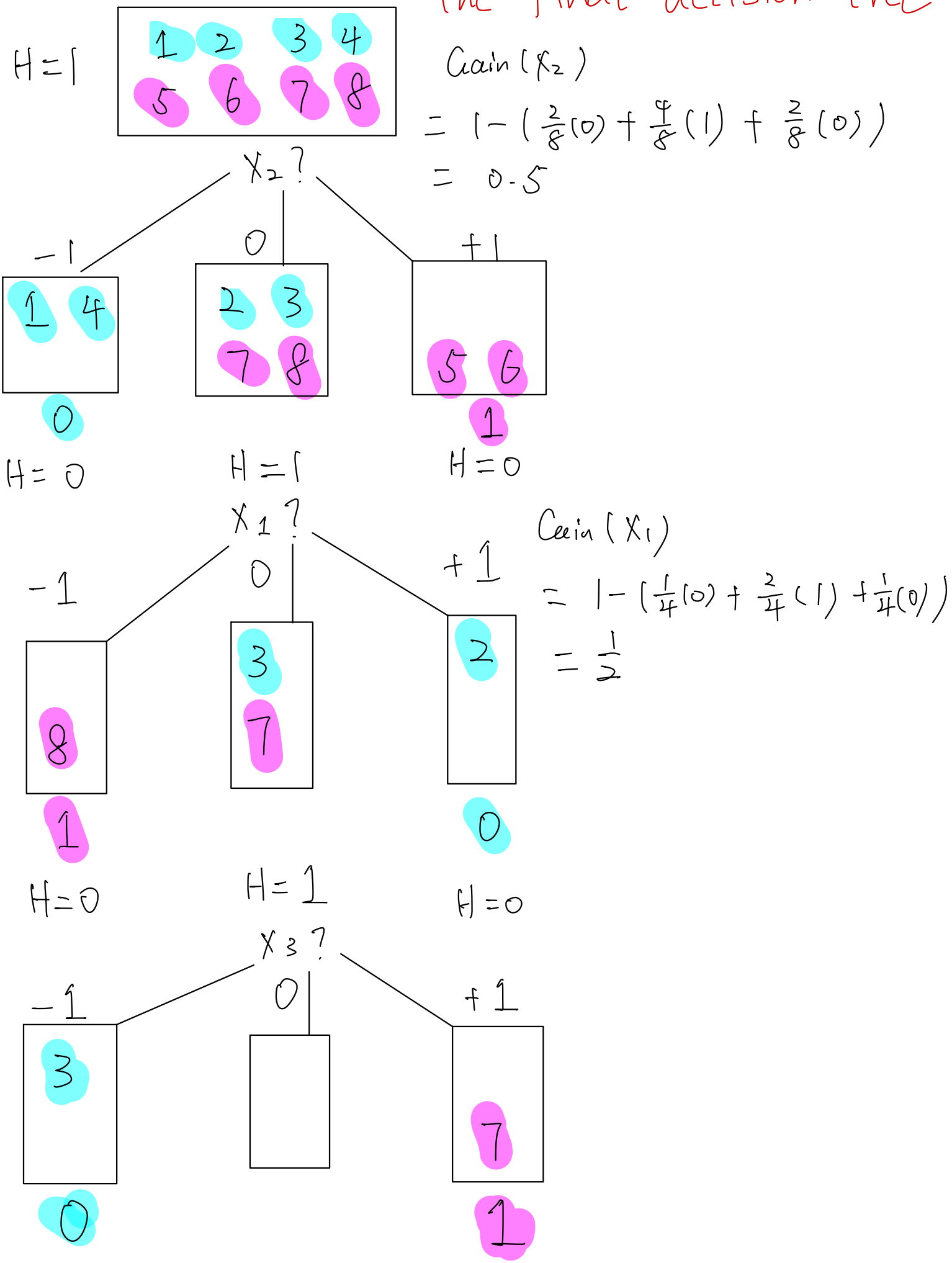
$= 0.312$

$H = -\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3}$

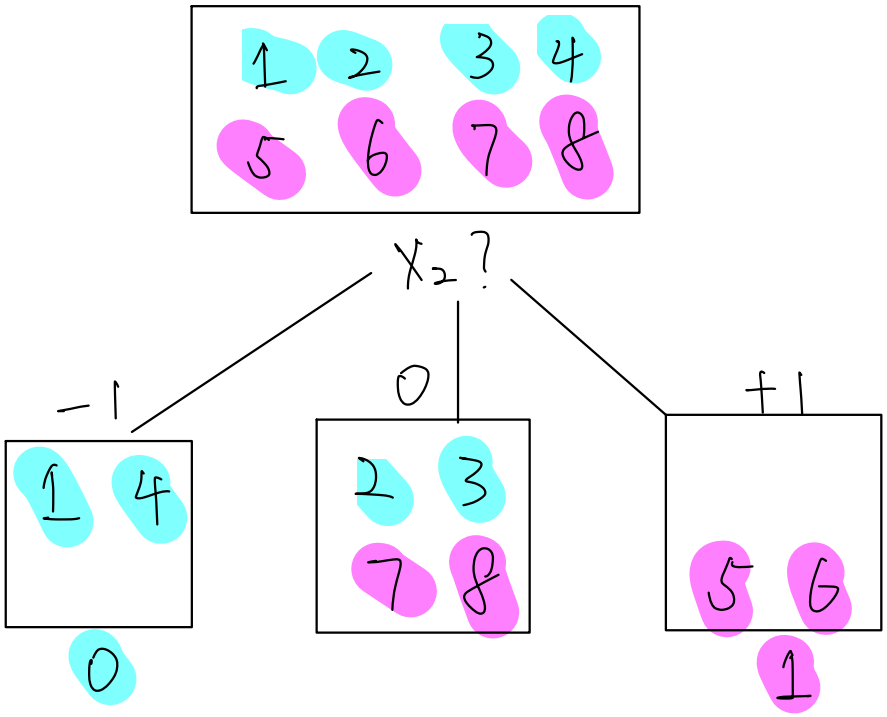
$= 0.918$

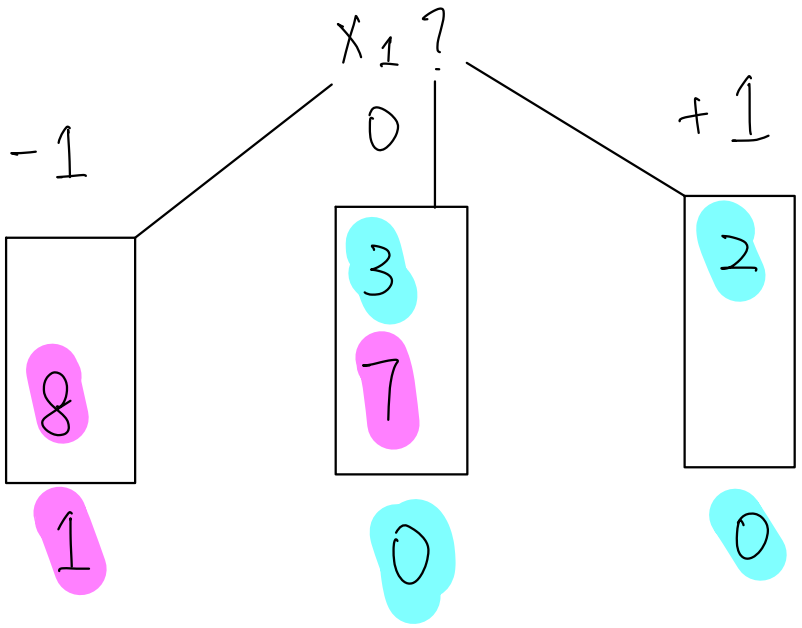
So X_1 has the largest information gain

The final decision tree

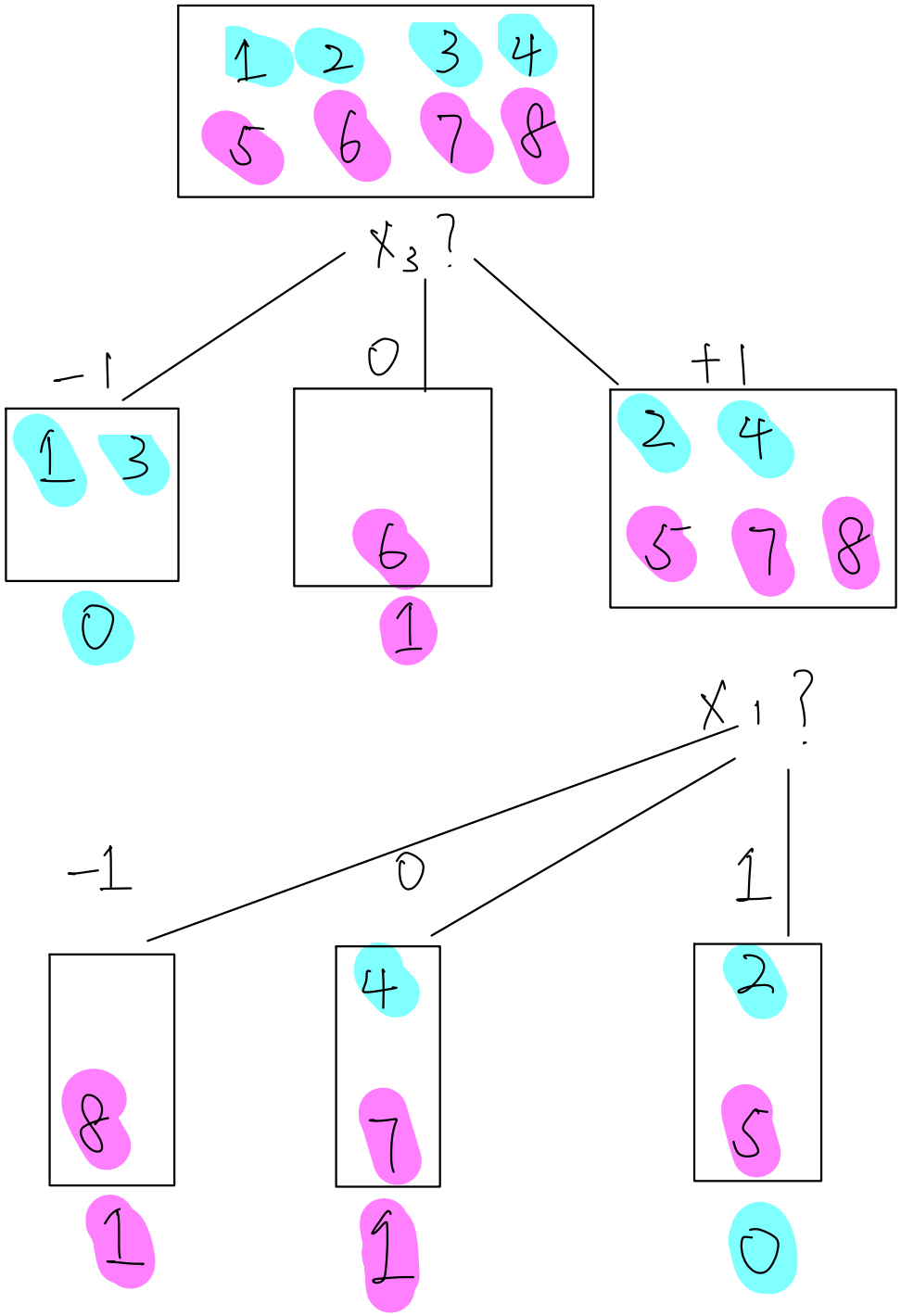


(b)

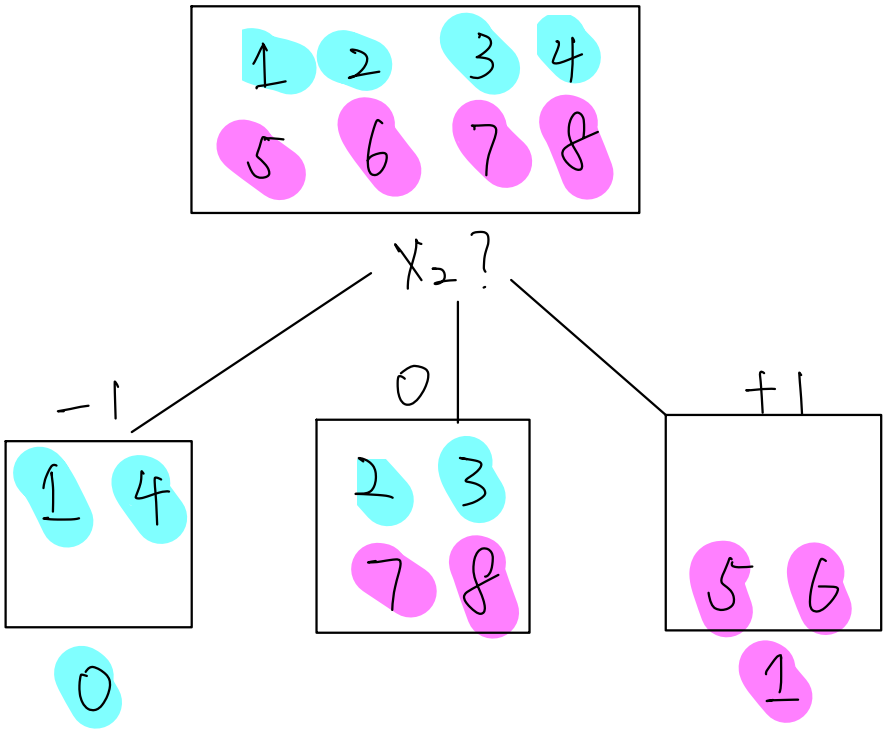


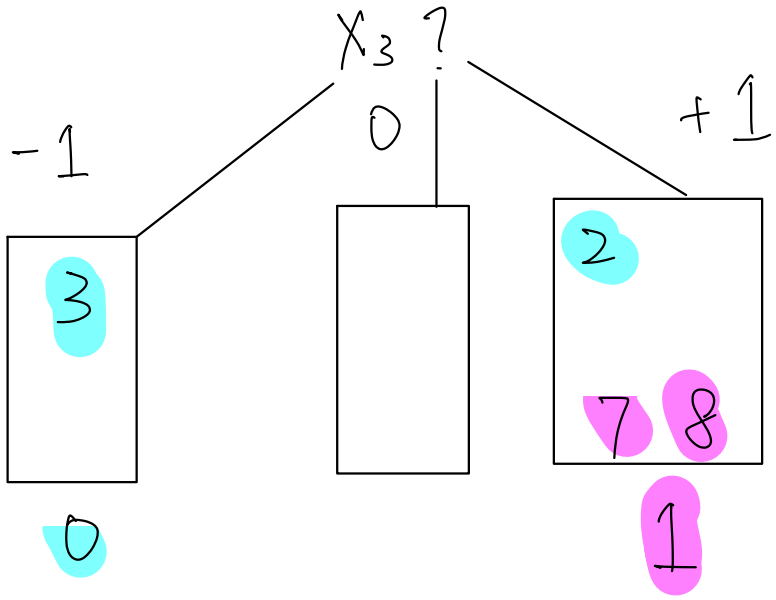


Training accuracy $\frac{7}{8}$



Training accuracy : $\frac{6}{8}$





Training accuracy: $\frac{7}{8}$

Sample	Tree(X_1, X_2)	Tree(X_1, X_3)	Tree(X_2, X_3)	Vote
1	0	0	0	0
2	0	0	1	0
3	0	0	0	0
4	0	1	0	0
5	1	0	1	1
6	1	1	1	1
7	0	1	1	1
8	1	1	1	1

Hightest training accuracy of the forest using majority vote is: 1

Problem 3

(a)

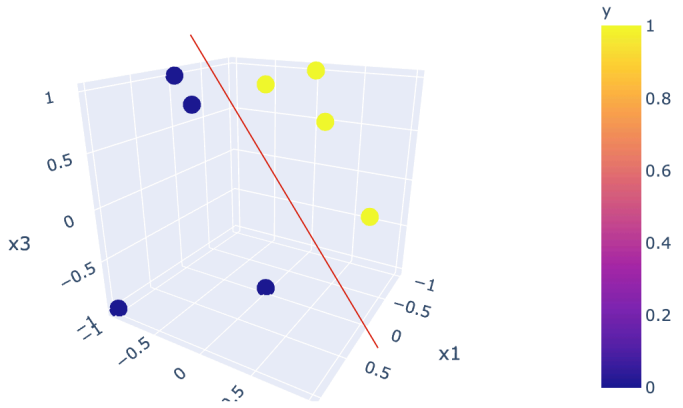
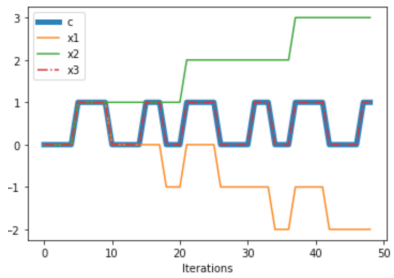


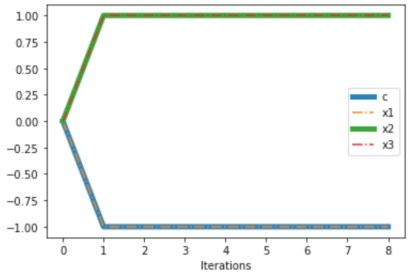
Figure 6: Plot of the data as points in x1-x2-x3 space

Clearly the data is linearly separable

(b)

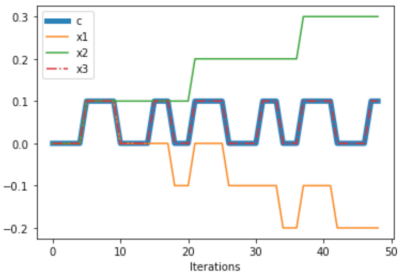


(a) Model 1, tiebreaker hw = 0 predicts y = 0

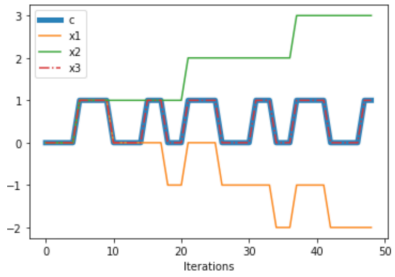


(b) Model 2, tiebreaker hw = 0 predicts y = 1

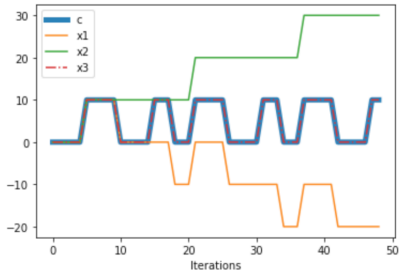
(c)



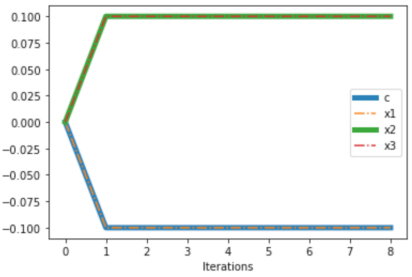
(a) Model 1, $\alpha = 0.1$



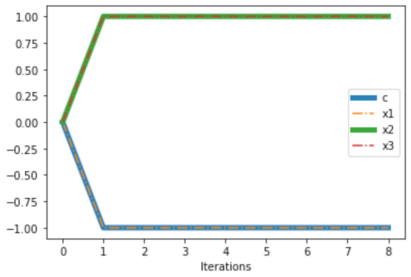
(b) Model 1, $\alpha = 1$



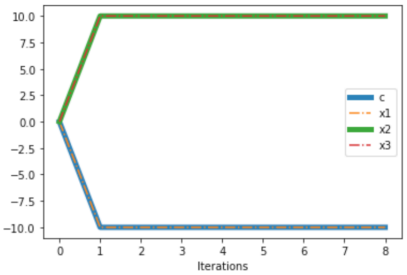
(c) Model 1, $\alpha = 10$



(a) Model 1, $\alpha = 0.1$



(b) Model 1, $\alpha = 1$



(c) Model 1, $\alpha = 10$

We can clearly see that for both models, the learning rate only act as a scaler of final learned weights, it will not affect the converge speed.

(d)

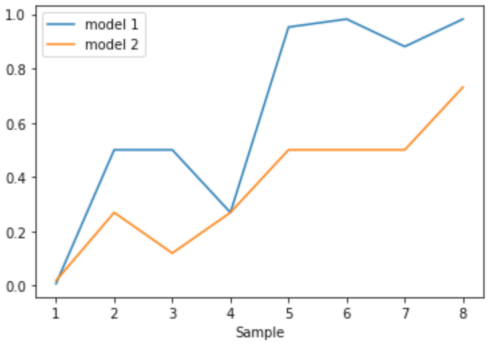


Figure 10: Plot of sigmoid values of each of the training data using the two learned models

```
[0.0066928509242848554,
 0.5,
 0.5,
 0.2689414213699951,
 0.9525741268224334,
 0.9820137900379085,
 0.8807970779778823,
 0.9820137900379085]
```

(a) Results of model 1

```
[0.01798620996209156,
 0.2689414213699951,
 0.11920292202211755,
 0.2689414213699951,
 0.5,
 0.5,
 0.5,
 0.7310585786300049]
```

(b) Results of model 2

For model 1, we are uncertain about sample 2 and 3, whose associated probabilities are both 0.5 in model 1. For model 2, we are uncertain about sample 5, 6 and 7, whose associated probabilities are all 0.5 in model 2.

Problem 4

(a)

$$\text{hidden layer output 1} = g_1(x_1 w_{11}^{(1)} + x_2 w_{21}^{(1)} + x_3 w_{31}^{(1)} + w_{01}^{(1)})$$

$$\text{hidden layer output 2} = g_2(x_1 w_{12}^{(1)} + x_2 w_{22}^{(1)} + x_3 w_{32}^{(1)} + w_{02}^{(1)})$$

$$\text{output layer output} = g_3((\text{hidden layer output 1}) w_{11}^{(2)} + (\text{hidden layer output 2}) w_{21}^{(2)} + w_{01}^{(2)})$$

(b)

$$\frac{\partial L}{\partial w_{01}^{(2)}} = (\hat{y} - y) \cdot g_3'$$

$$\frac{\partial L}{\partial w_{11}^{(2)}} = (\hat{y} - y) \cdot g_3' \cdot (\text{hidden layer output 1})$$

$$\frac{\partial L}{\partial w_{01}^{(1)}} = (\hat{y} - y) \cdot g_3' \cdot w_{11}^{(2)} \cdot g_1'$$

$$\frac{\partial L}{\partial w_{11}^{(1)}} = (\hat{y} - y) \cdot g_3' \cdot w_{11}^{(2)} \cdot g_1' \cdot x_1$$

(c)

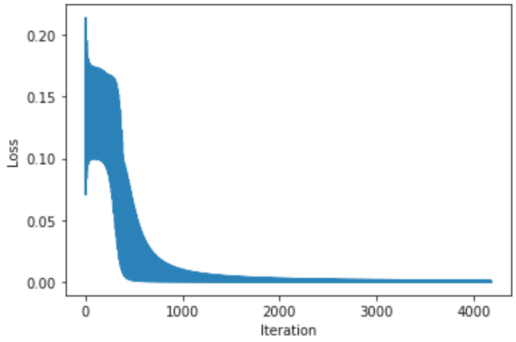
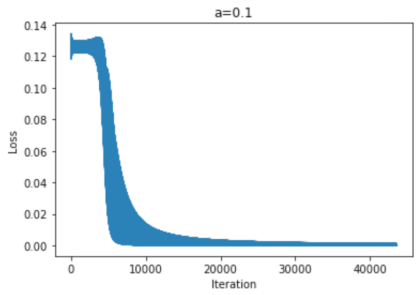
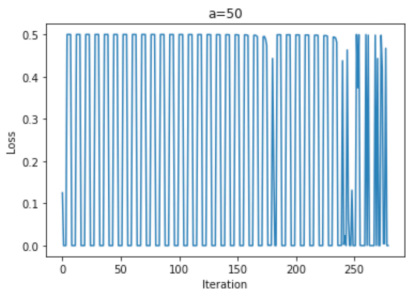


Figure 12: Plot of loss over each iteration, with threshold $loss < 0.001$

(d)



(a) Loss plot with $\alpha = 0.1$



(b) Loss plot with $\alpha = 50$

If learning rate α is small, it will takes a lot of iterations to converge. However if learning rate α is large, it will takes fewer iteration to converge. But large learning rate will often overshot the loss function, so it will have zig zag shape in the loss plot.

```
(array([[-0.0106906 ,  3.05963708, -3.47008551, -1.79713168],
        [-0.0106906 ,  3.05963708, -3.47008551, -1.79713168]]),
 array([ 4.46406161, -4.97669244, -4.97669244]))
```

(a) Learned wights with $\alpha = 0.1$. Top is weights of hidden layer, bottom is weights of output layer

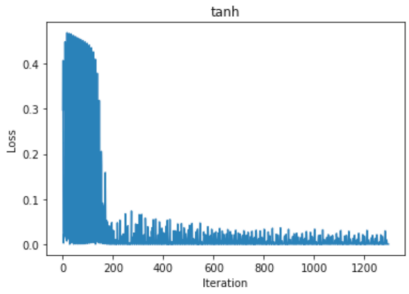
```
(array([[-2.69542992, -9.18086117, 10.13714819,  3.62093425],
        [-2.69542992, -9.18086117, 10.13714819,  3.62093425]]),
 array([-4.25017621,  7.32170114,  7.32170114]))
```

(b) Learned wights with $\alpha = 50$. Top is weights of hidden layer, bottom is weights of output layer

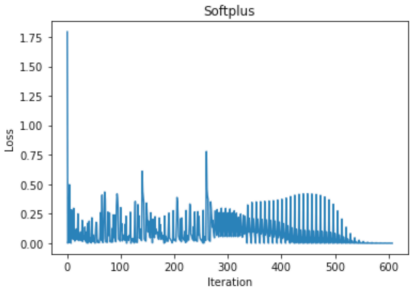
With different learning rate, the final learned wights are totally different. But the difference will not be extremely large.

(e)

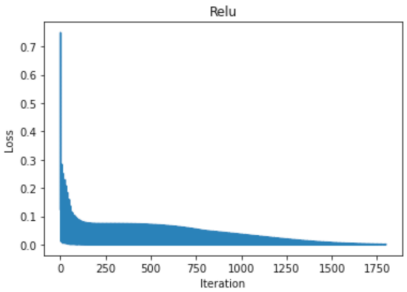
All threshold loss is 0.001 for three activations.



(a) tanh, $\alpha = 1$



(b) softplus, $\alpha = 1$



(c) Relu, $\alpha = 0.1$

We can see that they all have large loss at the beginning and then their loss gradually decrease with iteration. Softplus has fastest convergence rate among them, it takes fewer iterations to get to the loss threshold. For relu activation, it has to use smaller learning rate compare to others, otherwise it will always overshoot and will not converge. There is still obvious fluctuation at the end iterations for tanh activation, which means tanh may not be a good activation function comparing to the other two functions.