Statistical Theories for Brain and Parallel Computing

Assignment No.1

Year	B4
Student Number	14B06342
Name	Konno Takashi （紺野　隆志）

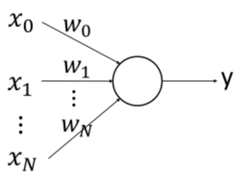
Assignment No.1

Compute the output of the probabilistic and binary for 1,000 times … 10,000 times.
And check if it follows the probability specified by the by the inputs.

1. Describe the methods equation and principle. (What you study in the course.)

Probabilistic and Binary Model

Probabilistic: Even for the same inputs, the output can take different values.
This model is visualized as shown in the Fig. 1.



$$x_0 = 1\ (Dummy\ neuron)$$
$$w_0 = -\theta\ (Threshold)$$

Fig.1 Probabilistic and Binary Model

Weighted sum is defined by equation (1). Probability is defined by equation (2).

$$S = \sum_{n=0}^{N} w_n x_n \tag{1}$$

$$p = \mathrm{sigmoid}(S) \tag{2}$$

sigmoid(S) is a sigmoid function with gain α and defined by equation (3). A sigmoid function is a mathematical function having a characteristic "S"-shaped curve or sigmoid curve like Fig.1.

$$\varsigma_\alpha(S) = \frac{1}{1+e^{-\alpha S}} \tag{3}$$

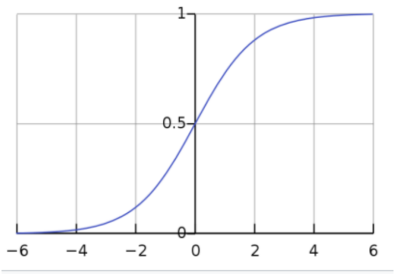


Fig.2 Sigmoid function

Output y is defined by equation (4).

$$y = \begin{cases} 0 & if\ rand > RAND\_MAX * sigmoid(S) \\ 1 & else \end{cases} \tag{4}$$

2. Explain the program by words, equation and figures. Attach the source code as separate files.

I wrote the program in MATLAB.

Firstly, I made the following function in probabilistic_binary.m

function [y, expected, experiment] = probabilistic_binary(x, w, a, iterate)

The argument of this function are 'x' which is the inputs of this model and 'w' which is the weights, 'a' which is gain of the sigmoid function, 'iterate' which is the iteration of the computing. The return value are 'y' which is the outputs of this model and 'expected' which is the ideal number for y taking 1. 'experiment' which is the experimental result.

This function compute the equation (1) ~ (3) and repeat computing the equation (4) by using 'rand' function which returns a single uniformly distributed random number in the interval (0,1) 'iterate' times and count the number for y taking 1. I write the Pseudocode below.

```
function probabilistic_binary
  Input: x, w, a, iterate
  Output: y, expected, experiment
 compute the equation (1) ~ (3)
 set iterate × 1 y matrix to 0
 for i from 1 to iterate do
   if rand ≤ rand_max(1) ∗ p then
     y[i] = 1
   end if
 end for
 expected is iteration ∗ probability
experiment is the number for y taking 1
end function
```

Secondly, I set the inputs x = [1 0 -1 0 1] and the weights w = [1 2 1.2 1.5 -0.5] which are five elements, then use the function probabilistic_binary. Finally, I changed the gain α of sigmoid function from 0 to 10 every 0.05 and the iteration from 1,000 to 10,000 every 500. And I record the output of function probabilistic_binary in Excel. I write the Pseudocode below.

```
set inputs x to  [1 0 -1 0 1 and weights w to [1 2 1.2 1.5 -0.5]
for iterate from 1,000 to 10,000 every 500 do
 for gain α from 0 to 10 every 0.05 do
```

4:     function probabilistic_binary(x, w, a, iterate)
5:   end for
6: end for
7: record output of function probabilistic_binary with the gain and the iterate in Excel

3. Show the results of computation for different conditions. (Change values for parameters. Apply the program to different inputs)

In this program, weighted sum was calculated $S = -0.70$.

Firstly, I show the Table. 1 of the expected number for y taking 1 and experimental result I counted with different gain from 0 to 10 every 0.5 and 1,000 iterations.

Table. 1 Comparison of Expected and Experiment with 1,000 iterations

gain	Expected	Experiment
0	500	470
0.5	413	407
1	332	344
1.5	259	268
2	198	194
2.5	148	161
3	109	108
3.5	79	82
4	57	60
4.5	41	48
5	29	28
5.5	21	23
6	15	13
6.5	10	7
7	7	16
7.5	5	2
8	4	9
8.5	3	5
9	2	1
9.5	1	0
10	1	0

Secondly, I show the graph of above date, Fig. 3 is expected and Fig. 4 is experiment, Fig. 5 is Comparison of Expected and Experiment

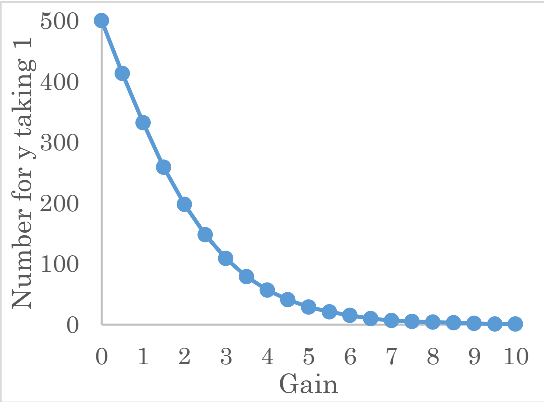


Fig. 3 Expected

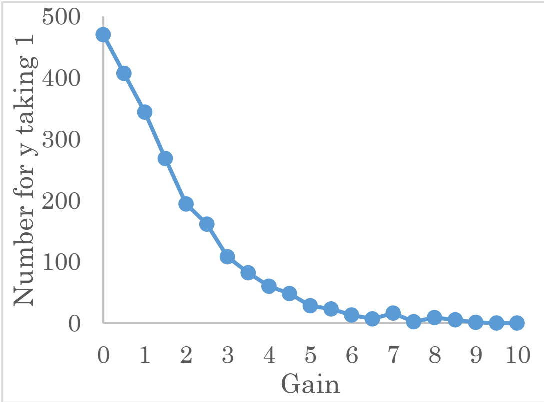


Fig. 4 Experiment

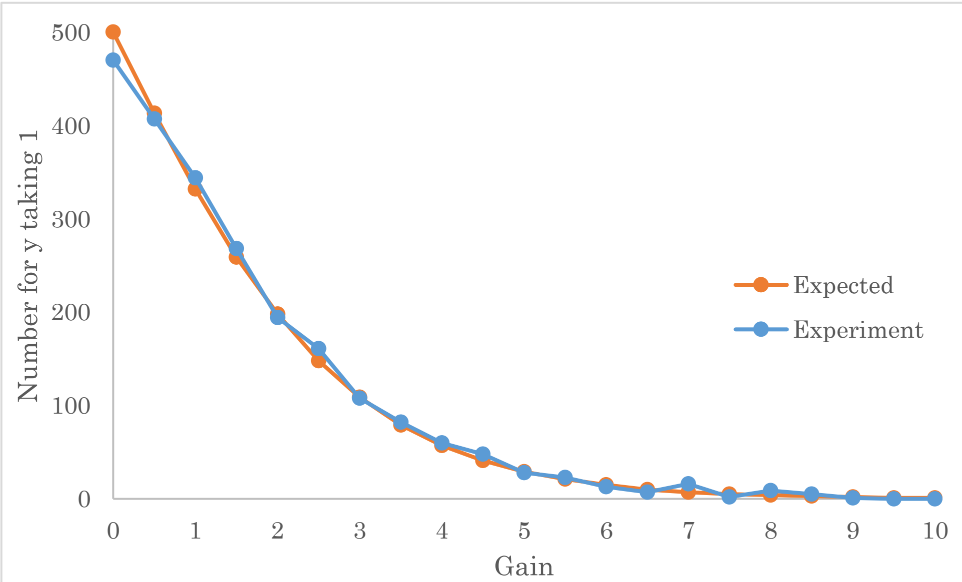


Fig. 5 Comparison of Expected and Experiment

Finally, I calculated the expected/iteration and experiment/iteration so I could compare them with different iterations. Then I calculated the relative error between the expected/iteration and experiment/iteration with all gains and the average at each iteration. This results is shown on Fig. 6.

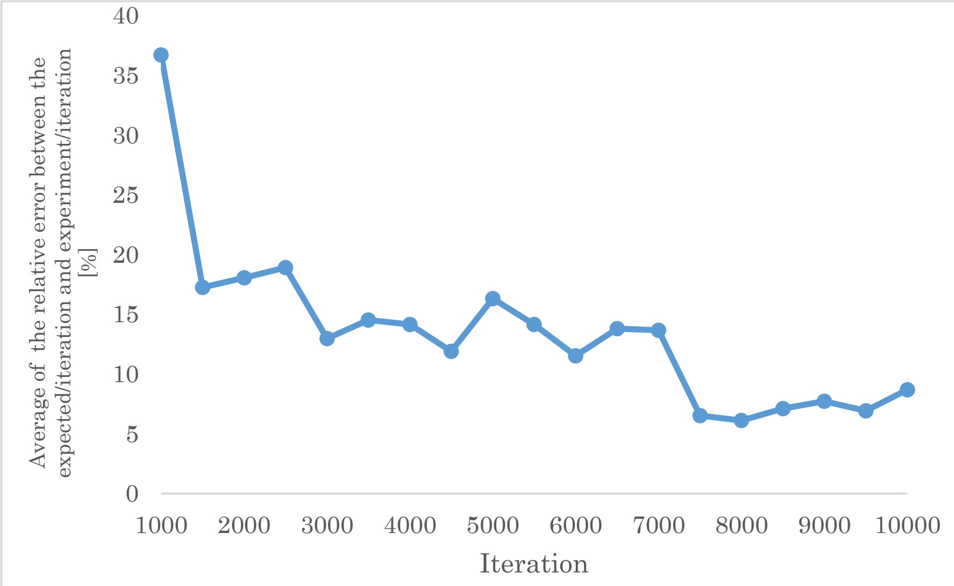


Fig. 6 Average of the relative error each iteration

4. Discussion on the results.

Firstly, I discuss Table. 1 and Fig. 3 – 5. From Table. 1 and Fig. 5, expected and experimental result which the number for y taking 1 are almost similar, so I can confirm the experimental result follows the probability specified by the inputs. But since experimental result is random variable, these variable are a little off the expected value. And if gain α is close to 0, the number for taking 1 is almost dependent on probability, so close to 500 which is expected value of probability. In contrast if gain α is large, this model get close to Deterministic model and in this program weighted sum S is negative value, so the number for y taking 1 is close to 0.

Secondly, I discuss Fig. 6. If iteration increase, the average of the relative error between the expected/iteration and the experiment/iteration almost decreased. And if iteration close to infinity, this error close to 0.