CHAPTER 4
ANALYSIS AND DESIGN

4.1 Analysis

In this project, the problem of the company is too much produced the stock of rice so some of rice stock can be wasted. If they can cope with this problem, not only can solve the problem of overproduction, but can get a better profit. This program also can predict how much stock is needed on the day.

Table 4.1: Tabel Analisis Data

Date	Sales Price	Temperature	Total Sales	Stock
01/04/2019	11800	35	15784	19680
02/04/2019	12000	38	24507	26110
03/04/2019	11500	36	24233	24380
04/04/2019	12000	36	34196	35500
05/04/2019	11500	33	17559	23040
06/04/2019	11800	36	15230	20400
07/04/2019	12000	33	26920	28660
08/04/2019	11000	33	14179	20370
09/04/2019	11500	36	4217	7920
10/04/2019	11500	34	13740	13980
11/04/2019	11000	30	31576	32750
...	...	...	...	...
24/03/2020	12000	37	13198	17820
25/03/2020	12000	31	19013	24840
26/03/2020	11000	30	20496	21720
27/03/2020	12000	32	17633	18170
28/03/2020	11500	38	16197	17230
29/03/2020	12000	38	12121	13580
30/03/2020	11000	35	3237	3710

The data in table 4.1 will be normalized so it can be calculated in the backpropagation algorithm to form 0-1 using min-max methods.

ND = (D – minD) / (maxD – minD)

ND = Normalized Data

D = Data

minD = Minimum Data Value

maxD = Maximum Data Value

After normalization, the data has been changed in the form of a range between 0 to 1. The next step is to enter the data into the backpropagation algorithm. the backpropagation algorithm process including :

1. Generate random value in range 0 – 1 for all weight

   W_{ia} = Weight for input index i to hidden layers index a

   W_{0a} = Weight for bias to hidden layers index a

   O_a = Weight for hidden layer index a to output

   O_0 = Weight for bias hidden layer to output

2. Calculate the neural activation for hidden layer

   $H\_Activation = W_{0a} + \sum D_i W_{ia}$

   $H\_TransferActivation = 1 / (1 + e^{-H\_Activation})$

   H_Activation = value of neural activation for hidden layer before transfered

   H_TransferActivation = value of neural activation for hidden layer

3. Calculate the neural activation for output layer

   O_Activation $= W_{0a} + \sum D_i W_{ia}$

   Output $= 1 / (1 + e^{-O\_Activation})$

   O_Activation = value of neural activation for hidden layer before transfered

   Output = value of neural activation for hidden layer

4. Calculate Δ weight for input to hidden layers

   Δ_1 = (expected – Output) * Output *

   (1 – Output)

   ΔO_a = lrate * $\Delta_{1\,k}$ * H_TransferActivation

   expected = output expectation

   ΔO_a = delta weight hidden layer index a to output index k

5. Calculate Δ weight for input to hidden layers

   Δ_net = $\Delta_{1\,k}$ * O_a

   Δ_2 = Δ_net * H_TransferActivation * (1 – H_TransferActivation)

   Δw_{ia} = lrate * Δ_2 * X_i

   X_i = Weight Input index i

   Δw_{ia} = delta weight input index i to hidden layer index j

6. Update Weight

   $O_a = O_a + \Delta O_a$

   $w_{ia} = w_{ia} + \Delta w_{ia}$

7. Repeat Step 2 – 6 Until Epoch ends or error < target error

   Error = (Expected – Output)^2

4.2 Desain

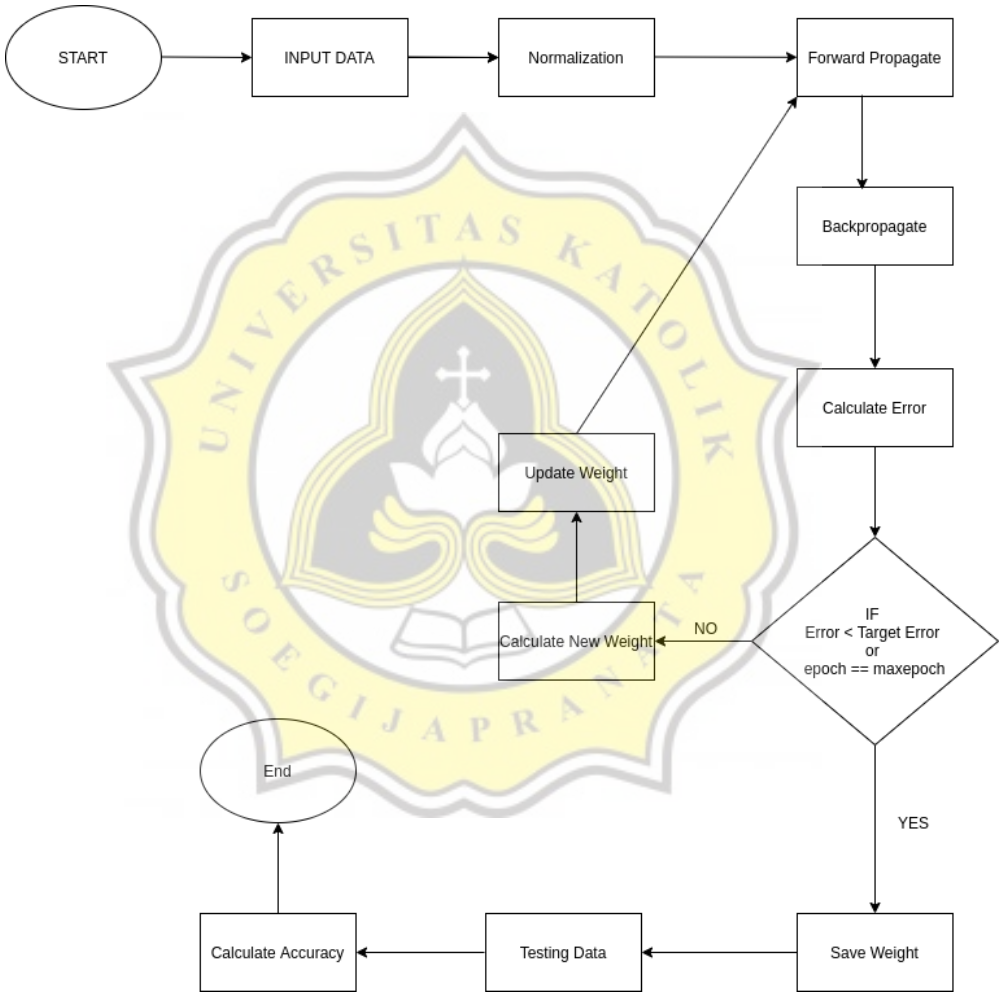

Illustration 4.1: Flowchart program

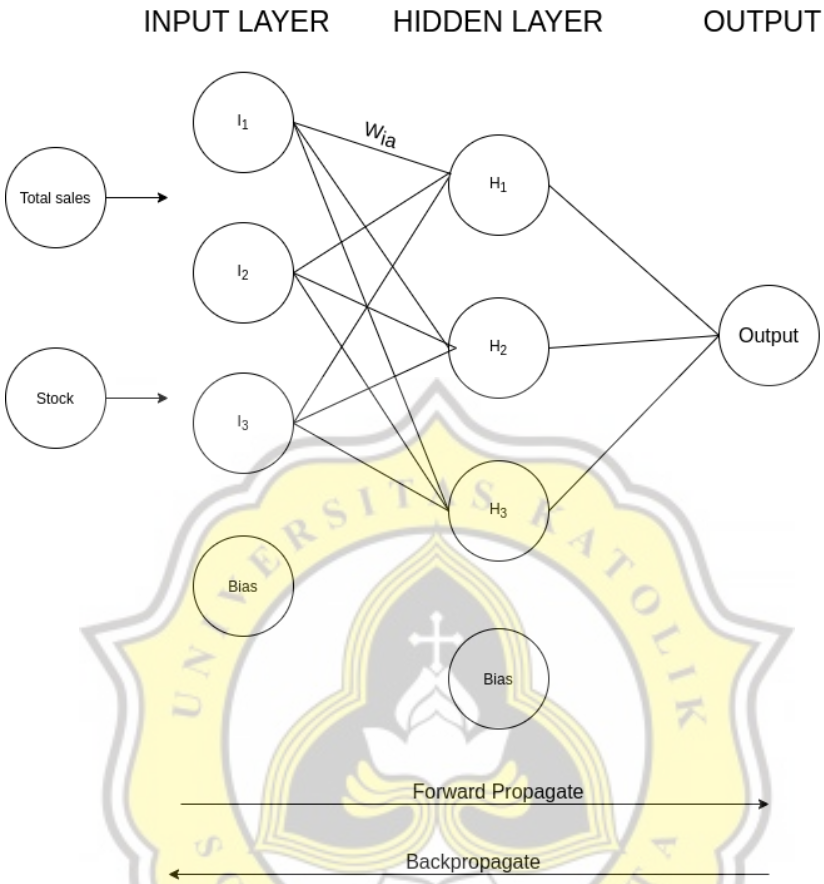

Illustration 4.2: Algorithm Backpropagation