CHAPTER 5
IMPLEMENTATION AND TESTING

5.1 Implementation

The implemantation of this program start from reading the data in form of CSV and processed in a Java Platform. The first thing the program do after reading the data from CSV file is to give a random weight to the first data of all variable (age, gender, body mass index, smoker) and hidden layers. The random weight used in this research is in range of -0.5 to 0.5.

The algorithm begin with a process called feedforward after the weight has been randomized. In this process, the goal is to calculate the output and compare the output to the target error user have set. The output can't be calculated because the hidden layers still doesn't have a value. In order to give a value for the hidden layer, first the input value and the bias input need to be multiplied with their weight with a formula, after obtaining the hidden layers value the output can be calculated using the same formula to calculate the hidden layers.

The next step begin when the output has been calculated. The next step is to calculate the loss. The loss can be calculated by using the formula to square the difference between output and targeted value (Math.pow(output-targeted_value)). The current weight can be called the best weight if the loss is smaller than target error or the max iteration has been reached, if the loss is bigger than target error then the iteration continue and the weight will be calculated again using a formula that involve learning rate of the AI.

The new weight is calculated using a formula with a learning. This learning rate determine how fast the neural network changes its mind. The calculation of the new weight begin with calculating the error rate and then calculate the differences weight using the error rate and add the difference weight to the real weight to update it for new weight and input it to the ArrayList, after

the ArrayList for the new hidden layer weight and new variable weight have been calculated replace the current weight with the new one. This step will be repeated with the feedforward until the output is smaller than target loss or the iteration have reached its limit.

The testing process in the program is using only the feedforward process. After the output has been calculated, the loss is also calculated. The loss is needed to calculate the error rate (RMSE).The formula to calculate the error rate is square of (sum of all loss divided size of data). The test can be called a success if the error rate is smaller than the targeted error rate.

5.2 Testing

The test is done with a certain hidden layers, max iteration, learning rate, target error inputed by user and 250 different data from training. Below is result of some testing that used 2000 iteration, 0.9 learning rate, 0.05 target error and inputted hidden layers.

Table 5.1: Error (RMSE) of testing using different hidden layers in percentage

Test Number	Error Rate (percentage)				
	hidden layer 1	hidden layer 2	hidden layer 3	hidden layer 4	hidden layer 5
1	14.62	4.56	6.07	4.21	4.05
2	9.39	10.67	6.29	5.82	3.03
3	15.41	5.57	5.36	3.65	4.71
4	3.83	10.16	5.67	3.47	3.46
5	15.14	4.87	4.98	4.16	4.86
6	7.14	4.77	6.04	4.94	4.62
7	15.69	7.32	4.6	2.7	3.09
8	9.83	7.8	5.68	4.74	5.87
9	10.44	5.71	7.44	4.8	4.87
10	14.69	3.05	7.84	6.9	3.94
11	13.62	7.56	5.97	4.69	4.61
12	17.4	6.13	4.12	4.95	4.35
13	6.56	6.66	5.05	4.65	3.92
14	13.82	4.1	6.03	3.07	4.63
15	11.9	3.84	6.01	2.28	5.26

16	12.12	8.68	4.73	4.39	4.04
17	17.65	11.97	5.21	3.7	4.22
18	9.6	8.76	6.19	2.69	3.96
19	6.45	9.07	7.15	3.77	4.32
20	17.57	8.81	5.89	3.97	3.27

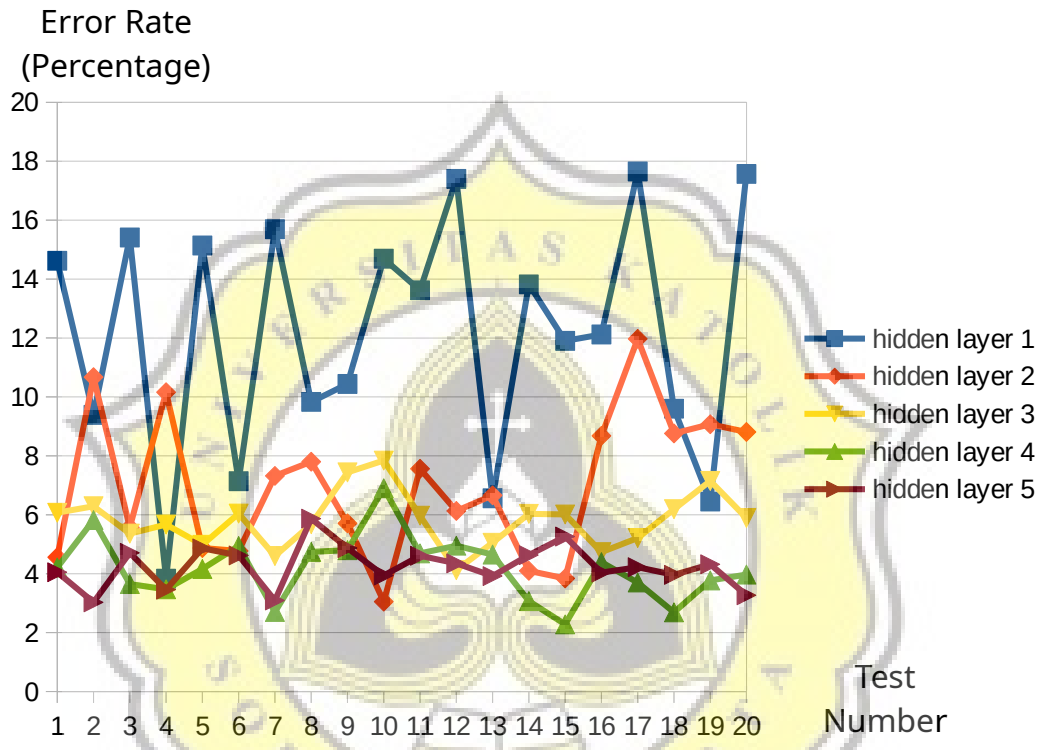

Illustration 5.1: Error rate graph for 5 different hidden layer

Can be seen from table and graph above using more hidden layer will have more accurate result. The test with 1 hidden layer had 19 RMSE that exceed the target error (5%) and the graph is not stable. The test with 4 hidden layer had 2 RMSE that exceed the target error. The test with 4 hidden layer had the same total of RMSE that exceed the target error with the test with 5 hidden layer but, the difference between min and max error rate is more bigger than the test with 5 hidden layer. The test with 5 hidden layer had 2 RMSE that exceed the target error. The learning rate set at 0.9 is because the total amount of training data in this project is small, around 1000 data. If the learning rate smaller, it means the neural network will change its mind slower and if it is higher, it means the neural

network will change its mind faster. The iteration is adjusting the hidden layer and the learning. If the iteration is smaller the iteration need to be higher because the neural network will change its mind slower and need more loop to get to the targeted error.

Table 5.2: The differences between real and predicted charges

Age	Gender	BMI	Smoker	Real Charges	Predicted Charges	Differences
33	male	22.705	no	21984	22114.61	130.61
32	male	28.88	no	3867	3923.1	56.1
31	female	25.74	no	3757	3969.51	212.51
46	female	33.44	no	8241	8233.98	7.02
37	female	27.74	no	7282	7247.05	34.95
37	male	29.83	no	6406	6326.8	79.2
60	female	25.84	no	28923	28944.97	21.97
25	male	26.22	no	2721	2729.63	8.63
62	female	26.29	yes	27809	27973.05	164.05
23	male	34.4	no	1827	2174.46	347.46
56	female	39.82	no	11091	11014.18	76.82
27	male	42.13	yes	39612	39791.67	179.67
19	male	24.6	no	1837	1960.23	123.23
52	female	30.78	no	10797	10748.34	48.66
23	male	23.845	no	2395	2431.73	36.73
56	male	40.3	no	10602	10528.28	73.72
30	male	35.3	yes	36837	36620.89	216.11
60	female	36.005	no	13229	13173.4	55.6
30	female	32.4	no	4150	4226.96	76.96

The table above is a portion of prediction output and the differences between the prediction and real charges. The average of the differences between predicted and real charges is 128.174. The prediction obtained from testing using 5 hidden layers, 2000 iteration, 0.9 learning rate, 0.05 target error and got a 4.15% error (RMSE).