A Brief Review of a Machine Learning Programming of Simple Linear Regression

Yong Suk Suh*, Seung Ki Shin, Dane Baang, Sang Mun Seo
Research Reactor System Design Div., Korea Atomic Energy Research Institute (KAERI), Daedeok-Daero 989-111, Yuseong-Gu, Daejeon, 34057, Korea
**Corresponding author: yssuh@kaeri.re.kr*

1. Introduction

The machine learning programming is a process of building computer programs so that computer can produce results given an input by itself. The machine learning is categorized into supervised, unsupervised and recurrent learning [1]. This paper concerns the supervised learning because the unsupervised and recurrent are rather sophisticated to deal with in this paper. The supervised learning is a method to train a computer with predefined inputs and then the computer can produce an expected output given an input by itself.

The programming is done by human beings known as programmers who determine a domain, logic, training data, and expected results. The computer must produce the result as expected by the programmer. The problem concerned in this paper is how exactly the computer can produce the result and how the programmer accept the result because most machine learning programs are based on probabilistic approaches for the production. The uncertain factors in the output produced by a machine learning shall be identified prior to applying it to nuclear fields. Thus, the purpose of this paper is to figure out the uncertain factors in the machine learning programming of simple linear regression prior to applying it to the nuclear fields

This paper briefly reviews the probabilistic approach in a machine learning program by selecting a simple linear regression method. This paper assumes that the linear regression is acceptable for a domain so that the domain is not discussed in this paper. The training data and results are explained in next section.

2. Simple Linear Regression (SLR)

The SLR models [2] a linear relationship between independent variable X and dependent variable Y. This paper considers four types of relationship between X and Y. X values of four types are identically from 0 to 9. Y values are shown in Table 1.

Table 1 Types of Y values for SLR

Type	Arbitrary Y values as training data per type									
1	1	2	3	5	4	8	7	6	9	10
2	0	1	2	3	4	5	6	7	8	9
3	5	5	5	5	5	5	5	5	5	5
4	10	7	3	5	4	8	7	6	9	10

The four SLRs are calculated by MS-Excel program as shown in Fig. 1, which can be also achieved with the Eq. (1) or Eq. (2).

$$w = \frac{\sum(x-\bar{x})(y-\bar{y})}{\sum(x-\bar{x})^2},\ b = (\bar{y}) - (w)(\bar{x}) \qquad (1)$$

$$w = \frac{\sum(xy) - \frac{\sum x \sum y}{n}}{\sum x^2 - \frac{(\sum x)^2}{n}},\ b = (\bar{y}) - (w)(\bar{x}) \qquad (2)$$

Where *w* is the slope of SLR and *b* is the bias of SLR. $\bar{x}$ and $\bar{y}$ are mean of *x* and *y*, respectively.

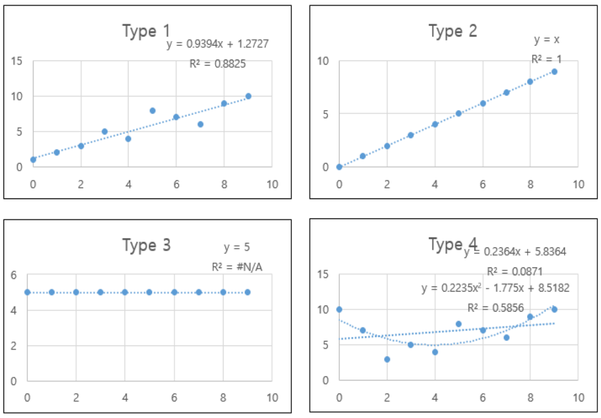

Figure 1 Four types of SLR calculated by MS-Excel

R^2 in Fig. 1 is defined as a goodness-of-fit measure for linear regression models. R^2 is called the coefficient of determination that evaluates the scatter of the Ys around the fitted regression line, which ranges from 0 to 1. When R^2 is 1 then the line is perfectly fitted as shown in Type 2 and 3 in Fig. 1. If R^2 is 1 then the SLR is not needed because the purpose of SLR is to approximate the linear model to the given data. R2 of polynomial in Type 4 shows better goodness-of-fit than R2 of linear.

3. Machine Learning Programming (MLP)

This paper uses TensorFlow [3] that was released by Google Company to build a machine learning program of linear regression. TensorFlow is an open source scientific library using Python [4] programming language. The MLP of SLR is not so complicated that we can easily build the program by keeping the following steps:

1) Prepare training data
2) Determine a cost function
3) Determine an optimization method
4) Train until the cost function is minimized

Type 1 in Table 1 is selected as training data, which is arbitrarily selected. The least square method is selected as a cost function, which is well known as a method to fit a line to data. This paper selects a gradient descent method as a cost optimization method because the least square method is a convex second order equation. The closer to zero results of the gradient descent is calculated, the more optimized the linear regression is.

In the MLP, the programmer must determine the initial value of *w* that indicates a weight or slope of linear line, learning rate that indicates steps of gradient descent, and epoch that indicates the number of training. There is no optimal ways to determine them. Therefore, this paper performs a quick-and-dirty way to determine them and analyzes the results in the following section.

4. Results and Discussions

The MLP of SLR was run with the test cases as shown in Table 2.

Table 2 Test cases

Initial value	Epoch	Learning rate
0	10	0.0001
1	100	0.001
	1000	0.01

Criteria of accepting the results of each test case is how fast *w* converges to 0.9394 shown in Type 1 in Fig. 1 with minimal cost.

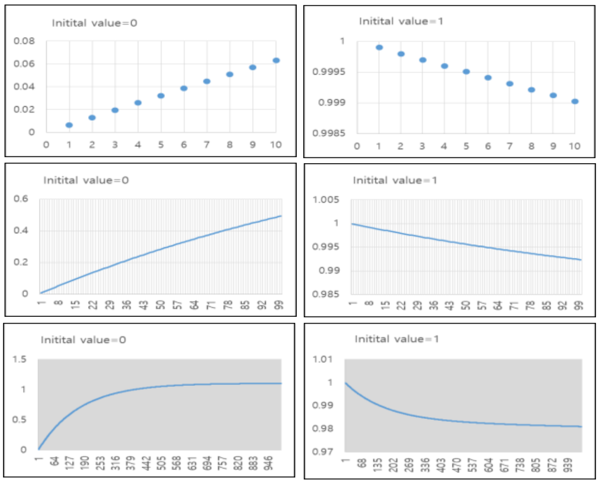

Figure 2 MLP results with learing rate = 0.0001

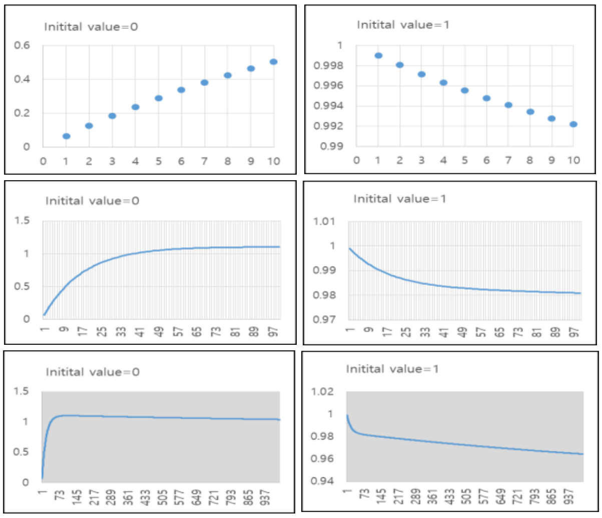

Figure 3 MLP results with learing rate = 0.001

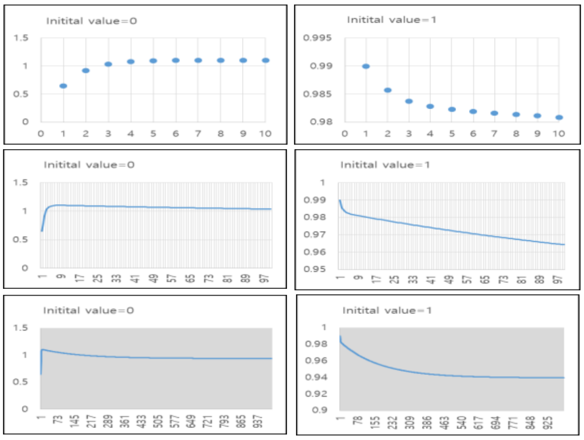

Figure 4 MLP results with learing rate = 0.01

From Fig. 2, 3 and 4, initial value = 1 and learning rate = 0.01 show better convergence to 0.9394 than others. When the cost of each learning rates is compared as shown in Fig. 5, epoch = 300 is enough for the convergence because more than 300 epochs does not lower the cost.

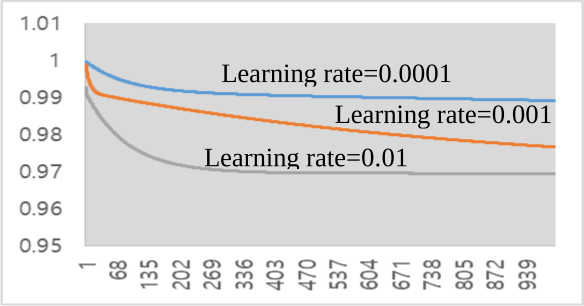

Figure 5 Cost of each MLP results

For Type 2 and 3 in Table 1, MLP produced the same results as in Fig. 2 within one epoch. Thus, it shows unnecessary to apply MLP to these cases. Uncertain factors in the MLP of SLR are initial value, learning rate and epoch in determining the slope *w* and bias *b*. It is necessary to develop a formal way to determine the slope *w* and bias *b* in order to apply the MLP of SLR to nuclear fields.

5. Conclusions

The supervised machine learning program was used for predicting an output given an input after trained with training data. The simple linear regression was reviewed in terms of machining learning program. This paper found that initial value, epoch and learning rate determined by a programmer affect the result of the machine learning program. It is necessary to establish a formal way to analytically determine the three parameters in a further study.

REFERENCES

[1] https://en.wikipedia.org/wiki/Machine_learning
[2] https://en.wikipedia.org/wiki/Linear_regression
[3] https://www.tensorflow.org/
[4] https://www.python.org/