Transactions of the Korean Nuclear Society Autumn Meeting
Yeosu, Korea, October 25-26, 2018

A Brief Review of a Machine Learning Programming of Simple Logistic 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 regression machine learning programming is a representative supervised machine learning program that predicts response (dependent) data with trained (independent) data. The simple linear regression uses the least square method to find a slope and y-intercept to determine the linearity of a set of trained data. The simple logistic regression uses the sigmoid function to find a parameter to classify the independent data into binary state: 0 or 1, which can be applied in nuclear power plants to classify measured data (or symptoms) into the binary state on the basis of experience, analysis or engineering judgement. The result of the simple linear regression machine learning program varies according to the initial value, learning rate and epoch [1]. The accuracy of the simple linear regression is determined with R-squared value that shows goodness of fitting the linearity. This paper is to find which factors impact the result of simple logistic regression machine learning program by briefly reviewing the characteristics of it. The trained data in this paper is just arbitrary data which is not real, only to see the algorithmic behavior of the simple logistic regression machine learning program.

2. Simple Logistic Regression

The logistic function was introduced as a model of population growth in 1838 [2] [3]. There are two types of population growth curves as shown in Fig. 1 [4].

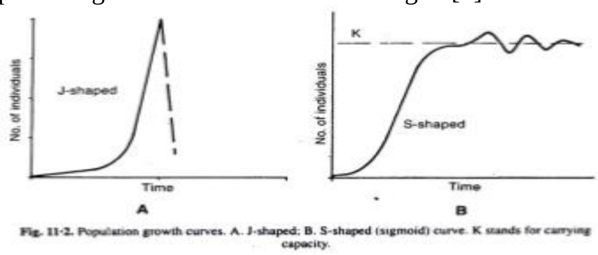

Figure 1 Two types of population growth curves (excerpted from [4])

This paper focuses on the S-shaped (sigmoid) curve that classifies the train data into two states: 0 or 1. From the population growth function, the logistic function can be drived as the following equation [5]:

$$f(\mathrm{x}) = \frac{L}{1 + e^{-k(x - x_0)}}$$

Where x is the real numbers in rage of [-∞,∞], x0 is the sigmoid's midpoint of x, L is the maximum value of the curve, k is the slope of the curve.

In this paper, L is assumed as 1. Thus, when L is 1, k is 1, and x0 is 0, then we can get the S-shaped curve as Fig. 2. The logistic function can be also drived from the concept of odds. The odds value is in range of [0,∞]. The odds is used as a ratio of two numbers or a measure of association between two likelihoods or probabilities, for example:

$$\text{Odds} = \frac{P}{1 - P}$$

Where P is the probability of the events

The odds ratio is used to evaluate how strongly the two groups "A" and "B" are related each other in terms of the odds:

$$\text{Odds Ratio} = \frac{Odds\ of\ group\ B}{Odds\ of\ group\ A}$$

When the odds ratio is equal to 1, the two groups do not affect each other. When the odds ratio is greater than 1, the probability of event happenings in group B is higher than that of group A, which means that the group A affects to increase outcome of group B.

The logit, called a log-odds, is defined as the natural logarithm of the odds:

$$\text{logit} = \ln\left(\frac{P}{1 - P}\right)$$

The logit maps the range of odds value to the range of real number [-∞,∞]. Thus, we can assume that the logit transformation of the dependent variable has a simple linear relationship with the independent variables:

$$\text{logit}(p) = \ln\left(\frac{P}{1 - P}\right) = wx$$

Where w is the slope of the linear relationship and x is the independent variable. The p is obtained as a logistic function:

$$\frac{p}{1 - p} = e^{wx}$$

$$\mathrm{p} = (1 - p)e^{wx}$$

$$\mathrm{p}(1 + e^{wx}) = \mathrm{e}^{wx}$$

$$p = \frac{e^{wx}}{1 + e^{wx}} = \frac{1}{1 + e^{-wx}}$$

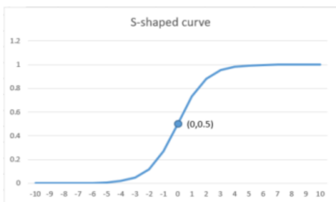

Figure 2 S-shaped curve

The logistic function is related to Bernoulli trial that outcomes 0 or 1. In the simple linear regression, the cost function, a mean squared error function, is optimized with the gradient descent method. This case is simple because the cost function is a convex function. However, when we apply the same method to the simple logistics regression, the cost function cannot be simply optimized because the gradient of logistic function is not a convex function. Therefore, we need to make the cost function possible to apply the gradient descent method. For this, the cross-entropy function is appropriate to apply the gradient descent method to optimize the cost function:

$$C(w) = -\frac{1}{n}\sum_{i=1}^{n}[y^{(i)}\log\left(L_w(x^{(i)})\right) + (1-y^{(i)})\log(1-L_w(x^{(i)}))]$$

Where C(w) is a cost function of parameter w, n is the number of train data, y that is 0 or 1 is classification of x, L_w is a result of the logistic function with the w.

Now, we can optimize the cost function with the gradient method. The accuracy of logistic regression is checked with AUC (Area Under the Curve) of ROC (Receiver Operating Characteristic) curve that is generated with the true positive rate (that is sensitivity) as a y-axis and false positive rate as a x-axis [6]. The more close to 1 the AUC is, the more accurate the regression is.

3. Machine Learning Programming (MLP)

This paper uses TensorFlow [7] released by Google Company to build a machine learning program of logistic regression and Anaconda [8] released by Anaconda Company for development environment. In this paper, the train data x and y in Table 1 are arbitral.

Table 1 Test case for logistic programming

Case-1	x	-4	-3	-2	-1	0	1	2	3	4	5
	y	0	0	0	0	1	1	1	1	1	1
Case-2	x	-4	-3	-2	-1	0	1	2	3	4	5
	y	0	0	0	0	1	1	1	0	1	1
Case-3	x	-4	-3	-2	-1	0	1	2	3	4	5
	y	0	1	0	0	1	1	1	0	1	1

The structure of simple logistic regression machine learning program using Tesnsorflow is shown in Fig. 3 with Tensorboard.

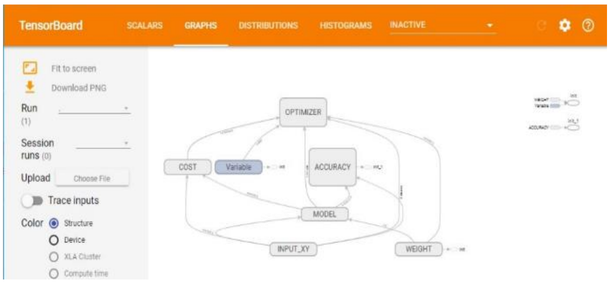

Figure 3 Structure of logistic machine learning program using Tensorflow

4. Results and Discussions

As shown in Fig. 4, the Case-1 data shows better S-shaped curve and higher AUC value than the Case 2&3. The Case-1 trained data is well organized in classifying a new output into 0 or 1. The learning rate is tested with the Case-1 data as shown in Fig. 5. The learning rate 1 shows better results than 0.1 and 0.01 of it. This is different phenomena from the linear regression which showed 0.1 is better than 1[1]. The initial value is tested with the Case-1 data as shown in Fig. 6. The initial value 3 shows better results than 1 and 2. This is the same phenomena as the linear regression [1]. In Fig. 6, the learning rate 1 shows better result than 0.1 in terms of epochs, which reaches the optimized value faster.

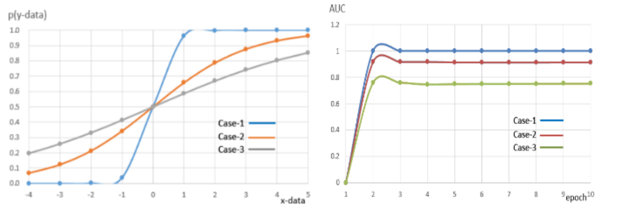

Figure 4 S-shape curve and value of AUC

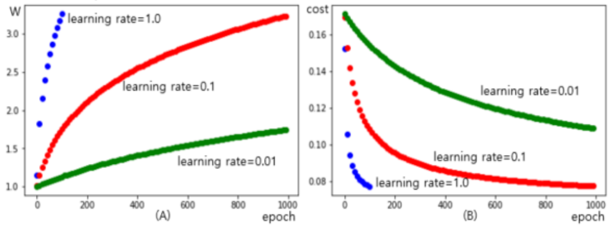

Figure 5 MLP results with learing rate

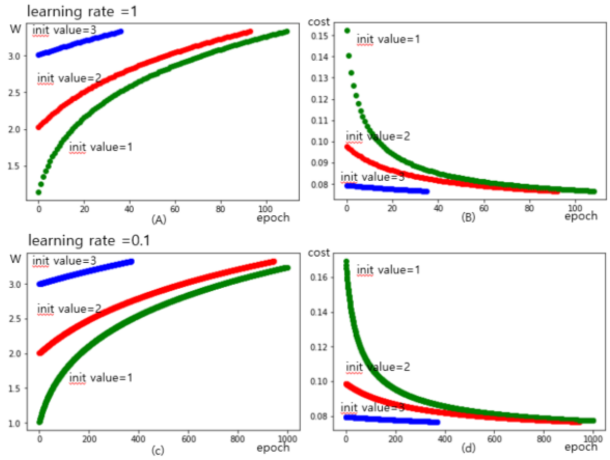

Figure 6 MLP results with initial value

5. Conclusions

As a result of brief review of the logistic regression machine learning program, outputs of it vary on the programmer's determination of initial value, epoch and learning rate. Thus, a formal method to determine them is needed to apply it to the nuclear power plants. The application of Softmax will be studied in the future.

REFERENCES

[1] Yong Suk Suh, et al., A Brief Review of a Machine Learning Programming of Simple Linear Regression, Transactions of the Korean Nuclear Society Spring Meeting, Jeju, Korea, May 17-18, 2018.
[2] http://mathworld.wolfram.com/LogisticEquation
[3] Dmitry Kucharavy and Roland De Guio, Application of S-shaped curves, TRIZ Future Conference, Procedia Engineering 9 (2011) 559-572.
[4]http://www.biologydiscussion.com/population/population-growth/population-growth-curves-ecology/51854
[5] https://en.wikipedia.org/wiki/Logistic_function
[6] https://en.wikipedia.org/wiki/Receiver_operating_characteristic
[7] https://www.tensorflow.org/
[8] https://www.anaconda.com/