

Research of Neural Network Learning Rate Control Methods for the Automatic Signal Type Recognition

T.A. Grigoryan

National Polytechnic University of Armenia, Teryan str. 105, 0009 Yerevan, Armenia

E-mail: grigoryan970@gmail.com

Received 12 April 2021

Abstract: A neural network with dynamic parameters and algorithms for the learning rate hyperparameter control has been developed in the LabVIEW software environment in order to increase the efficiency of the signal type recognition algorithm and reduce the network training time. The learning rate hyperparameter control methods testing results are presented. As a result, the most effective method for this task was selected.

<https://doi.org/10.52853/10.52853/18291171-2021.14.2-91>

Keywords: neural network, automatic signal type recognition, learning rate, linear and exponential decay, step decay, cyclic learning rate, LabVIEW software environment

1. Introduction

A neural network is a sequence of neurons connected by synapses [1, 2]. The structure of the neural network came to the programming world directly from biology. Due to this structure, the machine acquires the ability to analyze and even memorize various information.

There are many hyperparameters in a neural network that must be selected carefully for the efficient work of the algorithm. Hyperparameters are values that need to be chosen manually and often by experiments. The number of hidden layers and neurons in each layer, the presence or absence of bias neurons, the neuron activation function, the learning rate, etc. can be highlighted among these values.

The aim of this work is to research methods of the learning rate hyperparameter control and choosing the best method for the problem of automatic signal type recognition.

2. Neural network principles

The backpropagation method [4] was chosen to train the neural network in this work. The main idea of this method is to propagate the error value from the network outputs to its inputs in the direction opposite to the forward propagation of signals occurring during normal operation.

As it is known, the backpropagation algorithm uses the gradient descent method to update the neurons' weights [3]. The meaning of the gradient descent method is to find the local minimum or maximum of a function by moving along the gradient. In our case, the investigated function is the dependence of neuron loss on the selected weighting coefficient, and the task is to find the local minimum of this function.

$$w_{i,j} = w_{i,j} + \Delta w_{i,j} , \quad (1)$$

$$\Delta w_{i,j} = \eta \delta_j \frac{df(S_j)}{ds} f(S_i) . \quad (2)$$

where $w_{i,j}$ - is the weight of the synapse connecting the neurons i and j , $\Delta w_{i,j}$ - the value with which the synapse weight $w_{i,j}$ needs to be updated, η - the learning rate, δ_j - the loss value of the j -th neuron, S_j - the state of the j -th neuron, $f(S_j)$ - the value of the activation function of the j -th neuron, $f(S_i)$ - the value of the activation function of the i -th neuron.

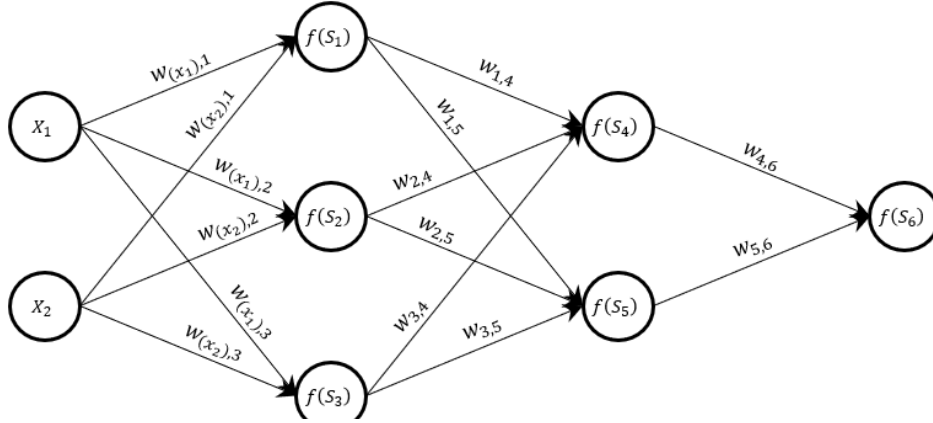


Fig. 1. An example of a multilayer neural network with two hidden layers.

From equation (2) it becomes clear that the learning rate is the magnitude of the gradient step with which the weights of the network are updated.

The learning rate is one of the most important hyperparameters of a neural network, and the wrong choice of this parameter can lead to an increase in the training time and even to the impossibility of training the network.

Fig. 2. shows the effect of learning rate on gradient descent.

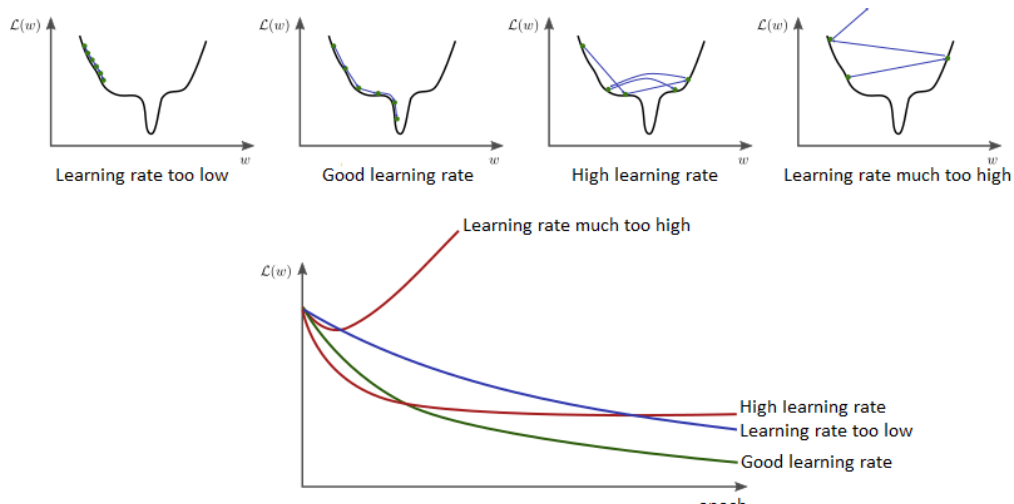


Fig. 2. The effect of learning rate on gradient descent.

A very slow learning rate will cause gradient descent to make small steps, even if the gradient is large, which slows down the learning process.

If the learning rate is high, it becomes impossible to internalize the very small parameter changes required for the exact tune of the model towards the end of the learning process, so the error is smoothed out very early. If the learning rate is very high, then gradient descent makes big steps and jumps around of minimums, which leads to an increase of error.

3. Learning rate control methods

There are many different methods for learning rate value control. The following four main methods of learning rate control are considered and compared for the problem of automatic signal type recognition:

1. Constant value,
2. Linear or exponential decay,
3. Step decay,
4. Cyclic change.

In the first method one value of the learning rate is selected and this value isn't changed anymore during the network training. Since only one value is selected for the learning rate parameter, it must be chosen carefully through experimentation, and this can be time-consuming. This method is classic and is intended for neural networks with simple architecture. In neural networks with a complex architecture, the learning process with a constant value of the learning rate parameter may fail, and the graph of the network loss vs epochs will jump up and down or freeze at a certain level.

In the second method the value of the learning rate parameter decreases linearly or exponentially at each iteration. In practice, when decreasing linearly, the learning rate often decreases to iteration n and then remains unchanged until the end of the learning process [7].

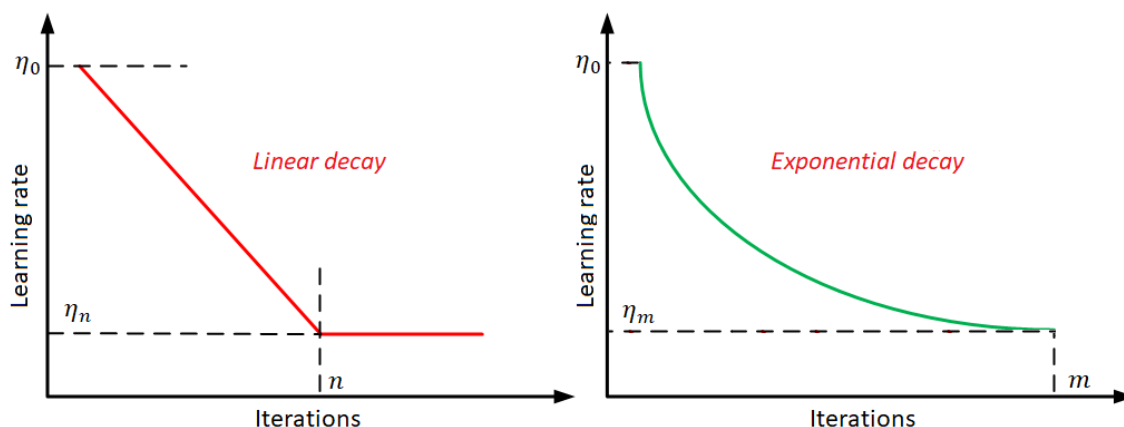


Fig. 3. Linear and exponential decay of the learning rate.

The essence of the step decay method is to divide the learning rate value by some coefficient every few epochs. Often, the learning rate value is halved, but the number of epochs, after which the value of the learning rate is necessary to divide, is a hyperparameter and depends on the network

architecture, the number of epochs for the entire learning process and the complexity of the problem [6].

The point of all the above methods was to start the learning process with a high learning rate and to decrease it during the learning process. In the cyclic change method, the learning rate graph varies between the low and high thresholds. The idea is that by increasing the learning rate, the value of the error can "jump off" the sharp minimums of the network loss graph, which will temporarily increase the network loss but may eventually lead to convergence to more desirable minimums. The upper and lower thresholds, between which the learning rate will vary, and the number of epochs during which the learning rate will reach from one threshold to another is to be chosen for this method.

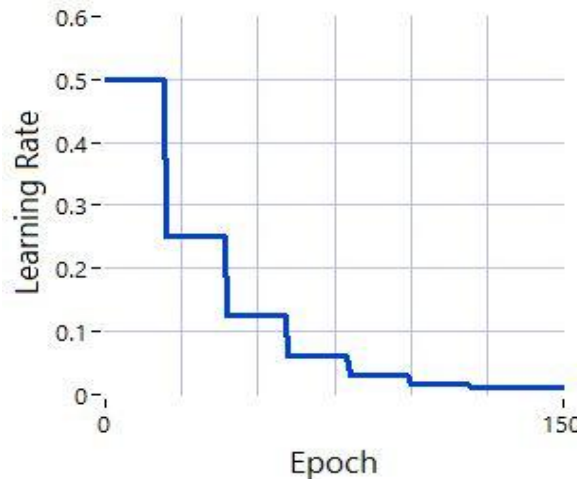


Fig. 4. An example of a step decay plot.

There are several variants of this method, for example, during one cycle, the learning rate can decrease and increase according to a linear or exponential law, and after each cycle, the value of the upper threshold can also decrease linearly or exponentially [5].

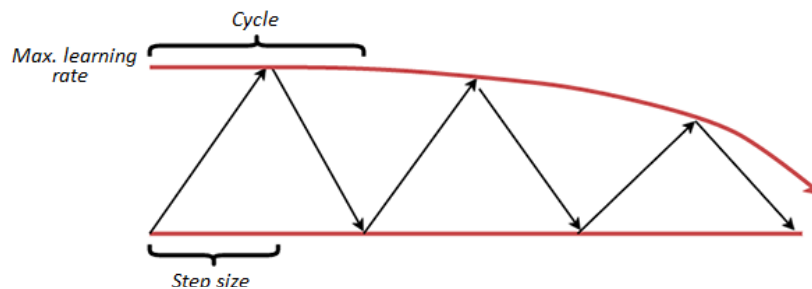


Fig. 5. An example of a learning rate value cycling change where the learning rate changes linearly, and the upper threshold value decreases exponentially.

4. Neural network training

In the frame of this work, network architecture with two hidden layers is used to solve the problem of automatic signal type recognition and to compare the methods of network learning rate control. The input of the network is the spectrum of the signal, which is represented using 500 samples. Four main types of signals in the LabVIEW software environment were selected as training data - sine, square, triangle, sawtooth. Based on this data, there will be 500 neurons in the

input layer, and 4 neurons in the output layer. The first hidden layer of the network consists of 20 neurons, and the second contains 10 neurons.

The sigmoid function is used as the activation function of neurons in the network.

$$f(x) = \frac{1}{1 + e^{-\alpha x}} . \quad (3)$$

The bias neurons also are used in the network. This is a special type of neuron, the output value of which is always equal to one, and it is not connected with the neurons of the previous layer. The bias neurons are needed so that an output result by shifting the activation function graph to the right or left can be obtained.

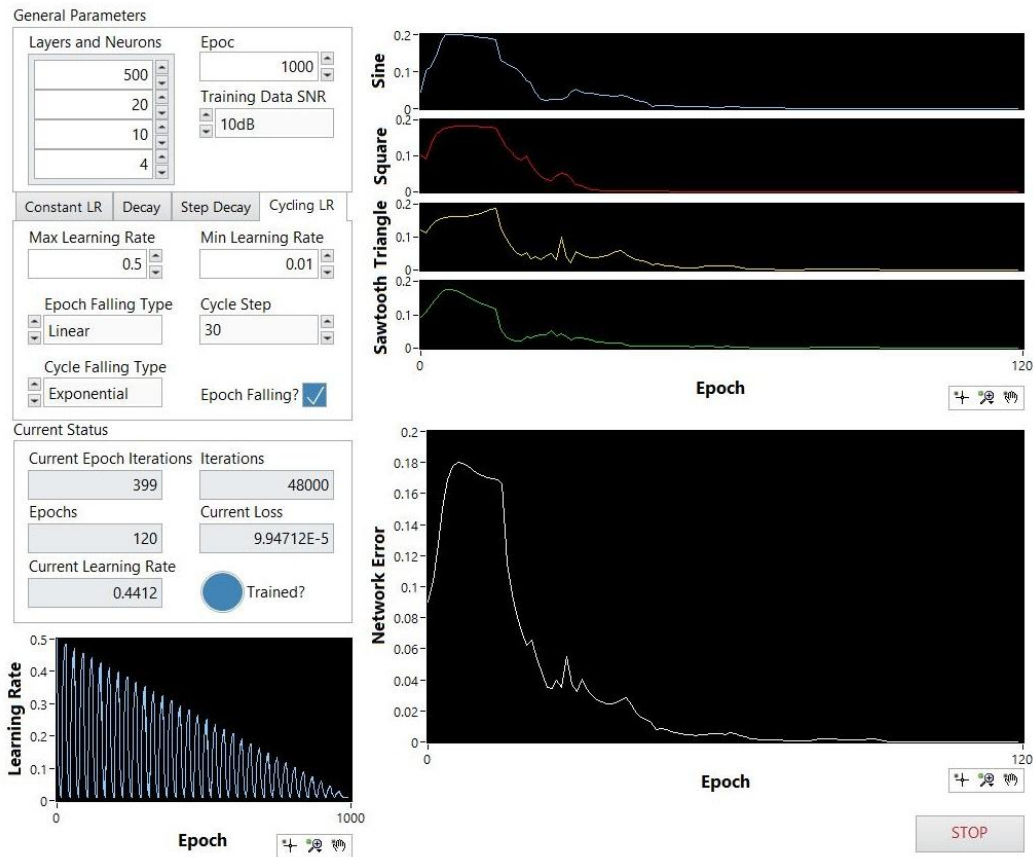


Fig. 6. The front panel of the developed program intended for neural network training.

To train the network, the corresponding types of signals with SNR = 10 dB were generated in the LabVIEW software environment. In one network training session, training data of 400 signals were generated, 100 signals for each type.

After each passed epoch, the root mean square error of the network is calculated, and if it is less than 0.0001, then the network is considered as trained.

$$\delta_{RMS} = \frac{1}{nm} \sum_{i=1}^n \sum_{j=1}^m \delta_m^2 . \quad (4)$$

where n - the size of the training data, m – the count of output neurons, δ_m - the error value of the m-th output neuron.

Table 1

<i>Training parameters and duration of the constant value method</i>			
№	Learning rate	Epochs	Iterations
1	0.5	183	73200
2	0.2505	275	101000
3	0.01	1827	730800

Table 2

<i>Training parameters and duration of the linear and exponential decay methods</i>				
№	Decay type	Threshold (epochs)	Epochs	Iterations
1	Linear	750	136	54400
2	Exponential	no	143	57200

Table 3

<i>Training parameters and duration of the step decay method</i>				
№	Step	Divide by	Epochs	Iterations
1	30	1.5	147	58800
2	30	1.4	143	57200
3	30	1.3	144	57600
4	30	1.2	167	66800

Table 4

<i>Training parameters and duration of the cyclic change method</i>					
№	Decay type (by epochs)	Decay type (by cycles)	Cycles (epochs)	Epochs	Iterations
1	no	Linear	20	162	64800
2	no	Exponential	30	111	44400
3	Linear	Linear	42	110	44000
4	Linear	Exponential	30	120	48000
5	Exponential	Linear	38	148	59200
6	Exponential	Exponential	46	126	50400

After training is done, the network parameters, in particular the weight coefficients, are saved in a text file for the further testing of the algorithm of recognition.

5. Test results

To test the trained network all saved network parameters are loaded into the second program. The program generates signals with the appropriate signal type and SNR value. The classification is carried out by the method of choosing the output neuron with the highest value. Based on the number and value of the selected neuron, the program gives the name of the corresponding signal type and the probability of guessing.

After 100 iterations, the mean probability is displayed on the front panel of the program.

Table 5. shows the results of the trained network testing for the corresponding learning rate control method.

Table 5

<i>Signal type recognition test results</i>				
Method	Sine	Square	Triangle	Sawtooth
Constant value № 1	99.47%	98.99%	98.93%	98.95%
Constant value № 2	98.58%	99.33%	98.31%	98.32%
Constant value № 3	98.5%	98.69%	98.69%	98.91%
Linear decay	98.42%	98.81%	99.31%	98.19%
Exponential decay	96.74%	98.93%	99.07%	99.15%
Step decay № 1	98.25%	98.98%	99.11%	98.54%
Step decay № 2	97.27%	99.23%	98.22%	98.82%
Step decay № 3	95.68%	98.64%	99.08%	98.68%
Step decay № 4	99.66%	99.18%	99.73%	98.04%

Cyclic change № 1	98.45%	99.02%	98.61%	98.45%
Cyclic change № 2	97.08%	98.78%	99.22%	98.31%
Cyclic change № 3	98.39%	98.98%	98.37%	98.62%
Cyclic change № 4	97.26%	98.94%	98.45%	98.45%
Cyclic change № 5	98.33%	98.81%	99.09%	98.14%
Cyclic change № 6	98.17%	98.82%	98.26%	99.84%

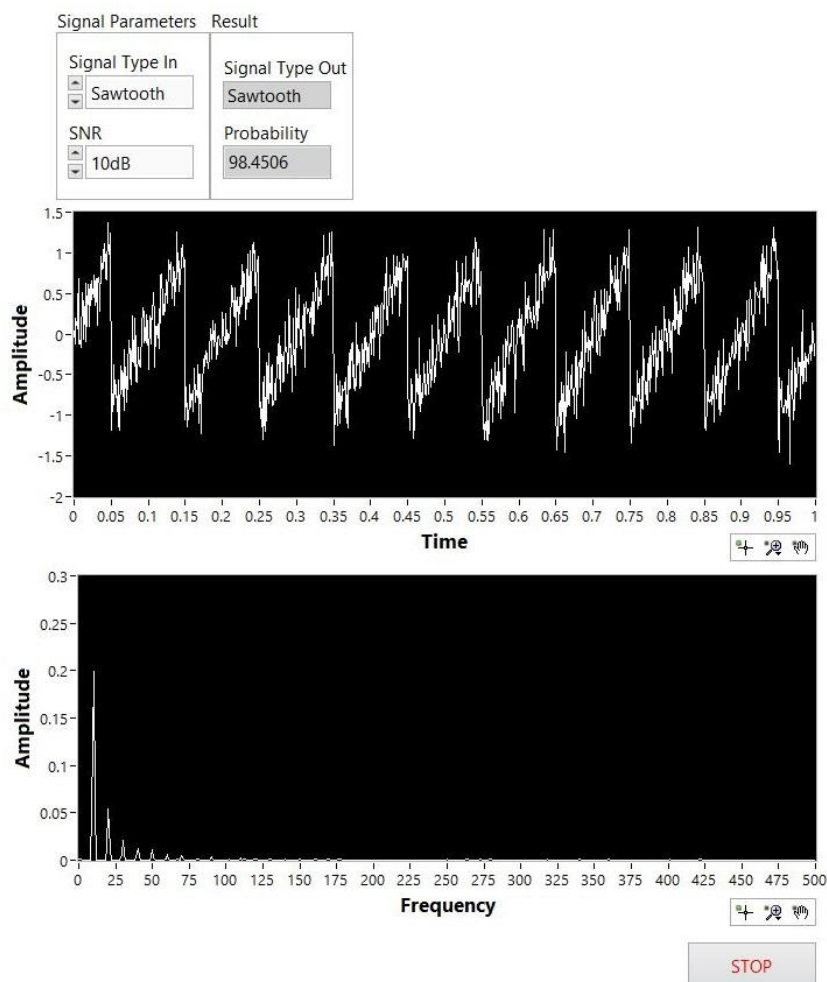


Fig. 7. The front panel of the developed program intended for trained network testing.

6. Conclusions

Thus, the software for the automatic signal type recognition based on the neural network and the algorithms for the considered methods of learning rate control have been developed.

It becomes clear from the test results that relatively better training results for the problem of automatic signal type recognition are obtained by the learning rate cyclic change method. This method is more modern and has a wide range of parameters. Various combinations of these parameters make it possible to adapt this method for a great amount of classification problems, from simple to the very complex ones.

All the main parameters of the developed neural network, particularly the network architecture, the learning rate control method and the type of input data are dynamic and can be

easily adapted for other more complex problems both in the field of radio engineering and other spheres.

References

- [1] F. Rosenblatt, Principles of neurodynamics. Perceptions and the theory of brain mechanism (Spartan Books. Washington, DC 1961).
- [2] G. Bebis, M. Georgiopoulos, Potentials IEEE **13** (1994) 27.
- [3] M.L.D. Wong, A.K. Nandi, Signal Processing N 84 (2004) 351.
- [4] W. An, H. Wang, Y. Zhang, Q. Dai, 2017 IEEE Visual Communications and Image Processing (2018), 1.
- [5] R. Ge, S. Kakade, R. Kidambi, P. Netrapalli, The step decay schedule: A near optimal, geometrically decaying learning rate procedure for least squares. arXiv:1904.12838 (2019) 1.
- [6] N. Smith, 2017 Winter Conference on Applications of Computer Vision (2017) 464.