ISSN 2224-087X. Електроніка та інформаційні технології. 2024. Випуск 25. С. 42–55
Electronics and Information Technologies. 2024. Issue 25. P. 42–55

UDK 004.9 DOI: https://doi.org/10.30970/eli.25.4

MULTILAYERED NEURAL NETWORK WITH AN AMSGrad OPTIMIZATION LEARNING METHOD

S. Sveleba[1], I. Katerynchuk[1], I. Kunyo[1], O. Semotiuk[2], Ya. Shmyhelskyy[1], S. Velhosh[1], V. Franiv[1]

[1]Ivan Franko National University of Lviv,
Universytetska str.1, Lviv, 79000, Ukraine,
incomlviv@gmail.com

[2]Ukrainian Academy of Printing,
Pid Goloskom str.19, Lviv, 79020, Ukraine

In this research, we have tested the AMSGrad optimization learning method for a multilayered neural network using the logistic function. This function describes the doubling process of the local minima number and the Fourier spectra for the error function. As a result of neural network retraining, the learning error function on each neuron is characterized by a set of wave vectors of different periodicities. The average value of the learning error for all neurons can be considered an average value for all existing periodicities. At the same time, the wave vector of the total oscillation can take both commensurate and incommensurate values. The appearance of local minima is shown to be caused by the non-homogeneous learning of the neural network, which is related to the retraining of individual neurons. As the local minimum number increases with the learning rate, so does the number of such neurons.

The AMSGrad optimization method reduces the number of retrained neurons by controlling the exponential rate of average gradients decay and the square of the error objective function gradient. In other words, the learning rate of each neuron is corrected, which removes this system's degeneracy by preventing the processes of each neuron's retraining.

Keywords: multilayered neural network, AMSGrad method, local minimums, block structure.

Introduction

Retraining is one of the crucial problems that arise during the learning process of neural networks. The retraining process occurs when the error function passes the global minimum. In addition, when the error function passes the global minimum, local minima appear. These local minima are caused by the non-homogeneous neural learning process in multilayered neural networks. The optimization methods of neural network learning are used to eliminate these shortcomings. In particular, they include Adam, AdamMax, AMSGrad, Adagrad, and other optimization methods [1]. For example, in the AMSGrad optimization method and Adagrad, the error function at each step should be considered as a set of learning errors on each neuron. The learning error on a single neuron is described by its functional dependence. It means that the learning error at a given speed and step is a symbiosis of all neurons' errors involved in the learning process. These functional dependencies of the learning error on each neuron are, as a first approximation, periodic functions with different periods.

The reason is that the AMSGrad optimization method, like Adagrad, uses a different learning rate for each parameter at each step [2]. Therefore, we first get the AMSGrad update

for each parameter and then vectorize it. In other words, in its updating rule, AMSGrad changes the overall learning rate **alpha** at each step for each parameter based on the previous gradients that have been calculated. That is why each step's error function should be considered a set of learning errors on each neuron. It is well known that when learning with a "teacher," the learning error is calculated as the standard deviation for neural networks. It means that these functional dependencies of the learning error on each neuron are, to a first approximation, periodic functions with different periods. Therefore, these functional dependencies can be characterized by a spectrum of wave vector values. These vectors have rational values (i.e., they describe commensurate fluctuations determined by the number of neurons in a layer and the number of hidden layers). The total value of the wave vector for such an ensemble of periodicities can be either commensurate or incommensurate.

The appearance of such a condition was previously observed in low-dimensional systems with a disproportional superstructure in its stochastic mode [3]. The appearance of such an incommensurate superstructure mode is caused by an increase in the anisotropic interaction strength, which leads to a decrease in the interaction between incommensalities (solitons). As a result, appear various superstructure periodicities (i.e., a periodic structure with different distances between solitons), and the average value of the incommensurate wave vector along such a structure takes on an incommensurate value. The resulting phase is incommensurate and hence chaotic.

Goals

As we know, in a multilayered neural network, all the neurons of the previous layer influence the weights correction of one neuron. Since this influence on the learning rate becomes heterogeneous when reaching a global minimum, it can lead to inhomogeneous learning. Such inhomogeneity should be detected using optimization methods based on the algorithm that updates exponentially moving average (m_t) and square gradients (v_t) based on previous values. In other words, such multilayered neural networks should be characterized by the appearance of a block structure, which is defined by the coexistence of both a chaotic state and a state with multiple harmonics.

This study aims to analyze the learning error function dependence on the learning rate to identify any such neural network state. According to the learning algorithm, such a state will be studied for a multilayered neural network using the AMSGrad optimization learning method.

The **AMSGrad method** is an advanced version of the Adam method designed to improve the convergence properties of the algorithm by avoiding large abrupt changes in the learning rate for each input variable. Technically, the descent gradient is called a first-order optimization algorithm because it explicitly uses the first-order derivative of the objective function.

The AMSGrad algorithm is known [4, 5] to update exponential mean gradient and gradient square, where the hyperparameters **β1** and **β2** (whose values vary in the interval [0,1]) control the exponential rates of decay of these mean gradients. The moving averages themselves are estimates of the 1st moment (average) and 2nd moment (non-centered variance) of the gradient [6].

In this way, AMSGrad leads to a non-increasing step size, which avoids the problems experienced by Adam. The authors of [4] also remove the offset step used in Adam to simplify the process. The complete AMSGrad update without corrected estimates can be described as follows:

$$m_t = \boldsymbol{\beta 1}^{N+1} \cdot m_{t-1} + (1 - \boldsymbol{\beta 1}^{N+1}) \cdot g_t,$$
$$v_t = \boldsymbol{\beta 2} \cdot v_{t-1} + (1 - \boldsymbol{\beta 2}) \cdot g_t^2,$$
$$\hat{v}_t = \max(\hat{v}_{t-1}, v_t),$$
$$\theta_{t+1} = \theta_t - \boldsymbol{\text{alpha}} \cdot m_t \Big/ \sqrt{\hat{v}_t + \varepsilon},$$

where g_t is the goal function (error function) gradient, m_t - is the is exponentially moving average, v_t - is the square gradients, $\hat{v}_t$ - is the maximum of the square gradients, N - is the iteration number, and ε - is the correction parameter.

At the same time, this raises the question of how the iterations' numbers affect the existence of chaotic states. Whether an increase in the iterations' number could bring the system out of the stochastic state, as long as we are in the range of changes in the learning rate corresponding to this state of the neural network? Does increasing the number of iterations lead to the disappearance of the neuronal retraining process?

Research methods

A program defining a multilayered neural network with hidden layers for recognizing printed digits was written in a Python programming environment. The array of each digit consisted of a set of "0" and "1" of 4x7 size. A sample of each digit contained a set of 4 possible digit distortions and a set of 3 arrays that did not correspond to any of the digits. Thus, the digit "0" will have the following array of values x:

```
Numt1 = [0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]
Numt2 = [1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1]
Numt3 = [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]
Num01 = [1,1,1,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,1,1,1]
Num02 = [1,1,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,1,1,1]
Num03 = [1,1,1,1,1,0,0,1,0,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,1,1,1]
Num04= [1,1,1,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,0,0,0,1,1,1,1,1]
Num05 = [1,1,1,1,1,0,0,1,1,0,1,1,1,0,0,1,1,0,0,1,1,0,0,1,1,1,1,1] ,
```

an array of values in:

```
Num0Y = [[0],[0],[0],[1],[1],[1],[1],[1]]
```

This program described a neural network with 3 hidden layers with 28 neurons in each layer. The choice of the hidden layers and neurons number in each layer was determined by the smallest learning error for digit recognition. According to [6], this is a three-layer neural network with 28 neurons in each layer. The parameters **β1** and **β2** values were selected as proposed in [7]. According to [8], the activation function was chosen as sigmoidal.

The following code was used to implement a given optimization method:

```
for i in range (num - 1):
  layer_errors.append(layer_deltas[i].dot(synapse[num - 1 - i].T))
  layer_deltas.append(layer_errors[i + 1] * sigmoid_output_to_derivative(layers[num - 1 - i]))
  layer_deltass=layer_errors[i + 1] * sigmoid_output_to_derivative(layers[num - 1 - i])
```

```
    # m(t) = beta1(t) * m(t-1) + (1 - beta1(t)) * g(t)
    m = beta1**(age+1) * m[i-1] + (1.0 - beta1**(age+1)) * layer_deltass
    # v(t) = beta2 * v(t-1) + (1 - beta2) * g(t)^2
    v = (beta2 * v[i-1]) + (1.0 - beta2) * layer_deltass**2
    # vhat(t) = max(vhat(t-1), v(t))
    vhat = max(max(vhat.reshape(-1,1)), max(v.reshape(-1,1)))
     dd= m / np.sqrt(vhat+ 1e-8)
     d.append(dd)
  for i in range (num):
     synapse[num - 1 - i] -= alpha * (layers[num - 1 - i].T.dot(d[i]))
```

The following logistic function was used to analyze the error function:

$$x_{n+1} = \mathbf{alpha} - x_n - x_n^2$$

when *n* is a step, and **alpha** - a parameter that defines the learning rate.

Its fixed points are:

$$x_{1,2}^{*} = -1 \pm \sqrt{\mathbf{alpha}+1}$$

eigenvalues that can be calculated as follows:

$$\rho_{1,2}^{*} = 1 \mp 2\sqrt{\mathbf{alpha}+1}$$

The choice of a given logistic mapping is based on the fact that it describes the doubling of the oscillation frequency [9]. In our case, this process is caused by the emergence of local minima when approaching the global minimum. For one-dimensional mappings, there are 2 ways to change the stability of a fixed point when the point multiplier is ρ = + 1 and ρ = - 1. However, the number of associated bifurcations (doublings) is much larger. It is explained by the fact that they often involve more than one fixed point. Such a situation corresponds to 4 variants of bifurcations: tangent bifurcation (fold, saddle-node); transcritical bifurcation; fork-shaped bifurcation (symmetry loss bifurcation); doubling bifurcation.

Experimental data results. The influence of learning speed

Considering that **β1** = 0.9 and **β2** = 0.999 within 100 epochs, Figure 1 shows the dependence of the logistic error function on the **alpha** parameter and the Fourier spectra. The resulting branching diagram indicates that the entire studied range of **alpha** changes (0.000001÷ 0.008) can be divided into 4 parts: 1) a range of sharp decrease in the error function value (**alpha**=0.000001÷0.00002) - no retraining; 2) a range of low-variable, monotonic behavior of the error function (**alpha**=0.00002÷0. 00025) - satisfactory learning process; 3) the range of bifurcation of the error function behavior (**alpha**=0.00025÷0.00047) - retraining process; 4) the range of chaotic non-monotonic behavior of the error function from **alpha** (**alpha**=0.00047÷0.008) - the appearance of chaos. When **β2**=0.999 with **alpha** =0.00045, the system falls into the range of retraining, followed by the appearance of harmonics. (This process is weakly manifested in the Fourier spectra at **β2**=0.999 and **alpha** =0.0004 (Fig. 1, b)).

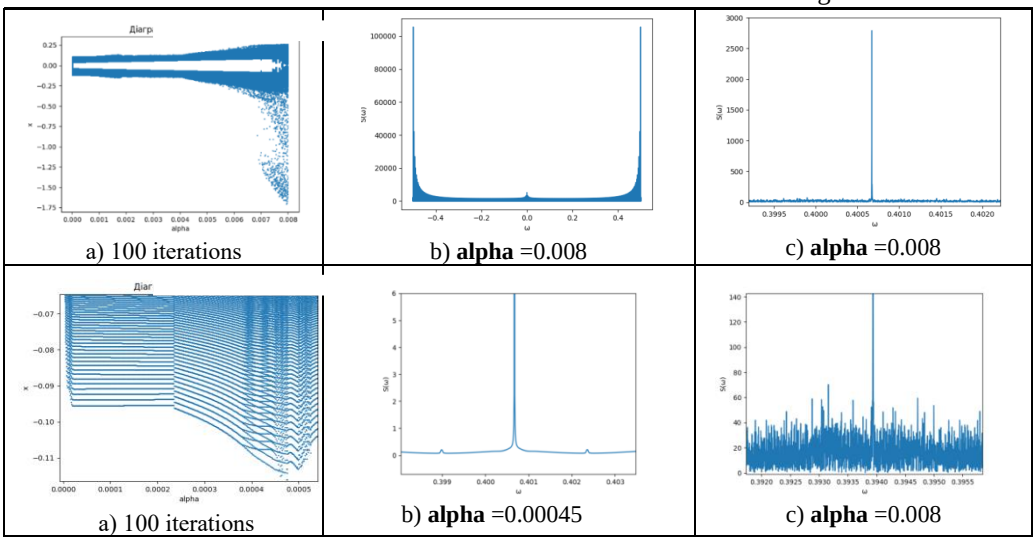


Fig. 1. Branching diagram (a) from learning rate **alpha**, Fourier spectra (b) - under satisfactory learning and retraining, and (c) - under chaos conditions, within 100 epochs, **β1**=0.9, **β2**=0.999, for the digit "0", using the AMSGrad optimization method.

Unlike the multilayered neural networks where optimization methods were not used, in the considered multilayered neural network, when using the AMSGrad optimization method, transitions cascades to and from the chaotic state are traced with increasing learning rate. Their number increases with the **alpha**-growing learning rate (Fig. 2).

When **β1**=0, the value of the moving average gradient m_t = layer_deltass, i.e., for a given multilayered neural network, will be equal to the value of the weight correction delta. The weight correction will be described by the following equation:

synapse[num - 1 - i] -= alpha * (layers[num - 1 - i].T.dot(layer_deltas[i] / np.sqrt(vhat+ 1e-8)))

In other words, the difference between a regular multilayered neural network with the backpropagation method of learning error and multilayered neural networks using the AMSGrad optimization method is the presence of the multiplier **1/np.sqrt((vhat)+ 1e-8)**. In regular multilayered neural networks, the weight correction is described by the following equation [7]:

synapse[num - 1 - i] -= alpha * (layers[num - 1 - i].T.dot(layer_deltas[i]))

The retraining process is accompanied by a passage through a global minimum and a doubling of local minimum numbers. This process is particularly well observed at small epoch values (Fig. 2). The process of blocking the doubling of the local minimum numbers, which makes it impossible for the system to transition to a chaotic state, also begins to be observed here (Fig. 2).

With an increase in the number of epochs (N), a decrease in the gradient of the error function, and an increase in the value of the equation (1- (**β1**)N), under the condition of small values of (1- **β2**), cause the ratio of the vectors of the first and second moments to begin to favor the second moment. It is manifested at small epoch values (N=5 at **alpha**>0.0017; (Fig. 2)). Thus, the optimization process manifests itself with an increase in the number of epochs, leading to a decrease in the gradient as it approaches the global minimum. It is reflected in the approximation of the value of the logistic error function to zero with an increase in the number of epochs (see Fig. 1 and Fig. 2 for branching diagrams).

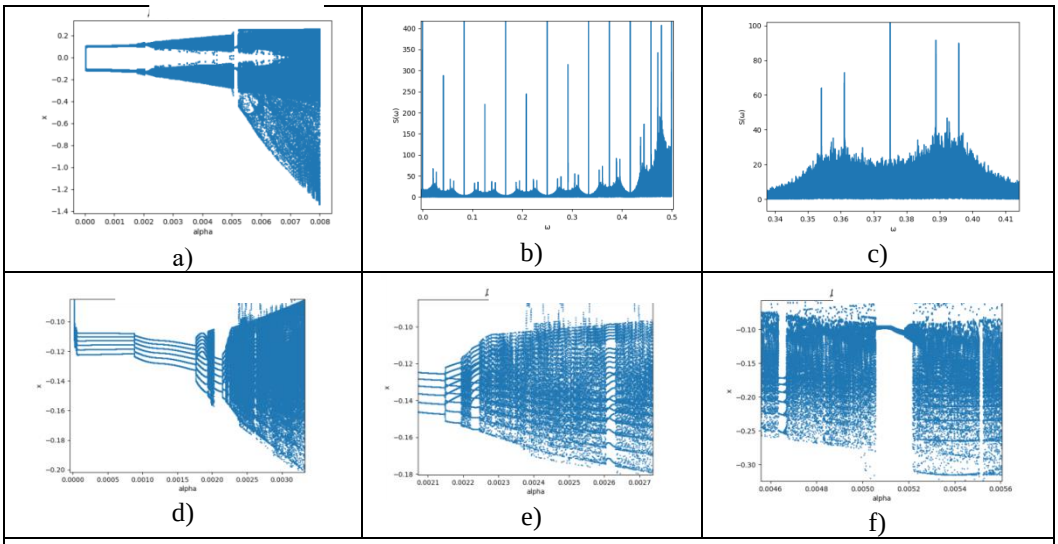


Fig. 2. Branching diagram (a) - from the learning rate **alpha**, Fourier spectra (b) - under chaos conditions, and (c) - in the enlarged version, (d), (e), (f) - branching diagrams from the learning rate alpha in the enlarged version, under the condition of 5 epochs, **β1**=0.9, **β2**=0.999, for the number "0", when using the AMSGrad optimization method.

The Fourier spectra corresponding to the maximum possible learning rate alpha show the coexistence of chaotic and periodic structures (Fig. 2 b, c). In other words, according to the branching diagram, the goal error function is characterized by the existence of a block structure, which (block), in turn, is characterized by existing of a corresponding number of local and global minima (Fig. 2, d-f).various blocks characterized by different numbers of local and global minima. Thus, the process of doubling the number of local and global minima is linked to the hyperparameter **β1**, and the hyperparameter **β2** causes the emergence of a block structure, i.e., it performs the process of gradient rarefaction.

Thus, when applying the AMSGrad optimization method to a multilayered neural network (a three-layer network with 28 neurons per layer) for recognizing printed digits, it was found that the hyperparameter **β1**, which describes the contribution of the linear gradient of the error function and is the basis of the power function of the number of epochs, is associated with a doubling of the number of local and global minima of the error function in the process of retraining the neural network. Moreover, the hyperparameter **β2**, which describes the

contribution of the square of the error function gradient, is associated with a block structure formation that prevents the doubling process and thus leads to the gradients' rarefaction.

As for the learning error, it is almost 3 times higher when the AMSGrad optimization method is not used (Table 1, Table 2). In other words, the multilayered neural network shows the best result when using the AMSGrad optimization method. While other optimization methods (Adam, AdamMax) show better teaching results compared to teaching a multilayered neural network without optimization methods, there is an uneven spread of the teaching error for different numbers. In addition, for these optimization methods (Adam, AdamMax), a satisfactory learning process is observed at 100 or more epochs, and at smaller epoch values (<10) is almost absent.

Table 1. The learning error of a multilayered neural network (with three hidden layers with 28 neurons in each layer) within 100 epochs, **alpha** = 0.001, when using stochastic optimization methods Adam, AdamMax, or AMSGrad during the recognition of printed digits, which are given by an array of 4x7 zeros and ones, if **β1** = 0.9, **β2** = 0.999.

Adam method	**AdamMax method**	**AMSGrad method**
0 error = 0.002477	0 error = 0.001186	0 error = 0.001258
1 error = 0.004718	1 error = 0.002267	1 error = 0.00134
2 error = NaN	2 error = NaN	2 error = 0.001264
3 error = NaN	3 error = 0.001677	3 error = 0.001288
4 error = NaN	4 error = NaN	4 error = 0.00137
5 error = 0.002492	5 error = 0.00011	5 error = 0.001293
6 error = 0.002069	6 error = 0.004569	6 error = 0.001298
7 error = NaN	7 error = 0.001867	7 error = 0.001317
8 error = 0.002493	8 error = 0.000294	8 error = 0.001244
9 error = 0.002439	9 error = 0.009408	9 error = 0.001294

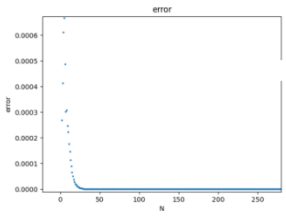
Thus, applying Adam, AdamMax, and AMSGrad optimization methods to train a multilayered neural network leads to better learning compared to a multilayered neural network, even at the optimal learning rate (the learning rate at which the number of existing local and global minima is doubled).

Table 2. The learning error of a multilayered neural network (with three hidden layers with 28 neurons in each layer) within 100 epochs, and the optimal learning rate during recognition of printed digits, which are given by an array of 4x7 zeros and ones.

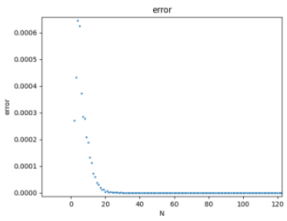
digit = 0; optimum alpha = 0.46; minimum error = 0.004829
digit = 1; optimum alpha = 0.45; minimum error = 0.004975
digit = 2; optimum alpha = 0.45; minimum error = 0.005025
digit = 3; optimum alpha = 0.45; minimum error = 0.005236
digit = 4; optimum alpha = 0.45; minimum error = 0.005088
digit = 5; optimum alpha = 0.45; minimum error = 0.00493
digit = 6; optimum alpha = 0.45; minimum error = 0.00496
digit = 7; optimum alpha = 0.46; minimum error = 0.00471
digit = 8; optimum alpha = 0.45; minimum error = 0.005261
digit = 9; optimum alpha = 0.46; minimum error = 0.004996

Impact of iterations number on the learning process

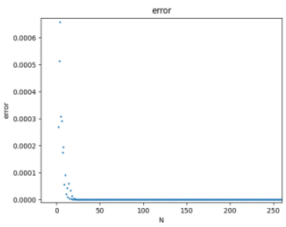
Fig. 3 shows the dependence of the learning error on the number of iterations in different modes of neural network learning (satisfactory learning mode - **alpha** = 0.0002, retraining - **alpha** = 0.0007; 0.0008; the appearance of higher harmonics - **alpha** = 0.0012; chaotic state - **alpha** = 0.0018). According to Figure 1, the learning process practically takes place in fifty iterations. Increasing the learning rate causes a slight decrease in the number of iterations. At the same time, with an increased learning rate, the process of increasing the non-monotonic behavior of the error function can be traced (Fig. 3, c-d). Thus, at a learning rate that corresponds to the neural network retraining mode, there is a non-monotonic learning process, which, with an increase in the number of iterations, turns (according to Fig. 3) into a monotonic and homogeneous learning process.



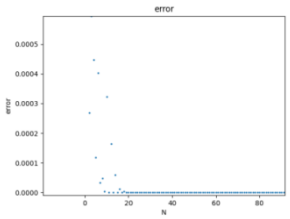
a**) alpha** = 0.0007



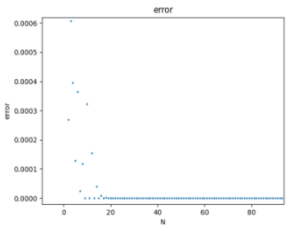
b) **alpha** = 0.0008



c) **alpha** = 0.0012



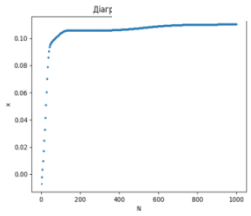
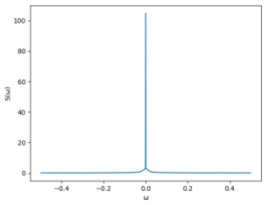
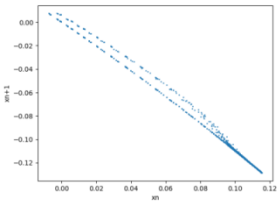
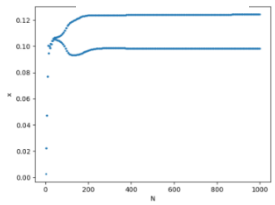
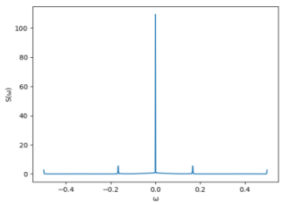
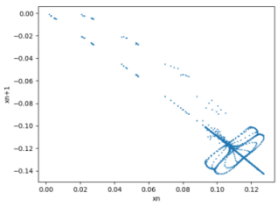
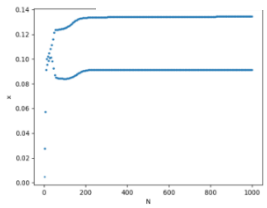
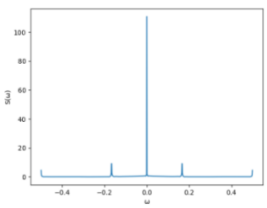
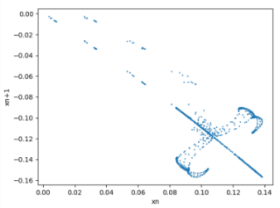
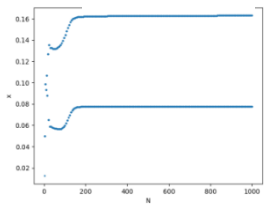
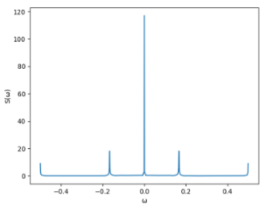
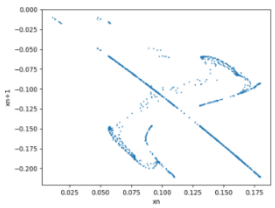
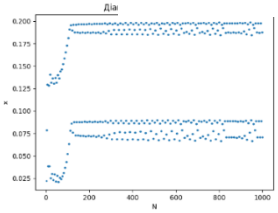
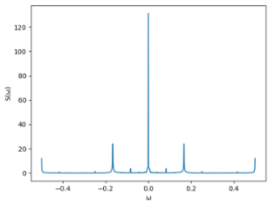
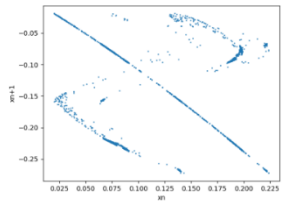
d) **alpha** = 0.0017

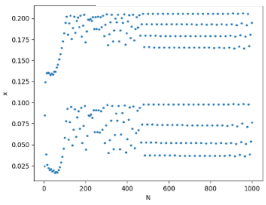


e) **alpha** = 0.0018

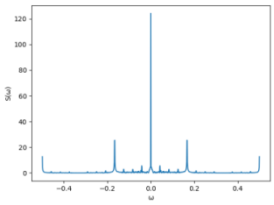
Fig. 3. Dependence of the learning error on the number of iterations at different learning rates (**alpha**), given **β1**=0.9, **β2**=0.999, for the digit "0", when using the AMSGrad optimization method.

Such a learning process can be associated with a decrease in the step size of the neural network parameter correction. However, the question arises whether, under these conditions, the process of retraining each neuron disappears and whether an increase in the number of iterations contributes to this. Therefore, let us consider the mapping function, branching diagram, and Fourier spectra for the error function. Fig. 4 shows the mapping function, branching diagram, and Fourier spectra of the error function as a function of the learning rate.

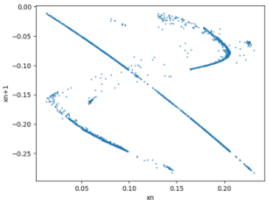
 a) **alpha**=0.0002; B1=0.9;B2=0.999	 b) **alpha**=0.0002; B1=0.9;B2=0.999	 c) **alpha**=0.0002; B1=0.9;B2=0.999
 a) **alpha**=0.0007; B1=0.9;B2=0.999	 b) **alpha**=0.0007; B1=0.9;B2=0.999	 c) **alpha**=0.0007; B1=0.9;B2=0.999
 a) **alpha**=0.0008; B1=0.9;B2=0.999	 b) **alpha**=0.0008; B1=0.9;B2=0.999	 c) **alpha**=0.0008; B1=0.9;B2=0.999
 a) **alpha**=0.0012; B1=0.9;B2=0.999	 b) **alpha**=0.0012; B1=0.9;B2=0.999	 c) **alpha**=0.0012; B1=0.9;B2=0.999
 a) **alpha**=0.0017; B1=0.9;B2=0.999	 b) **alpha**=0.0017; B1=0.9;B2=0.999	 c) **alpha**=0.0017; B1=0.9;B2=0.999



a) **alpha**=0.0018; B1=0.9;B2=0.999



b) **alpha**=0.0018; B1=0.9;B2=0.999



c) **alpha**=0.0018; B1=0.9;B2=0.999

Fig. 4. Branching diagrams by the number of iterations (a), Fourier spectra of the error function (b) and the appearance of the mapping function (c), the error function for the digit "0" when using the AMSGrad optimization method.

In other words, these dependencies were analyzed in different modes of neural network learning. Under the condition of retraining, the mapping function shows the appearance of additional oscillations, as indicated by the Fourier spectra of the error function (Fig. 4, b, and c, at **alpha** = 0.0007). With a further increase in the learning rate (**alpha** = 0.0008, **alpha** = 0.0012), the instability of the newly formed oscillations is observed. The process of retraining the neural network can be associated with the passage of a global minimum by individual neurons. In other words, the appearance of local minima on the error function is caused by the retraining of some individual neurons. Any further increase in the learning rate increases the heterogeneity of the learning process without the appearance of additional local minima. The increase in the heterogeneity of the learning process is caused by the neural network learning method itself (the method of back-propagation of error). According to the mapping function form, when the learning rate is increased to **alpha** = 0.0008, **alpha** = 0.0012, the retraining of individual neurons becomes unstable as well. Although the power of the first harmonic signal increases. It may indicate that with an increase in the learning rate, the number of neurons in which the retraining process is observed also increases. Such an increase in the number of neurons involved in retraining causes heterogeneous behavior of the error function on a single neuron.

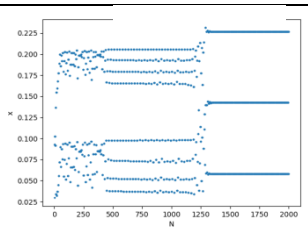
A further increase in the learning rate leads to a doubling of the local minimum number on the error function (Fig. 4, a-c, **alpha** = 0.0017). According to the authors, such a doubling is caused by an increase in the number of neurons for which the error function passes through a global minimum. The Fourier spectra of the error function begin to show the second and third harmonics (Fig. 4, b, **alpha** = 0.0017). The same process is visible on the branching diagram at **alpha** = 0.0017. A further increase in the learning rate is accompanied by an increase in the number of local minimum doublings. Under these conditions, according to the mapping function, the neuronal learning process is heterogeneous. It can ultimately lead to a chaotic learning process and hence to a chaotic state of the neural network.

However, the question arises as to the role of the AMSGrad optimization method in the learning process.

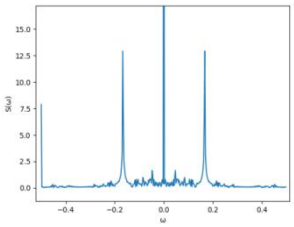
The branching diagrams on the number of iterations shown in Fig. 4, and at **alpha**=0.0017 and **alpha** = 0.0018, show the occurrence of a heterogeneous state of the error function in a certain range of changes in the iterations number. However, a further increase in iterations leads to the disappearance of such a heterogeneous state of the error function. In this case, the

neural network enters a stationary state. It is characterized by the process of retraining with the existence of several local minima (Fig. 4, a, b at **alpha** = 0.0018).

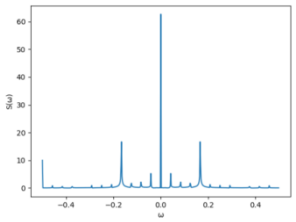
Now let us consider the most evident features of the iteration number influence on the learning process. After analyzing the neural network learning for different printed digits, we noticed that learning for different digits occurs in almost the same scenario. Namely, at a learning rate of **alpha** = 0.0018, with an increase in the number of iterations ($10<N<400$), the neural network retrains with the formation of local minima. Such retraining leads to a chaotic state of neural network learning (Fig. 5, a). A further increase in the number of iterations ($N>400$) is accompanied by a decrease in the number of neurons involved in the retraining process.



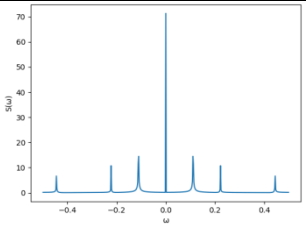
a) **alpha**=0.0018_digit 3



b) 100÷500 iterations



c) 500÷1000 iterations



d) 1500-2000 iterations

Fig. 5. Branching diagrams from the number of iterations (a), Fourier spectra of the error function (b), (c), and (d), under the condition **β1**=0.9, **β2**=0.999, for the digit "3", when using the AMSGrad optimization method.

In particular, it is indicated by the Fourier spectra obtained at $N>500$ (Fig. 5, c, d).

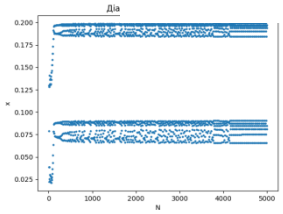
The AMSGrad algorithm is known to update the exponential moving averages of the gradient (m_t) and the square of the gradient (v_t), where the hyperparameters **β1** and **β2** control the exponential decay rates of these moving averages. Thus, with an increase in the number of iterations due to a decrease in the value of the exponential weight correction, the number of neurons involved in retraining decreases. Similar results were obtained when the average gradient moving averages (m_t) were updated according to the hyperbolic rule. That means $m_t=(1/(N+1))m_t-1+(1-(1/(N+1)))g_t$ instead of $m_t=\boldsymbol{\beta 1}^{(N+1)}m_{t-1}+(1-\boldsymbol{\beta 1}^{(N+1)})g_t$.

An analysis of the effect of the hyperparameter **β2** on the dependence of the learning error on the number of iterations showed a similar tendency of the learning process. The only difference was that the chaotic state of the neural network was observed at lower values of the alpha parameter (**alpha**=0.0006, **alpha**=0.0018).

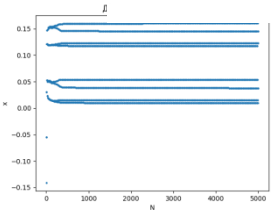
Thus, an increase in the number of neural network learning iterations when using the AMSGrad optimization method is accompanied by a decrease in the number of neurons that are inherent in the retraining process.

An increase in the number of hidden layers is known to lead to a better learning process of a multilayered neural network. Thus, for a conventional multilayered neural network,

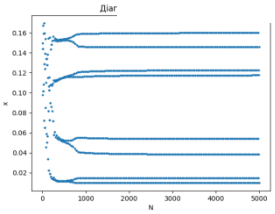
according to [7], the best learning result was achieved when considering a three-layer neural network. The AMSGrad optimization method plays an important role in teaching optimization by the square of the gradient, which is determined by the hyperparameter **β2**. Therefore, an increase in the number of hidden layers under these conditions, according to the authors, should lead to an improvement in the learning process. Figure 6 shows the branching diagrams obtained with different numbers of hidden layers. Provided that a certain number of neurons are retrained (Fig. 6, a, **alpha** = 0.0017), the branching diagram for a three-layer neural network shows the existence of a chaotic state. An increase in the number of hidden layers leads to a decrease in the number of neurons involved in retraining (Fig. 6, b-e). Such a process is accompanied by a decrease in the number of local minima. A further increase in the number of hidden layers (N>8) leads to the absence of the learning process, with the appearance of a chaotic state.



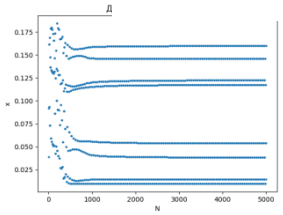
a) hidden layers=3, iterations=5000, **alpha**=0.0017



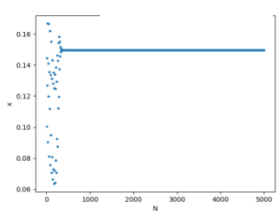
b) hidden layers=4, iterations=5000, **alpha**=0.0017



c) hidden layers=5, iterations=5000, **alpha**=0.0017



d) hidden layers=6, iterations=5000, **alpha**=0.0017



e) hidden layers=7, iterations=5000, **alpha**=0.0017

Fig. 6. Branching diagrams as a function of the number of iterations at different numbers of hidden layers, a) - 3; b) - 4; c) - 5; d) - 6; e) - 7, under the condition **β1**=0.9, **β2**=0.999, **alpha**=0.0017, for the digit "0", when using the AMSGrad optimization method.

Conclusion

Identification of funding sources and other support, and thanks to individuals and groups that assisted in the research and the preparation of the work should be included in an acknowledgment section, which is placed just before the reference section in your document.

The use of the AMSGrad optimization method for multilayered neural networks results in the appearance of a block structure of the error function during the learning process. The error function is described by a significant number of existing periodicities. This behavior of the error function is caused by the retraining of individual neurons. In other words, the increase in the number of local minima of a multilayered neural network when approaching the global minimum is caused by the process of retraining a certain number of neurons. Such retraining causes the appearance of periodic behavior of the error function. Since the error function of the neural network is a symbiosis of the error function of each neuron, its behavior will be characterized by a spectrum of possible oscillation frequencies. Depending on such a parameter as the learning rate **alpha**, the neural network error function will describe both the stationary

and chaotic states of the neural network. The stable state of the neural network is described by the retraining of a small number of neurons. Therefore, in this state, the error function is periodic and is described by several oscillations. Under these conditions, the error function is described by the existence of several local minima. As the number of local minima doubles, the neural network enters a chaotic state. Under this condition, the error function of the neural network is characterized by the spectrum of existing oscillations, and the average wave vector over such an ensemble of oscillations can take on an incommensurate value. Thus, the resulting chaotic state of the neural network is characterized by the retraining of the majority of neurons.

The optimization method AMSGrad, due to the correction of the contribution of the square of the error function gradient, leads to a decrease in the number of neurons that are retrained. It is especially evident when the number of iterations and the number of hidden layers increase.

Thus, the AMSGrad optimization method in multilayered neural networks leads to the emergence of a block structure, which indicates the heterogeneity of the neuronal learning process in the hidden layers. As the contribution from the square of the gradient increases, the neuronal learning process is equalized, which is accompanied by a decrease in the number of retrained neurons.

References

[1] *Engelbrecht A.* Computational intelligence: an introduction – Sidney: John Wiley & Sons, 2007. – 597 p. DOI: 10.1002/9780470512517

[2] *Hart P. E.* The condensed nearest neighbor rule. IEEE Transactions on Information Theory. – 1968. – Vol. 14. – P. 515–516. DOI: 10.1109/TIT.1968.1054155

[3] *Cummins H.Z.* Experimental Studies of structurally incommensurate crystal phases. Physics Reports. – 1990. – Vol.185, N 5,6. P. 211–409.

[4] *N. Jankowski, M. Grochowski*. Comparison of instance selection algorithms I. Algorithms survey. Artificial Intelligence and Soft Computing: 7th International Conference ICAISC-2004, Zakopane, 7–11 June, 2004: proceedings. – Berlin: Springer, 2004. – P. 598–603. – (Lecture Notes in Computer Science, Vol. 3070). DOI: 10.1007/978-3-540-24844-6_90

[5] *Reinartz T.* A unifying view on instance selection / T. Reinartz. Data Mining and Knowledge Discovery. – 2002. – № 6. – P. 191–210. DOI: 10.1023/A:1014047731786

[6] *S. Sveleba, I. Katerynchuk, I. Kunyo, O. Semotiuk, Ya. Shmyhelskyy, N. Sveleba*. Electronics and Information Technologies, 2021, Vol. 16, P. 20-35. DOI: https://doi.org/10.30970/eli.16.3

[7] *Tran Thi Phuong, Le Trieu Phong*. On the Convergence Proof of AMSGrad and a New Version [Submitted on 7 Apr 2019 (v1), last revised 31 Oct 2019 (this version, v4)] 1904.03590.pdf (arxiv.org).

[8] *S. Sveleba, I. Katerynchuk, I. Kunyo, O. Semotiuk, Ya. Shmyhelskyy, N. Sveleba*. Electronics and Information Technologies, 2022, 17. – P. 36–53. DOI: https://doi.org/10.30970/eli.16.3

[9] *Yu. Taranenko.* Information entropy of chaos URL: https://habr.com/ru/post/447874/

БАГАТОШАРОВА НЕЙРОМЕРЕЖА З ОПТИМІЗАЦІЙНИМ МЕТОДОМ НАВЧАННЯ AMSGrad

С. Свелеба[1], І. Катеринчук[1], І. Куньо[1], О. Семотюк[2], Я. Шмигельський[1], С. Вельгош[1], В. Франів[1]

[1] Львівський національний університет імені Івана Франка,
вул. Ген. Тарнавського, 107, 79017 Львів, Україна
incomlviv@gmail.com

[2]Українська Академія Друкарства,
вул. Під Голоском,19, 79020 Львів, Україна

В роботі з допомогою логістичної функції, що описує процес подвоєння кількості локальних мінімумів, та Фур'є спектрів функції похибки для багатошарової нейромережі, за умови застосування оптимізаційного методу AMSGrad здійснена аналіз похибки навчання. Встановлено, що застосування оптимізаційного метода AMSGrad спричиняє появу блочної структури функції похибки в процесі навчання багатошарової нейромережі. Показано, що функція похибки навчання описується значною кількісті існуючих періодичностей, які виникають внаслідок перенавчанням окремо взятих нейронів. Збільшення кількості локальних мінімумів багатошарової нейромережі при підході до глобального мінімуму, зумовлене процесом перенавчанням нейронів, яке спричиняє появу періодичної поведінки функції похибки. Оскільки функція похибки нейромережі є симбіозом від функції похибки кожного нейрону, то її поведінка буде характеризуватись спектром можливих періодичностей. В залежності від такого параметра як крок навчання **alpha**, функція похибки навчання нейромережі буде описувати як стаціонарним так і хаотичним режим навчання нейромережі. Стаціонарний режим навчання нейромережі описується перенавчанням незначної кількості нейронів, а функція похибки є періодичною функцією і описується кількома періодичнстями. За цих умові функція похибки описується існуванням кількох локальних мінімумів. При збільшенні кроку навчання **alpha**, внаслідок подвоєння кількості локальних мінімумів нейромережа переходить в хаотичний режим навчання. Показано, що даному режимі навчання функція похибки нейромережі характеризується спектром існуючих періодичностей, а середній хвильовий вектор по такому ансамблю може приймати неспівмірне значення.

Встановлено, що контроль експоненціальної швидкості спаду середніх градієнтів і квадрату градієнта цільової функції похибки в оптимізаційному методі AMSGrad приводить до зменшення кількості нейронів що перенавчаються. Тобто корекціє швидкості навчання кожного нейрона, знімає виродженість даної системи шляхом запобігання процесам перенавчання нейронів. Особливо яскраво це проявляється при збільшенні кількості ітерацій і кількості прихованих шарів. Показано, що оптимізаційний метод AMSGrad в багатошарових нейромережах, за умови неоднорідності вхідного масиву, спричиняє появу блочної структури функції похибки навчання, яка засвідчує про неоднорідність процесу навчання нейронів в прихованих шарах. Збільшення вкладу квадрата градієнта проходить до вирівнювання процесу навчання нейронів, що супроводжується зменшенням кількості нейронів які перенавчаються.

Ключові слова: багатошарова нейронна мережа, метод AMSGrad, локальні мінімуми, блочна структура

Стаття надійшла до редакції 26.12.2023.
Прийнята до друку 01.03.2024.