symmetry

MDPI

Article

Analysis of Recurrent Neural Network and Predictions

Jieun Park [1], Dokkyun Yi [1] and Sangmin Ji [2,*]

[1] Seongsan Liberal Arts College, Daegu University, Kyungsan 38453, Korea; writer2yah@daegu.ac.kr (J.P.); dkyi@daegu.ac.kr (D.Y.)

[2] Department of Mathematics, College of Natural Sciences, Chungnam National University, Daejeon 34134, Korea

* Correspondence: smji@cnu.ac.kr

Received: 14 March 2020; Accepted: 1 April 2020; Published: 13 April 2020

Abstract: This paper analyzes the operation principle and predicted value of the recurrent-neural-network (RNN) structure, which is the most basic and suitable for the change of time in the structure of a neural network for various types of artificial intelligence (AI). In particular, an RNN in which all connections are symmetric guarantees that it will converge. The operating principle of a RNN is based on linear data combinations and is composed through the synthesis of nonlinear activation functions. Linear combined data are similar to the autoregressive-moving average (ARMA) method of statistical processing. However, distortion due to the nonlinear activation function in RNNs causes the predicted value to be different from the predicted ARMA value. Through this, we know the limit of the predicted value of an RNN and the range of prediction that changes according to the learning data. In addition to mathematical proofs, numerical experiments confirmed our claims.

Keywords: recurrent neural network; analysis; ARMA; time series; prediction

1. Introduction

Artificial intelligence (AI) with machines are coming into our daily lives. In the near future, there will be no careers in a variety of fields, from driverless cars becoming commonplace, to personal-routine assistants, automatic response system (ARS) counsellors, and bank clerks. In the age of machines, it is only natural to let machines do the work [1–5], aiming for the operation principle of the machine and the direction of a machine's prediction. In this paper, we analyzed the principles of operation and prediction through a recurrent neural network (RNN) [6–8].

The RNN is an AI methodology that handles incoming data in a time order. This methodology learns about time changes and predicts them. This predictability is possible because of the recurrent structure, and it produces similar results as the time series of general statistical processing [9–12]. We calculate the predicted value of a time series by calculating the general term of the recurrence relation. Unfortunately, the RNN calculation method is very similar to that of the time series, but the activation function in a neural-network (NN) structure is a nonlinear function, so nonlinear effects appear in the prediction part. For this reason, it is very difficult to find the predicted value of a RNN. However, due to the advantages of the recurrent structure and the development of artificial-neural-network (ANN) calculation methods, the accuracy of predicted values is improving. This led to better development and greater demand for artificial neural networks (ANNs) based on RNNs. For example, long short-term memory (LSTM), gated recurrent units (GRU), and R-RNNs [13–16] start from a RNN and are used in various fields. In other words, RNN-based artificial neural networks are used in learning about time changes and the predictions corresponding to them.

There are not many papers attempting to interpret the structure of recurrent structures, and results are also lacking. First, the recurrent structure is used to find the expected value by using it iteratively according to the order of data input over time. This is to predict future values from past data. In a situation where you do not know a future value, it is natural to use the information you know to predict the future. These logical methods include the time-series method in statistical processing, which is a numerical method. The RNN structure is very similar to the combination of these two methods. Autoregressive moving average (ARMA) in time series is a method of predicting future values by creating a recurrence relation by the linear combination of historical data. More details can be found in [17,18]. Taylor's expanding RNN under certain constraints results in linear defects of historical data, such as the time series. More details are given in the text. From these results, this paper describes the range of the predicted value of a RNN.

This paper is organized as follows. Section 2 introduces and analyzes the RNN, and correlates it with existing methods. Section 3 explains the change of the predicted value through the RNN. Section 4 confirms our claim through numerical experiments.

2. RNN and ARMA Relationship

In this section, we explain how a RNN works by interpreting it. In particular, the RNN is based on the ARMA format in statistical processing. More details can be found in [19–21]. This is explained through the following process.

2.1. RNN

In this section, we explain RNN among various modified RNNs. For convenience, RNN refers to the basic RNN. The RNN that we deal with is

$$y_t = w_1 h_t + b_y, \tag{1}$$

where t represents time, y_t is a predicted value, w_1 is a real value, and h_t is a hidden layer. The hidden layer is computed by

$$h_t = \tanh(w_2 x_t + w_3 h_{t-1} + b_h), \tag{2}$$

where x_t is input data, w_2 and w_3 are real values, and h_{t-1} is the previous hidden layer. For machine learning, let LS be the set of learning data, and let $\kappa > 2$ be the number of the size of LS. In other words, when the first departure time of learning data is 1, we can say that $LS = \{x_1, x_2, ..., x_\kappa\}$. Assuming that the initial condition of the hidden layer is 0 ($h_0 = 0$), we can compute y_t for each time t. x_t is data on time and y_t is a predicted value, so we want to satisfy $y_t = x_{t+1}$. Because unhappiness does not establish the equation, an error occurs between y_t and x_{t+1}. So, let $E_t = (y_t - x_{t+1})^2$ and $E = \sum_{t=1}^{\kappa-1} E_t$. Therefore, machine learning based on RNN is the process of finding w_1, w_2, and w_3 that can minimize error value E. We used x_1, x_2,...,$x_{\kappa-1}$ in learning data LS to find w_1, w_2, and w_3 that minimize error E, and used them to predict the values (y_κ, $y_{\kappa+1}$,...) after time κ. More details can be found in [22–25].

2.2. ARMA in Time Series

People have long wanted to predict stocks. This required predictions from historical data on stocks, and various methods have been studied and utilized. In particular, the most widely and commonly used is the ARMA method, which was developed on the basis of statistics. This method simply creates a linear combination of historical data for the value to be predicted and calculates it on this basis.

$$\hat{x}_{\kappa+1} = C_0 x_\kappa + C_1 x_{\kappa-1} + C_2 x_{\kappa-2} +, \cdots, C_l x_0 + C^* e, \tag{3}$$

where $x_0, \cdots, x_\kappa$ are given data, and we can calculate predicted value $\hat{x}_{l+1}$ by calculating the values of $C_0, \cdots, C_\kappa$, and C^*. In order to obtain the values of $C_0, \cdots, C_\kappa$, and C^*, there are various methods, such as optimization by numerical data values, Yule–Walker estimation, and corelation calculation. This equation is used to predict future values through the calculation of general terms of the recurrence relation. More details can be found in [17].

2.3. RNN and ARMA

In RNN, the hidden layer is constructed by the hyperbolic tangent function that is

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \tag{4}$$

Function tanh is expanded:

$$\tanh(x) = x - \frac{1}{3}x^3 + \frac{2}{15}x^5 - \frac{17}{315}x^7 + ..., \tag{5}$$

where x is in $[-\pi/2, \pi/2]$. Using this fact and expanding h_t,

$$h_t = \tanh(w_2 x_t + w_3 h_{t-1}) = w_2 x_t + w_3 h_{t-1} + e_t, \tag{6}$$

where e_t is an error. Therefore, $y_t = w_1 w_2 x_t + w_1 w_3 h_{t-1} + w_1 e_t$.

Since the same process is repeated for h_{t-1},

$$\begin{aligned} y_t &= w_1 w_2 x_t + w_1 w_3 h_{t-1} + w_1 e_t & (7)\\ &= w_1 w_2 x_t + w_1 w_3 \left(w_2 x_{t-1} + w_3 h_{t-2} + e_{t-1}\right) + w_1 e_t & (8)\\ &= w_1 w_2 x_t + w_1 w_2 w_3 x_{t-1} + w_1 w_3^2 h_{t-2} + w_1 e_t + w_1 w_3 e_{t-1}. & (9) \end{aligned}$$

Repeatedly,

$$y_t = w_1 w_2 x_t + w_1 w_2 w_3 x_{t-1} + w_1 w_2 w_3^2 x_{t-2} + w_1 w_3^3 h_{t-3} + w_1 e_t + w_1 w_3 e_{t-1} + w_1 w_3^2 e_{t-2}. \tag{10}$$

Therefore,

$$y_t = \sum_{k=0}^{t-1} \left(w_1 w_2 w_3^k x_{t-k} + w_1 w_3^k e_{t-k}\right) + w_1 w_3^t h_0. \tag{11}$$

If w_3 is less than 0.1, the terms after the fourth order (w_3^4) are too small to affect the value to be predicted. Conversely, if w_3 is greater than 1, the value to be predicted increases exponentially. Under the assumption that we can expand hyperbolic tangent function (tanh), condition w_3 must be less than 1. Since we can change only w_1, w_2, and w_3, the RNN can be written as

$$\begin{aligned} y_t &= w_1 w_2 x_t + w_1 w_2 w_3 x_{t-1} + w_1 w_2 w_3^2 x_{t-2} + w_1 w_2 w_3^3 x_{t-3} + w_1 w_2 w_3^4 x_{t-4} & (12)\\ &\quad + w_1 w_2 e_t + w_1 w_3 e_{t-1} + w_1 w_3^2 e_{t-2} + w_1 w_3^3 e_{t-3} + w_1 w_3^4 e_{t-4}. & (13) \end{aligned}$$

This equation is an ARMA of order 5. More details can be found in [18]. This development method was developed on the premise that the variable part of the tanh function is smaller than a specific value ($\tanh(x)$ and $|x| < \pi/2$), and is limited in terms of utilization.

3. Analysis of Predicted Values

From the above section, w_1, w_2, w_3, b_y, and b_h are fixed. Then, we obtained sequence $\{y_\kappa\}$ by the following equality:

$$\begin{aligned} y_{\kappa+1} &= w_1 h_\kappa + b_y = w_1 \tanh\left(w_2 y_\kappa + w_3 h_{\kappa-1} + b_h\right) + b_y, \\ h_\kappa &= \tanh\left(\theta h_{\kappa-1} + b\right), \end{aligned} \tag{14}$$

where $\theta = w_1 w_2 + w_3$ and $b = b_h + w_2 b_y$.

Theorem 1. *Sequence $\{h_\kappa\}$ is bounded and has a converging subsequence.*

Proof. Since $|\tanh| \leq 1$, $|h_\kappa| \leq 1$ for all l. Using the Arzela–Ascoli theorem, there exists a converging subsequence. More details can be found in [26]. □

In order to see the change in the value of h_κ, if the limit of h_κ is h, Equation (14) is written as $h = \tanh(\theta h + b)$. Therefore, as the values of θ and b, the value of h that satisfies this equation changes.

3.1. Limit Points of Prediction Values

We now analyze the convergence value of the sequence. In order to see the convergence of the sequence, we introduced the following functions:

$$y = x \tag{15}$$

$$y = \tanh\left(\theta x + b\right). \tag{16}$$

For calculation convenience, this equation changes as follows.

$$z = \tanh\left(\theta z + b\right). \tag{17}$$

where z_0 is an initial condition, the convergence of z_κ is z_*, and z_* satisfies Equation (17) ($z_* = \tanh(\theta z_* + b)$). Therefore, we have to look at the roots that satisfy the expression in Equation (17).

Theorem 2. *There should be at least one solution to Equation (17)*

Proof. Let $g(z) = \tanh(\theta z + b) - z$. Function g is continuous and differentiable. If $z < -2$, then $g(z) > 0$; If $z > 2$, then $g(z) < 0$. Therefore, there exists at least one solution. □

Theorem 3. *If $\theta \leq 1$, then the equation has just one solution.*

Proof. If $\theta \leq 1$, then $g'(z) = \theta \mathrm{sech}^2(\theta z + b) - 1 \leq 0$. Therefore, g is a monotonically decreasing function. As a result of this, there exists only one solution satisfying $g = 0$. □

Under the assumption that the value of $\theta > 1$, two values satisfying $g'(z) = 0$ necessarily exist. Therefore, assuming $\theta > 1$, we find z_l and z_r satisfying $\theta \mathrm{sech}^2(\theta z_l + b) - 1 = \theta \mathrm{sech}^2(\theta z_r + b) - 1 = 0$, and have $g(z_l) < g(z_r)$ assuming $z_l < z_r$. Therefore, $g'(z) < 0$ on $z < z_l$, $g'(z) > 0$ on $z_l < z < z_r$, and $g'(z)$ on $z_r < z$ from computing g. Assuming $g(z_l) = 0$ and $g(z_r) = 0$, we have $b = b_l = \theta \tanh\left(\left(\mathrm{sech}^2\right)^{-1}(1/\theta)\right) - \left(\mathrm{sech}^2\right)^{-1}(1/\theta)$ and $b = b_r = \theta \tanh\left(\left(\mathrm{sech}^2\right)^{-1}(1/\theta)\right) - \left(\mathrm{sech}^2\right)^{-1}(1/\theta)$, respectively. From computing sech^2, $b_r < b_l$ is obtained.

Theorem 4. *Assuming $\theta > 1$, If $b = b_l$ or $b = b_r$ then, g has two solutions. If $b_r < b < b_l$, then g has three solutions. If $b_l < b$ or $b < b_r$, then g has one solution.*

Proof. This proof assumes that $\theta > 1$. If $b < b_r$, then we know $g(z_r) < 0$. Therefore, we have $g(z_l) < g(z_r) < 0$. Since $g(z)$ is a monotonically decreasing function on $z < z_l$, there exists a unique

solution, such that $g(z) = 0$. If $b = b_r$, then we know $g(z_r) = 0$. Therefore, we know $g(z_l) < g(z_r) = 0$, and there exists a unique solution, such that $g(z) = 0$ on $z < z_l$ for the same reason. So, if $b = b_r$, we have two solutions. One is $g(z) = 0$ on $z < z_l$ and the other is $g(z_r) = 0$. If $b_r < b < b_l$, we have $g(z_l) <$ and $g(z_r) > 0$. There are three solutions, such that $g(z) = 0$ on $z < z_l$, $g(z) = 0$ on $z_l < z < z_r$, and $g(z) = 0$ on $z_l < z$. If $b = b_l$, we know that $g(z_l) = 0$. Therefore, since $g(z_r) > 0$, and g is a monotonically decreasing function on $z_r < z$, there is a solution satisfying $g(z) = 0$. So, if $b = b_l$, we have two solutions, such that $g(z_l) = 0$ and $g(z) = 0$ on $z_r < z$. If $b_l < b$, then $g(z_l) > 0$. Since $g(z_r) > g(z_l)$ and g is a decrease function, there is a solution, such that $g(z) = 0$ on $z > z_r$. □

In this section, we see the change in the number of solutions that satisfy Equation (17) as the values of θ and b change. The change of the sequence according to the initial condition of the sequence and according to the number of each solution of Equation (17) is explained.

Figure 1 shows the graph of z_l and z_r. If point (θ, b) is contained in the white region, there is one solution. If point (θ, b) lies in the red curve, there are two solutions. If point (θ, b) is contained in the blue region, there are three solutions. In Section 4, we plot point (a, b) in the solution number region to check for the number of solutions of each case.

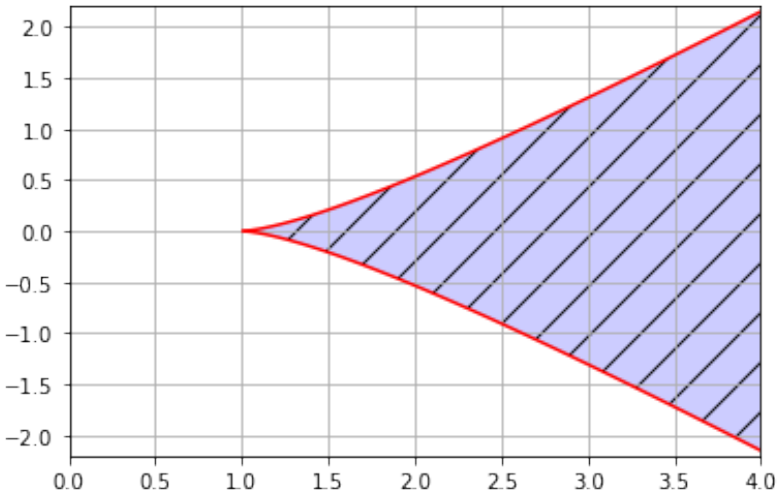


Figure 1. Solution number region.

3.2. Change of Prediction Values (Sequence)

We examined the number of the solutions of g depending on the values of θ and b. In order to see the change of the predicted value according to the change of θ and b, Equation (14) was changed to $z_{i+1} = \tanh(\theta z_i + b)$, and sequence $\{z_i\}$ was obtained. Sequences $\{z_i\}$, g, and h_κ have the following relationship: $z_{i+1} = z_i + g(z_i)$ and $z_0 = h_\kappa$. Therefore, the predicted value $y_{\kappa+m+1}$ was obtained by $y_{\kappa+m+1} = w_1 h_{\kappa+m} + b_y$ and $h_{\kappa+m} = z_m$. The solutions of g are the limit points of sequence $\{z_i\}$ by using $z_{i+1} = z_i + g(z_i)$. One of the reasons we interpreted the predictions was to identify the movement condensation (the changing value) of the predictions. We saw various cases that made function g zero from the previous theorem. The change of the sequence according to initial condition z_0 in each case is explained.

Theorem 5. *Assuming $\theta > 1$ and $b_l < b$, sequence $\{z_i\}$ converged to z_*, where z_* satisfies $g(z_*) = 0$.*

Proof. Under condition $\theta > 1$ and $b_l < b$, $g(z) > 0$ on $z < z_*$ and $g(z) < 0$ on $z_* < z$. If $z_0 < z_*$ then $g(z_0) > 0$. From computing, $\{z_i\}$ is a monotonically increasing sequence. So, sequence $\{z_i\}$ converges to z_*. If $z_* < z_0$ then $g(z_0) < 0$. From computing, $\{z_i\}$ is a monotonically decreasing sequence. Therefore, sequence $\{z_i\}$ converged to z_*. □

Theorem 6. *Assuming $\theta > 1$ and $b_l = b$, there exist two solutions z_l and z_* that satisfy $g(z) = 0$. If $z_0 < z_l$, sequence $\{z_i\}$ converges to z_l. If $z_l < z_0$, sequence $\{z_i\}$ converges to z_*,*

Proof. $0 \leq g(z)$ on $z < z_*$. So $\{z_i\}$ is a monotonically increasing sequence from computing. If $z_0 < z_l$, $\{z_i\}$ converges to z_l; if $z_l < z_0 < z_*$, $\{z_i\}$ converges to z_*. On z □

Theorem 7. *Assuming $\theta > 1$ and $b_r < b < b_l$, if $z_0 < z_*$, $\{z_i\}$ converges to z_l; if $z_0 > z_*$, $\{z_i\}$ converges to z_r, where z_0 is an initial condition.*

Proof. From computing $g'(z)$, we have $g(z) > 0$ on $z < z_l$, and $\tanh(\theta z_i + b) > z_i$ on $z_0 < z_l$. Therefore sequence $\{z_i\}$ is a monotonically increasing sequence, and $\{z_i\}$ converges to z_l. From $g'(z) > 0$, g is convex, and $g(z_l) = g(z_*) = 0$ on $z_l < z < z_*$, we have $g(z) < 0$ on $z_l < z < z_*$. On $z_l < z_0 < z_*$ we have $g(z_i) < 0$ and $g(z_i) = \tanh(\theta z_i + b) - z_i < 0$. Sequence $\{z_i\}$ is a monotonically decreasing sequence, and the convergence value is z_l. With the same calculation, g is concave, and $g(z_*) = g(z_r) = 0$. Therefore, $g(z) > 0$ on $z_* < z < z_r$ and $g(z_i) = \tanh(\theta z_i + b) - z_i > 0$ on $z_* < z_0 < z_r$. Sequence $\{z_i\}$ is a monotonically increasing sequence, and the convergence value is z_r. If $z > z_r$, $g(z) < 0$. Therefore, $g(z_i) = \tanh(\theta z_i + b) - z_i > 0$ on $z_0 > z_r$. Sequence $\{z_i\}$ is a monotonically decreasing sequence, and the convergence value is z_r. □

Theorem 8. *Assuming $\theta > 1$ and $b = b_r$, there exist two solutions z_r and z_* that satisfy $g(z) = 0$. If $z_r < z_0$, sequence $\{z_i\}$ converges to z_r. If $z_* < z_0 < z_r$, sequence $\{z_i\}$ converges to z_*. If $z_0 < z_*$, sequence $\{z_i\}$ converges to z_*,*

Proof. If $z_r < z_0$, $g(z_0) < 0$. Therefore, sequence $\{z_i\}$ is a monotonically decreasing sequence. So, sequence $\{z_i\}$ converges to z_r. If $z_* < z_0 < z_r$, $g(z_0) < 0$. Therefore, sequence $\{z_i\}$ is a monotonically decreasing sequence. So, sequence $\{z_i\}$ converges to z_*. If $z_0 < z_*$, $g(z_0) > 0$. Therefore, sequence $\{z_i\}$ is a monotonically increasing sequence. So, sequence $\{z_i\}$ converges to z_*. □

Theorem 9. *Assuming $\theta > 1$ and $b < b_r$, sequence $\{z_i\}$ converges to z_*, where z_* satisfies $g(z_*) = 0$.*

Proof. Under conditions ($\theta > 1$ and $b < b_r$), $g(z_r) < 0$. Therefore if $z_* < z_0$ then $g(z_0) < 0$. Therefore, sequence $\{z_i\}$ is a monotonically decreasing sequence. So, sequence $\{z_i\}$ converges to z_*. If $z_0 < z_*$, $g(z_0) > 0$. Therefore, sequence $\{z_i\}$ is a monotonically increasing sequence. So, sequence $\{z_i\}$ converges to z_*. □

Theorem 10. *Assuming $0 \leq \theta \leq 1$, sequence $\{z_i\}$ converges to z_*, where z_* satisfies $g(z_*) = 0$.*

Proof. Under condition ($0 \leq \theta \leq 1$), $g(z)$ has a unique solution satisfying $g(z) = 0$. If $z_0 < z_*$, $g(z_0) > 0$. Therefore, sequence $\{z_i\}$ is a monotonically increasing sequence. So, sequence $\{z_i\}$ converges to z_*. If $z_* < z_0$, $g(z_0) < 0$. Therefore, sequence $\{z_i\}$ is a monotonically decreasing sequence. So, sequence $\{z_i\}$ converges to z_*. □

In condition $\theta > 0$, function $\tanh(\theta z + b)$ is an increasing function, and there is no change of the sign of θz. However, in condition $\theta < 0$, function $\tanh(\theta z + b)$ is a decreasing function, and there is change of the sign of θz.

Theorem 11. *Assuming $-1 < \theta < 0$, sequence $\{z_i\}$ converges to z_*, where z_* satisfies $g(z_*) = 0$.*

Proof.

$$z_{i+1} - z_i = \tanh(\theta z_i + b) - \tanh(\theta z_{i-1} + b) = \theta \sec^2(\zeta)\,(z_i - z_{i-1})\,, \tag{18}$$

where ζ is between z_{i-1} and z_i. Therefore,

$$|z_{i+1} - z_i| \leq \theta \, |z_i - z_{i-1}| \,. \tag{19}$$

Sequence $\{z_i\}$ is a *Cauchy sequence* that converges to z_* □

Theorem 12. *Assuming $\theta \leq -1$, sequence $\{z_i\}$ converges to z_*, where z_* satisfies $g(z_*) = 0$, or sequence $\{z_i\}$ vibrates.*

Proof.

$$z_{i+1} - z_i = \tanh(\theta z_i + b) - \tanh(\theta z_{i-1} + b) = \theta \sec^2(\zeta) \, (z_i - z_{i-1}) \,, \tag{20}$$

where ζ is between z_{i-1} and z_i. Therefore,

$$|z_{i+1} - z_i| \leq |\theta| \sec^2(\zeta) \, |z_i - z_{i-1}| \,. \tag{21}$$

If $|\theta| \sec^2(\zeta) < 1$, sequence $\{z_i\}$ is a *Cauchy sequence* that converges to z_*. If $|\theta| \sec^2(\zeta) \geq 1$, sequence $\{z_i\}$ vibrates. □

4. Numerical Experiments

In this section, we confirmed the numerical results to identify RNN analysis interpreted in the previous section. As we saw in the previous section, RNN predictions appeared in three cases. Case 1 is Equation (17) that has one solution, Case 2 is Equation (17) that has two solutions, and Case 3 is Equation (17) that has three solutions. In Cases 1 to 3, we checked the number of solutions in Equation (17), and predicted the values according to the initial conditions. In Cases 4 through 7, experiments were conducted on the situation where learning data increase, learning data increase and decrease, learning data decrease and increase, and learning data vibrate. We obtained a picture from each numerical experiment. In each figure, (a) plots the RNN predictions and the learning data, the red curve is *sin*, (b) denotes θ and b in the area of existence of the solution, and (c) is a picture of z about Equation (17).

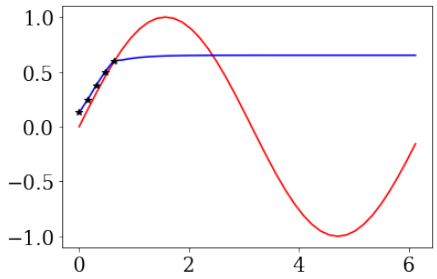
4.1. Case 1: One-Solution Case of Equation (17)

The situation with one solution was divided into the case where θ is less than 1 and θ is greater than 1.

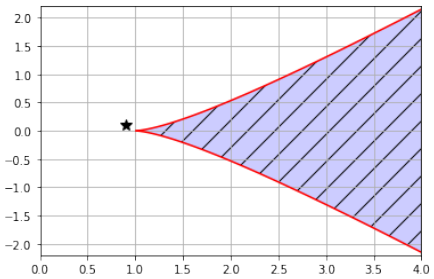
4.1.1. Theta < 1

Let $x_0 = 0$, $x_1 = 0.12$, $x_2 = 0.23$, $x_3 = 0.38$, and $x_4 = 0.5$. $x_0 \sim x_4$ are learning data. In this case, we obtained $w_1 = 0.9$, $w_2 = 0.9$, $w_3 = 0.09$, $b_y = 0.2$ and $b_h = -0.08$. Therefore, $\theta = 0.9$ and $b = 0.1$. The limit of the y_t is $y_*(0.65)$.

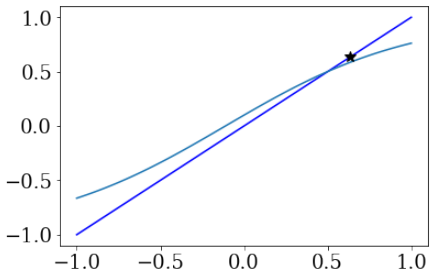
In Figure 2a, $x_0 \sim x_4$ are the black stars and $y_0 \sim y_{40}$ are the prediction values (blue line). Figure 2b shows θ and $b(* = (\theta, b))$. Figure 2c shows the result of Equation (17). In Figure 2c, $*$ is z_0. From Figure 2, we see that from the learning data, the solution of Equation (17) is one, initial value z_0 is 0.6, and z_{40} is 0.5.



(**a**) Plot of real and predicted values.



(**b**) Solution number region in Case1-1.



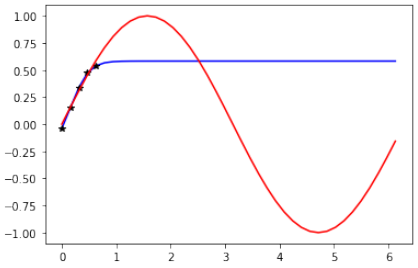
(**c**) Plot of z in Equation (17).

Figure 2. One-solution case of Equation (17) ($\theta < 1$).

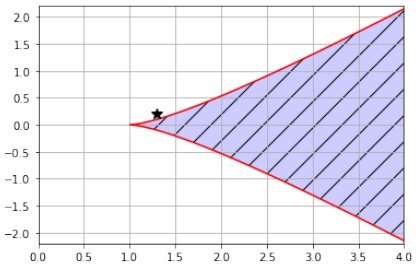
4.1.2. Theta > 1

Let $x_0 = 0$, $x_1 = -0.03$, $x_2 = 0.15$, $x_3 = 0.33$, and $x_4 = 0.4$. $x_0 \sim x_4$ are learning data. In this case, we obtained $w_1 = 0.9$, $w_2 = -0.1$, $w_3 = 1.39$, $b_y = -0.2$ and $b_h = 0.18$. Therefore, $\theta = 1.3$ and $b = 0.2$. The limit of y_t is $y_*(0.64)$.

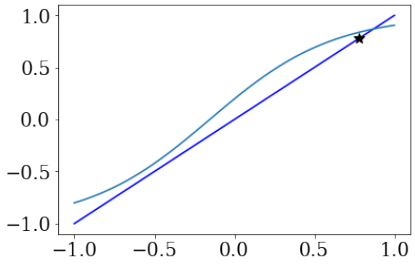
Figure 3 also shows results similar to those in Figure 2. Figure 3a shows $x_0 \sim x_4$ and $y_4 \sim y_{40}$ ($y_4 \sim y_{40}$ are the prediction values). Figure 3b shows θ and b. Figure 3c shows the result of Equation (17).



(**a**) Plot of real and predicted values.



(**b**) Solution number region in Case1-2.



(**c**) Plot of z in Equation (17).

Figure 3. One-solution case of Equation (17) ($\theta > 1$).

4.2. Case 2: Two-Solution Case of Equation (17)

This situation is two solutions of Equation (17) by (θ, b) = (1.3, 0.101). Let $x_0 = 0$, $x_1 = 0.02$, $x_2 = 0.19$, $x_3 = 0.36$, and $x_4 = 0.5$. $x_0 \sim x_4$ are learning data. Figure 4 shows the solution number region and (θ, b) (black star). As shown in Figure 4, there are two solutions to Equation (17) from the learning data. In this situation, we conducted two experiments. The first case was initial condition z_0 existing between z_l and z_r. The second case was initial condition z_0 being less than z_l. In the first case, the limited value of z_i from the proof had to go to z_r, and in the second case, the limited value of z_i from the proof had to go to z_l. This result was verified from the numerical experiments. The theory of the previous section was exempted through this numerical experiment.

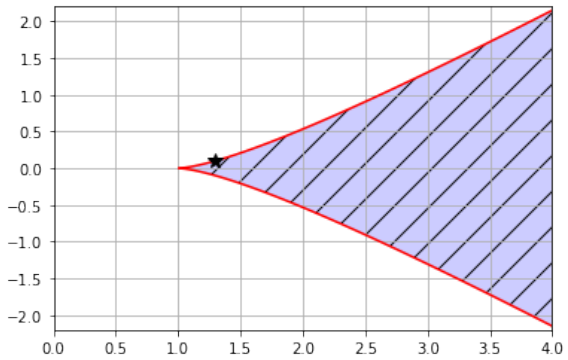
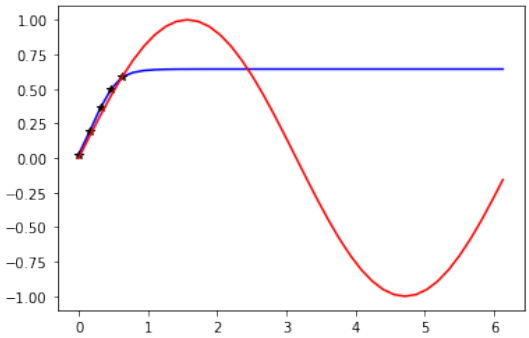


Figure 4. Solution number region in Case 2.

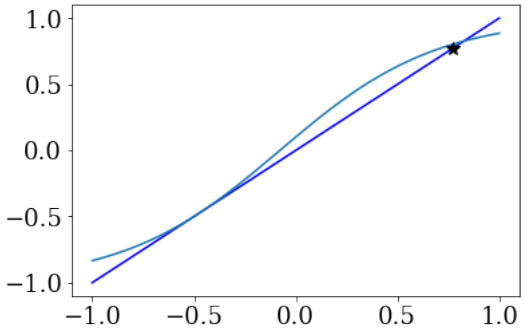
4.2.1. First Case

In this case, we obtained $w_1 = 0.9$, $w_2 = 0.4$, $w_3 = 0.94$, $b_y = -0.1$ and $b_h = 0.141$. Therefore, $\theta = 1.3$ and $b = 0.101$. The limit of y_t is $y_*(0.47)$.

Figure 5a shows that $x_0 \sim x_4$ are the black stars and $y_0 \sim y_{40}$ are the prediction values (blue line). Figure 5b shows the result of Equation (17). In Figure 5b, $*$ is z_0, and z_{40} is 0.71.



(**a**) Plot of real and predicted values.



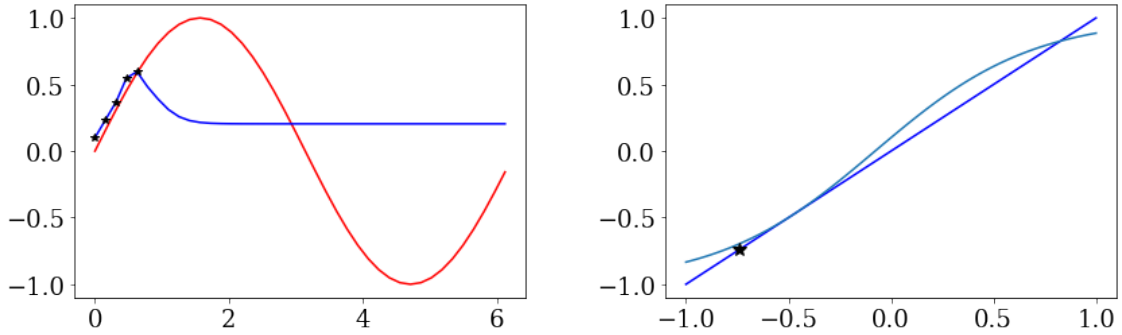
(**b**) Plot of z in Equation (17).

Figure 5. Two-solution case of Equation (17).

4.2.2. Second Case

In this case, we obtained $w_1 = -0.6$, $w_2 = -6.5$, $w_3 = -2.6$, $b_y = 0.7$ and $b_h = 4.65$. Therefore, $\theta = 1.3$ and $b = 0.101$. The limit of y_t is $y_*(0.2)$.

Figure 6a shows that $x_0 \sim x_4$ are the black stars and $y_0 \sim y_{40}$ are the prediction values (blue line). Figure 6b shows the result of Equation (17). In Figure 6b, $*$ is z_0, and z_{40} is -0.34.



(**a**) Plot of real and predicted values. (**b**) Plot of z in Equation (17).

Figure 6. Two-solution case of Equation (17).

4.3. Case 3: Three-Solution case of Equation (17)

This situation is three solutions of Equation (17) by (θ, b) = (2, 0.1). Let $x_0 = 0$, $x_1 = -0.01$, $x_2 = 0.16$, $x_3 = 0.37$, and $x_4 = 0.46$. $x_0 \sim x_4$ are learning data. Figure 7 shows the solution number region and (θ, b) (black star). As shown in Figure 7, there are three solutions from the learning data. In this situation, we conducted two experiments. For convenience, the three roots are indicated by z_l, z_*, and z_r, respectively, as in the notation above. The first case was initial condition z_0 existing between z_l and z_r. The second case is initial condition z_0 existing between z_l and z_*. In the first case, the limited value of z_i from the proof had to go to z_r, and in the second case, the limited value of z_i from the proof had to go to z_l. This result was verified from the numerical experiments. The theory of the previous section was exempted through numerical experiments.

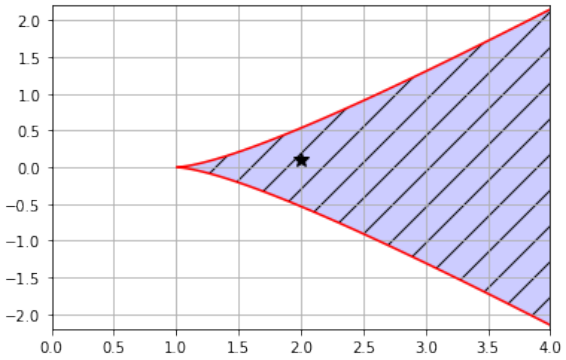
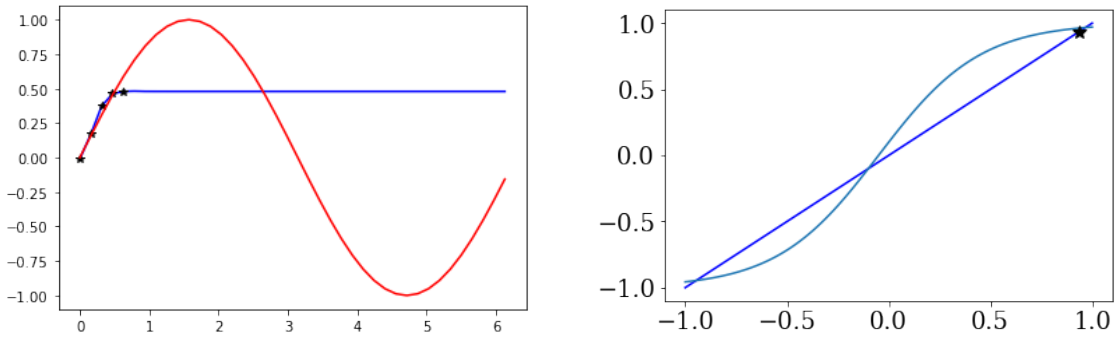


Figure 7. Solution number region in Case 3.

4.3.1. First Case

In this case, we obtained $w_1 = 0.6$, $w_2 = 0.5$, $w_3 = 1.7$, $b_y = -0.1$ and $b_h = 0.15$. Therefore, $\theta = 2$ and $b = 0.1$. The limit of the y_t is $y_*(0.58)$.

In Figure 8a, $x_0 \sim x_4$ are the black stars and $y_0 \sim y_{40}$ are the prediction values (blue line). Figure 8b shows the result of Equation (17). In Figure 8b, $*$ is z_0, and z_{40} is 0.79.



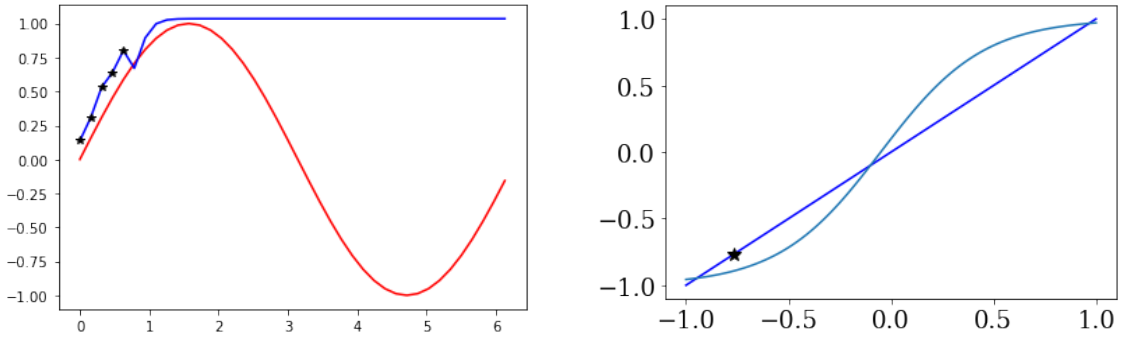
(**a**) Plot of real and predicted values. (**b**) Plot of z in Equation (17).

Figure 8. Three-solution case of Equation (17).

4.3.2. Second Case

In this case, we obtained $w_1 = -1.2$, $w_2 = -3$, $w_3 = -1.6$, $b_y = -0.1$, and $b_h = -0.2$. Therefore, $\theta = 2$ and $b = 0.1$. The limit of y_t is $y_*(1.03)$.

In Figure 9a, $x_0 \sim x_4$ are the black stars and $y_0 \sim y_{40}$ are the prediction values (blue line). Figure 9b shows the result of Equation (17). In Figure 5b, $*$ is z_0, and z_{40} is -0.86.



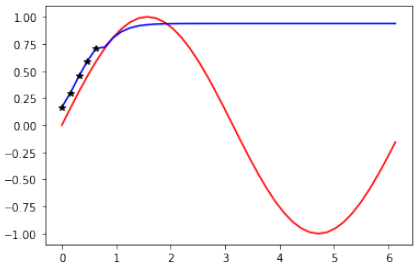
(**a**) Plot of real and predicted values. (**b**) Plot of z in Equation (17).

Figure 9. Three-solution case of Equation (17).

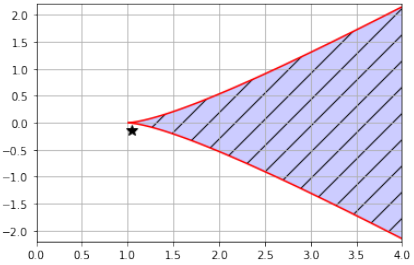
4.4. Case 4: Learning Data Increase

Let $x_0 = 0$, $x_1 = 0.15$, $x_2 = 0.3$, $x_3 = 0.45$, and $x_4 = 0.58$. $x_0 \sim x_4$ are learning data. In this case, we obtained $w_1 = -0.96$, $w_2 = -0.95$, $w_3 = 0.13$, $b_y = 0.24$, and $b_h = 0.08$. Therefore, $\theta = 1.04$ and $b = -0.15$. The limit of y_t is $y_*(0.93)$.

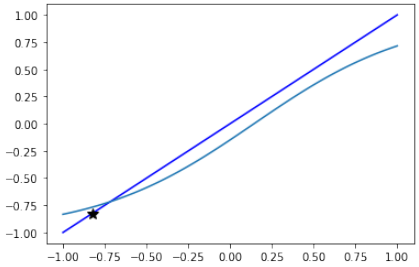
In Figure 10a, $x_0 \sim x_4$ are the black stars and $y_0 \sim y_{40}$ are the prediction values (blue line). Figure 10b shows θ and b. Figure 10c shows the result of Equation (17). In this case, x_4. From θ and b, Equation (17) has one solution. As can be seen in Figure 10, learning data increased and converged to a specific value.



(**a**) Plot of real and predicted values.



(**b**) Solution number region in Case 4.



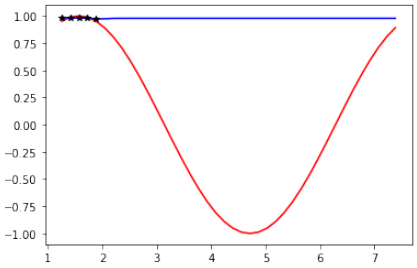
(**c**) Plot of z in Equation (17).

Figure 10. Learning data increase.

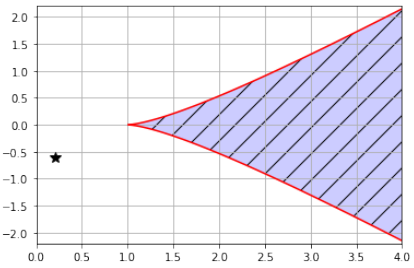
4.5. Case 5: Learning Data Increase and Decrease

Let $x_0 = 0.95$, $x_1 = 0.98$, $x_2 = 1$, $x_3 = 0.98$, and $x_4 = 0.95$. $x_0 \sim x_4$ are learning data. In this case, we obtained $w_1 = -0.49$, $w_2 = -0.58$, $w_3 = -0.07$, $b_y = 0.67$, and $b_h = -0.2$. Therefore, $\theta = 0.21$ and $b = -0.6$. The limit of y_t is $y_*(0.97)$.

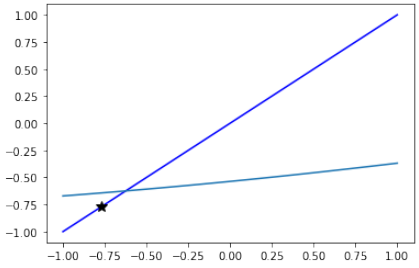
In Figure 11a, $x_0 \sim x_4$ are the black stars and $y_0 \sim y_{40}$ are the prediction values (blue line). Figure 11b shows θ and b. Figure 11c shows the result of Equation (17). In this case, x_4. From θ and b, Equation (17) has one solution. As can be seen in Figure 11, the training data converged to a specific value after increasing and decreasing. From θ and b, Equation (17) has one solution. As can be seen in Figure 11, the average value of the learning data gave the predicted value.



(**a**) Plot of real and predicted values.



(**b**) Solution number region in Case 5.



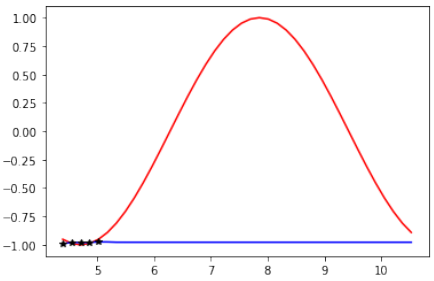
(**c**) Plot of z in Equation (17).

Figure 11. Learning data increase and decrease.

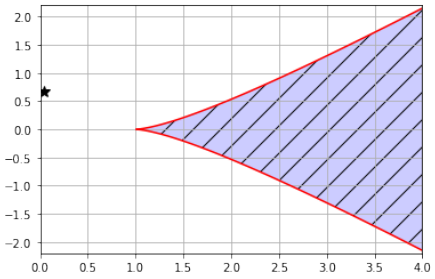
4.6. Case 6: Learning Data Decrease and Increase

Let $x_0 = -0.95$, $x_1 = -0.98$, $x_2 = -1$, $x_3 = -0.98$, and $x_4 = -0.95$. $x_0 \sim x_4$ are learning data. In this case, we obtained $w_1 = -0.32$, $w_2 = -0.55$, $w_3 = -0.14$, $b_y = -0.58$, and $b_h = 0.28$. Therefore, $\theta = 0.06$ and $b = -0.47$. The limit of y_t is y_*(-0.97).

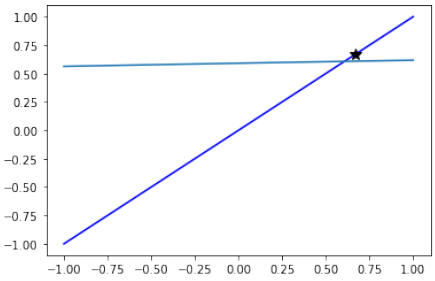
In Figure 12a, $x_0 \sim x_4$ are the black stars and $y_0 \sim y_{40}$ are the prediction values (blue line). Figure 12b shows θ and b. Figure 12c shows the result of Equation (17). In this case, x_4. From θ and b, Equation (17) has one solution. As can be seen in the Figure 12, data increased and converged to a specific value. From θ and b, Equation (17) has one solution. As can be seen in Figure 12, the average value of the learning data gave the predicted value.



(**a**) Plot of real and predicted values.



(**b**) Solution number region in Case 6.



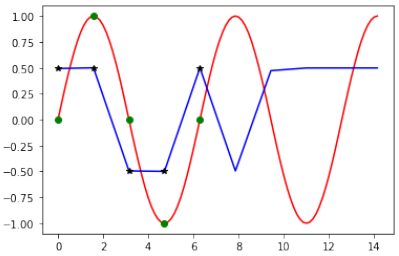
(**c**) Plot of z in Equation (17).

Figure 12. Learning data decrease and increase.

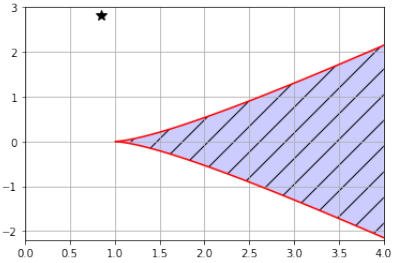
4.7. Case 7: Learning Data Vibrate

Let $x_0 = 1$, $x_1 = -1$, $x_2 = 1$, $x_3 = -1$, and $x_4 = 1$. $x_0 \sim x_4$ are learning data. In this case, we obtained $w_1 = 0.5$, $w_2 = 11.74$, $w_3 = -5.15$, $b_y = 0$, and $b_h = 2.48$. Therefore, $\theta = 0.71$ and $b = 2.48$ The limit of y_t is $y_*(0.5)$.

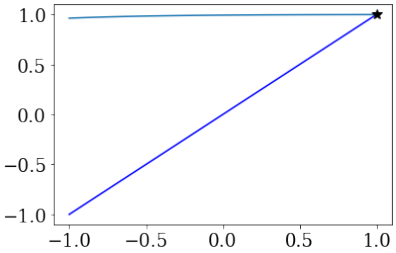
In Figure 13a, $x_0 \sim x_4$ are the green circles, $y_0 \sim y_4$ are the black stars, and $y_4 \sim y_{40}$ are the prediction values (blue line). In Figure 13a, the reason that the value of learning data (x_t) and the values of the learning result (y_t) are different is that the RNN structure was simple, and sufficient learning was not achieved. In future work, we aim to study the RNN structure to learn these complex learning data well. Figure 13b shows θ and b. Figure 13c shows the result of Equation (17). In this case, x_4. From θ and b, Equation (17) has one solution. As can be seen in Figure 13, data increased and converged to a specific value. In this case of θ and b, the solution of Equation (17) should be one. However, two contents are contradictory because learning data should be presented in two cases, 1 and -1. As a result, the cost function only increased.



(**a**) Plot of real and predicted values.



(**b**) Solution number region in Case 7.



(**c**) Plot of z in Equation (17).

Figure 13. Learning data vibrate.

5. Conclusions

In this paper, we interpreted the structure of the underlying the RNN and, on this basis, we found the principles that the RNN could predict. A basic RNN works like a time series in a very narrow range of variables. In a general range, a nonlinear function of which the maximum and minimum are specified causes the value of a function to fall within an iterative range. Because the function value is repeated within a certain range, the predicted value behaves like fixed-point iteration. In other words, we used the tanh (activation) function, so that the value was in the range of -1 to 1, and the absolute value of the predicted value in this range was less than 1. As a result, as the prediction value was repeated, the prediction value converged to a specific value. Through this paper, we found that the basic operating principle of a RNN is the operation principle of the time series, which we know as linear analysis and fixed-point iteration, which is nonlinear. In general, the solution of Equation (17) was one of the numerical calculations. Therefore, the present structure could not be solved in the case of numerical experiment Case 7 (learning data vibration). To solve this problem, it is necessary to diversify the structure, increase the number of layers, and switch to a vector structure. Next, we aim to further study RNNs in vector structures.

Author Contributions: Conceptualization, J.P. and D.Y.; Data curation, J.P.; Formal analysis, D.Y.; Funding acquisition, D.Y.; Investigation, J.P.; Methodology, D.Y. and S.J.; Project administration, J.P. and D.Y.; Resources, J.P.; Software, S.J.; Supervision, S.J.; Validation, S.J.; Visualization, S.J.; Writing—original draft, D.Y.; Writing—review & editing, J.P. and S.J. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education, Science, and Technology (grant number NRF-2017R1E1A1A03070311).

Acknowledgments: We sincerely thank the anonymous reviewers whose suggestions helped to greatly improve and clarify this manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536. [CrossRef]
2. Werbos, P.J. Generalization of backpropagation with application to a recurrent gas market model. *Neural Netw.* **1988**, *1*, 339–356. [CrossRef]

3. Schmidhuber, J. A Local Learning Algorithm for Dynamic Feedforward and Recurrent Networks. *Connect. Sci.* **1989**, *1*, 403–412. [CrossRef]
4. Cho, K.; Merrienboer, B.V.; Bahdanau, D.; Bengio, Y. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. *arXiv* **2014**, arXiv:1409.1259.
5. Jin, Z.; Zhou, G.; Gao, D.; Zhang, Y. EEG classification using sparse Bayesian extreme learning machine for brain—Computer interface. *Neural Comput. Appl.* **2018**, 1–9. [CrossRef]
6. Schmidhuber, J. A Fixed Size Storage $O(n^3)$ Time Complexity Learning Algorithm for Fully Recurrent Continually Running Networks. *Neural Comput.* **1992**, *4*, 243–248. [CrossRef]
7. Pascanu, R.; Mikolov, T.; Bengio, Y. On the difficulty of training recurrent neural networks. In Proceedings of the 30th International Conference on Machine Learning (ICML 2013), Atlanta, GA, USA, 16–21 June 2013; pp. 1310–1318.
8. Cho, K.; Merrienboer, B.V.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv* **2014**, arXiv:1406.1078.
9. Dangelmayr, G.; Gadaleta, S.; Hundley, D.; Kirby, M. Time series prediction by estimating markov probabilities through topology preserving maps. In *Applications and Science of Neural Networks, Fuzzy Systems, and Evolutionary Computation II*; International Society for Optics and Photonics: Bellingham, WA, USA, 1999; Volume 3812, pp. 86–93.
10. Wang, P.; Wang, H.; Wang, W. Finding semantics in time series. In Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data, Athens, Greece, 12–16 June 2011; pp. 385–396.
11. Afolabi, D.; Guan, S.; Man, K.L.; Wong, P.W.H.; Zhao, X. Hierarchical Meta-Learning in Time Series Forecasting for Improved Inference-Less Machine Learning. *Symmetry* **2017**, *9*, 283. [CrossRef]
12. Xu, X.; Ren, W. A Hybrid Model Based on a Two-Layer Decomposition Approach and an Optimized Neural Network for Chaotic Time Series Prediction. *Symmetry* **2019**, *11*, 610. [CrossRef]
13. Bengio, Y.; Simard, P.; Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Netw.* **1994**, *5*, 157–166. [CrossRef]
14. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
15. Gers, F.A.; Schraudolph, N.N.; Schmidhuber, J. Learning Precise Timing with LSTM Recurrent Networks. *J. Mach. Learn. Res.* **2002**, *3*, 115–143.
16. Graves, A.; Schmidhuber, J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Netw.* **2005**, *18*, 602–610. [CrossRef]
17. Brockwell, P.J.; Davis, R. *Introduction to Time-Series and Forecasting*; Springer: New York, NY, USA, 2002.
18. Shumway, R.H.; Stoffer, D.S. *Time Series Analysis and Its Applications*; Springer: New York, NY, USA, 2000.
19. Elman, J.L. Finding structure in time. *Cognit. Sci.* **1990**, *14*, 179–211. [CrossRef]
20. Rohwer, R. The moving targets training algorithm. In *Advances in Neural Information Processing Systems 2*; Touretzky, D.S., Ed.; Morgan Kaufmann: San Matteo, CA, USA, 1990; pp. 558–565.
21. Mueen, A.; Keogh, E. Online discovery and maintenance of time series motifs. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 25–28 July 2010; pp. 1089–1098.
22. Khaled, A.A.; Hosseini, S. Fuzzy adaptive imperialist competitive algorithm for global optimization. *Neural Comput. Appl.* **2015**, *26*, 813–825. [CrossRef]
23. Kingma, D.P.; Ba, J.L. Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference for Learning Representations (ICLR 2015), San Diego, CA, USA, 7–9 May 2015.
24. Zhang, Y.; Wang, Y.; Zhou, G.; Jin, J.; Wang, B.; Wang, X.; Cichocki, A. Multi-kernel extreme learning machine for EEG classification in brain-computer interfaces. *Expert Syst. Appl.* **2018**, *96*, 302–310. [CrossRef]
25. Zhang, X.; Yao, L.; Wang, X.; Monaghan, J.; Mcalpine, D.; Zhang, Y. A Survey on Deep Learning based Brain Computer Interface: Recent Advances and New Frontiers. *arXiv* **2019**, arXiv:1905.04149.
26. Yosida, K. *Functional Analysis*; Springer: New York, NY, USA, 1965.