Skipped Nonsynaptic Backpropagation for Interval-valued Long-term Cognitive Networks

Mabel Frias[1,5](✉), Gonzalo Nápoles[2], Yaima Filiberto[3], Rafael Bello[4], and Koen Vanhoof[1]

[1] Faculty of Business Economics, Universiteit Hasselt, Hasselt, Belgium
{mabel.friasdominguez,koen.vanhoof}@uhasselt.be
[2] Tilburg University, Tilburg, The Netherlands
g.r.napoles@uvt.nl
[3] AMV Solutions, Vigo, Spain
yaima.filiberto@amvsoluciones.com
[4] Department of Computer Science, Universidad Central de las Villas, Santa Clara, Cuba
rbellop@uclv.edu.cu
[5] Department of Computer Science, Universidad de Camaguey, Camaguey, Cuba
mabel.frias@reduc.edu.cu

Abstract. The recently published Interval-valued Long-term Cognitive Networks have shown promising results when reasoning under uncertainty conditions. In these recurrent neural networks, the interval weights are learned using a nonsynaptic backpropagation learning algorithm. Similar to traditional propagation-based algorithms, this variant might suffer from vanishing/exploding gradient issues. This paper proposes three skipped learning variants that do not use the backpropagation process to deliver the error signal to intermediate abstract layers (iterations in the recurrent neural network). The numerical simulations using 35 synthetic datasets confirm that the skipped variants work as well as the nonsynaptic backpropagation algorithm.

Keywords: Interval-valued Long-term Cognitive Networks · Interval sets · Nonsynaptic learning

1 Introduction

In the last years, several neural reasoning models based on Fuzzy Cognitive Maps (FCMs) [8] have been proposed. The need for models having improved approximation capabilities was concluded in the theoretical analysis presented by Concepción et al. [3]. Short-Term Cognitive Networks (STCNs) [14] is one of these models that attained superior predictive power by removing the constraint that weights should be confined to $[-1, 1]$ interval.

O. Pichardo Lagunas et al. (Eds.): MICAI 2022, LNAI 13612, pp. 3–14, 2022.
https://doi.org/10.1007/978-3-031-19493-1_1

Despite the improvements compared with traditional FCM models, STCNs often struggle to establish long-term dependencies. Long-term Cognitive Networks (LTCNs) [13] tried to overcome this issue by using a long-term reasoning mechanism. In principle, LTCNs allow learning longer dependencies with the aid of a nonsynaptic backpropagation learning algorithm.

The low simulation errors of LTCNs and their ability to retain the network interpretability have motivated the inclusion of mechanisms to deal with the uncertainty that may be present in the knowledge provided by human experts. The Long-term Grey Cognitive Networks (LTGCNs) [12] emerged as a partial solution to this issue. Even when the domain experts can express the weights using interval grey numbers in this model, the reasoning and learning processes are performed after whitening the interval grey numbers. In other words, neither the reasoning nor the learning steps fully handle the uncertainty. In contrast, the Interval-valued Long-term Cognitive Networks (IVLTCNs) [5] deal with complex systems involving uncertainty through a structure expressing the activation values and weights as interval grey numbers. The model neither imposes restrictions on the weights nor performs a whitenization process. A nonsynaptic backpropagation (IV-NSBP) is used to adjust the learnable parameters considering the uncertainty in the network without affecting the model's performance and retaining the knowledge provided by experts.

Unfortunately, the IV-NSBP method might suffer from vanishing/exploding gradient issues, a difficult task when training recurrent neural networks [2]. This paper presents three skipped IV-NSBP learning algorithms that attempt to circumvent these issues. The skipped variants modify the IV-NSBP algorithm by using different weights during the forward pass and backward pass in the training phase. Moreover, only two parameters associated with the generalized sigmoid transfer function are adjusted. It should be stated that our nonsynaptic versions are based on the algorithms presented in [11]. The simulation results show that by performing a skipping operation, we can bring the error signal directly from the output layer to any intermediate iteration (hidden abstract layer) without the need for the backpropagation process.

The outline of this paper is as follows. Section 2 briefly describes the IVLTCN model and the original IV-NSBP learning algorithm, while Sect. 3 introduces the skipped IV-NSBP variants proposed in this paper. In Sect. 4, the numerical simulations using synthetic datasets and the ensuing discussion of results are presented. Section 5 concludes the paper.

2 Interval-valued Long-term Cognitive Networks

The recently proposed IVLTCN model [5] is a recurrent neural network to deal with uncertainty evidenced with interval grey numbers. In this knowledge-based reasoning system, weights and the neuron's activation values are expressed as interval-valued grey numbers. Each problem variable is mapped into a grey neural concept, thus explicit hidden neurons are not allowed. The iterative reasoning process of the IVLTCN model is formalized below:

$$a_i^{\pm(t+1)}(k) = f_i^{\pm(t+1)}\left(\sum_{j=1}^{M} w_{ji}^{\pm} a_j^{\pm(t)}(k)\right) \tag{1}$$

where

$$f_i^{\pm(t)}(x) = L_i^{\pm} + \frac{U_i^{\pm} - L_i^{\pm}}{1 + e^{-\lambda_i^{\pm(t)}(x - h_i^{\pm(t)})}} \tag{2}$$

stands for the grey transfer function associated with the i-th neuron in the t-th iteration, M is the number of grey neurons, and $a_i^{\pm(t+1)}(k)$ represents the grey activation value for a given initial condition k. Moreover, $w_{ji}^{\pm}$ is the grey weight connecting two neurons, while $\lambda_i^{\pm(t)}$ and $h_i^{\pm(t)}$ are the parameters of the grey sigmoid function. The $\lambda_i^{\pm(t)}$ parameter is the function slope and $h_i^{\pm(t)}$ stand for the sigmoid offset. These parameters will be adjusted during the nonsynaptic learning phase, similar to a standard backpropagation algorithm [16] [15].

The parameters $L_i^{\pm}$ and $U_i^{\pm}$ are two white numbers that denote the lower and upper limits for the activation value of each neuron. These parameters are not optimized during the learning phase. Instead, they should be configured by experts taking into account the problem domain.

The IVLTCN model takes from LTCNs model [13] the structure and learning algorithm, and from the Grey System Theory (GST) [4] the grey arithmetic operations to perform the neural reasoning process. According to GST, an interval grey number can be denoted as $a^{\pm} \in [a^-, a^+] \mid a^- \leqslant a^+$ where (a^+) is the upper limit and (a^-) is the lower limit [17]. If the grey number $a^{\pm}$ only has an upper limit, then it is denoted by $a^{\pm} \in (-\infty, a^+]$. If the grey number only has a lower limit, then it is denoted by $a^{\pm} \in [a^-, +\infty)$. If both limits are unknown, then $a^{\pm} \in (-\infty, +\infty)$ is a black number. Finally, it is said that the number is white if both limits have the same value [10], that is to say, $a^- = a^+$.

Let $a^{\pm}$ and $b^{\pm}$ be two interval grey numbers. The arithmetic operations for these numbers are formalized as follows:

$$a^{\pm} + b^{\pm} \in [a^- + b^-, a^+ + b^+] \tag{3}$$

$$a^{\pm} - b^{\pm} \in [a^- - b^+, a^+ - b^-] \tag{4}$$

$$\begin{aligned} a^{\pm} \times b^{\pm} \in [&\min\{a^- \times b^-, a^+ \times b^+, a^- \times b^+, a^+ \times b^-\}, \\ &\max\{a^- \times b^-, a^+ \times b^+, a^- \times b^+, a^+ \times b^-\}] \end{aligned} \tag{5}$$

$$\begin{aligned} \frac{a^{\pm}}{b^{\pm}} \in [\min\{\frac{a^-}{b^-}, \frac{a^+}{b^+}, \frac{a^-}{b^+}, \frac{a^+}{b^-}\}, \max\{\frac{a^-}{b^-}, \frac{a^+}{b^+}, \frac{a^-}{b^+}, \frac{a^+}{b^-}\}] \\ \mid b^-, b^+ \neq 0 \end{aligned} \tag{6}$$

$$\begin{aligned} (a^{\pm})^x \in [&\min\{(a^-)^x, (a^+)^x\}, \max\{(a^-)^x, (a^+)^x\}] \\ &= [(a^-)^x, (a^+)^x]. \end{aligned} \tag{7}$$

Next, it will be describe the IV-NSBP learning algorithm [5], which is devoted to fine-tuning $\lambda_i^{\pm(t)}$ and $h_i^{\pm(t)}$ in Eq. (2). The first step in that regard is to formalize the error function as follows:

$$\mathcal{E}^{\pm} = \sum_{i=1}^{M} \frac{(y_i^{\pm}(k) - a_i^{\pm(t)})^2}{2} \tag{8}$$

where $y_i^{\pm}(k)$ is the value of the i-th variable for the k-th instance.

After computing $\partial \mathcal{E}^{\pm}/\partial a_i^{\pm(t)}(k)$ (see details in [5]), the partial derivative of the global error with respect to the target parameters can be calculated. Let $\Theta_i^{\pm(t)} = \{\lambda_i^{\pm(t)}, h_i^{\pm(t)}\}$ denote the set of grey parameters to be adjusted by the i-th neuron in the t-th iteration. The partial derivative of the global error with respect to the target parameters are computed as follows:

$$\frac{\partial \mathcal{E}^{\pm}}{\partial \theta_i^{\pm(t)} \in \Theta_i^{\pm(t)}} = \frac{\partial \mathcal{E}^{\pm}}{\partial a_i^{\pm(t)}(k)} \times \frac{\partial a_i^{\pm(t)}(k)}{\partial \theta_i^{\pm(t)} \in \Theta_i^{\pm(t)}}, \tag{9}$$

such that

$$\frac{\partial a_i^{\pm(t)}}{\partial h_i^{\pm(t)}} = \frac{-(U_i^{\pm} - L_i^{\pm})\Gamma_i^{\pm(t)}\lambda_i^{\pm(t)}}{(1+\Gamma_j^{\pm(t)})^2} \tag{10}$$

$$\frac{\partial a_i^{\pm(t)}}{\partial \lambda_i^{\pm(t)}} = \frac{(U_i^{\pm} - L_i^{\pm})\Gamma_i^{\pm(t)}(\bar{a}_i{}^{\pm(t)} - h_i^{\pm(t)})}{(1+\Gamma_j^{\pm(t)})^2}. \tag{11}$$

Equation (12) update the sigmoid function parameters associated to each neural processing entity in the t-th abstract layer. The momentum is represented by β and η is the learning rates,

$$\nabla \theta_i^{\pm(t)} \in \Theta_i^{\pm(t)} = \beta \left(\nabla \theta_i^{\pm(t)} \in \Theta_i^{\pm(t)} \right) - \eta \times \frac{\partial \mathcal{E}^{\pm}}{\partial \theta_i^{\pm(t)} \in \Theta_i^{\pm(t)}}. \tag{12}$$

The parameters' update was done using grey arithmetic operations instead of the standard vector-wise operations. This is required because the gradient vector is composed of grey numbers.

3 Skipped Nonsynaptic Backpropagation

In this section, we modify three versions of the nonsynaptic learning variants: Random Nonsynaptic Backpropagation (R-NSBP), Skipped Nonsynaptic Backpropagation (S-NSBP), and Random-Skipped Nonsynaptic Backpropagation (RS-NSBP) published in [11]. Those variants emerged because the NSBP backpropagation learning algorithm could fail when dealing with very long dependencies since the error signal reaching the first abstract layers might be weak [11]. Therefore, their authors proposed three strategies to modify the NSBP's backward step and prevent the network from stopping its learning. The nonsynaptic learning variants are based on the idea that employing the same weights during the forward and backward passes is not necessary to train a recurrent neural network. The results reported in [9] and [1] support this idea. The difference among the variants is given by the approach used to compute the partial derivatives of the error for the activation values of the abstract hidden neurons. This research proposed three versions that bring the advantage of being prepared to work in uncertain environments. Besides, the inference process in these new learning algorithms is done without a whitenization process in any way, and the learnable parameters to be optimized are just two; $\lambda_i^{\pm(t)}$ and $h_i^{\pm(t)}$.

3.1 Interval-value Random NSBP Algorithm, IVR-NSBP

Our first version is based on the Random NSBP (R-NSBP) introduced at [11] where the weight matrix in the backward pass is replaced with a matrix comprised of normally distributed random numbers. Equations (13) and (14) shows how to compute partial derivative of the total error with respect to the neuron's activation value in the current iteration, for the R-NSBP and the method proposed in this subsection IVR-NSBP;

$$\frac{\partial \mathcal{E}}{\partial a_i^{(t)}} = \sum_{j=1}^{M} \frac{\partial \mathcal{E}}{\partial a_j^{(t+1)}} \times \frac{\partial a_j^{(t+1)}}{\partial \bar{a}_j^{(t+1)}} \times \bar{w}_{ij}. \tag{13}$$

$$\frac{\partial \mathcal{E}^{\pm}}{\partial a_i^{\pm(t)}} = \sum_{j=1}^{M} \frac{\partial \mathcal{E}^{\pm}}{\partial a_j^{\pm(t+1)}} \times \frac{\partial a_j^{\pm(t+1)}}{\partial \bar{a}_j^{\pm(t+1)}} \times \bar{w}_{ij}^{\pm}. \tag{14}$$

where $\bar{w}_{ij}^{\pm}$ is the Gaussian random number generated with the following probability distribution function:

$$f(x|\mu_{ij}^{\pm}, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} * e^{-\frac{(x-\mu_{ij}^{\pm})^2}{2\sigma^2}} \tag{15}$$

where $\mu_{ij}^{\pm} = \bar{w}_{ij}^{\pm}$ denotes the mean and $\sigma^2 = 0.2$ represents the variance. The random weights can be forced to share the same sign as the weights defined during the network construction step. On the other hand, weights in the forward pass and the nonsynaptic parameters are employed as indicated in the original NSBP learning algorithm.

3.2 Interval-value Skipped NSBP Algorithm, IVS-NSBP

The second proposal is a method named Interval-value Skipped NSBP algorithm (IVS-NSBP), as its predecessor S-NSBP [11] uses deep learning channel to deliver the error signal directly to the current abstract hidden layer. The partial derivative of the global error concerning the neuron's output in the contemporary abstract layer can be computed by the Eq. (16) in the method S-NSBP and by Eq. (17) in IVS-NSBP.

$$\frac{\partial \mathcal{E}}{\partial a_i^{(t)}} = \sum_{j=1}^{M} -(y_j(k) - a_j^{(T)}) \times w_{ij}. \tag{16}$$

$$\frac{\partial \mathcal{E}^{\pm}}{\partial a_i^{\pm(t)}} = \sum_{j=1}^{M} -(y_j^{\pm}(k) - a_j^{\pm(T)}) \times w_{ij}^{\pm}. \tag{17}$$

One of the points in favor of this variant is that the skipping operations reduce the algorithm's computational border since the effort of computing the error signal does not escalate with the number of IVLTCN iterations.

3.3 Interval-value Random-Skipped NSBP Algorithm, IVRS-NSBP

The Interval-value Random-Skipped NSBP algorithm (IVRS-NSBP) is a method that allows skipping operations while use random Gaussian numbers with mean $\mu_{ij}^{\pm} = w_{ij}^{\pm}$ and $\sigma^2 = 0.2$. The IVRS-NSBP is based on the RS-NSBP [11]. A slight variance was adopted to avoid weight in the deep learning channel being too different from those used during the forward pass. This idea is formalizes by the Eq. (18) in the method RS-NSBP and Eq. (19) for IVRS-NSBP.

$$\frac{\partial \mathcal{E}}{\partial a_i^{(t)}} = \sum_{j=1}^{M} -(y_j(k) - a_j^{(T)}) \times \bar{w}_{ij} \tag{18}$$

$$\frac{\partial \mathcal{E}^{\pm}}{\partial a_i^{\pm(t)}} = \sum_{j=1}^{M} -(y_j^{\pm}(k) - a_j^{\pm(T)}) \times \bar{w}_{ij}^{\pm} \tag{19}$$

In essence, the three variants are different from the ones proposed at [11] because now the map can handle a higher level of uncertainty due to perform of the inference process being in a range of grey numbers and without any whitenization process. Also, just two learnable parameters associated with the generalized sigmoid transfer function are adjusted.

4 Numerical Simulations

This section presents an experimental study to evaluate the performance of the proposed nonsynaptic learning algorithms (IVR-NSBP, IVS-NSBP and IVRS-NSBP). To the simulations, it was used 35 synthetic datasets taken from the UCI machine learning repository[1] with a number of attributes that ranges from 3 to 22, and the number of instances goes from 106 to 625. The datasets have been modified as indicated [11] and [5] to simulate uncertainty environments.

Equation (20) shows how to estimate white weights from the numeric features in a dataset comprised of K instances,

$$w_{ji} = \frac{K \sum_k x_i(k) x_j(k) - \sum_k x_i(k) \sum_k x_j(k)}{K(\sum_k x_j(k)^2) - (\sum_k x_j(k))^2} \tag{20}$$

where $x_i(k)$ is the white value of the i-th variable for the k-th instance. As a second step, it can be transform them into grey weights such that $w_{ji}^{\pm} = [w_{ji} - \xi, w_{ji} + \xi]$, where $\xi \leq 0$ is the uncertainty threshold.

[1] http://www.ics.uci.edu/~mlearn/MLRepository.html.

4.1 Effect of Uncertainty Level on the Internal Size

The first experiment analyzes the relationship between the uncertainty added to the data and the size of the grey intervals produced by grey neurons (for all neurons in each iteration steps). The following equation is used to compute the average size of the prediction intervals in an IVLTCN model,

$$S(\mathcal{N}(k)) = \frac{1}{TM}\sum_{t=1}^{T}\sum_{i=1}^{M}|a_i^{+(t)}(k) - a_i^{-(t)}(k)| \tag{21}$$

where $\mathcal{N}(k)$ represent the IVLTCN model using the k-th initial activation vector (intance), $a_i^{-(t)}(k)$ and $a_i^{+(t)}(k)$ denote the rigtht and left activation value of the i-th neuron in the current iteration, respectively.

Figure 1 shows the average interval size values across the 35 grey datasets. As we can observe in these simulations, there is a proportional relationship between the size of intervals and the amount of uncertainty (defined by the threshold parameter used to build the grey weights).

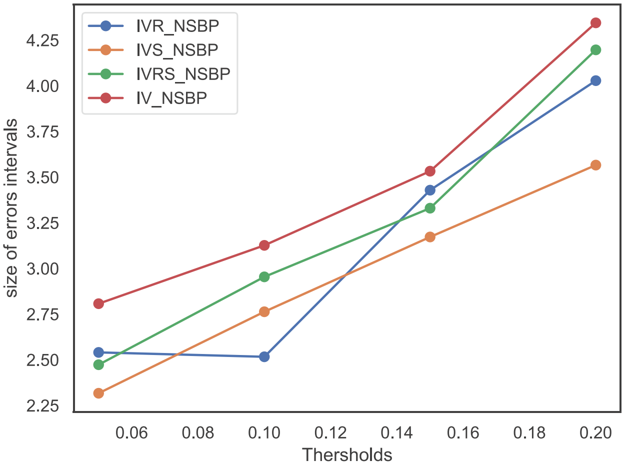


Fig. 1. Average interval size for different uncertainty levels.

Another interesting observation is that the intervals determined by the proposed nonsynaptic learning variants are smaller than those obtained by the original IV-NSBP algorithm. However, this does not imply that these learning algorithms produce more accurate models.

4.2 Assessing the Prediction Accuracy

In this subsection, it is compare the IV-NSBP learning algorithms in terms of the Grey Mean Squared Error ($MSE^{\pm}$). This performance metric is defined in terms of grey numbers and formalized as follows:

$$MSE^{\pm}(X,Y) = \frac{1}{MK} \sum_{x \in X, y \in Y} \sum_{i=1}^{M} (a_i^{\pm(T)}(x) - y_i^{\pm})^2 \tag{22}$$

where X represents the set of predicted intervals and Y is the set of original intervals, $a_i^{\pm(T)}(x)$ denotes the response of the i-th neural concept in the last iteration (i.e., the abstract output layer) for the corrupted pattern $x \in X$. Moreover, T is the number of iterations and can be equal to the maximum number of variables to be predicted in a pattern.

The configuration of the stochastic descendent gradient is as follows: momentum is set to 0.8, the learning rate is set to 0.004, and the number of epochs is set to 200. Figure 2 illustrates the median $MSE^{\pm}$ by each variant.

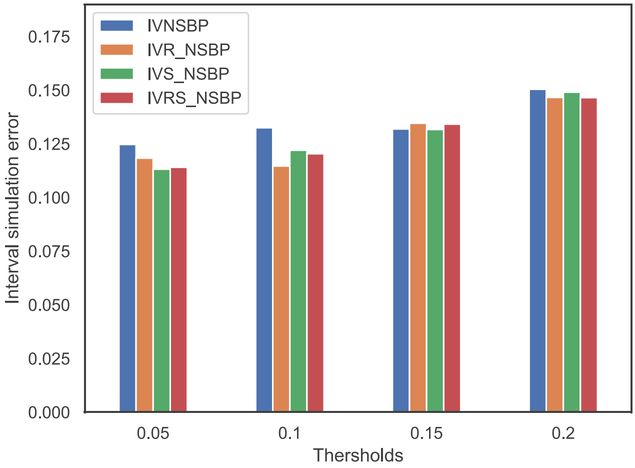


Fig. 2. Average interval simulation errors computed using different IVNSBP learning algorithms for different uncertainty levels.

The simulation results in Fig. 2 show a slight increase in the simulation errors as the uncertainty (thresholds) increases. These numbers do not reach high values because they do not exceed 0.15. It was verified the directly proportional relationship between the uncertainty and the $MSE^{\pm}$ values.

The authors in [11] found that the methods implementing the skipping operation performed slightly better than the NSBP algorithm. The best MSE values computed by each one of those variants are as follows: NSBP = 0.0569, R-NSBP = 0.0582, S-NSBP = 0.0558 and RS-NSBP = 0.0556.

Following this line of experimental study, Fig. 3 shows the lowest simulation error computed by each nonsynaptic learning variant: (a) IV-NSBP, (b) IVR-NSBP, (c) IVS-NSBP, and (d) IVRS-NSBP for different uncertainty level. The $MSE^{\pm}$ calculated with the variants proposed in this paper are as good and sometimes better (threshold = 0.05) than those obtained in the previous variants. Only when $\xi \geq 0.20$, we can observe a tendency for the error intervals to increase, which is to be expected.

As we can see in Fig. 3, the IVS-NSBP learning algorithm yields the best results. Another interesting conclusion is that the $MSE^{\pm}$ values obtained for uncertainty levels of 0.05 and 0.10 are lower than those obtained by the nonsynaptic learning variants proposed in [11].

Friedman test was used [6] to determine whether the performance differences are statistically significant or not. The p-values are 2.15E−01, 3.23E−01, 8.73E−01 and 3.42E−02, respectively, for a confidence interval is 95%. The Wilcoxon signed-rank test is used to perform pairwise comparisons. The p-values for the four uncertainty levels are displayed in Tables 1, 2, 3 and 4, using the IV-NSBP algorithm as the control method. Besides, we report the p-values computed by the Holm post-hoc procedure [7], the negative ranks (R^-), the positive ranks (R^+), and whether the null hypothesis H_0 is rejected or not.

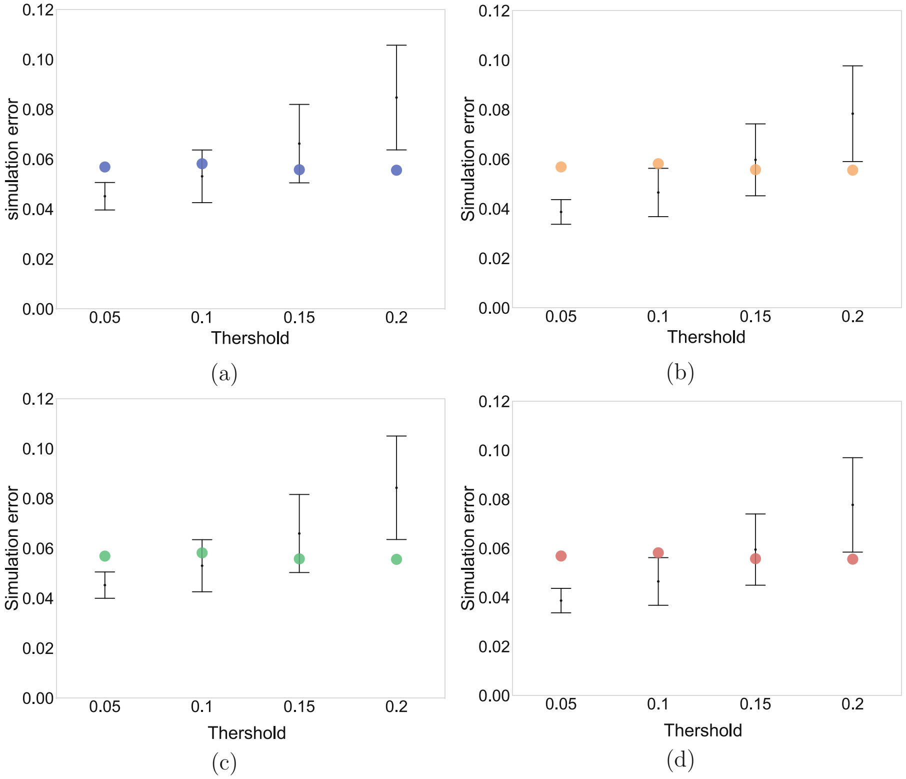
Table 1. Pairwise analysis for $\xi = 0.05$

Algorithm	p-value	R^-	R^+	Holm	H_0
IVR-NSBP	1.36E−01	13	22	2.72E−01	Fail to reject
IVS-NSBP	4.92E−01	16	19	4.92E−01	Fail to reject
IVRS-NSBP	7.97E−02	11	24	2.39E−01	Fail to reject

Table 2. Pairwise analysis for $\xi = 0.10$

Algorithm	p-value	R^-	R^+	Holm	H_0
IVR-NSBP	5.74E−02	14	21	1.72E−01	Fail to reject
IVS-NSBP	6.58E−01	15	20	6.58E−01	Fail to reject
IVRS-NSBP	2.52E−01	16	19	5.03E−01	Fail to reject

Tables 1, 2, 3 and 4 show that there are no significant differences between the algorithms. However, the rankings indicate that, as the uncertainty increases, the IVS-NSBP method performs slightly better than the other learning algorithms. This result agrees with the conclusions in [11], thus verifying that backpropagating the error signal through the inner (abstract) layers is an expendable operation to deliver the error to the first hidden layer.



(a)

(b)

(c)

(d)

Fig. 3. Simulation errors for the LTCN model (colored dots) and the IVLTCN model (intervals) for different uncertainty levels. The learning algorithms are (a) IV-NSBP, (b) IVR-NSBP, (c) IVS-NSBP, and (d) IVRS- NSBP.

Table 3. Pairwise analysis for $\xi = 0.15$

Algorithm	p-value	R^-	R^+	Holm	H_0
IVR-NSBP	8.06E−01	16	19	1.00E+00	Fail to reject
IVS-NSBP	6.58E−01	14	21	1.00E+00	Fail to reject
IVRS-NSBP	8.83E−01	20	15	1.00E+00	Fail to reject

Table 4. Pairwise analysis for $\xi = 0.20$

Algorithm	p-value	R^-	R^+	Holm	H_0
IVR-NSBP	5.53E−02	14	21	1.11E−01	Fail to reject
IVS-NSBP	2.18E−02	9	26	6.55E−02	Fail to reject
IVRS-NSBP	6.42E−02	15	20	1.11E−01	Fail to reject

5 Conclusions

This paper presented three learning algorithm variants for training the IVLTCN model, which operates with interval numbers. The median of the errors computed by these three variants (IVR-NSBP, IVS-NSBP, and IVRS-NSBP) does not exceed 0.15. Therefore, the proposals have an effectiveness of more than 85% of the cases in uncertain environments. In the comparative analysis between the best results of the predecessor models and our proposals, it is observed that the performance of the new variants was similar or slightly better. In other words, the simulation errors were as low as those obtained by the predecessor models and sometimes lower. It is worth highlighting that this result has a more significant impact because it was achieved in uncertainty environments (less information in the datasets) which shows that the new variants are as powerful as their predecessors. More importantly, the experiments confirmed that backpropagating the error signal through the abstract layers described by interval weights is not needed to train the network effectively.

It would be interesting to complement this research with new experimental studies such as the determination of the dispersion of $\lambda_i^{\pm(t)}$ and $h_i^{\pm(t)}$, parameters adjusted during the Nonsynaptic Backpropagation algorithms and considered key parameters on the algorithm's approximation ability. Also the model could be applied in real case studies, for example, in analyzing the incidence of features that determine the level of service at intersections without traffic lights. Usually, these studies are very tedious because of the field and cabinet work needed to collect the accurate values of the main factors. So getting a tool that handles data sets with uncertainty and allows valid conclusions to be drawn will shorten the measurement times of the variables and can form the basis for more in-depth traffic studies, with the same reliability as traditional procedures and incredible speed.

References

1. Baldi, P., Sadowski, P., Lu, Z.: Learning in the machine: random backpropagation and the deep learning channel. Artif. Intell. **260**, 1–35 (2018). https://doi.org/10.1016/j.artint.2018.03.003, https://www.sciencedirect.com/science/article/pii/S0004370218300985
2. Bengio, Y., Simard, P., Frasconi, P.: Learning long-term dependencies with gradient descent is difficult. IEEE Trans. Neural Netw. **5**(2), 157–166 (1994). https://doi.org/10.1109/72.279181
3. Concepción, L., Nápoles, G., Falcon, R., Vanhoof, K., Bello Perez, R.: Unveiling the dynamic behavior of fuzzy cognitive maps. IEEE Trans. Fuzzy Syst. **29**(5), 1252–1261 (2021). https://doi.org/10.1109/TFUZZ.2020.2973853
4. Deng, J.: Introduction to grey system theory. Grey Syst. **1**(1), 1–24 (1989)
5. Frias, M., Nápoles, G., Vanhoof, K., Filiberto, Y., Bello, R.: Nonsynaptic backpropagation learning of interval-valued long-term cognitive networks. In: 2021 International Joint Conference on Neural Networks (IJCNN), pp. 1–9 (2021). https://doi.org/10.1109/IJCNN52387.2021.9533586

6. Friedman, M.: The use of ranks to avoid the assumption of normality implicit in the analysis of variance. J. Am. Stat. Assoc. **32**(200), 675–701 (1937)
7. Holm, S.: A simple sequentially rejective multiple test procedure. Scand. J. Stat. **6**, 65–70 (1979)
8. Kosko, B.: Fuzzy cognitive maps. Int. J. Man-Mach. Stud. **24**(1), 65–75 (1986)
9. Lillicrap, T., Cownden, D., Tweed, D., Akerman, C.: Random synaptic feedback weights support error backpropagation for deep learning. Nat. Commun. **7**, 13276 (2016). https://doi.org/10.1038/ncomms13276
10. Liu, S., Forrest, J.: Grey Information: Theory and Practical Applications. Springer, London (2006). https://doi.org/10.1007/1-84628-342-6
11. Nápoles, G., Grau, I., Concepción, L., Salgueiro, Y.: On the performance of the nonsynaptic backpropagation for training long-term cognitive networks. In: 11th International Conference of Pattern Recognition Systems (ICPRS 2021), vol. 2021, pp. 25–30 (2021). https://doi.org/10.1049/icp.2021.1434
12. Nápoles, G., Salmeron, J., Vanhoof, K.: Construction and supervised learning of long-term grey cognitive networks. IEEE Trans. Cybern. (2019). https://doi.org/10.1109/TCYB.2019.2913960
13. Nápoles, G., Vanhoenshoven, F., Falcon, R., Vanhoof, K.: Nonsynaptic error backpropagation in long-term cognitive networks. IEEE Trans. Neural Netw. Learn. Syst. (2019). https://doi.org/10.1109/TNNLS.2019.2910555
14. Nápoles, G., Vanhoenshoven, F., Vanhoof, K.: Short-term cognitive networks, flexible reasoning and nonsynaptic learning. Neural Netw. (2019). https://doi.org/10.1016/j.neunet.2019.03.012
15. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by backpropagating errors. In: Neurocomputing: Foundations of Research, pp. 696–699. MIT Press, Cambridge (1988)
16. Werbos, P.J.: Beyond regression: new tools for prediction and analysis in the behavioral sciences. Ph.D. thesis, Harvard University (1974)
17. Yang, Y., John, R.: Grey sets and greyness. Inf. Sci. **185**, 249–264 (2012). https://doi.org/10.1016/j.ins.2011.09.029