Indonesian Journal of Electrical Engineering and Computer Science
Vol. 8, No. 3, December 2017, pp. 779 ~ 786
DOI: 10.11591/ijeecs.v8.i3.pp779-786

Modified BPNN via Iterated Least Median Squares, Particle Swarm Optimization and Firefly Algorithm

Nor Azura Md. Ghani*[1], Saadi bin Ahmad Kamaruddin[2], Norazan Mohamed Ramli[3], Ismail Musirin[4], Hishamuddin Hashim[5]
[1, 3] Center for Statistical Studies and Decision Sciences, Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Selangor Darul Ehsan, Malaysia

International Islamic University Malaysia
[2] Kulliyyah of Science, International Islamic University Malaysia, Pahang Darul Makmur, Malaysia
[4] Faculty of Electrical Engineering, Universiti Teknologi MARA, Shah Alam, Selangor Darul Ehsan, Malaysia
[5] Pilihsepakat Sdn. Bhd., Lot 5431, Tingkat 1(B), Jalan J9, Fasa 6, Taman Melawati, Kuala Lumpur, Malaysia
*Corresponding authors, e-mail: azura@tmsk.uitm.edu.my[1], mysaadiak@iium.edu.my[2], norazan@tmsk.uitm.edu[3], ismailbm@salam.uitm.edu.my4, pssbf6@gmail.com[5]

Abstract

There is doubtlessly manufactured artificial neural system (ANN) is a standout amongst the most acclaimed all-inclusive approximators, and has been executed in numerous fields. This is because of its capacity to naturally take in any example with no earlier suppositions and loss of all inclusive statement. ANNs have contributed fundamentally towards time arrangement expectation field, yet the nearness of exceptions that normally happen in the time arrangement information may dirty the system preparing information. Hypothetically, the most widely recognized calculation to prepare the system is the backpropagation (BP) calculation which depends on the minimization of the common ordinary least squares (OLS) estimator as far as mean squared error (MSE). Be that as it may, this calculation is not absolutely strong within the sight of exceptions and may bring about the bogus forecast of future qualities. Accordingly, in this paper, we actualize another calculation which exploits firefly calculation on the minimal middle of squares (FA-LMedS) estimator for manufactured neural system nonlinear autoregressive (BPNN-NAR) and counterfeit neural system nonlinear autoregressive moving normal (BPNN-NARMA) models to cook the different degrees of remote issue in time arrangement information. In addition, the execution of the proposed powerful estimator with correlation with the first MSE and strong iterative slightest middle squares (ILMedS) and molecule swarm advancement on minimum middle squares (PSO-LMedS) estimators utilizing reenactment information, in light of root mean squared blunder (RMSE) are likewise talked about in this paper. It was found that the robustified backpropagation learning calculation utilizing FA-LMedS beat the first and other powerful estimators of ILMedS and PSO-LMedS. As a conclusion, developmental calculations beat the first MSE mistake capacity in giving hearty preparing of counterfeit neural systems.

Keywords*: Anomalies; time series, learning algorithm, robust estimators, evolutionary algorithms.*

1. Introduction

The backpropagation calculation depends on the feedforward multilayer neural system for an arrangement of inputs with determined known orders. The calculation permits multilayer feedforward neural systems to take in info yield mappings from preparing tests [1]. Once every section of the specimen set is displayed to the system, its yield reaction will be analyzed by the system as for the example information design. The yield reaction is then contrasted with the known and sought yield and the blundering quality is figured, where the association weights are balanced. The backpropagation calculation depends on Widrow-Hoff delta learning principle in which the weight change is done through mean square error (MSE) of the yield reaction to the example info [2]. The arrangement of these specimen examples is over and again introduced to the system until the mistake quality is minimized. Despite the fact that ANNs have effectively caught the premium and worry of numerous specialists in numerous fields because of its

Received August 8, 2017; Revised October 26, 2017; Accepted November 13, 2017

widespread capacity as capacity approximator, the notable backpropagation learning calculation which depends on the minimization of the mean square mistake (MSE) cost capacity, is not vigorous within the sight of anomalies that may bring about blunder in information preparing process [3]. MSE is a mistake measure between the genuine and craved yield that is utilized as a part of the mainstream backpropagation learning calculation of multilayered feedforward neural systems (MFNNs) preparing. [3] Concur that this prominent calculation is not totally strong within the sight of exceptions. Indeed, even a solitary anomaly can destroy the whole neural system fit [4].

For all intents and purposes, acquiring great information is the most entangled a portion of estimating [5]. In this way, achieving complete and smooth genuine information are right around zero likelihood. Exceptions are information seriously going amiss from the example set of the dominant part information. It has been accounted for that the event of anomalies reaches about 1% to more than 10% in normal routine information [6] [7]. In view of past studies [8-10] the presence of these exceptions represents an extreme danger to the standard or customary minimum squares investigation.

In time arrangement investigation, the examiners need to depend on information to recognize which point in time are exceptions to gauge the proper remedial moves to be made so that the distorted occasions can be assessed precisely. Hypothesis and practice are greater part worried with direct techniques, such ARMA and ARIMA models [11]. Be that as it may, numerous arrangement show design which cannot be clarified by a linear system which triggers the need for non-direct models, for instance bilinear models [12] and non-straight ARMA models (NARMA) [13].

2. Material and method

In this examination, there were three distinctive reenactment information were utilized. Foundation Noise Data focuses were chosen aimlessly and after that substituted with likelihood δ with a foundation commotion consistently appropriated in the particular race.

Case 1- With a specific end goal to test our calculation on the 1-D approximation task, the capacity by [8] was considered in this examination, as likewise utilized by past works, for example, [14-17].

$$y = |x|^{-\frac{2}{3}} \tag{1}$$

The facts utilized because that looking after incorporate over N=400 focuses to that amount have been constructed by using trying out the autonomous variable within the scope on [-2, 2] with meantime 0.01. Durability.

Case 2-Another 1-D capability to lie approximated was once as regarded of much articles [14-18] characterised as:

$$y = \frac{\sin(x)}{x} \tag{2}$$

The records utilized for it evaluation comprise concerning N=1500 focuses that were built by inspecting the self sustaining variable into the scope on [-7.5, 7.5] along interim 0.01. Case 3-The 2nd calculation was once as proposed through [9][14] as do lie characterized as:

$$y = x_1 e^{-x_1^2 - x_2^2} \tag{3}$$

The records focuses have been committed by using tested capability of the honor 16 x 16 framework. Here, the informational collection used to be manufactured through checking out the fair factors, x1, x2 [-2, 2] including meantime 0.01. In the investigation flowchart of mass 4, the experiment method execute stand honestly observed.

Here, the current rigid estimators regarding backpropagation neural skeleton had been completed. To reply the middle goal of the investigation, the possible butter strenuous estimators of nonlinear autoregressive (NAR) yet nonlinear autoregressive transferring regular (NARMA) regarding the neural fabric time course regarding labor were completed the usage of MATLAB.

At this progression, MATLAB scripts or codings were composed parallel to the scientific plan done before. After that, the execution of the proposed robustified neural system models was thought about utilizing recreation information; 1-D and 2-D utilizing the standard execution measure, root mean square mistake (RMSE). At that point the powerful BPNN-NAR and BPNN-NARMA technique were tried on benchmark information. The similar results have attracted those strides.

3. Results and discussions

In view of the Tables 1, 2 and 3, the conventional calculation delivered the best results in light of the littlest RMSE values for the perfect information without exceptions. This result is parallel with the case that the MSE mistake capacity is ideal for the information without anomalies by [3][14][19].

In any case, the circumstance is changed since the information containing misleadingly produced anomalies where the MSE-based strategy totally loses its productivity. This can be demonstrated by the breakdowns of the technique as appeared in Tables 1, 2 and 3. Every single hearty calculation of ILMedS, PSO-LMedS and FA-LMedS perform fundamentally better contrasted with MSE-based cost capacity. Additionally, by simply looking at the two developmental improvement calculations, FA-LMedS has indicated more prominent execution with the most minimal RMSE values in every one of the four cases, as appeared in Tables 1, 2 and 3. This is maybe an immediate aftereffect of the reason that firefly estimation perform better for a greater measure of disturbance [20] and when it joined into a backpropagation neural system, preparing calculation, it unites at a quicker rate with least feedforward neural system plan [21]. As specified by [22], the fireflies' calculation is the exceptional instance of quickened molecule swarm enhancement calculation. Besides, contrasted with FA-LMedS and ILMedS, the PSO-LMedS calculation has a tendency to accomplish more noteworthy RSME as the rate of exceptions increment. In this situation, PSO-LMedS was seen to deliver more noteworthy blunders for the information comprising exceptions more than 60 percent, as appeared in Tables 1, 2 and 3. Here, it is trusted that the PSO-LMedS may perform better on the off chance that we expand the quantity of swarm size and cycle esteem. The modified neural network can be further implemented to solve many other real-life problems, such as image processing [23], water treatment plant [24] and power plant [25].

Acknowledgement

We might want to devote our gratefulness and appreciation to Unit Kerjasama Awam Swasta (UKAS) of Prime Minister's Department, Construction Industry Development Board (CIDB) and the Malaysian Statistics Department. Extraordinary thanks additionally go to Universiti Teknologi MARA and Malaysian Ministry of Higher Education (MOHE) for supporting this examination under the Research Grant No. 600-RMI/DANA 5/3/CIFI (65/2013) and No. 600-RMI/FRGS 5/3 (137/2014).

References

[1] P. Sibi, S. A. Jones, and P. Siddarth. Analysis Of Different Activation Functions Using Back Propagation Neural Networks. *Journal Of Theoretical And Applied Information Technology*. 2013; 47(3); 1264-1268.
[2] A. P. Engelbrecht. Computational intelligence: An introduction. *Wiley.com*. 2007.
[3] M. T. El-Melegy, M. H. Essai and A. A. Ali. Robust training of artificial feedforward neural networks. *Found Comput Intell 1*. 2009; 1: 217–242.
[4] Hector, M. Claudio, and S. Rodrigo. Robust Estimator for the Learning Process in Neural Network Applied in Time Series. *Artificial Neural Networks-ICANN 2002*. 2002; 1080-1086.
[5] L. J. Gitman, and C. McDaniel. The future of business, The essentials. *South-Western Pub*, 2008.

[6] Z. Zhang. Parameter estimation techniques: A tutorial with application to conic fitting. *Image and Vision Computing Journal*. 1997; 15(1); 59-76.
[7] R. K. Pearson. Anomalies in process modeling and identification. *IEEE Transactions on Control Systems Technology*. 2002; 10(1): 55-63.
[8] K. Liano. Robust error measure for supervised neural network learning with anomalies. *IEEE Trans Neural Networks*. 1996; 7(1): 246-250.
[9] P. J. Huber. *Robust Statistics*. John Wiley and Sons. New York, 1981.
[10] P. J. Rousseeuw, and A. M. Leroy. *Robust regression and outlier detection*. Wiley, New York, 1987
[11] G. E. Box, G. M. Jenkins and G. C. Reinsel. Time series analysis, forecasting and control. *Ed. Prentice Hall*, 1994.
[12] M. M. Gabr. Robust estimation of bilinear time series models. *Comm. Statist, Theory and Meth*. 1998; 27(1): 41-53.
[13] J. T. Connor and R. D. Martin. Recurrent neural networks and robust time series prediction. *IEEE Trans of Neural Networks*. 1994; 5(2): 240-253.
[14] A. Rusiecki. Robust Learning Algorithm Based on Iterative Least Median of Squares. *Neural Process Lett.* 2012; 36: 145-160.
[15] D. S. Chen and R. C. Jain. A robust back propagation learning algorithm for function approximation. *IEEE Trans. Neural Networks*. 1994; 5(3): 467–479.
[16] Chuang, S. Su, and C. Hsiao. The annealing robust backpropagation (ARBP) learning algorithm. *IEEE Trans Neural Networks*. 2000; 11(5): 1067–1076.
[17] A. L. Rusiecki. Fault tolerant feedforward neural network with median neuron input function. *Electronics Letters*. 2005; 41(10): 603-605.
[18] C. Chuang C, J. T. Jeng and P. T. Lin. Annealing robust radial basis function networks for function approximation with anomalies. *Neurocomputing*. 2004; 56: 123–139.
[19] G. Beliakov, A. Kelarev, and J. Yearwood. Derivative-free optimization and neural networks for robust regression. *Optimization.* 2012; 61(12): 1467-1490.
[20] S. K. Pal, C. S. Rai, and A. P. Singh. Comparative Study of Firefly Algorithm and Particle Swarm Optimization for Noisy Non-Linear Optimization Problems. *International Journal of Intelligent Systems and Applications*. 2012; 10: 50-57.
[21] S. Nandy, P. P. Sarkar and A. Das. Analysis of a Nature Inspired Firefly Algorithm based Back-propagation Neural Network Training. *International Journal of Computer Applications*. 2012; 42(22): 8-16.
[22] X. S. Yang. *Firefly Algorithms for Multimodal Optimization.* Proceeding of the 5th Symposium on Stochastic Algorithms, Foundations and Applications. 2009; 5792; 169-178.
[23] B. Saichandana, K. Srinivas and R. KiranKumar, "Image fusion in hyperspectral image classification using genetic algorithm", *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 2, no. 3, 703-711, 2017.
[24] M. S. Gaya, L. A.Yusuf, M. Mustapha, B. Muhammad, A. Sani, A. T. A. Tijjani, and M. T. M. Khairi. Estimation of Turbidity in Water Treatment Plant using Hammerstein-Wiener and Neural Network Technique. *Indonesian Journal of Electrical Engineering and Computer Science*. 2017; 5(3): 666-672.
[25] S. Chakraborty, P. K. Sadhu and N. Pal. A New Approach towards Ideal Location Selection for PV Power Plant in India. *Indonesian Journal of Electrical Engineering and Computer Science.* 2014; 12(11): 7681-7689.

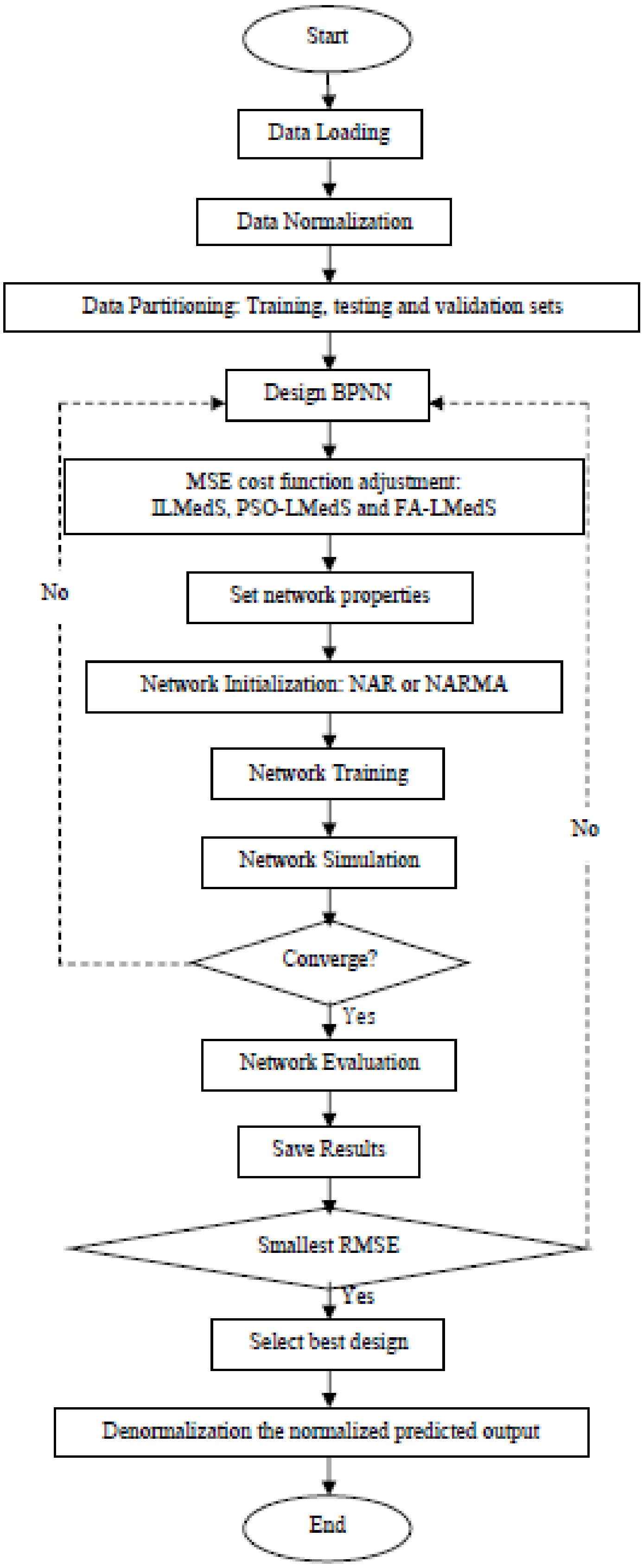

Figure 1. Process of the proposed BPNN-NAR and BPNN-NARMA

Table 1. The RMSE scores for test function in case

Input Lags (Ny)	Error Lags (Ne)	Hidden	Swarm Size	Iteration	Percentage of Anomalies (δ)	Algorithm MSE		ILMedS		PSO-LMedS		FA-LMedS	
						NAR	NARMA	NAR	NARMA	NAR	NARMA	NAR	NARMA
2	2	5	5	5	0	0.0165	0.0144	0.0575	0.0967	0.0070	0.0024	0.0006	0.0006
2	2	10	5	5	0	0.0317	0.0232	0.0428	0.0460	0.0094	0.0088	0.0009	0.0009
2	2	20	5	5	0	0.0559	0.0373	0.0190	0.0242	0.0095	0.0035	0.0010	0.0009
2	2	5	5	5	10	**2537.2571**	**2243.7403**	0.2666	0.2622	0.0096	0.0088	0.0073	0.0055
2	2	10	5	5	10	**7806.4086**	**6968.4181**	0.2729	0.2827	0.0076	0.0085	0.0040	0.0048
2	2	20	5	5	10	**3197.1237**	**2696.4526**	0.2304	0.3171	0.0061	0.0071	0.0024	0.0013
2	2	5	5	5	20	**2691.9232**	**2483.4234**	0.4367	0.4806	0.0274	0.0128	0.0005	0.0069
2	2	10	5	5	20	**2681.6996**	**1561.1672**	0.1847	0.1636	0.0236	0.0153	0.0041	0.0073
2	2	20	5	5	20	**3837.5206**	**3669.8813**	0.2282	0.2405	0.0291	0.0286	0.0092	0.0026
2	2	5	5	5	30	**3436.0919**	**4352.2928**	0.2321	0.2266	0.0376	0.0294	0.0003	0.0050
2	2	10	5	5	30	**3483.9997**	**3468.8221**	0.2042	0.1561	0.0321	0.2293	0.0047	0.0056
2	2	20	5	5	30	**8194.4084**	**9399.368**	0.2379	0.2445	0.0530	0.3392	0.0068	0.0071
2	2	5	5	5	40	**4842.7778**	**3768.5752**	0.2070	0.2182	0.0271	0.0250	0.0060	0.0024
2	2	10	5	5	40	**12529.7165**	**12785.6565**	0.1596	0.1331	0.1574	0.0971	0.0042	0.0018
2	2	20	5	5	40	**2616.636**	**2040.8494**	0.1534	0.1704	0.0213	0.0060	0.0023	0.0020
2	2	5	5	5	50	**4613.7007**	**4797.0081**	0.1421	0.1756	0.0503	0.0276	0.0028	0.0070
2	2	10	5	5	50	**6696.0732**	**7111.9116**	0.1207	0.1631	0.0688	0.0193	0.0071	0.0073
2	2	20	5	5	50	**5870.884**	**7450.8175**	0.1590	0.1733	0.0799	0.011	0.0007	0.0028
2	2	5	5	5	60	**6304.1263**	**5309.1228**	0.1658	0.1702	10.0524	7.0512	0.0037	0.0034
2	2	10	5	5	60	**15210.3341**	**19075.8857**	0.1980	0.1629	3.1016	5.6931	0.0005	0.0039
2	2	20	5	5	60	**2440.4538**	**3412.5697**	0.1611	0.1740	6.7627	8.3582	0.0266	0.092
3	3	5	5	5	0	0.2266	0.9105	0.0730	0.0137	0.0598	0.0061	0.0331	0.0381
3	3	10	5	5	0	0.1888	0.5777	0.0946	0.0687	0.0476	0.0737	0.0704	0.0363
3	3	20	5	5	0	0.8679	0.3433	0.0920	0.0176	0.0177	0.0954	0.0756	0.0810
3	3	5	5	5	10	**2422.4903**	**2354.6119**	0.1868	0.1870	0.0290	0.0106	0.0631	0.0243
3	3	10	5	5	10	**2961.5131**	**2019.4265**	0.1657	0.1843	0.0308	0.0471	0.0733	0.0626
3	3	20	5	5	10	**1703.9431**	**1457.9506**	0.2092	0.2251	0.0266	0.0561	0.0702	0.0752
3	3	5	5	5	20	**2782.5779**	**2530.3214**	0.2185	0.2076	0.0301	0.0203	0.0629	0.0421
3	3	10	5	5	20	**4023.1339**	**2749.3138**	0.2966	0.2736	0.0289	0.0742	0.0740	0.0395
3	3	20	5	5	20	**2388.1344**	**3923.2193**	0.1268	0.1406	0.0265	0.0052	0.0390	0.0408
3	3	5	5	5	30	**2340.4295**	**2597.6883**	0.1562	0.1727	0.0302	0.0089	0.0995	0.2822
3	3	10	5	5	30	**2565.6173**	**2779.8586**	0.2117	0.1821	0.0193	0.0020	0.0981	0.0690
3	3	20	5	5	30	**8138.588**	**10017.3766**	0.1410	0.1479	0.0897	0.0312	0.0137	0.0487
3	3	5	5	5	40	**3929.4474**	**4579.1315**	0.3575	0.3967	0.0392	0.0628	0.0687	0.0609
3	3	10	5	5	40	**3069.3175**	**2993.2578**	0.1428	0.1460	0.0456	0.0206	0.0176	0.0983
3	3	20	5	5	40	**3252.8795**	**3788.1158**	0.2190	0.2242	0.0284	0.0468	0.0870	0.0507
3	3	5	5	5	50	**4103.3899**	**3686.9169**	0.2666	0.2622	0.0473	0.0135	0.0843	0.0731
3	3	10	5	5	50	**3213.4135**	**3489.5679**	0.2729	0.2827	0.0349	0.0253	0.0251	0.0554
3	3	20	5	5	50	**2344.8786**	**2746.5734**	0.2304	0.3171	0.0339	0.0704	0.0076	0.0121
3	3	5	5	5	60	**4979.5091**	**4377.9622**	0.4367	0.4806	5.0661	3.5519	0.0736	0.0320
3	3	10	5	5	60	**4662.0489**	**47763.665**	0.1847	0.1636	4.3144	3.4804	0.0452	0.0295
3	3	20	5	5	60	**7100.6318**	**8801.7551**	0.2282	0.2405	7.4872	6.3311	0.0989	0.0926

Table 2. The RMSE scores for test function in case 2

Max Lags of inputs (Ny)	Max Lags of errors (Ne)	Hidden	Swarm Size	Iteration	Percentage of Anomalies (δ)	Algorithm MSE		ILMedS		PSO-LMedS		FA-LMedS	
						NAR	NARMA	NAR	NARMA	NAR	NARMA	NAR	NARMA
2	2	5	5	5	0	0.1288	0.1202	0.3209	0.2715	0.0305	0.0290	0.0297	0.0252
2	2	10	5	5	0	0.1782	0.1824	0.0272	0.0432	0.0716	0.0230	0.0392	0.0982
2	2	20	5	5	0	0.2366	0.2595	0.0494	0.0480	0.0225	0.0144	0.0456	0.0751
2	2	5	5	5	10	**1592.8770**	**1801.0386**	0.1236	0.1195	0.1631	0.2421	0.0284	0.0803
2	2	10	5	5	10	**2793.9951**	**2639.77614**	0.1136	0.1076	0.2785	0.3694	0.0473	0.0025
2	2	20	5	5	10	**1788.0502**	**1642.0878**	0.1378	0.1389	0.1734	0.2030	0.0349	0.0092
2	2	5	5	5	20	**1640.7081**	**1575.8881**	0.3490	0.2692	0.1657	0.1801	0.0339	0.0031
2	2	10	5	5	20	**1637.5895**	**1249.4667**	0.0446	0.0410	0.1538	0.1564	0.0506	0.0231
2	2	20	5	5	20	**1958.9590**	**1915.6934**	0.0806	0.0623	0.1705	0.2592	0.0431	0.0159
2	2	5	5	5	30	**1853.6698**	**2086.2149**	0.1412	0.1008	0.1941	0.2523	0.0748	0.0664
2	2	10	5	5	30	**1866.5475**	**1862.4774**	0.0836	0.0333	0.1792	0.2159	0.0197	0.0036
2	2	20	5	5	30	**2862.5877**	**3065.8388**	0.1255	0.0923	0.2302	0.3396	0.0298	0.0016
2	2	5	5	5	40	**2200.6312**	**1941.2818**	0.0933	0.1088	0.1647	0.2346	0.0331	0.0074
2	2	10	5	5	40	**3539.7339**	**3575.7036**	0.0379	0.0247	0.3967	0.5664	0.0218	0.0069
2	2	20	5	5	40	**1617.6019**	**1428.5830**	0.0454	0.0391	0.1460	0.1649	0.1915	0.1705
2	2	5	5	5	50	**2147.9526**	**2190.2073**	0.0407	0.0501	0.2242	0.2224	0.2086	0.1941
2	2	10	5	5	50	**2587.6771**	**2666.8167**	0.0224	0.0277	0.2622	0.3515	0.1862	0.1792
2	2	20	5	5	50	**2422.9907**	**2729.6185**	0.0605	0.0558	0.2827	0.3371	0.3065	0.2302
2	2	5	5	5	60	**2510.8019**	**2304.1533**	0.0779	0.0760	31.7163	37.1251	0.1941	0.1647
2	2	10	5	5	60	**3900.0428**	**4367.5949**	0.0526	0.0480	4.8062	5.9087	0.3575	0.0967
2	2	20	5	5	60	**1562.1951**	**1847.3141**	0.0371	0.0639	1.6367	2.1119	0.1428	0.1400
3	3	5	5	5	0	0.2388	0.2379	0.0527	0.0578	0.0445	0.0343	0.2304	0.0171
3	3	10	5	5	0	0.1973	0.2071	0.0693	0.0500	0.0182	0.0055	0.4367	0.4006
3	3	20	5	5	0	0.1596	0.1596	0.0890	0.1178	0.1331	0.1847	0.1847	0.1636
3	3	5	5	5	10	**1556.4351**	**1534.4744**	0.0496	0.0694	0.2704	0.2132	0.2282	0.2005
3	3	10	5	5	10	**1720.9047**	**1421.0652**	0.0572	0.0803	0.1756	0.1018	0.2321	0.0266
3	3	20	5	5	10	**1305.3517**	**1207.4562**	0.0714	0.0873	0.1631	0.1497	0.2042	0.0561
3	3	5	5	5	20	**1668.1068**	**1590.6983**	0.1202	0.0940	0.1733	0.2461	0.2379	0.0445
3	3	10	5	5	20	**2005.7751**	**1658.1054**	0.1271	0.1724	0.1702	0.2791	0.2070	0.0182
3	3	20	5	5	20	**1545.3589**	**1980.7118**	0.0234	0.0224	0.1629	0.2295	0.1596	0.0331
3	3	5	5	5	30	**1529.8462**	**1611.7345**	0.0301	0.0413	0.1740	0.1926	0.1534	0.0704
3	3	10	5	5	30	**1601.7544**	**1667.2908**	0.0877	0.0620	0.1390	0.1959	0.1421	0.0756
3	3	20	5	5	30	**2852.8210**	**3165.0239**	0.0241	0.0295	0.2995	0.5362	0.1207	0.0631
3	3	5	5	5	40	**1982.2833**	**2139.8905**	0.3209	0.2715	0.1981	0.3002	0.1590	0.0733
3	3	10	5	5	40	**1751.9467**	**1730.1034**	0.0272	0.0432	0.2137	0.0296	0.1658	0.0702
3	3	20	5	5	40	**18035.7409**	**19463.0826**	0.0494	0.0480	0.1687	0.0633	0.1980	0.0629
3	3	5	5	5	50	**20256.8258**	**19201.3461**	0.1236	0.1195	0.2176	0.0983	0.1611	0.0740
3	3	10	5	5	50	**17925.996**	**18680.385**	0.1136	0.1076	0.1870	0.0228	0.1667	0.0390
3	3	20	5	5	50	**1531.2996**	**1657.2788**	0.1378	0.1389	0.1843	0.0393	0.3165	0.0995
3	3	5	5	5	60	**2231.4813**	**2092.3580**	0.3490	0.2692	22.5109	20.7358	0.2139	0.0981
3	3	10	5	5	60	**2159.1776**	**2185.4899**	0.0446	0.0410	2.07621	2.4675	0.1730	0.0137
3	3	20	5	5	60	**2664.7010**	**2966.7752**	0.0806	0.0623	2.7367	2.5661	0.1946	0.0687

Table 3. The RMSE scores for test function in case 3

Max Lags of inputs (Ny)	Max Lags of errors (Ne)	Hidden	Swarm Size	Iteration	Percentage of Anomalies (δ)	Algorithm MSE		ILMedS		PSO-LMedS		FA-LMedS	
						NAR	NARMA	NAR	NARMA	NAR	NARMA	NAR	NARMA
2	2	5	5	5	0	0.0226	0.0252	0.1801	0.1933	0.0267	0.1636	0.0371	0.0224
2	2	10	5	5	0	0.0436	0.0497	0.1564	0.1718	0.0434	0.1184	0.0307	0.0244
2	2	20	5	5	0	0.1112	0.1186	0.2592	0.2266	0.1061	0.1258	0.0569	0.0672
2	2	5	5	5	10	**4359.7566**	**8641.6380**	0.2523	0.2763	0.0365	0.1912	0.057	0.0636
2	2	10	5	5	10	**1079.2603**	**1365.1487**	0.2159	0.2381	0.1014	0.1184	0.0525	0.0466
2	2	20	5	5	10	**3988.2749**	**4121.7564**	0.3396	0.2974	0.0388	0.1970	0.1150	0.0153
2	2	5	5	5	20	**3717.8782**	**3244.3472**	0.2346	0.2201	0.0373	0.1933	0.0756	0.0550
2	2	10	5	5	20	**3073.1547**	**2446.1292**	0.5664	0.5211	0.0295	0.1718	0.2471	0.0209
2	2	20	5	5	20	**5694.1254**	**6723.3495**	0.1649	0.2080	0.0513	0.1266	0.0424	0.0272
2	2	5	5	5	30	**5730.2224**	**6368.4176**	0.2224	0.2191	0.0763	0.1763	0.0517	0.0494
2	2	10	5	5	30	**5256.0255**	**4664.3883**	0.3515	0.3456	0.0566	0.1381	0.1019	0.0236
2	2	20	5	5	30	**11503.8636**	**11533.4735**	0.3371	0.3280	0.0884	0.1974	0.0969	0.0136
2	2	5	5	5	40	**7562.8682**	**5504.5465**	0.3712	0.3726	0.0484	0.1201	0.1372	0.0378
2	2	10	5	5	40	**2471.0467**	**3209.0136**	0.5908	0.5189	0.2715	0.1211	0.1727	0.0490
2	2	20	5	5	40	**4243.4113**	**2721.2212**	0.2111	0.2025	0.0432	0.1080	0.0336	0.0046
2	2	5	5	5	50	**5179.6874**	**4948.2608**	0.2839	0.2496	0.0480	0.1191	0.0696	0.0075
2	2	10	5	5	50	**10196.5565**	**12361.8442**	0.3758	0.3175	0.1195	0.1456	0.0269	0.0201
2	2	20	5	5	50	**9697.4329**	**11364.5493**	0.2892	0.1827	0.1076	0.1280	0.0248	0.0174
2	2	5	5	5	60	**13726.6642**	**13780.5056**	0.3543	0.3038	13.8992	17.2684	0.0156	0.0136
2	2	10	5	5	60	**1727.7562**	**3490.8264**	0.3055	0.3298	2.6921	1.1894	0.0366	0.0290
2	2	20	5	5	60	3368.2453	**4460.2079**	0.1947	0.1574	4.1081	1.0266	0.0435	0.0376
3	3	5	5	5	0	0.0937	0.1255	0.2461	0.2362	0.0923	0.1138	0.1278	0.1174
3	3	10	5	5	0	0.0749	0.0933	0.2791	0.2757	0.1088	0.0298	0.0204	0.0113
3	3	20	5	5	0	0.0381	0.0379	0.2295	0.2192	0.0247	0.0574	0.0479	0.0102
3	3	5	5	5	10	3636.3138	**4549.1711**	0.1926	0.2528	0.0391	0.0977	0.0711	0.0287
3	3	10	5	5	10	**6106.0489**	**4076.2713**	0.1959	0.1776	0.0501	0.0239	0.0745	0.0399
3	3	20	5	5	10	**2437.9512**	**2242.0911**	0.5362	0.4409	0.0277	0.0664	0.1014	0.0085
3	3	5	5	5	20	**6265.6872**	**6059.5287**	0.3002	0.3139	0.0558	0.0362	0.0388	0.0097
3	3	10	5	5	20	**7522.1978**	**7795.0577**	0.2296	0.2405	0.0760	0.0757	0.0373	0.0028
3	3	20	5	5	20	**4214.6130**	**5269.1885**	0.2633	0.2238	0.0488	0.0192	0.0295	0.0053
3	3	5	5	5	30	**3956.8955**	**3710.3527**	0.2983	0.3432	0.0639	0.0528	0.0513	0.0086
3	3	10	5	5	30	**4083.2797**	**3841.2332**	0.2228	0.2635	0.0315	0.1776	0.0763	0.0303
3	3	20	5	5	30	**2822.7018**	**2875.1272**	0.2393	0.2834	0.1944	0.1409	0.0566	0.0292
3	3	5	5	5	40	**6906.748**	**9012.3312**	0.2673	0.2954	0.0985	0.1139	0.0884	0.0391
3	3	10	5	5	40	**4870.0912**	**5275.4700**	0.3467	0.3066	0.0578	0.2405	0.0484	0.0050
3	3	20	5	5	40	**6093.7314**	**6934.9149**	0.3566	0.4152	0.0500	0.2238	0.2715	0.0970
3	3	5	5	5	50	**9831.8046**	**8903.4344**	0.1529	0.1498	0.1178	0.3432	0.0432	0.0059
3	3	10	5	5	50	**5076.8150**	**4965.6695**	0.1735	0.2032	0.0694	0.0635	0.0480	0.0275
3	3	20	5	5	50	**7313.6695**	**5727.4593**	0.2962	0.2491	0.0803	0.0834	0.1195	0.0193
3	3	5	5	5	60	**55470.641**	**7144.9387**	0.1553	0.1719	8.7385	9.5403	0.1076	0.0114
3	3	10	5	5	60	**11215.7927**	**12026.7402**	0.1801	0.1933	9.4028	6.6874	0.1389	0.0704
3	3	20	5	5	60	**13201.5013**	**12719.1829**	0.1564	0.1718	1.7246	1.5251	0.2692	0.0156