

## Improving self-organizing map with nguyen-widrow initialization algorithm

Maureen Nettie N. Linan<sup>1</sup>, Bobby D. Gerardo<sup>2</sup>, Ruji P. Medina<sup>3</sup>

<sup>1,3</sup>Graduate Programs, Technological Institute of the Philippines, Philippines

<sup>2</sup>College of Information and Communications Technology, West Visayas State University, Philippines

---

### Article Info

#### Article history:

Received Nov 29, 2018

Revised Jan 21, 2019

Accepted Mar 11, 2019

---

#### Keywords:

Nguyen-widrow algorithm

Self-organizing map

Weight initialization

---

### ABSTRACT

The quality of cluster result and the learning speed of Self-organizing map (SOM) are dependent on the initialization of weights since the initial values for weight vectors affect the performance of SOM training when applied to clustering. In this paper, the improvement of SOM was achieved with the application of the Nguyen-Widrow algorithm to initialize weights. Nguyen-Widrow initialization algorithm is a method for initialization of the weights of neural networks to speed up the training process. Performance of the modified SOM was determined in terms of cluster error rate and the number of iterations to achieve convergence using different datasets and results show that the modified SOM algorithm produces better cluster results and improved training speed compared to traditional SOM.

Copyright © 2019 Institute of Advanced Engineering and Science.

All rights reserved.

---

### Corresponding Author:

Maureen Nettie N. Linan

Graduate Programs,

Technological Institute of the Philippines,

Quezon CityAurora Blvd., Quezon City, Philippines

Email: maureennettie@gmail.com

---

## 1. INTRODUCTION

Clustering is an essential analytical method in data mining wherein objects within a dataset are grouped into clusters of objects with high similarity. Among clustering algorithms, the self-organizing map (SOM) [1], an artificial neural network (ANN) model, has found wide application in industry, finance, natural sciences, and linguistics [2-4]. This is due to the ability of SOM to represent multidimensional data in much lower dimensional space-usually in one or two dimensions.

Clustering of data using the SOM algorithm is achieved mainly by two steps: 1. training the data with the initialized parameters, and 2. clustering of data. SOM training results depend on the initialization of the weight vector along with the learning parameter and training inputs. The final values of these weights are then used for clustering [5].

Clustering performance of SOM is greatly dependent on the initial weight. With real-world datasets being high-dimensional, the complexity of SOM tends to increase resulting in the increase in the training time if initialization of weights is not proper [6]. The choice of these initial values for weight vectors affects the performance of SOM training when applied to clustering. The proper approach of initialization of the weight vector can contribute significantly to the improved performance of SOM.

Commonly, weights in SOM are randomly initialized with a low, close to zero value from the range of neurons observed in the data resulting in misclustering. Random initialization has the drawback of scrabbling the neurons, thus requiring greater processing or computational time to finish the mapping satisfactorily. With this, the quality of clustering results together with learning speed is greatly affected [7-8]. Therefore, it is important to find the way in which the training of the SOM algorithm can be improved.

Approaches concerning weight initialization have been carried out using several algorithms, such as combining Kmeans++ and SOM [9], Frequency Sensitive Competitive Learning (FSCL) [5], Principal Component Analysis (PCI) [7]. Performance of these initialization techniques depends on data distribution. Dimensionality reduction was also considered using different weight initialization strategies, PCI, Kernel Principal Component Analysis (KPCI), Fuzzy C-Means and Random initialization [6]. Furthermore, the initial value of neuron's weight and the number of clusters was addressed by using Fuzzy Subtractive Clustering (FSC) with SOM [10].

Nevertheless, the Nguyen-Widrow [11] has not been implemented for weight initialization in clustering. Hence, this study apply for the first time the Nguyen-Widrow initialization to eliminate the random weight initialization of the traditional SOM. Nguyen-Widrow initialization algorithm is a weight initialization method developed by Derrick Nguyen and Bernard Widrow [11] for improving learning speed. The algorithm uses small random numbers of values which are assigned for weight initialization [12]. This method generates initial weights and bias values for a layer so that active regions of the layer neurons will be distributed evenly over input space. This speeds up the training process by setting the initial weights of the first layer so that each node is assigned its own interval at the start of the training [13-14]. When applied to optimize the backpropagation algorithm, the Nguyen-Widrow was able to produce the best accuracy level and accelerate the training process [15]. Optimal performance was achieved at the backward process by adapting the learning rate parameter capable of affecting the value of the weighting and reduced the number epoch [16]. Furthermore, when the neural network is trained with conjugate gradient training algorithm having Fletcher-Reeves update, Nguyen-Widrow algorithm converge faster and also generalize better than other weight initialization technique [17]. Nguyen-Widrow weight algorithm was also applied on image compression using multilayer feed-forward artificial neural network. All weights in the network are adjusted in an identical manner using Nguyen-Widrow, and thus prevent the error function from being reduced. The Nguyen-Widrow improved the execution training time and peak-signal-to-noise (PSNR) [18].

Hence, this study implements the Nguyen-Widrow initialization for SOM weight initialization. This is expected to achieve faster learning by reducing training time. The performance of the improved SOM is evaluated in terms of error rate and the number of iterations needed to converge to final clustering.

## 2. PROPOSED METHOD

The SOM is consists of the input layer and the output layer. The input layer is one-dimensional and each data item is associated with an n-length of vector elements, while the output layer consists of radial units typically organized in one or two-dimensions. The map, as shown in Figure 1, is an array of nodes called neurons. Each node is a vector of N weights [1]. In this study, the Nguyen-Widrow algorithm is used to initialize the weights rather than random initialization.

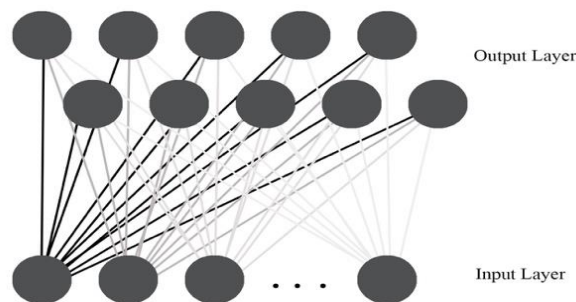


Figure 1. SOM architecture

Steps for Improved SOM algorithm:

Initialization. Initialized weights with small random numbers with the interval  $[-0.5, 0.5]$ . A magnification factor is defined by the relation:

$$\beta = 0.7 * H^{(1/n)} \quad (1)$$

where H is the number of hidden units and n is the number of inputs. The weights are randomly selected in the interval  $[-1, 1]$  and then scaled using (2):

$$w_{ij} = \beta * \frac{w_{ij}(random)}{\|w_{ij}(random)\|} \tag{2}$$

where  $w_{ij}$  is the first layer weight vector [12].

Set the learning rate ( $\alpha$ ). Randomly select the input vector.

Competition. Find the winning neurons using Euclidean function (3):

$$c = (\underset{1 < i < \text{mn}}{\text{MIN}}\{\|w_i(t) - x(t)\|\}) \tag{3}$$

where  $x(t)$  and  $w_i(t)$  are the input and weight vector of neuron  $i$  at iteration  $t$  respectively. This winner is called the best matching unit (BMU) of the input data pattern.

Updating. Select the spatial neighbors of winning neuron. Adjust the weight using (2):

$$w_i(t + 1) = w_i(t) + h_{c,i}(t)[x(t) - w_i(t)] \tag{4}$$

where  $h_{c,i}(t)$  is a Gaussian neighborhood function:

$$h_{c,i}(t) = \alpha(t) \cdot \exp\left(-\frac{\|r_c - r_i\|^2}{2\sigma^2(t)}\right) \tag{5}$$

where  $r$  is the coordinate position of the neuron on the map,  $\alpha(t)$  is the learning rate and  $\sigma(t)$  is the width of the neighborhood radius. Both  $\alpha(t)$  and  $\sigma(t)$  are decreased monotonically using the following:

$$\alpha(t) = \alpha(0) \left(\frac{\alpha(T)}{\alpha(0)}\right)^{t/T} \tag{6}$$

$$\sigma(t) = \sigma(0) \left(\frac{\sigma(T)}{\sigma(0)}\right)^{t/T} \tag{7}$$

where  $T$  is the training length. For all the input data, the same process is repeated from the random selection of input to weight adjustments [19]. Table 1 presents the proposed algorithm for improved SOM.

Table 1. SOM with Nguyen-Widrow Algorithm

ALGORITHM: SOM with Nguyen-Widrow algorithm	
1.	Initialization of weights using Nguyen-Widrow initialization algorithm
	a. Initialized all weights $w_i$ of hidden layers with random values over the interval [-0.5 to 0.5]
	b. For each hidden layer, calculate beta value $\beta$ using (1)
	c. For each synapse
	i. For each weight
	Adjust weight, $w_{ij}$ by dividing it with the norm of weight for neuron and multiplying beta value using the (2).
2.	Randomly select an input vector $x(t)$
3.	Determine the "Best Matching Unit (BMU)" $c$ , using the Euclidean distance formula (3).
4.	Update the weight vector of the neurons using (4).
5.	Repeat steps 2 to 5 for all the input data.

### 3. RESEARCH METHOD

The experiments were performed on different datasets. Results were checked for the proposed method as well as the previous methods that were used for initializing weights. Parameters considered for measuring the performance were the quality of cluster results in terms of error rate and the training speed in terms of the number of iterations.

Five datasets from UCI Machine Learning Repository [20] were used to test the algorithm: the iris, wine, seeds, diabetes, breast cancer datasets. The iris dataset consists of 150 samples belonging to one of three clusters. Each class has 50 samples with 4 attributes. On the other hand, the wine dataset contains 178 samples with 13 attributes belonging to one of three clusters. The seeds dataset has 210 instances belonging to one of the three clusters. Each instance has 7 attributes. The diabetes dataset has 768 instances with 8 attributes for each instance whereas the breast cancer dataset has 286 instances with 10 attributes. Instances in both diabetes and breast cancer datasets belong to one of the two clusters. To achieve better clustering

result and to avoid the negative effects produced by noise and outliers, all data were pre-processed using data cleaning normalization schemes. The data were normalized using the (8):

$$= \frac{X_i - \min(X_r)}{\max(X_r) - \min(X_r)} \tag{8}$$

where  $X_i$  is the raw data,  $\min(X_r)$  the smallest value in  $X_r$ , and  $\max(X_r)$  the largest value in  $X_r$ .

The SOM algorithm was performed on each dataset. 70% of the dataset was used for training and 30% for clustering. Using random value as initial weight vectors, the dataset was trained with an initial rate of 0.9 and an error limit of 0.01 was set to stop the training process. Tables 2 and 3 present the initial parameters setting for SOM clustering.

Table 2. Initial Parameters Setting for SOM (Iris, Wine, Seeds)

Parameters	Value
Initial learning rate	0.9
Initial weight vector	Random
Error threshold	0.01
Dimension	1 x 3

Table 3. Initial Parameters Setting for SOM (Diabetes, Breast Cancer)

Parameters	Value
Initial learning rate	0.9
Initial weight vector	Random
Error threshold	0.01
Dimension	1 x 2

Figures 2 and 3 shows the process flow and the sequence diagram of the SOM with random initialization respectively. The input vectors are randomly selected once weights value has been initialized. The winning neuron is computed using the Euclidean distance formula. Weights are then updated. The process is repeated for all the input vectors. The final weights generated from the training phase was used as the initial weight for clustering.

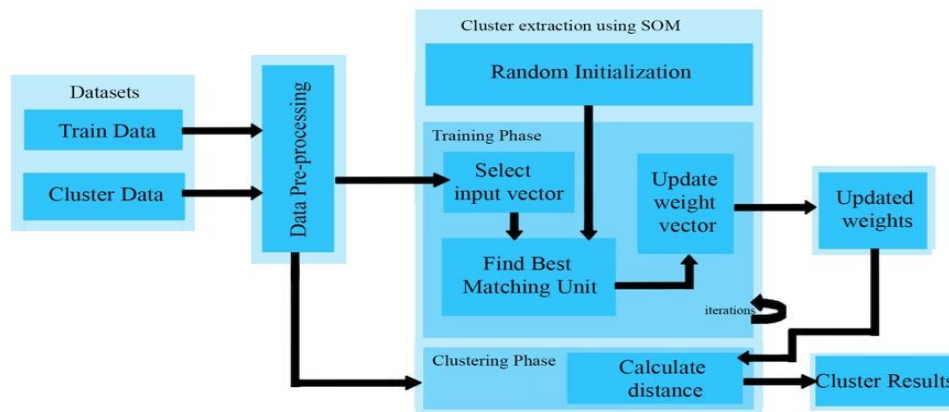


Figure 2. The process flow of SOM with random initialization

Furthermore, the improved SOM algorithm was also performed on each dataset with 70% of the dataset was used for training and 30% for clustering. Nguyen-Widrow algorithm was used to initialize the weight vector. An initial learning rate of 0.9 was set for each dataset train the data and an error limit of 0.01 to terminate the process. Initial parameters for improved SOM is presented in Tables 4 and 5.

The input vectors are randomly selected. BMU is determined using the Euclidean distance formula. Weights are updated. The process is iterative for all the input vectors. Clustering is performed using the final weights as its initial weights. The process flow and the sequence diagram of the improved SOM with Nguyen-Widrow initialization are shown in Figures 4 and 5 respectively.

Table 4. Initial Parameters Setting for Improved SOM (Iris, Wine, Seeds)

Parameters	Value
Initial learning rate	0.9
Initial weight vector	Nguyen-Widrow initialization
Error threshold	.01
Dimension	1 x 3

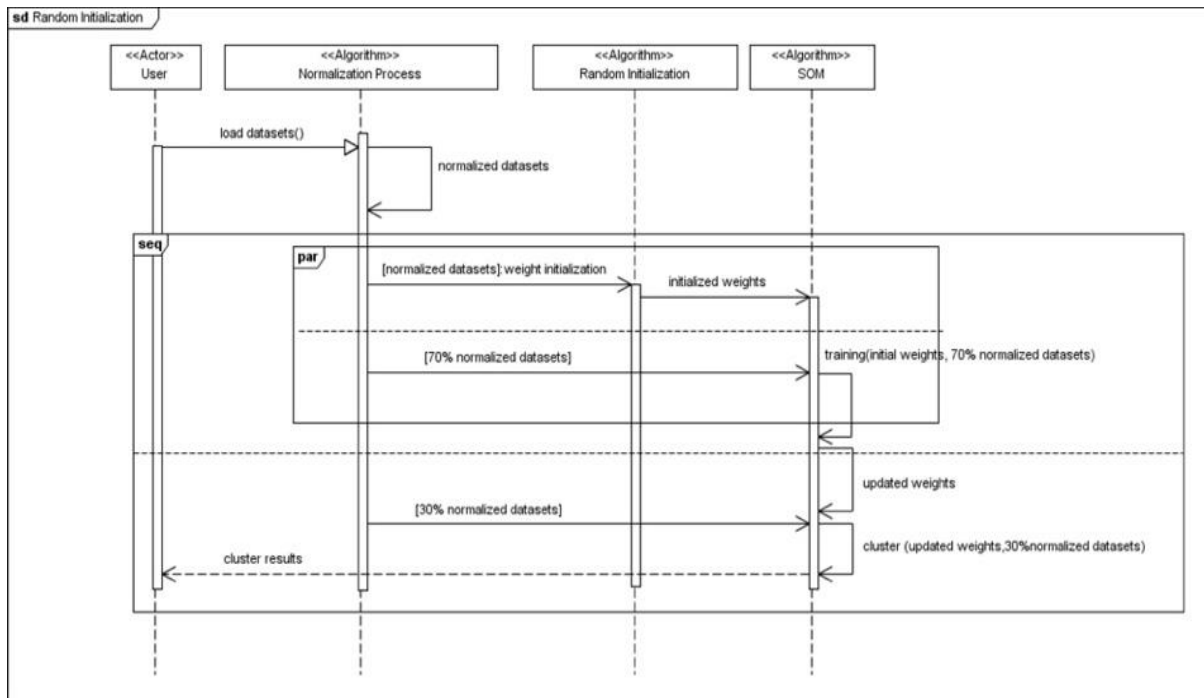


Figure 3. Sequence diagram for SOM using random initialization

Table 5. Initial Parameters Setting for Improved SOM (Diabetes, Breast Cancer)

Parameters	Value
Initial learning rate	0.9
Initial weight vector	Nguyen-Widrow initialization
Error threshold	.01
Dimension	1 x 2

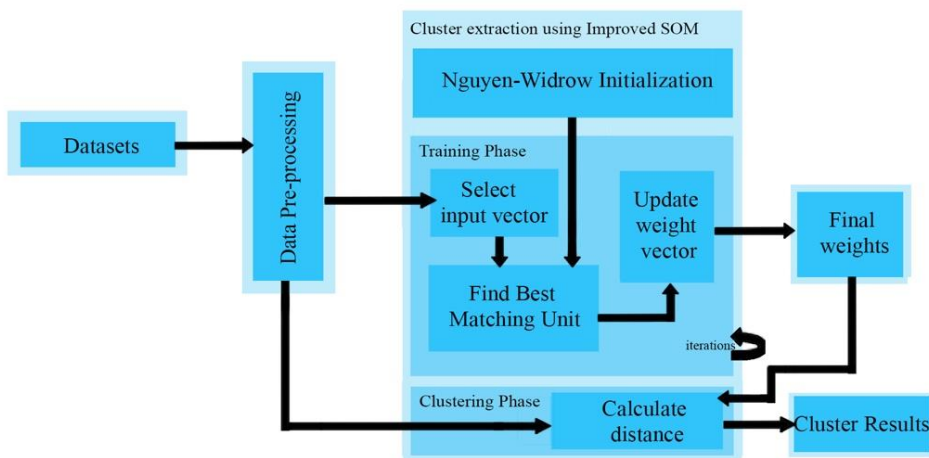


Figure 4. The process flow of the SOM with Nguyen-Widrow algorithm

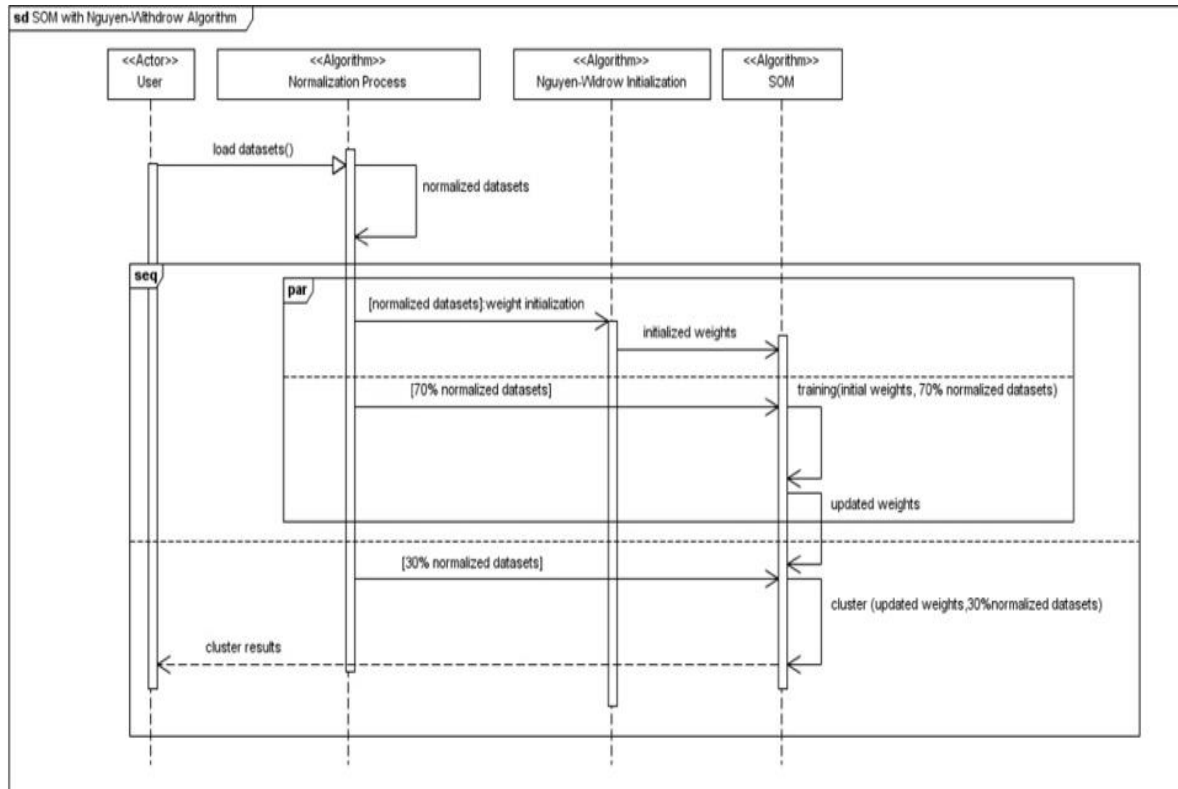


Figure 5. Sequence diagram for improved SOM using Nguyen-Widrow initialization

The performance comparison was based on the cluster error rate and the number of iterations. Error rate computation was done using (9), as follows:

$$Error\ rate\ \% = \frac{Number\ of\ incorrectly\ clustered\ data}{Total\ number\ of\ data} \tag{9}$$

Meanwhile, the number of iterations needed to achieve convergence to final clustering was noted and used as the measure for the speed of the algorithm.

#### 4. RESULTS AND ANALYSIS

The results of the study reveal that using the Nguyen-Widrow initialization algorithm for initializing weights value of SOM yields better cluster result as compared to random initialization. This is noteworthy evidence on the superiority of Nguyen-Widrow’s initialization algorithm against random initialization techniques as claimed [13].

The improved SOM using the Nguyen-Widrow initialization algorithm has improved the initial weights thus obtaining a better quality of cluster result. This further resulted in an observable decrease in error rate with an average of 23.80 %. Thus, the improved algorithm has a lower error rate as compared to the traditional SOM in all datasets that were tested. Table 6 presents the performance comparison of traditional SOM and improved SOM based on the error rate.

Table 6. Performance Comparison Of SOM and Improved SOM Based on the Error Rate

Datasets	SOM	Improved SOM
Iris	0.08890	0.06668
Seeds	0.07937	0.05952
Wine	0.09250	0.06938
Diabetes	0.30303	0.23636
Breast Cancer	0.27907	0.21767

Furthermore, a reduction in the number of iterations was recorded in all datasets using SOM with Nguyen-Widrow initialization. The improved SOM performs faster since it has lesser number of iterations to converge to final clustering than traditional SOM as presented in Figure 6. A reduction of an average of 7.39% on the number of iterations was recorded using improved SOM as compared to traditional SOM thus improving the training time. This was because Nguyen-Widrow initialization is known to accelerate the training process in a neural network [14-16]. Thus, the improved SOM yields a better quality of cluster results and speeds up the training speed which led to improved processing time.

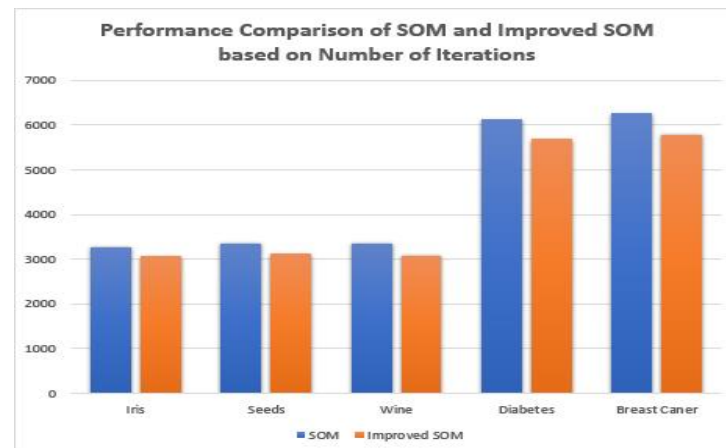


Figure 6. Performance comparison of SOM and Improved SOM based on the number of iterations

## 5. CONCLUSION

The improved SOM performs better in terms of cluster error rate and the number of iterations to converge to final clustering as compared to the traditional SOM. Using the Nguyen-Widrow algorithm for initialization of weights in the SOM thus yields better cluster result and improved training speed of the algorithm in terms of the number of iterations.

## REFERENCES

- [1] T. Kohonen, "Essentials of the Self-Organizing Map," *Neural Networks*, vol. 37, pp. 52-65, Jan. 2013.
- [2] S. A. D. Alalwan, "Diabetic Analytics: Proposed Conceptual Data Mining Approaches in Type 2 Diabetes Dataset," *Indonesian Journal of Electrical Engineering and Computer Science(IJEECS)*, vol. 14, no. 1, pp. 88-95, Apr. 2019.
- [3] G. A. V. M. Giri and A. Harjoko, "Music Recommendation System Based on Context Using Case-Based Reasoning and Self Organizing Map," *Indonesian Journal of Electrical Engineering and Computer Science(IJEECS)*, vol. 4, no. 2, pp. 459-464, Nov. 2016.
- [4] D. P. Kondisetty and M. A. Hussain, "SLIC Superpixel Based Self Organizing Maps Algorithm for Segmentation of Microarray Images," *International Journal of Advances in Applied Sciences(IJAAS)*, vol. 7, no. 1, pp. 78-85, May 2018.
- [5] V. Aggarwal, A. K. Ahlawat, and B. N. Pandey, "A Weight Initialization Approach for Training Self Organizing Maps for Clustering Applications," in *Advance Computing Conference (IACC), 2013 IEEE 3rd International*, 2013, pp. 1000-1005.
- [6] H. Haripriya, R. Devisree, D. Pooja, and P. Nedungadi, "A Comparative Performance Analysis of Self Organizing Maps on Weight Initializations using different Strategies," in *Advances in Computing and Communications (ICACC), 2015 Fifth International Conference on*, 2015, pp. 434-438.
- [7] A. A. Akinduko, E. M. Mirkes, and A. N. Gorban, "SOM: Stochastic initialization versus principal components," *Information Sciences*, vol. 364-365, pp. 213-221, Oct. 2016.
- [8] I. Valova, G. Georgiev, N. Gueorguieva, and J. Olson, "Initialization Issues in Self-Organizing Maps," *Procedia Computer Science*, vol. 20, pp. 52-57, 2013.
- [9] Y. Dogan, D. Birant, and A. Kut, "SOM++: Integration Of Self-Organizing Map and K-Means++ Algorithms," in *International Workshop on Machine Learning and Data Mining in Pattern Recognition*, 2013, pp. 246-259.
- [10] E. A. Lisangan, A. Musdholifah, and S. Hartati, "Two Level Clustering for Quality Improvement using Fuzzy Subtractive Clustering and Self-Organizing Map," *Indonesian Journal of Electrical Engineering and Computer Science(IJEECS)*, vol. 15, no. 2, pp. 373-380, Aug. 2015.

- [11] D. Nguyen and B. Widrow, "Improving the Learning Speed of 2-Layer Neural Networks by Choosing Initial Values of the Adaptive Weights," in *Neural Networks, 1990., 1990 IJCNN International Joint Conference on*, 1990, pp. 21-26.
- [12] S. P. Adam, D. A. Karras, G. D. Magoulas, and M. N. Vrahatis, "Solving the Linear Interval Tolerance Problem for Weight Initialization of Neural Networks," *Neural Networks*, vol. 54, pp. 17-37, Jun. 2014.
- [13] A. Pavelka and A. Procházka, "Algorithms for Initialization of Neural Network Weights," in *In Proceedings of the 12th Annual Conference, MATLAB*, 2004, pp. 453-459.
- [14] S. Aisyah, M. Harahap, A. M. Husein Siregar, and M. Turnip, "Optimization of Training Backpropagation Algorithm Using Nguyen Widrow for Angina Ludwig Diagnosis," *Journal of Physics: Conference Series*, vol. 1007, p. 012050, Apr. 2018.
- [15] R. Sofiana and S. Sutikno, "Optimization of Backpropagation for Early Detection of Diabetes Mellitu," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 5, pp. 3232-3237, Oct. 2018.
- [16] U. Andayani, E. B. Nababan, B. Siregar, M. A. Muchtar, T. H. Nasution, and I. Siregar, "Optimization Backpropagation Algorithm Based on Nguyen-Widrom Adaptive Weight and Adaptive Learning Rate," in *2017 4th International Conference on Industrial Engineering and Applications (ICIEA)*, 2017, pp. 363-367.
- [17] S. Masood, M. N. Doja, and P. Chandra, "Analysis of Weight Initialization Routines for Conjugate Gradient Training Algorithm with Fletcher-Reeves Updates," in *2016 International Conference on Computing, Communication and Automation (ICCCA)*, 2016, pp. 304-308.
- [18] K. Mishra, N. K. Mittal, and M. H. Mirja, "Image Compression Using Multilayer Feed Forward Artificial Neural Network with Nguyen Widrow Weight Initialization Method," *International Journal of Emerging Technology and Advanced Engineering*, vol. 4, no. 4, 2014.
- [19] V. Chaudhary, R. S. Bhatia, and A. K. Ahlawat, "A novel Self-Organizing Map (SOM) learning algorithm with nearest and farthest neurons," *Alexandria Engineering Journal*, vol. 53, no. 4, pp. 827-831, 2014.
- [20] "UCI Machine Learning Repository." [Online]. Available: <http://archive.ics.uci.edu/ml/index.php>. [Accessed: 08-Jul-2018].

## BIOGRAPHIES OF AUTHORS



**Maureen Nettie N. Linan** is a CHED-FDP II scholar currently taking her Doctor in Information Technology degree at the Technological Institute of the Philippines, Quezon City. She received her Bachelor's degree in Information and Computer Science from the Cebu Institute of Technology University, Philippines and her Masters in Information Systems from the University of the Philippines Open University as CHED-MIS scholar. Presently, she is an Associate Professor at the Iloilo Science and Technology University, Iloilo City, Philippines. Her research interest includes database management, software engineering, and data mining.



**Bobby D. Gerardo** Professor VI of the College of ICT, is currently the Vice President for Administration and Finance of West Visayas State University, Iloilo City, Philippines. His dissertation is on Discovering Driving Patterns using Rule-based intelligent Data Mining Agent (RiDAMA) in Distributed Insurance Telematic Systems. He has published more than 100 research papers in national and international journals and conferences. He is a referee to international conferences and journals such as in IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Knowledge and Data Engineering, Elsevier Journal on Telematics, Future Generation Computer Systems and on Bioinformatics. His research fields are in the area of distributed systems, telematics systems, CORBA, data mining, web services, ubiquitous computing and mobile communications.



**Ruji P. Medina** is Dean of the Graduate Programs of the Technological Institute of the Philippines in Quezon City. He holds a Ph.D. in Environmental Engineering from the University of the Philippines with sandwich program at the University of Houston, Texas where he worked on the synthesis of nanocomposite materials. He finished his MS in Environmental Engineering from the Mapúa Institute of Technology, graduating Summa Cum Laude. He obtained his Bachelor's degree in Chemical Engineering from the University of the Philippines in Diliman, Quezon City. His research interests include urban mining, electronic wastes, and nanomaterials. He counts among his expertise environmental modeling and mathematical modeling using multivariate analysis.