Appropriate Learning Rate and Neighborhood Function of Self-organizing Map (SOM) for Specific Humidity Pattern Classification over Southern Thailand

W. Natita, W. Wiboonsak, and S. Dusadee

***Abstract*—In this study, the patterns of specific humidity at 850 hPa during southwest monsoon over southern Thailand between the years 2000 to 2009 are investigated by self-organizing map (SOM). Learning rates and neighborhood functions are necessary parameters that influence the results. Bubble and Gaussian neighborhood functions and three learning rates (linear, inverse of time and power series) are analyzed by varying iteration. The quality of SOM is measured by the quantization error. The study finds that Gaussian function with linear learning rate gives the best result for pattern classification of the specific humidity.**

***Index Terms*—Learning rates, neighborhood functions, self-organizing map (SOM), specific humidity.**

I. Introduction

One of the most important things for human activities such as agriculture and transportation is climate. Climate is defined as an area's long term weather patterns. The simplest way to describe climate is to look at average temperature and precipitation over time [1]. The climate of southern Thailand is under the influence of southwest monsoon and northeast monsoon. The southwest monsoon is between mid-May and mid-October while the northeast monsoon is between mid-October and mid-February [2]. There are various climate variables that impact the monsoon but specific humidity is one of the most important variables that characterize the weather of maritime southern Thailand.

Self-Organizing Map (SOM) is introduced by Kohonen in 1989. It is an unsupervised learning algorithm that reduces the dimension of large data sets by grouping and organizing them into a two-dimensional array [3]. SOM is often used in the fields of data compression and pattern recognition and also in meteorology to study weather pattern. The categorize of the weather by SOM are applied in many researches. SOM is applied in [4] to identify sea level pressure patterns over the period 1961-1999 for study precipitation over Greenland. SOM is used in [5] to identify wind patterns over the Ross Ice Shelf, Antarctica, to study types, frequency and seasonality of these patterns. In [6], SOM and hierarchical ascendant classification are used to define the main synoptic weather regimes relevant to understanding the daily variability of rainfall over Senegal. However, the results obtained from SOM are sensitive to learning rate and neighbourhood function [7].

This research aims to find the proper learning rate and neighborhood function for 850 hectopascal (hPa) specific humidity pattern during summer monsoon over southern Thailand between the years 2000 and 2009. Two neighborhood functions (bubble and Gaussian) and three learning rates (inverse of time, linear and power series) are investigated. The learning rate is changed by iteration in the learning process. The quantization error is estimated for the quality of SOM.

II. Data

Specific humidity at 850 hPa from Climate Forecast System Reanalysis (CFSR) is used. CFSR is developed at the Environmental Modelling Center at National Centers for Environmental Prediction or NCEP [8]. The specific humidity data at 00UTC over the southernmost of Thailand during southwest monsoon (15 May to 15 October) over the years 2000 to 2009 are selected. The study domain covers latitude 5.5˚N-7˚N and longitude 99.5˚E-102˚E, as shown in Fig. 1.

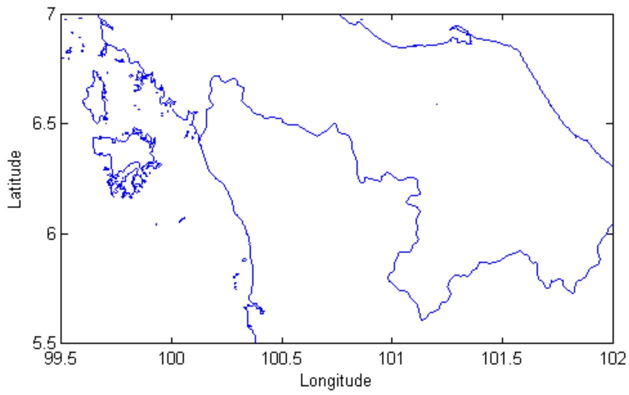


Fig. 1. The study domain.

Before training by using SOM procedure, the data have to be normalized to reduce the discrepancy in the magnitude of the input vectors. Data normalization is calculated from

$$v_{normalized} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{1}$$

Manuscript received December 23, 2015; revised February 25, 2016. This work was supported in part by Human Resource Development in Science Project (Science Achievement Scholarship of Thailand, SAST) and King Mongkut's University of Technology Thonburi.

W. Natita and W. Wiboonsak are with the Mathematics Department, King Mongkut's University of Technology Thonburi, Bangkok, Thailand (e-mail: natita.wang@gmail.com, iwibhayu@kmutt.ac.th).

S. Dusadee is with the Joint Graduate School of Energy and Environment, King Mongkut's University of Technology Thonburi, Bangkok, Thailand (e-mail: dusadee.suk@kmutt.ac.th).

DOI: 10.7763/IJMO.2016.V6.504

where x_{max} represents the maximum value of specific humidity at 850 hPa and x_{min} represents the minimum value of specific humidity at 850 hPa [9].

III. Self-Organizing Map (SOM)

A. SOM Procedure

SOM is a popular algorithm in neural network which is based on unsupervised learning. The structure of SOM is a single feedforward where each input node is connected to all output neurons. SOM is a concept of competition network that tries to find the most similar distance between the input vector and neuron with weight vector, $\boldsymbol{w}_i$. SOM always consist of both input vector, $\boldsymbol{x}$ and output vector, $\boldsymbol{y}$ with the architecture as shown in Fig. 2.

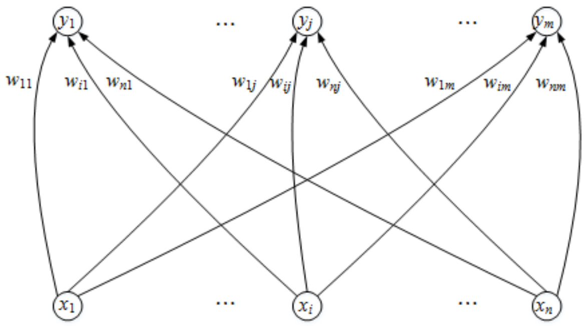


Fig. 2. The architecture of SOM [8].

At the start of the learning, all the weights (w_i) are initialized to small random numbers. The set of weights forms a vector $\boldsymbol{w}_i = w_{ij}, i = 1,2,...,k_x, j = 1,2,...,k_y$ where k_x is the number of row and k_y is the number of column. Euclidian distance, d between the input vector, $\boldsymbol{x}$ and the neuron with weight vector of the given neuron, $\boldsymbol{w}_c$ is computed by

$$d(x,w) = \left\| x(t) - w_c(t) \right\| \tag{2}$$

Next, SOM algorithm will search for the winner neuron using the minimum distance (best matching unit, BMU). BMU is calculated as follows

$$BMU = \arg\min \left\| \boldsymbol{x}(t) - \boldsymbol{w}_i(t) \right\| \tag{3}$$

To increase the similarity with the input vector, weights are adjusted after obtaining the winning neuron. The rule for updating the weight vector is given by

$$\boldsymbol{w}_i(t+1) = \boldsymbol{w}_i(t) + \alpha(t) h_{ij}^c(t)\left(\boldsymbol{x}(t) - \boldsymbol{w}_i(t)\right) \tag{4}$$

where $\alpha(t)$ is a learning rate, $h_{ij}^c(t)$ is a neighbourhood function and t is the order number of a current iteration.

For all cases of analysis, it is under the condition $\eta_{ij}^c \le \lfloor \alpha \max(k_x k_y), 1 \rfloor$. The learning rates and neighborhood functions can be applied in various ways. However, they should be decreasing functions [10].

B. Learning Rates

The learning rate is a training parameter that controls the size of weight vector in learning of SOM. There are many learning rate functions while linear, inverse of time and power series are mostly used in SOM [11]. Linear, inverse of time and power series are defined in (5), (6) and (7), respectively.

$$\alpha(t) = \alpha(0) \cdot \frac{1}{t} \tag{5}$$

$$\alpha(t,T) = \alpha(0) \cdot \left(1 - \frac{t}{T}\right) \tag{6}$$

$$\alpha(t,T) = \alpha(0) \cdot e^{\frac{t}{T}} \tag{7}$$

Here T is the number of iterations and t is the order number of a current iteration [7].

C. Neighborhood Functions

Neighborhood function determines the rate of change of the neighborhood around the winner neuron. Neighborhood function influences the training result of SOM procedure. Therefore, it is important to choose the proper neighborhood function with the data set. Same as learning rate, there are many functions but bubble and Gaussian are widely used in SOM. Bubble function is a constant function while Gaussian function is decreasing function in the defined neighborhood of the winner neuron. Bubble and Gaussian are defined in (8) and (9), respectively.

$$h_{ij}^c = \begin{cases} \alpha(t), (i,j) \in N_c \\ 0, (i,j) \notin N_c \end{cases}, \tag{8}$$

$$h_{ij}^c = \alpha(t) \cdot e^{\left(\frac{-\left\| R_c - R_{ij} \right\|^2}{2(\eta_{ij}^c(t))^2}\right)} \tag{9}$$

Here N_c is the index set of neighbor nodes around the node with indices c. The parameter η_{ij}^c represents the neighbor rank between nodes w_c and w_{ij} (the radius of neighborhood which is determined the number of neighborhood for SOM procedure). Two-dimensional vectors R_c and R_{ij} include indexes of w_c and w_{ij} (number of rows and columns) [7]. For this study, the radius of neighborhood is defined by an exponential decay function as follows

$$\eta_{ij}^c(t) = \eta(0) \cdot e^{\left(-\frac{t}{T}\right)} \tag{10}$$

where $\eta(0)$ is the initial value of the radius of the neighbourhood [9].

D. Quantization Error

Quantization error (QE) is a technique to measure the quality of SOM. It is computed from the average distance of input vectors, $\boldsymbol{x}$ to the weight vector on the winner node $\left(w_i^c\right)$ of the BMU. A SOM with lower average error is more accurate than a SOM with higher average error [12]. Quantization error is calculated by

$$QE = \frac{\sum_{i=1}^{N} \left\| \boldsymbol{x}_i - w_i^c \right\|}{N} \tag{11}$$

where N is the number of input vectors used to train the map.

E. Margin of Error

The margin of error (ME) is an estimate of a confidence interval for a given measurement or result which is computed by

$$ME = z \cdot \frac{\sigma}{\sqrt{N}} \tag{12}$$

where z is the confidence multiplier and σ is the standard deviation of the output vector [13].

In this study, 95% level of confidence is used for all experiments. Therefore, the percent of ME should be ranged in 3% [14].

IV. Experimental Design

Specific humidity at 850 hPa data is used as the input vector for SOM. It includes 154 days (15 May to 15 October) over the period from the years 2000 to 2009. Thus, the total number of day, i, is 1540. The input vectors x_{ij} are formed where $i = 1, 2, ..., 1540$ and $j = 1, 2, ..., 24$. Twenty four j represent the number of grid point from CFSR in the study domain. In this study, six experiments are investigated to find the appropriate neighborhood function and learning rate. The experiments include bubble and Gaussian functions with three learning rates (linear, inverse of time and power series). All initial parameters of SOM are shown in Table I.

TABLE I: The Initial Parameters Setting of SOM

Parameters	Value
Initial learning rate	0.85
Initial weight vector	random
Maximum radius of neighborhood	Size 5
Maximum number of iteration	1,000 to 10,000, with the step 1,000
SOM array size	3×3

The steps in SOM algorithm to find the appropriate case of learning rate and neighborhood function are described as follows.

Step 1: The input vector is in form of matrix with the size of 24×1540. So, the input matrix of each day from 15th May to 15th October between the years 2000 to 2009 is in size of 24×10 which is defined by (13).

$$\boldsymbol{x}_i = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,10} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,10} \\ x_{3,1} & x_{3,2} & \cdots & x_{3,10} \\ \vdots & \vdots & \ddots & \vdots \\ x_{24.1} & x_{24.2} & \cdots & x_{24,10} \end{bmatrix} \tag{13}$$

The weight vector is initialized by random value in form of matrix with the size of 9×24 which are defined by (14).

$$\boldsymbol{w}_i = \begin{bmatrix} w_{1,1} \\ w_{2,1} \\ \vdots \\ w_{24,1} \end{bmatrix} \tag{14}$$

Step 2: The radius of neighborhood is calculated as in (10) except the first iteration.

Step 3: Euclidian distance between input and weight vector is computed from (2) and the winning neuron is found by using (3).

Step 4: The winner neuron in Step 3 is used to update the neighborhood function of neuron which is calculated as in (8) or (9)

Step 5: The results of neighborhood function from Step 4 are used to update the weight vector of neuron by using (4).

Step 6: learning rate function in (5), (6) or (7) is updated for next iteration.

Step 7: For others input vector, Steps 2 to 5 are repeated.

Step 8: The quantization error is computed as in (11).

V. Results

The results from six experiment cases of two neighborhood functions and three learning rates to classify specific humidity pattern are shown in Table II, Table III and Fig. 3.

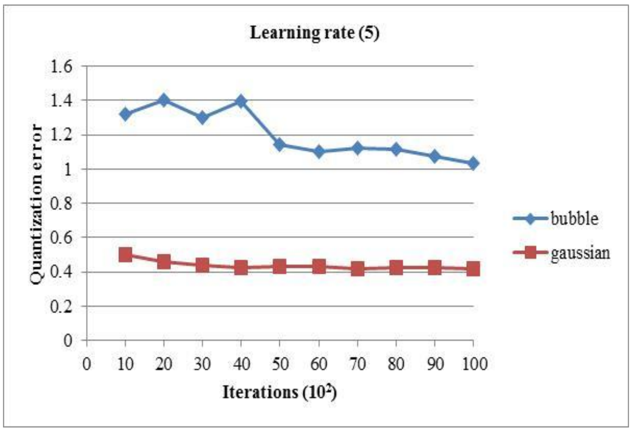
TABLE II: The Average of Margins of Errors of Specific Humidity with Confidence Probability 0.95

Learning rate	Bubble	Gaussian
Linear	0.0297	0.0122
Inverse of time	0.0131	0.0128
Power series	0.0129	0.0123

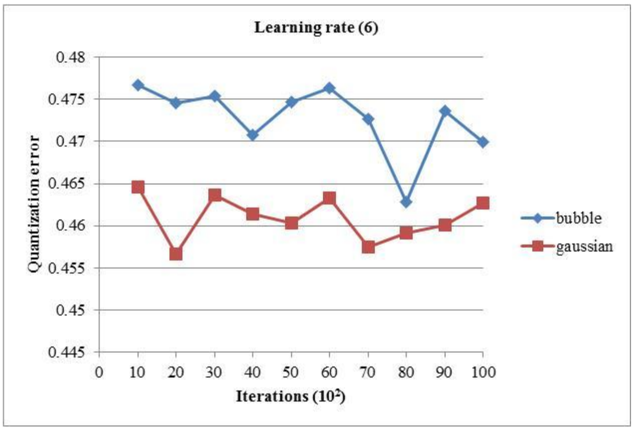
From Table II, the margins of errors are small enough for all neighborhood functions and learning rates. If the Bubble function and linear learning rate are used, the margins of errors give the largest value. If the Gaussian function and linear learning rate are used, the margins of errors give the smallest value. However, it can be seen that the margins of errors tend to give the smallest value when the Gaussian function and power series learning rate are used. The average quantization error for all cases of iterations is shown in Table III.

TABLE III: THE AVERAGE QUANTIZATION ERRORS OF SPECIFIC HUMIDITY FOR ALL CASES

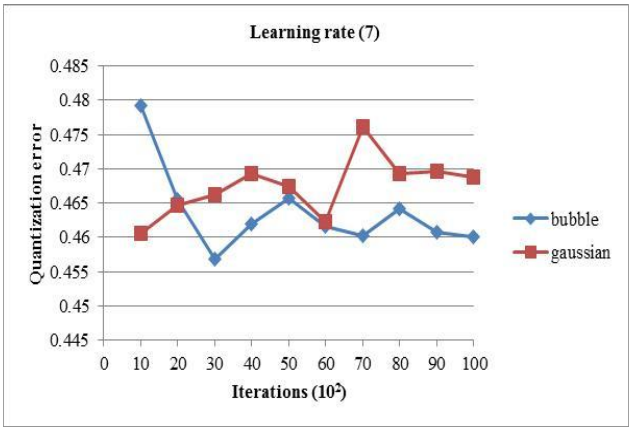
Learning rate	Bubble	Gaussian
Linear	1.0701	0.4373
Inverse of time	0.4728	0.4609
Power series	0.4636	0.4674



(a)



(b)



(c)

Fig. 3. The quantization errors of 850 hPa specific humidity at 00UTC, (a) linear learning rate, (b) inverse of time learning rate and (c) power series learning rate.

The results in each experiment are shown in Fig. 3. From Figs. 3a and 3b, it can be seen that quantization error of Gaussian is less than Bubble function for all cases of iterations. However, both quantization errors of Gaussian and Bubble function are oscillating. SOM gives the best performance with small quantization error for Gaussian function with linear and inverse of time learning rates (0.4207 and 0.4567). SOM provides the best result for both of Bubble and Gaussian functions when 6000 iterations are used (0.4616 and 0.4623). However, the average quantization error from Table III is about 0.4373 when Gaussian function with linear learning rate is used. The results from Table III and Fig. 3 are consistent. Therefore, it can be concluded that Gaussian function is appropriate for all cases of learning rates while linear learning rate gives the smallest quantization errors.

VI. CONCLUSION

SOM is an algorithm that is based on unsupervised learning method in neural network. To apply SOM in pattern classification, initial learning parameters such as learning rates and neighborhood functions have to be defined. Therefore, it is necessary to choose the proper learning parameters for the experiment. In this research, the appropriate learning rates and neighborhood functions for specific humidity pattern at 850 hPa during southwest monsoon over southern Thailand for the years 2000 to 2009 are investigated. The results illustrate that Gaussian function with linear learning rate gives the best performance with the smallest quantization error. Thus, Gaussian function with linear learning rates can be used to train for specific humidity pattern classification by SOM. However, the appropriate learning rate and neighborhood function can be changed which depends on the cases of data.

For further study to classify the weather pattern by using SOM, the appropriate learning rate and neighborhood function for temperature, pressure and geopotential height at 850 hPa during summer monsoon will be investigated.

ACKNOWLEDGMENT

The first author thanks Human Resource Development in Science Project (Science Achievement Scholarship of Thailand, SAST) and King Mongkut's University of Technology Thonburi for the financial support.

REFERENCES

[1] M. Littell. *What Is Climate?* [Online]. Available: https://www.classzone.com/books/earth_science/terc/content/investigations/es2101/es2101page01.cfm?chapter_no=investigation

[2] M. Eso, M. Kuning, and S. Chuai-Aree, "Analysis of daily rainfall during 2001-2012 in Thailand," *Songklanakarin journal of Science and Technology (SJST)*, vol. 35, pp. 81-88, February 2015.

[3] N. Skific and J. Francis. (2012). Self-organizing maps: A powerful tool for the atmospheric sciences. [Online]. Available: http://cdn.intechopen.com/pdfs-wm/40865.pdf

[4] K. C. Schuenemann, J. J. Cassano, and J. Finnis, "Synoptic forcing of precipitation over Greenland: Climatology for 1961-99," *Journal of Hydrometeorology*, vol. 10, pp. 60-78, June 2008.

[5] M. A. Nigro and J. J. Cassano, "Identification of surface wind patterns over the ross ice shelf, Antarcita, using self-organizing maps," *Monthly Weather Review*, vol. 142, pp. 2361-2378, July 2014.

[6] A. K. Guèye, S. Janicot, A. Niang, S. Sawadogo, B. Sultan, A. Diongue-Niang, and S. Thiria, "Weather regimes over Senegal during the summer monsoon season using self-orginizing maps and hierarchical ascendant classification, Part I: synoptic time scale," *Climate Dynamics*, vol. 36, pp. 1-18, March 2010.

[7] P. Stefanovič and O. Kurasova, "Visual analysis of self-organizing maps," *Nonlinear Analysis: Modelling and Control*, vol. 16, no. 4, pp. 488-504, December 2011.

[8] S. Saha, S. Moorthi, H. Pan, X. Wu *et al*., “The climate forecast system reanalysis,” *Bulletin of the American Meteroloical Society*, vol. 91, no. 8, pp. 1015-1057, January 2010.

[9] J. A. Sá, B. Rocha, A. Almeida, and J. R. Souza, “Recurrent self-organizing map for severe weather patterns recognition,” *Recurrent Neural Networks and Solf Computing*, pp. 151-175, March 2012

[10] L. Fausett, *Fundamental of Neural Networks: Architectures, Algorithms, and Applications*, USA: Prentice Hall International, 1994, pp. 169-175.

[11] J. Vesanto, J. Himberg, E. Alhoniemi, and J. Parhankangas. (April 2000). SOM Toolbox for Matlab 5. *Laboratory of Computer and Information Science (CIS)*. [Online]. Available: http://www.cis.hut.fi/somtoolbox/package/papers/techrep.pdf

[12] S. Gabrielsson and S. Gabrielsson, “The use of self-organizing maps in recommender systems,” M.S. thesis, Uppsala Univ., Uppsala, Sweden, 2006.

[13] L. Sullivan, W. W. LaMorte, and R. Kumar. (2012). Power and sample size determination. *Boston University School of Public Health*.[Online]. Available: http://sphweb.bumc.bu.edu/otlt/MPH-Modules/BS/BS704_Power/BS704_Power_print.html

[14] P. Hunter. (2012). Margin of error and confidence levels made simple. *Isixsigma*. [Online]. Available: http://www.isixsigma.com/tools-templates/sampling-data/margin-error-and-confidence-levels-made-simple/

Natita Wangsoh was born in Bangkok, Thailand, in 1989. She received the B.Sc. degree in mathematics and the M.Sc. degree in applied mathematics from King Mongkut’s University of Technology Thonburi, Bangkok, Thailand, in 2010 and 2012, respectively. In 2015, she is now a third year of Ph.D. student in applied mathematics from King Mongkut’s University of Technology Thonburi, Bangkok, Thailand.

Wiboonsak Watthayu received Ph.D in computer science from university of Maryland, USA. He had been the faculty of Department of Mathematics, King Mongkut’s University of Technology Thonburi, Bangkok, Thailand. His research interests are computational mathematics, evolutionary computing, and data science.

Dusadee Sukawat is a senior lecturer at the Joint Graduate School of Energy and Environment (JGSEE). JGSEE is an autonomous graduate school instituted under the jurisdiction of the Council of the King Mongkut University of Technology Thonburi (KMUTT), Bangkok, Thailand. Prior to joining JGSEE, Dr. Sukawat worked at the Department of Mathematics, KMUTT between 2002 and 2014. From 1977 to 2001, Dr. Sukawat worked at the Thai Meteorological Department. Dr. Sukawat holds a Ph.D. in meteorology from the Florida State University, USA, under a scholarship from the World Meteorological Organization.