Journal of Urban and Environmental Engineering, v.7, n.1, p.176-182

ISSN 1982-3932
doi: 10.4090/juee.2013.v7n1.176182

Journal of Urban and Environmental Engineering

www.journal-uee.org

KOHONEN NEURAL NETWORKS FOR RAINFALL-RUNOFF MODELING: CASE STUDY OF PIANCÓ RIVER BASIN

Camilo A. S. Farias[1*]; Celso A. G. Santos[2]; Artur M. G. Lourenço[3] and Tatiane C. Carneiro[4]

[1]*Academic Unit of Science and Technology, Federal University of Campina Grande, Brazil*
[2]*Department of Civil Engineering, Federal University of Paraíba, Brazil*
[3]*Civil and Environmental Engineering Graduate Program, Federal University of Campina Grande, Brazil*
[4]*Electrical Engineering Graduate Program, Federal University of Ceará, Brazil*

Received 3 January 2013; received in revised form 24 June 2013; accepted 30 June 2013

Abstract: The existence of long and reliable streamflow data records is essential to establishing strategies for the operation of water resources systems. In areas where streamflow data records are limited or present missing values, rainfall-runoff models are typically used for reconstruction and/or extension of river flow series. The main objective of this paper is to verify the application of Kohonen Neural Networks (KNN) for estimating streamflows in Piancó River. The Piancó River basin is located in the Brazilian semiarid region, an area devoid of hydrometeorological data and characterized by recurrent periods of water scarcity. The KNN are unsupervised neural networks that cluster data into groups according to their similarities. Such models are able to classify data vectors even when there are missing values in some of its components, a very common situation in rainfall-runoff modeling. Twenty two years of rainfall and streamflow monthly data were used in order to calibrate and test the proposed model. Statistical indexes were chose as criteria for evaluating the performance of the KNN model under four different scenarios of input data. The results show that the proposed model was able to provide reliable estimations even when there were missing values in the input data set.

Keywords: Self-organizing maps; artificial neural networks; rainfall-runoff model; semiarid area

[*] Correspondence to: Camilo A.S. Farias, Tel.: +55 83 3431 4000; Fax: +55 83 3431 4009.
E-mail: camilo@ccta.ufcg.edu.br

INTRODUCTION

The northeast region of Brazil is characterized by high rates of evaporation and irregular and intense rainfall through space and time. Such hydrological conditions, combined with the inadequate management of river basins, contribute to the occurrence of various types of problems such as alternating episodes of floods and droughts, and the entrainment of sediment into the riverbeds, reducing the ability of the water bodies and affecting the quality of its waters (Farias *et al.*, 2010; Vanmaercke *et al.*, 2010; Silva *et al.*, 2013). The need for a development that is compatible with the reality of the Brazilian semiarid hydrology has encouraged the study of strategies for a better management of existing water systems, both in terms of quality and quantity. However, the difficulty in obtaining long and reliable streamflow series has hampered the establishment of superior rules for the operation of water systems.

In places where the data of flows are limited or flawed, processes like rainfall-runoff should be investigated for the reconstruction and/or the extension of the series of flows. Over the years, several models have been developed with the intention to understand the processes of rainfall-runoff transformation in river basins, such as Stanford Watershed Model IV (Crawford & Lindsley, 1966), SSARR – Streamflow Syntesis and Reservoir Regulation (US Army Engineer Division, 1972) and SMAP – Soil Moisture Accounting Model (Lopes *et al.*, 1982). More recently, models based on artificial neural networks have been applied to the rainfall-runoff transformation, as shown in the work of Coulibaly *et al.* (2001), Jeong & Kim (2005), Farias *et al.* (2007), Wu & Chau (2011) and Santos *et al.* (2012 a,b). According to Haykin (1999) and Farias *et al.* (2010), artificial neural networks are mathematical models, inspired by the human nervous system, capable of detecting complex relationships between input and output variables.

This paper has as main objective the development and the verification of the implementation of a monthly rainfall-runoff model based on Kohonen Neural Networks (KNN) in order to estimate flows in the Piancó River, which is an intermittent river that is located in the Brazilian semiarid region.

The KNN are unsupervised neural networks that group data into classes according to their similarities through competitive learning (Kohonen, 1982; Haykin, 1999; Silva *et al.*, 2010). Also known as self-organizing maps, the KNN were proposed by Kohonen (1982) and have mostly been applied in pattern classification and data grouping. One of the main advantages of KNN is the ability to reduce a set of multidimensional data to a two-dimensional array of features which can be used for analysis and prediction purposes (Silva *et al.*, 2010; Adeloye *et al.*, 2011; Santos & Silva, 2013). The studies of Garcia & González (2004) and Adeloye *et al.* (2011) are examples of the few applications of KNN models in the area of water resources.

CASE STUDY

The watershed of the Piancó River is located in the southwest region of the state of Paraíba, northeastern Brazil. With a drainage area of 9228 km², it has semiarid climate and average annual values of precipitation and temperature around 821 mm and 24ºC, respectively. In this basin, the largest water reserve of the state, is located the system Coremas–Mãe d'Água. The affluent outflows to the system come from three major tributaries: Aguiar Creek, Emas Creek and Piancó River. The flows of the tributary Piancó are measured at the Piancó stream gauge station, which has a drainage area of 4170 km². The data collection was done in eight rain gauge stations and in one stream gauge station located in the basin of the Piancó River. Details of the studied stations are shown in **Fig. 1** and **Table 1**. The data has been obtained on the website of the National Water Agency (*Agência Nacional de Águas* – ANA, 2010). The period of analysis, knowingly chosen for presenting more complete information, comprises monthly data from January 1963 to December 1984, totaling 22 years of observations.

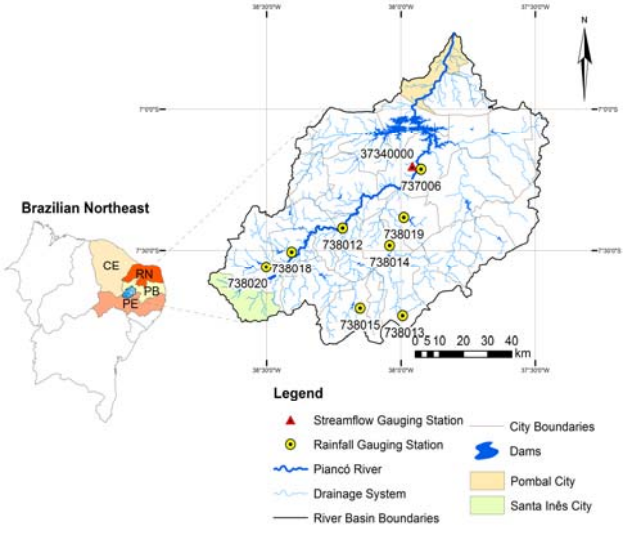

Fig. 1 Location of the rain and stream studied gauge stations in the Piancó River basin.

Table 1. Gauges that were employed in the present study

Gauge code	Gauge name	Type	City
737006 (P_1)	Piancó	Rainfall	Piancó
738020 (P_2)	Conceição	Rainfall	Conceição
738015 (P_3)	Manaíra	Rainfall	Manaíra
738013 (P_4)	Princesa Isabel	Rainfall	Princesa Isabel
738019 (P_5)	Santana dos Garrotes	Rainfall	Santana dos Garrotes
738012 (P_6)	Boa Ventura	Rainfall	Boa Ventura
738014 (P_7)	Nova Olinda	Rainfall	Nova Olinda
738018 (P_8)	Ibiara	Rainfall	Ibiara
37340000 (Q)	Piancó	Stream	Piancó

KNN MODEL

Architecture and training

The main objective of the Kohonen neural network consists of clustering vectors with similar characteristics in the same class (winner neuron) or similar classes (neighboring neurons).

The architectures of KNN contain a multi-dimensional input layer and an output layer which is either typically one-dimensional or two-dimensional. In the output layer, also known as competitive layer, the neurons compete among themselves and only one of them is considered the winner or, in simplified form, the class most suitable for a given input vector x. In these networks, each element of the input vector is connected to all the elements of the output layer. The strength of the connections is measured through weight w_{ij} between the input neurons j and the neurons of the output layer i.

During the training of the KNN model, the Euclidean distances DI_i between the input vector and the weights attached to each of the output neurons are calculated as shown by **Eq. (1)**.

$$DI_i = \sqrt{\sum_{j=1}^{J}\left(x_j - w_{ij}\right)^2}\ ;\ \text{to } i = 1, 2, ..., M. \quad (1)$$

in which x_j is the j-th component of the input vector x; J is the dimension of the input vector x; and M is the total number of neurons in the output layer.

The output neuron i that has the smallest Euclidean distance when compared to the input vector is considered the winner neuron. The weights connected to this neuron i^* and the neurons within a certain neighborhood radius V_{i^*} are then updated by the rule of Kohonen (Beale *et al.*, 2012), as shown by **Eq. (2)**.

$$w_{ij}(n) = w_{ij}(n-1) + \alpha \cdot \left[x_j(n) - w_{ij}(n-1)\right];\ \text{to } i \in V_{i^*} \text{ and } j = 1, 2, ..., J. \quad (2)$$

in which α is the learning rate, and n is an index that represents the sequence of sample presentation to the network.

The Kohonen rule forces the weights attached to the winner neuron and its neighbors to move in the direction of the input vector presented to the network, causing the Euclidean distance to become smaller and that these neurons learn to classify similar vectors.

The presentation of input vectors to the network can also be done using the entire data set before any weight update. This form of presentation is known as batch mode. In this case, the search for the winner neuron is performed for each input vector and the weight vector is then moved to a specific position calculated by the average of input vectors for which the neuron was the winner or the winner's neighbor. The weights tend to stabilize after multiple presentations of the set of input data. It is worth noting that the training of this neural network is of the unsupervised type since there are no desired outputs.

For purposes of determining the neighborhood, the distances between the neurons of the output layer can be defined in several ways (Beale *et al.*, 2012). Commonly, in a two-dimensional output layer, neurons are thought of as rectangular or hexagonal shapes and the distance are established by the number of steps between them. **Figure 2** shows how the distances between hexagonal neurons are obtained for purposes of determining the neighborhood.

The training takes place in two phases: ordering phase and tuning phase. In the first phase, training is limited by a given number of presentations of the data set and the radius of the neighborhood starts with a given distance that decreases to the unit value. This measure allows the weights of the neurons to organize themselves in the input space consistent with their positions. The tuning phase lasts the remaining number of presentations for the training defined. At this stage, the radius of the neighborhood is below unity, so that there is only update at the weight of the winner neuron. During the tuning phase, it is expected that the weights will modify themselves relatively evenly in the input space, while maintaining the topology defined in the ordering phase (Beale *et al.*, 2012).

Forecasting using the KNN model

Once trained, the KNN model can be used as predictive tools. For this, one should consider the input vector with the absence of the variable to be provided through the following steps:

(a) Calculate the Euclidean distances between the input vectors and weights attached to output neurons disregarding the element j to be provided. This can be done by including a Boolean variable m_j, as shown by **Eq. (3)**. The variable m_j is used to include (m_j = 1) or exclude (m_j = 0) the contribution of a given element j of the input vector in the calculation of Euclidean distances;

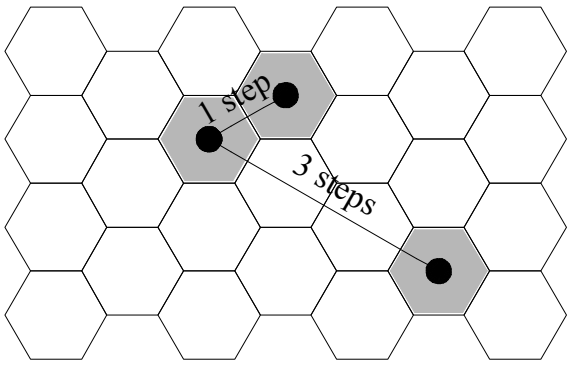

Fig. 2 Distances between neurons of a KNN model for the determination of the neighborhood.

(b) Determining the winner neuron based on the lowest Euclidean distance;
(c) Using the weight of the winner neuron connected to the missing element j of the input vector as the prediction.

$$DI_i = \sqrt{\sum_{j=1}^{J} m_j \left(x_j - w_{ij}\right)^2}; \text{ to } i = 1, 2, ..., M. \quad (3)$$

APPLICATION AND RESULTS

Application of the KNN model

In this study, the vectors of the input layer have 18 neurons representing the past and current flow, $Q(t–1)$ and $Q(t)$, and rain, $P_1(t–1)$, ..., $P_8(t–1)$, $P_1(t)$, ..., $P_8(t)$ monthly values. A two-dimensional output layer with hexagonal neurons was chosen. Based on the guidelines suggested by Garcia & González (2004), a grid of 9 × 9 neurons was used, providing a total of 81 neurons. **Figure 3** shows the structure of the KNN model of this paper and an example with a winner neuron and its neighbors.

The input data have been properly scaled to improve efficiency in the KNN model training. The scheduling process consisted of normalizing the data so that the average would be zero and the unit standard deviation (Beale *et al.*, 2012). The model training took place in batch mode, and in order to ensure a consistent learning, the dataset has been submitted 200 times to the KNN model. In the ordering phase, it was opted for 100 presentations of the dataset and an initial neighborhood radius equal to three steps. The tuning phase included the 100 remaining presentations. The KNN model was implemented in MATLAB R2012a by using the Neural Network Toolbox (Beale *et al.*, 2012).

The data used for training and testing the KNN model comprise the periods of 1963–1980 and 1981–1984, respectively.

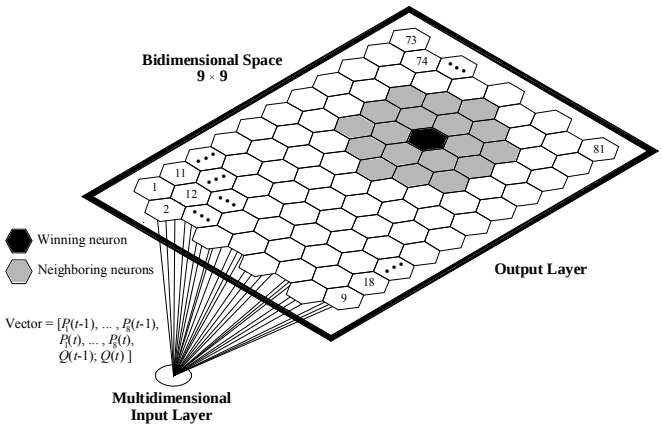

Fig. 3 Structure of KNN model and example with a winner neuron and its neighbors.

Detection of similarities

The detection of the similarities between the variables involved in this modeling can be visually performed through the plans of the components shown in **Fig. 4**. Those plans or maps represent the weights associated with each input variable. In order to facilitate the interpretation of the results, a color scale was displayed with the original dimensions of the weights, which are actually the values of the variables under study for different neurons or classes. The highest values correspond to yellow zones, and the smallest to the zones in black.

Correlations may be identified through color gradients on each plane component. Two variables with parallel gradients show a direct correlation while antiparallel gradients show an inverse correlation (Garcia & González, 2004). The analysis of **Fig. 4** allows the extraction of different information.

When analyzing the generated maps of rain data from the eight rain gauges, considering the same time period, it is found that low and high rainfall values were classified into similar categories for all the positions studied. Based on this result, it is understood that it is reasonable to use information from neighboring rain gauges for filling gaps in the series of rainfall in the region studied.

When comparing the flows with average (red cells) and high (yellow cells) magnitudes, it is clear that the map of $Q(t)$ has little similarity with the map of $Q(t–1)$. The map of $Q(t)$ has presented the higher flow rates at the bottom right. The investigation of maps focusing on a comparison of the flow $Q(t)$ with rainfall in the same period of time suggests that the flow data are strongly correlated with rainfall for the most rain gauges studied. Despite the lesser extent, the flow rates were also reasonably correlated with rainfall in the previous month. This is evidenced by the identification that the regions with low (black cells), medium and high flows have similar colors in most plans of rain at $t–1$.

Rainfall-runoff modeling

The performance evaluation of the KNN model for estimating the flow rates was based on the following indexes: correlation (R), relative bias (RB) and Nash-Sutcliffe efficiency ($NASH$). The correlation measures the degree of linear dependence between the predictions and the observed values of flow, actually expressing a potential value of good fit. The relative bias, in turn, shows that the streamflow forecasting system has a tendency to underestimate or overestimate the observed flow. The $NASH$ efficiency index, which can vary between $-\infty$ and 1, is traditionally used to express adhesion between simulated and observed flow rates.

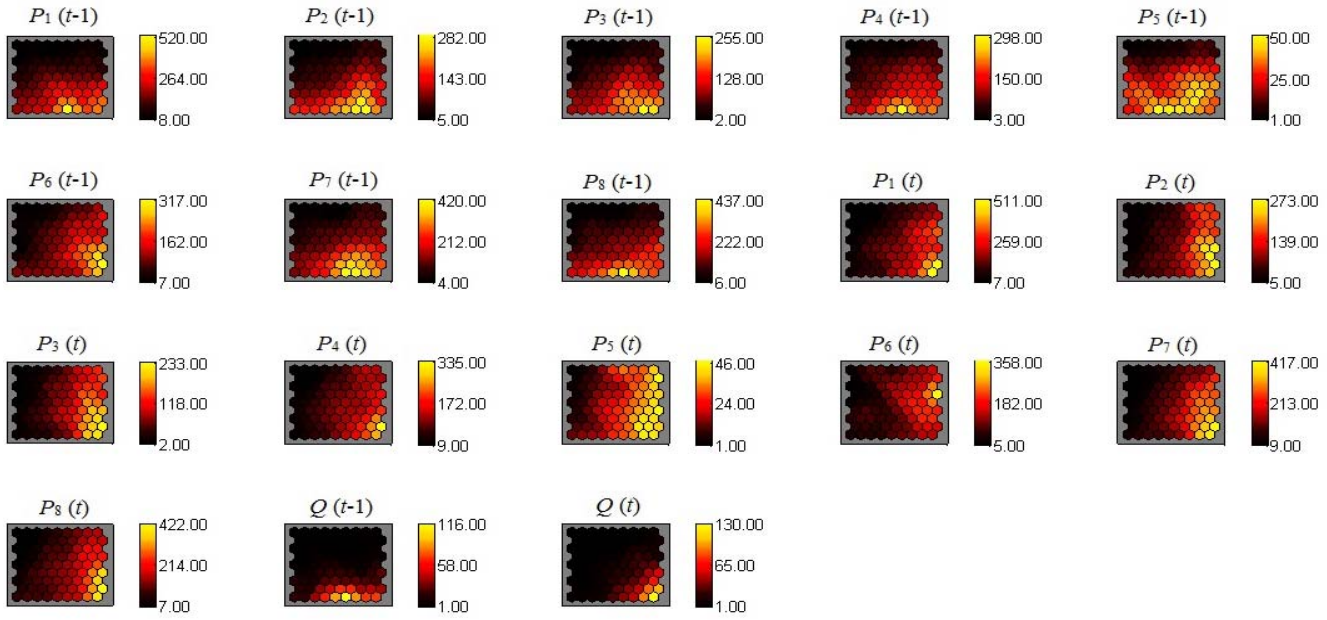

Fig. 4 Plans of the components obtained by the KNN model for rainfall data P (mm) and flow Q (m³/s) in the Piancó River basin.

This index considers both the systematic errors and the random errors, indicating that the fit is even better as its value is close to the unit. High correlation values do not mean, by itself, predictions with high accuracy. For example, a system with a very high bias, even if correlation is equal to the unit (r = 1), will give streamflow forecasts of low precision, although it is possible to remove this bias by statistic models. A perfect forecast system would have r = 1 and RB = 0. The equations for calculating these indexes can be found in Lettenmaier & Wood (1993).

Figure 5 shows a comparison between the monthly flow rate estimates obtained with the KNN model, considering the steps described in section 3.2, and the observed flow rate in the stream gauge investigated during the period of model training.

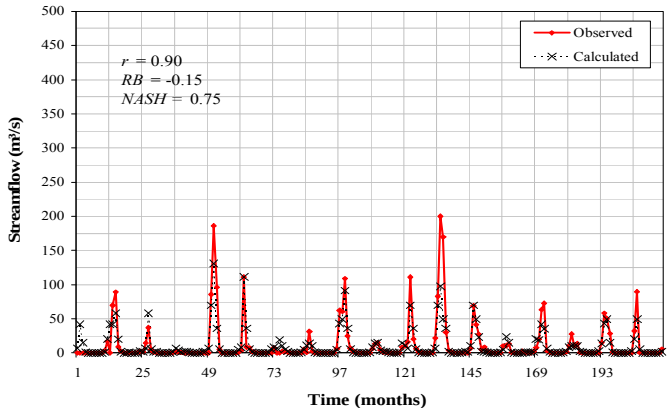

Fig. 5 Comparison between the monthly flow rates obtained with the KNN model and the observed values at the Piancó streamflow station during the 18 years of the training period (1963–1980).

Table 2. Input data for estimating flow rates by using the KNN model for four simulations

Simulation	Input data
SIM #1	$P_1(t–1)$, ..., $P_8(t–1)$, $P_1(t)$, ..., $P_8(t)$ and $Q(t–1)$
SIM #2	$P_1(t–1)$, ..., $P_8(t–1)$, $P_1(t)$, ..., $P_8(t)$
SIM #3	$P_1(t–1)$, ..., $P_8(t–1)$ and $Q(t–1)$
SIM #4	$P_1(t)$, ..., $P_8(t)$ and $Q(t–1)$

The correlation, relative bias and *NASH* results show that the KNN model could classify with good quality the flow rates of the training dataset. The KNN model was also evaluated for a period of tests represented by a series of data fully independent from those used for training the model. For this, four sets of input variables for estimating flow rates in the Piancó River were chosen and tested, as shown in **Table 2**.

Figure 6 shows the results of estimation of the flow rates for all simulations. The simulation SIM #1 shows that the estimates of the KNN model and the observed values have high correlation and a fairly low value of relative bias. The value of *NASH* was also high, indicating that the monthly flow rate estimates hold good quality. The indexes obtained for simulation SIM #2, in which some input variables have been deleted, show that the KNN model is able to produce reliable estimates even when there are failures in the input data. These results are justified by the powerful classification capabilities of the networks KNN, even in cases where some of the elements of the input vector are not present (Beale *et al.*, 2012).

Also analyzing the flow rates estimated by the simulation SIM #2, it is observed that the removal of the past flow from the set of input data did not impair the

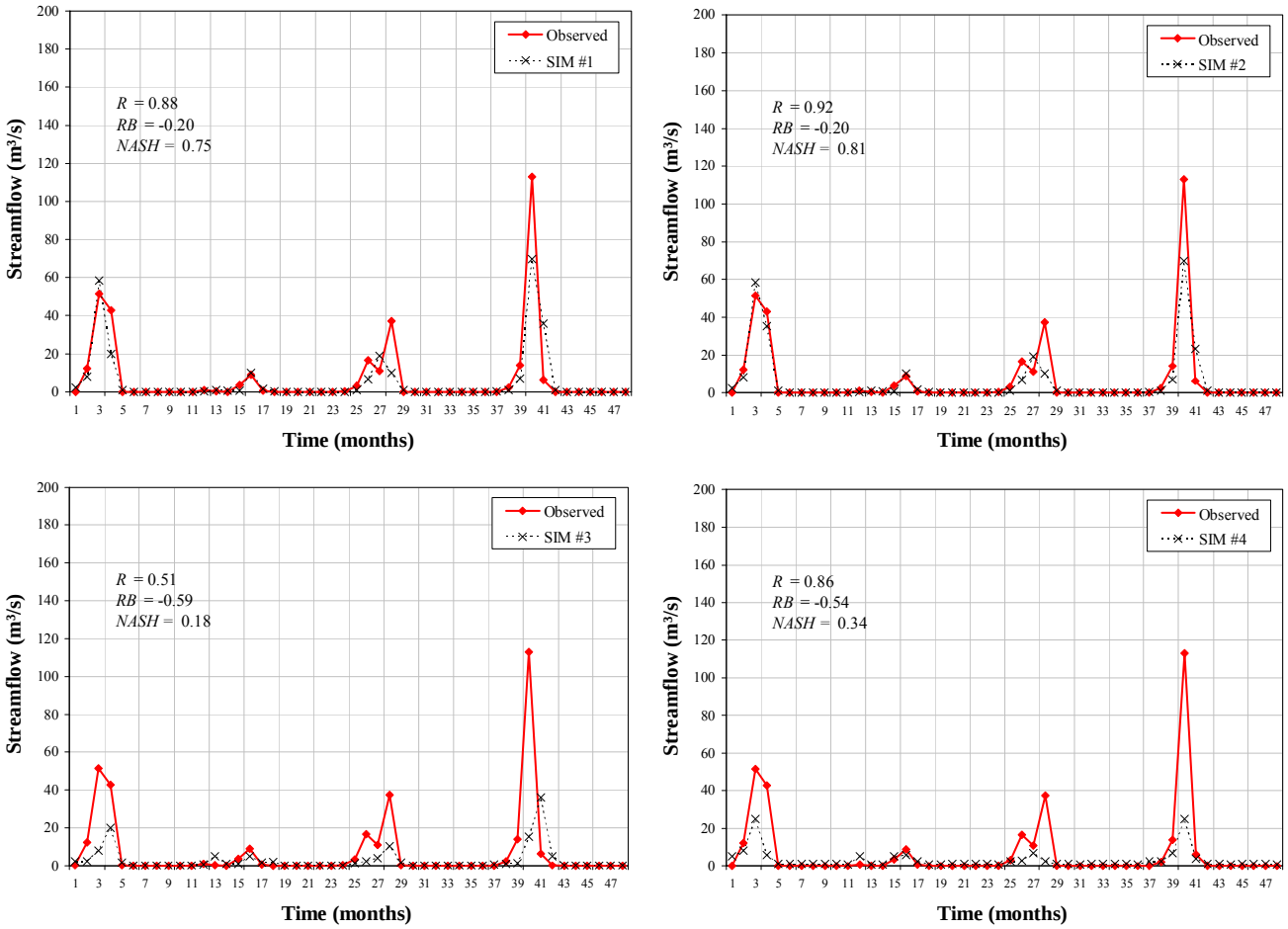

Fig. 6 Comparison between the monthly flow rate estimates obtained with the KNN model and the observed values at the Piancó stream gauge for various configurations of input data during the four years of the test period (1981–1984).

performance of the model. These results confirm the analysis performed for the detection of similarities, in which the weak relationship among the streamflows $Q(t)$ and $Q(t–1)$ was verified. On the other hand, the simulation SIM #3, which had no rain data for the current period, showed the least significant results, confirming the strong correlation identified by the maps of components between the rainfall $P(t)$ and the streamflow $Q(t)$ data. The indexes obtained from the SIM #4, which did not contain rainfall data in $P(t–1)$, outperformed the SIM #3, which in turn suppressed the rainfall data in $P(t)$. This suggests that the rainfall data $P(t)$ have more significant correlations with the streamflow $Q(t)$ than the rainfall $P(t–1)$.

CONCLUSION

This paper presented a model of Kohonen Neural Networks (KNN) for detecting similarities between monthly rainfall and runoff data, and it verified its applicability for estimating the monthly streamflow at a stream gauge on the Piancó River, which is located in the semiarid region of Paraíba state, Brazil.

The developed model was evaluated through a comparative study relating flow rates estimated by the KNN model with the data observed in the region. This comparison, by using a testing period regardless of the data used in the model training, has shown that the KNN model had a good performance for estimating the monthly flow rates.

The plans of the components generated by the KNN model were shown to be powerful analytical tools by allowing the visual identification of similarities between the variables involved in modeling. Simulations using four different configurations of inputs also indicated that the KNN model is able to produce reliable estimates even when there are faults in the input data, which is a common situation when dealing with hydrometeorological data.

The good results obtained for the stream gauge in the Piancó River suggest that this type of model can be used to reconstruct and/or extend streamflow series, especially in places where hydrometeorological data are limited or at fault. Further studies together with physically-based runoff-erosion models (e.g., Santos *et al.*, 1994, 2003, 2011a b, 2012a b, 2013; Zhang *et al.*, 2013) seems to be a promising tool for dealing with erosion issues, as well as the use of wavelet transform (e.g., Santos & Morais, 2013; Santos & Silva, 2013).

Acknowledgment The financial support provided by CNPq (National Council for Scientific and Technological Development, Brazil) is gratefully acknowledged.

REFERENCES

Adeloye A.J., Rustum, R. & Kariyama, D. (2011). Kohonen self-organizing map estimator for the reference crop evapotranspiration. *Water Resour. Res.* 47, W08523, 1-19.

ANA – Agência Nacional de Águas. Disponível em http://hidroweb.ana.gov.br/, accessed in 27/09/2010.

Beale, M., Hagan, M. & Demuth H. (2012). Neural Network Toolbox 7.0.3: User's Guide. The MathWorks Inc, Natick, USA, 404 p.

Coulibaly, P., Anctil F. & Bobée B. (2001) Multivariate reservoir inflow forecasting using temporal neural networks, *J. Hydrol. Engng*. ASCE, 6, 367–376.

Crawford, N.H. & Lindsley JR., R.K. (1966) Digital simulation in hydrology: Stanford Watershed Model IV, Department of Civil Engineering, Stanford University, Technical Report, n. 39.

Farias, C.A.S., Alves, F.M., Santos, C.A.G. & Suzuki, K. (2010) An ANN-based approach to modelling sediment yield: a case study in a semi-arid area of Brazil. *IAHS Publ.* 337, 316–321.

Farias, C.A.S., Kadota, A., Celeste, A.B. & Suzuki, K. (2007) RNN-based inflow forecasting applied to reservoir operation via implicit stochastic optimization, *IAHS Publ.* 313, 452–462.

García, H.L. & González, I.M. (2004) Self-organizing map and clustering for wastewater treatment monitoring. *Engng. Appl. Artificial Intellig.* **17**(3), 215–225.

Haykin, S. (1999). Neural Networks: a Comprehensive Foundation. Prentice Hall, Upper Saddle River, USA, 842 p.

Jeong, D.-II & Kim, Y.-O. (2005) Rainfall-runoff models using artificial neural networks for ensemble streamflow prediction, *Hydrol. Process.*, 19, 3819–3835.

Kohonen, T. (1982) Self-organized formation of topologically correct feature maps. *Biological Cybernetics*. 43, 59–69.

Lettenmaier D.P. & Wood, E.F. (1993) *Hydrologic forecasting.*. Org. by Maidment, D. R. Handbook of Hydrology. McGraw-Hill Inc., New York, USA, 26.1–26.30.

Lopes, J.E., Braga B.F.F. & Conejo, J.L. (1982) A simplified hydrologic model. In *Applied Modeling in Catchment Hydrology*, Water Resources Publication.

Santos, C.A.G. & Morais, B.S. (2013) Identification of precipitation zones within São Francisco River basin (Brazil) by global wavelet power spectra. *Hydrol. Sci. J.* 58, 789–796, 2013. doi: 10.1080/02626667.2013.778412

Santos, C.A.G. & Silva, G.B.L. (2013) Daily streamflow forecasting using a wavelet transform and artificial neural network hybrid models. *Hydrol. Sci. J.* 2013. doi: 10.1080/02626667.2013.800944

Santos, C.A.G., Freire, P.K.M.M. & Mishra, S.K. (2012a) Cuckoo search via Lévy flights for optimization of a physically-based runoff-erosion model. *J. Urban Environ. Engng*. **6**(2), 123–131. doi: 10.4090/juee.2012.v6n2.123131

Santos, C.A.G., Freire, P.K.M.M., & Arruda, P.M. (2012b) Application of a simulated annealing optimization to a physically-based erosion model. *Water Sci. and Technol.* **66**(10), 2099–2108. doi: 10.2166/wst.2012.426

Santos, C.A.G., Freire, P.K.M.M., Mishra, S.K. & Soares Júnior, A. (2011a) Application of a particle swarm optimization to the tank model. *IAHS Publ.* 347, 114–120.

Santos, C.A.G., Freire, P.K.M.M., Silva, R.M., Arruda, P.M. & Mishra, S.K. (2011b) Influence of the catchment discretization on the optimization of runoff-erosion modeling. *J. Urban Environ. Engng.***5**(2), 91–102. doi: 10.4090/juee.2011.v5n2.091102

Santos, C.A.G., Srinivasan, V.S., Suzuki, K. & Watanabe, M. (2003) Application of an optimization technique to a physically based erosion model. *Hydrol. Process.* **17**, 989–1003, doi: 10.1002/hyp.1176.

Santos, C.A.G., Suzuki, K., Watanabe, M. & Srinivasan, V.S. (1994) Optimization of coefficients in runoff-erosion modeling by Standardized Powell method, *J. Hydrosci. and Hydraul. Engng*. **12**(1), 67–78.

Silva, I.N., Spatti, D.H. & Flauzino R.A. (2010) *Redes neurais artificiais para engenharia e ciências aplicadas*, Artliber, São Paulo, 399 p.

Silva, R.M., Silva, V.C.L., Santos, C.A.G. & Silva, L.P. (2013) Erosivity, surface runoff and soil erosion estimation using GIS-coupled runoff-erosion model in the Mamuaba catchment, Brazil. *Environ. Monit. Assess*. **185**(7), 953–970. doi: 10.1007/s10661-013-3228-x.

US Army Engineer Division (1972) *Program description and user manual for SSARR*, North Pacific, Portland Oregon.

Vanmaercke, M., Zenebe, A., Poesen, J., Nyssen, J., Verstraeten, G. & Deckers, J. (2010) Sediment dynamics and the role of flash floods in sediment export from medium-sized catchments: a case study from the semi-arid tropical highlands in northern Ethiopia. *J. Soils Sediments* **10**(6), 611–627. doi: 10.1007/s11368-010-0203-9

Wu, C.L. & Chau, K.W. (2011)Rainfall–runoff modeling using artificial neural network coupled with singular spectrum analysis. *J. Hydrol.* **399**(3-4), 394–409.

Zhang, R., Santos, C.A.G., Moreira, M., Freire, P.K.M.M. & Corte-Real, J. (2013) Automatic Calibration of the SHETRAN Hydrological Modelling System Using MSCE. *Water Resour. Manag.* **27**(11), 4053–40681–16. doi: 10.1007/s11269-013-0395-z