

This discussion paper is/has been under review for the journal Hydrology and Earth System Sciences (HESS). Please refer to the corresponding final paper in HESS if available.

Generalized versus Non-Generalized Neural Network model for multi-lead inflow forecasting at Aswan High Dam

A. El-Shafie¹ and A. Nouredin²

¹Senior Lecturer, Civil and Structural Engineering Dept. University Kebangsaan Malaysia, Malaysia

²Associate Professor, Electrical and Computer Engineering, Royal Military College, Kingston, Canada

Received: 24 September 2010 – Accepted: 30 September 2010 – Published: 12 October 2010

Correspondence to: A. El-Shafie (elshafie@vlsi.eng.ukm.my)

Published by Copernicus Publications on behalf of the European Geosciences Union.

HESSD

7, 7957–7993, 2010

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Nouredin

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Abstract

Artificial Neural Networks (ANN) have been found efficient, particularly in problems where characteristics of the processes are stochastic and difficult to describe using explicit mathematical models. However, time series prediction based on ANN algorithms is fundamentally difficult and faces problems. One of the major shortcomings is the over-fitting while training session and occurs when a ANN loses its generalization. In this research, Generalized Neural Network (GNN) model is developed to overcome the drawbacks of conventional forecasting techniques. Using GNN helped avoid over-fitting of training data which was observed as a limitation of classical ANN models. Real inflow data collected over the last 130 years at Lake Nasser was used to train, test and validate the proposed model. Results show that the proposed GNN model outperforms Non-Generalized Neural Network and conventional auto-regressive models and it could provide accurate inflow forecasting.

1 Introduction

Developing optimal release policies for a multi-objective reservoir such as Lake Nasser is a complex process. The complexity is attributed to the explicit stochastic environment (e.g. uncertainty in future inflows) and the fact that when modeling such environments with high uncertainty, future returns cannot be predicted with acceptable accuracy. In this context, several forecasting models were developed using the univariate auto-regressive moving average representative of the natural inflow at Aswan High Dam (AHD) (see Fig. 1) (Georgakakos, A. and Yao, H., 1995; Georgakakos, A., 2007). These models tend to either overestimate low flows or underestimate high flows. The drawbacks are very significant when it comes to efficient and effective reservoir regulation. Therefore, it is essential to develop a forecasting model that is robust and free from these drawbacks.

HESSD

7, 7957–7993, 2010

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

training set. However, time series prediction based on ANN learning algorithms is fundamentally difficult and faces problems. One of the major shortcomings is that the ANN model experienced over-fitting problem while training session and occurs when a Neural Network loses its generalization.

The aim of this paper is to introduce two generalization methods to be integrated with classical MLPNN to overcome the over-fitting problem. Among different types of Neural Networks, this research mainly focus on (MLPNN) for prediction, because of the inherent simple architecture of these networks. However, any other Neural Network architecture will be acceptable through this approach. These methods are suitable to be applied to predict time series of complex systems' behavior, based on Neural Networks and using soft computing methods. The main idea of the proposed method is to introduce a technique to overcome the over-fitting problem while developing ANN prediction model. Finally, the proposed methods will be examined and compared with the developed non-generalized MLPNN for inflow forecasting at AHD.

2 Methodology

2.1 Artificial Neural Network and Over-fitting

Artificial Neural Networks (ANN) is densely interconnected processing units that utilize parallel computation algorithms. The basic advantage of ANN is that they can learn from representative examples without providing special programming modules to simulate special patterns in the data set (Gibson and Cowan, 1990). This allows ANN to learn and adapt to a continuously changing environment. Therefore, ANN can be trained to perform a particular function by tuning the values of the weights (connections) between these elements. The training procedure of ANN is performed so that a particular input leads to a certain target output as shown in Fig. 2.

The input and output layers of any network have numbers of neurons equal to the number of the inputs and outputs of the system respectively. The architecture of a

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

[Title Page](#)[Abstract](#)[Introduction](#)[Conclusions](#)[References](#)[Tables](#)[Figures](#)[⏪](#)[⏩](#)[◀](#)[▶](#)[Back](#)[Close](#)[Full Screen / Esc](#)[Printer-friendly Version](#)[Interactive Discussion](#)

multi-layer feed-forward Neural Network can have many layers between the input and the output layers where a layer represents a set of parallel processing units (or nodes) namely, the hidden layer. The main function of the hidden layer is to allow the network to detect and capture the relevant patterns in the data and to perform complex non-linear mapping between the input and the output variables. The sole role of the input layer of nodes is to relate the external inputs to the neurons of the hidden layer. Hence the number of input nodes corresponds to the number of input variables. The outputs of the hidden layer are passed to the last (or output) layer, which provides the final output of the network. Finding a parsimonious model for accurate prediction is particularly critical since there is no formal method for determining the appropriate number of hidden nodes prior to training. Therefore, here we resort to a trial-and-error method commonly used for network design.

One of the most important aspects of machine learning models is how well the model generalizes to unseen data. The over-fitting problem occurs when a Neural Network loses its generalization feature. In other words, it cannot generalize the relations which exist between training inputs and their related outputs to the similar hidden patterns of the unobserved data. In such cases the performance of Neural Network measured on the training set is much better compared to new inputs. In predicting time series, the aim is to be able to deal with time varying sequences. This can be achieved if the network input-output patterns involve in such a way that it can respond to temporal sequences. Consequently, networks within its structure should be considered as a good choice. However, whatever architecture is used some definite problems such as over-fitting will be met (Tetko et al., 1995; Haykin, 1994; Bishop, 1996; Duda et al., 2001; Hart and Stork, 2001; Box and Jenkins, 1970).

In the following section, a brief description for ANN model for inflow forecasting at AHD will be reported, (El-Shafie et al., 2008), then the proposed methods for generalization will be applied to overcome the experienced over-fitting in the model.

2.2 Inflow forecasting with Multi-Layer Perceptron (MLP) Neural Network

The inputs to the network are fixed length successive sequence of its recent behavior. The inputs are used to predict the next time-step. The general behaviors of the complex system are saved in the layers of networks. In the prediction stage, the input data together with this overall total behavior are presented to the hidden layers. The output of the hidden layer becomes a well-conditioned result of total system behavior and then the prediction can be done after this stage.

Comprehensive data analysis for the historical inflow pattern for each month has been carried out, (El-Shafie et al., 2008). In fact, the monitored inflow is random in nature and has to be modeled stochastically in order to develop an appropriate inflow forecasting method. Stochastic models are always established based on correlation analysis, (El-Shafie et al., 2008). Accordingly, an analysis of such random inflow data by studying the auto-correlation sequences for each month over the past 130 years and the cross-correlation between consecutive months in the same year was performed, (El-Shafie et al., 2008). The study of the auto-correlation function clearly informs us as to how the process is correlated with itself over time, while studying the cross-correlation sequences provides information about the mutual correlation between two consecutive months. Such analysis allows achievement of the appropriate number of inflows of the prior months utilized as input to the ANN model in order to provide accurate inflow forecasting at certain month. Moreover, the inflow forecasted at month t can be used with the monitored inflow of some previous months to provide a forecasting at month $t + 1$. This procedure of using the forecasted inflow can be repeated for L months with the value of L dependent upon the environmental conditions and the basin characteristics (Salem and Dorrah, 1982). It has been reported by Atiya et al. (1990) that the lead-time L cannot be more than three months.

Our pilot investigation showed that inflow forecasting at month t based on the monitored inflow from the previous years of the same month (instead of previous months of the same year) cannot provide reliable results. Therefore, in this study, ANN with its

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



nonlinear and stochastic modeling capabilities is utilized to develop a forecasting model that mimics the inflow pattern at AHD and predict the inflow pattern for three months ahead based upon the monitored/forecasted inflow from the three previous months (El-Shafie et al., 2008). The inflow Q_f forecasted at month t based on the inflow monitored Q_m at the previous three months can be expressed as:

$$Q_f(t) = f(Q_m(t-1), Q_m(t-2), Q_m(t-3)) \quad (1)$$

Consequently, the inflow for month $t + 1$ can be forecasted as follows:

$$Q_f(t+1) = f(Q_f(t), Q_m(t-1), Q_m(t-2)) \quad (2)$$

Similarly, the inflow for month $t + 2$ can be forecasted using the following equation:

$$Q_f(t+2) = f(Q_f(t+1), Q_f(t), Q_m(t-1)) \quad (3)$$

Q_f in all of the above equations represents forecasted inflow while Q_m is a monitored inflow. A schematic representation of the above procedure is given in Fig. 3.

The ANN model is established using the above three equations. The architecture of the network consists of an input layer of three neurons (corresponding to the monitored/forecasted inflow of the previous three months at the inputs to the network), an output layer of one neuron (corresponding to the forecasted inflow) and a number of hidden layers of arbitrary number of neurons at each layer. In order to achieve the desirable forecasting accuracy, twelve ANN architectures were developed (one for each month). Monthly natural inflows for the period of sixty years, from 1871 to 1930, were utilized in order to train the twelve networks. The performance and the reliability of the ANN models were examined using the inflow data monitored between 1931 and 1960. The capabilities of the developed ANN models were further verified by the inflow data between 1961 and 2000, which corresponds to the inflow monitored after the construction of AHD in 1960.

In order to accelerate the training procedure and to achieve minimum mean square estimation error, the inflow data was normalized. Different MLP-ANN architectures

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page	
Abstract	Introduction
Conclusions	References
Tables	Figures
⏪	⏩
◀	▶
Back	Close
Full Screen / Esc	
Printer-friendly Version	
Interactive Discussion	



(while keeping three neurons in the input layer and only one neuron in the output layer) were used to examine the best performance. The choice of the number of hidden layers and the number of neurons in each layer is based on two performance indices. The first index is the root mean square (RMS) value of the prediction error and the second index is the value of the maximum error. Both indices were obtained while examining the ANN model with the inflow data between 1931 and 1960. The last group of data (between 1961 and 2000) that was not used in training was used to verify the capabilities of the ANN model. An example of ANN architecture used for predicting the inflow for the month of August is presented in Fig. 4, (El-Shafie et al., 2008).

The number of hidden layers (R) and the number of neurons in each layer (N) for twelve networks are presented in Table 1. The transfer functions used in each layer of the networks are also listed in Table 1. All twelve networks utilize the backpropagation algorithm during the training procedure. Once the network weights and biases are initialized, during training process the weights and biases of the network are iteratively adjusted to minimize the network performance function mean square error MSE – the average squared error between the network outputs a and the target outputs t . In order to overcome and improve the proposed model performance two procedures are introduced hereafter in the following sub-sections.

Over-fitting has often been addressed using techniques such as weight decay, weight elimination and early stopping to control over-fitting (Weigend et al., 1992). Among these methods Early-stopping is the most well known solution (Prechelt, 1998). However using this method on time series of complex systems' behavior, stops the training process too early and the chance of detecting meaningful relations between the network outputs and actual behavior of the complex system does decrease. This indicates that the resulting model will not have proper features for predicting the time series of the system's behavior. In the proposed method we do not focus on removing the over-fitting problem for a single Neural Network. Instead the major effort is to find an algorithm which is applied on the outputs of the over-fitted networks to produce the correct results. This algorithm will be presented in the following sections.

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



2.3 Neural Network generalization

2.3.1 Regularization procedure

Network over-fitting is a classical machine learning problem that has been investigated by many researchers (Schaffer, 1993; Stallard and Taylor, 1999). Network over-fitting usually occurs when the network captures the internal local patterns of the training dataset rather than recognizing the global patterns of the datasets. The knowledge rule-base that is extracted from the training dataset is therefore not general. As a consequence, it is important to recognize that the specification of the training samples is a critical factor in producing a Neural Network capable of making the correct responses. The problem of over-fitting has also been investigated by researchers with respect to network complexity (Ripley, 1996; Ooyen and Nienhuis, 1992; Livingstone, 1997).

Here, to avoid an over-fitting problem, we utilized the regularization technique (Nordström and Svensson, 1992). This is known as a suitable technique when the scaled conjugate gradient descent method is adopted for training, as is the case in this study. The regularization technique involves modifying the performance function which is normally chosen to be the sum of squares of the network errors on the training set defined as:

$$\text{MSE} = \frac{1}{2} \sum_{P=1}^n (Y_O - Y_P)^2 \quad (4)$$

The modified performance function is defined by adding a term that consists of the mean of the sum of squares of the network weights and biases to the original Mean Square Error (MSE) function as:

$$\text{MSE}_{\text{reg}} = \gamma \times \text{MSE} + (1 - \gamma) \times \text{MSW} \quad (5)$$

Where γ is the performance ratio that takes values between 0 and 1; and MSW is computed as:

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



$$MSW = \frac{1}{M} \sum_{j=1}^M w_j^2 \quad (6)$$

where M is the number of weights utilized inside the network structure and w is weight matrix of the network. Using the performance function of Eq. (10), the Neural Networks to predict the inflow at AHD were developed with the intention to avoid over-fitting of data.

2.3.2 Assembly Neural Network Procedure

Initialization phase

In this procedure, it is proposed to find a technique based on ensemble Neural Networks (Chiewchanwattana et al., 2002; Drucker et al., 1994; Cheni et al., 2005) that by using over-fitted Neural Networks leads to generalization. In order to achieve this goal, we use a sequence of the previous behavior of the system as the training data then generate a sequence of inputs with proper length and their corresponding outputs from first the 90 percent of 60 years (training data) and with respect to the size of the best period regarded to the previous section. Subsequently we construct a series of networks with an initial guess for the number of hidden layers neurons and initialize their parameters randomly. Finally for every network, the parameters vector will stop on a local minimum of its performance surface. Up to this point all of the networks are over fitted on the training set. Afterward a Simulated Annealing process is applied on each network. To do this, the model is modified to generate a set of vectors named the noise vectors. Length of each noise vector is equal to the length of each network parameters vector and its components are random numbers with uniform distribution between -0.05 and $+0.05$. By adding noise vectors to the network parameter vectors a new set of network parameters are obtained. This action makes relatively minor changes in the location of each network in its state space.

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



Networks are trained with these noisy parameters until another local minimum is achieved. Making noise vectors and training are repeated for a number of times and the outputs of these networks are compared on the followed 10 percent of the 60 years which are not used during training steps. The winner has the best generalization amongst all and is selected as the first member of an ensemble of Neural Networks.

Learning phase

In this phase a random vector of length \mathbf{N} is generated where \mathbf{N} is the length of the sequence of the first 60 years of time series values. This vector is called data noise vector and shown by following equation:

$$V_{dn} = M \times 10^{-2} \times z \times \text{rand}(1, \mathbf{N}) \quad (7)$$

In this equation z is the number of networks added to the ensemble of Neural Networks before this step. $\text{Rand}(1, \mathbf{N})$ is a $1 \times \mathbf{N}$ vector. The components of this vector are uniformly distributed random numbers between -0.05 and $+0.05$. Also, $M = \text{Max} - \text{Min}$ where Max and Min are the maximum and minimum values of time series of the system's behavior respectively. Once again we select the network that has the best generalization on these new training data sets. But this time the number of neurons in the hidden layers of networks is calculated using the following equations:

$$n'_1 = \left\lfloor x \times \frac{n_1}{n_2} \right\rfloor \quad (8)$$

$$n'_2 = x + n_2 \quad (9)$$

Where n_1 and n_2 are the initial number of neurons in the first and second hidden layers of the first set of networks and n'_1 and n'_2 are new values. The value of x is gained through the following equation:

HESSD

7, 7957–7993, 2010

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

$$x = \left\{ \left\lfloor \frac{IN+1}{2} \right\rfloor \times \text{mod}((IN-1), 2) \right\} \quad (10)$$

$$- \left\{ \left[\left\lfloor \frac{IN+1}{2} \right\rfloor \times \left\lfloor \frac{1 + \text{sign}(n_2 - \left\lfloor \frac{IN+1}{2} \right\rfloor)}{2} \right\rfloor + \left((2 \times n_2 - \left\lfloor \frac{IN+1}{2} \right\rfloor) - 1 \right) \times \left\lfloor \frac{1 - \text{sign}(n_2 - \left\lfloor \frac{IN+1}{2} \right\rfloor)}{2} \right\rfloor + 0.5 \right] \times \text{mod}(IN, 2) \right\}$$

In this equation, IN is the iteration number. Following the initial step, if IN is even the networks will be constructed with the previous structure but with more neurons. However if IN is odd then the number of neurons will be decreased until a zero limit is met. At this step we will continue the process by increasing the number of neurons. This enables us to find more suitable number of neurons in the completion process of the ensemble if the initial guess was not accurate and networks need more (or less) number of neurons to achieve a good generalization.

After finding the best network in each set, we compute the sum of absolute errors in the prediction of the last 10 percent of data using the following equations:

$$e_1 = \frac{\sum_{i=1}^z \sum_{j=1}^k |T_s(j) - \text{Pr}(i, j)|}{z} \quad (11)$$

$$e_2 = \frac{\sum_{i=1}^{z+1} \sum_{j=1}^k |T_s(j) - \text{Pr}(i, j)|}{z+1} \quad (12)$$

In the above equations z is the number of networks that was added to the ensemble before this step. T_s is the sequence of the last 10 percent events of time series in training data set and k is the size of T_s . $\text{Pr}(i, j)$ is the value that i^{th} member of ensemble predicts for the j^{th} event in the last 10 percent of time series events. If $e_1 > e_2$, adding the best network of this step to the ensemble has caused improvements in the generalization of the ensemble totally. Otherwise, we don't add the selected network to the ensemble and repeat this step again with new noisy data sets and a new set of

networks with different number of neurons in their hidden layers. The terminating condition is as follows: a predefined number of iterations (namely i_{tr}) are considered and at the end of these iterations the improvement of ensemble predictions (on the last 10 percent) are measured. If this value is smaller than a predefined factor the termination condition is met. Otherwise this process will be repeated. Fig. 5 illustrates the learning phase for the proposed ensemble ANN model.

3 Results and discussions

The ANN model architecture of Fig. 3 is employed in this study to provide inflow forecasting at each month. The monitored inflow over sixty years between 1871 and 1930 was used to train twelve networks with each network corresponding to one month. All twelve networks successfully achieved the target MSE of 0.0001. For example, the training curve for the month of August is demonstrated in Fig. 6 showing convergence to the target MSE after 73 iterations. Two sets of analysis were performed: network without generalization and network with generalization.

3.1 Non-Generalized Network

The twelve non-regularized networks developed during the training procedure are used to provide the inflow forecasting for the next thirty years between 1931 and 1960. Since the inflow was accurately monitored over the thirty year period, the performance of the proposed ANN-based architecture can be examined and evaluated. The distribution of the percentage value of the error over these thirty years as well as its RMS value are the two statistical performance indices used to evaluate the model accuracy. The distribution of the percentage error between the monitored (actual) and the forecasted inflows for four different months over thirty years between 1931 and 1960 is shown in Fig. 7. It can be observed that the highest percentage error for these four specific months is 7%. However, other months (October, February, April, May, and July) show higher percentage errors, up to 18%, as depicted in Fig. 8.

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



of the months of August, September and October are utilized at the input. This error increases to 0.1441 BCM when the monitored inflows of August and September and the forecasted inflow of October are used. Furthermore, the RMSE has further increased to 0.1561 BCM when the monitored inflow of August and the forecasted inflows of September and October are used.

4 Conclusions

Although Neural Networks have been widely used as a proper tool for predicting time series, but they face few problems such as over fitting. This research proposed two different methods to resolve the over fitting problem which is based on training multi-layer perceptron Neural Networks and using the simulated annealing for the optimization purpose. To reduce the effects of over fitting, regularized and ensemble Neural Networks are used. In order to evaluate the proposed approach, the proposed generalized methods were examined for inflow forecasting of the Nile River at Aswan High Dam utilizing monitored 130 years monthly based inflow data. The outcomes clearly show that the proposed methods succeed to overcome the over fitting experienced for standard Neural Network model and perform well in characterizing and predicting complex time series events and improve the output accuracy when switching the model to the verification stage. In spite of the highly stochastic nature of the inflow data in this region, the proposed ensemble ANN model was capable of mimicking the inflow pattern accurately with relatively small inflow forecasting errors. Furthermore, the ensemble ANN model significantly outperformed the classical and the regularized Neural Network of similar architecture and the conventional ARMA models.

Acknowledgements. This research is supported by: (1) a research grants to the first author by Smart Engineering System, University Kebangsaan Malaysia; (2) a research grant to the second author from the Natural Science and Engineering Research Council (NSERC) of Canada. Support from the former chairperson of the National Water Research Center, Egypt, Mona El-Kady is highly appreciated.

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



References

- Atiya, A. F., El-Shoura, S. M., Shaheen, S. I., and El-Sherif, M. S.: A Comparison Between Neural-Network Forecasting Techniques—Case Study: river Flow Forecasting, *IEEE Transactions On Neural Networks*, 10(2), 402–409, 1999.
- 5 Bishop, C. M.: *Neural Networks For Pattern Recognition*, 1st Edition, Oxford: Oxford University Press, 1996.
- Box, G. H. and Jenkins, G.: *Time Series Analysis: forecasting And Control*, San Francisco, Holden-Day, 1970.
- Cheni, D. W. and Zhang, J. P.: Time series prediction based on ensemble anfis, *Proceedings Of The Fourth International Conference On Machine Learning And Cybernetics*, Guangzhou, 18–21, 2005.
- 10 Chiewchanwattana, S., Lursinsap, C., and Chu, C. H.: Timeseries Data Prediction Based On Reconstruction Of Missing Samples And Selective Ensembling Of FIR Neural Networks, *Proceedings Of The 9th International Conference On Neural Information Processing*, 2002.
- 15 Coulibaly, P. and Anctil, F.: Real-Time Short-Term Natural Water Inflows Forecasting Using Recurrent Neural Networks, in: *International Joint Conference On Neural Networks*, Proceedings Ijcn'n'99, Washington DC, 6, 3802–3805, 1999.
- Coulibaly, P., Anctil, F., and Bobée, B.: Real Time Neural Network-Based Forecasting System For Hydropower Reservoirs, in: *First Int. Conf. On New Inform. Tech. For Decision Making In Civil Eng.*, Montreal, 1001–1011, 1998.
- 20 Coulibaly, P., Anctil, F., and Bobée, B.: Daily Reservoir Inflow Forecasting Using Artificial Neural Networks With Stopped Training Approach, *J. Hydrol.*, 230, 244–257, 2000.
- Coulibaly, P., Anctil, F., and Bobée, B.: Multivariate Reservoir Inflow Forecasting Using Temporal Neural Networks, *ASCE Journal Of Hydrologic Engineering*, 65, 367–376, 2001.
- 25 Drucker, H., Cortes, C., and Jackel, D.: Boosting And Other Ensemble Methods, *Neural Comput.*, 6, 1289–1301, 1994.
- Duda, R. O., Hart, P. E., and Stork, D. G.: *Pattern Classification*, Second Edition: wiley Interscience, NY, USA, 2001.
- El-Shafie, A., Reda Taha, M., and Noureldin, A.: A Neuro-Fuzzy Model For Inflow Forecasting Of The Nile River At Aswan High Dam, *Water Resour. Manag.*, Springer, 21(3), 533–556, 2007.
- 30 El-Shafie, A., Noureldin, A., Taha, M. R., and Basri H.: *Neural Network Model For Nile River*

HESSD

7, 7957–7993, 2010

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Seno, E., Murakami, K., Izumida, M., and Matsumoto, S.: Inflow Forecasting Of The Dam By The Neural Network, In Artificial Neural Networks In Engineering Conference, St. Louis, USA, 2003.

5 Stallard, B. R. and Taylor, J. G.: Quantifying Multivariate Classification Performance – The Problem Of Overfitting, CD Proceedings, SPIE Annual Conference, Denver, 1999.

Tetko, I. V., Livingstone, D. J., Luik, A. I.: Neural Network Studies. Comparison Of Overfitting And Overtraining, Journal Of Chemistry And Information, Computer Science, 35, 826–833, 1995.

10 Weigend, A. S., Huberman, B. A., and Rumelhart, D. E.: Predicting Sunspots And Exchange Rates With Connectionist Networks, in: Nonlinear Modeling and forecasting, edited by: Casdagli, M. and Eubank, S., SFI Studies In The Sciences Of Complexity, Proceedings Vol. XII, Addison-Wesley, 395–432, 1992.

HESSD

7, 7957–7993, 2010

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Table 2. RMSE associated with NN forecasting model for each month.

Month	RMSE (BCM)	Maximum inflow (BCM)	Minimum inflow (BCM)	RMSE/(Ave Inflow)
Aug	0.4900	29.10	6.50	0.0275
Sep	0.7805	32.79	7.31	0.0389
Oct	1.0689	27.40	5.97	0.0640
Nov	0.1801	14.40	4.12	0.0194
Dec	0.2838	11.00	2.83	0.0410
Jan	0.2347	7.70	1.72	0.0490
Feb	0.2282	6.04	1.15	0.0634
Mar	0.0909	5.81	1.07	0.0264
Apr	0.2389	5.26	0.95	0.0760
May	0.3215	4.72	0.80	0.1164
Jun	0.2665	5.16	0.90	0.0879
Jul	0.5544	11.03	1.74	0.0868

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Table 3. RMSE associated with Ensemble ANN forecasting model before and after generalization (1931–1960).

Month	RMSE before Generalization (BCM)	RMSE after Generalization with Ensemble ANN (BCM)	Reduction in forecasting error (%)
Aug	0.4900	0.4100	16
Sep	0.7805	0.5805	26
Oct	1.0689	0.689	36
Nov	0.1801	0.1201	33
Dec	0.2838	0.2838	0
Jan	0.2347	0.2347	0
Feb	0.2282	0.2282	0
Mar	0.0909	0.0909	0
Apr	0.2389	0.1389	42
May	0.3215	0.1721	46
Jun	0.2665	0.2665	0
Jul	0.5544	0.3854	30

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Table 4. RE % associated with the output of Ensemble ANN and ARMA models on monthly basis for years 1999 and 2000.

Month	Ensemble ANN Model		Conventional Method ARMA (Salem and Dorrah, 1982)	
	Year		Year	
	1998–1999	1999–2000	1998–1999	1999–2000
Aug	4.70	6.19	–24.80	–27.57
Sep	1.79	–2.04	23.62	29.77
Oct	–6.38	–7.62	–22.49	32.15
Nov	5.76	–6.62	–21.42	–25.53
Dec	–5.26	4.54	27.60	35.07
Jan	0.08	2.01	–26.04	–35.43
Feb	1.91	5.96	31.64	35.78
Mar	6.90	2.60	29.26	–36.14
Apr	4.04	6.10	31.32	21.21
May	4.02	3.85	23.15	21.00
Jun	–0.20	–1.01	–24.31	–22.05
Jul	–6.68	–7.10	34.73	31.01

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Table 5. RMSE associated with Ensemble ANN forecasting model for the period 1961–2000 and the lead time for two months ahead.

Month	RMSE (BCM) Month (t)	RMSE (BCM) Month ($t + 1$)	RMSE (BCM) Month ($t + 2$)
Aug	0.4510	0.6966	0.8957
Sep	0.6385	0.8268	0.1561
Oct	0.7785	0.1441	0.3689
Nov	0.1369	0.3406	0.3051
Dec	0.3121	0.2816	0.2967
Jan	0.2640	0.2738	0.1182
Feb	0.2601	0.1091	0.1806
Mar	0.1027	0.1667	0.2238
Apr	0.1541	0.2066	0.3465
May	0.1996	0.3198	0.5011
Jun	0.2931	0.4625	0.6150
Jul	0.4432	0.5166	0.8127

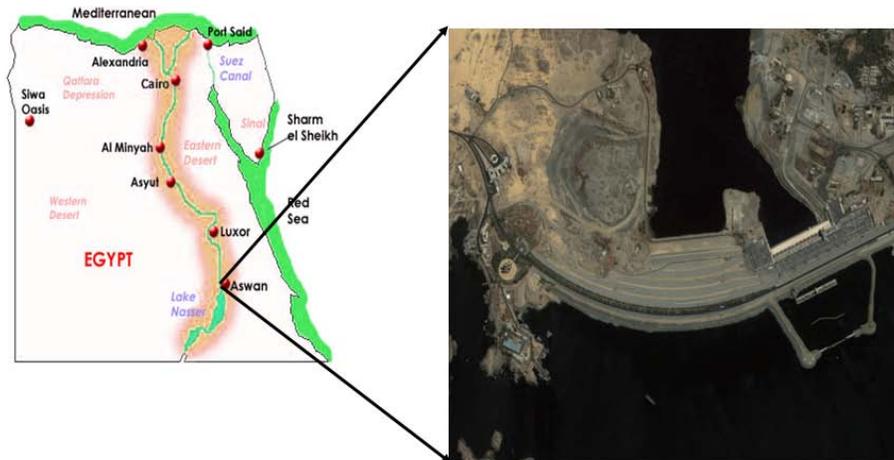


Fig. 1. Location of Aswan High Dam.

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page	
Abstract	Introduction
Conclusions	References
Tables	Figures
⏪	⏩
◀	▶
Back	Close
Full Screen / Esc	
Printer-friendly Version	
Interactive Discussion	



Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Nouredin

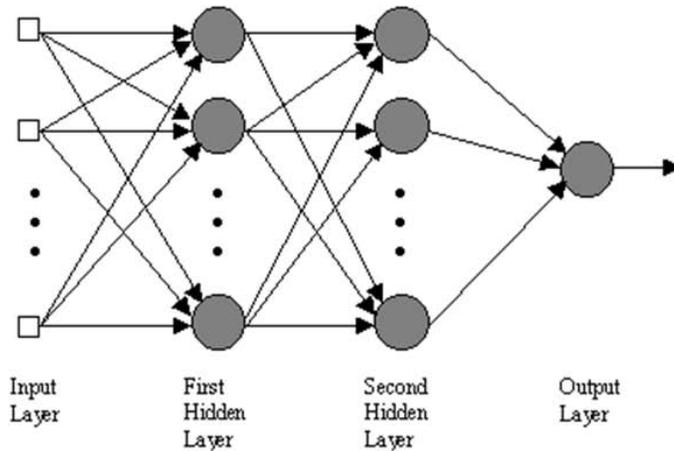


Fig. 2. Artificial Neural Network Model Diagram.

Title Page	
Abstract	Introduction
Conclusions	References
Tables	Figures
⏪	⏩
◀	▶
Back	Close
Full Screen / Esc	
Printer-friendly Version	
Interactive Discussion	

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

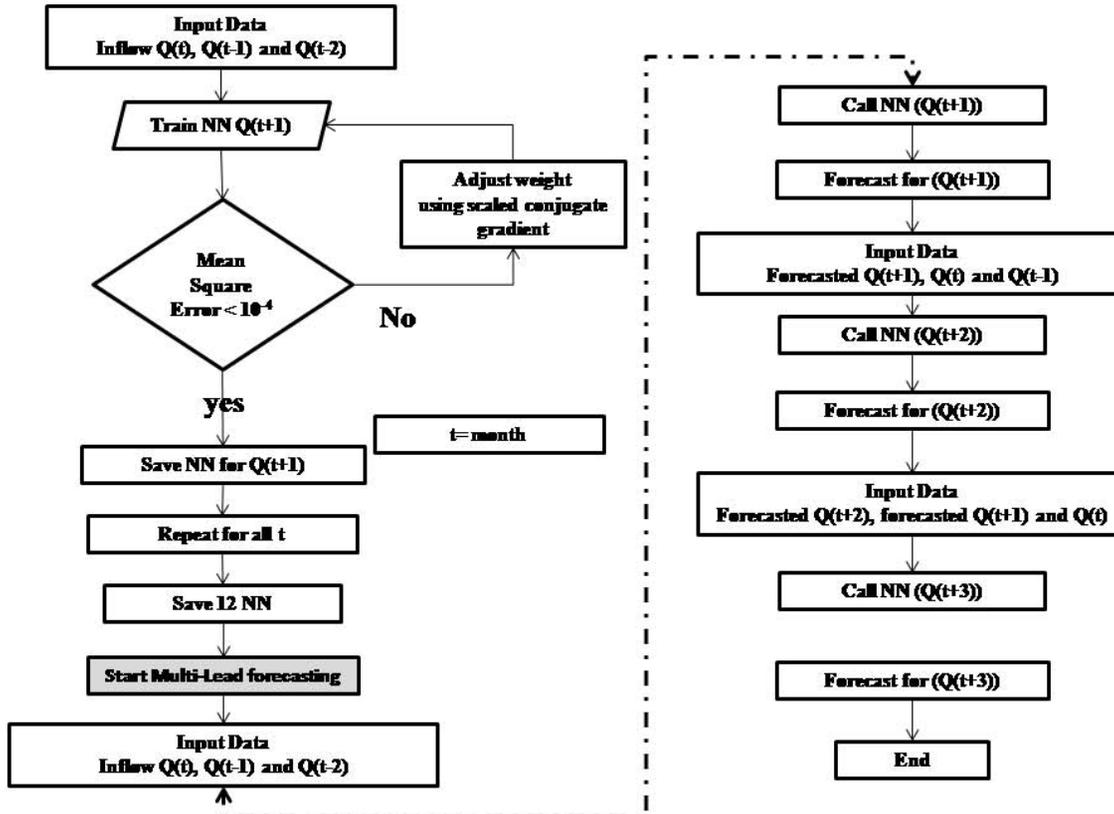


Fig. 3. Flow chart Representation of the Proposed Inflow Forecasting Procedure.

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

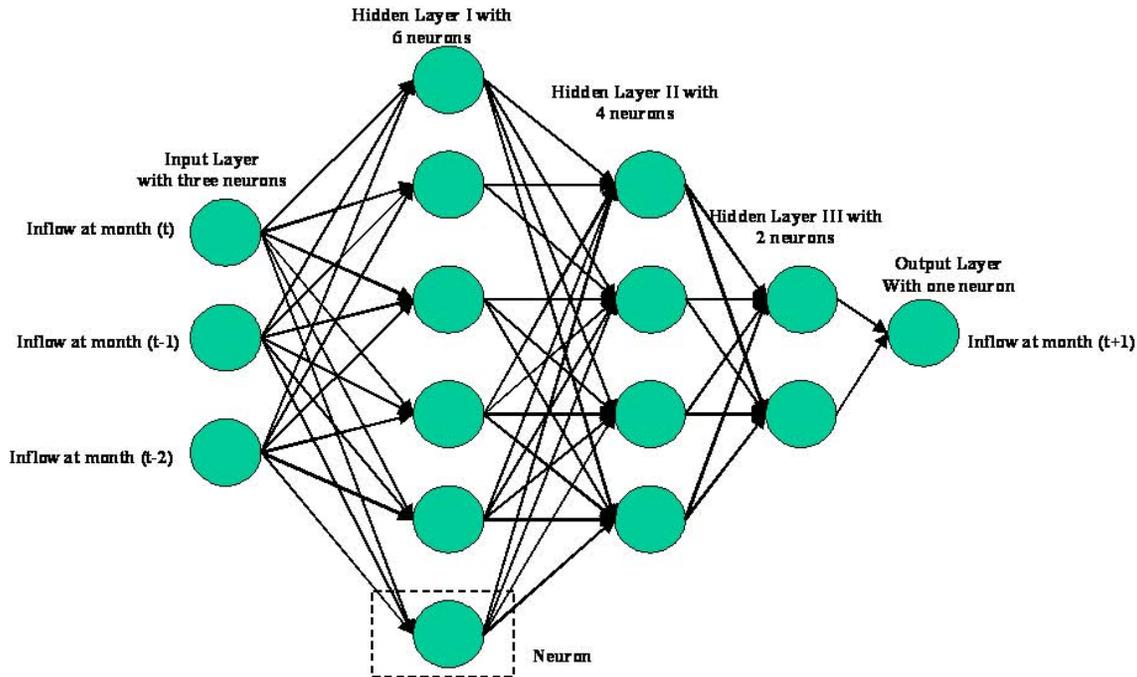


Fig. 4. The Neural Network Architecture for August.

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

Title Page	
Abstract	Introduction
Conclusions	References
Tables	Figures
⏪	⏩
◀	▶
Back	Close
Full Screen / Esc	
Printer-friendly Version	
Interactive Discussion	

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

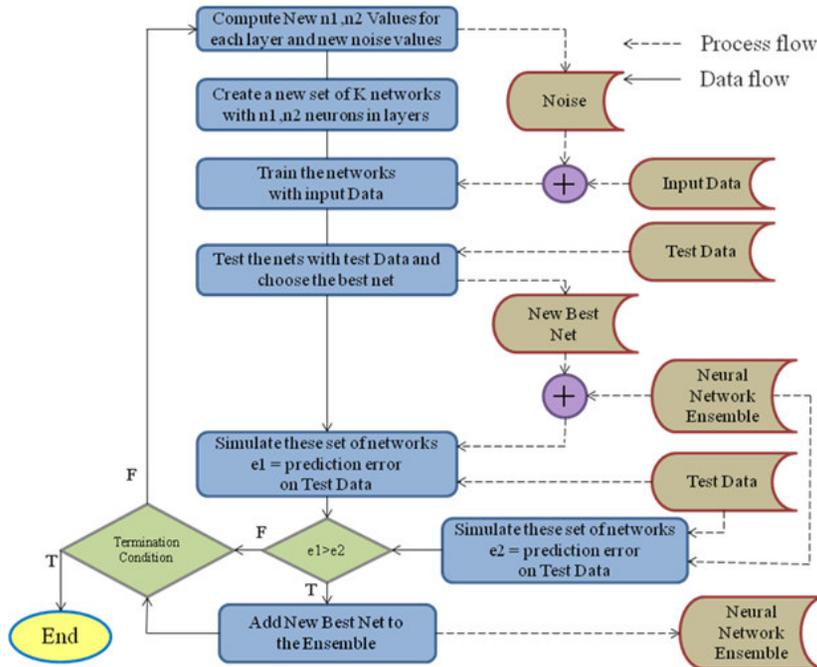


Fig. 5. Learning Phase Process for Ensemble Neural network.

Title Page

Abstract Introduction

Conclusions References

Tables Figures

⏪ ⏩

◀ ▶

Back Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

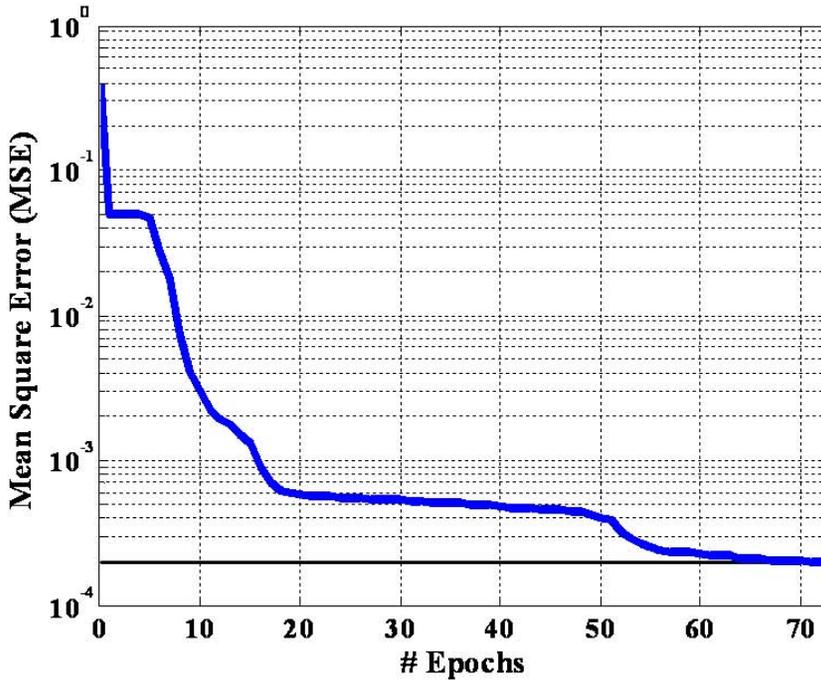
[Title Page](#)[Abstract](#)[Introduction](#)[Conclusions](#)[References](#)[Tables](#)[Figures](#)[⏪](#)[⏩](#)[◀](#)[▶](#)[Back](#)[Close](#)[Full Screen / Esc](#)[Printer-friendly Version](#)[Interactive Discussion](#)

Fig. 6. Training Curve for August.

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

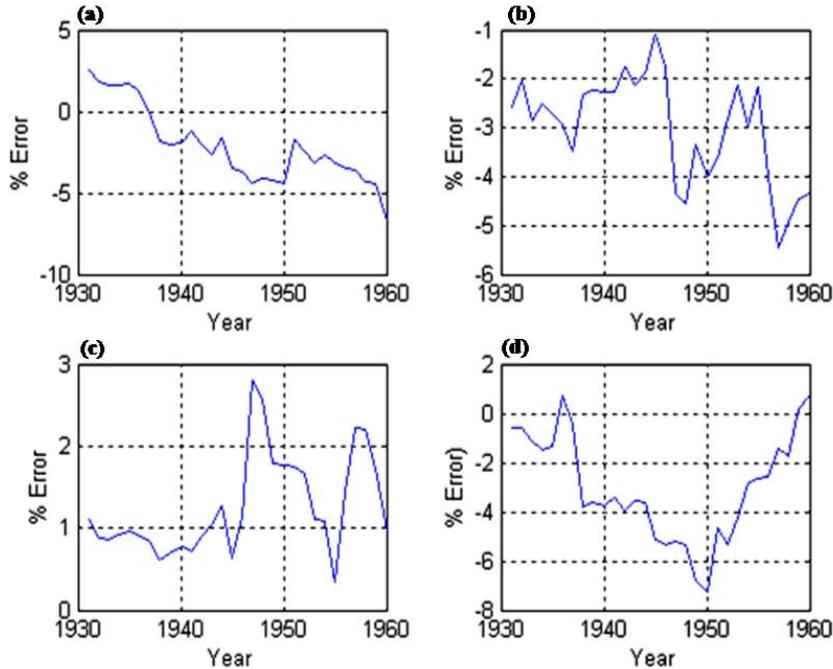


Fig. 7. Error distribution for (a) August (b) December (c) January (d) June.

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

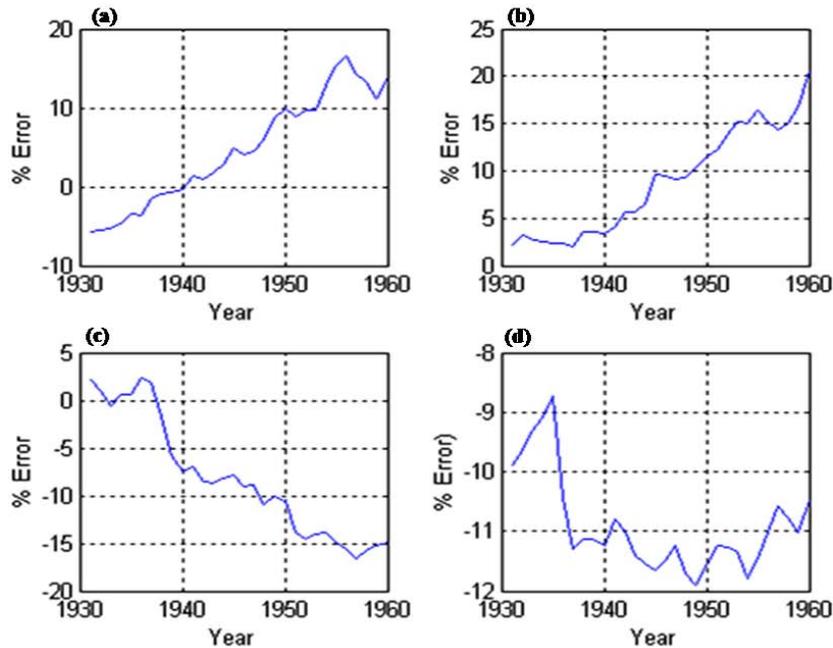


Fig. 8. Error distribution for **(a)** October **(b)** April **(c)** May **(d)** July.

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

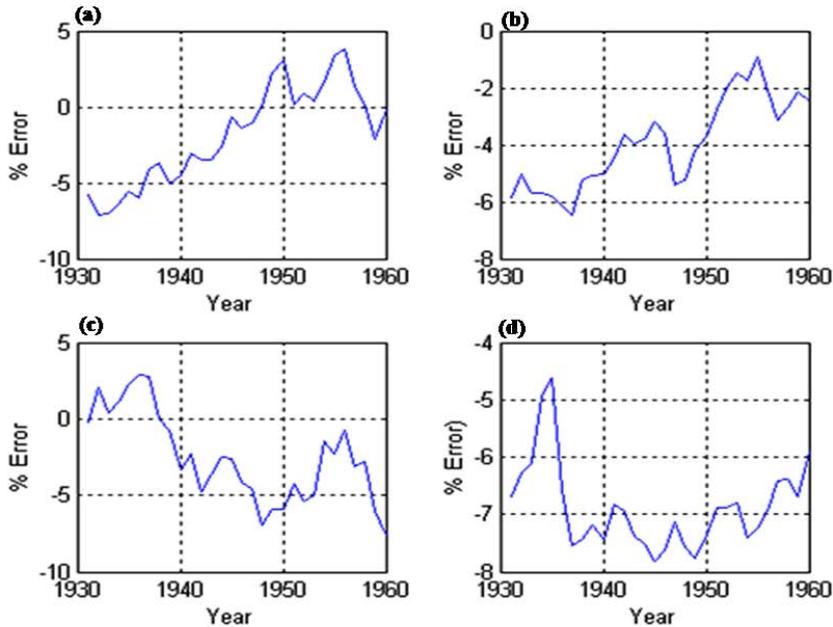


Fig. 9. Error distribution after Regularization for (a) October (b) April (c) May (d) July.

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Generalized versus Non-Generalized Neural Network

A. El-Shafie and A. Noureldin

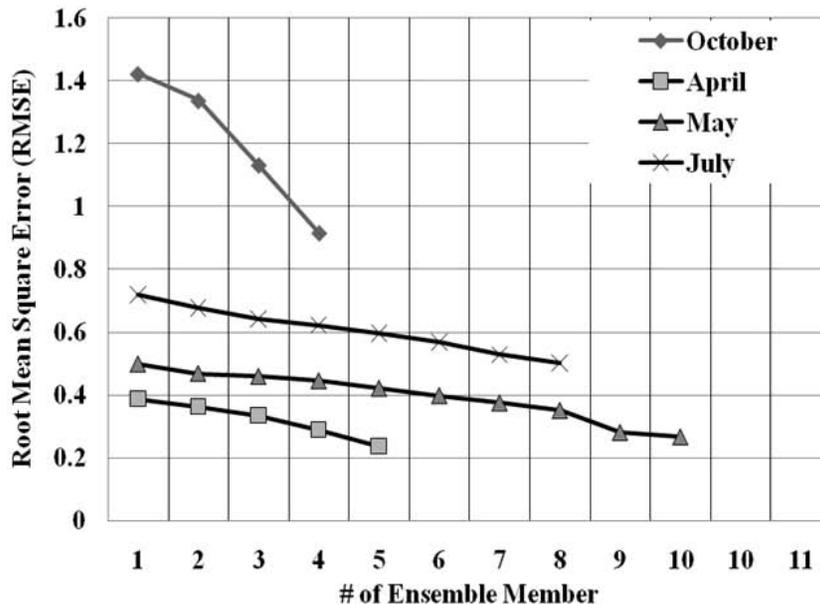


Fig. 10. The Effect of Increasing Number of Ensemble Networks on Root Mean Square Error (RMSE) for last 10 percent of the training data set for Month October, April, May and July during the period between 1871 and 1930.

Discussion Paper | Discussion Paper | Discussion Paper | Discussion Paper | Discussion Paper

Title Page

Abstract Introduction

Conclusions References

Tables Figures

⏪ ⏩

◀ ▶

Back Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion



Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

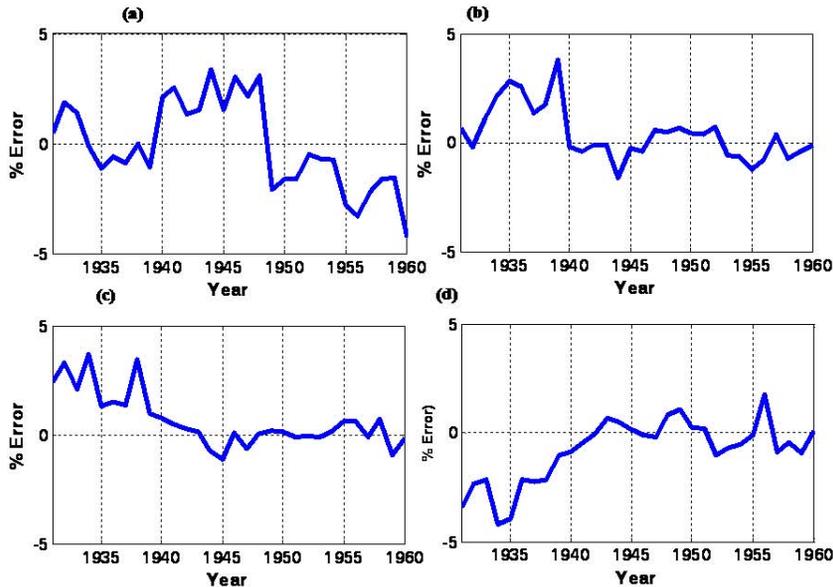


Fig. 11. Error distribution Utilizing Ensemble Neural Network for (a) October (b) April (c) May (d) July.

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures

⏪

⏩

◀

▶

Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion

Generalized versus Non-Generalized Neural Network

A. El-Shafie and
A. Noureldin

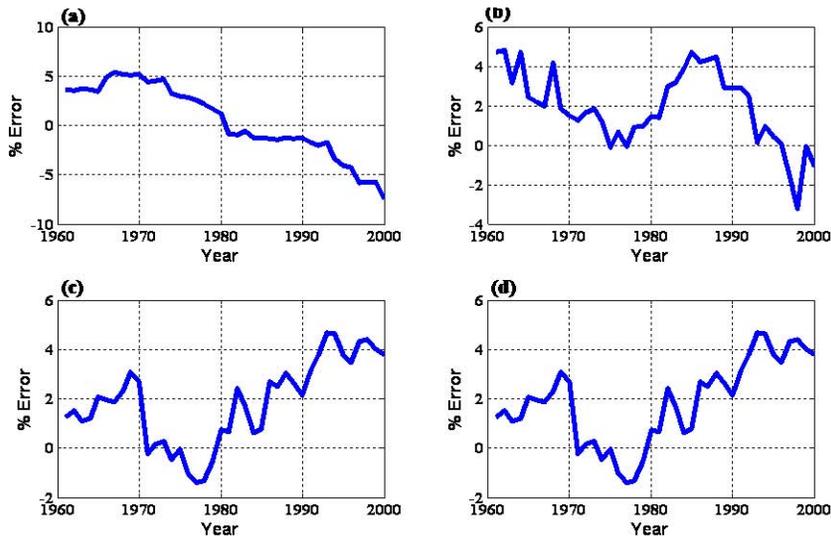


Fig. 12. Error distribution for (a) July (b) June (c) May (d) March.

Title Page

Abstract

Introduction

Conclusions

References

Tables

Figures



Back

Close

Full Screen / Esc

Printer-friendly Version

Interactive Discussion