Vol.3 No: 1 3rd Year 2018 Al-Kut Univ. College Journal مجلة كلية الكوت الجامعة
ISSN: 2616 – 7808 (Online) and ISSN: 2414 – 7419 (print) Pages 15- 31

Using a Hybrid System of Artificial Neural Network with Back Propagation Modified by Genetic Algorithm to Predict Air Temperature

Hind S. Harba and Thaer O. Roomi*
Department of Atmospheric Sciences, College of Science, University of Mustansiriyah
Eman S. Harba
Computer and Internet Unit / College of Arts / University of Baghdad
* th.roomi.atmsc@uomustansiriyah.edu.iq

Abstract

Two models of artificial neural networks with a hybrid backpropagation – genetic algorithm were used to make simulations of some averages of air temperatures above some Iraqi cities. The first model which consists of three layers (input, hidden and output) used to make simulations of the mean daily air temperatures for 11 years in January and July months for Baghdad weather station. The simulated values showed high agreement with the observed values. The root mean square error (RMSE) for January was 0.693 and for July was 0.589. These values prove that the simulation results are quite accurate. The coefficient of determination (R^2) for January was 0.936 and for July was 0.908 which are large enough to support using the model to give good simulations of the mean daily air temperatures. The second model is slightly different than the first one; it uses two hidden layers instead of one. The model run for 33 years' dataset of three monthly averages: mean daily, maximum, and minimum temperatures for four weather stations: Baghdad, Mosul, Basra, and Rutba in January and July. There was a high agreement between the simulated and the observed values especially for July. The percentage errors were (2.459 % to 13.611 %) for January and (0.728 % to 2.314) for July. Cleary, the percentage errors for July are less than that for January which suggests that the data of air temperatures for July are more consistence than that for January. The forecasted values are slightly underestimating the observed ones.

It was found that both models have the strengths to predict the air temperatures averages with just few layers. The researchers think that the genetic algorithm highly improved the skill of the models. The model with two hidden layers shows more accurate results than that of one hidden layer. In the study no significant relationship were found relating to the location of measurements.

Keywords: artificial neural network, back propagation, genetic algorithm, air Temperature.

الملخص

تم استخدام إنموذجين لشبكات عصبية اصطناعية هجينة من خوارزمية انتشار الخلفي وخوارزمية جينية لعمل محاكاة لبعض معدلات درجات الحرارة لبعض المدن العراقية. الإنموذج الأول الذي يتألف من ثلاث طبقات (إدخال، مخفية، واخراج) استخدم لعمل محاكاة للمتوسط اليومي لدرجات الحرارة لمدة 11 سنة ولشهري كانون الثاني وتموز لمحطة بغداد. أظهرت قيم المحاكاة توافق عالي مع القيم المرصودة. كان خطأ جذر متوسط المربعات RMSE لكانون الثاني مساوي الى 0.693 ولتموز 0.589. هذه القيم تثبت بأن نتائج المحاكاة كانت دقيقة جداً. معامل التحديد R^2 لشهر كانون الثاني كان مساوياً الى 0.936 ولشهر تموز مساوي الى 0.908 وهي قيم كبيرة كفاية لدعم استخدام الإنموذج لإعطاء محاكاة جيدة لمتوسط درجة الحرارة اليومية. الإنموذج الثاني مختلف قليلاً عن الأول، فهو يستخدم طبقتين مخفيتين بدل من واحدة. الإنموذج تم تشغيله لبيانات 33 سنة لثلاثة مجاميع من المعدلات الشهرية لدرجات الحرارة لكل من المتوسط اليومي والعظمى والصغرى، لاربعة محطات طقسية هي بغداد والبصرة والموصل والرطبة لكانون الثاني وتموز. كان هناك توافق كبير بين قيم المحاكاة والقيم المرصودة وخاصة في شهر تموز. قيم الخطأ المئوي كانت (2.459% الى 13.611%) لشهر كانون الثاني و(0.728% الى 2.314%) لشهر تموز. وبشكل واضح، قيم الأخطاء المئوية لشهر تموز هي أقل من تلك لشهر كانون الثاني والتي تقترح بأن بيانات درجة حرارة الهواء لشهر تموز هي أكثر تناسقاً من تلك الخاصة بشهر كانون الثاني. كانت القيم المتنبأ عنها أقل بقليل من القيم المرصودة.

وجد بأن كلا الإنموذجين يمتلكان مواضع قوة للتنبؤ عن معدلات درجة الحرارة باستخدام عدد محمدود منن الطبقات. الباحثون يعتقدون بأن الخوارزمية الجينية حسنت بشكل كبير مهارة النماذج. لقد أظهر الأنموذج بطبقتين مخفيتين نتائج أكثر دقة من الأنموذج بطبقة مخفية واحدة. في الدراسة لم يتم ايجاد علاقة محسوسة ترتبط بموقع القياسات.

1. Introduction

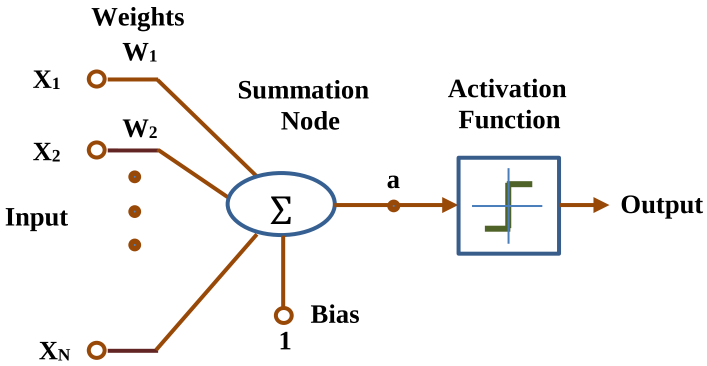
To be able to forecast what would happen in the future has been a dream of human beings since they became aware of their environment and their ability to manipulate it. The numerical prediction is a tool one can use to know what would happen in the next coming time periods numerically with the aid of computers (Singh and Gill, 2014). Weather is the most important phenomenon that people wish to know its prediction especially by numerical weather prediction (NWP) which is very common process to get sophisticated predictions. However, NWP models are so complicated and prone to errors in addition to its non-linear nature.

Artificial Neural Network (ANN) is a simplified model of biological nervous system and has taken the inspiration from the computations performed by human brain. Hence, ANN tries to mimic the behavior of biological neurons and their computations. Strictly speaking about ANN's, they are consisting of many simple processing elements which are interconnected with each other and layered. The best characteristics relating to using ANN's in weather forecasting are being non-linear, adaptive, and powerful (Vikas and Kumar, 2015). The information and knowledge is stored within interneuron connection known as synaptic weight (Malik et al., 2014). The feature of ANN's that they not only analyze the data but also learn from it for future forecasting make them suitable for prediction. These networks enable us to do many useful tasks such as pattern matching, object recognition, mapping capabilities and high speed information processing [Gill and Singh, 2015].

Of course, ANN's are far from the structure and performance of the human brain. The brain consists of about 10 to 15 billion neurons with an average of about one thousand connections each. Today's ANNs have rarely more than a few hundred nodes (Koehn, 1994). Generally, the hierarchical structure of neural networks consists of many layers (Rojas, 1996). The simplest network has one input layer, one hidden layer, and one output layer. The basic computational unit in a neural network is the neuron or perceptron (see Figure 1). The perception has 6 basic elements (Culclasure, 2013):

i. Inputs (X_1, X_2, etc.); ii. An n-vector input weights (W_1, W_2, etc.)

ii. Summing function (Σ); iii. A bias; iv. An activation function; v. An output.



$$a = W_1X_1 + W_2X_2 + W_NX_N + Bias$$

$$Output = Threshold(a)$$

$$where = Threshold(a) = \begin{Bmatrix} -1 \; for \; all \; a \leq 0 \\ 1 \; for \; all \; a \geq 0 \end{Bmatrix}$$

Fig. 1: The basic components of a neuron

2. Back propagation (BP)

The most popular ANN algorithm is Back Propagation (BP) algorithm. This algorithm has the ability to keep knowledge based on the experience and make your knowledge to be used (Caraka, 2017). The idea behind back propagation algorithm is quite simple: the output of ANN is evaluated against desired output. If the results are not satisfactory, the connections between layers are modified and the process is repeated more and more until the error is being small enough (Rojas, 2005). The back propagation is a gradient descent method in which gradient of the error is calculated with respect to the weights for a given input by propagating the error backwards from output layer to hidden layer and further to input layer. Unlike perceptron where step function is used, the sigmoid function is used as an activation threshold for the network (Gill and Singh, 2015).

A) Genetic algorithms (GA)

Genetic algorithms are heuristic search approaches that are applicable to a wide range of optimization problems. Through the evolution, species are able to adapt to their environment. They have developed to complex structures that allow the survival of different kinds of environments. These are good reasons for adapting evolutionary principles to solving optimization problems (Kramer, 2017). GA's were invented by John Holland who presented the genetic algorithm in the 1960s as an abstraction of biological evolution and gave a theoretical structure for adaptation under the GA (Melanie, 1999). Rather than starting from a single point (or guess) within the search space, GA's are initialized with a population of random guesses. The approach uses three operators: selection, crossover and mutation to make the population go to convergence at the global optimum. In selection operator, poorer performing individuals are filtered out and better performing individuals have a greater than average chance of promoting the information they contain within the next generation. Then the crossover permits solutions to exchange information. The method chooses pairs of individuals promoted by the selection operator. It is then randomly chooses a single point within the binary strings and exchanges all the information (digits) to the right of this point between the two individuals. After that mutation is used to randomly swap the value of single bits within individual strings. These three operators will be continued until a fixed number of generations have completed or come to some sort of convergence criterion (Coley, 1999). Neural networks and genetic algorithms represent powerful solving techniques that are based on quite simple principles, but take advantage of their mathematical nature: non-linear iteration. They are two techniques for optimization and learning (Mahajan and Kaur, 2013).

B) Related work

Many studies have been made in the field of environment and weather. Esfandani and Nematzadeh used ANN with BP in hybrid system with GA to predict the air pollution in Tehran based on particulate matter less than 10 microns (PM10) (Esfandany and Nematzadeh, 2016). Culclasure (2013) presented a survey of existing research on applying ANNs to weather prediction. He carried out an experiment in which neural networks are used to regress and classify minimum temperature and maximum gust weather variables.

Harba (2016) developed ANN model to predict the daily maximum and minimum temperatures of month July, 2015. Her model did not give sufficiently accurate prediction.

C) Methodology

In this section, the techniques of artificial neural network and genetic algorithm are used to estimate the air temperatures from forward multilayer network with back propagation algorithm followed by the genetic algorithm which optimizes the forecasting. The study used two datasets for two ANN models.

In the first model (Model I), a dataset of 11 years mean daily temperatures (January and July) for Baghdad weather station was used as input for a network of three layers (input, hidden, and output). This implies that using the data of 2005-2014 to predict the corresponding air temperatures for the year 2015. The predicted data would have compared with actual (targeted) data of 2015.

In the second model (Model II), a dataset of 33 years of monthly averages of each of mean daily temperature, maximum temperature, minimum temperatures (January and July) for four stations. The stations are representative of Iraq and include Baghdad (Middle), Mosul (North), Basra (South), and Rutba (West). The network consists of four layers (one input, two hidden, one output) and developed to predict the temperatures for 2015 by using the dataset from 1982-2014. Figure (2) shows a block diagram of the two models. The algorithms include a procedure on the genetic operators (selection, crossover, and mutation) (See Figure 3).

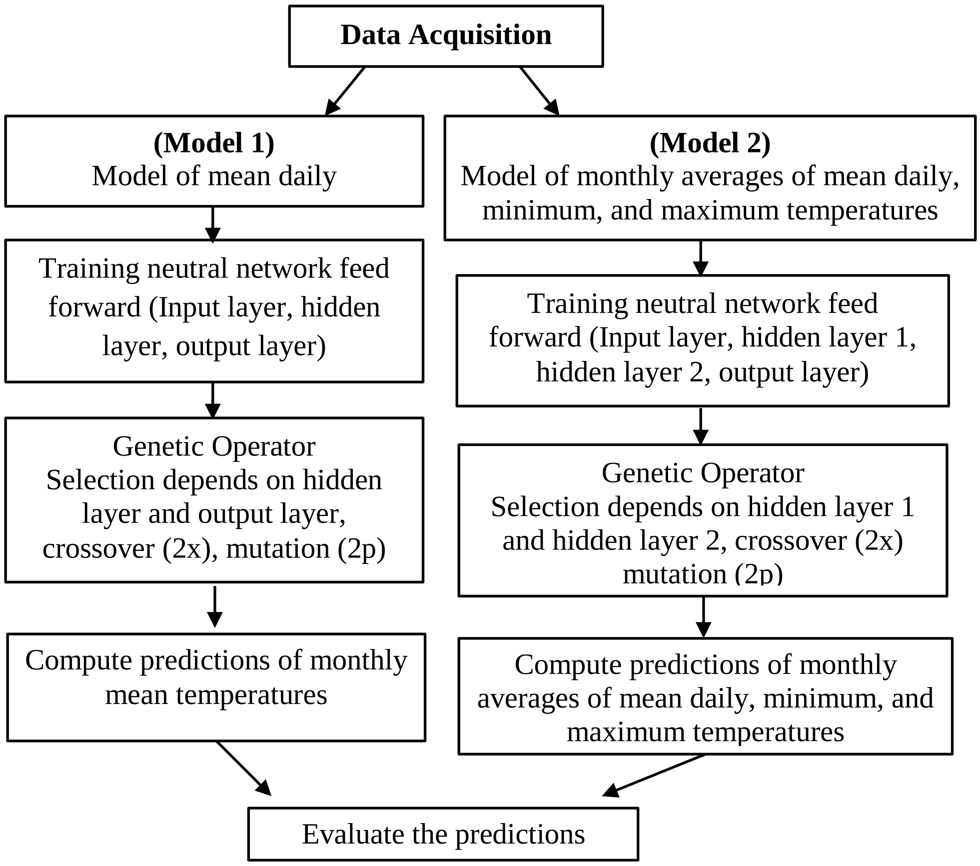


Figure 2: Block diagram of the prediction algorithms.

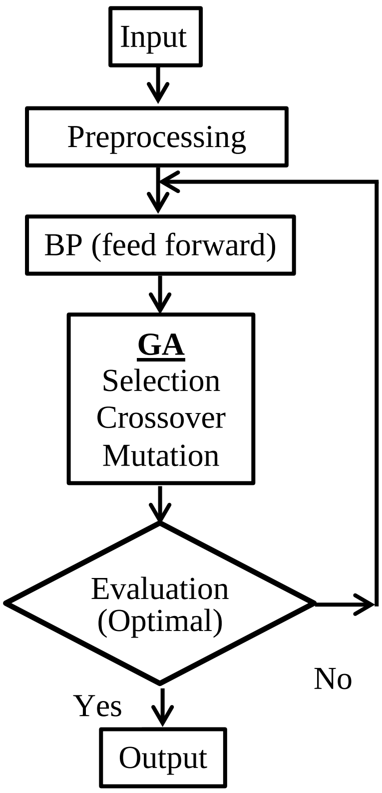


Figure 3: The main processes in the used models

ISSN: 2616 – 7808 (Online) and ISSN: 2414 – 7419 (Print)

D) Model (I) Mean daily temperatures prediction model

The model consists of 3-layers: one input layer, one hidden layer, and one output layer. The network generated 31-day air temperatures of Baghdad station in January and July (see Figure 4).

The models used the feed forward and then back propagation algorithm to train the network by updating all of the biases and weights depending on the obtained errors during the iteration process of learning and optimization. The process would eventually stop when reaching the minimal error or according to a certain number.

The using of genetic algorithm in the models is greatly improved the performance of the network and the predictions by the subsequent modifying of the weights. This proves that the hybrid system of back propagation algorithm with the genetic algorithm is effective in the weather forecasting of air temperatures.

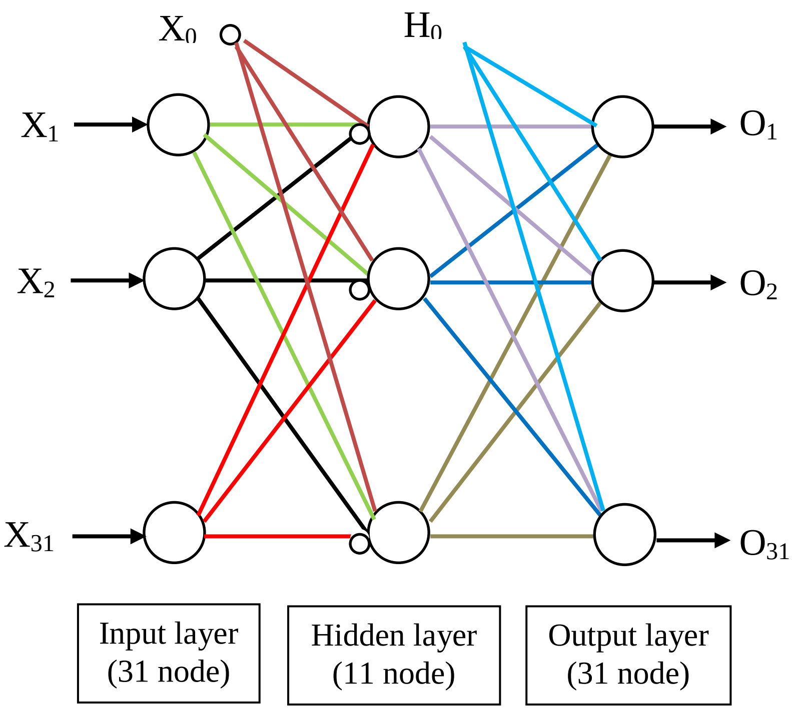


Figure 4: Feed forward neural network of three layers, 10 years, 31 days

The parameters X_1, X_2,…, X_{31} represent the inputs (31 days). There are also 11 nodes in the hidden layer and 31 nodes in the output layer (predicted 31 days). The weights, which represent the strength of the link between the nodes of the three layers, are modified by using the genetic algorithm. These weights were chosen to be between 0 and 1 depending on an experimental basis which gives the best prediction. The Figures (5-8) are examples of typical representations of two chromosomes in the genetic algorithm phase. Figure (5) is a depiction of the two chromosomes containing a hidden layer (H) and an output layer (O). Figures (6 and 7) represent the crossover between the two chromosomes. Genetic operator that includes the selection chooses two individuals (chromosomes) for crossover. In the crossover process two points (denoted 2x) were used. The resultant of the crossover was taken into account to make mutation by replacing the values in two positions randomly within the two cutoff points in one chromosome with two positions in another chromosome in the same order (see Figure 8).

H_1	H_2			…	H_{nin}

O_1	O_2			…	O_{nhid}

Figure 5: Depiction of two chromosomes (H) an (O)

		pc1	pc2		
1	0.78	0.543	0.87	…	H_{nin} (Child1)
0.67	0.99	0.98	0.96	…	O_{nout} (Child2)

Figure 6: Crossover between two chromosomes. pc1 and pc2 represent the cutoff points where switching between a cell from each chromosome is occurring through crossover process

ISSN: 2616 – 7808 (Online) and ISSN: 2414 – 7419 (Print)

0.67	0.78	0.543	0.87	…	H_{nin} (Child1)
1	0.99	0.99	0.96	…	O_{nhid} (Child2)

Figure 7: The result of crossover between two chromosomes

0.67	0.78	0.543	0.87	…	H_{nin} (Child1)
1	0.99	0.98	0.96	…	O_{nhid} (Child2)

Figure 8: Result of the mutation between two chromosomes

E) Model (2) Monthly averages of mean daily, minimum, Maximum temperatures prediction model

The second model uses 2 hidden layers with just one output. The inputs are the monthly averages of 33 years of January and July months for four weather stations: Baghdad, Mosul, Basra, and Rutba. The output is the value that represents the forecasted mean daily, minimum or maximum air temperatures (See Figure 5).

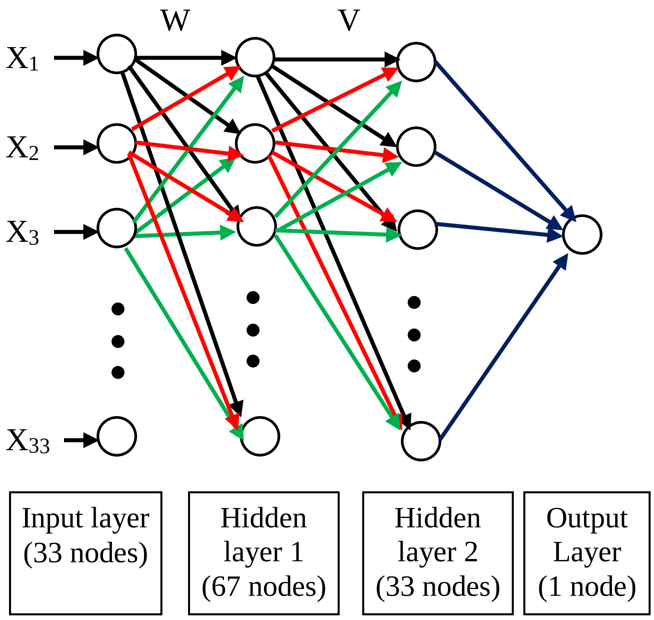


Figure 9: A depiction of the neural network of model

The same genetic algorithm which was used in Model (1) (see Figures (5-8)) was then used in Model (2) but with a certain difference. The three operators (selection, crossover and mutation) are achieved between the first hidden and the second hidden layer and not between the input layer and the hidden layer as in Model (1).

F) The steps of the models

The algorithm of our models can be summarized as:

1. Initialization: Applying inputs to the proposed network and generating random numbers of the initial weights and biases. Note that the activation of node biases is fixed at 1.
2. Forward Pass: calculation of activation:

A. The activation level of an input unit (S_i) is determined by the instance:

$$S_i = X_i, (0 \leq i \leq nin) \qquad (1)$$

Where X_i are input nodes including the bias node ($X_0 = 1$), nin: no of input nodes.

B. The activation level of a hidden unit (H_j) and the output (O_k) of an output unit are determined by the following equations (unipolar Sigmoid function):

$$f(net) = \frac{1}{1 + e^{-\lambda net}} \qquad (2)$$

where $\lambda \geq 1$

$$H_j = \mathrm{f}(net_j) = f\left(\sum_{i=0}^{nin} V_{ij} X_i\right) z \qquad (3)$$

3. Backword Pass (weight training)

Here we modify weights using genetic algorithm GA through genetic operators:

A. Chromosome Initialization: set values of hidden layer (H) and output layer (O).

B. Genetic operators (selection, crossover, and mutation) are set as follows:

(1) Selection of hidden and output chromosomes (H and O).

(2) Choosing two crossover points for crossover operation.

(3) Replace two positions for mutation operation.

C. Evaluation by using fitness function (compute the prediction)

$$P = D_k - O_k \qquad (4)$$

Where D_k is the desired output for node, O_k is the actual output for node.

D. Repeat Steps A and B for all training data.

A MATLAB program was written to achieve the models predictions and all calculations. The results then were evaluated with the actual values of the next year.

3. Results and discussion

A) Model (1): mean daily temperatures prediction model

The results of this model were clarified in Figure (10). The simulated air temperature values are drawn against the observed ones in a scatter graph for January and July, 2015. The simulated values show high agreement. Figure (11) is an illustration of the comparison between forecasted and observed air temperatures against the day number for January and July, 2015. The results clearly show the good agreement between the two sets of data, forecasted and observed. Table (1) lists the criteria that were used to evaluate the results of the simulations by using Model (1). The mean square error MSE is a much more common accuracy measure for field forecasts. Clearly the MSE for a perfectly forecast field is zero, with larger MSE indicating decreasing accuracy of the forecast:

$$MSE = \sum_{i=1}^{n} \frac{(O_i - F_i)^2}{n} \qquad (5)$$

where O_i and F_i refer to the observed and forecasted values, respectively. However, one can use the root mean square error which has the advantage that it retains the units of the forecast variable and is thus more easily interpretable as a typical error magnitude:

$$RMSE = \sqrt{MSE} \qquad (6)$$

Coefficient of determination (R^2) is another measure of the fit of a regression. It is the regression sum of squares divided by the total sum of squares. Also it is alternatively equals one minus the ratio of the error sum of squares to the total sum of squares:

$$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{\sum_{i=1}^{n}(F_i - O_i)^2}{\sum_{i=1}^{n}(F_i - \bar{F})^2} \qquad (7)$$

Since R^2 is a proportion, it is always a number between 0 and 1. R^2 measures the relative sizes of SSE and SST. The smaller SSE, the more reliable the predictions obtained from the model. R^2=1, means the dependent variable can be predicted without error from the independent variable.

It is clear from the Table (1) that the output of Model (1) is that RMSE for January is 0.6934 and for July is 0.5895. These values prove that the simulation results are quite accurate. The coefficient of determination for January is 0.93689 and for July is 0.9088. Hence, R^2 values are large enough that the model is giving good simulations of the mean air temperatures.

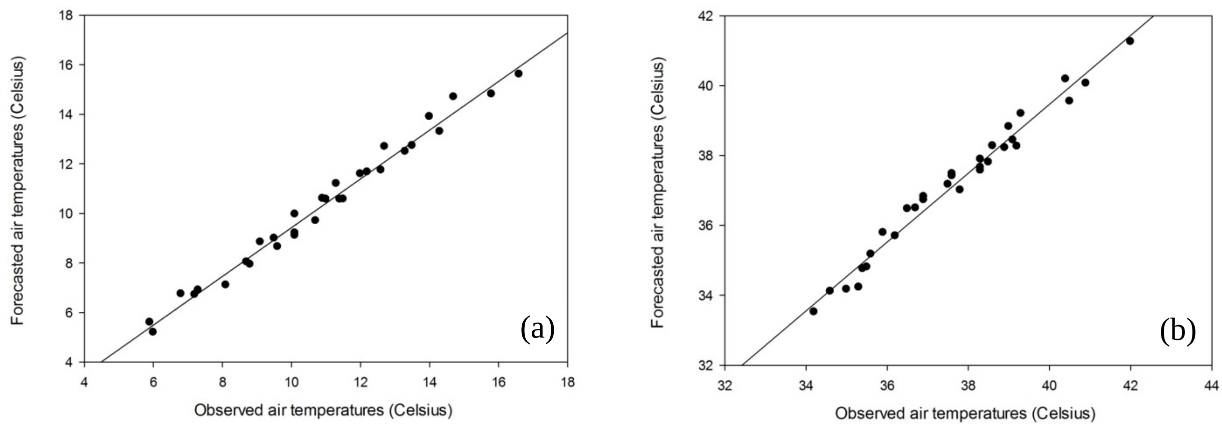


Figure 10: Scatter plots of the forecasted and observed air temperatures for (a) January and (b) July, 2015

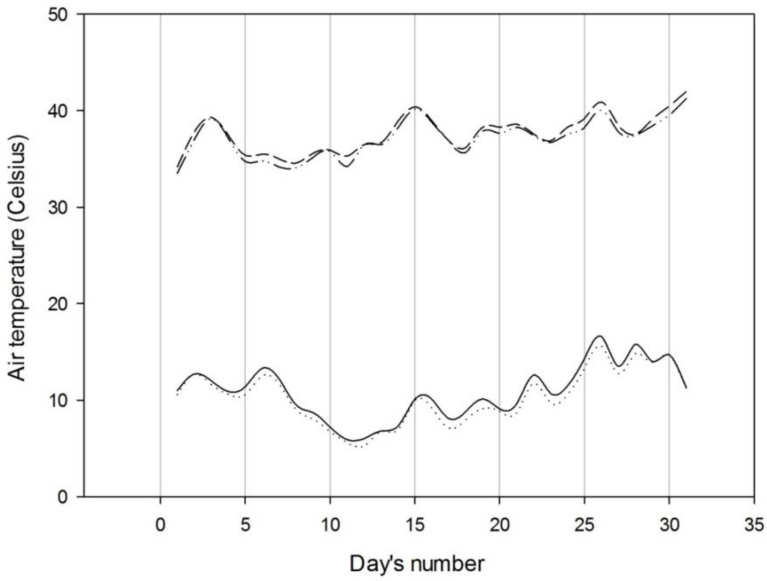


Figure 11: Comparisons between the simulated and the observed mean daily air temperature values for January (solid line for observed, dotted line for simulated) and for July (dashed line for observed, dashed-dotted line for simulated)

Table (1) Root mean square error and coefficient of determination.

Evaluation Criteria	January	July
RMSE	0.6934	0.5895
R2	0.93689	0.9088

B) Model (2): Monthly average of mean daily, maximum, minimum temperatures prediction model

Results of Model (2) are listed in Table (2). The percentage errors are calculated between the forecasted and observed air temperatures for January and July for four weather stations: Baghdad, Mosul, Basra, and Rutba. The percentage errors were (2.459 % to 13.611 %) for January and (0.728 % to 2.314) for July. Cleary, the percentage errors for July are less than that for January, which suggest that the data of air temperatures for July are more consistence than that of January. Figure (12) is a scatter plots for January and July simulations for four stations: Baghdad, Mosul, Basra, and Rutba on the same graph. It is clear that the observed and forecasted values are in good agreement. Figure (13) is a bar chart depicting the comparison between the observed and forecasted values of air temperatures averages for those stations. The forecasted values are slightly underestimating the observed ones.

Table 2 Results and analysis of model (2) simulation.

		January (30 years)			July (30 years)		
	Parameter	**Forecasted**	**Observed**	**Percentage error**	**Forecasted**	**Observed**	**Percentage error**
Baghdad	**Mean Daily Tempt. (T_mean)**	10.350	10.8	4.166	36.951	37.4	1.200
	Maximum Daily Tempt. (T_{max})	16.972	17.4	2.459	45.764	46.1	0.728
	Minimum Daily Tempt. (T_{max})	3.931	4.4	10.659	27.803	28.2	1.407
Mosul	**Mean Daily Tempt. (T_{max})**	8.121	8.6	5.569	35.194	35.7	1.417
	Maximum Daily Temperature (T_max)	14.572	15.1	3.496	42.829	43.4	1.315
	Minimum Daily Tempt. (T_min)	3.110	3.6	13.611	23.640	24.2	2.314
Basra	**Mean Daily Tempt. (T_mean)**	13.592	14.1	3.602	39.690	40.2	1.268
	Maximum Daily Tempt. (T_max)	19.912	20.5	2.868	47.717	48.2	1.002
	Minimum Daily Tempt. (T_min)	8.403	8.9	5.584	31.648	32.2	1.714
Rutba	**Mean Daily Tempt. (T_mean)**	7.761	8.2	5.353	31.175	31.7	1.656
	Maximum Daily Temperature (T_{max})	12.863	13.3	3.285	38.913	39.3	0.984
	Minimum Daily Temperature (T_min)	3.218	3.7	3.027	22.913	23.314	1.72

ISSN: 2616 – 7808 (Online) and ISSN: 2414 – 7419 (Print)

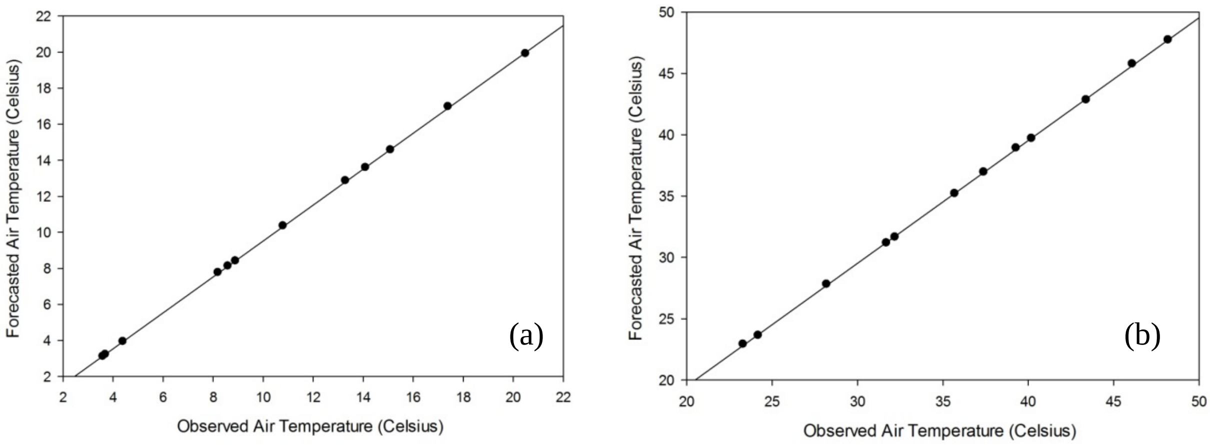


Figure 12: Scatter plots of the forecasted and observed air temperatures for (a) January and (b) July, 2015, for four weather stations: Baghdad, Mosul, Basra, and Rutba on the same graph.

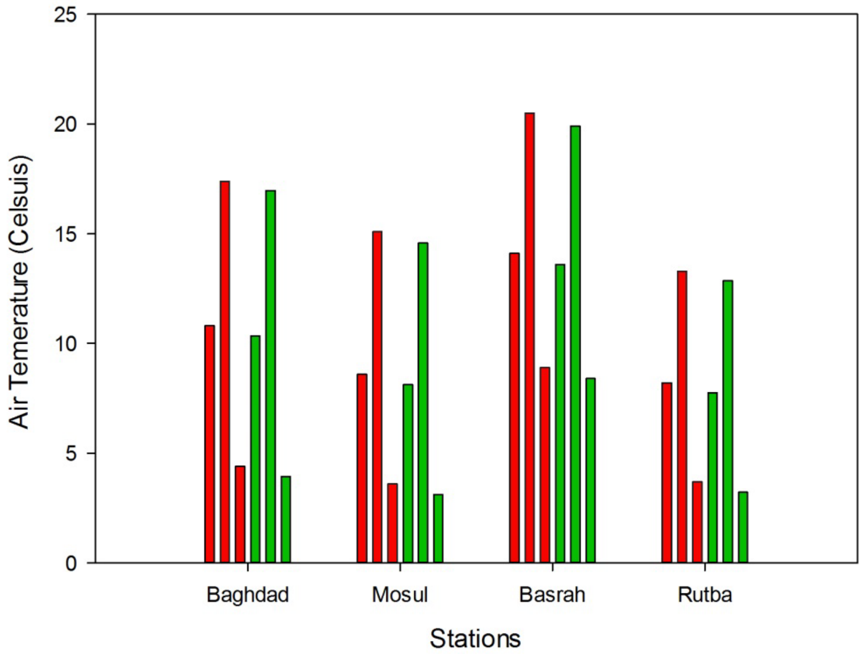


Fig.13: Comparison of the observed (green bars) and forecasted (red bars) air temperature averages: mean, maximum, and minimum, respectively.

Conclusions

1. The simulated and observed mean daily temperatures by using Model (1) show high agreement. RMSE of January was 0.693 and of July was 0.589 which prove that the simulation results are quite accurate. The coefficient of determination for January was 0.936 and for July was 0.908 which are large enough to confirm that the model is a good simulator of the mean air temperatures.

2. Model (2) which simulate the monthly average of mean daily, maximum, and minimum temperatures, shows a reliable agreement between the simulated (forecasted) and the observed values especially for July. The percentage errors for January were (2.459 % to 13.611 %) whereas for July were (0.728 % to 2.314). Cleary, the percentage errors for July are less than that for January which suggests that the data of air temperatures for July are more consistence than that for January. The forecasted values are slightly underestimating the observed ones.
3. It was found that both models have the strengths to predict the air temperatures parameters with just a few layers. The researchers think that the genetic algorithm highly improved the skill of the models. Model (2) shows more accurate results than Model (1) which may due to using two hidden layers.
4. It was found that there is no significant relationship due to the change in the location of measurements.

References

Coley, D. 1999: An introduction to genetic algorithms for scientists and engineers. World Scientific Publishing Co. Pte. Ltd., pp232.

Culclasure, A., 2013: Using neural networks to provide local weather forecasts. M. Sc. Thesis. Georgia Southern University, pp72.

Esfandani, M. A. and H. Nematzadeh, 2016: Predicting air pollution in Tehran: Genetic algorithm and back propagation neural network. *J. AI and Data Mining*, **4**, 49-54.

Gill, J. and S. Singh, 2015: An efficient neural networks based genetic algorithm model for soil temperature prediction. *Inter. J. Emerg. Eng. Res. Tech.*, **3**, 1-5.

Koehn, P., 1994: Combining Genetic Algorithms and Neural Networks: The Encoding Problem. M. Sc. Thesis, The University of Tennessee, Knoxville.

Kramer, O., 2017: *Genetic algorithm essentials*. Springer, Germany, pp92.

Mahajan, R. and G. Kaur, 2013: Neural networks using genetic algorithms. *Inter. J. Comput. Appl.*, **77**, 6-11.

Malik, P., S. Singh, and B. Arora, 2014: An effective weather forecasting using neural network. *Inter. J. Emerg. Eng. Res. Tech.*, **2**, 209-212.

Melanie, M., 1999: *An introduction to genetic algorithms*. MIT Press, USA, pp158.

Rojas, R., 1996: *Neural networks: A systematic Introduction*. Springer, Berlin, Germany, 453pp.

Singh, S. and J. Gill, 2014: Temporal weather prediction using back propagation based genetic algorithm technique. *I.J. Intelligent Sys. Appl.*, **12**, 55-61.

Vikas, S. and K. Kumar, 2015: A Symmetric key cryptography using genetic algorithm and error back propagation neural network. *2nd International Conference on Computing for Sustainable Global Development (India).* 1386-1391.