Journal of Agricultural Science; Vol. 9, No. 5; 2017
ISSN 1916-9752 E-ISSN 1916-9760
Published by Canadian Center of Science and Education

A Methodological Proposal Based on Artificial Neural Networks for Evapotranspiration Assessment

Alberto B. Mirambell[1], Clayton F. da Silva[2], Flavio de Souza Barbosa[2] & Celso Bandeira de Melo Ribeiro[1]

[1] Environmental and Sanitary Engineering Department, Federal University of Juiz de Fora, Minas Gerais, Brazil

[2] Computational and Applied Mechanics Department, Federal University of Juiz de Fora, Minas Gerais, Brazil

Correspondence: Celso Bandeira de Melo Ribeiro, Environmental and Sanitary Engineering Department, Federal University of Juiz de Fora, Rua José Lourenço Kelmer, s/n, São Pedro, 36036-900, Juiz de Fora, Minas Gerais, Brazil. Tel: 32-2102-3429(210). E-mail: celso.bandeira@ufjf.edu.br

Received: December 4, 2016 Accepted: March 15, 2017 Online Published: April 15, 2017

doi:10.5539/jas.v9n5p142 URL: https://doi.org/10.5539/jas.v9n5p142

Abstract

Evapotranspiration is the combined process in which water is transferred from the soil by evaporation and through the plants by transpiration to the atmosphere. Therefore, it is a central parameter in Agriculture since it expresses the amount of water to be returned by irrigation. Aiming to standardize Evapotranspiration estimate, the term "reference crop evapotranspiration (ETo)" was coined as the rate of Evapotranspiration from a hypothetical grass surface of uniform height, actively growing, completely shading the ground and well watered. ETo can be measured with lysimeters or estimated by mathematical approaches. Although, Penman-Monteith FAO 56 (PM) is the recommended method to estimate ETo by PM, it is necessary to register maximum and minimum temperatures (ºC), solar radiation (hours), relative humidity (%) and wind speed (m/seg.). Some of these parameters are missing in the historical meteorological registers. Here, Artificial Neural Networks (ANNs) can aid traditional methodologies. ANNs learn, recognise patterns and generalise complex relationships among large datasets to produce meaningful results even when input data is wrong or incomplete. The target of this study is to assess ANNs capability to estimatie ETo values. We have built and tested several architectures guided by Levenberg-Marquardt algorithm with 5 above mentioned parameters as inputs, from 1 to 50 hidden nodes and 1 parameter as output. Architectures with 10, 15 and 20 nodes in the hidden layer brought outsanding r^2 values: 0.935, 0.937, 0.937 along with the highest intercept and the lowest slope values, which demonstrate that ANNs approach was an afficient method to estimate ETo.

Keywords: evapotranspiration, Artificial Neural Networks, Penman-Monteith FAO 56, estimation, performance

1. Introduction

1.1 Evapotranspiration

Evapotranspiration (*ET*) is an important component of the hydrologic cycle. Its estimation plays a central role in different fields related to hydrology such as water balance, impact of land uses assessment, water resources planning and management and irrigation system design. Evapotranspiration is the physical process where water is transferred to the atmosphere both by evaporation from land and by transpiration from plants, so it is a combined process through which moisture returns to the atmosphere.

ET had always been a concept widely used as far as water resources management was concerned. There was still some ambiguity in the use of such terms as potential *ET* and reference crop *ET* though. In order to solve this issue, the Food and Agricultural Organization (FAO) of the United Nations and by means of the publication of the "FAO Irrigation and Drainage Paper No. 56" in 1990, shed some light on the problem, helping users with uniformity on the use of such terminology.

ET rate from a well-watered reference surface was called as "reference crop ET" or "Reference Evapotranspiration" and denoted as *ETo* (Allen & FAO, 1998). A reference surface, for the FAO must be a hypothetical surface *"closely resembling an extensive surface of green grass of uniform height, actively growing, completely shading the ground and with adequate water"*. Doorenbos and Pruitt (1977) defined reference crop evapotranspiration rate as *"the rate of evapotranspiration from an extensive surface of 0.8-1.5 m tall, green grass cover of uniform height, actively growing, completely shading the ground and no short of water"*.

A widespread method to estimate *ET* from a well-watered agricultural crop is to first estimate *ETo* and then to apply the appropriate empirical crop coefficient (*Kc*), which accounts for the difference between the reference surface and the actual crop. *ETo* can be directly measured using lysimeters by measuring different water balance components with highly construction and maintenance costs (Allen & FAO, 1998). Mostly, the use of lysimeters is considered as a time-consuming method and needs previous experience, so it is not a common choice on field researches.

A more affordable alternative to this method is the application of mathematical approaches based on several climate parameters, which are divided into empirical and physical models. The former are based on statistical functions of approximation between meteorological parameters and *ETo* values (Blaney & Criddle, 1950; Hargreaves & Samani, 1985; Jensen & Haise, 1963; Thornthwaite, 1948).

On the other hand, physical models are based on physical principles associated with the three most influential factors for *ET*: the amount of energy, the water vapor flux and the supply of moisture (Chow, Maidment, & Mays, 1988). The most representative of these methods is the Penman combined process (Penman, 1948). This approach relates evaporation dynamics to net radiation flux and aerodynamic transport characteristics of a natural surface. Observing that latent heat transference through plants is not only influenced by abiotic factors, Monteith introduced a surface conductance term that accounted for the response of leaf stomata to its hydrologic environment. This modified form of the Penman equation is widely known as the Penman-Monteith evapotranspiration model (Monteith, 1965).

Many scientists have proved the reliability of the Penman-Monteith FAO 56 (*PM56*) method for estimating *ETo* (Allen, 1986; Allen, Jensen, Wright, & Burman, 1989; Chiew, Kamaladasa, Malano, & McMahon, 1995; Irmak, Irmak, Allen, & Jones, 2003; Itenfisu, Elliott, Allen, & Walter, 2003; Kashyap & Panda, 2001; McNaughton & Jarvis, 1984; Souza & Yoder, 1994). Due to this backing, *PM56* was ranked in 1990 as the best model at estimating *ETo* by the FAO. Several studies have demonstrated that *ETo* estimates obtained by *PM*56 model significantly resembles to observed *ETo* values (Allen, Smith, Pereira, & Perrier, 1994); (Ventura, Spano, Duce, & Snyder, 1999); (Howell, Evett, Schneider, Dusek, & Copeland, 2000); (Wright, Allen, & Howell, 2000).

Penman-Monteith FAO 56 equation has basically two advantages: First, it is applicable to a wide range of climates and local conditions with no need to be calibrated; second, it is a method previously validated using lysimeters. It also has a drawback, though; *ET* is a complex and nonlinear phenomenon due to its dependency on several interconnected climatological parameters, such as air temperature (*Tmax*, *Tmin*), mean relative humidity (*HRm*), wind speed (U_2) and insolation as sunshine hours (*Ins*). A very common problem among researchers on this field is that only temperatures have been broadly registered via weather stations since the fifties. In other words, historical meteorological datasets are usually incomplete. In order to solve this issue we propose the use of Artificial Neural Networks (*ANNs*) as a high performance tool at estimating ETo values, both with all inputs are available and when the dataset is incomplete.

1.2 ANN's and Its Application on Water Resources Field

Artificial Neural Networks (*ANNs*) are mathematical models whose architecture has been inspired by biological neural networks and are considered as very appropriate tools for modelling nonlinear processes. This is the case of Evaportanspiration, a process influenced by several climatical parameters which does not behave linearly and that justifies the methodology proposed on this paper and its suitability to it. *ANNs* are capable of learning from examples, recognising repeated patterns and generalising complex relationships among a large amount of data to produce eventually meaningful results, even when input data contains errors or is incomplete, which is the problem this study expects to solve. The main advantage of the *ANNs* is its ability of solving problems which are difficult to formalize (Sudheer, Gosain, & Ramasastri, 2005). *ANNs* allow us to capture deep complex characteristics of data (Galvão, Becerra, Calado, & Silva, 2004).

In terms of internal structure and operations, an *ANN* basically consists in three layers: an input, a hidden and an output layer, each of them composed by an array of processing elements (*PE*). A *PE* is a model whose components are analogous to the elements of an actual neuron. An *ANN* is a network where all its components in a given layer are interconected to all components in the next layer, but not between elements within the same layer. Input data is stored in the input layer, in fact, each input parameter gets into and it is stored in the *ANN* through a neuron or node. That is to say that each parameter is represented by a neuron.

The function of this first layer is to provide information to the network. At the entrance of the hidden layer, all input parameters randomly receive a weight, which ranks them in terms of importance according to the model the *ANN* is trying to simulate. This represents the first part of a processing element; the second part consist in a nonlinear filter, usually called "transfer function". Its aim is to limit the output values between two asymptotes.

The most common transfer function is the sigmoidal function. It is a function that varies gradually between two asymptotic values, typically 0 and 1 or -1 and +1.

The hidden layer is actually which allows the network to model complex functions. The number of nodes compounding the hidden layer is determined by trial and error. There could be more than one hidden layer but the use of a single hidden layer along with a sigmoidal transfer function is widely recognised as the most frequent network topology (Cybenko, 1989; Hornik, Stinchcombe, & White, 1989).

Finally, the output layer can be understood as "the exit door" for values predicted by the network and it is composed only by one node, the output. The type of network topology described heres is known as Multilayer Perceptron (*MLP*) (Fausset, 1994). Within the *ANN*, as the sign spreads forward layer-by-layer (feed-forward) the error values propagate backwards (backpropagation) for a better adjustment of the weights (synaptic weight adjustments). Three different algorithms can be applied: Quasi-Newton (*Q-N*) (Haykin, 1994); Levenberg-Marquardt (*L-M*), (Hagan & Menhaj, 1994); Backpropagation with variable learning rate (*BPVL*) (Soares & Nadal, 1999).

An extensive review about several *ANN* applications in hydrology field is gathered in ASCE Task Committee (2000) and (Govindaraju & Rao, 2000). Furthermore, recent studies on *ANNs* applications in this area include rainfall–runoff modelling (Lin & Chen, 2004; Wilby, Abrahart, & Dawson, 2003); river stage forecasting (Campolo, Soldati, & Andreussi, 2003; Imrie, Durucan, & Korre, 2000; Lekkas, Imrie, & Lees, 2001); reservoir operation (Jain, Das, & Srivastava, 1999); land drainage design (Shukla, Kok, Prasher, Clark, & Lacroix, 1996; Yang, Lacroix, & Prasher, 1998); aquifer parameter estimation (Lingireddy, 1998); describing soil water retention curve (Sudheer & Jain, 2004) and optimization or control problems (Bhattacharya, Lobbrecht, & Solomatine, 2003; Wen & Lee, 1998). Some of those studies have also shown that *ANNs* are even more accurate than conventional methods in flow forecasting and drainage design (Yang, Prasher, & Lacroix, 1996; Zealand, Burn, & Simonovic, 1999).

ANNs can be also applied for estimating *ETo*. Several studies have been carried out during the past decade aiming the estimation of evapotranspiration through the application of *ANNs* (Chauhan & Shrivastava, 2009; Dai, Shi, Li, Ouyang, & Huo, 2009; Izadifar & Elshorbagy, 2010; Jain, Nayak, & Sudheer, 2008; Jothiprakash, RamaChandran, & Shanmuganathan, 2002; Kisi, 2006a, 2006b, 2006c, 2007, 2008; Kisi & Öztürk, 2007; Kumar, Raghuwanshi, Singh, Wallender, & Pruitt, 2002; Sudheer et al., 2005; Trajkovic, Todorovic, & Stankovic, 2003; Traore, Wang, & Kerh, 2010; Zanetti, Sousa, Oliveira, Almeida, & Bernardo, 2007).

Kumar et al. (2002) concluded that *ANNs* trained with lysimetric data performed better than *PM* equation in terms of *ETo* estimates. Sudheer et al. (2005) used a radial basis function neural network for estimating actual *ET* from limited data and compared with lysimeter *ETo* values. Kisi (2006a) used feed-forward neural networks with Levenberg-Marquardt training algorithm for estimating *ETo*. Results were compared with those obtained through linear regression. Zanetti et al. (2007) applied Levenberg-Marquardt training algorithm in *ANNs* for the estimation of *ETo* with minimum and maximum temperatures. *ANNs* have also been used for forecasting monthly *ETo* values (Tahir, 1998; Trajkovic et al., 2003).

All these studies are somewhat related to our proposal since they point out *ANNs* as useful tools at estimating *ETo*. Consequently, we also expect to get accurate *ETo* values when applying *ANNs* onto our incomplete dataset. Here, the scope of our study is to demonstrate that *ANNs* are an efficient methodology to estimate *ETo* values at high performance bringing accurate results even when the dataset is incomplete, as it is in our case.

2. Materials and Methods

2.1 Study Area and Climatic Dataset

In order to fulfil the targets of this study, daily meteorological dataset for a 16-years period ranging from January 1st, 2000 to December 31st, 2015 was obtained from a weather station belonged to the INMET (Meteorological National Institute of Brazil), located in Juiz de Fora (Minas Gerais), in the southeast of Brazil (latitude 21°45′S, altitude 43°20′W, elevation 939.96). The location is shown in Figure 1.

The climate of the area has been classified as Humid Subtropical (Cwa) according to Köppen climate classification (Geiger, 1961; Köppen, 1884, 1918, 1936, 2011). The weather station corresponding to the World Meteorological Organization code 83692 and INMET code A518 is specifically located within the Federal University of Juiz de Fora (UFJF), where an average annual rainfall of 1536 mm. has been registered for the last decades.

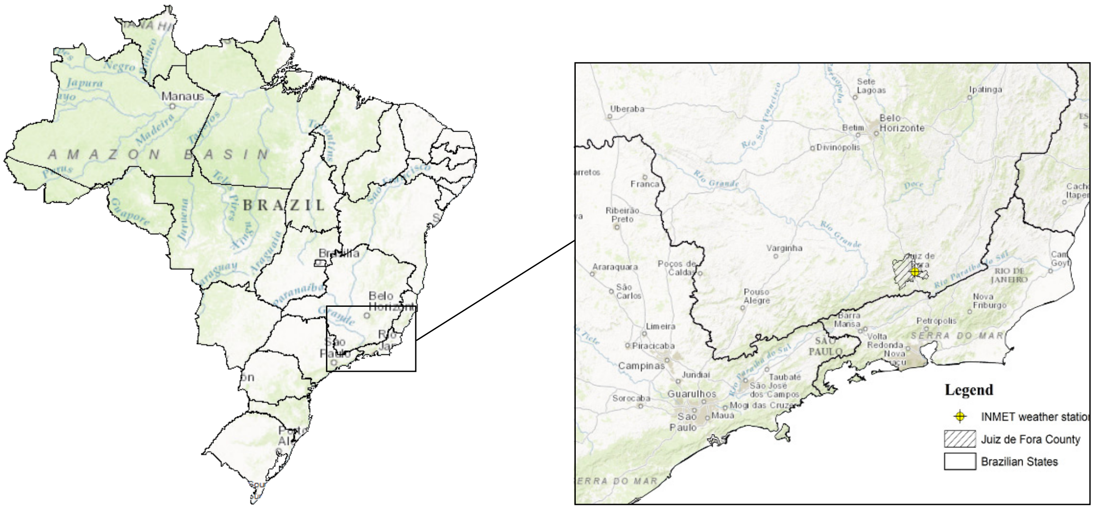

Figure 1. Study area: Juiz de Fora County, Minas Gerais, Brazil

Based on INMET information and dataset registered, January is the most humid month in Juiz de Fora, with roughly 20% (298 mm.) of the accumulated rainfall, whereas August usually registers precipitations around 16.5 mm. As far as temperatures are concerned, February is the hottest month with temperatures around 26 °C as an average. Oppositely, July is the coldest month since the temperatures drop off to 20 °C as an average. The daily mean relative humidity does not oscillate much during the year as ranges from 74% (August) to 86% (July), with a high annual average of 82%. Wind speed is usually lowest through the first part of the year (January-June) and it ranges from 2.6 to 27 with an average of 8 m/seg.

2.2 Penman-Monteith FAO 56 Method

As it was said in the introduction, Penman-Monteith FAO 56 (*PM*) is the model which best estimates *ETo* according to FAO and it is described as it follows (Allen & FAO, 1998):

$$ETo^{PM} = \frac{0.408\Delta(R_n G) + \gamma \frac{900}{T+273} U_2 (e_s - e_a)}{\Delta + \gamma(1 + 0.34U_2)} \tag{1}$$

Where, ETo^{PM} = Reference Crop Evapotranspiration estimated by *PM* method (mm/day); Δ = slope vapour pressure curve (kPa/°C); *Rn* = net radiation at the crop surface (MJ/m²day); *G* = soil heat lux (MJ/m²day); γ = psychrometric constant (kPa/°C); *T* = mean daily air temperature (ºC); U_2 = wind speed at 2 meters height (m/seg.); *es* = saturation vapour pressure (kPa); and *ea* = actual vapour pressure (kPa).

In this study, we applied the *PM* equation on daily measures of some climatic parametres as maximum air temperature (*Tmax*), *minimum air tempearture* (*Tmin*), mean relative humidity (*RHm*), wind speed (U_2) and insolation as sunshine hours (*Ins*). It gets necessary, at this point, to draw readers' attention on what the use of daily measures implies for soil heat flux (*G*). The quantity of soil heat flux (*G*) gained and lost throughout a 24-hours period is assumed to be approximately the same, so, at the end of the day, *G* = 0. *Rn*, Δ, *es* and *ea* were calculated using the equations given by Allen and FAO (1998) and gathered in the FAO Irrigation and Drainage Paper No. 56; *HRm*, *Tmax*, and *Tmin* were collected from an INMET weather station located within the study area and were the substrate to calculate *ea* and *es*; γ was calculated from the altitude of the weather station under study.

Here, *PM* method has been presented as the standard and the most used methodology to estimate *ETo* but we proposed *ANNs* as an alternative method to estimate *ETo* starting from the same amount of data. In other words, *ETo* values estimated by *PM* method (*ETo*(*PM*)) were used as reference for its comparison with *ETo* values estimated by *ANNs* (*ETo*(*ANN*)).

2.3 Estimating ETo through ANNs

The same weather dataset encompassed within January 1st, 2000 and December 31st, 2015 was used to apply the different *ANNs* built in order to test its potentiality for estimating *ETo*. Each daily measure, which comes into the network through the input layer, is considered as a pattern of the climatic behaviour within the study area and it is from this dataset where the network learns from.

It is usual to find missing measures within large datasets. In our case, 14.55% of the total register was missing. Despite this, a total of 4994 daily measures (85.45% of the entire period) were available to perform the networks. Thus, the dataset was split into three subsets:

- 70% of it was destined for training; during this phase, the network is trained to associate outputs (*ETo*(*PM*)) with input patterns (*Tmax*, *Tmin*, *RHm*, U_2, *Ins*). This process is usually known as Learning Process, as it requires a memorization process of the wide variety of input patterns and its associated outputs.

- The 30% spare was divided into two equal parts and used for validation and testing, respectively; during validation, the network emulates what it learned before and tries to perform the best on new patterns. It is here where the most important application of neural networks takes place: Pattern Recognision; during this step, the network identifies the input pattern and tries to produce the associated output. The power of neural networks makes sense when a pattern that has no output associated with it, is given as an input (testing phase). In this case, the network gives the output that corresponds to a taught input pattern that is least different from the given pattern. That is to say that during the testing phase the network estimates *ETo* values basing on what it learnt previously during the Learning Process.

2.4 Building and Running an ANN

Firstly, we built *ANNs* with an input layer composed by 5 input parameters: maximum air temperature (Tmax), minimum air temperature (*Tmin*), insolation as sunshine hours (*Ins*), mean rlative humidity (*RHm*) and mean wind speed (U_2). Latitude (decimal degrees) and altitude (meters above sea) were excluded from the study as they were considered as constants. The function of this first layer is to provide information to the network.

Secondly, as far as the nodes that compound the hidden layer (*Hi* in Figure 2) were concerned, networks with 1, 2, 3, 4, 5, 10, 15, 20, 25, 30, 35, 40, 45 and 50 nodes were trained. The number of nodes in the hidden layer is a recurrent issue as there is no an "absolute truth" about it (Coulibaly, Anctil, & Bobee, 2000). Indeed, some studies on the issue revealed that large-than-necessary networks tend to over-fit the training samples and bring poor performances. Some researches demonstrated that one hidden layer is enough to represent the non-linear relationship between the climatic parameters and *ETo* (Arca, Beniscasa, & Vincenzi, 2001; Kumar et al., 2002). At the entrance of the hidden layer, all input parameters randomly received a weight (Wij in Figure 2), which ranked them in terms of importance accordingly to the model the *ANN* is trying to simulate.

Finally, the output layer was only composed by one node instead: daily *ETo* values estimated by *PM* method (*ETo*(*PM*)), considered as the target values. The complete architecture just described is depicted in Figure 2.

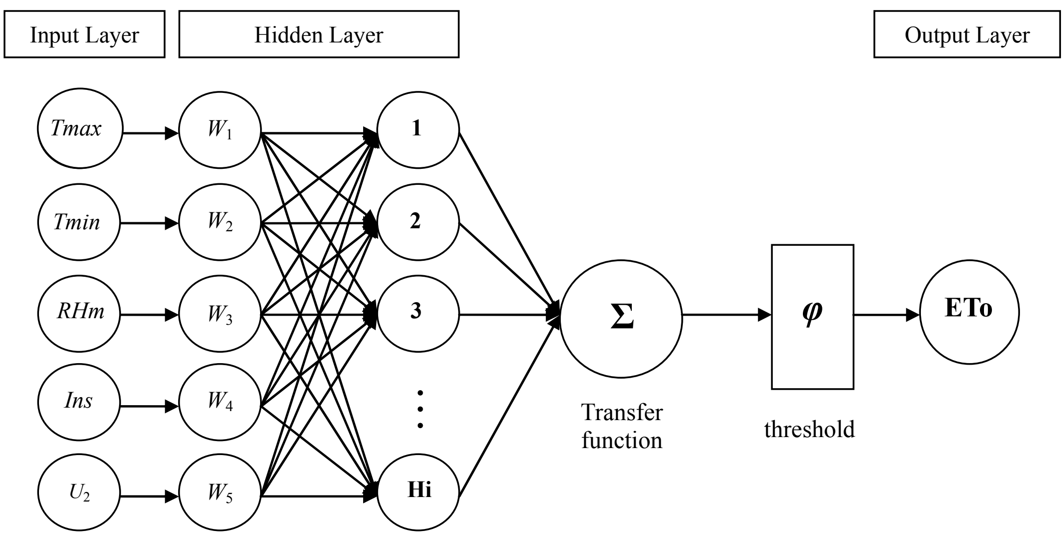

Figure 2. Basic architecture of an Artificial Neural Network

To accomplish this task, the software Matlab R2015a was used, both for applying Penman-Monteith method to estimate *ETo* values and for building and running of *ANNs*. Aiming to assess the *ANNs* performance when estimating *ETo* values (*ETo*(*ANN*)), the following statistical indices were applied:

$$MSE = \frac{\sum_{i=1}^{n}(x_i - y_i)^2}{n} \tag{2}$$

$$r^2 = \left\{ \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sigma_x \sigma_y} \right\}^2 \tag{3}$$

$$Y = mX + b \tag{4}$$

Where, *MSE* = Mean Squared Error (mm/day); *n* = number of observations; x_i = *ETo* (mm/day) estimated by *PM* (*ETo*(*PM*)); y_i = *ETo* (mm/day) estimated by the *ANNs* (*ETo*(*ANN*)); r^2 = *determination coefficient*; $\bar{x}$ = mean of x_i; $\bar{y}$ = mean of y_i; σ_x = standard deviation of x_i ; σ_y = standard deviation of y_i; *Y* = estimatted values (*ETo*(*ANN*)); *X* = target values (*ETo(PM)*); *m* and *b* = *Slope* and *Intercept* of the line of best fit between *ETo*(*PM*) and *ETo*(*ANN*).

This study began with the target of estimating *ETo* by means of another methodology instead of using the standard method, Penman-Monteith, as there were climatic parameters missing that did not allow the calculation of *ETo*.

INMET weather stations usually present non-registered periods or isolated blanks for some of the meteorological parameters monitored that can last from some days to entire years. Here, different *ANN* architectures were built, trained, tested with missing inputs. Once some inputs were removed (*RHm*, U_2), *ANNs* performances were compared with *ETo*(*PM*) by means of some statistical indices which allowed us to quantify the adjustment level between target values (*ETo*(*PM*)) and forecasting values (*ETo*(*ANN*)) as it follows**:**

$$c = dr \tag{5}$$

$$d = 1 - \frac{\sum_{i=1}^{n}(y_i - x_i)^2}{\sum_{i=1}^{n}(|y_i - \bar{x}|) + (|x_i - \bar{x}|)^2} \tag{6}$$

$$r = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sigma_x \sigma_y} \tag{7}$$

Where, *c* = performance index (Camargo & Sentelhas, 1997); *d* = adjustment coefficient (Willmott, 1981); *r* = correlation coefficient.

3. Results and Discussion

Different networks were built with a variety of nodes in the hidden layer (*ANN* architecture, Table 1) and the statistical indices described above are shown in Table 1 for each of them. *MSE* values acoount for the average difference between the target and the predicted values, so express the quality of an estimator, which is the *ANN*. The closer to 0, the better is the estimator and its performance. Learning cycles refers to the number of trials the network performs to reduce the differences previously mentioned to a minimum.

Table 1. *ANNs* performance indices

ANN* architecture	***MSE* validation**	**Learning cycles**	**r^2**	***m***	***b***	***ETo*(*PM*)-*ETo*(*ANN*)**
5_1_1	0.112	128	0.880	0.88	0.37	-0.052
5_2_1	0.089	162	0.899	0.89	0.33	0.010
5_3_1	0.074	34	0.906	0.91	0.28	0.074
5_4_1	0.083	13	0.906	0.91	0.29	0.057
5_5_1	0.079	20	0.910	0.91	0.28	0.051
5_10_1	0.146	10	0.935	0.93	0.24	-0.055
5_15_1	0.157	7	0.937	0.93	0.23	-0.068
5_20_1	0.156	20	0.937	0.92	0.24	0.028
5_25_1	0.159	17	0.935	0.91	0.26	0.066
5_30_1	0.145	6	0.929	0.95	0.19	0.037
5_35_1	0.158	13	0.931	0.92	0.30	-0.035
5_40_1	0.160	5	0.937	0.93	0.25	0.053
5_45_1	0.148	6	0.933	0.93	0.27	0.023
5_50_1	0.158	7	0.935	0.92	0.26	-0.073

Note. * ANN architecture: They all were built based on this topology: In_Hi_Op, where In = parameters in the input layer; Hi = nodes in the hidden layer; Op = output. *ETo*(*PM*)-*ETo*(*ANN*): difference between *ETo* values estimated by *PM* method and *ETo* values predicted by the *ANN*.

Architectures with 15 and 20 nodes in the hidden layer presented the highest r^2 values, 0.937 accompanied with a value of 0.935 for netwoks with 10 and 25 nodes in the hidden layer as is shown in Table 1. In other words, networks containing 10, 15, 20 and 25 hidden nodes could be considered as those reaching the minimum *MSE* values along with the highest r^2 values. In fact, except networks with 1 and 2 nodes in the hidden layer, all r^2 values obtained were above of 0.90 which is considered by most authors as an excellent *ANN* performance.

According to *MSE* values, networks with 2, 3, 4 and 5 neurons would be apparently the most appropriate for *ETo* estimates, as they present the lowest *MSE* values: 0.089, 0.074, 0.083 and 0.079, respectively. Furthermore, several authors used *MSE* as the criterion to assess *ANNs* topology (Jain et al., 2008; Kumar et al., 2002; Zanetti et al., 2007).

However, if we focus only on *MSE* values, the results obtained in this study and shown in Table 1 can lead us to a misunderstanding since the same networks showing the minimum *MSE* values also needed greater number of learning cycles and larger r^2 values. Although, 10, 15 and 20 hidden-nodes networks brought *MSE* values of 0.146, 0.157, 0.156, respectively along with less learning cycles required and much higher r^2 values, so these architectures were considered as the most suitable to estimate *ETo* and their performances are shown in Figure 3.

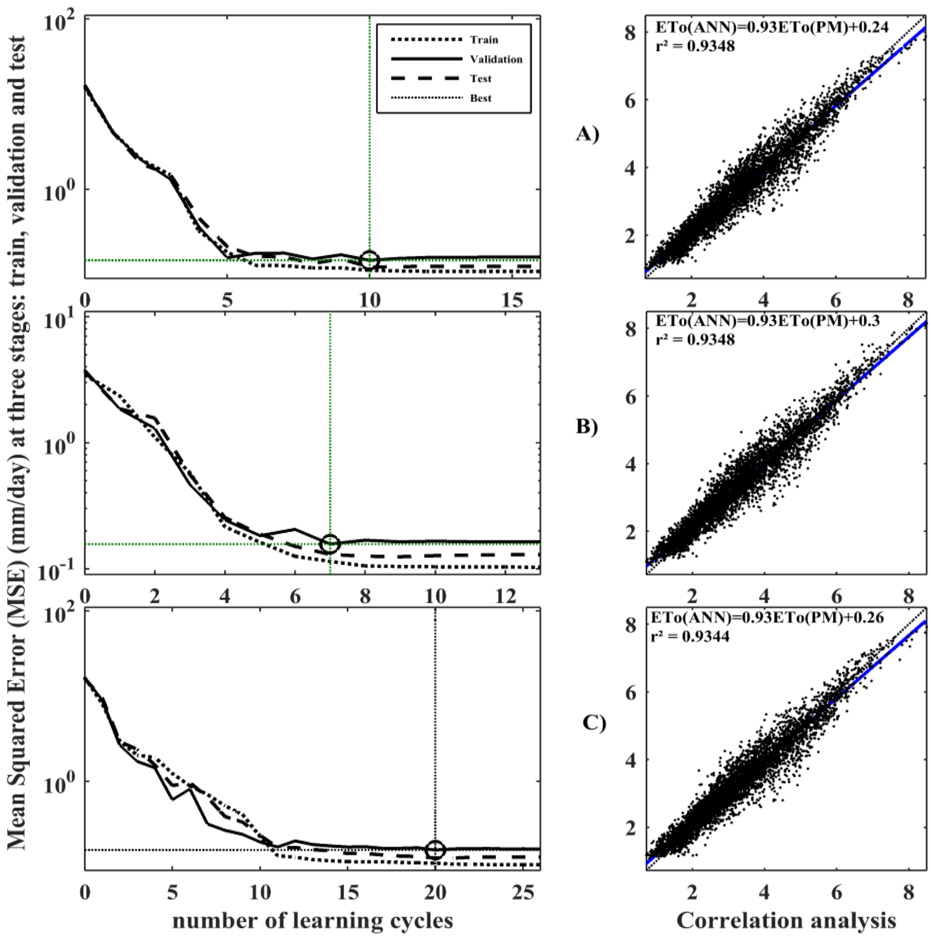

Figure 3. Performances of 10, 15 and 20-hidden-nodes architectures. Performance in terms of *MSE*, numbers of learning cycles needed to reach the minimum *MSE* value and determination coefficient (r^2) from correlation analyses for (A) 5_10_1, (B) 5_15_1 and (C) 5_20_1 topologies

Still focused on *MSE* values, we can check in Table 1 that an extraordinary value of 0.145 was obtained for a 30-hidden-nodes network. In concordance with it, the number of learning cycles required to reach the minimum error at validation stage gave us a raw idea of which *ANN* architectures neither over-fit the data nor underestimate the results.

As far as number of learning cycles are concerned, networks with 10, 15, 30, 40, 45 and 50 were ranked as the ones with less completed learning cycles: 10, 7, 6, 5, 6 and 7, namely. Again, 30-hidden-nodes network brought an unexpected result; 6 learning cycles were required to reach such a low *MSE* value as it is shown in Table 1 and clearly visible in Figure 4, A1. In addition, *Slope* (*m*) and *Intercept* (*b*) were also displayed in Table 1 so as to confirm the outstanding behaviour of 30-nodes network, which delivered remarkable values of 0.95 and 0.19, corresponding to the maximum *slope* and the minimum *intercept* among the architectures assessed. On the other hand, the r^2 value for this network (0.929) was actually the poorest among the architectures under study. Further analyses were carried out in order to determine wether 30-hidden-nodes network was, indeed, the best architecture or just an outlier.

After this unexpected performance of a 5_30_1 network, we carried out some further analyses. Training, validating and testing again the same 5_30_1 topology brought different results: now the network needed 36 learning cycles (6 times more) to reach the minimum *MSE*, which changed from 0.145 (Figure 4, A1) to a value of 0.195 (Figure 4, B1), the highest among the architectures under study. The difference between the *ETo*(*PM*) and *ETo*(*ANN*) can be checked in Figure 4, A3 and B3. It is presented as a histogram where all patterns were clustered in 20 classes or bins around the line of zero error. It gave us an idea on how accurate *ANNs* can be, since most bins yielded a difference below 0.1 mm/day.

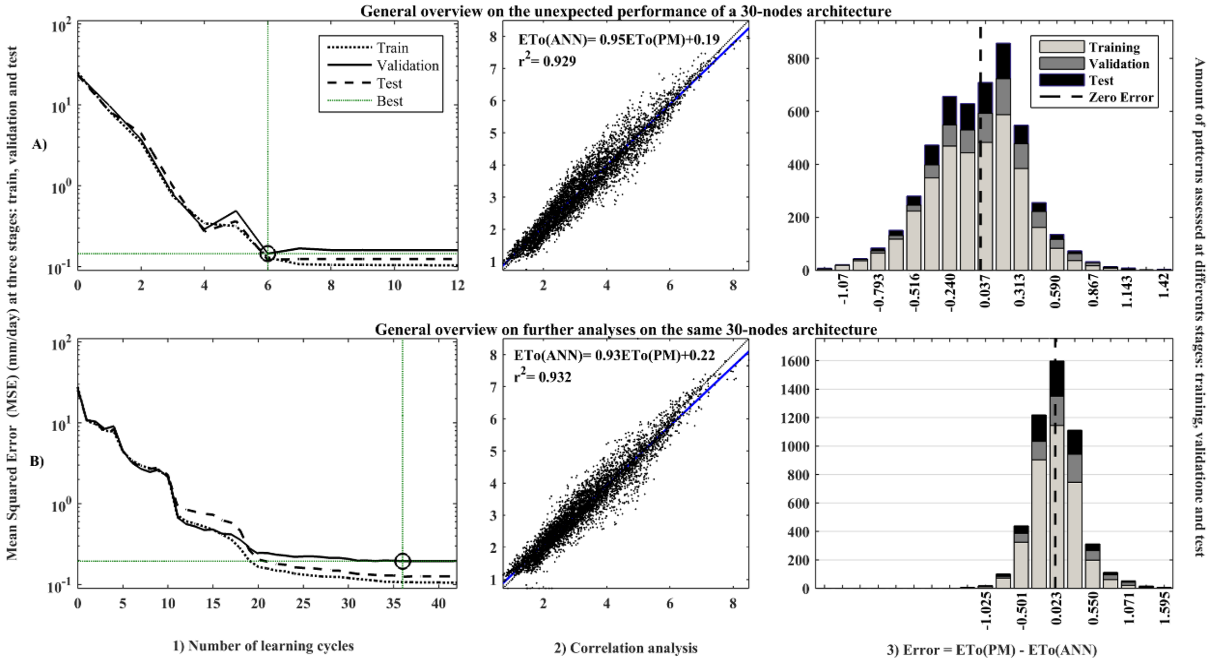

Figure 4. Unexpected behaviour of a 30-hidden-nodes architecture and results of further analyses. Comparison between (A) the first set of results of a 5_30_1 network and (B) further analyses. 1) Number of learning cycles required to reach the minimum *MSE* value; 2) Correlation analysis showing r^2, *Slope* and *Intercept* values; 3) *ETo*(*PM*)-*ETo*(*ANN*) histograms divided into 20 bins at three stages: train, validation and test pointing out the zero error point

Slope and *Intercept* moved to completely opposite extreme values: 0.93 and 0.22, namely, the lowest *slope* and the highest *intercept* (Figure 4, B2). Suprisingly, r^2 slightly increased until 0.932 (Figure 4, B2) and *ETo*(*PM*)-*ETo*(*ANN*) decreased meaningfully from 0.037 (Figure 4, A3) to 0.023 (Figure 4, B3).

Taking into account these results and probably requiring deeper analyses, we cannot consider a 30-hidden-nodes architecture as a trustworthy network, due to the poor *Slope* and *Intercept* values, the high *MSE* values and the large amount of learning cycles to reach it.

At this point, it gets necessary to remind that the main problem this study wanted to solve was to estimate *ETo* values using *ANNs* even when dataset is erroneous or incomplete. Until now, all the analyses carried out were applied on architectures with 5 input parameters: *Tmax*, *Tmin*, *RHm*, *Ins* and U_2, so no paramaters were missing. As mentioned in the introduction, *RHm* and U_2 were constantly lacking in the historical meteorological register used along this study. The lack of these two climatic parameters throughout our dataset was actually the reason why we considered *ANNs* as an approach to estimate *ETo* since Penman-Monteith method cannot be applied when there are missing inputs. As a secondary research, U_2 and *RHm* were removed from the input layer, as a one-at-a-time process in order to check wether it was still possible to estimate *ETo* at a high resemblance level.

Here, we built *ANNs* with 10, 15, 20, 30 nodes in the hidden layer, 1 output only (*ETo*(*PM*)) and with no U_2 as input (4U_2_10/15/20/30_1); with no *RHm* as input (4RHm_10/15/20/30_1); with neither U_2 nor *RHm* as inputs (3_10/15/20/30_1); and finally with only *Ins* as an input (1_10/15/20/30_1). Their performances were assessed using some statistical indices as Performance Index (c), Adjustment Coefficient (d) and Regression Coefficient (r), previously presented in the Materials and Methods section.

The comparison brought outstanding *c* and *d* values: 0.929 and 0.976, respectively for a 4U_2_15_1 architecture; 0.909 and 0.968 for a 4RHm_30_1 architecture; 0.890 and 0.961 for a 3_30_1 architecture. 1_10/15/20/30_1 network was rejected due to its low *r*, *c* and *d* values. That means that an *ANN* with only three input parameters (*Tmax*, *Tmin* and *Ins*) can offer excellent adjustments (Camargo & Sentelhas, 1997) as it is visible in Figure 5C. According to what it can be inferred from Figure 5 and confirmed by *c* and *d* values in Table 2, *ETo* values estimated by *ANNs* are highly resembled to those estimated by *PM* method, even when both U_2 and *RHm* are not available.

Table 2. Performance indices from architectures with missing inputs

Missing inputs	Architectures	c	d	r
None	5_10_1	0.95	0.983	0.967
	5_15_1	0.952	0.983	0.968
	5_20_1	0.952	0.983	0.968
	5_30_1	0.948	0.982	0.965
U_2	4_10_1	0.928	0.975	0.952
	4_15_1	0.929	0.976	0.952
	4_20_1	0.928	0.975	0.952
	4_30_1	0.921	0.972	0.948
RHm	4_10_1	0.907	0.967	0.938
	4_15_1	0.908	0.968	0.939
	4_20_1	0.909	0.968	0.939
	4_30_1	0.909	0.968	0.939
U_2*, *RHm	3_10_1	0.888	0.960	0.925
	3_15_1	0.890	0.961	0.927
	3_20_1	0.890	0.960	0.926
	3_30_1	0.890	0.961	0.927
$T^a max$*, *$T^a min$*, *U_2*, *RHm	1_10_1	0.664	0.859	0.773
	1_15_1	0.665	0.860	0.774
	1_20_1	0.663	0.858	0.773
	1_30_1	0.666	0.86	0.774

Note. c: performance index; d: adjustment coefficient; r: regression coefficient.

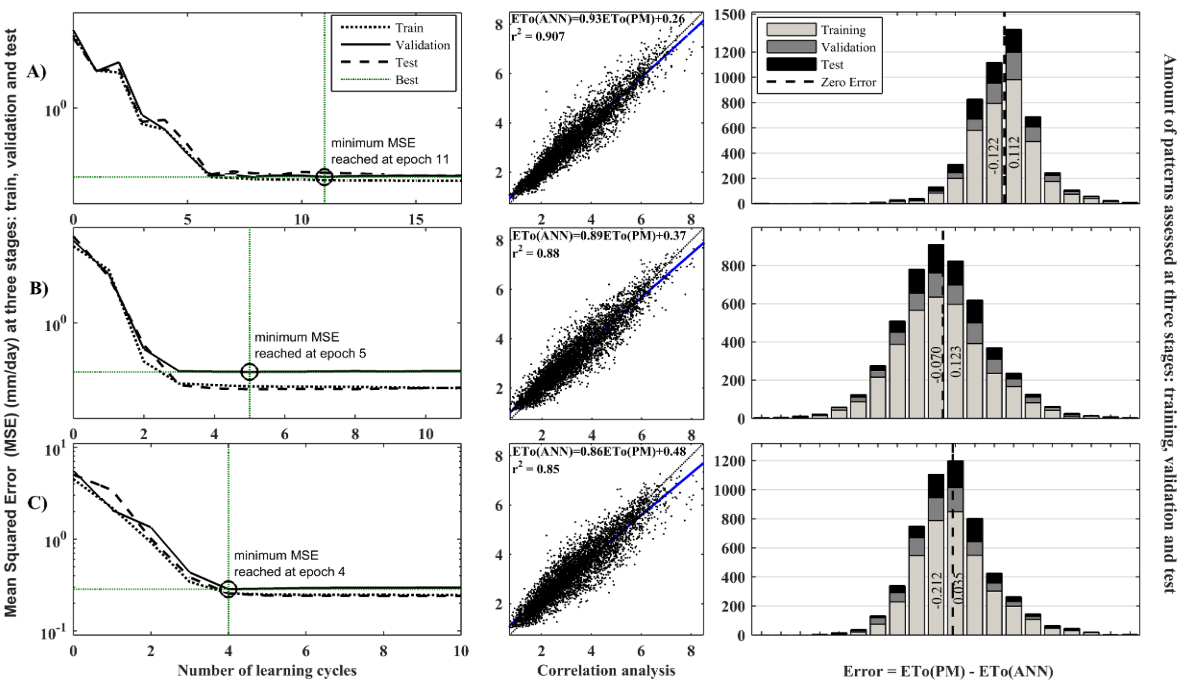

Figure 5. General overview on the performance of networks with missing inputs. Effect of removing some input parameters in the performance of (A) $4U_2$_15_1; (B) 4HRm_30_1; (C) 3_30_1 topologies in terms of *MSE*, correlation analyses between *ETo*(*PM*) and *ETo*(*ANN*) and Error, which accounts for the difference between the target *ETo*(*PM*) values and the predicted values *ETo*(*ANN*) values

The importance of these resuts and its correct reading demonstrated that *ANNs* approach can be an efficient tool to estimate *ETo* which can also be applied for water resources management, irrigation schedule or crop irrigation systems design. Since this study has *ETo* as the leading role of the story and it is completely related to water, we thought that the reading of the results reached and showed above could be more clearly expressed and interpreted in terms of difference in water column height. With it, we meant to demonstrate that the amount of evapotranspirated water predictted by *ANNs* method was outstandingly similar to those estimated by *PM* method. Those differences can be checked in Table 3.

Table 3. Differences between target (*ETo*(*PM*)) and predicted values (*ETo*(*ANN*))

Architectures assessed	Mean daily target height (mm/day)	Mean daily predicted height (mm/day)	Mean daily difference (mm/day)	Total difference (mm)
5_10_1	3.262	3.274	0.012	70.128
5_15_1	3.262	3.260	0.002	11.688
5_20_1	3.262	3.268	0.004	35.064
5_30_1	3.262	3.244	0.018	105.192
4U$_2$_10_1	3.262	3.272	0.010	58.434
4U$_2$_15_1	3.262	3.283	0.021	122.724
4U$_2$_20_1	3.262	3.310	0.048	280.512
4U$_2$_30_1	3.262	3.238	0.024	140.256
4RHm_10_1	3.262	3.250	0.012	70.128
4RHm_15_1	3.262	3.254	0.008	46.752
4RHm_20_1	3.262	3.250	0.012	70.128
4RHm_30_1	3.262	3.261	0.001	5.844
3_10_1	3.262	3.260	0.002	11.688
3_15_1	3.262	3.251	0.011	64.284
3_20_1	3.262	3.271	0.009	52.596
3_30_1	3.262	3.270	0.008	46.752

Table 3. Mean daily target height: mean daily milimeters of water estimated by *PM* method for the whole study period. Mean daily predicted height: mean daily milimeters of water estimated by *ANNs* for the whole study period. Mean daily difference: difference in millimeters of water between Mean daily target height and Mean daily predicted height. Total difference: accumulated difference in millimeters of water for the whole study period.

As we can check in Table 3, Mean daily predicted height barely varies ± 0.0015 mm/day from *ETo*(*PM*) values*,* which is a negligible difference. Even the total difference is low, with a mean accumulated difference of 74.51 mm. for the whole period under study, which emcompassed 16 years; in other words, an annual mean difference of 4.657 mm. Now it is clearer to perceive the high accuracy of the method proposed throughout this paper.

4. Conclusion

The scope of this study was to assess how accurate Artificial Neural Networks could be at estimating reference crop evapotranspiration (*ETo*). To accomplish this task, different *ANNs* were built as it was previously explained and fed with the same number of inputs as those required by Penman-Monteith method, which is considered as the standard method to estimate *ETo*. After training, validating and testing the *ANNs*, the results delivered by them were compared with those estimated by *PM* method and the diferrent statistical indices pointed out *ANN* approach as a high accurate tool to estimate *ETo*.

The neural networks performing the most accurate results were composed by a single hidden layer with no more than 20 neurons in it, fitted with a hyperbolic tangent sigmoid transfer function and guided by the Levenberg-Marquardt algorithm. This architecture was sufficient to reach outstanding results, with a resemblance above 93%, considered as an excellent performance according to the classification presented by Camargo and Sentelhas (1997) and an average MSE of 0.13 mm/day. These results confirmed our main hypothesis: *ANNs* were able to estimate *ETo* values with a high accuracy. Previous studies on the field reached the same conclusion (Chauhan & Shrivastava, 2012; Jain et al., 2008; Khoob, 2008; Kumar et al., 2002; Landeras, Ortiz-Barredo, & López, 2008; Zanetti et al., 2007).

The issue that motivated us to conduct this study was the recurrent problem among researchers of facing incomplete date registers. We were not an exception. Newly, *ANNs* were built presenting some parameters missing in the input layer. The results obtained were also outstanding with correlation coefficients above 0.92, even when two parameters were missing. It confirmed the unpredicted capability of an *ANN* to reproduce at a high resemblance to the target model, in our case, the *PM* model, even with incomplete datasets. Translating these results into water column height, the average difference between predicted *ETo* and target values was ± 0.0015 mm/day and 4.657 mm/year, negligible differences in both cases.

Despite the excellent results obtained throughout this research some other questions arose, such as:

- The possibility of training, validating and testing different *ANNs* with a particular dataset and then estimating new patterns from a different dataset, in those cases where the weather station has been out of order for long periods and its dataset presents great blanks.
- Testing to what extent climatic conditions could limit *ANNs* application. Different locations will probably point out different climatic parameters as those more influent on *ETo*, due to the difference in altitude, radiation, wind speed or humidity conditions. In those cases, Local Sensitivity Analysis would be a useful approach in order to confirm such influence.
- As *ANNs* approach demonstrated to be an efficient tool to estimate *ETo* accurately, further analyses may be focused on future forecasting as irrigation schedule planning and management tool for those public institutions in charge of dealing with water demand, licencing, infrastructure or services.
- The chance of validating *ANNs* approach by using the forecast *ETo* values into other methodologies which are based on a water balance and allow us to model other physical processes where *ETo* plays a significant role.

Thus, deeper analyses and different approaches may be carried out in further studies.

Acknowledgements

The authors would like to acknowledge the support provided by the Fundação de Amparo à Pesquisa de Minas Gerais (FAPEMIG), the Brazilian National Council for Scientific and Technological Development (CNPq) and the CAPES Foundation. We also would like to express our thanks to the Programa de Pós-Graduação em Ecologia (PGECOL) at the Universidade Federal de Juiz de Fora (UFJF), to the chief of Engineering School, PhD. Hélio Antônio da Silva and specially to PhD. Flávio Barbosa to support finantially this publication.

References

Allen, R. G. (1986). A Penman for all seasons. *Journal of Irrigation and Drainage Engineering, 112*(4), 348-368. https://doi.org/10.1061/(ASCE)0733-9437(1986)112:4(348)

Allen, R. G., & FAO. (1998). *Crop evapotranspiration: Guidelines for computing crop water requirements*. Rome: Food and Agriculture Organization of the United Nations.

Allen, R. G., Jensen, M. E., Wright, J. L., & Burman, R. D. (1989). Operational estimates of reference evapotranspiration. *Agronomy Journal, 81*(4), 650-662. https://doi.org/10.2134/agronj1989.00021962008100040019x

Allen, R. G., Smith, M., Pereira, L. S., & Perrier, A. (1994). An update for the calculation of reference evapotranspiration. *ICID Bulletin, 43*(2), 35-92.

Arca, B., Beniscasa, F., & Vincenzi, M. (2001). Evaluation of neural network techniques for estimating evapotranspiration. *Evolving Solution with Neural Networks* (pp. 62-97). Italia: Baratti, R. y De Canete.

Beale, M., Hagan, M., & Demuth, H. (2010). *Neural network toolbox 7*. User's Guide, MathWorks.

Bhattacharya, B., Lobbrecht, A., & Solomatine, D. (2003). Neural Networks and Reinforcement Learning in Control of Water System. *Journal of Water Resources Planning & Management, 129*(6), 458-465. https://doi.org/10.1061/(ASCE)0733-9496(2003)129:6(458)

Blaney, H. F., & Criddle, W. D. (1950). *Determining water requirements in irrigated areas from climatological and irrigation data*. US Soil Conservation Service.

Camargo, A. D., & Sentelhas, P. C. (1997). Avaliação do desempenho de diferentes métodos de estimativa da evapotranspiração potencial no Estado de São Paulo, Brasil. *Revista Brasileira de Agrometeorologia, 5*(1), 89-97.

Campolo, M., Soldati, A., & Andreussi, P. (2003). Artificial neural network approach to flood forecasting in the River Arno. *Hydrological Sciences Journal, 48*(3), 381-398. https://doi.org/10.1623/hysj.48.3.381.45286

Chauhan, S., & Shrivastava, R. (2009). Performance evaluation of reference evapotranspiration estimation using climate based methods and artificial neural networks. *Water Resources Management, 23*(5), 825-837. https://doi.org/10.1007/s11269-008-9301-5

Chauhan, S., & Shrivastava, R. (2012). Estimating reference evapo-transpiration using neural computing technique. *Journal of Indian Water resources Society, 32*, 22-30.

Chiew, F., Kamaladasa, N., Malano, H., & McMahon, T. (1995). Penman-Monteith, FAO-24 reference crop evapotranspiration and class-A pan data in Australia. *Agricultural Water Management, 28*(1), 9-21. https://doi.org/10.1016/0378-3774(95)01172-F

Chow, V., Maidment, D., & Mays, L. (1988). *Applied hydrology*. McGraw-Hill Higher Education.

Coulibaly, P., Anctil, F., & Bobee, B. (2000). Daily reservoir inflow forecasting using artificial neural networks with stopped training approach. *Journal of Hydrology, 230*(3), 244-257. https://doi.org/10.1016/S0022-1694(00)00214-6

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems, 2*(4), 303-314. https://doi.org/10.1007/BF02551274

Dai, X., Shi, H., Li, Y., Ouyang, Z., & Huo, Z. (2009). Artificial neural network models for estimating regional reference evapotranspiration based on climate factors. *Hydrological Processes, 23*(3), 442. https://doi.org/10.1002/hyp.7153

Demuth, H., Beale, M., & Hagan, M. (1992). *Neural network toolbox™ 6*. User Guide, Copyright, 2008.

Fausset, L. (1994). Fundamentals of neural networks. *Architecture, Algorithm and Application*. Prentice Hall.

Galvão, R. K., Becerra, V. M., Calado, J. M., & Silva, P. M. (2004). Linear-wavelet networks. *International Journal of Applied Mathematics and Computer Science, 14*(2), 221-232.

Geiger, R. (1961). *Überarbeitete Neuausgabe von Geiger, R. Köppen-Geiger/Klima der Erde* (Wandkarte 1: 16 Mill.). Darmstadt.

Govindaraju, R. S., & Rao, A. R. (2000). *Artificial neural networks in hydrology: ii, hydrologic applications*.

Hagan, M. T., & Menhaj, M. B. (1994). Training feedforward networks with the Marquardt algorithm. *IEEE Transactions on Neural Networks, 5*(6), 989-993. https://doi.org/10.1109/72.329697

Hargreaves, G. H., & Samani, Z. A. (1985). Reference crop evapotranspiration from temperature. *Applied Engineering in Agriculture, 1*(2), 96-99. https://doi.org/10.13031/2013.26773

Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation*. Macmillan College Publishing Company, New York.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks, 2*(5), 359-366. https://doi.org/10.1016/0893-6080(89)90020-8

Howell, T., Evett, S., Schneider, A., Dusek, D., & Copeland, K. (2000). *Irrigated fescue grass ET compared with calculated reference grass ET.* Paper presented at the National irrigation symposium. Proceedings of the 4th Decennial Symposium, Phoenix, Arizona, USA, November 14-16, 2000.

Imrie, C., Durucan, S., & Korre, A. (2000). River flow prediction using artificial neural networks: Generalisation beyond the calibration range. *Journal of Hydrology, 233*(1), 138-153. https://doi.org/10.1016/S0022-1694(00)00228-6

Irmak, S., Irmak, A., Allen, R., & Jones, J. (2003). Solar and net radiation-based equations to estimate reference evapotranspiration in humid climates. *Journal of Irrigation and Drainage Engineering, 129*(5), 336-347. https://doi.org/10.1061/(ASCE)0733-9437(2003)129:5(336)

Itenfisu, D., Elliott, R. L., Allen, R. G., & Walter, I. A. (2003). Comparison of reference evapotranspiration calculations as part of the ASCE standardization effort. *Journal of Irrigation and Drainage Engineering, 129*(6), 440-448. https://doi.org/10.1061/(ASCE)0733-9437(2003)129:6(440)

Izadifar, Z., & Elshorbagy, A. (2010). Prediction of hourly actual evapotranspiration using neural networks, genetic programming, and statistical models. *Hydrological Processes, 24*(23), 3413-3425. https://doi.org/10.1002/hyp.7771

Jain, S., Das, A., & Srivastava, D. (1999). Application of ANN for reservoir inflow prediction and operation. *Journal of Water Resources Planning and Management, 125*(5), 263-271. https://doi.org/10.1061/(ASCE)0733-9496(1999)125:5(263)

Jain, S., Nayak, P., & Sudheer, K. (2008). Models for estimating evapotranspiration using artificial neural networks, and their physical interpretation. *Hydrological Processes, 22*(13), 2225-2234. https://doi.org/10.1002/hyp.6819

Jensen, M. E., & Haise, H. R. (1963). Estimating evapotranspiration from solar radiation. *Proceedings of the American Society of Civil Engineers, Journal of the Irrigation and Drainage Division, 89*, 15-41.

Jothiprakash, V., RamaChandran, M., & Shanmuganathan, P. (2002). Artificial neural network model for estimation of REF-ET. *Journal of the Institution of Engineers(India), Part CV, Civil Engineering Division, 83*(1), 17-20.

Kashyap, P., & Panda, R. (2001). Evaluation of evapotranspiration estimation methods and development of crop-coefficients for potato crop in a sub-humid region. *Agricultural Water Management, 50*(1), 9-25. https://doi.org/10.1016/S0378-3774(01)00102-0

Khoob, A. R. (2008). Artificial neural network estimation of reference evapotranspiration from pan evaporation in a semi-arid environment. *Irrigation Science, 27*(1), 35-39. https://doi.org/10.1007/s00271-008-0119-y

Kisi, O. (2006a). Daily pan evaporation modelling using a neuro-fuzzy computing technique. *Journal of Hydrology, 329*(3), 636-646. https://doi.org/10.1016/j.jhydrol.2006.03.015

Kisi, O. (2006b). Evapotranspiration estimation using feed-forward neural networks. *Hydrology Research, 37*(3), 247-260. https://doi.org/10.2166/nh.2006.010

Kisi, O. (2006c). Generalized regression neural networks for evapotranspiration modelling. *Hydrological Sciences Journal, 51*(6), 1092-1105. https://doi.org/10.1623/hysj.51.6.1092

Kisi, O. (2007). Evapotranspiration modelling from climatic data using a neural computing technique. *Hydrological Processes, 21*(14), 1925-1934. https://doi.org/10.1002/hyp.6403

Kisi, O. (2008). The potential of different ANN techniques in evapotranspiration modelling. *Hydrological Processes, 22*(14), 2449-2460. https://doi.org/10.1002/hyp.6837

Kisi, O., & Öztürk, O. (2007). Adaptive neurofuzzy computing technique for evapotranspiration estimation. *Journal of Irrigation and Drainage Engineering, 133*(4), 368-379. https://doi.org/10.1061/(ASCE)0733-9437(2007)133:4(368)

Köppen, W. P. (1884). *Die Wärmezonen der Erde, nach der Dauer der heissen, gemässigten und kalten Zeit und nach der Wirkung der Wärme auf die organische Welt betrachtet*. Publisher not identified.

Köppen, W. P. (1918). *Klassifikation der Klimate nach Temperatur, Niederschlag und jahreslauf*. Perthes.

Köppen, W. P. (1936). *Das geographische system der klimate*.

Köppen, W. P. (2011). The thermal zones of the Earth according to the duration of hot, moderate and cold periods and to the impact of heat on the organic world. *Meteorologische Zeitschrift, 20*(3), 351-360. https://doi.org/10.1127/0941-2948/2011/105

Kumar, M., Raghuwanshi, N., Singh, R., Wallender, W., & Pruitt, W. (2002). Estimating evapotranspiration using artificial neural network. *Journal of Irrigation and Drainage Engineering, 128*(4), 224-233. https://doi.org/10.1061/(ASCE)0733-9437(2002)128:4(224)

Landeras, G., Ortiz-Barredo, A., & López, J. J. (2008). Comparison of artificial neural network models and empirical and semi-empirical equations for daily reference evapotranspiration estimation in the Basque Country (Northern Spain). *Agricultural Water Management, 95*(5), 553-565. https://doi.org/10.1016/j.agwat.2007.12.011

Lekkas, D., Imrie, C., & Lees, M. (2001). Improved non-linear transfer function and neural network methods of flow routing for real-time forecasting. *Journal of Hydroinformatics, 3*(3), 153-164.

Lin, G.-F., & Chen, L.-H. (2004). A non-linear rainfall-runoff model using radial basis function network. *Journal of Hydrology, 289*(1), 1-8. https://doi.org/10.1016/j.jhydrol.2003.10.015

Lingireddy, S. (1998). Aquifer parameter estimation using genetic algorithms and neural networks. *Civil Engineering Systems, 15*(2), 125-144. https://doi.org/10.1080/02630259808970234

McNaughton, K., & Jarvis, P. (1984). Using the Penman-Monteith equation predictively. *Agricultural Water Management, 8*(1-3), 263-278. https://doi.org/10.1016/0378-3774(84)90057-X

Monteith, J. (1965). Evaporation and environment. The state and movement of water in living organisms. *Symposium of the society of experimental biology* (Vol. 19, pp. 205-234). Cambridge: Cambridge University Press.

Penman, H. L. (1948). *Natural evaporation from open water, bare soil and grass*. Paper presented at the Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences. https://doi.org/10.1098/rspa.1948.0037

Shukla, M., Kok, R., Prasher, S., Clark, G., & Lacroix, R. (1996). Use of artificial neural networks in transient drainage design. *Transactions of the ASAE, 39*(1), 119-124. https://doi.org/10.13031/2013.27488

Soares, D. S., & Nadal, J. (1999). *Aplicação de uma Rede Neural Feedforward com Algoritmo de Levenberg-Marquardt para Classificação de Alterações do Segmento ST do Eletrocardiograma*.

Souza, D., & Yoder, R. (1994). *Et estimation in the northeast of Brazil: Hargreaves or Penman-Monteith equation?* Paper.

Sudheer, K., Gosain, A., & Ramasastri, K. (2005). Closure to "Estimating Actual Evapotranspiration from Limited Climatic Data Using Neural Computing Technique" by KP Sudheer, AK Gosain, and KS Ramasastri. *Journal of Irrigation and Drainage Engineering, 131*(2), 220-220. https://doi.org/10.1061/(ASCE)0733-9437(2005)131:2(220)

Sudheer, K., & Jain, A. (2004). Explaining the internal behaviour of artificial neural network river flow models. *Hydrological Processes, 18*(4), 833-844. https://doi.org/10.1002/hyp.5517

Tahir, S. (1998). *Estimating potential evaporation using artificial neural network*. Jakarta: INACID.

Thornthwaite, C. W. (1948). An approach toward a rational classification of climate. *Geographical Review, 38*(1), 55-94. https://doi.org/10.2307/210739

Trajkovic, S., Todorovic, B., & Stankovic, M. (2003). Forecasting of reference evapotranspiration by artificial neural networks. *Journal of Irrigation and Drainage Engineering, 129*(6), 454-457. https://doi.org/10.1061/(ASCE)0733-9437(2003)129:6(454)

Traore, S., Wang, Y.-M., & Kerh, T. (2010). Artificial neural network for modeling reference evapotranspiration complex process in Sudano-Sahelian zone. *Agricultural water management, 97*(5), 707-714. https://doi.org/10.1016/j.agwat.2010.01.002

Ventura, F., Spano, D., Duce, P., & Snyder, R. (1999). An evaluation of common evapotranspiration equations. *Irrigation Science, 18*(4), 163-170. https://doi.org/10.1007/s002710050058

Wen, C. G., & Lee, C. S. (1998). A neural network approach to multiobjective optimization for water quality management in a river basin. *Water Resources Research, 34*(3), 427-436. https://doi.org/10.1029/97WR02943

Wilby, R., Abrahart, R., & Dawson, C. (2003). Detection of conceptual model rainfall—Runoff processes inside an artificial neural network. *Hydrological Sciences Journal, 48*(2), 163-181. https://doi.org/10.1623/hysj.48.2.163.44699

Wright, J. L., Allen, R. G., & Howell, T. A. (2000). *Conversion between evapotranspiration references and methods*.

Yang, C.-C., Lacroix, R., & Prasher, S. (1998). The use of back-propagation neural networks for the simulation and analyses of time-series data in subsurface drainage systems. *Transactions of the ASAE, 41*(4), 1181. https://doi.org/10.13031/2013.17243

Yang, C.-C., Prasher, S., & Lacroix, R. (1996). Applications of artificial neural networks to land drainage engineering. *Transactions of the ASAE, 39*(2), 525-533. https://doi.org/10.13031/2013.27531

Zanetti, S., Sousa, E., Oliveira, V., Almeida, F., & Bernardo, S. (2007). Estimating evapotranspiration using artificial neural network and minimum climatological data. *Journal of Irrigation and Drainage Engineering, 133*(2), 83-89. https://doi.org/10.1061/(ASCE)0733-9437(2007)133:2(83)

Zealand, C. M., Burn, D. H., & Simonovic, S. P. (1999). Short term streamflow forecasting using artificial neural networks. *Journal of Hydrology, 214*(1), 32-48. https://doi.org/10.1016/S0022-1694(98)00242-X

Appendix

Appendix A. Code on Matlab R2015a language to build, train, validate and test an ANN with 10 nodes in the hidden layer.

```
Target = [ETo]; % our target values were those estimated by PM method
Input = [TmaxK TminK Insolation MeanRelativeHumidity WindSpeed]; % Here, we introduced our 5 input parameters
L = length(Target);
A = randperm(L); % It creates a vector with permuted values within 1 and L
portion_training = 0.70; % Portion of patterns used at training phase
portion_validation = 0.15; % Portion of patterns used at validating phase
portion_testing = 1 - porc_treino - porc_valida; % Portion of patterns used at testing phase.
Ntraining = round(portion_training*L);
Nvalidation = round(portion_validation*L);
Ntesting = L – Ntraining - Nvalidation;
% CREATING A NETWORK
hidden_neurons = 10; %(default = '10')
net = patternnet(hidden_neurons);
% CONFIGURING THE TRAINING PHASE
net.trainFcn = 'trainlm'; % Training guided by the Levenberg-Marquardt algorithm
net.performFcn = 'mse'; % Measuring network performance by Mean Squared Error
net.trainParam.epochs = 1000; % Limiting the maximum number of epochs to 1000
net.divideFcn='divideind'; % Dividing the dataset by index
net.divideParam.trainInd = 1:Ntreino;
net.divideParam.valInd = (Ntreino+1):L-Ntesting;
net.divideParam.testInd = L-Ntesting+1:L;
% RUNNING THE NETWORK
network = train(net,Input',Target');
Target_predicted = network(Input');
% PLOTTING
plot(Target) % Plotting the target values (ETo(PM))
hold on % it is to plot together into the same figure and compare both results
plot(Target_predicted, 'r') % Plotting the predicted values (ETo(ANN))
figure, plotregression(Target,Target_predicted) % Plotting the regression function between (ETo(PM)) and (ETo(ANN)).
```

Copyrights