*Original research / Artículo original - Tipo 1*

# Estimating missing data in historic series of global radiation through neural network algorithms

**Franklin García Acevedo** / franklinmeerga@ufps.edu.co

**Juan Rojas Serrano** / juanandresrs@ufps.edu.co

**Alejandro Vásquez Vega** / darioalejandrovv@ufps.edu.co

**Diego Parra Peñaranda** / diegoalejandropp@ufps.edu.co

**Erney Castro Becerra** / erneyfabiancb@ufps.edu.co

Universidad Francisco de Paula Santander, Cúcuta-Colombia

**ABSTRACT** Managing meteorological data is usual to find incomplete data of time series in some intervals; the issue is addresses commonly using the autoregressive integrated moving average (ARIMA) or the method by regression analysis (interpolation), both with certain limitations under particular conditions. This paper presents the results of an investigation aimed at solving the problem using neural networks reported. The analysis of a time series of global radiation obtained at the Francisco de Paula Santander University (UFPS) is presented, with basis in the recorded data by the weather station attached to the Department of Fluids and Thermals. Having a series of ten-year study for 125,658 records of temperature, radiation and energy with a percentage of 9.98 missing data, which were duly cleared and completed by a neural network using algorithms backpropagation in the mathematical software MATLAB.

**KEYWORDS** Neural network; weights; bias; editing; imputation.

## Estimación de datos faltantes en series históricas de radiación global mediante algoritmos de redes neuronales

RESUMEN En el tratamiento de datos de series de tiempo meteorológicas se encuentran problemas de datos incompletos en algunos intervalos de tiempo; el problema se aborda comúnmente utilizando el modelo auto-regresivo de media móvil (ARIMA) o el método por análisis de regresión (interpolación), ambos con ciertas limitaciones en condiciones particulares. En este documento se reportan los resultados de una investigación dirigida a resolver el problema utilizando redes neuronales. Se presenta el análisis efectuado a una serie histórica de radiación global obtenida en la Universidad Francisco de Paula Santander (Cúcuta, Colombia), con base en los datos registrados por su estación meteorológica, a partir de una serie de estudio de diez años (125.658 registros de temperatura, radiación y energía), con 9.98% datos faltantes. Los datos fueron debidamente depurados y completados mediante algoritmos de redes neuronales tipo backpropagation usando el software matemático MATLAB.

PALABRAS CLAVE Red neuronal; pesos; bias; depuración; imputación.

## Estimativa de dados faltantes em séries temporais de radiação global através de algotitmos de redes neurais

RESUMO No tratamento de dados de séries de tempo meteorológicas encontram-se problemas de dados incompletos em alguns intervalos de tempo; o problema é abordado geralmente usando o modelo auto-regressivo integrado de média móvel (ARIMA) ou o método de análise de regressão (interpolação), ambos com certas limitações em condições particulares. Este artigo apresenta os resultados de uma investigação que visa resolver o problema utilizando redes neurais. Apresenta-se a análise realizado a uma série histórica de radiação global obtida na Universidade Francisco de Paula Santander (Cucuta, Colômbia), com base nos dados registrados por sua estação meteorológica, a partir de uma série de estudo de dez anos (125,658 registros de temperatura, radiação e energia), com 9,98% de dados em falta. Os dados foram devidamente depurados e completados mediante algoritmos de rede neurais tipo backpropagation usando o software matemático MATLAB.

PALAVRAS-CHAVE Rede neuronal; pesos; bias; depuração; imputação.

*Received / Recepción: Junio 13, 2016 - Accepted / Aceptación: Junio 26, 2016* *doi. 10.18046/syt.v14i37.2239*

# I. Introduction

The Department of Fluids and Thermics of the University Francisco de Paula Santander (UFPS) in 1998 took the initiative to acquire a weather station in order to support research in areas related to alternative energy power generation and meteorological data analysis for regional agronomy. From that date, they recorded variables influenced by the weather, with recording problems over some time periods due to factors such as: lack of energy, datalogger failures, among others. In order to solve this problem, the Development Research Group of Industrial Processes (GIDPI) moved forward the current research, whose main objective is to complete the missing data of the variables that were lost in time.

Meteorological time series of data processing commonly merge with the problem of missing data in some time intervals; there is a large variety of alternative analyses to complete the missing data, among which stand out two common methods: the autoregressive integrated moving average model (ARIMA) and the regression analysis method (i.e. interpolation).

The autoregressive integrated moving average model is based on a dynamic temporary series that makes future estimates by observing past events in the series. This method is commonly used in analysis of phenomena that do not vary too much with time, e.g. temperature, humidity, pressure, wind speed, and other phenomena in which the tolerance of change in short periods of time is low (Vásquez, Rojas, & Duarte, 2015).

The application of the regression analysis method, for its part, requires selecting a data series with a similar behavior, within the same topoclimatic influence area (reference series), which must have records in the same time interval that are missing in the study series; in this way, mathematical regression between the study series and the regency series is performed (Medina, 2008).

Solar radiation is a natural phenomenon that has a cycle of action between 10–12 hours, generally between 6:00 AM and 6:00 PM, having a maximum value within half an hour of this interval. Because of these characteristics, the application of the ARIMA method for predicting the missing data is prevented. On the other hand, the interpolation method is shown to be feasible because the nearest weather station attached

## I. Introducción

El Departamento de Fluidos y Térmicas de la Universidad Francisco de Paula Santander (UFPS), en 1998, tomó la iniciativa de adquirir una estación meteoróloga con el fin apoyar la investigación en áreas afines con las energías alternativas de generación eléctrica y análisis de datos meteorológicos para la agronomía de la región, desde esta fecha se realiza el registro de las variables influenciadas por el clima, con inconvenientes de registro en algunos lapsos de tiempo, debido a factores como: falta de energía, fallas en el datalogger, entre otros. Con el fin de solucionar esta problemática, el Grupo de Investigación en Desarrollo de Procesos Industriales (GIDPI) adelantó la presente investigación, cuyo objetivo principal es completar los datos faltantes de las variables extraviadas en el tiempo.

Comúnmente el tratamiento de datos de series de tiempo meteorológicas se encuentra con el problema de datos incompletos en algún intervalo de tiempo; existe gran variedad de alternativas de análisis para completar los datos ausentes, entre las cuales se destacan dos métodos muy comunes: el modelo auto-regresivo de media móvil (ARIMA) y el método por análisis de regresión (interpolación).

El modelo auto-regresivo de media móvil se basa en una dinámica de series temporales que realiza estimaciones futuras observando los eventos pasados en la serie, este método es comúnmente usado en análisis de fenómenos poco variantes en el tiempo, caso particular para: temperatura, humedad, presión, velocidad del viento, y demás fenómenos en donde la tolerancia de cambio en periodos cortos de tiempo es baja (Vásquez, Rojas, & Duarte, 2015).

La aplicación del método por análisis de regresión, por su parte, requiere seleccionar una serie de datos con un comportamiento similar, dentro de la misma área de influencia topoclimática (serie de referencia), la cual debe poseer los registros que en el mismo intervalo de tiempo están ausentes en la serie de estudio, de esta manera se realiza una regresión matemática entre la serie de estudio y la serie de regencia (Medina, 2008).

La Radiación solar es un fenómeno natural que posee un ciclo de acción de entre 10 y 12 horas, por lo general entre las 6:00 AM y las 6:00 PM, el cual tiene un valor máximo en la hora media de este intervalo. Debido a estas características se impide la aplicación del método ARIMA para la predicción de los valores extraviados, por otra parte el método de interpolación se muestra viable debido a que la estación meteorológica más cercana adscrita al Instituto de Hidrología, Meteorología y Estudios Ambientales [IDEAM], se encuentra dentro del rango de influencia a 5 km de distancia, pero esta misma ofrece los valores de radiación global en intervalos diarios, por tal motivo no se consideró pues se presentaría mucho rango de incerteza en los resultados.

A diferencia de estos dos métodos que trabajan con ecuaciones estadísticas, las redes neuronales [RNA] se consideran como un modelado de datos de gran alcance, ya que

pueden capturar y representar relaciones implícitas complejas con variables de entrada/salida [E/S]. Las ventajas de las RNA es que tienen la capacidad de representar, tanto relaciones lineales, como no lineales, y tienen la habilidad de aprender relaciones directas entre dichas variables de E/S (Hamzaoui et al., 2011). Estas valiosas ventajas nos permiten proponer un método para completar datos faltantes, valiéndonos de las variables de E/S conocidas, fundamentado en las RNA.

Debe quedar claro que cualquier método utilizado es incapaz de reproducir los datos perdidos, pero el mejor método a implementar permite obtener valores razonables que son consistentes con la naturaleza de cualquier fenómeno variante en el tiempo (Serlin, 2010).

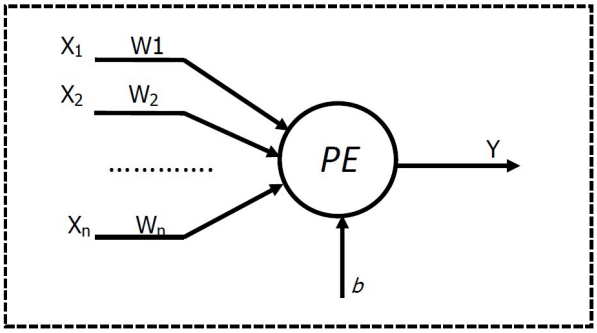


*Figure 1. Neural network architecture / Arquitectura de neurona artificial*

## II. Selección de la red neuronal

Existe variedad de criterios y tipos de algoritmos en la implementación de las RNA, pero su funcionamiento es similar, consta de múltiples neuronas artificiales interconectadas las cuales se comportan como Procesadores Elementales [PE] cuya función es relacionar los estímulos externos de un sistema con la respuesta del mismo –**Figura 1**– (Hamzaoui et al., 2011).

Con fin de modelar el comportamiento de un PE se halla la relación existente E/S, la cual se denomina como función de transferencia, donde la salida (*Y*) se obtiene a partir de la adición del estado inicial denotado por el coeficiente bias (*b*), el cual define el estado en el que se encuentra toda neurona, esto es: en reposo o en excitación (Ramírez, 2010), con la multiplicación entre los estímulos externos (*X*), y los coeficientes de interconexión entre los PE, conocidos como pesos (*W*) (1) (Ponce, 2010).

$$Y=X_1 W_1+X_2 W_2+X_n W_n+b \quad (1)$$

Las redes neuronales pueden clasificarse de diversas formas de acuerdo con una o más de sus características relevantes, entre las cuales se encuentran:

- la función a la cual están diseñados los PE (e.g., asociación de patrones de E/S);

to the Institute of Hydrology, Meteorology and Environmental Studies [IDEAM] is within the range of influence at 5 km away. Global radiation values are taken in daily intervals; for that reason it was not considered, since a high uncertainty range would be presented in the results.

Unlike these two methods that work with statistical equations, Neural Networks [NN] are considered as far-reaching data modeling, since they can capture and represent complex implicit relationships with input/output variables [I/O]. The advantages of NN are that they have the ability to represent both linear and nonlinear relationships, and also can learn a direct relationship between the variables of [I/O] (Hamzaoui et al., 2011). These valuable advantages allow us to propose a method to complete missing data, making use of the I/O known variables, based on the NN.

It should be clear that any method used is unable to reproduce the missing data, but the best method to be implemented allows us to obtain reasonable values that are consistent with the nature of any varying phenomenon in time (Serlin, 2010).

## II. Selection of the neural network

There is a variety of criteria and types of algorithms in the implementation of the NN, but its operation is similar, consisting of multiple interconnected artificial neurons that behave as Elemental Processors [EP] whose function is to relate the external stimuli of a system with the response of the same –**Figure 1**– (Hamzaoui et al., 2011).

In order to model the behavior of a EP, the existing relationship I/O has to be determined, which is referred to as the transference function, where the output (*Y*) is obtained from the addition of the initial state denoted by the coefficient bias (*b*), which defines the state of every neuron, i.e. inhibition or excitation (Ramírez, 2010), with the multiplication between external stimuli (*X*), and the interconnection coefficients between EP, known as weights (*W*) (1) (Ponce, 2010).

$$Y=X_1 W_1+X_2 W_2+X_n W_n+b \quad (1)$$

Neural networks can be classified in several ways according to one or more of their relevant characteristics, such as:

- the function for which EP are designed (e.g., pattern matching of I/O);
- degree of connectivity between EP, which can be partial or complete;
- direction of flow of information between the NN connections;
- type of learning algorithm, which represents the entirety of the existing relationship between the I/O under an arbitrary training in EP;
- learning rules, which can define relevant parameters such as time alliterations and learning coefficients; and
- degree of supervision of required learning for network formation (i.e. the percentage of data used for training and testing; Basheer & Hajmeer, 2000).

Neurons are grouped into different layers and interconnected according to their architecture. The network often has one or more hidden layers between the input layer and the output layer, which allow it to learn about nonlinear and linear relationships **Figura 2** (Infante, Ortega, & Cedeño, 2008).

The optimal number of neurons in the hidden layer is difficult to calculate, and depends on the type and dynamics of the phenomenon to work and its complexity. This number is generally determined in an iterative way, observing the uncertainty range between the training data and validation results in the learning phase of NN (Infante et al., 2008; Hernández, Bassam, Siqueiros, & Juarez, 2009).

Based on Hamzaoui et al., (2010), Basheer and Hajmeer (2000), and Colmenares (n.d), who recommend the use of backpropagation network type [MLP] for functions with patterns of reconstruction of damaged or completely missing data, the implementation of a type of MLP *Feed-Forward* called *Backpropagation* [BP] was determined. The BP term refers to how error is calculated in the output layer, and then propagated to the hidden layers, and subsequently to the input layer. This gives a major advantage because it provides versatility to the NN at the moment to be trained, allowing it to self-adjust the coefficient W of each connection and the B of each neuron used.

- grado de conectividad entre los PE la cual puede ser parcial o completa;
- dirección de flujo de información entre las conexiones de la RNA;
- tipo de algoritmo de aprendizaje, el cual representa el conjunto de la relación existente entre las E/S bajo un entrenamiento arbitrario en los PE;
- reglas de aprendizaje, las que pueden definir parámetros relevantes como: tiempos de aliteraciones y coeficientes de aprendizaje; y
- grado de supervisión de aprendizaje necesario para la formación de la red (porcentajes de datos utilizados para el entrenamiento y prueba) (Basheer & Hajmeer, 2000).

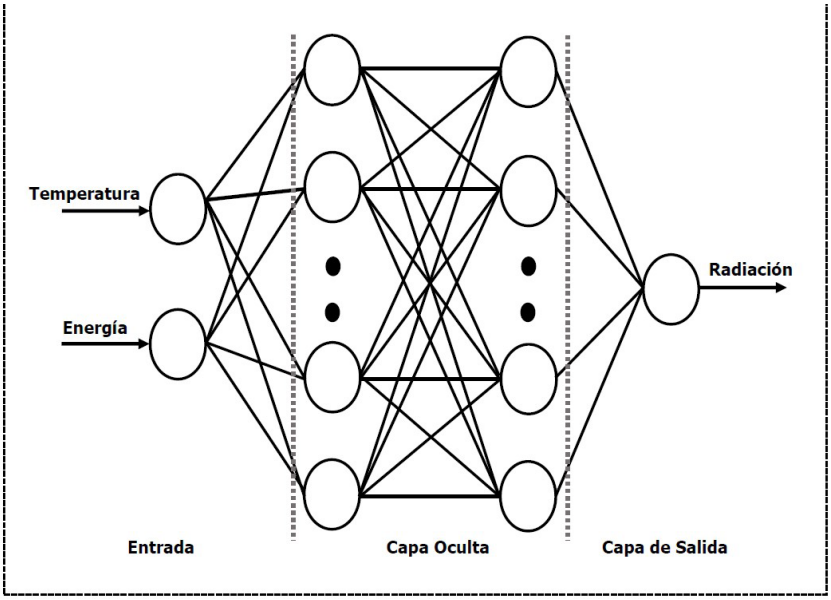


*Figure 2. Multilayer neural network architecture / Arquitectura de una red neuronal multicapa*

Las neuronas se agrupan en capas distintas e interconectadas de acuerdo con su arquitectura, la Red a menudo tiene una o más capas ocultas entre la capa de entrada y la capa de salida, las cuales le permiten a la red aprender relaciones no lineales y lineales –**Figura 2**– (Infante, Ortega, & Cedeño, 2008).

El número óptimo de neuronas en la capa oculta es difícil de calcular, y depende del tipo y la dinámica del fenómeno a trabajar y de la complejidad del mismo, este número se determina generalmente de forma iterativa, observando los rango de incerteza entre los datos de entrenamiento y los resultados de validación en la fase de aprendizaje de la RNA (Hernandez, Bassam, Siqueiros, & Juarez, 2009; Infante et al., 2008).

Con base en Hamzaoui et al., (2010), Basheer y Hajmeer (2000), y Colmenares (n.d), quienes recomiendan el uso del tipo de red de retropropagación [MLP] para funciones con patrones de reconstrucción de datos dañados o completamente faltantes, se determinó la implementación de un tipo de MLP *Feed-Forward* denominado *BackPropagation* [BP]. El

termino BP se refiere a la forma en que el error es calculado en la capa de salida, para luego ser propagados a la capas ocultas, y posteriormente a la capa de entrada, brindándole una gran ventaja, ya que provee a la RNA versatilidad al momento de ser entrenada, permitiéndole autoajustar los coeficiente W de cada conexión y los B de cada neurona utilizada.

Las características de la arquitectura de las RNA BP son:

- está conectada hacia adelante, es decir, su tipo de conexión es unidireccional entre las capas de la primera línea con las capas de la siguiente;
- posee una capa de entrada con nodos que representan los estímulos de entrada en RNA, esta capa actúa solo como punto de distribución por lo que algunos autores no la catalogan como capa;
- cuenta con una capa de salida con nodos que representa la variable dependiente (variable modelada); y
- cuenta con una o más capas ocultas que contienen neuronas para ayudar a capturar la no linealidad, usando una estrategia de aprendizaje supervisado.

Otros referentes a tener en cuenta son Ramírez (2010) y Jamett (2004), quienes recomiendan el uso de la función de entrenamiento *Levenberg Marquardt BP*, conocida como *Trainlm*, la cual ajusta cada variable según el algoritmo Levenberg-Marquardt con la finalidad de obtener mayor rendimiento con una velocidad de entrenamiento mayor. Además, las funciones de activación deben ser continuas, por tal motivo se debe realizar un tipo de restricción sigmoidal, la cual se encarga de normalizar los valores dependientes en intervalos entre 1 y 0.

## III. Data experimental

Las series de temperatura, radiación y energía utilizadas para este estudio son registros tomados de forma horaria a una altura de 15.4 metros y suministrados por la estación meteorológica Groweatherlink de Davis Instruments, ubicada en la UFPS, latitud 7°53'545''N, longitud 72°29'166''W

The characteristics of the architecture of the NN BP are:

- it is connected forward; that is, its connection type is unidirectional between the layers of the first line and the following layers;
- has an input layer with nodes that represent input stimuli in NN This layer acts only as a distribution point; for that reason, some workers do not categorize it as a layer;
- has an output layer with nodes that represent the dependent variable (i.e. the modeled variable); and
- has one or more hidden layers containing neurons that help to capture the nonlinearity using a supervised learning strategy.

Ramírez (2010) and Jamett (2004) recommend the use of the training function *Levenberg Marquardt BP*, known as *Trainlm*, which adjusts each variable according to Levenberg-Marquardt algorithm in order to obtain higher performance with higher training speed. In addition, the activation functions must be continuous; for that reason, a sigmoidal type of restriction must be performed, which is responsible for normalizing the dependent values in intervals between 0 and 1.

## III. Experimental data

The series of temperature, radiation and energy used here are records taken on an hourly basis at a height of 15.4 m and supplied by the weather station from Groweatherlink of Davis Instruments, located in the UFPS, latitude 7°53'545''N, longitude 72°29'166''W

*Table 1. Available radiation records / Registros de radiación*

| | Jan. | Feb. | Mar. | Apr. | May | Jun. | Jul. | Aug. | Sep. | Oct. | Nov. | Dec. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2001 | 0.25 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| 2002 | 1.00 | 0.00 | 0.00 | 0.24 | 1.00 | 1.00 | 1.00 | 0.97 | 1.00 | 1.00 | 1.00 | 1.00 |
| 2003 | 1.00 | 1.00 | 1.00 | 0.88 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0.95 |
| 2004 | 0.39 | 1.00 | 1.00 | 1.00 | 1.00 | 0.72 | 1.00 | 0.96 | 0.39 | 1.00 | 1.00 | 1.00 |
| 2005 | 1.00 | 1.00 | 1.00 | 1.00 | 0.51 | 0.72 | 1.00 | 0.31 | 1.00 | 1.00 | 1.00 | 0.54 |
| 2006 | 1.00 | 0.84 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| 2007 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| 2008 | 1.00 | 1.00 | 0.74 | 1.00 | 1.00 | 1.00 | 0.57 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| 2009 | 1.00 | 1.00 | 1.00 | 0.58 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 0.49 | 0.00 | 0.31 |
| 2010 | 1.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 1.00 | 1.00 | 0.48 | 0.85 | 0.73 | 0.77 |

*value x 100% / *valor x100%*

(Vásquez et al., 2015). Due to the periodicity of the solar radiation, data between 6:00 AM and 6:00 PM are analyzed, obtaining groups of records on a monthly basis. In this way the treatment is more convenient and simple.

**Table 1** shows the percentage of records supplied by the station; 120 months are shown, of which 10 contain less than 30% of data, 21 have between 30–97% and 88 have all the data.

In the treatment of databases with neural networks, it is recommended to distribute the records as follows (Jamett, 2014; San Juan, Jamett, Kaschel, & Sánchez, 2015):

- 70% of data must be for training;
- 20% must be for testing; and
- 10% must be for validation.

This is in order to check the results of the test phase of the NN. As shown in **Table 1**, months that contain all the data constitute 73.3%, which concludes the viability of the series for the validation process and record estimation.

As previously mentioned, the operation of the NN depends on the I/O relationship, where our output is the Global Radiation [R], and the independent entries must be classified according to the analysis of this phenomenon. In our case, the most influential external stimuli in R provided by the weather station are: temperature (T) and energy (E; see **Figure 2**).

## IV. Implementation of the NN

The tool that lets us create any type of network, train it and put it into operation and testing, is the *Neural Network Tool-Graphical User Interface* [*nntool*] from MATLAB, which, as its name implies, provides a graphical interface where the user interacts with the *I/O* variables.

As previously stated, the external stimuli are temperature (*T*) and energy (*E*), and the dependent variable is radiation (*R*). To operate nntool we must prepare these variables in such a way that the system output (*R*) must be defined by a 1 x n matrix, in which n is denoted as the amount of known output records; in our case, for the month of January, 3224, which corresponds to

(Vásquez et al., 2015). Debido a la periodicidad de la radiación solar se analizan los datos comprendidos entre las 6:00 AM y las 6:00 PM, obteniendo grupos de registros de forma mensual. De esta manera se hace más cómodo y sencillo el tratamiento.

En la **Tabla 1** se exhibe el porcentaje de registros suministrados por la estación, se muestran 120 meses de los cuales diez contienes menos del 30% de los datos, 21 poseen entre el 30 y el 97% y 88 poseen la totalidad de los datos.

En el tratamiento de bases de datos con redes neuronales es recomendado distribuir los registros de la siguiente manera (San Juan, Jamett, Kaschel, & Sánchez, 2015; Jamett, 2014):

- el 70% de los datos deben ser para entrenamiento;
- el 20% debe ser para pruebas; y
- el 10% debe ser para validación.

Esto con la finalidad de comprobar los resultados de la fase de prueba de la RNA. Como se observa en la **Tabla 1**, los meses con la totalidad de los datos completos comprende el 73.3% lo cual concluye la viabilidad de la serie para el proceso de validación y estimación de registros.

Como se ha definido, el funcionamiento de la RNA depende de la relación de E/S, en donde nuestra salida a evaluar es la Radiación Global [R], y las entradas independientes se deben clasificar de acuerdo con el análisis de este fenómeno. En nuestro caso, los estímulos externos más influyentes en R que suministraba la estación meteorológica son: Temperatura (T) y energía (E) (ver **Figura 2**).

## IV. Implementación de la RNA

Existe una herramienta que nos permite crear cualquier tipo de red, entrenarla y ponerla en funcionamiento y prueba, la *Neural Network Tool-Graphical User Interface* [*nntool*] de MATLAB, la cual, como su nombre lo indica, proporciona una interfaz gráfica donde el usuario interactúa con las variables de *E/S*.

Como está definido, los estímulos externos son temperatura (*T*) y energía (E), y la variable dependientes es la radiación (*R*), para poner en funcionamiento nntool debemos preparar estas variables de tal manera que la salida del sistema (*R*) debe estar definido por una matriz de 1xn, siendo n la cantidad de registros de salida conocidos, en nuestro caso, para el mes de enero, 3.224, lo cual correspode a la suma de los trece valores de radiacion existentes de 6:00 AM-6:00 PM en los ocho años conocidos del mes de enero, y luego ser importada en *Tarjet Data*. Las variables de entrada (*X*), las cuales deben ser importadas en *Input Data*, se deben agrupar en un vector, de tal manera que la primera fila corresponda a (*T*) y la segunda a (*E*), quedando de un tamaño de 2xn. Se observa que se deben importar dos entradas IN2001 y IN 2004, las cuales son usadas en la fase de funcionamiento y prueba de la red.

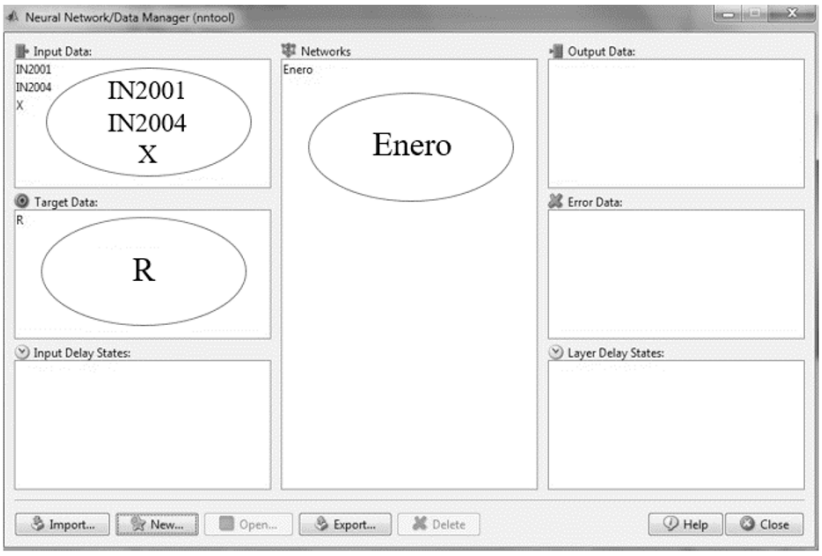


*Figure 3. Imported input and output variables / Variables de entrada y salida importadas*

El paso a seguir es ajustar los parametros de la red creada (**Figura 4**), en donde se deben modificar parametros como:

- tipo de red, como se definió, se implementa una RNA *feed-forward backpropagation;*
- estímulos de entrada *X* denotados por *T* y *E*;
- Salida de la red *R*;
- tipo de funcion de entrenamiento, definida por TRAINLM;
- función de aprendizaje LEARNGDM, la cual permite modificar los *W* y *b* de acuerdo con la cantidad de entrenamientos necesarios;
- tipo de cálculo de error MSE, el cual permite ajustar el rendimiento sobre el cuadrado significativo del error;
- cantidad de capas de la RNA definida de manera iterativa hasta lograr la mejor validación, en este caso se decide crear dos capas ocultas y la capa de salida;
- número de neuronas en cada capa, como ya se definió, no existe alguna metodología que indique cómo seleccionar el número de capas o número de neuronas; y
- tipo de relación de entradas y salidas, se asigna TANSIG para las dos capas ocultas, lo que permite el aprendizaje de la no-linealidad de los estímulos de entrada, y PURELIN para la capa de salida, la que relaciona los valores de las capas ocultas con la variable de salida, de manera lineal.

En la **Figura 5** se observa la estructura creada, la cual consta de las dos entradas (*T*) y (*E*); dos capas ocultas, cada una con 30 *PE* con una relacion TANSIG; y una capa de salida *R*, con una neurona y una función de relación PURELIN.

the sum of 13 radiation values from 6:00 AM to 6:00 PM in the eight known years in January. These records must then be imported into *Tarjet Data*. The input variables (*X*), which must be imported into *Input Data*, must be grouped into a vector in a way that the first row corresponds to (*T*) and the second to (*E*), giving as a result a 2 x n size. Two inputs IN2001 and IN 2004 must be imported, which are used in the operating phase and network testing.

The next step is to adjust the parameters of the network created (**Figure 4**), modify parameters such as:

- network type, as indicated previously, a NN *feed-forward backpropagation is implemented;*
- input stimuli *X* denoted by *T* and *E*;
- network output *R*;
- type of training function, defined by TRAINLM;
- learning function LEARNGDM, which allows one to modify *W* and *b* according to the amount of training required;
- type of calculation error MSE, which allows one to adjust the performance on the significant squared error;
- number of layers of the NN defined in an iterative way until the best validation is reached; in this case, it is decided to create two hidden layers and the output layer;
- number of neurons in each layer; as described above, there is no methodology that indicates how to select the number of layers or number of neurons; and
- type of relationship between inputs and outputs. TANSIG is assigned to the two hidden layers, allowing them to learn the nonlinearity of input stimuli, and PURELIN is assigned to the output layer, which relates the values of the hidden layers to the output variables in a linear way.

**Figure 5** shows the structure created, which consists of two inputs (*T*) and (*E*); two hidden layers, each with 30 *PE* with a TANSIG relationship; and an output layer R, with a neuron and a function relationship *R*.

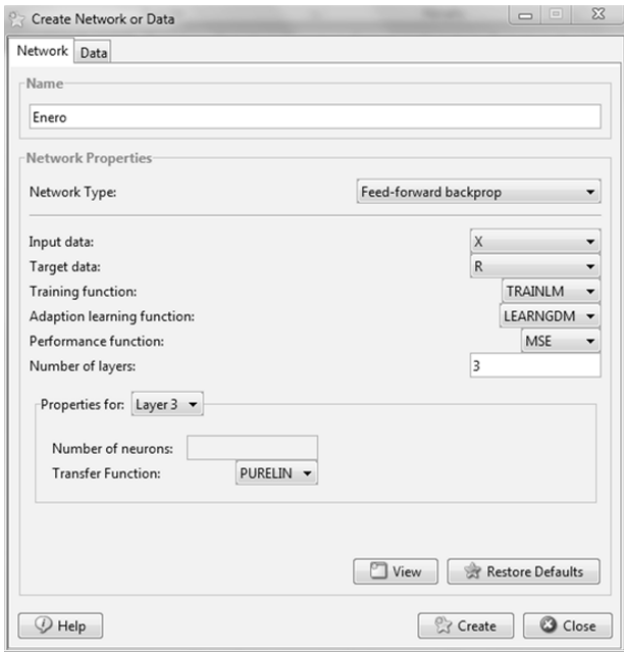


*Figure 4. Parameters of the network / Parámetros de la red*

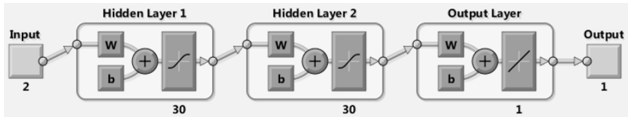


*Figure 5. Architecture of the network created / Arquitectura de la red creada*

## V. Training phase of NN

The ability to learn is a peculiar characteristic from the smart, biological or any kind of system. In artificial systems, learning is seen as the process of updating the internal representation of the system (*S*), as a response to external stimuli (*E*), in such a way that it can perform a specific task. This includes changing the network architecture, through adjustments in their connections (*W*) (Basheer & Hajmeer, 2000).

## V. Fase de Entrenamiento de la RNA

La capacidad de aprender es una característica peculiar de los sistemas inteligentes, biológicos o de otro tipo. En sistemas artificiales el aprendizaje es visto como el proceso de actualización de la representación interna del sistema (*S*), ante una respuesta a estímulos externos (*E*), de modo que pueda realizar una tarea específica. Esto incluye la modificación de la arquitectura de la red, a través de los ajustes en sus conexiones (*W*) (Basheer & Hajmeer, 2000).

El método general de entrenamiento de una RNA BP se resume en cinco pasos.

Pasos hacia delante:

1. Seleccionar un vector de entrada desde el conjunto de entrenamiento.
2. Aplicar esta entrada a la red y calcula la salida.

Pasos hacia atrás:

3. Calcular el error entre la salida calculada y la salida deseada de la entrada usada.
4. Ajustar los pesos para que el error cometido entre la salida calculada y la salida deseada disminuya
5. Repetir los pasos 1 al 5 para todas las entradas del conjunto de entrenamiento, hasta que el error global sea aceptablemente bajo.

Para iniciar la fase de entrenamiento de la RNA desde nntool debemos ingresar desde Training Info los parámetros (*E*/*S*) conocidos, y luego, desde Training Parameters, ajustar los valores de entrenamiento, como se muestra en la **Tabla 2** (Sumathi, Ashok, & Surekha, 2015).

Luego de importar las entradas y salida de la red y de ajustar los parametros se inicia el proceso de entrenamiento las veces que sea necesario, hasta alcanzar los índices de validación deseados, en este caso se logra obtener una validación de 11.16, con tres pruebas de entrenamiento.

*Table 2. TRAINLM training parameters / Parámetros de entrenamiento TRAINLM*

| Description / *Descripción* | Parameter / *Parámetro* | Value / *Valor* |
|---|---|---|
| Show information GIU / *Mostrar información GUI.* | showWindow | True |
| Generate output of command line / *Generar salida de línea de comandos.* | showCommandLine | False |
| Epochs between screens / *Épocas entre las pantallas.* | Show | 25 |
| Maximum number of epochs / *Número máximo de épocas.* | Epochs | 1000 |
| Maximum training time / *Tiempo máximo para entrenar.* | Time | Inf |
| Performance target / *Objetivo de rendimiento.* | Goal | 0 |
| Minimum performance slope / *Pendiente mínima de rendimiento.* | min_grad | 1e-07 |
| Maximum validation failures / *Fracasos máximos de validación.* | max_fail | 6 |
| Initial value mu/ *Valor inicial mu.* | Mu | 0.05 |
| Decrease factor mu/ *Factor de disminución mu.* | mu_dec | 0.9 |
| Increase factor mu / *Factor de aumento mu.* | mu_inc | 10 |
| Maximum value mu / *Valor máxima mu.* | mu_max | 1 e10 |

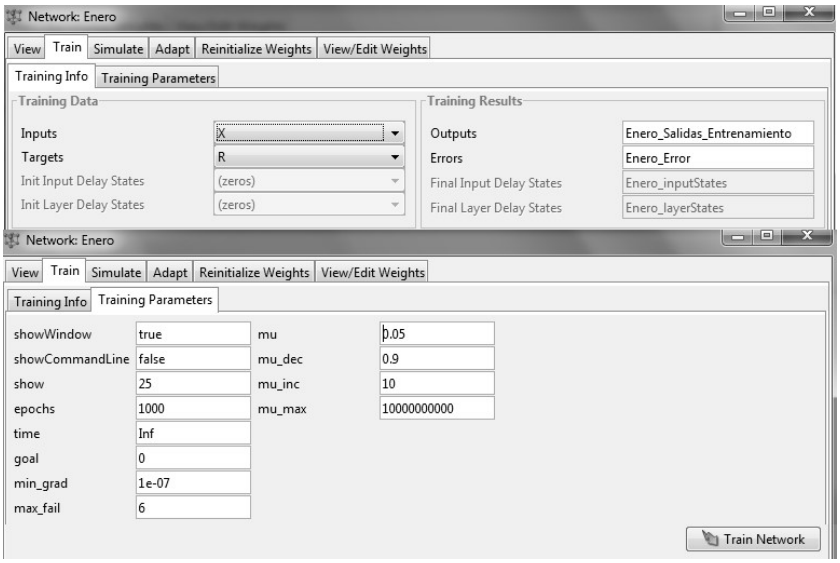


*Figure 6. Information and training parameters of the network / Información y parámetros de entrenamiento de la red*

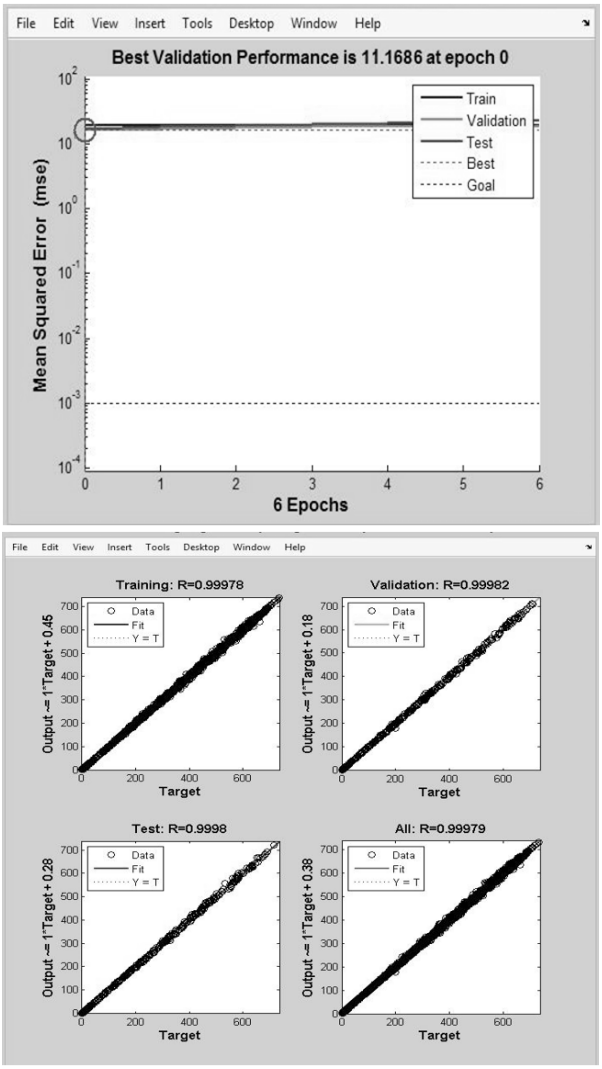


*Figure 7. Results of training (a) and validation (b) of the NN / Resultados de entrenamiento (a) y validacion (b) de la RNA*

The general method of training of an NN BP is summarized in five steps.

Forward steps:

1. Select an input vector from the training set.
2. Apply this entry to the network and calculate the output.

Backward steps:

3. Calculate the error between the calculated output and the desired output of the used input.
4. Adjust the weights so that the error between the calculated output and the desired output decreases.
5. Repeat steps 1–5 for all entries in the training set until the global error is acceptably low.

To start the training phase of NN from nntool we must enter from Training Info known parameters (*I/O*), and later, from Training Parameters, adjust training values, as shown in **Table 2** (Sumathi, Ashok, & Surekha 2015).

After importing inputs and outputs from the network and having adjusted the parameters, the training process begins as many times as neccessary, to reach the desired validation indices. In this case, it is possible to obtain a validation of 11.16, with three training tests.

## VI. Operation phase and NN testing

In the training process of the NN, the nntool tool updates the value of the weights (*W*) and the bias (*b*) automatically, allowing one to continue with the operating phase and testing from the same interface.

Since the implementation of the NN, vectors of the incomplete variables had to be imported (**Figure 3**). In the case of January, years with missing data are 2001 and 2004, of which only 25% and 39% are known data, respectively.

The input vector for 2001 was denoted as IN2001, where it was grouped in a way that the first row corresponds to (*T*) and the second to (*E*); as only 25% of these values are known, blanks or unknown spaces must be filled with any wrong number (see **Table 3**).

| Date / *Fecha* | Time / *Hora* | *T* | *E.* | R incomplete / *R incompleto* | Enero_ouput_2001 |
|---|---|---|---|---|---|
| 24/01/2001 | 11:00a | -9999 | -9999 | -- | 478,07 |
| 24/01/2001 | 12:00p | -9999 | -9999 | -- | 575,49 |
| 24/01/2001 | 1:00p | -9999 | -9999 | -- | 542,17 |
| 24/01/2001 | 2:00p | -9999 | -9999 | -- | 469,78 |
| 24/01/2001 | 3:00p | -9999 | -9999 | -- | 431,00 |
| 24/01/2001 | 4:00p | -9999 | -9999 | -- | 301,38 |
| 24/01/2001 | 5:00p | -9999 | -9999 | -- | 164,48 |
| 24/01/2001 | 6:00p | 28,7 | 2,6 | 42 | 39,22 |
| 25/01/2001 | 6:00a | 22,6 | 0 | 0 | 0,33 |
| 25/01/2001 | 7:00a | 21,9 | 1,4 | 16 | 16,61 |
| 25/01/2001 | 8:00a | 23,1 | 13,6 | 158 | 157,18 |
| 25/01/2001 | 9:00a | 25,7 | 27 | 307 | 310,09 |
| 25/01/2001 | 10:00a | 26,7 | 39,6 | 459 | 456,55 |

*Tabla 3. Input and output values in operation phase and RNA testing / Valores de entrada y salida en fase de funcionamiento y prueba de la RNA*

To run the NN test from nntool the IN2001 variable must be selected as input and assigned a name to the output variable, thus obtaining the vector corresponding to the values of completed radiation (**Figure 8**).

## VII. Results

Historical data series allow us to know the behavior of a variable with a high level of reliability. In this case, an incomplete series has many drawbacks at the moment of calculating averages due to missing data. Thanks to the procedure carried out, these problems were solved, and a multi-year series was obtained with time intervals that define the behavior of the global radiation.

The results show that most radiation occurs in the months between August and October, with an average between 3.9–4.2 kW/m2; and the lower radiation occurs between November and January, with an average between 3.4–3.2 kW/m2 (**Figure 9**). These values converge with the results obtained by UPME/IDEAM (2005) and Leal and Hernández (2013), who demonstrated multi-year average radiation for the city of Cúcuta of 3.5–4.0 kW/m2, an interval in which the value of 3.51 kW/m2 obtained here is adjusted.

The use of strategies that allow us to complete missing data provides advantages of obtaining mean values more accurately in short time intervals. Figure 10 shows the radiation scale (W/m2) in the hours and

## VI. Fase de funcionamiento y prueba de la RNA

En el proceso de entrenamiento de la RNA, la herramienta nntool actualiza el valor de los pesos (*W*) y los bias (*b*) automáticamente, lo cual brinda la facilidad de continuar con la fase de funcionamiento y prueba desde la misma interfaz.

Desde la implementación de la RNA se debió importar los vectores de las variables incompletas (**Figura 3**). Para el caso de mes de enero se observa que los años con datos faltantes son 2001 y 2004, de los cuales sólo se conoce el 25% y el 39% de los datos, respectivamente.

El vector de entrada para 2001 fue denotado como IN2001, donde se agrupó de tal manera que la primera fila corresponda a (*T*) y la segunda a (*E*); como solo se conoce el 25% de estos valores se debe rellenar los lugares en blanco o desconocidos con cualquier número erróneo (ver **Tabla 3**).

Para ejecutar la prueba de la RNA desde nntool se debe seleccionar la variable IN2001 como input y asignar un nombre a la variable de salida, y de esta manera obtener el vector que corresponde a los valores de radiación completados (**Figura 8**).

## VII. Resultados

Las series históricas permiten conocer el comportamiento de una variable con bastante confiabilidad, en este caso una serie incompleta posee muchos inconvenientes al momento de calcular promedios debido a los datos faltantes. Gracias al procedimiento llevado a cabo se resolvieron estos problemas y se obtuvo una serie multianual con periodicidad horaria que define el comportamiento de la radiación global.

Los resultados obtenidos muestran que la mayor radiación se presenta en los meses de agosto-octubre, con un prome-

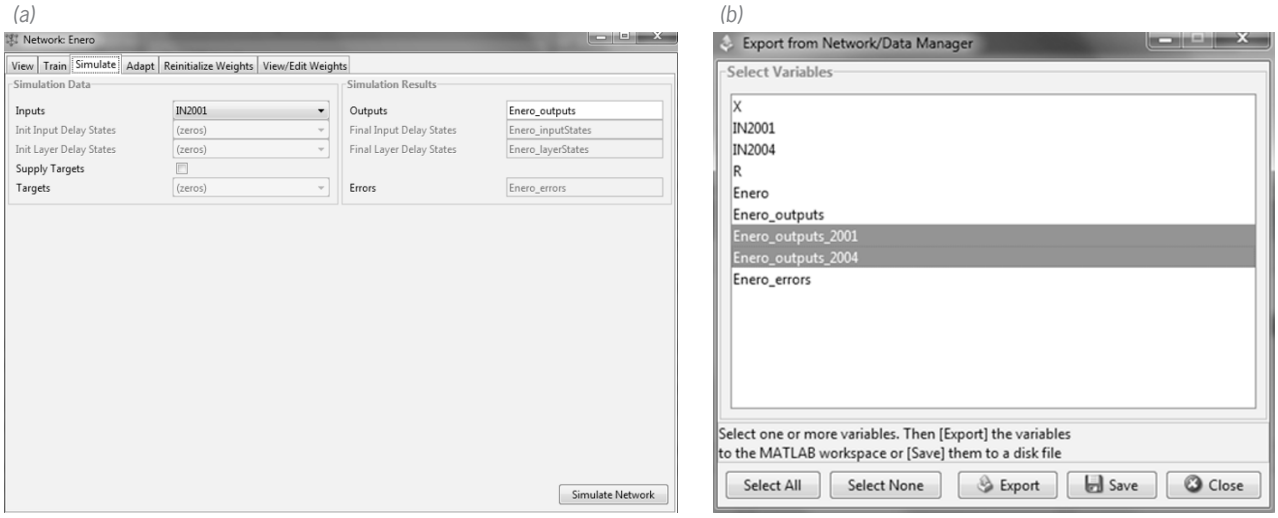


*Figure 8. Test phase parameter (a), output network variables (b) / Parámetro de fase de prueba (a), variables de salida de la red (b)*

dio entre 4.2-3.9 KW/m2 y que los meses de menor radiación se presentan de noviembre a enero, con promedios entre 3.4-3.2 KW/m2 (**Figura 9**), estos valores convergen con resultados similares a los obtenidos por UPME/IDEAM (2005) y Leal y Hernández (2013), quienes exhiben una radiación promedio multianual para la ciudad de Cúcuta de 3.5-4.0 KW/m2 , intervalo en el cual se ajusta el valor de 3.51 KW/m2 obtenido en este estudio.

La utilización de estrategias que permiten completar datos ausentes proporciona la ventaja de obtener valores medios con mayor exactitud en intervalos cortos de tiempo. En la **Figura 1**0 se observa la escala de radiación (W/m2) presente en las horas y meses del año, información vital en procesos que buscan optimizar el aprovechamiento de la energía. Un ejemplo claro de ello es un seguidor solar inteligente que busca tener un control en el punto de máxima captación de energía.

## VIII. Conclusiones

El uso de redes neuronales artificiales para completar datos ausentes en variables meteorológicas se convierte en una solución viable y de fácil manejo gracias a herramientas computacionales como *nntool*. Estas proporcionan porcentajes mínimos globales en la validación de resultados, algunos inferiores al 10%. La precisión de estas validaciones depende directamente del tipo de entrenamiento y del porcentaje de valores conocidos; también, depende del número de entradas a analizar en el sistema. En este caso, solo se analizaron dos (temperatura y energía), pero se recomienda introducir otras variables meteorológicas, como temperatura máxima, temperatura mínima, temperatura del viento, presión y humedad relativa, debido a su relación con la radiación global.

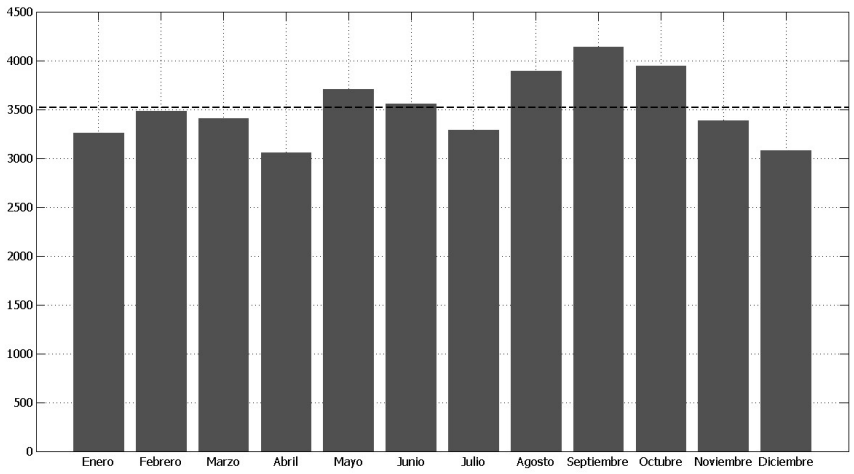


*Figure 9. Monthly average and annual average of global radiation in the city of Cúcuta / Promedios mensuales y media anuales de radiación global, en la ciudad de Cúcuta*

It is not recommended to complete values in series where lower percentages than 20% are known, for not having the appropriate amount of data for validation and test results, which provides a high level of uncertainty in the output variable. This particular case occurred in eight months of the series, in which all data are unk-

## VIII. Conclusions

The use of artificial neural networks to complete missing data in meteorological variables becomes a viable solution and has easy handling thanks to computational tools as *nntool*. They provide global minimum percentages in the validation of results, some less than 10%. The accuracy of these validations depends directly on the type of training and the percentage of known values; also, it depends on the number of entries to be analyzed in the system. In this case, only two were analyzed (temperature and energy), but it is recommended to introduce other meteorological variables, such as maximum temperature, minimum temperature, wind temperature, pressure and relative humidity, because of their relationship to the global radiation.

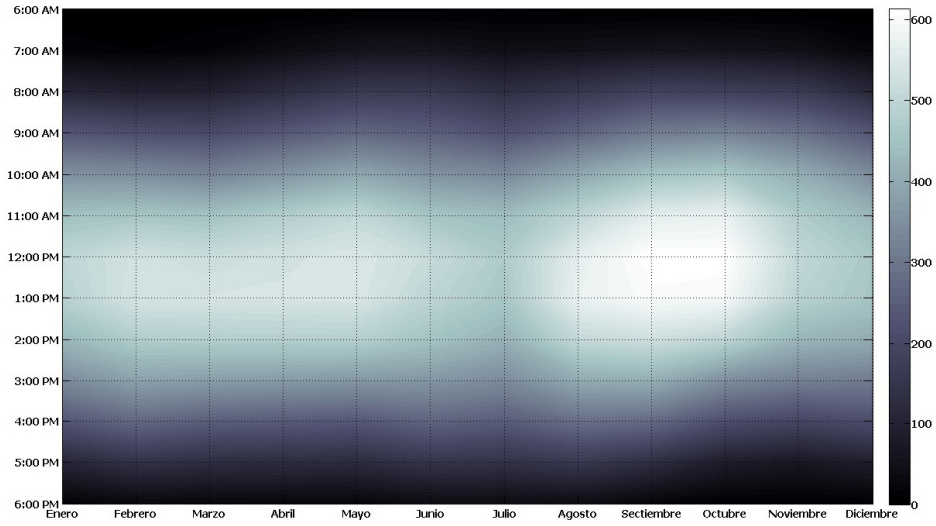


*Figure 10. Average hours in a multi-year series of global radiation in the city of Cúcuta / Promedio Horario en serie multianual de radiación global en la ciudad de Cúcuta*

nown; for that reason, it was decided to perform an average of the known input variables, and to enter them in the input variable for the test phase of the NN. ST

tacionales como *nntool*. Brinda porcentajes globales mínimos en la validación de resultados, en algún caso inferiores al 10%, la exactitud de estas validaciones depende directamente del tipo de entrenamiento y del porcentaje de valores conocidos, además, depende de la cantidad de entradas a analizar en el sistema. En este caso solo fueron analizadas dos de ellas, temperatura y energía, pero se recomienda introducir otras variables meteorológicas, tales como temperatura máxima, temperatura mínima, temperatura del viento, presión y humedad relativa, por su relación con la radiación global.

No se recomienda completar valores en series donde se conozcan porcentajes menores al 20%, por no poseer la cantidad de datos adecuados para la validación y prueba de resultados, lo cual proporciona gran incerteza en la variable de salida. Este caso particular sucedió en ocho meses de la serie, en los cuales se desconoce la totalidad de los datos; por tal motivo, se decidió realizar un promedio de las variables de entrada conocidas, e ingresarlas en la variable de entrada para la fase de prueba de la RNA. ST

## *References / Referencias*


Basheer, L., & Hajmeer, M., (2000). Artificial neural networks: fundamentals, computing, design, and application. *Journal of Microbiological Methods*, 43(1), 3-31.

Colmenares, G. (n.d.). *Análisis multivariante inteligencia artificial y sus aplicaciones* [material del curso - Postgrado en Economía]. Mérida, Venezuela: Universidad de los Andes. Retrieved from: http://www.webdelprofesor.ula.ve/economia/gcolmen/postgrado2.html

Hamzaoui A., Hernández J., Martinez, S., Bassam, A., Álvarez, A., & Lizama, C. (2011). Optimal performance of COD removal during aqueous treatment of alazine and gesaprim commercial herbicides by direct and inverse neural network. *Desalination*, 277(1), 325-337.

Hernández, J., Bassam, A., Siqueiros, j., & Juarez, D., (2009). Optimum operating conditions for a water purification process integrated to a heat transformer with energy recycling using neural network inverse. *Renewable Energy*, 34(4), 1084-1091.

Infante, S., Ortega J., & Cedeño, F. (2008). Estimación de datos faltantes en estaciones meteorológicas de Venezuela vía un modelo de redes neuronales. *Revista de climatología.* 8 51-70.

Jamett, M. (2004). *Feedforward convergence and stability analysis from a set perspective: State estimation approach* [doctoral thesis]. Universidad de Santiago de Chile.

Leal, F. & Hernández, M. (2013). Estudio del potencial eólico y solar de Cúcuta, Norte de Santander. *Revista Colombiana de Tecnologías de Avanzada*, 2(22), 27-33.

Medina, R. (2008). *Estimación estadística de valores faltantes en series históricas de lluvia* [thesis]. Pereira, Colombia: Universidad Tecnológica de Pereira.

Ponce, P. (2010). *Inteligencia artificial con aplicaciones a la ingeniería* [1ª ed.]. México DF: alfaomega.

Ramírez, F. (2010). *Sistema para la predicción de posición y seguimiento de un conjunto de náufragos basado en redes neuronales* [tesis de maestría]. Universidad Complutense de Madrid: España.

San Juan, E., Jamett, M., Kaschel, H., & Sánchez, L. (2015). Sistema de reconocimiento de voz mediante wavelets, predicción lineal y redes backpropagation. Ingeniare. *Revista chilena de ingeniería*, 24(1), 8-17.

Serlin, J. (2010). *Conocimiento de la gestión de las organizaciones: sistemas complejos dinámicos inestables adaptativos* [doctoral thesis]. Argentina: Universidad de Buenos Aires.

Sumathi, S., Ashok, L., & Surekha, P., (2015). Solar PV *and wind energy conversion systems: an introduction to theory, modeling with matlab/simulink, and the role of sofy computing techniques* [1a ed.]. Cham, Switzerland: Springer. doi:10.1007/978-3-319-14941-7.

Unidad de Planeación Minero Energética [UPME], Instituto de Hidrología, Meteorología y Estudios Ambientales [IDEAM]. (2005). *Atlas de radiación solar en Colombia*. Bogotá, Colombia: UPME/IDEAM.

Vásquez, A., Rojas, J., & Duarte, E. (2015). Evaluación y caracterización del recurso eólico en la Universidad Francisco de Paula Santander Cúcuta y prospectiva para el aprovechamiento energético en el Norte de Santander. *El Hombre y la Máquina*, 46, 144-152.

## *CURRICULUM VITAE*

*Franklin Meer García Acevedo* Student (Universidad Francisco de Paula Santander, Cúcuta-Colombia) and researcher at Grupo de Investigación en Procesos Industriales GIDPI / Estudiante investigador de la Universidad Francisco de Paula Santander (Cúcuta, Colombia), miembro del Grupo de Investigación en Procesos Industriales GIDPI.

*Juan Andrés Rojas Serrano* Electromechanical Engineer from Universidad Francisco de Paula Santander (Cúcuta, Colombia) and researcher at Fluidos y Térmicas research group, with experience in energy conversion systems / Ingeniero Electromecánico de la Universidad Francisco de Paula Santander (Cúcuta, Colombia), investigador del grupo de Fluidos y Térmicas [FLUTER], con experiencia en sistemas de conversión de energía.

*Darío Alejandro Vásquez Vega* Electromechanical Engineer from Universidad Francisco de Paula Santander (Cúcuta, Colombia) and researcher at Fluidos y Térmicas research group, with experience in renewable energies / Ingeniero Electromecánico de la Universidad Francisco de Paula Santander (Cúcuta, Colombia), investigador del grupo de Fluidos y Térmicas [FLUTER], con experiencia en energías renovables.

*Diego Alejandro Parra Peñaranda* Student (Universidad Francisco de Paula Santander, Cúcuta-Colombia) and researcher at Grupo de Investigación en Procesos Industriales GIDPI / Estudiante investigador de la Universidad Francisco de Paula Santander (Cúcuta, Colombia), miembro del Grupo de Investigación en Procesos Industriales GIDPI.

*Erney Fabián Castro Becerra* Student (Universidad Francisco de Paula Santander, Cúcuta-Colombia) and researcher at Grupo de Investigación en Procesos Industriales GIDPI / Estudiante investigador de la Universidad Francisco de Paula Santander (Cúcuta, Colombia), miembro del Grupo de Investigación en Procesos Industriales GIDPI.