SEj
Vol. 2, N° 2, 2018

Journal of Sciences and Engineering

A publication of
CINCADER
Centre of Research and Training for Regional Development
Online at www.journals.cincader.org

ISSN 2523-9503

Backpropagation to predict drinking water consumption

Raul Eduardo Huarote Zegarra[a,*], Mónica Patricia Romero Valencia[a]
[a]Universidad Cesar Vallejo
[b]Universidad Nacional Federico Villarreal
raulhuarote@yahoo.es

Resumen

El presente artículo de investigación se basa en el uso de las redes neuronales artificiales como herramienta para la predicción aplicado al consumo de agua potable, donde se aprovecha el aprendizaje artificial propio de una red multicapa backpropagation con los datos históricos consumidos en m3. En un computador de regular característica se logró implementar la red backpropagation en el lenguaje Python 3.7.0, tomando como objetos de estudio un caso publicado en su página web de la SUNEDU respecto al consumo de agua potable en el 2017 y comprobando los resultados en enero del 2018, tomando estos datos para el aprendizaje y las pruebas respectivas. Se ha logrado como resultado predecir la cantidad de consumo de agua de la institución. La prueba realizada dio como resultado en exceso 23 m3 lo que representa el 2.7% en exceso dado que en enero del 2018 se registró un consumo de 833m3. Tener en cuenta que se realizó para el proceso de entrenamiento con un error máximo para el entrenamiento de 0.000099 y con una cantidad máxima de iteraciones de 100000. Posterior al entrenamiento con los datos históricos del consumo de agua potable se logró predecir el consumo de la SUNEDU a enero del 2018

Palabras clave: Redes neuronales artificiales, Backpropagation, predicción

Abstract

The present research article is based on the use of artificial neural networks as a tool for the prediction applied to the consumption of drinking water, where the artificial learning of a multilayer backpropagation network with the historical data consumed in m3 is used. In a computer with a regular feature, the backpropagation network was implemented in the Python 3.7.0 language, taking as a study object a case published on its SUNEDU website regarding drinking water consumption in 2017 and checking the results in January of 2018, taking these data for learning and the respective tests. It has been achieved as a result to predict the amount of water consumption of the institution. The test performed resulted in an excess of 23 m3, which represents 2.7% in excess given that in January 2018 a consumption of 833m3 was recorded. Bear in mind that it was done for the training process with a maximum error for the training of 0.000099 and with a maximum number of iterations of 100000. After training with the historical data of drinking water consumption, it was possible to predict the consumption of SUNEDU to January 2018

Keywords: Artificial neural networks, BackPropagation, prediction.

Please cite this article as: Huarote Raul, Romero Monica; Backpropagation to predict drinking water consumption, Journal of Sciences and Engineering, vol. 2, no 2, 2018, pp. 31-39.

SEj
Vol. 2, N° 2, 2018
Journal of Sciences and Engineering
A publication of
CINCADER
Centre of Research and Training for Regional Development
Online at www.journals.cincader.org

ISSN 2523-9503

1. Introduction

We start from the general problem of being able to predict the consumption of drinking water in a given city, it is because of the need not to produce in excess or in defect mainly generating economic loss and dissatisfaction of the consumed respectively. Because knowing that it is impossible to give as consumption of drinking water in an exact quantity, the idea is to find a measure that comes close to an amount of consumption of a later date, therefore, the companies producing drinking water are going to measure their future production. This goes hand in hand with economic savings, decision making, consumer satisfaction, etc. Considering that the prediction error obtained is as small as possible, as shown in the following equation obtained from (Casimiro, 2009).

$$e_T(\ell) = Y_{T+\ell} - Y_T(\ell) \qquad (1)$$

Where it reflects that the prediction error is equal to the difference between the value obtained in a later time minus the predicted value.

Taking textually (Casimiro, 2009) where it says "In good logic a decision should not be taken without considering the future evolution of all those events that condition it", therefore it is a requirement to each of the entities to know subsequent information or to predict in order to make a better decision.

The field of application of the prediction is in multiple areas, without mentioning all of them, in reality it is where it requires future information in order to make better decisions.
Considering the different methods to predict, we have the most outstanding such as the statistical methodology of (Jenkins, 1970) ARIMA (autoregressive integrated moving average) dedicated to identify, estimate and diagnose dynamic models of time series where the time variable is considered a very important factor.

Also found according to (García, 2017) is the simple exponential smoothing method, where it estimates for each period T the parameter a as a weighted sum of all previous or historical observations, considering more relevant the more current observations compared to the older ones. As shown in the following equation.

$$\hat{a}_T = S_T = \propto X_T + (1 - \propto) S_{T-1} \qquad (2)$$

Where S_{T-1} is the estimation of a obtained in the period T-1 and α is the smoothing constant that takes values between 0 and 1.

According to (Janert, 2010) shows the regression technique is the most used numerical prediction method where it is used to predict the value of a continuous or ordered variable. Where it relates one or more independent variables (predictors) with a dependent variable or response variable. Considering for it the linear regression, where it is shown in the following equation.

$$y = w_0 + w_1 x \qquad (3)$$

Where the regression coefficients are w0 is the displacement w1 slope.

Another method of prediction is multiple linear regression, where many non-linear expressions can be expressed in a linear function. As shown in the following equation:

$$y = w_0 + w_1 x_1 + w_2 x_2 + \cdots \qquad (4)$$

SEj
Vol. 2, N° 2, 2018
Journal of Sciences and Engineering
A publication of
CINCADER
Centre of Research and Training for Regional Development
Online at www.journals.cincader.org

ISSN 2523-9503

For the Holt method, according to (Bowerman, 2009) it is used when there is a time series and there is an exponential trend, increasing or decreasing, and it can be modeled, based on the smoothing method, as shown in the following equation:

$$l_T = \propto y_T + (1 - \propto)[l_{T-1} + b_{T-1}] \qquad (5)$$

Where the smoothing constants are α and b for the time series.

In order to solve this type of problem in particular investigations have been carried out as (Romero, 2012) with his investigation "prediction of academic performance in the new degrees of the EPS" where he uses as method to predict the mining of data used the software weka. Also in the research of Jorge Álvarez Blanco (2016) entitled "Prediction of academic results of computer science students through the use of neural networks" where they use the method of artificial neural network FBR (Radial Base Function) using for its implementation the Matlab.
The model proposed in this research to identify using the artificial backpropagation neural network to predict water consumption in m3. According to (Rumelhart, 1986) they presented a method for a neural network to learn input patterns with similar characteristics, connecting more levels (layers) than Rosenblatt used to develop the simple Perceptron. Considered as supervised learning, since it requires a tutor to perform its process, backpropagation, is done through learning-error. And his proposed model is shown in Figure 1

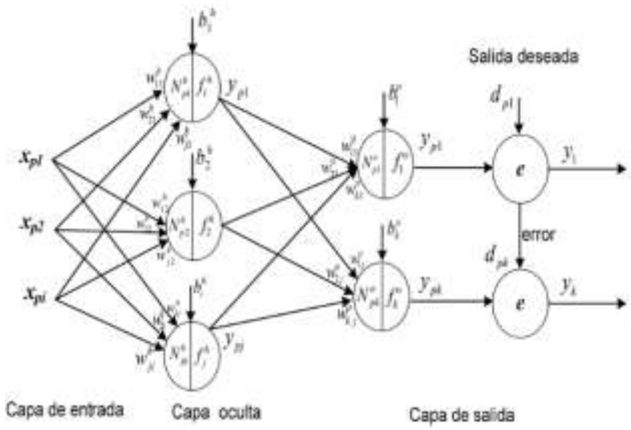

Figure 1. Back-propagation neural network model. Retrieved from (Rumelhart, 1986)

In figure 1 it represents a model of an artificial multilayer neural network, where it has 01 input layer, 02 hidden layers and 01 output layer, where [Xp1,Xp2,Xp3...Xpi] represents the input patterns, Wij represents the values of the synaptic weights, which represents what is learned. bk represents the pathways, which is nothing more than a neuron for balance. The fi represents the activation function where the sigmoid function is used. The value of yk represents the output values obtained, which will be compared with the desired output and the difference represents the e learning error.

The programming language Python 3.7.0 was used for the implementation of the neural network model.

For the implementation of the source code, Python 3.7.0 was used to show the results. Highlighting that the language has libraries that are added to Python, but it was decided to codify it to know how it works in the learning process and see how it can be tested for particular cases.

SEj
Vol. 2, N° 2, 2018
Journal of Sciences and Engineering
A publication of
CINCADER
Centre of Research and Training for Regional Development
Online at www.journals.cincader.org

ISSN 2523-9503

2. Materials and Methods

The model to be implemented is given by 1 entry layer with 2 neurons (additionally the 1 bia), 2 hidden layers with 8 neurons for the entry layer to the first hidden layer, 7 neurons for the first hidden layer to the second hidden layer and a neuron for the exit layer.

It should be noted that it has an output neuron, since it must learn a predicted value of a time series. Where the value of tn-2 represents the value in a certain time (prior to tn-1), and the value of t represents the value with respect to the current time. The model to be presented is shown in figure 2.

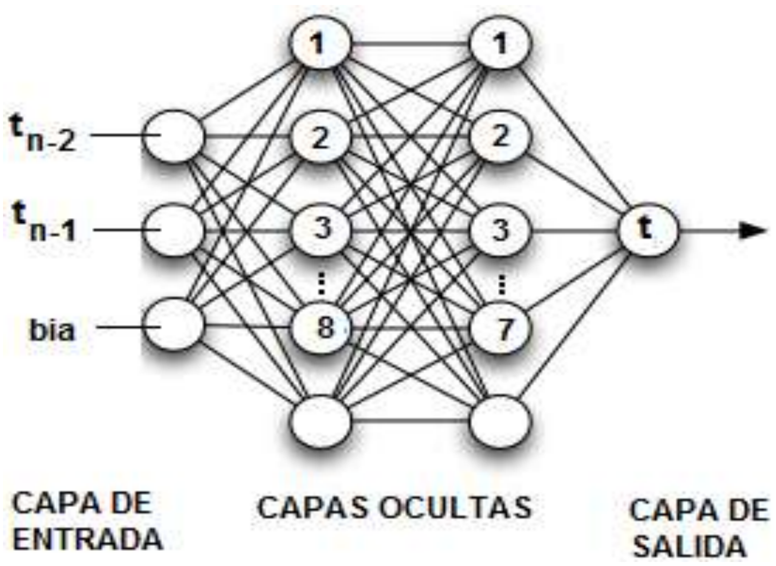

Figure 2. Proposed backpropagation neural network model for prediction.

Learning factor is considered sigmoid

$$f(x) = \frac{1}{1 + e^{-x}}$$

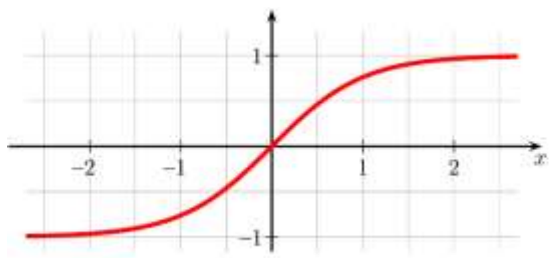

Figure 3. On the left side the equation and on the right side the graph of the sigmoid function as activation function.

Scale cubic meter values to a range of [0 to 1] by applying the following equation:

SEJ
Journal of Sciences and Engineering
Vol. 2, N° 2, 2018

ISSN 2523-9503
A publication of
CINCADER
Centre of Research and Training for Regional Development
Online at www.journals.cincader.org

Table 1. Equations for scaling and inverse process

Equation	Detail	Observation
$r = \frac{Y - Min}{Max - Min}$ $Y = r(Max - Min) + Min$	r: scaled value Y: courage to climb, Min: Minimum value of the dataset Max: maximum of the data set.	Necessary to scale in the range of 0 to 1 Necessary to return the original proportion as a function of r

Where it is shown the learned results converge in function of figure 4, this for the learning process is smooth.

The model that is going to be considered for the learning process of prediction, depends on the time, where tn-4 and tn-3 will generate tn-2, tn-3 and tn-2 will generate tn-1, tn-2 and tn-1 will generate t, considering these antecedents it is possible to predict the value of '?' where tn+1 represents. Therefore, for each value it will represent the input patterns and a respective output. As shown in figure 4:

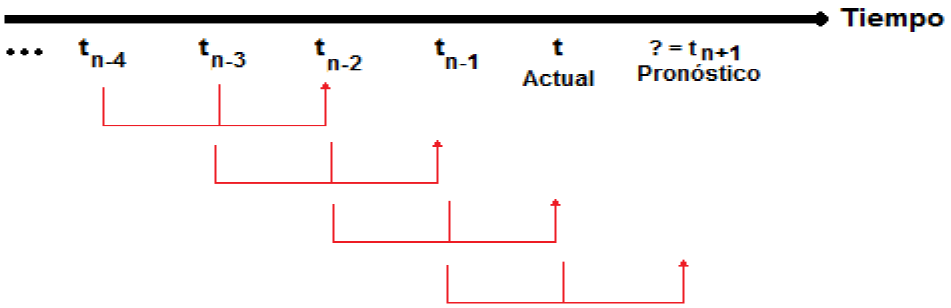

Figure 4. Timeline for obtaining data.

Algorithm for the network training process
WHILE error>ErrorMax and i<iteractions DO
Error = 1.0
FROM p TO DO patterns
GetError= backpropagation(input, output)
If error<ErrorMaximum
"I LEARN"
  END FROM
END WHILE

Analyzing the algorithm based on the backpropagation neuronal network to predict water consumption, in the first line indicates the learning process according to the number of iterations established or if the error is still greater than the allowed errorMax, in the second line indicates that the error will be assigned initially to 1.0 (to enter the for), in the third line will go according to the amount of input patterns that counts for the training process, in the third line you get the error while doing the training, in the fourth line evaluates whether the error is less than allowed to display a message "Learned" This is displayed in Figure 5.

SEj
Vol. 2, N° 2, 2018

Journal of Sciences and Engineering

A publication of
CINCADER
Centre of Research and Training for Regional Development
Online at www.journals.cincader.org

ISSN 2523-9503

```
def entrenamiento(self, patrones, iteraciones=100000, N=0.6, M=0.1, ErrorMax=0.000099):
    # N: razon de aprendizaje
    # M: momentum factor

    i = 0
    error = 1.0
    while (error>ErrorMax and i<iteraciones):
        error = 0.0
        for p in patrones:
            inputs = p[0]
            targets = p[1]
            self.update(inputs)
            error = error + self.backPropagate(targets, N, M)

        if(error<ErrorMax):
            print(i,' APRENDIO %-.5f' % error)

        if i % 1000 == 0:
            print('error %-.5f' % error)
        i = i+1
```

Figure 5. Source code for training

3. Results

It should be noted that the tests were done with encouraging results, was conducted with the institution SUNEDU which is a state entity that promotes research:
Of the 12 original values (for each month in 2017) where it represents in m3 the consumption of drinking water by SUNEDU, obtained from the URL https://intranet.sunedu.gob.pe/documentos/directorios/188/agua-potable-diciembre.pdf, where 22633 iterations were carried out for the training process, but I finish before completing them since an error of 0.00009 (less than the maximum permissible error) was obtained as shown in figure 7. With respect to the results obtained that were part of training and it was verified that the maximum error after the training is 0,006282469 therefore the following comparative figure is reflected after what was learned, where the values are superimposed (this for the error is almost 0), where the value predicted for t+1 is given by 0,2138466175 that represents in 809.91101 m3 (obtained by carrying out the inverse process of scaling y=r(Max-Min)+Min) of drinking water.

Table 2. Table of tests performed on values that are part of training

Time	Original Value	Scaled value (VE)	Value obtained (VO)	Error for each value (VE-VO)
t-11	1004	0,581439394	0,581439394	0
t-10	1005	0,583333333	0,583333333	0
t-9	1225	1	0,993717531	0,006282469
t-8	1032	0,634469697	0,639003762	-0,004534065
t-7	960	0,498106061	0,499525459	-0,001419398
t-6	697	0	0,00184481	-0,00184481
t-5	771	0,140151515	0,151187891	-0,011036376
t-4	768	0,134469697	0,145162373	-0,010692676
t-3	858	0,304924242	0,311021482	-0,006097239
t-2	911	0,40530303	0,40811843	-0,0028154
t-1	904	0,392045455	0,392919187	-0,000873732
t	789	0,174242424	0,174273629	-3,12049E-05
t+1			0,2138466175	

SEj
Vol. 2, N° 2, 2018

ISSN 2523-9503

Journal of Sciences and Engineering

A publication of
CINCADER
Centre of Research and Training for Regional Development
Online at www.journals.cincader.org

Figure 6. Results after the learning process of the data

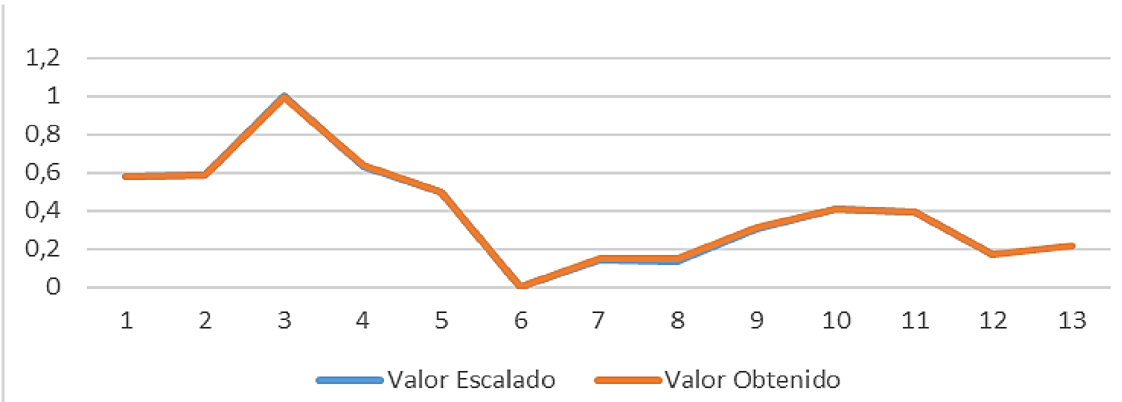

Figure 7. Graphical results after the learning process of the data

SOURCE CODE

In order to carry out the process of predicting the consumption of drinking water, it has been necessary to carry out the coding in a language that allows working with functions, this to separate each of the functions of the backpropagation network as shown below:

SEJ
Vol. 2, N° 2, 2018

Journal of Sciences and Engineering

A publication of CINCADER
Centre of Research and Training for Regional Development
Online at www.journals.cincader.org

ISSN 2523-9503

```python
import math
import random
import string

random.seed(0)

# calculate a random number where:  a <= rand < b
def rand(a, b):
    return (b-a)*random.random() + a

# Make a matrix (we could use NumPy to speed this up)
def makeMatrix(I, J, fill=0.0):
    m = []
    for i in range(I):
        m.append([fill]*J)
    return m

# our sigmoid function, tanh is a little nicer than the standard 1/(1+e^-x)
def sigmoid(x):
    return math.tanh(x)

# derivative of our sigmoid function, in terms of the output (i.e. y)
def dsigmoid(y):
    return 1.0 - y**2
```

Figure 8. Source code of the functions required to perform the training

```python
def demostracion():
    datos = [1004,1005,1225,1032,960,697,771,768,858,911,904,789]# datos de la linea de tiempo
    datosE = []
    pat1 = []
    maximo,minimo = float(max(datos)),float(min(datos))
    # realiza el escalado de los datos en el rango de [0 a 1]
    for d in datos:
        datosE.append((d-minimo)/(maximo-minimo))

    for i in range(0,len(datosE)-2):
        E = [[datosE[i],datosE[i+1]],[datosE[i+2]]]
        pat1.append(E)
    #valores para realizar el testeo obtenidos de la SUNEDU
    Test =[
        [0.581439394,0.583333333],[0.583333333,1],[1,0.634469697],[0.634469697,0.498106061],
        [0.498106061,0],[0,0.140151515],[0.140151515,0.134469697],[0.134469697,0.304924242],
        [0.304924242,0.40530303],[0.40530303,0.392045455],[0.392045455,0.174242424]
    ]
    #datosE : Datos escalados
    print(datosE)
    # create la red con una capa de entrada con 2 neuronas
    # primera capa oculta con 8 neuronas y la segunda capa oculta con 7 neuronas
    # capa de salida con 1 neurona
    n = NN(2, 8,7, 1)
    # entrenamiento con los patrones de entrada y salida
    n.entrenamiento(pat1)
    # se va a realizar el testeo
    n.test1(Test)
```

Figure 9. Source code for the training process

4. Conclusions

It is demonstrated that the prediction of water consumption can be made using as a tool the artificial neural network Backpropagation considering a maximum of iterations of 100000, learning

SEj
Vol. 2, N° 2, 2018

ISSN 2523-9503

Journal of Sciences and Engineering

A publication of
CINCADER
Centre of Research and Training for Regional Development
Online at www.journals.cincader.org

ratio of 0.6, moments of 0.1 and maximum allowed Error of 0.00009 and 02 hidden layers with size according to the proportion of data to train.
The results were encouraging, as it resulted in excess 23 m3 which represents 2.7% in excess given that the January 2018 publication of water consumption recorded a consumption of 833m^3

References

Bowerman, B. 2009. Pronósticos, series de tiempo y regresión. Un enfoque aplicado. México: C. Learning editores.

Casimiro, M. 2009. Técnicas de predicción económica. Vasco: sarriko-on.

Janert, P. 2010. Data Analysis Analysis with Open Source Tools. O'Reilly.

Jenkins, B. 1970. Time Series Analysis Forecasting, and Control. San Francisco: Soc. Ind. Appl. Math.

Álvarez Jorge, 2016. Predicción de resultados académicos de estudiantes de informática mediante el uso de redes neuronales. La habana: Scielo Chile.

García Matilla. 2017. Econometría y predicción. Madrid. España: McGraw-Hill Interamericana de España.

Romero C., 2012. Predicción del rendimiento académico en las nuevas titulaciones. Córdova - España: Taller JENUI,

Rumelhart, H. 1986. Learning internal representations by error propagation. USA: MIT Press Cambridge.