International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 12, Number 12 (2019), pp. 2870-2873

Training and Prediction of PUs Applying RNAs in Wireless Cognitive Radio Networks

Danilo Alfonso López Sarmiento
Universidad Distrital Francisco José de Caldas
Faculty of Engineering, Bogotá, Colombia.

ORCID: 0000-0002-6148-3099

Nelson Enrique Vera Parra
Universidad Distrital Francisco José de Caldas
Faculty of Engineering, Bogotá, Colombia.

Octavio José Salcedo Parra
Universidad Distrital Francisco José de Caldas
Faculty of Engineering, Bogotá, Colombia.

Abstract:

An important variable to implement the concept of dynamic spectrum management with Cognitive Radio (CR) is related to the correctness of the algorithms to estimate or predict the behavior of the primary user (PU) in the channel; if the success rate is high, the number of collisions between the primary user and the secondary user (SU) can be low by raising the system performance; if the success rate is low, interruptions in the transmission of PU data can be generated, a condition that is not acceptable in the world of wireless telecommunications. This article discusses the possible advantages and / or disadvantages of characterizing PUs using Artificial Neural Networks (ANNs).

Keywords: Primary user, Artificial neural networks, Cognitive radio, Characterization of PUs.

1. INTRODUCTION

ANNs are computational models that emerged as an attempt to achieve mathematical formalizations about the structure of the brain. These mimic the structure of the nervous system, focusing on the functioning of the human brain, based on learning through experience, with the consequent extraction of knowledge from it. An Artificial Neural Network can be considered as a mathematical model of theories of mental and cerebral activities, based on the exploitation of local processing in parallel and the properties of distributed representation [1].

These types of structures, due to their ability to learn, have the ability to model, predict and classify periodic time series; This paper does not preclude evaluating its condition to estimate chaotic series such as those found when characterizing licensed users (PUs) in wireless cognitive radio networks (CRNs) [2]; and where CRNs are a paradigm that aims to optimize the use of the electromagnetic spectrum allowing unlicensed users to opportunistically use the radio spectrum in an opportunistic manner, without affecting the communications of licensed users [3].

2. DESIGN OF THE NEURONAL NETWORK IN MATLAB.

The implemented neural network model is the one shown in the diagram in Figure 1.

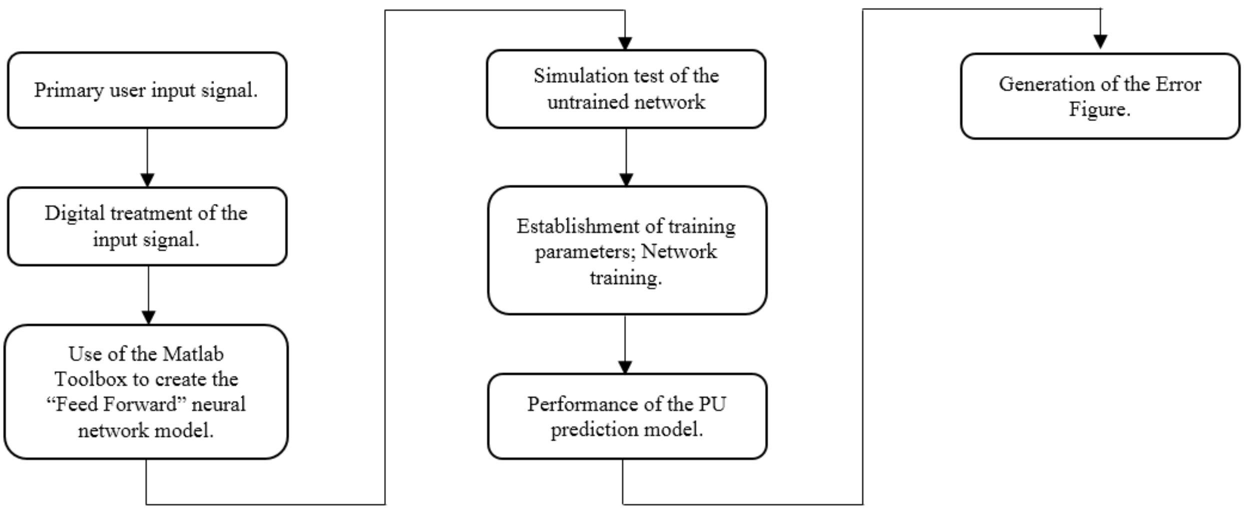


Figure 1. Scheme of the prediction model of PUs in WiFi with artificial neural networks.

International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 12, Number 12 (2019), pp. 2870-2873

2.1. Model Description.

The first thing it does is to receive the input data from the PU, these are stored in a database, the aspect of which is described in Figure 2.

E1 =AVERAGE(A1:C1)

	A	B	C	D	E
1	-95,593338	-96,763824	-104,888016		-99,081726
2	-88,880859	-93,108475	-93,444763		-91,8113657
3	-93,393761	-94,169952	-105,450821		-97,6715113
4	-105,492752	-104,074631	-93,503723		-101,023702
5	-90,379135	-95,767059	-97,903625		-94,683273
6	-109,086472	-94,597824	-106,234016		-103,306104
7	-89,1763	-95,69886	-90,530212		-91,8017907
8	-95,15271	-95,850258	-94,270752		-95,09124
9	-103,740204	-92,53614	-99,580093		-98,6188123
10	-99,216263	-95,329651	-99,093506		-97,8798067
11	-98,256218	-97,161346	-100,749809		-98,7224577
12	-102,272583	-104,281631	-101,080544		-102,544919
13	-96,024277	-97,767433	-104,629845		-99,4738517
14	-97,75737	-100,156021	-91,52858		-96,480657

Figure 2. Averaged power levels (dB) of the PUs in different spectral bands in WiFi.

Once this procedure has been performed, the algorithm establishes the amount of data that will be used in training and in the neural network test (see Figure 3). Once the sample is obtained, it is normalized to minimize the variation between the data so that the data is in a range between 0 and 1, considering the maximum value and the minimum value that were obtained from all of the data. records taken by the devices. This means that both inputs and outputs are within an acceptable length range and are dispersed within that range for easier data processing [4].

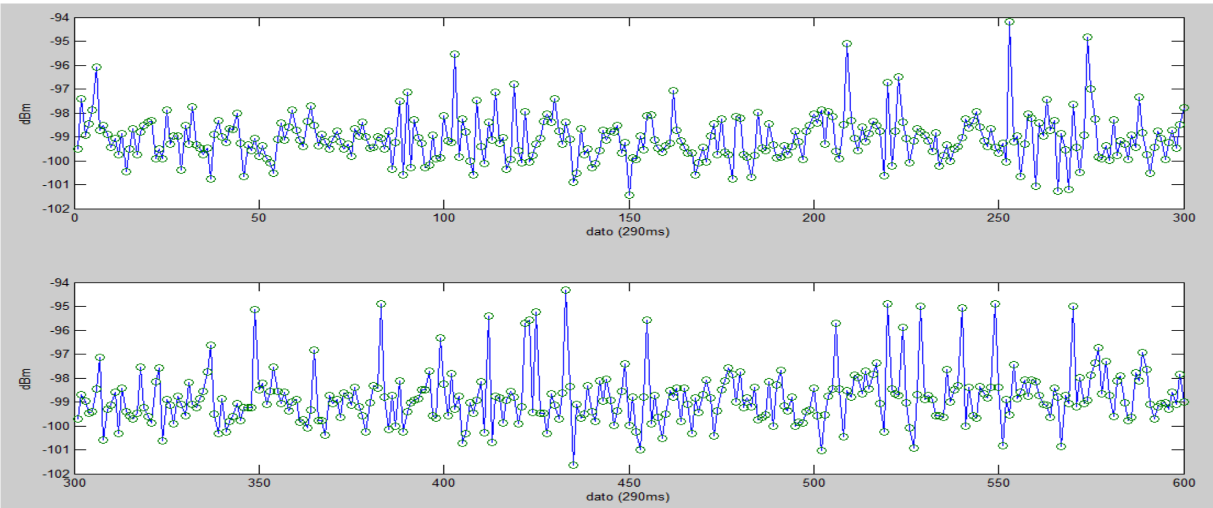


Figure 4. Sample data for training and prediction in the neural network.

The method of working with normalization is as follows (See Figure 5):

Figure 5. Data normalization method used [4].

For the normalization process each data was taken and the equation described below was applied:

$$y_t = \frac{y_t - y_{tmin}}{y_{tmax} - y_{tmin}} \qquad (1)$$

Where y_t it represents the data at the time t of the sample that is processed, , y_{tmin} and y_{tmax} represents the lowest and highest overall respectively of the total data obtained in the collection.

Having the PU data processed properly, a four-layer neural feed forward network [5] is created, with 50 neurons (Figure 6) per layer, with TANSIG and PURELI activation function. The training method used was backpropagation.

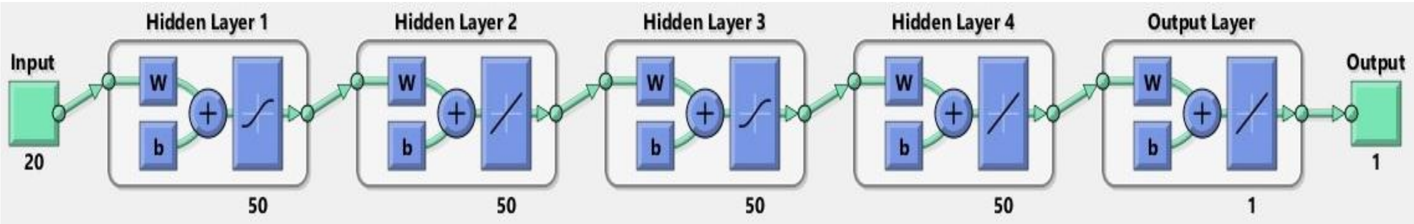


Figure 6. Neural network structure generated in Matlab.

In Figure 6, it can be seen how the training stage begins with the feeding of the network. Then the inputs are fed; these entries are sent to the computational or hidden layers as shown. For Figure 6 there is a total of 200 neural networks, 50 for each hidden layer. After each iteration in a calculation process, the error is said to have been generated. This error spreads back into the input layer of the hidden layers of the neural network. The learning process is generated by minimizing this error after each successive iteration. The output of the hidden layer is sent to the output layer, where the final computation and the visualization of the final prediction result or success in the future estimation of the PU in the WiFi band occurs.

The creation of the "Feed Forward" neural network in Matlab is generated using the Matlab command "newf" using the activation functions 'tansig' and 'purelin':

```
%Creation of the neural network
net = newff(Xecg,Yecg,[50 50 50 50],{'tansig' 'purelin' 'tansig' 'purelin'}
```

Then the training parameters are established, in this case 125 training times (used by the backpropagation algorithm), the network is trained, simulated and the trained network is displayed; finally, the data is compared and the figure of the mean square error is generated:

```
%Training parameters
net.trainParam.epochs = 125;
net.trainParam.goal = 0.02;

%Network training
net = train(net,Xpus,Ypus);

%Simulation of the training network
y = sim(net,Xpus);
figure
plot(t,Ypus,t,y)
title(['Simulation of the trained network' ],'fontsize',12);
xlabel('data','fontsize',10);
ylabel('dBm');
legend('Real', 'Simulated');
```

3. ANALYSIS OF RESULTS.

For the test presented in this article, 500 data representing the behavior of a PU in a WiFi channel are used, in which 70% of the information is used for training, and the remaining 30% of them for prediction (Figure 7); it is important to indicate that the axis of the abscissa indicates the amount of data used to represent the PU, and the axis of the ordinates the range of data in which they oscillate (dBm).

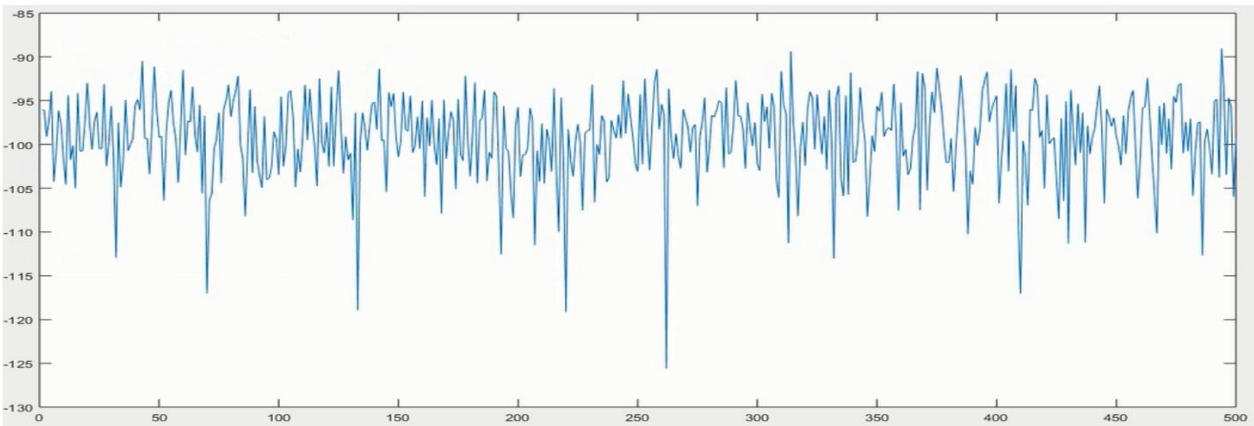


Figure 7. Training data and prediction of PUs using ANN.

International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 12, Number 12 (2019), pp. 2870-2873

In Figure 8, the behavior of the ANN is observed when the algorithm is not trained, it is clear that the response is completely arbitrary.

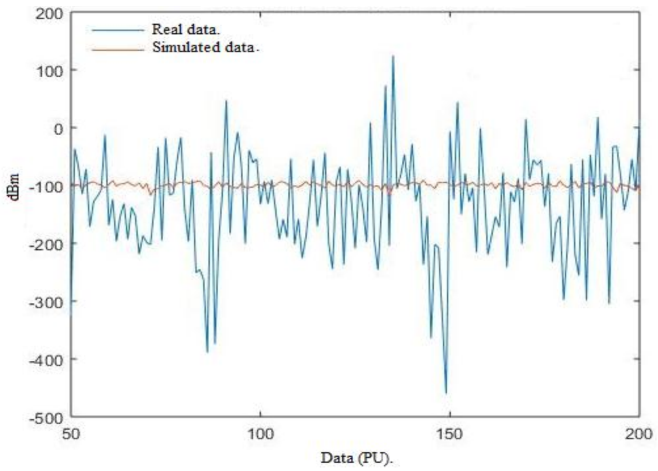


Figure 8. ANN behavior in the prediction stage when the network is not trained.

When the ANN is trained, the response of the system is that described in Figure 9; here it is noted that although the modeling stage of PU is not very successful, it significantly improves the results observed in Figure 8.

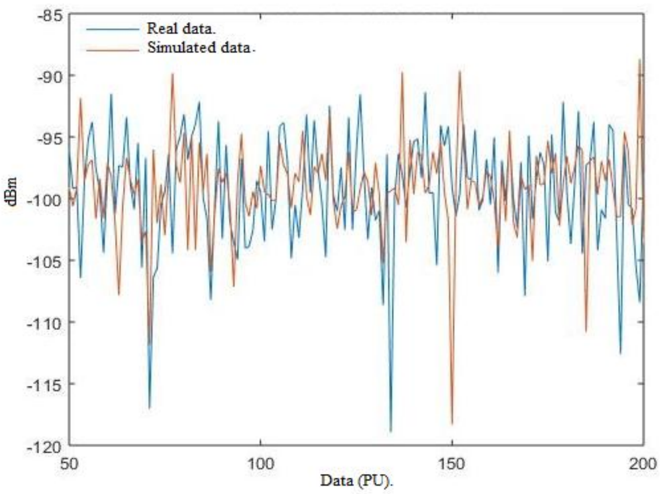
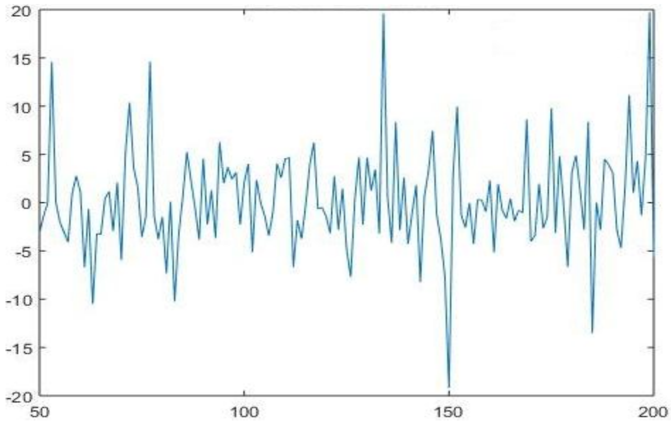


Figure 9. ANN behavior in the prediction stage when the network is trained.

When evaluating the characterization of PU (modeling + prediction), the results of Figure 9 suggest that the level of success is below 50%, a condition that is corroborated by illustrating the graph of the error (Figure 11) between the percentage of Successful prediction of PU and real behavior in the WiFi spectral band.



4. CONCLUSIONS.

From the simulations developed, it is observed that although the behavior in the prediction of the PUs could be monitored through ANN by adjusting the number of neurons and layers necessary to further reduce the Average Error, it is concluded that learning can become slow due to the increase of each of these parameters, in addition to the increase in the number of training times. This condition allows to infer that generally with ANN, better results can be achieved when the input signal has a periodic behavior.

5. REFERENCES.

[1] Raquel Florez, José Miguel Fernández; Las redes neuronales artificiales, Editorial Netbiblo, 2018.

[2] Danilo López, Edwin Rivas, Oscar Gualdron, Primary user characterization for cognitive radio wireless networks using a neural system based on deep learning, Journal Artificial Intelligent Review, Vol. 52, Pp. 169-195, 2019, doi: 10.1007/s10462-017-9600-4.

[3] Johana Hernández, Danilo López, Nelson Vera, Primary user characterization for cognitive radio wireless networks using Long Short-Term Memory, International Journal of Distributed Sensor Networks, Vol. 14, Issue 11, Pp. 1-20, 2018, doi: 10.1177/1550147718811828.

[4] Zarza Cano, Estefano Alfredo, Desarrollo de redes neuronales artificiales para el cálculo del transporte neoclásico en reactores de fusión, Proyecto Fin Carrera, Departamento de Física, Universidad Carlos III, pp. 1–93, 2012.

[5] Tin-Yau Kwok, Dit-Yan Yeung. Constructive algorithms for structure learning in feedforward neural networks for regression problems. IEEE Transactions on Neural Networks, Vol 8, Issue 3, 1997.