Proceedings of the 4th International IEEE EMBS Conference on Neural Engineering Antalya, Turkey, April 29 - May 2, 2009

A Type 2 Neuron Model for Classification and Regression Problems

Mehmet Önder Efe
Department of Electrical and Electronics Engineering
TOBB Economics and Technology University
Ankara, Turkey
onderefe@etu.edu.tr

***Abstract*— Type 2 fuzzy systems have been under investigation for a while and the projection of type 2 understanding for uncertainty management onto the connectionist models –i.e. neural networks- seems an interesting field of research. This paper considers neurons having multiple bias values defining a new structure that resembles the uncertainty handling capability of type 2 fuzzy models. Such a neuron provides many activation levels that are combined to obtain the neuron response. A neural network with this new model is presented. Several simulation results are shown and the universal approximation property is emphasized.**

Keywords-type 2 neuron model, type 2 neural networks

I. INTRODUCTION

A multilayer perceptron network with error backpropagation was a standard setting for a long time. Research towards improving this configuration has two major lines, first is the changing of the training scheme for better performance, e.g. Levenberg-Marquardt algorithm, terminal attractor based algorithms or those based on sliding mode learning, [1-3]. The other direction is to change the structure, reinterpreting the biology, assuming a different scheme of connections might result in better performance. This paper considers the second line for obtaining similar performance with lower computational cost.

In the literature, several studies reporting new neuron models have appeared. Among those, Scott discusses the construction of McCulloch neuron via utilizing the tools of linear programming, [1]. An optimal design is presented with the models having at most 3 inputs. A survey of artificial neuron models is presented in [2], where the models having tunable activation functions with a preselected type of neuronal activation function, or changeable activation functions with tunable parameters are reported together with models fuzzy neuron models, quantum neuron models and those based on chaos systems. In [3], Yadav *et al* present a generalized mean type activation performing a net sum computation according to $S=\Sigma_j(w_j x_j^r+\theta)^{1/r}$ where the inputs are powered up to n adjustable level of generalization parameter. Classification and function approximation examples are presented and superiority of the approach is emphasized. Use of logical operators to build logical separators is achieved by a multi valued logic neuron model discussed in [4], defining multiple dendritic activity and aggregation of these is presented in [5] and wavelet based neuronal structures are elaborated in [6], all having the goal of improving the capabilities of classical neuron model. Another model utilizing the residue reduction is discussed in [7] within the context of classification and random search method is adopted as the training scheme. In [8], Karaköse and Akın emphasize that a conventional activation function could be realized by an embedded type 2 fuzzy model, so a neural model exploiting type 2 formalism indirectly could help to some extent. This study differs from [8] in the implementation of activation functions. Current work directly describes the neuronal activation functions as type 2 entities whereas these functions are described by type 2 membership functions in [8].

Clearly, the efforts toward obtaining a versatile model is a continuing, and the results presented here extend the subject area to the realm of type 2 fuzzy systems, [9-10], and their capabilities in representing and manipulating vagueness hidden in the reality. This paper is organized as follows: The second section introduces the type 2 neuron model, the third section describes the data sets used for comparison, the fourth section presents a comparison of the performances and finally the concluding remarks are given at the end of the paper.

II. TYPE 2 NEURON MODEL

Consider a type 1 neuron model admitting the net sum S given by (1) as the input to the neuronal activation function. The inputs are denoted by u_js and the associated weights are w_js. Typically a bias value (θ) is added to the net sum (See (2)) to obtain a translation along the horizontal axis and this enables to realize maps that do not pass through the origin, i.e. $y \neq 0$ for $S = 0$.

$$S = \sum_{j=1}^{m} w_j u_j \tag{1}$$

$$y = \tanh(S+\theta) \tag{2}$$

A number of such neurons connected in a networked fashion constitute a neural network as shown in Figure 1. Typical feedforward computation for a single (linear) output, single hidden layer (having h_c neurons) neural network structure like the one depicted above can be computed as follows:

$$o_i = \tanh\left(\sum_{j=1}^{m+1} w_{ij}^1 u_j\right),\ i=1,\ldots,h_c \text{ and } u_{m+1}=1 \tag{3}$$

$$y_c = \sum_{j=1}^{h_c+1} w_{1j}^2 o_j, \; o_{h_c+1} = 1 \quad (4)$$

where w_{ij}^1 corresponds to the weight matrix on the left, which is the hidden layer's weight matrix, and w_{1j}^2 .denotes the entries of the output layer weight matrix. The output of this classical neural structure is denoted by y_c. Total number of adjustable parameters in this setting is C_p==$(m+2)h_c+1$.

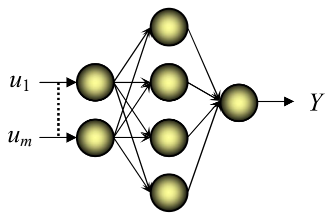

Figure 1. *m*-input single-output neural network structure

Expressing the model of a neuron in type 2 formalism would produce the following excitation picture.

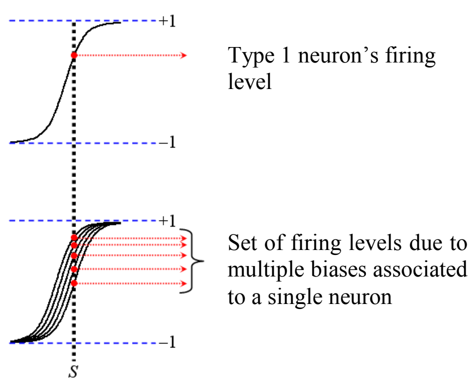

Figure 2. Activation scheme of a type 2 neuron compared to type 1 neuron. There are *n* different activation functions within a single type 2 neuron.

According to the excitation scheme depicted in Figure 2, a single neuron fires at multiple levels determined by the value of bias values. To obtain the response of the type 2 neuron, y_p, the aggregation of the computed activation levels is performed as given in (5).

$$S = \sum_{j=1}^{m} w_j u_j \quad (5)$$

where no biasing is done at this stage, instead, the scheme in (6) is followed to obtain multiple levels of activations (z_i) out of a single type 2 neuron.

$$z_i = \tanh(S + \theta_i), \; i = 1, \ldots, n \quad (6)$$

$$o = \sum_{j=1}^{n+1} \phi_j z_j, \; z_{n+1} = 1 \quad (7)$$

where o denotes the output of the neuron. Total number of adjustable parameters of a type 2 neuron is equal to $m+2n+1$, where n is the number of different bias values.

Now we are ready to define the type 2 neural network based on the type 2 neuron model introduced above. We consider that there are h_p hidden type 2 neurons and their outputs are summed up to compute the network response. More explicitly,

$$S_i = \sum_{j=1}^{m} w_{ij} u_j, \; i = 1, \ldots, h_p \quad (8)$$

$$z_k = \tanh\left(S_i + \theta_k\right), \; k = 1, \ldots, n, \; i = 1, \ldots, h_p \quad (9)$$

$$o_i = \sum_{j=1}^{n} \phi_{ij} z_j \quad (10)$$

$$y_p = \Theta + \sum_{i=1}^{h_p} o_i \quad (11)$$

where the adjustable parameters are w_{ij}, θ_k, ϕ_{ij}, Θ, the total number of which add up to $C_p=(m+2n)h_p+1$.

The potential advantage of constructing such a model is to obtain a distribution corresponding to a particular excitation (S) and to convert it into an output by utilizing a set of adjustable weights, also peculiar to each specific neuron.

III. Considered Data Sets

Six sets of data have been considered. These are explained below.

A. XOR Data

The XOR problem is always the first step of discussing the performance associated to a novel neural network approach. We consider this data set as a basic classification approach distinguishing the two different logic levels and interpolating in between.

B. Iris Data from UCI Repository

The problem here is to distinguish classes of Iris plant according to sepal length, sepal width, petal length and petal width, all in centimeters. The response of the neural classifier is one of the three labels Iris Setosa, Iris Versicolour or Iris Virginica. In the training we assign -1 for the label Iris Setosa, 0 for the label Iris Versicolour and +1 for the label Iris Virginica and convert the classification problem into a tractable regression problem in between the three integers. Thresholding is possible after the training is finished, yet, the performance of the classifier is directly proportional to final RMSE value achieved. In the UCI repository, 150 instances are provided. The training data has been selected in such a way that every class contributes 40 instances, adding up to 120 instances for three classes, and the remaining 30 instances have been used as testing data set, [11-12].

C. Subsonic Cavity Flow

Due to the infinite dimensional mathematical representation and spatially continuous nature, processes governed by partial differential equations display several difficulties. Particularly for the aerodynamic flow problems, the process is governed by Navier-Stokes equations, which are inextricably intertwined and difficult to handle in most cases. In [13-15], a flow modeling and control system is introduced.

The main goal was to describe a dynamic model predicting the pressures at the floor of the cavity (See Figure 3(a)-(c)). These locations include the signal to the host computer (x_1), the output of the actuator (x_2), the measurement at the receptivity point (x_3), the measurement at cavity trailing edge (x_5) and the measurements before and after the rectangular cutout (x_4, x_7) are also depicted in Figure 3(a). Conventional approach to model such systems utilize proper orthogonal decomposition with the method of snapshots, singular value decomposition and so on, however, these methods are very likely to yield a very complicated model composed of many ordinary differential equations having quadratic nonlinearities and drifts in the solutions. Since they work on the numerical data, neural models are good alternatives to build local models, e.g. at the cavity floor (x_6), of the entire process.

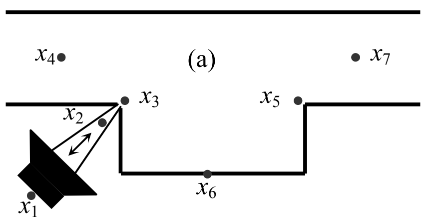

Figure 3. Schematic view and sensor locations on the setup

The experimental setup is composed of a host computer having a digital signal processor, a filter to remove the spurious flow content at high frequencies, a Titanium diaphragm synthetic actuator and an air system providing the flow at a desired Mach number. Since the spectral view is rich at Mach 0.25, we consider the modeling problem based on data obtained under the conditions mentioned above, [12]. The training and testing data sets have 250 and 100 pairs, respectively. The data have been collected at a sampling frequency of 50 kHz. The data have been obtained with a sinusoidal excitation at x_1 with frequency 3920 Hz and magnitude 4.06V. The neural network based modeling problem is to develop the function $f(\cdot)$ in (12) such that the response matches the real time data precisely and minimum amount of sensory information is exploited.

$$y_p = x_6^{k+1} = f(x_1^k, x_3^k, x_5^k, x_6^k, x_6^{k-1}) \qquad (12)$$

The function $f(\cdot)$ in (12) is highly nonlinear map, which cannot be obtained unless the approaches exploiting the input-output numerical data are utilized.

D. A Biochemical Benchmarking Process for Modeling and Control

In [16-19], the biochemical benchmark problem is defined as one of the challenging control problems displaying fast and highly nonlinear response with particular state and input constraints. The process is composed of a tank containing water, cells and nutrients denoted by c_1 and c_2, respectively. The constraints characterizing the motion are given by the following equations.

$$\dot{c}_1 = -c_1 u + c_1 (1 - c_2) \exp\left(\frac{c_2}{0.48}\right),\ 0 < c_1 < 1,\ 0 \le u \le 2 \qquad (13)$$

$$\dot{c}_2 = -c_2 u + c_1 (1 - c_2) \frac{1.02}{1.02 - c_2} \exp\left(\frac{c_2}{0.48}\right),\ 0 < c_2 < 1 \qquad (14)$$

The regression problem for this process has been split into two sub problems, one for the cell dynamics, the other for the nutrient dynamics. The regression problem in both is to predict the next value of the asked state variable ($c_i(t+1)$) given the current values of the current state variables ($c_1(t)$, $c_2(t)$) and the current input, $u(t)$. For randomly selected 250 initial states and input values, the next values are computed by utilizing the first order Euler discretization of the process equations with a sampling period of 0.01 sec.

E. Circle Data

This dataset is another standard set having two inputs denoted by $(x,y) \in [-1,+1] \times [-1,+1]$. The output is +1 if $x^2+y^2<0.25$ and −1 otherwise. A grid with resolution 0.1 in both directions is selected and training is carried out with 441 pairs of data with randomly selected another 100 is considered as the test set.

IV. PERFORMANCE COMPARISON STUDIES

In order to carry out a fair comparison, the only parameter that changes in between the classical and proposed models must be the structure. In other words, in comparing the performance of two structures, one has to maintain the equal number of adjustable parameters. To ensure this, we assume the following procedure: Since $C_c=(m+2)h_c+1$ and $C_p=(m+2n)h_p+1$, we set the number of biases, n, freely, and choose the number of hidden neurons in both networks according to $h_p=m+2$ and $h_c=m+2n$, so that we obtain $C_c=C_p$.

Once a dataset is chosen and the type 1 network is setup. All adjustable are initialized randomly from [−1,+1] and 1000 epochs of training is carried out. The same experiment is repeated 1000 times, yielding a 1000×1000 matrix of epoch error (also called the Mean Squared Error (MSE)) evolution for each experiment. Upon completion of this, type 2 network is initialized similarly and the same set of experiments is carried out. Next, the dataset is changed and the same data collection is repeated until the all six sets of classification/ regression problems are covered. During the experiments, we considered the Levenberg-Marquardt training scheme for parameter adjustment. We utilized an adaptive step size algorithm adjusting the balance between gradient descent and Newton method, and all training has been implemented on a Pentium IV computer. Aside from the completion of 1000 epochs, as a stopping criterion, we adopt a checking routine stop if the checking error was increasing for the latest 5 epochs. The final value of the epoch error in each particular experiment is considered as the ultimate level of performance of that particular training trial. A 100 bin histogram of the final MSE levels is build and the results are shown in Figure 4, where the horizontal axis is the Final MSE level and the vertical axis is the frequency of occurrence of a result out of the set of 1000 experiments.

According to the figure, it is seen that the XOR problem ends up with a final MSE level of order 10^{-30}. The top left subplot of the figure depicts the results of the XOR case and the mean of the distributions are also plotted with an ○ sign along the horizontal axis. Due to the negligible small values of the means, we consider both methods as equally successful. For the three-class Iris recognition problem, although the means of the two distributions almost coincide, the proposed technique is very slightly better as its curve is below the curve of classical approach. For the flow identification problem, which is a regression problem, classical approach results is slightly preferable. Predicting the cell mass (c_1) of the biochemical process model is depicted in the middle right subplot of the figure. The proposed technique performs better than the classical NN structure. When the nutrient amount behavior is taken into consideration, (bottom right subplot) almost the same mean values are obtained and both methods perform equally acceptable, yet the proposed technique displays smaller variance indicating better consistency. Regarding the circle dataset, the means of the two distributions are seen to be very close and the proposed technique has a larger variance.

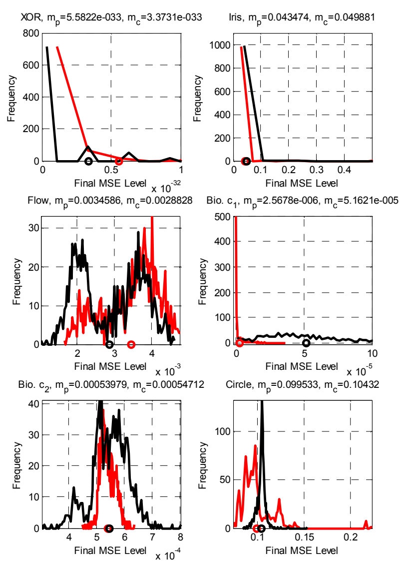

Figure 4. A comparison of the classical (type-1, black) and proposed (type-2, red) approaches. The mean of the distributions are marked on the horizontal axis and m_p denotes the mean of the distribution of the curve with the proposed technique, and m_c is that for the classical approach.

V. CONCLUSIONS

The classical feedforward NN structure is based on the computation of many net sums at the hidden layer with single biasing then linear (or nonlinear) aggregation of the hidden layer outputs, whereas, this paper describes a single neuron that is based on single net sum augmented with multiple biases and aggregation at the output. The proposed form of the neuron is more similar to the type 2 context of the fuzzy systems and is therefore named type 2 neuron. This form of neuron structure is simpler than the classical network structures and is able to yield better performance levels with better consistency, measured by the variance of the associated distribution.

ACKNOWLEDGMENTS

This work is supported by Turkish Scientific Council (TUBITAK) Contract 107E137.

REFERENCES

[1] R.J. Scott, “Construction of a Neuron Model” *IRE Trans. on Bio Medical Electronics*, July, pp.198-202, 1961.

[2] J. Lv, C. Guo, Z.-P. Shen, M. Zhao and Y. Zhang, "Summary of Artificial Neuron Model Research" The 33rd Annual Conference of the IEEE Industrial Electronics Society (IECON), Nov. 5-8, 2007, Taipei, Taiwan, pp.677-682.

[3] R.N.Yadav, N. Kumar, P.K. Kaka and J. John, "Multi-Layer Neural Networks Using Generalized-Mean Neuron Model" International Symposium on Communications and Infomation Technologies 2004 (ISCIT 2004) Sapporo, Japan, October 26- 29,2004, pp.93-97.

[4] T. Watanabe , M. Matsumoto , M. Enokida and T. Hasegawa, “A Design of Multiple - valued Logic Neuron” pp.418-425, 1990.

[5] S. Li, Y. Chen, E.L. Leiss, "GMDP: a Novel Unified Neuron Model for Multilayer Feedforward Neural Networks" pp.107-112, 1994.

[6] M.F. Azeem, A. Banakar, V. Kumar, "Comparative Study of Different Types of Wavelet Functions in Neural Network" 2006 Int. J. Conf. on Neural Networks, Vancouver, Canada July 16-21, 2006, pp.1061-1066.

[7] Y. Tianran, W. Xiaoming, S. Hong, "A Novel Neuron Model and Its Application to Classification" pp.1351-1354, 1994.

[8] Karaköse, M. and Akın, E. Type-2 fuzzy activation function for multilayer feedforward neural networks. IEEE Int. Conf. on Systems, Man and Cybernetics, v.4, 10-13 Oct. pp.3762-3767, 2004.

[9] J.M. Mendel, “Type 2 Fuzzy Sets and Systems: An Overview,” *IEEE Computational Intelligence Magazine*, pp.20-29, Feb. 2007.

[10] Ö. Uncu, İ.B. Türkşen, “Discrete Interval Type 2 Fuzzy System Models Using Uncertainty in Learning Parameters” *IEEE Trans. on Fuzzy Systems*, pp.90-106, Feb. 2007.

[11] Asuncion, A. and Newman, D.J. (2007). UCI Machine Learning Repository [http://www.ics.uci.edu/~mlearn/MLRepository.html]. Irvine, CA: University of California, School of Information and Computer Science.

[12] M.Ö. Efe, "Novel Neuronal Activation Functions for Feedforward Neural Networks," Neural Processing Letters, v.28, no.2, pp.63-79, October 2008.

[13] Efe M.Ö., Debiasi M., Yan P., Özbay H. and Samimy M. Neural network based generalization and modeling of subsonic cavity flow regimes. International Journal of Systems Science, v.39, n.2, pp.105-117, 2008.

[14] Samimy M., Debiasi M., Caraballo E., Malone J., Little J., Özbay H., Efe M.Ö., Yan P., Yuan X., DeBonis J., Myatt J.H. and Camphouse R.C. Exploring strategies for closed-loop cavity flow control. 42nd AIAA Aerospace Sciences Meeting and Exhibit, January 5-8, Reno, Nevada, U.S.A., 2004, (Paper # AIAA 2004-0576).

[15] Debiasi M. and Samimy M. Logic-based active control of subsonic cavity-flow resonance. AIAA Journal, v.42, pp.1901–1909, 2004.

[16] L.H. Ungar, "A Bioreactor Benchmark for Adaptive-Network Based Process Control," Neural Networks for Control, W.T. Miller III, R.S. Sutton, P.J. Werbos, Eds, MIT Press, pp.387-402, 1990.

[17] C.W. Anderson and W.T. Miller III "Challenging Control Problems," Neural Networks for Control, W.T. Miller III, R.S. Sutton, P.J. Werbos, Eds, MIT Press, pp.475-510, 1990.

[18] G.V. Puskorius and L.A. Feldkamp, "Neurocontrol of Nonlinear Dynamical Systems with Kalman Filter Trained Recurrent Networks," *IEEE Trans. on Neural Networks*, v.5, n.2, pp.279-297, 1990.

[19] M.Ö. Efe, E. Abadoglu and O. Kaynak, "A Novel Analysis and Design of a Neural Network Assisted Nonlinear Controller for a Bioreactor," *Int. Journal of Robust and Nonlinear Control*, v.9, no.11, pp.799-815, 1999.