International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-7, Issue-5S4, February 2019

Rule Based Experiment on Exponential Integrate and Fire Neuron Model

Vishnu Kumar, Ajeet Kumar Verma, Rajesh Dwivedi, Ebenezer Jangam

ABSTRACT--- Classification is a technique to deal with supervised learning of Artificial Neural Networks. In recent years, many methods are developed for classification. Conventional neurons are less efficient in classification accuracy. Spiking neuron is third generation neuron. Spiking neuron models are generating highly computationally accurate firing patterns of spikes. These spikes are using to process the information in human brain. So a novel learning rule is proposed for an Exponential Integrate and Fire Neuron Model. This model is used for Malaria disease prediction. We have collected dataset for malaria from govt. ID hospital, Goa. By using proposed classifier, we have obtained increased accuracy in classification of the data. Our classification results are better when compared with legacy model and Biological Neuron Model.

Keywords: IFN, MLP, EIFN, FFNN, H-H, QIFN, Learning rule LEIFN.

1. INTRODUCTION

Over the past decades, study of human brain and its mathematical modeling is a keen interest of scientists [1]. A Neuron is basic fundamental unit to understand the functionality of human brain [3]. A biological neuron has accumulated the detailed knowledge about the functioning and the structure of brain. There are many neuron model are developed such as MLP, FFNN, Back Propagation as conventional neurons [10][2]. These MLP, FFNN and Back Propagation etc. come under the second generation of Neuron Models. But these models are computationally poor and complex [2]. Therefore we need a neuron model which has high accuracy of computation and simple to understand. Thus spiking neuron models were developed to reduce computational complexity and to remove this drawback. These models produce very rich firing patterns as exhibited by a real biological neuron [3]. Various spiking neuron models are presented in earlier research such as Hodgkin–Huxley (H-H Model) type models, Integrate and Fire Neuron (IFN) model, Quadratic Integrate and Fire Neuron (QIFN) model [5] and Izhikevich model [3] Exponential Integrate and Fire Neuron [1]. Integrate and Fire Neuron (IFN) [4] model comes under family of spiking neuron models. Spiking neuron is the third generation of neuron models. IFN is the simplest spiking neuron model and is alone sufficient to give high computational performance as compare to conventional neuron model. This spiking neuron takes current signal as an input and produces spikes when the membrane potential crosses threshold level. Information is processed in terms of firing patterns of spikes. IFN is computationally effective and produces rich firing patterns. It is very simple model of biological neuron and efficient enough to process complex information. There are 20 types of cortical spikes that are generated through these spiking neurons. A biological neuron is consisting of three major parts soma, axon and dendrites. An input signal to a neuron is a short electrical pulse that can be observed by electrode near to the soma or axon of a neuron. Dendrites receives signal from the another neurons. Axon is like fine strands fiber which carries inputs to the soma or cell body. Soma is used for processing inside the neuron shown in Fig.1 [4].

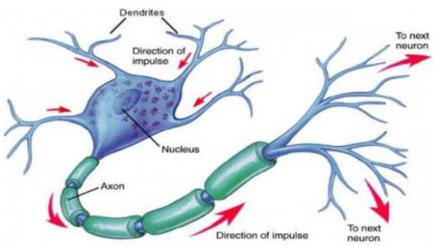

Figure 1: Biological Neuron Structure

Generated action potential or spikes have an amplitude potential of approximately 70mV and time duration of 1.0-2.2ms. The chain of spikes is fired by a biological neuron is called spike train.

Based on the type of sequence of spikes that can be regular or irregular, various cortical spiking firing pattern has be described. The membrane potential is integrated up to a threshold level when crossed threshold level, a spike is generated by neuron and just after this spike; the membrane potential is reset to zero shown in Fig.2. The next spike is not produced immediately after first one. The time duration between two successive spikes is called is inter- spike interval. The minimal time gap between two spikes is called as the refractory period.

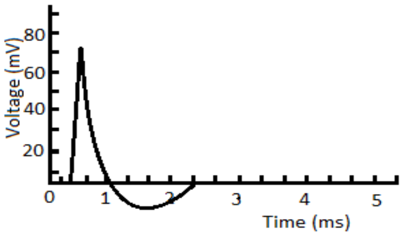

Figure 2: Spike Generation model

Revised Manuscript Received on February 11 , 2019.

Vishnu Kumar, Department of Computer Science and Engineering, Vignan Foundation For Science Technology and Research, Guntur, India. (vksingh843025@gmail.com)

Ajeet Kumar Verma, National Institute of Technology, Goa, India.

Rajesh Dwivedi, Department of Computer Science and Engineering, Vignan Foundation For Science Technology and Research, Guntur, India.

Ebenezer Jangam, Department of Computer Science and Engineering, Vignan Foundation For Science Technology and Research, Guntur, India.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

2. OVERVIEW OF EXISTING METHODS

Here this section is subdivided into three parts. Integrate and Fire Neuron model is discussed in section 2.1, along with the spike generating patterns and mathematical model is described. In section 2.2, the Exponential Integrate and Fire Neuron model is discussed along with the mathematical formula to exponential model and in the section 2.3, learning mechanism is explained and circuit diagram of the model is shown in Figure 3.

2.1. Integrate and Fire Neuron model

Integrate and Fire Neuron Model is the one of best representation of neurons electrical properties, hence it is widely used in neuroscience research domain. In this model, when external input current is given then the capacitor is charged up to a certain threshold and generates spikes. Just after first spike capacitor starts discharging and potential is going to reset again [9]. This model is given by following eq. (1) according to fig 3.

$$\tau_m \frac{dV}{dt} = E_L - V - R_m\, I_e \qquad (1)$$

Where τ_m is time constant and given by $\tau_m = R_m \times C_m$, R_m is resistance and C_m is the capacitance. E_L is E_{Leak} and I_e is injected current.

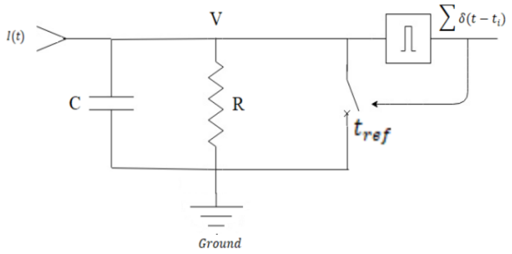

Figure 3: Membrane Circuit Structure

2.2 Exponential Integrate fire Neuron Model

The Exponential Integrate Fire Neuron Model (EIFN) [1] is realizes by the given one dimensional differential equation which is given by eq.(2):

$$C\frac{dV}{dt} = -g_L(V - V_L) + g_L\Delta_t \exp\left(\frac{V - V_t}{\Delta_t}\right) + I_{syn}(t) \qquad (2)$$

Where Δ_t and V_t are thesloping factor and spike threshold respectively. When $\Delta_t \rightarrow 0$ (representing a spike with sharp edged initiation) the EIF model starts behaving like LIF model with $V_{th} = V_t$.

2.3 Learning with S-IFN Neuron Model (S-IFN)

A training algorithm for Single IFN (single Integrate-and-Fire Neuron) is developed and validated against many applications for which generally MLP[10] and some other iterative neural network are used [4]. It has been observed that a single Integrate-and-Fire Neuron model is alone capable for all such applications that need a huge network made up of so many hidden layers having large number of neurons [2]. The basic circuit for this IFN model contains resistor R in parallel withcapacitor C is constructed with a external current I_{EXT} and capacitor C, and representation of circuit diagram of IFN model is shown in Figure 2. According to circuit diagram the driven current is divided into two parts, thus $I_{EXT} = I_R(t) + I_C(t)$ in this, the early first part $I_R(t)$ is the preventive current, it travels through resistor R and the other part is using to charge the capacitor C and to solve this Kirchhoff's Current Law (KCL) is applied. This learning rule for IAF model is influenced from the association between ISI time interval and injected external input current for an IFN model, following multiplicative aggregation function is accepted at place of the aggregated weighted sum of a regular neuron [4].

$$net = \prod_{i=1}^{p} (a_i \log(b_i * x_i) + d_i) \qquad (3)$$

Where, p is the total count of records in dataset. x_i is the input for neuron. a_i, b_i, d_i are the function parameters.

3. PROPOSED EXPONENTIAL SPIKING NEURON MODEL

This section is subdivided into three parts. First subsection 3.1 introduces Inter spike interval along with the mathematical formulation. Subsection 3.2 explains about proposed learning rule for Exponential Integrate and Fire Neuron Model and in section 3.3, the rules novelty is given be mathematical derivation of the proposed training and the formulation is given for computation (training) of EIFN model.

3.1 Exponential Inter spike Interval

Inter spike interval responsible for communication among neurons. Inter spike interval is obtained from the solution of Exponential Integrate Fire Neuron Model. Model is designed according to circuit diagram shown in Fig.3. Exponential Integrate and Fire Neuron Model represented in Eq. (2) cannot be solved mathematically [8]. To reduce this mathematical complexity, we add some slow adaptation current to solve this complex mathematical model. Inclusion of adaptation current allowed us to reduce the first part of equation where $-g_L(V - V_L)$ solved and equation is reduced to Equation 2 [8].

After leaving the term $-g_L(V - V_L)$ and adding the slow adaptation current $(-I_L)$ in Eq. (10) we get Eq. (11), which is given by

$$C\frac{dV}{dt} = -I_L + g_L\Delta_t \exp\left(\frac{V - V_t}{\Delta_t}\right) + I_{syn}(t) + I_{ext} \qquad (4)$$

The zero order hold solution for Eq. (4) is given by

$$V(t) = V_T + \Delta_T \ln\left(\frac{I_{ext}}{g_L\Delta_t}\left(I_{ext} - \frac{1}{1 - \exp^{\left(\frac{I_{ext}(t - t_0)}{C\Delta_T}\right)}}\right.\right. \qquad (5)$$

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Here we are assuming that $V(t)$ reaches V_{TH} at $t = T_{thres}$. Thus

$$V_{TH} = V_T + \Delta_T \ln\left(\frac{I_{ext}}{g_L \Delta_t}\left(I_{ext} - \frac{1}{1 - \exp\left(\frac{I_{ext}(T_{thres} - t_0)}{C\Delta_T}\right)}\right)\right) \quad (6)$$

Here Inter spike time interval T_{ISI} is the aggregated sum of T_{thres} and T_{REF}

$$T_{ISI} = T_{thres} + T_{REF}$$

3.2 Proposed learning rule for EIFN Model

Neurons exchange information in terms of Interspike interval. Inspired from the association between ISI time interval and injected external input current for an Exponential IFN model shown in Eq.(8), following multiplicative aggregation function is proposed at place of the aggregated weighted sum of a regular neuron.

$$NET = \prod_{i=1}^{m}\left\{a_i \log\left(\frac{k_i * e^{b_i * x_i - H_i}}{k_i + e^{b_i * x_i - H_i}}\right) + d_i\right\} \quad (8)$$

Where, m is the total count of samples in dataset, $d_i = T_{REF}$, $a_i = \frac{C\Delta_T}{I_{ext}}$, $b_i = \left(\frac{V_0}{\Delta_T}\right)$, $x_i = \left(\frac{V_T}{\Delta_T}\right)$ $k_i = g_L \Delta_t$, Eq. (11) shows the multiplicative aggregation function for proposed learning with exponential neuron model (LEIFN). This NET input is given to a sigmoidal function.

$$y = \frac{1}{1 + e^{-NET}} \quad (9)$$

3.3 Training Algorithm for LEIFN

A simple steepest descent method is applied to minimize the following error function:

$$e = \frac{1}{2}(y - t)^2 \quad (10)$$

Where, y is the actual output of the neuron, t is the target and e is the error. Function parameters are $a_i, H_i, b_i, k_i \,\&\, d_i$; $i = 1,2,\dots m$. Therefore, the parameter update rule (weight update rule) can be obtained by partially derivation of the error function with respect to the parameters individually.

$$a_i^{new} = a_i^{old} - \eta \frac{\partial e}{\partial a_i} \quad (11)$$

$$b_i^{new} = b_i^{old} - \eta \frac{\partial e}{\partial b_i} \quad (12)$$

$$k_i^{new} = k_i^{old} - \eta \frac{\partial e}{\partial k_i} \quad (13)$$

$$d_i^{new} = d_i^{old} - \eta \frac{\partial e}{\partial d_i} \quad (14)$$

$$H_i^{new} = H_i^{old} - \eta \frac{\partial e}{\partial H_i} \quad (15)$$

For $i = 1,2,3,\dots n$. Partial derivatives of e with respect to parameters $a_i, H_i, b_i, k_i, \& d_i$; $i = 1,2,\dots m$ can be given by the following equations:

$$\frac{\partial e}{\partial a_i} = (t - y)(y)(1 - y)net\left(\frac{(\log(x_i) + b_i e^{H_i} - 1}{a_i \log(k_i x_i + b_i e^{H_i} - 1) + d_i}\right) \quad (16)$$

$$\frac{\partial e}{\partial b_i} = (t - y)(y)(1 - y)net\left(\frac{1}{a_i \log(k_i x_i + b_i e^{H_i} - 1) + d_i}\right)\left(\frac{a_i * e^{H_i}}{k_i x_i + b_i e^{H_i}}\right. \quad (17)$$

$$\frac{\partial e}{\partial k_i} = (t - y)(y)(1 - y)net\left(\frac{1}{a_i \log(k_i x_i + b_i e^{H_i} - 1) + d_i}\right)\left(\frac{a_i * x_i}{k_i x_i + b_i e^{H_i}}\right. \quad (18)$$

$$\frac{\partial e}{\partial H_i} = (t - y)(y)(1 - y)net\left(\frac{1}{a_i \log(k_i x_i + b_i e^{H_i} - 1) + d_i}\right)\left(\frac{a_i * b_i * e}{k_i x_i + b_i e^{H_i}}\right. \quad (19)$$

$$\frac{\partial e}{\partial d_i} = (t - y)(y)(1 - y)net\left(\frac{1}{a_i \log(k_i x_i + b_i e^{H_i} - 1) + d_i}\right) \quad (20)$$

4. DATASETS AND CLASSIFICATION RESULTS

4.1. Discussion about Malaria Dataset

Malaria data is collected from Sub District Hospital, Ponda, Goa through NIT Goa, under DST-SERB Project. The data is of duration 2007 to 2017. The Data contains the number of malaria patients visited to the hospital for their treatment along with four attributes i.e Max_temp, in_temp, Humidity and Rainfall for the week. The class information is number of patients visited from the individual village. Sample dataset is shown in Table 1. Class information contains number of patients visited in each different village of Ponda i.e. Betora and Volvoi. The sample malaria dataset is shown in table 2.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Table 1: Used Variable in the model

S. No	Parameter	Explanation
4	d_i	Constant time
5	a_i	variable associated with input x_i
6	b_i	Random variable
7	k_i	Random variable
8	x_i	Input data
9	H_i	Random variable

Table 2: Data collection for the population of deferent villages

Village Name	Population in that village
Curti Census Town	16,380
Betora	6,1013
Volvoi	1845
Dubhat	3,445
Adcolna	1,680

Malaria is mainly affected by the some specific whether conditional factors. It can be maximum temperature, minimum temperature and humidity and rainfall etc. We have considered the number of patients visited hospital on the given weather conditions average in a day.

Table 3. Attributes of datasets with target classes (no.of patient visited)

S. No	Max_temp	Min_temp	Humidity	Rainfall	No. of patients visited from Betora in Ponda city.
1	30	17	55	0	5
2	30	17	56	5	3
3	31	17	55	10	3
4	32	18	62	20	1
5	33	19	60	25	3
6	34	20	58	34	1
7	34	20	62	5	3

4.2. Classification Results using Malaria Data on proposed LEIFN

The proposed LEIFN is applied on real time Malaria Dataset. Ten cross validation is applied to obtain accuracy, sensitivity and specificity.

Performance evaluation of proposed method is discussed on malaria data. Experimental results are performed on various sizes of the malaria data. Table 3 shows testing Accuracy obtained using proposed method and other recent existing method MLP. Figure 5 and Figure 6 shows graphical representation of testing error graphs on Betora and Volvoi respectively. The relationship between error and epoch is shown in graphs given here (from a and b). From the analysis of graphs, we can conclude that behavior of error is downward with respect to time and numbers of iterations. Error is reduced in up to this range **0.00025 to 0.028**, it means our proposed learning rule is performing well enough. It is learning faster and predicting accurately about the target. Our proposed learning rule for exponential integrate and fire neuron model is alone sufficient to classify the large amount of data with 2 classes as comparison with conventional neural network such as MLP. Our dataset is having multiple classes so for classify this type of dataset we have formed a network of EIFN neurons which comes under the series of spiking neurons and we have converted the target classes into the binary formats and then we have executed are learning because one neuron can classify as class 1or class 2. Therefore, for multiple classes we made a connected network of EIFN neurons. In the conventional neural networks, To classify the multi target dataset we need multiple number of hidden layers and hidden neurons. Because of that these networks becomes more complicated to understand and to evaluate also and they consume more time as compare to spiking neural network based classifiers. Our malaria dataset is having maximum 10 classes and to classify them we have taken 4 EIFN neurons in output layer (Fig.4). Sigmoid function is taken as activation function for EIFN neurons. This activation function will provide the outcome in the range between 0 and 1.

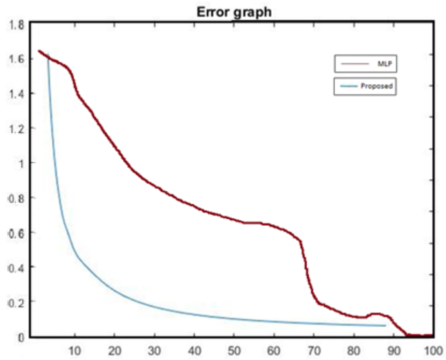

Figure 4: Graphical representation of Error vs. Epoch for Betora

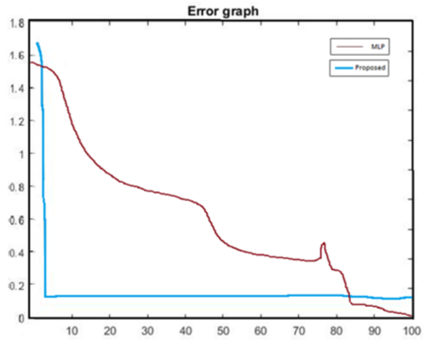

Figure 5: Graphical representation of Error vs. Epoch for Volvoi

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Table 4: Comparison of Accuracy Results of data

S. No	Town Name	MLP	Proposed Model
		Accuracy	Accuracy
1	Betora	0.7251	0.9625
2	Volvoi	0.7358	0.9693
3	Adcolna	0.7056	0.9784
4	Adpai	0.7458	0.9642
5	Betqui	0.7241	0.9827
6	Candola Town	0.7898	0.9724

Table 5: Comparison of Sensitivity measures

	Town Name	MLP	Proposed
		Sensitivity	Specificity
1	Betora	0.7451	0.9620
2	Volvoi	0.7758	0.9691
3	Adcolna	0.6956	0.9782
4	Adpai	0.8158	0.9643
5	Betqui	0.7241	0.9824
6	Candola Town	0.7898	0.9725
7	Boma	0.7451	0.9656

5. CONCLUSION

Using proposed learning rule, we obtained improved classification results as shown in accuracy table when compared with MLP. The graph shows results obtained from proposed model and MLP, it is clear that classification results from proposed model are better than MLP. Proposed learning rule is under supervised learning for Spiking Network. Here both MLP and proposed model is iterative models that are comparable to each other.

6. ACKNOWLEDGMENT:

We are thankful to the Department of Science and Technology, Science and Engineering Research Board, New Delhi, India vide, Project no. ECR/2017/001074, for providing us support for carrying out this work as a part of the sponsored project on developing spiking neural network algorithms.

REFERENCES

1. Brette, Romain, and Wulfram Gerstner. "Adaptive exponential integrate-and-fire model as an effective description of neuronal activity." Journal of neurophysiology 94, no. 5 (2005): 3637-3642.
2. Buscema, Massimo. "Back propagation neural networks." Substance use & misuse 33, no. 2 (1998): 233-270.
3. Izhikevich, Eugene M. "Simple model of spiking neurons." IEEE Transactions on neural networks 14, no. 6 (2003): 1569-1572.
4. Yadav, Abhishek, Deepak Mishra, R. N. Yadav, Sudipta Ray, and Prem K. Kalra. "Learning with single integrate-and-fire neuron." In Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on, vol. 4, pp. 2156-2161. IEEE, 2005.
5. Mishra, Deepak, Abhishek Yadav, and Prem K. Kalra. "Learning with single quadratic integrate-and-fire neuron." In International Symposium on Neural Networks, pp. 424-429. Springer, Berlin, Heidelberg, 2006.
6. Naud, Richard, Nicolas Marcille, Claudia Clopath, and Wulfram Gerstner. "Firing patterns in the adaptive exponential integrate-and-fire model." Biological cybernetics 99, no. 4-5 (2008): 335.
7. Barranca, Victor J., Daniel C. Johnson, Jennifer L. Moyher, Joshua P. Sauppe, Maxim S. Shkarayev, Gregor Kovačič, and David Cai. "Dynamics of the exponential integrate-and-fire model with slow currents and adaptation." Journal of computational neuroscience 37, no. 1 (2014): 161-180
8. Hertäg, Loreen, Joachim Hass, Tatiana Golovko, and Daniel Durstewitz. "An approximation to the adaptive exponential integrate-and-fire neuron model allows fast and predictive fitting to physiological data." Frontiers in computational neuroscience 6 (2012): 62.
9. Liu, Ying-Hui, and Xiao-Jing Wang. "Spike-frequency adaptation of a generalized leaky integrate-and-fire model neuron." Journal of computational neuroscience 10, no. 1 (2001): 25-45.
10. Lippmann, Richard P. "Pattern classification using neural networks." IEEE communications magazine 27, no. 11 (1989): 47-50.
11. X. S. Yang and X. He, "Bat algorithm: literature review and applications", International Journal of Bio-Inspired Computation, 5(3), 141–149, 2013.
12. A. H. Gandomi and A. H. Alavi, "Krill herd: a new bio-inspired optimization algorithm", Communications in Nonlinear Science and Numerical Simulation, 17(12), 4831–4845, 2012
13. L. Y. Chuang, H. W. Chang, C. J. Tu and C. H. Yang, "Improved binary pso for feature selection using gene expression data", Computational Biology and Chemistry, 32(1), 29–38, 2008.
14. D. Rodrigues, L. A. M. Pereira, R. Y. M. Nakamura, K. A. P. Costa, X. S. Yang, A. N. Souza and J. P. Papa, "A wrapper approach for feature selection based on bat algorithm and optimum path forest", Expert Systems with Applications, 41(5), 2250–2258, 2014.
15. J. Feng and G. Li, "Integrate-and-fire and Hodgkin Huxley models with current inputs", Journal of Physics A: Mathematical and General, 34(8), 1649, 2001.