



---

# MALARIA SEVERITY CLASSIFICATION THROUGH JORDAN-ELMAN NEURAL NETWORK BASED ON FEATURES EXTRACTED FROM THICK BLOOD SMEAR

*H. Chiroma,<sup>\*†</sup> S. Abdul-kareem,<sup>\*</sup> U. Ibrahim,<sup>†</sup> I. Gadam Ahmad,<sup>‡</sup> A. Garba,<sup>‡</sup>  
A. Abubakar,<sup>§</sup> M. Fatihu Hamza,<sup>¶</sup> T. Herawan<sup>\*</sup>*

---

**Abstract:** This article presents an alternative approach useful for medical practitioners who wish to detect malaria and accurately identify the level of severity. Malaria classifiers are usually based on feed forward neural networks. In this study, the proposed classifier is developed based on the Jordan-Elman neural networks. Its performance is evaluated using a receiver-operating characteristic curve, sensitivity, specificity, positive predictive value, negative predictive value, confusion matrix, mean square error, determinant coefficient, and reliability. The effectiveness of the classifier is compared to a support vector machine and multiple regression models. The results of the comparative analysis demonstrate a superior performance level of the Jordan-Elman neural network model. Further comparison of the classifier with previous literature indicates performance improvement over existing results. The Jordan-Elman neural networks classifier can assist medical practitioners in the fast detection of malaria and determining its severity, especially in tropical and subtropical regions where cases of malaria are prevalent.

Key words: *learning algorithms, Jordan-Elman neural network, malaria, model performance, evaluation*

Received: June 25, 2013

DOI: 10.14311/NNW.2015.25.028

Revised and accepted: November 26, 2014

## 1. Introduction

The tropical disease of malaria causes between one to three million deaths annually [17]. It is caused by a parasite (plasmodium) transmitted by female Anopheles

---

<sup>\*</sup>Haruna Chiroma – Corresponding author, Tutut Herawan, Sameem Abdul-kareem, Department of Artificial Intelligence, Faculty of Computer Science and Information Technology, Department of Biological Sciences, University of Malaya, Kuala Lumpur, Malaysia, E-mail: [freedonchi@yahoo.com](mailto:freedonchi@yahoo.com), [hchiroma@acm.org](mailto:hchiroma@acm.org)

<sup>†</sup>Haruna Chiroma, Umar Ibrahim, Federal College of Education (Technical), Gombe, Nigeria

<sup>‡</sup>Ibrahim Gadam Ahmad, Abdulmumini Garba, College of Medical Sciences, University of Maiduguri Teaching Hospital, Department of Computer Science, Maiduguri, Nigeria

<sup>§</sup>Adamu Abubakar, Kulliyah of Information and Communication Technology, International Islamic University Malaysia, Kuala Lumpur, Malaysia

<sup>¶</sup>Mukhtar Fatihu Hamza, Bayero University, Department of Mechatronics Engineering, 3011, Kano, Nigeria

mosquitoes. Malaria is prevalent in tropical Africa and ranked among the diseases with the highest mortality rates and a discernible economic and social impact [42]. The most severe and complicated effects of the disease are commonly found among infants (0–5 years) and pregnant women [41]. Malaria infection during pregnancy constitutes a major public health problem in the tropical and subtropical regions of the world [27]. The 90% of the global malaria cases occur in Sub-Saharan Africa [2]. Studies have shown that pregnancy exacerbates malaria through a non-specific hormone-dependent depression of the immune system. The protective anti-plasmodia activity is suppressed at pregnancy which has clinical consequences with important public health implications on pregnant women [28]. Malaria infection also causes premature deliveries of infants with low birth weight due to intrauterine growth retardation attributed to placental parasitisation [37]. Environmental changes such as global warming, civil disturbances, increased global mobility and increased drug resistance are also considered among the causes of the rising number of malaria cases worldwide [14]. In endemic areas, children under the age of five years bear the brunt of the burden because their immunity levels are considerably lower than those of the other members of their community [32]. Species of the plasmodium parasite that causes malaria in humans are: plasmodium (P) falciparum, P. Vivax, P. Ovale, and P. Malariae. While P. Vivax and P. Ovale cause the mildest form of malaria, P. Falciparum causes the severest. In Africa, the most dominant parasite is the virulent plasmodium falciparum [26, 28]. Thus, the present situation calls for an accurate diagnostic tool that detects the disease and instantly identifies the level of severity so that adequate treatment can be provided as fast as possible.

An important aspect of a diagnostic tool is machine learning. Major components of machine learning are learning domain, training set, learning system, and testing. Machine learning constitutes the process of identifying a learning domain, evaluating the learning system through testing, and generating results to be used [4]. The machine learning method that can be used in predicting the class of different categories of unknown item to a specific class label based on a training set is called classification [11].

Artificial Neural Networks (ANN) offer an alternative approach to solve problems in classification, pattern recognition, and assessment through non-linear mapping, classification learning, and real-time optimization [23, 31]. Prediction models are built with ANN because of their ability to learn from experience without prior information [15]. ANN possess the ability to create a model with continuous output [3] and have been proven to serve as an adequate tool for solving function approximation problems [20].

The classification of malaria parasites using feed-forward neural networks has been the object of a number of existing studies but applying a recurrent neural network in this domain is an alternative which deserves consideration and further examination. Gao *et al.* [12] applied a multi-layer perceptron to forecast malaria and their model achieved 84.85% forecast accuracy. Ross *et al.* [33] applied back-propagation neural networks to classify different malaria species isolated in infected red blood cells through thin blood smears. They achieved a positive prediction value of 81% with a sensitivity of 85%. More recently, Abu *et al.* [1] classified P. falciparum, P. viva and P. malariae parasites using multi-layer feed-forward neural networks, and their experimental results obtained a classification accuracy

of 89.80%. In another study, an asymptomatic malaria diagnosis tool with a multi-layer perceptron could diagnose with an accuracy of 80.0% [25]. Subsequently, Yunda *et al.* [44] used a multi-layer perceptron to classify malaria parasites in thick film blood images. Their experimental results translated into an accuracy of 77.19%. Most recently, the model created by Imroze *et al.* [19] based on multi-layer perceptron, achieved 81% correctness.

Recurrent neural networks are more appropriate for pattern detection and possess the ability to characterize the related computational structure in a more parsimonious way than the feed forward neural network. Various categories of RNN exist in literature but the most outstanding is the Jordan-Elman neural network [14]. Experimental evidence established in [13] pointed out that an RNN operates decidedly better than a feed-forward neural network in terms of prediction accuracy.

Studies on the classification of the severity level of malaria using recurrent neural network based on the size of the infected RBC (SRBCI), the shape of parasite (SP), the number of chromatin dots (NCD), the number of parasites per RBC (NPRBC), the texture of RBC (TRBC), and the location of chromatin (LC) are still scarce in the literature to the best of our knowledge because existing studies have concentrated on the application of the feed-forward neural network.

This study proposes an alternative type of malaria classification based on the Jordan-Elman neural network to improve classification accuracy. The proposed classifier has been evaluated using several performance metrics.

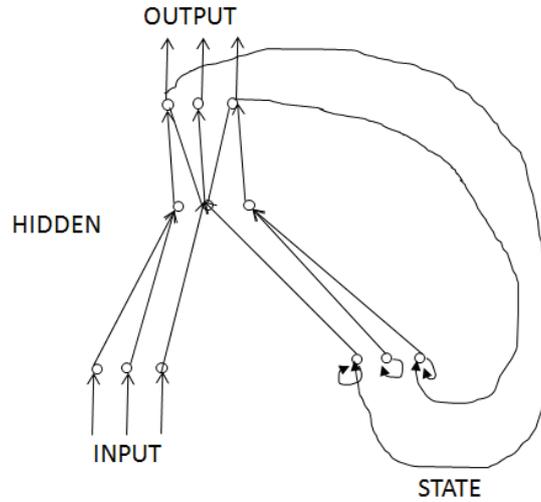
Section 2 of the paper presents the methodology which discusses the Jordan-Elman neural network, learning algorithms, data collection location and procedure, and description of the performance metrics used in evaluating our proposal. Section 3 presents the results and discussion followed by concluding remarks in Section 4.

## 2. Methodology

### 2.1 Jordan-Elman neural network

In feed-forward neural networks, neurons are connected together in the form of a network [5]. Information flows from input neurons into the network in one direction to output neurons, and there is no feedback connection in such networks architecture. A network with recurrent connections has been described by Jordan (see Fig. 1). These recurrent networks associate static patterns with output patterns that are sequentially ordered. Hidden nodes visualize their own previous output to serve as a guide for subsequent behavior. Memory is provided in the network by the recurrent connections [8].

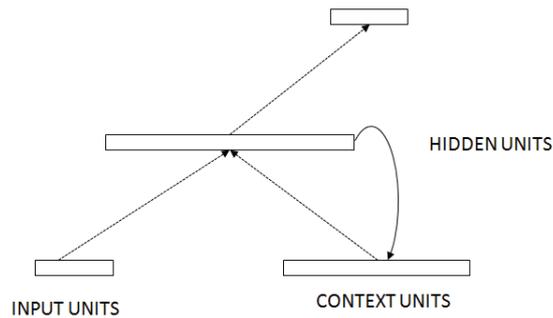
Elman [8] modified Jordan's network (Jordan-Elman neural network) to include context units seen in Fig. 2 and the refined version in Fig. 3, and augmented the model with an input layer. However, these units only interact with the hidden layer, not the external layer. Previous values of Jordan's neural network output fed back into the hidden units while the hidden neuron output fed back to itself in the Jordan-Elman neural network [13]. The network architecture of the Jordan-Elman neural network consists of three layers of neurons, namely internal and



**Fig. 1** Original design by Jordan. One-to-one connection is maintained from output to state units, the weight has a constant value of 1.0, and some connections are not shown.

external input neurons on the input layer. The internal input neurons are also referred to as context units or memory. Internal input neurons receive input from their hidden nodes. Hidden neurons accept their inputs from external and internal neurons. Previous outputs of hidden neurons are stored in neurons in the context units [30].

A mathematical description of the Jordan-Elman networks is presented in [34] and can be modified as follows for simple understanding. From Fig. 3,  $z^{-1}$  represent time delay,  $\mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3$  are the weight matrix between input and hidden layer, the weight matrix between hidden and output layer, and the weight matrix between



**Fig. 2** Jordan-Elman networks. Hidden layers transfer activations to the context units based on one-to-one relationships, constant weight of 1.0 is maintained, and all dotted lines indicate connections that are trainable.

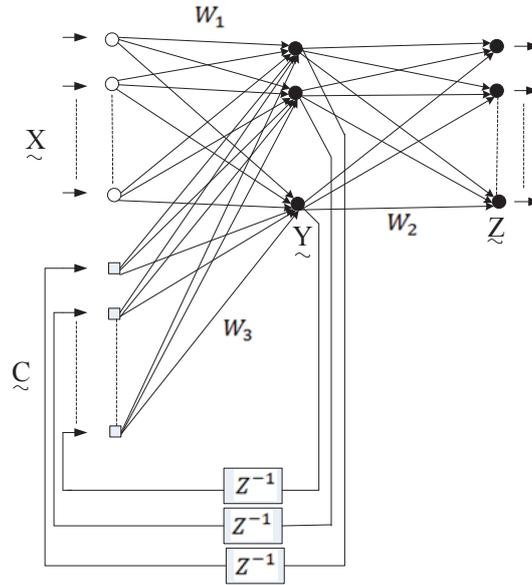


Fig. 3 Jordan-Elman neural network.

context and hidden layer, respectively. The components of the vectors at  $s$ -th iteration are:

$$\begin{aligned} \mathbf{x}_i^{(s)} &\in \tilde{X}, i = 1, 2, \dots, n, \mathbf{y}_j^{(s)} \in \tilde{Y}, j = 1, 2, \dots, m, \\ \mathbf{z}_k^{(s)} &= \tilde{Z}, k = 1, 2, \dots, l, \mathbf{c}_i^{(s)} = \tilde{C} \quad i' = j, \end{aligned}$$

where  $\tilde{X}$  denotes the input layer vectors,  $\tilde{Y}$  is the hidden layer vector,  $\tilde{Z}$  is the output layer vector and  $\tilde{C}$  is the context layer vector while  $i, j, k$  and  $i'$  are indices of their respective nodes. Computation at hidden node outputs requires an activation function. Therefore, consider the activation function  $f(\cdot)$  at  $j$ -th hidden node output at  $s$ -th iteration be given as  $\mathbf{y}_j^{(s)} = f(\mathbf{a}_j^{(s)})$ . Then, the relation for context layer input at  $s$ -th iteration is  $\mathbf{c}_i^{(s)} = \mathbf{y}_j^{(s-1)}$ , where  $\mathbf{a}_j^{(s)}$  is taken at  $s$ -th iteration and there is linear output of node  $j$ . The initialization of the Jordan-Elman neural network is as follows:

assuming  $\mathbf{y}_j^{(0)} = 0, j = 1, 2, \dots, m, s = 0$  at the context layer, the input results to  $\mathbf{c}_i^{(1)} = 0$ . In the context of the weight matrices  $\mathbf{W}_{1ij} \in \mathbf{W}_1, \mathbf{W}_{3ij} \in \mathbf{W}_3$  and  $\mathbf{W}_{2ij} \in \mathbf{W}_2$  The hidden layer neurons output is

$$\mathbf{y}_j^{(s)} = f \left( \sum_{i=1}^n w_{1ij} \mathbf{x}_i^{(s)} + \sum_{j=1}^m w_{3ij} \mathbf{y}_j^{(s-1)} \right), \quad t = j. \quad (1)$$

The output layer neurons output at  $s$ -th iteration is

$$\mathbf{z}_k^{(s)} = f \left( \sum_{j=1}^m w_{2jk} \mathbf{y}_j^{(s)} \right), \quad (2)$$

$$\mathbf{z}_k^{(s)} = f \sum_{j=1}^m w_{2jk} \left( \sum_{i=1}^n w_{2ij} \mathbf{x}_i^{(s)} + \sum_{j=1}^m w_{3ij} \mathbf{y}_j^{(s-1)} \right). \quad (3)$$

The error  $E$  is minimized by updating the weight coefficients at the output layer by

$$w^{\text{new}} = w^{\text{old}} + \eta \Delta w, \quad (4)$$

where  $\eta$  is the learning rate. The relationship that defined pattern vectors and output nodes is

$$E_{(w)} = \frac{1}{2} \sum_{s=1}^p \sum_{k=1}^l \left( \mathbf{Z}_k^{(s)} - \mathbf{z}_k^{(s)} \right)^2, \quad (5)$$

$$E_{(w)} = \frac{1}{2} \sum_{s=1}^p \sum_{k=1}^l \left( \mathbf{z}_k^{(s)} - \left[ f \sum_{j=1}^m w_{2jk} \left( \sum_{i=1}^n w_{2ij} \mathbf{x}_i^{(s)} + \sum_{j=1}^m w_{3ij} \mathbf{y}_j^{(s-1)} \right) \right] \right), \quad (6)$$

where  $Z_k^2$  is the desired value at  $s$ -th iteration and  $p$  represents the sequence of training length.  $\mathbf{W}_1$  and  $\mathbf{W}_3$  can be computed with back propagation algorithms as they conform to the feed-forward character.

The back propagation algorithms for computation in the context of  $\mathbf{W}_3$  are described as follows:

$$\mathbf{W}_3 = \sum_{i=1}^n w_{3ij} \mathbf{y}_j^{(s)}, \quad (7)$$

and we have the sigmoid function ( $f_j$ ) given as

$$f_j = \frac{1}{1 + e^{-\lambda \mathbf{W}_3}}. \quad (8)$$

The sigmoid activation function is used for the computation in the neurons of the Jordan-Elman neural network. Here,  $\lambda$  is the step size which controls the steepness of  $f_j$  defined as

$$f_j = \frac{1}{1 + e^{-\lambda w_{3ij} \mathbf{y}_j^{(s)}}}. \quad (9)$$

To obtain Eq. (9), we substitute Eq. (7) in the indicated activation function as expressed in Eq. (8). Hence, it was used in the computation of

$$\mathbf{W}_2 = \sum_{k=1}^l w_{2jk} f_j. \quad (10)$$

The activation function presented in Eq. (9) is passed to Eq. (10) for computation with the network weights in the neurons of the network. The expression for the

operation is given by

$$\mathbf{W}_2 = \sum_{k=1}^l \left( \frac{w_{2jk}}{1 + \ell^{-\lambda \sum_{i=1}^n w_{3ij} \mathbf{y}_j^{(s)}}}} \right). \quad (11)$$

The computation of Eq. (11) is in the hidden layer neurons of the Jordan-Elman neural network, whereas the computation at the output layer of the network can be expressed as

$$\tilde{Z} = \frac{1}{-\lambda \left( \sum_k^l \left( \frac{1}{1 + \ell^{-\lambda \left( \sum_{i=1}^n w_{2ij} \mathbf{y}_j^{(s)} \right)}} \right) \right)}. \quad (12)$$

Training output for  $\tilde{Z}$  neurons is

$$r_k^o = \left( \mathbf{z}_k^{(s)} - \mathbf{z}_k^{(s)} \right) \left( 1 - \mathbf{z}_k^{(s)} \right) \lambda \mathbf{z}_k^{(s)}, \quad (13)$$

where  $r_k^o$  is the training signal for output neuron and  $r_j^h$  is hidden layer neuron computed as

$$r_j^h = \left( \sum_{k=1}^l r_k^o w_{2jk} \right) \left( 1 - \mathbf{y}_j^{(s)} \right) \lambda \mathbf{y}_j^{(s)}. \quad (14)$$

Updating weight of the output layer at s-th iteration is

$$w_{2jk}(s) = w_{2jk}(s) + \eta r_k^o \mathbf{y}_j^{(s)}(s), \quad (15)$$

$$w_{2jk}(s) = w_{2jk}(s) + \mathbf{y}_j^{(s)} \mathbf{z}_k^{(s)} \eta \lambda \left( \mathbf{z}_k^{(s)} - \mathbf{z}_k^{(s)} \right) \left( 1 - \mathbf{z}_k^{(s)} \right) (s). \quad (16)$$

Updating weight of the hidden layer at s-th iteration, computation of  $w_{3ij}$  at s-th iteration can be expressed as

$$w_{3ij}(s) = w_{3ij}(s) + \eta r_j^h \mathbf{x}_i^{(s)}(s). \quad (17)$$

The substitution of Eq. (15) into Eq. (17) produces the expression

$$w_{3ij}(s) = w_{3ij}(s) + \eta \lambda \mathbf{x}_i^{(s)} \mathbf{y}_j^{(s)} \left( \sum_{k=1}^l r_k^o w_{2jk} \right) \left( 1 - \mathbf{y}_j^{(s)} \right) (s). \quad (18)$$

The value of Eq. (18) in training the Jordan-Elman neural networks is used to update coefficients of weight between hidden memory units. The value of Eq. (18) update coefficients of weight between memory units and hidden layer using the method of training applied in the Jordan-Elman neural networks.

## 2.2 Learning algorithms

Iterative non-linear optimization algorithms are responsible for minimizing error in most common ANN architectures. The accuracy of the global error function depends on the ANN weights. The best performing learning algorithms (LA) converged to minimum global error function with smaller number of iterations. Optimization algorithms are mostly based on gradient descent algorithms, in which the activation function is differentiable continuously. ANN learns by approximating unfamiliar mapping of input-output relation by modifying connection weights of ANN. Input and desired output are required for error function minimization. Many LA exist in literature, such as the most commonly used back-propagation algorithm, while others include scale conjugate gradient, conjugate gradient method, and back-propagation through time [29]. Back-propagation has been generally accepted as the suitable LA for searching correct weights of ANN [24]. The fastest or most exact LA for a specific problem is difficult to decide because of the many factors that need to be taken into account, such as problem complexity, volume of data available for training, number of network weights and biases in a given networks, error goal. Also, it needs to be decided first whether the task to be solved by neural network is pattern recognition or function approximation [35]. In a particular problem, searching for the optimum solution requires different ANN LA to be considered and experimentally tried. The decision to be made on architectural configurations of ANN is critical for topology description. Therefore, to establish the best performing ANN LA for a specific problem, there is the need to experimentally try different LA for several times using varying initial weights and then adopt the ANN LA with superior performance in a particular problem domain [36]. The LA, initial weights, and number of epochs are the three critical points required to ensure the accuracy and stability of the learning system [16]. The Quick propagation (QP), back propagation and resilient propagation learning algorithms have been experimentally tried but others were rejected in the study in favour of resilient propagation which generated more satisfactory results [9]. The present study's model considers QP, conjugate gradient (CG), Levenberg-Marquardt (LM), and Delta-Bar-Delta (DBD) for experimental trials.

## 2.3 Model performance evaluations

Any modeling procedure covers testing, validation and verification phases. These procedures suggest to model builders and other stakeholders whether the proposed model will achieve its desired purpose or not. Regardless of which modeling methodology is being employed, the performance exhibited by such a model gives a first insight into the representation expected in real-life systems. A factual appraisal of how good a model performs largely depends on individual views of the model. The verification process works similar to a debugging process in as much as its primary purpose is to confirm whether the model generates the desired results. Verification is not interchangeable with validation but validation and verification usually act in form of a combination, especially when model systems have available measured data. If the results produced by a model are correlated with that of actual systems, then the proper verification of the implementation assumption is achieved and it constitutes a true representation of a real-life system [38]. The

quality of a model is considered a critical issue in its design [18]. Therefore, this research employs model performance evaluation in two phases to ensure the quality and reliability of the proposed model.

**2.3.1 Phase 1: Testing the proposed classifier with out-of-sample data**

The classifier is tested using the following performance matrix: classification accuracy, confusion matrix, sensitivity, specificity, mean square error (MSE), and reliability. The classification accuracy determines how well a proposed classification model can correctly classify items according to their respective classes. The classification accuracy was computed using

$$\text{Accuracy} = \frac{\sum_{i=1}^N \text{assess}(x_i)}{N}, \quad t_i \in T, \tag{19}$$

where  $T$  is the dataset to be classified as

$$\text{assess}(x_i) = \begin{cases} 1, & \text{if classification of } x_i \text{ is correct} \\ 0 & \text{otherwise} \end{cases}$$

and  $N$  is the magnitude of the test dataset [43].

Another performance measure is the confusion matrix comprised of four performance indicators as shown in Tab. I. Listed as follows are true positive (TP), false positive (FP), false negative (FN), and true negative (TN).

Actual Class	Predicted Class	
	Positive	Negative
Positive	TP	FN
Negative	FP	TN

**Tab. I** Confusion matrix with four performance indicators.

For a classification problem of two classes, sensitivity, specificity, positive predictive value (PPV), and negative predictive value (NPV) are computed using Eqs. (20) – (23), as used in [43]. In Xu *et al.* [43], typically the receiver-operating characteristic curve (ROC) curve is used as the metrics to assess the performance of experiments related to classification. The ROC curves clearly indicate the TP and FP values concurrently. Each of the point in the ROC curve is the representation of the sensitivity-specificity pair which corresponds to a certain decision threshold. The metrics are defined as follows:

$$\text{Sensitivity} = \frac{TP}{TP + FN} \times 100\%, \tag{20}$$

$$\text{Specificity} = \frac{TN}{FP + TN} \times 100\%, \tag{21}$$

$$\text{PPV} = \frac{TP}{TP + FP} \times 100\%, \tag{22}$$

$$\text{NPV} = \frac{TN}{TN + FN} \times 100\%. \tag{23}$$

### 2.3.2 Phase 2: Comparison of the proposed classifier with support vector machine and multiple regression

A multiple regression and support vector machine models were chosen for the purpose of evaluating the performance of the proposed Jordan-Elman neural networks classification model. The MSE and determination of coefficient ( $R^2$ ) were computed using Eqns. (24) – (25) as used in [38]. The comparison results are listed in Tab. VII. We thus have

$$\text{MSE} = \frac{1}{2} \sum_{i=1}^N (T_i - O_i)^2, \quad (24)$$

$$R^2 = \left[ \frac{\sum_{i=1}^N (O_i - \bar{O}_i) (T_i - \bar{T}_i)}{\sqrt{\sum_{i=1}^N (O_i - \bar{O}_i)^2 \sum_{i=1}^N (T_i - \bar{T}_i)^2}} \right]^2, \quad (25)$$

where  $\bar{T}_i, \bar{O}_i, O_i, T_i, N$  are average of actual values, average of predicted set, predicted output, actual values and total number of data set, respectively.

### 2.3.3 Model reliability

In respect to model reliability, Wang *et al.* [40] states, “Higher model reliability implies a relaxed requirement to solution constraint. Conversely if the model reliability is weak, then the variation scope of the parameters becomes limited.” As such, the process of achieving a feasible solution can be more complex than anticipated. Reliability of the proposed model is computed with Eq. (26) [7] (see Tab. VIII):

$$\text{Reliability} = \frac{\text{Correct}}{\text{Correct} + \text{E}} \times 100\%. \quad (26)$$

## 2.4 Data collection and standardization

In this research, samples of 450 clinical data of six features (SRBC, SP, NCD, TRBC, LC, and NPRBC) extracted from thick blood smear were collected from the Department of Medical Microbiology, Clinical Parasitology Unit, University of Maiduguri Teaching Hospital. Maiduguri is the state capital of Borno, the north-eastern state of Nigeria, West Africa. Cho [7] pointed out that thick and thin blood smear are the two techniques required to detect the malaria parasite. Species of malaria parasites are determined by thin blood smear whereas the density of the malaria parasite per RBC is determined by thick blood smear. In detecting the level of severity of malaria, infected RBC were used to differentiate between mildly severe and highly severe cases, as indicated in Tab. II. Encoding for Plasmodium Falcifarun (PF), Ansocytos Pork (AP), and Ring Shape (RS) are in Tab. III.

The improved ANN performance depends on the data type presented to the network. Thus, encoding data into acceptable format plays a critical role in improving the neural network’s performance. The encoding process requires the representation of the data in such a way that ANN can easily classify inputs according to

LC	TRBC	SP	NPRBC	Malaria Severity
PF	AP	RS	above 4	Highly Severe (HS)
PF	AP	RS	1 – 4	Mildly Severe (MS)

**Tab. II** *Classification of malaria severity.*

their respective classes [22]. Tab. III shows the encoding method adopted from [22].

Specimen	Code
PF	10
AP	20
RS	30

**Tab. III** *Encoding.*

The inputs to the proposed Jordan-Elman neural network classifier are altogether five while outputs are two as depicted in Fig. 4. Inputs are defined as the independent variables whereas output represents the dependent variables. The data used in the experiments done in the course of this study were not normalized, a practice suggested by [39] who argued that the use of raw data values is preferable to normalized data because it allows to retain the original pattern in the data.

## 2.5 The design of the Jordan-Elman neural network classifier

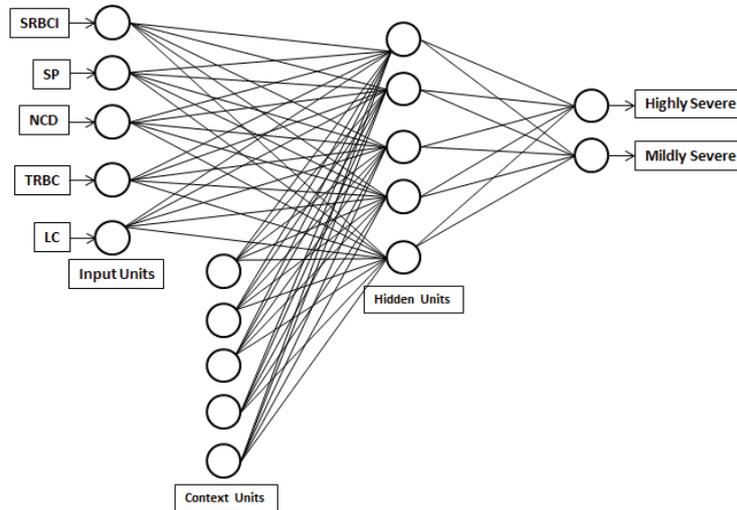
The length of the data measured altogether 450 rows. In a dynamic neural network, the order in which data are presented to the network is significant because inputs are sequentially presented to the network as elements of a cell array (MATLAB manual). Therefore, this study presented the features extracted from thick blood smears to the Jordan-Elman neural network model in the following order: SRBC, SP, NCD, TRBC and LC. Several experiments were conducted with different combinations of the network’s architectural configurations in search for the optimal Jordan-Elman neural networks classifier. In this study, 45 experiments were tried for classification of malaria severity with varying LA, hidden layer neurons, and ratio of data partition. A total of four LA were considered for the study. The experimental series begins with networks of three neurons on the hidden layer, five input neurons, one hidden layer, three context unit neurons, and one output neuron. The data partition ratio of 60:15:25 was a default value set by NeuroSolution (the neural network simulator used in this research) 6.0, and the gradient descent was the first algorithm to be tried.

To select the optimum architecture of the Jordan-Elman neural networks model, it was trained and tested for several configurations. The hidden neurons were determined by increasing the number from three to seven with a step interval of one.

AC	LA	HLAF	CUP	Epoch	CDMS [%]	CDHS [%]	MSEMS	MSEHS	DP
5-5-2	LM	SA	(0.8,SIA)	1000	96.40	91.20	0.0262	0.0614	70-10-20
5-5-2	CG	SA	(0.8,SIA)	1000	98.20	76.50	0.0556	0.1728	71-10-20
5-5-2	QP	SA	(0.8,SIA)	1000	0.00	100	0.0108	0.1926	72-10-20
5-5-2	DBD	SA	(0.8,SIA)	1000	96.10	89.50	0.0281	0.0694	73-10-20

**Tab. IV** Experimental trials to determine appropriate LA and optimum AC. Correct detection and classification of MS (CDMS), correct detection and classification of HS (CDHS), MSE of MS (MSEMS), MSE of HS (MSEHS), hidden layer activation function (HLAF), context units parameters (CUP) = (time, activation function), architectural configurations (AC), LA, data partitioning (training – validation – testing) expressed in percentage (DP), sigmoid axon (SA) and sigmoid integrated axon (SIA).

It was observed that the performance was increasing as the number of neurons increased. The performance started diminishing when the number of hidden neurons reached six and seven. In consequence, the experiments were stopped and the best performing network configuration with five hidden neurons was adopted. The Sigmoid activation function was used in the hidden layer and the linear activation function was used in the output layer as recommended by [38]. Time delay at the context units and epochs were a default setting of the simulator (NeuroSolution 6.0). The best trial results are reported in Tab. IV and the optimal architectural configurations of the proposed Jordan-Elman neural network classifier is depicted in Fig. 4.



**Fig. 4** Optimum architectural configurations of the Jordan-Elman neural network model.

LM algorithms were chosen as this proved to perform very well in classifying MS and HS malaria with an adequate level of accuracy as shown in Tab. IV. The best classification accuracy of MS malaria was achieved by CG although it did not perform as well in HS malaria classification as LM. The worst performance in classification of MS malaria was recorded by QP but generated excellent results in the classification of HS malaria. DBD performed satisfactory in the classification of both MS and HS malaria but less so in the classification of HS as compared to LM. LM achieved more than 90% classification accuracy in both cases while none of the LA was able to achieve similar results. Consequently, LM was identified as the best performing LA clearly outperforming other LA. The probable reason for its superior performance may be attributable to its suitability in less complex neural network architecture which makes it suitable for small and medium sized neural networks. The best and final MSE result during training measured 0.028. Cross validation was specified to terminate after 100 epochs without improvement. Cross validation was terminated at the 45<sup>th</sup> epoch with a minimum MSE of 0.0129 and final MSE of 0.0164. The training stopped as cross validation MSE was no longer improving. MSE of cross validation measured lower than training MSE indicating there was improvement over training MSE. The configurations of the Elman Neural Network realized in the experiments design were used for the classification of the malaria severity. Support vector machine, and multiple regression were also used for the classification of malaria severity.

### 3. Results and discussion

#### 3.1 Classification of malaria severity using the proposed Jordan-Elman neural network classifier

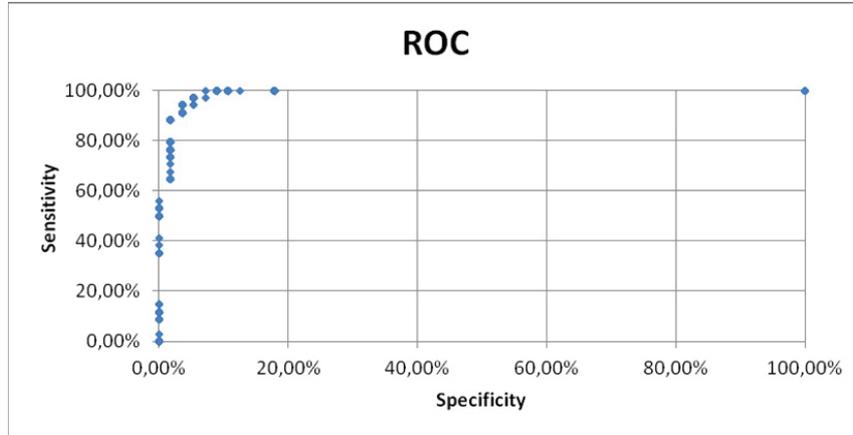
The Jordan-Elman neural network classifier described in Section 2.6 is applied for the classification of malaria severity. Tab. V displays the confusion matrix generated by the proposed Jordan-Elman neural networks classifier at a threshold of 0.5; of a total of 57 cases of MS malaria, 54 were correctly classified and three were wrongly classified as HS malaria. In the case of HS malaria, of a total of 33 cases, 31 were correctly classified and two were incorrectly classified as MS malaria. This result indicated that the proposed model did not perform perfectly but has very high classification accuracy.

	MS	HS
MS	54	3
HS	2	31

**Tab. V** *Confusion matrix.*

Due to the probability of misclassification of a patient’s malaria severity status, the medical practitioner and patient need to be aware that the classifier only provides knowledge of the level of severity. This does not change the fact that any case

of malaria calls for immediate medical treatment. Its use lies in supporting any decision made in regard to the type of treatment to be administered to a patient.



**Fig. 5** ROC curve obtained from testing.

The ROC depicted in Fig. 5 shows the sensitivity and specificity on a vertical and a horizontal axis respectively. The curve is represented by each dotted point shown in Fig. 5. Every sensitivity–specificity corresponds to a certain threshold decision. Values of sensitivity, specificity, positive predictive value, and negative predictive value at a threshold of 0.5 are listed in Tab. VI. Classification accuracy depends on how large the area under the ROC curve is. The larger the area, the more accurate will the classification result be and vice versa. The more the ROC curve extends to the top left hand corner, the larger is the ROC curve which translates into superior classification accuracy. In this case, the ROC curve is very close to the top left hand angle with promising values of sensitivity–specificity as displayed in Tab. VI. This suggests that the proposed Jordan-Elman neural network classifier performed at an acceptable classification accuracy level.

Indicators	Accuracy [%]
Sensitivity	91.2
Specificity	96.4
Positive Predictive Value	93.9
Negative Predictive Value	94.7

**Tab. VI** Performance values obtained at a threshold of 0.5.

Tab. VII shows the reliability of the proposed classifier in detecting and classifying malaria severity (MS or HS). The results reveal that reliability of the proposed model achieves more than 90%.

Malaria Severity	Detection and Classification Rate			
	Correct [%]	Error [%]	Reject	Reliability [%]
MS	96.4	3.60	0	96.4
HS	91.2	8.80	0	91.2

Tab. VII Measuring reliability of the proposed model (expressed as %).

### 3.2 Sensitivity analysis

The sensitive features of available inputs may be of assistance in the process of modeling because it allows handling such contributing features with great care and precision [21].

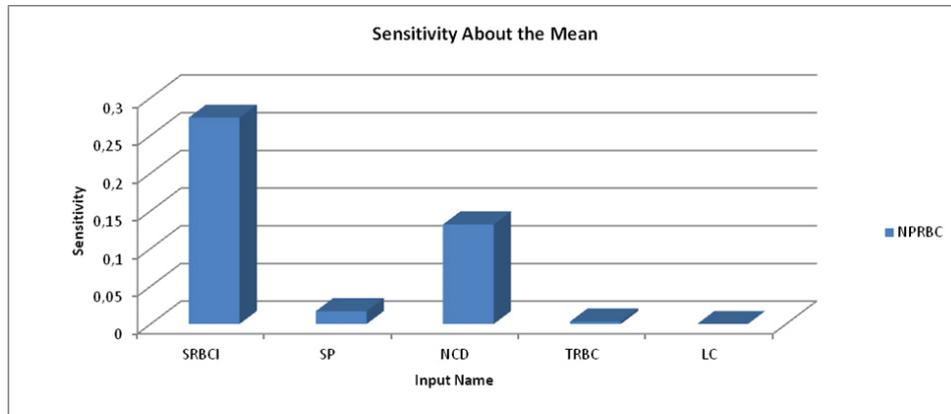


Fig. 6 Sensitivity of inputs variables to NPRBC.

The results obtained in the sensitivity analysis are depicted in Fig. 6 containing all independent variables. The sensitivity of SRBCI valued 0.274, SP 0.017, NCD 0.132, TRBC 0.003, and LC 5.579E-17. All input variables to the proposed Jordan-Elman neural networks model were subjected to sensitivity analysis in order to determine the most contributing factor in causing malaria severity. Both NCD and SRBCI make significant contributions to the model output with SRBCI contributing more than all other variables. This can be explained with the SRBCI having superior sensitivity and also the longest bar which indicates its strength in the cause of the malaria severity.

### 3.3 Comparing performance of the Jordan-Elman neural network with support vector machine, multiple regression, and previous results

Multiple regression analysis was conducted to classify the malaria severity. Six variables were identified, namely TRBC, LC, SP, SRBCI, NCD, and NPRBC, of

which the first five represent the independent variables and the sixth the dependent variable.

Performance Indicator	Our proposal	Support Vector Machine	Multiple Regression Model
MSE	(0.02, 0.06)	(16.92, 17.56)	0.72
R [%]	84	74.19	73

**Tab. VIII** Performance comparison of the proposed Jordan-Elman neural networks and the compared algorithms. HS MSE, MS MSE.

The R Square measure of variability described by the independent variables equaled 73%. The adjusted R Square value of 73% was identical with that of the original R Square indicating how well the model generalized since there was no shrinkage from adjusted values. The MSE measured 0.72, and the summary of the multiple regression results is shown in Tab. VIII.

The support vector machine was applied for the classification of the malaria severity. The results generated by support vector machine shows that a total of 36 cases of HS malaria were detected, 28 were correctly classified and 8 were wrongly classified as MS malaria. In the case of MS malaria, of a total of 84 cases, 68 were correctly classified and 16 were incorrectly classified as HS malaria. The accuracy for the classification of HS and MS are 63.63 and 77.78% respectively, this classification rate and the results in Tab. VIII clearly indicate that the Jordan-Elman neural network classifier performs better than the support vector machine classifier. In summary, the Jordan-Elman neural networks model performed better in terms of classification accuracy than the compared algorithm despite the good performance exhibited by the models.

Reference	Accuracy [%]
[12]	84.85
[33]	85
[25]	80
[19]	81.00
[44]	77.19

**Tab. IX** Earlier published results.

Tab. IX contains the results achieved in earlier studies in which feed-forward ANN was applied for classification in the domain of malaria. The classification of malaria in a clinical blood samples using feed-forward artificial neural networks does not constitute a new method of research, as pointed out in the introduction. However, the application of the Jordan-Elman neural network (recurrent neural network) in this domain does indeed constitute an alternative approach. The proposed Jordan-Elman neural network classifier generates more accurate results than the common feed-forward neural network method currently applied in this field.

Tabs. V to VIII and Fig. 5 have indicated that the proposed model achieved superior accuracy (more than 90%) compared to previous results of other models as reported in Tab. IX. The probable cause of its superior accuracy lies in the fact that SRBCI and NCD were extracted from thick blood smears whose salient features were easier detected by the proposed classifier than by the feed-forward neural networks, multiple regression, and support vector machine. Moreover, the presence of internal feedback makes the recurrent network a more generally applicable model than feed-forward neural networks, multiple regression, and support vector machine. The proposed model performs better than the multiple regression model, which was already expected as previous studies had shown that neural networks performed better than statistical tools. However, it needs to be noted that critical reviews such as that of [45] concluded that the performance of neural networks is not always better than that of statistical tools. Fernandez [10] also provided empirical evidence in which the statistical tool performed better than the support vector machine in short term forecasts. However, contradictory results were found in [6].

## 4. Conclusions

The speedy detection and classification of malaria severity are urgent requirements in order to assist medical practitioners in making accurate and proper decisions that lead to the administration of correct treatment plans. This will eventually reduce the rate of morbidity, premature births, and maternal and infant mortality, especially in the tropical and subtropical regions of the world which are prone to malaria infection. This study proposes the Jordan-Elman neural networks classifier in order to detect and correctly classify malaria severity based on features extracted from thick blood smears. The proposed classifier has proven to perform at a higher accuracy level than other classifiers.

## Acknowledgement

This research is supported by University of Malaya research grant vote No. BK011-2013.

## References

- [1] ABU N.S., ASHIDI N.M.I., CHIA L.L., MOHAMED Z., KALTHUM U.N., ZUHAIRI K.Z. Classification of malaria parasite species based on thin blood smears using multilayer perceptron network. *International Journal of the Computer, the Internet and Management*. 2008, 16(1), pp. 46–52.
- [2] ADEFIOYE O.A., ADEYABA A.O., HASSAN W.O., OYENIRAN O.A. Prevalence of malaria parasite in infection among pregnant women in Oshogbo, southwest Nigeria. *American Eurasian Journal of Scientific Research*. 2007, 2(1), pp. 43–45.
- [3] AO S.I., NG M.K. Gene expression time series modeling with principal component and neural network. *Soft Computing*. 2006, 10, pp. 351–358, doi: 10.1007/s00500-005-0494-8.
- [4] BANZHAF W.P., NORDIN R.E., KELLER R., FRANCONI F.D. Genetic programming: An introduction. San Francisco: Morgan Kaufmann Publishers, 1998.

- [5] BELOHLAVEK R. Feedforward networks with fuzzy signals. *Soft Computing*. 1999, 3, pp. 37–43, doi: 10.1007/s005000050089.
- [6] CHIROMA H., ABDULKAREEM S., ABUBAKAR A.I., SARI E.N., HERAWAN T. A Novel Approach to Gasoline Price Forecasting Based on Karhunen-Loève Transform and Network for Vector Quantization with Voronoid Polyhedral. In: LINAWATI, M.S. MAHENDRA, E.J. NEUHOLD, A.M. TJOA, I. YOU, eds. *Information and Communication Technology*. Berlin Heidelberg: Springer 2014, pp. 257–266, doi: 10.1007/978-3-642-55032-4\_25.
- [7] CHO S. Combining modular neural networks developed by evolutionary algorithms. In: T. BACK, Z. MICHALEWICZ, X. YAO, eds. *Proceedings of the 1997 IEEE International Conference on Evolutionary Computation*, Indianapolis, New York: IEEE, 2007, pp. 647–650, doi: 10.1109/ICEC.1997.592393.
- [8] ELMAN J.L. Finding Structure in Time. *Cognitive Science*. 1990, 14, pp. 179–211, doi: 10.1016/0364-0213(90)90002-E.
- [9] FAN Y., LIANG Q., WEI Y. A Generalized pattern matching approach for multi-step prediction of crude oil price. *Energy Economics*. 2008, 30, pp. 889–904, doi: 10.1016/j.eneco.2006.10.012.
- [10] FERNANDEZ V. Forecasting crude oil and natural gas spot prices by classification methods [online]. In: *IDEAS, 2006* [viewed 2012-04-6]. Available from: [http://www.webmanager.c1/prontus\\_cea/cea\\_2006/site/asocfile/ASOCFILE120061128105820.pdf](http://www.webmanager.c1/prontus_cea/cea_2006/site/asocfile/ASOCFILE120061128105820.pdf).
- [11] GANATRA A.Y.P., KOSTA G., PANCHAL C., GAJJAR K. Initial Classification through back propagation in a neural network following optimization through GA to evaluate the fitness of an algorithms. *International Journal of Computer Science and Information Technology*, 2011, 3(1), pp. 98–116, doi: 10.5121/ijcsit.2011.3108.
- [12] GAO C.Y., XIONG H.Y., YI D., CHAI G.J., YANG X.W., LIU L. Study on meteorological factors-based neural networks model malaria. *Chinese Journal of Epidemiology*, 2003, 24(9), pp. 831–834.
- [13] GENCAY R., LIU T. Nonlinear modeling and prediction with feedforward and recurrent networks. *Physica D*, 1997, 108, pp. 119–134, doi: 10.1016/S0167-2789(97)82009-X.
- [14] GREENWOOD B.M. The Epidemiology of malaria. *Annals of Tropical Medicine and Parasitology*. 1997, 91, pp. 763–769, doi: 10.1080/00034989760518.
- [15] GUNDÍN D.A., GARCÍA C., GOMEZ E., DIMITRIADIS Y.A., VEGA G. Short-term load forecasting for industrial customers using fasArt and fasBack neuro-fuzzy systems. In: Y. DIMITRIADIS, I. ZIGURS, E. GOMEZ-SANCHEZ, eds. *Proceedings of the 14th IEEE power systems computation Conference*, Sevilla, Spain. New York: IEE, 2002, pp. 221–227, doi: 10.1.1.12.6503.
- [16] HARJES F., SCHOLZ-REITER B., KAVIANI A.M. Elman networks for the prediction of inventory levels and capacity utilization. *International Journal of Applied Mathematics and Informatics*. 2011, 5(4), pp. 283–290.
- [17] HETAL P. K., PRATIBHA D.B. Evaluation of one Rapid Method for Diagnosis of Malaria The Optional and Better Replacement of Microscopy. *Research Journal of Recent Sciences*. 2012, 1, pp. 361–364.
- [18] HUSKEN M., JIN Y., SENDHOFF B. Structure optimization of neural networks for evolutionary design optimization. *Soft Computing*. 2005, 9, pp. 21–28, doi: 10.1007/s00500-003-0330-y.
- [19] IMROZE M.K., ACHARYA B., KUMAR B.S., SONI J. Content Based ImageRetrieval Approaches for Detection of Malarial Parasite in Blood Images. *International Journal of Biometrics and Bioinformatics*. 2011, 5(2), pp. 97–110.
- [20] JIN Y. A comprehensive survey of fitness approximation in evolutionary computation. *Soft Computing*. 2003, 9, pp. 3–12, doi: 10.1007/s00500-003-0328-5.
- [21] JONG Y.H., LEE C.I. Influence of geological conditions on the powder factor for tunnel blasting. *International Journal of Rock Mechanics and Mining Science*. 2004, 41, pp. 533–538, doi: 10.1016/j.ijrmmms.2004.03.095.

- [22] KUMAR A.B., KIRAN K., MURTY U.S.N., VENKATESWARLU C.H. Classification and identification of mosquito species using artificial neural networks. *Computational Biology and Chemistry*. 2008, 32, pp. 442–447, doi: 10.1016/j.compbiolchem.2008.07.020.
- [23] LIZHE Y., BO X., XIANJIE W. BP networks model optimization by adaptive genetic algorithms and the application on quality for class teaching. In: *IEEE Proceedings of International Conference on future computer and communication*, Wuhan, China. New York: IEEE 2010, 3, pp. 273–276, doi: 10.1109/ICFCC.2010.5497635.
- [24] MAEDA Y., WAKAMURA M. Simultaneous perturbation learning rule for recurrent neural networks and its FPGA implementation. *IEEE Transaction on Neural Network*. 2005, 16(6), pp. 1664–1671, doi: 10.1109/TNN.2005.852237.
- [25] MAGALHAES A.B.J., AMANCIO A.D., BARRAL M.N., BEZERRIL B.A. Artificial neural networks and Bayesian networks as supporting tools for diagnosis of asymptomatic malaria. In: *IEEE Proceedings of 12<sup>th</sup> International Conference on e-Health Networking Applications and Services*, Lyon, France. New York: IEEE, 2010, pp. 106–111, doi: 10.1109/HEALTH.2010.5556584.
- [26] MARIELLE K.B.A., DENISA E.I.C., MODESTE M.M., ERIC K., PIERRE-BLAISE M., ELIE M., VONNE M.K. Prevalence of Plasmodium falciparum infection in pregnant women in Gabon. *Malaria Journal*. 2003, 2, pp. 1–17, doi: 10.1186/1475-2875-2-18.
- [27] NOSTAN F., TERUILE F., MALANKIRI L. Malaria in pregnancy in an area of unstable endemicity. *Transaction on Royal Society of Tropical Medicine and Hygiene*. 1991, 48(1), pp. 154–160.
- [28] OMOLADE O.O. The status of malaria among pregnant women: a study in Lagos, Nigeria. *African Reproductive Health*. 2003, 7(3), pp. 77–83, doi: 10.2307/3583292.
- [29] PACIFICI F., DEL F.F., SOLIMINI C., WILLIAM J.E. Neural Networks for Land Cover Applications. *Computational Intelligence for Remote Sensing*. 2008, 133, pp. 267–293, doi: 10.1007/978-3-540-79353-3\_11.
- [30] PHAM D.T., KARABOGA D. Training Elman and Jordan Networks for system identification using genetic algorithms. *Artificial Intelligence in Engineering*. 1999, 13, pp. 107–117, doi: 10.1016/S0954-1810(98)00013-2.
- [31] QI D., LIU M., QIU M., ZHANG S. Exponential  $H_\infty$  Synchronization of General Discrete-Time Chaotic Neural Networks With or Without Time Delays. *IEEE Transaction on Neural Network*. 2010, 21(8), pp. 1358–1365, doi: 10.1109/TNN.2010.2050904.
- [32] RAMI O.G., KANU C.P. The prevalence of malaria infection in pregnant women living in suburb of Lagos, Nigeria. *African Journal of Biochemistry Research*. 2010, 4(10), pp. 243–245, doi: 10.5897/AJBR.
- [33] ROSS N.E., PRITCHARD C.J., RUBIN D.M., DUSE A.G. Automated image processing method for the diagnosis and classification of malaria on thin blood smears. *Medical and Biological Engineering and Computing*. 2006, 44, pp. 427–436, doi: 10.1007/s11517-006-0044-2.
- [34] SEKER S., AYAZ E., TURKCAN E. Elman’s recurrent neural networks applications to condition monitoring nuclear power plant and rotating machinery. *Engineering Application of Artificial Intelligence*. 2003, 16, pp. 647–656, doi: 10.1016/j.engappai.2003.10.004.
- [35] SERTER M.U., INAN O., YILMAZ N. A hybrid breast cancer detection system via neural network and feature selection based on SBS, SFS and PCA. *Neural Computing and Applications*. 2013, (3–4), pp. 719–728, doi: 10.1007/s00521-012-0982-6.
- [36] SHEN Y., LI T., HERMANS E., RUAN D., WETS G., VANHOOF K., BRIJS T. A hybrid system of neural networks and rough sets for road safety performance indicators. *Soft Computing*. 2010, 14, pp. 1255–1263, doi: 10.1007/s00500-009-0492-3.
- [37] STEKEETEE R.W., WIRIMA J.J., HIGHTOWER A.W., SLUTSKER L., HELIMAN D.L., BREMAN J.G. The effects of malaria. Malaria prevention in pregnancy on Offspring birth weight, Prematurity and Intra-uterine Retardation in rural Malawi. *American Journal of Tropical Medicine and Hygiene*. 1996, 2, pp. 824–832.
- [38] TAHER A.Z. Fast neural network learning algorithms for medical applications. *Neural Computing and Application*. 2013, 23(3–4), pp. 1019–1034, doi: 10.1007/s00521-012-1026-y.

- [39] VANSTON B.J. *Trading in the Australian stock market using artificial neural networks* [online]. Bond, 2006. PhD thesis, Bond University [viewed 2012–10–26]. Available from: [http://works.bepress.com/bruce\\_vanstone/1/](http://works.bepress.com/bruce_vanstone/1/).
- [40] WANG D., KIM Y., CHEON S.P. C.S.L., KYUNG Y.H. Learning based neural similarity metrics for multimedia data mining. *Soft Computing*. 2007, 11, pp. 335–340, doi: 10.1007/s00500-006-0086-2.
- [41] WARRELL D.A. CEREBRAL MALARIA. Clinic Features, Pathophysiology and Treatment. *Annals of Tropical Medicine Parasitology*. 1997, 91, pp. 875–884.
- [42] WORLD HEALTH ORGANIZATION. WHO Recommended Strategies for the Prevention and Control of Communicable Diseases [online]. World Health Organization, 2001 [viewed 2012-08-23]. Available from: <http://apps.who.int/iris/handle/10665/67088>.
- [43] XU Y., ZHU Q., WANG J. Breast cancer diagnosis based on a kernel orthogonal transform. *Neural Computing and Applications*. 2012, 21(8), pp. 1865–1870, doi: 10.1007/s00521-011-0547-0.
- [44] YUNDA L., ALARCON A., MILLAN J. Automated image analysis method for p. vivax malaria parasite detection in thick blood film blood images. *Revista S & T*. 2011, 10(20), pp. 9–25, doi: 10.18046/syt.v10i20.1151.
- [45] ZHANG G., HU M.Y., PATUWO B.E., INDRO D.C. Artificial neural networks in bankruptcy prediction: general framework and cross-validation analysis. *European Journal of Operation Research*. 1999, 116, pp. 16–32, doi: 10.1016/S0377-2217(98)00051-4.