International Journal of Computer Engineering Science (IJCES)
Volume 2 Issue 6 (June 2012) ISSN: 2250:3439
https://sites.goole.com/site/ijcesjournal http://www.ijces.com/

Tea Insect Pests Classification Based on Artificial Neural Networks

R. K. Samanta[*1] and Indrajit Ghosh[#2]

[*]Expert Systems Laboratory,
Dept. of Computer Science & Application, University of North Bengal,
Raja Rammuhunpur, 734013, India
[1]Email: rksamantark@gmail.com
[#]Dept. of Computer Science, A.C. College, Jalpaiguri, West Bengal, India
[2]Email: ighosh2002@hotmail.com

Abstract. Tea is one of the major health drinks of our society. It is a perennial crop in India and other countries. One of the production barriers of tea is insect pests. This paper presents an automatic diagnosis system for detecting tea insect pests based on artificial neural networks. We apply correlation-based feature selection (CFS) and incremental back propagation network (IBPLN). This is applied on a new database created by the authors based on the records of tea gardens of North Bengal Districts of India. We compare classification results with reduction of dimension and without reduction of dimension. The correct classification rate of the proposed system is 100% in both the cases.

Keywords: Tea insect pests, Neural networks, CFS, Incremental back propagation, Classification.

1 Introduction

Tea is one of the major sources of bread and butter to a significant number of people throughout the world. It is one of the major health drinks of our society. It is also one of the sources of earning foreign capita for countries like India, China, Sri Lanka and others. The demand of tea is increasing day by day. So, it is required to increase the productivity. One of the productivity barriers of tea leaves is the attacks of different insect pests. Crop loss due to pest damage may be from 10% to 40% [1]. Tea is grown under different agro-climatic conditions; thereby provides a favorable breeding ground for a variety of pests. The species causing significant damage are found in [2],[3]. Each species produces their own characteristic symptoms of damage without overlaps. Some of them are region specific. Moreover, there are scarcity of human experts in this domain. So, any automatic intelligent decision support system for proper identifications / classifications of insect pests is welcome. Furthermore, it might be useful to optimize the signs and symptoms for identification / classification process leading to reduced dimensionality.

Analyzing databases is one of the application areas of automated diagnostic systems. Different methodologies are being deployed in the name of data mining and knowledge discovery for finding the relationships embedded in data [4]. Artificial neural networks, in different forms, is one of the highly successful methodologies for control, data mining, classification, image analysis and pattern recognition, nonlinear system modeling, energy production, chemical industries, communications, electrical and electronics industry, and medical applications because of their parallel processing capabilities [5].

This paper, firstly, is to present a new database on tea insect pests created by the authors based on the records of tea gardens of North Bengal Districts of India. Secondly, to report the results of feature extractor, and classification results using incremental backpropagation learning neural networks. The rest of the paper is organized as follows. In the next section, we present our database. Section 3 presents the preliminaries on artificial neural networks (ANN), incremental backpropagation learning networks (IBPLN), and feature extraction and reduction. Section 4 discusses related works. In the next section, applications are discussed. Section 6 presents modeling results. Lastly, our conclusions are summarized.

2 Tea Insect Pests Database

Damage by insect pests is one of the major problems of tea production. We created a database concentrating on eight major insect pests from the records of different tea gardens of North-Bengal districts of India. The database consist of 609 instances belonging to eight classes described by 11 attributes (signs and symptoms); all of which are nominal. These 11 attributes are detailed in Table 1. There are 7 missing values. We discard those records during analysis. Class attribute contains 8 nominal attributes: *Aphid, Helopeltis, Jussid, Pink Mite, Purple Mite, Red Spider, Scarlet Mite, Thrips.* In this database, 60 records for *Aphid*, 120 records for *Helopelties,* 66 records for J*ussid*, 75 records for *Pink Mite*, 60 records for *Purple Mite*, 120 records for *Red Spider*, 60 records for *Scarlet Mite*, and 48 records for *Thrips* are stored. We randomize the records to remove any pre-determined preferential order.

3 Preliminaries

3.1 Artificial Neural Networks

Artificial neural networks (ANN) mimic the workings of the neurons of human brain. The neurons are connected to one another by connection links. Each link has a weight. A simple McCulloch-Pitts model of a neuron [6] is presented in Fig.1. which was presented in the year 1943.

International Journal of Computer Engineering Science (IJCES)
Volume 2 Issue 6 (June 2012) ISSN: 2250:3439
https://sites.goole.com/site/ijcesjournal http://www.ijces.com/

Table 1. North Bengal tea insect pests data description of attributes.

Attribute number	Attribute description	Attribute type	Number of attribute values
1	Site_of_damage	Nominal	8
2	Leaf_surface	Nominal	3
3	Leaf_appearance	Nominal	19
4	Leaf_color	Nominal	4
5	Leaf_spot	Nominal	2
6	Mid_ribcolor	Nominal	1
7	Edge_color	Nominal	1
8	Tip_color	Nominal	1
9	Vein_color	Nominal	1
10	Fingertip_test	Nominal	1
11	Bush_appearance	Nominal	2

This model of neuron is the basis of the discipline of artificial neural networks. In the literatures, different forms of ANNs are there for modeling different tasks. Depending upon the function to be performed, different neural network models assume different modes of operation for the network. Basically, they can be either feedforward type or feedback type.

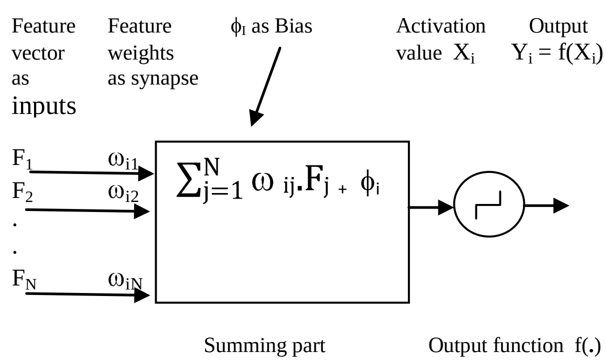

Fig. 1. McCulloch-Pitts model of a neuron.

Modeling with ANN involves two important tasks, namely, *design* and *training* the network. The design of a networks involves (1) fixing the number of layers, (2) the number of neurons for each layer, (3) the node function for each neuron, (4) whether feedback or feedforward, and (5) the connectivity pattern between the layers and the neurons. All these adjustments are to be taken care of for improved performance of the system. The training phase or the learning phase involves adjustments of weights as well as threshold values from a set of training examples. The kind of learning law was first proposed by Donald Hebb [7]. Currently, there are hundreds of such leaning algorithms in the literature [8], but the most well-known among them are backpropagation [9], [10], ART [11], and RBF networks [12].

3.2 Incremental Backpropagation Learning Networks

The normal backpropagation network is not an incremental by its nature [13]. The network learns by the backpropagation rule of Rumelhart et al.[14] under the constraint that the change to each weight for each instance is bounded. With this learning rule, it is likely that adjustments of different weights may be truncated at different proportions. As a result, the network weight vector may not move in the steepest descent during error minimization. In IBPLN, this problem is dealt with by introducing a scaling factor *s* which scales down all weight adjustments so that all of them are within bounds. The learning rule is now

$$\Delta W_{ij}(k) = s(k)\, \eta \delta_j(k) O_i(k) \qquad (1)$$

where W_{ij} is the weight from unit i to unit j, $\eta(0< \eta < 1)$ is a trial-independent learning rate, δ_j is the error gradient at unit j, O_i is the activation level at unit i, and the parameter k denotes the k-th iteration. In the incremental learning scheme, initial weights prior to learning any new instance represent knowledge accumulated so far. IBPLN introduced *two* structural adaptations; neuron generation and neuron elimination. The IBPLN proceeds as follows [13]:
Given a single misclassified instance:

Begin
Repeatedly apply the bounded weight adaptation
    learning rule (1) on the instance until stopping
    criteria are met.
If the instance can be correctly learned, then restore
    the old weights and apply the bounded weight
    adaptation learning rule once;
Else restore the old weights and apply
    the structural adaptation learning rules.
End.

The stopping criteria are: The instance can be correctly learned or the output error fluctuates in a small range.

3.3 Feature Extraction and Reduction

Feature extraction and reduction is one of the important steps for pattern recognition since even the best classifier may perform poorly if the features are not chosen well [15], [16]. The reduced feature vector includes most of the useful information of the original feature vector. This reduced dimensionality helps reducing the database size as well as speeds up the inference procedure especially for a large data base. Moreover, it is essential for a class of complex pattern recognition algorithms. There are different algorithms for the purpose. Correlation-based feature subset selection(CFS) [17], Principle Component Analysis (PCA) [15], Association Rules (AR) [5] are some of the algorithms to mention.

This work uses CFS. The central hypothesis is

"A good feature subset is one that contains features highly correlated with (predictive of) the class, yet uncorrelated with (not predictive of) each other."[17]

A feature evaluation formula, based on ideas from test theory [18], provides an operational definition of the above hypothesis as follows:

$$r_{fc} = \frac{k\overline{r_{fc}}}{\sqrt{\{\, k + k(k-1)\overline{r_{ff}}\,\}}} \qquad (2)$$

where r_{fc} is the correlation between the summed features and the class variable, k is the number of features, $\overline{r_{fc}}$ is the average of the correlation between the features and class variable, and $\overline{r_{ff}}$ is the average inter-correlation between features. CFS is an algorithm that couples this evaluation formula with an appropriate correlation measure and a heuristic search strategy.

To accommodate nominal or categorical as well as continuous or ordinal features in Equation 2, continuous features are transformed to categorical features using the supervised discretisation method of Fayyad and Irani [19] as a preprocessing step before applying in classification task. The theory of information gain [20] is applied estimating the degree of associations between nominal features. Moreover, there are 2^n (n is the number of possible initial features initially) possible subsets of reduced features are possible. It would be impractical especially for a large feature set to explore each and every such subset for finding the best subset. Heuristic search strategies, such as best first and hill-climbing [21] are often used to search the feature subset in reasonable amount of time. Moreover, both filter type as well as wrapper types of feature selection methods use correlation-based approach in different applications [22], [23].

4 Related Works

Artificial neural networks have been applied in agriculture for quite a long time. Combine harvester control [24], Yield prediction [25], Weed detection in sprayers [26], Blueberry bush pruning [27], Weed identification [28]; all of these are agricultural machinery applications using ANN [29]. It has also been applied to agricultural economics [30], agricultural data prediction [31], agricultural watershed [32], crop classification [33], modeling the terminal velocity of agricultural seeds [34], crop nutrition diagnosis [35] and many other applications in agriculture, in general.

In insect pests management, there are some reports available from the literature. Forecasting Aphid flight patterns [36], predict disease management [37], [38], insect pest management [39], [40]. But, however, to the best of our knowledge, we don't find any such work applying CFS and ANN to insect pests database for tea. This work is an attempt to fill this gap.

5 Applications

Basically, this study consists of two stages: The feature extraction and reduction phase by correlation-based feature selection(CFS) and classification phase by incremental back propagation neural networks (IBPLN). The schematic view of our system is shown in Fig. 2.

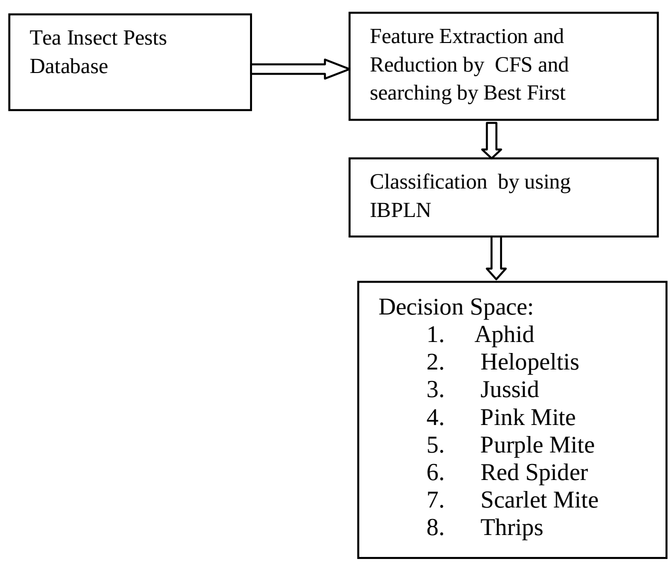

Fig. 2. Block diagram of CFS-IBPLN system for tea inset pests.

5.1 Data Preprocessing

Data preprocessing is the primary step for any model development. We completely randomize the data sets after missing records deletion. There is no outlier in our data. The data set is partitioned into three: Training set (68%), Validation set (16%), and Test set (16%). Nominal values are transformed into numerical ones using one-of-N encoding for applying these to neural networks. Data preprocessing results are shown in Table 2.

5.2 Feature Selection and Extraction

We apply CFS as attribute evaluator and BestFirst as search method to the original data of nominal types. The original attribute set and reduced features set are shown in Table 3.

Table 2. Data preprocessing.

Columns before preprocessing: 12
Columns after preprocessing: 57
Input columns scaling range: [-1..1]
Output column(s) scaling range: [0..1]
Categorical column encoding parameters:
Site_of_damage: One-of-8
Leaf_surface: One-of-4
Leaf_appearance: One-of-19
Leaf_color: One-of-5
Leaf_spot: One-of-3
Mid_ribcolor: Two-state
Edge_color: One-of-3
Tip_color: Two-state
Vein_color: Two-state
Fingertip_test: Two-state
Bush_appearance: One-of-3
Class: One-of-8

Table 3. Original and reduced features.

Sr. No.	Original Attributes	Reduced Attributes
1.	Site_of_damage	Site_of_damage
2.	Leaf_surface	Leaf_surface
3.	Leaf_appearance	Leaf_appearance
4.	Leaf_color	Leaf_color
5.	Leaf_spot	Leaf_spot
6.	Mid_ribcolor	Mid_ribcolor
7.	Edge_color	Tip_color
8.	Tip_color	Vein_color
9.	Vein_color	
10.	Fingertip_test	
11.	Bush_appearance	

5.3 Network Architecture Selection

In general, balancing the trade-off between accuracy and generalizability is the prime characteristic of selecting a model. The ANN model selection includes choice of network architecture and feature selection. The hold-out data set called the *validation set* would be useful helping all these decisions successful [41]. Validation set is a part of our data used to tune the network topology or network parameters other than weights. In our networks, we use logistic function of the form $F(x) = 1/(1+e^{-x})$ in the hidden and output nodes. Theoretically, a network with one hidden layer and logistic function as the activation function at the hidden and output nodes is capable of approximating any function arbitrarily closely, provided that the number of hidden nodes are large enough [42]. So, we use one input layer, one hidden layer, and one output layer. Too few hidden nodes as well as too many hidden nodes have certain problems; a stepwise searching is solicited starting with one hidden node. We primarily bank upon two parameters namely, fitness value (inverse of test error), and AIC (Akaike information criterion) value. Maximum fitness value and minimum AIC value are the prime factors for selecting the initial architecture. Finally, the selection is based on low network error, maximum fitness value and low AIC value. Our searched networks are shown in Table 4.

Table 4. Searched networks with original feature set

Hidden layer activation: Logistic
Output activation: Logistic
Output error function: Sum-of-squares
Classification model: Winner-takes-all
Fitness criteria: Inverse test error
Retrains: 10
Epoch: 500

Serial No.	Architecture	Weights	Fitness	AIC
1.	49-1-8	66	2.46	-2738.77
2.	49-2-8	124	612	-4885.93
3.	49-3-8	182	612	-4769.93
4.	49-4-8	240	612	-4653.93
5.	49-5-8	298	612	-4537.93
6.	49-6-8	356	612	-4421.93

From the Table 4, it is observed that the fitness value gets saturated at 612 for the network 49-2-8 onwards but the weights and AIC values are increasing. So, the initial natural choice is 49-2-8. For final selection, we apply IBPLN to each network of Table 4 to find the network error of each; the results of which are shown in Table 5. From the Table 5, we accept 49-3-8 as the final architecture because of the network error is within the accepted limit. Studies [5] show that sum-squared error = 0.01 might be considered good. We observe that the network errors are in decreasing order but they are getting complex in structure. So, we restrict on 49-3-8.

We conduct the same experiment on reduced feature set. The results of searched networks and errors of networks are shown in Tables 6 and 7, respectively.

Table 5. Errors of networks with original feature set.

Training algorithm : IBPLN
Overtraining control: Jitter to input
The initial weights and biases: Random
Learning rate: 0.1 ; Momentum: 0.1

Serial No.	Architecture	Network errors
1.	49-1-8	0.069135
2.	49-2-8	0.025637
3.	49-3-8	0.000612
4.	49-4-8	0.000255
5.	49-5-8	0.000149
6.	49-6-8	0.000092

Table 6. Searched networks with reduced feature set.

Hidden layer activation: Logistic
Output activation: Logistic
Output error function: Sum-of-squares
Classification model: Winner-takes-all
Fitness criteria: Inverse test error
Retrains: 10 ; Epoch: 500

Serial No.	Architecture	Weights	Fitness	AIC
1.	42-1-8	59	2.34	-2777.63
2.	42-2-8	110	612	-4913.93
3.	42-3-8	161	612	-4811.93
4.	42-4-8	212	612	-4709.93
5.	42-5-8	263	612	-4607.93
6.	42-6-8	314	612	-4505.93

Table 7. Errors of networks with reduced feature set.

Training algorithm : IBPLN
Overtraining control: Jitter to input
The initial weights and biases: Random
Learning rate: 0.1; Momentum: 0.1

Serial No.	Architecture	Network errors
1.	42-1-8	0.066048
2.	42-2-8	0.029238
3.	42-3-8	0.000718
4.	42-4-8	0.000262
5.	42-5-8	0.000197
6.	42-6-8	0.000100

We accept 42-3-8 architecture on the same argument as with original feature set.

6 Modeling Results

As the final step of the work, we calculate mean CCR (correct classification accuracy) of training set, validation set, and test set for both the original feature set and the reduced feature set. The results are shown in the Table 8.

Table 8. Searched networks with reduced feature set.

The classifier: IBPLN Architecture: 49-3-8 (original feature set) Architecture: 42-3-8 (reduced feature set)			
Architecture	Mean CCR%(Training set)	Mean CCR%(Validation set)	Mean CCR%(Test set)
49-3-8	100	100	100
42-3-8	100	100	100

As shown in Table 8, correct classification rate is 100 for both the cases using incremental back propagation neural network. So, we can use the reduced feature set for classification enjoying the advantages of reduced feature set. We use two tools in this study: WEKA [43], and Alyuda NeuroIntelligence [44].

7 Conclusions

In this study, an automatic classification system for detecting tea insect pests based on correlation-based feature selection (CFS) and incremental back propagation neural network is presented. There are some obvious advantages of reduced feature set provided the performance of the intelligent system is not degraded. We apply CFS for feature reduction. Next, the original feature set and reduced feature set were used for classification using incremental back propagation learning neural network (IBPLN). We compare both the results. This study demonstrates that CFS can be used for reducing the feature vector and CFS+ IBPLN combination can be used for other classification problems.

References

1. Banerjee, B.: Tea Production and Processing. Associated Publishing Company, New Delhi, India, 261-262 (1993).

2. TRA: Pests of tea in North-East India and their control, memorandum No. 27, 2nd Ed., Tea research Association, Jorhat, India (1994).
3. Ghosh, I., Samanta, R. K.: TEAPEST: An Expert System for Insect Pest Management in Tea. Applied Engg. in Agriculture, 19(5) 619-625 (2003).
4. Chou, S. M., Lee, T. S., Shao, Y. E., Chen, I. F.: Mining the breast cancer pattern using artificial neural networks and multivariate adaptive regression splines. Expert Systems with Applications. 27, 133-142 (2004).
5. Karabatak, M., Cevdet Ince, M.: An expert system for detection of breast cancer based on association rules and neural network. Expert Systems with Applications. 36, 3465 – 3469(2009).
6. McCulloch, W., Pitts, W.: A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics. 7, 115-133 (1943).
7. Hebb,D. O.: The Organization of Behavior, a Neuropsychological Theory. John Wiley, New York (1949).
8. Roy, A.: Artificial Neural Networks- A Science in Trouble. SIGKDD Explorations. 1(2), 33-38 (2000).
9. Rumelhart, D. E., McClelland (eds.), J. L.: Parallel Distributed Processing: Explorations in Microstructures of Cognition, vol. 1: Foundations, MIT Press, Cambridge, M.A., 318-362 (1986).
10. Rumelhart, D. E.: The Architecture of Mind: A Connectionist Approach. Chapter 8 in J. Haugeland (ed.), Mind_design II, pp. 205-232, MIT Press(1997).
11. Grossberg, S.: Nonlinear Neural Networks: Principles, Mechanisms, and Architectures. Neural Networks. 1, 17 -61(1988).
12. Moody, J., Darken, C.: Fast Learning in Networks of Locally-Tuned Processing Units. Neural Computation. 1, 281-294(1989).
13. Fu, L., Hsu, H., Principe, J. C.: Incremental Backpropagation Learning Networks. IEEE Trans. on Neural Networks. 7(3), 757-761(1996).
14. Rumelhart, D. E., Hinton, G. E., Williams, R. J.: Learning internal representation by error propagation. In Parallel Distributed Processing: Explorations in the Microstructures of Cognition, vol. 1, MA, MIT Press (1986).
15. Calisir, D., Dogantekin, E.: A new intelligent hepatitis diagnosis system: PCA-LSSVM. Expert Systems with Applications. 38, 10705-10708(2011).
16. Avci, E.: A new optimum feature extraction and classification method for speaker recognition: GWPNN. Expert Systems with Applications. 32(2), 485-498(2007).
17. Hall, M. A.: Correlation-based Feature Subset Selection for Machine Learning. Hamilton, New Zealand(1998).
18. Ghiselli, E. E.: Theory of Psychological Measurement. McGraw-Hill, New York (1964).
19. Fayyad, U. M., Irani, K. B.: Multi-interval Discretisation of Continuous-valued Attributes for Classification learning. In Proc. of the XIIIth Int. Joint Conf. on AI. Morgan Kaufmann (1993).

20. Quinlan, J. R.: C 4.5: Programs for Machine Learning. Morgan Kaufmann (1993).
21. Rich, E., Knight, K. : Artificial Intelligence. McGraw-Hill (1991).
22. Yu, L., Liu, H.: Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution. In: 20th Int. Conf. on Machine Learning (ICML-2003), 856-863, Washington DC (2003).
23. Michalak, K., Kwasnicka, H.: Correlation-Based Feature Selection Strategy in Classification Problems. Int. J. Appl. Math. Comput. Sci. 16(4) 503-511 (2006).
24. Hall, J. W.: Emulating Human Process Control Functions with Neural Networks. Unpublished Ph.D. Dissertation. Department of Mechanical Engineering. University of Illinois, Urbana Illinois (1992).
25. Drummond, S. T., Suddeth, K. A., Birrell, S. J. : Analysis and Correlation Methods for Spatial Data. ASAE Paper 95-1335, ASAE (1995).
26. Stone, M. L.: Embedded Neural Networks in Real Time Controls. SAE Paper 941067. In: 45th Annual Earthmoving Industry Conference, SAE, Warrendale PA(1994).
27. Zheng, D., Rohrbach, R. P.: Neural Networks for Ultrasonic Position Control during Blueberry Pruning. ASAE Paper No. 94-1058, ASAE (1994).
28. Zhang, N., Yang, Y., El-Faki, M.: Neural-Network Application in Weed Identification using Color Digital Images. ASAE Paper No. 94-3511, ASAE(1994).
29. Stone, M. L., Kranzler, G. A.: Artificial Neural Networks in Agricultural Machinery Applications. www.agen.okstate.edu/ home/mstone/nnet.htm [accessed on 12/11/2011].
30. Chen, J.: Neural Network Applications in Agricultural Economics. Unpublished Ph.D. thesis, University of Kentucky (2005).
31. Stastny, J., Konecny, V., Trenz, O.: Agricultural Data Prediction by means of Neural Networks. Agric. Econ. – Czech, 57(7), 356–361 (2011).
32. Mutlu, E., Chaubey, I., Hexmoor, H., Bajwa, S. G.: Comparison of artificial neural network models for hydrologic predictions at multiple gauging stations in an agricultural watershed. Hydrol. Process (2008).
33. Vieirai, A. O., Mather, P., Aplin, P.: Improving Artificial Neural Network Performance by using Temporal-Spectral Features for Agricultural Crop Classification. http://ducati.doc.ntu.ac.uk/ uksim/ESM2003/Papers/Track-AI/AI-14/paper.pdf [accessed on 10/12/20011].
34. Ghamari, S., Borghei, A. M., Rabbani, H., Khazaei, J., Basati, F.: Modeling the terminal velocity of agricultural seeds with artificial neural networks. African Journal of Agricultural Research. 5(5), 389-398(2010).
35. Song, H., He, Y. : Crop nutrition diagnosis expert system based on artificial neural networks. In: 3rd Int. Conf. on Information Technology and Applications (ICITA 2005), vol. 1, 357-362 (2005).
36. Lankin, G., Worner, S. P., Samarasinghe, S., Teulon, D. A. J.: Can Artificial Neural Network Systems be used for Forecasting Aphid Flight Patterns? New Zealand Plant Protection. 54, 188-192 (2001).

37. Batchelor, W. D., Yang, X. B., Tschanz, A.T.: Development of a neural network for soybean epidemics. Transactions ASAE. 40, 247-252 (1997).
38. Yang, X. B., Batchelor,W. D.: Modeling plant disease dynamics using neural networks. AI Application. 11, 47-55 (1997).
39. McClendon, R. W., Batchelor, W. D.: Insect pest management neural network. American Society of Agricultural Engineers, ASAE Paper No. 95-3560 (1995).
40. Chon Tae-Soo et al.: Use of an Artificial Neural Network to Predict Population Dynamics of the Forest-Pest Pine Needle Gall Midge (Diptera: Cecidomyiida). Environmental Entomology. 27(6), 1208-1215 (2000).
41. Hung, M. S., Shankar, M., Hu, M. Y.: Estimating Breast Cancer Risks Using Neural Networks. J. Operational Research Society. 52, 1-10 (2001).
42. Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. Neural Network. 2, 359-366 (1991).
43. Hall, E., Frank, G., Holmes, B., Pfahringer, P., Reutemann, I., Witten, H.: The WEKA Data Mining Software: An Update. SIGKDD Explorations. 11(1) (2009).
44. Alyuda NeuroIntelligence 2.2, http://www.alyuda.com.