International Journal of Engineering & Technology, 7 (4) (2018) 3255-3262

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 10.14419/ijet.v7i4.15154
Research paper

A modeling of animal diseases through using artificial neural network

Norma Alias [1] *, Fatin Naemah Mohd Farid 1, Waleed Mugahed Al-Rahmi [2], Noraffandy Yahaya [2], Qusay Al-Maatouk [3]

[1] Ibnu Sina Institute for Scientific &Industrial Research (ISI-SIR), Faculty of Science, Universiti Teknologi Malaysia, 81310, UTM Skudai, Johor, Malaysia
[2] Faculty of Education, Universiti Teknologi Malaysia, 81310, UTM Skudai, Johor, Malaysia
[3] Faculty of Engineering, Computing and Technology, Asia Pacific University of Technology & Innovation (APU), Technology Park Malay-sia, Bukit Jalil-57000 Kuala Lumpur, Malaysia
**Corresponding author E-mail: norm@ibnusina.utm.my*

Abstract

This paper studied the implementation of Artificial Neural Network (ANN) where it well-known recently in veterinary disease research field in Malaysia. The parameter identification under consideration is types of animal disease, types of species and locations of disease based on the Geographical Information System (GIS) data set. There are many types of animal diseases that affect farm animals in Malaysia. In this research, the method of multilayer perceptron neural network is used as main model since it is an effective solving method in predicting the future of veterinary disease. ANN has ability to visual animal diseases involving the computational model. The model is to present the rela-tionship between causes of the species and location and consequence of animal disease without emphasizing the process, considering the initial and boundary condition and considering the nature of the relations. The data collection of animal disease is considered as a large sparse data set. Therefore method of ANN is well suited for optimizing of the data, to train the data operational and to predict the parameter identification of animal disease. The output layers of ANN are plotted in SPSS software for statistical solution and MATLAB programming for sequential ANN implemented. The ANN will be compare to genetic algorithm for the performance and effectiveness of the method. The numerical simulation of ANN helps in future prediction of animal disease based on the species and location parameters.

Keywords: *Mathematical Modeling; Artificial Neural Network (ANN); Geographical Information System (GIS).*

1. Introduction

Neural network consist a number of simple neuron like processing elements and a number of weighted connections between the elements. Those weights are on connections encode the network (Volna, 2012). In this research, ANN is used to give an output to predict animal disease, hence, it will help to predict the disease based on location and species happen for future study. The basic definition or concept can be seen through paper as written by Krenker et al. in 2011 says that Artificial neural network is a computational model that tries to account for the parallel nature of human brain. Besides ANN is one of the most accurate and widely used forecasting model applied in forecasting social, and economic (Al Shamisi et al., 2011, Alias et al., 2012). The characteristics of ANN are fault tolerance, distributed memory, learning ability, collective solution, weighted parameter and network structures (Pintér, 2012). In machine learning and related fields, artificial neural network (ANN) is a computational model inspired by a central nervous system (neuron(s) in brain), and are used to estimate functions that can depend on a large number of inputs and are generally unknown (Martínez-Morales et al., 2014). The development of neurobiology study enables researchers to create mathematical models of neurons to stimulate neural behavior. Safi, (2011) stated that, to train the network, we used the well-back propagation algorithm (BP) of MLP, which consists of an optimization procedure aimed at minimizing the RMSE error observed at the output layer. This algorithm uses a supervised learning mode, meaning that the output corresponding to each input is a priori known, which makes it possible to compute signal errors and try to reduce them through epochs.

2. Multilayer perceptron (MLP) model

For multilayer perceptron, it contains several single layer perceptron. Those single layer perceptron were arranged according some hierarchy. The hierarchy should follow the characteristics. Some of the characteristics were inputs of first layer is taking with the number of perceptron same the number of vectors X of the problem. Then, the output layer generate outputs with the number of neuron equal to the desired number of quantities computed from the inputs as well as there is one layer between single perceptron with another (Morens and Fauci, 2013). Detailed description of MLP architecture of ANN is published previously by Seyam and Mogheir, (2011) where he was used identical approach to ANN construction as well as its training or testing ratio. Neural Networks are able to learn because they can change the connection weights between units. After learning, the knowledge is stored in the weights. For the purpose of training, 70% of the staring dataset is usually used. Training of the neural network is terminated when the network has learned to generalize the underlying trends of

Copyright © 2018 Norma Alias et. al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

relationships exemplified by the data. Generalization implies that the Neural Network can interpolate sensibly at points not contained in its training set. The ability of Neural Network to do so is typically assessed by means of cross-validation, where the performance of the network is evaluated against a novel set of test data, not used during training. This is also known as the testing stage, where the remaining 30% of the starting dataset is used (Milhajlovic et al., 2010).

2.1. Architecture of MLP neural network

Based on Figure 1 is illustrates the nodes with connection of weight. Nodes in input layer act as buffers for delivering the input to nodes in the hidden layer. Each processing node in the hidden layer gathers up the input data after weighting them with strength particular connections from input layer and calculates them (nodes in hidden layer) to be an output.

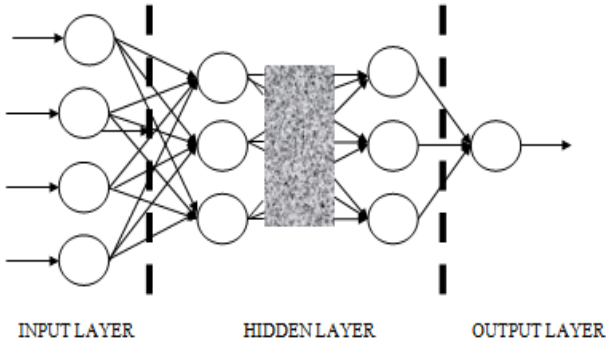


Fig. 1: Nodes with Connection of Weight.

Moreover, Figure 2, ANN consists of a number of simple interconnected processing units also known as neurons, which are analogous to the biological neurons. The important features of neural network (NN) are the ability of supplying fast answer to a problem. Gen and Cheng (1977) was introduced the gradient descent backpropogation algorithm (Sahimi et al., 2001). This method is widely applied in MLP training algorithm. The back propagation has 3 types of layers; an input layer, an output layer and a hidden layer.

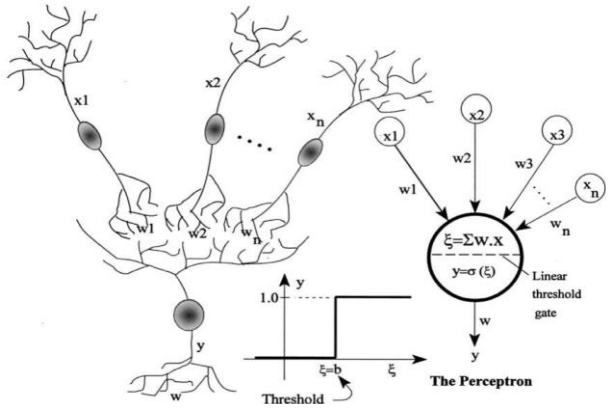


Fig. 2: Analogies between Artificial Neuron and Biological Neuron.

The function f is the overall calculation of the neural network as in Figure 3. It can be simple as equation below,

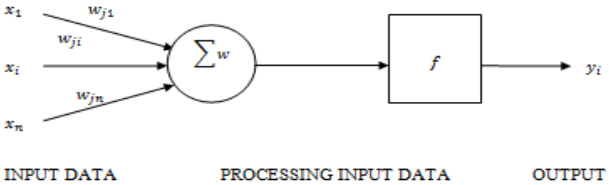


Fig. 3: Basic Elements of ANN.

$$y_i = f(\sum_{i=1}^{n} w_{ji}\, x_i)$$

It gives the chances to adjust weight, Δw_{ij} between i and j where:

$$\Delta w_{ij} = \eta \delta_j x_i$$

The output nodes are calculated as:

$$\delta_j = \left(\frac{\partial f}{\partial \mathrm{nnet}_j}\right)(y_j^{(t)} - y_j)$$

And in the hidden layer, the hidden nodes are calculated by using the formula below:

$$\delta_j = \left(\frac{\partial f}{\partial \mathrm{nnet}_j}\right)(\sum_q w_{jq}\, \delta_q)$$

Where
η- Learning rate
nnet_j- The aggregate weighted sum of input data to nodes j
w_{jq} - Weight for nodes q linked to the output of node j
δ_q- The terms calculated for nodes q linked to the output of node j
$y_j^{(t)}$- The target output for node j

2.2. Programming the neural network model

MATLAB is a numerical computing environment and also a programming language. It allows easy matrix manipulation, plotting of functions and data, implementation of algorithms and etc. There are many sources of code related with Artificial Neural Network implemented in MATLAB. Some coding was created to develop ANN with many terms. The coding was including declarations and calculation of data dependent settings, user definable setting, functions, variables, the outputs of the hidden neurons, the weights, the data and visualization.
There are some importances coding in creating network of neural network. The coding mainly used in artificial neural network is:

```
Net=newff(Pvi,Tvi,20,{'tansig','softmax'},'trainscg');
```

Function above recognize as neural feed-forward backpropagation network. The number of 20 is represented as number of neurons in hidden layer. It can be adjusted by any number in order to get the best result. The network is trained by some of activation function in inputs linked with nodes in hidden layer and also in output layer. Activation functions also can be adjusted by any activation functions where it must suitable with a given data. Activation function plays main role in network in order to avoid large values of error. MATLAB provides built-in transfer functions which are used in this study; linear (purelin), Logistic Sigmoid (logsig) and Hyperbolic Tangent Sigmoid (tansig). A normalization process is applied for training and testing the data. Pvi and Tvi are the arguments for normalized input and output, respectively. The output from the network is denormalized in order to compare it with the measured data.

2.3. Neural network SPSS Software

IBM SPSS Neural Networks software offers nonlinear data modeling procedures that enable us to discover more complex relationships in data. SPSS Neural Networks also provides an alternative predictive capability to approaches such as regression or classification architecture. The output of SPSS Neural Network for Multilayer Perceptron displays information about the neural network where it consists of network structure, network performance, case processing summary and independence variable importance analysis. Network Structure displays summary information about description which it is about the display the neural network information including dependent variables, number of input and output,

number of hidden layers and nodes, and activation function. The network structure also shows the network diagram. As the number of covariates and factor levels increases, the diagram becomes more difficult to interpret. Other than that, the network structure also display synaptic weights where the coefficient estimates that show the relationship between the nodes in a given layer to the nodes in following layer. The synaptic weights are based on the training sample even if the active dataset is divided into training, testing, and holdout data. The number of synaptic weights can become rather large and these weights are generally not used for interpreting network results. For Network Performance, it displays model summary, classification results, predicted by observed chart and residual by predicted chart. The model summary of the neural network resulting the partition and overall including the error, the relative error or percentage of incorrect predictions, the stopping rule used to stop training, and the training time. The error is the sum-of-squares error when the identity, sigmoid, or hyperbolic tangent activation function is applied to the output layer. It is the cross-entropy error when the softmax activation function is applied to the output layer.

2.4. Genetic algorithm

Genetic algorithm (GA) is an adaptive method to solve search and optimization problems. They are based on the genetic processes of biological organisms. GA has been used variety of classification tasks, such as pattern recognition, machine learning, image processing and expert systems as well as ANN. The basic of genetic algorithm is requiring a fitness function that assigns a figure merit of each code solution. As generally in genetic algorithm, parents have to be selected for reproduction and recombine to generate offspring. The process of genetic algorithm is presented in Figure 4.

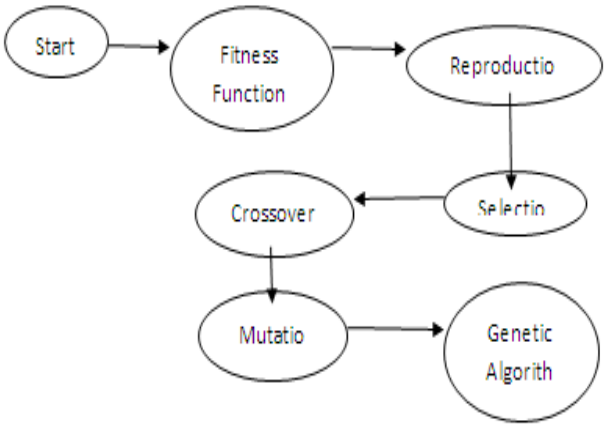


Fig. 4: Process of Genetic Algorithm.

The result from previous study of GA is obtained where it will compare to the Artificial Neural Network of this study.

2.5. Neural network application

Figure 5 shows the number of artificial neural network publication. Based on the graph, the neural network has been widely applied to real-world applications. The number of method has been used increase from 2005 until 2011 for many areas (Wijayasekara et al., 2011).

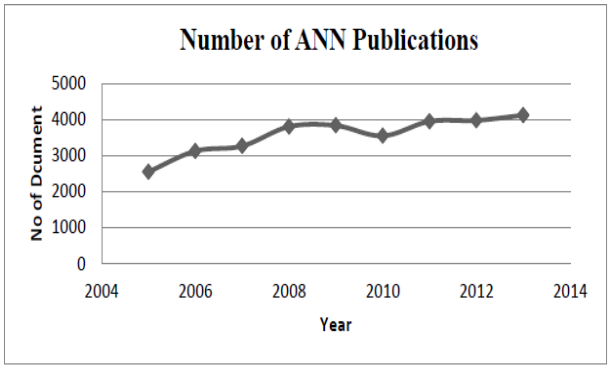


Fig. 5: Number of Artificial Neural Network Publications.

ANN learning is suitable to solve many problems in which the data corresponds to environment. The method is widely used in many areas shown in Table 1. Based on the Table below, the most subject area has been successfully applied the method of artificial neural network is computer science as we know that many computer software and programming can apply the neural network such as to SPSS, MATLAB, Fortran, C and etc. Mathematics is the third highest of subject area on using of Artificial Neural Network. It can be concluded that ANN is quite well-known among mathematicians. ANN also has been used in veterinary, means that ANN is suitable to solve all problems including veterinary without any boundaries.

Table 1: Number of Documents based on Scope Area

Subject Area	Documents	Subject Area	Documents
Computer Science	20686	Social Sciences	1251
Engineering	17865	Decision Sciences	757
Mathematics	5520	pharmacology, Toxicology	744
Medicine	4086	Business, Management and Account	703
Environmental Science	3588	Multidisciplinary	570
Neuroscience	3398	Health Professions	492
Biochemistry, Genetics and Molecular	3212	Immunology and Microbiology	375
Earth and Planetary Sciences	3191	Arts and Humanities	323
Physics and Astronomy	2973	Psychology	317
Materials Science	2922	Economics, Econometrics and Finance	244
Energy	2767	Undefined	220
Agricultural and Biological Sciences	2470	Veterinary	51

Chemistry	2434	Nursing	49
Chemical Engineering	2082	Dentistry	24
Total			47402

3. Research methodology

Figure 6, shows the flow chart for overall of this research. The study has 3 main parts. First is process understanding of overall of ANN. While this process of understanding, the data are manually extracted from VRI data into excel and SPSS software by selected parameters as input and output variables. Second is developing model and code for ANN to show the architecture of ANN, and to train the data by using the of MATLAB software. The process can be proceeding to mathematical modeling. Otherwise, the process needs to adapt with ANN. If satisfy with convergence criterion, visualization and prediction, the numerical analysis and performance evaluations can be performed based on the objectives.

3.1. Data collection and the selection of input and output variables

The overview of this project input and output is based on the disease rate in Malaysia based on the data collection from VRI, Malaysia. In order to test and train the model of ANN, there are a few variables that need to be considered as the input and output of the model. The data collected is distributed into two categories, which are input and also the output. The research started with identification of optimal operating parameters for training and testing. See table 2.

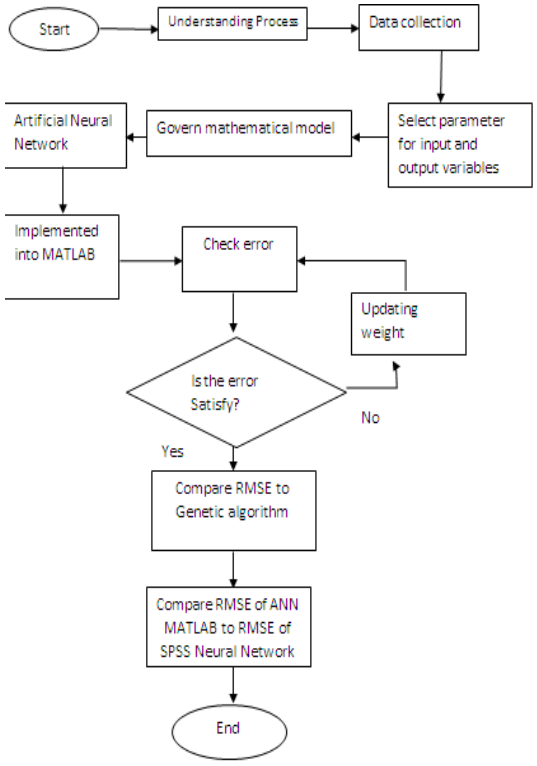


Fig. 6: Process Flow of ANN Modeling.

Table 2: List of Input & Output Variables with Locations

Input Variables		Output Variable	
Species	1.Buffalo 2. Cattle 3. Dairy Cow 4. Goat 5. Sheep 6. Cow	Disease	1. JOCFT 2. LMAT 3. CLA 4. BMCFT 5. MCFT 6. BACFT
Locations			
1. Johor	5. Melaka	9. Perlis	13. Selangor
2. Kedah	6. Negeri Sembilan	10. Pulau Pinang	14. Terengganu
3. Kelantan	7. Pahang	11. Sabah	
4.Kuala Lumpur	8. Perak	12. Sarawak	

3.2. Model development

The model is developed by using MATLAB and SPSS where a feed forward neural network and back propagation training algorithm was designed.
Step 1: Divide the experimental data for testing data and validation data.
Step2: Normalization of testing data and validation data, from 0-1.
Step 3: Set up network and present the training data and target values to the network.
Step 4: Train the network.
Step 5: Test the network in the recall and generalization phase
Step 6: Check RMS error
Step 7: Select the best network based on the smallest RMS error generated.

3.3. Performance measurement

Performance measurement is aim to measure the performance of the method and checks the error. This part shows the analyses that have been done for ANN result. There were mean square error and root mean square error. The result of these MSE and RMSE will then be compare to the MSE and RMSE of genetic algorithm and compare the RMSE between the software's.
Mean Square Error
It is the average of the square of the difference between each output processing element and the desired output. It is used to determine how well the network output fits the desired output, but it doesn't reflect whether two sets of the data move in the same direction (Nanda et al., 2013).

$$MSE = [\sum_1^N \left(\frac{Q_{ob}-Q_{pr}}{n}\right)]$$

Root Mean Square Error
RMSE is actually the root of MSE presents information on the short term efficiency, which is the benchmark of the difference of predicted values about observed values. The lower RMSE, the more accurate is the evaluation and the coefficient of determination (also called R square) measures the variance that is interpreted by the model, which is the reduction of variance when using the model.

$$RMSE = \sqrt{\frac{1}{n}[\sum_1^N \left(\frac{Q_{ob}-Q_{pr}}{Q_{ob}}\right)^2]}$$

4. Result and discussion

The original information data given by VRI that listed on excel was sorted into SPSS software to obtain some statistical results and show the model of ANN based on the veterinary disease. Then the sorted data were applied to develop a model of Artificial Neural Network visualized by MATLAB. The accuracy of the prediction can be measured by calculating the value of root means square error (RMSE) resulted from the simulation of the neural network. The smaller the value of RMSE the better the network predicts the output, where the RMSE is the value of percentage error between predicted values with the desired value. The result obtained will be present the dominant disease at specific locations in Malaysia using the Geographical Information System map.

4.1. Statistical results

The data of veterinary disease were collected based on selected input and output variables that generated by the geographical information system (GIS). The data were sorted using SPSS from VRI Excel data. The data were sorted in different disease, species and location. The total data used in this research are 698 data. From this data, 465 data have been used to train the network while another 233 data are used to test the network.

4.1.1. Location and type of disease that infected by species

The Table 3 shows the number of suspected diseases reported in each location. The highest number of cases reported is in Perak. The second highest is in Sabah and Pahang. There are 108 and 109 cases reported in each location, respectively. Perlis has the lowest number of suspected diseases attacked to species in this area which is 2. On the other hand, the most infected disease was observed in Perak is BMCFT.

Table 3: The Matrices Number of Suspected Disease Reported in Each Area

Disease State	BACFT	BMCFT	CLA	JOCFT	MCFT	Grand Total
Johor	4	17	4	5	9	41
Kedah	-	6	-	-	-	6
Kelantan	-	4	1	-	1	6
Melaka	26	17	4	30	4	84
Negeri Sembilan	5	28	11	15	25	98
Pahang	2	18	15	33	15	109
Perak	33	37	19	32	22	146
Perlis	-	1	-	-	1	2
Pulau Pinang	-	9	-	-	-	9
Sabah	4	28	8	32	36	108
Sarawak	1	1		1	1	5
Selangor	-	13	9	6	-	28
Terengganu	-	28	25	-	3	56
Grand Total	75	207	96	154	117	698

In this research we also found out that BMCFT is the highest disease that infected in all locations with grand totals of 207 cases reported following by JOCFT, MCFT, CLA, BACFT and LMAT by each reported 154, 117, 96, 75 and LMAT cases respectively. LMAT is a suspected disease that less happen in all locations which is 49 cases reported.

4.2. Type of disease that Infected by Species

Table 4 shows the number of species attacked by suspected disease in all locations. Goat has the highest number of infected by BMCFT which is 180 cases of infection. The second highest is 97 of cow which is infected by JOCFT. Buffalo is a species has less infected by suspected disease name LMAT. The goat is a species that have most infected by all the suspected disease with the grand total of 453 cases infected by suspected disease. Also, it was highlighted that BMCFT is the most disease attacked to goat as it stated 180 cases. Buffalo has least number of species to be infected by the suspected disease which it only has three types of disease infected; BACFT, LMAT and JOCFT.

Table 4: The Number of Species and Suspected Disease Cases Reported

Disease Species	BACFT	BMCFT	CLA	JOCFT	MCFT	Grand Total
Buffalo	5	-	-	4	-	11
Cow	65	3	-	97	5	193
Goat	5	180	93	50	104	453
Sheep	-	22	3	3	8	39
#NULL!	-	2	-	-	-	2
Grand Total	75	207	96	154	117	698

4.2. Architecture model of network

The total of 698 of sample data is divided into two parts which is 70% of data set has been set up as training, while the rest 30% of the data set is set up as testing as shown in Table 5.

Table 5: Data Set Division

		N	Percent
Sample Data	Training	502	70.0%
	Testing	194	30.0%
Total		698	100.0%

From the given data set division of Table 5, the architecture of the network is used to specify the neural network architecture in Figure 7. The network for this model has been built by two inputs, 16 nodes of hidden layer and one output by using SPSS Neural Network as we want to visualize the model that we use in MATLAB. Where the inputs are assign as state (location) and species, and the output is S.D (disease). See Figure 7.

From the Table 6, it shows the summary of architecture model of the network shown in Figure 7. In the input layer, there are two input covariates which are state and species. The model use 2 number of nodes (units) in the input layer where the covariates of input data are rescaling as normalized where it will normalize the input data in the range of 0 to 1 by correction 0.02.

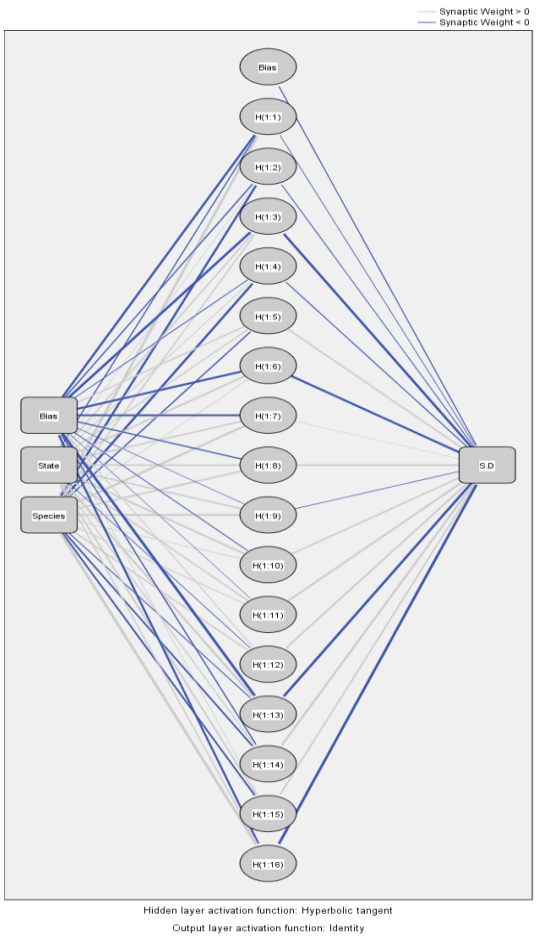


Fig. 7: Architecture of the Network.

Table 6: Network Information

Input Layer	Covariates	1	State
		2	Species
	Number of Units		3
	Rescaling Method for Covariates		Normalized
	Number of Hidden Layers		1
Hidden Layer(s)	Number of Units in Hidden Layer		16
	Activation Function		Hyperbolic tangent
Output Layer	Dependent Variables	1	Suspected Diseases
	Number of Units		1

Rescaling Method for Scale Dependents	Normalized
Activation Function	Identity
Error Function	Sum of Squares

While in the hidden layer of the model, there is only one layer has been used in this research. There are three nodes used in the hidden layer with the set up of the activation function using Hyperbolic tangent. This function takes real-valued arguments and transforms them to the range (-1,1). This function is used for all units in the hidden layers. In the output layer, there is only one dependent variable is used which is suspected disease. The data of suspected disease is rescaling as normalization, also in the range 0 to 1. The activation function links the weighted sums units in a layer to the values of units in the succeeding layer. The model in the output layer is use identity function or known as a linear fuction in the output layer. It takes real-valued arguments and returns them unchanged. This is the activation function for units in the output layer if there are any scale-dependent variables. and the error function used is sum of squares.

4.3. Analysis of neural network mat lab

Variety combination and manipulation processes have been considered to obtain the best performance of the network. The study showed that it is better to increase the number of nodes rather than add another layer of hidden layer to the network. Since there are no theoretical principles for choosing the proper network, then there are several different structures were trained. Neural networks were trained using different numbers of hidden nodes and learning epochs (Ramli, 2014). At the start of the training run, all weights and all biases were initialized with random values. During training, modifications of the network weights and biases were made by back-propagation of the error. When the network was optimized, the testing data were fed into the network to evaluate the trained network. The hidden neurons were varied from 4 to 20. Finding the optimal number of hidden nodes is important since their function is to detect the relationship between network inputs and outputs. If too many hidden nodes are used, the network may be losing its ability to generalize. For the selection of hidden nodes, since there are 2 input variables, we believe that neurons in hidden layer within the range (2-4) are enough, but we have to increase the number of nodes since it could not reach the smallest value of RMS error. For the selection of activation function, a sigmoid function adopted in the hidden layer while a linear function adopted in the output layer. In this study, the testing and training data are normalized within range of (0, 1). This range would allow model prediction of variables higher than the values used in the training phase. Prior to training, the connected weights and the biases in the model are initialized with random values in the range (-1, 1) and the stop criterion is dependent on either minimum mean squared error obtained or training iterations reached.

Table 7: Network Parameter

Network Parameters	Optimal Value
Learning Rate	0.4
Momentum	0.9
Error change	1.0e-4
Maximum epoch	100
Learning epochs	10

Based on the Table 7, the optimal values for each network parameter are selected due to allow the network will learn faster. A higher learning rate means that the network will train faster. We found that learning rate of 0.4 is most possible to train the network faster. To keep the learning speed as fast as possible, the learning rate is self-adjusted by the network to be 0, 1. The number of iterations of the training data, minimize the initial learning rate to the lower boundary of the learning rate when gradient descent is used. Moreover, we found that when the momentum was 0.9 the network could achieve faster convergence and avoid getting trapped in a local minimum. Low values of learning rate and momentum work well in many other applications. However, it should be noted that they tend to be application specific. Error procedure is used for identifying the number of neurons in the hidden layer, since no general guidelines exist for the identification of ANN structure for specific applications. The parameters were summaries in Table 7.

5. Implementation artificial neural network by using mat lab

This part will display and discuss the result from the implementation of artificial neural network by using MATLAB. The objective of using MATLAB is to understand more about how the data was trained, how many numbers of nodes in hidden layer are suited to be used, what types of activation functions is suitable to use for the input data and the output. Before train the data, the data was divided into two parts; 2/3 of data from 689 data is consisting as training data and the rest of the ratio is act as testing data. The data was normalized in range of 0 to 1 and use 20 nodes as the maximum number of neurons in the hidden layer. We use only one hidden layer to train the network since it is more efficient rather than using more than one hidden layer. During the training data, the progress part will continuously read the training until it stopped at the best result of validation. Epoch means the iteration of training has been used and stopped at the best result.

5.1. Number of neurons

Based on this study, the selection of number of nodes in the hidden layer was used by using the implementation of MATLAB. For each number of nodes, it undergoes 20 times of network running in order to approach the smallest value of RMSE. The optimal selection of neurons is 16 neurons in hidden layer as shown in Table 7. From the hidden nodes 4 to 18, the RMSE decrease gradually, this might be due to too many hidden nodes are used, hence, the network may lose its ability to generalize over trained or over forted and inspection of the cross validation set is crucial for the detection of this effect. In addition, the transfer function had been used in this network is hyperbolic tangent function in hidden layer and linear function in output layer. See table 8.

Table 8: RMSE for Every Numbers of Nodes

No. of Nodes in Hidden layer	RMSE
4	0.289259
5	0.277413
8	0.279916
9	0.290749
10	0.277169
12	0.276905
16	0.275055
18	0.276369

5.2. Optimal activation function

The behavior of an ANN depends on both weights and the input-output function or we call it as a transfer function. In every node of network model, we need a transfer function. Every feed forward neural network model often has one or more hidden layers. However we only use one hidden layer in this case. The combination of transfer function allows the network to learn the relationships between input and output vectors. See table 9.

Table 9: RMSE Value for Combination of Activation Functions

Activation-Transfer Function	RMSE
Logsig- Purelin	0.276191
Logsig-Logsig	0.279868
Tansig-Purelin	0.275055
Tansig-Tansig	0.279378
Logsig-Tansig	0.277455
Tansig-Logsig	0.279266

Based on the Table 9, the selection transfer functions in hidden and output layer were considered based on the smallest RMSE value that we have got by using implementation of MATLAB. We were using 16 nodes for every combination since we already got the smallest value of RMSE for tansig-purelin. Although, we had undergo about 20 times trials for each combination of transfer functions with different nodes in order to obtain the smallest value of RMSE. However, only a combination of transfer functions with a specific number of nodes got the smaller value of RMSE compare to any trials of other the final result. Thus, we obtain the best transfer function is tansig (hyperbolic tangent) in hidden layer and purelin (linear function) in output layer with 16 nodes in hidden layer.

5.3. The mapping of animal disease prediction

The Geographical Information system provides the map to locate the infected veterinary disease all over Peninsular Malaysia. The figure below shows the dominant veterinary disease that is infected at specific location. See Figure 8.

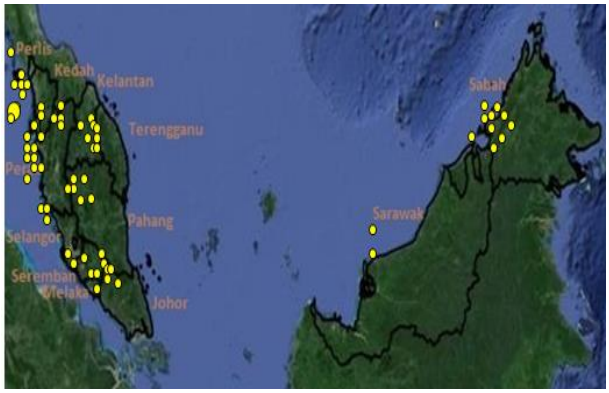


Fig. 8: The Mapping Predicted Disease in Malaysia.

5.4. Prediction of disease for both location and species

For prediction of disease in both condition which is location and species, we obtain the classification of predicted disease by using SPSS Neural Network. The data was divided by two parts which is 70% of data is observed as training sample and 30% of data is observed as testing sample. The prediction of disease is observed by both training and testing sample.

Table 10: Prediction Of Disease for Both Location and Species

Classification								
		Predicted						
Sample	Observed	JOCFT	LMAT	CLA	BMCFT	MCFT	BACFT	% Correct
Training	JOCFT	57	0	0	18	19	12	53.8%
	LMAT	12	0	0	12	0	2	.0%
	CLA	0	0	0	61	7	0	.0%
	BMCFT	2	0	0	109	20	0	83.2%
	MCFT	5	0	0	54	21	0	26.2%
	BACFT	33	0	0	5	0	18	32.1%
	Overall Percent	23.3%	.0%	.0%	55.5%	14.3%	6.9%	43.9%
Testing	JOCFT	24	0	0	7	9	8	50.0%
	LMAT	11	0	0	12	0	0	.0%
	CLA	0	0	0	27	1	0	.0%
	BMCFT	1	0	0	65	8	0	87.8%
	MCFT	0	0	0	22	15	0	40.5%
	BACFT	14	0	0	0	0	5	26.3%
	Overall Percent	21.8%	.0%	.0%	58.1%	14.4%	5.7%	47.6%

Table 10 shows the prediction of disease for both training sample and testing sample. It shows the comparison of percentage value of disease prediction with the percentage value for the actual of disease. From the table, we can observed that the highest percentage of correct disease is BMCFT for both training and testing sample which are 83.2% and 87.8% The second highest for correct percentage is JOCFT for both sample 53.8% and 50.0% . Other than that, The third highest of correct percentage for training sample is BACFT, 32.1%, while for testing sample is MCFT 40.5%. For the predicted classification of disease, we observe that BMCFT for both sample has the higest percentage of predicted disease. The second highest predicted is JOCFT for both sample. The third highest predicted is MCFT for both sample and the least percentage for predicted disease is BACFT. Hence for both observation of predicted percentage and correct percentage, we can conclude that, BMCFT has the highest percentage for both percentage compare to other disease. Thus, we can predict that BMCFT has the highest cases compare to other diseases.

5.5. Comparison RMSE performances

This part will discuss about the comparison of ANN result with the previous study of Genetic Algorithm (Rusli, 2014, Alias et al., 2014) based on the same problem of animal disease. See Table 11.

Table 11: Comparison of Performance Analysis between ANN and Genetic Algorithm

Measurement	ANN	Genetic Algorithm
Mean Square Error (MSE)	0.0756	0.325
Root Mean Square (RMSE)	0.275	0.570

Based on Table 11 above, the comparison MSE between ANN and genetic algorithm show the difference values of error. ANN has less value of error than the genetic algorithm means we can conclude that ANN is a superior method compare to method of genetic algorithm.

5.6. Comparison RMSE performances between software

Table 12 shows the comparison of ANN by using MATLAB and SPSS software. The MSE and RMSE show the differences of values for both MATLAB and SPSS where the SPSS has a smallest value of error compare to MATLAB means SPSS is the best software to use of neural network. See Table 12.

Table 12: Comparison of Performance Analysis between Software

Measurement	ANN MATLAB	ANN SPSS
Mean Square Error (MSE)	0.0756	0.05133
Root Mean Square (RMSE)	0.275	0.2266

6. Conclusion and future research

First of all, the objective of this study is to develop the ANN on some parameter identification that has been successfully developed which is disease, species and location. These two data of species and location are recognized as input data. While, the suspected disease data are observed as the target of output. Besides, the mathematical modeling of ANN is derived based on the architecture and method of ANN. The RMSE of ANN with the smallest value is obtained by using the implementation of programming of MATLAB. The best result of RMSE is obtained when we used 16 nodes in the hidden layer with a combination of transfer functions which is a tangent hyperbolic function in the hidden layer and linear function in the output layer. It is done after undergoing 20 times of trials for different nodes and different combination of transfer functions.

The implementation of ANN by using SPSS and programming of MATLAB is successfully built with the obtaining results that have been discussed in chapter 4. From the analysis obtained, goat is the species that has a lot of infected disease reported. While Johor is the location has most infected disease area. The main focuses of

this research is to predict the animal disease based on locations and species. Hence, the results obtained show the Brucellosis species which are BMCFT, it is the most infected disease detected at almost all locations and species. Last but not least, the result is comparing the RMSE of ANN method with RMSE of genetic algorithm with the same parameters of animal disease. The result shows that the RMSE of ANN has less value of the error which is 0.275 compare to the genetic algorithm which is 0.570. Hence, we can conclude that the ANN method is a superior method compare to genetic algorithm. Lastly, with the same parameter, 16 nodes in only one hidden layer, same transfer functions and other component, we can compare the RMSE value of ANN that implemented by using MATLAB with the RMSE of SPSS Neural Network. From the result obtain shows that SPSS Neural Network is the best software compare to ANN MATLAB with obtaining less value of MSE 0.05133 and RMSE 0.2266. The neural network technique is used to model non-linear problems and predict the output values for giving input parameters. Most of the infected diseases and the related veterinary cases are non-linear in nature and hence, neural network is finding application in veterinary. Since neural networks are known to be good at solving classification problems, it is not surprising if this research has been done in identification and classification. Therefore, the recommendation for further study of this animal disease, we suggest to add more parameter to get a more accurate result, such as the parameter of time and temperature, which is we can predict the future time of disease will occur and to predict a better temperature for habitat, hence to maintain the population of the species. Then, from the previous study of genetic algorithm, it is better if we use a huge data to obtain the optimal solution and it is desirable to maintain the population size as large as possible not only use a slight number of cases and parameter. Besides, the weakness GA method is using the limited data set in the previous study.

Acknowledgments

We would like to thank the Research Management Centre (RMC) at Universiti Teknologi Malaysia (UTM) for funding this project under grant number PY/2015/05399 / R.J130000.7826.4F795. We also would like to thank the Ibnu Sina Institute for Scientific and Industrial Research (ISI-SIR), Faculty of Science, Universiti Teknologi Malaysia for supporting this research.

References

[1] Al Shamisi, M. H., Assi, A. H., and Hejase, H. A. (2011). Using MATLAB to Develop Artificial Neural Network Models for Predicting Global Solar Radiation in Al Ain City-UAE. INTECH Open Access Publisher.

[2] Alias, N., Ghani, A. C. A., Saipan, H. F., Ramli, N., & Palil, S. Q. M. (2012, January). Wave equation for early detection of breast cancer growth using MATLAB Distributed Computing. In Enabling Science and Nanotechnology (ESciNano), 2012 International Conference on (pp. 1-1). IEEE.

[3] Alias, N., Saipol, H. F. S., & Ghani, A. C. A. (2014). Chronology of DIC technique based on the fundamental mathematical modeling and dehydration impact. Journal of food science and technology, 51(12), 3647-3657. https://doi.org/10.1007/s13197-012-0913-7.

[4] Gen, M., and Cheng, R. (1997). Genetic algorithms and engineering design. John Wily and Sons, New York.

[5] Krenker, A., Kos, A., & Bešter, J. (2011). Introduction to the Artificial Neural Networks. INTECH Open Access Publisher.

[6] Milhajlovic, I., Nikolic, Đ. Štrbac, N.,and Živkovic, Ž. (2010). Statistical modeling in ecological management using the artificial neural networks (ANNs). Serbian Journal of Management. 5(1):39-50. https://doi.org/10.5772/15751.

[7] Morens DM and Fauci AS (2013) Emerging Infectious Diseases: Threats to Human Health and Global Stability. PLoS Pathog 9(7): e1003467. https://doi.org/10.1371/journal.ppat.1003467.

[8] Nanda, S. K., Tripathy, D. P., Nayak, S. K., and Mohapatra, S. (2013). Prediction of Rainfall in India using Artificial Neural Network (ANN) Models. International Journal of Intelligent Systems and Applications (IJISA). 5 (12):1. https://doi.org/10.5815/ijisa.2013.12.01.

[9] Pintér, J. D. (2012). Calibrating Artificial Neural Networks by Global optimization. Expert Systems with Applications. 39(1):25-32. https://doi.org/10.1016/j.eswa.2011.06.050.

[10] Ramli N.A. (2014), Modeling Corrosion Rate in Gas Pipelines using Artificial Neural Network, Bachelor of Science (Mathematics).

[11] Rusli N.L. (2014), Mathematical Modeling of Veterinary Disease by using Genetic Algorithm, Bachelor of Science (Industrial Mathematics).

[12] Safi, Y., & Bouroumi, A. (2011). A neural network approach for predicting forest fires. In Multimedia Computing and Systems (ICMCS), 2011 International Conference on 1-5. IEEE. https://doi.org/10.1109/ICMCS.2011.5945716.

[13] Sahimi, M. S., Alias, N., & Sundararajan, E. (2001, May). The AGEB algorithm for solving the heat equation in three space dimensions and its parallelization using PVM. In International Conference on Computational Science (pp. 918-927). Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-45545-0_104.

[14] Seyam, M., and Mogheir, Y. (2011). Application of artificial neural networks model as analytical tool for groundwater salinity. Journal of Environmental Protection. 2(1):56. https://doi.org/10.4236/jep.2011.21006.

[15] Volna, E., Kotyrba, M., Kocian, V., and Janosek, M. (2012,). Cryptography Based On Neural Network. In ECMS. 386-391.

[16] Wijayasekara, D., Manic, M., Sabharwall, P., & Utgikar, V. (2011). Optimal artificial neural network architecture selection for performance prediction of compact heat exchanger with the EBaLM-OTR technique. Nuclear Engineering and Design, 241(7), 2549-2557. https://doi.org/10.1016/j.nucengdes.2011.04.045.