CLUSTERING PESTS OF RICE USING SELF ORGANIZING MAP

Mohd Noor Md Sap[1], Shafaatunnur Hasan[2]

Faculty of Computer Science and Information Systems
University Teknologi Malaysia
81300 Skudai, Johor

Email: [1]mohdnoor@utm.my, [2]shafaatunnur@gmail.com

Abstract: Rice, *Oryza sativa*, also called paddy rice, common rice, lowland and upland rice. This food grain is produced at least 95 countries around the globe, with China producing 36% of the world's production in 1999, followed by India at 21%, Indonesia at 8%, Bangladesh and Vietnam each producing about 5%. The United States produced about 1.5% of the world's accounts for about 15% of the annual world exports of rice. However the Modern agriculture is influenced by both the pressure for increased productivity and increased stresses caused by plant pests. Geographical Information Systems and Global Positioning Systems are currently being used for variable rate application of pesticides, herbicide and fertilizers in Precision Agriculture applications, but the comparatively lesser-used tools of Neural Network can be of additional value in integrated pest management practices. This study details spatial analysis and clustering using Neural Network based on Kohonen Self Organizing map (SOM) as applied to integrated agricultural rice pest management in Malaysia.

Keywords: Rice, Clustering, Neural Network, Kohonen Self Organizing Map (SOM)

1. INTRODUCTION

Rice cultivation in Peninsular Malaysia is nearly 383000ha in areas scattered over the 11 states. The rice bowl is the area under the Muda Irrigation Scheme on the north west coast. Here and in the states of Seberang Perai, Perak and Selangor, rice is grown extensively on lowland plains of marine alluvial clay. On the east coast in Kelantan and Terengganu rice is grown in the less fertile riverine clay soils. Apart from the coastal plains, paddy is cultivated in the flat narrow inland valleys of Melaka, Negeri Sembilan, Pahang and parts of Perak. Depending on the availability of water, rice in Peninsular Malaysia can be classified into 3

categories; the unirrigated, the partially irrigated and the fully irrigated (G.V Vreden et.al, 1986).

Parasites, predators and pathogens play a major role in the regulation of rice pests. Most parasites of rice pests belong to the order Hymenoptera and some few to Diptera. Egg parasites (mostly Hymenoptera) play a major role in limiting the growth of rice pests. A similar role, though to a lesser degree, is also played by larval, pupal and adult parasites. Major group of predators such as frogs, birds, and bats play a minor role (Bongiovanni et.al,2004). Predation need not to be confined to rice pests alone; beneficial species, if abundant, may also be attacked. When pray densities are low, spiders, dragonflies and damselflies become cannibalistic. Spiders have been known to eat their own offspring. The erratic feeding habits of predators make the assessment of their economic value difficult. The factors that play a role include: the capacity of the predator to feed and kill, its selectivity in this, but also to the ability to find the prey. The economic value of parasites is more easily determined, because of their more specific behaviour.The pathogens that attact insects include nematodes, viruses, bacteria and protozoa (Beeri et.al, 2006). Their importance as suppressing agents of rice pests have as yet received little attention. In Peninsular Malaysia, several pathogens have recently been identified.

The crop losses in 1979 an extensive outbreak of Surcifera occurred in the Muda Irrigation Scheme causing damage to an estimated 7163 ha, resulting in a loss of (MY) 1.5 million. However, the worst pest s of rice which caused considerable damage in almost all paddy fields in Malaysia is rat. Recently, losses at the national production level have been estimated to be around 7%, representing a monetary value of about (MY)6.2 million a year (MARDI, 2002). Otherwise, the estimations of overall crop losses due to the rice pests are complicated matter. The infestations differ from location to location and from season to season. In certain years there is hardly to mentioning of certain pests and then suddenly populations may build up without obvious reasons. The pests are sometimes thinly spread over large areas and in other occasions attacks are severe and localized. In all cases the loses result from from an accumulation of damage inflicted by one or a few major pests and many minor species. The species responsible and its share in the damage seems difficult to assess. However useful information about losses can be obtained by combining data from large-scale enquiries, sample surveys and field trials. In Peninsular Malaysia, the losses are from insects, birds and rats together to be between 10% and 15% (G.V Vreden et.al, 1986).

Spatial analysis can be a useful tool to explore the spatial distribution of pests, and help to formulate and test epidemiological hypothesis of pest establishment and spread (Groves et al., 2005; Perring et al., 2001; Wu et al., 2001). The co-occurance over space of pests and different aspects of hosts can help farmers and managers understand pest dynamics. Wu et al. (2001) applied geostatistical analyses to study the spatial variability of the lettuce downy

mildew in coastal California. The relatively short disease influence range, which was estimated by a semivariogram, suggested that the role of inoculum availability in the disease epidemics is less important than environmental variables. Perring et al. (2001) applied spatial analysis together with ANOVA analysis to study Pierce's disease (caused by the pathogen Xylella fastidiosa) in Temecula Valley, CA vineyards, and found that proximity to citrus orchards has influenced the incidence and severity of Pierce's disease. This was an important result, guiding potential management strategies for the vector of the disease, the glassy-wing sharpshooter (Homalodisca coagulata). In another study dealing with the same pathogen, but a different crop, Groves et al. (2005) used semivariograms to map the differing spatial pattern of almond leaf scorch over several different almond cultivars. Their results document both random and aggregate patterns of disease spatial distribution and illustrate how cultivar Susceptibility influences the distribution patterns of the disease (Groves et al., 2005).

2. CLUSTERING

Clustering is a data analysis technique that, when applied to a heterogeneous set of data items, produces homogenous subgroups as defined by a given model or measure of similarity or distance. It is an unsupervised process, where its job is to find any undefined or unknown clusters. In supervised learning method, there are some known clusters (groups), from which the algorithms learn the underlying relationship among the inputs and their corresponding outputs. In this way of learning, the model is developed and used for the prediction of target groups for new data elements whose groups are unknown.

For unsupervised scheme, there is no initial input and output relation. However, groups are only predicted from the input data. So, clustering can be thought of as an exploratory data analysis technique that can be used for the selection of diverse compound subsets and data reduction. Clustering as a methodology for partitioning of various types of datasets has been in use in almost all fields of social and technical sciences. However, the clustering tasks in research includes as a dimension –reduction tool when a data set has hundreds of attributes and for gene expression clustering, where very large quantities of genes may exhibit similar behavior. Clustering is often performed as preliminary step in a data mining process, with the resulting clusters being used as further inputs into a different technique downstream, such as neural networks. Due to the enormous size of many present-day databases, it is often helpful to apply clustering analysis first, to reduce the search space for the downstream algorithms.

Cluster analysis encounters many of the same issues in the classification. Researcher need to determine the similarity measure, recode categorical variables, standardize or normalize numerical variables and define the number of clusters (Jain, 1999).

3. KOHONEN SELF ORGANIZING MAP

Kohonen networks were introduced in 1982 by Finnish researcher Tuevo Kohonen. Although applied initially to image and sound analysis, Kohonen networks are an effective mechanism for clustering analysis. Kohonen networks represent a type of Self Organizing map (SOM), which itself represents a special class of neural network.

The goal of SOM is to convert a high dimensional input signal into a simpler low dimensional discrete .Thus, SOMs are nicely appropriate for cluster analysis, where underlying hidden patterns among records and fields are sought. SOM's structure the output nodes into cluster of nodes, where nodes in closer proximity are more similar to each other than to other nodes that are farther apart. Ritter had shown that SOMs represent a nonlinear generalization of principal component analysis, another dimension-reduction technique.

Self Organization Map are based on competitive learning, where the output nodes competes among themselves to be winning node (or neuron), the only node to be activated by a particular input observation. As (Haykin, 1999) describes it: "The neurons become selectively tuned to various input patterns (stimuli) or classes of input patterns in a course of competitive learning process". A typical SOM architecture is shown in figure 1. The input layer is shown at the bottom of the figure, with one input node for each field. Just as with neural networks, these input nodes do no processing themselves but simply pass the field input values along downstream

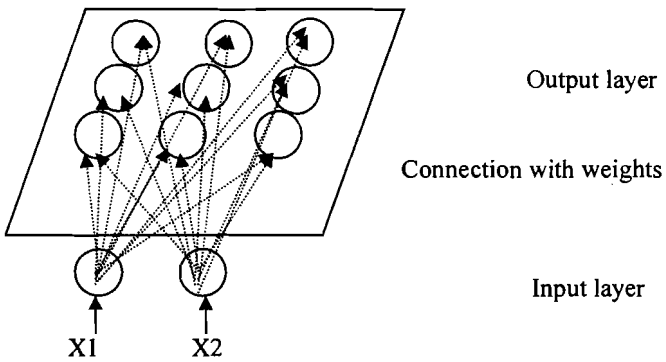

Figure 1: SOM Architecture

SOM are feedforward and completely connected. Feedforward networks do not allow looping or cycling. Completely connected means that every node in a given layer is connected to every node in the next layer, although not to other nodes in the same layer. Like neural networks, each connection between nodes have association with it, which at initialization is assigned randomly to a value between zero and one. Adjusting these weights represent the key for the learning mechanism in both neural networks and SOM. Variable values need to be

normalized or standardized, just for neural networks, so that certain variables do not overwhelm others in the learning algorithm.

Unlike most neural networks, however SOMs have no hidden layer. The data from the input layer is passed along directly to the output layer. The output layer is represented in the form of a lattice, usually in one or two dimension, and typically in the shape of a rectangle, although other shapes such as hexagons may be used. The output layer shows in figure 2.4 is a 3x3 square. Finally, SOM exhibit three characteristic processes which is competition, cooperation and adaptation.

3.1 Competition

The output nodes compete with each other to produce the best value for a particular scoring function, most commonly the Euclidean distance. In this case, the output node that has smallest Euclidean distance between the field inputs and the connection weights would be declared the winner.

3.2 Cooperative

The winning node therefore becomes the centre of a neighborhood of excited neurons. This emulates the behavior of human neurons, which are sensitive to the output of other neurons in their immediate neighborhood. In SOMs, all the nodes in the neighborhood share the adaptation given by the winning nodes. They tend to share common features, due to neighborliness parameter, even though the nodes in the output layer are not connected directly.

3.3 Adaptation

In the learning process, the nodes in the neighborhood of the winning node participate in adaptation. The weights of these nodes are adjusted so as to further improve in the score function. For a similar set of field values, these nodes will thereby have an increased chance of winning the competition once again.

3.4 Kohonen Network's Algorithm:

For each input vector x, do:

a) Initialization

   Set initial synaptic weights to small random values, say in a interval [0,1], and assign a small positive value to the learning rate parameter α.

b) Competition.

For each output node j, calculate the value D(wj,xn) of the scoring function. For example, for Euclidean distance, D(wj,xn)= $\sqrt{\sum_i (wij - xni)^2}$.

Find the winning node J that minimizes D(wj,xn) over all output nodes.

c) Cooperation.

Identify all output nodes j within the neighborhood of J defined by the neighborhood size R. For these nodes, do the following for all input records fields.

d) Adaptation

Adjust the weights:

Wij, new = Wij, current + η(xni-Wij, current)

Standard competitive learning rule (Haykin, 1999) defines the change $\triangle$ wij applied to synaptic weight wij as

$$\triangle \text{Wij} = \begin{cases} \alpha(x_i - wij) & \text{if neuron j wins the competition} \\ 0, & \text{if neuron j loses the competition} \end{cases}$$

Where xi is the input signal and α is the learning parameter. The learning rate parameter lies in the range between 0 and 1.

e) Iteration

Adjust the learning rate and neighborhood size, as needed until no change occur in the feature map. Repeat to step (b) and stop when the termination criteria are met.

4. TRAINING AND CLUSTERING SOM

The SOM consists of a regular, usually two-dimensional (2D), grid of map units. Each unit i is represented by a prototype vector $m_i = [m_{i1},\ldots,m_{id}]$, where d is input vector dimension. The units are connected to adjacent ones by a neighbourhood relation. The number of map units, which typically varies from a few dozen up to several thousand, determines the accuracy and generalization capability of the SOM. During training, the SOM forms an elastic net that folds onto the cloud formed by the input data. Data points lying near each other in the input space are mapped onto nearby map units. Thus, the SOM can be interpreted as a topology preserving mapping from input space onto the 2-D grid of map units.

The SOM is trained iteratively. At each training step, a sample vector x is randomly chosen from the input dataset. Distances between x and all the prototype vectors are computed. The best matching unit (BMU), which is denoted here by b, is the map unit with prototype closest to x

$$\|x-m_i\|=\min\{\|x-m_i\|\} \qquad (1.1)$$

Next, the prototype vectors are updated. The BMU and its topological neighbors are moved closer to the input vector in the input space. The update rule for the prototype vector of unit i is

$$M_i(t+1)=m_i(t)+\alpha(t)h_{bi}(t)[x-m_i(t)] \quad (1.2)$$

Where

t=time

$\alpha(t)$= adaptation coefficient

$h_{bi}(t)$=neighbourhood kernel centered on the winner unit

$$h_{bi}(t)=\exp\left(-\frac{\|r_b-r_i\|^2}{2\sigma^2(t)}\right) \quad (1.3)$$

where r_b and r_i are positions of neuron b and I on the SOM grid. Both $\alpha(t)$ and $\sigma(t)$ decrease monotonically with time. There is also a batch version of the algorithm where the adaptation coefficient is not used.

In the case of a discrete data set and fixed neighbourhood kernel, the error function of SOM can be shown to be

$$E=\sum_{I=1}^{N}\sum_{J=1}^{M} h_{bj}\|x_i-m_j\|^2 \quad (1.4)$$

Where N is number of training samples, and M is the number of map units. Neighborhood kernel h_{bj} is centered at unit b, which is the BMU of vector x_i and evaluated for unit j. If neighborhood kernel value is one for the BMU and zero elsewhere, the SOM reduces to adaptive k-means algorithm. If this is not the case, from (1.4), it follows the prototype vectors are not in the centroid of their Voronoi sets but are local averages of all vectors in the dataset weighted by neighbourhood function values.

A SOM was trained using the sequential training algorithm for each data set. All maps were linearly initialized in the subspace spanned by the two eigenvectors with greatest eigenvalues computed from the training data. The maps were trained in two phases: a rough training with large initial neighbourhood width and learning rate and fine-tuning phase with small initial neighbourhood width and learning rate. The neighbourhood width decreased linearly to 1,the neighbourhood function was Gaussian. The training length of the two phases were 3 and 10 epochs, and the initial learning rate decreased linearly to zero during the training.

The SOM consists of a regular, usually two-dimensional (2D), grid of map units. Each unit i is represented by a prototype vector $m_i = [m_{i1},\ldots,m_{id}]$, where d is input vector dimension. The units are connected to adjacent ones by a neighbourhood relation. The

number of map units, which typically varies from a few dozen up to several thousand, determines the accuracy and generalization capability of the SOM. During training, the SOM forms an elastic net that folds onto the cloud formed by the input data. Data points lying near each other in the input space are mapped onto nearby map units. Thus, the SOM can be interpreted as a topology preserving mapping from input space onto the 2-D grid of map units.

The SOM is trained iteratively. At each training step, a sample vector x is randomly chosen from the input dataset. Distances between x and all the prototype vectors are computed. The best matching unit (BMU), which is denoted here by b, is the map unit with prototype closest to x

$$\|x-m_i\|=\min\{\|x-m_i\|\} \quad (1.1)$$

Next, the prototype vectors are updated. The BMU and its topological neighbors are moved closer to the input vector in the input space. The update rule for the prototype vector of unit i is

$$M_i(t+1)=m_i(t)+\alpha(t)h_{bi}(t)[x-m_i(t)] \quad (1.2)$$

Where

t=time

$\alpha(t)$= adaptation coefficient

$h_{bi}(t)$=neighbourhood kernel centered on the winner unit

$$h_{bi}(t)=\exp\left(-\frac{\|r_b-r_i\|^2}{2\sigma^2(t)}\right) \quad (1.3)$$

where r_b and r_i are positions of neuron b and I on the SOM grid. Both $\alpha(t)$ and $\sigma(t)$ decrease monotonically with time. There is also a batch version of the algorithm where the adaptation coefficient is not used.

In the case of a discrete data set and fixed neighbourhood kernel, the error function of SOM can be shown to be

$$E=\sum_{I=1}^{N}\sum_{J=1}^{M} h_{bj}\|x_i-m_j\|^2 \quad (1.4)$$

Where N is number of training samples, and M is the number of map units. Neighborhood kernel h_{bj} is centered at unit b, which is the BMU of vector x_i and evaluated for unit j. If neighborhood kernel value is one for the BMU and zero elsewhere, the SOM reduces to adaptive k-means algorithm. If this is not the case, from (1.4), it follows the prototype vectors are not in the centroid of their Voronoi sets but are local averages of all vectors in the dataset weighted by neighbourhood function values.

A SOM was trained using the sequential training algorithm for each data set. All maps were linearly initialized in the subspace spanned by the two eigenvectors with greatest eigenvalues computed from the training data. The maps were trained in two phases: a rough training with large initial neighbourhood width and learning rate and fine-tuning phase with small initial neighbourhood width and learning rate. The neighbourhood width decreased linearly to 1,the neighbourhood function was Gaussian. The training length of the two phases were 3 and 10 epochs, and the initial learning rate decreased linearly to zero during the training.

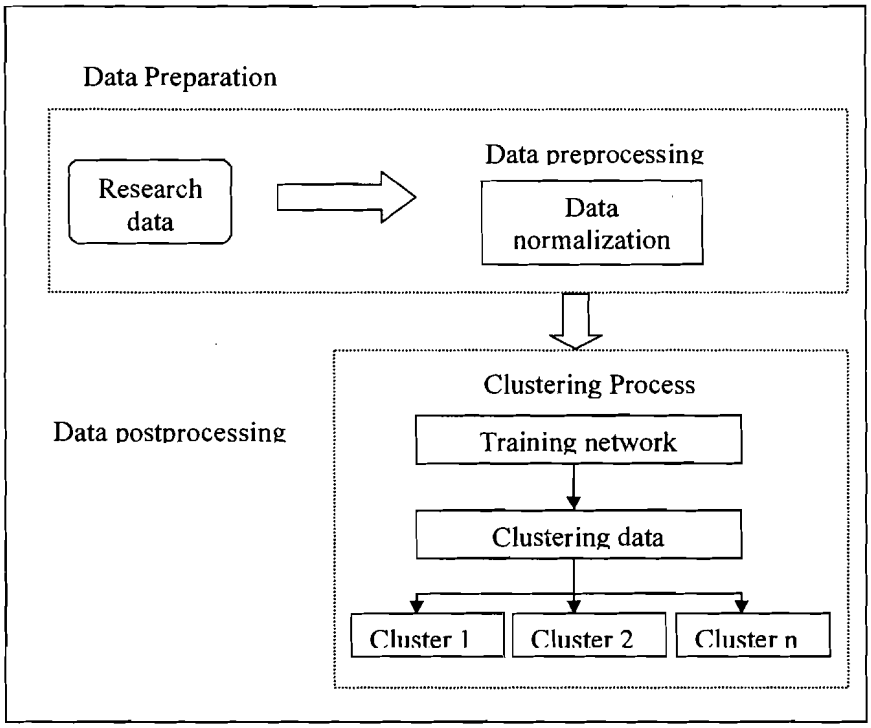

Figure 2: Clustering Process

5. DATA PREPARATION

The data (figure 2) were collected from Muda Agricultural Development Authority (MADA), Kedah, Malaysia ranging from 1996 to 1998. There are 4 areas with 27 locations. With two planting season for each year, total of 6 seasons is generated. There are 35 parameters that affect the rice yield. The parameters were classified to 5 groups. There are 3 types of weed; *rumpai*, *rusiga* and *daun lebar*, 3 types of pests; rats, type of worms and *bena perang*, 3 types of diseases; bacteria (*blb & bls*), *jalur daun merah* (*jdm*) *and hawar seludang*, one type of lodging and one type of wind paddy, making a total 11 input parameters. Out of 35 parameters, only 11 parameters are chosen since these are the most significant ones that

were recommended by the domain expert from MADA. For this study, the experiments are focus on clustering rice of pest.

6. EMPIRICAL ANALYSIS

Neural Network based clustering using Kohonen Self Organizing Map (SOM) with 2Dimensional and 10x10 lattice square neuron is applied in this study. There are 27 number of observation, 11 number of variables, 10 neurons, 1000 times learning cycle, learning parameter start from 0.9 to 0.1 and Sigma for the Gaussian Neighborhood as percentage map width start from 50 to 1. The learning parameter and Gaussian Neighborhood used Exponential Decay. After learning process, the results are shown in Table 1 and Figure 3.

Table 1: Location of each clusters

LOCATION	M196	M296	M197	M297	M198	M298
1(A1)	4	12	3	4	3	4
2(B1)	6	7	7	9	7	1
3(C1)	8	10	7	9	7	7
4(D1)	4	5	3	4	3	4
5(E1)	1	4	8	6	7	7
6(A2)	3	6	4	5	4	5
7(B2)	3	9	6	7	8	10
8(C2)	2	9	6	7	5	6
9(D2)	6	6	4	5	4	5
10(E2)	3	1	6	7	5	6
11(F2)	1	9	6	7	5	6
12(G2)	4	12	3	4	3	4
13(H2)	6	7	3	4	3	4
14(I2)	3	3	1	2	9	9
15(A3)	4	12	3	4	3	4
16(B3)	4	5	5	10	6	3
17(C3)	6	12	7	9	7	7
18(D3)	1	10	3	4	3	4
19(E3)	4	5	2	4	3	4
20(F3)	5	6	4	5	8	8
21(A4)	4	5	3	4	3	4
22(B4)	7	2	8	8	2	3
23(C4)	4	8	2	3	3	2
24(D4)	6	6	4	5	4	5
25(E4)	7	6	4	5	1	5
26(F4)	7	2	4	5	4	5
27(G4)	7	11	1	1	6	3

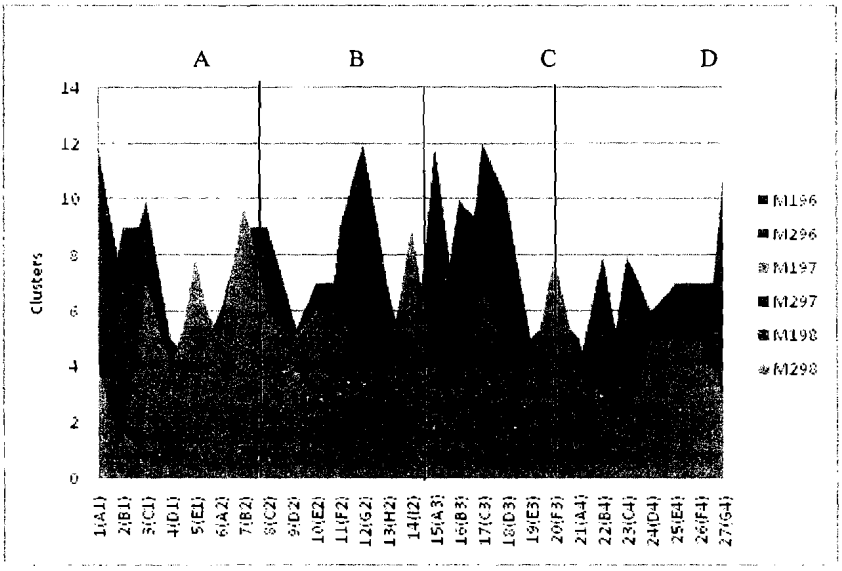

Figure 3: Location of each clusters

Table 1 and Figure 3 above present the clusters of rice parameters that affect rice yield by location. There are 4 locations which are A(A1 to E1),B(A2 to I2),C(A3 to F3) and D(A4 to G4). M196 to M298 are the season start from 1996 to 1998. In season 1, 1996, the parameter affect rice yield is type of pests in most of the location A and type of weeds in location D. In season 2, 1996, location B mostly infected by type of weeds and location D is bacteria (*blb and bls*) . Otherwise, season 1 and 2, 1997 and 1998, shows that all locations mostly infected by types of pests and weeds. This results proof that pests such as rats, type of worms and *bena perang* is one of the factor that effect rice production in MADA. The next experiment implements the specific type of pests for further analysis.

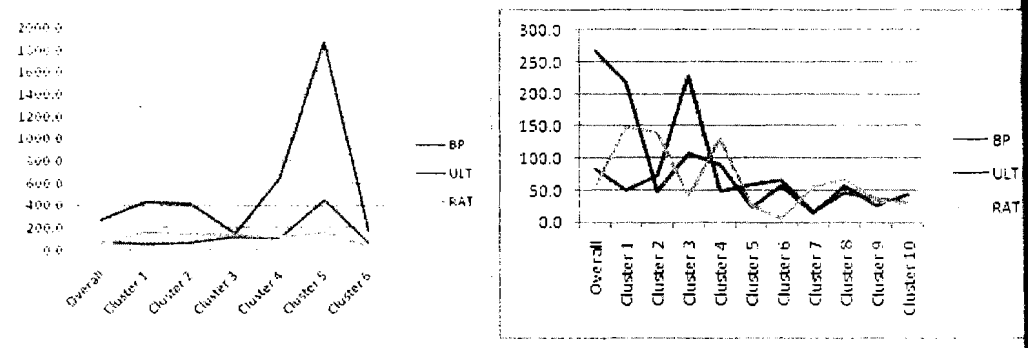

Figure 4: Cluster Means for Season1 and Season2 1996

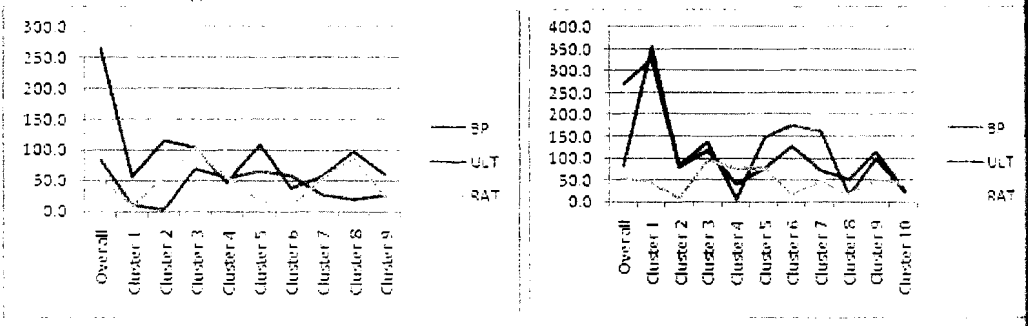

Figure 5 Cluster Means for Season1 and Season2

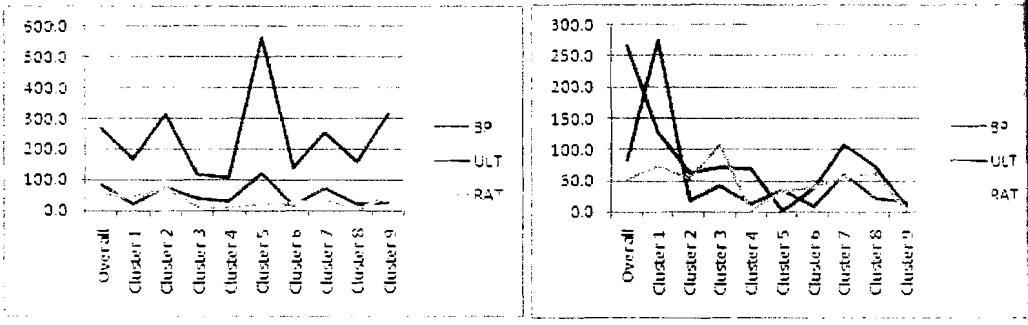

Figure 6: Cluster Means for Season1 and Season2

Figure 4, Figure 5 and Figure 6 above present the Cluster Means of rice pests. There are 3 types of pests parameter involves in this study which is BP for *bena perang*, ULT for type of worms and RAT for mouse. For most of the season, ULT have high range of Cluster Means. For season2 of each year, BP takes part in the high range of the Cluster Means while RAT is the lowest rate of Cluster Means with not more than 200.

6. CONCLUSION AND FUTURE WORK

Pests and weeds are the major factor of the rice yield losses in Malaysia. Clustering neural network based on Kohonen SOM can be successful applied in spatial analysis for

Integrated Pest Management (IPM). The future work will focus on spatial analysis and intelligent tools in machine learning for rice yield prediction. The transition to the utilization of a full suite of geospatial tools for integrated pest management is agricultural sector is mirrored in the realm of forestry, where increasing and large-scale pest and disease attacks are increasingly reported, and where the spatial pattern across landscape-scales of pest hosts, pest and pathogen population dynamics and landscape structure interact to at times promote pest establishment (Holdenrieder *et al.*, 2004). As in agricultural settings, geospatial technologies are making forestry management more precise and spatially comprehensive: and a better articulation of resources across space yields new insights to yield, pest and control dynamics. New access to data and technology will likely promote the transition of these tools from a research to an applied domain across both sectors.

ACKNOWLEDGEMENTS

This research was supported by the Research Management Center, University Technology Malaysia (UTM) and the Malaysian Ministry of Science, Technology and Innovation (MOSTI) under vote number 79094.

REFERENCES

Beeri, O., & Peled, A. (2006). Spectral indices for precise agriculture monitoring. *International Journal of Remote Sensing*, 27, 2039-2047.

Bongiovanni, R., & Lowenberg-DeBoer, J. (2004). Precision agriculture and sustainability. *Precision Agriculture*, 5, 359-387.

Groves, R. L., Chen, J., Civerolo, E. L., Freeman, M. W., & Viveros, M. A. (2005). Spatial analysis of almond leaf scorch disease in the San Joaquin Valley of California: factors affecting pathogen distribution and spread. *Plant Disease*, 89, 581-589.

G.V Vreden,Abdul Latif Ahmad Zabidi (1986). *Pest of rice and their natural enemies in Peninsular Malaysia.* Pudoc Wagenigen.

Haykin, S.(1999)*Neural Networks : A Comprehensive Foundation.* (2nd ed). Upper Saddle River, New Jersey.

Holdenrieder, O., Pautasso, M., Weisberg, P. J., & Londsdale, D. (2004). Tree diseases and landscape processes: the challenge of landscape pathology. *Trends in Ecology and Evolution*, 19, 446-452.

Jain, A. K., Murty, M.N. and Flynn, P. J. (1999). Data Clustering : A Review. *ACM Computing Surveys, Vol. 31 (3).* 264-362.

MARDI (2002), "Manual Penanaman Padi Berhasil Tinggi Edisi 1/2001", Malaysian Agriculture research and Development Institute. Malaysian Ministry of Agriculture.

Perring, T. M., Farrar, C. A., & Blua, M. J. (2001). Proximity to citrus influences Pierce's disease in the Temecula valley. *California Agriculture*, 55, 13-18.

Sudduth, K., Fraisse, C., Drummond, S. and Kitchen, N. (1996), "Analysis of Spatial Factors Influencing Crop Yield", in Proc. Of Int. Conf. on Precision Agriculture, pp. 129-140.

Wu, B. M., Bruggen, A. H. C. V., Subbarao, K. V., & Pennings, G. G. H. (2001). Spatial analysis of lettuce downy mildew using geostatistics and geographic information systems. *Phytopathology*, 91,134-142.