

การบูรณาการโครงข่ายใยประสาทเทียมและระบบสารสนเทศภูมิศาสตร์เพื่อการ  
พยากรณ์ผลผลิตทางการเกษตร : กรณีศึกษาผลผลิตลำไยในพื้นที่จังหวัด  
เชียงใหม่และลำพูน ประเทศไทย

นางสาวกาญจนา ทองบุญนาค

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรดุษฎีบัณฑิต  
สาขาวิชาภูมิสารสนเทศ  
มหาวิทยาลัยเทคโนโลยีสุรนารี  
ปีการศึกษา 2551

**INTEGRATION OF ARTIFICIAL NEURAL NETWORK  
AND GEOGRAPHIC INFORMATION SYSTEM FOR  
AGRICULTURAL YIELD PREDICTION : CASE STUDY  
OF LONGAN YIELD IN CHIANGMAI AND LAMPHUN,  
THAILAND**

**Kanchana Thongboonnak**

**A Thesis Submitted in Partial Fulfillment of the Requirements for the  
Degree of Doctor of Geoinformatics  
Suranaree University of Technology  
Academic Year 2008**

**INTEGRATION OF GEOGRAPHIC INFORMATION SYSTEM  
AND ARTIFICIAL NEURAL NETWORK FOR AGRICULTURAL  
YIELD PREDICTION : CASE STUDY OF YIELD PREDICTION  
OF LONGAN IN CHIANGMAI AND LAMPHUN**

Suranaree University of Technology has approved this thesis submitted in  
partial fulfillment of the requirements for a Degree of Doctor of Philosophy.

Thesis Examining Committee

\_\_\_\_\_  
(Dr.Suwit Ongsomwang)

Chairperson

\_\_\_\_\_  
(Asst.Prof.Dr.Sunya Sarapirome)

Member (Thesis Advisor)

\_\_\_\_\_  
(Asst.Prof.Dr.Songkot Dasananda)

Member

\_\_\_\_\_  
(Assoc.Prof.Dr.Charlie Navanugraha)

Member

\_\_\_\_\_  
(Assoc.Prof.Dr.Sura Pattanakaet)

Member

\_\_\_\_\_  
(Prof. Dr. Pairote Sattayatham)

Vice Rector for Academic Affairs

\_\_\_\_\_  
(Assoc. Prof. Dr. Prapan Manyam)

Dean of Institute of Science

กาญจนา ทองบุญนาค : การบูรณาการ โคร่งข่ายใยประสาทเทียมและระบบสารสนเทศ  
ภูมิศาสตร์เพื่อการพยากรณ์ผลผลิตทางการเกษตร: กรณีศึกษาผลผลิตลำไยในพื้นที่จังหวัด  
เชียงใหม่และลำพูน ประเทศไทย (INTEGRATION OF ARTIFICIAL NEURAL NETWORK  
AND GEOGRAPHIC INFORMATION SYSTEM FOR AGRICULTURAL YIELD  
PREDICTION : CASE STUDY OF LONGAN YIELD IN CHIANGMAI AND LAMPHUN,  
THAILAND) อาจารย์ที่ปรึกษา : ผู้ช่วยศาสตราจารย์ ดร.สัญญา สราภิรมย์, 123 หน้า

งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาโปรแกรมคอมพิวเตอร์ โดยใช้หลักการของโครงข่ายใยประสาทเทียม เพื่อใช้ในการพยากรณ์ผลผลิตทางการเกษตร โดยพัฒนาให้อยู่ในรูปของโปรแกรมเสริมความสามารถของโปรแกรม ArcMap และทดลองใช้กับการพยากรณ์ผลผลิตลำไยในพื้นที่จังหวัดเชียงใหม่และลำพูน การพัฒนาโปรแกรมโครงข่ายใยประสาทเทียมครอบคลุมสถาปัตยกรรมของโครงข่ายทั้งแบบ 1 ชั้นซ่อนตัว และ 2 ชั้นซ่อนตัว โดยใช้เทคนิคของการออกแบบและพัฒนาโปรแกรมเชิงวัตถุ โปรแกรมภาษาที่ใช้คือ VB.NET ข้อมูลที่ใช้เป็นข้อมูลนำเข้าให้กับโครงข่ายใยประสาทเทียมได้แก่ พื้นที่ครอบคลุมของกลุ่มชุดดิน และข้อมูลสภาพอากาศของปี พ.ศ. 2549 ถึงปี พ.ศ. 2551 ซึ่งส่งผลต่อปริมาณผลผลิตลำไยในรอบปี 2550 และ 2551 ข้อมูลทั้งหมดถูกปรับช่วงข้อมูลให้อยู่ในช่วงเดียวกันคือ 0-1 เพื่อให้เหมาะกับการนำเข้าสู่โครงข่ายใยประสาทเทียม ข้อมูลรายสัปดาห์ของอุณหภูมิสูงสุด อุณหภูมิต่ำสุด และอุณหภูมิเฉลี่ย ข้อมูลปริมาณแสงแดดเฉลี่ย และปริมาณน้ำฝนเฉลี่ยที่ได้หลังจากการปรับช่วง จะถูกนำมาผ่านกระบวนการประมาณค่า (interpolate) ข้อมูลที่ประมาณค่าแล้วจะนำมาเฉลี่ยเป็นรายอำเภอในพื้นที่ศึกษาซึ่งมีการปลูกลำไยเพื่อให้สอดคล้องกับข้อมูลผลผลิตที่มีการบันทึกเป็นรายอำเภอเช่นกัน นำข้อมูลเหล่านี้มาใช้ประกอบกันสร้างเป็นรูปแบบข้อมูลนำเข้าที่หลากหลายเพื่อหารูปแบบที่ให้ผลลัพธ์ที่ดีที่สุด โดยการทำการสวาลิเดชัน (cross-validation) สำหรับทุกรูปแบบ พารามิเตอร์ที่เหมาะสมสำหรับการฝึกสอนและทดสอบแบบจำลอง ใช้ค่าที่ได้จากการทดลองหาค่าที่ให้ผลลัพธ์ที่ดีที่สุด ได้แก่ ค่าอัตราการเรียนรู้คือ 0.4 จำนวนโหนดชั้นซ่อนตัวคือ 6 โหนด และจำนวนรอบสำหรับการฝึกสอนคือ 3,000 รอบ ผลการศึกษาพบว่า การพยากรณ์โดยใช้สถาปัตยกรรมโครงข่ายแบบ 1 ชั้นซ่อนตัว ให้ค่าความแม่นยำ 90.14% และการพยากรณ์โดยใช้สถาปัตยกรรมโครงข่ายแบบ 2 ชั้นซ่อนตัว ให้ค่าความแม่นยำ 88.50% สามารถสรุปได้ว่าโปรแกรมที่พัฒนาขึ้นทำงานได้ภายใต้สภาพแวดล้อมของโปรแกรม ArcMap และสามารถพยากรณ์ผลผลิตลำไยได้ในเกณฑ์ดี

สาขาวิชาการรับรู้จากระยะไกล

ปีการศึกษา 2551

ลายมือชื่อนักศึกษา \_\_\_\_\_

ลายมือชื่ออาจารย์ที่ปรึกษา \_\_\_\_\_

KANCHANA THONGBOONNAK : INTEGRATION OF ARTIFICIAL  
NEURAL NETWORK AND GEOGRAPHIC INFORMATION SYSTEM FOR  
AGRICULTURAL YIELD PREDICTION: CASE STUDY OF LONGAN YIELD  
IN CHIANGMAI AND LAMPHUN, THAILAND. THESIS ADVISOR : ASST.  
PROF. SUNYA SARAPIROME, Ph.D. 123 PP.

#### GIS/ ARTIFICIAL NEURAL NETWORK/ LONGAN/ YIELD PREDICTION

The main objective of this study was to develop the ANN modules for agricultural yield prediction as an extension of the ArcMap software. The modules include 1-hidden-layer and 2-hidden-layer architecture of the ANN model. The Object-Oriented technique was used for both designing and programming. VB.NET is a language applied to coding. The ANN modules developed were tested with longan yield prediction in Chiangmai and Lamphun. ANN input data are area extent of soil group and the climate data of years 2006-2008, which relate to longan yield in year 2007 and 2008. All data were normalized into the same range of 0-1 to be suitable as input of the ANN model. The normalized weekly highest, lowest and average temperature, average sunlight and rainfall were interpolated. They were then averaged to spatially represent Amphoes in the study area, which were corresponding to longan yield of Amphoes. These data were varied to be several input variations. The cross validation process was applied to each variation. The optimal parameters including learning rate, number of nodes in the hidden layer, and number of iteration obtained from testing were 0.4, 6, and 3,000 respectively. These parameters were applied for all training and testing processes. The result found that the best accuracies obtained

from applying 1-hidden-layer and 2-hidden-layer ANN modules were 90.14% and 88.50% respectively. The ANN modules developed can work under ArcMap environment very well for longan yield prediction with accurate results despite limitation of data set.

School of Remote Sensing

Academic Year 2008

Student's Signature\_\_\_\_\_

Advisor's Signature\_\_\_\_\_

## **ACKNOWLEDGEMENT**

This work is the result of study which takes many years to complete. By that time, I have worked with a great number of people whose contribution in many ways to the research. It is a pleasure to convey my gratitude to them all.

In the first place I would like to express profound gratitude to my advisor, Asst. Prof. Dr. Sunya Sarapirome for his supervision, advice, and guidance from the very early stage of this research. He provided me his precious time to discuss, continued encouragement and invaluable suggestions during this work. He is also helping me with the editing of the thesis. I am indebted to him more than he knows.

I am also very grateful to Dr. Suwit Ongsomwang for his useful suggestion about ANN and excellent comments. Sincere thanks are extended to Asst. Prof. Dr. Songkot Dasananda for his supporting and helping me. I also have to thank the members of my Ph.D. committee, Assoc. Prof. Dr. Charlie Navanugraha and Assoc. Prof. Dr. Sura Pattanakiat for their helpful advice and suggestions in general.

I would also like to thank the GISTDA, who support for remote sensing data.

My special thanks also go to Chiangmai Rajabhat University (CMRU) for providing me this opportunity with financial support.

To complete the thesis, I must express my special thanks to Mrs. Sirilak Tanang, Mr. Wipop Paengwongthong and all of my friends at School of Remote Sensing, Suranaree University of Technology for helping me and be good friends for long times.

I am deeply, specially thanks to Ms. Thirada Suwannachai, Ms. Parichat Suwanma, Mr. Sangchai Trongcharoensuk and my companies at Chiangmai province, for extremely encouragement and supporting me to complete the thesis.

A very special word of thanks goes for my family, who has given me the extra strength and motivation to get things done. This thesis is dedicated to them.

Kanchana Thongboonnak



# CONTENTS

	<b>Page</b>
ABSTRACT IN THAI.....	I
ABSTRACT IN ENGLISH.....	II
ACKNOWLEDGEMENT.....	IV
CONTENTS.....	VI
LIST OF TABLES.....	IX
LIST OF FIGURES.....	XI
LIST OF ABBREVIATIONS.....	XVII
<b>CHAPTER</b>	
<b>I INTRODUCTION.....</b>	<b>1</b>
1.1 Background problem of the study.....	1
1.2 Objective of the study.....	5
1.3 Scope and limitation.....	5
1.4 Study area.....	6
<b>II LITERATURE REVIEW.....</b>	<b>11</b>
2.1 Previous studies on yield prediction.....	11
2.2 Artificial Neural Network Theory.....	14
2.3 Application of ANN to crop yield prediction.....	17
2.4 Factors related to the longan blossom and fruit producing.....	18
2.4.1 The factors related to the longan trunk.....	21
2.4.2 The factors related to environment.....	21
2.5 ANN module development.....	23

## CONTENTS (Continued)

	<b>Page</b>
<b>III METHODOLOGY</b> .....	26
3.1 Conceptual overview of methods of the study .....	26
3.2 Data collection .....	27
3.2.1 Temperature, sunlight, and rainfall.....	27
3.2.2 Land-use and Amphoe data .....	28
3.2.3 Soil data .....	28
3.2.4 Longan yield data .....	28
3.3 Data preparation and manipulation.....	28
3.3.1 Temperature, sunlight, and rainfall data .....	28
3.3.2 Soil data and longan planting area of each Amphoe in the study area .....	29
3.3.3 Longan yield data .....	30
3.4 ANN module development .....	30
3.4.1 Analysis and design .....	31
3.4.2 Implementation .....	33
3.5 Training and testing .....	37
3.5.1 Variation of model parameters .....	31
3.5.2 Cross validation .....	39
3.5.3 Input data variation for cross validations.....	41
3.6 Prediction.....	44

## CONTENTS (Continued)

	<b>Page</b>
<b>IV RESULTS AND DISCUSSION</b> .....	45
4.1 Results of the study.....	45
4.1.1 Results of Data Collection, Preparation, and Manipulation .....	45
4.1.2 ANN Module Development.....	55
4.1.3 Results of Training and Testing.....	60
4.1.4 Prediction.....	62
4.2 Results discussion.....	68
4.2.1 Varying input factors .....	68
4.2.2 Varying number of weekly data .....	69
4.2.3 Varying number of record used .....	71
4.2.4 Comparing predicted yields from 1- hidden-layer and 2-hidden-layer modules with actual yields .....	71
<b>V CONCLUSION</b> .....	76
5.1 Conclusions .....	76
5.2 Recommendations and further improvements .....	79
<b>REFERENCES</b> .....	80
<b>APPENDICES</b> .....	87
APPENDIX A   ARTIFICIAL NEURAL NETWORK THEORY.....	88
APPENDIX B   SOFTWARE USER MANUAL .....	103
APPENDIX C   EXAMPLE OF SOURCE CODE .....	115
<b>CURRICULUM VITAE</b> .....	123

## LIST OF TABLES

<b>Table</b>	<b>Page</b>
1.1	Examples of researches on yield prediction of various crops ..... 2
1.2	The statistics of fresh longan export..... 3
2.1	The types of crops and factors used in yield prediction. .... 12
2.2	Factors influencing yield of Longan. .... 20
2.3	Longan crop calendar..... 23
2.4	Examples of researches using ANN and own-development programs..... 23
2.5	Some neural network software available at present..... 24
3.1	Data layers applied in the present study ..... 27
3.2	The tests with varying LR. The LRbest will be obtained from the tests ..... 37
3.3	Tests to choose the best number of hidden nodes. Iterations and the best LR are kept constant. .... 38
3.4	Tests to choose the best number of iterations. The best number of hidden nodes and LR are kept constant. .... 38
3.5	A number of variations of input data for 1-hidden-layer module..... 42
3.6	A number of variations of input data for 2-hidden-layer module..... 43
4.1	Example of normalized temperature, sunlight, and rainfall data of week N1 and N2 of each Amphoe ..... 52
4.2	Longan planting areas of each Amphoe within the study area. .... 53
4.3	Normalized area extents of soil group in longan planting areas of each Amphoe in the study area ..... 54
4.4	Longan yields of year 2007 and 2008 and their normalization ..... 55

## LIST OF TABLES (Continued)

<b>Table</b>	<b>Page</b>
4.5 Results of 12 tests to obtain the best model parameters which are LR, hidden nodes, and a number of iterations. ....	61
4.6 Results of cross validations of all variations of input data for 1-hidden-layer module .....	63
4.7 Results of cross validations of all variations of input data for 2-hidden-layer module .....	64
4.8 The input data of the year 2009 for the model for prediction.....	65
4.9 Difference of predicted yield and actual yield for two years using 1-hidden-layer module .....	66
4.10 Difference of predicted yield and actual yield for two years using 2-hidden-layer module .....	67

## LIST OF FIGURES

<b>Figure</b>	<b>Page</b>
1.1 The study area covering Amphoes in Chiangmai and Lumpun Provinces.....	7
1.2 Longan when they are being blossomed .....	8
1.3 Longan when they are providing fruits .....	8
1.4 Landsat 7 TM image covering the study area with longan plantation areas .....	9
1.5 Cultivation characteristic in rows of longan even in the big area can be hardly observed in the TM image.....	10
1.6 Typical pattern characteristic in rows of longan planting area is quite obviously observable in QuickBird image. ....	10
2.1 A diagram showing workflow of a multiple-input neuron. ....	15
2.2 Multilayer Network with Back-Propagation learning rule. ....	17
3.1 The flow chart of methods of the study. ....	26
3.2 The steps for the module development. ....	30
3.3 The software prototype or interface design of ANN model for longan prediction. ....	31
3.4 Class Diagram of Neural Network for Prediction Module.....	32
3.5 The steps to implement ArcMap extension module using VB.NET .....	33
3.6 The flowchart of Training function .....	35
3.7 The flowchart of Testing function .....	36
4.1 Meteorological stations distribution within and surrounding the study area .....	46

## LIST OF FIGURES (Continued)

<b>Figure</b>	<b>Page</b>
4.2	Example of raw data in .txt file of temperature, sunlight, and rainfall collected from a HAI station. ....47
4.3	Data layer of minimum temperatures of week N1 .....49
4.4	Data layer of maximum temperatures of week N1 .....50
4.5	Data layer of average temperatures of week N1 .....50
4.6	Data layer of average sunlight of week N1.....51
4.7	Data layer of average rainfall of week N1 .....51
4.8	Longan planting areas in selected Amphoes of Chiangmai and Lamphun ....53
4.9	GISANN icon as plugged into ArcMap of ArcGIS 9.1 .....56
4.10	User interface of “Training” tab of GISANN extension and other four tabs in the back.....57
4.11	“Training Result” tab .....57
4.12	User interface of “Testing” tab .....58
4.13	User interface of “Predict” tab.....59
4.14	User interface of “Config” tab .....60
4.15	Forecast performance of 1-hidden-layer module during year 2007 and 2008.....66
4.16	Forecast performance of 2-hidden-layer module during year 2007 and 2008 .....67
4.17	Shows the comparative accuracy of the different chosen inputs .....69
4.18	The comparative accuracy of the different weekly data used of 1-hidden-layer testing .....70

## LIST OF FIGURES (Continued)

<b>Figure</b>	<b>Page</b>
4.19	The comparative accuracy of the different weekly data used of 2-hidden-layer testing .....70
4.20	Shows the average accuracy of the data records.....71
4.21	(a) shows accuracy comparison of both modules according to data of the year 2007 and 2008 and in different Amphoes; (b) show accuracy comparison of both modules when accuracy of both year are averaged .....73
4.22	shows accuracy comparison of both modules according to average data of variation number 16/1, 17/2, 18/3, 19/16, 20/17, and 21/18 in form of the comparison of variation number in Table3.5/variation number in Table 3.6 .....74
4.23	shows yield comparison of both modules and the actual yields according to different Amphoes and years of data .....75
4.24	shows yield comparison of both modules and the actual yields according to different Amphoes and years of data when no need to be the same variation number is considered .....75
A.1	Artificial neuron cell .....89
A.2	Artificial neuron cell made up along with activation function .....90
A.3	The graph from Sigmoid Logistic Function and Derivative.....91
A.4	The Single Layer Artificial Neural Network Model.....92
A.5	The multilayer artificial neural network .....93
A.6	The connection in the network of two-layer back propagation .....97



## LIST OF FIGURES (Continued)

<b>Figure</b>		<b>Page</b>
A.7	The weight number adapted structure in output and hidden layer.....	99
A.8	The training weight adjustment using back propagation algorithm .....	102
B.1	Show the step to open Customize dialog box .....	104
B.2	Show the Customize dialog box .....	104
B.3	Show the Commands tap and Add from file button .....	105
B.4	Show the Added objects window.....	105
B.5	The GISANN in Custom Control category.....	106
B.6	The GISANN icon on the toolbar .....	106
B.7	The GISANN window .....	108
B.8	Show the user interface of Training tab.....	109
B.9	Show the message box after Training .....	110
B.10	Show Training Result window .....	110
B.11	Show the user interface of Testing tab. ....	111
B.12	Show the user interface of Predict tab. ....	112
B.13	Show the user interface of Config tab. ....	113

## **LIST OF ABBREVIATIONS**

ANN	Artificial Neural Network
DLL	Dynamic Linked Library
GIS	Geographic Information System
HAI	Hydro and Agro Information Institute
LDD	Land Development Department
LR	Learning Rate
RMSE	Root Mean Square Error
RSE	Root Square Error
UML	Unified Modeling Language

# CHAPTER I

## INTRODUCTION

### 1.1 Background problem of the study

The agricultural yield prediction is important for an agricultural country like Thailand. The prediction result could be used for agricultural and commercial organizations to decide how production and market should be managed and planned. Normally, the conventional prediction method has been based on field survey to collect data. This method has to observe the change of the area and then try to estimate the production. This method requires high budget and big time consuming including labors (Haig, 2003). With circumspect operation, this method will provide the trustable result. However, as mentioned, it deals with high effort, time and budget. Therefore, new approach with acceptable accuracy for this activity has been studied in the way to reduce all costs and time. The mathematical/statistical prediction model which is composed of various production-influencing variables is recently applicable and being in trend.

At the present, two well-known methods used in researches for agricultural yield prediction include the linear one such as regression model, and the non-linear such as Artificial Neural Network (ANN). The linear one has found to be too rigid for some cases while the non-linear one has more advantages on its ability to solve the complex condition problem. Moreover, the non-linear has the ability of self learning in set up relationship of input and output during training process, this can be operated

without conventional weighting and no requirement of any expert experience. The result of the non-linear one shows more accurate and could reduce the problem of the expert shortage.

According to related literatures (Pichaya Boonprasom, 2003; Safa *et al.*, 2004; Liu *et al.*, 2001; Drummond *et al.*, 2000), there were uses of ANN for crop predictions such as corn, wheat, sugar cane and orange. Each report concluded that the result showed reasonable accuracy of prediction.

Moreover, researches have tried to apply remote sensing technology to predict agricultural production, for example, Tucker *et al.* (1980) which was referred by Mohd *et al.* (1994). They have found the relationship of spectral data and agricultural production. Serele *et al.* (2000) has also tried to use various types of vegetation indexes to predict yield of corn.

Although there was quite a number of researches that related to yield prediction, most of them are more likely to apply for crops as shown in Table 1.1 Only few of them were dealt with yield prediction of fruit orchards. There was none dealing with longan.

**Table 1.1** Examples of researches on yield prediction of various crops.

Researchers	Types of plant
Land Development Department and Department of Agriculture (2003)	Sugar cane
Land Development Department and Department of Agriculture (2003)	Rice
Land Development Department and Department of Agriculture (2003) (กรมวิชาการเกษตร และกรมพัฒนาที่ดิน, 2546)	Cassava
Serele <i>et al.</i> (2000)	Corn
Liu, Goering and Tian (2001)	Corn
Baimoung and Oki (2004)	Corn
Safa <i>et al.</i> (2004)	Wheat

The longan is classified to be a kind of popular fruits in Thailand which is grown in form of orchard mostly in the north of the country. It is regarded as the first economical importance among fruits in the upper part of the northern region. It is as well considered to be the Champion product that provides values of thousands of million baths in the export business. Table 1.2 shows export values of longan products for from year 2002 to 2006. Therefore, possible yield prediction of it becomes very important for market management. In order to be able to predict yield, factors influencing longan bloom and production are required.

**Table 1.2** The statistics of fresh longan export

Year	Export product (kg.)	Value (Baht)
2002	113,168,334	1,940,325,574
2003	81,927,746	1,697,517,860
2004	115,479,809	2,160,027,093
2005	133,645,785	2,164,858,851
2006	58,130,772	1,140,374,264

**Source:** <http://www.thailandlongan.com/index.php> (กรมศุลกากร, 2551)

From the past, the method for longan yield prediction was simple. Both cultivated area and per rai (the Thai area measuring unit) production were predicted using information of the year before. The product of these two factors was the annual longan production. The disadvantage of this method is no trustable systematic data collection was performed and the error can be annually accumulated.

Just lately, since year 2003-2004, remote sensing, GIS and GPS technology have been brought to apply to this activity. Nevertheless, field survey which consumes

tremendous time and budget still has been the key method used (กรมวิชาการเกษตร และ กรมพัฒนาที่ดิน, 2546). The following problem is no complete field survey of the whole area could be performed. This can lead to arbitrarily data make up that ends up with no trustable result. To achieve trustworthiness method for longan yield prediction, the modeling is introduced. With this application, only data from sample areas can be selected for training and testing the model. Once the model set up, it will be able to use with incredibly less time and budget consuming. The more it is used with consecutive trustable data, the more accurate result can be anticipated.

Therefore instead of using conventional method, this research will be concentrated on using the modeling to predict yield of Longan in the northern part of Thailand. The unconventional method like ANN or Artificial Neural Network which is considered providing more efficient and accurate is selected to serve the purpose.

ANN has been recently encoded for few tasks to work with GIS. Unfortunately, each of them was developed for a specific work. They have been more likely not open enough for general applications even for similar purposes. This can be because of their fixed number of input factor data and number of nodes in the hidden layer. For example, (Pichaya Boonprasom, 2003) used ANN for yield prediction of tangerine. The factors employed as input data for ANN were only average monthly rainfall and average monthly temperature which in turn were related to the number of input node of ANN. Therefore, it could not be flexible to apply for any other works. It should be more benefit if it is developed in the form that can be applied for multi-purpose or at least applicable within the same area, for example, the ability to predict many types of products. The artificial model developed in the form of an extension of the ArcMap

program with certain input format to which any input data can be adjusted can be a good example. It could be more open for application of various sets of data to different prediction purposes. To develop in this form, the problem or error of transferring data across the software programs could be eliminated. In addition, it will meet the requirement in term of convenience.

## **1.2 Objective of the study**

The main objectives of this study are:

1.2.1 To develop ANN model for agriculture yield prediction.

1.2.1 To predict longan yield using ANN model.

## **1.3 Scope and limitations**

1.3.1 Scope

1.3.1.1 Modules developed will be in form of ArcMap extension

1.3.1.2 Land-use and soil data are directly refereed to LDD (2006) and LDD (1999)

1.3.2 Limitations

1.3.2.1 Age of longan is considered stable for the whole cultivated area in the level that can provide production.

1.3.2.2 Cultivation management is assumed to be constant for the whole cultivated area.

1.3.3.3 Input data relevant to climate are limited to the year 2006-2008. Data of some periods and Amphoes are missing.

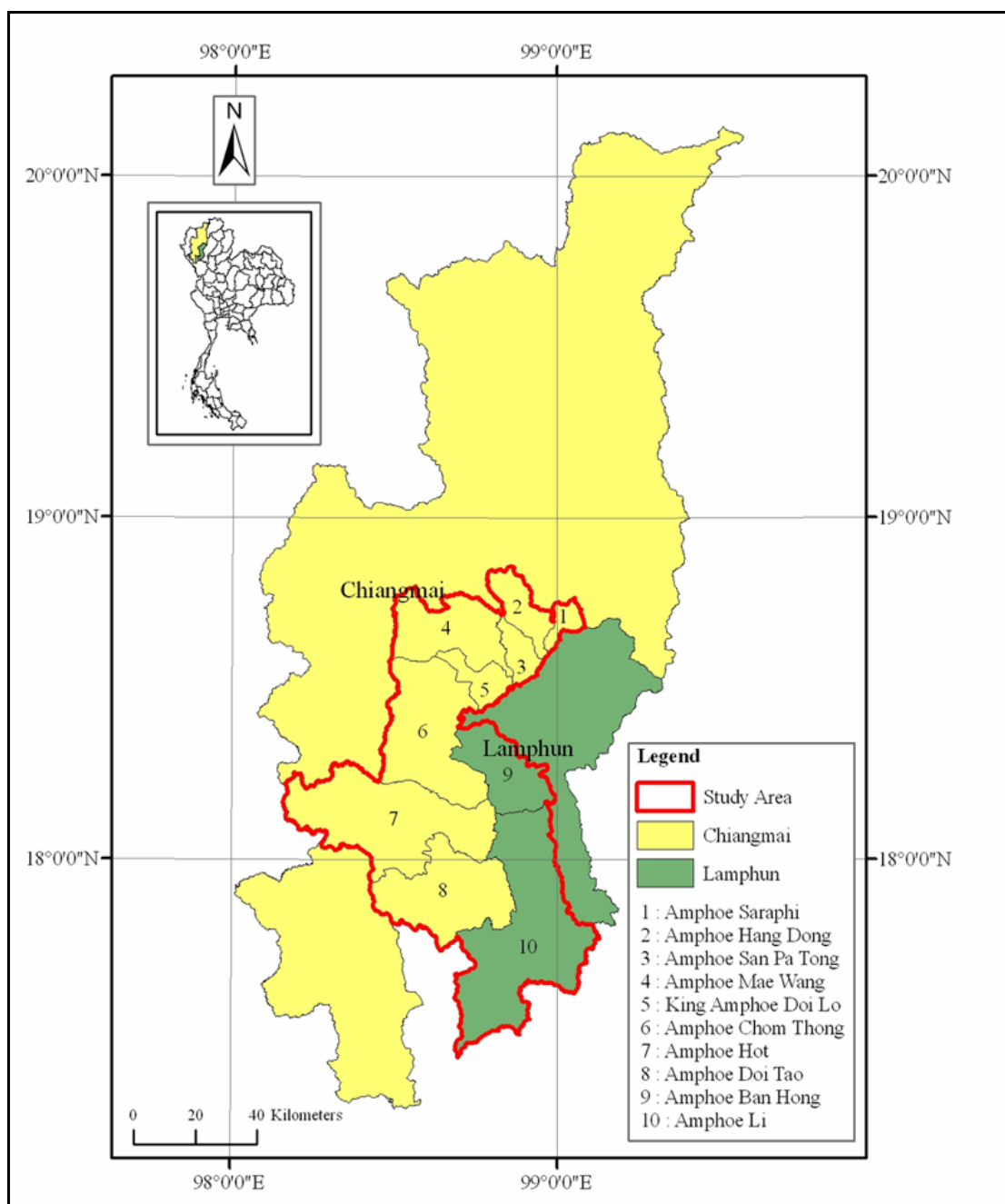
## 1.4 Study area

The study area, approximately 7,327 km<sup>2</sup>, covers parts of Chiang Mai and Lamphun province which is located at latitude 17°42'N-18°85'N and longitude 98°15'E-99°12'E with about 310 meters height over mean sea level (Figure 1.1). Most of the area is characterized by mountain ranges and intermontane basins. Agriculture has been active on the flat and undulating area of the intermontane basins. Weather of Chiang Mai and Lamphun is mild almost all of the year with five years average for the period of 2001-2005, the average temperature is 26 degree Celsius. The average highest temperature is 40.6 degree Celsius and the average lowest temperature is 11.1 degree Celsius. The humidity is between 47-89%. In general, the weather can be separated into 3 seasons. March-April is very hot, May-October has a lot of rain, and November-February is the mild winter. In winter and summer, it is very dry for about 6 months. The other 6 months are the rainy season and the weather is not as hot as summer and not as cool as winter.

This area is selected to be the study area because, in terms of production it can produce longan up to 280,000 metric tons in the drop year 2006 which is about 75% of yield of longan nationwide (กรมศุลกากร, 2551). Again, in terms of area extent for longan plantation, Chiangmai and Lumphun are ranked to be the one of the biggest province. The area of both for longan plantation is located on the South of Chaingmai and every district in Lamphun. However, due to requirement on completeness of data available, 8 Amphoes of Chiangmai and 2 Amphoes of Lumphun were selected to be the study area as shown in Figure 1.1. The 8 Amphoes in Chiangmai include Saraphi,



Hang Dong, San Pa Tong, Mae Wang, Doi Lo, Chom Thong, Hot, and Doi Tao. The 2 Amphoes in Lumphun are Amphoe Ban Hong and Li.



**Figure 1.1** The study area covering Amphoes in Chiangmai and Lumphun Provinces.

Figure 1.2 and 1.3 show longan trees at two periods, when they are being blossomed and providing fruits.



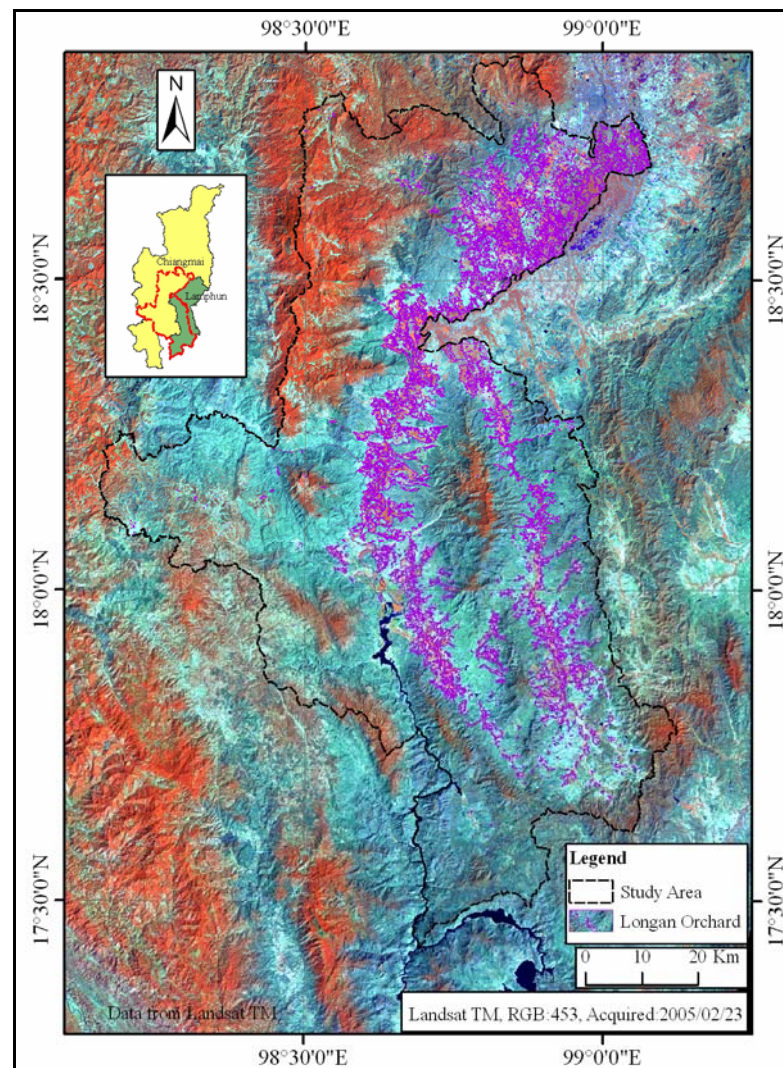
**Figure 1.2** Longan when they are being blossomed.



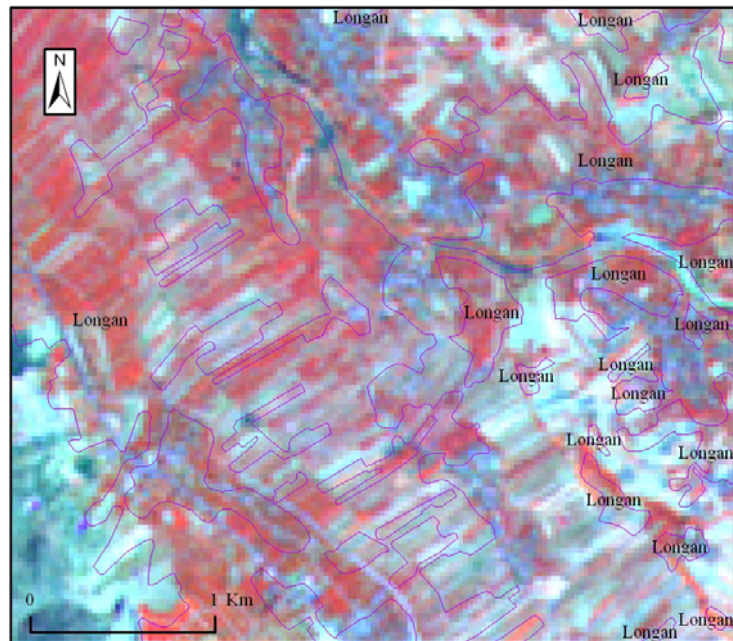
**Figure 1.3** Longan when they are providing fruits.

In Figure 1.4, Landsat 7 TM image covering the study area with longan plantation areas are shown. Considering in enlarged TM image as shown in Figure

1.5, longan plantation even in big area cannot hardly be visually recognized by using TM image due to the limitation of its spatial resolution. This can be big difference when the same area is observed from the QuickBird image which has very high spatial resolution as shown in Figure 1.6. Pattern of cultivation in rows in this kind of image expresses typical characteristic of longan planting area quite obviously.



**Figure 1.4** Landsat 7 TM image covering the study area with longan plantation areas (orchard).



**Figure 1.5** Cultivation characteristic in rows of longan even in the big area can be hardly observed in the TM image.



**Figure 1.6** Typical pattern characteristic in rows of longan planting area is quite obviously observable in QuickBird image.

## **CHAPTER II**

### **LITERATURE REVIEW**

#### **2.1 Previous studies on yield prediction**

From the previous studies, yield prediction of both plants and animals was a quantified prediction that mostly used an econometric model on regression analysis and could display the economic relation between related variables (Chiraporn Srichada, 2002). In addition, some also used Time Series such as the exponential smoothing method to predict the yield (Boken, 2000).

The yield estimation or prediction of any region results from the total of every province by which its own model is used. The plant yield estimation could be gained by using planting area and the product per rai. To predict the yield using regression model, prior information of these variables including the production which is the dependent variable is required. Input data preparation could be also significant because of specific format requirement and taking long time for preparation.

There are two general types of models developed for yield prediction i.e. linear and non-linear. The well-known linear-regression model is classified to be the linear type. While the non-linear one is Artificial Neural Network (ANN). Non-linear model occupies more advantage on the ability with more accuracy to solve the complicated condition problem. Moreover, it has the ability in learning the relationship of input and output during training practice process with no expertise required. As reported, of

the ANN model frequently provides more accurate results. It could as well solve the problem expert shortage. According to the researches reviewed, the ANN has been used to predict crops yield such as corn, wheat, sugar cane and orange. The reports revealed that prediction results were quite accurate (Pichaya Boonprasom, 2003; Liu *et al.*, 2001; Safa, 2004).

Drummond and Sudduth (2000) found that the linear methods generally failed to produce good functional approximations of the spatial yield variability, even within sub-field regions thought to be reasonably homogenous.

According to related researches, there were methods used various factors to predict different types of crop as shown in the following table.

**Table 2.1** The types of crops and factors used in yield prediction.

Research Titles/Author(s)	Methods	Types of crop	Factors/Variables	Accuracy/Correlation
Yield prediction of tangerine using Artificial Neural Network (ANN) Pichaya Boonprasom (2003)	ANN/Back propagation	Tangerine	1. Yield and production area 2. Average monthly rainfall 3. Average monthly temperature 4. Average monthly relative humidity	MAPE = 12 %
A study on estimating of sweet corn production using crop growth simulation model with satellite remote sensing and geographical information system techniques, in Nakhon Ratchasima Province, Thailand/ Somchai Baimoung and Oki (2003)	WOFOST model (crop growth simulation) and regression	Sweet corn	1. Daily meteorological data of year 1999-2000 2. Experimental field area 3. Soil data 4. Crop data 5. Actual yield during growing period. 6. NDVI from LANDSAT-TM imagery	$r = 0.98$

**Table 2.1** (Continued)

Research Titles/Author(s)	Methods	Types of crop	Factors/Variables	Accuracy/ Correlation
A neural network for setting target corn yields / Liu and Tian (2001)	ANN/Back propagation	Corn	1. Soil 2. Weather conditions 3. Factors related to management 4. Other factors that could occur such as storm, insect or disease	80%
A new technology for oil palm LAI and yield prediction / Awal and Ishak (2007)	OPLYM (crop growth model)	Oil palm	1. LAI 2. Actual yield	$r = 0.98$ $R^2 = 0.97$
Agriculture applications of remote sensing: paddy yield estimation from Landsat-5 Thematic Mapper Data / Mohd <i>et al.</i> (1994)	Regression	Paddy	1. Landsat-5 Thematic Mapper (TM) satellite data 2. Ratio Vegetation Index (RVI) 3. NDVI	$r = 0.85$ for NDVI and $r = 0.55$ for RVI
Artificial neural networks for corn and soybean yield prediction / Kaul <i>et al.</i> (2005)	ANN/Back propagation	Corn and soybean	1. Rainfall data 2. Soil data 3. Historical yield data	RMSE = 0.81
Rice Yield Prediction Using a Support Vector Regression Method / Ratchaphum Jaikla <i>et al.</i> (2008)	Vector regression method	Rice	1. Solar radiation 2. Maximum temperature 3. Minimum temperature 4. Precipitation	MAPE = 2.94%

Where : **r** is correlation coefficient

**R<sup>2</sup>** is coefficient of determination

**RMSE** is Root Mean Square Error

**MAPE** is Mean Absolute Percentage Error

Until now there has not been any research really deal with longan yield prediction. There is only the responsible section of government agency that has evaluated the longan product each year using an easy instrument. For example, this section has just explored the planting area and used it to predict yield in year 2004.

Previously the longan yield prediction was performed in a simple way by use of the data of the planting area and yield per rai of the previous year. The product of these two sets of data is used to indicate the predicted yield amount of the recent year. However, the Land Development Department (2004) has tried to explore and evaluate the longan yield in the crop year 2004 using remote sensing data, Geographic Information System and Global Positioning System. Information from satellite image together with surveyed area and farmer interviews was analyzed through the Geographic Information System, without using any sophisticated model. The problem found was this method spent time for field survey too long and was very costly.

## **2.2 Artificial Neural Network Theory**

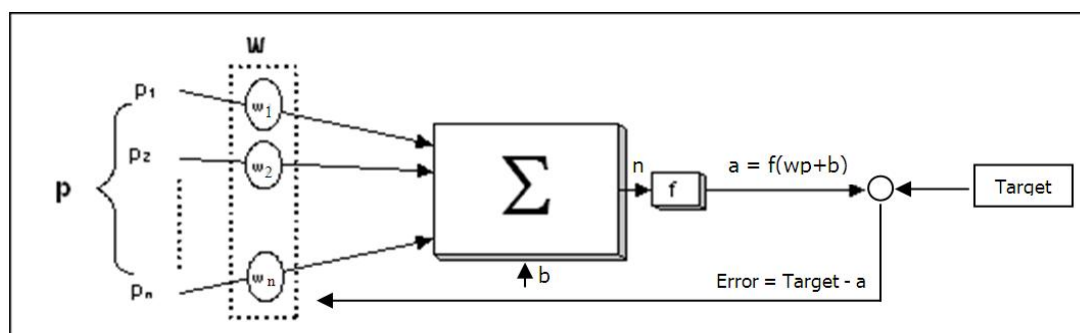
Artificial Neural Network is computer systems which deal with data by the use of simple processors, whose output is then entered into a second layer of processors, and so on through several layers. These networks can throw up pragmatic, rather than theory-based, models, thus producing new insights; for example, neural networks have been used to develop a classification of land use, based on images from remote sensors. They can work on limited data, and are flexible; new connections can be made with the input of further data. (Geography Dictionary, 2004)

Artificial Neural Network, the same as biological neural network, is made up of a set of neurons. These neurons process the presented input and matching output in a supervised manner and make extract non-linear relationships between those input and



output. Information processing in ANN is a form of parallel. (Hosseinia *et al.*, 2007) ANN offers an approach that is completely different from those offered by previous conventional methods, which need a algorithm to be specified and transformed by a computer program. (Grzesiak *et al.*, 2006)

Artificial Neural Network is a massive parallel process that is formed through a combination of neurons that emulates the neuro-synaptic structure of human brain. A neuron is the smallest computation unit of data. Each neuron is linked to its neighbors with varying coefficients of connectivity. They learn the correlation between the input and the target from examples and then solve the corresponding problems. These examples are called training sets which are entered into the network several times so that the network can learn from them. During the training process the connection weight of the network is adjusted. (Pichaya Boonprasom, 2003; Drummond and Sudduth, 2000; Ramyaa, 2004). Figure 2.1 shows workflow diagram of a multiple-input neuron.

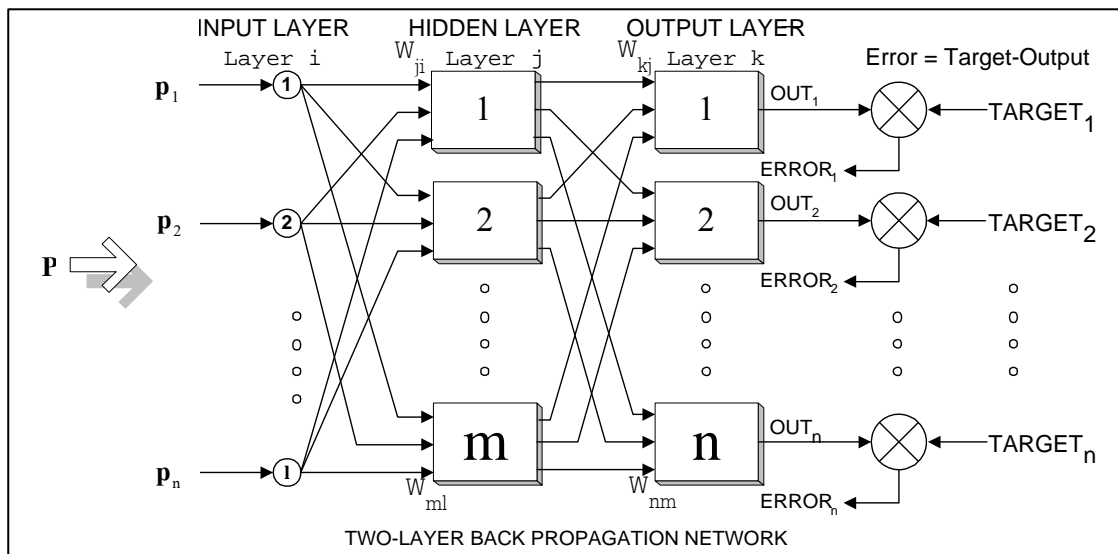


**Figure 2.1** A diagram showing workflow of a multiple-input neuron.

The scalar input  $p$  is multiplied by the scalar weight  $w$  to  $wp$ , one of the terms that is sent to the summation. The other input is added by a bias  $b$  and then passed to the summation. The summation output  $n$ , often referred to as the net input, going into a transfer function  $f$  which produces the scalar neuron output  $a$ .

Often, one neuron with many inputs may not be sufficient. We may need more than one parallel neuron operation, in what we will call a “layer”. A network with several layers is called a multilayer network. Each layer has its own weight matrix  $W$ , its own bias vector  $b$ , a net input vector  $n$  and an output vector  $a$ . Also, different layers may be formed by different transfer functions. The layer whose output results in the output of ANN is named an output layer. All layers which is not the input layer, are called hidden layers.

The best learning method for a multilayer process is the Back-Propagation learning rule which organizes two main paths as shown in Figure 2.2 The first path is called the forward path. In this path, the input vector is applied to the network. The results of the hidden layer are distributed into the output layer. The output vector from the output layer is the actual solution to the network. In this path, the network parameters such as weight and bias are constant. The second path is called the backward path. In this path, all network parameters are changed and regulated. This is accomplished by the error correction rule. An error signal is formed in the output layer. An error vector is equal to the difference between the actual answer and the estimate of the networks answer. The learning rule is then adjusted the weights of the network in order to move the network outputs closer to the targets. (Hagan, Demuth and Beale, 1995). It can be called the “training process” for neuron network. (see detail in Appendix A)



**Figure 2.2** Multilayer Network with Back-Propagation learning rule.

**Source:** Seri Pansang (เสรี, 2539)

### 2.3 Application of ANN to crop yield prediction

It is interesting to note that there is a considerably high number of applying ANN to yield prediction in field of agriculture. Considering Table 2.1, it shows that ANN can provide more effective results. The followings are examples of agricultural yield prediction using ANN for various types of crops.

Monisha (2005) investigated if Artificial Neural Network models could effectively predict Maryland corn and soybean yields for typical climatic conditions. Models were developed using historical yield data at multiple locations throughout Maryland. Field-specific rainfall data and the Soil Rating for Plant Growth values were used for each location. Adjusting ANN parameters such as learning rate and number of hidden nodes affected the accuracy of crop yield predictions. ANN models consistently produced more accurate yield predictions than regression models. ANN corn yield models for Maryland resulted in  $R^2$  of 0.77.

Yu *et al.* (2007) combined neural networks and fuzzy set theory for predicting crop yields. The results of modeling plant growth on the basis of the Crop Growth Monitoring System (CGMS) simulation model as well as the values of the vegetation index NDVI, calculated from the SPOT satellite data, are the input parameters. As a result of training the neural network, the accuracy of predicting yields is on average about 75%.

Chen and McNairn (2006) developed the system that delineated rice production areas in both wet and dry seasons. The system was able to extract information on rice cultivation as a function of different planting dates. A minimum mapping accuracy of 96% was achieved for both seasons. This information was then used in a neural network-based yield model to predict rice yield on a regional basis for the wet season. When the yields predicted by the neural network were compared with government statistics, the prediction accuracy was high up to 94%.

There has been a plenty of researches that employed ANN for prediction and still show high tendency with time. However, most of the works ANN is used particularly with a single plant or crop. It is so far unable to reuse the model developed to be suitable for one crop to other crop. This is possibly because each type of crop requires different inputs and also requires recent past input data of a certain type of crop to train itself in order that it can learn the relationships of varying input factors and yields of that crop. In addition, not all types of crops have been studied on yield prediction using ANN. Unfortunately; there has no any study on longan yield prediction using ANN as well.

## **2.4 Factors related to the longan blossom and fruit producing**

As the tropical country like Thailand, the factors that usually influence the growth of plants are light, temperature, humidity or water, turgor pressure of cells, most types of gases, and soil. Light is the energy source that plants require for photosynthesis process. Normally soil and environmental temperature control the temperature in plant stem, while humidity or water is the substance in photosynthesis process. Turgor pressure of cells relates to minerals and other substances transferring and exchange. Most types of gases and their quantity in the atmosphere are useful for plant too. These gases are oxygen, nitrogen and carbon dioxide. Roots take oxygen from the soil. If there is not enough oxygen, roots will not be able to grow and the root system will not be effective. The result is less water and minerals will be absorbed. Soil is the source of air supply and the place where microorganisms change several poisonous components into non-poisonous components. When the respiration process of plants occurs, they absorb necessary minerals from ground for growing. Minerals and soil condition are also important to the plants healthy development. As are minerals, gases such as carbon, hydrogen and oxygen which are produced from water and air are important as well.

Each factor mentioned above is a fundamental factor influencing growth of general types of plant. However, different types of plant could have different specific requirements to be taken care of. This can be an important role in the plants development, for example, factors influencing the longan product will be ones which effect its bloom. The more bloom is more likely to indicate the more production.

Therefore, influencing factors can be divided into 2 groups i.e. uncontrollable which are natural factors and cultural practices as listed in Table 2.2 (กรมวิชาการเกษตร, 2545).

**Table 2.2** Factors influencing yield of Longan.

Uncontrollable factors	Cultural practices
<p><b>Temperature:</b> Low temperatures of about 10-20 degree Celsius for at least two consecutive weeks is required for longan to start blooming.</p>	<p><b>Varieties:</b> The selection of longan varieties is based on the different blooming habit of each variety.</p>
<p><b>Sunlight:</b> Blooming of longan is well-developed at the top of the tree where more sunlight can be taken than other parts. This means the more sunlight it receives, the more it blooms.</p>	<p><b>Pruning:</b> Pruning is carried out to control tree size (height and width), and to maintain production of the lower tree canopy and light on all sides of the canopy.</p>
<p><b>Soil characteristic:</b> The soil characteristic should be highly fertile. In Thailand the soil data are classified and mapped in terms of soil series. Each is suited to different types of planting. The pH of the soil suitable for longan is 5.5 to 6. Low soil moisture and low temperature will lead to more blossoms of the longan.</p>	<p><b>Spacing:</b> The longan trees should be planted at around 22 to 25 feet or more apart. Planting longan too close to each other will lead to improper growth and low yield because of the shading.</p>
<p><b>Nutrient availability of longan trees:</b> With the same age, longan trees with more nutrient available will start blooming faster.</p>	<p><b>Irrigation:</b> Proper soil moisture is dependent on both irrigation method and quantitative supply. Trees should be irrigated regularly from the blooming period through the harvest</p>
	<p><b>Harvest:</b> Good harvest practice effects the good bloom in the next season</p>
	<p><b>Pests and diseases control:</b> Good practice is required.</p>
	<p><b>Fertilizing:</b> Good practice is required to maintain proper nutrient of plant and soil.</p>

According to Department of Agriculture (2002) the factors related to the longan blossom and fruit producing are listed as follows.

#### 2.4.1 The factors related to the longan trunk

Longan takes 6-8 months from the start of blossom to harvest time. It requires a lot of nutrients especially in the year that it produces a lot of fruits. Some nutrients in soil are taken away with fruits it produces. This later causes lacking of soil nutrients. Therefore, the soil requires fertilizers in time. Moreover, only 4-5 months of feeding time is not enough for the longan. If it is not taken care well enough, the tree will not be sufficiently fertilized. If the environment is not suitable, the next-year yield could be less or it could not provide blossom. The most important factor is the apical growth that strongly relates to the environment. During November to December it is the time before longan blossom. The longan could stop producing leaves and shoots until the winter came, and the opportunity for blossom could be higher.

#### 2.4.2 The factors related to environment

1) Temperature: temperature is the important factor for longan that relates to bud producing. The temperature suitable for longan to flowering is 10-12 degrees celcius which normally appears in winter during November and December. This low temperature for a period of time can stimulate longan flowers to be blossomed.

2) Sunlight: the longan usually produces flowers at the apical zone where the sun shines the most. It would produce lesser amount of flowers at shaded position. The longan growth rate is influenced by duration of sunlight. It relates to the amount of Carbohydrate inside the trunk. The longan flowers when Total Nonstructural Carbohydrate (TNC) in the longan trunk is at a high level. TNC level will decrease

when the longan blossoms and continue decreasing until harvesting time. According to the analytical results, TNC and TN rate will be higher in the year before longan has extraordinary blossom than the year before it has less flower blossom.

3) Soil humidity: if soil humidity shows low level and the weather is cool before flowers blossom, longan will be stimulated to produce better flowers. The degree of soil humidity will affect the water absorption ability of a plant and will decrease the nitrogen amount that a plant can absorb. If the plant can absorb more nitrogen, the rate of producing leaves would be higher while the rate of producing flowers would be lower. Furthermore, low soil humidity could affect Abscisic acid hormones that subsequently reduce longan growth rate. With low growth rate, longan will have higher opportunity for producing flowers.

4) Fertilizer feeding: the amount of minerals from fertilizer that longan need should be added with appropriate quantity and time. The right amount of minerals from fertilizer could stimulate the production of flowers. Minerals can relate to the plant healthy. For longan trees with similar age, the more fertilized longan trees would earlier produce flowers. Normal watering and fertilization should be operated particularly at the time the longan is producing flowers and fruits.

Apart from factors mentioned above, crop calendar should be considered for checking suitable time for taking data used for the model. Table 2.3 shows longan crop calendar.

**Table 2.3** Longan crop calendar

Jan	Feb	Mar	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Bring dividens	Blooming	Early fruit set	Green mature	Harvest period	Post harvest	Sprout period		Leaf mature period	

**Source:** Department of Agriculture (กรมวิชาการเกษตร, 2545)



## 2.5 ANN module development

A number of research projects has used the ANN model through simulator that can be represented the project purpose. The popular simulator is the MATLAB. However, some other research projects operate on models that have been programmed for specific purposes of the projects. The advantage of this kind of model is that it would fit more and support all requirements of the project purpose. Table 2.4 shows examples of ANN researches that used simulators and own-development programs.

**Table 2.4** Examples of researches using ANN and own-development programs.

Title	Neural Network Software
Classification of Slovak white wines using artificial neural networks and discrimination techniques	Trajan (Simulator)
A very fast learning method for neural networks based on sensitivity analysis	MATLAB (Simulator)
ANN+GIS: An automated system for property valuation	Trajan (Simulator)
Tunneling performance prediction using an integrated GIS and neural network	MATLAB (Simulator)
An ANN application for water quality forecasting	Neuroshell SW package
Artificial neural networks based internet hypertension prediction tool development and validation	Net2004 (Own-written SW using Delphi)

Recently, the ANN model has been becoming very popular. Therefore, software development of the model has been carried out for a variety of applications, namely, classification, clustering, prediction, and recognition. Table 2.5 lists some neural

network simulation software available at present and programming languages used for their developments.

**Table 2.5** Some neural network software available at present.

Simulator	Latest release	License	Language
Emergent (formerly PDP++ and PDP)	2008	GNU GPL	C++
FANN (Fast Artificial Neural Network Library)	2006	GNU LGPL	C
GENESIS (the GEneral NEural SIMulation System)	2007	GNU GPL	C
NEURON (originally CABLE)	2008	GNU GPL	C, C++
Neuron-C	2008	GNU GPL	C, C++
SNNAP (Simulator for Neural Networks and Action Potentials)	2007	©, Proprietary	Java
LENS (The light, efficient neural network simulator)	2000	Unknown	C
Catacomb2 (Components And Tools for Accessible Computer Modeling in Biology)	2003	GNU GPL	Java
Topographica Neural Map Simulator	2008	GNU GPL	Python/C++
(iNVT) iLab Neuromorphic Vision C++ Toolkit	2005	GNU GPL Must request access	C++
NEST (NEural Simulation Tool)	2006	©, Proprietary	C++
KInNeSS - KDE Integrated NeuroSimulation Software	2008	GNU GPL	C++
XNBC	2006	GNU GPL	C++
Stuttgart Neural Network Simulator	2001	Free for academic	C++

**Table 2.5** (Continued)

<b>Simulator</b>	<b>Latest release</b>	<b>License</b>	<b>Language</b>
MATLAB Neural Network Toolbox	2008	©, Proprietary	
Wolfram Mathematica Neural Networks	2003	©, Proprietary	
Peltarion Synapse	2008	©, Proprietary	.NET
PCSIM: A <i>Parallel neural Circuit SIMulator</i>	2008	GNU GPL	Python/C++/Java
Brian Spiking Neural Network Simulator	2008	CeCILL License	Python
Trajan	2008	©, Proprietary	

Source: [http://en.wikipedia.org/wiki/Neural\\_network\\_software](http://en.wikipedia.org/wiki/Neural_network_software) (Wikipedia, 2009)

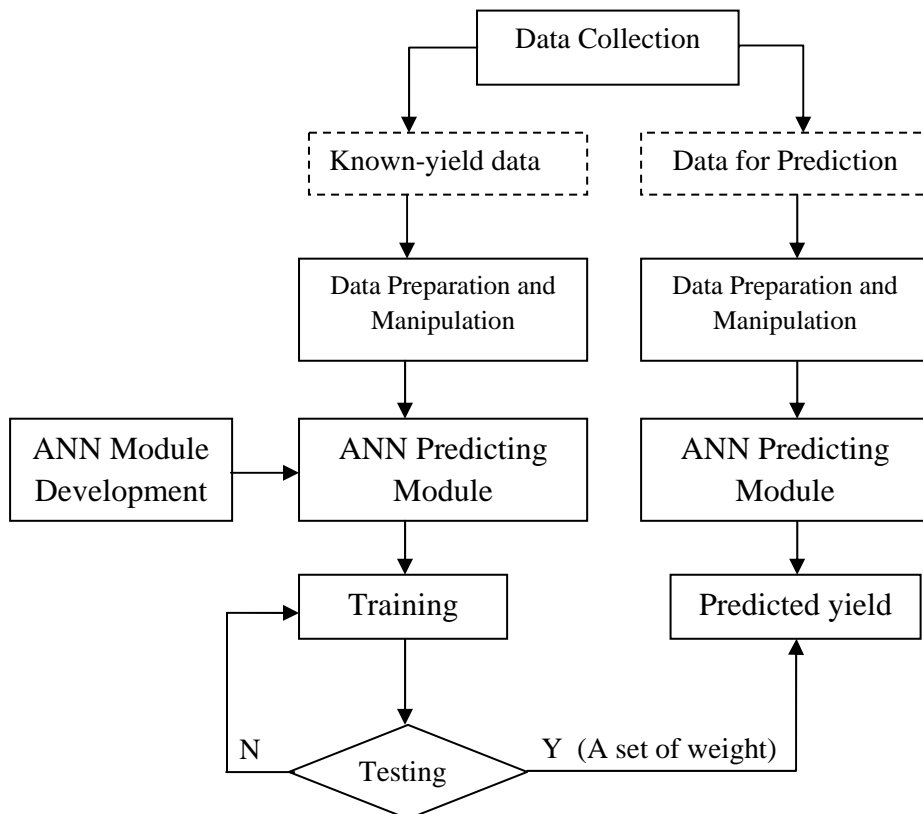
From the above table, it is very interesting to note that there has been none of simulators developed as a module embedded to work with ArcMap. Even though there are a number of simulators of ANN for various applications available, there is none specifically developed for longan yield prediction.

# CHAPTER III

## METHODOLOGY

### 3.1 Conceptual overview of methods of the study

The conceptual overview of methods of the study can be displayed as a flow chart in Figure 3.1. General steps are data collection, preparation and manipulation, ANN module development, training, testing, and yield prediction.



**Figure 3.1** The flow chart of methods of the study.

### 3.2 Data collection

From whole data survey, it is realized that all data required for the study are systematically captured very well in the limit of Amphoe based. Therefore, Amphoe is designed to be the spatial unit for analysis.

Factors involved include temperature, sunlight, rainfall, Amphoe boundary, land use and soil as listed in Table 3.1. Characteristics or format of all data collected were examined and furthered to preparation and manipulation to be in the required format and representation that will be effectual for ANN prediction module developed.

**Table 3.1** Data layers applied in the present study.

<b>Data Layer</b>	<b>Data Type</b>	<b>Scale</b>	<b>Source</b>	<b>Year</b>
Temperature	Point	-	HAI	2006-2008
Sunlight	Point	-	HAI	2006-2008
Rainfall	Point	-	HAI	2006-2008
Land Use	Polygon	-	LDD	2006
Soil Group	Polygon	1:100,000	LDD	1999
Amphoe	Polygon	1:100,000	LDD	1999

HAI: Hydro and Agro Informatics Institute

LDD: Land Development Department

#### 3.2.1 Temperature, sunlight, and rainfall

These data were collected from 20 Meteorological stations of Hydro and Agro Informatics Institute (HAI). Data from each station of these factors are separated to be hourly that cover the range of time from 2006-2008. Even though data of some instants are missing, the whole data available are still considered applicable for the study. These data were furthered prepared and manipulated in the next step.

### 3.2.2 Land-use and Amphoe data

These data covering the study area were classified by LDD (2006) as GIS data layer. Only areas with longan planting were selected and were used together with soil data layer to obtain total area extents of longan planting area of each Amphoe and soil group within its. It is assumed that the planting areas are the same with productive areas of longan.

### 3.2.3 Soil data

These data covering the study area were classified by LDD (1999). Soil data of the area were mapped as group in the scale of 1:100,000. Only soil group area fall into the longan planting area are considered for prediction activity. Therefore, area extent of each soil group within each Amphoe of longan planting area were further prepared and formatted as input data to the ANN analysis.

### 3.2.4 Longan yield data

Longan yield data of each Amphoe in year 2007-2008 of the study area were obtained from Chiangmai Provincial Agricultural Extension Office and Office of Agricultural Economics. The unit of yield production is tons per Amphoe.

## **3.3 Data preparation and manipulation**

These processes cover clipping, interpolation, formatting, and normalization data required as input to the ANN analytical model. The original data were prepared in forms of GIS data layers and .dbf files.

### 3.3.1 Temperature, sunlight, and rainfall data

These data are in the same original format of hourly records which were formatted to be spreadsheet data. Total data cover 16 weeks from the beginning of November to the end of February (see longan crop calendar in Table 2.2). Because

longan productivity is related to a certain temperature range in a certain period of time before production (กรมวิชาการเกษตร, 2545). N1 to N4 fall into November while D1 to D4, J1 to J4, and F1 to F4 fall into December, January, and February respectively. For temperature data, they were prepared to be daily minimum, maximum and average values and then to be weekly minimum, maximum and average values which are attached to each HAI stations. For sunlight and rainfall data, they were prepared to be daily average values and then to be weekly values. All weekly data of each HAI station were input through interpolation in order that the average data of each Amphoe polygon can be achieved. These data were normalized to be 0 to 1 by using performance of  $(\text{raw}-\text{min})/(\text{max}-\text{min})$ . These data will be later used as input data for ANN training, testing and prediction analyses.

### 3.3.2 Soil data and longan planting area of each Amphoe in the study area

Area extent of soil group within longan planting area of each Amphoe were obtained from soil data layer unioned with selected Amphoe data layer and longan planting area from land-use data layer. Then, querying was performed with condition of existing longan area and selected Amphoe. Area extents of soil group within each Amphoe were calculated to be percentage. By this method, the soil within the study area can be more than 70 groups which is too high for input to the analysis. Therefore, if considering one Amphoe at the time, soils group covering area less than average were grouped together to be “other” group. To manipulate like this, the biggest area for “other” unit can reach to 26.42% which is apparent in the Amphoe Hot, Chiangmai Province and a total number of soil group can be reduced to 21 groups. Area extents in percentage of soil group of each Amphoe were normalized to

be 0 to 1 by being divided by 100. Also, longan planting area of each Amphoe were normalized to be 0 to 1 by using performance of  $(\text{raw-min})/(\text{max-min})$ .

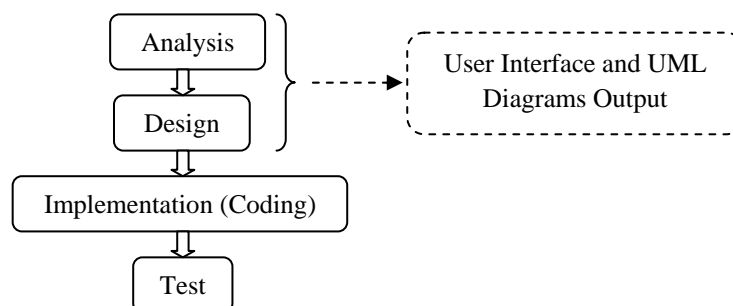
### 3.3.3 Longan yield data

With respect to the same influent year and Amphoe, these data were tabulated associated with 16 weekly items of temperature, sunlight, and rainfall data. They were normalized to be 0 to 1 by performing  $(\text{raw-min})/(\text{max-min})$  when yields of whole Amphoes in the study area were considered.

These above mentioned data were finally formatted to be .dbf.

## 3.4 ANN module developments

The module developed will cover functions of the ANN model and related data flow which involve input, training, and testing processes. Their relationship has to be set up properly to serve the purpose of the model. Coding and debugging will be conducted to achieve module which practically covers whole functions for yield prediction. Steps to develop ANN module can then be separated as displayed in Figure 3.2. These include analysis, design, implement/coding and test.

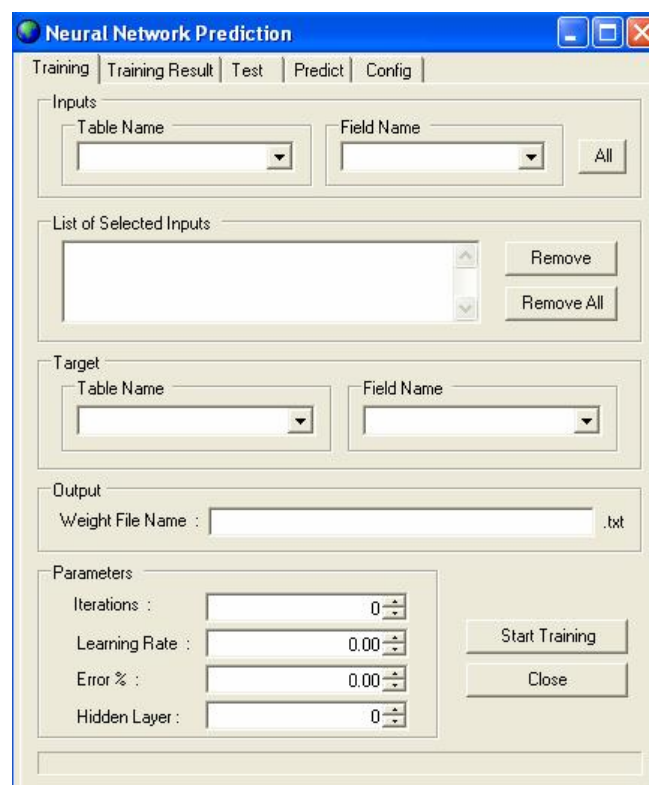


**Figure 3.2** The steps for the module development



### 3.4.1 Analysis and design

The step of analysis is to comprehend the whole functions of ANN model and their relationship so that the module schema or herein class diagram can be designed for further implement. As a certain level the general functions of the model are composed of input, feed forward, back propagation, weight adjustment, and error assessment. UML (Unified Modeling Language) was applied to interface and class diagram design/construction. The interface design should correspond to the requirement of the model, while class diagram is a map to tell how and which element of interface can connect to data for input and to analytical processes of the model. As the result, User Interface and Class Diagram of ANN for prediction module were analyzed and designed to serve the model purpose as shown in Figure 3.3 and 3.4 respectively.



**Figure 3.3** The software prototype or interface design of ANN model for longan prediction.

*The User Interface* was designed in order to relate processes and functions as follows:

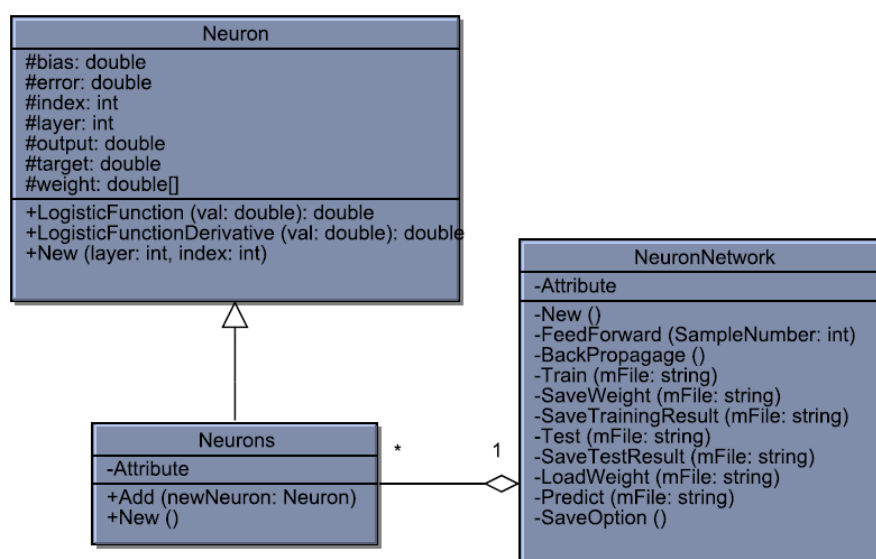
1) Training: Training is the process to input recent past data into the model so that the optimal weight set for each variables can be determined.

2) Training result: Training result is a checking process for error result of each loop and the number of training loops. These values can be saved and retrieved for later examination.

3) Test: Testing process is employed to assess the accuracy of the predicted yield, resulted from applying weight set from the training process, by comparing to the target which is the known yield.

4) Predict: Predicting is a process to predict yield when new data record without known yield/target is input to the model.

5) Config: Configuring process allows user to specify the path for input data and data from training.



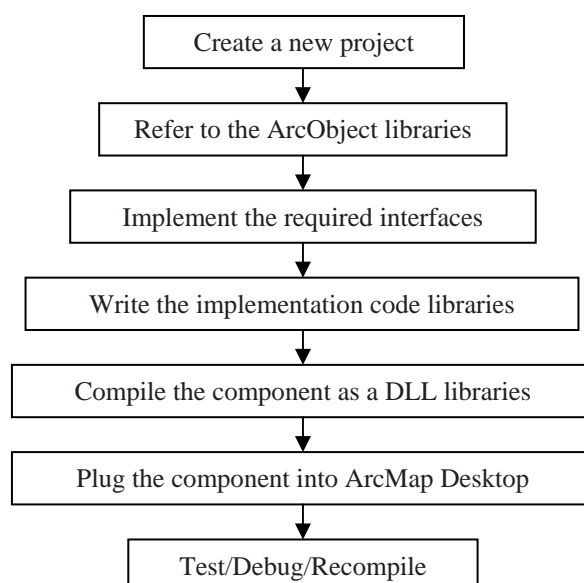
**Figure 3.4** Class Diagram of Neural Network for Prediction Module

The Class diagram is the design plan of objects, their characteristics and relationships within system. Each class was designed for each function as follows:

- 1) Neuron: A class provides the structure of each node.
- 2) Neurons: A class, as the layer in network, provides counting input and hidden node needed. The layer was designed from the user-specified parameters.
- 3) NeuralNetwork: A class works as collection of all layers of neurons that are linked and formed as network. The methods of the class include training, testing and predicting.

#### 3.4.2 Implementation

After the analyzing and designing processes, the module can be implemented or coded using VB.NET language. This language has no influence on working efficiency. The way to choose programming language will depend on the ability of the programmer. The implementation process can be shown as a diagram in Figure 3.5. It is divided into the following steps.



**Figure 3.5** The steps to implement ArcMap extension module using VB.NET

### 1) Create a new project

The way to create a custom component is shown as follows:

1.1) Choose New under File menu, then choose Project and ActiveX DLL (Class library), type the project name and then click OK.

1.2) Right click of project name, choose Add, choose Com Class, and then type the class name and open to add new class.

### 2) Refer to ESRI libraries

Reference ESRI libraries are needed to be connected and added so that Visual Basic will know about the ArcObject libraries, thus:

2.1) Go to solution explorer, then right click at Reference and Add Reference.

2.2) Go to Add Reference dialog box, then click tab .NET and select needed reference.

### 3) Determine the required interfaces

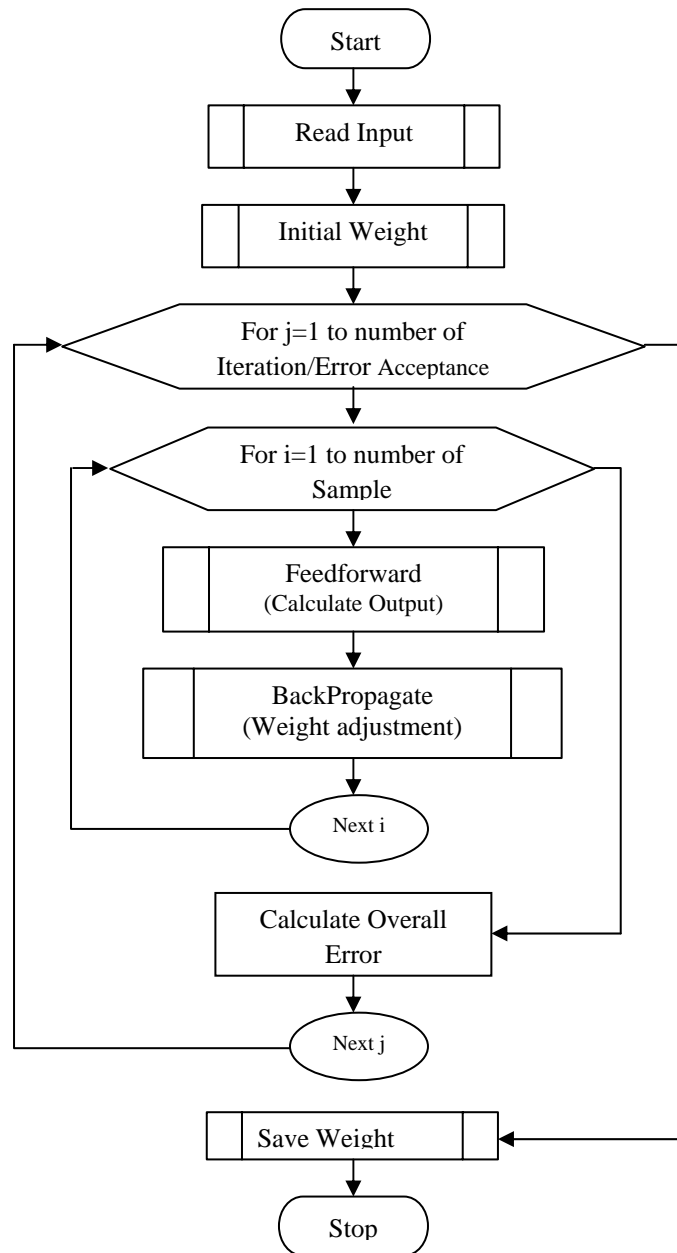
Interface components are appropriately implemented to ensure that all components could be understood by ArcMap. The way to implement interfaces is below.

3.1) Go to Solution explorer, double click at class to open code window.

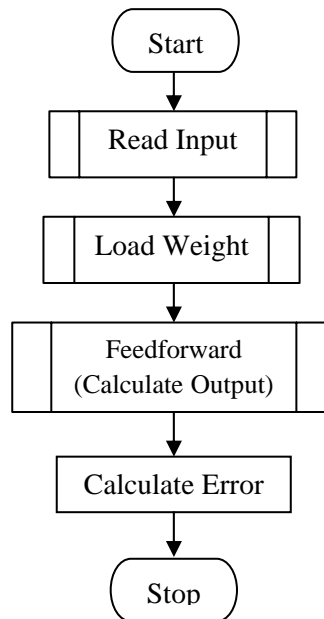
3.2) Go to the top part of the code, then import needed interface.

### 4) Write component code

Write the code for all methods and properties of each control in the interface. Figure 3.6 and Figure 3.7 showed the flowchart of the Training and Testing algorithm.



**Figure 3.6** The flowchart of Training function.



**Figure 3.7** The flowchart of the Testing function.

#### 5) Compile the component DLL

Compiling components to be a DLL component is carried out by using function `Make <project name>`

#### 6) Plug the component into ArcMap

Before using data in ArcGIS desktop application (including ArcMap), the registration of the DLL component with OS and Component Category of ArcGIS is needed to be done. The registration to ArcGIS is set before hand in Visual Studio.NET when working on coding and before compiling. After compiling the code to be the DLL components, the registration into the Component Category of ArcGIS will be done automatically.

#### 7) Test/ Debug/ Recompile

This step means that the coding will be tested while programming by running the program under the real environment. If there is error, debug, recompile and test will be operated again and again until free from error.

### 3.5 Training and testing

Variation of model parameters for cross validations which include the number of nodes, learning rate (LR), the number of iterations, and errors acceptance was conducted to obtain the best accuracy for the model. These selected parameters of the model were further used for cross validations with varying input data.

#### 3.5.1 Variation of model parameters

These model parameters include the number of hidden nodes, the number of iterations, and learning rate. These parameters were varied for tests. Total 12 tests will be performed to obtain the best model parameters. It means that 4 tests for varying of a certain parameter while other two parameters will be kept constant. To do like this until each parameter has gone through varying. The best of each model parameter obtained will be further used in the training and testing of the model. The 10 data records of 10 Amphoes were used as input.

The 4 tests with all input data, varying LR (0.1, 0.2, 0.4, and 0.8), and constant iterations and the number of nodes as 3,000 and 4 respectively are performed (see Table 3.2). Accuracy from each Amphoe will be reported. The test with the best average accuracy from all Amphoes will provide a certain value of LR. This LR, so called LRbest, will be furthered used for the process looking for the best number of hidden nodes as shown in Table 3.3.

**Table 3.2** The tests with varying LR. The LRbest will be obtained from the tests.

No. of test	No. of hidden nodes	No. of Iterations	LR
1	4	3000	0.1
2	4	3000	0.2
3	4	3000	0.4
4	4	3000	0.8

In Table 3.3, varying number of nodes which are 4, 5, 6, and 7 were input to run 4 tests with constant iterations (3,000) and the best LR obtained from the tests mentioned above. In Table 3.3, it is assumed that 0.4 is the  $LR_{best}$ . The number of hidden nodes with highest accuracy from the tests will be chosen for further process which looks for the best number of iterations.

**Table 3.3** Tests to choose the best number of hidden nodes. Iterations and the best LR are kept constant.

No. of test	No. of hidden nodes	No. of Iterations	LRbest
1	4	3000	0.4
2	5	3000	0.4
3	6	3000	0.4
4	7	3000	0.4

In Table 3.4, varying iterations which are 3,000, 5,000, 7,000, and 10,000 are input to run 4 tests with constants of the best number of hidden nodes and LR. For better understanding herein, these constants are assumed to be 6 and 0.4 for the best number of hidden nodes and LR, respectively. The iterations with highest accuracy from the tests will be chosen.

**Table 3.4** Tests to choose the best number of iterations. The best number of hidden nodes and LR are kept constant.

No. of test	No. of hidden nodesbest	No. of Iterations	LRbest
1	6	3000	0.4
2	6	5000	0.4
3	6	7000	0.4
4	6	10000	0.4



The model parameters assigned to stop the training process properly include either the number of iterations or the error acceptance assigned to the module. The training process will be stopped when either one of them is reached. To achieve the best result, the module developed allows users to be able to choose degree of error acceptance which is addition from the number of iterations. For example, if 95% or 90% of accuracy is required, error acceptance of 0.05 or 0.1 can be input to the model respectively and the process will stop when the required error acceptance is reached. However, if the required error acceptance cannot be reached, the process will stop when the number of iteration is reached to 5,000 as set as the highest limit.

The process with the low error acceptance can be expected to provide the better result. However, in some case the result with required error acceptance might not be able to reach, if the assigned accuracy is too high or other words the assigned error acceptance is too low. In addition, the time spent to process using these conditions can take too long. Therefore, it could be better idea if the limited number of iterations is set as an additional condition to stop the process.

### 3.5.2 Cross validation

The performance of ANN model can be evaluated using an  $n$ -fold cross-validation technique. The relevant data set was partitioned randomly into  $n$  equally sized sets. Training and testing were carried out  $n$  times. Each time will use one distinct set for the testing phase and the remaining  $n-1$  sets for the training phase (Kohavi, 1995).

For training, an input data set can contain as many records as required or as data available. Each record in the data set was prepared for annual data of one Amphoe for two years. Each record is composed of 1 item of Amphoe name, 16

weekly temperature, sunlight, and rainfall, which contain 48 items, 16 items, and 16 items, respectively; 21 items of area extents of soil group; and 1 items of total area of longan plantation. Another item in each record is yield data which is known as target data for error calculation.

Cross validations of 2-year data are described here for an example. According to the 2-year data collection from 10 Amphoes, thus, totally 20 records assumed were used as input data for cross validation of the ANN model. Therefore, 20 cross validations were performed one at a time by use of 19 records for training and one record for testing. In one validation, 19 records were input through ANN model one at a time and the model was performed for a number of loops/iterations as set, for example, 3,000 iterations. This means that if 3,000 iterations are performed, the number of training will reach to  $3,000 \times 19$  times while the number of testing is 19 times. Theoretically, an error (Root Square Error: RSE) in percentage from the training of a set of records will be reduced with increased number of iterations. Herein, a set of records is 19 records. However, the error will reach to the limit with no more reduction no matter how many iterations will be increased. In contrast, if the number of the iteration is too high, the error could be increased (Lawrence, Giles, and Tsoi, 1997).

As testing, after training of 19 records, a set of weight achieved was applied to one record with known yield. Then, an error was estimated when the predicted yield from the model was compared to the known yield of the testing record. Therefore, 20 errors were reported for 20 cross validations. To this end, the average error indicates the accuracy or efficiency of the model (Lawrence, Giles, and Tsoi, 1997).

Root Square Error (RSE) is the method used to estimate the error of each validation. Each yield achieved from prediction has value between 0 and 1. This prediction is the normalized yield and will be converted to be predicted actual yield. Then, the difference between the predicted and actual ones is calculated. Square and its root are applied to eliminate minus difference. The mean of 20 errors from 20 cross validations shows the root mean square error (RMSE) of the model.

For this research, cross validations were carried out with three sets of data in order that RMSE obtained from each set of data can be compared. These 3 sets of data are 10 records of 10 Amphoes during November 2006-February 2007, 12 records of 6 Amphoes during November 2006-February 2007 and November 2007-February 2008, and 20 records of 10 Amphoes during the same periods of time during 2006-2008. For the last set, due to the data availability, climate data of 4 Amphoes which are Hangdong, Saraphi, Li, and Banhong, were missing therefore they can be achieved by interpolation from the surrounding stations.

### 3.5.3 Input data variation for cross validations

Due to the limitation of a number of researches on physical factors influencing longan production, it should be better idea if input data can be varied and results are observed. In the present study, a number of variations of input data is set as shown in Table 3.5 for 1-hidden-layer and Table 3.6 for 2-hidden-layer and errors of results from the operations are observed. In each variation, the cross validation will be performed and average errors will be reported. The variation with the lowest average error or the highest accuracy indicates the best variation for this set of input data that should be used for further prediction. If the new dataset or the old data set with additional new data records is taken, this operation could be performed again.

**Table 3.5** A number of variations of input data for 1-hidden-layer module.

No. of variation set	No. of variation	No. of records	Weekly data	Temperature (min, max, ave.)	Sunlight	Rain fall	Soil Group
1	1	10	N1 to F4	☑	☑	☑	☑
	2	12	N1 to F4	☑	☑	☑	☑
	3	20	N1 to F4	☑	☑	☑	☑
2	4	10	N1 to F4	☑	☒	☑	☑
	5	12	N1 to F4	☑	☒	☑	☑
	6	20	N1 to F4	☑	☒	☑	☑
3	7	10	N1 to F4	☑	☑	☒	☑
	8	12	N1 to F4	☑	☑	☒	☑
	9	20	N1 to F4	☑	☑	☒	☑
4	10	10	N1 to F4	☑	☒	☒	☑
	11	12	N1 to F4	☑	☒	☒	☑
	12	20	N1 to F4	☑	☒	☒	☑
5	13	10	N1 to F4	☑	☑	☑	☒
	14	12	N1 to F4	☑	☑	☑	☒
	15	20	N1 to F4	☑	☑	☑	☒
6	16	10	N1 to D4	☑	☑	☑	☑
	17	12	N1 to D4	☑	☑	☑	☑
	18	20	N1 to D4	☑	☑	☑	☑
7	19	10	D1 to J4	☑	☑	☑	☑
	20	12	D1 to J4	☑	☑	☑	☑
	21	20	D1 to J4	☑	☑	☑	☑

**Table 3.6** A number of variations of input data for 2-hidden-layer module.

No. of variation set	No. of variation	No. of records	Weekly data	Temperature (min, max, ave.)	Sunlight	Rainfall	Soil Group
1	1	10	N1 to D4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	2	12	N1 to D4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	3	20	N1 to D4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	4	10	N1 to D4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	5	12	N1 to D4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	6	20	N1 to D4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	7	10	N1 to D4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	8	12	N1 to D4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	9	20	N1 to D4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	10	10	N1 to D4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	11	12	N1 to D4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	12	20	N1 to D4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	13	10	N1 to D4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	14	12	N1 to D4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	15	20	N1 to D4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
6	16	10	D1 to J4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	17	12	D1 to J4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	18	20	D1 to J4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7	19	10	J1 to F4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	20	12	J1 to F4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	21	20	J1 to F4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

The varying operation can be conducted when input data are prepared and manipulated in the format of .dbf file or they can be adjusted as required when work with the module developed.

### **3.6 Prediction**

The model parameters and varying input data with the best accuracy from the training and testing processes discussed above will be selected to configure the module. Then, the new dataset with the same required pattern can be input into the model/module and the yield can be predicted. The before-hand configuration set up is the recommendation that the best result can be expected by using the recent past information within the same area that model known by training and testing processes.

In addition, these processes mentioned above were applied through both one-hidden-layer module and two-hidden-layer module and results were compared and discussed.

# **CHAPTER IV**

## **RESULTS AND DISCUSSION**

### **4.1 Results of the study**

The results of the study will correspond to what mentioned in Chapter III. These include results of data collection, preparation and manipulation, ANN Module development, the results of trainings and testing, and the result of prediction using input data of the year 2009.

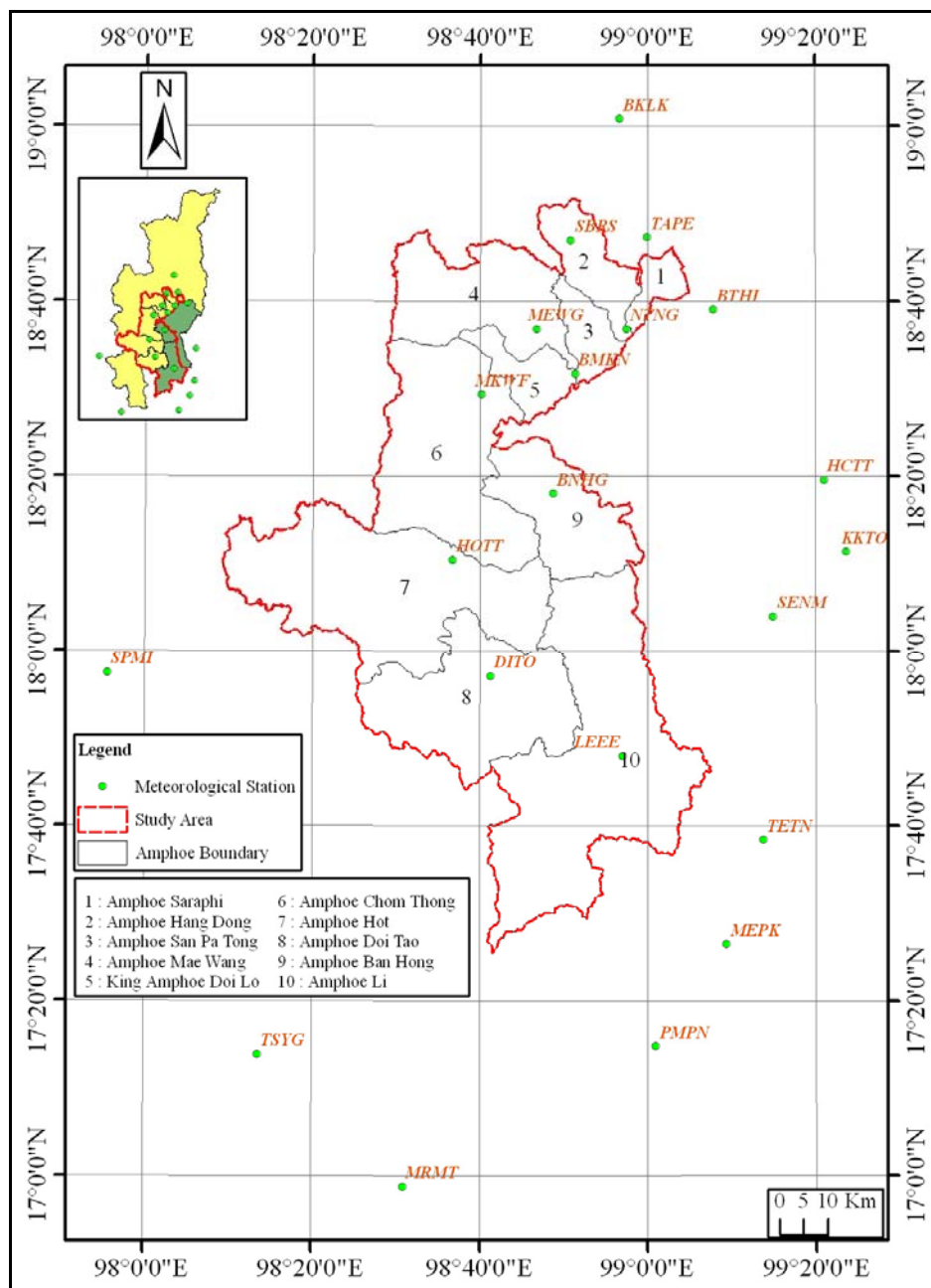
#### 4.1.1 Results of Data Collection, Preparation, and Manipulation

As mentioned in Chapter III, various GIS data layers from several sources were collected, prepared and manipulated to use in this research. These are climate data, land use, soil group, longan yield, and Amphoe. The climate data consist of temperature, sunlight, and rainfall while land use was considered only longan planting areas within each Amphoe of the study area. Soil groups were considered only ones that fall into longan planting areas of each Amphoe. They were achieved by various manipulation methods of selection, interpolation, clipping, average, etc. to make them suitable for input into the model.

##### 4.1.1.1 Temperature, sunlight, and rainfall

These climate data were collected from 20 stations of Hydro and Agro Informatics Institute (Figure 4.1). They are distributed within and surrounding the study area. The data collected were hourly recorded every day of year 2006-2008. The data were examined and considered good for all stations during November 2006-

February 2007 and missing for 9 stations during November 2007-February 2008. The original data are in .txt file format as shown in Figure 4.2. It is interesting to note that data of some hours and days are missing. Therefore, care must be taken when these data are manipulated from hourly to be daily and finally weekly.



**Figure 4.1** Meteorological stations distribution within and surrounding the study area.

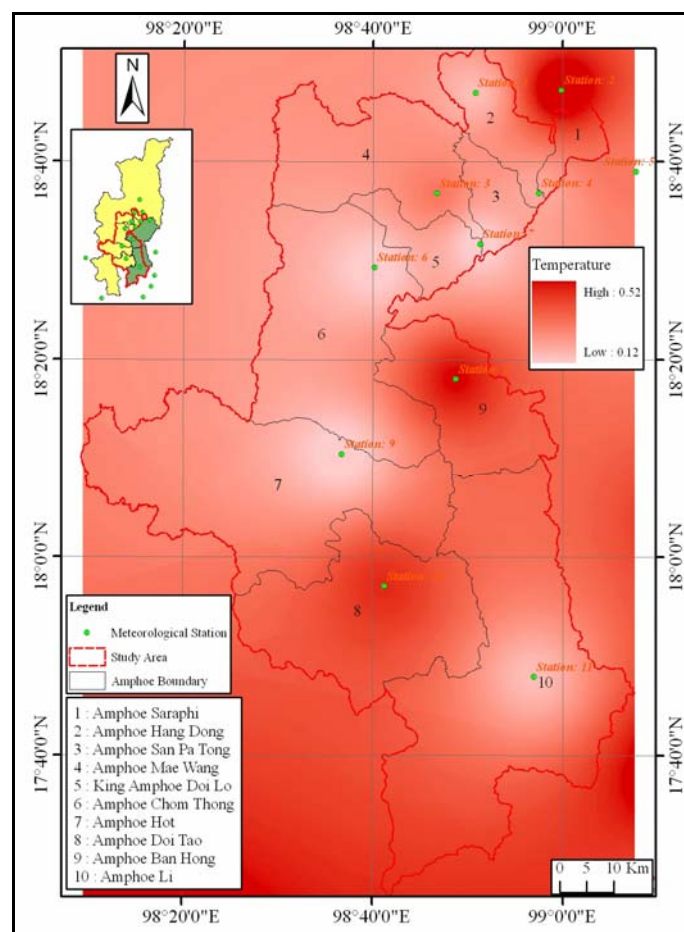




Unfortunately, climate data during November 2007-February 2008 of 9 stations were missing due probably to no active measurement operation or lacking of operator. This effected to 4 Amphoes, Ban Hong, Saraphi, Li, and Hang Dong, whose original climate data within them were missing. Therefore, after data preparing and manipulating only total 16 records of climate data could be gathered with equal amount and quality of data. However, these missing data can be recovered by interpolation from other stations surrounding those Amphoes of which stations had no data in that period. Therefore, to observe different results, data input into the ANN model of this research could be many options namely, 10 records during November 2006-February 2007 of 10 Amphoes, 20 records during November 2006-February 2007 and November 2007-February 2008 of 10 Amphoes with data missing of those 4 Amphoes, and 12 records during November 2006-February 2007 and November 2007-February 2008 of 6 Amphoes and 6 Amphoes respectively.

These data in the form of .txt file were formatted to be spreadsheet data. Data covering 16 weeks from the beginning of November (N1-N4) through December (D1-D4) and January (J1-J4) to the end of February (F1-F4) which were considered critical period influencing to longan production were extracted. Temperature data were prepared from hourly data of a whole week to obtain minimum, maximum and average values which are attached to each HAII stations. For sunlight and rainfall data, they were prepared by averaging from hourly to daily and then weekly values too. Weekly data of temperature, sunlight, and rainfall were then normalized to be 0 to 1. These data attached to 20 HAII stations were interpolated in raster format with spatial resolution 500 m x 500 m. By this process, weekly raster data layers of minimum, maximum, and average temperature, average sunlight, and average rainfall

were obtained. Data layers of minimum, maximum, and average temperatures including sunlight and rainfall of week N1 are shown as example in Figure 4.3-4.7 respectively. Data of all cells in each Amphoe of each layer were averaged to be a value representing that Amphoe. Example of these data of week N1 and N2 are displayed in Table 4.1. These data were already in normalized format and ready to input to the model.



**Figure 4.3** Data layer of minimum temperatures of week N1

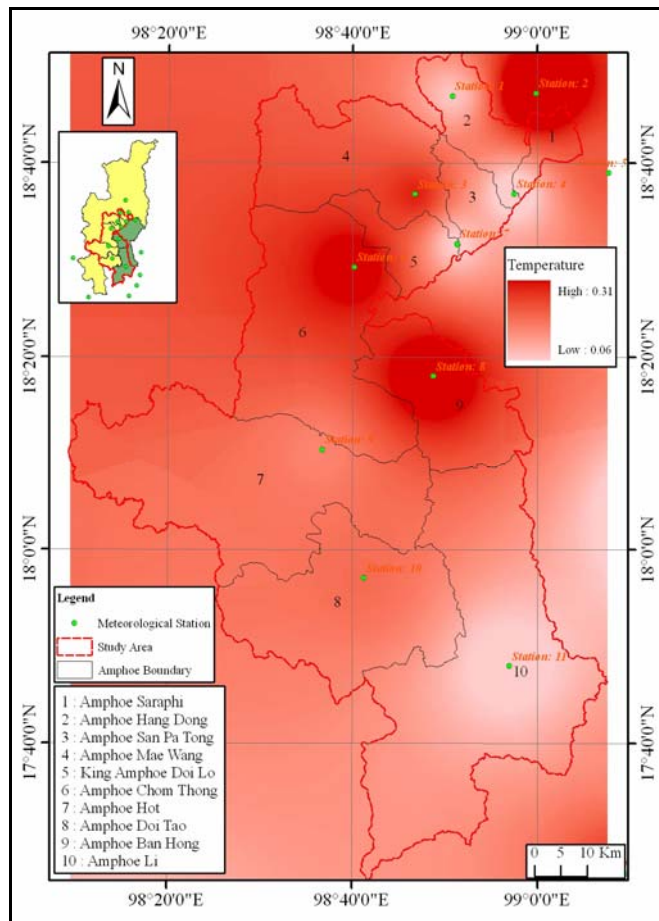


Figure 4.4 Data layer of maximum temperatures of week N1

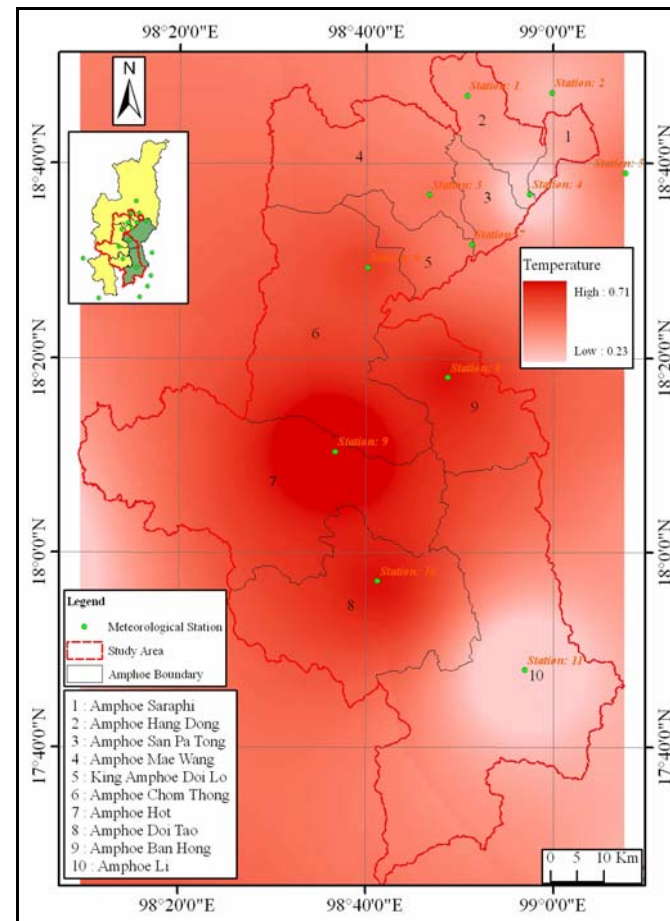
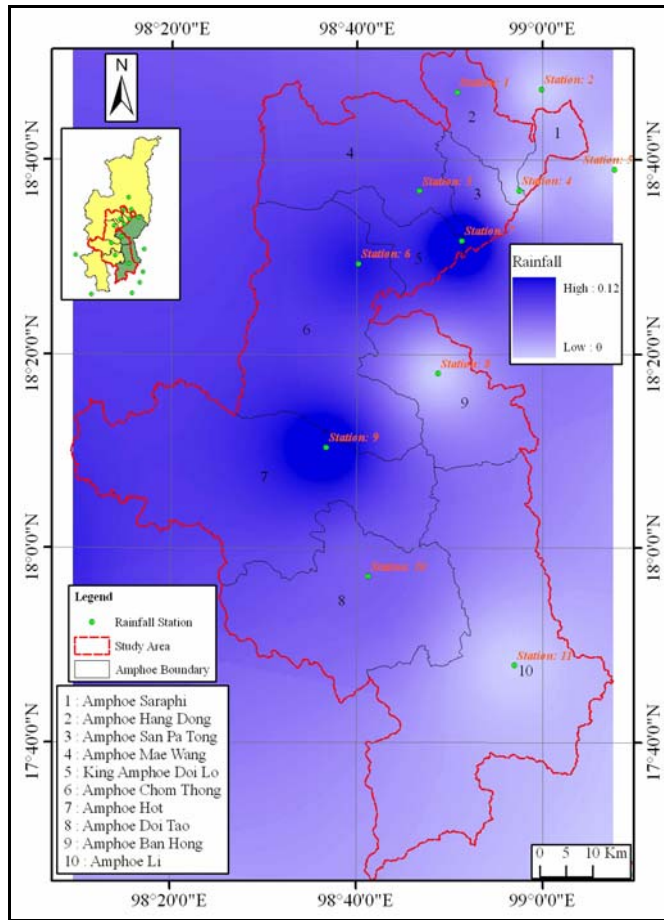
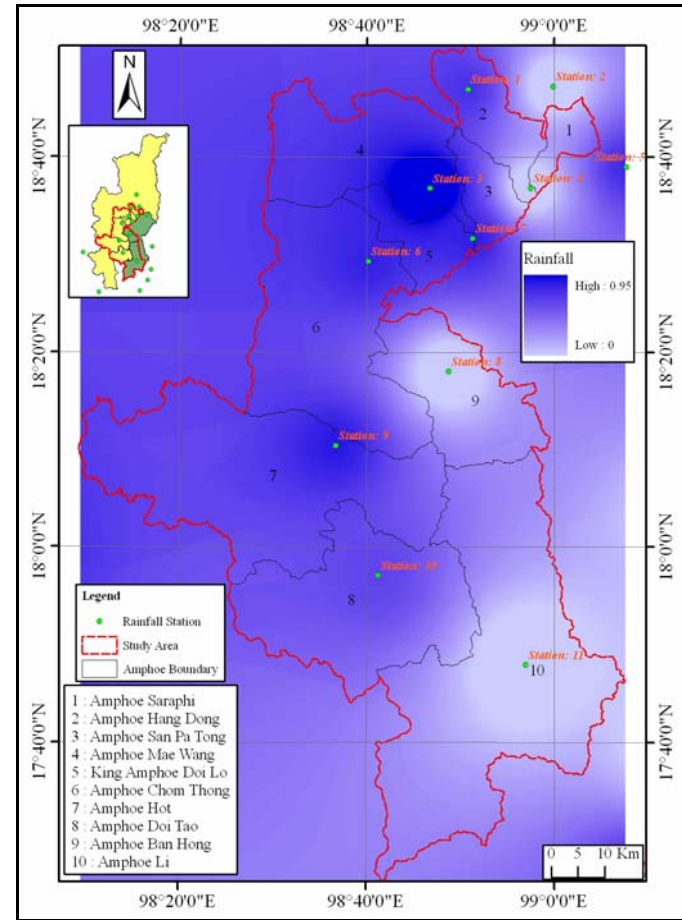


Figure 4.5 Data layer of average temperatures of week N1



**Figure 4.6** Data layer of average sunlight of week N1



**Figure 4.7** Data layer of average rainfall of week N1

**Table 4.1** Example of normalized temperature, sunlight, and rainfall data of week N1 and N2 of each Amphoe.

AMPHOE	TMAXN1	TMINN1	TAVGN1	SAVGN1	RAVGN1	TMAXN2	TMINN2	TAVGN2	SAVGN2	RAVGN2
Hang-Dong	0.38573	0.46906	0.48831	0.48037	0.00089	0.42410	0.59267	0.54169	0.45892	0.00065
Doi-Lo	0.47782	0.48286	0.53072	0.61865	0.00189	0.49484	0.52502	0.58747	0.66477	0.00953
Doi-Tao	0.54363	0.62914	0.58287	0.37828	0.01538	0.62900	0.62594	0.66259	0.39396	0.00946
Hot	0.55885	0.52972	0.52104	0.40086	0.00701	0.57570	0.57262	0.58444	0.38446	0.00446
Mae-Wang	0.46188	0.47452	0.53339	0.61395	0.00201	0.49272	0.55656	0.58503	0.65409	0.00583
Chom-Thong	0.56374	0.50044	0.54690	0.48794	0.00222	0.53762	0.53408	0.58835	0.49300	0.01334
San-Patong	0.39685	0.48559	0.50021	0.45881	0.00151	0.43270	0.58238	0.56961	0.47496	0.00140
Saraphi	0.56847	0.61506	0.58521	0.20920	0.01055	0.56743	0.59601	0.64352	0.20165	0.00459
Li	0.36687	0.55370	0.53048	0.25434	0.00134	0.32791	0.55060	0.49947	0.12781	0.00055
Ban-Hong	0.39613	0.53367	0.46701	0.21140	0.04540	0.42659	0.57218	0.55177	0.21302	0.03010

TMAXN1: Maximum temperature of the first week of November

TMINN1: Minimum temperature of the first week of November

TAVGN1: Average temperature of the first week of November

SAVGN1: Average sunlight of the first week of November

RAVGN1: Average rainfall of the first week of November

TMAXN2: Maximum temperature of the second week of November

TMINN2: Minimum temperature of the second week of November

TAVGN2: Average temperature of the second t week of November

SAVGN2: Average sunlight of the second week of November

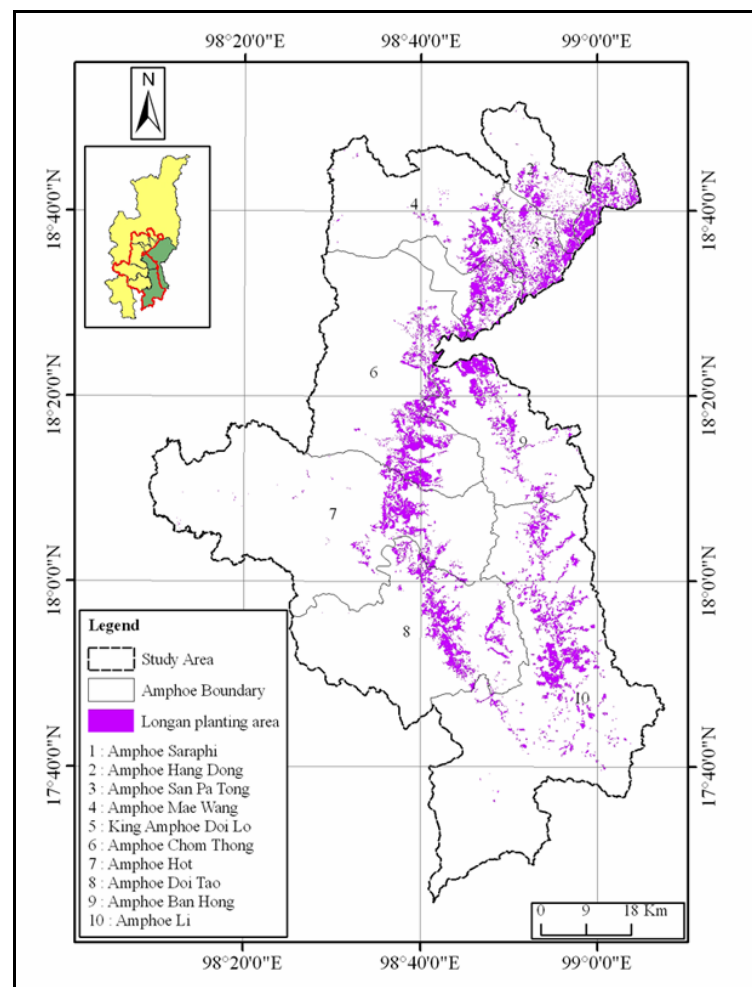
RAVGN2: Average rainfall of the second week of November

#### 4.1.1.2 Land-use and Amphoe data

Digital maps of administrative boundary and land use of the Land Development Department (LDD, 2006) were requisite to select longan planting areas of each Amphoe within the study area. Longan planting areas of Amphoes within the study area and their normalization were listed in Table 4.2. The GIS data layer of planting areas in Amphoes can be displayed as a lay out in Figure 4.8.

**Table 4.2** Longan planting areas of each Amphoe within the study area.

Amphoe	Logan planting area (rai)	Normalized planting area
Hang-Dong	15246	0.0114
Doi-Lo	23316	0.1454
Doi-Tao	14558	0
Hot	15225	0.0110
Mae-Wang	16841	0.0379
Chom-Thong	48330	0.5608
San-Patong	23852	0.1543
Saraphi	22963	0.1395
Li	74769	1
Ban-Hong	37212	0.3762

**Figure 4.8** Longan planting areas in selected Amphoes of Chiangmai and Lamphun.

#### 4.1.1.3 Soil data

Only soil group falling into the longan planting areas within the study area were considered for prediction activity. Total 70 soil group exist in the planting areas. Some soil group covering small areas of each Amphoe were grouped to be “other”. Area extents of soil group within longan planting area of each Amphoe were calculated to be percentage and then normalized to be 0-1 (Table 4.3).

**Table 4.3** Normalized area extents of soil group in longan planting areas of each Amphoe in the study area.

Amphoe	K.Doilo	Ban-Hong	Doi-Tao	Hot	Saraphi	Li	Mae-Wang	Chom-Thong	Hang-Dong	San-Patong
S30	0.00000	0.00000	0.00000	0.00000	0.12290	0.00000	0.00000	0.01186	0.00000	0.00000
S38	0.01092	0.06783	0.00889	0.06950	0.17808	0.00000	0.00000	0.04893	0.00022	0.02794
S5	0.05984	0.00868	0.00000	0.01263	0.40813	0.00018	0.10555	0.00000	0.40813	0.44047
S50	0.00000	0.00000	0.00000	0.00000	0.20771	0.00000	0.00000	0.00000	0.00000	0.00466
S22	0.08495	0.00000	0.00000	0.00000	0.03788	0.00000	0.04444	0.00000	0.07554	0.32499
S35B	0.15799	0.16749	0.13082	0.00919	0.00000	0.02522	0.11228	0.05702	0.00000	0.00071
S48B	0.48463	0.21269	0.23430	0.20317	0.00000	0.24598	0.08197	0.13035	0.00841	0.04295
S48C	0.00957	0.06418	0.04174	0.13021	0.00000	0.24239	0.22699	0.26836	0.13106	0.00855
S48D	0.00157	0.00342	0.02016	0.06925	0.00000	0.05130	0.14689	0.06706	0.04207	0.02694
S62	0.00236	0.01811	0.02131	0.06484	0.00000	0.03062	0.16146	0.04084	0.13179	0.00862
S35	0.00419	0.03492	0.00000	0.02832	0.00000	0.00000	0.00000	0.05664	0.00000	0.02655
S41B	0.06454	0.00000	0.01652	0.02417	0.00000	0.02068	0.06012	0.07506	0.10014	0.00152
S59	0.01386	0.09177	0.09427	0.07393	0.02152	0.02649	0.00164	0.10125	0.01936	0.00000
S44B	0.06664	0.00173	0.11186	0.08725	0.00000	0.02744	0.00000	0.00000	0.00000	0.00000
S60	0.00000	0.00000	0.02209	0.07127	0.00000	0.01696	0.00000	0.00000	0.00000	0.00000
S28B	0.00000	0.00000	0.06928	0.00000	0.00000	0.00013	0.00000	0.00000	0.00000	0.00000
S29B	0.00000	0.06445	0.00918	0.00000	0.00000	0.09489	0.00000	0.00000	0.00000	0.00000
S31B	0.00000	0.00000	0.00000	0.00000	0.00000	0.07465	0.00000	0.00000	0.00000	0.00000
S15	0.00000	0.06738	0.00000	0.00000	0.00659	0.00080	0.00000	0.00000	0.00000	0.00000
SOTHER	0.03894	0.12695	0.05629	0.14372	0.01719	0.10895	0.04039	0.14263	0.08329	0.07664
AREA	0.14546	0.37624	0.00000	0.01108	0.13959	1.00000	0.03792	0.56089	0.01143	0.15436



#### 4.1.1.4 Longan yield data

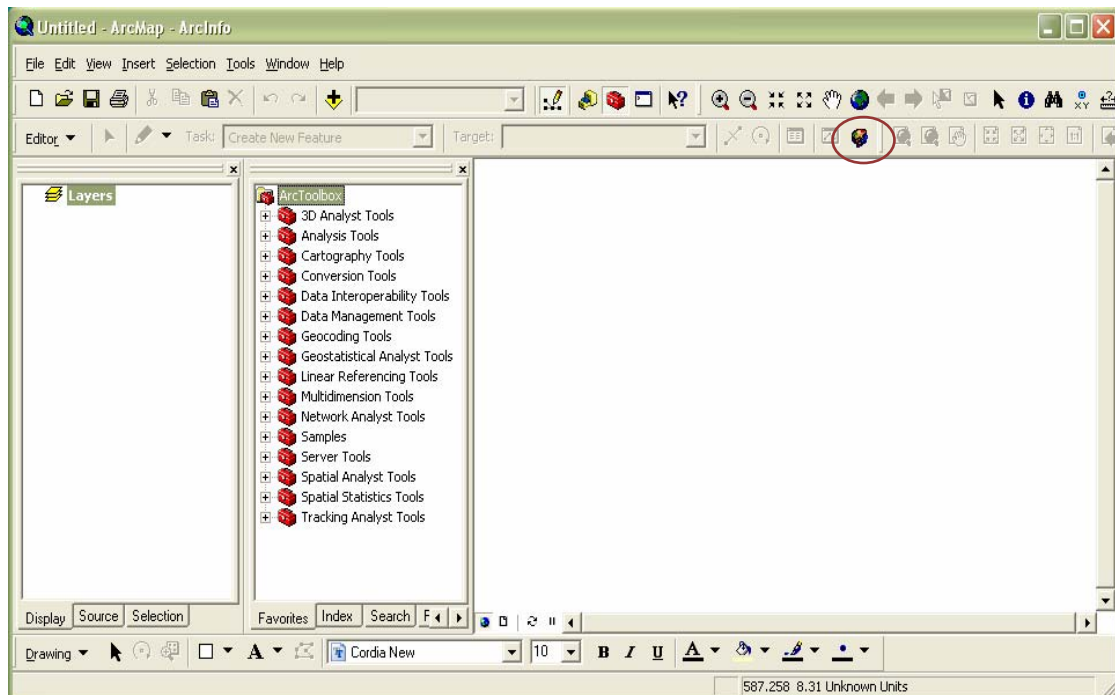
Longan yield data of each Amphoe in year 2007-2008 of the study area are listed in Table 4.4. It means that climate data during November 2006-February 2007 will affect longan yield of the year 2007 and climate data during November 2007-February 2008 will affect the one of the year 2008. The planting areas, soil group involved, and management were considered the same or constant with time for this research. In the final format, the yield data for 2007 and 2008 were tabulated as fields of each record of input data.

**Table 4.4** Longan yields of year 2007 and 2008 and their normalization.

Amphoe	2007 yield	Normalized 2007 yield	2008 yield	Normalized 2008 yield
K.Do-Lo	16,997	0.4219	16,285	0.3996
Ban-Hong	26,831	0.7802	18,270	0.4683
Doi-Tao	5,416	0	6,643	0.0447
Hot	8,648	0.1177	13,006	0.2765
Saraphi	16,051	0.3874	15,826	0.3792
Li	32,863	1	24,074	0.6797
Mae-Wang	12,749	0.2671	9,683	0.1555
Chom-Thong	31,849	0.9630	25,332	0.7256
Hang-Dong	9,986	0.1665	7,840	0.0883
San-Patong	16,005	0.3858	10,061	0.1692

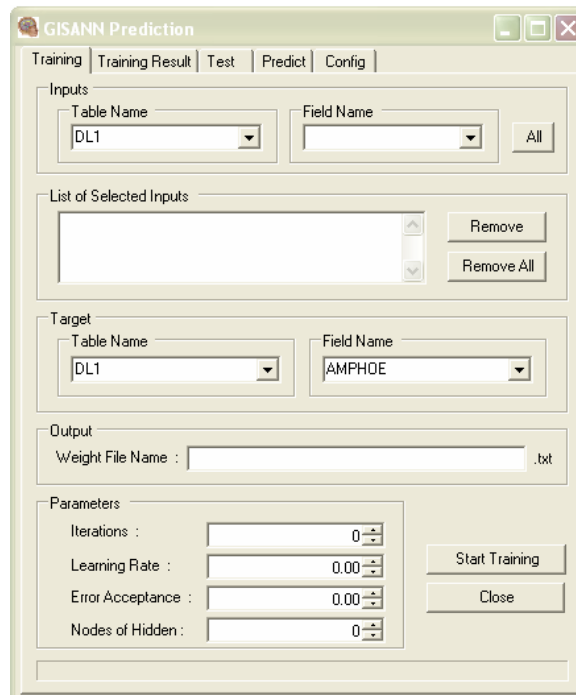
#### 4.1.2 ANN Module Development

After being through processes of analysis, design, implement/coding and test, the module was developed as a GISANN module extension plugged into ArcMap of ArcGIS 9.1 as an icon shown in Figure 4.9. The extension module will be easy to use. Source code of some functions, namely freed forward, train and back propagation are shown as example in Appendix C.

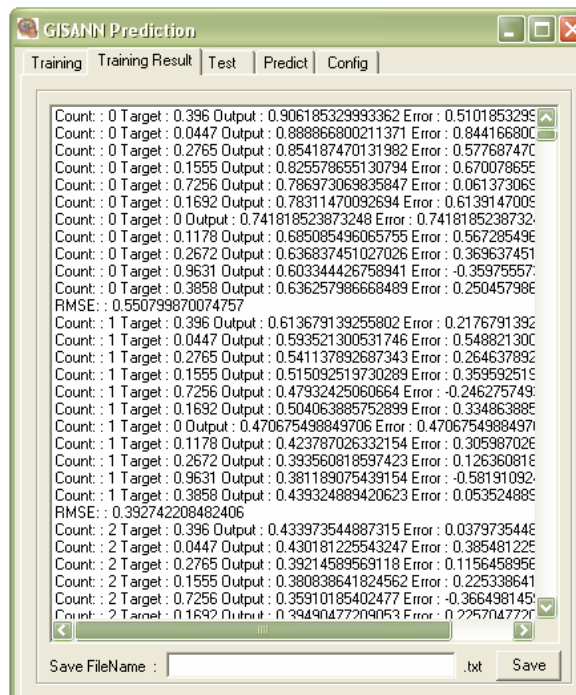


**Figure 4.9** GISANN icon as plugged into ArcMap of ArcGIS 9.1

Under this extension module five tabs are available for yield prediction. These tabs are “training, training result, test, predict, and config” as shown in Figure 4.10. The Figure shows the interface of “Training” tab which allows to input data table with all or selected fields. Input factors can be selected and filed name of target table can be identified. Weight file name will be output. Parameters including iterations, learning rate, error acceptance, and a number of hidden nodes can be varied as required. This tab interface allows the training to operate and the result will be shown in the “Training result” tab that allows the result to be saved as shown in Figure 4.11. This tab will report the error of training.



**Figure 4.10** User interface of “Training” tab of GISANN extension and other four tabs in the back.

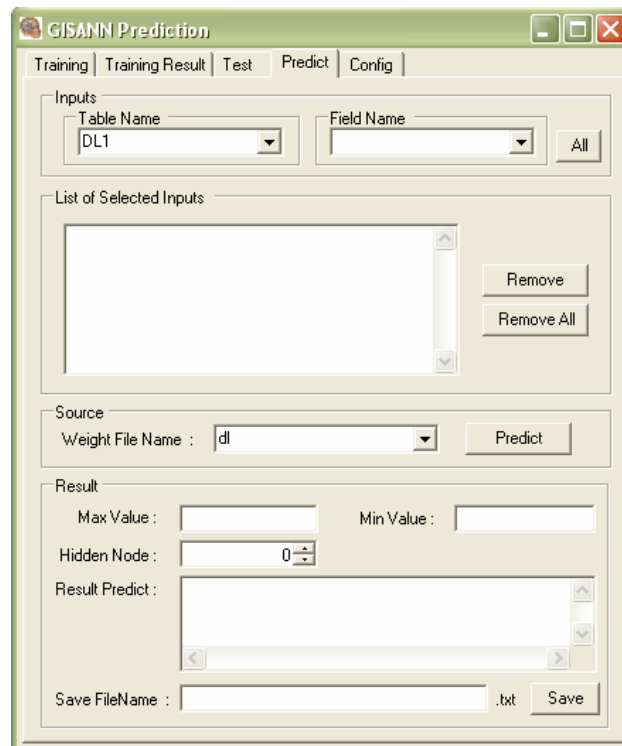


**Figure 4.11** “Training Result” tab.

Figure 4.12 shows the user interface of “Test” tab of which its first half is similar to the “Training” tab. The rest contains weight file name that will be used for testing. The result of the test includes maximum and minimum values of longan yield which was trained. It was the values before normalize, a number of hidden nodes applied, predicted result. The result of testing can be saved for further use.

**Figure 4.12** User interface of “Testing” tab.

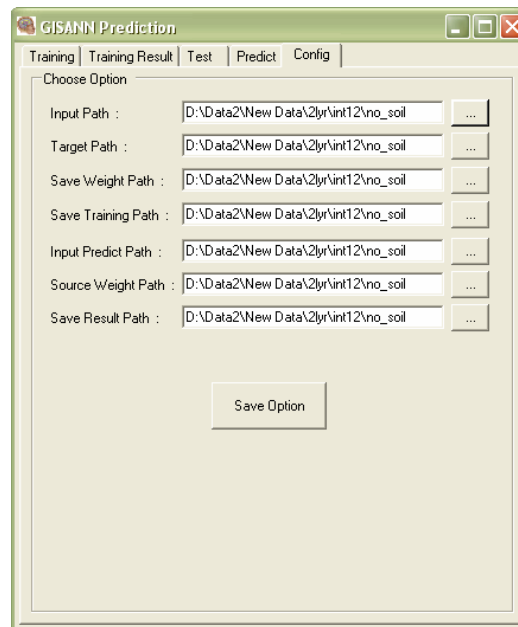
This extension module allows yield to be predicted with the user interface tab shown in Figure 4.13. This tab interface is very similar to the interface of “Testing” tab but no error can be detected.



**Figure 4.13** User interface of “Predict” tab.

“Config” tab interface as shown in Figure 4.14 allows paths of relevant data can be configured. These relevant data include input data to be trained, target data, weight to be saved, training data to be saved, input data to be predicted, source of weight, and result to be saved.

Steps to operate this extension module are explained clearly and practically in Appendix B.



**Figure 4.14** User interface of “Config” tab.

#### 4.1.3 Results of Training and Testing

Results of these processes dealt with variations of model parameters and cross validations of variations of input data. Certain values of the best model parameters including learning rate (LR), a number of hidden nodes, and a number of iterations were able to accomplish from the tests as described in 3.5.1. The tests showed that these values would provide better results or less error when they are applied in further validations of the ANN prediction model.

##### 4.1.3.1 Results of variations of model parameters

The results of choosing the best model parameters from the tests are displayed in Table 4.5. The results of 12 tests are listed to show that which varying value of each model parameter provides the highest average accuracy of all Amphoes. The test results reveal that the best LR, hidden nodes, and a number of iterations are 0.4, 6, and 3000 respectively.

**Table 4.5** Results of 12 tests to obtain the best model parameters which are LR, hidden nodes, and a number of iterations.

No. of Node	Iteration	LR	% Accuracy of each Amphoe										Average %Accuracy
			K.Doit-Lo	Ban-Hong	Doi-Tao	Hot	Saraphi	Li	Mae-Wang	Chom-Thong	Hang-Dong	San-Patong	
4	3000	0.1	91.81	63.88	38.13	84.39	84.92	50.22	97.98	52.29	74.80	84.99	72.34
4	3000	0.2	87.39	90.46	28.84	96.88	71.93	84.96	79.30	50.55	89.75	75.55	75.56
4	3000	0.4	67.79	99.96	73.78	91.30	91.94	86.51	87.95	49.10	79.06	52.76	78.02
4	3000	0.8	68.13	94.19	80.35	63.44	50.93	99.91	67.45	97.09	80.25	41.60	74.33
4	3000	0.4	67.79	99.96	73.78	91.30	91.94	86.51	87.95	49.10	79.06	52.76	78.02
5	3000	0.4	56.86	28.76	51.62	69.18	43.88	98.55	77.69	58.83	60.73	73.68	61.98
6	3000	0.4	78.17	87.14	67.74	75.03	61.85	97.07	90.56	60.32	96.72	86.75	80.13
7	3000	0.4	84.33	83.55	55.17	75.90	83.20	97.23	45.29	61.82	86.98	87.54	76.10
6	3000	0.4	78.17	87.14	67.74	75.03	61.85	97.07	90.56	60.32	96.72	86.75	80.13
6	5000	0.4	84.08	88.37	63.05	68.32	81.35	99.95	74.03	51.98	84.65	63.43	75.92
6	7000	0.4	89.12	29.29	52.71	82.37	64.66	98.05	62.82	80.12	54.25	78.49	69.19
6	10000	0.4	98.61	94.39	67.74	64.12	57.62	97.52	56.11	86.30	69.56	97.92	78.99

The module was developed to allow either of the number of iterations or error acceptance adjustable. However, the upper limit of the number of iterations was set at 5000. It means that even though the error acceptance set is still not reached, the process will be stopped when the number of iterations runs to 5000. In other word, the process will be stopped when the error acceptance set is reached before 5000 iterations.

#### 4.1.3.2 Results of cross validations of variations of input data

As discussed in 3.5.3, 21 of variations of input data were training and testing in order that the best variation can be determined. These basically include variations of climate data in different temporal ranges, certain climate factors selected, and soil group. It is noticeable that planting area and temperature are regularly used. For 1-hidden-layer, the results from training and testing of the cross validations of all variations are listed in Table 4.6. The results reveal that the variation no. 7 (temperature and sunlight of 16 week since the first week of November to the

end of February, 10 records, soil data, and no rain) provided the best accuracy. For 2-hidden-layer, the results show in Table 4.7 and the best accuracy is the variation no.16 (temperature, sunlight and rainfall of 8 week since the first week of December to the end of January, 10 records, and all the rest). Therefore, the combination set of the input data with respect to a certain module will be further used for prediction.

#### 4.1.4 Prediction

After having been through many validation tests and trainings discussed above, the model parameters and the combination sets of input data with the best accuracy were selected to configure the modules. The best model parameters from the trainings and tests were 0.4, 6, and 3000 for LR, hidden nodes, and a number of iterations respectively. With these model parameters configured, the input datasets for both 1-hidden-layer and 2-hidden layer modules were followed the variation numbers mentioned in 4.1.3.2. Table 4.8 shows input data of the year 2009. Result of predictions of the year 2007, 2008, and 2009 of each Amphoe were reported in Table 4.9 for 1-hidden-layer module and Table 4.10 for 2-hidden-layer module. The comparisons between predicted and actual yields of each Amphoe in years 2007 and 2008 using 1-hidden-layer and 2-hidden-layer modules were displayed in Figure 4.15 and 4.16 respectively. During this study the actual yield of the year 2009 has not been available yet. This process and the output proved that the model and software modules work correctly. In this case, the predicted result of the year 2009 can be expected to be as accurate as 90.14% for 1-hidden-layer module and 88.5% for 2-hidden-layer module which are adopted from the best of the tests in 4.1.3.2. However, it has to be in the restricted condition that all data of weather stations must be available.



**Table 4.6** Results of cross validations of all variations of input data for 1-hidden-layer module (according to Table 3.5).

Variation	2007											2008											Average of Accuracy
	K.Do Lo	Ban Hong	Doi Tao	Li	Chom Thong	Saraph i	San Patong	Hot	Mae Wang	Hang Dong	K.Do Lo	Ban Hong	Doi Tao	Li	Chom Thong	Saraph i	San Patong	Hot	Mae Wang	Hang Dong			
1	94.58	83.20	78.64	97.27	99.00	95.76	76.51	86.69	84.30	87.53												88.35	
2	53.39		78.64		78.73		88.43	51.50	94.02		57.38		91.25		94.93		40.41	79.37	89.80			74.82	
3	78.89	96.18	76.03	94.60	80.81	93.32	81.53	92.96	72.76	74.26	90.39	81.43	94.76	80.90	86.93	95.55	96.19	92.15	78.03	86.19		86.19	
4	92.49	89.05	89.72	98.88	99.81	70.67	82.42	92.36	96.99	80.04												89.24	
5	67.41		67.50		77.74		65.15	83.22	92.20		56.17		81.23		76.05		77.45	58.92	98.04			75.09	
6	90.95	84.23	70.13	96.41	86.82	92.04	87.27	79.01	76.85	84.89	94.07	77.59	75.31	88.65	96.69	72.84	91.29	53.00	75.30	81.52		82.74	
7	95.82	92.95	79.41	98.09	95.95	83.41	85.29	81.90	93.51	95.05												90.14	
8	49.21		68.45		79.61		89.88	59.37	77.28		61.95		86.24		76.49		58.75	74.40	83.06			72.06	
9	75.62	77.09	56.22	85.64	80.69	71.95	73.28	76.98	84.87	79.57	97.99	95.10	74.23	95.23	77.70	96.21	96.94	81.31	95.25	82.26		82.71	
10	97.14	93.62	76.26	96.59	82.78	86.48	90.78	93.84	79.07	88.51												88.51	
11	54.35		81.92		79.62		63.60	60.88	92.41		75.95		82.10		81.67		73.35	53.66	93.65			74.43	
12	68.57	83.27	59.51	84.42	81.40	85.36	65.95	56.70	82.13	95.12	68.89	79.16	90.53	94.87	77.70	76.86	74.87	63.79	84.99	99.35		78.67	
13	74.88	92.31	79.06	99.67	98.84	98.12	80.19	79.36	77.97	76.06												85.65	
14	73.25		53.81		81.06		93.50	92.01	71.4		95.97		51.67		83.35		62.79	82.68	90.75			77.69	
15	86.93	88.84	66.63	89.37	51.92	69.56	75.30	97.77	68.48	70.18	84.13	91.43	90.43	76.06	76.42	67.03	84.61	68.98	53.67	75.06		76.64	
16	75.46	80.31	65.45	96.32	95.68	97.03	97.73	81.54	74.87	98.14												86.25	
17	80.09		71.10		76.52		62.12	66.18	96.51		72.85		85.64		75.44		16.17	62.10	93.20			74.83	
18	79.35	71.07	70.75	81.63	79.32	72.11	70.55	69.92	82.78	73.62	60.99	51.70	70.29	71.23	68.78	72.84	69.35	71.59	71.94	73.33		71.66	
19	97.14	93.62	76.26	96.59	92.78	86.48	90.78	93.84	79.07	88.51												89.51	
20	85.27		67.24		82.09		76.18	92.18	75.44		66.13		70.53		75.15		20.85	63.09	90.21			72.03	
21	67.96	86.73	55.89	89.76	78.14	74.79	96.01	83.20	98.85	87.91	66.46	79.83	79.20	75.21	82.42	91.65	78.84	71.79	64.30	78.52		79.37	

**Note**

10 records : Variation no. : 1, 4, 7, 10, 13, 16, 19      Average of accuracy : 88.23  
 12 records : Variation no. : 2, 5, 8, 11, 14, 17, 20      Average of accuracy : 74.42  
 20 records : Variation no. : 3, 6, 9, 12, 15, 18, 21      Average of accuracy : 79.71

**Table 4.7** Results of cross validations of all variations of input data for 2-hidden-layer module (according to Table 3.6).

Variation	2007										2008										Average of Accuracy		
	K.Do	Lo	Ban Hong	Doi Tao	Li	Chom Thong	Saraph i	San Patong	Hot	Mae-Wang	Hang Dong	K.Do	Lo	Ban Hong	Doi Tao	Li	Chom Thong	Saraph i	San Patong	Hot		Mae-Wang	Hang Dong
1	80.50	71.34	52.66	97.82	87.53	85.58	96.01	93.91	82.48	99.92													84.78
2	84.77		67.43		69.12		69.16	81.05	86.94		88.47		71.70		79.03		78.15	87.24	85.91				79.08
3	97.37	90.32	49.58	95.89	82.76	91.23	87.74	99.87	74.89	64.96	97.97	96.54	95.51	67.48	90.04	95.63	98.50	71.18	67.62	83.28			84.92
4	87.99	71.34	42.28	77.20	80.98	67.30	98.36	91.76	86.56	60.97													76.47
5	99.90		58.84		59.79		55.35	75.58	95.21		74.23		85.01		79.91		47.23	72.97	76.66				73.39
6	87.39	71.34	59.27	97.01	99.88	90.67	95.52	79.69	87.21	95.40	82.47	95.24	48.68	73.31	83.09	84.20	64.94	82.45	70.28	80.84			81.44
7	86.73	80.99	42.28	86.30	88.94	62.19	86.85	69.94	94.31	72.58													77.11
8	62.42		74.69		86.75		98.95	91.36	80.41		62.33		71.22		71.35		86.74	69.11	80.35				77.97
9	95.50	72.91	47.19	95.25	72.86	78.38	97.13	79.58	92.25	34.44	83.26	90.71	52.45	64.13	72.64	69.44	41.51	52.25	87.01	77.50			72.82
10	52.17	92.35	74.30	95.06	87.68	72.40	93.01	79.79	87.37	75.85													81.00
11	96.79		71.23		75.45		83.67	70.29	93.74		70.56		70.53		73.20		57.18	82.74	77.63				76.92
12	98.96	77.20	65.21	96.50	89.86	72.82	79.47	73.49	99.74	88.94	74.82	85.96	97.89	74.88	83.10	74.51	76.79	88.82	94.27	98.75			84.60
13	87.85	84.35	72.25	94.77	97.51	98.44	72.43	83.07	78.39	81.32													85.04
14	86.81		81.41		77.23		70.33	94.87	73.76		84.72		57.82		82.69		82.79	74.73	96.14				80.27
15	97.84	96.40	74.50	72.28	98.90	74.86	97.64	74.16	75.50	94.50	59.12	93.02	69.58	67.88	98.08	50.02	82.24	73.18	98.89	84.50			81.65
16	74.23	80.20	92.61	96.16	95.89	97.00	98.66	70.73	81.98	97.51													88.50
17	97.43		62.04		75.02		72.44	71.29	78.25		74.02		76.56		65.75		57.17	76.93	69.09				73.72
18	90.24	77.78	58.01	96.54	72.81	89.35	80.34	72.70	94.86	78.46	94.30	65.31	67.26	76.83	73.66	97.55	48.07	63.42	67.65	95.70			78.04
19	76.30	79.94	97.82	92.33	86.67	93.73	86.73	99.54	81.98	73.43													86.85
20	77.02		71.38		76.62		70.37	99.85	84.37		78.39		89.88		69.62		56.87	71.55	80.82				77.23
21	97.86	87.90	51.86	96.04	97.51	65.86	87.58	84.08	81.00	95.51	84.39	81.44	47.25	64.42	70.56	84.94	31.46	67.05	94.02	70.96			77.08

**Note**

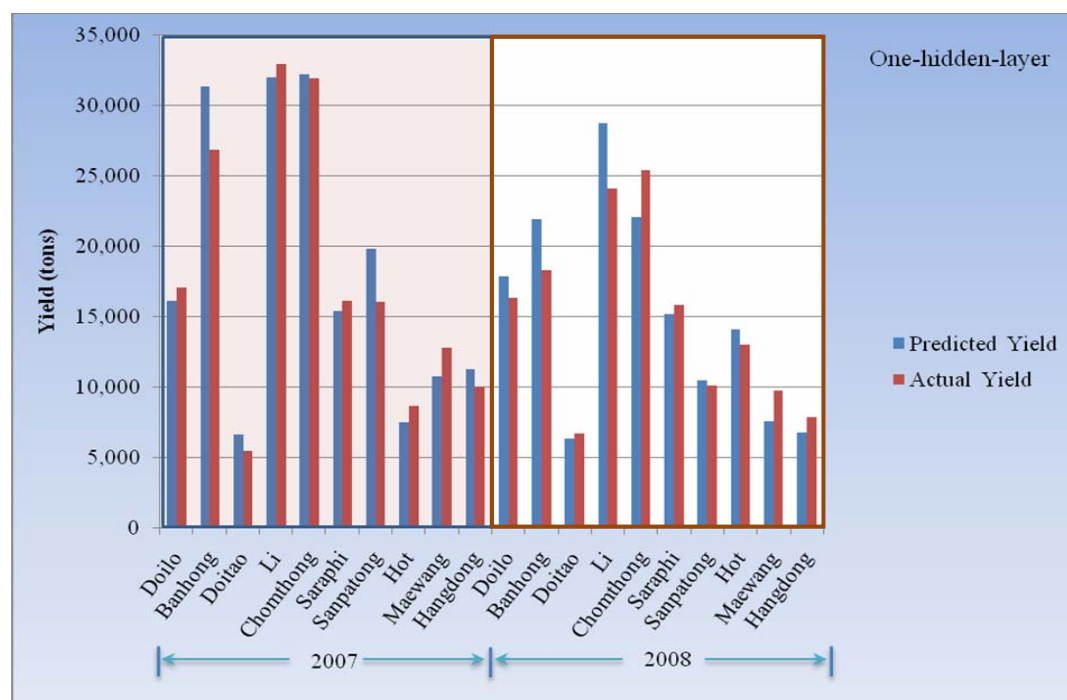
10 records : Variation no. : 1, 4, 7, 10, 13, 16, 19      Average of accuracy : 82.82  
 12 records : Variation no. : 2, 5, 8, 11, 14, 17, 20      Average of accuracy : 76.94  
 20 records : Variation no. : 3, 6, 9, 12, 15, 18, 21      Average of accuracy : 80.08

**Table 4.8** The input data of the year 2009 for the model for prediction.

Amphoe	K.Do Lo	Ban Hong	Doi Tao	Hot	Li	Mae Wang	Chom Thong	Saraphi	San Patong	Hang Dong
TMAXN1	0.5433	0.6817	0.6125	0.4913	0.4602	0.7474	0.7301	0.5260	0.6609	0.7405
TMINN1	0.8162	0.8529	0.8897	0.9632	0.5662	0.8897	0.9265	0.8971	0.8897	0.9191
TAVGN1	0.5315	0.6076	0.6185	0.6078	0.4776	0.6463	0.6930	0.6356	0.5761	0.6798
SAVGN1	0.1600	0.1400	0.1190	0.1800	0.1840	0.3151	0.0665	0.1133	0.9560	0.1889
RAVGN1	0.0000	0.0004	0.0197	0.7125	0.0004	0.8438	0.0109	0.0101	0.0750	0.0750
TMAXN2	0.4740	0.7474	0.5260	0.5260	0.5087	0.6782	0.6609	0.4913	0.5606	0.6713
TMINN2	0.3529	0.5294	0.4191	0.3897	0.4559	0.4191	0.4559	0.3529	0.3897	0.4412
TAVGN2	0.2972	0.7290	0.3875	0.3574	0.5247	0.4355	0.4908	0.3706	0.3631	0.4760
SAVGN2	0.2000	0.1700	0.1542	0.2400	0.1580	0.4102	0.0966	0.2149	0.9640	0.2570
RAVGN2	0.0000	0.0004	0.0000	0.2938	0.0002	0.0063	0.0000	0.0001	0.0000	0.0000
...	...	...	...	...	...	...	...	...	...	...
TMAXF3	0.6782	0.7820	0.8824	0.7301	0.6955	0.6955	0.7301	0.6609	0.7474	0.5779
TMINF3	0.3897	0.5294	0.5662	0.4191	0.4044	0.5662	0.3897	0.2794	0.3897	0.3897
TAVGF3	0.5166	0.7717	0.7504	0.6500	0.6274	0.6553	0.6121	0.4407	0.5332	0.4873
SAVGF3	0.1500	0.1580	0.1535	0.2354	0.2200	0.3491	0.1828	0.2383	0.3152	0.1600
RAVGF3	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0003	0.0000	0.0000
TMAXF4	0.7128	0.9689	0.8997	0.8824	0.7474	0.7993	0.8824	0.6782	0.7647	0.6955
TMINF4	0.5294	0.4926	0.7132	0.5662	0.3897	0.5662	0.5294	0.4559	0.4926	0.4191
TAVGF4	0.5342	0.9378	0.8496	0.8206	0.6814	0.7649	0.6604	0.5087	0.6867	0.5716
SAVGF4	0.1680	0.1800	0.1822	0.2391	0.2000	0.3207	0.1930	0.2317	0.2897	0.1740
RAVGF4	0.0002	0.0000	0.0000	0.0000	0.0000	0.0000	0.0003	0.0003	0.0000	0.0000
S30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0119	0.1229	0.0000	0.0000
S38	0.0109	0.0678	0.0089	0.0695	0.0000	0.0000	0.0489	0.1781	0.0279	0.0002
S5	0.0598	0.0087	0.0000	0.0126	0.0002	0.1056	0.0000	0.4081	0.4405	0.4081
S50	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2077	0.0047	0.0000
S22	0.0850	0.0000	0.0000	0.0000	0.0000	0.0444	0.0000	0.0379	0.3250	0.0755
S35B	0.1580	0.1675	0.1308	0.0092	0.0252	0.1123	0.0570	0.0000	0.0007	0.0000
S48B	0.4846	0.2127	0.2343	0.2032	0.2460	0.0820	0.1304	0.0000	0.0430	0.0084
S48C	0.0096	0.0642	0.0417	0.1302	0.2424	0.2270	0.2684	0.0000	0.0086	0.1311
S48D	0.0016	0.0034	0.0202	0.0693	0.0513	0.1469	0.0671	0.0000	0.0269	0.0421
S62	0.0024	0.0181	0.0213	0.0648	0.0306	0.1615	0.0408	0.0000	0.0086	0.1318
S35	0.0042	0.0349	0.0000	0.0283	0.0000	0.0000	0.0566	0.0000	0.0266	0.0000
S41B	0.0645	0.0000	0.0165	0.0242	0.0207	0.0601	0.0751	0.0000	0.0015	0.1001
S59	0.0139	0.0918	0.0943	0.0739	0.0265	0.0016	0.1013	0.0215	0.0000	0.0194
S44B	0.0666	0.0017	0.1119	0.0873	0.0274	0.0000	0.0000	0.0000	0.0000	0.0000
S60	0.0000	0.0000	0.0221	0.0713	0.0170	0.0000	0.0000	0.0000	0.0000	0.0000
...	...	...	...	...	...	...	...	...	...	...
S29B	0.0000	0.0645	0.0092	0.0000	0.0949	0.0000	0.0000	0.0000	0.0000	0.0000
S40B	0.0000	0.0647	0.0104	0.0125	0.0238	0.0183	0.0000	0.0000	0.0094	0.0000
S46B	0.0000	0.0000	0.0000	0.0000	0.0054	0.0000	0.0000	0.0000	0.0000	0.0000
SOTHER	0.0389	0.1270	0.0563	0.1437	0.1090	0.0404	0.1426	0.0172	0.0766	0.0833
AREA	0.1455	0.3762	0.0000	0.0111	1.0000	0.0379	0.5609	0.1396	0.1544	0.0114

**Table 4.9** Difference of predicted yield and actual yield for two years using 1-hidden-layer module.

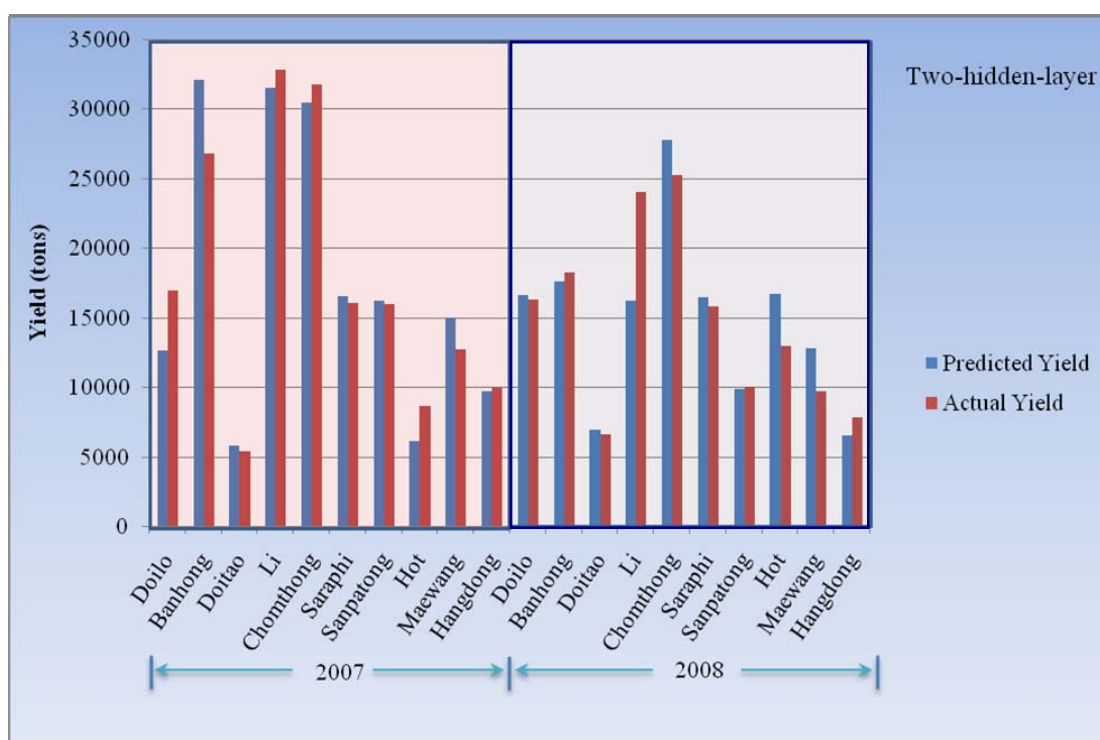
Amphoe	2007				2008				2009
	Predicted Yield (tons)	Actual Yield (tons)	Difference	Accuracy (%)	Predicted Yield (tons)	Actual Yield (tons)	Difference	Accuracy (%)	Predicted Yield
Doilo	16,076	16,997	-921	94.58	17,850	16,285	1,565	90.39	17,707
Banhong	31,339	26,831	4,508	83.20	21,849	18,270	3,579	80.41	9,305
Doitao	6,573	5,416	1,157	78.64	6,295	6,643	-348	94.76	9,748
Li	31,967	32,863	-896	97.27	28,671	24,074	4,597	80.90	32,235
Chomthong	32,166	31,849	317	99.00	22,022	25,332	-3,310	86.93	30,477
Saraphi	15,371	16,051	-680	95.76	15,122	15,826	-704	95.55	18,374
Sanpatong	19,764	16,005	3,759	76.51	10,444	10,061	383	96.19	18,184
Hot	7,497	8,648	-1,151	86.69	14,027	13,006	1,021	92.15	11,095
Maewang	10,748	12,749	-2,001	84.30	7,556	9,683	-2,127	78.03	6,349
Hangdong	11,231	9,986	1,245	87.53	6,757	7,840	-1,083	86.19	6,438
Sum	182,732	177,395	5,337		150,593	147,020	3,573		159,912



**Figure 4.15** Forecast performance of 1-hidden-layer module during year 2007 and 2008.

**Table 4.10** Difference of predicted yield and actual yield for two years using 2-hidden-layer module.

Amphoe	2007				2008				2009
	Predicted Yield (tons)	Actual Yield (tons)	Difference	Accuracy (%)	Predicted Yield (tons)	Actual Yield (tons)	Difference	Accuracy (%)	Predicted Yield
Doilo	12,617	16,997	-4380	74.23	16,615	16,285	330	97.97	19,529
Banhong	32,143	26,831	5,312	80.20	17,637	18,270	-633	96.54	22,964
Doitao	5,816	5,416	400	92.61	6,941	6,643	298	95.51	8,692
Li	31,601	32,863	-1262	96.16	16,246	24,074	-7,828	67.48	31,608
Chomthong	30,541	31,849	-1308	95.89	27,856	25,332	2,524	90.04	27,961
Saraphi	16,533	16,051	482	97.00	16,517	15,826	691	95.63	15,649
Sanpatong	16,220	16,005	215	98.66	9,910	10,061	-151	98.50	16,206
Hot	6,117	8,648	-2,531	70.73	16,754	13,006	3,748	71.18	5,570
Maewang	15,046	12,749	2,297	81.98	12,818	9,683	3,135	67.62	7,551
Hangdong	9,737	9,986	-249	97.51	6,529	7,840	-1,311	83.28	5,651
Sum	176,371	177,395	-1,024		147,823	147,020	803		161,381



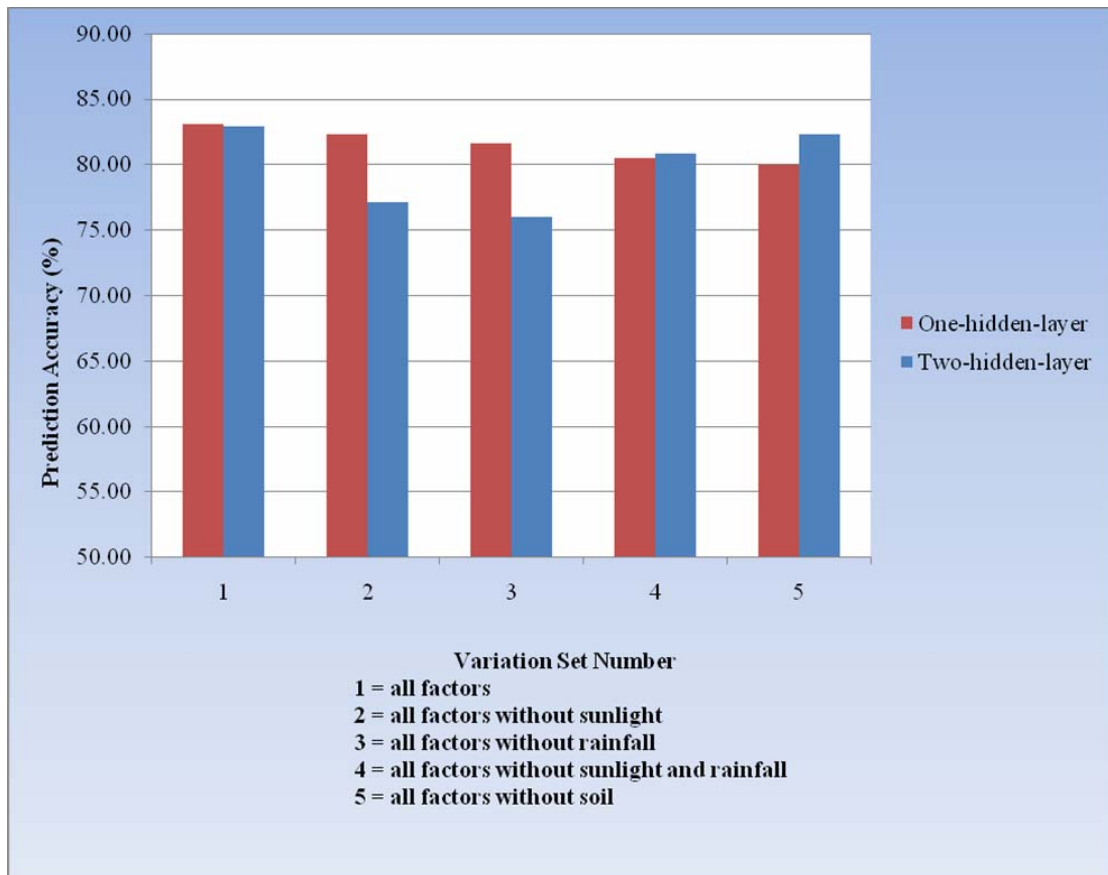
**Figure 4.16** Forecast performance of 2-hidden-layer module during year 2007 and 2008.

## 4.2 Results discussion

Results obtained from the study were the prediction results of longan productions of the year 2007 and 2008. They were obtained from varying inputs through training and testing processes of the model. These input variations cover varying types of physical properties, varying weekly data used, and varying a number of Amphoe of which their data were used. The data covering different number of Amphoe used will reflect spatially by using different number of records used which are 10, 12, and 20. Then, when other physical properties and weekly data used, it means that those variations have to go through training and testing processes of each different number of records used or different variation sets (see Table 3.5 and 3.6). The prediction results of production using 1-hidden-layer and 2-hidden-layer modules were considered and discussed in terms of accuracy by comparing to known values.

### 4.2.1 Varying input factors

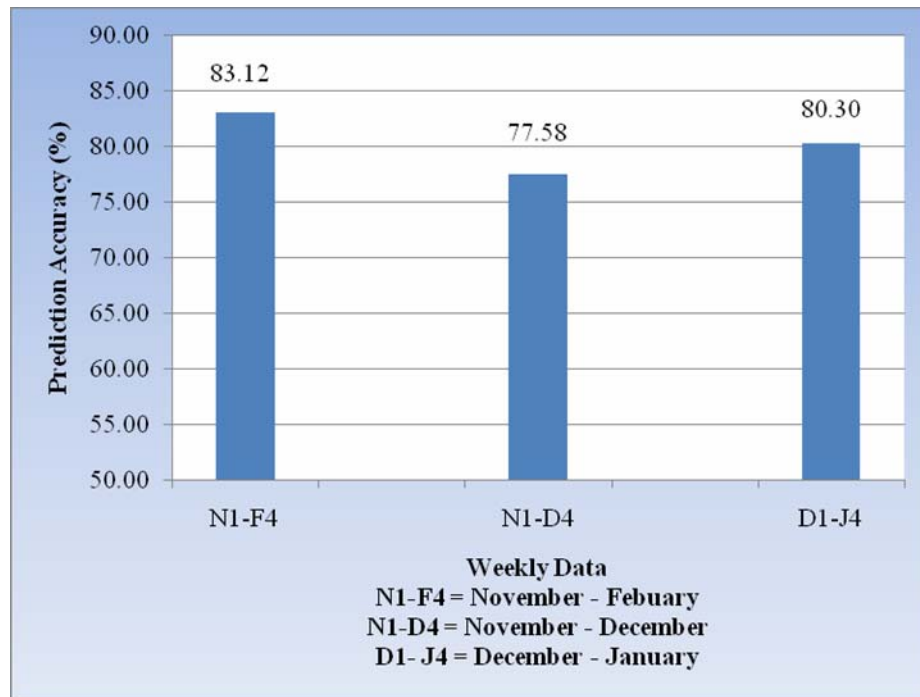
The average prediction accuracy of variation set number 1 to 5 (Figure 4.17) are between 80-83%. The best accuracy of both modules appeared when all input factors were used. Without either sunlight or rain data, results from 2-hidden-layer module are fairly lower than others. This might indicate that these data could effect to each other. However, with both of them or without soil, it does not affect the accuracy much.



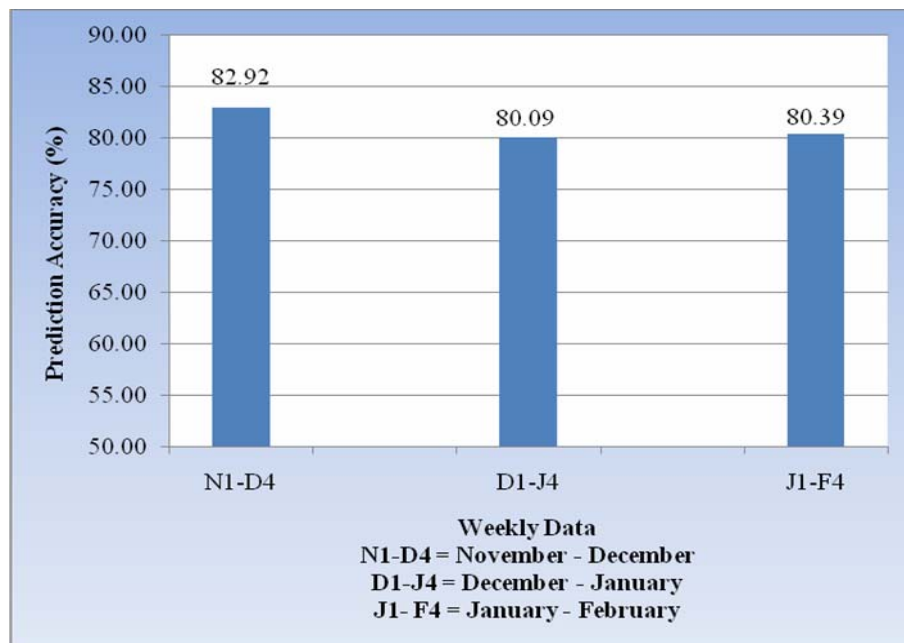
**Figure 4.17** Shows the comparative accuracy of the different chosen inputs.

#### 4.2.2 Varying number of weekly data

From 1-hidden-layer (Figure 4.18), weekly data of January and February show better accuracy than others in range of 2-6% which is not that high. From 2-hidden-layer (Figure 4.19), weekly data of November and December show better accuracy than others in range of 4-5% which is not that high too. It could be concluded that varying number of weekly data during November-February show not much significance on influencing the longan production. Anyhow, the more weekly data in this duration used, it seems to show a little higher accuracy. This is corresponding to what mentioned in Pawin Manochai (พาวิน, 2545).



**Figure 4.18** The comparative accuracy of the different weekly data used of 1-hidden-layer testing.

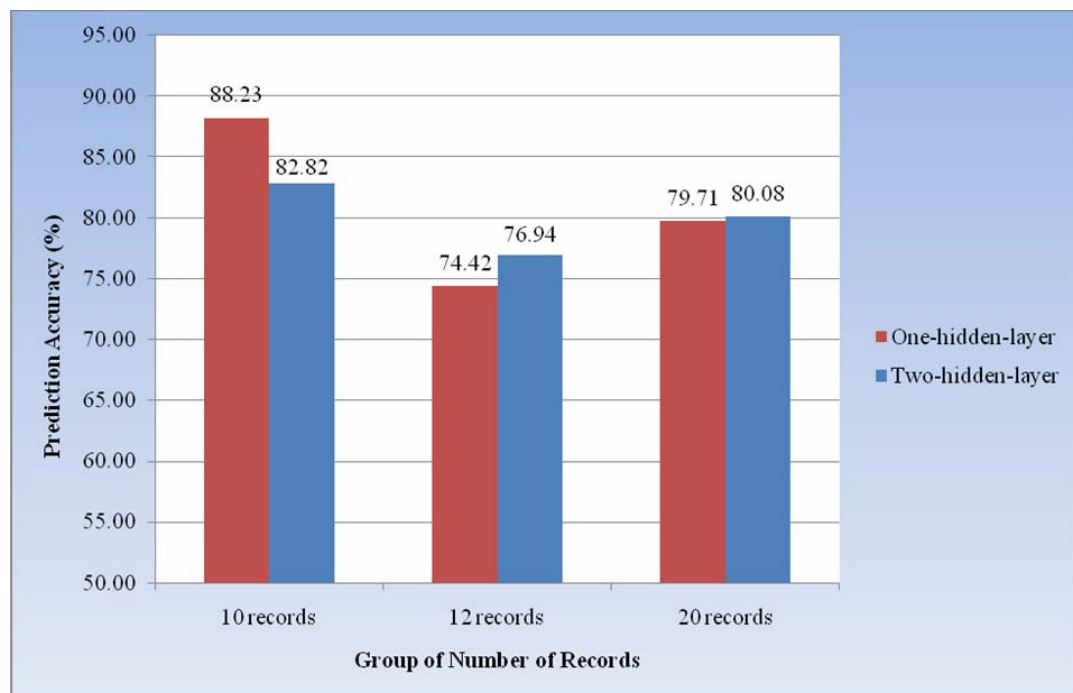


**Figure 4.19** The comparative accuracy of the different weekly data used of 2-hidden-layer testing.



#### 4.2.3 Varying number of record used

The 10 and 20 records in Figure 4.20 cover the same spatial variability (see Table 4.6) but 20 records have additional data of the year 2008. The 12 records cover different spatial variability. In this case the result of 10 records shows higher prediction accuracy than 20 records. This might be because of data of the year 2008 are plenty of no measurement. In addition, it expresses that the higher spatial variability shows tendency of having higher accuracy. The accuracy resulting from using 1-and 2-hidden-layer modules are corresponding to each other.



**Figure 4.20** shows the average accuracy of the data records

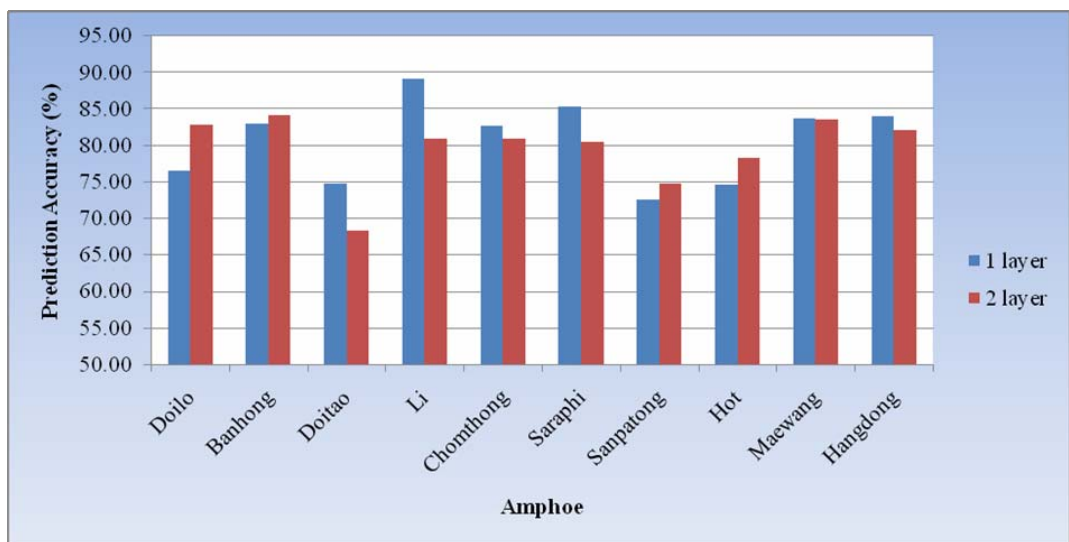
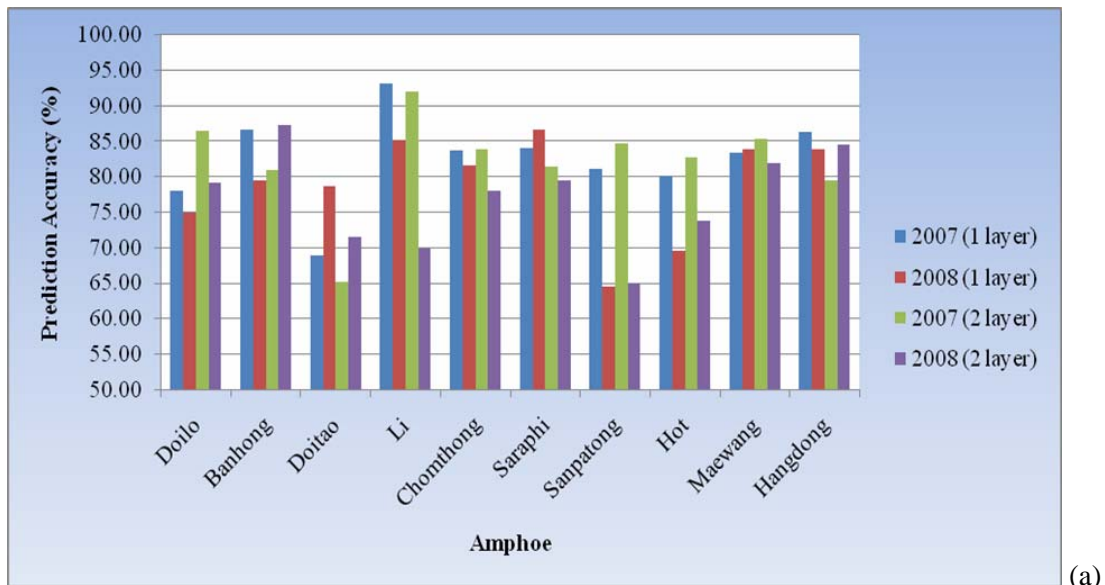
#### 4.2.4 Comparing predicted yields from 1-hidden-layer and 2-hidden-layer modules with actual yields

Figure 4.21 (a) shows that result from both modules of data in year 2007 are more corresponding than data in year 2008. This can confirm that data of the year

2007 are more complete and trustable. Comparing between the results of two modules, it cannot be concluded that which one perform better. However, this conclusion could be supported by the DTREG, the specific prediction software developing company, who states on-line (<http://www.dtreg.com/mlfn.htm>) that “For nearly all problems, one hidden layer is sufficient. Two hidden layers are required for modeling data with discontinuities such as a saw tooth wave pattern. Using two hidden layers rarely improves the model, and it may introduce a greater risk of converging to a local minima. There is no theoretical reason for using more than two hidden layers”.

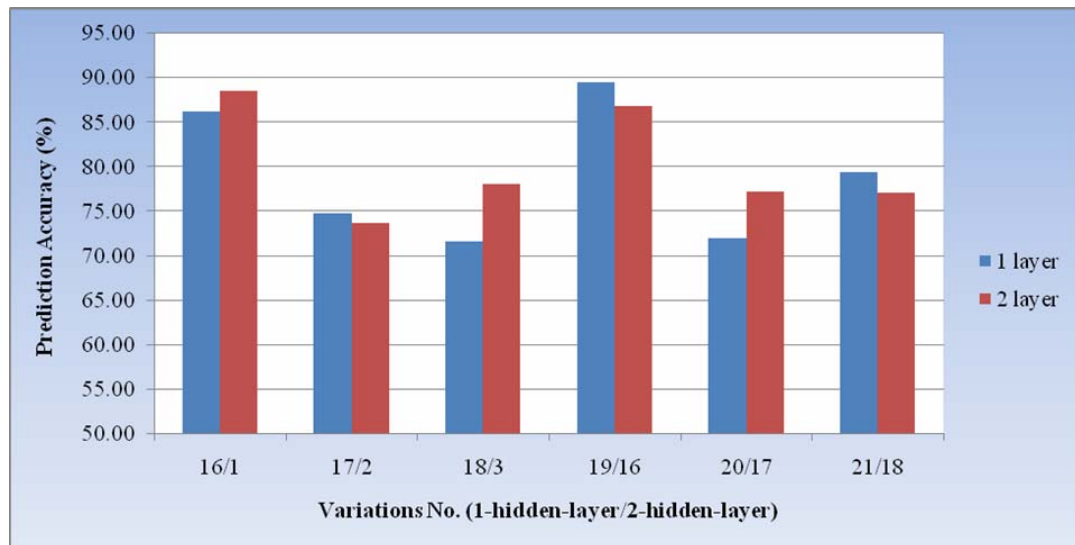
However, when accuracies of both years are averaged (Figure 4.21 (b)), the overall accuracy looks better. This means that if more trustable data involved in the process, the model performance can provide better result in general.

Figure 4.22 shows the comparison of prediction accuracies resulted from both modules when all variation of input data are the same. The average accuracy of 1-hidden-layer and 2-hidden-layer were 78.94% and 80.24% respectively. It also cannot be surely concluded which module performs better.



(b)

**Figure 4.21** (a) shows accuracy comparison of both modules according to data of the year 2007 and 2008 and in different Amphoes; (b) show accuracy comparison of both modules when accuracy of both year are averaged.



**Figure 4.22** shows accuracy comparison of both modules according to average data of variation number 16/1, 17/2, 18/3, 19/16, 20/17, and 21/18 in form of the comparison of variation number in Table 3.5/variation number in Table 3.6.

Figure 4.23 shows yield comparison of both modules and the actual yields according to different Amphoes and years of data. It shows that the actual yield corresponding quite well with the predicted yields from both modules particularly in the year 2007 of which data are more trustable. This comparison relies on the best average accuracy resulted from using the certain variation number.

In fact, if the best variation of each Amphoe (see Table 4.6 and 4.7) were input to the modules, the results can provide better accuracy as shown Figure 4.24.

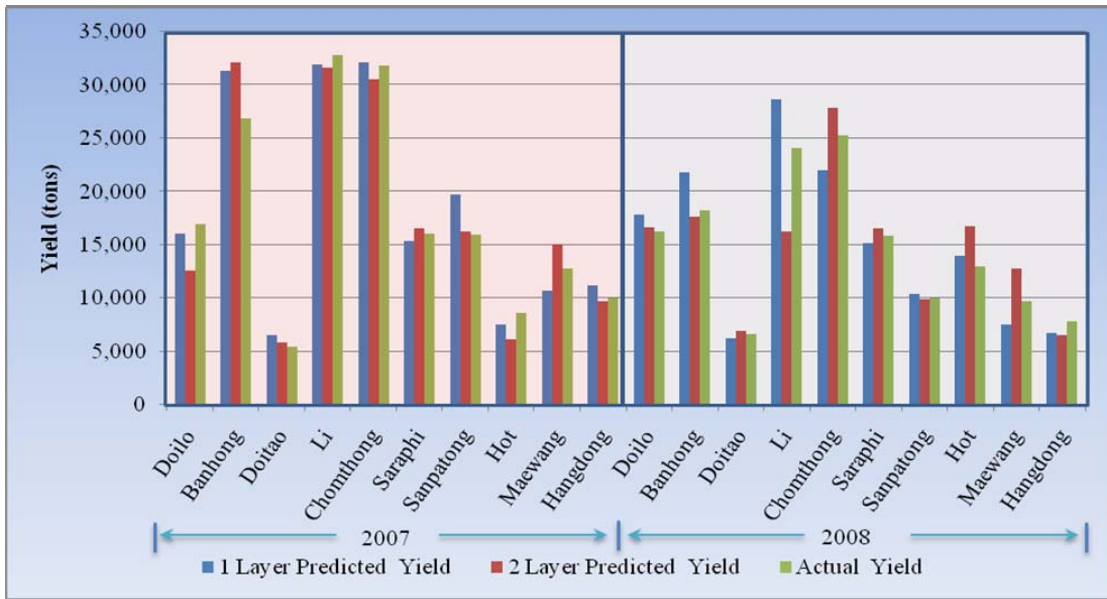


Figure 4.23 shows yield comparison of both modules and the actual yields according to different Amphoes and years of data.

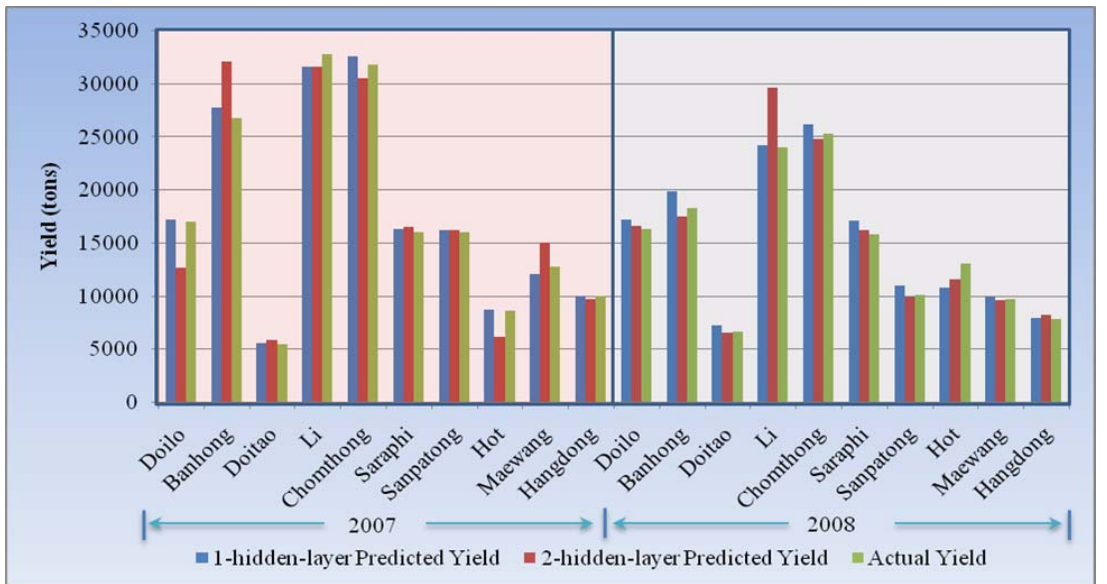


Figure 4.24 shows yield comparison of both modules and the actual yields according to different Amphoes and years of data when no need to be the same variation number is considered.

## **CHAPTER V**

### **CONCLUSION AND RECOMMENDATION**

#### **5.1 Conclusions**

The ANN modules for longan production prediction were developed using the Object-Oriented technique for both the designing and programming. VB.NET is a language applied to this coding. Relevant implement references available in ArcMap were embedded in the module coding. Thus, these modules can work as an extension of the ArcMap. The modules include 1-hidden-layer and 2-hidden-layer architecture of the ANN model.

The ANN modules developed, as an extension of the ArcMap, were applied to the longan yield prediction in the Chiangmai and Lamphun provinces. The climatic data used for training and testing processes cover data of years 2006-2008 from HAIL. The normalized (between 0 and 1) weekly highest, lowest and average temperature, average sunlight and rainfall were interpolated to distribute over the study area and then averaged to spatially represent in forms of each Amphoe. The soil data prepared by the Land Development Department were used in terms of percentage of existing soil groups in cultivated area of each Amphoe. The actual longan yields in years 2007 and 2008 were referred to the Office of Agricultural Economics and the Office of Chiang Mai Agriculture. All of these normalized data were arranged in the .dbf format for training and testing processes of the model. These inputs were varied to be several variations and result accuracies obtained from comparing the predictive

and actual values were observed and discussed. The cross validation process was applied to each record and variation. The optimal parameters such as learning rate, number of nodes in the hidden layer, and number of iteration obtained from testing were 0.4, 6, and 3,000 respectively. These parameters were applied for all training and testing processes.

The testing results can be concluded as follow:

1) The best accuracies obtained from applying 1-hidden-layer and 2-hidden-layer ANN modules are 90.14% and 88.50% respectively.

2) Considering physical data used, the average prediction accuracies of variation sets are between 80-83%. The best accuracy appeared when all input factors were used. Without either sunlight or rain data, results from 2-hidden-layer module are fairly lower than others. This might indicate that these data could effect to each other. However, with both of them or without soil, it does not affect the accuracy much.

3) Considering varying number of weekly data, from 1-hidden-layer, weekly data of January and February show better accuracy than others in range of 2-6% which is not that high. It could be concluded that varying number of weekly data during November-February show not high significance on influencing the longan production. Anyhow, the more weekly data in this duration used, it seems to show a little higher accuracy.

4) Considering number of records used, the result of 10 records shows higher prediction accuracy than 20 records. This might be because of data of the year 2008, which are in 12 and 20 records, are plenty of no measurement. In addition, it expresses that the higher spatial variability shows tendency of having higher

accuracy. The accuracy resulting from using 1-hidden-layer and 2-hidden-layer modules are corresponding to each other.

5) Considering the performance of 1-hidden-layer and 2-hidden layer modules, it cannot be definitely concluded that which one perform better. Beneficially, if these two modules are applied, it can differentiate that data from which year are more valid and trustable. From this study, it confirms that data of the year 2007 are more valid and trustable.

The ANN modules developed in this study are able to predict the longan yield. The modules allow user to set parameters for model operation. These parameters include the learning rate, number of iterations, number of hidden nodes, and error acceptance. In addition, the user can also vary fields of input data.

Nevertheless, the limitation of using 2-hidden-layer module in the certain technological environment of the study is existed. According to the architecture of the 2-hidden-layer module, the process performance is increased tremendously and cause low accurate and invalid result.

From training and testing processes of the study, it is also found that validity of input data used is very significant. The result of the study shows that results using data of the year 2007 is more valid than using data of the year 2008 of which a lot of climate data of stations are missing.

Modules developed are sufficiently flexible to be applicable to yield prediction of other crops. Certain types of crops could have their own factors influencing yield productivity. If they can be prepared to meet the requirement of input format of the modules developed, their yield prediction can be achieved by using these modules.



## 5.2 Recommendations and further improvements

From the experience gained from this study, the recommendation for further study that could expect to better result can be as the followings.

1) More valid input data which include climatic and yield data can assure the better and more trustable results.

2) If technological environment in terms of no limitation of the amount of input data for both modules can be allowed, the comparison study between two modules should be more accurate.

3) Genetic Algorithm should be added for providing the initiate set of weight using in the ANN model so that performance time to achieve the better result of the ANN modules can be reduced.

4) Theoretically, the yield of longan in each plot can be naturally good in every second year (พาวิน และคณะ, 2546). If this condition is set to involve in the model, the better result might be expected.

5) Different details in cultivation area can provide different yields. Therefore, if information of each plot relating management, production, and longan age can be input, the better result could be expected too.

6) If soil fertility properties are considered additionally from their coverage areas, the better result could be expected.

## **REFERENCES**

## REFERENCES

- กรมพัฒนาที่ดิน. (2547). รายงานการใช้เทคโนโลยีการสำรวจระยะไกลและระบบสารสนเทศภูมิศาสตร์  
สำรวจและประเมินผลผลิตลำไย ปีการผลิต 2546/47. กระทรวงเกษตรและสหกรณ์.
- กรมวิชาการเกษตร. (2545). เกษตรดีที่เหมาะสมสำหรับลำไย. กระทรวงเกษตรและสหกรณ์.
- กรมวิชาการเกษตร และกรมพัฒนาที่ดิน. (2546). การใช้เทคโนโลยีรีโมทเซนซิงและระบบ  
สารสนเทศ ภูมิศาสตร์เพื่อประเมินผลผลิต ปี 46: ข้าวนาปี, มันสำปะหลัง, อ้อย [ไฟล์  
คอมพิวเตอร์]. กรมพัฒนาที่ดิน กระทรวงเกษตรและสหกรณ์
- กรมศุลกากร. (2551). ตลาดต่างประเทศ [ออนไลน์]: ได้จาก [http://www.thailandlongan.com/  
index.php](http://www.thailandlongan.com/index.php)
- พาวิน มะโนชัย. (2545). ไม้ผลกิ่งเขตร้อน. สาขาไม้ผล ภาควิชาพืชสวน คณะผลิตกรรมการเกษตร  
มหาวิทยาลัยแม่โจ้.
- พาวิน มะโนชัย และคณะ. (2546). รายงานการวิจัยเรื่อง การแก้ปัญหาการให้ผลเว้นปี และการ  
ปรับปรุงเทคนิคการผลิตผลไม้นอกฤดูกาลในลันจี้ ลำไย และมะม่วง. สำนักงาน  
คณะกรรมการวิจัยแห่งชาติ
- เสรี ปานซาง. (2539). การรู้จำเสียงพูดคำไทยแบบไม่ขึ้นกับผู้พูดด้วยนิรวัลเน็ตเวอร์ค. วิทยานิพนธ์  
ปริญญาโทบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร  
ลาดกระบัง.
- สำนักสำรวจดินและวางแผนการใช้ที่ดิน. (2549). ข้อมูลการใช้ประโยชน์ที่ดินจังหวัดเชียงใหม่  
[ซีดี-รอม]. กรมพัฒนาที่ดิน กระทรวงเกษตรและสหกรณ์.
- \_\_\_\_\_ . (2549). ข้อมูลการใช้ประโยชน์ที่ดินจังหวัดลำพูน [ซีดี-  
รอม]. กรมพัฒนาที่ดิน กระทรวงเกษตรและสหกรณ์.

- Awal, M.A. and Ishak, W.I. (2007). A New Technology for Oil Palm LAI and Yield Prediction. In **Proceedings of the 2<sup>nd</sup> National Intelligent Systems and Information Technology Symposium**. Malaysia. 50-55
- Baimoung, S. and Oki, T. (2003). A Study on Estimating of Sweet Corn Production Using Crop Growth Simulation Model with Satellite Remote Sensing and Geographical Information System Techniques [On-line]. Available: [http://www.arcims.tmd.go.th/Research\\_files/AFITA1.pdf](http://www.arcims.tmd.go.th/Research_files/AFITA1.pdf)
- Boken, V.K. (2000). Forecasting Spring Wheat Yield Using Time Series Analysis A Case Study for the Canadian Prairies. **Agronomy Journal**. 92: 1047-1053.
- Chen, C. and McNairn, H. (2006). A neural network integrated approach for rice crop monitoring. **International Journal of Remote Sensing**. 27: 1367-1393.
- Chungsik, Y. and Joo-Mi, K. (2006). Tunneling performance prediction using an integrated GIS and neural network. **Computers and Geotechnics**. 34: 19-30
- Dasa, K. *et al.* (2008). Classification of Slovak white wines using artificial neural networks and discriminant techniques. **Food Chemistry**. 112: 1046-1052.
- Drummond, S.T. and Sudduth, K.A. (2000). Predictive Ability of Neural Networks for Site-Specific Yield Estimation [On-line]. Available: <http://mpac.missouri.edu/pubs/drummond.pdf>
- Fausett, L. (1994). **Fundamentals of Neural Networks: Architectures, Algorithms and Applications**. New Jersey: Prentice-Hall.
- Geography Dictionary. (2004). Neural Network [On-line]. Available: <http://www.answers.com/topic/neural-network>

- Hagan, M.T., Demuth, H.B. and Beale, M. (1995). **Neural Networks Design**. Boston: PWS.
- Haig, L.A.S. (2003). **Crop yield estimation: Integrating RS, GIS and Management factors a case study of Birkoor and Kortgiri Mandals-Nizamabad district, India**. M.S. thesis, International Institute for Geo-Information Science and Earth Observation, Netherlands.
- Hussain, W. and Ishak, W. (2004). Notes on Neural Network Learning and Training [On-line]. Available: <http://www.generation5.org/content/2004/NNTrLr.asp>
- Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. **Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence**. 2(12): 1137-1143.
- Kudova, P. *et al.* (2006). Categorical data clustering using statistical methods and neural networks. **In Proceedings of the 3<sup>rd</sup> spring young researcher's colloquium on database and information system**. Moscow, Russia: SYRCoDIS Russian foundation for basic research.
- Land Development Department. (1999). Din Thai [Computer software]. Bangkok: Land Development Department, Ministry of Agriculture and Cooperatives.
- Lawrence, S.C., Giles, L. and Tsoi, A.C. (1997). Lessons in Neural Network Training: Overfitting May be Harder than Expected. **In Proceedings of the Fourteenth National Conference on Artificial Intelligence** (July). Menlo Park, California. 540-545.

- Liu, J., Goering, C.E. and Tian, L. (2001). A Neural Network for Setting Target Corn Yields. **American Society of Agricultural Engineers (ASAE)**. 44(3): 705-713.
- Malik, R. *et al.* (1999). A Comparative Study of Artificial Neural Networks and Multiple Regression Analysis in Estimating Willingness to Pay for Urban Water Supply [On-line]. Available:<http://www.irbdirekt.de/daten/iconda/CIB8813.pdf>
- Melchiorre, C. *et al.* (2006). Artificial neural networks and cluster analysis in landslide susceptibility zonation. **Geomorphology**. 94: 379-400.
- Mohd, M.I., Ahmad, S. and Abdullah, A. (1994). Agriculture Applications of Remote Sensing: Paddy Yield Estimation from Lansat-5/Thematic Mapper Data [On-line]. Available: <http://www.gisdevelopment.net/aars/acrs/1994/ts/ts1003pf.htm>
- Monica, B. *et al.* (2007). Neural network models for land cover classification from satellite images. **Agricultura Technica**. 67: 414-421.
- Monisha, K. *et al.* (2005). Artificial neural networks for corn and soybean yield prediction. **Agricultural System**. 85: 1-18
- Punam, T. *et al.* (2007). Relative Performance of Artificial Neural Networks and Regression Models in Predicting Missing Water Quality Data [On-line]. Available:<http://www.liebertonline.com/doi/abs/10.1089/ees.2007.0045>
- Pichaya Boonprasom. (2003). **Yield Prediction of Tangerine Using Artificial Neural Network (ANN)**. Chiangmai: Chiangmai University.

- Rama Linga, R.K. *et al.* (2008). Multiscale feature and single neural network based face recognition. **Journal of Theoretical and Applied Information Technology**. 4(7): 571-576.
- Ramyaa. (2004). **Frost Prediction Using Artificial Neural Networks: A Classification Approach**. Master Degree, The Artificial Intelligence Center, University of Georgia, USA.
- Ratchaphum Jaikla *et al.* (2008). Rice Yield Prediction Using a Support Vector Regression Method. In **Proceedings of IEEE ECTI-CON 2008** (May). Krabi. Thailand. 29-32.
- Safa, B., Khalili, A., Teshnehlab, M. and Liaghat, A. (2004). Artificial neural networks application to predict wheat yield using climatic data. In **Proceedings of 20<sup>th</sup> International Conference on Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology** (January). Poster Session 1. Seattle, WA.
- Sebastial, P. and Aleksander, M. (2007). Artificial neural networks based internet hypertension prediction tool development and validation. **Applied Soft Computing**. 8: 734-739.
- Serele, C.Z. *et al.* (2000). Corn Yield Prediction with Artificial Neural Network trained using Airborne Remote Sensing and Topographic Data [On-line]. Available: <http://ieeexplore.ieee.org/iel5/6913/18663/00860527.pdf>
- Smith, M. (1993). **Neural networks for statistical modeling**. New York: Van Nostrand Reinhold.

- Tanprasert, C. and Koanantakool, T. (1996). Thai OCR: A Neural Network Application. In **Proceedings of 1996 IEEE TENCON** (November). Perth, Western Australia.1: 90-95.
- Turney, P.D. (1994). A Theory of Cross-Validation Error. **Journal of Experimental and Theoretical Artificial Intelligence**. 6: 361-391.
- Vladimr, Z. (2008). Neural Network [On-line]. Available: <http://www.britannica.com/EBchecked/topic/410549/neural-network>
- Wikipedia. (2009). Neural network software [On-line]. Available: [http://en.wikipedia.org/wiki/Neural\\_network\\_software](http://en.wikipedia.org/wiki/Neural_network_software)
- Yu, S.I. *et al.* (2007). Prediction of Crop Yields with the Use of Neural Networks. **Russian Agricultural Sciences**. 33: 361-363.



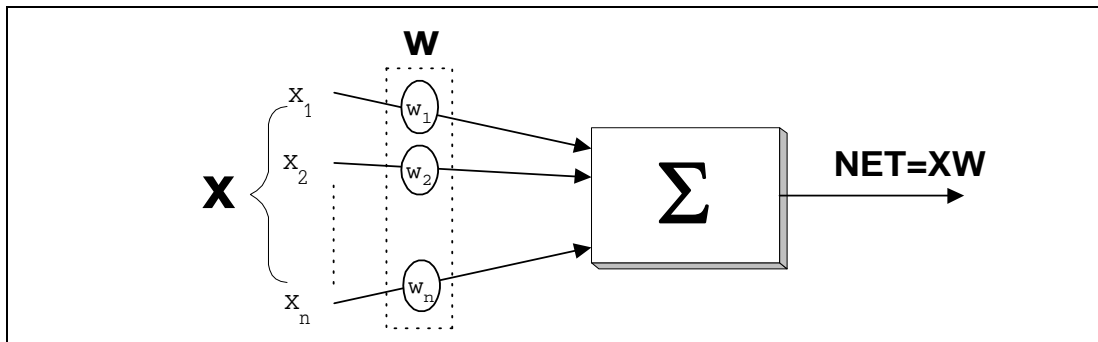
## **APPENDICES**

## **APPENDIX A**

### **ARTIFICIAL NEURAL NETWORK THEORY**

#### **A.1 Artificial neuron cell**

Artificial Neural Network was the emulation of the performance of biological-brain. The performance of neural-cell in biological brain was studied to specify the instruction to create the model and specify the performance hypothesis by making the model according to the similar mathematics model. Then it was calculated by computer program. It could be concluded according to the qualification of biological brain system that the input of receiving set of neural cell received output signal from other neural cell through axon, synapse and dendrite. The received data would be reduced the signal by neural transmission substance that was chemical substance namely;  $K^+$ ,  $Ca^+$ ,  $Na^+$ ,  $Cl^-$  that would have different chemical electricity signal transferring. So the artificial neural-net molecule must be weighted from the molecule before putting in artificial neural-net module. The input signal would be joined and decided by activation level of cell. Then output would exit at axon to other cell. The model of neuron cell could be shown as Figure A.1.



**Figure A.1** Artificial neuron cell

According to figure 1, the module made by biological brain cell concept. Input signal were  $X_1, X_2, \dots, X_n$  would be put into artificial neural-net cell that was created and was compared to the signal that was put into synapse of biological brain cell. This input signal would be multiplied by weight number  $W_1, W_2, \dots, W_n$  before it was done in  $\Sigma$  : Summation. This weight number would match with chemical substance at each single biological synaptic connection. Summation would work like biological brain cell. The mathematic total of input and weight would be output that was called NET that could be joined in Vector as following;

$$\mathbf{NET} = X_1W_1+X_2W_2+\dots+X_nW_n \quad (1)$$

$$\mathbf{NET} = \Sigma\mathbf{XW} \quad (2)$$

## A.2 Activation Function

The next process of artificial neural-net after getting NET was deciding. Activation Function must be specified to be used as the deciding level of artificial neural-net so that signal of artificial neural-net could give output that connected to other artificial neural-net as OUT-net. The function received could be simple linear function and the initial function for simple model could be specified output in binary

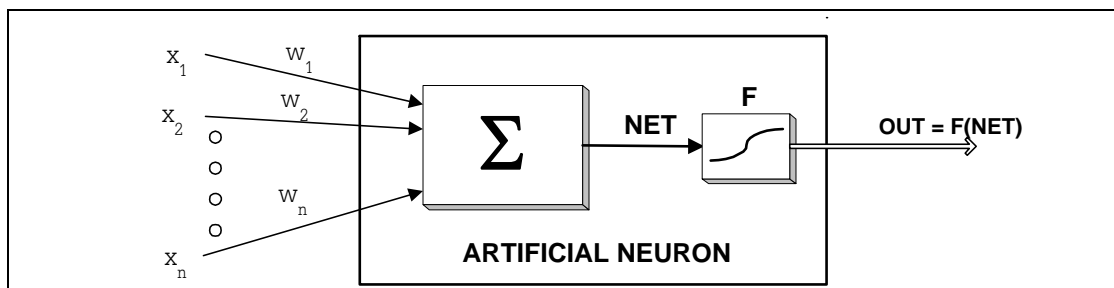
number as only 0 or 1 such as;

$$\text{OUT} = F(\text{NET}) \quad (3)$$

where  $F(\text{NET})$  was activation function specified as follows;

$$F(\text{NET}) = \begin{cases} 1 & \text{if } \text{NET} > T \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

$T$  was instant threshold number or could be any other types of function that could emulate non-linear function of biological neural-net cell and  $T$  could be closely used as the function for general network. This number was used to check the result of calculation after giving input to compare if the number was close to the fixed number.

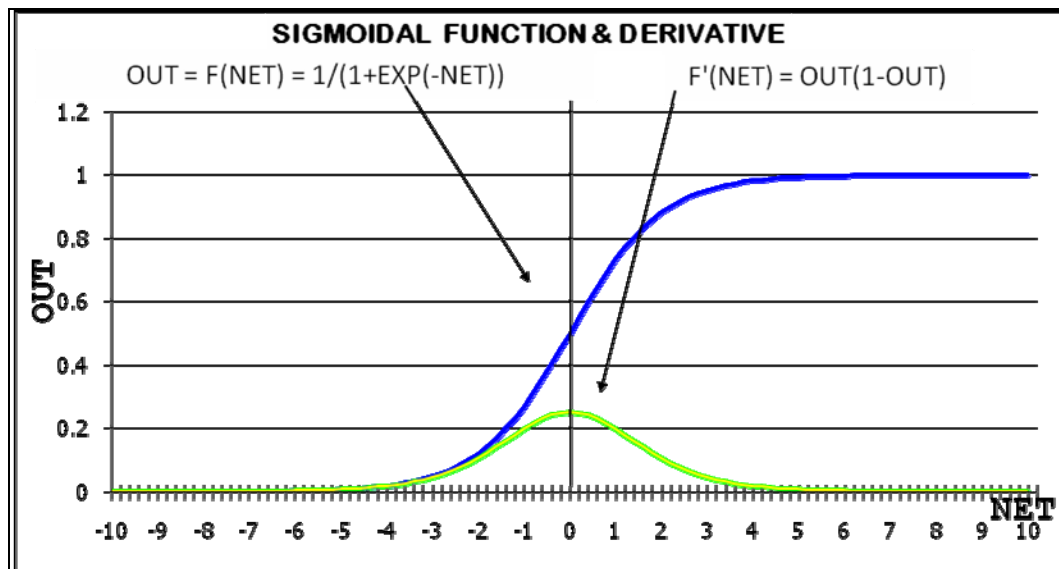


**Figure A.2** Artificial neuron cell made up along with activation function

The activation function specified by algebraic equation was applied to be used with artificial neural-net shown as Figure A.2. Block  $F$  would get the result for  $\text{NET}$  to make output at  $\text{OUT}$ . The process in block  $F$  would press the width of  $\text{OUT}$  to be in the limited area under the result of activation function so  $\text{OUT}$  number was not lower than the specified period number. The  $\text{NET}$  number specified in block  $F$  would be called “Logistic function” or “Sigmoid” that had the “S” shaped and could

be written as algebraic equation as the following;

$$F(\text{NET}) = \frac{1}{1 + \text{EXP}(-\text{NET})} \quad (5)$$



**Figure A.3** The graph from Sigmoid Logistic Function and Derivative

**Source:** Seri Pansang (เสรี ปานซาง, 2539)

Threshold function was not a linear function namely S-Curve. The output that was sensitive to the tiny input signal slightly plus signal, the “OUT” would be close to “1” (activate) or if the signal was slightly minus, the “OUT” would be close to “0” (stop). While the strong plus signal would give the result close to “1” and the strong minus signal would give the result close to “0”.

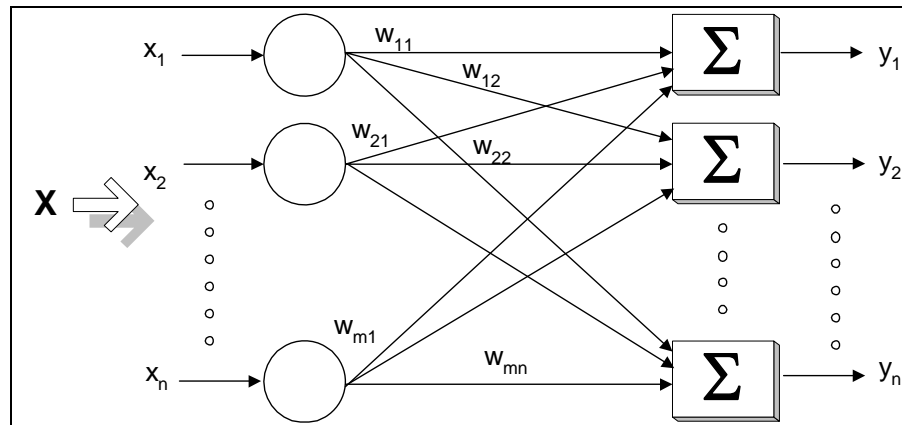
### A.3 ANN Architecture

As mentioned before, it was the rationale and background of one artificial neural network referring to the biochemical network cell process. The way to use artificial neural network cell needed the different qualification of artificial neural-net (weight number would change the qualification of artificial neural-net) to link as the

net similar to biological-brain cell. There were two types of linking as follows;

### A.3.1 Single Layer Artificial Neural Network

The ability to measure of artificial neural network was generally from group linked artificial model with plenty of layer but the single layer artificial neural network was composed of a lot of sets of single artificial neural that only linked among input layer and output layer, so the measurement ability of single artificial neural net was not much. The model of this network was shown as follows;



**Figure A.4** The Single Layer Artificial Neural Network Model

From Figure A.4, it was the single layer artificial neural network that composed of input layer and output layer. It wasn't considered to be the part of artificial neural since the input layer would link input get and gave output to each cell (output layer) in the next layer. Each input would be multiplied by weight number fixed for each input. The artificial neural network made in the first step was not complicated. Each artificial neural cell would give output as follows;

$$\text{OUT} = \text{Logistic Function} (\Sigma XW)$$

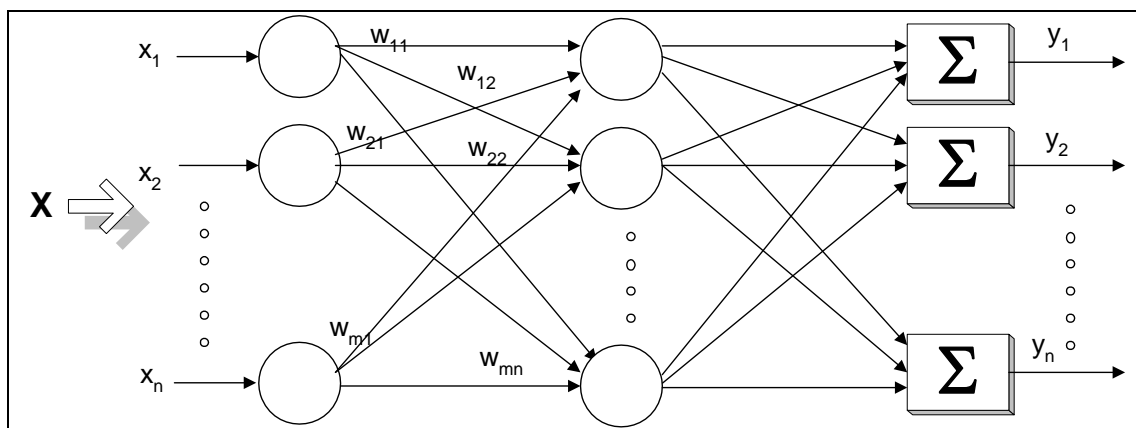
or

$$\text{OUT} = F(\text{NET}) \quad (6)$$

However, there was not only one type of network linked but there maybe were the return linked to the input layer. The biological neural network was as mentioned before. The weight number would be considered in the form of Weightt matrix that in case of several layers, the weight number would be easier specified. And to avoid the confusion, the Dimensions of matrix would be specified by using m as the row number or input number and n as artificial neural net made up. The example of linked weightt between 4<sup>th</sup> input and 2<sup>nd</sup> artificial neural network could be written as  $W_{42}$  (เสวี ปานชาง, 2539)

### A.3.2 Multilayer Artificial Neural Network

The complicated neural network would be able to do better measurement. The structural of network could be possible by linking artificial neural network cell to be several layers similar to part of brain. The multilayer network could be made of group of single layer network using layer 1 output to be the next layer input as Figure A.5



**Figure A.5** The multilayer artificial neural network

Figure A.5 showed the multilayer artificial neural network that linked full layer. The multilayer network was called as follows; the layer linked directly to input was called “input layer” that there wasn’t any measurement but the duty of this layer was to link the information to the next layer, the end right layer was called “output layer” that was the network which gave the result, the layer located between the input layer and output layer was called “Hidden layer” that if there were several layers in this layer, there would be the fixed name of each layer (เสวี ปานชาง, 2539)

#### **A.4 Learning of Artificial Neural Network**

According to the idea of cognitive psychology related to the new knowledge and remembrance in human brain that relied on linking the background knowledge and the new knowledge. Human could make the relation of human learning to the learning of artificial neural network. There was not any learning in baby brain, there was just the natural knowledge inborn. Baby would be developed step by step such as responding to the atmosphere when the brain learned and grew up. Brain cells would be developed as the learning and would grow up to be the network.

Neural Network made up was similarity in weight number that the number of knowledge. When starting, there was not any character of each brain cell since the suitable weight number was not specified for the work with cells or input in each network. The weight number in each network could be random weight or was as the mathematic questions that matched with output when the specified work was fixed.

#### **A.5 Objectives of Training**

Because weight number was random weight, the next work could not show any character. Training for network was adapting all weight number to match variety of



input in order to get output. Network training was like teaching human; sometimes human could not understand only onetime reading but there were giving examples and reviewing many times and learning would be better. In the similar way, network training would refer the error number from learning for adapting weight number in next process. After the weight number was adapted until get the least error number that matched the goal. The network structure would be ready for input analysis and give output as the way it learned for training. There would be many time for adapting weight number until the weight number was similar to a lot of examples and give output as the objective set. According to the research review, when the network got a lot of training examples, the result would be very accurately but it would take longer time for training. The artificial network made up had the behaviour similar to human learning. It was because the model was biological brain.

In conclusion, the effectiveness of learning of network would depend on weight number of network that the training was to find the suitable weight number for network.

#### 1) Supervised Learning

This training would specify teaching set for network that composed of the pair of training; input and set goal, and was be called “Learning pair”. The network would be taught along the pairs of training (the number of input pairs and the amount set that needed for network to know). The output from the training measurement would be compared to set objective. Then, the error number from the different of input and objective would be put back again in the network and would be adapted weight number to match the algorithm and decrease the trend of error number until

the error number is least to be able to accept, then the training stopped and the final weight number would be used as input function for other information.

## 2) Unsupervised Learning

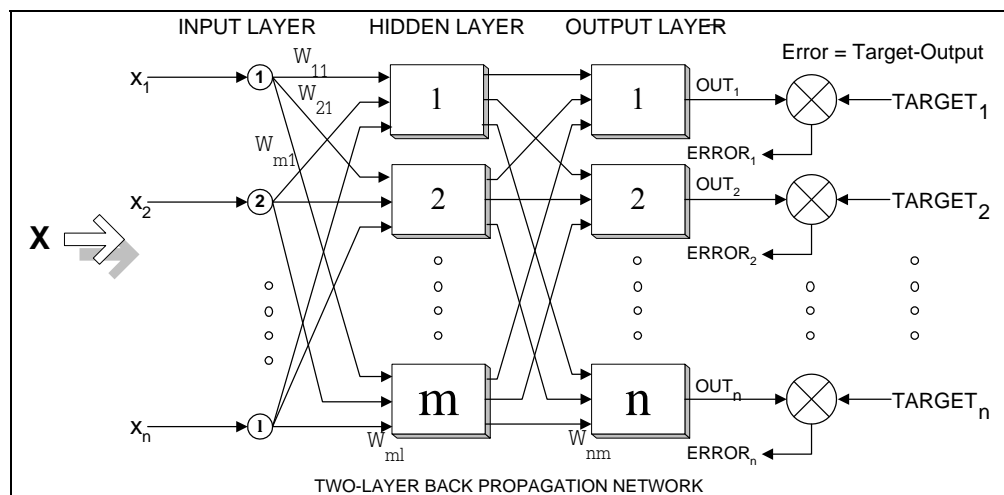
Although supervised learning could be used to adapt network characteristic, there were the critics that learning could not be as biological type and difficult to be believed that the learning process of the brain would need to compare the objective number to the real output using returning process to put the number to solve the network characteristic. If the brain had its own process, there must always be the person who needed to find the objective information. In real situation, no one could do it. In summarizing, there must be the person who create the objective for the network, the network itself could not create and adapt the characteristic. In the opposite way, even the baby's brain could manage the process itself, unsupervised learning still was impossible to be similar to human brain until 1984, the supervised learning was developed by Kohonen and other persons. They showed the learning without set objective and idealistic reason for considering. The set of learning was just input vector and input node inside network instead of similar qualification information. Algorithm of learning would change weight number of network when input was put in network, calculating relationship inside the input set that relied on weight number to divide the different input and kept inside output node of network. For example, if the network recognized the human face by using a lot of images even the face images were slightly changed, but the network could still specify that person. There wasn't prior deciding; no specified output (output would be specified by input vector) so, output of network was in the similar way, it was adapted to be able to understand after the learning process. So, it could not solve the specified or specific

problem, but this network was usually used in the simple work like comparing the unique of the relation of input and output that was fixed by network such as Clustering.

## A.6 Supervised Learning Process

Supervised Learning Process was the process to adapt the characteristic of neural network to be as the set qualification. This was controlled learning that composed of 2 parts; forward pass and reverse pass. Learning would happen in mixing way in both part until the error weight between input and set number was less than the set point as

Figure A.6;



**Figure A.6** The connection in the network of two-layer back propagation.

### 1) Forward Pass

This network would compose of some layers; the first layer was input and the last layer was output. Between input and output layer, there could be the layer inside depending on the algorithm that was used in network learning such as; if the network was multi-layer, there would be the layer between input and output that was called hidden layer that there could be more than one layer.

According to Figure 6, input  $x$  starting from  $x_1$  to  $x_i$  would be multiplied by weight number matrix. The total number of input after multiplied by weight number of each artificial network cell was called “NET” and could be write as the equation as following;

$$NET = X_1W_{j1} + X_2W_{j2} + \dots + X_iW_{ji} \quad (7)$$

or

$$NET_j = \sum_{i=1}^I X_i W_{ji} \quad (8)$$

where  $X_i$  was input number if node  $i$ ,  $i= 1, \dots, I$

$W_{ji}$  was weight number of input node  $i$  that sent to node  $j$  in next layer

After getting NET number for each artificial neural cell, the output would be adapted in the suitable period that would stop or activate next artificial cell through sigmoid function as the following equation (9)

$$Op_j = F_j(NET_j) = \frac{1}{1 + EXP(-NET)} \quad (9)$$

In the similar way, if there was the hidden layer between input and output, output number would be from the hidden layer and could be calculated to find  $NET_k$  that was the neural brain cell  $k$  output in output layer by  $k = 1, \dots, N$  as the equation (10) and (11)

$$NET_k = \sum_{j=1}^M Op_j W_{kj} \quad (10)$$

$$Op_k = F_k(NET_k) = \frac{1}{1 + \exp(-net)} \quad (11)$$

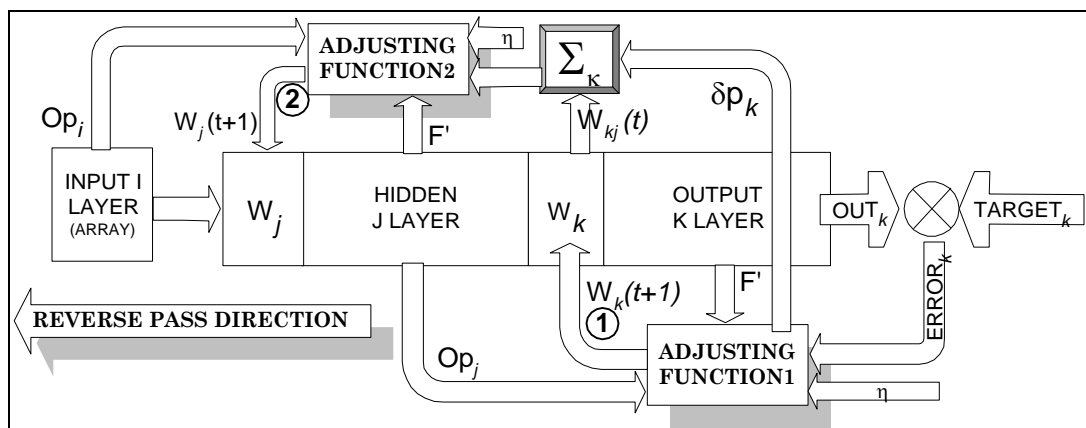
where  $Op_j$  was output that neural cell j of hidden layer (it was input of output layer)

$Op_k$  was output that neural cell k of output layer (it was output of output layer)

The calculation to find output would start at hidden layer j. When there were all outputs of hidden layer, the output of the output layer would be calculated. The input of the output layer was the output of the hidden layer. The output would be compared to the set number for error number. The error number would be used with differential number of F (that could be written using F') in output layer to adapt weight number for output layer as the next mention in reverse pass.

2) Reverse Pass

Reverse pass was the reverse of error number (set number-output) from output layer to be adapted weight number at prior layer. In case of there were more than one layer, the weight number adaptation of the next layer would use the same weight number of the higher layer along with the co-output number of the layer because there was not using error number that was different of the output layer. This information was shown as this figure;



**Figure A.7** The weight number adapted structure in output and hidden layer.

**Source:** Seri Pansang (เสรี ปานซาง, 2539)

According to Figure A.7, the changing of new weight would be along the arrows and the error numbers from the output layer would be used to start for specifying the direction of weight adjusting to make the artificial neural network to be as the set qualification. The steps of the reverse pass structure were concluded as follows:

1) Bring output from output layer to be minus by set number and finally get the error number.

2) Bring error number to multiply by  $F'$  from the output layer that was called  $\delta p_k$

3) Bring  $\delta p_k$  multiply by  $Op_j$  then multiply by  $\eta$  (Learning Rate 0.01-1)

4) The new weight number was the old weight number plus the result in step 3

The equation could be new written as:

$$W_{kj}(t+1) = W_{kj}(t) + \eta \delta p_k Op_j \quad (12)$$

$$\eta \delta p_k Op_j = \Delta W_{kj} \quad (13)$$

$$W_{kj}(t+1) = W_{kj}(t) + \Delta W_{kj} \quad (14)$$

where  $W_{kj}(t)$  was weight number before adjusting

$W_{kj}(t+1)$  was weight number after adjusting

$\eta$  was Learning Rate

$Op_j$  was output of j layer

$\delta p_k$  was the number getting from multiplying error number by

differential of function F

Adjusting the next layer (hidden layer) was as the similar way as output layer:

$$W_{ji}(t+1) = W_{ji}(t) + \eta \delta p_j O_{pi} \quad (15)$$

$$\eta \delta p_j O_{pi} = \Delta W_{ji} \quad (16)$$

$$W_{ji}(t+1) = W_{ji}(t) + \Delta W_{ji} \quad (17)$$

The equation 16 was different to the equation 13 as the way of getting  $\delta p_j$  because there was no target in hidden layer. The mentioned value could be used  $\sum_k (\delta p_k W_{kj})$  instead

$$\delta p_j = O_{p_j}(1 - O_{p_j})(\sum_k (\delta p_k W_{kj})) \quad (18)$$

Where  $W_{ki}(t)$  was weight number before adjusting

$W_{ki}(t+1)$  was weight number after adjusting

$\eta$  was Learning Rate

$O_{pi}$  was the output information from the input layer

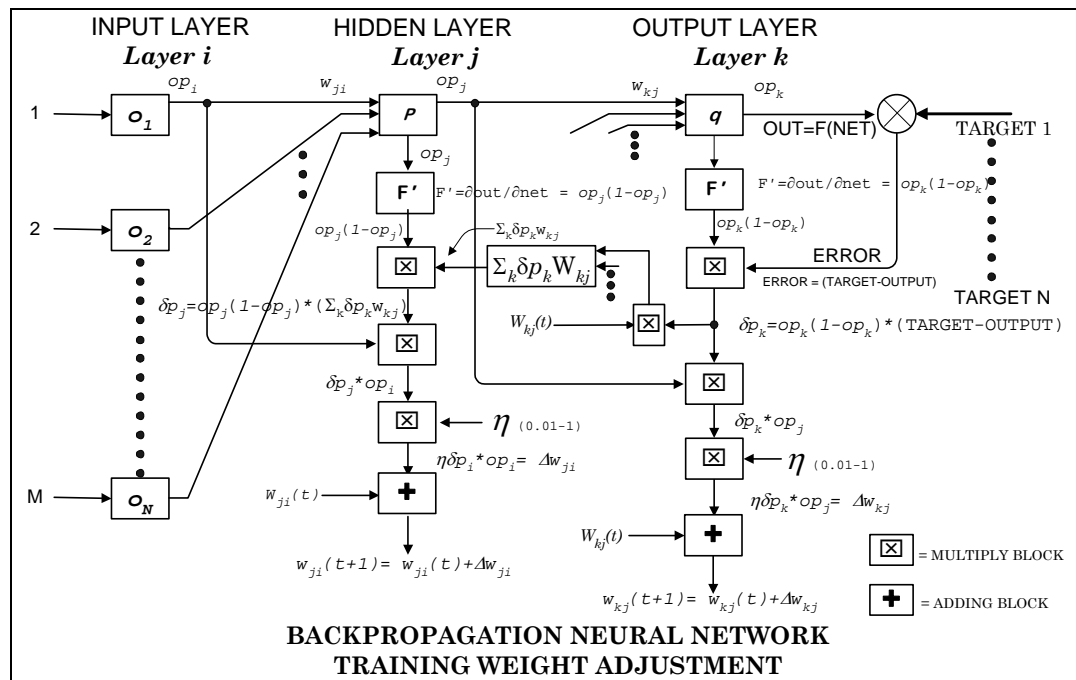
$\delta p_j$  was the value to control weight convergence to the best point

Adjusting weight number would be done layer by layer from the output layer reverse to the input layer. It was the part of reverse pass network process.

## A.7 Backpropagation

It was the way of supervised learning that was the step to teach multi-layer artificial neural network or it was called the step of Error Backpropagation that was the backpropagation of the error number from the output layer using the fixed input. The calculation was done by reversing from output layer through hidden layers and finally input layer to adjust weight number.

The adjusting learning to adjust weight of backpropagation neural network could be described as this diagram:



**Figure A.8** The training weight adjustment using back propagation algorithm

**Source:** Seri Pansang (เสรี ปานซาง, 2539)

According to figure A.8, the input would pass through artificial neural network then it would be learned to get the closest to the set goal output. The error number from the training of one circle would be put back to the network by reversing to each layer. After the process finished, the number would be put through the forward pass network to calculate output and compared to the set objective number and the adapting weight number would happen in reverse network until the error number was reduced and less than set number then the process stopped. The weight number would be suitable for all pairs of input and set number and was ready for testing process.



# **APPENDIX B**

## **USER MANUAL**

### **B.1 Introduction**

The GISANN is designed for agricultural yield prediction. Development in extension of ArcMap form by .NET technology and ANN theory. The software runs within the scope of ArcMap environment.

### **B.2 Copyright Notice**

Copyright © 2009 by Suranaree University of Technology

### **B.3 System Requirement**

Operating System : Windows 95/98/NT

.NET Framework : 1.0+

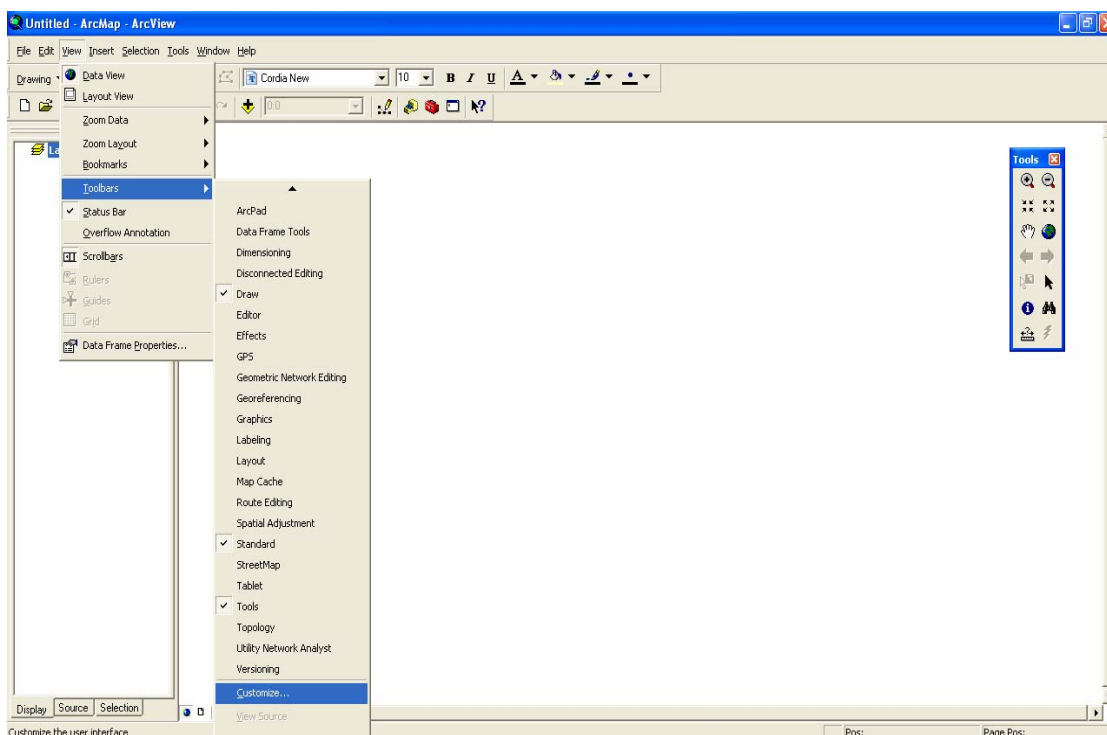
### **B.4 Data Format**

Data which input must be normalize in 0-1 scale before feed into program.

### **B.5 Installation**

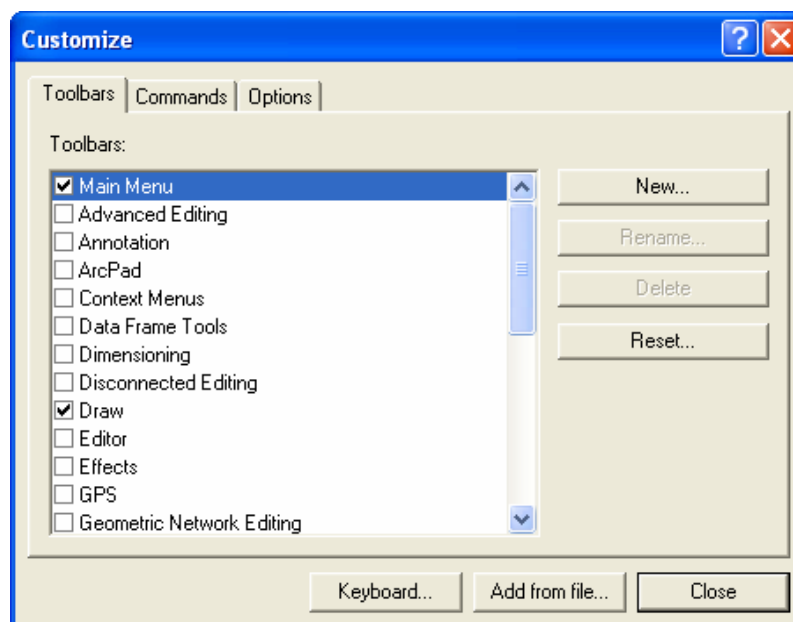
1. Choose the Customize from the Toolbar under View menu in ArcMap.

Figure B.1 show the step to open the customize window



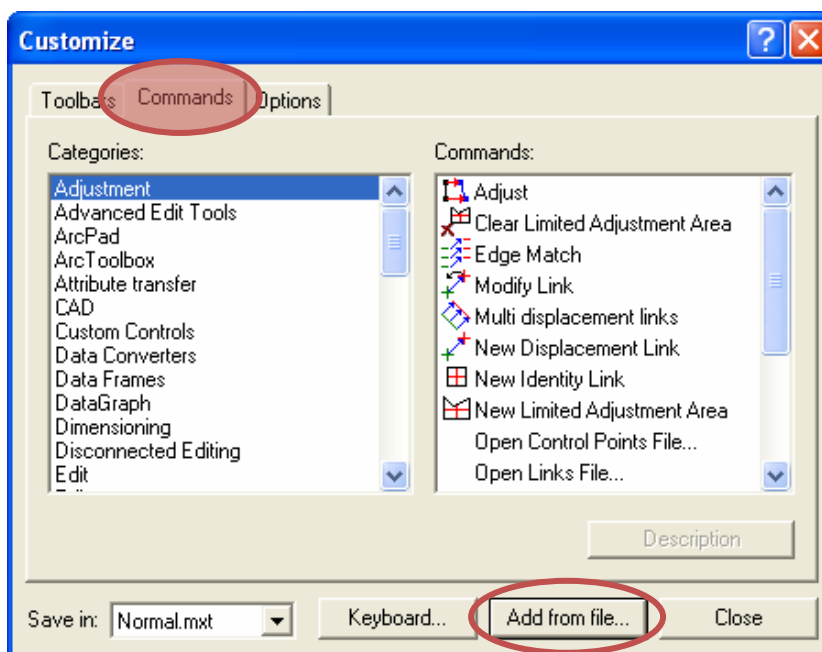
**Figure B.1** Show the step to open Customize dialog box.

The Customize dialog box would appear as show in Figure B.2.



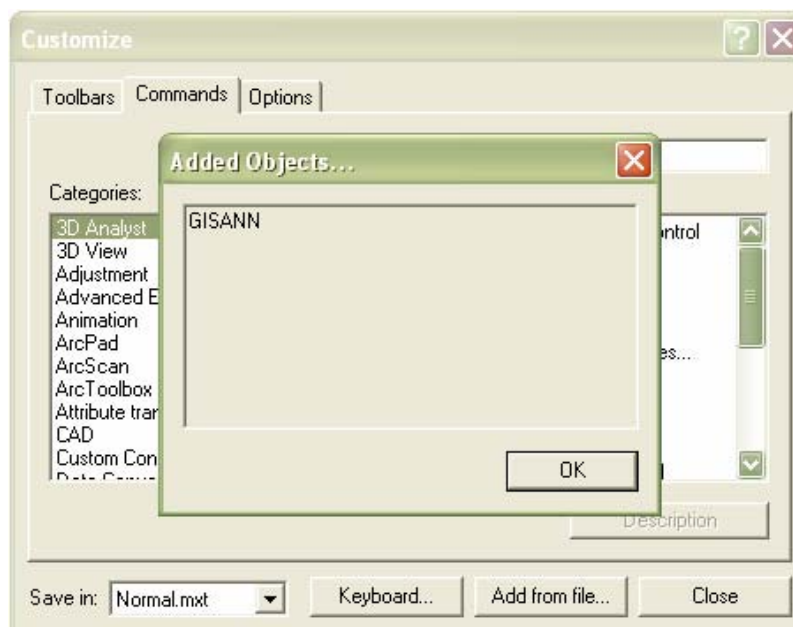
**Figure B.2** Show the Customize dialog box.

2. At the tap Command, click Add from file button.



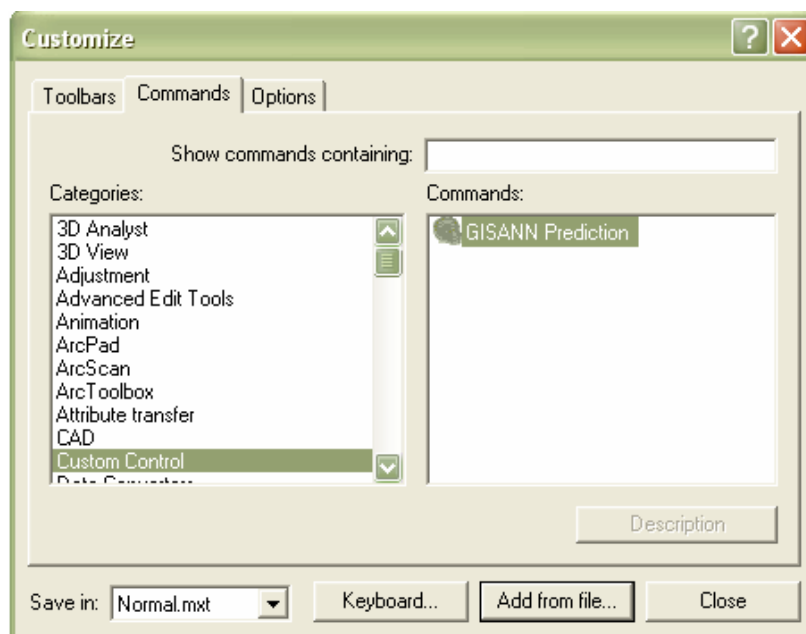
**Figure B.3** Show the Commands tap and Add from file button.

3. Choosing GISANN.tlb from the path that store the program



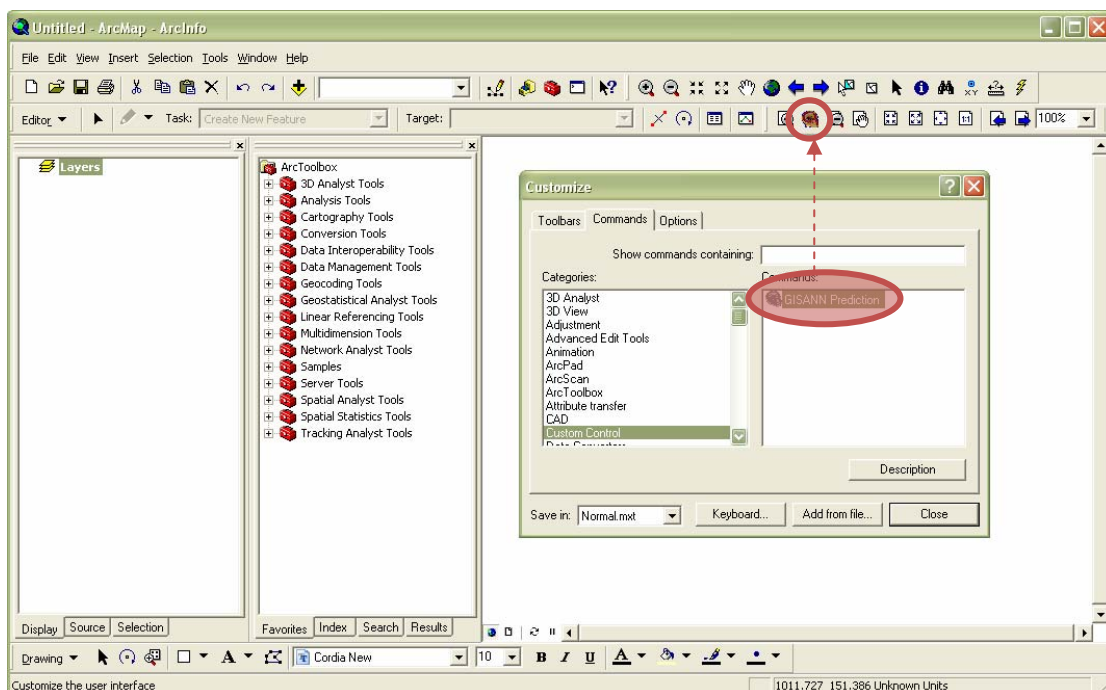
**Figure B.4** Show the Added objects window

The GISANN would appear in the Custom Control category.



**Figure B.5** The GISANN in Custom Control category

4. Drag the GISANN Prediction icon to anywhere on the toolbar



**Figure B.6** The GISANN icon on the toolbar

## B.6 Workflow Overview

The steps below serve as a general guide for the sequence

### 1. Training

The first step is Training, is the step of make relation between influence factors and yield. After the training step, the weight set would occur.

### 2. Testing

Using Testing tab to determine the weight set if appropriateness or not.

### 3. Predict

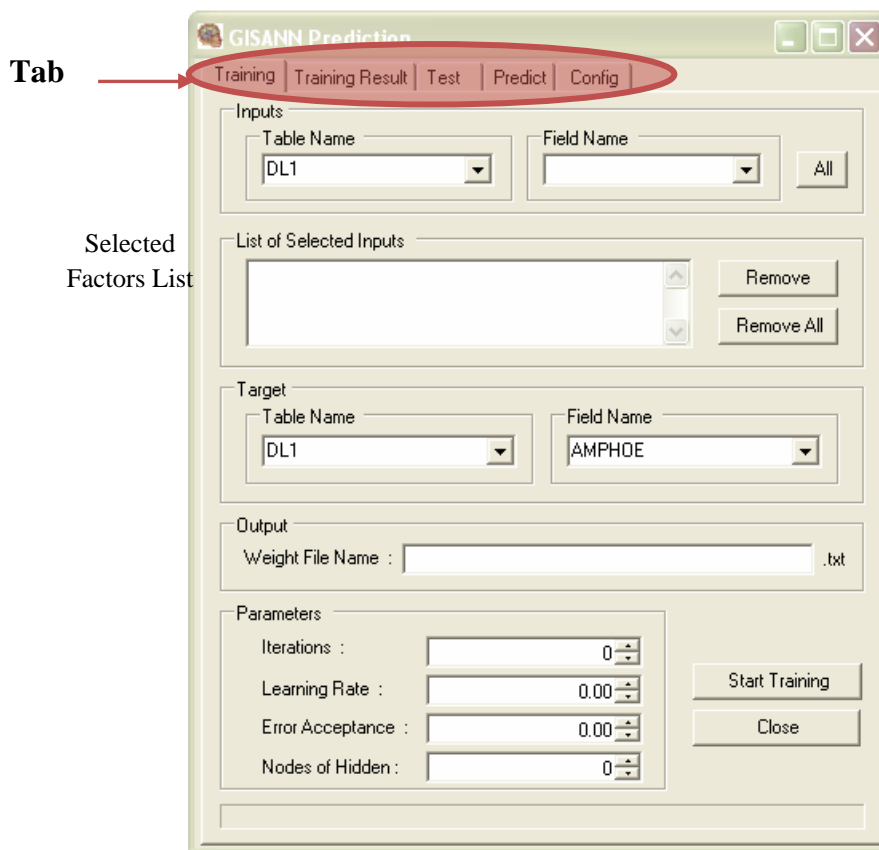
To predict yield, the optimal weight set is using to calculate the output by feeding input that relative with the input of training step.

## B.7 Instructions for using the system

### 1. Click GISANN icon



The GISANN window would appear



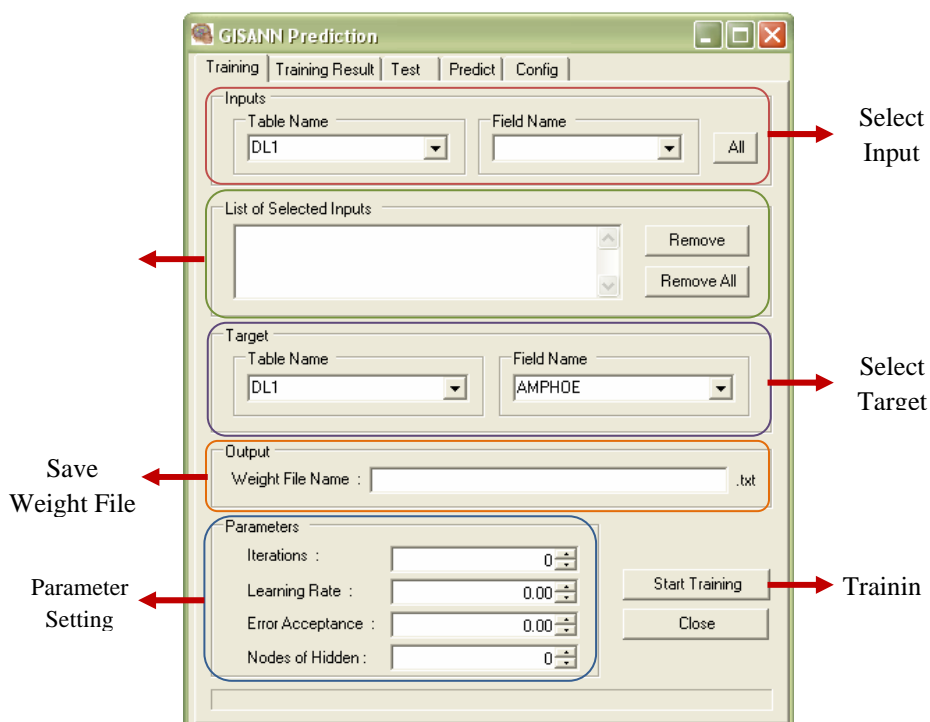
**Figure B.7** The GISANN window

The GISANN compose of 5 tabs are:

1. Training: Tab is the tap for training GISANN. Feeding necessary input and target into the network, then the network training itself and create the weight set for testing and predicting. The user interface of Training tab show in Figure B.8.

The steps to training are:

- 1) Select Input table and input fields: the input table must be .dbf file and field name at drop down list show the correspondence field in the table. The user can click “All” button to select all field in the input table. All selected fields would appear in the “List of Selected Inputs”. The user can remove some field or all fields from the list.



**Figure B.8** Show the user interface of Training tab

2) Define Target field: same as input table, the user must select the table and target field. The target field is use for training as goal of training output. For yield prediction, the target field is the yield of all input factors.

3) Typing Weight File name

4) Setting all parameter

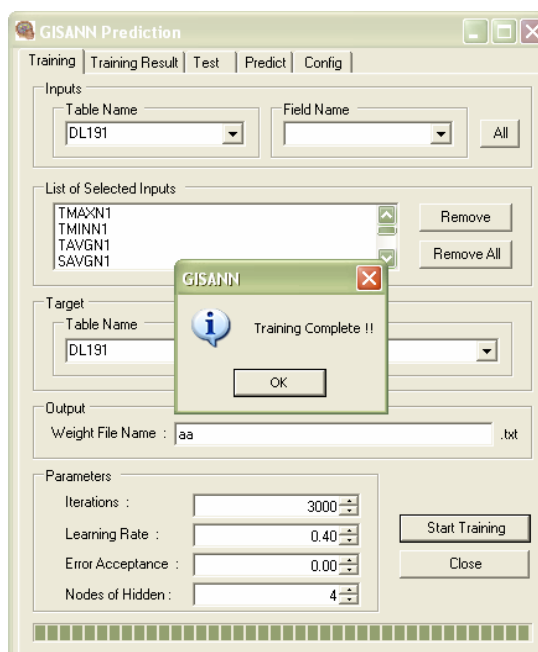
- Iterations: is the number of training round.

- Learning Rate: is the Rate of Learning, the value is between 0 to 1.

- Error Acceptance: is percentage of error that user determine to stop training. The value is between 0 to 1. If Error Acceptance set, the Iterations would set to 0, on the other hand, the Error Acceptance would be 0 if Iterations set to any value.

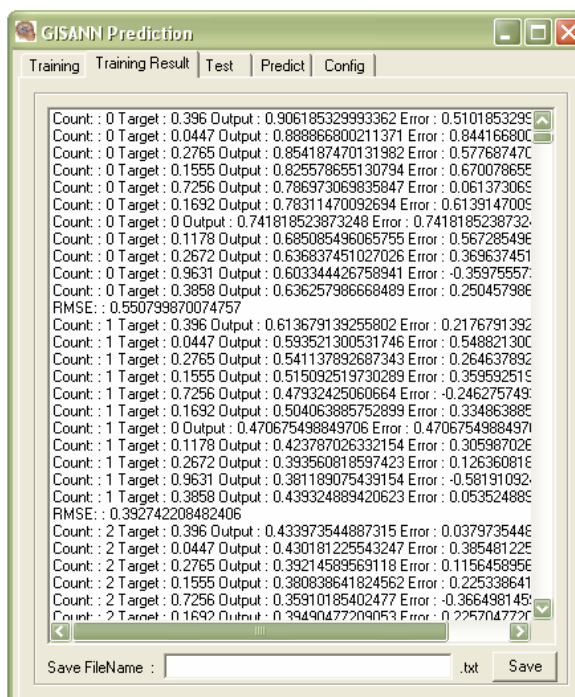
- Nodes of Hidden: is the number of hidden node.

5) Click Start Training



**Figure B.9** Show the message box after Training

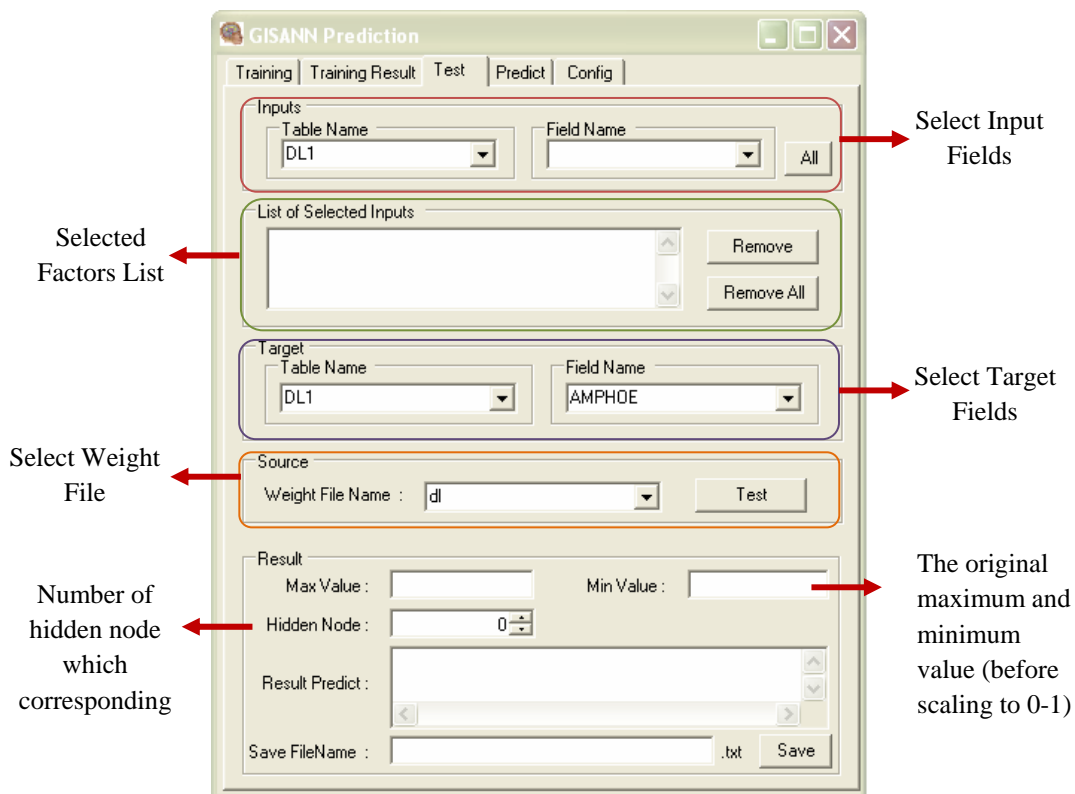
2. Training Result: Tab shows the result of training. The user can save the result for checking RMSE value.



**Figure B.10** Show Training Result window.



3. Testing Tab: It use for checking if the weight set is good enough for prediction. The user interface of Training tab show in Figure B.11.

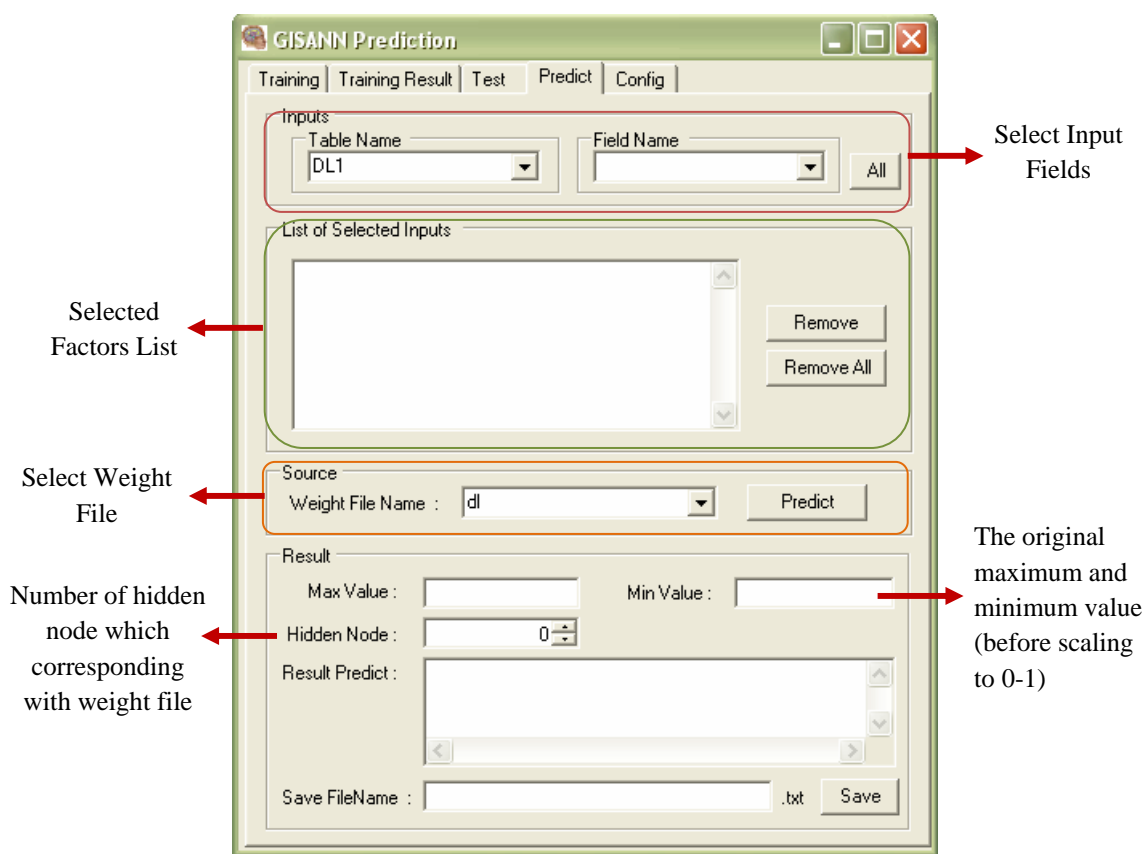


**Figure B.11** Show the user interface of Testing tab.

The steps to testing are:

- 1) Select Input fields
- 2) Define Target field
- 3) Select Weight File
- 4) Enter Max Value and Min Value
- 5) Define Hidden Node
- 6) Click Test

#### 4. Predict Tab

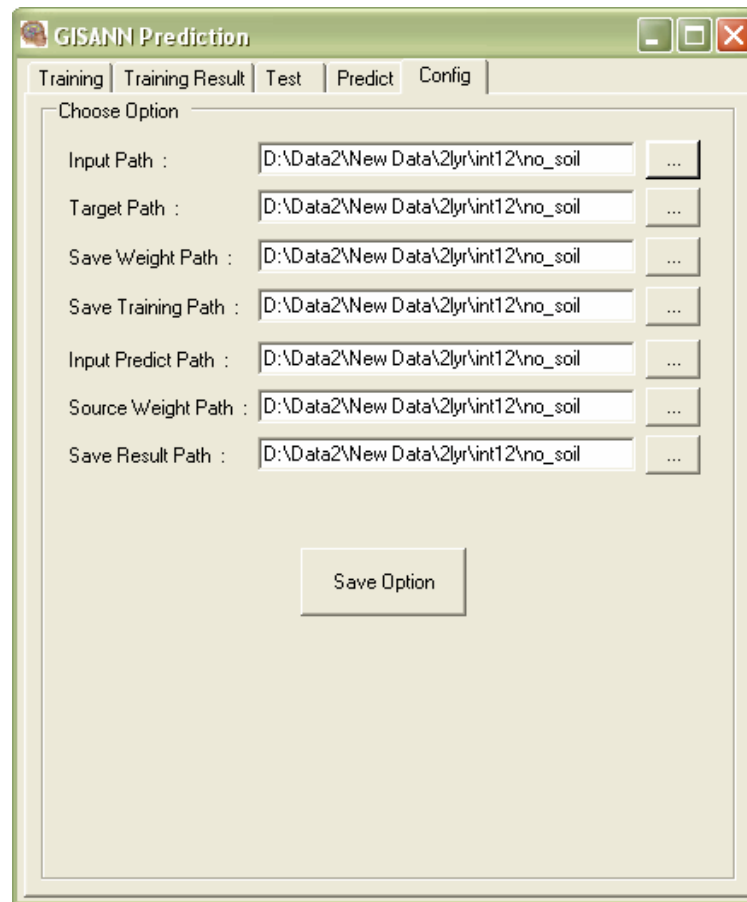


**Figure B.12** Show the user interface of Predict tab.

The prediction is not need to use target field. The steps to predict are :

- 1) Select Input fields
- 2) Weight File
- 3) Enter Max Value and Min Value
- 4) Define Hidden Node
- 5) Click Predict

5. Config Tab: It use for setting paths. The user can set and save paths for training, testing and prediction.



**Figure B.13** Show the user interface of Config tab.

## B.9 Error message and causes

Error Message	Cause
Please Select Data	Try to remove all fields from “List of Selected Field” while the list is empty.
Target Field Name is Invalid Number	Target filed must be the number data. Check field name to ensure that the target field is not the text field.
Please Enter Weight File Name	Try to training, testing or prediction without weight file.
Please Select INPUT Field(s)	Try to training, testing or prediction without input fields.
Attempted to divide by zero	The user do not input iteration number, go to define iterations number.
Input Save Filename	The user do not input filename before saving, go to enter file name before save.
Conversion from string to type Double is not valid	The user do not input maximum and minimum value, go to enter maximum and minimum value.

## APPENDIX C

### EXAMPLE OF SOURCE CODE

```
Public Overridable Sub FeedForward(ByVal sampleNumber As Integer)
    Dim total As Double
    Dim ch As Neuron = Nothing
    Dim ch2 As Neuron = Nothing
    Dim ci As Neuron = Nothing
    Dim co As Neuron = Nothing

    'feed the input data to the input Neurons Layer
    Dim i As Integer
    For i = 0 To input.Count - 1
        ci = input(i)
        ci.Input = Me.input_data(sampleNumber, i)
    Next i

    'feedforward from input to 1st Hidden Layer
    Dim h As Integer
    For h = 0 To hidden.Count - 1
        total = 0.0

        ch = hidden(h)

        For i = 0 To input.Count - 1
            ci = input(i)

            total = total + ci.Output * ch.Weight(i)
        Next i

        ch.Input = total + ch.Bias
    Next h

    'feedforward from 1st Hidden Layer to 2nd Hidden Layer

    Dim h2 As Integer
    For h2 = 0 To hidden.Count - 1
        total = 0.0

        ch2 = hidden2(h2)
```

```

    For h = 0 To hidden.Count - 1
        ch = hidden(h)
        total = total + ch.Output * ch2.Weight(h)
    Next h
    ch2.Input = total + ch2.Bias
Next h2

'feedforward from hidden to output Neurons
Dim o As Integer
For o = 0 To output.Count - 1
    total = 0.0

    co = output(o)

    'feed the expected training result to the output Layer
    co.Target = Me.output_data(sampleNumber, o)

    For h = 0 To hidden.Count - 1

        ch2 = hidden2(h)
        total = total + ch2.Output * co.Weight(h)
    Next h
    co.Input = total + co.Bias
Next o
End Sub 'FeedForward

```

```

Public Overridable Sub BackPropagate()
    Dim total As Double
    Dim ch As Neuron = Nothing
    Dim ch2 As Neuron = Nothing
    Dim ci As Neuron = Nothing
    Dim co As Neuron = Nothing

    'calculate error of Output layer (co.error)

    Dim o As Integer
    For o = 0 To output.Count - 1
        co = output(o)
        co.Error = co.Target - co.Output
    Next o

    'error of 2nd hidden layer (ch2.error)

    Dim h As Integer
    For h = 0 To hidden.Count - 1

        total = 0.0

```

```

ch2 = hidden2(h)

For o = 0 To output.Count - 1
    co = output(o)
    total = total + co.Error * co.Weight(h)
Next o
ch2.Error = total
Next h

'error of 1st hidden layer (ch.error)
Dim h2 As Integer
For h = 0 To hidden.Count - 1
    total = 0.0
    ch = hidden(h)

    For h2 = 0 To hidden.Count - 1
        ch2 = hidden2(h2)
        total = total + ch2.Error * ch2.Weight(h)
    Next h2
    ch.Error = total
Next h

'Update all weights in the network
'From Output Layer to 2nd Hidden Layer
'Dim o As Integer
For o = 0 To output.Count - 1
    co = output(o)

    For h = 0 To hidden.Count - 1
        ch2 = hidden2(h)
        co.Weight(h) += Me.learning_rate * co.Error *
co.LogisticFunctionDerivative(co.Output) * ch2.Output

    Next h

    co.Bias = co.Bias + (Me.learning_rate * co.Error) *
co.LogisticFunctionDerivative(co.Output)

Next o

'Update all weights in the network
'From 2nd Hidden Layer to 1st Hidden Layer

For h2 = 0 To hidden.Count - 1
    ch2 = hidden2(h2)

    For h = 0 To hidden.Count - 1

```

```

        ch = hidden(h)

        ch2.Weight(h) += Me.learning_rate * ch2.Error * (ch2.Output * (1 -
ch2.Output)) * ch.Output

    Next h

    ch2.Bias = ch2.Bias + (Me.learning_rate * ch2.Error) * (ch2.Output * (1 -
ch2.Output))

Next h2

'Update all weights in the network
'From 1st Hidden Layer to Input Layer
'Dim h As Integer

For h = 0 To hidden.Count - 1
    ch = hidden(h)

    Dim i As Integer
    For i = 0 To input.Count - 1
        ci = input(i)
        ch.Weight(i) += Me.learning_rate * ch.Error * (ch.Output * (1 - ch.Output))
* ci.Output

    Next i
    ch.Bias = ch.Bias + (Me.learning_rate * ch.Error) * (ch.Output * (1 -
ch.Output))

Next h

End Sub 'BackPropagate

Public Overridable Sub Train(ByVal mWeightFile As String)
    If TxtWeight.TextLength = 0 Then
        MsgBox("Please Enter Weight File Name", MsgBoxStyle.Critical)
        Exit Sub
    End If
    If ListBox1.Items.Count = 0 Then
        MsgBox("Please Select INPUT Field(s)", MsgBoxStyle.Critical)
        Exit Sub
    End If

    Dim inputDataArray(.) As Double = Nothing

    Dim outputDataArray(.) As Double = Nothing

```



```
' Input & Target Setting
```

```
Dim i, j As Integer
Dim ColCount As Integer
Dim RecCount As Integer
```

```
Try
```

```
    Dim Cmd As OleDbCommand = New OleDbCommand
    Dim Sqlldr As OleDbDataReader
    Dim Sqlcmd As String
    With myConn
        If .State = ConnectionState.Open Then .Close()
        .ConnectionString = myStrConnTraininput
        .Open()
    End With
    Sqlcmd = "Select Count(*) as mQuan From " & Trim(ComboTableinput.Text)
    With Cmd
        .CommandType = CommandType.Text
        .CommandText = Sqlcmd
        .Connection = myConn
        Sqlldr = .ExecuteReader()
    End With
    Sqlldr.Read()
    RecCount = Sqlldr.Item("mQuan")
    ColCount = ListBox1.Items.Count
    Sqlldr.Close()
    Sqlcmd = "Select "
    For i = 0 To ListBox1.Items.Count - 1
        If i <> ListBox1.Items.Count - 1 Then
            Sqlcmd &= " " & ListBox1.Items(i) & ","
        Else
            Sqlcmd &= " " & ListBox1.Items(i) & " "
        End If
    Next
    Sqlcmd &= " From " & Trim(ComboTableinput.Text)
    With Cmd
        .CommandType = CommandType.Text
        .CommandText = Sqlcmd
        .Connection = myConn
        Sqlldr = .ExecuteReader()
    End With
```

```
ReDim Preserve inputDataArray(RecCount - 1, ColCount - 1)
ReDim Preserve outputDataArray(RecCount - 1, 0)
```

```
i = 0
While Sqlldr.Read()
    For j = 0 To ColCount - 1
```

```

        Dim xx As String = Sqlldr.Item(j)
        inputDataArray(i, j) = xx
    Next
    i = i + 1
End While

Sqlldr.Close()

Sqlcmd = "Select " & ComboTargetfield.Text & " From " &
Trim(ComboTargettable.Text)
With Cmd
    .CommandType = CommandType.Text
    .CommandText = Sqlcmd
    .Connection = myConn
    Sqlldr = .ExecuteReader()
End With
Dim ix As Integer = 0
While Sqlldr.Read()
    outputDataArray(ix, 0) = Sqlldr.Item(0)
    ix = ix + 1
End While

Dim recordCount As Integer = RecCount

If recordCount <> outputDataArray.GetUpperBound(0) + 1 Then
    MsgBox("number of samples in input data is not equal to number of
samples in output data", MsgBoxStyle.Information)
End If

Initialize(inputDataArray, outputDataArray, txtHiddenLayer.Value)

Dim count As Integer
Dim err As Double = 0
Dim predict_err As Double = 0

ListBox3.Items.Clear()

Dim errsqr As Decimal
Dim RMSE As Decimal
Dim SSE As Decimal

If TxtIterations.Value > 0 Then
    ProgressBarX.Maximum = iterations

    For count = 0 To iterations - 1

        Dim sample As Integer

```

```

For sample = 0 To recordCount - 1

    FeedForward(sample)
    If count <> iterations - 1 Then
        BackPropagate()
    End If

    err = output(0).Output - output(0).Target

    ListBox3.Items.Add(("Count: : " + count.ToString() + " Target : " +
output(0).Target.ToString() + " Output : " + output(0).Output.ToString() + " Error : "
+ err.ToString()))
    errsqr = System.Math.Pow(err, 2)
    SSE = SSE + errsqr

    Next sample
    ProgressBarX.Value = count + 1
    SSE = SSE / recordCount
    RMSE = System.Math.Sqrt(SSE)
    ListBox3.Items.Add("RMSE: : " + SSE.ToString())

Next count

Else 'using RMSE as stoping criteria
    ProgressBarX.Maximum = Integer.MaxValue

    count = 0
    SSE = 1
    Do While ((SSE > txtError.Value) And (count < 5000))
        Dim sample As Integer
        count = count + 1
        For sample = 0 To recordCount - 1

            FeedForward(sample)
            BackPropagate()

            err = output(0).Output - output(0).Target

            ListBox3.Items.Add(("Count: : " + count.ToString() + " Target : " +
output(0).Target.ToString() + " Output : " + output(0).Output.ToString() + " Error : "
+ err.ToString()))

            errsqr = err * err
            SSE = SSE + errsqr
        Next sample
        SSE = SSE / (recordCount - 1)
        RMSE = System.Math.Sqrt(SSE)
        ListBox3.Items.Add("RMSE: : " + SSE.ToString())
    
```

```
    Loop

End If
SaveWeight(mWeightFile + "\")

MsgBox("Training Complete !!", MsgBoxStyle.Information)

Catch ex As Exception
    MsgBox("Cause : " + ex.ToString, MsgBoxStyle.Information)
End Try

End Sub
```

## **CURRICULUM VITAE**

Name : Kanchana Thonghoonnak (Ms.)

Date of Birth : September 13, 1970

Place of Birth : Nakhon Ratchasima, Thailand

Education Background :

1988-1992 B.Ed. Computer Education

Nakhon Ratchasima Teacher College, Nakhon Ratchasima,  
Thailand

1996-2000 M.Sc. Computer Science

Chiangmai University, Chiangmai, Thailand

Conference : ANN for Agricultural Yield Prediction. 2007 Annual National

Conference on Mapping and Geoinformatics, 18-21 October 2007,

Ambassder, Bangkok Thailand.

Work Experience :

1992 to date Instructor

Computer Department

Faculty of Science and Technology

Chiangmai Rajabhat University, Thailand

Office Address : Faculty of Science and Technology

Chiangmai Rajabhat University, Thailand, 50300