IMPROVING NEURAL NETWORKS TRAINING USING EXPERIMENT DESIGN APPROACH

CHONG WEI KEAN

A project report submitted in fulfillment of the requirements for the award of the degree of Master of Electrical Engineering (Mechatronics and Automatic Control)

Faculty of Electrical Engineering
Universiti Teknologi Malaysia

MAR 2005

DEDICATION

To my beloved father, mother, and Yih Leng

ACKNOWLEDGEMENT

I would like to express my deepest and sincere appreciation to my supervisor, Dr. Shahrum Shah Abdullah, for encouragement, guidance, critics, advices motivation and friendship. Without his continued support and interest, this report would not have been the same as presented here.

I would like to thank my fellow Dharma friend, Chia Yu Leng for his guidance and patience for teaching me the Matlab programming language.

My fellow postgraduate students should also be recognized for their support. My sincere appreciation also extends to all Southern College and my colleagues and others who have provided assistance at various occasions. Their view and tips are useful indeed. Unfortunately, it is not possible to list all of them in this limited space.

I am grateful to all my family members and Yih Leng, I could not make it without your full support and understanding.

ABSTRACT

This project involves the use of Neural Networks (NN) for function approximation. Conventionally, the parameters of a neural network are tuned by minimizing an objective function based on a pre-determined set of training data. This training paradigm is passive in the sense that the neural network only learns from the training patterns presented to it by the environment or a teacher. It may be more useful if the neural network could `actively' obtain the training samples itself sequentially by interacting with its environment. There are several methods of selecting training data from input space for neural networks which include D-optimal and Max-min design approaches. Consider a function approximation problem (Neural Network using Radial Basic Function structure) and limit the amount of training data, say *(m)* from *N* amount of possible data. Randomly select the *m* data set for conventional training algorithm. One more data (m+1) is entered to train the NN again. This data is selected by two methods: random and Experiment Design Approach (D-optimal and Maxmin Distance). The performances of the two approaches are then compared. It was found that the NN trained using the data obtained using experiment design approaches approximated the unknown function better than the NN that is trained when the data are selected randomly. Maxmin Distance Approach is independent of the NN model while D-optimal point is dependent on the NN model used.

ABSTRAK

Projek ini melibatkan penggunaan *Nueral Network* (NN) untuk tujuan penghampiran fungsi. Lazimnya, parameter yang digunakan dalam NN adalah dilaras berasaskan set data latihan yang ditetapkan pada tahap awal. Ini adalah pasif, NN hanya dapat belajar daripada corak latihan yang ditunjukkan oleh guru atau persekitarannya. Ini akan menjadi lebih berguna jika NN dapat memperoleh data latihan secara "aktif" melalui interaksi dengan persekitaran. Terdapat pelbagai cara untuk memilih data, ini termasuklah *D-optima* dan *Pendekatan Jarak Maxmin*. Pertama, tentukan satu masalah penghampiran fungsi (NN menggunakan *Radial Basis Function*) dan hadkan bilangan data, katakan (*m*) daripada *N*. Sebanyak *m* data dipilih secara rawak untuk latihan NN. Satu lagi data (*m*+1) dimasukkan dan menjalani latihan NN sekali lagi. Data ini boleh dipilih melalui dua cara iaitu secara rawak dan *Experiment Design (D-optima dan Pendekatan Jarak Maxmin)*. Bandingkan pencapaian mereka. Keputusannya, NN yang menggunakan *Experiment Design* adalah lebih baik daripada NN yang memilih data secara rawak. Diperhatikan bahawa, *Pendekatan Jarak Maxmin* tidak bergantung kepada model NN yang digunakan, tetapi titik yang dicadangkan oleh *D-optima* amat bergantung kepada model NN yang digunakan.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

x	-	Input
Error! Objects cannot be created from editing field codes.	-	Output
$\tilde{y}$	-	Predicted output
X	-	Environment
$E(\ )$	-	Expectation
η	-	Linear model with unknown parameter
N	-	Total number of observation to be taken or sampled from x
m	-	Order of polynomial
var	-	Variance
β	-	Vector of regression parameter
ε	-	Error vector
K	-	Positive constant
ξ	-	Probability measure on X
d	-	Exact design

M	-	Information matrix
p	-	Dimension of information matrix
$\Phi^T \Phi$	-	Data covariance matrix or design matrix
$x_{\tilde{i}}^{d}$	-	New sample data
$\breve{d}$	-	Distance between a possible new sample and existing sample
x_i	-	Existing sample
w_i	-	Weights
E	-	Total error of the system
$S1$	-	Numbers of neurons and centers
$\phi_k(.)$	-	Basic Function
$\Vert . \Vert_2$	-	Euclidean norm
ζ	-	Spread or “width” of RBF
μ_j	-	Means of data points
J	-	Sum of square clustering function

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Overview

The problem of function approximation involves the modeling of an unknown function from a training set of input/output pairs that represent the function. This project involves the use of Neural Networks (RBF) for function approximation.

The RBF neural network has been widely used for function approximation due to its structural simplicity and faster learning abilities. The training procedures are well known. An RBF neural network is generally trained in two steps one after another. In the first step, the centers of hidden layer neurons are selected. Then the weights between the hidden and output layers are estimated. The centers of the hidden layers neurons for an RBF neural network are selected in different ways. In this project, these centers are selected by using k-means clustering algorithm.

There are several methods of selecting training data from input space for neural networks which include D-optimal and Max-min design approaches. Consider a function approximation problem for a limited amount of data, say *(N)*. Randomly select the *N* data set for conventional training algorithm. Then select *N* data set by using experimental design approach to train the Neural Networks again. Compare both methods by their approximation to the actual function.

1.2 Objectives of the Project

The objective of this project is to improve the training process of neural network using experiment design approaches.

For phase I, the objective is to implement function approximation using Least Square (LS) Technique and to compare the performance between the classical approximation, D-optimal and Maxmin Distance (MD) Approach approximation.

While for phase II, the objective is to implement function approximation using Neural Network (Radial Basic Function as case study) and to compare the performance between the classical approximation, D-optimal and Maxmin Distance (MD) Approach approximation.

1.3 Scope Of Project

Phase I:

a) Construct a function approximation program using LS technique in Matlab environment.
b) Understand Experiment Design; focusing on D-optimal and MD.
c) Apply Experiment Design for function approximation problem.
d) Search for the best point for the approximation, assuming the unknown function is known.
e) Compare the best point with the point proposed by Experiment Design.

Phase II:

a) Perform function approximation using Neural Network (NN)
b) Radial Basis Function (RBF) learning algorithm is implemented using Matlab.
c) NN function approximation without Experiment Design
d) NN function approximation with Experiment Design
e) Compare the performance.

1.4 Project Layout

Firstly, chapter 1 comprises the problem statements, objectives and scopes of the project. An overall of the framework done is briefly specified in the objectives and scopes of the project.

Chapter two presents the introduction of Experiment Design, included D-optimal and Maxmin Distance Approach, Least Square Techniques, Neural Networks where RBF as the case study.

Chapter three shows the preliminary result and discussion base on implementation of function approximation using Least Square Technique.

While Chapter four illustrate the result and the discussion base on the RBF NN training using Experiment Design. Conclusion and recommendation for future works are written in Chapter five.

REFERENCES

1. M. Mihailescu, A. Samuila,A. Urs, R. Morar, A. Iuga,2000, " *Computer-Assisted Experimental Designs for the Optimization of* Electrostatic *Separation Processes*", 2000 IEEE pp.687-694
2. S. Cheng, X. Hong and C.J. Harris, **"***Sparse Multioutput Radial Basis Function Network Construction Using Combined Locally Regularized Orthogonal Least Square and D-optimality Experimental Design*", IEE Proc.-Control Theory Appl. Vol. 150. No.2 Mar 2003. pp.139-146
3. X. Hong and C.J. Harris, **"***Nonlinear Model Structure Design and Construction Using Orthogonal Least Squares and D-optimality Design*", IEEE Trans. Neural Network., 2002,13,(5), pp. 1245-1250
4. Timothy W. Simpson, Jesse D. Peplinski, Patrick N. Koch, and Janet K. Allien, 1997. "*On The Uses of Statistics in Design and the Implications for Deterministic Computer Experiments*."1997 ASME Design Engineering Technical Conferences.
5. T. Poggio and F Girosi, "*Network for Approximation and Learning*." Proceeding of IEEE, volume 78, pp1481-1497,1990
6. K.A. Toh and K.Z. Mao, "*A Global transformation Appraoch To RBF Neural Network Learning*", IEEE. pp96-99,2002
7. W. Kaminski and P Struwillo, "*Kemel Ortbogonalization in Radial Basis Function Neural Networks.*" 1EEF Trans, on Neural Network, pp1177-1183, 1997.
8. J. Moody and C. Darken, "*Fast Learning in Network for Locally-tuned Processing Unit*.", Neural Computation, pp 281-294, 1989.
9. T. Kohenen, " *Self-Organizaing Maps*", Spring-Verlag, Germany, 1999
10. W. Pedryez, "*Conditional Fuzzy Clustering in the Design Of Radial Basis Function* "

11. Z.Uykan, C. Guzelis, M.E.Celebi and H. N Kolvo, "*Analysis of Input-Output clustering for Determining Centers of RBF*", IEEE Trans. On Neural Networks, pp 851-888, 2000.
12. S. Chen, S.A. Billings, and P.M. Grant, "*Recursive Hybrid Algorithms for Nonlinear System Identification using Radial Basic Function Networks.*" International Journal of Control, pp1051-1070, 1992.
13. S. Chen, F. Cowan, and PM. Grant, "*Orthogonal Least Squares Learning Algorithms for Radial Basic Functions*" IEEE Trans. On Neural Networks, pp 302-309, 1991.
14. J.B. Gomm and D.L. Yu, "*Selecting Radial Basic Function Networks Centers with Recursive Orthogonal Least Squares Training*" IEEE Trans. On Neural Networks, pp 306-314, 2000.
15. R. Schokopf, K.K. Sung, C.J.C. Burges, F. Girosi, P. Niyogi, T. Poggio, and V.N. Vapnik, "*Comparing Support Vector Machines with Gaussian Kernels on Radial Basic Function Classifiers.* " IEEE Trans. On Signal Processing, pp 2758-2765, 1997.
16. K.A. Toh, " *Global Energy Minimization: A Transformation Approach.* " In Preceedings of Third International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR), pp 391-406. 2001.
17. D.Q. Gao, G.X. Yang, " *Adaptive RBF Neural Network For Pattern Classifications.* " 2002 IEEE pp 846-851.
18. A. Baraldi, P. Blonda, G. Satalino, A. D'Addabbo, and C. Tarantino, " *RBF Two Stage Leaning Networks Exploiting Supervised Data in The Selection of Hidden Unit Parameters : An Application to SAR Data Classification.* " IEEE, pp 672-674, 2000.
19. C. M. Bishop, Neural Network for Pattern Recognition, Clarendon Presss: Oxford (UK), 1995.
20. N. Karayiannis, "*Growing Radial Basic Neural Netowrks: Merging Supervised and Unsupervised Learning with Network Growth Techniques.*" IEEE Trans. On Neural Networks, 1998.
21. N.Chantarat, N.Zheng, T.T. Allen, D. Huang, "*Optimal Experimental Design For*

Systems Involving Both Quantitative and Qualitative Factors", Proceedings of the 2003 Winter Simulation Conference. pp 556-564

22. A.C Atkinson and A.N Donev, "*Optimum Experimental Designs*", Oxford Science Publications, Oxford, 1992
23. A. Biran and M. Breiner, " *Matlab for Engineers*", Addsion-Wesley Publishing Company,1992