A Givens Rotation Based Fast Backward Elimination Algorithm For RBF Neural Network Pruning

X.Hong and S. A. Billings
Department of Automatic Control and Systems Engineering,
University of Sheffield, Mappin Street, Sheffield S1 3JD

Abstract — A fast backward elimination algorithm is introduced based on a QR decomposition and Givens transformations to prune radial basis function networks. Nodes are sequentially removed using an increment of error variance criterion. The procedure is terminated by using a prediction risk criterion so as to obtain a model structure with good generalisation properties. The algorithm can be used to postprocess radial basis centers selected using a k means routine and in this mode provides a hybrid supervised center selection approach.

Keywords — neural networks, backward elimination, prediction risk, Givens rotation

Research Report No. 643

Aug, 1996

1 Introduction

The Radial basis function (RBF) model was traditionally used for strict interpolation in multi-dimensional space(Powell, 1985). More recently, RBF neural networks have been employed in non-linear systems identification. The centers are assumed to sample the data set and reflect the distribution of the data, but the set of candidate centers can be very large, and in practice, a network with a finite basis selected from the data set is usually adopted. One approach is to randomly select the centers from the data(Broomhead and Lowe, 1988). Alternatively a k-means clustering technique can be employed(Moody and Darken, 1989). Usually the network weights are learnt at a later stage using least squares methods.

Selecting a finite basis from a large set of candidates corresponds to the classical model subset problem in linear regression(Draper and Smith, 1981). The forward regression method has been successfully used for RBF neural networks and other nonlinear system models(S.Chen *et al*, 1989, 1991). An alternative approach is to use backward elimination. Backward elimination starts by building the full model using all the basis funtions, the full model could be the whole data set, or a predetermined set obtained from k-means clustering. The basis functions are then eliminated one at a time based on the least deterioration in model fit. The use of the backward elimination method for RBF networks is analogous to the pruning procedure in MLP networks(R.Reed, 1993). One drawback of the backward elimination procedure is that it is computationally expensive compared with forward regression. In this study, a new backward elimination algorithm that is computationally fast is introduced, based on the orthogonal-triangular decomposition of the regression matrix and a Givens transformation. A prediction risk criterion is used as a measure of the generalisation capability of the resulting model to terminate the procedure(Barron,A.R., 1984, Liu, 1995). The new method can be used as a supervised center selection approach from the full model set, or it can be used together with a k-means clustering method to constitute a hybrid supervised method.

2 RBF Neural Network Formulation and The k-means Clustering Method

A RBF neural network can be formulated as

$$y(t) = \sum_{i=1}^{M} p_i(t)\theta_i + \varepsilon(t) \tag{1}$$

where $t = 1, 2, 3, \cdots, N$, and N is the sample size of the estimation set. The regressors take the form

$$p_i(t) = \Phi(\|\mathbf{x}(t) - \mathbf{c}_i\|, \beta_i) \tag{2}$$

$$\mathbf{x}(t) = [y(t-1), \cdots, y(t-n_y), u(t-1), \cdots, u(t-n_u), \varepsilon(t-1), \cdots, \varepsilon(t-n_\varepsilon)] \tag{3}$$

$\|\bullet\|$ denotes the Euclidean norm, β_i are some positive scalars called widths, $\Phi(\|\bullet\|, \beta_i)$ is a function from $\mathbf{R}^+ \rightarrow \mathbf{R}$, and $\mathbf{c}_i \in \mathbf{R}^{(n_y+n_u+n_\varepsilon)}, 1 \leq i \leq M$ are the RBF centers. The thin-plate-spline function

$$\Phi(x) = x^2 \log x \qquad (x \geq 0) \tag{4}$$

will be used in the present study.

The k-means clustering algorithm partitions the data set into k clusters and determines k cluster centers. Usually, the number of centers are predetermined. The k-means clustering method starts with initial centers $\mathbf{c}_i(0) : 1 \leq i \leq M$ and an initial learning rate $\alpha(0)$, and computes the distances

$$\rho_i(t) = \|\mathbf{x}(t) - \mathbf{c}_i(t-1)\|, 1 \leq i \leq M \tag{5}$$

to find a minimum distance

$$k = \arg[\min\{\rho_i(t), 1 \leq i \leq M\}] \tag{6}$$

If $k = \arg[\min\{\rho_i(t), 1 \leq i \leq M\}]$ then

$$\begin{cases} \mathbf{c}_k(t) = \mathbf{c}_k(t-1) + \alpha(t)[\mathbf{x}(t) - \mathbf{c}_k(t-1)] \\ \mathbf{c}_i(t) = \mathbf{c}_i(t-1), \qquad \forall i \neq k \end{cases}$$

The learning rate should be $\alpha(t) < 1$ and should slowly decrease to zero. A typical choice is

$$\alpha(t) = \frac{\alpha(t-1)}{\sqrt{1 + \text{int}(t/M)}} \tag{7}$$

where $\text{int}(x)$ denotes the integral part of x.

3 A Givens Rotation Based Fast Backward Elimination Algorithm

Eq.(1) can also be written in a matrix form as

$$\mathbf{y} = \mathbf{P}\Theta + \Xi \tag{8}$$

where $\Theta^T = [\theta_1, \cdots, \theta_M]$, $\mathbf{y}^T = [y(1), \cdots, y(N)]$, $\Xi^T = [\varepsilon(1), \cdots, \varepsilon(N)]$, and

$$\mathbf{P} = \begin{bmatrix} p_1(1) & p_2(1) & \cdots & p_M(1) \\ p_1(2) & p_2(2) & \cdots & p_M(2) \\ \cdots & \cdots & \cdots & \cdots \\ p_1(N) & p_2(N) & \cdots & p_M(N) \end{bmatrix} = [\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_M]$$

A QR decomposition can then be performed on the regression matrix $\mathbf{P}$

$$\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_M] = [\mathbf{q}_1, \mathbf{q}_2, \cdots, \mathbf{q}_M]\mathbf{R} \tag{9}$$

The Increment of Error Variance due to the elimination of the last column $\mathbf{p}_M$ can be computed as $IEV_M = \frac{1}{N} < \mathbf{q}_M, \mathbf{y} >^2$, where $< \bullet >$ denotes the inner product. If each regressor in $\mathbf{P}$ is in turn moved to the last column, the respective increment of error variance due to the effect of this being removed can be calculated. For example, if the column $\mathbf{p}_i$ is permuted with the last column $\mathbf{p}_M$, then the permutation matrix of the regression matrix $\mathbf{P}$ is

$$\tilde{\mathbf{P}}_{i,M} = [\mathbf{p}_1, \cdots, \mathbf{p}_{i-1}, \mathbf{p}_M, \mathbf{p}_{i+1}, \cdots, \mathbf{p}_i] = [\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_M]\tilde{\mathbf{I}}_{i,M} \tag{10}$$

where $\tilde{\mathbf{I}}_{i,M}$ is the permutation matrix of a unit matrix where the ith and M'th column are interchanged. Substitute Eq.(9) into Eq.(10) to yield

$$\tilde{\mathbf{P}}_{i,M} = [\mathbf{q}_1, \mathbf{q}_2, \cdots, \mathbf{q}_M]\mathbf{R}\tilde{\mathbf{I}}_{i,M} \tag{11}$$

Since $\mathbf{R}\tilde{\mathbf{I}}_{i,M}$ is no longer triangular, $\mathbf{q}_1, \mathbf{q}_2, \cdots, \mathbf{q}_M$ are no longer an orthogonal basis of $\tilde{\mathbf{P}}_{i,M}$. However, a series of Givens transformations can be used to form a new QR decompostion for $\tilde{\mathbf{P}}_{i,M}$. Let $\hat{\mathbf{Q}} = [\hat{\mathbf{q}}_1, \cdots, \hat{\mathbf{q}}_M]$ denote the new basis. The increment of error variance due to the elimination of the regressor $\mathbf{p}_i$ can then be computed as $IEV_i = \frac{1}{N} < \hat{\mathbf{q}}_M, \mathbf{y} >^2$.

A total of $\frac{(M-i)(M-i+1)}{2}$ Givens rotations will be needed to compute the new QR decomposition of $\tilde{\mathbf{P}}_{i,M}$. The aim of the fast backward elimination algorithm is to sequentially change the position of each regressor to the last column with the main advantage of decreasing the number

of Givens rotations. The procedure can be summarized as:

(i). Initially, the full model is composed of M regressors.

(ii). Perform QR decomposition on the regression matrix P using Givens transformations(G.A.F, Seber, 1976). Compute IEV_M.

(iii). Permute $\mathbf{p}_{M-1}$ with $\mathbf{p}_M$, so that the new regression matrix becomes $[\mathbf{p}_1,\cdots,\mathbf{p}_{M-2},\mathbf{p}_M,\mathbf{p}_{M-1}]=\mathbf{QR}\tilde{\mathbf{I}}_{M-1,M}$. Only one Givens rotation $\mathbf{G}^{(1)}$ on the $(M-1)$th and Mth rows of the matrix $\mathbf{R}\tilde{\mathbf{I}}_{M-1,M}$ will be needed to retriangularize it. Denote $\mathbf{R}^{(1)}=\mathbf{G}^{(1)}\mathbf{R}\tilde{\mathbf{I}}_{M-1,M}$ as the new triangular matrix. The new regression matrix then has a new orthornormal-triangular decomposition

$$[\mathbf{p}_1,\cdots,\mathbf{p}_{M-2},\mathbf{p}_M,\mathbf{p}_{M-1}]=\mathbf{Q}^{(1)}\mathbf{R}^{(1)} \tag{12}$$

where $\mathbf{Q}^{(1)}=\mathbf{Q}\{\mathbf{G}^{(1)}\}^T$, which is computed by rotating the last two columns of the orthonormal matrix $\mathbf{Q}$. Compute IEV_{M-1}.

(iv). At the kth step, $k=2,\cdots,M-1$, the existing regression matrix takes the form $[\mathbf{p}_1,\mathbf{p}_2,\cdots,\mathbf{p}_{M-k},\mathbf{p}_M,\mathbf{p}_{M-1},\cdots,\mathbf{p}_{M-k+1}]$, and the existing orthonormal and triangular matrices from the last step are denoted as $\mathbf{Q}^{(k-1)}$ and $\mathbf{R}^{(k-1)}$ respectively. In order to place the regressor $\mathbf{p}_{M-k}$ in the last column, sequentially permute the regressor $\mathbf{p}_{M-k}$ with its right adjacent regressors $\mathbf{p}_M$, $\mathbf{p}_{M-1},\cdots$, and $\mathbf{p}_{M-k+1}$. Thus the new regression matrix

$$\begin{aligned}&[\mathbf{p}_1,\mathbf{p}_2,\cdots,\mathbf{p}_{M-k-1},\mathbf{p}_M,\mathbf{p}_{M-1},\cdots,\mathbf{p}_{M-k+1},\mathbf{p}_{M-k}]\\ =\;&\mathbf{Q}^{(k-1)}\mathbf{R}^{(k-1)}\tilde{\mathbf{I}}_{M-k,M-k+1}\tilde{\mathbf{I}}_{M-k+1,M-k+2}\cdots\tilde{\mathbf{I}}_{M-1,M}\end{aligned} \tag{13}$$

Only k Givens transformations are needed to retriangularize the matrix

$$\mathbf{R}^{(k-1)}\tilde{\mathbf{I}}_{M-k,M-k+1}\tilde{\mathbf{I}}_{M-k+1,M-k+2}\cdots\tilde{\mathbf{I}}_{M-1,M}$$

Denote

$$\mathbf{R}^{(k)}=\mathbf{G}^{(k)}\mathbf{R}^{(k-1)}\tilde{\mathbf{I}}_{M-k,M-k+1}\tilde{\mathbf{I}}_{M-k+1,M-k+2}\cdots\tilde{\mathbf{I}}_{M-1,M} \tag{14}$$

as the new triangular matrix, where $\mathbf{G}^{(k)}=\mathbf{G}^{(k)_k}\mathbf{G}^{(k)_{k-1}}\cdots\mathbf{G}^{(k)_1}$ is a series of k Givens transformations, in which, $\mathbf{G}^{(k)_j}$, $\quad j=1,\cdots,k$ is a Givens transformation applied to $\mathbf{R}^{(k)_{j-1}}\tilde{\mathbf{I}}_{M-k+j-1,M-k+j}$ on the $(M-k+j-1)$th, and $(M-k+j)$th rows, where $\mathbf{R}^{(k)_0}=\mathbf{R}^{(k-1)}$, and

$$\mathbf{R}^{(k)_j}=\mathbf{G}^{(k)_j}\mathbf{R}^{(k)_{j-1}}\tilde{\mathbf{I}}_{M-k+j-1,M-k+j},\quad j=1,\cdots,k \tag{15}$$

are triangular matrices. Denote

$$\mathbf{R}^{(k)} = \mathbf{R}^{(k)_k} \tag{16}$$

Eq.(13) can be rewritten as

$$\begin{aligned} & [\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_{M-k-1}, \mathbf{p}_M, \mathbf{p}_{M-1}, \cdots, \mathbf{p}_{M-k+1}, \mathbf{p}_{M-k}] \\ = \; & \mathbf{Q}^{(k-1)}\{\mathbf{G}^{(k)}\}^T \mathbf{G}^{(k)} \mathbf{R}^{(k-1)} \tilde{\mathbf{I}}_{M-k,M-k+1} \tilde{\mathbf{I}}_{M-k+1,M-k+2} \cdots \tilde{\mathbf{I}}_{M-1,M} \\ = \; & \mathbf{Q}^{(k)} \mathbf{R}^{(k)} \end{aligned} \tag{17}$$

where $\mathbf{Q}^{(k)} = \mathbf{Q}^{(k-1)}\{\mathbf{G}^{(k)}\}^T$ is the new orthorgonal basis. Numerically, the orthornormal matrix $\mathbf{Q}^{(k)}$ is computed by rotating columns of the $\mathbf{Q}^{(k-1)}$ using orthorgonal matrices $\{\mathbf{G}^{(k)_1}\}^T$,$\{\mathbf{G}^{(k)_2}\}^T$, $\cdots$, $\{\mathbf{G}^{(k)_k}\}^T$ sequentially. Compute IEV_{M-k}.

(vi). Find the minimum increment of error variance due to the elimination of $\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_M$.

$$n = \arg[\min\{IEV_i, i = 1, \cdots, M\}] \tag{18}$$

(v). A prediction risk describes the expected performance of an estimator in predicting new observations, which is defined as(Barron, A.R.,1984, Liu,1995)

$$\varepsilon_{val}^2 = \varepsilon_{est}^2 + \sigma^2 \frac{2n_{eff}}{N} \tag{19}$$

where the ε_{val}^2 and ε_{est}^2 are the sum of squared error on the validation set and the estimation set respectively. In practice, the variance of the noise σ^2 is replaced by the variance of prediction error in the estimation set ε_{est}^2. The regressor $\mathbf{p}_n$ is eliminated from the regression matrix $\mathbf{P}$ while the prediction risk is monitored simultaneously. If a minimum of the prediction risk is not reached, go back to step (ii). The new regression matrix is also denoted as $\mathbf{P}$ for simplicity of notation, and the number of the regressors M is decreased by 1. The procedure terminates at an optimal structure which is evaluated by the prediction risk criterion, where the $n_\theta = \arg[\min\{\varepsilon_{val}^2, \forall n_{eff}\}]$ is the number of the parameters.

Remarks:

(i). To eliminate one regressor or basis function from M regressors or basis functions, only $\frac{M(M-1)}{2}$ Givens row rotations are needed to form new triangular matrices, and another $\frac{M(M-1)}{2}$ Givens column rotations are needed to form a new orthnormal basis.

(ii). The Givens rotation-based computation approach retriangularizes the matrix by rows, and

re-orthornormalise the matrix by columns. The fetch and store is very efficient. Also, the inherent recursive nature of the Givens transformation minimizes the memory requirements for intermediate variables such as $\mathbf{R}^{(k)_1}, \mathbf{R}^{(k)_2}, \cdots, \mathbf{R}^{(k)_{k-1}}$. These matrices are accommodated by the same variable in the program which is updated by sequential row operations.

4 Numerical Examples

Example 1: Consider a NAR time series

$$\begin{aligned} y(t) &= (0.8 - 0.5\exp(-y^2(t-1)))y(t-1) - (0.3 + 0.9\exp(-y^2(t-1)))y(t-2) \\ &\quad +0.1\sin(3.1415926y(t-1)) + \varepsilon(t) \end{aligned} \tag{20}$$

where the noise $\varepsilon(t)$ was a gaussian white sequence with mean zero and variance 0.01. The estimation data set consisted of 500 data points. A thin-plate-spline RBF was used to model the system. The structure of the RBF model was defined by $n_y = 2$. The initial centers were randomly selected from the data set, and then the k-means clustering method was used to select the centers(S.Chen, *et al*, 1992). The parameters in the k-means clustering method were chosen to be $\alpha(0) = 0.9$. and the number of centers was set to be 30. The fast backward elimination algorithm was then applied to postprocess the centers obtained from the k-means routine. Twenty redundant centers were eliminated and the number of centers was reduced to 10. The distribution of the data set and the positions of the centers is plotted in Fig.1. The evolution of the increment of error variance in the training set as the centers were removed is shown in Fig.2.

Model validity tests are procedures which are used to detect the inadequacy of a fitted model. Correlation based validation involves computing correlation functions composed of model residuals and system inputs and testing if these satisfy certain conditions given in the form of confidence intervals. The new higher order correlation tests which use model residuals combined with system inputs and outputs(Billings and Zhu,1994) were used in the present study. The results plotted in Fig.3 indicate that the pruned network is appropriate.

Example 2. Consider a nonlinear system

$$\begin{aligned} y(t) &= -0.6377y(t-1) + 0.07298y(t-2) + 0.03597u(t-1) + 0.06622u(t-2) \\ &\quad +0.06568u(t-1)y(t-1) + 0.02375u^2(t-1) + 0.05939 + \varepsilon(t) \end{aligned} \tag{21}$$

where the system input $u(t)$ was a uniformly distributed sequence, and the noise $\varepsilon(t)$ was a

gaussian white sequence with mean zero and variance 0.05. A data sequence of 500 samples was generated, and a thin-plate-spline RBF network was used to model the system. The input nodes of the system were defined as $\{u(t-2), u(t-1), y(t-2), y(t-1)\}$. The initial centers were randomly selected from the data set, and the k-means clustering method was then used to select the centers. The number of centers was defined to be 40. The fast backward elimination algorithm was then applied, and the number of centers was reduced to 12. The increment of error variance in the training data set plotted against the number of eliminated centers is illustrated in Fig.4. and shows that 28 redundant centers were eliminated. The procedure stoped at the circle point. The one-step ahead prediction error was used for model validation of the final network and the results plotted in Fig.5 demonstrate that the pruned network is a valid model of the system.

5 Conclusions

A backward elimination algorithm has been introduced for RBF neural network pruning. An initial set of RBF neural network centers can be predetermined using k-means clustering or several alternative methods. But the complexity of the resulting network may be larger than necessary. The backward elimination algorithm starts with a comparatively large network and removes the centers which increase the training error least. The algorithm is based on the QR decomposition of the regression matrix and a Givens triangularisation method. The algorithm is effecient in that the number of Givens rotations is small due to the recursive nature of the algorithm. The effectiveness of the new algorithm has been demonstrated using numerical examples, and the final pruned networks were tested using model validity tests. Although the algorithm was introduced based on the RBF architecture, it is applicable to the general class of linear-in-the-parameters system models.

6 Acknowledgements

SAB gratefully acknowledges that part of this work was supported by EPSRC. XH expresses her thanks for the award of an ORS scholarship which made this study possible.

References

[1] Barron,A.R. (1984). Predicted squared error: A criterion for automatic model selection. In Stanley, J.F., editor, Self-organizing Methods in Modelling GMDH Type Algorithms, pages 87-103. Marcel Dekker, Inc. New York.

[2] S.A.Billings and Zhu,Q.M. (1994). Nonlinear model validation using correlation tests. *International Journal of Control*, Vol. 60, pp1107-1120.

[3] Broomhead,D.S., and Lowe, D. (1988). Multivariable functional interpolation and adaptive networks. *Complex Systems*, Vol. 2, pp321-355.

[4] S.Chen and S.A.Billings, C.F.N.Cowan and Grant, P.M. (1990). Non-linear system identification using radial basis functions.*International Journal of Systems Sci.*, Vol. 21, No.12, pp2513-2539

[5] S.Chen, S.A.Billings and Luo, W. (1991). Orthorgonal least squares methods and their applications to non-linear system identifiction.*International Journal of Control*, Vol. 50, pp1873-1896.

[6] S.Chen and S.A.Billings and Grant, P.M. (1990). Recursive hybrid algorithm for non-linear system identification using radial basis functions.*International Journal of Control*, Vol. 55, No.5, pp1051-1070

[7] Draper, N and Smith, H. (1981). *Applied Regression Analysis*, 2nd edn. Wiley, New York.

[8] Liu, Y. (1995). Unbiased estimate of generalisation error and model selection in neural network.*Neural Networks*, 8(2), pp311-341.

[9] Moody,J.,and Darken,C. (1989). Fast-learning in networks of locally-tuned processing units.*Neural computation*, NN-4, pp740-747.

[10] W.Morven Gentleman (1973). Least squares computations by Givens transformations without square roots.*J. Inst. Maths. Applics*, 12, pp329-336.

[11] Powell,M.J.D. (1985). *IMA Conf. on Algorithms for the Approximation of Functions and Data* RMCS Shrivenham.

[12] Reed, R. (1993). Pruning algorithm–a survey.*IEEE Trans. Neural Networks.*, NN-4, pp740-747.

[13] Seber, G.A.F. (1977). *Linear Regression Analysis*. John Wiley and Sons, New York.

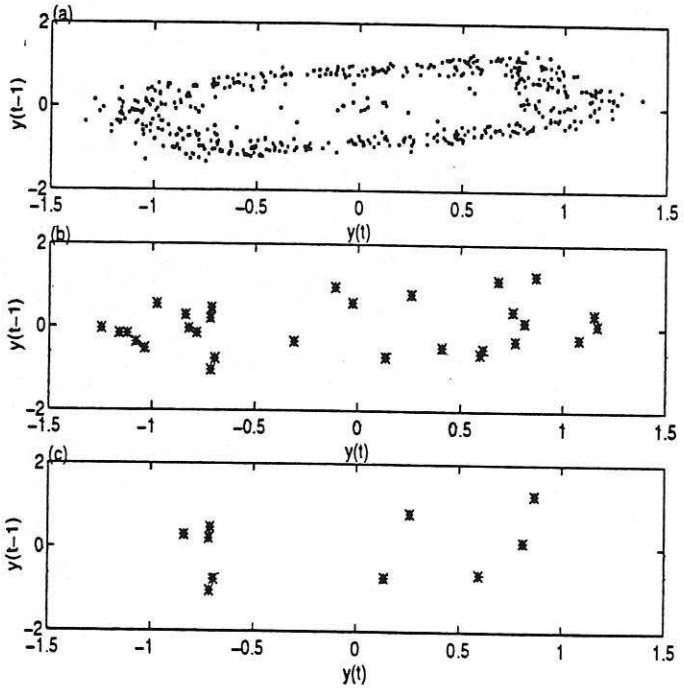

Figure 1: Distribution of observations and RBF centers in example 1; (a)Distribution of observations; (b) Positions of the original RBF centers using k-means clustering; and (c) Positions of the final centers after fast backward elimination

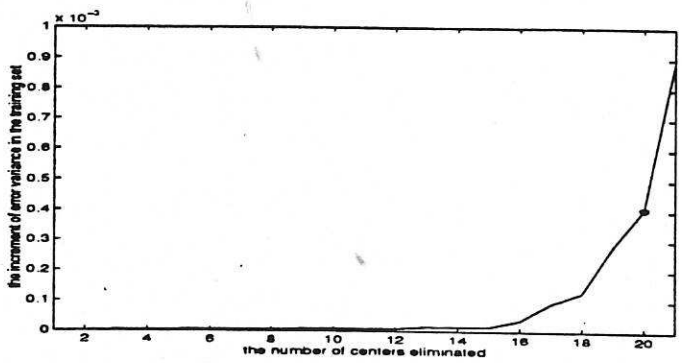

Figure 2: The increment of error variance in example 1: starting from 30 centers, the algorithm terminated at the circle point when 20 centers were eliminated.

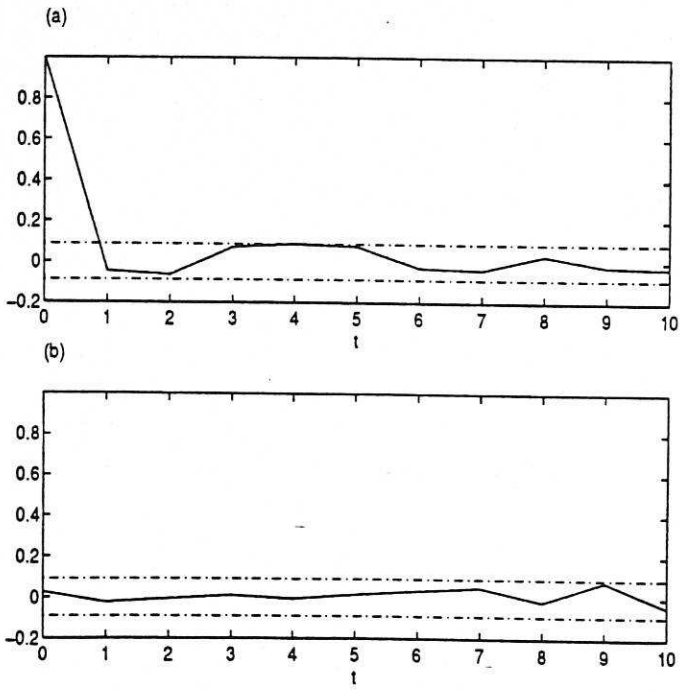

Figure 3: Model validity tests for example 1; (a) Model validity test $\Phi_{\varepsilon\varepsilon}(\tau)$; and (b) Model validity test $\Phi_{(y\varepsilon)\varepsilon^2}(\tau)$

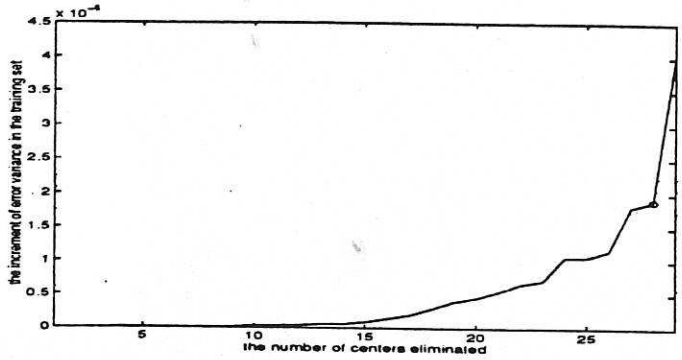

Figure 4: The increment of error variance in example 2: starting from 40 centers, the algorithm terminated at the circle point when 28 centers were eliminated

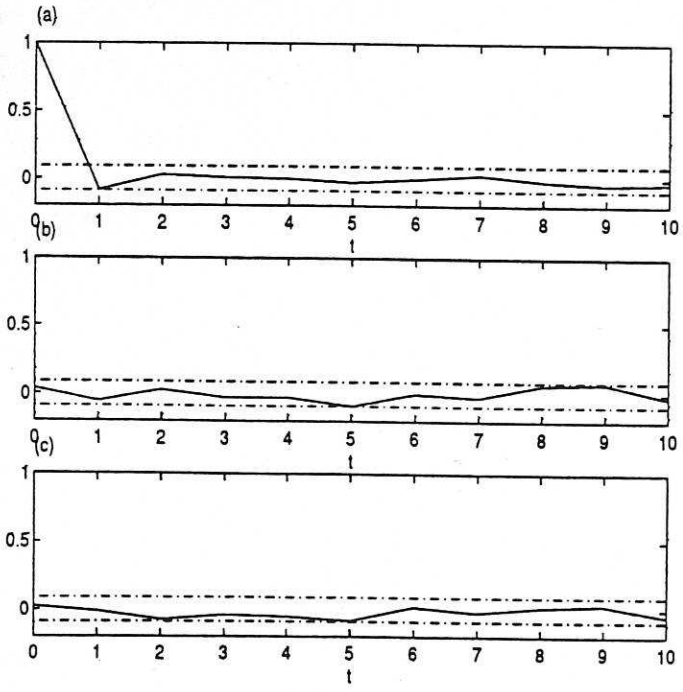

Figure 5: Model validity tests for example 2; (a) Model validity test $\Phi_{\varepsilon\varepsilon}(\tau)$; (b) Model validity test $\Phi_{(y\varepsilon)\varepsilon^2}(\tau)$ and (c) Model validity test $\Phi_{(y\varepsilon)u^2}(\tau)$