FINANCE & TECHNOLOGY

Journal of Computational Intelligence in Finance

ADAPTIVE SUPERVISED LEARNING DECISION NETWORKS FOR TRADING AND PORTFOLIO MANAGEMENT

LEI XU AND YIU-MING CHEUNG

A trading and portfolio management system is proposed, based on an Adaptive Supervised Learning Decision Network, *which learns the best past investment decisions directly instead of making predictions first and then making investment decisions based on the predictions. Without any additional effort, this network can be realized directly utilizing any existing adaptive supervised-learning neural network. Here, we propose to use a recently proposed adaptive* Extended Normalized Radial Basis Function *network with* Matched Competitive Learning. *We demonstrate with experimental results that the proposed approach can generate appreciable profits from trading in the foreign exchange markets.*

1. INTRODUCTION

Many trading systems have been proposed, based on different methodologies and investment strategies, to help traders maximize profits. One widely-used type of trading system consists of two modules: (1) a *prediction module* followed by (2) a *trading module*. First, a prediction is generated by a prediction module previously trained using some prediction criterion such as the minimization of the mean square error (MSE). Then, a trading module is used to produce a trading signal based on the prediction and some investment strategy. Since this type of trading system is optimized to a prediction criterion that is poorly correlated with profitability, it usually leads to sub-optimal performance as a trading system.

One alternative to the above approach is to utilize a prediction criterion more correlated with common trading stategies such as those proposed by Caldwell [1995]. Another approach involves the use of certain types of trading systems recently proposed by Bengio [1996] and Kang et al. [1996], in which the prediction module and the trading module are merged into a single system that optimizes returns directly instead of the prediction criterion. Preliminary experiments have shown that such trading systems outperform those optimized based on the best forecasts.

There is also another choice. We can build a system to learn the desired investment decision signal via a supervised learning algorithm. The desired investment decision for each day is obtained just after that day is passed. As an example, we propose an *Adaptive Supervised Learning Decision (ASLD) Network* as a trading and portfolio management system. Without extra efforts, the ASLD network can be easily implemented any existing adaptive supervised learning algorithms. Here, we use a recently proposed adaptive *Extended Normalized Radial Basis Function (ENRBF)* network trained by *Coordinated Competitive Learning (CCL)*, which is described in Xu [1997a,b], and summarized in Appendix A. Here, we demonstrate its utility for trading in the foreign exchange markets. Experimental results show that the proposed system can generate appreciable profits.

2. ADAPTIVE SUPERVISED LEARNING DECISION NETWORK

We begin with a portfolio of m foreign exchange rate series $\{z^i_t, i = 1,2,\ldots,m\}$ including one risk-free rate series (i.e., the one that results from no trading activity). We make investments according to the following trading decision signal:

$$I^i_t = I^{a^i}_t \bullet I^p_t \qquad (1)$$

which consists of two components.

The first component is the allocation signal, I^i_t

where $I^{a^i}_t = 1$

refers to the i^{th} foreign currency of interest. In this study, we imposed the following constraint:

$$\sum_{i=1}^{m} I_t^{a^i} = 1$$

That is, we are limited to investing in, at most, only one currency each day. The second component is the position signal, I^p_t, with

$$I^p_t = \begin{cases} 1, & \text{which means to take long position} \\ 0, & \text{which means to take neutral position} \\ -1, & \text{which means to take short position} \end{cases} \quad (2)$$

At the current day, t, we can calculate yesterday's return by

$$r_t^i = -I_{t-1}^i\left(z_t^i - z_{t-1}^i\right) - \left|I_{t-1}^i - I_{t-2}^i\right|\gamma \quad (3)$$

with $1 \le i \le m$, where γ is a transaction cost rate. We know that the best investment decision for yesterday

$$I_{t-1} = \left\{I_{t-1}^i\right\}_{i=1}^{m}$$

should be the one that optimizes total returns

$$\sum_{i=1}^{m} r_t^i$$

which results in

$$I_{t-1}^i = [I_{t-1}^{a^i}, I_{t-1}^p]$$

with

$$I^{a^i}_{t-1} = \begin{cases} 1, & \text{if } i = j \text{ with } j = \text{argmax}_{1\le i\le m}(r_t^i) \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

and

$$I_{t-1}^p = \begin{cases} 1, & \text{if } z_t^i - z_{t+1}^i > 0 \\ 0, & \text{if } z_t^i - z_{t+1}^i = 0 \\ -1, & \text{if } z_t^i - z_{t+1}^i < 0 \end{cases} \quad (5)$$

In this way, we obtain a series that consists of the best decision signals for all previous days, including yesterday. We assume that there is some relationship as follows

$$[I_t, I_t^p] = f[I_{t-1}, I_{t-1}^p, z_t^i, z_{t-1}^i, \ldots, z_{t-d+1}^i | 1 \le i \le m] \quad (6)$$

We can use a supervised learning network to implement this relationship. We can train the network to fit the past known series of best decision signals and the foreign exchange rates by use of the maximum likelihood method. Then, we can use eq.(6) to predict today's best decision, I_t, with its element given by

$$I_t^i = \begin{cases} \hat{I}^{a^j}_t \bullet \hat{I}^p_t, & \text{if } i = j \text{ with } j = \text{argmax}_{1\le i\le m}(\hat{I}^{a^i}_t) \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

with $[\hat{I}^{a^i}_t, \hat{I}^p_t]$

being an element of the prediction given by eq.(6). Then, we can invest according to the trading signals generated. At the close of each day, we can refine our network by applying an adaptive learning algorithm to eq.(6). We call such a trained network an *Adaptive Supervised Learning Decision (ASLD)* network.

3. Extended Normalized RBF Networks with Matched Competitive Learning

In this paper, we used the *Extended Normalized Radial Basis Function (ENRBF)* network as our network model. For learning, we utilized both the batch method and the adaptive *Coordinated Competitive Learning (CCL)* algorithm as described in Xu [1997a,b].

The ENRBF network is given by

$$y = f(x) = \frac{\sum_{j=1}^{k}(w_j^t x + c_j)\exp[-0.5(x - m_j)^T \Sigma_j^{-1}(x - m_j)]}{\sum_{j=1}^{k}\exp[-0.5(x - m_j)^T \Sigma_j^{-1}(x - m_j)]} \quad (8)$$

where y and x correspond to

$$[I_t, I^p_t]$$

and

$$f[I_{t-1}, I_{t-1}^p, z_t^i, z_{t-1}^i, \ldots, z_{t-d+1}^i | 1 \le i \le m]$$

respectively. w^t_j, c_j, m_j, Σ^{-1}_j, $j = 1, \ldots, k$ are parameters of the network that need to be learned. In the literature, such networks are expected to be trained using a least squares learning technique. However, due to the difficulty of determining the parameters m_j and Σ^{-1}_j of the basis functions in practice, learning usually occurs as an approximation method that based on two separate steps.

First, we usually assume that $\Sigma_j = \sigma^2 I$ and that σ^2 is estimated roughly and heuristically, and use some kind of clustering analysis (e.g., the k-means algorithm) to group a dataset, D_x, into k clusters with the centers of these clus-

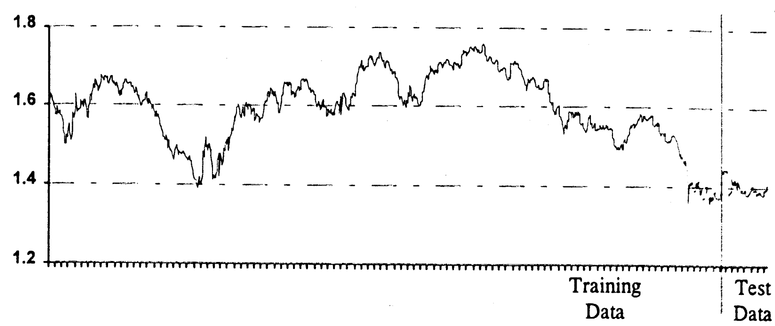

FIGURE 1. The USD-DEM rate series of 1096 data points. Each horizontal bar represents 10 data points.

ters characterized as m_j, $j = 1,\ldots, k$. Second, the output layer parameters, w_j and c_j, are determined using least squares learning techniques. Using this two-stage training approach, the centers are obtained directly from the input data without the need to make improvements to the regression relation between y and x.

By combining the ENRBF network with an alternative to the Mixture-of-Experts Model, which is described in Xu et al. [1995], the maximum likelihood learning of ENRBF networks, using the EM (expectation-maximization) algorithm described in Bishop [1995], can provide a more optimal solution to the parameters with regard to the basis functions and with regard to the output layer parameters. Of particular interest is that the algorithm can be simplified for both the batch and adaptive CCL algorithms with a speedup of more than one magnitude.

In this study, we used the batch CCL algorithm for the initial training based on past data points up to the data for yesterday, t-1. Then, we used the trained network to predict the best decision for today, t. For the next day, we updated the system using the adaptive CCL algorithm. Details on the batch and adaptive CCL algorithms are provided in Xu [1997a,b], and summarized in Appendix A.

4. Improving the Generalization Ability of the ASLD Network System

The generalization ability of the ASLD network system can be improved in two ways. First, we could select an appropriate k for the number of radial basis functions using a criterion and algorithm proposed in Xu [1997a]. Second, we could use the following formula as part of a simple heuristic softmax technique

$$I^{a_t^i} = \frac{\exp(r_{t+1}^i/\gamma)}{\sum_{i=1}^{m}\exp(r_{t+1}^i/\gamma)} \tag{9}$$

In the latter case, we refer to the resulting ASLD system as the Softmax ASLD (SASLD). In the next section, we will show that this simple regularization technique can improve performance considerably.

5. Experimental Results

We can demonstrate the usefulness of the ASLD system by simulating trading in the foreign exchange markets. For simplicity, here we will consider a single foreign exchange rate series of the US Dollar (USD) versus the German Deutschmark (DEM) for the period November 26, 1991 through August 14, 1995, as shown in Figure 1. The first 1025 data points are used as the training dataset, while the remaining 71 points are used in the testing stage. We assume that a trader can hold, at most, a long or a short contract valued at 50,000 DEM through a deposit of US$6,500 (which is the lowest marginal deposit required by the Hong Kong Financial Bureau). Also, we assume the transaction cost rate is 0.5% of the deposit. Note that the transaction costs for switching from a long contract to a short contract are twice the cost of switching from either a long or short contract to a neutral position.

We implemented the ASLD system using the ENRBF network as described by eq.(8) We trained the network using the following methods:

(a) an extended conventional two-stage algorithm as described by Jones et al. [1991], and denoted as the *Existing ENRBF ASLD* ,
(b) the above *Existing ENRBF ASLD* using the softmax technique described by eq.(9), and denoted as the *Existing ENRBF SASLD,*
(c) the adaptive CCL algorithm initialized using the training dataset by the batch CCL algorithm, and denoted as the *Adaptive CCL-ENRBF ASLD*,
(d) the above *Adaptive CCL-ENRBF ASLD* using the softmax technique described by eq.(9), and denoted as *Adaptive CCL-ENRBF SASLD.*

Shown in Figure 2 are the results using the *Existing ENRBF ASLD*. Profits gained using the trading signals indicated in Figure 2(a) are shown in Figure 2(b). Shown in Figure 3 are the results using the *Adaptive CCL-ENRBF ASLD*. Profits gained using the trading signals indicated in Figure 3(a) are shown in Figure 3(b). Shown in Figure 4 and 5 are the results using *Existing ENRBF SASLD* and *Adaptive CCL-ENRBF SASLD,* respectively. For comparison, all of the results are shown again in Figure 6. We also show the results of a prediction trading system reported in in Cheung et al. [1996].

From our experiments, we make the following observations:

1. In all cases, the ASLD systems considerably outperformed the prediction trading system described in Cheung et al. [1996],
2. The ASLD system trained using the *Adaptive CCL* algorithm considerably outperformed the extended conventional two-stage algorithm reported in Jones et al. [1991], and

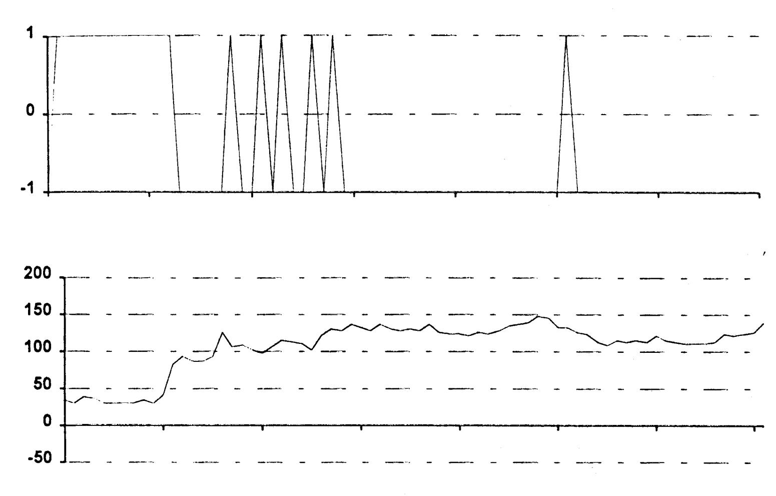

Figure 2. The results by *Existing ENRBF ASLD*. Upper graph: the trading signal on the test data [-1,1]. Lower graph: the profit gained (%).

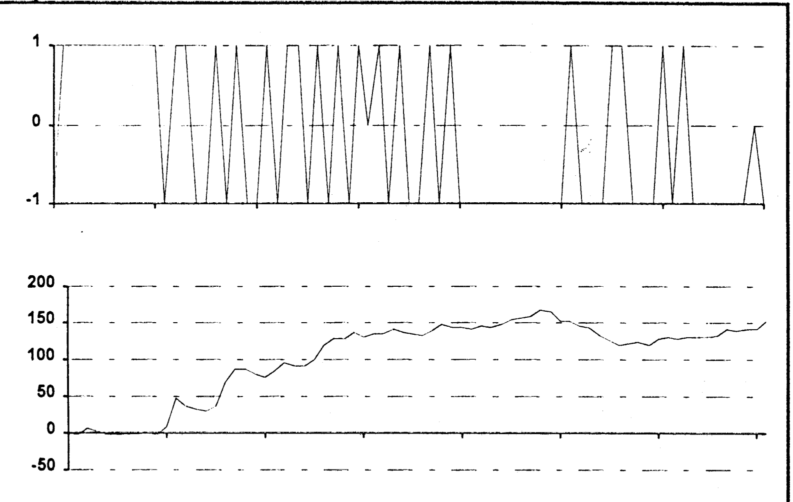

Figure 3. The results by *Adaptive CCL-ENRBF ASLD*. Upper graph: the trading signal on the test data [-1,1]. Lower graph: the profit gained (%).

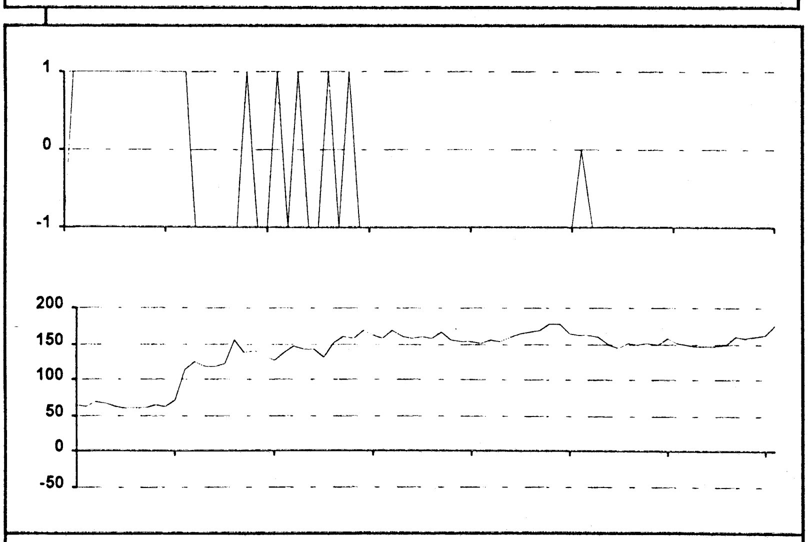

Figure 4. The results by *Existing ENRBF SASLD*. Upper graph: the trading signal on the test data [-1,1]. Lower graph: the profit gained (%).

3. The simple regularization by softmax according to eq.(9) considerably improved performance.

6. Conclusion and Discussion

As an alternative to those trading systems proposed in Bengio [1996] and Kang et al. [1996], which are based on optimizing returns, the proposed ASLD system predicts the best trading decisions directly. As shown in these preliminary experimental results, all of the ASLD systems considerably outperformed the prediction trading system reported in Cheung et al. [1996]; and the ASLD system trained using the *Adaptive CCL* algorithm described in Xu [1997a,b] considerably outperformed the extended conventional two-stage algorithm described in Jones et al. [1991]. Also of interest is that the use of the simple regularization by the softmax described by eq.(9) considerably improved performance. Based on the trading example using a foreign exchange rate, the profits obtained from the proposed ASLA system appear very promising, and, thus, the techniques presented deserve further exploration.

It can be expected that the performance of ASLD systems and the trading systems proposed by Bengio and Kang will be similar when the capacity of the network (e.g., the number k of the basis functions) is unlimited. However, when network capacity is limited, the ASLD systems and the others will produce different results. In either case, the characterization of the decision series and the generalization ability of the networks must be taken into account. The limited network size will introduce some amount of regularization and thus tend to improve a network's generalization ability. A criterion and algorithm for selecting the number of basis functions have been described for ENRBF networks. However, for the systems proposed by Bengio and Kang, such issues have not been considered, and, thus, some additional studies are needed, particularly given the differences in their learning techniques compared with the conventional supervised learning algorithms. In addition, the heuristic regularization technique described by eq.(9) appears quite beneficial, and deserves further analysis.

Finally, the methods presented overcome potential problems of similar systems with respect to (a) structural and temporal credit assignment difficulties, and (b) labeling procedures by human experts or automatic labeling algorithms which ignore input variables and do not consider the conditional distributions of price changes of the input variables, especially where actual transaction costs must be considered.

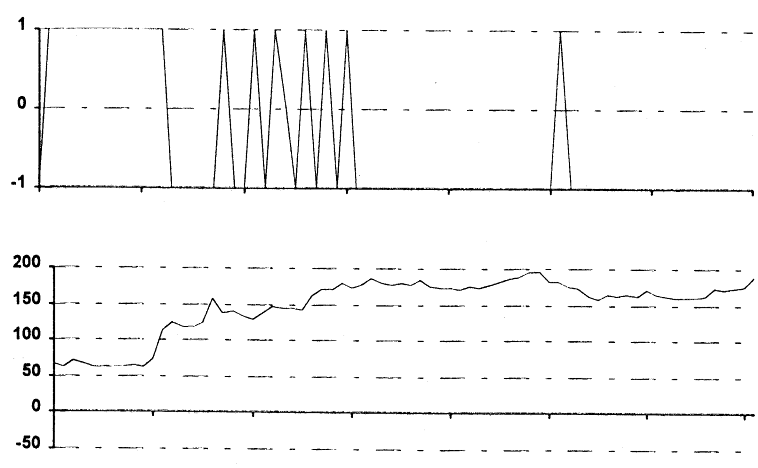

FIGURE 5. The results by *Adaptive CCL-ENRBF SASLD*. Upper graph: the trading signal on the test data [-1,1]. Lower graph: the profit gained (%).

ACKNOWLEDGMENT

This work was supported by HK RGC Earmarked Grant CUHK 484 / 95E, and Ho Sin-Hang Education Endowment Fund HSH 95 / 02.

REFERENCES

Bengio, Y. [1996] "Training a Neural Network with a Financial Criterion Rather than a Prediction Criterion," in *Neural Networks in the Capital Markets 1996*.

Bishop, C.M. [1995] *Neural Networks for Pattern Recognition*, Oxford University Press, New York, pp.65-72.

Caldwell, R. B. [1995] "Improved Prediction Performance Metrics for Neural Network-based Financial Forecasting Systems," *NEUROVEST JOURNAL*, Vol.3, No.5, pp. 22-26.

Cheung, Y.M., Helen, Z.H. Lai and L. Xu [1996] "Application of Adaptive RPCL-CLP with Trading Systems to Foreign Exchange Investment," in *Proc. of the International Conference on Neural Networks*, Vol. 1, pp. 131-136.

Jones, R.D., Lee, Y.C. and C.W. Barnes [1991] "Information Theoretic Derivation of Network Architecture and Learning Algorithms," in *Proc. of the International Joint Conference on Neural Networks*, Vol. 1, pp. 473-478.

Kang, J., Choey, M. and A.S. Weigend [1996] "Nonlinear Trading Models and Asset Allocation Based on Sharpe Ratio Optimization," in *Neural Networks in the Capital Markets 1996*.

Rumelhart, R.E., Hinton, G.E. and R.J. Williams [1986] "Learning Representations by Back-propagating Errors," *Nature* (London), 323, pp. 533-536.

Xu, L., Jordan, M.I. and G.E. Hinton [1995] "An Alternative Model for Mixtures of Experts," in *Advances in Neural Information Processing Systems* (J.D. Cowan, G. Tesauro and J. Alspector, Eds.), MIT Press, Cambridge, MA, pp.633--640.

Xu, L. [1997a] "RBF Networks, Mixture Experts, and Bayesian Ying-Yang Learning," *Neurocomputing* (forthcoming).

Xu, L. [1997b] "Bayesian Ying-Yang System and Theory as a Unified Statistical Learning Approach," *Proc. of the Intl. Workshop on Theoretical Aspects of Neural Computation*, Hong Kong (May 26-28, 1997), Springer-Verlag.

Lei Xu is a professor at the Chinese University of Hong Kong, the author of numerous papers on advanced technology and finance, and serves on the Editorial Board of the Journal of Computational Intelligence in Finance*. He can be reached at Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong. E-mail: lxu@cse.cuhk.edu.hk.*

Yiu-ming Cheung can be reached at the same postal address and by email at ymcheung@cse.cuhk.edu.hk.

See Appendix A on the following page.

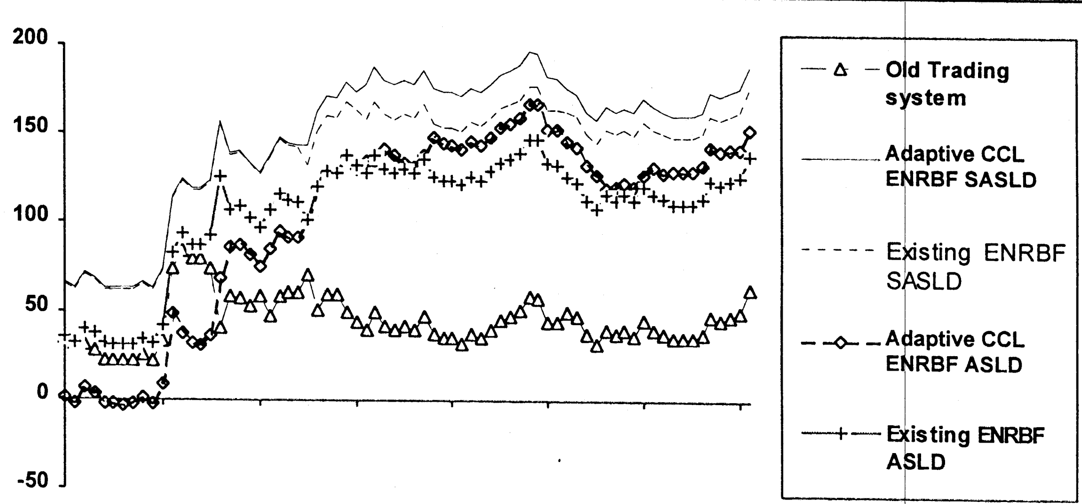

FIGURE 6. A plot of all results, along with results of an old prediction trading system by Cheung et al. [1996].

APPENDIX A

We denote $\Theta = \{m_j, \Sigma_j, w_j, c_j, \Pi_j\}^k_{j=1}$, where all the parameters are the same as given in eq.(8) in the text of the article, except Π_j which is the covariance matrix of the regression error $y_i - w^T_j x_i - c_j$. Also, below, R_{xy} is the cross correlation of x and y, Ey refers to the expectation of y, and $0<\gamma<1$ is the learning step size.

(a) Batch CCL-ENRBF Algorithm. Given the training set $\{x_i, y_i\}^N_{i=1}$, we have the following iterative procedure:

Step 1: Fix $\Theta = \Theta^{old}$, get $I(j|x_i)$ by

$$I(j|x_i) = \begin{cases} 1, & \text{if } j = \arg\min_r \{d_r(x_i) + \log|\Pi_r| + e_r^2(x_i)\} \\ 0, & \text{otherwise} \end{cases}$$

$$d_j(x_i) = (x_i - m_j)^T \Sigma_j^{-1}(x_i - m_j), \tag{1}$$

$$e_j^2(x_i) = (y_i - w_j^T x_i - c_j)^T \Pi_j^{-1}(y_i - w_j^T x_i - c_j)$$

Step 2(a): Update

$$\alpha_j^{new} = \frac{1}{N}\sum_{i=1}^N I(j|x_i)$$

$$m_j^{new} = \frac{1}{\alpha_j^{new} N}\sum_{i=1}^N I(j|x_i)x_i \tag{2}$$

$$\Sigma_j^{new} = \frac{1}{\alpha_j^{new} N}\sum_{i=1}^N I(j|x_i)(x_i - m_j^{new})(x_i - m_j^{new})^T$$

Step 2(b): Update

$$Ey_j = \frac{1}{\sum_{i=1}^N I(j|x_i)}\sum_{i=1}^N I(j|x_i)y_i$$

$$R_{xy} = \frac{1}{\sum_{i=1}^N I(j|x_i)}\sum_{i=1}^N I(j|x_i)(y_i - Ey_j)(x_i - m_j^{new})^T \tag{3}$$

$$w_j^{new} = \Sigma_j^{-1} R_{xy}, \quad c_j^{new} = Ey_j - (w_j^{new})^T m_j^{new}$$

$$\Pi_j^{new} = \frac{1}{\sum_{i=1}^N I(j|x_i)}\sum_{i=1}^N I(j|x_i)(y_i - (w_j^{new})^T x_i - c_j^{new})(y_i - (w_j^{new})^T x_i - c_j^{new})^T$$

Step 2(c): Let $\Theta^{new} = \Theta^{old}$.

(b) Adaptive CCL-ENRBF Algorithm. Given each pair $\{x_i, y_i\}$, go through the following steps once:

Step 1: Fix $\Theta = \Theta^{old}$, get $I(j|x_i)$ by eq.(1) and let $j^* = \arg\max_j I(j|x_i)$

Step 2(a): Update

$$\alpha_{j^*}^{new} = \frac{n_{j^*}}{\sum_{j=1}^k n_j}, \quad n_j \text{ is the number that } j = j^* \text{ in the past}$$

$$m_{j^*}^{new} = m_{j^*}^{old} + \gamma(x_i - m_{j^*}^{old}) \tag{4}$$

$$\Sigma_{j^*}^{new} = (1-\gamma)\Sigma_{j^*}^{old} + \gamma(x_i - m_{j^*}^{old})(x_i - m_{j^*}^{old})^T$$

Step 2(b): Update

$$Ey_{j^*}^{new} = Ey_{j^*}^{old} + \gamma(y_i - Ey_{j^*}^{old})$$

$$c_{j^*}^{new} = Ey_{j^*}^{new} - (w_{j^*}^{old})^T m_{j^*}^{old} \tag{5}$$

$$\Pi_{j^*}^{new} = (1-\gamma)\Pi_{j^*}^{old} + \gamma(y_i - (w_{j^*}^{old})^T x_i - c_{j^*}^{new})(y_i - (w_{j^*}^{old})^T x_i - c_{j^*}^{new})^T$$

$$w_{j^*}^{new} = (w_{j^*}^{old}) + \gamma(x_i - (w_{j^*}^{new})^T x_i - c_{j^*}^{new})x_i^T$$

Step 2(c): Let $\Theta^{new} = \Theta^{old}$.