New Approaches for financial prediction, portfolio management, and market modeling

Details references are given within slides and can be obtained from the following WWW site. Please cite accordingly whenever you make work basing on them.

Lei Xu

http://www.cse.cuhk.edu.hk/~lxu/

Department of Computer Science and Engineering,
The Chinese University of Hong Kong

---

1. Financial Prediction

- RPCL competitive learning based piecewise linear prediction
- Extended radial basis functions, Mixture of expert model and financial prediction
- Finite mixture of ARCH and GRACH models for prediction

2. Portfolio Management

- **Portfolio Management by Learned Decisions**
- **Markowitz Portfolio, Sharpe's ratio and Downside risk**
- **Improved Portfolio Sharpe Ratio Maximization with Diversification**
- **Adaptive Portfolio Management based on Extended RBF nets and analyses of market factors**

---

3. Arbitrage Pricing Theory

- Capital Asset Pricing Model vs. Arbitrage Pricing Theory
- Three Types of APT Implementation and Incapability of Factor analysis
- Temporal Factor Analysis (TFA) and APT
- TFA based APT for Prediction and Portfolio Management

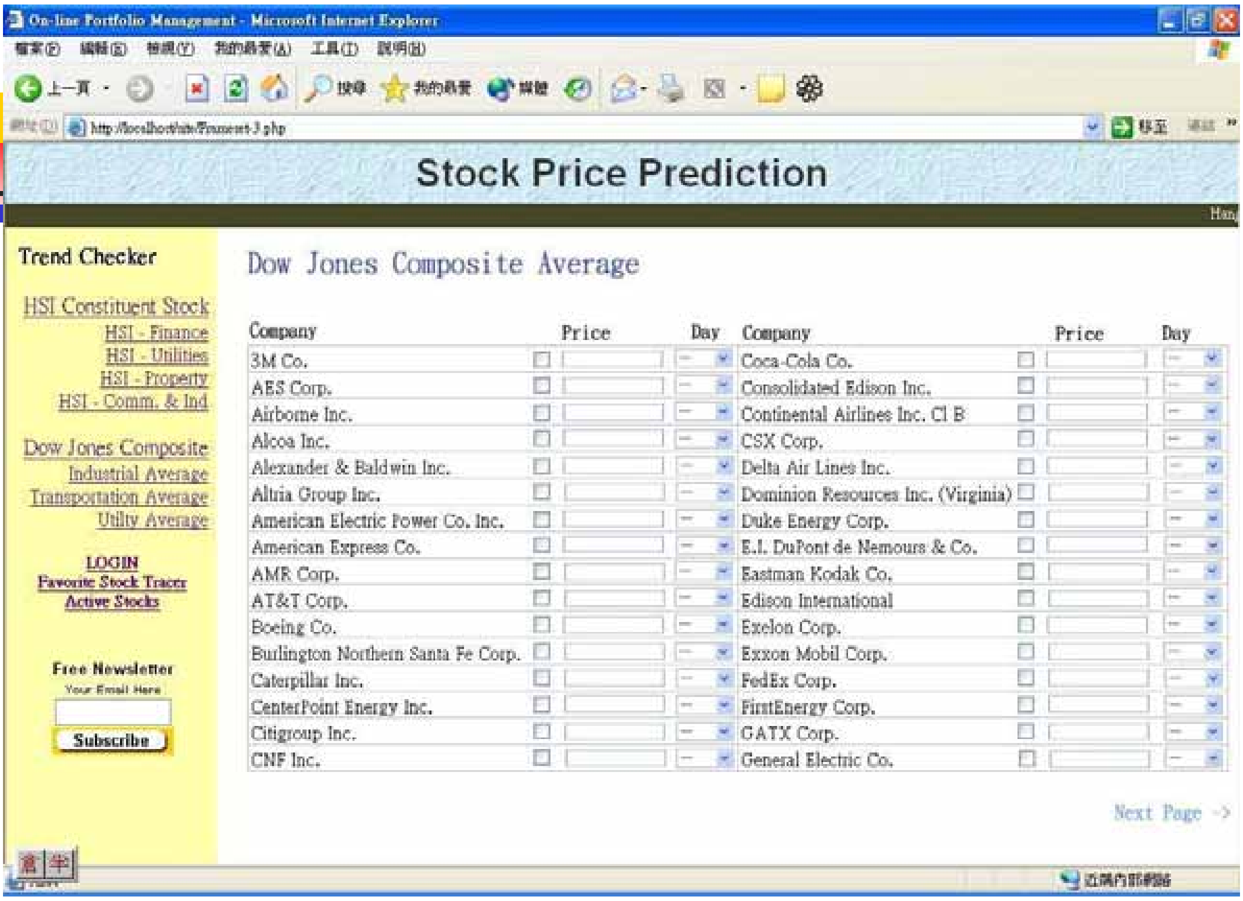
4. Challenges and Advances of Statistical Learning

- Two types of Intelligent Ability:
  Learning from Samples
- Key Ingredients of Statistical Learning
- Two Key Challenges and Advances on Seeking Solutions
- A Unified Theory:
  Bayesian Ying-Yang Harmony Learning

---

1. Financial Prediction

- RPCL competitive learning based piecewise linear prediction
- Extended radial basis functions, Mixture of expert model and financial prediction
- Finite mixture of ARCH and GRACH models for prediction
- APT-TFA based prediction



Stock Price Prediction

Trend Checker

HSI Constituent Stock
HSI - Finance
HSI - Utilities
HSI - Property
HSI - Comm. & Ind

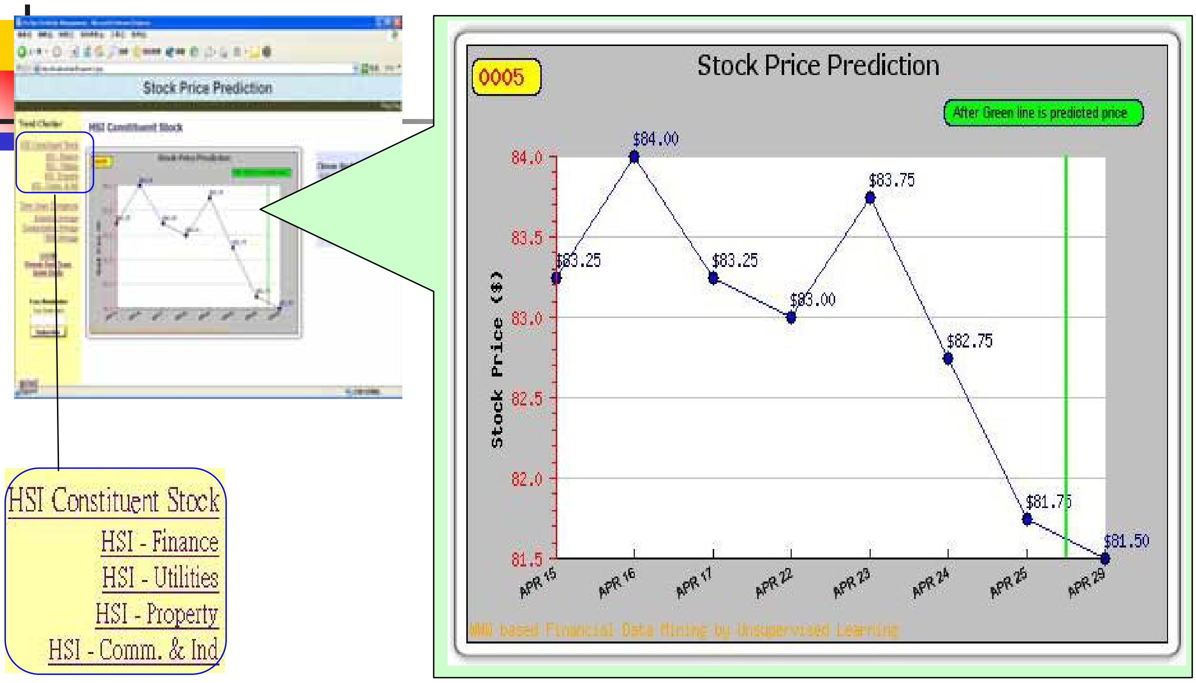
Dow Jones Composite
Industrial Average
Transportation Average
Utility Average

LOGIN
Favorite Stock Tracer
Active Stocks

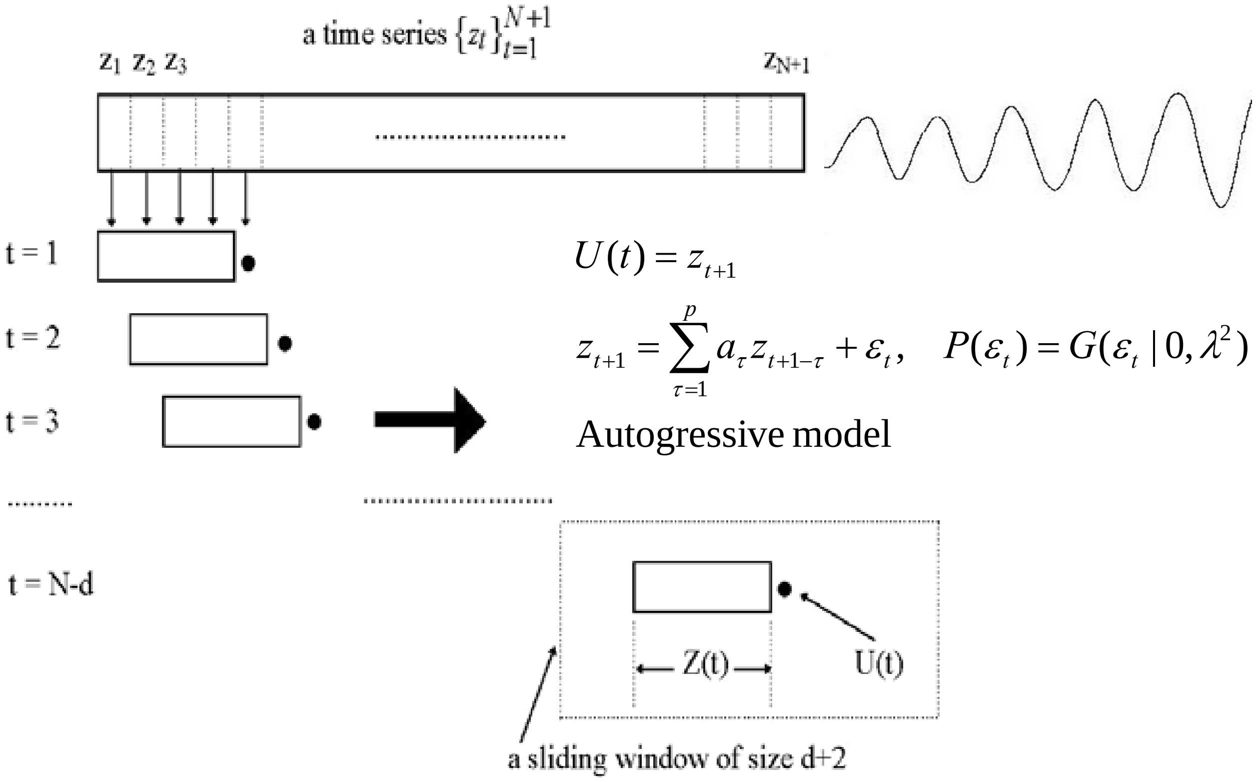
Free Newsletter
Your Email Here

Dow Jones Composite Average

Company		Price	Day	Company		Price	Day
3M Co.	☐		--	Coca-Cola Co.	☐		--
AES Corp.	☐		--	Consolidated Edison Inc.	☐		--
Airborne Inc.	☐		--	Continental Airlines Inc. Cl B	☐		--
Alcoa Inc.	☐		--	CSX Corp.	☐		--
Alexander & Baldwin Inc.	☐		--	Delta Air Lines Inc.	☐		--
Altria Group Inc.	☐		--	Dominion Resources Inc. (Virginia)	☐		--
American Electric Power Co. Inc.	☐		--	Duke Energy Corp.	☐		--
American Express Co.	☐		--	E.I. DuPont de Nemours & Co.	☐		--
AMR Corp.	☐		--	Eastman Kodak Co.	☐		--
AT&T Corp.	☐		--	Edison International	☐		--
Boeing Co.	☐		--	Exelon Corp.	☐		--
Burlington Northern Santa Fe Corp.	☐		--	Exxon Mobil Corp.	☐		--
Caterpillar Inc.	☐		--	FedEx Corp.	☐		--
CenterPoint Energy Inc.	☐		--	FirstEnergy Corp.	☐		--
Citigroup Inc.	☐		--	GATX Corp.	☐		--
CNF Inc.	☐		--	General Electric Co.	☐		--



The interface of HK stock analyzer and the graph shows the predicted stock price on 29th April and the historical stock price of HSBC Holding from 15th April to 25th April.



$$U(t) = z_{t+1}$$

$$z_{t+1} = \sum_{\tau=1}^{p} a_\tau z_{t+1-\tau} + \varepsilon_t, \quad P(\varepsilon_t) = G(\varepsilon_t \,|\, 0, \lambda^2)$$

Autogressive model

Autogressive model $z_{t+1} = \sum_{\tau=1}^{p} a_\tau z_{t+1-\tau} + \varepsilon_t, \quad P(\varepsilon_t) = G(\varepsilon_t \,|\, 0, \lambda^2)$

$$\begin{array}{llllll} z_1 & z_2 & \cdots & z_{q_1} & y_1 & y_1 = z_{q_1+1} \\ z_2 & z_3 & \cdots & z_{q_1+1} & y_2 & y_2 = z_{q_1+2} \\ \vdots & \vdots & & \vdots & & \vdots \\ z_N & z_{N+1} & \cdots & z_{N+q_1} & y_N & y_N = z_{q_1+N} \end{array}$$

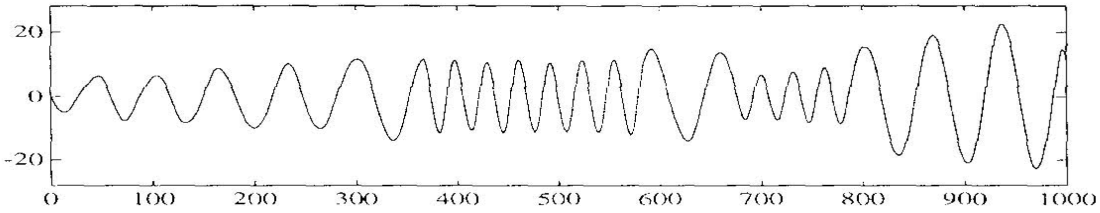
$$\mathbf{x}_t = [z_t \; z_{t+1} \; \cdots \; z_{t+q_1}]^{\mathrm{t}}$$

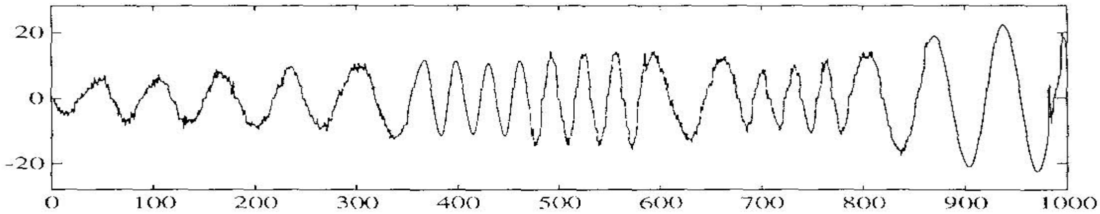
$$\mathbf{a} = [a_1 \cdots a_{q_1}]^{\mathrm{t}}$$

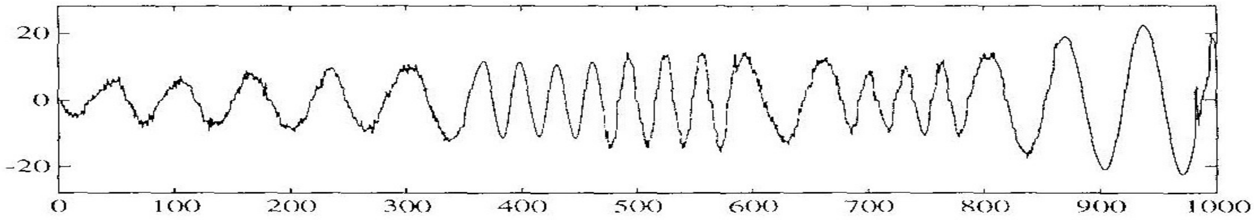
$$\mathbf{a}^t \mathbf{x}_t - y_t = \varepsilon \text{ or } y_t = \mathbf{a}^t \mathbf{x}_t + \varepsilon$$

$$P(\varepsilon) = G(\varepsilon \,|\, 0, \lambda^2)$$

$$\min E\|\varepsilon\|^2$$







$$\mathbf{x}_t = [z_t\ z_{t+1}\ \cdots\ z_{t+q_1}]^{\mathrm{t}} \qquad \mathbf{x}_t = [z_t\ z_{t+1}\ \cdots\ z_{t+q_2}]^{\mathrm{t}} \qquad \mathbf{x}_t = [z_t\ z_{t+1}\ \cdots\ z_{t+q_k}]^{\mathrm{t}}$$

$$\mathbf{a}_1 = [a_{11} \cdots\ a_{1q_1}]^{\mathrm{t}} \qquad \mathbf{a}_2 = [a_{21} \cdots\ a_{2q_2}]^{\mathrm{t}} \qquad \mathbf{a}_k = [a_{k1} \cdots\ a_{kq_k}]^{\mathrm{t}}$$

$$y_t = \mathbf{a}_1^t \mathbf{x}_t + c_1 + \varepsilon_1 \qquad y_t = \mathbf{a}_2^t \mathbf{x}_t + c_2 + \varepsilon_2 \qquad y_t = \mathbf{a}_k^t \mathbf{x}_t + c_k + \varepsilon_k$$

$$z_{t+1} = \sum_{\tau=1}^{q_j} a_{j.\tau} z_{t+1-\tau} + c_j + \varepsilon_j, \quad P(\varepsilon_j) = G(\varepsilon_j \mid 0, \lambda_j^2), \ \ j = 1, \cdots, k$$

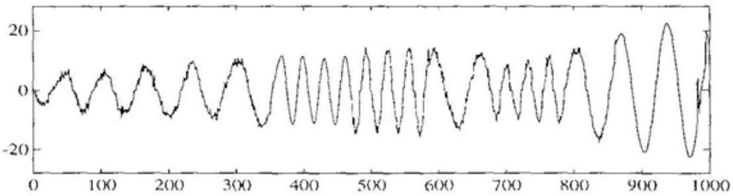
Maximum Learning on $P(\varepsilon_t \mid \theta) = \sum_{j=1}^{k} \alpha_j G(\varepsilon_t \mid 0, \lambda_j^2),$

$$\theta = \{\alpha_j, \lambda_j^2, a_{j,\tau}, \tau = 1, \cdots, q_j\}_{j=1}^{k}$$

Implemented by the EM algorithm (Xu, 1995, Proc. of IEEE NNSP-1995, USA)

Lei Xu(1995), ``Channel Equalization by Finite Mixtures and The EM Algorithm", *Proc. of IEEE Neural Networks and Signal Processing 1995 Workshop*, Vol.5, pp603-612, Aug. 31 – Sep. 2, 1995, Cambridge, Massachusetts, USA.

Another implementation: from Clustering to Gaussian mixture



$$\mathbf{x}_t = [z_t \ z_{t+1} \ \cdots \ z_{t+q}]^{\mathrm{t}}$$

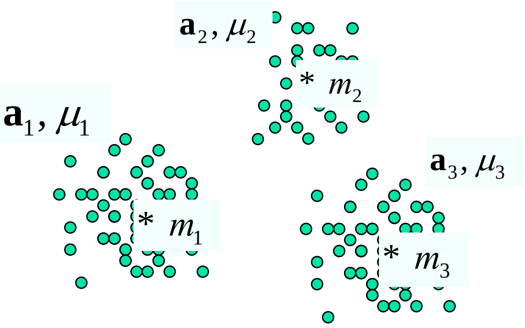
$$\mathbf{a}_1 = [a_{11} \cdots a_{1q}]^{\mathrm{t}} \qquad \mathbf{a}_2 = [a_{21} \cdots a_{2q}]^{\mathrm{t}} \qquad \mathbf{a}_k = [a_{k1} \cdots a_{kq}]^{\mathrm{t}}$$

$$y_t = \mathbf{a}_1^t \mathbf{x}_t + c_1 + \varepsilon_1 \qquad y_t = \mathbf{a}_2^t \mathbf{x}_t + c_2 + \varepsilon_2 \qquad y_t = \mathbf{a}_k^t \mathbf{x}_t + c_k + \varepsilon_k$$

$$y_t - \mu_1 = \mathbf{a}_1^t (\mathbf{x}_t - m_1) + \varepsilon_1 \qquad y_t - \mu_2 = \mathbf{a}_2^t (\mathbf{x}_t - m_2) + \varepsilon_2 \qquad y_t - \mu_k = \mathbf{a}_k^t (\mathbf{x}_t - m_k) + \varepsilon_k$$

$$c_j = \mu_j - \mathbf{a}_j^t m_j,$$
$$P(\varepsilon_t) = G(\varepsilon_t \mid 0, \lambda_j^2),$$
$$j = 1, \cdots, k$$

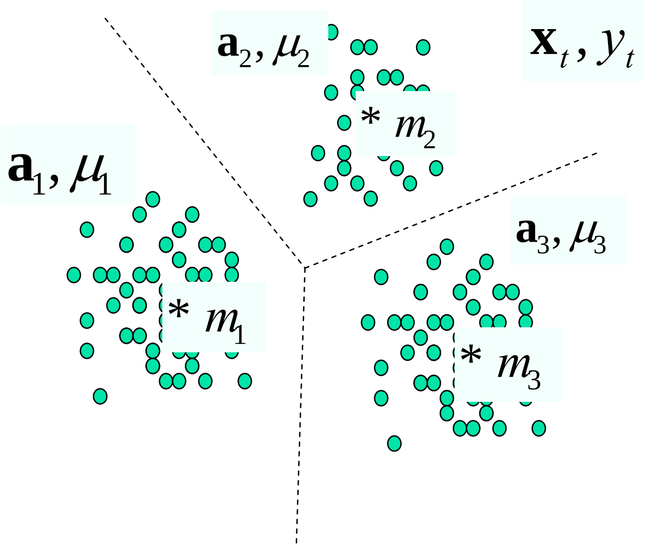
$$P(\varepsilon_t \mid \theta) = \sum_{j=1}^{k} \alpha_j G(\varepsilon_t \mid 0, \lambda_j^2)$$



Phase I:

When m_1, m_2, m_3 are given, we can get a_j, μ_j by the least square regression, i.e.

$$\min \sum_t \left\| \varepsilon_t \right\|^2, \ \varepsilon_t = y_t - \mu_2 - \mathbf{a}_2^t (\mathbf{x}_t - m_2)$$



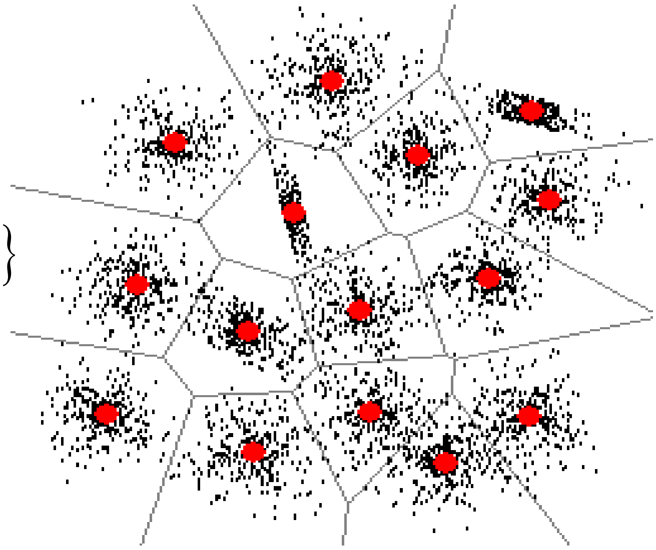
Phase II :
solve m_1, m_2, m_3

Mean Square Error (MSE) Clustering

Voronoi division

$$\min \quad \varepsilon(m) = \frac{1}{2}\sum_{k=1}^{M}\sum_{x_i \in S_k}\left\| x_i - m_k \right\|^2$$

$$S_k = \left\{ x_i : \left\| x_i - m_k \right\| < \left\| x_i - m_j \right\|, j \neq k \right\}$$

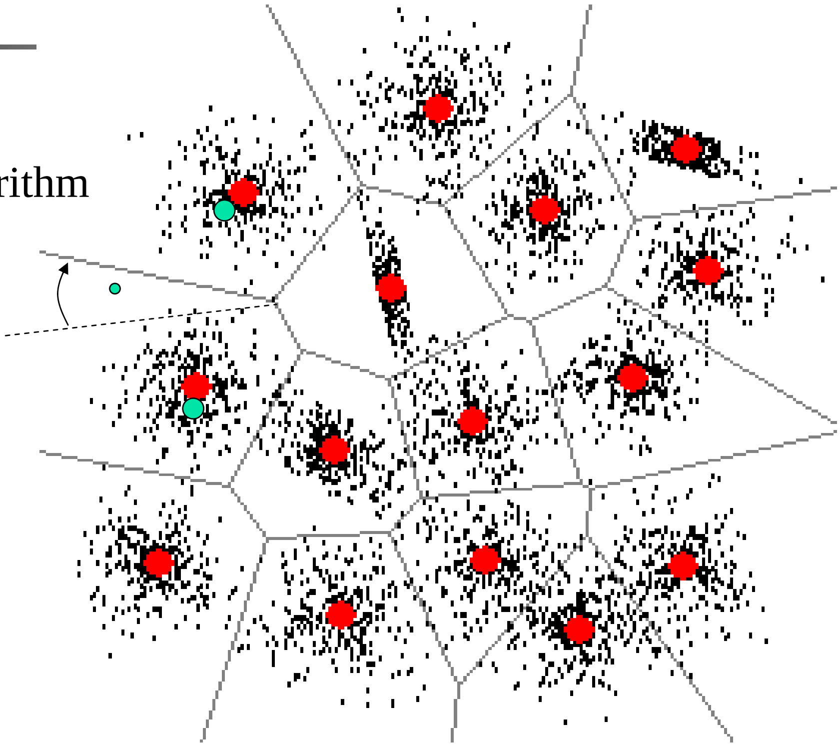


$$\min \quad \varepsilon(m) = \frac{1}{2}\sum_{k=1}^{M}\sum_{x_i \in S_k}\left\| x_i - m_k \right\|^2$$

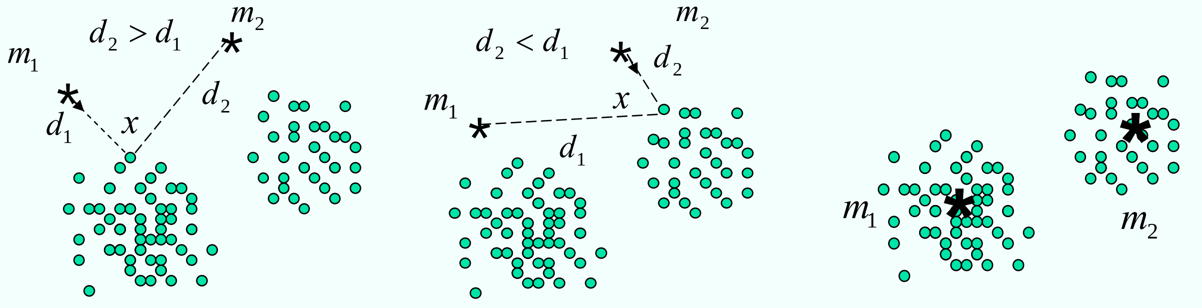
$$S_k = \left\{ x_i : \left\| x_i - m_k \right\| < \left\| x_i - m_j \right\|, j \neq k \right\}$$

K-means algorithm

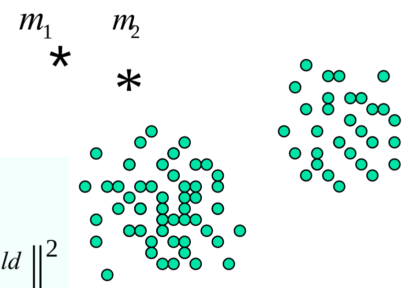
Local search



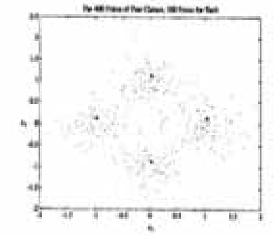
Competitive Learning

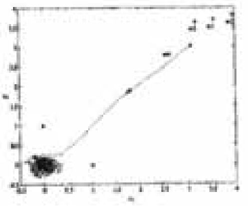


(a) m_1 is the winner (b) m_2 is the winner (c) m_1 and m_2 are converged



(a) initialization

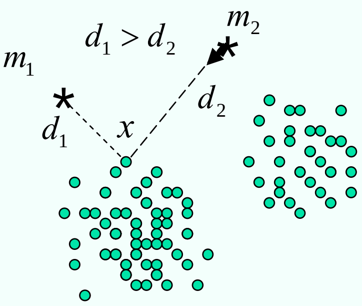




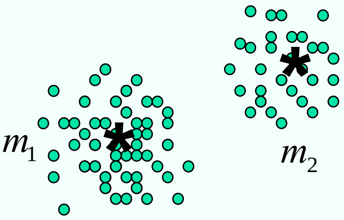
$$m_c^{new} = m_c^{old} + \eta(\mathbf{x}_t - m_c^{old})$$

$$c = \arg\min_j d_j, \quad d_j = \left\|\mathbf{x}_t - m_j^{old}\right\|^2$$

Frequency Sensitive Competitive Learning



(a) m_1 is the winner



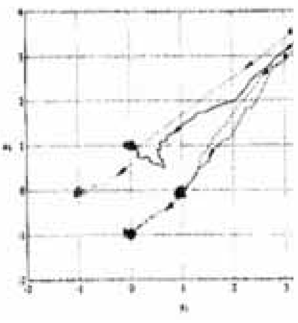
(c) m_1 and m_2 are converged

$$m_c^{new} = m_c^{old} + \eta(\mathbf{x}_t - m_c^{old})$$

$$c = \arg\min_j d_j, \quad d_j = f_j \left\|\mathbf{x}_t - m_j^{old}\right\|^2$$

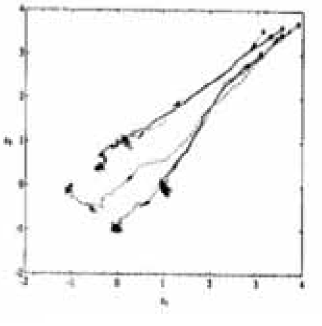
f_j frequency m_j wins

works

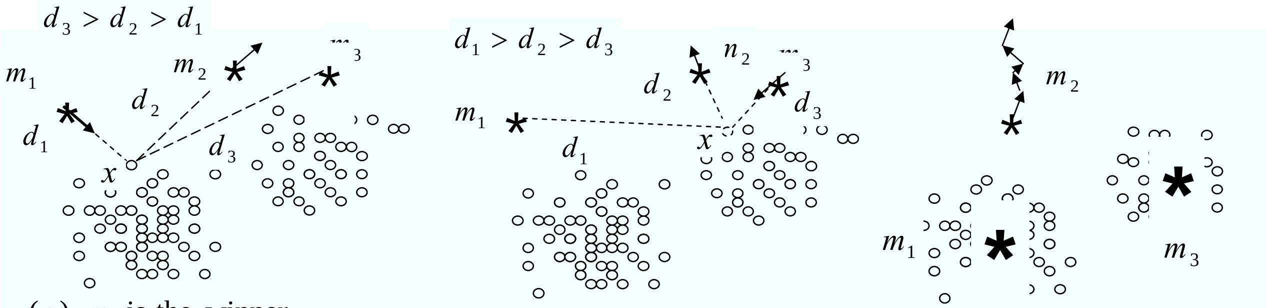


still a problem

How to decide k ?



Rival Penalized Competitive Learning (RPCL)



(a) m_1 is the winner
m_2 is the rival

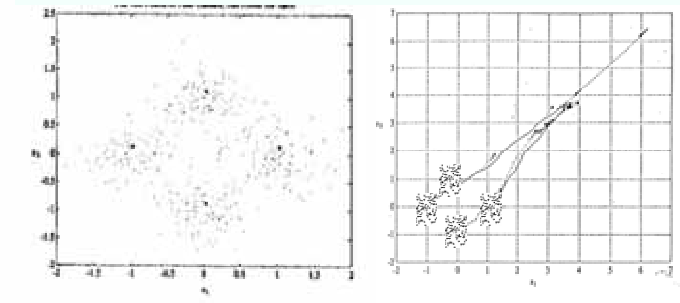
(a) m_3 is the winner
m_2 is the rival

(c) m_1 and m_3 are converged
m_2 is driven far away

$$m_c^{new} = m_c^{old} + \eta_c(\mathbf{x}_t - m_c^{old}),\ \ c = \arg\min_j d_j$$

$$m_r^{new} = m_r^{old} - \eta_r(\mathbf{x}_t - m_r^{old}),\ \ r = \arg\min_{j\neq c} d_j$$

k is determined automatically during learning



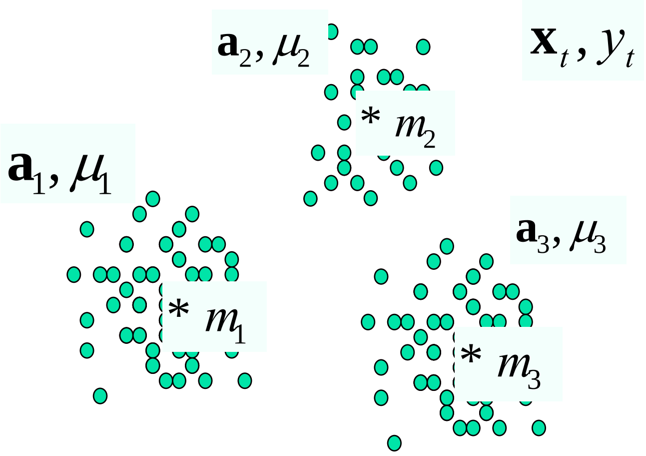
See Xu, Oja & Krzyzak, Proc. ICPR02, then
Xu, L., A. Krzyzak, and E.Oja, IEEE Tr. on Neural Networks, Vol.4, No.4, 1993, pp636-649.
（SCI紀錄之被引用量为72+5=77）
（Google Scholar 紀錄之被引用量为 124）

Lei Xu, A. Krzyzak & E.Oja, (1993), "Rival Penalized Competitive Learning for Clustering Analysis, RBF net and Curve Detection", *IEEE Trans. on Neural Networks*, Vol.4, No.4, pp636-649, 1993.

Phase III :

When a_j, μ_j, m_j are given, we can predict

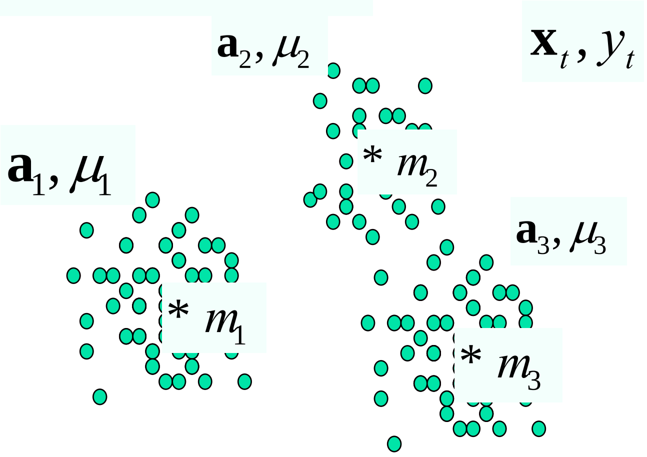
$$\hat{y}_t = \mu_{\mathrm{c}} + \mathbf{a}_c^t(\mathbf{x}_t - m_c),\ c = \arg\min_{\mathrm{j}} \left\| \mathbf{x}_t - m_j \right\|^2$$

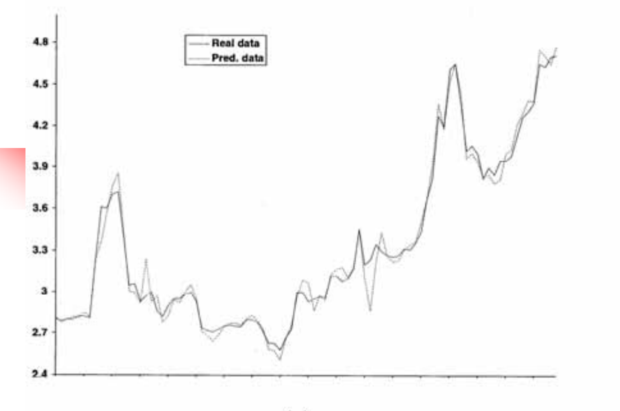


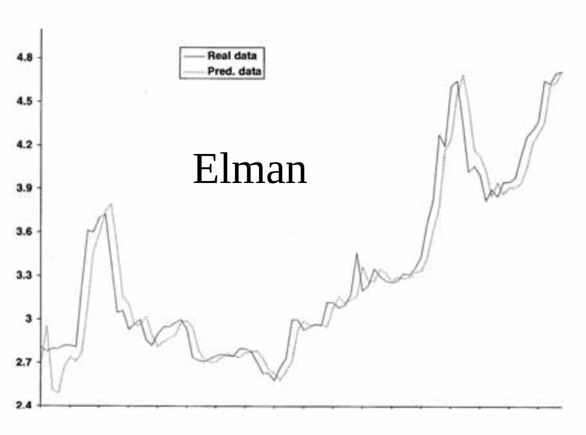
When a_j, μ_j, m_j are given, we can predict

$$\hat{y}_t = \sum_{j=1}^{k} p(j|\mathbf{x}_t)[\mu_{\mathrm{j}} + \mathbf{a}_j^t(\mathbf{x}_t - m_j)]$$

$$p(j|\mathbf{x}_t) = \frac{G(\mathbf{x}_t \mid m_j, \sigma_j^2)}{\sum_{j=1}^{k} G(\mathbf{x}_t \mid m_j, \sigma_j^2)}$$

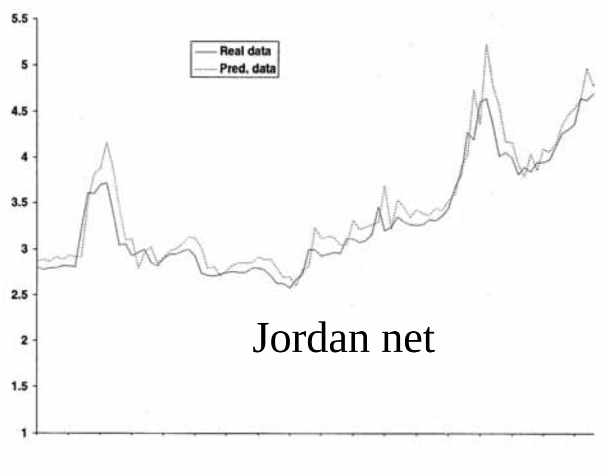






Elman

(b)



Jordan net

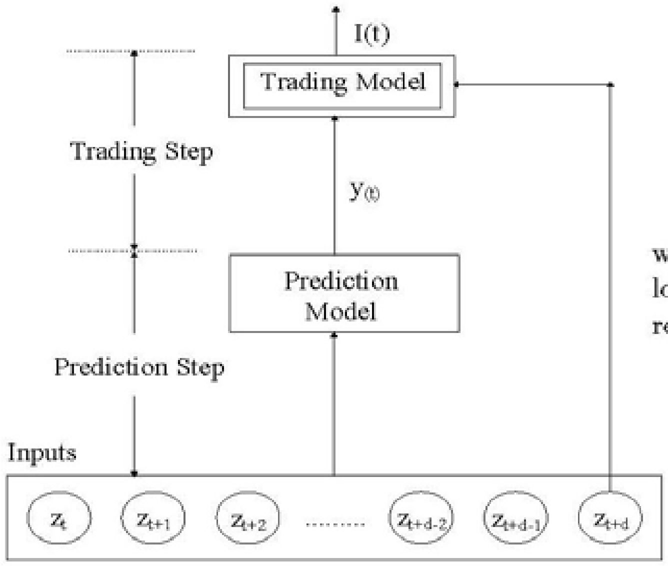


Fig. 5. Trading system with two steps: prediction step followed by trading step. In prediction step, the prediction model outputs $y_{(t)}$ — the estimation of desired output $U(t)$ when the input $Z(t) = [z_t, z_{t+1}, \ldots, z_{t+d-1}, z_{t+d}]$ is available. In trading step, the trading model will output a trading signal $I(t)$ based on $y_{(t)}$ and current information z_{t+d} for the next trading activity.

$$I(t) = \begin{cases} -1, & \text{if } \hat{z}_{t+1} - z_t > 0 \\ 0, & \text{if } \hat{z}_{t+1} - z_t = 0 \\ 1, & \text{otherwise} \end{cases} \qquad (27)$$

where $I(t) = 1$, 0 and -1 stand for the "buy long," "do nothing" and "buy short" signals respectively.

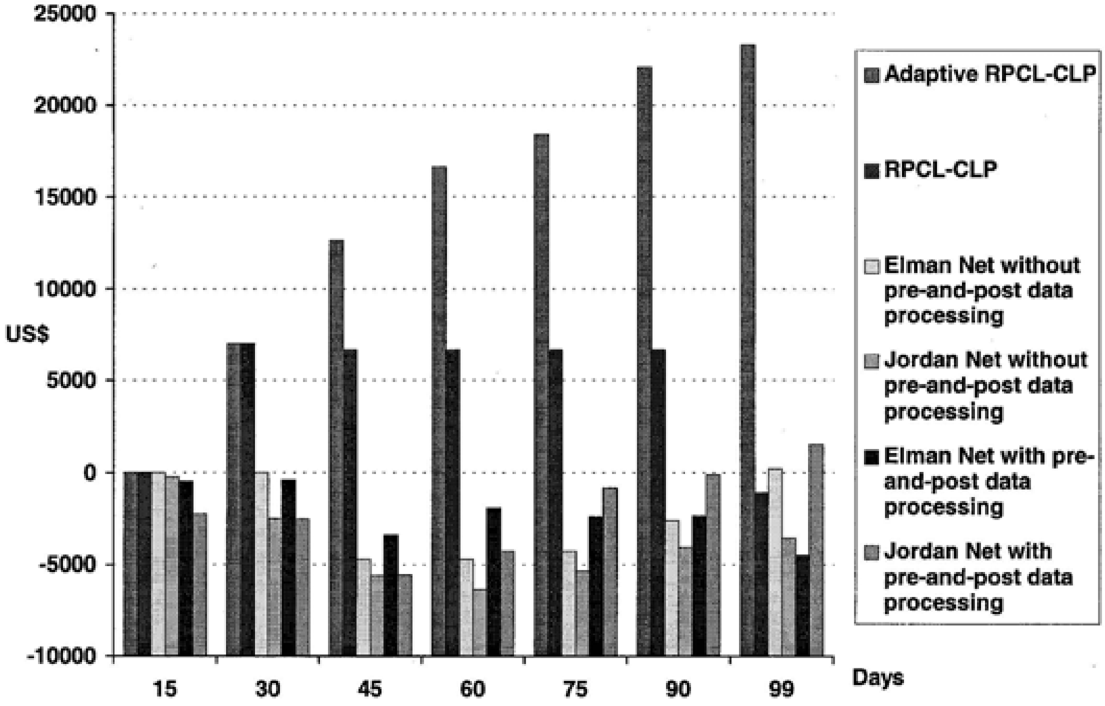
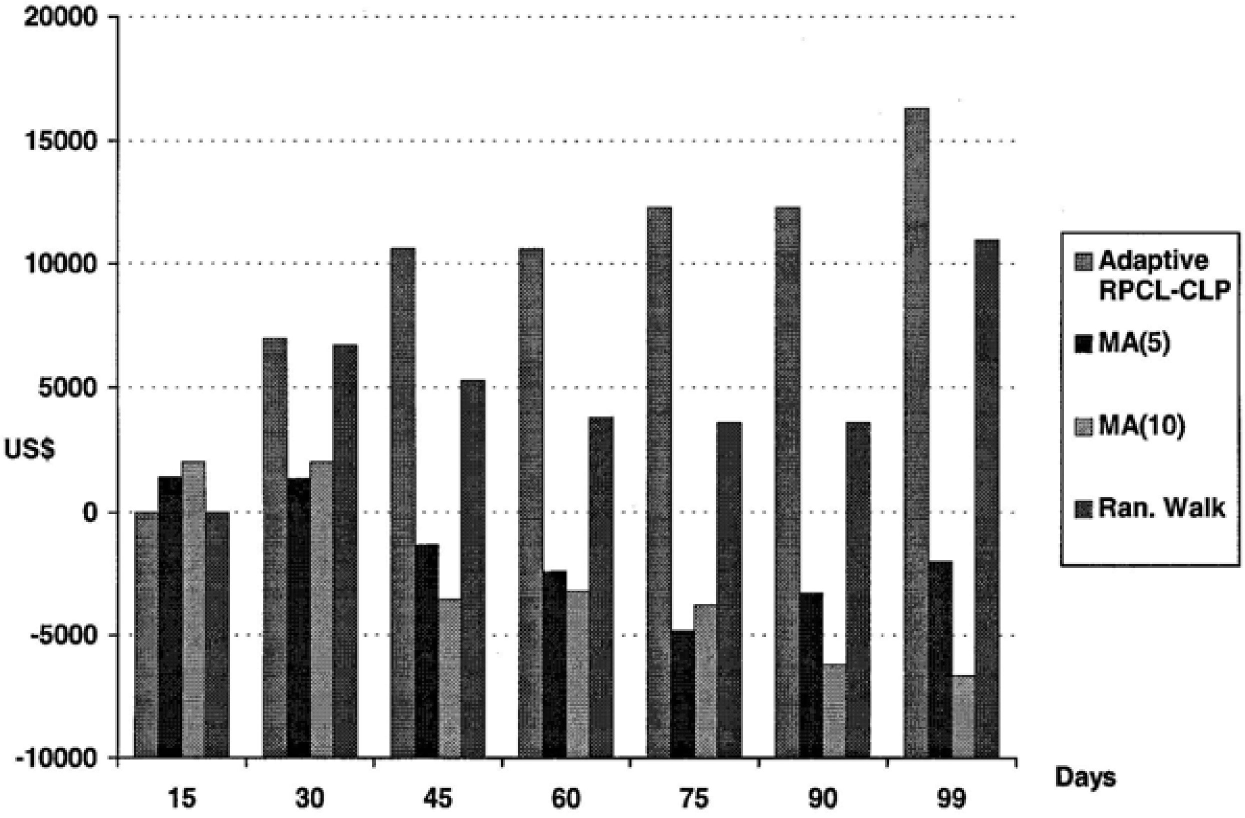
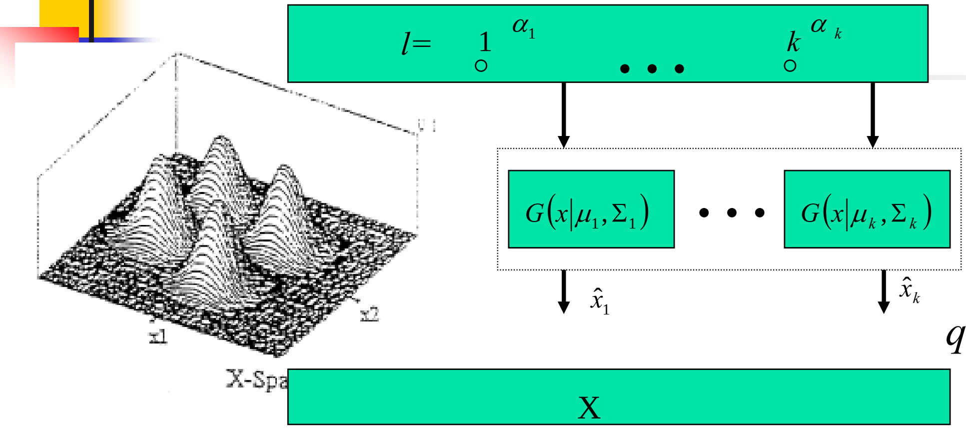


Fig. 12. Profits from different prediction models with trading strategy 1.



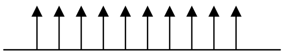
Gaussian Mixture

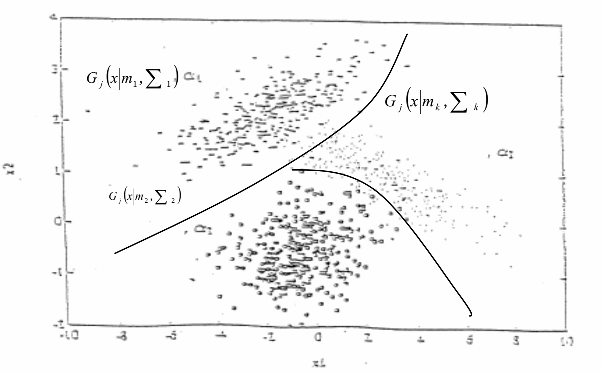
$$\alpha_j \ge 0, \quad \sum_{j=1}^{K} \alpha_j = 1$$



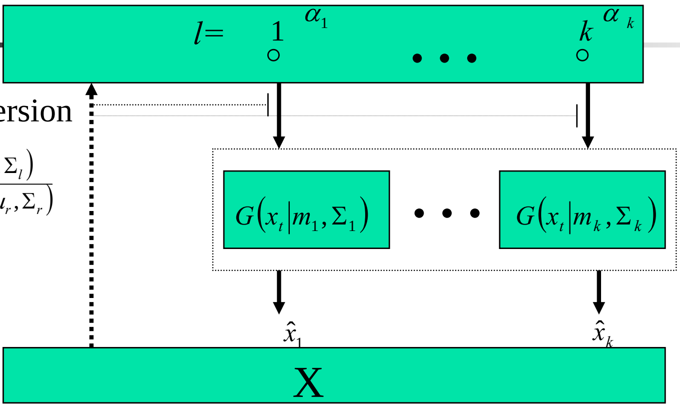
$$q(x)=\sum_{r=1}^{k}\alpha_r G(x|\mu_r,\Sigma_r)$$

$$p(x)=\frac{1}{N}\sum_{i=1}^{N}\delta(x-x_i)$$





The EM Algorithm



Bayesian Inversion

$$p(l|x_t)=\frac{\alpha_l G(x_t|\mu_l,\Sigma_l)}{\sum_r \alpha_r G(x_t|\mu_r,\Sigma_r)}$$

$$\Uparrow$$

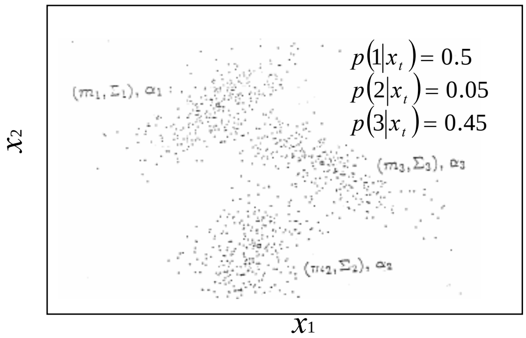
$p(l|x)$ is free

$$\alpha_l^{new} = \frac{1}{N}\sum_{t=1}^{N} p(l/x_t)$$

$$n_l^{new} = \alpha_l^{new} N$$

$$\mu_l^{new} = \frac{1}{n_l^{new}}\sum_{t=1}^{N} p(l/x_t)x_t$$

$$\Sigma_l^{new}=\frac{1}{n_l^{new}}\sum_{t=1}^{N}p(l/x_t)[x_t-\mu_l^{new}][x_t-\mu_l^{new}]^T$$



$p(1|x_t) = 0.5$
$p(2|x_t) = 0.05$
$p(3|x_t) = 0.45$

Comparison of EM Algorithm And Gradient Approach

- For Gaussian mixtures, connection between EM and gradient is set up

$$\Theta^{(k+1)} = \Theta^{(k)} + P\left(\Theta^{(k)}\right)\frac{\partial L}{\partial \Theta^{(k)}}$$

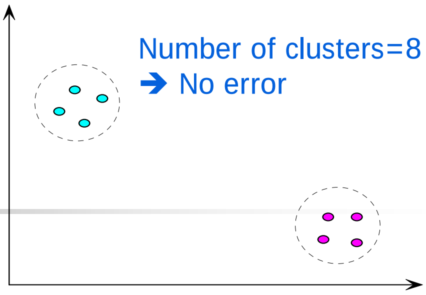
EM searches in a positive projection of $\frac{\partial L}{\partial \Theta^{(k)}}$

- EM converges from any initial conditions
  Automatic satisfaction of Constraints

- $P\left(\Theta^{(k)}\right)$ varies adaptively : Quasi - Newton Speed.

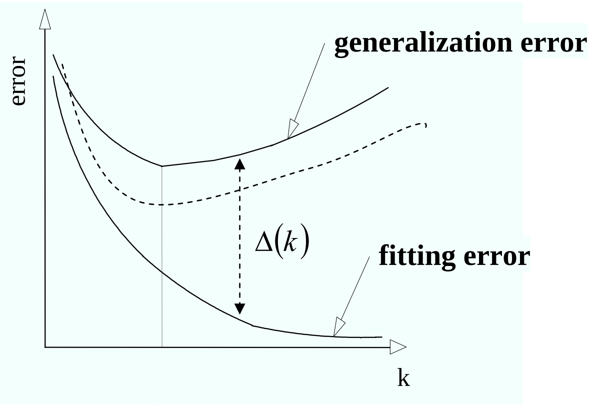
clarified a wide spreading misunderstanding (Xu & Jordan, MIT AI Memo, 1992, then Xu, L., and M.I.Jordan, “On Convergence Properties of The EM Algorithm for Gaussian Mixtures”, Neural Computation, Vol. 8, No.1, 1996, pp.129-151).
（**SCI**紀錄之被引用量为**52**）（**Google Scholar** 紀錄之被引用量为**131**）

A finite size N of samples


Number of clusters=8
➔ No error

Existing Efforts

- VC Dimension based SRM
- AIC
- BIC, SIC
- Cross Validation
- MML/MDL
- Bayesian Approach



$$k^* = \arg\min_k \left[\Delta(k) + F\left(p\left(x \mid \theta(k)\right), \mathrm{X}\right)\right]$$

The existing efforts usually lead to a rough estimate $\Delta(k)$

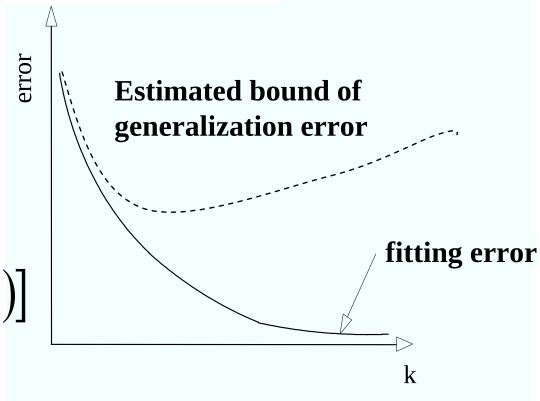
Two Steps of Solving

Step 1 Enumerate k for a set of candidate values, fixed at each candidate, make learning

$$\theta^*(k) = \arg\min_\theta F\left(p(x\,|\,\theta), \mathrm{X}\right)$$

Step 2 Select the best one k^* by

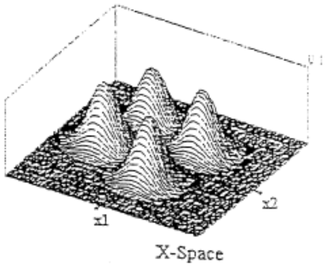
$$k^* = \arg\min_k \left[\Delta(k) + F\left(p(x\,|\,\theta(k)), \mathrm{X}\right)\right]$$



Very computational extensive !!!

Gaussian mixture and clustering analyses

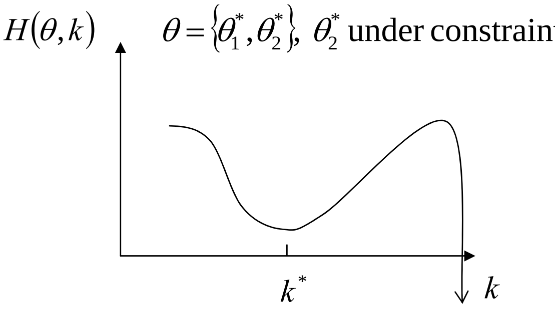
(Studied as a typical special case of BYY harmony learning)

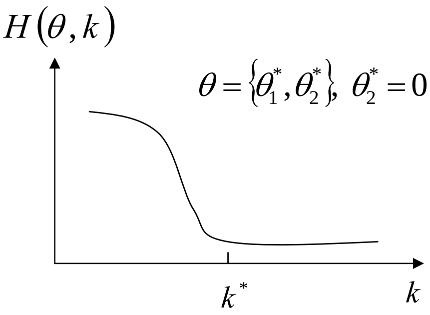


- J(k) for k, especially on a small size of samples,
- The smoothed EM algorithm,

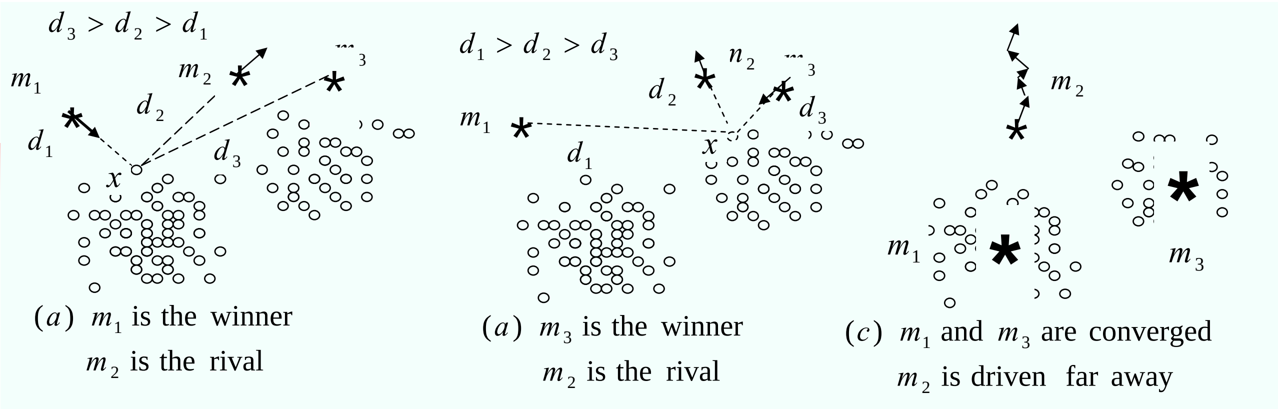
  (Xu, Pattern Recognition Letter, 1997; ICONIP97)
- Adaptive algorithms with k determined automatically during learning,
- Kernel density estimation via support vectors via BYY learning.

  (Xu, Int. J. Neural Systems, 2001; Neural Networks, 2002)





Gaussian Mixture based RPCL learning



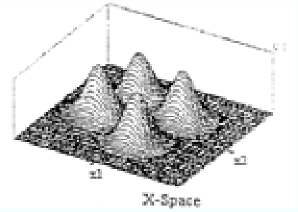
(a) m_1 is the winner
m_2 is the rival

(a) m_3 is the winner
m_2 is the rival

(c) m_1 and m_3 are converged
m_2 is driven far away

Gaussian cluster

$$d_l(\theta_l) = -\ln[\alpha_l G(x \mid m_l, \Sigma_l)]$$



$$\theta_c^{new} = \theta_c^{old} - \eta_c \frac{\partial d_c(\theta_c)}{\partial \theta_c}, \quad c = \arg\min_j d_j$$

$$\theta_r^{new} = \theta_r^{old} + \eta_r \frac{\partial d_r(\theta_r)}{\partial \theta_r}, \quad r = \arg\min_{j \neq c} d_j$$

$$m_c^{new} = m_c^{old} + \eta_c(\mathbf{x}_t - m_c^{old}), \quad c = \arg\min_j d_j$$

$$m_r^{new} = m_r^{old} - \eta_r(\mathbf{x}_t - m_r^{old}), \quad r = \arg\min_{j \neq c} d_j$$

$$\Sigma_j = S_j^T S_j, \quad \alpha_j = \gamma_j^2$$

L. Xu, IJCNN98, May 5-9, 1998, Alaska, Vol.II, pp2525-2530.
L. Xu, Intl J. Neural Systems, Vol.11, No.1, pp3-69, 2001.

Lei Xu (1997), ``Bayesian Ying-Yang Machine, Clustering and Number of Clusters'', *Pattern Recognition Letters, Vol.18, No.11-13*, pp1167-1178, 1997.

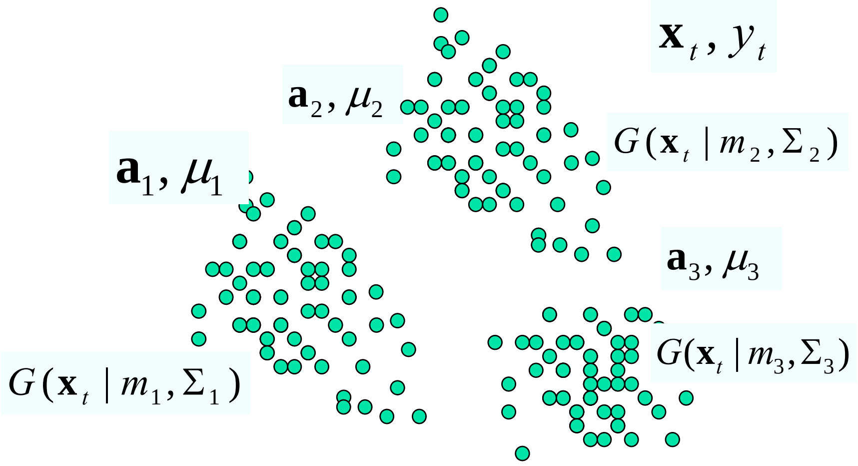
Lei Xu (1998) , ``Rival Penalized Competitive Learning, Finite Mixture, and Multisets Clustering'', *Proc. Intentional Joint Conference on Neural Networks*, Vol., May 5-9, 1998, Anchorage, Alaska.

Lei Xu (2001), ``Best Harmony, Unified RPCL and Automated Model Selection for Unsupervised and Supervised Learning on Gaussian Mixtures, ME-RBF Models and Three-Layer Nets '', *International Journal of Neural Systems, Vol.11, No.1*, pp3-69, 2001.

Lei Xu (2002), ``BYY harmony learning, structural RPCL, and topological self-organizing on mixture models'', *Neural Networks, Vol. 15*, pp1125-1151, 2002.

Then, we can get $\mathbf{a}_{\mathbf{j}}, \mu_j$ by the least square regression, i.e.

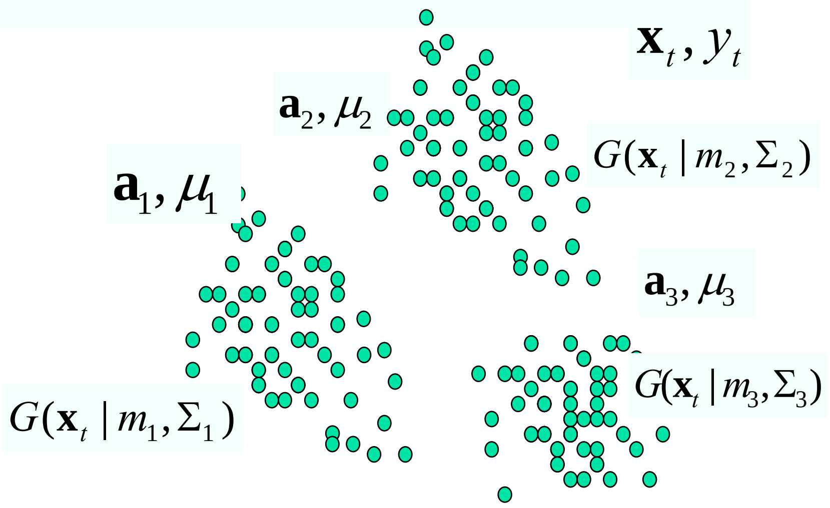
$$\min \sum_t \sum_{j=1}^{k} p(j|\mathbf{x}_t) \left\| y_t - \mu_{\mathrm{j}} - \mathbf{a}_j^t(\mathbf{x}_t - m_j) \right\|^2$$



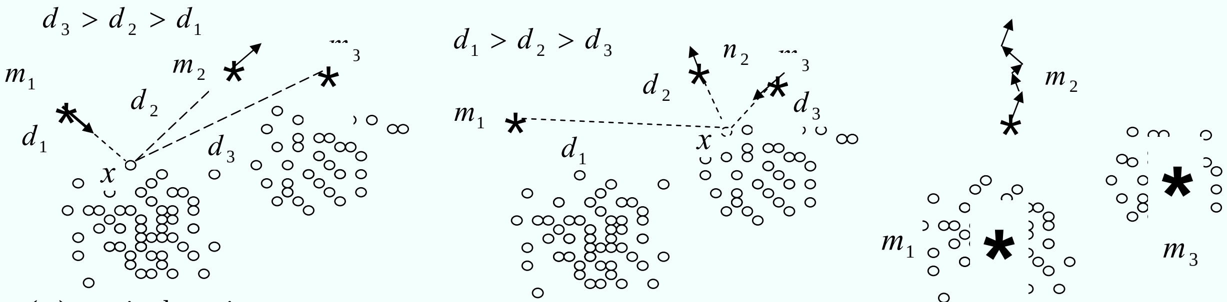
With $a_j, \mu_j, m_j, \Sigma_j$ obtained, we can predict

$$\hat{y}_t = \sum_{j=1}^{k} p(j|\mathbf{x}_t)[\mu_{\mathrm{j}} + \mathbf{a}_j^t(\mathbf{x}_t - m_j)]$$

$$p(j|\mathbf{x}_t) = \frac{\alpha_j G(\mathbf{x}_t \mid m_j, \Sigma_j)}{\sum_{j=1}^{k} \alpha_j G(\mathbf{x}_t \mid m_j, \Sigma_j)}$$



Prediction error based RPCL learning



(*a*) m_1 is the winner
m_2 is the rival

(*a*) m_3 is the winner
m_2 is the rival

(*c*) m_1 and m_3 are converged
m_2 is driven far away

$$d_l(\theta_l) = \left\| y_t - \mu_\mathrm{j} - \mathbf{a}_j^t(\mathbf{x}_t - m_j) \right\|^2$$

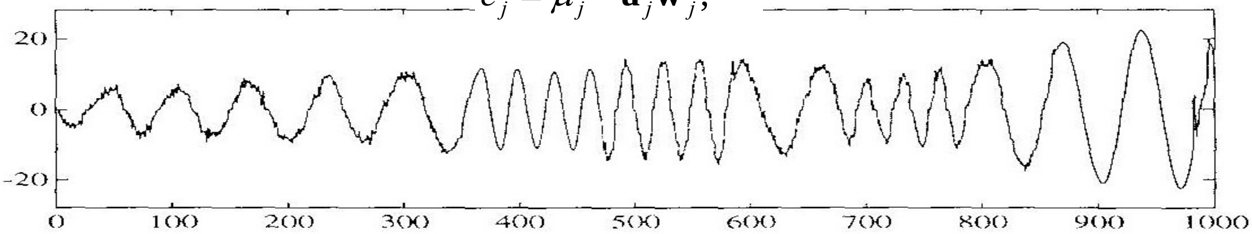
$$\theta_c^{new} = \theta_c^{old} - \eta_c \frac{\partial d_c(\theta_c)}{\partial \theta_c}, \quad c = \arg\min_j d_j$$

$$\theta_r^{new} = \theta_r^{old} + \eta_r \frac{\partial d_r(\theta_r)}{\partial \theta_r}, \quad r = \arg\min_{j \neq c} d_j$$

$$c_j = \mu_j - \mathbf{a}_j^t m_j,$$
$$P(\varepsilon_t) = G(\varepsilon_t \mid 0, \lambda_j^2),$$
$$j = 1, \cdots, k$$

Prediction error based Gaussian Mixture

$$c_j = \mu_j - \mathbf{a}_j^t \mathbf{w}_j,$$



$$\mathbf{x}_t = [z_t \; z_{t+1} \; \cdots \; z_{t+q_1}]^\mathrm{t} \qquad \mathbf{x}_t = [z_t \; z_{t+1} \; \cdots \; z_{t+q_2}]^\mathrm{t} \qquad \mathbf{x}_t = [z_t \; z_{t+1} \; \cdots \; z_{t+q_k}]^\mathrm{t}$$

$$\mathbf{a}_1 = [a_{11} \cdots \; a_{1q_1}]^\mathrm{t} \qquad \mathbf{a}_2 = [a_{21} \cdots \; a_{2q_2}]^\mathrm{t} \qquad \mathbf{a}_k = [a_{k1} \cdots \; a_{kq_k}]^\mathrm{t}$$

$$y_t = \mathbf{a}_1^t \mathbf{x}_t + c_1 + \varepsilon_1 \qquad y_t = \mathbf{a}_2^t \mathbf{x}_t + c_2 + \varepsilon_2 \qquad y_t = \mathbf{a}_k^t \mathbf{x}_t + c_k + \varepsilon_k$$

$$z_{t+1} = \sum_{\tau=1}^{q_j} a_{j.\tau} z_{t+1-\tau} + c_j + \varepsilon_j, \quad P(\varepsilon_j) = G(\varepsilon_j \mid 0, \lambda_j^2), \;\; j = 1, \cdots, k$$

$$\text{Maximum Learning on } P(\varepsilon_t \mid \theta) = \sum_{j=1}^{k} \alpha_j G(\varepsilon_t \mid 0, \lambda_j^2),$$

$$\theta = \{\alpha_j, \lambda_j^2, a_{j,\tau}, \tau = 1, \cdots, q_j\}_{j=1}^k$$

Lei Xu(1995), ``Channel Equalization by Finite Mixtures and The EM Algorithm",
Proc. of IEEE Neural Networks and Signal Processing 1995 Workshop, Vol.5, pp603-612,
Aug. 31 – Sep. 2, 1995, Cambridge, Massachusetts, USA.

E step :

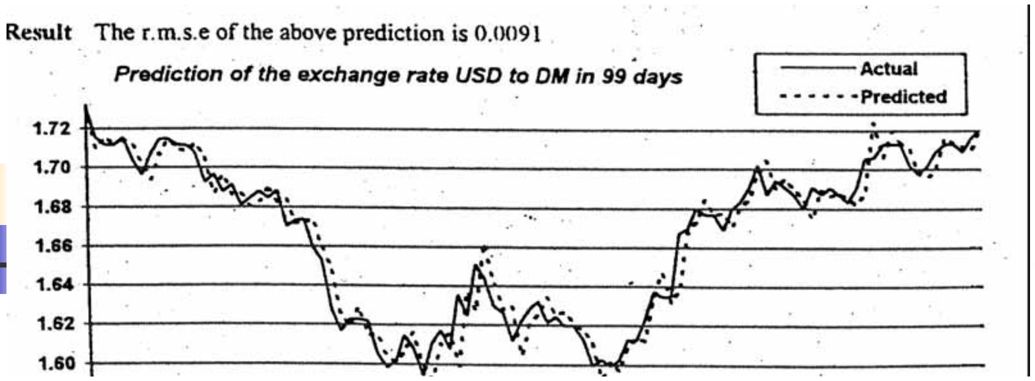
$$p(j|\ \varepsilon_t) = \frac{\alpha_j G(\varepsilon_t \,|\, 0, \lambda_j^2)}{\sum_{i=1}^{k} \alpha_j G(\varepsilon_t \,|\, 0, \lambda_j^2)}$$

M step :

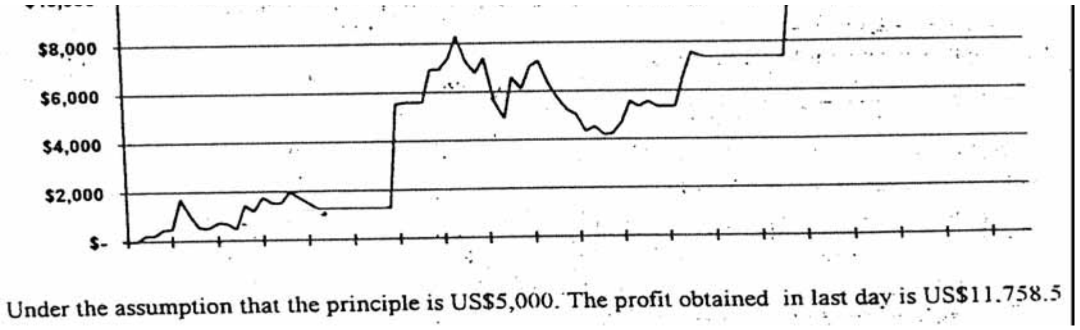
$$\max \sum_t \sum_{j=1}^{k} p(j\,|\varepsilon_t) \ln[\alpha_j G(\varepsilon_t \,|\, 0, \lambda_j^2)]$$

$$\max_{\alpha_j \ge 0, \lambda_j^2 \ge 0} \sum_t \sum_{j=1}^{k} p(j\,|\varepsilon_t) \ln[\alpha_j G(\varepsilon_t \,|\, 0, \lambda_j^2)]$$

$$\min_{\mu_j, \mathbf{a}_j, m_j} \sum_t \sum_{j=1}^{k} p(j\,|\varepsilon_t) \left\| y_t - \mu_j - \mathbf{a}_j^t (\mathbf{x}_t - m_j) \right\|^2$$



H. Y. Kwok, C. M. Chen, and Lei Xu, ``Comparison between Mixture of ARMA and Mixture of AR Model with Application to Time Series Forecasting", Proc. of International Conference on Neural Information Processing (ICONIP'98), October 21-23, 1998, Kitakyushu, Japan, Vol.2, pp1049-1052.



C.H.Wong, F. Yung, & L. Xu, Proc. of NCNN96, China

USD-SWF (40 experts used)

USD-DEM (40 experts used)

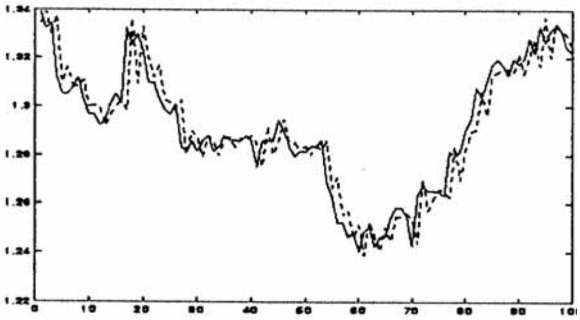


Fig. 9 Mixture of AR model (r.m.s.e.= 0.006265)

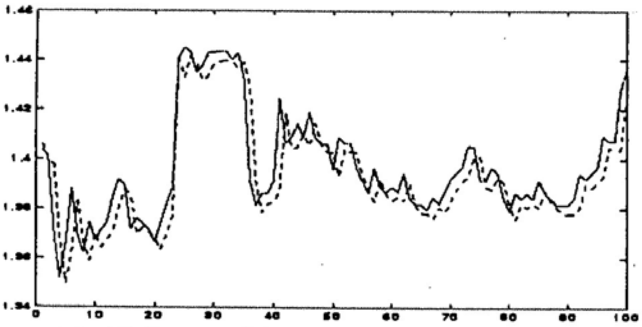


Fig.7 Mixture of AR model. (r.m.s.e. = 0.009321)

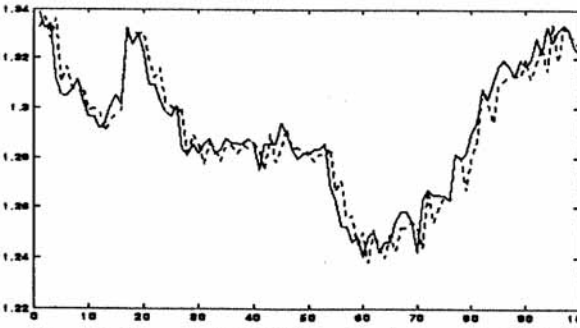


Fig.10 Mixture of ARMA model (r.m.s.e.=0.006932)

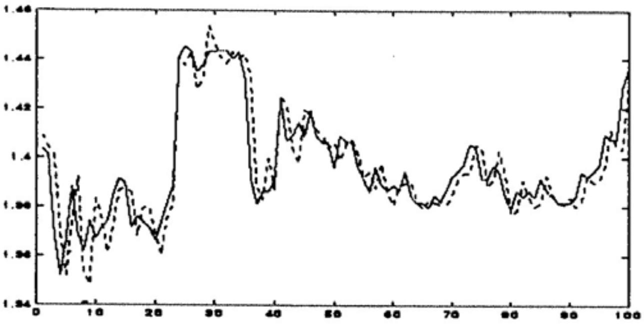
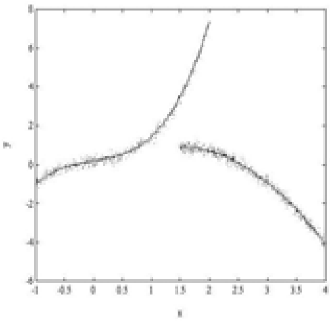


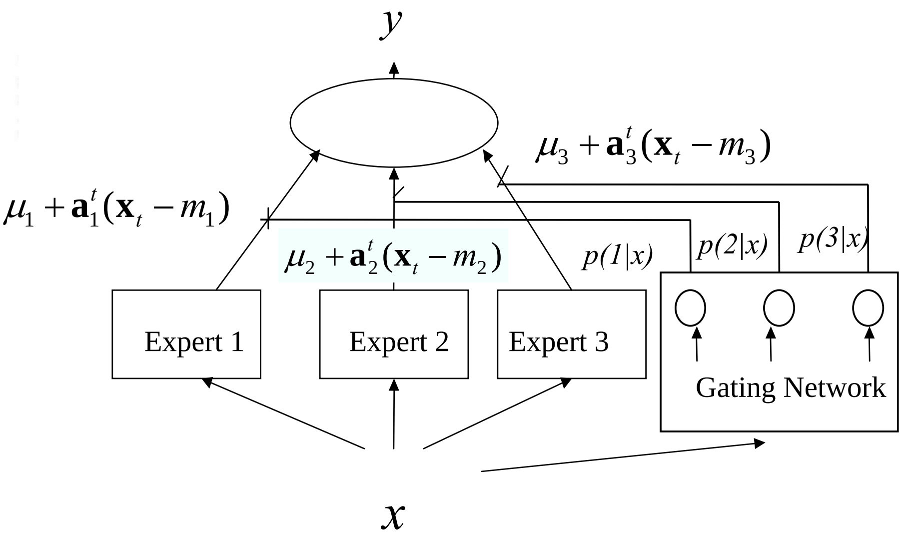
Fig. 8 Mixture of ARMA model.(r.m.s.e. =0.009946)

H.Y. Kowk, C.M.Chen, & L. Xu, Proc. ICONIP98

Summary

$$\hat{y}_t = \sum_{j=1}^{k} p(j|\mathbf{x}_t)[\mu_j + \mathbf{a}_j^t(\mathbf{x}_t - m_j)], \quad \hat{y}_t = \sum_{j=1}^{k} p(j|\mathbf{x}_t)[\sum_{\tau=1}^{q_j} a_{j.\tau} z_{t+1-\tau} + c_j],$$





Examples of Mixture of experts

1. Hard - competition

$$p(j\,|\,\mathbf{x}_t) = \begin{cases} 1, & \text{if } j = c \\ 0, & \text{otherwise.} \end{cases} \qquad c = \arg\min_j d_j$$

for different d_j.

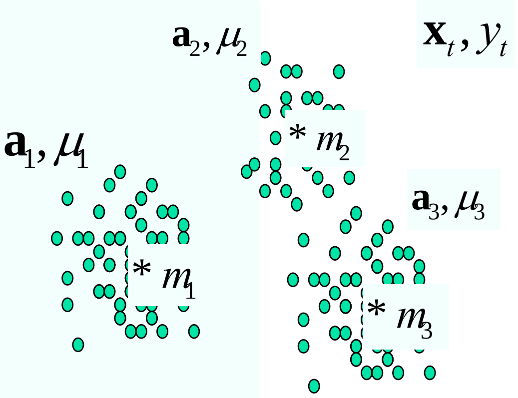
2. Rivial penalized competition

$$p(j\,|\,\mathbf{x}_t) = \begin{cases} 1, & \text{if } j = c,\ c = \arg\min_j d_j, \\ -\gamma, & \text{if } j = r,\ r = \arg\min_{j \neq c} d_j \\ 0, & \text{otherwise.} \end{cases}$$

for different d_j

3. Soft - aollcation

$$p(j|\mathbf{x}_t) = \frac{\alpha_j G(\mathbf{x}_t\,|\,m_j, \Sigma_j)}{\sum_{j=1}^{k} \alpha_j G(\mathbf{x}_t\,|\,m_j, \Sigma_j)}, \qquad p(j|\varepsilon_t) = \frac{\alpha_j G(\varepsilon_t\,|\,0, \lambda_j^2)}{\sum_{j=1}^{k} \alpha_j G(\varepsilon_t\,|\,0, \lambda_j^2)}$$

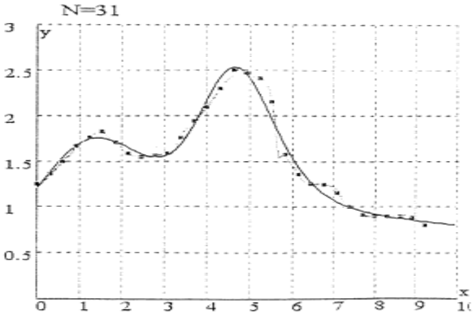


---

1. Financial Prediction

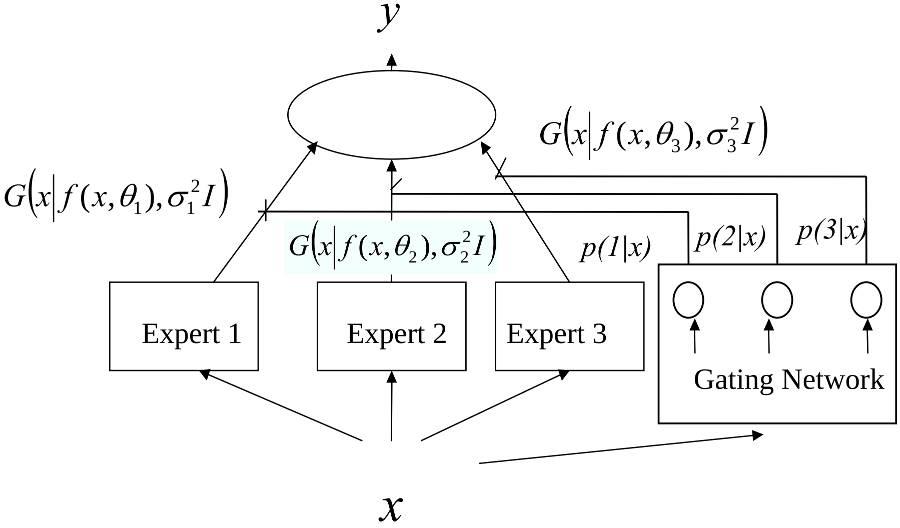
- RPCL competitive learning based piecewise linear prediction
- Extended radial basis functions, Mixture of expert model and financial prediction
- Finite mixture of ARCH and GRACH models for prediction
- APT-TFA based prediction

Mixture-of-experts

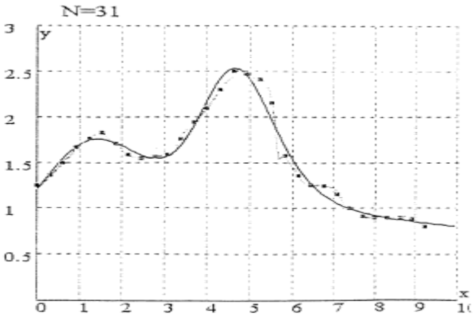


$$E(y\,|\,x) = f(x,\theta) = \sum_j p(j\,|\,x) f(x,\theta_j)$$

$$p(y\,|\,x) = \sum_j p(j\,|\,x) G\left(y \middle| f(x,\theta_j), \sigma_j^2 I\right)$$



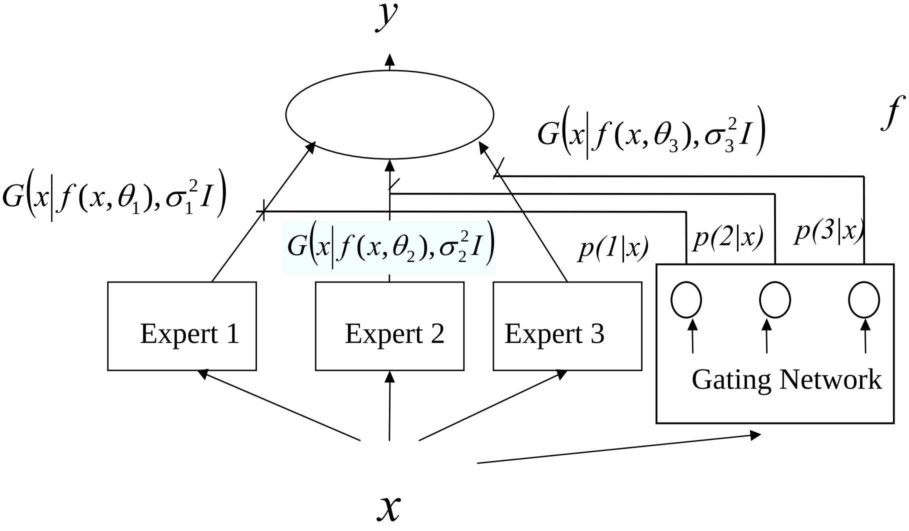
Linear Mixture-of-experts



$$E(y\,|\,x) = f(x,\theta) = \sum_j p(j\,|\,x) f(x,\theta_j)$$

$$p(y\,|\,x) = \sum_j p(j\,|\,x) G\left(y \middle| f(x,\theta_j), \sigma_j^2 I\right)$$

$$f(x,\theta_j) = \begin{cases} W_j x + c_j \\ c_j \end{cases}$$



RBF nets and Extended RBF nets

$$f_k(x) = \frac{\sum_{j=1}^{k}(0 \quad x + c_j)\phi([x-m_j]^{\mathrm{T}}\Sigma_j^{-1}[x-m_j])}{\sum_{j=1}^{k}\phi([x-m_j]^{\mathrm{T}}\Sigma_j^{-1}[x-m_j])},$$

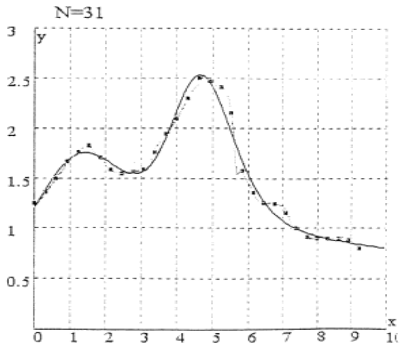
$\Sigma_j = \sigma_j^2 I,$ a RBF net

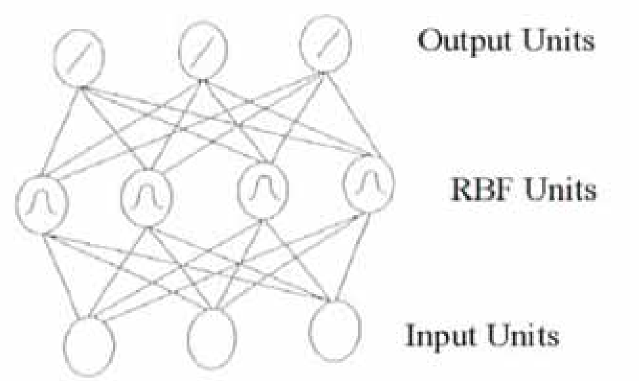
$$f_k(x) = \frac{\sum_{j=1}^{k}(0 \quad + c_j)\phi([x-m_j]^{\mathrm{T}}\Sigma_j^{-1}[x-m_j])}{\sum_{j=1}^{k}\phi([x-m_j]^{\mathrm{T}}\Sigma_j^{-1}[x-m_j])},$$

an ellipstic RBF net

$$f_k(x) = \frac{\sum_{j=1}^{k}(W_j^{\mathrm{T}}x + c_j)\phi([x-m_j]^{\mathrm{T}}\Sigma_j^{-1}[x-m_j])}{\sum_{j=1}^{k}\phi([x-m_j]^{\mathrm{T}}\Sigma_j^{-1}[x-m_j])},$$

Extended RBF net

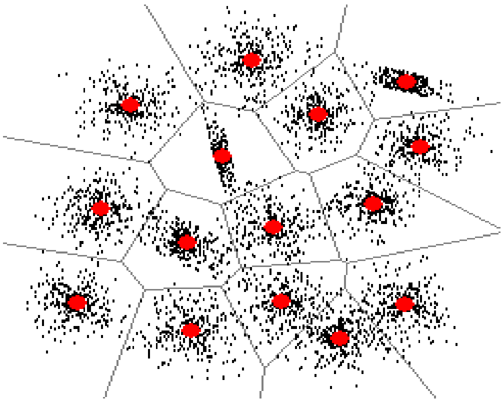


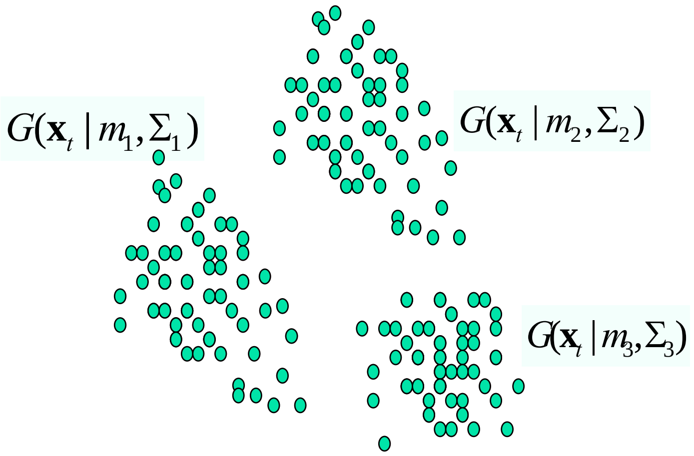


Step I of learning

Mean Square Error (MSE) Clustering

Gaussian Mixture





K-mean algorithm

EM algorithm

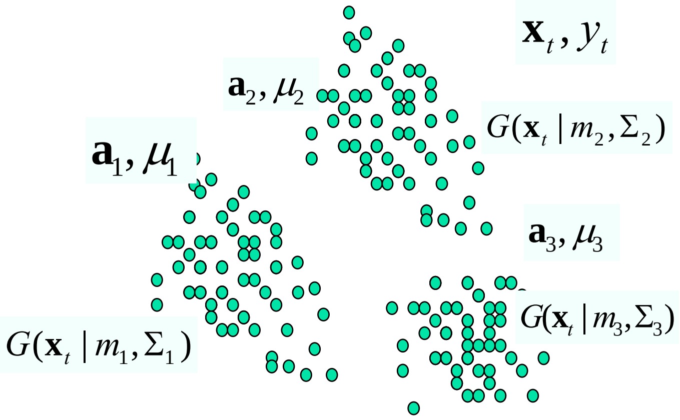
Step II of learning

$$f_k(x) = \frac{\sum_{j=1}^{k}(W_j^{\mathrm{T}}x + c_j)\phi([x-m_j]^{\mathrm{T}}\Sigma_j^{-1}[x-m_j])}{\sum_{j=1}^{k}\phi([x-m_j]^{\mathrm{T}}\Sigma_j^{-1}[x-m_j])},$$

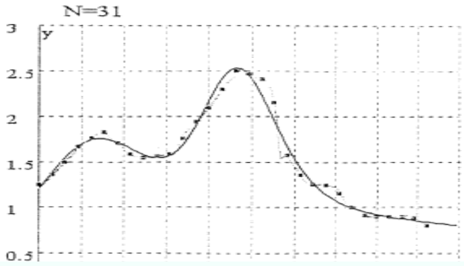
$$= p(j\,|\,\mathbf{x}_t)$$

We can get W_j, c_j by the least square regression, i.e.

$$\min \sum_t \sum_{j=1}^{k} p(j\,|\mathbf{x}_t)\left\| y_t \text{ -} W_j x - c_j \right\|^2$$



Extended RBF nets: Specific cases of Linear ME



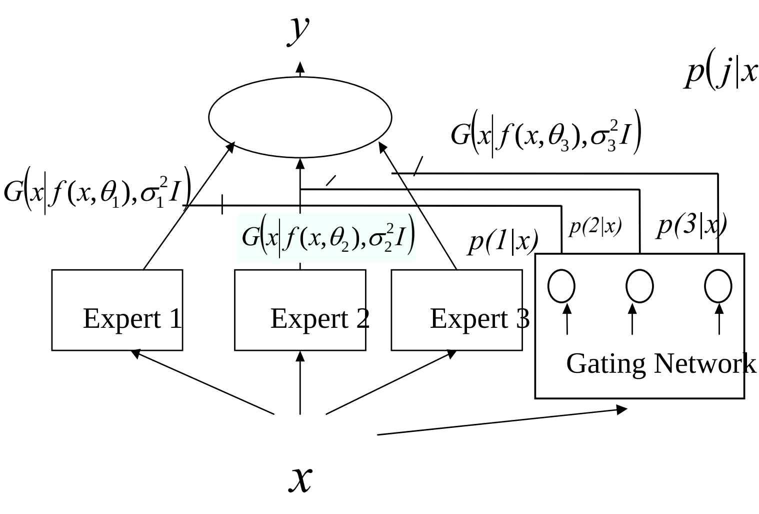
$$E(y\,|\,x) = f(x,\theta) = \sum_j p(j\,|\,x)(W_j x + c_j)$$

$$f_k(x) = \frac{\sum_{j=1}^{k}(W_j^{\mathrm{T}}x + c_j)\phi([x-m_j]^{\mathrm{T}}\Sigma_j^{-1}[x-m_j])}{\sum_{j=1}^{k}\phi([x-m_j]^{\mathrm{T}}\Sigma_j^{-1}[x-m_j])},$$

$$p(j|\mathbf{x}_t) = \frac{\alpha_j G(\mathbf{x}_t\,|\,m_j,\Sigma_j)}{\sum_{j=1}^{k}\alpha_j G(\mathbf{x}_t\,|\,m_j,\Sigma_j)}, \quad \alpha_j = \frac{|\Sigma_j|}{\Sigma_r|\Sigma_r|},$$

$$\Downarrow$$

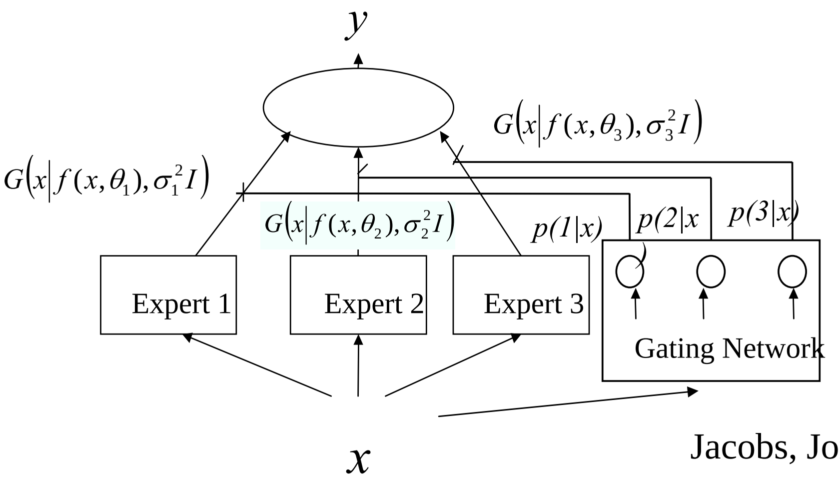
$$p(j|x) = \frac{\exp\left(-0.5(x-m_j)^T\Sigma_j^{-1}(x-m_j)\right)}{\sum_j \exp\left(-0.5(x-m_j)^T\Sigma_j^{-1}(x-m_j)\right)}$$



Mixture-of-experts

$$p(y\,|\,x)=\sum_j p(j\,|\,x)G\left(y\middle|f(x,\theta_j),\sigma_j^2 I\right)\quad E(y\,|\,x)=f(x,\theta)=\sum_j p(j\,|\,x)f(x,\theta_j)$$

- The EM algorithm (Jordan & Jacobs, 1994)
- Study on its convergence (Jordan & Xu, Neural Networks, 1995)
- J(k) for selecting k and automatic selection during learning (Xu, Neurocomputing, 98, Intl J Neural Systems, 01, Neural Networks, 02)



$$p(j\,|\,x)=\frac{e^{-h_j(x,\phi_j)}}{\sum_r e^{-h_r(x,\phi_r)}}$$

Jacobs, Jordan, Nowlan, & Hinton, 1991

Extended RBF nets

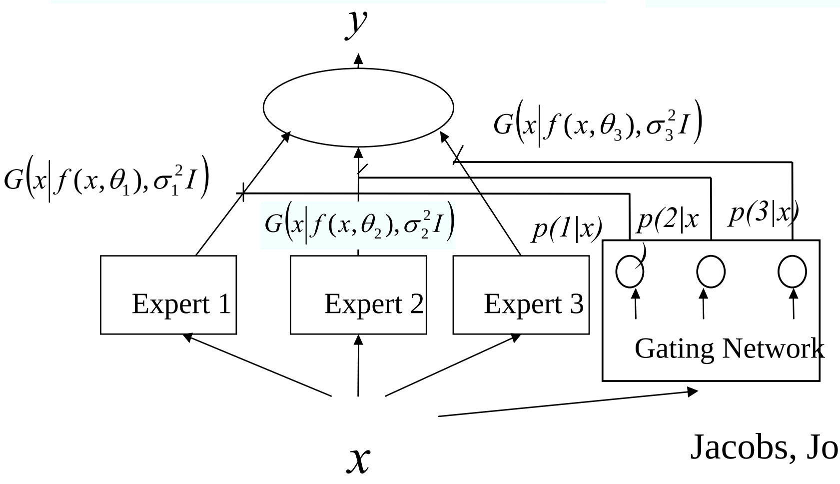
- statistical consistency, convergence rates and receptive field size, among early major theoretical results in the literature of RBF nets. (Xu, Krzyzak, & Yuille, Harvard Robotic Lab, T.Rep, 1992, Neural Networks, 94)
- EM algorithm in place of the suboptimal clustering +LMS way (Xu, Neurocomputing, 98)
- J(k) for selecting k and automatic selection (either RPCL or BYY learning) (Xu, Neurocomputing, 98, Intl J Neural Systems, 01, Neural Networks, 02)
- applied to time series prediction, financial portfolio management.

Mixture-of-experts

E step: $p(j \mid \mathbf{x}_t) = \frac{e^{-h_j(\mathbf{x}_t,\phi_j)}}{\sum_r e^{-h_r(\mathbf{x}_t,\phi_r)}}$

M step:

$$\max_{\phi_j} \sum_t \sum_{j=1}^{k} p(j \mid \mathbf{x}_t) \ln p(j \mid \mathbf{x}_t) \qquad \min_{\theta_j} \sum_t \sum_{j=1}^{k} p(j \mid \mathbf{x}_t) \left\| y_t - f(\mathbf{x}_t, \theta_j) \right\|^2$$

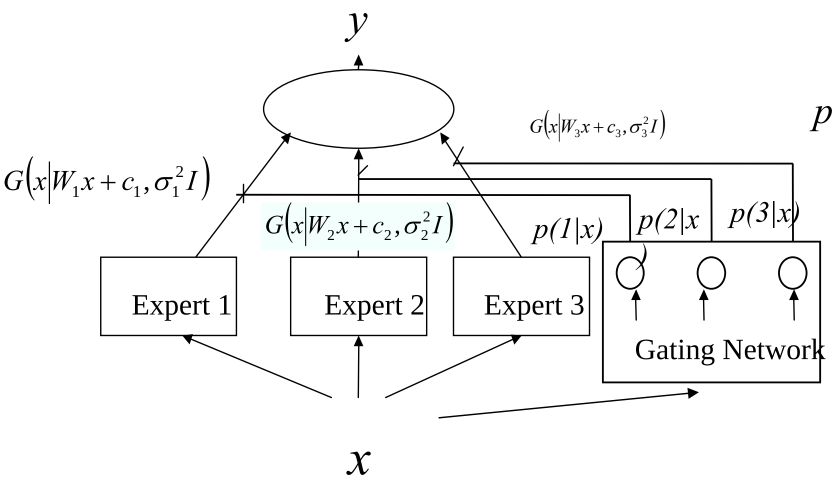


Jacobs, Jordan, Nowlan, & Hinton, 1991

Alternative Mixture-of-Experts (ME)

$$p(y \mid x) = \sum_j p(j \mid x) G\left(y \middle| W_j x + c_j, \sigma_j^2 I\right) \qquad E(y \mid x) = f(x,\theta) = \sum_j p(j \mid x)(W_j x + c_j)$$

- Easy to be implemented by the EM algorithm
  (Xu, Jordan & Hinton, IJCNN, 1994; A. NIPS, 1995)
- J(k) for selecting k and automatic selection during learning
  (Xu, Neurocomputing, 98, Intl J Neural Systems, 01, Neural Networks, 02)



$$p(j|\mathbf{x}_t) = \frac{\alpha_j G(\mathbf{x}_t \mid m_j, \Sigma_j)}{\sum_{j=1}^{k} \alpha_j G(\mathbf{x}_t \mid m_j, \Sigma_j)}$$

The EM algorithm for Alternative ME

E step:

$$p(j|\mathbf{x}_t) = \frac{\alpha_j G(\mathbf{x}_t \,|\, m_j, \Sigma_j) G\left(y_t \middle| W_j \mathbf{x}_t + c_j, \sigma_j^2 I\right)}{\sum_{j=1}^{k} \alpha_j G(\mathbf{x}_t \,|\, m_j, \Sigma_j) G\left(y_t \middle| W_j \mathbf{x}_t + c_j, \sigma_j^2 I\right)}$$

M step:

$$\max \sum_t \sum_{j=1}^{k} p(j|\mathbf{x}_t) \ln[\alpha_j G(\mathbf{x}_t \,|\, m_j, \Sigma_j)]$$

$$\min \sum_t \sum_{j=1}^{k} p(j|\mathbf{x}_t) \left\| y_t - (W_j \mathbf{x}_t + c_j) \right\|^2$$

Lei Xu (1998a), ``RBF Nets, Mixture Experts, and Bayesian Ying-Yang Learning'', *Neurocomputing, Vol. 19, No.1-3*, pp223-257, 1998

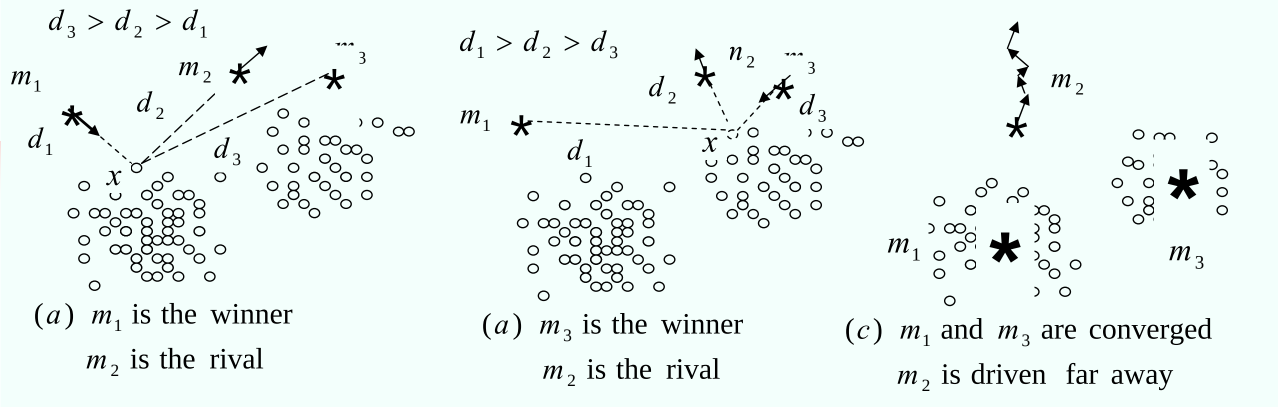
Lei Xu, M.I.Jordan & G. E. Hinton (1995), `` An Alternative Model for Mixtures of Experts'', *Advances in Neural Information Processing Systems 7*, eds., Cowan, J.D., Tesauro, G., and Alspector, J., MIT Press, Cambridge MA, 1995, pp633-640.

M.I.Jordan & **Lei Xu** (1995), `` Convergence results for the EM approach to mixtures-of-experts architectures'', *Neural Networks*, Vol.8, No.9, pp1409-1431.

Lei Xu (2001), ``Best Harmony, Unified RPCL and Automated Model Selection for Unsupervised and Supervised Learning on Gaussian Mixtures, ME-RBF Models and Three-Layer Nets '', *International Journal of Neural Systems, Vol.11, No.1*, pp3-69, 2001.

Lei Xu (2002b), ``BYY harmony learning, structural RPCL, and topological self-organizing on mixture models'', *Neural Networks, Vol. 15*, pp1125-1151, 2002.

RPCL Learning for Alternative ME



(a) m_1 is the winner m_2 is the rival

(a) m_3 is the winner m_2 is the rival

(c) m_1 and m_3 are converged m_2 is driven far away

$$d_j(\theta_j) = -\ln[\alpha_j G(\mathbf{x}_t \mid m_j, \Sigma_j) G\left(y_t \middle| W_j \mathbf{x}_t + c_j, \lambda_j^2 I\right)] = d_j^x(\theta_j^x) + d_j^y(\theta_j^y)$$

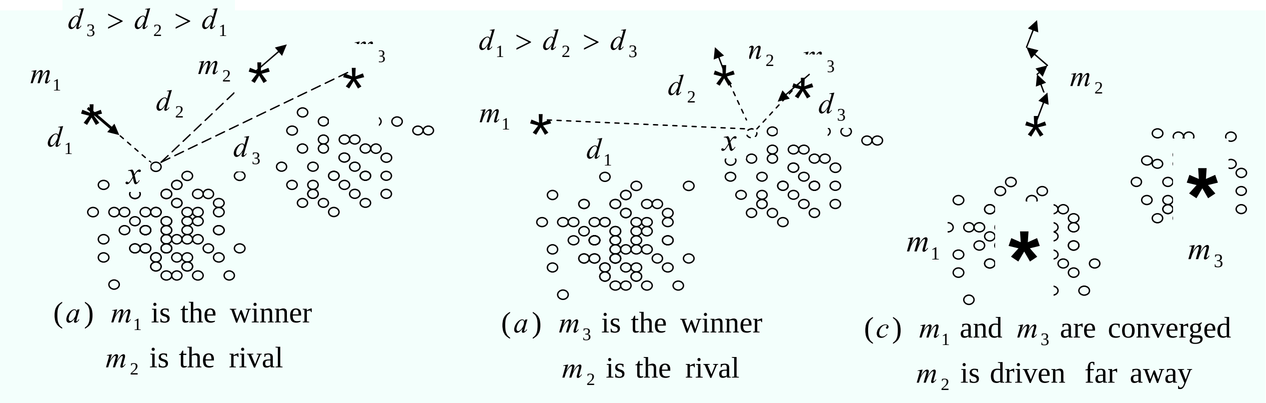
$$c = \arg\min_j d_j, \qquad r = \arg\min_{j \neq c} d_j$$

$$d_j^x(\theta_j^x) = -\ln[\alpha_j G(\mathbf{x}_t \mid m_j, \Sigma_j)], \quad d_j^y(\theta_j^y) = -\ln G\left(y_t \middle| W_j \mathbf{x}_t + c_j, \lambda_j^2 I\right)$$

$$\theta_c^{x\,new} = \theta_c^{x\,old} - \eta_c \frac{\partial d_c^x(\theta_c^x)}{\partial \theta_c^x}, \qquad \theta_c^{y\,new} = \theta_c^{y\,old} - \eta_c \frac{\partial d_c^y(\theta_c^y)}{\partial \theta_c^y},$$

$$\theta_r^{x\,new} = \theta_r^{x\,old} + \eta_r \frac{\partial d_r^x(\theta_r^x)}{\partial \theta_r^x}, \qquad \theta_r^{y\,new} = \theta_r^{y\,old} + \eta_r \frac{\partial d_r^y(\theta_r^y)}{\partial \theta_r^y},$$

Versus Gaussian Mixture based RPCL learning



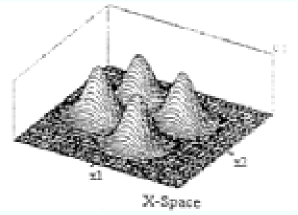
(a) m_1 is the winner m_2 is the rival

(a) m_3 is the winner m_2 is the rival

(c) m_1 and m_3 are converged m_2 is driven far away

Gaussian cluster

$$d_j^x(\theta_j^x) = -\ln[\alpha_j G(\mathbf{x}_t \mid m_j, \Sigma_j)],$$



$$\theta_c^{x\,new} = \theta_c^{x\,old} - \eta_c \frac{\partial d_c^x(\theta_c^x)}{\partial \theta_c^x},$$

$$\theta_r^{x\,new} = \theta_r^{x\,old} + \eta_r \frac{\partial d_r^x(\theta_r^x)}{\partial \theta_r^x},$$

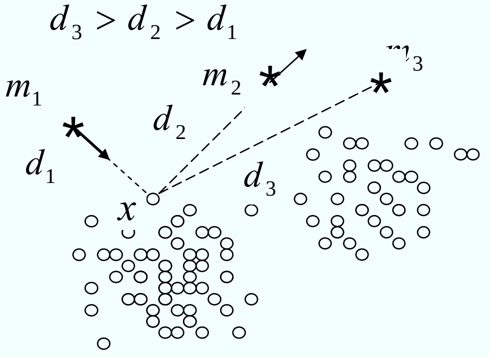
$$m_c^{new} = m_c^{old} + \eta_c(\mathbf{x}_t - m_c^{old}), \quad c = \arg\min_j d_j$$

$$m_r^{new} = m_r^{old} - \eta_r(\mathbf{x}_t - m_r^{old}), \quad r = \arg\min_{j \neq c} d_j$$

$$\Sigma_j = S_j^T S_j, \quad \alpha_j = \gamma_j^2$$

L. Xu, IJCNN98, May 5-9, 1998, Alaska, Vol.II, pp2525-2530.
L. Xu, Intl J. Neural Systems, Vol.11, No.1, pp3-69, 2001.

Prediction error based RPCL Learning



m_1 is the winner

m_2 is the rival

$$\varepsilon_{t,j} = y_t - W_j \mathbf{x}_t - c_j$$

$$W_c^{new} = W_c^{old} + \eta_c \varepsilon_{t,c} \mathbf{x}_t^T,$$

$$W_r^{new} = W_r^{old} - \eta_r \varepsilon_{t,r} \mathbf{x}_t^T$$

$$c_c^{new} = c_c^{old} + \eta_c \varepsilon_{t,c},$$

$$c_r^{new} = c_r^{old} - \eta_r \varepsilon_{t,r}$$

$$d_j^y(\theta_j^y) = -\ln G\left(y_t \middle| W_j \mathbf{x}_t + c_j, \lambda_j^2 I\right)$$

$$\lambda_c^{2new} = (1 - \eta_c)\lambda_c^{2old} + \eta_c \frac{\left\|\varepsilon_{t,c}\right\|^2}{d}$$

$$\theta_c^{y\,new} = \theta_c^{y\,old} - \eta_c \frac{\partial d_c^y(\theta_c^y)}{\partial \theta_c^y},$$

$$\lambda_r^{new} = \lambda_r^{old} + \frac{\eta_r}{\lambda_r^{old}}\left(\lambda_c^{2old} - \frac{\left\|\varepsilon_{t,c}\right\|^2}{d}\right)$$

$$\theta_r^{y\,new} = \theta_r^{y\,old} + \eta_r \frac{\partial d_r^y(\theta_r^y)}{\partial \theta_r^y},$$

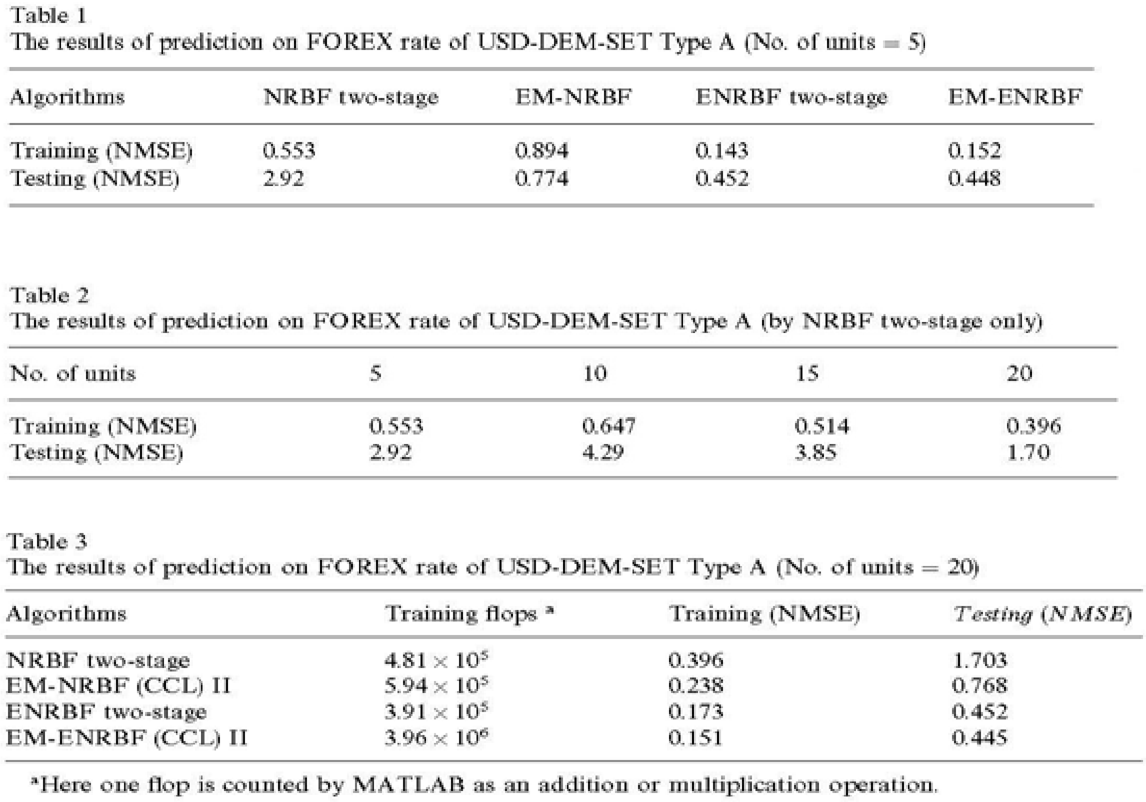


Table 1
The results of prediction on FOREX rate of USD-DEM-SET Type A (No. of units = 5)

Algorithms	NRBF two-stage	EM-NRBF	ENRBF two-stage	EM-ENRBF
Training (NMSE)	0.553	0.894	0.143	0.152
Testing (NMSE)	2.92	0.774	0.452	0.448

Table 2
The results of prediction on FOREX rate of USD-DEM-SET Type A (by NRBF two-stage only)

No. of units	5	10	15	20
Training (NMSE)	0.553	0.647	0.514	0.396
Testing (NMSE)	2.92	4.29	3.85	1.70

Table 3
The results of prediction on FOREX rate of USD-DEM-SET Type A (No. of units = 20)

Algorithms	Training flops [a]	Training (NMSE)	*Testing (NMSE)*
NRBF two-stage	4.81×10^5	0.396	1.703
EM-NRBF (CCL) II	5.94×10^5	0.238	0.768
ENRBF two-stage	3.91×10^5	0.173	0.452
EM-ENRBF (CCL) II	3.96×10^6	0.151	0.445

[a] Here one flop is counted by MATLAB as an addition or multiplication operation.

Table 4
The results of trading investment based on the prediction on USD-DEM-SET Type A

Algorithms	Net profit point	Profit in US$ (in 112 days)
EM-NRBF (CCL)	1425	9262.5
Adaptive EM-NRBF (CCL)	3966	25779.0
EM-ENRBF (CCL)	2063	13406.5
Adaptive EM-ENRBF (CCL)	2916	18954.0

Table 5
The results of trading investment by Supervised Decision Network on USD-DEM-SET Type A

Algorithms	Net profit point	Profit in US$ (in 112 days)
EM-NRBF (CCL)	1605	10432.5
Adaptive EM-NRBF (CCL)	4237	27540.5
EM-ENRBF (CCL)	2660	17290.0
Adaptive EM-ENRBF (CCL)	3207	20845.5

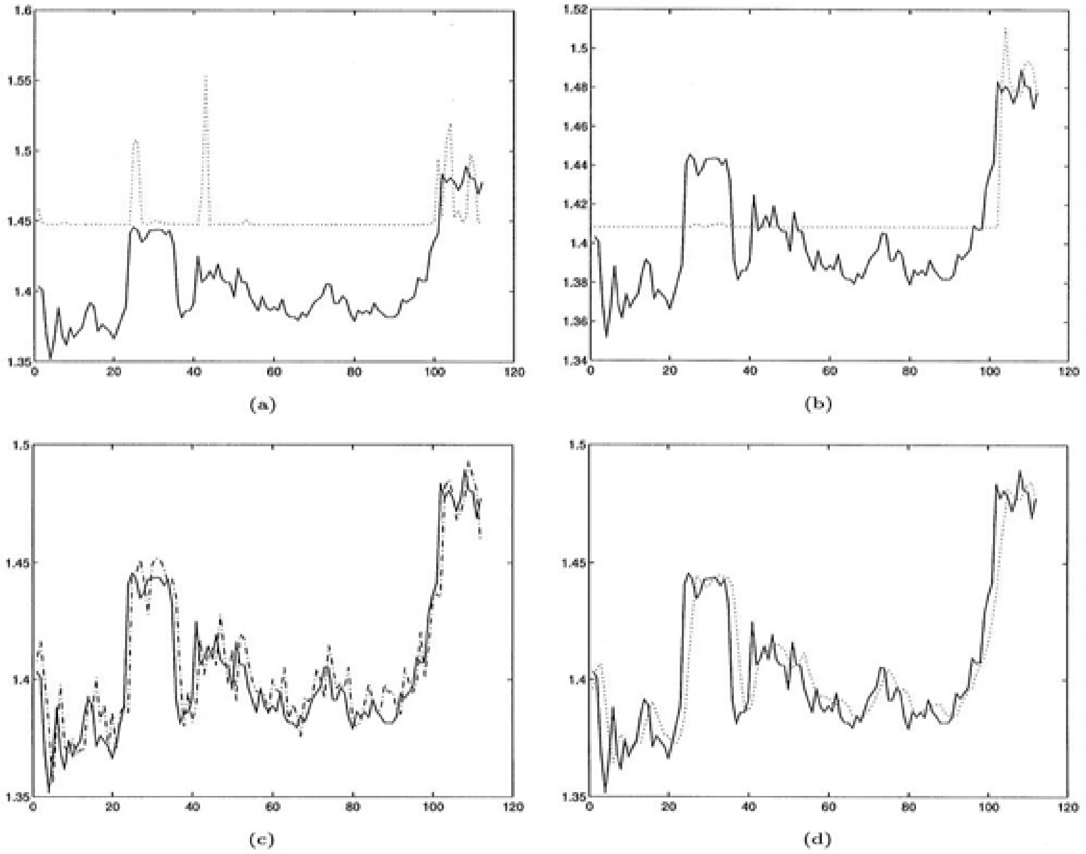


Fig. 2. The results of prediction on Forex data of USD-DEM-Set Type A (No. of units = 5), corresponding to Table 1, where the solid line is for data and the dashed line is for prediction result, and this convention is kept the same for all the figures in this paper: (a) by NRBF two-stage, (b) by EM-NRBF, (c) by ENRBF two-stage, (d) by EM-ENRBF.

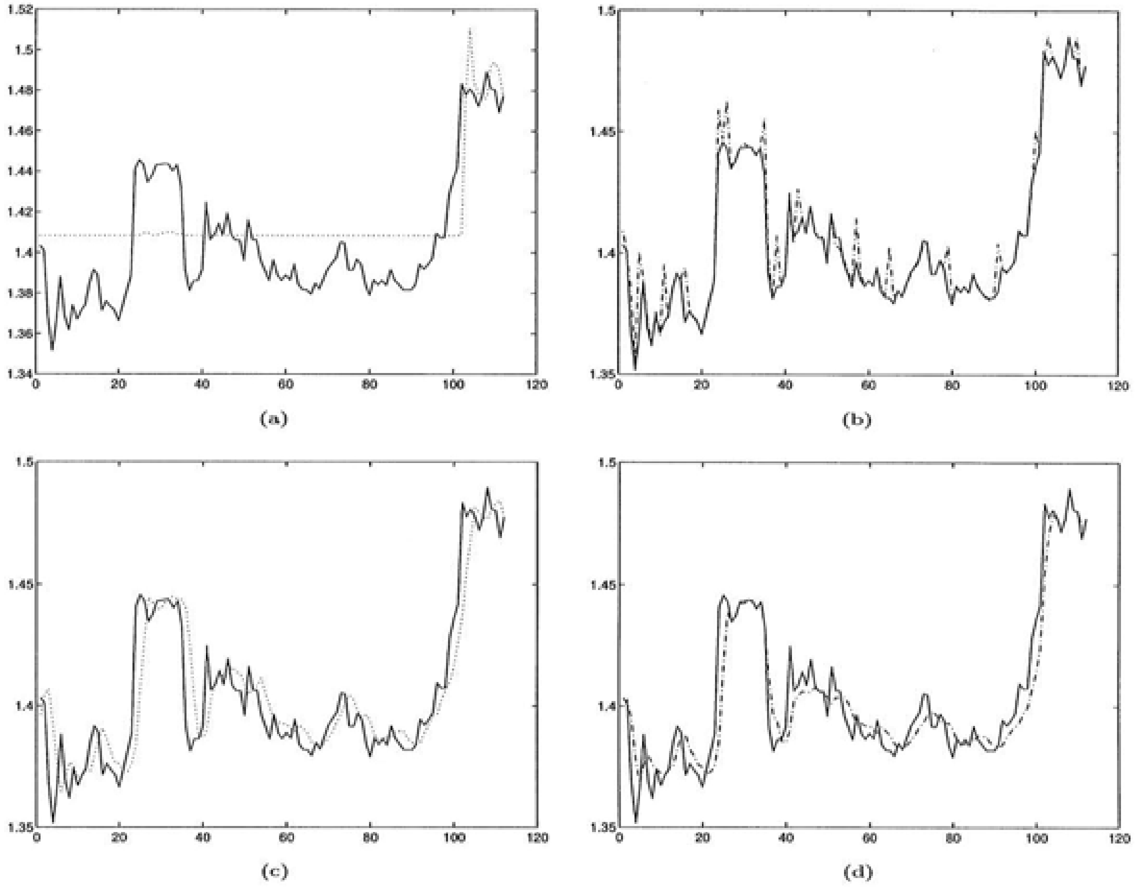
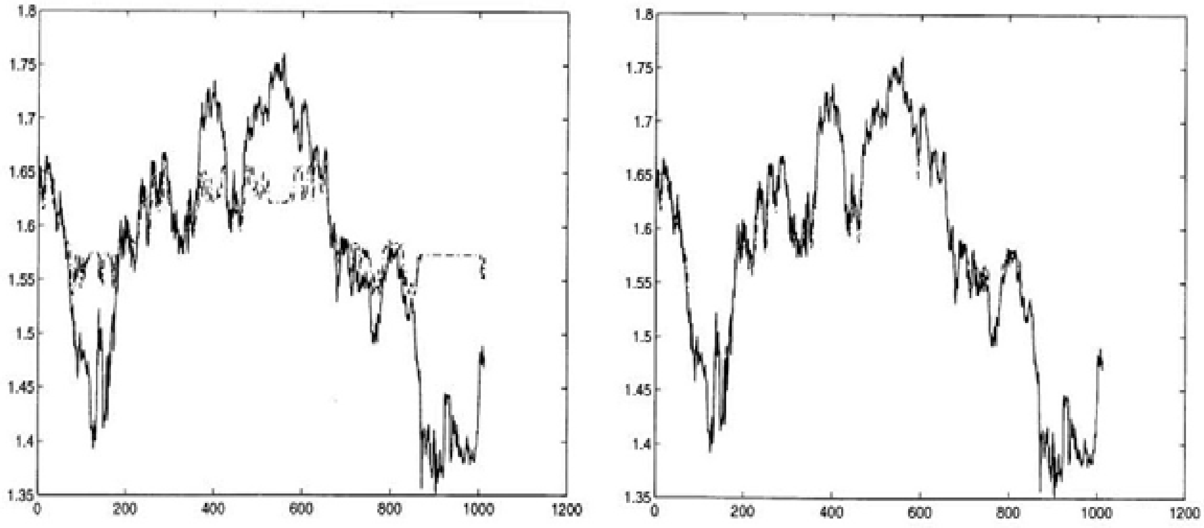


Fig. 3. The results of prediction on Forex data of USD-DEM-Set Type A (No. of units = 20), corresponding to Table 3: (a) by NRBF two-stage, (b) by EM-NRBF, (c) by ENRBF two-stage, (d) by EM-ENRBF Algorithm II.



(a) Normalized Testing Error = 0.0840

(b) Normalized Testing Error = 0.0083

Fig. 4. The results of prediction on Forex data of USD-DEM-SET Type B (No. of units = 20): (a) by EM-NRBF (CCL), (b) by Adaptive EM-NRBF (the prediction and the real data are almost overlapped).

1. Financial Prediction

- RPCL competitive learning based piecewise linear prediction
- Extended radial basis functions, Mixture of expert model and financial prediction
- Finite mixture of ARCH and GRACH models for prediction
- APT-TFA based prediction

Autogressive model $z_{t+1} = \sum_{\tau=1}^{p} a_\tau z_{t+1-\tau} + \varepsilon_t, \quad P(\varepsilon_t) = G(\varepsilon_t \,|\, 0, \lambda^2)$

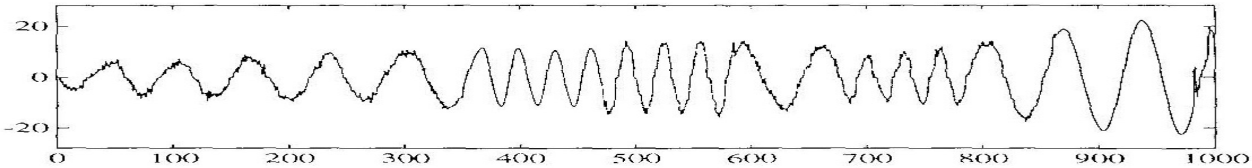
- ARCH model：

$$\lambda^2(t) = \alpha_0 + \sum_{i=1}^{q} \alpha_i \lambda^2(t-i)$$

- GARCH model：

$$\lambda^2(t) = \alpha_0 + \sum_{i=1}^{q} \alpha_i \lambda^2(t-i) + \sum_{i=1}^{p} \beta_i \varepsilon_{t-i}^2$$

a conditional heterodastic variance



$$z_{t+1} = \sum_{\tau=1}^{q_j} a_{j.\tau} z_{t+1-\tau} + c_j + \varepsilon_j, \quad P(\varepsilon_j) = G(\varepsilon_j \,|\, 0, \lambda_j^2(t)), \;\; j = 1, \cdots, k$$

- Mixture of ARCH models：

$$\lambda_j^2(t) = \alpha_{0,j} + \sum_{i=1}^{q} \alpha_{i,j} \lambda_j^2(t-i)$$

- Mixture of GARCH models：

$$\lambda_j^2(t) = \alpha_{0,j} + \sum_{i=1}^{q} \alpha_{i,j} \lambda_j^2(t-i) + \sum_{i=1}^{p} \beta_{i,j} \varepsilon_j^2(t-i)$$

a conditional heterodastic variance

outperforms GARCH model considerably

Best parametric model matching 参数模型最佳匹配

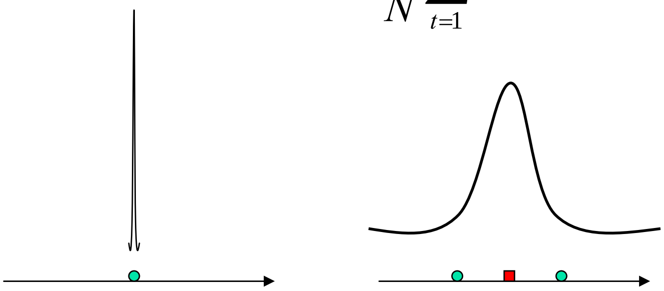
optimizing a matching cost

$$F\big(p(x\,|\,\theta), \mathrm{X}\big), \quad X = \{x_t\}_{t=1}^{N}$$

$$p\big(x\,|\,\hat{\theta}(X)\big)$$

Maximum Likelihood (ML) 最大似然

$$\max_\theta L(\theta), \quad L(\theta) = \frac{1}{N}\sum_{t=1}^{N} \ln p\big(x_t\,|\,\theta\big)$$



One piece of evidence,
take it by 100%

Two pieces of evidence
take each by 50% subject
to the template

More pieces of evidence
… …

The large number law

$$p_*(x|\theta) \to p_*(x|\theta_0) \qquad \hat{\theta}(\mathrm{X}) \to \theta_0 \qquad as\;\; N \to \infty$$

Maximum Likelihood Learning

W. Wong, F. Yip, and L. Xu, ``Financial Prediction by Finite Mixture GARCH Model'', Proc. of International Conference on Neural Information Processing (ICONIP'98), October 21-23, 1998, Kitakyushu, Japan, Vol.3, pp1351-1354.

$$\text{Maximum Learning on } P(\varepsilon_t \mid \theta) = \sum_{j=1}^{k} \pi_j G(\varepsilon_t \mid 0, \lambda_j^2(t)),$$

$$\theta = \{\pi_j, c_j, a_{j,\tau}, \tau = 1, \cdots, q_j, \alpha_{i,j}, \beta_{i,j}\}_{j=1}^{k}$$

- BHHH (Berndt, Hall, Hall and Hausman) algorithm:

$$\theta^{new} = \theta^{old} + \lambda_i [\widehat{I}_{\theta\theta}]^{-1} \frac{1}{T} \sum_t \frac{\partial l_t}{\partial \theta}$$

$$\widehat{I}_{\mathbf{a}_j \mathbf{a}_j} = \sum - \frac{1}{2} p(j|\varepsilon_t) \lambda_j^{-4}(t) \frac{\partial \lambda_j^3(t)}{\partial \mathbf{a}_j} \frac{\partial \lambda_j^2(t)}{\partial \mathbf{a}_j{}^T} \frac{\varepsilon_j^2(t)}{\lambda_j^2(t)}$$

The EM algorithm

E step :

$$p(j|\varepsilon_t) = \frac{\alpha_j G(\varepsilon_t \mid 0, \lambda_j^2(t))}{\sum_{j=1}^{k} \alpha_j G(\varepsilon_t \mid 0, \lambda_j^2(t))}$$

M step:

$$\max \sum_t \sum_{j=1}^{k} p(j|\varepsilon_t) \ln[\alpha_j G(\varepsilon_t \mid 0, \lambda_j^2(\mathrm{t}))]$$

$$\alpha_j = \frac{1}{N} \sum_t p(j \mid \varepsilon_t)$$

$$\max_{\alpha_j \geq 0, \alpha_{i,j}, \beta_{i,j}} \sum_t \sum_{j=1}^{k} p(j|\varepsilon_t) \ln[\alpha_j G(\varepsilon_t \mid 0, \lambda_j^2(t))]$$

$$\min_{\mu_{\mathrm{j}}, \mathbf{a}_j, m_j} \sum_t \sum_{j=1}^{k} p(j|\varepsilon_t) \left\| y_t - \mu_{\mathrm{j}} - \mathbf{a}_j^t (\mathbf{x}_t - m_j) \right\|^2$$

Prediction

After the completion of parameter training, we have the prediction formula:

$$\hat{y}_t = \sum_{j=1}^{k} p(j|\varepsilon_t)[\sum_{\tau=1}^{q_j} a_{j.\tau} z_{t+1-\tau} + c_j]$$

$$p(j|\varepsilon_t) = \frac{\alpha_j G(\varepsilon_t \,|\, 0, \lambda_j^2(t))}{\sum_{j=1}^{k} \alpha_j G(\varepsilon_t \,|\, 0, \lambda_j^2(t))}$$

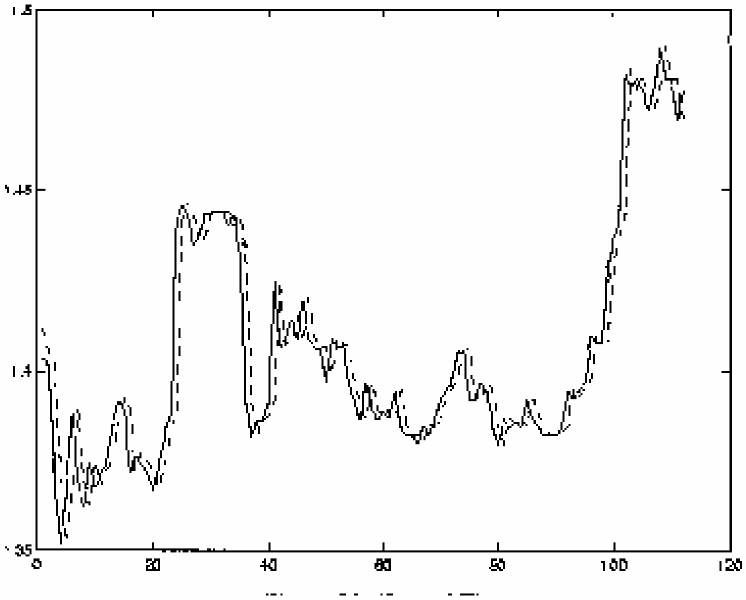
- Four experiments conducted with real foreign exchange rate
    - USD vs DEM
    - USD vs GRP
    - USD vs SWF
    - USD vs FRN

W.C.Wong, F.Yip, & L.Xu, Proc. ICONIP98

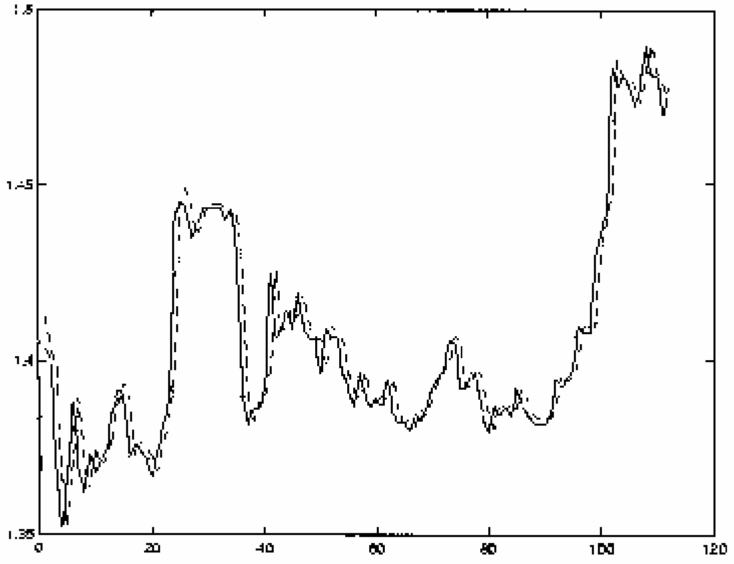
Root-mean-square Error

USD vs.	GARCH model	Finite Mixture of GARCH model	Improvement (%)
DEM	0.0114	0.0102	10.5
FRN	0.0405	0.0364	10.1
GRP	0.0028	0.0021	25.0
SWF	0.0107	0.0089	16.8

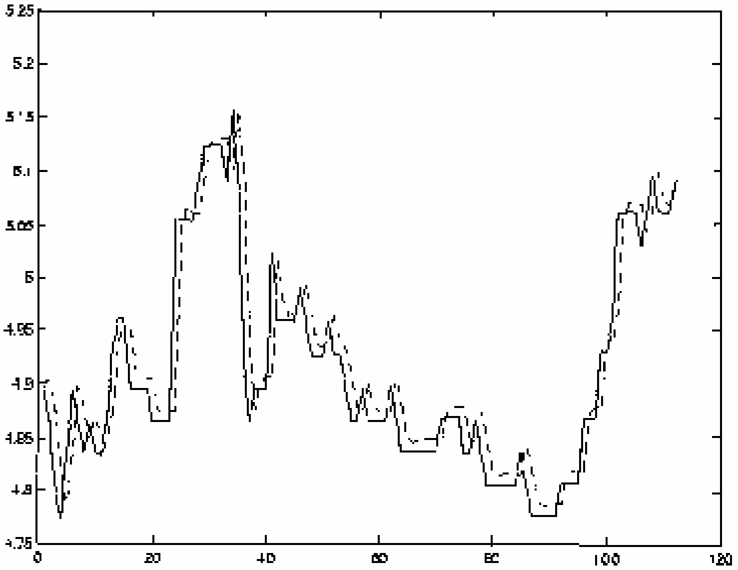
USD vs DEM(GARCH)



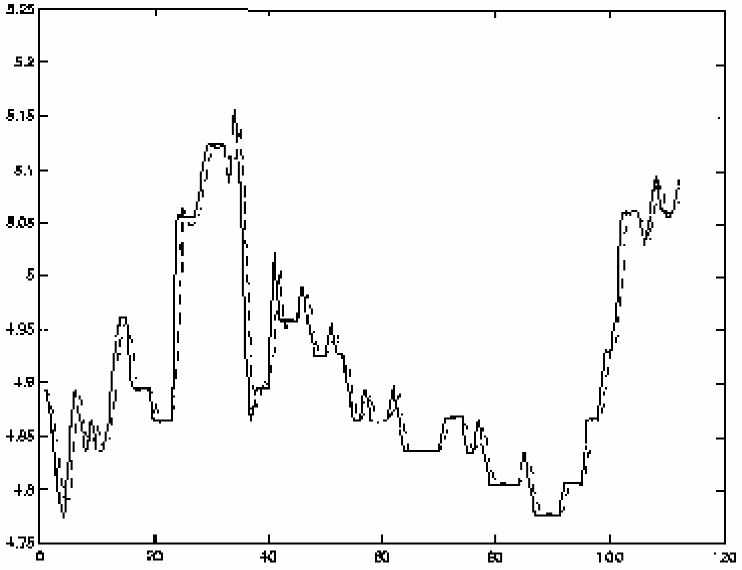
USE vs DEM(Finite Mixture of GARCH)



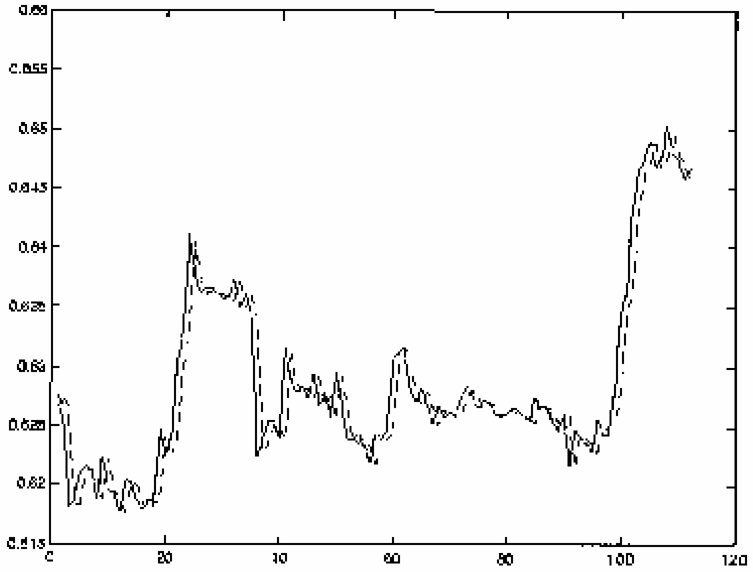
USD vs FRN(GARCH)



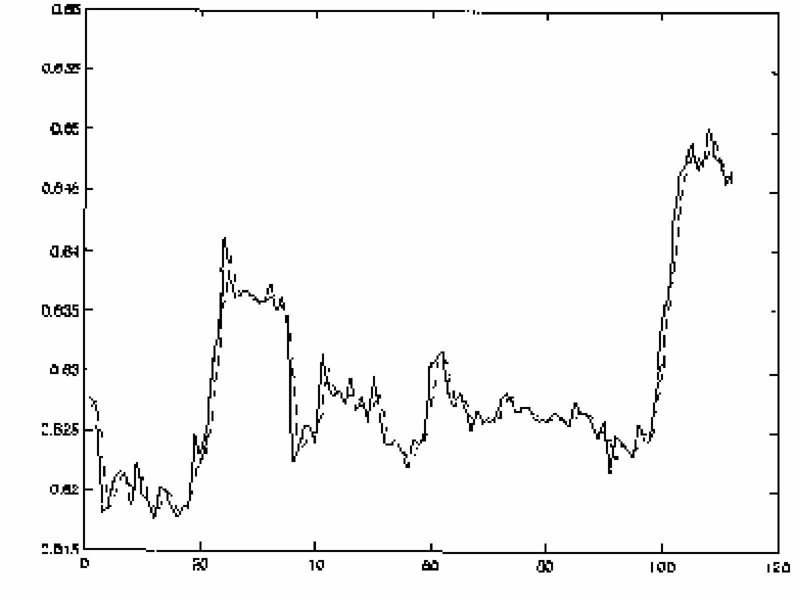
USD vs. FRN(Finite Mixture of GARCH)



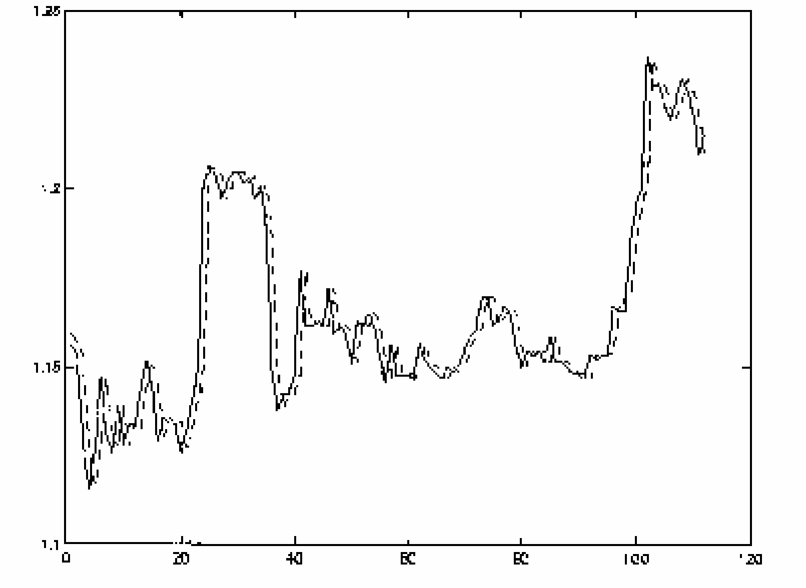
USD vs. GRP(GARCH)



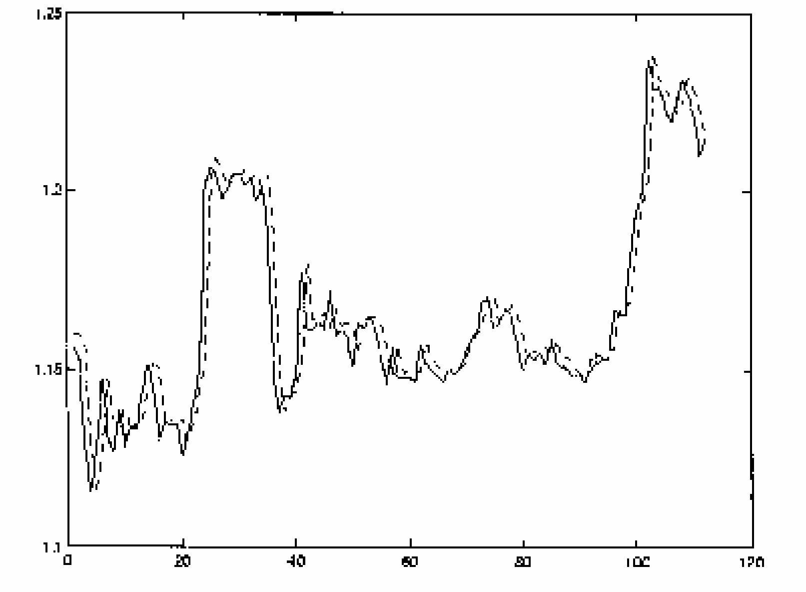
USD vs. GRP (Finite Mixture of GARCH)



USD vs. SWF(GARCH)



USD vs. SWF (Finite Mixture of GARCH)



Mixture of ARMA-GARCH models :

$$z_{t+1} = \sum_{\tau=1}^{q_j} a_{j.\tau} z_{t+1-\tau} + c_j + \varepsilon_j(t) + \sum_{\tau=1}^{p_j} b_{j.\tau} \varepsilon_j(t-\tau),$$

$$P(\varepsilon_j) = G(\varepsilon_j \mid 0, \lambda_j^2(t)), \ \ j = 1, \cdots, k$$

$$\lambda_j^2(t) = \alpha_{0,j} + \sum\nolimits_{i=1}^{q} \alpha_{i,j} \varepsilon_j^2(t-i) + \sum\nolimits_{i=1}^{p} \beta_{i,j} \lambda^2(t-i)$$

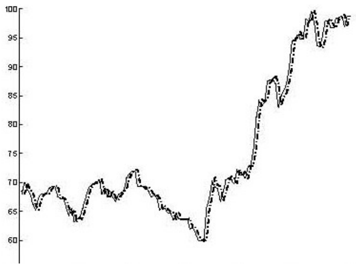
E step :

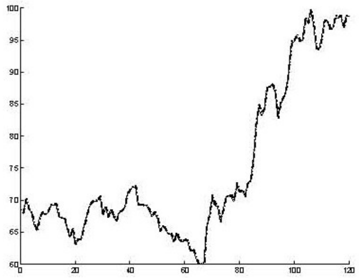
$$p(j|\varepsilon_t) = \frac{\alpha_j G(\varepsilon_t \mid 0, \lambda_j^2(t))}{\sum_{j=1}^{k} \alpha_j G(\varepsilon_t \mid 0, \lambda_j^2(t))}$$

M step :

$$\alpha_j = \frac{1}{N} \sum_t p(j \mid \varepsilon_t)$$

$$\theta^{new} = \theta^{old} + \eta \sum\nolimits_t \frac{\partial l_t}{\partial \theta}$$





Tang, H, Chiu KC and Lei Xu, ``Finite Mixture of ARMA-GARCH Model For Stock Price Prediction", Proc. of 3rd International Workshop on Computational Intelligence in Economics and Finance (CIEF'2003), North Carolina, USA, September 26-30, 2003, pp.1112-1119.

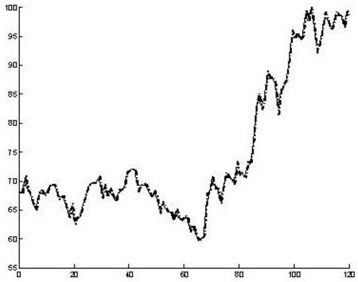


Figure 2: Prediction of CK prices with mixture of AR-GARCH.

	CK HDG	HSBC HDG
Conv. ARMA-GARCH	2.7550	2.3820
Mixture AR-GARCH	2.6025	2.2466
Mixture ARMA-GARCH	2.5524	2.1889

Table 1: A summary of mean square errors for different approaches.

H.Tang & L. Xu, CIEF03

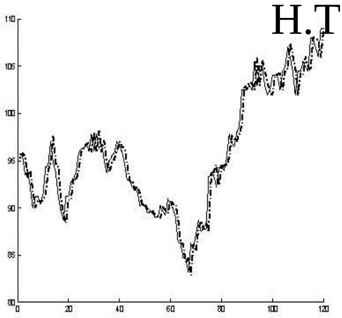


Figure 4: Prediction of HSBC prices with conventional ARMA-GARCH.

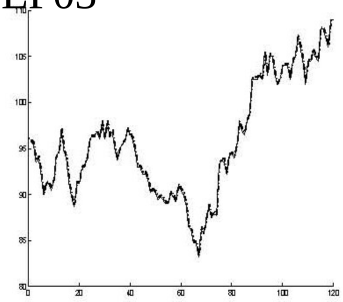


Figure 6: Prediction of HSBC prices with mixture of ARMA-GARCH.

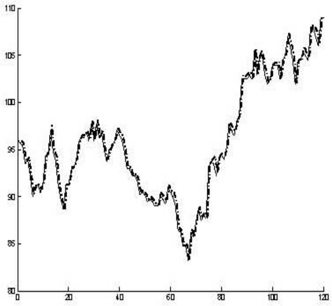


Figure 5: Prediction of HSBC prices with mixture of AR-GARCH.

	CK HDG	HSBC HDG
Conv. ARMA-GARCH	2.7550	2.3820
Mixture AR-GARCH	2.6025	2.2466
Mixture ARMA-GARCH	2.5524	2.1889

Table 1: A summary of mean square errors for different approaches.

Mixture-of-Experts of ARMA-GARCH models

$$p(z_{t+1}\,|\,\mathbf{x}_\mathbf{t}) = \sum_j p(j|\Xi_t) G\left(z_{t+1}\,|\, \sum_{\tau=1}^{q_j} a_{j.\tau} z_{t+1-\tau} + c_j + \sum_{\tau=1}^{p_j} b_{j.\tau}\varepsilon_j(t-\tau), \lambda_j^2(t) \right)$$

$$= \sum_j p(j|\Xi_t) G(\varepsilon_j(t)\,|\,0, \lambda_j^2(t))$$

$$p(j|\,\Xi_t) = \frac{\alpha_j G(\Xi_t\,|\,m_j, \Sigma_j)}{\sum_{j=1}^{k} \alpha_j G(\Xi_t\,|\,m_j, \Sigma_j)}, \quad \Xi_t = \{\mathbf{x}_\mathbf{t}, \varepsilon_j(t), others\}$$

$$z_{t+1} = \sum_{\tau=1}^{q_j} a_{j.\tau} z_{t+1-\tau} + c_j + \varepsilon_j(t) + \sum_{\tau=1}^{p_j} b_{j.\tau}\varepsilon_j(t-\tau),$$

$$P(\varepsilon_j) = G(\varepsilon_j\,|\,0, \lambda_j^2(t)), \quad j = 1, \cdots, k$$

$$\lambda_j^2(t) = \alpha_{0,j} + \sum\nolimits_{i=1}^{q} \alpha_{i,j}\varepsilon_j^2(t-i) + \sum\nolimits_{i=1}^{p} \beta_{i,j}\lambda^2(t-i)$$

Alternative Mixture-of-Experts

E step:

$$h_j(t) = \frac{\alpha_j G(\Xi_t\,|\,m_j, \Sigma_j) G(\varepsilon_t\,|\,0, \lambda_j^2(t))}{\sum_{j=1}^{k} \alpha_j G(\Xi_t\,|\,m_j, \Sigma_j) G(\varepsilon_t\,|\,0, \lambda_j^2(t))}$$

M step:

$$\max \sum_t \sum_{j=1}^{k} h_j(t) \ln[\alpha_j G(\Xi_t\,|\,m_j, \Sigma_j)]$$

$$\max \sum_t \sum_{j=1}^{k} h_j(t) \ln G(\varepsilon_t\,|\,0, \lambda_j^2(t))$$

Tang H and Lei Xu, `` MIXTURE-OF-EXPERT ARMA-GARCH MODELS FOR STOCK PRICE PREDICTION", Proc. of 2003 International Conference on Control, Automation, and Systems , Oct. 22-25, Gyeongju, KOREA, pp402-407.

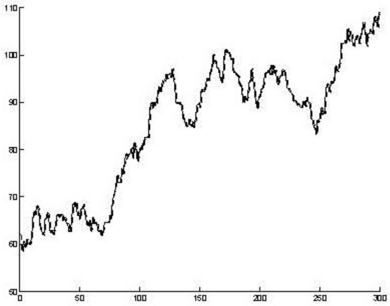


Figure 4: First step prediction of HSBC pric
Gaussian mixture ARMA-GARCH.

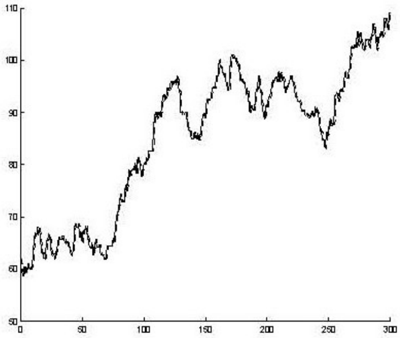


Figure 2: First step prediction of HSBC prices with conventional ARMA-GARCH.

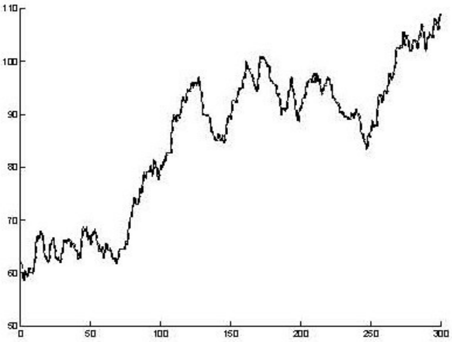


Figure 6: First step prediction of HSBC prices with mixture-of-expert ARMA-GARCH.

	CK HDG	HSBC HDG
Conventional	2.3799	2.1651
Gaussian Mixture	2.1101	1.9899
Mixture-of-Expert	2.0030	1.9225

Table 1: Mean square errors of first step prediction using different ARMA-GARCH models.

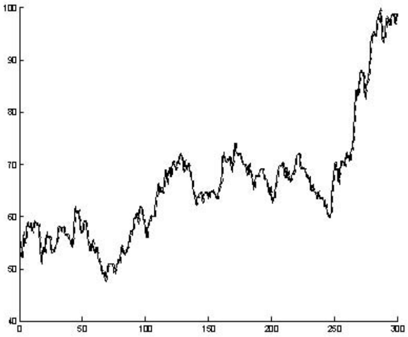


Figure 1: First step prediction of CK prices with
ventional ARMA-GARCH.

Figure 5: First step prediction of CK prices with mixture-of-expert ARMA-GARCH.

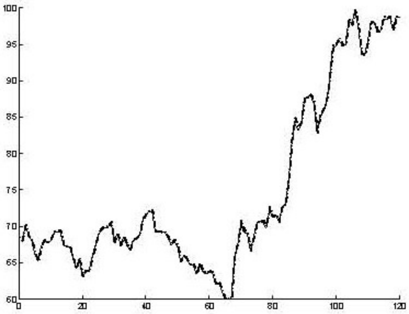


Figure 3: Prediction of CK prices with mixture of ARMA-GARCH.

	CK HDG	HSBC HDG
Conventional	2.3799	2.1651
Gaussian Mixture	2.1101	1.9899
Mixture-of-Expert	2.0030	1.9225

Table 1: Mean square errors of first step prediction using different ARMA-GARCH models.

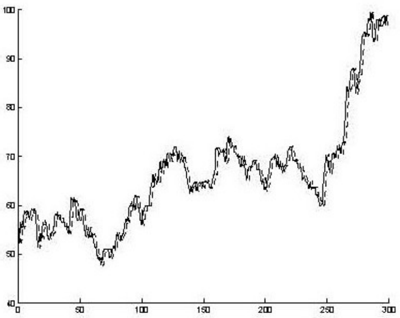


Figure 7: Second step prediction of CK pri
conventional ARMA-GARCH.

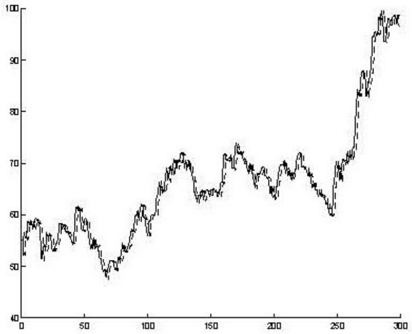


Figure 9: Second step prediction of CK prices with Gaussian mixture ARMA-GARCH.

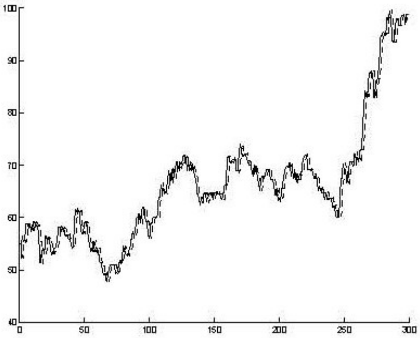


Figure 11: Second step prediction of CK prices with mixture-of-expert ARMA-GARCH.

	CK HDG	HSBC HDG
Conventional	4.9627	4.2001
Gaussian Mixture	4.8216	3.9935
Mixture-of-Expert	4.4147	3.7120

Table 2: Mean square errors of second step prediction using different ARMA-GARCH models.

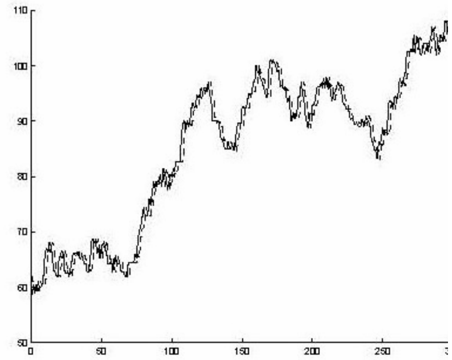


Figure 8: Second step prediction of HSBC
conventional ARMA-GARCH.

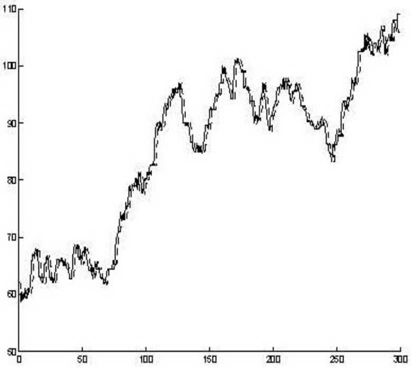


Figure 10: Second step prediction of HSBC prices with Gaussian mixture ARMA-GARCH.

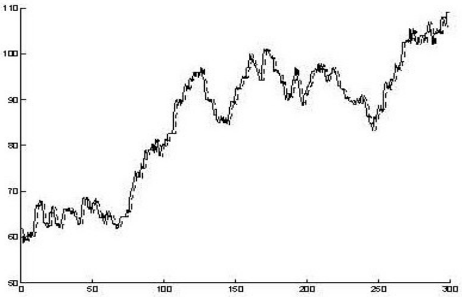


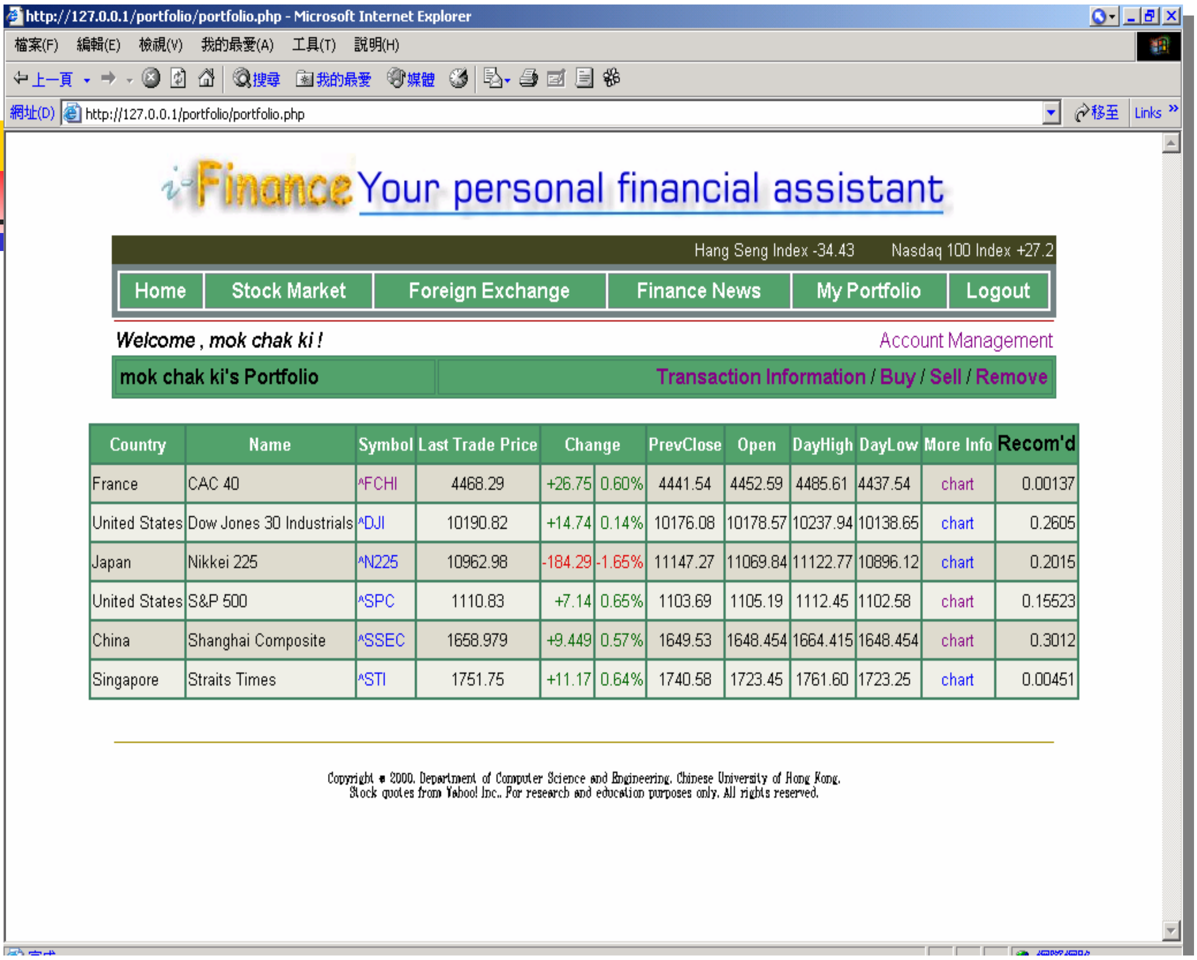
Figure 12: Second step prediction of HSBC prices with mixture-of-expert ARMA-GARCH.

	CK HDG	HSBC HDG
Conventional	4.9627	4.2001
Gaussian Mixture	4.8216	3.9935
Mixture-of-Expert	4.4147	3.7120

Table 2: Mean square errors of second step prediction using different ARMA-GARCH models.

2. Portfolio Management

- Portfolio Management by Learned Decisions
- Markowitz Portfolio, Sharpe's ratio and Downside risk
- Improved Portfolio Sharpe Ratio Maximization with Diversification
- Adaptive Portfolio Management based on Extended RBF nets and analyses of market factors



Your personal financial assistant

Hang Seng Index -34.43 Nasdaq 100 Index +27.2

Welcome , mok chak ki !

mok chak ki's Portfolio

Country	Name	Symbol	Last Trade Price	Change		PrevClose	Open	DayHigh	DayLow	More Info	Recom'd
France	CAC 40	^FCHI	4468.29	+26.75	0.60%	4441.54	4452.59	4485.61	4437.54	chart	0.00137
United States	Dow Jones 30 Industrials	^DJI	10190.82	+14.74	0.14%	10176.08	10178.57	10237.94	10138.65	chart	0.2605
Japan	Nikkei 225	^N225	10962.98	-184.29	-1.65%	11147.27	11069.84	11122.77	10896.12	chart	0.2015
United States	S&P 500	^SPC	1110.83	+7.14	0.65%	1103.69	1105.19	1112.45	1102.58	chart	0.15523
China	Shanghai Composite	^SSEC	1658.979	+9.449	0.57%	1649.53	1648.454	1664.415	1648.454	chart	0.3012
Singapore	Straits Times	^STI	1751.75	+11.17	0.64%	1740.58	1723.45	1761.60	1723.25	chart	0.00451

Portfolio Management by learning decision signals

$$I_t^i = I^{a}{}_t^{i} \bullet I^{p}{}_t$$

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

Lei Xu and Y.M. Cheung , ``Adaptive Supervised Learning Decision Networks for Trading and Portfolio Management", Journal of Computational Intelligence in Finance, Vol.5, No.6, pp11-16.

the ith foreign currency of interest.

$$\sum_{i=1}^{m} I^{a}{}_t^{i} = 1$$

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

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

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

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

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

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

$$I^i_{t-1} = [I^{a^i}_{t-1}, I^p_{t-1}]$$

with

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

and

$$I^p_{t-1} = \begin{cases} 1, & \text{if } z^i_t - z^i_{t+1} > 0 \\ 0, & \text{if } z^i_t - z^i_{t+1} = 0 \\ -1, & \text{if } z^i_t - z^i_{t+1} < 0 \end{cases} \qquad (5)$$

We assume that there is some relationship as follows

$$[I_t, I^p_t] = f[I_{t-1}, I^p_{t-1}, z^i_t, z^i_{t-1}, \ldots, z^i_{t-d+1} | 1 \le i \le m] \qquad (6)$$

Use Extended RBF nets to learn it and then make a decision

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

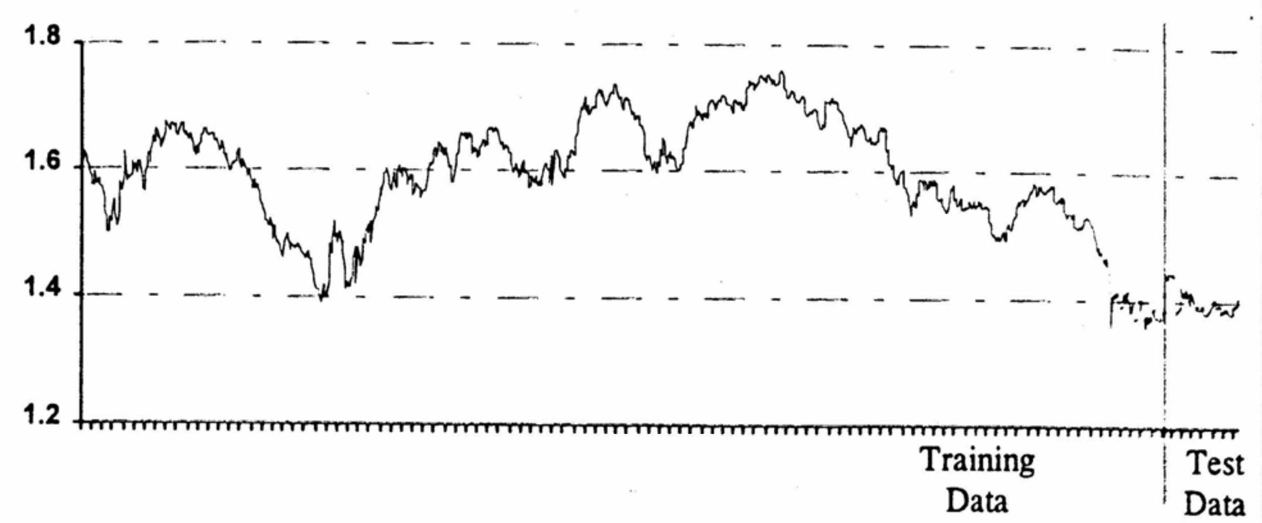


FIGURE 1. The USD-DEM rate series of 1096 data points. Each horizontal bar represents 10 data points.

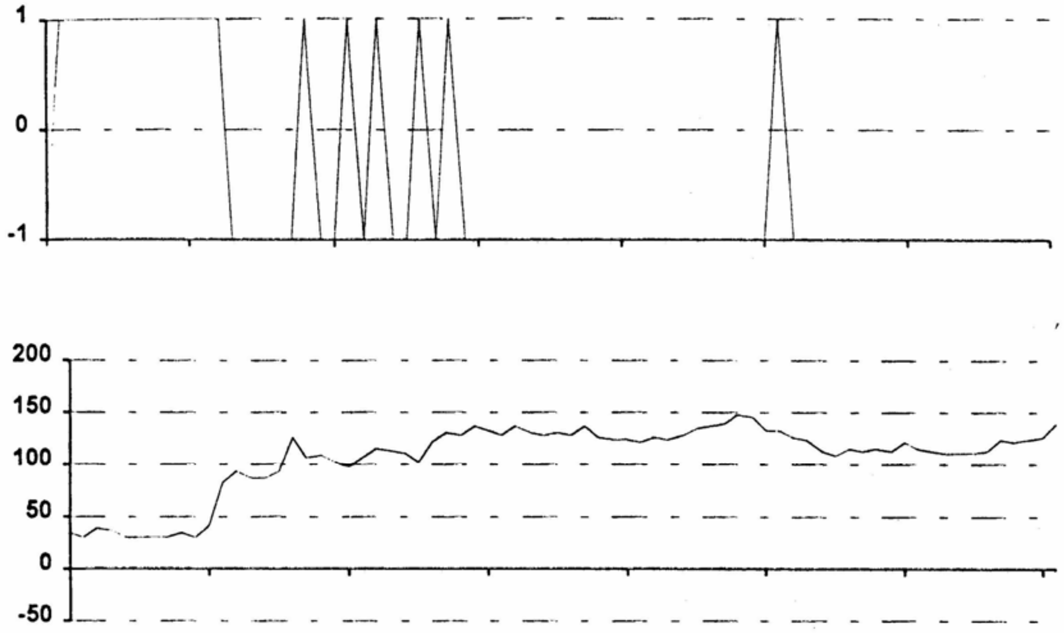


FIGURE 2. The results by *Existing ENRBF ASLD*. Upper graph: the trading signal on the test data [-1,1]. Lower graph: the profit gained (%).

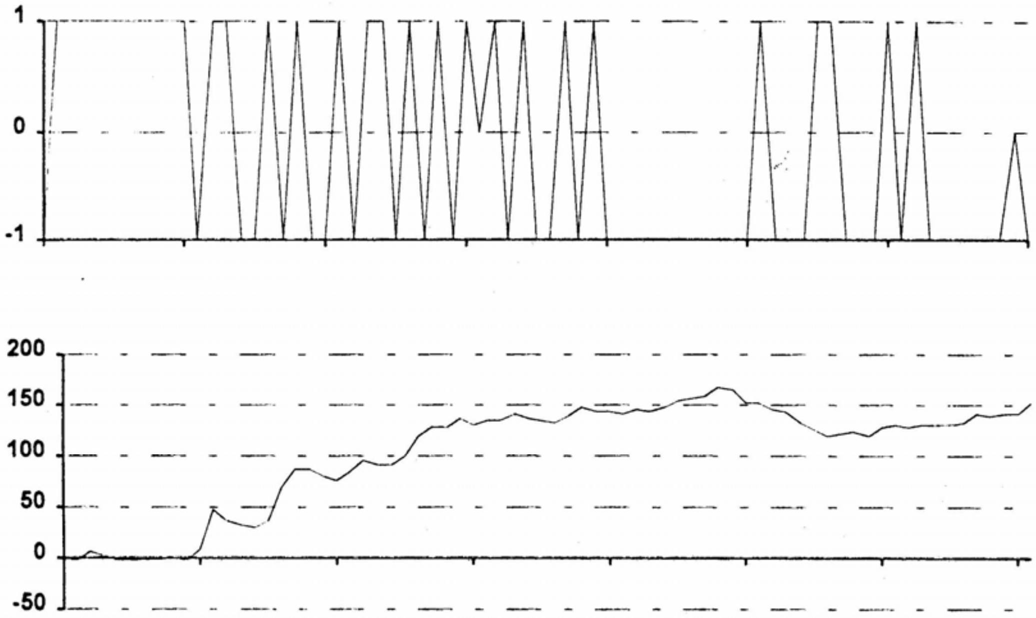


FIGURE 3. The results by *Adaptive CCL-ENRBF ASLD*. Upper graph: the trading signal on the test data [-1,1]. Lower graph: the profit gained (%).

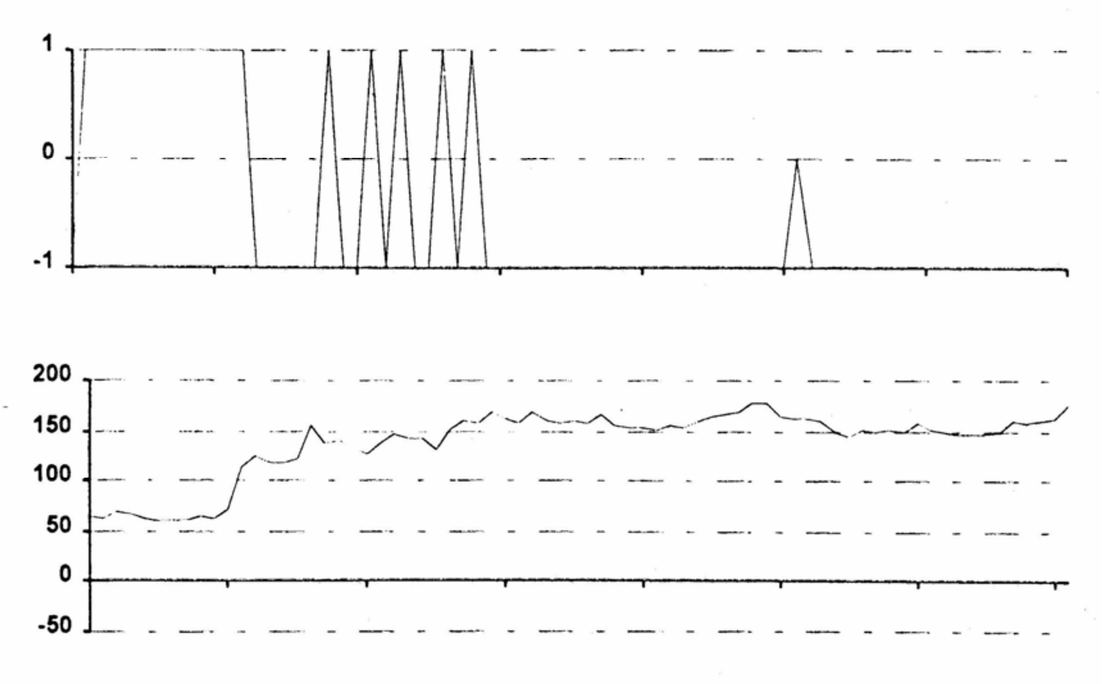


FIGURE 4. The results by *Existing ENRBF SASLD*. Upper graph: the trading signal on the test data [-1,1]. Lower graph: the profit gained (%).

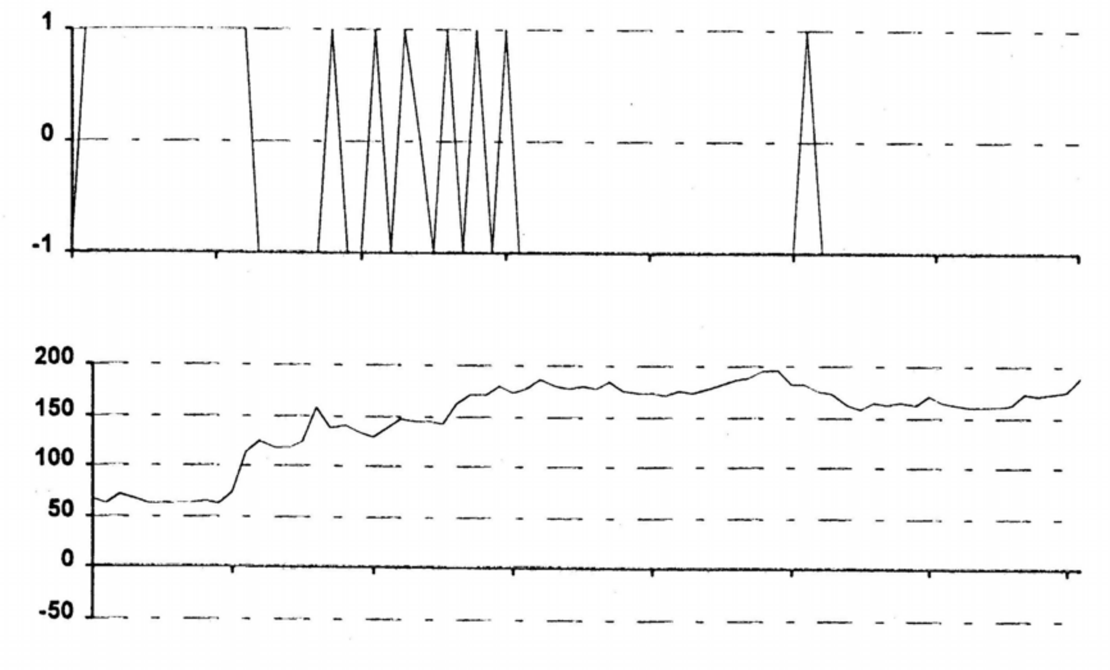


FIGURE 5. The results by *Adaptive CCL-ENRBF SASLD*. Upper graph: the trading signal on the test data [-1,1]. Lower graph: the profit gained (%).

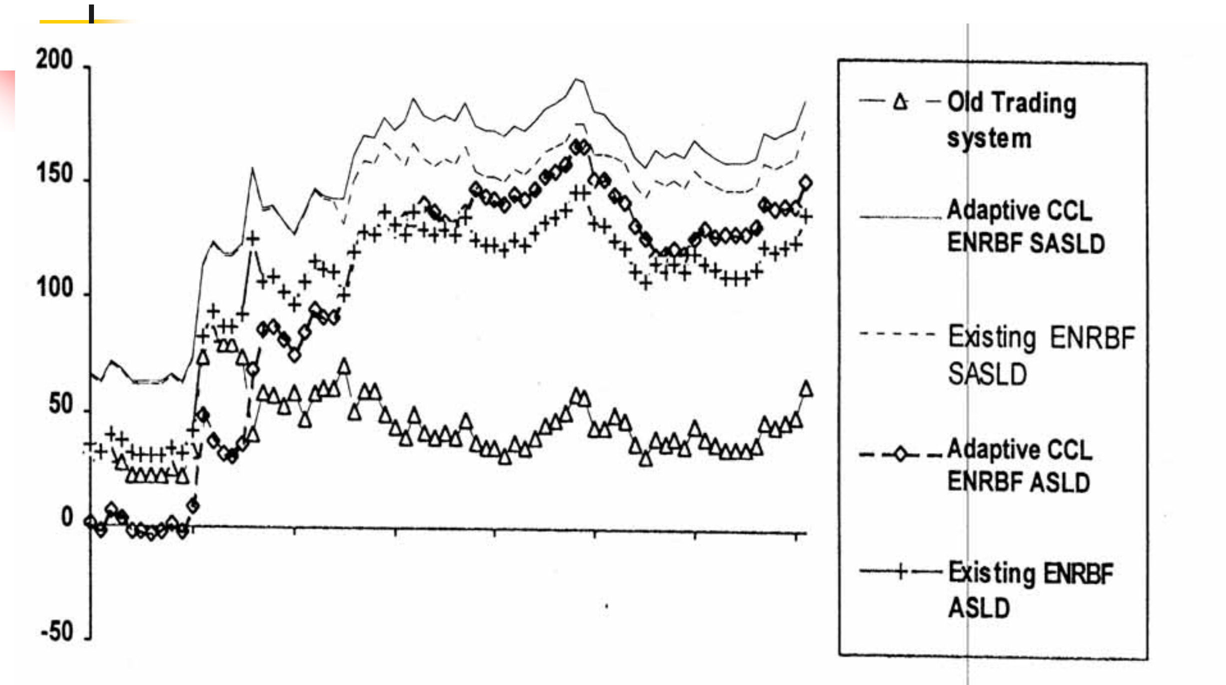


FIGURE 6. A plot of all results, along with results of an old prediction trading system by Cheung et al. [1996].

2. Portfolio Management

- **Portfolio Management by Learned Decisions**
- **Markowitz Portfolio, Sharpe's ratio and Downside risk**
- **Improved Portfolio Sharpe Ratio Maximization with Diversification**
- **Adaptive Portfolio Management based on Extended RBF nets and analyses of market factors**
- **TFA based Adaptive Portfolio Management**

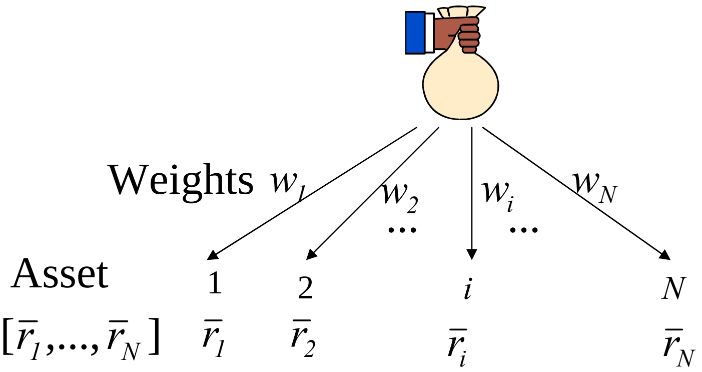
Existing Portfolio Selection Methods

- Standard Portfolio Optimization (SMPO)

  Markowitz's 1952 landmark paper "Portfolio Selection".

  - Choose portfolio weights that

$$\max \underbrace{u}_{\text{Acceptable level of risk set by user}} \underbrace{\mathbf{w}^T\bar{\mathbf{r}}}_{\text{Expected portfolio returns}} - \underbrace{\mathbf{w}^T\Sigma\mathbf{w}}_{\text{Portfolio risk}}$$



Expected returns $\bar{\mathbf{r}} = [\bar{r}_1, \ldots, \bar{r}_N]$

Sharpe's Method

- Sharpe's ratio of asset *i*

$$S_i = \frac{\bar{r}_i}{\sqrt{\mathrm{var}(r_i)}}$$

Expected return

Risk (standard deviation)

  - appropriateness of investing in asset *i*

- Single asset investment:
  - Choose the asset with highest S_i to invest in
- Portfolio investment:
  - Choey, Kang, Weigend (1997)
  - Moody & Wu (1997)

Downside risk

- Traditional risk (Markovtz 1952):

$\mathrm{var}(r_i)$ — Upside fluctuation

$\mathrm{var}(r_i)$ — Downside fluctuation

- Downside risk (Markowtz 1959, Fishburn 1977):
  - Only fluctuation below target counted as risk

$$downV_{\alpha}(G) = \int_{-\infty}^{G} (G-r)^{\alpha}\, dF(r)$$

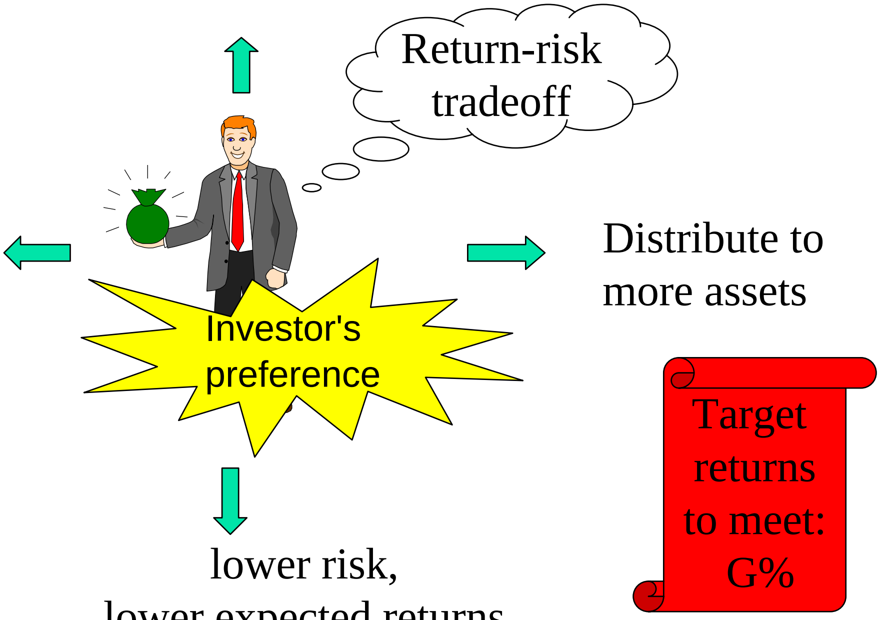
Target returns

---

2. Portfolio Management

- **Portfolio Management by Learned Decisions**
- **Markowian Portfolio, Sharpe's ratio and Downside risk**
- **Improved Portfolio Sharpe Ratio Maximization with Diversification**
- **Adaptive Portfolio Management based on Extended RBF nets and analyses of market factors**
- **TFA based Adaptive Portfolio Management**

cater for investor's preference



higher risk,
higher expected returns

Return-risk tradeoff

Concentrate on few assets

Investor's preference

Distribute to more assets

Target returns to meet: G%

lower risk,
lower expected returns

Kei Keung Hung, Yiu-ming Cheung, and **Lei Xu** , `` An Extended ASLD Trading System to Enhance Portfolio Management", *IEEE Transactions on Neural Networks, Vol. 14, No. 2*, 2003, 413-425.

- Improved Portfolio Sharpe Ratio Maximization with Diversification' (IPSRM-D)
  - Select portfolio weights according to:

$$\max_{\mathbf{w}} \frac{\overbrace{\mathbf{w}^T\bar{\mathbf{r}}}^{\text{Expected return}} + \overbrace{H}^{\text{coefficient}}\overbrace{\mathbf{w}^T\mathbf{U}\mathbf{w}}^{\text{Portfolio Upside volatility}}}{\underbrace{\mathbf{w}^T\mathbf{D}\mathbf{w}}_{\text{Portfolio Downside risk}}} + \underbrace{B}_{\text{coefficient}}\underbrace{\mathbf{w}^T([1]-\mathbf{w})}_{\text{Diversification term}}$$

$$s.t. \quad \sum_{i=1}^{N} w_i = 1, \quad w_i \geq 0$$

- Portfolio Downside Risk $\mathbf{w}^T\mathbf{D}\mathbf{w}$
  - measure fluctuation below target return G

$$\mathbf{D}=[d_{i,j}]$$

$$d_{i,j}=\int_{-\infty}^{G}\int_{-\infty}^{G}(G-r_i)^{\frac{\alpha}{2}}(G-r_j)^{\frac{\alpha}{2}}\,p(r_i,r_j)\,\mathrm{d}\,r_i\,\mathrm{d}\,r_j$$

- Portfolio Upside Volatility $\mathbf{w}^T\mathbf{U}\mathbf{w}$
  - measure fluctuation above target return G

$$\mathbf{U}=[u_{i,j}] \qquad u_{i,j}=\int_{-\infty}^{G}\int_{-\infty}^{G}(r_i-G)^{\frac{\alpha}{2}}(r_j-G)^{\frac{\alpha}{2}}\,p(r_i,r_j)\,\mathrm{d}\,r_i\,\mathrm{d}\,r_j$$

May be desired by active investor:

check performance frequently, sell assets at high point

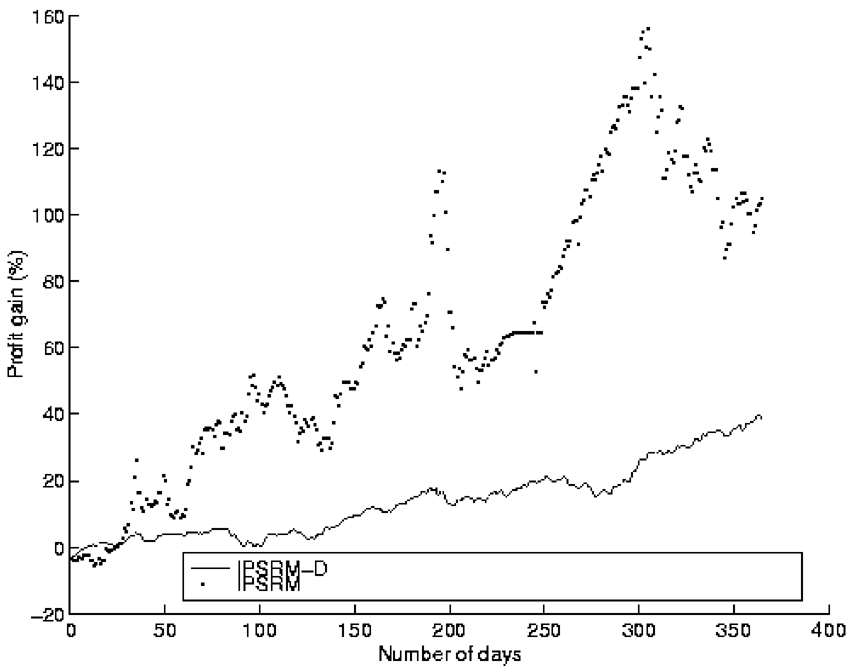
- Diversification term $\mathbf{w}^T([1]-\mathbf{w})$
  - min. when one of $\{w_i\}$ is 1 and others are 0
  - max. when all $\{w_i\}$ are equal

  - make the portfolio distribute to more assets

Experimental Demonstration (1)

- Six stocks:
  - S&P 500 Composite - Price Index (USA)
  - Hang Seng Index (Hong Kong)
  - NIKKEI 255 Stock Average (Japan)
  - Shanghai SE Composite - Price Index (China)
  - CAC 40 - Price Index (France)
  - Australia SE All Ordinary - Price Index (Australia)
- Transaction Cost 3%
- 1365 data points (1992 - 1997)
- In this experiment, $\alpha = 2, G = 0, H = B = 1$

Improved Portfolio Sharpe Ratio Maximization (IPSRM)

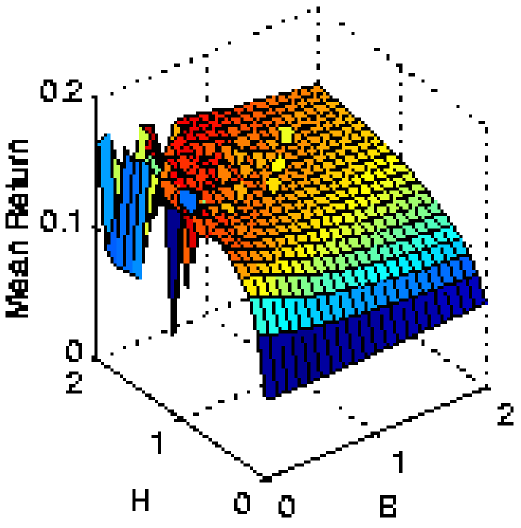


IPSRM w/wo diversification

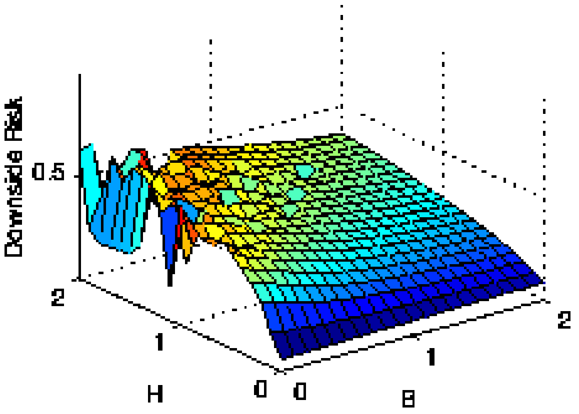
	IPSRM	IPSRM-D
No. of indexed involved	1	6
Degree of diversification	0.000	0.559
Mean return	0.239	0.140
Variance of return	6.271	1.937
IPSR	1.692	1.676
Upside volatility	0.980	0.556
Downside risk	0.721	0.416

Setting of *H* and *B*

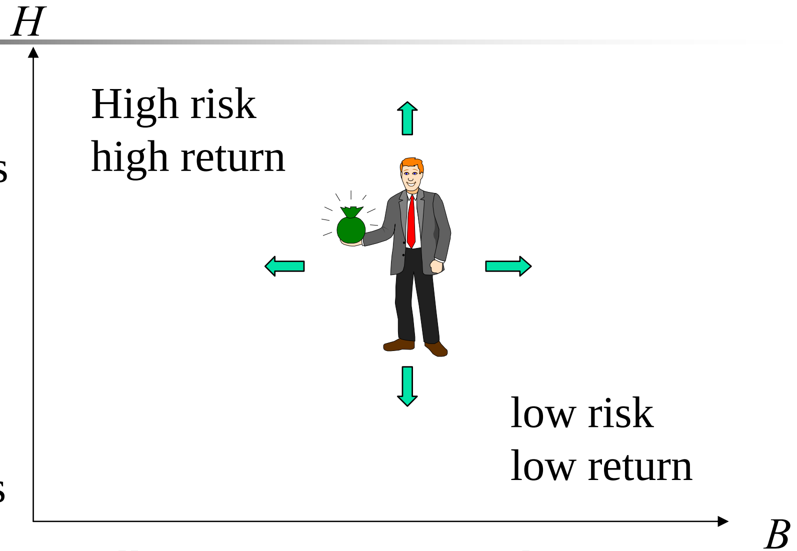
Mean Return vs. {H,B}



Downside Risk vs. {H,B}



Use of the parameters *H*, *B*



H

large *H*
more risky assets

High risk
high return

small *H*
less risky assets

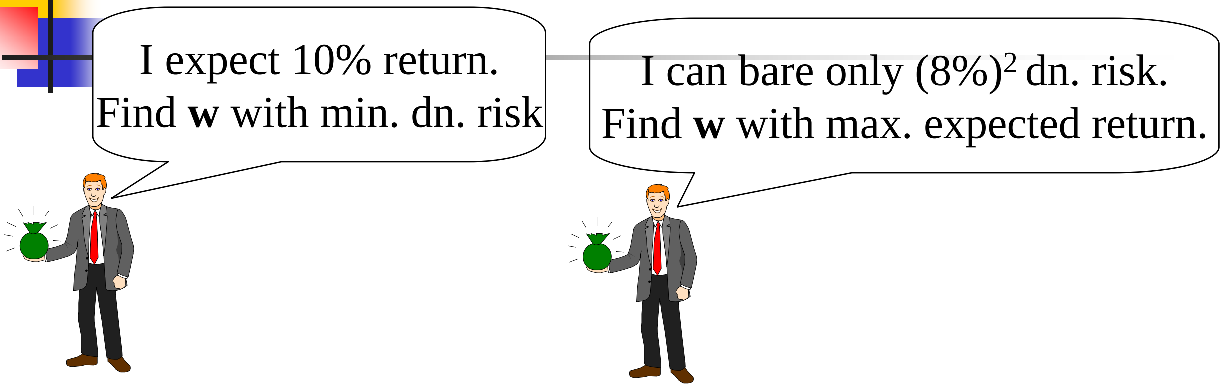
low risk
low return

B

small *B*
concentrate on
few assets

large *B*
distribute to
more assets

More convenient methods?



I expect 10% return.
Find **w** with min. dn. risk

I can bare only $(8\%)^2$ dn. risk.
Find **w** with max. expected return.

- In IPSRM-D
  - How to set parameters *H* & *B* to meet specific expected return or risk?
  - Difficult since the relationship is non-linear

The method with Control of Expected Portfolio Return

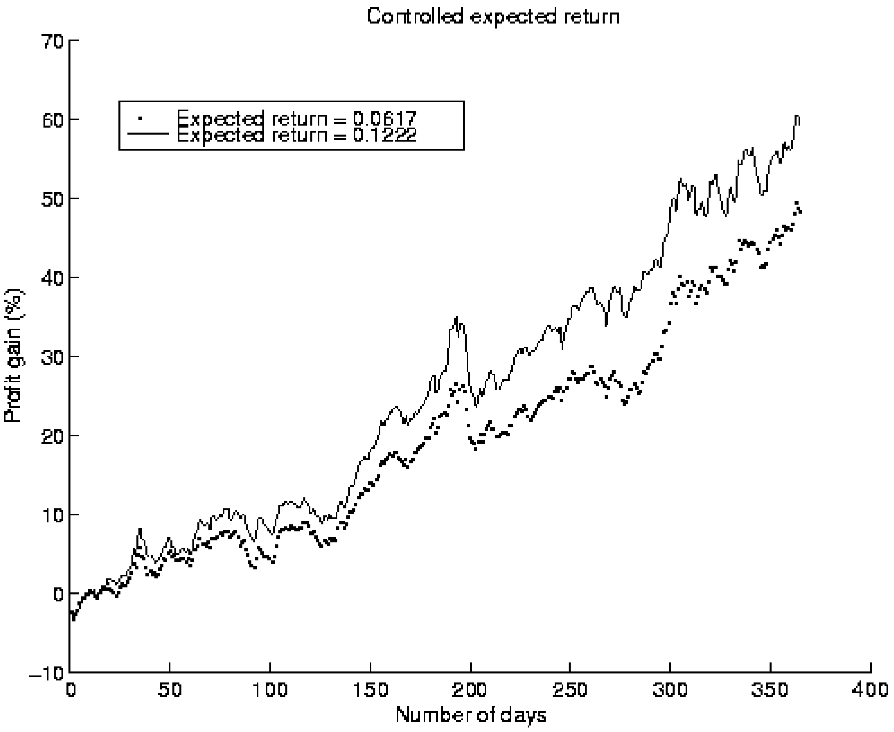
$$\max_{\mathbf{w}} \frac{r_{spec} + H\mathbf{w}^T\mathbf{U}\mathbf{w}}{\mathbf{w}^T\mathbf{D}\mathbf{w}} + B\mathbf{w}^T([1]-\mathbf{w})$$

$$\text{subject to} \quad \begin{cases} \displaystyle\sum_{i=1}^{N} w_i = 1, \quad w_i \geq 0 \\ \mathbf{w}^T\bar{\mathbf{r}} = r_{spec} \quad \text{fixed expected return} \end{cases}$$

Constrained Optimization:

by the Augmented Lagrange method

Experimental Demonstration (2)



Experimental demonstration (2)

Control of Expected Portfolio Return

	Expected return fixed at	
	0.062	0.122
Mean return	0.117	0.137
Variance of return	0.384	0.600
Improved Portfolio Sharpe Ratio	2.228	2.175
Upside Volatility	0.307	0.371
Downside Risk	0.190	0.234

The method with Control of Portfolio Downside Risk

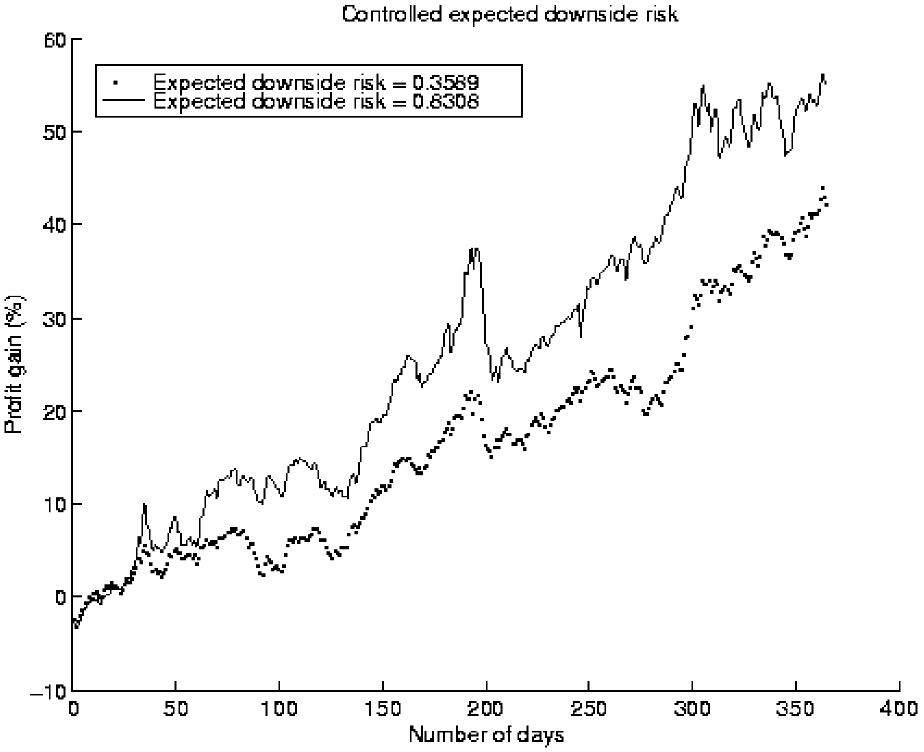
$$\max_{\mathbf{w}} \frac{\mathbf{w}^T\bar{\mathbf{r}} + H\mathbf{w}^T\mathbf{U}\mathbf{w}}{v_{spec}} + B\mathbf{w}^T([1]-\mathbf{w})$$

$$\text{subject to} \quad \begin{cases} \displaystyle\sum_{i=1}^{N} w_i = 1, \quad w_i \geq 0 \\ \mathbf{w}^T\mathbf{D}\mathbf{w} = v_{spec} \quad \text{fixed downside risk} \end{cases}$$

Constrained Optimization:

by the Augmented Lagrange method

Experimental demonstration (3)



Experimental demonstration (3)

Control of downside portfolio risk

	Expected downside risk fixed at	
	0.359	0.831
Mean return	0.105	0.132
Variance of return	0.320	0.741
Improved Portfolio Sharpe Ratio	2.193	2.027
Upside Volatility	0.282	0.388
Downside Risk	0.176	0.257

Summary

- Controlled Expected Return and Downside Risk
- Select **w** in accordance to investor's preference
- Extension of the Sharpe Ratio to portfolio case

New terms: Upside volatility, diversification term

---

2. Portfolio Management

- **Portfolio Management by Learned Decisions**
- **Markowitz Portfolio, Sharpe's ratio and Downside risk**
- **Improved Portfolio Sharpe Ratio Maximization with Diversification**
- **Adaptive Portfolio Management based on Extended RBF nets and analyses of market factors**

Adaptive Portfolio Management

Xu, L, ``BYY harmony learning, independent state space and generalized APT financial analyses ", IEEE Tr. on Neural Networks, 12 (4), 2001, 822-849.

The return of the portfolio on the t^th day is defined by :

$$R_t = (1-\alpha_t)r^f + \alpha_t \sum_{j=1}^{m} \beta_t^j x_t^j \qquad x_t^{(j)} = \frac{p_t^{(j)} - p_{t-1}^{(j)}}{p_{t-1}^{(j)}}$$

α_t : proportion of money spent on securities

- borrowing from a risk-free bond is allowed $\alpha_t > 0$

r^f : return from the risk-free bond

β_t^j : proportion of α_t spent on the j^{th} security

- short sale is not permitted $1 \geq \beta_t^j \geq 0$

$x_t^{(j)}$ and $p_t^{(j)}$ are the return and closing price for the j^th security on the t^th day respectively.

$$\boxed{R_t = (1-\alpha_t)r^f + \alpha_t \sum_{j=1}^{m} \beta_t^j x_t^j}$$

$$Sp = \frac{M(R_T)}{\sqrt{V(R_T)}}$$

: the return obtained per unit of risk

Observation based:

$$\alpha_t = e^{-\delta_t}$$

$$\beta_t^j = \frac{e^{\xi_t^j}}{\sum_{r=1}^{m} e^{\xi_t^r}}$$

where δ_t and ξ_t^j are controlled by observations

Hidden factors based:

$$\alpha_t = e^{-\delta_t}$$

$$\beta_t^j = \frac{e^{\xi_t^j}}{\sum_{r=1}^{m} e^{\xi_t^r}}$$

$$\delta_t = g(y_t, \psi)$$

$$\xi_t^j = f(y_t, \phi)$$

where δ_t and ξ_t^j are controlled by hidden market factors y_t

The exact functional form of both $g(y_t,\psi)$ and $f(y_t,\phi)$ are unknown, it can be approximated by the adaptive Extended Normalized Radial Basis Function (ENRBF) algorithm [Xu, 1998]

$$g(y_t,\psi)=\sum_{p=1}^{k}(W_p^T y_t+c_p)\varphi(\mu,\Sigma,k)$$

$$f(y_t,\phi)=\sum_{p=1}^{\hat{k}}(\hat{W}_p^T y_t+\hat{c}_p)\varphi(\hat{\mu},\hat{\Sigma},\hat{k})$$

$$\text{where } \varphi(\mu,\Sigma,k)=\frac{e^{-0.5(y_T-u_p)^T\Sigma_p^{-1}(y_T-u_p)}}{\sum_{p=1}^{k}e^{-0.5(y_T-u_p)^T\Sigma_p^{-1}(y_T-u_p)}}$$

An Adaptive Algorithm

- Use the gradient ascent approach

$$\theta^{new}=\theta^{old}+\eta\nabla_\theta S_p$$

$$\theta=\psi\cup\phi$$

$$\psi=\{u_p,\Sigma_p,W_p,c_p\}_{p=1}^{k},\qquad \psi=\{\hat{u}_p,\hat{\Sigma}_p,\hat{W}_p,\hat{c}_p\}_{p=1}^{\hat{k}}$$

Kai Chun Chiu, and Lei Xu, "Stock price and index forecasting by arbitrage pricing theory-based gaussian TFA learning", in H. Yin et al., eds., Lecture Notes in Computer Sciences, Vol.2412, pp366-371, 2002, Springer Verlag.

Detailed Updating Rules

Updating the parameter set ψ

$$u_p^{new} = u_p^{old} + \eta\ (\nabla_{\zeta_T} S_p)\varphi(\mu,\Sigma,k)\tau(\mu,\Sigma,W_p,c,k)(y_T\text{-}\mu_p)$$

$$\Sigma_p^{new} = \Sigma_p^{old} + \eta(\nabla_{\zeta_T} S_p)\varphi(\mu,\Sigma,k)\tau(\mu,\Sigma,W_p,c,k)\kappa(\mu,\Sigma)$$

$$W_p^{new} = W_p^{old} + \eta(\nabla_{\zeta_T} S_p)\varphi(\mu,\Sigma,k)y_T$$

$$c_p^{new} = c_p^{old} + \eta(\nabla_{\zeta_T} S_p)\varphi(\mu,\Sigma,k)$$

Updating the parameter set ϕ

$$\hat{u}_p^{new} = \hat{u}_p^{old} + \hat{\eta}\ (\nabla_{\xi_T^{(j)}} S_p)\varphi(\hat{\mu},\hat{\Sigma},\hat{k})\tau(\hat{\mu},\hat{\Sigma},\hat{W}_{p,r},\hat{c},\hat{k})(y_T\text{-}\hat{\mu}_p)$$

$$\hat{\Sigma}_p^{new} = \hat{\Sigma}_p^{old} + \eta(\nabla_{\xi_T^{(j)}} S_p)\varphi(\hat{\mu},\hat{\Sigma},\hat{k})\tau(\hat{\mu},\hat{\Sigma},\hat{W}_{p,r},\hat{c},\hat{k})\kappa(\hat{\mu},\hat{\Sigma})$$

$$W_{p,r}^{new} = W_{p,r}^{old} + \eta(\nabla_{\xi_T^{(j)}} S_p)\varphi(\hat{\mu},\hat{\Sigma},\hat{k})y_T$$

$$c_{p,r}^{new} = c_{p,r}^{old} + \eta(\nabla_{\xi_T^{(j)}} S_p)\varphi(\hat{\mu},\hat{\Sigma},\hat{k})$$

where η and $\hat{\eta}$ are learning rates,

$$M(R_T) = \frac{1}{T}\sum_{t=1}^{T} R_t, \qquad V(R_T) = \frac{1}{T}\sum_{t=1}^{T}[R_t - M(R_T)]^2$$

$$\nabla_{\zeta_T} S_p = \frac{\{V(R_T) - M(R_T)[R_T - M(R_T)](1 - \frac{1}{T})\}}{T\sqrt{[V(R_T)]^3}} \left(\frac{\sum_{j=1}^{m} e^{\xi_T^{(j)}} x_T^{(j)}}{\sum_{j=1}^{m} e^{\xi_T^{(j)}}} - r_t^f \right) e^{\zeta_T},$$

$$\nabla_{\xi_T^{(j)}} S_p = \frac{\{V(R_T) - M(R_T)[R_T - M(R_T)](1 - \frac{1}{T})\} e^{\zeta_T} x_T^{(j)} \left(\sum_{r=1}^{m} e^{\xi_t^{(r)}} - e^{\xi_T^{(j)}} \right) e^{\xi_T^{(j)}}}{T\sqrt{[V(R_T)]^3} \left(\sum_{r=1}^{m} e^{\xi_t^{(r)}} \right)^2},$$

$$\varphi(\mu, \Sigma, k) = \frac{e^{-0.5(y_T - u_p)^T \Sigma_p^{-1} (y_T - u_p)}}{\sum_{p=1}^{k} e^{-0.5(y_T - u_p)^T \Sigma_p^{-1} (y_T - u_p)}},$$

$$\kappa(\mu, \Sigma) = \Sigma_p^{-1}(y_T - u_p)(y_T - u_p)^T \Sigma_p^{-1} - 0.5\mathrm{diag}\left[\Sigma_p^{-1}(y_T - u_p)(y_T - u_p)^T \Sigma_p^{-1}\right],$$

$$\tau(\mu, \Sigma, W_p, c, k) = \frac{\left(W_p^T y_T + c_p\right) - \sum_{p=1}^{k} \left(W_p^T y_T + c_p\right)\varphi(\mu, \Sigma, k)}{\sum_{p=1}^{k} e^{-0.5(y_T - u_p)^T \Sigma_p^{-1} (y_T - u_p)}}$$

Simulation Result and Performance Evaluation

Data Considerations

- Experiments based on the interest rate and stock prices of Hong Kong stock market
- Data obtained from 1 Nov 2001 to 11/11/2002
- The performance was studied for 180 trading days.
- Eight stocks were selected to form the portfolios:

0001 CHECUNG KONG	0002 CLP HOLDINGS	0003 HK & CHINA GAS	0004 WHARF
0005 HSBC HOLDINGS	0008 PCCW	0992 LEGEND GROUP	1038 CKI HOLDINGS

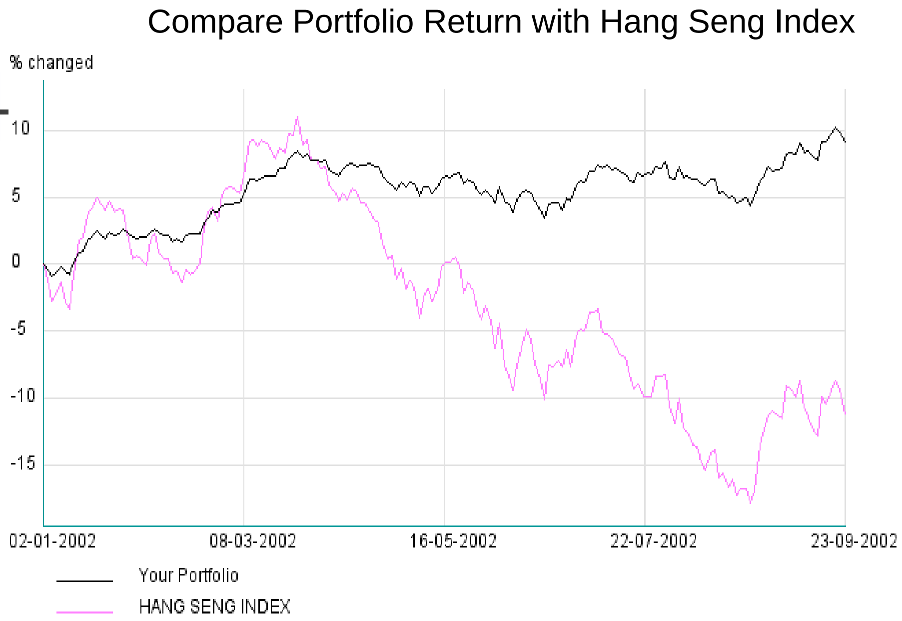
K.C., Chiu and Lei Xu, " Stock forecasting by ARCH driven gaussian TFA and alternative mixture experts models", Proc. of 3rd International Workshop on Computational Intelligence in Economics and Finance (CIEF'2003), paper CIEF3-80, NorthCarolina, USA, September 26-30, 2003, pp 1096 -1099.

Evaluate the performance of the portfolio management system in 4 different scenarios, where transactions were not in lots

The four scenarios are:

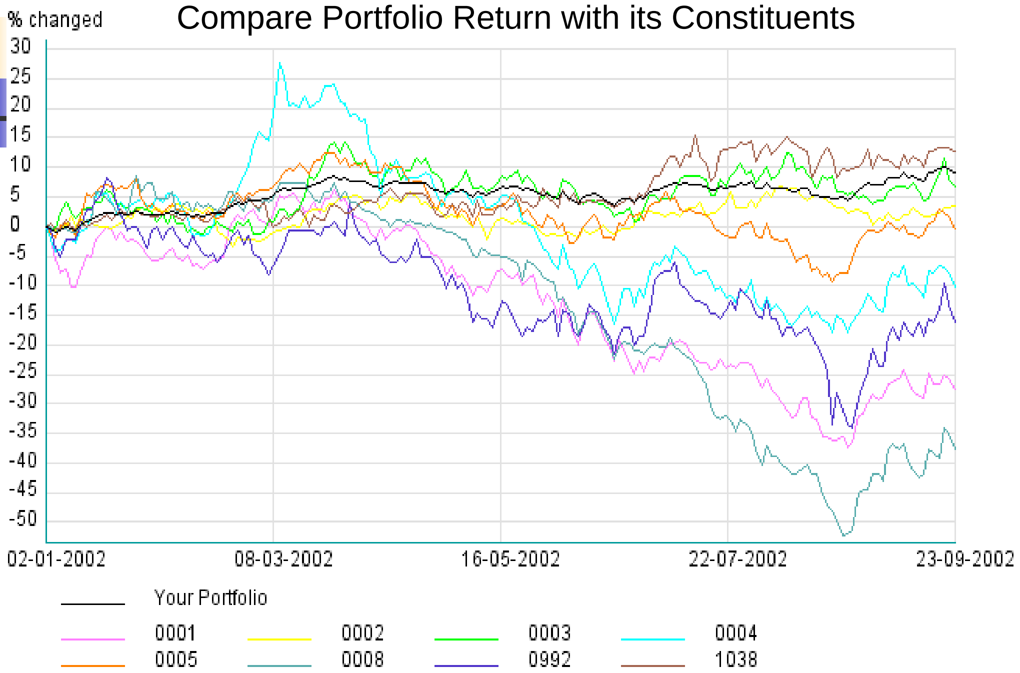
Experiment 1	**No Transaction Cost and Short Sale Not Permitted**
Experiment 2	**Has Transaction Cost, Short Sale Not Permitted**
Experiment 3	**No Transaction Cost, Short Sale Permitted**
Experiment 4	**Has Transaction Cost, Short Sale Permitted**

No Transaction Cost, Short Sale Not Permitted



	Mean	Standard Deviation	Max	Min	Sharpe Ratio
The Portfolio	***1.0541***	***0.0235***	***1.1020***	***0.9910***	***44.8553***
Hang Seng Index	0.9771	0.0727	1.1099	0.8211	13.4402

No Transaction Cost, Short Sale Not Permitted


Compare Portfolio Return with its Constituents

0001 CHECUNG KONG	0002 CLP HOLDINGS	0003 HK & CHINA GAS	0004 WHARF
0005 HSBC HOLDINGS	0008 PCCW	0992 LEGEND GROUP	1038 CKI HOLDINGS

No Transaction Cost, Short Sale Not Permitted

Compare Portfolio Return with its Constituents

	Mean	Standard Deviation	Max	Min	Sharpe Ratio
The Portfolio	*1.0541*	*0.0235*	*1.1020*	*0.9910*	*44.8553*
0001	**0.8712**	**0.1213**	**1.0621**	**0.6262**	**7.1822**
0002	**1.0188**	**0.0222**	**1.0680**	**0.9660**	**45.8919**
0003	**1.0572**	**0.0361**	**1.1413**	**0.9859**	**29.2853**
0004	**0.9938**	**0.1109**	**1.2764**	**0.8205**	**8.9612**
0005	**1.0257**	**0.0464**	**1.1239**	**0.9078**	**22.1056**
0008	**0.8678**	**0.1855**	**1.0864**	**0.4764**	**4.6782**
0992	**0.8985**	**0.0854**	**1.0815**	**0.6593**	**10.5211**
1038	**1.0591**	**0.0450**	**1.1545**	**0.9959**	**23.5356**

With Transaction Cost, Short Sale Not Permitted

every change in β_t^j involves a transaction of which a transaction cost

$$c_t = -\alpha_t \sum_{j=1}^{m} r_c \left| \beta_t^j - \beta_{t-1}^j \right| \left(1 + x_t^j\right)$$

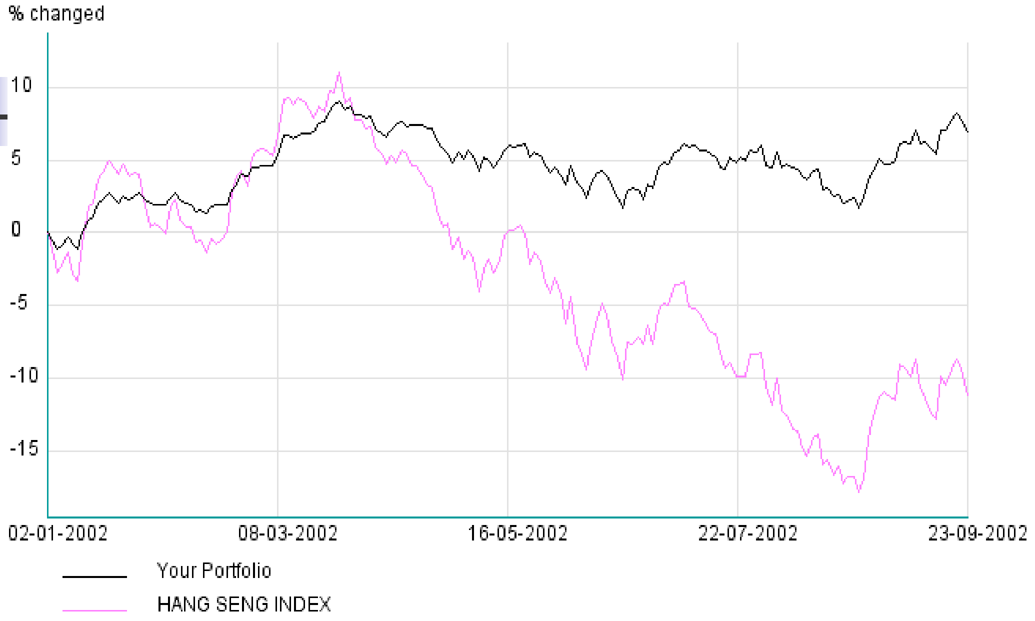
,where r_c is the rate of transaction cost

$$R_t = \left(1 - \alpha_t\right) r^f + \alpha_t \sum\nolimits_{j=1}^{m} \left(\beta_t^j x_t^j - r_c \sum_{j=1}^{m} r_c \left| \beta_t^j - \beta_{t-1}^j \right| \left(1 + x_t^j\right) \right)$$

$$\frac{dSp}{d\xi_t^j} = \left[V\left(R_T\right) - M\left(R_T\right)\left[R_T - M\left(R_T\right)\right]\right] e^{\xi_T^j} \left[x_t^j - r_c sign\left(e^{\xi_T^j} - e^{\xi_T^{j-1}}\right)\right]$$

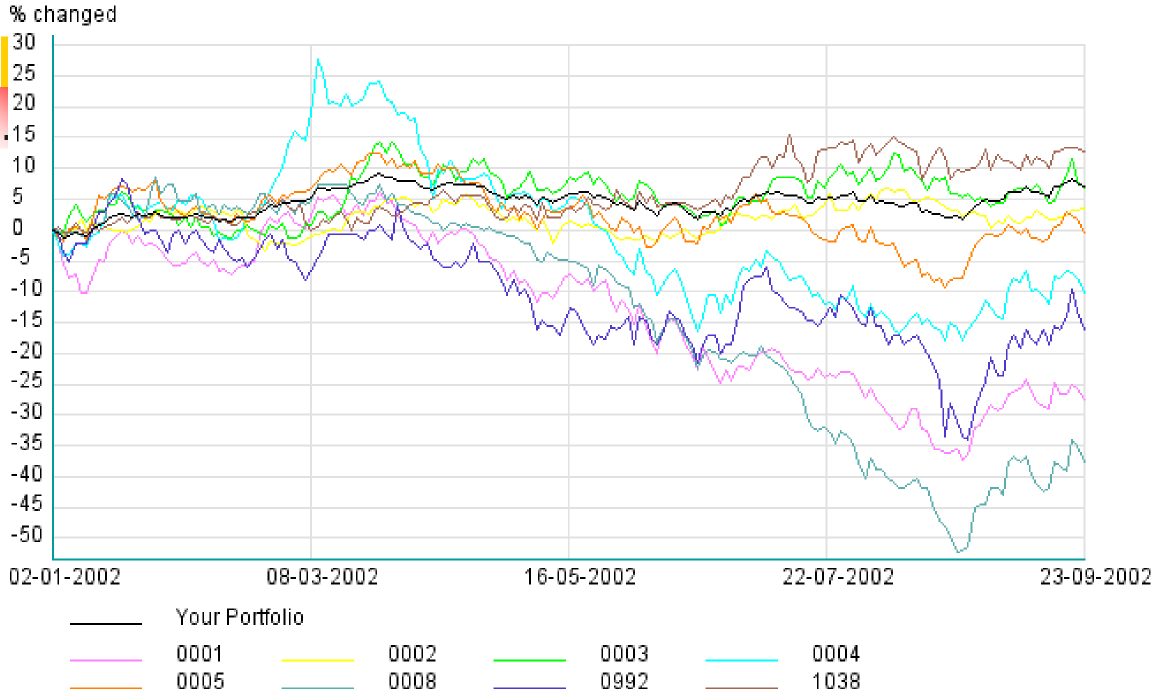
$$\cdot \left(\sum_{r=1}^{m} e^{\xi_T^r} - e^{\xi_T^j} \right) \frac{e^{\xi_T^j}}{T \sqrt{V\left(R_T\right)^3} \left(\sum\nolimits_{r=1}^{m} e^{\xi_T^r} \right)^2}$$

Compare Portfolio Return with Hang Seng Index



	Mean	Standard Deviation	Max	Min	Sharpe Ratio
The Portfolio	*1.0452*	*0.0220*	*1.0911*	*0.9886*	*47.5091*
Hang Seng Index	0.9771	0.0727	1.1099	0.8211	13.4402

Compare Portfolio Return with its Constituents



0001 CHECUNG KONG	**0002 CLP HOLDINGS**	**0003 HK & CHINA GAS**	**0004 WHARF**
0005 HSBC HOLDINGS	**0008 PCCW**	**0992 LEGEND GROUP**	**1038 CKI HOLDINGS**

Compare Portfolio Return with its Constituents

	Mean	Standard Deviation	Max	Min	Sharpe Ratio
The Portfolio	***1.0452***	***0.0220***	***1.0911***	***0.9886***	***47.5091***
0001	0.8712	0.1213	1.0621	0.6262	7.1822
0002	1.0188	0.0222	1.0680	0.9660	45.8919
0003	1.0572	0.0361	1.1413	0.9859	29.2853
0004	0.9938	0.1109	1.2764	0.8205	8.9612
0005	1.0257	0.0464	1.1239	0.9078	22.1056
0008	0.8678	0.1855	1.0864	0.4764	4.6782
0992	0.8985	0.0854	1.0815	0.6593	10.5211
1038	1.0591	0.0450	1.1545	0.9959	23.5356

With Transaction Cost, Short Sale Not Permitted

	Mean	Standard Deviation	Max	Min	Sharpe Ratio
Experiment 1	1.0541	0.0235	1.1020	0.9910	44.8553
Experiment 2	1.0452	0.0220	1.0911	0.9886	47.5091

- all the **return related attributes decreased**

  transaction cost was charged for every transaction

- The **standard deviation** and **Sharpe Ratio better**
- Still **better than the performance of Hang Seng Index**

No Transaction Cost, Short Sale Permitted

$$\alpha_t = \delta_t$$

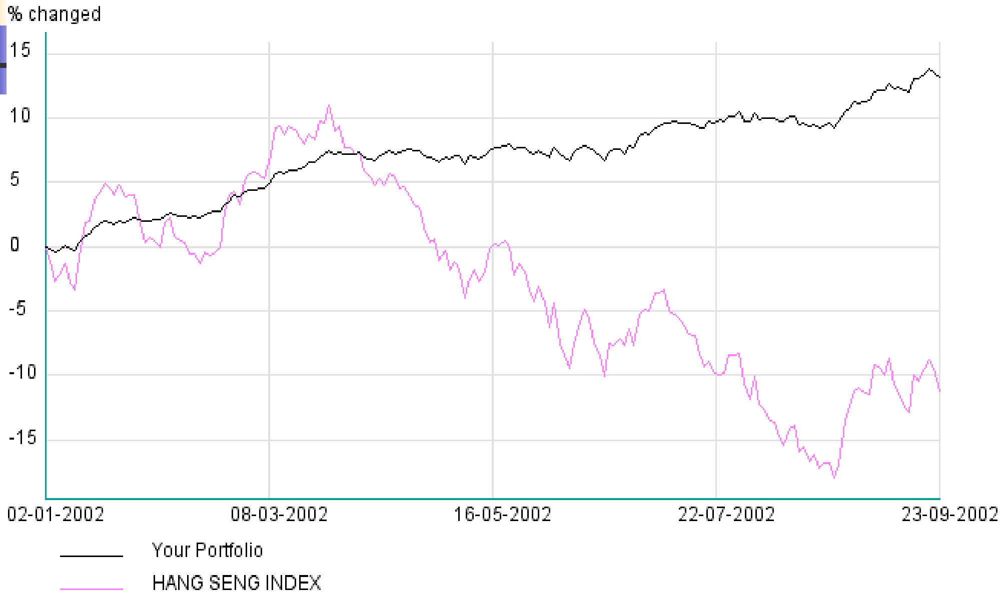
$$\beta_t^j = \frac{\xi_t^j}{\sum_{r=1}^m \xi_t^r}$$

$$\frac{\partial Sp}{\partial \delta_T} = \frac{[V(R_T) - M(R_T)(R_T - M(R_T))]}{T\sqrt{V(R_T)^3}} \left(\frac{\sum_{r=1}^m e^{\xi_T^r} x_T^r}{\sum_{r=1}^m e^{\xi_T^r}} - r^f \right)$$

$$\frac{\partial Sp}{\partial \xi_T^j} = \frac{[V(R_T) - M(R_T)(R_T - M(R_T))] e^{\delta_T} x_T^j \left(\sum_{r=1}^m e^{\xi_T^r} - e^{\xi_T^j} \right)}{T\sqrt{V(R_T)^3} \left(\sum_{r=1}^m e^{\xi_T^r} \right)^2}$$

No Transaction Cost, Short Sale Permitted

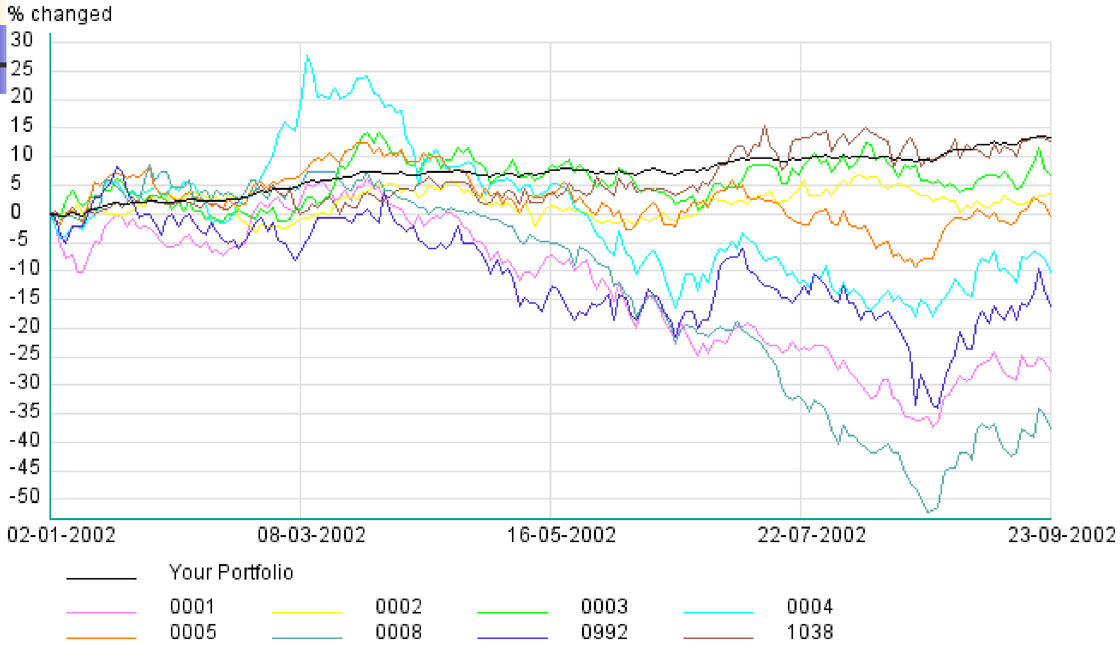
Compare Portfolio Return with Hang Seng Index



	Mean	Standard Deviation	Max	Min	Sharpe Ratio
The Portfolio	*1.0700*	*0.0336*	*1.1376*	*0.9953*	*31.8452*
Hang Seng Index	**0.9771**	**0.0727**	**1.1099**	**0.8211**	**13.4402**

No Transaction Cost, Short Sale Permitted

Compare Portfolio Return with its Constituents



0001 CHECUNG KONG	**0002 CLP HOLDINGS**	**0003 HK & CHINA GAS**	**0004 WHARF**
0005 HSBC HOLDINGS	**0008 PCCW**	**0992 LEGEND GROUP**	**1038 CKI HOLDINGS**

No Transaction Cost, Short Sale Permitted

Compare Portfolio Return with its Constituents

	Mean	Standard Deviation	Max	Min	Sharpe Ratio
Your Portfolio	*1.0700*	*0.0336*	*1.1376*	*0.9953*	*31.8452*
0001	**0.8712**	**0.1213**	**1.0621**	**0.6262**	**7.1822**
0002	**1.0188**	**0.0222**	**1.0680**	**0.9660**	**45.8919**
0003	**1.0572**	**0.0361**	**1.1413**	**0.9859**	**29.2853**
0004	**0.9938**	**0.1109**	**1.2764**	**0.8205**	**8.9612**
0005	**1.0257**	**0.0464**	**1.1239**	**0.9078**	**22.1056**
0008	**0.8678**	**0.1855**	**1.0864**	**0.4764**	**4.6782**
0992	**0.8985**	**0.0854**	**1.0815**	**0.6593**	**10.5211**
1038	**1.0591**	**0.0450**	**1.1545**	**0.9959**	**23.5356**

No Transaction Cost, Short Sale Permitted

	Mean	Standard Deviation	Max	Min	Sharpe Ratio
Experiment 2.1	1.0541	0.0235	1.1020	0.9910	44.8553
Experiment 2.2	1.0452	0.0220	1.0911	0.9886	47.5091
Experiment 2.3	1.0700	0.0336	1.1376	0.9953	31.8452

- Short sales were permitted → the portfolio was still able to generate money from stock market even though the stock prices declined

  The portfolio was doing very well

- Hang Seng Index decreased for more than 10%

  However, the portfolio increased for about 14%

- Short sales brought return to the portfolio, but also brought risk

  The risk was still low

With Transaction Cost, Short Sale Permitted

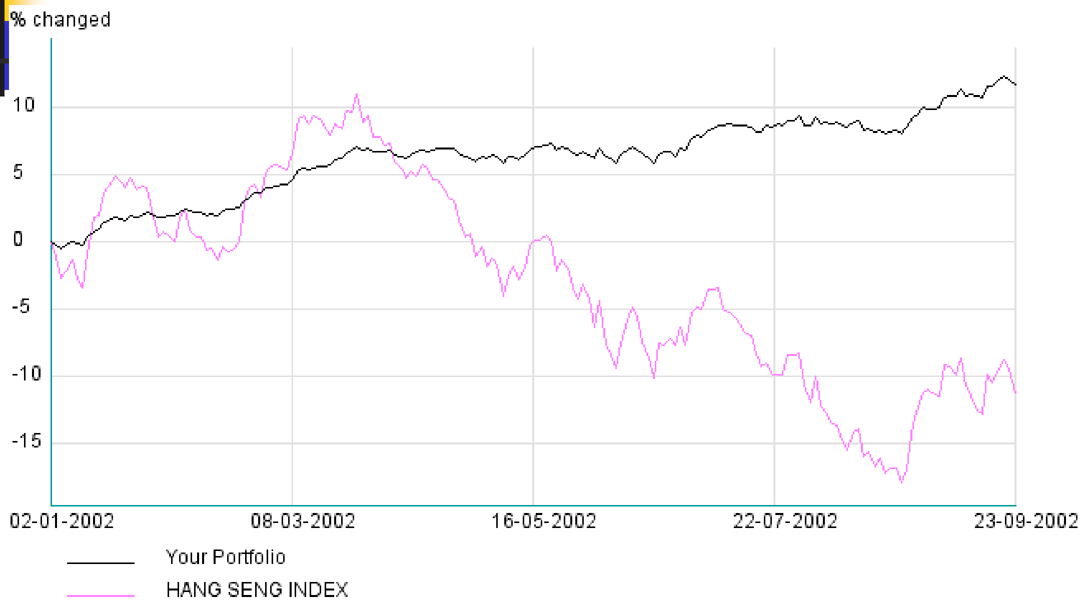
$$R_t = (1-\alpha_t) r^f + \alpha_t \sum\nolimits_{j=1}^{m} \left(\beta_t^j x_t^j - r_c \sum_{j=1}^{m} r_c \left| \beta_t^j - \beta_{t-1}^j \right| (1 + x_t^j) \right)$$

$$\frac{\partial Sp}{\partial \delta_T} = \frac{[V(R_T) - M(R_T)(R_T - M(R_T))]}{T\sqrt{V(R_T)^3}} \left(\frac{\sum\nolimits_{r=1}^{m} e^{\xi_T^r} x_T^r}{\sum\nolimits_{r=1}^{m} e^{\xi_T^r}} - r^f \right)$$

$$\frac{dSp}{d\xi_t^j} = \left[V(R_T) - M(R_T) \left[R_T - M(R_T) \right] \right] \xi_T^j \left[x_t^j - r_c sign\left(\xi_T^j - \xi_T^{j-1} \right) \right] \cdot \frac{\sum\limits_{r=1}^{m} \xi_T^r - \xi_T^j}{T\sqrt{V(R_T)^3} \left(\sum\nolimits_{r=1}^{m} \xi_T^r \right)^2}$$

With Transaction Cost, Short Sale Permitted

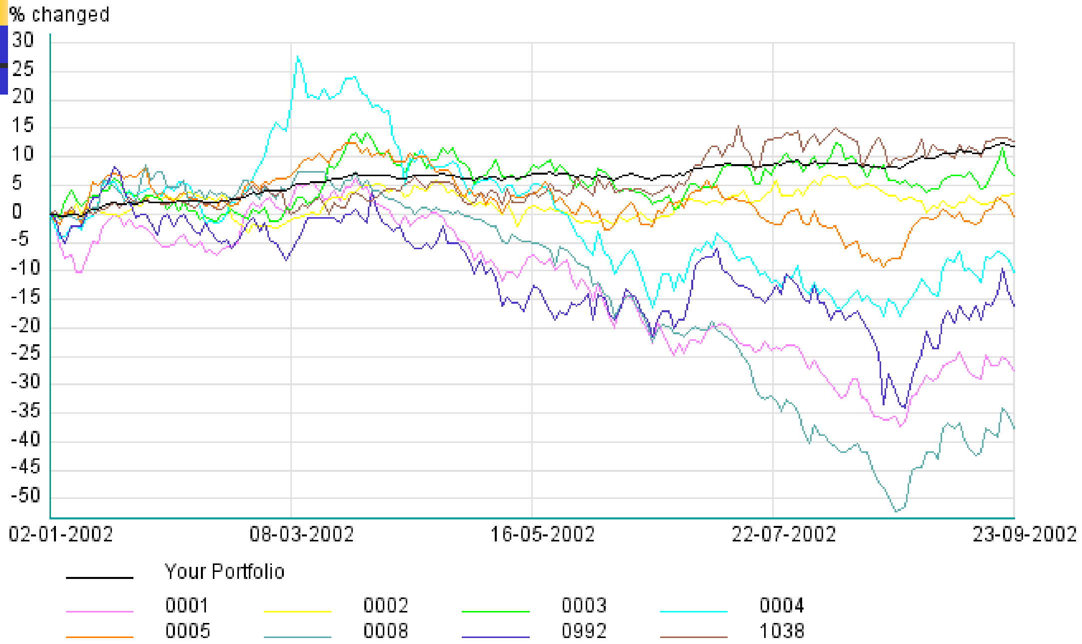
Compare Portfolio Return with Hang Seng Index



	Mean	Standard Deviation	Max	Min	Sharpe Ratio
The Portfolio	***1.0629***	***0.0296***	***1.1231***	***0.9952***	***35.9088***
Hang Seng Index	0.9771	0.0727	1.1099	0.8211	13.4402

With Transaction Cost, Short Sale Permitted

Compare Portfolio Return with its Constituents



0001 CHECUNG KONG	**0002 CLP HOLDINGS**	**0003 HK & CHINA GAS**	**0004 WHARF**
0005 HSBC HOLDINGS	**0008 PCCW**	**0992 LEGEND GROUP**	**1038 CKI HOLDINGS**

With Transaction Cost, Short Sale Permitted

Compare Portfolio Return with its Constituents

	Mean	Standard Deviation	Max	Min	Sharpe Ratio
The Portfolio	*1.0629*	*0.0296*	*1.1231*	*0.9952*	*35.9088*
0001	0.8712	0.1213	1.0621	0.6262	7.1822
0002	1.0188	0.0222	1.0680	0.9660	45.8919
0003	1.0572	0.0361	1.1413	0.9859	29.2853
0004	0.9938	0.1109	1.2764	0.8205	8.9612
0005	1.0257	0.0464	1.1239	0.9078	22.1056
0008	0.8678	0.1855	1.0864	0.4764	4.6782
0992	0.8985	0.0854	1.0815	0.6593	10.5211
1038	1.0591	0.0450	1.1545	0.9959	23.5356

With Transaction Cost, Short Sale Permitted

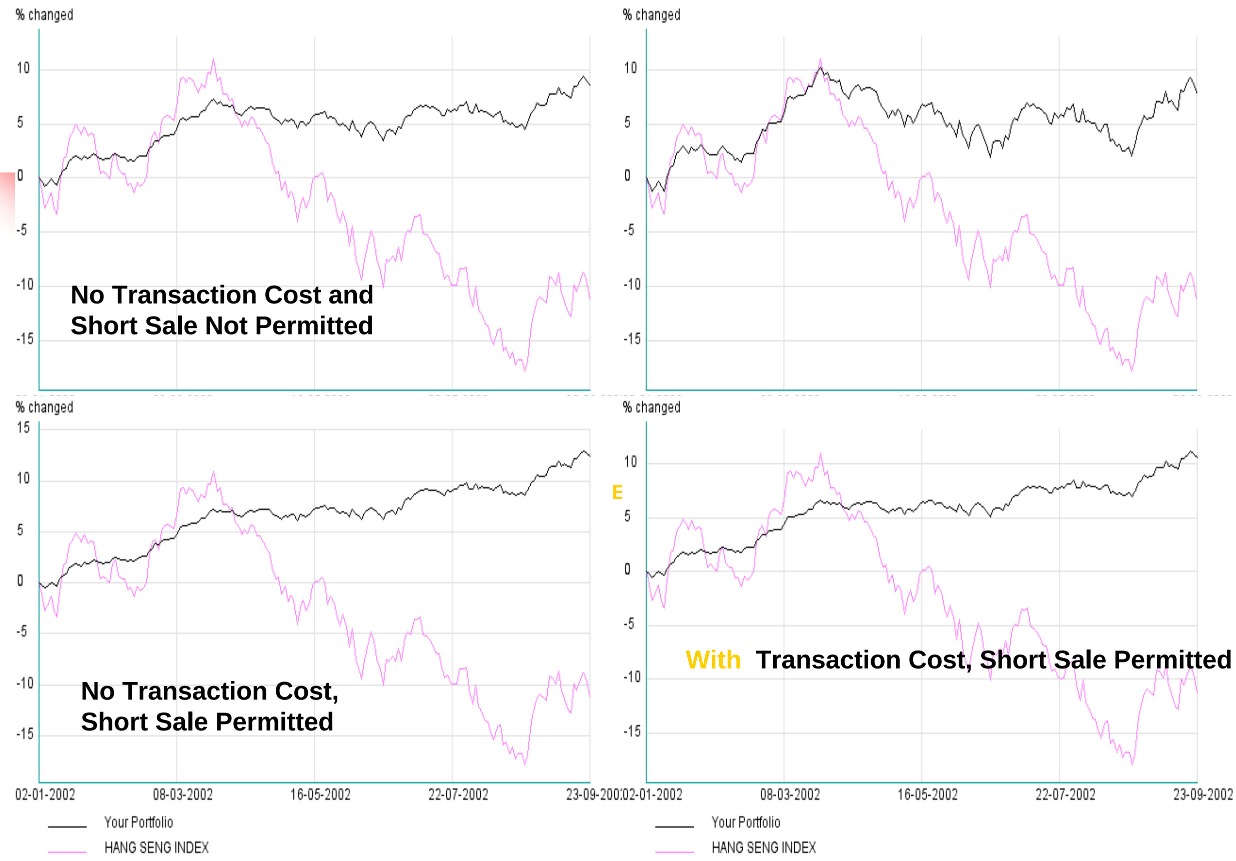
	Mean	Standard Deviation	Max	Min	Sharpe Ratio
Experiment 1	1.0541	0.0235	1.1020	0.9910	44.8553
Experiment 2	1.0452	0.0220	1.0911	0.9886	47.5091
Experiment 3	1.0700	0.0336	1.1376	0.9953	31.8452
Experiment 4	1.0629	0.0296	1.1231	0.9952	35.9088

Evaluate the performance of the portfolio management system in 4 different scenarios, where transactions were in lots

Stock Code	Company Name	No. of Shares per Board Lot
0001	CHEUNG KONG	1000
0002	CLP HOLDINGS	500
0003	HK & CHINA GAS	1000
0004	WHARF HOLDINGS	1000
0005	HSBC HOLDINGS	400
0008	PCCW	1000
0992	LEGEND GROUP	2000
1038	CKI HOLDINGS	1000

The four scenarios are:

Experiment 1	**No Transaction Cost and Short Sale Not Permitted**
Experiment 2	**Has Transaction Cost, Short Sale Not Permitted**
Experiment 3	**No Transaction Cost, Short Sale Permitted**
Experiment 4	**Has Transaction Cost, Short Sale Permitted**



No Transaction Cost and Short Sale Not Permitted

No Transaction Cost, Short Sale Permitted

With Transaction Cost, Short Sale Permitted

	Mean	Standard Deviation	Max	Min	Sharpe Ratio
Experiment 1	1.0541	0.0235	1.1020	0.9910	44.8553
Experiment 2	1.0493	0.0214	1.0936	0.9928	49.0327

- The following values decreased:
  - -Mean
  - -Standard deviation
  - -Maximum return

- Because the transactions were now in lots
  →Fewer stocks could be purchased
  →more money was placed in risk-free bond

- Return from stock market is usually greater than the return from risk-free bond
  → lowered the return of the portfolio