AM 207 Final Project

Forecasting "High" and "Low" of Financial Time Series by Particle Systems and Kalman Filters

Team Member: Chih-Kang Chang,
Jose Antonio Alatorre Sanchez,
Yuying Qian

I. Problem Statement

Forecasting future returns on assets is of obvious importance and interest in finance. If one was able to forecast tomorrow's returns on an asset with some degree of precision, one could use this information in an investment today and make profits without bearing the corresponding risk. However, the financial market is complex. It can follow different behaviors over time, such as overreaction, mean reversion, etc. A single process is hard to depict the market trend.

In this paper, we aim to predict asset price using high-frequency "tick-by-tick" financial time series and develop promising trading models under the prediction, without other economic information. Due to the nonlinear characteristics and complexity of financial time series, advanced scientific computing methods are applied to generate a better prediction.

II. Methods

A. Overview

First, we build a forecasting model of the asset price based on Functional Clustering and local Neural Networks methods. We use the EM algorithm to estimate the parameters for the functional clustering model. Once the price trend is known, we build a trading model regarding the first stopping time using an auto-adaptative Dynamic State Space Model. We apply Unscented Kalman Filters (and Particle Systems) for parameter estimation.

B. Technical content (details)

1. Price Forecasting Model

As mentioned before, financial markets should not be modeled by a single process, instead, a succession of different processes should be applied. So we build our forecasting model based on functional clustering analysis. In this method, we assume that a single model is not able to capture the dynamic of the whole time series. It splits the past of the series into clusters and generates a special local neural model for each of them. The local models are then combined in a probabilistic way, according to the distribution of the series in the past.

a. Functional Clustering Theory

Let $g_i(t)$ the hidden true value for the curve i at time t and gi , yi and εi, the random vectors of, respectively, hidden true values, measurements and errors. We have:

$$y_i \; = g_i + \epsilon_i , ,,,, \qquad\qquad ,i = 1, \cdots, N ,$$

where N is the number of curves. The random errors ε_i are assumed i.i.d., having a Gaussian distribution with zero mean and covariance σ^2 , uncorrelated with each other and with g_i .

Observing curve i on interval $[t_1, \ldots, t_{ni}]$ we define the vectors:

$$\mathbf{g}_i = \Big(g_i(t_1), \cdots, g_i(t_l), \cdots, g_i(t_{n_i})\Big)^T,$$
$$\mathbf{y}_i = \Big(y_i(t_1), \cdots, y_i(t_l), \cdots, y_i(t_{n_i})\Big)^T,$$

The unknown true functions g_i , are projected onto a functional basis by smoothing such as:

$$\widehat{g}_i(t) \approx \mathbf{s}^T(t)\ \boldsymbol{\eta}_i$$

where s(t) is the spline basis vector of dimension q , and η_i is a Gaussian random vector of spline coefficients:

$$\mathbf{s}(t) = \Big(s_1(t), \cdots, s_q(t)\Big)^T,$$
$$\boldsymbol{\eta}_i = \Big(\eta_{i1}, \cdots, \eta_{iq}\Big)^T.$$

The Gaussian coefficient ηi are split into two terms:

$$\boldsymbol{\eta}_i = \boldsymbol{\mu}_{ki} + \boldsymbol{\gamma}_i,$$

with $\gamma \sim N(\mathbf{0}, \boldsymbol{\Gamma})$.

We can represent the deviation between the centroid of cluster k and the global mean of the population λ_0 by:

$$\mu_k = \lambda_0 + \Lambda\alpha_k$$

where λ_0 is a q-dimensional vector, α_k a h-dimensional one, and Λ, a (q, h) matrix, with $h \le min(q, G - 1)$:

$$\boldsymbol{\lambda}_0 = \left(\lambda_{01}, \cdots, \lambda_{0q}\right)^T,$$
$$\boldsymbol{\alpha}_k = \left(\alpha_{k1}, \cdots, \alpha_{kh}\right)^T,$$
$$\boldsymbol{\Lambda} = \begin{pmatrix} \lambda_{11} & \cdots & \lambda_{1h} \\ \vdots & \ddots & \vdots \\ \lambda_{p1} & \cdots & \lambda_{qh} \end{pmatrix}.$$

With this formulation, the functional clustering model can be written as:

$$\mathbf{y}_i = \mathbf{S}_i(\boldsymbol{\lambda}_0 + \boldsymbol{\Lambda}\boldsymbol{\alpha}_{ki} + \boldsymbol{\gamma}_i) + \boldsymbol{\epsilon}_i \ , \qquad i = 1, \cdots, N,$$

with: $\varepsilon_i \sim N(0, R)$, and $\gamma_i \sim N(0, \Gamma)$.

S_i is the spline basis matrix for curve i :

$$\mathbf{S}_i = \begin{pmatrix} s_1(t_1) & \cdots & s_q(t_1) \\ \vdots & \ddots & \vdots \\ s_1(t_{n_i}) & \cdots & s_q(t_{n_i}) \end{pmatrix}.$$

$\alpha_{ki} = \alpha_k$ if curve i belongs to cluster k, where α_k is a representation of the centroid of cluster k in a reduced h-dimensional subspace, and

$$\boldsymbol{\alpha}_k = \left(\alpha_{k1}, \alpha_{k2} \ldots \alpha_{kh}\right)^T.$$

- $s(t)T\ \lambda_0$: the representation of the global mean curve,
- $s(t)T\ (\lambda_0 + \Lambda\alpha_k)$: the global representation of the centroid of cluster k,
- $s(t)T\ \Lambda\alpha_k$: the local representation of the centroid of cluster k in connection with the global mean curve,

- $s(t)T\ \gamma_i$: the local representation of the curve i in connection with the centroid of its cluster k.

b. Parameter Estimation by EM algorithm

Maximum Likelihood

We have to estimate the parameters λ_0 , Λ , α_k, Γ , $\sigma^2 et\ \pi_k$ by maximization of a likelihood function.

For y_i , we have a conditional distribution:

$$\mathbf{y}_i \sim N\Big(\mathbf{S}_i(\boldsymbol{\lambda}_0 + \boldsymbol{\Lambda}\boldsymbol{\alpha}_{ki}), \boldsymbol{\Sigma}_i\Big)$$

where:

$$\boldsymbol{\Sigma}_i = \sigma^2\mathbf{I} + \mathbf{S}_i\boldsymbol{\Gamma}\mathbf{S}_i^T$$

We define z_i as the unknown cluster membership vector of curve i, which will be treated as missing data,

$$\mathbf{z}_i = \begin{pmatrix} z_{1i} & z_{2i} & \cdots & z_{ki} & \cdots & z_{Gi} \end{pmatrix}$$

with:

$$z_{ki} = \begin{cases} 1 & \text{if curve } i \text{ belongs to cluster } k, \\ 0 & \text{otherwise.} \end{cases}$$

The probability that curve i belongs to cluster k is then:

$$\pi_{ki} = P\big(z_{ki} = 1\big)$$

When the observations of the different curves are independent, the joint distribution of y and z is given by:

$$f(\mathbf{y},\mathbf{z}) = \sum_{k=1}^{G} \pi_k \frac{1}{(2\pi)^{\frac{n}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}}} \exp\Big[-\frac{1}{2}\big(\mathbf{y} - \mathbf{S}(\boldsymbol{\lambda}_0 + \boldsymbol{\Lambda}\boldsymbol{\alpha}_k)\big)^T \boldsymbol{\Sigma}^{-1}\big(\mathbf{y} - \mathbf{S}(\boldsymbol{\lambda}_0 + \boldsymbol{\Lambda}\boldsymbol{\alpha}_k)\big)\Big]$$

and the likelihood for the parameters πk, λ0, Λ, αk, Γ, σ2, given observations $y_{1:N}$, and $z_{1:N}$ is

$$
\begin{aligned}
L(\pi_k, \boldsymbol{\lambda}_0, \boldsymbol{\Lambda}, \boldsymbol{\alpha}_k, \boldsymbol{\Gamma}, \sigma^2 | \mathbf{y}_{1:N}, \mathbf{z}_{1:N}) &= \prod_{i=1}^{N} \sum_{k=1}^{G} \pi_k \frac{1}{(2\pi)^{\frac{n_i}{2}} |\boldsymbol{\Sigma}_i|^{\frac{1}{2}}} \\
&\times \exp\Big[-\frac{1}{2}\big(\mathbf{y}_i - \mathbf{S}_i(\boldsymbol{\lambda}_0 + \boldsymbol{\Lambda}\boldsymbol{\alpha}_{ki})\big)^T \boldsymbol{\Sigma}_i^{-1} \big(\mathbf{y}_i - \mathbf{S}_i(\boldsymbol{\lambda}_0 + \boldsymbol{\Lambda}\boldsymbol{\alpha}_{ki})\big)\Big]
\end{aligned}
$$

and

$$
f(\mathbf{y}, \mathbf{z}, \boldsymbol{\gamma}) = f(\mathbf{y}|\mathbf{z}, \boldsymbol{\gamma}) f(\mathbf{z}) f(\boldsymbol{\gamma}) \ ,
$$

The joint distribution:

$$
\begin{aligned}
f(\mathbf{y}, \mathbf{z}, \boldsymbol{\gamma}) = \frac{1}{(2\pi)^{\frac{n+q}{2}} |\boldsymbol{\Gamma}|^{\frac{1}{2}}} \exp\big[-\frac{1}{2}\boldsymbol{\gamma}^T \boldsymbol{\Gamma}^{-1} \boldsymbol{\gamma}\big] \prod_{k=1}^{G} \Big\{ \pi_k \exp\big[-\frac{1}{2} n \log(\sigma^2)\big] \\
\exp\big[-\frac{1}{2\sigma^2}\big(\mathbf{y} - \mathbf{S}(\boldsymbol{\lambda}_0 + \boldsymbol{\Lambda}\boldsymbol{\alpha}_k + \boldsymbol{\gamma})\big)^T \big(\mathbf{y} - \mathbf{S}(\boldsymbol{\lambda}_0 + \boldsymbol{\Lambda}\boldsymbol{\alpha}_k + \boldsymbol{\gamma})\big)\big] \Big\}^{z_k},
\end{aligned}
$$

the likelihood of the parameters is:

$$
\begin{aligned}
& L(\pi_k, \boldsymbol{\lambda}_0, \boldsymbol{\Lambda}, \boldsymbol{\alpha}_k, \boldsymbol{\Gamma}, \sigma^2 | \mathbf{y}_{1:N}, \mathbf{z}_{1:N}, \boldsymbol{\gamma}_{1:N}) = \\
& \quad \prod_{i=1}^{N} \frac{1}{(2\pi)^{\frac{n_i+q}{2}} |\boldsymbol{\Gamma}|^{\frac{1}{2}}} \exp\big[-\frac{1}{2}\boldsymbol{\gamma}_i^T \boldsymbol{\Gamma}^{-1} \boldsymbol{\gamma}_i\big] \prod_{k=1}^{G} \Big\{ \pi_k \exp\big[-\frac{1}{2} n_i \log(\sigma^2)\big] \\
& \quad \exp\big[-\frac{1}{2\sigma^2}\big(\mathbf{y}_i - \mathbf{S}_i(\boldsymbol{\lambda}_0 + \boldsymbol{\Lambda}\boldsymbol{\alpha}_k + \boldsymbol{\gamma}_i)\big)^T \big(\mathbf{y}_i - \mathbf{S}_i(\boldsymbol{\lambda}_0 + \boldsymbol{\Lambda}\boldsymbol{\alpha}_k + \boldsymbol{\gamma}_i)\big)\big] \Big\}^{z_{ki}}.
\end{aligned}
$$

EM algorithm

Direct maximization of this likelihood is a difficult non-convex optimization problem, we use the EM algorithm to overcome this difficulty.

For parameter estimation by EM algorithm, we will use the log-likelihood:

$$
\begin{aligned}
l(\pi_k, \boldsymbol{\lambda}_0, \boldsymbol{\Lambda}, \boldsymbol{\alpha}_k, \boldsymbol{\Gamma}, \sigma^2 | \mathbf{y}_{1:N}, \mathbf{z}_{1:N}, \boldsymbol{\gamma}_{1:N}) \quad &= \\
- \quad & \frac{1}{2}\sum_{i=1}^{N}(n_i + q)\log(2\pi) \\
+ \quad & \sum_{i=1}^{N}\sum_{k=1}^{G} z_{ki}\log(\pi_k) \\
- \quad & \frac{1}{2}\sum_{i=1}^{N}\left[\log(|\boldsymbol{\Gamma}|) + \boldsymbol{\gamma}_i^T\boldsymbol{\Gamma}^{-1}\boldsymbol{\gamma}_i\right] \\
- \quad & \frac{1}{2}\sum_{i=1}^{N}\sum_{k=1}^{G} z_{ki}\left[n_i\log(\sigma^2) + \frac{1}{\sigma^2}\left\|\mathbf{y}_i - \mathbf{S}_i(\boldsymbol{\lambda}_0 + \boldsymbol{\Lambda}\boldsymbol{\alpha}_k + \boldsymbol{\gamma}_i)\right\|^2\right]
\end{aligned}
$$

The EM algorithm consists of iteratively maximizing the expected values of the above equations, given y_i and the current parameter estimates. As these three parts involve separate parameters, we can optimize them separately.

Initialization

First, we must initialize all parameters:

$$\{\lambda 0,\ \Lambda,\ \alpha,\ \Gamma,\ \sigma^2,\ zik,\ \pi k,\ \gamma i\}$$

E-step

The E-step consists of:

$$\widehat{\gamma}_i = E\Big\{\boldsymbol{\gamma}_i | \boldsymbol{y}_i, \boldsymbol{\lambda}_0, \boldsymbol{\Lambda}, \boldsymbol{\alpha}, \boldsymbol{\Gamma}, \sigma^2, z_{ik}\Big\}$$

With:

$$\widehat{\boldsymbol{\gamma}}_i = \left(\mathbf{S}_i^T\mathbf{S}_i + \sigma^2\boldsymbol{\Gamma}^{-1}\right)^{-1}\mathbf{S}_i^T\left(\mathbf{y}_i - \mathbf{S}_i\boldsymbol{\lambda}_0 - \mathbf{S}_i\boldsymbol{\Lambda}\boldsymbol{\alpha}_k\right)$$

M-step

The M-step involves maximizing:

$$Q = E\Big\{l(\pi_k, \boldsymbol{\lambda}_0, \boldsymbol{\Lambda}, \boldsymbol{\alpha}_k, \boldsymbol{\Gamma}, \sigma^2 | \boldsymbol{y}_{1:N}, z_{1:N}, \boldsymbol{\gamma}_{1:N})\Big\}$$

holding $\gamma_{1:N}$ fixed, and given by the E-step.

2. Trading Model

Once we get the price forecasting model from the above methods, we continue to look for a trading model based on the forecasting of the price movement. We aim to find out a stopping time that represents the 'High' and 'Low' price of prediction.

a. State Space Model

It is well-known in financial markets that stock price follows a diffusion geometrical Wiener process. Volatility is an important variable in the stock price function and volatility itself should follow a stochastic process. We have:

$$dS_t = S_t \left(\mu_t dt + \sqrt{V_t} dB_t\right)$$

$$dV_t = \alpha(S,t)dt + \beta(S,t)dZ_t$$

where $\alpha(S, t)$ and $\beta(S, t)$ are functions of V_t, B_t , and Z_t are correlated Brownian motions.

After taking the logarithms of stock price and volatility and using the Itô's formula, we derive the process in a continuous dynamic state-space formulation. Since stock price observation is discrete, we approximate the continuous-time stochastic volatility models by a system of stochastic difference equations, that is the Dynamic State Space Model (DSSM):

$$\begin{aligned}\log V_{t+1} &= \log V_t + \frac{1}{V_t}\big[\kappa(\theta - V_t) - \frac{1}{2}\xi^2 V_t^{2p-1} - \rho\xi V_t^{p-\frac{1}{2}}(\mu - \frac{1}{2}V_t)\big]\Delta t + \\ &\quad \rho\xi V_t^{p-\frac{3}{2}}\big(\ln S_t - \ln S_{t-1}\big) + \xi V_t^{p-1}\sqrt{\Delta t}\sqrt{1-\rho}Z_t \\ \ln S_t &= \ln S_{t-1} + \big(\mu - \frac{1}{2}V_t\big)\Delta t + \sqrt{\Delta t}\sqrt{V_t}B_t,\end{aligned}$$

Since we would like to forecast the future curve for a couple of hours and not only the next value of the time series, we derive a non-parametric Dynamic State Space Model where random variables may be scalars, vectors or curves:

$$\begin{aligned}\mathbf{x}_k &= \mathbf{f}_k(\mathbf{x}_{k-1}, \mathbf{u}_k, \mathbf{v}_k) \\ \mathbf{y}_k &= \mathbf{h}_k(\mathbf{x}_k, \mathbf{n}_k)\end{aligned}$$

Where:

- x_k is the state, $x_k \in R^n$,
- y_k is the observation, $y_k \in R^q$
- u_k is an exogenous input, it’s a deterministic sequence, $u_k \in R^m$
- v_k is the process noise, n_k is the measurement noise. $v_k \sim N(0, Q)$, $n_k \sim N(0,\ R)$
- f_k is the state equation, h_k is the measurement equation, these parametric functions may be time-variant but are known

b. Unscented Kalman Filter (UKF)

Due to the nonlinearity characteristic of stock price time series, we apply the Unscented Kalman filter method here. The Unscented Kalman filter approximates the distribution of the state variable by using an unscented transformation. It is an approach for Kalman filtering in the case of nonlinear equations.

Sigma-points

$$\mathcal{X}_{k-1}^a = \big(\, \hat{\mathbf{x}}_{k-1}^a \;\; \hat{\mathbf{x}}_{k-1}^a + \gamma\sqrt{\mathbf{P}_{k-1}^a} \;\; \hat{\mathbf{x}}_{k-1}^a - \gamma\sqrt{\mathbf{P}_{k-1}^a}\,\big)$$

Initialization

The algorithm is initialized with the initial weights for the sigma-points, and with an initial state and state covariance.

$$
\begin{aligned}
w_0^{(m)} &= \frac{\lambda}{L+\lambda} \\
w_0^{(c)} &= \frac{\lambda}{L+\lambda} + (1-\alpha^2+\beta) \\
w_i^{(m)} = w_i^{(c)} &= \frac{1}{2(L+\lambda)} \quad \mathit{for}\ i = 1, \cdots, 2L,
\end{aligned}
$$

$$
\begin{aligned}
\widehat{\mathbf{x}}_0 &= \mathbb{E}[\mathbf{x}_0] \\
\mathbf{P}_{\mathbf{x}_0} &= \mathbb{E}[(\mathbf{x}_0 - \widehat{\mathbf{x}}_0)(\mathbf{x}_0 - \widehat{\mathbf{x}}_0)^T] \\
\widehat{\mathbf{x}}_0^a &= \mathbb{E}[\mathbf{x}^a] = \mathbb{E}[\big(\ (\mathbf{x}_0)^T\ (\mathbf{0})^T\ (\mathbf{0})^T\ \big)^T] \\
\mathbf{P}_0^a &= \mathbb{E}[(\mathbf{x}_0^a - \widehat{\mathbf{x}}_0^a)(\mathbf{x}_0^a - \widehat{\mathbf{x}}_0^a)^T] = \begin{pmatrix} \mathbf{P}_{\mathbf{x}_0} & 0 & 0 \\ 0 & \mathbf{Q} & 0 \\ 0 & 0 & \mathbf{R} \end{pmatrix}
\end{aligned}
$$

Prediction step

The equations for the prediction of the state value and covariance are:

$$
\begin{aligned}
\mathcal{X}_{k|k-1}^x &= \mathbf{f}(\mathcal{X}_{k-1}^x, \mathcal{X}_{k-1}^v, \mathbf{u}_{k-1}) \\
\widehat{\mathbf{x}}_{k|k-1} &= \sum_{i=0}^{2L} w_i^{(m)} \mathcal{X}_{i,k|k-1}^x \\
\mathbf{P}_{\mathbf{x}_{k|k-1}} &= \sum_{i=0}^{2L} w_i^{(c)} (\mathcal{X}_{i,k|k-1}^x - \widehat{\mathbf{x}}_{k|k-1})(\mathcal{X}_{i,k|k-1}^x - \widehat{\mathbf{x}}_{k|k-1})^T
\end{aligned}
$$

Innovation

By using the state prediction, the innovation and the prediction error e_k are:

$$
\begin{aligned}
\mathcal{Y}_{k|k-1} &= \mathbf{h}(\mathcal{X}^{x}_{k|k-1}, \mathcal{X}^{n}_{k-1}) \\
\widehat{\mathbf{y}}_{k|k-1} &= \sum_{i=0}^{2L} w_i^{(m)} \mathcal{Y}_{i,k|k-1} \\
\mathbf{e}_k &= \mathbf{y}_k - \widehat{\mathbf{y}}_{k|k-1}
\end{aligned}
$$

Measurement Update step

Finally, by computing the predicted covariance, we get the Kalman gain:

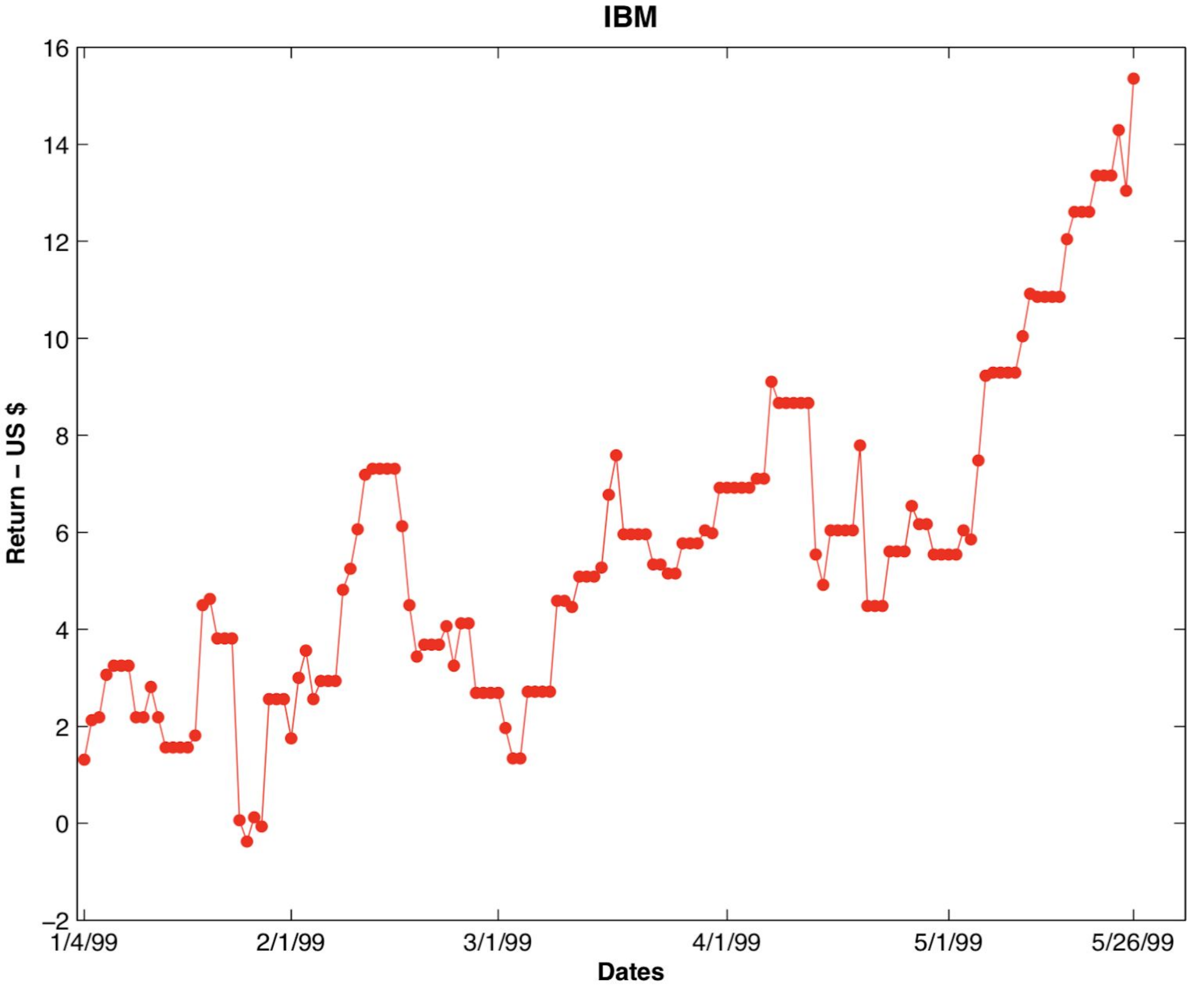
$$
\begin{aligned}
\mathbf{P}_{\widetilde{\mathbf{y}}_k} &= \sum_{i=0}^{2L} w_i^{(c)} (\mathcal{Y}_{i,k|k-1} - \widehat{\mathbf{y}}_{k|k-1})(\mathcal{Y}_{i,k|k-1} - \widehat{\mathbf{y}}_{k|k-1})^T \\
\mathbf{P}_{\mathbf{x}_k\mathbf{y}_k} &= \sum_{i=0}^{2L} w_i^{(c)} (\mathcal{X}^{x}_{i,k|k-1} - \widehat{\mathbf{x}}_{k|k-1})(\mathcal{Y}_{i,k|k-1} - \widehat{\mathbf{y}}_{k|k-1})^T \\
\mathbf{K}_k &= \mathbf{P}_{\mathbf{x}_k\mathbf{y}_k} \mathbf{P}^{-1}_{\widetilde{\mathbf{y}}_k}
\end{aligned}
$$

As before, we can now update the system state and covariance:

$$
\begin{aligned}
\widehat{\mathbf{x}}_{k|k} &= \widehat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \mathbf{e}_k \\
\mathbf{P}_{\mathbf{x}_{k|k}} &= \mathbf{P}_{\mathbf{x}_{k|k-1}} - \mathbf{K}_k \mathbf{P}_{\widetilde{\mathbf{y}}_k} \mathbf{K}_k^T
\end{aligned}
$$

III. Experiments

The experiment was performed on the IBM stock price the period Jan 3, 1995, to May 26, 1999. However, not much detail about the analysis is provided. The result shown in the paper is a return plot of the last 5 months, which is the testing data that was not used in training:



The result shows that the return is positive throughout 5 months of testing data. The result proves that the proposed method has some benefit in predicting the financial data, however, it does not compare the result with a dumb model, e.g. deciding the trading action based on history frequency, and it is vague on how well the model works without further analysis.

IV. Evaluation

The proposed model is innovative in the way that it smoothes the rough high-frequency data to a higher dimension with less correlation. This approach enhances the applicability of the data to implement different kinds of estimation methods. In addition, the dynamic state space model implemented in this paper is a strong method to estimate hidden variables. However, author's approach to approximate the unknown transition function and observation function may be improved. The reason the author chooses to approximate the functions using a sigle-layer RFBN is not clearly specified in the paper, and yet there may be other better approach to approximate the unknown function. Overall, the proposed method is definitely practical to use on real data and tasks, while there should be stronger experiments to show the evaluation of the implemented model.