Vrije Universiteit Amsterdam
Period 4, 2024-2025

Computer Assignment

Time Series Models

Instructors: Karim Moussa, Ilka van de Werve

Yunji Eo (2735445)
Yuval Nitzan (2775283)
Guo Yin (2798401)
Alexandra Delistrati (2838783)

Question a

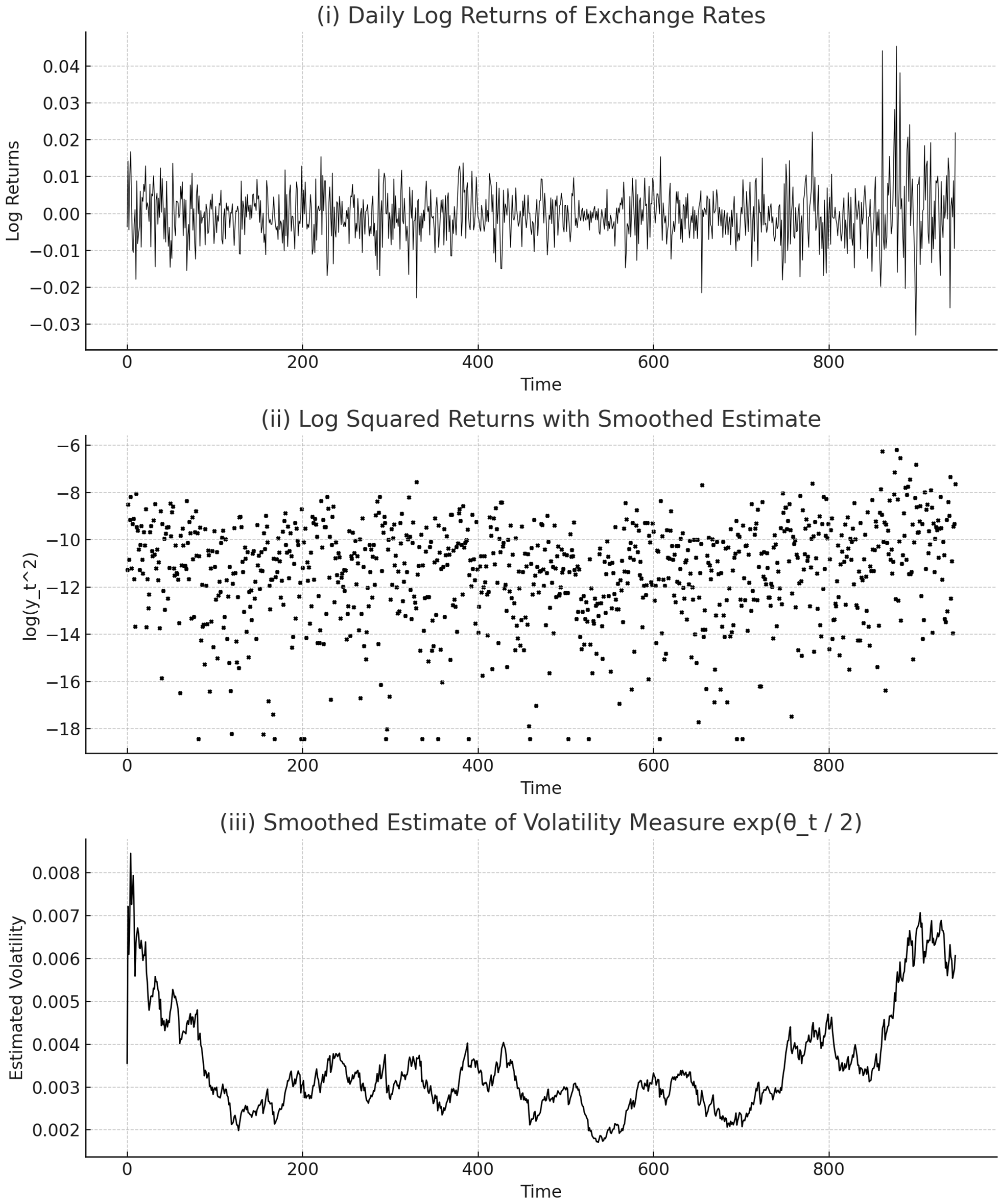

Figure 1: Time Series Analysis of Exchange Rate Log Returns: Daily Returns, Log Squared Returns, and Smoothed Volatility Estimates

The dataset contains the data of the financial series of daily log returns of exchange rates. The data is in log returns for better assessing the volatility behavior over the period. The daily log returns, which is denoted as y_t, will be analyzed by the stochastic volatility model, followed as:

$$y_t = \mu + \sigma \exp\left(\frac{\alpha_t}{2}\right)\epsilon_t, \quad \epsilon_t \sim N(0,1) \tag{1}$$

$$\alpha_{t+1} = \phi\alpha_t + \eta_t, \quad \eta_t \sim N(0, \sigma^2_\eta) \tag{2}$$

Where $\sigma \exp\left(\frac{\alpha_t}{2}\right)$ is the volatility and the observation error as ϵ_t.

The plots seen below are the daily log returns of exchange rates, log squared returns, and the smoothed estimate of volatility measure. From the plots, we can see that for the daily log returns, the time series shows a high level of fluctuations, especially for the later period. For the log squared returns, it shows the volatility clustering and will be used in estimations. For the smoothed volatility estimate, it captures the long-term trends in market fluctuations.

Table 1: Descriptive Statistics of GBP/USD Log Returns

Statistic	**Value**
Number of Observations	945
Mean	-0.000353
Standard Deviation	0.007111
Minimum	-0.032961
25th Percentile (Q1)	-0.004394
Median (Q2)	-0.000457
75th Percentile (Q3)	0.003644
Maximum	0.045345

The descriptive data are shown in the table above. There are 945 observations contained in the dataset, with the mean -0.000353, which does not contain much information due to the volatility of the data. The standard deviation is around 0.007, showing the dispersion of the daily log returns. The minimum and maximum represent the largest observed negative and positive returns in the data. The median is around -0.00046, smaller than the mean, which shows that the data is slightly skewed.

Question b

As the volatility and the observation error are stochastic processes, the SV model is a nonlinear time series model. To do the analysis on the linear model, we need to transform the log returns into a linearized form, which is based on:

$$x_t = \log((y_t - \mu)^2) \tag{3}$$

Which deducted the mean from the log returns, squared, and took the natural logarithm. This transformation will later allow the quasi-maximum likelihood approach and the estimation using the Kalman filter. The plot is shown below:

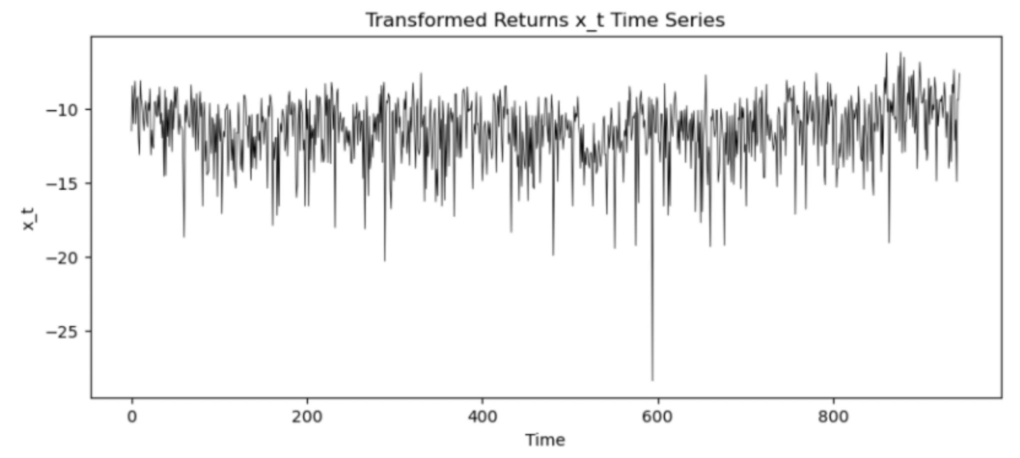

Figure 2: Time Series of Transformed Log Returns

Question c

Estimation of Parameters Using the QML Approach

To estimate the parameters κ, ϕ, and σ_η^2 in the stochastic volatility model, we employ the quasi-maximum likelihood (QML) approach based on the approximate linearized model:

$$x_t = \kappa + \alpha_t + \xi_t, \quad \xi_t \sim N(0, \pi^2/2), \tag{4}$$

$$\alpha_{t+1} = \phi\alpha_t + \eta_t, \quad \eta_t \sim N(0, \sigma_\eta^2), \tag{5}$$

where $\psi = (\kappa, \phi, \sigma_\eta^2)$ represents the parameter vector to be estimated. The disturbances $\{\xi_t\}$ and $\{\eta_t\}$ are assumed to be mutually and serially independent.

To apply the Kalman filter for parameter estimation, we first initialize the state process using its stationary unconditional distribution:

$$\alpha_1 \sim N\left(0, \frac{\sigma_\eta^2}{1-\phi^2}\right). \tag{6}$$

Following this initialization, the Kalman filter is implemented iteratively to update the state and observation equations, computing filtered estimates $\mathbb{E}[\alpha_t|x_1, \ldots, x_t]$. These filtered estimates are then used to compute the log-likelihood, which is maximized with respect to the parameters κ, ϕ, and σ_η^2.

An essential step is the estimation of ϕ, which can be obtained using the sample autocovariances:

$$\phi = \frac{\mathrm{Cov}[x_{t+1}, x_t]}{\mathrm{Cov}[x_t, x_t]}. \tag{7}$$

Finally, the parameter σ_η^2 is estimated using:

$$\sigma_\eta^2 = (1-\phi^2)\left(\mathrm{Var}[x_t] - \frac{\pi^2}{2}\right). \tag{8}$$

These estimates provide the necessary input for applying the Kalman filter and smoother in the subsequent step.

Parameter Estimates and Log-Likelihood

In our estimation procedure, the log-likelihood function is constructed using the Kalman filter, which recursively computes the prediction errors and their variances based on the linearized state-space model. At each time step t, the filter generates a one-step-ahead prediction error v_t and its variance F_t, which contribute to the overall likelihood of the observed data. The log-likelihood function is then defined as the sum of these terms across all observations. Since the log-likelihood function is nonlinear in the parameters $\psi = (\kappa, \phi, \sigma_\eta^2)$, we maximize it numerically using the *L-BFGS-B optimization algorithm*, which iteratively updates the parameter estimates to minimize the negative log-likelihood. The optimization ensures that constraints such as $0 < \phi < 1$ (to maintain stationarity) and $\sigma_\eta^2 > 0$ (to ensure a positive variance) are satisfied. The final parameter estimates $\hat{\kappa}, \hat{\phi}, \hat{\sigma}_\eta^2$ are obtained at the optimal point, where the log-likelihood function reaches its maximum, providing the best fit to the observed data. To summarize the estimated parameter values obtained from the QML approach, we present the following table:

Table 2: Parameter Estimates and Log-Likelihood

Parameter	**Estimate**
κ	-11.2769
ϕ	0.9911
σ_η^2	0.0070
Log-likelihood	-2083.6122

Question d

In this section, we estimate the latent state α_t of the stochastic volatility (SV) model using the Kalman filter and Kalman smoother based on the estimated parameters $\hat{\kappa}$, $\hat{\phi}$, and $\hat{\sigma}_\eta^2$. The Kalman filter iteratively updates real-time estimates of α_t using only past observations, while the Kalman smoother refines these estimates by incorporating future information, leading to a more accurate representation of the latent volatility process. The filtered estimates $\mathbb{E}[\alpha_t | x_1, \ldots, x_t]$ are obtained using the state-space representation of the transformed data x_t, following the recursive prediction and update steps:

$$\hat{\alpha}_{t|t-1} = \phi \hat{\alpha}_{t-1} \tag{9}$$

$$P_{t|t-1} = \phi^2 P_{t-1} + \sigma_\eta^2 \tag{10}$$

$$v_t = x_t - \kappa - \hat{\alpha}_{t|t-1} \tag{11}$$

$$F_t = P_{t|t-1} + \frac{\pi^2}{2} \tag{12}$$

$$K_t = \frac{P_{t|t-1}}{F_t} \tag{13}$$

$$\hat{\alpha}_{t|t} = \hat{\alpha}_{t|t-1} + K_t v_t \tag{14}$$

$$P_{t|t} = (1 - K_t)P_{t|t-1} \tag{15}$$

After filtering, the Kalman smoother further refines these estimates using backward recursion to incorporate future observations:

$$r_t = \phi r_{t+1} + \frac{v_t}{F_t} \tag{16}$$

$$\mathbb{E}[\alpha_t|x_1, \ldots, x_n] \tag{17}$$

where the smoothed estimate $\mathbb{E}[\alpha_t|x_1, \ldots, x_n]$ represents an improved approximation of the volatility component.

The result of the filtering and smoothing is plotted in 3. As expected, the smoothing produces a smoother plot compared to the filtering because it incorporates both past and future observations, reducing short-term fluctuations. Filtering, on the other hand, only uses past data, making it more sensitive to immediate changes in the series.

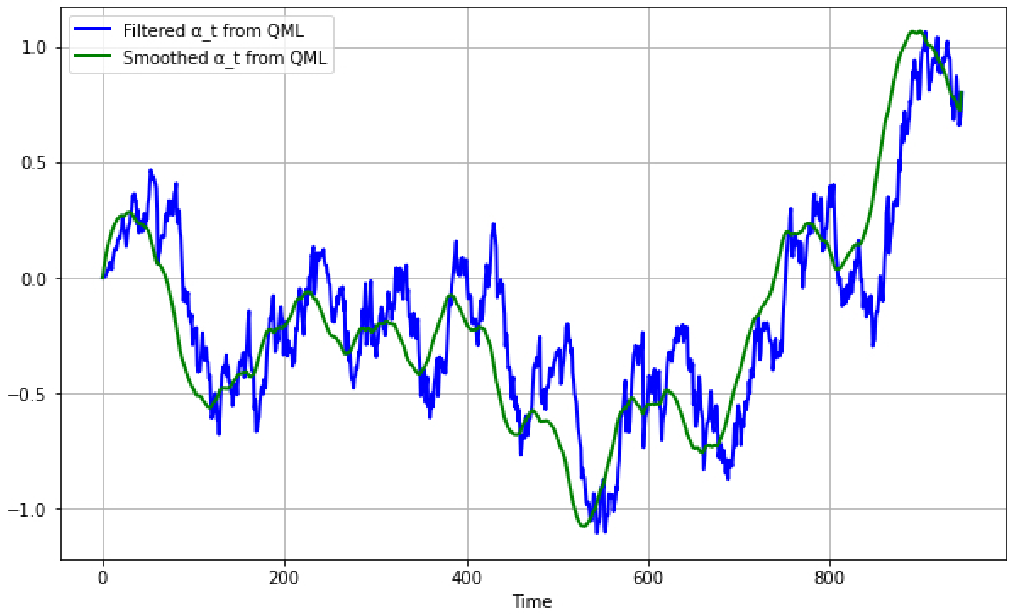

Figure 3: Filtered and smoothed estimates of α_t. The blue line represents the filtered estimates $\mathbb{E}[\alpha_t|x_1, \ldots, x_t]$, while the green line shows the smoothed estimates $\mathbb{E}[\alpha_t|x_1, \ldots, x_n]$.

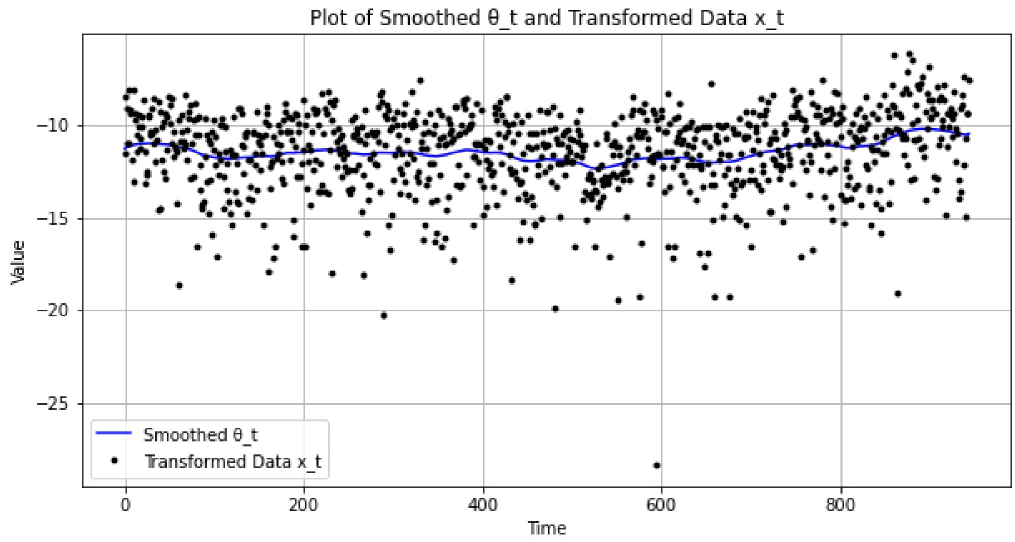

Figure 4: Comparison of Transformed Data x_t and Smoothed Volatility Estimates θ_t.

Analysis of Transformed Data x_t and Smoothed Estimates θ_t

Then, we obtained the smoothed volatility estimate θ, which is calculated as :

$$E[\theta_t|x_1,\dots,x_n], \quad \text{with } \theta_t = \kappa + \alpha_t. \tag{18}$$

In 4, the black dots represent the observed transformed data x_t, while the blue line corresponds to the smoothed estimate $\mathbb{E}[\alpha_t|x_1,\dots,x_n]$, shifted by the intercept term κ. This adjustment ensures comparability between the state estimates and the transformed observations, as required by the model formulation. Specifically, the plot highlights the ability of the Kalman smoother to capture the underlying volatility structure while filtering out short-term noise present in x_t. By comparing the observed transformed data to the smoothed estimates, we can assess the effectiveness of the QML approach in recovering the latent volatility process from the observed returns. The results demonstrate that the smoothed estimates track the general trend of x_t while reducing the high-frequency fluctuations, thereby providing a statistically efficient estimate of market volatility dynamics.

Question e

To estimate the mode, we reformulate the problem as a Linear Gaussian model, where z represents the observations and A is the variance matrix. The model is given by :

$$z_t = \alpha_t + \varepsilon_t, \quad \varepsilon_t \sim \mathcal{N}(0, A_t)$$

$$\alpha_{t+1} = \phi\alpha_t + \eta_t, \quad \eta_t \sim \mathcal{N}(0, \sigma_\eta^2)$$

where

$$A_t = 2\exp(g_t)/\left(\bar{y}_t\right)^2, z_t = g_t + 1 - \exp(g_t)/\left(\bar{y}_t\right)^2,$$

The estimated smoothed mean $\hat{\alpha}$ serves as the mode estimate, which is also the updated guess :

$$\hat{\alpha} = \hat{\theta} = g^+$$

If convergence is achieved, the mode is given by $\tilde{\theta} = \hat{\theta}$. Otherwise, we set $g = g^+$ and iterate until convergence. To check for convergence, we use this criteria

$$|g^+ - g| \leq 10^{-7} \text{ (pointwise)}$$

We begins this iteration loop by initializing A_t and z_t using :

$$A_t = 2, z_t = 2\log\left(|\bar{y}_t|\right);,$$

The iterative process for estimating the mode was carried out until convergence was achieved. The maximum change in g at each iteration is shown below:

$$\begin{aligned}
&\text{Iteration } 0: \quad \max \Delta g = 0.8631076833258597\\
&\text{Iteration } 1: \quad \max \Delta g = 0.5871472437485616\\
&\text{Iteration } 2: \quad \max \Delta g = 0.20834874073638288\\
&\text{Iteration } 3: \quad \max \Delta g = 0.021565141008124677\\
&\text{Iteration } 4: \quad \max \Delta g = 0.00042965383671067503\\
&\text{Iteration } 5: \quad \max \Delta g = 5.223828049516488 \times 10^{-6}\\
&\text{Iteration } 6: \quad \max \Delta g = 6.160795240628403 \times 10^{-8}
\end{aligned}$$

After 6 iterations, the convergence criterion was satisfied, indicating that the mode estimate has stabilized.

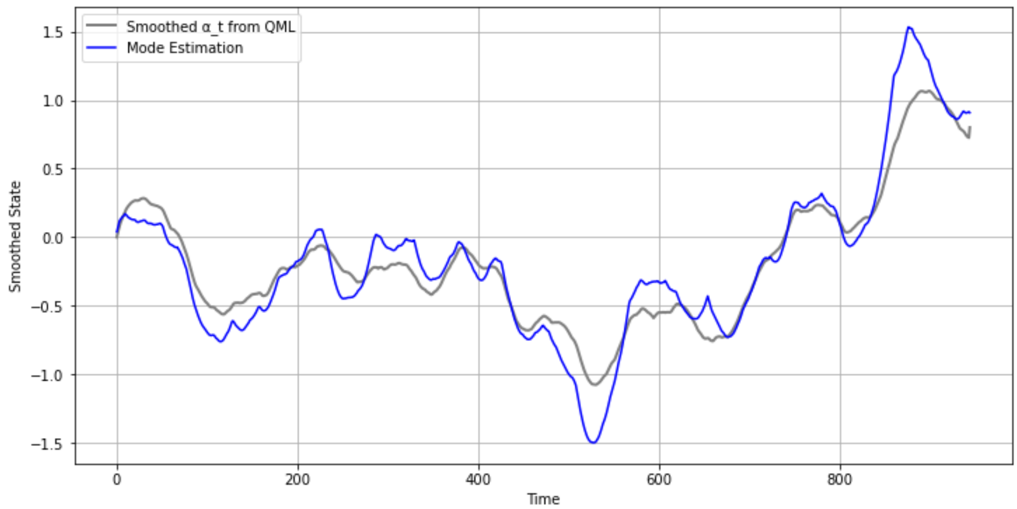

Figure 5: Mode Estimation and Smoothed QML estimates of α_t.

Figure 5 shows the plot of Mode Estimation after iteration alongside the smoothed QML estimates. The two plots exhibit similar overall trends, but the mode estimation fluctuates more than the QML estimates. This is expected because the iterative process for mode estimation works locally, optimizing for the mode at each step, making it more sensitive to short-term variations in the data. The QML smoothing approach, on the other hand, incorporates both past and future observations, leading to a smoother estimate of the latent volatility.

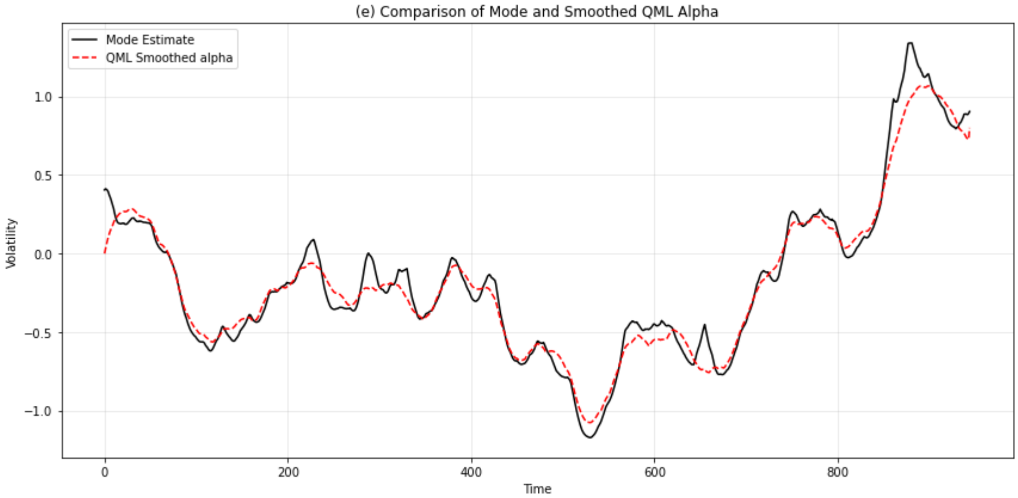

Figure 6: Comparison of Mode and Smoothed QML α_t

To obtain a smoother mode estimate, we also performed mode estimation based on density and its derivative, using the L-BFGS-B method for optimization. As shown in Figure 6, the mode exhibits smoother and less fluctuating plot compared to the iteration-based mode estimation.The key difference is that the derivative-based method finds a global mode by considering the overall shape of the density function rather than optimizing locally at each step. This makes the estimate more stable and less sensitive to short-term fluctuations in the data. While the previous mode estimation reacts more to short-term changes, this approach finds a better balance between capturing trends and avoiding unnecessary noise. It also looks more similar to the QML-based smoothing method.

Question f

Using the bootstrap particle filter, we estimate the filtered state α_t and compare it to the QML-based filtered estimates. Unlike QML, which relies on a linearized approximation, the bootstrap particle filter provides a fully non-linear filtering approach, enabling a more flexible estimation of the latent state. This method allows for the incorporation of non-Gaussian features and captures volatility dynamics more adaptively.

The implementation follows a sequential Monte Carlo framework, where a set of weighted particles approximates the posterior distribution of α_t. Initially, N particles, $\{\alpha_0^{(i)}\}_{i=1}^N$, are drawn from the stationary distribution:

$$\alpha_0 \sim \mathcal{N}\left(0, \frac{\sigma_\eta^2}{1-\phi^2}\right).$$

At each time step t, the weights are updated based on the observation density:

$$p(y_t|\alpha_t^{(i)}) \propto \exp\left\{-\frac{1}{2}\left[\log(2\pi\sigma^2 e^{\alpha_t^{(i)}}) + \frac{(y_t-\mu)^2}{\sigma^2 e^{\alpha_t^{(i)}}}\right]\right\}.$$

To mitigate particle degeneracy, systematic resampling is applied, replacing particles with lower weights with those that better represent the observed data. The resampled particles are then propagated forward using the state equation:

$$\alpha_{t+1}^{(i)} = \phi\alpha_t^{(i)} + \eta_t^{(i)}, \quad \eta_t^{(i)} \sim \mathcal{N}(0, \sigma_\eta^2).$$

The filtered estimate $\hat{\alpha}_t$ is then computed as the weighted mean of the resampled particles:

$$\hat{\alpha}_t = \sum_{i=1}^{N} \alpha_t^{(i)} w_t^{(i)}.$$

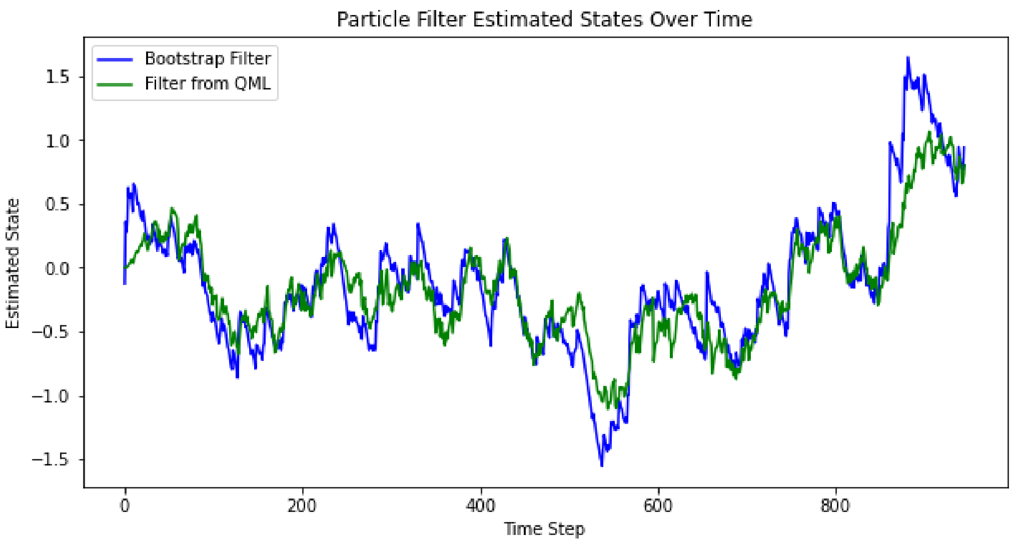

Figure 7: Comparison of Bootstrap Filtered and QML Filtered Volatility Estimates.

Figure 7 illustrates the differences between the bootstrap particle filter and the QML approach in estimating α_t. Both methods capture the underlying volatility structure, but they differ in how they respond to changes in market conditions. The bootstrap filter provides a fully non-linear approach, which can lead to more dynamic responses to sudden fluctuations. On the other hand, the QML filter relies on linearization, producing smoother estimates that may not fully capture extreme variations.

The results show that during periods of heightened volatility, the bootstrap filter exhibits greater sensitivity to rapid changes, whereas the QML approach tends to smooth these fluctuations. This difference arises from the methodological assumptions: the bootstrap filter maintains a sequential Monte Carlo framework that accounts for abrupt shifts, while the QML method follows a state-space approach based on linear approximations.

Both filtering techniques have distinct characteristics and computational considerations. The bootstrap particle filter offers flexibility in modeling non-Gaussian and highly volatile processes, making it particularly relevant in financial settings where sudden price swings are common. However, this flexibility comes with increased computational demands compared to QML filtering, which remains an efficient alternative for capturing overall volatility trends.

By comparing these two approaches, we can assess their suitability for different applications. The bootstrap filter's ability to incorporate full non-linearity makes it useful for scenarios requiring high adaptability, whereas QML filtering provides a more stable estimation framework with lower computational complexity. These findings highlight the importance of selecting a filtering method that aligns with the characteristics of the data and the objectives of the analysis.