

Factor and Idiosyncratic VAR Volatility Matrix Models for Heavy-Tailed High-Frequency Financial Observations

POSTECH IME
JAEWON SEO
seojw115@postech.ac.kr

Index

1. Introduction
2. FIVAR model
3. Estimation procedure for the heavy-tailed VAR model
4. Large volatility matrix prediction
5. Numerical study
6. Further research direction

Introduction

- Volatility analysis for high-frequency financial data is a vibrant research area in financial econometrics and statistics
- With the wide availability of high-frequency financial data, several well performing non-parametric estimation methods have been developed to estimate integrated volatilities
- Problem: Large number of assets leads to an excessive number of parameters for typical sample sizes

Introduction

- **Problem:** Large number of assets leads to an excessive number of parameters for typical sample sizes
- **Approach:** Approximate factor model structure
- **Limitation:** Previous research assumes idiosyncratic volatility process is martingale, only models factor components

Introduction

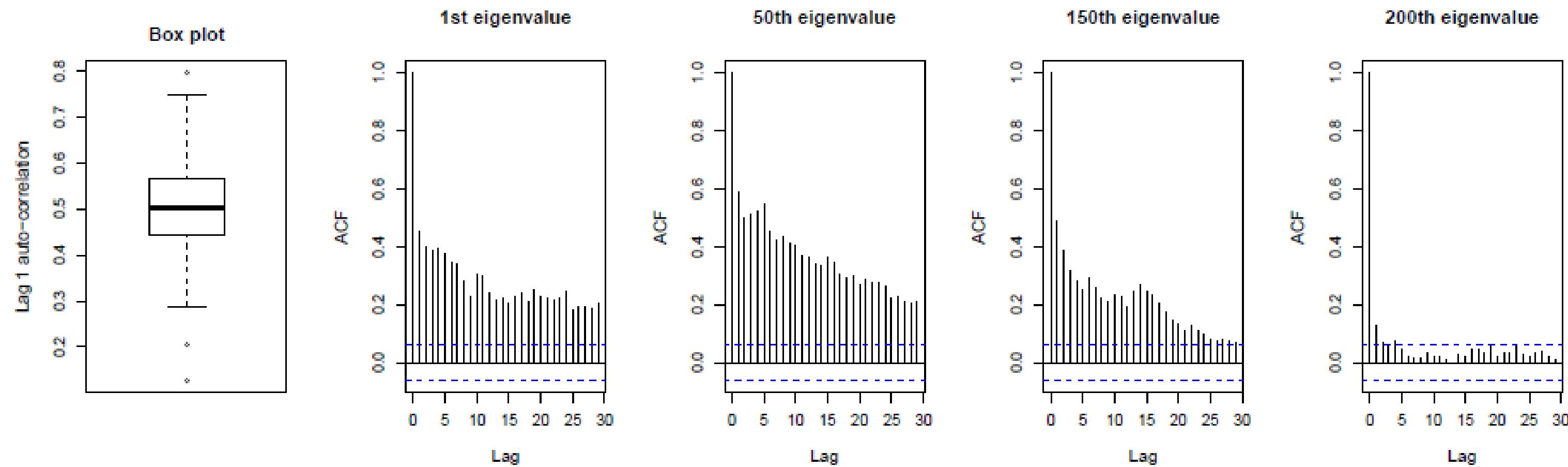


Figure 1: The box plot of the first-order auto-correlations for the time series of 200 daily estimated eigenvalues of the idiosyncratic volatility matrix and the ACF plots for the time series of the 1st, 50th, 150th, and 200th eigenvalues.

Introduction

- **Analysis:** Thus, modeling both factor and idiosyncratic volatility is important to capture volatility dynamics
- **Problem:** High dimensional trait of volatility causes over-parameterization
- **Approach:** Sparsity of model parameters is often assumed (LASSO, SCAD)
- **Limitation:**
 - Those approaches assume under sub-Gaussian condition
 - High frequency financial data tends to show heavy tail trait

Introduction

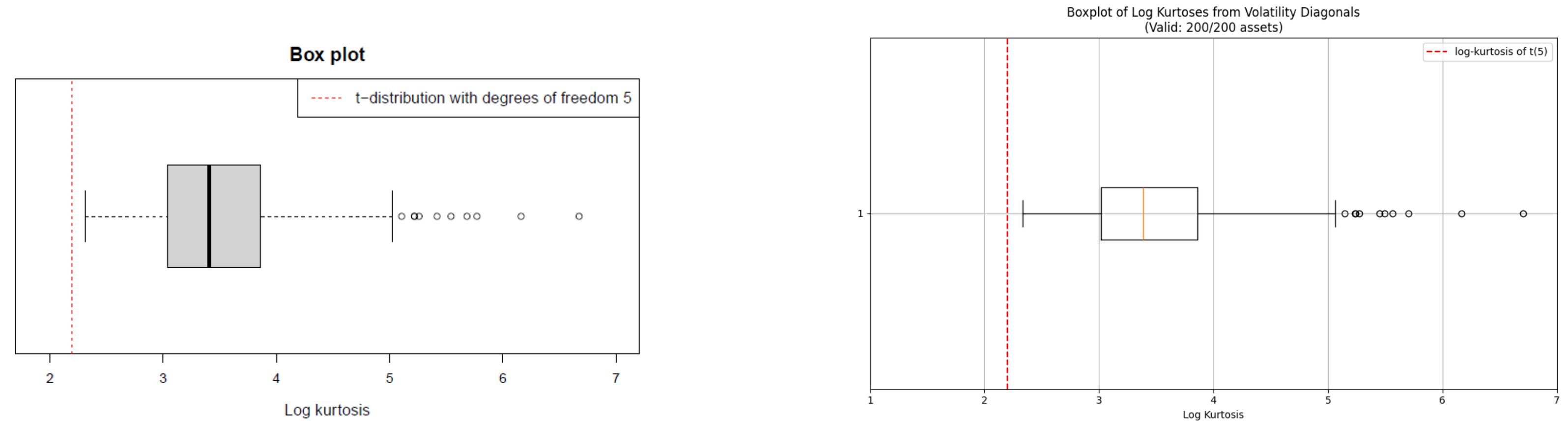


Figure 2: The boxplot of the 200 log kurtoses obtained from the daily jump adjusted pre-averaging realized volatility estimators for 997 trading days in the period 2016–2019. The daily jump adjusted pre-averaging realized volatility estimators are estimated using 1-min log-returns of the most liquid 200 assets in the S&P 500 index. The red dash represents the kurtosis of the t_5 -distribution.

Introduction

- **Concept:** Model both factor and idiosyncratic volatility matrix based on VAR (Vector Auto-Regressive) model with heavy-tailed innovations
- **Assumptions:**
 - Time invariant eigen-vectors
- **Approach:**
 - Decompose daily volatility into factor part and idiosyncratic part
 - Model dynamics of eigen values of each matrix with VAR structure
 - To handle heavy-tail and sparsity, use Huber-LASSO method
 - We call it FIVAR (Factor and Idiosyncratic Vector Auto-Regressive) model

FIVAR model

$$\xi_d = \nu + \sum_{k=1}^h A_k \xi_{d-k} + \epsilon_d \quad \text{a.s.}$$

where $\xi_d = (\xi_{d,1}, \dots, \xi_{d,p+r})^\top = \int_{d-1}^d \lambda_t(\theta) dt,$
 $\lambda_t(\theta) = (\lambda_{t,1}(\theta_1), \dots, \lambda_{t,p+r}(\theta_{p+r}))^\top,$
 $\nu = (\nu_1, \dots, \nu_{p+r})^\top, A_k = (A_{k,i,j})_{1 \leq i,j \leq p+r} \quad \text{for all } 1 \leq k \leq h,$
 $\epsilon_d = (\epsilon_{d,1}, \dots, \epsilon_{d,p+r})^\top$ is i.i.d. innovation at time $d,$
 $\mathbb{E}[\epsilon_d] = 0_{p+r}, \quad \text{which is independent of } \xi_{d-\ell} \text{ for all } \ell \in \mathbb{N}.$

Estimation procedure for the heavy-tailed VAR model

$$\ell_\tau(x) = \frac{x^2}{2} \cdot \mathbb{I}(|x| \leq \tau) + \left(\tau|x| - \frac{\tau^2}{2} \right) \cdot \mathbb{I}(|x| > \tau)$$

- Volatility often exhibits heavy tails in financial applications
- Huber loss is robust to outlier

$$\psi_\varpi(x) = x \cdot \mathbb{I}(|x| \leq \varpi) + \text{sign}(x) \cdot \varpi \cdot \mathbb{I}(|x| > \varpi)$$

- Limit the range of data by using truncation parameter (Winsorization)
- Two methods above makes model robust to heavy tailed high frequency financial data

Estimation procedure for the heavy-tailed VAR model

$$\hat{\beta}_i = \arg \min_{\beta_i \in \mathbb{R}^{h(p+r)+1}} \mathcal{L}_{\tau, \varpi}^{I,i}(\beta_i) + \eta_I \|\beta_i\|_1, \quad \text{for } i = r+1, \dots, p+r$$

$$\mathcal{L}_{\tau, \varpi}^{I,i}(\beta_i) = \frac{1}{n-h} \sum_{d=h+1}^n \ell_{\tau}^I \left(\hat{\xi}_{d,i} - \left\langle \psi_{\varpi}^I(\hat{\xi}_{d-1}^I), \beta_i \right\rangle \right)$$

Idiosyncratic Part

- Coefficient for L1 penalty is selected to minimize BIC (Bayesian Information Criterion)

$$\hat{\beta}_i = \arg \min_{\beta_i \in \mathbb{R}^{hr+1}} \mathcal{L}_{\tau, \varpi}^{F,i}(\beta_i), \quad \text{for } i = 1, \dots, r$$

$$\mathcal{L}_{\tau, \varpi}^{F,i}(\beta_i) = \frac{1}{n-h} \sum_{d=h+1}^n \ell_{\tau}^F \left(\hat{\xi}_{d,i} - \left\langle \psi_{\varpi}^F(\hat{\xi}_{d-1}^F), \beta_i \right\rangle \right)$$

Factor Part

- Since the sparsity of the coefficients for the factor part is known, it is a low-dimensional problem
- We do not need L1 penalty term

Large volatility matrix prediction

Pre-averaging Realized Volatility Matrix (PRVM)

- We use pre-averaging scheme to mitigate the effect of microstructure noise
- PRVM applies truncation thresholds to suppress the influence of extreme returns, providing robustness against heavy tails

$$\hat{\Gamma}_{d,ij} = \frac{1}{\psi K} \sum_{k=1}^{m-K+1} \left[\bar{Y}_i(t_{d,k}) \bar{Y}_j(t_{d,k}) - \frac{1}{2} \hat{Y}_{i,j}(t_{d,k}) \right] \cdot \mathbf{1}(|\bar{Y}_i(t_{d,k})| \leq u_{i,m}) \cdot \mathbf{1}(|\bar{Y}_j(t_{d,k})| \leq u_{j,m})$$

$$\bar{Y}_i(t_{d,k}) = \sum_{l=1}^{K-1} g\left(\frac{l}{K}\right) (Y_i(t_{d,k+l}) - Y_i(t_{d,k+l-1}))$$

$$\hat{Y}_{i,j}(t_{d,k}) = \sum_{l=1}^K \left[\left(g\left(\frac{l}{K}\right) - g\left(\frac{l-1}{K}\right) \right)^2 \cdot (Y_i(t_{d,k+l-1}) - Y_i(t_{d,k+l-2})) \cdot (Y_j(t_{d,k+l-1}) - Y_j(t_{d,k+l-2})) \right]$$

Large volatility matrix prediction

Factor Part

$$\Gamma_d = \Psi_d + \Sigma_d = \sum_{i=1}^r \xi_{d,i} q_i^F (q_i^F)^\top + \sum_{i=1}^p \xi_{d,i+r} q_i^I (q_i^I)^\top$$

- We assume that eigen vector is constant over time
- Average recent I days matrix and get time invariant eigen vector
- Divide obtained eigen value by p (number of assets) to normalize the range
- Project pre-averaged volatility matrix to PSD cone

$$\frac{1}{\ell} \sum_{d=n-\ell+1}^n \hat{\Gamma}_d \quad (\text{average of recent } \ell \text{ days' integrated volatility estimators})$$

Eigenvectors: $\hat{q}_1^F, \dots, \hat{q}_r^F$ (from the averaged matrix above)

$$\hat{\xi}_{d,i} = \frac{1}{p} (\hat{q}_i^F)^\top \hat{\Gamma}_d \hat{q}_i^F, \quad \text{for } d = 1, \dots, n, i = 1, \dots, r$$

Large volatility matrix prediction

$$\Gamma_d = \Psi_d + \Sigma_d = \sum_{i=1}^r \xi_{d,i} q_i^F (q_i^F)^\top + \sum_{i=1}^p \xi_{d,i+r} q_i^I (q_i^I)^\top$$

Idiosyncratic Part

- We first decompose the input volatility matrix (sort eigen values with descending order)
- Use hard thresholding scheme (keep if two assets are in same GICS sector, else set zero)

$$\bar{\Sigma}_d = \hat{\Gamma}_d - \sum_{k=1}^r \bar{\xi}_{d,k} \bar{q}_{d,k} \bar{q}_{d,k}^\top \quad \hat{\Sigma}_{d,ij} = \begin{cases} \bar{\Sigma}_{d,ij} \vee 0 & \text{if } i = j \\ g_{ij}(\bar{\Sigma}_{d,ij}) \cdot \mathbf{1}(|\bar{\Sigma}_{d,ij}| \geq v_{ij}) & \text{if } i \neq j \end{cases}$$

Large volatility matrix prediction

$$\Gamma_d = \Psi_d + \Sigma_d = \sum_{i=1}^r \xi_{d,i} q_i^F (q_i^F)^\top + \sum_{i=1}^p \xi_{d,i+r} q_i^I (q_i^I)^\top$$

Idiosyncratic Part

- We assume that eigen vector is constant over time
- Average recent I days matrix and get time invariant eigen vector

$$\hat{q}_1^I, \dots, \hat{q}_p^I \text{ from } \frac{1}{\ell} \sum_{d=n-\ell+1}^n \hat{\Sigma}_d$$

$$\hat{\xi}_{d,i+r} = (\hat{q}_i^I)^\top \hat{\Sigma}_d \hat{q}_i^I \quad \text{for } d = 1, \dots, n, i = 1, \dots, p$$

Large volatility matrix prediction

- With obtained volatility matrix, we can estimate the true model parameters using VAR model parameter estimation procedure

$$\hat{\xi}_{n+1} = [\hat{\xi}_{n+1,1}, \dots, \hat{\xi}_{n+1,p+r}]^\top = \hat{\nu} + \sum_{k=1}^h \hat{A}_k \hat{\xi}_{n+1-k}$$

$$\Gamma_{n+1} = \hat{\Psi}_{n+1} + \hat{\Sigma}_{n+1} = \sum_{i=1}^r \hat{\xi}_{n+1,i} \hat{q}_{Fi} \hat{q}_{Fi}^\top + \sum_{i=1}^p \hat{\xi}_{n+1,i+r} \hat{q}_{Ii} \hat{q}_{Ii}^\top$$

Numerical study

- Rank r has been estimated by following equation
- Rank r has been estimated to be 3 at all period
- Lag h has been estimated to be 1 by minimizing BIC (Bayesian Information Criterion)

$$\hat{r} = \arg \min_{1 \leq j \leq r_{\max}} \sum_{d=1}^n \left(\frac{1}{p} \bar{\xi}_{d,j} + j \cdot c_1 \left(\sqrt{\frac{\log p}{m^{1/2}}} + \frac{\log p}{p} \right)^{c_2} \right) - 1$$

Numerical study

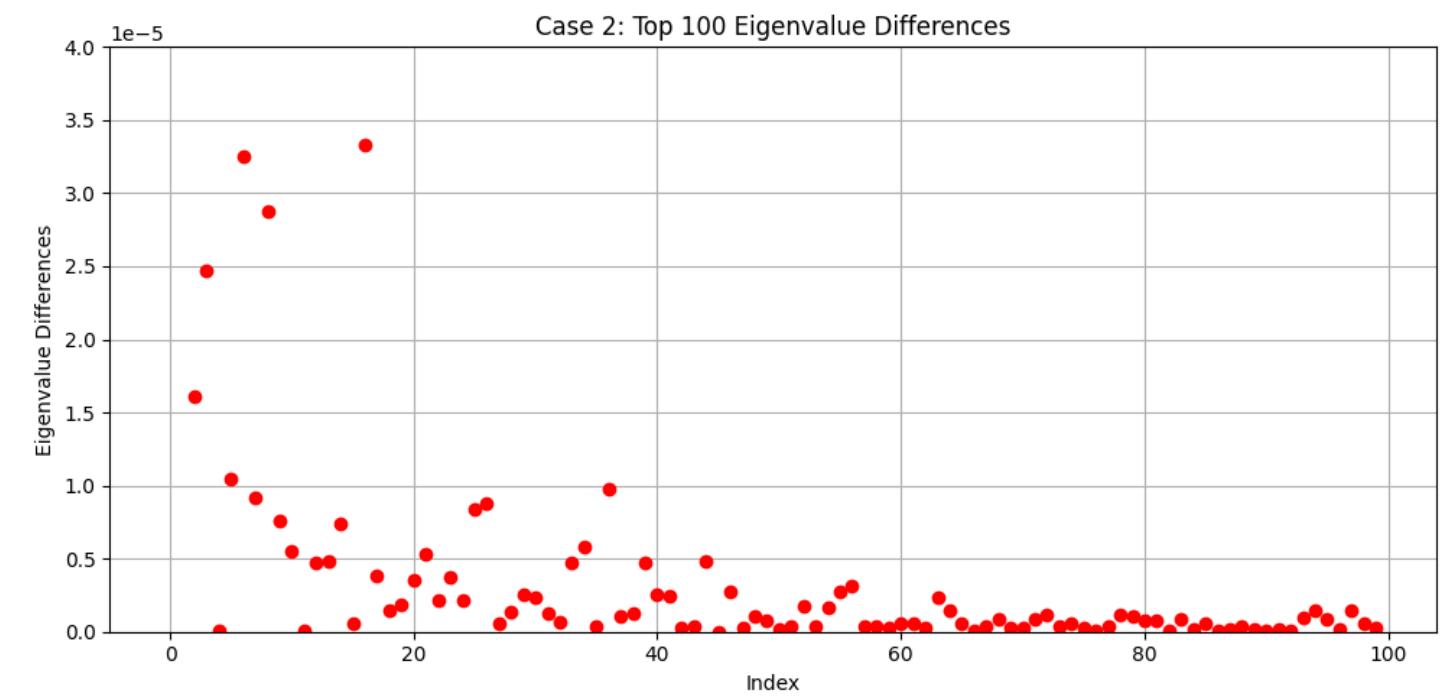
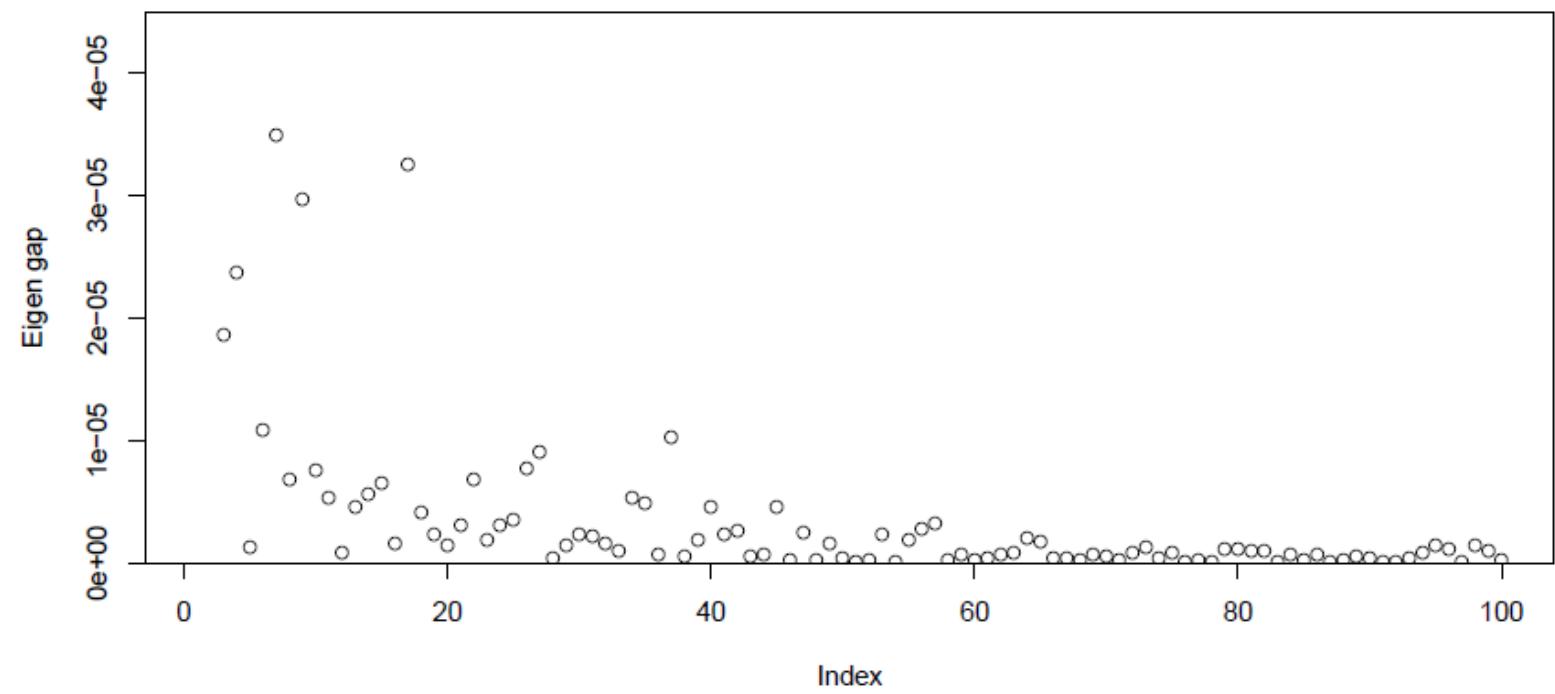


Figure 7: The plot of the first 100 differences between the consecutive eigenvalues of the average of 997 idiosyncratic volatility matrix estimators. We estimated the rank r based on the rank estimation procedure in (5.1) with $n = 997$. The result is $\hat{r} = 3$. We used 1-min log-returns of the top 200 large trading volume stocks among the S&P 500 from January 2016 to December 2019.

Numerical study

Comparison

- POET-PRVM: previous day's PRVM estimator from the POET procedure as the non-parametric benchmark
- OLS: Model factor part with OLS (Ordinary Least Square), average l days idiosyncratic matrix and use as idiosyncratic matrix for next day
- LASSO: Model both factor and idiosyncratic matrix using LASSO regression without truncation scheme (cannot account for heavy-tailedness)
- DCC-NL: Would not cover on this presentation
- HAR-DRD: Would not cover on this presentation

Numerical study

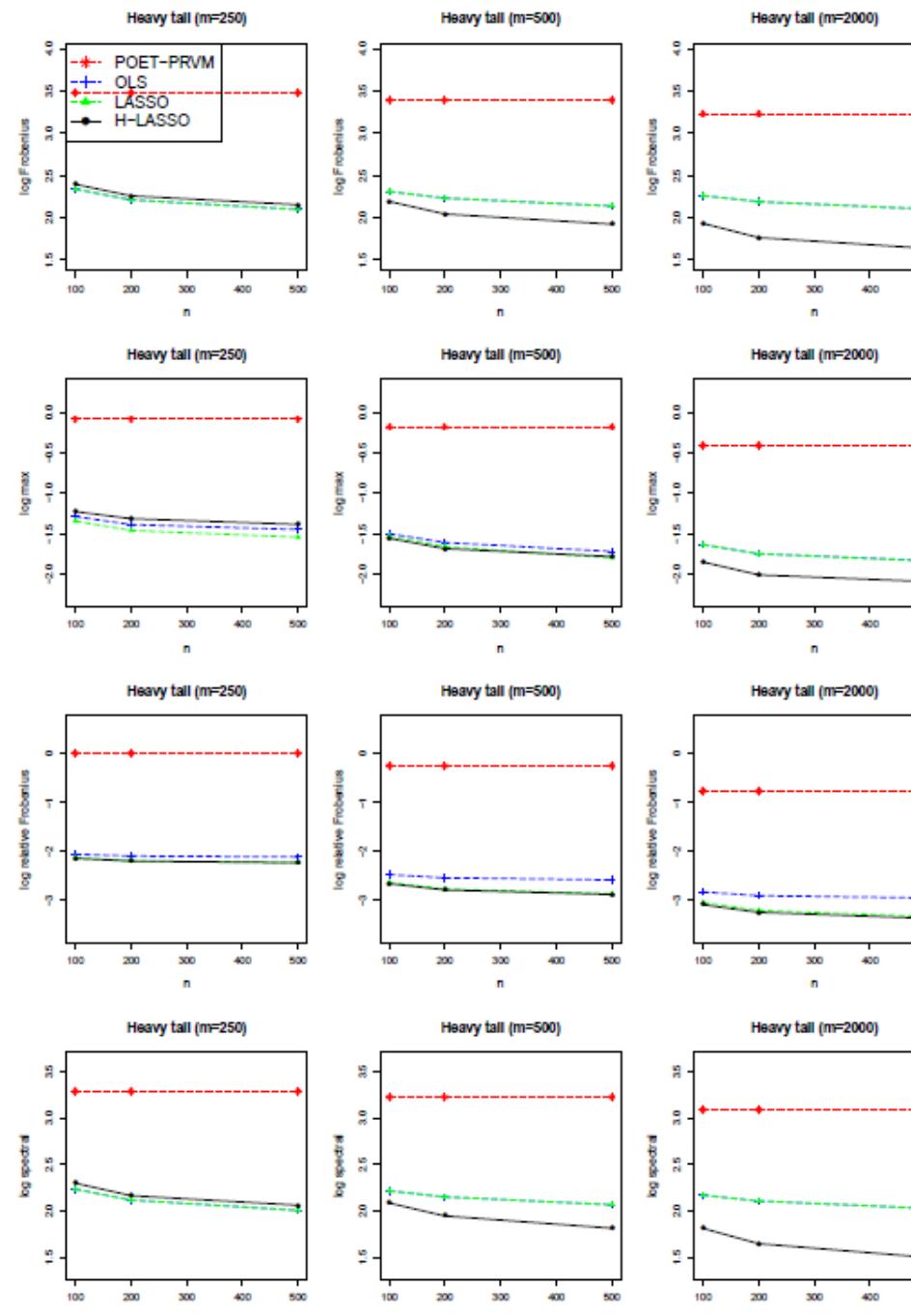


Figure 3: The log Frobenius, max, relative Frobenius, and spectral norm error plots of the POET-PRVM, OLS, LASSO, and H-LASSO estimators for the conditional expected integrated volatility matrix estimation with the heavy-tailed process, given $n = 100, 200, 500$ and $m = 250, 500, 2000$.

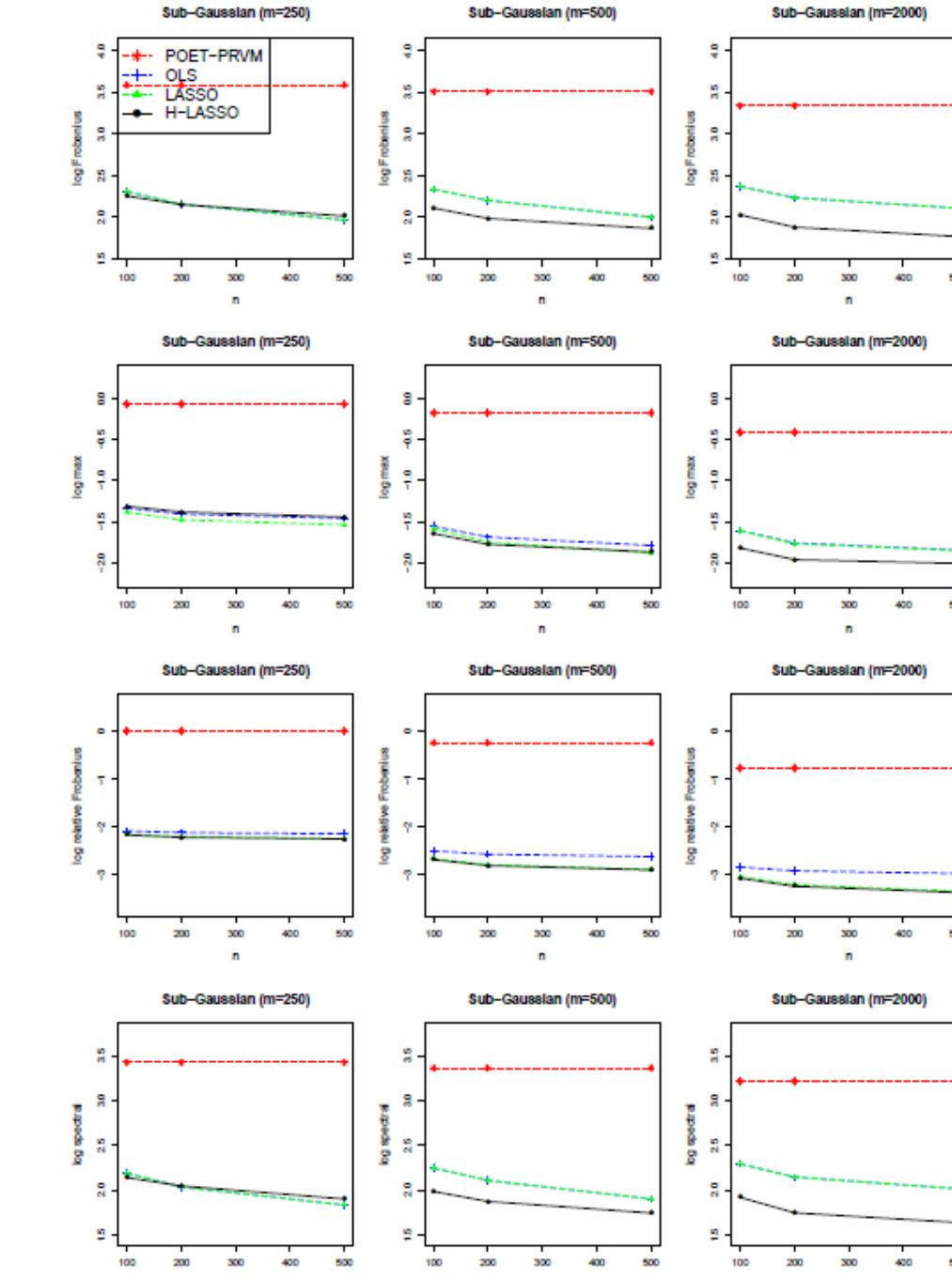


Figure 4: The log Frobenius, max, relative Frobenius, and spectral norm error plots of the POET-PRVM, OLS, LASSO, and H-LASSO estimators for the conditional expected integrated volatility matrix estimation with the sub-Gaussian process, given $n = 100, 200, 500$ and $m = 250, 500, 2000$.

Numerical study

$$\text{MSPE}(\tilde{\Gamma}) = \frac{1}{T} \sum_{d=1}^T \left\| \tilde{\Gamma}_d - \hat{\Gamma}_d^{\text{POET}} \right\|_F^2$$

$$\text{QLIKE}(\tilde{\Gamma}) = \frac{1}{T} \sum_{d=1}^T \left[\log \left(\det \left(\tilde{\Gamma}_d \right) \right) + \text{tr} \left(\Gamma_d^{-1} \hat{\Gamma}_d^{\text{POET}} \right) \right]$$

- We compute and compare MSPE and QLIKE to check the performance of estimation procedure
- POET-PRVM estimator acts as ground truth
- MSPE and QLIKE are both robust loss function to compare volatility

Numerical study

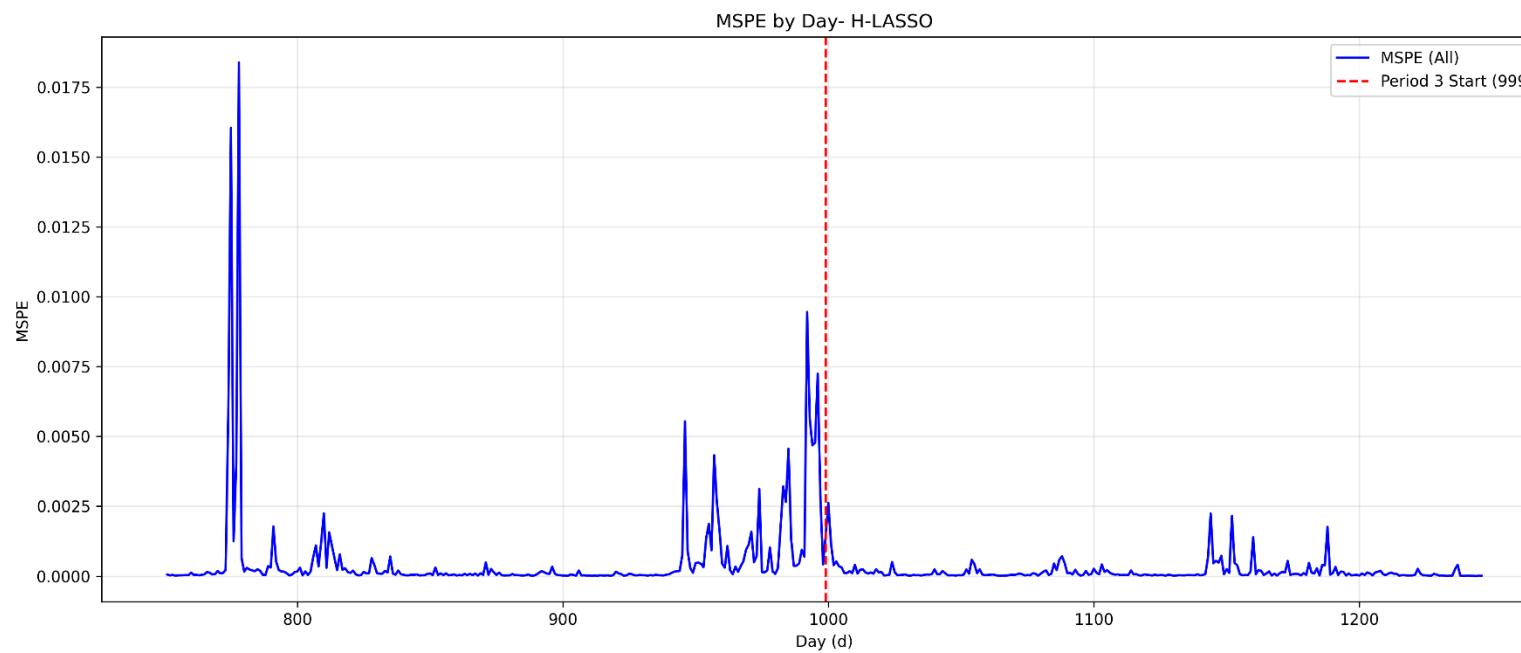
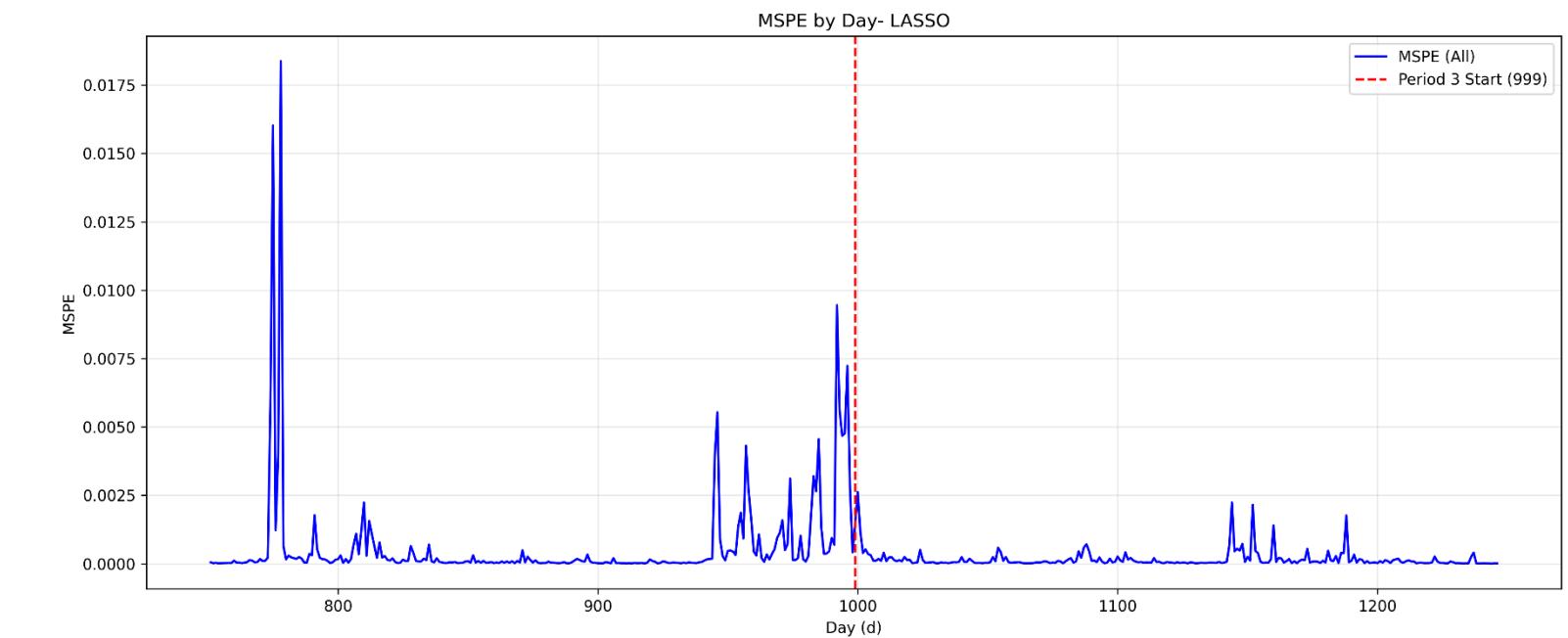
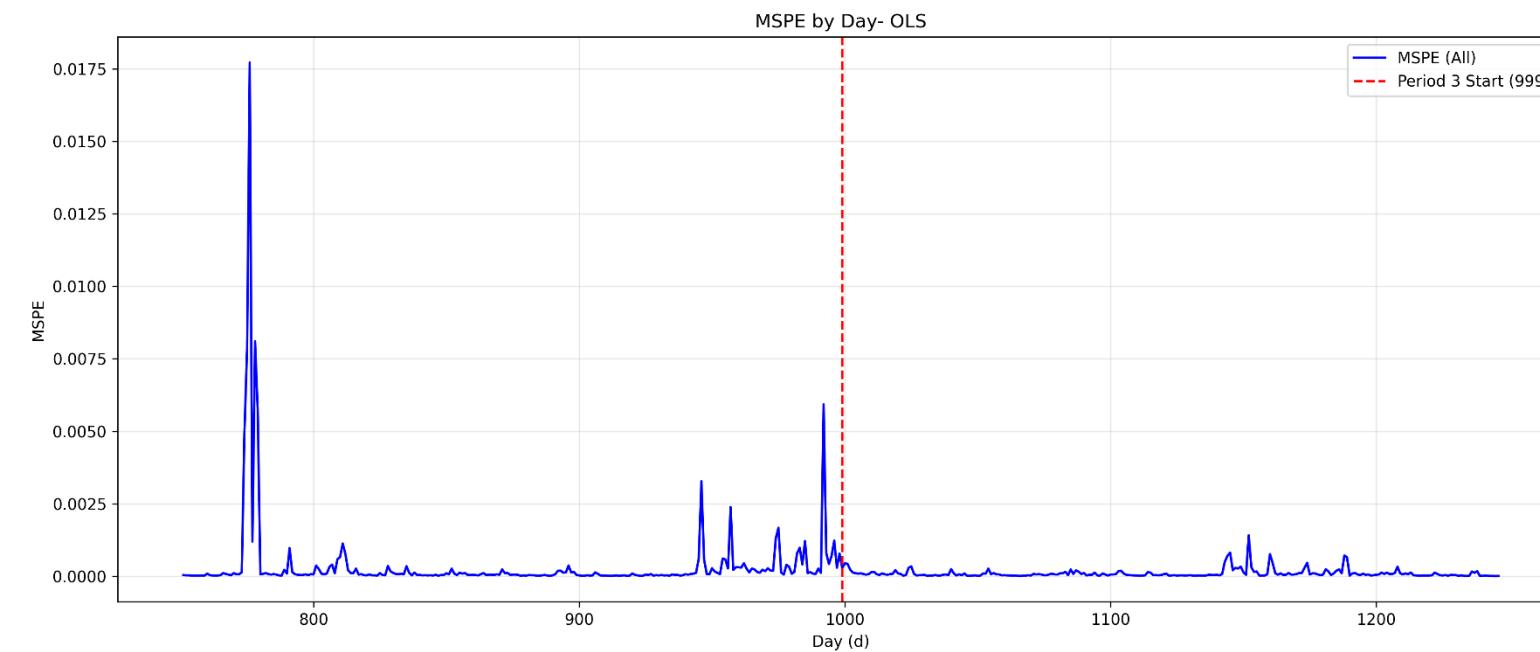
Table 2: The MSPE and QLIKE of the POET-PRVM, OLS, LASSO, H-LASSO, DCC-NL, and HAR-DRD estimators (period 1, from 2018 to 2019; period 2, 2018; period 3, 2019).

		POET-PRVM	OLS	LASSO	H-LASSO	DCC-NL	HAR-DRD
Period 1	MSPE $\times 10^4$	2.797	2.596	2.599	2.232	2.802	2.218
	QLIKE $\times 10^{-3}$	-1.192	-1.675	-1.683	-1.683	-1.635	-1.620
Period 2	MSPE $\times 10^4$	4.151	4.158	4.163	3.501	4.131	3.490
	QLIKE $\times 10^{-3}$	-1.169	-1.655	-1.660	-1.660	-1.617	-1.548
Period 3	MSPE $\times 10^4$	1.438	1.029	1.029	0.959	1.469	0.941
	QLIKE $\times 10^{-3}$	-1.215	-1.694	-1.706	-1.706	-1.652	-1.693

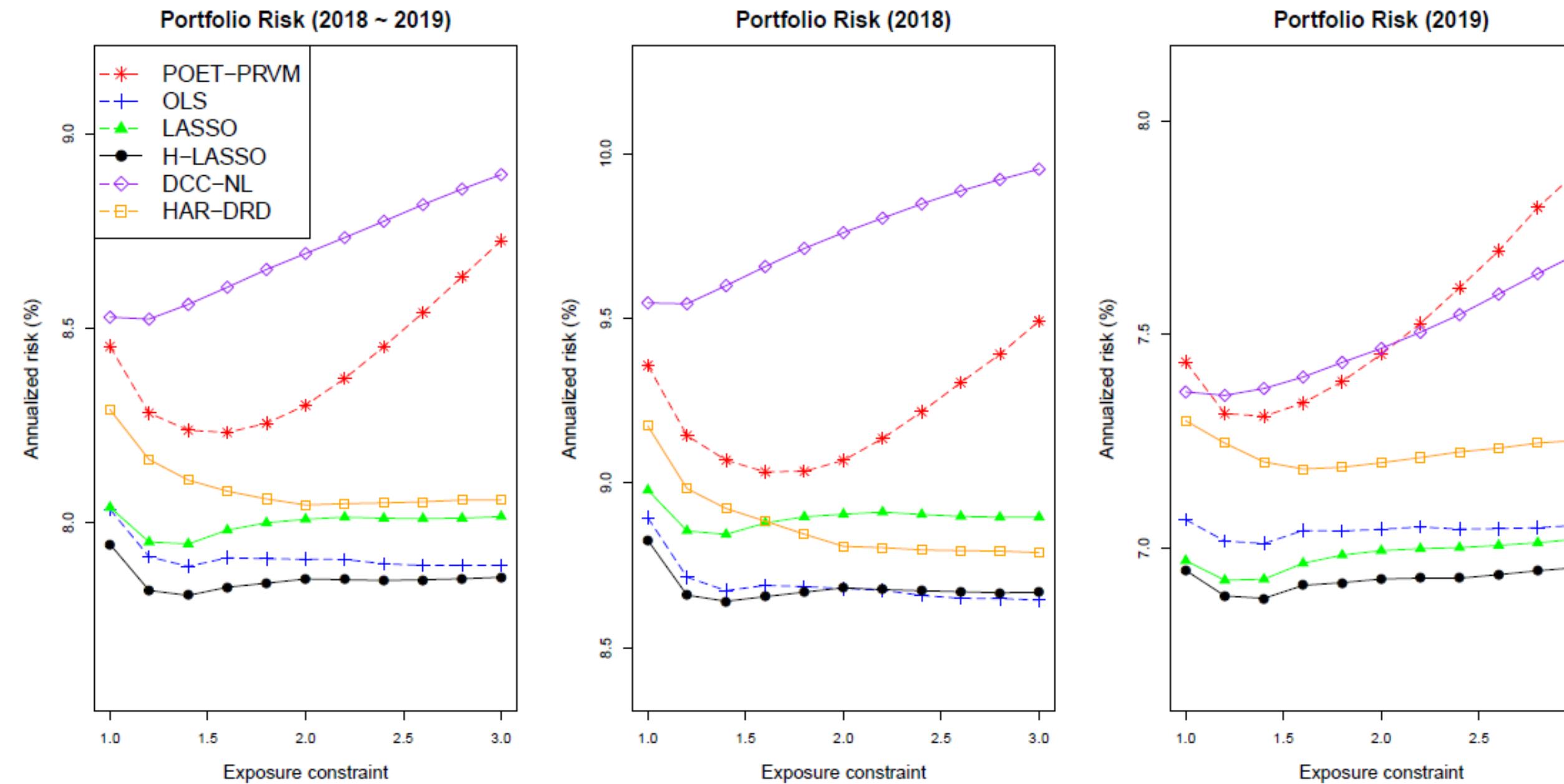
Experiment result (MSPE)

- POET-PRVM: 2.771 (0.93% diff), 4.111 (0.96% diff), 1.431 (0.49% diff) for each period
- OLS: 2.431 (6.35% diff), 3.823 (8.06% diff), 1.038 (0.87% diff) for each period
- Obtained bad value when estimating LASSO, H-LASSO (might had logical error when estimating idiosyncratic volatility matrix)

Numerical study



Numerical study



$$\min_{\omega} \omega^\top (\tilde{\Gamma}_d + J \tilde{V}_d) \omega, \text{ subject to } \omega^\top J = 1, \text{ and } \|\omega\|_1 \leq c_0$$

Further research direction

Future research mentioned in paper

- H-LASSO model assumes sparsity condition of model parameter; thus, it would be interesting to construct a test procedure for the sparsity condition
- We assumed that rank r is constant over time, but it may vary over time, also for eigen vector

Future research (Idea)

- Model eigen value dynamics using neural network
 - Learned that handling robustness is important when establishing model
- Apply scheme to more volatile assets (e.g. crypto currencies)