ONLINE APPENDIX FOR "MACHINE LEARNING VAST DYNAMIC CONDITIONAL COVARIANCE MATRICES: THE SPIRIT OF "DIVIDE AND CONQUER" "

ZHOUFAN ZHU [1,2] AND KE ZHU [3]

[1] *Wang Yanan Institute for Studies in Economics, Xiamen University, China*

[2] *Department of Finance, School of Economics, Xiamen University, China*

[3] *Department of Statistics and Actuarial Science, The University of Hong Kong, Hong Kong*

APPENDIX B: THE ARCHITECTURE OF GRU NETWORK

In this appendix, we introduce the architecture of two-layer GRU, which is used in (A1) to generate the temporal embedding $\boldsymbol{h}_{i,t}$. Let $\underline{\boldsymbol{x}}^s_{i,t-1} = \boldsymbol{x}_{i,t-1-S+s}$. The two-layer GRU modulates the flow of information recursively:

$$
\begin{aligned}
\boldsymbol{z}^s_{i,t,1} &= \text{sigmoid}\left(\boldsymbol{W}_{1z}\underline{\boldsymbol{x}}^s_{i,t-1} + \boldsymbol{U}_{1z}\boldsymbol{h}^{s-1}_{i,t,1} + \boldsymbol{b}_{1z}\right), \\
\boldsymbol{i}^s_{i,t,1} &= \text{sigmoid}\left(\boldsymbol{W}_{1i}\underline{\boldsymbol{x}}^s_{i,t-1} + \boldsymbol{U}_{1i}\boldsymbol{h}^{s-1}_{i,t,1} + \boldsymbol{b}_{1i}\right), \\
\boldsymbol{o}^s_{i,t,1} &= \tanh\left(\boldsymbol{W}_{1o}\underline{\boldsymbol{x}}^s_{i,t-1} + \boldsymbol{U}_{1o}\left(\boldsymbol{i}^s_{i,t,1} \odot \boldsymbol{h}^{s-1}_{i,t,1}\right) + \boldsymbol{b}_{1o}\right), \\
\boldsymbol{h}^s_{i,t,1} &= \boldsymbol{z}^s_{i,t,1} \odot \boldsymbol{h}^{s-1}_{i,t,1} + \left(1 - \boldsymbol{z}^s_{i,t,1}\right) \odot \boldsymbol{o}^s_{i,t,1},
\end{aligned} \tag{B1}
$$

$$
\begin{aligned}
\boldsymbol{z}^s_{i,t,2} &= \text{sigmoid}\left(\boldsymbol{W}_{2z}\boldsymbol{h}^s_{i,t,1} + \boldsymbol{U}_{2z}\boldsymbol{h}^{s-1}_{i,t,2} + \boldsymbol{b}_{2z}\right), \\
\boldsymbol{i}^s_{i,t,2} &= \text{sigmoid}\left(\boldsymbol{W}_{2i}\boldsymbol{h}^s_{i,t,1} + \boldsymbol{U}_{2i}\boldsymbol{h}^{s-1}_{i,t,2} + \boldsymbol{b}_{2i}\right), \\
\boldsymbol{o}^s_{i,t,2} &= \tanh\left(\boldsymbol{W}_{2o}\boldsymbol{h}^s_{i,t,1} + \boldsymbol{U}_{2o}\left(\boldsymbol{i}^s_{i,t,2} \odot \boldsymbol{h}^{s-1}_{i,t,2}\right) + \boldsymbol{b}_{2o}\right), \\
\boldsymbol{h}^s_{i,t,2} &= \boldsymbol{z}^s_{i,t,2} \odot \boldsymbol{h}^{s-1}_{i,t,2} + \left(1 - \boldsymbol{z}^s_{i,t,2}\right) \odot \boldsymbol{o}^s_{i,t,2},
\end{aligned} \tag{B2}
$$

where $\boldsymbol{W}_{1z}, \boldsymbol{W}_{1i}, \boldsymbol{W}_{1o} \in \mathcal{R}^{D\times P}$ and $\boldsymbol{U}_{1z}, \boldsymbol{U}_{1i}, \boldsymbol{U}_{1o} \in \mathcal{R}^{D\times D}$ are matrices of weight parameters in the first layer of GRU network, $\boldsymbol{b}_{1z}, \boldsymbol{b}_{1i}, \boldsymbol{b}_{1o} \in \mathcal{R}^{D}$ are vectors of bias parameters in the first layer

of GRU network, $\boldsymbol{W}_{2z}, \boldsymbol{W}_{2i}, \boldsymbol{W}_{2o}, \boldsymbol{U}_{2z}, \boldsymbol{U}_{2i}, \boldsymbol{U}_{2o} \in \mathcal{R}^{D\times D}$ are matrices of weight parameters in the second layer of GRU network, $\boldsymbol{b}_{2z}, \boldsymbol{b}_{2i}, \boldsymbol{b}_{2o} \in \mathcal{R}^{D}$ are vectors of bias parameters in the second layer of GRU network, sigmoid$(\cdot)$ and $\tanh(\cdot)$ are two element-wise vector-valued functions, $\odot$ is the element-wise production operation, and the initial values $\boldsymbol{h}^0_{i,t,1}$ and $\boldsymbol{h}^0_{i,t,2}$ are conventionally set to the D-dimensional vector of zeros.

Intuitively, equations in (B1) describe the flow of information from the raw input in the first layer, while equations in (B2) capture the potential temporal structure hidden in $\{\boldsymbol{h}^s_{i,t,1} : s = 1, \ldots, S\}$, thereby making the extracted temporal features more informative. Consequently, our temporal embedding $\boldsymbol{h}_{i,t}$ is taken as the final vector $\boldsymbol{h}^S_{i,t,2}$, that is,

$$\boldsymbol{h}_{i,t} = \boldsymbol{h}^S_{i,t,2} \in \mathcal{R}^D, \tag{B3}$$

and $\boldsymbol{\phi}$ in (A1) includes all network parameters in (B1)–(B2).

APPENDIX C: THE QCM LEARNING FOR CONDITIONAL VARIANCE

C.1. The QCM Learning for Conditional Variance $\boldsymbol{\sigma^2_{i,t}}$. Given L different quantile levels $\tau_1, ..., \tau_L$, let $\widehat{\boldsymbol{Q}}_t(\tau_1), ..., \widehat{\boldsymbol{Q}}_t(\tau_L)$ be the vectors of corresponding learned conditional quantiles via L different SGA-based quantile models, where $\widehat{\boldsymbol{Q}}_t(\tau_l) = (\widehat{Q}_{1,t}(\tau_l), \ldots, \widehat{Q}_{N,t}(\tau_l))'$ for $l = 1, \ldots, L$. For the fixed values of i and t, Cornish and Fisher (1938) introduce the following fundamental connection between conditional quantiles and conditional moments:

$$\begin{aligned} Q_{i,t}(\tau_l) = \mu_{i,t} + z(\tau_l)\sqrt{\sigma^2_{i,t}} + \left[z(\tau_l)^2 - 1\right]\frac{\sqrt{\sigma^2_{i,t}}s_{i,t}}{6} \\ + \left[z(\tau_l)^3 - 3z(\tau_l)\right]\frac{\sqrt{\sigma^2_{i,t}}(k_{i,t}-3)}{24} + \sqrt{\sigma^2_{i,t}}\omega_{i,t}(\tau_l) \end{aligned} \tag{C1}$$

for $l = 1, ..., L$, where $z(\tau_l)$ is the τ_l-th quantile of standard normal distribution, $s_{i,t}$ and $k_{i,t}$ are the conditional skewness and kurtosis of $r_{i,t}$, respectively, and $\omega_{i,t}(\tau_l)$ is the remainder of

this expansion. Since $Q_{i,t}(\tau_l)$ is unobserved in (C1), the QCM method replaces $Q_{i,t}(\tau_l)$ by the estimator $\widehat{Q}_{i,t}(\tau_l)$, yielding

$$\text{(C2)} \qquad \widehat{Q}_{i,t}(\tau_l) = \mu_{i,t} + \boldsymbol{Z}_l'\boldsymbol{\beta}_{i,t} + \varepsilon^{\bullet}_{i,t,l} \quad \text{for } l = 1, \ldots, L,$$

where

$$\begin{aligned}
&\varepsilon^{\bullet}_{i,t,l} = \varepsilon^{*}_{i,t,l} + \varepsilon^{\circ}_{i,t,l} \text{ with } \varepsilon^{*}_{i,t,l} = \sqrt{\sigma^2_{i,t}}\omega_{i,t}(\tau_l) \text{ and } \varepsilon^{\circ}_{i,t,l} = \widehat{Q}_{i,t}(\tau_l) - Q_{i,t}(\tau_l),\\
&\boldsymbol{Z}_l = \big(z(\tau_l), z(\tau_l)^2 - 1, z(\tau_l)^3 - 3z(\tau_l)\big)',\\
&\boldsymbol{\beta}_{i,t} \equiv (\beta_{i,t,1}, \beta_{i,t,2}, \beta_{i,t,3})' = \Big(\sqrt{\sigma^2_{i,t}}, \frac{\sqrt{\sigma^2_{i,t}}s_{i,t}}{6}, \frac{\sqrt{\sigma^2_{i,t}}(k_{i,t} - 3)}{24}\Big)'.
\end{aligned}$$

Here, $\varepsilon^{\bullet}_{i,t,l}$ in (C2) contains the errors from two sources: the expansion error $\varepsilon^{*}_{i,t,l}$ and the quantile estimation error $\varepsilon^{\circ}_{i,t,l}$.

Clearly, model (C2) can be regarded as a linear regression model with the response variable $\widehat{Q}_{i,t}(\tau_l)$, explanatory variables $\boldsymbol{Z}_l$, parameter vector $(\mu_{i,t}, \boldsymbol{\beta}_{i,t}')'$, and error term $\varepsilon^{\bullet}_{i,t,l}$. Since $\varepsilon^{\bullet}_{i,t,l}$ may not have zero mean for model identification, the QCM method adds a deterministic intercept term $\gamma_{i,t}$ into the equation (C2) to form the following linear regression model:

$$\text{(C3)} \qquad \begin{aligned}
\widehat{Q}_{i,t}(\tau_l) &= (\mu_{i,t} + \gamma_{i,t}) + \boldsymbol{Z}_l'\boldsymbol{\beta}_{i,t} + \varepsilon_{i,t,l}\\
&\equiv \bar{\boldsymbol{Z}}_l'\boldsymbol{\theta}_{i,t} + \varepsilon_{i,t,l} \quad \text{for } l = 1, \ldots, L,
\end{aligned}$$

where $\varepsilon_{i,t,l} = \varepsilon^{\bullet}_{i,t,l} - \gamma_{i,t}$, $\bar{\boldsymbol{Z}}_l = (1, \boldsymbol{Z}_l')'$, and $\boldsymbol{\theta}_{i,t} = (\beta_{i,t,0}, \boldsymbol{\beta}_{i,t}')'$ with $\beta_{i,t,0} = \mu_{i,t} + \gamma_{i,t}$. In this way, the ordinary least squares (OLS) estimator of $\boldsymbol{\theta}_{i,t}$ in (C3) is given by

$$\text{(C4)} \qquad \widehat{\boldsymbol{\theta}}_{i,t} \equiv (\widehat{\beta}_{i,t,0}, \widehat{\boldsymbol{\beta}}_{i,t}')' = (\bar{\boldsymbol{Z}}'\bar{\boldsymbol{Z}})^{-1}\bar{\boldsymbol{Z}}'\boldsymbol{Y}_{i,t},$$

where $\widehat{\boldsymbol{\beta}}_{i,t} = (\widehat{\beta}_{i,t,1}, \widehat{\beta}_{i,t,2}, \widehat{\beta}_{i,t,3})'$, $\boldsymbol{Y}_{i,t}$ is a $L \times 1$ vector with entries $\widehat{Q}_{i,t}(\tau_l)$, and $\bar{\boldsymbol{Z}}$ is a $L \times 4$ matrix with rows $\bar{\boldsymbol{Z}}_l'$. Zhang and Zhu (2026) show that $\widehat{\boldsymbol{\theta}}_{i,t} - \boldsymbol{\theta}_{i,t} \longrightarrow 0$ in probability as $L \to \infty$ under the following two classical conditions in the regression literature:

CONDITION C1. *$\bar{\boldsymbol{Z}}'\bar{\boldsymbol{Z}}$ is positive definite.*

CONDITION C2. *$\bar{\boldsymbol{Z}}'\boldsymbol{\varepsilon}_{i,t}/L \longrightarrow \mathbf{0}$ in probability as $L \to \infty$, where $\boldsymbol{\varepsilon}_{i,t} = (\varepsilon_{i,t,1}, \ldots, \varepsilon_{i,t,L})'$.*

Consequently, by the continuous mapping theorem, the limiting behaviors of estimated conditional moments can be established.

PROPOSITION 1. *Suppose that Conditions C1–C2 hold. Let*

$$\widehat{\sigma}_{i,t}^2 \equiv \widehat{\beta}_{i,t,1}^2, \quad \widehat{s}_{i,t} \equiv \frac{6\widehat{\beta}_{i,t,2}}{\widehat{\beta}_{i,t,1}}, \quad \textit{and} \quad \widehat{k}_{i,t} \equiv \frac{24\widehat{\beta}_{i,t,3}}{\widehat{\beta}_{i,t,1}} + 3. \tag{C5}$$

Then,

$$\widehat{\sigma}_{i,t}^2 - \sigma_{i,t}^2 \longrightarrow 0, \quad \widehat{s}_{i,t} - s_{i,t} \longrightarrow 0, \quad \textit{and} \quad \widehat{k}_{i,t} - k_{i,t} \longrightarrow 0$$

in probability as $L \to \infty$.

In (C5), $\widehat{\sigma}_{i,t}^2$, $\widehat{s}_{i,t}$, and $\widehat{k}_{i,t}$ are the so-called QCM estimators of $\sigma_{i,t}^2$, $s_{i,t}$, and $k_{i,t}$, respectively. Due to its regression nature, the QCM method has two remarkable advantages (Zhu, Zhang and Zhu, 2024; Zhang and Zhu, 2026). First, it has no need to pre-estimate the conditional mean $\mu_{i,t}$, since $\mu_{i,t}$ is absorbed into the intercept parameter $\beta_{i,t,0}$, and thus has no impact on the QCM estimators. Second, it allows the use of wrongly specified conditional quantile models to some extent. This is because the biases of estimated conditional quantiles can be aggregately offset by the term $\gamma_{i,t}$ nested in $\beta_{i,t,0}$.

Notably, $\mu_{i,t}$ cannot be estimated using the QCM method. This is because $\mu_{i,t}$ cannot be accurately estimated by $\widehat{\beta}_{i,t,0}$ in (C4) due to the presence of $\gamma_{i,t}$. Therefore, $\mu_{i,t}$ must be estimated separately using other methods (see, e.g., the graph-based method in Appendix D).

C.2. The QCM Learning for Conditional Covariance $\boldsymbol{\delta_{i,j,t}^2}$. Recall that $\delta_{i,j,t}^2$ is the conditional variance of $r_{i,j,t}$. Analogously, we let $\mu_{i,j,t}$, $s_{i,j,t}$, and $k_{i,j,t}$ be the conditional mean,

skewness, and kurtosis of $r_{i,j,t}$, respectively. As for (C3), we have

$$\widehat{Q}_{i,j,t}(\tau_l) = \bar{\boldsymbol{Z}}_l' \boldsymbol{\theta}_{i,j,t} + \varepsilon_{i,j,t,l} \quad \text{for } l = 1, \ldots, L,$$

where

$$\boldsymbol{\theta}_{i,j,t} \equiv (\beta_{i,j,t,0}, \beta_{i,j,t,1}, \beta_{i,j,t,2}, \beta_{i,j,t,3})' = \left(\mu_{i,j,t} + \gamma_{i,j,t}, \sqrt{\delta^2_{i,j,t}}, \frac{\sqrt{\delta^2_{i,j,t}} s_{i,j,t}}{6}, \frac{\sqrt{\delta^2_{i,j,t}}(k_{i,j,t} - 3)}{24}\right)',$$

$$\varepsilon_{i,j,t,l} = \varepsilon^{\bullet}_{i,j,t,l} - \gamma_{i,j,t} \text{ with } \varepsilon^{\bullet}_{i,j,t,l} = \varepsilon^{*}_{i,j,t,l} + \varepsilon^{\circ}_{i,j,t,l},$$

$$\varepsilon^{*}_{i,j,t,l} = \sqrt{\delta^2_{i,j,t}}\omega_{i,j,t}(\tau_l) \text{ and } \varepsilon^{\circ}_{i,j,t,l} = \widehat{Q}_{i,j,t}(\tau_l) - Q_{i,j,t}(\tau_l).$$

Here, $\omega_{i,j,t}(\tau_l)$ is the remainder of Cornish-Fisher expansion for $Q_{i,j,t}(\tau_l)$, and $\gamma_{i,j,t}$ is the nuisance parameter deduced by $\varepsilon^{\bullet}_{i,j,t,l}$. Let

$$\boldsymbol{Y}_{i,j,t} = \left(\widehat{Q}_{i,j,t}(\tau_1), \ldots, \widehat{Q}_{i,j,t}(\tau_L)\right)'.$$

Then, the OLS estimator of $\boldsymbol{\theta}_{i,j,t}$ is given by

$$\widehat{\boldsymbol{\theta}}_{i,j,t} \equiv (\widehat{\beta}_{i,j,t,0}, \widehat{\beta}_{i,t,1}, \widehat{\beta}_{i,t,2}, \widehat{\beta}_{i,t,3})' = (\bar{\boldsymbol{Z}}'\bar{\boldsymbol{Z}})^{-1}\bar{\boldsymbol{Z}}'\boldsymbol{Y}_{i,j,t}.$$

To ensure $\widehat{\boldsymbol{\theta}}_{i,j,t} - \boldsymbol{\theta}_{i,j,t} \longrightarrow 0$ in probability, we impose an additional condition:

CONDITION C3. $\bar{\boldsymbol{Z}}'\boldsymbol{\varepsilon}_{i,j,t}/L \longrightarrow \mathbf{0}$ *in probability as* $L \to \infty$, *where* $\boldsymbol{\varepsilon}_{i,j,t} = (\varepsilon_{i,j,t,1}, \ldots, \varepsilon_{i,j,t,L})'$.

Consequently, by the continuous mapping theorem, the limiting behaviors of estimated conditional moments can be established.

PROPOSITION 2. *Suppose that Conditions C1 and C3 hold. Let*

$$\text{(C6)} \qquad \widehat{\delta}^2_{i,j,t} \equiv \widehat{\beta}^2_{i,j,t,1}, \quad \widehat{s}_{i,j,t} \equiv \frac{6\widehat{\beta}_{i,j,t,2}}{\widehat{\beta}_{i,j,t,1}}, \quad \textit{and} \quad \widehat{k}_{i,j,t} \equiv \frac{24\widehat{\beta}_{i,j,t,3}}{\widehat{\beta}_{i,j,t,1}} + 3.$$

Then,

$$\widehat{\delta}^2_{i,j,t} - \delta^2_{i,j,t} \longrightarrow 0, \quad \widehat{s}_{i,j,t} - s_{i,j,t} \longrightarrow 0, \quad \textit{and} \quad \widehat{k}_{i,j,t} - k_{i,j,t} \longrightarrow 0$$

in probability as $L \to \infty$.

Finally, combining the results of Propositions 1–2, we can show the consistency of $\widehat{\sigma}_{i,j,t}$:

PROPOSITION 3. *Suppose that Conditions C1–C3 hold. Then,*

$$\widehat{\sigma}_{i,j,t} = \frac{1}{2}\left(\widehat{\delta}^2_{i,j,t} - \widehat{\sigma}^2_{i,t} - \widehat{\sigma}^2_{j,t}\right) \longrightarrow \sigma_{i,j,t} \text{ in probability as } L \to \infty,$$

where $\widehat{\sigma}^2_{i,t}$ and $\widehat{\delta}^2_{i,j,t}$ are defined in (C5) and (C6), respectively.

Based on Propositions 1 and 3, we have the following theorem for the GRAND estimator:

THEOREM 1. *Suppose that Conditions C1–C3 hold. Then, as $L \to \infty$, we have (i) $\widehat{\sigma}^2_{i,t} \to \sigma^2_{i,t}$ in probability for each fixed t and i; (ii) $\widehat{\sigma}_{i,j,t} \to \sigma_{i,j,t}$ in probability for each fixed t and $i \neq j$.*

Theorem 1 ensures the consistency of each element of the GRAND CCM estimator $\widetilde{\boldsymbol{\Sigma}}_t$. Fan, Fan and Lv (2008) and Cui et al. (2024) derive the convergence rate of factor-based CCM estimators. However, we find it challenging to derive the convergence rate of $\widetilde{\boldsymbol{\Sigma}}_t$ at this stage, since the learning procedure of $\widetilde{\boldsymbol{\Sigma}}_t$ depends on complex deep neural networks and the positive semidefinite matrix projection. We leave this interesting topic for future work.

APPENDIX D: LEARNING FOR CONDITIONAL MEAN

Let $\boldsymbol{\mu}_t \equiv \mathbb{E}(\boldsymbol{r}_t|\mathcal{F}_{t-1}) \in \mathcal{R}^N$ be the conditional mean of $\boldsymbol{r}_t$, where $\boldsymbol{\mu}_t = (\mu_{1,t}, \ldots, \mu_{N,t})'$. In this appendix, we propose the following SGA-based mean model:

$$\text{(D1)} \qquad \boldsymbol{r}_t = f(\boldsymbol{X}_{t-1}; \mathcal{G}_{t,\mu}, \boldsymbol{\theta}_\mu) + \boldsymbol{\epsilon}_t \text{ with } \mathcal{G}_{t,\mu} = q(\boldsymbol{X}_{t-1}; \boldsymbol{\phi}_\mu),$$

where $q(\cdot)$ and $f(\cdot)$ are the encoder and decoder defined as in (A4) and (A9), respectively, $\mathcal{G}_{t,\mu}$ represents the Granger causal graph for predicting the conditional mean, $\boldsymbol{\phi}_\mu$ and $\boldsymbol{\theta}_\mu$ are trainable parameters in the encoder and decoder, respectively, and $\boldsymbol{\epsilon}_t \in \mathcal{R}^N$ is the vector of

error terms with zero mean. To optimize model (D1) based on the mean squared error loss, we leverage the Adam algorithm (see Algorithm E4) to compute

$$(\widehat{\boldsymbol{\phi}}_\mu, \widehat{\boldsymbol{\theta}}_\mu) = \underset{\boldsymbol{\phi}_\mu, \boldsymbol{\theta}_\mu}{\operatorname{argmin}} \sum_{t\in\mathcal{T}} \sum_{i=1}^{N} \left[r_{i,t} - f_i(\boldsymbol{X}_{t-1}; q(\boldsymbol{X}_{t-1}; \boldsymbol{\phi}_\mu), \boldsymbol{\theta}_\mu) \right]^2$$
$$\equiv \underset{\boldsymbol{\phi}_\mu, \boldsymbol{\theta}_\mu}{\operatorname{argmin}} \sum_{t\in\mathcal{T}} \ell_\mu(\boldsymbol{r}_t, \boldsymbol{X}_{t-1}; \boldsymbol{\phi}_\mu, \boldsymbol{\theta}_\mu),$$

where $\mathcal{T}$ is the set of timepoints for training, and $f_i(\cdot)$ is the i-th element of $f(\cdot)$. After obtaining $\widehat{\boldsymbol{\phi}}_\mu$ and $\widehat{\boldsymbol{\theta}}_\mu$, we then learn $\boldsymbol{\mu}_t$ and $\mathcal{G}_{t,\mu}$ by $\widehat{\boldsymbol{\mu}}_t$ and $\widehat{\mathcal{G}}_{t,\mu}$, respectively, where

$$\widehat{\boldsymbol{\mu}}_t = f(\boldsymbol{X}_{t-1}; \widehat{\mathcal{G}}_{t,\mu}, \widehat{\boldsymbol{\theta}}_\mu) \text{ and } \widehat{\mathcal{G}}_{t,\mu} = q(\boldsymbol{X}_{t-1}; \widehat{\boldsymbol{\phi}}_\mu).$$

APPENDIX E: ALGORITHMS

This appendix provides detailed descriptions of four Adam algorithms. Specifically, Algorithms E1, E3, and E4 are applied to train the SGA-based quantile, variance, and mean models, respectively, and Algorithm E2 is designed to train the node fusion model.

Algorithm E1 The training procedure of $\widehat{\boldsymbol{\phi}}_\tau$ and $\widehat{\boldsymbol{\theta}}_\tau$ by the Adam algorithm.

Input:

The sample: $\{(\boldsymbol{r}_t, \boldsymbol{X}_{t-1})\}$;

The initial value of network parameters: $\boldsymbol{\phi}_\tau^{(0)}$ and $\boldsymbol{\theta}_\tau^{(0)}$;

Hyperparameter: learning rate γ;

$l = 0$;

repeat

  $(\boldsymbol{r}_l, \boldsymbol{X}_{l-1}) \leftarrow$ draw a random data point from $\{(\boldsymbol{r}_t, \boldsymbol{X}_{t-1})\}$ (A cross-sectional minibatch);

  $\boldsymbol{g}_\tau^{(l)} \leftarrow \nabla_{\boldsymbol{\phi}_\tau, \boldsymbol{\theta}_\tau} \left[\ell(\boldsymbol{r}_l, \boldsymbol{X}_{l-1}; \boldsymbol{\phi}_\tau^{(l)}, \boldsymbol{\theta}_\tau^{(l)}, \tau) \right]$ (Gradients of minibatch estimator);

  $\boldsymbol{\phi}_\tau^{(l+1)}, \boldsymbol{\theta}_\tau^{(l+1)} \leftarrow$ update parameters using learning rate γ and gradients $\boldsymbol{g}_\tau^{(l)}$ (Adam);

  $l \leftarrow l + 1$;

until convergence of parameters $\boldsymbol{\phi}_\tau^{(l+1)}$ and $\boldsymbol{\theta}_\tau^{(l+1)}$;

Output:

The values of $\boldsymbol{\phi}_\tau^{(l+1)}$ and $\boldsymbol{\theta}_\tau^{(l+1)}$, which are taken as the quantile estimators $\widehat{\boldsymbol{\phi}}_\tau$ and $\widehat{\boldsymbol{\theta}}_\tau$, respectively.

We further examine two alternative implementations of GRAND to balance the computational costs and accuracy in Algorithms E1 and E2. First, to reduce computational burden,

Algorithm E2 The training procedure of $\widehat{\boldsymbol{\psi}}_\tau$ by the Adam algorithm.

Input:

The sample: $\{(\boldsymbol{r}_t, \boldsymbol{X}_{t-1})\}$;

The initial value of network parameters: $\boldsymbol{\psi}_\tau^{(0)}$;

The pre-trained value of network parameters: $\widehat{\boldsymbol{\phi}}_\tau$ and $\widehat{\boldsymbol{\theta}}_\tau$;

Hyperparameter: learning rate γ, batch size M;

$\mathcal{J} \leftarrow \{(i,j) : 1 \le i < j \le N\}$ (all possible pairs);

$l = 0$;

repeat

  $(\boldsymbol{r}_l, \boldsymbol{X}_{l-1}) \leftarrow$ draw a random data point from $\{(\boldsymbol{r}_t, \boldsymbol{X}_{t-1})\}$ (A cross-sectional minibatch);

  $B^{(l)} \leftarrow$ the minibatch from $\mathcal{J}$ including M elements;

  $\boldsymbol{g}_\tau^{(l)} \leftarrow \nabla_{\boldsymbol{\psi}_\tau} \left[\sum_{(i,j)\in B^{(l)}} \ell_f(\boldsymbol{r}_t, \boldsymbol{X}_{t-1}; \boldsymbol{\psi}_\tau^{(l)}, \widehat{\boldsymbol{\phi}}_\tau, \widehat{\boldsymbol{\theta}}_\tau, \tau, i, j)\right]$ (Gradients of minibatch estimator);

  $\boldsymbol{\psi}_\tau^{(l+1)} \leftarrow$ update parameters using learning rate γ and gradients $\boldsymbol{g}_\tau^{(l)}$ (Adam);

  $l \leftarrow l + 1$;

until convergence of parameters $\boldsymbol{\psi}_\tau^{(l+1)}$;

Output:

The value of $\boldsymbol{\psi}_\tau^{(l+1)}$, which is taken as the quantile estimator $\widehat{\boldsymbol{\psi}}_\tau$.

Algorithm E3 The training procedure of $\widehat{\boldsymbol{\phi}}_\sigma$ and $\widehat{\boldsymbol{\theta}}_\sigma$ by the Adam algorithm.

Input:

The sample: $\{(\widehat{\boldsymbol{\sigma}}_t^2, \boldsymbol{X}_{t-1}^*)\}$;

The initial value of network parameters: $\boldsymbol{\phi}_\sigma^{(0)}$ and $\boldsymbol{\theta}_\sigma^{(0)}$;

Hyperparameter: learning rate γ;

$l = 0$;

repeat

  $(\widehat{\boldsymbol{\sigma}}_t^2, \boldsymbol{X}_{t-1}^*) \leftarrow$ draw a random data point from $\{(\widehat{\boldsymbol{\sigma}}_t^2, \boldsymbol{X}_{t-1}^*)\}$ (A cross-sectional minibatch);

  $\boldsymbol{g}_\sigma^{(l)} \leftarrow \nabla_{\boldsymbol{\phi}_\sigma, \boldsymbol{\theta}_\sigma} \left[\ell_\sigma(\widehat{\boldsymbol{\sigma}}_t^2, \boldsymbol{X}_{t-1}^*; \boldsymbol{\phi}_\sigma^{(l)}, \boldsymbol{\theta}_\sigma^{(l)})\right]$ (Gradients of minibatch estimator);

  $\boldsymbol{\phi}_\sigma^{(l+1)}, \boldsymbol{\theta}_\sigma^{(l+1)} \leftarrow$ update parameters using learning rate γ and gradients $\boldsymbol{g}_\sigma^{(l)}$ (Adam);

  $l \leftarrow l + 1$;

until convergence of parameters $\boldsymbol{\phi}_\sigma^{(l+1)}$ and $\boldsymbol{\theta}_\sigma^{(l+1)}$;

Output:

The values of $\boldsymbol{\phi}_\sigma^{(l+1)}$ and $\boldsymbol{\theta}_\sigma^{(l+1)}$, which are taken as the estimator $\widehat{\boldsymbol{\phi}}_\sigma$ and $\widehat{\boldsymbol{\theta}}_\sigma$, respectively.

one could set the initial values $\boldsymbol{\phi}_\tau^{(0)}$, $\boldsymbol{\theta}_\tau^{(0)}$, and $\boldsymbol{\psi}_\tau^{(0)}$ in Algorithms E1 and E2 to be the corresponding parameter estimators from the nearest quantile level. We call this adapted approach as the "adaptive GRAND" implementation. Second, for learning $Q_{i,j,t}(\tau)$ in Section 4.2, our baseline implementation simplifies the optimization of the node fusion parameter $\boldsymbol{\psi}_\tau$ in (11)

Algorithm E4 The training procedure of $\widehat{\boldsymbol{\phi}}_\mu$ and $\widehat{\boldsymbol{\theta}}_\mu$ by the Adam algorithm.

Input:

The sample: $\{(\boldsymbol{r}_t, \boldsymbol{X}_{t-1})\}$;

The initial value of network parameters: $\boldsymbol{\phi}_\mu^{(0)}$ and $\boldsymbol{\theta}_\mu^{(0)}$;

Hyperparameter: learning rate γ;

1: $l = 0$;

2: **repeat**

3: $(\boldsymbol{r}_l, \boldsymbol{X}_{l-1}) \leftarrow$ draw a random data point from $\{(\boldsymbol{r}_t, \boldsymbol{X}_{t-1})\}$ (A cross-sectional minibatch);

4: $\boldsymbol{g}_\mu^{(l)} \leftarrow \nabla_{\boldsymbol{\phi}_\mu, \boldsymbol{\theta}_\mu} \left[\ell_\mu(\boldsymbol{r}_l, \boldsymbol{X}_{l-1}; \boldsymbol{\phi}_\mu^{(l)}, \boldsymbol{\theta}_\mu^{(l)}) \right]$ (Gradients of minibatch estimator);

5: $\boldsymbol{\phi}_\mu^{(l+1)}, \boldsymbol{\theta}_\mu^{(l+1)} \leftarrow$ update parameters using learning rate γ and gradients $\boldsymbol{g}_\mu^{(l)}$ (Adam);

6: $l \leftarrow l + 1$;

7: **until** convergence of parameters $\boldsymbol{\phi}_\mu^{(l+1)}$ and $\boldsymbol{\theta}_\mu^{(l+1)}$;

Output:

The values of $\boldsymbol{\phi}_\mu^{(l+1)}$ and $\boldsymbol{\theta}_\mu^{(l+1)}$, which are taken as the estimators $\widehat{\boldsymbol{\phi}}_\mu$ and $\widehat{\boldsymbol{\theta}}_\mu$, respectively.

(i.e., Step 6 in Algorithm E2) by fixing the SGA-based quantile model parameters $\boldsymbol{\phi}_\tau$ and $\boldsymbol{\theta}_\tau$ at their preliminary estimators $\widehat{\boldsymbol{\phi}}_\tau$ and $\widehat{\boldsymbol{\theta}}_\tau$.[1] To evaluate the effect of this simplification, we also consider the joint optimization in (10) for forecasting $Q_{i,j,t}(\tau)$, and denote it as the "joint GRAND" implementation.

Compared to our used vanilla GRAND implementation, the adaptive GRAND implementation reduces the total training time (including quantile model training, variance model training, and covariance matrix filling) by approximately 5%, while the joint GRAND implementation increases the total training time by approximately 12%. Table E1 reports the out-of-sample tangency portfolio performance from three different implementations. The GRAND portfolio has the best values of AV, MDD, CR, and SR. The adaptive GRAND portfolio has a slightly lower SD and a higher WR, but its AV, CR, and SR are lower and its TO is higher. The joint GRAND portfolio has the lowest TO and delivers results close to the baseline GRAND implementation, but it does not improve the main performance measures despite its higher computational cost. These results above suggest that the baseline implementation provides a

[1] The preliminary estimators $\widehat{\boldsymbol{\phi}}_\tau$ and $\widehat{\boldsymbol{\theta}}_\tau$ are obtained by solving (6).

reasonable balance between computational burden and empirical performance. Therefore, we use the vanilla GRAND implementation in our empirical study.

TABLE E1
Out-of-sample tangency portfolio performances across different implementations.

	AV(%)	SD(%)	WR(%)	MDD(%)	TO(%)	CR	SR
GRAND	**35.87**	22.71	57.07	**19.92**	36.14	**1.80**	**1.48**
Adaptive GRAND	34.62	**22.53**	**59.24**	20.96	39.85	1.65	1.43
Joint GRAND	35.52	22.68	57.07	20.06	**35.49**	1.77	1.46

Note: The backtesting setting is the same as in Table 2. SR: annualized Sharpe ratio; AV: annualized average return; SD: annualized standard deviation of return; WR: winning rate; MDD: maximum drawdown; TO: turnover; CR: annualized Calmar ratio.

APPENDIX F: IMPLEMENTATION DETAILS OF THE GRAND METHOD

F.1. Training, Validation, and Testing. As most deep neural networks, our SGA network has the risk of overfitting. To alleviate this risk, we apply the standard approach to chronologically partition the full data into three disjoint parts: training sample, validation sample, and testing sample. The training and validation samples are taken to estimate network parameters and tune hyperparameters, respectively, while the testing sample is used to evaluate the truly out-of-sample performance. In this paper, we follow the implementations of these three samples in Yang et al. (2024), where the related details are omitted for saving the space.

F.2. Screening on Learned Conditional Quantiles. For the computation of learned conditional quantiles in Sections 4.1 and 4.2, we take the quantile levels $\tau_l = l/(L+1)$, $l = 1, \ldots, L$, for simplicity. In this large pool of learned conditional quantiles, some of them may be invalid and thus are better screened out before calculating the QCM estimators. As in Zhu, Zhang and Zhu (2024), we adopt the unconditional coverage test LR_{uc} in Kupiec (1995) and conditional coverage test LR_{cc} in Christoffersen (1998) to achieve this goal. Specifically,

we compute $\widehat{Q}_{i,t}(\tau_l)$ (i.e., the i-th entry of $\widehat{\boldsymbol{Q}}_t(\tau_l)$) on the training and validation samples, and apply LR_{uc} and LR_{cc} to detect whether the sequence of learned conditional quantiles $\mathcal{Q}_i(\tau_l) \equiv \{\widehat{Q}_{i,t}(\tau_l) : t \in \text{training and validation samples}\}$ is valid at the level α for each stock i and quantile level τ_l. Then, we build a valid quantile level set for stock i:

$$\Omega_i = \{\tau_l : \text{ the validity of } \mathcal{Q}_i(\tau_l) \text{ is accepted by both } LR_{uc} \text{ and } LR_{cc} \text{ at the level } \alpha\}.$$

That is, Ω_i groups all of those quantile levels τ_l, for which the sequence $\mathcal{Q}_i(\tau_l)$ is valid. Clearly, Ω_i depends on α and L jointly, and its size (denoted by $|\Omega_i|$) increases with the value of α (or L). As demonstrated in Zhu, Zhang and Zhu (2024), the performance of the QCM method is robust to the choice of L and α, so we follow their recommendation to take $L = 99$ and $\alpha = 0.01$ in the GRAND method.

After obtaining Ω_i, we use the predicted values $\{\widehat{Q}_{i,t}(\tau_l) : \tau_l \in \Omega_i\}$ to compute $\widehat{\sigma}^2_{i,t}$ for $t \in \{\text{the testing sample}\}$. As L is essentially replaced by $|\Omega_i|$ from the above manipulation, we need a large value of $|\Omega_i|$ to ensure the consistency of $\widehat{\sigma}^2_{i,t}$. This motivates us to discard those stocks with the value of $|\Omega_i|$ less than a predetermined tolerance L_0 (e.g., $L_0 = 20$).

Remark F.1. *The screening step also helps mitigate quantile crossing. Because the SGA-based quantile models are estimated separately at different quantile levels, monotonicity across quantiles is not mechanically imposed. In our empirical application, we observe at least one local crossing for each stock during the out-of-sample period. This is partly because with $L = 99$, our adjacent quantile levels differ by only 0.01, making small local violations more likely.*

After screening, severe crossing is not observed among valid quantile levels in Ω_i that are sufficiently separated. Specifically, in the out-of-sample period, there is no case in which

$$\widehat{Q}_{i,t}(\tau_1) > \widehat{Q}_{i,t}(\tau_2) \quad \textit{for valid } \tau_1 < \tau_2 \textit{ with } \tau_2 - \tau_1 \geq 0.1.$$

Thus, the remaining crossings are mild and local. Since the QCM regression uses the screened

quantile predictions across quantile levels, these local crossings mainly enter as quantile estimation noise rather than as a systematic distortion of the subsequent moment estimators.

Notably, the above screening procedure can also apply to the pool of $\widehat{Q}_{k,t}^{(i,j)}(\tau_l)$ (i.e., the k-th entry of $\widehat{\boldsymbol{Q}}_t^{(i,j)}(\tau_l)$). However, since there are $N(N-1)/2$ different pairs of (i,j) in total, the related screening procedure has a heavy computational burden when N is large (e.g., $N = 1950$ in our empirical analysis), and is thus not considered.

APPENDIX G: DETAILS OF THE GRAND$_1$ METHOD

The GRAND$_1$ method learns the conditional mean and quantile of stock returns through some linear models. Let $\tau \in (0,1)$ be the quantile level. For $Q_{i,t}(\tau)$, the GRAND$_1$ method assumes that for fixed i,

$$Q_{i,t}(\tau) = \alpha_\tau + \boldsymbol{\beta}_\tau' \boldsymbol{x}_{i,t-1} + \frac{\gamma_\tau}{n_{i,t,\tau}} \sum_{j \neq i} \overline{A}_{t,\tau}^{(i,j)} r_{j,t-1}, \tag{G1}$$

where $\alpha_\tau \in \mathcal{R}$, $\boldsymbol{\beta}_\tau \in \mathcal{R}^P$, and $\gamma_\tau \in \mathcal{R}$ are quantile regression coefficients, $\boldsymbol{x}_{i,t-1} \in \mathcal{R}^P$ is the last column of $\boldsymbol{X}_{i,t-1}$, $n_{i,t,\tau} = \sum_{j \neq i} \overline{A}_{t,\tau}^{(i,j)}$, and $\overline{A}_{t,\tau}^{(i,j)} = 1$ if stocks i and j share any linkage in $\widehat{\mathcal{G}}_{t,\tau}$ (defined in (7)), and $\overline{A}_{t,\tau}^{(i,j,k)} = 0$ otherwise. Compared with the linear quantile model in Zhu et al. (2019), model (G1) is more general by capturing the time-varying relationships of stocks.

Next, for $Q_{i,j,t}(\tau)$, the GRAND$_1$ method assumes that for fixed i and j,

$$Q_{i,j,t}(\tau) = \alpha_\tau^* + \boldsymbol{\beta}_\tau^{*\prime} \boldsymbol{x}_{i,j,t-1} + \frac{\gamma_\tau^*}{n_{i,j,t,\tau}} \sum_{k \neq i, k \neq j} \overline{A}_{t,\tau}^{(i,j,k)} r_{j,t-1}, \tag{G2}$$

where $\alpha_\tau^* \in \mathcal{R}$, $\boldsymbol{\beta}_\tau^* \in \mathcal{R}^P$, and $\gamma_\tau^* \in \mathcal{R}$ are quantile regression coefficients, $\boldsymbol{x}_{i,j,t-1}$ is the last column of $\boldsymbol{X}_{i,j,t-1}$ in (8) with $\boldsymbol{\psi} = \widehat{\boldsymbol{\psi}}_\tau$, $n_{i,j,t,\tau} = \sum_{k \neq i, k \neq j} \overline{A}_{t,\tau}^{(i,j,k)}$, and $\overline{A}_{t,\tau}^{(i,j,k)} = 1$ if the fused node (built up stocks i and j) connects another stock k with any relation in $\widehat{\mathcal{G}}_{t,\tau}^{(i,j)}$ (defined in

(12)), and $\overline{A}_{t,\tau}^{(i,j,k)} = 0$ otherwise. For models (G1)–(G2), they are estimated by minimizing the quantile loss from the combination of training and validation samples.

Finally, for fixed i, the GRAND$_1$ method learns the conditional mean $\mu_{i,t}$ by assuming

$$\text{(G3)} \qquad r_{i,t} = \alpha_\mu + \boldsymbol{\beta}_\mu' \boldsymbol{x}_{i,t-1} + \frac{\gamma_\mu}{n_{i,t,\mu}} \sum_{j \neq i} \overline{A}_{t,\mu}^{(i,j)} r_{j,t-1} + \varepsilon_{i,t,\mu},$$

where $\alpha_\mu \in \mathcal{R}$, $\boldsymbol{\beta}_\mu \in \mathcal{R}^P$, and $\gamma_\mu \in \mathcal{R}$ are regression coefficients, $\varepsilon_{i,t,\mu}$ is a zero-mean error term, $n_{i,t,\mu} = \sum_{i \neq j} A_{t,\mu}^{(i,j)}$, and $A_{t,\mu}^{(i,j)} = 1$ if stock i has a linkage stock j in graph $\widehat{\mathcal{G}}_{t,\mu}$ with any relation, and $A_{t,\mu}^{(i,j)} = 0$ otherwise. Note that model (G3) is more general than the network autoregressive model (Zhu et al., 2017) by capturing the time-varying relationships of stocks. For model (G3), it is estimated by minimizing the mean squared loss from the combination of training and validation samples.

APPENDIX H: LONG-SHORT MARKOWITZ PORTFOLIOS

In addition to the long-only Markowitz portfolios studied in Section 5.2, we further investigate the GRAND method from the perspective of constructing long-short Markowitz portfolios. Although short-selling is prohibited in the Chinese stock market, we relax this constraint and estimate the efficient frontier for long-short Markowitz portfolios. Specifically, the long-short Markowitz portfolios are obtained by solving the following constrained optimization problem:

$$\text{(H4)} \qquad \begin{aligned} \boldsymbol{\omega}_t^* &= \underset{\boldsymbol{\omega}_t}{\operatorname{argmin}}\ \boldsymbol{\omega}_t' \boldsymbol{\Sigma}_t \boldsymbol{\omega}_t, \\ &\text{subject to } \boldsymbol{\omega}_t' \mathbf{1} = 1,\ \ \boldsymbol{\omega}_t' \boldsymbol{\mu}_t = r_t^*,\ \text{and } \underline{\omega} \leq \omega_{i,t} \leq \overline{\omega} \ \text{ for } \ i = 1, \ldots, N, \end{aligned}$$

where $\underline{\omega}$ and $\overline{\omega}$ are minimum and maximum weights for each stock. Notably, when $\underline{\omega} = 0$, the optimization problem (H4) reduces to the long-only case in (16). When $\underline{\omega} < 0$, the short-sale constraint is relaxed, allowing $\sum_{i=1}^N |\omega_i| \geq 1$. Particularly, we set $\underline{\omega} = -5\%$ and $\overline{\omega} = 5\%$ to mitigate the risk of over-concentration in portfolio weights, and adopt the same approach as in Section 5.2 to select $\{r_t^*\}$ in (H4) for obtaining tangency portfolios.

Table H2 reports the out-of-sample performance for all considered tangency portfolios without the short-sell constraint. From this table, we can draw the following conclusions:

(i) In the long-short scenario, the GRAND portfolio still achieves the highest value of SR. This is consistent with the finding of Table 2 for the long-only scenario. In particular, the GRAND portfolio outperforms the DCC-NL portfolio (the best performer among shrinkage methods) with 178.7% higher value of SR, significantly exceeds the SR value of the FF5 portfolio (the leading factor model) by nearly 100%, and largely increases the value of SR by 191.1% compared to the FF5-DCC-NL portfolio (the top performer in the factor+shrinkage category).

(ii) Among the three graph-based tangency portfolios, the GRAND portfolio produces the best value of SR, which is 12.0% and 20.2% higher than those of the GRAND_1 and GRAND_2 portfolios, respectively. This finding is consistent with the findings in Table 2. It indicates that the SGA-based models, which capture nonlinear effects, can outperform linear models in learning both the conditional mean and quantiles of returns.

Overall, the comparisons in the long-short Markowitz portfolio selection further highlight that the GRAND method has robust performance in the short-sell-allowed context, underscoring its generalizability for portfolio selections.

APPENDIX I: GLOBAL MINIMUM VARIANCE PORTFOLIO

This appendix assesses the out-of-sample performance of global minimum variance (GMV) portfolios proposed by the GRAND method. Since the construction of GMV portfolios only requires the learning of CCM, the performance of GMV portfolios can reveal whether the CCM is accurately estimated by the GRAND method.

TABLE H2
Out-of-sample tangency portfolio performance with the minimum portfolio weight $\underline{\omega} = -5\%$ and maximum portfolio weight $\overline{\omega} = 5\%$.

	AV(%)	SD(%)	WR(%)	MDD(%)	TO(%)	CR	SR
Graph-based Method							
GRAND	**32.12**	22.74	**58.15**	**21.35**	80.09	**1.50**	**1.31**
GRAND_1	28.35	22.29	56.52	21.68	72.33	1.31	1.17
GRAND_2	26.52	22.22	57.61	23.58	59.84	1.12	1.09
Shrinkage Method							
DCC-NL	12.62	22.10	57.61	26.99	**21.08**	0.47	0.47
BDF	6.24	**22.01**	54.35	34.36	21.73	0.18	0.18
Factor Method							
FF3	10.22	31.12	56.52	38.52	53.45	0.13	0.25
FF5	13.27	31.52	52.72	29.98	56.29	0.16	0.35
FF6	12.83	32.17	51.63	36.78	57.64	0.15	0.33
CH4	2.74	31.93	54.89	45.14	55.52	0.03	0.01
HXZ4	4.73	32.64	54.35	48.82	53.71	0.06	0.07
PCA	-2.75	24.97	49.46	49.35	36.30	-0.06	-0.20
Factor+Shrinkage Method							
FF3-DCC-NL	9.95	30.84	56.52	38.24	53.42	0.13	0.25
FF5-DCC-NL	16.36	31.50	53.80	27.43	56.39	0.20	0.45
FF6-DCC-NL	10.21	32.29	51.63	38.74	57.59	0.26	0.24
CH4-DCC-NL	5.33	31.94	54.89	40.40	55.38	0.13	0.09
HXZ4-DCC-NL	9.92	32.75	54.89	44.09	53.71	0.23	0.23
PCA-DCC-NL	-7.45	25.15	48.91	53.80	40.07	-0.14	-0.39

Note: For each evaluation metric, the best result is highlighted in boldface. SR: annualized Sharpe ratio; AV: annualized average return; SD: annualized standard deviation of return; WR: winning rate; MDD: maximum drawdown; TO: turnover; CR: annualized Calmar ratio.

Specifically, the GMV portfolio selection problem at time t is formulated by solving

$$\begin{aligned} \boldsymbol{\omega}_t^G = &\underset{\boldsymbol{\omega}_t}{\operatorname{argmin}} \ \boldsymbol{\omega}_t' \boldsymbol{\Sigma}_t \boldsymbol{\omega}_t \\ &\text{subject to } \boldsymbol{\omega}_t' \mathbf{1} = 1, \end{aligned} \tag{I1}$$

where $\boldsymbol{\omega}_t \in \mathcal{R}^N$ is the vector of portfolio weights at time t, and $\mathbf{1} \in \mathcal{R}^N$ is the vector of ones. For $\boldsymbol{\omega}_t^G$ in (I1), it has an analytical form:

$$\boldsymbol{\omega}_t^G = \frac{\boldsymbol{\Sigma}_t^{-1}\mathbf{1}}{\mathbf{1}'\boldsymbol{\Sigma}_t^{-1}\mathbf{1}}.$$

Hence, based on the GRAND estimator of $\boldsymbol{\Sigma}_t$, we can directly obtain $\boldsymbol{\omega}_t^G$ to propose the GMV portfolio during the testing sample. Besides the GRAND method, we also construct the GMV portfolios by learning $\boldsymbol{\Sigma}_t$ from those competing methods in Section 5.2: GRAND$_1$, GRAND$_2$, DCC-NL, BDF, FF3, FF5, FF6, CH4, HXZ4, PCA, FF3-DCC-NL, FF5-DCC-NL, FF6-DCC-NL, CH4-DCC-NL, HXZ4-DCC-NL, and PCA-DCC-NL. The inclusion of these competing methods is to examine whether the GRAND method can produce more accurate CCM estimators. It is worth noting that (i) we do not need to pre-learn the dynamics of conditional mean for three GRAND-type methods, whereas the learning of the dynamics of conditional mean is necessary for the remaining 14 methods; (ii) we disregard short-sale and maximum weight constraints and transaction costs to create an idealized investing environment, since our main purpose is to check the accuracy of the CCM estimators of $\boldsymbol{\Sigma}_t$ via different methods.

For all of considered GMV portfolios above, we re-balance them every week, and assess their out-of-sample performance via three conventional metrics: annualized AV, annualized SD, and IR, where IR=AV/SD. Among these three metrics, SD is key to showing the learning accuracy of $\boldsymbol{\Sigma}_t$, and a lower value of SD indicates a better estimation method of $\boldsymbol{\Sigma}_t$.

Table I1 reports the out-of-sample performance of all considered GMV portfolios. From this table, we can have the following findings:

(i) The GRAND portfolio has the smallest value of SD. Specifically, the SD of the GRAND portfolio is 24.9% and 31% lower than that of the GRAND$_1$ and GRAND$_2$ portfolios, respectively. Compared with the FF5-DCC-NL portfolio (i.e., the portfolio with the lowest value of SD among all of shrinkage, factor, and factor+shrinkage methods), the GRAND portfolio performs much better with 43.8% lower value of SD.

(ii) Although the GRAND portfolio does not have the highest value of AV, it can still produce the highest value of IR, due to its dominating performance in terms of SD.

Overall, the above empirical results clearly demonstrate that the GRAND method can deliver accurate vast CCM estimators, leading to good performance in GMV portfolio selection.

TABLE I1
Out-of-sample GMV portfolio performance across different methods.

	AV(%)	SD(%)	IR
Graph-based Method			
GRAND	14.42	**10.06**	**1.43**
$GRAND_1$	12.97	13.18	0.98
$GRAND_2$	14.04	12.56	1.12
Shrinkage Method			
DCC-NL	21.16	16.82	1.27
BDF	-14.43	19.76	-0.73
Factor Method			
FF3	18.46	19.00	0.97
FF5	**21.47**	17.50	1.23
FF6	3.23	17.76	0.18
CH4	5.57	18.25	0.31
HXZ4	10.60	17.89	0.59
PCA	18.38	18.88	0.97
Factor + Shrinkage Method			
FF3-DCC-NL	11.88	17.49	0.68
FF5-DCC-NL	12.30	14.47	0.85
FF6-DCC-NL	1.71	16.15	0.11
CH4-DCC-NL	10.04	19.52	0.51
HXZ4-DCC-NL	3.74	19.43	0.19
PCA-DCC-NL	13.66	18.57	0.74

Note: For each evaluation metric, the best result is highlighted in boldface. All of portfolios are constructed without short-sale and maximum weight constraints and transaction costs. AV: annualized average return; SD: annualized standard deviation of return; IR: annualized information ratio.

APPENDIX J: PAIRS TRADING

As a well-known market neutral strategy, pairs trading strategy involves two key steps. The first step is to identify pairs of stocks that exhibit high correlations, meaning that the price of one moves in the same direction as the other. The second step is to detect divergences in prices between the paired stocks. When a divergence is observed, investors can take opposite positions in the stocks of the pair by going long on the underperforming stock and short on the overperforming one, anticipating that the price divergence will diminish to make profits.

To measure the correlations between stock returns, Chen et al. (2019) employ sample correlation matrices, where each correlation matrix is estimated using a 4-year rolling window of historical return data. Then, for each stock i, the top 50 stocks with the highest correlations are chosen to form its pairs (denoted as a set $\mathcal{P}_{i,t}$), and the pairs' portfolio excess return $\text{Cret}_{i,t}$ is defined as

$$\text{Cret}_{i,t} = \frac{1}{|\mathcal{P}_{i,t}|} \sum_{j \in \mathcal{P}_{i,t}} r_{j,t} - r_f,$$

where r_f is the risk-free rate. The divergence $\text{RetDiff}_{i,t}$ of stock i at time point t is defined as

$$\text{RetDiff}_{i,t} = \beta_{i,t}\text{Cret}_{i,t} - (r_{i,t} - r_f),$$

where $\beta_{i,t}$ is the regression coefficient that links portfolio excess return with idiosyncratic excess return, and is estimated using the four-year historical window for each stock i. Clearly, the coefficient $\beta_{i,t}$ accounts for differences in return variation between the stock and its pairs. Based on $\text{RetDiff}_{i,t}$, stocks are sorted into ten deciles. Traders can construct either an equal-weighted long-short portfolio (long the decile 10 and short the decile 1) or a long-only portfolio (long the decile 10 only).

Instead of using sample correlations, we can also learn correlations from the CCM estimators by adopting the GRAND method and other methods outlined in Section 5.2, leading to the

so-called CCM-based pairs trading portfolios. In addition, the group structure identified by the GRAND method provides another approach to select pairs based on the overall adjacency matrices in (4). Specifically, for stock i, we can define its set of pairs as $\mathcal{P}_{i,t} = \{j : \overline{\boldsymbol{A}}_t^{(i,j)} \neq 0\}$, indicating that each stock j in $\mathcal{P}_{i,t}$ has a nonzero connection to stock i. Then, by considering four types of group structures based on variance, mean, median, and tail forecasts, we can get four different adjacency matrices: $\overline{\boldsymbol{A}}_{t,\sigma}$, $\overline{\boldsymbol{A}}_{t,\mu}$, $\overline{\boldsymbol{A}}_{t,\text{VaR}(0.5)}$, and $\overline{\boldsymbol{A}}_{t,\text{VaR}(0.05)}$, which give us four types of graph-based pairs trading portfolios.

Table J1 reports the performance of long-short pairs trading portfolios across different methods, where each portfolio is rebalanced weekly with zero transaction costs, and it is evaluated out-of-sample over the period from January 1, 2019 to July 31, 2022. From this table, we can have the following key observations:

(i) The GRAND pairs trading portfolio achieves the highest values of AV, CR, and SR among all considered pairs trading portfolios. This highlights the superiority of the GRAND method over the other methods in learning high-dimensional CCM.

(ii) Regarding graph-based pairs trading portfolios, the mean-type portfolio ranks second in terms of AV, CR, and SR, while also exhibiting the lowest value of MDD; in contrast, the variance-type pairs trading portfolio offers the lowest value of SD, making it appealing for risk-averse investors. All of these results emphasize the effectiveness of the SGA model in capturing meaningful relationships between stocks.

(iii) While the sample CCM-based pairs trading portfolio (i.e., the portfolio proposed by Chen et al. (2019)) performs competitively, it ranks third in terms of SR among all considered pairs trading portfolios, with a significant gap to the top two.

As a complement, we also report the out-of-sample performance of long-only pairs trading portfolios in Table J2, which shows that the mean-type portfolio attains the highest values of AV, CR, and SR. This finding further supports the potential of using the SGA model to

TABLE J1
Out-of-sample performances of long-short pairs trading portfolios across different methods.

	AV(%)	SD(%)	WR(%)	MDD(%)	TO(%)	CR	SR
CCM-based Portfolio							
GRAND	**15.70**	17.19	57.69	14.26	338.08	**1.10**	**0.78**
DCC-NL	9.86	14.69	58.24	13.43	335.25	0.73	0.51
BDF	3.94	16.35	52.75	21.96	335.45	0.18	0.10
FF3	6.93	15.16	54.95	16.75	336.24	0.41	0.31
FF5	6.29	15.17	55.49	19.11	336.55	0.33	0.26
FF6	6.05	15.28	54.95	19.43	336.37	0.31	0.25
CH4	6.69	15.24	54.40	18.06	336.49	0.37	0.29
HXZ4	6.48	15.07	54.95	18.17	336.28	0.36	0.28
PCA	7.41	14.62	54.95	14.75	336.12	0.50	0.35
FF3-DCC-NL	8.64	15.06	57.69	17.45	335.27	0.50	0.42
FF5-DCC-NL	6.80	15.24	54.40	20.75	**334.78**	0.33	0.30
FF6-DCC-NL	9.83	15.07	56.04	15.10	335.82	0.65	0.50
CH4-DCC-NL	8.14	15.06	56.04	18.90	335.41	0.43	0.39
HXZ4-DCC-NL	9.62	15.34	**58.79**	16.93	335.59	0.57	0.48
PCA-DCC-NL	6.81	15.84	52.75	21.07	336.12	0.32	0.28
Sample	10.75	15.95	56.59	16.04	336.01	0.67	0.53
Graph-based Portfolio							
Variance-type	8.17	**13.69**	56.59	15.47	337.96	0.53	0.43
Mean-type	11.95	14.56	54.95	**12.11**	336.49	0.99	0.66
Median-type	7.65	15.28	52.75	17.12	337.55	0.45	0.35
Tail-type	10.24	15.99	54.95	14.45	336.29	0.71	0.50

Note: For each evaluation metric, the best result is highlighted in boldface. SR: annualized Sharpe ratio; AV: annualized average return; SD: annualized standard deviation of return; WR: winning rate; MDD: maximum drawdown; TO: turnover; CR: annualized Calmar ratio.

uncover meaningful graph structures among stocks.

APPENDIX K: ADDITIONAL CAUSAL GRAPH ANALYSIS

K.1. Sector-level Connectedness Analysis. We investigate the sector-level connectedness from two perspectives: sector-to-system connectedness and sector-to-sector connectedness.

TABLE J2
Out-of-sample performances of long-only pairs trading portfolios across different methods.

	AV(%)	SD(%)	WR(%)	MDD(%)	TO(%)	CR	SR
CCM-based Portfolio							
GRAND	16.08	24.31	51.10	25.96	172.34	0.62	0.57
DCC-NL	16.28	23.86	52.75	26.96	172.41	0.60	0.59
BDF	11.14	24.03	50.00	27.59	**170.93**	0.40	0.37
FF3	14.34	23.93	53.30	26.77	171.73	0.54	0.50
FF5	13.88	24.00	52.75	26.73	171.85	0.52	0.48
FF6	13.84	24.01	51.65	26.89	171.88	0.51	0.48
CH4	14.11	24.05	53.30	26.86	171.83	0.53	0.49
HXZ4	14.40	23.98	53.30	26.73	171.80	0.54	0.50
PCA	14.62	23.77	54.95	26.62	171.53	0.55	0.52
FF3-DCC-NL	14.48	23.86	53.30	26.51	171.56	0.55	0.51
FF5-DCC-NL	13.57	24.02	51.65	26.57	171.55	0.51	0.47
FF6-DCC-NL	15.43	24.12	**56.04**	26.62	171.77	0.58	0.54
CH4-DCC-NL	14.94	23.91	53.30	25.91	171.51	0.58	0.53
HXZ4-DCC-NL	15.72	24.10	51.65	26.44	172.01	0.59	0.56
PCA-DCC-NL	13.49	24.11	51.65	26.36	171.88	0.51	0.46
Sample	15.98	24.12	54.40	25.92	171.47	0.62	0.57
Graph-based Portfolio							
Variance-type	15.52	**23.75**	51.65	**25.71**	172.04	0.60	0.56
Mean-type	**16.47**	23.84	55.49	25.80	171.88	**0.64**	**0.59**
Median-type	13.48	23.76	52.20	27.95	171.39	0.48	0.47
Tail-type	15.24	24.08	52.20	25.78	171.47	0.59	0.54

Note: For each evaluation metric, the best result is highlighted in boldface. SR: annualized Sharpe ratio; AV: annualized average return; SD: annualized standard deviation of return; WR: winning rate; MDD: maximum drawdown; TO: turnover; CR: annualized Calmar ratio.

First, we examine the dynamic importance of a certain sector to the financial market, using two measures of the sector-to-system connectedness:

$$(\mathcal{S}_\alpha \to \mathcal{S})_{t,\sigma} = \frac{1}{\#\mathcal{S}_\alpha(N - \#\mathcal{S}_\alpha)} \sum_{i \in \mathcal{S}_\alpha} \sum_{j \notin \mathcal{S}_\alpha} I(\overline{A}_{t,\sigma}^{(i,j)} \neq 0),$$

$$(\mathcal{S}_\alpha \to \mathcal{S})_{t,\mathrm{VaR}} = \frac{1}{\#\mathcal{S}_\alpha(N - \#\mathcal{S}_\alpha)} \sum_{i\in\mathcal{S}_\alpha} \sum_{j\notin\mathcal{S}_\alpha} I(\overline{A}^{(i,j)}_{t,\mathrm{VaR}(0.05)} \neq 0),$$

where $\mathcal{S}_\alpha$ is the set of stocks in Sector α (e.g., α is Energy or Financials), and $\#\mathcal{S}_\alpha$ is the number of stocks in $\mathcal{S}_\alpha$.

Table K3 reports the descriptive statistics of two measures of sector-to-system connectedness for 11 sectors during the out-of-sample period. Several conclusions can be drawn from this table. First, the results related to $(\mathcal{S}_\alpha \to \mathcal{S})_{t,\mathrm{VaR}}$ are significantly smaller than those related to $(\mathcal{S}_\alpha \to \mathcal{S})_{t,\sigma}$. Notably, the mean value of $(\mathcal{S}_\alpha \to \mathcal{S})_{t,\mathrm{VaR}}$ shows that the Real Estate sector has a substantial impact on other sectors, while this phenomenon cannot be detected by that of $(\mathcal{S}_\alpha \to \mathcal{S})_{t,\sigma}$. It indicates that tail risk in the Real Estate sector could spread out to other sectors more significantly than regular risk, and consequently, it has the potential to induce large systemic risk. Second, the results of both measures reveal that the Energy, Financials, and Utilities sectors have a relatively large influence on other sectors. Thus, their contributions to systemic risk should be monitored prudentially. Third, sectors associated with primary industries, Information Technology, Consumer Staples, Industrials, and Materials exhibit small impact from the perspective of both variance and VaR. This finding aligns with the results reported in Barigozzi and Hallin (2017).

Next, we study the dynamic sector-to-sector connectedness by defining two measures:

$$(\mathcal{S}_\alpha \to \mathcal{S}_\beta)_{t,\sigma} = \frac{1}{\#\mathcal{S}_\alpha \#\mathcal{S}_\beta} \sum_{i\in\mathcal{S}_\alpha} \sum_{j\in\mathcal{S}_\beta} I(\overline{A}^{(i,j)}_{t,\sigma} \neq 0),$$

$$(\mathcal{S}_\alpha \to \mathcal{S}_\beta)_{t,\mathrm{VaR}} = \frac{1}{\#\mathcal{S}_\alpha \#\mathcal{S}_\beta} \sum_{i\in\mathcal{S}_\alpha} \sum_{j\in\mathcal{S}_\beta} I(\overline{A}^{(i,j)}_{t,\mathrm{VaR}(0.05)} \neq 0).$$

Fig K1 displays the temporal averages of $(\mathcal{S}_\alpha \to \mathcal{S}_\beta)_{t,\sigma}$ and $(\mathcal{S}_\alpha \to \mathcal{S}_\beta)_{t,\mathrm{VaR}}$, when α is each of the top three influential sectors, as shown in Table K3. Here, for better interpretation, we normalize the values of $(\mathcal{S}_\alpha \to \mathcal{S}_\beta)_{t,\sigma}$ and $(\mathcal{S}_\alpha \to \mathcal{S}_\beta)_{t,\mathrm{VaR}}$ at each time t such that $\sum_\beta (\mathcal{S}_\alpha \to \mathcal{S}_\beta)_{t,\sigma} = 1$ and $\sum_\beta (\mathcal{S}_\alpha \to \mathcal{S}_\beta)_{t,\mathrm{VaR}} = 1$ for given Sector α. From Fig K1, we observe that the

TABLE K3
The descriptive statistics for sector-to-system connectedness of 11 sectors.

Sector α	$(\mathcal{S}_\alpha \to \mathcal{S})_{t,\sigma}$				$(\mathcal{S}_\alpha \to \mathcal{S})_{t,\text{VaR}}$			
	Mean	SD	Max	Min	Mean	SD	Max	Min
Energy	**0.003**	0.003	0.010	0.000	**0.226**	0.060	0.330	0.126
Real Estate	0.000	0.000	0.002	0.000	**0.184**	0.024	0.238	0.134
Financial	**0.009**	0.003	0.015	0.003	**0.178**	0.034	0.275	0.123
Utilities	**0.003**	0.002	0.010	0.000	0.143	0.027	0.192	0.086
Health Care	0.001	0.001	0.005	0.000	0.122	0.030	0.163	0.066
Telecommunication Service	0.002	0.002	0.012	0.000	0.065	0.033	0.122	0.011
Consumer Discretionary	0.001	0.001	0.002	0.000	0.041	0.013	0.066	0.018
Industrials	0.000	0.000	0.001	0.000	0.037	0.012	0.065	0.014
Materials	0.000	0.000	0.001	0.000	0.027	0.011	0.049	0.011
Consumer Staples	0.001	0.001	0.005	0.000	0.010	0.004	0.021	0.002
Information Technology	0.001	0.001	0.003	0.000	0.002	0.001	0.005	0.000

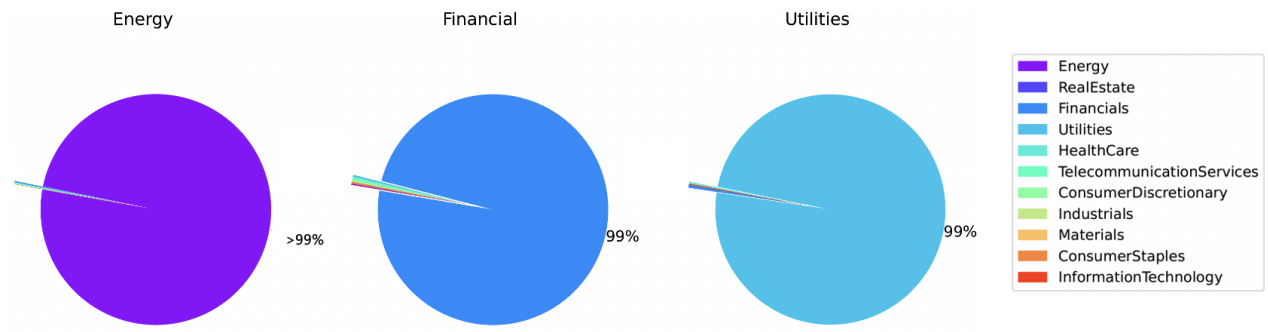
Note: For each measure, the top three influential sectors (with respect to the mean values) are in boldface.

number of connections between stocks within the same sector is much larger than that between stocks from two different sectors. This finding is more evident from the viewpoint of VaR (see the visual evidence in Fig K2).

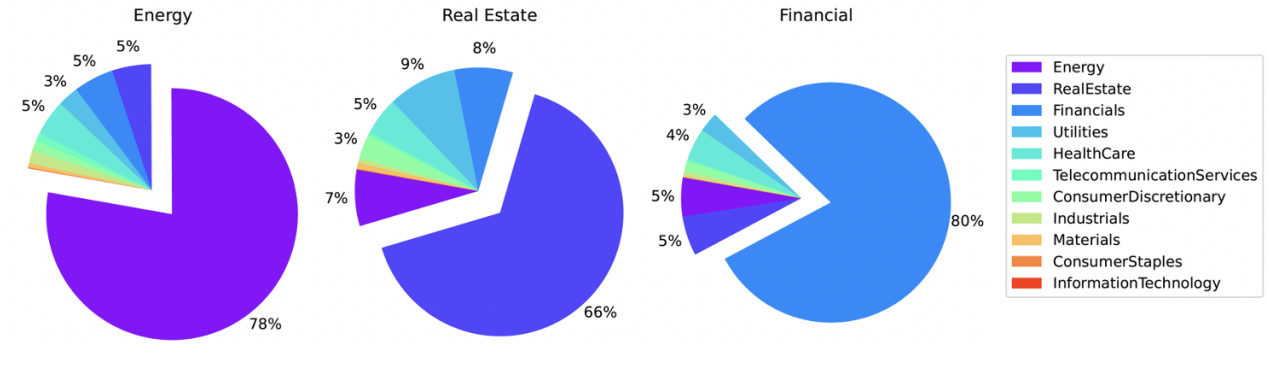
K.2. Stock-level Connectedness Analysis. We analyze the stock-level connectedness of all considered stocks in Section 5 through two measures:

$$n_{i,t,\sigma} = \frac{1}{N-1}\sum_{j\neq i} I(\overline{A}^{(i,j)}_{t,\sigma} \neq 0) \text{ and } n_{i,t,\text{VaR}} = \frac{1}{N-1}\sum_{j\neq i} I(\overline{A}^{(i,j)}_{t,\text{VaR}(0.05)} \neq 0),$$

for each given stock i. To explore the importance of each stock i over a long period, we calculate the temporal averages of $n_{i,t,\sigma}$ and $n_{i,t,\text{VaR}}$ (denoted by $\overline{n}_{i,\sigma}$ and $\overline{n}_{i,\text{VaR}}$) over the out-of-sample period. Tables K1–K2 report the top 20 most influential and least influential stocks in terms of these averages, respectively. Interestingly, we find that only three stocks appear in both lists of the most influential stocks made by $\overline{n}_{i,\sigma}$ and $\overline{n}_{i,\text{VaR}}$ in Table K1, whereas the lists of



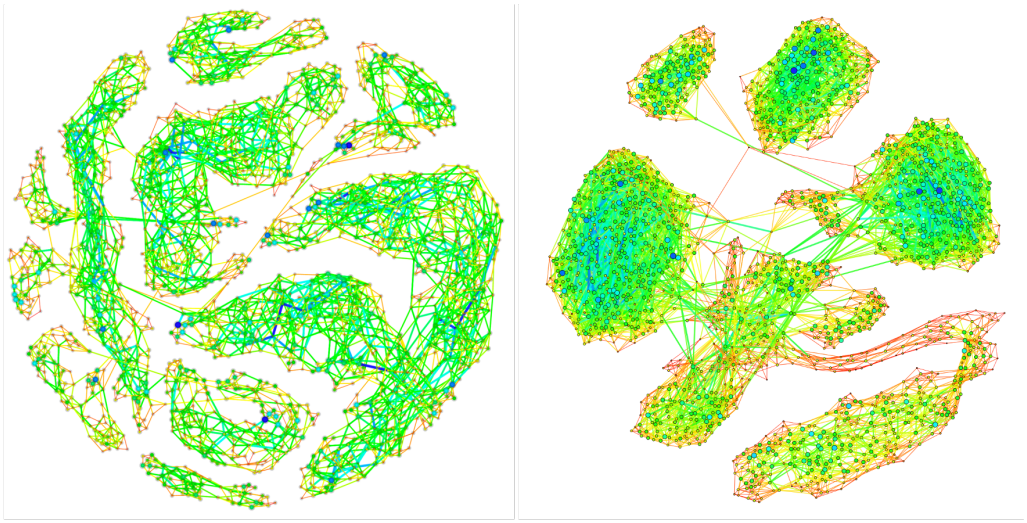
(a) $(\mathcal{S}_\alpha \to \mathcal{S}_\beta)_{t,\sigma}$



(b) $(\mathcal{S}_\alpha \to \mathcal{S}_\beta)_{t,\mathrm{VaR}}$

FIG K1. *(Color online) For each influential Sector α, the temporal averages of $(\mathcal{S}_\alpha \to \mathcal{S}_\beta)_{t,\sigma}$ (or $(\mathcal{S}_\alpha \to \mathcal{S}_\beta)_{t,\mathrm{VaR}}$) across Sector β are displayed in top (or bottom) panels. For ease of presentation, those values less than 2% are unlabeled in each panel.*

least influential stocks made by $\overline{n}_{i,\sigma}$ and $\overline{n}_{i,\mathrm{VaR}}$ share seven firms in Table K2. This finding implies that the top important stocks differ significantly in terms of VaR and variance, while the unimportant stocks from the perspective of VaR and variance are largely the same.



(a) (b)

FIG K2. *(Color online) (a) Visualization of the overall adjacency matrix* $\overline{A}_{t,\sigma}$ *on the first trading week of 2019. If there is an edge between two nodes in the figure, it means* $\overline{A}_{t,\sigma}^{(i,j)} = 1$*, and* $\overline{A}_{t,\sigma}^{(i,j)} = 0$ *otherwise. Moreover, the size of each node is proportional to the number of connected nodes, while the edge color captures the degree of connected nodes. (b) Visualization of the overall adjacency matrix* $\overline{A}_{t,\mathrm{VaR}(0.05)}$ *on the first trading week of 2019, where the presentations of edge and node are similar as those in (a).*

APPENDIX L: MONTE CARLO SIMULATION

To demonstrate the finite-sample performance of our GRAND method, we simulate the asset return data from the following model:

$$r_{i,t} = g(\boldsymbol{z}_{i,t-1}) + \epsilon_{i,t} \text{ for } i = 1, \ldots, N \text{ and } t = 1, \ldots, T, \tag{L1}$$

where $\boldsymbol{z}_{i,t} = (r_{i,t}, z_{i,t,1}, z_{i,t,2}, z_{i,t,3})' \in \mathcal{R}^4$ denotes a vector of asset characteristics, $g(\cdot) : \mathcal{R}^4 \to \mathcal{R}$ is a nonlinear function, and $\epsilon_{i,t}$ is the idiosyncratic error. Here, we take $N = 2000$ and $T = 600$, to align with the choices of N and T in our empirical analysis.

Below, we first present how we generate $g(\boldsymbol{z}_{i,t-1})$ and $\epsilon_{i,t}$ in (L1). To be specific, we simulate $g(\boldsymbol{z}_{i,t-1})$ according to

$$g(\boldsymbol{z}_{i,t-1}) = 0.1 r_{i,t-1} z_{i,t-1,1} + 0.1 z_{i,t-1,1}^2 + 0.05\cos(z_{i,t-1,2}) + 0.05\mathrm{sgn}(z_{i,t-1,3}),$$

TABLE K1
Top twenty stocks ranked by either $\overline{n}_{i,\sigma}$ or $\overline{n}_{i,\mathrm{VaR}}$.

	$\overline{n}_{i,\sigma}$		$\overline{n}_{i,\mathrm{VaR}}$	
Ranking	Symbol	Sector	Symbol	Sector
#1	002533	Industrials	600150	Industrials
#2	002406	Consumer Discretionary	**300215**	**Industrials**
#3	002360	Materials	300127	Information Technology
#4	002593	Industrials	600072	Industrials
#5	002434	Consumer Discretionary	**002482**	**Industrials**
#6	002540	Materials	002338	Information Technology
#7	002455	Materials	002613	Industrials
#8	002380	Industrials	600039	Industrials
#9	**002482**	**Industrials**	600278	Industrials
#10	002560	Industrials	300172	Industrials
#11	300217	Consumer Discretionary	300057	Materials
#12	300154	Industrials	002519	Information Technology
#13	**002520**	**Industrials**	**002520**	**Industrials**
#14	300141	Industrials	600153	Industrials
#15	002443	Materials	002441	Industrials
#16	300112	Information Technology	002468	Industrials
#17	600099	Consumer Discretionary	300093	Industrials
#18	002595	Industrials	300224	Materials
#19	**300215**	**Industrials**	002150	Materials
#20	300092	Industrials	600133	Industrials

Note: The stocks appear in both lists of most influential stocks made by $\overline{n}_{i,\sigma}$ and $\overline{n}_{i,\mathrm{VaR}}$ are in boldface. The symbol of each stock is given by Shanghai and Shenzhen Stock Exchanges. One can refer to https://www.sse.com.cn/ for more details.

where the characteristics $z_{i,t,k}$ satisfy different autoregressive models:

$$z_{i,t,k} = \rho_k z_{i,t-1,k} + \varepsilon_{i,t,k} \quad \text{for} \quad k = 1, 2, 3,$$

with $\rho_k \overset{i.i.d.}{\sim} U(0.8, 1)$ and $\varepsilon_{i,t,k} \overset{i.i.d.}{\sim} N(0, 1)$. Moreover, we simulate the vector of cross-sectional errors $\boldsymbol{\epsilon}_t \equiv (\epsilon_{1,t}, \ldots, \epsilon_{N,t})'$ as follows:

$$\boldsymbol{\epsilon}_t | \mathcal{F}_{t-1} \sim N(\mathbf{0}, \boldsymbol{\Sigma}_t),$$

TABLE K2
Bottom twenty stocks ranked by either $\overline{n}_{i,\mathrm{VaR}}$ or $\overline{n}_{i,\sigma}$.

	$\overline{n}_{i,\sigma}$		$\overline{n}_{i,\mathrm{VaR}}$	
Ranking	Symbol	Sector	Symbol	Sector
#-1	**601918**	**Energy**	**000607**	**Telecommunication Services**
#-2	600848	Real Estate	**601998**	**Financials**
#-3	**601801**	**Telecommunication Services**	**601988**	**Financials**
#-4	601668	Industrials	000006	Real Estate
#-5	**601939**	**Financials**	601918	Energy
#-6	**601588**	**Real Estate**	601991	Utilities
#-7	000059	Energy	**601939**	**Financials**
#-8	000617	Financials	000026	Consumer Discretionary
#-9	601999	Telecommunication Services	000014	Real Estate
#-10	**601998**	**Financials**	601898	Energy
#-11	000004	Health Care	**000046**	**Financials**
#-12	000027	Utilities	000016	Consumer Discretionary
#-13	601958	Materials	**601588**	**Real Estate**
#-14	600763	Health Care	601818	Financials
#-15	000060	Materials	601888	Consumer Discretionary
#-16	**000607**	**Telecommunication Services**	601328	Financials
#-17	000793	Telecommunication Services	601799	Consumer Discretionary
#-18	600892	Telecommunication Services	601933	Consumer Staples
#-19	601199	Utilities	**601801**	**Telecommunication Services**
#-20	**000046**	**Financials**	000025	Consumer Discretionary

Note: The stocks appear in both lists of least influential stocks made by $\overline{n}_{i,\sigma}$ and $\overline{n}_{i,\mathrm{VaR}}$ are in boldface.

where $\boldsymbol{\Sigma}_t$ is the nearest positive semidefinite approximation to a preliminary covariance matrix $\widetilde{\boldsymbol{\Sigma}}_t$ (with the diagonal elements unchanged). Here, by letting $\mathrm{Diag}(\boldsymbol{a})$ denote an $N \times N$ diagonal matrix with $\boldsymbol{a} \in \mathcal{R}^N$ on its main diagonal, $\widetilde{\boldsymbol{\Sigma}}_t$ evolves dynamically according to

$$\begin{aligned}\widetilde{\boldsymbol{\Sigma}}_t = {} & 0.05\boldsymbol{I}_N + 0.8\mathrm{Diag}(\widetilde{\boldsymbol{\sigma}}_{t-1}^2) + 0.1\mathrm{Diag}(\boldsymbol{W}_t\widetilde{\boldsymbol{\sigma}}_{t-1}^2) + 0.1\mathrm{Diag}(\widetilde{\boldsymbol{\epsilon}}_{t-1}^2) \\ & + \boldsymbol{W}_t \odot (0.2\boldsymbol{\epsilon}_{t-1}\boldsymbol{\epsilon}_{t-1}' + 0.2\widetilde{\boldsymbol{\Sigma}}_{t-1}),\end{aligned} \tag{L2}$$

where $\boldsymbol{I}_N \in \mathcal{R}^{N\times N}$ is the identity matrix, $\boldsymbol{W}_t \in \mathcal{R}^{N\times N}$ is the adjacency matrix specifying

Granger causality, $\widetilde{\boldsymbol{\sigma}}_{t-1}^2 \in \mathcal{R}^N$ is the vector on the main diagonal of matrix $\widetilde{\boldsymbol{\Sigma}}_{t-1}$, and $\widetilde{\boldsymbol{\epsilon}}_{t-1}^2 \in \mathcal{R}^N$ is the vector on the main diagonal of matrix $\boldsymbol{\epsilon}_{t-1}\boldsymbol{\epsilon}_{t-1}'$. Note that the setup in (L2) ensures sparsity in the off-diagonal entries of $\widetilde{\boldsymbol{\Sigma}}_t$, the dynamics of which share a linkage with the network autoregressive models (Zhu et al., 2017, 2019; Zhou et al., 2020) and classical GARCH-type models (Engle, 1982; Engle, Ledoit and Wolf, 2019). For $\boldsymbol{W}_t$ in (L2), we simulate it using a similar approach as in the encoder of the SGA model with $\boldsymbol{h}_{i,t} = \boldsymbol{z}_{i,t}$ and $K^* = 3$:

1. Initialize N central nodes $\boldsymbol{u}_{k,t} = \boldsymbol{z}_{k,t}$ for $k = 1, \ldots, N$;
2. Compute the pairwise cosine similarities: $\gamma_{i,k,t} = \text{Cosine}(\boldsymbol{z}_{i,t}, \boldsymbol{u}_{k,t})$ for $i = 1, \ldots, N$ and $k = 1, \ldots, N$;
3. Apply a threshold to obtain filtered similarities:

$$\gamma_{i,k,t}^* = \begin{cases} 1, & \text{if } k = i, \\ \gamma_{i,k,t}, & \text{if } |\gamma_{i,k,t}| > \text{Top}\big(\boldsymbol{\Gamma}_{i,t}, K^*\big), \\ 0, & \text{if } |\gamma_{i,k,t}| \leq \text{Top}\big(\boldsymbol{\Gamma}_{i,t}, K^*\big), \end{cases}$$

   where $\boldsymbol{\Gamma}_{i,t} = \{|\gamma_{i,k,t}| : k = 1, \ldots, i-1, i+1, \ldots, N\}$, $\text{Top}(\boldsymbol{A}, b)$ denotes the b-th largest value among the elements in set $\boldsymbol{A}$, and K^* is a hyperparameter controlling the number of central nodes that are similar to individuals.
4. Construct the groups $\{\mathcal{M}_{k,t} : k \in \mathcal{K}_t\}$ with membership defined as: $\mathcal{M}_{k,t} = \{i : \gamma_{i,k,t}^* \neq 0\}$, and $\mathcal{K}_t = \{k_1, \ldots, k_{K_t}\}$ is the set of central nodes, where each central node not just connects itself.
5. Construct the unweighted adjacency matrix $\overline{\boldsymbol{W}}_t$ with elements:

$$\overline{W}_t^{(i,j)} = \begin{cases} 1, & \text{if } i \neq j \text{ and } \ i, j \in \mathcal{M}_{k,t} \text{ for some } k \in \mathcal{K}_t \\ 0, & \text{otherwise.} \end{cases}$$

6. Normalize $\boldsymbol{W}_t^*$ to obtain the weighted adjacency matrix $\boldsymbol{W}_t$ with elements:

$$W_t^{(i,j)} = \frac{\overline{W}_t^{(i,j)}}{\sum_{j=1}^{N} \overline{W}_t^{(i,j)}}.$$

Next, for each Monte Carlo sample $\{(r_{i,t}, z_{i,t-1}) : i = 1, \ldots, N; t = 1, \ldots, T\}$ generated from model (L1), we split it into three equal-length consecutive subsamples: training, validation, and testing samples, where each subsample has the sample size $T/3$. Then, we treat $\boldsymbol{z}_{i,t-1}$ as the characteristic vector $\boldsymbol{x}_{i,t-1}$ of individual i at time $t-1$, and stack them to form the cross-sectional input $\boldsymbol{X}_{t-1}$. After forming $\boldsymbol{X}_{t-1}$, we follow the implementations as in Section 2 to compute the out-of-sample CCM estimators $\{\widetilde{\boldsymbol{\Sigma}}_t : t \in \text{testing sample}\}$ via the GRAND method, where the testing sample includes data that never enter into model estimation or tuning.

To evaluate the out-of-sample performance of the GRAND method, we compute the following three metrics:

$$\begin{aligned}
\ell_{\text{total}} &= \frac{3}{T} \sum_{t \in \text{testing}} \|\boldsymbol{\Sigma}_t - \widetilde{\boldsymbol{\Sigma}}_t\|_F, \\
\ell_{\text{diag}} &= \frac{3}{T} \sum_{t \in \text{testing}} \|\text{diag}(\boldsymbol{\Sigma}_t) - \text{diag}(\widetilde{\boldsymbol{\Sigma}}_t)\|_F, \\
\ell_{\text{off-diag}} &= \frac{3}{T} \sum_{t \in \text{testing}} \|\left[\boldsymbol{\Sigma}_t - \text{diag}(\boldsymbol{\Sigma}_t)\right] - \left[\widetilde{\boldsymbol{\Sigma}}_t - \text{diag}(\widetilde{\boldsymbol{\Sigma}}_t)\right]\|_F,
\end{aligned}$$

where $\boldsymbol{\Sigma}_t$ and $\widetilde{\boldsymbol{\Sigma}}_t$ are the true and predicted CCMs, respectively. The metrics ℓ_{total}, ℓ_{diag}, and $\ell_{\text{off-diag}}$ assess the predictive accuracy of the employed method from the perspectives of the total, diagonal, and off-diagonal elements of the CCM, respectively.

Table L3 reports the average values of ℓ_{total}, ℓ_{diag}, and $\ell_{\text{off-diag}}$ for the GRAND method across 100 Monte Carlo replications. As comparison, the corresponding values of ℓ_{total}, ℓ_{diag}, and $\ell_{\text{off-diag}}$ for the GRAND$_1$, DCC-NL (Engle, Ledoit and Wolf, 2019), BDF (Kastner, 2019),

PCA (Bai and Ng, 2002), and PCA-DCC-NL models are also provided[2]. From this table, we can obtain the following findings:

(i) The GRAND method outperforms all competitors with significantly smaller values of ℓ_{total}, ℓ_{diag}, and $\ell_{\text{off-diag}}$, when the simulated CCMs follow a graph-based sparse structure as defined in (L2). Furthermore, the GRAND_1 method ranks second, as it cannot adequately capture the nonlinear effects in quantiles.

(ii) The DCC-NL method ranks third. This can be attributed to the fact that the DCC-NL method could partly account for the autoregressive nature of CCMs, although it fails to capture the sparse structure and nonlinear effects in CCMs.

(iii) The BDF and PCA methods perform unsatisfactorily, as no underlying factor structure is considered in the data generation process.

(iv) The performance of the PCA-DCC-NL method lies between that of the DCC-NL method and the PCA method. This is probably because unlike the PCA method, the PCA-DCC-NL method benefits from the advantage of DCC-NL method in learning CCMs; however, its learning of conditional mean from the PCA method is less effective, so its performance is worse than that of the DCC-NL method.

Besides the learning of vast CCMs, another key advantage of the GRAND method is its capacity of detecting Granger causality. The unweighted adjacency matrix $\overline{\boldsymbol{W}}_t$, which generates $\boldsymbol{W}_t$ in (L2), indicates not only Granger causality from the viewpoint of conditional variance but also the non-zero elements of $\boldsymbol{\Sigma}_t$. Hence, for each Monte Carlo repetition, the GRAND method can apply the overall adjacency matrix $\overline{\boldsymbol{A}}_{t,\sigma}$ in (14) to learn $\overline{\boldsymbol{W}}_t$, and its learning

[2]Since the DCC-NL and BDF methods require zero-mean inputs, the conditional mean of the inputs is removed using the sample mean, as outlined in their respective papers.

TABLE L3
Out-of-sample ℓ_{total}, ℓ_{diag}, and $\ell_{off\text{-}diag}$ comparison.

	ℓ_{total}	ℓ_{diag}	$\ell_{\text{off-diag}}$
GRAND	**205.57**	**88.19**	**172.10**
GRAND_1	280.26	109.12	254.13
DCC-NL	425.14	270.12	265.62
BDF	701.33	312.44	445.36
PCA	3095.46	348.12	3038.12
PCA-DCC-NL	487.05	321.29	327.73

Note: For each evaluation metric, the best result is highlighted in boldface.

accuracy for Granger causality can be measured by

$$\text{Acc} = \frac{3}{T} \sum_{t \in \text{testing}} \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} I\left(\overline{A}_{t,\sigma}^{(i,j)} = \overline{W}_t^{(i,j)}\right),$$

where $\overline{A}_{t,\sigma}^{(i,j)}$ and $\overline{W}_t^{(i,j)}$ are the (i,j)-th elements of $\overline{\boldsymbol{A}}_{t,\sigma}$ and $\overline{\boldsymbol{W}}_t$, respectively.

Figure L1 presents the box plot of Acc across 100 Monte Carlo repetitions. Since the average value of Acc is 99.34% as shown in this figure, the GRAND method can be considered an efficient approach for detecting Granger causality in high-dimensional cases.

Lastly, we discuss the computational burden of the proposed GRAND method. Due to the use of deep learning models (such as the SGA model) and the high dimensionality of $\{r_{it}\}$, the computational burden of the GRAND method is expected to be heavy. For a single Monte Carlo sample, the average time to forecast one out-of-sample covariance matrix is approximately six hours when only one NVIDIA 4090 GPU is available. Fortunately, owing to the spirit of "divide and conquer" owned by the GRAND method, most computations can be parallelized, allowing substantial reductions in runtime. For instance, using two NVIDIA 4090 GPUs can shorten the computation time by about 50% compared to a single GPU. Regarding memory requirements, the GRAND method is relatively efficient. The training phase uses the Adam optimizer with a moderate batch size, and the forecast of each covariance matrix $\boldsymbol{\Sigma}_t$ only

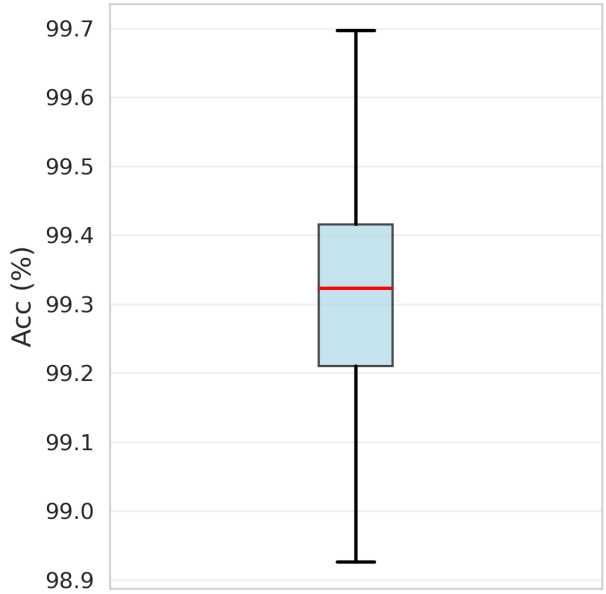


FIG L1. *Box plot of Acc for detecting Granger causality via the GRAND method.*

requires the corresponding cross-sectional input $\boldsymbol{X}_{t-1}$. As a result, a system with 32 GB of memory is sufficient to reproduce the simulation study without parallelization. In terms of scalability, given more computational resources, the GRAND method can be readily extended to deal with much larger N, benefiting from its divide-and-conquer design.

APPENDIX M: AN EXAMPLE FOR GROUP SELECTION

In this appendix, we provide an illustrating example to show how the encoder of the SGA discovers the group structure $\{\mathcal{M}_{k,t} : k \in \mathcal{K}_t\}$ (or equivalently, the graph $\mathcal{G}_t$) from the temporal embeddings $\{\boldsymbol{h}_{i,t}\}$, following the construction in Appendix A.1. For ease of presentation, we consider a small system with $N = 4$ individuals and set the hyperparameter $K^* = 2$ at a fixed time t. The whole procedure is summarized in Fig M1 and elaborated below.

First, as shown in Fig M1(a), we initially take all of $N = 4$ individuals as central nodes

(so $\boldsymbol{u}_{k,t} = \boldsymbol{h}_{k,t}$ for $k = 1, \ldots, 4$), and compute the cosine similarity $\gamma_{i,k,t} = \text{Cosine}(\boldsymbol{h}_{i,t}, \boldsymbol{u}_{k,t})$ between each individual i and central node k. To mitigate the impact of weak similarities, we then threshold $\{\gamma_{i,k,t}\}$ as in (A2): For each individual i, we retain $\gamma_{i,k,t}$ only for the $K^* = 2$ central nodes most similar to i (together with the node $k = i$ itself), and set the remaining entries to zero. This yields the screened similarities $\{\gamma^*_{i,k,t}\}$.

Second, as shown in Fig M1(b), we form a candidate group $\mathcal{M}_{k,t} = \{i : \gamma^*_{i,k,t} \neq 0\}$ for each central node k as in (A3), resulting in four candidate groups $\mathcal{M}_{1,t}, \ldots, \mathcal{M}_{4,t}$. Since a group containing only its own central node is meaningless, we discard $\mathcal{M}_{1,t}$ and $\mathcal{M}_{4,t}$, whose central nodes connect to no other individuals. Consequently, only two valid groups $\mathcal{M}_{2,t}$ and $\mathcal{M}_{3,t}$ remain, with the set of central nodes $\mathcal{K}_t = \{2, 3\}$ and the number of groups $K_t = 2$.

Third, as shown in Fig M1(c), for each remaining group $\mathcal{M}_{k,t}$ ($k \in \mathcal{K}_t$), we construct the adjacency matrix $\boldsymbol{A}_{k,t}$ whose (i, j)-th element equals one if $i, j \in \mathcal{M}_{k,t}$ with $i \neq j$, and zero otherwise. This gives two adjacency matrices $\boldsymbol{A}_{2,t}$ and $\boldsymbol{A}_{3,t}$, corresponding to $\mathcal{M}_{2,t}$ and $\mathcal{M}_{3,t}$, respectively, so the resulting graph is $\mathcal{G}_t = (\boldsymbol{V}, \boldsymbol{A}_t)$ with $\boldsymbol{V} = \{1, 2, 3, 4\}$ and $\boldsymbol{A}_t = \{\boldsymbol{A}_{2,t}, \boldsymbol{A}_{3,t}\}$. As illustrated in Fig M1(c), the groups $\mathcal{M}_{2,t}$ and $\mathcal{M}_{3,t}$ can be exactly re-constructed from $\boldsymbol{A}_{2,t}$ and $\boldsymbol{A}_{3,t}$, confirming the equivalence between $\mathcal{G}_t$ and $\{\mathcal{M}_{k,t} : k \in \mathcal{K}_t\}$ discussed in Appendix A.1 of the main text.

REFERENCES

Bai, J. and Ng, S. (2002). Determining the Number of Factors in Approximate Factor Models. *Econometrica* **70** 191–221.

Barigozzi, M. and Hallin, M. (2017). A Network Analysis of the Volatility of High Dimensional Financial Series. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* **66** 581–605.

Chen, H. J., Chen, S. J., Chen, Z. and Li, F. (2019). Empirical Investigation of an Equity Pairs Trading Strategy. *Management Science* **65** 370-389.

Christoffersen, P. F. (1998). Evaluating Interval Forecasts. *International Economic Review* **39** 841–862.

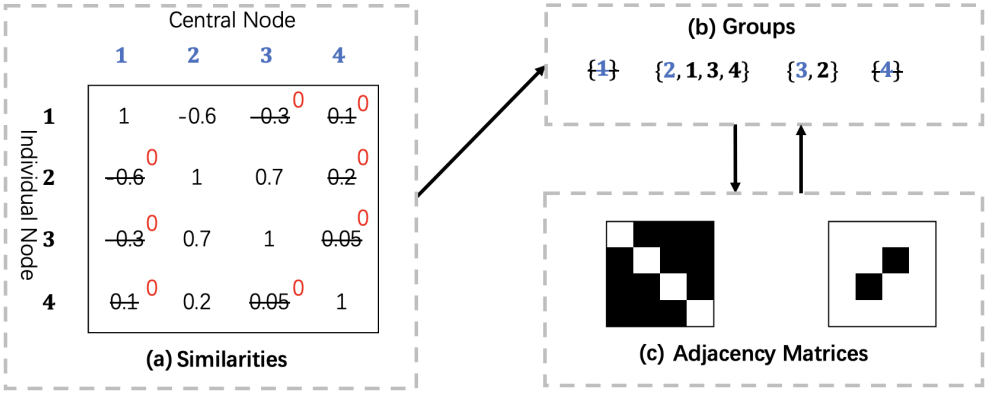


FIG M1. *(Color online) An example to illustrate spatial structure discovery with* $N = 4$ *and* $K^* = 2$ *at a fixed time* t*: (a) Calculate cosine similarities* $\{\gamma_{i,k,t}\}$ *and threshold them to obtain* $\{\gamma^*_{i,k,t}\}$*; (b) Construct four groups* $\{\mathcal{M}_{k,t} : k = 1, 2, 3, 4\}$ *based on* $\{\gamma^*_{i,k,t}\}$ *and discard two meaningless groups* $\mathcal{M}_{1,t}$ *and* $\mathcal{M}_{4,t}$*; (c) Form adjacency matrices* $\boldsymbol{A}_{2,t}$ *and* $\boldsymbol{A}_{3,t}$ *corresponding to groups* $\mathcal{M}_{2,t}$ *and* $\mathcal{M}_{3,t}$*, respectively, where black and white squares indicate one and zero, respectively. Note that groups* $\mathcal{M}_{2,t}$ *and* $\mathcal{M}_{3,t}$ *can also be re-constructed from adjacency matrices* $\boldsymbol{A}_{2,t}$ *and* $\boldsymbol{A}_{3,t}$*.*

CORNISH, E. A. and FISHER, R. A. (1938). Moments and Cumulants in the Specification of Distributions. *Revue de l'Institut international de Statistique* **5** 307–320.

CUI, L., HONG, Y., LI, Y. and WANG, J. (2024). A Regularized High-Dimensional Positive Definite Covariance Estimator with High-Frequency Data. *Management Science* **70** 7242–7264.

ENGLE, R. F. (1982). Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. *Econometrica* **50** 987–1007.

ENGLE, R. F., LEDOIT, O. and WOLF, M. (2019). Large Dynamic Covariance Matrices. *Journal of Business & Economic Statistics* **37** 363–375.

FAN, J., FAN, Y. and LV, J. (2008). High Dimensional Covariance Matrix Estimation Using a Factor Model. *Journal of Econometrics* **147** 186–197.

KASTNER, G. (2019). Sparse Bayesian Time-Varying Covariance Estimation in Many Dimensions. *Journal of Econometrics* **210** 98–115.

KUPIEC, P. H. (1995). Techniques for Verifying the Accuracy of Risk Measurement Models. *Journal of Derivatives* **3** 73–84.

YANG, X., ZHU, Z., LI, D. and ZHU, K. (2024). Asset Pricing via the Conditional Quantile Variational

Autoencoder. *Journal of Business & Economic Statistics* **42** 681–694.

Zhang, N. and Zhu, K. (2026). Quantiled Conditional Variance, Skewness, and Kurtosis by Cornish–Fisher Expansion. *Annals of Applied Statistics* **20** 131–147.

Zhou, J., Li, D., Pan, R. and Wang, H. (2020). Network GARCH Model. *Statistica Sinica* **30** 1723–1740.

Zhu, Z., Zhang, N. and Zhu, K. (2024). Big Portfolio Selection by Graph-based Conditional Moments Method. *Journal of Empirical Finance* **78** 101533.

Zhu, X., Pan, R., Li, G., Liu, Y. and Wang, H. (2017). Network Vector Autoregression. *Annals of Statistics* **45** 1096–1123.

Zhu, X., Wang, W., Wang, H. and Härdle, W. K. (2019). Network Quantile Autoregression. *Journal of Econometrics* **212** 345–358.