THE UNIVERSITY OF CHICAGO

ESSAYS IN FINANCIAL ECONOMETRICS

A DISSERTATION SUBMITTED TO
THE FACULTY OF THE UNIVERSITY OF CHICAGO
BOOTH SCHOOL OF BUSINESS
IN CANDIDACY FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

BY
CHAOXING DAI

CHICAGO, ILLINOIS
AUGUST 2024

To my younger brother Chaoqun Dai, in loving memory.

Table of Contents

LIST OF FIGURES vi

LIST OF TABLES vii

ACKNOWLEDGMENTS ix

ABSTRACT xi

1 KNOWING FACTORS OR FACTOR LOADINGS, OR NEITHER? EVALUATING ESTIMATORS OF LARGE COVARIANCE MATRICES WITH NOISY AND ASYNCHRONOUS DATA 1

1.1 Introduction 1

1.2 Model Setup and Assumptions 6

1.2.1 Notation 6

1.2.2 Factor Dynamics 6

1.2.3 Sparsity 9

1.2.4 Microstructure Noise 10

1.2.5 Asynchronicity 13

1.3 Three Estimators and Their Asymptotic Properties 15

1.3.1 Time-Series Regression (TSR) 16

1.3.2 Cross-Sectional Regression (CSR) 18

1.3.3 Principal Component Analysis (PCA) 21

1.3.4 A Comparison of the Three Estimators 24

1.4 Practical Considerations 25

1.4.1 Choice of Tuning Parameters 25

1.4.2 Choice of r and $f(n, d)$ 25

1.4.3 Choice of the Thresholding Methods 26

1.5 Monte Carlo Simulations 27

1.6 Empirical Applications 34

1.6.1 Data 34

1.6.2 Out-of-Sample Portfolio Allocation 41

1.7 Conclusion 45

1.8 Appendix: Mathematical Proofs 46

1.8.1 Proof of Theorem 1 46

1.8.2 Proof of Theorem 2 65

1.8.3 Proof of Theorem 3 66
1.8.4 Proof of Theorem 4 67

2 LARGE SCALE REALIZED VOLATILITY FORECASTING WITH MACHINE LEARNING 76
2.1 Introduction 76
2.2 Methodology 80
2.2.1 Problem Setup 80
2.2.2 OLS Methods 81
2.2.3 Machine Learning Methods 88
2.3 Data 94
2.3.1 Raw Data 94
2.3.2 Data Processing 96
2.3.3 Training Scheme 98
2.3.4 Evaluation Metrics 100
2.4 Predictive Performance 101
2.5 Utility Benefits 120
2.6 Conclusion 127

REFERENCES 129

List of Figures

1.1 Trading Intensity of Assets 35

1.2 Time Series of Factors Used in TSR 37

1.3 Time Series of Estimated Factors by CSR 38

1.4 Time Series of Estimated Factors by PCA 39

1.5 Generalized Correlation Plot 40

1.6 Out-of-Sample Risk of the S&P 500 Portfolio 42

1.7 Out-of-Sample Risk of the S&P 100 Portfolio 42

1.8 Out-of-Sample Risk of the Dow Jones 30 Portfolio 43

2.1 Quantiles of RVs 97

2.2 Variable Importance By Model, S&P 500 Stocks 114

2.3 Variable Importance By Model, US Stocks 115

2.4 Variable Importance Ranking, S&P 500 Stocks 116

2.5 Variable Importance Ranking, US Stocks 117

List of Tables

1.1 Pre-Averaging vs. Subsampling Using Location Thresholding 29

1.2 Pre-Averaging vs. Subsampling Using Hard and Soft Thresholding 31

1.3 Pre-Averaging vs. Subsampling Using SCAD, and AL Thresholding 32

1.4 Impact of the Selected Number of Factors on the PCA Method 33

1.5 Diebold-Mariano Tests for the Out-of-Sample Risk Comparison 44

1.6 Diebold-Mariano Tests for the Out-of-Sample Risk Comparison 45

2.1 Sample Size by Data Cleaning Steps 96

2.2 Hyperparameters for Machine Learning Models 99

2.3 Out-of-Sample Forecasting Performance for S&P 500 Stocks 102

2.4 Out-of-Sample Forecasting Performance for US Stocks 103

2.5 Individual vs Pooled Fit, S&P 500 Stocks 105

2.6 Individual vs Pooled Fit, US Stocks 105

2.7 Insanity Filter, S&P 500 Stocks 107

2.8 Insanity Filter, US Stocks 108

2.9 Diebold-Mariano Tests, S&P 500 Stocks 109

2.10 Diebold-Mariano Tests, US Stocks 110

2.11 Feature Estimation by Group, S&P 500 Stocks 112

2.12 Feature Estimation by Group, US Stocks 113

2.13 Relative R2 by Training Years, S&P 500 Stocks 118

2.14 Relative R2 by Random Seeds, S&P 500 Stocks 119

2.15 Relative R2 by Random Seeds, US Stocks 119

2.16 Realized Utility for S&P 500 Stocks 123

2.17 Realized Utility for US Stocks 124

2.18 Realized Utility on Size Portfolios for S&P 500 Stocks 125

2.19 Realized Utility on Size Portfolios for US Stocks 126

Acknowledgments

First and foremost, I would like to express my sincerest gratitude to my advisor, Professor Dacheng Xiu, for his guidance and support throughout my academic journey. Professor Xiu's profound insights, rigorous approach to research, and exceptional research taste continues to inspire me. I am truly grateful for his mentorship and dedication to my growth as a researcher.

I am deeply grateful to my committee members: Professor Jeffrey R. Russell, Professor Ekaterina Smetanina, and Professor Ruey S. Tsay. Their valuable guidance and constructive feedback have significantly shaped my research and academic journey. Additionally, I would like to express my gratitude to my master advisor, Professor Matthew Stephens, for introducing me to the field of statistical computing.

I would like to express my gratitude to Kun Lu for the collaboration in the first chapter of the dissertation, and to Yichen Ji, Chad Schmerling, and Shunqi Zhang for their research assistantship and insightful discussions in the second chapter.

I am grateful for the generous financial support and resources provided by the Booth School of Business. The PhD office at Booth has consistently shown responsiveness and attentiveness to our PhD students, and I would like to extend my thanks to Raven Davis, Cynthia Hillman, Amity James, and Kimberly Mayer for their outstanding service. Additionally, I would like to convey my heartfelt appreciation to Malaina Brown, the director of the PhD program, for her unwavering support throughout my entire PhD journey.

I am fortunate to have met a number of inspiring friends at Booth, especially Rui Da, Jinzhi Lu, Xiao Zhang, Shirley Zhang, Wenxi Li, Zhongli Li, Shirley Lu, and many others. I am sincerely grateful for their collaboration in coursework, stimulating research discussions, and enjoyable leisure hangouts. Their friendship and support have greatly enriched my academic experience and personal growth during my time at Booth.

Finally, I would like to thank my family members: my wife, Jin Deng, for her love and trust over the years, and our daughter Yiling, for bringing us happiness and joy. A special thank you to my cat, Sayuri, for her companionship since the inception of my PhD journey. My deepest gratitude to my parents for their endless love and support throughout my life.

Abstract

This dissertation consists of two essays in financial econometrics.

In the first essay, coauthored with Kun Lu and Dacheng Xiu, we investigate estimators of factor-model-based large covariance (and precision) matrices using high-frequency data, which are asynchronous and potentially contaminated by the market microstructure noise. Our estimation strategies rely on the pre-averaging method with refresh time to solve the microstructure problems, while using three different specifications of factor models with a variety of thresholding methods, respectively, to battle the curse of dimensionality. To estimate a factor model, we either adopt the time-series regression (TSR) to recover loadings if factors are known, or use the cross-sectional regression (CSR) to recover factors from known loadings, or use the principal component analysis (PCA) if neither factors nor their loadings are assumed known. We compare the convergence rates in these scenarios using the joint in-fill and increasing dimensionality asymptotics. To evaluate the empirical trade-off between robustness to model misspecification and statistical efficiency among all 30 combinations of estimation strategies, we run a horse race on the out-of-sample portfolio allocation with Dow Jones 30, S&P 100, and S&P 500 index constituents, respectively, and find the pre-averaging-based strategy using TSR or PCA with location thresholding dominates, especially over the subsampling-based alternatives.

In the second essay, we leverage high-frequency data from over 10,000 stocks spanning more than two decades to apply machine learning algorithms to the task of forecasting realized volatility (RV). By exploiting the nonlinear relationships between RV and a wide range of features and panel information, machine learning algorithms, particularly neural networks, exhibit significantly enhanced performance compared to traditional ordinary least square methods. Our findings suggest that employing a universal model that combines all stocks outperforms individual models tailored to each stock, resulting in more reliable and less extreme predictions. The ensemble neural networks achieve a relative R2 of over 18% compared to the benchmark HAR model for S&P 500 stocks and over 11% for US stocks. Furthermore, utilizing a straightforward utility-based framework, we show that neural networks offer a 40 basis points advantage over the HAR model for S&P 500 stocks and a 30 basis points advantage for US stocks.

Chapter 1

Knowing Factors or Factor Loadings, or Neither? Evaluating Estimators of Large Covariance Matrices with Noisy and Asynchronous Data

This chapter is a joint work with Kun Lu and Dacheng Xiu, and it is pulished on Journal of Econometrics (https://doi.org/10.1016/j.jeconom.2018.09.005).

1.1 Introduction

Factor models provide a parsimonious representation of the dynamics of asset returns, as motivated by Ross (1976)'s arbitrage pricing theory. Since this seminal work, researchers have devoted significant effort to the search for proxies of factors (e.g., Fama and French (1993) and Fama and French (2015)) or characteristics of stocks (e.g., Daniel and Titman (1997)) to explain the cross-sectional variation of expected returns. These factors and characteristics also serve as natural candidates for factors and loadings that drive the time-series dynamics of stock returns. In this paper, we make use of a factor model to assist the estimation of large covariance matrices among stock returns.

The factor-model specification leads to a low-rank plus sparse structure of the covariance matrix, which guarantees a well-conditioned estimator as well as a desirable performance of its inverse (precision matrix). To estimate the low-rank component, we consider three scenarios: known factors, known factor loadings, or unknown factors and factor loadings. In the first two scenarios, we employ a time-series regression (TSR) or a cross-sectional regression (CSR) to estimate the unknown loadings or factors, using either portfolios and ETFs as proxies for factors, or characteristics as proxies for factor loadings. In the third scenario, we employ the principal component analysis (PCA) to identify latent factors and their loadings. Combining the estimated factors and/or their loadings yields the low-rank component of the covariance matrix. With respect to the sparse component, we adopt a variety of thresholding methods that warrant positive semi-definite estimates of the covariance matrix.

In addition, we use the large cross section of transaction-level prices available at high frequencies. High-frequency data provide a unique opportunity to measure the variation and covariation among stock returns. The massive amount of data facilitates the use of simple nonparametric estimators within a short window, such as the sample covariance matrix estimator on a daily, weekly, or monthly basis, so that several issues associated with parametric estimation using low-frequency time series covering a long timespan become irrelevant, such as structural breaks and time-varying parameters; see, e.g., Aït-Sahalia and Jacod (2014) for a review. However, the microstructure noise and the asynchronous arrival of trades, which come together with intraday data, result in biases of the sample covariance estimator with data sampled at a frequency higher than, say, every 15 minutes, exacerbating the curse of dimensionality due to data elimination.

By adapting the pre-averaging estimator designed for low-dimensional covariance matrix estimation, e.g., Jacod, Li, Mykland, Podolskij, and Vetter (2009), we construct noise-robust estimators for large covariance matrices, making use of a factor model in each of the three aforementioned scenarios. Using the large deviation theory of martingales, we establish the desired consistency of our covariance matrix estimators under the infinity norm (on the vector space) and the weighted quadratic norm, as well as the precision matrix estimators under the operator norm. Moreover, we show TSR converges as the sample size increases, regardless of a fixed or an increasing dimension. By contrast, the convergence of CSR and PCA requires a joint increase of the dimensionality and the sample size – the so-called blessings of dimensionality; see Donoho et al. (2000).

Empirically, we analyze the out-of-sample risk of optimal portfolios in a horse race among various estimators of the covariance matrix as inputs. The portfolios comprise constituents of Dow Jones 30, S&P 100, and S&P 500 indices, respectively. We find covariance matrix estimators based on pre-averaged returns sampled at refresh times outperform those based on returns subsampled at a fixed 15-minute frequency, for almost all combinations of estimation strategies and thresholding methods. Moreover, either TSR or PCA, plus the location thresholding that utilizes the Global Industry Classification Standards (GICS) codes, yield the best performance for constituents of the S&P 500 and S&P 100 indices, whereas TSR dominates in the case of Dow Jones 30.

Our paper is closely related to a growing literature on continuous-time factor models for high-frequency data. Fan, Furger, and Xiu (2016) and Aït-Sahalia and Xiu (2017) develop the asymptotic theory for large dimensional factor models with known and unknown factors, respectively, assuming a synchronous and noiseless dataset. Their simulations show a clear breakdown of either TSR or PCA when noise is present and the sampling frequency is more than every 15 minutes. Pelger (2015a) and Pelger (2015b) develop the central limit results of such models in the absence of the noise. Their asymptotic results are element-wise, whereas the theoretical results in this paper focus on matrix-wise properties. Wang and Zou (2010) propose the first noise-robust covariance matrix estimator in the high-dimensional setting, by imposing the sparsity assumption on the covariance matrix itself; see also Tao, Wang, and Zhou (2013), Tao, Wang, Yao, and Zou (2011), Tao, Wang, and Chen (2013), and Kim, Wang, and Zou (2016) for related results. Brownlees, Nualart, and Sun (2017) impose the sparsity condition on the inverse of the covariance matrix or the inverse of the idiosyncratic covariance matrix. By contrast, we impose the sparsity assumption on the covariance of the idiosyncratic components of a factor model, as motivated by the economic theory, which also fits the empirical data better.

Our paper is also related to the recent literature on the estimation of the low-dimensional covariance matrix using noisy high-frequency data. The noise-robust estimators include, among others, the multivariate realized kernels by Barndorff-Nielsen, Hansen, Lunde, and Shephard (2011), the quasi-maximum likelihood estimator by Aït-Sahalia, Fan, and Xiu (2010) and Shephard and Xiu (2017), the pre-averaging estimator by Christensen, Kinnebrock, and Podolskij (2010), the local method of moments by Bibinger, Hautsch, Malec, and Reiss (2014), and the two-scale and multi-scale estimators by Zhang (2011) and Bibinger (2012). Shephard and Xiu (2017) document the advantage of using a factor model in their empirical study, even when the dimension of assets is as low as 13. We build our estimator based on the pre-averaging method because of its simplicity in deriving the in-fill and high-dimensional asymptotic results. Allowing for increasing dimensionality asymptotics sheds light on important statistical properties of the covariance matrix estimators, such as minimum and maximum eigenvalues, condition numbers, etc, which are critical for portfolio allocation exercises. Aït-Sahalia and Xiu (2019b) develop a related theory of PCA for low-dimensional high-frequency data.

Our paper is also related to the recent literature on large covariance matrix estimation with low-frequency data. Fan, Fan, and Lv (2008) propose a large covariance matrix estimator using a strict factor model with observable factors. Fan, Liao, and Mincheva (2011) extend this result to approximate factor models. Fan, Liao, and Mincheva (2013) develop the POET method for models with unobservable factors. Alternative covariance matrix estimators include the shrinkage method by Ledoit and Wolf (2004) and Ledoit and Wolf (2012), and the thresholding method proposed by Bickel and Levina (2008a), Bickel and Levina (2008b), Cai and Liu (2011), and Rothman, Levina, and Zhu (2009).

Our paper shares the theoretical insight with the existing literature of factor models specified in discrete time. Bai and Ng (2002) and Onatski (2010) propose estimators to determine the number of factors. Bai (2003) develops the element-wise inferential theory for factors and their loadings. These papers, including ours, allow for more general models than the approximate factor models introduced in Chamberlain and Rothschild (1983). The above factor models are static, as opposed to the dynamic factor models in which the lagged values of the unobserved factors may also affect the observed dependent variables. Inference on dynamic factor models are developed in Forni, Hallin, Lippi, and Reichlin (2000), Forni and Lippi (2001), Forni, Hallin, Lippi, and Reichlin (2004), and Doz, Giannone, and Reichlin (2011); see Croux, Renault, and Werker (2004) for a discussion.

Factor models based on stock characteristics date back to Rosenberg (1974), who suggests modeling factor betas of stocks as linear functions of observable security characteristics. Connor and Linton (2007) and Connor, Hagmann, and Linton (2012) further extend this model to allow for nonlinear or nonparametric functions. One of our covariance matrix estimators, namely, CSR, is designed to leverage the linear factor model with characteristics as loadings. Such an estimation strategy is widely used in the financial industry, but is largely ignored by the academic literature.[1] Our asymptotic analysis of this estimator fills in this gap.

The rest of the paper is structured as follows. In Section 1.2, we set up the model and discuss model assumptions. Section 1.3 proposes the estimation procedure for each scenario of the factor model and establishes the asymptotic properties of these estimators. Section 1.4 discusses the choice of tuning parameters. Section 1.5 provides Monte Carlo simulation evidence. Section 1.6 evaluates these estimators in an out-of-sample optimal portfolio allocation race. The appendix contains all mathematical proofs.

[1] Barra Inc., which was acquired by MSCI Inc., was a leading provider of this type of covariance matrix to practitioners; see, e.g., Kahn, Brougham, and Green (1998).

1.2 Model Setup and Assumptions

1.2.1 Notation

Let $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}, \mathbb{P})$ be a filtered probability space. Throughout this paper, we use $\lambda_j(A)$, $\lambda_{\min}(A)$, and $\lambda_{\max}(A)$ to denote the j-th (descending order), the minimum, and the maximum eigenvalues of a square matrix A, respectively. In addition, we use $\|A\|_1$, $\|A\|$, $\|A\|_{\mathrm{F}}$, and $\|A\|_\Sigma$ to denote the $\mathbb{L}_1$ norm, the operator norm (or $\mathbb{L}_2$ norm), the Frobenius norm, and the weighted quadratic norm of a matrix A, that is, $\max_j \sum_i |A_{ij}|$, $\sqrt{\lambda_{\max}(A^\intercal A)}$, $\sqrt{\mathrm{Tr}(A^\intercal A)}$, and $d^{-1/2}\|\Sigma^{-1/2} A \Sigma^{-1/2}\|_{\mathrm{F}}$, respectively. Note $\|A\|_\Sigma$ is only defined for a $d \times d$ square matrix. We use $\mathbb{I}_d$ to denote a $d \times d$ identity matrix. All vectors are regarded as column vectors, unless otherwise specified. When A is a vector, both $\|A\|$ and $\|A\|_{\mathrm{F}}$ are equal to its Euclidean norm. We also use $\|A\|_{\mathrm{MAX}} = \max_{i,j} |A_{ij}|$ to denote the $\mathbb{L}_\infty$ norm of A on the vector space. We use e_i to denote a d-dimensional column vector whose ith entry is 1 and 0 elsewhere. We write $A_n \asymp B_n$ if $|A_n|/|B_n| = O(1)$. We use C to denote a generic constant that may change from line to line.

1.2.2 Factor Dynamics

Let Y be a d-dimensional log-price process, X be an r-dimensional factor process, Z be the idiosyncratic component, and β be a constant factor loading matrix of size $d \times r$. We make the following assumption about their dynamic relationship:

Assumption 1. *Suppose Y_t follows a continuous-time factor model:*

$$Y_t = \beta X_t + Z_t, \tag{1.1}$$

in which X_t is a continuous Itô semimartingale, that is,

$$X_t = \int_0^t h_s \, ds + \int_0^t \eta_s dW_s,$$

and Z_t is another continuous Itô semimartingale, satisfying

$$Z_t = \int_0^t f_s ds + \int_0^t \gamma_s dB_s,$$

where W_s and B_s are standard Brownian motions. In addition, h_s and f_s are progressively measurable. Moreover, the processes η_s and γ_s are càdlàg, and, writing $e_s = \eta_s \eta_s^\intercal$, $g_s = \gamma_s \gamma_s^\intercal$, e_s, e_{s-}, g_s, and g_{s-} are positive-definite. Finally, for all $1 \leqslant i, j \leqslant r$, $1 \leqslant k, l \leqslant d$, $|\beta_{kj}| \leqslant C$, for some $C > 0$, and there exists a locally bounded process H_s, such that $|h_{i,s}|$, $|\eta_{ij,s}|$, $|\gamma_{kl,s}|$, $|e_{ij,s}|$, $|f_{kl,s}|$, and $|g_{kl,s}|$ are bounded by H_s uniformly for $0 \leqslant s \leqslant t$.

Assumption 1 is fairly general except for two important limitations: β is constant and jumps are excluded. The same constant β assumption has been adopted by e.g., Todorov and Bollerslev (2010) in a low-dimensional setting and Fan, Furger, and Xiu (2016) and Aït-Sahalia and Xiu (2017) in high-dimensional settings. In our empirical study, we impose a constant β within each month, because β is available from the MSCI Barra at a monthly frequency.

To emphasize and highlight the theoretical trade-offs in the estimation from a high-dimensional perspective, we exclude jumps from our theoretical analysis to avoid delving into unnecessary technicalities. As a result, our empirical covariance matrix estimates contain the quadratic covariation contributed by co-jumps. Although co-jumps may be an important component, we do not find it particularly important to separate them from the total quadratic covariation for the large portfolio allocation exercise in which we are interested. Moreover, separating jumps would substantially complicate the estimation procedure, e.g., with more tuning parameters. In this paper, we desire simpler estimators while leaving analysis of jumps for future work.

Next, we impose the usual exogeneity assumption:

Assumption 2. *For any $1 \leqslant k \leqslant d$, and $1 \leqslant l \leqslant r$, we have $[Z_{k,s}, X_{l,s}] = 0$, for any*

$0 \leqslant s \leqslant t$*, where* $[\cdot,\cdot]$ *denotes the quadratic covariation.*

Our main goal is to estimate the integrated covariance matrix of Y, denoted as $\Sigma = \frac{1}{t}\int_0^t c_s ds$, where c_s is the spot covariance of Y_s. Assumptions 1 and 2 infer a factor structure on c_s:

$$c_s = \beta e_s \beta^\intercal + g_s, \quad 0 \leqslant s \leqslant t.$$

As a result, we can decompose the quadratic covariation of Y within $[0,t]$, Σ, as

$$\Sigma = \beta \mathrm{E} \beta^\intercal + \Gamma,$$

where

$$\Sigma = \frac{1}{t}\int_0^t c_s ds, \quad \Gamma = \frac{1}{t}\int_0^t g_s ds, \quad \text{and} \quad \mathrm{E} = \frac{1}{t}\int_0^t e_s ds.$$

We omit the dependence of Σ, E, and Γ on t for brevity in notation.

Next we impose that factors are pervasive, in the sense that they influence a large number of assets; see, e.g., Chamberlain and Rothschild (1983).

Assumption 3. E *has distinct eigenvalues, with* $\lambda_{\min}(\mathrm{E})$ *bounded away from 0. Moreover, there exists some positive-definite matrix* B *such that* $\left\|d^{-1}\beta^\intercal\beta - B\right\| = o(1)$*, as* $d \to \infty$*, and* $\lambda_{\min}(B)$ *is bounded away from 0.*

As in Aït-Sahalia and Xiu (2017), Assumption 3 leads to the identification of the number of factors when factors and their loadings are latent. Fan, Furger, and Xiu (2016) also use it in the case of known factors, when building the operator norm bound for the precision matrix. Such an assumption may also be restrictive in that it excludes the existence of weak factors; see, e.g., Onatski (2010). Dealing with weak factors requires a rather different setup, so we leave it for future work.

1.2.3 Sparsity

For high-dimensional covariance matrix estimation, a certain "sparsity" condition is necessary for dimension reduction, in addition to a factor model, because the idiosyncratic component of the covariance matrix, once the low-rank component is removed, is equally large. One cannot obtain a good estimate of it without additional assumptions. Sparsity seems a reasonable choice for both known-factor and unknown-factor models given the empirical findings of Aït-Sahalia and Xiu (2017).

We define m_d as the degree of sparsity of Γ, where

$$m_d = \max_{i \leqslant d} \sum_{j \leqslant d} |\Gamma_{ij}|^q, \quad \text{for some } q \in [0, 1).$$

The sparsity assumption imposes $m_d/d \to 0$. This notion of sparsity follows from Rothman, Levina, and Zhu (2009) and Bickel and Levina (2008b). When $q = 0$, m_d is equal to the maximum number of non-zero elements in rows of Γ, the usual notion used by Bickel and Levina (2008a). In this case, sparsity simply means each row of Γ contains few non-zero elements. Fan, Furger, and Xiu (2016) and Aït-Sahalia and Xiu (2017) consider this special case, while requiring Γ to be block diagonal. Our assumption below is more general.

Under this notion of sparsity, we have

$$\|\Gamma\| \leqslant \|\Gamma\|_1 = \max_{i \leqslant d} \sum_{j \leqslant d}^{d} |\Gamma_{ij}| = O(m_d).$$

Therefore, imposing $m_d/d \to 0$ creates a gap between eigenvalues of the low rank ($\beta \mathrm{E} \beta^\intercal$) and the sparse ($\Gamma$) components of the covariance matrix Σ, which is essential for identification in a latent factor model specification, and for estimation of the factor models in general.

To use sparsity for estimation, we define a class of thresholding functions $s_\lambda(z) : \mathbb{R} \to \mathbb{R}$, which satisfies

$$(i)\ |s_\lambda(z)| \leqslant |z|; \quad (ii)\ s_\lambda(z) = 0 \text{ for } |z| < \lambda; \quad (iii)\ |s_\lambda(z) - z| \leqslant \lambda.$$

As discussed in Rothman, Levina, and Zhu (2009), Condition (i) imposes shrinkage, condition (ii) enforces thresholding, and condition (iii) restricts the amount of shrinkage to be no more than λ. The exact three requirements of $s_\lambda(\cdot)$ ensure desirable statistical properties of the estimated covariance matrix. Examples of such thresholding functions we use include hard thresholding, soft thresholding, smoothly clipped absolute deviation (SCAD) (Fan and Li (2001)), and adaptive lasso (AL) (Zou (2006)):

$$s_\lambda^{\text{Hard}}(z) = z\mathbf{1}(|z| > \lambda), \quad s_\lambda^{\text{Soft}}(z) = \text{sign}(z)(|z| - \lambda)_+, \quad s_\lambda^{\text{AL}}(z) = \text{sign}(z)(|z| - \lambda^{\eta+1}|z|^{-\eta})_+,$$

$$s_\lambda^{\text{SCAD}}(z) = \begin{cases} \text{sign}(z)(|z| - \lambda)_+, & \text{when } |z| \leqslant 2\lambda; \\ \frac{(a-1)z - \text{sign}(z)a\lambda}{a-2}, & \text{when } 2\lambda < |z| \leqslant a\lambda; \cdot \\ z, & \text{when } a\lambda < |z|. \end{cases}$$

where $a = 3.7$ and $\eta = 1$, as suggested by Rothman, Levina, and Zhu (2009). We adopt these functions in the construction of the estimators in Section 1.3. Although these choices lead to the same convergence rate from our analysis, the resulting finite sample performance of the covariance matrices differ quite a bit, which we investigate in simulations and the empirical study.

1.2.4 Microstructure Noise

We analyze three scenarios of factor models, depending on whether the factor X or its loading β are known. We use the term “known” instead of “observable”, because even if we assume factor X can be proxied by certain portfolios in the literature, for instance, the Fama-French three factors by Fama and French (1993), we allow for potential microstructure noise so that the true factors are always latent in our setup.

The first scenario assumes X is known, in which case, we denote the observed factor as $X^\star$:

$$Y^\star_{t^i_j} = Y_{t^i_j} + \varepsilon^y_{t^i_j}, \quad X^\star_{t^i_j} = X_{t^i_j} + \varepsilon^x_{t^i_j},$$

for $1 \leqslant i \leqslant d$ and $1 \leqslant j \leqslant N^i_t$, where ε^y and ε^x are some additive noises associated with the observations at sampling times t^i_js, t^i_j denotes the arrival time of the jth transaction of asset i, and N^i_t is the number of transactions for asset i. We can thereby rewrite the factor model (1.1) as

$$Y^\star_{t^i_j} = \beta X^\star_{t^i_j} + Z^\star_{o,t^i_j}, \tag{1.2}$$

where $Z^\star_{o,t^i_j} = Z_{t^i_j} + \varepsilon^y_{t^i_j} - \beta\varepsilon^x_{t^i_j}$. Barring from the noise, this model is a standard linear regression. In the empirical study, we regard those portfolios that are useful to explain the cross section of expected asset returns as factors, including the five Fama-French factors (Fama and French (2015)) and the momentum factor (Carhart (1997)).[2] We also add industry portfolios as suggested by King (1966).

The second scenario assumes β is known and perfectly observed, yet Y is again contaminated, so we can write the model as

$$Y^\star_{t^i_j} = \beta X_{t^i_j} + Z^\star_{u,t^i_j}, \tag{1.3}$$

where $Z^\star_{u,t^i_j} = Z_{t^i_j} + \varepsilon^y_{t^i_j}$. This model dates back to Rosenberg (1974), who developed a factor model of stock returns in which the factor loadings of stocks are linear functions of observable security characteristics. This model is equivalent to a model with characteristics as βs associated with some linear latent factors. In the empirical study, we use 13 characteristics

[2] Although these factors explain the cross section of expected returns, they also account for significant variations in the time series of realizations.

obtained from the MSCI Barra, a leading company that provides factors and covariance matrices using this method.

In the third scenario, we only observe a noisy Y, so the model can be written into the same form as (1.3). This model is a "noisy" version of the approximate factor model by Chamberlain and Rothschild (1983) and Bai (2003), which can only be identified as the dimension of Y increases to ∞, thanks to the "blessing" of the dimensionality.

With respect to the microstructure noises, following Kim, Wang, and Zou (2016), we assume

Assumption 4. *Both $\{\varepsilon_i^x\}$ and $\{\varepsilon_i^y\}$ have the following structure:*

$$\varepsilon_{i,t_j^i} = u_{i,t_j^i} + v_{i,t_j^i},$$

where, for each $1 \leqslant i \leqslant d$, $1 \leqslant j \leqslant N_t^i$, writing $\Delta_{t_l^i} = t_l^i - t_{l-1}^i$,

$$u_{i,t_j^i} = \sum_{l=0}^{\infty} \rho_{i,t_{j-l}^i} \xi_{i,t_{j-l}^i}, \quad v_{i,t_j^i} = \sum_{l=0}^{\infty} b_{i,t_{j-l}^i} \Delta_{t_{j-l}^i}^{-1/4} [\widetilde{B}_{i,t_{j-l}^i} - \widetilde{B}_{i,t_{j-l-1}^i}].$$

We assume ξ_{i,t_l^i} and $\xi_{j,t_{l'}^j}$ are random variables with mean 0, and independent when $l \neq l'$, but potentially dependent for $l = l'$. Also, ρ_{i,t_l^i} is bounded in probability with $\sum_{l=0}^{\infty} |\rho_{i,t_l^i}| < \infty$ uniformly in i, and ξ_{i,t_l^i} is independent of the filtration $\{\mathcal{F}_t\}$ generated by X and Z. Moreover, $\widetilde{B}_i$ is a Brownian motion independent of ξ_i but potentially correlated with W and B of Assumption 1, and b_{i,t_l^i} is adapted to the filtration $\{\mathcal{F}_{t_l^i}\}$, and bounded in probability with $\sum_{l=0}^{\infty} |b_{i,t_l^i}| < \infty$ uniformly in i. Additionally, we assume there exists $\varsigma > 2$, such that $\max_{i,j} \mathbb{E}|\varepsilon_{i,t_j^i}|^{2\varsigma} < \infty$.

The microstructure noise has two independent components: u and v, where u allows for serial dependence, and v allows for correlation with returns. Here v_{i,t_j^i} has scaled Brownian increments, where the scale factor is of order $\Delta_{t^i_{j-l}}^{-1/4}$.[3] This assumption is motivated from the microstructure theory and existing empirical findings that order flows tend to cluster in the same direction and to be correlated with returns in the short run; see, e.g., Hasbrouck (2007), Brogaard, Hendershott, and Riordan (2014). This assumption is therefore more realistic in particular for data sampled at ultra-high frequencies. Kalnina and Linton (2008) and Barndorff-Nielsen, Hansen, Lunde, and Shephard (2011) adopt a similar assumption.

1.2.5 Asynchronicity

Because the transactions arrive asynchronously, adpoting the refresh time sampling scheme proposed by Martens (2004) prior to estimation is common. The first refresh time is defined as $t_1 = \max\left\{t_1^1, t_1^2, \ldots, t_1^d\right\}$. The subsequent refresh times are defined recursively as

$$t_{j+1} = \max\left\{t^1_{N^1_{t_j}+1}, \ldots, t^d_{N^d_{t_j}+1}\right\},$$

where $N^i_{t_j}$ is the number of transactions for asset i prior to time t_j. We denote by n the resulting sample size after refresh time sampling.

Effectively, this sampling scheme selects the most recent transactions at refresh times, which avoids adding zero returns artificially, because by design, all assets have at least one update between two refresh times. By comparison, the alternative previous-tick subsampling scheme by Zhang (2011) discards more data in order to avoid artificial zero returns. That said, the refresh time scheme is notoriously influenced by the most illiquid asset, which largely determines the number of observations after sampling. Pair-wise refresh time, as suggested by Aït-Sahalia, Fan, and Xiu (2010), is more efficient for entry-wise consistency of the covariance matrix. In the same spirit, Hautsch, Kyj, and Oomen (2012) adopt a more general strategy that conducts refresh-time sampling on blocks of assets formed by sorting

[3] The scaled factor is slightly more restrictive than that of the low dimensional setting considered in Ikeda (2016) and Varneskov (2016), where they use $\Delta_{t^i_{j-l}}^{-1/2}$. This rate only allows for moderate endogeneity which our estimator is robust to.

on liquidity. The resulting covariance matrix has desirable entry-wise consistency, but its matrix-wise properties are rather involved to analyze. Since our focus is the consistency under matrix-wise norms, we adopt the refresh time sampling throughout. We also find empirically that the refresh time approach delivers satisfactory performance. We make the following assumption on the observation times, following Kim, Wang, and Zou (2016):

Assumption 5. *The observation time t^i_js, $1 \leqslant j \leqslant N^i_t$, $1 \leqslant i \leqslant d$, are independent of the price process X_t and Z_t, and the noise ε^x and ε^y. We assume the intervals between two adjacent observations are independent, and that there exist constants $\bar{n}$, C, and $\varsigma > 2$, such that*

$$\max_{1 \leqslant i \leqslant d} \mathbb{E}|t^i_j - t^i_{j-1}|^a \leqslant C\bar{n}^{-a}, \text{ for any } 1 \leqslant a \leqslant 2\varsigma,$$

and that $c_1\bar{n} \leqslant n \leqslant c_2\bar{n}$ holds with probability approaching 1, where c_1 and c_2 are some positive constants.

A large literature is devoted to the modeling of durations, i.e., time intervals between adjacent transactions, since the seminal paper by Engle and Russell (1998), which proposes an autoregressive conditional duration (ACD) model and shows this parametric model can successfully describe the evolution of time durations for (heavily traded) stocks. Our focus here is the covariance matrix of returns, so we are agnostic about the dynamics of durations. The independence between durations and prices is a strong assumption, yet is commonly used in the literature, with the exception of Li, Mykland, Renault, Zhang, and Zheng (2014). This assumption means we can make our inference regarding the times of trades, and therefore refresh times, fixed.

To simplify the notation, we treat n as if it were deterministic in what follows. Also, we use $Y^\star_{t_i}$ to denote the most recent observation prior to or at the ith refresh time, and relabel the associated noise as ε^x_i and ε^y_i. Note $Y^\star_{t_i} - \varepsilon^y_i$ is not necessarily equal to Y_{t_i}. Instead, it equals the value of Y at actual transaction times. The same convention applies to $X^\star$ and $Z^\star$.

1.3 Three Estimators and Their Asymptotic Properties

We now proceed to the estimators and their asymptotic properties. Our results rely on the joint large sample ($n \to \infty$) and increasing dimensionality ($d \to \infty$) asymptotics, with a fixed number of factors r and a fixed time window $[0, t]$.

To deal with the bias due to the microstructure noise, we adopt the pre-averaging method proposed by Jacod, Li, Mykland, Podolskij, and Vetter (2009) to pre-weight the returns. Specifically, we divide the whole sample into a sequence of blocks, with the size of each block being k_n, which satisfies:

$$\frac{k_n}{n^{1/2+\delta}} = \theta + o(n_\delta^{-1/2}), \tag{1.4}$$

where $\theta > 0$ and $\delta > 0$. We set $n_\delta^{-1/2} := n^{-1/4+\delta/2} + n^{-2\delta}$, in which $n^{-2\delta}$ reflects the bias due to the microstructure noise, whereas $n^{-1/4+\delta/2}$ is the convergence rate of the pre-averaging estimator given by Christensen, Kinnebrock, and Podolskij (2010) in the low dimensional setting. $n_\delta^{-1/2}$ is the effective sample size as we will see below.

Our choice of $\delta > 0$ is not optimal. Nonetheless, it results in a simpler estimator without need of bias-correction, which also relies on fewer tuning parameters. More importantly, it guarantees a semi-definite covariance matrix estimate in any finite sample, a desirable property on which our following-up thresholding procedure relies.

The returns in each block are weighted by a piecewise continuously differentiable function g on $[0, 1]$, with a piecewise Lipschitz derivative g'. Moreover, $g(0) = g(1) = 0$, and $\int_0^1 g^2(s)ds > 0$. A simple example of g would be $g(s) = s \wedge (1 - s)$. We define $\psi_1 = \phi_1(0)$, $\psi_2 = \phi_2(0)$, where

$$\phi_1(s) = \int_s^1 g'(u)g'(u-s)du, \quad \phi_2(s) = \int_s^1 g(u)g(u-s)du.$$

For a sequence of vectors V, we define its weighted average return as $\bar{V}$, given by

$$\bar{V}_i = \sum_{j=1}^{k_n-1} g\left(\frac{j}{k_n}\right) \Delta_{i+j}^n V, \quad \text{for } i = 0, \dots, n - k_n + 1,$$

and $\Delta_i^n V = V_i - V_{i-1}$, for $i = 1, 2, \dots, n$, and $V_i \in \{X_{t_i}, Y_{t_i}, Z_{t_i}, X_{t_i}^\star, Y_{t_i}^\star, Z_{t_i}^\star, \varepsilon_i^x, \varepsilon_i^y\}$.

In what follows, we propose three estimators corresponding to different scenarios of the factor model. Each estimator uses the sample covariance matrix of these pre-averaged returns as an input, which leads to robustness to the microstructure noise. Intuitively, the effect of the noise is dominated by a strengthened return signal of each block.

1.3.1 Time-Series Regression (TSR)

When factors are known, we adopt a time-series regression-based approach using $\bar{Y}^\star$ and $\bar{X}^\star$. We stack the d- and r-dimensional processes Y and X into U, and their pre-average returns $\bar{Y}^\star$, $\bar{X}^\star$ into $\bar{U}^\star$, respectively:

$$U := (Y^\intercal, X^\intercal)^\intercal, \quad \bar{U}^\star := (\bar{Y}^{\star\intercal}, \bar{X}^{\star\intercal})^\intercal,$$

where U is a $(d + r)$-dimensional process and $\bar{U}^\star$ is a $(d + r) \times (n - k_n + 2)$ dimensional matrix. The quadratic covariation of U is given by

$$\Pi := \frac{1}{t}\int_0^t [dU_s, dU_s] ds = \frac{1}{t}\int_0^t \begin{pmatrix} \beta e_s \beta^\intercal + g_s & \beta e_s \\ e_s \beta^\intercal & e_s \end{pmatrix} ds := \begin{pmatrix} \Pi_{11} & \Pi_{12} \\ \Pi_{21} & \Pi_{22} \end{pmatrix},$$

which can be estimated by the sample covariance matrix of $\bar{U}^\star$:

$$\widehat{\Pi} = \frac{n}{n - k_n + 2} \frac{1}{\psi_2 k_n t} \sum_{i=0}^{n-k_n+1} \bar{U}_i^\star \bar{U}_i^{\star\intercal},$$

where $\bar{U}_i^\star$ is the $(i+1)$th column of $\bar{U}^\star$. We then construct estimators of each component of the covariance matrix as:

$$\widehat{\beta} = \widehat{\Pi}_{12}(\widehat{\Pi}_{22})^{-1}, \quad \widehat{\mathrm{E}} = \widehat{\Pi}_{22}, \quad \text{and} \quad \widehat{\Gamma} = \widehat{\Pi}_{11} - \widehat{\Pi}_{12}(\widehat{\Pi}_{22})^{-1}\widehat{\Pi}_{21}.$$

We apply thresholding to the covariance matrix estimates and obtain:

$$\widehat{\Gamma}^S = \left(\widehat{\Gamma}^S_{ij}\right), \quad \widehat{\Gamma}^S_{ij} = \begin{cases} \widehat{\Gamma}_{ij} & i = j, \\ s_{\lambda_{ij}}(\widehat{\Gamma}_{ij}) & i \neq j, \end{cases} . \tag{1.5}$$

A plug-in covariance matrix estimator is therefore given by:

$$\widehat{\Sigma}_{\mathrm{TSR}} = \widehat{\beta}\widehat{\mathrm{E}}\widehat{\beta}^\intercal + \widehat{\Gamma}^S.$$

We postpone the choice of the thresholding method $s(\cdot)$ and tuning parameters λ_{ij} to Section 1.4, where we provide a procedure to guarantee the positive semi-definiteness of $\widehat{\Sigma}_{\mathrm{TSR}}$.

We provide the convergence rates under several matrix norms for both the covariance matrix $\widehat{\Sigma}_{\mathrm{TSR}}$ and its inverse:

Theorem 1. *Suppose Assumptions 1 - 5 hold, and $n_\delta^{-1/2}\sqrt{\log d} = o(1)$. Then we have*

$$\begin{aligned} \left\|\widehat{\Sigma}_{\mathrm{TSR}} - \Sigma\right\|_{\mathrm{MAX}} &= O_p\left(n_\delta^{-1/2}\sqrt{\log d}\right), \\ \left\|\widehat{\Sigma}_{\mathrm{TSR}} - \Sigma\right\|_{\Sigma} &= O_p\left(d^{1/2}n_\delta^{-1}\log d + (n_\delta^{-1}\log d)^{(1-q)/2}m_d\right), \\ \|\widehat{\beta} - \beta\| &= O_p\left(n_\delta^{-1/2}\sqrt{\log d}\right). \end{aligned}$$

Moreover, if $\left(n_\delta^{-1}\log d\right)^{(1-q)/2} m_d = o(1)$, we have

$$\left\|\left(\widehat{\Sigma}_{\mathrm{TSR}}\right)^{-1} - \Sigma^{-1}\right\| = O_p\left(\left(n_\delta^{-1}\log d\right)^{(1-q)/2} m_d\right),$$

and $\lambda_{\min}(\widehat{\Sigma}_{\mathrm{TSR}}) \geqslant \frac{1}{2}\lambda_{\min}(\Gamma)$, *with probability approaching 1.*

Theorem 1 establishes the convergence rate, which depends on the degree of sparsity m_d, q, the dimension d, and the local window length parameter δ. Under $\|\cdot\|_{\mathrm{MAX}}$ norm, covariance matrix estimators with/without a factor model deliver the same rate, because even in a factor model, too many parameters for estimation remain from the low-rank component, which determines the low convergence rate under $\|\cdot\|_{\mathrm{MAX}}$ norm.

In terms of the inverse, when $d > n$, estimating Σ^{-1} becomes infeasible without a factor model, whereas the factor-based covariance matrix is invertible with high probability, and the inverse converges to the target under the operator norm. Because $\|\cdot\|_{\Sigma}$ norm depends on both Σ and Σ^{-1}, and the latter is more accurately estimated using a factor model, under $\|\cdot\|_{\Sigma}$ norm, using a factor model gives a better rate than the rate without it, which would be $d^{1/2}n_{\delta}^{-1/2}\sqrt{\log d}$.

More importantly, the minimum eigenvalue of the resulting covariance matrix is bounded away from 0 with high probability, so that the covariance matrix estimate is well-conditioned. This property is essential to warrant an economically feasible optimal portfolio using the estimated covariance matrix as the input.

1.3.2 Cross-Sectional Regression (CSR)

If we observe the factor loading matrix β, we propose a cross-sectional regression approach that recovers X at first. We start with a scenario in which the data are synchronous and noise-free, because the asymptotic property of such an estimator is not available in the literature, to the best of our knowledge. The estimator can be constructed as

$$\widehat{\Sigma}^{*}_{\mathrm{CSR}} = \beta\check{\mathrm{E}}^{*}\beta^{\intercal} + \check{\Gamma}^{*S},$$

where

$$\breve{X} = (\beta^\intercal\beta)^{-1}\beta^\intercal Y, \quad \breve{\mathrm{E}}^* = \frac{1}{t}\breve{X}\breve{X}^\intercal, \quad \breve{\Gamma}^* = \frac{1}{t}\left(Y - \beta\breve{X}\right)\left(Y - \beta\breve{X}\right)^\intercal,$$

and

$$\breve{\Gamma}^{*S} = \left(\breve{\Gamma}^{*S}_{ij}\right), \quad \breve{\Gamma}^{*S}_{ij} = \begin{cases} \breve{\Gamma}_{ij} & i = j, \\ s_{\lambda_{ij}}(\breve{\Gamma}_{ij}) & i \neq j, \end{cases}.$$

This estimator is similar in spirit to the covariance matrix estimator provided by the MSCI Barra; see Kahn, Brougham, and Green (1998). We analyze its properties as follows:

Theorem 2. *Suppose Assumptions 1 - 3, 5 hold, $n^{-1/2}\sqrt{\log d} = o(1)$. Then we have*

$$\begin{aligned}
\left\|\widehat{\Sigma}^*_{\mathrm{CSR}} - \Sigma\right\|_{\mathrm{MAX}} =& O_p\left(n^{-1/2}\sqrt{\log d} + d^{-1/2}m_d^{1/2}\right), \\
\left\|\widehat{\Sigma}^*_{\mathrm{CSR}} - \Sigma\right\|_{\Sigma} =& O_p\left(\left[n^{-1/2}d^{1/4}\sqrt{\log d} + d^{-1/4}m_d^{1/2}\right]^2 + \left[n^{-1/2}\sqrt{\log d} + d^{-1/2}m_d^{1/2}\right]^{1-q} m_d\right), \\
\left\|\breve{X} - X\right\| =& O_p\left(d^{-1/2}m_d^{1/2}\right).
\end{aligned}$$

Moreover, if $\left[n^{-1/2}\sqrt{\log d} + d^{-1/2}m_d^{1/2}\right]^{1-q} m_d = o(1)$, we have

$$\left\|\left(\widehat{\Sigma}^*_{\mathrm{CSR}}\right)^{-1} - \Sigma^{-1}\right\| = O_p\left(\left[n^{-1/2}\sqrt{\log d} + d^{-1/2}m_d^{1/2}\right]^{1-q} m_d\right),$$

and $\lambda_{\min}(\widehat{\Sigma}^*_{\mathrm{CSR}}) \geqslant \frac{1}{2}\lambda_{\min}(\Gamma)$, with probability approaching 1.

Theorem 2 shows the CSR estimator does not converge under $\|\cdot\|_{\mathrm{MAX}}$ when d is fixed, due to the second term $d^{-1/2}m_d^{1/2}$, unlike the TSR estimator. This finding is not surprising, because the cross-sectional regression exploits an increasing dimensionality to estimate X. The first term $n^{-1/2}\sqrt{\log d}$ is the same as that in the convergence rate of TSR in the absence of noise (see Fan, Furger, and Xiu (2016)), because both approaches estimate Γ based on a thresholded sample covariance matrix estimator. Comparing this convergence rate with that of the PCA estimator given by Aït-Sahalia and Xiu (2017), which is $n^{-1/2}\sqrt{\log d}+d^{-1/2}m_d$, is also interesting. The rate improvement in the CSR estimator comes from the second term and is due to the knowledge of β. Overall, the convergence rate of CSR depends on a striking trade-off between n and d.

In the general scenario where the noise plagues the data, we construct a pre-averaging-based covariance matrix estimator:

$$\widehat{\Sigma}_{\mathrm{CSR}} = \beta \check{\mathrm{E}} \beta^{\mathsf{T}} + \check{\Gamma}^{S},$$

where

$$\begin{aligned}
\check{X}^{\star} &= (\beta^{\mathsf{T}}\beta)^{-1}\beta^{\mathsf{T}}\bar{Y}^{\star}, \\
\check{\mathrm{E}} &= \frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n t}\check{X}^{\star}\check{X}^{\star\mathsf{T}}, \\
\check{\Gamma} &= \frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n t}\left(\bar{Y}^{\star}-\beta\check{X}^{\star}\right)\left(\bar{Y}^{\star}-\beta\check{X}^{\star}\right)^{\mathsf{T}},
\end{aligned}$$

and

$$\check{\Gamma}^{S} = \left(\check{\Gamma}^{S}_{ij}\right), \quad \check{\Gamma}^{S}_{ij} = \begin{cases} \check{\Gamma}_{ij} & i=j, \\ s_{\lambda_{ij}}(\check{\Gamma}_{ij}) & i \neq j, \end{cases}.$$

The next theorem gives the convergence rates

Theorem 3. *Suppose Assumptions 1 - 5 hold, and* $n_{\delta}^{-1/2}\sqrt{\log d}+d^{-1/2}m_d^{1/2}=o(1)$. *Then we have*

$$\begin{aligned}
\left\|\widehat{\Sigma}_{\mathrm{CSR}}-\Sigma\right\|_{\mathrm{MAX}} =& O_p\left(n_{\delta}^{-1/2}\sqrt{\log d}+d^{-1/2}m_d^{1/2}\right), \\
\left\|\widehat{\Sigma}_{\mathrm{CSR}}-\Sigma\right\|_{\Sigma} =& O_p\left(\left[n_{\delta}^{-1/2}\sqrt{\log d}d^{1/4}+d^{-1/2}m_d^{1/2}\right]^2+\left[n_{\delta}^{-1/2}\sqrt{\log d}+d^{-1/2}m_d^{1/2}\right]^{1-q}m_d\right), \\
\left\|\check{X}^{\star}-\bar{X}^{\star}\right\| =& O_p\left(d^{-1/2}m_d^{1/2}\right).
\end{aligned}$$

Moreover, if $\left[n_\delta^{-1/2}\sqrt{\log d} + d^{-1/2}m_d^{1/2}\right]^{1-q} m_d = o(1)$*, we have*

$$\left\|\left(\widehat{\Sigma}_{\mathrm{CSR}}\right)^{-1} - \Sigma^{-1}\right\| = O_p\left(\left[n_\delta^{-1/2}\sqrt{\log d} + d^{-1/2}m_d^{1/2}\right]^{1-q} m_d\right),$$

and $\lambda_{\min}(\widehat{\Sigma}_{\mathrm{CSR}}) \geqslant \frac{1}{2}\lambda_{\min}(\Gamma)$*, with probability approaching 1.*

Compared to the results of Theorem 2, the rates in Theorem 3 remain the same except that n is replaced by n_δ, which is the effective sample size when noise is present. The same intuition as in the no-noise case holds as well. Moreover, as discussed previously, the choice of $\delta > 0$ leads to a simpler estimator despite it being less efficient.

1.3.3 Principal Component Analysis (PCA)

Without prior knowledge of factors or their loadings, we apply PCA to the pre-averaged covariance matrix estimate based on $\bar{Y}^\star$:

$$\widetilde{\Sigma} = \frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n t}\sum_{i=0}^{n-k_n+1} \bar{Y}_i^\star \bar{Y}_i^{\star\intercal}.$$

Suppose $\widehat{\lambda}_1 > \widehat{\lambda}_2 > \ldots > \widehat{\lambda}_d$ are the simple eigenvalues of $\widetilde{\Sigma}$, and $\widehat{\xi}_1, \widehat{\xi}_2, \ldots, \widehat{\xi}_d$ are the corresponding eigenvectors. Then $\widetilde{\Sigma}$ can be decomposed as

$$\widetilde{\Sigma} = \sum_{j=1}^{\widehat{r}} \widehat{\lambda}_j \widehat{\xi}_j \widehat{\xi}_j^\intercal + \widetilde{\Gamma}, \tag{1.6}$$

where $\widehat{r}$ is an estimator of r to be introduced below. Similar to TSR, we apply thresholding on $\widetilde{\Gamma}$ and obtain

$$\widetilde{\Gamma}^S = \left(\widetilde{\Gamma}_{ij}^S\right), \quad \widetilde{\Gamma}_{ij}^S = \begin{cases} \widetilde{\Gamma}_{ij} & i = j, \\ s_{\lambda_{ij}}(\widetilde{\Gamma}_{ij}) & i \neq j, \end{cases},$$

and the resulting estimator of Σ is

$$\widehat{\Sigma}_{\mathrm{PCA}} = \sum_{j=1}^{\widehat{r}} \widehat{\lambda}_j \widehat{\xi}_j \widehat{\xi}_j^{\intercal} + \widetilde{\Gamma}^S. \tag{1.7}$$

This estimator is motivated by the POET strategy by Fan, Liao, and Mincheva (2013) for low-frequency data, and adapted from the PCA approach by Aït-Sahalia and Xiu (2017) for high-frequency data. The idea behind this strategy is that the eigenvectors corresponding to the first r eigenvalues of the sample covariance matrix can be used to construct proxies of β, so that the low-rank component of Σ can be approximated by the first term on the right-hand side of (1.6). The second term then approximates the sparse component of Σ, which leads to the construction of the estimator given by (1.7).

This estimator can also be constructed from a least-squares point of view, which seeks F and G such that

$$(F, G) = \arg \min_{F \in \mathcal{M}_{d \times \widehat{r}}, G \in \mathcal{M}_{\widehat{r} \times n}} \left\| \bar{Y}^{\star} - FG \right\|_{\mathrm{F}}^2,$$

subject to the normalization:

$$d^{-1} F^{\intercal} F = \mathbb{I}_{\widehat{r}}, \quad GG^{\intercal} \text{ is an } \widehat{r} \times \widehat{r} \text{ diagonal matrix.}$$

Bai and Ng (2002) and Bai (2003) propose this estimator to estimate factors and their loadings in a factor model for low-frequency data.

Once we have estimates of factors and loadings, we can obtain the same $\widetilde{\Gamma}$ and $\widetilde{\Gamma}^S$ as above by:

$$\widetilde{\Gamma} = \frac{n}{n - k_n + 2} \frac{1}{\psi_2 k_n t} \left(\bar{Y}^{\star} - FG \right) \left(\bar{Y}^{\star} - FG \right)^{\intercal}, \quad \widetilde{\Gamma}^S = \left(\widetilde{\Gamma}_{ij}^S \right), \quad \widetilde{\Gamma}_{ij}^S = \begin{cases} \widetilde{\Gamma}_{ij} & i = j, \\ s_{\lambda_{ij}}(\widetilde{\Gamma}_{ij}) & i \neq j, \end{cases},$$

with which we can construct

$$\widehat{\Sigma}_{\text{PCA}} = t^{-1}FGG^{\intercal}F^{\intercal} + \widetilde{\Gamma}^{S}. \tag{1.8}$$

Although (1.7) is easier to implement, this equivalent form of $\widehat{\Sigma}_{\text{PCA}}$ is useful in the proof and provides estimates of factors and their loadings (up to some rotation).

To determine the number of factors r, we propose the following estimator using a penalty function:

$$\widehat{r} = \arg \min_{1 \leqslant j \leqslant r_{\max}} \left(d^{-1}\lambda_j(\widetilde{\Sigma}) + j \times f(n,d) \right) - 1,$$

where $r_{\max}$ is some upper bound. This estimator is similar to that of Aït-Sahalia and Xiu (2017), which shares the spirit with Bai and Ng (2002). The penalty function $f(n,d)$ satisfies two criteria. On the one hand, the penalty is dominated by the signal, i.e., the value of $d^{-1}\lambda_j(\Sigma)$, for $1 \leqslant j \leqslant r$. Because $d^{-1}\lambda_r(\Sigma)$ is $O_P(1)$ as d increases, we select a penalty that shrinks to 0. On the other hand, we require the penalty to dominate the estimation error as well as $d^{-1}\lambda_j(\Sigma)$ for $r+1 \leqslant j \leqslant d$ to avoid overshooting. The choice of $r_{\max}$ does not play any role in theory, yet it warrants an economically meaningful estimate of $\widehat{r}$ in a finite sample or in practice.

Theorem 4. *Suppose Assumptions 1 - 5 hold. Also,* $n_{\delta}^{-1/2}\sqrt{\log d} + d^{-1/2}m_d = o(1)$, $f(n,d) \to 0$, *and* $f(n,d)\left(n_{\delta}^{-1/2}\sqrt{\log d} + d^{-1}m_d\right)^{-1} \to \infty$. *Then we have*

$$\begin{aligned}\left\|\widehat{\Sigma}_{\text{PCA}} - \Sigma\right\|_{\text{MAX}} &= O_P\left(n_{\delta}^{-1/2}\sqrt{\log d} + d^{-1/2}m_d\right), \\ \left\|\widehat{\Sigma}_{\text{PCA}} - \Sigma\right\|_{\Sigma} &= O_P\left(d^{1/2}n_{\delta}^{-1}\log d + d^{-1/2}m_d^2 + m_d\left(n_{\delta}^{-1/2}\sqrt{\log d} + d^{-1/2}m_d\right)^{1-q}\right).\end{aligned}$$

Also, there exists a $r \times r$ matrix H, such that with probability approaching 1, H is invertible,

$\|HH^\intercal - \mathbb{I}_r\| = o_p(1)$*, and*

$$\begin{aligned}\|F - \beta H\|_{\mathrm{MAX}} &= O_p\left(n_\delta^{-1/2}\sqrt{\log d} + d^{-1/2}m_d\right),\\ \left\|G - H^{-1}X\right\| &= O_p\left(n_\delta^{-1/2}\sqrt{\log d} + d^{-1/2}m_d\right).\end{aligned}$$

If, in addition, $n_\delta^{-1/2}\sqrt{\log d} = o(1)$*, then* $\lambda_{\min}(\widehat{\Sigma}_{\mathrm{PCA}})$ *is bounded away from 0 with probability approaching 1, and*

$$\left\|\widehat{\Sigma}_{\mathrm{PCA}}^{-1} - \Sigma^{-1}\right\| = O_p\left(m_d^3\left(n_\delta^{-1/2}\sqrt{\log d} + d^{-1/2}m_d\right)^{1-q}\right).$$

Due to the fundamental indeterminacy of a factor model, we only identify the latent factors and their loadings up to some invertible matrix H. That said, the covariance matrix estimator itself is invariant to H.

1.3.4 A Comparison of the Three Estimators

So far, we have obtained the convergence rates of all scenarios of factor models for the covariance matrix and its inverse, under a variety of norms.

We observe the usual tradeoff between efficiency and robustness in all scenarios. In terms of efficiency, TSR dominates CSR, which in turn dominates PCA. Nonetheless, PCA is more robust to model misspecification in that its construction utilizes the least amount of prior information. Interestingly, the loss of efficiency diminishes as the dimension of assets increases, thanks to the blessings of dimensionality. In light of this tradeoff, we resort to simulations in Section 1.5 for further comparison of the finite sample performance of these estimators, and to empirical data in Section 1.6 for evaluation of their relevance in practice.

1.4 Practical Considerations

1.4.1 Choice of Tuning Parameters

As discussed in Christensen, Kinnebrock, and Podolskij (2010), the pre-averaging estimator we adopt is consistent in the low-dimensional case if $0 < \delta < 0.5$, but its CLT requires $0.1 < \delta < 0.5$ so that the asymptotic bias due to noise is negligible compared to the asymptotic variance.

The two terms in $n_\delta^{-1/2}$ exactly characterizes the bias-variance trade-off in our setting. Similar to Kim, Wang, and Zou (2016), the $n^{-2\delta}$ term is due to the bias of the microstructure noise, whereas $n^{-1/4+\delta/2}$ is due to the variance of the estimator. We thereby select $\delta = 0.1$ to balance the bias and variance.

The tuning parameter k_n is determined by θ, once δ is given. With a large number of observations, the estimates are not sensitive to the choice of k_n as long as d is moderately large. In simulations and empirical studies, we adopt a range of θs, and thus k_ns, all of which lead to similar estimates that do not change our interpretations.

1.4.2 Choice of r and $f(n, d)$

A sensible choice of the penalty function could be

$$f(n,d) = \mu \left(n_\delta^{-1/2} \sqrt{\log d} + d^{-1} m_d \right)^\kappa \cdot \text{median}(\{\hat{\lambda}_1, ..., \hat{\lambda}_d\}), \tag{1.9}$$

for some tuning parameters μ and κ. One might also use the perturbed eigenvalue ratio estimator in Pelger (2015a) to determine r, which gives almost the same result in simulations. The latter requires one less tuning parameter, but its proof is more involved. Alternatively, as argued by Aït-Sahalia and Xiu (2017), we can simply regard r as a tuning parameter from the practical point of view. In fact, the performance of the estimator is not sensitive to $\widehat{r}$, as long as $\widehat{r}$ is greater than or equal to r, yet is not too large, as shown from our simulation results. We conjecture that the same convergence rate holds for our estimators as long as $\widehat{r} \geqslant r$ and $\widehat{r}$ is finite, which is indeed the case for parameter estimation in the interactive effect models; see, e.g., Moon and Weidner (2015). The proof likely involves the random matrix theory that is not available for continuous martingales. We leave this investigation for future work. In our empirical studies, we find that as soon as r is greater than 3 but not as large as 20, the comparison results remain the same qualitatively and the interpretations are identical.

1.4.3 Choice of the Thresholding Methods

We compare two types of thresholding methods on the residual covariance matrix, e.g., $\widehat{\Gamma}$ constructed in TSR.[4] The same applies to $\check{\Gamma}$ (CSR) and $\widetilde{\Gamma}$ (PCA).

The first one is the location-based thresholding utilizing domain knowledge (denoted as location thresholding), as in Fan, Furger, and Xiu (2016). This approach preserves positive semi-definiteness in a finite sample and is computationally efficient, because neither tuning nor optimization is involved. Specifically, we first sort the residual covariance matrix $\widehat{\Gamma}$ into blocks by assets' industrial classifications (sector, industry group, industry, or sub-industry), and then apply a block-diagonal mask to this residual covariance matrix. The thresholding function can be written explicitly as:

$$s^{loc}_{\lambda_{ij}}(z) = z\mathbf{1}(\lambda_{ij} = 1), \quad \text{where } \lambda_{ij} = 1, \text{ if and only if } (i,j) \in \text{ the same block.}$$

For each block, we have a positive semi-definite sub-matrix because $\widehat{\Gamma}$ is positive semi-definite by construction, so that stacking these blocks on the diagonal produces a positive semi-definite $\widehat{\Gamma}^S$.

The second class of methods we consider employ a threshold based on the sample corre-

[4] Alternatively, one can adopt the adaptive thresholding method by Cai and Liu (2011). In our simulations, this approach does not perform as well as the location-based and correlation-based thresholding methods we consider.

lation matrix. Specifically, we set

$$\lambda_{ij} = \tau\sqrt{\widehat{\Gamma}_{ii}\widehat{\Gamma}_{jj}},$$

where τ is some constant to be determined. With this threshold, we then apply Hard, Soft, SCAD, and AL thresholding methods with $s_{\lambda_{ij}}(\cdot)$s given by Section 1.2.3, respectively, in the construction of (1.5). These methods do not always guarantee a positive semi-definite $\widehat{\Gamma}^S$ in a finite sample. Also, when τ is small, $\widehat{\Gamma}^S$ might not be sufficiently sparse.

To fix these issues, we find an appropriate τ via a grid search algorithm, following Fan, Liao, and Mincheva (2013). We start from a small value of τ, and gradually increase it until a positive semi-definite $\widehat{\Gamma}^S$ is obtained and the degree of sparsity is below a certain threshold. As τ increases, the degree of sparsity decreases, and the solution shrinks toward the diagonal of $\widehat{\Gamma}$, which is positive semi-definite. Thus, our grid search is guaranteed to produce a solution. In other words, this algorithm yields a positive semi-definite estimate in a finite sample. Note the grid search for τ is easier here than that of Fan, Liao, and Mincheva (2013), because τ is bounded between 0 and 1. In practice, a natural choice of the desired degree of sparsity can be obtained using that of the location-based thresholding, which is computationally less expensive than the cross-validation method.

1.5 Monte Carlo Simulations

In this section, we investigate the finite sample performance of the pre-averaging estimator and compare it with the subsampling method discussed in Fan, Furger, and Xiu (2016) and Aït-Sahalia and Xiu (2017). The latter estimators are built upon the realized covariance estimators using subsampled returns. We simulate 1,000 paths from a continuous-time r-factor model of d assets specified as

$$dY_{i,t} = \sum_{j=1}^{r} \beta_{i,j} dX_{j,t} + dZ_{i,t}, \quad dX_{j,t} = b_j dt + \sigma_{j,t} dW_{j,t}, \quad dZ_{i,t} = \gamma_i^{\intercal} dB_t,$$

where $W_{j,t}$ is a standard Brownian motion and B_t is a d-dimensional Brownian motion, for $i = 1, 2, \ldots, d$, and $j = 1, 2, \ldots, r$. They are mutually independent. X_j is the jth factor, and we set X_1 as the market factor, with the associated loadings being positive. The covariance matrix of Z, Γ, is a block-diagonal matrix with $\Gamma_{i,l} = \gamma_i^\intercal \gamma_l$. We also allow for time-varying $\sigma_{j,t}^2$ which evolves as

$$d\sigma_{j,t}^2 = \kappa_j(\theta_j - \sigma_{j,t}^2)dt + \eta_j \sigma_{j,t} d\widetilde{B}_{j,t}, \quad j = 1, 2, \ldots, r,$$

where $\widetilde{B}_j$ is a standard Brownian motion with $\mathbb{E}[dB_{j,t}d\widetilde{B}_{j,t}] = \rho_j dt$. We choose $d = 500$ and $r = 3$. We fix t at 21 trading days, i.e., $t = 1/12$. In addition, $\kappa = (3, 4, 5)$, $\theta = (0.09, 0.04, 0.06)$, $\eta = (0.3, 0.4, 0.3)$, $\rho = (-0.6, -0.4, -0.250)$, and $b = (0.05, 0.03, 0.02)$. As for the factor loadings, we sample $\beta_1 \sim \mathcal{U}[0.25, 1.75]$, and $\beta_2, \beta_3 \sim \mathcal{N}(0, 0.5^2)$. The diagonal elements of Γ are sampled independently from $\mathcal{U}[0.1, 0.2]$, with constant within-block correlations sampled from $\mathcal{U}[0.1, 0.4]$ for each block. To generate blocks with random sizes, we fix the largest block size at 35, and randomly generate the sizes of the blocks from a uniform distribution between 10 and 35, such that the total size of all blocks is d. βs and block sizes are randomly generated but fixed across Monte Carlo repetitions.

We simulate the noise in log prices for each asset as an MA(1) process, i.e., $\epsilon^y_{i,t^i_j} = \xi_{i,t^i_j} - 0.5\xi_{i,t^i_{j-1}}$, where ξ_{i,t^i_j} is an i.i.d. normal noise with mean 0 and variance 0.001^2. To mimic the asynchronicity, we censor the data using Poisson sampling, where the number of observations for each stock is sampled from a truncated log-normal distribution. The log-normal distribution $\log\mathcal{N}(\mu, \sigma^2)$ has parameters $\mu = \log(2500)$ and $\sigma = 0.8$, and the lower and upper truncation boundaries are 1,000 and 23,400, respectively, which matches the empirical data on S&P 500 index constituents.

For pre-averaging estimators, we compare a range of local window lengths by varying θ in (1.4). For subsampling estimators, we compare a range of subsampling frequencies from every 5 minutes to every 65 minutes, denoted as Δ_n in seconds. We also add the benchmark estimates using noiseless returns sampled at a 5-minute frequency without asynchronicity for comparison. In addition, we consider a “mixed” approach by applying the pre-averaging method to the subsampled data, in order to check the marginal effect of the refresh-time versus subsampling. Because the simulated Γ is a block-diagonal matrix, we regard the location-based thresholding method as the benchmark, and compare its performance with other thresholding methods in Section 1.4.3. We present the simulation results in Tables 1.1, 1.2, and 1.3.

Estimator		Pre-Averaging				Subsampling				Noiseless	Mixed
Tuning		θ	θ	θ	θ	Δ_n	Δ_n	Δ_n	Δ_n	Δ_n^*	θ, Δ_n
Parameters		0.04	0.06	0.08	0.10	300	900	$1,800$	$3,900$	300	$0.08, 900$
		Location Thresholding									
$\\|\widehat{\Sigma} - \Sigma\\|_{\text{MAX}}$	TSR	0.06	0.06	0.06	0.06	0.12	0.08	0.09	0.13	0.03	0.09
	CSR	0.05	0.04	0.04	0.04	0.11	0.07	0.07	0.09	0.03	0.06
	PCA	0.06	0.06	0.06	0.06	0.12	0.08	0.09	0.17	0.04	0.09
$\\|\beta\widehat{\text{E}}\beta^\intercal - \beta \text{E} \beta^\intercal\\|_{\text{MAX}}$	TSR	0.03	0.04	0.05	0.05	0.04	0.06	0.08	0.11	0.03	0.07
	CSR	0.02	0.02	0.03	0.03	0.02	0.03	0.04	0.06	0.02	0.04
	PCA	0.04	0.04	0.05	0.06	0.04	0.06	0.08	0.19	0.04	0.08
$\\|\widehat{\Gamma} - \Gamma\\|_{\text{MAX}}$	TSR	0.05	0.04	0.04	0.04	0.12	0.06	0.06	0.07	0.02	0.05
	CSR	0.05	0.04	0.03	0.03	0.11	0.06	0.06	0.07	0.02	0.05
	PCA	0.05	0.04	0.03	0.03	0.11	0.06	0.06	0.17	0.02	0.05
$\\|\widehat{\Sigma} - \Sigma\\|_{\Sigma}$	TSR	0.31	0.24	0.24	0.26	0.80	0.40	0.43	0.62	0.14	0.39
	CSR	0.30	0.23	0.22	0.23	0.79	0.37	0.37	0.48	0.13	0.33
	PCA	0.31	0.25	0.24	0.26	0.79	0.40	0.43	0.93	0.15	0.40
$\\|(\widehat{\Sigma})^{-1} - \Sigma^{-1}\\|$	TSR	5.70	4.62	4.42	5.92	9.43	6.50	10.45	30.16	3.57	9.82
	CSR	5.71	4.66	4.42	5.84	9.42	6.52	10.26	28.90	3.54	9.62
	PCA	5.70	4.63	4.44	5.93	9.41	6.50	10.49	28.68	3.56	9.84
$\\|(\widehat{\Gamma})^{-1} - \Gamma^{-1}\\|$	TSR	5.72	4.64	4.45	5.96	9.45	6.52	10.54	30.59	3.59	9.90
	CSR	5.74	4.68	4.45	5.89	9.44	6.54	10.38	29.33	3.57	9.71
	PCA	5.72	4.65	4.48	5.98	9.44	6.52	10.59	28.89	3.58	9.95
$\widehat{r}$	PCA	3.00	3.00	3.00	3.00	3.00	3.00	3.00	1.71	3.00	3.04

Table 1.1: Pre-Averaging vs. Subsampling Using Location Thresholding

Note: In this table, we compare the performance of pre-averaging estimators with subsampling estimators using location thresholding method under a variety of matrix norms. The tuning parameter θ is for the pre-averaging local window length ($k_n \approx \theta n^{1/2+\delta}$), whereas the Δ_n is the subsampling frequency in seconds. The “Noiseless” column provides the estimates using clean and synchronous data with a sampling frequency at $\Delta_n^* = 300$ (5-minute), and the “Mixed” column provides the estimates using the pre-averaging approach on the subsampled data ($\theta = 0.08, \Delta_n = 900$). Because the low-rank part $\beta\widehat{\text{E}}\beta^\intercal$ is identical across thresholding methods, we only report it in the upper panel. We also report the average estimated number of factors for the PCA approach. The number of Monte Carlo repetitions is 1,000.

First, we find pre-averaging estimators achieve smaller errors compared to subsampling estimators, and are closer to the noiseless benchmark, for almost all model specifications and almost all criteria. The pre-averaging method achieves a slightly better estimate of the low-rank part $\beta \mathrm{E} \beta^{\intercal}$, and a better residual part Γ, especially under the operator norm for Γ^{-1}, which in turn gives a better estimate of the precision matrix Σ^{-1}. Throughout, the pre-averaging estimators are robust to a wide range of tuning parameters θ. The sweet spot appears to be $\theta = 0.08$, whereas the optimal frequency for the subsampling method seems to be achieved near $\Delta_n = 900$.

Second, for comparison across various thresholding methods, we find in almost all scenarios that the Location thresholding achieves the best result, followed by Soft, AL, and SCAD thresholding, whereas Hard thresholding appears to be the worst. The differences are smaller in terms of $\|\widehat{\Sigma} - \Sigma\|_{\mathrm{MAX}}$ and $\|\widehat{\Gamma} - \Gamma\|_{\mathrm{MAX}}$, because the largest entry-wise errors are either achieved along the diagonal or on entries off the diagonal with large magnitudes, which are least affected by various thresholding methods. Nevertheless, the differences are most salient in terms of $\|(\widehat{\Sigma})^{-1} - \Sigma^{-1}\|$ and $\|(\widehat{\Gamma})^{-1} - \Gamma^{-1}\|$. The former is arguably the most important metric in this table, because it dictates the performance of the portfolio allocation in the empirical exercise.

Third, the comparison between the “mixed” approach versus subsampling depends on a bias-variance tradeoff. If the bias due to noise is large, the mixed approach outperforms; if the noise effect becomes negligible for a sufficiently low sampling frequency, the subsampling approach can outperform because its convergence rate is faster. By contrast, the “mixed” approach is always dominated by the pre-averaging based on the refresh time scheme, because the latter reserves more data and the two estimators are equally efficient given the same amount of data.

Finally, to demonstrate the impact of the selected number of factors on the PCA, we present in Table 1.4 the errors with a range of $\widehat{r}$ pre-set instead of being estimated. Due to

Estimator		Pre-Averaging				Subsampling				Noiseless	Mixed
Tuning		θ	θ	θ	θ	Δ_n	Δ_n	Δ_n	Δ_n	Δ_n^*	θ, Δ_n
Parameters		0.04	0.06	0.08	0.10	300	900	$1,800$	$3,900$	300	$0.08, 900$
		Hard Thresholding									
$\Vert\widehat{\Sigma}-\Sigma\Vert_{\mathrm{MAX}}$	TSR	0.08	0.08	0.09	0.09	0.12	0.09	0.11	0.13	0.08	0.10
	CSR	0.08	0.08	0.08	0.08	0.11	0.08	0.08	0.09	0.07	0.08
	PCA	0.08	0.08	0.08	0.08	0.12	0.09	0.10	0.17	0.08	0.10
$\Vert\widehat{\Gamma}-\Gamma\Vert_{\mathrm{MAX}}$	TSR	0.08	0.08	0.08	0.08	0.12	0.08	0.08	0.08	0.08	0.08
	CSR	0.08	0.08	0.08	0.08	0.11	0.08	0.08	0.08	0.08	0.08
	PCA	0.08	0.08	0.08	0.08	0.11	0.08	0.08	0.17	0.08	0.08
$\Vert\widehat{\Sigma}-\Sigma\Vert_{\Sigma}$	TSR	0.60	0.50	0.44	0.43	1.09	0.61	0.53	0.58	0.37	0.49
	CSR	0.57	0.46	0.41	0.39	1.04	0.56	0.46	0.41	0.35	0.42
	PCA	0.57	0.47	0.42	0.41	1.04	0.58	0.51	0.67	0.35	0.47
$\Vert(\widehat{\Sigma})^{-1}-\Sigma^{-1}\Vert$	TSR	8.40	7.65	7.20	7.02	10.65	8.52	7.79	7.56	6.58	7.44
	CSR	8.25	7.50	7.05	6.90	10.53	8.38	7.67	7.52	6.67	7.33
	PCA	8.23	7.46	7.13	7.22	10.52	8.35	7.61	8.45	7.43	7.38
$\Vert(\widehat{\Gamma})^{-1}-\Gamma^{-1}\Vert$	TSR	8.42	7.67	7.22	7.04	10.67	8.54	7.81	7.59	6.60	7.46
	CSR	8.27	7.52	7.07	6.92	10.55	8.41	7.70	7.55	6.69	7.36
	PCA	8.25	7.49	7.03	6.88	10.54	8.37	7.62	8.40	6.70	7.30
		Soft Thresholding									
$\Vert\widehat{\Sigma}-\Sigma\Vert_{\mathrm{MAX}}$	TSR	0.06	0.06	0.07	0.07	0.12	0.09	0.10	0.13	0.05	0.09
	CSR	0.06	0.05	0.05	0.06	0.11	0.07	0.07	0.09	0.04	0.07
	PCA	0.06	0.06	0.06	0.07	0.12	0.09	0.10	0.15	0.05	0.09
$\Vert\widehat{\Gamma}-\Gamma\Vert_{\mathrm{MAX}}$	TSR	0.05	0.05	0.05	0.05	0.12	0.06	0.06	0.08	0.04	0.06
	CSR	0.05	0.05	0.05	0.05	0.11	0.06	0.06	0.08	0.05	0.06
	PCA	0.05	0.05	0.05	0.05	0.11	0.06	0.06	0.17	0.05	0.06
$\Vert\widehat{\Sigma}-\Sigma\Vert_{\Sigma}$	TSR	0.51	0.40	0.36	0.35	1.05	0.55	0.50	0.60	0.25	0.45
	CSR	0.49	0.38	0.34	0.32	1.01	0.51	0.44	0.45	0.25	0.40
	PCA	0.50	0.40	0.35	0.35	1.02	0.53	0.49	0.74	0.25	0.45
$\Vert(\widehat{\Sigma})^{-1}-\Sigma^{-1}\Vert$	TSR	7.28	6.26	5.74	5.61	10.39	7.76	7.14	7.79	5.43	6.62
	CSR	7.27	6.26	5.76	5.64	10.36	7.76	7.18	7.99	5.54	6.67
	PCA	7.25	7.20	7.76	8.11	10.35	7.73	8.12	12.54	8.24	8.42
$\Vert(\widehat{\Gamma})^{-1}-\Gamma^{-1}\Vert$	TSR	7.31	6.28	5.76	5.63	10.42	7.78	7.16	7.82	5.68	6.64
	CSR	7.29	6.28	5.78	5.67	10.39	7.78	7.20	8.03	5.76	6.69
	PCA	7.28	6.26	5.75	5.63	10.38	7.75	7.17	12.05	5.77	6.68

Table 1.2: Pre-Averaging vs. Subsampling Using Hard and Soft Thresholding

Note: This table is a continuation of Table 1.1, where we report the simulation results using different thresholding methods. All other settings remain the same.

Estimator		Pre-Averaging				Subsampling				Noiseless	Mixed
Tuning		θ	θ	θ	θ	Δ_n	Δ_n	Δ_n	Δ_n	Δ_n^*	θ, Δ_n
Parameters		0.04	0.06	0.08	0.10	300	900	$1,800$	$3,900$	300	$0.08, 900$
		SCAD Thresholding									
$\lVert\widehat{\Sigma} - \Sigma\rVert_{\text{MAX}}$	TSR	0.06	0.06	0.07	0.07	0.12	0.09	0.10	0.13	0.05	0.09
	CSR	0.06	0.05	0.05	0.06	0.11	0.07	0.07	0.09	0.04	0.07
	PCA	0.06	0.06	0.06	0.07	0.12	0.09	0.10	0.15	0.05	0.09
$\lVert\widehat{\Gamma} - \Gamma\rVert_{\text{MAX}}$	TSR	0.05	0.05	0.05	0.05	0.12	0.06	0.06	0.08	0.04	0.06
	CSR	0.05	0.05	0.05	0.05	0.11	0.06	0.06	0.08	0.05	0.06
	PCA	0.05	0.05	0.05	0.05	0.11	0.06	0.06	0.17	0.05	0.06
$\lVert\widehat{\Sigma} - \Sigma\rVert_{\Sigma}$	TSR	0.51	0.40	0.36	0.36	1.05	0.55	0.52	0.61	0.26	0.47
	CSR	0.49	0.39	0.34	0.33	1.01	0.52	0.45	0.45	0.25	0.41
	PCA	0.50	0.40	0.36	0.35	1.02	0.54	0.51	0.74	0.26	0.46
$\lVert(\widehat{\Sigma})^{-1} - \Sigma^{-1}\rVert$	TSR	7.42	6.51	6.10	6.35	10.39	7.99	12.21	9.69	5.44	11.96
	CSR	7.37	6.53	6.14	6.22	10.36	7.94	10.78	9.96	5.55	10.75
	PCA	7.35	7.26	7.92	8.43	10.35	7.91	11.04	12.96	8.39	11.40
$\lVert(\widehat{\Gamma})^{-1} - \Gamma^{-1}\rVert$	TSR	7.44	6.53	6.12	6.38	10.42	8.01	12.36	9.78	5.68	12.10
	CSR	7.39	6.55	6.17	6.26	10.39	7.97	10.91	10.05	5.76	10.88
	PCA	7.37	6.53	6.14	6.25	10.38	7.94	10.68	12.60	5.77	10.97
		AL Thresholding									
$\lVert\widehat{\Sigma} - \Sigma\rVert_{\text{MAX}}$	TSR	0.06	0.06	0.06	0.07	0.12	0.09	0.10	0.13	0.05	0.09
	CSR	0.05	0.05	0.05	0.05	0.11	0.07	0.07	0.09	0.04	0.07
	PCA	0.06	0.06	0.06	0.07	0.12	0.09	0.10	0.16	0.05	0.09
$\lVert\widehat{\Gamma} - \Gamma\rVert_{\text{MAX}}$	TSR	0.05	0.04	0.04	0.05	0.12	0.06	0.06	0.08	0.04	0.06
	CSR	0.05	0.05	0.05	0.05	0.11	0.06	0.06	0.08	0.04	0.06
	PCA	0.05	0.05	0.05	0.05	0.11	0.06	0.06	0.17	0.04	0.06
$\lVert\widehat{\Sigma} - \Sigma\rVert_{\Sigma}$	TSR	0.48	0.38	0.35	0.36	1.03	0.55	0.55	0.67	0.23	0.50
	CSR	0.47	0.37	0.34	0.34	1.00	0.52	0.50	0.53	0.23	0.45
	PCA	0.48	0.38	0.35	0.36	1.00	0.54	0.55	0.75	0.24	0.50
$\lVert(\widehat{\Sigma})^{-1} - \Sigma^{-1}\rVert$	TSR	6.70	5.65	5.29	6.73	10.26	7.46	16.46	41.68	5.34	14.24
	CSR	6.76	5.72	5.37	6.67	10.30	7.53	17.40	40.75	5.46	15.06
	PCA	6.94	7.94	8.82	9.55	10.29	7.71	18.10	25.70	9.12	16.06
$\lVert(\widehat{\Gamma})^{-1} - \Gamma^{-1}\rVert$	TSR	6.72	5.66	5.32	6.79	10.29	7.48	16.70	42.81	5.59	14.42
	CSR	6.78	5.75	5.43	6.78	10.32	7.56	17.72	41.92	5.71	15.32
	PCA	6.77	5.73	5.44	6.89	10.32	7.54	17.95	25.58	5.71	15.87

Table 1.3: Pre-Averaging vs. Subsampling Using SCAD, and AL Thresholding

Note: This table is a continuation of Table 1.1, where we report the simulation results using different thresholding methods. All other settings remain the same.

space constraints, we only report the results with $\theta = 0.08$ for the pre-averaging, $\Delta_n = 900$ for the subsampling method, and the benchmark no-noise and synchronous case with $\Delta_n^* = 300$. Location thresholding is used in all cases. For the estimation of Σ and Σ^{-1}, we find when $\widehat{r} < r$, the performance is much worse in every metric than the case with $\widehat{r} = r$, whereas in the case of $\widehat{r} > r$, the performance is only slightly worse, in particular when $\widehat{r}$ is within a reasonable range (smaller than 20). For the purpose of covariance matrix estimation, this result justifies treating r as a tuning parameter without estimating it. With respect to estimating the low-rank and sparse components of Σ, using an incorrect number of factors is harmful.

		Location Thresholding											
$\widehat{r}$		1	2	3	4	5	6	8	10	15	20	30	50
$\Vert\widehat{\Sigma} - \Sigma\Vert_{\text{MAX}}$	Pre-Averaging	0.20	0.14	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.06
	Subsampling	0.20	0.15	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08
	Noiseless	0.20	0.14	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04
$\Vert\beta\widehat{\mathrm{E}}\beta^{\intercal} - \beta\mathrm{E}\beta^{\intercal}\Vert_{\text{MAX}}$	Pre-Averaging	0.24	0.16	0.05	0.08	0.09	0.10	0.10	0.10	0.11	0.11	0.12	0.14
	Subsampling	0.24	0.16	0.06	0.09	0.10	0.10	0.11	0.11	0.11	0.12	0.13	0.16
	Noiseless	0.24	0.16	0.04	0.08	0.08	0.09	0.09	0.09	0.09	0.10	0.10	0.12
$\Vert\widehat{\Gamma} - \Gamma\Vert_{\text{MAX}}$	Pre-Averaging	0.25	0.17	0.03	0.07	0.07	0.08	0.09	0.09	0.09	0.09	0.09	0.11
	Subsampling	0.27	0.19	0.06	0.07	0.08	0.08	0.09	0.09	0.09	0.09	0.09	0.10
	Noiseless	0.24	0.16	0.02	0.07	0.07	0.08	0.09	0.09	0.09	0.09	0.09	0.11
$\Vert\widehat{\Sigma} - \Sigma\Vert_{\Sigma}$	Pre-Averaging	1.11	0.66	0.24	0.25	0.26	0.27	0.29	0.31	0.36	0.39	0.55	0.75
	Subsampling	1.19	0.76	0.40	0.41	0.43	0.44	0.46	0.48	0.54	0.59	0.78	1.03
	Noiseless	1.07	0.62	0.15	0.15	0.16	0.16	0.18	0.19	0.21	0.23	0.34	0.49
$\Vert(\widehat{\Sigma})^{-1} - \Sigma^{-1}\Vert \times 10^{-2}$	Pre-Averaging	0.10	0.09	0.04	0.04	0.04	0.05	0.05	0.05	0.05	0.07	0.07	0.09
	Subsampling	0.10	0.09	0.07	0.06	0.06	0.06	0.06	0.06	0.06	0.07	0.08	0.09
	Noiseless	0.10	0.09	0.04	0.04	0.04	0.04	0.04	0.04	0.05	0.06	0.06	0.07
$\Vert(\widehat{\Gamma})^{-1} - \Gamma^{-1}\Vert \times 10^{-2}$	Pre-Averaging	0.10	0.09	0.04	0.08	0.13	0.21	0.36	0.50	0.87	1.35	1.69	2.06
	Subsampling	0.10	0.09	0.07	0.08	0.11	0.17	0.26	0.35	0.59	0.84	1.03	1.26
	Noiseless	0.10	0.09	0.04	0.14	0.23	0.42	0.79	1.09	2.03	3.23	4.36	5.01

Table 1.4: Impact of the Selected Number of Factors on the PCA Method

Note: We present the errors of the PCA approach for a variety of norms, using different number of factors $\widehat{r}$, with the true value of r being equal to 3. For tuning parameters, we select $\theta = 0.08$ for pre-averaging estimators, $\Delta_n = 900$ for subsampling estimators, and the benchmark no-noise and synchronous case with $\Delta_n^* = 300$. Location thresholding is used in all cases. The number of Monte Carlo repetitions is 1,000.

1.6 Empirical Applications

1.6.1 Data

We collect data from the TAQ database intraday transaction prices of the constituents of Dow Jones 30 index, S&P 100 index, and S&P 500 index from January 2004 to December 2013. The indices have 42, 152, and 735 stocks, respectively, during this sampling period.

We select stocks that are members of these indices on the last day of each month, and exclude those among them that have no trading activities on one or more trading days of this month, as well as the bottom 5% stocks in terms of the number of observations for liquidity concerns. To clean the data, we adopt the procedure detailed in Da and Xiu (2017), which only relies on the condition codes from the exchanges and the range of NBBO quotes to identify outliers. We exclude overnight returns to avoid dividend issuances and stock splits. Days with half trading hours are also excluded. We do not, however, remove jumps from these intraday returns as they do not seem to matter for our purpose. We sample the stocks using the refresh time approach, as well as the previous tick method at a 15-minute frequency. We select 15 minutes because the Hausman tests proposed in Aït-Sahalia and Xiu (2019a) suggest it is a safe frequency at which to use realized covariance estimators for this pool of stocks.

Figure 1.1 plots the daily sample sizes after refresh time sampling for S&P 500, S&P 100, and Dow Jones 30 index constituents. In addition, it presents the quartiles of the daily sample sizes for S&P 500 index constituents by different shading. As we increase the number of assets, the daily refresh time observations decrease rapidly. Still, we are able to obtain on average 284 observations per day for S&P 500, which is approximately equivalent to sampling every 90 seconds. The average number of observations for S&P 100 and Dow Jones 30 constituents are 905 and 2,105, respectively.

We collect the GICS codes from the Compustat database for the Location thresholding

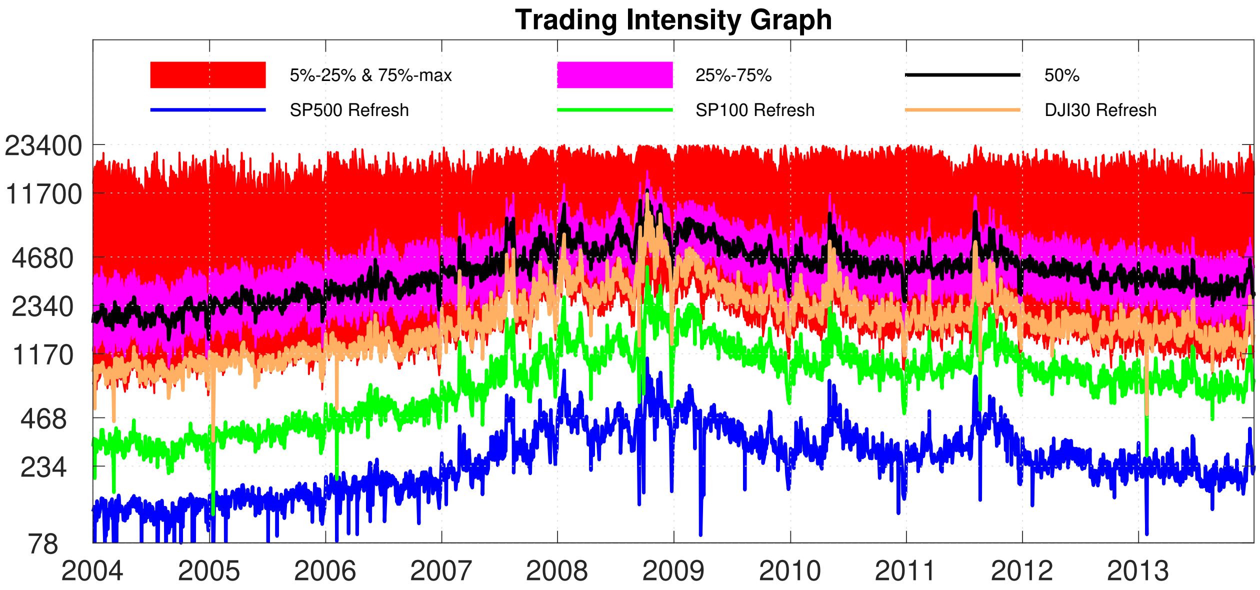

Figure 1.1: Trading Intensity of Assets

Note: This figure plots the max, 5%, 25%, 50%, and 75% quantiles of the daily number of observations for the S&P 500 index constituents after cleaning. It also plots the sample size after refresh time sampling for the Dow Jones 30, S&P 100, and S&P 500 index constituents, respectively.

method. The codes have 8 digits. Digits 1-2 indicate the company's sector; digits 3-4 describe the company's industry group; digits 5-6 describe the industry; digits 7-8 describe the sub-industry. Throughout 120 months and among the assets we consider, the time-series median of the largest block size is 80 for sector-based classification, 39 for industry group, 27 for industry, and 15 for sub-industry categories, for S&P 500 index constituents.

We construct observable factors from high-frequency transaction prices at a 15-minute frequency. The factors include the market portfolio, the small-minus-big (SMB) portfolio, the high-minus-low (HML) portfolio, the robust-minus-weak (RMW) portfolio, and the conservative-minus-aggressive (CMA) portfolio in the Fama-French 5-factor model. We also include the momentum (MOM) portfolio formed by sorting stock returns between the past 250 days and 21 days. We also collect the 9 industry SPDR ETFs from the TAQ database (Energy (XLE), Materials (XLB), Industrials (XLI), Consumer Discretionary (XLY), Consumer Staples (XLP), Health Care (XLV), Financial (XLF), Information Technology (XLK), and Utilities (XLU)). The time series of cumulative returns of all factors are plotted in Figure 1.2.

We obtain monthly factor loadings (exposures) from the MSCI Barra USE3 by Kahn, Brougham, and Green (1998). The loadings we utilize include Earnings Yield, Momentum, Trading Activity, Currency Sensitivity, Earnings Variation, Growth, Volatility, Dividend Yield, Size, Size Nonlinearity, Leverage, and Value. In addition, we construct and add the market exposure for each stock, using the slope coefficient in a time-series regression of its weighted daily returns on the weighted S&P 500 index returns over the trailing 252 trading days. The weights are chosen to have a half life of 63 days, so as to match the method documented by USE3. In total, we have 14 observable loadings including the intercept term. We normalize the factor exposures such that their cross-sectional means are 0 and variances are 1 for each month. Although the covariance matrix estimation is invariant under such transformations, the estimated factors now have similar scales. In case of missing exposures, we use their latest available values, or set them to 0 if they are missing throughout the entire sample period for certain stocks.[5] The cumulative returns of the estimated factors based on S&P 500 constituents are shown in Figure 1.3.

We plot the cumulative leading principal components of S&P 500 constituents using our PCA method in Figure 1.4. Recognizing a one-to-one correspondence among the factors in Figures 1.2, 1.3, and 1.4 is difficult, because the list of characteristics available from the MSCI Barra does not match the observed factors we obtain. Instead, we plot their generalized correlations using 15-minute returns in Figure 1.5, which measure how correlated two sets of factors are, as suggested by Bai and Ng (2006) and recently employed in Pelger (2015b). Indeed, strong coherence exists among the observed and inferred factors using different approaches, in particular among the PCA and the CSR factors.

[5] The empirical findings remain the same if we exclude stocks whose loadings are missing from the USE3 dataset.

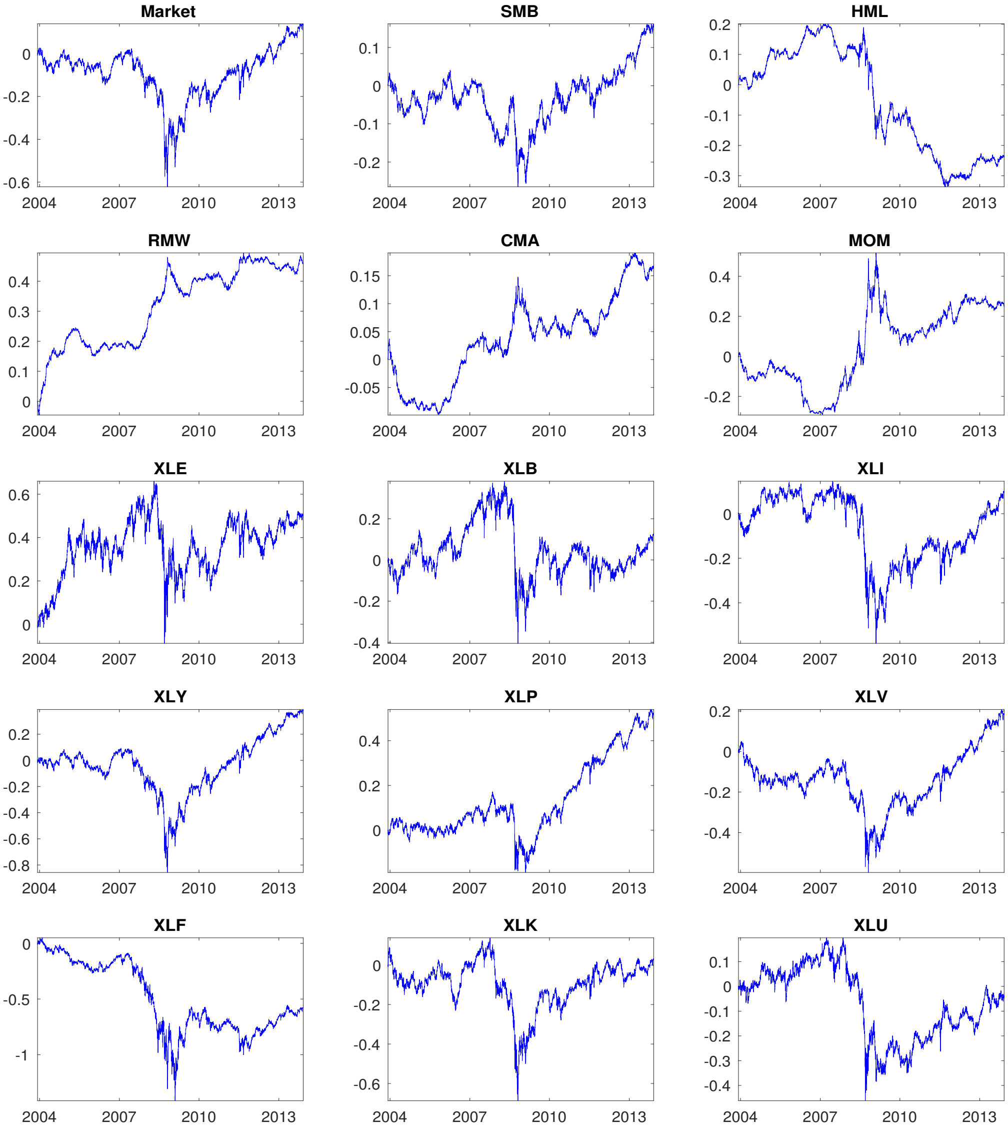

Figure 1.2: Time Series of Factors Used in TSR

Note: This figure plots the cumulative returns of the factors we have used in TSR, including the market portfolio, the small-minus-big market capitalization (SMB) portfolio, the high-minus-low price-earning ratio (HML) portfolio, the robust-minus-weak (RMW) portfolio, the conservative-minus-aggressive (CMA) portfolio, the momentum (MOM) portfolio, as well as 9 industry SPDR ETFs (Energy (XLE), Materials (XLB), Industrials (XLI), Consumer Discretionary (XLY), Consumer Staples (XLP), Health Care (XLV), Financial (XLF), Information Technology (XLK), and Utilities (XLU)). The overnight returns are excluded, same for the half trading days.

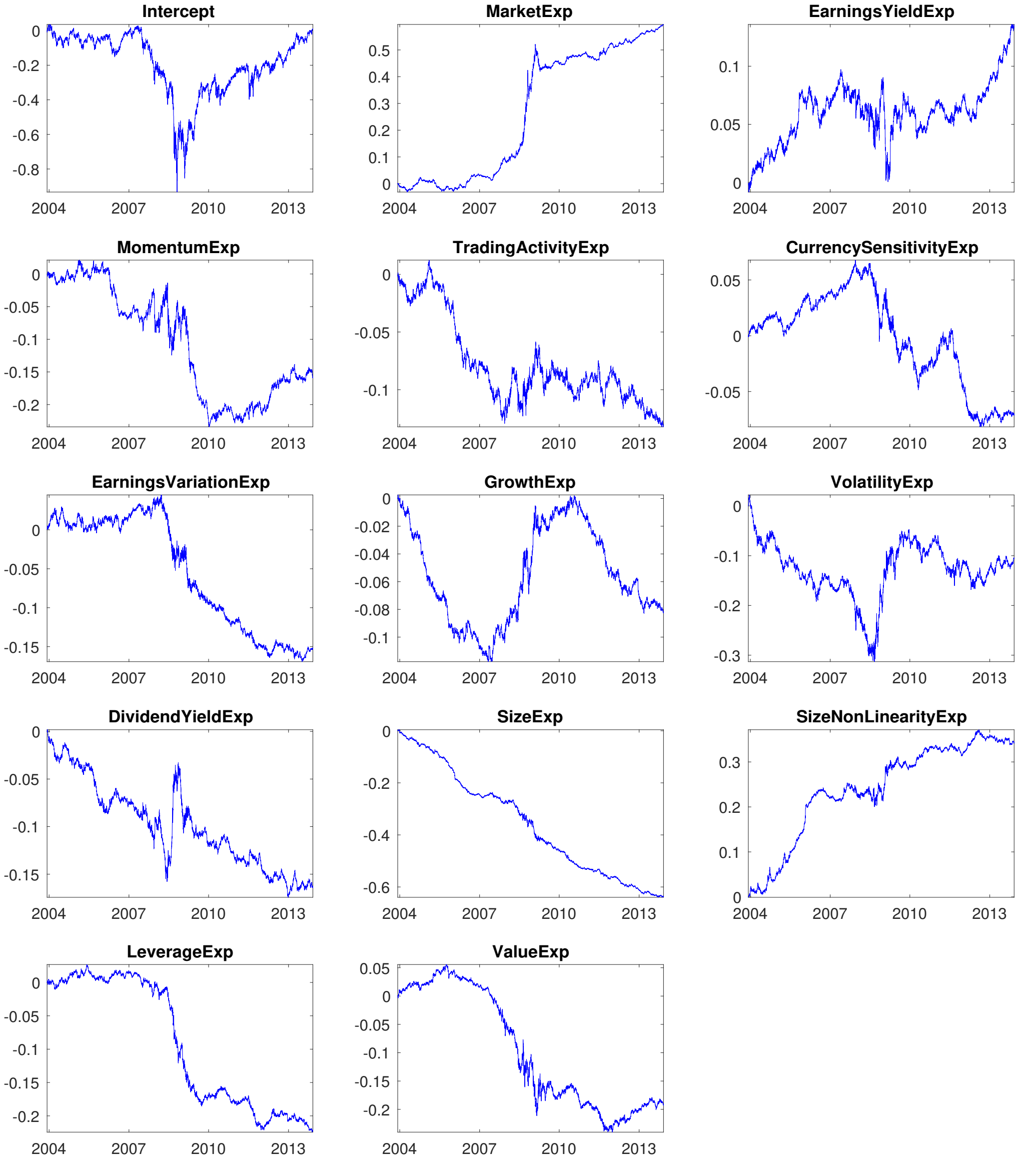

Figure 1.3: Time Series of Estimated Factors by CSR

Note: This figure plots the cumulative returns of factors we estimate using CSR, based on S&P 500 constituents. The corresponding factor exposures include the intercept, the market beta, and 12 other variables from MSCI Barra USE3 (Earnings Yield, Momentum, Trading Activity, Currency Sensitivity, Earnings Variation, Growth, Volatility, Dividend Yield, Size, Size Nonlinearity, Leverage, and Value).

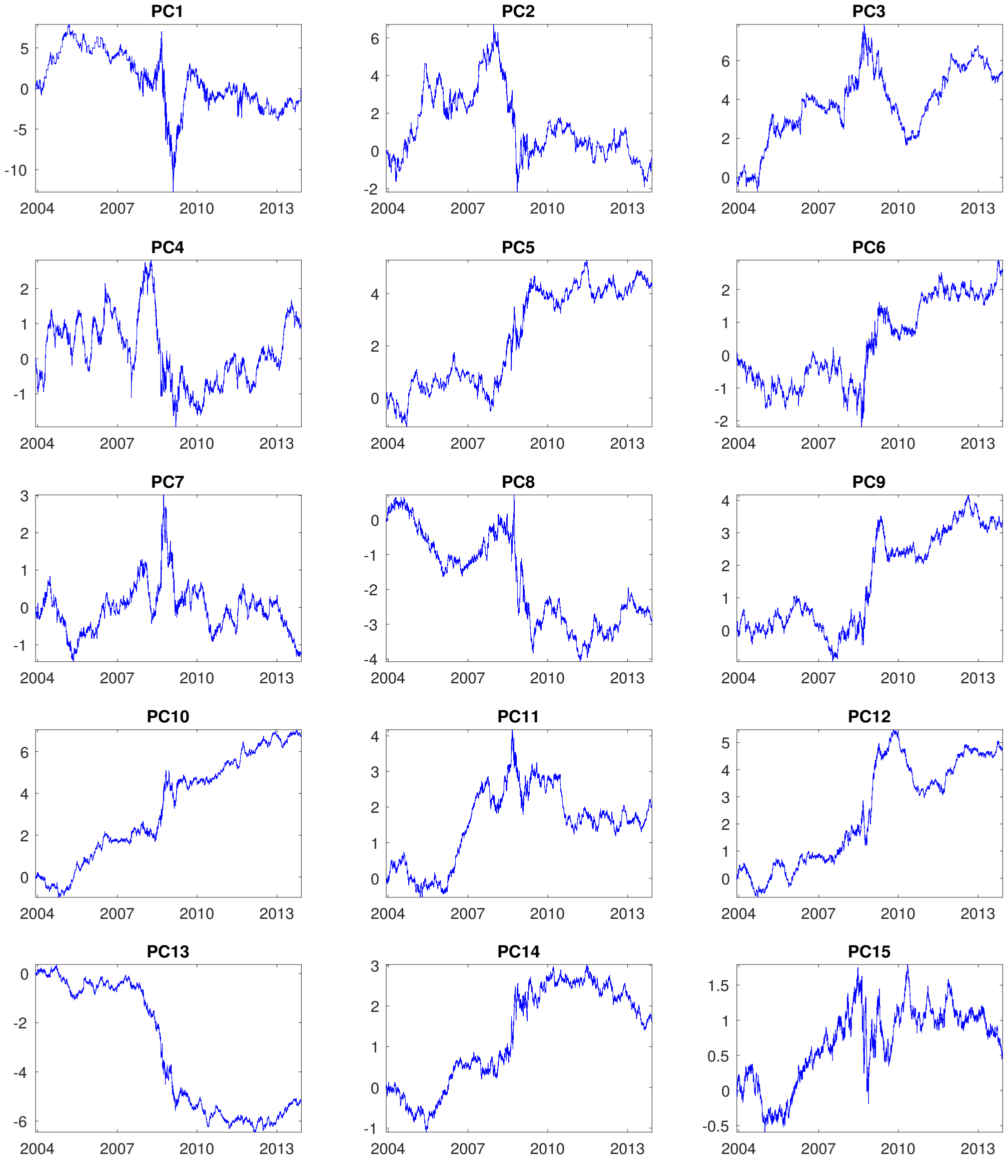

Figure 1.4: Time Series of Estimated Factors by PCA

Note: This figure plots the cumulative returns of the leading principal components of S&P 500 constituents we estimate using PCA.

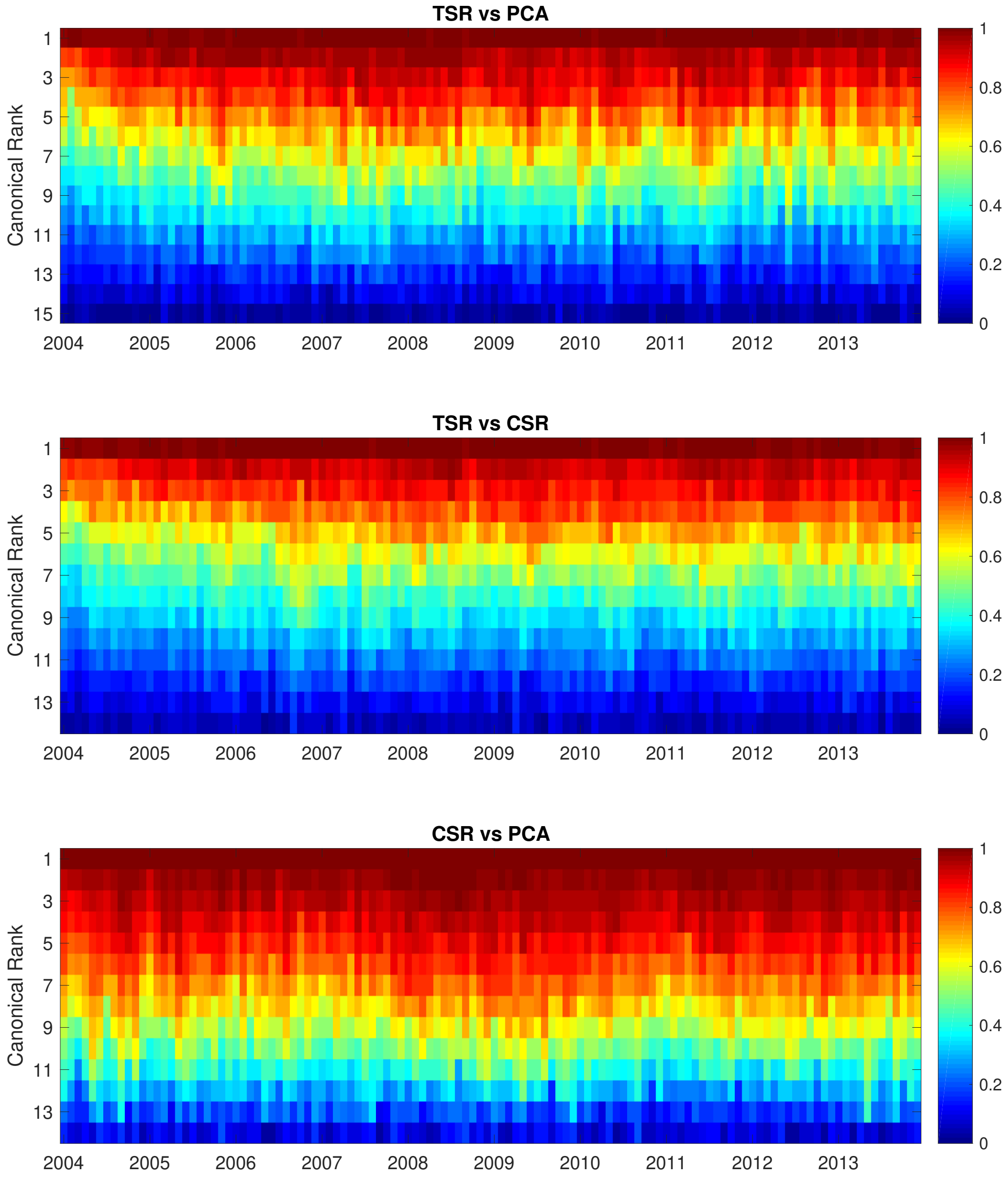

Figure 1.5: Generalized Correlation Plot

Note: This figure provides the heatmap of the monthly generalized (canonical) correlations among the time series of factors used in TSR and estimated from CSR and PCA methods. The y-axis is the rank of the generalized correlation. For two sets of factors X_a and X_b, the generalized correlation of rank k is calculated as the square-root of the kth-largest eigenvalue of the matrix $[X_a, X_a]^{-1}[X_a, X_b][X_b, X_b]^{-1}[X_b, X_a]$, where $[\cdot, \cdot]$ denotes the quadratic covariation.

1.6.2 Out-of-Sample Portfolio Allocation

We then examine the performance of the covariance matrix estimators in a constrained minimum variance portfolio allocation exercise because it requires only the estimated covariance matrix as an input. This approach to evaluating estimators of large covariance matrices is common; see, e.g., Fan, Zhang, and Yu (2012). Specifically, we consider the following optimization problem:

$$\min_{\omega} \omega^{\intercal}\widehat{\Sigma}\omega, \text{ subject to } \omega^{\intercal}\mathbf{1} = 1, \|\omega\|_1 \leqslant \gamma, \tag{1.10}$$

where $\|\omega\|_1 \leqslant \gamma$ imposes an exposure constraint. As explained in Fan, Furger, and Xiu (2016), when $\gamma = 1$, all portfolio weights must be non-negative, i.e., no short selling occurs. When γ is small, $\left\|\widehat{\Sigma} - \Sigma\right\|_{\text{MAX}}$ dictates the performance of the portfolio risk because the optimal portfolio comprises a relatively small number of stocks. By contrast, when γ is large, the portfolio is close to the global minimum variance portfolio, for which $\left\|\widehat{\Sigma}^{-1} - \Sigma^{-1}\right\|$ drives the performance of the portfolio risk. Therefore, investigating the out-of-sample risk of the portfolios in (1.10) for a variety of exposure constraints is informative about the quality of the covariance matrix estimation.

To focus on the evaluation of covariance matrix estimators, we intentionally adopt the simplest random walk forecasting model, i.e., $\widehat{\Sigma}_t \approx \mathbb{E}_t(\Sigma_{t+1})$, so that the estimated realized covariance matrices using data of the previous month are directly used for the portfolio construction the next month. For a range of exposure constraints, we measure the out-of-sample portfolio risk using 15-minute realized volatility. We compare the covariance matrices based on pre-averaging and subsampling methods, with many choices of θs and subsampling frequencies Δ_ns. Figures 1.6, 1.7, and 1.8 provide the results for the best choice of $\theta = 0.08$ and $\Delta_n = 900$ in simulations and the five thresholding methods we consider.

Figure 1.6 shows that (i) for S&P 500 constituents, the Location thresholding (black)

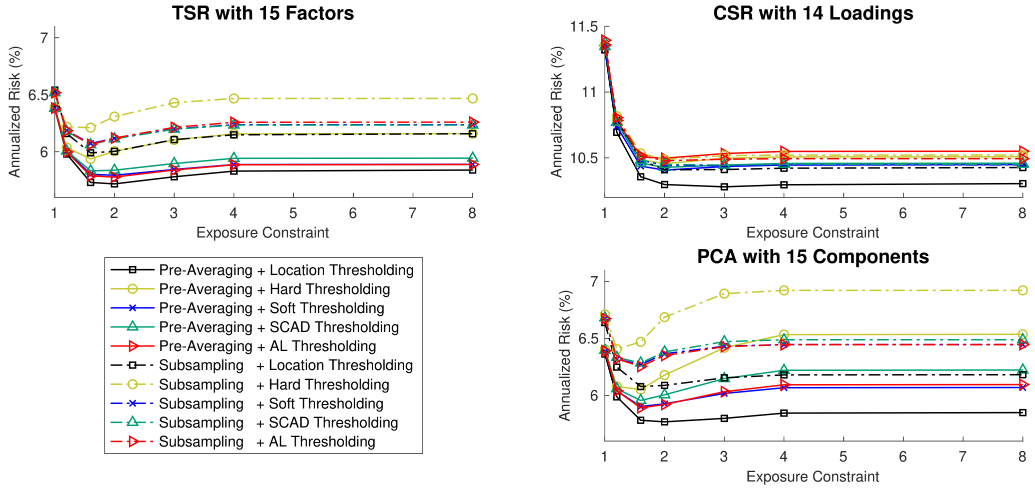

Figure 1.6: Out-of-Sample Risk of the S&P 500 Portfolio

Note: This figure compares the time-series average of the out-of-sample monthly volatility from 2004 to 2013 using S&P 500 Index constituents. The x-axis is the exposure constraint γ in the optimization problem (1.10). The results are based on a combination of methods: (pre-averaging, subsampling) × (TSR, CSR, PCA) × (Location, Hard, Soft, SCAD, AL thresholding). We use the GICS codes at the industry group level for the location thresholding method. The pre-averaging estimator uses $\theta = 0.08$, whereas the subsampling estimator uses 15-minute returns. The out-of-sample volatility is calculated using 15-minute subsampled returns each month.

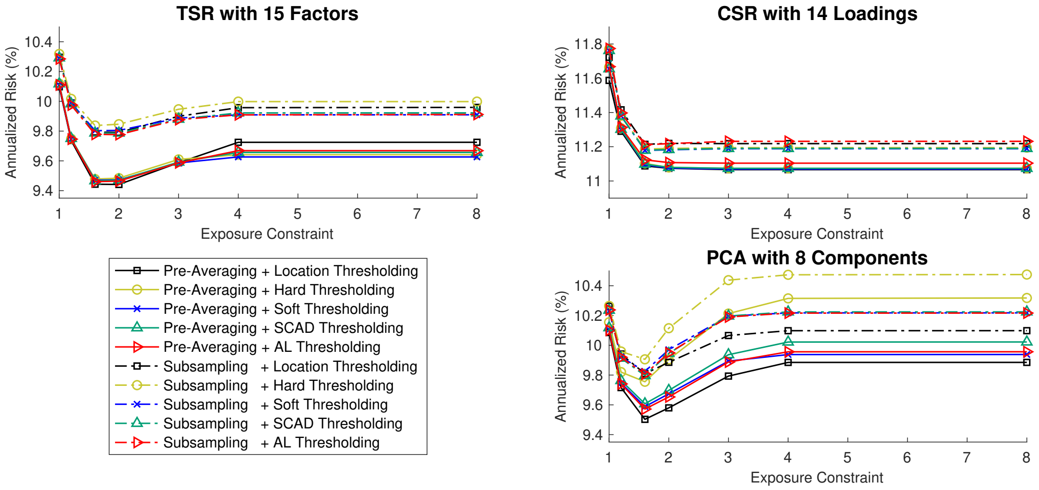

Figure 1.7: Out-of-Sample Risk of the S&P 100 Portfolio

Note: This figure compares the time-series average of the out-of-sample monthly volatility from 2004 to 2013 using S&P 100 Index constituents. All other settings remain the same as those in Figure 1.6.

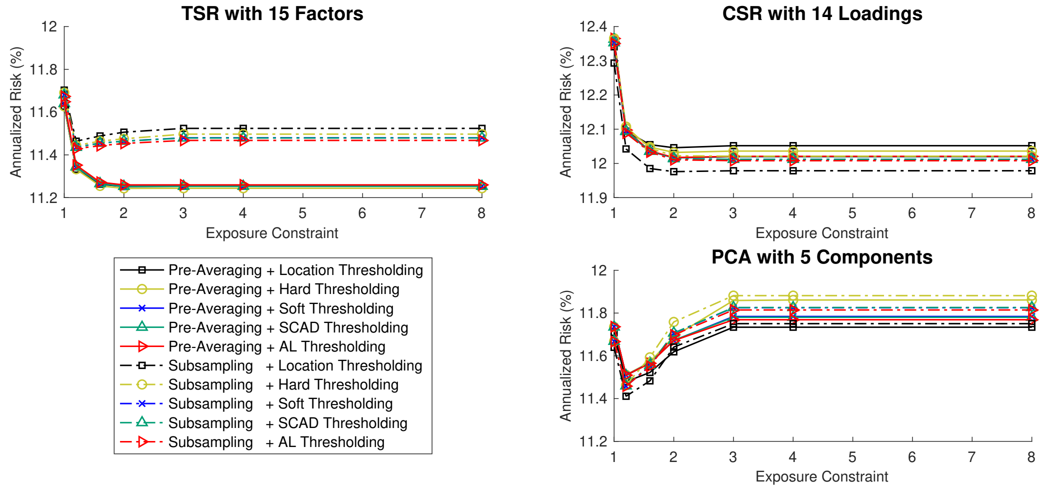

Figure 1.8: Out-of-Sample Risk of the Dow Jones 30 Portfolio

Note: This figure compares the time-series average of the out-of-sample monthly volatility from 2004 to 2013 using Dow Jones 30 Index constituents. All other settings remain the same as those in Figure 1.6.

performs the best among all thresholding methods, followed by Soft (blue), AL (red), and SCAD (green) approaches, with Hard thresholding (yellow) being the worst. (ii) The TSR approach appears to be the best, with the lowest out-of-sample risk and most stable performance across different thresholding methods. PCA is almost the same as TSR when Location thresholding is applied, but its performance deteriorates if we apply other thresholding techniques. CSR is dominated by the other two by a very large margin. This differs from our simulation results, indicating CSR suffers from more serious model misspecification. (iii) When the exposure constraint γ is small, the performance gap among different thresholding methods is small. This result is expected because the $\|\cdot\|_{\text{MAX}}$ norm differences across all methods are similar, and in this case, the portfolios are so heavily constrained that they are effectively low-dimensional. (iv) The pre-averaging estimators (solid lines) dominate the subsampling estimators (dash-dotted lines) across almost all cases, which agrees with our simulation results.

For S&P 100, we observe similar patterns from Figure 1.7, namely, that pre-averaging estimators outperform the subsampling estimators. PCA performs slightly worse than TSR. With respect to the Dow Jones 30, Figure 1.8 shows the CSR and PCA perform considerably worse than TSR. This finding is not surprising, given that our theory suggests consistency of these two estimators requires a large dimension, whereas for TSR, a smaller cross section works better.

Finally, we report in Table 1.5 and 1.6 the Diebold-Mariano (Diebold and Mariano (2002)) tests for comparison of the out-of-the-sample risk of portfolios based on the pre-averaging estimators against the subsampling estimators. Negative test statistics favor the pre-averaging approach. Similar to Figures 1.6 - 1.8, when γ is large, pre-averaging estimators deliver significantly smaller out-of-the-sample risk using TSR and PCA across all thresholding methods for S&P 500 index constituents. For S&P 100 and Dow-Jones index constituents, the difference between pre-averaging and subsampling is significant only if using TSR.

Exposure Constraint γ		1	1.2	1.6	2	3	4	8
					S&P 500			
Location	TSR	-1.51	-1.84*	-2.25**	-2.74***	-3.04***	-3.01***	-3.01***
	CSR	0.01	-0.39	-1.36	-1.43	-1.74*	-1.60	-1.56
	PCA	-2.20**	-2.44**	-2.53**	-2.69***	-2.85***	-2.69***	-2.66***
Hard	TSR	-1.43	-2.01**	-2.46**	-2.64***	-2.60***	-2.46**	-2.44**
	CSR	0.59	0.56	0.35	0.76	1.23	1.31	1.33
	PCA	-2.29**	-2.81***	-3.18***	-3.37***	-3.44***	-3.23***	-3.22***
Soft	TSR	-1.52	-2.09**	-2.79***	-3.28***	-3.44***	-3.42***	-3.40***
	CSR	0.49	0.35	0.08	0.43	0.78	0.85	0.89
	PCA	-2.21**	-2.65***	-3.20***	-3.42***	-3.59***	-3.50***	-3.48***
SCAD	TSR	-1.53	-2.04**	-2.57**	-2.98***	-3.08***	-2.98***	-2.97***
	CSR	0.56	0.50	0.31	0.51	0.99	1.09	1.11
	PCA	-2.21**	-2.65***	-3.09***	-3.32***	-3.23***	-2.91***	-2.88***
AL	TSR	-1.54	-2.15**	-2.84***	-3.33***	-3.53***	-3.46***	-3.44***
	CSR	0.61	0.45	0.47	0.80	1.30	1.43	1.44
	PCA	-2.18**	-2.60***	-3.10***	-3.27***	-3.32***	-3.10***	-3.09***

Table 1.5: Diebold-Mariano Tests for the Out-of-Sample Risk Comparison

Note: This table reports the Diebold-Mariano test statistics against the null that the out-of-sample risk of the portfolios based on the pre-averaging method is equal to that based on the subsampling approach. Negative numbers favor the pre-averaging approach. We use *, **, and *** to reveal the significance at the 10%, 5%, and 1% levels, respectively.

Exposure Constraint γ		1	1.2	1.6	2	3	4	8
		S&P 100						
Location	TSR	-1.57	-2.09**	-2.34**	-2.46**	-1.86*	-1.51	-1.53
	CSR	-2.43**	-2.79***	-3.23***	-3.19***	-3.30***	-3.30***	-3.30***
	PCA	-1.55	-1.93*	-1.58	-1.40	-0.83	-0.64	-0.64
Hard	TSR	-1.69*	-2.25**	-2.47**	-2.43**	-2.03**	-2.05**	-2.05**
	CSR	-1.87*	-1.89*	-1.67*	-1.87*	-1.93*	-1.93*	-1.93*
	PCA	-1.38	-1.64	-0.93	-1.06	-0.57	-0.31	-0.30
Soft	TSR	-1.67*	-2.31**	-2.63***	-2.59***	-2.12**	-2.05**	-2.05**
	CSR	-2.17**	-2.50**	-2.21**	-2.32**	-2.40**	-2.40**	-2.40**
	PCA	-1.70*	-2.16**	-1.53	-1.62	-1.22	-1.16	-1.16
SCAD	TSR	-1.69*	-2.34**	-2.63***	-2.56**	-2.02**	-1.93*	-1.93*
	CSR	-2.15**	-2.41**	-2.14**	-2.24**	-2.34**	-2.34**	-2.34**
	PCA	-1.60	-2.03**	-1.22	-1.32	-0.89	-0.72	-0.72
AL	TSR	-1.68*	-2.30**	-2.61***	-2.61***	-2.28**	-2.00**	-2.01**
	CSR	-2.17**	-2.39**	-2.11**	-2.25**	-2.35**	-2.35**	-2.35**
	PCA	-1.70*	-2.11**	-1.48	-1.48	-1.02	-0.91	-0.91
		Dow Jones 30						
Location	TSR	-1.83*	-2.22**	-2.70***	-2.64***	-2.55**	-2.55**	-2.55**
	CSR	-0.04	0.00	0.15	0.14	0.20	0.20	0.20
	PCA	0.71	0.38	-0.07	-0.69	-0.75	-0.75	-0.75
Hard	TSR	-1.74*	-1.91*	-2.65***	-2.50**	-2.44**	-2.44**	-2.44**
	CSR	-0.69	-0.76	-0.68	-0.60	-0.49	-0.49	-0.49
	PCA	0.63	0.23	-0.50	-1.06	-1.00	-0.97	-0.97
Soft	TSR	-0.61	-1.09	-2.39**	-2.28**	-2.21**	-2.21**	-2.21**
	CSR	-0.58	-0.76	-1.02	-0.97	-0.84	-0.84	-0.84
	PCA	0.62	0.23	-0.44	-0.97	-1.10	-1.10	-1.10
SCAD	TSR	-0.58	-1.07	-2.39**	-2.28**	-2.21**	-2.21**	-2.21**
	CSR	-0.58	-0.76	-1.02	-0.98	-0.85	-0.85	-0.85
	PCA	0.62	0.22	-0.46	-0.96	-1.10	-1.10	-1.10
AL	TSR	-0.06	-0.62	-2.08**	-2.01**	-1.98**	-1.98**	-1.98**
	CSR	-0.51	-0.65	-1.06	-1.01	-0.88	-0.88	-0.88
	PCA	0.64	0.25	-0.43	-0.85	-1.02	-1.02	-1.02

Table 1.6: Diebold-Mariano Tests for the Out-of-Sample Risk Comparison

Note: This table is a continuation of Table 1.5, where we report the Diebold-Mariano Tests for the S&P 100 and Dow Jones 30 index constituents. All other settings remain the same.

1.7 Conclusion

Leveraging a variety of factor models, we construct pre-averaging-based large covariance matrix estimators using high-frequency transaction prices, which are robust to the asynchronous arrival of trades and the market microstructure noise. We compare various estimators based on different combinations of factor model specifications and thresholding methods, in terms of their convergence rates, their finite sample behavior, and their empirical performance in a portfolio allocation horse race. Throughout, we find that pre-averaging plus TSR or PCA with Location thresholding dominates the other combinations, in particular the subsampling method. Also, CSR, the method MSCI Barra adopts for low-frequency data, performs considerably worse in almost all scenarios we study. This bad performance is perhaps driven by model misspecification, which can be alleviated with a potentially better set of factor exposures.

1.8 Appendix: Mathematical Proofs

1.8.1 Proof of Theorem 1

We need a few lemmas.

Lemma 1. *Suppose that* $n_\delta^{-1/2}\sqrt{\log d} = o(1)$*. Under Assumptions 1 - **??**, and for some constants* C_0*,* C_1 *and* C_2*, we have*

$$(i) \qquad \mathbb{P}\left(\left\|\widehat{\mathrm{E}} - \mathrm{E}\right\|_{\mathrm{MAX}} \geqslant C_0 n_\delta^{-1/2}\sqrt{\log d}\right) = O(r^2 C_1 d^{-C_0^2 C_2}), \tag{1.11}$$

$$(ii) \qquad \mathbb{P}\left(\left\|\widehat{\mathrm{E}} - \mathrm{E}\right\|_{\mathrm{F}} \geqslant C_0 r n_\delta^{-1/2}\sqrt{\log d}\right) = O(r^2 C_1 d^{-C_0^2 C_2}), \tag{1.12}$$

$$(iii) \qquad \mathbb{P}\left(\max_{\substack{1\leqslant k\leqslant r\\ 1\leqslant l\leqslant d}} \left|\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1} \bar{X}^\star_{k,i}\bar{Z}^\star_{o,li}\right| \geqslant C_0 n_\delta^{-1/2}\sqrt{\log d}\right) \tag{1.13}$$
$$= O(r C_1 d^{-C_0^2 C_2+1}),$$

$$(iv) \qquad \mathbb{P}\left(\max_{1\leqslant k,l\leqslant d} \left|\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1} \bar{Z}^\star_{o,ki}\bar{Z}^\star_{o,li} - \int_0^t g_{s,lk}ds\right|\right. \tag{1.14}$$
$$\left.\geqslant C_0 n_\delta^{-1/2}\sqrt{\log d}\right) = O(C_1 d^{-C_0^2 C_2+2}),$$

$$(v) \qquad \mathbb{P}\left(\max_{1\leqslant j\leqslant d} \left\|\widehat{\beta}_j - \beta_j\right\| \geqslant C_0 n_\delta^{-1/2}\sqrt{\log d}\right) = O(C_1 d^{-C_0^2 C_2+1}), \tag{1.15}$$

$$(vi) \qquad \mathbb{P}\left(\left\|\widehat{\beta} - \beta\right\|_{\mathrm{F}} \geqslant C_0 d^{1/2} n_\delta^{-1/2}\sqrt{\log d}\right) = O(C_1 d^{-C_0^2 C_2+1}), \tag{1.16}$$

$$(vii) \qquad \mathbb{P}\left(\left\|\widehat{\beta} - \beta\right\|_{\mathrm{MAX}} \geqslant C_0 n_\delta^{-1/2}\sqrt{\log d}\right) = O(C_1 d^{-C_0^2 C_2+1}), \tag{1.17}$$

$$(viii) \qquad \mathbb{P}\left(\max_{1\leqslant k\leqslant d}\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}\left(\left(\beta-\widehat{\beta}\right)^{\intercal}\bar{X}_i^{\star}\right)_k \geqslant C_0 r^2 n_{\delta}^{-1}\log d\right) \quad (1.18)$$
$$= O(C_1 d^{-C_0^2 C_2+1}),$$

$$(ix) \qquad \mathbb{P}\left(\max_{1\leqslant k,l\leqslant d}\left|\widehat{\Gamma}_{kl}-\Gamma_{kl}\right| \geqslant C_0 n_{\delta}^{-1/2}\sqrt{\log d}\right) = O(C_1 d^{-C_0^2 C_2+2}), \quad (1.19)$$

$$(x) \qquad \mathbb{P}\left(\max_{1\leqslant k,l\leqslant d}\left|\widehat{\Gamma}_{kl}^{S}-\Gamma_{kl}\right| \geqslant C_0 n_{\delta}^{-1/2}\sqrt{\log d}\right) = O(C_1 d^{-C_0^2 C_2+2}). \quad (1.20)$$

Proof of Lemma 1 . (i) Note that we have

$$\widehat{\mathrm{E}}_{kl} = \frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\left(\sum_{i=0}^{n-k_n+1}\bar{X}_{k,i}\bar{X}_{l,i} + \sum_{i=0}^{n-k_n+1}\bar{X}_{k,i}\bar{\varepsilon}_{l,i} + \sum_{i=0}^{n-k_n+1}\bar{X}_{l,i}\bar{\varepsilon}_{k,i} + \sum_{i=0}^{n-k_n+1}\bar{\varepsilon}_{k,i}\bar{\varepsilon}_{l,i}\right)$$
$$= T_1 + T_2 + T_3 + T_4,$$

therefore,

$$\mathbb{P}(|\widehat{\mathrm{E}}_{kl}-\mathrm{E}_{kl}| \geqslant u) \leqslant \mathbb{P}(|T_1-\mathrm{E}_{kl}| \geqslant u/4) + \mathbb{P}(|T_2| \geqslant u/4) + \mathbb{P}(|T_3| \geqslant u/4) + \mathbb{P}(|T_4| \geqslant u/4).$$

For T_1, this expression can be furthered decomposed as:

$$T_1 = \frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\left\{\sum_{i=1}^{n-k_n+1} a_{1,i}(k,l)(X_{k,t_i^k}-X_{k,t_{i-1}^k})(X_{l,t_i^l}-X_{l,t_{i-1}^l})\right.$$
$$\left. + \sum_{(i,j)\in\mathcal{F}} b_{1,ij}(k,l)(X_{k,t_i^k}-X_{k,t_{i-1}^k})(X_{l,t_i^l}-X_{l,t_{i-1}^l})\right\},$$

for certain numbers $a_{1,i}(k,l)$ and $b_{1,ij}(k,l)$ such that

$$|a_{1,i}(k,l)| + |b_{1,ij}(k,l)| \leqslant Ck_n.$$

The set $\mathcal{F}$ is given by

$$\mathcal{F} = \{(i,j)|1 \leqslant i \leqslant n - k_n + 1, 1 \leqslant j \leqslant n - k_n + 1, |i-j| \leqslant k_n - 1, i \neq j\}.$$

Let

$$A_{k,l}^i := \frac{n}{n-k_n+2}\frac{a_{1,i}(k,l)}{\psi_2 k_n} = O(1).$$

We insert synchronized true price X_{k,t_i} and $X_{k,t_{i-1}}$ between X_{k,t_i^k} and X_{k,t_{i-1}^k} and write

$$X_{k,t_i^k} - X_{k,t_{i-1}^k} = X_{k,t_i^k} - X_{k,t_i} + X_{k,t_i} - X_{k,t_{i-1}} + X_{k,t_{i-1}} - X_{k,t_{i-1}^k}.$$

Now using the above expression to expand $(X_{k,t_i^k} - X_{k,t_{i-1}^k})(X_{l,t_i^l} - X_{l,t_{i-1}^l})$, we obtain the following decomposition

$$\begin{aligned}
&\sum_{i=1}^{n-k_n+1} A_{k,l}^i (X_{k,t_i^k} - X_{k,t_{i-1}^k})(X_{l,t_i^l} - X_{l,t_{i-1}^l}) \\
=& \sum_{i=1}^{n-k_n+1} A_{k,l}^i \Big\{ (X_{k,t_i} - X_{k,t_{i-1}})(X_{l,t_i} - X_{l,t_{i-1}}) + (X_{k,t_i^k} - X_{k,t_i})(X_{l,t_i^l} - X_{l,t_i}) \\
&+ (X_{k,t_i^k} - X_{k,t_i})(X_{l,t_i} - X_{l,t_{i-1}}) + (X_{k,t_i^k} - X_{k,t_i})(X_{l,t_{i-1}} - X_{l,t_{i-1}^l}) \\
&+ (X_{k,t_i} - X_{k,t_{i-1}})(X_{l,t_i^l} - X_{l,t_i}) + (X_{k,t_i} - X_{k,t_{i-1}})(X_{l,t_{i-1}} - X_{l,t_{i-1}^l}) \\
&+ (X_{k,t_{i-1}} - X_{k,t_{i-1}^k})(X_{l,t_i^l} - X_{l,t_i}) + (X_{k,t_{i-1}} - X_{k,t_{i-1}^k})(X_{l,t_i} - X_{l,t_{i-1}}) \\
&+ (X_{k,t_{i-1}} - X_{k,t_{i-1}^k})(X_{l,t_{i-1}} - X_{l,t_{i-1}^l}) \Big\} \\
\equiv& \sum_{i=1}^{n-k_n+1} A_{k,l}^i \Delta_i^n X_k \Delta_i^n X_l + H_{kl}^1(1) + \cdots + H_{kl}^1(8).
\end{aligned}$$

For $\sum_{i=1}^{n-k_n+1} A_{k,l}^i \Delta_i^n X_k \Delta_i^n X_l$, denote $X_t^* = \int_0^t \sigma_s dW_s$, and denote for $1 \leqslant i \leqslant n$, $1 \leqslant k, l \leqslant$

p,

$$\zeta_{i,kl} = (\Delta_i^n X_k^*)(\Delta_i^n X_l^*), \quad \zeta'_{i,kl} = \mathbb{E}((\Delta_i^n X_k^*)(\Delta_i^n X_l^*)|\mathcal{F}_{(i-1)\Delta_n}), \quad \zeta''_{i,kl} = \zeta_{i,kl} - \zeta'_{i,kl},$$

then $M_t = \sum_{i=1}^{n-k_n+1} A^i_{k,l}\zeta''_{i,kl}$ is a continuous-time martingale. By Itô's lemma, we have

$$\begin{aligned}\left(X^*_{t,k} - X^*_{s,k}\right)\left(X^*_{t,l} - X^*_{s,l}\right) - \int_s^t (\sigma\sigma^\intercal)_{v,kl} dv = &\int_s^t \left(X^*_{v,k} - X^*_{s,k}\right)(\sigma_v dW_v)_l \\ &+ \int_s^t \left(X^*_{v,l} - X^*_{s,l}\right)(\sigma_v dW_v)_k.\end{aligned}$$

Therefore

$$\zeta''_{i,kl} = \int_{(i-1)\Delta_n}^{i\Delta_n} \left(X^*_{s,k} - X^*_{(i-1)\Delta_n,k}\right)(\sigma_s dW_s)_l + \int_{(i-1)\Delta_n}^{i\Delta_n} \left(X^*_{s,l} - X^*_{(i-1)\Delta_n,l}\right)(\sigma_s dW_s)_k.$$

Now we have

$$\begin{aligned}&\left|\sum_{i=1}^{n-k_n+1} A^i_{k,l}(X_{k,t_i^k} - X_{k,t_{i-1}^k})(X_{l,t_i^l} - X_{l,t_{i-1}^l}) - \sum_{i=1}^{n-k_n+1}\int_{(i-1)\Delta_n}^{i\Delta_n} A^i_{k,l}(\sigma\sigma^\intercal)_{s,kl} ds\right| \\ =&\left|\sum_{i=1}^{n-k_n+1} A^i_{k,l}\Delta_i^n X_k^* \int_{(i-1)\Delta_n}^{i\Delta_n} b_{s,l} ds + \sum_{i=1}^{n-k_n+1} A^i_{k,l}\Delta_i^n X_l^* \int_{(i-1)\Delta_n}^{i\Delta_n} b_{s,k} ds\right. \\ &\left.+ \sum_{i=1}^{n-k_n+1} A^i_{k,l} \int_{(i-1)\Delta_n}^{i\Delta_n} b_{s,l} ds \int_{(i-1)\Delta_n}^{i\Delta_n} b_{s,k} ds + M_t\right|.\end{aligned}$$

We proceed with each of the four terms, starting with M_t.

The quadratic variation of M_t is given by

$$\begin{aligned}[M,M]_t =&\Delta_n \sum_{i=1}^{n-k_n+1} \int_{(i-1)\Delta_n}^{i\Delta_n} (A^i_{k,l})^2 \Bigg(\left(X^*_{s,k} - X^*_{(i-1)\Delta_n,k}\right)^2 \sum_{r=1}^q \sigma^2_{s,lr} \\ &+ \left(X^*_{s,l} - X^*_{(i-1)\Delta_n,l}\right)^2 \sum_{r=1}^q \sigma^2_{s,kr}\end{aligned}$$

$$+2\left(X^*_{s,k}-X^*_{(i-1)\Delta_n,k}\right)\left(X^*_{s,l}-X^*_{(i-1)\Delta_n,l}\right)\sum_{r=1}^{q}\sigma_{s,lr}\sigma_{s,kr}\Bigg)\,ds.$$

According to Assumption 1, here we assume that $\|X_t\|_\infty \leqslant K$, $\|h_t\|_\infty \leqslant K$, and $\left\|\sigma_t\sigma_t^\intercal\right\|_{\mathrm{MAX}} \leqslant K$, for some constant $K > 0$. Therefore by Cauchy-Schwarz inequality, we have

$$[M,M]_t \leqslant 16K^3 t\Delta_n.$$

Then by the exponential inequality for a continuous martingale, we have

$$\mathbb{P}\left(\sup_{0\leqslant s\leqslant t}\left|\sum_{i=1}^{n-k_n+1} A^i_{k,l}\zeta''_{i,kl}\right| > u\right) \leqslant \exp\left(-\frac{nu^2}{32K^3t}\right). \tag{1.21}$$

In addition, by Cauchy-Schwarz inequality:

$$\begin{aligned}
&\mathbb{P}\left(\left|\sum_{i=1}^{n-k_n+1} A^i_{k,k}\Delta^n_i X^*_k\int_{(i-1)\Delta_n}^{i\Delta_n} b_{s,l}ds\right| > u\right)\\
\leqslant&\mathbb{P}\Bigg(\sum_{i=1}^{n-k_n+1} A^i_{k,k}(\Delta^n_i X^*_k)^2 - \sum_{i=1}^{n-k_n+1}\int_{(i-1)\Delta_n}^{i\Delta_n} A^i_{k,k}(\sigma\sigma^\intercal)_{s,kk}ds > \frac{x^2}{tK\Delta_n}\\
&-\sum_{i=1}^{n-k_n+1}\int_{(i-1)\Delta_n}^{i\Delta_n} A^i_{k,k}(\sigma\sigma^\intercal)_{s,kk}ds\Bigg)\\
\leqslant&\mathbb{P}\left(\left|\sum_{i=1}^{n-k_n+1} A^i_{k,k}(\Delta^n_i X^*_k)^2 - \sum_{i=1}^{n-k_n+1}\int_{(i-1)\Delta_n}^{i\Delta_n} A^i_{k,k}(\sigma\sigma^\intercal)_{s,kk}ds\right| > \frac{u^2}{tK\Delta_n} - tK\right)\\
\leqslant&\exp\left(-\frac{\left(\frac{u^2}{tK\Delta_n}-tK\right)^2}{32K^3t\Delta_n}\right),
\end{aligned}$$

where the last inequality follows from (1.21). Finally, notice that

$$\left|\sum_{i=1}^{n-k_n+1} A^i_{k,l}\int_{(i-1)\Delta_n}^{i\Delta_n} b_{s,l}ds\int_{(i-1)\Delta_n}^{i\Delta_n} b_{s,k}ds\right| \leqslant tK^2\Delta_n\max_i |A^i_{k,l}|,$$

we can derive

$$
\begin{aligned}
&\mathbb{P}\left(\left|\sum_{i=1}^{n-k_n+1} A^i_{k,l}(X_{k,t_i}-X_{k,t_{i-1}})(X_{l,t_i}-X_{l,t_{i-1}})-\sum_{i=1}^{n-k_n+1}\int_{(i-1)\Delta_n}^{i\Delta_n} A^i_{k,l}(\sigma\sigma^\intercal)_{s,kl}ds\right|>4u\right)\\
\leqslant&\mathbb{P}\left(|M_t|>u\right)+2\mathbb{P}\left(\left|\sum_{i=1}^{n-k_n+1} A^i_{k,l}\Delta^n_i X^*_k\int_{(i-1)\Delta_n}^{i\Delta_n} b_{s,l}ds\right|>u\right)+\mathbf{1}_{\{tK^2\Delta_n\max_i|A^i_{k,l}|>u\}}\\
\leqslant&\exp\left(-\frac{u^2}{32K^3t\Delta_n}\right)+2\exp\left(-\frac{\left(\frac{u^2}{tK\Delta_n}-tK\right)^2}{32K^3t\Delta_n}\right)\\
\leqslant&C_1\exp\left(-\frac{16C_2u^2}{\Delta_n}\right),
\end{aligned}
$$

where the above inequality holds if $x>(tK^2\Delta_n\max_i|A^i_{k,l}|)\vee(tK\sqrt{\Delta_n})\vee(tK\Delta_n/2\sqrt{1+4/\Delta_n})$, and $C_1\geqslant 3$, $C_2\leqslant(512K^3t)^{-1}$. On the other hand, if x violates this bound, i.e., $x\leqslant C'\sqrt{\Delta_n}$, we can choose C_1 such that $C_1\exp(-16C_2C'^2)\geqslant 1$, so that the inequality follows trivially. For $H^1_{kl}(1),\dots H^1_{kl}(8)$, we can use exactly the same technique for proving $\sum_{i=1}^{n-k_n+1}A^i_{k,l}\Delta^n_iX_k\Delta^n_iX_l$, and have that

$$
\mathbb{P}\left(\left|H^1_{kl}(1)+\cdots+H^1_{kl}(8)\right|\geqslant u\right)\leqslant C_pe^{-C_pnu^2}.
$$

Since it is easy to show that

$$
\sum_{i=1}^{n-k_n+1}\int_{(i-1)\Delta_n}^{i\Delta_n} A^i_{k,l}(\sigma\sigma^\intercal)_{s,kl}ds=\int_0^\intercal(\sigma\sigma^\intercal)_{s,kl}ds+o(n_\delta),
$$

we have

$$
\begin{aligned}
&\mathbb{P}\left(\left|\frac{n}{n-k_n+2}\frac{1}{\psi_2k_n}\sum_{i=1}^{n-k_n+1}a_{1,i}(k,l)A^i_{k,l}(X_{k,t^k_i}-X_{k,t^k_{i-1}})(X_{l,t^l_i}-X_{l,t^l_{i-1}})-\int_0^\intercal(\sigma\sigma^\intercal)_{s,kl}ds\right|\geqslant u\right)\\
\leqslant&\mathbb{P}\left(\left|\sum_{i=1}^{n-k_n+1} A^i_{k,l}(X_{k,t_i}-X_{k,t_{i-1}})(X_{l,t_i}-X_{l,t_{i-1}})-\sum_{i=1}^{n-k_n+1}\int_{(i-1)\Delta_n}^{i\Delta_n} A^i_{k,l}(\sigma\sigma^\intercal)_{s,kl}ds\right|>u/2\right)
\end{aligned}
$$

$$
\begin{aligned}
&+\mathbb{P}\left(\left|H_{kl}^{1}(1)+\cdots+H_{kl}^{1}(8)\right| \geqslant u/3\right)\\
&+\mathbb{P}\left(\left|\sum_{i=1}^{n-k_n+1}\int_{(i-1)\Delta_n}^{i\Delta_n} A_{k,l}^{i}(\sigma\sigma^{\mathsf{T}})_{s,kl}ds-\int_{0}^{\mathsf{T}}(\sigma\sigma^{\mathsf{T}})_{s,kl}ds\right| \geqslant u/3\right)\\
\leqslant &C_p e^{-C_p n u^2}+\mathbf{1}_{\{u<o(n_\delta)\}}\\
\leqslant &C_p e^{-C_p n u^2}.
\end{aligned}
$$

On the other hand, since we have

$$|a_{1,i}(k,l)|+|b_{1,ij}(k,l)| \leqslant Ck_n,$$

and

$$(X_{k,t_i^k}-X_{k,t_{i-1}^k})(X_{l,t_i^l}-X_{l,t_{i-1}^l})=O_p(n^{-1}),$$

$$(X_{k,t_i^k}-X_{k,t_{i-1}^k})(X_{l,t_i^l}-X_{l,t_{i-1}^l})=O_p(n^{-1}),$$

writing

$$X_{1,ij}=\frac{n}{n-k_n+2}\frac{n}{\psi_2 k_n}b_{1,ij}(k,l)(X_{k,t_i^k}-X_{k,t_{i-1}^k})(X_{l,t_i^l}-X_{l,t_{i-1}^l}),$$

then for $(i,j)\in\mathcal{F}$,

$$X_{1,ij}=O_p(n^{-1}).$$

Using a similar decomposition, we can obtain

$$\sum_{(i,j)\in\mathcal{F}}X_{1,ij}=\sum_{(i,j)\in\mathcal{F}}B_{k,l}^{i}\Delta_i^n X_k\Delta_i^n X_l+H_{kl}^{2}(1)+\cdots+H_{kl}^{2}(8).$$

Since $\|X_t\|_\infty$ is bounded, then $B_{k,l}^{i}\Delta_i^n X_k\Delta_i^n X_l-\mathrm{E}(B_{k,l}^{i}\Delta_i^n X_k\Delta_i^n X_l)$ is also bounded. Then according to the Hoeffding's lemma, we obtain $B_{k,l}^{i}\Delta_i^n X_k\Delta_i^n X_l-\mathrm{E}(B_{k,l}^{i}\Delta_i^n X_k\Delta_i^n X_l)$ is a sub-Gaussian random variable. Similar arguments can be extended to $H_{kl}^{2}(1),\ldots,H_{kl}^{2}(8)$. Then according to Hoeffding inequality, and $\sharp F_{kl}\leqslant Cnk_n$, where $\sharp F_{kl}$ denotes the number of elements in the set F_{kl}, then we have

$$\begin{aligned}
&\mathbb{P}\left(\left|\frac{1}{n}\sum_{(i,j)\in\mathcal{F}} X_{1,ij}\right| \geqslant u/8\right)\\
\leqslant&\mathbb{P}\left(\left|\frac{1}{n}\sum_{(i,j)\in\mathcal{F}} B_{k,l}^{i}\Delta_i^n X_k \Delta_i^n X_l - \mathbb{E}\left(\frac{1}{n}\sum_{(i,j)\in\mathcal{F}} B_{k,l}^{i}\Delta_i^n X_k \Delta_i^n X_l\right)\right| \geqslant u/24\right)\\
+&\mathbb{P}\left(\left|H_{kl}^2(1)+\cdots+H_{kl}^2(8)-\mathbb{E}(H_{kl}^2(1)+\cdots+H_{kl}^2(8))\right| \geqslant u/24\right) + \mathbf{1}\left(\left|\mathbb{E}\left(\frac{1}{n}\sum_{(i,j)\in\mathcal{F}} X_{1,ij}\right)\right| \geqslant u/24\right)\\
\leqslant&C_p e^{-\frac{C_p(nu)^2}{nk_n}} + \mathbf{1}_{\left\{\left|\mathbb{E}\left(\frac{1}{n}\sum_{(i,j)\in\mathcal{F}} X_{1,ij}\right)\right| \geqslant u/24\right\}}\\
\leqslant&C_p e^{-C_p n^{1/2-\delta}u^2}.
\end{aligned}$$

This inequality holds when $u \geqslant Ck_n/n$ for some constant C.

As for T_4, it can be decomposed as

$$\begin{aligned}
T_4 =&\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1} \bar{u}_{k,t_i^k}\bar{u}_{l,t_i^l} + \bar{v}_{k,t_i^k}\bar{v}_{l,t_i^k} + \bar{u}_{k,t_i^k}\bar{v}_{l,t_i^l} + \bar{v}_{k,t_i^k}\bar{u}_{l,t_i^l}\\
=&T_4^1 + T_4^2 + T_4^3 + T_4^4.
\end{aligned}$$

Using similar techniques of proving Proposition 10 in Kim, Wang, and Zou (2016), and under Assumption 4, we can prove that

$$\max_{1\leqslant k,l\leqslant d} \mathbb{E}\left|T_4\right| \leqslant C_p\left(n^{-2\delta} + n^{-1/4-2\delta} + n^{-1/2-2\delta}\right).$$

With respect to T_2, note that

$$T_2 = \frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1} \bar{X}_{k,t_i^k}\bar{u}_{l,t_i^l} + \bar{X}_{k,t_i^k}\bar{v}_{l,t_i^l} = T_2^1 + T_2^2.$$

For T_2^1, by Jensen's inequality, we have

$$\max_{1\leqslant k,l\leqslant d}\mathbb{E}|T_2^1| \leqslant C_p\frac{1}{k_n}\sum_{i=0}^{n-k_n+1}\mathbb{E}|\bar{X}_{k,t_i^k}|\mathbb{E}|\bar{u}_{l,t_i^l}|$$
$$\leqslant C_p\frac{1}{k_n}\sum_{i=0}^{n-k_n+1}\mathbb{E}|\bar{X}_{k,t_i^k}|k_n^{-1/2} \leqslant C_p\frac{1}{k_n}nn^{-1}k_n^{-1/2} = k_n^{-3/2}.$$

For T_2^2, by Jensen's inequality and Holder's inequality, we have

$$\max_{1\leqslant k,l\leqslant d}\mathbb{E}|T_2^2| \leqslant C_p\frac{1}{k_n}\sum_{i=0}^{n-k_n+1}[\mathbb{E}|\bar{X}_{k,t_i^k}|^2]^{1/2}[\mathbb{E}|\bar{v}_{l,t_i^l}^2]^{1/2} \leqslant C_pk_n^{-1}n^{1/4\varsigma}.$$

Therefore,

$$\max_{1\leqslant k,l\leqslant d}\mathbb{E}|T_2| \leqslant C_p(k_n^{-3/2}+k_n^{-1}n^{1/4\varsigma}).$$

Similarly, we can show that

$$\max_{1\leqslant k,l\leqslant d}\mathbb{E}|T_3| \leqslant C_p(k_n^{-3/2}+k_n^{-1}n^{1/4\varsigma}).$$

Using Talagrand's concentration inequality and some simple calculation, we have

$$\mathbb{P}(\max_{1\leqslant k,l\leqslant d}|T_2| \geqslant u) \leqslant d^2e^{-C_p(k_n^{-3}+k_n^{-2}n^{1/2\varsigma})^{-2}u^2},$$

$$\mathbb{P}(\max_{1\leqslant k,l\leqslant d}|T_3| \geqslant u) \leqslant d^2e^{-C_p(k_n^{-3}+k_n^{-2}n^{1/2\varsigma})^{-2}u^2},$$

and

$$\mathbb{P}(\max_{1\leqslant k,l\leqslant d}|T_4| \geqslant u) \leqslant d^2e^{-C_p(n^{-2\delta}+n^{-1/4-2\delta}+n^{-1/2-2\delta})^{-2}u^2}.$$

Above all, we prove that

$$\mathbb{P}(|\widehat{\mathrm{E}}_{kl}-\mathrm{E}_{kl}| \geqslant u) \leqslant C_1e^{-C_2[n^{1/2-\delta}+n^{4\delta}]u^2}.$$

Thus

$$\begin{aligned}&\mathbb{P}\left(\left\|\widehat{\mathrm{E}}-\mathrm{E}\right\|_{\mathrm{MAX}}\geqslant C_0 n_\delta^{-1/2}\sqrt{\log d}\right)\\ \leqslant & C_1 r^2 e^{-C_2[n^{1/2-\delta}+n^{4\delta}](C_0 n_\delta^{-1/2}\sqrt{\log d})^2}=C_1 d^{-C_0^2C_2}.\end{aligned}$$

(ii) Since

$$\left\|\widehat{\mathrm{E}}-\mathrm{E}\right\|_{\mathrm{F}}\leqslant r\left\|\widehat{\mathrm{E}}-\mathrm{E}\right\|_{\mathrm{MAX}},$$

then

$$\mathbb{P}\left(\left\|\widehat{\mathrm{E}}-\mathrm{E}\right\|_{\mathrm{F}}\geqslant C_0 r n_\delta^{-1/2}\sqrt{\log d}\right)\leqslant C_1 d^{-C_0^2C_2}.$$

(iii) The derivation of $\bar{X}^\star_{k,i}\bar{Z}^\star_{o,li}$ is similar to that of $\bar{X}^\star_{k,i}\bar{X}^\star_{l,i}$ given by (i), and under Assumption 3, we obtain

$$\mathbb{P}\left(\max_{1\leqslant k\leqslant r,1\leqslant l\leqslant d}\left|\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}\bar{X}^\star_{k,i}\bar{Z}^\star_{o,li}\right|\geqslant C_0 n_\delta^{-1/2}\sqrt{\log d}\right)\leqslant C_1 d^{-C_0^2C_2+1}.$$

(iv) By the similar argument as in (i), we have

$$\begin{aligned}&\mathbb{P}\left(\max_{1\leqslant k\leqslant r,1\leqslant l\leqslant d}\left|\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}\bar{Z}^\star_{o,ki}\bar{Z}^\star_{o,li}-\int_0^t g_{s,kl}ds\right|\geqslant C_0 n_\delta^{-1/2}\sqrt{\log d}\right)\\ \leqslant & C_1 d^{-C_0^2C_2+2}.\end{aligned}$$

(v)-(vii) Moreover, note that

$$\widehat{\beta}_j-\beta_j=\left(\widehat{\Pi}^{22}\right)^{-1}\widehat{\Pi}^{12}_j,$$

therefore, under the event that

$$A=\left\{\max_{1\leqslant i\leqslant s,1\leqslant j\leqslant d}\left|\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}\bar{X}^\star_{k,i}\bar{Z}^\star_{o,li}\right|\leqslant C_0 n_\delta^{-1/2}\sqrt{\log d}\right\}$$

$$\cap \left\{\lambda_{\min}\left(\widehat{\mathrm{E}}\right) \geqslant \frac{1}{2}\lambda_{\min}\left(\int_0^t e_s ds\right)\right\},$$

we have

$$\|\widehat{\beta}_j - \beta_j\|^2 \leqslant \frac{4}{\lambda_{\min}^2\left(\int_0^t e_s ds\right)} \sum_{i=1}^{r} \left(\widehat{\Pi}_{ij}^{12}\right)^2 \leqslant \frac{4rC_0^2 n_{\delta}^{-1}\log d}{\lambda_{\min}^2(\int_0^t e_s ds)}.$$

and

$$\|\widehat{\beta} - \beta\|_{\mathrm{F}}^2 \leqslant \frac{4rC_0^2 n^{-1/2+\delta} d\log d}{\lambda_{\min}^2(\int_0^t e_s ds)}.$$

Therefore, it suffices to show that $\mathbb{P}(A) \geqslant 1 - O(rn^{3/2+\delta-3\nu/4-\nu\delta/2}d^{-1})$.

We assume $\lambda_{\min}\left(\int_0^t e_s ds\right)$ is bounded away from 0 and r is finite, so it follows that

$$\begin{aligned}\mathbb{P}\left(\left\|\widehat{\mathrm{E}} - \int_0^t e_s ds\right\| \leqslant \frac{1}{2}\lambda_{\min}\left(\int_0^t e_s ds\right)\right) \geqslant & \mathbb{P}\left(r \max_{1\leqslant i,j\leqslant s}\left|\widehat{\mathrm{E}}_{ij} - \int_0^t e_{ij,s} ds\right| \leqslant \frac{1}{2}\lambda_{\min}\left(\int_0^t e_s ds\right)\right) \\ \geqslant & 1 - O(C_1 d^{-C_0^2 C_2}).\end{aligned}$$

By Lemma A.1 of Fan, Liao, and Mincheva (2011), we have

$$\mathbb{P}\left(\lambda_{\min}(\widehat{\mathrm{E}}) \geqslant \frac{1}{2}\lambda_{\min}\left(\int_0^t e_s ds\right)\right) \geqslant 1 - O(C_1 d^{-C_0^2 C_2}).$$

Combining this with (1.13), we have $\mathbb{P}(A) \geqslant 1 - O(C_1 d^{-C_0^2 C_2 + 1})$.

(viii) To prove (1.18), we note that

$$\begin{aligned}&\max_{1\leqslant k\leqslant d} \frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}\left(\sum_{l=1}^{r}(\beta_{k,l} - \widehat{\beta}_{k,l})\bar{X}_{l,i}\right)^2 \\ &\leqslant \max_{1\leqslant k\leqslant d}\|\widehat{\beta}_k - \beta_k\|^2 \frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}\|\bar{X}_i^{\star}\|^2.\end{aligned}$$

Then by (1.11) with $C > \max_{1 \leqslant k \leqslant r} \int_0^t e_{s,kk} ds$,

$$\begin{aligned}
&\mathbb{P}\left(\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1} \|\bar{X}_i^\star\|^2 \leqslant rC\right) \\
\geqslant&\mathbb{P}\left(s \max_{1\leqslant k\leqslant r}\left|\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1} \|\bar{X}_i^\star\|^2 - \int_0^t e_{s,kk} ds\right| + r\max_{1\leqslant k\leqslant r}\int_0^t e_{s,kk}ds \leqslant rC\right) \\
\geqslant&1 - O(C_1 d^{-C_0^2 C_2}).
\end{aligned}$$

By (1.15), we obtain

$$\begin{aligned}
&\mathbb{P}\left(\max_{1\leqslant k\leqslant d}\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}\left(\sum_{l=1}^r (\beta_{k,l}-\widehat{\beta}_{k,l})\bar{X}_{l,i}^\star\right)^2 > C[n^{-1/2+\delta}+n^{-4\delta}]r^2\log d\right) \\
&\leqslant O(C_1 d^{-C_0^2 C_2+1}).
\end{aligned}$$

(ix) Finally, under the event of

$$\begin{aligned}
&\left\{\max_{1\leqslant l\leqslant d}\left|\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}(\bar{Z}_{o,li}^\star)^2 - \int_0^t g_{s,ll}ds\right| \leqslant \frac{1}{4}\max_{1\leqslant l\leqslant d}\int_0^t g_{s,ll}ds\right\} \\
&\cap\left\{\max_{1\leqslant k\leqslant d}\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}\left(\sum_{l=1}^r(\beta_{k,l}-\widehat{\beta}_{k,l})\bar{X}_{l,i}^\star\right)^2 \leqslant C[n^{-1/2+\delta}+n^{-4\delta}]r^2\log d\right\},
\end{aligned}$$

according to Cauchy-Schwarz inequality, we have

$$\begin{aligned}
&\max_{1\leqslant k,l\leqslant d}\left|\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}\Big[(\bar{Y}_{k,i}^\star - (\widehat{\beta}\bar{X}_i^\star)_k)(\bar{Y}_{l,i}^\star-(\widehat{\beta}\bar{X}_i^\star)_l) - \bar{Z}_{o,ki}^\star\bar{Z}_{o,li}^\star\Big]\right| \\
\leqslant&\max_{1\leqslant k,l\leqslant d}\left|\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}((\widehat{\beta}-\beta)\bar{X}_i^\star)_k(\widehat{\beta}-\beta)\bar{X}_i^\star)_l\right| \\
&+2\max_{1\leqslant l,k\leqslant d}\left|\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}\bar{Z}_{o,ki}^\star((\widehat{\beta}-\beta)\bar{X}_i^\star)_l\right|
\end{aligned}$$

$$
\begin{aligned}
&\leqslant \max_{1\leqslant k\leqslant d} \frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}((\widehat{\beta}-\beta)\bar{X}_i^\star)_k^2\\
&\quad + 2\sqrt{\max_{1\leqslant k\leqslant d} \frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}(\bar{Z}_{o,ki}^\star)^2 \max_{1\leqslant k\leqslant d} \frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}((\widehat{\beta}-\beta)\bar{X}_i^\star)_k^2}\\
&\leqslant C_0[n^{-1/2+\delta}+n^{-4\delta}]r^2\log d + 2\sqrt{\left(\frac{5}{4}Ct\right)(C_0[n^{-1/2+\delta}+n^{-4\delta}]r^2\log d)}\\
&\leqslant C_0^{'} n_\delta^{-1/2} r\sqrt{\log d}.
\end{aligned}
$$

Consequently, we have

$$
\begin{aligned}
&\max_{1\leqslant k,l\leqslant d}\left|\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}\left[\left(\bar{Y}_{k,i}^\star-(\widehat{\beta}\bar{X}_i^\star)_k\right)\left(\bar{Y}_{l,i}^\star-(\widehat{\beta}\bar{X}_i^\star)_l\right)-\bar{Z}_{o,ki}^\star\bar{Z}_{o,li}^\star\right]\right|\\
&\leqslant C_0^{'} n_\delta^{-1/2} r\sqrt{\log d},
\end{aligned}
$$

with probability $1-O\left(n^{3/2+\delta-3\nu/4-\nu\delta/2}\right)$ by (1.14) and (1.16). Finally, by triangle inequality, we obtain

$$
\begin{aligned}
&\max_{1\leqslant l,k\leqslant d}\left|\widehat{\Gamma}_{kl}-\Gamma_{kl}\right|\\
&\leqslant \max_{1\leqslant k\leqslant r, 1\leqslant l\leqslant d}\left|\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}\bar{Z}_{k,i}^\star\bar{Z}_{l,i}^\star-\int_0^t g_{s,lk}ds\right|\\
&\quad + \max_{1\leqslant k,l\leqslant d}\left|\frac{n}{n-k_n+2}\frac{1}{\psi_2 k_n}\sum_{i=0}^{n-k_n+1}\left[\left(\bar{Y}_{k,i}^\star-(\widehat{\beta}\bar{X}_i^\star)_k\right)\left(\bar{Y}_{l,i}^\star-(\widehat{\beta}\bar{X}_i^\star)_l\right)-\bar{Z}_{o,ki}^\star\bar{Z}_{o,li}^\star\right]\right|,
\end{aligned}
$$

which leads to the result by (1.14).

(x) Under the event of

$$
B_1=\left\{\max_{k,l}|\widehat{\Gamma}_{kl}-\Gamma|\leqslant Csn_\delta^{-1/2}\sqrt{\log d}\right\},
$$

and

$$B_2 = \left\{ C_1 < \sqrt{\widehat{\Gamma}_{kk}\widehat{\Gamma}_{ll}} < C_2, \quad \textit{for all } k \leqslant d, l \leqslant d \right\},$$

here C_1 and C_2 are some constant, and $\omega_n = n_\delta^{-1/2}\sqrt{\log d}$. Because of $s_{kl}(z) = s_{kl}\left(z\mathbf{1}_{|z|>\tau\omega_n\sqrt{\widehat{\Gamma}_{kk}\widehat{\Gamma}_{ll}}}\right)$, we have

$$\begin{aligned}
\left\|\widehat{\Gamma}^S - \Gamma\right\|_{\mathrm{MAX}} \leqslant & \max_{1\leqslant k,l\leqslant d}\left|\widehat{\Gamma}_{kl}\mathbf{1}_{|\widehat{\Gamma}_{kl}|\geqslant\tau\omega_n\sqrt{\widehat{\Gamma}_{kk}\widehat{\Gamma}_{ll}}} + \Gamma_{kl}\mathbf{1}_{|\widehat{\Gamma}_{kl}|\geqslant\tau\omega_n\sqrt{\widehat{\Gamma}_{kk}\widehat{\Gamma}_{ll}}} - \Gamma_{kl}\mathbf{1}_{|\widehat{\Gamma}_{kl}|\leqslant\tau\omega_n\sqrt{\widehat{\Gamma}_{kk}\widehat{\Gamma}_{ll}}}\right| \\
\leqslant & \max_{1\leqslant k,l\leqslant d}\left|s_{kl}(\widehat{\Gamma}_{kl}) - \widehat{\Gamma}_{kl}|\mathbf{1}_{|\widehat{\Gamma}_{kl}|\geqslant\tau\omega_n\sqrt{\widehat{\Gamma}_{kk}\widehat{\Gamma}_{ll}}} + |\widehat{\Gamma}_{kl} - \Gamma_{kl}|\mathbf{1}_{|\widehat{\Gamma}_{kl}|\leqslant\tau\omega_n\sqrt{\widehat{\Gamma}_{kk}\widehat{\Gamma}_{ll}}}\right. \\
& \left. +\Gamma_{kl}\mathbf{1}_{|\widehat{\Gamma}_{kl}|\leqslant\tau\omega_n\sqrt{\widehat{\Gamma}_{kk}\widehat{\Gamma}_{ll}}}\right| \\
\leqslant & C\omega_n\sqrt{\widehat{\Gamma}_{kk}\widehat{\Gamma}_{ll}}\mathbf{1}_{|\widehat{\Gamma}_{kl}|\geqslant C\omega_n\theta_1} + C\omega_n\mathbf{1}_{|\widehat{\Gamma}_{kl}|\geqslant C\omega_n\theta_1} + C\omega_n\sqrt{\widehat{\Gamma}_{kk}\widehat{\Gamma}_{ll}} \\
\leqslant & C\omega_n.
\end{aligned}$$

Then we obtain

$$\left\|\widehat{\Gamma}^S - \Gamma\right\|_{\mathrm{MAX}} \leqslant Crn_\delta^{-1/2}\sqrt{\log d},$$

with probability at least $1 - O(C_1 d^{-C_0^2 C_2 + 2})$.

∎

Lemma 2. *Under Assumptions 1 - 4, and* $n_\delta^{-1/2}\sqrt{\log d} = o(1)$. *We have*

$$\textit{(i)}\ \mathbb{P}\left(\left\|\beta(\widehat{\mathrm{E}} - \mathrm{E})\beta^{\mathsf{T}}\right\|_\Sigma^2 + \left\|\beta\widehat{\mathrm{E}}(\widehat{\beta} - \beta)^{\mathsf{T}}\right\|_\Sigma^2 \geqslant C_0 d^{-1} n_\delta^{-1}\log d\right) = O(C_1 d^{-C_0^2 C_2 + 1}),$$

and

$$\textit{(ii)}\ \mathbb{P}\left(\left\|(\widehat{\beta} - \beta)\widehat{\mathrm{E}}(\widehat{\beta} - \beta)^{\mathsf{T}}\right\|_\Sigma^2 \geqslant C_0 d n_\delta^{-2}\log^2 dd\right) = O(C_1 d^{-C_0^2 C_2 + 1}).$$

Proof of Lemma 2. (i) For the first part, using the same argument in proof of theorem 2 in

Fan, Fan, and Lv (2008), we have

$$\left\| \beta^{\intercal} \Sigma^{-1} \beta \right\| \leqslant 2 \left\| cov^{-1}(X) \right\| .$$

Therefore

$$\begin{aligned}
\left\| \beta(\widehat{\mathrm{E}} - \mathrm{E}) \beta^{\intercal} \right\|_{\Sigma}^{2} =& d^{-1} tr \left(\Sigma^{-1/2} \beta(\widehat{\mathrm{E}} - \mathrm{E}) \beta^{\intercal} \Sigma^{-1} \beta(\widehat{\mathrm{E}} - \mathrm{E}) \beta^{\intercal} \Sigma^{-1/2} \right) \\
=& d^{-1} tr \left((\widehat{\mathrm{E}} - \mathrm{E}) \beta^{\intercal} \Sigma^{-1} \beta(\widehat{\mathrm{E}} - \mathrm{E}) \beta^{\intercal} \Sigma^{-1} \beta \right) \\
\leqslant& d^{-1} \left\| (\widehat{\mathrm{E}} - \mathrm{E}) \beta^{\intercal} \Sigma^{-1} \beta \right\|_{\mathrm{F}}^{2} \\
\leqslant& O(d^{-1}) \left\| \widehat{\mathrm{E}} - \mathrm{E} \right\|_{\mathrm{F}}^{2} .
\end{aligned}$$

On the other hand, we also have

$$\begin{aligned}
\left\| \beta \widehat{\mathrm{E}} (\widehat{\beta} - \beta)^{\intercal} \right\|_{\Sigma}^{2} \leqslant& \frac{1}{d} \left\| \beta^{\intercal} \Sigma^{-1} \beta \widehat{\mathrm{E}} (\widehat{\beta} - \beta) \right\|_{\mathrm{F}} \left\| \widehat{\mathrm{E}} \Sigma^{-1} (\widehat{\beta} - \beta)^{\intercal} \right\|_{\mathrm{F}} \\
\leqslant& \frac{1}{d} \left\| \beta^{\intercal} \Sigma^{-1} \beta \right\|_{\mathrm{F}} \left\| \widehat{\mathrm{E}} \right\|_{\mathrm{F}}^{2} \left\| \widehat{\beta} - \beta \right\|_{\mathrm{F}}^{2} .
\end{aligned}$$

Then by Lemma 1 (1.12) and (1.16), and $\mathbb{P}(\|\widehat{\mathrm{E}}\|_{\mathrm{F}}^{2} > C) = O(C_1 r^2 d^{-C_0^2 C_2})$. We can get the final results.

(ii) For the second part, we have

$$\begin{aligned}
\left\| (\widehat{\beta} - \beta) \widehat{\mathrm{E}} (\widehat{\beta} - \beta)^{\intercal} \right\|_{\Sigma}^{2} =& \frac{1}{d} tr \left((\widehat{\beta} - \beta) \widehat{\mathrm{E}} (\widehat{\beta} - \beta)^{\intercal} \Sigma^{-1} (\widehat{\beta} - \beta) \widehat{\mathrm{E}} (\widehat{\beta} - \beta)^{\intercal} \Sigma^{-1} \right) \\
\leqslant& \frac{1}{d} \left\| (\widehat{\beta} - \beta) \widehat{\mathrm{E}} (\widehat{\beta} - \beta)^{\intercal} \Sigma^{-1} \right\|_{\mathrm{F}}^{2} \\
\leqslant& \frac{1}{d} \lambda_{\mathrm{MAX}}^{2}(\Sigma^{-1}) \lambda_{\mathrm{MAX}}^{2}(\widehat{\mathrm{E}}) \left\| \widehat{\beta} - \beta \right\|_{\mathrm{F}}^{4} .
\end{aligned}$$

Since $\lambda_{\mathrm{MAX}}^{2}(\Sigma^{-1})$ and $\lambda_{\mathrm{MAX}}^{2}(\widehat{\mathrm{E}})$ are both bounded, then the result follows from (1.16). ∎

Lemma 3. *Under Assumptions 1 - 4, and* $n_\delta^{-1/2}\sqrt{\log d} = o(1)$*, we have*

$$
\begin{aligned}
&(i) && \mathbb{P}\left(\left\|\widehat{\Gamma}^S - \Gamma\right\| > C_0 m_d n_\delta^{-(1-q)/2}(\log d)^{(1-q)/2}\right) = O(n^{3/2+\delta-3\nu/4-\nu\delta/2}), && (1.22)\\
&(ii) && \mathbb{P}\left(\lambda_{\min}\left(\widehat{\Gamma}^S\right) \geqslant \frac{1}{2}\lambda_{\min}\left(\Gamma\right)\right) > 1 - O(n^{3/2+\delta-3\nu/4-\nu\delta/2}), && (1.23)\\
&(iii) && \mathbb{P}\left(\left\|(\widehat{\Gamma}^S)^{-1} - \Gamma^{-1}\right\| > C_0 m_d n_\delta^{-(1-q)/2}(\log d)^{(1-q)/2}\right) && \\
& && = O(n^{3/2+\delta-3\nu/4-\nu\delta/2}), && (1.24)\\
&(iv) && \mathbb{P}\left(\left\|\widehat{\beta}^\intercal(\widehat{\Gamma}^S)^{-1}\widehat{\beta} - \beta^\intercal\Gamma^{-1}\beta\right\| > C_0 m_d d n_\delta^{-(1-q)/2}(\log d)^{(1-q)/2}\right) && \\
& && = O(n^{3/2+\delta-3\nu/4-\nu\delta/2}), && (1.25)\\
&(v) && \mathbb{P}\left(\left\|\left(\widehat{\mathrm{E}}^{-1} + \widehat{\beta}^\intercal(\widehat{\Gamma}^S)^{-1}\widehat{\beta}\right) - \left(\mathrm{E}^{-1} + \beta^\intercal(\Gamma^S)^{-1}\beta\right)\right\|\right. && \\
& && \left. > C_0 m_d d n_\delta^{-(1-q)/2}(\log d)^{(1-q)/2}\right) = O(n^{3/2+\delta-3\nu/4-\nu\delta/2}), && (1.26)\\
&(vi) && \mathbb{P}\left(\left\|\left(\widehat{\mathrm{E}}^{-1} + \widehat{\beta}^\intercal(\widehat{\Gamma}^S)^{-1}\widehat{\beta}\right)^{-1}\right\| > C_0 m_d d^{-1}\right) = O(n^{3/2+\delta-3\nu/4-\nu\delta/2}), && (1.27)\\
&(vii) && \mathbb{P}\left(\left\|\widehat{\beta}\left(\widehat{\mathrm{E}}^{-1} + \widehat{\beta}^\intercal(\widehat{\Gamma}^S)^{-1}\widehat{\beta}\right)^{-1}\widehat{\beta}^\intercal(\widehat{\Gamma}^S)^{-1}\right\| > C_0 m_d\right) = O(n^{3/2+\delta-3\nu/4-\nu\delta/2}), && (1.28)\\
&(viii) && \mathbb{P}\left(\left\|\widehat{\beta}\left(\widehat{\mathrm{E}}^{-1} + \widehat{\beta}^\intercal(\widehat{\Gamma}^S)^{-1}\widehat{\beta}\right)^{-1}\widehat{\beta}^\intercal\Gamma^{-1}\right\| > C_0 m_d\right) = O(n^{3/2+\delta-3\nu/4-\nu\delta/2}). && (1.29)
\end{aligned}
$$

Proof of Lemma 3. (i) Since $\widehat{\Gamma}^S - \Gamma$ is symmetric, its operator norm is bounded by the ∞-norm:

$$\left\|\widehat{\Gamma}^S - \Gamma\right\| \leqslant \max_{1\leqslant l\leqslant d}\sum_{k=1}^{d}\left|\widehat{\Gamma}^S_{lk} - \Gamma_{lk}\right|.$$

Then using the same technique for proving (1.20), we can prove that, with probability no less than $O(n^{3/2+\delta-3\nu/4-\nu\delta/2})$, we have

$$\left\|\widehat{\Gamma}^S - \Gamma\right\| \leqslant C_0 m_d n_\delta^{-(1-q)/2}(\log d)^{(1-q)/2}.$$

Proofs of (1.23)-(1.29) are similar to that of Lemma 5 in Fan, Furger, and Xiu (2016), therefore we omit the details.

∎

Proof of Theorem 1. Based on Lemma 1, and following the same steps as that of theorem 1 in Fan, Furger, and Xiu (2016), we can obtain

$$\left\|\widehat{\Sigma}_{\mathrm{TSR}} - \Sigma\right\|_{\mathrm{MAX}} = O\left(n_{\delta}^{-1/2}\sqrt{\log d}\right).$$

For the next part, we will prove the convergence results based on the Σ norm:

$$\begin{aligned}\left\|\hat{\Sigma}_{\mathrm{TSR}} - \Sigma\right\|_{\Sigma}^{2} \leqslant & 4\left\|\beta(\widehat{\mathrm{E}} - \mathrm{E})\beta^{\intercal}\right\|_{\Sigma}^{2} + 24\left\|\beta\widehat{\mathrm{E}}(\widehat{\beta} - \beta)\right\|_{\Sigma}^{2} \\ & + 16\left\|(\widehat{\beta} - \beta)\widehat{\mathrm{E}}(\widehat{\beta} - \beta)^{\intercal}\right\|_{\Sigma}^{2} + 2\left\|\hat{\Gamma}^{S} - \Gamma\right\|_{\Sigma}^{2}. \end{aligned} \tag{1.30}$$

Finally, we have

$$\begin{aligned}\left\|\hat{\Gamma}^{S} - \Gamma\right\|_{\Sigma} = & d^{-1/2}\left\|\Sigma^{-1/2}(\hat{\Gamma}^{S} - \Gamma)\Sigma^{-1/2}\right\|_{\mathrm{F}} \leqslant \left\|\Sigma^{-1/2}(\hat{\Gamma}^{S} - \Gamma)\Sigma^{-1/2}\right\| \\ \leqslant & \left\|\hat{\Gamma}^{S} - \Gamma\right\|\lambda_{\max}(\Sigma^{-1}).\end{aligned}$$

Then based on (1.30), Lemma 2 and Lemma 3 (1.22), and the fact that

$$\begin{aligned}& d^{-1}n_{\delta}^{-1}\log d + dn_{\delta}^{-2}\log^{2} dd + m_{d}^{2}m_{d}n_{\delta}^{-(1-q)}(\log d)^{(1-q)} \\ = & O\left(dn_{\delta}^{-2}\log^{2} dd + m_{d}^{2}m_{d}n_{\delta}^{-(1-q)}(\log d)^{(1-q)}\right),\end{aligned}$$

we prove that

$$\left\|\widehat{\Sigma}_{\mathrm{TSR}} - \Sigma\right\|_{\Sigma} = O_{p}\left(d^{1/2}n_{\delta}^{-1}\log d + m_{d}n_{\delta}^{-(1-q)/2}(\log d)^{(1-q)/2}\right).$$

On the other hand, if we do not assume the factor structure, using a direct pre-averaging estimator $\hat{\Sigma}^{\star}$. Then we will get

$$\left\|\hat{\Sigma}^{\star}-\Sigma\right\|_{\Sigma}^{2}\leqslant C\left\|\beta(\widehat{\mathrm{E}}-\mathrm{E})\beta^{\intercal}\right\|_{\Sigma}^{2}+C\left\|\beta\bar{X}\bar{Z}^{\intercal}\right\|_{\Sigma}^{2}+C\left\|\bar{Z}\bar{Z}^{\intercal}-\Gamma\right\|_{\Sigma}^{2}.$$

According to the proof of Lemma 2, we obtain $\left\|\beta(\widehat{\mathrm{E}}-\mathrm{E})\beta^{\intercal}\right\|_{\Sigma}^{2}=O_{P}(d^{-1}n_{\delta}^{-1}\log d)$ and $\left\|\beta\bar{X}\bar{Z}^{\intercal}\right\|_{\Sigma}^{2}=O_{P}(d^{-1}n_{\delta}^{-1}\log d)$. We can also get $\left\|\bar{Z}\bar{Z}^{\intercal}-\Gamma\right\|_{\Sigma}^{2}=O_{P}(dn_{\delta}^{-1}\log d)$. Therefore $\left\|\hat{\Sigma}^{\star}-\Sigma\right\|_{\Sigma}=O_{P}(d^{1/2}n_{\delta}^{-1/2}\sqrt{\log d})$, which has slower convergence rate than our estimator.

For the inverse part, by the localization argument, we only need to prove the result under a stronger assumption that the entry-wise norms of all the processes are bounded uniformly in $[0,t]$. By the Sherman - Morrison -Woodbury formula, we have

$$\begin{aligned}
&\left\|(\widehat{\Sigma}_{\mathrm{TSR}})^{-1}-\Sigma^{-1}\right\|\\
\leqslant&\left\|(\widehat{\Gamma}^{S})^{-1}-\Gamma^{-1}\right\|+\left\|\left((\widehat{\Gamma}^{S})^{-1}-\Gamma^{-1}\right)\widehat{\beta}\left(\widehat{\mathrm{E}}^{-1}+\widehat{\beta}^{\intercal}(\widehat{\Gamma}^{S})^{-1}\widehat{\beta}\right)^{-1}\widehat{\beta}^{\intercal}(\widehat{\Gamma}^{S})^{-1}\right\|\\
&+\left\|\left((\widehat{\Gamma}^{S})^{-1}-\Gamma^{-1}\right)\widehat{\beta}\left(\widehat{\mathrm{E}}^{-1}+\widehat{\beta}^{\intercal}(\widehat{\Gamma}^{S})^{-1}\widehat{\beta}\right)^{-1}\widehat{\beta}^{\intercal}\Gamma^{-1}\right\|\\
&+\left\|\Gamma^{-1}(\widehat{\beta}-\beta)\left(\widehat{\mathrm{E}}^{-1}+\widehat{\beta}^{\intercal}(\widehat{\Gamma}^{S})^{-1}\widehat{\beta}\right)^{-1}\widehat{\beta}^{\intercal}\Gamma^{-1}\right\|+\left\|\Gamma^{-1}(\widehat{\beta}-\beta)\left(\widehat{\mathrm{E}}^{-1}+\widehat{\beta}^{\intercal}(\widehat{\Gamma}^{S})^{-1}\widehat{\beta}\right)^{-1}\beta^{\intercal}\Gamma^{-1}\right\|\\
&+\left\|\Gamma^{-1}\beta\left(\left(\widehat{\mathrm{E}}^{-1}+\widehat{\beta}^{\intercal}(\widehat{\Gamma}^{S})^{-1}\widehat{\beta}\right)^{-1}-\left(\mathrm{E}^{-1}+\beta^{\intercal}(\Gamma^{S})^{-1}\beta\right)^{-1}\right)\beta^{\intercal}\Gamma^{-1}\right\|\\
:=&L_{1}+L_{2}+L_{3}+L_{4}+L_{5}+L_{6}.
\end{aligned}$$

We now bound each term above with probability no less than $1-O(n^{3/2+\delta-3\nu/4-\nu\delta/2})$. First of all, by (1.22)

$$L_{1}\leqslant Cm_{d}n_{\delta}^{-(1-q)/2}(\log d)^{(1-q)/2}.$$

To bound L_2, by (1.24) and (1.28), we have

$$L_{2}\leqslant\left\|(\widehat{\Gamma}^{S})^{-1}-\Gamma^{-1}\right\|\cdot\left\|\widehat{\beta}\left(\widehat{\mathrm{E}}^{-1}+\widehat{\beta}^{\intercal}(\widehat{\Gamma}^{S})^{-1}\widehat{\beta}\right)^{-1}\widehat{\beta}^{\intercal}(\widehat{\Gamma}^{S})^{-1}\right\|\leqslant Cm_{d}^{2}n_{\delta}^{-(1-q)/2}(\log d)^{(1-q)/2}.$$

Similarly, L_3 can be bounded using (1.24) and (1.29).

Next, for L_4, we use (1.16), and (1.27), $\|\cdot\| \leqslant \|\cdot\|_{\mathrm{F}}$, and $\lambda_{\min}(\Gamma)$ is bounded below by some constant,

$$L_4 \leqslant \left\|\Gamma^{-1}\right\|^2 \cdot \left\|\widehat{\beta} - \beta\right\| \cdot \left\|\widehat{\beta}\right\| \cdot \left\|\left(\widehat{\mathrm{E}}^{-1} + \widehat{\beta}^{\intercal}(\widehat{\Gamma}^S)^{-1}\widehat{\beta}\right)^{-1}\right\| \leqslant C m_d^2 n_\delta^{-(1-q)/2} (\log d)^{(1-q)/2}.$$

Similarly, using the fact that $\|\beta\| \leqslant \|\beta\|_{\mathrm{F}} = O(\sqrt{d})$, we can establish the same bound for L_5.

Finally, we have

$$\begin{aligned}L_6 \leqslant & \left\|\Gamma^{-1}\right\|^2 \cdot \|\beta\|^2 \cdot \left\|\left(\widehat{\mathrm{E}}^{-1} + \widehat{\beta}^{\intercal}(\widehat{\Gamma}^S)^{-1}\widehat{\beta}\right)^{-1} - \left(\mathrm{E}^{-1} + \beta^{\intercal}(\Gamma^S)^{-1}\beta\right)^{-1}\right\| \\ \leqslant & \left\|\Gamma^{-1}\right\|^2 \cdot \|\beta\|^2 \cdot \left\|\left(\widehat{\mathrm{E}}^{-1} + \widehat{\beta}^{\intercal}(\widehat{\Gamma}^S)^{-1}\widehat{\beta}\right) - \left(\mathrm{E}^{-1} + \beta^{\intercal}(\Gamma^S)^{-1}\beta\right)\right\| \\ & \cdot \left\|\left(\widehat{\mathrm{E}}^{-1} + \widehat{\beta}^{\intercal}(\widehat{\Gamma}^S)^{-1}\widehat{\beta}\right)^{-1}\right\| \cdot \left\|\left(\mathrm{E}^{-1} + \beta^{\intercal}\Gamma^{-1}\beta\right)^{-1}\right\|.\end{aligned}$$

Note that for any vector v such that $\|v\| = 1$, by the definition of operator norm, we have

$$v^{\intercal}\beta^{\intercal}\Gamma^{-1}\beta v \geqslant \lambda_{\min}(\Gamma^{-1}) v^{\intercal}\beta^{\intercal}\beta v \geqslant \lambda_{\min}(\Gamma^{-1})\lambda_{\min}(\beta^{\intercal}\beta).$$

It then follows that

$$\lambda_{\min}(\beta^{\intercal}\Gamma^{-1}\beta) \geqslant \lambda_{\min}(\Gamma^{-1})\lambda_{\min}(\beta^{\intercal}\beta).$$

On the other hand, by Assumption 3, we have

$$\frac{1}{d} v^{\intercal}\beta^{\intercal}\beta v = v^{\intercal}Bv - v^{\intercal}(B - \frac{1}{d}\beta^{\intercal}\beta)v \geqslant \lambda_{\min}(B) - \left\|\frac{1}{d}\beta^{\intercal}\beta - B\right\| > C,$$

where C is some constant. Thus, $\lambda_{\min}(\beta^{\intercal}\beta) > Cd$. Therefore $\lambda_{\min}(\beta^{\intercal}\Gamma^{-1}\beta) > Cd$, following from the fact that $\lambda_{\max}(\Gamma) \leqslant Km_d$. It then implies that $\lambda_{\min}(\mathrm{E}^{-1} + \beta^{\intercal}\Gamma^{-1}\beta) \geqslant$

$\lambda_{\min}(\beta^{\intercal}\Gamma^{-1}\beta) > Cm_d^{-1}d.$

$$\left\| \left(\mathrm{E}^{-1} + \beta^{\intercal}\Gamma^{-1}\beta \right)^{-1} \right\| = O_p(m_d d^{-1}).$$

Using (1.26) and (1.27) , we have

$$L_6 \leqslant Cm_d^3 n_\delta^{-(1-q)/2} (\log d)^{(1-q)/2}.$$

Finally, combining these results, we can obtain, for some constant $C > 0$,

$$\left\| (\widehat{\Sigma}_{\text{TSR}})^{-1} - \Sigma^{-1} \right\| \leqslant C \left(m_d^3 n_\delta^{-(1-q)/2} (\log d)^{(1-q)/2} + m_d^2 n_\delta^{-(1-q)/2} (\log d)^{(1-q)/2} \right).$$

We find that the second term on the right is dominated by the first one, then replace the whole above equation by the first term, which yields the desired result.

To prove the second statement, note that for any vector v such that $\|v\| = 1$, we have

$$v^{\intercal}\widehat{\Sigma}_{\text{TSR}}v = v^{\intercal}\widehat{\beta}\widehat{\mathrm{E}}\widehat{\beta}^{\intercal}v + v^{\intercal}\widehat{\Gamma}^S v \geqslant \lambda_{\min}\left(\widehat{\Gamma}^S\right),$$

which implies that

$$\lambda_{\min}\left(\widehat{\Sigma}_{\text{TSR}}\right) \geqslant \lambda_{\min}\left(\widehat{\Gamma}^S\right).$$

This inequality, combining with (1.23) of Lemma 3, concludes the proof. ∎

1.8.2 Proof of Theorem 2

Proof of Theorem 2 follows the same arguments as that of Theorem 3.

1.8.3 Proof of Theorem 3

We note that

$$\check{X}^\star - \bar{X}^\star = (\beta^\intercal \beta)^{-1} \beta^\intercal \bar{Z}^\star.$$

Similar to the proof of 1.11-1.13, and by Hoeffding inequality, we have

$$\begin{aligned}
\|\beta^\intercal \bar{Z}^\star\| =& \sqrt{\|\beta^\intercal \bar{Z}^\star \bar{Z}^{\star\intercal} \beta\|} \leqslant \sqrt{\|\beta^\intercal (\bar{Z}^\star \bar{Z}^{\star\intercal} - \Gamma)\beta\| + \|\beta^\intercal \Gamma \beta\|} \\
\leqslant& \sqrt{\|\beta^\intercal\| \|\bar{Z}^\star \bar{Z}^{\star\intercal} - \Gamma\| \|\beta\| + \|\beta^\intercal\| \|\Gamma\| \|\beta\|} \\
\leqslant& C \sqrt{d m_d n_\delta^{-(1-q)/2} (\log d)^{(1-q)/2} + d m_d} \\
\leqslant& C d^{1/2} m_d^{1/2} n_\delta^{-(1-q)/4} (\log d)^{(1-q)/4} + d^{1/2} m_d^{1/2},
\end{aligned}$$

where we use $\|\Gamma\| \leqslant m_d$, and $\|\bar{Z}^\star \bar{Z}^{\star\intercal} - \Gamma\| \overset{p}{=} O_p(n_\delta \sqrt{\log d})$.

Then we have

$$\begin{aligned}
\left\|\check{X}^\star - \bar{X}^\star\right\| \leqslant& \|(\beta^\intercal \beta)^{-1}\| \|\beta^\intercal \bar{Z}^\star\| \leqslant \frac{\|\beta^\intercal \bar{Z}^\star\|}{\lambda_{\min}(\beta^\intercal \beta)} \\
\leqslant& \frac{\|\beta^\intercal \bar{Z}^\star\| d^{-1}}{\lambda_{\min}(d^{-1} \beta^\intercal \beta)} \leqslant C d^{-1/2} m_d^{1/2} n_\delta^{-(1-q)/4} (\log d)^{(1-q)/4} + d^{-1/2} m_d^{1/2}.
\end{aligned}$$

Moreover, we note that

$$\begin{aligned}
\max_{1 \leqslant k \leqslant d} \sum_{i=0}^{n-k_n+1} \left(\sum_{l=1}^{r} \beta_{k,l} (\widehat{X}^\star_{l,i} - \bar{X}^\star_{l,i}) \right)^2 \leqslant& \max_{1 \leqslant k \leqslant d, 1 \leqslant i \leqslant n} \|\beta_k\|^2 \|\check{X}^\star - \bar{X}^\star\|_{\mathrm{F}}^2 \\
\leqslant& \max_{1 \leqslant k \leqslant d, 1 \leqslant i \leqslant n} \|\beta_k\|^2 r \|\check{X}^\star - \bar{X}^\star\|^2, \\
\leqslant& C d^{-1} m_d n_\delta^{-(1-q)/2} (\log d)^{(1-q)/2} + d^{-1} m_d.
\end{aligned}$$

Using these estimates, we obtain

$$\begin{aligned}
&\max_{1\leqslant k,l\leqslant d}\left|\sum_{i=0}^{n-k_n+1}\Big[(\bar{Y}^\star_{k,i}-(\beta\widehat{X}^\star_i)_k)(\bar{Y}^\star_{l,i}-(\beta\widehat{X}^\star_i)_l)-\bar{Z}^\star_{o,ki}\bar{Z}^\star_{o,li}\Big]\right|\\
\leqslant&\max_{1\leqslant k,l\leqslant d}\left|\sum_{i=0}^{n-k_n+1}(\beta(\check{X}^\star-\bar{X}^\star_i))_k(\beta(\check{X}^\star-\bar{X}^\star_i))_l\right|+2\max_{1\leqslant l,k\leqslant d}\left|\sum_{i=0}^{n-k_n+1}\bar{Z}^\star_{o,ki}(\beta(\check{X}^\star-\bar{X}^\star_i))_l\right|\\
\leqslant&\max_{1\leqslant k\leqslant d}\sum_{i=0}^{n-k_n+1}(\beta(\check{X}^\star-\bar{X}^\star_i))_k^2+2\sqrt{\max_{1\leqslant k\leqslant d}\sum_{i=0}^{n-k_n+1}(\bar{Z}^\star_{o,ki})^2\max_{1\leqslant k\leqslant d}\sum_{i=0}^{n-k_n+1}(\beta(\check{X}^\star-\bar{X}^\star_i))_k^2}\\
\leqslant&Cd^{-1/2}m_d^{1/2}n_\delta^{-(1-q)/4}(\log d)^{(1-q)/4}+d^{-1/2}m_d^{1/2}.
\end{aligned}\tag{1.31}$$

Therefore, according to (1.31) and (1.14), and using triangle inequality, we have

$$\begin{aligned}
&\max_{1\leqslant l,k\leqslant d}\left|\widehat{\Gamma}'_{kl}-\Gamma_{kl}\right|\\
\leqslant&\max_{1\leqslant k\leqslant r,1\leqslant l\leqslant d}\left|\frac{n}{n-k_n+2}\frac{1}{\psi_2k_n}\sum_{i=0}^{n-k_n+1}\bar{Z}^\star_{k,i}\bar{Z}^\star_{l,i}-\int_0^t g_{s,lk}ds\right|\\
&+\max_{1\leqslant k,l\leqslant d}\left|\frac{n}{n-k_n+2}\frac{1}{\psi_2k_n}\sum_{i=0}^{n-k_n+1}\Big[\left(\bar{Y}^\star_{k,i}-(\beta\check{X}^\star_i)_k\right)\left(\bar{Y}^\star_{l,i}-(\beta\check{X}^\star_i)_l\right)-\bar{Z}^\star_{o,ki}\bar{Z}^\star_{o,li}\Big]\right|\\
\leqslant&C\left(n_\delta^{-1/2}\sqrt{\log d}+Cd^{-1/2}m_d^{1/2}n_\delta^{-(1-q)/4}(\log d)^{(1-q)/4}+d^{-1/2}m_d^{1/2}\right)\\
\leqslant&C\left(n_\delta^{-1/2}\sqrt{\log d}+d^{-1/2}m_d^{1/2}\right).
\end{aligned}$$

The rest steps are similar to the TSR case. This concludes the proof.

1.8.4 Proof of Theorem 4

Proposition 1. *Suppose that Assumptions 1 - 4 hold. Also, assume that* $\|\mathrm{E}\|_{\mathrm{MAX}}\leqslant K$, $\|\Gamma\|_{\mathrm{MAX}}\leqslant K$ *almost surely for some constant* K, $n_\delta^{-1/2}\sqrt{\log d}=o(1)$, *and* $d^{-1/2}m_d=o(1)$. *Then* r, $\beta\mathrm{E}\beta^\intercal$, *and* Γ *can be identified as* $d\to\infty$. *That is,* $\bar{r}=r$, *if* d *is sufficiently large.*

Moreover, we have

$$\left\| \sum_{j=1}^{\bar{r}} \lambda_j \xi_j \xi_j^\intercal - \beta \mathrm{E} \beta^\intercal \right\|_{\mathrm{MAX}} \leqslant C d^{-1/2} m_d, \quad \textit{and}$$
$$\left\| \sum_{j=\bar{r}+1}^{d} \lambda_j \xi_j \xi_j^\intercal - \Gamma \right\|_{\mathrm{MAX}} \leqslant C d^{-1/2} m_d,$$

where $\{\lambda_j, 1 \leqslant j \leqslant d\}$ *and* $\{\xi_j, 1 \leqslant j \leqslant d\}$ *are the eigenvalues and their corresponding eigenvectors of* Σ*, and* $\bar{r} = \arg\min_{1 \leqslant j \leqslant d}(\frac{\lambda_j}{d} + j d^{-1/2} m_d) - 1$.

Lemma 4. *Suppose Assumptions 1 - 4 hold, and* $n_\delta^{-1/2}\sqrt{\log d} = o(1)$*, then we have*

$$\max_{1 \leqslant k \leqslant r, 1 \leqslant l \leqslant d} \left| \frac{n}{n-k_n+2} \frac{1}{\psi_2 k_n} \sum_{i=0}^{n-k_n+1} \bar{X}_i^k \bar{X}_i^l - \int_0^t e_{s,kl} ds \right| = O_p(n_\delta^{-1/2}\sqrt{\log d}), \tag{1.32}$$

$$\max_{1 \leqslant k \leqslant r, 1 \leqslant l \leqslant d} \left| \frac{n}{n-k_n+2} \frac{1}{\psi_2 k_n} \sum_{i=0}^{n-k_n+1} \bar{X}_i^k \bar{Z}_{u,li}^\star \right| = O_p(n_\delta^{-1/2}\sqrt{\log d}), \tag{1.33}$$

$$\max_{1 \leqslant k \leqslant r, 1 \leqslant l \leqslant d} \left| \frac{n}{n-k_n+2} \frac{1}{\psi_2 k_n} \sum_{i=0}^{n-k_n+1} \bar{Z}_{u,ki}^\star \bar{Z}_{u,li}^\star - \int_0^t g_{s,kl} ds \right| = O_p(n_\delta^{-1/2}\sqrt{\log d}). \tag{1.34}$$

Recall that

$$\Lambda = \mathrm{Diag}\left(\widehat{\lambda}_1, \widehat{\lambda}_2, \ldots, \widehat{\lambda}_r\right), \quad F = d^{1/2}\left(\widehat{\xi}_1, \widehat{\xi}_2, \ldots, \widehat{\xi}_r\right), \quad \text{and} \quad G = d^{-1} F^\intercal \mathcal{Y}.$$

We write

$$H = \frac{n}{n-k_n+2} \frac{1}{\psi_2 k_n t} \bar{\mathcal{X}}^\star \bar{\mathcal{X}}^{\star\intercal} \beta^\intercal F \Lambda^{-1}.$$

It is easy to verify that

$$\widehat{\Sigma} F = F\Lambda, \quad GG^\intercal = t d^{-1} \times \Lambda, \quad F^\intercal F = d \times \mathbb{I}_r, \quad \text{and}$$

$$\widehat{\Gamma} = \frac{1}{t}\left(\mathcal{Y} - FG\right)\left(\mathcal{Y} - FG\right)^{\intercal} = \frac{1}{t}\mathcal{Y}\mathcal{Y}^{\intercal} - \frac{1}{d}F\Lambda F^{\intercal}.$$

Lemma 5. *Suppose Assumptions 1 - 4 hold with* $\lambda_{ij} = O(n_{\delta}^{-1/2}\sqrt{\log d})$. *Suppose* $d^{-1/2}m_d = o(1)$, $n_{\delta}^{-1/2}\sqrt{\log d} = o(1)$, *and* $\widehat{r} \to r$ *with probability approaching 1, then there exists a* $r \times r$ *matrix* H, *such that with probability approaching 1,* H *is invertible,* $\|HH^{\intercal} - \mathbb{I}_r\| = \|H^{\intercal}H - \mathbb{I}_r\| = o_p(1)$, *and more importantly,*

$$\begin{aligned}\|F - \beta H\|_{\mathrm{MAX}} &= O_p\left(n_{\delta}^{-1/2}\sqrt{\log d} + d^{-1/2}m_d\right), \\ \left\|G - H^{-1}\bar{\mathcal{X}}\right\| &= O_p\left(n_{\delta}^{-1/2}\sqrt{\log d} + d^{-1/2}m_d\right).\end{aligned}$$

Lemma 6. *Under Assumptions 1 - 4 ,* $d^{-1/2}m_d = o(1)$, *and* $n_{\delta}^{-1/2}\sqrt{\log d} = o(1)$, *we have*

$$\|F - \beta H\|_{\mathrm{MAX}} = O_p\left(n_{\delta}^{-1/2}\sqrt{\log d} + d^{-1/2}m_d\right). \tag{1.35}$$

$$\left\|H^{-1}\right\| = O_p(1). \tag{1.36}$$

$$\left\|G - H^{-1}\bar{\mathcal{X}}\right\| = O_p\left(n_{\delta}^{-1/2}\sqrt{\log d} + d^{-1/2}m_d\right). \tag{1.37}$$

Lemma 7. *Under Assumptions 1 - 4 ,* $d^{-1/2}m_d = o(1)$, *and* $n_{\delta}^{-1/2}\sqrt{\log d} = o(1)$, *we have*

$$\left\|\widetilde{\Gamma}^{S} - \Gamma\right\|_{\mathrm{MAX}} \leqslant \left\|\widehat{\Gamma} - \Gamma\right\|_{\mathrm{MAX}} = O_p\left(n_{\delta}^{-1/2}\sqrt{\log d} + d^{-1/2}m_d\right). \tag{1.38}$$

Lemma 8. *Under Assumptions 1 - 4,* $d^{-1/2}m_d = o(1)$, *and* $n_{\delta}^{-1}\log d = o(1)$, *we have*

$$\left\|\frac{1}{t}FGG^{\intercal}F^{\intercal} - \beta \mathrm{E}\beta^{\intercal}\right\|_{\mathrm{MAX}} = O_p\left(n_{\delta}^{-1/2}\sqrt{\log d} + d^{-1/2}m_d\right).$$

Proof of Proposition 1, Lemma 4-8. The proofs follow the same arguments as in Aït-Sahalia and Xiu (2017), thus we omit the details.

∎

Lemma 9. *Under Assumptions 1 - 4,* $d(n_\delta^{-1/2}\sqrt{\log d})^2 = o(1)$*,* $n^{\delta-1/2}\log d = o(1)$*, and* $d^{-1/2}m_d = o(1)$*, we have*

$$(i) \qquad \|F-\beta H\|_{\mathrm{F}}^2 = O_p\left(d(n_\delta^{-1/2}\sqrt{\log d})^2 + m_d^2\right). \tag{1.39}$$

$$(ii) \qquad \|(F-\beta H)(F-\beta H)^\intercal\|_{\mathrm{F}}^2 = O_p\left(dn^{2\delta-1}\log^2 d + d^{-1}m_d^4\right). \tag{1.40}$$

$$(iii) \qquad \|\beta H(F-\beta H)^\intercal\|_{\Sigma}^2 = O_p\left(n^{\delta-1/2}\log d + d^{-1}m_d^2\right). \tag{1.41}$$

$$(iv) \qquad \|\beta(H^\intercal H - I_r)\beta^\intercal\|_{\Sigma}^2 = o_p(1). \tag{1.42}$$

Proof of Lemma 9. (i) We have $\|F-\beta H\|_{\mathrm{F}}^2 \leqslant d\,\|F-\beta H\|_{\mathrm{MAX}}^2$.

(ii) According to the definition of $\|\cdot\|_\Sigma$, we have

$$\|(F-\beta H)(F-\beta H)^\intercal\|_{\Sigma}^2 = O_p(\frac{1}{d}\|F-\beta H\|_{\mathrm{F}}^4) = O_p(d\,\|F-\beta H\|_{\mathrm{MAX}}^4).$$

(iii) By $\left\|\beta^\intercal\Sigma^{-1}\beta\right\| = O(1)$, We have

$$\begin{aligned}\|\beta H(F-\beta H)^\intercal\|_{\Sigma}^2 =& \frac{1}{d}tr(H(F-\beta H)^\intercal\Sigma^{-1}(F-\beta H)H\beta^\intercal\Sigma^{-1}\beta) \\ \leqslant& \frac{1}{d}\,\|H\|^2\left\|\beta^\intercal\Sigma^{-1}\beta\right\|\left\|\Sigma^{-1}\right\|\|F-\beta H\|_{\mathrm{F}}^2 \\ =& O_p(\|F-\beta H\|_{\mathrm{MAX}}^2).\end{aligned}$$

(iv) by $\left\|\beta^\intercal\Sigma^{-1}\beta\right\| = O(1)$, We have

$$\begin{aligned}\|\beta(H^\intercal H - I_r)\beta^\intercal\|_{\Sigma}^2 \leqslant& \frac{1}{d}tr\{(H^\intercal H - I_r)\beta^\intercal\Sigma^{-1}\beta(H^\intercal H - I_r)\beta^\intercal\Sigma^{-1}\beta\} \\ \leqslant& \frac{1}{d}\left\|\beta^\intercal\Sigma^{-1}\beta\right\|^2\|H^\intercal H - I_r\|_{\mathrm{F}}^2 \\ =& o_p(1).\end{aligned}$$

∎

Lemma 10. *Under Assumptions 1 - 4,* $d^{-1/2}m_d = o(1)$, *and* $n_\delta^{-1/2}\sqrt{\log d} = o(1)$, *we have*

$$\left\|\widetilde{\Gamma}^S - \Gamma\right\| = O_p\left(m_d(n_\delta^{-1/2}\sqrt{\log d} + d^{-1/2}m_d)^{1-q}\right). \tag{1.43}$$

Moreover, if in addition, $d^{-1/2}m_d^2 = o(1)$ *and* $m_d n_\delta^{-1/2}\sqrt{\log d} = o(1)$ *hold, then* $\lambda_{\min}\left(\widehat{\Gamma}^S\right)$ *is bounded away from 0 with probability approaching 1, and*

$$\left\|\left(\widetilde{\Gamma}^S\right)^{-1} - \Gamma^{-1}\right\| = O_p\left(m_d(n_\delta^{-1/2}\sqrt{\log d} + d^{-1/2}m_d)^{1-q}\right). \tag{1.44}$$

Proof of Lemma 10. Note that since $\widehat{\Gamma}^S - \Gamma$ is symmetric,

$$\left\|\widetilde{\Gamma}^S - \Gamma\right\| \leqslant \left\|\widetilde{\Gamma}^S - \Gamma\right\|_\infty = \max_{1\leqslant l\leqslant d}\sum_{k=1}^{d}\left|\widehat{\Gamma}^S_{lk} - \Gamma_{lk}\right|.$$

By Lemma 7, and using the same technique as proving (1.20), we have

$$\left\|\widetilde{\Gamma}^S - \Gamma\right\| = O_p\left(m_d S^{-q}(n_\delta^{-1/2}\sqrt{\log d} + d^{-1/2}m_d) + m_d S^{1-q}\right).$$

Choosing $\lambda_{ij} = M'(n_\delta^{-1/2}\sqrt{\log d} + d^{-1/2}m_d)$, M' is some positive constant, we have

$$\left\|\widetilde{\Gamma}^S - \Gamma\right\| = O_p\left(m_d(n_\delta^{-1/2}\sqrt{\log d} + d^{-1/2}m_d)^{1-q}\right).$$

Moreover, since $\lambda_{\min}(\Gamma) > K$ for some constant K and by Weyl's inequality, we have $\lambda_{\min}(\widetilde{\Gamma}^S) > K - o_p(1)$. As a result, we have

$$\begin{aligned}\left\|\left(\widetilde{\Gamma}^S\right)^{-1} - \Gamma^{-1}\right\| =& \left\|\left(\widetilde{\Gamma}^S\right)^{-1}\left(\Gamma - \left(\widetilde{\Gamma}^S\right)\right)\Gamma^{-1}\right\| \leqslant \lambda_{\min}(\widetilde{\Gamma}^S)^{-1}\lambda_{\min}(\Gamma)^{-1}\left\|\Gamma - \widetilde{\Gamma}^S\right\| \\ \leqslant& O_p\left(m_d(n_\delta^{-1/2}\sqrt{\log d} + d^{-1/2}m_d)^{1-q}\right).\end{aligned}$$

∎

Proof of Theorem 4. Note that

$$\widehat{\Sigma}_{\mathrm{PCA}} = \frac{1}{d} F \Lambda F^{\intercal} + \widetilde{\Gamma}^{S} = \frac{1}{t} F G G^{\intercal} F^{\intercal} + \widetilde{\Gamma}^{S}.$$

By Lemma 7, we have

$$\left\| \widetilde{\Gamma}^{S} - \Gamma \right\|_{\mathrm{MAX}} = O_p\left(n_{\delta}^{-1/2} \sqrt{\log d} + d^{-1/2} m_d \right).$$

By the triangle inequality, we have

$$\left\| \widehat{\Sigma}_{\mathrm{PCA}} - \Sigma \right\|_{\mathrm{MAX}} \leqslant \left\| \frac{1}{d} F \Lambda F^{\intercal} - \beta \mathrm{E} \beta^{\intercal} \right\|_{\mathrm{MAX}} + \left\| \widetilde{\Gamma}^{S} - \Gamma \right\|_{\mathrm{MAX}}$$

Therefore, the desired result follows from Lemmas 7 and 8.

Using Lemma 9, for some constant C, we have

$$\begin{aligned} \|\widehat{\Sigma}_{\mathrm{PCA}} - \Sigma\|_{\Sigma}^2 \leqslant & C \Big[\|\beta(H^{\intercal} H - I_r)\beta^{\intercal}\|_{\Sigma}^2 + \|\beta H (F - \beta H)^{\intercal}\|_{\Sigma}^2 \\ & + \|(F - \beta H)(F - \beta H)^{\intercal}\|_{\Sigma}^2 + \left\| \widetilde{\Gamma}^{S} - \Gamma \right\| \Big] \\ = & O_p\big(d (n_{\delta}^{-1/2} \sqrt{\log d})^4 + \frac{1}{d} m_d^4 + m_d^2 (n_{\delta}^{-1/2} \sqrt{\log d} + d^{-1/2} m_d)^{2(1-q)}\big). \end{aligned}$$

For the inverse, firstly, by Lemma 10 and the fact that $\lambda_{\min}(\widehat{\Sigma}_{\mathrm{PCA}}) \geqslant \lambda_{\min}(\widetilde{\Gamma}^{S})$, we can establish the first two statements.

To bound $\left\| (\widehat{\Sigma}_{\mathrm{PCA}})^{-1} - \Sigma^{-1} \right\|$, by the Sherman - Morrison - Woodbury formula, we have

$$\begin{aligned} & \left(\widehat{\Sigma}_{\mathrm{PCA}} \right)^{-1} - \left(\widetilde{\Sigma} \right)^{-1} \\ = & \left(t^{-1} F G G^{\intercal} F^{\intercal} + \widetilde{\Gamma}^{S} \right)^{-1} - \left(t^{-1} \beta H H^{-1} \bar{\mathcal{X}}^{\star} \bar{\mathcal{X}}^{\star \intercal} (H^{-1})^{\intercal} H^{\intercal} \beta^{\intercal} + \Gamma \right)^{-1} \\ = & \left((\widetilde{\Gamma}^{S})^{-1} - \Gamma^{-1} \right) - \left((\widetilde{\Gamma}^{S})^{-1} - \Gamma^{-1} \right) F \left(d \Lambda^{-1} + F^{\intercal} (\widetilde{\Gamma}^{S})^{-1} F \right)^{-1} F^{\intercal} (\widetilde{\Gamma}^{S})^{-1} \end{aligned}$$

$$
\begin{aligned}
&- \Gamma^{-1} F \left(d\Lambda^{-1} + F^{\intercal} (\widetilde{\Gamma}^{S})^{-1} F \right)^{-1} F^{\intercal} \left((\widetilde{\Gamma}^{S})^{-1} - \Gamma^{-1} \right) \\
&+ \Gamma^{-1} (\beta H - F) \left(t H^{\intercal} \left(\bar{\mathcal{X}} \bar{\mathcal{X}}^{\intercal} \right)^{-1} H + H^{\intercal} \beta^{\intercal} \Gamma^{-1} \beta H \right)^{-1} H^{\intercal} \beta^{\intercal} \Gamma^{-1} \\
&- \Gamma^{-1} F \left(t H^{\intercal} \left(\bar{\mathcal{X}} \bar{\mathcal{X}}^{\intercal} \right)^{-1} H + H^{\intercal} \beta^{\intercal} \Gamma^{-1} \beta H \right)^{-1} (F^{\intercal} - H^{\intercal} \beta^{\intercal}) \Gamma^{-1} \\
&+ \Gamma^{-1} F \left(\left(t H^{\intercal} \left(\bar{\mathcal{X}} \bar{\mathcal{X}}^{\intercal} \right)^{-1} H + H^{\intercal} \beta^{\intercal} \Gamma^{-1} \beta H \right)^{-1} - \left(d\Lambda^{-1} + F^{\intercal} (\widetilde{\Gamma}^{S})^{-1} F \right)^{-1} \right) F^{\intercal} \Gamma^{-1} \\
=& L_1 + L_2 + L_3 + L_4 + L_5 + L_6.
\end{aligned}
$$

By Lemma 10, we have

$$
\|L_1\| = O_p \left(m_d (n_\delta^{-1/2} \sqrt{\log d} + d^{-1/2} m_d)^{1-q} \right).
$$

For L_2, because $\|F\| = O_p(d^{1/2})$, $\lambda_{\max} \left((\widetilde{\Gamma}^{S})^{-1} \right) \leqslant \left(\lambda_{\min} (\widetilde{\Gamma}^{S}) \right)^{-1} \leqslant K + o_p(1)$,

$$
\lambda_{\min} \left(d\Lambda^{-1} + F^{\intercal} (\widetilde{\Gamma}^{S})^{-1} F \right) \geqslant \lambda_{\min} \left(F^{\intercal} (\widetilde{\Gamma}^{S})^{-1} F \right) \geqslant \lambda_{\min} \left(F^{\intercal} F \right) \lambda_{\min} \left((\widetilde{\Gamma}^{S})^{-1} \right) \geqslant m_d^{-1} d,
$$

and by Lemma 10, we have

$$
\begin{aligned}
\|L_2\| &\leqslant \left\| \left((\widetilde{\Gamma}^{S})^{-1} - \Gamma^{-1} \right) \right\| \|F\| \left\| \left(d\Lambda^{-1} + F^{\intercal} (\widetilde{\Gamma}^{S})^{-1} F \right)^{-1} \right\| \left\| F^{\intercal} (\widetilde{\Gamma}^{S})^{-1} \right\| \\
&= O_p \left(m_d (n_\delta^{-1/2} \sqrt{\log d} + d^{-1/2} m_d)^{1-q} \right).
\end{aligned}
$$

The same bound holds for $\|L_3\|$. As for L_4, note that $\|\beta\| = O_p(d^{1/2})$, $\left\| \Gamma^{-1} \right\| \leqslant (\lambda_{\min}(\Gamma))^{-1} \leqslant K$, $\|H\| = O_p(1)$, and $\|\beta H - F\| \leqslant \sqrt{rd} \, \|\beta H - F\|_{\mathrm{MAX}} = O_p(n_\delta^{-1/2} d^{2/\nu + 1/2} + m_d)$, and that

$$
\begin{aligned}
\lambda_{\min} \left(t H^{\intercal} \left(\bar{\mathcal{X}} \bar{\mathcal{X}}^{\intercal} \right)^{-1} H + H^{\intercal} \beta^{\intercal} \Gamma^{-1} \beta H \right) &\geqslant \lambda_{\min} \left(H^{\intercal} \beta^{\intercal} \Gamma^{-1} \beta H \right) \\
&\geqslant \lambda_{\min}(\Gamma^{-1}) \lambda_{\min}(\beta^{\intercal} \beta) \lambda_{\min}(H^{\intercal} H) \\
&> K m_d^{-1} d,
\end{aligned}
$$

hence we have

$$\begin{aligned}\|L_4\| \leqslant& \left\|\Gamma^{-1}\right\| \|(\beta H - F)\| \left\|\left(tH^\intercal \left(\bar{\mathcal{X}}^\star \bar{\mathcal{X}}^{\star\intercal}\right)^{-1} H + H^\intercal \beta^\intercal \Gamma^{-1} \beta H\right)^{-1}\right\| \|H^\intercal \beta^\intercal\| \left\|\Gamma^{-1}\right\| \\ =& O_p(m_d n_\delta^{-1/2} \sqrt{\log d} + d^{-1/2} m_d^2).\end{aligned}$$

The same bound holds for L_5. Finally, with respect to L_6, we have

$$\begin{aligned}&\left\|\left(tH^\intercal \left(\bar{\mathcal{X}} \bar{\mathcal{X}}^\intercal\right)^{-1} H + H^\intercal \beta^\intercal \Gamma^{-1} \beta H\right)^{-1} - \left(d\Lambda^{-1} + F^\intercal (\widetilde{\Gamma}^S)^{-1} F\right)^{-1}\right\| \\ \leqslant& K d^{-2} m_d^2 \left\|\left(tH^\intercal \left(\bar{\mathcal{X}} \bar{\mathcal{X}}^\intercal\right)^{-1} H + H^\intercal \beta^\intercal \Gamma^{-1} \beta H\right) - \left(d\Lambda^{-1} + F^\intercal (\widetilde{\Gamma}^S)^{-1} F\right)\right\|.\end{aligned}$$

Moreover, since we have

$$\begin{aligned}\left\|tH^\intercal (\bar{\mathcal{X}} \bar{\mathcal{X}}^\intercal)^{-1} H - d\Lambda^{-1}\right\| &= \left\|\Lambda^{-1} F^\intercal (\beta H - F)\right\| \\ &= O_p\left(n_\delta^{-1/2} \sqrt{\log d} + d^{-1/2} m_d\right),\end{aligned}$$

and

$$\begin{aligned}&\left\|H^\intercal \beta^\intercal \Gamma^{-1} \beta H - F^\intercal (\widetilde{\Gamma}^S)^{-1} F\right\| \\ \leqslant& \left\|(H^\intercal \beta^\intercal - F^\intercal) \Gamma^{-1} \beta H\right\| + \left\|F^\intercal \Gamma^{-1} (\beta H - F)\right\| + \left\|F^\intercal \left(\Gamma^{-1} - (\widetilde{\Gamma}^S)^{-1}\right) F\right\| \\ =& O_p\left(d m_d (n_\delta^{-1/2} \sqrt{\log d} + d^{-1/2} m_d)^{1-q}\right).\end{aligned}$$

Combining these inequalities yields

$$\|L_6\| = O_p\left(m_d^3 (n_\delta^{-1/2} \sqrt{\log d} + d^{-1/2} m_d)^{1-q}\right).$$

On the other hand, using the Sherman - Morrison - Woodbury formula again,

$$\begin{aligned}
&\left\|\widetilde{\Sigma}^{-1}-\Sigma^{-1}\right\| \\
=&\left\|\left(t^{-1}\beta\bar{\mathcal{X}}\bar{\mathcal{X}}^{\intercal}\beta^{\intercal}+\Gamma\right)^{-1}-\left(\beta\mathrm{E}\beta^{\intercal}+\Gamma\right)^{-1}\right\| \\
\leqslant&\left\|\Gamma^{-1}\right\|^{2}\|\beta H\|^{2}\left\|\left(\left(tH^{\intercal}\left(\bar{\mathcal{X}}\bar{\mathcal{X}}^{\intercal}\right)^{-1}H+H^{\intercal}\beta^{\intercal}\Gamma^{-1}\beta H\right)^{-1}-\left(H^{\intercal}\mathrm{E}^{-1}H+H^{\intercal}\beta^{\intercal}\Gamma^{-1}\beta H\right)^{-1}\right)\right\| \\
\leqslant&Kd\left\|tH^{\intercal}\left(\bar{\mathcal{X}}\bar{\mathcal{X}}^{\intercal}\right)^{-1}H+H^{\intercal}\beta^{\intercal}\Gamma^{-1}\beta H\right\|^{-1}\left\|H^{\intercal}\mathrm{E}^{-1}H+H^{\intercal}\beta^{\intercal}\Gamma^{-1}\beta H\right\|^{-1}\left\|t\left(\bar{\mathcal{X}}\bar{\mathcal{X}}^{\intercal}\right)^{-1}-\mathrm{E}^{-1}\right\| \\
=&O_p\left(m_d n_{\delta}^{-1/2}\sqrt{\log d}\right).
\end{aligned}$$

By the triangle inequality, we obtain

$$\begin{aligned}
\left\|(\widehat{\Sigma}_{\text{PCA}})^{-1}-\Sigma^{-1}\right\| \leqslant& \left\|(\widehat{\Sigma}_{\text{PCA}})^{-1}-\widetilde{\Sigma}^{-1}\right\|+\left\|\widetilde{\Sigma}^{-1}-\Sigma^{-1}\right\| \\
=&O_p\left(m_d^3(n_{\delta}^{-1/2}\sqrt{\log d}+d^{-1/2}m_d)^{1-q}\right).
\end{aligned}$$

This concludes the proof. ∎

Chapter 2

Large Scale Realized Volatility Forecasting with Machine Learning

2.1 Introduction

Realized volatility forecasting is crucial in asset pricing and risk management and plays important roles for financial market practitioners and regulators. Despite the existence of many GARCH and stochastic volatility formulations in the literature, their performance is typically less competitive with high-frequency intraday data compared to the reduced-form forecasting models (Andersen, Bollerslev, Diebold, and Labys (2003)), due to the complication of the latent nature.

We develop realized volatility forecasting models by incorporating features from existing literature. These include the mixed data sampling (MIDAS) approach by Ghysels, Santa-Clara, and Valkanov (2006), the heterogeneous autoregressive (HAR) model by Corsi (2009), the semivariance-HAR model by Patton and Sheppard (2015), the HARQ model by Bollerslev, Patton, and Quaedvlieg (2016) that accounts for measurement error, and the heterogeneous exponential realized volatility with global risk factor (HExpGl) model by Bollerslev, Hood, Huss, and Pedersen (2018). These models utilize the weighted average of past RVs and sometimes combine realized quarticity with RV to predict the future RV, demonstrating decent out-of-sample forecasting performance.

Additionally, we explore non-RV based features that may contain incremental information for forecasting. These include implied volatilities derived from option prices of underlying assets (Christensen and Prabhala (1998), Busch, Christensen, and Nielsen (2011)), earnings announcement dates (Cao and Narayanamoorthy (2012), Barth and So (2014),Atilgan (2014),Lei, Wang, and Yan (2020)), trading volume (Liu, Choo, Lee, and Lee (2023)),and returns and overnight returns (Ahoniemi and Lanne (2013), Todorova and Souček (2014)). We find that incorporating these non-RV based features significantly enhances out-of-sample forecasting performance.

Our dataset comprises two extensive sets of stocks: all S&P 500 index constituents from 1996 to 2022 and all stocks and ETFs traded in the major United States stock exchanges over the same period (referred to as US stocks). To the best of our knowledge, the latter represents the largest empirical experiment conducted in the volatility forecasting literature to date.

This paper investigates several machine learning algorithms for forecasting the realized volatility of stocks by utilizing this large set of features, including LASSO (Tibshirani (1996)), Principal Component Regression, Random Forest (Leo (2001)), Gradient Boosted Regression Tree (Freund and Schapire (1995)), and neural networks (LeCun, Bengio, and Hinton (2015)). Machine learning algorithms have demonstrated superior performance in various areas, such as complex games like Go (Silver, Hubert, Schrittwieser, Antonoglou, Lai, Guez, Lanctot, Sifre, Kumaran, Graepel, et al. (2018)), physical control (Gu, Holly, Lillicrap, and Levine (2017)), and artificial intelligence-generated content models like ChatGPT (Schulman, Zoph, Kim, Hilton, Menick, Weng, Uribe, Fedus, Metz, Pokorny, et al. (2022)). Finance is no exception, with successful applications of machine learning algorithms in empirical asset pricing (Gu, Kelly, and Xiu (2020)), fraud detection (Ravisankar, Ravi, Rao, and Bose (2011)), solving dynamic equilibrium models (Scheidegger and Bilionis (2019)), and, of course, realized volatility forecasting (Li and Tang (2022), Zhang, Zhang, Cucuringu, and Qian (2024)). Notably, machine learning algorithms, particularly neural networks, outperform traditional ordinary least squares (OLS) based methods significantly, as measured by out-of-sample R-squares, mean squared errors, and quasi-likelihood. This holds true for both S&P 500 index constituents and all US stocks.

We observe that the OLS-based method performs much better when we pool all the data together and fit a universal model for all stocks, as opposed to fitting individual models for each stock. As we increase the number of features, the performance of individual models deteriorates, whereas for the pooled model, the opposite is true, with all features combined producing the best OLS model. In fact, even the worst-performing pooled model outperforms the best individual model in all metrics, even after correcting extreme predictions in the individual models via the insanity filter (Swanson and White (1997)). Such a filter is a must for the individual model and improves performance significantly, but it is rarely triggered for the pooled model and has a negligible effect on performance.

We evaluate the relative out-of-sample performance of models by assigning a value of 0 to a naive random walk model and a value of 100 to the pooled HAR model, based on the mean squared error. A higher value indicates better performance. On the set of S&P 500 stocks, the individual HAR model without the insanity filter achieves a score of 65.66. However, when incorporating the insanity filter correction, the score increases to 95.15. Parsimonious regression forms like HARQ achieve a performance value of 106.87. Additionally, the regression form OLSRV, which includes all RV-based features, scores 108.84. By adding implied volatility as additional predictors in the OLSRVIV specification, the performance score increases to 114.70. Surprisingly, by further incorporating non-RV and non-IV based features into the OLSRV specification, OLSVPOS achieves a score of 131.57. Finally, utilizing all available features in the OLSALL specification results in the best performing pooled OLS model with a score of 134.73. When using the same set of features, a neural network with

4 hidden layers achieves a score of 136.47, while taking an ensemble of 5 neural networks with different random seeds pushes the limit to 137.62. This indicates the ability of neural networks to capture the nonlinear dependencies of the features in predicting future RV.

For the set of US stocks, without the insanity filter, the individual HAR model performs very poorly, scoring -282.75, indicating inferior performance compared to even the simple random walk model. With the insanity filter incorporated, the individual HAR model barely scores 19.16, still falling short compared to the pooled HAR model's score of 100. HARQ and OLSRV achieve scores of 103.12 and 104.87, respectively, while OLSRVIV boosts the score to 105.77. OLSVPOS achieves a significantly higher score of 116.34, which is further increased to 116.89 by adding IV-based features. When using the same set of features as OLSALL, the neural network with 4 hidden layers achieves scores above 122.11, while the ensemble version reaches 122.97.

Due to computational constraints, we did not fit individual models for the machine learning algorithms. It is highly likely that the individual models would have suffered from the over-fitting issue, resulting in poor out-of-sample forecasting. However, we did not observe the need for an insanity filter when using the pooled data. We found that Lasso and principal component regression did not outperform the OLS model with all features. This may be because all features are important when dealing with such a large dataset, and these methods could not capture the nonlinear dependencies as effectively as neural networks. We also observed that tree-based methods, especially GBRT, tended to underperform other machine learning-based methods on average, but they produced the fewest extreme values and were more stable in that regard.

We employ the utility-based framework presented in Bollerslev, Hood, Huss, and Pedersen (2018), which relies on mean-variance preference and a constant Sharpe ratio, to quantify the economic gain of the volatility forecasting models constructed in this study. The more accurate the forecast, the higher the realized utility the investor would obtain. Under realistic assumptions of risk aversion and Sharpe ratio, the pooled models outperform the individual models by a large margin, even on the set of S&P 500 stocks, and the gap widens on the larger US set. On average, the neural network achieves a 44 basis point advantage over the benchmark HAR model with individual fit and 40 basis points over the pooled fit for the S&P 500 stocks, and 30 basis points over the pooled fit for the US stocks.

The rest of the paper is structured as follows. In Section 2.2, we set up the general framework of the problem and discuss the ordinary least squares based methods and machine learning methods. Section 2.3 details the data collection and cleaning process, the training scheme, and evaluation metrics. Section 2.4 provides the forecasting performance and discusses the empirical findings. Section 2.5 introduces the utility-based framework, quantifying the economic gain of the forecasting models. Finally, in Section 2.6, we conclude the paper.

2.2 Methodology

2.2.1 Problem Setup

We denote p_t as the log-price of an asset at time t, and assume that it follows a generic stochastic differential process

$$p_t = p_0 + \int_0^t \mu_s ds + \int_0^t \sigma_s dW_s \tag{2.1}$$

where μ_s represents the drift, σ_s represents the instantaneous volatility, W_s represents the standard Brownian motion. By normalizing the unit time interval to a day (Bollerslev, Patton, and Quaedvlieg (2016)), our aim is to forecast the latent Integrated Variance (IV) formally defined by,

$$IV_t \equiv \int_{t-1}^t \sigma_s^2 ds. \tag{2.2}$$

Due to its latent nature, IV is not directly observable, whereas the (daily) Realized Volatility (RV), defined by summation of high-frequency intraday squared log-returns, is observable:

$$RV_t \equiv RV_t^d = \sum_{i=1}^{M} [p_{t-1+i/M} - p_{t-1+(i-1)/M}]^2 = \sum_{i=1}^{M} r_{t,i}^2. \tag{2.3}$$

Here M is the daily sampling frequency, and $r_{t,i}$ is the intraday log-return. The RV_t provides a consistent estimator as the sampling frequency M increases (Barndorff-Nielsen and Shephard (2002)). We sometimes put a superscription d to RV_t^d to emphasis it is the daily RV, and omits it for the majority of the parts for simplicity. We further define the $t+1$ through $t+h$ average RV as

$$RV_{t+1|t+h} = \frac{1}{h}\sum_{j=1}^{h} RV_{t+j}. \tag{2.4}$$

There is an upper bound for the sampling frequency M due to data limitation and microstructure noise. The estimation error in RV is characterized by the asymptotic distribution theory of Barndorff-Nielsen and Shephard (2002), that

$$RV_t = IV_t + \eta_t, \eta_t \sim MN(0, \frac{2}{M} IQ_t), \quad IQ_t \equiv \int_{t-1}^{t} \sigma_s^4 ds,$$

where the Integrated Quarticity (IQ) may be consistently estimated by the Realized Quarticity (RQ),

$$RQ_t = \frac{M}{3}\sum_{i=1}^{M} r_{t,i}^4. \tag{2.5}$$

2.2.2 OLS Methods

We consider several classical reduced-form forecasting models that rely on RV features. The mixed data sampling (MIDAS) model, which was proposed by Ghysels, Santa-Clara, and

Valkanov (2006), will be discussed first. The model is specified as follows:

$$\text{MIDAS:} \quad RV_{t+1} = \beta_0 + \beta_m MIDAS_t + \epsilon_t, \tag{2.6}$$

where the $MIDAS_t$ term is a weighted averaged of past L days' RV, defined as

$$\begin{aligned} MIDAS_t &= \frac{1}{\sum_{j=1}^{L} a_j} \sum_{j=1}^{L} a_j RV_{t+1-j}, \\ a_j &= (\frac{j}{L})^{\theta_1 - 1}(1 - \frac{j}{L})^{\theta_2 - 1}, \quad j = 1, ..., L. \end{aligned}$$

MIDAS utilizes smooth lag polynomials to depict dynamic dependencies, with the a_j being the most commonly used specification, namely the scaled beta functions. It should be noted that the gamma terms in the beta function have been omitted as they will be cancelled out by normalization.

The long-memory Heterogeneous AR (HAR) model, proposed by Corsi (2009), is widely used to estimate RV and has become one of the preferred specifications for RV-based forecasting. This is due to its simplicity and superior performance compared to traditional GARCH and stochastic volatility models. The model defines the weekly, monthly, and quarterly RV as follows:

$$\begin{aligned} RV_t^w &= \frac{1}{5} \sum_{j=1}^{5} RV_{t+1-j} = RV_{t-4|t}, \\ RV_t^m &= \frac{1}{21} \sum_{j=1}^{21} RV_{t+1-j} = RV_{t-20|t}, \end{aligned}$$

$$RV_t^q = \frac{1}{63}\sum_{j=1}^{63} RV_{t+1-j} = RV_{t-62|t}.$$

The dynamic dependencies in the HAR model are limited to daily, weekly, monthly, and quarterly averages. The regression formula for the HAR model is as follows:

$$\text{HAR:} \quad RV_{t+1} = \beta_0 + \beta_d RV_t^d + \beta_w RV_t^w + \beta_m RV_t^m + \beta_q RV_t^q + \epsilon_t \tag{2.7}$$

A variant of the HAR model, proposed by Patton and Sheppard (2015), is known as Semivariance-HAR (SHAR). This model extends HAR by decomposing the daily RV into positive and negative semi-variances, RVP and RVN, where

$$\begin{aligned} RVP_t &= \sum_{i=1}^{M} r_{t,i}^2 \mathbf{1}_{\{r_{t,i}>0\}}, \\ RVN_t &= \sum_{i=1}^{M} r_{t,i}^2 \mathbf{1}_{\{r_{t,i}<0\}} = RV_t - RVP_t, \end{aligned}$$

arguing that the negative semi-variance has stronger predictive power, and thus extends the HAR model by incorporating this feature. The regression formula for SHAR may be written as follows:

$$\begin{aligned} \text{SHAR:} \quad RV_{t+1} = &\beta_0 + \beta_d^+ RVP_t + \beta_d^- (RV_t - RVP_t) \\ &+ \beta_w RV_t^w + \beta_m RV_t^m + \beta_q RV_t^q + \epsilon_t \end{aligned} \tag{2.8}$$

The HAR model is further extended by Bollerslev, Patton, and Quaedvlieg (2016) through the introduction of HARQ, which is a set of specifications that exploits the measurement error of RV forecasts. We will be considering their full specification, which allows all parameters to vary with an estimate of the measurement error variance:

$$\begin{aligned}\text{HARQ:}\quad RV_{t+1} =&\beta_0 + \beta_d RV_t^d + \beta_w RV_t^w + \beta_m RV_t^m + \beta_q RV_t^q + \phi_d RV_t^d \sqrt{RQ_t^d} \\ &+ \phi_w RV_t^w \sqrt{RQ_t^w} + \phi_m RV_t^m \sqrt{RQ_t^m} + \phi_q RV_t^q \sqrt{RQ_t^q} + \epsilon_t \end{aligned} \tag{2.9}$$

The Heterogeneous Exponential Realized Volatility with Global Risk Factor (HExpGl) model, proposed by Bollerslev, Hood, Huss, and Pedersen (2018), distinguishes itself from previous models by utilizing exponentially weighted moving averages (EWMA) of lagged RV instead of simple averages. Furthermore, the model incorporates a global risk factor that takes panel information into account. The EWMA of lagged RV is denoted as follows:

$$ExpRV_t^{CoM(\lambda)} = \frac{1}{\sum_{j=1}^{500} e^{-j\lambda}} \sum_{j=1}^{500} e^{-j\lambda} RV_{t+1-j},$$

here the center-of-mass is defined as $CoM(\lambda) \equiv \frac{e^{-\lambda}}{1-e^{-\lambda}}$, and Bollerslev, Hood, Huss, and Pedersen (2018) considers 4 values of CoM, 1, 5, 25, and 125, corresponding to $\lambda = \log(1 + \frac{1}{1})$,$\log(1+\frac{1}{5})$,$\log(1+\frac{1}{25})$, and $\log(1+\frac{1}{125})$. Denote K as the total number of stocks considered, $RV_{k,t}$ and $\overline{RV_{k,t}}$ as the RV and long-run mean of RV for stock k at time t. The EWMA of global risk factor GLRV is defined as

$$\begin{aligned} ExpGLRV_{k,t}^{CoM(\lambda)} &= \frac{1}{\sum_{j=1}^{500} e^{-j\lambda}} \sum_{j=1}^{500} e^{-j\lambda} GLRV_{k,t+1-j}, \\ GLRV_{k,t} &= (\frac{1}{K} \sum_{i=1}^{K} \frac{RV_{i,t}}{\overline{RV_{i,t}}})\overline{RV_{k,t}}, \\ \overline{RV_{k,t}} &= \frac{\sum_{j=1}^{t} RV_{k,j}}{t}. \end{aligned}$$

The resulting regression formula, dropping the k under-script, is then,

$$\begin{aligned}\text{HExpGl:}RV_{t+1} =&\beta_0 + \beta_1 ExpRV_t^1 + \beta_2 ExpRV_t^5 + \beta_3 ExpRV_t^{25} + \beta_4 ExpRV_t^{125} \\ &+ \beta_5 ExpGLRV_t^5 + \epsilon_t. \end{aligned} \tag{2.10}$$

According to Li and Tang (2022), combining all RV-based features leads to an improvement in overall forecasting performance. The resulting specification is referred to as OLSRV and includes 16 features, including the intercept. The regression formula for OLSRV is as follows:

$$\begin{aligned}\text{OLSRV:}RV_{t+1} =&\beta_0 + \beta_m MIDAS_t + \beta_p RVP_t + \beta_d RV_t^d + \beta_w RV_t^w + \beta_m RV_t^m + \beta_q RV_t^q \\ &+ \phi_d RV_t^d \sqrt{RQ_t^d} + \phi_w RV_t^w \sqrt{RQ_t^w} + \phi_m RV_t^m \sqrt{RQ_t^m} + \phi_q RV_t^q \sqrt{RQ_t^q} \\ &+ \beta_1 ExpRV_t^1 + \beta_2 ExpRV_t^5 + \beta_3 ExpRV_t^{25} + \beta_4 ExpRV_t^{125} \\ &+ \beta_5 ExpGLRV_t^5 + \epsilon_t \end{aligned} \tag{2.11}$$

In addition to RV-based features, we also include a set of features based on implied variances (IV) as suggested by the literature (Christensen and Prabhala (1998), Busch, Christensen, and Nielsen (2011), Li and Tang (2022)). Specifically, we consider the option-implied variance from call and put options denoted as $CIV_t^{m,\delta}$ and $PIV_t^{m,\delta}$, where the time to maturity $m \in \{30, 60, 91\}$, and $\delta \in \{0.1, 0.15, ..., 0.85, 0.9\}$. We augment the HAR specification with the these IV-based features, this results in 107 features for each stock, whenever available. [1] We define the specification using both the implied variances and HAR features as OLSIV:

[1] We have also considered the specification using only the IV features, but it resulted in inferior performance compared to HAR by a large margin.

$$\begin{aligned}\text{OLSIV:}\quad RV_{t+1} =& \beta_0 + \beta_d RV_t^d + \beta_w RV_t^w + \beta_m RV_t^m + \beta_q RV_t^q \\ &+ \sum_{m\in\{30,60,91\}} \sum_{\delta\in\{0.1,0.15,...,0.85,0.9\}} \left(\beta^C_{m,\delta} CIV_t^{m,\delta} + \beta^P_{m,\delta} PIV_t^{m,\delta}\right) + \epsilon_t,\end{aligned} \tag{2.12}$$

and the specification using all the RV-based features and implied variances as OLSRVIV:

$$\begin{aligned}\text{OLSRVIV:} RV_{t+1} =& \beta_0 + \beta_m MIDAS_t + \beta_p RVP_t + \beta_d RV_t^d + \beta_w RV_t^w + \beta_m RV_t^m + \beta_q RV_t^q \\ &+ \phi_d RV_t^d \sqrt{RQ_t^d} + \phi_w RV_t^w \sqrt{RQ_t^w} + \phi_m RV_t^m \sqrt{RQ_t^m} + \phi_q RV_t^q \sqrt{RQ_t^q} \\ &+ \beta_1 ExpRV_t^1 + \beta_2 ExpRV_t^5 + \beta_3 ExpRV_t^{25} + \beta_4 ExpRV_t^{125} + \beta_5 ExpGLRV_t^5 \\ &+ \sum_{m\in\{30,60,91\}} \sum_{\delta\in\{0.1,0.15,...,0.85,0.9\}} \left(\beta^C_{m,\delta} CIV_t^{m,\delta} + \beta^P_{m,\delta} PIV_t^{m,\delta}\right) + \epsilon_t.\end{aligned} \tag{2.13}$$

To enhance the forecasting performance, we incorporate several features that are not solely based on realized volatility or implied volatility. These additional features encompass the earnings announcement date, overnight return, daily returns, trading volumes, market capitalizations, and sector ETFs.

The earnings announcement date (EAD) is a crucial event for investors, as it provides valuable information into a company's business performance. Several studies, including Cao and Narayanamoorthy (2012), Barth and So (2014),Atilgan (2014),Lei, Wang, and Yan (2020), have shown that stock prices, and hence returns and realized volatilities, can react strongly to the new information provided during earnings announcements. As the announcement date is an anticipated event, it can be used to construct a feature for the task of forecasting. We construct 5 indicators of the EAD, $EAD_t^i, i \in \{-2, -1, 0, 1, 2\}$, indicating whether 2 days ago, 1 day ago, the day, the day after, and 2 days after time t is an earnings announcement date for that stock. By incorporating EAD_t^i as a feature in volatility models, we can more accurately capture the impact of earnings announcements on realized volatility and improve the accuracy of our forecasts.

We incorporate overnight returns, daily returns, trading volumes, and market capitalizations as additional features. To account for the polarity of overnight returns, we calculate the squared value of both the overall overnight return and the positive overnight returns (Ahoniemi and Lanne (2013), Todorova and Souček (2014)). The daily returns are derived from the logarithmic returns over the holding period sourced from CRSP. For trading volumes and market capitalization, we normalize the trading volume by shares outstanding and include the logarithm of the previous day's price multiplied by the trading volume, which provides a measure of the dollar value of market activity (Liu, Choo, Lee, and Lee (2023)). Additionally, we incorporate the logarithm of market capitalization as the size factor. For the first five features, we also include their corresponding weekly, monthly, and quarterly averages. However, for market capitalization, we solely utilize the daily value, as it exhibits less variation compared to the other features. This results in 21 additional features denoted as $VPO_t^{i,h}$, where $i \in \{1,2,3,4,5\}, h \in \{d,w,m,q\}$ and MkC_t.

We leverage the Global Industry Classification Standard (GICS) codes to assign industry classifications to stocks and then match them with their respective industry SPDR ETFs. We utilize the RVs of these ETFs as a proxy for the market factor. In cases where there is no corresponding ETF available, we employ the SPDR S&P 500 ETF Trust (SPY) as the proxy. This approach yields four additional features denoted as ETF_t^h, where $h \in \{d,w,m,q\}$, representing the daily value as well as the corresponding weekly, monthly, and quarterly averages.

Augmenting these 5 + 21 + 4 features to the OLSRV model, we name the resulting specification as OLSVPOS for its inclusion of non-RV based features. Furthermore, we consider combining all features, named OLSALL, which utilizes all the $p = 148$ features mentioned above. For simplicity we label the features as $x_{j,t}, j = 1, ..., p$. The regression formula for OLSALL is as follows:

$$\text{OLSALL:} \quad RV_{t+1} = \sum_{j=1}^{p} \beta_j x_{j,t} + \epsilon_t. \tag{2.14}$$

2.2.3 Machine Learning Methods

We consider several popular machine learning algorithms for regression, including LASSO, PCR, Random Forest, Gradient Boosted Regression Trees, and Neural Networks.

For simplicity, we denote our training data as $\{(x_i, y_i)\}_{i=1}^{n}$, where y_i represents the one-day-ahead $RV_{k,t+1}$ for some stock k and $x_i \in \mathbb{R}^p$ is the corresponding p-dimensional feature vector, using the information up to the end day t for the same stock. We use all the p features from OLSALL by default. Furthermore, we denote $\{(xv_i, yv_i)\}_{i=1}^{nv}$ and $\{(xt_i, yt_i)\}_{i=1}^{nt}$ as the validation and test data, respectively. Most machine learning algorithms require normalization of the features, and we only use the training data to compute the normalization coefficients, which are then applied to the validation and test data.

LASSO stands for least absolute shrinkage and selection operator (Tibshirani (1996)), it is a penalized least squares method that imposes an $\mathbf{L}_1$-penalty on the regression coefficients, and it performs continuous shrinkage and automatic variable selection simultaneously. Given the data $\{(x_i, y_i)\}_{i=1}^{n}$, LASSO solves the $\mathbf{L}_1$-penalized regression problem of finding the $\beta = \{\beta_j\}_{j=1}^{p}$ that minimizes the following expression:

$$\frac{1}{n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} x_{i,j} \beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \, . \tag{2.15}$$

Here the λ is the hyperparameter that controls the level of regularization. When λ is large, LASSO returns very few non-zeros values for β, and when it is 0, it reduces to usual OLS method.

PCR, or principal component regression, is motivated by the fact that the features are often highly correlated (Li and Tang (2022)). Specifically, we stack the training data as $Y = (y_1, ..., y_n)^T \in \mathbb{R}^{n \times 1}$, and $X = (x_1, ..., x_n)^T \in \mathbb{R}^{n \times p}$. The X and Y are already normalized such that they are centered and have 0 empirical means. We then perform the (compact) Singular Value Decomposition (SVD) on the matrix X, resulting in:

$$X = UDV^T, \text{where} \quad U \in \mathbb{R}^{n \times p}, D \in \mathbb{R}^{p \times p}, V \in \mathbb{R}^{p \times p}. \tag{2.16}$$

In the above expression, D is a diagonal matrix with diagonal elements $D_{1,1} \geqslant D_{2,2} \geqslant D_{p,p} \geqslant 0$, V is an orthogonal matrix, and U is composed of orthogonal columns. By selecting the number of components, J, and denote $V_{1:J} = [v_1, ..., v_J]$ as the sub-matrix of V with first J columns, PCR effectively regresses Y on the $W_J \equiv XV_{1:J}$. In fact, since W_p is composed of orthogonal columns, we may solve for a single β with all components, and perform cross-validation on the validation set to select the optimal J. Here J is a hyperparameter controlling the complexity of the model. When we set $J = p$, then it reduces to usual OLS method, and when J is small, then the regression is performed only on the top principal components.

The Random Forest (RF) algorithm, introduced by Leo (2001), is a nonlinear ensemble method that combines individual decision trees and is a major player in data-mining. Luong and Dokuchaev (2018) used RF for forecasting the direction and magnitude of RV for top stocks in the Australian Stock Exchange and demonstrated superior performance compared to traditional methods. To grow the random forest, we start from growing a regression tree, with K leaves (terminal nodes), and depth L,

$$f^{tree,b}(x) = \sum_{k=1}^{K} \theta_{k,b} \mathbf{1}_{\{x \in C_k(L)\}}, \tag{2.17}$$

where $C_k(L)$ is one of the K partitions of the data, a product up to L indicator functions, each is univariate on one entries of x (Gu, Kelly, and Xiu (2020)). For a given set of data, one may use the Classification and Regression Trees (CART) algorithm (Breiman, Friedman, Olshen, and Stone (1984)) to grow a decision tree $f^{tree,b}(x)$ that partitions the feature space into rectangles and fits a simple model (average) in each element of the partition. The tree is grown in a greedy way by searching for the optimal split over the features and dividing into leaf nodes until one reaches a minimum threshold of the leaf node, or one reaches a minimum deviance threshold that measures the goodness of fit, or one reaches the maximum depth.

RF has two key characteristics that make trees much better predictors. First, each of the individual trees is grown using a bootstrapped sample (random sampling with replacement) to reduce overfitting. Second, when growing the individual tree, instead of searching the entire set of features, RF randomly selects a subset of features and performs the search of optimal split only on this subset, thus reducing the possibility of overfitting (and increasing the computational efficiency).[2] The result is a collection of trees $\frac{1}{B}\sum_{b=1}^{B} f^{tree,b}(x)$.

For RF, we have a number of hyperparameters that control the complexity of the fitted models. One may select the number of trees, B, where a larger number results in a more complex model at the cost of computational resources. The maximum depth L and minimum number of observations at the leaf node control the complexity of each individual tree, where a larger maximum depth/smaller minimum number of observations at the leaf node results in a more complex tree.

The Gradient Boosted Regression Trees (GBRT) is a tree-based ensemble method that differs from Random Forest in that it grows trees sequentially and uses the residuals from previously grown trees to enhance the performance of the new tree. Random Forest, on the other hand, grows individual trees independently and combines them by equal weights

[2] The algorithm without the random selection of features is called Bagging by Breiman (1996), and it is generally not performing as good as RF in practice and much slower to train.

as a final predictor. This approach has been described in Freund and Schapire (1995) and Friedman (2001).

To grow each individual tree, GBRT begins with a loss function, such as squared loss, and evaluates the gradient with respect to the loss function using the previous collection of trees. This gradient serves as the target for the tree, instead of the actual y_i's. The resulting tree is multiplied by a shrinkage parameter λ, such that $0 < \lambda \leqslant 1$, before being added to the collection of trees. Unlike Random Forest, GBRT prefers many shallow trees over a few deep trees, as noted in Hastie, Tibshirani, Friedman, and Friedman (2009). In addition to the shrinkage parameter λ, GBRT also considers the maximum number of trees B, maximum depth, and minimum number of observations at leaf node. To produce a better model, one may perform sampling without replacement at each iteration to sample a fraction of training data and perform stochastic gradient boosting, as described in Friedman (2002).

Neural networks (NN), also known as deep learning, are considered by many to be the most powerful modeling device in machine learning, as noted in LeCun, Bengio, and Hinton (2015). They use compositions of simple nonlinear functions, indexed by coefficients, to approximate complex ones. According to the universal approximation theorem, as described in Hornik, Stinchcombe, and White (1989) and Cybenko (1989), a neural network with one hidden layer can approximate any Borel measurable function from one finite dimensional space to another with any desired degree of accuracy, given enough neurons. Empirically, neural networks with more hidden layers have been extremely successful in various tasks, including image recognition, game playing, and autonomous driving, among others. Deep neural networks can be efficiently estimated using variants of stochastic gradient descent methods with back-propagation algorithms, which can be implemented using standard software packages such as TensorFlow (Abadi, Barham, Chen, Chen, Davis, Dean, Devin, Ghemawat, Irving, Isard, et al. (2016)) and Keras (Chollet et al. (2015)).

The neural network training process involves determining the architecture of the network, the loss function, and the training pipeline. To build the volatility prediction function using a neural network $f(x, \theta)$ indexed by coefficients θ, the only requirement is that the input and output shapes must match the shapes of the feature vector and prediction target, namely p and 1. While we are not limited in the intermediate architecture, for simplicity, we will use a simple feed-forward neural network. Let K denote the number of hidden layers, and $\{n_1, ..., n_K\}$ be the number of hidden units in each layer, where layer 1 is connected to the input and layer K is connected to the output layer, denote $z_k, k \in \{1, ..., K+1\}$ be the output vector for layer k (also the input for layer $k+1$), and $z_0 \equiv x$, and $\sigma_k(\cdot)$ as the activation function, the simple feed-forward neural network with architecture $[n_1 \times n_2 \times ... \times n_K]$ may be written recursively as,

$$
\begin{aligned}
z_0 &= x, \\
z_k &= \sigma_k(b_k + W_k z_{k-1}), k \in \{1, ..., K+1\} \\
f(x, \theta) &= z_{K+1}, \text{where} \quad \theta = \{b_1, ..., b_{K+1}, W_1, ..., W_{K+1}\},
\end{aligned}
$$

Here $b_k \in \mathbb{R}^{n_k}$ and $W_k \in \mathbb{R}^{n_k \times n_{k-1}}$ are the bias term and weight term for layer k, and $n_0 = p$ is the dimension of input feature. The activation function $\sigma_k(\cdot)$ adds nonlinearity to the neural network model. Common functions for this purpose include the rectified linear unit (ReLU), Leaky ReLU, tanh, and others (LeCun, Bengio, and Hinton (2015))). In this case, we use Leaky ReLU for the intermediate layers for simplicity, and tanh for the final layer to limit the forecasts within the range of (-1,1). The scaled forecasts are then adjusted using the standardization from the training data.

$$
\begin{aligned}
\text{LeakyReLU}_\alpha(x) &= \alpha x \mathbf{1}_{x<0} + x \mathbf{1}_{x \geqslant 0}, \quad \text{where} \quad 0 \leqslant \alpha < 1, \\
\tanh(x) &= \frac{e^x - e^{-x}}{e^x + e^{-x}}.
\end{aligned}
$$

As the network becomes deeper (with a larger value of K) or wider (with larger values of n_k's), it becomes more complex. While this may increase its capacity to approximate the training data, it can also make it harder to train and may cause overfitting. For our network architecture, we choose a simple fully connected feed-forward network and determined the number of neurons in each layer using a geometric pyramid rule (Masters (1993), Gu, Kelly, and Xiu (2020)). We consider several architectures with varying numbers of hidden layers. The deepest architecture (NN6) has 6 hidden layers, starting with 64 neurons in the first layer and decreasing to 32, 16, 8, 4, and 2 neurons in each subsequent layer. The medium architecture (NN4) has four hidden layers, with 64, 32, 16, and 8 neurons in each layer. The shallowest architecture (NN2) has only two hidden layers, with 64 and 32 neurons in each layer.

The loss function, denoted as $Loss(f(x, \theta), y)$, can be any sensible function for which we can compute the gradients for back-propagation. In the training process, we employ two regularization techniques: the learning rate shrinkage method called "Adam", described in Kingma and Ba (2014), and early-stopping, as described in Morgan and Bourlard (1990). The learning rate is a key tuning parameter for the neural network, as it controls the step size of the gradient descent. Ideally, we would prefer a large learning rate away from the optimum to ensure fast convergence and a small learning rate close to the optimum to avoid oscillation."Adam" is an adaptive algorithm that performs learning rate shrinkage. Early stopping is a cross-validation regularization technique that determines when to stop the learning process to avoid overfitting. We select the patience threshold N_{pthres}, set the patience count N_p to 0, and generate a validation set that is not used for training. As we train the model, we evaluate the loss function on the validation set regularly. Each time, we increase N_p by 1, and if the validation loss is lower than previous lowest, we keep the current coefficient as ϑ^*, and reset N_p to 0. Once N_p exceeds N_{pthres}, we stop the training process. The resulting neural network $f(x, \vartheta^*)$ would be used for forecasting.

Due to the inherent stochastic nature of neural networks, which includes the initialization of weights and biases, as well as the random sampling of batches during stochastic descent, using different random seeds can result in different forecasts. In order to reduce the variance in predictions, we employ an ensemble approach (Hansen and Salamon (1990),Dietterich (2000)). This approach involves averaging the predictions from neural networks with the same architecture initialized with different random seeds. We refer to these ensemble neural networks as NN6E, NN4E, and NN2E, respectively.

In summary, Lasso and PCR are direct extensions of OLS-based methods. Lasso reduces dimensionality through variable selection, while PCR reduces dimensionality through principal components. RF and GBRT are both nonlinear tree-based methods, but they differ in their approach. RF utilizes bootstrapped samples and grows independent trees, while GBRT utilizes gradient information with respect to previously grown trees. Neural networks are the most flexible machine learning method, as their functional form can be any architecture as long as the gradients can be computed.

2.3 Data

2.3.1 Raw Data

We consider two large universes of stocks: the first one is the set of stocks that have ever been constituents of the S&P 500 index (denoted as S&P 500), and the second is the set of all stocks and ETFs traded in the major United States stock exchanges (denoted as US), from January 1996 to December 2022. The stocks (and ETFs) [3] are identified using permno, a unique permanent stock (share class) level identifier from CRSP. We collect the high-frequency intraday trade prices from the Trade and Quote (TAQ) database. Implied volatilities from call and put options for the same universe of stocks are collected from OptionMetrics over the same period, with maturities between one month and three months

[3] If there is no ambiguity, we refer to both stocks and ETFs simply as stocks.

and absolute delta between 0.1 and 0.9. Additionally, we collect opening and closing prices, holding period returns, trading volumes, and shares outstanding from the CRSP daily stock file to compute features related to overnight returns and daily trading volumes. The earnings announcement date is collected from the Institutional Broker's Estimate System (I/B/E/S) database. We collect the Global Industry Classification Standards (GICS) code from the Compustat database for the stocks to match their corresponding industry SPDR ETFs (Energy (XLE), Materials (XLB), Industrials (XLI), Consumer Discretionary (XLY), Consumer Staples (XLP), HealthCare (XLV), Financials (XLF), Information Technology (XLK), Communication Services (XLC), Utilities (XLU), Real Estate (XLRE)). If no classification is available, we match the stock to the SPDR S&P 500 ETF TRUST (SPY).

For the same stock, the ticker symbol may change over time[4], and we use the daily stock file table (dsenames) from CRSP to link the ticker symbols from TAQ to permnos through 2022 (CRSP (2021)). We use the 8-digit Named CUSIPs (NCUSIPs)[5] from CRSP to match the implied volatilities from OptionMetrics, where the implied volatilities data starts at 1996.

We follow the data cleaning procedure outlined in Da and Xiu (2021) for our analysis. This procedure involves constructing the national best bid and offer (NBBO) from all exchanges at a 1-second frequency and removing observations that fall outside the range of NBBO quotes. We exclude days with half trading hours. Stocks that terminate before January 1st, 1996, or start after December 31st, 2021, are removed to ensure they appear in the training or validation set at least once. We also remove stock-days with less than 12 observations after subsampling. Additionally, we eliminate potential outliers that exhibit a sudden rise or fall of RV followed by a bounce back on the second day. Finally, stocks with less than 150 days of data within this period are excluded from the analysis.

The matching procedure yields a sample of 11,771 unique stocks (permnos) for the US

[4] For example, Facebook Inc changed its ticker symbol from "FB" to "META" effectively on 2021-Nov-1st.

[5] CUSIPs change over time, and Header CUSIPs only report the latest CUSIP of a stock, while Names CUSIPs track the entire history of a stock

stock universe from 1996-Jan-02 to 2022-Dec-30, with 30,429,300 stock-day observations. A subset of S&P 500 stocks is selected using the same criteria, resulting in a sample of 1,162 unique stocks for the same period, with 5,023,069 stock-day observations. The exact number of stock-day observations after each data cleaning step is reported in Table 2.1.

	S&P 500		US	
Steps	Counts	Proportion	Counts	Proportion
Has Price	5089365	1	31997147	1
Exclude HTD	5040134	0.9903	31701305	0.9908
$\geqslant$ 12 Obs	5023629	0.9871	30445465	0.9515
Filtering	5023069	0.9870	30429300	0.9510

Table 2.1: Sample Size by Data Cleaning Steps

Note: This table shows the sample size at the end of each data cleaning step for the S&P 500 stocks and US stocks. Step 1: We collect the total number of stock-day combinations that have trading activities. Step 2: We remove the half trading dates. Step 3: We require each stock-day to have at least 12 observations (trading prices) after subsampling. Step 4: We apply a filter to remove potential outliers. We also report the proportion of data that remains, in addition to the counts.

2.3.2 Data Processing

To compute the RVs, we apply a 15-minute subsampling frequency (Liu, Patton, and Sheppard (2015), Li and Xiu (2016)) on the intraday prices and compute the returns accordingly. The overnight returns are corrected using the holding period returns and prices for opening and closing. Consistent with Andersen, Bollerslev, Diebold, and Labys (2003), Bucci (2020), and Zhang, Zhang, Cucuringu, and Qian (2024), we consider the logarithm of the annualized RVs to reduce the impact of extreme values.

Figure 2.1 shows the percentiles of RVs for the S&P 500 and US stock universe over time, supporting the transformation of RVs into log-scale. Without this transformation, the loss function would be dominated by observations of large RVs. The figure also reveals spikes around significant events such as the 2000 dot-com bubble, the 2008 financial crisis, the 2011 European debt crisis, and the 2020 Covid-19 pandemic, among others.

The implementation of MIDAS requires a choice of of the cutoff L, and the two tuning

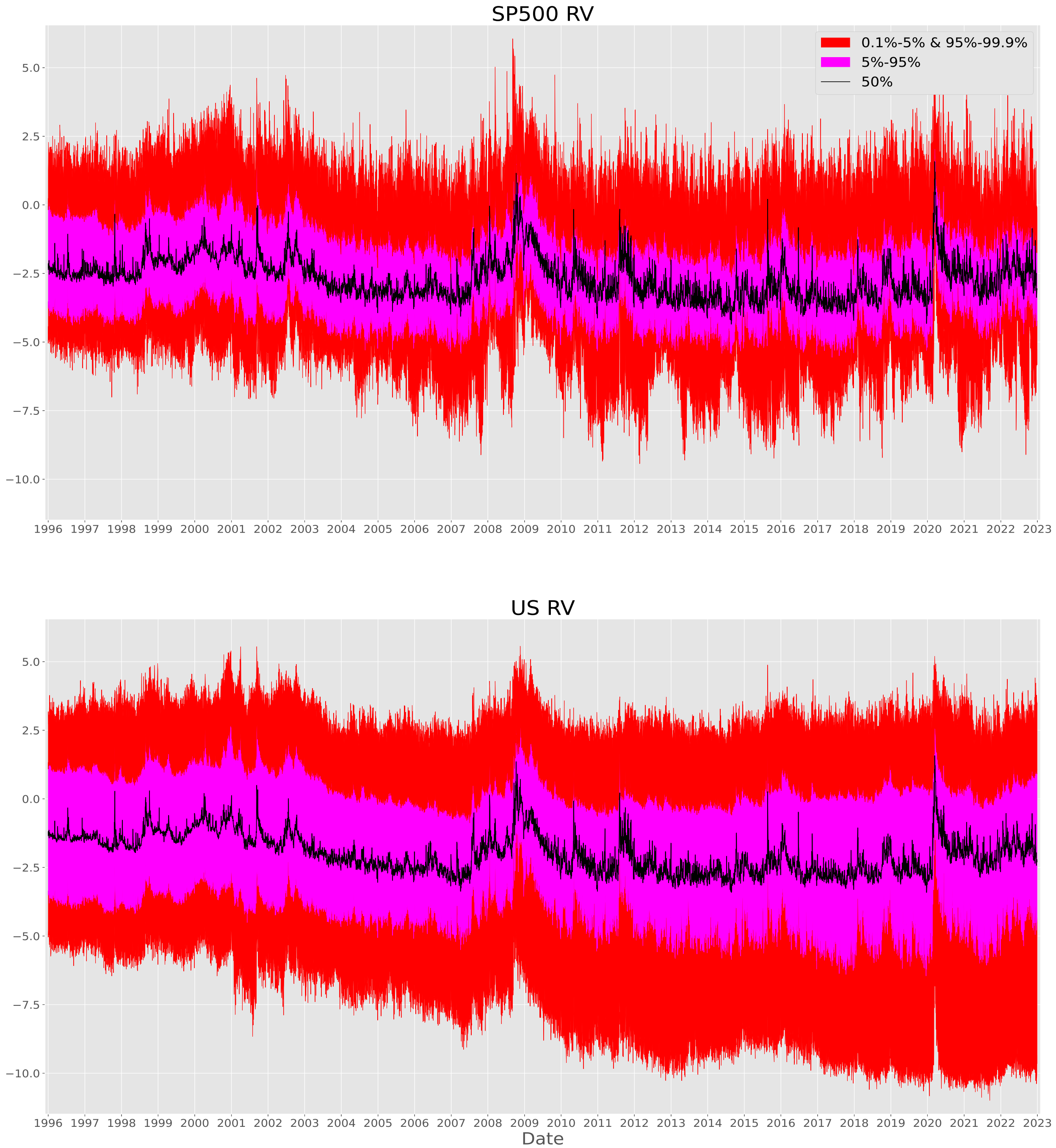

Figure 2.1: Quantiles of RVs

Note: This figure displays the 0.1st, 5th, 50th, 95th, and 99.9th percentiles of daily annualized realized volatilities (in logarithm) for stocks in the S&P 500 and US stock universe from 1996 to 2022. The daily realized volatilities are computed using intraday high-frequency returns sampled at 15-minute frequency.

parameters θ_1 and θ_2. Follow Ghysels, Santa-Clara, and Valkanov (2006), we fix the cutoff at $L = 50$ and set $\theta_1 = 1$. For θ_2, existing literature (Bollerslev, Hood, Huss, and Pedersen (2018), Ghysels and Qian (2019)) usually replies on a grid-search to find the best θ_2 that maximizes the predictability over the full-sample due to computation burden, resulting in potential look-ahead bias. Instead, we apply a year-specific rolling window to tune the θ_2. Specifically, we gather the RVs for all the stocks in one year and run a MIDAS regression to find the best θ_2, and use this as the θ_2 for data in the coming year for all the stocks. When generating the corresponding features for the actual prediction tasks, we compute the MIDAS feature year-by-year using the set of θ_2's, ensuring that there is no look-ahead bias.

2.3.3 Training Scheme

Due to the large amount of data, we rely on a rolling window estimation procedure. Specifically, we use five calendar years of data for training, the next year for validation, and the year after that for forecasting. Since our data starts in 1996, the first forecasting year would be 2002, where the data from 1996-2000 is used for training, and the data from 2001 is used for validation. For the S&P 500 stocks, we select the stocks for the test set based on their membership to S&P 500 constituents on the day prior to test year, and there are 499 stocks on average. We select the S&P 500 stocks for the training and validation set if the stock has ever been in the constituents before the day prior to the test year, and there are on average 713 stocks. For the US universe, there are on average 5215 stocks in the test year and 6292 stocks in the training and validation years.

For the ML models, we use the training data to fit the models and the validation data to perform cross-validation for the hyperparameters. For the OLS-based methods, no validation is required, so the validation year's data is also included for training. The set of hyperparameters used is reported in Table 2.2. We mainly rely on the "pooled" fit, where we fit one model using all the available data in the universe of stocks. For the OLS-based method, we also consider the "individual" fit, which uses the stock's own data.

Model	Hyperparameter	Value
Lasso	Number of λ's	100
	$\lambda_{Min}/\lambda_{Max}$	10^{-7}
	Maximum iteration	10^{6}
PCR	Maximum components	p
	Minimum variance ratio	10^{-8}
	SVD Solver	Full
RF	Number of trees	100
	Maximum depth	$\{10, 15, 20\}$
	Minimum sample at leaf node	10
	Number of features to consider for best split	$\sqrt{p}$
	Loss function	MSE
GBRT	Number of trees	100
	Learning rate	$\{0.1, 0.3, 0.5\}$
	Maximum depth	5
	Minimum sample at leaf node	10
	Number of features to consider for best split	$\sqrt{p}$
	Validation fraction	10%
	Loss function	MSE
NN	Architecture NN6	$64 \times 32 \times 16 \times 8 \times 4 \times 2$
	Architecture NN4	$64 \times 32 \times 16 \times 8$
	Architecture NN2	64×32
	Training batch size	10000
	Validation frequency	20
	Epoch	20
	Learning rate	$\{0.002, 0.001, 0.0005, 0.0002\}$
	Patience threshold N_{pthres}	20
	Activation function (hidden layer)	$\text{LeakyReLU}_{0.01}(\cdot)$
	Activation function (output layer)	$\tanh(\cdot)$
	Loss function	MSE
	Random seeds	{2020,2021,2022,2023,2024}

Table 2.2: Hyperparameters for Machine Learning Models

Note: This table reports the hyperparameters for the five machine learning models we considered in the paper, Lasso, Principal Component Regression (PCR), Random Forest (RF), Gradient Boosted Regression (GBRT), and Neural Network (NN).

We follow Swanson and White (1997) and apply an "insanity filter" to the forecast. If a forecast falls outside the range of values of the target variable observed in the estimation period, then the forecast is replaced by the unconditional mean over that period. The upper and lower bounds to trigger the insanity filter for each test year are computed using only the corresponding training data.

2.3.4 Evaluation Metrics

Denote $\widehat{y_{k,t}}$ as the predicted value of $y_{k,t}$, the log-scale RV for stock k at time t. Denote $\mathbf{1}_{(k,t)\in Test}$ as the indicator variable that RV for stock k at time time t exists in the test set, we compute the following out-of-sample metrics to compare the predictive performance of various forecasting models,

$$
\begin{aligned}
\text{Mean squared error (MSE) :}&\frac{1}{n_{\text{Test}}}\sum_{k=1}^{K}\sum_{t=1}^{T}\left(\widehat{y_{k,t}}-y_{k,t}\right)^2\mathbf{1}_{(k,t)\in\text{Test}},\\
\text{Quasi-likelihood (QLike) :}&\frac{1}{n_{\text{Test}}}\sum_{k=1}^{K}\sum_{t=1}^{T}\left[\frac{\exp(y_{k,t})}{\exp(\widehat{y_{k,t}})}-\left(y_{k,t}-\widehat{y_{k,t}}\right)-1\right]\mathbf{1}_{(k,t)\in\text{Test}},\\
\text{Relative R-squared (R2) :}&1-\frac{\sum_{k=1}^{K}\sum_{t=1}^{T}\left(\widehat{y_{k,t}}-y_{k,t}\right)^2\mathbf{1}_{(k,t)\in\text{Test}}}{\sum_{k=1}^{K}\sum_{t=1}^{T}\left(\widehat{y_{k,t}}^{Benchmark}-y_{k,t}\right)^2\mathbf{1}_{(k,t)\in\text{Test}}},\\
\text{where}\quad n_{\text{Test}}&=\sum_{k=1}^{K}\sum_{t=1}^{T}\mathbf{1}_{(k,t)\in\text{Test}}.
\end{aligned}
$$

Here, the $\widehat{y_{k,t}}^{Benchmark}$ is the forecast from a benchmark model, and we use the HAR model as the benchmark in this study for its simplicity yet appealing empirical performance in the literature. QLike is computed using the original scale by definition, while for MSE and R2, we use the log-scale of RV to reduce the impact of extreme RVs. Note that sometimes a poor point forecast might result in a very large error (For example, when $y_{k,t}$ is large and $\widehat{y_{k,t}}$ is small in QLike), which can lead to spurious overall performance. To mitigate the effect of possible extreme observations, we winsorize the squared error and quasi-likelihood at the 99.99th percentiles. The winsorized version is called MSE* and QLike*, respectively.

MSE and QLike are "robust" loss functions in preserving the rankings of competing forecasts (Hansen and Lunde (2006), Patton (2011)), while the relative out-of-sample predictive R^2 produces a unit-free measure that is easy to interpret. The modified Diebold-Mariano test is conducted to make pairwise comparisons of competing methods (Diebold and Mariano (2002)). Specifically, we compute the cross-sectional average of loss differentials, and compute the corresponding mean and Newey-West standard error over the test sample (Gu, Kelly, and Xiu (2020)).

2.4 Predictive Performance

Table 2.3 and 2.4 present the out-of-sample forecasting performance for S&P 500 stocks and US stocks, respectively. Each row represents a forecasting model, and each column represents a performance metric in the left panel. The best-performing model is highlighted in bold font in each column. Starting with Table 2.3, which focuses on S&P 500 stocks consisting of large and liquid companies, the performance metrics reported include R2, MSE, winsorized MSE*, QLike, and winsorized QLike*. The relative performance of each model is shown in the Gain(%) column, with the random walk (RW) model set to 0 and the HAR model set to 100 as benchmarks. The number of features used in each model is reported in the nFeature column. In the upper panel, OLS-based methods are presented, while the bottom panel consists of machine learning methods. The relative R2 for the HAR model is 0 by definition. The MIDAS model has a slightly worse performance with a relative R2 of -0.0144. Parsimonious regression forms like SHAR, HEXP, and HARQ achieve relative R2 values between 0.0036 and 0.0334. Combining all RV-based features in OLSRV yields a relative R2 of 0.0430. Including IV-based features to the HAR model (OLSIV) improves the relative R2 to 0.0415. Combining all RV and IV-based features, referred to as OLSRVIV, results in a relative R2 of 0.0716. The non-RV and non-IV based features proposed in this paper significantly enhance forecasting performance. Augmenting these features to OLSRV produces OLSVPOS, which achieves a relative R2 of 0.1537. Finally, the best OLS model considered here is OLSALL, which incorporates all 148 features and yields a relative R2 of 0.1691. Among the machine learning methods, Lasso and PCR perform comparably to OLSALL, while RF and GBRT underperform OLSALL. The neural network models with 6, 4, and 2 hidden layers, referred to as NN6, NN4, and NN2, respectively, achieve relative R2 values ranging from 0.1757 to 0.1780. However, when an ensemble of neural networks with the same architecture is formed, referred to as NN6E, NN4E, and NN2E, the relative R2 further increases to the range of 0.1810 to 0.1835.

Model	R2	MSE	MSE*	QLike	QLike*	Gain(%)	nFeature
RW	-0.4868	0.7484	0.7473	0.5999	0.5748	0	0
MIDAS	-0.0144	0.5106	0.5101	0.4559	0.4428	97.04	2
HAR	0	0.5033	0.5028	0.4557	0.4426	100	5
SHAR	0.0036	0.5015	0.5010	0.4544	0.4411	100.74	6
HEXP	0.0075	0.4995	0.4990	0.4536	0.4404	101.55	6
HARQ	0.0334	0.4865	0.4859	0.4441	0.4311	106.87	9
OLSRV	0.0430	0.4817	0.4811	0.4412	0.4278	108.84	16
OLSIV	0.0415	0.4825	0.4819	0.4237	0.4125	108.52	107
OLSRVIV	0.0716	0.4673	0.4668	0.4161	0.4045	114.70	118
OLSVPOS	0.1537	0.4260	0.4254	0.3374	0.3253	131.57	46
OLSALL	0.1691	0.4182	0.4177	0.3281	0.3169	134.73	148
LASSO	0.1673	0.4191	0.4185	0.3291	0.3178	134.37	148
PCR	0.1665	0.4195	0.4190	0.3292	0.3182	134.21	148
RF	0.1365	0.4346	0.4341	0.3437	0.3319	128.05	148
GBRT	0.1274	0.4392	0.4387	0.3404	0.3289	126.16	148
NN6	0.1757	0.4149	0.4143	0.3227	0.3098	136.09	148
NN4	0.1776	0.4140	0.4133	0.3252	0.3116	136.47	148
NN2	0.1780	0.4138	0.4132	0.3235	0.3102	136.56	148
NN6E	0.1810	0.4122	0.4116	0.3215	0.3088	137.18	148
NN4E	0.1832	0.4111	0.4105	0.3222	0.3088	137.62	148
NN2E	**0.1835**	**0.4110**	**0.4104**	**0.3212**	**0.3081**	**137.69**	148

Table 2.3: Out-of-Sample Forecasting Performance for S&P 500 Stocks

Note: This table presents metrics that measure the out-of-sample forecasting performance for various models discussed in the paper on the set of stocks that once belonged to the S&P 500 index. The upper panel includes all the ordinary least squares methods, while the bottom panel includes all the machine learning methods. R2 represents the relative out-of-sample predictive R^2, and the larger the value, the better the performance. MSE stands for the mean squared error, and MSE* stands for the winsorized mean squared error, where extreme errors (larger than the 99.99th percentile) are replaced by the boundary value (99.99th percentile). QLike stands for the quasi-likelihood, and QLike* stands for the winsorized QLike. The smaller the value of the latter four metrics, the better the performance. The Gain(%) column set the relative performance of a random walk model (RW) as 0, and HAR model as 100% based on MSE for better comparison among models. The final column nFeature reports the number of features used for the corresponding model. The best-performing model is highlighted in bold in each column, and the (ensemble) neural networks outperform all other methods.

When analyzing the set of US stocks in Table 2.4, we observe similar patterns in the OLS-based methods, although the MSE and QLike metrics are considerably higher in every entry.

Model	R2	MSE	MSE*	QLike	QLike*	Gain(%)	nFeature
RW	-0.4948	0.9167	0.9147	0.8089	0.7578	0	0
MIDAS	-0.0163	0.6232	0.6224	0.5438	0.5157	96.71	2
HAR	0	0.6133	0.6123	0.5366	0.5095	100	5
SHAR	0.0019	0.6121	0.6112	0.5350	0.5082	100.38	6
HEXP	0.0067	0.6092	0.6083	0.5292	0.5035	101.35	6
HARQ	0.0154	0.6038	0.6028	0.5311	0.5044	103.12	9
OLSRV	0.0241	0.5985	0.5975	0.5226	0.4974	104.87	16
OLSIV	0.0058	0.6097	0.6089	0.5259	0.5014	101.18	107
OLSRVIV	0.0286	0.5957	0.5949	0.5153	0.4920	105.77	118
OLSVPOS	0.0809	0.5637	0.5627	0.4514	0.4272	116.34	46
OLSALL	0.0835	0.5620	0.5612	0.4475	0.4246	116.89	148
LASSO	0.0823	0.5628	0.5620	0.4483	0.4252	116.63	148
PCR	0.0818	0.5631	0.5623	0.4493	0.4256	116.53	148
RF	0.0851	0.5610	0.5603	0.4517	0.4270	117.21	148
GBRT	0.0702	0.5702	0.5694	0.4563	0.4312	114.19	148
NN6	0.1032	0.5500	0.5492	0.4346	0.4104	120.85	148
NN4	0.1094	0.5462	0.5453	**0.4323**	**0.4063**	122.11	148
NN2	0.1108	0.5453	0.5445	0.4347	0.4065	122.40	148
NN6E	0.1102	0.5457	0.5449	0.4327	0.4087	122.27	148
NN4E	**0.1136**	**0.5436**	**0.5427**	0.4325	0.4066	**122.97**	148
NN2E	0.1127	0.5442	0.5433	0.4348	0.4086	122.77	148

Table 2.4: Out-of-Sample Forecasting Performance for US Stocks

Note: This is a continuation of Table 2.3, where the forecasting exercise is conducted on the set of US stocks. The best-performing model is highlighted in bold in each column. The (ensemble) neural network achieves the best performance among all.

This is due to the inclusion of smaller stocks that are more challenging to predict. While the boost in performance relative to HAR by incorporating additional features is weaker for this set of stocks, the inclusion of these features still improves performance. The largest additional gain still comes from the set of non-RV and non-IV based features. OLSALL now achieves a relative R2 of 0.0835, representing a 16.89% additional boost against the naive random walk model relative to HAR. Among the machine learning algorithms, RF now performs better than OLSALL. This is likely because tree-based methods like RF rarely produce extreme values, as the predictions are based on averages from certain leaf nodes. Neural networks still demonstrate the best performance across all metrics, producing relative R2 values ranging from 0.1032 to 0.1108. The relative R2 further increases to the range of 0.1102 to 0.1136 for the ensemble neural networks. The gaps between MSE and MSE*, as well as between QLike and QLike*, are slightly larger when more smaller stocks with lower market capitalization are included compared to the constituents of the S&P 500 index.

Table 2.5 examines the impact of using a pooled approach versus an individual approach for the S&P 500 universe. The pooled approach utilizes all available data to fit a single model, while the individual approach builds separate models for each stock using its own data. Results indicate that, for each model specification, the pooled approach yields higher R2 values and lower MSE* and QLike* values compared to the individual approach. In the individual approach, simpler specifications (MIDAS/HAR/HARQ) demonstrate better performance, which deteriorates as more features are included. In contrast, the pooled approach shows improved performance as more features are incorporated. This finding potentially explains why Lasso and PCR do not outperform OLSALL. With a substantial amount of data, nearly all considered features become influential.

Table 2.6 compares the pooled and individual approaches for the US stock universe. Similar patterns are observed as in the S&P 500 universe, where the individual approach underperforms compared to the pooled approach, and this difference becomes more pronounced

	R2		MSE*		QLike*	
Model	Individual	Pooled	Individual	Pooled	Individual	Pooled
MIDAS	-0.0296	-0.0144	0.5177	0.5101	**0.4498**	0.4428
HAR	-0.0236	0	0.5147	0.5028	0.4576	0.4426
SHAR	-0.0201	0.0036	0.5129	0.5010	0.4545	0.4411
HEXP	-0.0560	0.0075	0.5303	0.4990	0.5030	0.4404
HARQ	**-0.0166**	0.0334	**0.5103**	0.4859	0.5099	0.4311
OLSRV	-0.0388	0.0430	0.5214	0.4811	0.4933	0.4278
OLSIV	-0.2920	0.0415	0.6481	0.4819	0.5675	0.4125
OLSRVIV	-0.3170	0.0716	0.6604	0.4668	0.5942	0.4045
OLSVPOS	-0.2180	0.1537	0.6115	0.4254	0.8627	0.3253
OLSALL	-0.5436	**0.1691**	0.7749	**0.4177**	1.3462	**0.3169**

Table 2.5: Individual vs Pooled Fit, S&P 500 Stocks

Note: This table compares the forecasting performance between the pooled and individual fit for various ordinary least squares methods for the S&P 500 universe. We evaluate the relative out-of-sample predictive R^2 (R2), winsorized mean squared error (MSE*), and winsorized quasi-likelihood (QLike*), and highlight the best-performing model in bold font in each column. For each model, the pooled fit using all the data outperforms the individual fit using only single stock data. The OLSALL with all the features is the best-performing ordinary least square method using the pooled data. The HARQ stands out as the best-performing ordinary least square method using individual stock data under R2 and MSE*, while MIDAS with hyper-parameters tuned using the pooled data performs the best using individual stock data under QLike.

	R2		MSE*		QLike*	
Model	Individual	Pooled	Individual	Pooled	Individual	Pooled
MIDAS	**-0.1961**	-0.0163	**0.7317**	0.6224	**0.7128**	0.5157
HAR	-0.4000	0	0.8564	0.6123	3.3431	0.5095
SHAR	-0.3995	0.0019	0.8562	0.6112	3.4558	0.5082
HEXP	-0.6146	0.0067	0.9880	0.6083	6.1034	0.5035
HARQ	-0.6117	0.0154	0.9864	0.6028	6.7636	0.5044
OLSRV	-0.7815	0.0241	1.0906	0.5975	10.3853	0.4974
OLSIV	-1.3102	0.0058	1.4148	0.6089	13.2184	0.5014
OLSRVIV	-1.6054	0.0286	1.5958	0.5949	17.3691	0.4920
OLSVPOS	-2.1049	0.0809	1.9021	0.5627	46.0299	0.4272
OLSALL	-2.7719	**0.0835**	2.3113	**0.5612**	48.1355	**0.4246**

Table 2.6: Individual vs Pooled Fit, US Stocks

Note: This is a continuation of Table 2.5, where the forecasting exercise is done on the set of US stocks. We observe a similar pattern that the pooled fits dominate the individual fits for all the models and all the metrics. (Winsorized) MSE* and QLike* increase compared to the S&P 500 universe for the pooled fit, and there are even larger increases in the individual fit. This is probably due to the fact that the stocks with small market capitalizations are harder to predict. MIDAS achieves the best performance for individual fit for the reason that the hyper-parameter is tuned using the pooled data. OLSALL is still the best-performing pooled model for R2 and MSE*, and for QLike*.

with the inclusion of more features. One notable distinction is the performance of MIDAS in the individual approach, which stands out. This could be attributed to the fact that the hyperparameter in MIDAS is tuned using the pooled data. The gap between the individual approach and pooled approach is now wider in the US stock universe.

To examine the impact of the “Insanity Filter”, we compare the MSEs before and after applying the filter in Tables 2.7 and 2.8. In the individual approach, where separate models are created for each stock using their own data, the Insanity Filter is frequently triggered and significantly improves out-of-sample performance by correcting outliers. For example, in the S&P 500 universe, the MSE for HAR decreases by 12.30%, and the MSE for OLSALL decreases by 41.32%. In contrast, the pooled approach has only a few trigger counts, resulting in minimal change in MSE. In the US universe, we observe even more trigger counts for the individual approach, with a 51.62% reduction in MSE for HAR and an 82.56% reduction in MSE for OLSALL. The pooled approach still has few trigger counts, resulting in little change in MSE. While the Insanity Filter proves crucial in eliminating potential outliers and enhancing forecasting performance for individual stocks, it may not be necessary for large datasets. However, its inclusion does not harm the analysis. Despite the correction by the Insanity Filter, the individual approach still underperforms the pooled approach prior to correction. Due to computational constraints, we did not analyze the individual approach for machine learning models, and it is likely that it will underperform.

We perform the modified Diebold-Mariano test on selected models to assess the significance of differences in forecasting performance, following the approach in Gu, Kelly, and Xiu (2020). The results are reported in Tables 2.9 and 2.10 for the S&P 500 and US universes, respectively. We utilize squared errors to calculate the loss differentials and compute cross-sectional averages, along with the corresponding mean and Newey-West standard error. In the tables, a positive value indicates that the model in the corresponding row outperforms the model in the corresponding column, as we subtract the loss from the row model from

		Individual MSE			Pooled MSE	
Model	Pre	Counts	Post	Pre	Counts	Post
RW	0.7466	203	0.7484	0.7466	207	0.7484
MIDAS	**0.5182**	4	0.5182	0.5105	5	0.5106
HAR	0.5875	474	0.5152	0.5033	2	0.5033
SHAR	0.5856	474	0.5135	0.5015	3	0.5015
HEXP	0.7768	4308	0.5315	0.4995	0	0.4995
HARQ	0.6167	2075	**0.5117**	0.4864	10	0.4865
OLSRV	1.1684	8171	0.5229	0.4817	2	0.4817
OLSIV	0.8369	4549	0.6503	0.4825	0	0.4825
OLSRVIV	0.9964	5436	0.6629	0.4673	3	0.4673
OLSVPOS	1.7648	13962	0.6131	0.4259	9	0.4260
OLSALL	1.3240	10707	0.7770	0.4182	4	0.4182
LASSO	-	-	-	0.4191	4	0.4191
PCR	-	-	-	0.4195	3	0.4195
RF	-	-	-	0.4346	0	0.4346
GBRT	-	-	-	0.4392	2	0.4392
NN6	-	-	-	0.4148	12	0.4149
NN4	-	-	-	0.4137	33	0.4140
NN2	-	-	-	0.4136	27	0.4138
NN6E	-	-	-	0.4121	12	0.4122
NN4E	-	-	-	0.4110	29	0.4111
NN2E	-	-	-	**0.4109**	25	**0.4110**

Table 2.7: Insanity Filter, S&P 500 Stocks

Note: This table reports the effect of having an insanity filter on the out-of-sample forecasting performance for the S&P 500 universe. It reports the MSE for each model before passing the predictions to an insanity filter (Pre), the counts of triggering the insanity filter (Counts), and the MSE after the insanity filter (Post). The left panel reports the individual fit while the right panel reports the pooled fit, and we only report the pooled fit for the machine learning models due to computation constraints. It is clear that the insanity filter improves the out-of-sample performance for individual OLS fits as the MSE decreases by a large magnitude when the counts are large, except for MIDAS. For the pooled fits, the insanity filter rarely triggers, and as a result, the MSE does not change much. The takeaway is that the insanity filter greatly improves individual fits, but they still perform much worse compared to the pooled fit. The best-performing column is highlighted in bold font.

	Individual MSE			Pooled MSE		
Model	Pre	Counts	Post	Pre	Counts	Post
RW	0.9725	1029	0.9746	0.9144	1138	0.9167
MIDAS	**0.7595**	4145	**0.7335**	0.6231	48	0.6232
HAR	1.7746	83630	0.8585	0.6130	89	0.6133
SHAR	1.7624	83251	0.8583	0.6118	91	0.6121
HEXP	5.6763	337759	0.9902	0.6089	101	0.6092
HARQ	3.3154	213536	0.9884	0.6033	150	0.6038
OLSRV	8.5373	524750	1.0925	0.5980	172	0.5985
OLSIV	8.5821	549885	1.4168	0.6096	28	0.6097
OLSRVIV	8.9928	583681	1.5978	0.5955	85	0.5957
OLSVPOS	14.1244	966067	1.9041	0.5641	207	0.5637
OLSALL	13.2639	925902	2.3131	0.5625	160	0.5620
LASSO	-	-	-	0.5632	169	0.5628
PCR	-	-	-	0.5633	117	0.5631
RF	-	-	-	0.5610	0	0.5610
GBRT	-	-	-	0.5702	0	0.5702
NN6	-	-	-	0.5500	10	0.5500
NN4	-	-	-	0.5460	84	0.5462
NN2	-	-	-	0.5452	87	0.5453
NN6E	-	-	-	0.5457	11	0.5457
NN4E	-	-	-	**0.5434**	55	**0.5436**
NN2E	-	-	-	0.5440	72	0.5442

Table 2.8: Insanity Filter, US Stocks

Note: This is a continuation of Table 2.7, where the forecasting exercise is done on the set of US stocks. We observe a similar pattern that the insanity filter is triggered much more frequently in the individual fits, and it improves the performance, yet there is a large gap to the pooled fits. For the US stocks, the insanity filter is triggered even more frequently compared to S&P 500 stocks for the individual fit, yet the pooled fit is rather robust as it has few trigger counts, and the MSE does not change much as a result.

the loss of the column model. For the S&P 500 stocks in Table 2.9, we observe large and significant test statistics as we increase the number of features from HAR to OLSRV, OLSVPOS, and OLSALL, thus validating the effectiveness of feature construction. Lasso and PCR significantly underperform OLSALL, while RF and GBRT perform considerably worse. On the other hand, all three neural network architectures exhibit significantly better performance than OLSALL. However, pairwise comparisons among the neural network architectures do not yield significant differences in performance. The ensemble neural networks even outperform the original networks, with large positive test statistics against OLSALL. Similar patterns are observed for the US stocks in Table 2.10, with the exception of RF, which exhibits comparable results to OLSALL. Neural networks once again demonstrate strong significance against OLSALL, while the ensemble neural network further enhances the performance.

Model	HAR	OLSRV	OLSVPOS	OLSALL	LASSO	PCR	RF	GBRT	NN6	NN4	NN2	NN6E	NN4E
OLSRV	11.6	-	-	-	-	-	-	-	-	-	-	-	-
OLSVPOS	18.3	17.1	-	-	-	-	-	-	-	-	-	-	-
OLSALL	17.7	17.3	9.0	-	-	-	-	-	-	-	-	-	-
LASSO	17.6	16.9	7.2	-2.3	-	-	-	-	-	-	-	-	-
PCR	17.6	16.8	6.8	-3.1	-1.9	-	-	-	-	-	-	-	-
RF	15.7	14.1	-6.6	-14.2	-14.6	-13.9	-	-	-	-	-	-	-
GBRT	14.8	11.9	-8.9	-12.9	-13.9	-13.4	-4.4	-	-	-	-	-	-
NN6	16.2	15.9	6.4	3.2	3.7	3.9	13.2	11.6	-	-	-	-	-
NN4	18.0	17.6	10.6	5.3	5.7	6.0	16.9	15.6	1.0	-	-	-	-
NN2	18.7	18.4	10.4	4.0	4.1	4.4	14.5	13.8	0.9	0.3	-	-	-
NN6E	18.5	18.2	12.1	6.5	6.4	6.6	17.1	15.4	2.6	3.5	3.4	-	-
NN4E	18.3	18.2	11.7	8.2	7.8	8.0	18.5	15.6	4.4	7.1	4.4	2.8	-
NN2E	18.4	18.3	11.9	8.0	7.6	7.8	18.0	15.5	4.2	6.3	5.5	3.7	0.9

Table 2.9: Diebold-Mariano Tests, S&P 500 Stocks

Note: This table displays the modified Diebold-Mariano test statistics for the out-of-sample forecasting performance of various models using panel data for the S&P 500 universe. The loss differential is calculated using squared errors, and the cross-sectional average is taken. A positive test statistic indicates that the model in the corresponding row performs better than the model in the corresponding column. Among the ordinary least square methods, OLSALL demonstrates the best performance. However, the ensemble neural networks outperform all other competing models.

Model	HAR	OLSRV	OLSVPOS	OLSALL	LASSO	PCR	RF	GBRT	NN6	NN4	NN2	NN6E	NN4E
OLSRV	12.0	-	-	-	-	-	-	-	-	-	-	-	-
OLSVPOS	18.0	18.0	-	-	-	-	-	-	-	-	-	-	-
OLSALL	18.0	18.2	10.4	-	-	-	-	-	-	-	-	-	-
LASSO	17.4	17.1	1.6	-2.4	-	-	-	-	-	-	-	-	-
PCR	17.2	16.8	0.9	-3.2	-3.3	-	-	-	-	-	-	-	-
RF	20.6	20.5	3.0	0.9	2.2	2.5	-	-	-	-	-	-	-
GBRT	17.4	15.1	-7.0	-8.4	-7.8	-7.4	-17.0	-	-	-	-	-	-
NN6	21.2	19.6	8.8	7.4	7.0	7.2	6.7	12.2	-	-	-	-	-
NN4	20.2	21.3	20.9	20.1	16.8	17.2	12.7	18.2	2.5	-	-	-	-
NN2	18.4	19.3	16.2	16.3	15.6	16.0	11.0	15.1	2.1	1.0	-	-	-
NN6E	23.1	22.0	12.6	10.9	10.1	10.2	10.1	16.1	14.3	0.4	-0.1	-	-
NN4E	21.1	22.4	25.4	24.7	20.1	20.5	15.1	20.2	4.4	13.7	2.8	1.5	-
NN2E	22.8	23.3	20.6	18.1	15.6	15.7	14.6	21.2	6.9	2.8	1.0	2.2	-0.8

Table 2.10: Diebold-Mariano Tests, US Stocks

Note: This is a continuation of Table 2.9, where the forecasting exercise is conducted on the set of US stocks. A positive test statistic indicates that the model corresponding to the row outperforms the model corresponding to the column. We observe a similar pattern to Table 2.9, where OLSALL with all features is the best performing ordinary least square method, and the ensemble neural networks significantly outperform all other models.

In Table 2.11, we provide a detailed examination of the features considered in this paper for the set of S&P 500 stocks. The features are divided into 12 groups, each labeled accordingly. The groups include MIDAS, SHAR, HARQ, and HEXP, which represent additional features compared to the HAR model. IV represents implied-volatility based features, EAD represents earnings announcement indicators, OVN represents overnight return features, VoSVoP represents volume-based features (turnover and dollar volume), Ret represents daily return features, MkC represents market capitalization features, and ETF represents sector ETFs. The table includes the number of features in each group, ranging from 1 to 102, reported in the nFeatures column. We report the in-sample Newey-West t-statistics for each group based on the OLSALL model, and all groups show statistical significance. Additionally, we present the out-of-sample relative R2 when augmenting the group of features to the HAR model in the R2(Inclusion) column. It is observed that, except for MkC, all groups exhibit additional forecasting power compared to the HAR model. Furthermore, we assess the impact of removing each group of features from the OLSALL model. With the exception of MIDAS, all other groups show a decrease in forecasting performance when removed. It is worth noting that even after removing MIDAS from the model, the out-of-sample relative R2 (0.1698) remains significantly lower than that of neural networks, such as 0.1780 in NN2. Additionally, when considering only the intercept, the R2 value drops to -2.0332, indicating poor forecasting performance. Based on these findings, it can be concluded that the earnings announcements indicators (EAD) are likely the most important group of features, second only to the HAR model, in terms of their contribution to forecasting accuracy.

Similar observations can be made for US stocks in Table 2.12, where each group contributes to improved forecasting performance with significant in-sample t-statistics. It is evident that earnings announcements indicators are the most important group of features, in addition to the HAR model, in terms of their impact on forecasting accuracy.

We examine the influential features by ranking them according to the variable importance as in Gu, Kelly, and Xiu (2020). Specifically, we calculate the reduction in out-of-sample predictive R^2 when setting all values of a particular feature to its mean value in the training data, while keeping the model estimates and other features fixed. We then normalize the values so that they sum to one in each model. Figure 2.2 for S&P 500 stocks illustrates that the models generally agree on the most influential features. The top four features across all the models considered are daily and weekly realized volatility (RV, RVw) and daily and weekly realized quarticity (RVQ, RVwQ). Following closely is the earnings announcement indicator (EAD0), which consistently ranks high in importance. Besides various weighted average of past RVs, the daily, weekly, and monthly dollar volumes (VoPd, VoPw, VoPm) as well as the sector ETFs (ETFd, ETFw, ETFm) also have a significant impact on forecasting. One interesting observation in the GBRT model is that the rankings of IV-based features are surprisingly high compared to other models. This might potentially explain the differences in performance between the models. When examining the US stocks in Figure 2.3, we observe

Group	nFeature	AvgT	MaxT	MinT	R2(Inclusion)	R2(Exclusion)
HAR	4	97.89	150.41	46.38	0	0.1614
MIDAS	1	54.49	-	-	0.0037	**0.1698**
SHAR	1	140.40	-	-	0.0036	0.1687
HARQ	4	38.19	67.66	14.90	0.0334	0.1606
HEXP	5	25.92	42.03	10.75	0.0088	0.1690
IV	102	4.33	141.32	0.02	0.0415	0.1537
EAD	5	111.05	**182.53**	45.80	0.0922	0.0901
OVN	8	56.89	158.45	2.98	0.0179	0.1677
VoSVoP	8	16.25	48.67	1.22	0.0104	0.1645
Ret	4	21.06	32.56	5.25	0.0157	0.1665
MkC	1	**149.57**	-	-	-0.0000	0.1690
ETF	4	79.25	130.02	**46.47**	0.0177	0.1629
OLSALL	**143**	5.99	168.93	0.01	**0.1691**	-2.0332

Table 2.11: Feature Estimation by Group, S&P 500 Stocks

Note: This table presents the in-sample t-statistics and out-of-sample R2 for various feature groups considered in this paper on the set of S&P 500 stocks. The features are divided into 12 groups, and the column nFeature reports the number of features in each group. The groups are labeled as follows: MIDAS, SHAR, HARQ, and HEXP indicate additional features compared to the HAR model. IV represents implied-volatility based features, EAD represents earnings announcement indicators, OVN represents overnight return features, VoSVoP represents volume-based features, Ret represents daily return features, MkC represents market capitalization feature, and ETF represents sector ETFs. We present the in-sample average t-statistics (AvgT), maximum t-statistics (MaxT), and minimum t-statistics (MinT) when running the OLSALL regression. If there is only one feature in the group, the MaxT and MinT values are skipped. The R2(Inclusion) reports the out-of-sample relative R2 when augmenting the group of features to the HAR model, while the R2(Exclusion) reports the out-of-sample relative R2 when excluding the group of features in the OLSALL model (the intercept is always kept). From the table, we observe that the inclusion of each group of features is significant in-sample and provides additional forecasting power compared to the HAR model. Conversely, excluding these features results in a decrease in forecasting power.

Group	nFeature	AvgT	MaxT	MinT	R2(Inclusion)	R2(Exclusion)
HAR	4	235.98	364.48	**104.46**	0	0.0776
MIDAS	1	138.29	-	-	0.0043	**0.0840**
SHAR	1	311.10	-	-	0.0019	0.0824
HARQ	4	75.77	122.97	42.85	0.0154	0.0782
HEXP	5	56.31	102.44	15.99	0.0077	0.0827
IV	102	11.10	363.23	0.02	0.0058	0.0809
EAD	5	246.43	357.63	102.25	0.0432	0.0426
OVN	8	134.82	**379.47**	7.30	0.0123	0.0812
VoSVoP	8	28.67	101.04	0.65	0.0044	0.0806
Ret	4	38.65	66.89	4.08	0.0058	0.0816
MkC	1	**364.52**	-	-	0.0004	0.0834
ETF	4	208.57	336.35	102.30	0.0084	0.0786
OLSALL	**143**	12.67	303.78	0.03	**0.0835**	-3.4874

Table 2.12: Feature Estimation by Group, US Stocks

Note: Continuing from Table 2.11, which focuses on the estimation conducted on the set of US stocks, we observe that the inclusion of each group of features to the HAR model leads to an increase in the out-of-sample R2. Conversely, excluding any one of these groups (except MIDAS) results in a decrease in the out-of-sample R2. Additionally, we note that the in-sample average t-statistics within each group are significantly large.

increases in variable importance for the quarterly RV and realized quarticity (RVq, RVqQ), and this could be attributed to the volatile nature of inclusion of small stocks.

To obtain an overview of the feature rankings across models, we calculate the importance of each feature for each model in each test year and sum the ranks. The features are then ordered based on their ranks across models, with the highest-ranked features at the top and the lowest-ranked features at the bottom. The overall rankings are visualized in Figure 2.4 and 2.5, where darker colors indicate higher ranks within each model, as in Gu, Kelly, and Xiu (2020). Besides the overall agreement on the top features, we see that neural networks put more emphasis on daily/weekly/monthly/quarterly returns (Retd, Retw, Retm, Retq). The effect of the earnings announcement indicator fades quickly, as indicated by the low rankings of EAD2 and EAD-2 (2 days after and 2 days prior to), despite the top ranking of EAD0.

In Table 2.13, we examine the effect of varying the number of years used for training on different models for the S&P 500 stocks. The minimum number of years for training

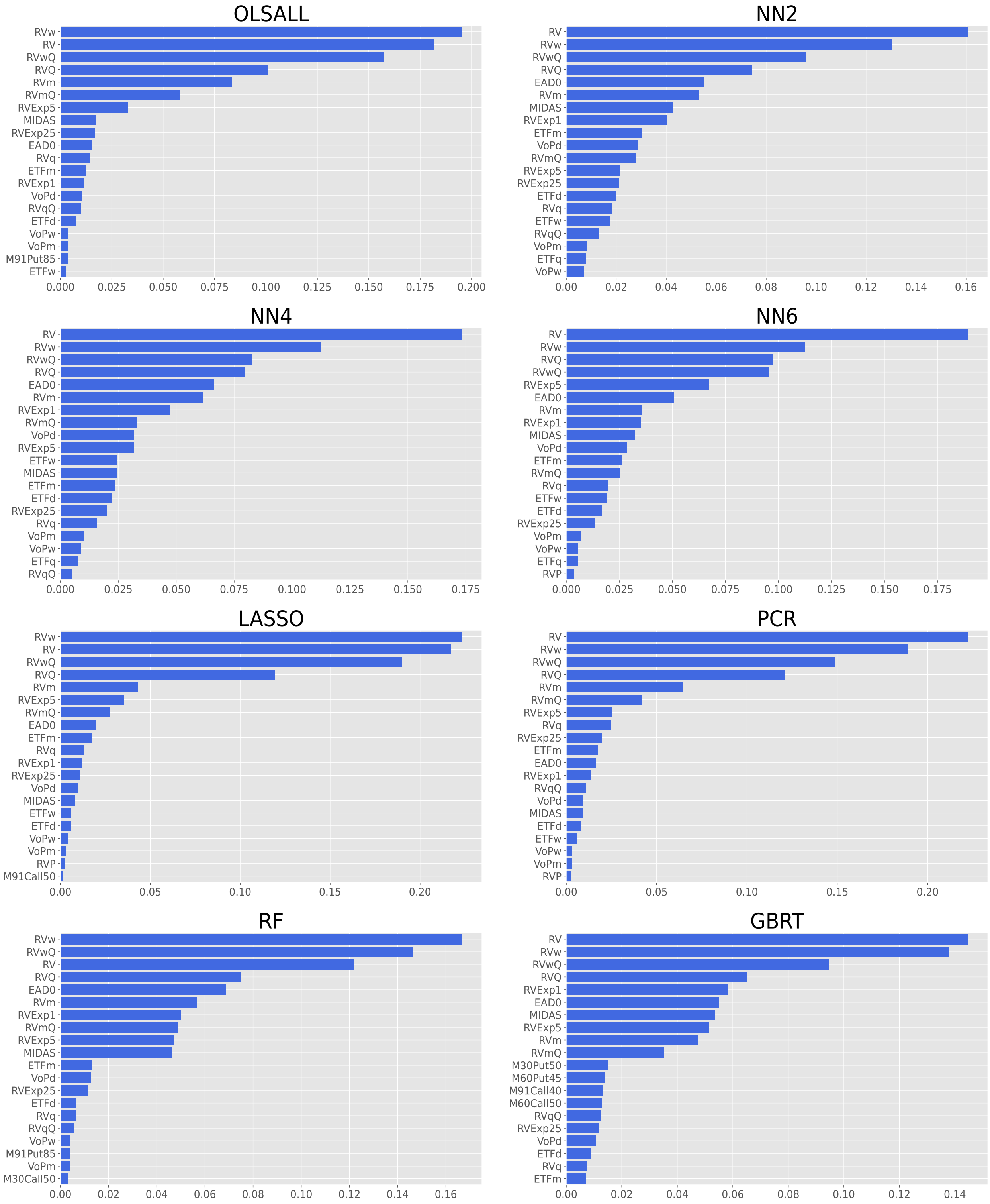

Figure 2.2: Variable Importance By Model, S&P 500 Stocks

Note: The variable importance for the top 20 most influential features for S&P 500 stocks in each model is presented, the variable importance values are normalized within each model to sum to one.

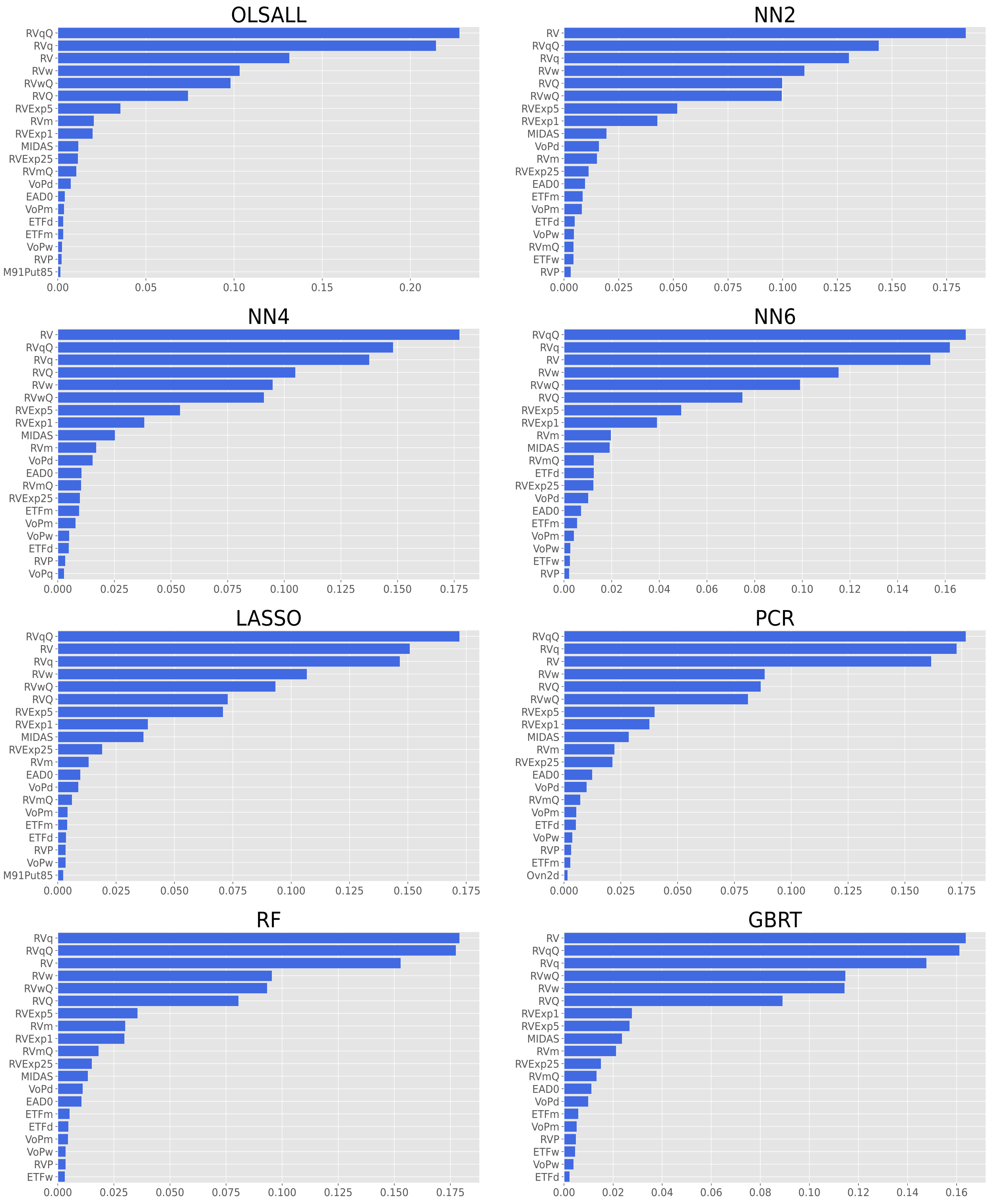

Figure 2.3: Variable Importance By Model, US Stocks

Note: The variable importance for the top 20 most influential features for US stocks in each model is presented, the variable importance values are normalized within each model to sum to one.

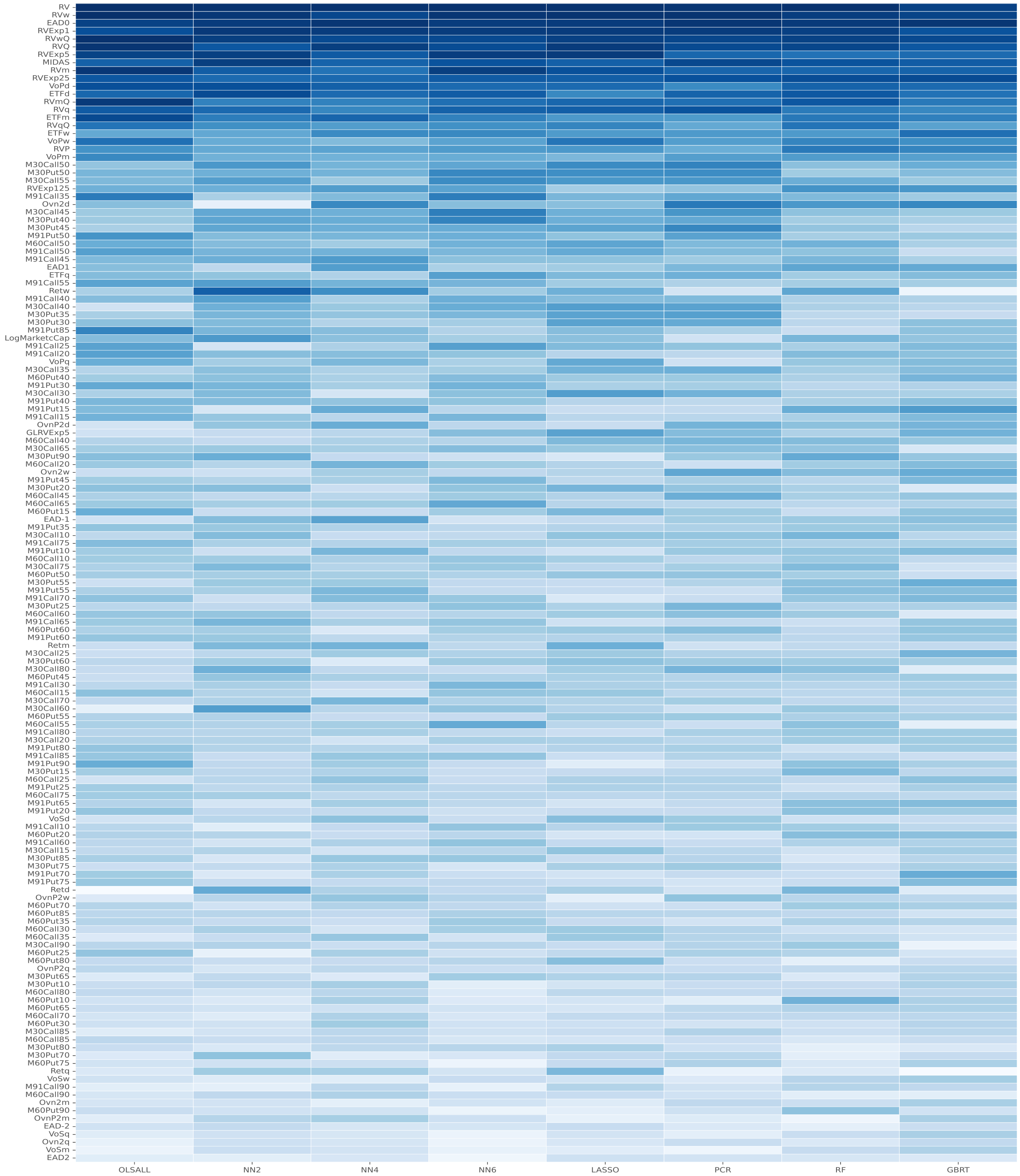

Figure 2.4: Variable Importance Ranking, S&P 500 Stocks

Note: We calculate the importance of each feature for each model in each test year and sum the ranks for the S&P 500 stocks. The features are then ordered based on their ranks across models, with the highest-ranked features at the top and the lowest-ranked features at the bottom, with the color gradient indicates the relative rankings within each model.

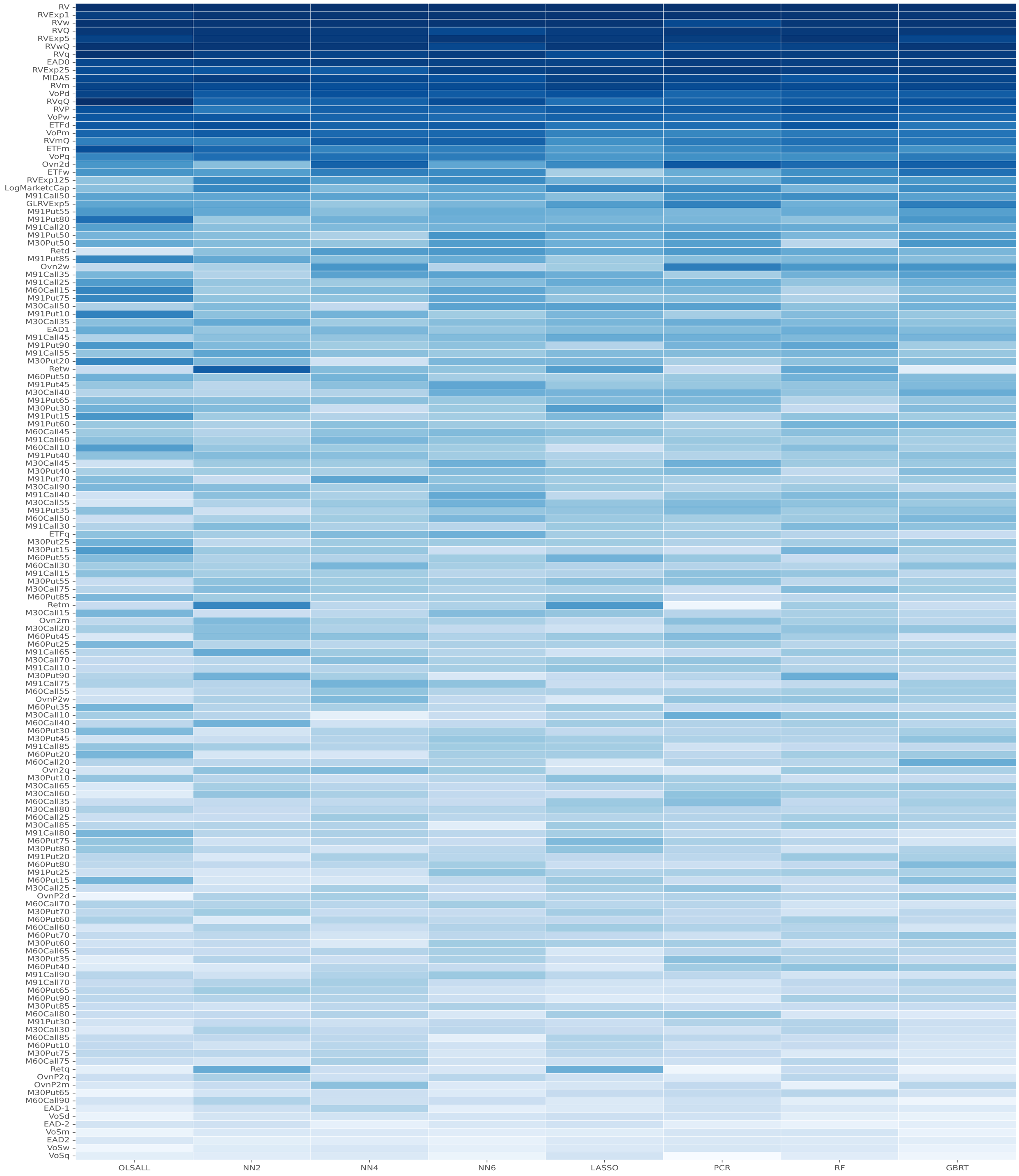

Figure 2.5: Variable Importance Ranking, US Stocks

Note: This is a continuation of Table 2.4, where the importance and rankings are calculated for the US stocks.

is one, while the maximum is ten. We report the relative R2, benchmarked against the HAR model using five years of training data. The rankings among the OLS models remain relatively consistent, with OLSALL producing the best out-of-sample results. Although the HAR model tends to favor more years of training, OLSALL achieves the best out-of-sample performance using five years of training data. For neural networks, as long as there is a sufficient amount of training data, they consistently outperform all the OLS models. The forecasting performance is generally not highly sensitive to the choice of years of training data.

	Number of Training Years				
Model	1	4	5	6	10
MIDAS	-0.0137	-0.0148	-0.0144	-0.0144	-0.0140
HAR	-0.0021	-0.0014	0	0.0002	0.0016
SHAR	0.0008	0.0022	0.0036	0.0039	0.0053
HEXP	0.0055	0.0062	0.0075	0.0075	0.0091
HARQ	0.0308	0.0326	0.0334	0.0334	0.0338
OLSRV	0.0276	0.0410	0.0430	0.0432	0.0445
OLSIV	0.0410	0.0404	0.0415	0.0414	0.0398
OLSRVIV	0.0624	0.0702	0.0716	0.0715	0.0708
OLSVPOS	0.1430	0.1524	0.1537	0.1533	0.1510
OLSALL	**0.1618**	0.1683	0.1691	0.1686	0.1653
NN6	0.1285	0.1697	0.1757	0.1757	0.1765
NN4	0.1526	0.1712	0.1776	0.1755	**0.1808**
NN2	0.1438	**0.1726**	**0.1780**	**0.1804**	0.1768

Table 2.13: Relative R2 by Training Years, S&P 500 Stocks

Note: This table presents the out-of-sample forecasting performance of various models using different numbers of training years, measured by relative R2. The benchmark relative R2 is calculated using 5 years of training data with the HAR model. Generally, the HAR model benefits from more years of training data, while the OLSALL model achieves the best performance with 5 years of data. When there is only one year of training data, neural networks underperform compared to OLSALL. However, as the number of years for training increases, neural networks consistently outperform all the OLS methods.

We investigate the impact of changing the random seeds when training different neural network models, as shown in Tables 2.14 and 2.15. Randomness arises from initializing the weights and biases of the networks, as well as shuffling the training data for stochastic gradient descent. It is observed that the forecasting performance is quite robust to the choice of random seed. All seeds yield decent relative R2 values, which are higher than 0.1691 from OLSALL for the S&P 500 stocks in Table 2.14, and higher than 0.0835 from OLSALL for the US stocks in Table 2.15. The ensemble of neural networks trained with different seeds further enhances the performance.

	Random Seeds					
Model	2020	2021	2022	2023	2024	Ensemble
NN6	0.1735	0.1745	0.1769	0.1757	0.1739	**0.1810**
NN4	0.1786	0.1804	0.1760	0.1776	0.1784	**0.1832**
NN2	0.1782	0.1764	0.1792	0.1780	0.1793	**0.1835**

Table 2.14: Relative R2 by Random Seeds, S&P 500 Stocks

Note: This table showcases the out-of-sample forecasting performance of neural network models that are initialized with different random seeds, as well as their ensemble model. The relative R2 is used as the performance measure. The best performing model in each row is highlighted in bold, and it is achieved by the ensemble model, which takes the simple average of predictions from each individual network. In general, the forecasting performance varies as the random seed is changed, but all models outperform the OLSALL model with a relative R2 of 0.1691. Furthermore, utilizing the ensemble approach further enhances the performance.

	Random Seeds					
Model	2020	2021	2022	2023	2024	Ensemble
NN6	0.1051	0.1061	0.1040	0.1032	0.1024	**0.1102**
NN4	0.1083	0.1098	0.1094	0.1094	0.1070	**0.1136**
NN2	0.1061	0.1054	0.1069	0.1108	0.1080	**0.1127**

Table 2.15: Relative R2 by Random Seeds, US Stocks

Note: Continuing from Table 2.14, where the forecasting exercise is conducted on the set of US stocks, we observe a similar pattern. There are variations in performance when different random seeds are used, but all models perform better than the OLSALL model with a relative R2 of 0.0835. Notably, the ensemble of these models achieves the best performance.

2.5 Utility Benefits

To quantify the benefits of having a better forecasting model, we consider the utility-based framework by Bollerslev, Hood, Huss, and Pedersen (2018) that an investor with mean-variance preferences investing in an asset with time-varying volatility and a constant sharpe ratio. Specifically, denote $u(\cdot)$ as the utility function of an investor, and W_t and W_{t+1} as the wealth at time t and $t+1$. The investor is deciding the proportion of wealth, x_t, at time t to invest in a risky asset with return r_{t+1}, and the remaining proportion, $1-x_t$, in the risk-free asset with return r_t^f, to maximize the expected utility at time t,

$$\max_{x_t} \mathbb{E}_t \left[u(W_{t+1}) \right] \tag{2.18}$$

$$\text{subject to } W_{t+1} = W_t \left[1 + x_t r_{t+1} + (1-x_t) r_t^f \right] \tag{2.19}$$

Write $e_{t+1}^e \equiv r_{t+1} - r_t^f$, by a Taylor series expansion of $u(\cdot)$ and ignoring the higher order terms and constants that depends only on time-t variables, we may rewrite expected utility function as,

$$\begin{aligned} U(x_t) &\approx W_t \left[x_t \mathbb{E}_t(r_{t+1}^e) - \frac{\gamma}{2} x_t^2 Var_t(r_{t+1}^e) \right] \\ &= W_t \left[x_t \mathbb{E}_t(r_{t+1}^e) - \frac{\gamma}{2} x_t^2 \mathbb{E}_t(RV_{t+1}) \right] \end{aligned} \tag{2.20}$$

where $\gamma \equiv \frac{-u'' W_t}{u'}$ is the relative risk aversion of the investor. The optimal portfolio to maximize the investor's expected utility would be

$$x_t^* = \frac{\mathbb{E}_t(r_{t+1}^e)}{\gamma \mathbb{E}_t(RV_{t+1})} = \frac{SR/\gamma}{\sqrt{\mathbb{E}_t(RV_{t+1})}}, \tag{2.21}$$

where $SR \equiv \frac{\mathbb{E}_t(r_{t+1}^e)}{\sqrt{\mathbb{E}_t(RV_{t+1})}}$ is the conditional Sharpe ratio that is assumed to be constant. This x_t^* may be interpreted as "volatility-timing", because the expected standard deviation of the resulting portfolio equals SR/γ, and the investor would adjust the position based on the predicted volatility $\sqrt{\mathbb{E}_t(RV_{t+1})}$ to reach the target. Let $\mathbb{E}_t(\cdot)$ denote the expectations from the true (unknown) risk model, while $\mathbb{E}_t^{\theta}(\cdot)$ denote expectation from a particular risk model θ, we may plug in the feasible x_t^{θ} portfolio decision in the expected utility function divided by W_t to get the per unit of wealth utility

$$UoW_t^{\theta} \equiv \frac{U(x_t^{\theta})}{W_t} = \frac{SR^2}{\gamma}\left[\frac{\sqrt{\mathbb{E}_t(RV_{t+1})}}{\sqrt{\mathbb{E}_t^{\theta}(RV_{t+1})}} - \frac{\mathbb{E}_t(RV_{t+1})}{2\mathbb{E}_t^{\theta}(RV_{t+1})}\right] \tag{2.22}$$

The maximum utility is achieved when $\mathbb{E}_t^{\theta}(RV_{t+1}) = \mathbb{E}_t(RV_{t+1})$ with value $\frac{SR^2}{2\gamma}$, which is the proportion of wealth an investor is willing to give to access the x_t^* portfolio instead of simply investing in the risk-free asset. We follow Bollerslev, Hood, Huss, and Pedersen (2018) and set $SR = 0.4$ and $\gamma = 2$, and $\frac{SR^2}{2\gamma} = 4\%$ consequently. The expected per unit of wealth utility can be empirically evaluated by averaging over the out-of-sample forecasting periods.

$$UoW^{\theta} = \frac{1}{T}\sum_{t=1}^{T}\frac{SR^2}{\gamma}\left[\frac{\sqrt{RV_{t+1}}}{\sqrt{\mathbb{E}_t^{\theta}(RV_{t+1})}} - \frac{RV_{t+1}}{2\mathbb{E}_t^{\theta}(RV_{t+1})}\right]. \tag{2.23}$$

Table 2.16 presents the results of implementing the "volatility-timing" portfolio using various prediction models for the S&P 500 stocks, measured in basis points. The "Ideal" column represents the maximum utility achievable by employing an infeasible look-into-the-future strategy, which amounts to 400 basis points. The pooled HAR model demonstrates a 4.5 basis points advantage over the individual HAR model. Furthermore, the pooled OLSALL model outperforms the pooled HAR model by 38.6 basis points. The neural networks exhibit an advantage of 39.2 to 40.1 basis points over the pooled HAR model, while the ensemble neural networks achieves 40.0 to 40.4 basis points. It is important to note that, since the utility function has an upper bound, and for portfolio allocation x_t near x_t^*, the derivative $U'(x_t)$ is close to 0. Therefore, the first-order difference in forecasting performance only results in a second-order difference when prediction models perform well (Bollerslev, Hood, Huss, and Pedersen (2018)). When considering the percentiles, the advantage of pooling and utilizing all the features is more significant in the lower percentiles. In fact, all individual OLS models yield negative realized utility under the 1st percentile. Interestingly, the random forest model, despite not being the top-performing model, achieves the highest realized utility at the 1st percentile.

Table 2.17 presents the results for US stocks, which include more volatile small stocks, leading to more erratic numbers. In fact, the simple averages of the realized utility for all individual OLS models are negative, with only MIDAS showing positive winsorized realized utility. The OLSALL model now exhibits a 27.8 basis points advantage over the pooled HAR model, while the neural networks show a 30.8 to 31.8 basis points advantage over the pooled HAR model. Once again, the random forest model shows promise for the lowest percentiles. For the 99th percentile, Lasso, OLSALL, and neural networks are all very close to each other. Overall, the neural networks deliver the best performance.

To further investigate the impact of size on realized utility and the underlying forecasting performance, we divided the set of stocks into five quintiles based on market capitalization (size) and analyzed the realized utility within each quintile. We considered two weighting schemes for aggregating the individual stocks' realized utility: one based on equal weighting, as shown in the left panels of Tables 2.18 and 2.19, and the other weighted based on size in the right panels. The components of each quintile were selected for each test year based on the corresponding size prior to the test year, and the weighted realized utility was then averaged across the test years. For the S&P 500 stocks in Table 2.18, we observed a gradual increase in realized utility for all models, starting from the portfolio with the smallest stocks to the portfolio with the largest stocks. The neural networks produced the best realized utility in each quintile for both equal-weighted and size-weighted portfolios.

When analyzing Table 2.19, the differences among quintiles become significantly more

Model	Mean	1%	5%	25%	Median	75%	99%	Mean*	Ideal
MIDAS Ind	276.8	-13.5	153.8	248.7	284.1	313.1	352.3	277.1	400
HAR Ind	273.6	-95.3	139.0	247.4	284.4	312.6	352.2	275.8	400
SHAR Ind	274.6	-142.1	137.6	247.2	284.5	312.9	352.8	276.0	400
HEXP Ind	239.8	-307.5	122.0	244.2	284.3	311.4	350.5	273.6	400
HARQ Ind	215.6	-1115.9	134.6	246.3	286.2	313.5	353.2	269.4	400
OLSRV Ind	187.0	-1071.5	105.1	244.1	284.8	312.6	353.6	268.3	400
OLSIV Ind	201.4	-2826.1	-131.6	210.1	273.9	307.5	350.8	231.1	400
OLSRVIV Ind	181.3	-3135.7	-263.1	206.5	271.9	307.0	350.6	220.4	400
OLSVPOS Ind	-112.5	-	-459.4	261.4	302.5	324.2	356.2	45.0	400
OLSALL Ind	-475.4	-	-1566.0	202.9	276.8	311.1	355.6	-323.0	400
MIDAS	278.4	-26.2	146.3	250.5	286.2	315.6	352.9	278.8	400
HAR	278.1	4.1	151.5	251.5	286.6	314.8	351.4	278.4	400
SHAR	278.4	7.4	151.3	252.4	286.6	315.5	351.7	278.7	400
HEXP	278.6	13.0	149.7	251.3	287.8	314.4	351.6	278.9	400
HARQ	281.0	-18.0	155.5	253.7	290.0	317.4	354.3	281.3	400
OLSRV	281.7	-14.5	156.0	254.9	291.3	317.4	355.0	282.0	400
OLSIV	288.0	38.0	187.7	263.0	295.9	317.6	353.0	288.4	400
OLSRVIV	289.6	31.3	184.0	265.4	297.9	319.1	354.2	290.0	400
OLSVPOS	313.9	61.6	232.5	296.8	321.6	337.6	362.7	314.5	400
OLSALL	316.7	68.5	238.6	300.8	324.0	338.7	362.8	317.1	400
LASSO	316.4	75.8	238.5	300.5	323.9	338.4	362.6	316.8	400
PCR	316.4	77.6	237.7	300.4	323.8	338.4	362.6	316.7	400
RF	312.4	**108.8**	232.6	295.7	319.8	334.3	360.1	313.2	400
GBRT	313.8	74.7	236.0	296.9	320.4	334.8	360.3	314.7	400
NN6	318.1	54.3	**238.8**	303.3	326.1	**340.2**	362.8	318.9	400
NN4	317.3	59.5	234.3	302.1	325.3	339.3	362.8	318.2	400
NN2	317.8	62.6	233.8	303.4	326.0	339.6	363.1	318.7	400
NN6E	318.4	65.5	236.3	303.5	326.1	339.9	**363.5**	319.2	400
NN4E	318.1	64.7	233.8	303.0	326.2	340.2	362.6	319.0	400
NN2E	**318.4**	65.5	233.3	**303.6**	**326.5**	340.1	362.8	**319.2**	400

Table 2.16: Realized Utility for S&P 500 Stocks

Note: This table displays the realized utility, measured in basis points, of executing the volatility-timing portfolio on the S&P 500 stocks. The “Ideal” column represents the upper bound that could be achieved through an optimal yet infeasible trading strategy, assuming perfect knowledge of future realized volatility. The table reports the average, 1st percentile, 25th percentile, median, 75th percentile, 99th percentile, and winsorized average at 1st percentile of the realized utility. The best performing column, if unique, is highlighted in bold font. The upper panel shows the results of fitting ordinary least squares models using individual stock data, while the middle and lower panels show the results of fitting both ordinary least squares models and machine learning models to the pooled data. The results indicate that the pooled fit performs significantly better than the individual fit, especially for the lower percentiles.

Model	Mean	1%	5%	25%	Median	75%	99%	Mean*	Ideal
MIDAS Ind	-36.6	-3442.0	-521.8	152.1	244.8	289.8	343.6	202.1	400
HAR Ind	-3200.8	-	-2570.3	100.3	234.0	285.2	342.0	-713.8	400
SHAR Ind	-3076.6	-	-2350.9	100.6	234.5	285.2	342.7	-853.8	400
HEXP Ind	-5715.7	-	-	-26.1	220.8	282.2	343.2	-2354.9	400
HARQ Ind	-4706.1	-	-	-23.6	219.9	280.9	343.0	-2747.2	400
OLSRV Ind	-7428.2	-	-	-239.8	201.6	277.2	343.6	-3814.8	400
OLSIV Ind	-8310.3	-	-	-3817.0	-309.3	189.1	330.3	-6272.0	400
OLSRVIV Ind	-	-	-	-4643.1	-445.6	170.9	330.0	-8009.2	400
OLSVPOS Ind	-	-	-	-8349.8	-140.0	261.1	340.6	-	400
OLSALL Ind	-	-	-	-	-1376.5	99.9	328.2	-	400
MIDAS	254.4	-344.7	71.2	214.0	266.3	303.8	349.2	257.0	400
HAR	256.4	-329.0	78.8	219.6	267.9	303.7	348.8	258.7	400
SHAR	256.8	-339.2	81.2	220.2	268.3	304.1	349.0	259.2	400
HEXP	258.7	-314.4	82.1	222.1	269.6	304.4	349.5	260.7	400
HARQ	257.7	-331.5	78.1	219.6	268.8	305.5	350.2	260.0	400
OLSRV	260.3	-335.0	81.1	222.6	271.2	306.5	351.4	262.3	400
OLSIV	260.0	-266.5	97.9	225.1	270.1	305.2	349.4	261.9	400
OLSRVIV	262.8	-274.0	94.2	226.3	272.8	307.3	351.6	264.5	400
OLSVPOS	282.9	-328.3	104.4	249.9	295.0	323.3	356.9	284.8	400
OLSALL	284.2	-253.8	111.7	251.7	295.5	323.5	357.3	285.9	400
LASSO	283.9	-259.6	112.5	251.5	295.5	323.6	**357.7**	285.8	400
PCR	283.6	-273.6	110.1	251.5	295.5	323.6	357.7	285.7	400
RF	282.6	**-225.5**	**118.4**	251.2	293.2	321.2	355.3	284.9	400
GBRT	281.6	-250.3	106.3	247.7	293.1	320.8	353.4	283.9	400
NN6	287.7	-250.8	117.7	255.4	298.8	325.6	357.2	290.2	400
NN4	**288.2**	-260.8	116.5	**256.9**	**300.3**	326.6	357.2	**291.2**	400
NN2	287.1	-303.9	111.1	256.3	300.1	**326.7**	357.1	290.8	400
NN6E	288.1	-259.0	114.8	255.8	299.3	325.8	356.7	290.4	400
NN4E	287.9	-281.5	115.8	256.4	299.8	326.3	356.7	291.0	400
NN2E	287.2	-288.7	113.4	255.9	299.4	326.1	356.9	290.2	400

Table 2.17: Realized Utility for US Stocks

Notes: This table presents the realized utility, expressed in basis points, from executing the volatility-timing portfolio on the set of US stocks, and is a continuation of Table 2.16. Entries with values lower than -10000 basis points are not displayed. The mean column is greatly affected by stocks that fall below the 5th percentile for the upper panel, resulting in negative realized utility for the individual OLS models.

	Equal Weighted					Size Weighted				
Model	Q1	Q2	Q3	Q4	Q5	Q1	Q2	Q3	Q4	Q5
MIDAS Ind	259.2	271.5	281.1	283.0	289.3	260.0	271.6	280.7	280.6	290.7
HAR Ind	260.0	270.9	280.4	276.2	280.4	260.7	271.0	279.9	274.8	286.6
SHAR Ind	259.8	270.7	280.7	279.0	282.9	260.6	270.9	280.3	277.3	288.3
HEXP Ind	207.7	202.7	275.6	245.0	267.9	227.9	192.5	275.1	249.2	279.0
HARQ Ind	109.0	201.0	278.2	229.2	259.0	113.6	205.9	278.2	237.3	258.1
OLSRV Ind	246.0	11.5	251.0	223.3	204.5	251.5	21.1	248.3	232.4	228.7
OLSIV Ind	176.9	113.1	244.6	212.2	261.3	165.3	96.6	238.9	211.1	273.6
OLSRVIV Ind	184.8	146.6	114.6	252.5	209.5	182.5	138.9	98.5	248.9	191.5
OLSVPOS Ind	-273.7	-175.9	-389.7	144.7	139.2	-403.4	-187.8	-345.9	164.8	116.9
OLSALL Ind	-177.1	-348.7	-1159.7	-364.3	-316.0	-211.8	-442.4	-1155.7	-312.7	-627.4
MIDAS	258.4	272.4	283.4	284.8	292.6	259.3	272.5	283.0	282.5	294.2
HAR	260.8	272.7	282.1	283.7	290.8	261.3	272.7	281.5	281.4	291.9
SHAR	261.0	272.8	282.4	284.3	291.3	261.5	272.9	281.8	282.0	292.4
HEXP	261.2	274.0	282.8	284.1	290.7	262.0	274.2	282.2	281.8	292.0
HARQ	264.4	276.0	285.6	286.0	292.7	264.8	275.9	284.9	283.6	293.8
OLSRV	264.9	277.2	286.4	286.8	293.0	265.4	277.2	285.8	284.5	294.3
OLSIV	276.0	284.0	291.8	291.3	297.0	276.2	283.9	291.3	289.4	298.8
OLSRVIV	276.9	286.1	294.1	292.7	298.1	277.2	286.0	293.6	290.7	300.1
OLSVPOS	305.4	313.4	317.9	313.8	319.0	304.3	312.5	316.9	310.7	322.1
OLSALL	308.3	315.6	320.2	316.4	322.8	307.3	314.6	319.4	313.7	326.6
LASSO	308.1	315.3	319.8	316.1	322.4	307.3	314.4	319.0	313.5	326.2
PCR	308.0	315.2	319.8	316.0	322.8	307.1	314.3	318.9	313.4	326.6
RF	303.6	311.4	316.0	311.3	319.6	302.0	310.4	315.1	308.5	322.6
GBRT	304.1	312.6	317.8	312.6	321.7	302.7	311.4	317.0	309.7	324.9
NN6	309.9	317.2	321.4	**318.6**	323.5	**308.1**	316.1	320.7	**315.8**	327.8
NN4	308.5	316.7	320.5	316.1	324.2	306.5	315.5	319.4	312.8	327.9
NN2	309.2	317.1	321.1	317.1	324.6	307.0	315.8	320.1	313.7	328.3
NN6E	309.8	317.5	**321.6**	318.0	324.9	307.8	316.4	**320.9**	315.0	**328.8**
NN4E	309.8	317.5	321.1	317.0	324.8	307.9	316.4	320.0	313.8	328.5
NN2E	**310.0**	**317.8**	321.5	317.8	**324.9**	308.1	**316.6**	320.4	314.7	328.6

Table 2.18: Realized Utility on Size Portfolios for S&P 500 Stocks

Note: This table displays the average realized utility, expressed in basis points, of executing the "volatility-timing" trading strategy on 5 quintiles of S&P 500 stocks sorted by their market capitalization (size). The best performing model in each column is highlighted in bold font. The sorting is based on the available stocks' size before each forecasting year, where Q1 represents the 1st quintile consisting of stocks with the smallest size in the set, and so on for Q2, Q3, Q4, and Q5, which consist of the largest stocks. The left panel shows an equal weighting strategy for the stocks within each quintile, while the right panel weights the stocks' realized utility by their previous year's size. As the size of the stocks increases, we generally observe an increase in the realized utility for all models, indicating more accurate predictions. Further, we almost observe uniform improvements of realized utility for neural networks over the OLSALL in every quintile. The neural networks have the best performing models within each portfolio.

	Equal Weighted					Size Weighted				
Model	Q1	Q2	Q3	Q4	Q5	Q1	Q2	Q3	Q4	Q5
MIDAS Ind	-387.4	-157.1	249.9	265.8	276.2	-1575.9	-190.8	243.0	262.4	284.3
HAR Ind	-6509.9	-5298.4	-1183.1	-1002.7	49.3	-	-5013.3	-1110.6	-664.9	186.1
SHAR Ind	-6180.9	-5494.8	-1223.7	-757.7	13.4	-9518.1	-5408.2	-1096.1	-475.0	167.8
HEXP Ind	-	-9732.1	-4164.9	-1136.0	9.6	-	-9730.0	-3857.8	-1049.5	172.6
HARQ Ind	-	-5734.6	-4154.3	-1230.3	-558.5	-9642.4	-6066.2	-3927.0	-1011.1	-157.4
OLSRV Ind	-	-8639.5	-7388.2	-1973.9	-471.6	-	-8652.5	-6874.2	-1900.2	-200.7
OLSIV Ind	-	-	-8774.1	-3281.4	-676.2	-	-	-8557.9	-3200.1	-46.5
OLSRVIV Ind	-	-	-	-6527.6	-814.0	-	-	-	-8227.8	-225.3
OLSVPOS Ind	-	-	-	-8129.6	-3166.4	-	-	-	-8089.4	-1225.2
OLSALL Ind	-	-	-	-8373.8	-4004.5	-	-	-	-9271.0	-3174.9
MIDAS	214.9	256.4	258.1	267.9	280.3	202.3	248.3	253.5	264.8	289.3
HAR	224.3	259.2	261.6	267.1	277.6	211.7	250.6	257.6	263.6	286.1
SHAR	225.0	259.6	262.0	267.5	277.9	212.5	250.9	258.0	264.1	286.5
HEXP	227.2	261.9	264.2	268.7	278.6	214.9	253.7	260.3	265.5	286.8
HARQ	219.8	260.3	264.2	270.1	280.6	206.5	251.0	260.2	266.6	289.0
OLSRV	222.3	263.3	267.5	272.5	282.2	209.1	254.2	263.6	269.3	290.3
OLSIV	229.2	264.3	265.2	269.0	277.3	217.7	257.1	261.4	265.6	285.0
OLSRVIV	225.7	267.2	270.0	273.6	281.4	213.5	259.6	266.4	270.4	288.7
OLSVPOS	235.0	281.7	295.4	301.9	305.8	223.0	274.2	291.9	298.3	310.4
OLSALL	236.3	283.6	**296.3**	302.2	306.1	225.2	277.2	**292.9**	298.8	311.1
LASSO	235.1	283.7	296.1	302.1	305.7	225.3	277.3	292.7	298.6	310.5
PCR	233.3	283.7	296.2	302.1	305.7	225.0	277.3	292.8	298.6	310.4
RF	240.3	280.2	289.4	299.8	306.2	229.9	273.4	285.0	296.4	312.4
GBRT	232.7	278.0	289.7	301.5	309.4	223.9	271.1	285.0	298.2	316.2
NN6	246.4	284.7	295.8	305.6	311.6	233.2	277.1	291.3	301.8	318.8
NN4	**247.3**	**286.2**	293.9	306.5	313.1	**234.0**	**278.1**	288.6	302.5	320.2
NN2	244.8	285.6	291.2	**306.8**	312.9	231.4	276.7	285.1	302.8	319.3
NN6E	246.5	285.4	295.8	305.9	312.2	233.3	277.7	291.2	302.1	319.3
NN4E	246.0	285.9	292.6	306.7	**313.5**	232.7	278.0	287.0	**302.9**	**320.4**
NN2E	244.4	285.2	293.0	306.2	312.7	230.9	276.9	287.6	302.3	319.4

Table 2.19: Realized Utility on Size Portfolios for US Stocks

Note: This table displays the average realized utility, expressed in basis points, of executing the "volatility-timing" trading strategy on 5 quintiles of US stocks that are sorted by their market capitalization (size), and is a continuation of Table 2.18. Entries with values lower than -10000 basis points are not displayed, while the best performing model in each column is highlighted in bold font. In the upper panel, most OLS models with individual fit now show huge negative realized utility, indicating that the agent would prefer investing in the risk-free asset instead, except for MIDAS where the tuning is done over the pooled data. As we go down the quintiles, smaller stocks are associated with lower realized utility, indicating that they are harder to predict.

pronounced. In fact, several individual OLS models generate extremely negative realized utility values (lower than -10,000 basis points), suggesting that investing in the risk-free asset would have been a better option. The only exception to this is the MIDAS model, which is specifically tuned using panel data. The first quintile exhibits considerably lower realized utility compared to the other four quintiles for both the pooled OLS models and machine learning models. This indicates that predicting the performance of stocks with smaller market capitalizations is particularly challenging. Neural networks generally demonstrate a larger advantage in the first and fifth quintiles, a smaller advantage in the second and fourth quintiles, while the OLSALL model prevails in the third quintile, albeit by a small margin.

2.6 Conclusion

In conclusion, we applied machine learning algorithms to the task of realized volatility forecasting on two very large sets of stocks. By pooling the data of all stocks and fitting one universal model, we found that utilizing all the features produces the best ordinary least squares model, which is very stable and produces few extreme predictions, unlike the individual models. However, the neural network, with the same set of features as input, yielded an even better prediction model and significantly outperformed all others. The difference in performance was larger when we focused on the set of US stocks that includes those with smaller market capitalization, and we showed large economic gains in switching from individual OLS models to pooled OLS models and to neural networks.

An accurate volatility forecasting model is useful in estimating the volatility risk premium, which is the difference between the risk-neutral expectation of volatility derived from option prices and the actual expectation of volatility (Bollerslev, Patton, and Quaedvlieg (2016)). In fact, Conrad and Loch (2015) and Bekaert and Hoerova (2014) use many OLS-based realized volatility forecasting models to estimate the volatility risk premium, and the choice of forecasting model matters in both the numerical result and interpretation. Since we provide an even better forecasting model than the OLS-based methods, we can obtain a more accurate estimate of the volatility risk premium and gain deeper insights into the underlying economic story. Additionally, one may combine the forecast of expected return (Gu, Kelly, and Xiu (2020)) with the forecast of realized volatility to study a much broader set of investors' behavior beyond the agent considered here, which we leave for future work.

References

Abadi, M., P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. (2016): "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pp. 265–283.

Ahoniemi, K., and M. Lanne (2013): "Overnight stock returns and realized volatility," *International Journal of Forecasting*, 29(4), 592–604.

Aït-Sahalia, Y., J. Fan, and D. Xiu (2010): "High-Frequency Covariance Estimates with Noisy and Asynchronous Financial Data," *Journal of the American Statistical Association*, 105(492), 1504–1517.

Aït-Sahalia, Y., and J. Jacod (2014): *High Frequency Financial Econometrics*. Princeton University Press.

Aït-Sahalia, Y., and D. Xiu (2017): "Using Principal Component Analysis to Estimate a High Dimensional Factor Model with High-Frequency Data," *Journal of Econometrics*, 201(2), 384–399.

——— (2019a): "A Hausman test for the presence of market microstructure noise in high frequency data," *Journal of Econometrics*, 211(1), 176–205.

——— (2019b): "Principal component analysis of high-frequency data," *Journal of the American Statistical Association*, 114(525), 287–303.

Andersen, T. G., T. Bollerslev, F. X. Diebold, and P. Labys (2003): "Modeling and forecasting realized volatility," *Econometrica*, 71(2), 579–625.

Atilgan, Y. (2014): "Volatility spreads and earnings announcement returns," *Journal of Banking & Finance*, 38, 205–215.

Bai, J. (2003): "Inferential Theory for Factor Models of Large Dimensions," *Econometrica*, 71(1), 135–171.

Bai, J., and S. Ng (2002): "Determining the Number of Factors in Approximate Factor Models," *Econometrica*, 70(1), 191–221.

——— (2006): "Evaluating Latent and Observed Factors in Macroeconomics and Finance," *Journal of Econometrics*, 131(1), 507–537.

Barndorff-Nielsen, O. E., P. R. Hansen, A. Lunde, and N. Shephard (2011): "Multivariate Realised Kernels: Consistent Positive Semi-Definite Estimators of the Covariation of Equity Prices with Noise and Non-Synchronous Trading," *Journal of Econometrics*, 162(2), 149–169.

Barndorff-Nielsen, O. E., and N. Shephard (2002): "Econometric analysis of realized volatility and its use in estimating stochastic volatility models," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 64(2), 253–280.

Barth, M. E., and E. C. So (2014): "Non-diversifiable volatility risk and risk premiums at earnings announcements," *The Accounting Review*, 89(5), 1579–1607.

Bekaert, G., and M. Hoerova (2014): "The VIX, the variance premium and stock market volatility," *Journal of econometrics*, 183(2), 181–192.

Bibinger, M. (2012): "An Estimator for the Quadratic Covariation of Asynchronously Observed Itô Processes with Noise: Asymptotic Distribution Theory," *Stochastic Processes and their Applications*, 122, 2411–2453.

Bibinger, M., N. Hautsch, P. Malec, and M. Reiss (2014): "Estimating the Quadratic Covariation Matrix from Noisy Observations: Local Method of Moments and Efficiency," *Annals of Statistics*, 42(4), 80–114.

Bickel, P. J., and E. Levina (2008a): "Regularized Estimation of Large Covariance Matrices," *Annals of Statistics*, 36(1), 199–227.

——— (2008b): "Covariance Regularization by Thresholding," *Annals of Statistics*, 36(6), 2577–2604.

Bollerslev, T., B. Hood, J. Huss, and L. H. Pedersen (2018): "Risk everywhere: Modeling and managing volatility," *The Review of Financial Studies*, 31(7), 2729–2773.

Bollerslev, T., A. J. Patton, and R. Quaedvlieg (2016): "Exploiting the Errors: A Simple Approach for Improved Volatility Forecasting," *Journal of Econometrics*, 192(1), 1–18.

Breiman, L. (1996): "Bagging predictors," *Machine learning*, 24, 123–140.

Breiman, L., J. Friedman, R. Olshen, and C. Stone (1984): "Classification and Regression Trees," .

Brogaard, J. A., T. Hendershott, and R. Riordan (2014): "High Frequency Trading and Price Discovery," *Review of Financial Studies*, 27, 2267–2306.

Brownlees, C. T., E. Nualart, and Y. Sun (2017): "Realized Networks," Discussion paper, Universitat Pompeu Fabra - Department of Economics and Business; Barcelona Graduate School of Economics (Barcelona GSE).

Bucci, A. (2020): "Realized volatility forecasting with neural networks," *Journal of Financial Econometrics*, 18(3), 502–531.

Busch, T., B. J. Christensen, and M. Ø. Nielsen (2011): "The role of implied volatility in forecasting future realized volatility and jumps in foreign exchange, stock, and bond markets," *Journal of Econometrics*, 160(1), 48–57.

Cai, T., and W. Liu (2011): "Adaptive Thresholding for Sparse Covariance Matrix Estimation," *Journal of the American Statistical Association*, 106(494), 672–684.

Cao, S. S., and G. S. Narayanamoorthy (2012): "Earnings volatility, post–earnings announcement drift, and trading frictions," *Journal of Accounting Research*, 50(1), 41–74.

Carhart, M. M. (1997): "On Persistence in Mutual Fund Performance," *The Journal of Finance*, 52(1), 57–82.

Chamberlain, G., and M. Rothschild (1983): "Arbitrage, Factor Structure, and Mean-Variance Analysis on Large Asset Markets," *Econometrica*, 51(5), 1281–1304.

Chollet, F., et al. (2015): "Keras," `https://keras.io`.

Christensen, B. J., and N. R. Prabhala (1998): "The relation between implied and realized volatility," *Journal of financial economics*, 50(2), 125–150.

Christensen, K., S. Kinnebrock, and M. Podolskij (2010): "Pre-Averaging Estimators of the Ex-Post Covariance Matrix in Noisy Diffusion Models with Non-Synchronous Data," *Journal of Econometrics*, 159(1), 116–133.

Connor, G., M. Hagmann, and O. Linton (2012): "Efficient Semiparametric Estimation of the Fama–French Model and Extensions," *Econometrica*, 80(2), 713–754.

Connor, G., and O. Linton (2007): "Semiparametric Estimation of a Characteristic-Based Factor Model of Common Stock Returns," *Journal of Empirical Finance*, 14(5), 694–717.

Conrad, C., and K. Loch (2015): "The variance risk premium and fundamental uncertainty," *Economics Letters*, 132, 56–60.

Corsi, F. (2009): "A simple approximate long-memory model of realized volatility," *Journal of Financial Econometrics*, 7(2), 174–196.

Croux, C., E. Renault, and B. Werker (2004): "Dynamic Factor Models," *Journal of Econometrics*, 119, 223–230.

CRSP (2021): "Data Description Guide CRSP US Stock & US Index Databases," `https://wrds-www.wharton.upenn.edu/documents/399/Data_Descriptions_Guide.pdf`.

Cybenko, G. (1989): "Approximation by superpositions of a sigmoidal function," *Mathematics of control, signals and systems*, 2(4), 303–314.

Da, R., and D. Xiu (2017): "When Moving-Average Models Meet High-Frequency Data: Uniform Inference on Volatility," Discussion paper, University of Chicago.

——— (2021): "Disentangling Autocorrelated Intraday Returns," *Chicago Booth Research Paper*, (21-05).

Daniel, K., and S. Titman (1997): "Evidence on the Characteristics of Cross Sectional Variation in Stock Returns," *The Journal of Finance*, 52(1), 1–33.

Diebold, F. X., and R. S. Mariano (2002): "Comparing Predictive Accuracy," *Journal of Business & Economic Statistics*, 20(1), 134–144.

Dietterich, T. G. (2000): "Ensemble methods in machine learning," in *International workshop on multiple classifier systems*, pp. 1–15. Springer.

Donoho, D. L., et al. (2000): "High-Dimensional Data Analysis: The Curses and Blessings of Dimensionality," *AMS Math Challenges Lecture*, pp. 1–32.

Doz, C., D. Giannone, and L. Reichlin (2011): "A Two-Step Estimator for Large Approximate Dynamic Factor Models Based on Kalman Filtering," *Journal of Econometrics*, 164, 188–205.

Engle, R. F., and J. R. Russell (1998): "Autoregressive Conditional Duration: a New Model for Irregularly Spaced Transaction Data," *Econometrica*, pp. 1127–1162.

Fama, E. F., and K. R. French (1993): "Common Risk Factors in the Returns on Stocks and Bonds," *Journal of Financial Economics*, 33(1), 3–56.

——— (2015): "A Five-Factor Asset Pricing Model," *Journal of Financial Economics*, 116(1), 1–22.

Fan, J., Y. Fan, and J. Lv (2008): "High Dimensional Covariance Matrix Estimation Using a Factor Model," *Journal of Econometrics*, 147(1), 186–197.

Fan, J., A. Furger, and D. Xiu (2016): "Incorporating Global Industrial Classification Standard Into Portfolio Allocation: A Simple Factor-Based Large Covariance Matrix Estimator With High-Frequency Data," *Journal of Business & Economic Statistics*, 34(4), 489–503.

Fan, J., and R. Li (2001): "Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties," *Journal of the American Statistical Association*, 96(456), 1348–1360.

Fan, J., Y. Liao, and M. Mincheva (2011): "High-Dimensional Covariance Matrix Estimation in Approximate Factor Models," *Annals of Statistics*, 39(6), 3320–3356.

——— (2013): "Large Covariance Estimation by Thresholding Principal Orthogonal Complements," *Journal of the Royal Statistical Society, B*, 75(4), 603–680, With 33 discussions by 57 authors and a reply by Fan, Liao and Mincheva.

Fan, J., J. Zhang, and K. Yu (2012): "Vast Portfolio Selection with Gross-Exposure Constraints," *Journal of the American Statistical Association*, 107(498), 592–606.

Forni, M., M. Hallin, M. Lippi, and L. Reichlin (2000): "The Generalized Dynamic-Factor Model: Identification and Estimation," *The Review of Economics and Statistics*, 82, 540–554.

——— (2004): "The Generalized Dynamic Factor Model: Consistency and Rates," *Journal of Econometrics*, 119(2), 231–255.

Forni, M., and M. Lippi (2001): "The Generalized Dynamic Factor Model: Representation Theory," *Econometric Theory*, 17, 1113–1141.

Freund, Y., and R. E. Schapire (1995): "A desicion-theoretic generalization of on-line learning and an application to boosting," in *European conference on computational learning theory*, pp. 23–37. Springer.

Friedman, J. H. (2001): "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232.

——— (2002): "Stochastic gradient boosting," *Computational statistics & data analysis*, 38(4), 367–378.

Ghysels, E., and H. Qian (2019): "Estimating MIDAS regressions via OLS with polynomial parameter profiling," *Econometrics and statistics*, 9, 1–16.

Ghysels, E., P. Santa-Clara, and R. Valkanov (2006): "Predicting volatility: getting the most out of return data sampled at different frequencies," *Journal of Econometrics*, 131(1-2), 59–95.

Gu, S., E. Holly, T. Lillicrap, and S. Levine (2017): "Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates," in *2017 IEEE international conference on robotics and automation (ICRA)*, pp. 3389–3396. IEEE.

Gu, S., B. Kelly, and D. Xiu (2020): "Empirical asset pricing via machine learning," *The Review of Financial Studies*, 33(5), 2223–2273.

Hansen, L. K., and P. Salamon (1990): "Neural network ensembles," *IEEE transactions on pattern analysis and machine intelligence*, 12(10), 993–1001.

Hansen, P. R., and A. Lunde (2006): "Consistent ranking of volatility models," *Journal of Econometrics*, 131(1-2), 97–121.

Hasbrouck, J. (2007): *Empirical Market Microstructure*. Oxford University Press, New York, NY.

Hastie, T., R. Tibshirani, J. H. Friedman, and J. H. Friedman (2009): *The elements of statistical learning: data mining, inference, and prediction*, vol. 2. Springer.

Hautsch, N., L. M. Kyj, and R. C. Oomen (2012): “A blocking and regularization approach to high-dimensional realized covariance estimation,” *Journal of Applied Econometrics*, 27(4), 625–645.

Hornik, K., M. Stinchcombe, and H. White (1989): “Multilayer feedforward networks are universal approximators,” *Neural networks*, 2(5), 359–366.

Ikeda, S. S. (2016): “A Bias-Corrected Estimator of the Covariation Matrix of Multiple Security Prices when both Microstructure Effects and Sampling Durations are Persistent and Endogenous,” *Journal of Econometrics*, 193(1), 203–214.

Jacod, J., Y. Li, P. A. Mykland, M. Podolskij, and M. Vetter (2009): “Microstructure Noise in the Continuous Case: the Pre-Averaging Approach,” *Stochastic Processes and Their Applications*, 119(7), 2249–2276.

Kahn, R. N., P. Brougham, and P. Green (1998): *United States Equity Model (USE3) - Risk Model Handbook*MSCI INC.

Kalnina, I., and O. Linton (2008): “Estimating quadratic variation consistently in the presence of endogenous and diurnal measurement error,” *Journal of Econometrics*, 147, 47–59.

Kim, D., Y. Wang, and J. Zou (2016): “Asymptotic Theory for Large Volatility Matrix Estimation Based on High-Frequency Financial Data,” *Stochastic Processes and Their Applications*, 126(11), 3527–3577.

King, B. F. (1966): “Market and Industry Factors in Stock Price Behavior,” *The Journal of Business*, 39(1), 139–190.

Kingma, D., and J. Ba (2014): “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980.*

LeCun, Y., Y. Bengio, and G. Hinton (2015): “Deep learning,” *Nature*, 521(7553), 436.

Ledoit, O., and M. Wolf (2004): “Honey, I Shrunk the Sample Covariance Matrix,” *Journal of Portfolio Management*, 30(4), 110–119.

——— (2012): “Nonlinear Shrinkage Estimation of Large-Dimensional Covariance Matrices,” *Annals of Statistics*, 40(2), 1024–1060.

Lei, Q., X. W. Wang, and Z. Yan (2020): “Volatility spread and stock market response to earnings announcements,” *Journal of Banking & Finance*, 119, 105126.

Leo, B. (2001): “Random forests,” *Machine learning*, 45, 5–23.

Li, J., and D. Xiu (2016): “Generalized method of integrated moments for high-frequency data,” *Econometrica*, 84(4), 1613–1633.

Li, S. Z., and Y. Tang (2022): "Automated risk forecasting," *Available at SSRN 3776915.*

Li, Y., P. Mykland, E. Renault, L. Zhang, and X. Zheng (2014): "Realized Volatility When Sampling Times are Possibly Endogenous," *Econometric Theory*, 30, 580–605.

Liu, L. Y., A. J. Patton, and K. Sheppard (2015): "Does anything beat 5-minute RV? A comparison of realized measures across multiple asset classes," *Journal of Econometrics*, 187(1), 293–311.

Liu, M., W.-C. Choo, C.-C. Lee, and C.-C. Lee (2023): "Trading volume and realized volatility forecasting: Evidence from the China stock market," *Journal of Forecasting*, 42(1), 76–100.

Luong, C., and N. Dokuchaev (2018): "Forecasting of realised volatility with the random forests algorithm," *Journal of Risk and Financial Management*, 11(4), 61.

Martens, M. (2004): "Estimating Unbiased and Precise Realized Covariances," Discussion paper, Erasmus University Rotterdam (EUR); Robeco Asset Management.

Masters, T. (1993): *Practical neural network recipes in C++*. Morgan Kaufmann.

Moon, H. R., and M. Weidner (2015): "Linear Regression for Panel with Unknown Number of Factors as Interactive Fixed Effects," *Econometrica*, 83(4), 1543–1579.

Morgan, N., and H. Bourlard (1990): "Generalization and parameter estimation in feedforward nets: Some experiments," in *Advances in neural information processing systems*, pp. 630–637.

Onatski, A. (2010): "Determining the Number of Factors from Empirical Distribution of Eigenvalues," *Review of Economics and Statistics*, 92, 1004–1016.

Patton, A. J. (2011): "Volatility forecast comparison using imperfect volatility proxies," *Journal of Econometrics*, 160(1), 246–256.

Patton, A. J., and K. Sheppard (2015): "Good volatility, bad volatility: Signed jumps and the persistence of volatility," *Review of Economics and Statistics*, 97(3), 683–697.

Pelger, M. (2015a): "Large-Dimensional Factor Modeling Based on High-Frequency Observations," *Available at SSRN 2584172.*

——— (2015b): "Understanding Systematic Risk: A High-Frequency Approach," *Available at SSRN 2647040.*

Ravisankar, P., V. Ravi, G. R. Rao, and I. Bose (2011): "Detection of financial statement fraud and feature selection using data mining techniques," *Decision support systems*, 50(2), 491–500.

Rosenberg, B. (1974): "Extra-Market Components of Covariance in Security Returns," *Journal of Financial and Quantitative Analysis*, 9(02), 263–274.

Ross, S. A. (1976): "The Arbitrage Theory of Capital Asset Pricing," *Journal of Economic Theory*, 13(3), 341–360.

Rothman, A. J., E. Levina, and J. Zhu (2009): "Generalized Thresholding of Large Covariance Matrices," *Journal of the American Statistical Association*, 104(485), 177–186.

Scheidegger, S., and I. Bilionis (2019): "Machine learning for high-dimensional dynamic stochastic economies," *Journal of Computational Science*, 33, 68–82.

Schulman, J., B. Zoph, C. Kim, J. Hilton, J. Menick, J. Weng, J. F. C. Uribe, L. Fedus, L. Metz, M. Pokorny, et al. (2022): "ChatGPT: Optimizing language models for dialogue," *OpenAI blog*.

Shephard, N., and D. Xiu (2017): "Econometric Analysis of Multivariate Realised QML: Estimation of the Covariation of Equity Prices under Asynchronous Trading," *Journal of Econometrics*, 201(1), 19–42.

Silver, D., T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, et al. (2018): "A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play," *Science*, 362(6419), 1140–1144.

Swanson, N. R., and H. White (1997): "Forecasting economic time series using flexible versus fixed specification and linear versus nonlinear econometric models," *International journal of Forecasting*, 13(4), 439–461.

Tao, M., Y. Wang, and X. Chen (2013): "Fast Convergence Rates in Estimating Large Volatility Matrices Using High-Frequency Financial Data," *Econometric Theory*, 29(4), 838–856.

Tao, M., Y. Wang, Q. Yao, and J. Zou (2011): "Large Volatility Matrix Inference via Combining Low-Frequency and High-Frequency Approaches," *Journal of the American Statistical Association*, 106(495), 1025–1040.

Tao, M., Y. Wang, and H. H. Zhou (2013): "Optimal Sparse Volatility Matrix Estimation for High-Dimensional Itô Processes with Measurement Errors," *Annals of Statistics*, 41(4), 1816–1864.

Tibshirani, R. (1996): "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1), 267–288.

Todorov, V., and T. Bollerslev (2010): "Jumps and Betas: A New Framework for Disentangling and Estimating Systematic Risks," *Journal of Econometrics*, 157, 220–235.

Todorova, N., and M. Souček (2014): "The impact of trading volume, number of trades and overnight returns on forecasting the daily realized range," *Economic modelling*, 36, 332–340.

Varneskov, R. T. (2016): “Flat-Top Realized Kernel Estimation of Quadratic Covariation with Nonsynchronous and Noisy Asset Prices,” *Journal of Business & Economic Statistics*, 34(1), 1–22.

Wang, Y., and J. Zou (2010): “Vast Volatility Matrix Estimation for High-Frequency Financial Data,” *Annals of Statistics*, 38(2), 943–978.

Zhang, C., Y. Zhang, M. Cucuringu, and Z. Qian (2024): “Volatility forecasting with machine learning and intraday commonality,” *Journal of Financial Econometrics*, 22(2), 492–530.

Zhang, L. (2011): “Estimating Covariation: Epps effect, Microstructure Noise,” *Journal of Econometrics*, 160(1), 33–47.

Zou, H. (2006): “The Adaptive Lasso and its Oracle Properties,” *Journal of the American Statistical Association*, 101(476), 1418–1429.