A New Method for Computing Standard Errors of Risk and Performance Estimators with Serially Correlated Returns

Xin Chen

A dissertation submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2018

Reading Committee:
R. Douglas Martin, Chair
Aleksandr Y. Aravkin
Mark Kot

Program Authorized to Offer Degree:
Applied Mathematics

University of Washington

Abstract

A New Method for Computing Standard Errors of Risk and Performance Estimators with Serially Correlated Returns

Xin Chen

Chair of the Supervisory Committee:

Professor R. Douglas Martin

Applied Mathematics

It is well known that small values of unsuspected returns serial correlation result in substantially inflated standard errors of sample mean estimates of mean returns, and that use of standard error estimates based on assuming independent and identically distributed (i.i.d.) returns can result in serious under-estimation of the true standard errors. It turns out that nonparametric sample estimators of risk and performance measures may be represented as sums of influence function based nonlinear transformations of returns, thereby suffers from the same problem as sample means in the presence of serial correlation. We solve this problem by developing a new general method for computing the standard errors of risk and performance measure estimators for serially correlated returns. The method uses a frequency domain periodogram transformation of the time series of risk or performance measure influence functions as the observations for a polynomial based generalized linear model (GLM) fitting method for exponential distributions. The GLM fitting is a maximum likelihood method with elastic net regularization (glmExpEN). The latter alleviates

collinearity problems associated with polynomial regression and encourages parsimonious model selection. We provide an example of applying the method to the returns of a collection of hedge funds, and compare the results to four alternative standard error computation methods: (a) an influence function based formula for standard errors based on i.i.d. returns, (b) a bootstrap method based on the i.i.d. assumptions, (c) a bootstrap method for serially correlated data, and (d) a Newey–West method. The results show that use of our new method often improves on all three of these methods when there is serial correlation. We also evaluate the performance of our method via Monte Carlo for a number of well-known risk and performance measures for AR(1) and MA(1) returns models, and in the process compare the performance of our method with the performance to the well-known Newey–West method. The results show that the mean-squared-error performance of our new method is better than Newey–West type methods for a number of common risk and performance estimators. Our overall method, which we refer to as the seCorIF method, is implemented in the R package EstimatorStandardErrors.

In the previous chapter, we considered GLM with elastic net regularization for exponentially distributed random variables. Exponential distribution is a special case of the more general Gamma distribution, which is widely used for modeling data that has a continuous non-negative distribution, such as insurance claims and survival data. Therefore, it is highly desirable to extend glmExpEN method to work for gamma distribution. In the second chapter, we develop a variable selection method for glmGamma models using elastic net regularization (glmGammaEN), for which we provide an algorithm and implementation. The glmGammaEN model is more challenging than other more common GLMs as the likelihood function has no global quadratic upper bound, and we develop an efficient accelerated proximal gradient algorithm using a local model. We report simulation study results and discuss the choice of regularization parameter. The method is implemented in the R package glmGammaEN.

Contents

ACKNOWLEDGEMENT **11**

1 Standard Errors of Risk and Performance Estimators for Serially Correlated Returns 12

1.1 Introduction 12

1.2 Influence Functions for Risk and Performance Measures 15

1.2.1 Influence Functions for the Risk and Performance Measures Studied . . . 16

1.2.2 Two Important Properties of Influence Functions 22

1.2.3 Asymptotic Variance of Estimators via Influence Functions 23

1.3 The Spectral Density at the Origin Approach 29

1.3.1 Spectral Densities and Estimator Variance with Serial Correlation 29

1.3.2 Periodograms and their Properties 30

1.4 Periodogram Based Generalized Linear Model Method 31

1.4.1 Generalized Linear Model for Periodogram 32

1.4.2 GLM with Elastic Net Regularization 37

1.5 Two R Packages and an Example 41

1.6 Performance Simulation Studies 43

1.6.1 Root-Mean-Squared Error Performance 44

1.6.2 Simulation Performance Results Relative to the Newey–West Method . . . 51

1.6.3 SE Estimator Choice Recommendations 54

1.7 Summary and Discussion 55

Appendices **58**

1A Sharpe Ratio Influence Function Derivation 58

1B Standard Error of Sample Mean for AR(1) Returns 58

1C Newey–West Method Details for the Simulations 59

1D Cross Validation Algorithm 60

1E R Packages Installation and Section 5 Code 62

1F RMSE and Relative RMSE and AR(1) Simulation Results for Sample Size 200 . . 65

1G Bias and Standard Deviation AR(1) Simulation Results 68

1H Relative Bias and Standard Deviation Comparisons AR(1) Simulation Results . . . 75

2 Generalized Linear Model for Gamma Distributed Variables via Elastic Net Regularization **83**

2.1 Introduction 83

2.2 First-Order Methods for Regularized GLM 85

2.3 Adapting FISTA to the Gamma Family with EN Regularization 89

2.3.1 $G(\beta)$, $\nabla G(\beta)$ and Proximal of EN 89

2.3.2 Computing a Local Upper Bound 91

2.3.3 Optimal λ via Cross Validation 92

2.3.4 GLM for Gamma Response Variables with Elastic Net (glmGammaEN) . . 94

2.4 Numerical Experiment 95

2.4.1 Error of Fitted Coefficients 96

2.4.2 Variable Selection Performance Analysis 97

2.5 Expected Shortfall Standard Error of Hedge Fund Returns 99

2.6 Discussion 101

3 Practical Impact of Our Results and Future Work 103

3.1 Practical Impact of Our Results 103

3.2 Future Work 105

List of Tables

1.1 SE and ER Values Assuming i.i.d. Returns but are Actually Gaussian AR(1) for $n = 60$ and $n = \infty$ (shown in parentheses) 25

1.2 New IF Method ES Standard Errors for Hedge Fund Returns 42

1.3 Fitted glmExpEN Coefficients for Hedge Fund Returns 43

1.4 $\hat{\gamma}$ for Gaussian AR(1) Returns 61

2.1 Comparison of R implementations for GLM 84

2.2 Performance Summary of Different GLM methods (the last two methods fit regular glmGamma model with variables with non-zero coefficients) 97

2.3 Performance Summary of Different GLM methods 98

2.4 ES Standard Errors for Hedge fund Returns 100

2.5 Fitted glmGammaEN Coefficients for Hedge Fund Returns for Hedge fund Returns 101

List of Figures

1.1 Influence Functions of the Risk and Performance Measures Used in this Study . . 20

1.2 Influence Functions Time Series of CA Hedge Fund Returns 21

1.3 RMSE Comparison of New Standard Error Estimators for AR(1) Returns, $n = 60$. 47

1.4 RMSE Comparison of New Standard Error Estimators for AR(1) Returns, $n = 100$ 48

1.5 Percent RMSE Increase of seCorIFPW and HW Methods over seCorIF Method for AR(1) Returns, $n = 60$ 49

1.6 Percent RMSE Increase of HW and seCorIFPW over seCorIF for AR(1) Returns, $n = 100$ 50

1.7 RRMSE Comparison of New Standard Error Estimators for AR(1) Returns, $n = 60$ 52

1.8 RRMSE Comparison of New Standard Error Estimators for AR(1) Returns, $n = 100$ 53

1.9 RMSE Comparison of New Standard Error Estimators for AR(1) Returns, $n = 200$ 66

1.10 Percent RMSE Increase of HW and seCorIFPW over seCorIF for AR(1) Returns, $n = 200$ 67

1.11 RRMSE Comparison of New Standard Error Estimators for AR(1) Returns, $n = 200$ 68

1.12 Bias Comparison of New Standard Error Estimators for AR(1) Returns, $n = 60$ . . 70

1.13 Bias Comparison of New Standard Error Estimators for AR(1) Returns, $n = 100$ . . 71

1.14 Bias Comparison of New Standard Error Estimators for AR(1) Returns, $n = 200$ . . 72

1.15 Standard Deviation Comparison of New Standard Error Estimators for AR(1) Returns, $n = 60$ 73

1.16 Standard Deviation Comparison of New Standard Error Estimators for AR(1) Returns, $n = 100$ 74

1.17 Standard Deviation Comparison of New Standard Error Estimators for AR(1) Returns, $n = 200$ 75

1.18 S.E. Estimators Relative Biases for AR(1) Returns, $n = 60$ 77

1.19 S.E. Estimators Relative Biases for AR(1) Returns, $n = 100$ 78

1.20 S.E. Estimators Relative Biases for AR(1) Returns, $n = 200$ 79

1.21 S.E. Estimators Relative Standard Deviation for AR(1) Returns, $n = 60$ 80

1.22 S.E. Estimators Relative Standard Deviation for AR(1) Returns, $n = 100$ 81

1.23 S.E. Estimators Relative Standard Deviation for AR(1) Returns, $n = 200$ 82

2.1 Histogram of number of zero coefficients selected over 1000 simulations 99

3.1 Optimal Bias Robust Psi Function with 99% Efficiency 109

3.2 Time Series of the Coefficients for Size Factor using Robust Cross-Sectional Regression 110

ACKNOWLEDGEMENT

It has been such a wonderful journey for me to complete this Ph.D. study. I have been given the privilege to spend the six years in the Department of Applied Mathematics at the University of Washington, and the opportunity to work with the greatest talents of my field. The department and its people are so influential to me and will always be dear to my heart.

My first and foremost gratitude goes to my supervisor, Dr. R. Douglas Martin. His dedication and enthusiasm towards academia have been so empowering that made me simply want to follow his path. Never in my wildest dream would I believe I can become who I am today. It is Dr. Martin that has always pushed me into doing things that I normally would not do myself, that oftentimes turned out to be such a transformation on me.

Special thanks to my Ph.D. committee members, Dr. Aleksandr Y. Aravkin, Dr. Mark Kot, Dr. Jose Nathan Kutz and Dr. Les Atlas, for their support, guidance and valuable inputs on my research work. My deepest heartfelt thanks and appreciation for their time and help in reviewing my work. I would also like to thank Dr. Eric Zivot for serving on the general exam committee and providing comments and suggestions on my research work.

I would like to thank the Google Summer of Code (GSoC) projects for providing summer funding in 2016 and 2017 where I was able to turn my theoretical PhD research work into open source software packages that are shared among both academic and industrial users. The GSoC mentors, Mr. Brian G. Peterson and Mr. Peter Carl, have been particularly helpful during this process by providing industrial insights on how to build the software packages.

Lastly and most importantly, I want to thank my family. My wife Xiaoyue Cathy Liu is my best friend, and the love of my life. I cannot tell you how lucky I am to find you, and thank you for the tremendous love and support you give me every single day. My parents, Gang Chen and Mei Jin, I thank you for your unconditional love, and I dedicate this dissertation to you.

Chapter 1

Standard Errors of Risk and Performance Estimators for Serially Correlated Returns

1.1 Introduction

Risk and performance measure estimators have become standard tools in decision-making processes in the context of portfolio optimization, risk management and performance assessment. Furthermore, statistical methods are available to assess the variability of such estimators in terms of their standard errors when returns are independent and identically distributed. The two most common general methods are the use of estimator asymptotic variance formulas to get approximate finite-sample standard error estimates, and the use of bootstrap methods that can also provide distribution approximations. However, both methods can fail rather badly when there is serial correlation in returns. For example, the results of Lo (2002) show how optimistic a Sharpe ratio may be due to under-estimation of volatility in the presence of serial correlation.

The best known method of dealing with serial correlation in computing standard errors and associated confidence intervals and *t*-tests was introduced in the econometrics literature by Newey

and West (1987) and further developed in Newey et al. (1994) and Andrews (1991), Andrews and Monahan (1992) and Hirukawa (2010). The Newey–West class of methods estimates the variance of the error term in regression models with a kernel-weighted sum of lag-l covariance estimates of the time series of the errors. This method is very similar to the kernel-weighted Fourier transform method of estimating spectral densities in the days prior to the advent of the Fast Fourier Transform (Brillinger 2001).

Another approach for obtaining reliable variance and standard error estimators with serially correlated data is the Generalized Method of Moments (GMM) that is widely used in Econometrics. See for example Campbell et al. (1997) and Cochrane (2009). The GMM method was used by Lo (2002) to derive the standard error of Sharpe ratio estimators for serially correlated returns. While the GMM could in principle be used to derive variance and standard error estimators for any risk and performance measure estimator, this has not so far happened. This is due, at least in part, to the relative mathematical complexity of GMM methods.

There is yet a third method that works in the presence of serial correlation for estimators that can be represented as a sum of the values of a wide-sense stationary process. The method is based on the fact that the variance of such a sum is the value of the spectral density of the process at zero frequency. Some time ago, Heidelberger and Welch (1981) (HW) made use of this fact and proposed a method of estimating the spectral density at zero frequency by using regression to fit a polynomial to cleverly transformed periodograms of the time series. The HW method has two main short-comings, the first being that the method does not provide the best way to deal with the exponentially distributed periodogram values in their regression model, and the second is the lack of a good model selection method. In this chapter, we provide a general solution to the problem of constructing standard errors of risk and performance measures by estimating a spectral density at zero frequency in a way that improves on the HW method and Newey–West type methods.

Our new method of computing standard errors for serially correlated data contains two main components, the first of which is the transformation of the time series of returns to the time series of

influence function of returns. The second component of the method is the use of a maximum-likelihood generalized linear model for exponential distributions to fit a polynomial to the periodogram of the influence-function transformed returns, with an elastic net (EN) penalty, i.e., with weighted ℓ_1 and ℓ_2 coefficient norm penalty terms (Zou et al. 2005). The elastic net has a useful model selection capability with respect to the polynomial order, as well as coping with polynomial collinearity. We refer to the second component as the glmExpEN model fitting method, and refer to the overall method as the "standard errors of risk and performance estimators for serially correlated returns using influence functions" (seCorIF).

We note that the importance of using the time series of influence functions of risk and performance measure estimator is due to two critical features of influence functions: (1) An influence function provides a representation for an estimator in terms of a summation of the influence functions of the returns, and (2) The variance of the estimator is given by the variance of the sum of the influence functions, and an estimate of this variance can be obtained by estimating the spectral density at zero frequency of the times series of influence functions. For details about influence functions, see Hampel (1974), Hampel et al. (1986) and Maronna et al. (2018).

The remainder of the paper is organized as follows. Section 2 provides the definition and basic properties of influence functions, derives their formulas for six important risk and performance measure estimators, briefly reviews the Newey–West method, and discusses the use of influence functions for estimating the standard error of risk and performance estimators for serially correlated returns. Section 3 reviews basic frequency domain spectral density properties and the relationship between a sum of serially correlated returns and the value of a spectral density at zero frequency. Section 4 develops the glmExpEN method, and discusses a prewhitening version (seCorIFPW) version of the basic seCorIF method. Section 5 describes an `R` package that implements our new method, and illustrates its use in computing the standard errors of expected shortfall (ES) for 13 hedge funds. Section 6 reports the results of a Monte Carlo simulation study of the bias, standard deviation and root mean-squared-error (RMSE) performance of our seCorIF and seCorIFPW

methods as applied to six risk and performance measure estimators for first-order autoregression returns processes. Overall, our new methods outperform both the HW frequency domain methods and the Newey–West method.

1.2 Influence Functions for Risk and Performance Measures

Influence functions are very useful analytic tools for studying the influence of data on estimators, and for obtaining asymptotic variance formulas for estimators. The basic definition and first discussion of influence functions is due to Hampel (1974). Discussions of influence functions and their use in statistics is available in the books on robust statistics by Hampel et al. (1986) and Maronna et al. (2018). Applications of influence functions in quantitative finance may be found in Cont et al. (2010) and in Martin and Zhang (2017).

Influence functions are based on representations of the asymptotic value of estimators as functionals on a suitably restricted space of cumulative distribution functions F, which we refer to as "distribution functions" for short, such that the sample-based estimator is obtained by evaluating the functional at the empirical distribution F_n of the data. For example, in the case of the sample mean estimator of the unknown mean value μ of returns $r_1, r_2, \cdots, r_n$, the functional is

$$\mu(F) = \int r dF(r) \tag{1.1}$$

for all distributions such that the integral exists, and where $dF(r) = f(r)dr$ in the case of continuous distributions with probability density f . Evaluation of the above integral at an empirical distribution F_n that has a jump of height $1/n$ at each return r_t, $t = 1, 2, \cdots, n$ results in the sample mean:

$$\hat{\mu}_n = \frac{1}{n} \sum_{t=1}^{n} r_t \,. \tag{1.2}$$

For the sample volatility estimator the functional is

$$\sigma(F) = \left[\int (r - \mu(F))^2 dF(r)\right]^{1/2} \tag{1.3}$$

and one has

$$\hat{\sigma}_n = \left[\int (r - \mu(F_n))^2 dF_n(r)\right]^{1/2} = \left[\frac{1}{n}\sum_{t=1}^{n}(r_t - \hat{\mu}_n)^2\right]^{1/2}. \tag{1.4}$$

1.2.1 Influence Functions for the Risk and Performance Measures Studied

We first provide the general definition of an influence function, and then introduce the influence functions of the specific risk and performance measure estimators whose standard errors under serial correlation we study in the remainder of the paper.

Consider an estimator asymptotic functional $T = T(F)$ and a mixture distribution

$$F_\gamma(x) = (1 - \gamma)F(x) + \gamma\delta_r(x), \tag{1.5}$$

where $\delta_r(x)$ is point mass discrete distribution function with a jump of height one located at return value r. The influence function of T is defined as

$$IF(r; T, F) = \lim_{\gamma \to 0} \frac{T(F_\gamma) - T(F)}{\gamma} = \frac{d}{d\gamma} T(F_\gamma)\Big|_{\gamma=0}. \tag{1.6}$$

The influence function is a special directional derivative (a.k.a., a Gateaux derivative) of the functional T on an infinite dimensional space of distribution function, in the direction of point mass distribution δ_r evaluated at F. For details concerning influence functions and their use see Maronna et al. (2018).

Straightforward use of the above formula for the mean functional (1.1) shows that its influence function is

$$IF(r; \mu, F) = r - \mu, \tag{1.7}$$

where we abbreviate $\mu = \mu(F)$. Likewise, for the volatility functional (1.3), the influence function is

$$IF(r; \sigma, F) = (2\sigma)^{-1} \cdot \left((r-\mu)^2 - \sigma^2\right), \tag{1.8}$$

with the abbreviation $\sigma = \sigma(F)$ as well as $\mu = \mu(F)$ for notational convenience.

Throughout the remainder of the paper we will make use of the influence functions of the Sharpe ratio, Sortino ratio, value-at-risk (VaR) and expected shortfall (ES), and also provide the formulas for functionals and influence functions for these quantities.

Sharpe Ratio Functional and Influence Function

The Sharpe ratio functional is

$$\mathrm{SR}(F) = \frac{\mu(F) - r_f}{\sigma(F)}, \tag{1.9}$$

where $\mu(F)$ is given by (1.1) and $\sigma(F)$ is given by (1.3). The Sharpe ratio sample-based estimator SR_n is obtained by replacing the mean and volatility functionals by their sample-based estimators (1.2) and (1.4).

The formula for the influence function for the Sharpe ratio was derived in Zhang (2009) and is given by

$$IF(r; \mathrm{SR}) = -\frac{\mathrm{SR}}{2\sigma^2}(r-\mu)^2 + \frac{1}{\sigma}(r-\mu) + \frac{\mathrm{SR}}{2}, \tag{1.10}$$

where we have suppressed the dependence of the SR, μ and σ on the underlying distribution function. For the readers' convenience, the derivation is also provided in Appendix 1A.

Sortino Ratio Functional and Influence Function

The Sortino ratio (SoR) functional is

$$SoR(F) = \frac{\mu(F) - \mu_f}{\sigma_{ssd}(F)}, \tag{1.11}$$

where $\sigma_{ssd}(F)$ is the semi-standard deviation functional

$$\sigma_{ssd}(F) = \left(\int_{-\infty}^{\mu(F)} (r - \mu(F))^2 dF(r) \right)^{1/2}. \tag{1.12}$$

For normal distributions we have $\sigma_{ssd}(F) = \sigma/\sqrt{2}$. In order to derive the influence function for the Sortino ratio, we need the influence function for semi-standard deviation, which is shown via straightforward derivation in Martin et al. (2017) to be:

$$IF_{SSD}(r; F) = \frac{(r - \mu(F))^2 \, \mathbb{1}(r \le \mu(F)) - 2 \int_{-\infty}^{\mu(F)} (x - \mu(F)) dF(x) \cdot (r - \mu(F)) - \sigma_{ssd}^2}{2\sigma_{ssd}}. \tag{1.13}$$

For a normal distribution, $\int_{-\infty}^{\mu(F)} (x - \mu(F)) dF(x) = \sigma/\sqrt{2}$ in the above expression.

Then applying (1.6) to (1.11) and dropping the F arguments for convenience, we have

$$IF_{SoR}(r; F) = \frac{1}{\sigma_{ssd}(F)} \cdot IF_{mean}(r; \mu(F)) - \frac{\mu(F) - r_f}{\sigma_{ssd}^2(F)} \cdot IF_{SSD}(r; F). \tag{1.14}$$

Using the influence function (1.7) of the mean and the normal distribution version of (1.13), and rearranging terms, the above becomes

$$IF_{SoR}(r) = -\frac{SoR}{2\sigma_{ssd}^2}(r - \mu)^2 \cdot \mathbb{1}(r \le \mu) + \left[\frac{1}{\sigma_{ssd}(F)} + \frac{SoR \cdot \sigma}{\sqrt{2\pi} \cdot \sigma_{ssd}^2} \right](r - \mu) + \frac{SoR}{2} \tag{1.15}$$

$$= -\frac{SoR}{\sigma^2}(r - \mu)^2 \cdot \mathbb{1}(\mathrm{r} \le \mu) + \frac{1}{\sigma}\left(\sqrt{\pi} + SoR \right)(r - \mu) + \frac{SoR}{2}, \tag{1.16}$$

where $\mathbb{1}(x) = 1$ for $x \le 0$ and equals zero otherwise, and we have used the fact that $\sigma^2 = 2 \cdot \sigma_{ssd}^2$ to obtain the form in second line.

Value-at-Risk and Expected Shortfall Functionals and Influence Functions

With losses taken as a positive quantity, value-at-risk (VaR) for positive tail probability α is the negative of the quantile functional $q_\alpha(F) = \inf\{r|F(r) \geq \alpha\}$, i.e., $VaR_\alpha(F) = -q_\alpha(F)$. The latter is equal to $F^{-1}(\alpha)$ for a continuous and strictly increasing distribution function F. Let F have a probability density function f. The formula for the influence function of a quantile $q_\alpha(F)$ is well-known, see for example Hampel (1968) or Maronna et al. (2018), and is easily modified for VaR, i.e., for the negative of a quantile $q_\alpha = q_\alpha(F)$:

$$IF_{VaR_\alpha}(r; F) = \frac{1}{f(q_\alpha)} \left[\mathbb{1}(\mathrm{r} \leq q_\alpha) - \alpha\right] . \tag{1.17}$$

The expected shortfall (ES) functional is given by

$$\mathrm{ES}_\alpha(F) = -\frac{1}{\alpha} \int_{-\infty}^{q_\alpha(F)} r \cdot dF(r). \tag{1.18}$$

The above well-known expression (see for example McNeil et al. 2015) is used in Martin and Zhang (2017) to derive the following expression for an ES influence function:

$$IF_{\mathrm{ES}_\alpha}(r) = -\frac{r - q_\alpha}{\alpha} \mathbb{1}(r \leq q_\alpha) - q_\alpha - \mathrm{ES}_\alpha. \tag{1.19}$$

The influence functions of mean, standard deviation (volatility), Sharpe ratio, Sortino ratio, VaR_α and ES_α are shown in Figure 1.1 for the case $\mu = 0.01$ and $\sigma = 0.07$.

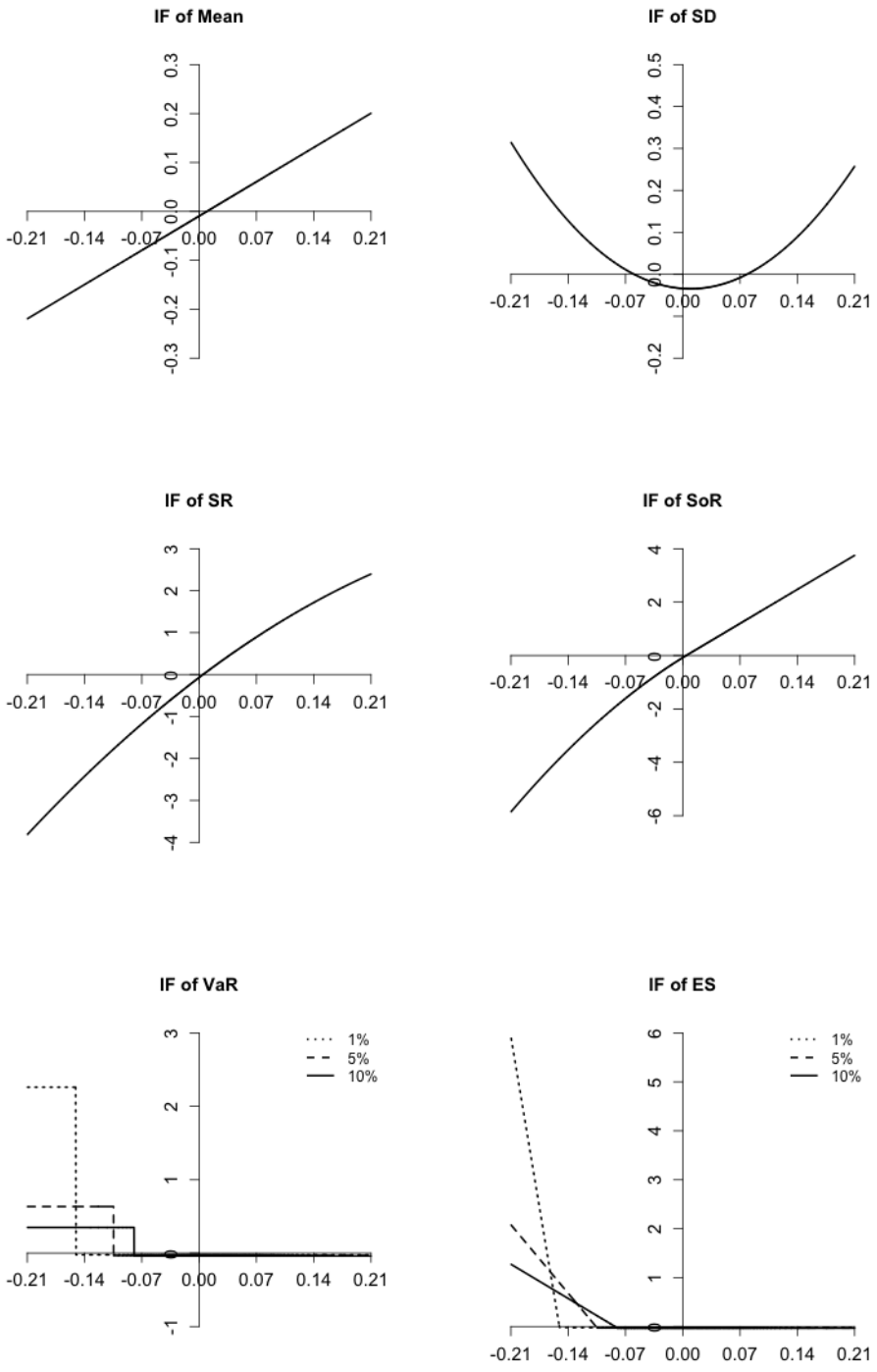

Figure 1.1: Influence Functions of the Risk and Performance Measures Used in this Study

Implementation of Influence Functions

Influence function formulas depend on distribution dependent quantities, such as returns mean value μ, standard deviation σ, and quantile q_α, and the risk or performance measure itself, such as the Sharpe ratio (SR), Sortino ratio (SoR) and expected shortfall ES_α. For use in our standard error estimation method, such unknown values are replaced by sample-based estimates, e.g., for the ES example below q_α is replaced by the empirical quantile and ES_α is replaced by the sample estimator.

Figure 1.2 shows the influence function time series for Convertible Arbitrage (CA) hedge fund monthly returns from 1997/01 to 2009/08.

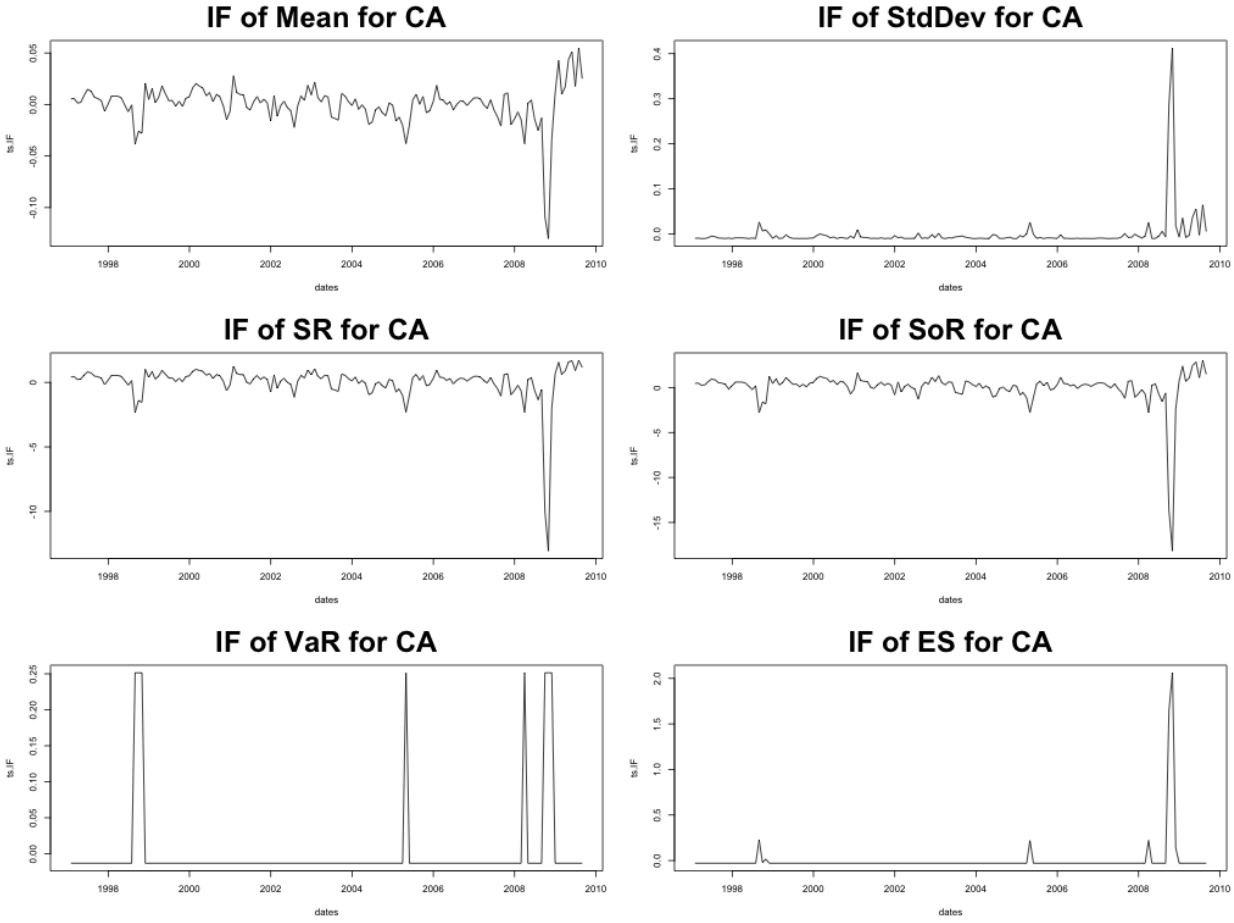

Figure 1.2: Influence Functions Time Series of CA Hedge Fund Returns

1.2.2 Two Important Properties of Influence Functions

The first property of an influence function $IF(r; T, F)$ for an estimator $T(F_n)$ obtained from a functional representation $T(F)$ is that its expectation is zero for returns with distribution

$$E_F\{IF(r; T, F)\} = 0. \tag{1.20}$$

This property was stated without proof in Hampel (1974), and further details are provided in Hampel et al. (1986) and Huber and Ronchetti (2009).

The second property is that for well behaved estimator functionals, the difference between the estimator $T(F_n)$ and its asymptotic value $T(F)$ can be expressed as the following linear combination of influence functions of the returns at each point of time

$$T(F_n) - T(F) = \frac{1}{n}\sum_{t=1}^{n} IF(r_t; T, F) + \text{remainder}, \tag{1.21}$$

where the remainder goes to zero as $n \to \infty$ in a probabilistic sense. The proof of this is more involved than the simple local linear approximation implied by the influence function definition (1.6). Details are provided in Filippova (1962).

Let $V_n(T(F_n))$ be the variance of the series on the right-hand side of the above equation:

$$V_n(T(F_n)) = var\left[\frac{1}{n}\sum_{t=1}^{n} IF(r_t; T, F)\right]. \tag{1.22}$$

The quantity $V_n(T(F_n))$ is a finite-sample approximation of the variance of $T(F_n)$, and the approximation error goes to zero as $n \to \infty$. Equivalently, $n \cdot V_n(F_n) \to V(T(F_n))$ as $n \to \infty$, where $V(T(F_n))$ is the asymptotic variance of $\sqrt{n} \cdot (T(F_n) - T(F))$.[1]

[1] Here "asymptotic variance" means the variance of a limiting, typically normal, distribution of $\sqrt{n} \cdot (T(F_n) - T(F))$.

1.2.3 Asymptotic Variance of Estimators via Influence Functions

Asymptotic Variance for Independent and Identically Distributed Returns

For independent and identically distributed (i.i.d.) returns, the series expansion (1.21) indicates that the asymptotic variance of risk and performance measure estimators can be obtained in terms of their influence functions with the formula

$$V(T(F_n)) = E\{IF^2(r_1; T, F)\} = \int IF^2(r; T, F)dF(r). \tag{1.23}$$

Direct proofs for the i.i.d. case can found in Hampel et al. (1986) and Huber et al. (2009).

Furthermore, in the i.i.d. case, explicit asymptotic variance formulas exist for most risk and performance measure estimators, including those in Section 2.1 that we study in this chapter. For example, see Martin and Zhang (2017) for the analytical formula for the asymptotic variance of expected shortfall estimator for the special cases of normal and t-distributed returns. Alternatively, a non-parametric empirical estimator of the asymptotic variance in the i.i.d. case is obtained by replacing F with the empirical distribution function F_n in (1.23), thereby obtaining the simple finite-sample variance estimator:

$$\hat{V}_n(T(F_n)) = \frac{1}{n}\sum_{t=1}^{n} IF^2(r_t; T, F_n)\,. \tag{1.24}$$

However, the above does not work when the returns are serially correlated.

Variance and Standard Error of Sample Mean for Correlated Returns

In preparation for using the series on the right-hand side of (1.21) to evaluate the finite sample variance of a risk or performance measure estimator for serially correlated returns series, it will be useful to first consider the variance and standard error of the sample mean when returns are serially

correlated. After all, ignoring the remainder term we have a sample mean, albeit of influence function transformed returns, whose standard error we want to estimate.

Consider the case of a sample mean $\hat{\mu}_n$ of a time series r_t, $t = 1, 2, \cdots, n$ of returns with a constant mean value $\mu = E(r_t)$, and covariances $\mathrm{Cov}(r_t, r_u)$ that depend only on $|t-u|$, i.e., $\mathrm{Cov}(r_t, r_u) = C(l)$, $|t-u| = l$. We have $C(0) = V(r_t) = \sigma_r^2$, and lag-$l$ correlations $cor(r_t, r_u) = \rho(l) = C(l)/C(0) = C(l)/\sigma_r^2$. The standard error (SE) of $\hat{\mu}_n$ for such a process is

$$\mathrm{SE}(\hat{\mu}_n) = \left[V\left(\frac{1}{n}\sum_{t=1}^{n} r_t\right)\right]^{1/2} = \frac{1}{n}\left[n \cdot C(0) + 2 \cdot \sum_{l=1}^{n-1}(n-l)C(l)\right]^{1/2} = \frac{\sigma_r}{\sqrt{n}}\left[1 + 2 \cdot \sum_{l=1}^{n-1}\left(1 - \frac{l}{n}\right)\rho(l)\right]^{1/2}. \tag{1.25}$$

The asymptotic version of the above standard error, corresponding to the asymptotic distribution of $\sqrt{n}\,(\hat{\mu}_n - \mu)$, obtained by multiply the right-hand sides above by $\sqrt{n}$, and letting $n \to \infty$, is :

$$\mathrm{SE}_\infty(\hat{\mu}_n) = \left[C(0) + 2 \cdot \sum_{l=1}^{\infty} C(l)\right]^{1/2} = \sigma_r \cdot \left[1 + 2 \cdot \sum_{l=1}^{\infty} \rho(l)\right]^{1/2}. \tag{1.26}$$

To illustrate the known fact that non-zero correlations for positive lag values can considerably inflate the SE of the sample mean, consider the case of Gaussian AR(1) $r_t = \phi r_{t-1} + \epsilon_t$ with transition parameter ϕ, error variance σ_ϵ^2, and returns variance $\sigma_r^2 = \sigma_\epsilon^2/\left(1-\phi^2\right)$. The lag-$l$ covariance for this model is $C(l) = \sigma_r^2 \phi^{|l|}$, and it is straightforward to derive the following expressions (details are provided in Appendix 1B):

$$\mathrm{SE}(\hat{\mu}_n) = \frac{\sigma_r}{\sqrt{n}}\left[1 + 2\frac{\phi - \phi^{n-1}}{1-\phi} - \frac{2}{n}\frac{\phi - \phi^n}{(1-\phi)^2}\right]^{1/2}, \tag{1.27}$$

$$\mathrm{SE}_\infty(\hat{\mu}_n) = \sqrt{\frac{1+\phi}{1-\phi}} \cdot \sigma_r, \quad \text{as } n \to \infty\,. \tag{1.28}$$

Table 1.1 shows the behavior of SE and the error rate (ER) of normal distribution 95% confidence intervals, based on the naive assumption of i.i.d returns, for several values of ϕ at sample size $n = 60$, e.g., as in five years of monthly returns.[2]

ϕ	SE	PI-SE	ER	PI-ER
0	0.129	0	0.05	0
0.1	0.142	10.4 (10.6)	0.076 (0.076)	51.5 (52.5)
0.2	0.158	22.0 (22.5)	0.108 (0.110)	117 (119)
0.3	0.175	35.5 (36.3)	0.148 (0.150)	196 (200)

Table 1.1: SE and ER Values Assuming i.i.d. Returns but are Actually Gaussian AR(1) for $n = 60$ and $n = \infty$ (shown in parentheses)

The second and third columns of the tables contain the values of SE and their percent increase (PI-SE) relative to the $\phi = 0$ case. The fourth and fifth columns contain the error rates (ER) and the percent increase of the error rates (PI-ER) relative to the $\phi = 0$ case. The numbers in parenthesis in the PI-SE, ER and PI-ER columns are the asymptotic values of these quantities.

Needless to say, SE and ER values are quite sensitive to small increases in ϕ, particularly so for the ER which increases by about 200% when ϕ increases from zero to just 0.3. This is particularly troublesome since a correlation of 0.3 is hardly noticeable in a lag-one scatter plot of returns. Clearly one needs to take into account small serial correlations when computing standard errors, and even more importantly when constructing confidence intervals or hypothesis testing.

Newey–West Type Standard Error Estimators

In practice one has to estimate the unknown σ^2 and $C(l)$ in order to use the finite-sample formula (1.25), and a naive approach would be to plug such estimates into that formula:

$$\mathrm{SE}(\hat{\mu}_n) = \frac{1}{\sqrt{n}}\left[\hat{\sigma}_r^2 + 2 \cdot \sum_{l=1}^{n-1}\left(1 - \frac{l}{n}\right)\hat{C}(l)\right]^{1/2}. \tag{1.29}$$

[2]The error rate $\alpha_{iidAssumed}$ based on the assumption of i.i.d. returns when in fact the returns are correlated is given by $\alpha_{iidAssumed} = 2 \cdot P(Z \leq (\sigma_{iid}/\sigma_{corr}) \cdot z_{\alpha/2}$, where σ_{iid} is the standard deviation when the returns are i.i.d., σ_{corr} is the standard deviation when the returns are correlated, and z_α is the α quantile of a standard normal distribution.

However, the above estimator is not usable in practice because for long-lags l there is not much data with which to compute the estimates $\hat{C}(l)$. For example e.g. at lag $l = n - 1$ there is only a single pair of returns $r_1$and r_n to use to estimate $C(n-1)$. It is known that the use of such naive estimators sometimes results in negative estimates for the standard error (correspondingly the covariance matrix determined by the covariance estimates at all lags fails to be positive definite).

Econometricians have developed methods of dealing with the above problem. In particular, Newey and West (1987) and Newey and West (1994), in a more general multivariate setting, proposed to use weights that go to zero smoothly at some well-chosen lag m. For our simple one-dimensional problem, a Newey–West type variance estimator, from which standard errors are obtained, has the form

$$V(\hat{\mu}_n) = \hat{C}(0) + 2 \cdot \sum_{l=1}^{m} w(l,m) \cdot \hat{C}(l), \tag{1.30}$$

where $w(j,m)$ is zero for $j > m$ and $m \ll n$. In their original work, Newey and West (1987) suggested that $w(l,m) = 1 - [l/(m+1)]$ and m as a function of the sample size n. Appendix 1C provides details concerning the growth rate of m as a function of sample size n.

Subsequent to Newey's original paper, much work has been done to study other forms of kernel functions $w(l,m)$ and the optimal choice for the value of the bandwidth m. See for example Andrews (1991), Andrews and Monahan (1992), Newey and West (1994), Hirukawa (2010). Nonetheless, the Newey–West form above is quite commonly used, and we do so in our subsequent comparisons of our new method with the Newey–West method.

Standard Errors of Risk and Performance Measure Estimator via Influence Functions

Since influence functions are just nonlinear transformations of returns, we assume that like the returns themselves, the covariances of the time series $IF(r_t)$ at times t and u depend only on $|t-u|$:

$$C_{IF}(\ell) = \mathrm{Cov}\left[IF(r_t), IF(r_u)\right], \; |t-u| = l\,. \tag{1.31}$$

This will hold for example if the returns are strictly stationary and the influence function transformed returns have finite variance.

Applying (1.25) to the right-hand side of (1.22) gives

$$V_n(T(F_n)) = \frac{1}{n}\left[C_{IF}(0) + 2\cdot\sum_{l=1}^{n-1}\left(1-\frac{l}{n}\right)C_{IF}(l)\right], \tag{1.32}$$

and in view of the first property of influence functions (1.20), we have:

$$V_n(T(F_n)) = \frac{1}{n}\left[E[IF^2(r_1)] + 2\cdot\sum_{l=1}^{n-1}\left(1-\frac{l}{n}\right)E\left[IF(r_1)\cdot IF(r_{1+l})\right]\right]. \tag{1.33}$$

The corresponding asymptotic variance $V(T(F_n))$ obtained from the above expression by dropping the factor $1/n$ and letting $n \to \infty$ is:

$$\begin{aligned} V(T(F_n)) &= C_{IF}(0) + 2\cdot\sum_{l=1}^{\infty} C_{IF}(l) \\ &= E[IF^2(r_1)] + 2\cdot\sum_{l=1}^{\infty} E\left[IF(r_1)\cdot IF(r_{1+l})\right]. \end{aligned} \tag{1.34}$$

A finite-sample standard error of a risk or performance measure estimator $T(F_n)$ can of course in principle be obtained by computing an estimate of $V_n(T(F_n))$ and taking its square root. The remainder of this chapter is devoted to a frequency domain approach to do so. But first we briefly note the intractability of obtaining analytic formulas for the terms in the series of the above formulas.

Except for the sample mean estimator, the influence functions in Section 2.1 are all highly nonlinear functions of returns, and this is indeed the case for most risk and performance estimators. Consequently, there is in general little hope for analytic evaluation of the terms in the finite-sample and asymptotic variance expressions (1.33) and (1.34) when there is serial correlation, even in the case of normally distributed returns. For example, consider the case of VaR, whose influence function nonlinearity (1.17) is the simplest of the nonlinear influence functions in Section 2.1. In this case one must resort to numerical integration methods to evaluate the expected value of the product of influence functions of the returns at lags $l \geq 1$. This is because for VaR we have

$$E\left[IF_{VaR}(r_1)IF_{VaR}(r_{1+l})\right] = \frac{E\left[\mathbb{1}(r_1 \leq q_\alpha(F)) \cdot \mathbb{1}(r_{1+l} \leq q_\alpha(F))\right] - \alpha^2}{f^2(q_\alpha(F))}\,. \tag{1.35}$$

For serially correlated normally distributed returns, the first term in the numerator above is the cumulative bivariate normal distribution evaluated at diagonal points $x = y = q_\alpha(F)$, for which a formula exists only for $q_\alpha(F) = 0$. See for example Genz (2004) and Meyer (2013) for numerical solutions for general points along the diagonal.

For non-normal serially correlated processes, one has the additional challenge of specifying a non-normal marginal and non-normal bivariate distributions. In summary, there is in general little hope for obtaining asymptotic variance formulas for risk and performance measures with serial correlation. One can of course adapt Newey–West type methods for estimating the finite-sample variance in (1.33). However, our focus in on a new alternative methods that we describe in the remainder of the paper, and which we show to be as good as, and often better than a Newey–West method for computing standard errors of risk and performance measure estimators in a manner that accounts for both serial correlation and non-normality of returns.

1.3 The Spectral Density at the Origin Approach

1.3.1 Spectral Densities and Estimator Variance with Serial Correlation

Here we assume that $\{y_t\}_{-\infty}^{\infty}$ is a wide sense stationary discrete time process with covariances $C_y(l) = \text{Cov}(y_t, y_{t+l})$ that by definition have the symmetry property $C_y(l) = C_y(-l)$, and furthermore the sequence $\left\{C_y(l)\right\}_{-\infty}^{\infty}$ is absolutely summable. Such a process has a spectral density representation

$$S_y(f) = \sum_{\ell=-\infty}^{\infty} C_y(\ell)\text{exp}(-i2\pi f l), \tag{1.36}$$

and correspondingly the covariance sequence is given by the Fourier coefficients:

$$C_y(\ell) = \int_{-1/2}^{1/2} S_y(f)\text{exp}(i2\pi f \ell)df\,. \tag{1.37}$$

Thus the spectral density $S_y(f)$ and the covariance function $C_y(\ell)$ are a Fourier transform pair (taking a Fourier series as a discrete Fourier transform). Furthermore, the symmetry of $C_y(l)$ implies that the spectral density $S_y(f)$ is symmetric.

It follows from (1.36) that the spectral density at $f = 0$ is simply the sum of the covariances:

$$\begin{aligned} S_y(0) &= \sum_{\ell=-\infty}^{\infty} C_y(\ell) \\ &= C_y(0) + 2\sum_{\ell=1}^{\infty} C_y(\ell)\,. \end{aligned} \tag{1.38}$$

It follows from (1.34) and the above formula with $y_t = IF(r_t; T(F_n))$, that the asymptotic variance $V(T(F_n))$ of an estimator $T(F_n)$ with influence function $IF = IF(r_t; T(F_n))$ is given by

$$V(T(F_n)) = S_{IF}(0). \tag{1.39}$$

Thus, one may estimate the asymptotic variance of an estimator $T(F_n)$ by estimating the spectral density of the process $IF(r_t; T(F_n))$ at frequency zero. With a good estimate $\hat{S}_{IF}(0)$ in hand, one computes an approximate finite-sample variance of $T(F_n)$ by dividing $\hat{S}_{IF}(0)$ by the sample size n. Taking the square root of that result to gives an approximate finite-sample standard error for $T(F_n)$.

1.3.2 Periodograms and their Properties

The periodogram of a discrete time series $y(t)$, $t = 1, 2, \cdots, n$ is defined in terms frequencies $f_k = k/n$ by

$$p_{y,k} = \frac{1}{n}\left|\sum_{t=1}^{n} y(t)e^{-2\pi i\cdot(t-1)\cdot f_k}\right|^2, \; k = 1, 2, \cdots, n/2, \tag{1.40}$$

where for notational convenience we will always choose n to be even, and $y(t)$ is typically a "de-meaned" series obtained by subtracting the sample mean from the original series. The quantity inside the absolute value signs above is known as the discrete Fourier transform (DFT), and f_k are called the DFT frequencies. DFT's are very efficiently computed using a fast Fourier transform (FFT).

Periodograms are known to have the following approximate finite-sample statistical properties, that become exact asymptotically in n:

- $E[p_{y,k}] \approx S_y(f_k), \; k = 1, 2, \cdots, n/2$
- $V[p_{y,k}] \approx S_y^2(f_k), \; k = 1, 2, \cdots, n/2$
- $\mathrm{Cov}[p_{y,k}, \; p_{y,l}] \approx 0, \; k, \; l = 1, 2, \cdots, n/2, \; k \neq l$
- $p_{y,k}$ is approximately exponentially distributed

See for example Walker (1965), Olshen (1967) and Brillinger (2001).

Spectral Density Estimation Methods

The rapid increase in the use of the FFT subsequent to the seminal paper by Cooley and Tukey (1965), spawned a very large literature and a number of books on the use of the periodogram to compute non-parametric spectral density estimates for a range of frequencies f . The two main approaches based on use of the FFT to compute DFT's are frequency domain periodogram averaging methods and time domain periodogram averaging methods. See for example (Priestley 1981; Percival and Walden 1993). However, for estimating the spectral density at zero frequency we use instead a periodogram-based maximum likelihood regression method.

1.4 Periodogram Based Generalized Linear Model Method

The modeling method we use was inspired by Heidelberger and Welch (1981) (HW) who wanted to compute variance estimates of sample means in the presence of serial correlation. The basic idea of their method was to compute the periodogram of the time series of their data, fit a polynomial in DFT frequencies to the periodogram, and use the intercept as an estimate of the spectral density at frequency zero. Although their method was shown to perform well in some examples, two problems with their method remain to be addressed. The first problem is how to properly deal with the exponential distributions of periodograms. The second issue is that the performance of a polynomial regression method depends on having a good model selection method, i.e., a good method for selecting the order and coefficients of the polynomial, and HW provided only an ad hoc method.

We overcome the two limitations of the HW method by using: (1) a periodogram-based generalized linear model (GLM) for exponential distributions (glmExp), and (2) an elastic net (EN) variant of glmExp regression that provides parsimonious polynomial model selection (glmExpEN). We refer to the overall procedure as the seCorIF method.

First, we briefly point out how one would use the HW method in the context of our approach based on influence functions. For a time series $IF_{T(F_n)}(r_t)$ of influence functions where the returns series has a smooth spectral density $S_r(f)$, it is reasonable to assume that the spectral density $S_{IF}(f)$ is a smooth function that can be well represented by a polynomial. This would lead to the use of a polynomial regression model

$$p_{IF,k} = \beta_0 + f_k\beta_1 + f_k^2\beta_2 + f_k^3\beta_3 + \cdots + f_k^d\beta_d + \epsilon_k,\ k = 1, 2, \cdots, n/2, \tag{1.41}$$

where $p_{IF,k}$ is the periodogram of $IF_{T(F_n)}(r_t)$, and

$$E(p_{IF,k}) = \beta_0 + f_k\beta_1 + f_k^2\beta_2 + f_k^3\beta_3 + \cdots + f_k^d\beta_d \approx S(f_k). \tag{1.42}$$

The problem with this model is that since the $p_{IR,k}$ are exponentially distributed, the errors ϵ_k are exponentially distributed, and least squares is not a maximum-likelihood estimator. Following the HW method of dealing with this problem, one would use a well-selected normalizing transformation $y_k = h(p_{IF,k})$ of the $p_{IF,k}$ to achieve a more symmetric normal-like distribution and use the y_k as the response variables. Although this can often work reasonably well for some problems, a better general solution may be obtained using a generalized linear model (GLM) for exponential distributions as the basic building block.

1.4.1 Generalized Linear Model for Periodogram

We begin by very briefly describing the key elements of a generalized linear model, and then specialize it to the relevant case of exponential distributions.

Generalized Linear Modeling Framework

A generalized linear model (GLM) method assumes independent observations $y_1, y_2, \cdots, y_K$, whose mean values are $\mu_1, \mu_2, \cdots, \mu_K$, and are primarily focused on the family of exponential distributions

$$f(y_k; \theta_k) = \exp\{(y_k\theta_k - b(\theta_k))/a(\phi_k) + c(y_k, \phi_k)\}, \quad k = 1, 2, \cdots, K. \tag{1.43}$$

The form stated above with y_k in the first term in the exponent rather than some function $a(y_k)$ is called the *canonical* form, and θ_k is called the *natural parameter*. [3] See for example McCullagh and Nelder (1989) and Fox (2016), where one finds simple derivations of the following general expressions for the mean and variance of the observations from an exponential family:

$$\mu_k = E(y_k) = b'(\theta_k), \tag{1.44}$$

$$V(y_k) = b''(\theta_k)a(\phi_k). \tag{1.45}$$

A key aspect of a GLM is the specification of a link function g that maps the mean values $\mu_1, \mu_2, \cdots, \mu_K$ into regression model linear forms $\eta_1, \eta_2, \cdots, \eta_K$:

$$\eta_k = \boldsymbol{x}_k'\boldsymbol{\beta} = g(\mu_k), \; k = 1, 2, . \cdots, K. \tag{1.46}$$

Note that for the usual case of an invertible link function g, an implicit connection between the linear model regression coefficient vector $\boldsymbol{\beta}$ and the exponential family vector parameter $\boldsymbol{\theta}$ is given by

$$g^{-1}(\boldsymbol{x}_k'\boldsymbol{\beta}) = \mu_k = b'(\theta_k), \; k = 1, 2, \cdots, K, \tag{1.47}$$

and when the inverse $(b')^{-1}$ exists we have

$$\theta_k = (b')^{-1}\left[g^{-1}(\boldsymbol{x}_k'\boldsymbol{\beta})\right], \; k = 1, 2, \cdots, K. \tag{1.48}$$

[3] Other authors use slightly different but equivalent mathematical forms of an exponential family, e.g., see Dobson and Barnett (2008)

To keep the dependence of $\boldsymbol{\theta}$ on $\boldsymbol{\beta}$ in mind we write $\boldsymbol{\theta} = \boldsymbol{\theta}(\boldsymbol{\beta})$.

The likelihood function for an exponential family has the following general form

$$L(\boldsymbol{\theta}) = L(\boldsymbol{\theta}(\boldsymbol{\beta})) = \prod_{k=1}^{K} \exp\left[(y_k\theta_k - b(\theta_k))/a(\phi_k) + c(y_k, \phi_k)\right] . \tag{1.49}$$

The Periodogram and Exponential Distributions

In our application we use the natural logarithm function as the link function g

$$\eta_k = g(\mu_k) = \log(\mu_k) = \boldsymbol{x}_k'\boldsymbol{\beta}, \tag{1.50}$$

with inverse

$$\mu_k = \exp\left(\boldsymbol{x}_k'\boldsymbol{\beta}\right), \tag{1.51}$$

and since we will be using a polynomial linear model, the independent variables $\boldsymbol{x}_k$ will be vectors of the powers of the frequency variable f_k:

$$\boldsymbol{x}_k' = \left(1, f_k, f_k^2, \cdots, f_k^d\right), \qquad k = 1, 2, \cdots, n/2, \tag{1.52}$$

with $x_{kj} = f_k^j,\ j = 0, 1, \cdots, d$.

Note that we now have $K = n/2$. In view of the approximate exponential distribution of periodograms, we focus on the exponential distributions

$$f(y_k; s_k) = \frac{1}{s_k} \cdot \exp\left(-\frac{y_k}{s_k}\right)\ y_k \geq 0,\ s_k > 0,\ k = 1, 2, \cdots, n/2 \tag{1.53}$$

obtained from exponential family by the choices $a(\phi) = 1$, $c(y_k, \phi_k) = 0$, and

$$\theta_k = -1/s_k, \tag{1.54}$$

$$b(\theta_k) = -\log(-\theta_k),\ \theta_k < 0. \tag{1.55}$$

It is well known that exponential distributions in the form (1.53) have mean s_k and variance s_k^2 (representing the periodogram approximate mean $S(f_k)$ and variance $S^2(f_k)$, respectively). But we need the mean and variance in terms of the exponential family parameterization as given by (1.44) and (1.45):

$$\mu_k = b'(\theta_k) = -\frac{1}{\theta_k},\ \theta_k < 0, \tag{1.56}$$

$$V\{y_k\} = b''(\theta_k) \cdot 1 = \frac{1}{\theta_k^2} = \mu_k^2\,. \tag{1.57}$$

Note that in this case the inverse of $b'$is given by

$$(b')^{-1}(t) = -\frac{1}{t},\ t > 0. \tag{1.58}$$

Exponential Distribution Log-Likelihood and its Gradient

Specializing the exponential distribution likelihood function (1.49) to the exponential distribution and taking its logarithm gives the log-likelihood

$$l(\boldsymbol{\theta}) = \sum_{k=1}^{n/2} \left[y_k \theta_k - b(\theta_k) \right], \tag{1.59}$$

where $\boldsymbol{\theta} = \boldsymbol{\theta}(\boldsymbol{\beta})$. From (1.48), using (1.50) and (1.58), we have

$$\theta_k = -\exp(-\boldsymbol{x}_k'\boldsymbol{\beta}) \tag{1.60}$$

and this, along with (1.55), shows that $b(\theta_k) = -\boldsymbol{x}_k'\boldsymbol{\beta}$. Thus we can write the log-likelihood as

$$l(\boldsymbol{\beta}) = \sum_{k=1}^{n/2} \left[-y_k \cdot \exp(-\boldsymbol{x}_k'\boldsymbol{\beta}) - \boldsymbol{x}_k'\boldsymbol{\beta}\right] . \tag{1.61}$$

Maximizing the above likelihood is the same as choosing $\boldsymbol{\beta}$ as

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \sum_{k=1}^{n/2} \left[y_k \cdot \exp(-\boldsymbol{x}_k'\boldsymbol{\beta}) + \boldsymbol{x}_k'\boldsymbol{\beta}\right], \tag{1.62}$$

which stands in interesting contrast with linear least squares regression that minimizes the sum of the squared residuals.

For optimization purposes we need the gradient $\boldsymbol{\nabla} l(\boldsymbol{y}, \boldsymbol{\theta}(\boldsymbol{\beta}))$ of the log-likelihood, the components of which are easily obtained in the following form from (1.62):

$$\left[\boldsymbol{\nabla} l(\boldsymbol{\theta}(\boldsymbol{\beta}))\right]_j = \sum_{k=1}^{n/2} \left[\frac{(y_k - \exp(\boldsymbol{x}_k'\boldsymbol{\beta}))}{\exp(\boldsymbol{x}_k'\boldsymbol{\beta})} \cdot x_{kj},\ j = 0, 1, 2, \cdots, d\right] . \tag{1.63}$$

An old classical method of computing the β maximum likelihood estimates is to use the gradient and the Hessian (known in statistics as "score" function and "observed information matrix") in an iterated weighted least squares (IWLS) algorithm. See for example McCullagh and Nelder (1989) and Fox (2016). However, it is not necessary to use a second-order optimization and we find that our first-order optimization methods described in Section 4.2 work well.

The Zero Frequency Spectral Density Estimator of $V_n(T(F_n))$.

Once we find $\hat{\boldsymbol{\beta}}$, the solution to the optimization problem (1.62), we can obtain $\widehat{V}_n(T(F_n))$, the estimator for the finite sample variance $V_n(T(F_n))$, as follows. Let y_k be the random variable for

the periodogram of the influence function series at the $k-$th frequency, and let μ_k be the expectation of y_k. From equation (1.51) we get the estimate $\hat{\mu}_k$ for μ_k as

$$\hat{\mu}_k = \exp\left(\boldsymbol{x}_k'\hat{\boldsymbol{\beta}}\right), \tag{1.64}$$

where $\boldsymbol{x}_k'$ is the vector of the powers of the frequencies defined in equation (1.52). Set $k = 0$, we have

$$\boldsymbol{x}_0' = (1,\, 0,\, 0, \cdots,\, 0)\,. \tag{1.65}$$

Therefore, the estimator of the spectral density $S_{IF}(0)$ at zero frequency in (1.39) is given by

$$\hat{S}_{IF}(0) = \hat{\mu}_0 = \exp\left(\boldsymbol{x}_0'\hat{\boldsymbol{\beta}}\right) = \exp(\hat{\beta}_0), \tag{1.66}$$

which is also the estimator $\widehat{V}_n(T(F_n))$ of the finite-sample variance $V_n(T(F_n))$:

$$\widehat{V}_n(T(F_n)) = \hat{S}_{IF}(0) = \exp(\hat{\beta}_0). \tag{1.67}$$

1.4.2 GLM with Elastic Net Regularization

One problem with the standard GLM methods is that they do not have a regularization method to avoid over-fitting. Furthermore, polynomial regression is subject to problems due to the collinearity of polynomials and a method is needed to take care of this. Regularization obtained by augmenting the original log-likelihood objective function with an appropriate penalty term can help with both problems.

Standard Linear Least Squares Regularization Methods

Here we summarize three standard regularization methods for linear least squares regression, representing the sum-of-squared residuals in terms of normal distribution log-likelihood functions. We do so to facilitate transition to our regularized GLM method.

The earliest regularization method for linear least squares was motivated by multicollinearity problems that commonly occur in applications, and are particularly severe in the case of polynomial regressions. A solution to this problem, called "ridge regression", was provided very early on by Hoerl and Kennard (1970). Ridge regression coefficients are defined by

$$\hat{\beta}_{Ridge} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}[-l(\beta)] + \lambda||\boldsymbol{\beta}||_2^2 \tag{1.68}$$

for some appropriately chosen penalty parameter λ, where the $||\boldsymbol{\beta}||_2^2 = \sum \beta_i^2$. See also Hastie et al. (2009).

A different kind of regularization, which has been popular for many years now, is referred to as a "least absolute shrinkage and selection operator" (LASSO), is due to Tibshirani (1996). The idea is similar to that of ridge regression, except that the motivation is to obtain a sparse model that avoids having many small regression coefficients. This is achieved by using ℓ_1 norm of the coefficients as a penalty term:

$$\hat{\beta}_{LASSO} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}[-l(\boldsymbol{\beta})] + \lambda||\boldsymbol{\beta}||_1, \tag{1.69}$$

where $||\beta||_1 = \sum |\beta_i|$. The sparsity of $\hat{\beta}_{LASSO}$ is controlled by adjusting the value of λ.

A further development in regularization comes in the form of the Elastic Net (EN), which combines ridge regression and LASSO regression by using a weighted sum of ℓ_1 norm and ℓ_2 norm coefficient penalties:

$$\hat{\beta}_{EN} = \underset{\beta}{\operatorname{argmin}}[-l(\boldsymbol{\beta})] + \lambda\left(\alpha\|\boldsymbol{\beta}\|_1 + \frac{1-\alpha}{2}\|\boldsymbol{\beta}\|_2^2\right). \tag{1.70}$$

The parameter $\alpha \in [0, 1]$ expresses one's belief about the relative importance of sparsity and ameliorating collinearity. The extent to which the mix of ℓ_1 and ℓ_2 penalties determine sparseness of coefficients and combat collinearity effects are controlled by adjusting the parameter λ.

The Elastic Net Regularized GLM for Exponential Distributions

We use a first-order optimization approach in creating an Elastic Net GLM method for exponential distributions. We do so by using (1.70) with $l(\boldsymbol{\beta})$ equal to the expression (1.61), and using a proximal gradient method. For details on proximal gradient methods, see for example Parikh et al. (2014). For details on our use of a proximal gradient method, see Chapter 2. We use glmExpEN as the name for our elastic net regularized GLM maximum-likelihood estimator method for exponential distributions.

Computational Details for glmExpEN

In general we compute λ using K-fold cross-validation, details of which are presented in Appendix 1D. For purposes of the example in Section 5 and the simulation studies in Section 6, we choose $\alpha = 1/2$ in order to equally balance the ℓ_1 and ℓ_2 penalty terms in (1.70), and we constrain the maximum order of the GLM polynomial to be 12.

A small but important computational detail is that the periodogram values used by the glmExpEN algorithm are sometimes very close to zero, and this can sometimes cause numerical problems for the algorithm.[4] To avoid such problems, we use a thresholding function $f(x) = \max\{x, \epsilon\}$, where $\epsilon = 0.001$ on the periodogram values.

[4]This is likely why the function `glm()` in `R` based on periodogram values as inputs occasionally fails.

Use of Prewhitening

Prewhitening is a widely used technique for spectral density function estimation in the field of signal processing and other applications areas in engineering and science as reflected, for example, in Percival and Walden (1993). It is also important to note that prewhitening of Newey–West (1987) type estimators was explored by Andrews and Monahan (1992) and by Hirukawa (2010), who showed that prewhitening can often improve the performance of those type of estimators. Since the core of our method is estimation of a spectral density at frequency zero as accurately as possible, we are interested to see if prewhitening can improve the accuracy of our basic seCorIF method.

In our context it will suffice to use the following model for prewhitening the IF_t time series :

$$IF_t^{pw} = IF_t - \hat{\rho} \cdot IF_{t-1}, \tag{1.71}$$

where $\hat{\rho}$ is a lag-one serial correlation coefficient estimate. In general IF_t^{pw} is not an uncorrelated ("white noise") series, but it has considerably less serial correlation than IF_t, and a periodogram estimator based on IF_t^{pw} will suffer from relatively little bias compared with one based on r_t. Standard linear filtering theory shows that the relationship between the spectral densities $S_{IF}(f)$ and $S_{IF}^{pw}(f)$ of IF_t and IF_t^{pw}, respectively is:

$$S_{IF}^{pw}(f) = \left|1 - \hat{\rho} e^{-i2\pi f}\right|^2 S_{IF}(f)\,. \tag{1.72}$$

Thus we use our glmExpEN method to compute an estimate $\hat{S}_{IF}^{pw}(0)$ of the spectral density $S_{IF}^{pw}(f)$ at $f = 0$ as in (1.39), and use it to compute the following estimate of $S_r(0)$:

$$\hat{S}_{IF}(0) = \frac{\hat{S}_r^{pw}(0)}{(1-\hat{\rho})^2} = \widehat{V}_n(T(F_n))\,. \tag{1.73}$$

We use the above prewhitening based glmExpEN method in our example in the next section, and in the simulation study of Section 6, where we compare both our standard method (1.39) and the prewhitening method (1.73).

1.5 Two R Packages and an Example

The optimization necessary to implement the glmExpEN fitting method has been developed using C++ for efficiency. The C++ code is interfaced with R using `Rcpp` and `RcppEigen`. The resulting R package is called `glmnetRcpp`, which is available at https://github.com/chenx26/glmnetRcpp. The overall computational procedures for computing risk and performance measures standard errors, based on use of `glmnetRcpp`, are implemented in the R package called `EstimatorStandardError`, which is available at https://github.com/chenx26/EstimatorStandardError. See Appendix 1E for instructions on installing the packages and code for computing the results for Table 3 discussed.

`EstimatorStandardError` currently supports computing standard errors for the following risk and performance measure estimators: mean, standard deviation, value-at-risk, expected shortfall, Sharpe ratio and Sortino ratio. We illustrate using the package to compute standard errors for those estimators, using the monthly returns of thirteen hedge funds from 1997/01/30 to 2009/08/31 that are contained in the `edhec` data set in the `PerformanceAnalytics` package. For each of the thirteen hedge funds, we compute standard errors using our new influence function based estimator for serially correlated returns (seCorIF), along with the estimator (1.24) based on the assumption of i.i.d. returns (seIidIF). In addition we provide bootstrap standard errors based on both the i.i.d. assumption (BOOTiid) and the assumption of serial correlation (BOOTcor). [5] The results are shown in Table 1.2, where the PI.seCorIF column gives the percent increase of seCorIF values relative to seIidIF values.[6]

[5]The BOOTiid() function uses the function boot::boot() and BOOTcor uses the function boot::tsboot() with fixed segment length of n/5.

[6]The code to produce this table is provided in Appendix 1E.

	ES	seIidIF	seCorIF	PI.seCorIF	BOOTiid	BOOTcor	ar1	IF.ar1
CA	-0.049	0.018	0.032	82	0.016	0.020	0.60	0.51
CTAG	-0.045	0.004	0.004	3	0.004	0.004	0.00	0.00
DIS	-0.043	0.013	0.018	43	0.011	0.013	0.53	0.36
EM	-0.089	0.024	0.028	15	0.022	0.023	0.34	0.12
EMN	-0.019	0.008	0.008	6	0.007	0.009	0.28	0.00
ED	-0.044	0.011	0.015	28	0.011	0.009	0.41	0.23
FIA	-0.042	0.016	0.025	60	0.013	0.016	0.50	0.46
GM	-0.023	0.003	0.003	-2	0.003	0.003	0.00	0.00
L/S	-0.044	0.009	0.012	35	0.008	0.009	0.29	0.33
MA	-0.025	0.006	0.007	10	0.005	0.004	0.32	0.00
RV	-0.031	0.010	0.014	52	0.008	0.010	0.48	0.49
SS	-0.111	0.013	0.015	13	0.012	0.018	0.15	0.12
FoF	-0.039	0.009	0.012	32	0.009	0.009	0.35	0.30

Table 1.2: New IF Method ES Standard Errors for Hedge Fund Returns

It is evident that the seCorIF values are almost always as large or larger than the seIidIF values, occasionally larger by more or less small amounts, but larger by at least 40% in the case of the four hedge funds CA, DIS, FIA and RV. The larger values of seCorIF, especially for those four hedge funds, reflects the substantial increase in the SE values due to the presence of serial correlation, over and above SE values based on an i.i.d. assumption. It is not surprising that those four largest percent increases in SE values are consistent with the fact, shown in the ar1 and IF.ar1 columns, that those four hedge funds have the four largest estimated AR1 correlation for the returns and for the influence function of the returns. It is interesting to note that the values in the latter column are typically smaller than those in the former column.

Note also that the bootstrap standard errors based on an i.i.d. assumption in the BOOTiid column are rather consistent with the standard errors using estimates of the asymptotic variance formulas based on influence functions in the SE-seIidIF column. On the other hand, the standard error estimates obtained by a bootstrap for serially correlated data in the BOOTcor column are usually smaller than those in the seCorIF column, and sometimes considerably so. This reflects the inability of BOOTcor to adequately capture the influence of serial correlation on standard errors.

Table 1.3 shows the fitted glmExpEN coefficients for each of the hedge fund returns time series. In

order make the table fit the page width, we are only showing the first and last four fitted coefficients. The point is to illustrate that: (1) The glmExpEN algorithm is actually performing model selection by setting some of the coefficients to zero, and (2) By applying the formula

$$\mathrm{SE}_n(T(F_n)) = \sqrt{\exp(\hat{\beta}_0)/n} \tag{1.74}$$

to the first column (beta0) in Table 1.3, one obtains the sample standard errors in the third column (seCorIF) of Table 1.2.[7]

	beta0	beta1	beta2	beta3	beta9	beta10	beta11	beta12	zero.coeffs
CA	-1.843	-5.344	-3.021	-1.460	-0.013	-0.006	-0.002	-0.001	1
CTAG	-5.935	-0.234	0.109	0.081	0.001	0.000	0.000	0.000	4
DIS	-3.016	-2.927	-1.229	-0.472	-0.001	0.000	0.000	0.000	4
EM	-2.147	-1.004	-0.477	-0.244	-0.003	-0.001	0.000	0.000	2
EMN	-4.641	-0.656	0.406	0.362	0.006	0.002	0.001	0.000	1
ED	-3.433	-1.837	-0.743	-0.281	0.000	0.000	0.000	0.000	4
FIA	-2.361	-4.064	-1.984	-0.861	-0.005	-0.002	0.000	0.000	2
GM	-6.520	-0.211	0.765	0.584	0.009	0.004	0.002	0.000	1
L/S	-3.856	-2.425	-1.129	-0.494	-0.003	-0.001	0.000	0.000	2
MA	-4.980	-0.679	-0.143	-0.030	0.000	0.000	0.000	0.000	5
RV	-3.444	-3.873	-1.886	-0.838	-0.006	-0.002	-0.001	0.000	2
SS	-3.438	-1.157	0.042	0.150	0.004	0.002	0.001	0.000	2
FoF	-3.827	-2.208	-0.945	-0.396	-0.002	-0.001	0.000	0.000	3

Table 1.3: Fitted glmExpEN Coefficients for Hedge Fund Returns

1.6 Performance Simulation Studies

This section reports results of simulation studies for AR(1) models of returns that demonstrate the performance of our new methods for computing standard errors of risk and performance measure estimators for sample sizes $n = 60$ and $n = 100$. Results for $n = 200$ are provided in Appendix 1F

[7]The coefficients set to zero are the ones with the value 0.0000. For rows in the table that do not show in columns beta9 through beta12 the number of zero values indicated in the zero.coeffs column, a zero coefficient value occurs in one of the beta4 through beta8 columns.

for the interested reader. Those sample sizes correspond to commonly encountered portfolio management contexts based on five years of monthly returns, and approximately two years of weekly returns. The following six estimators are studied: sample mean (Mean), sample standard deviation (StdDev), Sharpe ratio (SR), Sortino ratio (SoR), 10% tail probability value-at-risk (VaR) and 10% tail probability expected shortfall (ES). The rest of the section is organized into three subsections. The first subsection reports root mean-squared error (RMSE) performance results for our new method without pre-whitening (seCorIF), our new method with prewhitening (seCorIFPW), and the Heidelberger et al. (1981) method (HW). The second subsection reports the RMSE performance results of the seCorIF, seCorIFPW and Newey–West method with prewhitening (NWPW) relative to the standard Newey–West method (NW). The third subsection discusses the choice of SE estimator type. All simulations for seCorIF and seCorIFPW use 5-fold cross-validation.

1.6.1 Root-Mean-Squared Error Performance

This section shows root-mean-squared-error performance results for the seCorIF, seCorIFPW and HW methods for stationary AR(1) time series models

$$r_t = \phi r_{t-1} + \epsilon_t,\ t = 1, 2, \cdots, n \tag{1.75}$$

for $n = 60,\ 100$, with i.i.d. normal ϵ_t and ϕ ranging from 0.0 to 0.9 in steps of 0.1.

Using the notation in section 1.3.2, the HW method used in the simulation is implemented as follows:

1. Compute the periodogram $p_{y,k},\ k = 1, 2, \cdots, n/2$.
2. Compute the log of the average for adjacent periodogram $J(f_k) = \log\left(\left(p_{y,(2k-1)} + p_{y,2k}\right)/2\right)$, where $f_k = (4k-1)/2n$.

3. Fit a polynomial $g(f_k) = \sum_{i=0}^{d} \beta_i f_k^i$ to $J(f_k)$+0.27 for $k = 1, \ldots, K$. This is because $\log\left(p_{y,k}\right) = \mathrm{E}\left[J\left(f_k\right)\right] + 0.27$.
4. With $\hat{\beta}_0$ be the resulting estimate for β_0, the HW estimate of the spectral density at zero frequency is $\hat{S}_{IF}(0) = \hat{p}_{y,0} = C \cdot e^{\hat{\beta}_0}$, where C is a tuning constant determined by the choice of K and d.

In our simulation study, we choose $K = 25$, $d = 2$ and $C = 0.882$ for the HW method, which were used in the numerical experiments of Heidelberger and Welch (1981).These time series are scaled just as we did for the influence function plots in Figure 1.1 so that their marginal distributions are $N\left(0.01,\ 0.07^2\right)$. For each time series model, the true standard error of each risk and performance estimator is approximated by the following method:

- Generate M_0 =10,000 independent samples, each consisting of a model generated time series of length n.
- For each sample time series sample of length n, compute the risk and performance estimators. This results in 10,000 estimator values for each type of estimator.
- Compute the standard deviation of the 10,000 estimator values for each of the estimators. This is an approximation of true standard error of each of the risk and performance estimators, denoted as $\mathrm{SE}_{\mathrm{True}}$.

For each estimator and each of the estimator standard error methods, we compute the absolute performance metrics as follows:

- Generate M =10,000 sample time series of length n.
- For the m-th such Monte Carlo sample from one of the models, for one of the estimators, compute a standard error estimate $\mathrm{SE}_{METHOD,m}$ using one of the three standard error methods. This results in 5,000 standard error estimates $\mathrm{SE}_{METHOD,m}$, $m = 1, 2, \cdots, M$, for each of the three methods.

- For each estimator:
  - Compute an approximation SE_{METHOD} for the expected value of each standard error estimator

$$\mathrm{SE}_{METHOD} = \frac{1}{M}\sum_{m=1}^{M} \mathrm{SE}_{METHOD,m}. \tag{1.76}$$

  - Compute an approximation for the square root of mean squared error (RMSE):

$$\mathrm{RMSE}_{METHOD} = \sqrt{\frac{1}{\mathrm{M}}\sum_{m=1}^{\mathrm{M}} (\mathrm{SE}_{METHOD,m} - \mathrm{SE}_{\mathrm{True}})^2}. \tag{1.77}$$

Figures 1.3 and 1.4 show the RMSE's of the three standard error estimator methods for sample sizes 60 and 100, respectively. The RMSE's of all estimators increase with ϕ monotonically and as one expects the RMSE's of all estimators decrease with increasing sample size.

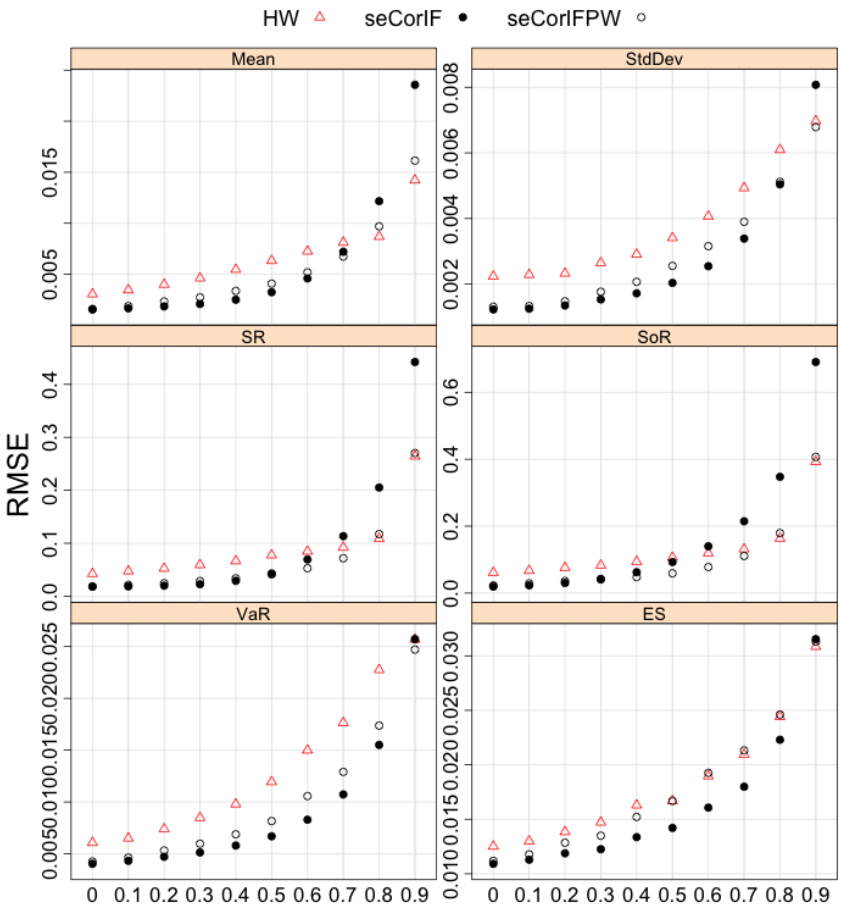

Figure 1.3: RMSE Comparison of New Standard Error Estimators for AR(1) Returns, $n = 60$

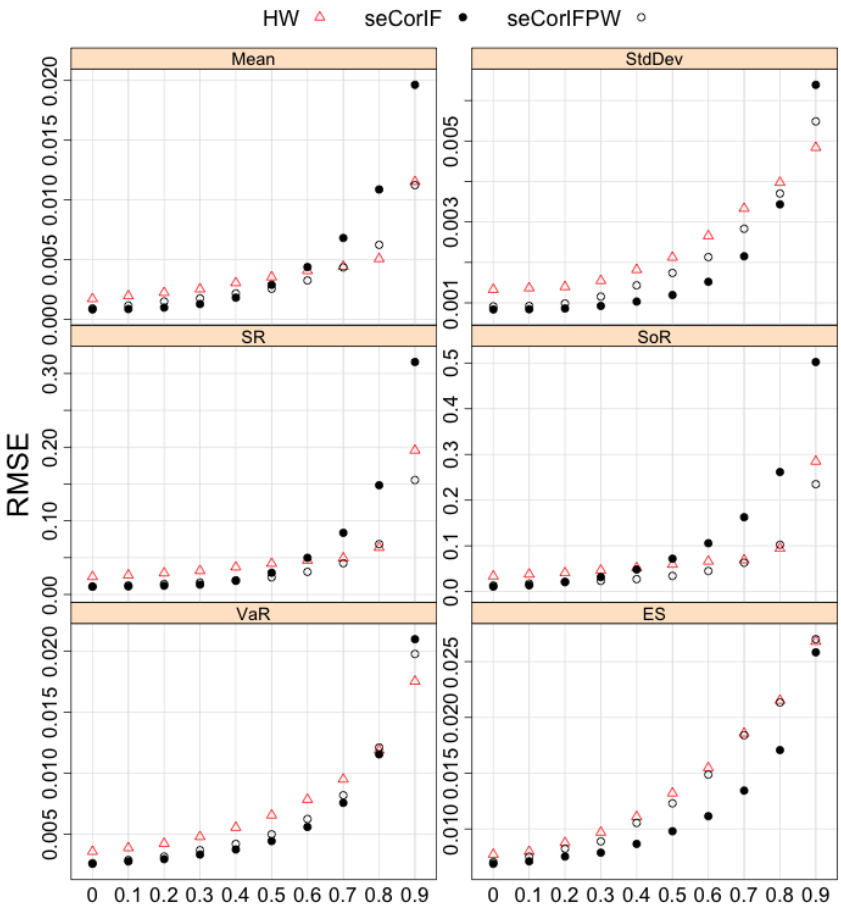

Figure 1.4: RMSE Comparison of New Standard Error Estimators for AR(1) Returns, $n = 100$

RSME Performance of the seCorIF and seCorIFPW Methods Relative to the HW Method

Figures 1.3 and 1.4 show that for sample sizes 60 and 100, except for a few cases where $\phi = 0.8$ or $\phi = 0.9$, one or both of the seCorIF and seCorIFPW methods exhibit performance in terms standard error estimator RMSE that is as good or better, and often considerably better, than that of the HW method.

Comparative RSME Performance of the seCorIF and seCorIFPW Methods

Because the differences between the RMSE's of the seCorIF and seCorIFPW methods are quite small on the vertical axis scales of the figures 1.3 and 1.4, it is difficult to tell in the above figures how much better or worse the seCorIF method is than the seCorIFPW method. Thus Figures 1.5 and 1.6 below for $n = 60$ and $n = 100$ show the percent increase of the seCorIFPW RMSE's, as well as that of the HW method, over that of the seCorIF RMSE.

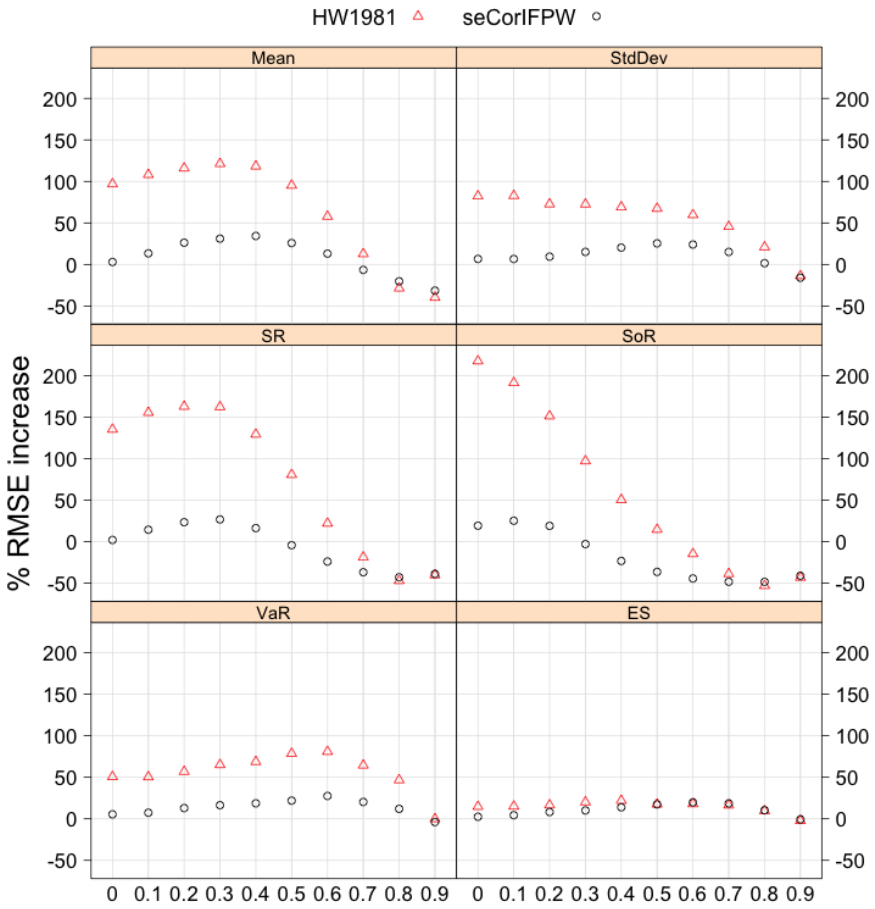

Figure 1.5: Percent RMSE Increase of seCorIFPW and HW Methods over seCorIF Method for AR(1) Returns, $n = 60$

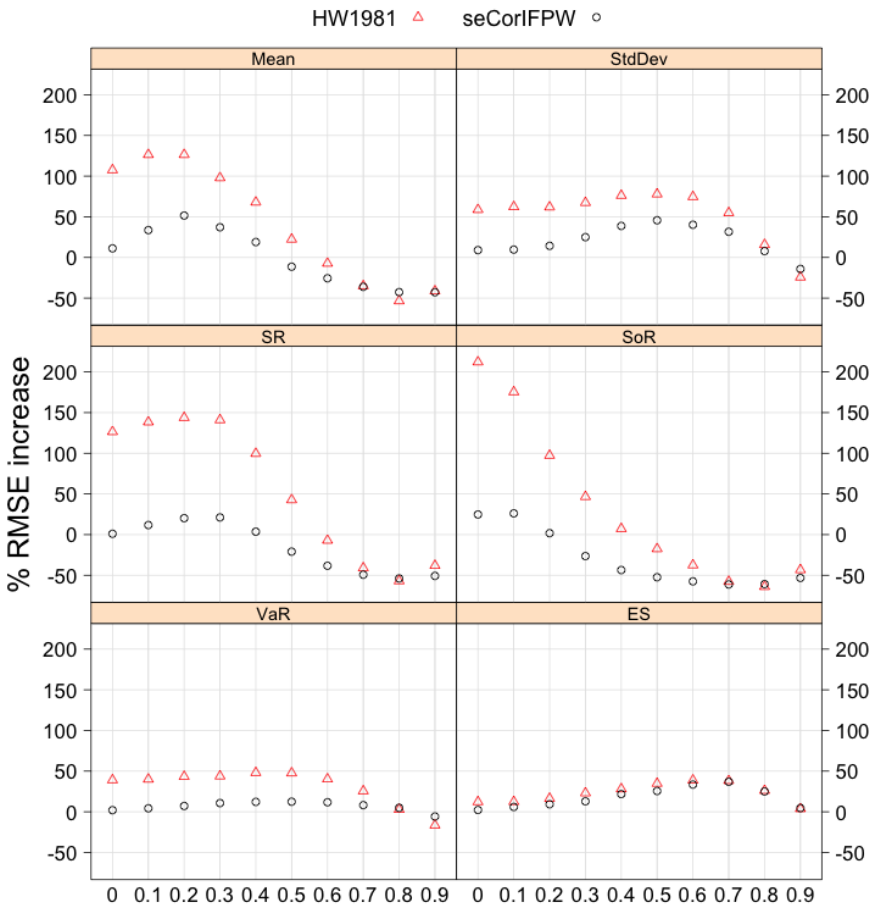

Figure 1.6: Percent RMSE Increase of HW and seCorIFPW over seCorIF for AR(1) Returns, $n = 100$

Figures 1.5 and 1.6 reveal immediately and more dramatically than in the previous figures that, except for the ES estimator and except for large values of ϕ for the other estimators, the HW method has worse performance than that of both the seCorIF and seCorIFPW methods, and considerably worse for smaller values of ϕ. Thus, we no longer consider the HW method and concentrate in the remainder of our discussion on the relative performance of the seCorIF and seCorIFPW methods.[8]

The general behavior of the percentage RMSE increases of the seCorIFPW method over that of the seCorIF method is that they are close to zero for $\phi = 0$, then rise gradually for a while as ϕ increases, and then decrease to negative values for all sufficiently large ϕ, except for VaR and

[8]It is interesting to note that the HW method converges in performance to that of the seCorIFPW method as ϕ tends toward $\phi = 0.9$. However, such large values of ϕ are rarely encountered in asset returns.

ES for which the percentage RMSE increases go to values close to zero as $\phi \to 0.9$. As for the choice between these two methods, one would use the seCorIFPW method rather than the seCorIF method for those ϕ values such that the percent RMSE increase of the seCorIFPW is negative, and vice-versa. But one does not know the value of ϕ in advance and would have to use a returns based test of whether or not the lag-one correlation coefficient is larger or smaller than the cross-over value of ϕ in order to take advantage of this behavior of the percent increase of RMSE.

Bias and Standard Deviation Performance of Standard Error Estimator Methods

Since mean-squared error has both variance and squared bias components, one may wonder how these components trade off against one another in the above RMSE plots. With that in mind, Appendix 1G provide plots similar to those above for the biases of the three methods for sample sizes $n = 60, 100, 200$, and likewise for their standard deviations.

1.6.2 Simulation Performance Results Relative to the Newey–West Method

The well-established and highly popular Newey–West method for dealing with serial correlation in the residuals of econometric time series models can also be applied to the time series of influence-function transformed returns in order to estimate the standard errors of risk and performance measures. Since it is important to know whether or not our new methods are more or less accurate than such a Newey–West method, we carried out simulation studies to compare the performance of our new seCorIF and seCorIFPW methods, as well as the performance of a Newey–West method with prewhitening (NWPW), to the performance of a commonly used basic Newey–West method with an automatic lag selector (NW). In the MC simulation, the NW and NWPW estimators are computed using the long-run variance function lrvar() in the sandwich package in R, with the prewhitening set to no prewhitening and AR Order 1 prewhitening, respectively. We define the relative root mean-squared error RRMSE_{METHOD} of each of three methods (seCorIF, seCorIFPW, NWPW)

relative to the root mean-squared error RMSE_{NW} of the basic Newey–West method without pre-whitening, as:

$$\text{RRMSE}_{METHOD} = \left(\frac{\text{RMSE}_{METHOD}}{\text{RMSE}_{NW}} \right) \times 100\,. \tag{1.78}$$

The results are shown in Figures 1.7 and 1.8 for $n = 60$ and $n = 100$, respectively. The first thing one notices in these two figures is that for the Mean, StdDev, SR and SoR estimators, the seCorIFPW and NWPW methods converge in performance as the values of ϕ increase beyond about 0.5.or 0.6.

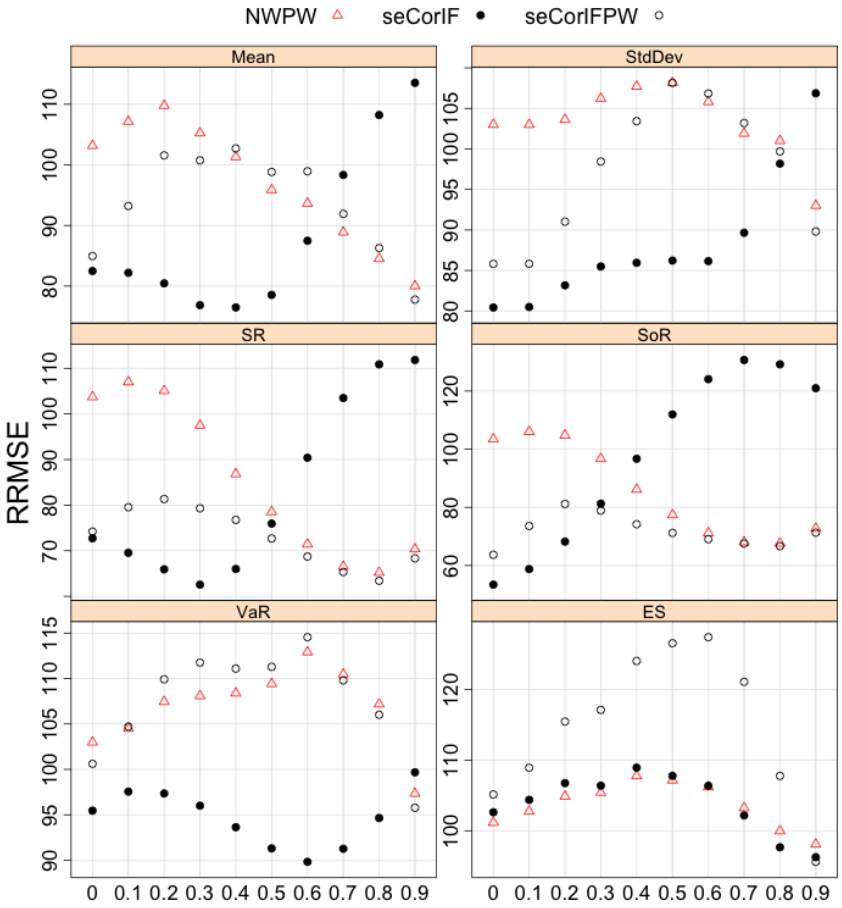

Figure 1.7: RRMSE Comparison of New Standard Error Estimators for AR(1) Returns, $n = 60$

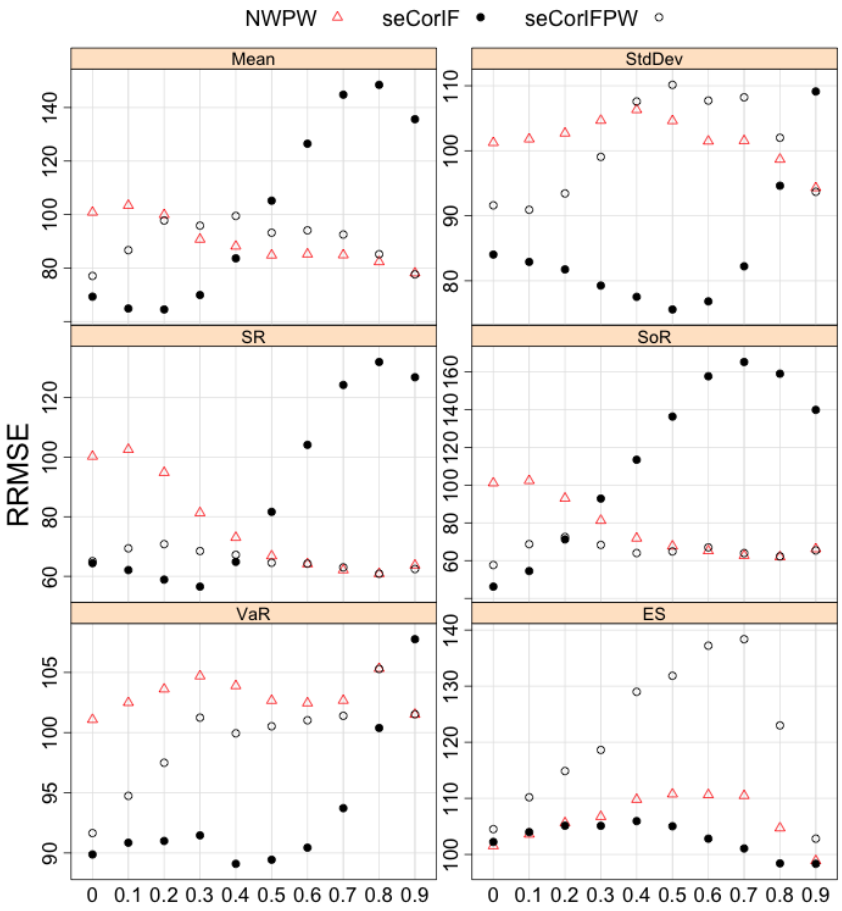

Figure 1.8: RRMSE Comparison of New Standard Error Estimators for AR(1) Returns, $n = 100$

The second thing that one notices in these figures is that the pattern of the plots for the seCorIF, seCorIFPW and NWPW methods can be classified into two groups. The first group includes the three risk estimators (StdDev, VaR, ES), and the second group consists of the three performance estimators (Mean, SR, SoR). The patterns within each group are very similar, but differ across these two groups. Focusing on the StdDev, VaR and ES estimators, it is immediately evident that, except for StdDev and VaR at $\phi = 0.9$ the seCorIF method is uniformly the best method. The relative performance of the seCorIF, seCorIFPW and NWPW methods is more complicated for the performance estimators, with a dependence on both performance estimator and sample size. The results in Figure 1.7 for $n = 60$ show that: (a) for the Mean estimator the seCorIF method performs best among all four methods for $\phi \leq 0.6$; (b) for the SR estimator the seCorIF method is better

than both the NW and NWPW methods for $\phi \leq 0.5$; (c) for the SoR estimator seCorIF is better than both the NW and NWPW methods for $\phi \leq 0.3$. But for those three performance estimators for all sufficiently larger ϕ values than those, the seCorIFPW method performs best among all four methods, and as mentioned above the seCorIFPW method coincides with the NWPW method for all sufficiently large values of ϕ.[9]

Most importantly, it should be noted that for the widely used SR and SoR estimators the seCorIFPW method performs considerably better than the NWPW method for small values of ϕ, while having performance essentially the same as that of the NWPW method for the largest values of ϕ. In the case of the Mean estimator, the seCorIFPW method has somewhat better or only slightly worse performance than the NWPW method across the full range of ϕ, while at the same time being much better than the NW method for small and large values of ϕ, and about the same performance as NW for middling values of ϕ.

It should also be noted that, in the case of the three performance estimators, the NWPW performs a little worse (about the same) as NW for $n = 60$ ($n = 100$), and $\phi \leq 0.3$ ($\phi \leq 0.2$), but that for larger values of ϕ the NWPW method performs increasingly better than NW.

Bias and Standard Deviation Relative Performance of Standard Error Estimator Methods

Bias and standard deviation relative performance results are presented in Appendix 1H.

1.6.3 SE Estimator Choice Recommendations

Using the results of Figures 1.3 and 1.4 as the primary guidelines for choosing one of our two proposed estimator methods (seCorIF or seCorIFPW), and without any assumptions concerning values of ϕ encountered in practice, we make the following recommendation:

[9] It appears that the poor performance of seCorIF for larger values of ϕ for the SR and SoR inherits this from that of the Mean SE estimator.

- For StdDev, VaR and ES estimators, use the seCorIF method. The reason for this is that the seCorIF method has a smaller standard error RMSE than the seCorIFPW method for all ϕ except $\phi = 0.9$ at both sample sizes 60 and 100, and at $\phi = 0.9$ the seCorIF standard error RMSE is only very slightly greater than that of the seCorIFPW method.

With the realistic assumption that asset returns seldom have values of ϕ larger than 0.5, we make the following recommendations.[10]

- For the Mean and SR estimators use the seCorIF method. The reason for this is that for these performance estimators for $\phi \leq 0.5$, the seCorIF standard error RMSE is less than that of the seCorIFPW method, except for being only slightly larger at $\phi = 0.5$ for $n = 100$.
- For the SoR estimator, use the seCorIFPW. This seems to be a reasonable compromise across the range of ϕ values from 0.0 to 0.5, since standard error RMSE of the seCorIFPW is only slightly larger than that of the seCorIF method for the smallest values of ϕ, but is equal to or smaller for values of ϕ greater than 0.3 (0.2) for $n = 60$ ($n = 100$).

NOTE: For applications where the portfolio manager believes that values of ϕ as large as 0.7 or 0.8 can occur, the best choice on balance is to use the seCorIFPW estimator.

The above recommendations for Mean, SR and SoR estimators leave open the question of whether one can do better by estimating ϕ and choosing either the seCorIF or seCorIFPW methods depending on whether the estimate $\hat{\phi}$ is smaller or larger than a threshold determined separately for each risk and performance estimator at a given sample size n.

1.7 Summary and Discussion

We have extended the Heidelberger and Welch (1981) frequency domain method of computing the standard error of a time series sample mean with serially correlated data to a general method

[10] Table 1.2 provides some typical estimate values for ϕ.

for estimating the standard errors of asset returns risk and performance estimators. The resulting extension, which we call the seCorIF method, along with its prewhitened variant seCorIFPW, consists of two key ingredients. The first is to use a representation of risk and performance estimators in terms of a sum of their influence function transformed returns. The second ingredient is to use a generalized linear polynomial model for the spectral density of the influence function transformed returns time series, with exponentially distributed periodogram values as the observables, and fit the model using an elastic net extension of the GLM maximum-likelihood estimator.

The results presented in Section 6 show that our new method substantially out-performs the original HW method. The results also show that one or both of out seCorIF and seCorIFPW methods outperform both a basic Newey–West methods (NW) and a prewhitening versions thereof (NWPW). As a further step in our research we will see if we can uniformly improve performance across values of the AR(1) parameter ϕ by first estimating this parameter, and then choosing seCorIF or seCorIFPW according to the value of the estimate $\hat{\phi}$. We note that although the simulation results presented in Section 6 only considered AR(1) returns models, we did briefly study the performance for MA(1) models and found that the seCorIF method performs uniformly better than NW, NWPW and seCorIFPW across all risk/performance measures and moving average parameters ranging from 0 to 0.9.

We believe the reason that the seCorIF and seCorIFPW methods deliver better performance than the Newey–West methods is because the latter methods are totally non-parametric, whereas our new methods are semi-parametric and make use of model selection with the elastic net modified GLM maximum-likelihood method for exponentially distributed periodogram values.

As with any new estimator, we need to study the method further in a number of ways. First of all, with regard to the elastic net model selection, we have just used a fixed equally weighted mix between the ℓ_1 and ℓ_2 norm. We need to study whether or not some form of cross-validation to choose a better mix will help improve the performance of the method. On the other hand, it might be that we could replace the elastic-net with AIC or BIC based model selection using polynomials

of relatively small order such as 5 or 6. We note that the simple polynomial approach poses a collinearity problem for which one cannot drop the ℓ_2 penalty, but one might be able to drop the ℓ_2 penalty by using alternative orthogonal basis functions.

Another question, noting that the periodogram values are only asymptotically exactly exponentially distributed, is whether or not an extension of our GLM method to work for the general family of gamma distributions can improve performance for finite samples. We are currently working on an such an extension.

Returns time series frequently contain outliers associated with either market events or asset specific behavior. For example the outlier during 2008 in the time series plots of influence function transformed returns in Figure 1.2 is associated with the financial crisis of 2008. Such outliers can have an adverse influence on risk and performance estimators, and one should examine the results obtained with a good robust method of reducing the influence of outliers, as a complement to analyses such as the one in this chapter. Robust pre-processing methods that can readily be used in the context of the methodology of the present paper include the multivariate robust shrinkage type outlier cleaning of Boudt et al. (2008), and univariate robust filtering method of Kleiner et al. (1979). Further details on robust time series filtering methods may be found in Maronna et al. (2018).

Finally, there is a need to study the efficacy of parametric models of the spectral density at zero frequency, the leading contender being higher order autoregressions, as an alternative to our semi-parametric method. Our initial research on this approach have to date been surprisingly disappointing. But in view of the simplicity of such an approach, further research seems warranted.

Appendices

1A Sharpe Ratio Influence Function Derivation

The derivation of the influence function for SR is as follows:

$$\begin{aligned}
IF(r;\mathrm{SR}) =& \frac{d}{d\gamma}\frac{\mu(F_\gamma)}{\sigma(F_\gamma)}\Big|_{\gamma=0} \\
=& \frac{1}{\sigma}IF(x;\mu,F) - \frac{\mu}{\sigma^2}IF(x;\sigma,F) \\
=& \frac{1}{\sigma}\int_{-\infty}^{\infty} sd[\delta_x - F](s) - \frac{\mu}{2\sigma^3}IF(x;\sigma^2,T) \\
=& \frac{1}{\sigma}(x-\mu) - \frac{\mu}{2\sigma^3}\int_{-\infty}^{\infty} 2(s-\mu)IF(x;\mu,T)dF(s) \\
& - \frac{\mu}{2\sigma^3}\int_{-\infty}^{\infty}(s-\mu)^2 d[\delta_x - F](s) \\
=& \frac{1}{\sigma}(x-\mu) - \frac{\mu}{2\sigma^3}\left[(x-\mu)^2 - \sigma^2\right] \\
=& -\frac{\mathrm{SR}}{2\sigma^2}(r-\mu)^2 + \frac{1}{\sigma}(r-\mu) + \frac{\mathrm{SR}}{2}.
\end{aligned} \tag{1.79}$$

1B Standard Error of Sample Mean for AR(1) Returns

For AR(1) Returns

$$r_t = \phi r_{t-1} + \epsilon_t,\ t = 1, 2, \cdots, n \tag{1.80}$$

with i.i.d. normal $\epsilon_t \sim N(0, \sigma_\epsilon^2)$, the stationary distribution is also a normal distribution with zero mean and standard deviation

$$\sigma_r^2 = \frac{\sigma_\epsilon^2}{1-\phi^2}. \tag{1.81}$$

The derivation of the finite-sample standard error of sample mean is as follows:

$$\begin{aligned} V(\hat{\mu}_n) &= \frac{1}{n^2}\sum_{t=1}^{n}\sum_{s=1}^{n} Cov\{r_t, r_s\} \\ &= \frac{\sigma_r^2}{n^2}[n + 2\sum_{t=1}^{n-1}(n-t)\phi^t] \\ &= \frac{\sigma_r^2}{n}\left[1 + 2\frac{\phi - \phi^{n-1}}{1-\phi} - \frac{2}{n}\phi\left(\frac{1-\phi^{n-1}}{1-\phi}\right)\right] \\ &= \frac{\sigma_r^2}{n}\left[1 + 2\frac{\phi - \phi^{n-1}}{1-\phi} - \frac{2}{n}\frac{\phi - \phi^n}{(1-\phi)^2}\right]. \end{aligned} \tag{1.82}$$

So the standard error is

$$\begin{aligned} \mathrm{SE}_{cor,n} &= \mathrm{SE}_{iid,n} \cdot \left(1 + 2\frac{\phi - \phi^{n-1}}{1-\phi} - 2\frac{\phi - \phi^n}{n\cdot(1-\phi)^2}\right)^{1/2} \\ &\to \mathrm{SE}_{iid,n} \cdot \left(\frac{1+\phi}{1-\phi}\right)^{1/2} \quad \text{as} \quad n \to \infty\,. \end{aligned} \tag{1.83}$$

1C Newey–West Method Details for the Simulations

From Newey and West (1987), the estimate of the variance of a sample mean is given by

$$\hat{\sigma}_{NW}^2 = \hat{\sigma}_0^2 + 2 \cdot \sum_{j=1}^{m} k_j \cdot \hat{\sigma}_j^2, \tag{1.84}$$

where

$$\hat{u}_t = y_t - \bar{y} \tag{1.85}$$

are the de-meaned residuals of y_t and

$$\hat{\sigma}_j^2 = \frac{1}{n} \sum_{t=j+1}^{m} \hat{u}_t \hat{u}_{t-j}, \quad \text{for } j \geq 0 \tag{1.86}$$

are lag-j sample covariances of the residuals u_t. Newey and West (1994) developed a method for automatically selecting m based on minimizing the mean squared error of the estimator was introduced. For the Bartlett triangular shaped kernel, where

$$k_j = 1 - j/(m+1), \tag{1.87}$$

their solution for m has the form of

$$m = \left\lceil \hat{\gamma} n^{1/3} \right\rceil. \tag{1.88}$$

The detailed derivation of a formula for $\hat{\gamma}$ can be found in Newey and West (1994). We have used their formula to compute the following expected values of $\hat{\gamma}$ for a Gaussian AR(1) model with correlation coefficient ϕ ranging from 0 to 0.5 and sample sizes $n = 60, 100, 200$ by running 100,000 Monte Carlo simulations.

1D Cross Validation Algorithm

The full algorithm of the K-fold cross validation procedure for glmExpEN is shown in the CV algorithm below, where A is the matrix

ϕ	n=60	n=100	n=200
0	1.39	1.31	0.88
0.1	1.15	1.16	0.79
0.2	0.98	1.04	0.76
0.3	0.90	0.96	0.81
0.4	0.90	0.97	0.94
0.5	0.98	1.08	1.13

Table 1.4: $\hat{\gamma}$ for Gaussian AR(1) Returns

$$A = \begin{bmatrix} f_1 & f_1^2 & \cdots & f_1^{d-1} & f_1^d \\ \vdots & & & & \vdots \\ f_{n/2} & f_{n/2}^2 & \cdots & f_{n/2}^{d-1} & f_{n/2}^d \end{bmatrix}, \tag{1.89}$$

$b = (p_{IF,1}, \ldots, p_{IF,n/2})$ is the vector for the periodogram of the influence function series, c is a constant, n_λ is the number of candidate λs, and NLL is the negative log-likelihood defined by equation (1.61). Step 2 of the algorithm ensures that the solution is a vector of all zeros under λ_{max}, step 3 constructs the vector of candidate λs as a sequence equally spaced between λ_{min} and λ_{max} in the log scale, and the FISTA algorithm used in step 5 stands for Fast Iterative Shrinkage-Thresholding Algorithm. See Chapter 2 for details.

Algorithm 1 CV Algorithm

1. Set A, b, c and n_λ
2. Compute $\lambda_{max} = \max_j \left|\sum_{i=1}^{N} (1 - b_i) A_{ij}\right|$.
3. Compute $\lambda_{min} = c * \lambda_{max}$
4. Compute vector of candidate λ's, $\lambda_{vec} = \exp(\text{seq}(\log(\lambda_{min}), \log(\lambda_{max}), \text{length} = n_\lambda))$
5. For $j = 1 : n_\lambda$
   - $\lambda = \lambda_{vec}(j)$
   - for $i = 1 : K$
     - randomly divide (A, b) into K partitions. Let $(A, b)_{test}$ be one of the partitions, $(A, b)_{train}$ be the union of the rest of the partitions.
     - use the FISTA algorithm on the model defined A_{train}, b_{train} and λ to find the solution x_{train}
     - $\text{NLL}_{ij} = NLL(x_{train}, A_{test}, b_{test})$
   - $\text{NLL}_j = \sum_i \text{NLL}_{ij}$
6. λ_{best} is the λ that results in the smallest value among all NLL_j
7. use the FISTA algorithm on the model defined A, b and λ_{best} to find the solution x_{best}

1E R Packages Installation and Section 5 Code

We developed two packages that work together for the computation of standard errors of risk/performance measures. The EstimatorStandardError package is the main package and the glmnetRcpp is the C++ implementation of the optimization procedure for fitting GLM-EXP/EN model. To install these packages, run the following code:

```
Sys.setenv( "PKG_CXXFLAGS"="-std=c++1y" )
library(devtools)
install_github("chenx26/glmnetRcpp")
install_github("chenx26/EstimatorStandardError")
```

And the following R code is used to generate the table of standard errors of the expected shortfall estimator for the EDHEC database.

```
rm(list=ls())
library(EstimatorStandardError)
#--------- Load data
data(edhec)
colnames(edhec)=c("CA", "CTAG", "DIS", "EM",
                  "EMN", "ED", "FIA", "GM", "L/S",
                  "MA", "RV", "SS", "FoF")
nboot=500
return.coeffs = TRUE
d.GLM.EN = 5
alpha.ES = 0.05
alpah.EN = 0.5
#--------- Set random seed
seed = 12345
k_fold_iter = 1000
set.seed(seed)
#--------- Expected Shortfall estimates and their S.E.'s
se.ES=ES.SE(edhec, p = 1 - alpha.ES, method="historical",nsim = nboot,
            se.method = c("IFiid","IFcor","BOOTiid","BOOTcor"),
            standardize = FALSE, return.coeffs = return.coeffs,
            k_fold_iter = k_fold_iter,
            d.GLM.EN = d.GLM.EN,
            alpha = alpha.EN)
```

```
tmp = se.ES$IFcor
SEs = sapply(tmp, function(x) x[[1]])
se.ES$IFcor = SEs

se.ES.df = printSE(se.ES, round.digit  = 3, valonly = TRUE)

ar1 = apply(edhec, 2,
            function(x) {
                    res = ar(x, order.max = 1)
                    if(length(res$ar) == 0)
                            return(0)
                    return(res$ar[1])
            }
)

IF.ar1 = apply(edhec, 2,
               function(x) {
                       res = ar(ES.IF(x), order.max = 1)
                       if(length(res$ar) == 0)
                               return(0)
                       return(res$ar[1])
               }
)

PI.IFcor = (se.ES.df$IFcor / se.ES.df$IFiid - 1) * 100

# use Newey-West to compute SE's
```

```
NW = NULL
for(i in 1:ncol(edhec)){
        data = coredata(edhec[,i])
        data.IF = ES.IF(data, alpha.ES = alpha.ES)
        NW = c(NW, sqrt(nse.nw(data.IF)))
}
PI.NW = (NW / se.ES.df$IFiid - 1) * 100

se.ES.df = cbind(se.ES.df, PI.IFcor, NW, PI.NW)[c(1,2,3,7,6,8,4,5)]
se.ES.df = cbind(se.ES.df, ar1, IF.ar1)

knitr::kable(se.ES.df, digits = c(3,3,3,3,0,0,3,3,2,2), align = 'c')
```

1F RMSE and Relative RMSE and AR(1) Simulation Results for Sample Size 200

This appendix presents the RMSE and relative RMSE for sample size 200. Figure 1.9 is like Figures 1.3 and 1.4. Figure 1.10 is like Figures 1.5 and 1.6. Figure 1.11 is like Figures 1.7 and 1.8.

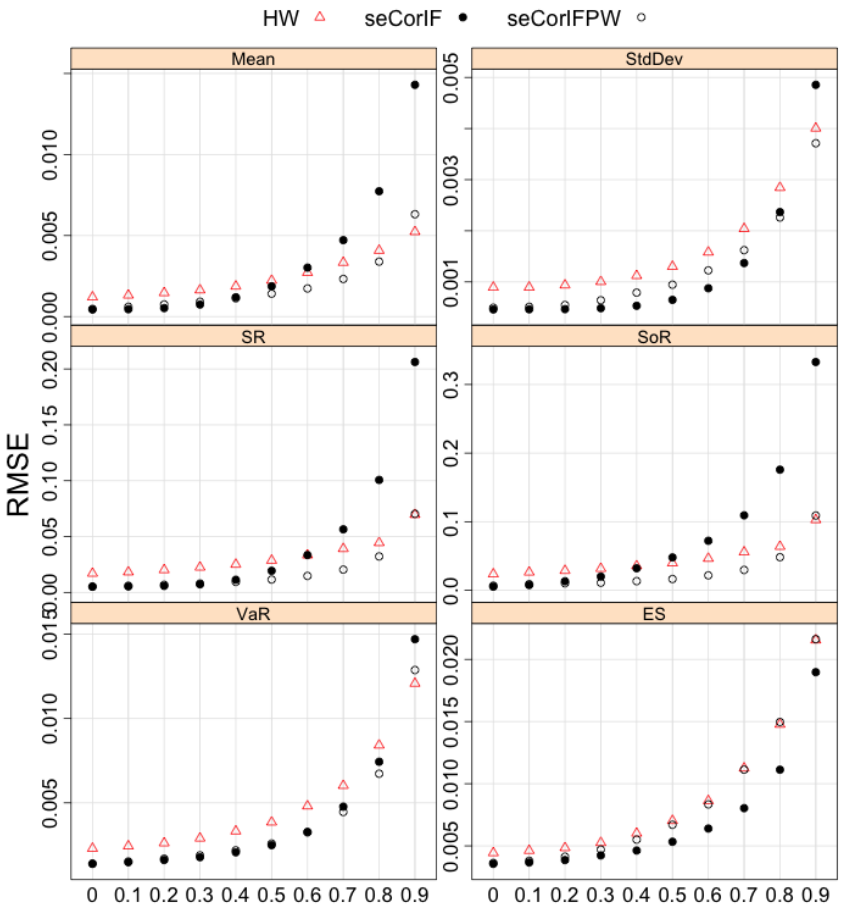

Figure 1.9: RMSE Comparison of New Standard Error Estimators for AR(1) Returns, $n = 200$

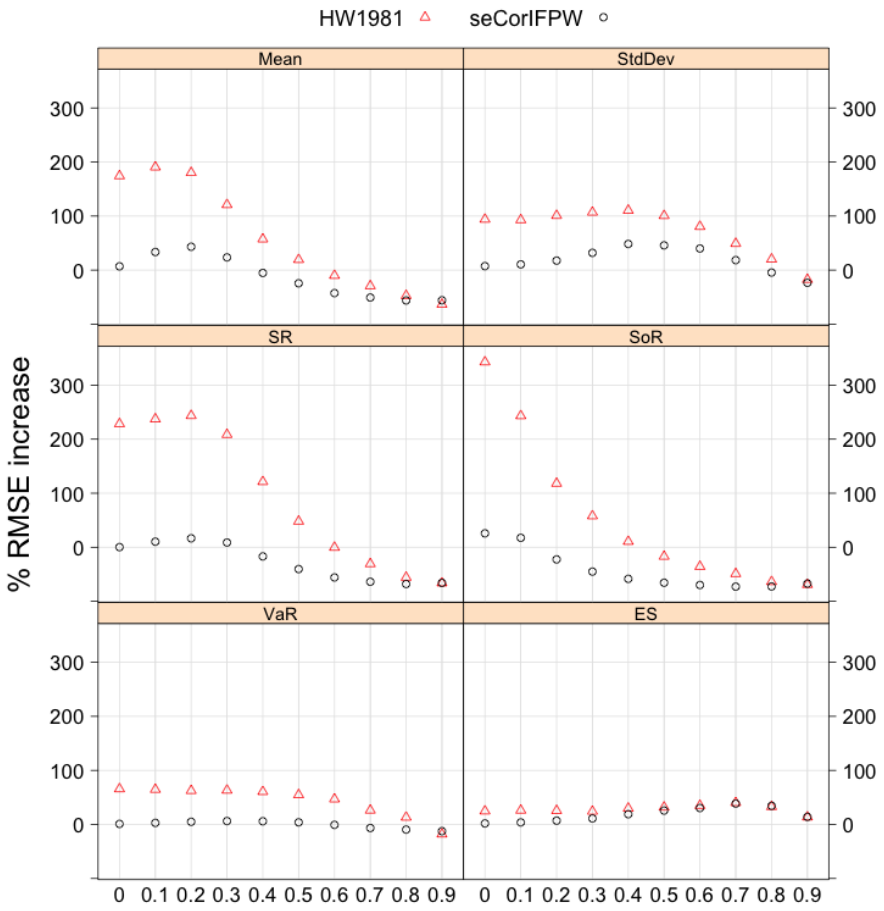

Figure 1.10: Percent RMSE Increase of HW and seCorIFPW over seCorIF for AR(1) Returns, $n = 200$

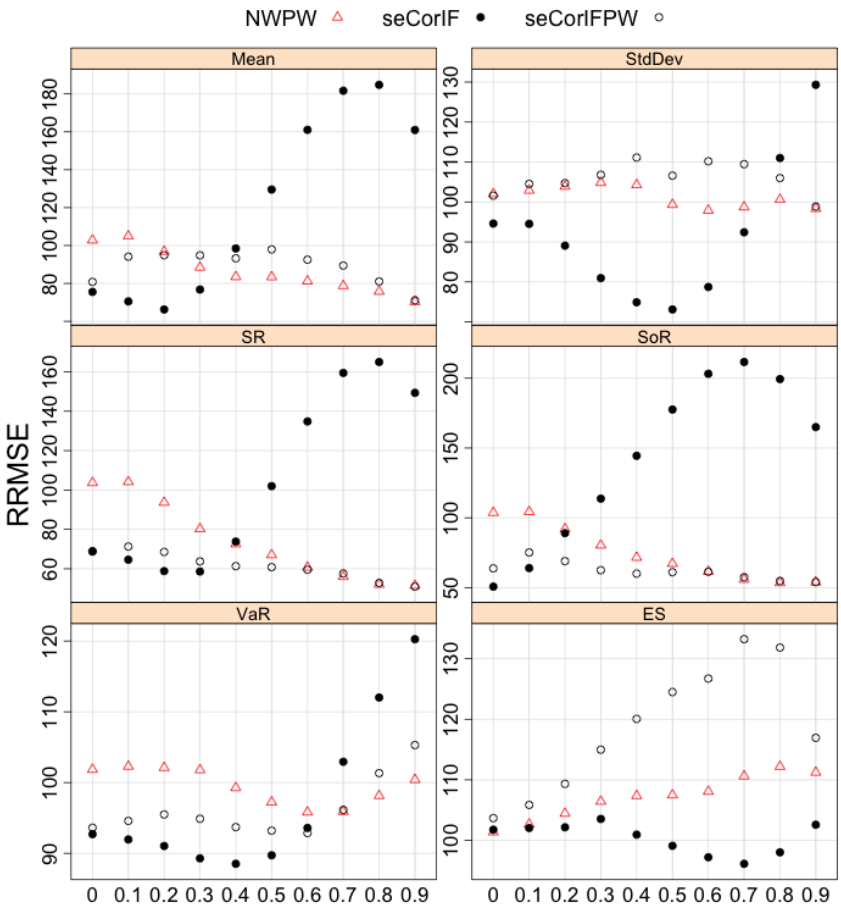

Figure 1.11: RRMSE Comparison of New Standard Error Estimators for AR(1) Returns, $n = 200$

1G Bias and Standard Deviation AR(1) Simulation Results

For the results in this Appendix we compute the following:

- An approximate bias Bias_{METHOD} of the each method-dependent standard error estimator:

$$\text{Bias}_{METHOD} = \text{SE}_{METHOD} - \text{SE}_{\text{True}} \tag{1.90}$$

- An approximate standard deviation SD_{METHOD} of each method-dependent standard error estimator:

$$\mathrm{SD}_{METHOD} = \sqrt{\frac{1}{M}\sum_{m=1}^{M}\left(\mathrm{SE}_{METHOD,m} - \mathrm{SE}_{METHOD}\right)^2} \tag{1.91}$$

The bias results for n =60, n = 100 and n = 200 are shown in Figures 1.12, 1.13 and 1.14, respectively.

The results in Figure 1.12 for n = 60 show that all three SE estimator methods have biases that are generally negative with an essentially zero value at ϕ = 0, and remain relatively small to various degrees for ϕ less or equal to 0.4 or so, beyond which the biases become rapidly more negative. Overall, the seCorIF method has the most negative biases, dramatically so for the larger values of ϕ, the HW method has the least negative biases, and the seCorIFPW method biases fall in between those two. The HW method has the curious feature of having distinctly positive biases for middle values of the sample mean estimator (the Mean), with slightly small positive biases for the Mean, Sharpe ratio and Sortino ratio estimators (which use the sample mean in the numerator). In typical portfolio applications where asset returns have lag-one correlations not greater than 0.5 or so, the seCorIFPW estimator would be the best choice with regard to bias. Furthermore, in the event returns have lag-one correlations larger than 0.5, the seCorIFPW estimator has considerably smaller biases than the seCorIF estimator and biases that are only slightly worse than the HW estimator.

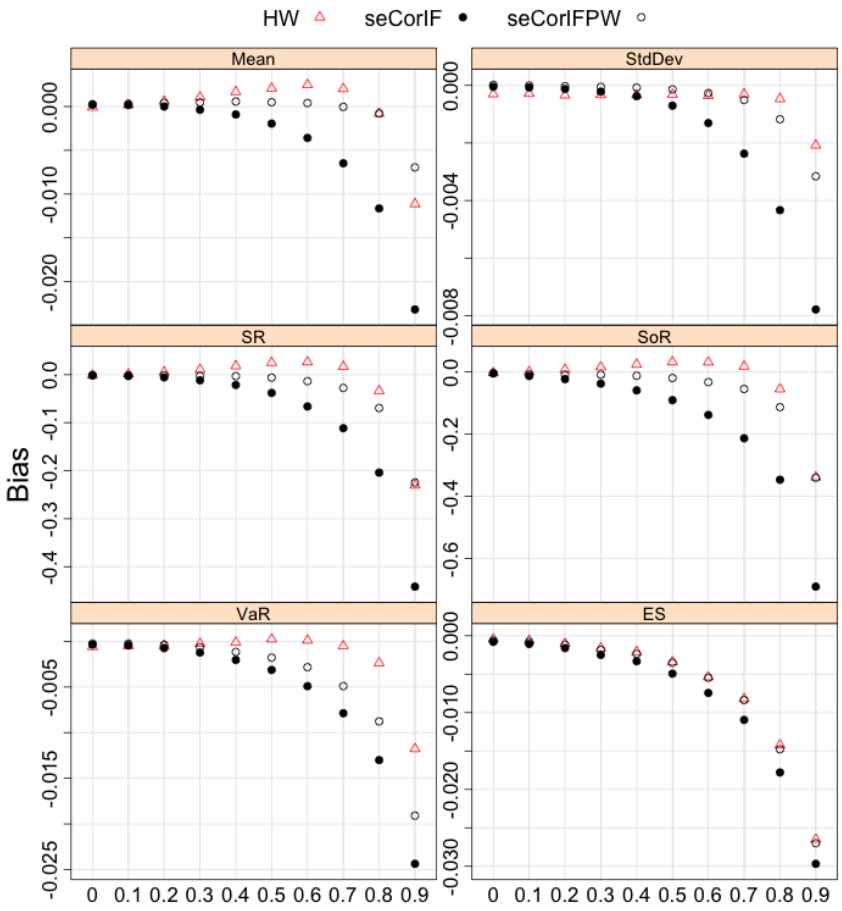

Figure 1.12: Bias Comparison of New Standard Error Estimators for AR(1) Returns, $n = 60$

The bias results for $n = 100$ and $n = 200$ in Figures 1.13 and 1.14 show that the general shapes and ordering of the bias versus ϕ curves are quite similar to those for $n = 60$, with absolute values of the biases decreasing with increasing n.

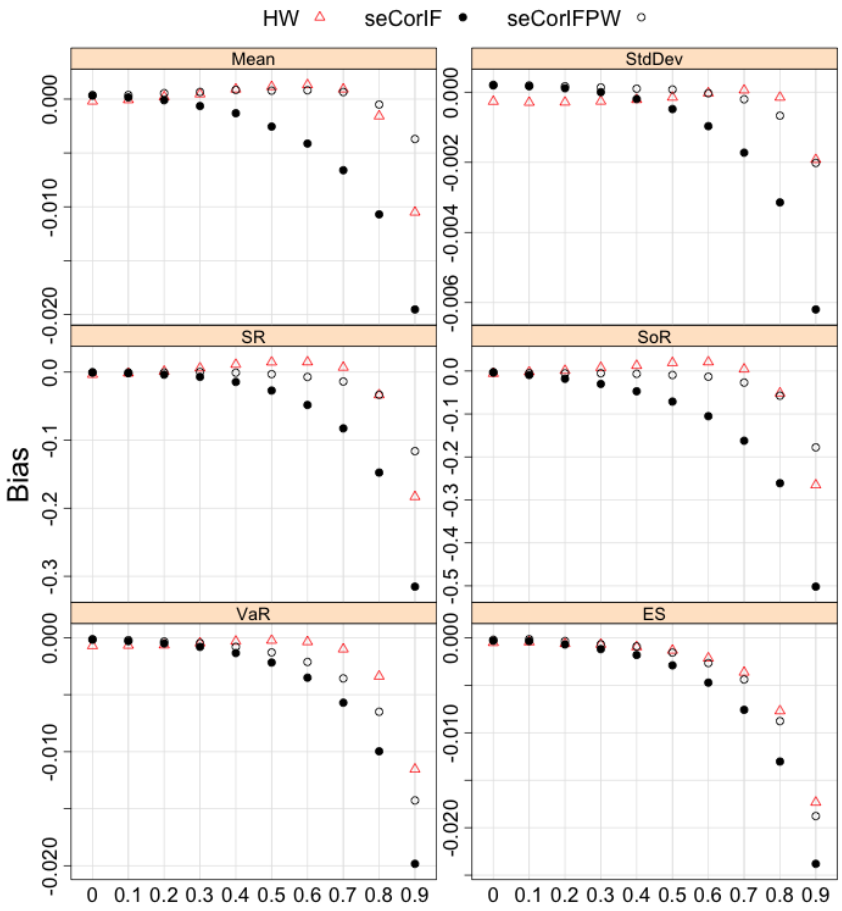

Figure 1.13: Bias Comparison of New Standard Error Estimators for AR(1) Returns, $n = 100$

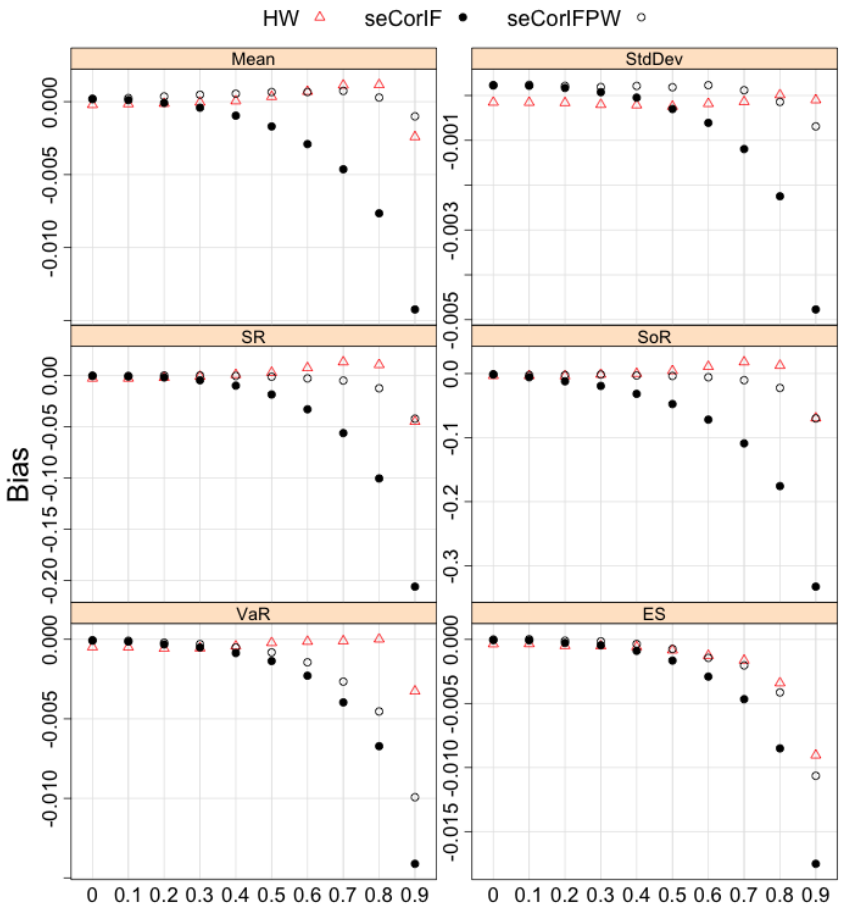

Figure 1.14: Bias Comparison of New Standard Error Estimators for AR(1) Returns, $n = 200$

The standard deviation results for n =60, $n = 100$ and $n = 200$ are shown in Figures 1.15, 1.16 and 1.17, respectively.

The results in Figure 1.15 for $n = 60$ show that standard deviations of all estimators increase with ϕ, except for StdDev and VaR at $\phi = 0.9$ and ES at the two largest values of ϕ. Here the seCorIF method has uniformly the smallest standard deviations, and the HW method has the uniformly largest standard deviations except for a few exceptions at the largest values of ϕ. The seCorIFPW method has standard deviation values in between those of the seCorIF and HW methods, with values closer to those of the HW method for larger values of ϕ, except for the VaR estimator.

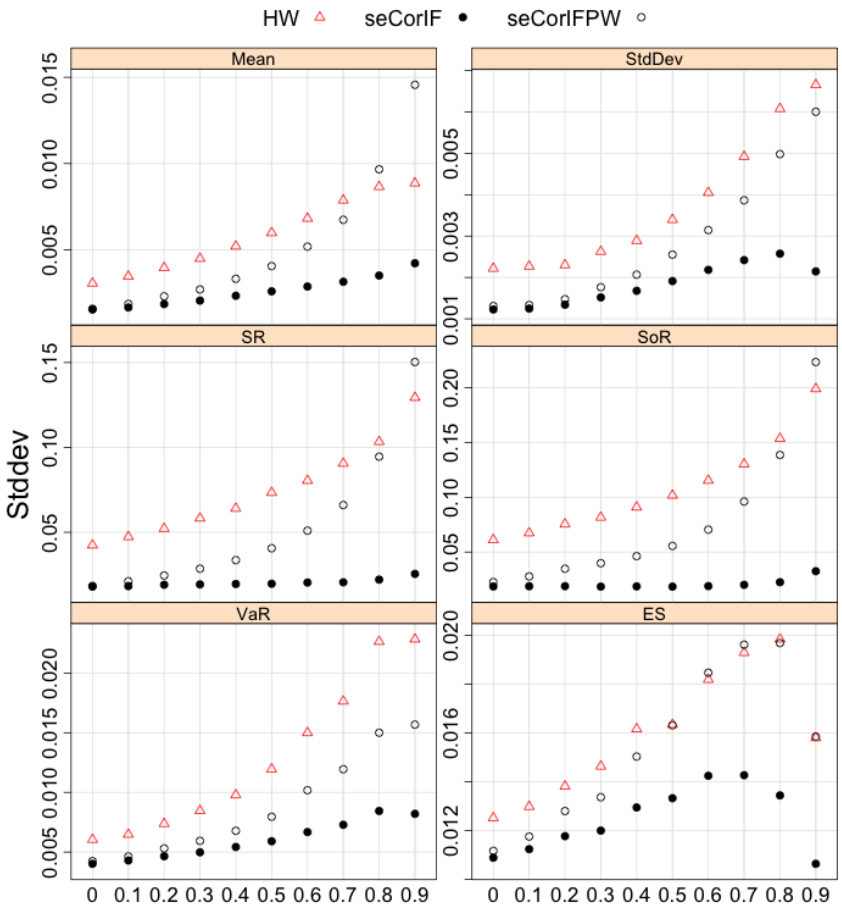

Figure 1.15: Standard Deviation Comparison of New Standard Error Estimators for AR(1) Returns, $n = 60$

The standard deviation results for $n = 100$ and $n = 200$ in Figures 1.16 and 1.17 show that the general shapes and ordering of the standard deviation versus ϕ curves are quite similar to those for $n = 60$, with values decreasing roughly like $n^{-1/2}$ as one would expect.

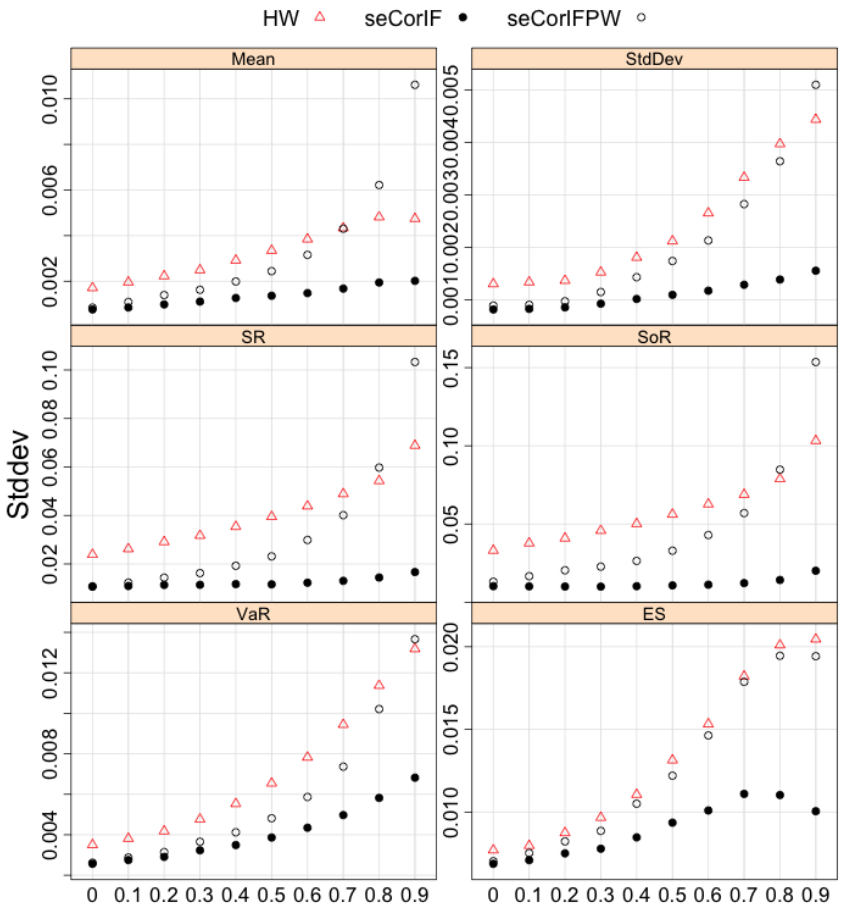

Figure 1.16: Standard Deviation Comparison of New Standard Error Estimators for AR(1) Returns, $n = 100$

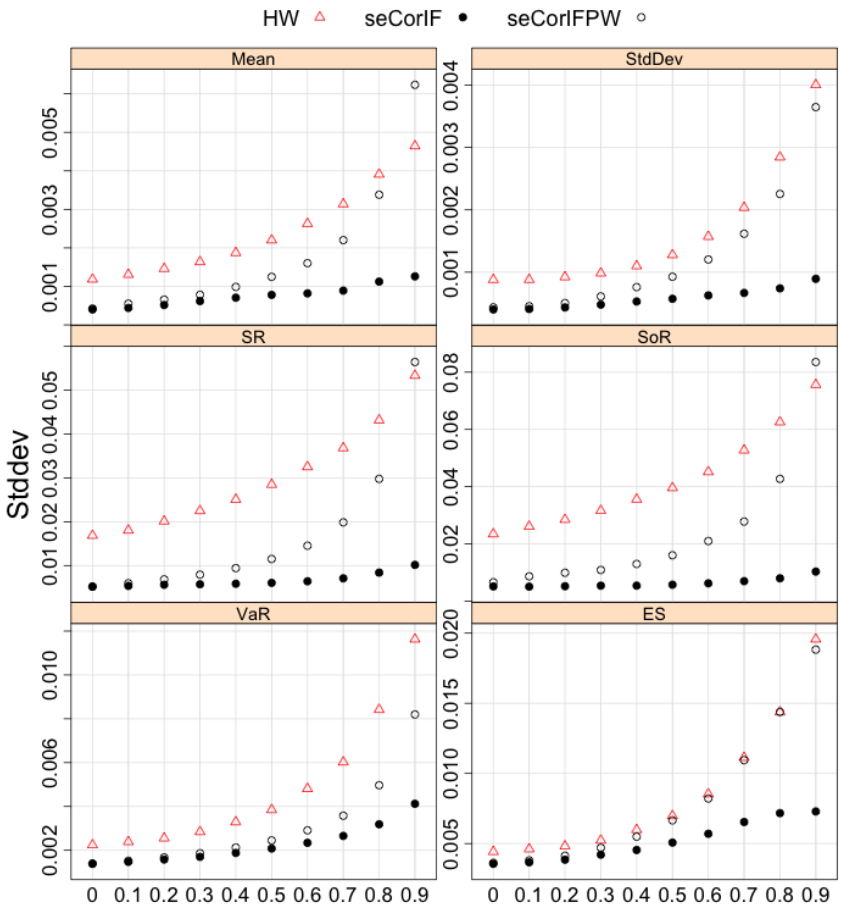

Figure 1.17: Standard Deviation Comparison of New Standard Error Estimators for AR(1) Returns, $n = 200$

1H Relative Bias and Standard Deviation Comparisons AR(1) Simulation Results

Similar to the definition of the RRMSE in Equation (1.78), we define relative bias and relative standard deviation as

$$\text{RBias}_{METHOD} = \left(\frac{\text{Bias}_{METHOD}}{\text{Bias}_{NW}} \right) \times 100\,. \tag{1.92}$$

$$\text{RStdDev}_{METHOD} = \left(\frac{\text{RStdDev}_{METHOD}}{\text{RStdDev}_{NW}} \right) \times 100 \,. \tag{1.93}$$

Figures 1.18, 1.19 and 1.20 contain relative bias comparisons of the seCorIF, seCorIFPW and NWPW methods relative to the NW method for sample sizes $n = 60$, $n = 100$ and $n = 200$, respectively. These results show that in general, the seCorIFPW method has the smallest biases, while the biases of seCorIF are larger than those of NWPW except for very small ϕ. These results suggest that prewhitening helps to reduce bias very effectively.

Figures 1.21, 1.22 and 1.23 report the relative standard deviation of the seCorIF, seCorIFPW and NWPW methods. The results are almost the exact opposite compared with the relative bias results. The seCorIF has the smallest RStdDev across almost all risk/performance measures, often substantially so. The RStdDev's of NWPW and seCorIFPW are in general comparable with each other, except for ES, where NWPW has much smaller RStdDev's. These results suggest that the main source of the SE estimator methods' variability is the prewhitening.

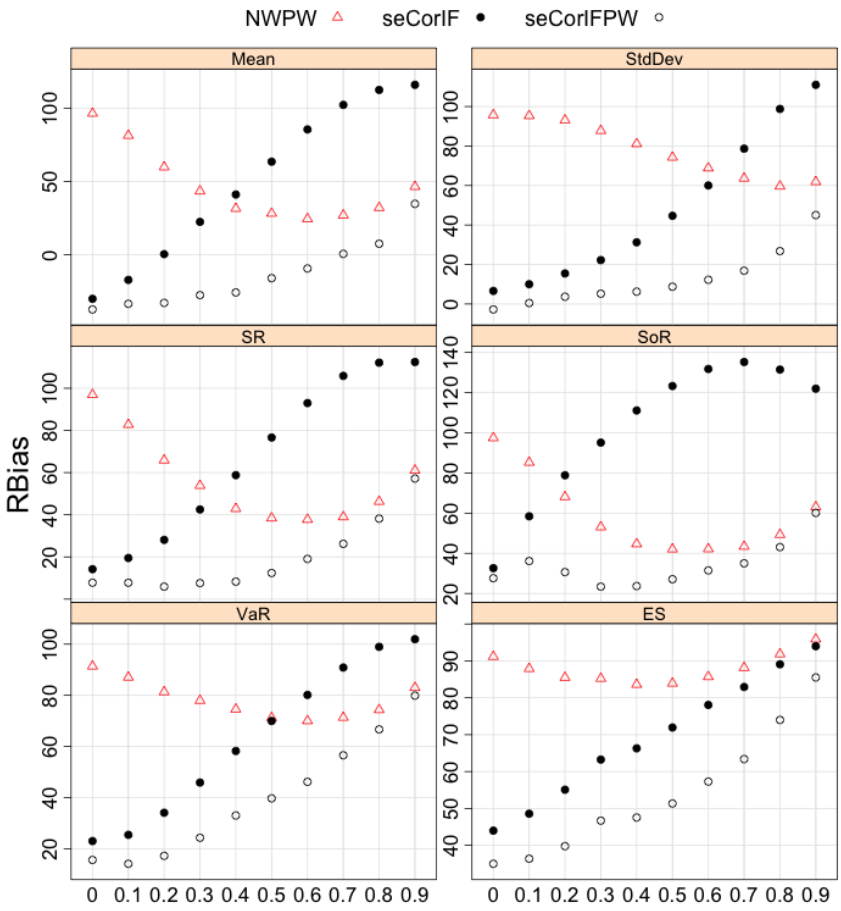

Figure 1.18: S.E. Estimators Relative Biases for AR(1) Returns, $n = 60$

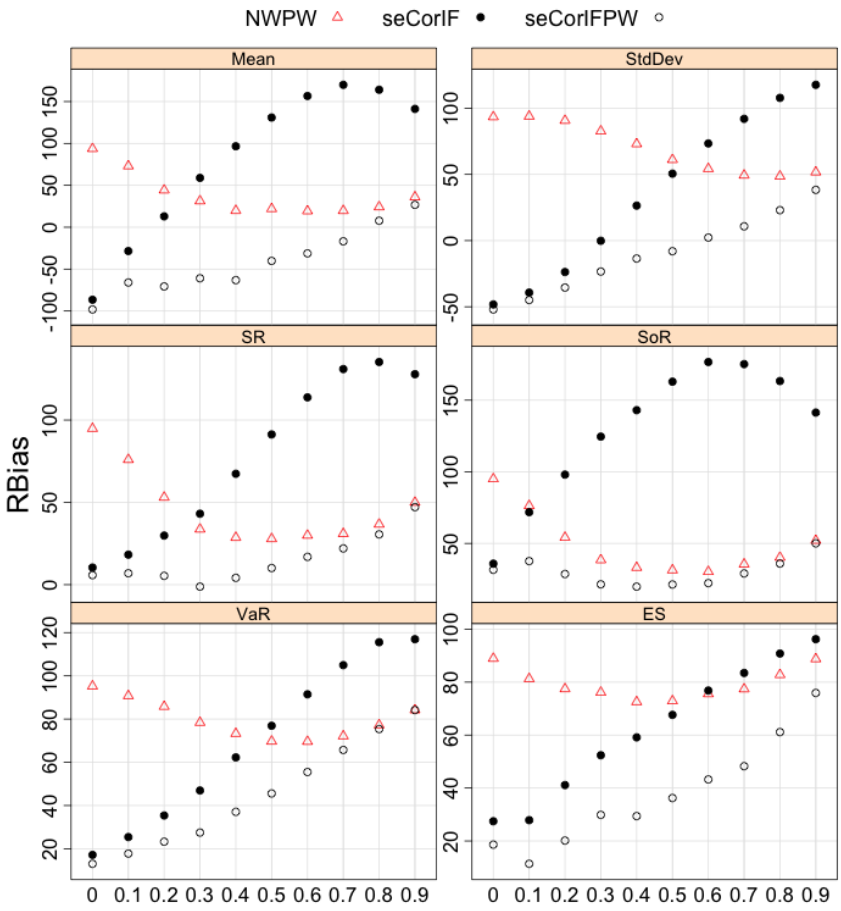

Figure 1.19: S.E. Estimators Relative Biases for AR(1) Returns, $n = 100$

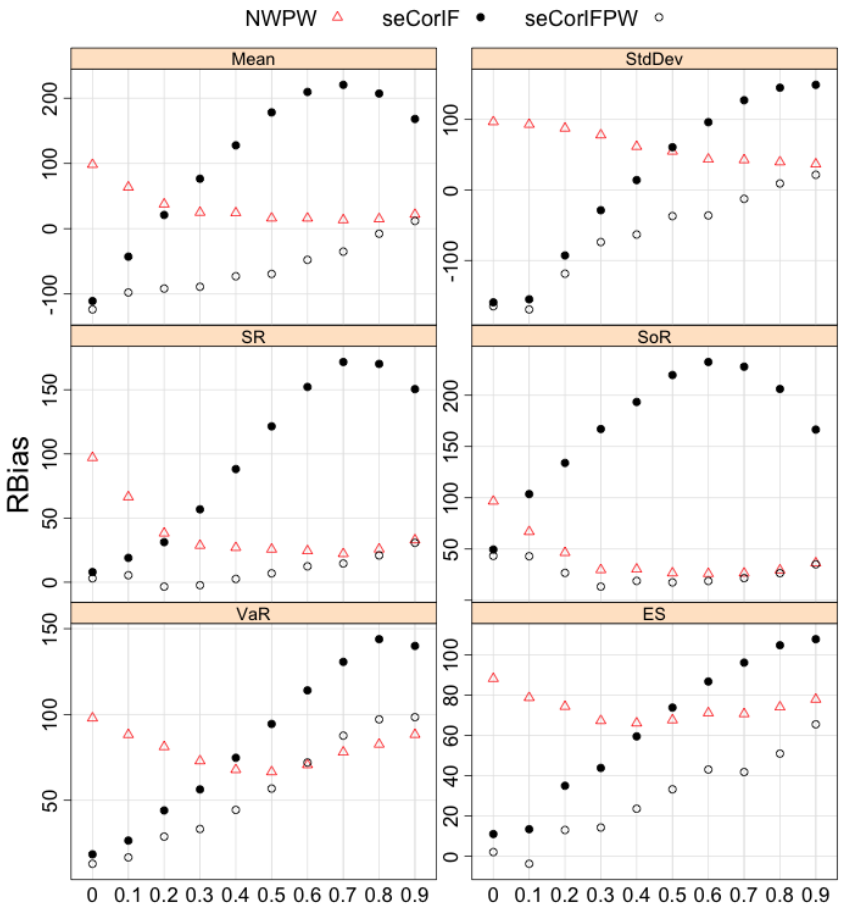

Figure 1.20: S.E. Estimators Relative Biases for AR(1) Returns, $n = 200$

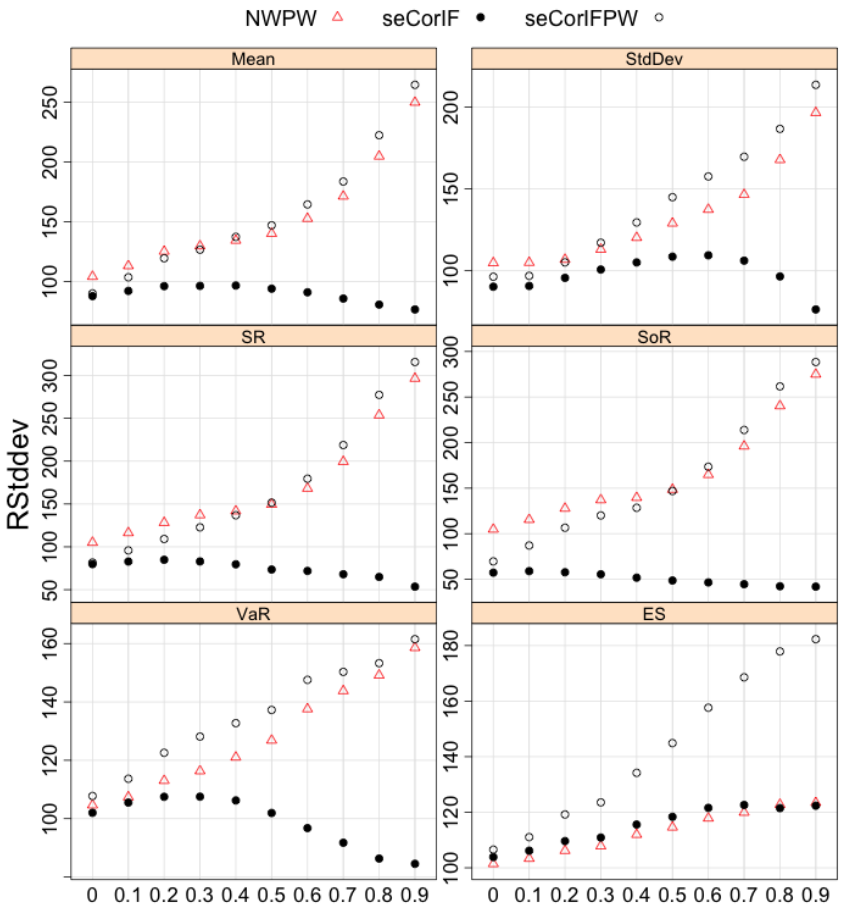

Figure 1.21: S.E. Estimators Relative Standard Deviation for AR(1) Returns, $n = 60$

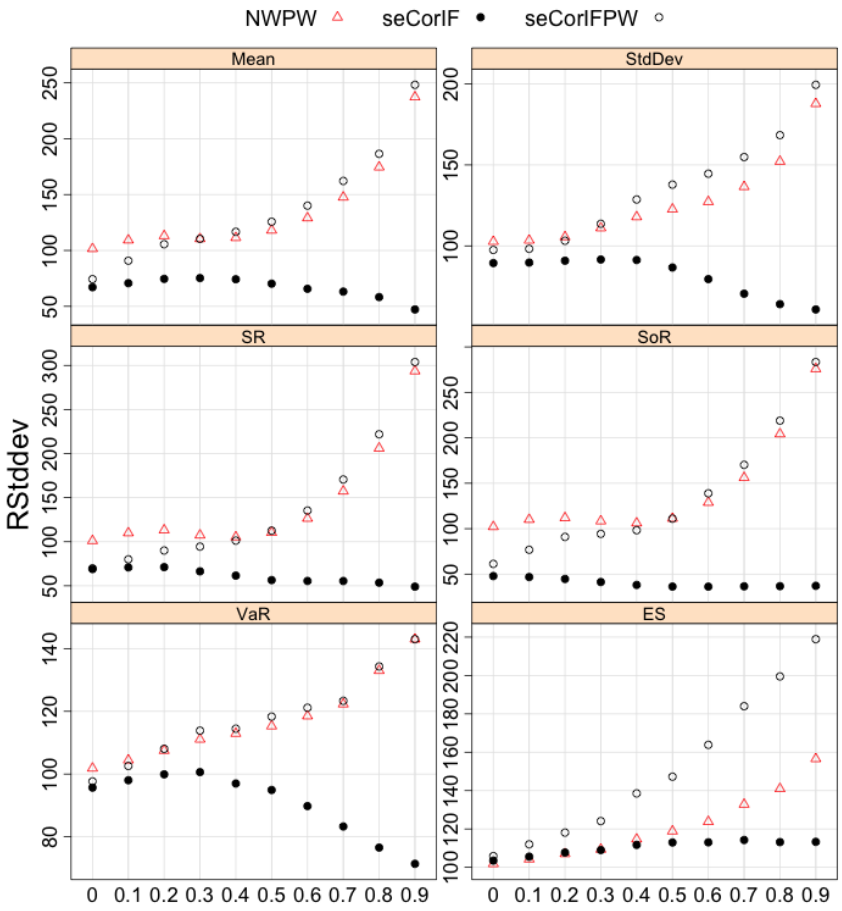

Figure 1.22: S.E. Estimators Relative Standard Deviation for AR(1) Returns, $n = 100$

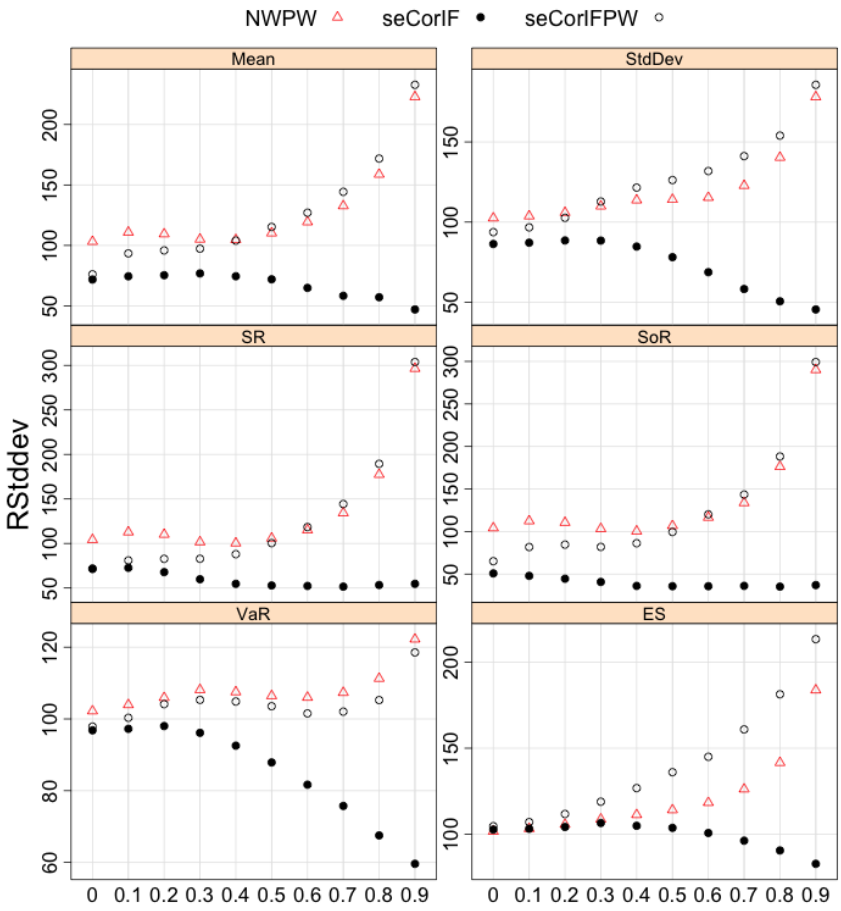

Figure 1.23: S.E. Estimators Relative Standard Deviation for AR(1) Returns, $n = 200$

Chapter 2

Generalized Linear Model for Gamma Distributed Variables via Elastic Net Regularization

2.1 Introduction

In Chapter 1, we have developed a general method for computing standard errors of risk and performance estimators for financial assets with serially correlated returns that relies on a GLM for exponentially distributed data with elastic net (EN) regularization, which is a special case of a gamma distribution. Though less prevalent, GLMs for gamma distribution have been used to model positive data in applications such as insurance claims (Jong and Heller 2008) and clotting times of blood (McCullagh and Nelder 1989), among others. The motivation for Chapter 2 is two-fold. First of all, we discuss in detail the special optimization method used for the elastic net regularized GLM fitting, and second we extend the glmExpEN model in Chapter 1 to generalized linear model for gamma-distributed variables with elastic net regularization. There is currently no R package that implements a parallelizable GLM for gamma-distributed variables with elastic net

regularization (glmGammaEN), so the work in Chapter 2 fills this gap. Table 2.1 is a summary of existing R packages for GLM, to the authors' best knowledge. In particular, we provide an efficient, parallelizable R package that can fit a GLM model with EN regularization for the gamma family.

Package	Function	Support Gamma Dist	Model Selection	Multicore Parallel
glmnet	glmnet()	No	EN Regularization	Yes
h2o	h2o.glm()	No	EN Regularization	No
stats	glm()	Yes	AIC/BIC	Yes
bestglm	bestglm()	No	Subset AIC/BIC	Yes
glmGammaEN	**glmGammaEN()**	**Yes**	**EN Regularization**	**Yes**

Table 2.1: Comparison of R implementations for GLM

The optimization problem for glmGammaEN is more challenging than that for linear or logistic regression. First, the objective function required to perform the inference does not have a global quadratic upper bound. Such bounds are very useful for designing simple and efficient first-order methods for regularized log-likelihood estimation. Without the bound, a line search is needed in theory to ensure descent. Instead, we estimate a quadratic bound locally using the functional form of the Gamma to get a fast and robust method for the problem. We implement a safeguard line search, but it is never activated. Second, the addition of the EN regularization term makes the objective function non-smooth. We use a proximal gradient-based first order method (Parikh et al. 2014) instead of the classical gradient descend method for non-smooth optimization.

In Section 2.2, we formulate the GLM inference problem, discuss first-order methods for elastic net (EN) regularization. Section 2.3 specializes the results in Section 2.2 to Gamma distributions and detail the algorithm we implemented. Section 2.4 presents simulation results. Section **??** shows a real world example of using glmGammaEN to compute the standard errors of the expected shortfall estimates for a collection of hedge fund return time series. We end with a discussion in Section 2.6.

2.2 First-Order Methods for Regularized GLM

The GLM inference problem is formulated as follows. Suppose we wish to predict an output y of a certain system on an input $x \in \mathbb{R}^n$. Let us also make the following two assumptions: (*i*) the relationship between the input x and the output y is fairly simple and (*ii*) we have available examples $x_i \in \mathbb{R}^n$ together with inexactly observed responses y_i whose expectations are μ_i for $i = 1, \ldots, m$. The tuples $\{(y_i, x_i)\}_{i=1}^m$ are the observed sample data. The responses y_i can have special restrictions; for example they may be counts, indicate class membership, or be non-negative, such as concentration of sugar in the blood. To build the GLM, suppose the distribution of y_i is given by the following exponential family distribution:

$$f(y_i; \theta_i) = \exp\{(y_i\theta_i - b(\theta_i))/\phi + c(y_i, \phi)\} \quad i = 1, 2, \cdots, m\,. \tag{2.1}$$

The expectation of y_i is given by

$$\mu_i = E(y_i) = b^{'}(\theta_i). \tag{2.2}$$

It is worth noting that the gamma distribution we are interested in is a special case of the exponential family distribution. We will specialize the results in this section to gamma distribution in Section 2.3.

Let g be the link function, then the following relationship holds

$$\eta_i = \boldsymbol{x}_i^T\boldsymbol{\beta} = g(\mu_i),\ i = 1, 2, \cdots, m\,. \tag{2.3}$$

Therefore, the negative loglikelihood for the observation paris $\{(y_i, x_i)\}_{i=1}^m$ is given by

$$L(\boldsymbol{\beta}) = \sum_{i=1}^{m}(y_i\theta_i - b(\theta_i))/\phi + c(y_i, \phi). \tag{2.4}$$

Since y_i are known and we assume ϕ is a known positive cosntant, minimizing $L(\boldsymbol{\beta})$ is equivalent

to the optimization problem

$$\min_{\boldsymbol{\beta}} -\sum_{i=1}^{m} y_i\theta_i - b(\theta_i). \tag{2.5}$$

To get the explicit form of the objective function in terms of β, from (2.2) and (2.3), we can write θ_i as a function of β:

$$\theta_i = (b')^{-1}\left(g^{-1}\left(x_i^T\boldsymbol{\beta}\right)\right). \tag{2.6}$$

Then the optimization problem is:

$$\min_{\boldsymbol{\beta}} -\sum_{i=1}^{m} y_i\,(b')^{-1}\left(g^{-1}\left(x_i^T\boldsymbol{\beta}\right)\right) - b\left((b')^{-1}\left(g^{-1}\left(x_i^T\boldsymbol{\beta}\right)\right)\right). \tag{2.7}$$

We are interested in an extension of (2.5) that includes non-smooth regularization terms $R(\boldsymbol{\beta})$ (including 1-norm, elastic net, or constraints):

$$\min_{\boldsymbol{\beta}} -\sum_{i=1}^{m} y_i\theta_i - b(\theta_i) + R(\boldsymbol{\beta}). \tag{2.8}$$

Due to the non-smoothness introduced by the regularization term, model (2.8) cannot be solved using naive gradient descent or Newton's method. For the remainder of this section, we give a general introduction to the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA), which is especially powerful to solve this kind of problem.

Let H be the objective function in (2.7). Its gradient is said to satisfy the Lipschitz property with constant L if

$$\|\nabla H(\alpha) - \nabla H(\beta)\| \leq L\|\alpha - \beta\| \quad \forall \alpha, \beta. \tag{2.9}$$

When H is twice continuously differentiable, any bound on the operator norm of $\nabla^2 H$ is a Lipschitz constant for ∇H. For example, under normal distribution and identity link function, $H(\beta) = \frac{1}{2}||X\beta - y||^2$. The Lipschitz constant for $\nabla H(\beta) = X^T(X\beta - y)$ is the largest eigenvalue of $X^T X$. Any Lipschitz constant for ∇H gives a simple tight global upper bound for H:

$$H(\beta) \leq H(\beta_0) + (\beta - \beta_0)^T \nabla H(\beta_0) + \frac{L}{2}||\beta - \beta_0||^2. \tag{2.10}$$

A simple iterative strategy is to minimize the upper bound for H at each iteration, without modifying $R(x)$, which may be non-smooth (1-norm) or infinite valued (box-constraint). Given iterate β^k, the next iterate β^+ is found as follows:

$$\begin{aligned} \beta^+ &= \arg\min_{\beta} (\beta - \beta^k)^T \nabla H(\beta^k) + \frac{L}{2}||\beta - \beta^k||^2 + R(\beta) \\ &= \arg\min_{\beta} \frac{1}{2L}||\beta - (\beta^k - \frac{1}{L}\nabla H(\beta^k)||^2 + R(\beta) \\ &:= \text{prox}_{\frac{1}{L}R}(\beta^k - \frac{1}{L}\nabla H(\beta^k)). \end{aligned} \tag{2.11}$$

The proximity operator $\text{prox}_{\frac{1}{L}R}(z)$ defined in (2.11) should be thought of as a simple subroutine. It is the minimizer of the problem

$$\min_{\beta} \frac{1}{2L}||\beta - z||^2 + R(\beta), \tag{2.12}$$

and is available in closed form for a wide variety of regularizers $R(\beta)$, including 1-norm , the elastic net, and simple constraints (Combettes et al. 2011).

Algorithm 2 FISTA for Regularized GLM

1. Initialize $\beta^1 = 0$, $\omega = 0$, $\kappa = 0$, $s_1 = 1$, compute $d^1 = \nabla H(\beta^1)$. Let L be a Lipschitz constant for H.
2. While $||\text{prox}_R(\omega^\kappa - d^\kappa)|| > \epsilon$
   - Set $\kappa = \kappa + 1$.
   - update $\beta^\kappa = \text{prox}_{L^{-1}R}(\omega^{\kappa-1} - \alpha d^{\kappa-1})$.
   - set $s_\kappa = \frac{1+\sqrt{1+4s_{\kappa-1}^2}}{2}$
   - set $\omega^\kappa = \beta^\kappa + \frac{s_{\kappa-1}-1}{s_\kappa}(\beta^\kappa - \beta^{\kappa-1})$.
   - Compute $d^\kappa = \nabla H(\beta^\kappa)$.
3. Output β^κ.

The iteration (2.11) is known as the proximal gradient iteration, and converges with the same rates as gradient descent on the smooth function H (Nesterov 2013). The iteration can be accelerated to achieve a better rate of convergence by using an auxiliary iterative sequence; the most famous example of such an algorithm is FISTA (Beck et al. 2009). The FISTA algorithm is only slightly more complicated than (2.11), and is detailed in Algorithm 2.

Linear regression and logistic regression have easily computable Lipschitz constants of $||X||^2$ and $\frac{1}{4}||X^2||$ respectively. Unfortunately, the models for the Gamma GLMs are constructed using logarithmic and exponential b, and these functions do not have a global quadratic upper bound. A safeguard linesearch is required to ensure descent, i.e., that the objective at β^+ is smaller than the objective at β^k unless β^k is a stationary point. A FISTA with line search replaces L^{-1} in Algorithm 2 with an iterative step-size selection procedure to ensure

$$H(\beta^{\kappa+1}) + R(\beta^{\kappa+1}) < H(\beta^\kappa) + R(\beta^\kappa). \tag{2.13}$$

However, in practice Algorithm 2 is a descent method when L is locally estimated at each iteration using a simple heuristic. We discuss the heuristic and other specifics of the Gamma GLM in the

next section.

2.3 Adapting FISTA to the Gamma Family with EN Regularization

In this section, we adapt the general scheme of fitting the GLM model with a regularization term to the special case of GLM for gamma-distributed response variables with the elastic net regularization (glmGammaEN). The optimization problem is as follows

$$\min_{\beta} G(\beta) + R_{EN}(\beta; \lambda, \alpha), \tag{2.14}$$

where $G(\beta)$ is the general objective function $H(\beta)$ specialized to the gamma distribution, λ is the regularization parameter and $R_{EN}(\beta; \lambda, \alpha)$ is the Elastic Net regularization term. The section proceeds as follows. Section 2.3.1 derives the formula for $G(\beta)$, the gradient of $G(\beta)$ and the proximal of the EN regularization term. In Section 2.3.2, we develop a customized method of obtaining a local approximation of the Lipschitz constant to avoid a line search (which requires additional function evaluations). Section 2.3.3 shows how cross-validation can be used to select the optimal regularization parameter and presents the complete algorithm to fit GLM for gamma-distributed responses with EN regularization.

2.3.1 $G(\beta)$, $\nabla G(\beta)$ and Proximal of EN

The probability density function of the gamma distribution is given by

$$f(y; k, \theta_i) = \frac{1}{\Gamma(k)\theta_i^k} y^{k-1} e^{-\frac{y}{\theta_i}}, \; k > 0, \; \theta_i > 0, \tag{2.15}$$

where k is the shape parameter and θ is the scale parameter. The expectation of a gamma random variable Y_i is given by

$$E(Y_i) = \mu_i = k\theta_i. \tag{2.16}$$

Using the natural logarithm link function, the relationship between the expectation of Y_i and the linear component of GLM is given by

$$\log\left(E(Y_i)\right) = \log(k\theta_i) = x_i^T\beta, \tag{2.17}$$

where k is the shape parameter, which is assumed to be the same for all examples, θ_i is the scale parameter for the ith example, x_i is the ith row of the predictor matrix X, β is the coefficients for the GLM model. Therefore, the scale parameter can be written as

$$\theta_i = e^{x_i^T\beta}/k. \tag{2.18}$$

The objective function of problem 2.14 over all examples is given by

$$\begin{aligned} F(\beta; y,\, x,\, k,\, \lambda,\, \alpha) =& G(y; k,\, \beta) + R_{EN}(\beta; \lambda,\, \alpha) \\ =& -\sum_{i=1}^{m} \log f(y_i; k,\, \theta_i) + R_{EN}(\beta; \lambda,\, \alpha) \\ =& \sum_{i=1}^{m} \log\Gamma(k) + k\log\theta_i - (k-1)\log y_i + \frac{y_i}{\theta_i} + R_{EN}(\beta; \lambda,\, \alpha) \\ =& \sum_{i=1}^{m} \log\Gamma(k) + k\cdot x_i^T\beta - k\cdot\log k - (k-1)\log y_i + k\cdot y_i e^{-x_i^T\beta} + \lambda\left(\alpha\left\|\beta\right\|_1 + \frac{1-\alpha}{2}\left\|\beta\right\|_2^2\right). \end{aligned} \tag{2.19}$$

The partial derivative of G with respect to β_j is

$$\frac{\partial G}{\partial \beta_j} = \sum_{i=1}^{m} k \cdot \left(1 - y_i e^{-x_i^T \beta}\right) x_{ij}. \tag{2.20}$$

Note that in this work, we get an estimate of the shape parameter k by fitting a classic GLM on the data first, then use it as a constant to fit the glmGammaEN model. Therefore, in the following sections, only the coefficients β_j are variables.

In order to use FISTA, we also need the proximity operator (prox) of the elastic regularization term

$$R_{EN}(\beta; \lambda, \alpha) = \lambda \left(\alpha \|\beta\|_1 + \frac{1-\alpha}{2} \|\beta\|_2^2 \right). \tag{2.21}$$

From (Parikh et al. 2014), the prox of $R_{EN}(\beta; \lambda, \alpha)$ is given by

$$\text{prox}_{tR_{EN}}(v) = \frac{1}{1 + t\lambda(1-\alpha)} \text{sgn}(v) \max(|v| - t\lambda\alpha, 0), \tag{2.22}$$

where v is a vector, while sgn(v) and max(v) act on v element-wise.

2.3.2 Computing a Local Upper Bound

Algorithm 2 requires a global Lipschitz constant for the gradient of the objective. However, the gradient (2.20) does not have a global Lipschitz constant, because the exponential function cannot have a global quadratic upper bound. A local quadratic approximation can be computed efficiently and used in lieu of a global Lipschitz constant. The local quadratic upper bound for gamma model (2.19) is given by

$$L(x) = \|X\|_F^2 \left(\sum_{i=1}^{m} k^2 \cdot \left(1 - y_i e^{-x_i^T \beta}\right)^2 \right), \tag{2.23}$$

where X is a matrix constructed by stacking the regressor vectors x_i as the ith row. The idea behind (2.23) is to get a data-dependent local quadratic upper bound, analogous to those we have for linear and logistic regression. The bound we use is conservative in practice, since we never need to activate the safe-guard line search.

2.3.3 Optimal λ via Cross Validation

With the results from section (2.3.1) and (2.3.2), we are able to solve the optimization problem (2.14) if λ and α are given. The choice of λ and α has a strong impact on the problem. Following the suggestions by Friedman et al. (2010), we assume that α is determined by the user and focus on using cross-validation to find the optimal value for λ. The cross-validation procedure is as follows:

1. Compute the smallest λ that gives an all-zero solution for the regularized problem, call this λ_{max}.

2. The lower bound of the grid is given by $\lambda_{min} = \epsilon * \lambda_{max}$, where ϵ is a user-defined constant with a default value of 0.001.

3. The grid of λ consists of n_λ values between λ_{min} and λ_{max} equally spaced in the log scale, where n_λ is a user-defined constant with a default value of 100.

4. For each λ, perform n-fold cross validation and compute the average G, where n is a user-defined constant with a default value of 10.

5. Choose the λ with the smallest average G for use in the final model.

Computing λ_{max}

The value λ_{max} should be such that $\beta_k = 0$ satisfies the optimality conditions for the problem. Denoting the jth element of β_k as β_k^j, optimality is equivalent to a fixed point condition across all j:

$$\begin{aligned}\beta_k^j =& \text{prox}_{\frac{1}{L}R_{EN}}(\beta_k^j - \frac{1}{L}\nabla G^j) \\ =& \frac{1}{1+\frac{1}{L}\lambda(1-\alpha)}\text{sgn}\left(\beta_k^j - \frac{1}{L}\nabla G^j\right)\max\left(\left|\beta_k^j - \frac{1}{L}\nabla G^j\right| - \frac{1}{L}\lambda\alpha,\ 0\right).\end{aligned} \tag{2.24}$$

The definition of λ_{max} requires the fixed point condition above to hold when $\beta_k^j = 0$ for all j, so we have

$$\begin{aligned}0 =& \text{prox}_{\frac{1}{L}R_{EN}}(-\frac{1}{L}\nabla G^j) \\ =& \frac{1}{1+\frac{1}{L}\lambda(1-\alpha)}\text{sgn}\left(-\frac{1}{L}\nabla G^j\right)\max\left(\left|\frac{1}{L}\nabla G^j\right| - \frac{1}{L}\lambda\alpha,\ 0\right)\end{aligned} \tag{2.25}$$

A sufficient condition to ensure above is that

$$\lambda\alpha \geq \left|\nabla G^j\right| = \left|\sum_{i=1}^{m}\left(k - \frac{y_i}{\theta_i}\right)x_{ij}\right| = \left|\sum_{i=1}^{m}\left(k - \frac{y_i}{1/k}\right)x_{ij}\right| = \left|\sum_{i=1}^{m}k(1-y_i)\,x_{ij}\right|. \tag{2.26}$$

To make sure that $\beta_k = 0$ satisfies the optimality condition across all j, we take λ_{max} to be the largest of all such λ:

$$\lambda_{max} = \alpha^{-1}\max_j\left|\sum_{i=1}^{m}k(1-y_i)\,x_{ij}\right|. \tag{2.27}$$

2.3.4 GLM for Gamma Response Variables with Elastic Net (glmGammaEN)

Algorithm 3 gives the complete pseudo code for glmGammaEN.

Algorithm 3 glmGammaEN

1. Set the matrix of regressors x, the vector of the observations y, λ_{max}, ϵ and n_λ
2. get an esitmate of k by fitting a GLM model without regularization
3. Compute $\lambda_{max} = \alpha^{-1} \max_j \left| \sum_{i=1}^{N} k\,(1 - y_i)\, x_{ij} \right|$
4. Compute $\lambda_{min} = \epsilon * \lambda_{max}$
5. Compute vector of candidate λ's, $\lambda_{vec} = \exp(\text{seq}(\log(\lambda_{min}), \log(\lambda_{max}), \text{length} = n_\lambda))$
6. For $j = 1 : n_\lambda$
   - $\lambda = \lambda_{vec}(j)$
   - for $i = 1 : k$
     - randomly divide the pairs (y_i, x_i) into k partitions. Let $(y, x)_{test}$ be one of the partitions, $(y, x)_{train}$ be the union of the rest of the partitions.
     - use the FISTA algorithm on the model defined by $(y, x)_{train}$ and λ to find the solution β_{train}
     - $G_{ij} = G(\beta_{train}, x_{test}, y_{test})$
   - $G_j = \sum_i G_{ij}$
7. λ_{best} is the λ that results in the smallest value among all G_j
8. use the FISTA algorithm on the model defined x, y and λ_{best} to find the solution β_{best}

Two alternative ways of choosing the best λ

In Algorithm 3, we choose the λ that minimizes G, which seems to be the optimal choice. However, this is not always the case. It is worth noting that G computed in Algorithm 3 are just estimates of the true prediction errors. Therefore, there are uncertainties associated with these estimates. To account for these uncertainties, we propose two alternative ways of choosing the "best" λ. The first alternative is to choose the maximum λ with the corresponding $G \leq G_{min} + \text{SD}_{G_{min}}$. The second alternative is to choose the maximum λ with the corresponding G smaller than the αth percentile of

all G from cross-validation. We discuss the performance of both alternatives along with Algorithm 3 in the numerical experiment section.

2.4 Numerical Experiment

We use Monte Carlo (MC) simulations to demonstrate the superior performance of the three variants of glmGammaEN compared with the standard GLM method. The tested methods include:

1. Standard GLM for gamma responses (glmGamma),
2. Our method described in Algorithm 3 (glmGammaEN), which is more conservative in terms of variable selection,
3. The percentile variant described in section 2.3.4 with 10th percentile threshold (glmGammaEN.percentile), which is more aggressive in variable selection,
4. Fitting the glmGamma without the zero coefficients identified by glmGammaEN.percentile (glmGammaEN.percentile.nonzero), which is the most aggressive in variable selection,
5. The one-standard-deviation variant described in section 2.3.4 (glmGammaEN.1sd),
6. Fitting the glmGamma without the zero coefficients identified by glmGammaEN.1sd (glmGammaEN.1sd.nonzero).

We run 1000 MC situations. In the ith MC run, the following steps are performed:

1. Set $n = 100$ and $p = 15$, where n is the number of examples and p is the dimension of the coefficient vector β.
2. Generate the $n \times p$ predictor matrix X from i.i.d. normal distribution $N \sim (0, 1)$.

3. Generate vector of length p from i.i.d. normal distribution $N \sim (0, 1)$ and randomly set 10 of the elements to zero. Denote the resulting vector as β_{true}.

4. Compute vector of the expectations $\mu_{true} = \exp(X\beta_{true})$.

5. Compute the vector of scales $\theta_{true} = \mu_{true}/k$.

6. Generate $n \times 1$ vector of response variables y by getting one sample from the gamma distribution specified by each element in the vector θ_{true}.

7. Use different methods to compute the solution $\beta^{(i)}_{METHOD}$ using X, y as the input and save $\beta^{(i)}_{METHOD}$.

2.4.1 Error of Fitted Coefficients

We compute the following performance metrics for the error of the fitted coefficients

1. L1 Norm of the difference between β_{METHOD} and β_{true}

$$\text{error.L1}_{METHOD} = ||\beta_{METHOD} - \beta_{true}|| \,. \tag{2.28}$$

2. error.L1_{METHOD} as a percentage of $||\beta_{true}||$

$$\%\ \text{error.L1}_{METHOD} = ||\beta_{METHOD} - \beta_{true}|| \,/\, ||\beta_{true}|| \cdot 100. \tag{2.29}$$

Table 2.2 summarizes the error of fitted coefficients for different GLM methods.

As shown in Table 2.2, the glmGamma method does a reasonably good job, with a percentage error of 10.4%. By adding Elastic Net regularization and choosing the λ with the smallest G in cross-validation, we get a slight improvement in percentage error, down to 9.3%. The glmGammaEN.percentile method has a similar percentage error of 9.2%. The glmGammaEN.1sd method

	error.L1	% error.L1
glmGamma	0.41	10.4
glmGammaEN	0.37	9.3
glmGammaEN.percentile	0.36	9.2
glmGammaEN.1sd	0.55	14.0
glmGammaEN.percentile.nonzero	0.33	8.3
glmGammaEN.1sd.nonzero	0.23	5.8

Table 2.2: Performance Summary of Different GLM methods (the last two methods fit regular glmGamma model with variables with non-zero coefficients)

has the highest percentage error of 14%. However, if we drop the zero coefficients identified by glmGammaEN.percentile and glmGammaEN.1sd, and then perform a regular glmGamma, the percentage errors drop significantly. The glmGammaEN.percentile.nonzero method has a percentage error of 8.3%, down from 9.2% and the glmGammaEN.1sd.nonzero method has a percentage error of 5.8%, down from 14%. We conjecture that the reduction in percentage error is due to the variable selection power of our new methods. In the next subsection we explore the variable selection performance of different methods.

2.4.2 Variable Selection Performance Analysis

We compare the variable selection performance of different GLM methods by examining the following two statistics:

1. Number of correctly identified zero coefficients, denoted as zeros.correct$_{METHOD}$
2. Number of correctly identified nonzero coefficients denoted as nonzeros.correct$_{METHOD}$

Table 2.3 summarizes the variable selection performance for different GLM methods.

	zeros.correct	nonzeros.correct
glmGamma	0	5
glmGammaEN	1.976	5
glmGammaEN.percentile	4.771	5
glmGammaEN.1sd	7.815	5

Table 2.3: Performance Summary of Different GLM methods

All the methods studied have successfully included the nonzero coefficients in their solution, yet the number of identified zero coefficients varies a lot. Out of the 10 zero coefficients in the true solution, the glmGamma method fails to identify any zero coefficients. On average, the glmGammaEN method manages to find roughly 2 zero coefficients. The glmGammaEN.percentile method identifies approximately 5 zero coefficients, which presents a great improvement. Most notably, the glmGammaEN.1sd method find an incredible 8 zero coefficients on average and 80% chance of correctly identifying 7 or more zero coefficients. By dropping the variables with zero coefficients, we are essentially removing noise in the dataset.Therefore, the percentage error of these methods are much better, as shown in the previous section.

Figure 2.1 further visualizes the distribution of the number of correctly identified zero coefficients. The blue bars represent the results for glmGammaEN. Notice that there is a 35% chance that glmGammaEN does not identify any zero coefficients and the probability decreases as the number of correctly identified zero coefficients increases. This shows that glmGammaEN is very conservative in terms of variable selection. The red bars show the results for glmGammaEN.percentile. The distribution is roughly bell-shaped and the peak occurs at 6 zero coefficients with a probability of 15%. This indicates that glmGammaEN.percentile is considerably more aggressive than glmGammaEN, but still not satisfactory. The black bars show the results for glmGammaEN.1sd. The distribution is concentrated around 8 and 9 zero coefficients, which account for more than 0.5 probability. This shows that the glmGammaEN.1sd method performs extremely well in variable selection.

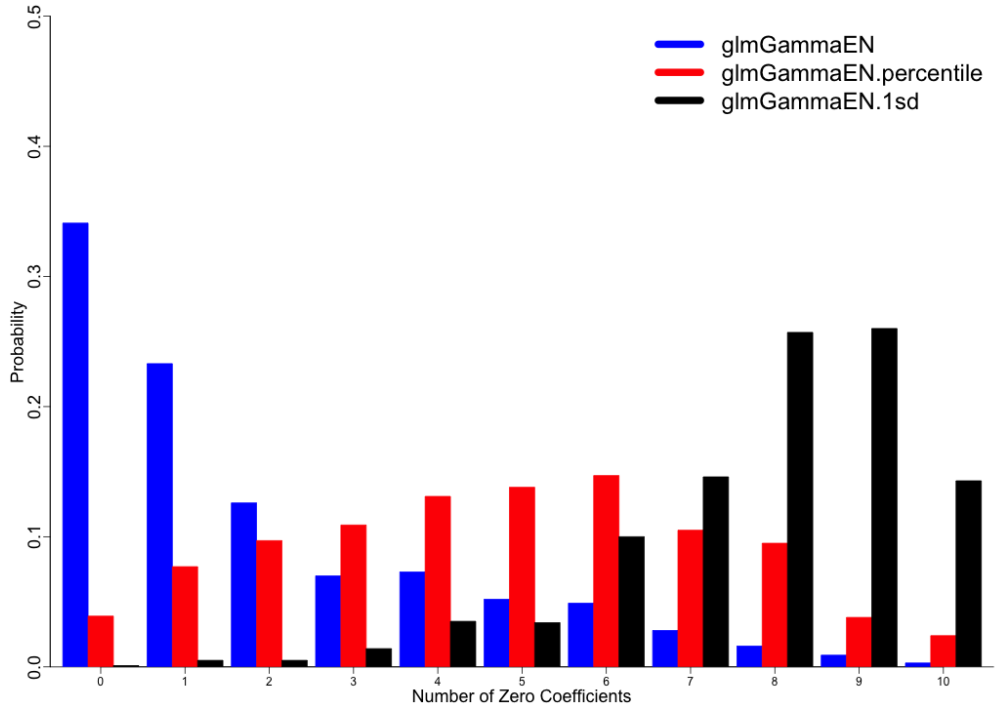

Figure 2.1: Histogram of number of zero coefficients selected over 1000 simulations

2.5 Expected Shortfall Standard Error of Hedge Fund Returns

We apply the glmGammaEN algorithm to compute the standard error of expected shortfall (ES) estimator for 13 hedge fund returns series. The methodology is same as Chapter 1 except we use glmGammaEN instead of glmExpEN to fit the generalized linear model with elastic net constraint.

Table 2.4 shows the standard errors estimated using different methods. The results from seCorIF and seCorIF.gamma are quite similar to each other.

	ES	seIidIF	seCorIF	PI.seCorIF	seCorIF.gamma	PI.seCorIF.gamma	ar1	IF.ar1
CA	-0.049	0.018	0.032	82	0.027	50	0.6	0.51
CTAG	-0.045	0.004	0.004	3	0.004	3	0	0
DIS	-0.043	0.013	0.018	43	0.018	45	0.53	0.36
EM	-0.089	0.024	0.028	15	0.029	21	0.34	0.12
EMN	-0.019	0.008	0.008	6	0.012	64	0.28	0
ED	-0.044	0.011	0.015	28	0.014	27	0.41	0.23
FIA	-0.042	0.016	0.025	60	0.024	55	0.5	0.46
GM	-0.023	0.003	0.003	-2	0.004	17	0	0
L/S	-0.044	0.009	0.012	35	0.012	38	0.29	0.33
MA	-0.025	0.006	0.007	10	0.007	9	0.32	0
RV	-0.031	0.01	0.014	52	0.015	62	0.48	0.49
SS	-0.111	0.013	0.015	13	0.018	43	0.15	0.12
FoF	-0.039	0.009	0.012	32	0.012	36	0.35	0.3

Table 2.4: ES Standard Errors for Hedge fund Returns

Table 2.5 shows the fitted coefficients for each hedge fund returns series. We can see that the variable selection is working very well. The glmGammaEN model seems to result in more zero coefficients than glmExpEN model. Notice that the fitted shape parameters are bigger than 1 for all the hedge funds, which show that using the gamma distribution instead of exponential distribution is likely to have improved the model.

	beta0	beta1	beta2	beta3	beta9	beta10	beta11	beta12	zero.coeffs	shape
CA	-2.237	-0.819	-8.106	-9.561	-0.5	-0.251	-0.108	-0.027	0	5.73
CTAG	-5.929	0	0	0	0	0	0	0	12	1.36
DIS	-2.982	-2.718	-3.925	2.087	0	0	0	0	5	1.78
EM	-2.055	-2.916	8.393	-6.364	-0.059	0	0	0	3	1.78
EMN	-3.759	-9.592	-21.114	140.738	-20.777	-10.717	-5.338	-2.578	0	12.85
ED	-3.448	-1.903	-0.265	0	0	0	0	0	10	1.87
FIA	-2.43	-3.372	-3.619	-0.019	0	0	0	0	7	1.17
GM	-6.171	-4.017	5.607	4.989	0	0	0	0	5	1.19
L/S	-3.819	-1.888	-3.495	-0.008	0	0	0	0	6	1.4
MA	-5.003	-0.525	0	0	0	0	0	0	11	2.23
RV	-3.313	-4.304	-2.201	-0.442	0	0	0	0	9	2.44
SS	-2.979	-9.025	21.119	-4.339	0	0	0	0	4	1.27
FoF	-3.761	-2.204	-1.917	0	0	0	0	0	8	1.63

Table 2.5: Fitted glmGammaEN Coefficients for Hedge Fund Returns for Hedge fund Returns

2.6 Discussion

In this chapter, we developed an implementation for GLM fitting with gamma-distributed data and elastic net regularization. One reason the gamma distribution may not be available in standard software is that the objective function is a composition of an exponential model with a linear map, and so does not have a global quadratic upper bound. We developed a customized, accelerated proximal gradient method by using local quadratic estimates; although a safeguard line search is implemented, it was never activated across our entire suite of experiments. We also provide a straightforward cross-validation scheme to determine the optimal value of the regularization parameter. Numerical experiments show the advantage of these methods over standard GLM without regularization. The new methods have both smaller error in fitted coefficients and superior variable selection performance. The choice of regularization parameter is very important, and we recommend two simple strategies: (1) conservative: using the parameter that corresponds to the smallest negative log-likelihood in the cross validation, and (2) aggressive: using the one-standard-deviation

rule.

Chapter 3

Practical Impact of Our Results and Future Work

3.1 Practical Impact of Our Results

We introduced a uniformed framework for computing the standard errors of risk and performance measures under serial correlation. It performs as well or better than the state-of-the-art Newey--West method. We also provided an efficient way of fitting glmGammaEN model. We implemented our research in open source R packages for both academics and practitioners to use. The influence function based framework to compute the standard errors for risk and performance measures under serial correlation was conceptualized in early 2016. By that time, we were quite confident about the overall structure of the framework, which consists of four main steps. First, we compute influence function transformed time series (IFTS). Second, we fit a glmExpEN model to the periodogram of IFTS. Third, we use the fitted glmExpEN model to estimate the spectral density of IFTS at frequency zero, which is an approximation of the asymptotic variance of the risk/performance estimator. Finally, we use a standard finite sample approximation method to compute the standard error of the risk/performance estimator from the asymptotic variance es-

timate. We were fortunate to secure a funding of $5500 from Google for a the Google Summer of Code (GSoC) project to implement the framework from May 2016 to August 2016 (https://summerofcode.withgoogle.com/archive/2016/projects/5770545454907392/). The project went smoothly and resulted in the open source R package EstimatorStandardError (ESE). The R package is freely available for anyone to use or modify at https://github.com/chenx26/EstimatorStandardError, and a detailed vignette is available at https://github.com/chenx26/EstimatorStandardError/blob/master/vignettes/EstimatorSEVignette.pdf.

However, the original ESE package depended on the H2O package (https://cran.r-project.org/web/packages/h2o/index.html) to fit the glmExpEN model. The H2O package was useful for us to write a prototype R package due to the limited time for GSoC projects but was too slow to run tens of thousands of Monte Carlo simulations we needed to do for our research. We believe there are two reasons for this. First, one cannot parallelize the computation using foreach R package in conjunction with the H2O package. Second, for each Monte Carlo simulation run, the overhead of using R to call external JAVA code used in H2O package seemed to be too high. These aspects, in combination with the large number of MC simulations we needed to run, resulted in painfully long times for the MC studies, often more than five days. This situation motivated us to work on an efficient implementation of the glmExpEN model using Rcpp, which is an R package that allows seamless integration of efficient C++ code into R packages.. So we wrote a proposal for a second GSoC project, and obtained another $6000 funding from Google to develop this capability from May 2017 to August 2017. This resulted in the R package glmnetRcpp, available at https://github.com/chenx26/glmnetRcpp. A detailed vignette can be found at https://github.com/chenx26/glmnetRcpp/blob/master/vignettes/glmnetRcpp_vignette.pdf. The ESE package was modified to use glmnetRcpp instead of H2O, which led to a reduction of computation time of MC simulations by an order of magnitude.

In 2018 we continued to expand upon the above work. Since exponential distribution is a special case of gamma distribution, and the exponential distribution of periodograms is only asymptotically exact, we believed that extended our work on glmExpEN to accommodate gamma-distributed variables might result in better performance of our standard errors estimation method for serially correlated returns. This has led to the work described in Chapter 2 and an R package called glmGammaEN. I then served as a mentor of the GSoC project "PerformanceAnalytics Standard Errors" from May 2018 to August 2018, the Google awarded to a very strong University of British Columbia Statistics Department Graduate student Anthony Christidis. During that period of time, I helped Anthony add more risk and performance measures to the `ESE` package, and begin the integration of `ESE` with the glmGammaEN package, and overall integration with the widely used `PerformanceAnalytics R` package. The `PerformanceAnalytics` (`PA`) package is an open source R package that is believed to be used by hundreds of organizations that manage billions of dollars in assets. One of the most important features of the `PA` package is that it is able to compute dozens of risk and performance measures for a given returns series. One of the most important features of the `PA` package is that it is able to compute dozens of risk and performance measures for a given returns series. Our research results very considerably enhance the `PA` package greatly by providing reliable standard errors for the corresponding risk and performance measure estimates for serially correlated returns as well as for independent and identically distributed returns. We anticipate that this important new capability of the `PA` package will result in portfolio and risk managers to carry out better risk and performance analysis.

3.2 Future Work

One very important ongoing research we are working on is the computation of robust standard errors for robust regression estimators for serially correlated time series. Let Y_t and $\boldsymbol{X_t}$ $(p \times 1)$ be the observations and regressors at time t, the robust MM-regression parameters $\boldsymbol{\beta}$ $(p \times 1)$, the initial regression parameters $\boldsymbol{\beta_0}$ $(p \times 1)$ and the scale parameter σ are given by the solutions of the

equations

$$E\left[\rho'\left(\frac{Y_t - \boldsymbol{X}_t'\boldsymbol{\beta_0}}{\sigma}\right)\boldsymbol{X_t}\right] = 0, \tag{3.1}$$

$$E\left[\rho\left(\frac{Y_t - \boldsymbol{X}_t'\boldsymbol{\beta_0}}{\sigma}\right) - b\right] = 0, \tag{3.2}$$

$$E\left[\psi\left(\frac{Y_t - \boldsymbol{X}_t'\boldsymbol{\beta}}{\sigma}\right)\boldsymbol{X_t}\right] = 0. \tag{3.3}$$

The functions ψ and ρ are non-constant, scalar-valued and differentiable almost everywhere. Furthermore, ψ is odd, ρ is even and non-decreasing on $[0, \infty)$ with $\rho(0) = 0$ and b is a user-defined constant. The sample estimators for $\boldsymbol{\theta} = (\boldsymbol{\beta}, \boldsymbol{\beta_0}, \sigma)$ is $\hat{\boldsymbol{\theta}} = \left(\hat{\boldsymbol{\beta}}_{\boldsymbol{MM}}, \hat{\boldsymbol{\beta}}_{\boldsymbol{S}}, \hat{\sigma}_S\right)$, where

$$\frac{1}{T}\sum_{t=1}^{T}\rho'\left(\frac{Y_t - \boldsymbol{X}_t'\tilde{\boldsymbol{\beta}}_{\boldsymbol{S}}}{\tilde{\sigma}_S}\right)\boldsymbol{X_t} = 0, \tag{3.4}$$

$$\frac{1}{T}\sum_{t=1}^{T}\rho\left(\frac{Y_t - \boldsymbol{X}_t'\tilde{\boldsymbol{\beta}}_{\boldsymbol{S}}}{\tilde{\sigma}_S}\right) - b = 0, \tag{3.5}$$

$$\frac{1}{T}\sum_{t=1}^{T}\psi\left(\frac{Y_t - \boldsymbol{X}_t'\tilde{\boldsymbol{\beta}}_{\boldsymbol{MM}}}{\tilde{\sigma}_S}\right)\boldsymbol{X_t} = 0. \tag{3.6}$$

Croux et al. (2003) introduced a method to compute the standard errors of $\hat{\theta}$ based on Generalized Method of Moments, GMM for short, introduced by (Hansen et al. 1982), and heavily used in Econometrics research. Define the standardized residuals

$$r_t = \frac{y_t - \boldsymbol{X}_t'\tilde{\boldsymbol{\beta}}_{\boldsymbol{MM}}}{\tilde{\sigma}_S} \quad \text{and} \quad r_{0t} = \frac{y_t - \boldsymbol{X}_t'\tilde{\boldsymbol{\beta}}_{\boldsymbol{S}}}{\tilde{\sigma}_S}, \tag{3.7}$$

and shorthand notations

$$\psi_t = \psi(r_t), \quad \rho_t = \rho(r_t), \quad \text{and} \quad \rho_{0t} = \rho_0(r_{0t}). \tag{3.8}$$

The moment condition is given by

$$\boldsymbol{m_t}(\boldsymbol{\theta}) = \begin{pmatrix} \psi_t \boldsymbol{X_t} \\ \rho'_{0t} X_t \\ \rho_{0t} - b \end{pmatrix}. \tag{3.9}$$

The MM-Estimate are the solution to the equation

$$\frac{1}{T}\sum_{t=1}^{T} \boldsymbol{m_t}(\boldsymbol{\theta}) = 0. \tag{3.10}$$

(Hansen et al. 1982) established the asymptotic normality of $\hat{\theta}$. As $T \to \infty$,

$$\sqrt{T}\left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right) \to N(0, \boldsymbol{V_{GMM}}), \tag{3.11}$$

where

$$\boldsymbol{V_{GMM}} = \left(\boldsymbol{G}'\boldsymbol{\Omega}^{-1}\boldsymbol{G}\right)^{-1}, \tag{3.12}$$

$$\boldsymbol{G} = E\left[\frac{\partial \boldsymbol{m_t}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}'}\right], \tag{3.13}$$

$$\boldsymbol{\Omega} = \lim_{n\to\infty} E\left[\frac{1}{n}\sum_{t=1}^{n}\sum_{s=1}^{n} \boldsymbol{m_t}(\boldsymbol{\theta})\boldsymbol{m_s}(\boldsymbol{\theta})'\right]. \tag{3.14}$$

Substituting (3.9) into (3.13) and (3.14), the asymptotic variance of $\hat{\beta}_{MM}$ is given by

$$\begin{aligned} \mathbf{Avar}\left(\hat{\boldsymbol{\beta}}_{\boldsymbol{MM}}\right) =& \boldsymbol{A}\sum_{j=-\infty}^{\infty} E\left(\psi_t\psi_{t-j}\boldsymbol{X_t}\boldsymbol{X_{t-j}}'\right)\boldsymbol{A} - \boldsymbol{a}\sum_{j=-\infty}^{\infty} E\left(\rho_{0t}\psi_{t-j}\boldsymbol{X_{t-j}}'\right)\boldsymbol{A} \\ & - \boldsymbol{A}\sum_{j=-\infty}^{\infty} E\left(\psi_t\rho_{0,t-j}\boldsymbol{X_t}\right)\boldsymbol{a}' + \sum_{j=-\infty}^{\infty} E\left(\rho_{0t}\rho_{0,t-j} - b^2\right)\boldsymbol{a}\boldsymbol{a}', \end{aligned} \tag{3.15}$$

where

$$\boldsymbol{A} = \sigma \left[E \left(\psi_t' \boldsymbol{X_t} \boldsymbol{X_t}' \right) \right]^{-1} \quad \text{and} \quad \boldsymbol{a} = \boldsymbol{A} \frac{E \left(\psi_t' X_t r_t \right)}{E \left(\rho_{0t}' r_{0t} \right)}.$$

We apply the following rules to (3.15) to get $\mathbf{Avar}\left(\hat{\boldsymbol{\beta}}_{\boldsymbol{MM}}\right)$, the empirical estimate:

1. Replace $\boldsymbol{\theta} = (\boldsymbol{\beta}, \boldsymbol{\beta_0}, \sigma)$ with $\hat{\boldsymbol{\theta}} = \left(\hat{\boldsymbol{\beta}}_{\boldsymbol{MM}}, \hat{\boldsymbol{\beta}}_{\boldsymbol{S}}, \hat{\sigma}_S\right)$.
2. Replace $E(\cdot)$ by $T^{-1} \sum_{t=1}^{T} (\cdot)$ and put any term outside the observation window equal to zero.
3. Replace the infinite sum $\sum_{j=-\infty}^{\infty} (\cdot)$ by the truncated weighted sum $\sum_{j=-q}^{q} w_j (\cdot)$, using Bartlett weights $w_j = 1 - |j| / (q+1)$ and $q = q(T) \to \infty$ at a slow rate in T. This is essentially applying the method by Newey and West (1987).

Robust location estimators are special cases of robust regression where the regressor vector $\boldsymbol{X_t}$ is a real number 1 and the coefficient vector $\boldsymbol{\beta}$ is a scalar μ. Maronna et al. (2018) recommends using the optimal bias robust psi function given by

$$\psi_a(x) = \operatorname{sgn}(\mathrm{x}) \left(|\mathrm{x}| - \frac{\mathrm{a}}{\phi(\mathrm{x})} \right)^{+} = \begin{cases} x - \operatorname{sgn}(x) \dfrac{a}{\phi(x)} & |x| \in (\text{lower}, \text{upper}) \\ 0 & \text{otherwise} \end{cases}, \tag{3.16}$$

where sgn(x) is the sign function and a is a tuning constant that controls the support of $\psi_\alpha(x)$. lower and upper are constants determined by the value of the tuning constant a. The optimal bias robust psi function with 99% efficiency is plotted in Figure 3.1.

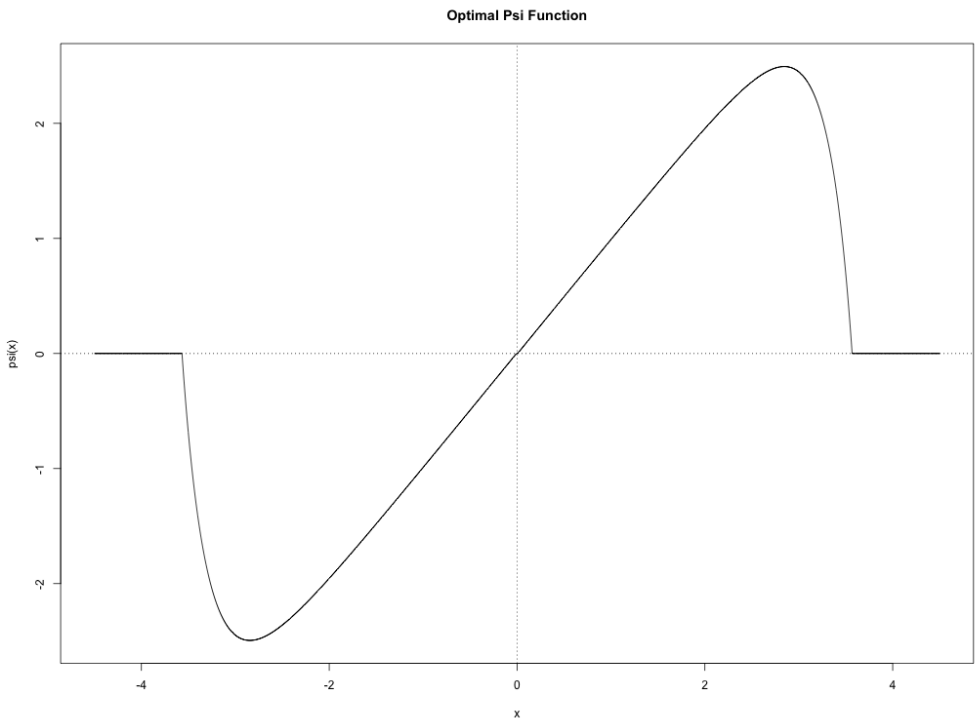

Figure 3.1: Optimal Bias Robust Psi Function with 99% Efficiency

Robust location estimators are particularly useful in analyzing the time series of regression slopes for the size factor (Green et al. 2017). The time series of regression slopes for the size factor using least squares regression (Size_LS) and robust regression (Size_Rob) from 1980 to 2015 are shown in Figure 3.2.

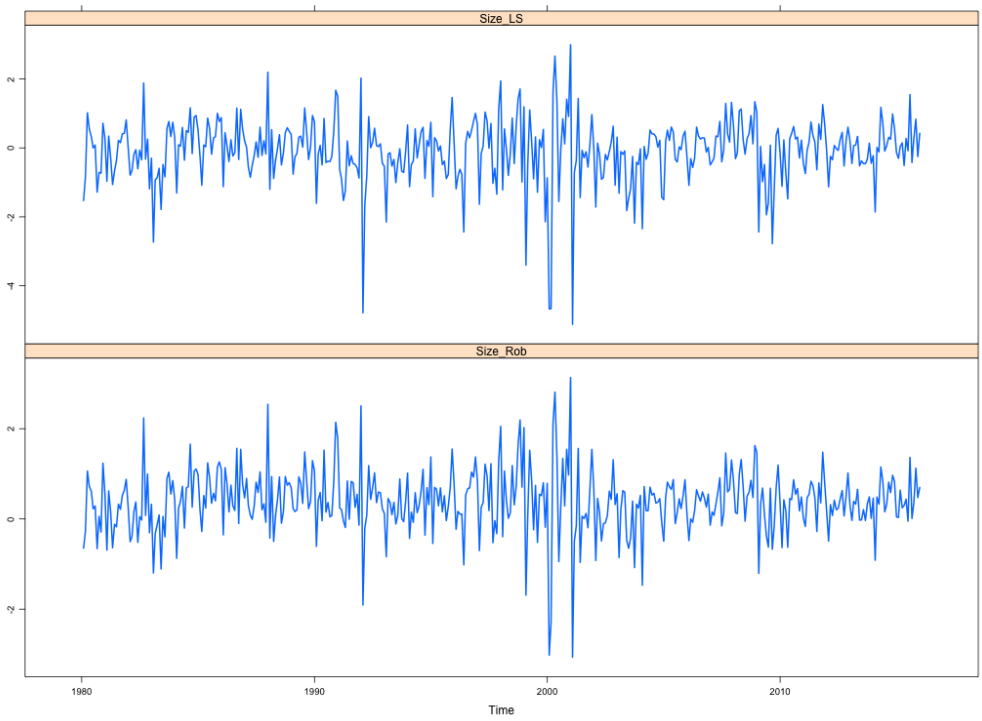

Figure 3.2: Time Series of the Coefficients for Size Factor using Robust Cross-Sectional Regression

We realized that the influence function approach we used in Chapter 1 results in the same asymptotic variance formula as Croux et al. (2003). Since the seCorIF method works better than the Newey–West method used in Croux et al. (2003), for future work, we are going to start with robust location estimate and conduct Monte Carlo simulations to see if our method works better. If the results are in our favor, we plan to move on to robust regression in the empirical asset pricing context.

Another potential research topic is developing an adaptive method (seCorIFAdapt) to compute the standard errors of risk and performance measure estimators depending on the value of the autoregression coefficient of the returns time series. As shown in Chapter 1, the seCorIF method works well when the autoregression coefficient is small while the seCorIFPW method works better when there is substantial serial correlation in the returns time series. The idea of the seCorIFAdapt method is to get an estimate of the autoregression coefficient first, then use seCorIF method if the

autoregression coefficient is below a certain threshold. Otherwise, the seCorIFPW method will be used. One important question to answer is whether or not the uncertainty in the AR1 parameter estimate adds too much to the variability of the adaptive method for it to be effective.

The glmGammaEN algorithm in Chapter 2 estimates the shape parameter and the coefficients for the linear component in two separate steps. It would be ideal to estimate these parameters in one step simultaneously. The challenge is that the minimum value of the elastic net regularization parameter that forces an all-zero solution for the linear coefficients depends on the shape parameter. We are going to explore the possibility of a two dimensional cross validations scheme. In this scheme, we first generate a sequence of possible values for the shape parameter. For each of these values, we can then compute the corresponding minimum value of the elastic net regularization parameter that forces an all-zero solution for the linear coefficients. Then we use cross validation two find the pair of shape parameter and elastic net regularization parameter that results in the smallest cross validation error. This pair will be considered the solution to the glmGammaEN optimization problem.

Bibliography

Andrews, D. W. K. (1991). “Heteroskedasticity and autocorrelation consistent covariance matrix estimation”. *Econometrica* 59.3, p. 817 (cit. on pp. 13, 26).

Andrews, D. W. K. and J. C. Monahan (1992). “An improved heteroskedasticity and autocorrelation consistent covariance matrix estimator”. *Econometrica* 60.4, pp. 953–966 (cit. on pp. 13, 26, 40).

Beck, A. and M. Teboulle (2009). “A fast iterative shrinkage-thresholding algorithm for linear inverse problems”. *SIAM Journal on Imaging Sciences* 2.1, pp. 183–202 (cit. on p. 88).

Boudt, K., B. Peterson and C. Croux (2008). “Estimation and decomposition of downside risk for portfolios with non-normal returns”. *The Journal of Risk* 11.2, pp. 79–103 (cit. on p. 57).

Brillinger, D. R. (2001). *Time Series: Data Analysis and Theory*. SIAM, Philadelphia (cit. on pp. 13, 30).

Campbell, J. Y., A. W.-C. Lo and A. C. MacKinlay (1997). *The Econometrics of Financial Markets*. Princeton University Press, Princeton, NJ (cit. on p. 13).

Cochrane, J. H. (2009). *Asset Pricing*. Princeton University Press, Princeton, NJ (cit. on p. 13).

Combettes, P. L. and J.-C. Pesquet (2011). “Proximal splitting methods in signal processing”. *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*. Ed. by H. Bauschke et al. Springer, New York, NY, pp. 185–212 (cit. on p. 87).

Cont, R., R. Deguest and G. Scandolo (2010). “Robustness and sensitivity analysis of risk measurement procedures”. *Quantitative Finance* 10.6, pp. 593–606 (cit. on p. 15).

Cooley, J. W. and J. W. Tukey (1965). "An algorithm for the machine calculation of complex fourier series". *Mathematics of Computation* 19.90, pp. 297–301 (cit. on p. 31).

Croux, C., G. Dhaene and D. Hoorelbeke (2003). *Robust standard errors for robust estimators* (cit. on pp. 106, 110).

Dobson, A. J. and A. Barnett (2008). *An Introduction to Generalized Linear Models*. CRC Press, Boca Raton, Florida (cit. on p. 33).

Filippova, A. (1962). "Mises' theorem on the asymptotic behavior of functionals of empirical distribution functions and its statistical applications". *Theory of Probability & Its Applications* 7.1, pp. 24–57 (cit. on p. 22).

Fox, J. (2016). *Applied Regression Analysis and Generalized Linear Models*. Sage Publications, Thousand Oaks, CA (cit. on pp. 33, 36).

Friedman, J., T. Hastie and R. Tibshirani (2010). "Regularization paths for generalized linear models via coordinate descent". *Journal of Statistical Software* 33.1, p. 1 (cit. on p. 92).

Genz, A. (2004). "Numerical computation of rectangular bivariate and trivariate normal and t probabilities". *Statistics and Computing* 14.3, pp. 251–260 (cit. on p. 28).

Green, C. G. and R. D. Martin (2017). "Fama-french 1992 redux with robust statistics". *Available at ssrn: https://ssrn.com/abstract=2963855* (cit. on p. 109).

Hampel, F. R. (1974). "The influence curve and its role in robust estimation". *Journal of the American Statistical Association* 69.346, pp. 383–393 (cit. on pp. 14, 15, 22).

Hampel, F. R., E. M. Ronchetti, P. Rousseeuw and W. A. Stahel (1986). *Robust Statistics: the Approach based on Influence Functions*. Wiley-Interscience, New York, NY (cit. on pp. 14, 15, 22, 23).

Hampel, F. R. (1968). *Contributions to the Theory of Robust Estimation*. PhD Thesis, University of California, Berkeley (cit. on p. 19).

Hansen, L. P. and K. J. Singleton (1982). "Generalized instrumental variables estimation of nonlinear rational expectations models". *Econometrica* 50.5, pp. 1269–1286 (cit. on pp. 106, 107).

Hastie, T., R. Tibshirani and J. Friedman (2009). *The Elements of Statistical Learning*. Springer Series in Statistics, New York, NY (cit. on p. 38).

Heidelberger, P. and P. D. Welch (1981). "A spectral method for confidence interval generation and run length control in simulations". *Communications of the ACM* 24.4, pp. 233–245 (cit. on pp. 13, 31, 44, 45, 55).

Hirukawa, M. (2010). "A two-stage plug-in bandwidth selection and its implementation for covariance estimation". *Econometric Theory* 26.03, pp. 710–743 (cit. on pp. 13, 26, 40).

Hoerl, A. E. and R. W. Kennard (1970). "Ridge regression: biased estimation for nonorthogonal problems". *Technometrics* 12.1, pp. 55–67 (cit. on p. 38).

Huber, P. J. and E. M. Ronchetti (2009). *Robust Statistics*. John Wiley & Sons, Inc., Hoboken, New Jersey (cit. on pp. 22, 23).

Jong, P. de and G. Z. Heller (2008). *Generalized Linear Models for Insurance Data*. Cambridge University Press, Cambridge, UK (cit. on p. 83).

Kleiner, B., R. Martin and D. Thomson (1979). "Robust estimates of spectra (with discussion)". *Journal of Royal Statistical Society: Series B* 41, pp. 313–351 (cit. on p. 57).

Lo, A. W. (2002). "The statistics of sharpe ratios". *Financial Analysts Journal* 58.4, pp. 36–52 (cit. on pp. 12, 13).

Maronna, R. A., R. D. Martin, V. J. Yohai and M. Salibián-Barrera (2018). *Robust Statistics: Theory and Methods (with R), 2nd Edition*. John Wiley & Sons, Inc., Hoboken, New Jersey (cit. on pp. 14–16, 19, 57, 108).

Martin, R. D. and S. Zhang (2017). "Nonparametric versus parametric expected shortfall (september 10, 2017)". *Available at ssrn: https://ssrn.com/abstract=2747179* (cit. on pp. 15, 18, 19, 23).

McCullagh, P. and J. A. Nelder (1989). *Generalized Linear Models*. CRC Press, Boca Raton, Florida (cit. on pp. 33, 36, 83).

McNeil, A. J., R. Frey and P. Embrechts (2015). *Quantitative Risk Management: Concepts, Techniques and Tools*. Princeton University Press, Princeton, NJ (cit. on p. 19).

Meyer, C. (2013). "Recursive numerical evaluation of the cumulative bivariate normal distribution". *Journal of Statistical Software* 52.i10 (cit. on p. 28).

Nesterov, Y. (2013). *Introductory Lectures on Convex Optimization: A Basic Course*. Springer Science & Business Media, New York, NY (cit. on p. 88).

Newey, W. K. and K. D. West (1987). "A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix". *Econometrica* 55.3, pp. 703–708 (cit. on pp. 12, 26, 59, 108).

– (1994). "Automatic lag selection in covariance matrix estimation". *The Review of Economic Studies* 61.4, pp. 631–653 (cit. on pp. 13, 26, 60).

Olshen, R. A. (1967). "Asymptotic properties of the periodogram of a discrete stationary process". *Journal of Applied Probability* 4.3, pp. 508–528 (cit. on p. 30).

Parikh, N., S. Boyd et al. (2014). "Proximal Algorithms". *Foundations and trends® in optimization* 1.3, pp. 127–239 (cit. on pp. 39, 84, 91).

Percival, D. B. and A. T. Walden (1993). *Spectral Analysis for Physical Applications*. Cambridge University Press, Cambridge, UK (cit. on pp. 31, 40).

Priestley, M. B. (1981). *Spectral Analysis and Time Series*. Academic Press, London, UK (cit. on p. 31).

Tibshirani, R. (1996). "Regression Shrinkage and Selection via the Lasso". *Journal of the Royal Statistical Society. Series B (Methodological)* 58.1, pp. 267–288 (cit. on p. 38).

Walker, A. (1965). "Some asymptotic results for the periodogram of a stationary time series". *Journal of the Australian Mathematical Society* 5.01, pp. 107–128 (cit. on p. 30).

Zhang, S. (2009). *Statistical analysis of portfolio risk and performance measures: the influence function approach*. PhD Thesis, University of Washington, Seattle, WA (cit. on p. 17).

Zou, H. and T. Hastie (2005). "Regularization and variable selection via the elastic net". *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67.2, pp. 301–320 (cit. on p. 14).