THE FLORIDA STATE UNIVERSITY

COLLEGE OF ARTS & SCIENCES

ANALYSIS OF JAMES-STEIN FOR THE LEADING EIGENVECTOR

By

SIMON RIBAS

A Thesis submitted to the
Department of Mathematics
in partial fulfillment of the requirements for graduation with
Honors in the Major

Degree Awarded:
Spring, 2023

The members of the Defense Committee approve the thesis of Simon Ribas defended on April 18, 2023.

Dr. Alec Kercheval
Thesis Director

Dr. Fred Huffer
Outside Committee Member

Dr. Bhargav Karamched
Committee Member

Contents

1 Introduction

1.1 Abstract

We examine the results of ”James Stein for the leading eigenvector” (JSE) [6], which corrects excess dispersion in the leading eigenvector of a factor-based covariance matrix estimated from a high dimension low sample size regime data set, on real market data. We start by replicating a one-factor and four-factor model simulation and create a testing environment such that empirical results can be obtained. While JSE demonstrated significant performance gains over its competitors in the simulation, the benefits of using JSE in real market conditions remain unclear due to the complexity of market dynamics, making it challenging to draw definitive conclusions from the empirical analysis.

1.2 Brief Discussion

In various fields, such as physical, social, and data sciences, it is common to estimate covariance matrices of large random vectors using limited sample sizes. However, the sampling error inherent in this process can lead to excess dispersion of the leading eigenvector [7]. To address this issue, Goldberg and Kercheval develop James-Stein for the leading eigenvector (JSE) and demonstrate that eigenvector bias can have a significant impact on variance-minimizing optimization in the high-dimensional, low-sample size regime, whereas bias in estimated eigenvalues has little effect. Their research is motivated by the field of quantitative finance, where noisy covariance matrices are utilized to construct portfolios with mean-variance optimization. However, this approach can result in over-weighting of securities whose volatilities and correlations with other securities are underestimated due to sampling errors, resulting in highly inefficient portfolios [7]. JSE significantly improves upon this, both in the context of a one-factor and multi-factor model, resulting in better estimates of minimum variance portfolios.

In finance, empirical testing is often used to evaluate the performance of investment strategies, pricing models, and risk management techniques. In this study, we conduct one-factor and four-factor numerical experiments similar to those of [6] and [8]. We then extend the simulations to estimate the covariance matrix of the largest 100 stocks in the S&P 500 using 50 observations of asset returns under a one-factor-based covariance matrix. This experiment's efficacy is based on established empirical evidence, which indicates that a single, positive (or market-like) factor is a major driver of returns and risks in equity markets. Furthermore, this factor plays a significant role in determining the composition of mean-variance optimized portfolios.

To test the JSE estimator's performance, we conduct both in-sample and out-of-sample tests. In the in-sample tests, we use portfolio weight forecasts provided by the optimization for the minimum variance portfolio. In-sample tests allow us to examine the portfolio variance over the period in which the portfolio was derived from. This variance should be relatively low due to the mean-variance optimization with a large number of assets. Here, the JSE estimator achieved higher variance than its counterparts showcasing its ability to correct for optimization bias.

In the out-of-sample tests, we examine the JSE estimator's ability to predict portfolio weights in periods not used in the original optimization. These tests should tell us which estimator is better at estimating the population covariance matrix. However, we find no significant improvement in the performance of the JSE estimator.

Finally, we conduct a bias test to measure the quality of the variance estimate that we think we made. The bias test shows that the JSE estimator outperforms its competitors. This tells us the the JSE estimator is more accurate at estimating the variance of our portfolio. Overall, while the JSE estimator showed significant improvement in simulation experiments, neither eigenvector nor eigenvalue correction provided substantial improvement in out-of-sample portfolio variance, which is the most relevant empirical evidence for its implementation in industry.

2 Modern Portfolio Theory

2.1 The Efficient Frontier

In 1952 Harry Markowitz introduced the paper, "Portfolio Selection", which was featured in *The Journal of Finance* [15]. His groundbreaking work lives on today as a foundation for portfolio optimization and is known as "Modern Portfolio Theory" (MPT) or, mean-variance analysis. The theory assumes that investors are risk averse, which implies that they prefer to take portfolios with lower risk if they offer the same expected return. Thus, an investor who takes on more risk should be rewarded with higher expected returns.

Markowitz's Theory begins by building a mathematical framework for assembling a portfolio of assets such that the expected return is maximized at any given level of risk. We can quantify any given stock's risk and reward through variance and expectation of return, respectively. In general the expected return of any portfolio is given by,

$$\mathrm{E}(R_p) = \sum_i w_i \mathrm{E}(R_i) = \mathbf{w}^\top R \tag{1}$$

where w_i is the weight of any individual stock in the portfolio and R_i is it's expected return. The portfolio variance can be found by,

$$w^\top \Sigma w \tag{2}$$

where Σ is the covariance matrix of the portfolios returns. A common theme among the investment community is diversification. MPT says that investors can reduce the variance of their portfolios by holding combinations of stocks that are not perfectly correlated. The idea behind diversification is to diversify away unsystematic risk, or risk inherent to a particular company.

To demonstrate, we can consider a multi-asset portfolio. For different percentages of allocation weight we will have inherently different portfolios, with some more risky than others. We can plot these portfolios on a graph with the portfolios standard deviation on the X-axis and the Expected Returns on the Y-axis. If we repeat this for many combinations of portfolios we will get something like this,

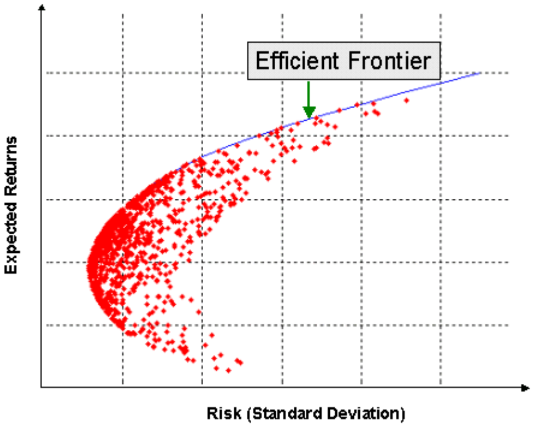


Figure 1: Sahajwani, Manish. "Efficient Frontier." financetrain, https://financetrain.com/constructing-an-efficient-frontier

We find that at each level of expected return, there is a portfolio that has a minimum standard deviation. These portfolios are the ones that an investor would choose based on their risk appetite, and make up what is known as the efficient frontier. Any portfolio inside the efficient frontier is said to be "inefficient." Naturally the risk averse investor would choose only portfolios on the efficient frontier. We are particularly interested in finding the minimum variance portfolio.

2.2 Mean-Variance Optimization

We have presented Modern Portfolio Theory and the efficient frontier. The efficient frontier poses new questions, such as how to find these portfolios. For our research we will be focusing on the minimum variance portfolio which has the least risk and lies furthest to the left on the efficient frontier. The process of finding optimal portfolios is known as mean-variance optimization [15].

Consider a universe of p financial securities. We can represent a portfolio by a vector whose i^{th} entry is the fraction or *weight* of the portfolio invested in security i [6]. Thus finding the minimum variance portfolio will consist of minimizing (2),

$$\min_{w \in \mathbb{R}^p} w^\top \hat{\Sigma} w \tag{3}$$

subject to $w^\top \mathbf{1} = 1$ where the $p \times p$ matrix $\hat{\Sigma}$ is a non-singular estimate of the unknown true security covariance matrix Σ. If the estimate $\hat{\Sigma}$ is derived from observed data, then $\hat{w}^*$ is a data-driven approximation of the true optimum w^*, defined as the solution to (3) with $\hat{\Sigma}$ replaced by Σ [6]. Naturally, the mean-variance optimization problem becomes one of covariance estimation.

Portfolio managers need to estimate the covariance matrices among asset returns in a high dimensional, low sample size regime (HL) to find optimal portfolios. However, when considering a portfolio with p assets and n observations such that $p >> n$, the sample covariance matrix will be of rank n since there are n linearly independent columns in our observation matrix. Thus the resulting covariance matrix is singular, or "ill-conditioned." Our sample covariance matrix does not serve as a covariance matrix estimator.

2.3 Factor Models

Factor models are used to reduce the dimension of a setting where many factors can be considered, but only a few dominate. The simplest factor model is the one factor model. We will use the one factor model to estimate covariance matrices. Consider the one factor model,

$$r = \beta f + \epsilon \tag{4}$$

where r is an observable p-vector, β is a p-vector of factor loadings, f is a common factor through which the observable variables are correlated, and ϵ is a p-vector of variable-specific effects that are uncorrelated with f and each other. We can use a factor-based covariance matrix to model the covariance structure of our model. Setting the factor variance to be σ^2 and the specific variance to be δ^2, the population covariance matrix for the one factor model becomes,

$$\Sigma = \sigma^2 \beta \beta^\top + \delta^2 I \tag{5}$$

We can write the factor loadings in terms of a scale factor and unit vector $\beta = |\beta| b$. Then our population covariance matrix becomes,

$$\Sigma = (\sigma^2 |\beta|^2) b b^\top + \delta^2 I. \tag{6}$$

Notice, the quantities σ^2 and $|\beta|^2$ cannot be determined from data, but we can make an effort to estimate their product, $\eta = \sigma^2 |\beta|^2$ [6]. So the problem of estimating the true covariance matrix Σ has been reduced to finding the estimators $\hat{b} \in \mathbb{R}^p$, and $\hat{\eta}, \hat{\delta}^2 \in \mathbb{R}$. Thus our estimated covariance matrix is now,

$$\hat{\Sigma} = \hat{\eta} \hat{b} \hat{b}^\top + \hat{\delta}^2 I \tag{7}$$

The idea is to use information about our sample covariance matrix $\mathbf{S}$, such as the leading eigenvector, to solve for the estimators. However, if we consider a four factor model

$$R = \mathbf{B}_* \psi + \varepsilon \tag{8}$$

where $\psi = (\psi_1, ..., \psi_4)$ is a vector of returns to the the factors, $\mathbf{B}_*$ is a $N \times 4$ matrix of exposures to the factors, and $\varepsilon = (\varepsilon_1, ..., \varepsilon_N)$ is the vector of heterogeneous specific returns, then the covariance matrix of R takes the form

$$\Sigma_* = \mathbf{B}_* \Omega \mathbf{B}_*^\top + \Delta \tag{9}$$

assuming all the ψ_k and ε_n are mean zero and pairwise uncorrelated. Here, Ω and Δ denote the covariance matrices of ψ and ε respectively. It is clear that under our assumptions, both Ω and Δ are diagonal. To estimate Σ_* we can simply use principal component analysis to extract the first four leading eigenvectors of $\mathbf{S}$. These principal components will make up $\mathbf{B}_*$ and Ω will have the market variances across the diagonal which differ for each covariance matrix estimation. Thus we will have to estimate the market variance and the specific returns variance, or Δ.

3 The Covariance Matrix

3.1 Principal Component Analysis

Before we go on to solve the estimators for (7) it is important to discuss why we are taking the leading eigenvector. As discussed in the previous section, we are estimating the true covariance matrix by using the one factor model (4). However, we have mentioned little of what it is that we are really doing by taking the eigenvectors of a covariance matrix.

Principal component analysis (PCA) is a technique for reducing the dimensionality of large datasets, increasing interpret-ability but at the same time minimizing information loss [11]. The problem with large data sets is that dimensions can grow fast while observations are fixed. PCA essentially comes up with a new set of fewer dimensions such that they preserve most of the structure in the data.

PCA reduces the dimension by looking for the direction of greatest variance. It continues this process by picking greatest variance orthogonal vectors. Picking these vectors preserves the structure of the data. It turns out, that we can find these new vectors by centering our data points (subtracting the mean), computing the covariance matrix, and finally extracting the eigenvalues and eigenvectors of the covariance matrix. These eigenvectors are the new dimensions. The vector that explains most of the variance in the data set will be the leading eigenvector which is associated with the largest eigenvalue.

In the case of our one factor model and covariance matrix estimation (7), we can use the leading eigenvector of $\mathbf{S}$ to preserve as much information possible of the sample covariance matrix. The leading eigenvector of $\mathbf{S}$ is known as the principal component of $\mathbf{S}$.

3.2 Covariance Matrix Estimation

3.2.1 One Factor Covariance Matrix Estimation

We will now continue with (7) to construct $\hat{\Sigma}$ in the HL regime under the one factor model. Much of this discussion is taken from [6]. The $p \times p$ sample covariance matrix $\mathbf{S}$ will have the spectral decomposition:

$$\mathbf{S} = \lambda^2 hh^\top + \lambda_2^2 v_2 v_2^\top + \lambda_3^2 v_3 v_3^\top \cdots + \lambda_p^2 v_p v_p^\top \tag{10}$$

in terms of the non-negative eigenvalues $\lambda^2 \geq \lambda_2^2 \geq \cdots \geq \lambda_p^2 \geq 0$ and orthonormal eigenvectors $\{h, v_2, ..., v_p\}$ of $\mathbf{S}$. Notice, λ^2 is the leading eigenvalue of our rank n covariance matrix $\mathbf{S}$. We can let ℓ^2 denote the average of the remaining non-zero eigenvalues of $\mathbf{S}$,

$$\ell^2 = \frac{tr(\mathbf{S}) - \lambda^2}{n-1} \tag{11}$$

where $tr(\mathbf{S})$ is the trace of $\mathbf{S}$ (sum of the eigenvalues). Under assumptions of Theorem 0.1 in reference [6], the Lemma A.2 of [7] provides the asymptotic relationships between eigenvalues of $\mathbf{S}$ and factor model parameters. For large p,

$$\lambda^2 \approx \frac{|\beta|^2|f|^2}{n} + \frac{p}{n}\delta^2 \tag{12}$$

where $f = (f_1, ..., f_n)$ is the vector of realizations of the common factor return corresponding to the n observations, and

$$\ell^2 \approx \frac{p}{n}\delta^2. \tag{13}$$

We can then approximate the trace of $\mathbf{S}$ in terms of our factor model:

$$tr(\mathbf{S}) \approx \frac{|\beta|^2|f|^2}{n} + p\delta^2. \tag{14}$$

Notice that we don't have access to f. However, $|f|^2/n$ is an unbiased estimator of the true factor variance σ^2 [6]. We can replace $|f|^2/n$ with σ^2 and apply it to formulas (12) and (13) to get:

$$\hat{\sigma}^2|\beta|^2 \approx \lambda^2 - \ell^2. \tag{15}$$

From (7) we see that $\hat{\eta} = \lambda^2 - \ell^2$ and $\hat{\delta}^2 = (n/p)\ell^2$. Plugging back in we get,

$$\hat{\Sigma}(\hat{b}) = (\lambda^2 - \ell^2)\hat{b}\hat{b}^\top + (n/p)\ell^2 I. \tag{16}$$

What remains is to choose $\hat{b}$. The simple solution is to substitute h from (10), the leading eigenvector of $\mathbf{S}$. Although, we consider alternatives to the naive estimate later on in the paper.

3.2.2 Four Factor Covariance Matrix Estimation

Estimating a four factor covariance matrix is similar yet different from the one factor case. If we wish to construct a factor based covariance matrix. The fist step to constructing (9) is to take the first four leading eigenvectors and eigenvalues of **S** the sample covariance matrix. We denote λ^2 to be the largest eigenvalue of **S**. The estimated specific returns **Z** are the residuals to the regression of the security returns onto the factors. Consequently, the specific variance of security n is estimated to be

$$\delta_n^2 = \frac{1}{T} \sum_{t=1}^{T} \mathbf{Z}_{nt}^2 \tag{17}$$

We must also provide an estimate for σ^2, or the market variance given by

$$\sigma^2 = \lambda^2 - \delta^2 \tag{18}$$

where δ^2 is the average of the specific variances. Based on [8], when the returns are believed to follow the Gaussian distribution, we can improve δ^2 and σ^2 in (17) and (18) by using a Marchenko-Pastur correction such that δ^2 becomes

$$\delta_{mp}^2 = \frac{tr(\mathbf{S}) - (\lambda^2 + ... + \lambda_4^2)}{N - K(1 - N/T)} \tag{19}$$

where λ_i^2 is the i^{th} largest eigenvalue of **S**. We then let σ^2 become

$$\sigma_{mp}^2 = \lambda^2 - \delta_{mp}^2(1 + N/T) \tag{20}$$

using the Marchenko-Pastur (MP) corrected delta. The source suggests using these adjusted values for σ^2 and δ^2 in practice, even if the returns are not believed to be Gaussian.

3.2.3 Marchenko Pastur Correction

The Marchenko-Pastur (MP) correction is a statistical method used to estimate specific and market variances in finance. It is named after the mathematicians Vladimir Marchenko and Leonid Pastur, who first proposed the method in their paper ”Distribution of eigenvalues for some sets of random matrices” in 1967 [14].

The MP correction is based on the Marchenko-Pastur distribution, which describes the distribution of eigenvalues of a sample covariance matrix when the population covariance matrix is a low-rank matrix plus a diagonal matrix. The MP distribution has a bell-shaped form, with most of the eigenvalues concentrated around the population eigenvalues, and a tail that accounts for the noise [1]. By comparing the empirical distribution of the eigenvalues of the estimated covariance matrix with the MP distribution, one can identify the noisy eigenvalues and remove them from the analysis. This correction results in a more accurate estimate of the covariance matrix and its eigenvalues, allowing for more reliable applications of techniques such as PCA.

4 Previous Work

4.1 The Dispersion Bias

Although h is a solution to $\hat{b}$, there are flaws in this estimate due to sampling error. Sampling error has been a long time issue for investors since Markowitz introduced Modern Portfolio Theory. In the context of of Markowitz portfolios, the impact of eigenvalue bias and optimal corrections are investigated in [5][12]. Much of the work which aims to reduce sampling error, such as [4], considers eigenvalue corrections in "spiked" covariance matrices, which are similar to the covariance matrices we consider in the HL regime.

Although eigenvector bias is acknowledged, direct bias corrections are made only to the eigenvalues corresponding to the principal components in the HL regime [18]. Other approaches alter the sample eigenvectors such as reference [13] where sample covariance matrices are shrunk toward a structured covariance matrix. However, these approaches are not focused on characterizing the bias inherent to the sample eigenvectors themselves [7]. The sampling error inherent to the HL regime leads to excess dispersion in the leading eigenvector where dispersion is given by:

$$d^2(h) = \frac{1}{p}\sum_{i=1}^{p}\left(\frac{h_i - m(h)}{m(h)}\right)^2. \tag{21}$$

Implementing a one factor model helps to mitigate the impact of sampling error on an estimated covariance matrix by reducing the number of required parameters. This embedded sampling error tricks (3) into constructing distorted and highly inefficient portfolios [7]. In fact, simulation in a one-factor PCA model, such as (4), reveals that errors in security weights and risk forecasts of the minimum variance portfolio are driven by errors in the leading eigenvector and not in its associated eigenvalue (variance) [7].

To gain some intuitive understanding of why there might be excess dispersion in the leading eigenvector, we present an example from [7]. We can consider a market where correlations are driven by a single factor with the assumption that all security exposures to that factor are identical. Then with high probability, a PCA estimate of the leading factor will have higher dispersion of its entries. By decreasing the dispersion we can mitigate the estimation error.

Section 3 of [7] demonstrates that it is the excess dispersion in the leading eigenvector of a sample covariance matrix that must be removed to address the impact of the optimization bias. Goldberg and Kercheval show both theoretically and with numerical experiments, that, for our mean-variance optimization problem (3), efforts to correct eigenvalues have little value in comparison to correction of the leading eigenvector through an applied statistical technique: James Stein for eigenvectors (JSE) [6]. Thus we will obtain a better estimate for $\hat{b}$, improving upon h, by correcting dispersion bias through the James Stein estimator.

4.2 James Stein

For background on how we will reduce the dispersion bias on the leading eigenvector of $\mathbf{S}$, we introduce the James Stein estimator. Suppose we have a set of data such that it follows a distribution of some unknown mean μ_1 with variance 1. Randomly picking a sample z_1 would be the best estimate of μ_1 given that most of the data in a normal distribution is centered around its mean. If we play the same game with two sets of data that are completely unrelated to each other but follow a normal distribution, then again z_1 and z_2 would be the best estimators of μ_1 and μ_2 respectively. However, suppose there are $\mu = (\mu_1, \mu_2, ..., \mu_p)$ means to be estimated such that $p > 3$. From the discovery of Stein [16] and James & Stein [9] we learn that there are better ways to estimate μ. James and Stein obtain better estimates by shrinking the sample averages toward their collectives averages.

Let $m(z) = \sum_{i=1}^{p} z_i/p$ denote the collective average, and $\mathbf{1} = (1, 1, ..., 1)$, the p-dimensional vector of 1s. James and Stein define:

$$\hat{\mu}^{JS} = m(z)\mathbf{1} + c^{JS}(z - m(z)\mathbf{1}). \tag{22}$$

The shrinkage constant c^{JS} is given by

$$c^{JS} = 1 - \frac{v^2}{s^2(z)}, \tag{23}$$

where

$$s^2(z) = \frac{1}{p-3}\sum_{i=1}^{p}(z_i - m(z))^2 \tag{24}$$

is a measure of the variation of the sample averages z_i around their collective average $m(z)$, and v^2 is an estimate of the conditional variance of each sample average around its unknown mean [6]. James and Stein showed that their estimator, $\hat{\mu}^{JS}$, dominates z by the measure of expected mean squared error (MSE):

$$E_{\mu,v}[|\hat{\mu}^{JS} - \mu|^2] < E_{\mu,v}[|z - \mu|^2] \tag{25}$$

For any fixe μ and v, the conditional expected mean squared error is imporved when using $\hat{\mu}^{JS}$ instead of z. Essentially, James and Stein proposed a technique that has been shown to improve estimates in terms of MSE by introducing a shrinkage constant. This is a classical case of the bias-variance trade off, since MSE also takes the form of

$$MSE(\hat{\mu}) = Var(\hat{\mu}) + Bias^2(\hat{\mu}). \tag{26}$$

Once again, consider our original estimate z_1. Since z_1 is a random sample it will have a distribution with mean μ_1. Thus the expected value of z_1 is μ_1 which implies the bias is 0. However, it's variance turns out to be quite large. Thus our estimation is prone to having lots of error. What we have done with the James Stein estimator is to introduce a bias such that the variance of our estimates is dramatically reduced. By definition the mean squared error of the estimate is also reduced.

4.3 James Stein for the Leading Eigenvector

As presented by Goldberg, Papanicolaou, and Shkolnik the James Stein estimator can be extended to eigenvectors. From [6] we see that the leading eigenvector becomes,

$$h^{\text{JSE}} = m(h)\mathbf{1} + c^{\text{JSE}}(h - m(h)\mathbf{1}). \tag{27}$$

The estimator h^{JSE} improves on h by reducing excess dispersion given by (21). As with regular James Stein, James Stein for eigenvectors calls for a lot of shrinkage when the average of the non-zero smaller eigenvalues dominates the variation of the entries of λh around their average, and only a little shrinkage when the reverse it true [6].

To numerically illustrate the results of h^{JSE}, Goldberg and Kercheval consider the problem of estimating a covariance matrix of stock returns from a year's worth of daily observations for an index like the S&P 500. They examine a hypothetical market driven by the one-factor model (4) with covariance matrix (6). Goldberg and Kercheval use covariance matrix estimators that preserve the trace $tr(\mathbf{S})$ since it is an unbiased estimator of the sum $tr(\Sigma)$ of the population variances. The diagonal entries of $\mathbf{S}$ are sample variances of individual assets, which are unbiased estimators of the diagonal entries of the true covariance matrix Σ. This means that adding up the eigenvalues of $\mathbf{S}$ provides an unbiased estimate of the sum of Σ's eigenvalues. Therefore, the authors use three data-driven, trace-preserving estimators:

$$\Sigma_{\text{raw}} = (\lambda^2 - \frac{n-1}{p-1}\ell^2)hh^\top + \frac{n-1}{p-1}\ell^2 I \tag{28}$$

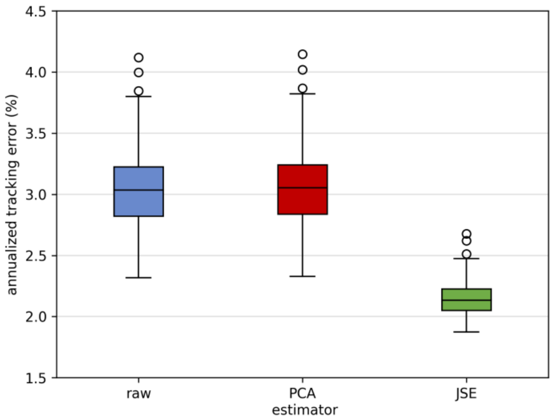
$$\Sigma_{\text{PCA}} = (\lambda^2 - \ell^2)hh^\top + (n/p)\ell^2 I \tag{29}$$

$$\Sigma_{\text{JSE}} = (\lambda^2 - \ell^2)h^{\text{JSE}}(h^{\text{JSE}})^\top + (n/p)\ell^2 I. \tag{30}$$

These estimators are different in that Σ_{raw} matches the leading eigenvalue and eigenvector of $\mathbf{S}$ without correction, Σ_{PCA} has the corrected leading eigenvalue, and Σ_{JSE} improves further by substituting h^{JSE} of (27) for h. Goldberg and Kercheval conduct these numerical experiments under factor model parameters taken from [7] and [8] which attempt to replicate the behavior of the financial markets. From here they draw factor and specific returns f and ϵ independently with mean 0 and standard deviations 16% and 60%, respectively. Entries of β are inspired by market betas and are drawn independently from a normal distribution with mean 1 and variance of 0.25.

To compare (28-30) they examine the tracking error, variance forecast ratio, and true variance forecast ratio for the estimators after 400 simulations. These performance metrics and results are elaborately illustrated in [6]. In summary, (30) outperformed (28) and (29) by all metrics. In fact, correcting the eigenvalue had little to no improvement from the raw estimate, whereas it is clear in [6] that Σ_{JSE} dominates in all categories.

For demonstration purposes, we borrow the tracking error figure in [6]. Tracking error is commonly used by portfolio managers to measure the width of the distribution of the difference in return of two portfolios, typically a portfolio and its benchmark [6]. Goldberg and Kercheval utilize this industry measure to perform an analysis on estimated and true covariance matrices which are synonymous to portfolios and their benchmark.



An annualized tracking error closer to 0% is optimal. Notice how the whiskers of the histograms are compressed in JSE, lowering the total variance of the annualized tracking error increasing accuracy and precision in the results.

4.4 Better Betas

In a separate research article entitled ”Better Betas” predating the JSE paper, Goldberg, Papanicolaou, Shkolnik, and Ulucam propose a data-centric modification for estimated betas, resulting in significant enhancements in the precision and allocation of minimum variance portfolios. Their main focus is PCA-betas, which are exposures to the dominant factor. They mention that the dominant PCA factor in US equities is market-like, meaning that it has relatively large variance and mostly positive exposures [8]. Thus ”Better Betas” looks at new ways of approaching beta estimation through the lens of shrinkage. The word shrinkage refers to the fact that β^{adj} is simply the β^{raw} estimate shrunk towards its mean. For reference, an adjusted beta formula could look like the following:

$$\beta^{\text{adj}} = c\beta^{\text{raw}} + (1 - c) \tag{31}$$

where the parameter c lies between zero and one. ”Better Betas” compares the Blume adjustment which sets c to $2/3$ for β^{adj} to become

$$\beta^{\text{Blume}} = \frac{2}{3}\beta^{\text{raw}} + \frac{1}{3}. \tag{32}$$

The financial industry widely implements the Blume adjustment for betas, which accounts for the tendency of true betas to revert to one due to economic reasons. The study employs the Blume 2/3 as a benchmark in the financial sector to compare against their own beta estimations: the ”Raw Beta” and the ”GPS-adjusted Beta”. Similarly to their later work on ”James Stein for the Leading Eigenvector”, they employ numerical simulations to compute tracking error and variance forecast ratios as performance metrics for their minimum variance portfolio estimates. However, what sets their research apart is their factor model of returns. They not only consider a four-factor model of returns, but also test the effectiveness of their approach under ”stressed” and ”calm” market conditions to examine how different beta estimations adapt to varying volatility regimes. By changing the level of market volatility, the authors provide a more comprehensive understanding of how various beta estimations respond to different market conditions. They find that their GPS adjustment mitigates estimation error on minimum variance portfolios, absent knowledge of true betas, in both stressed and calm market conditions [8].

5 Methodology

5.1 Factor Simulations

5.1.1 One Factor

We consider the same problem presented by Goldberg and Kercheval in section **4.1**. Similarly the experiments are based on $p = 500$ securities and $n = 252$ trading days. Since $p > n$, this problem falls under the high dimension low sample size regime. We then construct a library in python which simulates a hypothetical market driven by the one-factor model (4) with covariance matrix (6) that allows us to evaluate the impact of shrinkage on the leading eigenvector. This library allows us to draw factor and specific returns f and ϵ independently with mean 0 and standard deviations 16% and 60%, respectively. Like in [6], factor returns are normal, and specific returns are drawn from a t-distribution with 5 degrees of freedom. [6] uses the fat tail t-distribution to illustrate that the results do not require Gaussian assumptions. We then compare the effect of eigenvalue vs. eigenvector correction with the portfolio performance metrics described below. The use of simulation allows us to easily compare estimates with actual values, but the accuracy and credibility of the simulation results depend on the quality of the underlying model. To ensure accuracy in financial market calibration, we adhere to [6] which takes from [8] and [7].

We simulate 252 days of independent daily returns for 500 securities for each experiment. This is achieved by generating calibrated returns into a large matrix, resulting in a returns matrix of size 500 x 252. A total of 400 such matrices will be generated for the 400 experiments, providing a comprehensive data set. From here we can use each of these return matrices to calculate a sample covariance matrix $\mathbf{S}$ from which we construct estimators (28), (29), and (30) by following the outline in section **3.2.1**. Finally by solving (3) we can conduct the performance metrics described in section **5.1.3**.

5.1.2 Four Factor

Aside from a one factor model, can the James Stein for Eigenvectors shrinkage help estimate population covariance matrices in multi-factor models? We try to answer this question by following the outline of the "Better Betas" paper where Goldberg, Papanicolaou, Shkolnik, and Ulucam consider a four factor model of returns and shrink the leading eigenvector of the sample covairance matrix in various ways. We consider testing the same estimators as in section **5.1.1** and utilize the same performance metrics. The problem will again be the same as in section **5.1.1** where we must estimate the population covariance matrix of asset returns based off $p = 500$ securities and $n = 252$ trading days. We then construct a library in python which simulates a hypothetical market driven by the four-factor model (8) with covariance matrix (9) that allows us to evaluate the impact of shrinkage on the leading eigenvector.

We assume that the returns R are Gaussian and calibrate our model to the calm regime in [8]. Since the dominant factor is market-like, or the first column of $\mathbf{B}_*$, it can be thought of as the vector of market betas. Since market betas are distributed around one, we set the beta-mean of the the market-beta vector to that value. We keep the same beta dispersion as in the one factor case with a standard deviation of 0.5. The three remaining factors are modeled on equity styles such as volatility, earnings yield and size [10]. We draw the exposures of each security, the next three vectors in $\mathbf{B}_*$, from a normal distribution and standardize to a z-score with mean 0, and variance 1. We follow [8] which takes from [2] providing guidance on the volatility of equity style factors, with estimates typically less than 10% per year.

After building the factor-model simulator we must go through process of constructing Σ_{raw}, Σ_{PCA}, and Σ_{JSE} for the four factor case. To construct the four factor based matrices we can reference the outline of section **3.2.2**. The only changes in our estimates will be on the adjustments made to the first column of $\mathbf{B}_*$ in (9). But since the original eigenvalue correction is now replaced by the Marchenko-Pastur adjustments, where $\delta^2_{mp} = \frac{tr(\mathbf{S}-(\lambda^2+...+\lambda_4^2)}{N-K(1-N/T)}$ and $\sigma^2_{mp} = \lambda^2 - \delta^2_{mp}(1 + N/T)$, Σ_{raw} will have no eigenvalue correction and our PCA estimate will differ by having δ^2 and σ^2 estimated using Marchenko-Pastur. When the returns are believed (or assumed) to follow the Gaussian distribution, we can improve δ^2 and σ^2 with this correction.

Similar to the one-factor case, we conduct 400 experiments where we simulate a year's worth of independent daily returns. From this data set we get the sample covariance matrix $\mathbf{S}$, construct our factor-based covariance matrix estimates, solve (3), and compare with the performance metrics below.

5.1.3 Performance Metrics

The study examines three distinct performance metrics that provide insight into different aspects of the impact of covariance matrix estimation error on optimization for both the one and four factor simulations. One of these metrics is the variance forecast ratio (VFR), which calculates the estimated variance of a linear combination of random variables against the true variance.

$$\mathrm{VFR}(\hat{w}^*) = \frac{\hat{w}^{*\top}\hat{\Sigma}\hat{w}^*}{\hat{w}^{*\top}\Sigma\hat{w}^*} \tag{33}$$

Stein introduced this metric in 1956 [17] for arbitrary combinations, but when applied to optimized quantities such as a minimum variance portfolio, it can be substantially lower than the maximum value of 1. This is because variance-minimizing optimization tends to overweight securities with underforecast variances and correlations with other securities, leading to an overestimate of risk. Bianchi et al. [3] use the VFR to assess risk under-forecasting in optimized portfolios. However, by utilizing additional metrics, we are able to evaluate the accuracy of optimized portfolios themselves, rather than just their risk forecasts.

In contrast to the VFR, the true variance ratio (TVR) is only applicable to optimized combinations of random variables. The TVR is the ratio of the true variance of the true optimum to the true variance of the estimated optimum, measuring the excess variance in the latter.

$$\mathrm{TVR}(\hat{w}^*) = \frac{w^{*\top}\Sigma w^*}{\hat{w}^{*\top}\Sigma\hat{w}^*} \tag{34}$$

To assess the accuracy of an optimized quantity, we use tracking error as a direct measure. Specifically, we define tracking error as

$$\mathrm{TE}^2(\hat{w}^*) = (\hat{w}^* - w^*)^\top\Sigma(\hat{w}^* - w^*) \tag{35}$$

for the minimum variance portfolio, which is widely adopted by portfolio managers to determine the difference in return between two portfolios and their distance from a benchmark. As these performance metrics depend on the true covariance matrix 6, they cannot be applied directly in an empirical study. However, in out-of-sample empirical tests, we can approximate the denominator of VFR, which is the true variance of the optimized quantity.

5.2 Empirical Methodology

5.2.1 Data and Data Collection

For all empirical tests below, data was gathered in the same fashion. For our market portfolio, we used the top 100 stocks by market cap of the SP-500 stocks at the end of 2022. This list was then used by the Yahoo Finance API to gather stock data accordingly. It should be mentioned that for the charts below, we remove times of market turbulence from 01-01-2020 to 09-01-2020 due to the large distortion it has on visualization of results.

5.2.2 In-Sample Testing

The objective of the in-sample tests is to assess the performance of the minimum variance portfolios that we solved for using $\Sigma_{\text{raw}}, \Sigma_{\text{JSE}}, \Sigma_{\text{PCA}}$ during the time period that we used to calculate their respective covariance matrices. Specifically, we want to evaluate how much variance each portfolio has during this time period. We then annualize the variance with respect to the return type we use in the specific experiment. We anticipate that the minimum variance portfolios derived from PCA and RAW estimates would exhibit lower in-sample variance than the JSE-based portfolio. This is because the former estimates are prone to high optimization biases, potentially leading to over-fitting of the data and consequent out-performance relative to the JSE estimator.

To begin the in-sample test, we select a time period over which we would like to conduct the test. For example, if we choose a 10-year period, this would result in 120 monthly returns per asset, resulting in a returns table of size (120 x 100). In Python, we can obtain this data frame by using the Yahoo Finance API and calling the history function. By specifying the period as '10y', the return type as '1mo', and indicating that we want the close price, we can calculate the percent change to obtain our monthly returns table. Since we want a sample covariance matrix of dimension 100 (assets) x 50 (months), we can start with the first 50 months of our returns table, now size (119x100) as the first row is deleted (since it contains a NaN value from the .pct_change() function). Using Python's built-in ".cov" function, we can compute the sample covariance matrix for the first 50 months, and extract the principal component and eigenvalues using Python's linear algebra library. With this information, we can construct equations (28), (29), and (30) and solve equation (3) with the covariance matrix estimates to obtain their respective minimum variance portfolio forecasts. Now that we have obtained w_{JSE}, w_{PCA}, and w_{raw}, we can finally compute equation (2) replaced with the sample covariance matrix to get the in-sample variances for JSE:

$$w_{\text{JSE}}^{\top} \mathbf{S} w_{\text{JSE}} \tag{36}$$

To get the in-sample variance for PCA and RAW estimates we simply replace the portfolio weights in (36).

The above description outlines the in-sample variance for the first time period. To compute a time series of annualized in-sample variances, we can repeat the procedure while shifting the time frame we select. In the example above, we chose the first 50 months of the returns table size (119x100), or the Python slice [0:49]. We can keep this slice size constant and increase it by one for our next in-sample test, or [1:50]. The next slices, or iterations, would be [2:51], [3:52], ..., [70:119], resulting in a total of 70 in-sample variances across time. We can plot the annualized variance versus the last month used to compute the weights.

5.2.3 Out of Sample Testing

The primary objective of this test is to assess the performance of the portfolios in out-of-sample data. The desired outcome is to observe a lower portfolio annualized variance for the JSE estimator compared to the PCA and RAW estimators, indicating superior performance.

Similar to the in-sample tests, we begin by selecting a time period for which we would like to conduct this test on. The same returns table is constructed by utilizing the Yahoo Finance API. From this returns table we can start constructing our covariance matrices with the same technique described in section **5.2.2**. However, since we are now conducting out of sample tests, there is one big change to how this test is done. For every iteration or step that we calculate our weights on, we must let the portfolio perform for a few out-of-sample iterations. These iterations are out of sample because we are now calculating the portfolios variance outside of the 50 observations used to calculate the covariance matrix. The number of iterations that we let our portfolio run out-of-sample will be called the window size. The window size is specified under the captions in the results section. For example, say the window size was window = 6. Now consider the first iteration where we calculate the covariance matrix from the returns table slice [0:50]. From here we can solve for the weights and collect the portfolio returns on the window [50:56] which gives a total of 6 returns. We then take the variance and continue the process for [1:51], [2:52], and so on. This will give us a time-series of out-of-sample variances.

The window sizes chose in section **6.3** come from experimental results. The goal was to choose the smallest window possible while having enough observations such that portfolio variance is not too large. However, one must consider realistic scenarios such as portfolio rebalancing. This is why the "10y/1mo" portfolios only perform for a maximum of 6 out-of-sample iterations (6 months), while the "5y/1wk" and "2y/1d" are allowed to perform for 12 iterations out of sample.

5.2.4 Bias Test

Consider a sequence of out of sample demeaned portfolio returns $x^{(1)}, x^{(2)}, ..., x^{(n)}$. Then suppose that we have the correct estimator for the standard deviation in the forecast, i.e. we have solved

$$w_{\text{true}} = \arg\min_{w} \sqrt{w^T \Sigma w} \tag{37}$$

subject to $w^T \mathbf{e} = 1$ using the true covariance matrix of asset returns, or Σ. Now consider the following sequence of random variables:

$$\frac{x^{(1)}}{\sigma_{\text{true}}}, \frac{x^{(2)}}{\sigma_{\text{true}}}, ..., \frac{x^{(n)}}{\sigma_{\text{true}}} \tag{38}$$

where σ_{true} is true standard deviation of the minimum variance portfolio. Although we cannot achieve this sequence with historical data, if we could, then we would essentially be drawing returns from a distribution with that specific standard deviation (true min var portfolio std). So if we were to take a large enough subset of (38), assuming n is sufficiently large, the standard deviation of this subset is going to be approximately 1.

Now although we do not have Σ, and thus can't solve for w_{true}, we can use this normalization technique to decide which $\hat{\Sigma}$ is closest to the true covariance matrix. This would lead us to arriving at weights which are closest to w_{true}. Now with a better solution of w, the closer the standard deviation of a subset of (38) will be to 1. However, since we rely on data, the process will be different. Specifically (38) will become:

$$\frac{x_{\text{JSE}}^{(1)}}{\sigma_{\text{JSE}}^{(1)}}, \frac{x_{\text{JSE}}^{(2)}}{\sigma_{\text{JSE}}^{(2)}}, ..., \frac{x_{\text{JSE}}^{(n)}}{\sigma_{\text{JSE}}^{(n)}} \tag{39}$$

Each $x^{(i)}$, which are out of sample returns, will have to be computed by some portfolio. Recall the table of returns, if we take a slice the size of our observations [0:49] and compute our weights, then $x^{(1)}$ is the return of this portfolio at time 50. $\sigma^{(1)}$ would just be the in-sample standard deviation, or $\sqrt{w^\top \Sigma w}$ where w is the weights computed from [0:49]. $x^{(2)}$ is computed by taking the slice [1:50], computing the weights, and getting the return of the weights at time 51. $\sigma^{(2)}$ is computed similarly. We continuously do this for as long as our returns table has space. We can view each element of this sequence as a normalized random variable whose standard deviation *should* be one. Thus we expect the standard deviation of the sequence (39) to be closer to 1 for JSE indicating the JSE estimator helps shrink the leading eigenvector of estimated covariance matrix of to that of the population.

To plot a time-series of the bias test we capture a large sequence of these random variables and utilize a sliding window technique to take standard deviations accordingly. The sliding window technique works similar to the way we iterated across the returns table for the in-sample test. Take equation (39) for instance and say it was of size 100. A sliding window technique of subset size 20 would take slices [0:19], [1:20], ..., [80:100] and take the standard deviations of each of the subsets. We could then plot these standard deviations against time. In order to identify how large the sliding window must be, we take accumulated standard deviations of (39) to see the speed of convergence towards one. Based on experimental results for different return types, a subset of 20 for monthly and 100 for weekly and daily best fits the data. We then plot these standard deviations as a function of time where the standard deviation to time is plotted from the last element in the sliding window similar to moving averages. These calculations are also made for JSE, PCA, and RAW covariance matrices.

6 Results and Analysis

6.1 One Factor Simulation

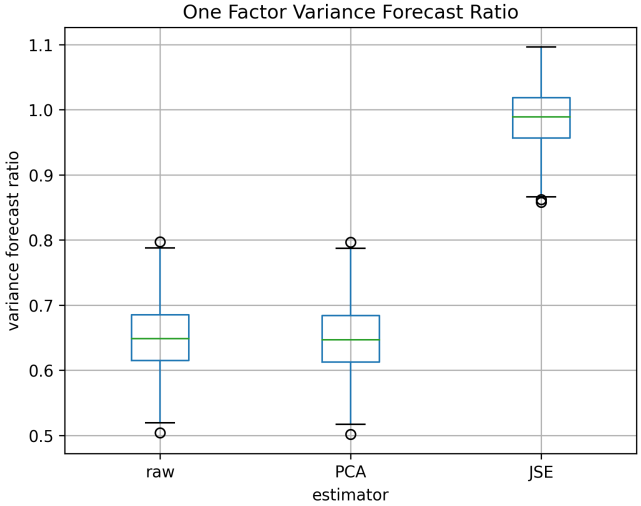


Figure 2: Variance forecast ratio for 400 experiments under the one factor model of returns. A perfect variance forecast ratio is equal to 1.

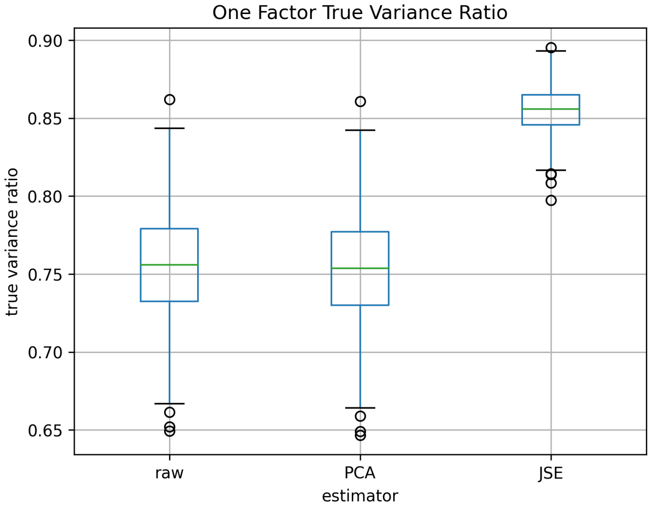


Figure 3: True variance ratio for 400 experiments under the one factor model of returns. A perfect true variance ratio is equal to 1

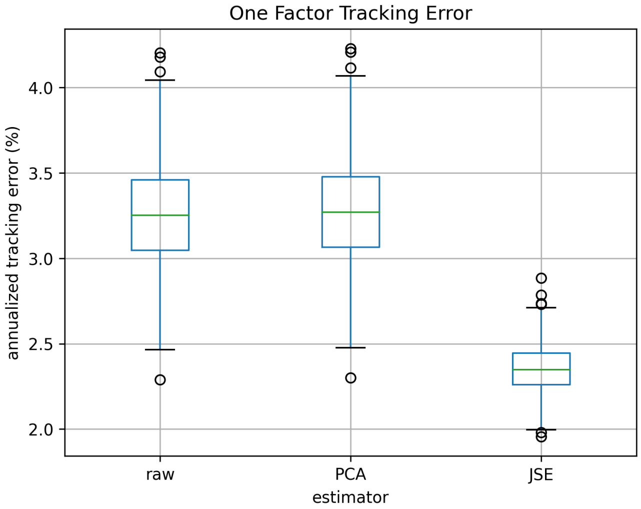


Figure 4: Tracking Error for 400 experiments under the one factor model of returns. A perfect tracking error is 0%.

6.1.1 Analysis of One Factor Simulation

In this study, we evaluate the impact of eigenvalue and eigenvector correction on the performance metrics of our portfolio under the one factor model of returns. We conducted 400 experiments using fixed values of $p = 500$ and $n = 252$, and analyzed the tracking error, variance forecast ratio, and true variance ratio for three different estimators: Σ_{raw}, Σ_{PCA}, and Σ_{JSE}. Our findings are summarized in figures 2-4. Correcting the leading eigenvalue in the PCA estimator had little to no effect, while shrinking the leading eigenvector with JSE led to significant improvement across all performance metrics.

6.2 Four Factor Simulation

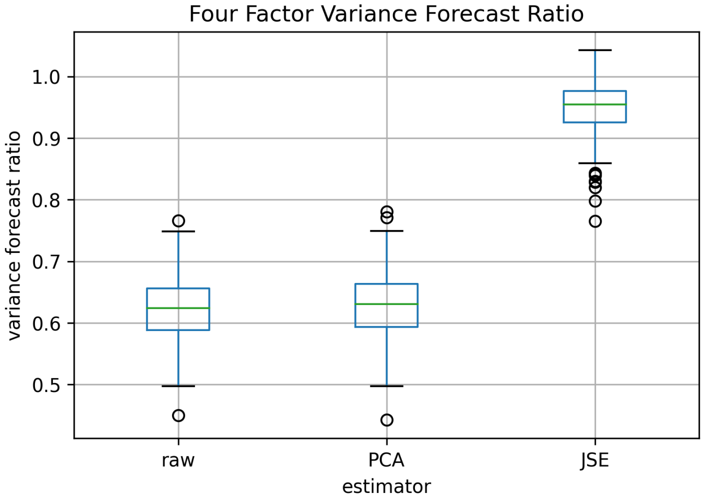


Figure 5: Variance forecast ratio for 400 experiments under the four factor model of returns. A perfect variance forecast ratio is equal to 1.

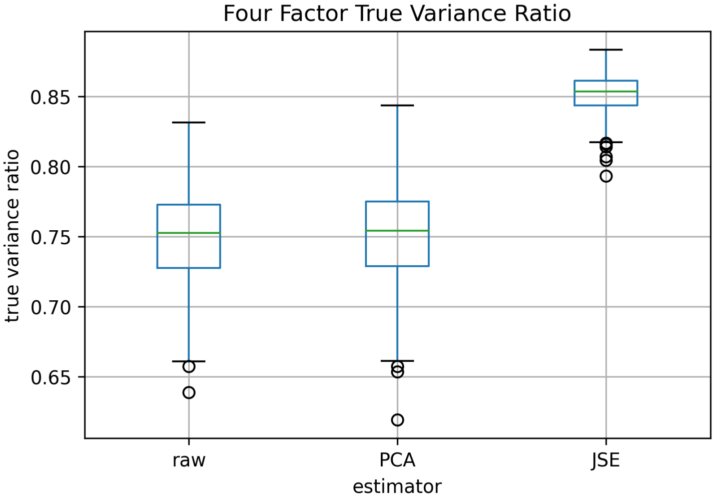


Figure 6: True variance ratio for 400 experiments under the four factor model of returns. A perfect true variance ratio is equal to 1.

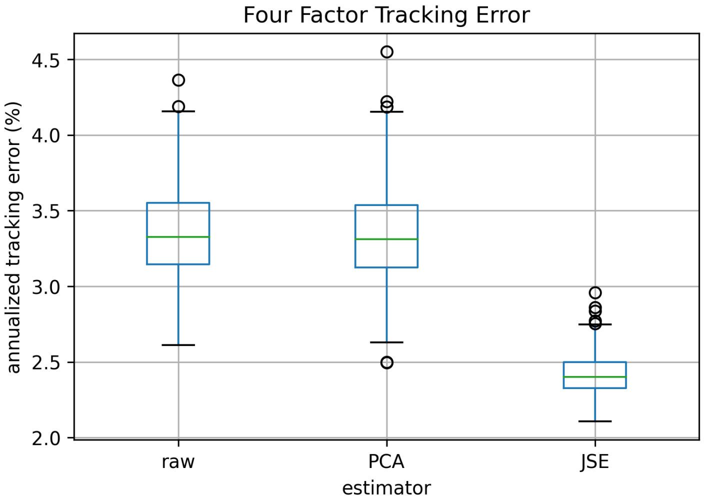


Figure 7: Tracking error for 400 experiments under the four factor model of returns. A perfect tracking error is equal to 0%.

6.2.1 Analysis of Four Factor Simulation

In this study, we evaluate the impact of the Marchenko Pastur and eigenvector correction on the performance metrics of our portfolio under the four factor model of returns. We conducted 400 experiments using fixed values of $p = 500$ and $n = 252$, and analyzed the tracking error, variance forecast ratio, and true variance ratio for three different estimators: Σ_{raw}, Σ_{PCA}, and Σ_{JSE}. Our findings are summarized in figures 5-7. Correcting the market variances and specific variance forecasts through the Marchenko-Pastur correction showed little to no improvement when compared to the shrinking of the leading eigenvector with JSE which led to significant improvements across all performance metrics.

6.3 Empirical Results

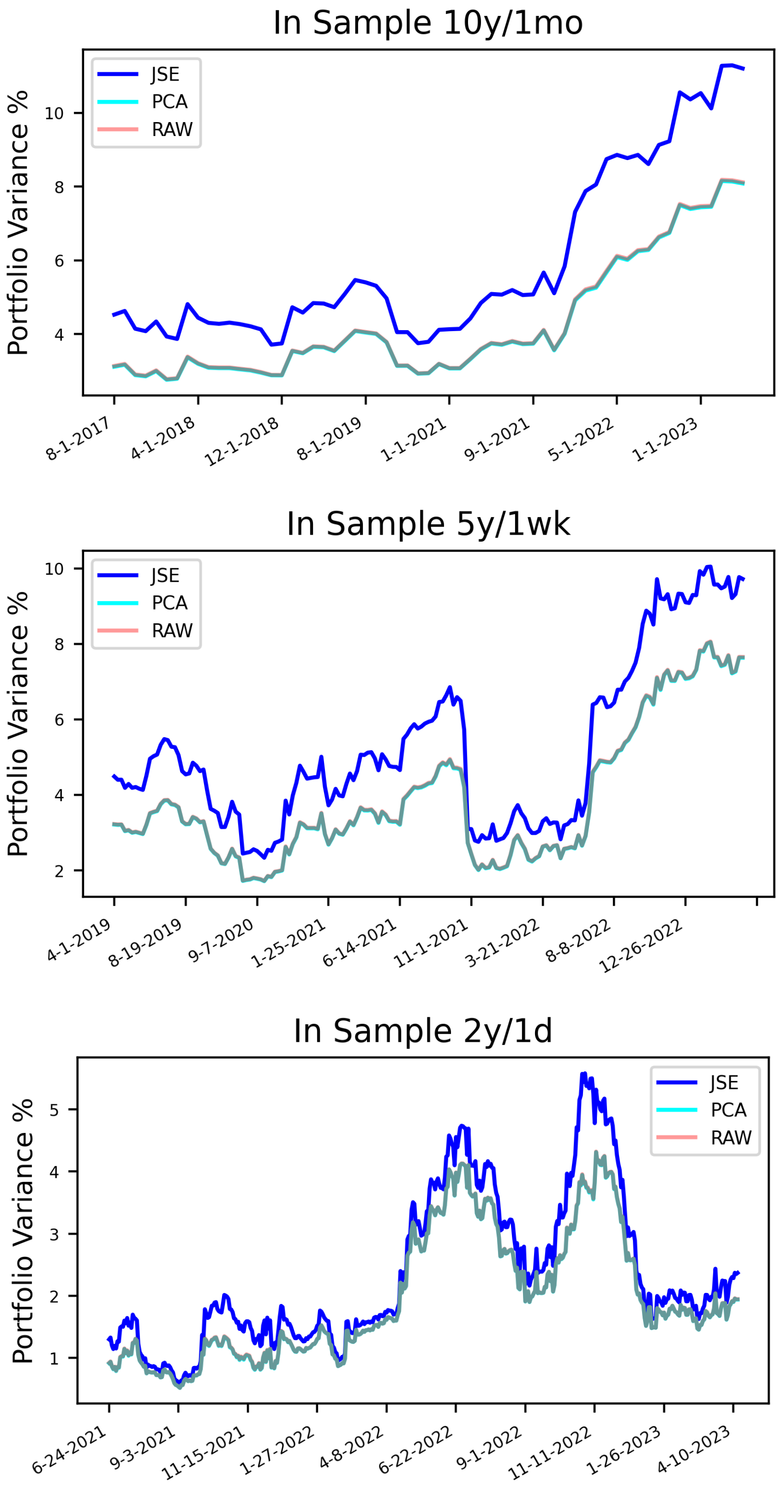


Figure 8: In-sample portfolio variance. The title describes the years of data used to generate the plot and the type of returns used to calculate the factor based covariance matrix. For example, for the "In Sample 5y/1wk" we use 5 years of data and look at weekly returns. We then follow the outline in methodology to compute the in-sample variances as needed for the top 100 stocks of SP500 using 50 observations.

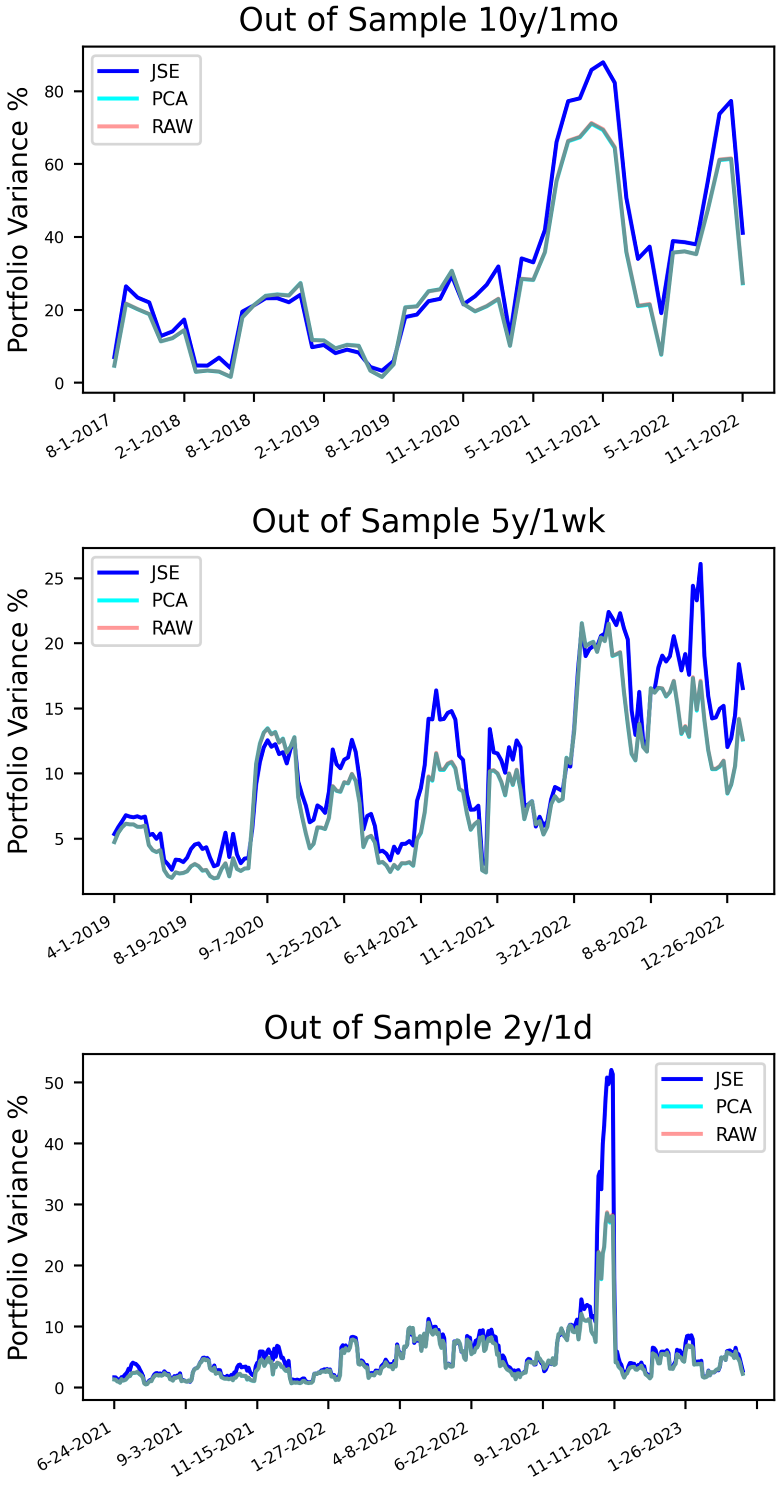


Figure 9: Out-of-sample portfolio variance. The titles describe the years of data used to generate the plot and the type of returns used to calculate the factor based covariance matrix as in the figures for the in-sample test. For the 10y/1mo chart, we let the portfolios at each time-step perform for 6 iterations out-of-sample and collect variances as necessary. For the 5y/1wk and 2y/1d charts we let the portfolios perform for 12 iterations out-of sample at each iteration.

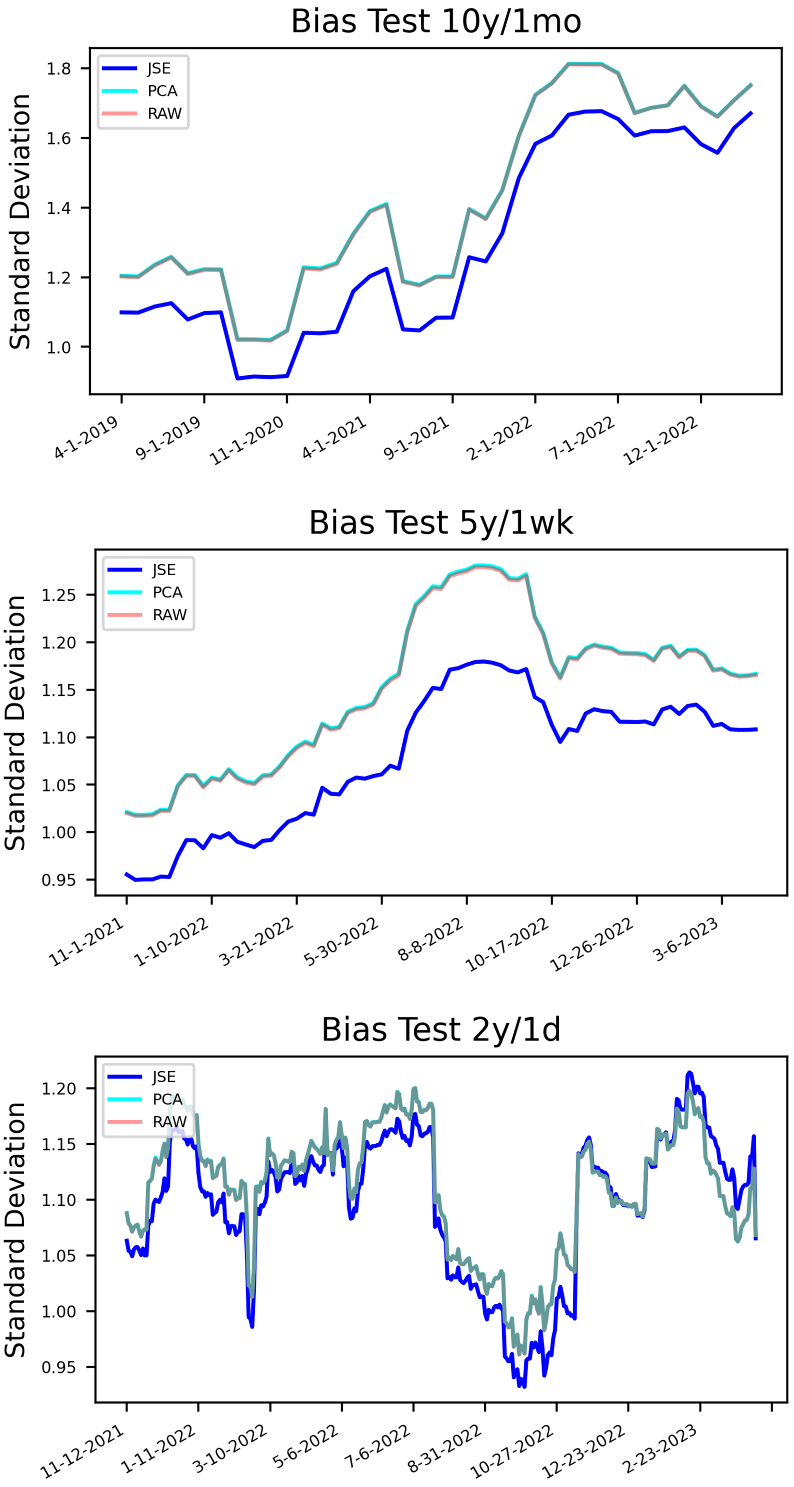


Figure 10: The Bias Test. The titles describe the years of data used to generate the plot and the type of returns used to calculate the factor based covariance matrix as in the figures for the in-sample test. In section **5.2.4**, we describe our method for capturing a large sequence of data, as shown in equation 39. To visualize this data over time, we use a sliding window technique to calculate the standard deviation at each window position. This process is similar to plotting a moving average, but instead, we are plotting the standard deviations.

6.3.1 Analysis of Empirical Results

We will now consider the results presented in figures 8-10. For the in-sample tests, we observed what was expected. In terms of minimum variance, PCA and raw estimators were more "successful" in producing in-sample results. However, as mentioned in [7], in-sample minimum variance is severely underestimated for large portfolios, relative to that encountered out of sample. This is because the optimization in (3) exploits the deviations of h from the true vector b to hedge out the perceived systematic risk, yielding a deceptively small portfolio variance [7]. As hoped for, the JSE estimator increases the in-sample portfolio variance reducing the optimization bias. As to which estimator was better in performing out-of-sample, there is no clear winner as JSE, PCA, and raw estimates tend to overlap indicating similar results. Notice though, that the significantly higher variance in the 10y/1mo is due to the fact that we are taking variances of less out of sample returns. In addition, although the JSE estimator did not clearly out-perform its counter parts in any of the images on figure 9, it came closest in the 10y/1mo test. This could be due to the fact that monthly returns suffer the least from serial correlation. Finally for the Bias test, the JSE estimator out-performs for all return types as it's almost consistently closer to the optimal value of 1. Please note that PCA and RAW in figures 8-10 overlap.

7 Conclusions

In many applications, accurate estimation of population covariance matrices in high dimension low sample size scenarios is crucial. This thesis investigates the performance of eigenvector versus eigenvalue correction in the HL through simulation experiments, considering both one-factor and four-factor return environments. The results indicate that the James Stein estimator (JSE) outperforms in terms of variance forecast ratio, true variance ratio, and tracking error.

To further investigate the JSE estimator's efficacy, we conducted an empirical study, looking at in-sample portfolio variance. The JSE estimator had slightly higher portfolio variance, suggesting less optimization bias and potentially better out-of-sample results. However, the out-of-sample test produced mixed results, making it challenging to draw firm conclusions about the JSE estimator's ability to produce better estimates of minimum variance portfolios.

Nonetheless, the bias test indicated that the JSE estimator performed well in estimating variance. The simulation experiments' limitations, such as the assumption of constant volatility and the limited representation of the market portfolio, could explain the differences in the results between simulation and empirical studies.

Overall, while the JSE estimator showed significant improvement in simulation experiments, neither eigenvector nor eigenvalue correction provided a substantial improvement in out-of-sample portfolio variance. These findings suggest that further research is necessary to determine the best approach for accurate PCA factor-based covariance matrix estimates in real market conditions.

Limitations: There are many limitations in the implications of the simulation results and empirical study. In theory, the simulations are indicating that the shrinkage done by the JSE estimator *should* translate to real market conditions given that the simulation attempts to reflect market conditions. From the standpoint of a portfolio manager, we would want to see the correction in optimization bias translate to lower out-of-sample variance for the JSE estimator when compared to PCA and RAW and thus a better estimate of the true minimum variance portfolio. However, as we see in figure 9, this is not always the case. In future experiments we would hope to use something more industry grade such as a Barra-Type factor model which decomposes risk and return of a portfolio into several different factors including exposure to industry, interest rates, exchange rates, earnings, volatility, sales growth and more. By including more factors we would hope to establish a more realistic risk model such that shrinkage could be analyzed at the industry level.

It is also important to note that the simulations themselves produce independent returns. This is not the case in the real world as serial correlation tends to be magnified in daily, weekly, and even monthly returns. Some future experiments could be conducting simulations where serial correlation is accounted for.

In terms of the Empirical study, the further back we look the less accurate our stocks are of representing the "market portfolio" as stock dominance tends to change over time. In simple terms, the less accurate our representation of the market portfolio, the less accurate our principal component, which translates into worst estimates of the minimum variance portfolio. In the future we could build a system that can account for the fluctuation of the market portfolio and stocks entering and exiting the index. In addition it is important to note the limitations of the in-sample test. For example, we produce in sample portfolio variances for a fixed portfolio over approximately 4 years every iteration. Portfolio managers tend to re-balance portfolios at least quarterly to reflect market changes. Thus, the in-sample test tells us very little from an industry stand point. Still, it allows us to see the correction in optimization bias for the JSE estimator.

It is also worth noting that the removal of the COVID-19 era does not completely mitigate the sharp changes in volatility regimes which set the empirical study even further from simulation. Further investigation in simulation for dynamic volatility accountability could help understand JSE's ability to reflect empirically.

Code: All code used in conducting these simulations and empirical experiments can be accessed in my GitHub repository by following this link: https://github.com/simonribas3/HITM_Research

References

[1] Zhidong Bai and Jack W Silverstein. *Spectral analysis of large dimensional random matrices*, volume 20. Springer, 2010.

[2] M Bayraktar, Igor Mashtaler, Nicolas Meng, and Stan Radchenko. Barra us total market equity model for long-term investors: Empirical notes. *MSCI Model Insight*, 2014.

[3] Stephen W Bianchi, Lisa R Goldberg, and Allan Rosenberg. The impact of estimation error on latent factor model forecasts of portfolio risk. *The Journal of Portfolio Management*, 43(5):147–156, 2017.

[4] David L Donoho, Matan Gavish, and Iain M Johnstone. Optimal shrinkage of eigenvalues in the spiked covariance model. *Annals of statistics*, 46(4):1742, 2018.

[5] Noureddine El Karoui. High-dimensionality effects in the markowitz problem and other quadratic programs with linear constraints: Risk underestimation. 2010.

[6] Lisa R Goldberg and Alec N Kercheval. James–stein for the leading eigenvector. *Proceedings of the National Academy of Sciences*, 120(2):e2207046120, 2023.

[7] Lisa R Goldberg, Alex Papanicolaou, and Alex Shkolnik. The dispersion bias. *SIAM Journal on Financial Mathematics*, 13(2):521–550, 2022.

[8] Lisa R Goldberg, Alex Papanicolaou, Alex Shkolnik, and Simge Ulucam. Better betas. *The Journal of Portfolio Management*, 47(1):119–136, 2020.

[9] W James and C Stein. Proc. fourth berkeley symp. math. statist. probab. *Estimation with quadratic loss*, 1:361–379, 1961.

[10] Michael C Jensen, Fischer Black, and Myron S Scholes. The capital asset pricing model: Some empirical tests. 1972.

[11] Ian T Jolliffe and Jorge Cadima. Principal component analysis: a review and recent developments. *Philosophical transactions of the royal society A: Mathematical, Physical and Engineering Sciences*, 374(2065):20150202, 2016.

[12] Noureddine El Karoui. On the realized risk of high-dimensional markowitz portfolios. *SIAM Journal on Financial Mathematics*, 4(1):737–783, 2013.

[13] Olivier Ledoit and Michael Wolf. Honey, i shrunk the sample covariance matrix. *The Journal of Portfolio Management*, 30(4):110–119, 2004.

[14] Vladimir Alexandrovich Marchenko and Leonid Andreevich Pastur. Distribution of eigenvalues for some sets of random matrices. *Matematicheskii Sbornik*, 114(4):507–536, 1967.

[15] HM Markowitz. Portfolio selection. e journal of finance, 7 (1), 77-91, 1952.

[16] Charles Stein. Inadmissibility of the usual estimator for the mean of a multivariate normal distribution. Technical report, Stanford University Stanford United States, 1956.

[17] Charles Stein. *Some problems in multivariate analysis*. 1956.

[18] Weichen Wang and Jianqing Fan. Asymptotics of empirical eigenstructure for high dimensional spiked covariance. *Annals of statistics*, 45(3):1342, 2017.