

Portfolio Optimization With Stochastic Volatility Models

Ioannis Patsilivas

Imperial College London

Department of Mathematics

Submitted in fulfillment of the requirements for the degree of
Doctor of Philosophy in Statistics

Declaration of Originality

I hereby declare that this thesis and the work presented herein are the result of my own original research. To the best of my knowledge and belief, this thesis contains no material previously published or written by another person, except where due reference is made in the text.

Ioannis Patsilivas

December 2024

Department of Mathematics
Imperial College London

Copyright Declaration

The copyright of this thesis rests with the author. Unless otherwise indicated, its contents are licensed under a Creative Commons Attribution-Non Commercial 4.0 International Licence (CC BY-NC).

Under this licence, you may copy and redistribute the material in any medium or format. You may also create and distribute modified versions of the work. This is on the condition that: you credit the author and do not use it, or any derivative works, for a commercial purpose.

When reusing or sharing this work, ensure you make the licence terms clear to others by naming the licence and linking to the licence text. Where a work has been adapted, you should indicate that the work has been changed and describe those changes.

Please seek permission from the copyright holder for uses of this work that are not included in this licence or permitted under UK Copyright Law.

Abstract

This thesis explores portfolio optimization, balancing theoretical advancements with practical applications. It demonstrates that optimal portfolios, constructed from Dow Jones stocks, can outperform the benchmark index. From a methodological perspective, it pioneers the use of the SMC² algorithm in portfolio optimization, enhancing robustness through theoretical innovations. By automating the optimization process, it mitigates biases from human intervention, offering practical insights for academics and investors. This research bridges the gap between mathematical rigor and real-world applicability, contributing to the fields of financial mathematics and investment management.

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisor, Dr. Nikolas Kantas, for his unwavering support, guidance, and encouragement throughout my PhD journey. His insights and expertise have been invaluable in shaping this research.

I also wish to acknowledge the financial support provided by CDT in Financial Computing, which is located at University College London, department of computing engineering, and made this research possible. The resources and facilities provided by Imperial College were instrumental in conducting this work.

To my wife Ausrine for always believing in me, your support, patience, and encouragement have been my anchor throughout this challenging journey. A special thanks to my family members for always supporting me and cheering me on.

Lastly, I would like to thank anyone else who has contributed, directly or indirectly, to the completion of this thesis.

Thank you all.

Dedicated to the cherished memory of my beloved parents,

Panagiotis & Olymbia

“Measure what is measurable, and make measurable what is not so.”

— *Archimedes*

Contents

Declaration of Originality	i
Copyright Declaration	ii
1 Introduction	1
1.1 Motivation and Contributions	1
1.2 Organization of Thesis	2
1.3 Literature Review	3
1.3.1 Covariance Matrix Estimation	3
1.3.2 Portfolio Optimization	4
1.3.3 Stochastic Volatility Models	5
2 How big are your covariance matrix eigenvalues? The effect in portfolio optimization	8
2.1 Introduction	8
2.1.1 Contributions and Organisation of Chapter	10
2.2 Markowitz Model with Short Selling	11
2.2.1 The Merton, Roll and the Efficient Frontier	11
2.2.2 The Profit and Loss function	12
2.2.3 A Simple Example	13
2.3 Estimation of μ , Σ	14
2.3.1 Ledoit-Wolf (LW) shrinkage estimator for Σ	14
2.3.2 Single-Index Factor Model	15
2.3.3 Direct Estimation of the Precision Matrix	16
2.4 Numerical Results	17
2.4.1 Accuracy of Covariance Estimators	18
2.4.2 Simulated Profit and Loss of Precsion Estimators	22
2.4.3 DJ30 Efficient Stock Portfolios PL	24
2.5 Discussion	25
3 Portfolio Optimization With Stochastic Volatility Models	26
3.1 General problem statement	26
3.2 Cost functions	27

3.3	State space models and particle filtering	28
3.3.1	State space models	28
3.3.2	Bayesian filtering	28
3.3.3	Particle filters	30
3.3.4	Bayesian parameter estimation and predictive posteria	32
3.4	Models	34
3.4.1	An Independent Stochastic Volatility (ISV) model	34
3.4.2	A Multivariate Stochastic Volatility (MSV) model	35
3.4.3	A Factor Model (FM) approach	37
3.5	Application to Dow Jones Industrial Average 30 data	38
3.5.1	MCMC estimation	38
3.5.2	Predicted credible intervals and realised PL	40
3.5.3	Predicted asset allocations under various constraints	41
3.6	Bayesian VS Frequentist optimal portfolios	42
3.6.1	MSV Model P&L With Credible Intervals	43
3.6.2	MSV Model \$1 P&L Distribution	44
3.6.3	MSV Model Sharpe Ratio Distribution	45
3.7	Conclusions	45
4	Volatility Estimation With Sequential Monte Carlo Squared	49
4.1	Introduction	49
4.2	Iterated batch importance sampling, IBIS	49
4.3	Sequential MonteCarlo Squared, SMC ²	50
4.4	Experimental Setup	53
4.4.1	Training Data Preprocessing	54
4.5	Training SMC ² Algorithm for MSV Model	55
4.5.1	Sampling SMC ² Prior	55
4.5.2	Rejuvenation step	56
4.5.3	Residual Resampling	60
4.6	Applications	60
4.6.1	Case Study 1	62
4.6.2	Case Study 2	69
4.7	Conclusions	73
5	Conclusions & Future Work	75
5.1	Conclusions	75
5.2	Future Work	76
A	Σ Matrix	78
A.1	Preliminaries	78
A.2	The eigenvalues of Σ and their support	79
A.3	Inverting Σ	80
A.4	The determinant of Σ	80

A.5	Cholesky decomposition of Σ	81
B	Chapter 2 Appendix	83
B.1	Average Frobenius Error for Covariance, Precision Estimation	83
C	Chapter 3 Appendix	86
C.1	ISV Model Results	86
C.1.1	ISV Model Apple Stock MCMC Chains	86
C.1.2	ISV Model Apple Stock MCMC Autocorrelations	87
C.1.3	ISV Model Apple Stock θ Posterior Distribution	88
C.1.4	ISV Model P&L With Credible Intervals	89
C.1.5	ISV Model \$1 P&L Distribution	90
C.1.6	ISV Model Sharpe Ratio Distribution	91
C.2	MSV Model Results	92
C.2.1	MSV Model MCMC Chains	92
C.2.2	MSV Model θ Posterior Distribution	93
C.3	One Factor Model Results	94
C.3.1	One Factor Model MCMC Chains & Posterior Distributions for Bank Of America Stock	94
C.3.2	One Factor Model P&L With Credible Intervals	95
C.3.3	One Factor Model \$1 P&L Distribution	96
C.3.4	One Factor Model Sharpe Ratio Distribution	97
C.3.5	Optimal Allocations Per Cost Function	98
D	Chapter 4, Case Study 1	101
D.1	Estimating $p(\sigma_\epsilon^2 \mathbf{y}_{1:T})$ Sequentially in Time	101
D.2	Optimal Allocations	107
D.3	P&L Results	109
D.3.1	Realized P&L Tables	109
D.3.2	Out Of Sample Realized Trajectory, When \$1 Is Invested	111
D.3.3	Credible Intervals	113
D.3.4	Optimal Portfolio Terminal Value Distribution	115
D.3.5	Optimal Portfolio Sharpe Ratio Distribution	117
E	Chapter 4 Case Study 2	119
E.1	Optimal Allocations	119
E.2	P&L Results	121
E.2.1	Realized P&L Tables	121
E.2.2	Out Of Sample Realized Trajectory, When \$1 Is Invested	123
E.2.3	Credible Intervals	125
E.2.4	Optimal Portfolio Terminal Value Distribution	127
E.2.5	Optimal Portfolio Sharpe Ratio Distribution	129

List of Figures

2.1	Efficient Frontier Curve	12
2.2	Incremental PL and Efficient Frontier of DJ30 stocks	14
2.3	Performance comparison of covariance and precision matrix estimators (Sample covariance, Ledoit-Wolf, and Factor Model). Eigenvalues greater than one are used for PCA and precision estimators. DJ30 data is divided into 22 periods, each with 176 sampled returns. Covariance is estimated using the first 88 vectors; Frobenius error is evaluated using the remaining 88 vectors. . . .	19
2.4	Comparison of Frobenius error distributions for covariance matrix estimation methods (Sample covariance, Ledoit-Wolf, and Factor Model) using boxplots. Simulated DJ30 data is derived from a multivariate normal distribution, and the eigenvalues greater than one are used for PCA and precision estimators.	20
2.5	Comparison of Frobenius error distributions for precision matrix estimation methods (Sample covariance, Ledoit-Wolf, and Factor Model) using boxplots. Simulated DJ30 data is derived from a multivariate normal distribution, and the eigenvalues greater than one are used for PCA and precision estimators.	21
2.6	Sorted spectra of covariance matrices for the four cases. The black horizontal lines represent eigenvalues greater than one for each case, from right to left: Case 1, Case 4, case 3 and case 2.	22
2.7	Incremental portfolio loss (PL) for six precision estimators: Sample (S), LW, FM, and their respective PCA estimators when $\lambda_k > 1$. The dataset consists of 1000 returns drawn from a multivariate normal distribution, divided into training (200 samples) and testing (800 samples) sets. Results are averaged over 2000 simulations. Each estimator is used as a proxy to define the optimal allocation using Equation (2.4), and the incremental PL path is calculated according to Equation (2.14).	23
2.8	DJ30 Covariance Eigenvalues	24
2.9	DJ30 incremental PL	24
3.1	A graphical model for the Hidden Markov Model and volatility estimation .	29
3.2	Trajectory of \$1 invested in Dow Jones 30 index for the time period 03/01/2011 - 01/07/2011. Blue vertical line indicates the split of the time into training and testing set respectively.	38

3.3	MSV P&L clouds created by 1000 simulated portfolio trajectories for $K = 66$ future time steps, based on the optimal portfolio $\mathbf{w}_{T+K}$. Red and blue lines represent the realized P&L paths of the optimal allocation $\mathbf{w}_{T+K}$ and the Dow Jones, respectively, when \$1 is invested at $K = 0$. Here, $\mathbf{w}_{T+K}$ minimizes various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$	43
3.4	Distribution of the 1000 simulated optimal portfolio values for day $T + K$, where future time steps $K = 66$ and the initial investment equals \$1 at $K = 0$. Red and blue lines represent the realized portfolio values for $\mathbf{w}_{T+K}$ and the Dow Jones, respectively. $\mathbf{w}_{T+K}$ optimizes various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$	44
3.5	Distribution of the 1000 simulated optimal portfolio Sharpe Ratios for day $T + K$, where $K = 66$ future time steps and the initial investment equals \$1 at $K = 0$. Red and blue lines represent the realized Sharpe Ratios for $\mathbf{w}_{T+K}$ and the Dow Jones, respectively. $\mathbf{w}_{T+K}$ optimizes the various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$	45
3.6	Portfolio value trajectories for various models and utility functions. Each panel shows the performance of optimal portfolio allocations under different utility function specifications, with or without boundary constraints.	46
4.1	Dow Jones 30 stock three-days log-returns, from 04/01/2013 to 30/09/2013 .	62
4.2	θ Prior	63
4.3	Sequential Posterior Estimation of σ_ϵ^2 , at time $t = 1, 8, 39, 43$	64
4.4	θ -particles scatter plots before and after rejuvenation steps at time $t = 5$. . .	66
4.5	Acceptance ratio, contrasting adaptive step size per θ parameter	66
4.6	Number of particles that constitute positive definite $\Sigma_{x_t x_{t-1}, y_{t-1}}$ matrices. . .	67
4.7	Global minimum variance optimal portfolio results when the first 15 data points are trained and calculate the out-of-sample P&L over a forward period of 15 time points.	68
4.8	Standardization per stock after capping and flooring for each quarter.	70
4.9	Third quarter training diagnostics.	71
4.10	Global minimum variance optimal portfolio results when the third quarter data points are trained and calculate the out-of-sample P&L over a forward period of 21 time points.	72
B.1	Performance comparison of covariance and precision matrix estimators (Sample covariance, Ledoit-Wolf, and Factor Model) when returns follow a Student-t distribution. Eigenvalues greater than one are used for PCA and precision estimators. Simulated DJ30 data is divided into 22 periods, each with 176 sampled returns. Covariance is estimated using the first 88 vectors; Frobenius error is evaluated using the remaining 88 vectors.	83

B.2	Comparison of Frobenius error distributions for covariance matrix estimation methods (Sample covariance, Ledoit-Wolf, and Factor Model) using boxplots. Simulated DJ30 data is derived from a multivariate t-distribution with 3 degrees of freedom, and the eigenvalues greater than one are used for PCA and precision estimators.	84
B.3	Comparison of Frobenius error distributions for precision matrix estimation methods (Sample covariance, Ledoit-Wolf, and Factor Model) using boxplots. Simulated DJ30 data is derived from a multivariate t-distribution with 3 degrees of freedom, and the eigenvalues greater than one are used for PCA and precision estimators.	85
C.1	Trace plots for ISV Model parameters θ (Apple MCMC chain results): α , β , r , ρ , σ^2	86
C.2	MCMC chain autocorrelations for ISV Model parameters θ (Apple Stock).	87
C.3	Marginal posterior distributions of ISV Model parameters θ (Apple Stock). Parameters include α , β , r , ρ , and σ^2	88
C.4	P&L clouds from 2000 simulated portfolio trajectories over $K = 66$ future steps. Red/blue lines show realized P&L of optimal allocation $\mathbf{w}_{T+K}$ and Dow Jones index, respectively.	89
C.5	Distribution of the 1000 simulated optimal portfolio values for day $T + K$, where future time steps $K = 66$ and the initial investment equals \$1 at $K = 0$. Red and blue lines represent the realized portfolio values for $\mathbf{w}_{T+K}$ and the Dow Jones, respectively. $\mathbf{w}_{T+K}$ optimizes various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$	90
C.6	Distribution of the 1000 simulated optimal portfolio Sharpe Ratios for day $T + K$, where $K = 66$ and the initial investment equals \$1 at $K = 0$. Red and blue lines represent the realized Sharpe Ratios for $\mathbf{w}_{T+K}$ and the Dow Jones, respectively. $\mathbf{w}_{T+K}$ optimizes the various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$	91
C.7	MCMC chains for parameters θ in the MSV model.	92
C.8	Posterior distributions for parameters θ in the MSV model.	93
C.9	MCMC chains and marginal posterior distributions for Bank of America under the one-factor model. The figure presents the factor load, log-variance σ^2 , and parameter ϕ	94
C.10	One Factor model P&L clouds created by 1000 simulated portfolio trajectories for $K = 66$ future time steps, based on the optimal portfolio $\mathbf{w}_{T+K}$. Red and blue lines represent the realized P&L paths of the optimal allocation $\mathbf{w}_{T+K}$ and the Dow Jones, respectively, when \$1 is invested at $K = 0$. Here, $\mathbf{w}_{T+K}$ minimizes various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$	95

C.11	Distribution of the 1000 simulated optimal portfolio values for day $T + K$, where future time steps $K = 66$ and the initial investment equals \$1 at $K = 0$. Red and blue lines represent the realized portfolio values for $\mathbf{w}_{T+K}$ and the Dow Jones, respectively. $\mathbf{w}_{T+K}$ optimizes various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$	96
C.12	Distribution of the 1000 simulated optimal portfolio Sharpe Ratios for day $T + K$, where $K = 66$ and the initial investment equals \$1 at $K = 0$. Red and blue lines represent the realized Sharpe Ratios for $\mathbf{w}_{T+K}$ and the Dow Jones, respectively. $\mathbf{w}_{T+K}$ optimizes the various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$	97
C.13	Optimal portfolio allocations for Dow Jones 30 stocks using global variance minimization.	98
C.14	Optimal portfolio allocations for Dow Jones 30 stocks using return maximization.	99
C.15	Optimal portfolio allocations for Dow Jones 30 stocks using fixed-return variance minimization.	100
D.21	Sequential Posterior Estimation of σ_ϵ^2 for 61 observations in Time	106
D.22	Portfolio optimal weight allocation using the global minimum variance cost function.	107
D.23	Portfolio optimal weight allocation using the global minimum variance cost function for a fixed expected return.	108
D.24	Portfolio realized value trajectory for the various data sets when \$1 invested in the optimal portfolio given by global minimum variance cost function.	111
D.25	Portfolio realized value trajectory for the various data sets when \$1 invested in the optimal portfolio given by global minimum variance cost function for a fixed level of expected return.	112
D.26	MSV PL clouds created by 10000 simulated portfolio trajectories for $K = 15$ future time steps , based on optimal portfolio w_{T+K} realized value trajectory for the various data sets when \$1 invested in the optimal portfolio given by global minimum variance cost function. Red and blue lines represent the realized PL path of the optimal allocation w_{T+K} and the Dow Jones respectively.	113
D.27	MSV PL clouds created by 10000 simulated portfolio trajectories for $K = 15$ future time steps , based on optimal portfolio w_{T+K} realized value trajectory for the various data sets when \$1 invested in the optimal portfolio given by global minimum variance cost function for a given expected return. Red and blue lines represent the realized PL path of the optimal allocation w_{T+K} and the Dow Jones respectively.	114
D.28	Distribution of MSV terminal values generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function. The red and blue lines indicate the realized profit and loss of the optimal allocation w_{T+K} and the Dow Jones, respectively, for an initial investment of \$1.	115

D.29	Distribution of MSV terminal values generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function for a given level of expected return. The red and blue lines indicate the realized profit and loss of the optimal allocation w_{T+K} and the Dow Jones, respectively, for an initial investment of \$1.	116
D.30	Distribution of MSV SR generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function. The red line indicates the realized SR of the optimal allocation w_{T+K} for an initial investment of \$1.	117
D.31	Distribution of MSV SR generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function for a given level of expected return. The red line indicates the realized SR of the optimal allocation w_{T+K} for an initial investment of \$1.	118
E.1	Portfolio optimal weight allocation using the global minimum variance cost function.	119
E.2	Portfolio optimal weight allocation using the global minimum variance cost function for a fixed expected return.	120
E.3	Portfolio realized value trajectory for the various quarters when \$1 invested in the optimal portfolio given by global minimum variance cost function. . .	123
E.4	Portfolio realized value trajectory for the various quarters when \$1 invested in the optimal portfolio given by global minimum variance cost function for a fixed level of expected return.	124
E.5	MSV PL clouds created by 10000 simulated portfolio trajectories for $K = 15$ future time steps , based on optimal portfolio w_{T+K} realized value trajectory for the various quarters when \$1 invested in the optimal portfolio given by global minimum variance cost function. Red and blue lines represent the realized PL path of the optimal allocation w_{T+K} and the Dow Jones respectively.	125
E.6	MSV PL clouds created by 10000 simulated portfolio trajectories for $K = 15$ future time steps , based on optimal portfolio w_{T+K} realized value trajectory for the various quarters when \$1 invested in the optimal portfolio given by global minimum variance cost function for a given expected return. Red and blue lines represent the realized PL path of the optimal allocation w_{T+K} and the Dow Jones respectively.	126
E.7	Distribution of MSV terminal values generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function. The red and blue lines indicate the realized profit and loss of the optimal allocation w_{T+K} and the Dow Jones, respectively, for an initial investment of \$1.	127

E.8	Distribution of MSV terminal values generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function for a given level of expected return. The red and blue lines indicate the realized profit and loss of the optimal allocation w_{T+K} and the Dow Jones, respectively, for an initial investment of \$1.	128
E.9	Distribution of MSV SR generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function. The red line indicates the realized SR of the optimal allocation w_{T+K} for an initial investment of \$1.	129
E.10	Distribution of MSV SR generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function for a given level of expected return. The red line indicates the realized SR of the optimal allocation w_{T+K} for an initial investment of \$1.	130

List of Tables

3.1	Annualized performance metrics of optimal portfolios evaluated out-of-sample	47
4.1	Summary of θ Priors	57
4.2	Summary of θ Samplers and Jacobians	58
4.3	Setting the number of particles and rejuvenation steps for SMC ²	61
4.4	First 15 data points training set, fifteenth out-of-sample optimal portfolio realized performance.	69
4.5	Third quarter training set, 21 periods out-of-sample optimal portfolio realized performance.	73
D.1	First 15 data points training set, fifteenth out-of-sample optimal portfolio realized performance.	109
D.2	First 30 data points training set, fifteenth out-of-sample optimal portfolio realized performance.	109
D.3	First 45 data points training set, fifteenth out-of-sample optimal portfolio realized performance.	110
D.4	First 61 data points training set, fifteenth out-of-sample optimal portfolio realized performance.	110
E.1	First quarter training set, 21 periods out-of-sample optimal portfolio realized performance.	121
E.2	Second quarter training set, 21 periods out-of-sample optimal portfolio realized performance.	121
E.3	Third quarter training set, 21 periods out-of-sample optimal portfolio realized performance.	122
E.4	Fourth quarter training set, 21 periods out-of-sample optimal portfolio realized performance.	122

Chapter 1

Introduction

1.1 Motivation and Contributions

Portfolio optimization is a cornerstone of modern financial theory and practice, aimed at constructing investment portfolios that maximize expected returns while minimizing risk. The foundation of this field traces back to Harry Markowitz's groundbreaking work on mean-variance optimization in 1952, which introduced a systematic framework for balancing risk and return. While Markowitz's framework provided profound insights, the evolving complexity of financial markets and the increasing availability of high-dimensional data have necessitated continuous innovation and research in this area.

This thesis focuses on portfolio selection and optimization not just as a mathematical challenge but as a practical application that delivers meaningful and attractive results for investors. Much of the existing literature prioritizes mathematical properties and performance metrics, often sidelining critical aspects of practical value. For instance, limited attention is given to questions such as the distribution of weights in optimal portfolio allocations across different models and cost functions, or how the profit and loss (P&L) trajectory of an optimal portfolio behaves compared to benchmark indices over its investment horizon. Similarly, discussions around the underlying risk of the portfolio, the establishment of reliable future P&L bounds, or the impact of reduced reallocation frequency on minimizing transaction costs remain insufficiently explored. Questions about the relative performance of sophisticated stochastic volatility models versus simpler, computationally efficient frequentist models also remain open. Furthermore, the potential benefits of combining models compared to relying on a single robust approach are rarely addressed.

This thesis seeks to fully or partially address these questions. Among its contributions, two stand out as particularly significant. The first contribution lies in demonstrating the practical value of portfolio optimization for practitioners, portfolio managers, investors, and individual pensioners allocating their endowments. By examining various cost functions under diverse scenarios, such as permitting or restricting short selling, using real-world data over realistic investment horizons, we establish that the portfolio optimization problem can yield effective solutions. Our analysis reveals that optimal portfolios constructed using stocks

from traditional indices, such as the Dow Jones, often outperform the indices themselves. Moreover, frequent reallocation is unnecessary. Reallocations performed quarterly, with partial adjustments at the end of the holding period, are sufficient to achieve strong results while keeping transaction costs low.

The second major contribution of this thesis is its mathematical and algorithmic innovation. We delve into frequentist and Bayesian covariance estimators derived from stochastic volatility models and rigorously compare their effectiveness in the portfolio optimization problem. Notably, for the first time, the SMC² algorithm is applied to portfolio optimization. Through theoretical advancements, including the proof and application of key lemmas and propositions, as well as the reparameterization of a Multivariate Stochastic Volatility (MSV) model, we enhanced the robustness of the algorithm. Our findings show that portfolio optimization can be fully automated, eliminating the need for human intervention. This is particularly significant, as human involvement in investment decisions is often biased by psychological factors such as fear and greed, which can impair the decision-making process.

In summary, this thesis bridges the gap between theoretical developments and practical applications in portfolio optimization. By addressing important unanswered questions, delivering innovative algorithmic solutions, and demonstrating real-world applicability, this research makes meaningful contributions to both academic and professional financial communities.

1.2 Organization of Thesis

Chapter one provides an introduction to the thesis, outlining its motivations and including a review of relevant literature. Chapter two introduces three frequentist covariance estimators: the empirical covariance, the Ledoit-Wolf shrinkage covariance (Ledoit & Wolf 2004a), and a modified version of the latter (Ledoit & Wolf 2003). Chapter two addresses the portfolio optimization problem and introduces the performance metrics, including profit and loss (P&L) tools, that will be utilized for analyzing the results of the case studies. Additionally, diagnostic experiments are conducted to evaluate the accuracy and performance of these estimators under various market conditions, with a particular focus on the behavior of their eigenvalues as market dynamics change.

Chapter three introduces the cost functions to be optimized and provides an overview of state space models. Three stochastic volatility models are examined: a scalar reparameterized stochastic volatility model and two multivariate stochastic volatility models based on Xu & Jasra (2019) and Kastner (2019). These models are applied to the log-returns of stocks constituting the Dow Jones Index. Optimal allocations derived from these models are compared in terms of their P&L performance across different estimators and cost functions. This chapter establishes a framework for evaluating the effectiveness of optimization techniques in practical scenarios.

Chapter four focuses on the implementation of the SMC² Algorithm for a multivariate stochastic volatility model, building on the work of (Xu & Jasra 2019). A reparameterized form of the model is presented, and mathematical tools are developed to enhance the ro-

bustness of the algorithm. The model is applied to two case studies, and its P&L results are compared to those obtained using the Ledoit-Wolf shrinkage covariance estimator (Ledoit & Wolf 2004a).

In Chapter five, the main conclusions of this work are presented, along with suggestions for potential directions to further advance research in this area. The majority of the results are provided in the Appendices, which are organized by chapter. Specifically, separate Appendices are included for Chapters 2 and 3, detailing the respective findings and analysis. For Chapter 4, two dedicated Appendices are included, each corresponding to one of the case studies. Additionally, the first Appendix is provided containing all the mathematical proofs and results utilized in Chapter 4.

1.3 Literature Review

This section reviews key literature relevant to the estimation of covariance and precision matrices, portfolio optimization, and stochastic volatility models. The discussion is organized to provide a high-level overview of the methods and findings, with specific details elaborated in later sections as necessary.

1.3.1 Covariance Matrix Estimation

Covariance and precision (inverse covariance) matrix estimation are foundational to many areas of multivariate statistical analysis, with applications ranging from finance to biology (Fan et al. 2014). Estimating large (high-dimensional) covariance matrices is a particularly challenging problem in the big data era (Fan et al. 2016). This section highlights general approaches to covariance and precision matrix estimation, categorized into four main methodologies:

1. Shrinkage techniques;
2. Structured models;
3. Random Matrix Theory (RMT);
4. Bayesian methods.

Shrinkage Techniques Shrinkage methods are introduced by Efron & Morris (1977) and later expanded in Daniels & Kass (2001). Shrinkage involves shrinking the sample covariance matrix towards a structured target matrix to improve estimation accuracy. A well-known shrinkage estimator is the Ledoit-Wolf (LW) (Ledoit & Wolf 2004a) estimator, which is suitable for large dimensional covariance matrices, especially when the number of samples is small compared to the number of features. In this case, the stability and accuracy of the empirical covariance is improved by shrinking the sample covariance matrix toward a structured target, typically the identity matrix. However, there exists the improved version

of LW (Ledoit & Wolf 2003), where the covariance is shrunk towards a matrix, which depicts the covariance of each underlying stock with a stock index. Further elaboration with these two versions of LW estimators will take place in Chapter 2.

Structured Models Structured models, often referred to as "Factor models," include observable and latent factor models (Bai & Shi 2011). These approaches exploit structure in the covariance matrix to improve estimation, combining observable and latent factors as necessary.

Random Matrix Theory (RMT) Recent advancements in covariance and precision matrix estimation leverage RMT, grounded in principles like Wigner’s semicircle law (Wigner 1958) and the Marcenko-Pastur law (Marčenko & Pastur 1967). RMT addresses eigenvalue distributions in large random matrices, with applications such as portfolio optimization highlighted by Bun et al. (2016). Specific findings, including the Tracy-Widom law (Tracy & Widom 2002) and eigenvalue dispersion results (Ledoit & Wolf 2004a), are critical to understanding covariance estimation.

Bayesian Methods Bayesian estimation involves assigning priors to the covariance matrix. Common choices include the inverse Wishart distribution (Barnard et al. 2000) and its scaled or hierarchical variants (O’Malley & Zaslavsky 2008, Huang et al. 2013). Separation strategies for Bayesian covariance estimation are explored by Barnard et al. (2000).

1.3.2 Portfolio Optimization

Portfolio optimization, a cornerstone of financial theory, is addressed from various perspectives in the literature and is a well-studied and understood topic. As a result, we will not provide a comprehensive literature review of the entire subject area and only focus on selected papers that are close in spirit with this work. The classical Markowitz MV portfolio optimisation theory (Markowitz 1952) assumes that the mean and variance of the asset returns are known. Lai et al. (2011) depart from the classic Markowitz MV setting by maximising the expected return of their portfolio while simultaneously penalising the portfolio variance. The expected returns and covariance are obtained using Bayesian inference.

Kourtis et al. (2012) uses Shrinkage theory to estimate directly the inverse covariance matrix using two non-parametric methods with the aim to enhance portfolio performance. The first estimator minimizes the out-of-sample portfolio variance and the second aims to increase the out-of-sample risk-adjusted returns. The resulting estimators are used to compute the optimal weights that minimize the portfolio variance. The derived portfolios often provide higher risk-adjusted returns and lower levels of risk.

Ait-Sahalia & Xiu (2017) construct an estimator for the number of common factors in a setting where the sampling frequency and the number of variables increase. Empirically, Ait-Sahalia & Xiu (2017) document that the covariance matrix of a large portfolio of US equities is well represented by a low-rank common structure with a sparse residual matrix.

When employed for an out-of-sample portfolio allocation, Ait-Sahalia & Xiu (2017) conclude that the proposed estimator outperforms the sample covariance estimator.

A continuous time framework to solve the Markowitz MV problem has been applied by (Lindberg et al. 2009). Also, Gandy & Veraart (2013) look at the Merton problem in continuous time with the asset model being geometrical BM and consider the effect of the number of assets in the portfolio together with the estimation of the covariance. In many ways, the approach followed here is similar, but we work in discrete time with a moderate number of assets and without the existence of a riskless bond.

Bayesian Extensions: The Black-Litterman Model

In the context of Bayesian portfolio optimization, the Black-Litterman model (Black & Litterman 1992) represents a significant development. It overcomes the sensitivity and instability problems of classical MV optimization by integrating market equilibrium assumptions with subjective investor views. Rather than directly specifying expected returns, which are difficult to estimate accurately, the model starts with a set of implied equilibrium returns derived from the market portfolio:

$$\pi = \delta \cdot \Sigma \cdot w_{mkt}$$

where π is the implied return vector, δ is the risk aversion coefficient, Σ is the covariance matrix of asset returns, and w_{mkt} represents market capitalization weights.

Investor views are then modeled as:

$$Q = P \cdot \mu + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \Omega)$$

Here, Q contains the views, P is a matrix specifying the assets to which the views apply, and Ω is a diagonal covariance matrix encoding the confidence in each view. Using a Bayesian framework, the model updates the prior (market) distribution with these views to form a posterior distribution of returns:

$$\mu_{BL} = [(\tau\Sigma)^{-1} + P^T\Omega^{-1}P]^{-1} [(\tau\Sigma)^{-1}\pi + P^T\Omega^{-1}Q]$$

This posterior is then used for optimal allocation. The Black-Litterman model leads to more stable and diversified portfolios and is particularly suitable in institutional investment settings where subjective views need to be systematically incorporated into portfolio construction. Its flexibility has led to numerous extensions including factor-based and dynamic versions (Meucci 2006, He & Litterman 1999).

1.3.3 Stochastic Volatility Models

Stochastic volatility (SV) models describe time-varying volatility effects on asset prices, critical for derivatives pricing and risk management. Introduced by Taylor (1986), SV models feature two processes:

- Observed process y_t : asset returns;
- Latent process x_t : volatilities.

One of the main reasons that the popularity of SV has increased in financial applications is that the previously intractable likelihood function of the underlying asset return can be numerically approximated. Modern inference methods, such as Sequential Monte Carlo (SMC) (Doucet et al. 2009) and Sequential Monte Carlo Squared (SMC²) (Chopin et al. 2013) enable numerical likelihood approximation for Bayesian parameters. The particle approximation of the marginal likelihoods can be used within Particle MCMC to infer θ in a Bayesian manner.

SV has the capacity to be flexible enough to capture the leverage effect (Geweke et al. 1994), which refers to the observed tendency of an asset’s volatility to be negatively correlated with the asset’s returns. For example, when the stock’s price falls sharply, it has been observed a considerable spike-up of the stock’s volatility (Bouchaud et al. 2001). This is an important feature that plays a big role in derivatives pricing and risk management, especially when stock prices fall, as this phenomenon partially explains why stocks have increased correlations during bear market periods. The leveraged effect in SV can be introduced allowing a correlation between the log-variance and asset’s return disturbances (Harvey & Shephard 1996, Jacquier et al. 2004). SV is also known for being able to capture two other important characteristics of financial time series, which are the volatility clustering (Ding & Granger 1996) and the fat-tailed distributions of asset returns, known as excessive kurtosis (Fama 1965). This effect can be incorporated allowing the asset’s return disturbance to have a Student t -distribution (Ruiz 1994, Chib et al. 2002, Jacquier et al. 2004).

The natural extension of the univariate SV model is the multivariate stochastic volatility model (MSV), which is particularly useful in portfolio selection and risk management. The MSV simultaneously models the movements in the volatility of a number of assets and tries to capture the covariation effect, which refers to the parallel movement phenomenon of volatilities of different financial time series. This is because assets can be formally linked together, for example the bankruptcy of a company can affect the counter-party risk of other companies, or because asset prices can be affected by unobserved common factors. In addition, the globalization of equity has caused an increase in the correlation of stock returns (Ghysels et al. 1996). The extension from the univariate SV to multivariate MSV is a very challenging numerical task (Dellaportas et al. 2021) because the computational complexity increases quadratically as the number of assets in focus gets bigger, and also because the covariance matrix Σ_t , which describes the volatilities of the underlying asset returns, must be positive definite. Also for MSV models, while there are reliable numerical approximations, like particle filters, the same methods can be computationally very expensive and it may be challenging to obtain accurate results for θ (Kantas et al. 2015).

Developing MSV models is a classic research area starting from Harvey et al. (1994), Pitt & Shephard (1999), Aguilar & West (2000), where the correlations are treated as constants over time or modelled through factor models, with factors being independent univariate stochastic volatility models. These models were extended among others by Philipov &

Glickman (2006), Han (2006), Chib et al. (2006), Carvalho & West (2007), Gouriéroux et al. (2009), Nakajima & West (2012) and Ishihara & Omori (2017). Lately, advances have been made so that MSV models are able to incorporate a very large number of assets inducing parsimony via various methods. Kastner (2019) makes use of a global–local shrinkage prior, which pulls the elements of the factor loadings towards zero. Xu & Jasra (2019) uses two advanced particle filters proposed by Beskos et al. (2017), named the space-time particle filter and the marginal space-time particle filter. Dellaportas et al. (2021) has a different approach where Gaussian latent processes for functions of the eigenvalues and rotation angles of Σ_t are used. By invert-transforming back to Σ_t , the positive definiteness is ensured.

Chapter 2

How big are your covariance matrix eigenvalues? The effect in portfolio optimization

2.1 Introduction

Mean Variance (MV) optimization (Markowitz 1952) dictates that the investor should select a set of weights $(w_1, w_2, \dots, w_d)$ for a set of assets $i = 1, 2, \dots, d$ that minimize the portfolio risk, subject to meeting a level of expected returns. The main idea of the MV model is to treat the returns of the available securities as random variables, and adopt the expected value and variance to quantify the investment return and risk respectively. Mathematically, the MV problem can be formulated as:

$$\mathbf{w}_* = \arg \min_{\mathbf{w}} \mathbf{w}^T \Sigma \mathbf{w} \quad (2.1)$$

subject to:

$$\begin{aligned} \mathbf{w}^T \boldsymbol{\mu} &= R \\ \mathbf{w}^T \mathbf{1}_d &= 1 \\ w_i &\geq 0, \quad i = 1, 2, \dots, d, \end{aligned} \quad (2.2)$$

where Σ is the $d \times d$ covariance matrix of the d asset returns r_i , $\boldsymbol{\mu}$ is the d dimensional vector of the asset's expected returns μ_i , $\mathbf{1}_d$ is the d dimensional vector with each element equal to unity, and R is an exogenously specified scalar that represents the desired target return of the portfolio. The original MV portfolio optimization problem (Markowitz, 1972) assumes that $w_i \geq 0$, which means that short selling is not allowed. Later, short selling will be incorporated by Merton (1972) and by Roll (1977).

Note that in the original formulation of the MV problem, weights are *not* updated dynamically. The investor makes their optimal allocation at the start of their investment horizon and holds it until the end. The Markowitz model assumes that the investor does not adjust their portfolio selection $\mathbf{w}$ during their investment time span. In this way, the

investor decides which assets they will choose and how to allocate their initial endowment, but they do not change their portfolio allocation until the end of their investment. Such a scenario is unrealistic because the prices of the stocks can form non-stationary time series, and this means that the investor may need to change their allocation in the underlying stocks in order to increase their performance. For example, a stock that was in an up-trend for a period of time may now be in a down-trend; such a case would mean that the investor may have to diminish the weight of this stock in their portfolio and possibly increase it in some other stocks. To improve on this, one would need to formulate the weights at a given time as a stochastic process adapted to the available information on the assets up to that time. This would require solving a stochastic control problem (Zhou & Li 2000), but here we will not deal with this case.

In addition, the correlations among the stocks may change in time. This means that the covariance matrix will have to be adjusted to the prevailing market conditions as the previously estimated covariance is no longer accurate. A change in the covariance matrix would definitely change the optimal allocation. As a result, practitioners should choose values of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ using historical data over relevant periods. This means that to obtain a solution of the Markowitz portfolio model (2.1) one needs to:

- Find an estimator $\hat{\boldsymbol{\mu}}$ for the expected return $\boldsymbol{\mu}$, and set an appropriate level R for the desired target return;
- Find an estimator $\hat{\boldsymbol{\Sigma}}$ or $\hat{\boldsymbol{\Sigma}}^{-1}$ for the covariance matrix $\boldsymbol{\Sigma}$ or the precision matrix $\boldsymbol{\Sigma}^{-1}$ respectively;
- Solve the optimization problem for the allocation $\mathbf{w}_*$ that minimizes the objective function $\mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w}$.

Regarding the modelling framework for an MV problem, financial returns are typically modelled using standard time series. In this chapter, we assume that returns are i.i.d. and we will focus on parameter estimation techniques for $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ using methods based on sample averages. This is in contrast to traditional methods in time series (MLE, Bayesian) but allows simplicity in estimation and it also seems appropriate as a first step for considering portfolio optimization.

To assess the performance of the estimation for $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, we will use both real and simulated data. Data will be generated using the benchmark index of Dow Jones stocks. We will split datasets into training and testing subsets to estimate $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, then assess the performance of the optimal portfolio. The portfolio performance will be evaluated at every time step in the investment horizon (not just the terminal time). To do this, we will use a Profit and Loss (PL) function (to be defined later in Section 2.2.2) and the Sharpe Ratio (Sharpe 1994)

$$SR = \frac{R_t - R_I}{\sigma_p}, \quad (2.3)$$

where R_t and R_I denote the return of the optimal portfolio at a given time t and the return of a benchmark index or asset (e.g. 10-year Treasury) in the test set respectively, and σ_p

denotes the standard deviation of the optimal portfolio's excess return. The estimation method for μ and Σ affects the trajectory of both the PL function and the terminal value of the portfolio. For this reason, it will be prudent to devote Section 2.3 to review the popular estimators of μ , Σ and discuss how they affect the PL function.

To understand how the various underlying estimators affect the portfolio's performance, we investigate the case where short selling is allowed, namely $w_i \in \mathfrak{R}$. In contrast to the case where $w_i \geq 0$, when $w_i \in \mathfrak{R}$, computing $\mathbf{w}_*$ is analytically tractable. In the former case, numerical optimization is required (Boyd & Vandenberghe 2004). As mentioned above, the positivity condition on the weights is involved in the classic Markowitz model and is relevant to real applications because many financial institutions, for example, pension funds have to conform with particular regulations that prohibit short selling.

As mentioned above, we will break the investment horizon into multiple time windows where at the start of each window we re-estimate μ and Σ and the optimal weights according to (2.4). However, this task raises questions such as what is the optimal length of the estimation window, and the appropriate investment horizon. The resolution of these issues is model specific and beyond the scope of this report. We could also employ adaptive estimation or calibration methods that do not require specifying explicit window or horizon sizes (Bickel 1982, Moore & Weiss 1979) but again this is left for future studies.

2.1.1 Contributions and Organisation of Chapter

A summary of the novel contributions made in this work can be summarized as follows:

- We attempt to investigate the effect of singular values of estimated covariance or precision matrix for the PL (Profit and Loss) function of the optimal portfolio. This gives a clearer intuition to the practitioner in terms of performance and risk.
- We present numerical results on historical data of the stocks that comprise the Dow Jones. We present a thorough comparison of different covariance estimators and their Principal Components Analysis (PCA) counterparts. In numerical simulations not shown here, we obtain similar conclusions for the Standard and Poors 100 index.

Section 2.2 describes the Markowitz model when short selling is permissible. Section 2.3 is devoted to the presentation of the most widely known covariance estimators applied to log-stock returns. Section 2.4 contains results from numerical simulations to validate the theoretical results contained in Sections 2.2 and 2.3. Examples of the numerical simulations in Section 2.4 include bull and bear markets characterised by low and high volatility regimes. Monte Carlo simulations that were performed indicate that PCA increases the bias of covariance and precision estimators. The numerical results contained in Section 2.4 indicate that PCA can improve the incremental PL of the optimal portfolio; this could possibly be attributed to the few large eigenvalues of the covariance matrix formed from the log-stock returns.

2.2 Markowitz Model with Short Selling

2.2.1 The Merton, Roll and the Efficient Frontier

We commence this section with some background details on the solution of the portfolio allocation problem when $w_i \in \mathfrak{R}$. Most of the discussion in this section is motivated by material in Merton (1972) and Roll (1977). Our aim is to provide some intuition on how investors should choose R in 2.1 and how this will affect the variance of the optimal portfolio. Clearly, these quantities depend on the estimated μ, Σ , so when assessing the performance of the optimal portfolio this explicit dependence should be taken into account together with the estimation error. An MV model admits the following closed-form solution when $w_i \in \mathfrak{R}$:

$$\mathbf{w}_* = \frac{1}{ac - b^2} \Sigma^{-1} [(c - bR)\mathbf{1} + (aR - b)\boldsymbol{\mu}] \quad (2.4)$$

$$a = \mathbf{1}^T \Sigma^{-1} \mathbf{1}, \quad b = \mathbf{1}^T \Sigma^{-1} \boldsymbol{\mu}, \quad c = \boldsymbol{\mu}^T \Sigma^{-1} \boldsymbol{\mu} \quad (2.5)$$

This optimal solution is a function of the precision matrix Σ^{-1} . From a computational perspective, it seems easier to work with the precision matrix than the covariance matrix, but we will return to this point in Section 2.3. Substituting the optimal $\mathbf{w}_*$ into equation (2.1), we obtain the optimal portfolio variance as a function of the target return R and the parameters a, b, c :

$$\sigma^2 = \frac{aR^2 - 2bR + c}{ac - b^2}. \quad (2.6)$$

Equation (2.6) is a quadratic with respect to the target return R , and defines the set of points (R, σ^2) that the portfolio allocation is optimal when short selling is allowed. Substituting the values of the parameters a, b, c defined by (2.5) into (2.6) we get:

$$\sigma^2 = \frac{(R\mathbf{1} - \boldsymbol{\mu})^T \Sigma^{-1} (R\mathbf{1} - \boldsymbol{\mu})}{ac - b^2} > 0. \quad (2.7)$$

The numerator of equation (2.7) is positive definite, and the quantity $b^2 - ac$ is strictly negative. Hence, there are no real values of R that are roots of (2.6). Let R_* denote the minimizer of σ^2 , and we find that:

$$\sigma_*^2 = \frac{1}{a} \quad \text{for} \quad R_* = \frac{b}{a}. \quad (2.8)$$

The curve of (R, σ^2) points has the shape depicted in Figure 2.1. A rational investor will choose values of R that lie on the yellow part of the curve. This curve represents all the points of maximum target returns for a fixed amount of variance. For this reason, the yellow curve is also known as the *efficient frontier*. The point where the color of the curve in Figure 2.1 changes from blue to yellow is the global minimum variance point (R_*, σ_*^2) on the efficient frontier represented by (2.8). Hence, a rational investor should choose an R such that $R \geq b/a$.

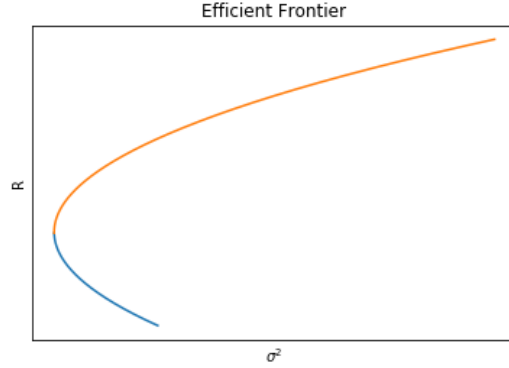


Figure 2.1: Efficient Frontier Curve

The efficient frontier provides an unlimited number of optimal solutions for various target returns, and this fact raises the question as to which of these target returns should be chosen. The most common answer to this question is to choose the so-called tangency portfolio, which is the portfolio that maximizes the Sharpe Ratio (Sharpe 1966, 1994). Sharpe Ratio is a measure of portfolio performance and counts the portfolio's return per unit of standard deviation. This could be described as :

$$\mathbf{w}_{ta} = \arg \max_{\mathbf{w}} \frac{\mathbf{w}^T \boldsymbol{\mu}}{\sqrt{\mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w}}}, \quad (2.9)$$

where $\mathbf{w}_{ta}$ represents the allocation that maximizes the Sharpe Ratio and belongs to the efficient portfolio space. Equation (2.9) can be expressed in terms of the difference of the expected return and a benchmark value. This performance ratio is also known as the *information ratio* (Grinold & Kahn 2000) and commonly used benchmark values are market indices such as SP500 and Dow Jones.

2.2.2 The Profit and Loss function

The data that we are going to work with are adjusted daily returns of stocks and stock indices. Adjusted stock daily returns are calculated by the stock's daily closing prices, where the splits and the dividends are included in the price of the securities. Daily stock returns can be expressed in simple or compound returns.

For stock i , the compound interest rate at the end of time t is given by the formula

$$r_{i,t} = \log(P_{i,t}/P_{i,t-1}) \quad (2.10)$$

where $P_{i,t}$ and $P_{i,t-1}$ are the daily stock closing prices for the days t and $t - 1$ respectively. To find the daily compounded portfolio rate r_t at time t we have to sum up the compound interest rates $r_{i,t}$ multiplied by $w_{i,t}$, which represents the proportion of wealth invested in security i .

Note that $w_{i,t}$ does change in time, because while we start with an optimal allocation w_{*i,t_0} at time t_0 , the stock prices $p_{i,t}$ at time t change. This change affects the initial portfolio w_{*i,t_0} allocation during the investment horizon. Thus to compute $w_{i,t}$ we need to first calculate the quantity $q_{i,0}$ of each stock that we must buy or short sell at time t_0 . Quantity $q_{i,0}$ is constant in time and can be calculated as $q_{i,0} = w_{*i,t_0} V_0 / p_{i,0}$. So to compute $w_{i,t}$ we use the formula:

$$w_{i,t} = \frac{p_{i,t} q_{i,0}}{\sum_{i=1}^d p_{i,t} q_{i,0}}. \quad (2.11)$$

Hence, the portfolio daily compound return r_t is:

$$r_t = \sum_{i=1}^d w_{i,t} r_{i,t}. \quad (2.12)$$

We can then express the daily value of the portfolio V_t at time t as a function of the portfolio daily compound return:

$$V_t = e^{r_t} V_{t-1}. \quad (2.13)$$

The factor e^{r_t} is also known as the future factor because when it is applied to the price of an asset it returns the future value of the asset.

Combining equations (2.12) and (2.13), and taking into account all the trading days of the testing set $t = 1, 2, \dots, T$ from the start until the end of the investment horizon, we can express the final value of the portfolio V_T as a function of the daily returns r_t when the initial investment is $V_0 = 1$ as:

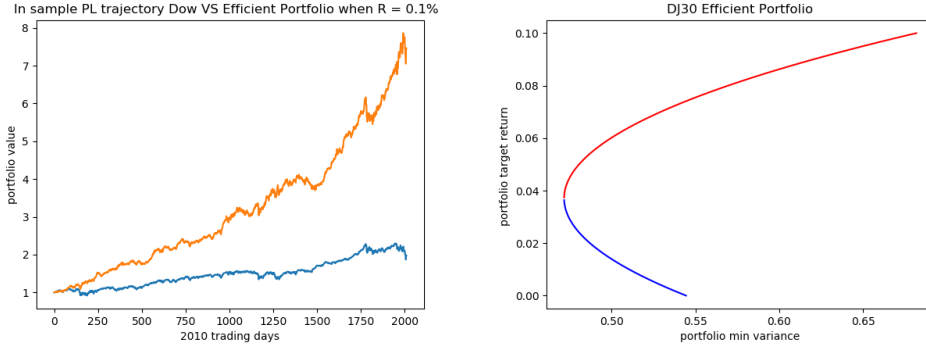
$$V_T = e^{\sum_{t=1}^T \sum_{i=1}^d w_{i,t} r_{i,t}}. \quad (2.14)$$

From equation (2.14) we can retrieve the value of the portfolio trajectory when 1 dollar is invested for a particular initial allocation. Equation (2.14) establishes the PL function at time T and will be used as a benchmark for analysing the performance of the various investment strategies in the testing set. Also from now on, we will use the convention of referring to compound returns simply as returns. Often we are interested in the full path and not just in V_T , for example we would like to avoid excessive draw-downs. As a result, we will assess the performance using V_t , $t \leq n$. Note that the solution in (2.4) does not take this into account, but this can motivate an area of future study.

2.2.3 A Simple Example

Consider the daily returns $\mathbf{r}_t$ of the Dow Jones index and its 30 stock constituents. We will take into account the time period starting from 2011 until the end of 2018. We will calculate the covariance and the stock returns for the whole time span, and we will solve the MV optimization setting the daily target return equal to 0.1 percent. After we find $\mathbf{w}_*$ according to (2.4), where Σ is replaced with the sample covariance

$$\bar{\mathbf{r}} = (1/n) \sum_{t=1}^n \mathbf{r}_t \quad (2.15)$$



(a) DJ30 and Efficient Portfolio incremental PL when $R = 0.1\%$ (b) DJ30 Stocks Curve Of Efficient Portfolios

Figure 2.2: Incremental PL and Efficient Frontier of DJ30 stocks

and

$$\mathbf{S} = (\mathbf{1}/n) \sum_{t=1}^n (\mathbf{r}_t - \bar{\mathbf{r}})(\mathbf{r}_t - \bar{\mathbf{r}})^T. \quad (2.16)$$

Figure 2.2a shows the PL trajectory of 1 dollar invested in the optimal portfolio (orange line) when the target return R is set equal to 0.1 percent daily. This is contrasted with the PL trajectory of 1 dollar invested in the Dow Jones Index. Figure 2.2b depicts the DJ30 efficient portfolio as described in Section 2.2.1. We see that the optimal portfolio outperforms the index DJ30 considerably. However, this assumes that we know the real mean and covariance for the examined time span, something that is not true in practice.

2.3 Estimation of μ , Σ

Finding the optimal allocation $\mathbf{w}_*$ requires first estimating μ and Σ . In this section, the focus will be on estimating Σ only. The $\mathbf{r}_t$ are i.i.d. so for μ we will use the sample mean estimator, $\hat{\mu} = (1/n) \sum_{t=1}^n \mathbf{r}_t$, where $\mathbf{r}_t$ is the vector of stock daily returns. A standard choice for estimating Σ is the sample covariance, but this can be problematic, especially when $n \leq d$, so we will review some more sophisticated extensions. This is not meant as an exhaustive review, and the methods below were chosen based on their extensive use in financial applications.

2.3.1 Ledoit-Wolf (LW) shrinkage estimator for Σ

The first estimator to review is the Ledoit-Wolf (LW) shrinkage estimator (Ledoit & Wolf 2004a), which is a well-known estimator suitable for large dimensional covariance matrices. When $d \geq n$, $\mathbf{S}$ in (2.16) is ill-conditioned and possibly singular. LW suggests an estimator that bypasses these issues and shows consistent asymptotic behavior. LW consists of a linear combination of two matrices, the identity matrix and the sample covariance matrix. By

default, this linear matrix combination will always be invertible. LW assumes that the r_i 's are i.i.d., but without the need to specify the form of the underlying distribution. To extract the optimal estimator, LW minimizes the expected quadratic loss (Frobenius norm) between the linear matrix combination and the real covariance matrix Σ according to

$$\arg \min_{\rho, \nu} E[||\hat{\Sigma} - \Sigma||^2] \quad (2.17)$$

$$st : \hat{\Sigma} = \rho\nu\mathbf{I} + (1 - \rho)\mathbf{S}. \quad (2.18)$$

In Ledoit-Wolf (2004) optimal solutions for ρ and ν are given as functions of four scalar parameters $(\alpha, \beta, \delta, \mu)$ that are functions of the unknown covariance Σ . LW overcomes this problem by substituting these four scalar parameters with respective consistent estimators. Here ρ acts like a shrinkage factor because the greater ρ is, the less the dominance of the sample covariance is going to be in the resulting estimator. This shrinkage technique is well-known in statistics and dates back to Stein (1956) and Haff (1980).

2.3.2 Single-Index Factor Model

Another approach is to estimate Σ as a weighted average of the sample covariance matrix and covariance by a k -factor model, involving factors that are explanatory macroeconomic variables for the underlying stock returns. This has the advantage that we can introduce a structure in the final estimator that could have a financial meaning. However, there is no consensus on the identity and the number of the factors to be used (Connor & Korajczyk 1995). One common practice is to use one factor that represents the main stock market index return of the underlying stocks (Sharpe 1963).

Sharpe's (1963) single index model stock returns come from the linear regression:

$$x_{it} = \alpha_i + \beta_i x_{0t} + \epsilon_{it} \quad (2.19)$$

where the assumptions of the standard regression model hold and the noise ϵ_{it} is i.i.d. with zero mean and constant variance. In (2.19) x_{it} denotes the stock returns and x_{0t} the market index returns respectively. Using (2.19) we can calculate the elements of the covariance matrix Φ of stock returns x_{it} :

$$\text{Cov}[x_{it}, x_{jt}] = \beta_i \beta_j \sigma_0^2 \quad (2.20)$$

$$\Phi = \text{Cov}(\mathbf{X}) \quad (2.21)$$

$$\Phi = \sigma_0^2 \beta \beta^T + \Delta. \quad (2.22)$$

Vector β is the vector of the slopes β_i and Δ is a diagonal matrix that contains the variances of the residuals σ_i^2 . The single-index model gives the estimated covariance matrix $\hat{\Phi}$ of stock returns:

$$\hat{\Phi} = s_0^2 \hat{\beta} \hat{\beta}^T + \hat{\Delta}, \quad (2.23)$$

where s_0^2 is the sample variance of the index return, $\hat{\beta}$ is the vector of the estimated slopes $\hat{\beta}_i$ and $\hat{\Delta}$ a diagonal matrix containing the estimated residual variances $\hat{\sigma}_i^2$. $\hat{\Phi}$ is an input in the LW (Ledoit & Wolf 2003) estimation procedure using a factor model structure to obtain:

$$\arg \min_{\alpha} E[||\hat{\Sigma} - \Sigma||^2] \quad (2.24)$$

$$st : \hat{\Sigma} = \alpha \hat{\Phi} + (1 - \alpha) \mathbf{S}, \quad (2.25)$$

where α represents the shrinkage parameter.

2.3.3 Direct Estimation of the Precision Matrix

In this section, we represent an alternative way to estimate directly the precision matrix with the help of Principal Component Analysis (PCA) (Hotelling 1933). Later we will see that this can improve performance in terms of P&L compared to the other estimation methods above. To the best of our knowledge, this comparison is largely absent in the literature, and there is limited understanding of how model reduction methods affect the performance of the optimal portfolio.

With some abuse of notation for this subsection, we will suppose that Σ is an arbitrary real symmetric matrix and therefore possesses real eigenvalues and eigenvectors. The eigenvectors can be chosen to form an orthonormal set. So if Σ has d dimensions and distinct eigenvalues, Σ can be diagonalised and written as:

$$\Sigma = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^T, \quad \mathbf{\Lambda} = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_d), \quad \lambda_1 > \lambda_2 > \dots > \lambda_d, \quad (2.26)$$

or equivalently:

$$\Sigma = \sum_{i=1}^d \lambda_i \mathbf{u}_i \mathbf{u}_i^T, \quad (2.27)$$

where $\mathbf{u}_i$ are the corresponding orthogonal eigenvectors of λ_i . Similarly to (2.27), the precision matrix can be expressed as :

$$\Sigma^{-1} = \sum_{i=1}^d (1/\lambda_i) \mathbf{u}_i \mathbf{u}_i^T. \quad (2.28)$$

The properties of the λ_i 's, such as how quickly they decay or how many are too close to zero, are important both when a statistical procedure is used for estimation and when this estimate is used within an MV portfolio allocation. For instance, when the covariance matrix has a few very large eigenvalues and the rest close to zero, *one could choose to ignore the small eigenvalues and keep only the larger ones*.

In our numerical examples, we noticed that this could improve PL performance and this could possibly be explained intuitively by the following arguments:

- The few largest eigenvalues account for explaining most of the variance and covariance of the stocks (Chamberlin & Rothschild 1983);
- Inclusion of eigenvalues close to zero possibly increases the error of the covariance estimation;
- Close to zero eigenvalues can cause difficulty in the inversion of the covariance matrix.

In this way we could estimate the precision matrix by:

$$\hat{\Sigma}^{-1} = \sum_{i=1}^k (1/\hat{\lambda}_i) \hat{\mathbf{u}}_i \hat{\mathbf{u}}_i^T, \quad k < d, \quad (2.29)$$

where $\hat{\lambda}_i$ and $\hat{\mathbf{u}}_i$ denote the eigenvalues and eigenvectors, respectively, of an estimated covariance matrix $\hat{\Sigma}$ and k is the number of the largest components that we use for the estimation. Substituting (2.28) into (2.8), the global minimum variance can be written as a function of the covariance eigenvalues. Taking the first derivative with respect to an arbitrary eigenvalue λ_k we find that

$$\frac{\partial \sigma_*^2}{\partial \lambda_k} = \frac{\sigma_*^2 (\mathbf{u}_k^T \mathbf{1})^T (\mathbf{u}_k^T \mathbf{1})}{\lambda_k^2}. \quad (2.30)$$

An interesting conclusion here is that the variance of the efficient frontier is a strictly increasing function with respect to the covariance eigenvalues, but the rate of change is inversely proportional to the $\lambda_k > 1$ and proportional to $\lambda_k < 1$. So one may get a portfolio with less variance and better Sharpe Ratio if they estimate the covariance matrix by deleting the small eigenvalues, and especially the ones that are smaller than one, that is $\lambda_k < 1$.

2.4 Numerical Results

The methods in Plerou et al. (2002) inspire this work, which analyzes the eigenvalues of covariance and precision matrices. Unlike Plerou et al. (2002), this study emphasizes:

1. Estimation of precision matrices using extreme eigenvalues of covariance matrices;
2. Defining optimal portfolios based on precision matrices;
3. Conducting numerical simulations under bull and bear market scenarios to study PnL trajectories.

Our results suggest that eigenvalues may distinguish between bull and bear markets, a novel contribution beyond Plerou et al. (2002). In this section, we present results on how different covariance estimation affects the cumulative PL trajectory for two different data sets. The data set includes the 30 stocks of the Dow Jones 30 (from 01/01/2011 - 31/12/2018). The data is freely available from Yahoo Finance.

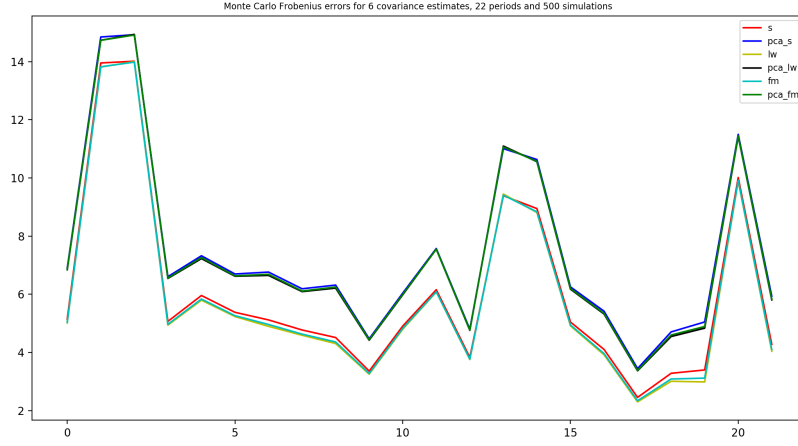
2.4.1 Accuracy of Covariance Estimators

We begin our numerical investigations by testing the accuracy of the different estimators under the assumptions that the returns are i.i.d. and generated by multivariate Normal and Student-t distributions with known mean and covariance, respectively. To adjust our simulations to the real data, we partition the whole time span into four-month sub-periods, and for each sub-period, we calculate the mean vector and the covariance matrix. The calculated mean and covariance will be used as the real parameters to generate returns.

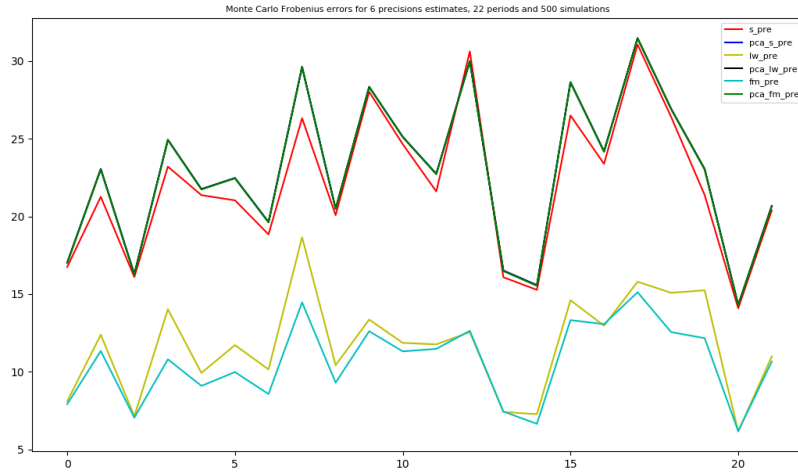
Figure 2.3 depicts the average error in Frobenius norm after 500 independent replications of the estimation procedure, when returns come from a multivariate normal distribution. Note that for the calculation of the PCA precision estimators we have used (2.28). Figure 2.3 shows that LW and FM have the smallest bias compared to all other estimators, while applying PCA to each estimator, the bias increases.

We repeat the same experiment, but now we sample from a multivariate t-Student distribution with three degrees of freedom. The respective results are located in Appendix B.1. Figure B.1 shows that the covariance errors for all the estimators are much larger than those of Figure 2.3, and this is something to expect since the samples come from a distribution with much heavier tails. Furthermore, Figure B.1 shows that the majority of the precision estimators have similar performance, with the exception of the sample precision, which has a noticeably better performance.

In Figure 2.4 and 2.5, we represent the boxplots of the estimator errors of Figures 2.3a and 2.3b respectively. When the returns follow Student-t, the relative figures are provided in the Appendix B.1. Observing Figure 2.4, we see that when returns are normally distributed, the range of the covariance errors is relatively small, except for periods 1, 2, 13, 14, and 20. These periods refer to bear markets (returns of stocks are mostly negative in a period of time) where, in such cases, the returns of the stocks are characterized by high variance. Figure 2.5, which shows the range of the precision estimation errors, shows that when we apply PCA to the estimators, we increase their bias, but we decrease their variance. A closer look at Figures 2.4 and 2.5 shows that LW and FM estimators, especially in the case of precisions, consistently indicate superiority compared to the others. However, as Figure B.3 indicates, this is not confirmed when the returns follow multivariate Student-t, where the sample covariance seems to have a smaller error.

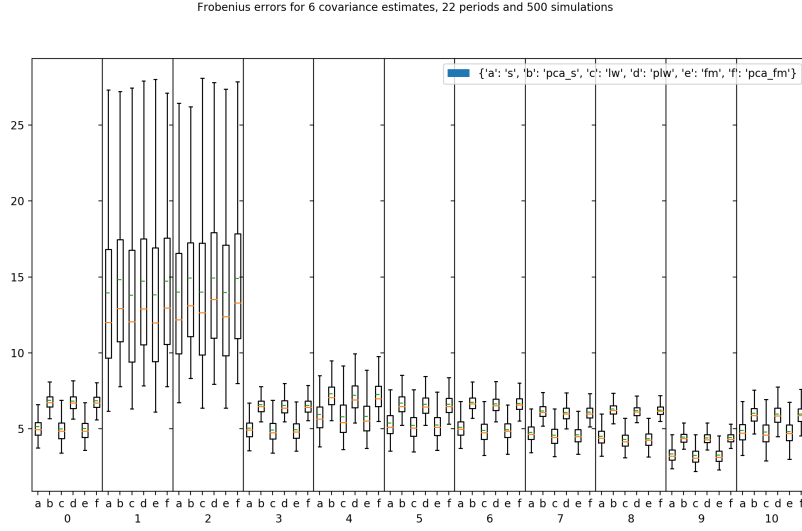


(a) Average Frobenius error for covariance matrix estimation over 500 independent runs using simulated data representative of DJ30 with Normal returns.

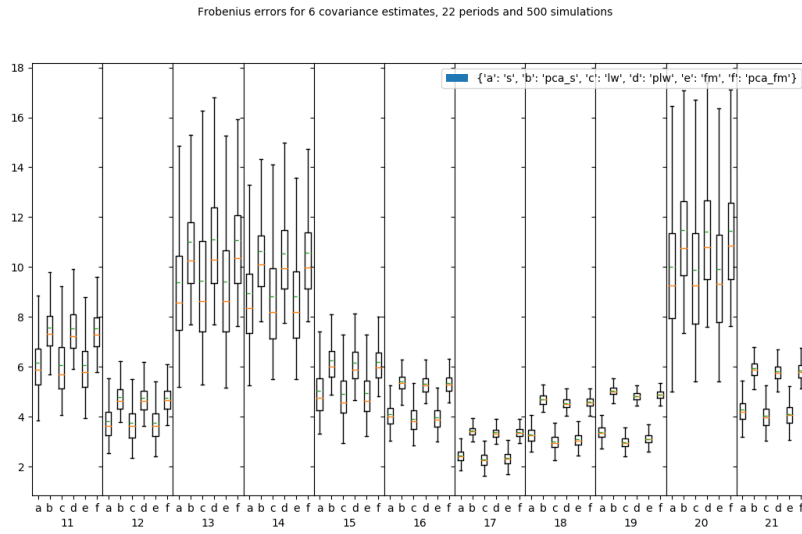


(b) Average Frobenius error for precision matrix estimation over 500 independent runs using simulated data representative of DJ30 with Normal returns.

Figure 2.3: Performance comparison of covariance and precision matrix estimators (Sample covariance, Ledoit-Wolf, and Factor Model). Eigenvalues greater than one are used for PCA and precision estimators. DJ30 data is divided into 22 periods, each with 176 sampled returns. Covariance is estimated using the first 88 vectors; Frobenius error is evaluated using the remaining 88 vectors.

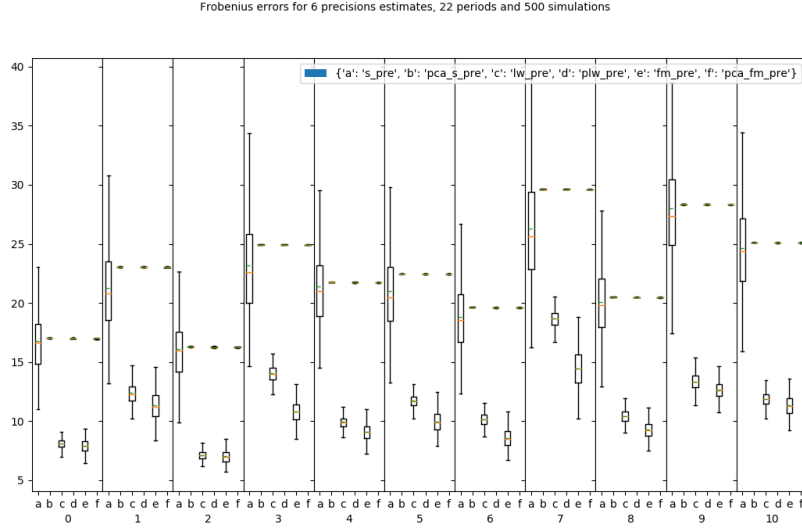


(a) Boxplots of Frobenius error for covariance matrix estimation using simulated DJ30 data from a multivariate normal distribution.

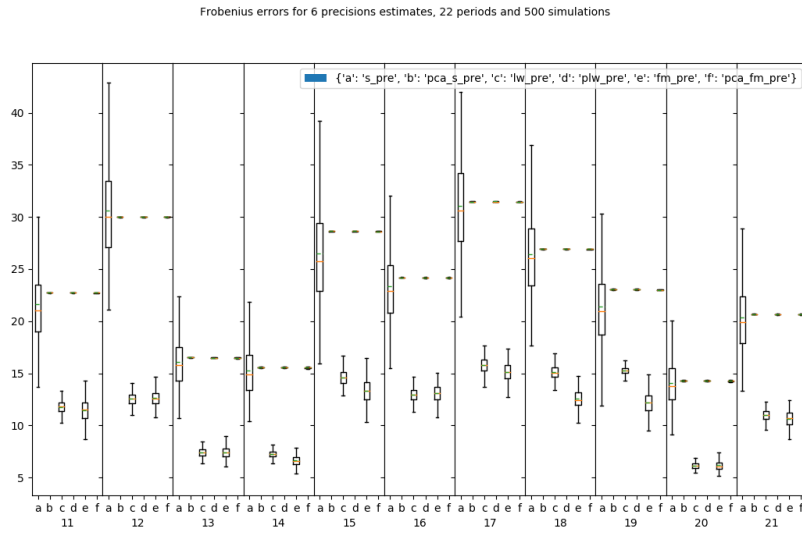


(b) Boxplots of Frobenius error for covariance matrix estimation using simulated DJ30 data from a multivariate normal distribution.

Figure 2.4: Comparison of Frobenius error distributions for covariance matrix estimation methods (Sample covariance, Ledoit-Wolf, and Factor Model) using boxplots. Simulated DJ30 data is derived from a multivariate normal distribution, and the eigenvalues greater than one are used for PCA and precision estimators.



(a) Boxplots of Frobenius error for precision matrix estimation using simulated DJ30 data from a multivariate normal distribution.



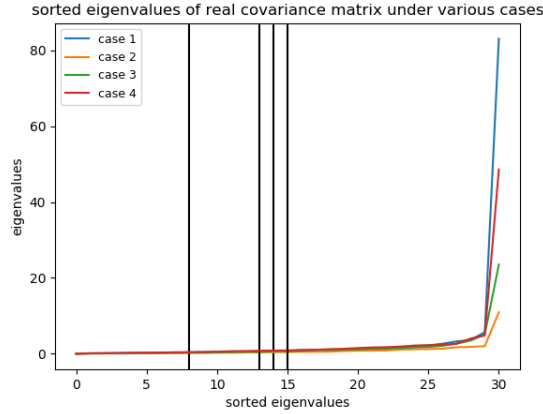
(b) Boxplots of Frobenius error for precision matrix estimation using simulated DJ30 data from a multivariate normal distribution.

Figure 2.5: Comparison of Frobenius error distributions for precision matrix estimation methods (Sample covariance, Ledoit-Wolf, and Factor Model) using boxplots. Simulated DJ30 data is derived from a multivariate normal distribution, and the eigenvalues greater than one are used for PCA and precision estimators.

2.4.2 Simulated Profit and Loss of Precsion Estimators

Next we want to understand how the various precision estimators discussed in Section 2.4.1 perform with respect to the PL function of the optimal portfolio according to (2.4). We will use four different parameter configurations taken from Section 2.4.1 (Figure 2.4), the ones that correspond to the sub-periods (cases) 2,9,11, and 20 (as case 1, 2, 3, 4 respectively). The criterion by which these cases are chosen is the variability of the errors in the boxplots. Note that cases 9 and 11 represent bull-market (up-trend) periods while cases 2 and 20 represent bear-market (down-trend) periods. Here we will discuss only the results that come from a multivariate normal distribution.

First, we plot the covariance spectrum for each case in Figure 2.6. Here we can make two observations. First, for each case, the covariance spectrum has only a few eigenvalues that take large values, and all the rest fall close to one and zero, which is in agreement with (Chamberlin & Rothschild 1983). Second, we notice that the first few covariance eigenvalues in the bear market cases are much larger than those of the bull market cases. This suggests that when stocks are traded in bear market territory, their correlations are much stronger than those of a bull market. This in turn implies that as stocks get into a transition period from a bull to a bear market, one should observe a spike in the covariance eigenvalues. For this reason, the classification of the covariance eigenvalues, using Machine Learning, e.g. SVM or Logistic Regression, could possibly reveal information about the market's phase: bull or bear market. Such knowledge would be valuable to an asset manager because he could be proactive and protect his profits as the market shifts from a bull market to a bear one.



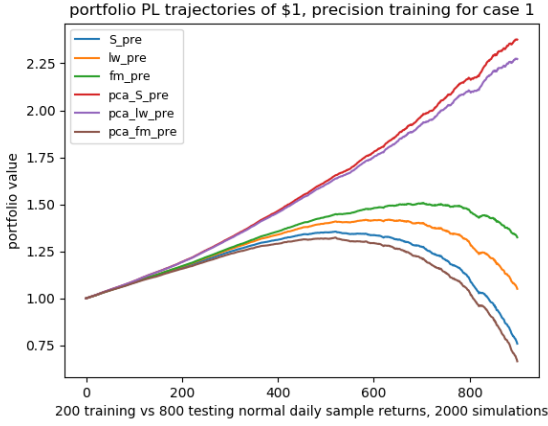
(a) Covariance eigenvalues

Figure 2.6: Sorted spectra of covariance matrices for the four cases. The black horizontal lines represent eigenvalues greater than one for each case, from right to left: Case 1, Case 4, case 3 and case 2.

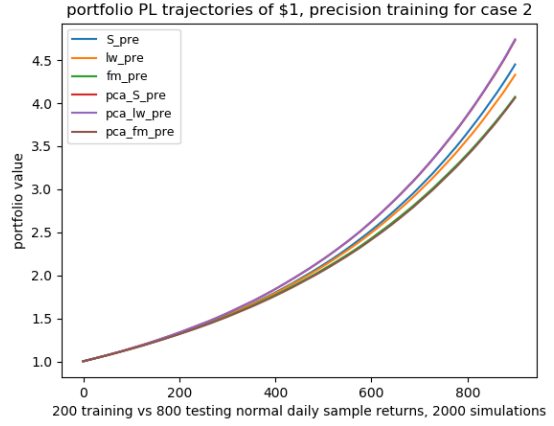
In Figure 2.7, we represent average simulated PL results using the model in Section 2.2 and the estimation methods described in Section 2.3. From 2000 independent training sets,

which include 200 i.i.d. normal returns, we estimated the average precisions of Sample (S), LW, and FM estimators with or without PCA model reduction. For the vector of expected returns and the target return, we used the real mean vector.

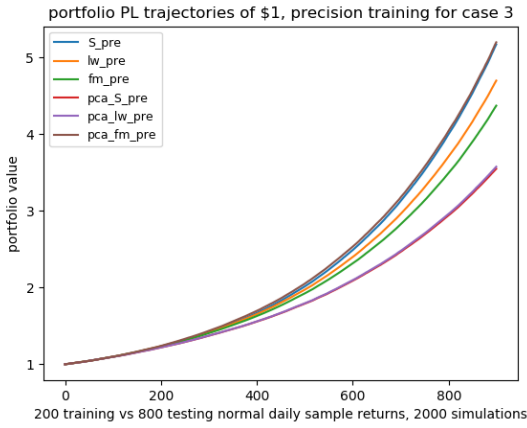
For case 1 in Figure 2.7a we see an improvement in the PL trajectory when we apply PCA to the Sample and LW, taking into account only the covariance eigenvalues greater than one. In all other cases, the results are essentially more similar to each other.



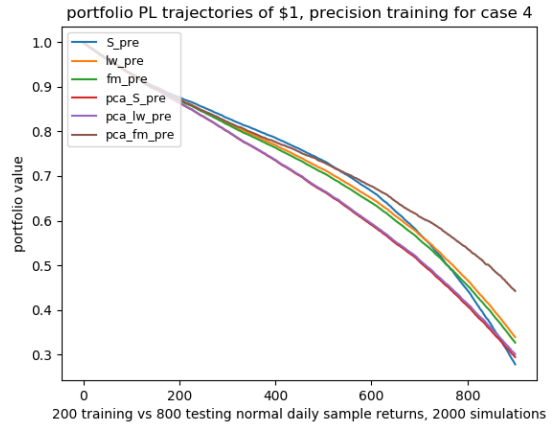
(a) Case 1: Bear Market



(b) Case 2: Bull Market

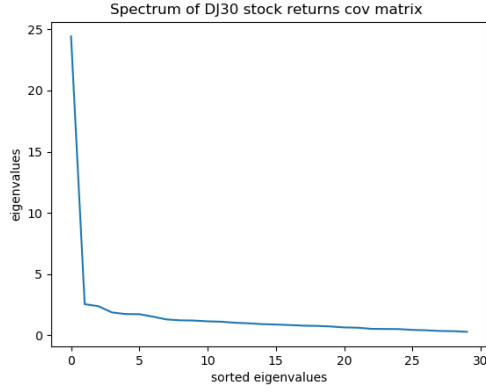


(c) Case 3: Bull Market



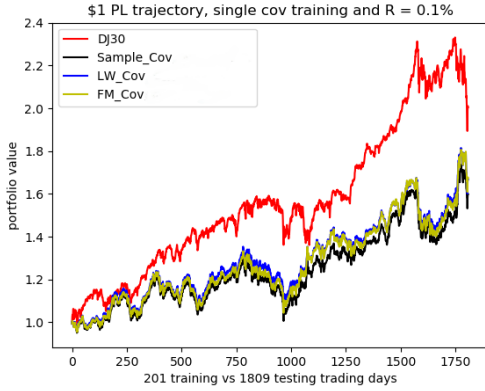
(d) Case 4: Strong Bear Market

Figure 2.7: Incremental portfolio loss (PL) for six precision estimators: Sample (S), LW, FM, and their respective PCA estimators when $\lambda_k > 1$. The dataset consists of 1000 returns drawn from a multivariate normal distribution, divided into training (200 samples) and testing (800 samples) sets. Results are averaged over 2000 simulations. Each estimator is used as a proxy to define the optimal allocation using Equation (2.4), and the incremental PL path is calculated according to Equation (2.14).

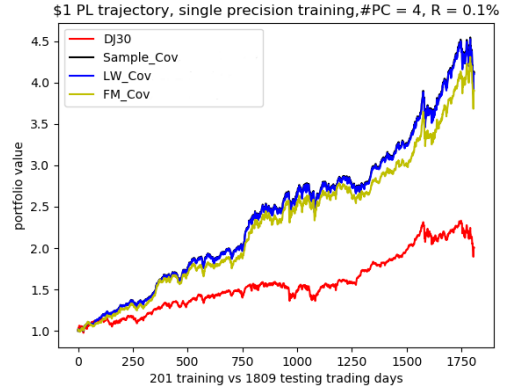


(a) DJ30 sorted Eigenvalues

Figure 2.8: DJ30 Covariance Eigenvalues



(a) DJ30 Covariance Estimators



(b) DJ30 Precisions Estimators 4 PC

Figure 2.9: DJ30 incremental PL

2.4.3 DJ30 Efficient Stock Portfolios PL

Before we discuss the results of a real data application, it is useful to get an intuition for the spectrum of the DJ30 stocks covariance matrix. Taking a close look into Figure 2.8a, we observe that only a few large eigenvalues play a role in the explanation of the system's variability. Such a property could explain why the precision matrix could be a good candidate for the MV model.

Figure 2.9a depicts the PL trajectories of 1 dollar invested in the efficient portfolio when the target return has been set equal to 0.1 percent and the optimal MV allocation has been calculated using the three different estimators of the covariance matrix: Sample (S), LW and FM. In contrast, Figure 2.9b represents the PL with PCA. We trained the covariances using the first 201 observations, and when we retrieved the optimal allocation, we substituted it into the PL functions to reproduce the trajectories of 1809 testing days. It is obvious from

the charts that the precision training estimators, including 4 principal components, definitely generate better trajectories than those of the full covariance estimators with better Sharpe ratios.

2.5 Discussion

Figure 2.6 reveals that the largest covariance eigenvalues of stock returns spike as the market moves from a bull to a bear market. We could take advantage of this property and create feature vectors including the largest covariance eigenvalues, with the aim of determining whether we are in a bull or bear market. Ridge Regression and SVM (Friedman et al. 2001) can be employed to produce the necessary classification. Investors could use these models to hedge their portfolio against future down-trends. For example, one could specify the Beta (percentage change of our portfolio when a market index changes by one percent) of his optimal portfolio, then take a short position in the futures of the underlying index (Hull 2003).

Chapter 3

Portfolio Optimization With Stochastic Volatility Models

3.1 General problem statement

In this Chapter, we focus on the 30 Dow Jones stocks and employ three different cost functions to build optimal portfolios, with and without boundary constraints on the stock weights. We estimate the covariance matrix of the time variant Σ_t , which is the main input in the cost functions, of the 30 stock returns and has dimensions 30×30 . The covariance matrix is estimated with two different classes of models, frequentist and Bayesian. For the frequentist cases, we employ the sample and the Ledoit & Wolf (2004a) covariance matrix, while for the Bayesian case we employ three models. The first Bayesian model assumes no correlation among stocks and estimates different volatilities via a scalar reparametrized SV model. The second Bayesian model is based on Xu & Jasra (2019) and the third on Kastner (2019). The training set consists of a three-month daily returns set for the 30 Dow Jones stocks, and the testing set consists of the consecutive 3 months period. The data set is chosen so as to be versatile, having both upward and downward price directions during the training and testing set. Splitting the six-month period into three-month training and testing sets, respectively, gives a reasonable time window for training the models reliably and also avoiding the cost of frequent re-balancing.

The main contribution of this chapter is that it offers a realistic and balanced approach to the solution of the portfolio optimization problem. Our approach shows to be able to derive attractive results in risk management and performance terms for real life applications, and also reveals the strengths and weaknesses of each selected model. In more detail, all the models' P&L and Sharpe Ratio perform consistently better than those of the Dow Jones Index during the testing period. The Bayesian models derive more diversified and evenly distributed portfolio weights than those of the frequentist ones. The Bayesian models give reliable credible intervals and distributions for the P&L and Sharpe Ratios. The frequentist models are computationally much cheaper to train, and in many cases perform better than the Bayesian ones in our testing set; however, they carry more unobserved risk due to more

extreme allocations and the lack of deriving credible intervals.

3.2 Cost functions

To assess the suitability of the described models in asset allocation, we need to employ objective functions that will provide the vector of the optimal portfolio allocation $\mathbf{w}_*$ of the portfolio. We denote as $\mathbf{w} = [w_1, \dots, w_d]^T$ an arbitrary vector of portfolio values such that $\sum_i w_i = 1$. We wish to optimize these weights under different cost functions that involve the mean and covariance of the total wealth process. These optimal portfolios will help us investigate the predicted performance for different candidate models via an out-of-sample data set. Here, we present the three different objective functions that we seek to optimize as follows:

$$\begin{aligned} \mathbf{w}_* &= \arg \min_{\mathbf{w}} \mathbf{w}^T \Sigma \mathbf{w} \\ \text{st: } \quad &\mathbf{w}^T \mathbf{1}_d = 1, \\ &w_i \in \mathbb{R} \end{aligned} \quad (3.1a)$$

$$\begin{aligned} \mathbf{w}_* &= \arg \min_{\mathbf{w}} \mathbf{w}^T \Sigma \mathbf{w} \\ \text{st: } \quad &\mathbf{w}^T \mathbf{1}_d = 1, \\ &0 \leq w_i \leq 0.1 \end{aligned} \quad (3.1b)$$

$$\begin{aligned} \mathbf{w}_* &= \arg \max_{\mathbf{w}} \mathbf{w}^T \boldsymbol{\mu} - \lambda \mathbf{w}^T \Sigma \mathbf{w} \\ \text{st: } \quad &\mathbf{w}^T \mathbf{1}_d = 1, \\ &w_i \in \mathbb{R} \end{aligned} \quad (3.2a)$$

$$\begin{aligned} \mathbf{w}_* &= \arg \max_{\mathbf{w}} \mathbf{w}^T \boldsymbol{\mu} - \lambda \mathbf{w}^T \Sigma \mathbf{w} \\ \text{st: } \quad &\mathbf{w}^T \mathbf{1}_d = 1, \\ &0 \leq w_i \leq 0.1 \end{aligned} \quad (3.2b)$$

$$\begin{aligned} \mathbf{w}_* &= \arg \min_{\mathbf{w}} \mathbf{w}^T \Sigma \mathbf{w} \\ \text{st: } \quad &\mathbf{w}^T \boldsymbol{\mu} = \bar{r}, \\ &\mathbf{w}^T \mathbf{1}_d = 1, \\ &w_i \in \mathbb{R} \end{aligned} \quad (3.3a)$$

$$\begin{aligned} \mathbf{w}_* &= \arg \min_{\mathbf{w}} \mathbf{w}^T \Sigma \mathbf{w} \\ \text{st: } \quad &\mathbf{w}^T \boldsymbol{\mu} = \bar{r}, \\ &\mathbf{w}^T \mathbf{1}_d = 1, \\ &0 \leq w_i \leq 0.1 \end{aligned} \quad (3.3b)$$

Here $\boldsymbol{\mu}$ and Σ denote the expected return vector of the assets after K time intervals in time and the corresponding covariance matrix respectively. Optimization problems (3.1a) and (3.3a) have closed-form solutions given respectively by:

$$\mathbf{w}_* = \frac{\Sigma^{-1} \mathbf{1}_d}{\mathbf{1}_d^T \Sigma^{-1} \mathbf{1}_d} \quad (3.4)$$

$$\begin{aligned} \mathbf{w}_* &= \frac{1}{ac - b^2} \Sigma^{-1} [(c - bR) \mathbf{1} + (aR - b) \boldsymbol{\mu}] \\ a &= \mathbf{1}^T \Sigma^{-1} \mathbf{1}, \quad b = \mathbf{1}^T \Sigma^{-1} \boldsymbol{\mu}, \quad c = \boldsymbol{\mu}^T \Sigma^{-1} \boldsymbol{\mu} \end{aligned}$$

while for equations (3.1b), (3.2a), (3.2b) and (3.3b) we use the Python package CVXPY (Diamond & Boyd 2016, Agrawal et al. 2018) to find the optimal $\mathbf{w}_*$. When $w_i \in \mathbb{R}$ short selling is allowed and when $w_i \in [0, 0.1]$ short selling is not allowed and also stocks cannot get weight higher than 10% of the portfolio value. The reason that we choose these two scenarios is that we want to see how extreme optimal allocations tend to be without constraints. This investigation is more realistic for practitioners, who are incentivized to diversify.

3.3 State space models and particle filtering

3.3.1 State space models

We are primarily interested in the prediction of a financial returns time series $\{\mathbf{y}_t\}$, where $t = 1, 2, \dots, T$. There are many possible parametric models used in finance and econometrics. In this report, we will focus on hidden Markov models (HMM), also known as state space models, whereby $\{\mathbf{y}_t\}$ is generated by a hidden latent Markov process of interest $\{\mathbf{x}_t\}$. We assume that the time series follows a particular state space model with parameter θ that adequately describes the evolution of the random variables in interest. These models are extremely popular in other scientific fields such as: Epidemiology, Biology, Engineering, Ecology and Telecommunications. The basic idea is that the first process, $\{\mathbf{x}_t\}$, is an unobserved Markov process, and the second one, $\{\mathbf{y}_t\}$, is directly observed (see Figure 3.1 for a graphical model) and each $\mathbf{y}_t$ conditional on $\mathbf{x}_t$ is independent from all other random variables. We assume that both $\mathbf{x}_t$ and $\mathbf{y}_t$ are real vectors, with $\mathbf{x}_t \in \mathbb{R}^{d_x}$ and $\mathbf{y}_t \in \mathbb{R}^d$, and we denote the HMM as follows:

$$\begin{aligned}\mathbf{x}_t &\sim f_\theta(\mathbf{x}_t|\mathbf{x}_{t-1}), \\ \mathbf{y}_t &\sim g_\theta(\mathbf{y}_t|\mathbf{x}_t).\end{aligned}$$

Where $f_\theta(\mathbf{x}_t|\mathbf{x}_{t-1})$ denotes the transition density of the hidden Markov process and $g_\theta(\mathbf{y}_t|\mathbf{x}_t)$ denotes the observation density. Finally, θ denotes the static vector of the model parameters, which are unknown.

3.3.2 Bayesian filtering

Just for this part suppose that θ is known. In a Bayesian context one goal is to compute the joint posterior distribution of all the volatilities given all the measurements. This can be done with the help of Bayes' rule

$$p_\theta(\mathbf{x}_{1:T}|\mathbf{y}_{1:T}) = \frac{\prod_{t=1}^T f_\theta(\mathbf{x}_t|\mathbf{x}_{t-1})g_\theta(\mathbf{y}_t|\mathbf{x}_t)}{p_\theta(\mathbf{y}_{1:T})} \quad (3.5)$$

where $\mathbf{x}_{1:T} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T)$, $\mathbf{y}_{1:T} = (\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_T)$. In equation (3.5) the product of $f_\theta(\mathbf{x}_t|\mathbf{x}_{t-1})$ acts as the prior density, while the product of $g_\theta(\mathbf{y}_t|\mathbf{x}_t)$ represents the likelihood

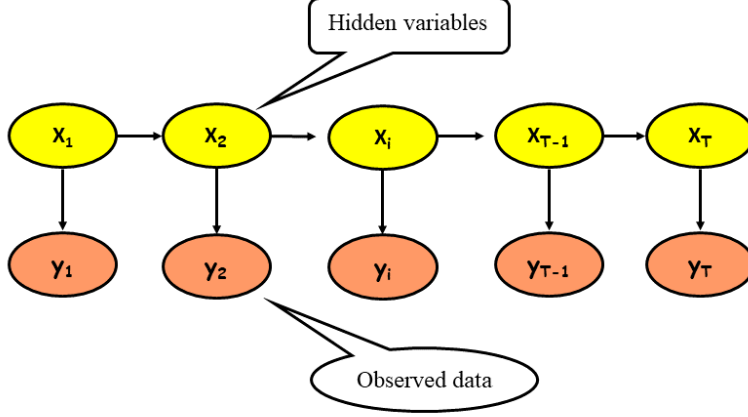


Figure 3.1: A graphical model for the Hidden Markov Model and volatility estimation

of the measurements. Finally, $p_\theta(\mathbf{y}_{1:T})$ represents the normalization constant, where

$$p_\theta(\mathbf{y}_{1:T}) = \int \prod_{t=1}^T f_\theta(\mathbf{x}_t | \mathbf{x}_{t-1}) g_\theta(\mathbf{y}_t | \mathbf{x}_t) d\mathbf{x}_{1:T}. \quad (3.6)$$

Bayesian Filtering refers to recursively estimating conditional probability distributions as new measurements are received, and this also enables prediction, which is the focus of this report. More specifically, we are interested in computing recursively the marginal posterior distribution $p_\theta(\mathbf{x}_t | \mathbf{y}_{1:t})$ of the state $\mathbf{x}_t$ at each time step t given the history of the measurements up to time t . At each time step t , we take the following steps:

Prediction step:

$$p_\theta(\mathbf{x}_t | \mathbf{y}_{1:t-1}) = \int p_\theta(\mathbf{x}_t | \mathbf{x}_{t-1}) f_\theta(\mathbf{x}_{t-1} | \mathbf{y}_{1:t-1}) d\mathbf{x}_{t-1} \quad (3.7)$$

Update step:

$$p_\theta(\mathbf{x}_t | \mathbf{y}_{1:t}) = \frac{1}{Z_t} g_\theta(\mathbf{y}_t | \mathbf{x}_t) p_\theta(\mathbf{x}_t | \mathbf{y}_{1:t-1}), \quad (3.8)$$

Normalization constant:

$$p_\theta(\mathbf{y}_t | \mathbf{y}_{1:t-1}) = \int g_\theta(\mathbf{y}_t | \mathbf{x}_t) p_\theta(\mathbf{x}_t | \mathbf{y}_{1:t-1}) d\mathbf{x}_t. \quad (3.9)$$

$$p_\theta(\mathbf{x}_{1:t}|\mathbf{y}_{1:t}) = p_\theta(\mathbf{x}_{1:t-1}|\mathbf{y}_{1:t-1}) \frac{g_\theta(\mathbf{y}_t|\mathbf{x}_t)f(\mathbf{x}_t|\mathbf{x}_{t-1})}{p_\theta(\mathbf{y}_t|\mathbf{y}_{1:t-1})}, \quad t = 1, 2, \dots, T \quad (3.10)$$

For linear Gaussian models, such as Kalman Filters, equation (3.5) can be computed analytically; however, for non-linear and non-Gaussian models, equation (3.5) is intractable, and we need to employ numerical methods using equation (3.10). Equation (3.10) is useful later when sequential importance sampling is used. Prediction of $\mathbf{x}_{T+K}$ and $\mathbf{y}_{T+K}$ can be trivially performed using Chapman-Kolmogorov recursions with $f(\mathbf{x}_t|\mathbf{x}_{t-1})$ and $g_\theta(\mathbf{y}_t|\mathbf{x}_t)$. Similar recursions are possible for the path space filtering problem that is interested in the conditional density of the whole trajectory $\mathbf{x}_{1:T}$.

3.3.3 Particle filters

We are interested in on-line numerical methods where we require that the computational complexity of a single time step should not increase as the number of time steps increases. Sequential Monte Carlo (SMC) methods, which are also known as Particle Filters (PF), are the state of the art of numerical approximations. One simulates samples that are approximately distributed according to the targeted posterior distribution $p_\theta(\mathbf{x}_{1:t}|\mathbf{y}_{1:t})$ for each time t , and also provides an approximation of the marginal distributions $p_\theta(\mathbf{x}_t|\mathbf{y}_{1:t})$ and the marginal likelihood $p_\theta(\mathbf{y}_{1:t})$. SMC propagates samples, named also as particles, through a sequence of posterior distributions and incorporates two well-known algorithms, sequential importance sampling and resampling. This is referred to as sequential importance resampling (SIR), Algorithm 1.

Sequential importance sampling (SIS): We target densities of increasing dimension $\{p(\mathbf{x}_{1:t}|\mathbf{y}_{1:t})\}_{t \geq 1}$. SIS avoids the increasing computational complexity as time increases by applying IS recursively where the proposal density follows the recursive structure: $q(\mathbf{x}_{1:t}) = q(\mathbf{x}_1) \prod_{k=2}^t q(\mathbf{x}_k|\mathbf{x}_{1:k-1})$. Using the notation of equation (3.10), for each time step we can calculate the current weights as follows:

$$w(\mathbf{x}_{1:t}^i) = w(\mathbf{x}_{1:t-1}^i) \frac{g_\theta(\mathbf{y}_t|\mathbf{x}_t^i)f_\theta(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)}{q_\theta(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i, \mathbf{y}_{1:t})} \quad (3.11)$$

Resampling: In order to avoid carrying forward particles with low weights, at each time step we can resample N particles from the existing sampled particles. Then the normalised weight $W(\mathbf{x}_{1:t}^i)$ corresponds to the probability of resampling the particle $\mathbf{x}_{1:t}^i$, and in this way we render N new equally-weighted particles $\bar{\mathbf{x}}_{1:t}^i$. At time t , the respective approximations of $p(\mathbf{x}_{1:t}|\mathbf{y}_{1:t})$ and $p(\mathbf{y}_t|\mathbf{y}_{1:t-1})$ after the resampling step are:

$$\hat{p}(d\mathbf{x}_{1:t}|\mathbf{y}_{1:t}) = \frac{1}{N} \sum_{i=1}^N \delta_{\bar{\mathbf{x}}_{1:t}^i}(d\mathbf{x}_{1:t}), \quad (3.12)$$

$$\hat{p}(\mathbf{y}_{1:T}|\theta) = \prod_{t=1}^T \frac{1}{N} \sum_{i=1}^N w_t(\mathbf{x}_{1:t-1}^i, \mathbf{x}_{1:t}^i), \quad (3.13)$$

Algorithm 1 Sequential Importance Resampling

At time $t = 1$

1. Draw N samples from the importance distribution:

$$\mathbf{x}_1^i \sim q_\theta(\mathbf{x}_1), \quad i = 1, 2, \dots, N$$

2. Compute the weights $w_\theta(\mathbf{x}_1^i)$ and normalize weights $W_\theta(\mathbf{x}_1^i) = \frac{w_\theta(\mathbf{x}_1^i)}{\sum_{j=1}^N w_\theta(\mathbf{x}_1^j)}$

3. Resample $\{W_\theta(\mathbf{x}_1^i), \mathbf{x}_1^i\}$ to obtain N equally-weighted particles: $\{\frac{1}{N}, \bar{\mathbf{x}}_1^i\}$

At time $t \geq 2$

1. Draw N samples from the importance distribution:

$$\mathbf{x}_t^i \sim q_\theta(\mathbf{x}_t | \bar{\mathbf{x}}_{1:t-1}^i, \mathbf{y}_{t-1}), \quad i = 1, 2, \dots, N$$

2. Compute the weights $w_\theta(\mathbf{x}_t^i)$ and normalize weights $W_\theta(\mathbf{x}_{1:t}^i) = \frac{w_\theta(\mathbf{x}_{1:t}^i)}{\sum_{j=1}^N w_\theta(\mathbf{x}_{1:t}^j)}$

3. Resample $\{W_\theta(\mathbf{x}_t^i), \mathbf{x}_t^i\}$ to obtain N equally-weighted particles: $\{\frac{1}{N}, \bar{\mathbf{x}}_{1:t}^i\}$
-

where here $\delta_{\mathbf{x}'}(d\mathbf{x})$ denotes the Dirac measure, such that $\delta_{\mathbf{x}'}(A) = 1_{\mathbf{x}' \in A}$.

The convergence properties of the method have been studied extensively (Cappé et al. 2005, Del Moral et al. 2012), we mention indicative results. Let φ be a bounded measurable test function and let also the integral of interest to be:

$$I_T = \int \varphi(\mathbf{x}_{1:T}) p_\theta(\mathbf{x}_{1:T} | \mathbf{y}_{1:T}) d\mathbf{x}_{1:T}, \quad (3.14)$$

then the SIS particle approximation of I_T will be:

$$\hat{I}_T = \int \hat{\varphi}(\mathbf{x}_{1:T}) p_\theta(\mathbf{x}_{1:T} | \mathbf{y}_{1:T}) d\mathbf{x}_{1:T} = \frac{1}{N} \sum_{i=1}^N \varphi(\bar{\mathbf{x}}_{1:T}^i). \quad (3.15)$$

One can show that $\hat{I}_T \xrightarrow{\text{a.s.}} I_T$ and a Central Limit Theorem holds:

$$\sqrt{N}(\hat{I}_T - I_T) \xrightarrow{d} N(0, \sigma_{\varphi, T}^2). \quad (3.16)$$

In addition we have the following L_p bound:

$$\mathbb{E}[|\hat{I}_T - I_T|]^{1/p} \leq \frac{c}{\sqrt{N}}, \quad (3.17)$$

where c is a constant that depends on the model itself and its dimensionality. Note that if the HMM has an exponential forgetting property this constant is uniform in time, in terms of

non-asymptotic results, for the marginal likelihood we have. If one sets a particular threshold for this Monte Carlo variance, then she should set N proportional to T :

$$\text{Var}\left(\frac{\widehat{p}(\mathbf{y}_{1:T}|\theta)}{p(\mathbf{y}_{1:T}|\theta)}\right) \leq c(d) \frac{T}{N} \quad (3.18)$$

This is important in the context of Particle MCMC later.

3.3.4 Bayesian parameter estimation and predictive posteria

Applying Algorithm 1 requires not only the data $\mathbf{y}_t$ but also knowledge of θ , which in most practical situations needs to be estimated from the data itself. One could use different inferential procedures from principles such as maximum likelihood estimation or Bayesian inference. In this report, we will only use Bayesian inference because it is easier to characterise the uncertainty of predictive posteriors and use state-of-the-art algorithms such as Particle Markov Chain Monte Carlo (PMCMC) (Andrieu et al. 2010).

Bayesian inference treats unknown parameter vector θ as a random variable with a defined prior distribution $p(\theta)$ based on our prior beliefs. More specifically, we want to estimate the posterior distribution $p(\theta|\mathbf{y}_{1:T})$, which can be computed by:

$$p(\theta|\mathbf{y}_{1:T}) \propto p(\mathbf{y}_{1:T}|\theta)p(\theta). \quad (3.19)$$

Algorithm 2 PMCMC

At iteration $k = 1$

- *Draw $\theta[1] \sim p(\theta)$*
- *Run a SMC algorithm targeting $p(\mathbf{x}_{1:T}, \theta|\mathbf{y}_{1:T})$, sampling N particles $\mathbf{x}[k]_{1:T} \sim \widehat{p}(d\mathbf{x}_{1:T}|\mathbf{y}_{1:T}, \theta[1])$ and compute marginal likelihood $\widehat{p}(\mathbf{y}_{1:T}|\theta[1])$*

At iteration $k \geq 2$

- *Sample a proposal $\theta' \sim q(\theta|\theta[k-1])$*
 - *Run a SMC algorithm targeting $p(\mathbf{x}_{1:T}, \theta'|\mathbf{y}_{1:T})$, sampling N particles $\mathbf{x}[k]_{1:T} \sim \widehat{p}(d\mathbf{x}_{1:T}|\mathbf{y}_{1:T}, \theta')$ and compute marginal likelihood $\widehat{p}(\mathbf{y}_{1:T}|\theta')$*
 - *Set $\theta[k] = \theta'$ with probability: $1 \wedge \frac{\widehat{p}(\mathbf{y}_{1:T}|\theta')p(\theta')q(\theta[k-1]|\theta')}{\widehat{p}(\mathbf{y}_{1:T}|\theta[k-1])p(\theta[k-1])q(\theta'|\theta[k-1])}$, otherwise set $\theta[k] = \theta[k-1]$*
-

The marginal likelihood $p(\mathbf{y}_{1:T}|\theta)$ is intractable, so one needs to resort to approximations, see Algorithm 2. In this report, we use Particle Marginal Metropolis Hastings (PMMH) to estimate the unknown vector θ . These are pseudo-marginal MCMC methods (Andrieu & Roberts 2009), which substitute $\widehat{p}(\mathbf{y}_{1:T}|\theta)$ in a Metropolis Hastings procedure targeting (3.19). This is presented in Algorithm 2, where each MCMC iteration runs an SMC algorithm. It is remarkable that this algorithm is marginally invariant to $p(\theta|\mathbf{y}_{1:T})$. In short,

the algorithm is a M-H algorithm targeting the joint distribution of θ and all the random variables used to generate the particle filter conditional on $\mathbf{y}_{1:T}$.

Predictive posterior approximations based on PMMH

Investors wish to predict the value of the assets K time steps forward after time T , so that we can approximate $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ in (3.1)-(3.3) and thus design optimal portfolios. This information is contained in the predictive posterior $p(\mathbf{y}_{T+K}|\mathbf{y}_{1:T})$, which in turn can be derived from $p(\theta|\mathbf{y}_{1:T})$ as follows:

$$p(\mathbf{y}_{T+K}|\mathbf{y}_{1:T}) = \int p(\mathbf{y}_{T+1:T+K}|\mathbf{y}_{1:T})d\mathbf{y}_{T+1:T+K-1} \quad (3.20)$$

$$p(\mathbf{y}_{T+K}|\mathbf{y}_{1:T}) = \int \prod_{l=T+1}^K f_{\theta}(\mathbf{x}_l|\mathbf{x}_{l-1})g_{\theta}(\mathbf{y}_l|\mathbf{x}_l)p(\mathbf{x}_T, \theta|\mathbf{y}_{1:T})d\mathbf{x}_{T:T+K}d\mathbf{y}_{T+1:T+K-1}d\theta. \quad (3.21)$$

We are interested to approximate the above quantity as a Monte Carlo integral by:

$$\widehat{p}(\mathbf{y}_{T+1:T+K}|\mathbf{y}_{1:T}) = \frac{1}{P} \sum_{p=1}^P \delta_{\mathbf{y}_{T+1:T+K}^p} \quad (3.22)$$

where one obtains the required samples i.i.d. (in p) as follows:

$$(\theta^p, \mathbf{x}_T^p) \sim \frac{1}{M} \sum_{m=1}^M \delta_{(\mathbf{x}_T[m], \theta[m])}, \quad \mathbf{x}_t^p \sim f(\mathbf{x}_{t-1}^p, \cdot), \quad \mathbf{y}_t^p \sim f(\mathbf{y}_t^p, \cdot), \quad (3.23)$$

where in (3.23) we are using the output of the MCMC samples $(\theta_K, \mathbf{x}_T)_{k=1, \dots, M}$ from Algorithm 2. We will use (3.22) to calculate empirical predicted expected returns $\widehat{\mu}_{T+K}$ and covariances $\widehat{\Sigma}_{T+K}$ of the optimal portfolios generated according to (3.1)-(3.3).

Monte Carlo covariance estimator for Bayesian models

To estimate the credible intervals of the optimal portfolio $\mathbf{w}_*$ in terms of P&L and annualized Sharpe Ratio, we work as follows:

- From (3.1)-(3.3), we first choose an objective function which we want to optimize with respect to the optimal allocation $\mathbf{w}_*$.
- We calculate the estimators $\widehat{\Sigma}_{T+K}$ and $\widehat{\mu}_{T+K}$ and we plug it in the optimization problem and retrieve the optimal allocation $\mathbf{w}_*$.
- Based on the $\mathbf{w}_*$ and the simulated log-returns $\mathbf{y}_t^i$, we can reproduce respectively a cloud of portfolio value trajectories, also known as profit and loss (P&L), when \$1 is invested and render the relative Sharpe Ratio performance measures.

- The expected optimal portfolio variance and mean can be estimated by:

$$\sigma_p^2 = \mathbf{w}_*^T \widehat{\Sigma}_{T+K} \mathbf{w}_*, \quad \mu_p = \mathbf{w}_*^T \widehat{\mu}_{T+K} \quad (3.24)$$

From now on we replace $\mathbf{w}_*$ with $\mathbf{w}_{T+k}$.

3.4 Models

We proceed with a brief overview of the different stochastic volatility models that we will use. The first one is a naive extension of a scalar model to the multivariate case, by assuming all assets are independent, and we call this independent stochastic volatility (ISV) model. The second one is a modification of a recent multivariate SV (MSV) model introduced by Xu & Jasra (2019) that allows for a general covariance structure using common persistence and cross-leverage effects. The third model is another recent approach by Kastner (2019) based on shrinkage ideas. The last two models were selected on their reported performance, their interpretability and generality. Other models could be added to this list (e.g. Chan et al. (2018), Plataniotis & Dellaportas (2010), Dellaportas et al. (2021)), but the methods chosen here are easier to interpret from a practitioner's point of view. The ISV is merely used as a naive benchmark, so that one can assess the benefit of cross-covariance terms in portfolio optimisation.

The selection of numerical methods in this chapter is guided by the trade-off between accuracy, scalability, and interpretability. PMCMC was chosen for the ISV and MSV models due to its capacity to jointly estimate static parameters and latent states in complex, non-linear time-series models. Particularly for the MSV model, PMCMC enables posterior inference in high dimensions while maintaining stability under particle degeneracy through adaptive proposals and transformation-based sampling. In contrast, for the Factor Model, we utilized the factorstochvol package (Hosszejni & Kastner 2019) due to its robust hierarchical shrinkage priors and computationally efficient implementation tailored to factorized covariance structures. These methods were selected to ensure tractable and interpretable posterior estimation across models with varying complexity.

3.4.1 An Independent Stochastic Volatility (ISV) model

We begin with a scalar discrete-time SV, first introduced by Taylor (1986, 1994) and investigated among others such as Harvey & Shephard (1996), Jacquier et al. (2004), Meyer & Yu (2000) and Yu (2005). The model is represented as follows:

$$Y_t = r + \beta \exp(X_t/2) W_t, \quad (3.25)$$

$$X_{t+1} = \alpha X_t + \sigma V_{t+1}, \quad (3.26)$$

$$E(W_t V_{t+1}) = \rho, \quad (3.27)$$

$$V_t, W_t \stackrel{iid}{\sim} N(0, 1), \quad t = 1, 2, \dots, T \quad (3.28)$$

where $X_t, Y_t \in \mathbb{R}$ and $\theta = (\alpha, \beta, \sigma, r, \rho)$. Here, X_t represents the logarithm of the asset's underlying variance for one period and forms a hidden random sequence with $X_1 \sim N(0, \frac{\sigma^2}{1 - \alpha^2})$. Y_t represents the asset's log return for the same period and it provides the observations sequence in the data. Between the disturbances W_t and V_{t+1} , we assume that there exists a correlation ρ , which means that a contemporaneous dependence is allowed in the SV model. The β parameter is a scale factor, which subsumes the effect of the constant term in the auto-regression of X_t . In contrast to large parts of the literature, we include an additional parameter r , which represents the stock's constant rate of change or drift term in one period of time. In the literature, often $r = 0$ is used as a common modeling assumption. Whilst this can make sense financially in some cases, here we impose estimation of r from the data, as it will provide better predictions of short and medium returns that are useful for portfolio optimisation.

The correlation ρ between the disturbances helps the model to pick up the asymmetric pattern often described in stock prices. In simple terms, an increase in predicted volatility tends to induce a fall in the stock price, suggesting a negative correlation between the disturbances of stock returns and volatility, respectively. This tendency is known in the literature as the financial leverage effect. Hull & White (1987) has emphasized the crucial role of this correlation in stock option pricing. To obtain a more convenient parametrisation, we will use the transformation:

$$Z_{t+1} = \frac{V_{t+1}}{\sqrt{1 - \rho^2}} - \frac{\rho W_t}{\sqrt{1 - \rho^2}}, \quad E(W_t Z_{t+1}) = 0. \quad (3.29)$$

Solving for V_{t+1} gives:

$$V_{t+1} = \rho W_t + \sqrt{1 - \rho^2} Z_{t+1} \quad (3.30)$$

and now the SV model can be re-parametrised as follows:

$$Y_t = r + \beta \exp(X_t/2) W_t, \quad (3.31)$$

$$X_{t+1} = \alpha X_t + \frac{\rho \sigma}{\beta} (Y_t - r) \exp(-X_t/2) + \sigma \sqrt{1 - \rho^2} Z_{t+1}, \quad (3.32)$$

$$E(W_t Z_{t+1}) = 0, \quad (3.33)$$

$$V_t, Z_t \stackrel{iid}{\sim} N(0, 1), \quad t = 1, 2, \dots, T \quad (3.34)$$

This parametrization of the equation facilitates the introduction of ρ in the PMCMC and provides good mixing in MCMC, as we will see later.

3.4.2 A Multivariate Stochastic Volatility (MSV) model

We proceed with the multivariate extension of SV models by Xu & Jasra (2019):

$$\mathbf{y}_t = \mathbf{V}_t^{1/2} \epsilon_t \quad t = 1, 2, \dots, T, \quad (3.35)$$

$$\mathbf{x}_{t+1} = \mathbf{\Phi} \mathbf{x}_t + \eta_t \quad t = 1, 2, \dots, T - 1 \quad (3.36)$$

where

$$\mathbf{V}_t = \begin{pmatrix} e^{X_{1t}} & & \\ & \ddots & \\ & & e^{X_{dt}} \end{pmatrix}, \quad \mathbf{\Phi} = \begin{pmatrix} \phi_1 & & \\ & \ddots & \\ & & \phi_d \end{pmatrix}$$

$$\begin{pmatrix} \epsilon_t \\ \eta_t \end{pmatrix} \sim N_{2d}(\mathbf{0}, \mathbf{\Sigma}), \quad \mathbf{x}_1 \sim N_d(\mathbf{0}, \mathbf{\Sigma}_0).$$

Here $\mathbf{y}_t \in R^d$ is the log-returns vector, with d being the number of underlying stocks, and $\mathbf{x}_{t+1} \in R^d$ the log-variance vector, which is unobservable. $\mathbf{V}_t$ and $\mathbf{\Phi}$ are diagonal matrices and denote the underlying volatility of each stock and the persistence of each variant respectively. Finally, ϵ_t and η_t denote zero mean Gaussian noise variables with covariance $\mathbf{\Sigma}$, which can be partitioned as follows:

$$\mathbf{\Sigma} = \begin{pmatrix} \mathbf{\Sigma}_{\epsilon\epsilon} & \mathbf{\Sigma}_{\epsilon\eta} \\ \mathbf{\Sigma}_{\eta\epsilon} & \mathbf{\Sigma}_{\eta\eta} \end{pmatrix}, \quad \mathbf{\Sigma}_{\eta\epsilon} = \mathbf{\Sigma}_{\epsilon\eta}^T.$$

$\mathbf{\Sigma}_{\epsilon\epsilon}$ and $\mathbf{\Sigma}_{\eta\eta}$ denote the marginal covariance matrices of the disturbances ϵ_t and η_t respectively, and the nonzero diagonal and off-diagonal elements of $\mathbf{\Sigma}_{\epsilon\eta}$ indicate the existence of leverage and cross-leverage effects respectively. In other words, the $(\mathbf{\Sigma}_{\epsilon\eta})_{ij}$ element shows the covariance between the i^{th} stock return at time t and the future volatility of the j^{th} stock at time $t + 1$. Hence, cross-leverage effects describe possible interactions between different stock log-returns time series. Finally, the vector $\mathbf{x}_1$ has covariance matrix $\mathbf{\Sigma}_0$, where:

$$(\mathbf{\Sigma}_0)_{ij} = \frac{(\mathbf{\Sigma}_{\eta\eta})_{ij}}{1 - \phi_i \phi_j} \quad (3.37)$$

Recall, our goal is to estimate the posterior distribution $p(\theta|\mathbf{y}_{1:T})$ of parameter $\theta = (\mathbf{\Phi}, \mathbf{\Sigma})$ given the model of equation (3.35) and the observed stock log-return $\mathbf{y}_{1:T}$. As before, the marginal likelihood is intractable and the SMC algorithm has to be employed. From equation (3.35) we have:

$$(\mathbf{y}_t, \mathbf{x}_{t+1}|\mathbf{x}_t) \sim N_{2d} \left(\begin{pmatrix} \mathbf{0} \\ \mathbf{\Phi}\mathbf{x}_t \end{pmatrix}, \begin{pmatrix} \mathbf{V}_t^{1/2}\mathbf{\Sigma}_{\epsilon\epsilon}\mathbf{V}_t^{1/2} & \mathbf{V}_t^{1/2}\mathbf{\Sigma}_{\epsilon\eta} \\ \mathbf{\Sigma}_{\eta\epsilon}\mathbf{V}_t^{1/2} & \mathbf{\Sigma}_{\eta\eta} \end{pmatrix} \right) \quad (3.38)$$

In the interest of applying the particle filter, we present the transition density $f(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{y}_{t-1})$ and the measurement density $g(\mathbf{y}_t|\mathbf{x}_t)$ for MSV respectively as follows:

$$\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{y}_{t-1} \sim N_d(\mathbf{\Phi}\mathbf{x}_{t-1} + \mathbf{\Sigma}_{\eta\epsilon}\mathbf{\Sigma}_{\epsilon\epsilon}^{-1}\mathbf{V}_{t-1}^{-1/2}\mathbf{y}_{t-1}, \mathbf{\Sigma}_{\eta\eta} + \mathbf{\Sigma}_{\eta\epsilon}\mathbf{\Sigma}_{\epsilon\epsilon}^{-1}\mathbf{\Sigma}_{\epsilon\eta}) \quad (3.39)$$

$$\mathbf{y}_t|\mathbf{x}_t \sim N_d(\mathbf{0}, \mathbf{V}_t^{1/2}\boldsymbol{\Sigma}_{\epsilon\epsilon}\mathbf{V}_t^{1/2}) \quad (3.40)$$

Algorithm 2 works very well for the parameter estimation of the MSV model; however, as the dimension d increases, the algorithm is susceptible to variance issues in estimating $p(\mathbf{y}_{1:T}|\theta)$. To deal with this problem, it is important to sample a sufficiently large number of particles N .

In the acceptance ratio there is randomness due to the presence of an estimator for $\widehat{p}(\mathbf{y}_{1:T}|\theta')$. If at some iteration this likelihood is overestimated this will slow down the mixing of the chain. This improves as the number of particles N gets larger, but one needs to be prudent how much N is increased especially when a fixed total computational budget is available, however, high N means fewer MCMC iterations. As time T increases N increases linearly (total computational cost is proportional to N^2) In our case where $d = 30$, we set the number of particles $N = 3000$.

3.4.3 A Factor Model (FM) approach

Finally, the last model to be investigated is the Factor Model by Kastner (2019). Consider $\mathbf{y}_t \in R^d$ has a conditional Normal distribution such that:

$$\mathbf{y}_t|\boldsymbol{\Sigma}_t \sim N_d(\mathbf{0}, \boldsymbol{\Sigma}_t), \quad (3.41)$$

where $\boldsymbol{\Sigma}_t$ contains $d(d+1)/2$ free elements to be estimated. One can regress $\mathbf{y}_t$ on a vector of factors $\mathbf{f}_t$ as follows:

$$\mathbf{y}_t = \mathbf{\Lambda}\mathbf{f}_t + \epsilon_t, \quad \mathbf{f}_t \sim N_r(\mathbf{0}, \mathbf{V}_t), \quad \epsilon_t \sim N_d(\mathbf{0}, \mathbf{U}_t), \quad (3.42)$$

where $\mathbf{\Lambda}$ is a $d \times r$ time invariant matrix and $\mathbf{V}_t, \mathbf{U}_t$ are time variant diagonal matrices of r and d dimensions respectively. These two diagonal matrices are evolved over time through parametric SV models as follows:

$$\mathbf{V}_t = \text{diag}(\exp(h_{1t}), \dots, \exp(h_{rt})) \quad \mathbf{U}_t = \text{diag}(\exp(h_{1t}), \dots, \exp(h_{dt})) \quad (3.43)$$

where h_{it} and h_{jt} represent respectively hidden stock log-variances and factor log-variances and follow the AR(1) processes as follows:

$$\begin{aligned} h_{it} &\sim N(\mu_i + \phi(h_{i,t-1} - \mu_i, \sigma_i^2) \quad i = 1, 2, \dots, d \\ h_{jt} &\sim N(\phi(h_{j,t-1}, \sigma_j^2) \quad j = 1, 2, \dots, r \end{aligned}$$

Following equation (3.42), $\boldsymbol{\Sigma}_t$ can be decomposed as:

$$\boldsymbol{\Sigma}_t = \mathbf{\Lambda}\mathbf{V}_t\mathbf{\Lambda}^T + \mathbf{U}_t. \quad (3.44)$$

The above decomposition induces parameter parsimony and reduces the amount of free parameters from $d(d+1)/2$ to $dr + d + r$, since the r is chosen to be much smaller than d . Furthermore, to get rid of the unimportant factors for each stock and thus introduce

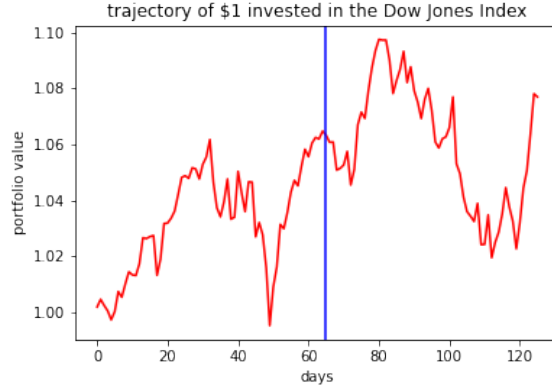


Figure 3.2: Trajectory of \$1 invested in Dow Jones 30 index for the time period 03/01/2011 - 01/07/2011. Blue vertical line indicates the split of the time into training and testing set respectively.

sparsity in $\mathbf{\Lambda}$ matrix, Kastner (2019) uses a hierarchical Normal-Gamma prior (Griffin & Brown 2017), (Huber & Feldkircher 2019) to the Λ_{ij} elements:

$$\Lambda_{ij} \sim N(0, \tau_{ij}^2), \quad \tau_{ij}^2 | \lambda_i^2 \sim G(\alpha_i, \alpha_i \lambda_i^2), \quad \lambda_i^2 \sim G(c_i, d_i) \quad (3.45)$$

The hyperparameters α_i , c_i and d_i are fixed a priori and α_i especially determines the amount of shrinkage. The smaller α_i is, the stronger the shrinkage towards zero, and the opposite. Finally, for the stock and factor hidden AR(1) processes the set of parameters μ_i , $(\phi_i + 1)/2$ and σ_i^2 have priors $N(b_\mu, B_\mu)$, $\mathcal{B}(\alpha_0, b_0)$ and $G(0.5, 1/(2B_\sigma))$ respectively.

3.5 Application to Dow Jones Industrial Average 30 data

We choose a set of thirty stocks ($d = 30$), which are the constituents of the Dow Jones Index. We collect from Yahoo Finance the adjusted stock prices of the 30 stocks for the period ranging from 03/01/2011 to 01/07/2011; these stock prices correspond to 126 daily log-returns. The returns are defined by the log difference of the consecutive-day stock prices. We treat the first 63 days as the training set and the remaining 63 as the testing set. To get an idea of how the Dow-Jones index has moved over the 126 days, we depict in Figure 3.2 the trajectory of \$1 invested in the index. Below we present implementation details and estimation results separately for each model.

3.5.1 MCMC estimation

For positive or constrained parameters in $[-1, 1]$ the proposals we use are transformed random walks in log or logit scale, so we have to adjust the acceptance ratio for the Jacobian

accordingly. For the log scale, the sampler because of changing variables becomes:

$$\theta' = \exp(Z), \quad (3.46)$$

$$Z = \log(\theta) + s\epsilon, \quad \epsilon \sim N(0, 1), \quad (3.47)$$

leading to an acceptance ratio:

$$\frac{\widehat{p}(\mathbf{y}_{1:T}|\theta')p(\theta')\theta'}{\widehat{p}(\mathbf{y}_{1:T}|\theta)p(\theta)\theta}. \quad (3.48)$$

Similarly, for the logit scale, the proposal is given:

$$\theta' = \frac{\exp(Z) - 1}{\exp(Z) + 1}, \quad (3.49)$$

$$Z = \log(\theta + 1) + \log(1 - \theta) + s\epsilon, \quad \epsilon \sim N(0, 1). \quad (3.50)$$

while the acceptance ratio is:

$$\frac{\widehat{p}(\mathbf{y}_{1:T}|\theta')p(\theta')(\theta' + 1)(1 - \theta')}{\widehat{p}(\mathbf{y}_{1:T}|\theta)p(\theta)(\theta + 1)(1 - \theta)}. \quad (3.51)$$

The variable s in equations (3.46) and (3.50) denotes the step size, and for all the estimated parameters, s has been set in the range between 0.2 and 0.5.

To estimate the parameter vector $\theta = (r, \beta, \alpha, \sigma, \rho)^T$ for the scalar SV model, we apply the Algorithm 2 as described above. For the parameter α , r and ρ we use uniform prior $Unif(-1, 1)$ and the proposals are chosen to be random walks on the logit scale. For β and σ we use $Exp(0.5)$ and $IG(1, 1)$ priors respectively and the proposals are chosen to be random walks on the logit scale. For each iteration of the PMCMC we compute the marginal likelihood according to SIR Algorithm 1. We note that we choose proposal density $q_\theta(x_t^i|x_{t-1}^i, y_{t-1}) = f_\theta(x_t^i|x_{t-1}^i, y_{t-1})$ with $w_t(x_t^i) = g_\theta(y_t|x_t^i)$.

Applying Algorithm 2 in the training set, we train the ISV model for each stock independently. We set the number of iterations $k = 25000$ (see Algorithm 2) and we retrieve the MCMC chains for θ . In the Appendix, at Figure C.1, we represent indicatively the MCMC chain of θ for the Apple stock after 5000 burn-in. In this case, the MCMC acceptance ratio is 22.86%, and for the majority of the underlying stocks, the MCMC acceptance ratio lies around 23.00%. In Figure C.2, we represent the charts of the autocorrelations for the MCMC chains of θ . We observe that after a time-lag of 50, the autocorrelation falls below 20%. From the MCMC chains, we can easily retrieve the marginal posterior distributions of $p(\theta|\mathbf{y}_{1:T})$, which are represented in Figure C.3. In the posterior distributions, we include the MCMC chains after 5000 burn-in iterations. In each histogram, we indicate the mode and the mean, respectively.

For the MSV model, we apply again Algorithm 2 to estimate the parameter vector $\theta = (\sigma_\epsilon^2, \sigma_\eta^2, (\Sigma_{\epsilon\epsilon})_{ij}, (\Sigma_{\eta\eta})_{ij}, (\Sigma_{\epsilon\eta})_{ii}, (\Sigma_{\epsilon\eta})_{ij}, \alpha)^T$. For the scalar parameters $\sigma_\epsilon^2, \sigma_\eta^2$ we use $IG(2, 1)$ prior and the proposals are chosen to be random walks on the log scale. The covariances

$(\Sigma_{\epsilon\epsilon})_{ij}, (\Sigma_{\eta\eta})_{ij}, (\Sigma_{\epsilon\eta})_{ii}, (\Sigma_{\epsilon\eta})_{ij}$ are calculated indirectly via the relative correlations, which follow uniform prior $Unif(-1, 1)$ and the proposals are chosen to be random walks on the logit scale. Again as above, we choose proposal density $q_\theta(x_t^i|x_{t-1}^i, y_{t-1}) = f_\theta(x_t^i|x_{t-1}^i, y_{t-1})$ with $w_t(x_t^i) = g_\theta(y_t|x_t^i)$.

For the MSV model, we apply the Algorithm 2 for all the stocks simultaneously. The MCMC chains for the parameter vector θ with the relative posterior histograms are cited in the Appendix at Figure C.7 and C.8 respectively. Here, the number of MCMC iterations is $k = 30000$ after 5000 burn-in, and the acceptance ratio is 28%.

For the FM, we use one factor $r = 1$ and to train the model we use the R package ‘factorstochvol’ (Hosszejni & Kastner 2019), where the hyperparameters are set as follows: $\alpha_i = 0.1, c_i = 1, d_i = 1, b_\mu = 0, B_\mu = 100, \alpha_0 = 10, b_0 = 3, B_\sigma = 1$.

Finally for the FM model at Figure C.9, we indicatively represent the MCMC trace plots and the relative histograms of the factor loading, the variance σ^2 of the hidden variable h_t and the persistence ϕ for the Bank of America stock. For this model, we set $k = 110000$ with a 10000 burn-in and for the remaining 100000 draws every 10th draw is kept, resulting in 10000 draws used in the posterior histograms. As a general comment, we observe very good mixing for θ MCMC draws for each model.

3.5.2 Predicted credible intervals and realised PL

Following the steps of Section 3.3.4 after the MCMC results, we can now predict the credible intervals (CI) of the optimal portfolio $\mathbf{w}_{T+K}$. From the CI, we can estimate additional features, such as the distributions of the optimal portfolio terminal values PL_{T+K} and the annualized Sharpe Ratios SR_{T+K} . We generated a P&L cloud of 1000 different P&L trajectories for the optimal allocation $\mathbf{w}_{T+K}$. For the MSV model, the relative P&L clouds are depicted in Figure 3.3, while for ISV and FM models the relative Figures, C.4 and C.10, are placed in Appendix Sections C.1.4 and C.3.2, respectively. Additionally to CIs, in these Figures we also depict the realized P&L trajectories of $\mathbf{w}_{T+K}$ portfolio and the Dow Jones when \$ 1 is invested in each of those. From these Figures, we make the following conclusions:

- The realized P&L trajectories of the optimal portfolios for all the models and for each cost function consistently outperform that of Dow Jones for the whole out-of-sample time span.
- The estimated drift term r in ISV gives an upwards bias in the P&L cloud. This bias in combination with the fact that cross correlations are set equal to zero may lead to overestimated CI forecast.
- The CIs of MSV and FM seem to give a more reliable CI for the P&L over the time, since the realized P&L trajectories consistently lie between the cloud bounds.
- The CI bandwidth of FM is narrower than that of the other models and this gives a narrower forecast for the portfolio value.

For the MSV model, the predicted distributions of P&L and Sharpe Ratios of the optimal portfolios on time $K = 66$ can be viewed respectively in Figures 3.4 and 3.5. In these Figures, we also place two vertical lines that show where the realized P&L and Sharpe Ratios of $\mathbf{w}_{T+K}$ and Dow Jones portfolios lie in the distributions. For ISV and FM models, the relative distribution Figures are placed in Appendix C Sections C.1.5, and C.3.3, , respectively. Here we observe:

- The realized PL_{T+K} and the annualized Sharpe Ratios SR_{T+K} of the optimal portfolios outperform those of Dow Jones index for all the models and cost functions.
- Again ISV model seems to overestimate the predicted distributions of the PL_{T+K} and the SR_{T+K} , as the realized values lie in the left tail of the distributions.
- On the other hand, the predicted distributions of MSV and FM for the PL_{T+K} and SR_{T+K} seem to be more plausible, since the realized values lie not far from the modes.
- The MSV and FM distributions are more positive skewed than those of ISV for both PL_{T+K} and SR_{T+K} .

We note that credible intervals depend on training data in all cases. If there are large discrepancies in the characteristics of the test and training time series, this will affect the results for all models.

3.5.3 Predicted asset allocations under various constraints

In portfolio optimization, terminal credible intervals of P&L and Sharpe Ratio are important measures of performance but maybe do not reflect the risk inherited from a not well diversified portfolio. For this reason, it is necessary to look into the distribution of the weights in the optimal allocation $\mathbf{w}_{T+k}$. So in this section we discuss how each model, under the various cost functions and constraints, as discussed in Section 3.2, distributes the weights over the 30 Dow Jones stocks. To highlight the above argument later on, we compare and contrast the Bayesian models' allocation against the allocation of the frequentist ones. In Appendix C, in Figures C.13, C.14 and C.15 we depict the weight distributions of the optimal allocations $\mathbf{w}_{T+k}$ for each model and cost function with the same order as labeled in 3.2. We make the following conclusions:

- Bayesian models have a much more smooth weight allocation comparing to those of frequentist ones under all cost functions.
- The constraints affect the allocation in Bayesian models, however, frequentist allocation changes greatly.
- Frequentist allocations tend to give very high weights in few stocks, especially when short selling is allowed.
- Frequentist allocations are much more sparse than the Bayesian ones.

3.6 Bayesian VS Frequentist optimal portfolios

In Table 3.1, we represent the main performance results for each model under the various cost functions as they have been labelled in Section 3.2. The first column shows the terminal out-of-sample realised P&L values after 66 days, when \$1 is invested in the optimal portfolio. The second and third columns show the annualised mean and standard deviation of the optimal portfolio realised daily returns, respectively. The annualised mean and standard deviation have been calculated as:

$$\text{Mean Return} = 252\mu_d \quad \text{SD} = \sqrt{252}\sigma_d, \quad (3.52)$$

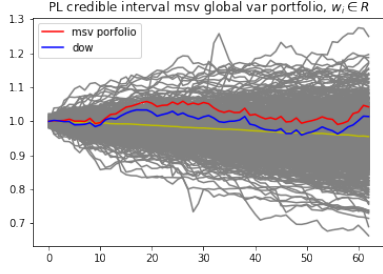
where 252 indicates the number of trading days in a year, while the μ_d and σ_d stand respectively for the daily mean and standard deviation of the portfolio daily log returns, expressed in percentages, during the 66 days out-of-sample period. The fourth and fifth columns show respectively the Sharpe Ratio $\mathbf{SR}_{T+K}$, which is the quotient of column two and three, and the L_2 , which is the Euclidean distance of the optimal allocation $\mathbf{w}_{T+K}$. The reason that we are interested in L_2 is that this can be treated as a measure of how well diversified a portfolio is. A higher value of L_2 of one portfolio compared to another indicates that the first portfolio has overall more extreme allocation than the second one, and the opposite.

Finally, the last column shows the values of the ratio of $\mathbf{SR}_{T+K}$ to L_2 . Such a ratio tries to reflect the portfolio performance against its diversification. Despite such an indicator not being investigated in the literature to the best of our knowledge, the last column of Table 3.1 indicates that there is a physical meaning in this measure. For example, on the cost function (3.2a) the FM model appears to have the poorest performance in Sharpe Ratio terms; however, it has by far the best in terms of the ratio.

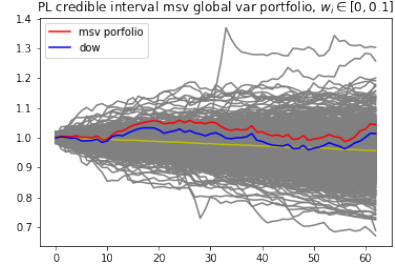
The relative out-of-sample trajectories for each model can be seen in Figure 3.6, where for each cost function we gather all the models and Dow Jones trajectories to make comparisons easier. Despite their simplicity and lack of credible interval forecasts, frequentist models produce very good results in absolute P&L and Sharpe Ratio terms, but many times behind these good figures a more extreme allocation is hiding. In the literature, big weight has been placed on the Sharpe Ratio as a portfolio performance measure, but little focus is given on how good the portfolio allocation is, which has the potential to be a serious underlying risk in extreme scenarios.

The empirical results in Table 3.1 and Figures 3.3–3.6 clearly support the claim that Bayesian models, particularly the MSV and FM models, yield consistently strong out-of-sample performance. For instance, under cost function (3.2b), the MSV model achieved a Sharpe Ratio of 1.34, outperforming both the ISV (0.83) and the Dow Jones benchmark (0.45). Moreover, the realized P&L trajectories for MSV-based portfolios remain within their predicted credible intervals (Figures 3.3 and 3.4), demonstrating the accuracy of posterior-based forecasting. These observations substantiate the claim that Bayesian estimators not only yield competitive returns but also provide reliable distributional forecasts for portfolio performance, which is especially valuable for risk-aware asset allocation.

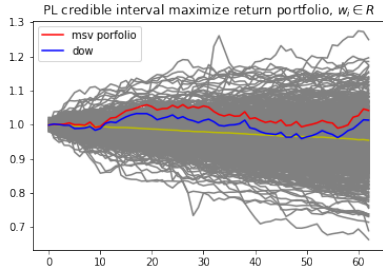
3.6.1 MSV Model P&L With Credible Intervals



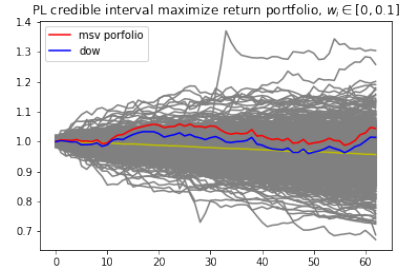
(a) Minimize global variance without boundaries (3.1a)



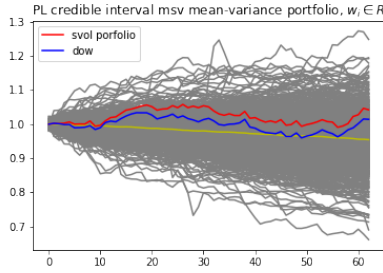
(b) Minimize global variance with boundaries (3.1b)



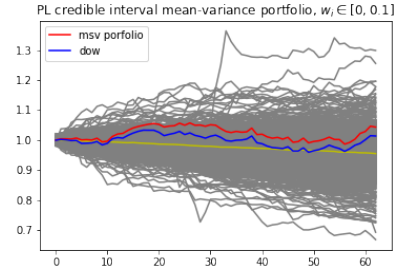
(c) Maximize expected return without boundaries (3.2a)



(d) Maximize expected return with boundaries (3.2b)



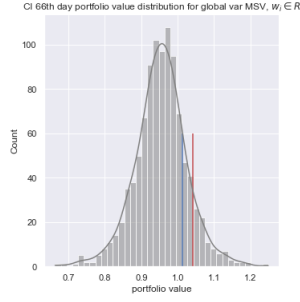
(e) Minimize variance for a fixed return with boundaries (3.3a)



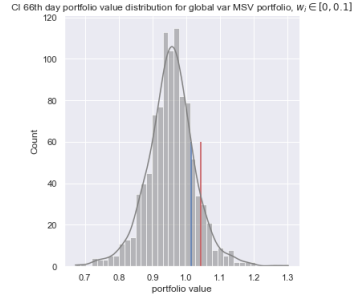
(f) Minimize variance for a fixed return without boundaries (3.3b)

Figure 3.3: MSV P&L clouds created by 1000 simulated portfolio trajectories for $K = 66$ future time steps, based on the optimal portfolio $\mathbf{w}_{T+K}$. Red and blue lines represent the realized P&L paths of the optimal allocation $\mathbf{w}_{T+K}$ and the Dow Jones, respectively, when \$1 is invested at $K = 0$. Here, $\mathbf{w}_{T+K}$ minimizes various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$.

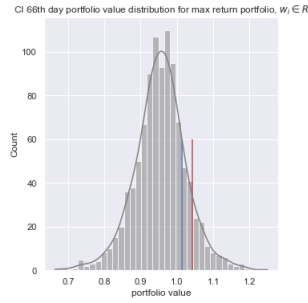
3.6.2 MSV Model \$1 P&L Distribution



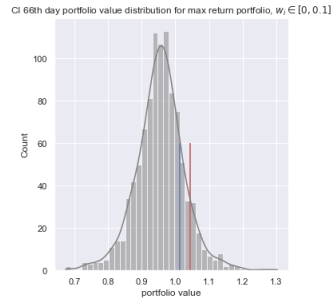
(a) Minimize global variance without boundaries, cost function (3.1a)



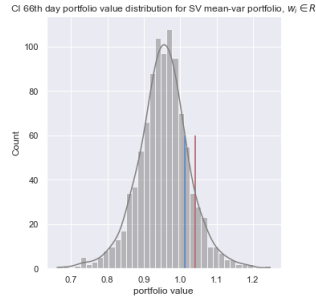
(b) Minimize global variance with boundaries, cost function (3.1b)



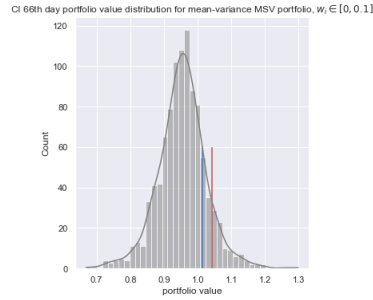
(c) Maximize expected return without boundaries, cost function (3.2a)



(d) Maximize expected return with boundaries, cost function (3.2b)



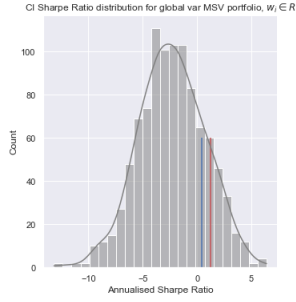
(e) Minimize variance for a fixed return with boundaries, cost function (3.3a)



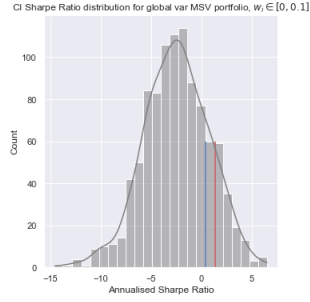
(f) Minimize variance for a fixed return without boundaries, cost function (3.3b)

Figure 3.4: Distribution of the 1000 simulated optimal portfolio values for day $T + K$, where future time steps $K = 66$ and the initial investment equals \$1 at $K = 0$. Red and blue lines represent the realized portfolio values for $\mathbf{w}_{T+K}$ and the Dow Jones, respectively. $\mathbf{w}_{T+K}$ optimizes various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$.

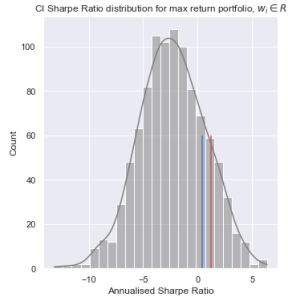
3.6.3 MSV Model Sharpe Ratio Distribution



(a) Global variance without boundaries (3.1a)



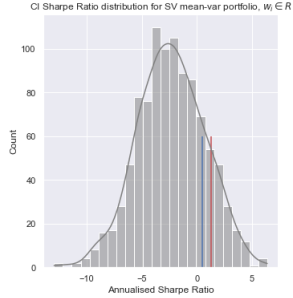
(b) Global variance with boundaries (3.1b)



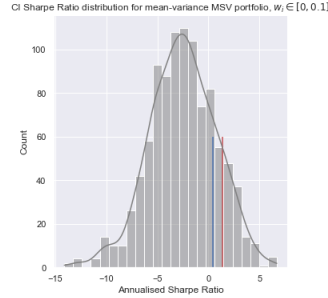
(c) Maximize return without boundaries (3.2a)



(d) Maximize return with boundaries (3.2b)



(e) Variance for fixed return with boundaries (3.3a)

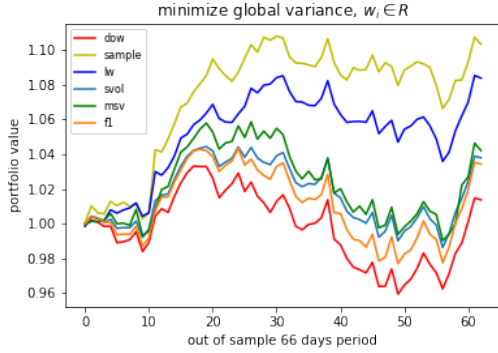


(f) Variance for fixed return without boundaries (3.3b)

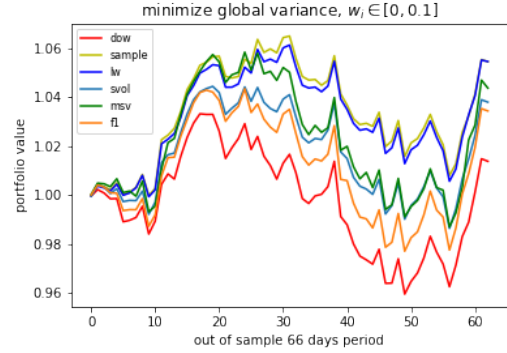
Figure 3.5: Distribution of the 1000 simulated optimal portfolio Sharpe Ratios for day $T + K$, where $K = 66$ future time steps and the initial investment equals \$1 at $K = 0$. Red and blue lines represent the realized Sharpe Ratios for $\mathbf{w}_{T+K}$ and the Dow Jones, respectively. $\mathbf{w}_{T+K}$ optimizes the various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$.

3.7 Conclusions

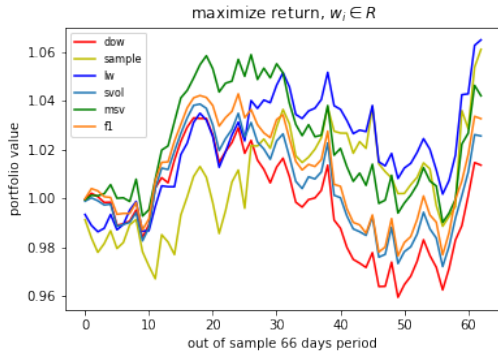
In this Chapter, we presented efficient ways of estimating covariance return matrices using Bayesian inference for Bayesian models. We applied these estimators in various portfolio



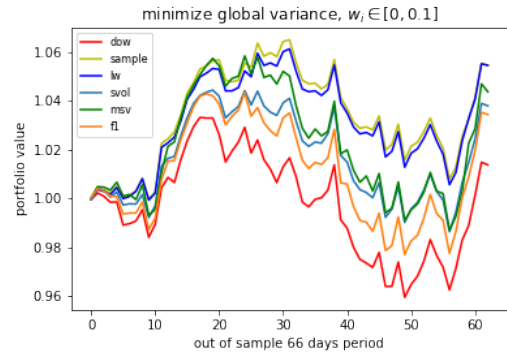
(a) Global variance, no boundaries (3.1a)



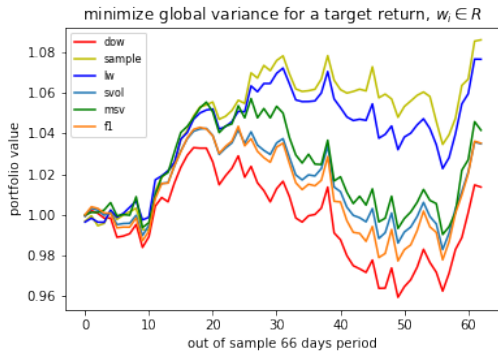
(b) Global variance, boundaries (3.1b)



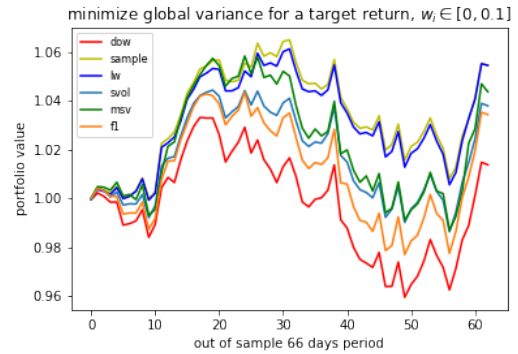
(c) Maximize return, no boundaries (3.2a)



(d) Maximize return, boundaries (3.2b)



(e) Variance-fixed return, no boundaries (3.3a)



(f) Variance-fixed return, boundaries (3.3b)

Figure 3.6: Portfolio value trajectories for various models and utility functions. Each panel shows the performance of optimal portfolio allocations under different utility function specifications, with or without boundary constraints.

	PL_{T+K}	Mean R(%)	SD	SR_{T+K}	L₂	Ratio
DJ Index	1.014	5.44	12.08	0.45		
<i>Cost Function 3.1a</i>						
S	1.103	32.29	11.72	3.35	0.697	4.85
LW	1.083	32.16	10.23	3.15	0.418	7.54
ISV	1.037	14.86	11.06	1.34	0.201	6.67
MSV	1.042	16.47	13.02	1.26	0.325	3.88
FM	1.034	15.11	11.51	1.31	0.216	6.06
<i>Cost Function 3.1b</i>						
S	1.054	23.19	10.46	2.03	0.294	6.90
LW	1.054	21.21	10.42	2.04	0.286	7.13
ISV	1.037	14.87	11.06	1.34	0.201	6.67
MSV	1.043	17.07	12.74	1.34	0.255	5.25
FM	1.034	15.15	11.52	1.31	0.215	6.09
<i>Cost Function 3.2a</i>						
S	1.061	23.77	16.64	1.43	1.001	1.43
LW	1.065	25.24	12.72	1.98	0.498	3.98
ISV	1.025	10.14	12.39	0.82	0.218	3.76
MSV	1.042	16.52	13.1	1.26	0.327	3.85
FM	1.032	15.36	11.48	1.34	0.215	6.23
<i>Cost Function 3.2b</i>						
S	1.058	22.83	10.73	2.13	0.299	7.12
LW	1.052	20.37	10.81	1.89	0.289	6.54
ISV	1.026	10.32	12.41	0.83	0.216	3.84
MSV	1.043	17.12	12.76	1.34	0.257	5.21
FM	1.032	15.39	11.49	1.34	0.214	6.26
<i>Cost Function 3.3a</i>						
S	1.086	33.06	10.76	3.07	0.611	5.02
LW	1.076	29.53	10.48	2.82	0.396	7.12
ISV	1.035	13.76	11.32	1.22	0.198	6.16
MSV	1.041	16.34	12.85	1.27	0.322	3.94
FM	1.035	17.69	11.31	1.56	0.211	7.39
<i>Cost Function 3.3b</i>						
S	1.054	24.04	10.68	2.25	0.289	7.79
LW	1.054	23.44	10.63	2.21	0.285	7.75
ISV	1.037	13.76	11.32	1.22	0.198	6.16
MSV	1.043	16.99	12.63	1.35	0.252	5.36
FM	1.034	17.69	11.31	1.56	0.211	7.39

Table 3.1: Annualized performance metrics of optimal portfolios evaluated out-of-sample

optimisation problems. We used a popular data set, the Dow Jones stocks, where an asset manager can invest a considerable amount of money and also attain a well diversified portfolio. All the models produced very good performance and outperformed consistently the Dow Jones index. The evidence of it is not only indicated by the numbers in Table 3.1 but also by the charts in Figure 3.6. This is important because this means that we can also outperform exchange-traded funds (ETF), which replicate the Dow Jones Index and are very popular among investors. In addition, we used 66 trading days of historical data for training our models and make our optimal allocation, intended to be held at least for the subsequent 66 days without reallocation. This is efficient in terms of commission and transaction cost. The process presented here is a model based and data driven approach that does not take into account micro fundamental factors of each company, so that the stock’s quality is reflected on its price and on its underlying volatility.

An interesting finding in this case study is that the frequentist methods using empirical covariance approximations perform really well, and this is in line with other works in the literature (Ledoit & Wolf 2003, 2004b, Kastner 2019). A major advantage of these estimators is that they are simple and very cheap in calculation. However, they tend to give high allocation weights in a few stocks, especially when short selling is allowed, and sparse optimal allocations. We see that almost half of the 30 stocks are not included in the optimal allocation. Sparsity, beyond the fact that it diminishes portfolio diversification, narrows down the number of stocks that an asset manager can choose. In this way, the asset manager has to allocate his budget in fewer stocks, and this can cause adverse price shifts when he reallocates his portfolio. Furthermore, for these models we cannot easily provide a reliable confidence interval. This can cause some concern from a risk analysis perspective.

Bayesian estimators are inferred through volatility models that have been shown to predict well the stocks’ returns. This is also clear from the credible intervals, which give very good forecasts and accurate predictions for the future value of the optimal portfolio. They also give very good performance results, but often not as strong as the frequentist competitors. Given LW can be thought of as a Bayesian point estimate due to the penalisation term in the estimation, this further motivates the search for better priors and parameterisations in volatility models. New volatility models often come with a dedicated MCMC calibration method so the overall approach presented here can be extended for more models. In addition, the Bayesian optimal portfolios are very well diversified without giving extreme allocation weights to few stocks. This also might be the reason why Bayesian portfolios seem to lose a bit of performance compared to the frequentist ones. On the other hand, MCMC is very expensive computationally.

While the models were tested rigorously on the Dow Jones Industrial Average 30 stocks using data from early 2011, it is important to acknowledge that conclusions regarding model performance are inherently tied to the characteristics of this dataset—specifically, large-cap U.S. equities during a relatively stable post-crisis recovery phase. Future research could extend this framework to different timeframes and market conditions to test the robustness of Bayesian volatility models under structural shifts and macroeconomic shocks.

Chapter 4

Volatility Estimation With Sequential Monte Carlo Squared

4.1 Introduction

In the previous chapter, we investigated the application of the PMCMC algorithm in MSV models and the estimate of the unknown parameter vector θ for a given batch of data. One basic drawback of this algorithm is that as we receive new data in time to estimate θ , we need to rerun the algorithm from scratch. Apart from being computationally expensive, PMCMC does not naturally allow us to observe how the parameter vector θ evolves as new observations occur. Another drawback of PMCMC is that sometimes it is difficult to choose a priori step sizes when new parameters are proposed so that we get a suitable acceptance ratio. This task gets even more difficult as the dimension of θ vector increases or the model becomes more complicated. Thankfully, the answers to the above questions have been given by an algorithm called SMC² (Chopin et al. 2013). This is a sequential Monte Carlo algorithm and can be used for sequential inference as new information arrives in time. The aim of this chapter is to apply SMC² for the optimal portfolio allocation in the context discussed previously. We begin the chapter with a general presentation of IBIS (a precursor to SMC²) and then present the methodology for Hidden Markov Models.

4.2 Iterated batch importance sampling, IBIS

The base of SMC² algorithm is the Iterated Batch Importance Sampling (IBIS) algorithm (Chopin 2002). IBIS is an algorithm that targets the estimation of vector θ of fixed dimension d from realized observations iteratively. In practice, we need to estimate the posterior distribution $p(\theta|\mathbf{y}_{1:T})$. However, we perform intermediate posterior estimations of partial distributions $p(\theta|\mathbf{y}_{1:t})$, $t < T$, by setting $(\theta_t^{1:N_\theta}, w_t^{1:N_\theta})$, where θ_t^i are now parameter particles. The algorithm is called "batch" because the posterior distribution $p(\theta|\mathbf{y}_{1:T})$ does not need to be updated in every single observation, but in batches of size p . For a selected batch size

p the estimated incremental weight for particle θ_{t+p-1}^i can be adjusted by:

$$w_{t,p}^i(\theta_{t+p-1}^i) \propto \frac{p(y_{1:t+p-1}|\theta_{t+p-1}^i)}{p(y_{1:t-1}|\theta_{t-1}^i)} = p(y_{t:t+p-1}|y_{1:t-1}, \theta_{t-1}^i). \quad (4.1)$$

When we set $p = 1$ then the incremental weight coincides with the incremental likelihood of the respective particle θ_{t+p-1}^i .

The move step, also known as rejuvenation, is a crucial aspect of the algorithm. Its purpose is to prevent the degeneration and depletion of particles, as resampling alone does not entirely safeguard against this. Resampling merely substitutes particles with very low weights with those possessing higher weights, essentially generating numerous duplicates. Rejuvenation involves subsequently relocating resampled particles based on a certain MCMC kernel K_t that is invariant to the posterior distribution $p(\theta|\mathbf{y}_{1:t})$. This infusion introduces greater diversity into the particle population while maintaining or even improving the accuracy of the approximation (Andrieu et al. 1999). The Metropolis-Hastings kernel is a commonly employed choice for the rejuvenation step. The outline of the IBIS Algorithm 3 is depicted below.

Algorithm 3 Iterated Batch Importance Sampling (IBIS)

```

1: for  $i = 1$  to  $N_\theta$  do
2:   Sample initial parameters from prior:  $\theta_0^{(i)} \sim p(\theta)$ 
3:   Set initial weight:  $w_0^{(i)} \leftarrow 1$ 
4: end for
5: for  $t = 1$  to  $T$  do
6:   Compute incremental likelihood:  $p_\theta(y_t|y_{1:t-1})$ 
7:   Set new weights:  $w_t^{(i)} \leftarrow p_\theta(y_t|y_{1:t-1})$ 
8:   Resample new indices  $I_{t-1}^{(i)}$  from  $w_{t-1}^{(i)}$ 
9:   Rejuvenation Step: move the particles,  $\theta_t^{(i)} \sim K_t(\theta_{t-1}^{(I_{t-1}^{(i)})})$ 
10:  Set weights:  $w_t^{(i)} \leftarrow 1$ 
11: end for

```

In many case state-space models set IBIS inapplicable due to the intractable likelihood $p_\theta(\mathbf{y}_t|\mathbf{y}_{1:t-1})$, except in cases such as linear Gaussian models (Chopin 2007). The solution to this problem is the employment of SMC².

4.3 Sequential MonteCarlo Squared, SMC²

For HMMs implementing IBIS requires using an approximation of $p(y_t|y_{1:t-1})$ as it is intractable. This approximation can be provided by by a particle filter and the resulting algorithm is called SMC² (Chopin et al. 2013). SMC² consists of the application of two algorithms that we have already discussed, PMCMC (Algorithm 2) and IBIS (Algorithm3). It

consists of two nested particle filters; IBIS propagates θ -particles each of which propagating its own particle filter (for the given θ -particle value) as new data are received sequentially. θ 's and their particle filters are resampled jointly and a suitable PMCMC algorithm is used in the rejuvenation step with the aim to smooth out and diversify the existing distribution of θ -particles. SMC² at time t , involves the following particle system:

$$\{\theta_t^i, w_{\theta,t}^i, \{x_t^{i,j}\}_{j=1:N_x}\}_{i=1:N_\theta} \quad (4.2)$$

which defines the posterior $p(\theta|y_{1:T})$ at time t , given all the observations up to t . So, we have N_θ number of particles θ_t^i , and each of those is assigned a new set of N_x number of x -particles $x_t^{i,j}$, which will be resampled and will be substituted by equally weighted x -particles $\bar{x}_t^{i,j}$. Finally, $w_{\theta,t}^i$ and $w_{x,t}^{i,j}$ represent the weights for θ_t^i and $x_t^{i,j}$ respectively. For each θ_t^i the incremental likelihood is calculated by

$$w_{\theta,t}^i \propto \hat{p}_{\theta_t^i}(y_t|y_{1:t-1}) = \frac{1}{N_x} \sum_{j=1}^{N_x} w_{x,t}^{ij}(x_t^{ij}, \bar{x}_{t-1}^{ij}) \quad (4.3)$$

and the likelihood by

$$\hat{Z}_t^i(\theta_t^i, x_t^{i,1:N_x}, \bar{x}_{1:t-1}^{i,1:N_x}) = \frac{1}{N_x} \sum_{j=1}^{N_x} w_{x,1}^{ij}(x_1^{ij}) \prod_{n=2}^t \frac{1}{N_x} \sum_{j=1}^{N_x} w_{x,n}^{ij}(x_n^{ij}, \bar{x}_{n-1}^{ij}) \quad (4.4)$$

which is basically the product of all incremental likelihoods up to time t , or in a recursive form we can write alternatively:

$$\hat{Z}_t^i(\theta_t^i, x_t^{i,1:N_x}, \bar{x}_{1:t-1}^{i,1:N_x}) = \hat{p}_{\theta_t^i}(y_t|y_{1:t-1}) \hat{Z}_{t-1}^i \quad (4.5)$$

SMC² is described below in Algorithm 4. Algorithm 4 may have various implementations but here we chose the one that was simple effective in the experiments carried out later in this chapter. Due to this customization, we need to point out that resampling of θ and x particles takes place at every iteration. Adaptive resampling could be possible but would lead to a more complicated algorithm. For example on line 22 in Algorithm 4 the θ -weights are updated directly as $w_t^{(i)} \leftarrow p_\theta(y_t|y_{1:t-1})$ instead of $w_t^{(i)} \leftarrow w_{t-1}^{(i)} p_\theta(y_t|y_{1:t-1})$.

Before kicking off the algorithm, the number of θ -particles, N_θ , and the number of x -particles, N_x , has to be defined. This is an important design choice, because as the number of N_θ or N_x increases, the computational time also increases. However, having not enough θ or x particles will lead in weight degeneracy or impoverishment of particles. For the rejuvenation step, Andrieu et al. (2010) showed that acceptance rates might become too low if N_x is small compared to observed data. This implies that N_x should be tuned proportional to time T . Chopin et al. (2013) and Chopin et al. (2015) also proposed techniques to automatically add more particles when acceptance rate falls below a preset threshold.

A big part of the SMC² algorithm is the rejuvenation step, which consists of N_r iterates of a standard MCMC kernel. Its role is to smooth out the distribution of the θ -particles so

as to induce particle diversity and mitigate particle degeneracy using a kernel K_t . Normally, N_r equals to a small number like three or four but this is also application and dimension dependent. The rejuvenation step is similar to Algorithm 2 and every time is called, it starts running from time $t = 1$ until the last observation at time $t = T$. This process constructs new proposed $\tilde{\theta}_t^i$ with respective likelihood $\tilde{Z}_t^i(\tilde{\theta}_t^i)$ and $\tilde{x}_{1:t}^{i,1:N_x}$ particles. The acceptance ratio for each particle at any time t is given by:

$$\min \left(1, \frac{\tilde{Z}_t^i(\tilde{\theta}_t^i)p(\tilde{\theta}_t^i)q(\theta_t^i|\tilde{\theta}_t^i)}{\tilde{Z}_t^i(\theta_t^i)p(\theta_t^i)q(\tilde{\theta}_t^i|\theta_t^i)} \right) \quad (4.6)$$

If the move is accepted, the new θ -particle, x -particles and likelihood must replace the current ones. It is worth noting that the denominator of (4.6) does not need to be computed in each rejuvenation step, as it has already been computed up to previous time. When the parameter Θ space is of high dimension, we need to carry out the rejuvenation step in such a way so we ensure that Θ space is explored thoroughly enough. This means that we need to pick a suitable sampler for $\tilde{\theta}^i$ and set the the number of rejuvenation steps accordingly.

The most time-consuming part of SMC² is the particle rejuvenation step, which has a computational complexity $O(TN_\theta N_x)$. The overall time consumption of one iteration is not growing perfectly linearly in the case N_x is allowed to change. Finally, the overall complexity of SMC² is $O(T^2 N_r N_\theta N_x)$, which is square in time. However, the computational time is also a function of the Θ and x space dimension, which means in more advanced applications this is computationally intensive. However, SMC² facilitates the tuning and estimation of various measures (e.g. empirical covariance) dependent on θ -particles as these are observable in each time iteration.

SMC² enables the sequential estimation of parameters as new observations are received. However, it is not considered a fully online algorithm, as at each t requires processing t seen data-points when computing acceptance probabilities. Despite this, it behaves in a sequential manner in the sense that, at any time t , SMC² provides the posterior distribution $p(\theta|\mathbf{y}_{1:t})$. This is a great advantage when we are interested in observing the changes of θ , or any estimator based on it, as new data received in time.

Algorithm 4 SMC²

```
1: Sample  $\theta_1^i \sim p(\theta)$  for  $i = 1, \dots, N_\theta$ 
2: Set  $w_{\theta,1}^i \leftarrow 1$  for  $i = 1, \dots, N_\theta$ 
3: for  $i = 1$  to  $N_\theta$  do
4:   for  $j = 1$  to  $N_x$  do
5:     Sample  $x_1^{ij} \sim f_{\theta^i}(\cdot)$ 
6:     Compute weights  $w_{x,1}^{ij} = g_{\theta^i}(y_1|x_1^{ij})$ 
7:     Compute  $\hat{Z}_1^i = \frac{1}{N_x} \sum_{j=1}^{N_x} w_{x,1}^{ij}(x_1^{ij})$ 
8:     Normalize  $W_{x,1}^{ij} \propto w_{x,1}^{ij}$ ,  $\sum_{j=1}^{N_x} W_{x,1}^{ij} = 1$ 
9:     Resample  $\{W_{x,1}^{ij}, x_1^{ij}\}$  to obtain  $N_x$  new equally-weighted particles:  $\{1/N_x, \bar{x}_1^{ij}\}$ 
10:   end for
11: end for
12: for  $t = 2$  to  $T$  do
13:   for  $i = 1$  to  $N_\theta$  do
14:     for  $j = 1$  to  $N_x$  do
15:       Sample  $x_t^{ij} \sim f_{\theta^i}(x_t|y_{t-1}, \bar{x}_{t-1}^{ij})$ 
16:       Compute weights  $w_{x,t}^{ij} = g_{\theta^i}(y_t|x_t^{ij})$ 
17:       Compute incremental likelihood  $\hat{p}_{\theta^i}(y_t|y_{1:t-1})$  as in equation (4.3)
18:       Normalize  $W_{x,t}^{ij} \propto w_{x,t}^{ij}$ ,  $\sum_{j=1}^{N_x} W_{x,t}^{ij} = 1$ 
19:       Resample  $\{W_{x,t}^{ij}, x_t^{ij}\}$  to obtain  $N_x$  new equally-weighted particles:  $\{1/N_x, \bar{x}_t^{ij}\}$ 
20:     end for
21:   end for
22:   Set new  $\theta$ -weights  $w_{\theta,t}^i \leftarrow \hat{p}_{\theta^i}(y_t|y_{1:t-1})$ 
23:   Compute  $\hat{Z}_t^i(\theta_t^i, x_t^{i,1:N_x}, \bar{x}_{1:t-1}^{i,1:N_x})$  as in equation (4.5)
24:   Normalize  $W_{\theta,t}^i \propto w_{\theta,t}^i$ ,  $\sum_{i=1}^{N_\theta} W_{\theta,t}^i = 1$ 
25:   Resample index  $\{I_t^i\}_{i=1}^{N_\theta}$  from  $(W_{\theta,t}^{i:N_\theta}, N_\theta)$  and set new particle system:
   •  $\{\theta_t^{I^i}, \bar{x}_t^{I^i,1:N_x}, \hat{Z}_t^{I^i}\}_{i=1}^{N_\theta}$ 
26:   Rejuvenation Step (repeat  $N_r$  times):
27:   for  $r = 1$  to  $N_r$  do
   • Move the resampled particles  $\{\theta_t^i, \bar{x}_{1:t}^{i,1:N_x}, \hat{Z}_t^i\}_{i=1}^{N_\theta} \sim K_t \left( \{\tilde{\theta}_t^i, \tilde{x}_{1:t}^{i,1:N_x}, \hat{\tilde{Z}}_t^i\}_{i=1}^{N_\theta}, \cdot \right)$ 
28:   end for
29: end for
```

4.4 Experimental Setup

In this section, we present two experiments based on the log-returns of the adjusted stock prices for the Dow Jones 30 constituents. We apply the SMC² algorithm to estimate the

posterior distribution $p(\theta|y_{1:T})$ using the MSV model, as outlined in subsection 3.4.2. Here, the parameter vector θ includes the following components:

$$\theta = (\alpha, \sigma_\epsilon^2, \sigma_\eta^2, \text{corr}_{\epsilon\epsilon}, \text{corr}_{\eta\eta}, \text{corr}_{\epsilon\eta})^T \quad (4.7)$$

Rather than directly targeting the elements of the covariance matrices, we focus on the corresponding correlations and then infer the underlying covariances. Correlations are restricted to the range between -1 and 1, while covariances are not confined to a specific range. This focus on correlations enhances the efficiency of the SMC² algorithm in sampling within a constrained domain, thereby increasing the algorithm's stability, especially when log-return volatility rises and parameter shifts are more significant.

Additionally, we omit the estimation of the cross-leverage effect correlation, meaning we set $(\Sigma_{\epsilon\eta})_{ij} = 0$, effectively ignoring the covariance between the i -th stock's return and the j -th stock's volatility. By excluding this parameter, we reduce the dimensionality of θ by one, which speeds up the algorithm without significantly affecting the explanatory power of the MSV model.

4.4.1 Training Data Preprocessing

Before initiating the SMC² algorithm, we preprocess the training data of log-returns. First, we cap and floor outliers in the log-returns using a threshold based on standard deviations. Specifically, we apply a three standard deviations rule, which assumes that, in a normal distribution, 99.7% of the data lies within plus or minus three standard deviations from the mean. Any outliers beyond this range are capped (for upper outliers) or floored (for lower outliers). The mean and standard deviation are calculated at each time step across the 30 Dow Jones stocks, and we apply the capping and flooring accordingly. This helps the SMC² algorithm to update the posterior $p(\theta|y_{1:T})$ smoothly over time, preventing degeneracy in the θ -particles when one or a few stocks exhibit extreme behavior due to stock-specific micro-events. Every log-return for each stock i at time t , y_{it} , can be floored and capped as follows:

$$\bar{y}_t = \frac{1}{d} \sum_{i=1}^d y_{it}, \quad \sigma_t = \sqrt{\frac{1}{d} \sum_{i=1}^d (y_{it} - \bar{y}_t)^2} \quad (4.8)$$

$$y_{it} \leftarrow \max(\bar{y}_{it} - 3\sigma_t, y_{it}), \quad y_{it} \leftarrow \min(\bar{y}_{it} + 3\sigma_t, y_{it}) \quad (4.9)$$

where $\bar{y}_t$ and σ_t represent the cross mean and standard deviation respectively at time t across the d stocks, where in our case $d = 30$. Equation (4.9) represents capping and flooring the value y_{it} within the range $\bar{y}_t \pm 3\sigma_t$, first ensuring it's no smaller than the lower bound and then no larger than the upper bound.

Additionally, we standardize the log-returns for each stock, ensuring that each log-return time series has a mean of zero and a standard deviation of one. This transformation scales the data without affecting the correlations between variables or time points, and it is a common preprocessing step in statistical analysis, time series modeling and machine learning, especially when working with financial data like stock returns. Standardization ensures the

data is centered and has unit variance, which is crucial for avoiding disproportionate influence of certain stocks in the MSV model due to varying log-return scales. We observed that, without standardization, when volatility in log-returns increases unevenly among stocks, the posterior distribution domain tends to shrink, and degeneracy in the θ -particles becomes more frequent, leading to potential collapse of the SMC² algorithm in just a few time steps. For standardising the log-returns we use the following linear transformation:

$$\mathbf{z}_t = \mathbf{D}^{-1}(\mathbf{y}_t - \bar{\mathbf{y}}) \quad (4.10)$$

where $\bar{\mathbf{y}}$ and $\mathbf{D}^{-1}$ are constants computed by the log-returns training vectors as follows:

$$\bar{\mathbf{y}} = \frac{1}{T} \sum_{t=1}^T \mathbf{y}_t \quad (4.11)$$

$$\mathbf{D}^{-1} = \text{diag}\left\{\frac{1}{\sigma_1}, \frac{1}{\sigma_2} \dots \frac{1}{\sigma_d}\right\}, \quad \sigma_i = \sqrt{\frac{1}{T} \sum_{t=1}^T (y_{it} - \bar{y}_i)^2} \quad (4.12)$$

We recall that $\mathbf{y}_t$ denotes the d -dimensional log-returns vector, and $\mathbf{D}^{-1}$ is a diagonal $d \times d$ matrix with the reciprocals of the standard deviations along the diagonal, computed individually for each stock over time. Using (4.10), once the expected value and covariance of z_t at time $t = T + k$ are known, the corresponding measures for y_t can be determined as follows:

$$\mathbb{E}[\mathbf{y}_{T+k}] = \bar{\mathbf{y}} + D\mathbb{E}[\mathbf{z}_{T+k}], \quad \text{Cov}(\mathbf{y}_{T+k}) = D\text{Cov}(\mathbf{z}_{T+k})D \quad (4.13)$$

4.5 Training SMC² Algorithm for MSV Model

Before we get in the analysis of the results of the experiments, in this section we discuss some key points about how we applied SMC² in the two experiments and the main challenges faced while we tried to estimate the posterior θ .

4.5.1 Sampling SMC² Prior

The first step of the SMC² algorithm involves selecting a number N_θ of θ -particles and sampling them from a prior distribution. These sampled θ -particles are then used to construct the matrices $\Sigma_{\epsilon\epsilon}$, $\Sigma_{\eta\eta}$, and $\Sigma_{\epsilon\eta}$. As highlighted in the MSV model in Section 3.4.2, $\Sigma_{\epsilon\epsilon}$ and $\Sigma_{\eta\eta}$ represent covariance matrices. However, selecting an appropriate prior for θ is not straightforward. First, the initial θ_0^i must be chosen such that both $\Sigma_{\epsilon\epsilon}$ and $\Sigma_{\eta\eta}$ are positive definite. Second, this requirement extends to the matrix $\Sigma_{x_t|x_{t-1}, y_{t-1}}$, meaning that θ_0^i must also enable the later sampling of x -particles from the transition distribution $f(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{y}_{t-1})$ (see Algorithm 4, line 15). This transition density, as described in equation 3.39, has a covariance matrix defined as:

$$\Sigma_{x_t|x_{t-1}, y_{t-1}} = \Sigma_{\eta\eta} + \Sigma_{\eta\epsilon} \Sigma_{\epsilon\epsilon}^{-1} \Sigma_{\epsilon\eta} \quad (4.14)$$

Thus, it is crucial to initialize each θ_0^i in such a way that not only are $\Sigma_{\epsilon\epsilon}$ and $\Sigma_{\eta\eta}$ positive definite, but (4.14) also yields a positive definite matrix.

The matrices $\Sigma_{\epsilon\epsilon}$ and $\Sigma_{\eta\eta}$ follow the same algebraic form as equation (A.1) in Appendix Section A.1. This allows us to apply Corollary 2 in the Appendix to determine the support of $Corr_{\epsilon\epsilon}$ and $Corr_{\eta\eta}$, which are respectively $[-\frac{1}{d-1}, 1]$. It is important to note that this support does not depend on the level of variances σ_ϵ^2 and σ_η^2 but only on d . This is in contrast to cases where the support of covariances depends on the level of the corresponding variances. As a result, sampling correlations becomes independent of sampling the respective variances, contributing to the stability of the algorithm. This is particularly beneficial during the rejuvenation step, where good proposal samples are crucial to avoid degeneration and impoverishment.

However, while sampling θ from the prior or proposals ensures that $\Sigma_{\epsilon\epsilon}$ and $\Sigma_{\eta\eta}$ are positive definite, it does not guarantee that $\Sigma_{x_t|x_{t-1}, y_{t-1}}$ will also be positive definite. Fortunately, when $\Sigma_{\epsilon\epsilon}$ and $\Sigma_{\eta\eta}$ are positive definite, approximately 90% of the sampled θ^i will also produce a positive definite $\Sigma_{x_t|x_{t-1}, y_{t-1}}$. Furthermore, since $\Sigma_{x_t|x_{t-1}, y_{t-1}}$ has the same algebraic form as $\Sigma_{\epsilon\epsilon}$, $\Sigma_{\epsilon\epsilon}^{-1}$ and $\Sigma_{\eta\eta}$, according to Proposition 1 in the Appendix A.1, we can apply Corollary 1 and filter out which θ^i lead to non-positive definite matrices for $\Sigma_{x_t|x_{t-1}, y_{t-1}}$. To do this we just check if the diagonal or non diagonal elements of $\Sigma_{x_t|x_{t-1}, y_{t-1}}$ lie out of the support defined from Corollary 1. The same filter will be used later in the rejuvenation step, when we need to check if the proposals do not constitute positive definite matrices for $\Sigma_{x_t|x_{t-1}, y_{t-1}}$. This gives us the advantage of initializing and sampling θ -particles that consistently yield positive definite matrices. Particularly in the rejuvenation step, any sampled θ -particles that do not produce positive definite $\Sigma_{x_t|x_{t-1}, y_{t-1}}$ are immediately rejected. However, as the θ -particles converge to the true posterior distribution, this sampling issue becomes negligible. Due to the availability of analytical solutions for the covariance eigenvalues, these processes do not incur any substantially additional computational cost for the SMC² algorithm as a whole.

Now that we have addressed the primary challenges in selecting suitable θ priors, we can define the priors and supports for the θ parameters, as shown in (4.7). For σ_ϵ^2 and σ_η^2 , we adopt an inverse gamma $IG(1, 1)$ prior. This distribution has a diffuse tail, which allows it to take on large values with non-negligible probability, making it ideal for modeling variances or scales where significant deviations are expected. For $corr_{\epsilon\epsilon}$ and $corr_{\eta\eta}$, we could assign uniform priors with support $[-\frac{1}{d-1}, 1]$. However, for $corr_{\epsilon\epsilon}$, we can further restrict the support to $[0, 1]$, as stocks are generally expected to be positively correlated. For α , we assign a uniform prior with support $[-1, 1]$ to ensure stationarity in the MSV model. Finally, for $corr_{\epsilon\eta}$, we assign a uniform prior with support $[-1, 1]$, as there are no specific sampling restrictions. A synopsis for the θ prior is given in the Table 4.1 below.

4.5.2 Rejuvenation step

As previously mentioned, the success of SMC² in terms of convergence to the posterior distribution $p(\theta|\mathbf{y}_{1:T})$ heavily relies on the rejuvenation step. This step will be repeated N_r times before algorithm moves to the next time step t . The primary goal of rejuvenation

Table 4.1: Summary of θ Priors

θ Parameter	Support	Prior Distribution
α	$[-1, 1]$	Uniform(-1, 1)
σ_ϵ^2	$[0, \infty)$	InverseGamma(1, 1)
σ_η^2	$[0, \infty)$	InverseGamma(1, 1)
$corr_{\epsilon\epsilon}$	$[0, 1]$	Uniform(0, 1)
$corr_{\eta\eta}$	$[-\frac{1}{d-1}, 1]$	Uniform($-\frac{1}{d-1}$, 1)
$corr_{\epsilon\eta}$	$[-1, 1]$	Uniform(-1, 1)

is to promote diversity among the θ -particles and reduce particle degeneracy. In many applications a degeneracy criterion must be fulfilled so that the rejuvenation step will be triggered. In the experiments we carried out, in every time step we utilize the rejuvenation step for more algorithmic stability. The rejuvenation step is outlined in Algorithm 2, but here, we will delve into more detail regarding the proposal sampling process over time.

Sampling Proposals and Adaptive Step Size

In rejuvenation step there are various proposals one can employ. For example Chopin et al. (2013) suggests that new particles are proposed using a Gaussian random walk:

$$\tilde{\theta} \mid \theta \sim \mathcal{N}(\theta, c\hat{\Sigma}),$$

with $c\hat{\Sigma}$ being the empirical covariance calculated from the θ -particles and a step size. Alternatively, particles can be independently drawn as

$$\tilde{\theta} \sim \mathcal{N}(\hat{\mu}, \hat{\Sigma}),$$

with $\hat{\mu}$ being the empirical mean calculated from the existing θ -particles.

In our case, we observed that we can effectively scan the Θ space by employing proposal for each component of the parameter θ . We will use Gaussian random walks, where each parameter has its own adaptive step size s_t at time t . It is crucial for each parameter to have its own step size at time t , as each one exhibits different scaling and influence as new data is incorporated. For simplicity, we suppose that θ_t refers to a scalar parameter here. The proposal can be represented as follows:

$$\tilde{\theta}_t = \theta_t + s_t \epsilon, \quad \epsilon \sim N(0, 1) \tag{4.15}$$

where

$$s_t = \sqrt{\frac{1}{N_\theta} \sum (\theta_t^i - \bar{\theta}_t)^2}, \quad \bar{\theta}_t = \frac{1}{N_\theta} \sum \theta_t^i \tag{4.16}$$

As explained in Section 4.5.1 to produce positive definite $\Sigma x_t | x_{t-1}, y_{t-1}$ and to ensure we are always sampling efficiently within the appropriate support for each parameter, we employ transformed random walks. Therefore, for the proposals of σ_ϵ^2 and σ_η^2 , we use transformed random walks on a log scale, as outlined in equation 3.46 in Section 3.5.1. For parameters constrained within $[a, b]$, the proposals follow transformed random walks on a logit scale, as follows:

$$\tilde{\theta}_t = \frac{b \exp(Z_t) - a}{\exp(Z_t) + 1}, \quad (4.17)$$

$$Z_t = \log(\theta_t - a) + \log(b - \theta_t) + s_t \epsilon, \quad \epsilon \sim N(0, 1). \quad (4.18)$$

In all instances, the acceptance ratio must be properly adjusted for the Jacobian. A summary of the θ sampler, including the Jacobian adjustments and the simplifications arising from the symmetric proposal distributions, is presented in Table 4.2 below.

Table 4.2: Summary of θ Samplers and Jacobians

θ Parameter	Support	Sampler	$\frac{q(\theta \tilde{\theta})}{q(\tilde{\theta} \theta)}$
α	$[-1, 1]$	Logit	$\frac{(\tilde{\alpha}+1)(1-\tilde{\alpha})}{(\alpha+1)(1-\alpha)}$
σ_ϵ^2	$[0, \infty)$	Log	$\frac{\tilde{\sigma}_\epsilon^2}{\sigma_\epsilon^2}$
σ_η^2	$[0, \infty)$	Log	$\frac{\tilde{\sigma}_\eta^2}{\sigma_\eta^2}$
$corr_{\epsilon\epsilon}$	$[0, 1]$	Logit	$\frac{\widetilde{corr}_{\epsilon\epsilon}(1-\widetilde{corr}_{\epsilon\epsilon})}{corr_{\epsilon\epsilon}(1-corr_{\epsilon\epsilon})}$
$corr_{\eta\eta}$	$[-\frac{1}{d-1}, 1]$	Logit	$\frac{(\widetilde{corr}_{\eta\eta} + \frac{1}{d-1})(1-\widetilde{corr}_{\eta\eta})}{(corr_{\eta\eta} + \frac{1}{d-1})(1-corr_{\eta\eta})}$
$corr_{\epsilon\eta}$	$[-1, 1]$	Logit	$\frac{(\widetilde{corr}_{\epsilon\eta}+1)(1-\widetilde{corr}_{\epsilon\eta})}{(corr_{\epsilon\eta}+1)(1-corr_{\epsilon\eta})}$

Rejuvenation Algorithm

Following all the above discussion we can now represent a single rejuvenation step suitable for the MSV model as in Algorithm 5, which is part of Algorithm 4 in lines 26 - 28.

Algorithm 5 Rejuvenation Step

At time t :

Input: Current particle system $(\theta_t^i, \widehat{Z}_t^i, \bar{x}_{1:t}^{i,1:N_x})$

for $i = 1$ to N_θ **do**

- Sample N_θ proposals $\tilde{\theta}_t^i$ according to Table 4.2

if $\tilde{\theta}_t^i$ does not constitute $\Sigma_{\epsilon\epsilon}$, $\Sigma_{\eta\eta}$, and $\Sigma_{x_t|x_{t-1}, y_{t-1}}$ positive definite **then**

 Leave $(\theta_t^i, \widehat{Z}_t^i, \bar{x}_{1:t}^{i,1:N_x})$ unchanged

else

- Run a PF to get $(\widetilde{\widehat{Z}}_t^i, \widetilde{\bar{x}}_{1:t}^{i,1:N_x})$ for $\tilde{\theta}_t^i$

- Calculate acceptance probability:

$$\alpha = \min \left(1, \frac{\widetilde{\widehat{Z}}_t^i(\tilde{\theta}_t^i) p(\tilde{\theta}_t^i) q(\theta_t^i | \tilde{\theta}_t^i)}{\widehat{Z}_t^i(\theta_t^i) p(\theta_t^i) q(\tilde{\theta}_t^i | \theta_t^i)} \right)$$

- **if** $u \sim \mathcal{U}(0, 1) < \alpha$ **then**

- Accept the proposal: $(\theta_t^i, \widehat{Z}_t^i, \bar{x}_{1:t}^{i,1:N_x}) \leftarrow (\tilde{\theta}_t^i, \widetilde{\widehat{Z}}_t^i, \widetilde{\bar{x}}_{1:t}^{i,1:N_x})$

- **else**

- Reject the proposal, leave $(\theta_t^i, \widehat{Z}_t^i, \bar{x}_{1:t}^{i,1:N_x})$ unchanged.

- **end if**

end if

end for

4.5.3 Residual Resampling

In SMC², resampling is a key step that occurs at different phases of the algorithm, either within the IBIS procedure or during the rejuvenation phase. As previously mentioned, we perform resampling consistently across our applications, without using a threshold criterion. Various resampling methods are available (Douc & Cappé 2005), and in our experiments, we found that residual resampling (Beadle & Djuric 1997) performs effectively.

The primary advantage of residual resampling in particle filtering is its efficiency in mitigating sample degeneracy while reducing randomness. By assigning deterministic copies of particles based on the integer part of their weights, residual resampling decreases the variability in the process. This reduction in variance helps achieve a more accurate representation of the underlying probability distribution. Particles with higher weights receive a proportionally greater number of deterministic copies, ensuring that the most probable states are better represented, which enhances the quality of the particle set. This approach also avoids discarding high-weight particles, a common issue in purely random resampling techniques. A basic structure of the residual resampling process is outlined in Algorithm 6.

Algorithm 6 Residual Resampling

- 1: **Input:** Particle weights $w = [w_1, w_2, \dots, w_N]$
- 2: Initialize an empty list *resampled_indices*
- 3: **for** each particle i from 1 to N **do**
- 4: Calculate the deterministic copies: $c_i = \lfloor w_i \cdot N \rfloor$
- 5: Add c_i copies of particle i to *resampled_indices*
- 6: Calculate the residual weight: $r_i = w_i \cdot N - c_i$
- 7: **end for**
- 8: Normalize the residual weights: $\tilde{r}_i = \frac{r_i}{\sum_{j=1}^N r_j}$
- 9: Determine the number of remaining particles needed:
 - Let $M = N - \sum_{i=1}^N c_i$ be the number of particles still needed
- 10: Draw M samples from the multinomial distribution with probabilities given by the normalized residual weights $\tilde{r}_i$:

$$\mathbf{O} = (O_1, O_2, \dots, O_N) \sim \text{Multinomial}(M, (\tilde{r}_1, \tilde{r}_2, \dots, \tilde{r}_N))$$

where O_i represents the number of times particle i is selected.

- 11: **Update and Return** the list *resampled_indices*
-

4.6 Applications

The experiments are conducted in a similar setting to those in Chapter 3, but here we use only stochastic volatility estimators derived from the MSV model. Additionally, for

comparison we employ both the LW shrinkage approach (Ledoit & Wolf 2004a) and the sample covariance matrix when feasible.

The selection of the SMC² algorithm for the MSV model was driven by the need to simultaneously estimate both parameters and latent states in a high-dimensional, non-linear system. Compared to traditional MCMC approaches, SMC² offers better scalability in sequential settings and allows for online updating of the posterior distribution. The inclusion of adaptive step sizes, rejuvenation steps, and reparametrization ensures stability and computational efficiency, particularly for covariance structures like $\Sigma_{\epsilon\epsilon}$ and $\Sigma_{x_t|x_{t-1}, y_{t-1}}$. These properties are critical in financial applications, where rapid shifts in market conditions require responsive models. The design choices in Table 4.3 reflect a practical balance between convergence quality and computational cost, enabling consistent performance across different datasets and portfolio horizons.

The main objective is to determine the optimal allocation $\mathbf{w}_{T+K}$ across 30 stocks. The steps we follow are as follows:

- Process training data set from $\mathbf{y}_{1:T}$ to $\mathbf{z}_{1:T}$.
- Estimate posterior distribution $p(\theta|\mathbf{z}_{1:T})$ using SMC².
- Simulate 10,000 paths for K future time steps, acquire $\mathbf{z}_{T+1:T+K}^{1:10000}$, and calculate both $\widehat{\text{Cov}}(\mathbf{z}_{T+k})$ and $\widehat{\mathbb{E}}[\mathbf{z}_{T+k}]$.
- Bring simulated data set from $\mathbf{z}_{T+1:T+K}^{1:10000} \rightarrow \mathbf{y}_{T+1:T+K}^{1:10000}$, using (4.10).
- Calculate both $\widehat{\text{Cov}}(\mathbf{y}_{T+k})$ and $\widehat{\mathbb{E}}[\mathbf{y}_{T+k}]$ via (4.13).
- Apply these estimators to the cost function to find the optimal stock allocation $\mathbf{w}_{T+K}$.
- Compute performance metrics and compare the results.

In this analysis, we use only two cost functions, specifically equations (3.1b) and (3.3b). Short selling is not allowed, and each stock's allocation is capped at 10%.

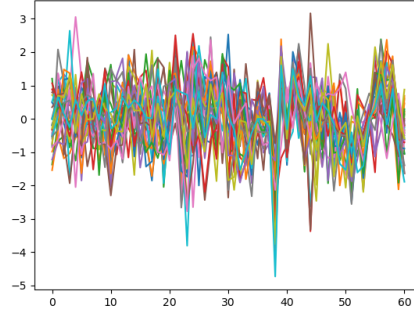
To initialize SMC² algorithm, we must predefine the number of particles for N_θ and N_x as well as the number of rejuvenation steps N_r . The specific values used in all experiments are listed in Table 4.3:

	Application 1	Application 2
Number of θ -particles	$N_\theta = 5000$	$N_\theta = 4000$
Number of x -particles	$N_x = 1000$	$N_x = 1000$
Number of of rejuvenation steps	$N_r = 3$	$N_r = 3$

Table 4.3: Setting the number of particles and rejuvenation steps for SMC² per application.

4.6.1 Case Study 1

In this chapter we train one data set that spans from 04/01/2013 to 30/09/2013. The data set consists of the log-returns for each stock that is a member of Dow Jones 30. The log-returns are calculated every three days, resulting in 61 data points. Having estimated θ for the whole time span, we use the trained subsets of the first 15, 30, 45, and 61 data points and we run simulations to calculate the out-of-sample portfolio P&L over a forward period of 15 time points (approximately two months and a week). According to the methodology discussed in section 4.4.1, the underlying data set with the preprocessing is represented in Figure 4.1.



(a) Standardization per stock after capping and flooring.

Figure 4.1: Dow Jones 30 stock three-days log-returns, from 04/01/2013 to 30/09/2013

Prior Illustration

By sampling θ_1^i (line 1 in algorithm 4) according to the priors outlined in table 4.1, we ensure positive definite covariance matrices $\Sigma_{\epsilon\epsilon}$, $\Sigma_{\eta\eta}$, and $\Sigma_{x_t|x_{t-1}, y_{t-1}}$, resulting in the prior histograms shown in Figure 4.2. One can observe that $corr_{\epsilon\epsilon}$ and $corr_{\eta\eta}$ priors despite of following uniform distributions, they are right skewed due to the filter applied on sampling for positive definite covariance matrices.

Estimating $p(\theta|\mathbf{z}_{1:T})$ Sequentially in Time

Since we initialize the prior on θ , we sequentially use the training data $\mathbf{z}_{1:T}$ over time to estimate the final posterior distribution $p(\theta|\mathbf{z}_{1:T})$. For instance in Appendix B, Section D.1 shows how the posterior of σ_ϵ^2 converges over time. Here, we observe some key behaviors of the SMC² algorithm in this applied context using Figure 4.3:

- **Convergence Speed:** The algorithm converges efficiently. By time $t = 8$, the Θ -space has already narrowed significantly, and the posterior appears close to convergence.

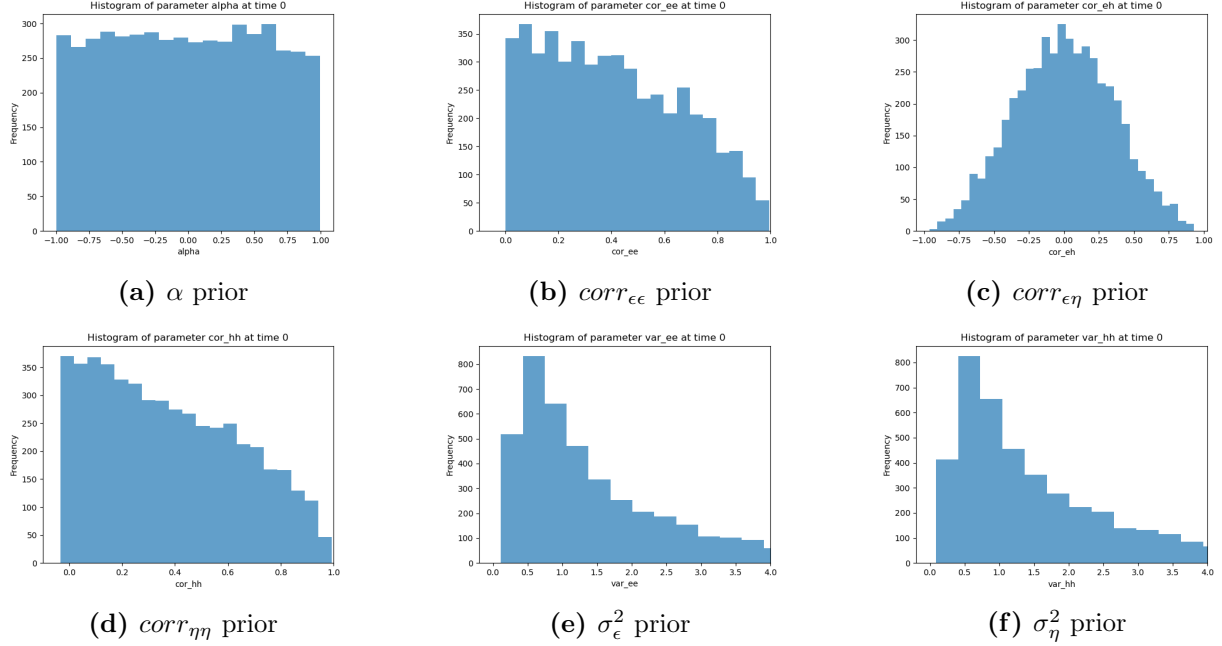


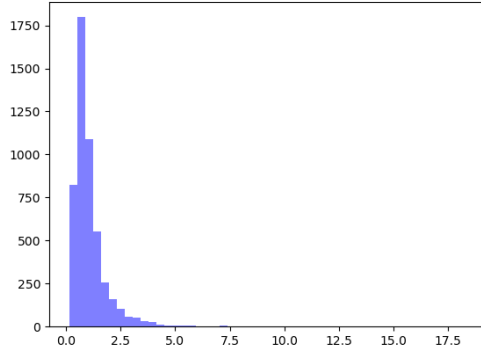
Figure 4.2: θ Prior

- **Smooth Progression:** The algorithm moves smoothly as long as there are no substantial jumps in stock prices.
- **Response to Large Price Jumps:** When substantial price jumps occur, the algorithm takes several steps to recalibrate. For instance, at $t = 39$ a significant downward stock price shift temporarily destabilizes the algorithm, causing minimal particle degeneracy. It takes approximately three time steps for the algorithm to regain stability. In such situations, increasing the number of rejuvenation steps might be necessary until the price action normalizes. In our case, no such adjustment was required.

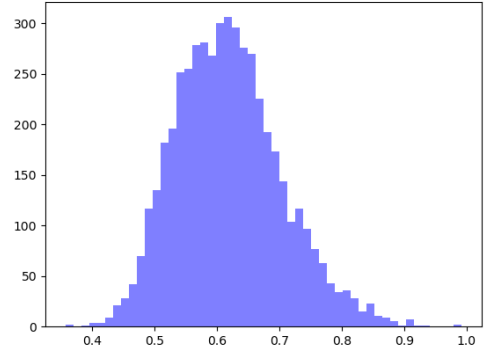
SMC² Diagnostics

Continuous monitoring of the algorithm's "health" at each time step is crucial for diagnosing potential issues and applying corrective measures to counteract particle impoverishment and degeneracy. Below, we outline the diagnostics used to track the algorithm's progress.

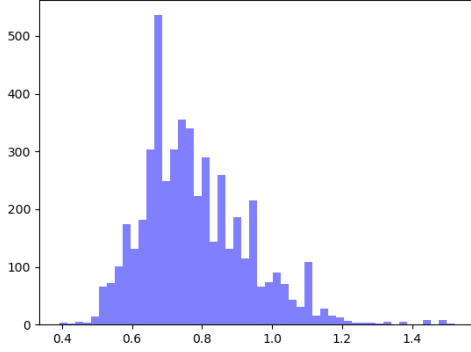
- **θ Posterior Histograms:** Sequential examination of histograms for one or more θ parameters, as illustrated in Section D.1.
- **θ -Particles Scatter Plots:** Scatter plots of θ -particles provide insight into their movement before and after rejuvenation steps. Ideally, particles should diverge from the 45-degree line between these steps. An example of this behavior is shown in Figure 4.4, depicting particle deviation at time $t = 5$ before and after the rejuvenation step.



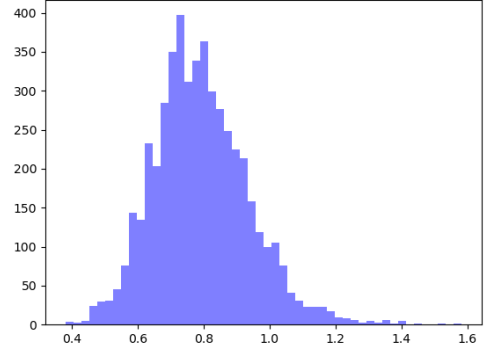
(a) Histogram $\sigma_\epsilon^2, t = 1$



(b) Histogram $\sigma_\epsilon^2, t = 8$



(c) Histogram $\sigma_\epsilon^2, t = 39$



(d) Histogram $\sigma_\epsilon^2, t = 43$

Figure 4.3: Sequential Posterior Estimation of σ_ϵ^2 , at time $t = 1, 8, 39, 43$

- **Acceptance Ratio & Adaptive Step Size:** The acceptance ratio is a widely recognized metric indicating the algorithm's progress during rejuvenation steps. It represents the percentage of proposals accepted in a single rejuvenation step, as outlined in Algorithm 5. Figure 4.5 illustrates the acceptance ratio (blue-dashed line) during the last rejuvenation step alongside the adaptive step size for each parameter. When the acceptance ratio remains within 0.25-0.6, the algorithm operates effectively. Ratios below 0.25 signal potential particle degeneration, and sustained values under 0.20 can lead to severe algorithmic issues. For instance, at time $t = 39$, when a stock price jump occurs, the acceptance ratio drops sharply from approximately 0.6 to 0.2. Fortunately, as the algorithm adapts, the acceptance ratio improves. Another observation is that as the algorithm converges, the step size per parameter decreases. However, the σ_ϵ^2 parameter demonstrates sensitivity to heightened stock price volatility, as expected, given its role in modeling market volatility.
- **Number of Proposals Yielding Positive Definite Matrices for $\Sigma_{\epsilon\epsilon}$, $\Sigma_{\eta\eta}$, and $\Sigma_{x_t|x_{t-1},y_{t-1}}$:** In Algorithm 5, we continuously track the number of proposals that result in positive definite underlying matrices. Ideally, this number should increase as the algorithm progresses. Figure 4.6 illustrates that, initially, 4700 out of 5000 proposals yield positive definite matrices. However, within a few time steps, this number approaches nearly 5000, with minor fluctuations observed subsequently.
- **Effective Sample Size (ESS):** ESS quantifies the extent of weight degeneracy within the particle set and indicates the number of particles effectively contributing to the posterior distribution and is defined as:

$$\text{ESS} = \frac{1}{\sum_{i=1}^N W_i^2}, \quad (4.19)$$

N is the total number of particles and W_i is the normalized weight of the i th particle, such that $\sum_{i=1}^N W_i = 1$. A low ESS suggests that a small number of particles dominate the weight distribution.

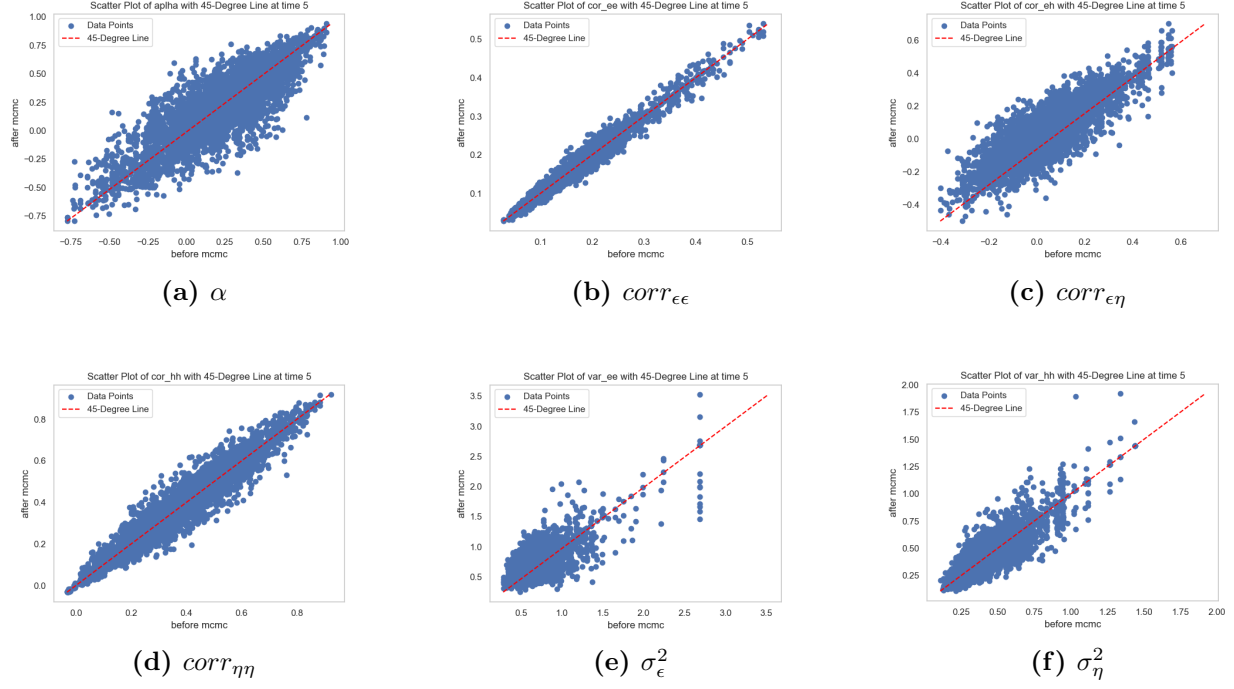


Figure 4.4: θ -particles scatter plots before and after rejuvenation steps at time $t = 5$.

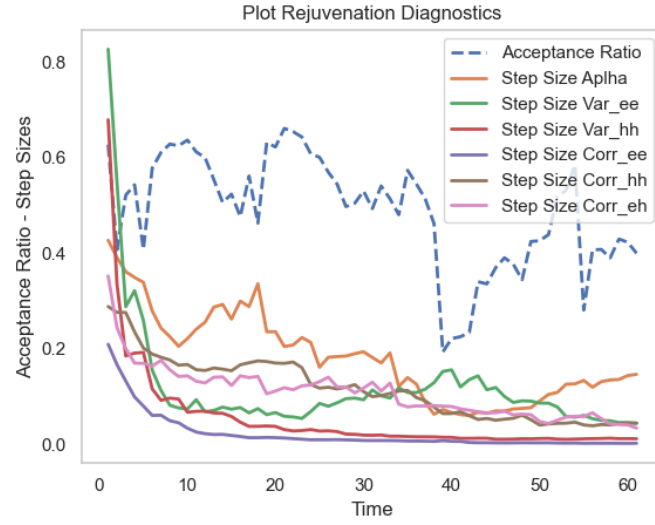


Figure 4.5: Acceptance ratio, contrasting adaptive step size per θ parameter

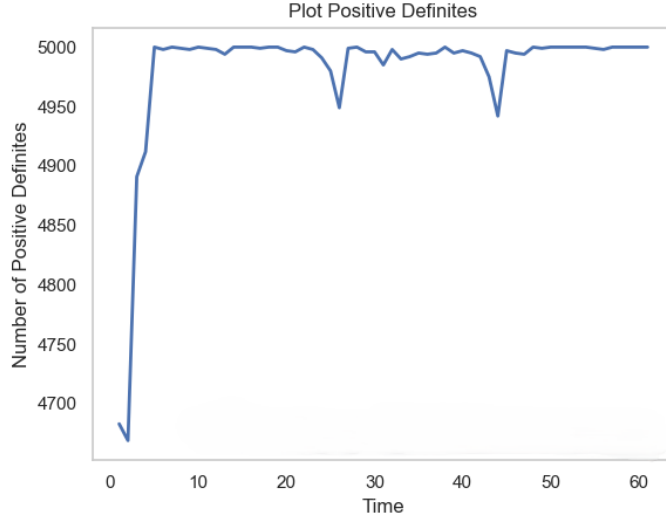


Figure 4.6: Number of particles that constitute positive definite $\Sigma_{x_t|x_{t-1},y_{t-1}}$ matrices.

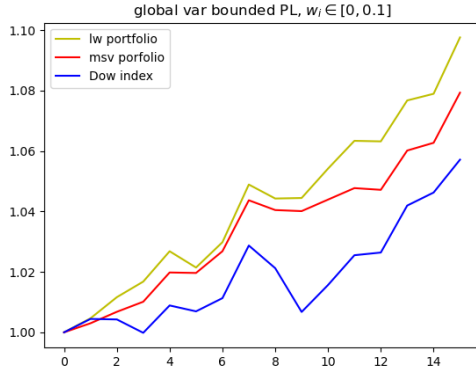
Predicted Credible Intervals and Realized P&Ls

As outlined in the introduction of Section 4.6, following the computation of the SMC² results, we obtain the realized P&L trajectories and forecast credible intervals (CI) for the optimal portfolios $\mathbf{w}_{T+K}$ under the two cost functions for each training dataset. Additionally, we derive the respective distributions of the terminal values PL_{T+K} and annualized Sharpe Ratios SR_{T+K} for the MSV optimal portfolio when \$1 is invested, as described in Chapter 3. The visualizations corresponding to each dataset and cost function are collectively presented in the figures located in Appendix D.3.

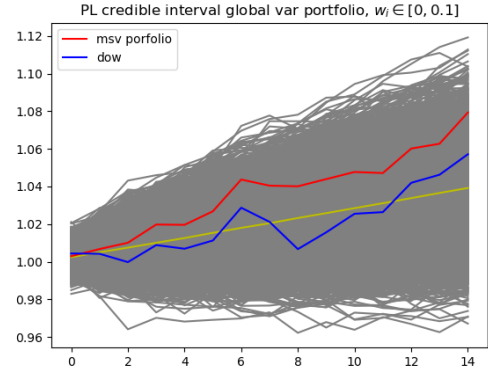
For example, using the first 15 data points as the training set, Figure 4.7a illustrates the realized P&L trajectory of the optimal portfolio under the global minimum variance cost function, as defined in (3.1b), utilizing the MSV and Ledoit-Wolf covariance estimators. In Appendix D.3, the realized P&L trajectory derived from the empirical covariance is also provided for the remaining datasets.

Regarding the CI, a P&L cloud comprising 10,000 simulated P&L trajectories is generated based on the MSV optimal allocation $\mathbf{w}_{T+K}$. This P&L cloud is depicted in Figure 4.7b, alongside the realized P&L trajectories of the MSV optimal portfolio and the Dow Jones index, assuming \$1 is invested in each. From these simulated trajectories, the distributions of the terminal values and Sharpe Ratios for the MSV optimal portfolio, projected 15 days ahead, are derived. These distributions are visualized in Figures 4.7c and 4.7d.

Finally, Figure 4.7d showcases the optimal stock allocations for the MSV and Ledoit-Wolf estimators. The overall performance metrics are summarized in Table 4.4, in a manner consistent with the discussions in Chapter 3.

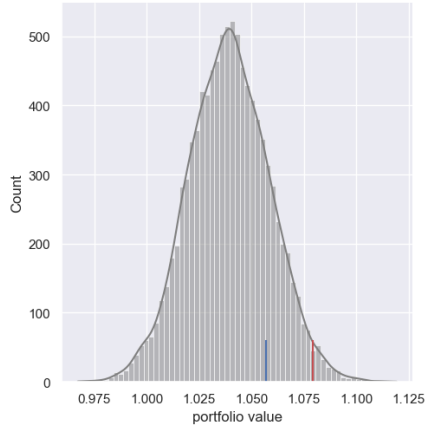


(a) out of sample realized P&L trajectory



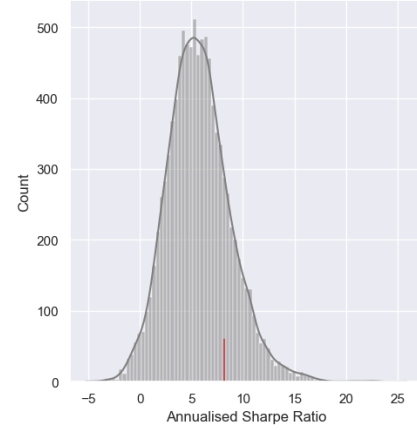
(b) credible interval forecast

CI 66th day portfolio value distribution for mean-variance MSV portfolio, $w_l \in [0, 0.1]$

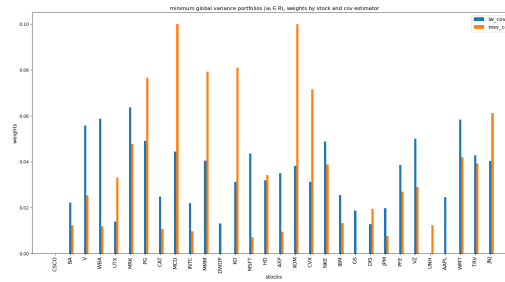


(c) \$1 P&L distribution forecast

CI Sharpe Ratio distribution for mean-variance MSV portfolio, $w_l \in [0, 0.1]$



(d) SR distribution forecast



(e) optimal allocation

Figure 4.7: Global minimum variance optimal portfolio results when the first 15 data points are trained and calculate the out-of-sample P&L over a forward period of 15 time points.

	Mean R	Std. Dev R	Sharpe Ratio	L2	Ratio
Dow	31.12	7.51	4.14	-	-
lw_global	52.15	6.19	8.42	0.20	42.10
msv_global	42.74	5.22	8.19	0.25	32.76
lw_meanvar	52.17	6.26	8.34	0.20	41.70
msv_meanvar	47.89	5.27	9.09	0.24	37.88

Table 4.4: First 15 data points training set, fifteenth out-of-sample optimal portfolio realized performance.

4.6.2 Case Study 2

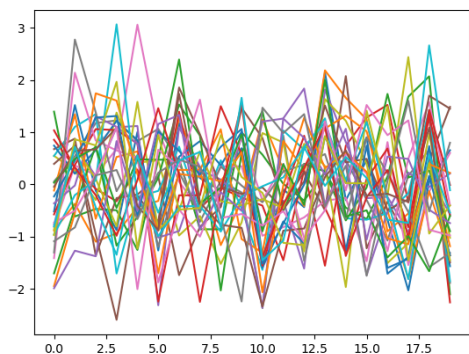
For the second experiment, the dataset spans from 04/01/2013 to 31/12/2013 and is divided into four quarterly periods: 04/01/2013 to 29/03/2013, 01/04/2013 to 31/06/2013, 01/07/2013 to 30/09/2013, and 01/10/2013 to 31/12/2013. Here, the algorithm is trained independently for each quarter, and after estimating θ , we forecast the portfolio P&L for the following quarter. As in the first experiment, the log-returns are computed every three days, so each quarter consists of roughly 21 data points. The post-processed data set for each quarter is represented in Figure 4.8.

In Appendix E.2, we provide a comprehensive presentation of all P&L results for each quarter. In this section, we briefly showcase selected diagnostics and P&L results related to the dataset from the third quarter. Figure 4.9 displays three diagnostic sub-figures. The first sub-figure illustrates the number of proposals yielding positive definite $\Sigma_{x_t|x_{t-1}, y_{t-1}}$ matrices during the rejuvenation step. Notably, after period $t = 8$, almost all proposals meet the positive definiteness criterion. The second sub-figure depicts the ESS observed during training, while the third sub-figure demonstrates a smooth upward trend in the acceptance ratio and rapid convergence of step sizes. Together, these results confirm that the sampling methodology developed and implemented in the rejuvenation step ensures high-quality θ proposals and efficient convergence to the posterior distribution.

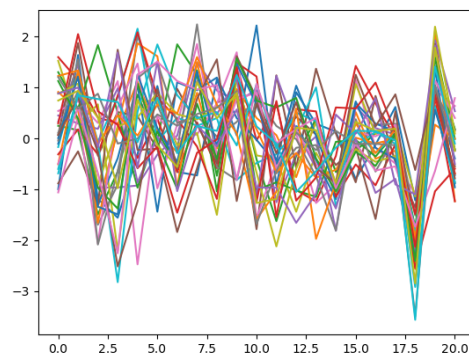
Finally, Figure 4.10 and Table 4.5 present the P&L results obtained after training the third quarter dataset, following a format consistent with the approach described in Case Study 1.

Empirical Substantiation: The performance claims made in this chapter are supported by detailed diagnostics and quantitative results. In Case Study 1, for example, the MSV model under the mean-variance cost function achieves a Sharpe Ratio of 9.09, significantly outperforming the Dow Jones benchmark at 4.14, as shown in Table 4.4. Additionally, the realized P&L trajectories and credible intervals in Figure 4.7 demonstrate the predictive accuracy of the MSV-based approach, with actual portfolio returns consistently lying within the forecast bounds. Case Study 2 further supports these findings, where despite increased

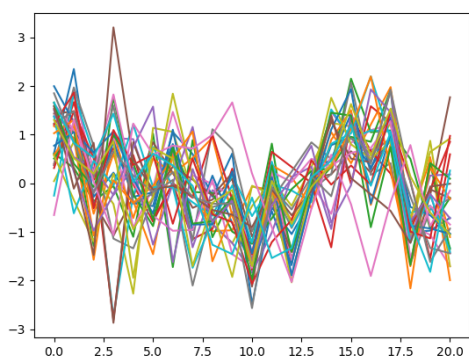
volatility in Q3, the MSV model delivers a Sharpe Ratio of 3.16, close to the best-performing Ledoit-Wolf model (3.17), and within more stable credible bounds (Figure 4.10). These results affirm the reliability of MSV-based forecasts in both stable and moderately volatile environments.



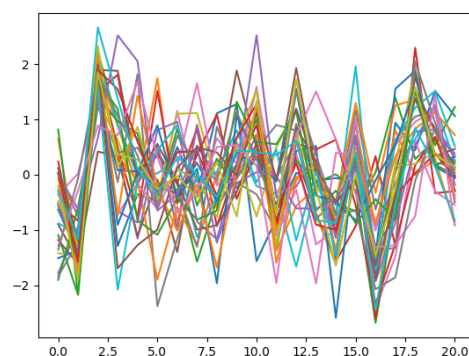
(a) First quarter



(b) Second quarter

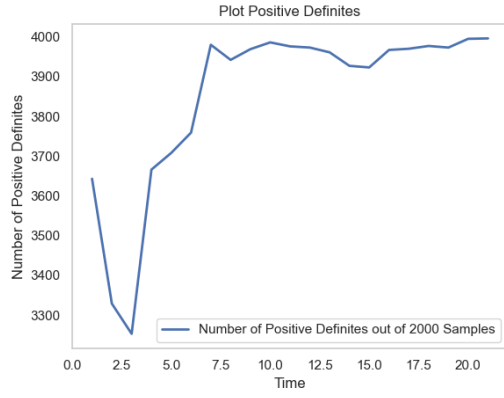


(c) Third quarter

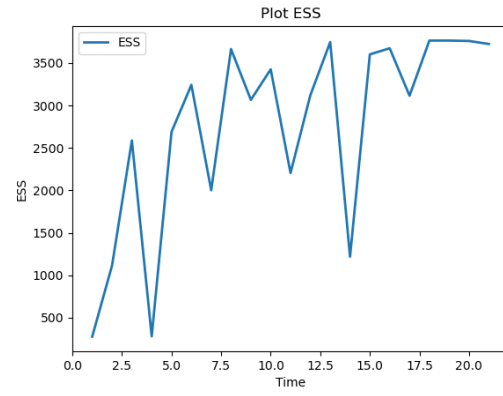


(d) Fourth quarter

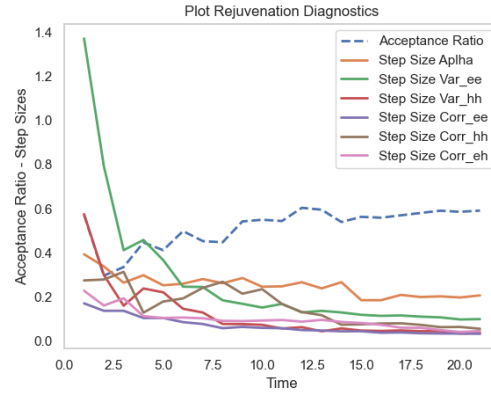
Figure 4.8: Standardization per stock after capping and flooring for each quarter.



(a) Number of particles that constitute positive definite $\Sigma_{x_t|x_{t-1}, y_{t-1}}$ matrices



(b) Effective Sample Size



(c) Acceptance ratio, contrasting adaptive step size per θ parameter

Figure 4.9: Third quarter training diagnostics.

	Mean R	Std. Dev R	Sharpe Ratio	L2	Ratio
Dow	34.9	11	3.17	-	-
lw_global	32.8	9.87	3.32	0.27	12.3
msv_global	29.85	10	2.99	0.31	9.65
lw_meanvar	32.16	10.14	3.17	0.27	11.74
msv_meanvar	33.01	10.44	3.16	0.29	10.9

Table 4.5: Third quarter training set, 21 periods out-of-sample optimal portfolio realized performance.

4.7 Conclusions

In this chapter, we presented a methodology to train the SMC² algorithm for estimating covariance and expected returns within the Bayesian framework for the MSV model. These estimators were applied to the defined cost functions, and we utilized multiple datasets, based on Dow Jones stocks, to construct optimal portfolios. Additionally, we employed the Ledoit-Wolf and empirical estimators to derive their respective optimal portfolios.

In our preliminary investigation we found that naive implementation of SMC² to MSV was too inefficient to be a realistic option in practice. The primary contribution of this chapter is that reparametrizing on θ and leveraging the structure of the Σ matrix, we enhanced the robustness, efficiency and performance of SMC².

In particular, Corollary 1 acts as a cornerstone to sample θ from the prior and generate proposals during the rejuvenation step. This ensured that $\Sigma_{\epsilon\epsilon}$, $\Sigma_{\eta\eta}$, and $\Sigma_{x_t|x_{t-1},y_{t-1}}$ were positive definite matrices; a critical requirement for sampling x -particles. The details are in Appendix A, where using careful linear algebra we can significantly expedite computations, particularly for incremental likelihoods $\hat{p}_{\theta^i}(y_t|y_{1:t-1})$, which are frequently evaluated within SMC². In the rejuvenation step, having reparametrized θ , we managed to restructure all samplers to efficiently explore the Θ -space. We also implemented adaptive step sizes, which ensured qualitative proposals and maintained the acceptance ratio within the range $[0.23, 0.6]$. This was crucial to mitigating particle degeneracy and impoverishment. Finally, leveraging the results in Appendix A, we successfully vectorized portions of the SMC² algorithm. This included simultaneous sampling of all θ and x -particles, as well as concurrent computation of x -weights for each θ -particle. These optimizations greatly reduced computational overhead and improved the algorithm’s stability.

In conclusion, all the estimators exhibited strong performance, frequently outperforming the Dow Jones index, as demonstrated by the collective P&L tables and figures. Additionally, the MSV model provides a significant advantage: the ability to generate credible intervals for the future performance of the optimal portfolio’s P&L. This feature is highly valuable for

practitioners, enabling effective and continuous risk management over time. However, this advantage comes at the expense of increased computational cost.

The datasets used in this chapter are derived from Dow Jones 30 stocks over the year 2013, a period of relatively stable market conditions. While this provides a reliable benchmark for performance evaluation, it also limits the generalizability of the results. The conclusions drawn may not directly apply to markets characterized by higher volatility, structural breaks, or reduced liquidity. For instance, emerging markets or alternative asset classes such as bonds or commodities may exhibit different statistical properties that challenge the assumptions embedded in the MSV model and the SMC² algorithm. Future studies should aim to test the robustness of the methodology under diverse economic scenarios, including crisis periods, regime changes, and cross-asset settings.

Chapter 5

Conclusions & Future Work

5.1 Conclusions

This study investigates portfolio optimization, employing advanced statistical and computational methodologies to improve risk management and investment outcomes. By leveraging empirical analysis with Dow Jones stocks, it introduces innovative approaches to estimating covariance matrices and expected returns, applying both frequentist and Bayesian methods. A significant focus is placed on enhancing the SMC² algorithm within the Bayesian framework of the Multivariate Stochastic Volatility (MSV) model.

The research demonstrates that the largest covariance eigenvalues of stock returns spike during transitions from bull to bear markets. This characteristic enables classification of market conditions using machine learning techniques such as supervised or reinforcement learning. These models empower investors to adapt their portfolios, hedging against potential downturns through adjustments to portfolio Beta and strategic short positions in index futures.

Efficient estimation methods are explored, with frequentist approaches such as empirical covariance approximations and the Ledoit-Wolf estimator showing strong performance. These methods yield more sparse portfolios with concentrated allocation weights, making them computationally efficient but less diversified, which can pose challenges for risk management. Bayesian methods, while computationally intensive due to MCMC calibration, provide better diversification and predictive accuracy, offering credible intervals for future portfolio performance.

The study adapts the SMC² algorithm for the MSV model, addressing initial inefficiencies through reparametrization and optimization of the Σ matrix structure. These enhancements improve robustness and computational efficiency, ensuring positive definiteness of key matrices and facilitating efficient sampling. Further optimizations include adaptive step sizes and vectorization, enabling parallel processing and reducing computational overhead.

Portfolios constructed using these methods consistently outperform the Dow Jones index, as demonstrated through P&L metrics and visual comparisons. The Bayesian models provide the added benefit of generating credible intervals for future portfolio performance,

supporting effective risk management. However, this advantage is accompanied by increased computational costs.

This work bridges theoretical advancements and practical applications, presenting a spectrum of tools for portfolio optimization tailored to different investor needs and computational constraints. The integration of SMC² with novel optimization techniques establishes a foundation for future research aimed at developing more efficient and scalable models, advancing both academic understanding and practical portfolio management.

5.2 Future Work

In addition to the cost functions explored in this work, future research could investigate alternative cost functions that directly target the control of diversification or aim to maximize expected returns.

For instance, the mean-variance model often encounters challenges, particularly when short selling is permitted. The optimal solution vector $\mathbf{w}_*$ may include elements w_i that are disproportionately larger than others, leading to allocations concentrated in a few stocks and violating the principle of diversification. To address this, a regularization term can be introduced to the objective function:

$$\mathbf{w}_* = \arg \min_{\mathbf{w}} \mathbf{w}^T \Sigma \mathbf{w} - \lambda f(\mathbf{w}), \quad (5.1)$$

where λ is a regularization factor determined through cross-validation, penalizing large w_i values to smooth the allocation. Larger λ values impose stronger regularization. Different forms of $f(\mathbf{w})$ have been proposed (Bishop 2006):

- $f(\mathbf{w}) = \|\mathbf{w}\|_1 = \sum_{i=1}^d |w_i|$ (Lasso)
- $f(\mathbf{w}) = \|\mathbf{w}\|_2^2 = \sum_{i=1}^d w_i^2$ (Ridge)

Ridge regression reduces the magnitude of w_i , while Lasso produces sparse solutions. The role of such regularization in high-dimensional portfolios has been discussed by Gandy & Veraart (2013), and future work could extend these models with numerical optimization techniques, constraining the norms to specified thresholds (DeMiguel et al. 2009).

Another promising direction comes from Lai et al. (2011), who demonstrated the potential to directly maximize the expected portfolio return while penalizing portfolio variance:

$$\mathbf{w}_* = \arg \max_{\mathbf{w}} E(\mathbf{w}^T \mathbf{r}_{t+1}) - \lambda \text{Var}(\mathbf{w}^T \mathbf{r}_{t+1}), \quad (5.2)$$

where $\mathbf{r}_{t+1}$ is the vector of stock returns at time $t + 1$, which are not necessarily independent or identically distributed. This approach aligns with methods in control theory (Zhou & Li 2000). Equation (5.2) can also incorporate a regularization term as in (5.1).

Model performance depends heavily on the chosen training and testing datasets, posing challenges for generalizing conclusions. A combined investment strategy could allocate initial

wealth based on model performance, while optimizing individual allocations. Such a strategy would operate sequentially, using metrics such as the Sharpe Ratio, historical predictions, portfolio diversity, and drawdown statistics to dynamically update weights as new data becomes available.

For frequentist methods, updating estimates with new data is straightforward. However, for volatility models, sequential Bayesian learning remains complex (Kantas et al. 2015). A potential extension could involve combining models via a particle method algorithm, assigning wealth to portfolios constructed from various models and cost functions. Weights could be computed using a Softmax function of performance metrics, and the resampling-like process would probabilistically allocate wealth. Outputs of the SMC² method could serve as inputs to this particle-based allocation algorithm.

Finally, while some models may exhibit superior P&Ls, the most promising direction is developing a model capable of generating hedging signals during the investment horizon. Such a model would partially or fully hedge the portfolio when the market shifts to bearish momentum. The portfolio itself could be treated as an asset, with its log-returns modeled using a stochastic volatility framework like that in Section 3.4.1 (3.25). Using SMC², posterior parameter distributions could reveal, in real time, the need for hedging and its intensity. Simulations could explore transitions from bullish to bearish returns, identifying behavior in posterior distributions that signal when hedging should commence.

Appendix A

Σ Matrix

A.1 Preliminaries

Here we are going to derive some important results, which are crucial in the MSV models. These results are used to greatly accelerate computations, when we calculate the likelihood $\hat{p}(\mathbf{y}_{1:T}|\theta)$, to derive prior samples and proposals in the rejuvenation step efficiently. Let's consider the symmetric matrix Σ of dimension d that on and off the main diagonal lie the elements a and b respectively.

$$\Sigma = b\mathbf{M}_d + (a - b)\mathbf{I}_d = \begin{pmatrix} a & b & \cdots & b \\ b & a & \cdots & b \\ \vdots & \vdots & \ddots & \vdots \\ b & b & \cdots & a \end{pmatrix}. \quad (\text{A.1})$$

Here $\mathbf{M}_d$ is a d dimensional matrix where every entry is equal to one and $\mathbf{I}_d$ is the d dimensional identity matrix. Σ is a linear combination of matrices $\mathbf{M}_d$ and $\mathbf{I}_d$. Additions and multiplications among matrices of this type, also return matrices that are linear combinations of $\mathbf{M}_d$ and $\mathbf{I}_d$. This is due to the fact that $\mathbf{M}_d$ to the power of n is a linear function of $\mathbf{M}_d$, that is $\mathbf{M}_d^n = d^{n-1}\mathbf{M}_d$. Further more, any matrix multiplication and summation among matrices that have the algebraic representation of (A.1) will return a matrix that has exactly the same algebraic representation. Taking advantage of this particular property, in the following paragraphs we define a number of analytical formulas that we are interested in.

Proposition 1. *Let*

$$\mathcal{A} = \{A \in \mathbb{R}^{d \times d} : a\mathbf{M}_d + b\mathbf{I}_d\} \quad (\text{A.2})$$

for some $a, b \in \mathbb{R}$. Then if $\Sigma \in \mathcal{A}$, $\alpha \in \mathbb{R}$ and $\mathbf{B} \in \mathcal{A}$, we have that $\alpha\Sigma \in \mathcal{A}$ and $\Sigma\mathbf{B} \in \mathcal{A}$.

Proof. The first statement is obvious from the definition of $\mathcal{A}$. For the second statement, note that $\mathbf{M}_d^2 = d\mathbf{M}_d$ and the result follows. $\square$

A.2 The eigenvalues of Σ and their support

Proposition 2. *Let $\Sigma \in \mathcal{A}$, the Σ matrix has d eigenvalues, from which there exist one unique and $d - 1$ identical eigenvalues given from the following equations respectively:*

$$\begin{aligned}\lambda_1 &= a + (d - 1)b \\ \lambda_2 &= a - b\end{aligned}\tag{A.3}$$

Proof. Starting from the characteristic equation of Σ we have: $\det(\Sigma - \lambda \mathbf{I}) = 0$. Then making use of equation (A.1) we get:

$$\begin{aligned}\det(b\mathbf{M}_d + (a - b)\mathbf{I}_d - \lambda\mathbf{I}_d) &= \\ \det(b\mathbf{M}_d + (a - b - \lambda)\mathbf{I}_d) &= \\ (a - b - \lambda)^d \det\left(\frac{b}{a - b - \lambda}\mathbf{M}_d + \mathbf{I}_d\right) &= \\ (a - b - \lambda)^d \det(\mathbf{ii}^T + \mathbf{I}_d) &= \\ (a - b - \lambda)^d (\mathbf{i}^T \mathbf{i} + 1) &= \\ (a - b - \lambda)^d \left(\frac{db}{a - b - \lambda} + 1\right) &= \\ (a - b - \lambda)^{d-1} (db + (a - b - \lambda)) &= \\ (a - b - \lambda)^{d-1} (a + (d - 1)b - \lambda) &= \\ \lambda_1 \lambda_2^{d-1} &= 0.\end{aligned}$$

where in third equality we set $\mathbf{i}$ a d -dimensional column vector where each element equals to $\sqrt{\frac{b}{a-b-\lambda}}$ and then we made use of Sylvester's determinant theorem.

Concluding that the characteristic equation has one root $\lambda_1 = a + (d - 1)b$ and $d - 1$ identical roots $\lambda_2 = a - b$ $\square$

Corollary 1. *The matrix Σ is positive definite if and only if $a > b$ and $b \in \left[-\frac{a}{d-1}, a\right]$.*

Proof. According to Proposition 2, Σ possesses one eigenvalue λ_1 and $d - 1$ instances of the eigenvalue λ_2 . Therefore, Σ is positive definite if both eigenvalues are positive, which implies that $b \in \left[-\frac{a}{d-1}, a\right]$ and vice versa. $\square$

Corollary 2. *If Σ represents a covariance matrix of a vector $\mathbf{y}$, then all the random variables y_i that are elements of vector $\mathbf{y}$ have common marginal variance $\text{Var}(y_i) = a$, covariance $\text{Cov}(y_i, y_j) = b$ and correlation $\text{Corr}(y_i, y_j) = \frac{b}{a}$ for $i \neq j$. Moreover it is $a \in [0, \infty]$, $b \in \left[-\frac{a}{d-1}, a\right]$ and $\text{Corr}(y_i, y_j) \in \left[-\frac{1}{d-1}, 1\right]$*

Proof. The proof follows since $\Sigma \in \mathcal{A}$ $\square$

A.3 Inverting Σ

Theorem 1. Let $\Sigma = b\mathbf{M}_d + (a - b)\mathbf{I}_d$ then Σ^{-1} is given by:

$$\Sigma^{-1} = \frac{-b}{\lambda_1\lambda_2}\mathbf{M}_d + \frac{1}{\lambda_2}\mathbf{I}_d, \quad (\text{A.4})$$

where the λ_1 and λ_2 are the two distinct eigenvalues of Σ , given from (A.3).

Proof. Suppose Σ^{-1} has a similar structure as the one of Σ matrix and we can write Σ^{-1} as a linear combination of matrices $\mathbf{M}_d$ and $\mathbf{I}_d$.

$$\Sigma^{-1} = x_1\mathbf{M}_d + x_2\mathbf{I}_d, \quad (\text{A.5})$$

where x_1 and x_2 are unknown scalars that need to be computed. We require $\Sigma\Sigma^{-1} = \mathbf{I}_d$, substituting we get:

$$(b\mathbf{M}_d + (a - b)\mathbf{I}_d)(x_1\mathbf{M}_d + x_2\mathbf{I}_d) = \mathbf{I}_d.$$

Making use of $\mathbf{M}_d^2 = d\mathbf{M}_d$ we have:

$$\begin{aligned} (dbx_1 + bx_2 + (a - b)x_1)\mathbf{M}_d + (a - b)x_2\mathbf{I}_d = \\ [((d - 1)b + a)x_1 + bx_2]\mathbf{M}_d + (a - b)x_2\mathbf{I}_d. \end{aligned}$$

Setting $((d - 1)b + a)x_1 + bx_2 = 0$, $(a - b)x_2 = 1$ and solving for x_1 and x_2 , we find:

$$\begin{aligned} x_1 &= \frac{-b}{(a + (d - 1)b)(a - b)} = \frac{-b}{\lambda_1\lambda_2}, \\ x_2 &= \frac{1}{a - b} = \frac{1}{\lambda_2} \end{aligned}$$

□

A.4 The determinant of Σ

Theorem 2. The determinant $\det(\Sigma)$ of matrix Σ has an analytical formula given by:

$$\det(\Sigma) = \lambda_1\lambda_2^{d-1}, \quad (\text{A.6})$$

where the λ_1 and λ_2 are the two distinct eigenvalues of Σ , given from (A.3).

Proof. Plugging (A.1) in determinant we get:

$$\det(\Sigma) = \det(b\mathbf{M}_d + (a - b)\mathbf{I}_d) = (a - b)^d \det\left(\frac{b}{a - b}\mathbf{M}_d + \mathbf{I}_d\right).$$

Set $\mathbf{i}$ a d -dimensional column vector where each element equals to $\sqrt{\frac{b}{a - b}}$, we can write the above equation as:

$$(a - b)^d \det(\mathbf{i}\mathbf{i}^T + \mathbf{I}_d).$$

Making use of Sylvester's determinant theorem, we get:

$$\begin{aligned}
& (a-b)^d(\mathbf{i}^T \mathbf{i} + 1) \\
&= (a-b)^d \left(\sum_{i=1}^d \frac{b}{a-b} + 1 \right) \\
&= (a-b)^d \left(\frac{db}{a-b} + 1 \right) \\
&= (a-b)^{d-1} (a + (d-1)b) = \lambda_1 \lambda_2^{d-1}.
\end{aligned}$$

□

A.5 Cholesky decomposition of Σ

The lower triangular matrix $\mathbf{L}_d$, which is part of $d \times d$ dimensional matrix Σ_d Cholesky's decomposition

$$\Sigma_d = \mathbf{L}_d \mathbf{L}_d^T \quad (\text{A.7})$$

must have the following representation:

$$\mathbf{L}_d = \begin{pmatrix} l_1 & \cdots & 0 & 0 & 0 & 0 \\ x_1 & l_2 & \cdots & 0 & 0 & 0 \\ x_1 & x_2 & l_3 & \cdots & 0 & 0 \\ x_1 & x_2 & x_3 & l_4 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ x_1 & x_2 & x_3 & x_4 & \cdots & l_d \end{pmatrix} \quad (\text{A.8})$$

This form denotes that all the elements under the diagonal element for each column of $\mathbf{L}$ are equal. This comes from the fact that all the elements of the Σ matrix that lie off the diagonal are equal to b . Also, the lower triangular matrix $\mathbf{L}_{d-1}$ of $d-1 \times d-1$ dimensional matrix Σ_{d-1} has the representation

$$\mathbf{L}_{d-1} = \begin{pmatrix} l_1 & \cdots & 0 & 0 & 0 & 0 \\ x_1 & l_2 & \cdots & 0 & 0 & 0 \\ x_1 & x_2 & l_3 & \cdots & 0 & 0 \\ x_1 & x_2 & x_3 & l_4 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ x_1 & x_2 & x_3 & x_4 & \cdots & l_{d-1} \end{pmatrix}. \quad (\text{A.9})$$

$\mathbf{L}_{d-1}$ matrix has the same elements as that of $\mathbf{L}_d$ if we eliminate the last row and column. Taking into account this representation, we can state the following theorem:

Lemma 1. *In the Cholesky decomposition, for the $d \times d$ dimensional matrix Σ_d , the matrix $\mathbf{L}_d$ is defined as a lower triangular matrix where its elements are typically defined:*

$$L_{ij} = \begin{cases} \sqrt{(a-b)(1 + \frac{b}{a+(i-2)b})}, & \text{if } i = j \\ \frac{1}{l_j} (b - \mathbf{x}_{j-1}^T \mathbf{x}_{j-1}), & \text{if } i > j \\ 0, & \text{if } i < j \end{cases}$$

where $\mathbf{x}_{j-1}$ denotes the column vector $\mathbf{x}_{j-1} = [x_1 \ x_2 \ \dots \ x_{j-1}]^T$ for $j \geq 2$ and $x_1 = \frac{b}{l_1}$.

Proof. From Theorem 2 we know that:

$$\det(\Sigma_d) = \det(\mathbf{L}_d \mathbf{L}_d^T) = \left(\prod_{i=1}^d l_i \right)^2. \quad (\text{A.10})$$

We deduce that

$$l_d^2 = \frac{\det(\Sigma_d)}{\det(\Sigma_{d-1})} = (a-b) \left(1 + \frac{b}{a+(d-2)b} \right). \quad (\text{A.11})$$

Recursively we calculate all the diagonal elements as:

$$L_{ii} = \sqrt{l_i^2} = \sqrt{(a-b) \left(1 + \frac{b}{a+(i-2)b} \right)} \quad (\text{A.12})$$

For elements below the main diagonal based on (A.7), we get:

$$\begin{aligned} l_1 x_1 &= b \\ x_1^2 + l_2 x_2 &= b \\ x_1^2 + x_2^2 + l_3 x_3 &= b \\ &\dots \\ x_1^2 + x_2^2 + \dots + x_{n-1}^2 + l_n x_n &= b. \end{aligned}$$

Setting $\mathbf{x}_{j-1} = [x_1 \ x_2 \ \dots \ x_{j-1}]^T$, we deduce that:

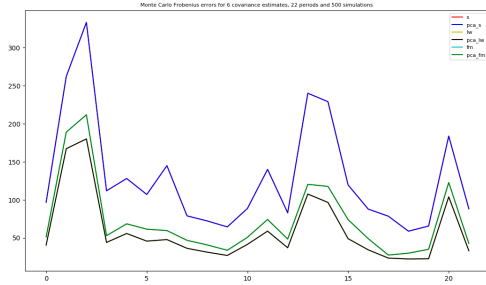
$$L_{ij} = x_j = \frac{1}{l_j} (b - \mathbf{x}_{j-1}^T \mathbf{x}_{j-1}) \quad (\text{A.13})$$

□

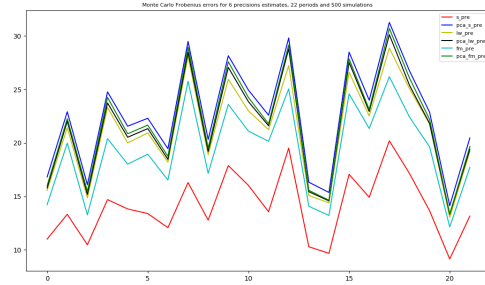
Appendix B

Chapter 2 Appendix

B.1 Average Frobenius Error for Covariance, Precision Estimation

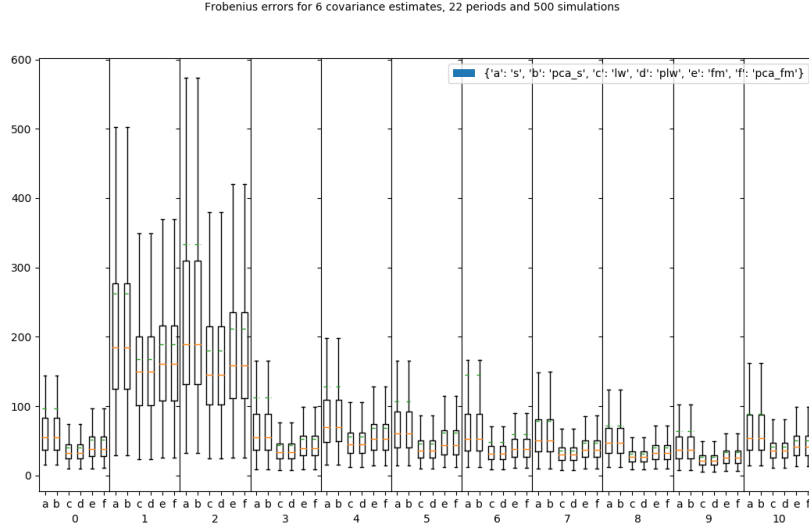


Average Frobenius error for covariance matrix estimation over 500 independent runs using simulated data representative of DJ30 with Student-t returns.

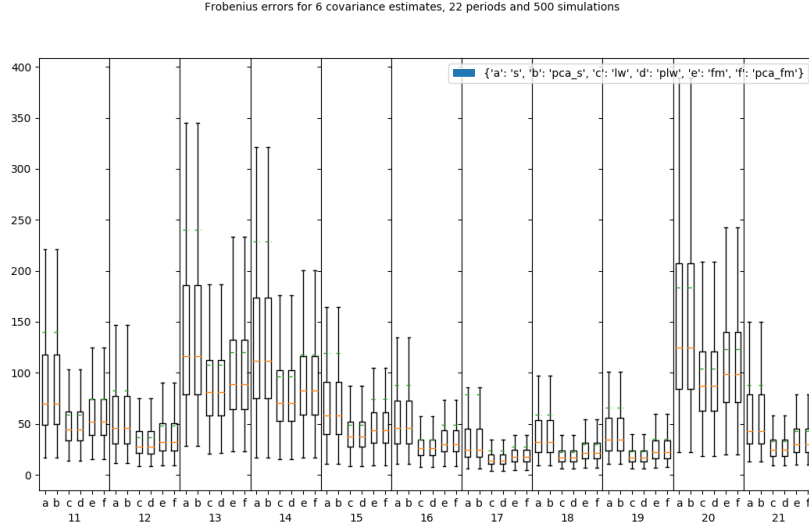


Average Frobenius error for precision matrix estimation over 500 independent runs using simulated data representative of DJ30 with Student-t returns.

Figure B.1: Performance comparison of covariance and precision matrix estimators (Sample covariance, Ledoit-Wolf, and Factor Model) when returns follow a Student-t distribution. Eigenvalues greater than one are used for PCA and precision estimators. Simulated DJ30 data is divided into 22 periods, each with 176 sampled returns. Covariance is estimated using the first 88 vectors; Frobenius error is evaluated using the remaining 88 vectors.

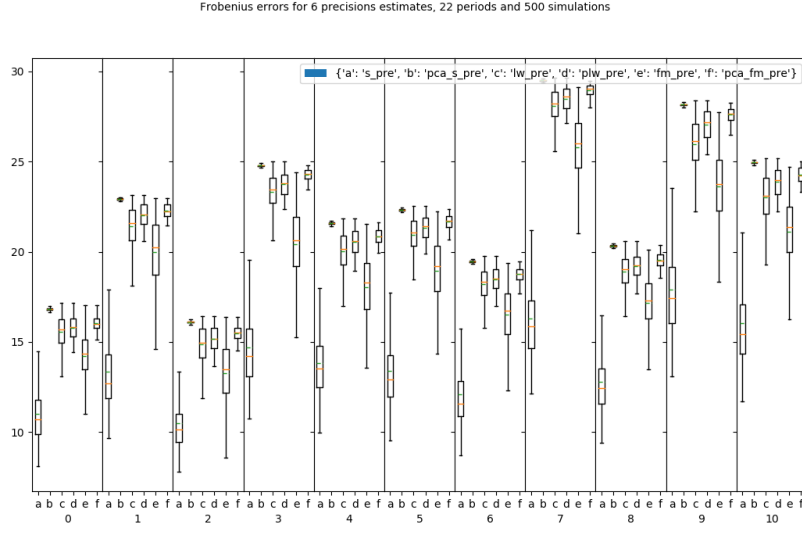


Boxplots of Frobenius error for covariance matrix estimation using simulated DJ30 data from a multivariate t-distribution with 3 degrees of freedom.

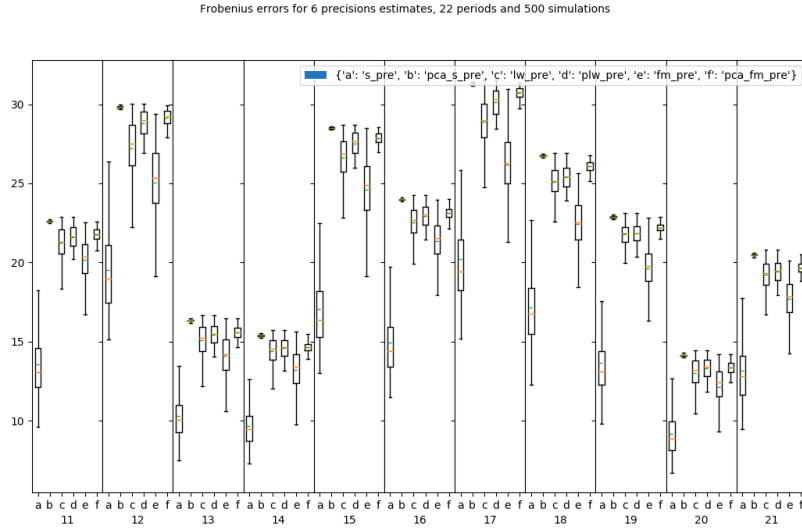


Boxplots of Frobenius error for covariance matrix estimation using simulated DJ30 data from a multivariate t-distribution with 3 degrees of freedom.

Figure B.2: Comparison of Frobenius error distributions for covariance matrix estimation methods (Sample covariance, Ledoit-Wolf, and Factor Model) using boxplots. Simulated DJ30 data is derived from a multivariate t-distribution with 3 degrees of freedom, and the eigenvalues greater than one are used for PCA and precision estimators.



Boxplots of Frobenius error for precision matrix estimation using simulated DJ30 data from a multivariate t-distribution with 3 degrees of freedom.



Boxplots of Frobenius error for precision matrix estimation using simulated DJ30 data from a multivariate t-distribution with 3 degrees of freedom.

Figure B.3: Comparison of Frobenius error distributions for precision matrix estimation methods (Sample covariance, Ledoit-Wolf, and Factor Model) using boxplots. Simulated DJ30 data is derived from a multivariate t-distribution with 3 degrees of freedom, and the eigenvalues greater than one are used for PCA and precision estimators.

Appendix C

Chapter 3 Appendix

C.1 ISV Model Results

C.1.1 ISV Model Apple Stock MCMC Chains

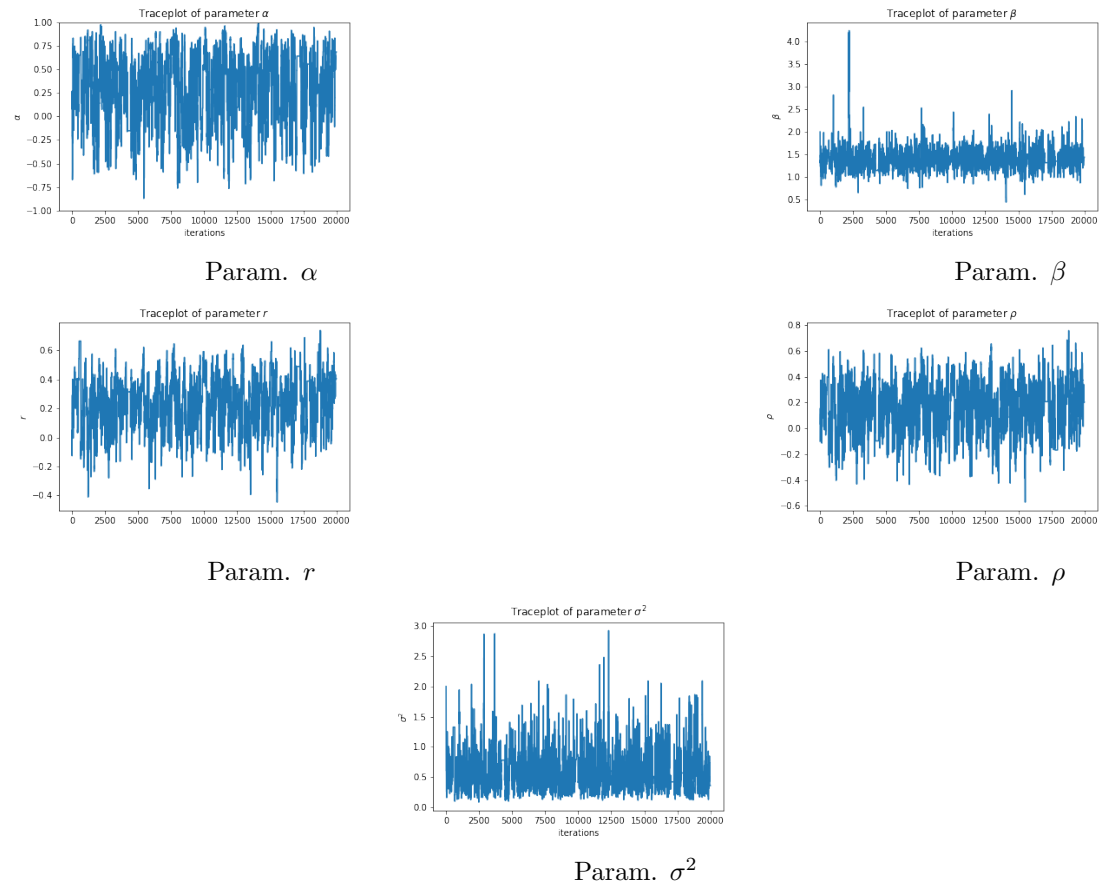
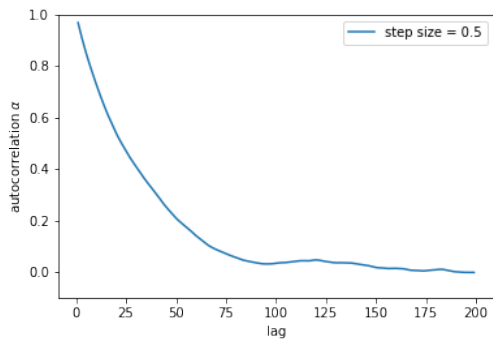
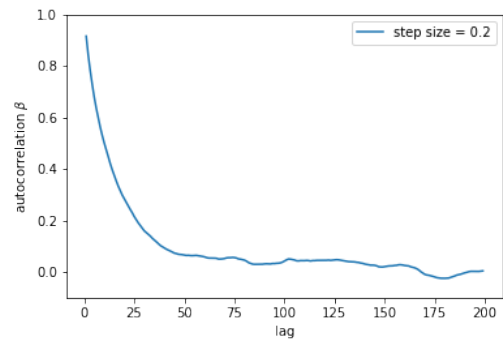


Figure C.1: Trace plots for ISV Model parameters θ (Apple MCMC chain results): $\alpha, \beta, r, \rho, \sigma^2$.

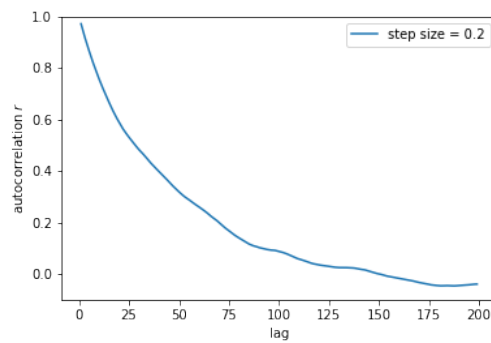
C.1.2 ISV Model Apple Stock MCMC Autocorrelations



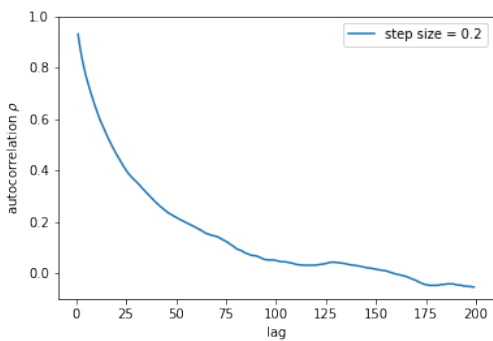
Parameter α



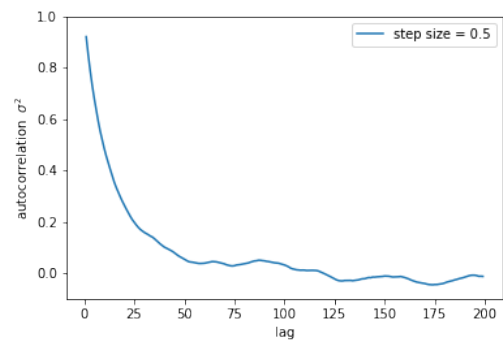
Parameter β



Parameter r



Parameter ρ



Parameter σ^2

Figure C.2: MCMC chain autocorrelations for ISV Model parameters θ (Apple Stock).

C.1.3 ISV Model Apple Stock θ Posterior Distribution

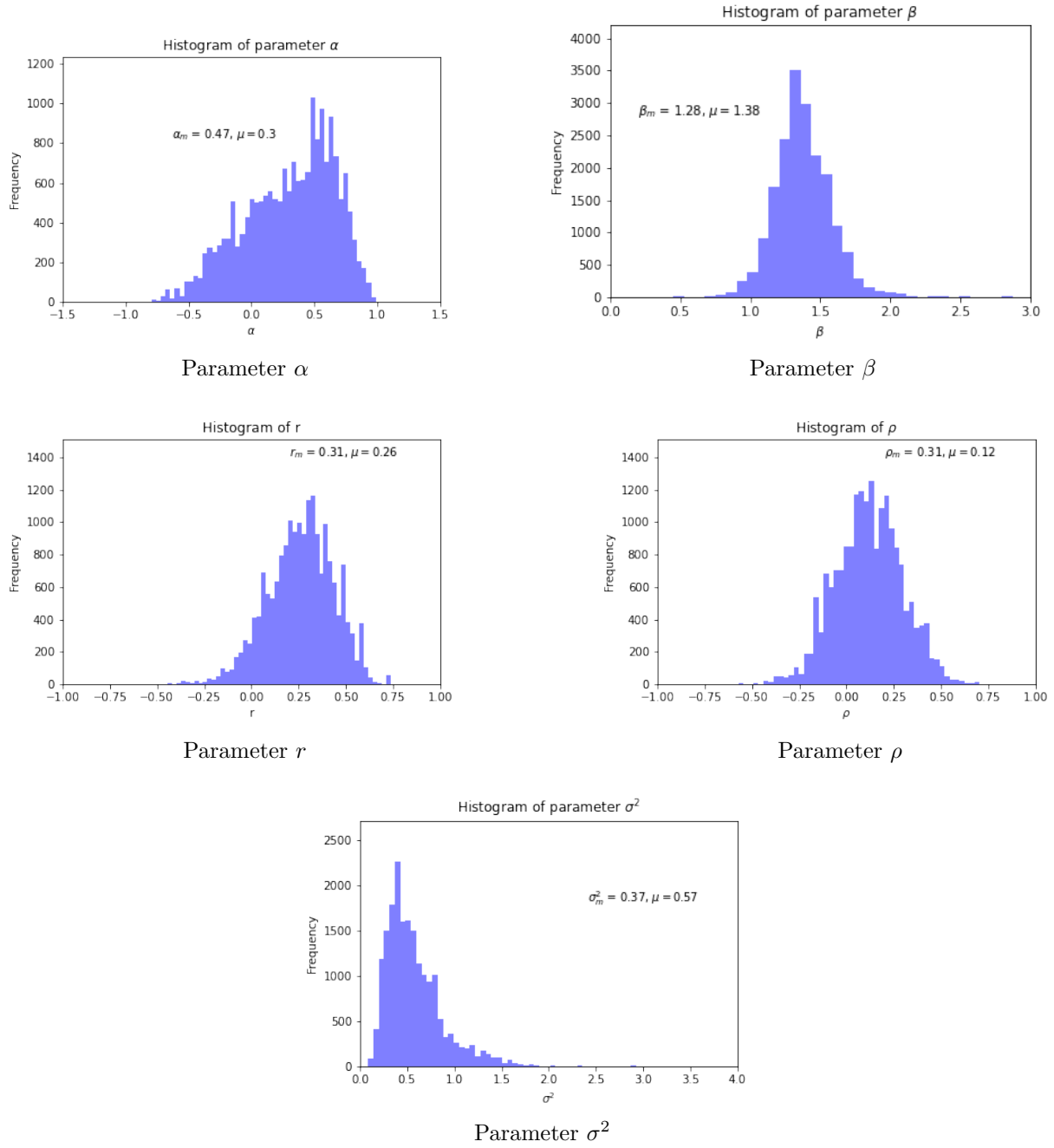
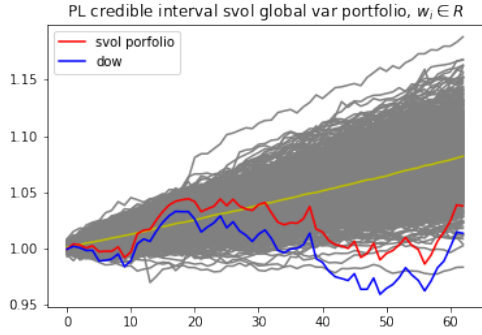
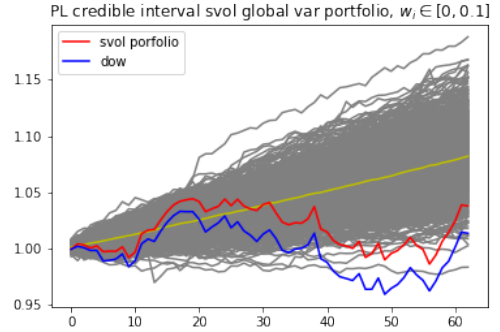


Figure C.3: Marginal posterior distributions of ISV Model parameters θ (Apple Stock). Parameters include α , β , r , ρ , and σ^2 .

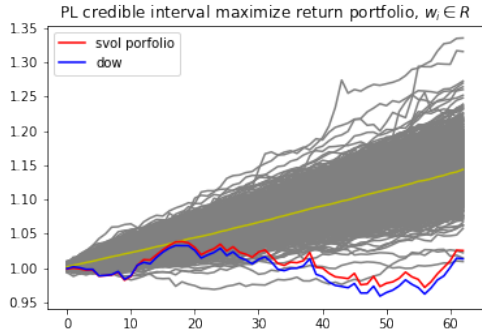
C.1.4 ISV Model P&L With Credible Intervals



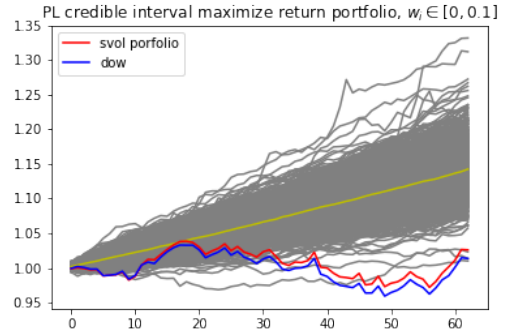
Minimize global variance without boundaries, (3.1a)



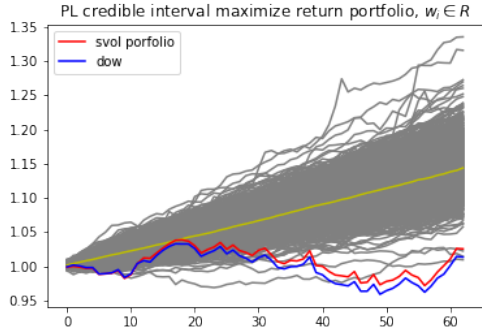
Minimize global variance with boundaries, (3.1b)



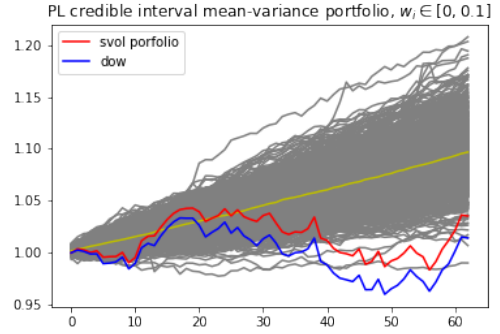
Maximize expected return without boundaries, (3.2a)



Maximize expected return with boundaries, (3.2b)



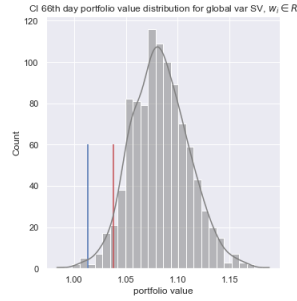
Minimize variance for a fixed return with boundaries, (3.3a)



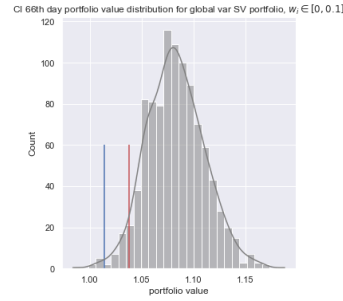
Minimize variance for a fixed return without boundaries, (3.3b)

Figure C.4: P&L clouds from 2000 simulated portfolio trajectories over $K = 66$ future steps. Red/blue lines show realized P&L of optimal allocation $\mathbf{w}_{T+K}$ and Dow Jones index, respectively.

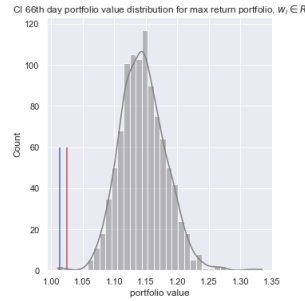
C.1.5 ISV Model \$1 P&L Distribution



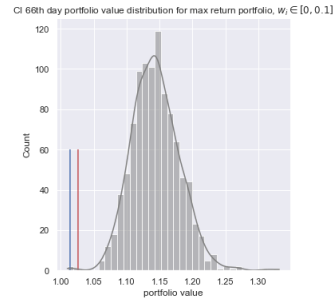
Minimize global variance without boundaries, cost function (3.1a)



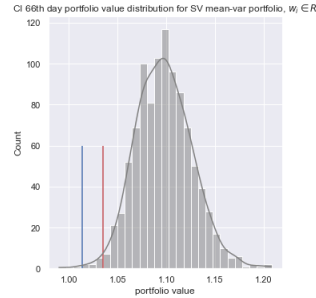
Minimize global variance with boundaries, cost function (3.1b)



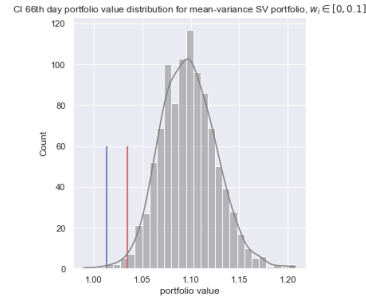
Maximize expected return without boundaries, cost function (3.2a)



Maximize expected return with boundaries, cost function (3.2b)



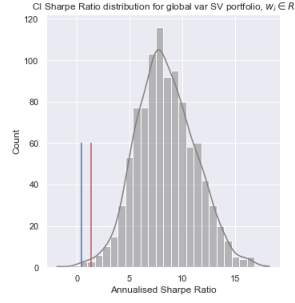
Minimize variance for a fixed return with boundaries, cost function (3.3a)



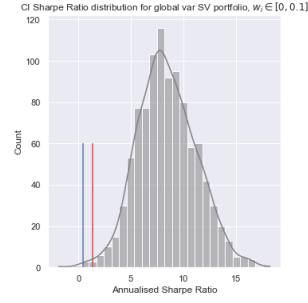
Minimize variance for a fixed return without boundaries, cost function (3.3b)

Figure C.5: Distribution of the 1000 simulated optimal portfolio values for day $T + K$, where future time steps $K = 66$ and the initial investment equals \$1 at $K = 0$. Red and blue lines represent the realized portfolio values for $\mathbf{w}_{T+K}$ and the Dow Jones, respectively. $\mathbf{w}_{T+K}$ optimizes various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$.

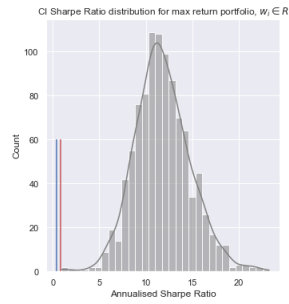
C.1.6 ISV Model Sharpe Ratio Distribution



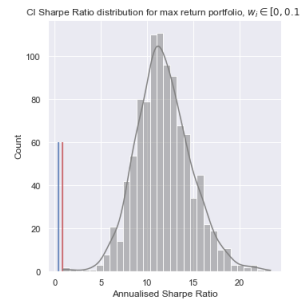
Global variance without boundaries
(3.1a)



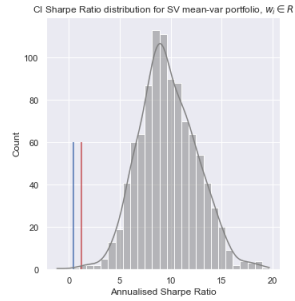
Global variance with boundaries
(3.1b)



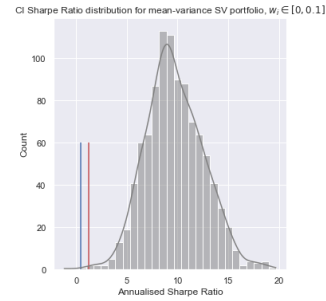
Max return without boundaries
(3.2a)



Max return with boundaries (3.2b)



Fixed return with boundaries (3.3a)



Fixed return without boundaries
(3.3b)

Figure C.6: Distribution of the 1000 simulated optimal portfolio Sharpe Ratios for day $T + K$, where $K = 66$ and the initial investment equals \$1 at $K = 0$. Red and blue lines represent the realized Sharpe Ratios for $\mathbf{w}_{T+K}$ and the Dow Jones, respectively. $\mathbf{w}_{T+K}$ optimizes the various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$.

C.2 MSV Model Results

C.2.1 MSV Model MCMC Chains

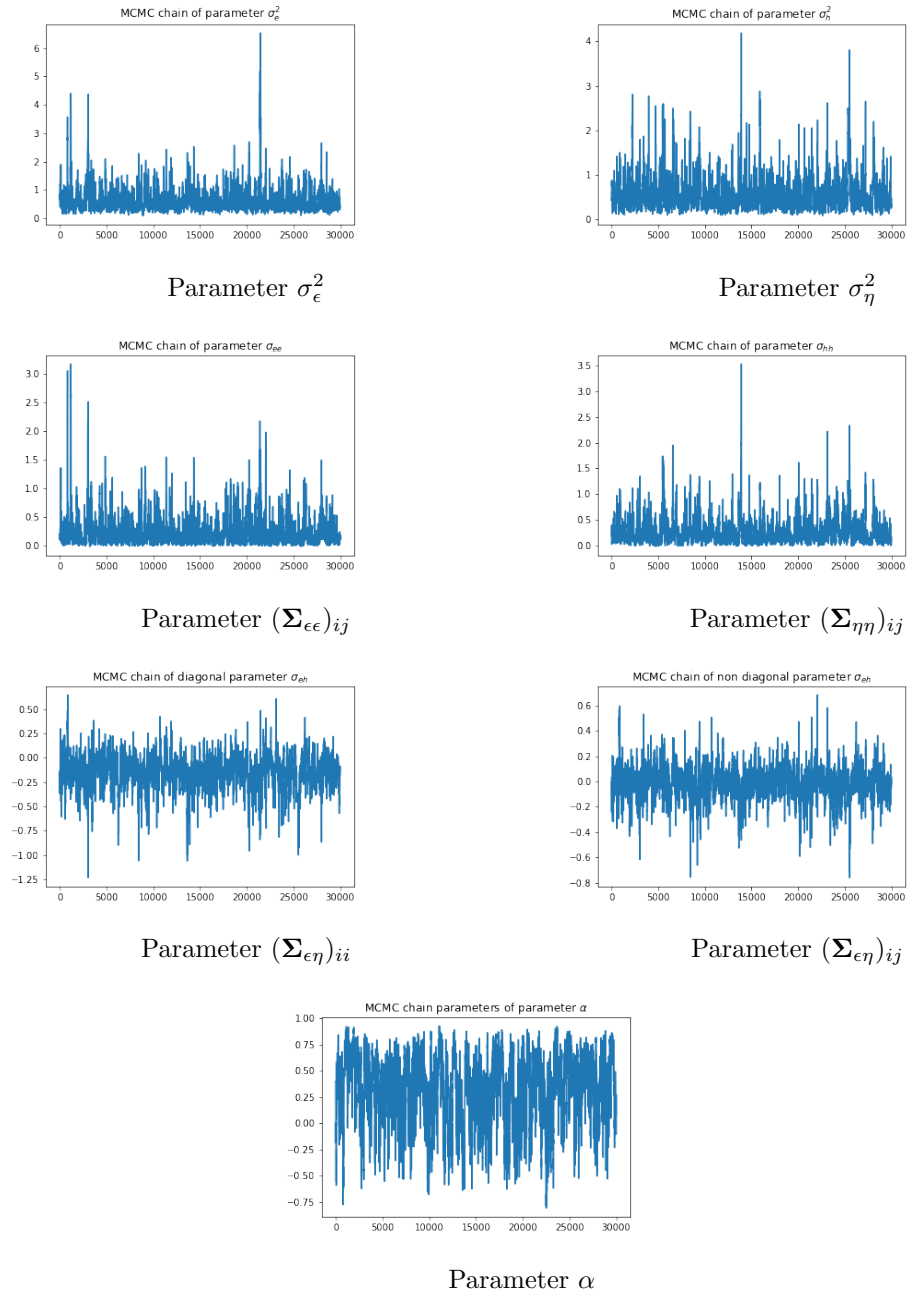


Figure C.7: MCMC chains for parameters θ in the MSV model.

C.2.2 MSV Model θ Posterior Distribution

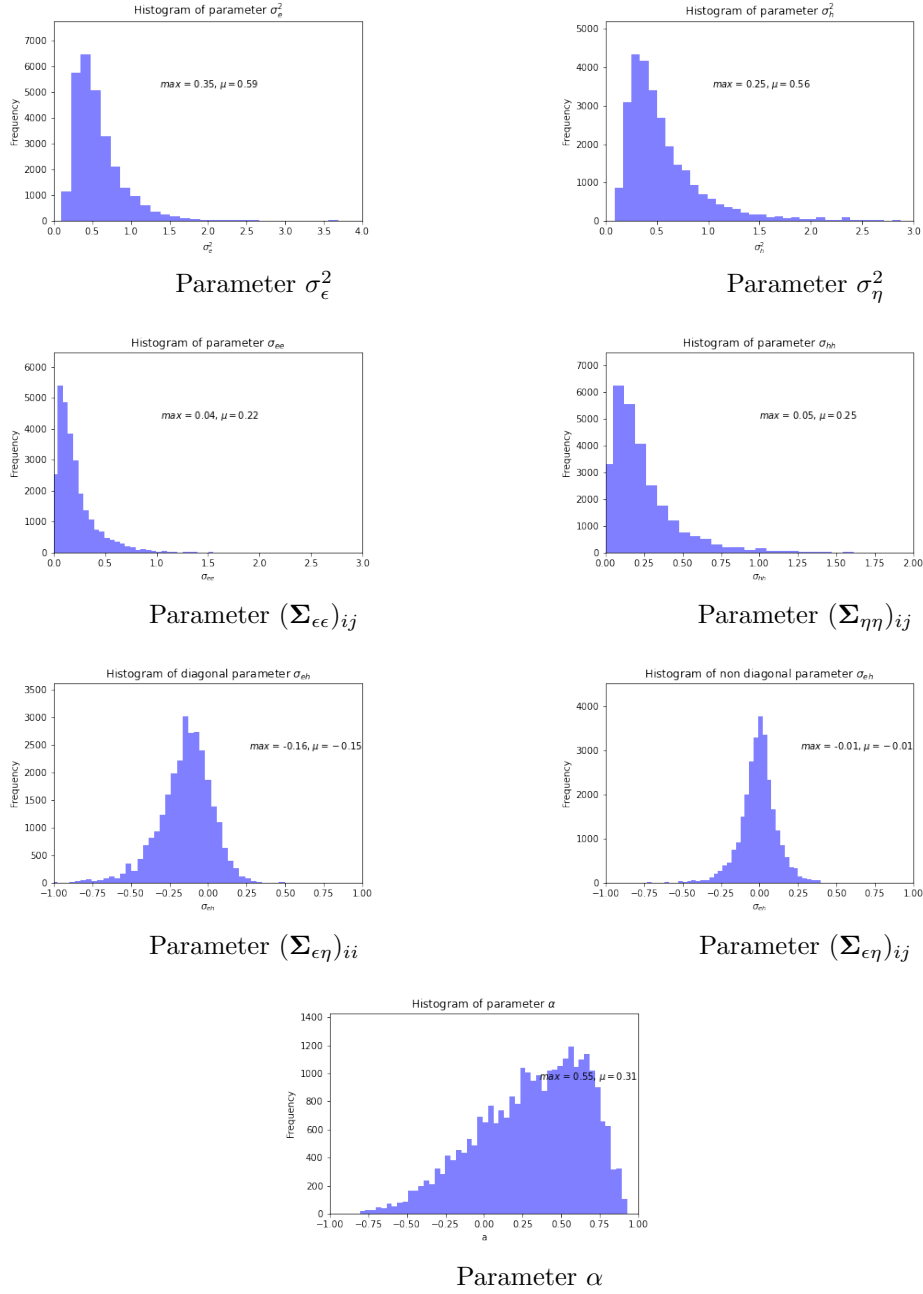
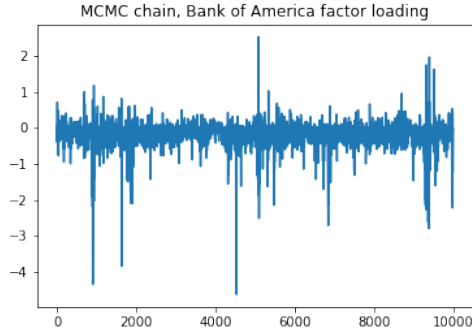


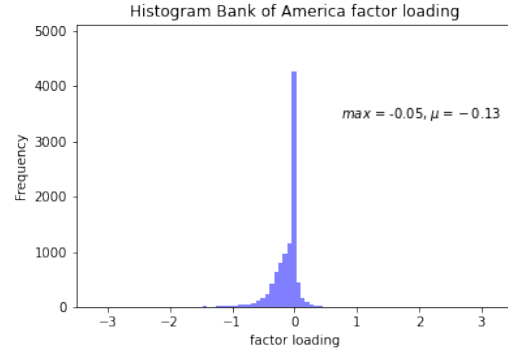
Figure C.8: Posterior distributions for parameters θ in the MSV model.

C.3 One Factor Model Results

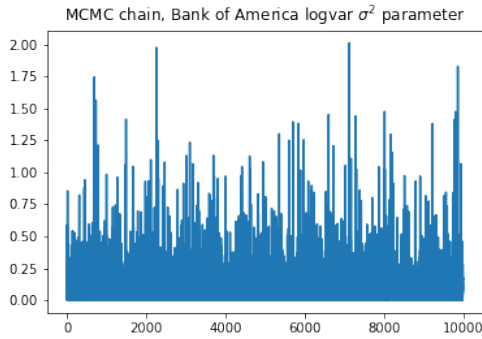
C.3.1 One Factor Model MCMC Chains & Posterior Distributions for Bank Of America Stock



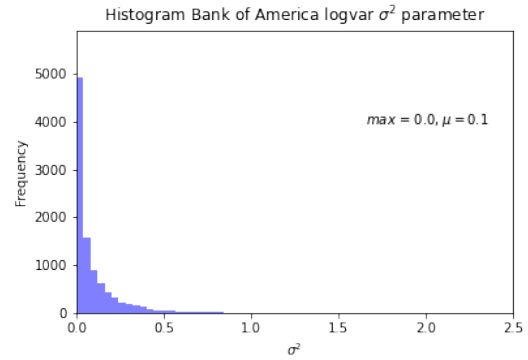
MCMC Bank of America factor load



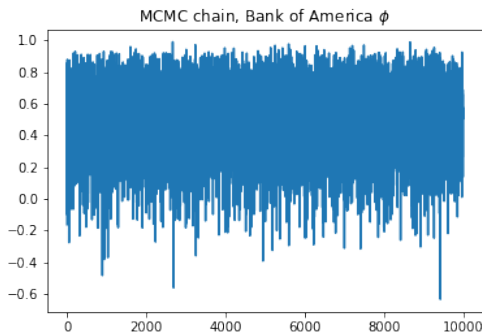
Marginal posterior of factor load



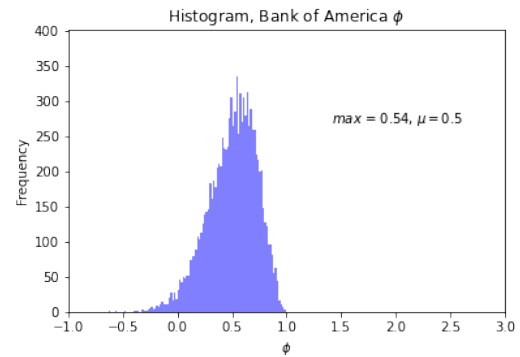
MCMC Bank of America log-variance σ^2



Marginal posterior of log-variance σ^2



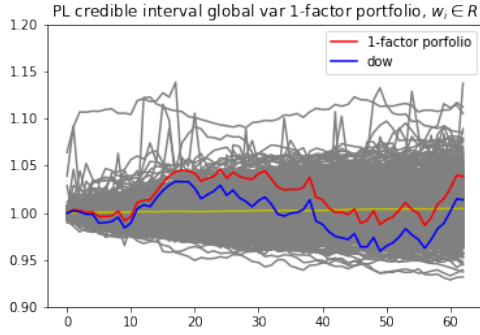
MCMC Bank of America ϕ



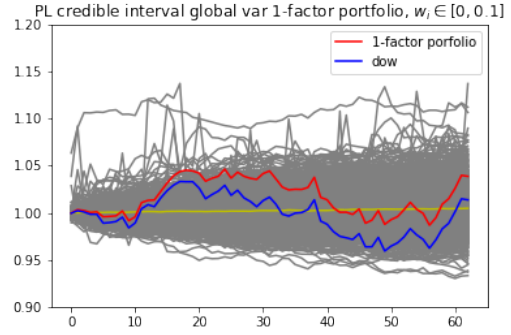
Marginal posterior of ϕ

Figure C.9: MCMC chains and marginal posterior distributions for Bank of America under the one-factor model. The figure presents the factor load, log-variance σ^2 , and parameter ϕ .

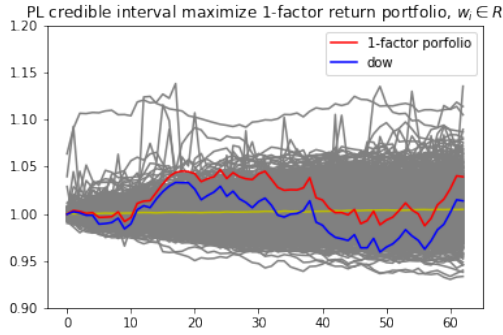
C.3.2 One Factor Model P&L With Credible Intervals



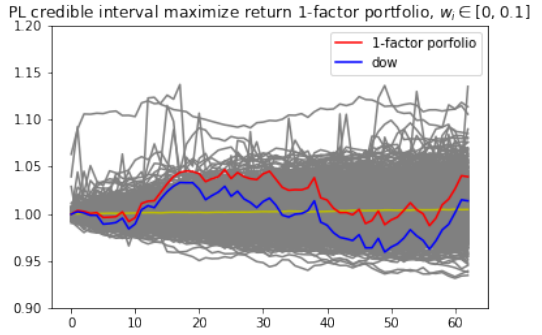
Global variance, no boundaries (3.1a)



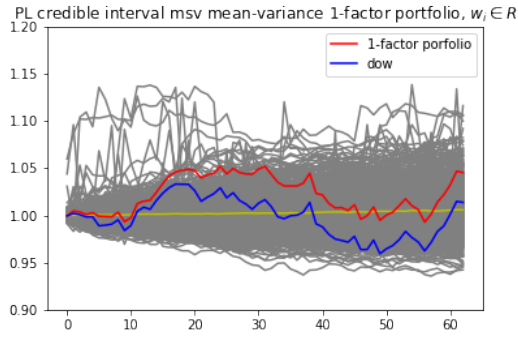
Global variance, boundaries (3.1b)



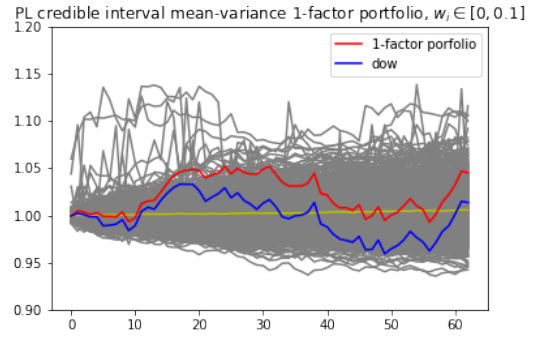
Maximize return, no boundaries (3.2a)



Maximize return, boundaries (3.2b)



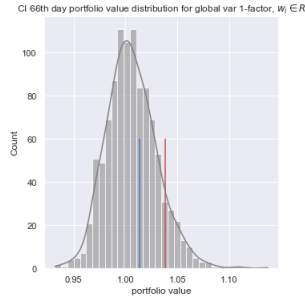
Variance-fixed return, boundaries (3.3a)



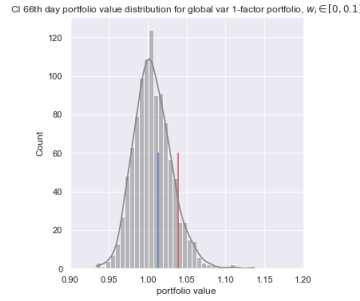
Variance-fixed return, no boundaries (3.3b)

Figure C.10: One Factor model P&L clouds created by 1000 simulated portfolio trajectories for $K = 66$ future time steps, based on the optimal portfolio $\mathbf{w}_{T+K}$. Red and blue lines represent the realized P&L paths of the optimal allocation $\mathbf{w}_{T+K}$ and the Dow Jones, respectively, when \$1 is invested at $K = 0$. Here, $\mathbf{w}_{T+K}$ minimizes various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$.

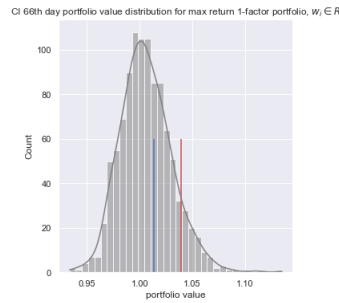
C.3.3 One Factor Model \$1 P&L Distribution



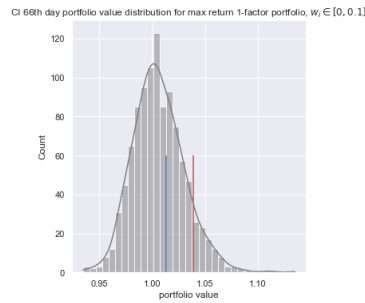
Minimize global variance without boundaries, cost function (3.1a)



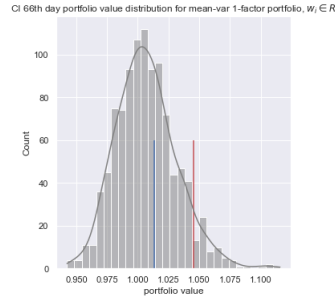
Minimize global variance with boundaries, cost function (3.1b)



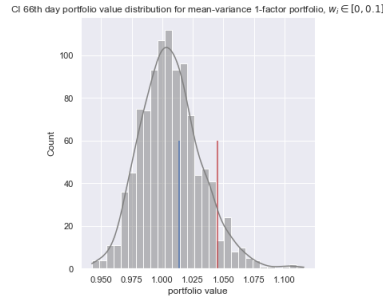
Maximize expected return without boundaries, cost function (3.2a)



Maximize expected return with boundaries, cost function (3.2b)



Minimize variance for a fixed return with boundaries, cost function (3.3a)



Minimize variance for a fixed return without boundaries, cost function (3.3b)

Figure C.11: Distribution of the 1000 simulated optimal portfolio values for day $T + K$, where future time steps $K = 66$ and the initial investment equals \$1 at $K = 0$. Red and blue lines represent the realized portfolio values for $\mathbf{w}_{T+K}$ and the Dow Jones, respectively. $\mathbf{w}_{T+K}$ optimizes various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$.

C.3.4 One Factor Model Sharpe Ratio Distribution

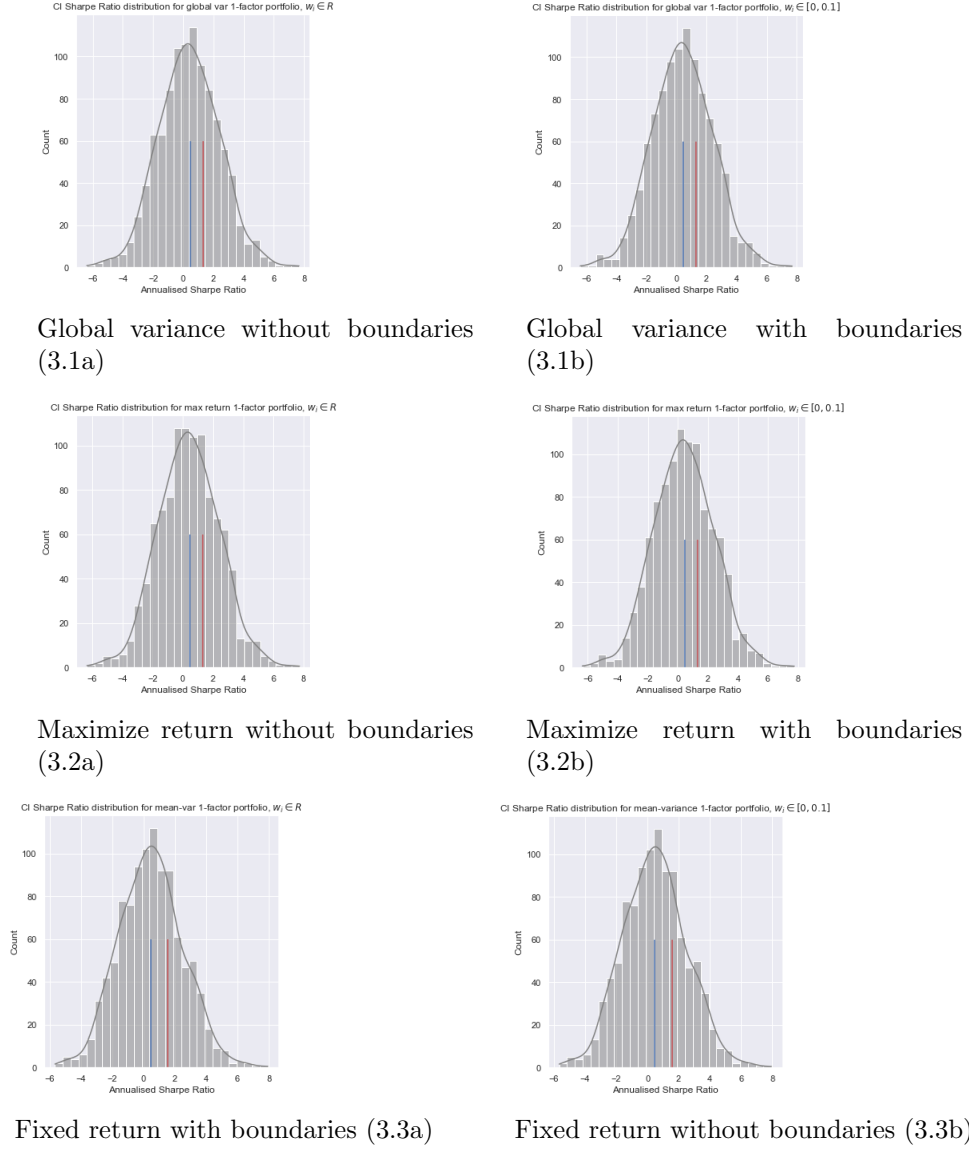
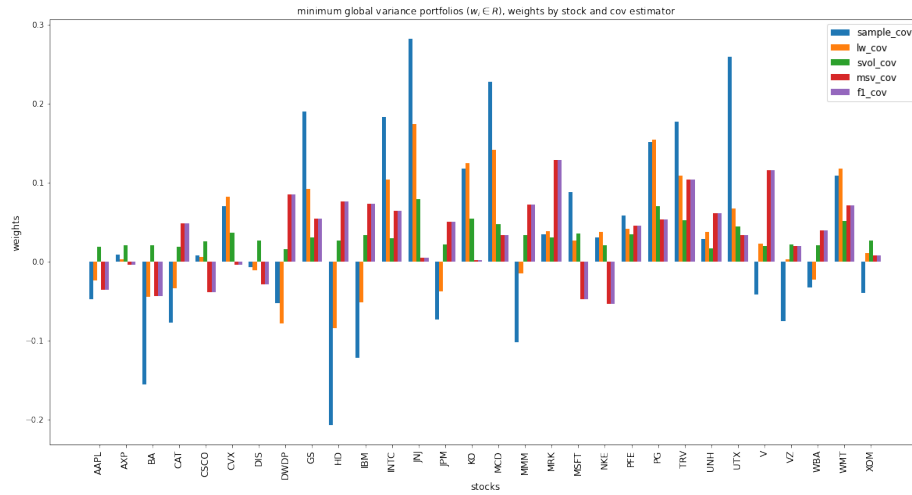
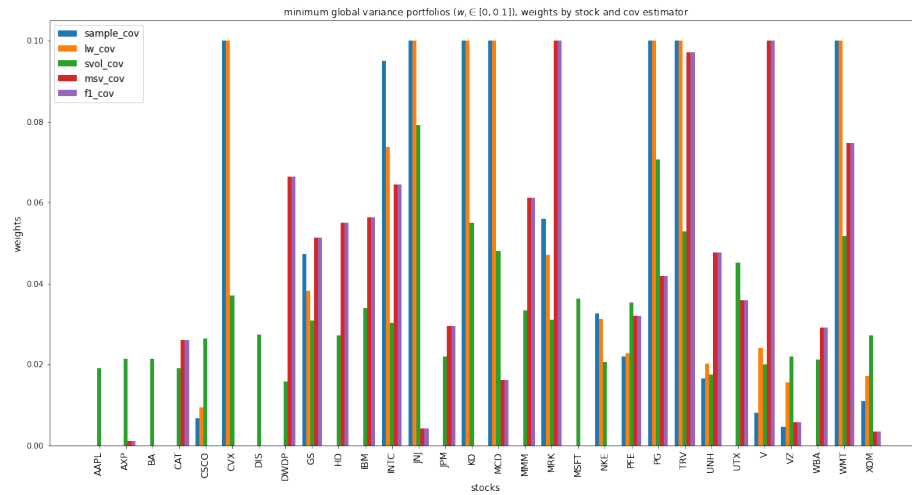


Figure C.12: Distribution of the 1000 simulated optimal portfolio Sharpe Ratios for day $T + K$, where $K = 66$ and the initial investment equals \$1 at $K = 0$. Red and blue lines represent the realized Sharpe Ratios for $\mathbf{w}_{T+K}$ and the Dow Jones, respectively. $\mathbf{w}_{T+K}$ optimizes the various cost functions for the estimated $\hat{\Sigma}_{T+K}$ and $\hat{\mu}_{T+K}$.

C.3.5 Optimal Allocations Per Cost Function

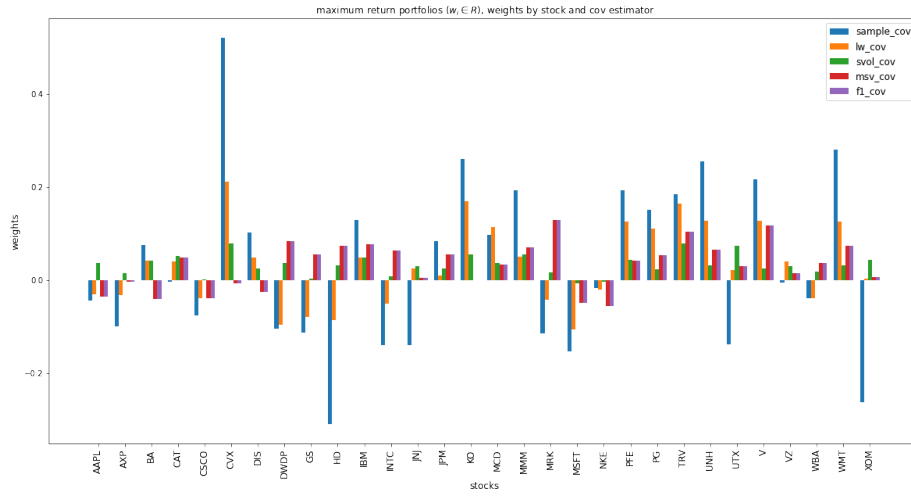


Minimize global variance without boundaries, cost function (3.1a)

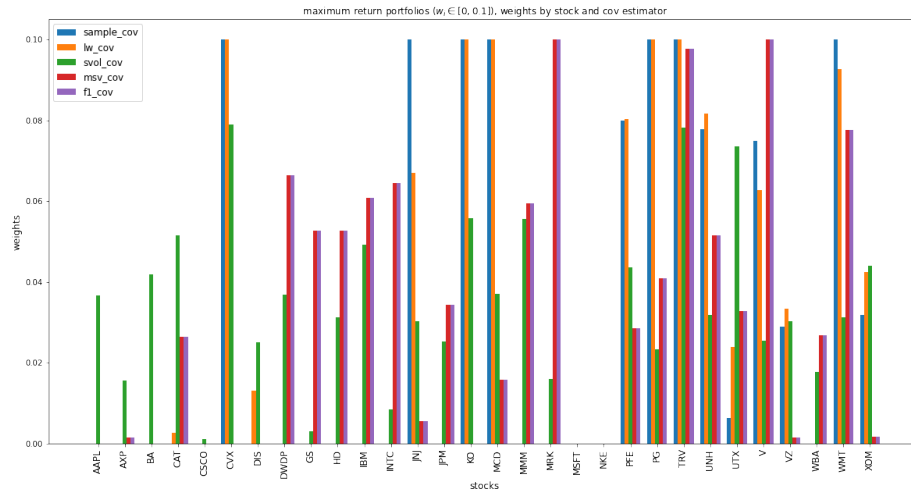


Minimize global variance with boundaries, cost function (3.1b)

Figure C.13: Optimal portfolio allocations for Dow Jones 30 stocks using global variance minimization.

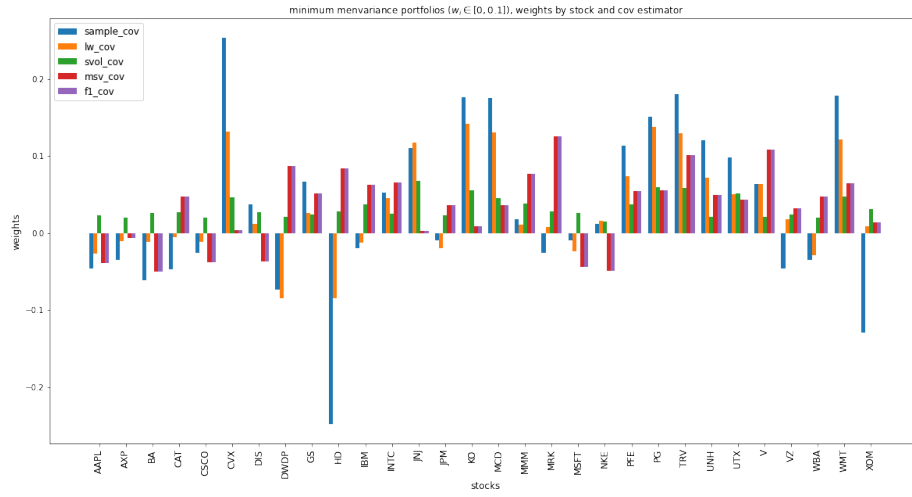


Maximize expected return without boundaries, cost function (3.2a)

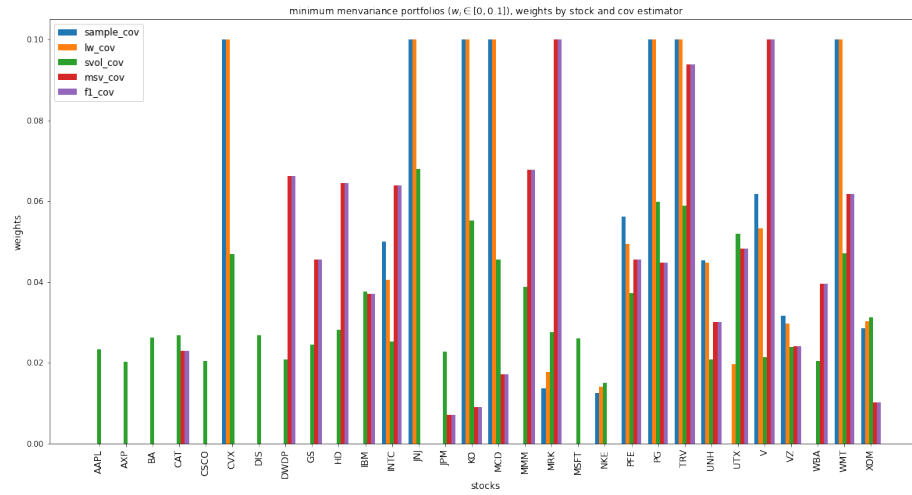


Maximize expected return with boundaries, cost function (3.2b)

Figure C.14: Optimal portfolio allocations for Dow Jones 30 stocks using return maximization.



Minimize variance for a fixed return without boundaries, cost function (3.3a)



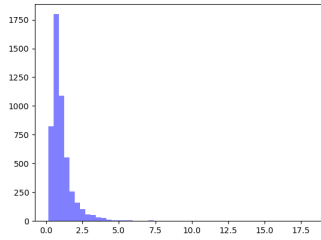
Minimize variance for a fixed return with boundaries, cost function (3.3b)

Figure C.15: Optimal portfolio allocations for Dow Jones 30 stocks using fixed-return variance minimization.

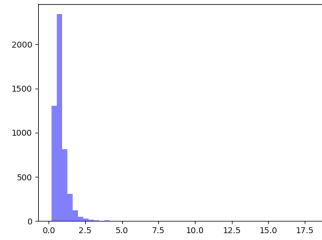
Appendix D

Chapter 4, Case Study 1

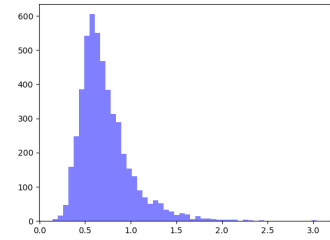
D.1 Estimating $p(\sigma_\epsilon^2 | \mathbf{y}_{1:T})$ Sequentially in Time



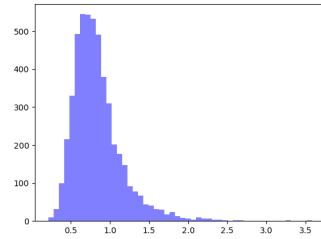
Histogram $\sigma_\epsilon^2, t = 1$



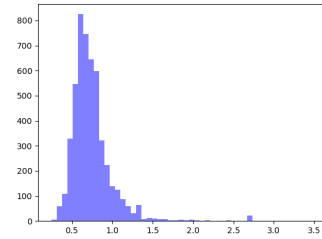
Histogram $\sigma_\epsilon^2, t = 2$



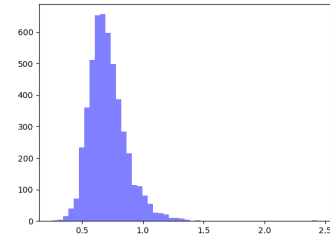
Histogram $\sigma_\epsilon^2, t = 3$



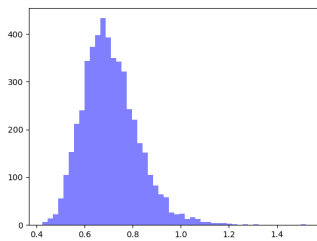
Histogram $\sigma_\epsilon^2, t = 4$



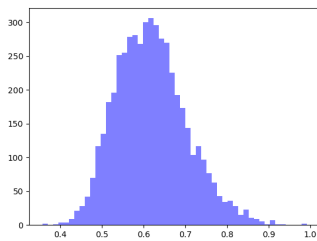
Histogram $\sigma_\epsilon^2, t = 5$



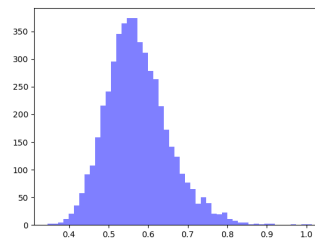
Histogram $\sigma_\epsilon^2, t = 6$



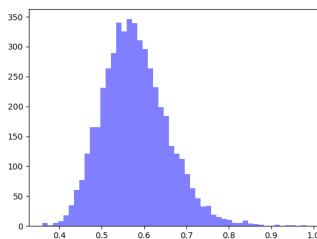
Histogram $\sigma_\epsilon^2, t = 7$



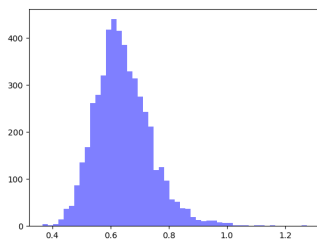
Histogram $\sigma_\epsilon^2, t = 8$



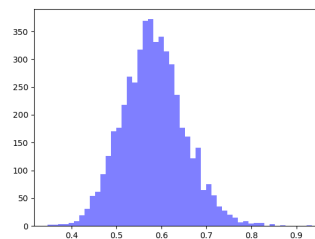
Histogram $\sigma_\epsilon^2, t = 9$



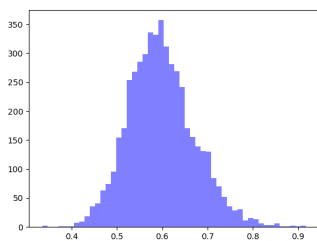
Histogram $\sigma_\epsilon^2, t = 10$



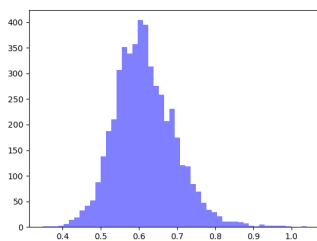
Histogram $\sigma_\epsilon^2, t = 11$



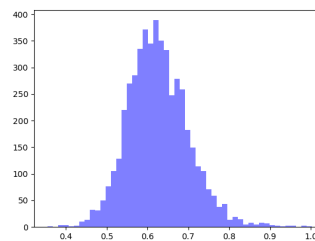
Histogram $\sigma_\epsilon^2, t = 12$



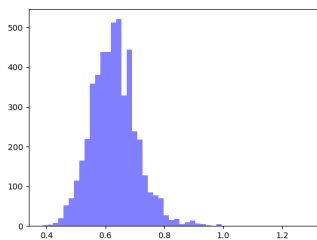
Histogram $\sigma_\epsilon^2, t = 13$



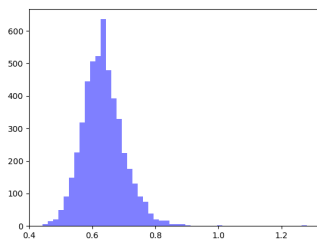
Histogram $\sigma_\epsilon^2, t = 14$



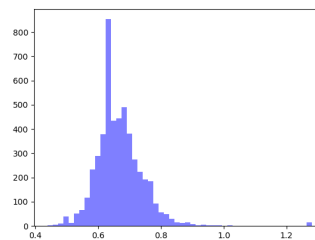
Histogram $\sigma_\epsilon^2, t = 15$



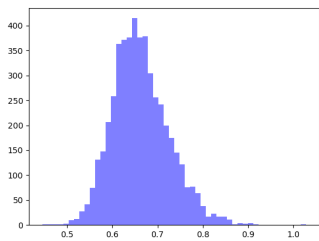
Histogram $\sigma_\epsilon^2, t = 16$



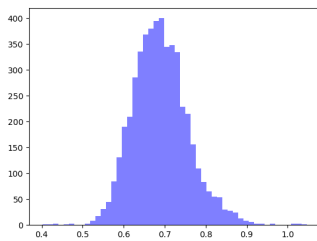
Histogram $\sigma_\epsilon^2, t = 17$



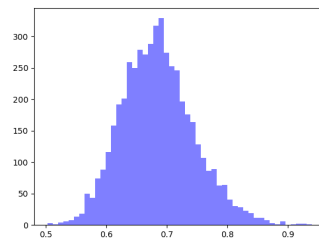
Histogram $\sigma_\epsilon^2, t = 18$



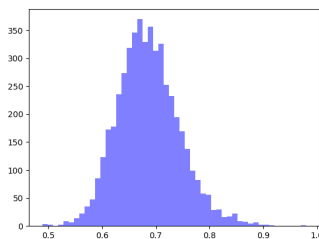
Histogram $\sigma_\epsilon^2, t = 19$



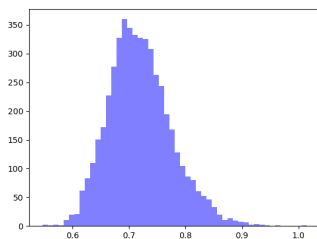
Histogram $\sigma_\epsilon^2, t = 20$



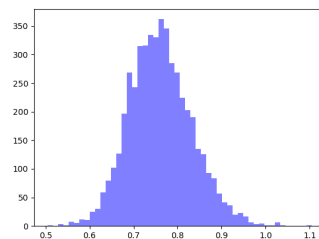
Histogram $\sigma_\epsilon^2, t = 21$



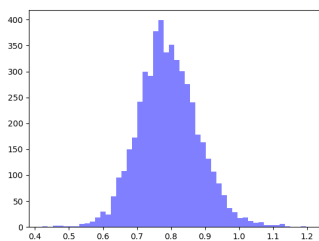
Histogram $\sigma_\epsilon^2, t = 22$



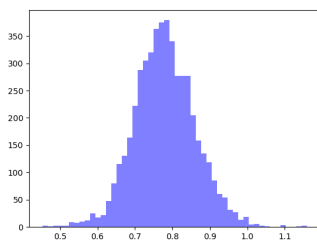
Histogram $\sigma_\epsilon^2, t = 23$



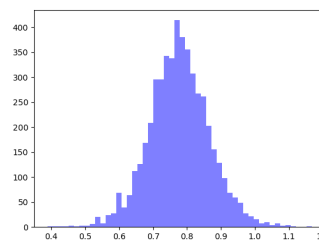
Histogram $\sigma_\epsilon^2, t = 24$



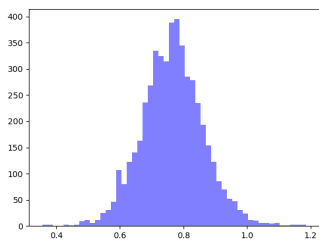
Histogram $\sigma_\epsilon^2, t = 25$



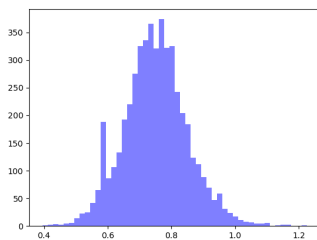
Histogram $\sigma_\epsilon^2, t = 26$



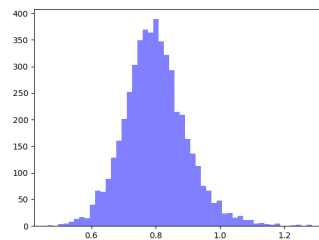
Histogram $\sigma_\epsilon^2, t = 27$



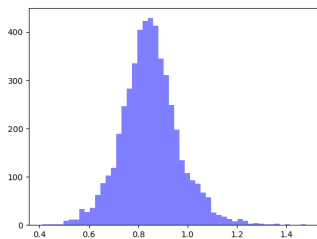
Histogram $\sigma_\epsilon^2, t = 28$



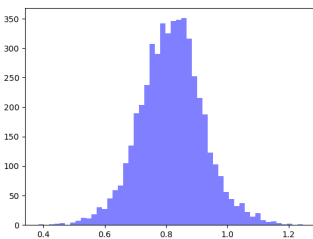
Histogram $\sigma_\epsilon^2, t = 29$



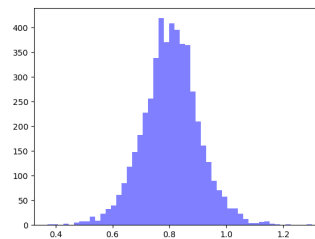
Histogram $\sigma_\epsilon^2, t = 30$



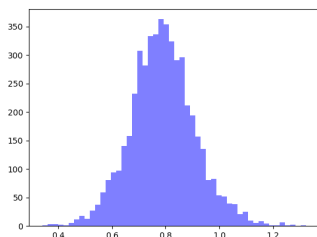
Histogram $\sigma_\epsilon^2, t = 31$



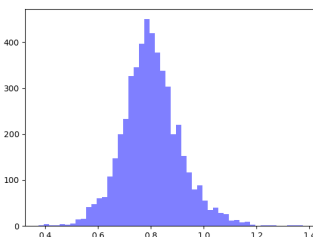
Histogram $\sigma_\epsilon^2, t = 32$



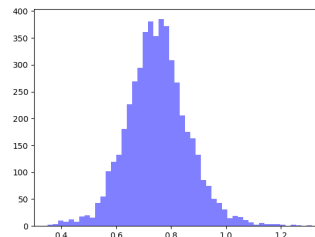
Histogram $\sigma_\epsilon^2, t = 33$



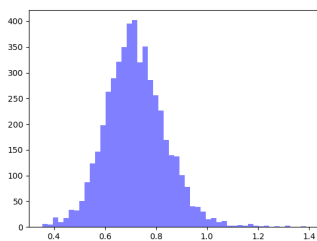
Histogram $\sigma_\epsilon^2, t = 34$



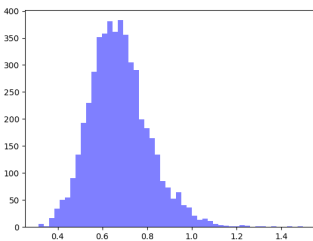
Histogram $\sigma_\epsilon^2, t = 35$



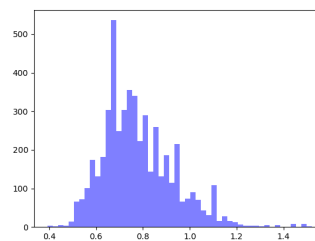
Histogram $\sigma_\epsilon^2, t = 36$



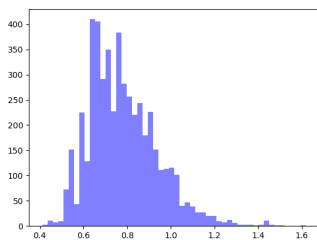
Histogram $\sigma_\epsilon^2, t = 37$



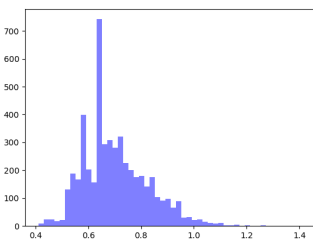
Histogram $\sigma_\epsilon^2, t = 38$



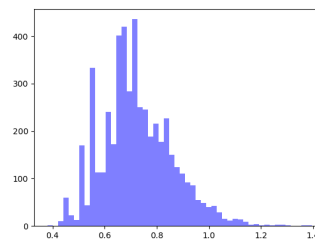
Histogram $\sigma_\epsilon^2, t = 39$



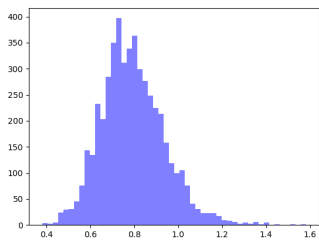
Histogram $\sigma_\epsilon^2, t = 40$



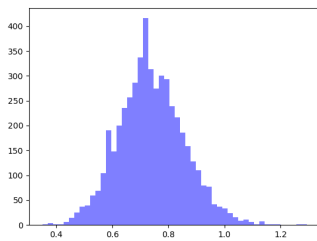
Histogram $\sigma_\epsilon^2, t = 41$



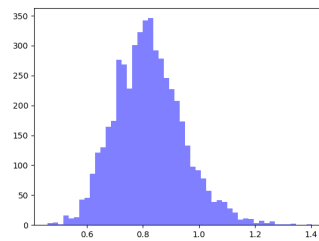
Histogram $\sigma_\epsilon^2, t = 42$



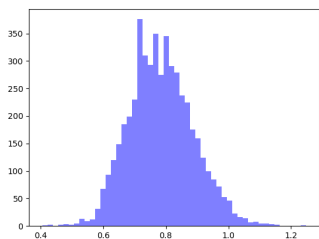
Histogram $\sigma_\epsilon^2, t = 43$



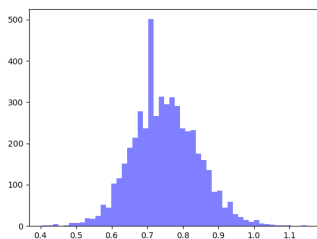
Histogram $\sigma_\epsilon^2, t = 44$



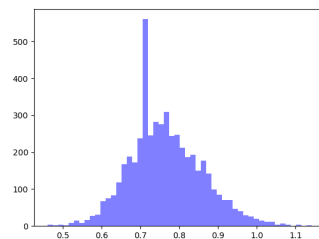
Histogram $\sigma_\epsilon^2, t = 45$



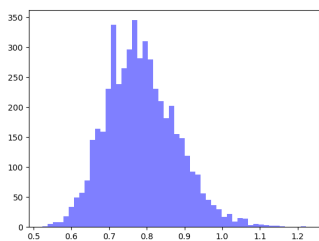
Histogram $\sigma_\epsilon^2, t = 46$



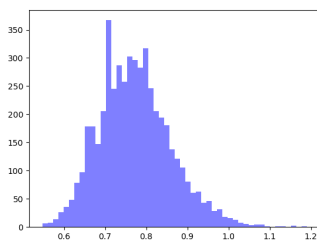
Histogram $\sigma_\epsilon^2, t = 47$



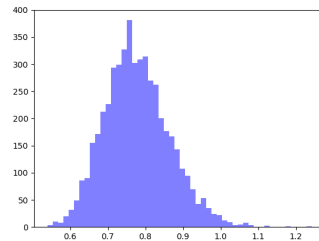
Histogram $\sigma_\epsilon^2, t = 48$



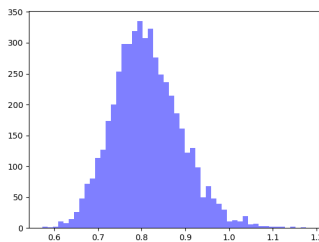
Histogram $\sigma_\epsilon^2, t = 49$



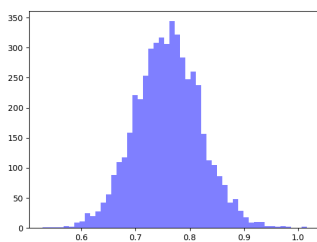
Histogram $\sigma_\epsilon^2, t = 50$



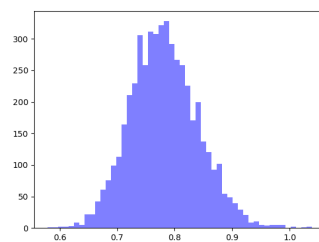
Histogram $\sigma_\epsilon^2, t = 51$



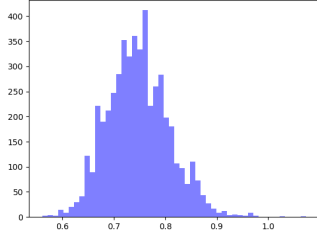
Histogram $\sigma_\epsilon^2, t = 52$



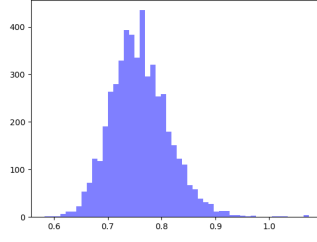
Histogram $\sigma_\epsilon^2, t = 53$



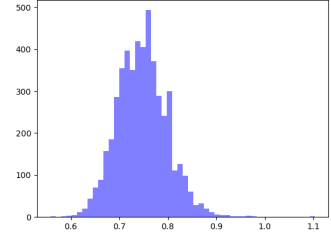
Histogram $\sigma_\epsilon^2, t = 54$



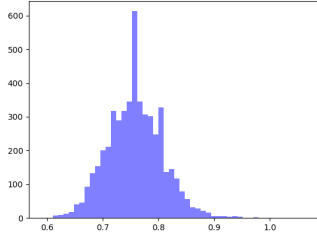
Histogram $\sigma_\epsilon^2, t = 55$



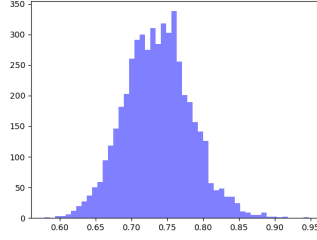
Histogram $\sigma_\epsilon^2, t = 56$



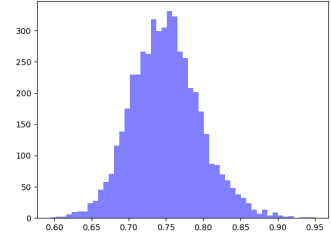
Histogram $\sigma_\epsilon^2, t = 57$



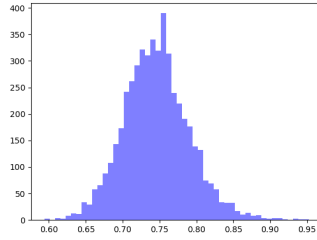
Histogram $\sigma_\epsilon^2, t = 58$



Histogram $\sigma_\epsilon^2, t = 59$



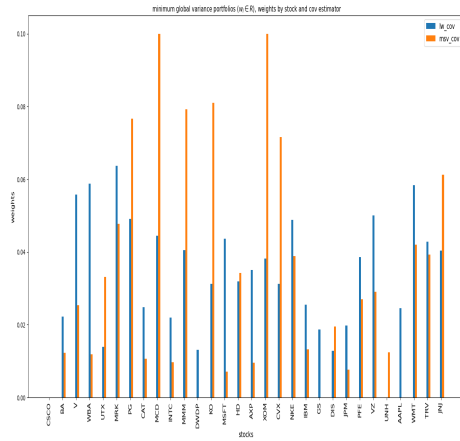
Histogram $\sigma_\epsilon^2, t = 60$



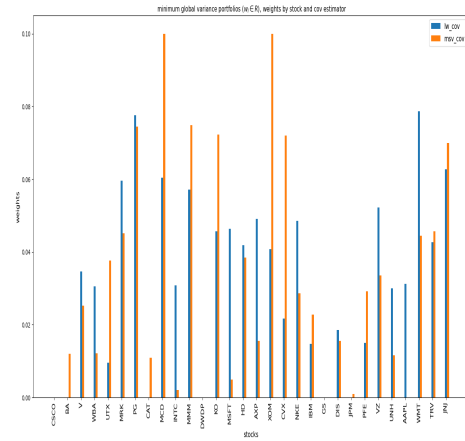
Histogram $\sigma_\epsilon^2, t = 61$

Figure D.21: Sequential Posterior Estimation of σ_ϵ^2 for 61 observations in Time

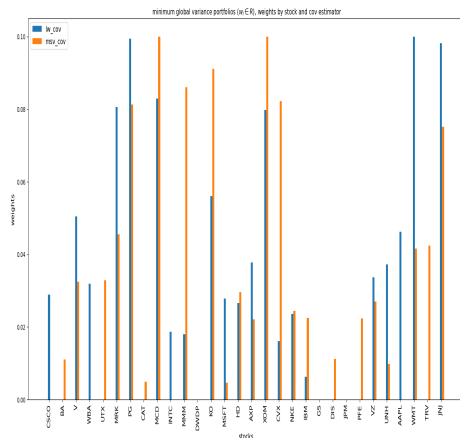
D.2 Optimal Allocations



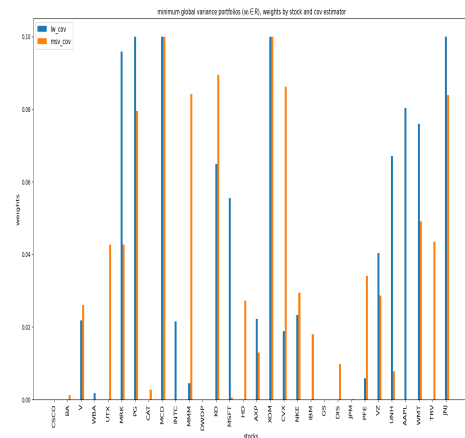
Optimal weight with 15 training data points



Optimal weight with 30 training data points

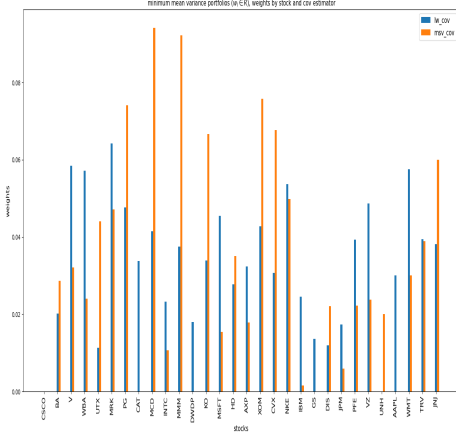


Optimal weight with 45 training data points

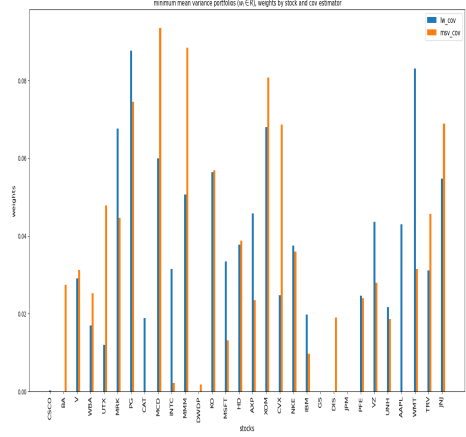


Optimal weight with 61 training data points

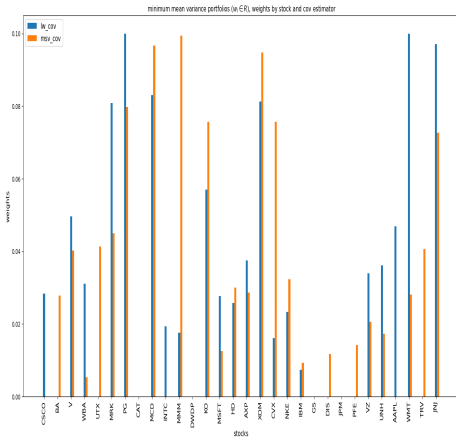
Figure D.22: Portfolio optimal weight allocation using the global minimum variance cost function.



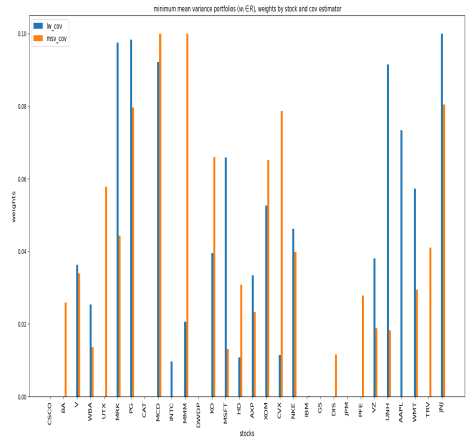
Optimal weight with 15 training data points



Optimal weight with 30 training data points



Optimal weight with 45 training data points



Optimal weight with 61 training data points

Figure D.23: Portfolio optimal weight allocation using the global minimum variance cost function for a fixed expected return.

D.3 P&L Results

D.3.1 Realized P&L Tables

	Mean R	Std. Dev R	Sharpe Ratio	L2	Ratio
Dow	31.12	7.51	4.14	-	-
lw_global	52.15	6.19	8.42	0.20	42.10
msv_global	42.74	5.22	8.19	0.25	32.76
lw_meanvar	52.17	6.26	8.34	0.20	41.70
msv_meanvar	47.89	5.27	9.09	0.24	37.88

Table D.1: First 15 data points training set, fifteenth out-of-sample optimal portfolio realized performance.

	Mean R	Std. Dev R	Sharpe Ratio	L2	Ratio
Dow	9.74	11.77	0.83	-	-
sample_global	2.89	9.58	0.30	0.27	1.11
lw_global	3.21	9.97	0.32	0.22	1.45
msv_global	8.04	10.50	0.77	0.25	3.08
sample_meanvar	2.89	9.58	0.30	0.27	1.11
lw_meanvar	3.59	9.76	0.37	0.23	1.61
msv_meanvar	10.65	10.77	0.99	0.24	4.12

Table D.2: First 30 data points training set, fifteenth out-of-sample optimal portfolio realized performance.

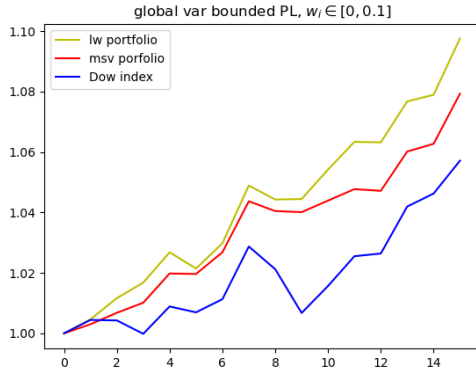
	Mean R	Std. Dev R	Sharpe Ratio	L2	Ratio
Dow	-5.15	10.56	-0.49	-	-
sample_global	-1.31	7.99	-0.16	0.28	-0.57
lw_global	0.70	8.16	0.09	0.26	0.35
msv_global	-4.53	9.24	-0.49	0.26	-1.88
sample_meanvar	-1.31	7.99	-0.16	0.28	-0.57
lw_meanvar	0.61	8.14	0.07	0.26	0.27
msv_meanvar	-0.42	9.41	-0.04	0.25	-0.16

Table D.3: First 45 data points training set, fifteenth out-of-sample optimal portfolio realized performance.

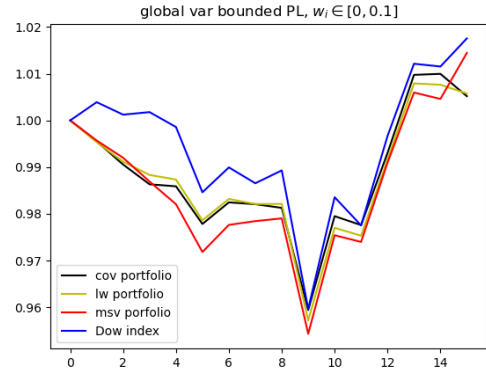
	Mean R	Std. Dev R	Sharpe Ratio	L2	Ratio
Dow	27.03	10.42	2.59	-	-
sample_global	42.58	7.85	5.42	0.29	18.69
lw_global	42.24	8.03	5.26	0.28	18.79
msv_global	32.82	9.26	3.54	0.26	13.62
sample_meanvar	42.58	7.85	5.42	0.29	18.69
lw_meanvar	43.84	8.47	5.17	0.27	19.15
msv_meanvar	35.97	9.66	3.72	0.25	14.88

Table D.4: First 61 data points training set, fifteenth out-of-sample optimal portfolio realized performance.

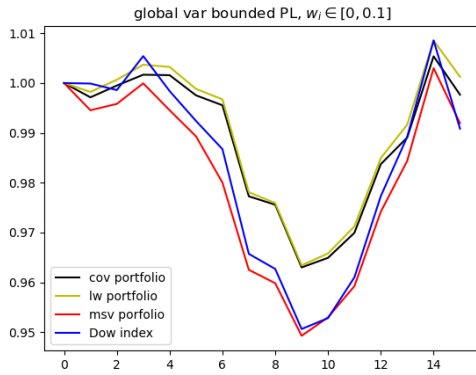
D.3.2 Out Of Sample Realized Trajectory, When \$1 Is Invested



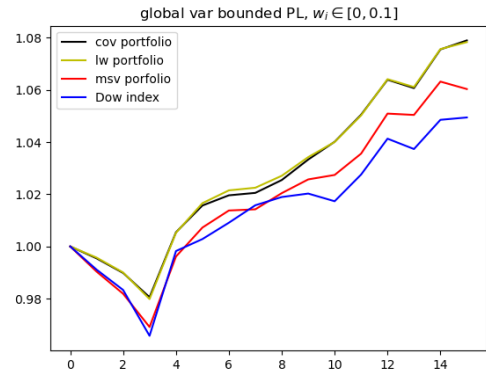
trained first 15 data points, out of sample realized P&L for next 15 data points



trained first 30 data points, out of sample realized P&L for next 15 data points

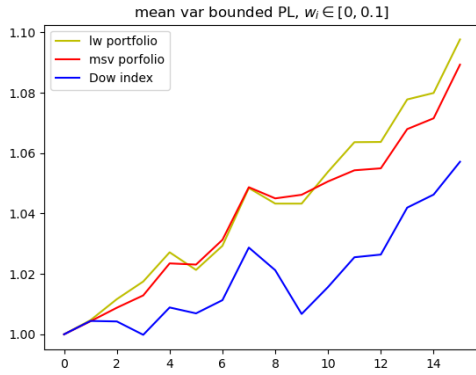


trained first 45 data points, out of sample realized P&L for next 15 data points

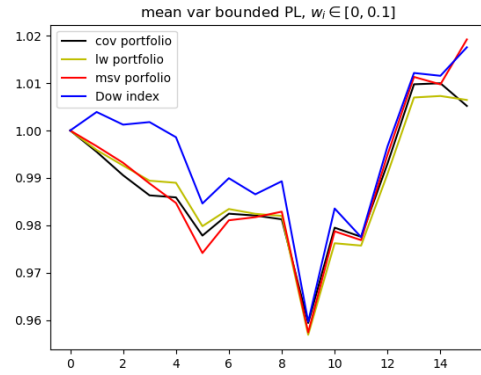


trained first 61 data points, out of sample realized P&L for next 15 data points

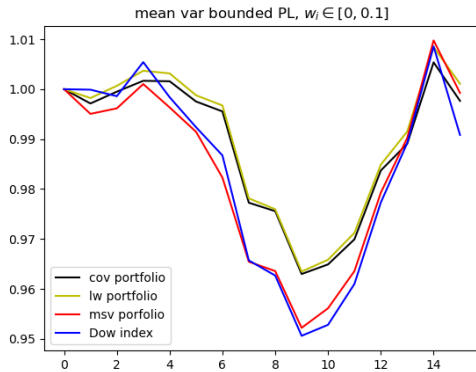
Figure D.24: Portfolio realized value trajectory for the various data sets when \$1 invested in the optimal portfolio given by global minimum variance cost function.



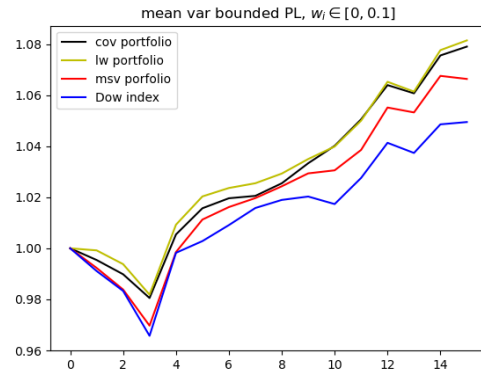
trained first 15 data points, out of sample realized P&L for next 15 data points



trained first 30 data points, out of sample realized P&L for next 15 data points



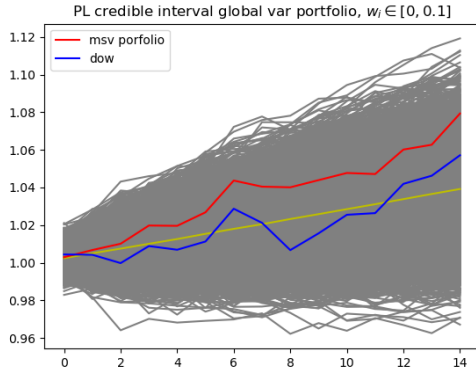
trained first 45 data points, out of sample realized P&L for next 15 data points



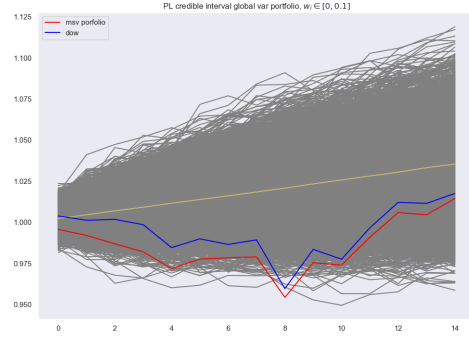
trained first 61 data points, out of sample realized P&L for next 15 data points

Figure D.25: Portfolio realized value trajectory for the various data sets when \$1 invested in the optimal portfolio given by global minimum variance cost function for a fixed level of expected return.

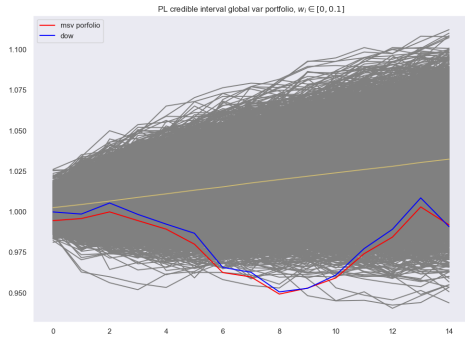
D.3.3 Credible Intervals



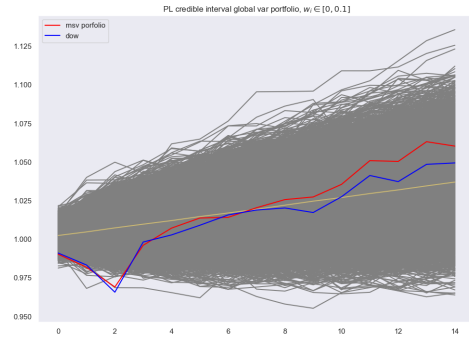
trained first 15 data points, out of sample realized P&L for next 15 data points



trained first 30 data points, out of sample realized P&L for next 15 data points

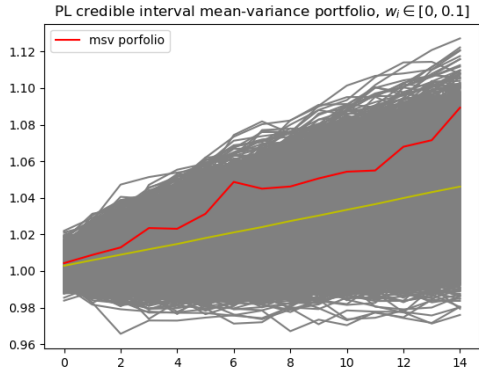


trained first 45 data points, out of sample realized P&L for next 15 data points

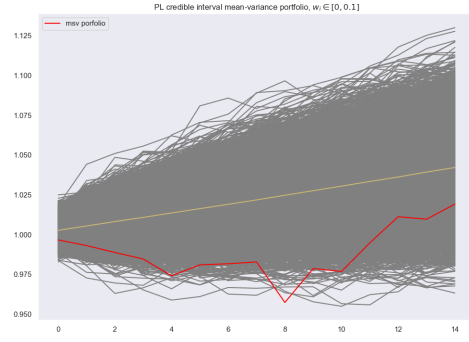


trained first 61 data points, out of sample realized P&L for next 15 data points

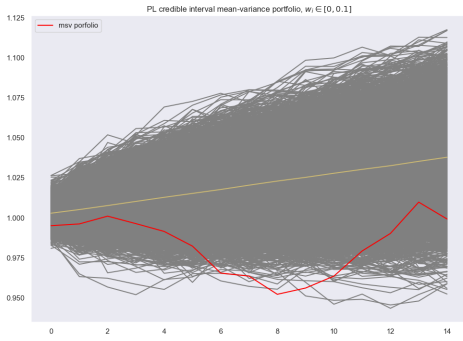
Figure D.26: MSV PL clouds created by 10000 simulated portfolio trajectories for $K = 15$ future time steps, based on optimal portfolio w_{T+K} realized value trajectory for the various data sets when \$1 invested in the optimal portfolio given by global minimum variance cost function. Red and blue lines represent the realized PL path of the optimal allocation w_{T+K} and the Dow Jones respectively.



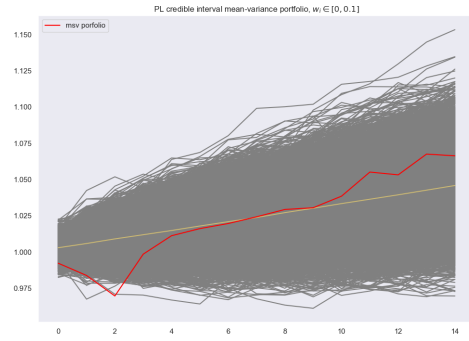
trained first 15 data points, out of sample realized P&L for next 15 data points



trained first 30 data points, out of sample realized P&L for next 15 data points



trained first 45 data points, out of sample realized P&L for next 15 data points

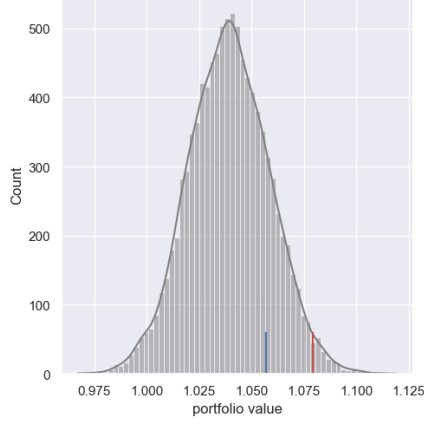


trained first 61 data points, out of sample realized P&L for next 15 data points

Figure D.27: MSV PL clouds created by 10000 simulated portfolio trajectories for $K = 15$ future time steps, based on optimal portfolio w_{T+K} realized value trajectory for the various data sets when \$1 invested in the optimal portfolio given by global minimum variance cost function for a given expected return. Red and blue lines represent the realized PL path of the optimal allocation w_{T+K} and the Dow Jones respectively.

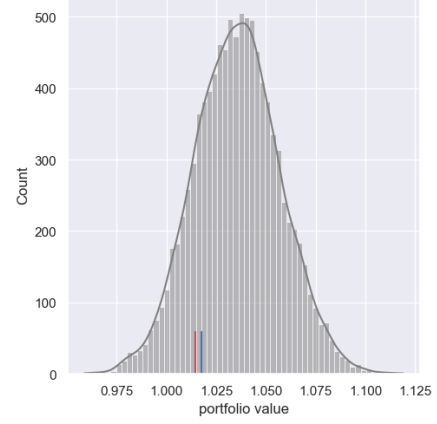
D.3.4 Optimal Portfolio Terminal Value Distribution

CI 66th day portfolio value distribution for mean-variance MSV portfolio, $w_t \in [0, 0.1]$



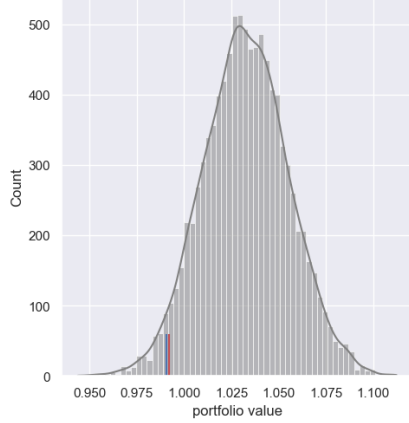
trained first 15 data points, out of sample expected P&L for next 15 data points

CI 66th day portfolio value distribution for mean-variance MSV portfolio, $w_t \in [0, 0.1]$



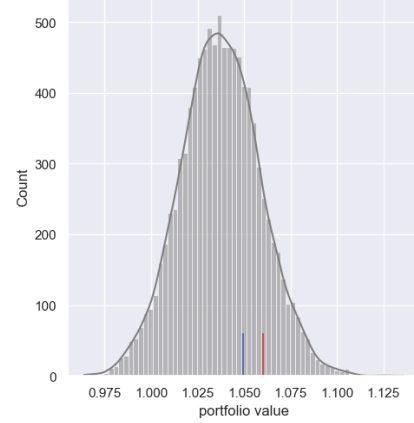
trained first 30 data points, out of sample expected P&L for next 15 data points

CI 66th day portfolio value distribution for mean-variance MSV portfolio, $w_t \in [0, 0.1]$



trained first 45 data points, out of sample expected P&L for next 15 data points

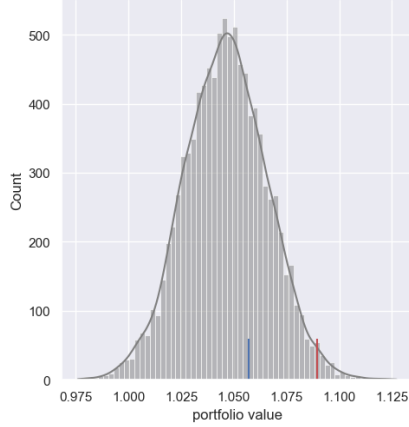
CI 66th day portfolio value distribution for mean-variance MSV portfolio, $w_t \in [0, 0.1]$



trained first 61 data points, out of sample expected P&L for next 15 data points

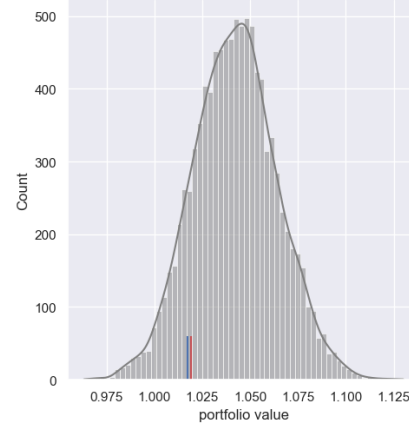
Figure D.28: Distribution of MSV terminal values generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function. The red and blue lines indicate the realized profit and loss of the optimal allocation w_{T+K} and the Dow Jones, respectively, for an initial investment of \$1.

CI 66th day portfolio value distribution for mean-variance MSV portfolio, $w_i \in [0, 0.1]$



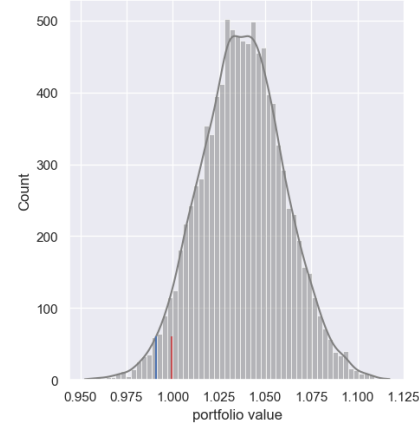
trained first 15 data points, out of sample expected P&L for next 15 data points

CI 66th day portfolio value distribution for mean-variance MSV portfolio, $w_i \in [0, 0.1]$



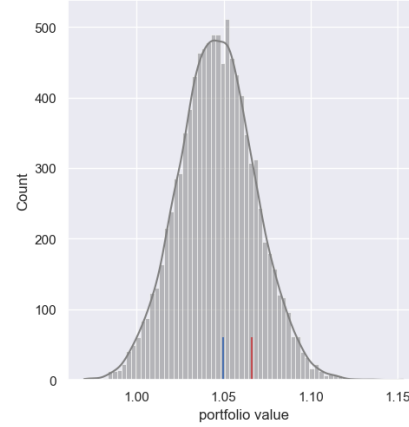
trained first 30 data points, out of sample expected P&L for next 15 data points

CI 66th day portfolio value distribution for mean-variance MSV portfolio, $w_i \in [0, 0.1]$



trained first 45 data points, out of sample expected P&L for next 15 data points

CI 66th day portfolio value distribution for mean-variance MSV portfolio, $w_i \in [0, 0.1]$

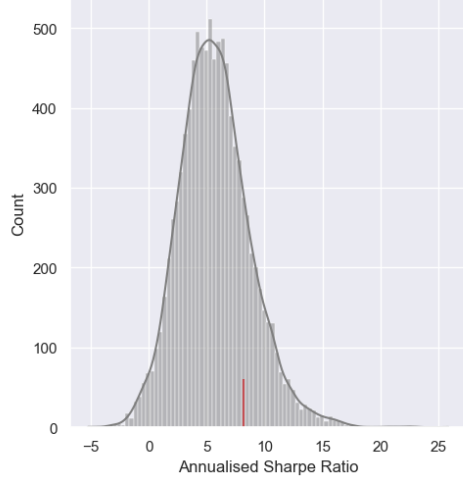


trained first 61 data points, out of sample expected P&L for next 15 data points

Figure D.29: Distribution of MSV terminal values generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function for a given level of expected return. The red and blue lines indicate the realized profit and loss of the optimal allocation w_{T+K} and the Dow Jones, respectively, for an initial investment of \$1.

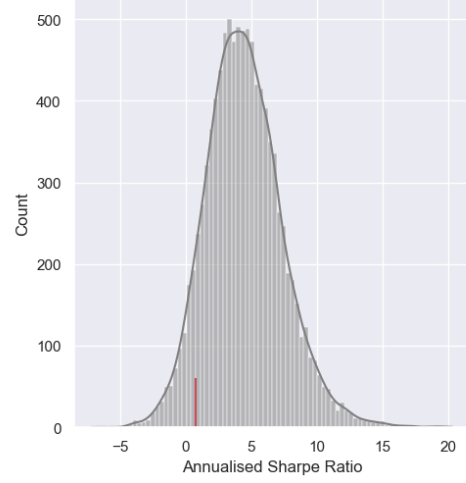
D.3.5 Optimal Portfolio Sharpe Ratio Distribution

CI Sharpe Ratio distribution for mean-variance MSV portfolio, $w_l \in [0, 0.1]$



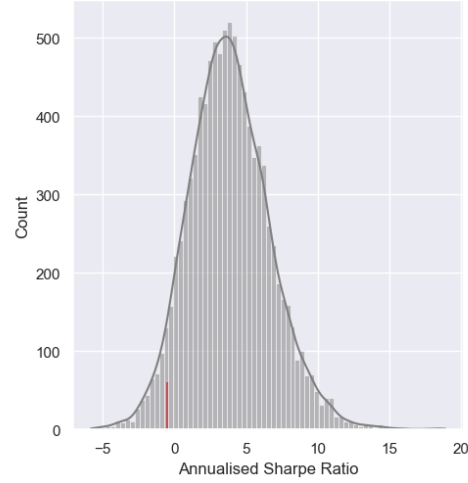
trained first 15 data points, out of sample expected SR for next 15 data points

CI Sharpe Ratio distribution for mean-variance MSV portfolio, $w_l \in [0, 0.1]$



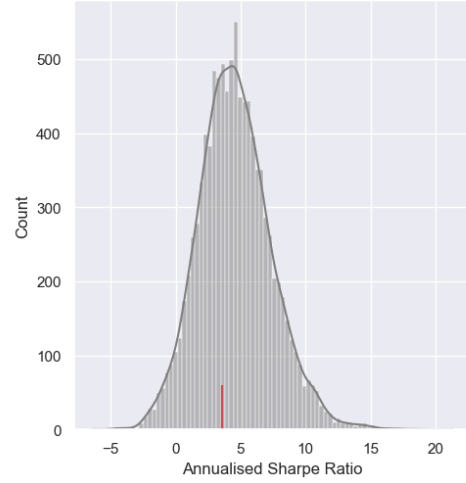
trained first 30 data points, out of sample expected SR for next 15 data points

CI Sharpe Ratio distribution for mean-variance MSV portfolio, $w_l \in [0, 0.1]$



trained first 45 data points, out of sample expected SR for next 15 data points

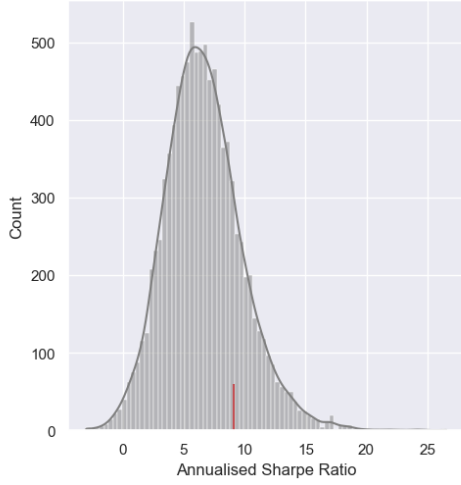
CI Sharpe Ratio distribution for mean-variance MSV portfolio, $w_l \in [0, 0.1]$



trained first 61 data points, out of sample expected SR for next 15 data points

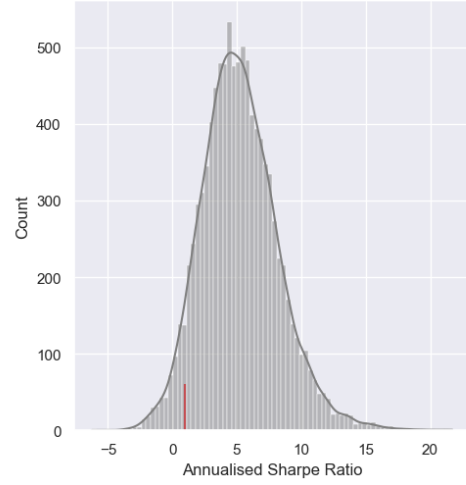
Figure D.30: Distribution of MSV SR generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function. The red line indicates the realized SR of the optimal allocation w_{T+K} for an initial investment of \$1.

CI Sharpe Ratio distribution for mean-variance MSV portfolio, $w_I \in [0, 0.1]$



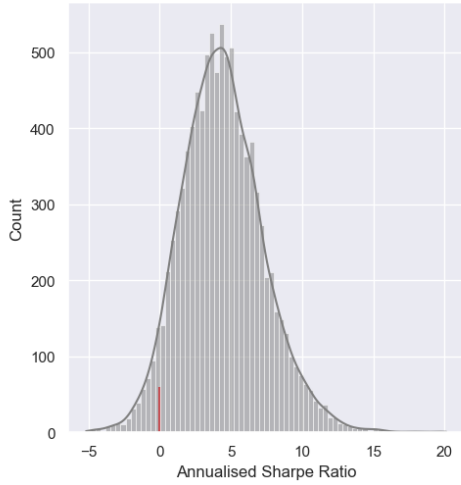
trained first 15 data points, out of sample expected SR for next 15 data points

CI Sharpe Ratio distribution for mean-variance MSV portfolio, $w_I \in [0, 0.1]$



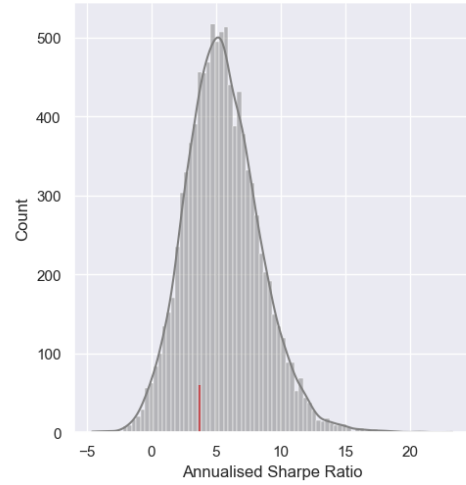
trained first 30 data points, out of sample expected SR for next 15 data points

CI Sharpe Ratio distribution for mean-variance MSV portfolio, $w_I \in [0, 0.1]$



trained first 45 data points, out of sample expected SR for next 15 data points

CI Sharpe Ratio distribution for mean-variance MSV portfolio, $w_I \in [0, 0.1]$



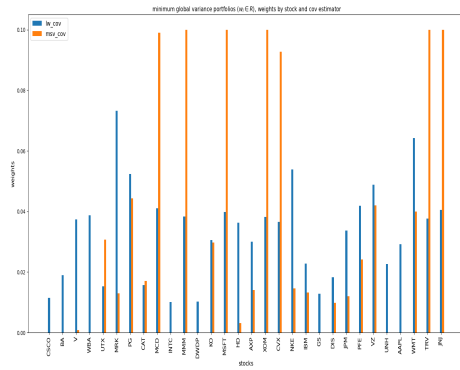
trained first 61 data points, out of sample expected SR for next 15 data points

Figure D.31: Distribution of MSV SR generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function for a given level of expected return. The red line indicates the realized SR of the optimal allocation w_{T+K} for an initial investment of \$1.

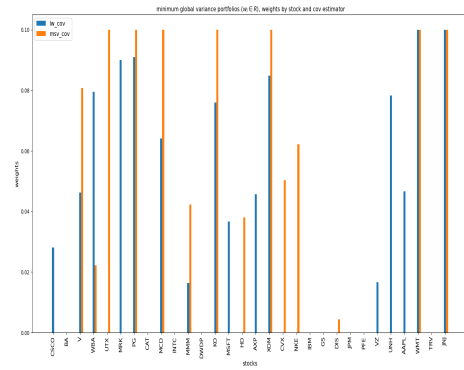
Appendix E

Chapter 4 Case Study 2

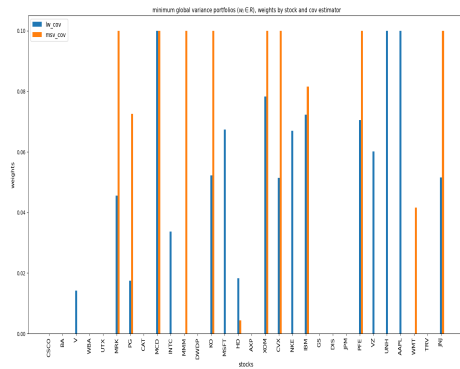
E.1 Optimal Allocations



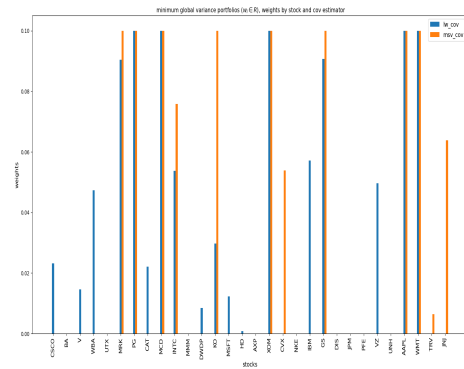
Optimal allocation first quarter



Optimal allocation second quarter

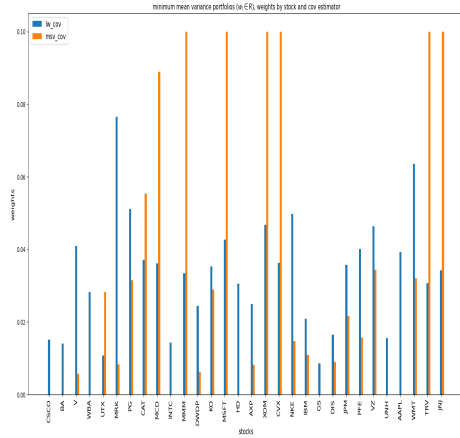


Optimal allocation third quarter

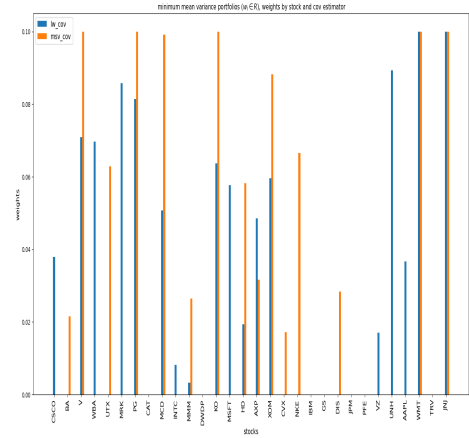


Optimal allocation fourth quarter

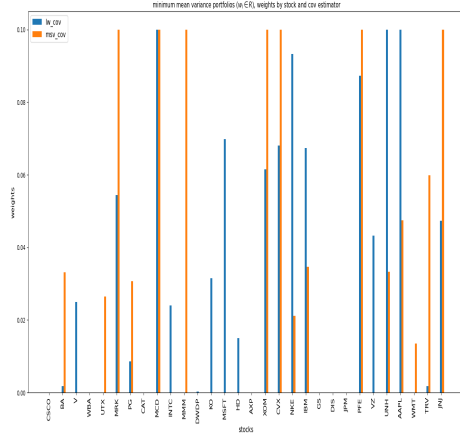
Figure E.1: Portfolio optimal weight allocation using the global minimum variance cost function.



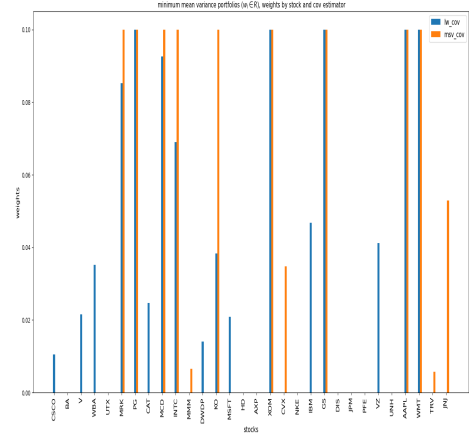
Optimal allocation first quarter



Optimal allocation second quarter



Optimal allocation third quarter



Optimal allocation fourth quarter

Figure E.2: Portfolio optimal weight allocation using the global minimum variance cost function for a fixed expected return.

E.2 P&L Results

E.2.1 Realized P&L Tables

	Mean R	Std. Dev R	Sharpe Ratio	L2	Ratio
Dow	10.37	10.78	0.96	-	-
lw_global	17.75	9.63	1.84	0.2	9.2
msv_global	14.44	9.88	1.46	0.28	5.21
lw_meanvar	17.8	9.8	1.82	0.2	9.1
msv_meanvar	14.67	10.37	1.41	0.28	5.04

Table E.1: First quarter training set, 21 periods out-of-sample optimal portfolio realized performance.

	Mean R	Std. Dev R	Sharpe Ratio	L2	Ratio
Dow	6.89	10.46	0.66	-	-
lw_global	14.95	8.94	1.67	0.27	6.19
msv_global	10.64	10.26	1.04	0.29	3.59
lw_meanvar	14.02	9.04	1.55	0.27	5.74
msv_meanvar	7.75	10.1	0.77	0.29	2.66

Table E.2: Second quarter training set, 21 periods out-of-sample optimal portfolio realized performance.

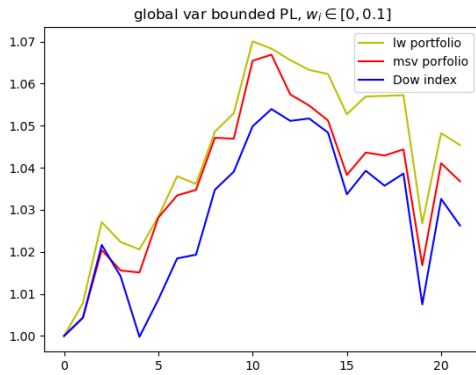
	Mean R	Std. Dev R	Sharpe Ratio	L2	Ratio
Dow	34.9	11	3.17	-	-
lw_global	32.8	9.87	3.32	0.27	12.3
msv_global	29.85	10	2.99	0.31	9.65
lw_meanvar	32.16	10.14	3.17	0.27	11.74
msv_meanvar	33.01	10.44	3.16	0.29	10.9

Table E.3: Third quarter training set, 21 periods out-of-sample optimal portfolio realized performance.

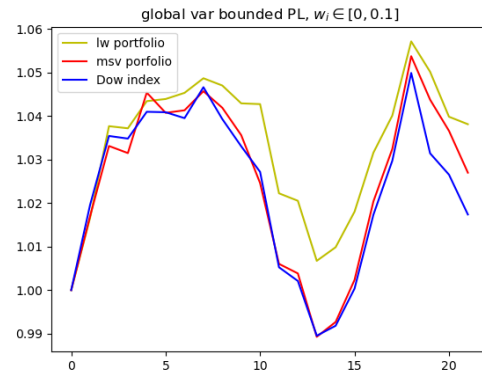
	Mean R	Std. Dev R	Sharpe Ratio	L2	Ratio
Dow	-6.48	9.59	-0.68	-	-
lw_global	3.03	8.77	0.35	0.28	1.25
msv_global	-2	9.22	-0.22	0.3	-0.73
lw_meanvar	1.97	8.83	0.22	0.28	0.79
msv_meanvar	-2.11	9.33	-0.23	0.31	-0.74

Table E.4: Fourth quarter training set, 21 periods out-of-sample optimal portfolio realized performance.

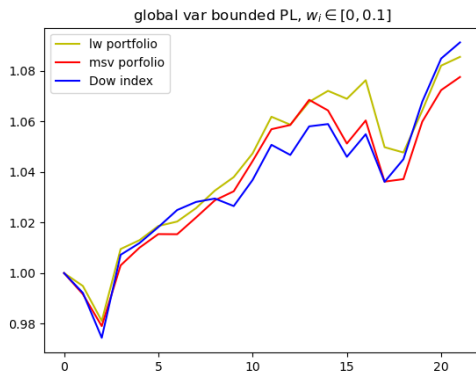
E.2.2 Out Of Sample Realized Trajectory, When \$1 Is Invested



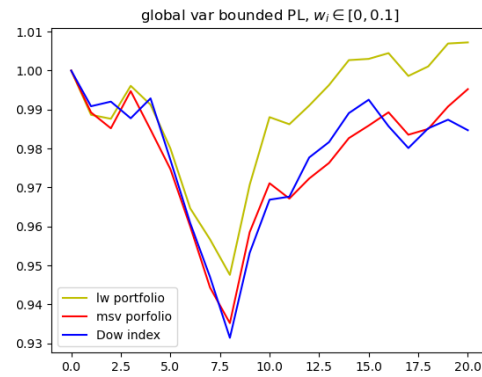
First quarter training, out of sample realized P&L trajectory for next 21 data points



Second quarter training, out of sample realized P&L trajectory for next 21 data points

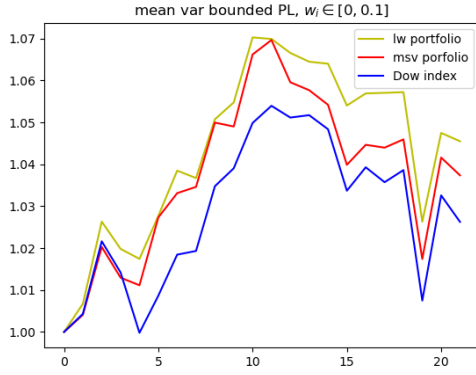


Third quarter training, out of sample realized P&L trajectory for next 21 data points

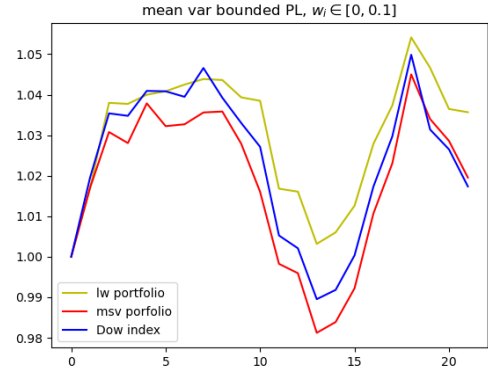


Fourth quarter training, out of sample realized P&L trajectory for next 20 data points

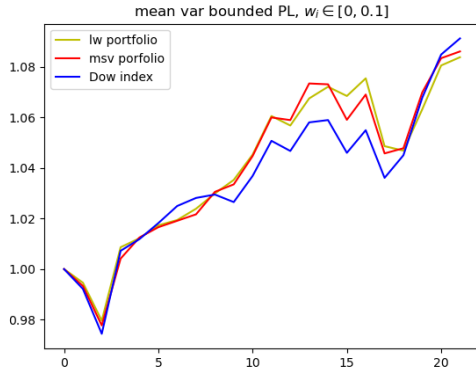
Figure E.3: Portfolio realized value trajectory for the various quarters when \$1 invested in the optimal portfolio given by global minimum variance cost function.



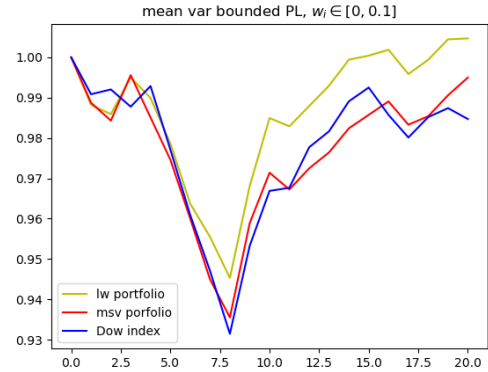
First quarter training, out of sample realized P&L trajectory for next 21 data points



Second quarter training, out of sample realized P&L trajectory for next 21 data points



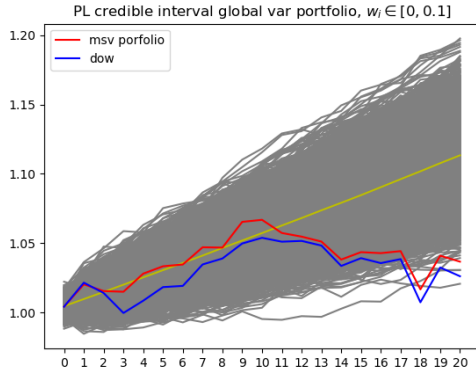
Third quarter training, out of sample realized P&L trajectory for next 21 data points



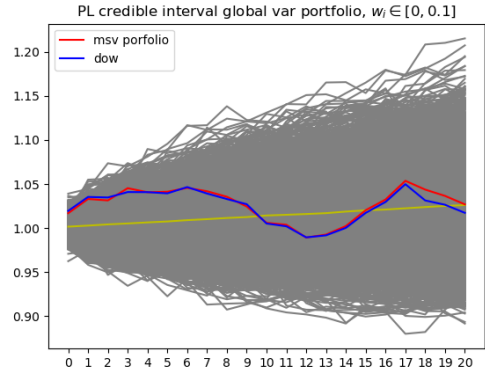
Fourth quarter training, out of sample realized P&L trajectory for next 20 data points

Figure E.4: Portfolio realized value trajectory for the various quarters when \$1 invested in the optimal portfolio given by global minimum variance cost function for a fixed level of expected return.

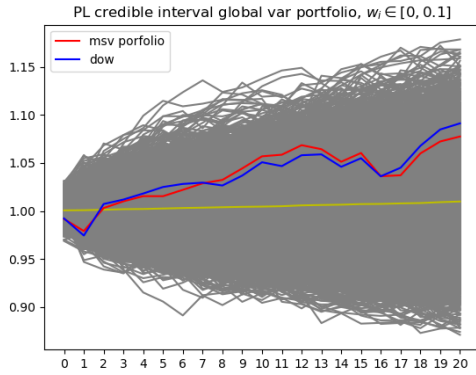
E.2.3 Credible Intervals



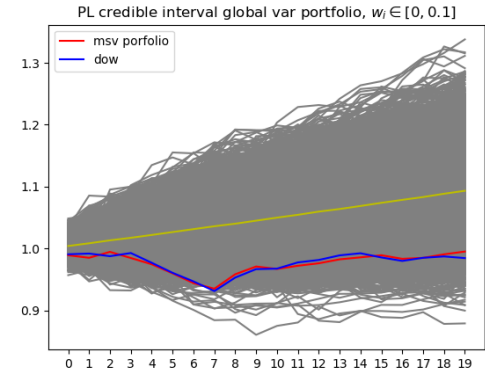
First quarter training, out of sample realized P&L trajectory for next 21 data points



Second quarter training, out of sample realized P&L trajectory for next 21 data points

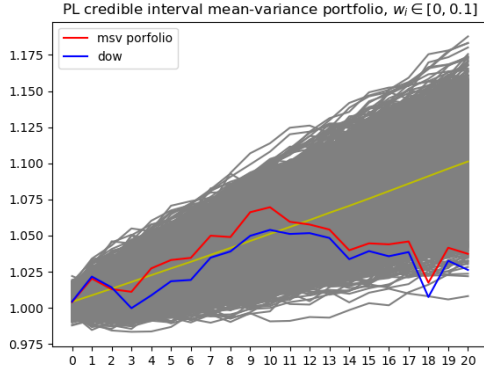


Third quarter training, out of sample realized P&L trajectory for next 21 data points

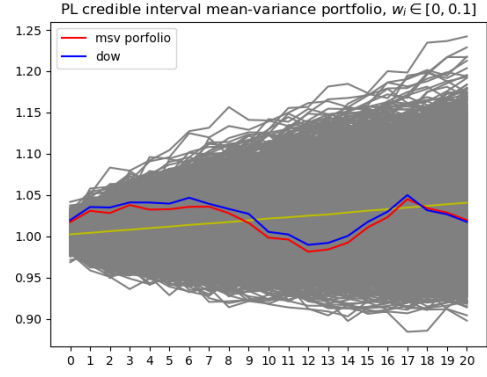


Fourth quarter training, out of sample realized P&L trajectory for next 20 data points

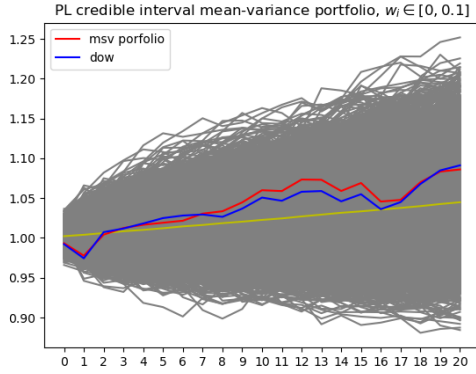
Figure E.5: MSV PL clouds created by 10000 simulated portfolio trajectories for $K = 15$ future time steps, based on optimal portfolio w_{T+K} realized value trajectory for the various quarters when \$1 invested in the optimal portfolio given by global minimum variance cost function. Red and blue lines represent the realized PL path of the optimal allocation w_{T+K} and the Dow Jones respectively.



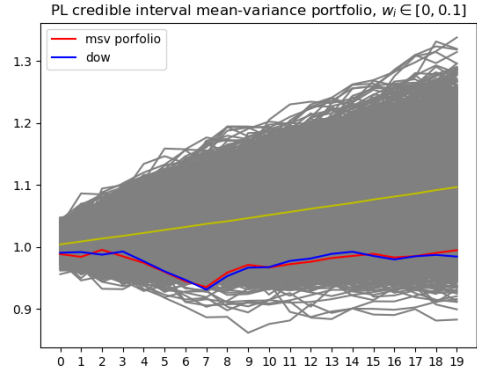
First quarter training, out of sample realized P&L trajectory for next 21 data points



Second quarter training, out of sample realized P&L trajectory for next 21 data points



Third quarter training, out of sample realized P&L trajectory for next 21 data points

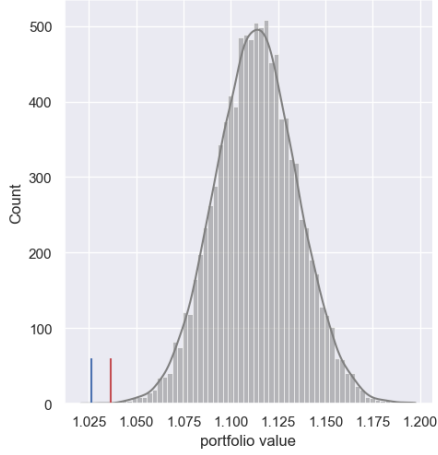


Fourth quarter training, out of sample realized P&L trajectory for next 20 data points

Figure E.6: MSV PL clouds created by 10000 simulated portfolio trajectories for $K = 15$ future time steps, based on optimal portfolio w_{T+K} realized value trajectory for the various quarters when \$1 invested in the optimal portfolio given by global minimum variance cost function for a given expected return. Red and blue lines represent the realized PL path of the optimal allocation w_{T+K} and the Dow Jones respectively.

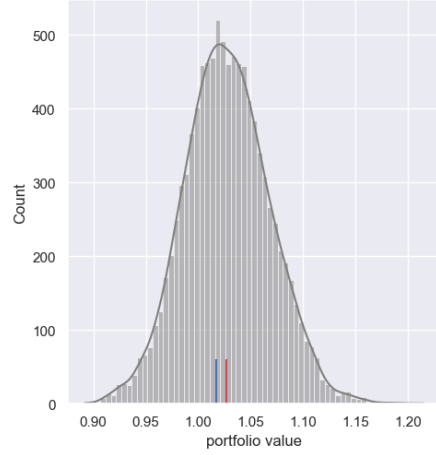
E.2.4 Optimal Portfolio Terminal Value Distribution

CI 21st day portfolio value distribution for global var MSV portfolio, $w_i \in [0, 0.1]$



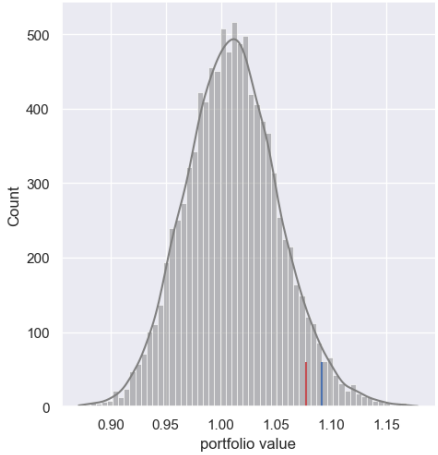
First quarter training, \$1 P&L distribution forecast

CI 21st day portfolio value distribution for global var MSV portfolio, $w_i \in [0, 0.1]$



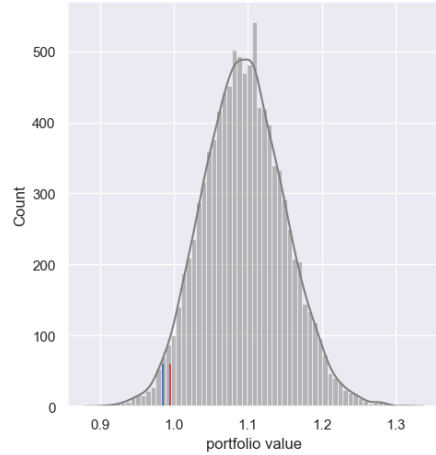
Second quarter training, \$1 P&L distribution forecast

CI 21st day portfolio value distribution for global var MSV portfolio, $w_i \in [0, 0.1]$



Third quarter training, \$1 P&L distribution forecast

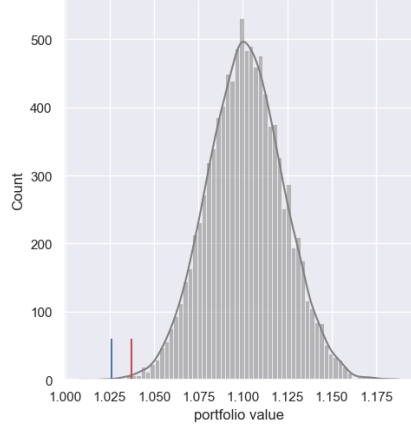
CI 21st day portfolio value distribution for global var MSV portfolio, $w_i \in [0, 0.1]$



Fourth quarter training, \$1 P&L distribution forecast

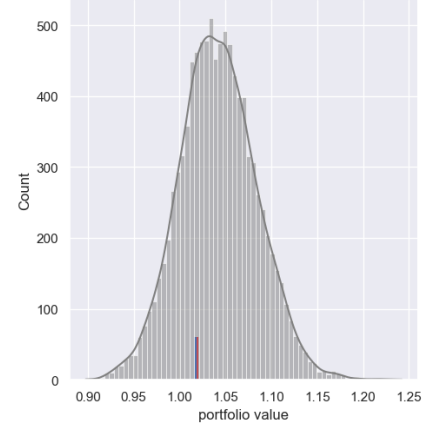
Figure E.7: Distribution of MSV terminal values generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function. The red and blue lines indicate the realized profit and loss of the optimal allocation w_{T+K} and the Dow Jones, respectively, for an initial investment of \$1.

CI portfolio 21st day value distribution for mean-variance MSV portfolio, $w_i \in [0, 0.1]$



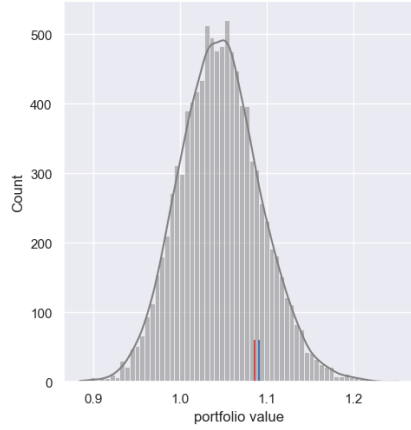
First quarter training, \$1 P&L distribution forecast

CI portfolio 21st day value distribution for mean-variance MSV portfolio, $w_i \in [0, 0.1]$



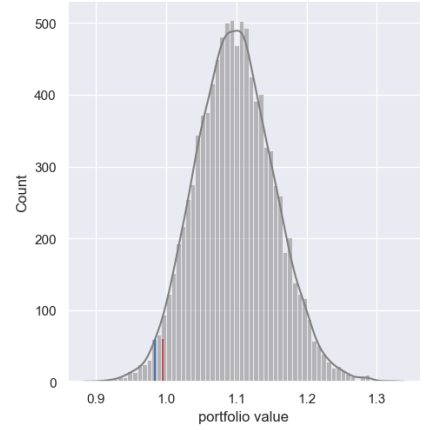
Second quarter training, \$1 P&L distribution forecast

CI portfolio 21st day value distribution for mean-variance MSV portfolio, $w_i \in [0, 0.1]$



Third quarter training, \$1 P&L distribution forecast

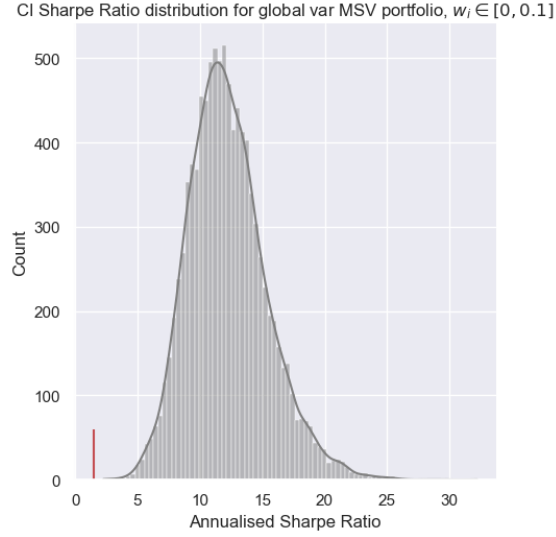
CI portfolio 21st day value distribution for mean-variance MSV portfolio, $w_i \in [0, 0.1]$



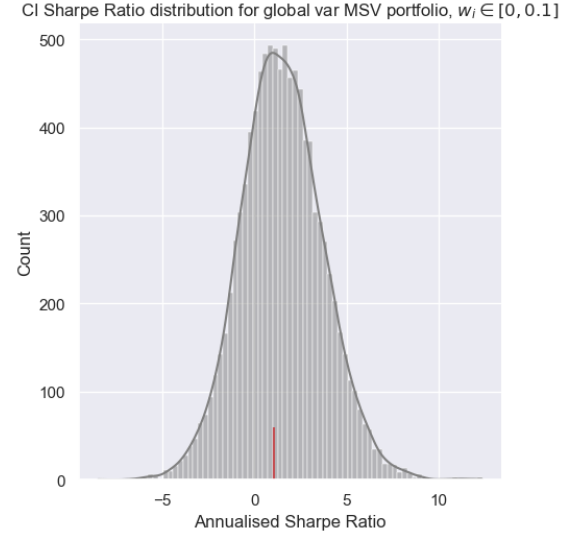
Fourth quarter training, \$1 P&L distribution forecast

Figure E.8: Distribution of MSV terminal values generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function for a given level of expected return. The red and blue lines indicate the realized profit and loss of the optimal allocation w_{T+K} and the Dow Jones, respectively, for an initial investment of \$1.

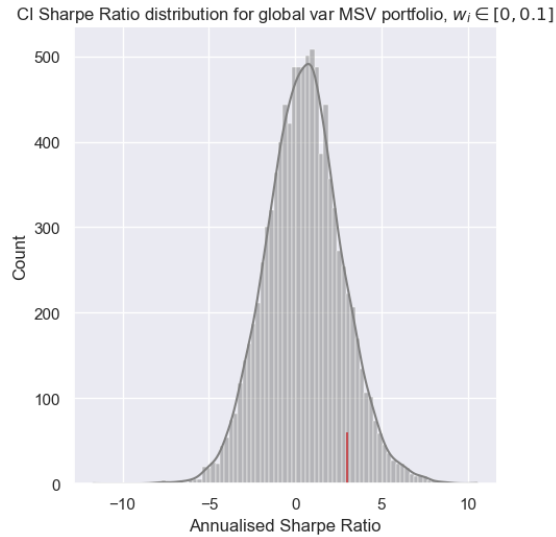
E.2.5 Optimal Portfolio Sharpe Ratio Distribution



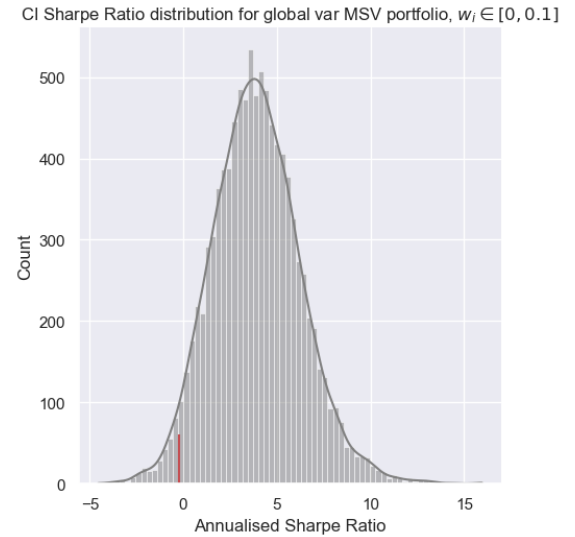
First quarter training, SR distribution forecast



Second quarter training, SR distribution forecast



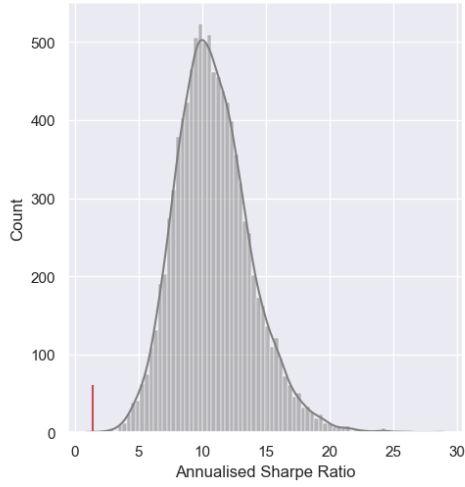
Third quarter training, SR distribution forecast



Fourth quarter training, SR distribution forecast

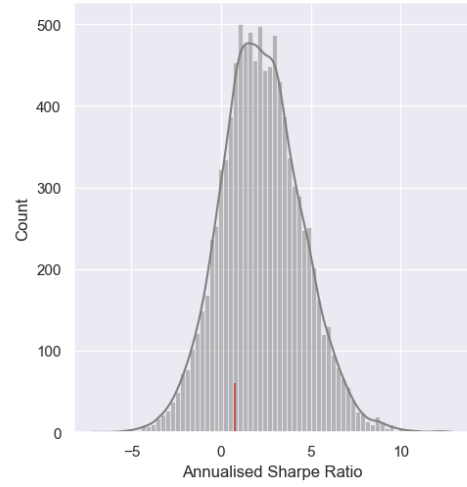
Figure E.9: Distribution of MSV SR generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function. The red line indicates the realized SR of the optimal allocation w_{T+K} for an initial investment of \$1.

CI Sharpe Ratio distribution for mean-variance MSV portfolio, $w_l \in [0, 0.1]$



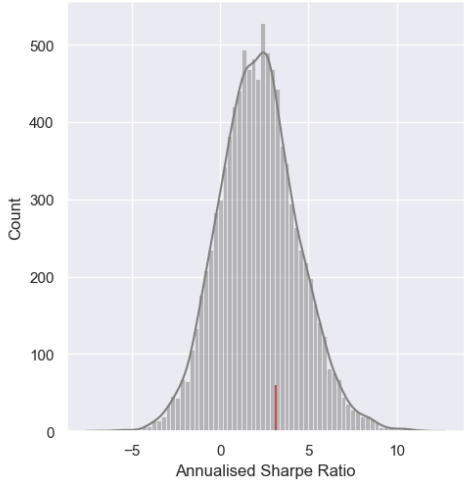
First quarter training, SR distribution forecast

CI Sharpe Ratio distribution for mean-variance MSV portfolio, $w_l \in [0, 0.1]$



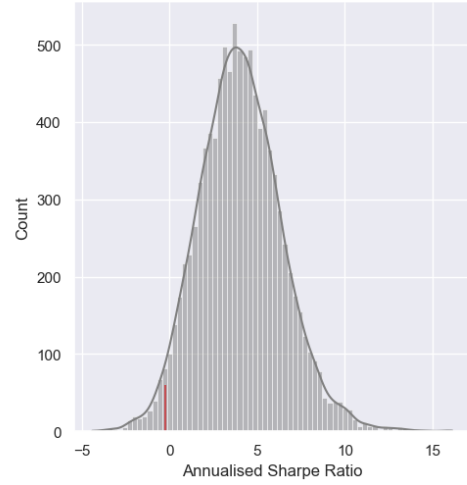
Second quarter training, SR distribution forecast

CI Sharpe Ratio distribution for mean-variance MSV portfolio, $w_l \in [0, 0.1]$



Third quarter training, SR distribution forecast

CI Sharpe Ratio distribution for mean-variance MSV portfolio, $w_l \in [0, 0.1]$



Fourth quarter training, SR distribution forecast

Figure E.10: Distribution of MSV SR generated from 10,000 simulated portfolio paths over $K = 15$ future time steps, based on the optimal portfolio w_{T+K} determined by the global minimum variance cost function for a given level of expected return. The red line indicates the realized SR of the optimal allocation w_{T+K} for an initial investment of \$1.

Bibliography

- Agrawal, A., Verschueren, R., Diamond, S. & Boyd, S. (2018), ‘A rewriting system for convex optimization problems’, *Journal of Control and Decision* **5**(1), 42–60.
- Aguilar, O. & West, M. (2000), ‘Bayesian dynamic factor models and portfolio allocation’, *Journal of Business & Economic Statistics* **18**(3), 338–357.
- Ait-Sahalia, Y. & Xiu, D. (2017), ‘Using principal component analysis to estimate a high dimensional factor model with high-frequency data’, *Journal of Econometrics* **201**(2), 384–399.
- Andrieu, C., De Freitas, N. & Doucet, A. (1999), Sequential mcmc for bayesian model selection, in ‘Proceedings of the IEEE Signal Processing Workshop on Higher-Order Statistics. SPW-HOS’99’, IEEE, pp. 130–134.
- Andrieu, C., Doucet, A. & Holenstein, R. (2010), ‘Particle markov chain monte carlo methods’, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **72**(3), 269–342.
- Andrieu, C. & Roberts, G. O. (2009), ‘The pseudo-marginal approach for efficient Monte Carlo computations’, *The Annals of Statistics* **37**(2), 697–725.
- Bai, J. & Shi, S. (2011), ‘Estimating high dimensional covariance matrices and its applications’.
- Barnard, J., McCulloch, R. & Meng, X.-L. (2000), ‘Modeling covariance matrices in terms of standard deviations and correlations, with application to shrinkage’, *Statistica Sinica* pp. 1281–1311.
- Beadle, E. R. & Djuric, P. M. (1997), ‘A fast-weighted bayesian bootstrap filter for non-linear model state estimation’, *IEEE Transactions on Aerospace and Electronic Systems* **33**(1), 338–343.
- Beskos, A., Crisan, D., Jasra, A., Kamatani, K. & Zhou, Y. (2017), ‘A stable particle filter for a class of high-dimensional state-space models’, *Advances in Applied Probability* **49**(1), 24–48.
- Bickel, P. J. (1982), ‘On adaptive estimation’, *The Annals of Statistics* pp. 647–671.

- Bishop, C. M. (2006), *Pattern recognition and machine learning*, springer.
- Black, F. & Litterman, R. (1992), ‘Global portfolio optimization’, *Financial Analysts Journal* **48**(5), 28–43.
URL: <https://www.jstor.org/stable/4479546>
- Bouchaud, J.-P., Matacz, A. & Potters, M. (2001), ‘Leverage effect in financial markets: The retarded volatility model’, *Physical review letters* **87**(22), 228701.
- Boyd, S. & Vandenberghe, L. (2004), *Convex optimization*, Cambridge university press.
- Bun, J., Bouchaud, J.-P. & Potters, M. (2016), ‘Cleaning correlation matrices’, *Risk magazine* **2015**.
- Cappé, O., Moulines, E. & Rydén, T. (2005), ‘Springer series in statistics’.
- Carvalho, C. M. & West, M. (2007), ‘Dynamic matrix-variate graphical models’, *Bayesian analysis* **2**(1), 69–97.
- Chamberlin, G. & Rothschild, M. (1983), ‘Arbitrage, factor structure, and mean-variance analysis on large asset markets’, *Econometrica* **51**(5), 1281–1304.
- Chan, J. C., Doucet, A., León-González, R. & Strachan, R. W. (2018), Multivariate stochastic volatility with co-heteroscedasticity, Working paper, Centre for Applied Macroeconomic Analysis (CAMA).
- Chib, S., Nardari, F. & Shephard, N. (2002), ‘Markov chain monte carlo methods for stochastic volatility models’, *Journal of Econometrics* **108**(2), 281–316.
- Chib, S., Nardari, F. & Shephard, N. (2006), ‘Analysis of high dimensional multivariate stochastic volatility models’, *Journal of Econometrics* **134**(2), 341–371.
- Chopin, N. (2002), ‘A sequential particle filter method for static models’, *Biometrika* **89**(3), 539–552.
- Chopin, N. (2007), ‘Inference and model choice for sequentially ordered hidden markov models’, *Journal of the Royal Statistical Society Series B: Statistical Methodology* **69**(2), 269–284.
- Chopin, N., Jacob, P. E. & Papaspiliopoulos, O. (2013), ‘Smc2: an efficient algorithm for sequential analysis of state space models’, *Journal of the Royal Statistical Society Series B: Statistical Methodology* **75**(3), 397–426.
- Chopin, N., Ridgway, J., Gerber, M. & Papaspiliopoulos, O. (2015), ‘Towards automatic calibration of the number of state particles within the smc 2 algorithm’, *arXiv preprint arXiv:1506.00570*.

- Connor, G. & Korajczyk, R. A. (1995), ‘The arbitrage pricing theory and multifactor models of asset returns’, *Handbooks in operations research and management science* **9**, 87–144.
- Daniels, M. J. & Kass, R. E. (2001), ‘Shrinkage estimators for covariance matrices’, *Biometrics* **57**(4), 1173–1184.
- Del Moral, P., Doucet, A. & Jasra, A. (2012), ‘An adaptive sequential monte carlo method for approximate bayesian computation’, *Statistics and computing* **22**(5), 1009–1020.
- Dellaportas, P., Titsias, M. K., Petrova, K. & Plataniotis, A. (2021), ‘Scalable inference for a full multivariate stochastic volatility model’, *Journal of Econometrics* .
- DeMiguel, V., Garlappi, L., Nogales, F. J. & Uppal, R. (2009), ‘A generalized approach to portfolio optimization: Improving performance by constraining portfolio norms’, *Management science* **55**(5), 798–812.
- Diamond, S. & Boyd, S. (2016), ‘CVXPY: A Python-embedded modeling language for convex optimization’, *Journal of Machine Learning Research* **17**(83), 1–5.
- Ding, Z. & Granger, C. W. (1996), ‘Modeling volatility persistence of speculative returns: a new approach’, *Journal of econometrics* **73**(1), 185–215.
- Douc, R. & Cappé, O. (2005), Comparison of resampling schemes for particle filtering, in ‘ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005.’, Ieee, pp. 64–69.
- Doucet, A., Johansen, A. M. et al. (2009), ‘A tutorial on particle filtering and smoothing: Fifteen years later’, *Handbook of nonlinear filtering* **12**(656-704), 3.
- Efron, B. & Morris, C. (1977), ‘Stein’s paradox in statistics’, *Scientific American* **236**(5), 119–127.
- Fama, E. F. (1965), ‘The behavior of stock-market prices’, *The journal of Business* **38**(1), 34–105.
- Fan, J., Han, F. & Liu, H. (2014), ‘Challenges of big data analysis’, *National science review* **1**(2), 293–314.
- Fan, J., Liao, Y. & Liu, H. (2016), ‘An overview of the estimation of large covariance and precision matrices’, *The Econometrics Journal* **19**(1), C1–C32.
- Friedman, J., Hastie, T. & Tibshirani, R. (2001), *The elements of statistical learning*, Vol. 1, Springer series in statistics New York.
- Gandy, A. & Veraart, L. A. (2013), ‘The effect of estimation in high-dimensional portfolios’, *Mathematical Finance: An International Journal of Mathematics, Statistics and Financial Economics* **23**(3), 531–559.

- Geweke, J. et al. (1994), Bayesian comparison of econometric models, Technical report, Citeseer.
- Ghysels, E., Harvey, A. C. & Renault, E. (1996), ‘5 stochastic volatility’, *Handbook of statistics* **14**, 119–191.
- Gouriéroux, C., Jasiak, J. & Sufana, R. (2009), ‘The wishart autoregressive process of multivariate stochastic volatility’, *Journal of Econometrics* **150**(2), 167–181.
- Griffin, J. & Brown, P. (2017), ‘Hierarchical shrinkage priors for regression models’, *Bayesian Analysis* **12**(1), 135–159.
- Grinold, R. C. & Kahn, R. N. (2000), ‘Active portfolio management’.
- Haff, L. (1980), ‘Empirical Bayes estimation of the multivariate normal covariance matrix’, *The Annals of Statistics* pp. 586–597.
- Han, Y. (2006), ‘Asset allocation with a high dimensional latent factor stochastic volatility model’, *The Review of Financial Studies* **19**(1), 237–271.
- Harvey, A. C. & Shephard, N. (1996), ‘Estimation of an asymmetric stochastic volatility model for asset returns’, *Journal of Business & Economic Statistics* **14**(4), 429–434.
- Harvey, A., Ruiz, E. & Shephard, N. (1994), ‘Multivariate stochastic variance models’, *The Review of Economic Studies* **61**(2), 247–264.
- He, G. & Litterman, R. (1999), ‘The intuition behind black-litterman model portfolios’, Goldman Sachs Asset Management White Paper.
URL: <https://www.quantresearch.org/downloads/BLHeLitterman.pdf>
- Hosszejni, D. & Kastner, G. (2019), ‘Modeling univariate and multivariate stochastic volatility in R with stochvol and factorstochvol’, *R package vignette* .
URL: <https://CRAN.R-project.org/package=factorstochvol/vignettes/paper.pdf>
- Hotelling, H. (1933), ‘Analysis of a complex of statistical variables into principal components.’, *Journal of educational psychology* **24**(6), 417.
- Huang, A., Wand, M. P. et al. (2013), ‘Simple marginally noninformative prior distributions for covariance matrices’, *Bayesian Analysis* **8**(2), 439–452.
- Huber, F. & Feldkircher, M. (2019), ‘Adaptive shrinkage in bayesian vector autoregressive models’, *Journal of Business & Economic Statistics* **37**(1), 27–39.
- Hull, J. C. (2003), *Options futures and other derivatives*, Pearson Education India.
- Hull, J. & White, A. (1987), ‘The pricing of options on assets with stochastic volatilities’, *The journal of finance* **42**(2), 281–300.

- Ishihara, T. & Omori, Y. (2017), ‘Portfolio optimization using dynamic factor and stochastic volatility: evidence on fat-tailed errors and leverage’, *The Japanese Economic Review* **68**(1), 63–94.
- Jacquier, E., Polson, N. G. & Rossi, P. E. (2004), ‘Bayesian analysis of stochastic volatility models with fat-tails and correlated errors’, *Journal of Econometrics* **122**(1), 185–212.
- Kantas, N., Doucet, A., Singh, S. S., Maciejowski, J., Chopin, N. et al. (2015), ‘On particle methods for parameter estimation in state-space models’, *Statistical science* **30**(3), 328–351.
- Kastner, G. (2019), ‘Sparse bayesian time-varying covariance estimation in many dimensions’, *Journal of Econometrics* **210**(1), 98–115.
- Kourtis, A., Dotsis, G. & Markellos, R. N. (2012), ‘Parameter uncertainty in portfolio selection: Shrinking the inverse covariance matrix’, *Journal of Banking & Finance* **36**(9), 2522–2531.
- Lai, T. L., Xing, H., Chen, Z. et al. (2011), ‘Mean–variance portfolio optimization when means and covariances are unknown’, *The Annals of Applied Statistics* **5**(2A), 798–823.
- Ledoit, O. & Wolf, M. (2003), ‘Improved estimation of the covariance matrix of stock returns with an application to portfolio selection’, *Journal of empirical finance* **10**(5), 603–621.
- Ledoit, O. & Wolf, M. (2004a), ‘Honey, i shrunk the sample covariance matrix’, *The Journal of Portfolio Management* **30**(4), 110–119.
- Ledoit, O. & Wolf, M. (2004b), ‘A well-conditioned estimator for large-dimensional covariance matrices’, *Journal of multivariate analysis* **88**(2), 365–411.
- Lindberg, C. et al. (2009), ‘Portfolio optimization when expected stock returns are determined by exposure to risk’, *Bernoulli* **15**(2), 464–474.
- Marčenko, V. A. & Pastur, L. A. (1967), ‘Distribution of eigenvalues for some sets of random matrices’, *Mathematics of the USSR-Sbornik* **1**(4), 457.
- Markowitz, H. (1952), ‘Portfolio selection’, *The journal of finance* **7**(1), 77–91.
- Merton, R. C. (1972), ‘An analytic derivation of the efficient portfolio frontier’, *Journal of financial and quantitative analysis* **7**(4), 1851–1872.
- Meucci, A. (2006), ‘Beyond black-litterman: Views on non-normal markets’, *Risk* **19**(2), 87–92.
URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1117574
- Meyer, R. & Yu, J. (2000), ‘Bugs for a Bayesian analysis of stochastic volatility models’, *The Econometrics Journal* **3**(2), 198–215.

- Moore, J. B. & Weiss, H. (1979), ‘Recursive prediction error methods for adaptive estimation’, *IEEE Transactions on Systems, Man, and Cybernetics* **9**(4), 197–205.
- Nakajima, J. & West, M. (2012), ‘Dynamic factor volatility modeling: A bayesian latent threshold approach’, *Journal of Financial Econometrics* **11**(1), 116–153.
- O’Malley, A. J. & Zaslavsky, A. M. (2008), ‘Domain-level covariance analysis for multilevel survey data with structured nonresponse’, *Journal of the American Statistical Association* **103**(484), 1405–1418.
- Philipov, A. & Glickman, M. E. (2006), ‘Multivariate stochastic volatility via wishart processes’, *Journal of Business & Economic Statistics* **24**(3), 313–328.
- Pitt, M. & Shephard, N. (1999), ‘Time varying covariances: a factor stochastic volatility approach’, *Bayesian statistics* **6**, 547–570.
- Plataniotis, A. & Dellaportas, P. (2010), ‘A multivariate stochastic volatility model’.
- Plerou, V., Gopikrishnan, P., Rosenow, B., Amaral, L. A. N., Guhr, T. & Stanley, H. E. (2002), ‘Random matrix approach to cross correlations in financial data’, *Physical Review E* **65**(6), 066126.
- Roll, R. (1977), ‘A critique of the asset pricing theory’s tests part i: On past and potential testability of the theory’, *Journal of financial economics* **4**(2), 129–176.
- Ruiz, E. (1994), ‘Quasi-maximum likelihood estimation of stochastic volatility models’, *Journal of econometrics* **63**(1), 289–306.
- Sharpe, W. F. (1963), ‘A simplified model for portfolio analysis’, *Management science* **9**(2), 277–293.
- Sharpe, W. F. (1966), ‘Mutual fund performance’, *The Journal of business* **39**(1), 119–138.
- Sharpe, W. F. (1994), ‘The sharpe ratio’, *Journal of portfolio management* **21**(1), 49–58.
- Stein, C. (1956), Inadmissibility of the usual estimator for the mean of a multivariate normal distribution, Technical report, Stanford University Stanford United States.
- Taylor, S. (1986), ‘Modelling financial time series’, *Chichester* .
- Taylor, S. J. (1994), ‘Modeling stochastic volatility: A review and comparative study’, *Mathematical finance* **4**(2), 183–204.
- Tracy, C. A. & Widom, H. (2002), ‘Distribution functions for largest eigenvalues and their applications’, *arXiv preprint math-ph/0210034* .
- Wigner, E. P. (1958), ‘On the distribution of the roots of certain symmetric matrices’, *Annals of Mathematics* pp. 325–327.

- Xu, Y. & Jasra, A. (2019), ‘Particle filters for inference of high-dimensional multivariate stochastic volatility models with cross-leverage effects’, *Foundations of Data Science* **1**(1), 61.
- Yu, J. (2005), ‘On leverage in a stochastic volatility model’, *Journal of Econometrics* **127**(2), 165–178.
- Zhou, X. Y. & Li, D. (2000), ‘Continuous-time mean-variance portfolio selection: A stochastic lq framework’, *Applied Mathematics and Optimization* **42**(1), 19–33.