COVARIANCE MATRIX ESTIMATION

AND ITS APPLICATIONS

By

YONG HU

Bachelor of Science
Zhejiang University
Hangzhou, China
1993

Master of Science
Zhejiang University
Hangzhou, China
1996

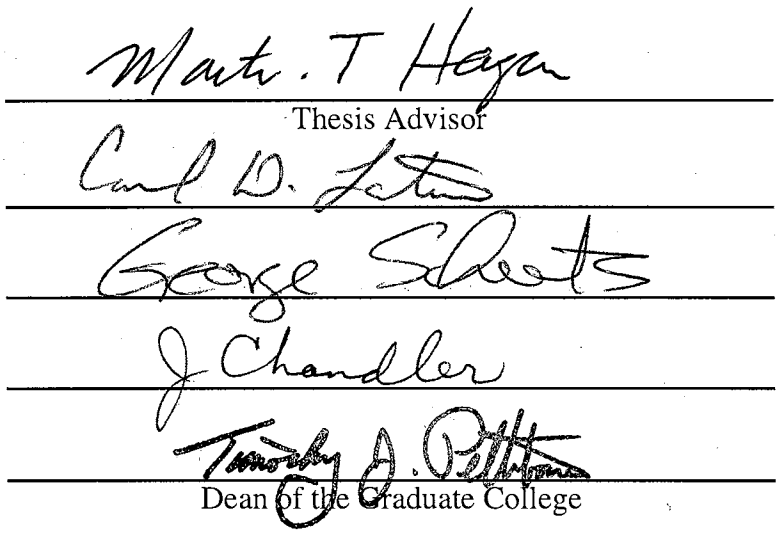
Master of Science
Oklahoma State University
Stillwater, Oklahoma
2000

Submitted to the Faculty
of the Graduate College of
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
August, 2003

COVARIANCE MATRIX ESTIMATION

AND ITS APPLICATIONS

Thesis Approved:



Thesis Advisor

Dean of the Graduate College

ACKNOWLEDGMENTS

I would like to express my deep gratitude to my major advisor, Dr. Martin Hagan, for guiding me through the past three years of study, for the time and energy he put in developing my research skills, for his encouragement when I was frustrated with my thesis work, for his patience in correcting my English writing, and for his kindness and friendship showed not only to me, but also to his other students.

I would also like to thank my other committee members Dr. John P. Chandler, Dr. Carl D. Latino, and Dr. George Scheets Jr. for their helpful suggestions and assistance.

Special thanks go to M. Fun, A. Khaled, A. Pukrittayakamee, for their friendship and heated discussion of almost everything in this research group. Their encouragement and support have made my days at Oklahoma State University pleasant and unforgettable.

I would like to express my deep appreciation to my wife, Jing Ding, for her years of loving support and encouragement. I would also thank my uncles, Zhongyi Song and Zhongyang Song, for their encouragement and guidance since my early childhood. I wish to thank my parents and my wife's parents for all they have done to support all my endeavors, especially the encouragement for this one.

TABLE OF CONTENTS

Chapter Page

1 INTRODUCTION 1

2 SHRINKAGE COVARIANCE ESTIMATION 6

Loss Functions 6

Shrinking the Eigenvalues 7

Shrinking towards Structures 11

Shrinking toward Identity Matrix 12

Ridge Regression 12

Ledoit Covariance Matrix Estimator (LCME) 13

3 RIDGE COVARIANCE MATRIX ESTIMATOR 18

Derivation and Interpretation 18

Estimating Optimal Shrinkage Intensity 23

Ledoit's method 24

Filtering method 26

Constraint method 27

4 SHRINKAGE LEAST SQUARES AND SHRINKAGE RECURSIVE LEAST SQUARES 29

James-Stein Estimation (JS) 30

Shrinkage Least Squares (SLS) 32

The Model 32

James-Stein Least Squares (JSLS) 32

Shrinkage Least Squares (SLS) 34

Shrinkage Recursive Least Squares (SRLS) 35

Chapter Page

The Model 36
Recursive Least Squares (RLS) 37
James-Stein Recursive Least Squares (JSRLS) 39
Shrinkage Recursive Least Squares (SRLS) 41
James-Stein Ledoit Recursive Least Squares (JSLRLS) 44

5 SIMULATION RESULTS: RIDGE COVARIANCE ESTIMATION 46

RCME: Comparison by Monte Carlo Simulation 46
SRLS: Comparison by Monte Carlo Simulation 52

6 BAYESIAN METHODS 58

Bayes' Theorem 58
Bayesian Covariance Matrix Estimation 59
Bayesian Regularization 63

7 HIERARCHICAL BAYESIAN COVARIANCE MATRIX ESTIMATOR 69

Assumptions 69
Derivation 70
Likelihood function 71
Prior density function 72
Posterior density function 73
Total probability 73
Estimating a and b 75
Estimating $\mathbf{C}^{MP}$ 78
Computing $\mathbf{H}^{MP}$ 82
Other Computational Issues 83
Summary of the Algorithm 84
Extension to $p + 1$ unknown parameters 85

8 SIMULATION RESULTS: HIERARCHICAL BAYESIAN COVARIANCE MATRIX ESTIMATOR 89

Chapter Page

Monte Carlo Simulation: basic HBCME 90
Effect of standard deviation σ 92
Effect of sample size N 97
Effect of n_c 101
Monte Carlo Simulation: Extended HBCME 105

9 APPLICATION TO PORTFOLIO OPTIMIZATION 111

Background 112
Simulated Stock Data 117
Real Stock Data 121

10 CONCLUSIONS 125

Ridge Covariance Matrix Estimator 125
Hierarchical Bayesian Covariance Matrix Estimator 126
Shrinkage Least Squares and Shrinkage Recursive Least Squares 127
Portfolio Optimization 127

REFERENCES 129

APPENDIX: SOME THEOREMS ON MATRIX DERIVATIVES 132

LIST OF TABLES

Table		Page
5-1	Relative improvement for $\mathbf{y} = [0.6, 4, 1, 2, 3, 4]^T$	55
5-2	Relative improvement for $\mathbf{y} = [0.6, 0.4, 0.1, 0.2, 0.3, 0.4]^T$	57
9-1	Portfolio Performance Comparison with Simulated Data	120
9-2	Sharpe Ratio Comparison for Real Stock Data with $T = 100$	123
9-3	Sharpe Ratio Comparison for Real Stock Data with $T = 60$	124

LIST OF FIGURES

Figure		Page
3-1	Errors of Ridge Covariance Matrix Estimator	21
5-1	PRIAL of $\hat{\mathbf{C}}_R$, $\hat{\mathbf{C}}_R^o$, and $\hat{\mathbf{C}}_L$	49
5-2	Shrinkage Intensity	51
8-1	RMS estimation error vs. σ (covariance structure: multiple of identity)	93
8-2	t-statistic vs. σ (covariance structure: multiple of identity)	95
8-3	E_C and E_D vs. σ (covariance structure: multiple of identity)	95
8-4	a and b vs. σ (covariance structure: multiple of identity)	97
8-5	RMS estimation error vs. N (covariance structure: multiple of identity)	98
8-6	t-statistic vs. N (covariance structure: multiple of identity)	99
8-7	E_C and E_D vs. N (covariance structure: multiple of identity)	99
8-8	a and b vs. N (covariance structure: multiple of identity)	100
8-9	RMS estimation error vs. n_c (covariance structure: multiple of identity)	102
8-10	t-statistic vs. n_c (covariance structure: multiple of identity)	103
8-11	E_C and E_D vs. n_c (covariance structure: multiple of identity)	103
8-12	a and b vs. n_c (covariance structure: multiple of identity)	104
8-13	RMS estimation error vs. σ (covariance structure: diagonal)	106
8-14	t-statistic vs. σ (covariance structure: diagonal)	107
8-15	RMS estimation error vs. N (covariance structure: diagonal)	108

Figure Page

8-16 t statistic vs. N (covariance structure: diagonal) 108

8-17 RMS estimation error vs. n_c (covariance structure: diagonal) 109

8-18 t statistic vs. n_c (covariance structure: diagonal) 110

9-1 Efficient Frontier 113

LIST OF SYMBOLS

A - lower triangular matrix with positive diagonal elements (see Eq. (2-5))

B - matrix with normalized eigenvector of **S** (see Eq. (2-7))

C - true covariance matrix

$\hat{\mathbf{C}}_L$ - covariance matrix estimated by the LCME (see Eq. (2-27))

$\hat{\mathbf{C}}_R$ - covariance matrix estimated by the RCME (see Eq. (3-2))

D - diagonal matrix, with $\mathbf{D} = diag(d_1, d_2, \ldots, d_p)$ (see Eq. (2-5))

G - prior of the covariance matrix (see Eq. (2-29))

H - Hessian of F (see Eq. (7-17))

I - identity matrix

L - diagonal matrix, with $\mathbf{L} = diag(l_1, l_2, \ldots, l_p)$ (see Eq. (2-7))

O - defined as $(\mathbf{T}\mathbf{T}^T)^{-1}$ (see Eq. (4-6))

P - sample correlation matrix (see Eq. (2-10))

Q - diagonal matrix, with $\mathbf{Q} = diag(q_1, q_2, \ldots, q_p)$ (see Eq. (3-10))

R - observation matrix with dimension $N \times p$ (see Eq. (4-5))

S - sample covariance matrix

T - measurement noise matrix with dimension $N \times N$ (see Eq. (4-5))

$\mathbf{V}$ - defined as $\sqrt{diag(\mathbf{S})}$ (see Eq. (2-10))

$\mathbf{X}$ - $N \times p$ observation matrix (see Eq. (2-11))

$\mathbf{Y}$ - decomposition of $\mathbf{C}$, $\mathbf{C} = \mathbf{Y}\mathbf{Y}^T$, see HBCME derivation

A - population parameters in Bayes' theorem (see Eq. (6-1))

B - observed data in Bayes' theorem (see Eq. (6-1))

C - an arbitrary positive definite matrix (see Eq. (6-9))

D - observed data set

E_C - prior error of $\mathbf{C}$ (see Eq. (7-9))

E_D - data error in Bayesian regularization (see Eq. (6-13) and Eq. (7-5))

E_W - regularization error in Bayesian regularization (see Eq. (6-14))

E_M - model error in Bayesian regularization (see Eq. (6-15) and Eq. (7-13))

F - total error in HBCME (see Eq. (7-15))

J - objective function or cost function

L_i - loss function (see Eq. (2-1), (2-2), and (2-3))

M - a specific neural network model (see Eq. (6-16))

N - number of observations

R_i - risk function (see Eq. (2-4))

Z_C - coefficient of prior density function for $\mathbf{C}$ (see Eq. (7-10))

Z_D - coefficient of likelihood function for $\mathbf{C}$ (see Eq. (7-6))

$\mathbf{c}$ - vector containing upper triangular elements in $\mathbf{C}$ (see Eq. (7-16))

$\mathbf{r}(n)$ - known observation and input set at time instant n (see Eq. (4-21))

$\mathbf{w}$ - $n \times n$ noise vector (see Eq. (4-5))

$\mathbf{w}_p$ - portfolio weights (see Eq. (9-1))

$\mathbf{x}$, $\mathbf{x}_i$ - $p \times 1$ observation vector

$\bar{\mathbf{x}}$ - sample mean of $\mathbf{x}$ (see Eq. (6-2))

$\mathbf{y}$ - parameter vector (see Eq. (4-5)); or defined as $vec(\mathbf{Y})$ (see Eq. (7-32))

$\mathbf{y}^{\circ}$ - transformation of $\mathbf{x}$ in James-Stein's estimator (see Eq. (4-2))

$\mathbf{z}$ - $n \times 1$ vector containing observations (see Eq. (2-11) and Eq. (4-5))

$\mathbf{z}(n)$ - $\mathbf{z}$ at time instant n (see Eq. (4-23))

a - inverse of prior variance in HBCME (see Eq. (7-1)); or parameter in Bayesian Regularization (see Eq. (6-15))

a_i - neural network output (see Eq. (6-12))

b - the multiple of the identity matrix in HBCME (see Eq. (7-1)); or parameter in Bayesian Regularization (see Eq. (6-15))

c - generic estimator coefficient (see Eq. (2-28) and Eq. (6-9))

c_{ij} - elements of $\mathbf{C}$

d - coefficient in Dey and Srinivasan's estimator (see Eq. (2-9))

d_i - elements of $\mathbf{D}$ in Stein's estimator (see Eq. (2-6))

e - estimation error

k - shrinkage intensity in RCME (see Eq. (3-2))

k_o - optimal value of k (see Eq. (3-7))

$\hat{k}_o^L$ - Ledoit's estimate of k_o

$\hat{k}_o^F$ - estimate of k_o by the filtering method

k_1, k_2 - shrinkage intensity in LCME (see Eq. (2-15))

l_i - eigenvalues of $(N-1)\mathbf{S}$

n - time instant; or, number of degree of freedom (see Eq. (6-3))

n_a, n_b - orders in ARX model (see Eq. (4-19))

n_{max} - defined as $max(n_a, n_b) + 1$

p - number of variables

p_i - neural network input (see Eq. (6-12))

p^e - effective dimension, defined as $\frac{\mathrm{tr}(\mathbf{C})}{\lambda_{max}(\mathbf{C})}$ (see Eq. (4-4))

q - desired portfolio return (see Eq. (9-5))

r - risk aversion factor (see Eq. (9-1))

t_i - neural network target value (see Eq. (6-12))

u - Dey and Srinivasan's estimator coefficient (see Eq. (2-9)); or the utility of a stock portfolio (see Eq. (9-1))

$u(n)$ - exogenous input at time instant n

$w(n)$ - white noise with normal distribution at time instant n (see Eq. (4-19))

$z(n)$ - observed output at time instant n (see Eq. (4-19))

μ, α, β, δ - Ledoit estimator parameters (see Eq. (2-17) - (2-20))

μ - true mean vector

μ_p - expected return of the portfolio (see Eq. (9-3))

$\hat{\mu}^{JS}$ - μ estimated by the James-Stein estimator (see Eq. (4-1))

σ^2 - variance of a random variable

σ_P^2 - variance of the portfolio return (see Eq. (9-3))

θ - ridge regression parameter (see Eq. (2-11))

υ - ridge perturbation factor, ridge regression (see Eq. (2-13))

λ - forgetting factor in RLS

$\lambda_{max}(\mathbf{C})$ - the maximum eigenvalue of $\mathbf{C}$

λ - the set containing eigenvalues of $\mathbf{C}$.

ε, ε - representing scalar and vector error, respectively

ν - the number of degree of freedom in Chen's algorithm (see Eq. (2-29))

γ - defined as $\|\mathbf{C} - \mathbf{I}\|^2$ (see Eq. (3-4))

η - defined as $E[\|\mathbf{S} - \mathbf{C}\|^2]$ (see Eq. (3-5))

ω - weighting factor in filtering method for estimating k_o (see Eq. (3-21))

κ - weighting factor in JSRLS and SRLS (see Eq. (4-57) and (4-79))

ι - prior mean of the mean, in Anderson's method (see Eq. (6-5))

δ_{ij} - delta function (see Eq. (7-2))

Σ - prior mean of $\mathbf{C}$

$\|\cdot\|$ - Frobenius norm

$(\cdot)^+$ - defined as $\max(0, \cdot)$

$\|\cdot\|_N$ - normalized Frobenius norm, defined as $\frac{\|\cdot\|}{p}$

$N(\cdot, \cdot)$ - normal distribution

$E[\cdot]$ - mathematical expectation

$f(\cdot)$, $g(\cdot)$ - generic functions with range in R^1

$l(\cdot)$ - likelihood function (see Eq. (4-8))

$tr(\cdot)$ - trace of a matrix

LIST OF ACRONYMS

ARX - AutoRegressive with eXogenous input

BFGS - Broyden-Fletcher-Goldfarb-Shanno

HBCME - Hierarchical Bayesian Covariance Matrix Estimator

i.i.d. - Independent and Identically Distributed

JSLRLS - James-Stein-Ledoit Recursive Least Squares

JSLS - James-Stein Least Squares

JSRLS - James-Stein Recursive Least Squares

LCME - Ledoit Covariance Matrix Estimator

LS - Least Squares

MLE - Maximum Likelihood Estimate

MSE - Mean Square Error

MV - Mean Variance

PRIAL - Percentage Relative Improvement In Average Loss

RCME - Ridge Covariance Matrix Estimator

RLS - Recursive Least Squares

RMS - Root Mean Square

SLS - Shrinkage Least Squares

SRLS - Shrinkage Recursive Least Squares

SR_p - Sharpe Ratio of a portfolio

CHAPTER 1

INTRODUCTION

Covariance matrix estimation is a fundamental problem, and has applications in many areas. The sample covariance matrix is the most widely used covariance estimator. The sample covariance matrix has some appealing properties, such as being unbiased and asymptotically normally distributed. Maximum Likelihood Estimation (MLE) is another popular method for estimating the covariance matrix. It is similar to the sample covariance matrix, and is asymptotically normally distributed. However, it is biased.

If the sample size is small, the estimation errors of both the sample covariance matrix and the MLE can be so large that the estimated covariance matrix is of little use. In addition, many applications require the calculation of the inverse of the covariance matrix. If the sample size is too small, the estimated covariance matrix can be singular or close to singular. For example, in mean-variance portfolio optimization (see Chapter 9), one needs the inverse of the covariance matrix to calculate the optimal portfolio weights. Often the sample size can be much smaller than the number of stocks in the portfolio. In this case, the inverse of the covariance matrix cannot be calculated by the sample covariance matrix nor by MLE.

There are generally two approaches to improve the covariance matrix estimation when the sample size is small. Both approaches assume some prior knowledge of the covariance matrix. The first approach is the shrinkage method, which shrinks the eigenvalues of the sample covariance matrix or shrinks the sample covariance matrix to some known structure. The second approach is based on Bayes' Theorem, which treats the covariance

matrix as a random variable with prior probability density function. By applying Bayes' theorem, the posterior density function of the covariance matrix can be found. The covariance matrix is then estimated by maximizing the posterior density function.

In this work, two methods for estimating the covariance matrix are proposed (especially for small sample size). The first is the Ridge Covariance Matrix Estimator (RCME), and the second is the Hierarchical Bayesian Covariance Matrix Estimator (HBCME). The covariance matrix estimation methods are then applied to improve the Recursive Least Squares (RLS) algorithm and mean-variance portfolio optimization.

The contributions of this work are as follows:

1) Proposed the RCME. The RCME can be viewed as one of the shrinkage estimators which shrinks the eigenvalues of the sample covariance matrix. The RCME is a weighted average of the sample covariance matrix and the identity matrix. The weight on the identity matrix is the shrinkage intensity, which is the only parameter that needs to be estimated. For sufficiently small shrinkage intensity, the covariance matrix estimated by the RCME is guaranteed to have smaller Mean Square Error (MSE) than the sample covariance matrix. Three methods for estimating the shrinkage intensity are proposed.
2) Proposed the HBCME. The HBCME is based on the Bayes' theorem. The HBCME applies Bayes's theorem hierarchically. Each element of the covariance matrix is assumed to be a random variable with normal distribution and unknown parameters. In the first level, Bayes' theorem is applied to find the posterior density function of the covariance matrix. The covariance matrix is then found by maximizing this posterior density function using the Broyden-

Fletcher-Goldfarb-Shanno (BFGS) optimization algorithm. In the second level, Bayes' theorem is applied to find the posterior density function for the unknown parameters. By maximizing this posterior density function, the values of the unknown parameters can be found.

3) Proposed the Shrinkage Least Squares algorithm (SLS) and the Shrinkage Recursive Least Squares (SRLS) algorithm. The SRLS algorithm is the recursive form of the SLS algorithm. SLS is an improved version of James-Stein Least Squares (JSLS), in which the covariance matrix is estimated using the RCME. SRLS is an improved version of James-Stein Recursive Least Squares (JSRLS), in which the covariance matrix is estimated using the RCME.

4) Proposed two methods for Portfolio Optimization: Shrinkage Portfolio Optimization and Bayesian Portfolio Optimization. In stock portfolio optimization, the covariance matrix and the mean of the stocks in a portfolio are parameters that require accurate estimation. The Shrinkage Portfolio Optimization method estimates the covariance matrix by the RCME. The Bayesian Portfolio Optimization method estimates the covariance matrix by the HBCME. The estimated covariance matrix is also used in the James-Stein estimator to estimate the mean. The improved estimates of the covariance matrix and the mean lead to improved portfolio optimization results.

This dissertation contains the following chapters: Chapter 2-5 are related to the RCME. Chapter 6-8 are related to the HBCME. Chapter 9 uses both the RCME and the HBCME. Chapter 10 concludes the current work.

Chapter 2 reviews various shrinkage covariance matrix estimators. There are generally two approaches for shrinkage estimators. One approach shrinks the eigenvalues of the sample covariance matrix, the other shrinks the sample covariance matrix to some known structure. Both approaches are reviewed in detail.

Chapter 3 describes the RCME. A detailed derivation of the estimator and interpretation of the results are presented. Three methods for estimating the shrinkage intensity - Ledoit's method, Filtering method and Constraint method, are also shown.

Chapter 4 presents the SLS and SRLS algorithm. First, James-Stein estimation is reviewed. Then the JSLS algorithm is reviewed and the SLS algorithm is proposed. Later the RLS and JSRLS algorithms are reviewed and the SRLS is proposed. Finally, the James-Stein Ledoit Recursive Least Squares (JSLRLS) is also proposed.

Chapter 5 presents two simulation results related to the RCME. The first is a Monte Carlo simulation for comparing different covariance matrix estimators. The second is a Monte Carlo simulation for comparing different RLS algorithms.

Chapter 6 reviews the Bayesian methods related to the covariance matrix estimation. Bayes' theorem is first presented, then existing Bayesian methods used for covariance matrix are reviewed. Finally, the Bayesian regularization method used in neural network training is reviewed, since this is the basis for developing the HBCME.

Chapter 7 presents the HBCME. The assumptions and the derivation of the estimator are given in detail. Two forms of the HBCME are proposed: The first form assumes the prior covariance matrix has two unknown parameters; The second form assumes the prior covariance matrix has $p + 1$ unknown parameters, where p is the dimension of the covariance matrix.

Chapter 8 contains the simulation results of the HBCME. Monte Carlo simulation results for both forms of the HBCME are presented.

Chapter 9 proposes methods for performing portfolio optimization. Background information on portfolio optimization is presented. Then two methods for performing portfolio optimization are proposed. Simulation results using both simulated stock return data and real stock return data are also presented.

Chapter 10 concludes the current work.

CHAPTER 2

SHRINKAGE COVARIANCE ESTIMATION

There are basically two classes of covariance estimation methods when the sample size is small. One class consists of the shrinkage methods, which shrink the eigenvalues of the sample covariance matrix, or shrink the sample covariance matrix to some structure. The other class is the Bayesian method, which applies Bayes' theorem to get the posterior mean of the covariance matrix by using prior information about the population covariance matrix. In this chapter, shrinkage methods will be reviewed. Bayesian methods will be discussed in Chapter 6.

This chapter has three sections. Section one briefly introduces different types of loss functions. Section two reviews methods for shrinking the eigenvalues of the sample covariance matrix. Section three presents methods for shrinking the sample covariance to some structures. The main focus is on the ridge regression [32] and the Ledoit Covariance Matrix Estimator (LCME) [37], because both methods are the inspiration for developing the Ridge Covariance Matrix Estimator (RCME) in Chapter 3.

1. Loss Functions

There are generally three types of loss functions found in the literature. The most commonly used is the entropy loss function first introduced by Stein [59] (see [63])

$$L_1(\hat{\mathbf{C}}, \mathbf{C}) = tr(\hat{\mathbf{C}}\mathbf{C}^{-1}) - \log\left|\hat{\mathbf{C}}\mathbf{C}^{-1}\right| - p, \tag{2-1}$$

where $\hat{\mathbf{C}}$ is the estimated covariance matrix, $\mathbf{C}$ is the true covariance matrix, and tr de-

notes the trace of a square matrix. The second loss function is quadratic:

$$L_2(\hat{\mathbf{C}}, \mathbf{C}) = tr(\hat{\mathbf{C}}\mathbf{C}^{-1} - \mathbf{I})^2, \tag{2-2}$$

where $\mathbf{I}$ is the identity matrix. The third loss function is also quadratic, but with the form

$$L_3(\hat{\mathbf{C}}, \mathbf{C}) = \|\hat{\mathbf{C}} - \mathbf{C}\|^2. \tag{2-3}$$

The norm $\|\cdot\|$ is defined as the Frobenius norm.

The corresponding risk functions are defined by

$$R_i(\hat{\mathbf{C}}, \mathbf{C}) = E[L_i(\hat{\mathbf{C}}, \mathbf{C})], \quad i = 1, 2, 3 \tag{2-4}$$

where E is mathematical expectation.

2. Shrinking the Eigenvalues

Although the sample covariance matrix is an unbiased estimate of the covariance matrix, it is known (for example, [4] [37]) that the eigenvalues of the sample covariance matrix $\mathbf{S}$ are more spread out than those of the true covariance matrix $\mathbf{C}$. This fact suggests that if the large eigenvalues of $\mathbf{S}$ are shrunk or the small eigenvalues are expanded, an improved covariance matrix estimator may result.

The idea of shrinking the eigenvalues was first proposed by James and Stein [34]. Based on the entropy loss shown in Eq. (2-1), a minimax estimator can be formulated as

$$\hat{\mathbf{C}} = \mathbf{A}\mathbf{D}\mathbf{A}^T, \tag{2-5}$$

where $\mathbf{A}$ is a lower triangular matrix with positive diagonal elements, and $\mathbf{A}$ satisfies $\mathbf{A}\mathbf{A}^T = (N-1)\mathbf{S}$. $\mathbf{D}$ is a diagonal matrix, $\mathbf{D} = diag(d_1, d_2, \ldots, d_p)$, where

$$d_i = \frac{1}{N+p-2i} \tag{2-6}$$

for $i = 1, 2, \ldots, p$. The risk, defined by Eq. (2-4) based on the entropy loss Eq. (2-1), is constant and uniformly smaller than that of $\mathbf{S}$. Olkin and Selliah [51] also proposed a minimax covariance estimator similar to the decomposition of Eq. (2-5).

Stein proposed another estimator with a decomposition structure similar to Eq. (2-5). Dey and Srinivasan [12] reported that Stein [60] [61] [62] considered the estimator

$$\hat{\mathbf{C}} = \mathbf{B}f(\mathbf{L})\mathbf{B}^T, \tag{2-7}$$

where $\mathbf{L} = diag(l_1, l_2, \ldots, l_p)$, l_i is an eigenvalue of $(N-1)\mathbf{S}$, $l_1 \geq l_2 \geq \ldots \geq l_p$. $\mathbf{B}$ contains the normalized eigenvectors of $\mathbf{S}$, and $f(\mathbf{L})$ is a diagonal matrix with

$$f(\mathbf{L}) = diag(f_1(\mathbf{L}), f_2(\mathbf{L}), \ldots, f_p(\mathbf{L})),$$

where

$$f_i(\mathbf{L}) = \frac{l_i}{N-p+2\sum_{j\neq i}\frac{l_i}{l_i-l_j}}, \; i = 1, 2, \ldots, p \tag{2-8}$$

(we also have the relationship $(N-1)\mathbf{S} = \mathbf{B}\mathbf{L}\mathbf{B}^T$). p in the denominator is to make the estimator approximately unbiased.

In this estimator, the large eigenvalues are shrunk and the small eigenvalues are expanded. For example, since $l_1 \geq l_2 \geq \ldots \geq l_p$, $l_1 - l_j$ is positive for all $j \neq 1$, it follows that $\sum_{j\neq i}\frac{l_1}{l_1-l_j}$ is positive, and $f_1(\mathbf{L}) < \frac{l_1}{N-p}$. The largest eigenvalue is shrunk. Similarly, since

$l_p - l_j$ is negative for all $j \neq p$, $\sum_{j \neq i} \frac{l_p}{l_p - l_j}$ is negative, and $f_p(\mathbf{L}) > \frac{l_p}{N-p}$. The smallest eigenvalue is then expanded. However, the relationship $f_1(\mathbf{L}) \geq f_2(\mathbf{L}) \geq \ldots \geq f_p(\mathbf{L})$ cannot be guaranteed.

A further modification of this estimator was developed to ensure the relation $f_1(\mathbf{L}) \geq f_2(\mathbf{L}) \geq \ldots \geq f_p(\mathbf{L})$. If the order is not preserved, it can be shown [58] that the estimator may not be admissible. Stein provided a method to ensure the order of the eigenvalues in his lectures [60] [61] [62]. Perron [53] also proposed a family of minimax estimators to ensure the order. Haff [29] proposed an estimator that uses constrained optimization and enforces the order relation by directly minimizing the entropy loss function, while Stein's method minimizes the entropy loss function in a heuristic way. However, it has not been proven that these estimators dominate $\mathbf{S}$, i.e., the risks of the estimators are less than or equal to the risk of $\mathbf{S}$, with the risks of the estimators are strictly less than that of $\mathbf{S}$ for at least one true covariance matrix $\mathbf{C}$. Nevertheless, simulation results [40] showed that Stein's ordered eigenvalue estimator not only dominates $\mathbf{S}$, but also outperforms the estimator given in Eq. (2-6) for a wide range of covariance matrices.

Dey and Srinivasan [12] [13] proposed an estimator which dominates $\mathbf{S}$ if $p > 2$. This estimator is in the form of Eq. (2-7), in which $f_i(\mathbf{L})$ is given by

$$f_i(\mathbf{L}) = \frac{l_i}{N-1} - \frac{(l_i \log l_i) g(u)}{d+u}, \tag{2-9}$$

where $u = \sum_{i=1}^{p} (\log l_i)^2$, d is a constant which satisfies $d > \frac{144(p-2)^2}{25(N-1)^2}$, and $g(u)$ is a

monotone non-decreasing function satisfying $0 < g(u) < \frac{12(p-2)}{5(N-1)^2}$ and $E\left[\frac{dg(u)}{du}\right] < \infty$.

This estimator shrinks or expands the eigenvalues toward or away from the origin. In fact, the origin can be replaced by a point determined by the positions of all the eigenvalues. An improved estimator that can adaptively find this point was also reported in [12]. The conditions are similar to those given above and will not be reproduced here.

Loh [42] [43] [44] extended Stein's method to the estimation of two covariance matrices simultaneously. Let $\mathbf{G}_1$ be a $p \times p$ matrix with Wishart distribution $W(\mathbf{C}_1, N_1 - 1)$, and $\mathbf{G}_2$ be a $p \times p$ matrix with Wishart distribution $W(\mathbf{C}_2, N_2 - 1)$, where $\mathbf{C}_1$ and $\mathbf{C}_2$ are the true covariance matrices for two different processes, and $\mathbf{G}_1$, $\mathbf{G}_2$ are independent. If prior information suggests that the eigenvalues of $\mathbf{C}_1\mathbf{C}_2^{-1}$ are close together, then Loh's estimator can get substantial savings in the loss function, defined as the sum of the entropy loss function of the two estimated covariance matrices. The idea of estimating two covariance matrices simultaneously was also proposed by Dey [14] almost at the same time. His method also estimated the eigenvalues of the covariance matrices simultaneously.

Recent developments in covariance matrix estimation by shrinking the eigenvalues have included the Bayesian method. Leonard and Hsu [39] assume some prior density function for the true covariance matrix $\mathbf{C}$. Yang and Berger [63] assume a reference prior for the parameter pair $(\mathbf{B}, \mathbf{L})$. Daniels and Kass [9] assume a prior distribution for the eigenvalues, and they concentrated on the estimation of the posterior mean of the eigenvalues.

An extension of the eigenvalue shrinkage method is to shrink the correlation structure of the sample covariance matrix. Instead of decomposing the sample covariance matrix

$\mathbf{S}$ to eigenvector-eigenvalue-eigenvector structure, decomposing $\mathbf{S}$ to variance-correlation-variance structure leads to the correlation shrinkage method with the form

$$\mathbf{S} = \mathbf{VPV}, \tag{2-10}$$

where $\mathbf{P}$ is the sample correlation matrix, $\mathbf{V}$ is a diagonal matrix with diagonal elements equal to the sample standard deviation of each variable, i.e., $\mathbf{V} = \sqrt{diag(\mathbf{S})}$. The idea was first proposed by Lin and Perlman [40]. They estimated $\mathbf{V}$ and $\mathbf{P}$ by the James-Stein type shrinkage method. Later, Daniels and Kass [8] [9] concentrated on the shrinkage of the correlation matrix $\mathbf{P}$. It was assumed that Fisher's z-transform [1] of the correlation is zero mean and normally distributed, thus forcing the off-diagonal elements of the covariance matrix toward zero. This is also equivalent to shrinking towards a diagonal matrix. This variance-correlation-variance decomposition method is not guaranteed to produce a positive definite matrix. Simulation results [9] suggested that if the prior structure is close to the true structure, the estimator did very well. When the structure is not close, it can do poorly on small samples.

3. Shrinking towards Structures

For small samples, the sample covariance matrix is not well conditioned. If some prior structural information about the true covariance matrix is available, and this structural information can be translated to a structural matrix, then we can shrink the sample covariance matrix to this structure matrix. The resulting covariance matrix estimates would be the weighted average of the sample covariance matrix and this structural matrix. Ideally, the resulting estimates would be well conditioned and still have all the asymptotic properties of the sample covariance matrix.

In this section, we review some of the choices of the structural matrix and show how it can be weight-averaged to the sample covariance matrix.

Shrinking toward Identity Matrix

If little or no information is available about the true covariance matrix, a flat prior can be assumed: all variances are the same and all covariances are zero. The resulting structure is a multiple of the identity matrix. This is the simplest structural assumption for the covariance matrix. The resulting covariance matrix is a weighted average of the sample covariance and the identity matrix.

Shrinking to the identity matrix can be traced back to ridge regression in the 1970's, which was proposed by Hoerl and Kennard [32] in the context of multiple linear regression. Ledoit formally shrinks the sample covariance matrix to the identity matrix in covariance matrix estimation. Some details can be found in the following two subsections.

Ridge Regression

Assume there are N sample observations, $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$. Each observation is a p dimensional vector. Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]^T$, $\mathbf{z}$ be the dependent variable, θ be the p dimensional unknown vector, and ε be the error term, then linear regression can be represented by

$$\mathbf{z} = \mathbf{X}\theta + \varepsilon. \tag{2-11}$$

Where $E[\varepsilon] = \mathbf{0}$, and $E[\varepsilon\varepsilon^T] = \sigma^2\mathbf{I}$. The Least Square (LS) estimate of θ is

$$\hat{\theta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{z}, \tag{2-12}$$

and the variance of $\hat{\theta}$ is $\sigma^2(\mathbf{X}^T\mathbf{X})^{-1}$. If the sample size is small, then $\mathbf{X}^T\mathbf{X}$ may not be well conditioned, and variance of $\hat{\theta}$ could be very large. The solution is to add a multiple of identity matrix, $\upsilon\mathbf{I}$, to $\mathbf{X}^T\mathbf{X}$, i.e.,

$$\hat{\theta} = (\mathbf{X}^T\mathbf{X} + \upsilon\mathbf{I})^{-1}\mathbf{X}^T\mathbf{z}, \tag{2-13}$$

where υ is the ridge perturbation factor. The resulting $\hat{\theta}$ is a biased estimate but with smaller Mean Square Error (MSE).

Several methods have been proposed for estimating υ during the past 30 years. Gruber [24] has a very good summary of the different methods.

How is the ridge regression related to covariance matrix estimation? If the sample observations, $\mathbf{x}_1$, $\mathbf{x}_2$, ..., $\mathbf{x}_N$, are zero mean, or the mean is removed from these observations, the sample covariance matrix can be expressed as

$$\mathbf{S} = \frac{1}{N-1}\mathbf{X}^T\mathbf{X}. \tag{2-14}$$

For the ridge regression, we add $\upsilon\mathbf{I}$ to $\mathbf{X}^T\mathbf{X}$, resulting in a better parameter estimation in terms of MSE. In a similar way, it is possible to get a better covariance matrix estimation method. In Chapter 3, we propose a new ridge covariance estimation method.

Ledoit Covariance Matrix Estimator (LCME)

Ledoit [37] formally proposed the idea of shrinking the sample covariance matrix to the identity matrix. The estimator is the optimal linear combination of the sample covariance matrix and the identity matrix under the quadratic loss Eq. (2-3). This estimator can be used even when the dimension of the matrix p is smaller than the sample size N. How-

ever, simulation studies [9] show it can over-shrink, resulting in poor performance. The details of the algorithm follow.

Using the quadratic loss function of Eq. (2-3), the minimization of the corresponding risk function can be formalized as follows,

$$\begin{aligned} &\min_{k_1, k_2} J = E[\|\mathbf{C}_L - \mathbf{C}\|^2] \\ &\text{subject to } \mathbf{C}_L = k_1\mathbf{I} + k_2\mathbf{S} \end{aligned} \tag{2-15}$$

where the coefficients k_1 and k_2 are deterministic, and the subscript L represents "Ledoit". The solution to Eq. (2-15) is

$$\mathbf{C}_L = \frac{\beta^2}{\delta^2}\mu\mathbf{I} + \frac{\alpha^2}{\delta^2}\mathbf{S}, \tag{2-16}$$

where μ, α^2, β^2, and δ^2 are defined as follows,

$$\mu = \mathbf{C} \mathrm{o} \mathbf{I} \tag{2-17}$$

$$\alpha^2 = \|\mathbf{C} - \mu\mathbf{I}\|^2 \tag{2-18}$$

$$\beta^2 = E[\|\mathbf{S} - \mathbf{C}\|^2] \tag{2-19}$$

$$\delta^2 = E[\|\mathbf{S} - \mu\mathbf{I}\|^2] \tag{2-20}$$

The operator o is a matrix operation, defined as $\mathbf{A}\mathrm{o}\mathbf{B} \equiv \frac{\mathrm{tr}(\mathbf{A}\mathbf{B}^T)}{p}$, for $p \times p$ matrices $\mathbf{A}$ and $\mathbf{B}$.

The minimum objective function J is given by

$$J = \frac{\alpha^2\beta^2}{\delta^2} \tag{2-21}$$

The value $\frac{\beta^2}{\delta^2}$ is the shrinkage intensity, which asymptotically approaches zero as the number of samples increases. This desirable, since as more samples are available, the covariance matrix can be estimated more accurately, dependence on prior information is reduced, and the shrinkage intensity declines accordingly.

The above solution can not be used in practice, since the parameters μ, α^2, β^2 and δ^2 are all computed from the unobservable $\mathbf{C}$. The following method provides consistent estimates for these parameters:

$$m = \hat{\mu} = \frac{\mathrm{tr}((\mathbf{X}^T\mathbf{X})/N)}{p} \tag{2-22}$$

$$d^2 = \hat{\delta}^2 = \|\mathbf{S} - \hat{\mu}\mathbf{I}\|^2 \tag{2-23}$$

$$\bar{b}^2 = \sum_{i=1}^{N} \|x_i x_i^T - \mathbf{S}\| \tag{2-24}$$

$$b^2 = \hat{\beta}^2 = \min\left(\frac{1}{N^2}\bar{b}^2, d^2\right) \tag{2-25}$$

$$a^2 = \hat{\alpha}^2 = d^2 - b^2 \tag{2-26}$$

Using these parameters, a consistent estimate of the covariance matrix is

$$\hat{\mathbf{C}}_L = \frac{b^2}{d^2}m\mathbf{I} + \frac{a^2}{d^2}\mathbf{S} \tag{2-27}$$

Ledoit proved that, asymptotically, $\hat{\mathbf{C}}_L$ has uniformly minimum quadratic risk among all the linear combinations of the identity $\mathbf{I}$ with the sample covariance $\mathbf{S}$.

Neither ridge regression nor the LCME assumes any statistical distribution for any of the variables. However, there are some methods that do assume a normal distribution for

the observation data. Efron and Morris [17], Haff [25] [26] [27] estimated the inverse of the covariance matrix instead of the covariance matrix itself. All of the estimators have the following form,

$$\hat{\mathbf{C}}^{-1} = [c + f(\mathbf{S})]\mathbf{S}^{-1} + g(\mathbf{S})\mathbf{I}, \tag{2-28}$$

where c is a non-negative constant, $f(\mathbf{S})$ and $g(\mathbf{S})$ are real functions of $\mathbf{S}$. All of these types of estimators are minimax estimators which dominate the maximum likelihood estimator of the covariance matrix. However, if the sample size is small, $\mathbf{S}$ may not be invertible or the inversion may induce large errors. Therefore, this method is only good for large samples.

Instead of shrinking toward the identity matrix, any other matrix can be used, if proper justification can be given. Ledoit [38] suggested a structure derived from the Capital Asset Pricing model [23]. The simulation results for portfolio optimization were reported to be successful.

Chen [5] proposed shrinking toward the Wishart prior with an unknown number of degrees of freedom. It was assumed that the prior distribution of the true covariance matrix is of the Wishart form

$$\mathbf{C} \sim W((\nu\mathbf{G})^{-1}, \nu), \tag{2-29}$$

where $\mathbf{G}$ is the prior mean of the true covariance matrix, and ν is the degree of freedom of $\mathbf{G}$. The estimator has following form,

$$\hat{\mathbf{C}} = \frac{N-1}{N+\nu^{*}-1}\mathbf{S} + \frac{\nu^{*}}{N+\nu^{*}-1}\mathbf{G}^{*}, \tag{2-30}$$

The unknown hyperparameters $\mathbf{G}^{*}$ and ν^{*} in the Wishart distribution were estimated by

the EM algorithm [10] through iteration. However, the degree of freedom has a lower bound of p, and it is the only adjustable parameter in this formulation. Simulation results [8] indicate that using the Wishart prior is not a good choice. The estimation error can be very large in some cases.

Based on the ridge regression [32] and the LCME [37], we have developed the ridge covariance matrix estimator (RCME) and it will be discussed in detail in the next chapter.

CHAPTER 3

RIDGE COVARIANCE MATRIX ESTIMATOR

In order to improve the covariance estimate when the sample size is small, one possible solution is to shrink the sample covariance matrix to some structure. A Ridge Covariance Matrix Estimator (RCME) is proposed in this chapter, which shrinks the sample covariance matrix to the identity matrix based on a quadratic loss function. The resulting estimate is biased and is a weighted average of the sample covariance matrix and the identity matrix.

This chapter has two sections. Section one is the derivation and interpretation of the RCME. Section two suggests some methods for the estimation of the shrinkage intensity parameter.

1. Derivation and Interpretation

The RCME is inspired by ridge regression [32] and the Ledoit Covariance Matrix Estimator (LCME) [37]. Ridge regression adds a multiple of the identity matrix to the sample covariance matrix (See Chapter 2). The LCME tries to find the optimal weighted-average of the sample covariance matrix and the identity matrix, using two parameters. The RCME is also a weighted average of the sample covariance matrix and the identity matrix, but it uses only one parameter, as does ridge regression.

The problem of shrinking the estimate to the identity matrix is stated as follows,

$$\min_{k} \; J = E[\left\| \hat{\mathbf{C}}_R - \mathbf{C} \right\|^2] \tag{3-1}$$

subject to

$$\hat{\mathbf{C}}_R = k\mathbf{I} + (1-k)\mathbf{S} \tag{3-2}$$

where the subscript R stands for "Ridge", and k is the shrinkage intensity. J is the risk function of the estimate $\hat{\mathbf{C}}_R$ based on the quadratic loss, as in Eq. (2-3).

From elementary statistics, we can show that

$$J = \left\| E[\hat{\mathbf{C}}_R] - \mathbf{C} \right\|^2 + E[\left\| \hat{\mathbf{C}}_R - E[\hat{\mathbf{C}}_R] \right\|^2] \tag{3-3}$$

where the first term on the right hand side of the equation is the squared bias ($bias^2$) and the second term is the variance var. The bias term is

$$\begin{aligned} bias^2 &= \left\| E[\hat{\mathbf{C}}_R] - \mathbf{C} \right\|^2 \\ &= \| E[k\mathbf{I} + (1-k)\mathbf{S}] - \mathbf{C} \|^2 \\ &= \| k(\mathbf{I} - \mathbf{C}) \|^2 \\ &= k^2 \| \mathbf{C} - \mathbf{I} \|^2 \\ &= k^2 \gamma \end{aligned} \tag{3-4}$$

where $\gamma = \| \mathbf{C} - \mathbf{I} \|^2$. The variance terms can be decomposed to

$$\begin{aligned} var &= E[\left\| \hat{\mathbf{C}}_R - E[\hat{\mathbf{C}}_R] \right\|^2] \\ &= E[\| (k\mathbf{I} + (1-k)\mathbf{S}) - E[k\mathbf{I} + (1-k)\mathbf{S}] \|^2] \\ &= E[\| (1-k)(\mathbf{S} - \mathbf{C}) \|^2] \\ &= (1-k)^2 E[\| \mathbf{S} - \mathbf{C} \|^2] \\ &= (1-k)^2 \eta \end{aligned} \tag{3-5}$$

where $\eta = E[\| \mathbf{S} - \mathbf{C} \|^2]$.

From Eq. (3-3), (3-4), and (3-5),

$$J = k^2\gamma + (1-k)^2\eta . \tag{3-6}$$

Take the derivative of J with respect to k, set the results to 0, and solve the equation for k:

$$k_o = \frac{\eta}{\gamma + \eta} = \frac{E[\|\mathbf{S} - \mathbf{C}\|^2]}{\|\mathbf{C} - \mathbf{I}\|^2 + E[\|\mathbf{S} - \mathbf{C}\|^2]}. \tag{3-7}$$

Here the subscript o indicates the optimal value.

Notice that

$$\lim_{k \to 0^+} \frac{\partial}{\partial k} bias^2 = \lim_{k \to 0^+} 2k\gamma = 0, \tag{3-8}$$

and

$$\lim_{k \to 0^+} \frac{\partial var}{\partial k} = \lim_{k \to 0^+} (-2)(1-k)\eta = -2\eta < 0. \tag{3-9}$$

Since $\hat{\mathbf{C}}_R = \mathbf{S}$ for $k = 0$, and the derivative of J with respect to k at $k = 0$ is negative, there exists some k, $0 < k < 1$, such that the MSE of $\hat{\mathbf{C}}_R$ is smaller than that of $\mathbf{S}$.

Figure 3-1 illustrates $bias^2$ and var as functions of k. Since $bias^2(k)$ is monotonically increasing with zero slope at $k = 0$, and $var(k)$ is monotonically decreasing with negative slope at $k = 0$, there must exist some $k > 0$ where the MSE reaches a minimum. Figure 3-1 is plotted for the special case where $\gamma = 5$ and $\eta = 2$.

The RCME has several useful properties. First of all, since $\mathbf{S}$ is a consistent estimator of $\mathbf{C}$, when more samples are available, there should be less shrinkage toward $\mathbf{I}$. In other words, k_o should approach zero as the number of samples approaches infinity. From the definition of η, we can see that η approaches zero as the number of samples approaches infinity. It follows from Eq. (3-7), that if γ is a non zero constant, then k_o approaches zero. If γ is zero, which means the population covariance matrix is the identity matrix, then $k_o = 1$, and $\hat{\mathbf{C}}_R = \mathbf{I}$, a trivial case of covariance matrix estimation.

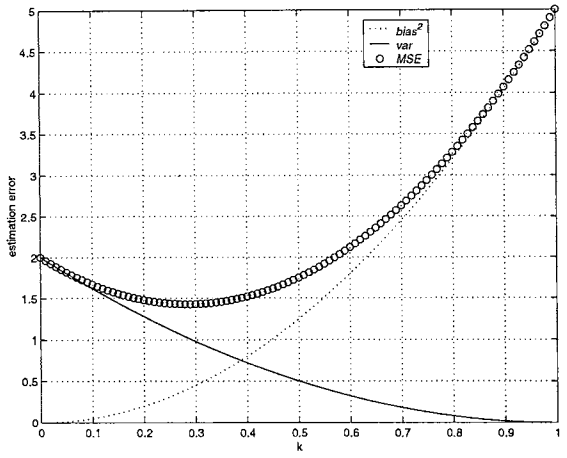


Figure 3-1 Errors of Ridge Covariance Matrix Estimator

Secondly, instead of shrinking toward the identity matrix, we could use other matrices if proper justification can be made. Let the prior mean for $\mathbf{C}$ be the matrix $\mathbf{F}$, then γ can be redefined to equal $\|\mathbf{C} - \mathbf{F}\|^2$. The closer the true covariance matrix $\mathbf{C}$ is to the prior mean matrix $\mathbf{F}$, the larger k_o will be and the more shrinkage can be made toward $\mathbf{F}$.

Third, the RCME can be viewed as a method of shrinking or expanding the eigenvalues of the sample covariance matrix toward 1. In fact, the sample covariance matrix can be decomposed as

$$\mathbf{S} = \mathbf{B}\mathbf{Q}\mathbf{B}^T, \tag{3-10}$$

where $\mathbf{Q} = diag(q_1, q_2, \ldots, q_p)$, q_i is an eigenvalue of $\mathbf{S}$, and $q_1 \geq q_2 \geq \ldots \geq q_p$. The columns of $\mathbf{B}$ are the normalized eigenvectors of $\mathbf{S}$. Comparing with Eq. (2-7) in Chapter 2, we can see that the relation $\mathbf{Q} = \dfrac{\mathbf{L}}{N-1}$ holds. For the RCME in the form of Eq. (3-2),

$$\begin{aligned}\hat{\mathbf{C}}_R &= k\mathbf{I} + (1-k)\mathbf{S} \\ &= \mathbf{B}[k\mathbf{I}]\mathbf{B}^T + \mathbf{B}[(1-k)\mathbf{Q}]\mathbf{B}^T \cdot \\ &= \mathbf{B}[k\mathbf{I} + (1-k)\mathbf{Q}]\mathbf{B}^T\end{aligned} \tag{3-11}$$

If $\hat{\mathbf{C}}_R$ is rearranged in the form of

$$\hat{\mathbf{C}}_R = \mathbf{B}f(\mathbf{Q})\mathbf{B}^T, \tag{3-12}$$

with $f(\mathbf{Q}) = diag(f_1(\mathbf{Q}), f_2(\mathbf{Q}), \ldots, f_p(\mathbf{Q}))$, then we can show that

$$f_i(\mathbf{Q}) = q_i + k(1-q_i),\ i = 1, 2, \ldots, p \tag{3-13}$$

For $0 < k < 1$, if $q_i > 1$, then $f_i(\mathbf{Q}) < q_i$, and therefore the eigenvalues of the covariance matrix estimated by the RCME shrink toward 1. If $q_i < 1$, then $f_i(\mathbf{Q}) > q_i$, and the eigenvalues of the covariance matrix estimated by the RCME expand toward 1. Therefore, the RCME shrinks or expands the eigenvalues of the sample covariance matrix toward 1.

Fourth, the condition number of $\hat{\mathbf{C}}_R$ is smaller than that of $\mathbf{S}$. The condition number of a covariance matrix can be defined as the ratio of the maximum eigenvalue to the minimum eigenvalue. From Eq. (3-12), the condition number of $\hat{\mathbf{C}}_R$ is $\frac{q_1 + k(1-q_1)}{q_p + k(1-q_p)}$. From Eq. (3-10), the condition number of $\mathbf{S}$ is $\frac{q_1}{q_p}$. It is not difficult to prove the following:

$$\frac{q_1 + k(1-q_1)}{q_p + k(1-q_p)} > \frac{q_1}{q_p}. \tag{3-14}$$

Lastly, the RCME preserves the order of the eigenvalues. From Eq. (3-13), if $i < j$ (which means that $q_i < q_j$), then $f_i(\mathbf{Q}) < f_j(\mathbf{Q})$. Therefore, no additional step is needed to

maintain the order of the eigenvalues, as was required for Stein [60] [61] [62], Perron [53], and Half [29].

From previous discussion, we can see that $\hat{\mathbf{C}}_R$ is a biased estimate of the true covariance matrix, but the MSE of $\hat{\mathbf{C}}_R$ is smaller than that of $\mathbf{S}$. This is not an uncommon result. In fact, the mean estimated by the James-Stein estimator, which will be discussed in Chapter 4, is also biased, but its MSE is smaller than that of the sample mean. This is the famous Stein's Paradox. For detailed discussion, see [16] [18].

Compared to the LCME, the RCME has two advantages. First, the covariance matrix estimated by the RCME is guaranteed to have smaller MSE than the sample covariance matrix, provided that the parameter k is small enough. The LCME cannot make this guarantee. Secondly, the parameter estimation in the RCME is easier. There is only one parameter to be estimated in the RCME, while in LCME, two parameters are needed.

2. Estimating Optimal Shrinkage Intensity

Eq. (3-7) gives the optimal value for k. However, since Eq. (3-7) involves the unknown true covariance matrix $\mathbf{C}$, k_o cannot be computed in practice. In this section we propose three methods for estimating k. The first is Ledoit's method, which is based on the asymptotic convergence theorems proposed by Ledoit [37]. The second is the filtering method, which filters the estimate of k from Ledoit's method. The third is the constraint method, which is an extension of the Ledoit's method that limits the maximum allowable value of k.

Ledoit's method

The first method for estimating the shrinkage parameter k is based on Ledoit's method [37] of covariance matrix estimation. Based on Ledoit's theorem, we can prove the following lemmas:

Lemma 1: $\|\mathbf{S}-\mathbf{I}\|^2$ converges to $E[\|\mathbf{S}-\mathbf{I}\|^2]$ in a mean-squared sense, i.e.,

$$\lim_{N\to\infty} Var[\|\mathbf{S}-\mathbf{I}\|^2] = 0, \tag{3-15}$$

where Var is the variance and N is the number of sample points.

Proof: Refer to the proof of Theorem 2.4 in Ledoit's paper [37]. Omitted here. □

Lemma 2: $\frac{1}{N^2}\sum_{i=1}^{N}\left\|\mathbf{x}_i\mathbf{x}_i^T-\mathbf{S}\right\|^2$ converges to $E[\|\mathbf{S}-\mathbf{C}\|^2]$ in a mean-squared sense, i.e.,

$$E\left[\frac{1}{N^2}\sum_{i=1}^{N}\left\|\mathbf{x}_i\mathbf{x}_i^T-\mathbf{S}\right\|^2\right] = E[\|\mathbf{S}-\mathbf{C}\|^2], \tag{3-16}$$

$$\text{and } \lim_{N\to\infty} Var\left[\frac{1}{N^2}\sum_{i=1}^{N}\left\|\mathbf{x}_i\mathbf{x}_i^T-\mathbf{S}\right\|^2\right] = 0. \tag{3-17}$$

Proof: Refer to the proof of Theorem 2.5 in Ledoit's paper [37]. Omitted here. □

Based on Lemma 1 and Lemma 2, the estimation of k_o can be given by $\hat{k}_o^L$, which is defined by

$$\hat{k}_o^L = min\left(\frac{\frac{1}{N^2}\sum_{i=1}^{N}\left\|\mathbf{x}_i\mathbf{x}_i^T - \mathbf{S}\right\|^2}{\|\mathbf{S}-\mathbf{I}\|^2}, 1\right), \tag{3-18}$$

where the subscript L indicates Ledoit's estimate.

This Ledoit estimate of k_o can be derived as follows. First, recall the formula for k_o:

$$k_o = \frac{\eta}{\gamma+\eta} = \frac{E[\|\mathbf{S}-\mathbf{C}\|^2]}{\|\mathbf{C}-\mathbf{I}\|^2 + E[\|\mathbf{S}-\mathbf{C}\|^2]}. \tag{3-19}$$

We will first find an estimate for the denominator of k_o, then we will estimate the numerator. For the denominator, we can show

$$\begin{aligned} E[\|\mathbf{S}-\mathbf{I}\|^2] &= E[\|\mathbf{S}-\mathbf{C}+\mathbf{C}-\mathbf{I}\|^2] \\ &= E[\|\mathbf{S}-\mathbf{C}\|^2 + 2(\mathbf{S}-\mathbf{C})o(\mathbf{C}-\mathbf{I}) + \|\mathbf{C}-\mathbf{I}\|^2] \\ &= E[\|\mathbf{S}-\mathbf{C}\|^2] + \|\mathbf{C}-\mathbf{I}\|^2 \\ &= (\gamma+\eta) \end{aligned} \tag{3-20}$$

From Lemma 1, $E[\|\mathbf{S}-\mathbf{I}\|^2]$ can be estimated by $\|\mathbf{S}-\mathbf{I}\|^2$, therefore this will be our estimate of the denominator of k_o.

Now we want to estimate the numerator of k_o, $\eta = E[\|\mathbf{S}-\mathbf{C}\|^2]$. From Lemma 2, we can use $\frac{1}{N^2}\sum_{i=1}^{N}\left\|\mathbf{x}_i\mathbf{x}_i^T - \mathbf{S}\right\|^2$.

To complete the estimate of k_o, we notice that in some cases, especially when the sample size N is small, it is possible that $\frac{1}{T^2}\sum_{i=1}^{T}\|\mathbf{x}_i\mathbf{x}_i^T-\mathbf{S}\|^2 > \|\mathbf{S}-\mathbf{I}\|^2$. However, k should be always less than 1. Therefore, $\hat{k}_o^L$ is taken to be the minimum of $\frac{\frac{1}{N^2}\sum_{i=1}^{N}\|\mathbf{x}_i\mathbf{x}_i^T-\mathbf{S}\|^2}{\|\mathbf{S}-\mathbf{I}\|^2}$ and 1.

In practice, $\hat{k}_o^L$ approaches its true value k_o only as the sample size goes to infinity. However, the objective of the ridge covariance estimation is to improve the covariance estimation for small samples. Therefore, Ledoit's estimate of k_o is only of theoretical value.

Filtering method

Ledoit's estimate $\hat{k}_o^L$ can fluctuate with relatively large amplitude, especially when the sample size is small. This is because in Eq. (3-18), both the denominator $\|\mathbf{S}-\mathbf{I}\|^2$ and the numerator $\frac{1}{N^2}\sum_{i=1}^{N}\|\mathbf{x}_i\mathbf{x}_i^T-\mathbf{S}\|^2$ can have large variances for small sample sizes. This fact has been demonstrated through extensive simulations. On the other hand, k_o is a relatively stable value. k_o only becomes close to zero when the sample size N gets large. Therefore, a filtering method is proposed to reduce the fluctuation in Ledoit's estimate of k_o.

We will indicate the filtered estimate of k_o at time t by $\hat{k}_o^F(t)$. At time $t+1$, one more sample is available. Based on all the available samples, Ledoit's estimate $\hat{k}_o^L$ can be computed. The updated filtered estimate $\hat{k}_o^F(t+1)$ can then be computed as the weighted average of $\hat{k}_o^F(t)$ and $\hat{k}_o^L$,

$$\hat{k}_o^F(t+1) = \omega\hat{k}_o^F(t) + (1-\omega)\hat{k}_o^L, \tag{3-21}$$

where ω is the forgetting factor, $0 \le \omega \le 1$. Larger ω will produce less fluctuations in $\hat{k}_o^F$, while smaller ω produces more fluctuation. If $\omega = 0$, $\hat{k}_o^F = \hat{k}_o^L$, and filtering method is equivalent to Ledoit's estimate.

Later in Chapter 5, the filtering method will be demonstrated on the stock portfolio optimization problem.

Constraint method

The derivation of the RCME indicates that if k is small enough, it is guaranteed that the MSE of $\hat{\mathbf{C}}_R$ is smaller than that of $\mathbf{S}$, because the derivative of J with respect to k at $k = 0$ is negative. Therefore, if we limit the maximum k to some small positive value less than 1, instead of 1 as in Eq. (3-18), we can obtain improved estimates.

Later in Chapter 5, a constraint of 0.02 will be used in the shrinkage recursive least squares problem.

In practice, if the variables differ significantly in magnitude, it is best to normalize the data. In this case, we are actually estimating the correlation matrix.

In Chapter 4, we will apply RCME to the least squares algorithm (both in the batch form and the recursive form). In chapter 5, we will provide simulation results for comparing the RCME with other covariance matrix estimators. As an application, the RCME will also be used in solving the stock portfolio optimization problem.

CHAPTER 4

SHRINKAGE LEAST SQUARES AND SHRINKAGE RECURSIVE LEAST SQUARES

In this chapter, we apply the Ridge Covariance Matrix Estimator (RCME) developed in Chapter 3 to the Least Squares (LS) algorithm, both in the batch form and the recursive form. Our development of the two algorithms are based on the James-Stein Least Squares (JSLS) algorithms, both in the batch form and the recursive form, proposed by Manton et.al. [46]. The JSLS algorithms improve the least squares estimate of the parameter by estimating the parameter mean using the James-Stein estimator [34]. We propose an improvement to the algorithm by applying the RCME. The covariance matrix estimated by the RCME is used to improve the estimate of the parameter mean, which is estimated by the James-Stein estimator. Since both the mean and the covariance matrix are estimated using shrinkage, in the sequel, we will call the resulting batch estimator Shrinkage Least Squares (SLS), and the recursive estimator Shrinkage Recursive Least Squares (SRLS), respectively.

This chapter has three sections. Section one briefly reviews the James-Stein Estimator. Section two presents the SLS algorithm. In this section, the JSLS algorithm is first reviewed and some proofs omitted in [46] are bridged. Then the SLS is developed. Section three presents the SRLS algorithm and the James-Stein Ledoit Recursive Least Squares (JSLRLS) algorithm. In this section, standard Recursive Least Squares (RLS) is first reviewed, then the James-Stein Recursive Least Squares (JSRLS) algorithm [46] is present-

ed. Later, we develop the SRLS algorithm, which is the shrinkage version of the RLS algorithm and the recursive form of the SLS. Finally, we present the JSLRLS algorithm, which is proposed mainly for comparing to the SRLS algorithm.

1. James-Stein Estimation (JS)

Let $\mathbf{y}$ be a $p \times 1$ random vector with $p > 2$. $\mathbf{y}$ is normally distributed with mean μ and covariance matrix $\mathbf{I}$, i.e., $\mathbf{y} \sim N(\mu, \mathbf{I})$. Given the single observation $\mathbf{y}$ (we abuse the notation here by letting $\mathbf{y}$ represent both the random vector and an observation of this random vector), the James-Stein (JS) estimate of the mean [15] is

$$\hat{\mu}_{\mathbf{y}}^{JS} = \left(1 - \frac{p-2}{\mathbf{y}^T\mathbf{y}}\right)\mathbf{y} \tag{4-1}$$

where the superscript JS stands for James-Stein and the subscript $\mathbf{y}$ denotes the random variable. In the sequel, if it is clear from the context, we may omit the subscript.

It has been proved that the risk (MSE, defined as $\frac{E[(\hat{\mu}-\mu)^T(\hat{\mu}-\mu)]}{p}$) of the James-Stein estimator is smaller than the risk of the mean estimated by Maximum Likelihood Estimation (MLE). However, the mean estimated by the James-Stein estimator is biased.

If the covariance matrix of $\mathbf{y}$ is $\mathbf{C}$, instead of $\mathbf{I}$, i.e., $\mathbf{y} \sim N(\mu, \mathbf{C})$, we can make the transformation $\mathbf{y}^\circ = \mathbf{C}^{-\frac{1}{2}}\mathbf{y}$. Since $E[\mathbf{y}^\circ] = \mathbf{C}^{-\frac{1}{2}}\mu$ and $Var[\mathbf{y}^\circ] = \mathbf{I}$, it follows $\mathbf{y}^\circ \sim N(\mu^\circ, \mathbf{I})$, where $\mu^\circ = \mathbf{C}^{-\frac{1}{2}}\mu$. Apply Eq. (4-1) to $\mathbf{y}^\circ$,

$$\hat{\mu}_{\mathbf{y}^\circ}^{JS} = \left(1 - \frac{p-2}{\mathbf{y}^{\circ T}\mathbf{y}^\circ}\right)\mathbf{y}^\circ, \tag{4-2}$$

where $\hat{\mu}_{\mathbf{y}^\circ}^{JS}$ is the James-Stein estimate of the mean of $\mathbf{y}^\circ$. Substitute $\mathbf{y}^\circ = \mathbf{C}^{-\frac{1}{2}}\mathbf{y}$ into Eq. (4-2) and notice that $\hat{\mu}_{\mathbf{y}}^{JS} = \mathbf{C}^{\frac{1}{2}}\hat{\mu}_{\mathbf{y}^\circ}^{JS}$. The resulting James-Stein estimator would be

$$\hat{\mu}_{\mathbf{y}}^{JS} = \left(1 - \frac{p-2}{\mathbf{y}^T\mathbf{C}^{-1}\mathbf{y}}\right)\mathbf{y}. \tag{4-3}$$

There have been many improvements to James-Stein estimation. One of the improvements is from [22]. Define $(\cdot)^+ = \max(0, \cdot)$, denote the maximum eigenvalue of the covariance matrix $\mathbf{C}$ as $\lambda_{max}(\mathbf{C})$, and define p^e to be the effective dimension of $\mathbf{y}$, $p^e = \frac{\mathrm{tr}(\mathbf{C})}{\lambda_{max}(\mathbf{C})}$. The James-Stein estimate of the mean can then be written as

$$\hat{\mu}_{\mathbf{y}}^{JS} = \left(1 - \frac{(min\{(p-2), 2(p^e-2)\})^+}{\mathbf{y}^T\mathbf{C}^{-1}\mathbf{y}}\right)^+\mathbf{y}, \tag{4-4}$$

where $(min\{(p-2), 2(p^e-2)\})^+$ replaces $p-2$ in Eq. (4-3). The inside $(\cdot)^+$ ensures that if $p \leq 2$, no shrinkage is given. The outside $(\cdot)^+$ ensures that the shrinkage is always in the same direction as $\mathbf{y}$.

The mean estimated by the James-Stein estimator is biased. However, its MSE is smaller than that of the sample mean. This is the famous Stein's Paradox. In fact, according to Efron and Morris [18], "The rationale of the method is to reduce the overall risk by assuming that the true means are more similar to one another than the observed data."

2. Shrinkage Least Squares (SLS)

In this section, first the linear model for the least squares estimate is given, then the James-Stein least squares algorithm developed by Manton et.al. [46] is presented. Finally, we develop the shrinkage least squares algorithm.

The Model

We consider the following linear model,

$$\mathbf{z} = \mathbf{R}\mathbf{y} + \mathbf{T}\mathbf{w}, \tag{4-5}$$

where $\mathbf{y}$ is the parameter vector with dimension $p \times 1$, $\mathbf{z}$ is the measurement vector with dimension $N \times 1$, $\mathbf{w}$ is the measurement noise vector with dimension $N \times 1$, $\mathbf{R}$ is the observation matrix with dimension $N \times p$, and $\mathbf{T}$ is the measurement noise matrix with dimension $N \times N$. Both $\mathbf{R}$ and $\mathbf{T}$ are known matrices with rank equal to the number of columns. It is assumed that the elements of $\mathbf{w}$, $w(1)$, $w(2)$, ..., $w(N)$, are independent and identically distributed (i.i.d.) with Gaussian distribution. It follows that $\mathbf{w} \sim N(0, \sigma^2\mathbf{I})$. We assume that σ^2 is unknown and $n > p$. Therefore, it is possible that σ^2 can be estimated from observation data.

James-Stein Least Squares (JSLS)

Manton et.al. [46] applied James-Stein's estimator [34] to the least squares regression problem. The strategy is to obtain the maximum likelihood estimate of the parameter vector $\mathbf{y}$, and then apply James-Stein's estimator to reduce the risk of the maximum likelihood estimate.

First we derive the maximum likelihood estimate of $\mathbf{y}$. The derivation was omitted in [46], therefore we bridge the gap here. From the distribution of $\mathbf{w}$, we get the distribution of $\mathbf{Tw}$ and $\mathbf{z}$

$$\mathbf{Tw} \sim N(\mathbf{0}, \sigma^2\mathbf{O}^{-1}) \tag{4-6}$$

$$\mathbf{z} \sim N(\mathbf{Ry}, \sigma^2\mathbf{O}^{-1}) \tag{4-7}$$

where $\mathbf{O} = (\mathbf{TT}^T)^{-1}$. The likelihood function $l(\mathbf{y})$ is

$$\begin{aligned} l(\mathbf{y}) &= f(\mathbf{z}|\mathbf{y}) \\ &= \frac{1}{(2\pi)^{k/2}|\sigma^2\mathbf{O}^{-1}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{z}-\mathbf{Ry})^T\sigma^{-2}\mathbf{O}(\mathbf{z}-\mathbf{Ry})\right\}. \end{aligned} \tag{4-8}$$

Take the log of $l(\mathbf{y})$:

$$\log l(\mathbf{y}) = -\frac{k}{2}\log(2\pi) - \frac{1}{2}\log(|\sigma^2\mathbf{O}^{-1}|) - \frac{1}{2}(\mathbf{z}-\mathbf{Ry})^T\sigma^{-2}\mathbf{O}(\mathbf{z}-\mathbf{Ry}). \tag{4-9}$$

Take the derivative of $\log l(\mathbf{y})$ and set it to zero:

$$\begin{aligned} \frac{\partial}{\partial \mathbf{y}}\log l(\mathbf{y}) &= \frac{\partial}{\partial \mathbf{y}}\left\{-\frac{1}{2}\sigma^{-2}(\mathbf{z}^T\mathbf{Oz} - 2\mathbf{y}^T\mathbf{R}^T\mathbf{Oz} + \mathbf{y}^T\mathbf{R}^T\mathbf{ORy})\right\} \\ &= -\frac{\sigma^{-2}}{2}(-2\mathbf{R}^T\mathbf{Oz} + 2\mathbf{R}^T\mathbf{ORy}) \\ &= \mathbf{0} \end{aligned} \tag{4-10}$$

Therefore

$$\hat{\mathbf{y}}^{ML} = (\mathbf{R}^T\mathbf{OR})^{-1}\mathbf{R}^T\mathbf{Oz}, \tag{4-11}$$

where the superscript ML denotes Maximum Likelihood. Let $\mathbf{A}^\circ = (\mathbf{R}^T\mathbf{OR})^{-1}\mathbf{R}^T\mathbf{O}$, and combine with Eq. (4-7)

$$\hat{\mathbf{y}}^{ML} = \mathbf{A}^\circ\mathbf{z} \sim N(\mathbf{A}^\circ\mathbf{Ry}, \sigma^2\mathbf{A}^\circ\mathbf{O}^{-1}\mathbf{A}^{\circ T}). \tag{4-12}$$

Simplify $\hat{\mathbf{y}}^{ML}$ to

$$\hat{\mathbf{y}}^{ML} \sim N(\mathbf{y}, \mathbf{C}), \tag{4-13}$$

where

$$\mathbf{C} = \sigma^2(\mathbf{R}^T\mathbf{O}\mathbf{R})^{-1}. \tag{4-14}$$

The unknown σ^2 can be estimated by

$$\hat{\sigma}^2 = \frac{\left\|\mathbf{T}^{-1}(\mathbf{z} - \mathbf{R}\hat{\mathbf{y}}^{ML})\right\|^2}{N - p + 2} \tag{4-15}$$

Now we are ready to relate the maximum likelihood estimate to the James-Stein estimate. The James-Stein estimate of $\mathbf{y}$ can then be written as

$$\hat{\mathbf{y}}^{JSLS} = \left(1 - \frac{(min\{(p-2), 2(p^e-2)\})^+}{(\hat{\mathbf{y}}^{ML})^T\mathbf{C}^{-1}(\hat{\mathbf{y}}^{ML})}\right)^+ \hat{\mathbf{y}}^{ML}, \tag{4-16}$$

where the superscript *JSLS* denotes James-Stein Least Squares.

The JSLS estimation can be summarized as follows:

1) Calculate the maximum likelihood estimate of $\mathbf{y}$, using Eq. (4-11).

2) Estimate σ^2 using Eq. (4-15), then estimate $\mathbf{C}$, using Eq. (4-14).

3) Calculate the James-Stein least squares estimate of $\mathbf{y}$, using Eq. (4-16).

Shrinkage Least Squares (SLS)

In JSLS, since σ^2 is unknown and has to be estimated from observation data. It follows that the covariance matrix $\mathbf{C}$ is indirectly calculated from observation data contaminated by noise. Meanwhile, it is possible that both $\mathbf{R}$ and $\mathbf{T}$ are estimated from observation data with uncertainty. Therefore, the estimation of $\mathbf{C}$ from Eq. (4-14) can be quite inaccu-

rate. The RCME can be applied to reduce the estimation error. The estimated $\mathbf{C}$ in JSLS, Eq. (4-16), can be viewed as the sample estimate of the covariance matrix, and therefore the $\mathbf{C}$ estimated by the RCME is

$$\hat{\mathbf{C}}_R = k\mathbf{I} + (1-k)\left(\frac{\left\|\mathbf{T}^{-1}(\mathbf{z}-\mathbf{R}\mathbf{y}^{ML})\right\|^2}{N-p+2}\right)(\mathbf{R}^T\mathbf{O}\mathbf{R})^{-1}, \tag{4-17}$$

where the shrinkage intensity k can be estimated from methods introduced in Chapter 3. $\mathbf{y}$ is estimated by

$$\hat{\mathbf{y}}^{SLS} = \left(1 - \frac{(min\{(p-2), 2(p^e-2)\})^+}{(\hat{\mathbf{y}}^{ML})^T\mathbf{C}_R^{-1}(\hat{\mathbf{y}}^{ML})}\right)^+ \hat{\mathbf{y}}^{ML}, \tag{4-18}$$

where the superscript SLS denotes Shrinkage Least Squares.

The SLS estimation can be summarized as follows:

1) Calculate the maximum likelihood estimate of $\mathbf{y}$, using Eq. (4-11),

2) Estimate $\mathbf{C}$, using Eq. (4-17).

3) Calculate the shrinkage least square estimate of $\mathbf{y}$, using Eq. (4-18).

We call this improved least squares algorithm Shrinkage Least Squares (SLS), because both of the improvements are based on shrinkage: the shrinkage of the covariance matrix by the RCME, and the shrinkage of the mean vector by the James-Stein estimator.

3. Shrinkage Recursive Least Squares (SRLS)

In this section, we first present the model, then we summarize recursive least squares and James-Stein recursive least squares. Later, we develop the shrinkage recursive least squares algorithm. Finally, the James-Stein Ledoit Recursive Least Squares algorithm is developed.

The Model

In this section we consider the ARX model (Auto Regressive with eXogenous input). At time instant n,

$$z(n) = \sum_{i=1}^{n_a} a_i z(n-i) + \sum_{i=1}^{n_b} b_i u(n-i) + w(n) \tag{4-19}$$

where $z(n)$ is the observed output, $u(n)$ is the known exogenous input, and $w(n)$ is Gaussian white noise, $w(n) \sim N(0, \sigma^2)$. The model can be rearranged as follows:

$$z(n) = \mathbf{r}^T(n)\mathbf{y}(n) + w(n), \tag{4-20}$$

where $\mathbf{r}(n)$ is defined as

$$\mathbf{r}(n) = [z(n-1), z(n-2), \ldots, z(n-n_a), u(n-1), u(n-2), \ldots, u(n-n_b)]^T, \tag{4-21}$$

and $\mathbf{y}(n)$ contains the parameters estimated at n,

$$\mathbf{y}(n) = [a_1, a_2, \ldots, a_{n_a}, b_1, b_2, \ldots, b_{n_b}]^T. \tag{4-22}$$

At time instant n, we seek the estimate of $\mathbf{y}(n)$ based on all available observations $z(1), z(2), \ldots, z(n)$ and inputs $u(1), u(2), \ldots, u(n)$. It is assumed that $z(i) = 0$ and $u(i) = 0$ for $i \leq 0$.

Rewrite Eq. (4-20) in the form of Eq. (4-5),

$$\mathbf{z}(n) = \mathbf{R}(n)\mathbf{y}(n) + \mathbf{T}(n)\mathbf{w}(n). \tag{4-23}$$

Each term in Eq. (4-23) is defined as follows

$$\mathbf{z}(n) = [z(1), z(2), \ldots, z(n)]^T,$$

$$\mathbf{R}(n) = [\mathbf{r}(1), \mathbf{r}(2), \ldots, \mathbf{r}(n)]^T,$$

$$\mathbf{T}(n) = \mathbf{I}.$$

If the forgetting factor λ is considered, then

$$\mathbf{T}(n) = [\mathbf{O}(n)]^{-1/2},$$

where $\mathbf{O}(n) = diag(\lambda^{n-1}, \lambda^{n-2}, \ldots, 1)$.

Recursive Least Squares (RLS)

We consider the ARX model in the form of Eq. (4-23) and derive the recursive update of the parameters $\mathbf{y}(n)$. The forgetting factor λ is used to account for parameter variation in a nonstationary environment.

The cost function is

$$\begin{aligned} J &= \mathbf{w}^T(n)\mathbf{w}(n) \\ &= [\mathbf{T}^{-1}(n)(\mathbf{z}(n) - \mathbf{R}(n)\mathbf{y}(n))]^T[\mathbf{T}^{-1}(n)(\mathbf{z}(n) - \mathbf{R}(n)\mathbf{y}(n))] \\ &= \mathbf{z}^T(n)\mathbf{O}(n)\mathbf{z}(n) - \mathbf{z}^T(n)\mathbf{O}(n)\mathbf{R}(n)\mathbf{y}(n) + \mathbf{y}^T(n)\mathbf{R}^T(n)\mathbf{O}(n)\mathbf{R}(n)\mathbf{y}(n) \end{aligned} \quad (4\text{-}24)$$

Take the derivative of J with respect to $\mathbf{y}(n)$ and set the value to 0,

$$\hat{\mathbf{y}}(n) = [\mathbf{R}^T(n)\mathbf{O}(n)\mathbf{R}(n)]^{-1}\mathbf{R}^T(n)\mathbf{O}(n)\mathbf{z}(n). \quad (4\text{-}25)$$

Since

$$\mathbf{R}^T(n)\mathbf{O}(n)\mathbf{R}(n) = \lambda\mathbf{R}^T(n-1)\mathbf{O}(n-1)\mathbf{R}(n-1) + \mathbf{r}(n)\mathbf{r}^T(n), \quad (4\text{-}26)$$

let $\mathbf{P}(n)$ be the inverse correlation matrix defined by

$$\mathbf{P}(n) = [\mathbf{R}^T(n)\mathbf{O}(n)\mathbf{R}(n)]^{-1}. \quad (4\text{-}27)$$

By the *matrix inversion lemma* (see Appendix)

$$\mathbf{P}(n) = \lambda^{-1}\mathbf{P}(n-1) - \frac{\lambda^{-1}\mathbf{P}(n-1)\mathbf{r}(n)\mathbf{r}^T(n)\mathbf{P}(n-1)}{\lambda + \mathbf{r}^T(n)\mathbf{P}(n-1)\mathbf{r}(n)}. \quad (4\text{-}28)$$

Define $\mathbf{k}(n)$ as the gain vector

$$\mathbf{k}(n) = \frac{\mathbf{P}(n-1)\mathbf{r}(n)}{\lambda + \mathbf{r}^T(n)\mathbf{P}(n-1)\mathbf{r}(n)}. \tag{4-29}$$

Then Eq. (4-28) can be simplified to

$$\mathbf{P}(n) = \lambda^{-1}[\mathbf{P}(n-1) - \mathbf{k}(n)\mathbf{r}^T(n)\mathbf{P}(n-1)]. \tag{4-30}$$

By rearranging Eq. (4-29), we can get

$$\begin{aligned} \mathbf{k}(n) &= \lambda^{-1}[\mathbf{P}(n-1)\mathbf{r}(n) - \mathbf{r}^T(n)\mathbf{P}(n-1)\mathbf{r}(n)] \\ &= \lambda^{-1}[\mathbf{P}(n-1) - \mathbf{k}(n)\mathbf{r}^T(n)\mathbf{P}(n-1)]\mathbf{r}(n) \\ &= \mathbf{P}(n)\mathbf{r}(n) \end{aligned} \tag{4-31}$$

Now we derive the update of $\hat{\mathbf{y}}(n)$. Notice that

$$\mathbf{R}^T(n)\mathbf{O}(n)\mathbf{z}(n) = \lambda\mathbf{R}^T(n-1)\mathbf{O}(n-1)\mathbf{z}(n-1) + \mathbf{r}^T(n)\mathbf{z}(n), \tag{4-32}$$

From Eq. (4-25), (4-27), (4-29), (4-31), and (4-32),

$$\begin{aligned} \hat{\mathbf{y}}(n) &= [\mathbf{R}^T(n)\mathbf{O}(n)\mathbf{R}(n)]^{-1}\mathbf{R}^T(n)\mathbf{O}(n)\mathbf{z}(n) \\ &= \mathbf{P}(n)[\lambda\mathbf{R}^T(n-1)\mathbf{O}(n-1)\mathbf{z}(n-1) + \mathbf{r}^T(n)\mathbf{z}(n)] \\ &= \mathbf{P}(n-1)\mathbf{R}^T(n-1)\mathbf{O}(n-1)\mathbf{z}(n-1) \\ &\quad - \mathbf{k}(n)\mathbf{r}^T(n)\mathbf{P}(n-1)\mathbf{R}^T(n-1)\mathbf{O}(n-1)\mathbf{z}(n-1) + \mathbf{P}(n)(\mathbf{r}^T(n)\mathbf{z}(n)) \\ &= \hat{\mathbf{y}}(n-1) + \mathbf{k}(n)[\mathbf{z}(n) - \mathbf{r}^T(n)\hat{\mathbf{y}}(n-1)] \end{aligned} \tag{4-33}$$

The weighted recursive least squares can be summarized as follows [41]:

1) Initialization: At $n = 0$, set

$$\mathbf{P}(0) = 100\mathbf{I}, \text{ and } \hat{\mathbf{y}}(0) = \mathbf{0}. \tag{4-34}$$

2) Updating Equations: At each time instant $n = 1, 2, \ldots, N$, calculate $\mathbf{k}(n)$, $\hat{\mathbf{y}}(n)$, and $\mathbf{P}(n)$ by Eq. (4-29), (4-33), and (4-30), respectively.

James-Stein Recursive Least Squares (JSRLS)

Since at each time instant, n, Eq. (4-23) holds, it is possible to improve the algorithm for estimating $\mathbf{y}(n)$ by using the James-Stein estimator, as in the James-Stein Least Squares algorithm. However, the development of this improved algorithm is a little bit different than that of the James-Stein Least Squares algorithm. First of all, as we mentioned before, the James-Stein estimator can only be applied to Eq. (4-5). In general, Eq. (4-23) is equivalent to Eq. (4-5) only asymptotically. Secondly, this algorithm requires certain care to be taken to account for the forgetting factor.

In order to apply the James-Stein estimator to recursive least squares, we need the recursive estimate of $\hat{\sigma}^2$. The recursive estimate of $\hat{\sigma}^2$ is derived as follows. From Eq. (4-15),

$$\begin{aligned}\hat{\sigma}^2(n) &= \frac{[\mathbf{T}^{-1}(n)z(n) - \mathbf{T}^{-1}(n)\mathbf{R}(n)\mathbf{y}(n)]^T[\mathbf{T}^{-1}(n)z(n) - \mathbf{T}^{-1}(n)\mathbf{R}(n)\mathbf{y}(n)]}{n-p+2} \\ &= (z^T(n)\mathbf{O}(n)z(n) - 2z^T(n)\mathbf{O}(n)\mathbf{R}(n)\mathbf{y}(n) \\ &\quad + \mathbf{y}^T(n)\mathbf{R}^T(n)\mathbf{O}(n)\mathbf{R}(n)\mathbf{y}(n)]/(n-p+2) \\ &= \frac{s(n) - 2\mathbf{d}^T(n)\mathbf{y}(n) + \mathbf{y}^T(n)\mathbf{P}^{-1}(n)\mathbf{y}(n)}{n-p+2}\end{aligned}, \quad (4\text{-}35)$$

where $s(n)$ and $\mathbf{d}(n)$ are defined by

$$s(n) = z^T(n)\mathbf{O}(n)z(n),$$

$$\mathbf{d}(n) = \mathbf{R}^T(n)\mathbf{O}(n)z^T(n).$$

The recursive update of $s(n)$ and $\mathbf{d}(n)$ can be shown to be

$$s(n) = \lambda s(n-1) + [z(n)]^2, \quad (4\text{-}36)$$

$$\mathbf{d}(n) = \lambda \mathbf{d}(n-1) + \mathbf{r}(n)z(n). \quad (4\text{-}37)$$

In order to update $\hat{\sigma}^2$, n should also be replaced by $N^{eff}(n)$ and updated according to

$$N^{eff}(n) = \lambda N^{eff}(n) + 1 . \tag{4-38}$$

Let $n_{max} = max(n_a, n_b) + 1$. The algorithm is summarized as follows [46]:

Initialization: At time instant $n = n_{max}$,

$$\mathbf{P}(n) = (\mathbf{R}^T(n)\mathbf{O}(n)\mathbf{R}(n))^{-1}, \tag{4-39}$$

$$\mathbf{d}(n) = \mathbf{R}^T(n)\mathbf{O}(n)\mathbf{z}(n), \tag{4-40}$$

$$\hat{\mathbf{y}}(n) = \mathbf{P}(n)\mathbf{d}(n), \tag{4-41}$$

$$s(n) = \mathbf{z}^T(n)\mathbf{O}(n)\mathbf{z}(n), \tag{4-42}$$

$$\mathbf{Q}(n) = \mathbf{R}^T(n)\mathbf{O}(n)\mathbf{R}(n), \tag{4-43}$$

$$N^{eff}(n) = n, \tag{4-44}$$

$$\bar{\mathbf{y}}(n) = \begin{cases} \text{a priori estimation of } \mathbf{y}(n) & \text{if available} \\ \mathbf{0} & \text{otherwise} \end{cases} \tag{4-45}$$

Updating Equations: At time instant $n = n_{max} + 1, n_{max} + 2, \ldots, N$,

$$\mathbf{k}(n) = \frac{\mathbf{P}(n-1)\mathbf{r}(n)}{\lambda + \mathbf{r}^T(n)\mathbf{P}(n)\mathbf{r}(n)}, \tag{4-46}$$

$$\hat{\mathbf{y}}(n) = \hat{\mathbf{y}}(n-1) + \mathbf{k}(n)[z(n) - \mathbf{r}^T(n)\hat{\mathbf{y}}(n-1)], \tag{4-47}$$

$$\mathbf{P}(n) = \lambda^{-1}[\mathbf{P}(n-1) - \mathbf{k}(n)\mathbf{r}^T(n)\mathbf{P}(n-1)], \tag{4-48}$$

$$\mathbf{d}(n) = \lambda\mathbf{d}(n-1) + z(n)\mathbf{r}(n), \tag{4-49}$$

$$s(n) = \lambda s(n-1) + [z(n)]^2, \tag{4-50}$$

$$N^{eff}(n) = \lambda N^{eff}(n) + 1, \tag{4-51}$$

$$\mathbf{Q}(n) = \lambda \mathbf{Q}(n-1) + \mathbf{r}(n)\mathbf{r}^T(n), \tag{4-52}$$

$$p^e = \frac{tr(\mathbf{P}(n))}{\lambda_{max}(\mathbf{P}(n))}, \tag{4-53}$$

$$\hat{\sigma}^2(n) = \frac{s(n) - 2(\hat{\mathbf{y}}(n) - \bar{\mathbf{y}}(n-1))^T \mathbf{d}(n) + [\hat{\mathbf{y}}(n) - \bar{\mathbf{y}}(n)]^T \mathbf{Q}(n)[\hat{\mathbf{y}}(n) - \bar{\mathbf{y}}(n)]}{N^{eff}(n) - p + 2}, \tag{4-54}$$

$$\mathbf{S}^{-1}(n) = \frac{\mathbf{Q}(n)}{\hat{\sigma}^2(n)}, \tag{4-55}$$

$$\hat{\mathbf{y}}^{JSRLS}(n) = \left(1 - \frac{(min\{(p-2), 2(p^e-2)\})^+}{[\hat{\mathbf{y}}(n) - \bar{\mathbf{y}}(n-1)]^T \mathbf{S}^{-1}(n)[\hat{\mathbf{y}}(n) - \bar{\mathbf{y}}(n-1)]}\right)^+ \hat{\mathbf{y}}(n), \tag{4-56}$$

$$\bar{\mathbf{y}}(n) = \kappa \hat{\mathbf{y}}^{JSRLS}(n) + (1-\kappa)\bar{\mathbf{y}}(n-1), \; 0 \le \kappa \le 1 \tag{4-57}$$

In the above formulation, the technique of *shifting the origin* is applied [46], which can decrease the risk of the James-Stein estimator by shifting the origin from 0 to $\bar{\mathbf{y}}(n-1)$ at time instant n. Specifically, the variable $\bar{\mathbf{y}}(n)$ is introduced and initialized at Eq. (4-45), and in the James-Stein estimator formulation Eq. (4-35) and Eq. (4-4), $\hat{\mathbf{y}}(n)$ is replaced by $\hat{\mathbf{y}}(n) - \bar{\mathbf{y}}(n-1)$, resulting in Eq. (4-54) and (4-56). At each iteration, $\bar{\mathbf{y}}(n)$ is updated by Eq. (4-57).

Shrinkage Recursive Least Squares (SRLS)

In shrinkage recursive least squares, we propose a further improvement to recursive least squares based on James-Stein recursive least squares. The RCME is used to estimate the covariance matrix, as we did in shrinkage least squares. Specifically, we find the recur-

sive version of Eq. (4-17) and estimate the shrinkage intensity using Ledoit's method with constraint.

For simplicity, we start at $n = 0$, as in recursive least squares. The new SRLS algorithm is summarized as follows:

Initialization: At time instant $n = 0$

$$\mathbf{P}(0) = (\mathbf{R}^T(0)\mathbf{O}(0)\mathbf{R}^T(0))^{-1} = 100\mathbf{I}, \tag{4-58}$$

$$s(0) = z^T(0)\mathbf{O}(0)z(0) = 0, \tag{4-59}$$

$$\mathbf{d}(0) = \mathbf{R}^T(0)\mathbf{O}(0)z(0) = \mathbf{0}, \tag{4-60}$$

$$N^{eff}(0) = 0, \tag{4-61}$$

$$\bar{\mathbf{y}}(n) = \begin{cases} \text{a priori estimation of } \mathbf{y}(n) & \text{if available} \\ \mathbf{0} & \text{otherwise} \end{cases}, \tag{4-62}$$

$$\bar{b}(0) = 0 \tag{4-63}$$

Updating Equations: At time instant $n = 1, 2, \ldots, N$,

$$k(n) = \frac{\mathbf{P}(n-1)\mathbf{r}(n)}{\lambda + \mathbf{r}^T(n)\mathbf{P}(n)\mathbf{r}(n)}, \tag{4-64}$$

$$\hat{\mathbf{y}}(n) = \hat{\mathbf{y}}(n-1) + \boldsymbol{k}(n)[z(n) - \mathbf{r}^T(n)\hat{\mathbf{y}}(n-1)], \tag{4-65}$$

$$\mathbf{P}(n) = \lambda^{-1}[\mathbf{P}(n-1) - \mathbf{k}(n)\mathbf{r}^T(n)\mathbf{P}(n-1)], \tag{4-66}$$

$$\mathbf{d}(n) = \lambda\mathbf{d}(n-1) + z(k)\mathbf{r}(n), \tag{4-67}$$

$$s(n) = \lambda s(n-1) + [z(n)]^2, \tag{4-68}$$

$$N^{eff}(n) = \lambda N^{eff}(n) + 1, \tag{4-69}$$

$$\hat{\sigma}^2(n) = \frac{s(n) - 2(\hat{\mathbf{y}}(n) - \bar{\mathbf{y}}(n-1))^T \mathbf{d}(n) + (\hat{\mathbf{y}}(n) - \bar{\mathbf{y}}(n-1))^T \mathbf{P}_k^{-1} (\hat{\mathbf{y}}(n) - \bar{\mathbf{y}}(n-1))}{N^{eff}(n) - p + 2} \quad (4\text{-}70)$$

$$\mathbf{S}(n) = \hat{\sigma}^2 \mathbf{P}(n), \quad (4\text{-}71)$$

$$\bar{b}(n) = \lambda \bar{b}(n-1) + \left\| \hat{\mathbf{y}}(n) \hat{\mathbf{y}}^T(n) - \mathbf{S}(n) \right\|^2, \quad (4\text{-}72)$$

$$b(n) = \frac{\bar{b}(n)}{(N^{eff}(n))^2}, \quad (4\text{-}73)$$

$$c(n) = \|\mathbf{S}(n) - \mathbf{I}\|^2, \quad (4\text{-}74)$$

$$k(n) = \min\left(\frac{b(n)}{c(n)}, k^{up}\right), \quad (4\text{-}75)$$

$$\hat{\mathbf{C}}(n) = k(n)\mathbf{I} + (1 - k(n))\mathbf{S}(n), \quad (4\text{-}76)$$

$$p^e(n) = \frac{\mathrm{tr}(\hat{\mathbf{C}}(n))}{\lambda_{max}(\hat{\mathbf{C}}(n))}, \quad (4\text{-}77)$$

$$\hat{\mathbf{y}}^{SRLS}(n) = \left(1 - \frac{(min\{(p-2), 2(p^e - 2)\})^+}{(\hat{\mathbf{y}}(n) - \bar{\mathbf{y}}(n-1))^T \hat{\mathbf{C}}^{-1}(n)(\hat{\mathbf{y}}(n) - \bar{\mathbf{y}}(n-1))}\right)^+ \boldsymbol{x}_k, \quad (4\text{-}78)$$

$$\bar{\mathbf{y}}(n-1) = \kappa \hat{\mathbf{y}}^{SRLS}(n) + (1 - \kappa)\bar{\mathbf{y}}(n-1), \quad (4\text{-}79)$$

where k^{up} is the upper bound for the $k(n)$ value, and $\hat{\mathbf{C}}(n)$ is the estimated covariance matrix by the RCME.

Let's take a further look to see how the algorithm works. Eq. (4-64) - (4-69) and Eq. (4-70) are the same as Eq. (4-46) - (4-51) and Eq. (4-54). Eq. (4-71) is the "sample covariance matrix". Eq. (4-72) - (4-75) estimate the shrinkage intensity by the constraint method introduced in Chapter 3, where $\bar{b}(n)$ is initialized in Eq. (4-64) and Eq. (4-72) - (4-75) use

Ledoit's method for estimating shrinkage intensity as in Eq. (3-18) with the upper limit replaced by k^{up}. Eq. (4-76) is the covariance matrix estimated by the RCME. Eq. (4-77) gives the effective dimension of $\mathbf{y}(n)$ based on the covariance matrix estimated by the RCME. Eq. (4-78) is the same as Eq. (4-56), except that the sample covariance matrix $\mathbf{S}^{-1}(n)$ is replaced by $\hat{\mathbf{C}}^{-1}$, the covariance matrix estimated by the RCME. Finally, Eq. (4-79) is the same as Eq. (4-57).

The name Shrinkage Recursive Least Squares (SRLS) comes from the fact that the covariance matrix is estimated by the RCME and the mean is estimated by James-Stein's estimator, both of which belong to the class of shrinkage methods.

James-Stein Ledoit Recursive Least Squares (JSLRLS)

We want to propose one more new recursive algorithm - JSLRLS. The JSLRLS is similar to the SRLS. The only difference is that the covariance matrix is estimated by the LCME instead of the RCME.

In the JSLRLS, the following updating equations replace Eq. (4-72) - (4-76),

$$m(n) = \frac{\mathrm{tr}(\mathbf{S}(n))}{p}, \tag{4-80}$$

$$d(n) = \|\mathbf{S}(n) - m(n)\mathbf{I}\|^2, \tag{4-81}$$

$$\bar{b}(n) = \lambda \bar{b}(n-1) + \left\|\hat{\mathbf{y}}(n)\hat{\mathbf{y}}^T(n) - \mathbf{S}(n)\right\|^2, \tag{4-82}$$

$$b(n) = \min\left[\frac{\bar{b}(n)}{(N^{eff}(n))^2}, d(n)\right], \tag{4-83}$$

$$a(n) = d(n) - b(n), \tag{4-84}$$

$$\hat{\mathbf{C}}(n) = \frac{b(n)}{d(n)} m(n)\mathbf{I} + \frac{a(n)}{d(n)} \mathbf{S}(n). \tag{4-85}$$

Eq. (4-80) - (4-85) are the recursive version of the LCME. They are essentially the same as Eq. (2-22) - (2-27), with $m(n)$, $d(n)$, $\bar{b}(n)$, $b(n)$, and $a(n)$ corresponds to m, d^2, $\bar{b}^2$, b^2, and a^2 in Eq. (2-22) - (2-27).

Later in Chapter 5, we will provide the simulation results for comparing the standard RLS, JSRLS, SRLS, and JSLRLS.

CHAPTER 5

SIMULATION RESULTS: RIDGE COVARIANCE ESTIMATION

In this chapter, we present the simulation results for the Ridge Covariance Matrix Estimator (RCME) proposed in Chapter 3, and its application to Recursive Least Squares (RLS) - the Shrinkage Recursive Least Squares (SRLS) algorithm, as proposed in Chapter 4.

This chapter contains two sections. Section one is the comparison of different shrinkage covariance matrix estimators by Monte Carlo simulation. Section two is the Monte Carlo simulation for comparing different recursive least squares algorithms.

1. RCME: Comparison by Monte Carlo Simulation

In this section, we present the simulation results for different covariance matrix estimators. The first is the sample covariance matrix. The second is the RCME with Ledoit's method for estimating shrinkage intensity. The third is the RCME with optimal shrinkage intensity value, in which the true covariance matrix is assumed to be known and shrinkage intensity is chosen by the golden section search with parabolic interpolation optimization. The third method is only used for comparison purposes and is not practical since the true covariance matrix is unknown beforehand. The fourth is the LCME. The three estimated covariance matrices are denoted by $\hat{\mathbf{C}}_R$, $\hat{\mathbf{C}}_R^o$, and $\hat{\mathbf{C}}_L$, respectively. We also compare the

shrinkage intensity estimate by Ledoit's method with the optimal shrinkage intensity estimated by the golden section search with parabolic interpolation optimization method.

The Monte Carlo simulations were carried out with different numbers of sample points, different eigenstructures of the covariance matrix, and different matrix dimensions to show how the estimators perform.

The Monte Carlo simulation steps are as follows:

1) Set matrix dimension $p = 3$.

2) Set the eigenvalues of the true covariance matrix $\lambda = \{2, 1.5, 1\}$.

3) Set number of sample points $N = 10$.

4) Randomly generate true covariance matrix $\mathbf{C}$ and N samples.

5) Compute sample covariance matrix $\mathbf{S}$, $\hat{\mathbf{C}}_R$ - covariance matrix estimated by the RCME with shrinkage intensity k estimated by the Ledoit method, $\hat{\mathbf{C}}_R^o$ - covariance matrix estimated by the RCME with k estimated by the golden section search with parabolic interpolation optimization method, and $\hat{\mathbf{C}}_L$ - covariance matrix estimated by the LCME. Calculate the squared error of these estimations, which are defined by $\|\mathbf{S} - \mathbf{C}\|^2$, $\|\hat{\mathbf{C}}_L - \mathbf{C}\|^2$, $\|\hat{\mathbf{C}}_L^o - \mathbf{C}\|^2$, and $\|\hat{\mathbf{C}}_L - \mathbf{C}\|^2$, respectively.

6) Repeat steps 4-5 1000 times. Calculate the average values of the MSE of the covariance estimation for the four estimators.

7) Set $N = 20, 30, \ldots, 200$, repeat step 4-6. Record the four average MSEs.

8) Set $\lambda = \{10, 5, 1\}$, and $\lambda = \{100, 50, 1\}$, respectively, repeat step 3-7.

9) Set matrix dimension $p = 5$, repeat step 2-8.

We compared the four different covariance matrix estimators by computing the Percentage Relative Improvement in Average Loss (PRIAL) [37] [8] defined by

$$\frac{E[\|\mathbf{S}-\mathbf{C}\|^2]-E[\|\hat{\mathbf{C}}-\mathbf{C}\|^2]}{E[\|\mathbf{S}-\mathbf{C}\|^2]}, \tag{5-1}$$

where $\hat{\mathbf{C}}$ can be $\hat{\mathbf{C}}_R$, $\hat{\mathbf{C}}_R^o$, or $\hat{\mathbf{C}}_L$. Since we are averaging MSEs over 1000 times at step 6, the average MSE is a good estimate of the Expectations in Eq. (5-1). The PRIAL can not be greater than 1. The larger the value, the better the improvement of the estimator over the sample covariance matrix $\mathbf{S}$. Negative PRIAL indicates that the estimator performs worse than the sample covariance estimate.

Figure 5-1 is the plot of the PRIAL vs. number of samples N for different matrix dimension p and different eigenstructures. In each of the plots, the solid line is the PRIAL of $\hat{\mathbf{C}}_R$, the dotted line is the PRIAL of $\hat{\mathbf{C}}_R^o$, and the dash-dotted line is the PRIAL of $\hat{\mathbf{C}}_L$.

From Figure 5-1, the following observations can be made:

- RCME with optimal shrinkage intensity: In all cases, $\hat{\mathbf{C}}_R^o$ has the largest PRIAL, which means $\hat{\mathbf{C}}_R^o$ is the best covariance estimator among all the four estimators. This implies that if a good method for estimating the optimal shrinkage intensity k_o exists, the RCME can outperform the LCME.
- Sample covariance matrix: Since all three PRIALs are greater than 0 for all cases, the $\hat{\mathbf{C}}_R$ and $\hat{\mathbf{C}}_R^o$ by the RCME, and the $\hat{\mathbf{C}}_L$ by the LCME all outperform the sample covariance matrix in terms of MSE.

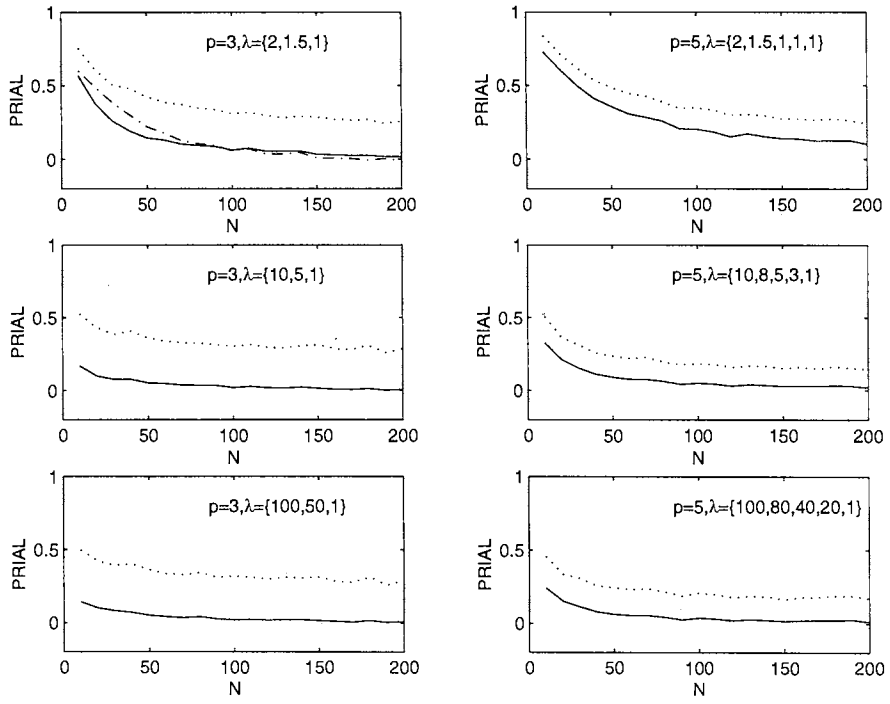


Figure 5-1 PRIAL of $\hat{\mathbf{C}}_R$, $\hat{\mathbf{C}}_R^o$, and $\hat{\mathbf{C}}_L$

($\hat{\mathbf{C}}_R$ - solid line, $\hat{\mathbf{C}}_R^o$ - dotted line, $\hat{\mathbf{C}}_L$ - dash-dotted line)

- Comparison of $\hat{\mathbf{C}}_R$ and $\hat{\mathbf{C}}_L$: The PRIALs of $\hat{\mathbf{C}}_R$ and $\hat{\mathbf{C}}_L$ are about the same, except for the first case. This implies there is no big advantage to choosing the LCME. The disadvantage of the LCME is obvious: two parameters need to be estimated, and there is no other way (but Ledoit's method) to estimate both parameters. The PRIAL of $\hat{\mathbf{C}}_L$ is larger than that of $\hat{\mathbf{C}}_R$ only in the first case, when the dimension of the covariance matrix is small ($p = 3$), the eigenstructure of the covariance matrix is close to the identity matrix ($\lambda = \{2, 1.5, 1\}$), and the number of observations is small ($N < 80$).

- Effect of sample size: When the sample size N gets larger, the PRIAL of $\hat{\mathbf{C}}_R$, $\hat{\mathbf{C}}_R^o$, and $\hat{\mathbf{C}}_L$ gets smaller. This is because the sample covariance matrix $\mathbf{S}$ is a consistent estimator of the true covariance matrix $\mathbf{C}$. When more samples are available, less shrinkage toward the identity matrix is required.
- Effect of matrix dimension: Although only two cases were studied ($p = 3$ and $p = 5$), the effect of p is clear. When p is small, for the same PRIAL, fewer sample points are needed.
- Effect of the eigenstructures: The identity matrix has equal eigenvalues. If the eigenvalues of a covariance matrix are close to each other, then we say that the eigenstructure of the covariance matrix is close to the identity matrix. Simulation results show that if the true covariance matrix has an eigenstructure close to the identity matrix, the PRIALs of $\hat{\mathbf{C}}_R$, $\hat{\mathbf{C}}_R^o$, and $\hat{\mathbf{C}}_L$ are large. This is because all three shrinkage estimated covariance matrices, $\hat{\mathbf{C}}_R$, $\hat{\mathbf{C}}_R^o$, and $\hat{\mathbf{C}}_L$, shrink toward the identity matrix.

Figure 5-2 is the plot of the shrinkage intensity k vs. the number of samples N for different matrix dimensions and different eigenstructures. In each of the plots, the solid line is the shrinkage intensity estimated by the Ledoit's method, the dotted line is the optimal shrinkage intensity estimated by the golden section search with parabolic interpolation optimization method.

From Figure 5-2, the following observations can be made:

- Effect of sample size: As sample size gets larger, shrinkage intensity becomes smaller, which means less shrinkage towards the identity matrix. When the sample size is small, Ledoit's method tends to over estimate the optimal k. On

the other hand, when the sample size is large, Ledoit's method tends to under estimate the optimal k. This indicates that when we estimate the covariance matrix with small samples, it may help to put a cap on the maximum allowable value of k. This observation results in the Constraint method for shrinkage intensity estimation in Chapter 3 and using Eq. (4-75) in Shrinkage Recursive Least Squares.

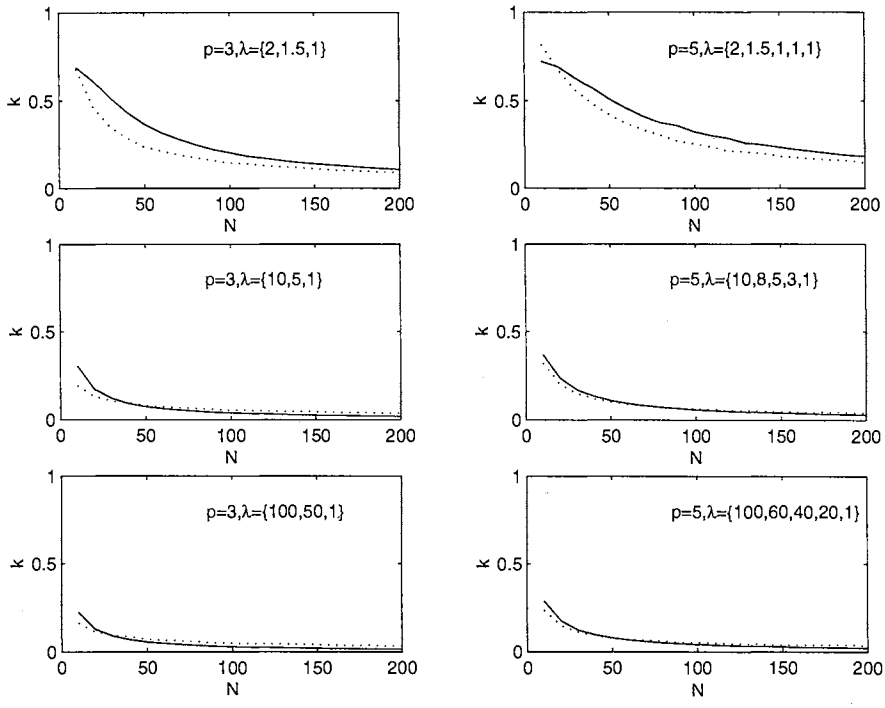


Figure 5-2 Shrinkage Intensity
(k estimated by the Ledoit method - solid line, optimal k - dotted line)

- Effect of matrix dimension: Although only two cases were simulated ($p = 3$ and $p = 5$), the effect of p is clear. When p is larger, for the same sample size, k tends to be larger, which means more shrinkage toward the identity ma-

trix is necessary.

- Effect of the eigen structure: If the true covariance matrix has an eigen structure close to the identity matrix, for the same sample size, k is larger, which means more shrinkage towards the identity matrix is required.

Because of the wide range of N, p and the eigenstructure, it is difficult to apply the Filtering method and the Constraint method (see Chapter 4) in this simulation. We will see how these two methods can be used in the estimation of k_o in the next sections and in Chapter 9.

2. SRLS: Comparison by Monte Carlo Simulation

In this section, we compare four different recursive least squares algorithms: The first is standard Recursive Least Squares (RLS) as described by Eq. (4-34), (4-29), (4-33), and (4-30) in Chapter 4. The second is James-Stein Recursive Least Squares (JSRLS) as described by Eq. (4-39) - (4-57). The third is Shrinkage Recursive Least Squares (SRLS) as described by Eq. (4-58) - (4-79). (In SRLS, the filtering method (see Chapter 3) was used for estimating the shrinkage intensity.) The fourth is James-Stein-Ledoit Recursive Least Squares (JSLRLS), in which the Ledoit's covariance matrix estimator is used for estimating the covariance matrix.

In the following, we first present the ARX model used in the Monte Carlo simulation and the evaluation criteria for comparing different models. Later, detailed simulation steps are addressed. Finally, the simulation results are presented and discussed.

The following ARX model is used in the simulation:

$$z(n) = \sum_{i=1}^{1} a_i z(n-i) + \sum_{i=1}^{5} b_i u(n-i) + w(n). \tag{5-2}$$

This model is the same model as described in Eq. (4-19) of Chapter 4 with $n_a = 1$ and $n_b = 5$. In JSRLS, SRLS, and JSLRLS, we assume no prior knowledge of the parameters are available. Therefore, $\bar{\mathbf{y}}(n) = \mathbf{0}$, as shown in Eq. (4-45) and Eq. (4-62).

In comparing the performance of the four different algorithms, we calculate the improvement in the squared error of the parameter estimates using the RLS estimates as a baseline. The squared error $e(n)$ at time instant n is defined as

$$e(n) = \left\| \begin{bmatrix} \hat{a}_1 \\ \hat{b}_1 \\ \hat{b}_2 \\ \hat{b}_3 \\ \hat{b}_4 \\ \hat{b}_5 \end{bmatrix} - \begin{bmatrix} a_1 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix} \right\|^2, \tag{5-3}$$

where $\hat{a}_1$, $\hat{b}_1$, $\hat{b}_2$, $\hat{b}_3$, $\hat{b}_4$, and $\hat{b}_5$ are the estimates of the parameters a_1, b_1, b_2, b_3, b_4, and b_5 at time instant n.

We denote the $e(n)$ for RLS, JSRLS, SRLS, and JSLRLS as $e^{RLS}(n)$, $e^{JSRLS}(n)$, $e^{SRLS}(n)$, and $e^{JSLRLS}(n)$, respectively. The RLS algorithm will be used to represent standard performance, and we will measure the improvements obtained by the other three algorithm. The improvements will be measured for three time intervals: initial, final and total. For JSRLS, the improvements are defined as follows:

$$\text{Initial} = \frac{\sum_{n=15}^{45} e^{RLS}(n) - \sum_{n=15}^{45} e^{JSRLS}(n)}{\sum_{n=15}^{45} e^{RLS}(n)}, \tag{5-4}$$

$$\text{Final} = \frac{\sum_{n=970}^{1000} e^{RLS}(n) - \sum_{n=970}^{1000} e^{JSRLS}(n)}{\sum_{n=970}^{1000} e^{RLS}(n)}, \tag{5-5}$$

$$\text{Total} = \frac{\sum_{n=15}^{1000} e^{RLS}(n) - \sum_{n=15}^{1000} e^{JSRLS}(n)}{\sum_{n=15}^{1000} e^{RLS}(n)}. \tag{5-6}$$

For SRLS and JSLRLS, the definitions of the relative improvements are defined in the same way with $e^{JSRLS}(n)$ replaced by $e^{SRLS}(n)$ and $e^{JSLRLS}(n)$.

The simulation steps are as follows:

1) Set the forgetting factor $\lambda = 0.95$ for all four recursive least squares algorithms. Set the number of sample points $N = 1000$. Set the true parameters $[a_1, b_1, b_2, b_3, b_4, b_5]^T = [0.6, 4, 1, 2, 3, 4]^T$.

2) Generate N points of input data $u(n)$, $n = 1, 2, \ldots, N$. Add independent noise with normal distribution $N(0, 0.01)$ to the true parameters. Generate observation data $z(n)$, $n = 1, 2, \ldots, N$ from Eq. (5-2), where $w(n)$, $n = 1, 2, \ldots, N$ is white noise with normal distribution $N(0, 0.25)$.

3) Estimate the parameters using RLS, JSLRLS, SRLS, SLRLS. Compute the estimation errors for each algorithm. In SRLS, the shrinkage intensity is estimated using the constraint method with constraint equal to 0.02.

4) Repeat Steps 2 - 3 750 times. Compute the average estimation error for each algorithm. Calculate the relative improvements for JSLRLS, SRLS, and SLRLS.

5) Set $\lambda = 1.0$, which is equivalent to no forgetting. Repeat Step 2 - 4.

6) Set $[a_1, b_1, b_2, b_3, b_4, b_5]^T = [0.6, 0.4, 0.1, 0.2, 0.3, 0.4]^T$. Repeat Step 2 - 5.

Table 5-1 compares the squared error improvement for $\mathbf{y} = [0.6, 4, 1, 2, 3, 4]^T$.

TABLE 5-1 Relative improvement for $\mathbf{y} = [0.6, 4, 1, 2, 3, 4]^T$

λ	0.95			1.0		
Method	JSRLS	JSLRLS	SRLS	JSLS	JSLRLS	SRLS
Initial	0.0010	0.0024	0.0035	0.0008	0.0017	0.0034
Final	0.0009	0.0020	0.0025	0.0004	0.0012	0.0467
Total	0.0011	0.0029	0.0044	0.0005	0.0015	0.0134

From the table, we can make the following conclusions:

- The shrinkage algorithms, JSRLS, JSLRLS, and SRLS, all outperform the standard RLS. This indicates that the improved estimation of either the mean (JSRLS) or the mean and the covariance matrix (JSLRLS, SRLS), can improve the accuracy of the parameter estimation.
- Both the JSLRLS and the SRLS outperform the JSRLS. This indicates that if the covariance matrix is estimated by some shrinkage estimator (the LCME, as

in the JSLRLS, or the RCME, as in the SRLS), instead of by the sample covariance matrix, as in the JSRLS, the accuracy of the parameter estimation can be improved.

- The SRLS outperforms the JSLRLS. This indicates that the RCME performs better than the LCME.
- Among all the four algorithms, the parameters estimated by the SRLS are the most accurate, no matter whether the forgetting factor is used or not.
- Effect of sample size in SRLS: When the forgetting factor is considered, SRLS has better final estimation error improvement than initial estimation error improvement. This is mainly caused by the Filtering method for estimating the shrinkage intensity. Initially, the incorrect initialization of $\mathbf{P}(n)$ leads to incorrect $\mathbf{S}(n)$, $\bar{b}(n)$, $b(n)$, $c(n)$, and finally $k(n)$, as shown in Eq. (4-58), (4-71), (4-72), (4-73), (4-74), and (4-75). With the progress of the algorithm, $\mathbf{P}(n)$ approaches the correct value, and causes $k(n)$ to approach the correct value also. The final result is that the error of the parameter estimation becomes smaller.

Table 5-2 compares the squared error improvement for

$$\mathbf{y} = [0.6, 0.4, 0.1, 0.2, 0.3, 0.4].$$

For all cases, the conclusions from Table 5-1 are also valid here. The reason that we used another set of parameters is to show that if the parameter set is closer to $\mathbf{0}$, the relative improvements of the JSRLS, JSLRLS, and SRLS are greater. This is because we have assumed the prior $\bar{\mathbf{y}}(n) = \mathbf{0}$ in the James-Stein estimator part of the algorithms. The results indicate that if a good prior can be given, we can further improve the accuracy of the pa-

rameter estimation.

TABLE 5-2 Relative improvement for $\mathbf{y} = [0.6, 0.4, 0.1, 0.2, 0.3, 0.4]^T$

λ	0.95			1.0		
Method	JSRLS	JSLRLS	SRLS	JSRLS	JSLRLS	SRLS
Initial	0.0187	0.0305	0.0321	0.0121	0.0211	0.0306
Final	0.0117	0.0155	0.0195	0.0001	0.0002	0.0004
Total	0.0135	0.0194	0.0220	0.0018	0.0034	0.0042

Till now, we have presented the RCME, which belongs to one class of covariance matrix estimators - the shrinkage estimators. Two applications of the RCME were also presented: shrinkage portfolio optimization and the shrinkage least squares algorithm. Monte Carlo simulation results were also given.

In the next three chapters, we will discuss another class of covariance matrix estimators - Bayesian covariance matrix estimators. Chapter 6 is a survey of the available methods. In Chapter 7, we propose the hierarchical Bayesian covariance matrix estimator. Chapter 8 presents the simulation results.

CHAPTER 6

BAYESIAN METHODS

In this chapter, we review Bayesian estimation methods. This chapter provides background information for developing the Hierarchical Bayesian Covariance Matrix Estimator (HBCME), which will be presented in Chapter 7.

This chapter has three sections. Section one briefly introduces Bayes theorem. Section two reviews the available Bayesian covariance matrix estimation methods and the available choice of priors. Section three presents the Bayesian regularization method for neural network training proposed by MacKay [45] with implementation suggested by Foresee and Hagan [20]. The basic idea of the HBCME is inspired by Bayesian regularization.

1. Bayes' Theorem

Bayes' theorem describes the relationships that exist within a class of simple and conditional probabilities. It was proposed by Thomas Bayes [2]. A reprint of [2] can be found in [54].

Consider a statistical experiment: Let B be the observed data and A be the unobservable quantities or population parameters. Let $P(B)$ be the probability that B occurs, and $P(A)$ be the probability that A occurs. Let $P(A|B)$ be the probability that A occurs conditional on B, and $P(B|A)$ be the probability that B occurs conditional on A. Bayes's Theorem states that (assume $P(B) > 0$)

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}, \tag{6-1}$$

where $P(A|B)$ is the posterior probability, $P(B|A)$ is the likelihood function, $P(B)$ is the prior probability of B, and $P(A)$ is the total probability. The total probability $P(B)$ is a normalizing factor such that $P(B) = \int P(B|A)P(A)dA$.

In Bayesian estimation, A contains parameters that we would like to estimate. The Bayesian estimate of A will be the value that maximizes the posterior density $P(A|B)$.

2. Bayesian Covariance Matrix Estimation

When we are using Bayesian methods to estimate the covariance matrix, the A in Eq. (6-1) consists of the unknown elements of the covariance matrix, and possibly the mean. The trick to Bayesian estimation is in choosing the prior density $P(A)$. We want to choose a prior that reflects pre-existing knowledge of A, and also allows for efficient computation of the value of A that maximizes the posterior density $P(A|B)$.

In this section, we review several approaches for covariance matrix estimation by Bayesian methods: Anderson's method [1], Chen's method [5], and Haff's method [28]. At the end of this section, we briefly introduce some popular priors that can be used for covariance matrix estimation.

First we review Anderson's results. Anderson [1] showed two results for Bayesian covariance matrix estimation. The first result assumes a prior distribution for the covariance matrix. The second result assumes a joint prior distribution for the covariance matrix and the mean.

Assume there are N independent observations, $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$, where each observation $\mathbf{x}_i$ is a $p \times 1$ vector with multivariate normal distribution $N(\mu, \mathbf{C})$. The sample mean $\bar{\mathbf{x}}$ and sample covariance matrix $\mathbf{S}$ are

$$\bar{\mathbf{x}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{x}_i, \tag{6-2}$$

$$\mathbf{S} = \frac{1}{n}\sum_{i=1}^{N} (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T, \tag{6-3}$$

where $n = N - 1$ is the number of degrees of freedom. It can be shown that, since $\mathbf{x}_i$ is normal, $n\mathbf{S}$ will have a Wishart distribution $W(\mathbf{C}, n)$. The first result of Anderson shows that if we assume that the prior distribution of the true covariance matrix $\mathbf{C}$ is inverted Wishart, represented by $W^{-1}(\Sigma, m)$, then the posterior distribution of $\mathbf{C}$ is

$$W^{-1}(n\mathbf{S} + \Sigma, n + m). \tag{6-4}$$

We can then estimate $\mathbf{C}$ by maximizing the posterior distribution.

The second Anderson result assumes a joint density function for the mean and the covariance matrix. The observations $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ are assumed to have a multivariate normal distribution $N(\mu, \mathbf{C})$. Suppose the μ and $\mathbf{C}$ have a joint normal-inverted Wishart prior density

$$N_\mu\left(\iota, \frac{1}{K}\mathbf{C}\right) \times W_{\mathbf{C}}^{-1}(\Sigma, m) \quad , \tag{6-5}$$

then the joint posterior density of the μ and $\mathbf{C}$ is

$$N_{\mu}\left(\frac{1}{N+K}(N\bar{\mathbf{x}}+K\iota),\frac{1}{N+K}\mathbf{C}\right)$$
$$\times W_{\mathbf{C}}^{-1}\left(\Sigma+n\mathbf{S}+\frac{NK}{N+K}(\bar{\mathbf{x}}-\iota)(\bar{\mathbf{x}}-\iota)^{T},N+m\right) \tag{6-6}$$

This can be maximized to obtain estimates of μ and $\mathbf{C}$.

We next review Chen's result. Chen [5] assumed the true covariance matrix has a Wishart prior distribution, as we discussed in Chapter 2, with the form $\mathbf{C} \sim W((\nu\mathbf{G})^{-1},\nu)$, where $\mathbf{G}$ is the prior mean of the true covariance matrix, and ν is the number of degrees of freedom of $\mathbf{G}$. By Bayes's theorem, the posterior distribution of $\mathbf{C}$ is an inverted Wishart distribution

$$\mathbf{C}|(\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_N) \sim W^{-1}(n\mathbf{S}+\nu^{*}\mathbf{G}^{*},N+p+\nu^{*}). \tag{6-7}$$

Therefore the Bayes estimate of $\mathbf{C}$ is the mode of the posterior density:

$$\hat{\mathbf{C}} = \frac{N-1}{N+\nu^{*}-1}\mathbf{S}+\frac{\nu^{*}}{N+\nu^{*}-1}\mathbf{G}^{*}, \tag{6-8}$$

where the unknown hyperparameters $\mathbf{G}^{*}$ and ν^{*} in the Wishart distribution can be estimated by the EM algorithm [10] through iteration. Simulation results showed that the estimation error can be very large in some cases.

Now we present Haff's results. Haff [28] also assumed normally distributed measurements, which produces a Wishart distribution for the sample covariance matrix $\mathbf{S} \sim W(\Sigma,\nu)$. He derived an empirical Bayes covariance estimator of the form

$$\hat{\mathbf{C}} = c[\mathbf{S}+\mathbf{u}t(\mathbf{u})C]. \tag{6-9}$$

where $0 < c < \frac{1}{\nu}$, $\mathbf{u} = \frac{1}{tr(\mathbf{S}^{-1}C)}$, $t(\cdot)$ is nonincreasing, and C is an arbitrary positive definite matrix. He proved that the best estimator among the scalar multiples of $\mathbf{S}$ according to the loss function L_1 (see Eq. (2-1)) is

$$\hat{\mathbf{C}}_1 = \frac{1}{\nu}\mathbf{S}, \tag{6-10}$$

and the best estimator among the scalar multiples of $\mathbf{S}$ according to the loss function L_2 (see Eq. (2-2)) is

$$\hat{\mathbf{C}}_2 = \frac{1}{\nu + p + 1}\mathbf{S}. \tag{6-11}$$

If the parameters c, $\mathbf{u}$, C and the function $t(\cdot)$ satisfy certain conditions, the estimator given by Eq. (6-9) has smaller risk than $\hat{\mathbf{C}}_1$ in terms of loss function L_1 for every Σ. For some given conditions (other than the ones mentioned above) for the parameters c, $\mathbf{u}$, C and function $t(\cdot)$, the estimator of Eq. (6-9) has smaller risk than $\hat{\mathbf{C}}_2$ in terms of loss function L_2 for every Σ.

Efron and Morris [17] presented similar results as Haff. Instead of estimating the true covariance matrix directly, they estimate the inverse of the true covariance matrix. The estimated covariance matrix has less risk than any scalar multiple of the sample covariance matrix $\mathbf{S}$.

In addition to the Wishart prior, there are other priors that can be used for covariance matrix estimation, such as the Jeffreys' prior [35], the reference prior [3], the uniform

shrinkage prior [6] [7], and the log matrix prior [39]. Daniels and Kass [7] [8] has a good review of the possible priors.

3. Bayesian Regularization

In this section, we review the method of Bayesian Regularization for training neural networks. The Bayesian Regularization method was first proposed by MacKay [45]. Later, Foresee and Hagan [20] applied the Gauss-Newton approximation to the estimation of the Hessian matrix in the Bayesian Regularization algorithm.

In the following we show in detail the Gauss-Newton approximation to Bayesian Regularization (GNBR) algorithm.

Let M represent a specific multi-layer feedforward neural network model (for details on neural networks, see [30]), and $\mathbf{w}$ be the weight vector pertaining to this model, which contains n_w elements, $w_1, w_2, \ldots, w_{n_w}$. Let D represent the training data set containing N input-target pairs, $\{p_1, t_1\}, \{p_2, t_2\}, \ldots, \{p_N, t_N\}$, where p_i is the input and t_i is the target. For input p_i, let the corresponding neural network output be a_i, calculated by

$$a_i = f(p_i) + \varepsilon_i, \tag{6-12}$$

where $f(\cdot)$ is the neural network mapping function, and ε_i is the Gaussian noise pertaining to the i th input.

Define the data error E_D to be the sum squared error:

$$E_D = \sum_{i=1}^{N} (t_i - a_i)^2. \tag{6-13}$$

Define the regularization error E_W to be the sum of squares of the network weights:

$$E_W = \sum_{i=1}^{N} w_i^2 . \tag{6-14}$$

The objective of the neural network training is to minimize the following objective function,

$$E_M = bE_D + aE_W, \tag{6-15}$$

where the subscript M represents the total modeling error, a is a parameter related to the variance of the weights, and b is a parameter related to variances of the noise ε_i. The definition of a and b will be given later.

In the Bayesian setting, we assume the weights $\mathbf{w}$ are random variables. After the data is taken, the posterior probability of the weights given the data is

$$P(\mathbf{w}|D, a, b, M) = \frac{P(D|\mathbf{w}, b, M)P(\mathbf{w}|a, M)}{P(D|a, b, M)}, \tag{6-16}$$

where $P(\mathbf{w}|D, a, b, M)$ is the posterior density function of $\mathbf{w}$, $P(D|\mathbf{w}, b, M)$ is the likelihood function, $P(\mathbf{w}|a, M)$ is the prior density function, and $P(D|a, b, M)$ is the total probability density.

If the noises ε_i, $i = 1, 2, \ldots, N$, in the training set data are assumed to be independent and identically distributed (i.i.d.), with distribution $N(0, \sigma^2)$, the likelihood function can be written as

$$P(D|\mathbf{w}, b, M) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{1}{2\sigma^2}(t_i - a_i)^2\right\}$$
$$= \frac{1}{(2\pi\sigma^2)^{N/2}} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{N}(t_i - a_i)^2\right\}. \quad (6\text{-}17)$$

We define $b = \frac{1}{2\sigma^2}$ and $z_D(b) = \left(\frac{\pi}{b}\right)^{N/2}$, then $P(D|\mathbf{w}, b, M)$ simplifies to

$$P(D|\mathbf{w}, b, M) = \frac{1}{z_D(b)} \exp(-bE_D). \quad (6\text{-}18)$$

If the weights $\mathbf{w}$ are also i.i.d. with distribution $N(0, \sigma_w^2)$, then the prior density of the weights can be written as

$$P(\mathbf{w}|a, M) = \prod_{i=1}^{n_w} \frac{1}{\sqrt{2\pi}\sigma_w} \exp\left\{-\frac{1}{2\sigma_w^2} w_i^2\right\}$$
$$= \frac{1}{(2\pi\sigma_w^2)^{n_w/2}} \exp\left\{-\frac{1}{2\sigma_w^2}\sum_{i=1}^{N} w_i^2\right\}, \quad (6\text{-}19)$$

We define $a = \frac{1}{2\sigma_w^2}$ and $z_W(a) = \left(\frac{\pi}{a}\right)^{n_w/2}$, then $P(\mathbf{w}|a, M)$ simplifies to

$$P(\mathbf{w}|a, M) = \frac{1}{z_W(a)} \exp(-aE_W). \quad (6\text{-}20)$$

Since the total probability $P(D|a, b, M)$ is only a normalizing factor in Eq. (6-16), from Eq. (6-18) and Eq. (6-20), we obtain

$$P(\mathbf{w}|D, a, b, M) = \frac{\frac{1}{z_D(b)z_W(a)}\exp(-(bE_D + aE_W))}{P(D|a, b, M)} = \frac{1}{z_M(a, b)}\exp\{-E_M\} \quad (6\text{-}21)$$

where $z_M(a, b)$ is the normalization factor. Since by the fundamental theorem of the probability, $\int P(\mathbf{w}|D, a, b, M)d\mathbf{w} = 1$ for all possible $\mathbf{w}$, we have

$$z_M(a, b) = \int \exp\{-E_M\}d\mathbf{w}. \quad (6\text{-}22)$$

E_M can be approximated by a quadratic function using a second order Taylor expansion

$$E_M(\mathbf{w}) = E_M(\mathbf{w}^{MP}) + \frac{1}{2}(\mathbf{w} - \mathbf{w}^{MP})^T\mathbf{H}^{MP}(\mathbf{w} - \mathbf{w}^{MP}), \quad (6\text{-}23)$$

where $\mathbf{w}^{MP}$ is the most probable estimate (maximum of the posterior distribution) of $\mathbf{w}$ and $\mathbf{H}^{MP}$ is the Hessian of $E_M(\mathbf{w})$ estimated at point $\mathbf{w}^{MP}$, i.e.,

$$\mathbf{H}^{MP} = \nabla^2 E_M(\mathbf{w})\Big|_{\mathbf{w}=\mathbf{w}^{MP}} = (b\nabla^2 E_D + a\nabla^2 E_W)\Big|_{\mathbf{w}=\mathbf{w}^{MP}}. \quad (6\text{-}24)$$

It follows from Eq. (6-22) and Eq. (6-23) that

$$z_M(a, b) = (2\pi)^{n_w/2}|\mathbf{H}^{MP}|^{-1/2}\exp\{-E_M(\mathbf{w}^{MP})\}. \quad (6\text{-}25)$$

In order to find the posterior density $P(\mathbf{w}|D, a, b, M)$ from Eq. (6-21), we need to find the variance parameters a and b. At this point we take another step in a hierarchical Bayesian analysis: We assume that a and b are random variables, with a given prior density. Apply Bayes's theorem again, we find

$$P(a, b|D, M) = \frac{P(D|a, b, M)P(a, b|M)}{P(D|M)}, \quad (6\text{-}26)$$

where the total probability $P(D|a, b, M)$ in Eq. (6-16) becomes the likelihood function in Eq. (6-26). If we assume a flat prior for a and b, then $P(a, b|M)$ is a constant. $P(D|M)$ is the normalization factor and not a function of a and b. Therefore, maximizing the posterior density $P(a, b|D, M)$ is equivalent to maximizing the likelihood function $P(D|a, b, M)$. From Eq. (6-16), we get

$$P(D|a, b, M) = \frac{P(D|\mathbf{w}, b, M)P(\mathbf{w}|a, M)}{P(\mathbf{w}|D, a, b, M)}. \tag{6-27}$$

By Eq. (6-17), (6-19), and (6-21), Eq. (6-27) can be expressed as

$$\begin{aligned} P(D|a, b, M) &= \frac{\frac{1}{z_D(b)}\exp(-bE_D)\left(\frac{1}{z_W(a)}\exp(-aE_W)\right)}{\frac{1}{z_M(a, b)}\exp\{-E_M\}} \\ &= \frac{z_M(a, b)}{z_D(b)z_W(a)} \end{aligned}. \tag{6-28}$$

In order to find the optimal value of a and b, which maximizes the posterior density, we can take the derivative of the log of the right hand side of the Eq. (6-28) and set it to zero, solving for a and b. The result is

$$a^{MP} = \frac{p^e}{2E_W(\mathbf{w}^{MP})}, \text{ and } b^{MP} = \frac{N - p^e}{2E_D(\mathbf{w}^{MP})}, \tag{6-29}$$

where

$$p^e = n_w - 2a^{MP}tr((\mathbf{H}^{MP})^{-1}) \tag{6-30}$$

is the effective number of parameters, which is a measure of how many parameters in the neural network can effectively reduce the error function E_M. The superscript MP refers to Most Probable value.

The Levenberg-Marquardt optimization algorithm is used in Eq. (6-15) to find the minimal point $\mathbf{w}^{MP}$ [30]. The Gauss-Newton approximation can be used to approximate $\mathbf{H}^{MP}$, that is

$$\mathbf{H} = \nabla^2 E_M(\mathbf{w}) \approx 2b\mathbf{J}^T\mathbf{J} + 2a\mathbf{I}, \tag{6-31}$$

where $\mathbf{J}$ is the Jacobian matrix of the training set errors, which is readily available in the Levenberg-Marquardt algorithm.

The GNBR algorithm can be summarized as follows:

1) Initialization: set $a = 0$, $b = 1$. Randomly initialize the weights $\mathbf{w}$.
2) Minimize the objective function Eq. (6-15). Only one step of the Levenberg-Marquardt algorithm is sufficient.
3) Compute $\mathbf{H}$ by Eq. (6-31). Compute p^e by Eq. (6-31).
4) Compute a and b by Eq. (6-29).
5) Repeat Step 2-4 until convergence.

In the next chapter, we will apply the idea of Hierarchical Bayesian regularization to the estimation of the covariance matrix.

CHAPTER 7

HIERARCHICAL BAYESIAN COVARIANCE MATRIX ESTIMATOR

In this chapter, we propose the Hierarchical Bayesian Covariance Matrix Estimator (HBCME). HBCME is inspired by the Bayesian Regularization for neural network training [45] [20] that we reviewed in Chapter 6.

There are three sections in this chapter, Section one is the assumptions made in order to develop the HBCME, in which only 2 unknown parameters in the prior covariance matrix are assumed. Section two presents detailed derivation of the estimator. In section three, we relax the assumptions made in Section one and develop the extended HBCME., in which $p + 1$ unknown parameters in the prior covariance matrix are assumed.

1. Assumptions

Let c_{ij}, $1 \le i, j \le p$, denote the elements of the covariance matrix $\mathbf{C}$. We make the following assumptions on the prior of the covariance matrix $\mathbf{C}$:

Assumption 1: The mean of the covariance matrix is $b\mathbf{I}$, where b is a positive real number.

We assume that the prior structure of the covariance matrix is a multiple of the identity matrix. This is equivalent to assuming that the variances of the elements of the measurements are equal and that the covariance between any two of the elements is zero.

Assumption 2: Each element c_{ij} has the same variance $\frac{1}{a}$.

Assumption 3: Each element of the upper triangle of the covariance matrix (including the diagonal elements), c_{ij} $(i \le j)$, is normal, independent, and identically distributed (i. i. d.).

Since the covariance matrix is symmetric, only the elements in the upper triangle of the covariance matrix (including the diagonal elements) are independent.

Based on these assumptions, the distribution of c_{ij} $(i \le j)$ is

$$c_{ij} \sim N\left(b\delta_{ij}, \frac{1}{a}\right) \tag{7-1}$$

where δ_{ij} is the delta function defined as

$$\delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \tag{7-2}$$

2. Derivation

Assume there are N i. i. d. observations $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$. Each observation is a $p \times 1$ vector with normal distribution $N(\mu, \mathbf{C})$ for $i = 1, 2, \ldots, N$, where μ is the mean vector and $\mathbf{C}$ is the covariance matrix.

Based on Bayes' theorem, after the data is taken, the posterior probability of the covariance matrix given the data is

$$P(\mathbf{C}|D, a, b, \mu) = \frac{P(D|\mathbf{C}, \mu) \cdot P(\mathbf{C}|a, b)}{P(D|a, b, \mu)} \tag{7-3}$$

where D denotes the data set $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$. In Eq. (7-3), $P(D|\mathbf{C}, \mu)$ is the likelihood

function, $P(\mathbf{C}|a, b)$ is the prior density function, $P(\mathbf{C}|D, a, b, \mu)$ is the posterior density function, and $P(D|a, b, \mu)$ is the total probability.

The essence of the HBCME is to find $\mathbf{C}$ which maximizes the posterior density function $P(\mathbf{C}|D, a, b, \mu)$. We first need to find the analytical function for $P(\mathbf{C}|D, a, b, \mu)$ in terms of D, a, b and μ before performing the optimization. Secondly, since a and b are unknown parameters, they need to be estimated from the data. We estimate a and b by maximizing the posterior density function of a and b through a second (hierarchical) level of Bayes's theorem. Thirdly, in performing the optimization on the posterior density function, we need to ensure the optimization is along the path in which $\mathbf{C}$ is positive definite. We will show our solution to this problem.

In the following, we first find the analytical function for the likelihood function, prior density function, posterior density function, and total probability density function. Then we present a method for estimating a and b. Later we perform the optimization on the posterior density function. Finally we discuss some computational issues which affect the speed of computation.

Likelihood function

Since $\mathbf{x}_i$ is i.i.d with normal distribution $N(\mu, \mathbf{C})$, the likelihood function $P(D|\mathbf{C}, \mu)$ can be expressed as

$$P(D|\mathbf{C},\mu) = \left(\prod_{i=1}^{N} \frac{1}{(2\pi)^{p/2}|\mathbf{C}|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}_i - \mu)^T \mathbf{C}^{-1}(\mathbf{x}_i - \mu) \right] \right)$$
$$= \frac{1}{(2\pi)^{pN/2}|\mathbf{C}|^{N/2}} \exp\left[-\sum_{i=1}^{N} \frac{1}{2}(\mathbf{x}_i - \mu)^T \mathbf{C}^{-1}(\mathbf{x}_i - \mu) \right] . \quad (7\text{-}4)$$

Define the data error

$$E_D(c_{ij}, \mu) = \sum_{i=1}^{N} \frac{1}{2}(\mathbf{x}_i - \mu)^T \mathbf{C}^{-1}(\mathbf{x}_i - \mu), \quad (7\text{-}5)$$

and

$$Z_D(c_{ij}) = (2\pi)^{\frac{pN}{2}} |\mathbf{C}|^{\frac{N}{2}}, \quad (7\text{-}6)$$

then the likelihood function of Eq. (7-4) can be expressed as

$$P(D|\mathbf{C},\mu) = \frac{\exp(-E_D)}{Z_D} . \quad (7\text{-}7)$$

Prior density function

By Assumptions 1, 2 and 3, the prior $P(\mathbf{C}|a,b)$ has normal distribution with

$$P(\mathbf{C}|a,b) = \left(\prod_{j=i}^{p} \prod_{i=1}^{p} \frac{1}{(2\pi)^{\frac{1}{2}}(a)^{-\frac{1}{2}}} \exp\left[-\frac{a}{2}(c_{ij} - b\delta_{ij})^2 \right] \right)$$
$$= \frac{1}{(2\pi)^{\frac{p(p+1)}{4}} (a)^{-\frac{p(p+1)}{4}}} \exp\left[-a \sum_{i=1}^{p} \sum_{j=i}^{p} \frac{1}{2}(c_{ij} - b\delta_{ij})^2 \right] . \quad (7\text{-}8)$$

Define the prior error E_C as

$$E_C(c_{ij}, b) = \sum_{j=i}^{p} \sum_{i=1}^{p} \frac{1}{2}(c_{ij} - b\delta_{ij})^2, \tag{7-9}$$

and

$$Z_C(a) = \left(\frac{2\pi}{a}\right)^{\frac{p(p+1)}{4}}, \tag{7-10}$$

then the prior $P(\mathbf{C}|a, b)$ in Eq. (7-8) can be written as

$$P(\mathbf{C}|a, b) = \frac{\exp(-aE_C)}{Z_C} \tag{7-11}$$

Posterior density function

From Eq. (7-3), Eq. (7-7) and Eq. (7-11), the posterior probability $P(\mathbf{C}|D, a, b, \mu)$ can be rewritten as

$$P(\mathbf{C}|D, a, b, \mu) = \frac{\exp(-E_M)}{Z_D \cdot Z_C \cdot P(D|a, b, \mu)}, \tag{7-12}$$

where

$$E_M = E_M(c_{ij}, a, b) = E_D(c_{ij}) + aE_C(c_{ij}, b), \tag{7-13}$$

and the subscript M stands for "Model", representing the total modeling error.

Total probability

The denominator in Eq. (7-3) (total probability) can be computed by integrating the numerator:

$$
\begin{aligned}
P(D|a, b, \mu) &= \int (P(D|\mathbf{C}, \mu) P(\mathbf{C}|a, b)) d\mathbf{C} \\
&= \frac{1}{Z_C} \int \frac{1}{Z_D} \cdot \exp(-E_M) d\mathbf{C} \\
&= \left(\frac{2\pi}{a}\right)^{-\frac{p(p+1)}{4}} (2\pi)^{-\frac{pN}{2}} \int |\mathbf{C}|^{-\frac{N}{2}} \exp(-E_M) d\mathbf{c}
\end{aligned}
\tag{7-14}
$$

In order to find the analytical expression for the total probability $P(D|a, b, \mu)$, we need to evaluate the integral in Eq. (7-14). Before we do the evaluation, we define the function F,

$$
F = \frac{N}{2} \log|\mathbf{C}| + E_M. \tag{7-15}
$$

Let $\mathbf{c}$ be a $\frac{p(p+1)}{2} \times 1$ vector with the following elements,

$$
\mathbf{c} = [c_{11}, c_{12}, \ldots, c_{1p}, c_{22}, c_{23}, \ldots, c_{2p}, \ldots, c_{pp}]^T. \tag{7-16}
$$

The Taylor expansion of F near the most probable value $\mathbf{c}^{MP}$ is

$$
F = F^{MP} + \frac{1}{2} (\mathbf{c} - \mathbf{c}^{MP})^T \mathbf{H}^{MP} (\mathbf{c} - \mathbf{c}^{MP}), \tag{7-17}
$$

where the superscript MP on F and $\mathbf{H}$ means they are evaluated at $\mathbf{c} = \mathbf{c}^{MP}$, i.e.

$F^{MP} = F|_{\mathbf{c} = \mathbf{c}^{MP}}$, and $\mathbf{H}^{MP} = \nabla_{\mathbf{c}}^2 F|_{\mathbf{c} = \mathbf{c}^{MP}}$. $\mathbf{H}^{MP}$ is a $\frac{p(p+1)}{2} \times \frac{p(p+1)}{2}$ matrix.

Now we can evaluate the integral in Eq. (7-14)

$$
\begin{aligned}
&\int |\mathbf{C}|^{-\frac{N}{2}} \exp(-E_M)\, d\mathbf{c} \\
&= \int \exp\left(-\frac{N}{2}\log|\mathbf{C}| - E_M\right) d\mathbf{c} \\
&= \exp(-F^{MP}) \int \frac{(2\pi)^{\frac{p(p+1)}{4}} \left|\mathbf{H}^{MP}\right|^{-\frac{1}{2}}}{(2\pi)^{\frac{p(p+1)}{4}} \left|\mathbf{H}^{MP}\right|^{-\frac{1}{2}}} \exp\left[-\frac{1}{2}(\mathbf{c}-\mathbf{c}^{MP})^T \mathbf{H}^{MP}(\mathbf{c}-\mathbf{c}^{MP})\right] d\mathbf{c} \\
&= \exp\left(-\frac{N}{2}\log\left|\mathbf{C}^{MP}\right| - E_M^{MP}\right)(2\pi)^{\frac{p(p+1)}{4}} \left|\mathbf{H}^{MP}\right|^{-\frac{1}{2}}
\end{aligned}
\tag{7-18}
$$

From Eq. (7-14) and Eq. (7-18), we have

$$
P(D|a,b,\mu) = \exp\left(-\frac{N}{2}\log\left|\mathbf{C}^{MP}\right| - E_M^{MP}\right)\left|\mathbf{H}^{MP}\right|^{-\frac{1}{2}} (2\pi)^{-\frac{pN}{2}} (a)^{\frac{p(p+1)}{4}} \tag{7-19}
$$

Estimating a and b

In order to find the parameters a and b, we apply the Bayes' rule to the total probability,

$$
P(a,b|D,\mu) = \frac{P(D|a,b,\mu)\cdot P(a,b)}{P(D|\mu)}, \tag{7-20}
$$

If we assume a flat prior distribution for a and b (i.e, $P(a,b)$ is a constant), we have

$$
P(a,b|D,\mu) \propto P(D|a,b,\mu). \tag{7-21}
$$

Maximizing the posterior density function $P(a,b|D,\mu)$ is equivalent to maximizing the likelihood function $P(D|a,b,\mu)$ in Eq. (7-20), which is also the total probability in Eq. (7-3), as expressed in Eq. (7-19).

We can take the log of the right hand side of Eq. (7-19), take the derivative with respect to a and b, set both derivatives to zero, and solve for a and b.

From Eq. (7-19)

$$\begin{aligned}
&\log P(D|a,b,\mu) \\
&= -\frac{N}{2}\log|\mathbf{C}^{MP}| - E_D^{MP} - \alpha E_C^{MP} - \frac{1}{2}\log|\mathbf{H}^{MP}| \\
&\quad -\frac{pN}{2}\log 2\pi + \frac{p(p+1)}{4}\log a
\end{aligned} \tag{7-22}$$

Take the derivative of $\log P(D|\alpha, \beta, \mu)$ with respect to a and set the result to 0:

$$\frac{\partial \log P(D|a,b,\mu)}{\partial a} = -E_C^{MP} - \frac{1}{2}\frac{\partial \log|\mathbf{H}^{MP}|}{\partial a} + \frac{p(p+1)}{4a} = 0. \tag{7-23}$$

The only difficulty is to evaluate the second term on the right hand side of Eq. (7-23), $\frac{\partial \log|\mathbf{H}^{MP}|}{\partial a}$. Notice that

$$\mathbf{H}^{MP} = \frac{N}{2}\nabla_{\mathbf{c}}^2 \log|\mathbf{C}^{MP}| + \nabla_{\mathbf{c}}^2 E_D^{MP} + a\nabla_{\mathbf{c}}^2 E_C^{MP}. \tag{7-24}$$

Therefore, we have

$$\begin{aligned}
\frac{d\mathbf{H}^{MP}}{da} &= \nabla_{\mathbf{c}}^2 E_C^{MP} \\
&= \frac{\partial^2}{\partial c_{ij}^2} \sum_{j=ii=1}^{p}\sum^{p} \frac{1}{2}(c_{ij} - b\delta_{ij})^2 \Big|_{\mathbf{c}=\mathbf{c}^{MP}} \\
&= \frac{\partial}{\partial c_{ij}} \sum_{j=ii=1}^{p}\sum^{p} (c_{ij} - b\delta_{ij}) \Big|_{\mathbf{c}=\mathbf{c}^{MP}} \\
&= (\mathbf{I})_{\frac{p(p+1)}{2}}
\end{aligned} \tag{7-25}$$

where $(\mathbf{I})_{\frac{p(p+1)}{2}}$ is the identity matrix with dimension $\frac{p(p+1)}{2}$. Therefore, using some theorems for matrix derivatives (see Appendix),

$$
\begin{aligned}
\frac{\partial \log|\mathbf{H}^{MP}|}{\partial a} &= \frac{\partial \log|\mathbf{H}|}{\partial (vec\mathbf{H})^T} \cdot \frac{\partial (vec\mathbf{H})^T}{\partial a} \\
&= \left(vec\frac{\partial \log|\mathbf{H}|}{\partial \mathbf{H}} \cdot vec\mathbf{I} \right)\Big|_{\mathbf{c}=\mathbf{c}^{MP}} \\
&= tr\left(\frac{\partial \log|\mathbf{H}|}{\partial \mathbf{H}} \right)\Big|_{\mathbf{c}=\mathbf{c}^{MP}} \\
&= tr[(2(\mathbf{H}^{MP}))^{-1} - diag((\mathbf{H}^{MP})^{-1})] \\
&= tr[(\mathbf{H}^{MP})^{-1}]
\end{aligned}
, \tag{7-26}
$$

where $diag((\mathbf{H}^{MP})^{-1})$ is the diagonal matrix formed by taking the diagonal elements of $(\mathbf{H}^{MP})^{-1}$. Caution must be taken to insure that $\mathbf{H}$ is symmetric. In the above derivation, we used *Theorems 2* and *3* in Appendix.

Solve Eq. (7-23) for a:

$$
a = \frac{p(p+1)}{2tr[(\mathbf{H}^{MP})^{-1}] + 4E_C^{MP}}. \tag{7-27}
$$

Similarly, take the derivative of $\log P(D|a, b, \mu)$ with respect to b, set the result to 0, and solve for b,

$$
\begin{aligned}
\frac{\partial \log P(D|a, b, \mu)}{\partial b} &= -a\frac{\partial E_C^{MP}}{\partial b} \\
&= -a\frac{\partial}{\partial b}\sum_{i=1}^{p}\frac{1}{2}(c_{ii}-b)^2\Big|_{\mathbf{c}=\mathbf{c}^{MP}} \\
&= a\sum_{i=1}^{p}(c_{ii}-b)\Big|_{\mathbf{c}=\mathbf{c}^{MP}} \\
&= a(pb - tr(\mathbf{C}^{MP})) \\
&= 0
\end{aligned}
\tag{7-28}
$$

$$b = \frac{tr(\mathbf{C}^{MP})}{p} \tag{7-29}$$

From Eq. (7-29), we can see that b is the average of the diagonal elements of the covariance matrix.

There is still one problem left in estimating a. From Eq. (7-27), we can see that $\mathbf{H}^{MP}$ is needed to calculate a. Numerical methods for calculating the second derivative should be avoided if possible, since they can induce large numerical errors. Later we present a method that can calculate the gradient analytically. Then numerical differentiation can be used on the gradient to calculate the second derivatives.

Estimating $\mathbf{C}^{MP}$

We have found analytical forms for the likelihood function $P(D|\mathbf{C}, \mu)$, the prior density function $P(\mathbf{C}|a, b)$, and the equation for estimating a and b in Eq. (7-3). (The estimation for the total probability density function $P(D|a, b, \mu)$ was used for estimating a and b, but was not directly used for evaluating the posterior density function.) It is time to perform the optimization on the posterior density function $P(\mathbf{C}|D, a, b, \mu)$.

From Eq. (7-3), the posterior density function $P(\mathbf{C}|D, a, b, \mu)$ is

$$\begin{aligned} P(\mathbf{C}|D, a, b, \mu) &\propto P(D|\mathbf{C}, \mu)P(\mathbf{C}|a, b) \\ &= \frac{1}{(2\pi)^{\frac{pN}{2}}|\mathbf{C}|^{\frac{N}{2}}} \exp\left[-\sum_{i=1}^{N} \frac{1}{2}(\mathbf{x}_i - \mu)^T \mathbf{C}^{-1}(\mathbf{x}_i - \mu)\right] \\ &\quad \times \frac{1}{\left(\frac{2\pi}{a}\right)^{\frac{p(p+1)}{4}}} \exp\left[-a\sum_{j=i}^{p}\sum_{i=1}^{p} \frac{1}{2}(c_{ij} - b\delta_{ij})^2\right] \end{aligned} \tag{7-30}$$

Maximizing $P(\mathbf{C}|D, a, b, \mu)$ is equivalent to minimizing the following objective function,

$$\min_{c_{ij}} J = \sum_{i=1}^{N} \frac{1}{2}(\mathbf{x}_i - \mu)^T \mathbf{C}^{-1}(\mathbf{x}_i - \mu) + a \sum_{j=i}^{p} \sum_{i=1}^{p} \frac{1}{2}(c_{ij} - b\delta_{ij})^2 + \frac{N}{2}\log|\mathbf{C}| \qquad (7\text{-}31)$$

The BFGS algorithm [21] with backtracking line search [11] is used for finding c_{ij}. However, if we directly optimize $\mathbf{c}$, it is not guaranteed that $\mathbf{C}$ is positive definite. Therefore, in order to keep $\mathbf{C}$ positive definite, we decompose $\mathbf{C} = \mathbf{Y}\mathbf{Y}^T$ (any matrix decomposition method will work, for example, Cholesky Factorization [48]). Instead of trying to find c_{ij}, we will find y_{ij}.

Let $\mathbf{y}$ be a $p^2 \times 1$ vector defined as $\mathbf{y} = vec\mathbf{Y}$. The BFGS algorithm requires the gradient of the objective function $\nabla_{\mathbf{y}} J$. This gradient can be computed from Eq. (7-31) as follows

$$\begin{aligned} \nabla_{\mathbf{y}} J &= \nabla_{\mathbf{y}} E_D + a\nabla_{\mathbf{y}} E_C + \frac{N}{2}\nabla_{\mathbf{y}}(\log|\mathbf{C}|) \\ &= \left(\frac{\partial E_D}{\partial (vec\mathbf{C})^T} + a\frac{\partial E_C}{\partial (vec\mathbf{C})^T} + \frac{N}{2}\frac{\partial \log|\mathbf{C}|}{\partial (vec\mathbf{C})^T} \right) \frac{\partial vec\mathbf{C}}{\partial \mathbf{y}^T}. \end{aligned} \qquad (7\text{-}32)$$

There are 4 terms that need to be estimated, namely $\frac{\partial E_D}{\partial (vec\mathbf{C})^T}$, $\frac{\partial E_C}{\partial (vec\mathbf{C})^T}$, $\frac{\partial \log|\mathbf{C}|}{\partial (vec\mathbf{C})^T}$, and $\frac{\partial vec\mathbf{C}}{\partial \mathbf{y}^T}$. In estimating these terms, caution must be taken, since $\mathbf{C}$ is symmetric. Some matrix derivative theorems can be found in Appendix. We use these theorems to derive the expressions for each term in Eq. (7-32).

First, we compute $\frac{\partial E_D}{\partial (vec\mathbf{C})^T}$. Since $\frac{\partial E_D}{\partial (vec\mathbf{C})^T} = \left(vec\frac{\partial E_D}{\partial \mathbf{C}}\right)^T$, we only need to estimate $\frac{\partial E_D}{\partial \mathbf{C}}$. By *Theorem 1* and *3* in the Appendix, if we let

$$\begin{aligned}\mathbf{M}_1 &= \sum_{i=1}^{N} \frac{1}{2}[-\mathbf{C}^{-1}(\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T\mathbf{C}^{-1}] \\ &= -\frac{1}{2}\mathbf{C}^{-1}\left[\sum_{i=1}^{N}(\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T\right]\mathbf{C}^{-1}, \\ &= -\frac{1}{2}\mathbf{C}^{-1}\mathbf{X}^T\mathbf{X}\mathbf{C}^{-1}\end{aligned} \tag{7-33}$$

then

$$\frac{\partial E_D}{\partial \mathbf{C}} = 2\mathbf{M}_1 - diag(\mathbf{M}_1), \tag{7-34}$$

where $\mathbf{X}$ is an $N \times p$ matrix with each row containing the observation data.

Next we evaluate the second term in Eq. (7-32), $\frac{\partial E_C}{\partial (vec\mathbf{C})^T}$. Similar to the first case, since $\frac{\partial E_C}{\partial (vec\mathbf{C})^T} = \left(vec\frac{\partial E_C}{\partial \mathbf{C}}\right)^T$, we only need to estimate $\frac{\partial E_C}{\partial \mathbf{C}}$. In fact,

$$\frac{\partial E_C}{\partial \mathbf{C}} = \begin{bmatrix} \frac{\partial E_C}{\partial c_{11}} & \frac{\partial E_C}{\partial c_{12}} & \cdots & \frac{\partial E_C}{\partial c_{1p}} \\ \frac{\partial E_C}{\partial c_{21}} & \frac{\partial E_C}{\partial c_{22}} & \cdots & \frac{\partial E_C}{\partial c_{2p}} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial E_C}{\partial c_{p1}} & \frac{\partial E_C}{\partial c_{p2}} & \cdots & \frac{\partial E_C}{\partial c_{pp}} \end{bmatrix} = \mathbf{C} - \beta\mathbf{I}, \tag{7-35}$$

Now we evaluate the third term in Eq. (7-32), $\frac{\partial \log|\mathbf{C}|}{\partial (vec\mathbf{C})^T}$. Similar to the cases before, since $\frac{\partial \log|\mathbf{C}|}{\partial (vec\mathbf{C})^T} = \left(vec\frac{\partial \log|\mathbf{C}|}{\partial \mathbf{C}}\right)^T$, we only need to estimate $\frac{\partial \log|\mathbf{C}|}{\partial \mathbf{C}}$. From *Theorems 2* and *3* in Appendix,

$$\frac{\partial \log|\mathbf{C}|}{\partial \mathbf{C}} = 2\mathbf{C}^{-1} - diag(\mathbf{C}^{-1}). \tag{7-36}$$

Finally we evaluate the last term in Eq. (7-32), $\frac{\partial vec\mathbf{C}}{\partial \mathbf{y}^T}$. Since

$$\mathbf{C} = \mathbf{Y}\mathbf{Y}^T = \begin{bmatrix} \sum_{i=1}^{p} y_{1i}^2 & \sum_{i=1}^{p} y_{1i}y_{2i} & \cdots & \sum_{i=1}^{p} y_{1i}y_{pi} \\ \sum_{i=1}^{p} y_{2i}y_{1i} & \sum_{i=1}^{p} y_{2i}^2 & \cdots & \sum_{i=1}^{p} y_{2i}y_{pi} \\ \cdots & \cdots & & \cdots \\ \sum_{i=1}^{p} y_{pi}y_{1i} & \sum_{i=1}^{p} y_{pi}y_{2i} & \cdots & \sum_{i=1}^{p} y_{pi}^2 \end{bmatrix} \tag{7-37}$$

From Eq. (7-37), we find the $p^2 \times p^2$ Jacobian matrix in Eq. (7-38). The rows in Eq. (7-38) are $\frac{\partial c_{11}}{\partial \mathbf{y}^T}, \frac{\partial c_{21}}{\partial \mathbf{y}^T}, \ldots, \frac{\partial c_{p1}}{\partial \mathbf{y}^T}, \frac{\partial c_{12}}{\partial \mathbf{y}^T}, \frac{\partial c_{22}}{\partial \mathbf{y}^T}, \ldots, \frac{\partial c_{p2}}{\partial \mathbf{y}^T}, \ldots, \frac{\partial c_{p1}}{\partial \mathbf{y}^T}, \frac{\partial c_{p2}}{\partial \mathbf{y}^T}, \ldots, \frac{\partial c_{pp}}{\partial \mathbf{y}^T}$. The columns in Eq. (7-38) are $\frac{\partial vec\mathbf{C}}{\partial y_{11}}, \frac{\partial vec\mathbf{C}}{\partial y_{21}}, \ldots, \frac{\partial vec\mathbf{C}}{\partial y_{p1}}, \frac{\partial vec\mathbf{C}}{\partial y_{12}}, \frac{\partial vec\mathbf{C}}{\partial y_{22}}, \ldots, \frac{\partial vec\mathbf{C}}{\partial y_{p2}}, \ldots,$

$\frac{\partial vec\mathbf{C}}{\partial y_{1p}}, \frac{\partial vec\mathbf{C}}{\partial y_{2p}}, \ldots, \frac{\partial vec\mathbf{C}}{\partial y_{pp}}$.

$$\frac{\partial vec\mathbf{C}}{\partial \mathbf{y}^T} = \begin{bmatrix} 2y_{11} & 0 & \cdots & 0 & 2y_{12} & 0 & \cdots & 0 & \cdots & 2y_{1p} & 0 & \cdots & 0 \\ y_{21} & y_{11} & \cdots & 0 & y_{22} & y_{12} & \cdots & 0 & \cdots & y_{2p} & y_{1p} & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ y_{p1} & 0 & \cdots & y_{11} & y_{p2} & 0 & \cdots & y_{12} & \cdots & y_{pp} & 0 & \cdots & y_{1p} \\ y_{21} & y_{11} & \cdots & 0 & y_{22} & y_{12} & \cdots & 0 & \cdots & y_{2p} & y_{1p} & \cdots & 0 \\ 0 & 2y_{21} & \cdots & 0 & 0 & 2y_{22} & \cdots & 0 & \cdots & 0 & 2y_{2p} & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & y_{p1} & \cdots & y_{21} & 0 & y_{p2} & \cdots & y_{22} & \cdots & 0 & y_{pp} & \cdots & y_{2p} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ y_{p1} & 0 & \cdots & y_{11} & y_{p2} & 0 & \cdots & y_{12} & \cdots & y_{pp} & 0 & \cdots & y_{1p} \\ 0 & y_{p1} & \cdots & y_{21} & 0 & y_{p2} & \cdots & y_{22} & \cdots & 0 & y_{pp} & \cdots & y_{2p} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 2y_{p1} & 0 & 0 & \cdots & 2y_{p2} & \cdots & 0 & 0 & \cdots & 2y_{pp} \end{bmatrix} \tag{7-38}$$

To summarize, in this section we have developed a procedure for computing $\mathbf{C}^{MP}$. The idea is to use the BFGS algorithm, with gradient computed using Eq. (7-32). The four individual terms in Eq. (7-32) are computed using Eq. (7-34), Eq. (7-35), Eq. (7-36), and Eq. (7-38).

We still need to estimate the parameter a, which requires the Hessian matrix $\mathbf{H}^{MP}$, as in Eq. (7-27). The computation of $\mathbf{H}^{MP}$ is described in the next section.

Computing $\mathbf{H}^{MP}$

Previously we mentioned that the Hessian $\mathbf{H}^{MP}$ from Eq. (7-24) will be inaccurate if it is calculated numerically. To improve the accuracy, we will first calculate the gradient analytically, then we will apply numerical differentiation to the gradient to compute $\mathbf{H}^{MP}$.

First we calculate the gradient of the objective function J as

$$\nabla_{\mathbf{C}} J = \frac{\partial E_D}{\partial \mathbf{C}} + a\frac{\partial E_C}{\partial \mathbf{C}} + \frac{N}{2}\frac{\partial \log|\mathbf{C}|}{\partial \mathbf{C}}. \tag{7-39}$$

From Eq. (7-34), (7-35), (7-36) and (7-39), we can compute $\nabla_{\mathbf{C}} J$. By picking up the corresponding elements of $\nabla_{\mathbf{C}} J$, we can obtain $\nabla_{\mathbf{c}} J$. Then numerical differentiation can be used to compute $\mathbf{H}^{MP}$.

Caution must be taken in the calculation of $\mathbf{H}^{MP}$. Notice that $\mathbf{H}^{MP}$ is defined by Eq. (7-24), which is the Hessian with respect to $\mathbf{c}$. The BFGS algorithm calculates a Hessian as a by-product. However, the BFGS algorithm calculates $\nabla^2_{\mathbf{y}} J$ instead of $\nabla^2_{\mathbf{c}} J$, so we can't use the BFGS computations for calculating $\mathbf{H}^{MP}$.

Other Computational Issues

In MATLAB, in order to speed up calculations, we can use matrix operations instead of the `for` loops. The strategy is applied to the calculation of E_D and E_C.

The data error E_D can be calculated as follows,

$$\begin{aligned} E_D &= \sum_{i=1}^{N} \frac{1}{2}(\mathbf{x}_i - \mu)^T \mathbf{C}^{-1}(\mathbf{x}_i - \mu) \\ &= \sum_{i=1}^{N} \frac{1}{2} tr[\mathbf{C}^{-1}((\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T)] \\ &= tr\left[\frac{1}{2}\mathbf{C}^{-1}\sum_{i=1}^{N}(\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T\right] \\ &= tr\left(\frac{\mathbf{C}^{-1}\mathbf{B}}{2}\right) \end{aligned}, \tag{7-40}$$

where

$$\mathbf{B} = \sum_{i=1}^{N} (\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T$$
$$= \left(\mathbf{X} - \begin{bmatrix} \mu^T \\ \mu^T \\ \cdots \\ \mu^T \end{bmatrix} \right)^T \left(\mathbf{X} - \begin{bmatrix} \mu^T \\ \mu^T \\ \cdots \\ \mu^T \end{bmatrix} \right), \tag{7-41}$$

The structure error E_C is in fact one half of the sum of the squares of the elements in the upper triangle of the difference (including the diagonal elements) between $\mathbf{C}$ and $b\mathbf{I}$, therefore

$$E_C = \frac{1}{2} sum(sum(triu[(\mathbf{C} - b\mathbf{I}).*(\mathbf{C} - b\mathbf{I})])), \tag{7-42}$$

where $triu$ is a function that sets all the lower triangular elements (not including the diagonal elements) to 0, ".*" is element by element matrix multiplication.

Summary of the Algorithm

Now we summarize the HBCME algorithm:

0) Initialize a and b to random initial values.

1) Find $\mathbf{C}^{MP}$ through $\mathbf{Y}^{MP}$, where $\mathbf{C}^{MP} = (\mathbf{Y}^{MP})(\mathbf{Y}^{MP})^T$. $\mathbf{Y}^{MP}$ is found by minimizing the following objective function:

$$\min_{y_{ij}} J = \sum_{i=1}^{N} \frac{1}{2} (\mathbf{x}_i - \mu)^T (\mathbf{Y}\mathbf{Y}^T)^{-1} (\mathbf{x}_i - \mu) + a \sum_{i=1}^{p} \sum_{j=i}^{p} \frac{1}{2} (c_{ij} - b\delta_{ij})^2 + \frac{N}{2} \log |\mathbf{Y}\mathbf{Y}^T|, \tag{7-43}$$

where c_{ij} is the i th row and j th column element of $\mathbf{YY}^T$. The optimization is performed by the BFGS algorithm with backtracking line search. The gradient is calculated using Eq. (7-32).

2) Compute E_C^{MP} using Eq. (7-9):

$$E_C(c_{ij}, b) = \sum_{i=1}^{p} \sum_{j=i}^{p} \frac{1}{2}(c_{ij} - b\delta_{ij})^2. \tag{7-44}$$

3) Compute $\mathbf{H}^{MP}$ using Eq. (7-39) and numerical differentiation.

4) Compute a and b using Eq. (7-27) and Eq. (7-29):

$$a = \frac{p(p+1)}{2tr[(\mathbf{H}^{MP})^{-1}] + 4E_C^{MP}} \tag{7-45}$$

$$b = \frac{tr(\mathbf{C}^{MP})}{p} \tag{7-46}$$

5) Iterate steps 1) through 4) until convergence.

3. Extension to $p+1$ unknown parameters

In this section, we relax *assumption 1* to the following assumption:

Assumption 1': The prior mean of diagonal elements of the covariance matrix is $\mathbf{b}$.

We assume that the means of the diagonal elements are not neccessarily equal. In essence, we are assuming the prior structure of the covariance matrix is a diagonal matrix, but is not a multiple of the identity matrix.

According to this new assumption, we need to modify some of the derivations presented earlier in this chapter. The changes are as follows:

- The prior distribution of c_{ij} in Eq. (7-1) changes to

$$c_{ij} \sim N\left(b_i \delta_{ij}, \frac{1}{a}\right), \tag{7-47}$$

where b_i is the ith element of $\mathbf{b}$.

- The prior density $P(\mathbf{C}|a, b)$ in Eq. (7-8) changes to

$$\begin{aligned} P(\mathbf{C}|a, \mathbf{b}) &= \left(\prod_{j=i}^{p} \prod_{i=1}^{p} \frac{1}{(2\pi)^{\frac{1}{2}} (a)^{-\frac{1}{2}}} \exp\left[-\frac{a}{2} (c_{ij} - b_i \delta_{ij})^2 \right] \right) \\ &= \frac{1}{(2\pi)^{\frac{p(p+1)}{4}} (a)^{-\frac{p(p+1)}{4}}} \exp\left[-a \sum_{j=i}^{p} \sum_{i=1}^{p} \frac{1}{2} (c_{ij} - b_i \delta_{ij})^2 \right]. \end{aligned} \tag{7-48}$$

- The prior error E_C in Eq. (7-9) changes to

$$E_C(c_{ij}, \mathbf{b}) = \sum_{j=i}^{p} \sum_{i=1}^{p} \frac{1}{2} (c_{ij} - b_i \delta_{ij})^2 . \tag{7-49}$$

- The derivative of $\log P(D|a, \mathbf{b}, \mu)$ with respect to b_i in Eq. (7-28) changes to

$$\begin{aligned} \frac{\partial \log P(D|a, \mathbf{b}, \mu)}{\partial b_i} &= -a \frac{\partial E_C^{MP}}{\partial b_i} \\ &= -a \frac{\partial}{\partial b_i} \sum_{i=1}^{p} \frac{1}{2} (c_{ii} - b_i)^2 \Big|_{\mathbf{c} = \mathbf{c}^{MP}} \\ &= a(c_{ii} - b_i) \Big|_{\mathbf{c} = \mathbf{c}^{MP}} \\ &= a(c_{ii}^{MP} - b_i) \\ &= 0 \end{aligned} \tag{7-50}$$

The estimate of b_i, by solving Eq. (7-50), is

$$b_i = c_{ii}^{MP} . \tag{7-51}$$

It is reasonable that the value of b_i equals the corresponding diagonal element of the covariance matrix.

- The posterior density function $P(\mathbf{C}|D, a, \mathbf{b}, \mu)$ in Eq. (7-30) changes to

$$P(\mathbf{C}|D, a, \mathbf{b}, \mu) \propto P(D|\mathbf{C}, \mu)P(\mathbf{C}|a, \mathbf{b})$$

$$= \frac{1}{(2\pi)^{\frac{pN}{2}}|\mathbf{C}|^{\frac{N}{2}}} \exp\left[-\sum_{i=1}^{N} \frac{1}{2}(\mathbf{x}_i - \mu)^T \mathbf{C}^{-1}(\mathbf{x}_i - \mu)\right] \cdot \frac{1}{\left(\frac{2\pi}{\alpha}\right)^{\frac{p(p+1)}{4}}} \exp\left[-a\sum_{j=ii=1}^{p}\sum^{p} \frac{1}{2}(c_{ij} - b_i\delta_{ij})^2\right] \tag{7-52}$$

- The objective function in Eq. (7-31) changes to

$$\min_{c_{ij}} J = \sum_{i=1}^{N} \frac{1}{2}(\mathbf{x}_i - \mu)^T \mathbf{C}^{-1}(\mathbf{x}_i - \mu) + a\sum_{j=ii=1}^{p}\sum^{p} \frac{1}{2}(c_{ij} - b_i\delta_{ij})^2 + \frac{N}{2}\log|\mathbf{C}| . \tag{7-53}$$

- Eq. (7-35) changes to

$$\mathbf{M}_2 = \mathbf{C} - diag(\mathbf{b}) . \tag{7-54}$$

The above list does not include expressions which require only the change of b to $\mathbf{b}$, as in Eq. (7-11), (7-12), (7-13), (7-14), (7-19), (7-20), (7-21), (7-22), (7-23), etc.

The HBCME algorithm for $p + 1$ unknown parameters is summarized as follows:

0) Initialize a, $\mathbf{b}$ to random values.

1) Find $\mathbf{C}^{MP}$ through $\mathbf{Y}^{MP}$, where $\mathbf{C}^{MP} = (\mathbf{Y}^{MP})(\mathbf{Y}^{MP})^T$. $\mathbf{Y}^{MP}$ is found by minimizing the following objective function:

$$\min_{y_{ij}} J = \sum_{i=1}^{N} \frac{1}{2}(\mathbf{x}_i - \mu)^T (\mathbf{Y}\mathbf{Y}^T)^{-1}(\mathbf{x}_i - \mu) + a\sum_{i=1}^{p}\sum_{j=i}^{p} \frac{1}{2}(c_{ij} - b_i\delta_{ij})^2 + \frac{N}{2}\log|\mathbf{Y}\mathbf{Y}^T| \quad , \tag{7-55}$$

where c_{ij} is the i th row and j th column element of $\mathbf{Y}\mathbf{Y}^T$. The optimization is performed by the BFGS algorithm with backtracking line search. The gradient is calculated using Eq. (7-32).

2') Compute E_C^{MP} using Eq. (7-49):

$$E_C(c_{ij}, \mathbf{b}) = \sum_{i=1}^{p}\sum_{j=i}^{p} \frac{1}{2}(c_{ij} - b_i\delta_{ij})^2 . \tag{7-56}$$

3') Compute $\mathbf{H}^{MP}$ using Eq. (7-39) and numerical differentiation.

4') Compute a and $\mathbf{b}$ using Eq. (7-27) and Eq. (7-51):

$$a = \frac{p(p+1)}{2tr[(\mathbf{H}^{MP})^{-1}] + 4E_C^{MP}}, \tag{7-57}$$

$$b_i = c_{ii}^{MP} . \tag{7-58}$$

5') Iterate steps 1') through 4') until convergence.

This completes the development of the HBCME. In practice, if the sample data are normalized, the HBCME with 2 unknown parameters is sufficient. Otherwise, the HBCME with $p + 1$ parameter is suggested. In Chapter 8, we will present the simulation results of the HBCME algorithm.

CHAPTER 8

SIMULATION RESULTS: HIERARCHICAL BAYESIAN COVARIANCE MATRIX ESTIMATOR

In this chapter, we present the simulation results for the Hierarchical Bayesian Covariance Matrix Estimator (HBCME), both in the basic form, in which 2 unknown parameters of the prior covariance matrix are assumed. and in the extended form, in which $p + 1$ unknown parameters of the prior covariance matrix are assumed.

There are two sections in this chapter. In section one, we compare the performance of the covariance matrix estimated by the basic HBCME to the sample covariance matrix when the true covariance matrix is a multiple of the identity matrix. We study the effect of the standard deviation of c_{ij}, the sample size N, and the structure of the covariance matrix. In section two, we compare the performances of the covariance matrices estimated by the basic HBCME and the extended HBCME to the sample covariance matrix. In this section, we assume that the true covariance matrix is a diagonal matrix instead of a multiple of the identity matrix.

We denote the covariance matrix estimated by the basic HBCME as $\hat{\mathbf{C}}_B^2$, and the covariance matrix estimated by the extended HBCME as $\hat{\mathbf{C}}_B^{p+1}$.

1. Monte Carlo Simulation: basic HBCME

In this section, Monte Carlo simulation is carried out to compare the sample covariance matrix, $\mathbf{S}$, to the covariance matrix estimated by the HBCME, $\hat{\mathbf{C}}_B^2$. The true covariance matrix is assumed to be a multiple of the identity matrix.

We vary three variables in the true covariance matrix to study their effects on the estimation errors. The first is σ, which is the standard deviation of c_{ij}. The second is the sample size N. The third is n_c, which is the number of samples used to generate the true covariance matrix. n_c is a structure parameter, which measures the closeness of the structure of the true covariance matrix to that of the identity matrix. The larger the value of n_c, the closer the structure of the true covariance matrix is to the identity matrix.

The meaning of n_c can be better understood if we consider how data is generated for the Monte Carlo experiments. For each set of Monte Carlo simulations, we first generate a mean and a covariance matrix at random. The procedure is to first generate n_c random vectors, where each element is independent with distribution $N(0, 1)$. The sample mean of these n_c vectors becomes the true mean, and the sample covariance matrix becomes the true covariance matrix. Then the true mean and covariance matrix are used to generate data for the experiment.

In the following, we assume the covariance matrix dimension is 3×3.

The Monte Carlo simulation has the following steps:

1) Initialization: Set the parameters to their nominal values $\sigma = 10$, $N = 20$, $n_c = 20$.

2) Generating Data: First generate a data set with n_c samples. Each sample has standard normal distribution $N(0, 1)$ with dimension 3×1. Calculate the sample mean $\bar{\mathbf{x}}^o$ and the sample covariance matrix $\mathbf{S}^o$ from these n_c samples. Set $\mu = \bar{\mathbf{x}}^o$ and $\mathbf{C} = \mathbf{S}^o$. Generate a sample data set with N samples, which has normal distribution $N(\mu, \sigma\mathbf{C})$.

3) Calculate the sample covariance matrix, $\mathbf{S}$, and the covariance estimated by the HBCME, $\hat{\mathbf{C}}_B^2$. Calculate E_D, E_C, a, and b, which are the outputs of the HBCME.

4) Calculate the estimation errors for $\mathbf{S}$ and $\hat{\mathbf{C}}_B^2$. The errors are defined by the Frobenius norm as $\|\mathbf{S} - \mathbf{C}\|$ and $\left\|\hat{\mathbf{C}}_B^2 - \mathbf{C}\right\|$, respectively.

5) Repeat Steps 2-4 100 times. Calculate the mean and standard deviation of the errors of $\mathbf{S}$ and $\hat{\mathbf{C}}_B^2$, and the mean of E_D, E_C, a, b.

6) Effect of standard deviation σ: Set the variance $\sigma = 1, 3, \ldots, 99$, with $N = 20$ and $n_c = 20$. Repeat Steps 2-5.

7) Effect of sample size N: Set the sample size $N = 10, 12, \ldots, 100$, with $\sigma = 10$ and $n_c = 20$. Repeat Steps 2-5.

8) Effect of structure parameter n_c: Set $n_c = 10, 12, \ldots, 100$, with $\sigma = 10$ and $N = 20$. Repeat Steps 2-5.

From step 2, we can see the role of n_c. If n_c gets larger, the absolute value of the off-diagonal elements of the true covariance matrix $\mathbf{C}$ are closer to 0. Meanwhile, the diagonal elements of $\mathbf{C}$ are always close to 1. Therefore, n_c is a measure of how close the true covariance matrix is to the identity matrix.

Now we present the simulation results, studying the effects of the parameters σ, N, and n_c on the estimation errors for $\mathbf{C}$ and on the value of E_C, E_D, a, and b.

Effect of standard deviation σ

In this section, we study the how the standard deviation σ affects the covariance estimation errors, and how E_C, E_D, a and b change when σ increases. The standard deviation σ takes on the values 1, 3, ..., 99, while N and n_c remain at their nominal values, i.e., $N = 20$ and $n_c = 20$.

Figure 8-1 shows the estimation errors and the error standard deviations for $\mathbf{S}$ and $\hat{\mathbf{C}}_B^2$. Error is defined by Frobenius norm as $\|\mathbf{S} - \mathbf{C}\|$ for $\mathbf{S}$, and $\left\|\hat{\mathbf{C}}_B^2 - \mathbf{C}\right\|$ for $\hat{\mathbf{C}}_B^2$. The solid line is the error of $\mathbf{S}$. The two dotted lines are one standard deviation above and below the error of $\mathbf{S}$. The dashed line is the error of $\hat{\mathbf{C}}_B^2$. The two dash dotted lines are one standard deviation above and below the error of $\hat{\mathbf{C}}_B^2$.

From Figure 8-1, we make the following conclusions. First consider the effect of standard deviation σ. The larger the standard deviation σ, the larger the error of the covariance matrices. The relation is linear. The error we calculated is an estimate of the Root

Mean Square (RMS) error, $E[\|\sigma\hat{\mathbf{C}} - \sigma\mathbf{C}\|]$, where $\hat{\mathbf{C}}$ can be $\mathbf{S}$ or $\hat{\mathbf{C}}_B^2$. This is because we take an average of 100 iterations of $\|\sigma\hat{\mathbf{C}} - \sigma\mathbf{C}\|$. Because we have

$$E[\|\sigma\hat{\mathbf{C}} - \sigma\mathbf{C}\|] = \sigma E[\|\hat{\mathbf{C}} - \mathbf{C}\|] \tag{8-1}$$

the relationship between the RMS error and σ is linear. The larger σ, the larger the error.

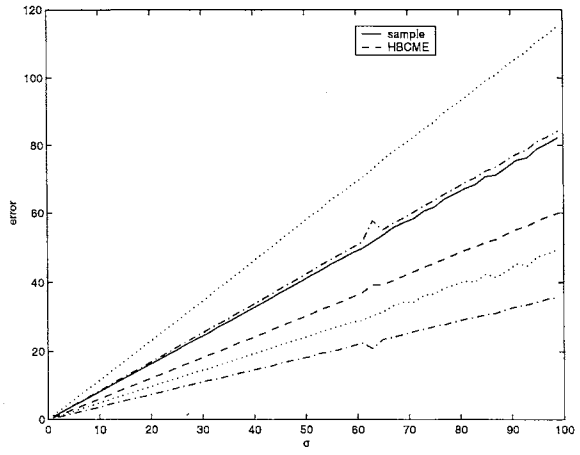


Figure 8-1 RMS estimation error vs. σ (covariance structure: multiple of identity)

The second conclusion is that the estimation error of $\hat{\mathbf{C}}_B$ is smaller than that of $\mathbf{S}$. This means that the covariance estimated by HBCME is better than the sample covariance matrix in terms of error. A more rigorous proof would be to perform a *t*-test on the error. We demonstrate this as follows.

Let $\overline{\|\sigma\mathbf{S} - \sigma\mathbf{C}\|}$ and $\overline{\|\sigma\hat{\mathbf{C}}_B^2 - \sigma\mathbf{C}\|}$ be the average of 100 iterations of the error of $\mathbf{S}$ and $\hat{\mathbf{C}}_B^2$, respectively. Let s_{es} and s_{ecb} be the standard deviation of the error of $\mathbf{S}$ and $\hat{\mathbf{C}}_B^2$, respectively. For the given σ, we test the null hypothesis H_0: $\overline{\|\sigma\mathbf{S} - \sigma\mathbf{C}\|} = \overline{\|\sigma\hat{\mathbf{C}}_B - \sigma\mathbf{C}\|}$

versus the alternative hypothesis H_a: $\overline{\|\sigma\mathbf{S}-\sigma\mathbf{C}\|} > \overline{\|\sigma\hat{\mathbf{C}}_B^2-\sigma\mathbf{C}\|}$ with significance level equal to 0.05. The test statistics is

$$t = \frac{\overline{\|\sigma\mathbf{S}-\sigma\mathbf{C}\|} - \overline{\|\sigma\hat{\mathbf{C}}_B^2-\sigma\mathbf{C}\|}}{\sqrt{\frac{s_{es}^2}{N_0}+\frac{s_{ecb}^2}{N_0}}}, \tag{8-2}$$

where N_0 is the number of sample points. The number of degrees of freedom is calculated by

$$\nu = \frac{\left(\frac{s_{es}^2}{N_0}+\frac{s_{ecb}^2}{N_0}\right)^2}{\frac{s_{es}^4}{N_0^2(N_0-1)}+\frac{s_{ecb}^4}{N_0^2(N_0-1)}} = (N_0-1)\frac{(s_{es}^2+s_{ecb}^2)^2}{s_{es}^4+s_{ecb}^4}. \tag{8-3}$$

Figure 8-2 shows the *t*-statistic value and the critical value at each σ. The solid line is the *t*-statistic value and the dotted line is the critical value. Since for every σ, the *t*-statistic value is greater than the critical value, the null hypothesis is rejected. Therefore, The estimation error of $\mathbf{S}$ is greater than that of $\hat{\mathbf{C}}_B^2$. Or equivalently, the estimation error of $\hat{\mathbf{C}}_B^2$ is smaller than that of $\mathbf{S}$.

Figure 8-3 shows the changes in E_C and E_D when σ increases. The solid line is E_C and the dotted line is E_D. We can see that E_C increases while E_D remains almost constant as σ increases.

From Eq. (7-9), by substituting c_{ij} with σc_{ij}, we have

$$E_C = \sum_{j=ii=1}^{p\ p} \frac{1}{2}(\sigma c_{ij} - b\delta_{ij})^2$$
$$= \sigma^2 \sum_{j=ii=1}^{p\ p} \frac{1}{2}\left(c_{ij} - \frac{b}{\sigma}\delta_{ij}\right)^2 \qquad (8\text{-}4)$$

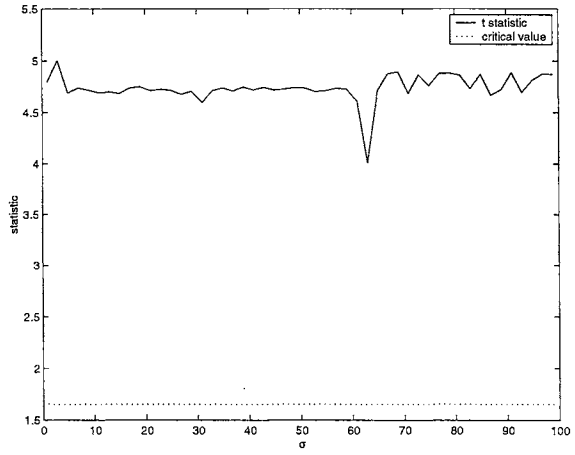


Figure 8-2 t-statistic vs. σ (covariance structure: multiple of identity)

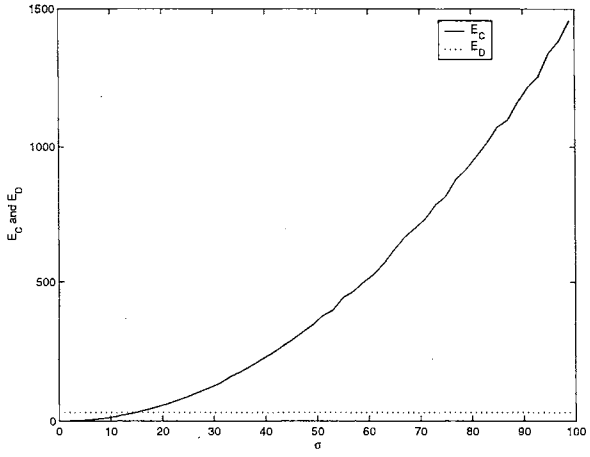


Figure 8-3 E_C and E_D vs. σ (covariance structure: multiple of identity)

As we will demonstrate shortly, $\frac{b}{\sigma}$ is a constant. We can therefore conclude that E_C is a quadratic function of σ, as we see in Figure 8-3.

From Eq. (7-5), since

$$E_D = \sum_{k=1}^{N} \frac{1}{2}(\sqrt{\sigma}\mathbf{x}_k - \sqrt{\sigma}\mu)^T(\sigma\mathbf{C})^{-1}(\sqrt{\sigma}\mathbf{x}_k - \sqrt{\sigma}\mu) = \sum_{k=1}^{N} \frac{1}{2}(\mathbf{x}_k - \mu)^T(\mathbf{C})^{-1}(\mathbf{x}_k - \mu), \quad (8\text{-}5)$$

where $\mathbf{x}_k$ can be viewed as the sample generated when $\sigma = 1.0$, and μ is the mean when $\sigma = 1.0$. We can see that E_D is not a function of σ, as we see in Figure 8-3.

Figure 8-4 shows the change in a and b as σ increases. The solid line represents a and the dotted line represents b. We can see that a decreases as σ increases, and b increases as σ increases. The relationship between b and σ is linear. In other words, $\frac{b}{\sigma}$ is a constant.

From Eq. (7-27), repeated here,

$$a = \frac{p(p+1)}{2tr[(\mathbf{H}^{MP})^{-1}] + 4E_C^{MP}}, \quad (8\text{-}6)$$

when σ increases, since E_C increases quadratically (see the discussion above), a decreases as σ^{-2}.

From Eq. (7-29), by substituting $\mathbf{C}^{MP}$ with $\sigma\mathbf{C}^{MP}$, we have

$$b = \frac{tr(\sigma\mathbf{C}^{MP})}{p} = \sigma\frac{tr(\mathbf{C}^{MP})}{p}, \quad (8\text{-}7)$$

Clearly, b is a linear function of σ.

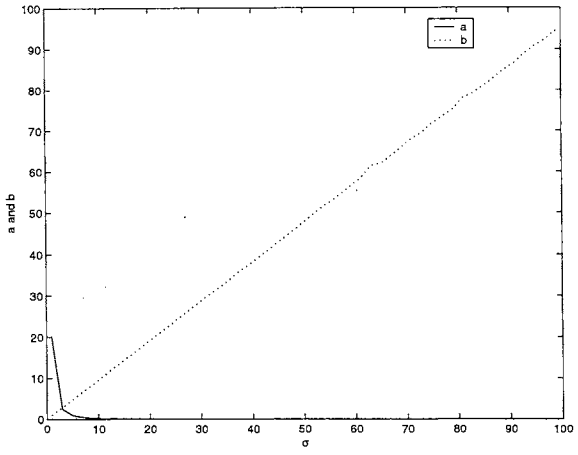


Figure 8-4 a and b vs. σ (covariance structure: multiple of identity)

Effect of sample size N

In this section, we study how the sample size N affects the covariance estimation errors, and how E_C, E_D, a and b change as N increases. The sample size takes on the values 10, 12, ..., 100, while σ and n_c remain at their nominal values, i.e., $\sigma = 10$ and $n_c = 20$.

Figure 8-5 shows the estimation errors and error standard deviations for $\mathbf{S}$ and $\hat{\mathbf{C}}_B^2$. The solid line is the error of $\mathbf{S}$. The two dotted lines are one standard deviation above and below the error of $\mathbf{S}$. The dashed line is the error of $\hat{\mathbf{C}}_B^2$. The two dash dotted lines are one standard deviation above and below the error of $\hat{\mathbf{C}}_B^2$.

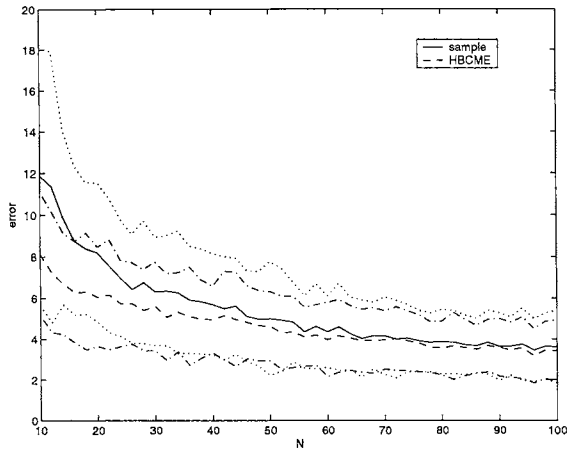


Figure 8-5 RMS estimation error vs. N (covariance structure: multiple of identity)

From Figure 8-5, we can see that the larger the sample size N, the smaller the error of the covariance matrices. This is true for both $\mathbf{S}$ and $\hat{\mathbf{C}}_B^2$. We already know that $\mathbf{S}$ approaches $\mathbf{C}$ asymptotically. Now we see that $\hat{\mathbf{C}}_B^2$ also has this property. Meanwhile, the average error of $\hat{\mathbf{C}}_B^2$ is always smaller than that of $\mathbf{S}$. Also, as the sample size increases, the improvement that $\hat{\mathbf{C}}_B^2$ provides decreases.

Figure 8-6 shows the t-statistic value and the critical value at each N. The solid line is the t-statistic value and the dotted line is the critical value. If $N < 40$, the t-statistic value is greater than the critical value, the null hypothesis is rejected. Therefore, the estimation error of $\hat{\mathbf{C}}_B^2$ is smaller than that of $\mathbf{S}$ when $N < 40$. When $N > 40$, from Figure 8-5, we can find that the error of $\hat{\mathbf{C}}_B^2$ is smaller than that of $\mathbf{S}$, but statistically, it is not significant.

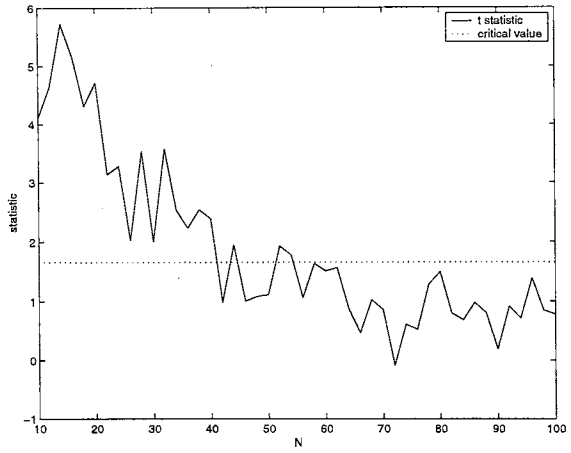


Figure 8-6 t-statistic vs. N (covariance structure: multiple of identity)

Figure 8-7 shows the change of E_C and E_D as N increases. The solid line is E_C and the dotted line is E_D. We can see that E_C remains almost constant, and E_D increases linearly as N increases.

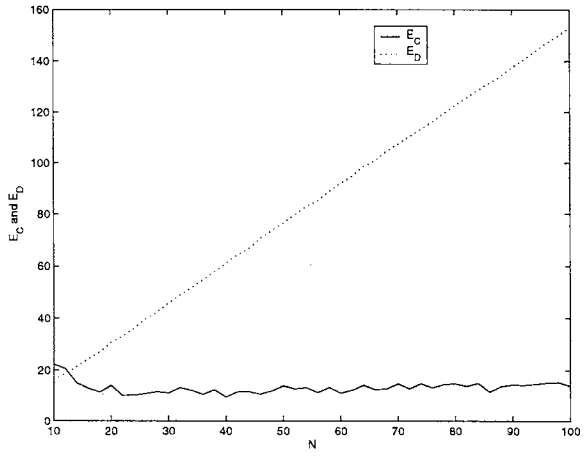


Figure 8-7 E_C and E_D vs. N (covariance structure: multiple of identity)

Eq. (7-9) is repeated here,

$$E_C = \sum_{j=i}^{p} \sum_{i=1}^{p} \frac{1}{2}(c_{ij} - b\delta_{ij})^2 . \tag{8-8}$$

Since b is not a function of N, we can see that E_C is not a function of N, as shown in Figure 8-7.

Eq. (7-5) is repeated below,

$$E_D = \sum_{k=1}^{N} \frac{1}{2}(\mathbf{x}_k - \mu)^T (\mathbf{C})^{-1} (\mathbf{x}_k - \mu) . \tag{8-9}$$

We can see that E_D is a linear function of N. E_D increases when N increases, as we see in Figure 8-7.

Figure 8-8 shows the change of a and b as N increases. The solid line represents a and the dotted line represents b. We can see that both a and b remain almost constant.

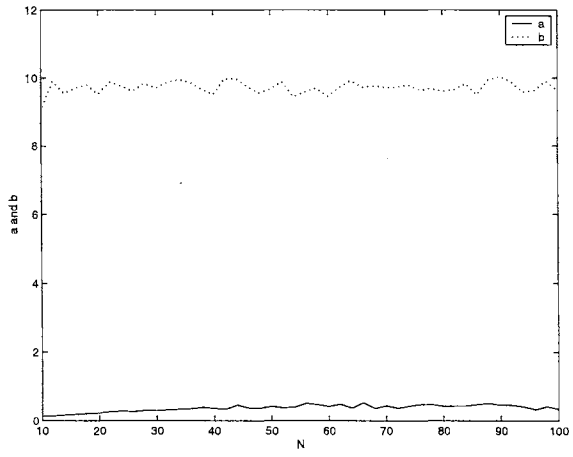


Figure 8-8 a and b vs. N (covariance structure: multiple of identity)

Eq. (7-27) is repeated here,

$$a = \frac{p(p+1)}{2tr[(\mathbf{H}^{MP})^{-1}] + 4E_C^{MP}}. \tag{8-10}$$

Since E_C is not a function of N (as discussed before), and N has almost no effect on $\mathbf{H}^{MP}$, therefore a is not a function of N, as we see in Figure 8-8.

Eq. (7-29) is repeated here

$$b = \frac{tr(\mathbf{C}^{MP})}{p}. \tag{8-11}$$

Clearly, b is not a function of N, as we see in Figure 8-8.

Effect of n_c

In this section, we study how the parameter n_c affects the covariance estimation errors, and how E_C, E_D, a and b change as n_c increases. n_c is a measure of how close the structure of the true covariance matrix is to the structure of the identity matrix. n_c takes on the values 10, 12, ..., 100, while σ and N remain at their nominal values, i.e., $\sigma = 10$ and $N = 20$.

Figure 8-9 shows the estimation errors and error standard deviations for $\mathbf{S}$ and $\hat{\mathbf{C}}_B^2$. The solid line is the error of $\mathbf{S}$. The two dotted lines are one standard deviation above and below the error of $\mathbf{S}$. The dashed line is the error of $\hat{\mathbf{C}}_B^2$. The two dash dotted lines are one standard deviation above and below the error of $\hat{\mathbf{C}}_B^2$.

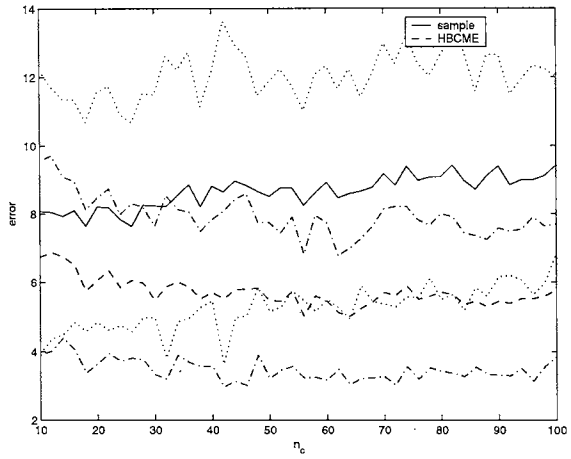


Figure 8-9 RMS estimation error vs. n_c (covariance structure: multiple of identity)

From Figure 8-9, we can see the estimation errors for both $\mathbf{S}$ and $\hat{\mathbf{C}}_B^2$ remain almost constant with the increase of n_c. This means that the structure of $\mathbf{C}$ has little effect on the errors of the estimate.

Figure 8-10 shows the *t*-statistic value and the critical value at each n_c. The solid line is the *t*-statistic value and the dotted line is the critical value. Since for every n_c, the *t*-statistic value is greater than the critical value, the null hypothesis is rejected. Therefore, the estimation error of $\hat{\mathbf{C}}_B^2$ is smaller than that of $\mathbf{S}$.

Figure 8-11 shows the changes of E_C and E_D when n_c increases. The solid line is E_C and the dotted line is E_D. We can see that E_C decreases, and E_D remains almost constant when n_c increases.

Eq. (7-9) is repeated here,

$$E_C = \sum_{j=ii=1}^{p \quad p} \frac{1}{2}(c_{ij} - b\delta_{ij})^2. \tag{8-12}$$

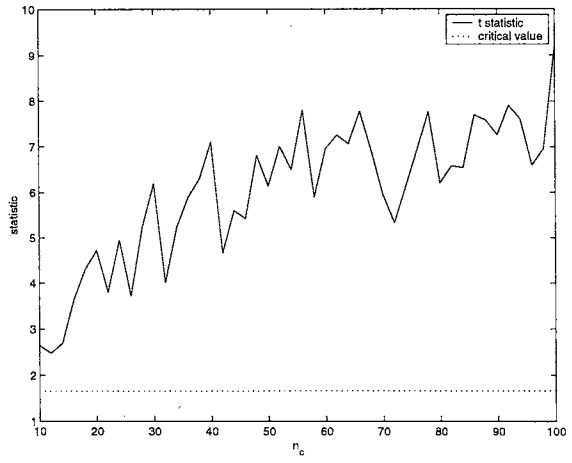


Figure 8-10 t-statistic vs. n_c (covariance structure: multiple of identity)

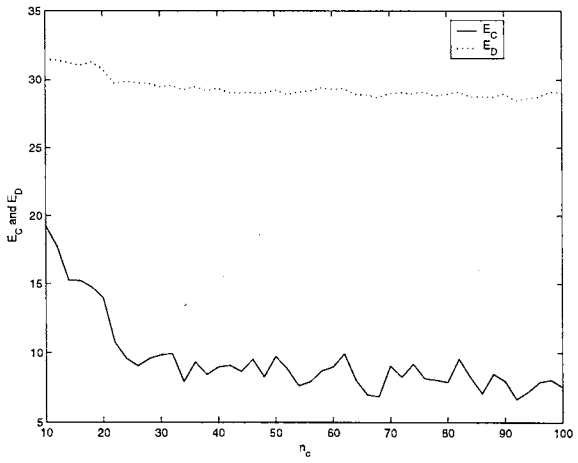


Figure 8-11 E_C and E_D vs. n_c (covariance structure: multiple of identity)

We can see that as n_c increases, the off-diagonal elements (c_{ij}, for $i \neq j$) become smaller, and E_C becomes smaller accordingly.

Eq. (7-5) is repeated here,

$$E_D = \sum_{k=1}^{N} \frac{1}{2}(\mathbf{x}_k - \mu)^T \mathbf{C}^{-1}(\mathbf{x}_k - \mu). \tag{8-13}$$

We can see that E_D is not a function of n_c, i.e., is not affected by the structure of the true covariance matrix. We can see this in Figure 8-11.

Figure 8-12 shows the change of a and b when n_c increases. The solid line represents a and the dotted line represents b. We can see that a increases, and b remains almost constant as n_c increases.

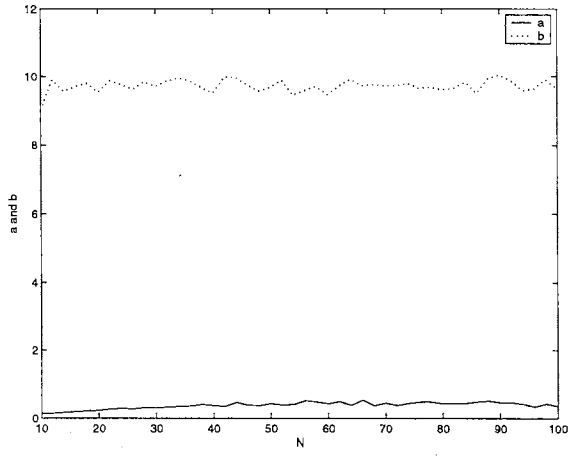


Figure 8-12 a and b vs. n_c (covariance structure: multiple of identity)

Eq. (7-27) is repeated here,

$$a = \frac{p(p+1)}{2tr[(\mathbf{H}^{MP})^{-1}] + 4E_C^{MP}}. \tag{8-14}$$

Since E_C decreases as n_c increases (as discussed before), and n_c has almost no effect on

$\mathbf{H}^{MP}$, therefore a increases as n_c increases, as we see in Figure 8-12.

Eq. (7-29) is repeated here,

$$b = \frac{tr(\mathbf{C}^{MP})}{p}. \tag{8-15}$$

Clearly, b is not a function of n_c, as we see in Figure 8-12.

In conclusion, when the true covariance matrix structure is a multiple of the identity matrix, we can see that the estimation error of $\hat{\mathbf{C}}_B^2$ is smaller than that of $\mathbf{S}$, no matter how σ, N and n_c change. In this case, $\hat{\mathbf{C}}_B^2$ is a better estimate of the true covariance matrix than $\mathbf{S}$. In the next section, we will see that when the true covariance structure is only a diagonal matrix (instead of a multiple of the identity matrix), $\hat{\mathbf{C}}_B^2$ is still a better estimate of the true covariance matrix than $\mathbf{S}$.

2. Monte Carlo Simulation: Extended HBCME

The simulation procedure applied in this section is similar to the one we used in section 1. The only difference is the configuration of the covariance matrix. Specifically, in step 2, we let $\mathbf{C} = \mathbf{S}^o + diag(\begin{bmatrix}5 & 2 & 1\end{bmatrix})$. Therefore, the structure of the true covariance matrix is not a multiple of the identity matrix anymore. We only assume the structure of the true covariance matrix to be diagonal, thus relaxing the assumption.

Figure 8-13 shows the estimation errors for $\mathbf{S}$, $\hat{\mathbf{C}}_B^2$ and $\hat{\mathbf{C}}_B^{p+1}$ when σ increases. The solid line is the errors of $\mathbf{S}$. The dotted line is the errors of $\hat{\mathbf{C}}_B^2$. The dash-dotted line

is the errors of $\hat{\mathbf{C}}_B^{p+1}$. Clearly, with the increase of σ, $\mathbf{S}$ has the largest estimation errors, $\hat{\mathbf{C}}_B^{p+1}$ has the smallest estimation errors, and the errors from $\hat{\mathbf{C}}_B^2$ stay in between.

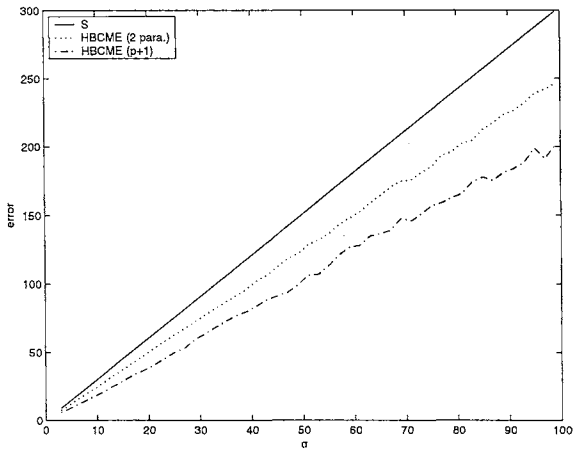


Figure 8-13 RMS estimation error vs. σ (covariance structure: diagonal)

Figure 8-14 shows the *t*-statistic value and the critical value at each σ. The solid line is the *t*-statistic value for $\hat{\mathbf{C}}_B^2$, the dash-dotted line is the *t*-statistic value for $\hat{\mathbf{C}}_B^{p+1}$, and the dotted line is the critical value. Since for every σ, the *t*-statistic value for both $\hat{\mathbf{C}}_B^2$ and $\hat{\mathbf{C}}_B^{p+1}$ are greater than the critical value, the null hypothesis is rejected. Therefore, The estimation error of $\mathbf{S}$ is greater than that of $\hat{\mathbf{C}}_B^2$ and $\hat{\mathbf{C}}_B^{p+1}$.

Figure 8-15 shows the estimation errors for $\mathbf{S}$, $\hat{\mathbf{C}}_B^2$ and $\hat{\mathbf{C}}_B^{p+1}$ when N increases. The solid line is the error of $\mathbf{S}$. The dotted line is the error of $\hat{\mathbf{C}}_B$s. The dash-dotted line is the error of $\hat{\mathbf{C}}_B^{p+1}$.

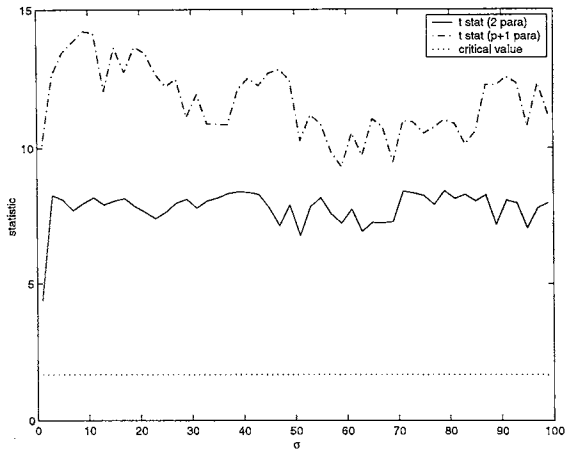


Figure 8-14 t-statistic vs. σ (covariance structure: diagonal)

From Figure 8-15, three conclusions can be made. First, $\hat{\mathbf{C}}_B^{p+1}$ has the smallest estimation errors. Secondly, for small N, $\mathbf{S}$ has the largest estimation errors. Thirdly, for large N, the error of $\mathbf{S}$ and the error of $\hat{\mathbf{C}}_B^2$ are about the same. Therefore, if the structure of the true covariance matrix is not a multiple of the identity matrix, $\hat{\mathbf{C}}_B^2$ does not provide much improvement over $\mathbf{S}$. However, if the sample size is very small (compared to the dimension of the matrix), $\hat{\mathbf{C}}_B^2$ is a better choice than $\mathbf{S}$.

Figure 8-16 shows the t-statistic value and the critical value at each N. The solid line is the t-statistic value for $\hat{\mathbf{C}}_B^2$, the dash-dotted line is the t-statistic value $\hat{\mathbf{C}}_B^{p+1}$, and the dotted line is the critical value. Since for every N, the t-statistic value for $\hat{\mathbf{C}}_B^{p+1}$ is greater than the critical value, the null hypothesis is rejected. Therefore, the estimation error of $\mathbf{S}$ is always statistically greater than that of $\hat{\mathbf{C}}_B^{p+1}$. For small N, the t-statistic value for $\hat{\mathbf{C}}_B^2$

is greater than the critical value, and the null hypothesis is rejected. Therefore, the estimation error of $\mathbf{S}$ is statistically larger than that of $\hat{\mathbf{C}}_B^2$ for small N. However, for large N, we can not reject the hypothesis "the estimation errors of $\mathbf{S}$ and $\hat{\mathbf{C}}_B^2$ are the same".

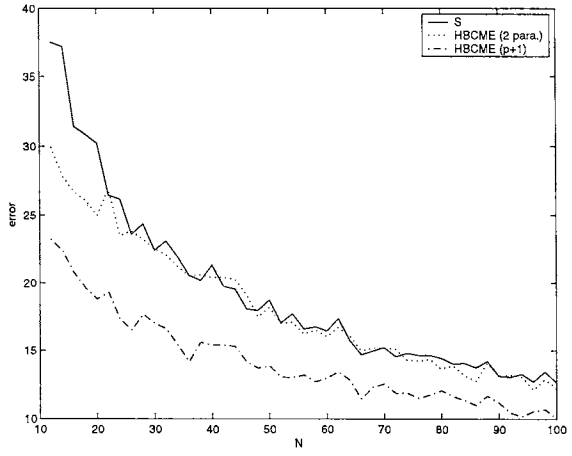


Figure 8-15 RMS estimation error vs. N (covariance structure: diagonal)

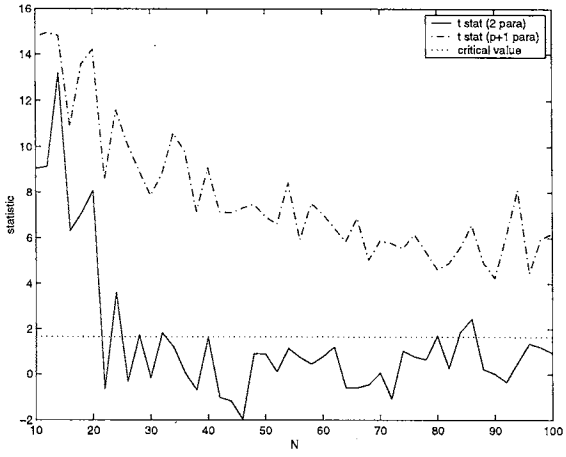


Figure 8-16 t statistic vs. N (covariance structure: diagonal)

Figure 8-17 shows the estimation errors for $\mathbf{S}$, $\hat{\mathbf{C}}_B^2$, and $\hat{\mathbf{C}}_B^{p+1}$ when n_c increases. The solid line is the error of $\mathbf{S}$. The dotted line is the error of $\hat{\mathbf{C}}_B^2$. The dash-dotted line is the error of $\hat{\mathbf{C}}_B^{p+1}$. Ranking the three covariance matrix estimates in terms of estimation error, from the largest to the smallest, gives $\mathbf{S}$, $\hat{\mathbf{C}}_B^2$ and $\hat{\mathbf{C}}_B^{p+1}$. Again, we can see that n_c has little effect on the errors of the estimates.

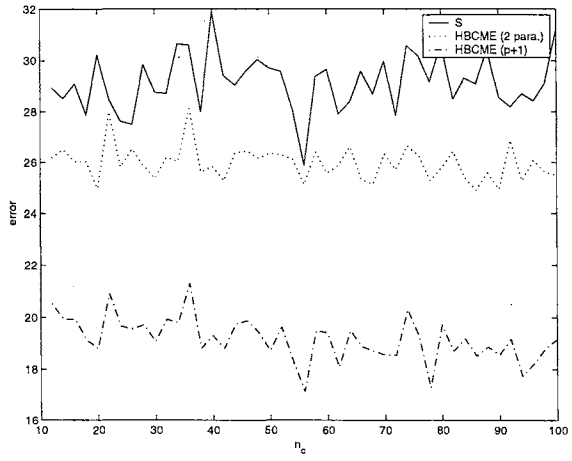


Figure 8-17 RMS estimation error vs. n_c (covariance structure: diagonal)

Figure 8-18 shows the *t*-statistic value and the critical value at each n_c. The solid line is the *t*-statistic value for $\hat{\mathbf{C}}_B^2$, the dash-dotted line is the *t*-statistic value for $\hat{\mathbf{C}}_B^{p+1}$, and the dotted line is the critical value. Since for almost every n_c, the *t*-statistic values for both $\hat{\mathbf{C}}_B^2$ and $\hat{\mathbf{C}}_B^{p+1}$ are greater than the critical value, the null hypothesis is rejected. Therefore, the estimation error of $\mathbf{S}$ is greater than that of $\hat{\mathbf{C}}_B^2$ and $\hat{\mathbf{C}}_B^{p+1}$.

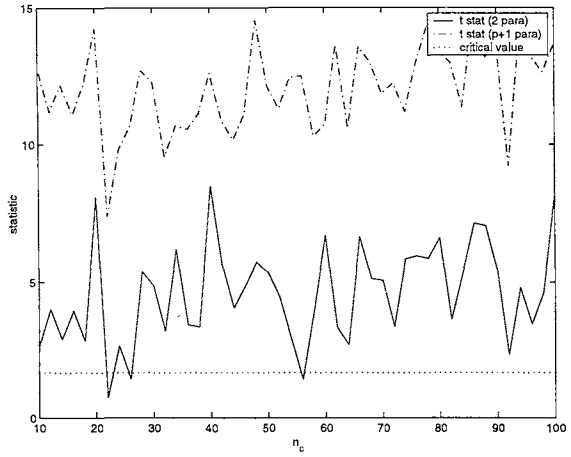


Figure 8-18 t statistic vs. n_c (covariance structure: diagonal)

In conclusion, when the true covariance matrix structure is a diagonal matrix (instead of a multiple of the identity matrix), $\hat{\mathbf{C}}_B^{p+1}$ has the smallest estimation error, and $\mathbf{S}$ has the largest estimation error. The estimation errors of both $\hat{\mathbf{C}}_B^{p+1}$ and $\hat{\mathbf{C}}_B^{2}$ are smaller than that of $\mathbf{S}$, no matter how σ, N and n_c change.

W e have finished the comparison of the sample covariance matrix and the covariance matrix estimated by the HBCME (with both 2 and $p+1$ unknown parameters in the prior covariance matrix) through Monte Carlo simulation. In Chapter 9, we will apply the HBCME to stock portfolio optimization.

CHAPTER 9

APPLICATION TO PORTFOLIO OPTIMIZATION

Stock portfolio optimization in itself is just a standard quadratic programming problem, and there are many different methods to solving it (see [19]). The difficulty is to estimate the mean and covariance matrix of the stocks in the portfolio, which are two parameters required in the optimization.

We propose two methods for performing the portfolio optimization. The first is Shrinkage Portfolio Optimization, and the second is Bayesian Portfolio Optimization. In the first approach, we first estimate the covariance matrix of the portfolio from the RCME, as described in Chapter 3. The estimated covariance matrix can be used in the James-Stein estimator for estimating the portfolio mean, and in the portfolio optimization itself. The name Shrinkage Portfolio Optimization comes from the fact that both the James-Stein estimator and the RCME belong to the class of shrinkage methods.

The second approach is similar to the first one. In this approach, instead of using the RCME to estimate the covariance matrix, we use the HBCME, as described in Chapter 7. The name Bayesian Portfolio Optimization comes from the fact that the HBCME applies Bayes' Theorem, and the James-Stein estimator can also be interpreted using Bayes' Theorem [24].

In this chapter, we first give some background information on stock portfolio optimization, then simulated stock return data are generated to perform both shrinkage portfo-

lio optimization and Bayesian portfolio optimization. Finally, both shrinkage portfolio optimization and Bayesian Portfolio optimization are used on real stock return data.

1. Background

In the following, we provide some background information on stock portfolio optimization. First, the efficient frontier is briefly reviewed. Then two mathematical forms for Mean Variance (MV) optimization are presented, and we explain why accurate estimates of both the mean and the covariance matrix are necessary in portfolio optimization. Finally the Sharpe Ratio is introduced for comparing the portfolio performance.

First we define what we mean by "efficient frontier". A portfolio is "efficient" if it has least risk for a given level of expected return, or equivalently, if it has the maximum expected return for a given level of risk. Figure 9-1 shows the concept of efficient frontier. The curve is the efficient frontier for a specific stock portfolio. Portfolio A is the investor's current portfolio, with certain given return and standard deviation. Portfolio B is efficient in the sense that it has the same expected return as A, but with the least possible risk (standard deviation). Portfolio C is also efficient in the sense that it has the maximum possible expected return at the same level of risk as A.

MV optimization is used to find the efficient frontier for a portfolio. MV optimization was first proposed by Markowitz [47]. It is the foundation of modern finance for efficient allocation of capitals among risky assets. Two versions of the MV optimization problem are considered in the following. One is the standard form with all constraints considered. The other relaxes the short selling constraints (allow short selling), resulting in a simplified solution with analytical form. There is another version of MV optimization,

which takes transaction cost into consideration [52] [56]. For simplicity, in the simulations we performed, the transaction cost is not considered.

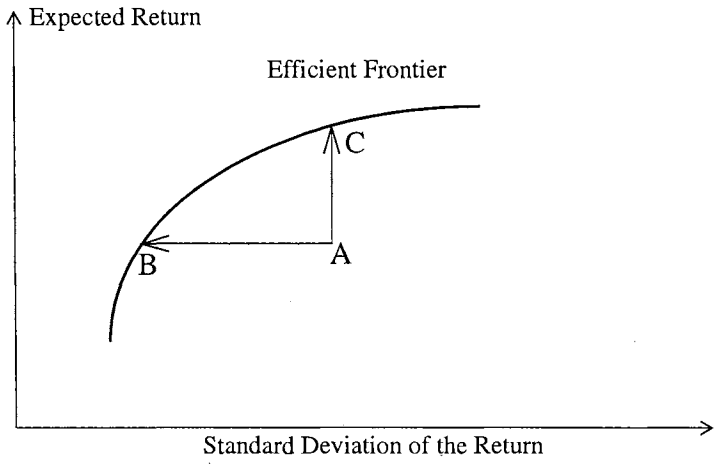


Figure 9-1 Efficient Frontier

The first form of MV optimization is the standard form which considers all the constraints. The standard MV optimization problem can be expressed as

$$\max_{\mathbf{w}_p} \quad u = \mu_P - r\sigma_P^2 \tag{9-1}$$

subject to

$$\begin{aligned} &\sum_{i=1}^{N} w_{p_i} = w_p \\ &w_{p_i}^L \le w_{p_i} \le w_{p_i}^U, i = 1, 2, \ldots, p \end{aligned}, \tag{9-2}$$

where

$$\begin{aligned} \mu_P &= \mu^T \mathbf{w}_p \\ \sigma_P^2 &= \mathbf{w}_p^T \mathbf{C} \mathbf{w}_p \end{aligned}, \tag{9-3}$$

and p is the number of stocks in the portfolio, u is the utility of the portfolio, μ_P is the expected return of the portfolio, σ_P^2 is the variance of the portfolio return, r is the risk aversion factor, μ is the expected return of each stock in the portfolio, $\mathbf{w}_p$ contains the portfolio weights, i.e., proportions of the total capital invested in each stock, w_{p_i} is the ith element of $\mathbf{w}_p$, $\mathbf{C}$ is the covariance matrix of the portfolio, $w_{p_i}^L$ and $w_{p_i}^U$ are the lower and upper bound of the weights of the ith stock, respectively.

The MV optimization problem stated in Eq. (9-1) - (9-3) is a standard quadratic programming problem. All standard methods for solving quadratic programming problems can be used (for a list of methods, see [19]). In the simulations below, we will use Sharpe's gradient method [56] [57], which is an intuitive optimization method.

Generally, the optimization assumes budget constraints ($w_p = 1$) and no-short-selling constraints ($w_{p_i}^L \geq 0$). However, in some situations, we can relax the no-short selling constraints. If shorting selling is allowed, the solution to the MV problem can be greatly simplified. This is the second form of our MV optimization formulation.

The second form of MV optimization is to consider the budget constraint, but short selling is allowed. This MV optimization problem can be expressed as finding the least risk for a given level of expected return. Mathematically, it can be formulated as

$$\max_{\mathbf{w}_p} \quad J = \frac{1}{2}\mathbf{w}_p^T \mathbf{C} \mathbf{w}_p \tag{9-4}$$

subject to

$$\begin{aligned} \sum_{i=1}^{N} w_{p_i} &= 1 \\ \mu^T \mathbf{w}_p &= q \end{aligned}, \tag{9-5}$$

where q is the desired portfolio return. w_p and q are known.

We proved in [33] that the solution for the MV optimization problem, Eq. (9-4) and Eq. (9-5), is

$$\mathbf{w}_p = \frac{b - qa}{bc - a^2}\mathbf{C}^{-1}\mathbf{1} + \frac{qc - a}{bc - a^2}\mathbf{C}^{-1}\mu \tag{9-6}$$

where $\mathbf{1}$ is a $p \times 1$ vector whose elements are all ones, and

$$\begin{aligned} a_p &= \mathbf{1}'\mathbf{C}^{-1}\mu \\ b_p &= \mu'\mathbf{C}^{-1}\mu \\ c_p &= \mathbf{1}'\mathbf{C}^{-1}\mathbf{1} \end{aligned} \tag{9-7}$$

At the first glance, the MV optimization problem seems easy to solve, but it assumes accurate estimates of μ and $\mathbf{C}$. The MV optimization can propagate and maximize the estimation errors of the mean μ and the covariance matrix $\mathbf{C}$ of the stocks in the portfolio. Michaud [49] pointed out that the MV optimization is actually an "estimation-error maximizer". The optimization procedure uses statistically estimated information and magnifies the impact of estimation errors. Michaud [49] pointed out that the MV optimizer significantly overweights those stocks that have large estimated returns, negative correlations and small variances, and on the other hand, underweights those stocks that have small estimated returns, positive correlations and large variances. These stocks are the ones most likely to have large estimation errors.

In the real world, it is very difficult to get accurate estimates of the mean μ and the covariance matrix $\mathbf{C}$ because of the small sample size. For example, U.S. domestic stock portfolios typically include 100-500 stocks, and international stock portfolios may include as many as 4000-5000 stocks [50]. There are $\frac{p^2+3p}{2}$ parameters to be estimated for a portfolio containing p stocks. The estimation requires a large amount of historical data, but few companies last even 50 years. In addition, the mean μ and the covariance matrix $\mathbf{C}$ of the stocks is time varying. We can only assume it is a constant over a short period of time. The sample size is always too small to accurately estimate the mean μ and covariance matrix $\mathbf{C}$ by using the sample mean and the sample covariance matrix.

Therefore, finding good estimators for estimating the mean and the covariance matrix is thus crucial for accurate MV optimization. This is where shrinkage portfolio optimization and Bayesian portfolio optimization come into play. In shrinkage portfolio optimization, the covariance matrix is estimated by the RCME. In Bayesian portfolio optimization, the covariance matrix is estimated by the HBCME. The estimated covariance matrix is subsequently used in the James-Stein estimator for estimating the stock mean, as in Eq. (4-3).

Before we perform the Monte Carlo simulation, we introduce a measure of the performance of a portfolio - the Sharpe Ratio.

For a stock portfolio, the excess return is the return of the portfolio minus the return of a risk-free asset, usually the 3-month US treasury bill rate. The Sharpe Ratio is defined as the expected excess return divided by the risk of the portfolio, i.e.

$$SR_p = \frac{\mu_P}{\sigma_P} = \frac{\mu^T \mathbf{w}_p}{\sqrt{\mathbf{w}_p^T \mathbf{C} \mathbf{w}_p}}. \tag{9-8}$$

where SR_p denotes the Sharpe Ratio of the portfolio. A higher Sharpe Ratio indicates a higher return μ_P for given amount of risk σ_P, or less risk for given amount of return. Therefore, the higher the sharpe ratio, the better the performance of the portfolio.

Next we present the Monte Carlo simulation results for portfolio optimization with both shrinkage portfolio optimization and Bayesian portfolio optimization. Two sets of stock return data are considered. One is randomly generated simulated stock data, the other is real stock return data. We use the Sharpe Ratio to judge the performance of the portfolio performance.

2. Simulated Stock Data

In this section, portfolio optimization is performed on simulated stock return data. The covariance matrix is estimated by the RCME, the LCME, and the HBCME, respectively. The mean is estimated by the James-Stein estimator.

Steps for performing the simulation are described as follows:

1) Set desired portfolio return $q = 0.01$, the portfolio size $p = 10$ and sample size $N = 60$.

2) Randomly generate mean μ and covariance $\mathbf{C}$. Then randomly generate N sample points with distribution $N(\mu, \mathbf{C})$.

3) Estimate the covariance matrix: The covariance matrix is estimated by the sample covariance matrix, the LCME, the RCME, and the HBCME. In the

RCME, the optimal shrinkage intensity k_o is estimated by both the Filtering method and the Constraint method (see Chapter 3). Set the upper limit $k_o^{up} = 0.1$ in the Constraint method.

4) Estimate the mean vector: The estimated covariance matrices from Step 3 are used as input to the James-Stein estimator to get four different portfolio mean, James-Stein sample mean, James-Stein Ledoit mean, James-Stein ridge mean, and James-Stein Bayesian mean.

5) Compute estimation error: compute the error of the estimates of mean and covariance matrix. The error of the mean vector is defined by $\|\hat{\mu} - \mu\|$, and the error of the covariance matrix is defined by $\|\hat{\mathbf{C}} - \mathbf{C}\|$, where the norm is the Frobenius norm, $\hat{\mu}$ and $\hat{\mathbf{C}}$ are estimated mean and covariance, respectively.

6) MV optimization: Apply Sharpe's gradient method [56] [57] to calculate the weight of each stock in the portfolio.

7) Generate 1000 data points with distribution $N_p(\mu, \mathbf{C})$. Apply the portfolio weights calculated in Step 6 to get the portfolio return at each point. Calculate the overall portfolio return for all three cases.

8) Repeat Step 2 - 7 1000 times. Calculate average and standard deviation of the error of the mean vector and the error of the covariance matrix. Calculate the average and the standard deviation of the portfolio return and then get the Sharpe Ratio.

9) Perform *t*-test for the error of the James-Stein estimator in which the covariance matrix is estimated by the LCME, the RCME and the HBCME, respec-

tively. Perform t-test for the error of the covariance matrix estimated by the LCME, the RCME, and the HBCME.

The t-test can be explained as follows (we take the example of the covariance matrix estimated by the HBCME): We test the null hypothesis H_0: $\|\mathbf{S}-\mathbf{C}\| = \|\hat{\mathbf{C}}_B-\mathbf{C}\|$ versus the alternative hypothesis H_a: $\|\mathbf{S}-\mathbf{C}\| > \|\hat{\mathbf{C}}_B-\mathbf{C}\|$ with significance level equal to 0.05. The test statistics is

$$t = \frac{\|\mathbf{S}-\mathbf{C}\| - \|\hat{\mathbf{C}}_B-\mathbf{C}\|}{\sqrt{\frac{s_{es}^2}{N_0}+\frac{s_{ecb}^2}{N_0}}}, \tag{9-9}$$

where s_{es} and s_{ecb} are the standard deviations of the errors of $\mathbf{S}$ and $\hat{\mathbf{C}}_B$, $N_0 = 1000$ is the number of sample points. The number of degrees of freedom is calculated by

$$\nu = \frac{\left(\frac{s_{es}^2}{N_0}+\frac{s_{ecb}^2}{N_0}\right)^2}{\frac{s_{es}^4}{N_0^2(N_0-1)}+\frac{s_{ecb}^4}{N_0^2(N_0-1)}} = (N_0-1)\frac{(s_{es}^2+s_{ecb}^2)^2}{s_{es}^4+s_{ecb}^4}. \tag{8-10}$$

Table 9-1 gives the simulation results. Optimal Sharpe ratios were calculated by using the true mean and covariance matrix. The last column shows if H_0 is rejected.

From the simulation results, the following observations can be made:

- The mean estimation error: The stock mean is estimated by the James-Stein estimator with the covariance matrix estimated by four different estimators, the sample covariance matrix, the LCME, the RCME, and the HBCME. The estimation error, ranked in the descending order, are: the sample covariance ma-

trix, the HBCME, the LCME, and the RCME. According to the *t*-test results, the errors of the James-Stein estimator in which the covariance matrix is estimated by the LCME, the RCME and the HBCME are all significantly smaller than the error of the James-Stein estimator in which the covariance matrix is estimated by the sample covariance matrix.

TABLE 9-1 Portfolio Performance Comparison with Simulated Data

		mean	standard deviation	*t*-statistics	reject H_0?
$\|\hat{\mu} - \mu\|$	Sample	0.3879	0.1146		
	LCME	0.2364	0.0977	31.8351	Yes
	RCME	0.2226	0.0933	35.3691	Yes
	HBCME	0.2439	0.1010	29.8267	Yes
$\|\hat{\mathbf{C}} - \mathbf{C}\|$	Sample	1.3771	0.2965		
	LCME	1.3526	0.2898	1.8719	Yes
	RCME	1.2724	0.2584	8.4243	Yes
	HBCME	1.3336	0.2717	3.1866	Yes
SR_p	Sample	0.0029			
	LCME	0.0036			
	RCME	0.0037			
	HBCME	0.0035			
	Optimal	0.0261			

- The covariance matrix estimation error: The covariance matrix estimated by the RCME has the smallest estimation error (defined as $\|\hat{\mathbf{C}} - \mathbf{C}\|$). The sample covariance matrix has the largest estimation error. The errors of the covariance matrices estimated by the LCME, the RCME, and the HBCME are all statistically smaller than the error of the sample covariance matrix.
- Sharpe Ratio of the portfolio: In comparing the Sharpe Ratio, the "optimal" estimator uses the true mean and covariance matrix to calculate Sharpe Ratio,

which is the true efficient frontier of the portfolio. Therefore, it has the highest achievable Sharpe Ratio. However, since we don't know the true mean and covariance matrix, the optimal estimator can not be used in practical situations. Comparing the sample covariance matrix estimators, the LCME, and the RCME, and the HBCME, the portfolio in which the covariance matrix is estimated by the RCME gives the highest Sharpe Ratio. The portfolio in which the covariance matrix is estimated by the sample covariance matrix estimator has the lowest Sharpe Ratio. The portfolio in which the covariance matrix is estimated by the LCME and the HBCME produce intermediate Sharpe Ratios.

3. Real Stock Data

In this section, portfolio optimization is performed using real stock data. Budget constraints and no-short selling constraints are assumed. Therefore, the standard form of the portfolio optimization as in Eq. (9-1) - (9-3) is used. Sharpe's gradient method [56] [57] is used for the optimization.

The candidate stocks for constructing the portfolio are from the S&P 500. Stocks were chosen from the S&P 500 composite as of June 25, 2001. A stock is included in the portfolio only if it has been publicly traded on the stock market no later than January, 1980. There are 68 stocks which meet this criteria. Stock monthly return data were obtained from Yahoo.com, which had been adjusted for dividends and splits. The three-month US treasury bill rate published by Federal Reserve Bank was taken as the riskless asset to calculate the excess return for each stock. The return data and treasury bill rate data were adjusted to reflect the annual return and the annual treasury bill rate, respectively.

Five portfolios were constructed based on the 68 candidate stocks. The first is the equal-weight portfolio, which is constructed by assigning each of the 68 stocks with equal weight. The second is the sample portfolio, in which the covariance matrix is estimated by the sample covariance matrix. The third is the Ledoit portfolio, in which the covariance matrix is estimated by the LCME. The fourth one is the ridge portfolio, which is constructed by the Shrinkage Portfolio Optimization method. The last one is the Bayesian portfolio, which is constructed by the Bayesian Portfolio Optimization method.

The simulation is performed as follows:

1) Set sample size $N = 100$, portfolio size $p = 68$. Set the desired return $q = 5\%$.

2) Take sample points $1 \sim N$, perform portfolio optimization. Construct the five portfolios by calculating the weight of each stock: the equal-weight portfolio, sample portfolio, Ledoit portfolio, ridge portfolio and the Bayesian portfolio.

3) Apply the weights of each portfolio to the $N + 1$ th sample to get each portfolio return and variance. Calculate the Sharpe Ratio of each portfolio at $N + 1$.

4) Take sample point $2 \sim N + 1$, repeat Step 2 - 3, calculate the Sharpe Ratio of each portfolio at $N + 2$. Continue with sample point $3 \sim N + 2$, $4 \sim N + 3$, ..., to calculate the Sharpe Ratio of each portfolio at $N + 3$, $N = 4, \ldots$, until we have Sharpe Ratios at each sample point except the first 100 points.

5) Set desired return $q = 10\%$, repeat Step 2 - 4.

6) Set sample size $N = 60$, repeat Step 2 - 5. Since in this case the sample size is smaller than the number of stocks in the portfolio, the sample portfolio can not be constructed because the sample covariance matrix can not be inverted.

In the Bayesian portfolio construction, it worth mentioning that the HBCME is very computational intensive. This is because the estimation of the covariance matrix requires iterative optimization. With each iteration, we need to calculate a large covariance matrix (68×68) and we need to compute the Hessian $\mathbf{H}^{MP}$ at each iteration. As the dimension of the covariance matrix increases, the computation time will increase geometrically.

Table 9-2 shows the results for $N = 100$ and $q = 5\%$, and 10%. The performance of the portfolios, in terms of Sharpe Ratio, from the best to the worst, are: ridge portfolio, Bayesian portfolio, equal-weight portfolio, Ledoit portfolio, and sample portfolio. Since the Ledoit portfolio and the sample portfolio perform worse than the equal-weight portfolio, there is no reason to use these two in practice. Both the ridge portfolio and the Bayesian portfolio perform better than the equal-weight portfolio in terms of Sharpe Ratio.

TABLE 9-2 Sharpe Ratio Comparison for Real Stock Data with $T = 100$

q	Equal Weight	Sample	Ledoit	Ridge	Bayesian
5%	0.4749	0.4072	0.4689	0.5014	0.4901
10%	0.4749	0.4118	0.4687	0.5086	0.4914

Table 9-3 shows the simulation results for $N = 60$. Since the portfolio size $p = 68$ is greater than N, the sample covariance matrix is not invertible. Therefore, the sample portfolio cannot be constructed here. We compare the performance of three portfolios: the equal-weight portfolio, the Ledoit portfolio, and the ridge portfolio. Results show

that both the Ledoit and the ridge portfolio perform better than the equal-weight portfolio in terms of Sharpe Ratio. The ridge portfolio performs the best amongst the three.

TABLE 9-3 Sharpe Ratio Comparison for Real Stock Data with $T = 60$

q	Equal Weight	Ledoit	Ridge
0.05	0.4869	0.5948	0.6077
0.1	0.4869	0.5963	0.6050

CHAPTER 10

CONCLUSIONS

In this work, we proposed two covariance matrix estimators and their applications to Least Squares (LS), Recursive Least Squares (RLS), and portfolio optimization. The two covariance matrix estimators are the Ridge Covariance Matrix Estimator (RCME) and the Hierarchical Bayesian Covariance Matrix Estimator (HBCME). The RCME is used to improve the LS and RLS algorithms, leading to methods called Shrinkage Least Squares (SLS) and Shrinkage Recursive Least Squares (SRLS). In the application to portfolio optimization, both the RCME and the HBCME are applied to the estimation of the covariance matrix, leading to the Shrinkage Portfolio Optimization algorithm and the Bayesian Portfolio Optimization algorithm.

Ridge Covariance Matrix Estimator

The RCME is developed to obtain a better estimate of the covariance matrix than the sample covariance matrix in terms of Mean Square Error (MSE) when the sample size is small. The RCME is a weighted average of the sample covariance matrix and the identity matrix. The RCME can also be viewed as a shrinkage method, in which we shrink the eigenvalues of the sample covariance matrix. There is only one parameter, the shrinkage intensity, that needs to be estimated in RCME. For sufficiently small shrinkage intensity, the RCME is guaranteed to have smaller MSE than the sample covariance matrix.

The RCME has some nice properties: First, if the sample size is large, the covariance matrix estimated by the RCME is close to the sample covariance matrix, which is a

consistent estimator of the true covariance matrix. Second, instead of shrinking toward the identity matrix, we could use other matrices if proper justification can be made. Third, the RCME can be viewed as a method of shrinking or expanding the eigenvalues of the sample covariance matrix toward 1 . Fourth, the condition number of the covariance matrix estimated by the RCME is smaller than that of the sample covariance matrix. Lastly, the RCME preserves the order of the eigenvalues.

Three methods for estimating the shrinkage intensity are proposed. The first is Ledoit's method, which is based on Ledoit's asymptotic estimation theorems. The second is a filtering method that takes the moving average value from Ledoit's method. The third is a constraint method that limits the upper bound of the shrinkage intensity.

Hierarchical Bayesian Covariance Matrix Estimator

The second estimator for covariance matrix estimation is the Hierarchical Bayesian Covariance Matrix Estimator (HBCME). The HBCME applies Bayes' theorem in two levels. The first level is used to estimate the true covariance matrix by maximizing the posterior density function of the covariance matrix. In this level, the structure of the covariance matrix is assumed to be a multiple of the identity matrix, and each element of the covariance matrix is assumed to have equal variance. Based on prior knowledge of the covariance matrix, a prior density function for the covariance matrix is assumed, with unknown parameters. Using Bayes' theorem, the analytical form of the posterior density function is found. An optimization method is used to find the Most Probable (MP) estimate of the covariance matrix by maximizing the posterior density function. The second level of Bayes' Theorem is used to estimate the unknown parameters in the prior density function from the first level.

Bayes's theorem is used to find the MP estimate of these two parameters. Simulation results show that the HBCME has superior performance than the sample covariance matrix.

If we assume that the prior covariance matrix is a multiple of the identity matrix, there are only 2 unknown parameters in the second level of Bayes' Theorem. If we assume the prior covariance matrix is a diagonal matrix, then there are $p + 1$ unknown parameters in the second level, where p is the dimension of the covariance matrix.

In practice, if the sample data are normalized, the HBCME with 2 unknown parameters is sufficient. Otherwise, the HBCME with $p + 1$ parameter is suggested.

Shrinkage Least Squares and Shrinkage Recursive Least Squares

Shrinkage Least Squares (SLS) and Shrinkage Recursive Least Squares (SRLS) improve on Least Squares (LS) and Recursive Least Squares (RLS). In the SLS algorithm, our improvement is based on James-Stein Least Squares (JSLS), which is an improvement to the standard LS. In JSLS, the covariance matrix is approximated by the sample covariance matrix. In SLS, the covariance matrix is estimated by the RCME. The recursive version of SLS, Shrinkage Recursive Least Squares (SRLS), is based on James-Stein Recursive Least Squares (JSRLS), which is an improvement to the standard Recursive Least Squares (RLS). Simulation results show that SRLS performs better than both JSRLS and the standard RLS.

Portfolio Optimization

The RCME and the HBCME were applied to the covariance matrix estimation in the stock portfolio optimization, leading to the Shrinkage Portfolio Optimization algorithm and the Bayesian Portfolio Optimization algorithm.

In stock portfolio optimization, due to the nature of the stock return data, the sample size is small. The optimization problem itself is not difficult to solve, but it assumes accurate estimates of the covariance matrix and the mean of the stocks in the portfolio. In Shrinkage Portfolio Optimization, the RCME first provides a better estimate of the covariance matrix (than the sample covariance matrix), then the estimated covariance matrix is used in the James-Stein estimator to get a better estimate of the mean (than the sample mean). The estimated covariance and mean are then used as parameters in the optimization algorithm. Improved estimates of the covariance matrix and the mean result in an improved optimization solution. Since both the covariance matrix and the mean of the stocks in the portfolio are estimated by shrinkage methods (the James-Stein estimator is also a shrinkage method), we call this approach Shrinkage Portfolio Optimization.

Bayesian Portfolio Optimization is similar to Shrinkage Portfolio Optimization. The only difference is that in Bayesian Portfolio Optimization, we use the HBCME to estimate the covariance matrix, instead of using the RCME. The name Bayesian comes from the fact that both the covariance matrix and the mean of the stocks in the portfolio are estimated using Bayesian methods (the James-Stein estimator can also be derived from Bayes' Theorem).

Simulation results show that the portfolios constructed by using Shrinkage Portfolio Optimization and Bayesian Portfolio Optimization perform better than the portfolio constructed using portfolio optimization with the sample covariance matrix or using an equal-weight strategy.

REFERENCES

[1] Anderson, T. W., *An Introduction to Multivariate Statistical Analysis*, 2nd Ed., New York: John Wiley & Sons, 1984.

[2] Bayes, T., "An essay towards solving a problem in the doctrine of chances," *Philos. Trans. Roy. Soc.*, 53, 370-418, 1763.

[3] Bernardo, J. M., "Reference posterior distributions for Bayes inference (with discussion)," *Journal of the Royal Statistical Society, Series B*, 41, 113-147, 1979.

[4] Brown, S. J., "The number of factors in security returns," *Journal of Finance*, 44, 5, 1247-1262, 1989.

[5] Chen, C-F, "Bayesian inference for a normal dispersion matrix and its application to stochastic multiple regress analysis," *Journal of the Royal Statistical Society*, Series B, 41, 2, 235-248, 1979.

[6] Christiansen, C. L., Morris, C. N., "Hierarchical Poisson regression modeling," *Journal of the American Statistical Association*, 92, 618-632, 1997.

[7] Daniels, M. J., "A prior for the variance in hierarchical models," *Canadian Journal of Statistics*, 1999.

[8] Daniels, M. J., and Kass, R. E., "Nonconjugate bayesian estimation of covariance matrices and its use in hierarchical models," *Journal of the American Statistical Association*, 94, 448, 1254-1263, 1999.

[9] Daniels, M. J., and Kass, R. E., "Shrinkage estimators for covariance matrices," to appear in *Biometrics*.

[10] Dempster, A. P., Laird, N. M., and Rubin, D. B., "Maximum likelihood from incomplete data via the EM algorithm (with discussion)," *Journal of the Royal Statistical Society, Series B*, 39, 1-38, 1977.

[11] Dennis, J. E., and Schnabel, R. B., *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Englewood Cliffs, NJ: Prentice-Hall, 1983.

[12] Dey, D. K., and Srinivasan, C., "Estimation of a covariance matrix under Stein's loss," *Annals of Statistics*, 13, 4, 1985.

[13] Dey, D. K., and Srinivasan, C., "Trimmed minimax estimator of a covariance matrix," *Annals of the Institute of Statistical Mathematics*, 38, Part A, 101-108, 1986.

[14] Dey, D. K., "Simultaneous estimation of eigenvalues," *Annals of the Institute of Statistical Mathematics*, 40, 1, 137-147, 1988.

[15] Efron, B., and Morris, C., "Stein's estimation rule and its competitors - an empirical Bayes approach," *Journal of the American Statistical Association*, 68, 117-130, 1973.

[16] Efron, B., "Biased versus unbiased estimation," *Advances in Mathematics*, 16, 3, 259-277, 1975.

[17] Efron, B., and Morris, C., "Multivariate empirical bayes and estimation of covariance matrices," *Annals of Statistics*, 4, 1, 22-32, 1976.

[18] Efron, B., and Morris, C., "Stein's Paradox in Statistics," *Scientific American*, 236, 5, 119-127, 1977.

[19] Fletcher, R., *Practical Methods of Optimization*, 2nd Ed., New York: John Wiley & Sons, 1987.

[20] Foresee, F. D., and Hagan, M. T., "Gauss-Newton approximation to Bayesian learning," *Proceedings of the 1997 International Joint Conference on Neural Network*, 1997.

[21] Gill, P. E., Murray, W., and Wright, M. H., *Practical Optimization*, New York: Academic Press, 1981.

[22] Greenberg, E, and Webster, C. E., *Advanced Econometrics: A Bridge to the Literature*, New York: Wiley, 1983.

[23] Grinold, R. C., and Kahn, R. N., *Active Portfolio Management: a Quantitative Approach for Providing Superior Returns and Controlling Risk*, New York: McGraw-Hill, 2000.

[24] Gruber, M. H. J., *Improving Efficiency by Shrinkage: the James-Stein and Ridge Regression Estimators*, New York: Marcel Dekker, 1998.

[25] Haff, L. R., "Minimax estimators for a multinormal precision matrix," *Journal of Multivariate Analysis*, 7, 374-385, 1977.

[26] Haff, L. R., "Estimation of the inverse covariance matrix: random mixtures of the inverse Wishart matrix and the identity," *Annals of Statistics*, 7, 6, 1264-1276, 1979.

[27] Haff, L. R., "An identity for the Wishart distribution with applications," *Journal of Multivariate Analysis*, 9, 531-544, 1979.

[28] Haff, L. R., "Empirical Bayes estimation of the multivariate normal covariance matrix," *Annals of Statistics*, 8, 3, 586-597, 1980.

[29] Haff, L. R., "The variational form of certain Bayes estimator," *The Annals of Statistics*, 19, 3, 1163-1190, 1991.

[30] Hagan, M. T., Demuth, H. B., and Beale, M., *Neural Network Design*, Boston: PWS Publishing Co., 1996.

[31] Haykin, S., *Adaptive Filter Theory*, 3rd Ed., Upper Saddle River, NJ: Prentice Hall, 1996.

[32] Hoerl, A. E., and Kennard, R. W., "Ridge regression: biased estimation for non orthogonal problems," *Technometrics*, 12, 55-67, 1970.

[33] Hu. Y., and Hagan, M., *Asset Return Forecasting and Equity Portfolio Management*, School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK, technical report, 2001.

[34] James, W., and Stein, C., "Estimation with quadratic loss," *Proceedings of the Fourth Berkeley Symposium on Mathematics and Statistics*, Berkeley: University of California Press 1, 361-379, 1961.

[35] Jeffreys, H., *Theory of Probability*, Oxford: Oxford University Press, 1961.

[36] Korn, R., *Optimal Portfolio*, Singapore: World Scientific Publishing, 1997.

[37] Ledoit, O., "A well-conditioned estimator for large dimensional covariance matrices," working paper, the Anderson School at UCLA, Los Angeles, California, 1996. http://www.anderson.ucla.edu/acad_unit/finance/wp/1995/24-95.pdf. (last accessed: Oct. 9, 2002).

[38] Ledoit, O., "Improved estimation of the covariance matrix of stock returns with an application to portfolio selection," working paper, the Anderson School at UCLA, Los Angeles, California, 1997. http://www.anderson.ucla.edu/acad_unit/finance/wp/1997/6-97.pdf (last accessed: Oct. 16, 2002).

[39] Leonard, T., and Hsu, J. S. J., "Bayesian inference for a covariance matrix," *Annals of Statistics*, 20, 4, 1669-1696, 1992.

[40] Lin, S. P., and Perlman, M. D., "A Monte Carlo comparison of four estimators of a covariance matrix," in *Multivariate Analysis 6*, Krishnaiah, P. R., ed., Amsterdam, North Holland: Elsevier Science Publishers, 411-429, 1985.

[41] Ljung, L, *System Identification: Theory for the User*, Upper Saddle River, NJ: Prentice Hall, 1999.

[42] Loh, W. L., "Estimating covariance matrices." Ph.D. dissertation, Department of Statistics, Stanford University, 1988.

[43] Loh, W. L., "Estimating covariance matrices," *Annals of Statistics*, 19, 1, 283-296, 1991.

[44] Loh, W. L., "Estimating covariance matrices II," *Journal of Multivariate Analysis*, 36, 163-174, 1991.

[45] MacKay, D. J. C., "Bayesian interpolation," *Neural Computation*, 4, 415-447, 1992.

[46] Manton, J. H., Krishnamurthy, V., and Poor, H. V., "James-Stein state filtering algorithms," *IEEE Transactions on Signal Processing*, 46, 9, 1998.

[47] Markowitz, H., *Portfolio Selection: Efficient Diversification of Investments*, New York: Wiley, 1959.

[48] Matlab online documentation, the Mathworks, Natrick, MA. 2002. http://www.mathworks.com/access/helpdesk/help/techdoc/ref/chol.shtml (last accessed: Dec. 06, 2002).

[49] Michaud, R. O., "The Markowitz optimization enigma: is optimized optimal?" *Financial Analysts Journal*, 1, 45, 31-42, 1989.

[50] Michaud, R. O., *Efficient Asset Management*, Boston, MA: Harvard Business School Press, 1998.

[51] Olkin, I., and Selliah, J. B., "Estimating covariances in a multivariate normal distribution," *Statistical Decision Theory and Related Topics II* (Gupta S. S. and Moore, D. S., Eds.), 313-326, New York: Academic, 2431-2447, 1977.

[52] Perold, A. F., "Large-scale portfolio optimization," *Management Science*, 30, 1143-1160, 1984.

[53] Perron, F., "Minimax estimator of a covariance matrix," *Journal of Multivariate Analysis*, 43, 16-28, 1992.

[54] Press, S. J., *Applied Statistics: Principles, Models, and Applications*, New York: Wiley, 1989.

[55] Rogers, G. S., *Matrix Derivatives*, New York: Marcel Dekker, 1980.

[56] Sharpe, W. F., "An algorithm for portfolio improvement," *Advances in Mathematical Programming and Financial Planning*, 1, 155-169, 1987.

[57] Sharpe, W. F., "The gradient Method," Graduate School of Business, Stanford University, Stanford, CA, 1999. http://www.stanford.edu/~wfsharpe/mia/opt/mia_opt1.htm (last accessed: Nov. 9, 2002).

[58] Sheena, Y., and Takekemura, A., "Inadmissibility of non-order-preserving orthogonally invariant estimators of the covariance matrix in the case of Stein's loss," *Journal of Multivariate Analysis*, 41, 117-131, 1992.

[59] Stein, C., *Some problems in multivariate analysis, Part I*, Technical Report 6, Dept. Statistics, Stanford University, 1956.

[60] Stein, C., Rietz lecture, 39th annual meeting IMS, Atlanta, Georgia, 1975.

[61] Stein, C., Unpublished notes on estimating the covariance matrix, 1977.

[62] Stein, C., "Lectures on the theory of estimation of many parameters," *Studies in the Statistical Theory of Estimation, Part I, Proceedings of Scientific Seminars of the Steklov Institute, Leningrad Division*, 74, 4-65, 1977.

[63] Yang, R., and Berger, J. O., "Estimation of a covariance matrix using the reference prior," *The Annals of Statistics*, 22, 3, 1195-1211, 1994.

APPENDIX

SOME THEOREMS ON MATRIX DERIVATIVES

The following theorems are from [55]:

Theorem 1 (page 52) Let $\mathbf{U}$ be $n \times n$ and nonsingular; let $\mathbf{A}$, $\mathbf{B}$ be constant,

$$\frac{\partial(\mathbf{A}\mathbf{U}^{-1}\mathbf{B})}{\partial \mathbf{U}} = -vec((\mathbf{A}\mathbf{U}^{-1})^T) \cdot vec^T(\mathbf{U}^{-1}\mathbf{B}).$$

Theorem 2 (page 51) Let $\mathbf{U}$ be $n \times n$ and nonsingular, then

$$\frac{\partial \log|\mathbf{U}|}{\partial \mathbf{U}} = (\mathbf{U}^{-1})^T.$$

Theorem 3 (page 80) Let $\mathbf{U}$ be an $n \times n$ matrix of distinct elements $u = vec(\mathbf{U})$ in an open ball Ω of R^{n^2}. Let f be a scalar function of $\mathbf{U}$ and differentiable in Ω. Let $\mathbf{V}$ be $n \times n$ but symmetric such that $v = vec(\mathbf{V}) \in \Omega$. Then

$$\frac{\partial f(\mathbf{V})}{\partial \mathbf{V}} = \left[\frac{\partial f(\mathbf{U})}{\partial \mathbf{U}} + \frac{\partial f(\mathbf{U})}{\partial \mathbf{U}^T} - diag\left(\frac{\partial f(\mathbf{U})}{\partial \mathbf{U}}\right)\right]\Bigg|_{\mathbf{U}=\mathbf{V}}.$$

VITA

Yong Hu

Candidate for the Degree of

Doctor of Philosophy

Thesis: COVARIANCE MATRIX ESTIMATION AND ITS APPLICATIONS

Major Field: Electrical Engineering

Biographical:

Personal Data: Born in Hefei, Anhui, China, on February 16, 1971, the son of Xueming Hu and Daimin Song.

Education: Received Bachelor of Science degree and Master of Science degree in Chemical Engineering from Zhejiang University, Hangzhou, Zhejiang, China, in July 1993 and March 1996, respectively. Received Master of Science degree in Chemical Engineering from Oklahoma State University, Stillwater, Oklahoma, in July 2000. Completed the requirements for the Doctor of Philosophy degree in Electrical and Computer Engineering at Oklahoma State University in August 2003.

Experience: Employed by Zhejiang Fuzzy Technology Company, Hangzhou, China, as a control systems design engineer from 1993 to 1997; Employed by Oklahoma State University, School of Chemical Engineering as a graduate teaching and research assistant from 1997 to 2000. Employed by Oklahoma State University, School of Electrical and Computer Engineering as a graduate research assistant from 2000 to present.

Professional Membership: IEEE.