DIMENSION REDUCTION AND COVARIANCE STRUCTURE FOR MULTIVARIATE DATA, BEYOND GAUSSIAN ASSUMPTION

A Dissertation

by

MEHDI MAADOOLIAT

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

August 2011

Major Subject: Statistics

DIMENSION REDUCTION AND COVARIANCE STRUCTURE FOR MULTIVARIATE DATA, BEYOND GAUSSIAN ASSUMPTION

A Dissertation

by

MEHDI MAADOOLIAT

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Co-Chairs of Committee,	Jianhua Huang
	Jianhua Hu
Committee Members,	Mohsen Pourahmadi
	Faming Liang
	Peng Li
Head of Department,	Simon J. Sheather

August 2011

Major Subject: Statistics

ABSTRACT

Dimension Reduction and Covariance Structure for Multivariate Data, Beyond Gaussian Assumption. (August 2011)

Mehdi Maadooliat, B.S., Sharif University of Technology;
M.S., Marquette University

Co-Chairs of Advisory Committee: Dr. Jianhua Huang
Dr. Jianhua Hu

Storage and analysis of high-dimensional datasets are always challenging. Dimension reduction techniques are commonly used to reduce the complexity of the data and obtain the informative aspects of datasets. Principal Component Analysis (PCA) is one of the commonly used dimension reduction techniques. However, PCA does not work well when there are outliers or the data distribution is skewed.

Gene expression index estimation is an important problem in bioinformatics. Some of the popular methods in this area are based on the PCA, and thus may not work well when there is non-Gaussian structure in the data. To address this issue, a likelihood based data transformation method with a computationally efficient algorithm is developed. Also, a new multivariate expression index is studied and the performance of the multivariate expression index is compared with the commonly used univariate expression index.

As an extension of the gene expression index estimation problem, a general procedure that integrates data transformation with the PCA is developed. In particular, this general method can handle missing data and data with functional structure.

It is well-known that the PCA can be obtained by the eigen decomposition of the sample *covariance* matrix. Another focus of this dissertation is to study the covariance (or *correlation*) structure under the non-Gaussian assumption. An important issue in modeling the covariance matrix is the positive definiteness constraint. The modified Cholesky decomposition of the inverse covariance matrix has been considered to address this issue in the literature. An alternative Cholesky decomposition of the covariance matrix is considered and used to construct an estimator of the covariance matrix under multivariate-t assumption. The advantage of this alternative Cholesky decomposition is the decoupling of the correlation and the variances.

DEDICATION

my mom, dad, sisters for their continuous support and my wife for her priceless love.

ACKNOWLEDGEMENTS

I would like to express my greatest gratitude to all the people who made this dissertation possible. I am deeply indebted to my supervisor, Professor Jianhua Huang, whose guidance, encouragement, patience, and support during my entire doctoral study enabled me to finish my Ph.D. successfully. Prof. Huang provided invaluable ideas and help on every single goal I have accomplished. He is the best and the kindest professor I have ever met.

My special thanks goes to my co-advisor, Dr. Jianhua Hu, for her trust in giving me the opportunity to get involve in modeling the gene expression project and to work under her supervision at the MD Anderson Cancer Center.

Special thanks to my committee members, Profs. Mohsen Pourahmadi, Faming Liang, and Peng Li, for taking their valuable time and serving as my committee members. I especially thank Prof. Pourahmadi for all his invaluable advice. I have learned a lot from him.

I would also like to thank Professor Gholamhossein G. Hamedani at Marquette University who provided much support in my being here. I am grateful for his guidance. Special thanks to my great friend Ali Nakhaee for his endless support in every aspect.

I thank my dear friends, Maysam, Negin, Mahmood, Ehsan, Ameneh, Saeed, all Maryams, Alis, Mohammads, Vahids, Arashs, Anh, Alex, Maurice, Youjin, Trijya, Tanya, Ying, Brian, Priya, Sahar, Shahrooz, Elham, Roozbeh, Saleh, Hassan, Mostafa, Salman, Fatemeh, Armin, Ahmad, Bijan, Hessam, Hoda, Amin, Emad, Sadegh, Alireza, and the rest of my unnamed friends in College Station. I honor their friendship and so many good memories throughout my journey at Texas A&M

University.

I would like to thank my beloved Zahra, my parents, and my sisters for their patience, care, and endless devotion. I can never thank my dad Reza, enough for teaching me to be thoughtful like him, my mom Susan, for her dedicated love, my sisters Soheila and Leila, for their strong support during the past years. I do not have the words to express my gratitude to my lovely wife Zahra, for the emotional and unconditional support she has brought into my life.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

FIGURE Page

CHAPTER I

INTRODUCTION

Having a valid model is the key to obtain satisfactory results from any given statistical procedure. There are a variety of tools to check the validity of model assumptions, but they are sometimes ignored by the end users. The packages given in statistical softwares could be called by the clients without checking the model conditions, and hence, the results in such cases could be misleading. Therefore, it is common to perform some preprocessing steps on the raw dataset to ensure that model assumptions are satisfied before the model fitting. One of these preprocessing methods is the transformation technique. The data transformation can be performed such that the transformed data appear to more closely meet the assumptions of a statistical procedure that is to be applied.

In the next two chapters we discuss the use of the data transformation technique as a preprocessing tool before PCA and functional PCA analysis, and provide applications for the proposed models. The motivation for the first part of the work is the Gene expression index estimation. Microarray technologies have been used to monitor expression intensities of thousands of genes simultaneously in a wide range of organisms. In the popular Affymetrix short oligonucleotide array platform, each gene is represented by multiple oligonucleotide probes, or a "probe set." A probe set contains 10-20 probe pairs whose expression intensities are measured via hybridization to the targeted sample cRNA. Each probe pair consists of the perfect match

The style of this dissertation follows Journal of Statistical Planning & Inference.

(PM) probe with the target mRNA sequence of 25 nucleotides and the counterpart mismatch (MM) probe that is identical to the PM probe except for a base change at the middle (13th) position.

A number of statistical methods have been proposed to address the gene expression index problem. GeneChip software (MAS version 5.0; Affymetrix 2004) computes a robust form of mean differences between PM and MM probes. Chu et al. (2002) applied mixed linear models to account for the dependence among logarithm-transformed probe intensities. Li and Wong (2001) proposed a multiplicative model-based expression index, which has been used and implemented in the dChip software (www.dchip.org). Its superiority over several existing methods has been shown in Lemon et al. (2002) via both analytic arguments and empirical data. Hu et al. (2006) observed the heterogeneity of the residuals obtained from the Li-Wong model, and they proposed to use the transformation model based on the ad-hoc entropy criteria. In Chapter II we explore this problem in details and propose a principled statistical procedure to obtain informative expression indexes and stabilize the variance of the residuals based on a data transformation routine.

In the second phase of this thesis we change the gear toward the use of the data transformation technique in the PCA for some special cases (i.e., functional data structure, presence of missing observations, and/or non-Gaussian behavior). The data transformation technique has been commonly used before the PCA and the functional PCA analysis (i.e., Huang et al., 2008; Hu et al., 2006). Usually, the authors proposed the transformation based on evidences of non-normality. The necessity of the Gaussian assumption comes from the fact that the aforementioned dimension reduction methods are essentially based on finding the directions with the highest variability (variance) in the dataset. Also, it is well known that the variance component, explains the properties of the dataset best, under the normality assumption.

The importance of the connection between the variance structure and the normality assumption clarifies the poor performance of the PCA and the functional PCA under the skewed and/or the heavy-tail distributed datasets. Moreover, the normality of the residuals have been considered in some of the missing data imputation techniques. Hence, it is quite possible to obtain unreliable imputed values for the missing observations in such techniques for heavy-tailed, asymmetric error distributions.

The essence of the relationship between the normality assumption and the variance structure motivates the search for a desired transformation to obtain normal behavior for the transformed dataset. The transformation usually comes from extensive data analysis, previous studies or the experience of expertise. Therefore, there is no automatic data driven procedure to select the appropriate transformation and it has been done manually in the literature. Chapter III focuses on the integration of the data transformation technique in the functional PCA based on an automatic data driven statistical procedure. First, we propose a probabilistic model to describe the problem. Based on the proposed model we explore some solutions for handling the missing data issue. Finally, in the later part of this chapter the focus goes toward the smoothed transformed functional principal components analysis.

Instead of using transformation to obtain the normality, one can use a rich family of the distributions (i.e., heavy tail distributions) in the modeling. In the last chapter, we consider the modeling of the covariance (correlation) matrix using the multivariate t-distribution. The most important concern is to take in to account the positive definiteness constraint. The modified Cholesky decomposition (MCD) of the inverse covariance matrix is one of the known solutions that has been widely used in the literature (Pourahmadi, 1999; Holan and Spinka, 2007). Most existing works have used precision matrix $\mathbf{\Sigma}^{-1}$, though Rothman et al. (2010) have proposed sparse estimation of $\mathbf{\Sigma}$ itself using its MCD.

Pourahmadi (2007) suggests that, if one replaces the modeling of MCD of a covariance matrix by its alternative Cholesky decomposition (ACD) introduced in Chen and Dunson (2003), then one obtains independent structure for the modeling of the correlation and the variances. Following this suggestion and pushing further the multivariate normality assumption of the data toward the multivariate t-distribution (similar to Lin and Wang, 2009), we are able to obtain parsimonious and robust estimation of the correlation matrix (not the covariance). The work in Chapter IV helps to understand the structural, computational and statistical differences that may exist between the MCD used in Pourahmadi (1999) and the ACD in Chen and Dunson (2003). Note that the former corresponds to the class of autoregressive (AR) models and the latter to the moving average (MA) models from time series analysis. Therefore, one expects more computational difficulties in computing the MLE parameters of the ACD.

CHAPTER II

ANALYZING MULTIPLE-PROBE MICROARRAY: ESTIMATION AND APPLICATION OF GENE EXPRESSION INDEXES

Gene expression index estimation is an essential step in analyzing multiple probe microarray data. Various modeling methods have been proposed in this area. Amidst all, a popular method proposed in Li and Wong (2001) is based on a multiplicative model, which is similar to the additive model discussed in Irizarry et al. (2003a) at the logarithm scale. Along this line, Hu et al. (2006) proposed data transformation to improve expression index estimation based on an ad hoc entropy criteria and naive grid search approach. In this work, we re-examined this problem using a new profile likelihood-based transformation estimation approach that is more statistically elegant and computationally efficient. We demonstrate the applicability of the proposed method using a benchmark Affymetrix U95A spiked-in experiment. Moreover, We introduced a new multivariate expression index and used the empirical study to shows its promise in terms of improving model fitting and power of detecting differential expression over the commonly used univariate expression index. As the other important content of the work, we discussed two generally encountered practical issues in application of gene expression index: normalization and summary statistic used for detecting differential expression. Our empirical study shows somewhat different findings from the MAQC project (MAQC, 2006).

2.1 Introduction

Microarray technologies have been used to monitor expression intensities of thousands of genes simultaneously in a wide range of organisms. We focus on the popular

Affymetrix short oligonucleotide array platform (Lockhart et al., 1996; Parmigiani et al., 2003).

The first part of this work concerns an important statistical problem in analyzing affymetrix array data, that is, estimation of gene expression based on the multiple-probe information. The so-called Li-Wong Reduced (LWR) model was proposed based on the differences between PM and MM intensities. Since the MM probes are designed originally for measuring the background/nonspecific intensities, the differences are considered to be the signal intensities in LWR. The model is expressed as

$$PM_{ij} - MM_{ij} = \theta_i \phi_j + \epsilon_{ij}, \tag{2.1}$$

where PM_{ij} and MM_{ij} are the PM and MM intensity values for the i^{th} $(i = 1, \cdots, I)$ array and the j^{th} $(j = 1, \cdots, J)$ probe pair for the gene, θ_i is the true expression index, ϕ_j is the rate of response in the corresponding PM probe, and the residuals $\epsilon_{ij} \sim N(0, \sigma^2)$. The model identifiability is ensured by posing the constraint $\sum_j \phi_j^2 = J$. Later work showed that the underlying distribution assumptions of normality and constant variance across the probes do not hold and data transformation techniques can be used to resolve this problem (Geller et al., 2003; Hu et al., 2006).

Hu et al. (2006) established the connection between the Li-Wong model and the first characteristic mode of the singular value decomposition (SVD) of the probe intensity matrix, and proposed a parametric transformation model on PM intensities. More specifically, they proposed a grid search over a parametric transformation family (e.g., Box-Cox) and selected the optimal value of the transformation parameters by maximizing the normalized discrete Shannon entropy defined on the singular values of the residual matrix. Note that, the empirical results provided in their paper show a good level of improvement in homogeneity of variance of the residuals and efficiency of the expression index. Moreover, the transformation model does not require knowledge of the experimental design.

We propose a more statistically principled estimation method than the entropy-based procedure in Hu et al. (2006). The transformation model can be written as

$$f(y_{ij}|\boldsymbol{\eta}) = \theta_i \phi_j + \epsilon_{ij}, \tag{2.2}$$

where $f(\cdot|\boldsymbol{\eta})$ is a monotonic transformation, $\boldsymbol{\eta}$ is the vector of transformation parameters, and ϵ_{ij}'s are independent normal random errors with mean 0 and constant variance σ^2. The goal of this transformation model is to stabilize the variance of the residuals and thus achieve an efficient estimates for the gene expression index under normality assumption. This is similar in spirit to performing logarithm transformation often seen in analyzing microarray experiments, as discussed in Geller et al. (2003); Hu et al. (2006) and Irizarry et al. (2003b). We consider a wider class of transformations which contains the logarithmic transformation and LWR model (identity transformation) as a special case. We expect to achieve a better performance based on the general model, since our empirical investigation shows that the logarithm transformation is not always optimal.

The proposed likelihood based method and the entropy based procedure in Hu et al. (2006) are closely related since the normal distribution has the maximum entropy property. However, our simulation study as reported in the supplementary material shows that the proposed method exhibits smaller variability of parameter estimation than the ad hoc entropy procedure. Application to a benchmark Affymetrix U95A spiked-in experiment (Irizarry et al., 2003a,b) also indicates that the proposed method has superior performance in terms of reflecting the true patterns in a controlled experiment, comparing to the Li-Wong model.

We also introduce a new multivariate gene expression index obtained through the connection between the multiplicative model and SVD. It is noted that all the existing methods concern only the univariate expression index. Through extensive real data exploration, we show the benefit of using the multivariate index in terms of model fitting and applications, such as differential expression detection.

The second part of the work is devoted to discussion of two practical issues in analysis of microarray data using estimated gene expression index, namely, normalization and summary statistic used for detecting differential expression. These issues are important but still under debate. MAQC (2006) used empirical studies over some of the known techniques, and concluded that normalization has little impact on the result of detecting differentially expressed genes, and p-value has no gain over fold-change in terms of gene ranking of differential expression. We use the benchmark spiked-in data to re-investigate these two issues for the expression indexes based on our proposed model. Our empirical study seems telling a different story from MAQC project.

We describe the likelihood based estimation procedure and introduce the two-dimensional expression index model in subsection 2.2. We use the well known benchmark human spiked-in dataset to demonstrate the applicability of the proposed expression index estimation method in subsection 2.4, and explore the two practical issues of normalization and summary statistic for detecting differential expression in subsection 2.5. Some final remarks are given in Chapter V.

2.2 Profile Likelihood Based Expression Index Estimation

2.2.1 Estimation Procedure

We consider the transformation model (2.2) in which y_{ij} takes the value of pre-processed and normalized multiple-probe index. We denote the parameter vector $\mathbf{\Theta} = (\boldsymbol{\eta}^T, \boldsymbol{\theta}^T, \boldsymbol{\phi}^T, \sigma^2)^T$, where $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_I)^T$, $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_J)^T$, and write out the log-likelihood function as

$$\ell(\mathbf{\Theta}) = -\frac{IJ}{2}\log(\sigma^2) \quad + \quad \sum_{i=1}^{I}\sum_{j=1}^{J}\Big\{\log|f'(y_{ij}|\boldsymbol{\eta})| - \frac{\big(f(y_{ij}|\boldsymbol{\eta}) - \theta_i\phi_j\big)^2}{2\sigma^2}\Big\}. \qquad (2.3)$$

It is noticeable that maximization of (2.3) with respect to $\mathbf{\Theta}$ simultaneously is computationally expensive. Hence, we turn to the profile likelihood method which comprises of two phases.

In the first phase, we consider the transformation parameter vector $\boldsymbol{\eta}$ to be fixed at $\boldsymbol{\eta_0}$. The parameters to be estimated are only $\boldsymbol{\theta}, \boldsymbol{\phi}$, and σ^2. The maximum likelihood estimates (MLEs) of the parameters can be viewed as functions of $\boldsymbol{\eta_0}$, and can be easily obtained using the nice result of connection between SVD and the least square estimates (or equivalently MLEs with normal residuals) established in Hu et al. (2006). The explicit forms of the parameter estimates are

$$\begin{aligned} \widehat{\boldsymbol{\theta}(\boldsymbol{\eta_0})} &= \boldsymbol{u_1}\frac{\sigma_1}{\sqrt{J}}, & \widehat{\boldsymbol{\phi}(\boldsymbol{\eta_0})} &= \boldsymbol{v_1}\sqrt{J}, \qquad (2.4)\\ \widehat{\sigma^2(\boldsymbol{\eta_0})} &= \frac{1}{IJ}\sum_{i=1}^{I}\sum_{j=1}^{J}\big(f(y_{ij}|\boldsymbol{\eta_0}) - \widehat{\theta(\boldsymbol{\eta_0})_i}\widehat{\phi(\boldsymbol{\eta_0})_j}\big)^2, \end{aligned}$$

where, $\boldsymbol{u_1}$ and $\boldsymbol{v_1}$ are the left and right singular vectors, respectively, corresponding to the largest singular value σ_1 by performing SVD on the matrix $\{f(y_{ij}|\boldsymbol{\eta_0})\}$. We do not need the full SVD decomposition, and efficient algorithm for low-rank matrix approximation can be used to speed up the calculation of the leading vectors (e.g. Achlioptas and Mcsherry, 2007).

In the second phase, we aim to obtain the estimate of $\boldsymbol{\eta}$ via maximizing the profile log-likelihood function

$$\begin{aligned} \ell_p(\boldsymbol{\eta}) = &- \frac{IJ}{2}\log(\sigma^2(\boldsymbol{\eta})) \\ &+ \sum_{i=1}^{I}\sum_{j=1}^{J}\Big\{\log|f'(y_{ij}|\boldsymbol{\eta})| - \frac{\big(f(y_{ij}|\boldsymbol{\eta}) - \theta(\boldsymbol{\eta})_i\phi(\boldsymbol{\eta})_j\big)^2}{2\sigma^2(\boldsymbol{\eta})}\Big\}. \qquad (2.5) \end{aligned}$$

Notice that the profile log-likelihood function only contains the vector of parameters $\boldsymbol{\eta}$ with all the other parameters being expressed as the functions of $\boldsymbol{\eta}$. The variety of appropriate optimization techniques such as Downhill Simplex or gradient based algorithms (e.g. Avriel, 1976) can be used subject to the structure of the family of transformations. It is worth emphasizing that the obtained estimates based on the profile likelihood are the MLEs of the parameters, and thus have good theoretical properties.

We abbreviate the whole profiling and singular value decomposition procedure as PSVD. The general iterative optimization procedure follows:

1. Start from an initial estimate of $\boldsymbol{\eta}$, denoted by $\boldsymbol{\eta}_t$ $(t = 0)$. Usually we pick the initial estimate associated to the model with no transformation.

2. Let $\boldsymbol{X}_t$ be the $I \times J$ matrix with $(i, j)^{\text{th}}$ entry $f(y_{ij}|\boldsymbol{\eta_t})$. Perform SVD $\boldsymbol{X}_t = \boldsymbol{U\Sigma V}^T$ and use (2.4) to obtain $\widehat{\boldsymbol{\theta}(\boldsymbol{\eta_t})}$, $\widehat{\boldsymbol{\phi}(\boldsymbol{\eta_t})}$, and $\widehat{\sigma^2(\boldsymbol{\eta_t})}$.

3. Obtain the updated value of $\boldsymbol{\eta_{t+1}}$ via an optimization algorithm to increase the value of the profile log-likelihood function $\ell_p(\boldsymbol{\eta_t})$ defined in (2.5).

4. Iterate between the last two steps until convergence is reached.

Up to this stage our development is only for a single gene. However, we encounter thousands of genes in real microarray experiments and aim at estimating a common transformation for all these genes. Our algorithm can be easily extended to handle this general case. Let H denote the total number of genes. For each gene h $(h = 1, \ldots, H)$, the log-likelihood $\ell_h(\boldsymbol{\Theta})$ is similar to (2.3) and is given by

$$\ell_h(\boldsymbol{\Theta}) = -\frac{IJ}{2}\log(\sigma_h^2) \quad + \quad \sum_{i=1}^{I}\sum_{j=1}^{J}\Big\{\log|f'(y_{ijh}|\boldsymbol{\eta})| - \frac{\big(f(y_{ijh}|\boldsymbol{\eta}) - \theta_{ih}\phi_{jh}\big)^2}{2\sigma_h^2}\Big\},$$

and the profile log-likelihood $\ell_{hp}(\boldsymbol{\eta})$ is similar to (2.5) as following

$$\ell_{hp}(\boldsymbol{\eta}) = -\frac{IJ}{2}\log(\sigma_h^2(\boldsymbol{\eta})) + \sum_{i=1}^{I}\sum_{j=1}^{J}\Big\{\log|f'(y_{ijh}|\boldsymbol{\eta})| - \frac{\big(f(y_{ijh}|\boldsymbol{\eta}) - \theta(\boldsymbol{\eta})_{ih}\phi(\boldsymbol{\eta})_{jh}\big)^2}{2\sigma_h^2(\boldsymbol{\eta})}\Big\}.$$

In this scenario, the objective function to be maximized is $\sum_{h=1}^{H}\ell_h(\boldsymbol{\Theta})$. In terms of algorithm implementation, we only need to make a modification to step 3 of the described procedure, in which $\boldsymbol{\eta_t}$ is updated to increase the value of $\sum_{h=1}^{H}\ell_{hp}(\boldsymbol{\eta})$. Despite of the huge dimensionality, the SVD technique enables the computational feasibility of handling all the genes simultaneously in our procedure.

Given this general framework, we focus on the popular Box-Cox transformation family as an example hereafter. Let $\boldsymbol{Y}$ denote the untransformed data. The Box-Cox transformation for each element of $\boldsymbol{Y}$ is defined as

$$f(y_{ij}|\beta) = \begin{cases} \dfrac{y_{ij}^{\beta} - 1}{\beta}\ , & \beta \neq 0, \\ \log(y_{ij})\ , & \beta = 0. \end{cases} \tag{2.6}$$

Since the transformation parameter $\boldsymbol{\eta} = \beta$ is one dimensional, the 3^{rd} step of the PSVD procedure becomes a simple optimization problem for a univariate concave function. In our implementation, we adopt the popular L-BFGS-B optimization algorithm (Byrd et al., 1994).

We conducted simulation studies to make comparisons between the PSVD method and the entropy-based procedure in Hu et al. (2006). We studied both cases of normally and non-normally distributed data. The observation is that the PSVD method yields more efficient parameter estimate than the entropy-based procedure. The simulation study also demonstrates the advantage of data transformation in improving the model fit. We include the detailed description of the simulation studies in the supplementary material.

2.2.2 Multivariate Expression Index

It is aforementioned that the MLE of $\boldsymbol{\theta}, \boldsymbol{\phi}$ in model (2.2) are the singular vectors corresponding to the largest singular value of data matrix $\{f(y_{ij}|\boldsymbol{\eta})\}$. It is also known that the data matrix can be represented as the sum of rank-one matrices based on SVD. The most important rank-one matrix is associated with the largest singular value and it captures the highest "energy" in the data, where "energy" is defined by either the 2-norm or Frobenius norm(e.g. Trefethen and Bau, 1997) of the matrix. An intuitive question is whether this rank-one matrix is sufficient to capture the majority of energy in the data and it is interesting to investigate if the rank-one matrix corresponding to the second largest singular value contains nontrivial information about gene expression index.

One of the typical way of assessing the relative importance of the low-rank matrices generated by the SVD is to look at the ratio of the corresponding singular values. Following this practice, we include the rank-one matrix associated with the second largest singular value in expression index estimation only if $\frac{d_2}{d_3} > c$, where d_i denotes the i^{th} largest singular value of the transformed data $f(y_{ij}|\hat{\beta})$ in (2.2). In our empirical investigation, we take the relatively conservative threshold $c = 3$. This idea also has been used in Hu et al. (2009). The model can be explicitly written as

$$f(y_{ij}|\hat{\beta}) = \begin{cases} \theta_i \phi_j + \epsilon_{ij} \ , & \frac{d_2}{d_3} \leqslant c, \\ \theta_i^{(1)} \phi_j^{(1)} + \theta_i^{(2)} \phi_j^{(2)} + \varpi_{ij} \ , & \frac{d_2}{d_3} > c. \end{cases} \tag{2.7}$$

When a gene can be represented using one component, we take θ_i as its univariate expression index for array i; Otherwise, we take the vector $(\theta_i^{(1)}, \theta_i^{(2)})$ as its two-dimensional expression index.

Note that more components can be considered in the similar way. Our empirical investigation suggests that the third and higher-order singular vectors mainly contain noises for short oligonucleotide array experiments and thus they will not be considered. More detailed description is deferred to subsection 2.4.2.

2.3 Simulation Studies

We conducted simulation studies to evaluate the performance of the PSVD method in terms of parameter estimation and make comparisons to the entropy-based procedure in Hu et al. (2006).

For the purpose of demonstration, we simulated the expression intensity data using model (2.2) for $H = 100$ genes with $I = 59$ (arrays) and $J = 16$ (probes). For each gene, we generated θ_i's from gamma distribution with shape parameter 3 and rate 1, multiplying by 150. We generated ϕ_j's from normal distribution with mean 1 and standard deviation 0.1 , and then scaled them to satisfy the constraint $\sum_j \phi_j^2 = J$. We considered $\beta = 2$ in the Box-Cox transformation throughout the simulation studies. We investigated four cases of the error distribution.

2.3.1 Normal Errors

We generated the errors (ϵ_{ij}'s) from mean zero normal distribution with the standard deviation of 25. The simulated data was generated from

$$y_{ij} = f^{-1}(\theta_i \phi_j + \epsilon_{ij} | \beta = 2), \quad i = 1 \ldots I \quad j = 1 \ldots J. \tag{2.8}$$

The PSVD estimate of β using all the data of 100 genes is 2.029, while the entropy method estimate of β is 2.023. The plot of profile log-likelihood versus the value of β is contained in the top panel of Figure 1. A smooth concave function is clearly seen and the maximum location is indicated by the vertical line. It is not surprising for β estimates to be similar between the PSVD and entropy based methods in the normal case.

Table 1: Comparisons between the PSVD and entropy-based methods in four cases of residual distribution.

Error Distribution	PSVD	Entropy
Normal	2.026(0.081)	2.017(0.176)
t with $df = 3$	1.977(0.226)	1.961(0.609)
Double Exp(1.5)	2.034(0.127)	2.001(0.258)
Skew Normal(1)	2.017(0.084)	2.024(0.183)

The normal quantile-quantile plots of the estimated residuals before and after the transformation are shown in the left and right middle panels of Figure 1, respectively. It is obvious that the residual distribution is closer to normal with the transformation. We also displayed the plot of mean expression estimates $(\hat{\theta}_i\hat{\phi}_j)$'s obtained from the Li-Wong model (i.e., without transformation) versus y_{ij} in the left bottom panel and that of mean expression estimates obtained from the transformation model versus $\widehat{f(y_{ij})}$'s in the right bottom panel. It is notable that the transformation results in more or less homoscedastic residuals, indicating good model fit.

To assess the variability of parameter estimation, we implemented the PSVD and entropy based methods for each gene separately and obtained 100 estimates of β using each method. We reported the average value of $\hat{\beta}$ and the corresponding standard error in Table 1. We observe that the two methods yielded very similar mean values (close to the true value of 2) but the standard error using the PSVD method is only 45.5% of that using the entropy based method.

2.3.2 Non-normal Errors

We also investigated the robustness of the two methods through three non-normal cases. We considered model (2.2) with errors generated from the following zero-mean distribution: (a) The t distribution with 3 degrees of freedom, multiplied by 15; (b) The double exponential distribution with the scale parameter 1.5; (c) The skew

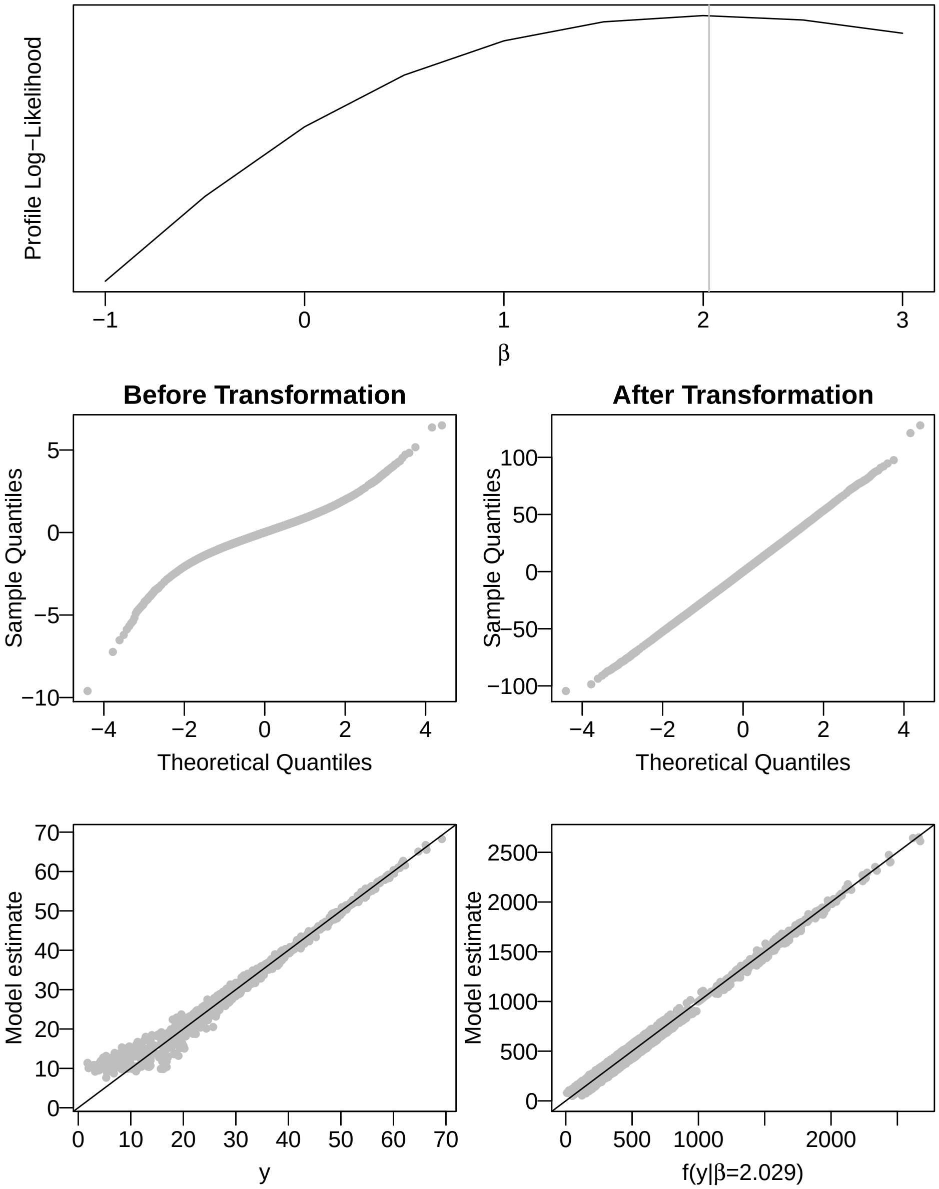

Figure 1: The top panel contains the plot of the profile log-likelihood for β. The second row contains the QQ-plot of residuals for the Li-Wong and PSVD models; The left bottom panel contains the plot of $(\hat{\theta}_i\hat{\phi}_j)$'s obtained from LWR (without transformation) versus y_{ij}; and the right bottom panel contains the plot of mean expression estimates obtained from the transformation model (PSVD) versus $\widehat{f(y_{ij})}$.

normal distribution with location parameter being $\frac{1}{\sqrt{\pi}}$, scale parameter 1 and shape parameter 1. We again obtained the simulated data using (2.8). The last three rows of Table 1 contain the average values of $\hat{\beta}$ and the corresponding standard errors. The parameter estimation variability of the entropy method is always at least twice as large as that of the PSVD method, which is consistent with the finding in the normal distribution case.

2.3.3 Sensitivity to Model Misspecification

Lastly, we considered the case $\beta = 1$, and studied the sensitivity of the proposed model to misspecification of the error distribution. We have generated the pseudo-observations y_{ij}'s, where $y_{ij} = \theta_i \phi_j + \epsilon_{ij}$, and ϵ_{ij}'s are from either non-normal or unequal variances distributions. We observed that fitting model (2.2) is essentially finding a transformation model ($f(y_{ij}|\hat{\beta}) = \hat{\theta}_i \hat{\phi}_j + \hat{\epsilon}_{ij}$), that automatically incorporates the followings: 1- After the transformation, the distribution of the residuals ($\hat{\epsilon}_{ij}$) is as closest as possible to homogeneous mean zero normal distributions. 2- The variability of the transformed observations (i.e., $f(y_{ij}|\hat{\beta})$) has been explained mainly by a single index model ($\hat{\theta}_i \hat{\phi}_j$). We have tried different mean zero heavy tailed and skewed distributions for the error terms. Most of them suggested $\hat{\beta}$ to be close to 1, which is equivalent to no transformation is necessary.

For evaluating the performance of the model for unequal variance errors, we generated the errors (ϵ_{ij}'s) from normal distributions with mean zero and the standard deviation 25 times the magnitude of ϕ_j's. It's clear that here we loose the constant variability assumption across different probs. The PSVD algorithm suggested almost a log-transformation ($\beta = 0.06$) to obtain the equality of variances across the probs. Figure 2 shows the advantage of using such transformation model for a randomly chosen gene. The increasing pattern of variability disappears after the transformation.

This can be seen in the residual plots and the plot of mean expression estimates obtained from the PSVD model versus $\widehat{f(y_{ij})}$.

2.4 Application to a Benchmark Spiked-in Data

We used a benchmark Affymetrix U95A spiked-in experiment as an example. It consists of $12,610$ genes non-differentially expressed across all the arrays, and 16 genes spiked-in at 14 known concentration levels ranging from 0 to $1,024$ picomolars, each of which has at least 3 replicate arrays. A Latin square design was used for the arrangement of 16 genes at different concentration levels in 59 arrays. The detailed description of the experimental design can be found in Irizarry et al. (2003a,b).

In real data analysis, we need to first perform data preprocessing procedure. Irizarry et al. (2003a,b) introduced quantile normalization and an alternative background intensity estimation method rather than directly using MM intensities as the Li-Wong model. Because of its good empirical performance, this preprocessing procedure has been widely used. Therefore, we adopted this approach and took the background adjusted and normalized PM probe intensities PM_{ij}^* as the data value prior to transformation in our development.

2.4.1 Comparisons of Expression Index Estimates

We estimated the expression indexes of all the genes using the Li-Wong and PSVD methods, respectively. The PSVD method applied to all the genes simultaneously yields the estimate of the transformation parameter $\hat{\beta} = 0.177$. We notice there is a difference of the data scale among the two methods: the Li-Wong method is conducted at the original scale, and the PSVD procedure is conducted at the nonlinear scale of the specific Box-Cox transformation. Because of the different scales in the Li-Wong and PSVD models, we transform the estimated expression index using the

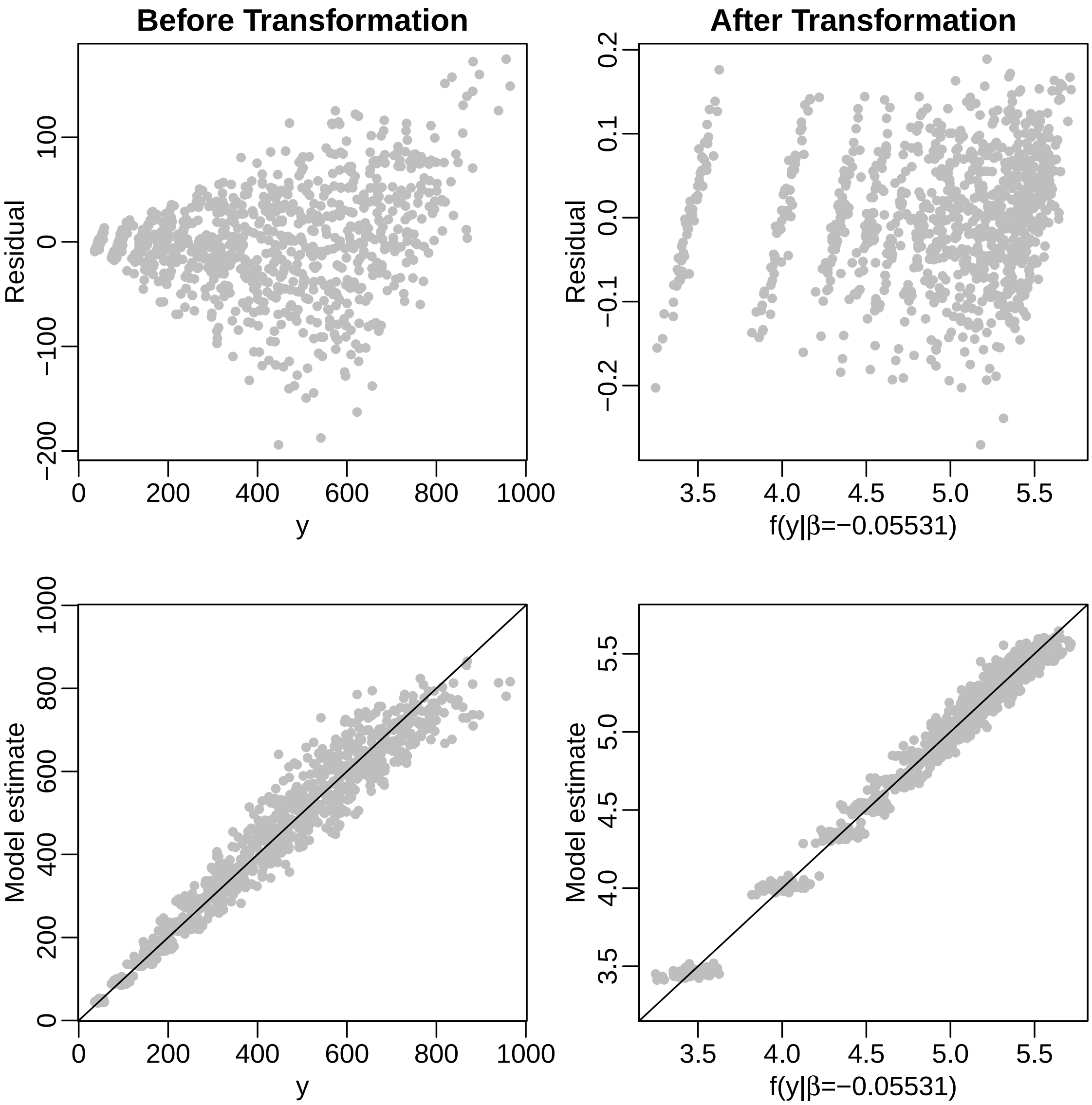

Figure 2: (Left column:) The results with no transformation, and the (Right columns:) results after the transformation. The upper row is the residual plots. The left bottom panel contains the plot of $(\hat{\theta}_i\hat{\phi}_j)$'s obtained from LWR (without transformation) versus y_{ij}; and the right bottom panel contains the plot of mean expression estimates obtained from the transformation model (PSVD) versus $\widehat{f(y_{ij})}$.

PSVD method back to the original scale for fair comparison. This can be performed using $\hat{\theta}_i^{\text{final}} = (\hat{\theta}_i + 1)^{\frac{1}{\beta}}$ for $i = 1, \ldots, I$, where $\hat{\theta}_i^{\text{final}}$ is a reasonable approximate for the expression index in the original scale.

It is known that all the genes but the spiked-in ones should be constantly expressed. So a good method should yield small variations in estimated expression indexes of the non-spiked-in genes across all the arrays. The left panel in Figure 3 shows the box plot of the standard errors of the estimated expression indexes of the nonspiked-in genes for both of the methods at logarithmic scale. The PSVD method is clearly superior to the Li-Wong method with overall smaller and less variable standard errors. The average (sample standard deviation) of the standard errors of all the genes are 40.49(77.53), and 11.05(25.41), respectively, for the Li-Wong and PSVD methods.

We also interrogated the 16 spiked-in genes spotted to the arrays at 14 concentration levels of 0, 0.25, 0.5, 1, 2, 4, 8, 16, 32, 64, 128, 256, 512 and 1024 picomolars. It is intuitive to assess the correlation between the estimated expression indexes and the corresponding true concentration levels for each gene. Because the concentration levels are designed in the two-fold fashion, we examined the Pearson's correlation coefficient posterior to base 2 logarithm transformation of both the concentration levels and $\hat{\boldsymbol{\theta}}_{final}$. The box plots of the correlation coefficients of the 16 genes are displayed in the right panel of Figure 3. We can see that the PSVD method outperforms the Li-Wong model. The average (sample standard deviation) of the correlation coefficients for the Li-Wong and PSVD methods are, respectively, 0.98(0.02) and 0.99(0.01). In summary, the PSVD method empirically performs better than the Li-Wong method.

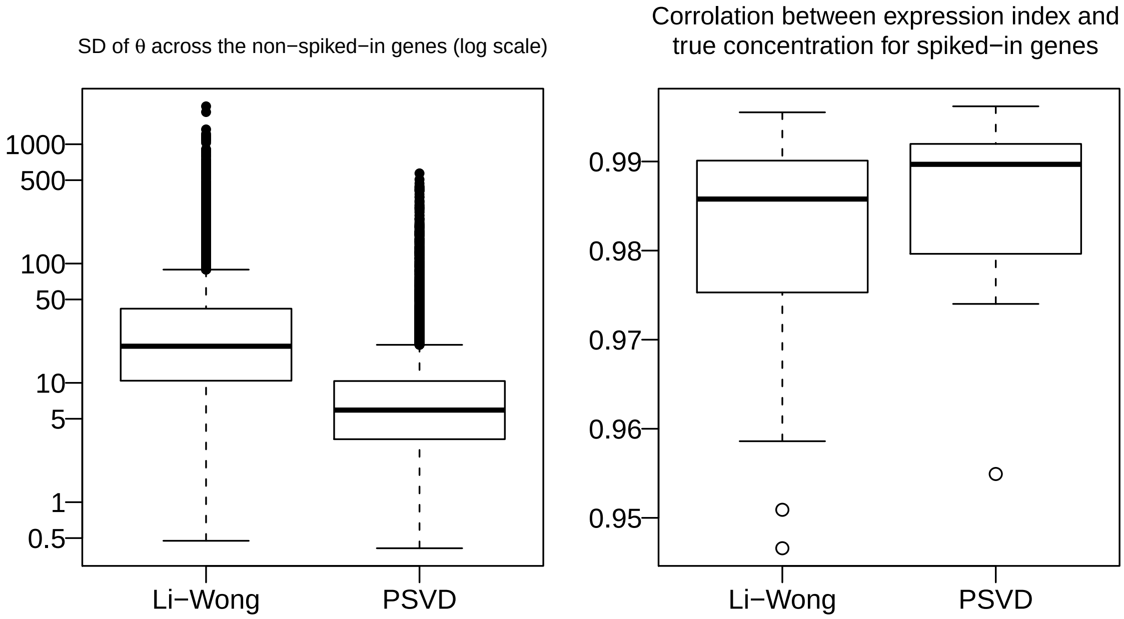

Figure 3: (Left panel:) Standard errors of estimated expression indexes of the nonspiked-in genes using the Li-Wong and PSVD methods. (Right panel:) Pearson's correlation coefficients between expression index($\hat{\boldsymbol{\theta}}_{final}$) and true concentration in base 2 logarithm of the spiked-in genes using the Li-Wong and PSVD methods.

2.4.2 Value of Multivariate Expression Index

We used the same dataset to explore applicability of the two-dimensional expression index. First, we observed that the second component is required by none of the 16 spiked-in genes without the transformation (or under the Li-Wong model) but required by 6 genes with the transformation (Box-Cox with $\hat{\beta} = 0.177$), based on the criterion defined in subsection 2.2.2. On the other hand, the second component is required by $1,044$ nonspiked-in genes without the transformation, but only by 38 with the transformation.

We focus on the 6 spiked-in genes that require the two dimensional expression index for further investigation. Figure 4 shows an example gene *546_at*, in which the left upper panel corresponds to the case of using only the first singular vector and the left lower panel corresponds to the case of using the first two singular vectors. In both panels, the probes having the largest and the second largest residual variance upon fitting model (2.2) are highlighted with the symbols of the cross and the black circle, respectively. The plot of the residuals ϵ_{ij} upon using only the first singular vector and the residuals ϖ_{ij} upon using the first two singular vectors against the transformed data are contained in the left column. It is clearly seen that the residuals of the most variable probes using just one singular vector become much more homogeneous by using two singular vectors. We also examined the normal quantile-quantile plots of the residuals and observed that the residual distribution is closer to the normal distribution using two singular vectors.

The right column of Figure 4 indicates that the fit of the most variable probe 15, indicated by the crosses, is unsatisfying using the rank-one matrix approximation. It is intriguing to interrogate the benefit of using the two singular vectors for this probe. Model (2.2) tells that a linear relation between the transformed data $f(PM^*_{ij}|\hat{\beta})$ and

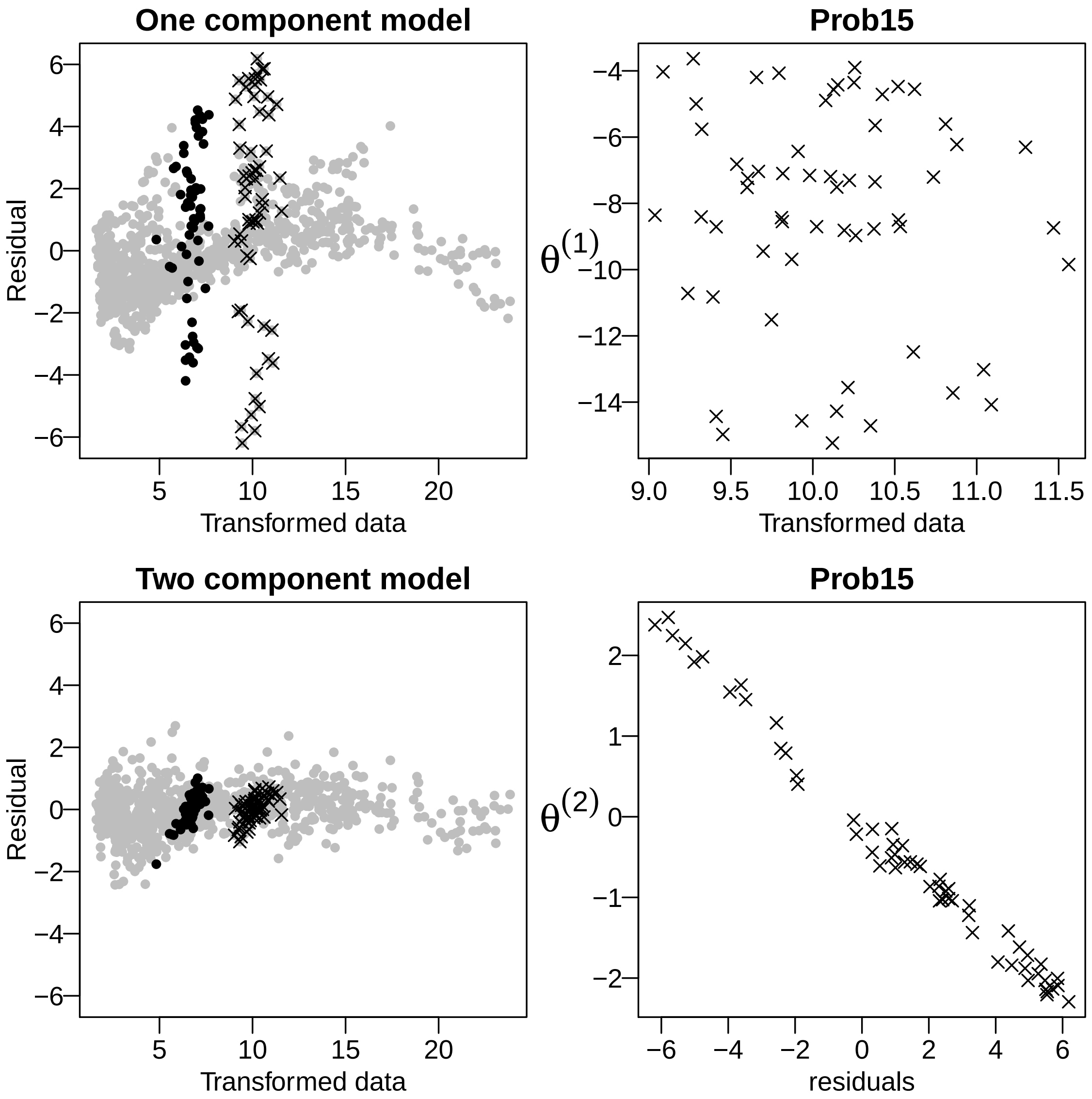

Figure 4: Model fitting using only one singular vector (upper panels) and two singular vectors (lower panels). The left column contains the plot of residuals versus the transformed data. The right upper panel contains the plots of the transformed data $f(PM^*_{ij}|\hat{\beta})$ versus the estimated expression index $\widehat{\theta_i^{(1)}}$ for probe 15 of gene *546_at*; The right lower panel contains the plots of residuals $f(PM^*_{ij}|\hat{\beta}) - \widehat{\theta_i^{(1)}}\widehat{\phi_j^{(1)}}$ versus $\widehat{\theta_i^{(2)}}$ for the same probe.

the univariate expression index $\hat{\theta}_i$ $(i = 1, \cdots, I)$ should be observed for any probe j if the probes behave consistently. Such a plot for probe 15 is displayed in the right upper panel of Figure 4 and a random pattern is shown. We examined the normally behaved probes and observed a clear linear pattern for all of them. In the right lower panel, we presented the plots of $\widehat{\theta_i^{(2)}}$ versus $f(PM^*_{ij}|\hat{\beta}) - \widehat{\theta_i^{(1)}}\widehat{\phi_j^{(1)}}$ obtained using the two-component model. We can observe a clear linear pattern for probe 15, implying that the second singular vector contains important information about this probe and makes substantial contribution to $\widehat{\theta_i^{(2)}}$.

The discussion above demonstrates the merit of using the second singular vector from the perspective of model fitting. Next, we investigate its empirical value using the known feature of the spike-in experiment. We focus on two spiked-in genes: *1091_at* and the earlier discussed *546_at*. With the known concentration level of each array for a spiked-in gene, we can examine the capability of distinguishing different concentration levels by using the two-dimensional expression index. For this purpose, we adopt K-means clustering procedure implemented using R function “pam” (Theodoridis and Koutroumbas, 2006). For gene *546_at*, we focused on the arrays at the concentration levels of $32, 64, 128, 256, 512,$ and 1024 picomolars. The other gene *1091_at* uses the arrays with the lowest concentration levels of $0, 0.25, 0.5,$ and 1 picomolars. This represents the most difficult case for discrimination because the concentration levels are the most close to each other. We apply the K-means procedure to cluster the arrays into three classes, using separately the univariate and two-dimensional expression indexes as the input. Performance of the two expression indexes will be evaluated based on the output of the K-means procedure.

An expression index is considered to be good if the corresponding clustering succeeds in assigning the same group membership to the arrays at the same concentration level and cluster together the arrays with the concentration levels at the similar magnitude. Figure 5 contains the plots of the concentration levels of the arrays versus their group membership produced by clustering. The left and right panels correspond to the univariate and two-dimensional expression index, respectively. Genes *546_at* and *1091_at* are shown respectively in the upper and lower regions, separated by the middle line in black, of the two panels. For gene *546_at* and based on univariate expression index, we observe that an array is assigned to cluster B while two others are assigned to cluster A at the concentration level of 64 picomolar, and similarly an array is assigned to cluster C while two others are assigned to cluster B at the concentration level of 256 picomolar. In contract, the two-dimensional expression index succeeds in clustering all the replicate arrays together and grouping the arrays at different concentration levels correctly according to the order of the concentration magnitudes. For gene *1091_at*, the two-dimensional expression index again shows its promise in differentiating groups of different concentration levels.

For the genes *546_at*, and *1091_at* we repeated the K-means procedure to cluster the arrays into three classes using the expression indexes obtained from the Li-Wong model. The result is given in the Figure 6. Here again we can see the outperformance of the PSVD over the Li-Wong model.

It is common to use two-sample t-test between two groups of arrays to check for differentially expressed genes. Also the negative of p-values for such tests are commonly used for ranking the genes. In the proposed model, given that there will be a mixture of univariate and multivariate indexes we can use the t-test and Hotelling's T-square statistics (Mardia et al., 1979) respectively. The p-values of such tests can be used for testing the difference in the expression indexes or the gene ranking. For gene *1091_at* we performed Hotelling's T-square test between two groups of arrays composed of low concentration levels $\ell 1 = \{0, 0.25\}$, and $\ell 2 = 0.5$. The first group includes two concentration levels because of their small array replicates. For the test of

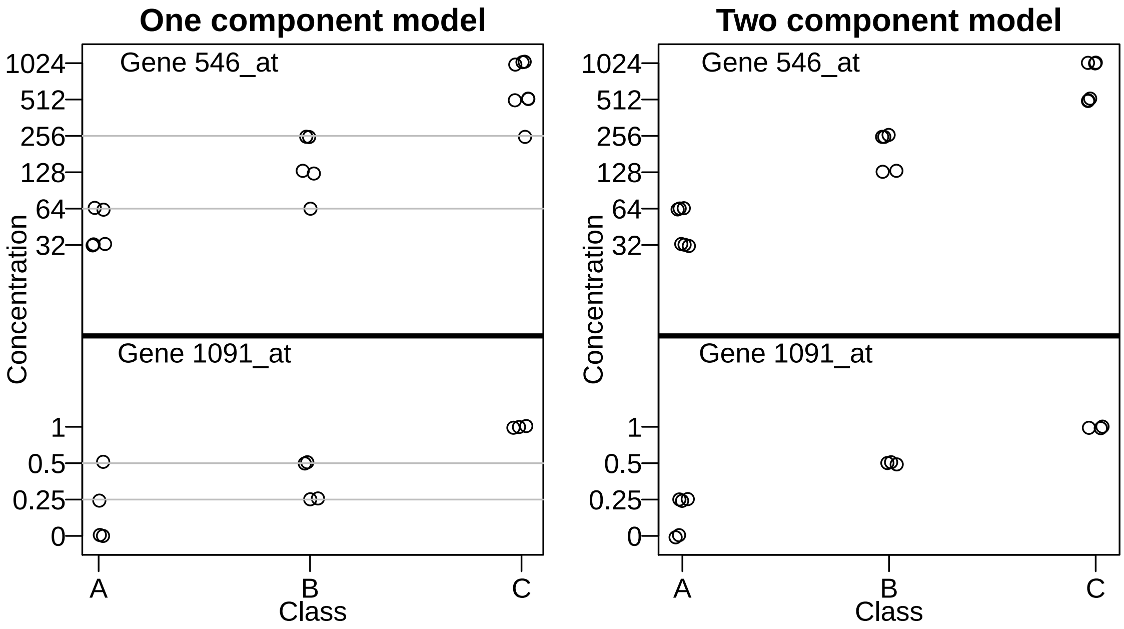

Figure 5: K-means clustering using univariate(left column) and two-dimensional(right column) expression index. The upper region corresponds to gene *546_at* at the concentration levels of $32, 64, 128, 256, 512$, and 1024 picomolars; The lower region corresponds to gene *1091_at* at the concentration levels of $0, 0.25, 0.5$, and 1 picomolars. Each panel contains the plot of the concentration levels of the arrays versus their group memberships produced by K-means clustering.

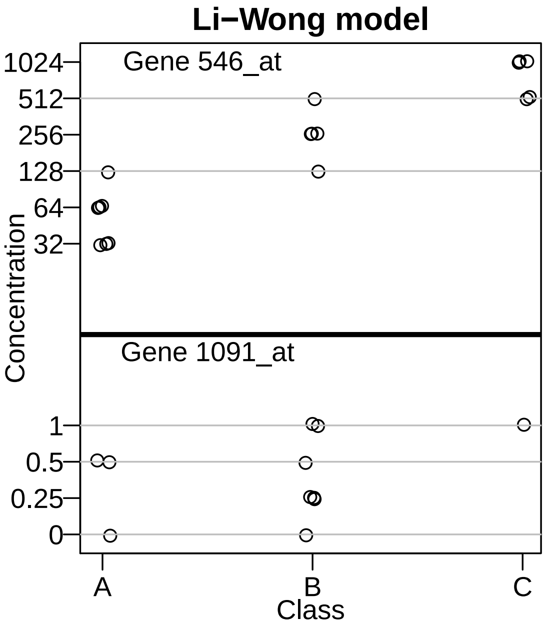

Figure 6: K-means clustering using Li-Wong model. The upper region corresponds to gene *546_at* at the concentration levels of $32, 64, 128, 256, 512$, and 1024 picomolars; The lower region corresponds to gene *1091_at* at the concentration levels of $0, 0.25, 0.5$, and 1 picomolars. Each panel contains the plot of the concentration levels of the arrays versus their group memberships produced by K-means clustering.

equal mean expression intensities between the two groups, using the univariate expression index obtains the p-value of 0.2 and the 95% confidence interval of $(-0.14, 0.51)$, while using the two-dimensional expression index obtains the p-value of 0.02 and the simultaneous 95% confidence interval of $(-0.29, 0.66)$ and $(-0.82, -0.07)$ for two dimensions of the expression indexes. Note that the two-dimensional expression index is the result of SVD and hence it is with respect to two orthogonal directions. Therefore, the two-dimensional expression index shows better power of detecting differential expression than the univariate expression index.

In summary, our empirical investigation shows that usage of the second singular vector has the empirical value in terms of improvement of both the model fit and capability of detecting differential expression.

2.5 Discussion of Two Practical Issues

There is much discussion of array normalization in the literature. Many researchers view normalization as a way to make intensities of arrays comparable. For example, Bolstad et al. (2003) studied various normalization methods and demonstrated the good performance of quantile normalization. Quantile normalization intends to make a set of quantiles identical across the arrays and was incorporated in the robust multi-array analysis (RMA) method described in Irizarry et al. (2003b). The detailed discussion of normalization techniques is referred to Bolstad et al. (2003). The data preprocessing steps including normalization are conducted to perform more meaningful downstream analysis, in which detection of differential expression is usually a primary task. This problem is typically tackled using some classical test statistics or equivalently the associated p-value. For example, ordinary two-sample t test statistic is usually used for two-group comparison and preferred over the naive summary statistic, fold-change (FC, ratio of mean group intensity) rule, which does not account for the variation among the samples.

On the other hand, MAQC (2006) drew rather different conclusions via some real data studies. In particular, they showed that the impact of array normalization on detection of differential expression is trivial for various existing techniques and the classical test statistics or the corresponding p-values do not show the advantage over the simple fold change rule in terms of gene ranking.

Herein we re-examine these two issues using the same Affymetrix U95A spiked-in data set. The known truth associated with the specific design of this data set allows assessing various criteria including the performance of gene ranking. Our studies are based on gene expression index estimates obtained from both our PSVD and the Li-Wong model.

In this data set, all except for 16 spiked-in genes have constant expression across all the 59 arrays. The Latin square design creates multiple sets of 12 arrays where all the arrays in a set have the common concentration levels for each of the 16 spiked-in genes. Therefore, such 12 arrays in a set can be treated as array replicates. Also among the sets, a spiked-in gene has different concentration levels. In our study, we focus on two such sets of 24 arrays and are interested in comparison of the arrays between these two groups.

We obtained the expression index estimates without normalization and with quantile normalization, respectively. Then we consider both the p-value obtained from the two-sample test and the simple absolute value of logarithm-transformed fold change (FC score) as the score for each gene. Negative p-value is used as the p-value score to be concordant with the order of the FC score. Intuitively, we expect the spiked-in genes to have smaller p-values and larger FC scores than all the other non-spiked-in genes. So it is sensible to assess the performance of gene summary score in terms of the power to distinguish between the two groups of non-spiked-in and spiked-in genes. For this purpose, we take a look at the receiver operating characteristic (ROC) curve and its commonly used summary statistic, the area under the curve (AUC) measurement. The results based on PSVD expression index estimates are shown in Figure 7. The ROC curves without normalization and using quantile normalization are plotted in the left upper and lower panels, respectively. In each panel, the p-value score and the FC score are plotted in solid and dotted lines, respectively, and the AUC measurements of the two scores are shown. We observe that quantile normalization clearly improves the discriminatory power of both scores over no normalization and the p-value score also has better performance than the FC score.

We also study the performance of gene ranking from another perspective. We plot the ranks of all the genes based on the p-value score along with those based on the FC score in the right panels of Figure 7. The 16 spiked-in genes are indicated as the bold dots. In truth, the 16 genes should have higher ranks than all the other genes and, thus, they should locate in the extreme right upper corner. Clearly, the normalization performs better than the no normalization case. The p-value score also outperforms the FC score, whereas the latter has more non-spiked-in genes ranking higher than the spiked-in ones.

Next, we focus on study of the 16 spiked-in genes since their true concentration levels are known on all the arrays. Figure 8 shows the plot of ranks of the 16 genes based on each score along with their ranks based on the absolute difference of concentration levels between the two groups. We expect a score and the absolute concentration difference to be positively correlated. The cases without normalization and with quantile normalization are displayed in the upper and lower rows, respectively; and the FC score and the p-value score are displayed in the left and right columns, respectively. The corresponding Spearman's correlation coefficient is displayed in each panel. We notice that no normalization obtains correlation coefficients closer to zero than quantile normalization. The advantage of the p-value score is obvious over the FC score when the quantile normalization is applied, with the correlation coefficients of 0.202 and -0.141, respectively. The FC score yields the negative correlation coefficient and shows group-splitting pattern where some genes are positively correlated while the others show the opposite pattern. In the contrast, the p-value score in the right lower panel shows the highest correlation in the correct direction. This panel shows clearly that most genes are positively correlated, except for the three outliers in the right low corner. It appears that all the three genes have the highest concentration levels in at least a group, and thus this phenomenon could be caused by the saturation problem.

We obtained the similar results based on the expression index estimates of the Li-Wong model. The results are included in the supplementary material.

In summary, our empirical investigations provide some evidence that: (1) normalization has significant impact on detection of differentially expressed genes; (2) p-value has favorable performance in terms of ranking over the naive fold change score.

The Figures 9 and 10 contain the results based on the expression index estimates of the Li-Wong model. They are similar to Figures 7 and 8 of the results based on the PSVD method, respectively. In summary, we obtain the similar results as described for the PSVD model.

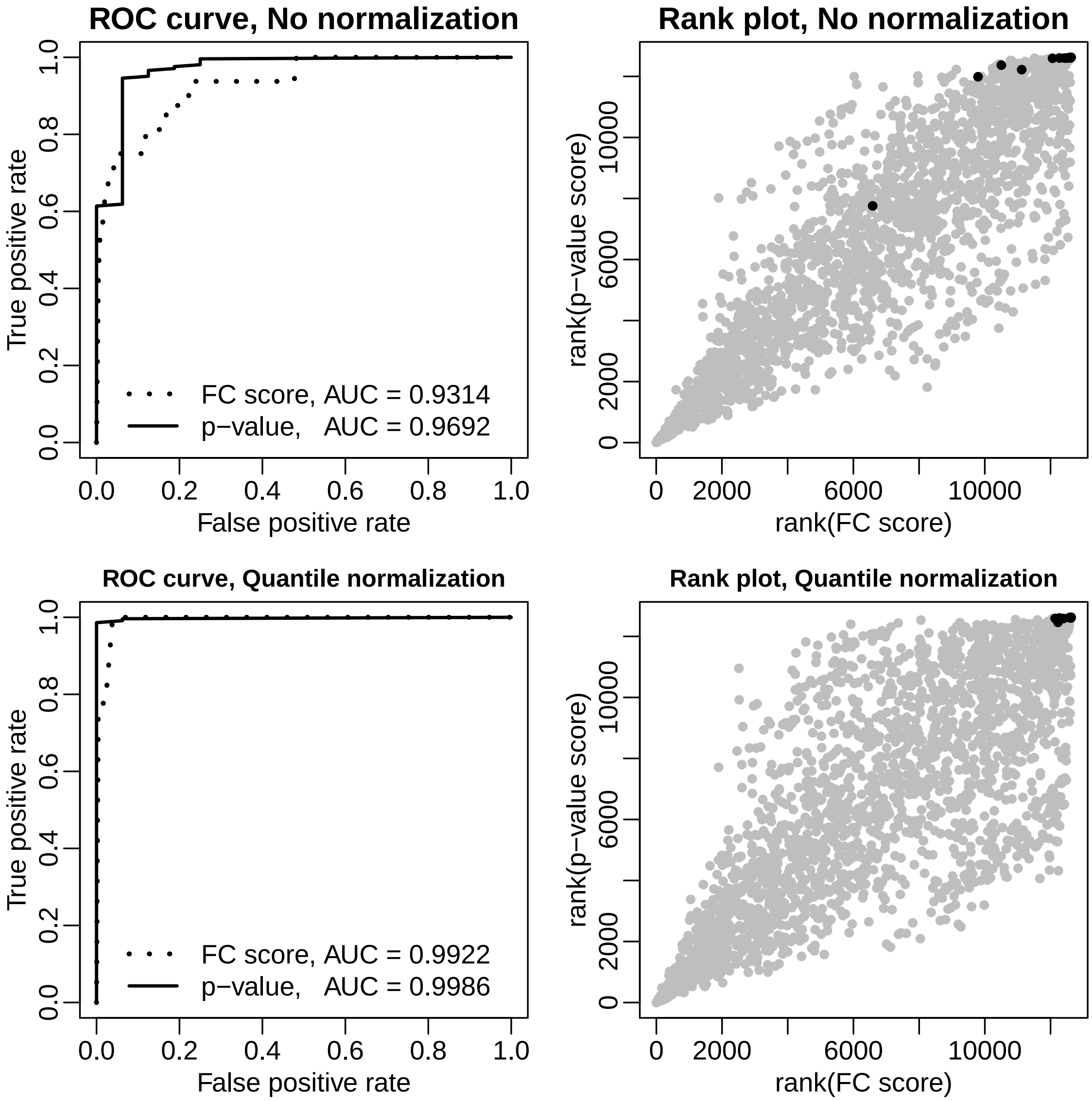

Figure 7: ROC curve and the rank plot for the PSVD Model. ROC curve for comparing the p-value and FC scores is given in the left column, and the rank plot for the rank(p-value score) vs. rank(FC score) is given in the right column. Upper row represents no normalization and the lower row represents the quantile normalization technique.

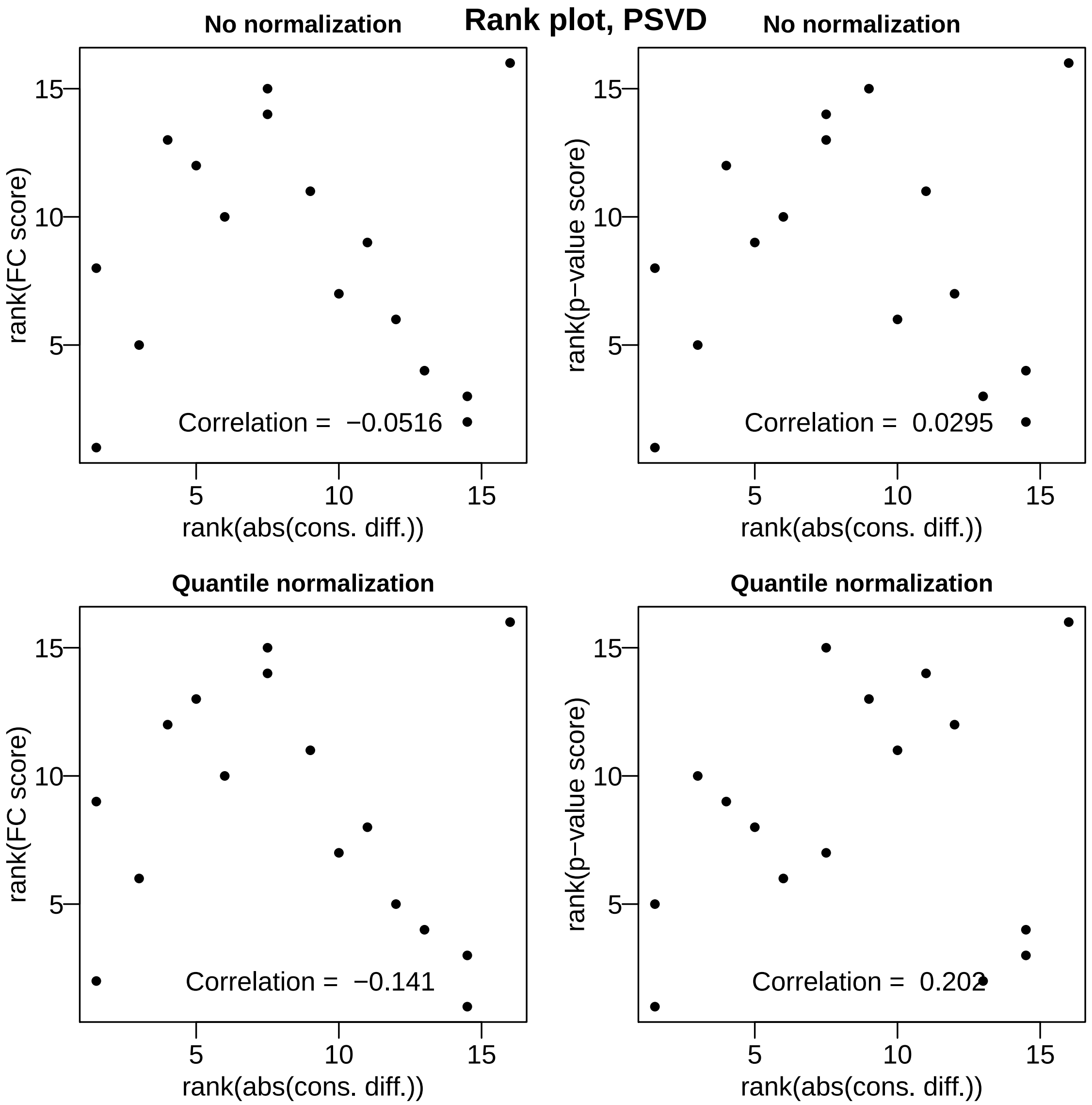

Figure 8: Rank plot for the 16 spiked-in genes based on the PSVD Model. Rank plot for the rank(FC score) vs. rank(abs(cons. diff.)) is given in the left column, and the rank plot for the rank(p-value score) vs. rank(abs(cons. diff.)) is given in the right column. Upper row represents no normalization and the lower row represents the quantile normalization technique.

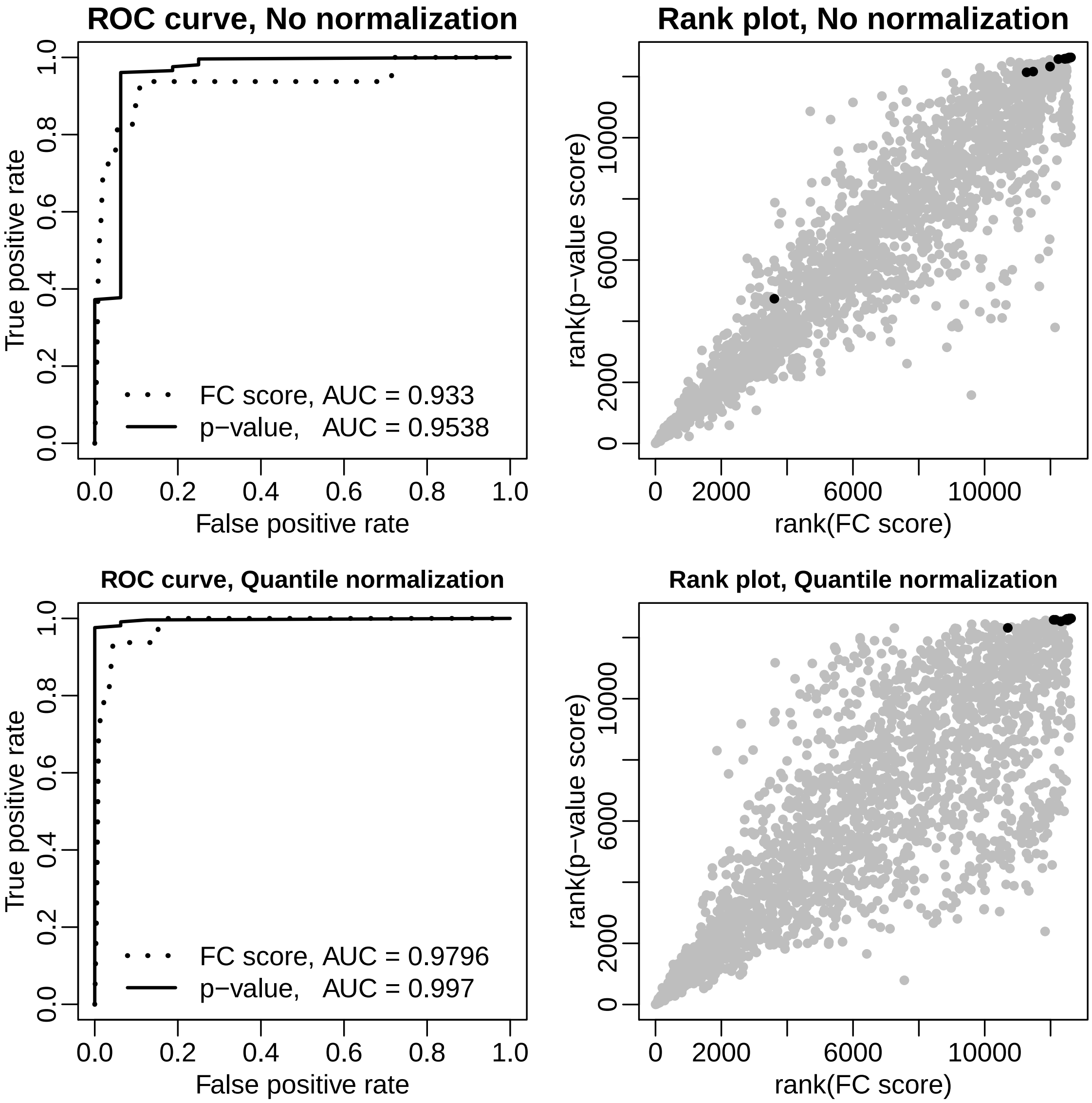

Figure 9: ROC curve and the rank plot for the Li-Wong Model. ROC curve for comparing the p-value and FC scores is given in the left column, and the rank plot for the rank(p-value score) vs. rank(FC score) is given in the right column. Upper row represents no normalization and the lower row represents the quantile normalization technique.

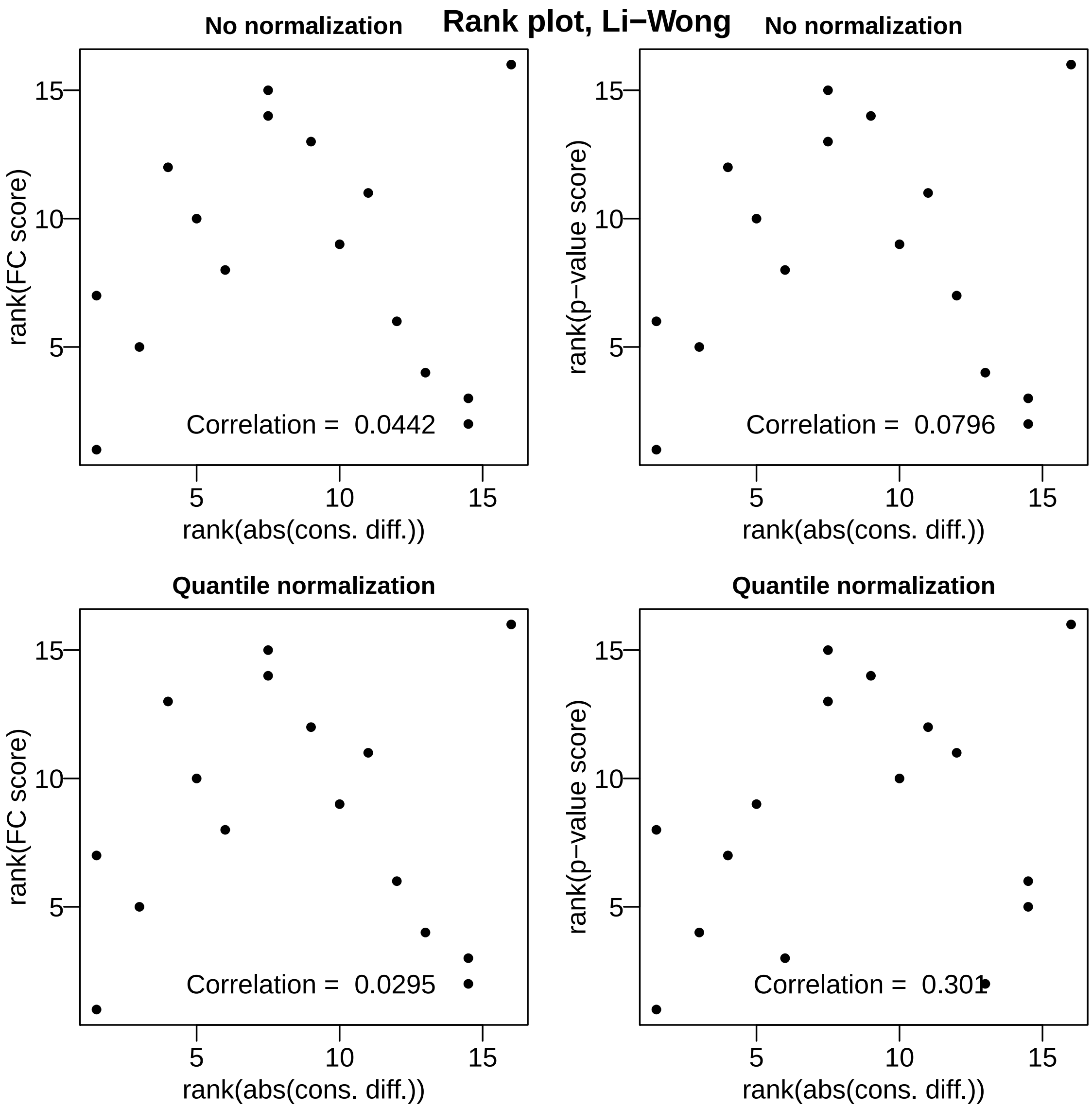

Figure 10: Rank plot for the 16 spiked-in genes based on the Li-Wong Model. Rank plot for the rank(FC score) vs. rank(abs(cons. diff.)) is given in the left column, and the rank plot for the rank(p-value score) vs. rank(abs(cons. diff.)) is given in the right column. Upper row represents no normalization and the lower row represents the quantile normalization technique.

CHAPTER III

INTEGRATING DATA TRANSFORMATION IN PRINCIPAL COMPONENTS ANALYSIS

The collected data points over time sequences and/or in ordinal spatial structure motivate the use of functional Principal Component Analysis (FPCA). However, PCA and FPCA does not work well when there are outliers or the data distribution is skewed. One popular solution is to transform the data to resolve this abnormal behavior caused from skewness or presence of outliers. Usually, such transformations can be obtained based on extensive data analysis, previous studies, or prior knowledge of expertise. In this work, we present an automatic procedure to achieve this goal based on a statistical model with extensions for handling the missing data and functional data structure. The proposed technique transforms the data to vanish the skewness of the data distribution; and simultaneously perform the functional PCA procedure. The new method is cast into a profile likelihood framework for efficient computation.

3.1 Introduction

In general, the necessity of a transformation can come from the particular statistical analysis to be performed, i.e., assumption of mean zero Gaussian distribution for residuals. Furthermore, the data transformation technique can help for better visualization as well as interpretability of the results. Also, variance-stabilizing transformations aim to remove the mean/variance relationship, i.e., Hawkins (1989) discussed the asymptotic distribution of the Fisher transformation applied to the sample correlation. Furthermore, Bar-Lev and Enis (1988) introduced a class of variance stabilizing transformations, which includes the Anscombe transform (Anscombe, 1948) for Poisson, binomial and negative binomial distributions. One of the most popular method is the Power transformation (Box and Cox, 1964). The power (Box-Cox) transform is parameterized by a nonnegative parameter β, that includes the logarithm, square root, and multiplicative inverse as special cases. Since the power transform family includes the identity transform as a special case, the estimation of the parameter β can be used to identify a transformation that is approximately the best for the provided model setting if it is necessary. This method is widely used in various fields of statistical data analysis, medical research, modeling of physical processes, and geochemical data analysis. In regression analysis, this approach is known as the Box-Cox technique. In this work, we want to extend this technique to the functional Principal Component Analysis.

PCA is commonly known dimension reduction technique that transform a class of correlated variables into a smaller class of uncorrelated variables called principal components. PCA is known for more than 100 years, and closely related to the factor analysis. It can be obtained by either eigenvalue decomposition of the sample covariance or the singular value decomposition (SVD) of the data matrix. It is well known that this technique is sensitive to outliers, and skewness. Various robust alternative procedures have been proposed (see for example Croux and Ruiz-Gazen, 2005; Higuchi and Eguchi, 2004; Hubert et al., 2002; Locantore et al., 1999; Maronna, 2005; Hubert et al., 2009). PCA also can be used as a key tool for unsupervised functional data analysis (Ramsay and Silverman, 2005). Rice and Silverman (1991) and Silverman (1996) presented the functional principal components by maximizing the variance of a standardized linear combination of variables based on two different approaches to impose smoothness on principal components using roughness penalty. Huang et al. (2008) proposed an alternative approach using Penalized lower rank matrix approximation to the data matrix. In this study, the focus is based on the latter approach.

The Data transformation technique has been used as a preprocessing tool before PCA and functional PCA analysis (i.e., Huang et al., 2008; Hu et al., 2006). Similarly it has been used before investigating the structure of the covariance matrix (i.e., Zimmerman and Núñez Antón, 2009), which has a close connection with PCA. However, there is no automatic statistical procedure to obtain such transformations. Generally the authors proposed their transformation based on evidences of non-normality of the data. They showed normal behavior of the dataset after the desired transformation. Mostly, the proposed transformation has been obtained based on extensive data analysis, previous studies or the experience of expertise. Therefore, there is no automatic procedure to select the appropriate transformation and it has been done manually in the literature. In order to produce the automatic procedure we need to utilize the connection of functional PCA and a probabilistic model. Herein we propose a statistically elegant procedure to estimate the appropriate transformation automatically. The transformation model can be written as

$$f(y_{ij}|\boldsymbol{\eta}) = (\Psi_{i.})^{\top}\Phi_{j.} + \epsilon_{ij}, \tag{3.1}$$

where $f(\cdot|\boldsymbol{\eta})$ is a monotonic function, $\boldsymbol{\eta}$ is the vector of transformation parameters, y_{ij} is the measurement of the j^{th} variable for the i^{th} observation, ($\{f(y_{ij}|\boldsymbol{\eta})\}$ is an $n \times m$ matrix), ϵ_{ij}'s are independent normal random errors with mean 0 and constant variance σ^2, Ψ is an $n \times d$ and Φ is an $m \times d$ matrices ($d \leq \min(m, n)$). The goal of this transformation model is to stabilize the variance of the residuals and satisfy the normality assumption of the residuals. Ψ and Φ represents the mappings of observations and variables to the smaller dimensions respectively. In the FPCA context the columns of Φ contains the functional principal components.

For simplicity, at the first step we skip the functional structure of the principal components and obtain an estimation procedure for the proposed model disregarding the smoothness penalty term and discuss the possible difficulties in the optimization procedure for this special case. Next, we consider another common restriction seen by data analysts. The reports of missing observations are commonly observed fact in variety of studies, but not always there is a quick solution to incorporate these missing data appropriately in the modeling structure. Studying the missing data mechanism is a crucial step and it has been studied widely in literature i.e., Daniels and Hogan (2008). The proposed model is highly tied to the SVD of the data matrix and hence it captures the most informative dependence structure based on the leading singular vectors of the dataset. However, we are not able to use the SVD directly in presence of missing observations. We showed that, considering the proposed probabilistic PCA model this concern can be handled based on some iterative procedures. Finally our focus in the later part of this work goes to the smoothed functional principal components analysis. The Functional data, observed in discrete observation points $t_1, \dots, t_m$ is considered and by using penalized likelihood approach we extend our model to the functional domain.

We review the likelihood based estimation procedure to obtain the transformation parameters automatically in subsection 3.2.1. Handling the missing data and the implementation of the functional structure for the variables are given in subsections 3.2.2, and 3.2.3 respectively. We use simulations and two datasets to demonstrate the applicability of the proposed model in subsection 3.3. Some concluding remarks are given in Chapter V.

3.2 Methodology

3.2.1 PSVD Algorithm

We consider the transformation model (3.1), and denote the parameter vector $\boldsymbol{\Theta}$, as $\boldsymbol{\Theta} = (\boldsymbol{\eta}^\top, \text{vec}(\Psi), \text{vec}(\Phi), \sigma^2)^\top$, and write out the log-likelihood function as

$$\begin{aligned} \ell(\boldsymbol{\Theta}) &= -\frac{1}{2\sigma^2}||\{f(y_{ij}|\boldsymbol{\eta})\} - \Psi\Phi^\top||^2 \\ &\quad -\frac{nm}{2}\log(\sigma^2) + \sum_{i=1}^{n}\sum_{j=1}^{m}\big\{\log|f'(y_{ij}|\boldsymbol{\eta})|\big\}. \end{aligned} \tag{3.2}$$

One can note that for any invertible matrix Δ, by letting $\tilde{\Psi} = \Psi(\Delta)^\top$, and $\tilde{\Phi} = \Phi(\Delta)^{-1}$, we obtain the same log-likelihood function as with Ψ and Φ. We resolve this identifiability issue by the following constraints:

- $\Psi^\top\Psi$ is a diagonal matrix.
- $\Phi^\top\Phi$ is the identity matrix.

It is noticeable that maximization of (3.2) with respect to $\boldsymbol{\Theta}$ simultaneously is computationally expensive. Hence, we turn to the profile likelihood method which comprises of two phases. In the first phase, we consider the transformation parameter vector $\boldsymbol{\eta}$ to be fixed at $\boldsymbol{\eta_0}$. The parameters to be estimated are only Ψ, Φ, and σ^2. The maximum likelihood estimates (MLEs) of the parameters can be viewed as functions of $\boldsymbol{\eta_0}$, and can be easily obtained using the nice result of connection between SVD and the least square estimates (or equivalently MLEs with normal residuals) referred to Stewart (1993) as the Approximation theorem (a.k.a EckartYoung theorem). The explicit forms of the parameter estimates are

$$\begin{aligned} \widehat{\Psi(\boldsymbol{\eta_0})} &= U_d\Sigma_d, & \widehat{\Phi(\boldsymbol{\eta_0})} &= V_d, \\ \widehat{\sigma^2(\boldsymbol{\eta_0})} &= \frac{1}{nm}||\{f(y_{ij}|\boldsymbol{\eta_0})\} - \widehat{\Psi(\boldsymbol{\eta_0})}\widehat{\Phi(\boldsymbol{\eta_0})}^\top||^2, \end{aligned} \tag{3.3}$$

where having the SVD matrix of the $\{f(y_{ij}|\boldsymbol{\eta_0})\} = U\Sigma V^\top$, U_d and V_d are the first d columns of U and V, respectively, corresponding to the d largest singular values in a diagonal matrix, Σ_d. We do not need the full SVD, and efficient algorithm for low-rank matrix approximation can be used to speed up the calculation of the leading vectors (e.g. Achlioptas and Mcsherry, 2007).

In the second phase, we aim to obtain the estimate of $\boldsymbol{\eta}$ via maximizing the profile log-likelihood function

$$\begin{aligned} \ell_p(\boldsymbol{\eta}) &= -\frac{1}{2\sigma^2(\boldsymbol{\eta})}||\{f(y_{ij}|\boldsymbol{\eta})\} - \Psi(\boldsymbol{\eta})\Phi(\boldsymbol{\eta})^\top||^2 \\ & \quad -\frac{nm}{2}\log(\sigma^2(\boldsymbol{\eta})) + \sum_{i=1}^{n}\sum_{j=1}^{m}\big\{\log|f'(y_{ij}|\boldsymbol{\eta})|\big\}. \end{aligned} \tag{3.4}$$

Notice that the profile log-likelihood function only contains the vector of parameters $\boldsymbol{\eta}$ with all the other parameters being expressed as the functions of $\boldsymbol{\eta}$. The variety of appropriate optimization techniques such as Downhill Simplex or gradient based algorithms (e.g. Avriel, 1976) can be used subject to the structure of the family of transformations. It is worth emphasizing that the obtained estimates based on the profile likelihood are the MLEs of the parameters, and thus have good theoretical properties.

We abbreviate the whole profiling and singular value decomposition procedure as PSVD. The general iterative optimization procedure are as follows:

1. Start from an initial estimate of $\boldsymbol{\eta}$, denoted by $\boldsymbol{\eta}_t$ $(t = 0)$. Usually we pick the initial estimate associated to the model with no transformation.

2. Let X_t be the $n \times m$ matrix with $(i, j)^{\text{th}}$ entry $f(y_{ij}|\boldsymbol{\eta_t})$. Perform SVD $X_t = U\Sigma V^\top$ and use (3.3) to obtain $\widehat{\Psi(\boldsymbol{\eta_t})}$, $\widehat{\Phi(\boldsymbol{\eta_t})}$, and $\widehat{\sigma^2(\boldsymbol{\eta_t})}$.

3. Obtain the updated value of $\boldsymbol{\eta_{t+1}}$ via an optimization algorithm to increase the value of the profile log-likelihood function $\ell_p(\boldsymbol{\eta_t})$ defined in (3.4).

4. Iterate between the last two steps until convergence is reached.

Given this general framework, we focus on the popular Box-Cox transformation family as an example hereafter. Let Y denote the untransformed data. The Box-Cox transformation for each element of Y is defined as

$$f(y_{ij}|\beta) = \begin{cases} \dfrac{y_{ij}^{\beta} - 1}{\beta} , & \beta \neq 0, \\ \log(y_{ij}) , & \beta = 0. \end{cases} \tag{3.5}$$

Since the transformation parameter $\boldsymbol{\eta} = \beta$ is one dimensional, the 3^{rd} step of the PSVD procedure becomes a simple optimization problem for a univariate concave function. In our implementation, we adopt the popular L-BFGS-B optimization algorithm (Byrd et al., 1994).

3.2.2 Handling the Missing Data

Dealing with missing values is one of the practical issues in variety of statistical methods. Different approaches have been developed to resolve the incompleteness of the datasets. While some uses an objective function, i.e., maximum likelihood estimation, others rely on imputation techniques. In the cases that the missing values are non-ignorable or so called "MNAR" (Missing Not at Random), the sample will not be an unbiased representative for the population of the interest. Therefore, it is difficult to correctly estimate the population parameters, and we need to take in to account the missing data mechanism. The detailed explanation can be found in Daniels and Hogan (2008). Here we consider the missing observations have the ignorable structure. Now let's consider the case that we did not observe the whole data matrix Y, instead we have the data matrix Y_m with some missing observations. Let's define an indicator matrix I_m such that, the $(i, j)^{\text{th}}$ elements of I_m is set to be 1, if it has been observed and it is set to be 0, otherwise. Similarly we can define the logical matrix I_m^c to be $\mathbf{1}\mathbf{1}^\top - I_m$. Let's use the symbol $\odot$ for the Schur product, which is defined as the entrywise product of two matrices of the same dimensions, and in analogy we can also define symbol $\oslash$ for the Schur division. One can easily note that, for a fixed transformation parameter η_0, there is no closed form solution for the model (3.1) anymore, since we can not obtain the SVD of the incomplete data matrices. Although there is no immediate solution based on the SVD technique, we present two algorithms which are highly tied to the SVD of the complete data matrices and consequently we have all of the nice properties of the SVD of a matrix.

Without loss of generality, let's fix the transformation parameter to be $\boldsymbol{\eta}_0$. We would like to obtain the MLEs based on observed data for the given $\boldsymbol{\eta}_0$. This can be done using the "Generalized Expectation-Maximization Algorithm" (a.k.a. GEM Algorithm, Dempster et al., 1977). Let's define X to be the complete $m \times n$ data matrix, where the observed elements are $I_m \odot \{f(y_{ij}|\boldsymbol{\eta}_0)\}$, and $I_m^c \odot X$ are the missing elements. Consider the following complete log-likelihood function:

$$\begin{aligned}\ell_c(\boldsymbol{\Theta}) &= -\frac{1}{2\sigma^2}||X - \Psi\Phi^\top||^2 - \frac{nm}{2}\log(\sigma^2)\\ &= -\frac{1}{2\sigma^2}\left\{\sum_{(i,j)\in I_m}(x_{ij} - (\Psi_{i.})^\top\Phi_{j.})^2 - \sum_{(i,j)\in I_m^c}(x_{ij} - (\Psi_{i.})^\top\Phi_{j.})^2\right\}\\ &\quad -\frac{nm}{2}\log(\sigma^2).\end{aligned}$$

In the E-step, let's define $Q(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)}) = E(\ell_c(\boldsymbol{\Theta})|I_m \odot X, \boldsymbol{\Theta}^{(t)})$, where $\boldsymbol{\Theta}^{(t)}$ is the estimated parameters in the t^{th} step. It's easy to see that

$$\begin{aligned}Q(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)}) &= -\frac{1}{2\sigma^2}\left\{\sum_{(i,j)\in I_m}(x_{ij} - (\Psi_{i.})^\top\Phi_{j.})^2 - N(I_m^c)\sigma^{(t)2}\right\}\\ &\quad -\frac{nm}{2}\log(\sigma^2),\end{aligned}$$

where $N(I_m^c)$ is the number of missing observations. In the M-step we need to maximize the $Q(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)})$ which does not seems to be intuitive, while by introducing the new matrix $Z^{(t)}$ as following:

$$Z^{(t)} = \begin{cases} \Psi_{i.}^{(t)\top}\Phi_{j.}^{(t)} \ , & \text{(i,j) is missing,} \\ x_{ij} \ , & \text{otherwise,} \end{cases}$$

we can define a new objective function $\tilde{Q}(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)})$, as

$$\begin{aligned} \tilde{Q}(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)}) &= -\frac{1}{2\sigma^2}||Z^{(t)} - \Psi\Phi^\top||^2 \\ &\quad -\frac{nm}{2}\log(\sigma^2) - \frac{N(I_m^c)\sigma^{(t)2}}{2\sigma^2}. \end{aligned}$$

Defining the SVD of the matrix $Z^{(t)} = U\Sigma V^\top$, the updated values for $\boldsymbol{\Theta}^{(t+1)}$ which maximizes the $\tilde{Q}(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)})$, will have the following form:

$$\begin{aligned} \Psi^{(t+1)} &= U_d\Sigma_d, & \Phi^{(t+1)} &= V_d, \\ \sigma^{(t+1)2} &= \frac{1}{nm}||Z^{(t)} - \Psi^{(t+1)}\Phi^{(t+1)^\top}||^2 + \frac{N(I_m^c)\sigma^{(t)2}}{nm}. \end{aligned}$$

Note that $\boldsymbol{\Theta}^{(t+1)}$ that maximizes $\tilde{Q}(\boldsymbol{\Theta}|\boldsymbol{\Theta}^{(t)})$ will satisfy the following condition:

$$Q(\boldsymbol{\Theta}^{(t+1)}|\boldsymbol{\Theta}^{(t)}) \geq Q(\boldsymbol{\Theta}^{(t)}|\boldsymbol{\Theta}^{(t)}),$$

and hence the following algorithm is the results of merging the GEM steps:

Algorithm 1:(GEM Algorithm)

1. Let $X_0 = I_m \odot \{f(y_{ij}|\eta_0)\} + I_m^c \odot \{0\}$. Set $t \leftarrow 0$.

2. Obtain the SVD for the matrix X_t ($X_t = U\Sigma V^\top$).

3. Define $X_{t+1} = I_m \odot \{f(y_{ij}|\eta_0)\} + I_m^c \odot (U_d\Sigma_d V_d^\top)$. Set $t \leftarrow t+1$.

4. Iterate between the last two steps until convergence is reached.

The obtained GEM Algorithm has been reported previously in Beckers and Rixen (2003). Hereby we showed that this algorithm has connection with the MLEs of the missing values under the normal model. It has the advantage of obtaining the first d components of the SVD simultaneously together based on the maximizing the complete log-likelihood function; but similar to the other EM approaches, it is very slow. Faster solutions can be obtained by extending the known power algorithm based on the observed data as given in the following algorithm.

Algorithm 2:(Power Algorithm)

1. Let $X = I_m \odot \{f(y_{ij}|\eta_0)\} + I_m^c \odot \{0\}$. Set $k \leftarrow 1$.

2. Obtain the rank one approximation matrix to X as $\sigma_0 \boldsymbol{u}_0 \boldsymbol{v}_0^\top$. Set $t \leftarrow 1$.

3. Set $\boldsymbol{u}_{t+1} = \{(I_m \odot X)\boldsymbol{v}_t\} \oslash \{I_m \boldsymbol{v}_t^2\}$, and $\boldsymbol{v}_{t+1} = \{(I_m \odot X)^\top \boldsymbol{u}_t\} \oslash \{I_m^\top \boldsymbol{u}_t^2\}$. Set $t \leftarrow t+1$, and normalize $\boldsymbol{u}_t$, and $\boldsymbol{v}_t$ to the norm one vectors.

4. Repeat step 3 until convergence. After convergence, let $U_{.k} = \boldsymbol{u}_t$, $V_{.k} = \boldsymbol{v}_t$, and $\Sigma_{kk} = \dfrac{\boldsymbol{u}_t^\top (I_m \odot X)\boldsymbol{v}_t}{\boldsymbol{u}_t^{2\top} I_m \boldsymbol{v}_t^2}$.

5. Set $k \leftarrow k+1$. If $k \leq d$ then

$$\{\text{let } X = X - I_m \odot (\Sigma_{kk} U_{.k} V_{.k}^\top), \text{ and go to step 2.}\}.$$

The second algorithm is the generalization of the power algorithm based on the observed values to obtain the first d components of SVD sequentially. It is much faster than the first algorithm; but it may suffer from the higher computational inaccuracy based on the round-off error arising on the sequential procedure. Note that, it can be shown that the simultaneous iteration of the power algorithm cannot be extended to obtain the SVD components together when the missing values are involved. In each run of the simultaneous iteration (Trefethen and Bau, 1997), the QR decomposition is called to obtain orthogonalized directions, so the reason that the simultaneous iteration fails in this setup is not considering the missing data structure in obtaining the QR decomposition.

Proposition 1. Using the algorithms 1 or 2, the MLEs of the parameters for model (3.1) in presence of missing values based on the observed log-likelihood is

$$\begin{aligned} \widehat{\Psi(\boldsymbol{\eta_0})} &= U_d\Sigma_d, \qquad\qquad \widehat{\Phi(\boldsymbol{\eta_0})} = V_d, \\ \widehat{\sigma^2(\boldsymbol{\eta_0})} &= \frac{1}{N(I_m)}||I_m \odot \left(\{f(y_{ij}|\boldsymbol{\eta_0})\} - \widehat{\Psi(\boldsymbol{\eta_0})}\widehat{\Phi(\boldsymbol{\eta_0})}^\top\right)||^2, \end{aligned} \tag{3.6}$$

where $N(I_m)$ is the total number of the observed data points, and U_d, V_d, and Σ_d are obtained from one of the above algorithms.

It can be shown that, the algorithms 1 and 2 are the maximizer of the following observed profile log-likelihood function for a fixed $\boldsymbol{\eta}_0$:

$$\begin{aligned} \ell_p(\boldsymbol{\eta}) &= -\frac{1}{2\sigma^2(\boldsymbol{\eta})}||I_m \odot \left(\{f(y_{ij}|\boldsymbol{\eta})\} - \Psi(\boldsymbol{\eta})\Phi(\boldsymbol{\eta})^\top\right)||^2 \\ &\quad -\frac{N(I_m)}{2}\log(\sigma^2(\boldsymbol{\eta})) + \sum_{(i,j)\in I_m}\left\{\log|f'(y_{ij}|\boldsymbol{\eta})|\right\}. \end{aligned} \tag{3.7}$$

As a direct consequence of the Proposition 1, we may update the PSVD algorithm to the PSVDM (PSVD with missing data). The main differences are: 1- The $\widehat{\Psi(\boldsymbol{\eta_0})}$, $\widehat{\Phi(\boldsymbol{\eta_0})}$, and $\widehat{\sigma^2(\boldsymbol{\eta_0})}$ will not be the direct solution of the SVD of the data matrix anymore and we need to use either the GEM algorithm or the Power algorithm to obtain it in the 2nd step of the algorithm. 2- The objective function in the 3rd step is not the profile log-likelihood (3.4) anymore and we should substitute it with the observed profile log-likelihood (3.7). Hence the PSVDM Algorithm can be written as follows:

1. Start from an initial estimate of $\boldsymbol{\eta}$, denoted by $\boldsymbol{\eta}_t$ $(t = 0)$. Usually we pick the initial estimate associated to the model with no transformation.

2. Use either the GEM or the Power algorithms and (3.6) to obtain $\widehat{\Psi(\boldsymbol{\eta_t})}$, $\widehat{\Phi(\boldsymbol{\eta_t})}$, and $\widehat{\sigma^2(\boldsymbol{\eta_t})}$.

3. Obtain the updated value of $\boldsymbol{\eta_{t+1}}$ via an optimization algorithm to increase the value of the observed profile log-likelihood function $\ell_p(\boldsymbol{\eta_t})$ defined in (3.7).

4. Iterate between the last two steps until convergence is reached.

3.2.3 Functional Data Structure

Three different approaches from Rice and Silverman (1991), Silverman (1996), and Huang et al. (2008) to smoothed functional principal components analysis (FPCA) were proposed. The two early works are based on maximizing the variance but with different penalties, while the latest approach is based on penalized rank one approximation of the data matrix. We would like to take in to account the smoothness of the principle components to our PSVD model similar to the Huang et al. (2008) works, since (a) it is invariance under scale transformation of the measurements; (b) it can naturally incorporates spline smoothing of discretized functional data; (c) the connection with smoothing splines let us to use cross-validation or generalized cross-validation criteria for smoothing parameter selection; (d) different smoothing parameters are permitted for different FPCs.

The penalized profile log-likelihood based on $\boldsymbol{\eta}$, $\ell_p^*(\boldsymbol{\eta})$ can be obtained by introducing a roughness penalty matrix Ω over Φ and including this term in (3.4), so we have

$$\ell_p^*(\boldsymbol{\eta}) = \ell_p(\boldsymbol{\eta}) - \text{pen}(\Phi).$$

Following the Huang et al. (2008), we can define the penalty term based on different smoothing parameters α_ks as follows:

$$\ell_p^*(\boldsymbol{\eta}) = \ell_p(\boldsymbol{\eta}) - \sum_{k=1}^{d} \alpha_k \Psi(\boldsymbol{\eta})_{.k} {\Psi(\boldsymbol{\eta})_{.k}}^\top \Phi(\boldsymbol{\eta})_{.k} \Omega {\Phi(\boldsymbol{\eta})_{.k}}^\top.$$

This is the penalized profile log-likelihood function in general framework, which has the advantage of allowing different smoothing parameters for different FPCs. In a simpler case we can have a common smoothing parameter α for all of the FPCs,

$$\ell_p^*(\boldsymbol{\eta}) = \ell_p(\boldsymbol{\eta}) - \alpha \mathrm{tr}(\Psi(\boldsymbol{\eta})\Psi(\boldsymbol{\eta})^\top \Phi(\boldsymbol{\eta})\Omega\Phi(\boldsymbol{\eta})^\top). \tag{3.8}$$

We focus on the last equation, that is given in (3.8), and derive the first d functional principal components together. Using a single smoothness parameter makes it easier to interpret the results, but the proposed algorithm is general; and similar results can be derived based on different smoothness penalties for different smoothed PC functions. Now by fixing $\boldsymbol{\eta}$, and $\Phi(\boldsymbol{\eta})$; and letting $Y(\boldsymbol{\eta}) = \{f(y_{ij}|\boldsymbol{\eta})\}$, the $\Psi(\boldsymbol{\eta})$ that maximizes the (3.8) is

$$\Psi(\boldsymbol{\eta}) = Y(\boldsymbol{\eta})\Phi(\boldsymbol{\eta})\{\Phi(\boldsymbol{\eta})^\top(I + \alpha\Omega)\Phi(\boldsymbol{\eta})\}^{-1}.$$

Plugging the $\Psi(\boldsymbol{\eta})$ back to the penalized profile log-likelihood, it can be shown that our criterion function is maximizing the

$$\mathrm{tr}\Big(\Phi(\boldsymbol{\eta})^\top Y(\boldsymbol{\eta})^\top Y(\boldsymbol{\eta})\Phi(\boldsymbol{\eta})\{\Phi(\boldsymbol{\eta})^\top(I + \alpha\Omega)\Phi(\boldsymbol{\eta})\}^{-1}\Big),$$

with respect to $\Phi(\boldsymbol{\eta})$. This is essentially the Silverman (1996) approach which we briefly describe it here. Let $X(\cdot)$ stands for a random function that can be observed repeatedly. To find the j^{th} smooth principal component weight function $\gamma_j(\cdot)$, the Silverman approach maximizes $\dfrac{\mathrm{var}(\int \gamma X)}{\int \gamma^2 + \alpha \int \gamma''^2}$, subject to $\alpha \int \gamma'' \hat{\gamma}_k'' = 0$ for $k < j$.

For a fixed parameter $\boldsymbol{\eta}$, we can use one of the two proposed algorithm given in Huang et al. (2008) to obtain the smoothed PCs. The first method is a variate of power algorithm and the second method is based on half smoothing. Moreover, they have proposed a cross-validation technique to choose the smoothness parameter, which can be borrowed in our algorithm as well.

Here the challenge is to estimate $\boldsymbol{\eta}$ based on the likelihood function and simultaneously tune up the smoothness parameter α based on the cross validation technique.

It is clear that relationship between this two parameters are twisted; therefore we use the back-fitting procedure to provide a reasonable estimate for these parameters in our algorithm.

First let's give a short explanation about the half-smoothing technique. Maximizing the penalized profile log-likelihood for fixed parameters $\boldsymbol{\eta}$, and α with respect to Ψ, and Φ can be simplified to the minimization of the following penalized reconstruction error criterion

$$||Y(\boldsymbol{\eta}) - \Psi\Phi^\top||^2 + \alpha \mathrm{tr}\big(\Psi^\top\Psi\Phi^\top\Omega\Phi\big),$$

and that can be simplified further to

$$||Y(\boldsymbol{\eta})||^2 - ||\widetilde{Y(\boldsymbol{\eta})}||^2 + ||\widetilde{Y(\boldsymbol{\eta})} - \Psi\widetilde{\Phi}^\top||^2.$$

Here $\widetilde{Y(\boldsymbol{\eta})} = Y(\boldsymbol{\eta})S^{1/2}(\alpha)$, and $\widetilde{\Phi} = S^{-1/2}(\alpha)\Phi$, where $S(\alpha) = (I + \alpha\Omega)^{-1}$. Hence Ψ, and $\widetilde{\Phi}$ can be easily estimated based on the first d components of the SVD of $\widetilde{Y(\boldsymbol{\eta})}$. Interpreting $S^{1/2}(\alpha)$ as a half-smoothing operator, the transformed matrix $\widetilde{Y(\boldsymbol{\eta})}$ is obtained by half-smoothing the rows of the transformed data matrix $Y(\boldsymbol{\eta})$. After $\widetilde{\Phi}$ is obtained as the first d right singular vectors of $\widetilde{Y(\boldsymbol{\eta})}$, we half-smooth it to obtain the smoothed PC function $\Phi = S^{1/2}(\alpha)\widetilde{\Phi}$. Following the given descriptions we provide the Functional PSVD (FPSVD) algorithm as below:

1. Start from an initial estimate of $\boldsymbol{\eta}$, denoted by $\boldsymbol{\eta}_t$ $(t = 0)$. Usually we pick the initial estimate associated to the model with no transformation.

2. Let X_t be the $n \times m$ matrix with $(i, j)^{\text{th}}$ entry $f(y_{ij}|\boldsymbol{\eta_t})$. Use the cross-validation technique given in subsection 3.2.3.1 to obtain the α for fixed $\boldsymbol{\eta}_t$.

3. Define $\widetilde{X}_t = X_t S^{1/2}(\alpha)$. Perform SVD $\widetilde{X}_t = U\Sigma\widetilde{V}^\top$ and let $V = S^{1/2}\widetilde{V}$ and use (3.3) to obtain $\widehat{\Psi(\boldsymbol{\eta_t})}$, $\widehat{\Phi(\boldsymbol{\eta_t})}$, and $\widehat{\sigma^2(\boldsymbol{\eta_t})}$.

4. Obtain the updated value of $\boldsymbol{\eta_{t+1}}$ via an optimization algorithm to increase the value of the penalized profile log-likelihood function $\ell_p^*(\boldsymbol{\eta_t})$ defined in (3.8).

5. Iterate between the last three steps until convergence is reached.

3.2.3.1 Choosing the Smoothing Parameter

Huang et al. (2008) developed computationally efficient cross-validation(CV) and generalized cross-validation (GCV) criteria for selecting smoothing parameters. For a fixed parameter $\boldsymbol{\eta}$, we would adopt the CV and GCV criteria, similar to the selecting the spline tuning parameter α in Green and Silverman (1994) for FPCA. The CV score is defined as

$$CV(\alpha) = \frac{1}{m}\sum_{j=1}^{m} \frac{[\{(I - S(\alpha))(Y(\boldsymbol{\eta})^\top U_d)\}_{jj}]}{(1 - \{S(\alpha)\}_{jj})},$$

and the GCV score is defined as

$$GCV(\alpha) = \frac{||V_d\Sigma_d - Y(\boldsymbol{\eta})^\top U_d||^2/m}{\{1 - \text{tr}(S(\alpha))/m\}^2}.$$

Here we used the SVD of $Y(\boldsymbol{\eta})S^{1/2}(\alpha) = U\Sigma\widetilde{V}$ and defined $V = S^{1/2}(\alpha)\widetilde{V}$. U_d and V_d are the first d columns of U and V respectively, and Σ_d is a diagonal matrix formed based on the d largest singular values of Σ.

The CV and the GCV scores given above can indeed be derived from the basic idea of cross-validation and generalized cross-validation (Craven and Wahba, 1979). The main difference here, compared to those for smoothing splines is that in this context we delete one column of $Y(\boldsymbol{\eta})$ at a time, rather than a point deletion (Huang et al., 2008).

3.3 Data Analysis

In the previous Section first we motivated the PSVD model, next we discussed the missing data scenario (PSVDM), and finally we explored the FPCA for the proposed model (FPSVD). In this Section we implement these techniques in simulations and two different datasets to demonstrate the applicability of each method in analyzing the real dataset.

3.3.1 Simulation

We conducted the following simulation studies to evaluate the performance of the FPSVD method. The data generating model is:

$$x_{ij} = u_{i1}v_1(t_j) + u_{i2}v_2(t_j) + \epsilon_{ij}, \quad i = 1, \ldots, n;\ j = 1, \ldots, m,$$

where $u_{i1} \overset{\text{i.i.d}}{\sim} N(0, \sigma_1^2)$, $u_{i2} \overset{\text{i.i.d}}{\sim} N(0, \sigma_2^2)$, and $\epsilon_{ij} \overset{\text{i.i.d}}{\sim} N(0, \sigma^2)$. The parameters set to be $n = m = 101$, $\sigma_1 = 40$, $\sigma_2 = 10$, $\sigma = 4$, and the 101 grid points t_j are considered equally distance. Denote the underlying functional principal components as following:

$$v_1(t) = \frac{1}{s_1}\{t + \sin(\pi t)\} \quad \text{and} \quad v_2(t) = \frac{1}{s_2}\cos(3\pi t),$$

where s_1 and s_2 are the normalizing constants to make v_1 and v_2 normalized unit vectors. First, we considered $\beta = 0.25$ in the Box-Cox transformation in the above simulation setup and generate one hundred simulated data set. We generated the errors (ϵ_{ij}'s) from mean zero normal distribution with the standard deviation of σ. The simulated data was generated from

$$y_{ij} = f^{-1}(x_{ij}|\beta = 0.25), \quad i = 1 \ldots m, \quad j = 1 \ldots n. \tag{3.9}$$

The FPSVD estimate of β based on the hundred datasets has the mean value of 0.2465 with the standard deviation 0.0256. The plot of the FPC components of the non-transformed data is contained in the top row panel of Figure 11. We also displayed the FPC components based on the FPSVD procedure in the bottom panel. The gray dashed line is used to show the true simulated FPC components. the noisy black curve is the result of the PCA and the black dashed line indicates the results

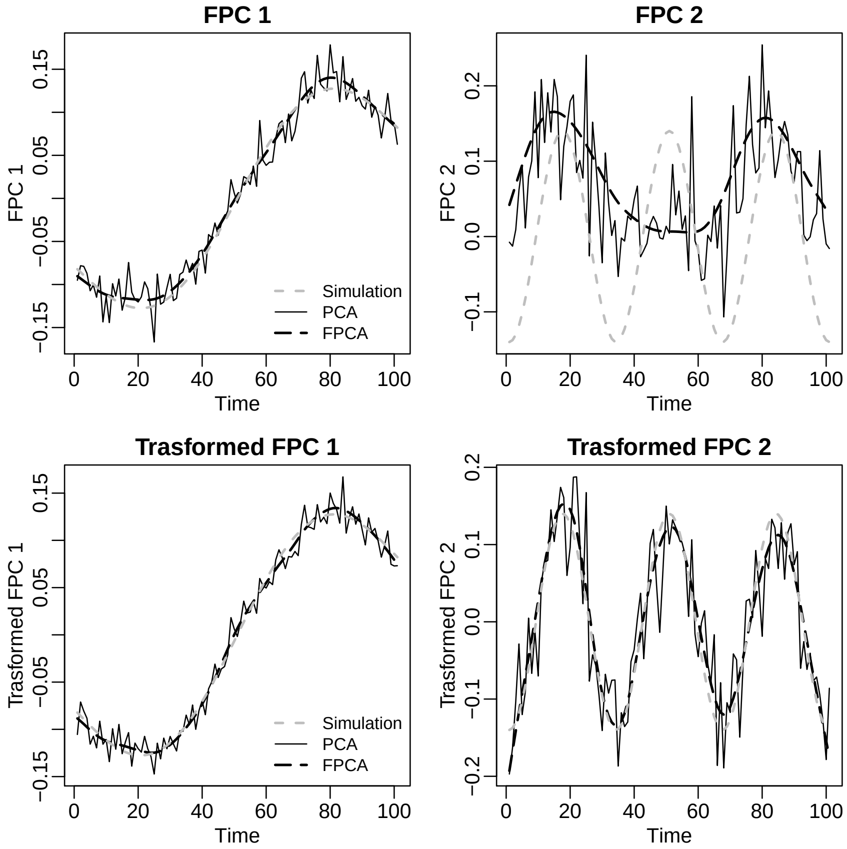

Figure 11: (First row:) The FPCs of the non-transformed data. (Last row:) the FPCs based on the FPSVD procedure. The gray dashed line is the true simulated FPCs. The noisy black curve is the result of the PCA and the black dashed line indicates the results of the FPCA.

of the FPCA. Although there is not much differences in the estimates of the first FPCs in this example, it is clear that by not using the FPSVD procedure we miss the true structure of the second FPC, and we may need to consider the third component while we know that the data has been generated from just two functional principal components. This is becoming worse as we decrease the value of β in the simulation setup.

We defined two measurements to compare the proposed model FPSVD, with the FPCA model. 1- The “Residual SVR” is defined to be the ratio of the first largest singular value to the second largest singular value obtained from SVD of the residual matrix. Since after subtracting the two FPCs, the residuals should be noninformative, obtaining a ratio close to one is desirable. 2- Since we know the true value of the FPCs at each grid t_j, $j = 1, \ldots, m$, we define the "SAD" to be the total sum of absolute deviation of the true FPCs and estimated FPCs. It is clear that, the smaller SAD is equivalent to better estimation of the FPCs.

Table 2 compares the transformed FPCA (T. FPCA) versus FPCA model for four different values of β, $\{0.5, 0.25, 0.1, 0.01\}$. For each β we generated one hundred simulated data sets based on the setup given as before. It is clear that using the FPSVD model the "SAD" and "Residual SVR" are independent of the value of β and those values are close to the true values. One can see that by not using the FPSVD the results that we are obtaining from the FPCA procedure becomes less reliable as the value of β is decreasing and it that become more crucial as we get closer to the logarithmic transformation. It is noticeable as the true β becomes closer to zero, the results obtained from the regular FPCA tends to become more unreliable.

We also performed the Shapiro Wilks test for normality of residuals in all of the cases and as we expected for the FPSVD procedures we obtained large p-values and for the common FPCA procedures the p-values are almost zero and we obviously reject the normality of residuals. Figure 12 shows this fact for the case $\beta = 0.25$. The top row panel is associated to the FPCA model and we see highly skewed pattern while in the transformed FPCA model we can see the normality of the residuals in the bottom row panel has been obtained.

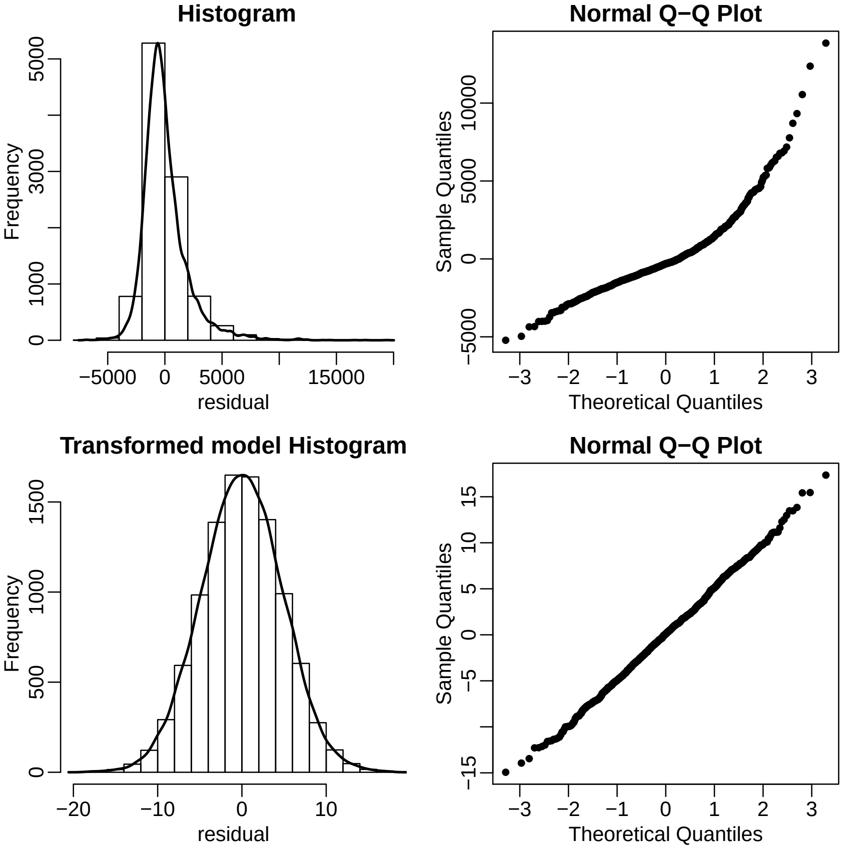

Figure 12: Normality of the residuals for the FPCA (first row) and the FPSVD (last row) for the simulation. The true β is 0.25.

Table 2: Comparisons between proposed transformation and no transformation model under different values of β.

Model-True β	$\hat{\beta}$	SAD	Residual SVR	Normality test
FPCA	-	4.55	1.04	0
$\beta = 0.5$	-	(0.48)	(0.37)	-
T. FPCA	0.4965	1.53	1.03	0.48
$\beta = 0.5$	(0.0516)	(0.32)	(0.02)	-
FPCA	-	11.52	1.09	0
$\beta = 0.25$	-	(1.10)	(1.58)	-
T. FPCA	0.2465	1.57	1.03	0.47
$\beta = 0.25$	(0.0256)	(0.43)	(0.02)	-
FPCA	-	15.24	1.34	0
$\beta = 0.1$	-	(1.68)	(4.38)	-
T. FPCA	0.0997	1.59	1.03	0.48
$\beta = 0.0$	(0.0029)	(0.29)	(0.02)	-
FPCA	-	18.12	4.93	0
$\beta = 0.01$	-	(1.17)	(18.02)	-
T. FPCA	0.0101	1.56	1.03	0.45
$\beta = 0.01$	(0.0012)	(0.30)	(0.02)	-

3.3.2 Datasets

3.3.2.1 Fruit Fly Mortality Data

To illustrate and assess the performance of our approach in presence of missing data, we consider the "fruit fly mortality" (FFM) data (Zimmerman and Núñez Antón, 2009). The FFM data are age-specific measurements of mortality for 112 cohorts of a common fruit fly, "Drosphila melanoster". Everyday, dead flies were counted for each cohort, and these counts were pooled into 11 5-day intervals. The raw mortality rate was recorded as $-\log\left(\frac{N(t+1)}{N(t)}\right)$, where $N(t)$ is the number of alive flies in the cohort at the beginning of time $t(t = 0, 1, \ldots, 10)$. For an unknown reasons 22% of the data is missing. We would like to investigate under what transformation we obtain more normally distributed responses with constant variability.

3.3.2.2 Call Center Data

The data contains the number of calls that got connected to a call center during every quarter hour from 7 : 00AM to midnight, weekdays between January 1 and October 26 in the year 2003. In total, there are 42 whole weeks and each day consists of 68 quarter hours. Let's denote the call volume during the j^{th} time interval on day i with y_{ij}. Huang et al. (2008) used the square-root transformation to stabilize variance and make the distribution close to normal. They choose the square root transformation manually, based on exploring the dataset and experience of expertise (i.e., Brown et al., 2005; Shen and Huang, 2008, used the same transformation previously). Here we would like to check the performance of the FPSVD algorithm in finding the appropriate transformation automatically and simultaneously obtaining the smooth principal components.

3.3.3 Results

First we considered the FFM data. Since it's a longitudinal study, so it is reasonable to obtain the functional principal components. Considering the 22% of the missing values, one can use the given algorithm in Beckers and Rixen (2003) to fill in the missing values and obtain the FPCs. Figure 13, shows the residual plot obtained from the FPCA procedure in the first row panel. It is clear that the residuals are far from normal, and presence of outliers confirms the heavy tail distribution of the errors which makes the imputation procedure to be not reliable. This opposes the normality assumption given in the Proposition 1. We have performs the FPSVD procedure and obtained the estimate of $\hat{\beta} = 0.007$. This is very close to logarithmic transformation that has been suggested by Zimmerman and Núñez Antón (2009). Looking to the residuals obtained from the FPSVD algorithm in the last row panel of the Figure 13, the behavior of the FPSVD residuals seems to be very close to normal family with

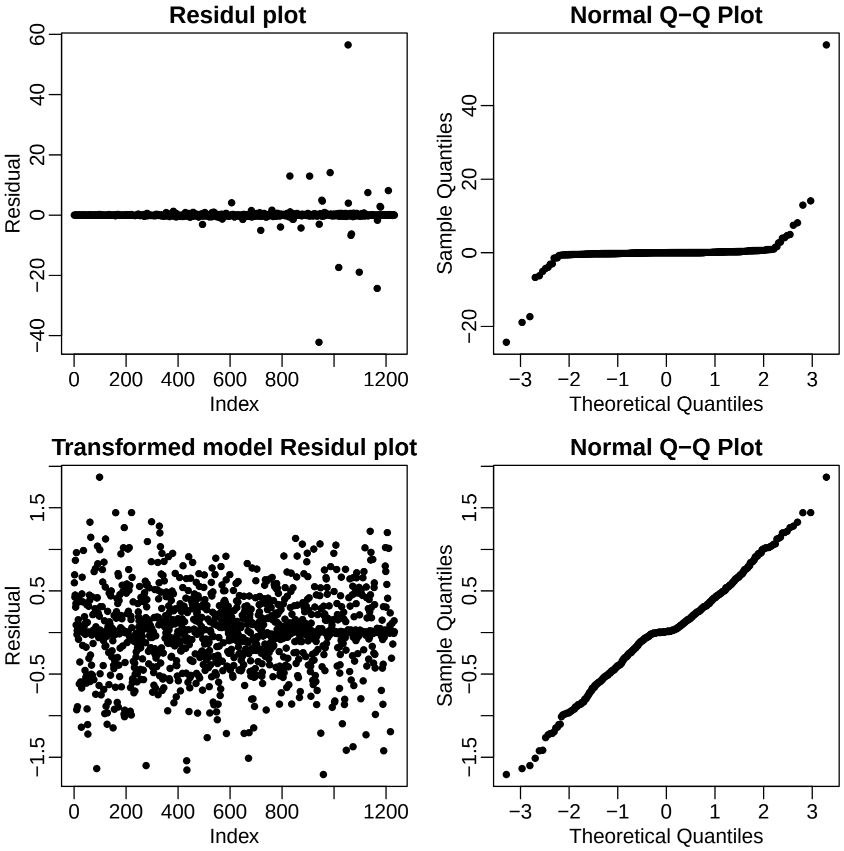

Figure 13: Residual plots and the QQ plots for the FFM data. First row: the FPCA results. Last row: the FPSVD results.

constant variability.

Finally, the FPSVD has been implemented for the call center data and the result for the estimated value of $\hat{\beta}$ is 0.517, which is almost same as the square root transformation. Here the suggested transformation has been obtained automatically and confirms the ad-hoc findings in Huang et al. (2008); Shen and Huang (2008), and Brown et al. (2005). Figure 14, the left panels shows the FPCA results. The bottom left panel for the residual plot indicates the non constant variability of the residuals, while the bottom right residual plot associated to the FPSVD model seems to have constant variability. The top right panels are associated to the FPCs of the FPSVD.

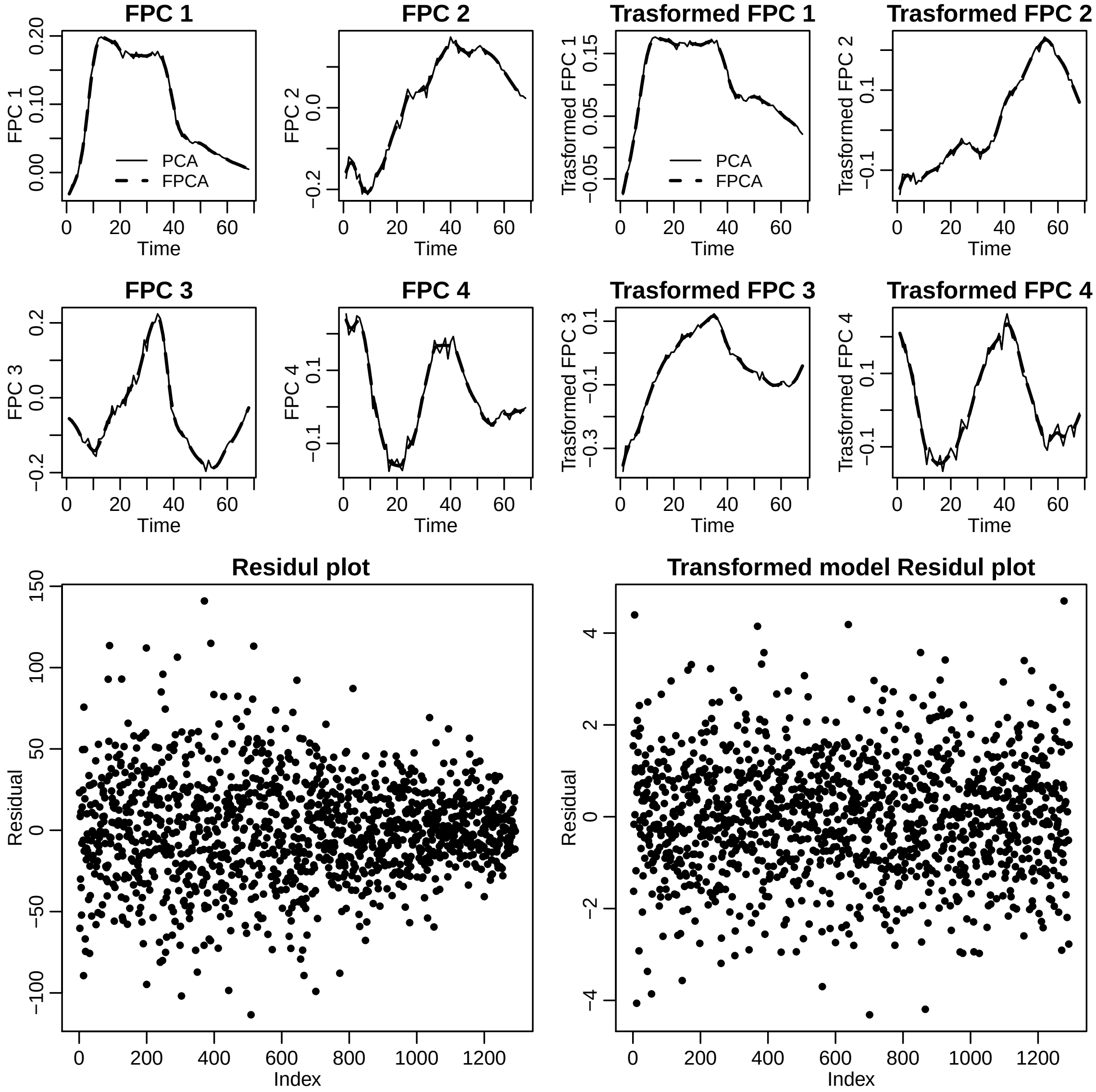

Figure 14: Call Center data: FPCs for the FPCA model (top left panels). FPCs for the FPSVD model (top right panels). Residual plots based on FPCA model (bottom left), and FPSVD model (bottom right).

CHAPTER IV

ROBUST ESTIMATION OF THE CORRELATION MATRIX OF LONGITUDINAL DATA

We propose a double-robust procedure for modeling the correlation matrix of a longitudinal dataset. It is based on an alternative Cholesky decomposition of the form $\boldsymbol{\Sigma} = \boldsymbol{D}\boldsymbol{L}\boldsymbol{L}^\top\boldsymbol{D}$ where $\boldsymbol{D}$ is a diagonal matrix proportional to the square roots of the diagonal entries of $\boldsymbol{\Sigma}$ and $\boldsymbol{L}$ is a unit lower-triangular matrix determining solely the correlation matrix. The first robustness is with respect to model misspecification for the innovation variances in $\boldsymbol{D}$, and the second is robustness to outliers in the data. The latter is handled using heavy-tailed multivariate t-distributions with unknown degrees of freedom. We develop a Fisher scoring algorithm for computing the maximum likelihood estimator of the parameters when the non-redundant and unconstrained entries of $(\boldsymbol{L}, \boldsymbol{D})$ are modeled parsimoniously using covariates. We compare our results with those based on the modified Cholesky decomposition of the form $\boldsymbol{L}\boldsymbol{D}^2\boldsymbol{L}^\top$ using simulations and a real dataset.

4.1 Introduction

Longitudinal data arise frequently in the biomedical, epidemiological and social sciences, where subjects are measured repeatedly over time and the observations on the same subject are intrinsically correlated (Diggle et al., 2002). The technique of generalized estimating equations (GEE) introduced in Liang and Zeger (1986) is widely used when the focus is on modeling the mean. In GEE and many of its extensions, in the interest of expediency, parsimony and ensuring the positive-definiteness of the estimated correlation matrix, it is common to pick a *working correlation* matrix, from a long menu of structured correlation matrices. Although consistency of the estimators of the mean parameters is not affected, misspecification of the correlation may result in a great loss of efficiency (Wang and Carey, 2003) and may lead to invalid inferences (Cannon et al., 2001; Carroll, 2003). The correlation matrix itself might be of scientific interest (Diggle and Verbyla, 1998) in which case it is desirable to develop a bona fide data-based framework for modeling correlation matrices following the familiar three stages of model formulation, estimation and diagnostics in the modeling process for the mean vector (McCullagh and Nelder, 1989). Attempts to develop such methods have been made in recent years by Chiu et al. (1996), Pourahmadi (1999, 2000), Pan and MacKenzie (2003), Ye and Pan (2006), Lin and Wang (2009), Leng et al. (2010) and references therein, using the spectral and Cholesky decompositions of covariance matrices, respectively.

A methodology based on the modified Cholesky decomposition (MCD) of the covariance matrix $\boldsymbol{\Sigma}$ of a random vector $\boldsymbol{y} = (y_1, \ldots, y_p)^\top$ has proved quite successful for longitudinal data in the sense that the positive-definiteness of the estimated covariance is guaranteed and parsimony can be achieved using covariates. However, it seems for historical reasons the focus has been mostly on specific transitional models of autoregressive (AR) type for the actual successive measurements on a subject:

$$y_t = \phi_{t,t-1} y_{t-1} + \ldots + \phi_{t,1} y_1 + \epsilon_t, \qquad t = 1, 2, \ldots, p, \tag{4.1}$$

where the $\phi_{t,j}$'s are the so-called generalized autoregressive parameters (GARPs) with $\phi_{1,0} = 0$, and ϵ_t's are the prediction errors or innovations with $Var(\epsilon_t) = \sigma_t^2$; see Pourahmadi (1999, 2000), Pan and MacKenzie (2003), Ye and Pan (2006), Lin and Wang (2009), and Leng et al. (2010). Although the idea of inverting the AR model (4.1) and writing it as a moving average (MA) of the actual response in terms of the present and past innovations was mentioned in (Pourahmadi, 2001, Sec. 3.5;

Rothman et al., 2010), the idea and its potential has not been pursued vigorously in the literature of longitudinal and correlated data. Given the duality and synergy between the AR and MA models in the theory of finite parameter stationary time series (Brockwell and Davis, 1991), one would expect a level of similar fruitful connections to exist between such type of models for nonstationary longitudinal data. For example, inverting (4.1) gives rise to the generalized moving average parameters (GMAPs) which are known (Pourahmadi, 2001, Sec. 3.5; Rothman et al., 2010) to be useful in parsimonious modeling and guaranteeing the positive-definiteness of $\boldsymbol{\Sigma}$ itself. These models, whether of AR or MA type, lead to a factorization of the form $\boldsymbol{\Sigma}^{\pm} = \boldsymbol{L}\boldsymbol{D}^2\boldsymbol{L}^{\top}$, where $\boldsymbol{L}, \boldsymbol{D}$ are generic unit lower triangular and diagonal matrices, respectively. Since $\boldsymbol{D}^2$ is trapped in the middle, the correlation matrix corresponding to $\boldsymbol{\Sigma}^{\pm}$ depends on the innovation variances represented by the diagonal entries of $\boldsymbol{D}^2$, and hence is not necessarily *robust* to their model misspecifications.

By contrast, there is an alternative Cholesky decomposition (ACD), due to Chen and Dunson (2003), which is of the generic form $\boldsymbol{\Sigma} = \boldsymbol{D}\boldsymbol{L}\boldsymbol{L}^{\top}\boldsymbol{D}$ with the diagonal matrix $\boldsymbol{D}$ of innovation standard deviations placed outside. Consequently, such factorization amounts to directly modeling the covariance matrix but in a manner that its estimated correlation matrix $\boldsymbol{R}$ does not depend on the quality of modeling and estimation of the innovation variances σ_t^2's, see (4.3). In other words, estimation of $\boldsymbol{R}$ is robust to misspecification of models for σ_t^2's, the component shared by both the MCD and ACD. Other than this theoretical observation, not much is known about the consequences of using ACD in modeling covariance and correlation matrices other than Chen and Dunson (2003), and Cai et al. (2006) in the context of random-effects selection. This factorization is more closely related to the MA representation of a "standardized" version of repeated measures on a subject, see (4.2), Pourahmadi (2007) and Rothman et al. (2010).

In this paper, our primary objective is to study some of the consequences of modeling the components of the ACD factorization on estimating the correlation matrix of longitudinal data. The secondary objective is to have procedures for estimating correlation matrices that are robust to outliers. We use the multivariate t distributions with ν the degrees of freedom unknown, as a model for the data and focus on accurate estimation of the df.

We point out some other structural, computational and statistical differences that exist between the MCD in Pourahmadi (2000) and the ACD in Chen and Dunson (2003). For example, recognizing that the MCD and ACD of a covariance matrix correspond to AR and MA representations of the underlying nonstationary longitudinal data (Pourahmadi, 2001, Sec. 3.5; Pourahmadi, 2007; Rothman et al., 2010), therefore one expects more computational difficulties in computing the MLE of the parameters of the ACD than those from MCD (Brockwell and Davis, 1991, Chaps 5 and 9). It is common to think of the MCD framework as being related to modeling the precision matrix, though recently, Rothman et al. (2010) have proposed sparse estimation of $\mathbf{\Sigma}$ itself based on its MCD and a related regression/MA interpretation of the entries of the factors. They show that there are significant structural and computational differences when working with $\mathbf{\Sigma}$, $\mathbf{\Sigma}^{-1}$ and their respective correlation matrices. A somewhat surprising result is that banding the Cholesky factor of the precision matrix coincides with constrained maximum likelihood, but banding the Cholesky factor of the covariance matrix itself does not. Such results are based on some interesting relationships between zero patterns of covariance matrices and their Cholesky factors. For example, the Cholesky factor of either the covariance matrix or its inverse is k-banded if and only if the corresponding matrix itself is k-banded, see Propositions 1-3 in Rothman et al. (2010).

The outline of the paper is as follows: In subsection 4.2, MCD and ACD are reviewed along with the statistical interpretations of the entries of their Cholesky decompositions. Subsection 4.3 discusses the multivariate t-distribution and the MLE of its parameters with a particular focus on the orthogonality of the parameters estimate. Subsection 4.4 illustrates the methodology using a real dataset, and assess its performance using a simulation experiment. Some conclusions are given in Chapter V.

4.2 MCD and ACD of a Covariance Matrix

In this Section, we review properties of two distinct Cholesky decompositions of the positive-definite covariance matrix of a longitudinal dataset, and discuss their roles in estimating the correlation matrix.

It is known that any $p \times p$ positive-definite covariance matrix can be factorized as $\boldsymbol{\Sigma} = \boldsymbol{C}\boldsymbol{C}^\top$, referred to as its standard Cholesky decomposition, where $\boldsymbol{C}$ is a unique lower triangular matrix with positive diagonal entries. What are the statistical relevance of the diagonal and sub-diagonals entries of $\boldsymbol{C}$? Letting $\boldsymbol{D} = \mathrm{diag}(c_{11}, \ldots, c_{pp})$, this factorization can take the following two distinct forms depending on whether the matrix $\boldsymbol{D}$ is inserted between the two lower triangular matrices or outside.

The MCD for $\boldsymbol{\Sigma}$ keeps $\boldsymbol{D}^2$ inside:

$$\boldsymbol{\Sigma} = \boldsymbol{C}\boldsymbol{D}^{-1}\boldsymbol{D}\boldsymbol{D}\boldsymbol{D}^{-1}\boldsymbol{C}^\top = \boldsymbol{L}\boldsymbol{D}^2\boldsymbol{L}^\top,$$

where $\boldsymbol{L} = \boldsymbol{C}\boldsymbol{D}^{-1}$ is a "standardized" version of $\boldsymbol{C}$, dividing each column by its diagonal entry. Defining $\boldsymbol{T} = \boldsymbol{L}^{-1}$, it is known (Pourahmadi, 1999) that the entries of $\boldsymbol{T}$ and $\boldsymbol{D}^2$, respectively, are negative of the GARPs in (4.1) and the prediction error variances σ_t^2's, when a measurement is regressed on its predecessors. Details of formulating parsimonious models using graphical tools like regressograms and estimating the ensuing parameters of $\boldsymbol{T}$ and $\boldsymbol{D}$ are given in Pourahmadi (1999).

The ACD in Chen and Dunson (2003) keeps $\boldsymbol{D}$ outside:

$$\boldsymbol{\Sigma} = \boldsymbol{D}\boldsymbol{D}^{-1}\boldsymbol{C}\boldsymbol{C}^\top\boldsymbol{D}^{-1}\boldsymbol{D} = \boldsymbol{D}\boldsymbol{L}\boldsymbol{L}^\top\boldsymbol{D},$$

where now $\boldsymbol{L} = \boldsymbol{D}^{-1}\boldsymbol{C}$ is obtained from $\boldsymbol{C}$ using a slightly different "standardization", namely dividing each row of $\boldsymbol{C}$ by its diagonal entries. In Pourahmadi (2001, 2007), the statistical interpretation of entries of $\boldsymbol{L}$ is given as the moving average coefficients when a standardized measurement is regressed on its past and present innovations, see also Rothman et al. (2010). Let $(y_1, \ldots, y_p)^\top$ be a zero mean random vector with covariance matrix $\boldsymbol{\Sigma}$. Denote $\boldsymbol{L}_{p\times p} = (\theta_{tj})$ and $\boldsymbol{D}_{p\times p} = \mathrm{diag}(\sigma_t)$. It's clear that $\boldsymbol{D}^{-1}\boldsymbol{y}$ has the covariance $\boldsymbol{L}\boldsymbol{L}^\top$. More precisely, defining $\boldsymbol{\epsilon} = (\boldsymbol{D}\boldsymbol{L})^{-1}\boldsymbol{y}$, it follows that $\mathrm{cov}(\boldsymbol{\epsilon}) = \boldsymbol{I}_p$ and then $\boldsymbol{D}^{-1}\boldsymbol{y} = \boldsymbol{L}\boldsymbol{\epsilon}$, from which we obtain a variable-order, varying-coefficients moving average representation for the standardized y_t/σ_t as

$$y_t/\sigma_t = \epsilon_t + \sum_{j=1}^{t-1} \theta_{tj}\epsilon_j. \tag{4.2}$$

From (4.2), for any $1 \leq s, t \leq p$, it follows that

$$\mathrm{cov}(y_s, y_t) = \sigma_s\sigma_t \sum_{j=1}^{s\wedge t} \theta_{tj}\theta_{sj},$$

so that the correlation between y_s and y_t given by

$$\mathrm{corr}(y_s, y_t) = \frac{\sum_{j=1}^{s\wedge t}\theta_{sj}\theta_{tj}}{\sqrt{\left(\sum_{j=1}^{s}\theta_{sj}^2 \sum_{j=1}^{t}\theta_{tj}^2\right)}}, \tag{4.3}$$

is solely determined by the $\boldsymbol{L}$ matrix. This property is a great motivation for modeling a correlation matrix using ACD, so that it is robust to model misspecifications for the innovation variances, $\sigma_t^2,\ t = 1, \ldots, p$.

4.3 MLEs for the ACD Model: The Multivariate t_ν

The assumption of multivariate normality commonly made for the vector of repeated measures on a subject may not be tenable in many practical situations when outliers exist or the underlying data exhibit heavy-tails. In this situation, a number of authors have used the multivariate t-distribution for robust estimation of the parameters of general linear models (Zellner, 1976; Lange et al., 1989); Lin and Wang (2009) has used it for robust estimation under the MCD decomposition. Robust estimation for linear mixed models using the multivariate t-distribution has been studied by Welsh and Richardson (1997) and Pinheiro et al. (2001).

In the sequel, for $i = 1, \ldots, n$, we assume that the vector of repeated measures on the i-th subject $\boldsymbol{y}_i \sim t(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}, \nu)$. This means that the p-dimensional vector $\boldsymbol{y}_i$ is following a multivariate t-distribution with degrees of freedom (df) ν, location vector $\boldsymbol{\mu}_i$ and scale matrix $\boldsymbol{\Sigma}$ with the probability density function given as

$$
\begin{aligned}
f(\boldsymbol{y}_i|\boldsymbol{\mu}_i \quad , \quad \boldsymbol{\Sigma}, \nu) &= \frac{\Gamma\Big(\dfrac{\nu+p}{2}\Big)}{\Gamma\Big(\dfrac{\nu}{2}\Big)(\pi\nu)^{p/2}}|\boldsymbol{\Sigma}|^{-1/2} \\
&\times \left(1 + \frac{(\boldsymbol{y}_i - \boldsymbol{\mu}_i)^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{y}_i - \boldsymbol{\mu}_i)}{\nu}\right)^{-(\nu+p)/2},
\end{aligned}
$$

where ν is a positive real number. For $\nu > 1$ the mean vector is defined to be $\boldsymbol{\mu}_i$, the covariance matrix exists for $\nu > 2$ and is equal to $\dfrac{\nu}{\nu-2}\boldsymbol{\Sigma}$.

Following the general approach in Pourahmadi (2000); Lin and Wang (2009) we model $\boldsymbol{\mu}_i, \boldsymbol{L} = (\theta_{tj})$ and $\boldsymbol{D} = \text{diag}(\sigma_t)$ as

$$
\boldsymbol{\mu}_i = \boldsymbol{X}_i\beta, \quad \theta_{tj} = d(\boldsymbol{z}_{tj}, \boldsymbol{\gamma}), \quad \text{log}\sigma_t = v(\boldsymbol{z}_t, \boldsymbol{\lambda}), \tag{4.4}
$$

where $d(\cdot,\cdot)$, $v(\cdot,\cdot)$ are known functions, $\boldsymbol{X}_i$, $\boldsymbol{z}_{tj}$ and $\boldsymbol{z}_t$ are $p \times m$, $d \times 1$ and $q \times 1$ matrices of covariates, $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_m)^\top, \boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_d)^\top$ and $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_q)^\top$ are parameters of the mean, log-innovation and the moving average parameters $\boldsymbol{y}$ in the ACD, respectively. When $d(\cdot,\cdot)$, $v(\cdot,\cdot)$ are polynomials, we use the notation Poly(d, q) as a shorthand for two distinct polynomials of degrees d, q in the lagged times $(t - j)$ and t for θ_{tj} and logσ_t, respectively. Specifically, in this case the covariates $\boldsymbol{z}_t$ and $\boldsymbol{z}_{tj}$

are of the form:

$$\boldsymbol{z}_{tj} = (1, (t-j), \dots, (t-j)^d)^\top, \qquad j = 1, \dots, t-1,$$
$$\boldsymbol{z}_t = (1, t, \dots, t^q)^\top, \qquad t = 1, \dots, p.$$

For example, in most of our simulation work we use Poly$(3, 3)$ as models for the components of $\boldsymbol{L}, \boldsymbol{D}$.

Assuming $m, q,$ and d are known, let $\boldsymbol{\theta} = (\boldsymbol{\beta}^\top, \ \boldsymbol{\gamma}^\top, \boldsymbol{\lambda}^\top, \nu)^\top$ be the partitioned vector of all parameters in the model, then the log-likelihood function $\ell(\boldsymbol{\theta})$ is

$$\begin{aligned} \ell(\boldsymbol{\theta}) &= \Bigg(\log\Gamma\Big(\frac{\nu+p}{2}\Big) - \log\Gamma\Big(\frac{\nu}{2}\Big) - \frac{p}{2}\log(\pi\nu)\Bigg) \\ &- \frac{n}{2}\log|\boldsymbol{D}^2| - \frac{1}{2}(\nu+p)\sum_{i=1}^{n}\log\Big(1 + \frac{\Delta_i(\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\lambda})}{\nu}\Big), \end{aligned}$$

where $\Delta_i(\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\lambda}) := (\boldsymbol{y}_i - \boldsymbol{X}_i\boldsymbol{\beta})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{y}_i - \boldsymbol{X}_i\boldsymbol{\beta})$. We suppress its arguments and use the abbreviation Δ_i in the sequel.

4.3.1 Maximum Likelihood Estimation Using Fisher Scoring

In this Section, we study some computational and statistical implications of using covariates in the parsimonious modeling of $\boldsymbol{L}$ in (4.4) as compared to the same approach in modeling $\boldsymbol{T}$ in the MCD approach studied in Pourahmadi (2000); Lin and Wang (2009). It turns out that there is no closed-form solution for the MLEs of ACD models under the multivariate t_ν setup, thus iterative algorithms like the Newton-Raphson or Fisher scoring as in Pourahmadi (2000) and Lin and Wang (2009) is developed here.

The Fisher scoring algorithm is developed in this subsection. For the partitioning of $\boldsymbol{\theta}$ as above, the blocks of the score function $U(\boldsymbol{\theta}) = \big(U^\top(\boldsymbol{\beta}), U^\top(\boldsymbol{\gamma}), U^\top(\boldsymbol{\lambda}), U(\nu)\big)^\top$

can be obtained and simplified as

$$\begin{aligned}
U(\boldsymbol{\beta}) &= \sum_{i=1}^{n} \omega_i \boldsymbol{X}_i^{\top} \boldsymbol{\Sigma}^{-1} \boldsymbol{r}_i, \\
U(\gamma_r) &= \mathrm{tr}\Big((\boldsymbol{T}\boldsymbol{D}^{-1})\Big(\sum_{i=1}^{n} \omega_i \boldsymbol{S}_i\Big)(\boldsymbol{T}\boldsymbol{D}^{-1})^{\top}\boldsymbol{T}\boldsymbol{L}_{\gamma_r}\Big), \\
U(\lambda_s) &= \mathrm{tr}\Big(\Big(\Big(\sum_{i=1}^{n} \omega_i \boldsymbol{S}_i\Big)\boldsymbol{\Sigma}^{-1} - n\boldsymbol{I}\Big)\boldsymbol{D}^{-1}\boldsymbol{D}_{\lambda_s}\Big),
\end{aligned}$$

$$\begin{aligned}
U(\nu) = \frac{1}{2}\sum_{i=1}^{n}\Big(\phi\Big(\frac{\nu+p}{2}\Big) &- \phi\Big(\frac{\nu}{2}\Big) - \frac{p}{\nu} \\
&- \log\Big(1+\frac{\Delta_i}{\nu}\Big) + \frac{\omega_i}{\nu}\Delta_i\Big),
\end{aligned}$$

where $r = 1, \ldots, d$, $s = 1, \ldots, q$, $\boldsymbol{L}_{\gamma_r} = \dfrac{\partial}{\partial \gamma_r}\boldsymbol{L}$, $\boldsymbol{D}_{\lambda_s} = \dfrac{\partial}{\partial \lambda_s}\boldsymbol{D}$, $\omega_i = \dfrac{\nu+p}{\nu+\Delta_i}$, $\boldsymbol{r}_i = (\boldsymbol{y}_i - \boldsymbol{X}_i\boldsymbol{\beta})$, $\boldsymbol{S}_i = \boldsymbol{r}_i\boldsymbol{r}_i^{\top}$ and $\phi(x) = \frac{d}{dx}\log\Gamma(x)$.

Now, we have the necessary ingredients to present the Fisher information in terms of the blocks of a partitioned 4×4 matrix corresponding to $\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\lambda}$, and ν. The blocks of the Fisher information that involve $\boldsymbol{\beta}$ (the location parameter) are as follows:

$$\begin{aligned}
\boldsymbol{I}_{11}(\boldsymbol{\beta}) = -\mathbb{E}(\ell_{\boldsymbol{\beta}\boldsymbol{\beta}}) &= \frac{\nu+p}{\nu+p+2}\sum_{i=1}^{n}\boldsymbol{X}_i^{\top}\boldsymbol{\Sigma}^{-1}\boldsymbol{X}_i, \\
\boldsymbol{I}_{12}(\boldsymbol{\beta}, \boldsymbol{\gamma}) &= -\mathbb{E}(\ell_{\boldsymbol{\beta}\boldsymbol{\gamma}}) = \boldsymbol{0}, \\
\boldsymbol{I}_{13}(\boldsymbol{\beta}, \boldsymbol{\lambda}) &= -\mathbb{E}(\ell_{\boldsymbol{\beta}\boldsymbol{\lambda}}) = \boldsymbol{0}, \\
\boldsymbol{I}_{14}(\boldsymbol{\beta}, \nu) &= -\mathbb{E}(\ell_{\boldsymbol{\beta}\nu}) = \boldsymbol{0}.
\end{aligned}$$

In addition, we obtain other blocks of the Fisher information matrix using Proposition 4 of Lange et al. (1989). We state two versions of the result corresponding to the parameterizations based on MCD and ACD.

Let $\boldsymbol{\varphi}$ denote a generic parametrization of either $\boldsymbol{\Sigma}$ or $\boldsymbol{\Sigma}^{-1}$ for the p-variate t_ν distribution with the scale matrix $\boldsymbol{\Sigma}$, the contribution of a single observation to the

Fisher information block for the scale parameter and the degrees of freedom are as follows:

$$\begin{aligned}
I_{i,j}(\boldsymbol{\varphi}) &= \frac{1}{2(\nu+p+2)}\Big[(\nu+p)\mathrm{tr}\Big(\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}_{\varphi_i}\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}_{\varphi_j}\Big) \\
&\quad -\mathrm{tr}\Big(\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}_{\varphi_i}\Big)\mathrm{tr}\Big(\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}_{\varphi_j}\Big)\Big] \\
&= \frac{1}{2(\nu+p+2)}\Big[(\nu+p)\mathrm{tr}\Big(\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{-1}{}_{\varphi_i}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{-1}{}_{\varphi_j}\Big) \\
&\quad -\mathrm{tr}\Big(\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{-1}{}_{\varphi_i}\Big)\mathrm{tr}\Big(\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{-1}{}_{\varphi_j}\Big)\Big], \\
I_i(\boldsymbol{\varphi},\nu) &= -\frac{1}{(\nu+p+2)(\nu+p)}\mathrm{tr}\Big(\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}_{\varphi_i}\Big) \\
&= -\frac{1}{(\nu+p+2)(\nu+p)}\mathrm{tr}\Big(\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{-1}{}_{\varphi_i}\Big).
\end{aligned}$$

The equations involving $\boldsymbol{\Sigma}_{\boldsymbol{\varphi}}$ $\left(\text{i.e., } \dfrac{\partial\boldsymbol{\Sigma}}{\partial\boldsymbol{\varphi}}\right)$ are useful for the ACD model, while those involving $\boldsymbol{\Sigma}^{-1}{}_{\boldsymbol{\varphi}}$ $\left(\text{i.e., } \dfrac{\partial\boldsymbol{\Sigma}^{-1}}{\partial\boldsymbol{\varphi}}\right)$ can be used for modeling $\boldsymbol{\Sigma}^{-1}$. In the application to model (4.4), $\boldsymbol{\varphi}^\top = (\boldsymbol{\gamma}^\top, \boldsymbol{\lambda}^\top)^\top$ is for parameterizing the scale matrix.

Once the information matrix is computed, the iterative Fisher scoring algorithm can be used to compute the MLE of the parameters by updating the current value of $\tilde{\boldsymbol{\theta}}$ to $\hat{\boldsymbol{\theta}}$:

$$\hat{\boldsymbol{\theta}} = \tilde{\boldsymbol{\theta}} + I^{-1}(\tilde{\boldsymbol{\theta}})U(\tilde{\boldsymbol{\theta}}).$$

Note that when using linear link functions for $d(\cdot,\cdot)$, and $v(\cdot,\cdot)$ in (4.4), simpler structures for the score function and the Fisher information will result. Also, when $\nu \to \infty$, the results in this Section reduce to those for an iterative procedure for computing the MLEs of the ACD model parameters under the multivariate normal setup.

Computation and the form of the entries of the Fisher information matrix are slightly different for ACD and MCD and are summarized in the following two subsections.

4.3.2 Fisher Information Matrix for ACD

As an immediate consequence of the results given in subsection 4.3.1 we obtain the Fisher information blocks for the parameters of the components of the scale matrix and the degrees of freedom for the ACD model.

$$\begin{aligned}
I_{22,rs}(\boldsymbol{\gamma}) &= \frac{(\nu+p)n}{\nu+p+2}\mathrm{tr}(\boldsymbol{L}_{\gamma_r}\boldsymbol{L}_{\gamma_s}^{\top}\boldsymbol{T}^{\top}\boldsymbol{T}),\\
I_{33,rs}(\boldsymbol{\lambda}) &= \Bigg[\mathrm{tr}\big(\boldsymbol{L}\boldsymbol{L}^{\top}\boldsymbol{D}_{\lambda_r}\boldsymbol{\Sigma}^{-1}\boldsymbol{D}_{\lambda_s}+\boldsymbol{D}^{-2}\boldsymbol{D}_{\lambda_r}\boldsymbol{D}_{\lambda_s}\big)\\
&\quad -\frac{2\mathrm{tr}(\boldsymbol{D}^{-1}\boldsymbol{D}_{\lambda_r})\mathrm{tr}(\boldsymbol{D}^{-1}\boldsymbol{D}_{\lambda_s})}{(\nu+p)}\Bigg]\frac{n(\nu+p)}{\nu+p+2},\\
I_{44}(\nu) &= \frac{n}{4}\Bigg[\psi\Big(\frac{\nu}{2}\Big)-\psi\Big(\frac{\nu+p}{2}\Big)\\
&\quad -\frac{2p(\nu+p+4)}{\nu(\nu+p)(\nu+p+2)}\Bigg],\\
I_{23,rs}(\boldsymbol{\gamma},\boldsymbol{\lambda}) &= \frac{(\nu+p)n}{\nu+p+2}\mathrm{tr}(\boldsymbol{D}\boldsymbol{L}_{\gamma_r}\boldsymbol{L}^{\top}\boldsymbol{D}_{\lambda_s}\boldsymbol{\Sigma}^{-1}),\\
I_{24,r}(\boldsymbol{\gamma},\nu) &= 0,\\
I_{34,s}(\boldsymbol{\lambda},\nu) &= -\frac{2n}{(\nu+p+2)(\nu+p)}\mathrm{tr}(\boldsymbol{D}^{-1}\boldsymbol{D}_{\lambda_s}),
\end{aligned}$$

where $\psi(x)=\frac{d^2}{dx^2}\mathrm{log}\Gamma(x)$ stands for the trigamma function.

Letting $\nu\to\infty$ in the above identities, we obtain the corresponding results for the multivariate normal model where the log-likelihood function $\ell(\theta)$, up to an additive constant is

$$-\frac{2}{n}\ell(\theta)=\mathrm{log}|\boldsymbol{D}^2|+n^{-1}\sum_{i=1}^{n}\Delta_i=\sum_{t=1}^{p}\mathrm{log}\sigma_t^2+\mathrm{tr}\boldsymbol{S}\boldsymbol{\Sigma}^{-1}.$$

The score function and the Fisher information for the multivariate normal distribution is easy to obtain by considering the following facts and substituting in the previous results:

$$\omega_i\to 1,\quad \frac{(\nu+p)n}{\nu+p+2}\to n,\quad \frac{2n}{\nu+p+2}\to 0,$$

and $\sum_{i=1}^{n}\omega_i\boldsymbol{S}_i = n\boldsymbol{S}$, where $\boldsymbol{S} = n^{-1}\sum_{i=1}^{n}\boldsymbol{r}_i\boldsymbol{r}_i^\top$.

4.3.3 Comparison with the Fisher Information Matrix for MCD

In this Section, we find the Fisher information matrix for the MCD and compare it with that for the ACD models. For simplicity, we use the same notation for the information matrices corresponding to ACD and MCD. Using the result given in subsection 4.3.1, the entries of the Fisher information associated to the scale parameter and the degrees of freedom for the MCD model are

$$\begin{aligned}
I_{22,rs}(\boldsymbol{\gamma}) &= \frac{(\nu+p)n}{\nu+p+2}\mathrm{tr}(\boldsymbol{T}_{\gamma_r}^\top\boldsymbol{D}^{-2}\boldsymbol{T}_{\gamma_s}\boldsymbol{\Sigma}),\\
I_{33,rs}(\boldsymbol{\lambda}) &= \frac{n}{2(\nu+p+2)}\Bigg[(\nu+p)\mathrm{tr}(\boldsymbol{D}^{-4}\boldsymbol{D}^2{}_{\lambda_r}\boldsymbol{D}^2{}_{\lambda_s})\\
&\qquad -\mathrm{tr}(\boldsymbol{D}^{-2}\boldsymbol{D}^2{}_{\lambda_r})\mathrm{tr}(\boldsymbol{D}^{-2}\boldsymbol{D}^2{}_{\lambda_s})\Bigg],\\
I_{23,rs}(\boldsymbol{\gamma},\boldsymbol{\lambda}) &= 0,\\
I_{24_r}(\gamma,\nu) &= 0,\\
I_{34,s}(\boldsymbol{\lambda},\nu) &= -\frac{n}{(\nu+p+2)(\nu+p)}\mathrm{tr}(\boldsymbol{D}^{-2}\{\boldsymbol{D}^2\}_{\lambda_s}).
\end{aligned}$$

Comparing similar entries in the two Sections, it is evident that their forms and values are quite different for the ACD and MCD models even for general link functions $d(\cdot,\cdot)$, $v(\cdot,\cdot)$. However, some notable and computationally useful differences are singled out below:

1. The parameters $\boldsymbol{\gamma}$ and $\boldsymbol{\lambda}$ are asymptotically orthogonal in the MCD, but not in the ACD. It is known that for the multivariate normal distribution, the $\boldsymbol{\gamma}$ and $\boldsymbol{\lambda}$ are asymptotically orthogonal in the MCD model (Ye and Pan, 2006; Holan and Spinka, 2007), but not in the ACD model (Pourahmadi, 2007). Here we have shown the same to be true for the multivariate t_ν setup. Our finding is different from that in Lin and Wang (2009), p. 3016.

2. The parameters ν and $\boldsymbol{\gamma}$ are asymptotically orthogonal in both the ACD and MCD models, this is not the case for ν and $\boldsymbol{\lambda}$, the parameters of the innovation variance.

3. Since $\boldsymbol{D} = \text{diag}(\sigma_t)$ is a diagonal matrix, letting $\log(\sigma_t) := \boldsymbol{z}_t^\top \boldsymbol{\lambda}$, the derivative of $\boldsymbol{D}$ with respect to λ_s is $\boldsymbol{D}_{\lambda_s} = (\boldsymbol{Z}_{D,s})\boldsymbol{D}$, where

$$\boldsymbol{Z}_{D,s} = \text{diag}(z_{1,s}, \dots, z_{p,s}), \qquad s = 1, \dots, q.$$

   Thus, replacing the matrix $\boldsymbol{D}^{-1}\boldsymbol{D}_{\lambda_s}$ by $\boldsymbol{Z}_{D,s}$ using the above results lead to simpler forms for parts of the score function and the Fisher information that involve $\boldsymbol{\lambda}$. Also, using the log-linear models for the innovation standard deviation, both MCD and ACD models have the same quantity for $I_{34}(\boldsymbol{\lambda}, \nu)$.

4.4 Data Analysis

In this Section, we compare the robustness and capabilities of the ACD and MCD for modeling various correlation structures using simulated and real data. We denote the MCD and ACD when used in conjunction with the multivariate normal and t distributions as MCDN and ACDN, MCDT and ACDT, respectively.

We compare estimators of correlation matrices using the following two loss functions and their corresponding risks:

$$\Delta_1(\boldsymbol{R}, \boldsymbol{G}) = \text{tr}\boldsymbol{R}^{-1}\boldsymbol{G} - \log|\boldsymbol{R}^{-1}\boldsymbol{G}| - n,$$

$$\text{and} \quad \Delta_2(\boldsymbol{R}, \boldsymbol{G}) = \text{tr}(\boldsymbol{R}^{-1}\boldsymbol{G} - \boldsymbol{I})^2,$$

where $\boldsymbol{R}$ is the target correlation matrix and $\boldsymbol{G}$ is another positive-definite correlation matrix of the same size. The loss $\Delta_1(\boldsymbol{R}, \boldsymbol{G})$ is known as the entropy loss and $\Delta_2(\boldsymbol{R}, \boldsymbol{G})$ as the quadratic loss. Both of these loss functions are 0, when $\boldsymbol{G} = \boldsymbol{R}$ and positive, when $\boldsymbol{G} \neq \boldsymbol{R}$. Their corresponding risk functions are

$$R_i(\boldsymbol{R}, \boldsymbol{G}) = E_R\{\Delta_i(\boldsymbol{R}, \boldsymbol{G})\}, \qquad i = 1, 2.$$

An estimator $\hat{\boldsymbol{R}}$ is better than $\tilde{\boldsymbol{R}}$, if its associated risk is smaller, that is, $R_i(\boldsymbol{R}, \hat{\boldsymbol{R}}) < R_i(\boldsymbol{R}, \tilde{\boldsymbol{R}})$.

4.4.1 Simulation

We fix the true parameters (mean, covariance/correlation matrix) for the simulation setup using those of the well-known Kenward (1987)'s cattle data. Here the weight of thirty cattle were recorded 11 times over a 133-day period, the dataset has been analyzed by several authors Zimmerman and Núñez Antón (2009). As in Pourahmadi (1999), cubic polynomials were fitted to the Cholesky factors $\boldsymbol{T}, \boldsymbol{D}$ of the sample covariance matrix of the treatment A of the cattle data.

For simulating data, we construct two true 11×11 covariance matrices corresponding to those of the cattle data fitted with MCDN-Poly$(3, 3)$ and ACDN-Poly$(3, 3)$ denoted by $\boldsymbol{\Sigma}_{\text{mcd}}$ and $\boldsymbol{\Sigma}_{\text{acd}}$, respectively. Thus, the true covariance (correlation) matrices are known and correspond to the above fits.

We generated $m = 100$ datasets from a multivariate t_ν-distribution with the mean vector equal to the sample mean of the cattle data and the scale matrix equal to $\boldsymbol{\Sigma}_{\text{mcd}}$ and $\boldsymbol{\Sigma}_{\text{acd}}$, respectively, and for the following combinations of (ν, n): "$\nu = 4, 50$" (*df*), "$n = 25, 100$" (sample sizes). We calculated the entropy and quadratic risks after fitting MCDN, MCDT, ACDN and ACDT using the Fisher scoring algorithm described in subsection 4.3. Note that here we fit cubic polynomials both to the GARPs (GMAPs) and the log-innovation variances, the same models as their true counterparts. The results in Table 3(a) show that the risks in the third and forth columns are much smaller than those in the first two columns of both panels. This

Table 3: (a) Simulating data from $\mathbf{\Sigma}_{\text{mcd}}$ and fitting Poly$(3, 3)$ (cubic fit for innovation variance). Values within parentheses are empirical standard errors. (b) Simulating data from $\mathbf{\Sigma}_{\text{acd}}$ and fitting Poly$(3, 3)$ (cubic fit for innovation variance).

(a)

	Simulating from $\mathbf{\Sigma}_{\text{mcd}}$							
	$\nu = 4$				$\nu = 50$			
Risk type	ACDT	ACDN	MCDT	MCDN	ACDT	ACDN	MCDT	MCDN
n=25	0.8379	1.1563	0.4009	0.7870	0.9169	0.9334	0.5043	0.5144
Entropy	(0.4901)	(0.8780)	(0.3581)	(0.8900)	(0.4661)	(0.4563)	(0.3958)	(0.4043)
n=25	2.4038	3.5832	0.9126	1.9400	2.5983	2.6308	1.1184	1.1392
Quadratic	(2.0487)	(3.9302)	(1.2406)	(2.8187)	(1.7479)	(1.7201)	(1.1642)	(1.1911)
Entropy	0.6206	0.7224	0.1215	0.2653	0.6555	0.6544	0.1118	0.1124
n=100	(0.1591)	(0.2539)	(0.0920)	(0.2827)	(0.1857)	(0.1825)	(0.1189)	(0.1205)
Quadratic	1.7016	2.0171	0.2490	0.5677	1.8151	1.8118	0.2470	0.2480
n=100	(0.5412)	(0.8908)	(0.1976)	(0.6303)	(0.6137)	(0.6037)	(0.3166)	(0.3227)

(b)

	Simulating from $\mathbf{\Sigma}_{\text{acd}}$							
	$\nu = 4$				$\nu = 50$			
Risk type	ACDT	ACDN	MCDT	MCDN	ACDT	ACDN	MCDT	MCDN
n=25	0.3460	0.7072	0.5866	0.9780	0.3583	0.3641	0.6337	0.6516
Entropy	(0.3597)	(0.9609)	(0.3735)	(0.9712)	(0.2682)	(0.2813)	(0.3476)	(0.3820)
n=25	0.8250	1.8045	1.0928	2.1981	0.7807	0.7846	1.0805	1.0848
Quadratic	(1.2135)	(3.6170)	(1.1615)	(4.6778)	(0.7516)	(0.7764)	(0.5850)	(0.5876)
n=100	0.0917	0.2681	0.3283	0.5041	0.0826	0.0849	0.3116	0.3152
Entropy	(0.0747)	(0.4659)	(0.1573)	(0.4378)	(0.0807)	(0.0830)	(0.1597)	(0.1597)
n=100	0.1813	0.6898	0.5320	0.9643	0.1694	0.1750	0.5146	0.5204
Quadratic	(0.1495)	(2.1286)	(0.2159)	(1.7977)	(0.1819)	(0.1883)	(0.2425)	(0.2425)

indicates the improved performance of MCD over ACD, when the data are actually generated from the same MCD covariance (correlation) structure. Furthermore, in the left panel corresponding to $\nu = 4$, a smaller degrees of freedom, the risks for MCDT and ACDT are much smaller than MCDN and ACDN, and this difference disappears, as expected, for $\nu = 50$. Similar statements can be made about the results in Table 3(b) where the data are generated using the ACD covariance structure, but now one can see that the first two columns of the two panels are smaller than their counterparts in the last two columns. In summary, the simulation results reported in Table 3 show the importance of knowing the structure of the underlying covariance matrix, where the MCD works better for datasets coming from MCD structure, and the ACD fits the covariance matrix better if the data is coming from an ACD structure.

Next, the theoretical result in Chen and Dunson (2003) and subsection 4.2 suggest that the estimate of the correlation matrix is robust to model misspecification of the innovation variances when using the ACD. To verify this empirically, we rely on the same dataset used for the simulations in Table 3, but for log-innovation variances we fit a linear structure rather than the true cubic polynomial. The impact of this innovation variance misspecification on estimating the correlation matrix can be seen in Table 4. More precisely, we observe the followings:

1. Comparing the first two ACD columns of Table 3 with the first two columns of Tables 4 in both panels, shows that the correlation estimation is robust to the model misspecification for innovation variances. This conclusion seems to be independent of the structure of the covariance matrix used for the simulation ($\boldsymbol{\Sigma}_{\text{mcd}}$ or $\boldsymbol{\Sigma}_{\text{acd}}$).

2. The last two MCD columns of Table 3(a) (Simulation from $\boldsymbol{\Sigma}_{\text{mcd}}$) have smaller risks compare to the last two columns of Table 4(a) in both panels. This confirms

that the correlation estimation is not robust to the model misspecification for innovation variances in the MCD structure.

Finally, we undertook a simulation study to examine the performance and flexibility of the proposed ACDT approach. The main objective is to study the robustness or sensitivity to the true distribution. For example, it is important to know when data are from a t distribution, how bad the MCD or ACD will perform when we use the normal distribution to estimate the parameters, and vice versa? For the sake of diversity, now the true parameters are set to be those of the tumor data (discussed in subsection 4.4.2) analyzed next and fitted with ACDT-Poly$(3,3)$, except that the *df* is specified at two different settings. For the *df*'s, we take a low value ($\nu = 4$) corresponding to heavy-tailed distributions and a high value ($\nu = 50$) corresponding to near normality. The two sample sizes were from small ($n = 25$) to a relatively large ($n = 100$). Simulations were run with $m = 500$ replications for each combination of ν and n and each simulated data set was fitted under ACDT and ACDN scenarios. The detailed numerical results, including the average ML estimates for the fixed effects, the moving average parameters and the scale innovation variances, the average of maximized log-likelihood values $\ell_{\max}$, the average of associated BIC values and the median estimates for the *df*, together with their standard errors in parentheses, are summarized in Table 5. It shows that for smaller ν the point estimators of the parameters under the ACDT and ACDN scenarios are generally the same, but their SE's differ with the normal distributions leading to larger SE's. Furthermore, the estimated *df* has a downward bias for the smaller sample size $n = 25$.

4.4.2 The Tumor Growth Data

We apply our methodology to the *in vivo* growth of lung tumor for the control group of 22 xenografted nude mice, which has been also analyzed in Lin and Wang

Table 4: (a) Simulating data from $\mathbf{\Sigma}_{\text{mcd}}$ and fitting Poly$(3, 1)$ (linear fit for innovation variance). Values within parentheses are empirical standard errors. (b) Simulating data from $\mathbf{\Sigma}_{\text{acd}}$ and fitting Poly$(3, 1)$ (linear fit for innovation variance).

(a)

	Simulating from $\mathbf{\Sigma}_{\text{mcd}}$							
	$\nu = 4$				$\nu = 50$			
Risk type	ACDT	ACDN	MCDT	MCDN	ACDT	ACDN	MCDT	MCDN
n=25	0.8651	1.1367	0.7890	1.0247	0.9580	0.9846	0.8655	0.8867
Entropy	(0.4200)	(0.7285)	(0.3993)	(0.7108)	(0.4393)	(0.4437)	(0.3905)	(0.3985)
n=25	2.4347	3.3864	2.1492	2.9354	2.6603	2.7305	2.3016	2.3523
Quadratic	(1.7953)	(3.1821)	(1.5791)	(2.9867)	(1.6925)	(1.7229)	(1.4214)	(1.4603)
n=100	0.6523	0.7618	0.5892	0.6865	0.6739	0.6742	0.6072	0.6071
Entropy	(0.1467)	(0.2444)	(0.1431)	(0.2362)	(0.1449)	(0.1429)	(0.1470)	(0.1473)
n=100	1.7762	2.0849	1.5803	1.8342	1.8601	1.8593	1.6486	1.6466
Quadratic	(0.5336)	(0.8488)	(0.4979)	(0.7583)	(0.5209)	(0.5124)	(0.5060)	(0.5064)

(b)

	Simulating from $\mathbf{\Sigma}_{\text{acd}}$							
	$\nu = 4$				$\nu = 50$			
Risk type	ACDT	ACDN	MCDT	MCDN	ACDT	ACDN	MCDT	MCDN
n=25	0.3942	0.7456	0.4181	0.7617	0.4552	0.4665	0.4592	0.4712
Entropy	(0.2791)	(0.8003)	(0.2773)	(0.8229)	(0.3258)	(0.3577)	(0.3159)	(0.3406)
n=25	0.8791	1.8830	0.8315	1.8184	0.7768	0.7818	0.7841	0.7868
Quadratic	(0.4917)	(2.4344)	(0.5498)	(2.6867)	(0.5345)	(0.5773)	(0.5383)	(0.5585)
n=100	0.1289	0.3237	0.2158	0.4358	0.1276	0.1323	0.1995	0.2042
Entropy	(0.1548)	(0.8596)	(0.1412)	(0.8468)	(0.1383)	(0.1390)	(0.1234)	(0.1242)
n=100	0.2295	0.7257	0.3456	0.7676	0.2030	0.2102	0.3294	0.3364
Quadratic	(0.2135)	(1.7008)	(0.1913)	(1.7750)	(0.1978)	(0.1982)	(0.1780)	(0.1778)

Table 5: Average estimates for $\boldsymbol{\gamma}, \boldsymbol{\lambda}, \ell_{\max}$ and BIC and the median estimate for ν based on 500 replications. Values within parentheses are empirical standard errors.

Param.	True Param.	n=25				n=100			
		$\nu = 4$		$\nu = 50$		$\nu = 4$		$\nu = 50$	
		ACDT	ACDN	ACDT	ACDN	ACDT	ACDN	ACDT	ACDN
γ_0	0.9318	0.9293	0.9261	0.9381	0.9373	0.9326	0.9288	0.9268	0.9268
		(0.0085)	(0.0115)	(0.0081)	(0.0081)	(0.0040)	(0.0060)	(0.0037)	(0.0038)
γ_1	0.0962	0.1230	0.1442	0.1533	0.1538	0.1106	0.1220	0.0979	0.0988
		(0.0200)	(0.0275)	(0.0183)	(0.0183)	(0.0091)	(0.0141)	(0.0086)	(0.0086)
γ_2	0.0898	0.1012	0.1119	0.1116	0.1101	0.0985	0.1020	0.0935	0.0942
		(0.0100)	(0.0127)	(0.0095)	(0.0095)	(0.0047)	(0.0072)	(0.0045)	(0.0045)
γ_3	0.3041	0.3087	0.3095	0.3076	0.3063	0.3076	0.3053	0.3011	0.3015
		(0.0054)	(0.0074)	(0.0052)	(0.0052)	(0.0027)	(0.0043)	(0.0025)	(0.0025)
λ_0	-1.6379	-1.6706	-1.7175	-1.7031	-1.6693	-1.6919	-1.6668	-1.6690	-1.6468
		(0.0044)	(0.0064)	(0.0022)	(0.0021)	(0.0020)	(0.0037)	(0.0011)	(0.0010)
λ_1	-0.5685	-0.5801	-0.5989	-0.5844	-0.5821	-0.5760	-0.5799	-0.5721	-0.5725
		(0.0071)	(0.0099)	(0.0068)	(0.0069)	(0.0033)	(0.0050)	(0.0033)	(0.0033)
λ_2	0.5416	0.5241	0.5121	0.5280	0.5264	0.5396	0.5307	0.5375	0.5376
		(0.0058)	(0.0075)	(0.0055)	(0.0055)	(0.0026)	(0.0045)	(0.0026)	(0.0027)
λ_3	-0.2795	-0.2766	-0.2745	-0.2812	-0.2822	-0.2800	-0.2776	-0.2774	-0.2774
		(0.0044)	(0.0061)	(0.0040)	(0.0040)	(0.0021)	(0.0031)	(0.0021)	(0.0021)
ν		4.1040	.	34.5796	.	4.0612	.	49.7055	.
		(0.2006)	.	(2.8122)	.	(0.0329)	.	(2.7051)	.
$\ell_{\max}$		121.67	89.560	76.112	75.110	451.18	297.40	275.4831	273.40
		(1.3179)	(1.9102)	(0.6318)	(0.6385)	(2.4676)	(4.4406)	(1.2285)	(1.2430)
BIC		-7.0301	-4.5897	-3.3851	-3.4337	-8.0566	-5.0270	-4.5426	-4.5470
		(0.1054)	(0.1528)	(0.0505)	(0.0511)	(0.0494)	(0.0888)	(0.0246)	(0.0249)

(2009) using MCDT. Figure 15 shows the profile plot of the logarithm of tumor growth volumes over an unequally spaced 28-day period for the 22 mice, together with the sample regressograms of the generalized moving average parameters (GMAPs), and the sample innovation standard deviations. It should be noted that our analysis is based on the saturated model for the mean function. In fact, following the analysis of Lin and Wang (2009) and using the design matrix for the mean response to be $\boldsymbol{X}_i = [\mathbf{1}\ \boldsymbol{k}]$, where $\mathbf{1} = (1, 1, \ldots, 1)^\top$, $\boldsymbol{k} = (0, 1, 2.5, 3.5, 4.5, 6, 7, 8, 10, 11.5, 13, 14)^\top$, the optimization procedure using the Newton-Raphson algorithm for the ACDT model will converge only to a local maximum which depends noticeably on the choice of the initial values. However, using the saturated mean model the algorithm converges to the global maximum for both ACDT and ACDN. We fit the tumor data using ACDN and ACDT for various choices of the degrees of the Poly(d, q) models. The values of $\ell_{\max}$, together with the corresponding number of parameters and BIC values for selected pairs(d, q) are listed in Table 6. Judging from the BIC values, Poly$(6, 5)$ is the best and also Poly$(3, 5)$ is relatively parsimonious and a competitive choice for both ACDN and ACDT models. Table 7 shows the ML estimates and the associated standard errors for the best two fitting ACDN and ACDT. It is noteworthy that the estimates of the *df* for the two fitted ACDT are somewhat small, suggesting that the error distribution has a larger tail than the normal distribution, which confirms the finding of Lin and Wang (2009). Finally note that, based on the different interpretation of ACD and MCD parameters, the GMAPs and GARPs are not comparable.

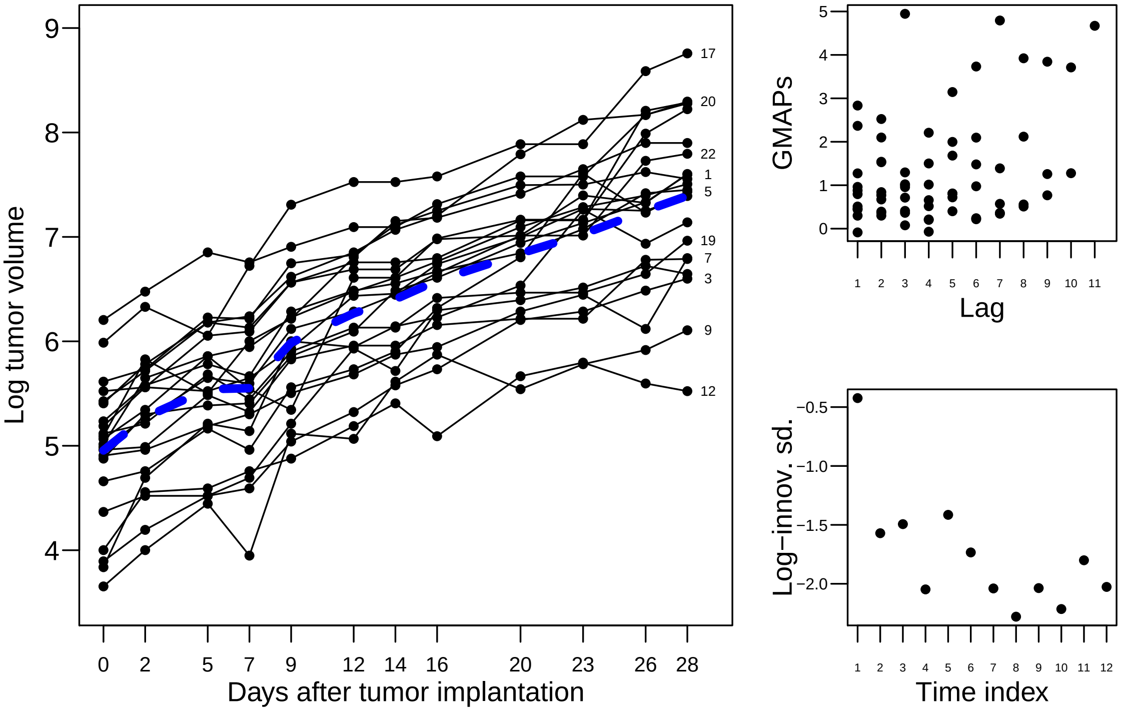

Figure 15: Profile plot of the tumor data, and the plots of GMAPs and log-innovation standard deviation.

Table 6: Comparison of $\ell_{\max}$, number of parameters, and BIC values for some Poly(d, q) choices of ACDN and ACDT models.

Poly(d, q)	# of param.		$\ell_{\max}$		BIC	
	ACDN	ACDT	ACDN	ACDT	ACDN	ACDT
(1,1)	4	5	-26.94	-9.575	4.697	3.259
(1,2)	5	6	-14.64	-0.134	3.719	2.541
(1,3)	6	7	-14.21	1.661	3.821	2.519
(1,4)	7	8	-13.70	3.041	3.915	2.534
(1,5)	8	9	-10.17	5.541	3.734	2.447
(1,6)	9	10	-10.15	5.596	3.873	2.582
(2,1)	5	6	-24.36	-8.929	4.603	3.341
(2,2)	6	7	-14.17	1.244	3.817	2.556
(2,3)	7	8	-14.01	2.168	3.943	2.613
(2,4)	8	9	-13.48	3.577	4.036	2.625
(2,5)	9	10	-10.09	5.982	3.868	2.547
(2,6)	10	11	-10.07	6.021	4.007	2.684
(3,1)	6	7	-22.01	-7.462	4.530	3.348
(3,2)	7	8	-13.10	2.239	3.860	2.606
(3,3)	8	9	-11.68	5.364	3.872	2.463
(3,4)	9	10	-10.44	8.812	3.899	2.290
(3,5)	10	11	-7.911	10.42	3.810	**2.284**
(3,6)	11	12	-7.909	10.43	3.951	2.424
(4,1)	7	8	-22.01	-7.362	4.670	3.479
(4,2)	8	9	-13.06	2.428	3.998	2.730
(4,3)	9	10	-11.65	5.822	4.010	2.562
(4,4)	10	11	-10.00	8.928	4.000	2.420
(4,5)	11	12	-6.377	11.21	3.811	*2.353*
(4,6)	12	13	-6.328	11.23	3.947	2.492
(5,1)	8	9	-21.96	-7.316	4.807	3.616
(5,2)	9	10	-13.06	2.768	4.138	2.839
(5,3)	10	11	-11.62	6.695	4.147	2.623
(5,4)	11	12	-9.914	9.848	4.133	2.477
(5,5)	12	13	-4.365	13.98	3.769	*2.241*
(5,6)	13	14	-4.242	14.34	3.898	2.349
(6,4)	12	13	-7.730	12.34	4.075	2.391
(6,5)	13	14	-2.415	16.81	3.732	**2.125**
(6,6)	14	15	-2.247	16.84	3.857	2.263
(7,4)	13	14	-7.481	12.40	4.193	2.526
(7,5)	14	15	-2.297	16.81	3.862	*2.266*
(7,6)	15	16	-2.145	16.84	3.989	2.404

Table 7: Parameter estimates for the best two Poly(d, q) choices of ACDN and ACDT.

	Poly$(6,5)$				Poly$(3,5)$			
	ACDN		ACDT		ACDN		ACDT	
	MLE	SE	MLE	SE	MLE	SE	MLE	SE
γ_0	1.0026	0.1828	0.9755	0.1957	0.9393	0.1722	0.9292	0.1845
γ_1	0.4299	0.4335	0.0853	0.4538	0.3374	0.4037	0.0841	0.4290
γ_2	0.1969	0.2155	0.1426	0.2372	0.1463	0.2007	0.1666	0.2260
γ_3	0.4648	0.1451	0.5574	0.1585	0.2430	0.1137	0.3917	0.1270
γ_4	0.3352	0.1172	0.2655	0.1207	.	.	.	.
γ_5	0.1627	0.0922	0.1616	0.0926	.	.	.	.
γ_6	-0.1458	0.0679	-0.1693	0.0679	.	.	.	.
λ_0	-1.4098	0.0435	-1.7097	0.1003	-1.3890	0.0435	-1.6733	0.0981
λ_1	-0.7341	0.1505	-0.5697	0.1597	-0.6866	0.1501	-0.5311	0.1584
λ_2	0.5473	0.1204	0.6631	0.1271	0.5148	0.1244	0.6080	0.1308
λ_3	-0.1595	0.0961	-0.2767	0.1001	-0.1500	0.1010	-0.2727	0.1049
λ_4	-0.1174	0.0850	-0.2006	0.0849	-0.0436	0.0864	-0.1522	0.0870
λ_5	-0.2492	0.0752	-0.2164	0.0731	-0.1348	0.0716	-0.1080	0.0711
ν	.	.	3.4490	1.1747	.	.	3.6626	1.2688

4.5 Technical Details

4.5.1 $\boldsymbol{L}_\gamma$ and $\boldsymbol{D}_\lambda$, for Linear Link Functions $v(\cdot,\cdot)$ and $d(\cdot,\cdot)$

Considering $\theta_{tj} = d(\boldsymbol{z}_{tj}, \boldsymbol{\gamma}) := \boldsymbol{z}_{tj}^\top \boldsymbol{\gamma}$ and $\log(\sigma_t) = v(\boldsymbol{z}_t, \boldsymbol{\lambda}) := \boldsymbol{z}_t^\top \boldsymbol{\lambda}$, where the matrices $\boldsymbol{L}$ and $\boldsymbol{D}$ has been defined as following:

$$\boldsymbol{L} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ \theta_{21} & 1 & 0 & \ddots & \vdots \\ \theta_{31} & \theta_{32} & 1 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \theta_{p1} & \dots & \theta_{p(p-2)} & \theta_{p(p-1)} & 1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{D} = \begin{pmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \sigma_p \end{pmatrix}$$

Simply, taking termwise derivative of $\boldsymbol{L}$ and $\boldsymbol{D}$ with respect to γ_r and λ_s lead us to the following:

$$\boldsymbol{L}_{\gamma_r} = \boldsymbol{Z}_{L,r}, \quad \text{and} \quad \boldsymbol{D}_{\lambda_s} = (\boldsymbol{Z}_{D,s})\boldsymbol{D},$$

where

$$\boldsymbol{Z}_{L,r} = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 \\ z_{21_r} & 0 & 0 & \ddots & \vdots \\ z_{31_r} & z_{32_r} & 0 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ z_{p1_r} & \dots & z_{p(p-2)_r} & z_{p(p-1)_r} & 0 \end{pmatrix} \quad \text{and} \quad \boldsymbol{Z}_{D,s} = \begin{pmatrix} z_{1_s} & 0 & \dots & 0 \\ 0 & z_{2_s} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & z_{p_s} \end{pmatrix}.$$

Using the above linear link functions, make the score function and the Fisher information matrix more simpler. $U(\boldsymbol{\beta})$, $U(\nu)$ remain unchanged but $U(\gamma_r)$, and $U(\lambda_r)$ becomes simpler as following:

$$\begin{aligned} U(\gamma_r) &= \operatorname{tr}\Big((\boldsymbol{T}\boldsymbol{D}^{-1})\Big(\sum_{i=1}^{n} \tau_i \boldsymbol{S}_i\Big)(\boldsymbol{T}\boldsymbol{D}^{-1})^\top \boldsymbol{T}\boldsymbol{Z}_{L_r}\Big), \\ U(\lambda_s) &= \operatorname{tr}\Big(\Big(\Big(\sum_{i=1}^{n} \tau_i \boldsymbol{S}_i\Big)\boldsymbol{\Sigma}^{-1} - n\boldsymbol{I}\Big)\boldsymbol{Z}_{D,s}\Big), \end{aligned}$$

Also the Fisher information that involves β remains unchanged

$$\boldsymbol{I}_{11}(\boldsymbol{\beta}) = \frac{\nu+p}{\nu+p+2}\sum_{i=1}^{n} \boldsymbol{X}_i^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{X}_i, \quad \boldsymbol{I}_{12}(\boldsymbol{\beta},\boldsymbol{\gamma}) = \boldsymbol{I}_{13}(\boldsymbol{\beta},\boldsymbol{\lambda}) = \boldsymbol{I}_{14}(\boldsymbol{\beta},\nu) = 0.$$

Finally $\boldsymbol{I}_{24}(\boldsymbol{\gamma},\nu)$ and $I_{44}(\nu)$ remain the same but the rest of Fisher information have the following simpler form:

$$\begin{aligned} I_{22,rs}(\boldsymbol{\gamma}) &= \frac{(\nu+p)n}{\nu+p+2}\operatorname{tr}(\boldsymbol{Z}_{L,r}\boldsymbol{Z}_{L_s}^\top \boldsymbol{T}^\top \boldsymbol{T}), \\ I_{33,rs}(\boldsymbol{\lambda}) &= \frac{(\nu+p)n}{\nu+p+2}\operatorname{tr}\Big((\boldsymbol{Z}_{D,r} + \boldsymbol{L}\boldsymbol{L}^\top \boldsymbol{Z}_{D,r}\boldsymbol{T}^\top \boldsymbol{T})\boldsymbol{Z}_{D,s}\Big) \\ &\quad - \frac{2n}{\nu+p+2}\operatorname{tr}(\boldsymbol{Z}_{D,r})\operatorname{tr}(\boldsymbol{Z}_{D,s}), \\ I_{23,rs}(\boldsymbol{\gamma},\boldsymbol{\lambda}) &= \frac{(\nu+p)n}{\nu+p+2}\operatorname{tr}(\boldsymbol{Z}_{L,r}\boldsymbol{L}^\top \boldsymbol{Z}_{D,s}\boldsymbol{T}^\top \boldsymbol{T}), \\ I_{34,r}(\boldsymbol{\lambda},\nu) &= -\frac{2n}{(\nu+p+2)(\nu+p)}\operatorname{tr}(\boldsymbol{Z}_{D,r}). \end{aligned}$$

4.5.2 Fisher Scoring for the Multivariate Normal distribution

We can let $\nu \to \infty$, and obtain the associated score function and the fisher information for the multivariate normal distribution. Since we did not find the results in the literature, it's worth to be included here. The log likelihood function $\ell(\theta)$, up to an additive constant is

$$-\frac{2}{n}\ell(\theta) = \log|\boldsymbol{D}^2| + n^{-1}\sum_{i=1}^{n} r_i^\top \boldsymbol{\Sigma}^{-1} r_i = \sum_{t=1}^{p} \log\sigma_t^2 + \mathrm{tr}\boldsymbol{S}\boldsymbol{\Sigma}^{-1}. \tag{4.5}$$

4.5.2.1 The Score Function

Under the normality assumption given in (4.5), the score function can be obtained using the results given in subsection 4.5.4 with some simplification as following:

$$\begin{aligned}
U_r(\boldsymbol{\beta}) &= \sum_{i=1}^{n}\Big(\frac{\partial}{\partial\beta_r}\boldsymbol{\mu}_i\Big)^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{r}_i \quad\Rightarrow\quad U(\boldsymbol{\beta}) = \sum_{i=1}^{n}\boldsymbol{X}_i^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{r}_i, \\
U(\gamma_r) &= n\mathrm{tr}\bigg((\boldsymbol{T}\boldsymbol{D}^{-1})\boldsymbol{S}(\boldsymbol{T}\boldsymbol{D}^{-1})^\top \boldsymbol{T}\boldsymbol{L}_{\gamma_r}\bigg), \\
U(\lambda_s) &= n\mathrm{tr}\bigg((\boldsymbol{S}\boldsymbol{\Sigma}^{-1} - I)\boldsymbol{D}^{-1}\boldsymbol{D}_{\boldsymbol{\lambda}_s}\bigg).
\end{aligned}$$

Therefore the $U(\boldsymbol{\theta}) = \big(U^\top(\boldsymbol{\beta}), U^\top(\boldsymbol{\gamma}), U^\top(\boldsymbol{\lambda})\big)^\top$ is easy to calculate.

4.5.2.2 The Fisher Information

Fisher information $\boldsymbol{I}(\boldsymbol{\theta})$ has been considered as a 3×3 matrix of submatrices based on expectation of Hessian matrix as following:

$$\begin{aligned}
\boldsymbol{H}(\boldsymbol{\theta}) &= \begin{pmatrix} \boldsymbol{H}(\boldsymbol{\beta}) & \boldsymbol{H}(\boldsymbol{\beta},\gamma) & \boldsymbol{H}(\boldsymbol{\beta},\boldsymbol{\lambda}) \\ \boldsymbol{H}^\top(\boldsymbol{\beta},\gamma) & \boldsymbol{H}(\gamma) & \boldsymbol{H}(\gamma,\boldsymbol{\lambda}) \\ \boldsymbol{H}^\top(\boldsymbol{\beta},\boldsymbol{\lambda}) & \boldsymbol{H}^\top(\gamma,\boldsymbol{\lambda}) & \boldsymbol{H}(\boldsymbol{\lambda}) \end{pmatrix} \quad\Rightarrow \\
\boldsymbol{I}(\boldsymbol{\theta}) = -\mathbb{E}(H(\boldsymbol{\theta})) &= \begin{pmatrix} \boldsymbol{I}_{11}(\boldsymbol{\beta}) & \boldsymbol{I}_{12}(\boldsymbol{\beta},\gamma) & \boldsymbol{I}_{13}(\boldsymbol{\beta},\boldsymbol{\lambda}) \\ \boldsymbol{I}_{21}(\gamma,\boldsymbol{\beta}) & \boldsymbol{I}_{22}(\gamma) & \boldsymbol{I}_{23}(\gamma,\boldsymbol{\lambda}) \\ \boldsymbol{I}_{31}(\boldsymbol{\lambda},\boldsymbol{\beta}) & \boldsymbol{I}_{32}(\boldsymbol{\lambda},\gamma) & \boldsymbol{I}_{33}(\boldsymbol{\lambda}) \end{pmatrix}.
\end{aligned}$$

Now by using the score function $U(\boldsymbol{\theta})$ and the results given in subsection 4.5.4, we can obtain the elements in each submatrices that involves $\boldsymbol{\beta}$ as following:

$$\begin{aligned}
h_{rs}(\boldsymbol{\beta}) &= \sum_{i=1}^{n}\Bigg\{\Big(\frac{\partial}{\partial\beta_s}(\frac{\partial}{\partial\beta_r}\boldsymbol{\mu}_i)\Big)^{\top}\boldsymbol{\Sigma}^{-1}\boldsymbol{r}_i - (\frac{\partial}{\partial\beta_r}\boldsymbol{\mu}_i)^{\top}\boldsymbol{\Sigma}^{-1}(\frac{\partial}{\partial\beta_s}\boldsymbol{\mu}_i)\Bigg\},\\
h_{rs}(\beta_r,\gamma_s) &= \sum_{i=1}^{n}(\frac{\partial}{\partial\beta_r}\boldsymbol{\mu}_i)^{\top}(\boldsymbol{\Sigma}^{-1})^{(\gamma_s)}\boldsymbol{r}_i,\\
h_{rs}(\beta_r,\lambda_s) &= \sum_{i=1}^{n}(\frac{\partial}{\partial\beta_r}\boldsymbol{\mu}_i)^{\top}(\boldsymbol{\Sigma}^{-1})^{(\lambda_s)}\boldsymbol{r}_i.
\end{aligned}$$

Hence the associated submatrices of the Fisher information could be obtained by following:

$$\boldsymbol{I}_{11}(\boldsymbol{\beta}) = \sum_{i=1}^{n}\boldsymbol{X}_i^{\top}\boldsymbol{\Sigma}^{-1}\boldsymbol{X}_i, \qquad \boldsymbol{I}_{12}(\boldsymbol{\beta},\boldsymbol{\gamma}) = \boldsymbol{0}, \qquad \boldsymbol{I}_{13}(\boldsymbol{\beta},\boldsymbol{\lambda}) = \boldsymbol{0}.$$

In addition we can use the results given in subsection 4.5.3 and obtain the rest of the elements in the Fisher information matrix as following:

$$\begin{aligned}
I_{22,rs}(\boldsymbol{\gamma}) = \mathbb{E}(-\ell_{\gamma_r\gamma_s}) = \mathbb{E}(\ell_{\gamma_r}\ell_{\gamma_s}) &= n\mathrm{tr}(\boldsymbol{L}_{\gamma_r}\boldsymbol{L}_{\gamma_s}^{\top}\boldsymbol{T}^{\top}\boldsymbol{T}),\\
I_{33,rs}(\boldsymbol{\lambda}) = \mathbb{E}(-\ell_{\lambda_r\lambda_s}) = \mathbb{E}(\ell_{\lambda_r}\ell_{\lambda_s}) &= n\mathrm{tr}(\boldsymbol{D}^{-2}\boldsymbol{D}_{\lambda_r}\boldsymbol{D}_{\lambda_s} + \boldsymbol{L}\boldsymbol{L}^{\top}\boldsymbol{D}_{\lambda_r}\boldsymbol{\Sigma}^{-1}\boldsymbol{D}_{\lambda_s}),\\
I_{23,rs}(\boldsymbol{\gamma},\boldsymbol{\lambda}) = \mathbb{E}(-\ell_{\gamma_r\lambda_s}) = \mathbb{E}(\ell_{\gamma_r}\ell_{\lambda_s}) &= n\mathrm{tr}(\boldsymbol{D}\boldsymbol{L}_{\gamma_r}\boldsymbol{L}^{\top}\boldsymbol{D}_{\lambda_s}\boldsymbol{\Sigma}^{-1}).
\end{aligned}$$

4.5.2.3 $\boldsymbol{L}_\gamma$ and $\boldsymbol{D}_\lambda$, for Linear Link Functions $v(\cdot,\cdot)$ and $d(\cdot,\cdot)$.

Similar to the subsection 4.5.1, we could take $\theta_{tj} = d(\boldsymbol{z}_{tj},\boldsymbol{\gamma}) := \boldsymbol{z}_{tj}^{\top}\boldsymbol{\gamma}$ and $\log(\sigma_t) = v(\boldsymbol{z}_t,\boldsymbol{\lambda}) := \boldsymbol{z}_t^{\top}\boldsymbol{\lambda}$. The resulting score function and the Fisher information becomes much simpler as following:

$$\begin{aligned}
U(\boldsymbol{\beta}) &= \sum_{i=1}^{n}\boldsymbol{X}_i^{\top}\boldsymbol{\Sigma}^{-1}\boldsymbol{r}_i,\\
U(\gamma_r) &= n\mathrm{tr}\Bigg((\boldsymbol{T}\boldsymbol{D}^{-1})\boldsymbol{S}(\boldsymbol{T}\boldsymbol{D}^{-1})^{\top}\boldsymbol{T}\boldsymbol{Z}_{L_r}\Bigg),\\
U(\lambda_s) &= n\mathrm{tr}\Bigg((\boldsymbol{S}\boldsymbol{\Sigma}^{-1} - \boldsymbol{I})\boldsymbol{Z}_{D,r}\Bigg).
\end{aligned}$$

Also the Fisher information that involves $\boldsymbol{\beta}$ remains unchanged

$$\boldsymbol{I}_{11}(\boldsymbol{\beta}) = \sum_{i=1}^{n} \boldsymbol{X}_i^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{X}_i, \quad \boldsymbol{I}_{12}(\boldsymbol{\beta}, \boldsymbol{\gamma}) \;=\; \boldsymbol{I}_{13}(\boldsymbol{\beta}, \boldsymbol{\lambda}) \;=\; \mathbf{0}.$$

Finally the rest of Fisher information have the following form:

$$\begin{aligned} I_{22,rs}(\boldsymbol{\gamma}) &= n\mathrm{tr}(\boldsymbol{Z}_{L,r}\boldsymbol{Z}_{L_s}^\top \boldsymbol{T}^\top \boldsymbol{T}), \\ I_{33,rs}(\boldsymbol{\lambda}) &= n\mathrm{tr}\bigg((\boldsymbol{Z}_{D,r} + \boldsymbol{L}\boldsymbol{L}^\top \boldsymbol{Z}_{D,r}\boldsymbol{T}^\top \boldsymbol{T})\boldsymbol{Z}_{D,s}\bigg), \\ I_{23,rs}(\boldsymbol{\gamma}, \boldsymbol{\lambda}) &= n\mathrm{tr}(\boldsymbol{Z}_{L,r}\boldsymbol{L}^\top \boldsymbol{Z}_{D,s}\boldsymbol{T}^\top \boldsymbol{T}). \end{aligned}$$

4.5.2.4 Penalized Normal Likelihood

Similar to the Pourahmadi (1999), one can use the regressogram for the sample covariance to obtain the appropriate design matrices $\boldsymbol{z}_t$ for modeling the log-innovation standard deviation and $\boldsymbol{z}_{tj}$ for modeling the generalized moving average parameters. Figure 16 is showing the regressogram for the Kenward (1987)'s cattle data and the least square and maximum likelihood estimates based on a cubic MCDN, ACDN and ACDT models respectively. However, not always regressogram provides reasonable relationship to obtain such design matrices. Alternative approach is to use the penalized likelihood for the large sparse covariance structure. Huang et al. (2006) provided the penalized normal likelihood(L_1 and L_2 penalty) to model the inverse of the covariance matrix. We extend their algorithms to obtain a similar machinery for estimating the shrinkage estimate of the $\boldsymbol{\Sigma}$ directly. The difference is just adding the penalty term to (4.5), and maximizing the following objective function:

$$\ell_p(\boldsymbol{\theta}) = -\frac{n}{2}\log|\boldsymbol{D}^2| - \frac{n}{2}\mathrm{tr}\boldsymbol{S}\boldsymbol{\Sigma}^{-1} - \frac{\alpha}{2}\sum_{t=2}^{p}\sum_{j=1}^{t-1}|\theta_{tj}|^p. \tag{4.6}$$

We can extend our Fisher scoring algorithm by using the associated quadratic form to the L_2 penalty or approximating the associated L_1 penalty with a reasonable quadratic form (which leads to the special case of MM algorithm).

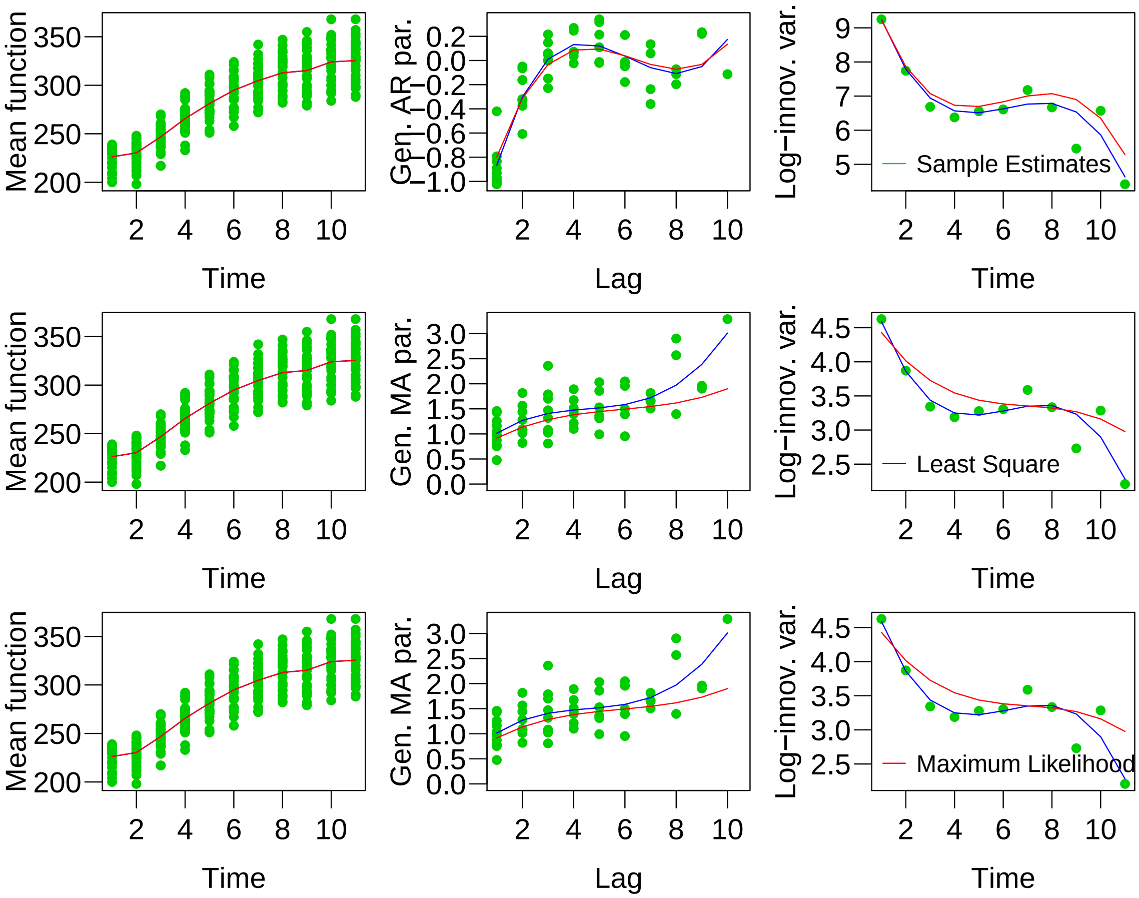

Figure 16: (First row:) The profile plot, GARPs and log-innovation standard deviation plots for the MCDN model. (Second row:) is associated to the ACDN fit and associated profile plot, GMAPs and the log-innovation standard deviation plots, and (Last row:) is for the ACDT model. Green color is for the sample covariance estimates, red indicates the MLEs and blue color is associated to the LSEs.

4.5.3 Quadratic Forms

If $\boldsymbol{\epsilon}$ is a vector of n random variables, and $\boldsymbol{\Lambda}$ is a symmetric matrix, then the $\boldsymbol{\epsilon}^\top\boldsymbol{\Lambda}\boldsymbol{\epsilon}$ is known as a quadratic form in $\boldsymbol{\epsilon}$. It is known that

$$\mathbb{E}(\boldsymbol{\epsilon}^\top\boldsymbol{\Lambda}\boldsymbol{\epsilon}) = \mathrm{tr}(\boldsymbol{\Lambda}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^\top\boldsymbol{\Lambda}\boldsymbol{\mu},$$

where $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are the expected value and covariance matrix of $\boldsymbol{\epsilon}$, respectively. Here, the normality of $\boldsymbol{\epsilon}$ is not required. In general, the variance of a quadratic form depends greatly on the distribution of $\boldsymbol{\epsilon}$.

4.5.3.1 The Multivariate Normal

Assuming $\boldsymbol{\epsilon}$ follow a multivariate normal distribution and $\boldsymbol{\Lambda}$ is a symmetric matrix. The variance of the quadratic form can be obtained as

$$\mathrm{var}(\boldsymbol{\epsilon}^\top\boldsymbol{\Lambda}\boldsymbol{\epsilon}) = 2\mathrm{tr}(\boldsymbol{\Lambda}\boldsymbol{\Sigma}\boldsymbol{\Lambda}\boldsymbol{\Sigma}) + 4\boldsymbol{\mu}^\top\boldsymbol{\Lambda}\boldsymbol{\Sigma}\boldsymbol{\Lambda}\boldsymbol{\mu}.$$

In fact, this can be generalized to find the covariance between two quadratic forms on the same $\boldsymbol{\epsilon}$ (once again, $\boldsymbol{\Lambda}_1$ and $\boldsymbol{\Lambda}_2$ must both be symmetric),

$$\mathrm{cov}(\boldsymbol{\epsilon}^\top\boldsymbol{\Lambda}_1\boldsymbol{\epsilon}, \boldsymbol{\epsilon}^\top\boldsymbol{\Lambda}_2\boldsymbol{\epsilon}) = 2\mathrm{tr}(\boldsymbol{\Lambda}_1\boldsymbol{\Sigma}\boldsymbol{\Lambda}_2\boldsymbol{\Sigma}) + 4\boldsymbol{\mu}^\top\boldsymbol{\Lambda}_1\boldsymbol{\Sigma}\boldsymbol{\Lambda}_2\boldsymbol{\mu}.$$

4.5.3.2 Multivariate t_ν

If $\boldsymbol{\epsilon}$ follow a multivariate t_ν distribution, using Kim and Mallick (2003) we have:

$$\mathrm{var}(\boldsymbol{\epsilon}^\top\boldsymbol{\Lambda}\boldsymbol{\epsilon}) = \frac{2\nu^2}{(\nu-2)(\nu-4)}\mathrm{tr}(\boldsymbol{\Lambda}\boldsymbol{\Sigma}\boldsymbol{\Lambda}\boldsymbol{\Sigma}) + \frac{2\nu^2}{(\nu-2)^2(\nu-4)}(\mathrm{tr}(\boldsymbol{\Lambda}\boldsymbol{\Sigma}))^2 + \frac{4\nu}{\nu-2}\boldsymbol{\mu}^\top\boldsymbol{\Lambda}\boldsymbol{\Sigma}\boldsymbol{\Lambda}\boldsymbol{\mu},$$

or more generally:

$$\begin{aligned}\mathrm{cov}(\boldsymbol{\epsilon}^\top\boldsymbol{\Lambda}_1\boldsymbol{\epsilon}, \boldsymbol{\epsilon}^\top\boldsymbol{\Lambda}_2\boldsymbol{\epsilon}) &= \frac{2\nu^2}{(\nu-2)(\nu-4)}\mathrm{tr}(\boldsymbol{\Lambda}_1\boldsymbol{\Sigma}\boldsymbol{\Lambda}_2\boldsymbol{\Sigma}) \\ &+ \frac{2\nu^2}{(\nu-2)^2(\nu-4)}\mathrm{tr}(\boldsymbol{\Lambda}_1\boldsymbol{\Sigma})\mathrm{tr}(\boldsymbol{\Lambda}_2\boldsymbol{\Sigma})^2 + \frac{4\nu}{\nu-2}\boldsymbol{\mu}^\top\boldsymbol{\Lambda}_1\boldsymbol{\Sigma}\boldsymbol{\Lambda}_2\boldsymbol{\mu}.\end{aligned}$$

4.5.3.3 Non-Symmetric $\boldsymbol{\Lambda}$

The case for general $\boldsymbol{\Lambda}$ can be derived by noting that $\boldsymbol{\epsilon}^\top \boldsymbol{\Lambda}^\top \boldsymbol{\epsilon} = \boldsymbol{\epsilon}^\top \boldsymbol{\Lambda} \boldsymbol{\epsilon}$ so that, $\boldsymbol{\epsilon}^\top \tilde{\boldsymbol{\Lambda}} \boldsymbol{\epsilon} = \boldsymbol{\epsilon}^\top \left(\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^\top\right) \boldsymbol{\epsilon}/2$. But this is a quadratic form in the symmetric matrix $\tilde{\boldsymbol{\Lambda}} = \left(\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^\top\right)/2$, therefore the mean and variance expressions are the same, provided $\boldsymbol{\Lambda}$ is replaced by $\tilde{\boldsymbol{\Lambda}}$ therein.

4.5.4 Multivariate Calculus and Some Useful Identities

Let $\boldsymbol{x}$ be a vector, and $\boldsymbol{A}(y), \boldsymbol{X}, \boldsymbol{Y}$ and $\boldsymbol{W}$ four different matrices. Using Petersen and Pedersen (2008) we have the following useful results:

- $\dfrac{\partial}{\partial \boldsymbol{x}} \boldsymbol{X}^\top = \left(\dfrac{\partial}{\partial \boldsymbol{x}} \boldsymbol{X}\right)^\top.$
- $\dfrac{\partial}{\partial \boldsymbol{x}} (\boldsymbol{X}\boldsymbol{Y}) = \left(\dfrac{\partial}{\partial \boldsymbol{x}} \boldsymbol{X}\right)\boldsymbol{Y} + \boldsymbol{X}\left(\dfrac{\partial}{\partial \boldsymbol{x}} \boldsymbol{Y}\right).$
- $\dfrac{\partial}{\partial \boldsymbol{x}} \operatorname{tr}(\boldsymbol{X}) = \operatorname{tr}\left(\dfrac{\partial}{\partial \boldsymbol{x}} \boldsymbol{X}\right).$
- $\dfrac{\partial \boldsymbol{X}^{-1}}{\partial \boldsymbol{x}} = -\boldsymbol{X}^{-1} \dfrac{\partial \boldsymbol{X}}{\partial \boldsymbol{x}} \boldsymbol{X}^{-1}.$
- $\dfrac{\partial}{\partial y} \left(\ln(\det(\boldsymbol{A}(y)))\right) = \operatorname{tr}\left(\boldsymbol{A}^{-1}(y) \dfrac{\partial \boldsymbol{A}(y)}{\partial y}\right).$
- $\dfrac{\partial}{\partial y} \left(\left((\boldsymbol{x} - \boldsymbol{A}(y)\right)^\top \boldsymbol{W} \left(\boldsymbol{x} - \boldsymbol{A}(y)\right)\right) = -2\left(\dfrac{\partial}{\partial y} \boldsymbol{A}(y)\right) \boldsymbol{W} \left(\boldsymbol{x} - \boldsymbol{A}(y)\right).$

CHAPTER V

CONCLUSION

First, we proposed a profile likelihood approach to estimate a transformation model for affymetrix short oligonucleotide array data. The proposed method is more statistically principled and efficient than the ad-hoc entropy based method introduced in Hu et al. (2006) that is evident by our simulation studies. Its computationally efficiency comes with the simple SVD method used for parameter estimation. The real data example shows its superiority of empirical performance to the popular Li-Wong model. We also introduced a multivariate expression index which utilizes the first two singular vectors. Our empirical investigation shows the promise of using multivariate index in terms of both model fitting and differential expression detection. In addition, we re-examined two important practical issues in gene expression analysis, the value of normalization and statistical p-values. Our study shows that, when using the proposed method for generating expression indexes, normalization has impact on differential expression detection and statistical p-values have better performance than the simple fold change criterion in terms of gene ranking.

Next, we introduced a probabilistic model based on statistically principled procedure to obtain an appropriate transformation for the PCA and the FPCA analysis that commonly has been selected by researchers based on some ad-hoc explanations. The proposed method tends to obtain non-informative homogeneous normally distributed residuals from a likelihood based approach. It has been illustrated that how crucial it could be to choose the right transformation for obtaining the FPCs using simulations and a real data example (Call Center data). The automaticity and the high performance of the proposed profile likelihood approach are the main advantages of this routine. Also we considered two algorithms for imputing the missing values, and we studied the connection of those algorithms with normality of the residuals. Using the proposed PSVDM method for imputation in the “fruit fly mortality” data, we showed another importance of the proposed automatic procedure to obtain the right transformation.

Finally, we have established the role of an alternative Cholesky decomposition of the covariance matrix of a longitudinal dataset in providing robust estimator of its correlation matrix. Robustness to outliers is handled using heavy-tailed multivariate t-distributions with unknown degrees of freedom. Newton-Raphson algorithm with Fisher scoring for computing the maximum likelihood estimators of the parameters of the alternative Cholesky decomposition turns out to be more complicated than the standard Cholesky decomposition. This computational complexity is comparable to maximum likelihood estimation of parameters of the moving average models from time series analysis.

REFERENCES

Achlioptas, D., Mcsherry, F., 2007. Fast computation of low-rank matrix approximations. Journal of the ACM 54.

Anscombe, F.J., 1948. The transformation of poisson, binomial and negative-binomial data. Biometrika 35, 246–254.

Avriel, M., 1976. Nonlinear Programming: Analysis and Methods. Prentice-Hall, Englewood Cliffs, NJ.

Bar-Lev, S.K., Enis, P., 1988. On the classical choice of variance stabilizing transformations and an application for a poisson variate. Biometrika 75, 803–804.

Beckers, J.M., Rixen, M., 2003. Eof calculations and data filling from incomplete oceanographic datasets. Journal of Atmospheric and Oceanic Technology 20, 1839–1856.

Bolstad, B.M., Irizarry, R.A., Astrand, M., Speed, T.P., 2003. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. Bioinformatics 19, 185–193.

Box, G.E.P., Cox, D.R., 1964. An analysis of transformations. Journal of the Royal Statistical Society. Series B 26, 211–252.

Brockwell, P.J., Davis, R.A., 1991. Time Series: Theory and Methods. 2nd edition. Springer, New York.

Brown, L., Gans, N., Mandelbaum, A., Sakov, A., Shen, H., Zeltyn, S., Zhao, L.,

2005. Statistical analysis of a telephone call center: A queueing-science perspective. Journal of the American Statistical Association 100, 36–50.

Byrd, R.H., Byrd, R.H., Lu, P., Lu, P., Nocedal, J., Nocedal, J., Zhu, C., Zhu, C., 1994. A limited memory algorithm for bound constrained optimization. SIAM Journal on Scientific Computing 16, 1190–1208.

Cai, B., Dunson, D.B., Gladen, T.B., 2006. Bayesian covariance selection in generalized linear mixed models. Biometrics 62, 446–457.

Cannon, M.J., Warner, L., Taddei, J.A., Kleinbaum, D.G., 2001. What can go wrong when you assume that correlated data are independent: An illustration from the evaluation of a childhood health intervention in Brazil. Statistics in Medicine 20, 1461–1467.

Carroll, R.J., 2003. Variances are not always nuisance parameters. Biometrics 59, 211–220.

Chen, Z., Dunson, D.B., 2003. Random effects selection in linear mixed models. Biometrics 59, 762–769.

Chiu, T.Y.M., Leonard, T., Tsui, K.W., 1996. The matrix-logarithmic covariance model. Journal of the American Statistical Association 91, 198–210.

Chu, T.M., Weir, B., Wolfinger, R., 2002. A systematic statistical linear modeling approach to oligonucleotide array experiments. Mathematical Biosciences 176, 35–51.

Craven, P., Wahba, G., 1979. Smoothing noisy data with spline functions: Estimating the correct degree of smoothing by the method of generalized cross-validation. Numerische Mathematik 31, 377–403.

Croux, C., Ruiz-Gazen, A., 2005. High breakdown estimators for principal components: the projection-pursuit approach revisited. Journal of Multivariate Analysis 95, 206 – 226.

Daniels, M.J., Hogan, J.W., 2008. Missing data in longitudinal studies: Strategies for Bayesian modeling and sensitivity analysis; electronic version. Monographs on Statistics and Applied Probability. Taylor & Francis Ltd, Hoboken, NJ.

Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the em algorithm. Journal of the Royal Statistical Society. Series B 39, 1–38.

Diggle, P., Heagerty, P., Liang, K.Y., Zeger, S., 2002. Analysis of Longitudinal Data. 2nd edition. Oxford University Press, New York.

Diggle, P.J., Verbyla, A.n.P., 1998. Nonparametric estimation of covariance structure in longitudinal data. Biometrics 54, 401–415.

Geller, S.C., Gregg, J.P., Hagerman, P., Rocke, D.M., 2003. Transformation and normalization of oligonucleotide microarray data. Bioinformatics 19, 1817–1823.

Green, P.J., Silverman, B.W., 1994. Nonparametric regression and generalized linear models: A roughness penalty approach, Vol 58 of Monographs on Statistics and Applied Probability. Chapman & Hall, London.

Hawkins, D.L., 1989. Using u statistics to derive the asymptotic distribution of fisher's z statistic. The American Statistician 43, 235–237.

Higuchi, I., Eguchi, S., 2004. Robust principal component analysis with adaptive selection for tuning parameters. Journal of Machine Learning Research 5, 453–471.

Holan, S., Spinka, C., 2007. Maximum likelihood estimation for joint mean-covariance models from unbalanced repeated-measures data. Statistics & Probability Letters 77, 319–328.

Hu, J., He, X., Cote, G., Krahe, R., 2009. Singular value decomposition-based alternative splicing detection. Journal of the American Statistical Association 104, 944–953.

Hu, J., Wright, F.A., Zou, F., 2006. Estimation of expression indexes for oligonucleotide arrays using the singular value decomposition. Journal of the American Statistical Association 101, 41–50.

Huang, J.Z., Liu, N., Pourahmadi, M., Liu, L., 2006. Covariance matrix selection and estimation via penalised normal likelihood. Biometrika 93, 85–98.

Huang, J.Z., Shen, H., Buja, A., 2008. Functional principal components analysis via penalized rank one approximation. Electronic Journal of Statistics 2, 678–695.

Hubert, M., Rousseeuw, P., Verdonck, T., 2009. Robust pca for skewed data and its outlier map. Computational Statistics & Data Analysis 53, 2264 – 2274.

Hubert, M., Rousseeuw, P.J., Verboven, S., 2002. A fast method for robust principal components with applications to chemometrics. Chemometrics and Intelligent Laboratory Systems 60, 101 – 111.

Irizarry, R.A., Bolstad, B.M., Collin, F., Cope, L.M., Hobbs, B., Speed, T.P., 2003a. Summaries of affymetrix genechip probe level data. Nucleic Acids Res 31.

Irizarry, R.A., Hobbs, B., Collin, F., Beazer-Barclay, Y.D., Antonellis, K.J., Scherf, U., Speed, T.P., 2003b. Exploration, normalization, and summaries of high density oligonucleotide array probe level data. Biostatistics 4, 249–264.

Kenward, M.G., 1987. A method for comparing profiles of repeated measurements. Applied Statistics 36, 296–308.

Kim, H.M., Mallick, B.K., 2003. Moments of random vectors with skew t distribution and their quadratic forms. Statistics & Probability Letters 63, 417 – 423.

Lange, K.L., Little, R.J.A., Taylor, J.M.G., 1989. Robust statistical modeling using the t distribution. Journal of the American Statistical Association 84, 881–896.

Lemon, W.J., Palatini, J.J., Krahe, R., Wright, F.A., 2002. Theoretical and experimental comparisons of gene expression indexes for oligonucleotide arrays. Bioinformatics 18, 1470–1476.

Leng, C., Zhang, W., Pan, J., 2010. Semiparametric mean-covariance regression analysis for longitudinal data. Journal of the American Statistical Association 105, 181–193.

Li, C., Wong, W.H., 2001. Model-based analysis of oligonucleotide arrays: Expression index computation and outlier detection. PNAS 98, 31–36.

Liang, K.Y., Zeger, S.L., 1986. Longitudinal data analysis using generalized linear models. Biometrika 73, 13–22.

Lin, T.I., Wang, Y.J., 2009. A robust approach to joint modeling of mean and scale covariance for longitudinal data. Journal of Statistical Planning and Inference 139, 3013 – 3026.

Locantore, N., Marron, J., Simpson, D., Tripoli, N., Zhang, J., Cohen, K., Boente, G., Fraiman, R., Brumback, B., Croux, C., Fan, J., Kneip, A., Marden, J., P, D., 1999. Robust principal component analysis for functional data. TEST: An Official Journal of the Spanish Society of Statistics and Operations Research 8, 1–73.

Lockhart, D.J., Dong, H., Byrne, M.C., Follettie, M.T., Gallo, M.V., Chee, M.S., Mittmann, M., Wang, C., Kobayashi, M., Norton, H., Brown, E.L., 1996. Expression monitoring by hybridization to high-density oligonucleotide arrays. Nature Biotechnology 14, 1675–1680.

MAQC, 2006. The microarray quality control (maqc) project shows inter- and intraplatform reproducibility of gene expression measurements. Nature Biotechnology 24, 1151–1161.

Mardia, K.V., Kent, J.T., Bibby, J.M., 1979. Multivariate Analysis. Academic Press, London.

Maronna, R., 2005. Principal components and orthogonal regression based on robust scales. Technometrics 47, 264–273.

McCullagh, P., Nelder, J.A., 1989. Generalized Linear Models. 2nd edition. Chapman & Hall, London.

Pan, J.X., MacKenzie, G., 2003. On modelling mean-covariance structures in longitudinal studies. Biometrika 90, 239–244.

Parmigiani, G., Garrett, E.S., Irizarry, R.A., Zeger, S.L., 2003. The analysis of gene expression data: Methods and Software. Springer, New York.

Petersen, K.B., Pedersen, M.S., 2008. The matrix cookbook. `http://www2.imm.dtu.dk/pubdb/p.php?3274`. [Online; accessed 1-January-2011].

Pinheiro, J.C., Liu, C., Wu, Y.N., 2001. Efficient algorithms for robust estimation in linear mixed-effects models using the multivariate t distribution. Journal of Computational and Graphical Statistics 10, 249–276.

Pourahmadi, M., 1999. Joint mean-covariance models with applications to longitudinal data: Unconstrained parameterisation. Biometrika 86, 677–690.

Pourahmadi, M., 2000. Maximum likelihood estimation of generalised linear models for multivariate normal covariance matrix. Biometrika 87, 425–435.

Pourahmadi, M., 2001. Foundations of Time Series Analysis and Prediction Theory. Wiley, New York.

Pourahmadi, M., 2007. Cholesky decompositions and estimation of a covariance matrix: Orthogonality of variance correlation parameters. Biometrika 94, 1006–1013.

Ramsay, J., Silverman, B., 2005. Functional Data Analysis. 2nd edition. Springer, New York.

Rice, J.A., Silverman, B.W., 1991. Estimating the mean and covariance structure nonparametrically when the data are curves. Journal of the Royal Statistical Society. Series B 53, 233–243.

Rothman, A.J., Levina, E., Zhu, J., 2010. A new approach to Cholesky-based covariance regularization in high dimensions. Biometrika 97, 539–550.

Shen, H., Huang, J.Z., 2008. Interday forecasting and intraday updating of call center arrivals. Manufacturing & Service Operations Management 10, 391–410.

Silverman, B.W., 1996. Smoothed functional principal components analysis by choice of norm. The Annals of Statistics 24, 1–24.

Stewart, G.W., 1993. On the early history of the singular value decomposition. SIAM Rev. 35, 551–566.

Theodoridis, S., Koutroumbas, K., 2006. Pattern Recognition. 3rd edition. Academic Press, London.

Trefethen, L.N., Bau, D., 1997. Numerical Linear Algebra. SIAM: Society for Industrial and Applied Mathematics, Philadelphia, PA.

Wang, Y.G., Carey, V., 2003. Working correlation structure misspecification, estimation and covariate design: Implications for generalised estimating equations performance. Biometrika 90, 29–41.

Welsh, A., Richardson, A., 1997. 13 approaches to the robust estimation of mixed models, in: Maddala, G., Rao, C. (Eds.), Robust Inference: Vol 15 of Handbook of Statistics, pp. 343 – 384. Elsevier, Amsterdam.

Ye, H., Pan, J.X., 2006. Modelling of covariance structures in generalised estimating equations for longitudinal data. Biometrika 93, 927–994.

Zellner, A., 1976. Bayesian and non-bayesian analysis of the regression model with multivariate student-t error terms. Journal of the American Statistical Association 71, 400–405.

Zimmerman, D.L., Núñez Antón, V., 2009. Antedependence Models for longitudinal Data. Chapman & Hall, New York.

VITA

Mehdi Maadooliat was born in 1981. He majored in applied mathematics at Sharif University of Technology, where he obtained his Bachelor of Science in 2003. He received his Master of Science in mathematics, statistics and computer science from Marquette University in 2006. He received his Ph.D. in statistics from Texas A&M University in August 2011. His research interests include computational biology, machine learning, multivariate/functional data analysis, modeling covariance matrices, and skewed non-Gaussian distributions.

He may be reached at:
Department of Statistics
Texas A&M University
3143 TAMU
College Station, TX 77843-3143.
His web page URL is http://stat.tamu.edu/~madoliat
and his email address is madoliat@stat.tamu.edu.