Estimation of Large Covariance Matrices of Longitudinal Data with Basis Function Approximations

Jianhua Z. Huang, Linxu Liu and Naiping Liu *

Abstract

The major difficulties in estimating a large covariance matrix are the high dimensionality and the positive definiteness constraint. To overcome these difficulties, we propose to apply smoothing-based regularization and utilize the modified Cholesky decomposition of the covariance matrix. In our proposal, the covariance matrix is diagonalized by a lower triangular matrix, whose subdiagonals are treated as smooth functions. These functions are approximated by splines and estimated by maximizing the normal likelihood. In our framework, the mean and the covariance of the longitudinal data can be modeled simultaneously and missing data can be handled in a natural way using the EM algorithm. We illustrate the proposed method via simulation and applying it to two real data examples, which involve estimation of 11 by 11 and 102 by 102 covariance matrices.

Keywords: Basis expansion; BIC; Cholesky decomposition; Covariance estimation; Longitudinal study; Regression spline.

*Jianhua Huang is Associate Professor, Department of Statistics, Texas A & M University, College Station, TX 77843-3143 (Email: *jianhua@stat.tamu.edu*). Linxu Liu is Assistant Professor, Department of Biostatistics, Mailman School of Public Health, Columbia University, New York, NY, 10032 (Email: *lxliu@biostat.columbia.edu*). Naiping Liu is graduate student, Department of Statistics, University of Pennsylvania, Philadelphia, PA 19104-6340 (Email: *nliu@wharton.upenn.edu*). Jianhua Huang's work is partially supported by US NSF grants DMS-0204556 and DMS-0606580. The authors would like to thank Mohsen Pourahmadi for helpful discussion. The comments from an AE and three referees help significantly improve the paper.

1 INTRODUCTION

The important task of estimation of a covariance matrix is difficult mainly because of the high dimensionality of the number of parameters and the positive definiteness constraint. The widely used sample covariance matrix can be highly unstable when the dimension is high (Lin and Perlman, 1985). In the longitudinal-data literature, it is a common practice to use parametric models for the covariance structure. Of course, the estimated covariance matrix could have considerable bias when the specified parametric model is far from the truth. Diggle and Verbyla (1998) introduced a nonparametric estimator for the covariance matrix for longitudinal data by smoothing the sample variogram ordinates and squared residuals. Their estimated covariance matrix, however, is not guaranteed to be positive-definite. To alleviate the positive-definiteness constraint, various covariance matrix decompositions have been proposed in the literature which include the variance-correlation decomposition (Barnard, McCulloch and Meng, 2000), the spectral decomposition (Chiu, Leonard and Tsui, 1996) and the Cholesky decomposition (Pourahmadi, 1999, 2000; Smith and Kohn, 2002).

In this paper we adopt the approach based on the Cholesky decomposition. The key idea is that the covariance matrix Σ of a zero-mean random vector $\mathbf{y} = (y_1, \ldots, y_m)^T$ has the following unique modified Cholesky decomposition (Newton, 1988)

$$T \Sigma T^T = D \tag{1}$$

where T is a lower triangular matrix with 1's as its diagonal entries and $D = \mathrm{diag}(d_1^2, \ldots, d_m^2)$ is a diagonal matrix. An attractive feature of this decomposition is that unlike the entries of Σ, the subdiagonal entries of T and the log of the diagonal elements of D, $\log(d_t^2), t = 1, \ldots, m$, are not constrained. Thus one can impose structures on the unconstrained parameters without worrying about the positive-definiteness constraint. More precisely, if we denote estimators of T and D in (1) by $\widehat{T}$ and $\widehat{D}$, obtained by fitting some structural models such as linear models, an estimator of Σ given by $\widehat{\Sigma} = \widehat{T}^{-1} \widehat{D} (\widehat{T}^{-1})^T$ is guaranteed to be positive-definite. This approach to covariance modeling can be thought as an extension of generalized linear models (McCullagh and Nelder, 1989), where the factorization of Σ in (1) supplies a link function $g(\Sigma) = (T, \log(D))$ where $\log(D) = \mathrm{diag}\{\log(\sigma_1^2), \ldots, \log(\sigma_m^2)\}$, and the unconstrained entries of T and $\log(D)$ can then be modeled parametrically, nonparametrically or in a Bayesian way (Pourahmadi, 1999, page 680).

Recently, Wu and Pourahmadi (2003) proposed use of nonparametric smoothing to regularize estimation of large covariance matrix and developed a two-step estimation procedure. Specifically, they first derive a raw estimate of the covariance matrix and the estimated covariance matrix is subject to the modified Cholesky decomposition. Local polynomial smoothing (Fan and Gijbles, 1996) is then applied to the diagonal elements of D and the subdiagonals of T. Their proposed estimate is guaranteed to be positive-definite. However, their two-step method did not utilize the information that many of the subdiagonals of T are essentially zeros at the first step. Inefficient estimation may result because of ignoring regularization structure in constructing the raw estimator. Furthermore, no method has been suggested to handle missing data in their procedure.

In this paper, we propose a more direct approach of smoothing. We model the main diagonal of D and the subdiagonals of T by spline functions and then employ the maximum likelihood estimation. There is no need to construct a raw estimator in our approach and our estimator is statistically more efficient. For example, in our simulation study reported in Section 5, the reduction of risk by using our direct approach over the two-step procedure is substantial, the reduction is more than 50% (see Section 5) for most cases considered. Since we use the framework of maximal likelihood, EM algorithm can be developed to handle missing data. Moreover, smooth estimate of the mean can be integrated into our method naturally for simultaneous (nonparametric) estimation of the mean and covariance matrix of longitudinal data. The simultaneous maximum likelihood estimation methodology is similar to Pourahmadi (2000) but is hard to extend to local polynomial smoothing.

In Section 2, we discuss the maximum likelihood of covariance matrix using the parametrization based on the modified Cholesky decomposition. The proposed spline smoothing method with implementation details is given in Section 3. Section 4 developed the EM algorithm for dealing with missing data. We illustrate the proposed procedure via simulation in Section 5 and then apply it to the cattle and call center data examples in Section 6. Some discussions are given in Section 7.

2 THE CHOLESKY DECOMPOSITION AND THE MLE

Let $\mathbf{y} = (y_1, \dots, y_m)^T$ be a random vector with mean 0 and variance-covariance matrix Σ. We can think of $\mathbf{y}$ as the time-ordered observations of one subject in a longitudinal study.

The modified Cholesky decomposition (1) provides an unconstrained reparametrization of Σ. The matrices T and D in the decomposition have nice statistical interpretation (Pourahmadi, 1999). Specifically, for $t = 2, \ldots, m$, regress y_t on its predecessors $y_1, \ldots, y_{t-1}$, that is

$$y_t = \sum_{j=1}^{t-1} \phi_{tj} y_j + \epsilon_t \tag{2}$$

with $d_t^2 = \mathrm{var}(\epsilon_t)$. Then $-\phi_{tj}$ is the (t, j)th entry of T for $j < t$, and $\sigma_t^2 = \mathrm{Var}(\epsilon_t)$ is the t-th diagonal element of D. Pourahmadi called $\{\phi_{tj}, j = 1, \ldots, t-1, t = 2, \ldots, m\}$ the generalized autoregressive parameters (GARP) and $\{d_t^2, t = 1, \ldots, m\}$ the innovation variances. Note that the interpretation of the elements of T and D relies on the order among the variables $\{y_t, t = 1, \ldots, m\}$ which is natural for longitudinal data.

Maximal likelihood estimator (MLE) of Σ is highly unstable when the dimension of $\mathbf{y}$ is high. Regularization is thus necessary to improve the MLE. An attractive feature of the decomposition (1) is that unlike the entries of Σ, the subdiagonal entries of T and the log of the "innovation variance" parameters $\log(d_t^2), t = 1, \ldots, m$, are not constrained. It is relative easy to impose structures on the unconstrained parameters and thereby achieve regularization. Before we introduce smoothing based regularization, we first discuss the calculation of the MLE without regularization for the parametrization based on (1).

Assume that $\mathbf{y}$ has a multivariate normal distribution with mean zero and covariance matrix Σ. Let $\mathbf{y}_1, \ldots, \mathbf{y}_n$ be an i.i.d. sample from the distribution of $\mathbf{y}$. Denote $S = (1/n)\sum_{i=1}^n \mathbf{y}_i \mathbf{y}_i^T$. By using the modified Cholesky decomposition (1), the log likelihood of $\mathbf{y}_1, \ldots, \mathbf{y}_n$ can be written as

$$\begin{aligned} l(\Sigma) &= -\frac{n}{2}\log|\Sigma| - \frac{n}{2}\mathrm{tr}(\Sigma^{-1}S) \\ &= -\frac{n}{2}\sum_{t=1}^m \log(d_t^2) - \frac{n}{2}\mathrm{tr}(TST^T D^{-1}) \end{aligned} \tag{3}$$

up to a constant that can be neglected. Denote $G = TST^T$, with g_{jk} as the (j, k)-th element. Then the log likelihood (3) becomes

$$l(\Sigma) = -\frac{n}{2}\sum_{t=1}^m \Big(\log d_t^2 + \frac{g_{tt}}{d_t^2}\Big). \tag{4}$$

Note that $g_{tt} = (1/n)\sum_{i=1}^n (y_{it} - \sum_{j=1}^{t-1} \phi_{tj} y_{ij})^2$. Thus the MLE of ϕ_{tj} does not depend on the MLE of d_t and can be obtained using the method of least squares. Given the MLE of ϕ_{tj}'s, the MLE of d_t^2 is $\hat{d}_t^2 = (1/n)\sum_{i=1}^n (y_{it} - \sum_{j=1}^{t-1} \phi_{tj} y_{ij})^2$.

It is easy to extend the log likelihood to incorporate covariates. Suppose $\mathbf{y}$ has mean $\mathbf{X}\boldsymbol{\alpha}$ where $\mathbf{X}$ is the m by q design matrix which denotes the possible covariates linearly related to the mean of $\mathbf{y}$ and $\boldsymbol{\alpha}$ is a q by 1 vector. Let $(\mathbf{X}_i, \mathbf{y}_i), i = 1, \ldots, n$ be an i.i.d. sample from the joint distribution of $(\mathbf{X}, \mathbf{y})$. Then the log likelihood has the form (3) and (4) with S being replaced by $\hat{S} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\alpha})(\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\alpha})^T$. The maximization of the log likelihood can proceed by iterating between maximizing over $\boldsymbol{\alpha}$ and Σ. More specifically, denote the current estimate of Σ as $\hat{\Sigma}$, then $\boldsymbol{\alpha}$ can be estimated by maximizing

$$-\frac{1}{2}\,\mathrm{tr}\biggl(\hat{\Sigma}^{-1}\sum_{i=1}^{n}(\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\alpha})(\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\alpha})'\biggr)$$

with respect to $\boldsymbol{\alpha}$, which results in the generalized least squares estimate $\hat{\boldsymbol{\alpha}}$. With the current estimate $\hat{\boldsymbol{\alpha}}$, Σ can be estimated by maximizing

$$-\frac{n}{2}\log|\Sigma| - \frac{n}{2}\mathrm{tr}(\Sigma^{-1}\hat{S}) \tag{5}$$

with respect to Σ with $\hat{S} = \frac{1}{n}\sum_{i=1}^{n}[(\mathbf{y}_i - \mathbf{X}_i\hat{\boldsymbol{\alpha}})(\mathbf{y}_i - \mathbf{X}_i\hat{\boldsymbol{\alpha}})^T]$. The similarity of (5) and (3) suggests that the same algorithm can be used to solve both maximization problems.

3 REGULARIZATION BY SPLINE SMOTHING

We introduce regularization to the MLE by applying spline smoothing to the diagonal elements of D and subdiagonals of T in the modified Cholesky decomposition of Σ. This section gives detailed description of our method and discusses computation issues.

3.1 SPLINE SMOOTHING

For simplicity of presentation, we will focus on covariance matrix estimation and assume that $\mathbf{y}$ has mean 0 in our discussion below. Extension to the case that the mean of $\mathbf{y}$ depends on time and other covariates is discussed at the end of this subsection.

We model the diagonal elements of D as realizations of some smooth function f_0,

$$\log(d_t^2) = f_0\biggl(\frac{t}{m+1}\biggr), \qquad t = 1, \ldots, m.$$

As in nonparametric smoothing, we observe $f_0(\cdot)$ on a finer grid as m gets larger. While there are many ways to do nonparametric smoothing, here we adopt the approach based on basis expansions. Specifically, we approximate $f_0(u)$ as a spline function which in turn can be represented by a basis expansion

$$f_0(u) = \sum_{j=1}^{J_0} B_{0j}(u)\beta_{0j} = \mathbf{B}_0^T(u)\boldsymbol{\beta}_0$$

where $\mathbf{B}_0(u) = (B_{01}(u), \ldots, B_{0J_0}(u))^T$ are the B-spline basis functions (de Boor, 2001), and $\boldsymbol{\beta}_0 = (\beta_{01}, \ldots, \beta_{0J_0})^T$ are unknown spline coefficients to be estimated. In principle, it is the linear space spanned by the B-splines that uniquely defines the estimate and any other basis, for example the truncated power basis, can be used in the above expansion. We prefer to use B-spline basis because of its numerical stability. The number of spline coefficients J_0 is determined by the degree of the splines and the number of knots, the choices of which will be discussed later in Section 3.4.

Introducing smoothness for T is not as straightforward as that for D. There are options of smoothing the rows, columns and subdiagonals of T viewed as realizations of some smooth univariate functions. There is also the possibility of smoothing the lower half of T viewed as realizations of a smooth bivariate function. For many applications, it is more relevant to smooth T along its subdiagonals, since its jth subdiagonal entries stand for lag-j regression coefficients over time and relate to time-varying autoregressive models (Wu and Pourahmadi, 2003). Thus, we model the kth ($k = 1, \ldots, m-1$) subdiagonal of T as realizations of some smooth function f_k,

$$\phi_{t,t-k} = f_k\left(\frac{t-k}{m-k+1}\right), \qquad t = k+1, \ldots, m.$$

Here we map the $m-k$ numbers on the kth subdiagonal of T to the function f_k evaluated at $m-k$ equally spaced grid points on the $[0,1]$ interval. As in smoothing D, we approximate each of these smooth functions as a spline function

$$f_k(u) = \sum_{j=1}^{J_k} B_{kj}(u)\beta_{kj} = \mathbf{B}_k^T(u)\boldsymbol{\beta}_k$$

where $\mathbf{B}_k(u) = (B_{k1}(u), \ldots, B_{kJ_k}(u))^T$ are B-spline basis functions which might be different from $\mathbf{B}_0(u)$, $\boldsymbol{\beta}_k = (\beta_{k1}, \ldots, \beta_{kJ_k})^T$ are the spline coefficients, and J_k denotes the number of B-spline basis functions.

The last subdiagonals of T are shorter, so it is unlikely to get a reliable smooth estimate of them. One needs to choose the number of subdiagonals to smooth. Since $\phi_{t,t-j}$ is the lag-j

regression coefficient (see equation (2)), one expects it to be small for a fixed t and large j and the sequence $\phi_{t,t-j}$ for $j = 1, \ldots, t-1$ is expected to be monotone decreasing (Pourahmadi, 1999). This is especially true for ante-dependence models AD(r) (Gabriel, 1962), where $\phi_{j,j-s}$ is zero when $s > r$. We thus decide to smooth only the first m_0 subdiagonals of T for some nonnegative integer m_0 and set all the elements of the last $m - 1 - m_0$ subdiagonals of T to be 0. The same strategy has been adopted by Wu and Pourahmadi (2003). The parameter m_0 is to be determined by the data and its selection is discussed in Section 3.4.

The above formulation has been rather flexible, where we allow different number of basis functions for different subdiagonals of T and even the spline bases used for fitting $f_k(u), k = 1, \ldots, m_0$, can differ. For simplicity in implementation, we take $J_1 = \cdots = J_{m_0} = J$ and $\mathbf{B}_1 = \cdots = \mathbf{B}_{m_0} = \mathbf{B}$. Then we have

$$\log(d_t^2) = \boldsymbol{\beta}_0^T \mathbf{B}_0\left(\frac{t}{m+1}\right), \qquad t = 1, \ldots, m; \tag{6}$$

$$\begin{aligned} \phi_{tj} = \boldsymbol{\beta}_{t-j}^T \mathbf{B}\left(\frac{j}{m-(t-j)+1}\right), \qquad t = 2, \ldots, m, \\ \max\{1, t-m_0\} \le j \le t-1; \\ \phi_{tj} = 0, \qquad t = 2, \ldots, m,\ j < t - m_0. \end{aligned} \tag{7}$$

Such simplification works well in the simulation study and real data examples.

Note that (6) and (7) impose some restrictions on D and T and thereby introduce regularization in maximal likelihood estimation of the covariance matrix. As we will see in our simulation studies, such regularization is useful, especially when the dimension of the covariance matrix is high. In the next two subsections, we develop two algorithms for calculating the MLE under models (6) and (7). For ease of presentation in discussion to follow, we denote $\boldsymbol{\beta}_k \equiv 0$ for all $k > m_0$ and assume (7) holds for $t = 2, \ldots, m$, $j = 1, \ldots, t-1$. The values of $\boldsymbol{\beta}_k, k > m_0$ are not updated in our iterative estimation procedures.

Remark. When the mean $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_m)^T$ of $\mathbf{y}$ is not zero and can be viewed as a discretization of a continuous function $\mu(\cdot)$ of time t, we can smooth the mean just as we smoothed the diagonal elements of D. Specifically, we represent $\mu(\cdot)$ by a linear combination of B-splines. As a result, $\boldsymbol{\mu}$ can be represented as $\mathbf{X}\boldsymbol{\alpha}$ for appropriately defined $\mathbf{X}$ involving B-splines. The discussion at the end of Section 2 then applies to handle the estimation of $\boldsymbol{\alpha}$. If, in addition to the time effect, there are other covariates that influence the mean, we can add more columns to $\mathbf{X}$ and the same fitting procedure applies.

3.2 ALGORITHM 1

The first algorithm for computing the proposed estimator is based on alternating optimization over $\boldsymbol{\beta}_0$ and $\boldsymbol{\beta}$. The dependence of the log likelihood function (4) on $\boldsymbol{\beta}_0$ is through an exponential function. Given the current values of $\boldsymbol{\beta}_0$ and $\boldsymbol{\beta} = (\boldsymbol{\beta}_1^T, \ldots, \boldsymbol{\beta}_{m_0}^T)^T$, we can update the estimate of $\boldsymbol{\beta}_0$ using one step of the Newton-Raphson method:

$$\boldsymbol{\beta}_0 \leftarrow \boldsymbol{\beta}_0 - \{H_{\boldsymbol{\beta}_0,\boldsymbol{\beta}_0}\}^{-1} S_{\boldsymbol{\beta}_0},$$

where the Score $S_{\boldsymbol{\beta}_0}$ and Hessian $H_{\boldsymbol{\beta}_0,\boldsymbol{\beta}_0}$ are given in the Appendix. For fixed $\boldsymbol{\beta}_0$, the log likelihood (4) as a function of $\boldsymbol{\beta}$ is a quadratic form in $\boldsymbol{\beta}$, and its maximizer has a closed form expression. Denote the (j,k)-th element of S as s_{jk}. Denote

$$\mathbf{z}_{tj} = \Bigg(\underbrace{0,\ldots,0}_{(t-j-1)J}, B_1\left(\frac{j}{m-(t-j)+1}\right), \ldots, B_J\left(\frac{j}{m-(t-j)+1}\right), \underbrace{0,\ldots,0}_{\{m-1-(t-j)\}J}\Bigg)^T,$$

for $j = 1, \ldots, t-1, t = 2, \ldots, m$, then $\phi_{tj} = \mathbf{z}_{tj}^T \boldsymbol{\beta}$. In the Appendix, we will show that $\boldsymbol{\beta} = A^{-1}\mathbf{b}$ with

$$A = \sum_{t=2}^{m} \exp\left\{-\boldsymbol{\beta}_0^T \mathbf{B}_0\left(\frac{t}{m+1}\right)\right\} \sum_{j=1}^{t-1}\sum_{k=1}^{t-1} s_{jk}(\mathbf{z}_{tj}\mathbf{z}_{tk}^T + \mathbf{z}_{tk}\mathbf{z}_{tj}^T) \tag{8}$$

and

$$\mathbf{b} = 2\sum_{t=2}^{m} \exp\left\{-\boldsymbol{\beta}_0^T \mathbf{B}_0\left(\frac{t}{m+1}\right)\right\} \sum_{k=1}^{t-1} s_{tk}\mathbf{z}_{tk}. \tag{9}$$

We summarize the above optimization procedure in Algorithm 1.

3.3 ALGORITHM 2

Note that the above procedure requires inverting the matrix A whose dimension Jm_0 can be large in practice. The matrix A may be ill-conditioned to prevent a stable calculation of its

Algorithm 1

0. Choose an initial value for $\boldsymbol{\beta}_0^{(0)}$ and $\boldsymbol{\beta}^{(0)}$.
1. Given the current values $\boldsymbol{\beta}_0^{(s)}$ and $\boldsymbol{\beta}^{(s)}$, update $\boldsymbol{\beta}_0^{(s+1)}$ according to one Newton-Raphson step
$$\boldsymbol{\beta}_0^{(s+1)} = \boldsymbol{\beta}_0^{(s)} - \{H_{\boldsymbol{\beta}_0^{(s)},\boldsymbol{\beta}_0^{(s)}}\}^{-1} S_{\boldsymbol{\beta}_0^{(s)}}.$$
2. Given the current values $\boldsymbol{\beta}_0^{(s)}$, obtain the updated estimate of $\boldsymbol{\beta}$ as $\boldsymbol{\beta}^{(s+1)} = A^{-1}\mathbf{b}$ with A and $\mathbf{b}$ defined in equations (8) and (9), where the parameters on the right-hand of the equations are fixed at their current values.
3. Repeat Steps 1 and 2 until some convergence criterion is met.

inverse. To avoid inverting the matrix A, we propose an alternative procedure in which we update the spline coefficients $\boldsymbol{\beta}_k$ for $k = 0, 1, \ldots, m_0$ sequentially. Given the current value of $\boldsymbol{\beta}_0$, $\boldsymbol{\beta} = (\boldsymbol{\beta}_1^T, \ldots, \boldsymbol{\beta}_{m_0}^T)^T$, update of $\boldsymbol{\beta}_0$ can be done as in Algorithm 1. Update of $\boldsymbol{\beta}$ is done differently from Algorithm 1. Fixing the rest of the parameters, the log likelihood is a quadratic form in $\boldsymbol{\beta}_j$, its maximizer has a closed form expression $\boldsymbol{\beta}_j = A_j^{-1}\mathbf{b}_j$, where

$$\begin{aligned} A_j = {} & \mathbf{B}\left(\frac{t-j}{m-j+1}\right)\mathbf{B}^T\left(\frac{t-j}{m-j+1}\right) \\ & \sum_{t=j+1}^{m} \exp\left\{-\boldsymbol{\beta}_0^T\mathbf{B}_0\left(\frac{t}{m+1}\right)\right\} s_{t-j,t-j} \end{aligned} \tag{10}$$

and

$$\begin{aligned} \mathbf{b}_j = {} & -\frac{1}{2}\mathbf{B}\left(\frac{t-j}{m-j+1}\right) \times \left[\sum_{k\neq j}^{m-1} \boldsymbol{\beta}_k^T\mathbf{B}\left(\frac{t-k}{m-k+1}\right)\right. \\ & \left.\sum_{t=(k+1)\vee(j+1)}^{m} \exp\left\{-\boldsymbol{\beta}_0^T\mathbf{B}_0\left(\frac{t}{m+1}\right)\right\} s_{t-j,t-k}\right] \\ & + \mathbf{B}\left(\frac{t-j}{m-j+1}\right) \sum_{t=j+1}^{m} \exp\left\{-\boldsymbol{\beta}_0^T\mathbf{B}_0\left(\frac{t}{m+1}\right)\right\} s_{t,t-j}. \end{aligned} \tag{11}$$

See Appendix for details.

The second procedure is summarized in Algorithm 2.

Note as we discussed before, only $\boldsymbol{\beta}_j$ for $1 \leq j \leq m_0$ need to be updated and the rest of $\boldsymbol{\beta}$ are fixed at 0. One advantage of Algorithm 2 is that in each of the updating step, the

Algorithm 2

```
0. Choose an initial value for $\boldsymbol{\beta}_0^{(0)}$ and $\boldsymbol{\beta}^{(0)}$.
1. Given the current values $\boldsymbol{\beta}_0^{(s)}$ and $\boldsymbol{\beta}^{(s)}$, update $\boldsymbol{\beta}_0^{(s+1)}$ according to one Newton-Raphson step
       $\boldsymbol{\beta}_0^{(s+1)} = \boldsymbol{\beta}_0^{(s)} - \{H_{\boldsymbol{\beta}_0^{(s)},\boldsymbol{\beta}_0^{(s)}}\}^{-1} S_{\boldsymbol{\beta}_0^{(s)}}.$
2. Given the current values $\boldsymbol{\beta}_0^{(s)}$ and $\boldsymbol{\beta}_k^{(s)}$ for $k = 1, \ldots, m_0$ and $k \neq j$, obtain the updated estimate of $\boldsymbol{\beta}_j$ as $\boldsymbol{\beta}_j^{(s+1)} = A_j^{-1}\mathbf{b}_j$ for $j = 1, \ldots, m_0$ with $A_j$ and $\mathbf{b}_j$ defined as in equations (10) and (11). The parameters on the right-hand side of the equations are fixed at their current values.
3. Repeat steps 1 and 2 until some convergence criterion is met.
```

dimension of A_j is J which can be substantially smaller than the dimension of A when the number of non-zero subdiagonals of T is moderate to large. However, in Algorithm 1, we update $\boldsymbol{\beta}$ as a whole vector and iteration is not needed unlike in Algorithm 2.

3.4 CHOICE OF TUNING PARAMETERS

Our method only involves splines on the interval [0,1]. The knots of the splines are evenly placed over this interval. The numbers of spline coefficients J_0 and J are determined by the degree of splines and the number of knots. We used quadratic splines in our implementation and found they work well. Certainly, other degree splines such as cubic splines can be used. The number of knots together with the tuning parameter m_0 are determined based on the BIC criterion, which is defined as

$$\mathrm{BIC} = -\frac{2}{n} l(\boldsymbol{\beta}_0, \boldsymbol{\beta}) + \frac{\log n}{n}(J_0 + m_0 J). \tag{12}$$

We found that searching the number of knots in the range of 3 to 10 is usually sufficient in our simulation examples (Section 5). We explored a wider range in the call center data (Section 6.2) . Note that the m_0-th sub-diagonal has only $m - m_0$ elements so the number of basis functions J should not exceed $m - m_0$. This will not cause a problem in general when m is large and m_0 is relatively small. Crossvalidated likelihood can also be used to select the tuning parameters (Huang et al., 2006, Section 4.2). Another method for selecting tuning parameters based on prediction performance on a test dataset is illustrated in Section 6.2.

4 INCOMPLETE DATA AND THE EM ALGORITHM

Missing data arise frequently in practice especially in the longitudinal settings. One of the most often used strategies to deal with the missing data problem is the EM algorithms (Dempster, Laird and Rubin, 1977). Since we are using the likelihood framework, the EM algorithm can be derived to compute our estimators when missing data exist.

Assume $\mathbf{y}_i \sim N(\mathbf{X}_i\boldsymbol{\alpha}, \Sigma)$, $i = 1, \ldots, n$, independently and we want to estimate $\boldsymbol{\alpha}$ and Σ in the presence of missing values in $\mathbf{y}_i$. Let $\mathbf{y}_i = (\mathbf{y}_{i,obs}^T, \mathbf{y}_{i,mis}^T)^T$, $i = 1, \ldots, n$, be the complete data where $\mathbf{y}_{i,obs}$ and $\mathbf{y}_{i,mis}$ are the observed and missing data in $\mathbf{y}_i$. Suppose $\mathbf{X}_i$ are the corresponding covariates without missing values. For example, $\mathbf{X}_i$ are obtained from B-spline basis functions of observation times as discussed in the Remark at the end of Section 3.1. Denote $\mathbf{y}_{obs} = \{\mathbf{y}_{1,obs}, \ldots, \mathbf{y}_{n,obs}\}$ and $\mathbf{y}_{mis} = \{\mathbf{y}_{1,mis}, \ldots, \mathbf{y}_{n,mis}\}$. Here we assume that $\Theta = (\boldsymbol{\alpha}, \Sigma)$, the log likelihood (3) can be written as $l(\Theta) = l(\Theta; \mathbf{y}_{obs}, \mathbf{y}_{mis})$. The EM algorithm that iterates between the E-step and M-step proceeds as the following.

E-step:

Recall that the complete data log likelihood is

$$l(\Theta; \mathbf{y}_1, \ldots, \mathbf{y}_n) = -\frac{n}{2}\log|\Sigma| - \frac{1}{2}\mathrm{tr}\left\{\Sigma^{-1}\sum_{i=1}^{n}(\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\alpha})(\mathbf{y}_i - \mathbf{X}_i\boldsymbol{\alpha})^T\right\}.$$

The expectation of the complete data log likelihood, given the observed data and the current estimate of the parameters $\Theta^{(c)}$ is

$$\begin{aligned} & E[l(\Theta; \mathbf{y}_{obs}, \mathbf{y}_{mis})|\mathbf{y}_{obs}, \Theta^{(c)}] \\ &= -\frac{n}{2}\log|\Sigma| - \frac{1}{2}\mathrm{tr}\Big[\Sigma^{-1}\sum_{i=1}^{n}\{E(\mathbf{y}_i\mathbf{y}_i^T|\mathbf{y}_{i,obs}, \Theta^{(c)}) \\ &\qquad - 2E(\mathbf{y}_i|\mathbf{y}_{i,obs}, \Theta^{(c)})\boldsymbol{\alpha}^T\mathbf{X}_i^T + \mathbf{X}_i\boldsymbol{\alpha}\boldsymbol{\alpha}^T\mathbf{X}_i^T\}\Big]. \end{aligned} \tag{13}$$

Note that

$$\mathbf{y}_i\mathbf{y}_i^T = \begin{pmatrix} \mathbf{y}_{i,obs}\mathbf{y}_{i,obs}^T & \mathbf{y}_{i,obs}y_{i,mis}^T \\ \mathbf{y}_{i,mis}\mathbf{y}_{i,obs}^T & \mathbf{y}_{i,mis}\mathbf{y}_{i,mis}^T \end{pmatrix}.$$

Thus we need only calculate $E(\mathbf{y}_{i,mis}|\mathbf{y}_{i,obs},\Theta^{(c)})$ and $E(\mathbf{y}_{i,mis}{\mathbf{y}_{i,mis}}^T|\mathbf{y}_{i,obs},\Theta^{(c)})$. Let $\mathbf{X}_i = (\mathbf{X}_{i,obs}^T, \mathbf{X}_{i,mis}^T)^T$ and

$$\Sigma = \begin{pmatrix} \Sigma_{i,obs} & \Sigma_{i,om} \\ \Sigma_{i,mo} & \Sigma_{i,mis} \end{pmatrix},$$

where the partition of $\mathbf{X}_i$ and Σ correspond to the partition of $\mathbf{y}_i$. By the properties of multivariate normal distribution, we have that

$$E(\mathbf{y}_{i,mis}|\mathbf{y}_{i,obs},\Theta^{(c)}) = \mathbf{X}_{i,mis}\boldsymbol{\alpha}^{(c)} + \Sigma_{i,mo}^{(c)}\Sigma_{i,obs}^{(c)^{-1}}(\mathbf{y}_{i,obs} - \mathbf{X}_{i,obs}\boldsymbol{\alpha}^{(c)})$$

and

$$\begin{aligned} E(\mathbf{y}_{i,mis}{\mathbf{y}_{i,mis}}^T|\mathbf{y}_{i,obs},\Theta^{(c)}) &= \mathrm{Var}(\mathbf{y}_{i,mis}|\mathbf{y}_{i,obs},\Theta^{(c)}) \\ &\quad + E(\mathbf{y}_{i,mis}|\mathbf{y}_{i,obs},\Theta^{(c)})E(\mathbf{y}_{i,mis}|\mathbf{y}_{i,obs},\Theta^{(c)})^T \end{aligned}$$

where $\mathrm{Var}(\mathbf{y}_{i,mis}|\mathbf{y}_{i,obs},\Theta^{(c)}) = \Sigma_{i,mis}^{(c)} - \Sigma_{i,mo}^{(c)}\Sigma_{i,obs}^{(c)^{-1}}\Sigma_{i,om}^{(c)}$. This completes the E-step.

M-step:

The maximization step can proceed as in the case with no missing data, that is, by iterating between maximizing over $\boldsymbol{\alpha}$ and Σ as discussed at the end of Section 2. Note that (13) has the same form as (3) with S replaced by

$$\frac{1}{n}\sum_{i=1}^{n}\{E(\mathbf{y}_i\mathbf{y}_i^T - 2\mathbf{y}_i\boldsymbol{\alpha}^T\mathbf{X}_i^T|\mathbf{y}_{i,obs},\Theta^{(c)}) + \mathbf{X}_i\boldsymbol{\alpha}\boldsymbol{\alpha}^T\mathbf{X}_i^T\}.$$

Thus both Algorithms 1 and 2 can be used to optimize (13) to get the updated value of Σ. This completes the description of the M-step.

The above EM algorithm requires that the covariate vectors $\mathbf{X}_i$ have complete observations. When $\mathbf{X}_i$ have missing values with the same missing pattern as $\mathbf{y}_i$, we can move the update of $\boldsymbol{\alpha}$ outside the EM loop and update $\boldsymbol{\alpha}$ with the generalized least squares using only observed data. This way avoids imputation of missing values in $\mathbf{X}_i$.

5 SIMULATIONS

In this section, we compare by simulation the performance of our spline-based covariance matrix estimator and the local polynomial-based estimator of Wu and Pourahmadi (2003).

We also include the sample covariance matrix in the comparison. We restrict our attention to the case without missing data since the method of Wu and Pourahmadi cannot deal with missing data directly.

To gauge the performance of the covariance matrix estimators, we consider the following two widely used loss functions:

$$\Delta_1(\Sigma, G) = \mathrm{tr}(\Sigma^{-1}G) - \log|\Sigma^{-1}G| - m$$

and

$$\Delta_2(\Sigma, G) = \mathrm{tr}(\Sigma^{-1}G - I)^2,$$

where Σ is the true covariance matrix and G is a positive definite matrix. The first loss is typically called the entropy loss and the second one the quadratic loss. Each of these losses is 0 when $G = \Sigma$ and is positive when $G \neq \Sigma$. Both loss functions are invariant with respect to transformations $G^* = CGC^T$, $\Sigma^* = C\Sigma C^T$ for nonsingular matrix C (Andersen, 2003). The corresponding risk functions are defined as

$$R_i(\Sigma, G) = E_\Sigma\{\Delta_i(\Sigma, G)\}, \qquad i = 1, 2.$$

An estimator $\hat{\Sigma}_1$ is considered better than another estimator $\hat{\Sigma}_2$ if its risk function is smaller, that is, $R_i(\Sigma, \hat{\Sigma}_1) < R_i(\Sigma, \hat{\Sigma}_2)$. The risk function of an estimator is approximated by Monte Carlo simulation. To produce the results presented below, $N = 100$ simulation runs are used for each setup.

We consider the following three covariance matrices that have been considered in Wu and Pourmahmadi (2003):

- Σ_1: $\phi_{t,t-j} \equiv 0$, $\sigma_t^2 \equiv 1$, corresponding to the identity covariance matrix;

- Σ_2: $\phi_{t,t-1} = 2(t/m)^2 - 0.5$, $\phi_{t,t-j} \equiv 0, j \geq 2$, $\sigma_t = \log(t/10 + 2)$, corresponding to varying coefficient AR(1);

- Σ_3: $\phi_{t,t-j} = m^{-2}\min\{t + j, t^{1.5}\}\exp\{-j/4\}$, $\sigma_t = \log(t/10 + 2)$.

For each Σ from the above list, we simulate n i.i.d. $N_m(0, \Sigma)$ random vectors for different combinations of n and m. For each of the above two loss functions, the values of $\Delta_i(\Sigma, S)$ and $\Delta_i(\Sigma, \hat{\Sigma})$ are evaluated and the corresponding risks are obtained by averaging the values of the losses across the 100 simulation runs.

The risk functions for the sample covariance matrix, the two-step local polynomial smoothed estimator of Wu and Pourahmadi (2003) and the proposed spline smoothed estimator for the three different covariance matrices are presented in Tables 1 to 3 respectively. The BIC is used to select the tuning parameters for the spline estimator. The results for the local polynomial smoothed estimator for the entropy loss are reproduced from Wu and Pourahmadi (2003) and those for the quadratic loss are reproduced from an earlier version of Wu and Pourahmadi (2003). Based on the estimated risks as shown in these tables, it is evident that both smoothed covariance estimators outperform the sample covariance matrix for every combination of (Σ, n, m) under both loss functions. The improvement is substantial especially when m is large. The spline smoothed covariance estimators also significantly improve over the two-step local polynomial smoothed covariance estimators of Wu and Pourahmadi (2003); for all cases with $m \geq 20$ except for Σ_3 with $n = 100$ and $m = 30$, the reductions of risk are all bigger than 50% for both the entropy loss and the quadratic loss. We think the reason that the two-step method does not work as well as our method is that the first step raw estimator is too noisy.

Table 1: Simulations results for Σ_1. Risks functions for the sample covariance matrix, local polynomial smoothed estimator and the spline smoothed estimator. The results are based on 100 simulation runs.

			Entropy loss			Quadratic loss		
	n	m	Sample	Local	Spline	Sample	Local	Spline
Σ_1	50	10	1.215	0.246	0.145	2.281	0.648	0.308
		20	5.023	0.384	0.164	8.679	1.036	0.341
		30	12.410	0.399	0.159	19.050	1.101	0.328
		40	25.326	0.448	0.139	33.606	1.151	0.284
	100	10	0.575	0.118	0.078	1.127	0.291	0.159
		20	2.290	0.178	0.076	4.263	0.456	0.154
		30	5.237	0.194	0.085	9.311	0.517	0.171
		40	9.647	0.216	0.091	16.500	0.661	0.184

Table 2: Simulations results for Σ_2. Risks functions for the sample covariance matrix, local polynomial smoothed estimator and the spline smoothed estimator. The results are based on 100 simulation runs.

			Entropy loss			Quadratic loss		
	n	m	Sample	Local	Spline	Sample	Local	Spline
Σ_2	50	10	1.198	0.274	0.152	2.252	0.809	0.316
		20	5.039	0.439	0.140	8.629	1.150	0.287
		30	12.472	0.462	0.148	19.045	1.244	0.303
		40	25.672	0.506	0.169	33.616	1.316	0.346
	100	10	0.562	0.159	0.079	1.099	0.501	0.162
		20	2.269	0.213	0.071	4.242	0.609	0.143
		30	5.265	0.242	0.088	9.452	0.703	0.177
		40	9.685	0.271	0.098	16.569	0.764	0.197

Table 3: Simulations results for Σ_3. Risks functions for the sample covariance matrix, local polynomial smoothed estimator and the spline smoothed estimator. The results are based on 100 simulation runs.

			Entropy loss			Quadratic loss		
	n	m	Sample	Local	Spline	Sample	Local	Spline
Σ_3	50	10	1.220	0.274	0.195	2.264	0.626	0.388
		20	5.059	0.411	0.176	8.700	1.086	0.353
		30	12.493	0.389	0.159	19.164	1.136	0.320
		40	25.740	0.472	0.187	33.861	1.491	0.378
	100	10	0.563	0.154	0.138	1.101	0.448	0.270
		20	2.271	0.206	0.097	4.229	0.690	0.191
		30	5.245	0.196	0.106	9.365	0.663	0.211
		40	9.726	0.202	0.101	16.595	0.744	0.202

6 REAL DATA ANALYSIS

In this section we illustrate the proposed method using two real datasets. The first dataset has been studied extensively in the literature of longitudinal data analysis to gauge the performance of new methods. In the second real data example, estimation of covariance matrix is used in forecasting call arrival pattern to a telephone call center.

6.1 CATTLE DATA

Kenward (1987) reports an experiment in which cattle were assigned randomly to two treatment groups A and B, and their weights were recorded to study the effect of treatments on intestinal parasites. The animals were weighed $m = 11$ times over a 133-day period and the data are balanced. No observation was missing. Of 60 cattle $n = 30$ received treatment A and the other 30 received treatment B. Zimmerman and Núñez-Antón (1997) rejected equality of the two within treatment-group covariance matrices using the classical likelihood ratio test. Thus, it is advisable to study each treatment group's covariance matrix separately; here we report our results for the group A cattle.

Applying the proposed method, we modeled the mean and the covariance matrix of the cattle data simultaneously using quadratic splines. Based on the BIC, we chose to smooth the first two subdiagonals of T and set the rest as zeros. The diagonal of D is smoothed with 4 B-spline basis functions, the two subdiagonals of T are smoothed with 3 basis functions, and the means are fitted with 9 B-spline basis functions. Our results with two non-zero subdiagonals lend more support to an ante-dependence model of order 2 for this data (Macchiavelli and Arnold, 1994). The proposed spline method has the same power as the cubic polynomial based method of Pourahmadi (2000) in suggesting a parsimonious model for Σ. While the unstructured covariance has 66 parameters for the 11×11 covariance matrix, the cubic polynomial model (Pourahmadi, 2000) and the spline model used 8 and 10 parameters, respectively. The BIC for our spline model is 51.61, which can be compared with BIC value of 70.84 for the cubic polynomial model and the BIC value of 75.35 for the unstructured model.

To examine the effect of missing data on our proposed procedure, we randomly removed some proportion of the data, then applied the EM algorithm as described in Section 4. We considered percentage of missing being 5%, 10% and 20%. Figure 1 shows the fitted mean function, the smoothed diagonal of matrix D and the smoothed first two subdiagonals of matrix T with complete data and incomplete data. As indicated in the plot, we can barely tell the difference of the estimates of the mean function. The estimated diagonal of D and the first two subdiagonals of T with missing data follow similar patterns as those without missing data. As the percentage of missing increases, the smoothed curves using incomplete data moves away from those obtained using the complete data.

6.2 TELEPHONE CALL CENTER DATA

Telephone call centers have become an integral part of the operations of many large organizations. With their growing presence and importance in organizations, managing call center operations more efficiently has become an issue of significant economic interest (Brown et al. 2005). In this subsection we consider analysis of data from one call center that belongs to a major US northeastern financial organization. We illustrate our method for estimating large covariance matrices by an application in forecasting the call arrival pattern to a call center. Such forecasts are useful for call center management such as staffing and scheduling.

The original database has the information about the time every call arrives to the service queue. For each day in 2002 (except 6 days where the data collecting equipment went out of order), the records of phone calls start from 7:00AM until midnight. We divided the 17-hour period into 102 10-minute intervals, and counted the number of calls arrived to the service queue during each interval. Here the length of the intervals, 10 minutes, is chosen rather subjectively as a way of smoothing the data and for illustration. Since the arrival patterns of weekdays and weekends differ, we focus on weekdays here. Using the singular value decomposition to screen out outliers that include holidays and recording equipment ill-functioning days (Shen and Huang, 2005), we obtain observations for 239 regular days. Denote the data for day i as $\mathbf{N}_i = (N_{i1}, \ldots, N_{i,102})^T$, $i = 1, \ldots, 239$, where N_{it} is the number of calls arriving to the call center for the t-th 10-minute interval for day i. Let $y_{it} = \sqrt{N_{it} + 1/4}$, $i = 1, \ldots, 239$, $t = 1, \ldots, 102$. The square root transformation is used to make the data distribution close to normal (Brown et al. 2005).

Consider forecasting the number of arrivals later in the day using arrival patterns at earlier times of the day. Write $\mathbf{y}_i = (y_{i1}, \ldots, y_{i,102})^T$. Form the partition $\mathbf{y}_i = (\mathbf{y}_i^{(1)^T}, \mathbf{y}_i^{(2)^T})^T$, where $\mathbf{y}_i^{(1)}$ and $\mathbf{y}_i^{(2)}$ measure the arrival patterns in the early and later times of day i. For example, we can take $\mathbf{y}_i^{(1)} = (y_{i1}, \ldots, y_{i,51})^T$ and $\mathbf{y}_i^{(2)} = (y_{i,52}, \ldots, y_{i,102})^T$, which measure

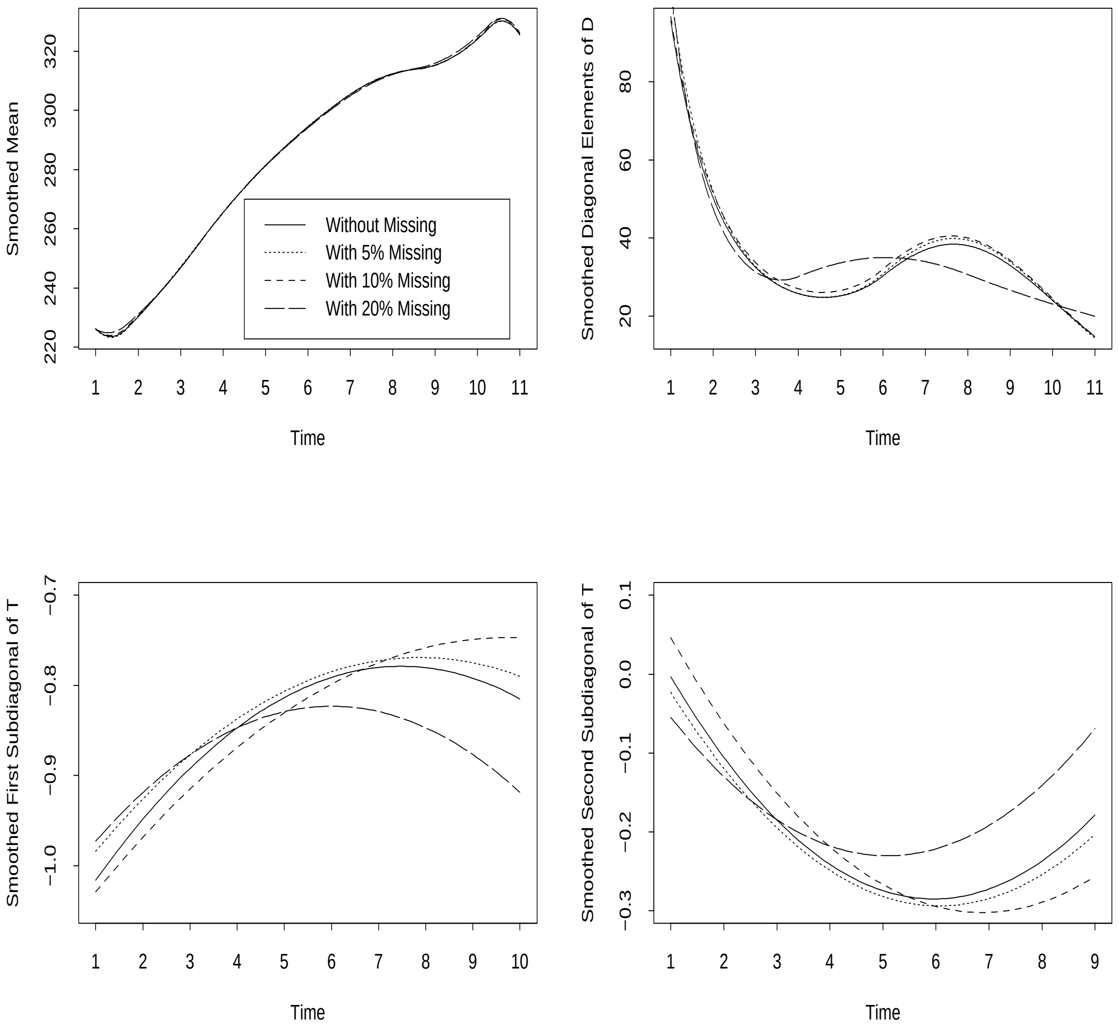

Figure 1: The cattle data. The spline estimate of the mean function, the diagonal of D and the first two subdiagonals of T for the complete and incomplete data. The percentages of missing values for the incomplete data are 5%, 10%, and 20%.

respectively the arrival patterns in the early and later halves of a day. The corresponding partition of the mean and covariance matrix is denoted by

$$\boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{pmatrix}, \ \Sigma = \begin{pmatrix} \Sigma_{11}, \Sigma_{12} \\ \Sigma_{21}, \Sigma_{22} \end{pmatrix}.$$

Assuming multivariate normality, the best mean squared error forecast of $\mathbf{y}_i^{(2)}$ using $\mathbf{y}_i^{(1)}$ is

$$E(\mathbf{y}_i^{(2)}|\mathbf{y}_i^{(1)}) = \boldsymbol{\mu}_2 + \Sigma_{21}\Sigma_{11}^{-1}(\mathbf{y}_i^{(1)} - \boldsymbol{\mu}_1). \tag{14}$$

Without the normality assumption, this formula gives the best mean squared error linear forecast. In practice, we need to plug in estimates of $\boldsymbol{\mu}$ and Σ. We used saturated mean model to estimate the mean $\boldsymbol{\mu}$ and we considered two methods of estimation of Σ, one is the proposed maximal likelihood estimate with smoothness restrictions on the Cholesky factor of Σ, the other is the maximal likelihood without smoothness restrictions. We did not apply spline smoothing to the mean $\boldsymbol{\mu}$ since the estimate from the saturated mean model is already very smooth.

To compare the out-of-sample forecast performance using different methods, we split the 239 days into training and test datasets. The data from the first 222 days, corresponding to January to November, form the training dataset that is used to estimate the mean and covariance structure. The estimates are then applied for forecasting using formula (14) for the 17 days in the test set, corresponding to December. We used the 51 square-root-transformed arrival counts in the early half of a day to forecast the square-root-transformed arrival counts in the later half of the day. For each time interval $t = 52, \ldots, 102$, define the average absolute forecast error by

$$\mathrm{AE}_t = \frac{1}{17} \sum_{i=223}^{239} |\hat{y}_{it} - y_{it}|,$$

where y_{it} and $\hat{y}_{it}$ are the observed and forecast values respectively.

We need to choose the tuning parameters for the proposed method. We used a method that is related to the objective of application. We split the training dataset into two parts – the first part includes the first 205 days, corresponding to January to October, and the second part is from day 206 to day 222 which corresponds to November. The first part in the training dataset is used for parameter estimation and the second part is used for tuning parameter selection. Fix a set of candidate tuning parameters and calculate the estimates of

$\boldsymbol{\mu}$ and Σ using the first part of the training dataset. Based on these estimates, we forecast the arrival pattern in the later half of the day with information from the earlier half of the day using formula (14), for each of the 17 days in the second part of the training dataset. Define the total average absolute forecast error in the training data as

$$\widehat{\mathrm{AE}} = \frac{1}{51 \times 17} \sum_{t=52}^{102} \sum_{i=206}^{222} |\hat{y}_{it} - y_{it}|, \tag{15}$$

where y_{it} and $\hat{y}_{it}$, $i = 206, \ldots, 222$, $t = 52, \ldots, 102$, are the actual and forecast values respectively. This is used as the criterion for selecting the tuning parameters. According to this criterion, we kept the first two subdiagonals of T and set the rest subdiagonals to 0. We also chose to use 15 B-spline basis functions for the diagonal elements of D and 8 basis functions for each of the subdiagonals of T. These tuning parameters were then used when we re-estimated Σ using the whole training dataset. The resulting estimates were employed to generate out-of-sample forecast using formula (14).

In the upper panel of Figure 2, we plot the test dataset AE_t for the forecasts using the spline smoothed estimate and the estimate without smoothing. The benefit of smoothing is obvious. In the lower panel of Figure 2, we plot the percentage of times, among 17 days in the test dataset, on which the forecast based on spline smoothed estimate has smaller AE_t. It shows clearly that the forecast based on the spline smoothed estimate outperforms that without smoothing. Based on AE_t, the former does better in 49 out of the 51 time intervals. The average of the AE_t over the 51 time intervals is 0.844 based on the spline smoothed estimate, whereas it is 1.367 if smoothing is not employed.

7 Discussion

This paper builds on the recent literature on modeling of large covariance matrices using a modified Cholesky factorization. For longitudinal data, the Cholesky factor has an interpretation as a matrix of regression coefficients when y_t is regressed on its predecessors $y_{t-1}, \ldots, y_1$, and the diagonal matrix contains the innovation variances. We regularize the maximum likelihood estimation by smoothing along subdiagonals of the Cholesky factor as well as the logarithm of the innovation variances. Furthermore, we only smooth a selected number of subdiagonals and set the rest, shorter diagonals, as zeros. While there are many ways to do smoothing, we apply basis expansions with B-splines.

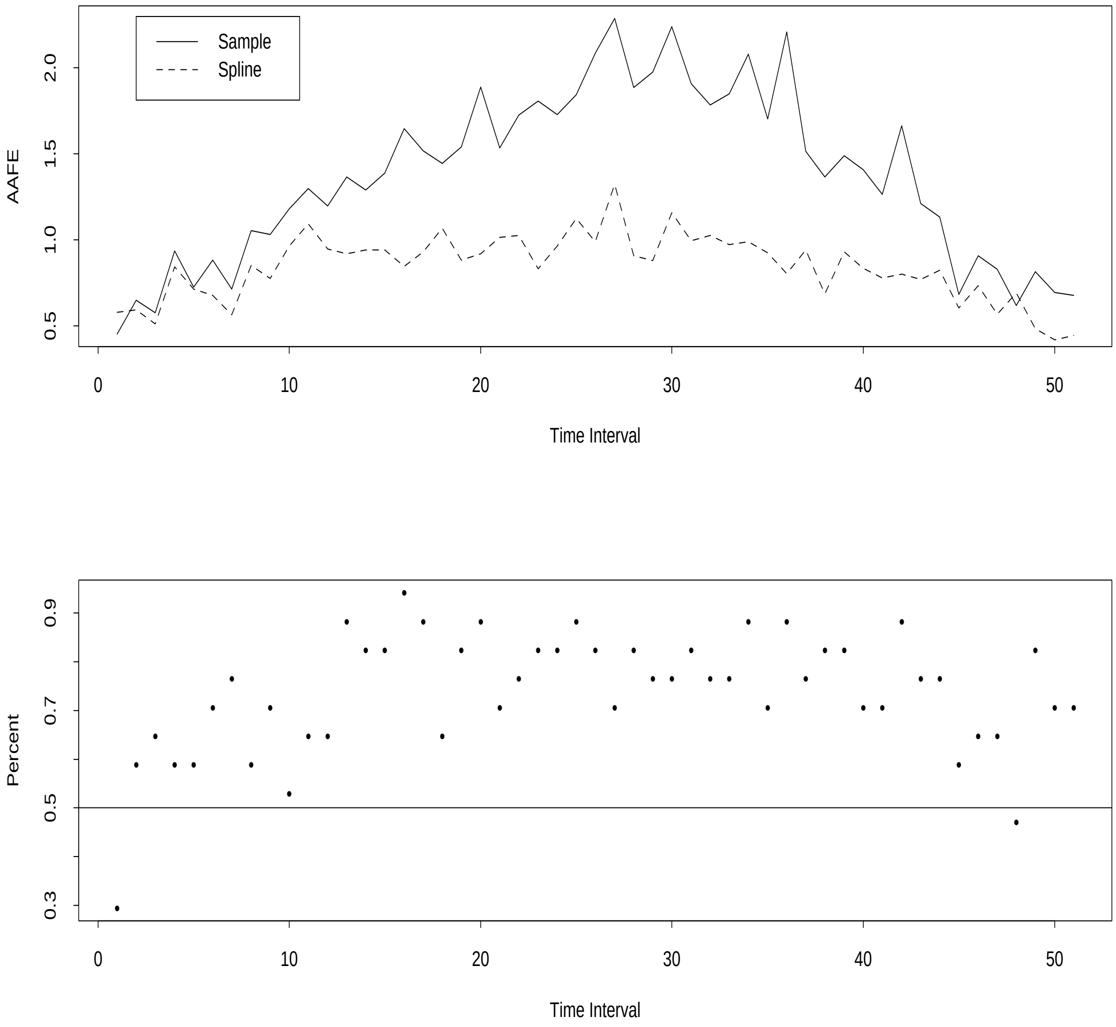

Figure 2: The call center data. The upper panel is the plot of AE_t for the forecasts using the spline smoothed estimate, dashed and the estimate without smoothing, solid. The lower panel is the percentage of times, among 17 days in the test dataset, on which the spline smoothed covariance based forecast has smaller absolute forecast error.

The advantage of the basis expansion approach is that it easily fits in the maximal likelihood framework. As a result, the proposed method is statistically more efficient than the two-step procedure of Wu and Pourahmadi (2003). Adopting the maximal likelihood framework also allows natural handling of missing values that occur frequently in practice. Since our estimator does not rely on a first step raw estimator of the Cholesky factor of a covariance matrix (cf. Wu and Pourahmadi, 2003), it can be computed even when the raw estimator is not well-defined (for example when the dimension is higher than the sample size). Nevertheless, given the small sample size, the reliability and stability of maximum likelihood estimates need to be critically assessed. In this paper, we employed B-spline basis, although other linear basis could just as easily be employed. We focused on fixed knot splines and the method works well. It might be useful to consider data driven method to place knots of splines (Stone et al. 1997).

We refer to our approach as regularized MLE using basis expansion. An alternative approach to regularization is penalized likelihood. Huang et al. (2006) proposed to shrink the Cholesky factor by adding an L_1 or L_2 penalty to the negative log likelihood. They observed that, use of the L_1 penalty is effective if the subdiagonals of the Cholesky factor contain many zeros, while use of the L_2 penalty is effective if the subdiagonals of the Cholesky factor contain many small nonzero values. The L_1 penalty sets some elements in the Cholesky factor to zero, where these zero elements can be irregularly placed. The method in the current paper also sets zero of some elements in the Cholesky factor, but in a quite restrictive manner, the whole subdiagonal is set to zero for all short subdiagonals.

Within the framework of penalized likelihood, roughness penalties as those in Eilers and Marx (1996) can be employed to introduce smoothness in the Cholesky factor. The benefit of using penalized likelihood for smoothing is that we can introduce shrinkage and smoothing altogether in covariance matrix estimation. Combining shrinkage and smoothing can do better than using shrinkage and smoothing alone. We have experimented with this idea and obtained promising results. This work will be reported elsewhere.

Bayesian methods have been developed in the literature to cope with dimensionality problems in estimation of covariance matrices. Daniels and Kass (2001) and Daniels and Pourahmadi (2002) studied Bayesian models for shrinking a covariance matrix towards some structure. Smith and Kohn (2002) proposed a Bayesian method to identify parsimony in the covariance matrix of longitudinal data. To get some idea how the regularized MLE compares

Table 4: Simulations results for the four examples in Smith and Kohn (2002). Reported are the sample medians of the entropy loss for 100 simulation runs. The results for the Smith and Kohn method (SK) are adapted from the original paper.

	$n = 40$, $m = 15$		$n = 100$, $m = 30$	
	SK	Spline	SK	Spline
Eg1	0.409	0.159	0.291	0.081
Eg2	0.356	0.308	0.310	0.284
Eg3	0.963	0.473	0.656	0.441
Eg4	3.854	2.748	5.661	7.059

with the Bayesian model averaging, we applied the proposed method to the four examples used in the simulations in Smith and Kohn (2002). Results for the entropy loss are reported in Table 4. The proposed method does better than the Smith-Kohn method in the first three examples while the message on the fourth example is mixed. Obviously this simulation study is very preliminary and non-conclusive. More systematic comparison of the Bayesian methods and the regularized MLE is an important area for future research. Progress to expedite computation of the Bayesian methods would be critical for extensive Monte Carlo studies.

Appendix

A.1 Derivation for Algorithm 1

Updating $\boldsymbol{\beta}_0$. Using (6) and (7), the log likelihood (4), viewed as a function of $\boldsymbol{\beta}_0$ and denoted as $l_1(\boldsymbol{\beta}_0)$, can be written as

$$l_1(\boldsymbol{\beta}_0) = -\frac{n}{2}\sum_{t=1}^{m}\left[\boldsymbol{\beta}_0^T\mathbf{B}_0\left(\frac{t}{m+1}\right) + g_{tt}\exp\left\{-\boldsymbol{\beta}_0^T\mathbf{B}_0\left(\frac{t}{m+1}\right)\right\}\right].$$

The Score and Hessian of $l_1(\boldsymbol{\beta}_0)$ with respect to $\boldsymbol{\beta}_0$ are

$$\begin{aligned} S_{\boldsymbol{\beta}_0} = \frac{\partial l_1(\boldsymbol{\beta}_0)}{\partial \boldsymbol{\beta}_0} = -\frac{n}{2}\sum_{t=1}^{m}\Bigg[&\mathbf{B}_0\left(\frac{t}{m+1}\right) \\ &- \exp\left\{-\boldsymbol{\beta}_0^T\mathbf{B}_0\left(\frac{t}{m+1}\right)\right\}\mathbf{B}_0\left(\frac{t}{m+1}\right)g_{tt}\Bigg] \end{aligned}$$

and

$$H_{\boldsymbol{\beta}_0,\boldsymbol{\beta}_0} = \frac{\partial^2 l_1(\boldsymbol{\beta}_0)}{\partial \boldsymbol{\beta}_0 \boldsymbol{\beta}_0^T} = -\frac{n}{2}\sum_{t=1}^{m}\Bigg[\exp\bigg\{-\boldsymbol{\beta}_0^T \mathbf{B}_0\Big(\frac{t}{m+1}\Big)\bigg\} \\ \mathbf{B}_0\Big(\frac{t}{m+1}\Big)\mathbf{B}_0^T\Big(\frac{t}{m+1}\Big) g_{tt}\Bigg].$$

The score and Hessian are used in the Newton-Raphson updates.

Updating $\boldsymbol{\beta}$. Recall that the (j,k)-th element of S is s_{jk}. The relevant pieces of the log likelihood with respect to the GARP parameters $\{\phi_{tj}, j=1,\ldots,t-1, t=2,\ldots,m\}$, can be written as

$$l_2(\boldsymbol{\beta}) = -\sum_{t=2}^{m}\frac{1}{d_t^2}\sum_{j=1}^{t}\sum_{k=1}^{t}\phi_{tj}\phi_{tk}s_{jk}, \tag{16}$$

where $\phi_{tt}=-1$ for $t=1,\ldots,m$. Note that $s_{tk}=s_{kt}$, the log likelihood (16) as a function of $\boldsymbol{\beta}$ can be written as

$$\begin{aligned} l_2(\boldsymbol{\beta}) &= -\sum_{t=2}^{m}\frac{1}{d_t^2}\Bigg(\sum_{j=1}^{t-1}\sum_{k=1}^{t-1}\phi_{tj}\phi_{tk}s_{jk} - 2\sum_{k=1}^{t-1}\phi_{tk}s_{tk} + s_{tt}\Bigg) \\ &= -\sum_{t=2}^{m}\exp\bigg\{-\boldsymbol{\beta}_0^T\mathbf{B}_0\Big(\frac{t}{m+1}\Big)\bigg\}\times \\ &\quad \Bigg(\sum_{j=1}^{t-1}\sum_{k=1}^{t-1}\boldsymbol{\beta}^T\mathbf{z}_{tj}\mathbf{z}_{tk}^T\boldsymbol{\beta}s_{jk} - 2\sum_{k=1}^{t-1}\boldsymbol{\beta}^T\mathbf{z}_{tk}s_{tk} + s_{tt}\Bigg), \end{aligned} \tag{17}$$

where $\mathbf{z}_{tj}$ is defined in Section 3.2. Set the first derivative of $l_2(\boldsymbol{\beta})$ with respect to $\boldsymbol{\beta}$ to 0, we have

$$\begin{aligned} -\sum_{t=2}^{m}\exp\bigg\{-\boldsymbol{\beta}_0^T\mathbf{B}_0\Big(\frac{t}{m+1}\Big)\bigg\}\bigg\{\sum_{j=1}^{t-1}\sum_{k=1}^{t-1}s_{jk}(\mathbf{z}_{tj}\mathbf{z}_{tk}^T \\ + \mathbf{z}_{tk}\mathbf{z}_{tj}^T)\boldsymbol{\beta} - 2\sum_{k=1}^{t-1}s_{tk}\mathbf{z}_{tk}\bigg\} = 0 \end{aligned} \tag{18}$$

or $A\boldsymbol{\beta}=\mathbf{b}$ with A and $\mathbf{b}$ defined in (8) and (9). This implies that $\boldsymbol{\beta}=A^{-1}\mathbf{b}$.

A.2 Derivation for Algorithm 2

Note that $\boldsymbol{\beta}^T \mathbf{z}_{tk} = \boldsymbol{\beta}_{t-k}^T \mathbf{B}(k/(m-(t-k)+1))$. The log likelihood (17) can be written as $l_2(\boldsymbol{\beta}) = \mathrm{I} + 2\,\mathrm{II} + \mathrm{III}$, where

$$\mathrm{I} = -\sum_{t=2}^{m}\Bigg[\exp\bigg\{-\boldsymbol{\beta}_0^T \mathbf{B}_0\bigg(\frac{t}{m+1}\bigg)\bigg\} \times \sum_{j=1}^{t-1}\sum_{k=1}^{t-1} \boldsymbol{\beta}_{t-j}^T \\ \mathbf{B}\bigg(\frac{j}{m-(t-j)+1}\bigg)\mathbf{B}^T\bigg(\frac{k}{m-(t-k)+1}\bigg)\boldsymbol{\beta}_{t-k} s_{jk}\Bigg],$$

$$\mathrm{II} = \sum_{t=2}^{m}\exp\bigg\{-\boldsymbol{\beta}_0^T \mathbf{B}_0\bigg(\frac{t}{m+1}\bigg)\bigg\}\sum_{k=1}^{t-1}\boldsymbol{\beta}_{t-k}^T \mathbf{B}\bigg(\frac{k}{m-(t-k)+1}\bigg)s_{tk}$$

and

$$\mathrm{III} = -\sum_{t=2}^{m}\exp\bigg\{-\boldsymbol{\beta}_0^T \mathbf{B}_0\bigg(\frac{t}{m+1}\bigg)\bigg\}s_{tt}.$$

Since III does not depend on $\boldsymbol{\beta}$, it is irrelevant when we update $\boldsymbol{\beta}$. Now we re-arrange I and II. Let $j_1 = t-j$ and $k_1 = t-k$, we have that

$$\begin{aligned}\mathrm{I} &= -\sum_{t=2}^{m}\Bigg[\exp\bigg\{-\boldsymbol{\beta}_0^T \mathbf{B}_0\bigg(\frac{t}{m+1}\bigg)\bigg\}\sum_{j_1=1}^{t-1}\sum_{k_1=1}^{t-1}\boldsymbol{\beta}_{j_1}^T \\ &\qquad \mathbf{B}\bigg(\frac{t-j_1}{m-j_1+1}\bigg)\mathbf{B}^T\bigg(\frac{t-k_1}{m-k_1+1}\bigg)\boldsymbol{\beta}_{k_1} s_{t-j_1,t-k_1}\Bigg] \\ &= -\sum_{j=1}^{m-1}\Bigg[\boldsymbol{\beta}_j^T \mathbf{B}\bigg(\frac{t-j}{m-j+1}\bigg)\sum_{k=1}^{m-1}\boldsymbol{\beta}_k^T \mathbf{B}\bigg(\frac{t-k}{m-k+1}\bigg) \\ &\qquad \sum_{t=(k+1)\vee(j+1)}^{m}\exp\bigg\{-\boldsymbol{\beta}_0^T \mathbf{B}_0\bigg(\frac{t}{m+1}\bigg)\bigg\}s_{t-j,t-k}\Bigg],\end{aligned}$$

where $a \vee b = \max\{a, b\}$. Similarly,

$$\begin{aligned}\mathrm{II} &= \sum_{t=2}^{m}\exp\bigg\{-\boldsymbol{\beta}_0^T \mathbf{B}_0\bigg(\frac{t}{m+1}\bigg)\bigg\}\sum_{k_1=1}^{t-1}\boldsymbol{\beta}_{k_1}^T \mathbf{B}\bigg(\frac{t-k_1}{m-k_1+1}\bigg)s_{t,t-k_1} \\ &= \sum_{k=1}^{m-1}\boldsymbol{\beta}_k^T \mathbf{B}\bigg(\frac{t-k}{m-k+1}\bigg)\sum_{t=k+1}^{m}\exp\bigg\{-\boldsymbol{\beta}_0^T \mathbf{B}_0\bigg(\frac{t}{m+1}\bigg)\bigg\}s_{t,t-k}.\end{aligned}$$

Thus we can write $l_2(\boldsymbol{\beta}) = \sum_{j=1}^{m-1} l_{2j}(\boldsymbol{\beta}_j)$ by ignoring irrelevant constants, where

$$
\begin{aligned}
l_{2j}(\boldsymbol{\beta}_j) = -\boldsymbol{\beta}_j^T \mathbf{B}\left(\frac{t-j}{m-j+1}\right) \Bigg[& \sum_{k=1}^{m-1} \boldsymbol{\beta}_k^T \mathbf{B}\left(\frac{t-k}{m-k+1}\right) \\
& \sum_{t=(k+1)\vee(j+1)}^{m} \exp\left\{-\boldsymbol{\beta}_0^T \mathbf{B}_0\left(\frac{t}{m+1}\right)\right\} s_{t-j,t-k} \\
& + 2\boldsymbol{\beta}_j^T \mathbf{B}\left(\frac{t-j}{m-j+1}\right) \sum_{t=j+1}^{m} \exp\left\{-\boldsymbol{\beta}_0^T \mathbf{B}_0\left(\frac{t}{m+1}\right)\right\} s_{t,t-j}.
\end{aligned}
$$

Simple calculation yields $\partial l_{2j}(\boldsymbol{\beta}_j)/\partial \boldsymbol{\beta}_j = -2A_j\boldsymbol{\beta}_j + 2\mathbf{b}_j$, $j = 1, \ldots, m_0$, where A_j are $\mathbf{b}_j$ are given in (10) and (11). Set the partial derivatives to 0, we obtain that $\hat{\boldsymbol{\beta}}_j = A_j^{-1}\mathbf{b}_j$, $j = 1, \ldots, m_0$. These equations are used to sequentially update $\boldsymbol{\beta}_j$.

References

[1] Anderson, T. W. (2003), *An Introduction to Multivariate Statistical Analysis* (3rd ed.), New York: Wiley.

[2] Barnard, J., McCulloch, R. and Meng, X.-L. (2000), "Modeling covariance matrices in terms of standard deviations and correlations, with application to shrinkage", *Statistica Sinica*, 10, 1281-1311.

[3] Brown, L., Gans, N., Mandelbaum, A., Sakov, A., Shen, H., Zeltyn, S. and Zhao, L. (2005), "Statistical analysis of a telephone call center: a queueing-science perspective," *Journal of the American Statistical Association*, 100, 36-50.

[4] Chiu, T.Y.M., Leonard, T. and Tsui, K.W. (1996), "The matrix-logarithm covariance model," *Journal of the American Statistical Association*, 91, 198–210.

[5] de Boor, C. (2001), *A Practical Guide to Splines* (Revised ed.), New York: Springer.

[6] Daniels, M. J. and Kass, R. E. (2001). Shrinkage estimators for covariance matrices. *Biometrics*, 57, 1173–84.

[7] Daniels, M. J. and Pourahmadi, M. (2002). Bayesian analysis of covariance matrices and dynamic models for longitudinal data. *Biometrika*, 89, 553–66.

[8] Dempster, A.P, Laird, N.M and Rubin, D.B., (1977), "Maximum likelihood from incomplete data via the EM algorithm (with discussion)," *Journal of the Royal Statistical Society*, Ser. B, 39, 1-38.

[9] Diggle, P.J. and Verbyla, A.P. (1998), "Nonparametric estimation of covariance structure in longitudinal data," *Biometrics*, 54, 401–415.

[10] Eilers, P.H.C. and Marx, B.D. (1996), "Flexible Smoothing Using B-splines and Penalized Likelihood" (with comments and rejoinder), *Statistical Science*, 11, 89–121.

[11] Fan, J. and Gijbels, I. (1996), *Local Polynomial Modelling and its Applications*, London: Chapman and Hall.

[12] Gabriel, K. R. (1962), "Ante-dependence analysis of an ordered set of variables," *The Annals of Mathematical Statistics*, 33, 201-212.

[13] Huang, J. Z., Liu, N., Pourahmadi, M. and Liu, L. (2006), "Covariance matrix selection and estimation via penalised normal likelihood", *Biometrika*, 93, 85–98.

[14] Kenward, M. G. (1987), " A method for comparing profiles of repeated measurements," *Applied Statistics*, 36, 296-308.

[15] Lin, S.P. and Perlman, M.D. (1985), "A Monte Carlo comparison of four estimators of a covariance matrix," In *Multivariate Analysis*, 6, Ed. P. R. Krishnaiah, pp. 411-429, Amsterdam: North-Holland.

[16] Macchiavelli, R.E. and Arnold, S.F. (1994), "Variable order antedependence models," *Communications in Statistics, Part A - Theory and Methods*, 23, 13-22.

[17] Newton, H.J. (1988), *TIMESLAB: A Time Series Analysis Laboratory*, Pacific Grove, CA: Wadsworth & Brooks/Cole.

[18] Pourahmadi, M. (1999), "Joint mean-covariance models with applications to longitudinal data: Unconstrained parameterisation," *Biometrika*, 86, 677-690.

[19] Pourahmadi, M. (2000), "Maximum likelihood estimation of generalized linear models for multivariate normal covariance matrix," *Biometrika*, 87, 425-435.

[20] Shen, H. and Huang, J.Z. (2005), "Analysis of call center data using singular value decomposition," *Applied Stochastic Models in Business and Industry*, 21, 251-263.

[21] Smith, M., and Kohn, R. (2002), "Parsimonious Covariance Matrix Estimation for Longitudinal Data", *Journal of the American Statistical Association*, 97, 1141-1153.

[22] Stone, C. J., Hansen, M., Kooperberg, C. and Truong, Y. K. (1997). Polynomial splines and their tensor products in extended linear modeling (with discussion). *Annals of Statistics* , 25, 1371-1470.

[23] Wu, W.B. and Pourahmadi, M. (2003), "Nonparametric estimation of large covariance matrices of longitudinal data," *Biometrika*, 90, 831-844.

[24] Zimmerman, D.L. and Núñez-Antón, V. (1997), "Structured antedependence models for longitudinal data," In *Modeling Longitudinal and Spatially Correlated Data: Methods, Applications, and Future Directions,* Springer Lecture Notes in Statistics, No. 122, Ed. T.G. Gregoire et al., pp. 63–76. New York: Springer-Verlag.