Covariance selection and estimation via penalised normal likelihood

By JIANHUA Z. HUANG
Department of Statistics, Texas A & M University, College Station, TX 77843-3143, USA
jianhua@stat.tamu.edu

NAIPING LIU
Department of Statistics, University of Pennsylvania, Philadelphia, PA 19104-6340, USA
nliu@wharton.upenn.edu

MOHSEN POURAHMADI
Division of Statistics, Northern Illinois University, DeKalb, IL 60115-2854,USA
pourahm@math.niu.edu

AND
LINXU LIU
Department of Biostatistics, Mailman School of Public Health, Columbia University, New York, NY, 10032, USA.
lxliu@biostat.columbia.edu

Summary

We propose a nonparametric method to identify parsimony and to produce a statistically efficient estimator of a large covariance matrix. We reparameterise a covariance matrix through the modified Cholesky decomposition of its inverse or the one-step-ahead predictive representation of the vector of responses and reduce the nonintuitive task of modelling covariance matrices to the familiar task of model selection and estimation for a sequence of regression models. The Cholesky factor containing these regression coefficients is likely to have many off-diagonal elements that are zero or close to zero. Penalised normal likelihoods in this situation with L_1 and L_2 penalties are shown to be closely related to Tibshirani's (1996) LASSO approach and to ridge regression. Adding either penalty to the likelihood helps to produce more stable estimators by introducing shrinkage to the elements in the Cholesky factor, while, because of its singularity, the L_1 penalty will set some elements to zero and produce interpretable models. An algorithm is developed to compute the estimator and select the tuning parameter. The proposed maximum penalised likelihood estimator is illustrated using simulation and a real dataset involving estimation of a 102×102 covariance matrix.

Some key words: Cholesky decomposition; Crossvalidation; LASSO; L_p penalty; Model selection; Penalised likelihood; Shrinkage.

1. Introduction

The sample covariance matrix, the most commonly used estimator of a covariance matrix is known to be positive-definite and unbiased, but highly unstable for large covariance matrices (Lin & Perlman, 1985; Wong et al. 2003; Ledoit & Wolf, 2004). Recently, structured covariance matrices, with few parameters, reflecting characteristics such as compound symmetry, and autoregression of order one, have become popular in longitudinal studies and related areas, though using a structure far from the true covariance could lead to severe bias. Between these two extremes lies a wealth of structures that might yield data-driven methods that strike a balance between the variance and bias of the covariance estimator.

Estimation of covariance matrices is difficult because the number of unknown elements in the covariance matrix grows quadratically with the size of the matrix and because of the positive-definiteness constraint. Many existing methods deal directly with the individual elements of the covariance matrix; for a review of some of these methods see Diggle & Verbyla (1998), Diggle et al. (2002), Boik (2002) and Wong et al. (2003). Dempster (1972) was the first to recognise the inverse covariance matrix as the canonical parameter of a multivariate normal distribution. His covariance selection method which identifies zeros in the inverse covariance matrix offers parsimony, but does not guarantee positive-definiteness of the estimator. The positive-definiteness was taken care of by Leonard & Hsu (1992) and Chiu et al. (1996) who modelled the matrix logarithm of a covariance matrix, and by Pourahmadi (1999, 2000), who considered generalised linear models for covariances using components of the modified Cholesky decomposition of the inverse covariance matrix whose nonredundant entries are unconstrained and enjoy statistical interpretation as certain regression coefficients and variances.

In this paper, we develop a nonparametric and data-driven method in the spirit of Dempster's covariance selection to identify parsimony in the covariance matrix through the unit triangular factor T of its modified Cholesky decomposition. The nonredundant entries of the rows of this matrix are the regression coefficients of one variable based on its predecessors, so that the nonintuitive task of modelling a covariance matrix can be reduced to that of modelling several regression models (Wu & Pourahmadi, 2003). Thus, familiar regression techniques such as ridge regression and variable selection can be used to shrink the off-diagonal elements of T, and identify any existing structural zeros. To this end, we use a penalised normal likelihood function with an L_p penalty for the nonredundant entries of T. Since the matrix T, in essence, gauges the degrees of 'dependence' in the vector of responses, imposing such a penalty will reduce the risk of using too many parameters to capture the dependence.

Our approach is more flexible than but related to the recent work of Wu & Pourahmadi (2003) and the unpublished 2004 University of Pennsylvania Ph. D. thesis of N. Liu. They applied nonparametric smoothing, by local polynomials and splines, respectively, to the first few subdiagonals of the Cholesky factor and set to

zero the remaining subdiagonals, thereby restricting T to be a banded lower triangular matrix. In contrast, our approach here allows the zeros in the Cholesky factor to be irregularly placed. This seems to be an advantage over Wu & Pourahmadi's (2003) and N. Liu's thesis; in addition, we do not impose classical nonparametric smoothness restrictions on the Cholesky factor. Our approach is also related to a Bayesian approach proposed by Smith & Kohn (2002) which places a hierarchical prior to allow zero entries in T. Ledoit & Wolf (2004) considered shrinkage estimation of covariance matrices in a way rather different from our approach.

2. Modified Cholesky decomposition

In this section, we review the role of the modified Cholesky decomposition in the unconstrained reparametrisation of a covariance matrix and express the normal likelihood as a quadratic function of these new parameters (Pourahmadi, 1999, 2000).

For a positive-definite covariance matrix Σ, its modified Cholesky decomposition can be written as

$$T\Sigma T' = D, \tag{1}$$

where T is a unit lower-triangular matrix having ones on its diagonal and D is a diagonal matrix. The elements of T and D are uniquely defined and have interpretations as the successive regression coefficients and prediction error variances when measurements are regressed on their predecessors. To be more precise, let $y = (y_1, \ldots, y_n)'$ be a time-ordered random vector with mean zero and positive-definite covariance matrix Σ. For $1 \leq t \leq n$, let $\hat{y}_t$ stand for the linear least-squares predictor of y_t based on its predecessors $y_{t-1}, \ldots, y_1$, and let $\epsilon_t = y_t - \hat{y}_t$ be its prediction error with variance $\sigma_t^2 = \mathrm{var}(\epsilon_t)$. Thus, for $t = 1$, $\hat{y}_1 = E(y_1) = 0$, and, for $1 < t \leq n$, there are unique scalars ϕ_{tj} such that

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

Let $\epsilon = (\epsilon_1, \ldots, \epsilon_n)'$ be the vector of successive prediction errors. Then (2) written in matrix form becomes

$$\epsilon = Ty, \tag{3}$$

where T is a unit lower triangular matrix with $-\phi_{t,j}$ in the (t, j)th position for $2 \leq t \leq n$ and $j = 1, 2, \ldots, t-1$. Note that $\mathrm{cov}(\epsilon) = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_n^2) = D$. Since the ϵ_t's are uncorrelated, (1) follows from (3); that is, the matrix T diagonalises the covariance matrix Σ. The $\phi_{t,j}$'s are called the generalised autoregressive parameters and the σ_t^2's are the corresponding innovation (residual) variances.

Under the multivariate normal assumption on y, the loglikelihood function $\ell(\Sigma; y)$, ignoring an irrelevant constant, satisfies

$$-2\ell(\Sigma; y) = \log|\Sigma| + y'\Sigma^{-1}y.$$

Since, from (1), $|\Sigma| = |D| = \Pi_{t=1}^{n}\sigma_t^2$ and $\Sigma^{-1} = T'D^{-1}T$, we have

$$\begin{aligned} -2\ell(\Sigma; y) &= \log|D| + y'T'D^{-1}Ty \\ &= \sum_{t=1}^{n}\log\sigma_t^2 + \sum_{t=1}^{n}\frac{\epsilon_t^2}{\sigma_t^2}, \end{aligned} \tag{4}$$

which is written in terms of prediction errors and their variances or the nonredundant entries of the pair (T, D). Thus, the modified Cholesky decomposition of a covariance matrix provides a parameterisation of the covariance matrix with unconstrained parameters and transfers the difficult task of modelling a covariance matrix to that of modelling the sequence of regressions in (2). Parsimony in the Cholesky factor corresponds to zeros in the regression coefficients, and to identify such zeros is a familiar variable selection problem in regression.

3. Penalised likelihood

The above regression interpretation suggests that the familiar ideas of variable selection and regularisation for least-squares regression can be used for covariance matrix modelling. We explore in this section two such ideas, ridge regression (Hoerl & Kennard, 1970a,b) and LASSO (Tibshirani, 1996), using the general framework of penalised likelihood for regression models (Fan & Li, 2001).

Suppose that we observe $y_i = (y_{i1}, \ldots, y_{in})'$, $i = 1, \ldots, m$, a random sample from $N(0, \Sigma)$. Consider the modified Cholesky decomposition of Σ as described in (1). According to (4), the loglikelihood function $\ell(\Sigma; y_1, \ldots, y_m)$ of Σ based on $y_1, \ldots, y_m$, up to an additive constant, satisfies

$$-2\ell(\Sigma; y_1, \ldots, y_m) = \sum_{t=1}^{n}\left(m\log\sigma_t^2 + \sum_{i=1}^{m}\frac{\epsilon_{it}^2}{\sigma_t^2}\right),$$

where $\epsilon_{i1} = y_{i1}$ and $\epsilon_{it} = y_{it} - \sum_{j=1}^{t-1} y_{ij}\phi_{tj}$ for $t = 2, \ldots, n$. For a given $\lambda > 0$, define the penalised negative loglikelihood as

$$-2\ell(\Sigma; y_1, \cdots, y_m) + \lambda p(\{\phi_{tj}\}), \tag{5}$$

where $p(\cdot) \geq 0$ is a specified penalty function, and λ is a tuning parameter whose selection will be discussed in § 4.2. For fixed λ, minimising (5) with respect to $\{\phi_{tj}\}$ and σ_t^2 leads to a penalised likelihood estimator of T and D and hence of Σ. When $\lambda = 0$, minimisation of (5) simply gives the maximum likelihood estimator. We consider in this paper only the class of penalty functions that can be written as an L_p norm of the generalised autoregressive parameters. For $p > 0$, the penalised likelihood objective function with an L_p penalty has the form

$$-2\ell(\Sigma; y_1, \cdots, y_m) + \lambda\sum_{t=2}^{n}\sum_{j=1}^{t-1}|\phi_{tj}|^p. \tag{6}$$

The L_p penalty class has been considered for regression problems by Frank & Friedman (1993) and Fu (1998).

We focus here on two important members of the L_p penalty class, the L_2 penalty $p(\{\phi_{tj}\}) = \sum_{t=2}^{n}\sum_{j=1}^{t-1}\phi_{tj}^2$ and the L_1 penalty $p(\{\phi_{tj}\}) = \sum_{t=2}^{n}\sum_{j=1}^{t-1}|\phi_{tj}|$. As in ridge regression and LASSO, using these two penalties will introduce shrinkage estimates of the generalised autoregressive parameters ϕ_{tj} and hence the covariance matrix. The L_1 penalty also implements selection by making some generalised autoregressive parameter estimates to be exactly zero. As in least-squares regression, shrinkage and selection trade off bias against variance.

The penalised likelihood estimator based on the L_2 penalty can be derived as the Bayes posterior mode under independent diffuse priors for the innovation standard deviations σ_t and independent normal priors for the generalised autoregressive parameters ϕ_{tj}'s, with densities $f(\phi_{tj}) = \{\lambda/(2\pi)\}^{1/2}\exp(-\lambda\phi_{tj}^2)$. Similarly, the penalised likelihood estimator based on the L_1 penalty is the Bayes posterior mode under independent diffuse priors for the innovation standard deviations and independent double-exponential priors for the generalised autoregressive parameters ϕ_{tj}'s, with densities $f(\phi_{tj}) = (\lambda/2)\exp(-\lambda|\phi_{tj}|)$.

4. Computing the penalised likelihood estimates

4·1 *The algorithm*

The algorithm amounts to applying a similar regression algorithm repeatedly to the rows of the Cholesky factor T.

For the L_p penalty, the penalised negative loglikelihood (6) becomes

$$\begin{aligned}&\sum_{t=1}^{n}\left(m\log\sigma_t^2 + \sum_{i=1}^{m}\frac{\epsilon_{it}^2}{\sigma_t^2}\right) + \lambda\sum_{t=2}^{n}\sum_{j=1}^{t-1}|\phi_{tj}|^p \\ &= \left(m\log\sigma_1^2 + \sum_{i=1}^{m}\frac{\epsilon_{i1}^2}{\sigma_1^2}\right) + \sum_{t=2}^{n}\left(m\log\sigma_t^2 + \sum_{i=1}^{m}\frac{\epsilon_{it}^2}{\sigma_t^2} + \lambda\sum_{j=1}^{t-1}|\phi_{tj}|^p\right).\end{aligned}$$

To minimise it, we need only to minimise

$$m\log\sigma_1^2 + \sum_{i=1}^{m}\frac{\epsilon_{i1}^2}{\sigma_1^2} \tag{7}$$

and

$$m\log\sigma_t^2 + \sum_{i=1}^{m}\frac{\epsilon_{it}^2}{\sigma_t^2} + \lambda\sum_{j=1}^{t-1}|\phi_{tj}|^p, \qquad t = 2,\cdots,n. \tag{8}$$

The minimiser of (7) is given by $\sigma_1^2 = \sum_{i=1}^{m} y_{i1}^2/m$. For $t = 2,\ldots,n$, the expression in (8) can be minimised by alternating minimisation over σ_t and $\phi_{tj}, j = 1,\ldots,t-1$:

for fixed $\phi_{tj}, j = 1, \ldots, t-1$, (8) is minimised with respect to σ_t by

$$\sigma_t^2 = \frac{1}{m}\sum_{i=1}^{m}\epsilon_{it}^2 = \frac{1}{m}\sum_{i=1}^{m}\Big(y_{it} - \sum_{j=1}^{t-1} y_{ij}\phi_{tj}\Big)^2; \tag{9}$$

for fixed σ_t, (8), as a function of ϕ_{tj}, $j = 1, \ldots, t-1$, is minimised by the minimiser of

$$\sum_{i=1}^{m}\frac{(y_{it} - \sum_{j=1}^{t-1} y_{ij}\phi_{tj})^2}{\sigma_t^2} + \lambda\sum_{j=1}^{t-1}|\phi_{tj}|^p. \tag{10}$$

An iterative procedure for minimising (8) starts by first initialising σ_t, using for example the innovation standard error estimated without the penalty. We then minimise (10) to obtain ϕ_{tj}, $j = 1, \ldots, t-1$, and revise σ_t^2 as in (9). We iterate the process until convergence for each t, $t = 2, \cdots, n$. For details about minimisation of (10) with fixed σ_t see the Appendix.

4·2 *Selection of the tuning parameter*

We used crossvalidation and generalized crossvalidation to choose λ.

For fast computation of a value of λ, we prefer K-fold crossvalidation to leave-one-out crossvalidation, with $K = 5$ or 10 in practice. We randomly split the full dataset S into K subsets of about the same size, denoted by S^ν, $\nu = 1, \ldots, K$. For each ν, we use the data $S - S^\nu$ to estimate the parameters and S^ν to validate. The loglikelihood is used as the performance measure. For each λ, the K-fold crossvalidated loglikelihood criterion is

$$\mathrm{CV}(\lambda) = \frac{1}{K}\sum_{\nu=1}^{K}\bigg(s_\nu \log|\widehat{\Sigma}_{-\nu}| + \sum_{i\in I_\nu} y_i'\widehat{\Sigma}_{-\nu}^{-1}y_i\bigg),$$

where I_ν is the index set of the data in S^ν, s_ν is the size of I_ν, and $\widehat{\Sigma}_{-\nu}$ is the variance-covariance matrix estimated using the training data set $S - S^\nu$. Note that, for data in S^ν, the expected loglikelihood for variance-covariance matrix Σ is given by

$$E\bigg(s_\nu \log|\Sigma| + \sum_{i\in I_\nu} y_i'\Sigma^{-1}y_i\bigg).$$

We choose $\lambda = \hat{\lambda}$ to minimise $CV(\lambda)$. Our final estimate of Σ is based on $\hat{\lambda}$ and the full dataset.

Following Craven & Wahba (1979), we derive the generalised crossvalidation criterion as an approximation to the leave-one-out crossvalidation criterion

$$\frac{1}{mn}\sum_{i=1}^{m}\sum_{t=1}^{n}(y_{it} - \hat{y}_{it}^{(-i)})^2 = \frac{1}{mn}\sum_{t=1}^{n}\sum_{i=1}^{m}(y_{it} - \hat{y}_{it}^{(-i)})^2,$$

where $\hat{y}_{it}^{(-i)}$ are fitted values when the ith vector of observations y_i is removed from the sample. For $t = 1, \ldots, n$, let $y_{i(t)} = (y_{i1}, \ldots, y_{i,t-1})'$, $H_t = (\sum_{i=1}^m y_{i(t)} y_{i(t)}')/\sigma_t^2$, and

$$X_t = \frac{1}{\sigma_t} \begin{pmatrix} y_{1(t)}' \\ \vdots \\ y_{m(t)}' \end{pmatrix} = \frac{1}{\sigma_t} \begin{pmatrix} y_{11} & y_{12} & \cdots & y_{1,t-1} \\ y_{21} & y_{22} & \cdots & y_{2,t-1} \\ \vdots & \vdots & \vdots & \vdots \\ y_{m1} & y_{m2} & \cdots & y_{m,t-1} \end{pmatrix}.$$

Consider first the L_2 penalty and let $\hat{y}_{it}$ denote the fitted values of y_{it}. Then connection to ridge regression, it is easily seen that

$$\begin{pmatrix} \hat{y}_{1t} \\ \vdots \\ \hat{y}_{mt} \end{pmatrix} = X_t (H_t + \lambda I_t)^{-1} X_t' \begin{pmatrix} y_{1t} \\ \vdots \\ y_{mt} \end{pmatrix} = S_t \begin{pmatrix} y_{1t} \\ \vdots \\ y_{mt} \end{pmatrix},$$

$$y_{it} - \hat{y}_{it}^{(-i)} = \left(\frac{y_{it} - \hat{y}_{it}}{1 - S_{t,ii}} \right)^2,$$

where $S_t = X_t (H_t + \lambda I_t)^{-1} X_t$ with its (i,i)-element being $S_{t,ii}$. We approximate $S_{t,ii}$ by $\sum_{i=1}^m S_{t,ii}/m = \mathrm{tr}(S_t)/m$ in the leave-one-out crossvalidation criterion to obtain the generalised crossvalidation criterion

$$\mathrm{GCV}(\lambda) = \frac{1}{mn} \sum_{t=1}^n \sum_{i=1}^m \left(\frac{y_{it} - \hat{y}_{it}}{1 - \mathrm{tr}(S_t)/m} \right)^2.$$

In the calculation of $\mathrm{GCV}(\lambda)$, σ_t's should be replaced by their estimated values. For the L_1 penalty, an iterative algorithm is needed to minimise (10) and there is no closed-form expression that links $(y_{1t}, \ldots, y_{mt})$ to their predicted values $(\hat{y}_{1t}, \ldots, \hat{y}_{mt})$. Using outcomes from the last iteration of the minimisation of (10), we have approximately that

$$\begin{pmatrix} \hat{y}_{1t} \\ \vdots \\ \hat{y}_{mt} \end{pmatrix} = X_t (H_t + \lambda L_t^{(k)})^{-1} X_t' \begin{pmatrix} y_{1t} \\ \vdots \\ y_{mt} \end{pmatrix},$$

where $L_t^{(k)}$ is defined as in (A1). We thus define $\mathrm{GCV}(\lambda)$ for the case of the L_1 penalty using the same formula as for the L_2 penalty case except that, in the definition of S_t, we replace I_t by the matrix $L_t^{(k)}$.

5. Performance

In this section, we evaluate by simulations, the two tuning-parameter selection methods and we compare the performance of the maximum penalised likelihood estimator using the L_1 penalty to that using the L_2 penalty. We also consider three

other methods of estimating a covariance matrix, the sample covariance matrix and two minimax estimators (Muirhead, 1982, § 4.3). Robustness of the proposed method to the normality assumption is investigated by applying it to data from a multivariate t distribution.

The iterative algorithm described in § 4 is implemented using Compaq Visual Fortran 6. The IMSL Fortran subroutine `UVMIF` is used for the optimisation required to select the tuning parameter.

To gauge performance, we consider two loss functions, namely

$$\Delta_1(\Sigma, G) = \mathrm{tr}\Sigma^{-1}G - \log|\Sigma^{-1}G| - n \quad \text{and} \quad \Delta_2(\Sigma, G) = \mathrm{tr}(\Sigma^{-1}G - I)^2,$$

where Σ is the true covariance matrix and G is a positive-definite matrix. The first loss is usually called the entropy loss, while the second is typically called 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 $G^* = CGC'$, $\Sigma^* = C\Sigma C'$ for a nonsingular matrix C (Anderson, 2003, § 7.8). The corresponding risk functions are defined by

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

An estimator $\hat{\Sigma}_1$ is considered better than an estimator $\hat{\Sigma}_2$ for Δ_i if its risk function is smaller, that is, $R_i(\Sigma, \hat{\Sigma}_1) < R_i(\Sigma, \hat{\Sigma}_2)$. For more information about simulation-based comparison of covariance matrix estimators, see Lin & Perlman (1985).

The risk function of the proposed penalised likelihood estimator is approximated by Monte Carlo simulation. For the results presented below, $N = 100$ simulation runs were used. The risks of the sample covariance matrix and of the minimax estimators, corresponding to the entropy loss and the quadratic loss respectively, have closed-form expressions; see §4.3 of Muirhead (1982). Note that the minimax estimator depends on the risk function used.

We considered the following four covariance matrices.

Case 1. $\Sigma_1 = I$, the identity matrix.

Case 2. $\Sigma_2 = \mathrm{diag}(n, n-1, n-2, \ldots, 1)$.

Case 3. $\Sigma_3^{-1} = T'D^{-1}T$, where $D = 0.01 \times I$, and $T = (-\phi_{t,s})$, with $\phi_{t,t} = 1$, $\phi_{t+1,t} = 0.8$, and $\phi_{t,s} = 0$ otherwise: the AR(1) model.

Case 4. $\Sigma_4^{-1} = T'D^{-1}T$, where $D = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_n^2)$ with $\sigma_t^2 = \sigma^2\{1 - \frac{(t-1)\rho^2}{1+(t-1)\rho}\}$, $t \geq 1$, and $T = (-\phi_{t,s})$ with $\phi_{t,t} = 1$, $\phi_{t,j} = \rho\{1+(t-1)\rho\}^{-1}$, $t \geq 2$, $j = 1, \ldots, t-1$, $\sigma = 1$, and $\rho = 0.5$: the compound symmetry model.

We calculated the risks of various estimators for each Σ from the above list for different combinations of m and n where the data are multivariate normal. The results for $m = 100$ and $n = 30$ are given in Table 1. These results and those from different choices of m and n yield the following conclusions.

(i) For the L_2 penalty, performance of the estimator using generalised crossvalidation to select the tuning parameter is similar to that of using 5-fold crossvalidation. Thus either method can be used in practice for tuning parameter selection for the L_2 penalty.

(ii) For the L_1 penalty, performance of the penalised likelihood estimator using 5-fold crossvalidation is better than that of using generalised crossvalidation when the T factor of the modified Cholesky decomposition of the covariance matrix is sparse, containing many zeros. This suggests that the approximation involved in deriving the formula for GCV(λ) for the L_1 penalty case may be too crude. The 5-fold crossvalidation method should be the recommended method for tuning-parameter selection for the L_1 penalty.

(iii) When there are many zeros in the T matrix, the L_1 penalty with 5-fold crossvalidation does better than the L_2 penalty, because the L_1 penalty can effectively identify the sparsity of the T matrix while the L_2 penalty cannot. That generalised crossvalidation does not do as well as 5-fold crossvalidation for the L_1 penalty can also be seen by its ineffectiveness in identifying zeros in the T matrix; see Table 2.

(iv) When there are many small values in the T matrix, as with Σ_4, the L_2 penalty does better than the L_1 penalty.

(v) The penalised likelihood estimators almost always outperform the sample covariance matrix and the minimax estimator and in most cases the improvements are substantial, especially when n is large, $n \geq 10$.

To explore the robustness of the proposed methods to the normality assumption, we generated data from the multivariate t distribution, by taking data from $Y = X/\sqrt{(Z/\nu)}$, where $X \sim N(0, \Omega)$, $Z \sim \chi^2(\nu)$, and X and Z are independent. Since the t distribution has fat tails, the quadratic loss has a large variance and thus its mean is not a stable measure of performance. We focused on the entropy loss and observed results that are consistent with those for normal data; that is, the penalised likelihood methods substantially improve over the sample covariance matrix and the minimax estimator associated with the entropy loss. Table 3 gives results for $m = 100$, $n = 30$, and $\nu = 5$.

6. Telephone call centre data

In this section we illustrate our method for estimating a large covariance matrix by an application in forecasting the call arrival pattern to a telephone call centre. The data come from one call centre in a major U.S. northeastern financial organisation, containing the information about the time every call arrives at the service queue. For each day in 2002, except for 6 days when the data-collecting equipment was out of order, phone calls are recorded from 7:00am until midnight. We divided the 17-hour period into 102 10-minute intervals, and counted the number of calls arriving at the service queue during each interval. Here the interval length of 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 days when the recording equipment was faulty (Shen & Huang, 2005), we obtain observations for 239 days.

Denote the data for day i by $N_i = (N_{i1}, \ldots, N_{i,102})'$, $i = 1, \ldots, 239$, where N_{it} is the number of calls arriving at the call centre for the t-th 10-minute interval on 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). We apply our proposed penalised likelihood method to estimate the 102×102 covariance matrix based on the residuals from a fit of the saturated mean model. The L_1 penalty is preferred to the L_2 penalty based on 5-fold crossvalidation, see Table 4, and has helped identify a parsimonious structure of the T matrix in the modified Cholesky decomposition. Of the 5151 elements below the main diagonal of the estimated T matrix, 4144 are essentially zero, with absolute values less than 0.01. Several different random partitions of the data for 5-fold crossvalidation have been tried and similar results were obtained. With a Pentium III PC running our Fortran code, the computing time for calculating the penalised likelihood estimate, including tuning parameter selection using 5-fold crossvalidation, is about 20 minutes.

The estimated covariance matrix can be used for forecasting the number of arrivals later in the day using arrival patterns at earlier times of the day. Write $y_i = (y_{i1}, \ldots, y_{i,102})'$. Form the partition $y_i = (y_i^{(1)'}, y_i^{(2)'})'$, where $y_i^{(1)}$ and $y_i^{(2)}$ measure the arrival patterns in the early and later times of day i. For example, we can take $y_i^{(1)} = (y_{i1}, \ldots, y_{i,51})'$ and $y_i^{(2)} = (y_{i,52}, \ldots, y_{i,102})'$, which measure 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

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \ \Sigma = \begin{pmatrix} \Sigma_{11}, \Sigma_{12} \\ \Sigma_{21}, \Sigma_{22} \end{pmatrix}.$$

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

$$E(y_i^{(2)}|y_i^{(1)}) = \mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(y_i^{(1)} - \mu_1). \tag{11}$$

Without the normality assumption, this formula gives the best mean squared error linear forecast. In practice, we need to plug in estimates of μ and Σ. We can fit a saturated mean model for μ and use either the sample covariance matrix or the penalised likelihood covariance matrix estimate for Σ.

To compare the forecast performance using different covariance matrix estimates, we split the 239 days into training and test datasets. The data from the first 205 days, corresponding to January to October, form the training dataset that is used to estimate the mean and covariance structure. The estimates are then applied for forecasting using formula (11) for the 34 days in the test set, corresponding to November and 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

$$\mathrm{AE}_t = \frac{1}{34}\sum_{i=206}^{239} |\hat{y}_{it} - y_{it}|,$$

where y_{it} and $\hat{y}_{it}$ are the observed and forecast values respectively. In Fig. 1(a), we plot the AE_t for the forecast using the sample covariance matrix and the penalised likelihood covariance matrix estimate. In Fig. 1(b) we plot the percentage of times, among 34 days in the test dataset, on which the forecast based on penalised likelihood has smaller absolute forecast error. It shows clearly that the forecast based on penalised likelihood covariance matrix estimates outperforms that based on the sample covariance matrix. Based on AE_t, the former does better in 50 out of the 51 time intervals. Also, the percentage of days in 34 test days on which the former has smaller absolute forecast error exceeds 50% at 46 out of the 51 forecast points.

7. Discussion

Use of the L_1 or L_2 penalty introduces regularisation in the estimation of a covariance matrix. An alternative way of regularisation is through smoothing the T matrix in the modified Cholesky decomposition of covariance matrix; see Wu & Pourahmadi (2003) and N. Liu's thesis. The two methods complement each other. The smoothing method is better if T is indeed smooth. This has been confirmed in our simulation study, results not shown. When T is not smooth, but is sparse or contains many small elements, the penalised likelihood method proposed in this paper would be better. To illustrate the latter point, Table 5 reports the risks of various methods for estimating the matrix $\Sigma^{-1} = T'DT$ with $m = 40$ and $n = 15$, where $D = I$ and $T = (-\phi_{i,j})$ with $\phi_{i+1,i} = 0.8$ for odd i and $\phi_{i,j} = 0$ otherwise. Clearly the smoothing method yields worse results than the proposed penalisation method.

Acknowledgement

Jianhua Huang's and Mohsen Pourahmadi's work is partially supported by grants from the U.S. National Science Foundation. The editor and a referee have provided helpful comments which led to significant improvement of the paper. We would like to thank Avi Mandelbaum for making the call centre data available to us. Haipeng Shen provided help with data management. We would also like to thank Larry Brown, Avi Mandelbaum and Haipeng Shen for helpful discussion about the call centre data analysis.

Appendix

Minimisation of expression (10).

We first give some implementation details of the minimisation of (10) for $p = 2$, because it is an important component of our proposed iterative procedure when $p = 1$. If $\phi_{t(t)} = (\phi_{t1}, \phi_{t2}, \ldots, \phi_{t,t-1})'$ and $y_{i(t)} = (y_{i1}, y_{i2}, \ldots, y_{i,t-1})'$, the first term of (10) can be written as

$$\frac{1}{\sigma_t^2}\sum_{i=1}^{m}(y_{it} - y'_{i(t)}\phi_{t(t)})^2 = c_t - 2g'_t\phi_{t(t)} + \phi'_{t(t)}H_t\phi_{t(t)},$$

where $c_t = (\sum_{i=1}^m y_{it}^2)/\sigma_t^2$, $g_t = (\sum_{i=1}^m y_{it}y_{i(t)})/\sigma_t^2$ and $H_t = (\sum_{i=1}^m y_{i(t)}y'_{i(t)})/\sigma_t^2$. Thus, for the L_2 penalty, minimisation of (10) leads to a closed-form solution. Indeed,

$$\sum_{i=1}^{m}\frac{(y_{it} - \sum_{j=1}^{t-1} y_{ij}\phi_{tj})^2}{\sigma_t^2} + \lambda\sum_{j=1}^{t-1}\phi_{tj}^2 = c_t - 2g'_t\phi_{t(t)} + \phi'_{t(t)}(H_t + \lambda I_t)\phi_{t(t)},$$

which is minimised by $\phi_{t(t)} = (H_t + \lambda I_t)^{-1}g_t$ for fixed σ_t, where I_t is the $(t-1) \times (t-1)$ identity matrix.

For the L_1 penalty, minimisation of (10) does not have a closed-form solution and an iterative algorithm is necessary. For each t the problem is equivalent to the minimisation of

$$\sum_{i=1}^{m}\frac{(y_{it} - \sum_{j=1}^{t-1} y_{ij}\phi_{tj})^2}{\sigma_t^2} \qquad \text{subject to} \sum_{j=1}^{t-1}|\phi_{tj}| \le u,$$

which is the same optimisation problem as LASSO. This can be thought of as a quadratic programming problem with linear inequality constraints, so standard numerical techniques could be applied; see Tibshirani (1996). However, we use an iterative algorithm that can be coded directly. It worked well in our simulation study and data analysis. The main idea of the algorithm is an iterative local quadratic approximation of $\sum_{j=1}^{t-1}|\phi_{tj}|$ (Fan & Li, 2001; Öjelund et al., 2001). The initial value of the iteration is taken to be the minimiser of (10) without the penalty term, that is, $\phi_{t(t)}^{(0)} = H_t^{-1}g_t$ or, when H_t is singular, $\phi_{t(t)}^{(0)}$ is the minimiser of (10) with an L_2 penalty. Denote the value of $\phi_{t(t)}$ at step k of the iteration by $\phi_{t(t)}^{(k)} = (\phi_{t1}^{(k)}, \phi_{t2}^{(k)}, \cdots, \phi_{t,t-1}^{(k)})'$. Since $|\phi_{tj}|$ can be approximated by the quadratic function (Fan & Li, 2001, §3.3)

$$\frac{|\phi_{tj}^{(k)}|}{2} + \frac{\phi_{tj}^2}{2|\phi_{tj}^{(k)}|}$$

in the neighbourhood of $\phi_{tj}^{(k)}$, then $\sum_{j=1}^{t-1}|\phi_{tj}|$ can be approximated by

$$\sum_{j=1}^{t-1}\frac{|\phi_{tj}^{(k)}|}{2} + \sum_{j=1}^{t-1}\frac{\phi_{tj}^2}{2|\phi_{tj}^{(k)}|} = c_t^k + \phi'_{t(t)}L_t^{(k)}\phi_{t(t)}$$

in the neighbourhood of $\phi^{(k)}_{t(t)} = (\phi^{(k)}_{t1}, \phi^{(k)}_{t2}, \cdots, \phi^{(k)}_{t,t-1})'$, where $c^k_t = \sum_{j=1}^{t-1} |\phi^{(k)}_{tj}|/2$ is a constant and

$$L^{(k)}_t = \text{diag}\left(\frac{1}{2|\phi^{(k)}_{t1}|} \; \frac{1}{2|\phi^{(k)}_{t2}|} \; \cdots \; \frac{1}{2|\phi^{(k)}_{t,t-1}|}\right) \tag{A1}$$

is a $(t-1) \times (t-1)$ diagonal matrix. Note that $|\phi^{(k)}_{i,j}|$ appears in the denominator; when any of $|\phi^{(k)}_{i,j}|$ falls below a preset threshold, such as 10^{-10}, replace it by the threshold value. Thus, for the L_1 penalty, (10) can be approximated by

$$\begin{aligned}
&\frac{1}{\sigma^2_t}\sum_{i=1}^{m}(y_{it} - y'_{i(t)}\phi_{t(t)})^2 + \lambda\sum_{j=1}^{t-1}|\phi_{tj}| \\
&= c_t - 2g'_t\phi_{t(t)} + \phi'_{t(t)}H_t\phi_{t(t)} + \lambda c^k_t + \lambda\phi'_{t(t)}L^{(k)}_t\phi_{t(t)} \\
&= c_t + \lambda c^k_t - 2g'_t\phi_{t(t)} + \phi'_{t(t)}(H_t + \lambda L^{(k)}_t)\phi_{t(t)}.
\end{aligned}$$

Hence, at step $(k+1)$, the minimiser of (10) for $p = 1$ is $\phi^{(k+1)}_{t(t)} = (H_t + \lambda L^{(k)}_t)^{-1}g_t$. Repeat this process until convergence. The iterative procedure described above can be viewed as an application of the MM algorithms, the convergence of which has been studied in Hunter and Li (2005).

References

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

Boik, R.J. (2002). Spectral models for covariance matrices. *Biometrika* **89**, 159–82.

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. *J. Am. Statist. Assoc.* **100**, 36–50.

Chiu, T.Y.M., Leonard, T. & Tsui, K.W. (1996). The matrix-logarithm covariance model. *J. Am. Statist. Assoc.* **91**, 198–210.

Craven, P. & Wahba, G. (1979). Smoothing noisy data with spline functions. *Numer. Math.* **31**, 377–403.

Dempster, A. (1972). Covariance selection. *Biometrics* **28**, 157–75.

Diggle, P.J. & Verbyla, A.P. (1998). Nonparametric estimation of covariance structure in longitudinal data. *Biometrics* **54**, 401–15.

Diggle, P.J., Heagerty, P., Liang, K.-Y. & Zeger, S.L. (2002). *Analysis of Longitudinal Data*, 2nd ed. Oxford: Oxford University Press.

Fan, J., & Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *J. Am. Statist. Assoc.*, **96**, 1348–60.

Frank, I.E. & Friedman, J.H. (1993). A statistical view of some chomometrics regression tools. *Technometrics* **35**, 109–48.

Fu, W.J. (1998). Penalized regressions: The bridge versus the LASSO. *J. of Comp. and Graph. Statist.* **7**, 397–416.

Hoerl, A.E. & Kennard, R.W. (1970a). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* **12**, 55–67.

Hoerl, A.E. & Kennard, R.W. (1970b). Ridge regression: Application to nonorthogonal problems. *Technometrics* **12**, 69–82.

Hunter, D.R. and Li, R. (2005). Variable selection using MM algorithms. *Ann. Statist.* **33**, 1617–42.

Ledoit, O. & Wolf, M. (2004). Honey, I shrunk the sample covariance matrix. *J. Portfolio Manag.* **4**, 110-9.

Leonard, T. & Hsu, J.S.J. (1992). Bayesian inference for a covariance matrix. *Ann. Statist.* **36**, 1669–96.

Lin, S.P. & Perlman, M.D. (1985). A Monte Carlo comparison of four estimators of a covariance matrix. In *Multivariate Analysis*, **6**, Ed. P. R. Krishnaiah, pp. 411–29. Amsterdam: North-Holland.

Muirhead, R.J. (1982). *Aspects of Multivariate Statistical Theory.* John Wiley & Sons.

Öjelund, H., Madsen, H. & Thyregod, P. (2001). Calibration with absolute shrinkage. *J. Chemomet.* **15**, 497–509.

Pourahmadi, M. (1999). Joint mean-covariance models with applications to longitudinal data: Unconstrained parameterisation. *Biometrika* **86**, 677–90.

Pourahmadi, M. (2000). Maximum likelihood estimation of generalized linear models for multivariate normal covariance matrix. *Biometrika* **87**, 425–35.

Shen, H. & Huang, J.Z. (2005). Analysis of call center arrival data using singular value decomposition. *Appl. Stoch. Models Bus. and Ind.* **21**, 251–63.

Smith, M., & Kohn, R. (2002). Parsimonious covariance matrix estimation for longitudinal Data. *J. Am. Statist. Assoc.* **97**, 1141–53.

Tibshirani, R. J. (1996). Regression shrinkage and selection via the LASSO. *J. R. Statist. Soc.* B **58**, 267–88.

Wong, F., Carter, C.K. & Kohn, R. (2003). Efficient estimation of covariance selection models. *Biometrika* **90**, 809–30.

Wu, W.B. & Pourahmadi, M. (2003). Nonparametric estimation of large covariance matrices of longitudinal data. *Biometrika* **90**, 831–44.

Table 1: *Risk comparision for* $m = 100, n = 30$. *Sample, minimax,* L_2 *penalty, and* L_1 *penalty in the table represent respectively the sample covariance matrix, the minimax estimator and the covariance matrix estimator based on the penalised likelihood with the* L_2 *and* L_1 *penalties. GCV and 5-fold CV denote the method used for selecting the tuning parameters in the penalised likelihood. The risks of the first two estimators are obtained using the formula in Muirhead (1982, § 4.3), and the others are based on average losses in 100 simulation runs.*

		Sample	Minimax	L_2 Penalty		L_1 Penalty	
				GCV	5-fold CV	GCV	5-fold CV
Entropy Loss	Σ_1	5.268	4.801	0.380	0.378	1.572	0.315
	Σ_2	5.268	4.801	0.785	0.785	2.089	0.303
	Σ_3	5.268	4.801	3.619	3.691	1.892	1.215
	Σ_4	5.268	4.801	1.571	1.423	2.475	2.388
Quadratic Loss	Σ_1	11.228	7.627	0.720	0.716	2.750	0.623
	Σ_2	11.228	7.627	1.382	1.382	3.563	0.601
	Σ_3	11.228	7.627	6.570	7.059	3.311	2.176
	Σ_4	11.228	7.627	2.729	2.435	4.592	4.465

GCV, generalised crossvalidation; CV, crossvalidation

Table 2: *Percentages of zeros identified among the zeros in the subdiagonal of the* T *matrix. Calculated based on 100 simulation runs.*

	GCV			5-fold CV		
	L. Quartile	Median	U. quartile	L. Quartile	Median	U. quartile
Σ_1	28.3%	29.9%	31.0%	77.7%	87.4%	94.7%
Σ_2	22.5%	24.4%	25.5%	75.5%	87.6%	89.7%
Σ_3	30.5%	32.5%	34.2%	51.2%	52.7%	54.2%

GCV, generalised crossvalidation; CV, crossvalidation; L., lower; U., upper

Table 3: *Risk comparison for multivariate t distribution using the entropy loss. The methods are as in Table 1. The risks are calculated using average losses over 100 simulation runs, with* $m = 100$, $n = 30$ *and* $\nu = 5$.

	Sample	Minimax	L_2 Penalty		L_1 Penalty	
			GCV	5-fold CV	GCV	5-fold CV
Σ_1	9.128	8.042	2.713	0.858	4.359	0.791
Σ_2	9.262	8.006	3.920	1.753	5.202	0.911
Σ_3	9.257	7.814	6.620	5.983	5.125	2.586
Σ_4	9.192	7.934	4.109	2.362	5.204	3.895

GCV, generalised crossvalidation; CV, crossvalidation

Table 4: *Call centre data. Selection of tuning parameters using 5-fold CV.*

	λ	5-fold CV
L_1 penalty	76.25	3316.83
L_2 penalty	451.60	5288.38

CV, crossvalidation

Table 5: *Risks for estimating a covariance matrix with a non-smooth T. The 'Smooth' method refers to the spline-smoothing method developed in N. Liu's thesis. Other methods in the table are as in Table 1. The risks of the first two estimators are obtained using the formula in Muirhead (1982, § 4.3), and the others are based on average losses in 100 simulation runs.*

Loss	Sample	Minimax	Smooth	L_1 Penalty	
				GCV	5-fold CV
Entropy Loss	3.582	3.226	3.275	1.621	1.317
Quadratic Loss	7.627	4.947	10.690	2.817	2.676

GCV, generalised crossvalidation; CV, crossvalidation

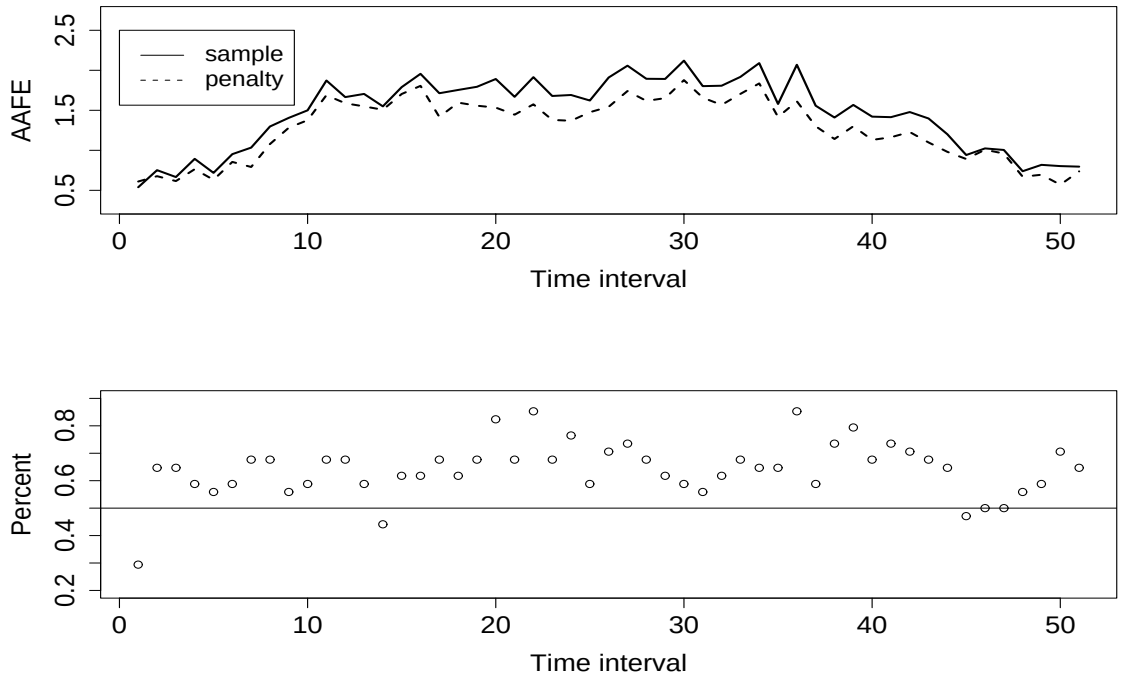

Figure 1: Call centre data. (a) Plot of AE_t for the forecasts using the sample covariance matrix, solid, and the penalised likelihood covariance matrix estimate, dashed. (b) Percentage of times, among 34 days in the test dataset, on which the penalised likelihood based forecast has smaller absolute forecast error.