Sparse Cholesky Covariance Parametrization for Recovering Latent Structure in Ordered Data

IRENE CÓRDOBA[1], CONCHA BIELZA[1], PEDRO LARRAÑAGA[1], AND GHERARDO VARANDO[2]
[1]Universidad Politécnica de Madrid, 28040 Madrid, Spain
[2]University of Copenhagen, 1165 København, Denmark

Corresponding author: Irene Córdoba (irene.cordoba@upm.es)

ABSTRACT The sparse Cholesky parametrization of the inverse covariance matrix is directly related to Gaussian Bayesian networks. Its counterpart, the covariance Cholesky factorization model, has a natural interpretation as a hidden variable model for ordered signal data. Despite this, it has received little attention so far, with few notable exceptions. To fill this gap, in this paper we focus on arbitrary zero patterns in the Cholesky factor of a covariance matrix. We discuss how these models can also be extended, in analogy with Gaussian Bayesian networks, to data where no apparent order is available. For the ordered scenario, we propose a novel estimation method that is based on matrix loss penalization, as opposed to the existing regression-based approaches. The performance of this sparse model for the Cholesky factor, together with our novel estimator, is assessed in a simulation setting , as well as over spatial and temporal real data where a natural ordering arises among the variables. We give guidelines, based on the empirical results, about which of the methods analysed is more appropriate for each setting.

INDEX TERMS Covariance matrix, sparse matrices, regression analysis, graphical model, Gaussian distribution

I. INTRODUCTION

The multivariate Gaussian distribution is central in both statistics and machine learning because of its wide applicability and well theoretical behaviour. Whenever it is necessary to reduce model dimension, such as in a high dimensional setting [1] or in graphical models [2], [3], sparsity is imposed in either the covariance matrix Σ or its inverse $\Omega = \Sigma^{-1}$.

The inverse covariance matrix Ω directly encodes information about partial correlations; therefore, when a zero pattern is present in Ω, it represents absent edges in the undirected graph of a Gaussian Markov network [4], [5]. Furthermore, letting $\Omega = \mathbf{W}\mathbf{W}^t$ be its Cholesky decomposition, a zero pattern in the lower triangular matrix $\mathbf{W}$ yields the acyclic digraph associated with a Gaussian Bayesian network [6], [7] model, up to a permutation of the variables [8]. As a result, much of the academic focus has been on sparsity in either the inverse covariance matrix or its Cholesky decomposition [9]–[13].

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Liu.

Conversely, a zero pattern in the covariance matrix Σ represents missing edges from the undirected graph of a covariance graph model [14]–[17]. However, a structured zero pattern on the Cholesky decomposition $\Sigma = \mathbf{T}\mathbf{T}^t$ of the covariance matrix has only been addressed by few works: [18] and [19]. In [18] the authors briefly analyse zeros in $\mathbf{T}$ mainly as a tool for better understanding of a higher-level graphical model called covariance chain, which is the main focus of their work. More directly related to our interests is the work of Rothman *et al.* [19], who directly explore a regression interpretation of the Cholesky factor $\mathbf{T}$ over the *error* variables. However, the authors do not address a structured zero pattern directly; instead, they focus on a banding structure for $\mathbf{T}$, inspired by the popularity of this estimator for the covariance matrix. In fact, a significant amount of the paper is devoted to analysing the relationship between the covariance matrix, or its inverse, and banded Cholesky factorization.

To fill this gap, in this paper we focus on arbitrary zero patterns in the Cholesky factor $\mathbf{T}$ of the covariance matrix Σ. We argue how this naturally models scenarios with

ordered variables representing noisy inputs from hidden signal sources, see for example [20] where latent brain regions are assumed to influence neural measurements. We discuss how this model could be extended to unordered variables by defining a new Gaussian graphical model over the Cholesky factorization of the covariance matrix. This new graphical model can be thought of as the analogue of a Gaussian Bayesian network, which is defined via sparse factorizations of the *inverse* covariance matrix. For ordered variables, we propose a novel learning method by directly penalising a matrix loss, in contrast with existing regression-based approaches in the literature [19]. This new estimator is feasible to compute thanks to a simplification of the loss gradient computation similar to the one proposed in [21]. Finally, we empirically assess the model performance in a broad range of experiments, exploring multiple simulation scenarios as well as real data where a natural spatial or temporal order arises between the variables.

The rest of the paper is organized as follows. In Section II we introduce the theoretical preliminaries necessary to follow the exposition. Then we detail the proposed sparse model for the Cholesky factor $\mathbf{T}$ of the covariance matrix $\mathbf{\Sigma}$, as well as introduce its extension to a Gaussian graphical model, in Section III. Afterwards, we discuss existing state-of-the-art estimation methods in Section IV, where we also detail our novel penalized matrix loss proposal. We assess the previously explained estimation methods in both simulated and real experiments, whose results are shown in Section V. Finally, we close the paper in Section VI, discussing the conclusions that can be drawn from the presented work, and also outlining the planned lines of future research. Appendices A, B and C are referenced throughout the paper and contain additional material for the interested reader.

II. THEORETICAL PRELIMINARIES

We will denote as $\mathcal{N}(\boldsymbol{\mu}, \mathbf{\Sigma})$ the multivariate Gaussian distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\mathbf{\Sigma}$. Since we focus on the zero pattern in $\mathbf{\Sigma}$, in the following we will set $\boldsymbol{\mu} = \mathbf{0}$ for notational simplicity, without loosing generality. As in the previous section, we will denote the inverse covariance matrix as $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$.

A. THE INVERSE COVARIANCE MATRIX AND SYSTEMS OF RECURSIVE REGRESSIONS

As we stated previously, the inverse covariance matrix and its decompositions have been thoroughly researched. If X is a p-variate Gaussian random vector, that is, if $X \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, then the *upper* Cholesky factorization of $\mathbf{\Omega}$ can be used to model ordered sequences of regressions [7], [9], as follows.

For $J \subseteq \{1, \ldots, p\}$, $i \notin J$ and $j \in J$, let $\boldsymbol{\beta}_{i|J} = (\mathbf{\Sigma}_{iJ} \mathbf{\Sigma}_{JJ}^{-1})^t$ denote the vector of regression coefficients of X_i on X_J, with its j-th entry being $\beta_{ij|J}$, the coefficient corresponding to variable X_j. We may equivalently write the statistical model for X as a system of recursive linear regression equations (also called a linear structural equation model [22]), where

for each $i \in \{1, \ldots, p\}$,

$$X_i = \sum_{j=1}^{i-1} \beta_{ij|1,\ldots,i-1} X_j + \mathcal{E}_i, \tag{1}$$

with $\mathcal{E}_i$ independent Gaussian random variables of zero mean and $\text{var}(\mathcal{E}_i) = \text{var}(X_i|X_1, \ldots, X_{i-1})$. The above regression system can also be expressed as $X = \mathbf{B}X + \mathcal{E}$, where $\mathbf{B}$ is a strictly lower triangular matrix with entries $b_{ij} = \beta_{ij|1,\ldots,i-1}$. Rearranging the previous equation, we obtain

$$X = \mathbf{L}\mathcal{E}, \tag{2}$$

where $\mathbf{L} = (\mathbf{I}_p - \mathbf{B})^{-1}$ with $\mathbf{I}_p$ the p-dimensional identity matrix. Now if we take variances and inverses on Equation (2), we arrive at the upper Cholesky decomposition of the inverse covariance matrix,

$$\mathbf{\Omega} = \mathbf{\Sigma}^{-1} = \mathbf{L}^{-t}\mathbf{D}^{-1}\mathbf{L}^{-1} = \mathbf{U}\mathbf{D}^{-1}\mathbf{U}^t = \mathbf{W}\mathbf{W}^t, \tag{3}$$

where $\mathbf{U} = \mathbf{L}^{-t} = (\mathbf{I}_p - \mathbf{B})^t$ and $\mathbf{W} = \mathbf{U}\sqrt{\mathbf{D}^{-1}}$ are upper triangular matrices, and $\mathbf{D}$ is a diagonal matrix containing the variances of $\mathcal{E}$.

Equations (1) and (3) are intimately related: all parameters of the recursive regression system are encoded in the upper Cholesky factorization. Indeed, we have that $\mathbf{D} = \text{var}(\mathcal{E})$ and the (j, i) entry of matrix $\mathbf{U}$, u_{ji}, is equal to $-\beta_{ij|1,\ldots,i-1}$, therefore all the regression coefficients can also be recovered, and $\mathbf{U}$ is explicitly written as

$$\mathbf{U} = \begin{pmatrix} 1 & -\beta_{21|1} & \cdots & -\beta_{p1|1,\ldots,p-1} \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & -\beta_{pp-1|1,\ldots,p-1} \\ 0 & \cdots & 0 & 1 \end{pmatrix}. \tag{4}$$

B. THE CHOLESKY DECOMPOSITION OF A COVARIANCE MATRIX

From Equation (2) we could have proceeded differently. Instead of taking variances and then inverses, most suited for analysing ordered sequences of variable regressions and subsequent Gaussian Bayesian networks (GBNs), we could have just taken variances and would have arrived at the Cholesky decomposition of the covariance matrix [19]

$$\mathbf{\Sigma} = \mathbf{L}\mathbf{D}\mathbf{L}^t = \mathbf{T}\mathbf{T}^t, \tag{5}$$

where now $\mathbf{L} = (\mathbf{I}_p - \mathbf{B})^{-1}$ and $\mathbf{T} = \mathbf{L}\sqrt{\mathbf{D}}$ are lower triangular.

Observe that Equation (5) is a direct analogue of Equation (3); although now there is not a natural interpretation of $\mathbf{L}$ entries in terms of regression coefficients. However, we will now show that the entries of $\mathbf{L}$ are in fact regression coefficients, just over a different conditioning set. Alternative derivations of this result can be found in [23, p. 158] and [18, p. 846]. The one in [23] is computational, based on the sweep matrix operator [24], whereas [18] provides a sketch based on a recursive expression for regression coefficients. We use instead simple identities over partitioned matrices.

Proposition 1: For $i \in \{1, \ldots, p\}$ and $j < i$, the (i, j) entry of matrix $\mathbf{L}$, denoted as l_{ij}, is equal to $\beta_{ij|1,\ldots,j}$.

Proof: For each $i \in \{1, \ldots, p\}$, $j < i$ and $J = \{1, \ldots, j\}$, the following partitioned identities hold [23, Equation (4.2.18)]

$$\boldsymbol{\Sigma}_{iJ} = \mathbf{L}_{iJ}\mathbf{D}_{JJ}\mathbf{L}_{JJ}^t,$$
$$\boldsymbol{\Sigma}_{JJ} = \mathbf{L}_{JJ}\mathbf{D}_{JJ}\mathbf{L}_{JJ}^t.$$

Therefore,

$$\begin{aligned}
\boldsymbol{\beta}_{i|J}^t &= \boldsymbol{\Sigma}_{iJ}\boldsymbol{\Sigma}_{JJ}^{-1} \\
&= \mathbf{L}_{iJ}\mathbf{D}_{JJ}\mathbf{L}_{JJ}^t(\mathbf{L}_{JJ}\mathbf{D}_{JJ}\mathbf{L}_{JJ}^t)^{-1} \\
&= \mathbf{L}_{iJ}\mathbf{L}_{JJ}^{-1}.
\end{aligned}$$

Furthermore, observe that, since $\mathbf{L}_{JJ}$ is lower triangular with ones along the diagonal, the last column of $\mathbf{L}_{JJ}^{-1}$ is always a vector of zero entries except the last entry, which is 1. This means, in particular, that for each $i \in \{1, \ldots, p\}$, $j < i$ and $J = \{1, \ldots, j\}$, the j-th element of row vector $\mathbf{L}_{iJ}\mathbf{L}_{JJ}^{-1}$ is equal to l_{ij}, which in turn is equal to the j-th entry of $\boldsymbol{\beta}_{i|J}$, $\beta_{ij|J}$. $\square$

The explicit expression in terms of regression coefficients for $\mathbf{L}$ is therefore,

$$\mathbf{L} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ \beta_{21|1} & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ \beta_{p1|1} & \cdots & \beta_{pp-1|1,\ldots,p-1} & 1 \end{pmatrix}. \quad (6)$$

From Equations (4) and (6) useful relationships can be determined; see, for example, [18, p.847] and Appendix A.

C. THE GAUSSIAN BAYESIAN NETWORK MODEL AND SPARSE CHOLESKY FACTORIZATIONS OF Ω

The upper Cholesky factorization in Equation (3) is typically used as a parametrization for GBNs, see, for example, [8], [13], [25]. In the following we will explain why, since this is closely related to the extension of our proposal to a graphical model.

Let $\mathcal{G} = (V, E)$ be an acyclic digraph, where $V = \{1, \ldots, p\}$ is the vertex set and $E \subseteq V \times V$ is the edge set. Assume that $1 \prec \cdots \prec p$ is a *topological order* of V, which means that for all $i \in V$, denoting as $\text{pa}(i)$ the parent set in $\mathcal{G}$ of node i, it holds that $\text{pa}(i) \subseteq \{1, \ldots, i-1\}$. The ordered Markov property of Bayesian networks states that [7]

$$X_i \perp\!\!\!\perp X_j | X_{\text{pa}(i)} \quad \text{for all } j < i \text{ and } j \notin \text{pa}(i), \quad (7)$$

where $X_i \perp\!\!\!\perp X_j | X_{\text{pa}(i)}$ stands for conditional independence [26], that is, X_i and X_j are independent given $X_{\text{pa}(i)} = \boldsymbol{x}_{\text{pa}(i)}$, for any value of $\boldsymbol{x}_{\text{pa}(i)}$. In the multivariate Gaussian distribution, the above conditional independence is equivalent to the regression coefficient $\beta_{ij|1,\ldots,i-1}$ being zero (see for example [27]). Since $\beta_{ij|1,\ldots,i-1} = 0$ if and only if $u_{ji} = 0$, then the zero pattern containing absent edges in $\mathcal{G}$ can be read off from $\mathbf{U}$ in Equation (3).

Now allow the ancestral order to be arbitrary, and denote as $\mathcal{U}(\mathcal{G})$ the set of matrices that have positive diagonal and zeros

compatible with a given acyclic digraph $\mathcal{G} = (V, E)$; that is, such that if $(j, i) \notin E$, $j \neq i$, then $m_{ji} = 0$ for all $\mathbf{M} \in \mathcal{U}(\mathcal{G})$. The GBN model can thus be expressed as

$$\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}) \text{ s.t. } \boldsymbol{\Sigma}^{-1} = \mathbf{W}\mathbf{W}^t \text{ with } \mathbf{W} \in \mathcal{U}(G). \quad (8)$$

Remark 1: Observe that if $\boldsymbol{X} = (X_1, \ldots, X_p)^t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ follows a GBN model with graph $\mathcal{G}$, the parameter matrix $\mathbf{W}$ of Equation (8) is not the Cholesky factor of $\boldsymbol{\Omega} = \boldsymbol{\Sigma}^{-1}$. This only occurs when the ancestral order of the nodes in $\mathcal{G}$ is $1 \prec \cdots \prec p$; that is, when the variables follow a natural order (the direct analogue of our model).

However, if we denote as $\kappa(\mathbf{M})$ the permutation of rows and columns in $\mathbf{M}$ following the ancestral order of $\mathcal{G}$, then $\kappa(\mathbf{W})$ is the upper Cholesky factor of $\kappa(\boldsymbol{\Omega})$.

III. SPARSE CHOLESKY DECOMPOSITION OF THE COVARIANCE MATRIX

We will hereby introduce the sparse Cholesky decomposition model $\boldsymbol{\Sigma} = \mathbf{T}\mathbf{T}^t$ for the covariance matrix, which consists of allowing an arbitrary zero pattern in $\mathbf{T}$. The entries of $\mathbf{T}$ are in correspondence with those in $\mathbf{L}$ and $\mathbf{D}$ (see Equation (5), $t_{ij} = l_{ij}\sqrt{d_{ii}}$), and therefore sometimes we will indistinctly refer to each of these matrices .

A. THE REGRESSION INTERPRETATIONS FOR THE COVARIANCE CHOLESKY FACTOR

In terms of model estimation, in $\mathbf{U}$ (Equation (4)) each column corresponds to the parameters of a single recursive regression, whereas each entry of matrix $\mathbf{L}$ (Equation (6)) corresponds to a different regression model.

Fortunately, [19] gave an alternative regression interpretation for matrix $\mathbf{L}$: recalling Equation (2), where the original variables X are written as a linear function of the error terms $\mathcal{E}$, and unfolding this matrix equation, we obtain

$$X_i = \sum_{j=1}^{i-1} l_{ij}\mathcal{E}_j + \mathcal{E}_i. \quad (9)$$

Thus obtaining an analogue of Equation (1), but now instead of regressing the original variables, the regression is performed over the *error* terms of the ordered recursive regressions in Equation (1).

Remark 2: The (i, j) entry of matrix $\mathbf{L}$ for $j < i$, l_{ij}, has therefore two interpretations as a regression coefficient:
1) It is the coefficient of the error $\mathcal{E}_j$ on the regression of X_i over $\mathcal{E}_1, \ldots, \mathcal{E}_{i-1}$.
2) It is the coefficient of variable X_j in the regression of X_i over $X_1, \ldots, X_j$.

Furthermore, from Equations (1) and (9) we have another dual interpretation for variable $\mathcal{E}_i$: it is the error of the regression of X_i onto $X_1, \ldots, X_{i-1}$, but also of X_i onto $\mathcal{E}_1, \ldots, \mathcal{E}_{i-1}$.

B. A HIDDEN VARIABLE MODEL INTERPRETATION

The sparse Cholesky parametrization of the covariance matrix naturally models a hidden variable structure [28]–[31]

over ordered Gaussian observables (Equation (2)). Interpreting the *error* terms $\mathcal{E}$ as latent signal sources, then the model is a sort of restricted GBN. This interpretation is naturally associated with the first regression coefficients outlined in Remark 2.

The constraints for this GBN that represents our model are:

- All arcs are from hidden variables $\mathcal{E}$ to the observed ones X.
- There is always an arc from $\mathcal{E}_i$ to X_i, for all $i \in \{1, \dots, p\}$.
- For each $i \in \{1, \dots, p\}$, only variables $\mathcal{E}_1, \dots, \mathcal{E}_{i-1}$ can have arcs to X_i.

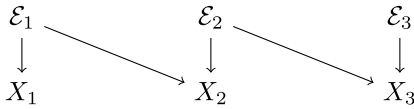Figure 1 represents one such restricted GBN compatible with our model.

FIGURE 1. Hidden variable model interpretation. We have omitted the index set notation for vertices and directly used variables for clarity.

The sparse Cholesky factor for Figure 1 would have the following lower triangular pattern

$$\begin{pmatrix} * & & \\ & * & \\ 0 & & * \end{pmatrix}, \tag{10}$$

where an asterisk means a non-zero entry. Observe that the zero value in entry $(3, 1)$ corresponds to the missing edge from $\mathcal{E}_1$ to X_3 in Figure 1.

C. A GRAPHICAL MODEL EXTENSION FOR UNORDERED VARIABLES

In direct analogy with GBNs and Equation (8), we can define a graphical model which is parametrized by a Cholesky factorization, up to a permutation. Let $\mathcal{G} = (V, E)$ be an arbitrary given acyclic digraph, and denote with $\prec$ its ancestral order. Let $\mathrm{pr}_\prec(i) = \{j \in V : j \prec i\}$ be the *predecessor* set of a node $i \in V$.

Analogous to $\mathcal{U}(\mathcal{G})$ in GBNs, denote as $\mathcal{L}(\mathcal{G})$ the set of matrices which have a zero pattern compatible with $\mathcal{G}$. By this we mean that if $(j, i) \notin E, j \neq i$, then $t_{ij} = 0$. We may define the Gaussian graphical model (comparable to Equation (8))

$$\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}) \text{ s.t. } \boldsymbol{\Sigma} = \mathbf{T}\mathbf{T}^t \text{ with } \mathbf{T} \in \mathcal{L}(\mathcal{G}). \tag{11}$$

Remark 3: As in the case of GBNs (Remark 1), the parameter matrix $\mathbf{T}$ in Equation (11) is lower triangular, and thus coincides with the Cholesky factor of $\boldsymbol{\Sigma}$, when the variables are already ancestrally ordered. Otherwise, the sparse Cholesky factorization model applies when the ancestral order $\prec$ is known, that is, denoting as $\kappa(\mathbf{M})$ the permutation of rows and columns of $\mathbf{M}$ following $\prec$, then $\kappa(\mathbf{T})$ is the Cholesky factor of $\kappa(\boldsymbol{\Sigma})$.

This extension has a more natural correspondence with the second regression interpretation of Remark 2, which holds

after reordering rows and columns of $\mathbf{T}$ to comply with $\prec$, as follows. First note that Equation (6) holds for an unordered version of $\mathbf{L}$, and thus we have that $t_{ij} \propto \beta_{ij|\,\mathrm{pr}_\prec(j)}$ for $j \prec i$. Therefore, we retrieve a sort of *ordered Markov property* (comparable to Equation (7))

$$X_i \perp\!\!\!\perp X_j | X_{\mathrm{pr}_\prec(j)} \text{ for all } j \prec i \text{ and } j \notin \mathrm{pa}(i). \tag{12}$$

A simple example of an arbitrary graph would be that in Figure 2. In this model, factor $\mathbf{T}$ would be lower triangular after reordering its rows and columns following the ancestral ordering of the graph, $2 \prec 1 \prec 3$,

$$\mathbf{T} = \begin{pmatrix} * & * & 0 \\ 0 & * & 0 \\ * & 0 & * \end{pmatrix}, \quad \kappa(\mathbf{T}) = \begin{pmatrix} * & 0 & 0 \\ * & * & 0 \\ 0 & * & * \end{pmatrix}.$$

FIGURE 2. Graph with ancestral order $2 \prec 1 \prec 3$. We have omitted the index set notation for vertices and directly used variables for clarity.

In the example of Figure 1, where variables already exhibit a natural order, the graph that would represent such interactions would be that in Figure 3, whose parameter matrix $\mathbf{T}$ is already lower triangular (see also Remark 3).

FIGURE 3. Graph corresponding to the model in Figure 1. We have omitted the index set notation for vertices and directly used variables for clarity.

A further study of this graphical model extension is out of the scope of this work, since we focus on naturally ordered variables from hidden signal sources, but we have discussed it here for completeness and interest for future work.

IV. MODEL ESTIMATION

We will first review two regression-based existing estimators for this model that can be found in [19], and then will detail our proposed penalized matrix loss estimation method.

A. EXISTING WORK: BANDING AND LASSO

Throughout this section denote as $\boldsymbol{x}_i$ a sample of size N corresponding to variable X_i, where $X = (X_1, \dots, X_p)$ is assumed to follow the regression model of Equation (9).

The banding estimate for $\mathbf{T}$ builds upon the respective for $\mathbf{L}$. The idea is to estimate by standard least squares only the first k sub-diagonals of $\mathbf{L}$ and set the rest to zero. Specifically, if $b(k) = \max(1, i - k)$ denotes the starting index, with respect to the band parameter k, of the i-th row vector $\boldsymbol{l}_i = (l_{ib(k)}, \dots, l_{ii-1})^t$ in matrix $\mathbf{L}$, then, letting $\hat{\boldsymbol{\varepsilon}}_{b(k)} = \boldsymbol{x}_{b(k)}$,

$$\hat{\boldsymbol{l}}_i = \arg\min_{\boldsymbol{l}_i} \|\boldsymbol{x}_i - (\hat{\boldsymbol{\varepsilon}}_{b(k)} \cdots \hat{\boldsymbol{\varepsilon}}_{i-1})\boldsymbol{l}_i\|_2^2,$$

$$\hat{\boldsymbol{\varepsilon}}_i = \boldsymbol{x}_i - (\hat{\boldsymbol{\varepsilon}}_{b(k)} \cdots \hat{\boldsymbol{\varepsilon}}_{i-1})\hat{\boldsymbol{l}}_i,$$

$$\hat{d}_{ii} = \frac{1}{N} \|\hat{\boldsymbol{\varepsilon}}_i\|_2^2, \tag{13}$$

where $\|\cdot\|_2$ is the Euclidean or l_2-norm. In order to ensure positive definiteness of all matrices involved in the computations, k must be smaller than $\min(N - 1, p)$ [19]. Matrix $\widehat{\mathbf{T}} = \widehat{\mathbf{L}}\sqrt{\widehat{\mathbf{D}}}$ inherits the band structure from $\widehat{\mathbf{L}}$. The main drawback of this banding estimator is the restrictive zero pattern that it imposes. Note also that this method requires previous selection of the parameter k.

An alternative to banding which gives more flexibility over the zero pattern is to use lasso regression over Equation (9),

$$\hat{\boldsymbol{l}}_i = \arg\min_{\boldsymbol{l}_i} \|\boldsymbol{x}_i - (\hat{\boldsymbol{\varepsilon}}_1 \cdots \hat{\boldsymbol{\varepsilon}}_{i-1})\boldsymbol{l}_i\|_2^2 + \lambda \|\boldsymbol{l}_i\|_1,$$

$$\hat{\boldsymbol{\varepsilon}}_i = \boldsymbol{x}_i - (\hat{\boldsymbol{\varepsilon}}_1 \cdots \hat{\boldsymbol{\varepsilon}}_{i-1})\hat{\boldsymbol{l}}_i,$$

$$\hat{d}_{ii} = \frac{1}{N}\|\hat{\boldsymbol{\varepsilon}}_i\|_2^2, \qquad (14)$$

where this time $\hat{\boldsymbol{\varepsilon}}_1 = \boldsymbol{x}_1$, $\boldsymbol{l}_i = (l_{i1}, \ldots, l_{ii-1})^t$, and $\|\cdot\|_1$ is the l_1-norm and $\lambda > 0$ is the penalisation parameter. Observe that such penalty could be replaced with any other sparsity inducing penalty over $\boldsymbol{l}_i$, for example, [19] discussed also the nested lasso [32] because the sparsity pattern is preserved in the covariance matrix. We have discarded such penalty because we are primarily interested in an arbitrary zero pattern in $\widehat{\mathbf{T}}$ and not worried about the induced one in $\widehat{\boldsymbol{\Sigma}}$.

B. PENALIZED GRADIENT-BASED LEARNING OF THE COVARIANCE SPARSE CHOLESKY FACTOR

The above approaches are based on the regression interpretation of the sparse Cholesky factor model, Equation (9). By contrast, we propose to directly estimate all of the parameters, that is, matrix $\mathbf{T}$ entries, by solving one optimization problem. This allows for example to recover maximum likelihood estimates, as well as have the potential to be easily extended to the graphical model interpretation (Equation (11)) following an approach similar to [33].

Denote as $\boldsymbol{\Sigma}(\mathbf{T})$ the parametrization of a covariance matrix $\boldsymbol{\Sigma}$ with its Cholesky factor $\mathbf{T}$ (Equation (5)). We propose to learn a sparse model for $\mathbf{T}$ by solving the following optimization problem

$$\arg\min_{\mathbf{T}} \phi\left(\boldsymbol{\Sigma}(\mathbf{T})\right) + \lambda\|\mathbf{T}\|_1, \qquad (15)$$

where $\phi(\cdot)$ is a differentiable loss function over covariance matrices, $\lambda > 0$ is the penalisation parameter, and $\|\cdot\|_1$ is the l_1-norm for matrices, which induces sparsity on $\mathbf{T}$ [34]. Note that, as in the regression case, the l_1 penalty could be replaced with any other sparsity inducing matrix norm. Solving Equation (15) can be done via proximal gradient algorithms, which have optimal convergence rates among first-order methods [34] and are tailored for a convex ϕ but also competitive in the non-convex case [21]. In this work we have used two such smooth loss functions: the negative Gaussian log-likelihood and the triangular Frobenious norm.

The negative Gaussian log-likelihood for a sample $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ when $\boldsymbol{\mu}$ is assumed to be zero is [27] proportional to

$$\phi_{NLL}(\boldsymbol{\Sigma}) = \ln\det(\boldsymbol{\Sigma}) + \text{tr}(\boldsymbol{\Sigma}^{-1}\hat{\boldsymbol{\Sigma}}), \qquad (16)$$

where $\hat{\boldsymbol{\Sigma}} = 1/N \sum_{n=1}^{N} \boldsymbol{x}_n \boldsymbol{x}_n^t$ is the maximum likelihood estimator for $\boldsymbol{\Sigma}$. On the other hand, the Frobenious norm loss that we also consider is

$$\phi_{FR}(\boldsymbol{\Sigma}) = \|\boldsymbol{\Sigma} - \hat{\boldsymbol{\Sigma}}\|_F^2 = \sum_{i=1}^{p}\sum_{j=1}^{p}(\sigma_{ij} - \hat{\sigma}_{ij})^2. \qquad (17)$$

Both ϕ_{NLL} and ϕ_{FR} are smooth, and in general ϕ_{NLL} renders the optimization problem of Equation (15) non-convex [35], whereas ϕ_{FR} is a convex function. In Appendix B we provide the details on gradient computations for ϕ_{NLL} and ϕ_{FR}, as well as the proximal algorithm pseudocode.

V. EXPERIMENTS

In all of the experiments we compare the four estimation methods outlined in the previous section: banding $\mathbf{T}$ (Equation (13)), lasso regressions (Equation (14)), and our two proposed penalized losses ϕ_{NLL} (Equation (16)) and ϕ_{FR} (Equation (17)). These four methods will be denoted in the remainder as band, lasso, grad_lik and grad_frob, respectively. All data was standardized, and therefore in our proposed losses the sample correlation matrix was used instead of $\hat{\boldsymbol{\Sigma}}$. The implementation of our loss optimization methods grad_frob and grad_lik can be found in the R [36] package *covchol*[1].

A. SIMULATION

We have conducted simulation experiments in two different scenarios. First, because as the work of Rothman *et al.* [19] is the most directly related to our model, we have replicated their simulation setting for completeness. Therein they select three fixed covariance matrices with either a fixed known banded sparsity pattern or no zeros at all. By contrast, in the second experiment we explore how the methods perform when the sparsity pattern is arbitrary.

In both experiments we have measured two statistics in order to assess both model selection (how precise the zero pattern is recovered) and estimation (how numerically close is the retrieved matrix to the original one). These metrics are evaluated over $\boldsymbol{\Sigma}$ and $\hat{\boldsymbol{\Sigma}}$ in the second experiment instead of $\mathbf{T}$ and $\widehat{\mathbf{T}}$, for better comparability with [19]. Specifically, we use the F1 score for evaluating the zero pattern,

$$\text{F1}(\mathbf{T}, \widehat{\mathbf{T}}) = 2\frac{\text{TPR}(\mathbf{T}, \widehat{\mathbf{T}})\,\text{TDR}(\mathbf{T}, \widehat{\mathbf{T}})}{\text{TPR}(\mathbf{T}, \widehat{\mathbf{T}}) + \text{TDR}(\mathbf{T}, \widehat{\mathbf{T}})}, \qquad (18)$$

where TPR and TDR are the true positive and discovery rate, respectively,

$$\text{TPR}(\mathbf{T}, \widehat{\mathbf{T}}) = \frac{|\{t_{ij} \neq 0 \text{ and } \hat{t}_{ij} \neq 0\}|}{|\{t_{ij} \neq 0\}|},$$

$$\text{TDR}(\mathbf{T}, \widehat{\mathbf{T}}) = \frac{|\{t_{ij} \neq 0 \text{ and } \hat{t}_{ij} \neq 0\}|}{|\{\hat{t}_{ij} \neq 0\}|};$$

---

[1]Version under development: https://github.com/irenecrsn/covchol. The experiments described throughout this section can be reproduced following the instructions and using the code available at the repository https://github.com/irenecrsn/chol-inv.

and the induced matrix 1-norm,

$$\text{NORM}(\mathbf{T}, \widehat{\mathbf{T}}) = \|\mathbf{T} - \widehat{\mathbf{T}}\|_1 = \max_{1 \le j \le p} \sum_{i=1}^{p} |\hat{t}_{ij} - t_{ij}|.$$

1) FIXED COVARIANCE MATRICES

The fixed covariance matrices used in the simulations by Rothman *et al.* [19] are:

- The autoregressive model of order 1, where the true covariance matrix $\boldsymbol{\Sigma}_1$ has entries $\sigma_{ij} = \rho^{|i-j|}$, with $\rho = 0.7$.
- The 4-banded correlation matrix $\boldsymbol{\Sigma}_2$ with entries $\sigma_{ij} = 0.4\mathbb{I}(|i-j| = 1) + 0.2\mathbb{I}(2 \le |i-j| \le 3) + 0.1\mathbb{I}(|i-j| = 4)$ for $i \ne j$, $\mathbb{I}$ being the set indicator function.
- The dense correlation matrix $\boldsymbol{\Sigma}_3$ with 0.5 in all of its entries except for the diagonal.

Similarly to [19], we use for matrix dimension p values ranging from 30 to 500, and draw from the respective distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_i)$, $i \in \{1, 2, 3\}$, a sample of size 200 which allows to visualize both the $p > N$ and $p < N$ scenarios. With this experiment we measure how sparsity inducing methods for learning $\mathbf{T}$ behave in scenarios which are not specially suited for them, except for band and $\boldsymbol{\Sigma}_2$.

Figure 4 shows the results of the aforementioned simulation scenario. The norms are shown in logarithmic scale for a better comparison between the methods, since there were significant disparities. $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_3$ are both dense matrices,
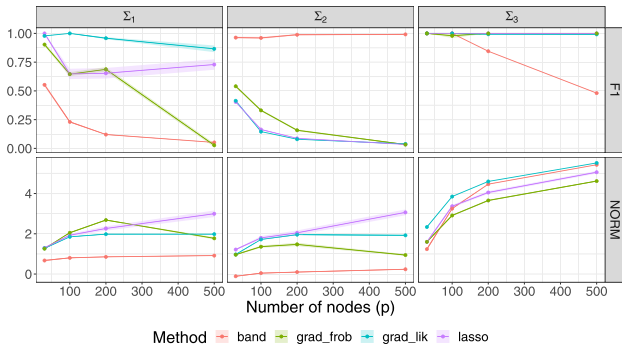

FIGURE 4. Results of the simulation experiment set out in [19]. Metric NORM is in logarithmic scale.

with $\boldsymbol{\Sigma}_1$ having entries that decay when moving away from the diagonal. We observe that the inexistent sparsity pattern is best approximated by grad_lik and lasso, but interestingly grad_frob and band achieve competitive norm results, sometimes even outperforming the rest. Matrix $\boldsymbol{\Sigma}_2$ is banded, therefore, as expected, band achieves both the highest F1 measure and lowest norm difference.

2) ARBITRARY SPARSITY PATTERN IN THE CHOLESKY FACTOR T

In this experiment the sparse Cholesky factor $\mathbf{T}$ is simulated using essentially the method of [37, Algorithm 3] with a random acyclic digraph to represent zero pattern, that is,

the latent structure (see Figure 1). Observe that in general this does not yield a uniformly sampled Cholesky factor, but it is more flexible that the standard diagonal dominance procedure, see [37] for further discussion on this issue. We generate three Cholesky factors $\mathbf{T}_i$ with a proportion of i/p non-zero entries (density/average edge number of the corresponding acyclic digraph), where $i \in \{1, 2, 3\}$. Then we draw a sample of size 200 from $\mathcal{N}(\mathbf{0}, \mathbf{T}_i\mathbf{T}_i^t)$ for p ranging between 30 and 500, as in the previous experiment.

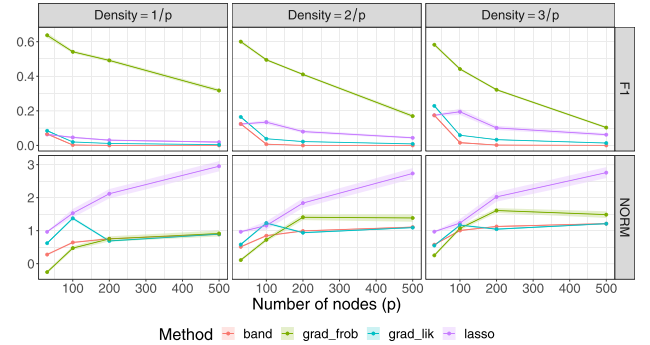Figure 5 depicts the results. Note that as the density

FIGURE 5. Results of the simulation experiment for an arbitrary sparsity pattern in $\mathbf{T}$. Metric NORM is in logarithmic scale. Density indicates the average proportion of lower triangular non-zero entries in the simulated $\mathbf{T}$ Cholesky factors.

decreases, the F1 score and matrix norm results slightly worsen, but in general the methods behaviour is maintained. The band estimator exhibits a performance similar to the previous experiment: although achieving a small F1 score, it has a relatively small matrix norm difference. This behaviour is shared in this case with grad_lik, which has in general poor performance. However, the worst performing method is lasso, which neither is able to recover the sparsity pattern, nor gets numerically close to the original Cholesky factor (it has a significantly high value for the norm difference). Conversely, method grad_frob has the best performance, with a significantly high F1 score when compared with the rest and competitive or best norm difference results.

B. REAL DATA

In this section we have selected two data sets from the UCI machine learning repository [38] where a natural order arises among the variables, and which are therefore suitable for our sparse covariance Cholesky factorization model. Both of them are labelled with a class variable, therefore after estimating the respective Cholesky factors with each method, we assess classification performance.

For classifying a sample $\boldsymbol{x}$ we use quadratic discriminant analysis [19], where $\boldsymbol{x}$ is classified in the class value c that maximizes

$$\ln \hat{f}(\boldsymbol{x}, c) = \ln \hat{f}(\boldsymbol{x}|c) + \ln \hat{f}(c),$$

where $\hat{f}(c)$ is the proportion of observations for class c and we have expressed $\hat{f}(x|c)$ in terms of $\mathbf{T}$ instead of $\boldsymbol{\Sigma}$,

$$
\begin{aligned}
\ln \hat{f}(\boldsymbol{x}|c) &\propto \frac{1}{2}\ln\det(\widehat{\boldsymbol{\Sigma}}_c) - \frac{1}{2}(\boldsymbol{x}-\hat{\boldsymbol{\mu}}_c)^t \widehat{\boldsymbol{\Sigma}}_c^{-1}(\boldsymbol{x}-\hat{\boldsymbol{\mu}}_c) \\
&= \ln\det(\widehat{\mathbf{T}}_c) - \frac{1}{2}(\boldsymbol{x}-\hat{\boldsymbol{\mu}}_c)^t \widehat{\mathbf{T}}_c^{-t}\widehat{\mathbf{T}}_c^{-1}(\boldsymbol{x}-\hat{\boldsymbol{\mu}}_c) \\
&= \sum_{i=1}^{p} \ln t_{ii} - \frac{1}{2}(\widehat{\mathbf{T}}_c^{-1}(\boldsymbol{x}-\hat{\boldsymbol{\mu}}_c))^t \widehat{\mathbf{T}}_c^{-1}(\boldsymbol{x}-\hat{\boldsymbol{\mu}}_c), \quad (19)
\end{aligned}
$$

with $\hat{\boldsymbol{\mu}}_c$, $\widehat{\boldsymbol{\Sigma}}_c$ and $\widehat{\mathbf{T}}_c$ the respective estimates from training samples belonging to class c.

Finally, for evaluating classification performance we have used:

- The F1 score, already defined in Equation (18) in terms of TDR and TPR, but adapted to classification instead of matrix entries.
- The true negative rate, TNR, since it is not contained in the F1 score, which is the proportion of observations that have been correctly *not* classified as class c.
- The accuracy, ACC, which measures the proportion of observations that have been correctly assigned a class. Observe that this last metric, unlike the other two, is not class-dependent, but instead global.

1) SONAR: MINE vs. ROCKS

The first real data set we explore is the *Connectionist Bench (Sonar, Mines vs. Rocks)* data set, which contains numeric observations from a sonar signal bounced at both a metal cylinder (mine) and rocks. It contains 60 variables and 208 observations. Each variable corresponds to the energy within a certain frequency band, integrated over a period of time, in increasing order. Each observation represents a different beam angle for the same object under detection. Over this data set the objective is to classify a sample as rock or mine. This data set was also analysed in [19], but without the expression in terms of $\mathbf{T}$ for Equation (19) and only using method band for $\mathbf{T}$.

As a first exploratory step, we have applied each of the methods for learning the Cholesky factor $\mathbf{T}$ to all the instances labelled as M (mines), and R (rocks), which we show as a heatmap in Figure 6. The Cholesky factor for mines retrieved

by grad_lik and lasso look fairly similar, whereas the one for rocks that lasso estimates is nearly diagonal. Bands can be clearly observed from heatmaps by band, and all methods impose zero values for variables near to or higher than 50, which could be motivated by the problem characteristics and hint at high sonar frequencies being nearly noiseless. This latter behaviour is inherited by the covariance matrices, whose heatmaps are shown in Appendix C, Figure 9. The entries in the Cholesky factor estimated by grad_frob are the most extreme, since most of them are zero, and the ones which are not have the highest and lowest values among all the estimates recovered. Despite the outlined differences among the Cholesky factors, the induced covariance matrices exhibit rather similar heatmaps and eigenvalues (Figures 9 and 10 in Appendix C).

For the quadratic discriminant analysis we have used leave-one-out cross-validation, since the sample size was sufficiently small to allow it. Table 1 contains the results thus obtained. We see that lasso is the method that performs poorest overall. Conversely, band is arguably the best for this problem, except for the TNR of rock samples, which is highest for grad_frob. However, observing the rest of statistics for grad_frob, it can be deduced that this method over-classifies samples as mines: it has the lowest TNR for them. On the other hand, grad_lik performs competitively for this problem, but is in general outperformed or matched by band. Since the sonar behaviour hints at a band structure for the covariance (frequency patterns being related to those close to them), and therefore for its Cholesky factor, the good performance of band could be expected.

TABLE 1. Statistics for the sonar problem. M: mines; R: rocks.

	band	grad_frob	grad_lik	lasso
TNR (M)	**0.78**	0.08	**0.78**	0.62
F1 (M)	**0.8**	0.7	0.55	0.33
TNR (R)	0.79	**0.97**	0.45	0.26
F1 (R)	**0.78**	0.15	0.65	0.5
ACC	**0.79**	0.56	0.61	0.43

2) WALL-FOLLOWING ROBOT NAVIGATION

The other real data set we use is the *Wall-Following Robot Navigation* one. Here a robot moves in a room following the wall clockwise. It contains 5456 observations and 24 variables. Each variable corresponds to the value of an ultrasound sensor, which are arranged circularly over the robot's body. Here the increasing order reflects the reference angle where the sensor is located. Since the robot is moving clockwise, here the classification task is between four possible class values: Move-Forward, Sharp-Right-Turn, Slight-Left-Turn or Slight-Right-Turn.

As in the previous problem, we obtain the Cholesky factors for each of the movements, depicted in Figure 7. We notice that grad_frob outputs a similar matrix (except for the Slight-Left-Turn movement) than the other three methods, which means that the extreme behaviour we observed in the sonar experiment was problem related. By contrast,
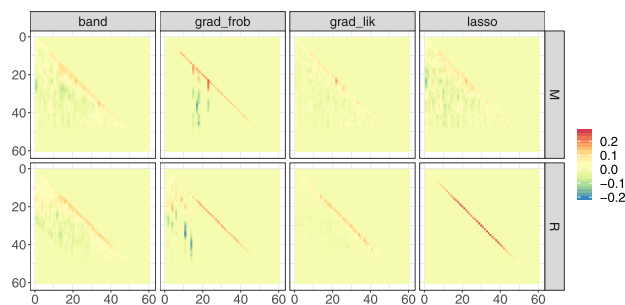

FIGURE 6. Heatmaps of the Cholesky factors of rock and mine samples. M: Mines; R: Rocks.

the Cholesky factor for Slight-Left-Turn is nearly diagonal. The other matrices are rather similar among the methods, with band notably choosing in general a high banding parameter k (few to no bands). Here we have a similar structure as in the sonar problem: we appreciate for all the movements except Slight-Left-Turn that most entries close to the diagonal are positive, whereas distant ones are frequently negative. Regarding Slight-Left-Turn, these matrices are the sparsest and have near zero values on the diagonal. Since the robot is moving clockwise, this movement is related to obstacles, therefore it could hint that sensor readings are correctly identifying them.
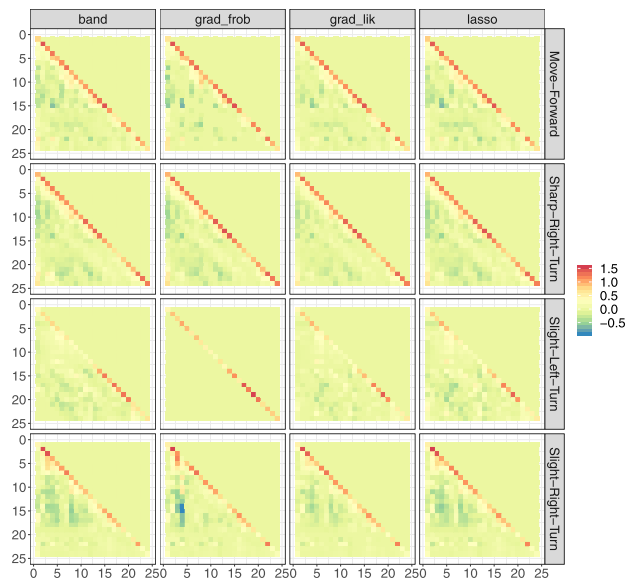
FIGURE 7. Heatmaps of the Cholesky factor of the wall-following robot navigation samples.

In this problem we have a larger sample size, and therefore we split the data into train and test, with half of the samples on each set. The classification results are shown in Table 2. We observe that all of the methods perform arguably good, in fact they achieve nearly identical accuracy. It is noticeable how competitive are lasso and grad_lik, which performed much worse in the sonar problem. We also notice that arguably the best results are obtained for the Slight-Left-Turn movement, which confirms our previous intuition over the heatmaps about sensors correctly identifying obstacles. The worst performance over all methods is for the Slight-Right-Turn movement, but is not noteworthy when compared with the rest (except for Slight-Left-Turn).

C. DISCUSSION

We can draw several conclusions from both the simulated and real experiments. First, we report in Figure 8 the execution time for each method, where it can be seen that grad_lik is the slowest one and band is the fastest.

Whenever there is a clear dependence between variables that are close in the ordering, such as in the sonar example, the band method could be preferred, because it is the one

TABLE 2. Statistics for the robot problem. MF: Move-Forward; SHR: Sharp-Right-Turn; SLL: Slight-Left-Turn; SLR: Slight-Right-Turn.

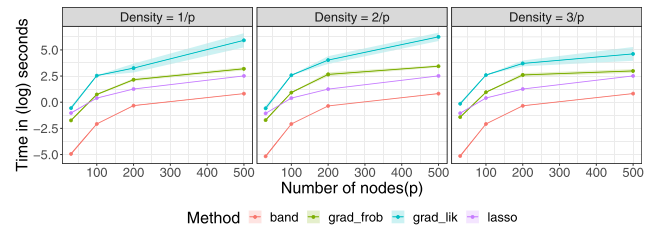|	band	grad_frob	grad_lik	lasso
TNR (MF)	**0.87**	0.86	0.85	0.84
F1 (MF)	**0.64**	**0.64**	0.61	0.62
TNR (SHR)	**0.89**	0.88	0.87	0.85
F1 (SHR)	0.72	0.72	**0.73**	**0.73**
TNR (SLL)	0.96	0.95	**0.98**	**0.98**
F1 (SLL)	0.67	0.65	**0.72**	0.7
TNR (SLR)	0.82	0.83	0.81	**0.84**
F1 (SLR)	0.56	**0.57**	0.55	**0.57**
ACC	**0.66**	**0.66**	0.65	**0.66**

FIGURE 8. Logarithm of the execution time (in seconds) for each of the methods under evaluation. Density indicates the average proportion of lower triangular non-zero entries in the simulated T Cholesky factors.

that more naturally approximates the structure induced in the Cholesky factor (as happened in the sonar example).

Our new proposed method grad_frob has shown to be competitive both in execution time as well as recovery results: when interested in model selection, that is, how accurately zeros in the Cholesky factor are estimated, it yields the best results. Conversely, grad_lik has shown to be the most robust: in simulations it achieved reasonable performance even when the true covariance matrix was dense, and it also performed competitively in the sonar example, which was mostly suited for band as we discussed.

Finally, lasso has achieved overall poor results, except for the wall-following robot navigation data set. Specially, in simulations it failed to correctly recover the zero pattern in the Cholesky factor and was the numerically farthest away from the true matrix. Despite this, it is the second fastest of the four methods, so when model selection or robustness are not a concern it is a good alternative to band, since it provides more flexibility over the zero pattern in the Cholesky factor.

VI. CONCLUSION AND FUTURE WORK

In this paper we have proposed a sparse model for the Cholesky factor of the covariance matrix of a multivariate Gaussian distribution. Few other works in the literature have previously addressed this problem, and we expand them in many ways. We have formalised the extension of an arbitrary zero pattern in the Cholesky factorization to a Gaussian graphical model. We have proposed a novel estimation method that directly optimizes a penalized matrix loss, and we have compared it with the other already existing regression-based approaches in different simulation and real experiments. We have finally discussed which estimation

method is preferable depending on the problem characteristics, and argue how our estimation proposal is preferable in many scenarios than those already existing.

As future research, the most direct and interesting derivative work would be to further analyse, both theoretically and empirically, the Gaussian graphical model extension to unordered variables of the sparse covariance Cholesky factorization model. Also in this direction, its relationship with the already established covariance graph [14]–[17], an undirected graphical model over the covariance matrix, could yield interesting results. Regarding our novel estimation method, we could explore other alternatives to solve the optimization problems that arise from the two losses that we compute in Appendix B. For large data sets, the use of deep learning models to find the hidden Cholesky factor could be also explored.
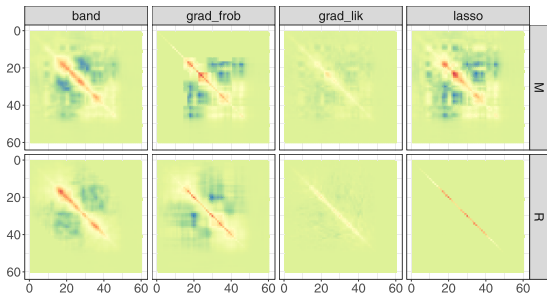

FIGURE 9. Heatmaps of the covariance matrix of rock and mine samples. M: Mines; R: Rocks.

APPENDIX

A. A TRANSFORMATION BETWEEN THE CHOLESKY FACTORIZATIONS OF Σ AND Ω

The following result gives how to obtain the elements of $\mathbf{L}$ from those in $\mathbf{U}$. Recall that, since $\mathbf{D}$ is diagonal, the (i, i) entry in $\mathbf{D}^{-1}$ is d_{ii}^{-1}.

Proposition 1: For each $i \in \{1, \ldots, p\}$ with $j \in \{1, \ldots, i-1\}$ the following identity holds:

$$\beta_{ij|1,\ldots,j} = \beta_{ij|1,\ldots,i-1} + \sum_{k=j+1}^{i-1} \beta_{ik|1,\ldots,i-1}\beta_{kj|1,\ldots,j}.$$

Proof: First we recall that $\mathbf{L} = \mathbf{U}^{-t}$ and $\mathbf{U}\mathbf{U}^{-1} = I_p$. Thus, for each $i \in \{2, \ldots, p\}$ and each $j < i$, if we multiply the i-th row in $\mathbf{U}$ with the j-th column of $\mathbf{U}^{-1}$, this must be equal to 0. Specifically, we obtain the following equation,

$$\begin{aligned}0 = {}& -\beta_{ij|1,\ldots,i-1} \\ & -\beta_{ij+1|1,\ldots,i-1}\beta_{j+1j|1,\ldots,j} \\ & \vdots \\ & -\beta_{ii-1|1,\ldots,i-1}\beta_{i-1j|1,\ldots,j} \\ & +\beta_{ij|1,\ldots,j}.\end{aligned}$$
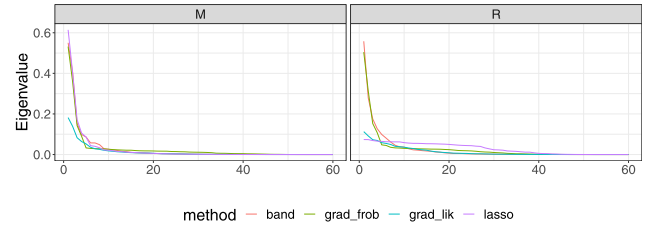

FIGURE 10. Scree plot of the covariance matrices of rock and mine samples. M: Mines; R: Rocks.
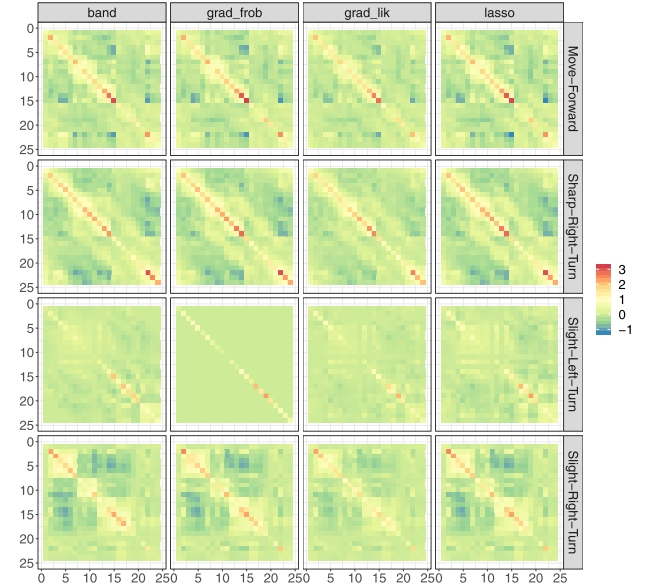

FIGURE 11. Heatmaps of the covariance matrix of the wall-following robot navigation samples.
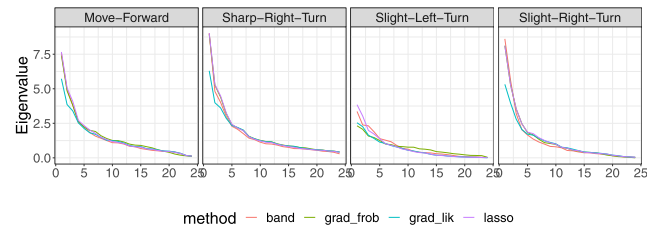

FIGURE 12. Scree plot of the covariance matrices of the wall-following robot navigation samples.

Therefore,

$$0 = \beta_{ij|1,\ldots,j} - \sum_{k=j+1}^{i-1} \beta_{ik|1,\ldots,i-1}\beta_{kj|1,\ldots,j} - \beta_{ij|1,\ldots,i-1},$$

which yields the desired result. □

B. PROXIMAL GRADIENT ALGORITHM AND GRADIENT COMPUTATIONS FOR ϕ_{NLL} AND ϕ_{FR}

We will show how to obtain a simplified expression for the gradient of $\phi(\Sigma(\mathbf{T}))$ with respect to $\mathbf{T}$ as a function of the one with respect to Σ. We will denote these gradients as $\nabla_{\mathbf{T}}\phi$ and $\nabla_{\Sigma}\phi$, respectively.

Algorithm 1 Proximal Gradient Algorithm for Minimization of l_1-Penalized Loss

---

Require: ϕ differentiable function over positive definite matrices,

    $\mathbf{T} \in \mathbb{R}^{p \times p}$ lower triangular,

    $M \in \mathbb{N}, \varepsilon, \lambda, \alpha \in (0, 1)$

1: $f = \phi(\mathbf{\Sigma}(\mathbf{T}))$
2: $g = \lambda ||\mathbf{T}||_1$
3: **repeat**
4:     $\mathbf{D} = \nabla_{\mathbf{T}}\phi(\mathbf{\Sigma}(\mathbf{T}))$
5:     $s = 1$
6:     **loop**
7:         $\mathbf{T}' = \mathbf{T} - s\mathbf{D}$
8:         soft thresholding $\mathbf{T}'$ at level $s\lambda$
9:         $f' = \phi(\mathbf{\Sigma}(\mathbf{T})), g' = \lambda ||\mathbf{T}||_1$
10:        $v = \frac{1}{2s}(||\mathbf{T} - \mathbf{T}'||_F^2) + \text{tr}((\mathbf{T}' - \mathbf{T})\mathbf{D})$
11:        **if** $f' + g' \leq f + g$ **and** $f' \leq f + v$ **then**
12:            **break**
13:        **else**
14:            $s = \alpha s$
15:        **end if**
16:     **end loop**
17:     $\delta = (f + g - f' - g')$
18:     $\mathbf{T} = \mathbf{T}', f = f', g = g'$
19: **until** $k > M$ **or** $\delta < \varepsilon$
Ensure: T

Proposition 2: For any differentiable loss function $\phi(\mathbf{\Sigma}(\mathbf{T}))$,

$$\nabla_{\mathbf{T}}\phi = 2\nabla_{\mathbf{\Sigma}}\phi\mathbf{T}. \tag{20}$$

Proof: This proof follows some ideas from [21, Proposition 2.1]. By matrix calculus [39], we have the following gradient expression for $j < i$,

$$\frac{\partial \phi(\mathbf{\Sigma}(\mathbf{T}))}{\partial t_{ij}} = tr\left(\nabla_{\mathbf{\Sigma}}\phi \frac{\partial \mathbf{\Sigma}(\mathbf{T})}{\partial t_{ij}}\right), \tag{21}$$

where we have used that $\nabla_{\mathbf{\Sigma}}\phi$ is symmetric. Furthermore, note that [39]

$$\frac{\partial \mathbf{\Sigma}(\mathbf{T})}{\partial t_{ij}} = \frac{\partial \mathbf{T}\mathbf{T}^t}{\partial t_{ij}} = \mathbf{T}\mathbf{E}^{ij} + \mathbf{E}^{ji}\mathbf{T}^t,$$

where $\mathbf{E}^{ij}$ ($\mathbf{E}^{ji}$) has its (i, j) $((j, i))$ entry equal to one and zero elsewhere. Then from Equation (21) we have

$$\begin{aligned}\frac{\partial \phi(\mathbf{\Sigma}(\mathbf{T}))}{\partial t_{ij}} &= \text{tr}\left(\nabla_{\mathbf{\Sigma}}\phi(\mathbf{T}\mathbf{E}^{ij} + \mathbf{E}^{ji}\mathbf{T}^t)\right) \\ &= \text{tr}\left(\nabla_{\mathbf{\Sigma}}\phi\mathbf{T}\mathbf{E}^{ij}\right) + \text{tr}\left(\nabla_{\mathbf{\Sigma}}\phi\mathbf{E}^{ji}\mathbf{T}^t\right) \\ &= \text{tr}\left(\mathbf{E}^{ji}\mathbf{T}^t\nabla_{\mathbf{\Sigma}}\phi\right) + \text{tr}\left(\mathbf{E}^{ji}\mathbf{T}^t\nabla_{\mathbf{\Sigma}}\phi\right) \\ &= 2\,\text{tr}\left(\mathbf{E}^{ji}\mathbf{T}^t\nabla_{\mathbf{\Sigma}}\phi\right).\end{aligned}$$

Since $a_{ji} = \text{tr}(\mathbf{E}^{ji}\mathbf{A})$ for any matrix $\mathbf{A}$, we have the desired result. □

The above proposition implies that once a loss function $\phi(\mathbf{\Sigma}(\mathbf{T}))$ is fixed, it is only necessary to compute $\nabla_{\mathbf{\Sigma}}\phi$ in order to obtain $\nabla_{\mathbf{T}}\phi$. We would then use it for the proximal gradient method (Algorithm 1).

We thus can easily obtain the gradient for ϕ_{NLL} and ϕ_{FR}, which we consider in this work. Standard matrix calculus [39] gives $\nabla_{\mathbf{\Sigma}}\phi_{NLL} = \mathbf{\Sigma}^{-1} - \mathbf{\Sigma}^{-1}\hat{\mathbf{\Sigma}}\mathbf{\Sigma}^{-1}$. Therefore we have

$$\begin{aligned}\nabla_{\mathbf{T}}\phi_{NLL} &= 2\nabla_{\mathbf{\Sigma}}\phi_{NLL}\mathbf{T} \\ &= 2\mathbf{\Sigma}^{-1}(\mathbf{T})(\mathbf{I}_p - \hat{\mathbf{\Sigma}}\mathbf{\Sigma}^{-1}(\mathbf{T}))\mathbf{T} \\ &= 2\mathbf{T}^{-t}(\mathbf{I}_p - \mathbf{T}^{-1}\hat{\mathbf{\Sigma}}\mathbf{T}^{-t}).\end{aligned}$$

Conversely, from [39] we have that $\nabla_{\mathbf{\Sigma}}\phi_{FR}(\mathbf{\Sigma}) = 2(\mathbf{\Sigma} - \hat{\mathbf{\Sigma}})$. Thus $\nabla_{\mathbf{T}}\phi_{FR} = 2(\mathbf{T}\mathbf{T}^t - \hat{\mathbf{\Sigma}})\mathbf{T}$.

### C. MORE FIGURES FOR REAL DATA EXPERIMENTS
See Figure 9–12.

REFERENCES
[1] P. Bühlmann and S. van de Geer, *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer, 2011.
[2] M. Maathuis, M. Drton, S. Lauritzen, and M. Wainwright, Eds., *Handbook of Graphical Models*. Boca Raton, FL, USA: CRC Press, 2018.
[3] T. Li, C. Qian, E. Levina, and J. Zhu, "High-dimensional Gaussian graphical models on network-linked data," *J. Mach. Learn. Res.*, vol. 21, no. 74, pp. 1–45, 2020.
[4] A. P. Dempster, "Covariance selection," *Biometrics*, vol. 28, no. 1, pp. 157–175, Mar. 1972.
[5] M. Yuan and Y. Lin, "Model selection and estimation in the Gaussian graphical model," *Biometrika*, vol. 94, no. 1, pp. 19–35, Feb. 2007.
[6] S. Wright, "The method of path coefficients," *Ann. Math. Statist.*, vol. 5, no. 3, pp. 161–215, Sep. 1934.
[7] N. Wermuth, "Linear recursive equations, covariance selection, and path analysis," *J. Amer. Stat. Assoc.*, vol. 75, no. 372, pp. 963–972, Dec. 1980.
[8] S. van de Geer and P. Bühlmann, "ℓ_0-penalized maximum likelihood for sparse directed acyclic graphs," *Ann. Statist.*, vol. 41, no. 2, pp. 536–567, 2013.
[9] M. Pourahmadi, "Joint mean-covariance models with applications to longitudinal data: Unconstrained parameterisation," *Biometrika*, vol. 86, no. 3, pp. 677–690, Sep. 1999.
[10] A. d'Aspremont, O. Banerjee, and L. El Ghaoui, "First-order methods for sparse covariance selection," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 1, pp. 56–66, Jan. 2008.
[11] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432–441, Jul. 2008.
[12] A. J. Rothman, P. J. Bickel, E. Levina, and J. Zhu, "Sparse permutation invariant covariance estimation," *Electron. J. Statist.*, vol. 2, pp. 494–515, 2008.
[13] I. Córdoba, C. Bielza, and P. Larrañaga, "A review of Gaussian Markov models for conditional independence," *J. Stat. Planning Inference*, vol. 206, pp. 127–144, May 2020.
[14] G. Kauermann, "On a dualization of graphical Gaussian models," *Scand. J. Statist.*, vol. 23, no. 1, pp. 105–116, 1996.
[15] D. R. Cox and N. Wermuth, "Linear dependencies represented by chain graphs," *Stat. Sci.*, vol. 8, no. 3, pp. 204–218, Aug. 1993.
[16] S. Chaudhuri, M. Drton, and T. S. Richardson, "Estimation of a covariance matrix with zeros," *Biometrika*, vol. 94, no. 1, pp. 199–216, Feb. 2007.
[17] J. Bien and R. J. Tibshirani, "Sparse estimation of a covariance matrix," *Biometrika*, vol. 98, no. 4, pp. 807–820, Dec. 2011.
[18] N. Wermuth, D. R. Cox, and G. M. Marchetti, "Covariance chains," *Bernoulli*, vol. 12, no. 5, pp. 841–862, Oct. 2006.

[19] A. J. Rothman, E. Levina, and J. Zhu, "A new approach to Cholesky-based covariance regularization in high dimensions," *Biometrika*, vol. 97, no. 3, pp. 539–550, Sep. 2010.

[20] A. Das, D. Sexton, C. Lainscsek, S. S. Cash, and T. J. Sejnowski, "Characterizing brain connectivity from human electrocorticography recordings with unobserved inputs during epileptic seizures," *Neural Comput.*, vol. 31, no. 7, pp. 1271–1326, Jul. 2019.

[21] G. Varando and N. R. Hansen, "Graphical continuous Lyapunov models," in *Proc. 36th conference Uncertainty Artif. Intell.*, 2020. [Online]. Available: https://arxiv.org/abs/2005.10483

[22] M. Drton, "Algebraic problems in structural equation modeling," in *Proc. 50th Anniversary Gröbner Bases*. Tokyo, Japan: Mathematical Society of Japan, 2018, pp. 35–86.

[23] A. P. Dempster, *Elements of Continuous Multivariate Analysis*. Reading, MA, USA: Addison-Wesley, 1969.

[24] A. E. Beaton, "The use of special matrix operators in statistical calculus," *ETS Res. Bull. Ser.*, vol. 1964, no. 2, pp. i–222, Dec. 1964.

[25] Q. Ye, A. Amini, and Q. Zhou, "Optimizing regularized cholesky score for order-based learning of Bayesian networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Apr. 27, 2020, doi: 10.1109/TPAMI.2020.2990820.

[26] A. P. Dawid, "Conditional independence in statistical theory," *J. Roy. Stat. Soc., B (Methodol.)*, vol. 41, no. 1, pp. 1–15, 1979.

[27] T. W. Anderson, *An Introduction to Multivariate Statistical Analysis*, 3rd ed. Hoboken, NJ, USA: Wiley, 2003.

[28] V. Chandrasekaran, P. A. Parrilo, and A. S. Willsky, "Latent variable graphical model selection via convex optimization," *Ann. Statist.*, vol. 40, no. 4, pp. 1935–1967, Aug. 2012.

[29] D. Yatsenko, K. Josić, A. S. Ecker, E. Froudarakis, R. J. Cotton, and A. S. Tolias, "Improved estimation and interpretation of correlations in neural circuits," *PLOS Comput. Biol.*, vol. 11, no. 3, 2015, Art. no. e1004083.

[30] M. Zorzi and R. Sepulchre, "AR identification of latent-variable graphical models," *IEEE Trans. Autom. Control*, vol. 61, no. 9, pp. 2327–2340, Sep. 2016.

[31] S. Basu, X. Li, and G. Michailidis, "Low rank and structured modeling of high-dimensional vector autoregressions," *IEEE Trans. Signal Process.*, vol. 67, no. 5, pp. 1207–1222, Mar. 2019.

[32] E. Levina, A. Rothman, and J. Zhu, "Sparse estimation of large covariance matrices via a nested lasso penalty," *Ann. Appl. Statist.*, vol. 2, no. 1, pp. 245–263, Mar. 2008.

[33] X. Zheng, B. Aragam, P. Ravikumar, and E. P. Xing, "DAGs with NO TEARS: Continuous optimization for structure learning," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.* Red Hook, NY, USA: Curran Associates, 2018, pp. 9492–9503.

[34] F. Bach, "Optimization with sparsity-inducing penalties," *Found. Trends Mach. Learn.*, vol. 4, no. 1, pp. 1–106, 2011.

[35] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[36] R. C. Team. (2020). *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing. [Online]. Available: https://www.R-project.org/

[37] I. Córdoba, G. Varando, C. Bielza, and P. Larrañaga, "On generating random Gaussian graphical models," *Int. J. Approx. Reasoning*, vol. 125, pp. 240–250, Oct. 2020.

[38] D. Dua and C. Graff. (2020). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml

[39] K. Petersen and M. Pedersen, *The Matrix Cookbook*. Lyngby, Denmark: Technical Univ. Denmark, 2008.

CONCHA BIELZA received the M.Sc. degree in mathematics from the Universidad Complutense de Madrid, in 1989, and the Ph.D. degree in computer science from the Universidad Politécnica de Madrid (UPM), in 1996 (extraordinary doctorate award). Since 2010, she has been a Full Professor of statistics and operations research with the Departamento de Inteligencia Artificial, UPM. Her research interests are primarily in the areas of probabilistic graphical models, decision analysis, metaheuristics for optimization, machine learning, and real applications, such as biomedicine, bioinformatics, neuroscience, industry 4.0, and sports. She has published over 120 articles in high impact factor journals. She has supervised 17 Ph.D. theses. She has received the 2014 UPM Research Prize.
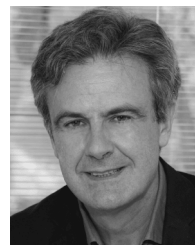
PEDRO LARRAÑAGA received the M.Sc. degree in mathematics (statistics) from the University of Valladolid and the Ph.D. degree in computer science from the University of the Basque Country (excellence award). He has been a Full Professor in computer science and artificial Intelligence with the Universidad Politécnica de Madrid (UPM), since 2007. Before moving to UPM, his academic career developed at the University of the Basque Country (UPV-EHU) at several faculty ranks: an Assistant Professor, from 1985 to 1998, an Associate Professor, from 1998 to 2004, and a Full Professor, from 2004 to 2007. He received the distinction for Full Professor, in 2003. He has published over 200 articles in high-impact factor journals. He has supervised over 30 Ph.D. theses. His research interests include the areas of probabilistic graphical models, metaheuristics for optimization, data mining, classification models, and real applications, such as biomedicine, bioinformatics, neuroscience, industry 4.0, and sports. He is Fellow of the European Association for Artificial Intelligence, since 2012, and a Fellow of the Academia Europaea, since 2018. He has received the 2013 Spanish National Prize in computer science and the prize of the Spanish Association for Artificial Intelligence, in 2018.

IRENE CÓRDOBA received the B.Sc. degree in mathematics and the B.Sc. degree in computer engineering from the Universidad Autónoma de Madrid, in 2014, and the M.Sc. degree in artificial intelligence from the Universidad Politécnica de Madrid (UPM), where she is currently pursuing the Ph.D. degree with the Departamento de Inteligencia Artificial, supported by the Spanish Ministry of Science, Innovation and Universities under the program for university teachers training (FPU).

GHERARDO VARANDO received the Ph.D. degree in artificial intelligence from the Universidad Politécnica de Madrid, in June 2018, under the supervision of Concha Bielza and Pedro Larrañaga. His research is in probability and statistics applied to machine learning problems. He is currently working as a Postdoctoral Researcher at the University of Copenhagen.

• • •