Sparse Cholesky Precision Matrix Estimation

Kingsley Yeon
email yeon@uchicago.edu

March 13, 2025

1 Introduction

In many multivariate settings, we observe data $u_1, u_2, \ldots, u_m \in \mathbb{R}^n$ drawn independently from a Gaussian distribution $\mathcal{N}(0, \Sigma)$. Our objective is to estimate the precision matrix $P = \Sigma^{-1}$. A common approach is to reparameterize P via its Cholesky factorization, namely,

$$P = LL^T,$$

where L is a lower–triangular matrix with strictly positive diagonal entries. This parametrization not only guarantees positive definiteness but also simplifies the likelihood function.

The log-likelihood function for the Gaussian model is given by

$$\ell(P) = \frac{m}{2} \log \det(P) - \frac{1}{2} \sum_{i=1}^{m} u_i^T P u_i.$$

Substituting $P = LL^T$ and noting that $\det(P) = (\det(L))^2 = \prod_{j=1}^{n} L_{jj}^2$, we have

$$\log \det(P) = 2 \sum_{j=1}^{n} \log L_{jj}.$$

Thus, the log-likelihood becomes

$$\ell(L) = m \sum_{j=1}^{n} \log L_{jj} - \frac{1}{2} \sum_{i=1}^{m} u_i^T L L^T u_i.$$

Since maximizing the likelihood is equivalent to minimizing the negative log-likelihood, we define the loss function as

$$f(L) = -\ell(L) = -m \sum_{j=1}^{n} \log L_{jj} + \frac{1}{2} \sum_{i=1}^{m} u_i^T L L^T u_i.$$

Exploiting the associativity of matrix multiplication, we can rewrite the quadratic term as

$$u_i^T L L^T u_i = \|L^T u_i\|^2 = \sum_{j=1}^{n} (l_j^T u_i)^2,$$

where l_j denotes the jth column of L. We obtain the loss function in the form

$$f(L) = \sum_{j=1}^{n} \left[-m \log L_{jj} + \frac{1}{2} \sum_{i=1}^{m} (l_j^T u_i)^2 \right].$$

This formulation is central to our maximum likelihood estimation procedure, as it converts the problem of estimating the precision matrix P into an unconstrained optimization problem over the entries of the Cholesky factor L.

2 Optimization and Delta method for asymptotics

We begin with the loss

$$f(L) = \sum_{j=1}^{n} \left[-m \log L_{jj} + \frac{1}{2} \sum_{i=1}^{m} \left(l_j^T u_i \right)^2 \right],$$

where for each $j = 1, \ldots, n$ the jth column l_j of the lower-triangular matrix L (with $L_{jj} > 0$) parameterizes the precision matrix via

$$P = LL^T,$$

and the observations u_i are drawn from a Gaussian distribution $N(0, \Sigma)$ with $\Sigma = P^{-1}$.

For a fixed column j, the loss contribution is

$$f_j(l_j) = -m \log L_{jj} + \frac{1}{2} \sum_{i=1}^{m} \left(l_j^T u_i \right)^2.$$

Its gradient (with respect to the free parameters in l_j) is

$$g_j(l_j) = -\frac{m}{L_{jj}} e_j + \sum_{i=1}^{m} (l_j^T u_i)\, u_i,$$

where e_j is the unit vector with 1 in the jth coordinate.

Taking expectations (and noting that the u_i are i.i.d.) we have

$$E\big[g_j(l_{0,j})\big] = -\frac{m}{L_{0,jj}} e_j + m\, E\big[(l_{0,j}^T u) u\big].$$

Because the covariance of u is Σ, we obtain

$$E\big[(l_{0,j}^T u_i) u_i\big] = E\big[u_i\, u_i^T\big]\, l_{0,j} = \Sigma\, l_{0,j}.$$

Setting the expected score to zero gives

$$-\frac{m}{L_{0,jj}} e_j + m\, \Sigma\, l_{0,j} = 0 \quad \Longrightarrow \quad \Sigma\, l_{0,j} = \frac{1}{L_{0,jj}} e_j.$$

Now, to see the implication for the quadratic form $l_{0,j}^T \Sigma\, l_{0,j}$, we multiply both sides on the left by $l_{0,j}^T$:

$$l_{0,j}^T \Sigma\, l_{0,j} = \frac{1}{L_{0,jj}} l_{0,j}^T e_j.$$

Since L is lower–triangular, its jth column $l_{0,j}$ has $L_{0,jj}$ as its jth (first nonzero) entry, so that

$$l_{0,j}^T e_j = L_{0,jj}.$$

Thus,

$$l_{0,j}^T \Sigma\, l_{0,j} = \frac{L_{0,jj}}{L_{0,jj}} = 1.$$

Define the per-observation score contribution as

$$s_i = (l_{0,j}^T u_i) u_i.$$

Then the full score is

$$g_j(l_{0,j}) = \sum_{i=1}^{m} \Big[s_i - E[s_i] \Big],$$

with

$$E[s_i] = \Sigma\, l_{0,j} = \frac{1}{L_{0,jj}}\, e_j.$$

By the central limit theorem, the aggregated (centered) score satisfies

$$\sqrt{m}\, \frac{1}{m}\, g_j(l_{0,j}) \xrightarrow{d} N(0, \Omega_j),$$

with

$$\Omega_j = \mathrm{Var}\big(s_i\big) = E\big[(l_{0,j}^T u_i)^2\, u_i u_i^T\big] - \big(\Sigma\, l_{0,j}\big)\big(\Sigma\, l_{0,j}\big)^T.$$

Using Isserlis' (or Wick's) theorem for Gaussian random vectors, one can show that

$$E\big[(l_{0,j}^T u_i)^2\, u_i u_i^T\big] = \Big(l_{0,j}^T \Sigma\, l_{0,j}\Big)\Sigma + 2\,\Sigma\, l_{0,j} l_{0,j}^T \Sigma.$$

Because the identification condition gives $l_{0,j}^T \Sigma\, l_{0,j} = 1$ and since

$$\Sigma\, l_{0,j} l_{0,j}^T \Sigma = \frac{1}{L_{0,jj}^2}\, e_j e_j^T,$$

we deduce that

$$\Omega_j = \Sigma + 2\, \frac{1}{L_{0,jj}^2}\, e_j e_j^T - \frac{1}{L_{0,jj}^2}\, e_j e_j^T = \Sigma + \frac{1}{L_{0,jj}^2}\, e_j e_j^T.$$

On the other hand, the Hessian (second derivative) of the per–observation loss for column j is computed by differentiating the score. The quadratic term contributes

$$\frac{\partial^2}{\partial l_j \partial l_j^T}\, \frac{1}{2} (l_j^T u_i)^2 = u_i u_i^T,$$

so that summing over i and adding the contribution from the $-m \log L_{jj}$ term (whose second derivative is $\frac{m}{L_{jj}^2}\, e_j e_j^T$) we have

$$H_j = \sum_{i=1}^{m} u_i u_i^T + \frac{m}{L_{jj}^2}\, e_j e_j^T.$$

Dividing by m (to obtain the per–observation Hessian) yields

$$\mathcal{H}_j = \Sigma + \frac{1}{L_{0,jj}^2}\, e_j e_j^T.$$

Notice that this is identical to Ω_j.

Now, by a first–order Taylor expansion (the delta method), if $\hat{l}_j$ is an estimator of $l_{0,j}$ that (approximately) satisfies

$$g_j(\hat{l}_j) = 0 \quad \text{and} \quad g_j(\hat{l}_j) \approx g_j(l_{0,j}) + H_j\, (\hat{l}_j - l_{0,j}),$$

then

$$\hat{l}_j - l_{0,j} \approx -H_j^{-1}\, g_j(l_{0,j}).$$

Since

$$\sqrt{m}\,\frac{1}{m}\, g_j(l_{0,j}) \xrightarrow{d} N(0, \Omega_j),$$

it follows that by the delta method

$$\sqrt{m}\,(\hat{l}_j - l_{0,j}) \xrightarrow{d} N\Big(0,\, H_j^{-1}\,\Omega_j\, H_j^{-1}\Big).$$

Because $H_j/m \to \mathcal{H}_j$ and we have shown that $\Omega_j = \mathcal{H}_j$, the asymptotic covariance simplifies to

$$\sqrt{m}\,(\hat{l}_j - l_{0,j}) \xrightarrow{d} N\Big(0,\, \mathcal{H}_j^{-1}\Big)$$

or, equivalently,

$$\hat{l}_j - l_{0,j} = O_P\Big(\frac{1}{\sqrt{m}}\Big) \quad \text{with} \quad \mathrm{Var}\big(\sqrt{m}\,(\hat{l}_j - l_{0,j})\big) = \mathcal{H}_j^{-1}.$$

Therefore, using the delta method we conclude that each column l_j of the Cholesky factor L is estimated at a $\sqrt{m}$–rate, with the asymptotic distribution

$$\sqrt{m}\Big(\hat{l}_j - l_{0,j}\Big) \xrightarrow{d} N\left(0, \Big(\Sigma + \frac{1}{L_{0,jj}^2}\, e_j e_j^T\Big)^{-1}\right).$$

Because the per–observation Hessian $\mathcal{H}_j = \Sigma + \frac{1}{L_{0,jj}^2}\, e_j e_j^T$ matches the asymptotic covariance of the score, the asymptotic variance of $\hat{l}_j$ is given by the inverse of this matrix. This result provides a basis for constructing confidence intervals and hypothesis tests for the entries of L in large samples.

2.1 Result

In our simulation, we first generate a 15-dimensional precision matrix using a Gaussian kernel, compute its Cholesky factor L_{true}, and obtain the corresponding covariance matrix Σ. We then focus on estimating the second column ($j = 1$) of L. For that column, only the entries from the 2nd row onward (i.e. L_{jj} and below) are free parameters. Using a Newton method that minimizes the loss

$$f(L) = \sum_{j=1}^{n} \left[-m \log L_{jj} + \frac{1}{2} \sum_{i=1}^{m} \big(l_j^T u_i\big)^2 \right],$$

we estimate the free parameter vector z for this column based on $m = 10{,}000$ samples drawn from $N(0, \Sigma)$. Repeating the experiment over 500 Monte Carlo replications, we compute the scaled estimation error $\sqrt{m}(\hat{z} - z_0)$. The delta method predicts that this scaled error is asymptotically normal with covariance matrix

$$V_{\text{pred}} = \mathcal{H}^{-1},$$

where

$$\mathcal{H} = \Sigma_{\text{sub}} + \frac{1}{L_{0,jj}^2} e_1 e_1^T.$$

Figure 1 displays a set of subplots—one for each coordinate of the estimated vector z. In each subplot, the histogram shows the empirical distribution of the scaled error for that coordinate, and the red curve overlays the theoretical normal density using the corresponding predicted standard deviation. The close agreement between the histograms and the overlaid curves provides visual evidence that our simulation results align with the asymptotic distribution derived via the delta method.

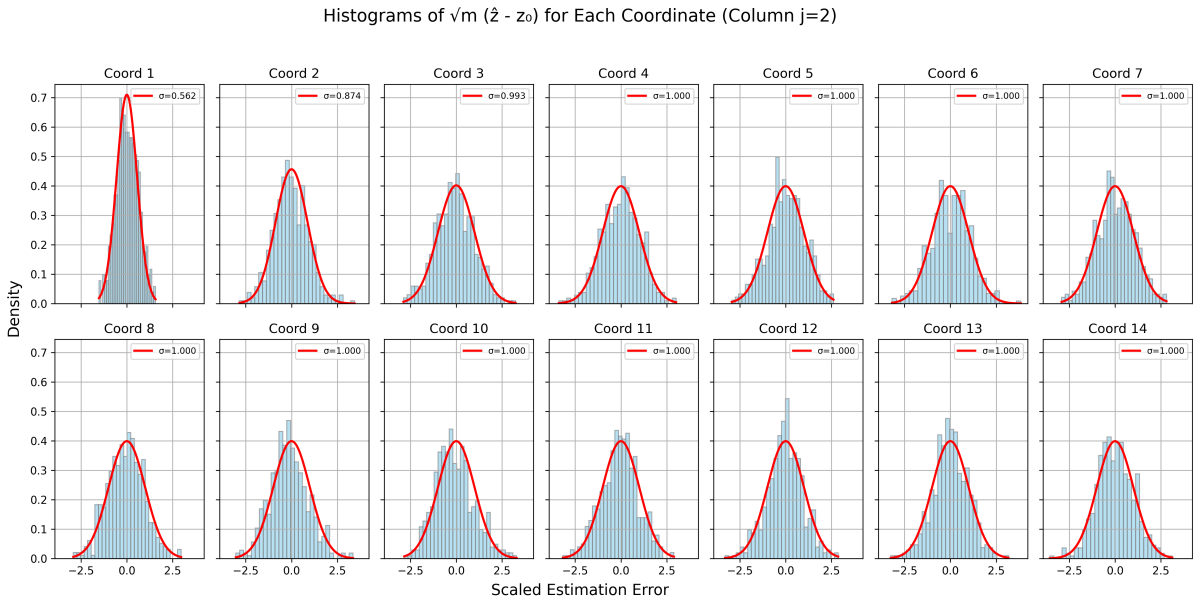

Figure 1: Histograms of the scaled estimation errors, $\sqrt{m}(\hat{z} - z_0)$. The simulation is based on 500 Monte Carlo replications with a $m = 10\,000$ sample size per trial.

3 Sparsity Pattern

To find an appropriate sparsity pattern for the Cholesky factorization of a precision matrix, one effective strategy is to first order the variables using a reverse-maximin criterion, whereby each variable is sequentially selected based on being maximally distant from those already chosen, thereby capturing the intrinsic geometric structure of the data and assigning each variable a local length scale. The sparsity pattern is then determined by retaining only those entries in the lower-triangular factor for which the distance between corresponding data points is less than a threshold proportional to the local length scale, modulated by a tuning parameter. This method leverages the screening effect typical of many kernel matrices, ensuring that only the most significant conditional dependencies are preserved. For a comprehensive treatment of this sparsity selection process, including theoretical error bounds and computational complexity considerations, refer to [1].

4 Computational Complexity

Assume that for each column j the number of nonzero entries is roughly d (with $d \ll n$). Under this sparsity assumption, the computational cost per column is significantly reduced compared to the dense case. For each observation u_i, computing the inner product $l_j^T u_i$ involves only $O(d)$ operations rather than $O(n)$. Consequently, evaluating the gradient over m observations costs $O(m\,d)$ per column, and when accounting for the additional multiplications inherent in forming the gradient terms, the cost becomes $O(m\,d^2)$.

The Hessian is assembled by accumulating contributions from each observation in the d-dimensional subspace, incurring $O(m\,d^2)$ operations. In each Newton iteration, the subsequent update step involves solving a linear system of size $d \times d$, which typically requires $O(d^3)$ operations. Thus, each Newton iteration for a single column has a complexity of

$$O\big(m\,d^2 + d^3\big).$$

Since the algorithm proceeds column–wise for $j = 1, \ldots, n$, the total cost per full sweep (i.e., one Newton iteration for every column) is

$$\sum_{j=1}^{n} O\big(m\,d^2 + d^3\big) = O\big(m\,n\,d^2 + n\,d^3\big).$$

Let T denote the number of Newton iterations per column. In our experiments, Newton's method converges very rapidly—typically within 3–5 iterations—so that T is effectively a small constant that can be omitted from the asymptotic expression. Accordingly, the overall computational complexity of the algorithm is

$$O\Big(T\big(m\,n\,d^2 + n\,d^3\big)\Big) \approx O\big(m\,n\,d^2 + n\,d^3\big).$$

An additional advantage of the proposed method is that the algorithm is inherently parallelizable in n, the number of columns (or data dimensions). Since the computations for each column of the Cholesky factor L are independent, the cost associated with processing n columns can be distributed across multiple processors. This parallelism is particularly beneficial when n is large, as it allows the per-iteration computational cost to be reduced to that of processing a single column.

In practice, the hierarchy $m \gg n \gg d$ holds: m represents the number of observations and is very large compared to n, the full data dimension, while n is substantially larger than d, the effective sparsity level per column. Under these conditions, the dominant term in the computational complexity is $m\,n\,d^2$, which is effectively linear in m. Moreover, since the algorithm is parallelizable in n, the cost per processor primarily scales with m, making the method highly scalable even in high-dimensional settings.

References

[1] Florian Schafer, Matthias Katzfuss, and Houman Owhadi. Sparse cholesky factorization by kullback–leibler minimization. *SIAM Journal on scientific computing*, 43(3):A2019–A2046, 2021.