Algorithm 2.7 Sampling from $\pi(\boldsymbol{x}|\boldsymbol{e})$ where $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{Q}^{-1})$ and $\boldsymbol{e}|\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{x}, \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}})$

1: Compute the Cholesky factorization, $\boldsymbol{Q} = \boldsymbol{L}\boldsymbol{L}^T$
2: Sample $\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$
3: Solve $\boldsymbol{L}^T\boldsymbol{v} = \boldsymbol{z}$
4: Compute $\boldsymbol{x} = \boldsymbol{\mu} + \boldsymbol{v}$
5: Compute $\boldsymbol{V}_{n\times k} = \boldsymbol{Q}^{-1}\boldsymbol{A}^T$ using Algorithm 2.2 using $\boldsymbol{L}$ from step 1
6: Compute $\boldsymbol{W}_{k\times k} = \boldsymbol{A}\boldsymbol{V} + \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}$
7: Compute $\boldsymbol{U}_{k\times n} = \boldsymbol{W}^{-1}\boldsymbol{V}^T$ using Algorithm 2.2
8: Sample $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{e}, \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}})$ using Algorithm 2.3.
9: Compute $\boldsymbol{c} = \boldsymbol{A}\boldsymbol{x} - \boldsymbol{\epsilon}$
10: Compute $\boldsymbol{x}^* = \boldsymbol{x} - \boldsymbol{U}^T\boldsymbol{c}$
11: **Return** $\boldsymbol{x}^*$

the log density. Note that all Cholesky triangles required to evaluate the log density are already computed in Algorithm 2.7.

The stochastic version of Example 2.3 now follows.

Example 2.4 *Let $x_1, \ldots, x_n$ be independent normal variables with variance σ_i^2 and mean μ_i. We now observe $e \sim \mathcal{N}(\sum_i x_i, \sigma_\epsilon^2)$. To sample from $\pi(\boldsymbol{x}|e)$, we sample $x_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$, unconditionally, for $i = 1, \ldots, n$ while we condition on $\epsilon \sim \mathcal{N}(e, \sigma_\epsilon^2)$. A conditional sample $\boldsymbol{x}^*$ is then*

$$x_i^* = x_i - c\,\sigma_i^2, \quad \text{where} \quad c = \frac{\sum_j x_j - \epsilon}{\sum_j \sigma_j^2 + \sigma_\epsilon^2}.$$

We can merge soft and hard constraints into one framework if we allow $\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}}$ to be SPSD, but we have chosen not to, as the details are somewhat tedious.

2.4 Numerical methods for sparse matrices

This section will give a brief introduction to numerical methods for sparse matrices. During our discussion of simulation algorithms for GMRFs, we have shown that they all can be expressed such that the main tasks are to

1. Compute the Cholesky factorization of $\boldsymbol{Q} = \boldsymbol{L}\boldsymbol{L}^T$ where $\boldsymbol{Q} > 0$ is sparse, and
2. Solve $\boldsymbol{L}\boldsymbol{v} = \boldsymbol{b}$ and $\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{z}$

The second task is faster to compute than the first, but sparsity of $\boldsymbol{Q}$ is also advantageous in this case. We restrict the discussion to sparse Cholesky factorizations but the ideas also apply to sparse $\boldsymbol{LU}$ factorizations for non symmetric matrices.