Mathematical Statistics
Stockholm University

On the precision matrix of an irregularly sampled AR(1) process

Benjamin Allévius

Research Report 2018:1

ISSN 1650-0377

Postal address:

Mathematical Statistics
Dept. of Mathematics
Stockholm University
SE-106 91 Stockholm
Sweden

Internet:

http://www.math.su.se

Mathematical Statistics
Stockholm University
Research Report **2018:1**,
http://www.math.su.se

On the precision matrix of an irregularly sampled AR(1) process

Benjamin Allévius*

January 2018

Abstract

This text presents an analytical expression for the inverse covariance matrix of a stationary AR(1) process with Gaussian errors, sampled with irregular spacing. Due to the sparse form of this matrix, considerable improvement in the computational cost of density evaluation and random number generation (both unconditional and conditional) can be made, and these points are discussed as well.

Keywords: AR(1) process, time series, precision matrix, missing data.

*Stockholm University, Sweden; benjak@math.su.se

1 Introduction

A zero-mean autoregressive process of order one can be expressed as

$$X_t = \rho X_{t-1} + \epsilon_t, \quad t = 2, 3, \ldots. \tag{1}$$

Here, we will assume normally distributed errors $\epsilon_t \overset{\text{iid}}{\sim} \text{Normal}(0, \sigma^2)$, and make the restriction $|\rho| < 1$. We consider the zero-mean AR(1) process here because a mean term can always be added back later.

With these assumptions, the stationary distribution of the process is normal with mean 0 and variance $\sigma^2/(1-\rho^2)$. As shown in Lindsey (2004, p. 217), if we assume that the marginal distribution of X_1 is the stationary one, the joint distribution of $\tilde{\boldsymbol{x}} = (X_1, X_2, \ldots, X_n)'$ is multivariate normal with the zero vector as mean and a covariance matrix given by

$$\tilde{\boldsymbol{\Sigma}} = \frac{\sigma^2}{1-\rho^2} \begin{bmatrix} 1 & \rho & \ldots & \rho^{n-2} & \rho^{n-1} \\ \rho & 1 & \ldots & \rho^{n-3} & \rho^{n-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \rho^{n-2} & \rho^{n-3} & \ldots & 1 & \rho \\ \rho^{n-1} & \rho^{n-2} & \ldots & \rho & 1 \end{bmatrix}. \tag{2}$$

Further, the precision matrix $\tilde{\boldsymbol{Q}} = \tilde{\boldsymbol{\Sigma}}^{-1}$ is tridiagonal and may be expressed as

$$\tilde{\boldsymbol{Q}} = \frac{1}{\sigma^2} \begin{bmatrix} 1 & -\rho & \ldots & 0 & 0 \\ -\rho & 1+\rho^2 & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & 1+\rho^2 & -\rho \\ 0 & 0 & \ldots & -\rho & 1 \end{bmatrix}. \tag{3}$$

As shown in Rue and Held (2005), the sparsity of the precision matrix $\tilde{\boldsymbol{Q}}$ carries over to its Cholesky decomposition, which enables extremely fast (linear in n) density evaluation and random number generation. In comparison, a Cholesky or eigendecomposition of the (dense) covariance matrix has a computational complexity of order n^3.

In some applications it may be impossible to sample a process modeled by Equation (1) at consecutive time points. In such circumstances, it may still be of interest have a computationally convenient representation of the distribution of the sample $\boldsymbol{x} = (X_{t_1}, X_{t_2}, \ldots, X_{t_m})'$, where $1 \leq t_1 < t_2 < \ldots < t_m \leq n$ and $t_{i+1} - t_i \geq 1$. For this irregularly sampled stationary AR(1) process, we see from the expression of $\tilde{\boldsymbol{\Sigma}}$ in Equation (2) that the

(marginal) distribution of $\boldsymbol{x}$ will also be multivariate normal, with the zero vector (of length m) as mean, and a covariance matrix given by

$$\boldsymbol{\Sigma} = \frac{\sigma^2}{1-\rho^2}\begin{bmatrix} 1 & \rho^{t_2-t_1} & \dots & \rho^{t_{m-1}-t_1} & \rho^{t_m-t_1} \\ \rho^{t_2-t_1} & 1 & \dots & \rho^{t_{m-1}-t_2} & \rho^{t_m-t_2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \rho^{t_{m-1}-t_1} & \rho^{t_{m-1}-t_2} & \dots & 1 & \rho^{t_m-t_{m-1}} \\ \rho^{t_m-t_1} & \rho^{t_m-t_2} & \dots & \rho^{t_m-t_{m-1}} & 1 \end{bmatrix}. \tag{4}$$

That is, element (i, j) of this matrix is given by $\Sigma_{ij} = \frac{\sigma^2}{1-\rho^2}\rho^{|t_i-t_j|}$. The aim is now to find an expression for $\boldsymbol{Q} = \boldsymbol{\Sigma}^{-1}$ similar in neatness to $\tilde{\boldsymbol{Q}}$, with implications for density evaluation and simulation of the irregularly sampled process $\boldsymbol{x}$.

2 Results

Theorem 1. *Let $\boldsymbol{x} = (X_{t_1}, X_{t_2}, \dots, X_{t_m})'$ be the values of the AR(1) process described in Equation (1), sampled at times $t_1 < t_2 < \dots < t_m$, where $t_{i+1} - t_i \geq 1$. Assume the process is in its stationary state, and for brevity of notation, assume $\sigma = 1$. The precision matrix $\boldsymbol{Q}$ of $\boldsymbol{x}$ then has elements*

$$\begin{aligned} Q_{1,1} &= \frac{1-\rho^2}{1-\rho^{2(t_2-t_1)}} \\ Q_{m,m} &= \frac{1-\rho^2}{1-\rho^{2(t_m-t_{m-1})}} \\ Q_{i,i} &= \frac{(1-\rho^2)\left(1-\rho^{2(t_{i+1}-t_{i-1})}\right)}{\left(1-\rho^{2(t_i-t_{i-1})}\right)\left(1-\rho^{2(t_{i+1}-t_i)}\right)}, \quad 1 < i < m, \\ Q_{i+1,i} &= Q_{i,i+1} = -\frac{(1-\rho^2)\,\rho^{t_{i+1}-t_i}}{1-\rho^{2(t_{i+1}-t_i)}}, \quad 1 \leq i < m, \\ Q_{i+k,i} &= Q_{i,i+k} = 0, \quad k = 2, 3, \dots, m-i, \quad 1 \leq i < m-1. \end{aligned}$$

$\boldsymbol{Q}$ is thus a tridiagonal matrix.

Proof. The fact that $\boldsymbol{Q}$ is a tridiagonal matrix follows from the interpretation of the off-diagonal elements of $\boldsymbol{Q}$ as the negated and scaled conditional correlations of $\boldsymbol{x}$ (Theorem 2.2 in Rue and Held, 2005):

$$\mathrm{Corr}(X_i, X_j|\boldsymbol{x}_{-ij}) = -\frac{Q_{ij}}{\sqrt{Q_{ii}Q_{jj}}}, \quad \text{where} \tag{5}$$

$$Q_{ii} = \mathrm{Prec}(X_i|\boldsymbol{x}_{-i}) \tag{6}$$

and the analogously defined Q_{jj} are the conditional precisions (variance reciprocals). Here, $\boldsymbol{x}_{-ij}$ denotes all elements of $\boldsymbol{x}$ except the ith and jth.

Now take $t_i, t_j, t_\ell, t_k \in \{t_1, t_2, \ldots, t_m\}$ with $1 \leq i < \ell \leq k < j \leq m$ and assume that t_ℓ is the nearest observed time after t_i, and likewise that t_k is the nearest observed time before t_j. Note then that Equation (1) allows the representation

$$X_{t_i} = \rho^{-(t_\ell - t_i)} \left(X_{t_\ell} - \sum_{b=0}^{t_\ell - t_i - 1} \rho^b \epsilon_{t_\ell - b} \right) \tag{7}$$

$$X_{t_j} = \rho^{t_j - t_k} X_{t_k} + \sum_{b=0}^{t_j - t_k - 1} \rho^b \epsilon_{t_j - b} \tag{8}$$

The quantities X_{t_ℓ} and X_{t_k} are assumed known in the conditional correlation $\text{Corr}(X_{t_i}, X_{t_j} | \boldsymbol{x}_{-ij})$, and no same error term appears in both sums above. Thus, the correlation equals zero, which implies that $Q_{ij} = 0$ when we have at least one observation between t_i and t_j. $\boldsymbol{Q}$ is then (at most) tridiagonal.

With this knowledge, one way to determine the explicit form of the elements of $\boldsymbol{Q}$ is to solve the system of equations $\boldsymbol{\Sigma Q} = \boldsymbol{I}_m$, where $\boldsymbol{I}_m$ is the $m \times m$ identity matrix. For notational convenience, we define $\rho_{ji} = \rho^{j-i}$ with $j > i$. Because $\boldsymbol{Q}$ is symmetric ($Q_{k,k+1} = Q_{k+1,k}$) and tridiagonal, we only have $2m-1$ unknowns to solve for, and the system $\boldsymbol{\Sigma Q} = \boldsymbol{I}_m$ can be reduced to the set of equations

$$\begin{cases} Q_{11} + \rho_{21} Q_{12} & = 1 - \rho^2 \\ \rho_{21} Q_{11} + Q_{12} & = 0 \end{cases}, \tag{9}$$

$$\begin{cases} \rho_{k,k-1} Q_{k-1,k} + Q_{kk} + \rho_{k+1,k} Q_{k,k+1} & = 1 - \rho^2 \\ \rho_{k+1,k-1} Q_{k-1,k} + \rho_{k+1,k} Q_{kk} + Q_{k,k+1} & = 0 \end{cases}, \quad 1 < k < m, \tag{10}$$

$$\Big\{ \rho_{m,m-1} Q_{m-1,m} + Q_{mm} \quad = 1 - \rho^2. \tag{11}$$

The system (9) may be solved for Q_{11} and Q_{12}, after which system (10) can be solved iteratively for $k = 2, 3, \ldots, m-1$ by inserting $Q_{k-1,k}$ found in the previous iteration. Lastly, Q_{mm} is easily solved for in Equation (11) when $Q_{m-1,m}$ has been found in the previous step.

If the assumption $\sigma = 1$ is relaxed, the non-zero elements of $\boldsymbol{Q}$ should be divided by σ^2 to yield the correct precision matrix. □

Since $\boldsymbol{Q}$ has only $2m-1$ non-zero elements, it can be constructed using only $\mathcal{O}(m)$ flops (floating point operations). Likewise, it needs only $\mathcal{O}(m)$ space for storage if stored in a sparse format.

Corollary 1.1. *Let $Y_{t_i} = X_{t_i} + \mu_{t_i}$, with X_{t_i} an element of $\boldsymbol{x}$ as in Theorem 1 and the μ_{t_i}'s fixed. Also let $\boldsymbol{y} = (Y_{t_1}, \ldots, Y_{t_m})'$. Then by Theorem 2.3 of Rue and Held (2005), the following conditional expected values hold for the elements of $\boldsymbol{y}$:*

$$\mathbb{E}[Y_{t_1}|\boldsymbol{y}_{-1}] = \mu_{t_1} + \rho^{t_2-t_1}(y_{t_2} - \mu_{t_2}) \tag{12}$$

$$\begin{aligned}\mathbb{E}[Y_{t_i}|\boldsymbol{y}_{-i}] = \mu_{t_i} &+ \rho^{t_i-t_{i-1}}\frac{1-\rho^{2(t_{i+1}-t_i)}}{1-\rho^{2(t_{i+1}-t_{i-1})}}(y_{t_{i-1}} - \mu_{t_{i-1}}) \\ &+ \rho^{t_{i+1}-t_i}\frac{1-\rho^{2(t_i-t_{i-1})}}{1-\rho^{2(t_{i+1}-t_{i-1})}}(y_{t_{i+1}} - \mu_{t_{i+1}})\end{aligned} \tag{13}$$

$$\mathbb{E}[Y_{t_m}|\boldsymbol{y}_{-m}] = \mu_{t_m} + \rho^{t_m-t_{m-1}}(y_{t_{m-1}} - \mu_{t_{m-1}}) \tag{14}$$

Together with the expression for the conditional precision given in Equation (6), Corollary 1.1 specifies the full conditional (normal) distributions of the irregularly sampled AR(1) process.

3 Implications

Typical density evaluation and random number generation of multivariate normal variables involves a Cholesky or eigendecomposition of the covariance matrix (Ripley, 1987; Venables and Ripley, 2002). For an $m \times m$ matrix, the computational cost (in terms of the number of flops) associated with either method is $\mathcal{O}(m^3)$ (Trefethen and Bau, 1997). For large m, this cost becomes prohibitive, and even with m smaller such decompositions can become the bottleneck when they need to be performed repeatedly. For example, a Markov Chain Monte Carlo algorithm trying to infer the value (distribution) of ρ may need to evaluate the density of $\boldsymbol{x}$ thousands of times, if not more. However, if the sparse structure of $\boldsymbol{Q}$ can be used, this cost can be drastically reduced.

3.1 Cholesky factorization

Indeed, the sparsity of $\boldsymbol{Q}$ carries over to its Cholesky decomposition. By Theorem 2.9 in Rue and Held (2005), the (lower) Cholesky decomposition $\boldsymbol{L}$ of $\boldsymbol{Q}$, i.e. $\boldsymbol{Q} = \boldsymbol{L}\boldsymbol{L}^T$, will have a lower bandwidth of 1—that is, only the main diagonal and (first) subdiagonal will have non-zero elements. This decomposition is computable in linear time using Algorithm 2.9 in Rue and Held (2005), described next.

Let v be a vector of length m and let $v_{i:j}$ be elements i to j of this vector. Denote by $Q_{i:k,j}$ elements i to k in column j of $\boldsymbol{Q}$, and let the same notation

be applicable to $\boldsymbol{L}$ and its elements $L_{i,j}$, which are initialized to zero. The matrix $\boldsymbol{L}$ can then be computed using the following algorithm:

Algorithm 1 Band-Cholesky factorization of $\boldsymbol{Q}$ (bandwidth 1)

for $j = 1$ to m **do**
  $\lambda \leftarrow \min\{j+1, m\}$
  $v_{j:\lambda} \leftarrow Q_{j:\lambda,j}$
  if $j > 1$ **then**
    $v_j \leftarrow v_j - L_{j,j-1}^2$
  $L_{j:\lambda,j} \leftarrow v_{j:\lambda}/\sqrt{v_j}$
Return $\boldsymbol{L}$

This algorithm is seen to involve only $\mathcal{O}(m)$ flops. Additionally, if $\boldsymbol{L}$ is stored in a sparse format, the storage is of size $\mathcal{O}(m)$ as well.

3.2 Unconditional simulation

If we wish to sample $\boldsymbol{x} \sim \text{MVN}\left(\boldsymbol{\mu}, \boldsymbol{Q}^{-1}\right)$ as in Theorem 1, but now with a mean vector $\boldsymbol{\mu}$, we can use the following algorithm (Algorithm 2.4 in Rue and Held, 2005):

Algorithm 2 Sampling $\boldsymbol{x} \sim \text{Normal}(\boldsymbol{\mu}, \boldsymbol{Q}^{-1})$

1: Compute $\boldsymbol{L}$ using Algorithm 1.
2: Sample m standard normal variables and store them in a vector $\boldsymbol{z}$.
3: Solve $\boldsymbol{L}^T\boldsymbol{v} = \boldsymbol{z}$ using sparse back substitution (see Algorithm 3 below).
4: Compute $\boldsymbol{x} = \boldsymbol{\mu} + \boldsymbol{v}$.
5: **Return** $\boldsymbol{x}$

This algorithm is seen to be of order $\mathcal{O}(m)$ in computational complexity. The sparse back substitution in step 3 of Algorithm 2 computes the elements of $\boldsymbol{v}$ as follows:

Algorithm 3 Solving $\boldsymbol{L}^T\boldsymbol{v} = \boldsymbol{z}$ when $\boldsymbol{L}$ has bandwidth 1

$v_m = z_m/L_{m,m}$
for $i = m-1$ to 1 **do**
  $v_i = (z_i - L_{i+1,i}v_{i+1})/L_{i,i}$
Return $\boldsymbol{v}$

Only $3m - 2$ flops are used to produce the solution $\boldsymbol{v}$.

3.3 Conditional simulation

Assume now that $\boldsymbol{t}_{\mathrm{o}} = \{t_1, \ldots, t_m\}$ are the time points at which observations $\boldsymbol{x}_{\mathrm{o}}$ are available, and let $\boldsymbol{t}_{\mathrm{p}} = \{s_1, \ldots, s_k\}$ be another set of time points disjoint from $\boldsymbol{t}_{\mathrm{o}}$. Suppose that we wish to simulate values $\boldsymbol{x}_{\mathrm{p}}$ from the distribution of the process at times $\boldsymbol{t}_{\mathrm{p}}$, conditional on $\boldsymbol{x}_{\mathrm{o}}$. From standard facts about the multivariate normal distribution (see e.g. Rue and Held, 2005), we know that

$$\begin{aligned}
\boldsymbol{x}_{\mathrm{p}} | \boldsymbol{x}_{\mathrm{o}} &\sim \text{Normal}\left(\boldsymbol{\mu}_{\mathrm{p|o}}, \boldsymbol{\Sigma}_{\mathrm{p|o}}\right), \quad \text{where} \\
\boldsymbol{\mu}_{\mathrm{p|o}} &= \boldsymbol{\mu}_{\mathrm{p}} + \boldsymbol{\Sigma}_{\mathrm{po}} \boldsymbol{\Sigma}_{\mathrm{oo}}^{-1} \left(\boldsymbol{x}_{\mathrm{o}} - \boldsymbol{\mu}_{\mathrm{o}}\right), \\
\boldsymbol{\Sigma}_{\mathrm{p|o}} &= \boldsymbol{\Sigma}_{\mathrm{pp}} - \boldsymbol{\Sigma}_{\mathrm{po}} \boldsymbol{\Sigma}_{\mathrm{oo}}^{-1} \boldsymbol{\Sigma}_{\mathrm{op}},
\end{aligned}$$

and

$$\boldsymbol{\mu}_{\mathrm{a}} = \begin{bmatrix} \boldsymbol{x}_{\mathrm{p}}^* \\ \boldsymbol{x}_{\mathrm{o}}^* \end{bmatrix} \quad \text{and} \quad \boldsymbol{\Sigma}_{\mathrm{a}} = \begin{bmatrix} \boldsymbol{\Sigma}_{\mathrm{pp}} & \boldsymbol{\Sigma}_{\mathrm{po}} \\ \boldsymbol{\Sigma}_{\mathrm{op}} & \boldsymbol{\Sigma}_{\mathrm{oo}} \end{bmatrix}$$

are the mean vector and covariance matrix of $\boldsymbol{x}_{\mathrm{a}} = (\boldsymbol{x}_{\mathrm{p}}^T, \boldsymbol{x}_{\mathrm{o}}^T)^T$. In general, even if the matrix $\boldsymbol{\Sigma}_{\mathrm{oo}}^{-1}$ is sparse, the computation of $\boldsymbol{\Sigma}_{\mathrm{p|o}}$ will be demanding for $k >> m$, and further so if a Cholesky or eigendecomposition of the result is to be computed as well.

A more efficient way of sampling from the distribution of $\boldsymbol{x}_{\mathrm{p}} | \boldsymbol{x}_{\mathrm{o}}$, especially when $k >> m$, was described by Hoffman and Ribak (1991). To use this method, we need to be able to sample (unconditionally) from the joint distribution of $\boldsymbol{x}_{\mathrm{a}}$. One way of doing so is to just iteratively simulate from the definition of the process (Equation 1) using a starting value drawn from the stationary distribution (and with mean terms added back to the right hand side), and then pick out the values for the times $\boldsymbol{t}_{\mathrm{p}} \cap \boldsymbol{t}_{\mathrm{o}}$. Another is to first sort $\boldsymbol{s}_k \cap \boldsymbol{m}_m$, order and combine the mean vectors accordingly, create the corresponding precision matrix $\boldsymbol{Q}_{\mathrm{a}} = \boldsymbol{\Sigma}_{\mathrm{a}}^{-1}$, draw samples using Algorithm 2, and re-order the samples according to $\boldsymbol{t}_{\mathrm{p}}$ and $\boldsymbol{t}_{\mathrm{o}}$.

Let $\boldsymbol{Q}_{\mathrm{o}}$ be the precision matrix of $\boldsymbol{x}_{\mathrm{o}}$, i.e. $\boldsymbol{Q}_{\mathrm{o}} = \boldsymbol{\Sigma}_{\mathrm{oo}}^{-1}$. Then we can sample from the distribution of $\boldsymbol{x}_{\mathrm{p}} | \boldsymbol{x}_{\mathrm{o}}$ as follows (Hoffman and Ribak, 1991):

Algorithm 4 Sampling $\boldsymbol{x}_{\mathrm{p}} | \boldsymbol{x}_{\mathrm{o}} \sim \text{Normal}\left(\boldsymbol{\mu}_{\mathrm{p|o}}, \boldsymbol{\Sigma}_{\mathrm{p|o}}\right)$

1: Sample $\boldsymbol{x}_{\mathrm{a}}^* = \begin{bmatrix} \boldsymbol{x}_{\mathrm{p}}^* \\ \boldsymbol{x}_{\mathrm{o}}^* \end{bmatrix} \sim \text{Normal}\left(\boldsymbol{\mu}_{\mathrm{a}}, \boldsymbol{Q}_{\mathrm{a}}\right)$.
2: **Return** $\boldsymbol{x}_{\mathrm{p}} = \boldsymbol{x}_{\mathrm{p}}^* + \boldsymbol{\Sigma}_{\mathrm{po}} \boldsymbol{Q}_{\mathrm{o}} (\boldsymbol{x}_{\mathrm{o}} - \boldsymbol{x}_{\mathrm{o}}^*)$.

Because $\boldsymbol{Q}_{\mathrm{o}}$ is tridiagonal, the matrix product $\boldsymbol{\Sigma}_{\mathrm{po}} \boldsymbol{Q}_{\mathrm{o}}$ involves only $\mathcal{O}(km)$ flops, rather than the $\mathcal{O}(km^2)$ flops that are needed if $\boldsymbol{Q}$ is dense.

Thus the complexity of Algorithm 4 is $\mathcal{O}(km + (k+m)\log(k+m))$, where the latter term is due to the sorting required for Step 1.

3.4 Density evaluation

Many applications require the evaluation of probability density functions. For example, typical MCMC algorithms calculate a quotient (or log difference) of densities repeatedly in the evaluation of acceptance ratios. It is therefore of interest to make this computation as efficient as possible (or nearly so) for the type of irregularly sampled AR(1) process considered in this paper. Typically, the evaluation of a m-dimensional multivariate normal density involves the Cholesky decomposition of the covariance matrix, which as discussed previously has a computational complexity of $\mathcal{O}(m^3)$. With a sparse precision matrix $\boldsymbol{Q}$ however, this cost can be drastically reduced.

Let $\boldsymbol{x} \sim \text{Normal}\left(\boldsymbol{\mu}, \boldsymbol{Q}^{-1}\right)$ be a vector of values from an irregularly sampled AR(1) process as in Theorem 1, and let $p(\boldsymbol{x})$ be probability density function evaluated at $\boldsymbol{x}$. To calculate $\log p(\boldsymbol{x})$ of a sample $\boldsymbol{x}$ from this distribution, first calculate the Cholesky decomposition $\boldsymbol{L}$ of $\boldsymbol{Q}$ using Algorithm 1. Then, by Rue and Held (2005, p. 35), the log-density can be computed as follows:

$$\log p(\boldsymbol{x}) = -\frac{m}{2}\log(2\pi) + \sum_{i=1}^{m} \log L_{i,i} - \frac{1}{2}q, \quad \text{where} \tag{15}$$

$$q = (\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{Q} (\boldsymbol{x} - \boldsymbol{\mu}). \tag{16}$$

If $\boldsymbol{x}$ was generated using Algorithm 2, q simplifies to $q = \boldsymbol{z}^T\boldsymbol{z}$. By utilizing the sparsity of $\boldsymbol{Q}$, the evaluation of the log-density has a computational complexity of $\mathcal{O}(m)$.

4 Conclusions

This paper provides analytical expressions for the elements of the precision matrix $\boldsymbol{Q}$ of a stationary Gaussian AR(1) process sampled with irregular spacing. The sparsity of this matrix was shown in Section 3 to yield efficient algorithms for density evaluation and simulation of such a process. Applications of AR(1) processes are abound in finance and other fields, and the results of this paper should prove relevant for those in need of computational efficiency. More generally, the results are valuable from a missing data perspective. A possible extension of this paper is to relate the results to the distribution of a sample from a continuous-time Ornstein-Uhlenbeck process, also sampled at irregular intervals.

Acknowledgements

I acknowledge support from the Swedish Research Council, grant 2013:05204.

References

Hoffman, Y. and Ribak, E. (1991). Constrained realizations of Gaussian fields - a simple algorithm. *The Astrophysical Journal*, 380:L5–L8.

Lindsey, J. K. (2004). *Statistical Analysis of Stochastic Processes in Time*. Cambridge University Press.

Ripley, B. D. (1987). *Stochastic Simulation*. Wiley.

Rue, H. and Held, L. (2005). *Gaussian Markov Random Fields: Theory and Applications*. Chapman and Hall/CRC.

Trefethen, L. N. and Bau, D. I. (1997). *Numerical Linear Algebra*. SIAM.

Venables, W. N. and Ripley, B. D. (2002). *Modern Applied Statistics with S*. Springer, New York, fourth edition.