Modern spatial statistics

Geir-Arne Fuglstad

March 2, 2022

Outline

My plan

Focus will be on efficient methods for inference and I will

- give an overview of the main categories of approaches
- describe main ideas in each category
- delve into more details for a couple of methods

A good starting point for reading

The Handbook of Spatial Statistics[1] is a good and comprehensive resource for finding condensed information or references about different topics in spatial statistics. It is from 2010 and is missing developments in 2010–2021, though.

[1] Gelfand, A. E., Diggle, P., Guttorp, P., & Fuentes, M. (Eds.). (2010). *Handbook of spatial statistics*. CRC press

Some core challenges

Focus will be on Gaussian random fields (GRFs).

Some core challenges

Focus will be on Gaussian random fields (GRFs).

- **Modelling complex phenomena**
  - Environmental data, extreme data, survival data, count data, (marked) point process data.
  - E.g., precipitation, flooding, pollution, disease cases, mortality, earthquakes.

Some core challenges

Focus will be on Gaussian random fields (GRFs).

- **Modelling complex phenomena**
  - Environmental data, extreme data, survival data, count data, (marked) point process data.
  - E.g., precipitation, flooding, pollution, disease cases, mortality, earthquakes.
- **Constructing *useful* flexible covariance models**
  - How many parameters and how to ensure validity?
  - Are the parameters interpretable?
  - Are the parameters consistently estimable?

Some core challenges

Focus will be on Gaussian random fields (GRFs).

- **Modelling complex phenomena**
  - Environmental data, extreme data, survival data, count data, (marked) point process data.
  - E.g., precipitation, flooding, pollution, disease cases, mortality, earthquakes.
- **Constructing *useful* flexible covariance models**
  - How many parameters and how to ensure validity?
  - Are the parameters interpretable?
  - Are the parameters consistently estimable?
- **Inference with large datasets**
  - Naive methods stop working around 10^3.
  - Satellites could lead to 10^5–10^6

The computational challenge

A GRF u observed at locations $\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_n \in \mathbb{R}^d$, where $d \geq 1$, gives rise to a multivariate Gaussian distribution

$$\boldsymbol{u} = (u(\boldsymbol{s}_1), \ldots, u(\boldsymbol{s}_n)) \sim \mathcal{N}_n(0, \Sigma),$$

where Σ is a positive definite $n \times n$ matrix.

The computational challenge

A GRF u observed at locations $\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_n \in \mathbb{R}^d$, where $d \geq 1$, gives rise to a multivariate Gaussian distribution

$$\boldsymbol{u} = (u(\boldsymbol{s}_1), \ldots, u(\boldsymbol{s}_n)) \sim \mathcal{N}_n(0, \Sigma),$$

where Σ is a positive definite $n \times n$ matrix.

Inference requires many computations of the log-likelihood

$$\log(\pi(\boldsymbol{u})) = -\frac{n}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma| - \frac{1}{2}\boldsymbol{u}^{\mathrm{T}}\Sigma^{-1}\boldsymbol{u}.$$

The computational challenge

A GRF u observed at locations $\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_n \in \mathbb{R}^d$, where $d \geq 1$, gives rise to a multivariate Gaussian distribution

$$\boldsymbol{u} = (u(\boldsymbol{s}_1), \ldots, u(\boldsymbol{s}_n)) \sim \mathcal{N}_n(0, \Sigma),$$

where Σ is a positive definite $n \times n$ matrix.

Inference requires many computations of the log-likelihood

$$\log(\pi(\boldsymbol{u})) = -\frac{n}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma| - \frac{1}{2}\boldsymbol{u}^{\mathrm{T}}\Sigma^{-1}\boldsymbol{u}.$$

The problem is that both $\log|\Sigma|$ and $\Sigma^{-1}\boldsymbol{u}$ have complexity $\mathcal{O}(n^3)$!

What can we do?

There are a few main strategies

What can we do?

There are a few main strategies

- Reduce the dimensionality of the problem.
  - E.g., use a basis decomposition of the GRF.

What can we do?

There are a few main strategies

- Reduce the dimensionality of the problem.
  - E.g., use a basis decomposition of the GRF.
- Reduce the complexity of the problem.
  - E.g., use algorithms for sparse matrices

What can we do?

There are a few main strategies

- Reduce the dimensionality of the problem.
  - E.g., use a basis decomposition of the GRF.
- Reduce the complexity of the problem.
  - E.g., use algorithms for sparse matrices
- Approximate the likelihood $\log(\pi(\boldsymbol{u}))$.
  - E.g., $\pi(u_i|u_{i-1},\ldots,u_1) \approx \pi(u_i|u_{i-1},\ldots,u_{i-k})$ for a small k and $i = 1,\ldots,n$.

Reduce dimensionality: basis decomposition

This usually fits under the name **Fixed Rank Kringing**[2].

[2] Cressie, N., & Johannesson, G. (2008). Fixed rank kriging for very large spatial data sets. JRSS: Series B, 70(1), 209–226.

Reduce dimensionality: basis decomposition

This usually fits under the name **Fixed Rank Kringing**[2].

Introduce a set of deterministic basis functions $\{f_1, f_2, \ldots, f_k\}$ and multivariate Gaussian weights $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_k) \sim \mathcal{N}_k(0, \mathrm{K})$.

[2] Cressie, N., & Johannesson, G. (2008). Fixed rank kriging for very large spatial data sets. JRSS: Series B, 70(1), 209–226.

Reduce dimensionality: basis decomposition

This usually fits under the name **Fixed Rank Kringing**[2].

Introduce a set of deterministic basis functions $\{f_1, f_2, \ldots, f_k\}$ and multivariate Gaussian weights $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_k) \sim \mathcal{N}_k(0, \mathrm{K})$.

Require that $u(\boldsymbol{s}) = \sum_{i=1}^{k} \alpha_i f_i(\boldsymbol{s}) = \boldsymbol{a}(\boldsymbol{s})^{\mathrm{T}} \boldsymbol{\alpha}$, where

$$\boldsymbol{a}(\boldsymbol{s})^{\mathrm{T}} = \begin{bmatrix} f_1(\boldsymbol{s}) & f_1(\boldsymbol{s}) & \cdots & f_k(\boldsymbol{s}) \end{bmatrix}.$$

[2] Cressie, N., & Johannesson, G. (2008). Fixed rank kriging for very large spatial data sets. JRSS: Series B, 70(1), 209–226.

Reduce dimensionality: computational complexity

Problem 1:
We observe $y_i = u(\boldsymbol{s}_i) + \epsilon_i$ for $i = 1, 2, \ldots, n$, where u is a GRF described by a basis of k elements, and $\epsilon_1, \ldots, \epsilon_n \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2_{\text{N}})$ are measurement errors.

Derive the computational complexity of evaluating the log-likelihood $\log(\pi(y_1, y_2, \ldots, y_n))$ as a function of n and k.

Hint:
$(\mathsf{B} + \mathsf{UCV})^{-1} = \mathsf{B}^{-1} - \mathsf{B}^{-1}\mathsf{U}(\mathsf{C}^{-1} + \mathsf{VB}^{-1}\mathsf{U})^{-1}\mathsf{VB}^{-1}$
$\det(\mathsf{B} + \mathsf{UCV}) = \det(\mathsf{C}^{-1} + \mathsf{VB}^{-1}\mathsf{U})\det(\mathsf{C})\det(\mathsf{B})$

Reduce dimensionality: challenges

What are the deficiencies in this approach?

Reduce dimensionality: challenges

What are the deficiencies in this approach?

- Need to select a good set of basis functions

Reduce dimensionality: challenges

What are the deficiencies in this approach?

- Need to select a good set of basis functions
- Need to estimate $k(k+1)/2$ parameters for a $k \times k$ covariance matrix

Reduce dimensionality: challenges

What are the deficiencies in this approach?

- Need to select a good set of basis functions
- Need to estimate $k(k+1)/2$ parameters for a $k \times k$ covariance matrix
- Not good for capturing small scale spatial variation
- Tricky to reconstruct a specific covariance function such as the Matérn

There are two factors contributing to the computational complexity

The computation complexity is $\mathcal{O}(n^3)$.

The basis decomposition can reduce n to $k \ll n$.

If k needs to be larger, can we reduce the exponent 3?

Reducing the exponent: sparse matrices

If $\boldsymbol{u} = (u(\boldsymbol{s}_1), u(\boldsymbol{s}_2), \ldots, u(\boldsymbol{s}_n)) \sim \mathcal{N}_n(0, \Sigma)$, there are, in principle, two approaches:

1. Make the **covariance matrix** Σ sparse
   - E.g., covariance tapering[3]

[3] Furrer, R., Genton, M. G., & Nychka, D. (2006). Covariance tapering for interpolation of large spatial datasets. *Journal of Computational and Graphical Statistics*, 15(3), 502-523.

Reducing the exponent: sparse matrices

If $\boldsymbol{u} = (u(\boldsymbol{s}_1), u(\boldsymbol{s}_2), \ldots, u(\boldsymbol{s}_n)) \sim \mathcal{N}_n(0, \Sigma)$, there are, in principle, two approaches:

1. Make the **covariance matrix** Σ sparse
   - E.g., covariance tapering[3]
2. Make the **precision matrix** $\mathrm{Q} = \Sigma^{-1}$ sparse
   - E.g., enforce Markov properties through Gaussian Markov random fields (GMRFs)[4]

[3] Furrer, R., Genton, M. G., & Nychka, D. (2006). Covariance tapering for interpolation of large spatial datasets. *Journal of Computational and Graphical Statistics*, 15(3), 502-523.

[4] Rue, H., & Held, L. (2005). *Gaussian Markov random fields: theory and applications*. CRC press.

Reducing the exponent: Sparse matrix algorithms

Covariance matrix:

$$\log(\pi(\boldsymbol{u})) = -\frac{n}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma| - \frac{1}{2}\boldsymbol{u}^{\mathrm{T}}\Sigma^{-1}\boldsymbol{u}$$

Precision matrix:

$$\log(\pi(\boldsymbol{u})) = -\frac{n}{2}\log(2\pi) + \frac{1}{2}\log|\mathrm{Q}| - \frac{1}{2}\boldsymbol{u}^{\mathrm{T}}\mathrm{Q}\boldsymbol{u}$$

If Q (or Σ) is sparse, the sparse Cholesky factorization $\mathrm{Q} = \mathrm{L}\mathrm{L}^{\mathrm{T}}$ (or $\Sigma = \mathrm{L}\mathrm{L}^{\mathrm{T}}$) can be computed fast. We will discuss how fast later!

Problem 2: What is the computational complexity of the log-likelihood if the sparse Cholesky factor L is available?

Reducing the exponent: how to achieve sparsity?

I will only talk about making the precision matrix sparse and will give one simple example and then describe a complex approach termed "the SPDE approach".

Simple example: AR(1) process

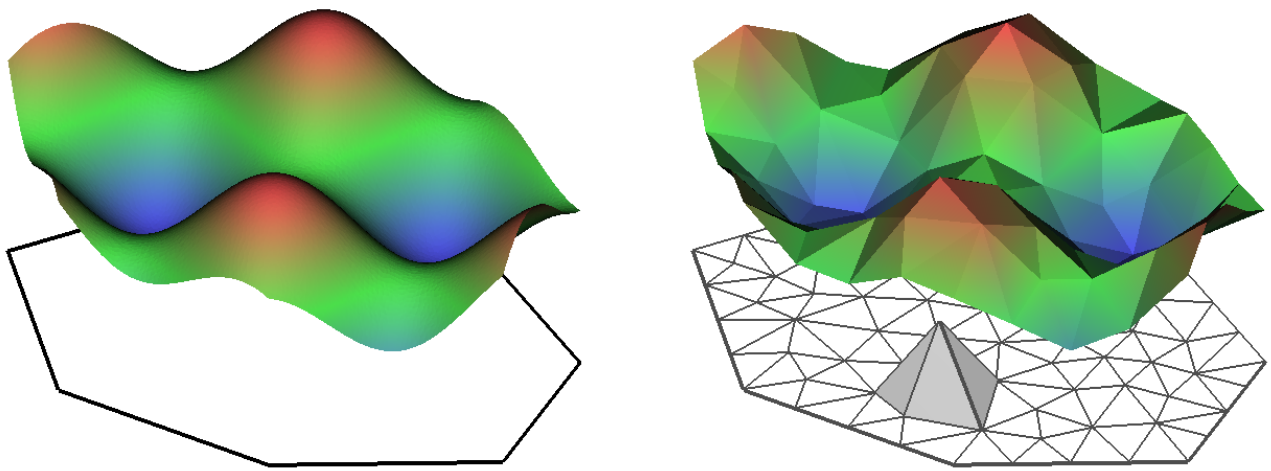
The basic idea of the "SPDE approach"

- Forget about covariance matrices
- A GRF changes gradually and little information is lost by using a piece-wise linear approximation of "good enough" resolution.
  NB: need a lot more functions than Fixed Rank Kriging.
- Use *numerics* + *stochastic partial differential equations (SPDEs)* to automatically create sparse precision matrices for the multivariate Gaussian distribution of the coefficients

The basic idea of the "SPDE approach"

Replace the random field u with a piece-wise linear random field $\tilde{u}$ described by $\boldsymbol{w} = (w_1, w_2, \ldots, w_k) \sim \mathcal{N}(0, \mathrm{Q}^{-1})$ and pyramidal basis functions $\{f_i\}_{i=1}^k$,

$$\tilde{u}(\boldsymbol{s}) = \sum_{i=1}^{k} w_i f_i(\boldsymbol{s})$$



Why are we going to talk about SPDEs?

- If $\boldsymbol{w}$ requires a dense precision matrix, we did not improve on fixed rank kriging
- We need to ensure both that the precision matrix is sparse *and* that we get a useful GRFs described by meaningful parameters
- This is where the SPDEs enter!

The SPDE

It can be shown that specifying the Matérn covariance function

$$c(d) = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{8\nu}}{\rho} d \right) K_\nu \left(\frac{\sqrt{8\nu}}{\rho} d \right)$$

is equivalent to requiring that the GRF u satisfies

$$[\kappa^2 - \Delta]^{\alpha/2} (\tau u(\boldsymbol{s})) = \mathcal{W}(\boldsymbol{s}), \quad \boldsymbol{s} \in \mathbb{R}^2,$$

where there is a simple conversion between (σ^2, ρ, ν) and (κ, τ, α).

This is hard; see paper for details[5]!

[5] Lindgren, F., Rue, H., & Lindström, J. (2011). An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach. *JRSS: Series B*, 73(4), 423-498.

Connecting sparsity and the SPDE

The equation itself is important when implementing the approach, but not important when using the approach.

The key ideas are

- If we require $\tilde{u}$ to approximately satisfy the SPDE, the covariance structure of $\tilde{u}$ is approximately Matérn
- Using *finite element methods* or *finite volume methods* from numerics gives a way to map parameters (κ, τ, α) to a sparse precision matrix, $\mathsf{Q}(\kappa, \tau, \alpha)$, for $\boldsymbol{w}$
- This gives an **automatic** way to calculate sparse precision matrices that are **consistent** for different resolutions.

Note: There are boundary effects and discretization effects.

Some key points

- Not dimension reduction. Usually, $k \geq n$.
- $\tilde{u}(\boldsymbol{s}) = \boldsymbol{a}(\boldsymbol{s})^{\mathrm{T}}\boldsymbol{w}$, where $\boldsymbol{a} = (f_1(\boldsymbol{s}), \ldots, f_m(\boldsymbol{s}))$.
- Posterior of parameters evaluated indirectly:

$$\pi(\boldsymbol{\theta}|\boldsymbol{y}) \propto \frac{\pi(\boldsymbol{\theta})\pi(\boldsymbol{w}|\boldsymbol{\theta})\pi(\boldsymbol{y}|\boldsymbol{w},\boldsymbol{\theta})}{\pi(\boldsymbol{w}|\boldsymbol{y},\boldsymbol{\theta})}.$$

Often we use $\boldsymbol{w} = \mathrm{E}[\boldsymbol{w}|\boldsymbol{y},\boldsymbol{\theta}]$ to evaluate right-hand-side.

What is the computational complexity

You will see the details about Gaussian Markov random fields (GMRFs) later, but we can exploit results for spatial GMRFs to see that

- for $\mathbb{R}^2$, we have $\mathcal{O}(k^{1.5} + n)$

What is the computational complexity

You will see the details about Gaussian Markov random fields (GMRFs) later, but we can exploit results for spatial GMRFs to see that

- for $\mathbb{R}^2$, we have $\mathcal{O}(k^{1.5} + n)$
- for $\mathbb{R}^1$, we would get $\mathcal{O}(k + n)$
- for $\mathbb{R}^3$, we would get $\mathcal{O}(k^2 + n)$

Are there any reasons to use SPDEs that are not computationally motivated?

- The entire field is explicitly modelled, predictions of the spatial effect is automatically available everywhere
- We can easily make GRFs on spheres (or other manifolds)
- It is easy to observe linear transformations of the GRF, such as integrals over sub-regions
- The SPDE formulation allows for interesting ways to create non-stationarity, multivariate GRFs, oscillating GRFs and so on

Why is this not enough?

The number of basis functions needs to increase with the dimension d of the base space $\mathbb{R}^d$.

If you need $k = m$ basis functions in 2D, you probably need $k = (\sqrt{m})^3 = m^{1.5}$ basis functions in 3D.

So the complexity for 3D, $\mathcal{O}(k^2 + n) = \mathcal{O}(m^3 + n)$, is actually much worse than it seems.

Approximating the likelihood

If you want something really quick, you may have to give up exact calculations of the likelihood.

E.g., composite likelihoods such as pairwise likelihood $\pi_{\mathrm{C}}(\boldsymbol{u}) = \prod_{i<j} \pi(u_i, u_j)$. Speedup is achieved if only nearby pairs are considered. However, this does not correspond to a well-defined GRF.

Vecchia approximations

We can also exploit the screening effect, choose a small $k \geq 1$, and employ Vecchia approximations[6],

$$\pi(\boldsymbol{u}) = \pi(u_1) \prod_{i=2}^{n} \pi(u_i | u_{i-1}, \ldots, u_1) \approx \pi(u_1) \prod_{i=2}^{n} \pi(u_i | u_{i-1}, \ldots, u_{i-k})$$

This corresponds to approximating the full covariance structure with a sparse precision matrix.

This is has a linear complexity in n, but is sensitive to ordering and it is hard to assess how good the approximation will be.

[6] Katzfuss, M., & Guinness, J. (2021). A general framework for Vecchia approximations of Gaussian processes. *Statistical Science*, 36(1), 124-141.

Some final points

- Parallelization is becoming important
- Spatio-temporal models are still a huge computational challenge
- Even when the ideas look simple, you will find that implementation can be tricky
- For GRFs, everything comes down to fast linear algebra in the end