# Computational Methods for Large Spatial Data

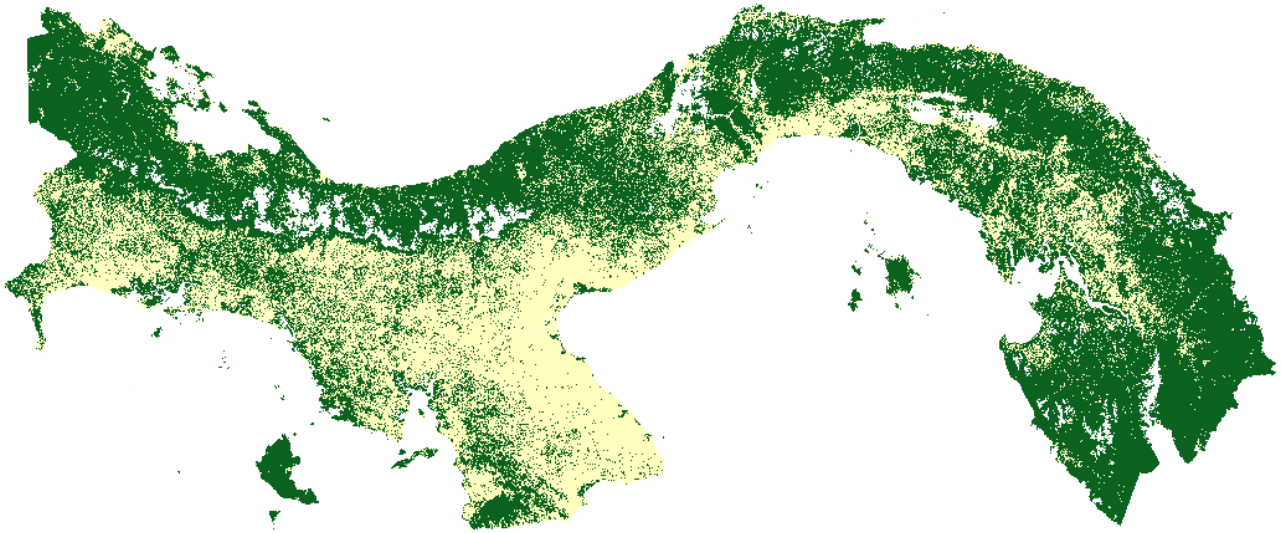
Reetam Majumder

Applied Spatial Statistics, Mizzou
April 15, 2025

# Overview of Kriging

# Motivating example

- LANDSAT and MODIS are satellites which provide optical information of the planet
- What they actually 'measure' is spectral and thermal data - affected by cloud cover etc



- A common statistical problem is to make predictions at unobserved locations

# The spatial model

- Let $Y_i$ be a measure of NDVI, a greenness metric used to monitor changes in land use (e.g., urbanization, agriculture, fires)
- $Y_i$ is observed at locations $\mathbf{s}_i = (s_{i1}, s_{i2}), i = 1 : n$.
- $\mathbf{X}_i$ are $p$ covariates at location $i$ - e.g., elevation data.
- A standard spatial model representation is $Y_i = \mu_i + Z_i + \epsilon_i$
- $\mu_i = \mathbf{X}\beta$; similar to linear regression
- There are 2 error terms:
  - $\epsilon_i \overset{iid}{\sim} N(0, \tau^2)$; called the **nugget**
  - $Z_i$ is mean 0, spatially correlated
- $Z_i$ captures spatial correlation not explained by $\mathbf{X}$

# Spatial covariance

- $E(Y_i) = \mu_i$
- $Z_i$ is independent of $\epsilon_j$ for all $(i, j)$ pairs, and so:
  - $\Sigma_{ii}(\theta) := V(Y_i) = \sigma^2 + \tau^2$
  - $\Sigma_{ij}(\theta) = Cov(Y_i, Y_j) = \sigma^2 \rho(d_{ij}, \phi)$
- $d_{ij}$ is the distance between $\mathbf{s}_i$ and $\mathbf{s}_j$, $\phi$ is the spatial range
- Common forms for $\rho(\cdot)$ include exponential and squared exponential, and Matern.
- We'll denote the coviance matrix as $\Sigma(\theta)$; dimensions = $n \times n$
- Stationarity and isotropy are common assumptions - strong, but often necessary

# Kriging

- Given all this we want to predict $\hat{Y}_0$ at $\mathbf{s}_0$
- Ideally, some uncertainty quantification (standard deviation, prediction interval etc)
- Kriging just assumes a constant mean, and known covariance
- Gaussian data is not necessary, but it makes things easier
- The 'optimal' prediction is given by

$$\hat{Y}_0 = \mu_0(\hat{\beta}) + \Sigma_0(\hat{\theta})\Sigma(\hat{\theta})^{-1}\{\mathbf{Y} - \mu(\hat{\beta})\}$$

- Inverting $\Sigma(\hat{\theta})$ takes $\mathcal{O}(n^3)$ computational cost and $\mathcal{O}(n^2)$ storage
- Panama has $\sim 1.7 \times 10^7$ observed pixels

This is a major bottleneck.

# Dealing with Large Datasets

# Some common approaches

- Brian's class notes
- The Vecchia approximation[1] has taken off again in recent years with the proliferation of large datasets in environment, ecology, epidemiology etc.

[1] Wiki article

# Example: time series data

Say $X_1, \ldots, X_t, \ldots X_n$ follow a time series such that:

$$X_t = \Phi X_{t-1} + \epsilon_t,$$
$$\epsilon_t \overset{iid}{\sim} N(0, \sigma^2).$$

Note that for the joint likelihood,

$$f(x_1, \ldots, x_n) \neq \prod_{t=1}^{n} f(x_t)$$

However, the following relationship holds:

$$\begin{aligned} f(x_1, \ldots, x_n) &= f(x_1) \cdot f(x_2|x_1) \cdot f(x_3|x_2, x_1) \ldots f(x_n|x_{n-1}, \ldots, x_1) \\ &\approx \prod_{t=1}^{n} f(x_t|x_{t-1}) \end{aligned}$$

However, there is no natural ordering in spatial data.

# The Vecchia approximation

- Let $Y_1, \ldots, Y_n$ be an ordered set of random variables
- For any ordering, you can express their joint distribution as

$$f(y_1, \ldots, y_n; \theta) = f(y_1; \theta) \prod_{i=2}^{n} f(y_i | y_{i-1}, \ldots y_1; \theta)$$

- For every $y_i, i > 1$, consider the set $\mathcal{N}_i \subset \{1, \ldots, i-1\}$
- The Vecchia approximation is

$$f(y_1, \ldots, y_n; \theta) \approx f(y_1) \prod_{i=2}^{n} f(y_i | y_{(i)}; \theta),$$

  where $y_{(i)} = \{y_j; j \in \mathcal{N}_i\}$
- $\mathcal{N}_i$ is often called the Vecchia neighbor set; $|\mathcal{N}_i| \leq m$

How to order? How to choose m?

# Simplifying the precision matrix for spatial data

- $\Omega(\theta) = \Sigma(\theta)^{-1}$ is defined as the precision matrix. Sparsity of the precision matrix simplifies computations
- Consider the following Vecchia approximation

$$\begin{aligned} f(y_1, \ldots, y_5) &= f(y_1)f(y_2|y_1)\ldots f(y_4|y_3, y_2, y_1) \\ &\approx f(y_1)f(y_2|y_1)\ldots f(y_5|y_4) \end{aligned}$$

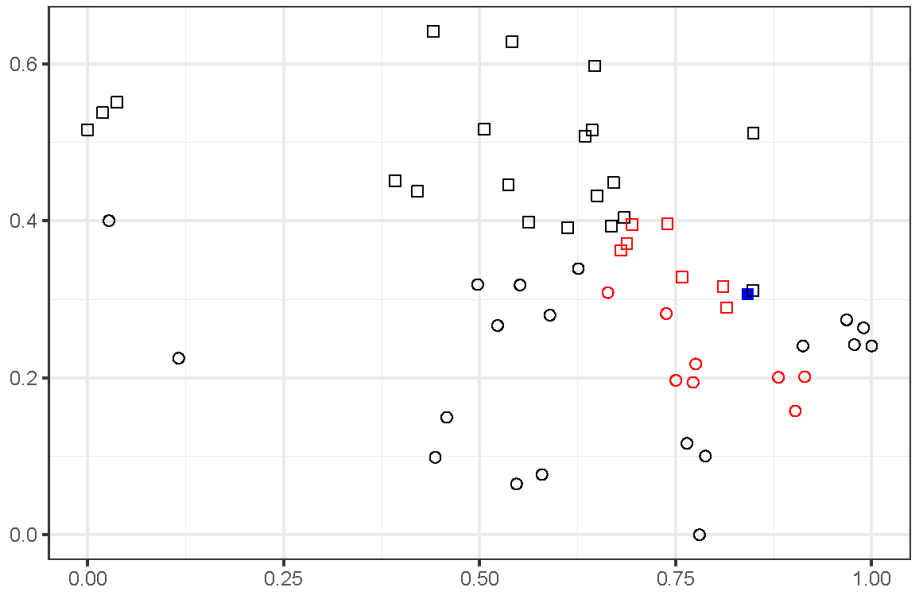
- This elicits a sparse precision matrix proportional to

$$\begin{bmatrix} 1 & k_{12} & 0 & 0 & 0 \\ & 1 & k_{23} & 0 & 0 \\ & & 1 & k_{34} & 0 \\ & & & 1 & k_{54} \\ & & & & 1 \end{bmatrix}$$

- The (structural) sparsity makes Cholesky decompositions easier
- Working with a Vecchia approximated process has $\mathcal{O}(nm^3)$ computational cost and needs $\mathcal{O}(nm^2)$ storage
- In practice, $m << n$

# Use in spatial modeling

- The approximation is usually applied to $\{Z_i\}$ and not $\{Y_i\}$
- In its simplest form, the ordering is done based on some coordinate system




- $m$ is often the set of nearest neighbors
- The general consensus is that for processes modeling mean behavior, the approximation is more sensitive to $|\mathcal{N}|$ than the ordering of locations

# Potential orderings

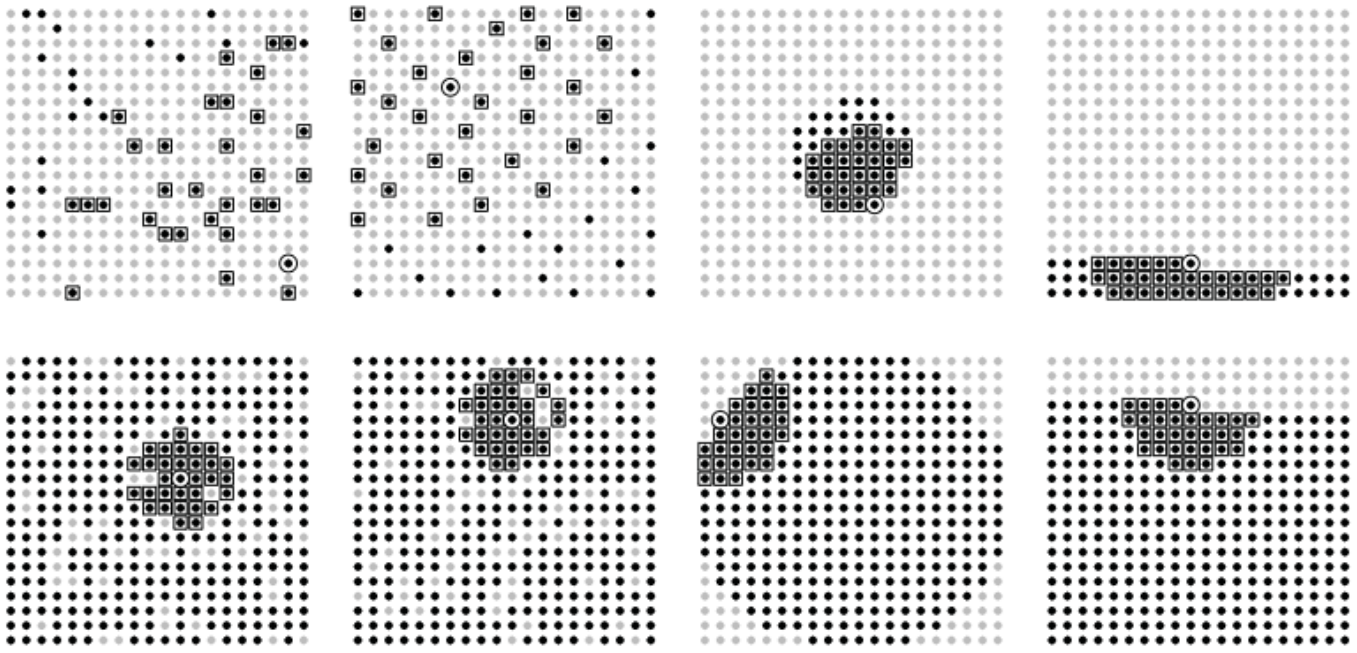
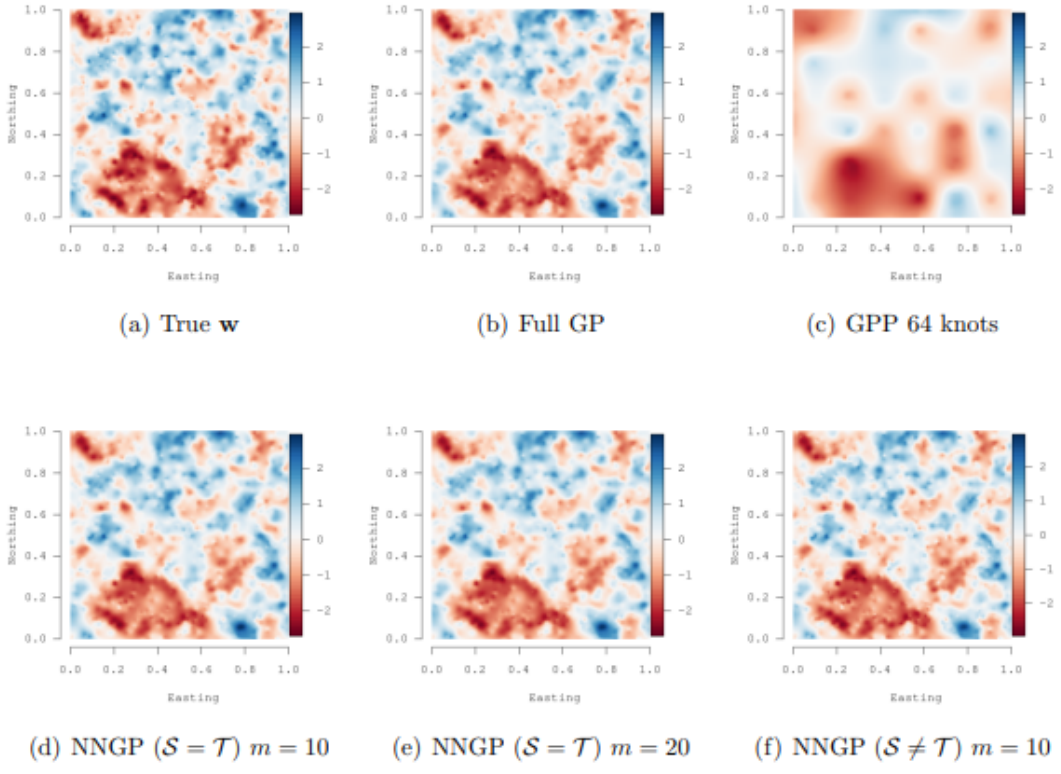


**Figure 1.** Examples of four orderings of 400 locations on a 20 × 20 grid. From left to right are completely random, maximum–minimum distance, middle out, and sorted on vertical coordinate. On the top row, the black circles are ordered 1–50, circled point is number 50, and squared points are 30 nearest neighbors to point 50 among previous points. Bottom row is the same for point ordered number 330.

Image from Guinness (2018)

# An example




(a) True $\mathbf{w}$ (b) Full GP (c) GPP 64 knots

(d) NNGP $(\mathcal{S} = \mathcal{T})$ $m = 10$ (e) NNGP $(\mathcal{S} = \mathcal{T})$ $m = 20$ (f) NNGP $(\mathcal{S} \neq \mathcal{T})$ $m = 10$

Image from Datta et al (2016)

# How do you use this?

- GPvecchia is another similar package, but which has a Bayesian implementation. [vignette link]
- NNGP (and the corresponding spNNGP package) is another common approach
- I've found GpGp to be more robust and faster
- Downside - not fully Bayesian
- Upside - supports a large number of covariance kernels
- More importantly, even if you don't use GpGp to fit/predict, you can always use its ordering function to create your own Vecchia approximation.