Spatial Inference with R-package BRISC
CUGOS, 2023 Spring Fling

Arkajyoti Saha, April 21, 2023
University of Washington
Department of Statistics
Outline

• What problem does BRISC solve?

• What can you do with BRISC?

• Applications of BRISC.
Outline

• What problem does BRISC solve?
• What can you do with BRISC?
• Applications of BRISC.
Geospatial/point-referenced data

Data: \{ (Y_i, X_i, s_i) : i = 1, ..., n \}
- \textbf{S} = (s_1, s_2, ..., s_n) : locations
- \textbf{Y} = \left( y(s_1), y(s_2), ..., y(s_n) \right) : observed response
- \textbf{X} = \left( x(s_1), x(s_2), ..., x(s_n) \right) : explanatory variables
Geospatial/point-referenced data

Data: \{ (Y_i, X_i, s_i) : i = 1, \ldots, n \}
- \textbf{S} = (s_1, s_2, \ldots, s_n) : locations
- \textbf{Y} = \left( y(s_1), y(s_2), \ldots, y(s_n) \right) : observed response
- \textbf{X} = \left( x(s_1), x(s_2), \ldots, x(s_n) \right) : explanatory variables

Objectives:
- Understand relationship between \textbf{X} and \textbf{Y}.
- Inference on spatial structure.
- Predict at a new location \( s_0 \).
How do we currently model this?
How do we currently model this?

Classical solution: **Ordinary Least Square Regression (OLS)**

\[ Y(s) = X(s) \beta + e(s) \]
How do we currently model this?

Classical solution: **Ordinary Least Square Regression (OLS)**

\[ Y(s) = X(s)\beta + \epsilon(s) \]
How do we currently model this?

Classical solution: **Ordinary Least Square Regression (OLS)**

\[ Y(s) = X(s) \beta + \epsilon(s) \]

- Linear Covariate effect
- White noise
How do we currently model this?

Classical solution: **Ordinary Least Square Regression (OLS)**

\[ Y(s) = X(s) \beta + e(s) \]

- **Linear Covariate effect**
- **White noise**
- Usually modeled with \( N(0, \tau^2) \)
How do we currently model this?

Classical solution: **Ordinary Least Square Regression (OLS)**

\[ Y(s) = X(s)\beta + \epsilon(s) \]

- Linear Covariate effect
- White noise

Usually modeled with \( N(0,\tau^2) \)

Doesn’t account for spatial effect.
How do we currently model this?

Account for spatial error: **Linear Mixed Model (LMM)**

$$Y(s) = X(s) \beta + e(s) + W(s)$$
How do we currently model this?

Account for spatial error: **Linear Mixed Model (LMM)**

$$Y(s) = X(s)\beta + \epsilon(s) + W(s)$$

- **Spatial random effect**
- Linear Covariate effect
- White noise
How do we currently model this?

Account for spatial error: **Linear Mixed Model (LMM) with GP**

\[ Y(s) = X(s)\beta + e(s) + W(s) \]

- **Spatial random effect**
- **Linear Covariate effect**
- **White noise**

Usually modeled with Gaussian Process (GP)
How do we estimate this?
How do we estimate this?

Maximum Likelihood Estimation

$$\text{Likelihood } (y) \propto |\Sigma|^{-\frac{1}{2}} \exp \left\{ (y - X\beta)^T \Sigma^{-1} (y - X\beta) \right\}$$
How do we estimate this?

Maximum Likelihood Estimation

\[
\text{Likelihood } (y) \propto \left| \Sigma \right|^{-\frac{1}{2}} \exp \left\{ (y - X\beta)^T \Sigma^{-1} (y - X\beta) \right\}
\]
How do we estimate this?

Maximum Likelihood Estimation

$$\text{Likelihood } (y) \propto |\Sigma|^{-\frac{1}{2}} \exp \left\{ (y - X\beta)^T \Sigma^{-1} (y - X\beta) \right\}$$

- Spatial error
- White noise
What can go wrong?

Maximum Likelihood Estimation

$$\text{Likelihood} (y) \propto |\Sigma|^{-\frac{1}{2}} \exp \left\{ (y - X\beta)^{\top} \Sigma^{-1} (y - X\beta) \right\}$$
Maximum Likelihood Estimation

\[
\text{Likelihood } (y) \propto \left| \Sigma \right|^{-\frac{1}{2}} \exp \left\{ (y - X\beta)^T \Sigma^{-1} (y - X\beta) \right\}
\]

What can go wrong?

Dense \( n \times n \)
What can go wrong?

Maximum Likelihood Estimation

\[
\text{Likelihood } (y) \propto |\Sigma|^{-\frac{1}{2}} \exp \left\{ (y - X\beta)^T \Sigma^{-1} (y - X\beta) \right\}
\]

\(n \times n\) Dense

\(O(n^3)\). Infeasible in large data!!!
How do we propose to solve this?

Maximum Likelihood Estimation

\[
\text{Likelihood } (y) \propto \lvert \Sigma \rvert^{-\frac{1}{2}} \exp \left\{ (y - X\beta)^T \Sigma^{-1} (y - X\beta) \right\}
\]
How do we propose to solve this?

Maximum Likelihood Estimation

\[
\text{Likelihood } (y) \propto \left| \Sigma \right|^{-\frac{1}{2}} \exp \left\{ (y - X\beta)^T \Sigma^{-1} (y - X\beta) \right\}
\]

\[
\Sigma^{-1} \text{ Cholesky } = \times
\]
How do we propose to solve this?

Maximum Likelihood Estimation

$$\text{Likelihood } (y) \propto \left| \Sigma \right|^{-\frac{1}{2}} \exp \left\{ (y - X\beta)^T \Sigma^{-1} (y - X\beta) \right\}$$

$$\Sigma^{-1} \text{ Cholesky } = \times$$

Dense
Maximum Likelihood Estimation

Likelihood \( (y) \propto |\Sigma|^{-\frac{1}{2}} \exp \left\{ (y - X\beta)^T \Sigma^{-1} (y - X\beta) \right\} \)

\[ \Sigma^{-1} \overset{\text{Cholesky}}{=} \]

\[ O\left( n^3 \right) \]

Dense

How do we propose to solve this?

\[ \times \]
Maximum Likelihood Estimation

How do we propose to solve this?

\[
\text{Likelihood } (y) \propto \left| \Sigma \right|^{-\frac{1}{2}} \exp \left\{ (y - X\beta)^T \Sigma^{-1} (y - X\beta) \right\}
\]

\(\Sigma^{-1} \text{ Cholesky} \equiv \times\)

"everything is related to everything else, but near things are more related than distant things."

For any location, only consider its correlation with its \(m\) nearest neighbors!!
How do we propose to solve this?

Maximum Likelihood Estimation

\[ \text{Likelihood} (y) \propto \left| \Sigma \right|^{-\frac{1}{2}} \exp \left\{ \left( y - X\beta \right)^{\top} \Sigma^{-1} \left( y - X\beta \right) \right\} \]

\[ \Sigma^{-1} \overset{\text{Cholesky}}{=} \begin{bmatrix} O(n^3) \\ \text{Dense} \end{bmatrix} \]

“everything is related to everything else, but near things are more related than distant things.”

For any location, only consider its correlation with its \( m \) nearest neighbors!!

How do we propose to solve this?

Maximum Likelihood Estimation

Likelihood \( (y) \propto \left| \Sigma \right|^{-\frac{1}{2}} \exp \left\{ (y - X\beta)^T \Sigma^{-1} (y - X\beta) \right\} \)

\( \Sigma^{-1} \text{ Cholesky} \equiv O(n^3) \approx \times \) Dense \times
How do we propose to solve this?

Maximum Likelihood Estimation

\[
\text{Likelihood } (y) \propto |\Sigma|^{-\frac{1}{2}} \exp \left\{ (y - X\beta)^T \Sigma^{-1} (y - X\beta) \right\}
\]

\[\Sigma^{-1} \text{ Cholesky} \approx \begin{pmatrix} O(n^3) \end{pmatrix} \times \begin{pmatrix} \text{Dense} \end{pmatrix} \]

\[\leq m \text{ non-zero per row} \]
How do we propose to solve this?

Maximum Likelihood Estimation

\[
\text{Likelihood } (y) \propto \left| \Sigma \right|^{-\frac{1}{2}} \exp \left\{ (y - X\beta)^T \Sigma^{-1} (y - X\beta) \right\}
\]

\[
\Sigma^{-1} \text{ Cholesky } \equiv O(n^3) \times \text{ Dense } \approx O(n) \times \leq m \text{ non-zero per row}
\]
- BRSIC implements this in R, a wrapper around C++ code.

- Embarrassingly parallel computation!

### Package ‘BRISC’

October 12, 2022

<table>
<thead>
<tr>
<th>Type</th>
<th>Package</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Fast Inference for Large Spatial Datasets using BRISC</td>
</tr>
<tr>
<td>Version</td>
<td>1.0.5</td>
</tr>
<tr>
<td>Maintainer</td>
<td>Arkajyoti Saha <a href="mailto:arkajyotisaha93@gmail.com">arkajyotisaha93@gmail.com</a></td>
</tr>
<tr>
<td>Author</td>
<td>Arkajyoti Saha [aut, cre], Abhirup Datta [aut], Jorge Nocedal [ctb], Naoaki Okazaki [ctb], Lukas M. Weber [ctb]</td>
</tr>
<tr>
<td>Depends</td>
<td>R (&gt;= 3.3.0), RANN, parallel, stats, rdist, matrixStats, pbapply, graphics</td>
</tr>
</tbody>
</table>
Outline

• What problem does BRISC solve?

• **What can you do with BRISC?**

• Applications of BRISC.
Inference

Significant improvement over state-of-the-art algorithms.

Training data ~ 105K

Test data ~ 45K
Inference

Significant improvement over state-of-the-art algorithms.

Training data ~ 105K

Test data ~ 45K

Classical methods do not work!!
Inference

Significant improvement over state-of-the-art algorithms.

Training data ~ 105K

Test data ~ 45K

Classical methods do not work!!

- NNGP with Bayesian (Datta et al.)
- NNGP with BRISC
Inference

Significant improvement over state-of-the-art algorithms.

Training data ~ 105K

Test data ~ 45K

Classical methods do not work!!

- NNGP with Bayesian (Datta et al.)
- NNGP with BRISC

Inference

- Estimation: `estimation_result <- BRISC_estimation(coords, y, x)`
Inference

• Estimation: \( \text{estimation\_result} \leftarrow \text{BRISC\_estimation}(\text{coords}, y, x) \)

• Uncertainty (via bootstrap): \( \text{BRISC\_bootstrap}(\text{estimation\_result}) \)
Inference

• Estimation: `estimation_result <- BRISC_estimation(coords, y, x)`

• Uncertainty (via bootstrap): `BRISC_bootstrap(estimation_result)`

• Prediction: `BRISC_prediction(estimation_result, coords_pred, x_pred)`
Inference

- Estimation: `estimation_result <- BRISC_estimation(coords, y, x)`
- Uncertainty (via bootstrap): `BRISC_bootstrap(estimation_result)`
- Prediction: `BRISC_prediction(estimation_result, coords_pred, x_pred)`
Simulating from Gaussian Process

Simulating LARGE data from Gaussian Process is computationally challenging.
Simulating from Gaussian Process

Simulating LARGE data from Gaussian Process is computationally challenging.

Simulate from NNGP with BRISC: \texttt{BRISC\_simulation(coords)}
Simulating from Gaussian Process

Simulating LARGE data from Gaussian Process is computationally challenging.

Simulate from NNGP with BRISC: BRISC_simulation(coords)
Simulating from Gaussian Process

Simulating LARGE data from Gaussian Process is computationally challenging.

Simulate from NNGP with BRISC: \texttt{BRISC\_simulation(coords)}

<table>
<thead>
<tr>
<th>Sample size</th>
<th>NNGP</th>
<th>full GP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>0.7 (0.04)</td>
<td>2.6 (0.08)</td>
</tr>
<tr>
<td>2500</td>
<td>1.6 (0.29)</td>
<td>31.8 (2.02)</td>
</tr>
<tr>
<td>5000</td>
<td>3.3 (0.25)</td>
<td>262.3 (9.33)</td>
</tr>
<tr>
<td>10000</td>
<td>8.3 (0.23)</td>
<td>NA</td>
</tr>
<tr>
<td>100000</td>
<td>121.5 (9.53)</td>
<td>NA</td>
</tr>
</tbody>
</table>
Probit Model

Estimates parameters by maximizing likelihood with a grid search.
### Probit Model

Estimates parameters by maximizing likelihood with a grid search.

<table>
<thead>
<tr>
<th>Methods</th>
<th>$n = 15^2$</th>
<th>$n = 25^2$</th>
<th>$n = 50^2$</th>
<th>$n = 100^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) Grid search for <strong>one</strong> parameter combination</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>probit-NNGP</td>
<td>0.065</td>
<td>0.5</td>
<td>9</td>
<td>166</td>
</tr>
<tr>
<td>TLR</td>
<td>0.57</td>
<td>2.9</td>
<td>28</td>
<td>187</td>
</tr>
<tr>
<td>(b) Prediction at <strong>one</strong> out-of-sample location following estimation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>probit-NNGP</td>
<td>&lt; 0.01</td>
<td>&lt; 0.01</td>
<td>&lt; 0.01</td>
<td>0.025</td>
</tr>
<tr>
<td>TLR</td>
<td>1.2</td>
<td>5.8</td>
<td>40</td>
<td>271</td>
</tr>
</tbody>
</table>

TLR = Low rank approximation of covariance matrix.
Probit Model

Estimates parameters by maximizing likelihood with a grid search.

<table>
<thead>
<tr>
<th>Methods</th>
<th>$n = 15^2$</th>
<th>$n = 25^2$</th>
<th>$n = 50^2$</th>
<th>$n = 100^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) Grid search for <strong>one</strong> parameter combination</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>probit-NNGP</td>
<td>0.065</td>
<td>0.5</td>
<td>9</td>
<td>166</td>
</tr>
<tr>
<td>TLR</td>
<td>0.57</td>
<td>2.9</td>
<td>28</td>
<td>187</td>
</tr>
<tr>
<td>(b) Prediction at <strong>one</strong> out-of-sample location following estimation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>probit-NNGP</td>
<td>&lt; 0.01</td>
<td>&lt; 0.01</td>
<td>&lt; 0.01</td>
<td>0.025</td>
</tr>
<tr>
<td>TLR</td>
<td>1.2</td>
<td>5.8</td>
<td>40</td>
<td>271</td>
</tr>
</tbody>
</table>

TLR = Low rank approximation of covariance matrix.

Probit Model

Estimates parameters by maximizing likelihood with a grid search.

Likelihood evaluation for estimation: `llk_binary <- Binary_estimation(coords, y)`
Probit Model

Estimates parameters by maximizing likelihood with a grid search.

Likelihood evaluation for estimation: \[ \text{llk_binary} \leftarrow \text{Binary_estimation}(\text{coords}, y) \]

Prediction: \[ \text{Binary_prediction}(\text{llk_binary}, \text{coords_pred}) \]
Probit Model

Estimates parameters by maximizing likelihood with a grid search.

Likelihood evaluation for estimation: `llk_binary <- Binary_estimation(coords, y)`

Prediction: `Binary_prediction(llk_binary, coords_pred)`

Tutorial: https://github.com/ArkajyotiSaha/probit-NNGP-code
Outline

• What problem does BRISC solve?

• What can you do with BRISC?

• **Applications of BRISC.**
Applications of BRISC

• nn-SVG: Identifies spatially variable genes transcriptomics data with BRISC.
Applications of BRISC

- nn-SVG: Identifies spatially variable genes transcriptomics data with BRISC.
- RF-GLS: \[ Y(s) = X(s) \beta + \epsilon(s) + W(s) \]
Applications of BRISC

- nn-SVG: Identifies of spatially variable genes transcriptomics data with BRISC.

- RF-GLS: \( Y(s) = X(s)\hat{\beta} + \epsilon(s) + W(s) \)
Applications of BRISC

- **nn-SVG**: Identifies of spatially variable genes transcriptomics data with BRISC.

- **RF-GLS**: $Y(s) = X(s)\beta + \epsilon(s) + W(s)$
  
  Fits Random Forest in spatially dependent data.
Applications of BRISC

- nn-SVG: Identifies of spatially variable genes transcriptomics data with BRISC.

- RF-GLS: \( Y(s) = X(s)\beta + \epsilon(s) + W(s) \)

  Fits Random Forest in spatially dependent data.

  Rewrite the split criteria as a Generalized Least Square (GLS) Loss.
Applications of BRISC

• **nn-SVG**: Identifies spatially variable genes transcriptomics data with BRISC.

• **RF-GLS**: $Y(s) = f(X) + \epsilon(s) + W(s)$

  Fits Random Forest in spatially dependent data.

  Rewrite the split criteria as a Generalized Least Square (GLS) Loss.
Applications of BRISC

• nn-SVG: Identifies of spatially variable genes transcriptomics data with BRISC.

• RF-GLS: $Y(s) = X(s)\beta + \epsilon(s) + W(s)$
  
  Fits Random Forest in spatially dependent data.

  Rewrite the split criteria as a Generalized Least Square (GLS) Loss.

Applications of BRISC

- **nn-SVG**: Identifies spatially variable genes transcriptomics data with BRISC.

- **RF-GLS**: $Y(s) = X(s)\beta + \epsilon(s) + W(s)$

  Fits Random Forest in spatially dependent data.

  Rewrite the split criteria as a Generalized Least Square (GLS) Loss.

  - **Continuous Outcome**: Saha A et al. Random forests for spatially dependent data. JASA. 2023.
Applications of BRISC

- **nn-SVG**: Identifies spatially variable genes transcriptomics data with BRISC.

- **RF-GLS**: \[ Y(s) = X(s)\beta + \epsilon(s) + W(s) \]
  Fits Random Forest in spatially dependent data.
  Rewrite the split criteria as a Generalized Least Square (GLS) Loss.

**Continuous Outcome**: Saha A et al. Random forests for spatially dependent data. JASA. 2023.


Acknowledgements

Abhirup Datta
Biostatistics, BSPH, JHU

Sumanta Basu
Statistics & Data Science, Cornell