Gaussian Process-based Spatially Varying Coefficient Models

Jakob A. Dambon[1, 2], Fabio Sigrist[2], Reinhard Furrer[1, 3]

[1]University of Zurich,
Department of Mathematics

[2]Lucerne University of Applied Sciences and Arts,
Institute of Financial Services Zug

[3]University of Zurich,
Department of Computational Science

Outline

Spatially Varying Coefficient Models

Gaussian Processes

MLE of GP-based SVC Models

Variable Selection for GP-based SVC Models

Spatially Varying Coefficient Models

What is a Spatially Varying Coefficient Model?

From a classical linear model

$$y_i = \beta_1 x_i^{(1)} + ... + \beta_p x_i^{(p)} + \varepsilon_i$$

to an SVC model

$$y_i = \beta_1(\boldsymbol{s}_i) x_i^{(1)} + ... + \beta_p(\boldsymbol{s}_i) x_i^{(p)} + \varepsilon_i.$$

for some spatial points data set.

The coefficients $\beta_j(\boldsymbol{s}_i)$ now depend on the location $\boldsymbol{s}_i$.

What is a Spatially Varying Coefficient Model?

From a classical linear model

$$y_i = \beta_1 x_i^{(1)} + ... + \beta_p x_i^{(p)} + \varepsilon_i$$

to an SVC model

$$y_i = \beta_1(\boldsymbol{s}_i) x_i^{(1)} + ... + \beta_p(\boldsymbol{s}_i) x_i^{(p)} + \varepsilon_i.$$

for some spatial points data set.

The coefficients $\beta_j(\boldsymbol{s}_i)$ now depend on the location $\boldsymbol{s}_i$.

What is a Spatially Varying Coefficient Model?

From a classical linear model

$$y_i = \beta_1 x_i^{(1)} + ... + \beta_p x_i^{(p)} + \varepsilon_i$$

to an SVC model

$$y_i = \beta_1(\boldsymbol{s}_i) x_i^{(1)} + ... + \beta_p(\boldsymbol{s}_i) x_i^{(p)} + \varepsilon_i.$$

for some spatial points data set.

The coefficients $\beta_j(\boldsymbol{s}_i)$ now depend on the location $\boldsymbol{s}_i$.

Simple Methodology

Geographically Weighted Regression (Brunsdon et al., 1998; Fotheringham et al., 2002):

- For each location $\boldsymbol{s}_i$, calculate distance d (e.g. Euclidean norm) to observation $\boldsymbol{s}_{i'}$.
- Use a *kernel function* $k(\cdot)$ on distances d to define a weighting matrix $\mathbf{W}_i$.
  - Kernel function: *exponential, Gaussian, etc.*
  - Depending on bandwidth b: either for **scaling** (d/b, for some $b > 0$) or **adaptive** with b nearest neighbors of $\boldsymbol{s}_i$ for some $b \in \mathbb{N}$.
- Then:

$$\hat{\beta}(\boldsymbol{s}_i) = (\mathbf{X}^\top \mathbf{W}_i \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{W}_i \boldsymbol{y}$$

Simple Methodology

Geographically Weighted Regression (Brunsdon et al., 1998; Fotheringham et al., 2002):

- For each location $\boldsymbol{s}_i$, calculate distance d (e.g. Euclidean norm) to observation $\boldsymbol{s}_{i'}$.
- Use a *kernel function* $k(\cdot)$ on distances d to define a weighting matrix $\mathbf{W}_i$.
  - Kernel function: *exponential, Gaussian, etc.*
  - Depending on bandwidth b: either for **scaling** (d/b, for some $b > 0$) or **adaptive** with b nearest neighbors of $\boldsymbol{s}_i$ for some $b \in \mathbb{N}$.
- Then:

$$\hat{\beta}(\boldsymbol{s}_i) = (\mathbf{X}^\top \mathbf{W}_i \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{W}_i \boldsymbol{y}$$

Simple Methodology

Geographically Weighted Regression (Brunsdon et al., 1998; Fotheringham et al., 2002):

- For each location $\boldsymbol{s}_i$, calculate distance d (e.g. Euclidean norm) to observation $\boldsymbol{s}_{i'}$.
- Use a *kernel function* $k(\cdot)$ on distances d to define a weighting matrix $\mathbf{W}_i$.
  - Kernel function: *exponential, Gaussian, etc.*
  - Depending on bandwidth b: either for **scaling** (d/b, for some $b > 0$) or **adaptive** with b nearest neighbors of $\boldsymbol{s}_i$ for some $b \in \mathbb{N}$.
- Then:

$$\hat{\beta}(\boldsymbol{s}_i) = (\mathbf{X}^\top \mathbf{W}_i \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{W}_i \boldsymbol{y}$$

Simple Methodology

Geographically Weighted Regression (Brunsdon et al., 1998; Fotheringham et al., 2002):

- For each location $\boldsymbol{s}_i$, calculate distance d (e.g. Euclidean norm) to observation $\boldsymbol{s}_{i'}$.
- Use a *kernel function* $k(\cdot)$ on distances d to define a weighting matrix $\mathbf{W}_i$.
  - Kernel function: *exponential, Gaussian, etc.*
  - Depending on bandwidth b: either for **scaling** (d/b, for some $b > 0$) or **adaptive** with b nearest neighbors of $\boldsymbol{s}_i$ for some $b \in \mathbb{N}$.
- Then:

$$\hat{\beta}(\boldsymbol{s}_i) = (\mathbf{X}^\top \mathbf{W}_i \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{W}_i \boldsymbol{y}$$

Other Methodologies

Overview and Comparison (Dambon et al., 2021b):

- *Bayesian SVC Processes* (Gelfand et al., 2003)
- *Eigenvector Spatial Filtering* (Murakami and Griffith, 2015)
- *SPDE link and INLA* (Lindgren and Rue, 2015; Lindgren et al., 2011)

Goal of Research: Provide a statistical sound methodology that scales both in the number of observations n and the number of SVCs p.

Other Methodologies

Overview and Comparison (Dambon et al., 2021b):

- *Bayesian SVC Processes* (Gelfand et al., 2003)
- *Eigenvector Spatial Filtering* (Murakami and Griffith, 2015)
- *SPDE link and INLA* (Lindgren and Rue, 2015; Lindgren et al., 2011)

Goal of Research: Provide a statistical sound methodology that scales both in the number of observations n and the number of SVCs p.

Gaussian Processes

Idea

Define each coefficients $\beta_j(\cdot)$ as a *Gaussian Process (GP)*, i.e.,

$$\beta_j(\cdot) \sim \mathcal{GP}(\boldsymbol{\mu}_j(\cdot), \boldsymbol{c}_j(\cdot)),$$

with following assumptions:

- constant mean $\boldsymbol{\mu}_j(\cdot) \equiv \mu_j$
- covariance function, similar to kernel in GWR: $c_j(d; \theta_j)$ with covariance parameters $\theta_j = (\rho_j, \sigma_j^2)$ with range ρ_j and variance σ_j^2.
- mutual, prior independence

Additional *nugget* (error term) for ε with variance τ^2.

Idea

Define each coefficients $\beta_j(\cdot)$ as a *Gaussian Process (GP)*, i.e.,

$$\beta_j(\cdot) \sim \mathcal{GP}(\boldsymbol{\mu}_j(\cdot), \boldsymbol{c}_j(\cdot)),$$

with following assumptions:

- constant mean $\boldsymbol{\mu}_j(\cdot) \equiv \mu_j$
- covariance function, similar to kernel in GWR: $\boldsymbol{c}_j(\boldsymbol{d}; \boldsymbol{\theta}_j)$ with covariance parameters $\boldsymbol{\theta}_j = (\rho_j, \sigma_j^2)$ with range ρ_j and variance σ_j^2.
- mutual, prior independence

Additional *nugget* (error term) for ε with variance τ^2.

Idea

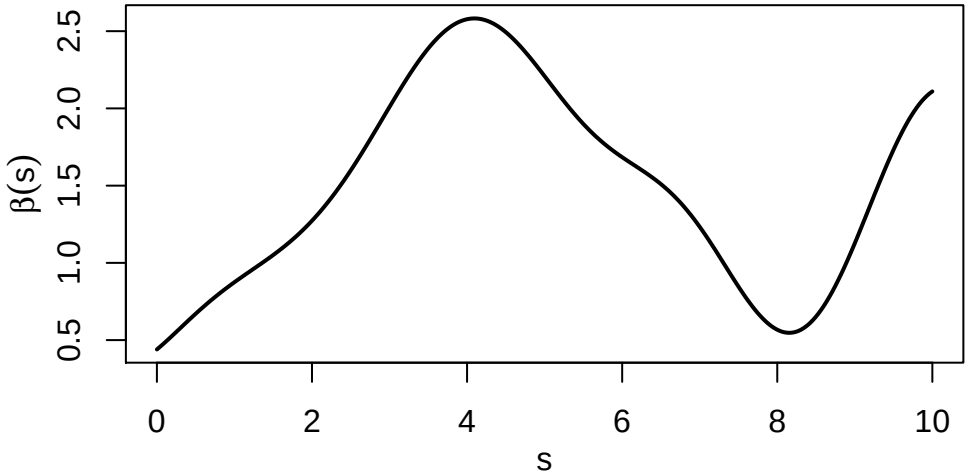
Define each coefficients $\beta_j(\cdot)$ as a *Gaussian Process (GP)*, i.e.,

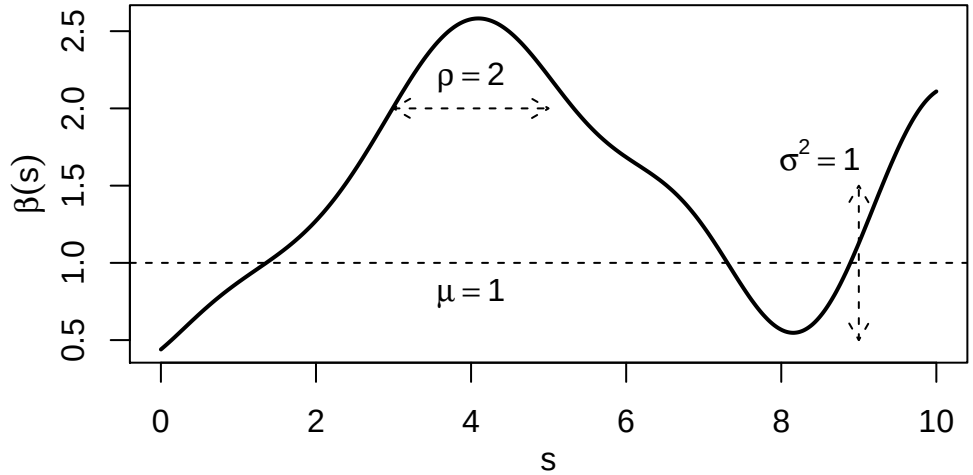
$$\beta_j(\cdot) \sim \mathcal{GP}(\boldsymbol{\mu}_j(\cdot), \boldsymbol{c}_j(\cdot)),$$

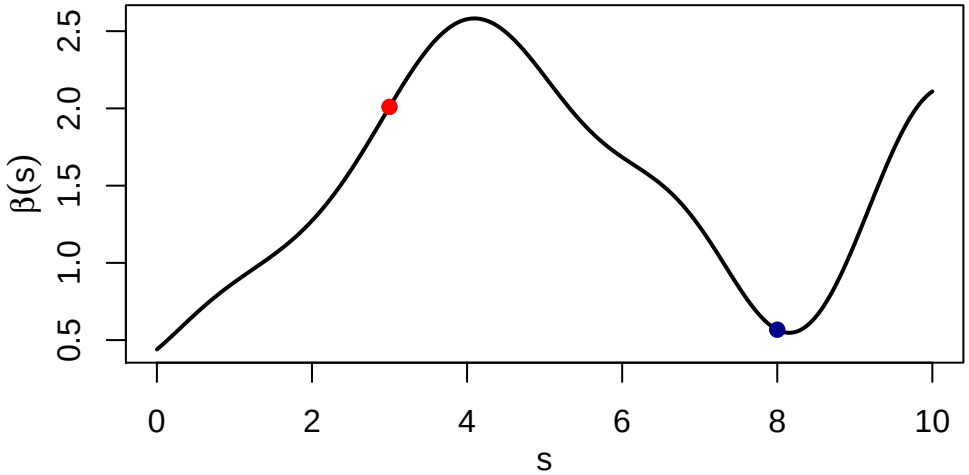
with following assumptions:

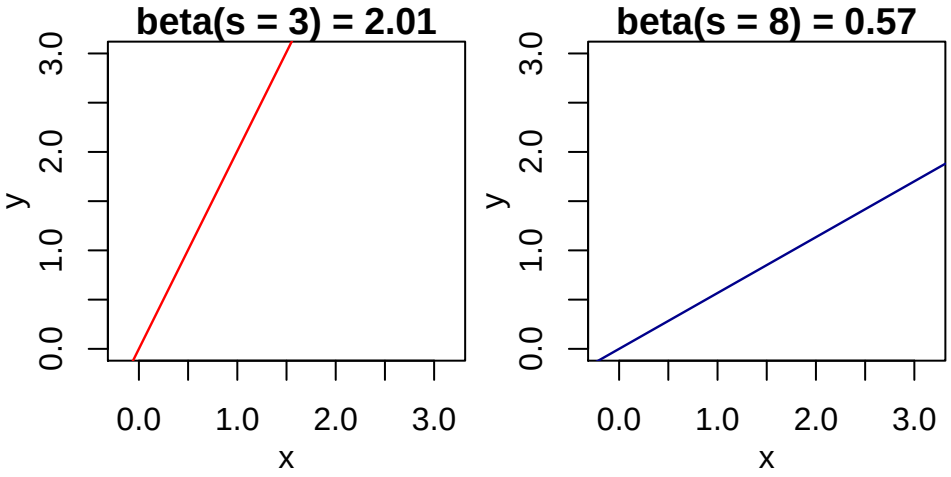
- constant mean $\boldsymbol{\mu}_j(\cdot) \equiv \mu_j$
- covariance function, similar to kernel in GWR: $\boldsymbol{c}_j(\boldsymbol{d}; \boldsymbol{\theta}_j)$ with covariance parameters $\boldsymbol{\theta}_j = (\rho_j, \sigma_j^2)$ with range ρ_j and variance σ_j^2.
- mutual, prior independence

Additional *nugget* (error term) for ε with variance τ^2.


β(s)
0.5
1.0
1.5
2.0
2.5
0
2
4
6
8
10
s


β(s)
0.5
1.0
1.5
2.0
2.5
ρ = 2
σ² = 1
μ = 1
0
2
4
6
8
10
s


β(s)
0.5
1.0
1.5
2.0
2.5
0
2
4
6
8
10
s


beta(s = 3) = 2.01
y
0.0
1.0
2.0
3.0
x
0.0
1.0
2.0
3.0
beta(s = 8) = 0.57
y
0.0
1.0
2.0
3.0
x
0.0
1.0
2.0
3.0

MLE of GP-based SVC Models

Think "Gaussian Processes times p"

All GPs β_j are separated into a pure fixed effect μ_j and a zero-mean GP η_j:

$$\beta_j(\cdot) = \mu_j + \eta_j(\cdot)$$

For a finite set of observations $\boldsymbol{s} = (\boldsymbol{s}_1, ..., \boldsymbol{s}_n)^\top$, the SVCs are given by:

$$\eta_j(\boldsymbol{s}) \sim \mathcal{N}_n(\mathbf{0}_n, (\Sigma_j)_{lm} := c_j(d_{lm} = \|\boldsymbol{s}_l - \boldsymbol{s}_m\|, \theta_j))$$

$$\eta(\boldsymbol{s}) := (\eta_1(\boldsymbol{s})^\top, ..., \eta_p(\boldsymbol{s})^\top)^\top \sim \mathcal{N}_{np}(\mathbf{0}_{np}, \Sigma_\eta := \mathrm{diag}(\Sigma_1, ..., \Sigma_p))$$

Think "Gaussian Processes times p"

All GPs β_j are separated into a pure fixed effect μ_j and a zero-mean GP $\boldsymbol{\eta}_j$:

$$\boldsymbol{\beta}_j(\cdot) = \mu_j + \boldsymbol{\eta}_j(\cdot)$$

For a finite set of observations $\boldsymbol{s} = (\boldsymbol{s}_1, ..., \boldsymbol{s}_n)^\top$, the SVCs are given by:

$$\boldsymbol{\eta}_j(\boldsymbol{s}) \sim \mathcal{N}_n(\mathbf{0}_n, (\boldsymbol{\Sigma}_j)_{lm} := c_j(d_{lm} = \|\boldsymbol{s}_l - \boldsymbol{s}_m\|, \boldsymbol{\theta}_j))$$

$$\boldsymbol{\eta}(\boldsymbol{s}) := (\boldsymbol{\eta}_1(\boldsymbol{s})^\top, ..., \boldsymbol{\eta}_p(\boldsymbol{s})^\top)^\top \sim \mathcal{N}_{np}(\mathbf{0}_{np}, \boldsymbol{\Sigma}_\eta := \mathrm{diag}(\boldsymbol{\Sigma}_1, ..., \boldsymbol{\Sigma}_p))$$

Think "Gaussian Processes times p"

All GPs β_j are separated into a pure fixed effect μ_j and a zero-mean GP $\boldsymbol{\eta}_j$:

$$\boldsymbol{\beta}_j(\cdot) = \mu_j + \boldsymbol{\eta}_j(\cdot)$$

For a finite set of observations $\boldsymbol{s} = (\boldsymbol{s}_1, ..., \boldsymbol{s}_n)^\top$, the SVCs are given by:

$$\boldsymbol{\eta}_j(\boldsymbol{s}) \sim \mathcal{N}_n(\mathbf{0}_n, (\boldsymbol{\Sigma}_j)_{lm} := c_j(d_{lm} = \|\boldsymbol{s}_l - \boldsymbol{s}_m\|, \boldsymbol{\theta}_j))$$

$$\boldsymbol{\eta}(\boldsymbol{s}) := (\boldsymbol{\eta}_1(\boldsymbol{s})^\top, ..., \boldsymbol{\eta}_p(\boldsymbol{s})^\top)^\top \sim \mathcal{N}_{np}(\mathbf{0}_{np}, \boldsymbol{\Sigma}_{\boldsymbol{\eta}} := \mathrm{diag}(\boldsymbol{\Sigma}_1, ..., \boldsymbol{\Sigma}_p))$$

GP-based SVC Model

The **GP-based SVC model** can be written as a **linear mixed model** (see Dambon et al. (2021b) for details):

$$\boldsymbol{y} = \mathbf{X}\boldsymbol{\mu} + \mathbf{W}\boldsymbol{\eta}(\boldsymbol{s}) + \boldsymbol{\varepsilon}. \tag{1}$$

where:

- $\mathbf{X} \in \mathbb{R}^{n \times p}$ is the model matrix
- $\mathbf{W} \in \mathbb{R}^{n \times np}$ is a reparametrized version of $\mathbf{X}$ and a sparse matrix.

GP-based SVC Model

The **GP-based SVC model** can be written as a **linear mixed model** (see Dambon et al. (2021b) for details):

$$\boldsymbol{y} = \mathbf{X}\boldsymbol{\mu} + \mathbf{W}\boldsymbol{\eta}(\boldsymbol{s}) + \boldsymbol{\varepsilon}. \tag{1}$$

where:

- $\mathbf{X} \in \mathbb{R}^{n \times p}$ is the model matrix
- $\mathbf{W} \in \mathbb{R}^{n \times np}$ is a reparametrized version of $\mathbf{X}$ and a sparse matrix.

Maximum Likelihood Estimation

Derive log likelihood of model (1):

$$\mathbf{Y} \sim \mathcal{N}_n \left(\mathbf{X}\boldsymbol{\mu}, \boldsymbol{\Sigma}_{\mathbf{Y}}(\boldsymbol{\theta}) := \sum_{j=1}^{p} \left(\boldsymbol{w}^{(j)} \boldsymbol{w}^{(j)\top} \right) \odot \boldsymbol{\Sigma}_j) + \tau^2 \mathbf{I}_{n\times n} \right)$$

$$\ell(\boldsymbol{\omega}) = -\frac{1}{2} \left(n \log(2\pi) + \log\det \boldsymbol{\Sigma}_{\mathbf{Y}}(\boldsymbol{\theta}) + (\boldsymbol{y} - \mathbf{X}\boldsymbol{\mu})^\top \boldsymbol{\Sigma}_{\mathbf{Y}}(\boldsymbol{\theta})^{-1} (\boldsymbol{y} - \mathbf{X}\boldsymbol{\mu}) \right)$$

Computationally efficient *maximum likelihood estimation* (Dambon et al., 2021b,c):

$$\hat{\omega}(\text{MLE}) = \underset{\omega\in\Omega}{\operatorname{argmax}}\, \ell(\boldsymbol{\omega}),$$

with $\boldsymbol{\omega} := (\rho_1, \sigma_1^2, ..., \rho_p, \sigma_p^2, \tau^2, \mu_1, ..., \mu_p) \in \Omega := \left(\mathbb{R}_{>0} \times \mathbb{R}_{\geq 0}\right)^p \times \mathbb{R}_{>0} \times \mathbb{R}^p$.

Maximum Likelihood Estimation

Derive log likelihood of model (1):

$$\mathbf{Y} \sim \mathcal{N}_n \left(\mathbf{X}\boldsymbol{\mu}, \boldsymbol{\Sigma}_{\mathbf{Y}}(\boldsymbol{\theta}) := \sum_{j=1}^{p} \left(\boldsymbol{w}^{(j)} \boldsymbol{w}^{(j)\top} \right) \odot \boldsymbol{\Sigma}_j) + \tau^2 \mathbf{I}_{n\times n} \right)$$

$$\ell(\boldsymbol{\omega}) = -\frac{1}{2} \left(n \log(2\pi) + \log\det \boldsymbol{\Sigma}_{\mathbf{Y}}(\boldsymbol{\theta}) + (\boldsymbol{y} - \mathbf{X}\boldsymbol{\mu})^\top \boldsymbol{\Sigma}_{\mathbf{Y}}(\boldsymbol{\theta})^{-1} (\boldsymbol{y} - \mathbf{X}\boldsymbol{\mu}) \right)$$

Computationally efficient *maximum likelihood estimation* (Dambon et al., 2021b,c):

$$\hat{\boldsymbol{\omega}}(\text{MLE}) = \underset{\boldsymbol{\omega}\in\Omega}{\operatorname{argmax}}\, \ell(\boldsymbol{\omega}),$$

with $\boldsymbol{\omega} := (\rho_1, \sigma_1^2, ..., \rho_p, \sigma_p^2, \tau^2, \mu_1, ..., \mu_p) \in \Omega := \left(\mathbb{R}_{>0} \times \mathbb{R}_{\geq 0}\right)^p \times \mathbb{R}_{>0} \times \mathbb{R}^p$.

How does it become computationally efficient?

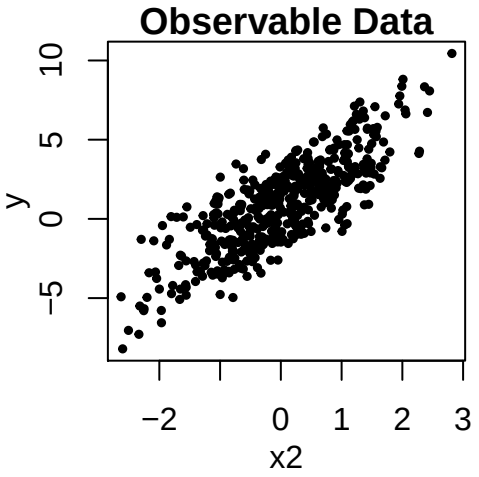
- Optimizing **profile likelihood**, i.e., over covariance parameters θ.
  - Lower dimension of parameter space
- Using **covariance tapering** (Furrer et al., 2006)
  - Induces *sparse matrices*
  - Faster Cholesky decomposition
- Using **parallelized optimization** (R package `optimParallel`, Gerber and Furrer, 2019)
  - Faster gradient approximation

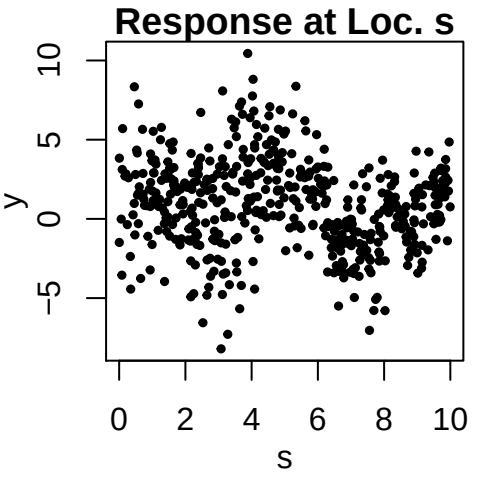
How does it become computationally efficient?

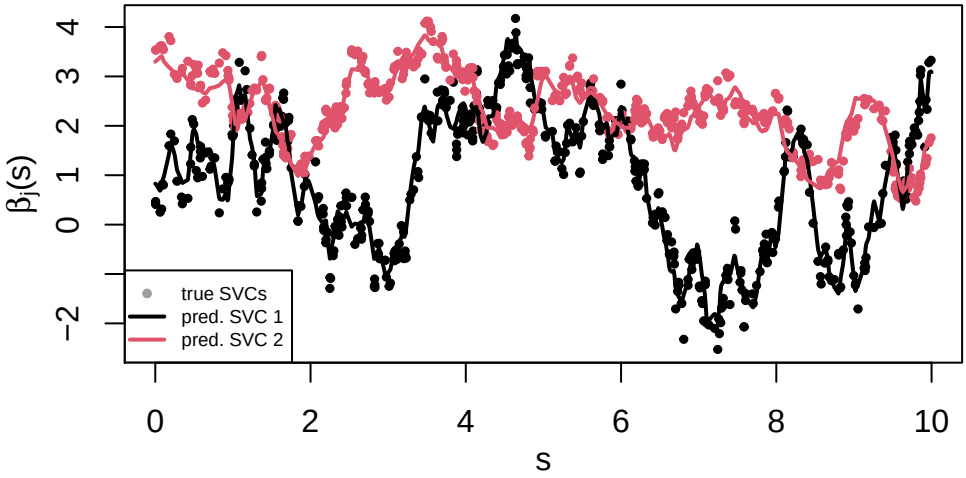
- Optimizing **profile likelihood**, i.e., over covariance parameters θ.
  - Lower dimension of parameter space
- Using **covariance tapering** (Furrer et al., 2006)
  - Induces *sparse matrices*
  - Faster Cholesky decomposition
- Using **parallelized optimization** (R package `optimParallel`, Gerber and Furrer, 2019)
  - Faster gradient approximation

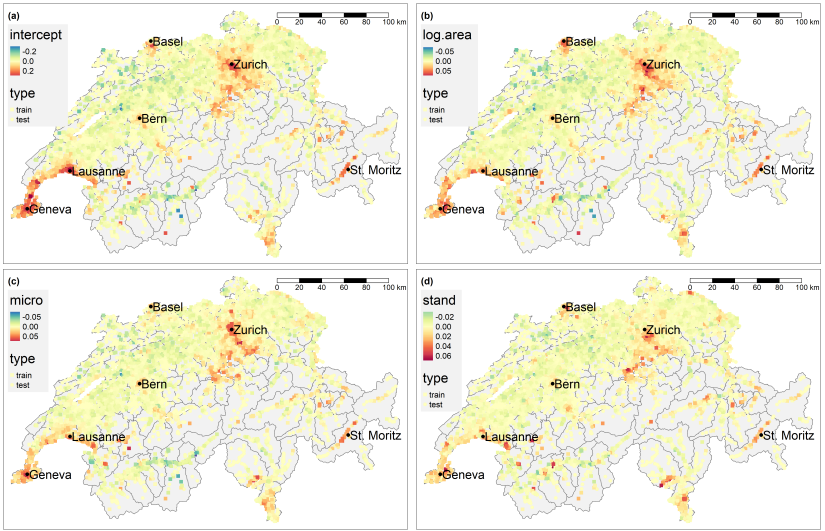
How does it become computationally efficient?

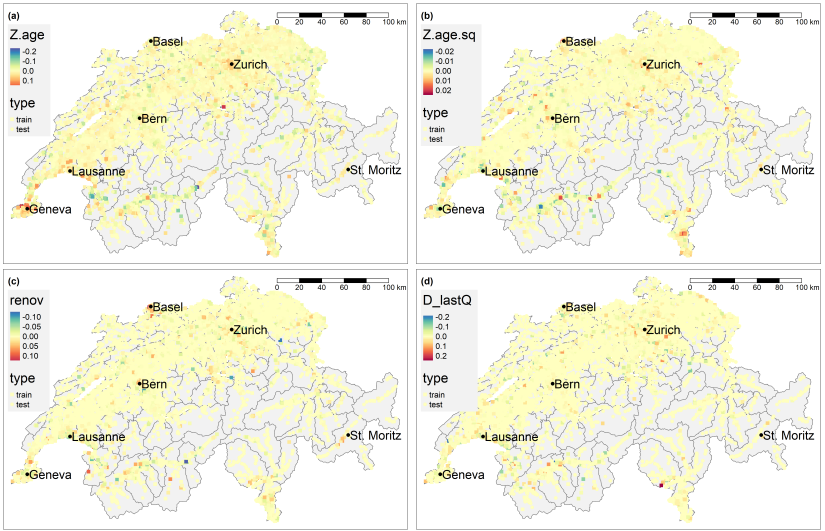
- Optimizing **profile likelihood**, i.e., over covariance parameters θ.
  - Lower dimension of parameter space
- Using **covariance tapering** (Furrer et al., 2006)
  - Induces *sparse matrices*
  - Faster Cholesky decomposition
- Using **parallelized optimization** (R package `optimParallel`, Gerber and Furrer, 2019)
  - Faster gradient approximation


Observable Data
y
10
5
0
−5
−2
0
1
2
3
x2


Response at Loc. s
y
10
5
0
−5
0
2
4
6
8
10
s


β_j(s)
4
3
2
1
0
−2
true SVCs
pred. SVC 1
pred. SVC 2
0
2
4
6
8
10
s


(a)
intercept
-0.2
0.0
0.2
type
train
test
0 20 40 60 80 100 km
Basel
Zurich
Bern
Lausanne
St. Moritz
Geneva
(b)
log.area
-0.05
0.00
0.05
type
train
test
0 20 40 60 80 100 km
Basel
Zurich
Bern
Lausanne
St. Moritz
Geneva
(c)
micro
-0.05
0.00
0.05
type
train
test
0 20 40 60 80 100 km
Basel
Zurich
Bern
Lausanne
St. Moritz
Geneva
(d)
stand
-0.02
0.00
0.02
0.04
0.06
type
train
test
0 20 40 60 80 100 km
Basel
Zurich
Bern
Lausanne
St. Moritz
Geneva


(a)
Z.age
-0.2
-0.1
0.0
0.1
type
train
test
0 20 40 60 80 100 km
Basel
Zurich
Bern
Lausanne
St. Moritz
Geneva
(b)
Z.age.sq
-0.02
-0.01
0.00
0.01
0.02
type
train
test
0 20 40 60 80 100 km
Basel
Zurich
Bern
Lausanne
St. Moritz
Geneva
(c)
renov
-0.10
-0.05
0.00
0.05
0.10
type
train
test
0 20 40 60 80 100 km
Basel
Zurich
Bern
Lausanne
St. Moritz
Geneva
(d)
D_lastQ
-0.2
-0.1
0.0
0.1
0.2
type
train
test
0 20 40 60 80 100 km
Basel
Zurich
Bern
Lausanne
St. Moritz
Geneva

Arising Question

Which of the given covariates do indeed have a non-zero spatially varying coefficient?

Variable Selection for GP-based SVC Models

Variable Selection

- For fixed effects: Effect Present or not?

$$\mu_j \neq 0 \text{ or } \mu_j = 0$$

- For random effects: SVC present or flat?

$$\sigma_j^2 > 0 \text{ or } \sigma_j^2 = 0$$

What's new?

Penalties for μ_j and σ_j^2, similar to LASSO (Tibshirani, 1996).

Penalized maximum likelihood estimation:

$$p\ell(\boldsymbol{\omega}) = \ell(\boldsymbol{\omega}) - n\sum_{j=1}^{p}\lambda_\mu|\mu_j| - n\sum_{j=1}^{p}\lambda_\theta|\sigma_j^2|$$

$$\hat{\omega}(\text{PMLE}) = \underset{\omega\in\Omega}{\operatorname{argmax}}\, p\ell(\boldsymbol{\omega}),$$

with *shrinkage parameters* $\lambda_\mu, \lambda_\theta > 0$.

What's new?

Penalties for μ_j and σ_j^2, similar to LASSO (Tibshirani, 1996).

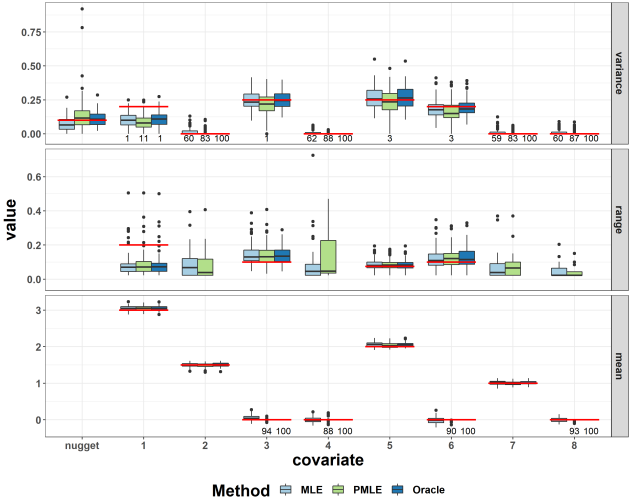
Penalized maximum likelihood estimation:

$$p\ell(\boldsymbol{\omega}) = \ell(\boldsymbol{\omega}) - n\sum_{j=1}^{p} \lambda_{\boldsymbol{\mu}} |\mu_j| - n\sum_{j=1}^{p} \lambda_{\boldsymbol{\theta}} |\sigma_j^2|$$

$$\hat{\boldsymbol{\omega}}(\text{PMLE}) = \underset{\boldsymbol{\omega}\in\Omega}{\operatorname{argmax}}\, p\ell(\boldsymbol{\omega}),$$

with *shrinkage parameters* $\lambda_{\boldsymbol{\mu}}, \lambda_{\boldsymbol{\theta}} > 0$.

SVC Model Selection: Parameter Estimates



Overview

- MLE of GP-based SVC Models (Dambon et al., 2021b)
  - Further Analysis of Real Estate Pricing (Dambon et al., 2020)
- Joint Variable Selection for GP-based SVC Models (Dambon et al., 2021a)
- Software Implementation: R package `varycoef` (Dambon et al., 2021c)

Thank you for your attention!

Overview

- MLE of GP-based SVC Models (Dambon et al., 2021b)
  - Further Analysis of Real Estate Pricing (Dambon et al., 2020)
- Joint Variable Selection for GP-based SVC Models (Dambon et al., 2021a)
- Software Implementation: R package `varycoef` (Dambon et al., 2021c)

Thank you for your attention!

References I

Bischl, B., Richter, J., Bossek, J., Horn, D., Thomas, J., and Lang, M. (2017). mlrMBO: A Modular Framework for Model-Based Optimization of Expensive Black-Box Functions. http://arxiv.org/abs/1703.03373.

Brunsdon, C., Fotheringham, S., and Charlton, M. (1998). Geographically weighted regression. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 47(3):431–443.

Dambon, J. A., Fahrländer, S. S., Karlen, S., Lehner, M., Schlesinger, J., Sigrist, F., and Zimmermann, A. (2020). Examining the Vintage Effect in Hedonic Pricing using Spatially Varying Coefficients Models: A Case Study of Single-Family Houses in the Canton of Zurich. https://www.researchgate.net/publication/346392838_Examining_the_Vintage_Effect_in_Hedonic_Pricing_using_Spatially_Varying_Coefficients_Models_A_Case_Study_of_Single-Family_Houses_in_the_Canton_of_Zurich.

Dambon, J. A., Sigrist, F., and Furrer, R. (2021a). Joint Variable Selection of both Fixed and Random Effects for Gaussian Process-based Spatially Varying Coefficient Models. http://arxiv.org/abs/2101.01932.

Dambon, J. A., Sigrist, F., and Furrer, R. (2021b). Maximum Likelihood Estimation of Spatially Varying Coefficient Models for Large Data with an Application to Real Estate Price Prediction. *Spatial Statistics*, 41:100470.

Dambon, J. A., Sigrist, F., and Furrer, R. (2021c). `varycoef`: Modeling Spatially Varying Coefficients. R package version 0.3.0, https://cran.r-project.org/package=varycoef.

Fotheringham, A. S., Brunsdon, C., and Charlton, M. (2002). *Geographically Weighted Regression: The Analysis of Spatially Varying Relationships*. Wiley, Chichester.

References II

Furrer, R., Genton, M. G., and Nychka, D. W. (2006). Covariance Tapering for Interpolation of Large Spatial Datasets. *Journal of Computational and Graphical Statistics*, 15(3):502–523.

Gelfand, A. E., Kim, H.-J., Sirmans, C. F., and Banerjee, S. (2003). Spatial modeling with spatially varying coefficient processes. *Journal of the American Statistical Association*, 98(462):387–396.

Gerber, F. and Furrer, R. (2019). `optimParallel`: An R Package Providing a Parallel Version of the L-BFGS-B Optimization Method. *The R Journal*, 11(1):352–358.

Lindgren, F. K. and Rue, H. (2015). Bayesian spatial modelling with R-INLA. *Journal of Statistical Software*, 63(19):1–25.

Lindgren, F. K., Rue, H., and Lindström, J. (2011). An Explicit Link between Gaussian Fields and Gaussian Markov Random Fields: The Stochastic Partial Differential Equation Approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(4):423–498.

Murakami, D. and Griffith, D. A. (2015). Random effects specifications in eigenvector spatial filtering: A simulation study. *Journal of Geographical Systems*, 17(4):311–331.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288.