POLITECNICO
MILANO 1863

Iterative Methods for the Mixed-Effect Model in Spatial Regression with Partial Differential Equation Regularization

M.Sc. Thesis of
Martino Ischia

Supervisor:
Prof. Laura Sangalli

Co-supervisor:
Dr. Eleonora Arnone

Programme:
Mathematical Engineering

Department of Mathematics
Politecnico di Milano

Martino Ischia
Iterative Methods for the Mixed-Effect Model in Spatial Regression with Partial Differential Equation Regularization

Contents

List of Figures

Abstract

aa

Sommario

Traduzione dell'abstract in italiano.

1. Synopsis

Here goes the synopsis of your thesis. It consists of an introductory part followed by a description on the thesis content.

This thesis is organized as follows.

2. My chapter

2.1. Functional

...by minimizing the following functional:

$$J_{\Omega_i,\lambda}\left(\boldsymbol{\beta}', \mathbf{b}'_1, \ldots, \mathbf{b}'_{\mathbf{m}}, f_1, \ldots, f_m\right) = \sum_{i=1}^{m}\left(\sum_{j=1}^{n_i}\left(z_{ij} - \mathbf{w}'^{\top}_{\mathbf{ij}}\boldsymbol{\beta}' - \mathbf{v}'^{\top}_{\mathbf{ij}}\mathbf{b}'_{\mathbf{i}} - f_i(\mathbf{p_{ij}})\right)^2 + \lambda\int_{\Omega_i}\Delta f_i\,(\mathbf{p})^2\,d\Omega_i\right) \tag{2.1}$$

2.2. Matrices

Assuming W'_i and V'_i for $i = 1, \ldots, m$ full rank, define the following matrices:

$$X = \begin{bmatrix} W'_1 & V'_1 & 0 & \ldots & 0 & 0 \\ W'_2 & 0 & V'_2 & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ W'_{m-1} & 0 & 0 & \ldots & V'_{m-1} & 0 \\ W'_m & 0 & 0 & \ldots & 0 & V'_m \end{bmatrix} \tag{2.2}$$

with $X \in \mathbb{R}^{N\times(m-1)p+q}$, $H = X\left(X^{\top}X\right)^{-1}X^{\top}$ and $Q = I_N - H$, $\in \mathbb{R}^{N\times N}$, which are the matrixes that project a vector, respectively, onto the subspace spanned by the columns of X and onto its orthogonal complement with respect to $\mathbb{R}^N$. Notice that despite the matrix $X^{\top}X$ exhibits a pattern where only the first row, the first column and the diagonal are different from 0, the inverse of this type of matrix is usually dense (*might be interesting studying an incomplete factorization of this matrix though*).

The discrete problem leads to the solution of the following linear system of equations:

$$\begin{bmatrix} \tilde{\Psi}^{\top}Q\tilde{\Psi} & -\lambda\tilde{R}_1 \\ -\lambda\tilde{R}_1 & -\lambda\tilde{R}_0 \end{bmatrix}\begin{bmatrix} \hat{\mathbf{f}} \\ \hat{\mathbf{g}} \end{bmatrix} = \begin{bmatrix} \tilde{\Psi}^{\top}Q\mathbf{z} \\ 0 \end{bmatrix} \tag{2.3}$$

We call it monolithic because the number of units might be big and so the number of nodes for each unit. Therefore our aim is to avoid the solution of such high dimension linear system, in favour of splitting it into many systems of lower dimension. In case the dimensions of the monolithic system are treatable, the Woodbury decomposition formula, described in appendix A.1, can be used to speed up the computation of the solution for different values of λ. The decomposition is entirely analogous to the one

described in the appendix, with

$$E = \begin{bmatrix} \tilde{\Psi}^T\tilde{\Psi} & -\lambda\tilde{R}_1 \\ -\lambda\tilde{R}_1 & -\lambda\tilde{R}_0 \end{bmatrix} \quad U = \begin{bmatrix} \tilde{\Psi}^T X \\ 0 \end{bmatrix} \tag{2.4}$$

$$C = -\left[(X^T X)^{-1}\right] \qquad V = U^T$$

Once the unknown spatial field $\hat{\mathbf{f}}$ has been obtained, the vector of unknown coefficients $\hat{\boldsymbol{\nu}} = (\boldsymbol{\beta}, \mathbf{b_1}, \ldots, \mathbf{b_m})$ can be computed solving a least-squares problem: the corresponding normal equations are written as

$$\hat{\boldsymbol{\nu}} = \left(X^T X\right)^{-1} X^T \left(z - \hat{\mathbf{f}}_N\right) \tag{2.5}$$

where I recall that $\hat{\mathbf{f}}_N = \tilde{\Psi}\hat{\mathbf{f}}$

2.3. Iterative methods

Following the ideas stemmed from the spatio-temporal regression in Pollini and Ponti [PP13] and Massardi and Spaziani [MS21], we consider an iterative scheme. At each step i the scheme computes an approximate solution $(\hat{\mathbf{f}}^i, \hat{\mathbf{g}}^i)$ of the monolithic system by solving a single-unit problem for each statistical unit. The algorithm stops with two possible criteria:

- A maximum number of iterations is reached;
- The following two conditions are true.
  The first one is that the functional 2.1 evaluated in the estimated solution $J^i = J_{\Omega_i,\lambda}\left(\hat{\boldsymbol{\beta}}'^i, \hat{\mathbf{b}}_1'^i, \ldots, \hat{\mathbf{b}}_m'^i, \hat{f}_1^i, \ldots, \hat{f}_m^i\right)$ has reached stagnation, that is the relative increment $\left(J^i - J^{i-1}\right)/J^i$ is below a certain threshold (in the code such threshold is an input parameter, it was set to 10^{-8}). The term $\int_{\Omega_i} \Delta f_i(\mathbf{p})^2 \, d\Omega_i$ of the functional 2.1 is computed by exploiting the expansion into the finite element basis functions, leading to $\hat{\mathbf{g}}_i^T R_0 \hat{\mathbf{g}}_i$, where $\hat{\mathbf{g}}_i$ is the sub-vector of $\hat{\mathbf{g}}$ corresponding to unit i.
  The second one is that the estimated solution is very close to the exact solution of the system 2.3, condition that is verified by checking that the residual, normalized by the Euclidian norm of the right-hand side, is below a certain threshold (this is another input parameter, 10^{-8} was used).

The details of a first possible implementation are described in the following section.

2.3.1. The block diagonal approach

Looking at the monolithic equation 2.3, the question of how to formulate an approximation of the term $\tilde{\Psi}^T Q \tilde{\Psi}$ naturally arises. In particular, a suitable block approximation

allows to make the estimated field $\hat{f}_i$ of statistical unit i independent from the observations in the other units, for every unit i.

To this purpose, the following approximation is first considered:

$$\tilde{\Psi}^T Q \tilde{\Psi} \simeq \Gamma := \begin{bmatrix} \Psi^T Q_1 \Psi & 0 & \dots & 0 \\ 0 & \Psi^T Q_2 \Psi & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \Psi^T Q_m \Psi \end{bmatrix} \tag{2.6}$$

Here Q_i indicates the i-th diagonal block of Q of dimension $n \times n$. By definition of Q, it is equal to $I_n - X_i \left(X^T X\right)^{-1} X_i^T$, with $X_i = X\left((i-1)\,m + 1 : im\,,\ :\right)$, where the typical notation of the MATLAB language has been used to express a suitable submatrix.
Another possible way to express the blocks $\Psi^T Q_i \Psi$ of matrix 2.6 is through Woodbury decomposition (appendix A.1) with a use analogous to the one in 2.4. In fact, $\Psi^T Q_i \Psi$ is the i-th block diagonal of matrix $\tilde{\Psi}^T Q \tilde{\Psi}$, which in turn can be expressed as $\tilde{\Psi}^T \tilde{\Psi} - \tilde{\Psi}^T H \tilde{\Psi}$. Thus, defining $U = \tilde{\Psi}^T X$, $C = -\left(X^T X\right)^{-1}$,$V = U^T$, matrix $\Psi^T Q_i \Psi$ can be expressed as $\Psi^T \Psi + U_i C V_i$, where $U_i = U\left((i-1)\,m + 1 : im\,,\ :\right)$, $V_i = U_i^T$.

2.3.2. Initialization

The inizialization consists in finding a good guess $(\hat{\mathbf{f}}^0, \hat{\mathbf{g}}^0)$ to start the algorithm from; for this purpose, the following m problems are solved: for $i = 1, \dots, m$ solve

$$\begin{bmatrix} \Psi^T Q_i \Psi & -\lambda R_1 \\ -\lambda R_1 & -\lambda R_0 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{f}}_i^0 \\ \hat{\mathbf{g}}_i^0 \end{bmatrix} = \begin{bmatrix} \mathbf{u}_i \\ 0 \end{bmatrix} \tag{2.7}$$

where $\mathbf{u}_i$ is the vector whose components are the first n components of $\tilde{\Psi}^T Q \mathbf{z}$ starting from the $n(i-1) + 1$-th component.

2.3.3. Iterations

The idea behind the iterative scheme is, having a guess of a solution $(\hat{\mathbf{f}}^{k-1}, \hat{\mathbf{g}}^{k-1})$, to compute a new guess $(\hat{\mathbf{f}}^k, \hat{\mathbf{g}}^k)$ by replacing the mutual interaction of the system variables corresponding to different units with their contibution given by their previous value. For example, the first n equations of the monolithic system 2.3 read

$$\begin{bmatrix} \Psi^T Q_{1,1} \Psi & \Psi^T Q_{1,2} \Psi & \dots & \Psi^T Q_{1,m} \Psi & -\lambda R_1 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{f}}_1 \\ \hat{\mathbf{f}}_2 \\ \vdots \\ \hat{\mathbf{f}}_m \\ \hat{\mathbf{g}}_1 \end{bmatrix} = \begin{bmatrix} \mathbf{u}_1 \end{bmatrix} \tag{2.8}$$

where $Q_{i,j}$ indicates block of row i and column j of matrix Q, which also coincides with $-H_{i,j}$ for $i \neq j$. Substituting $\hat{\mathbf{f}}_j$ for $j = 1, \dots, m$ with $\hat{\mathbf{f}}_j^{k-1}$, taking it to the right-hand side

and generalizing, the iterative scheme reads: for $i = 1, \dots, m$ solve

$$\begin{bmatrix} \Psi^T Q_i \Psi & -\lambda R_1 \\ -\lambda R_1 & -\lambda R_0 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{f}}_i^k \\ \hat{\mathbf{g}}_i^k \end{bmatrix} = \begin{bmatrix} \mathbf{r}_i \\ 0 \end{bmatrix} \tag{2.9}$$

where

$$\mathbf{r}_i = \mathbf{u}_i - \sum_{\substack{j=1 \\ j \neq i}}^{m} \Psi^T Q_{i,j} \Psi \hat{\mathbf{f}}_j^{k-1} \tag{2.10}$$

An estimate of $\hat{\mathbf{v}} = \boldsymbol{\beta}, \mathbf{b_1}, \dots, \mathbf{b_m}$ has to be computed at each iteration for the estimation of the functional 2.1, according to equation 2.5.

2.3.4. The iterative method as a preconditioned Richardson method

Given a generic preconditioning matrix P, calling $\mathbf{r}_k$ the residual at step k, Richardson method consists in solving (or rather trying to) a linear system characterized by matrix A iterating the following steps:

1. Solve $P\mathbf{z}_k = \mathbf{r}_k$
2. Compute the acceleration parameter α_k (for simplicity we can use $\alpha_k = 1$)
3. Update the solution $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{z}_k$
4. Update the residual $\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k A \mathbf{z}_k$

The iterative scheme described in the previous section is a Richardson scheme with the following preconditioning matrix:

$$P = \begin{bmatrix} \Gamma & -\lambda \tilde{R}_1 \\ -\lambda \tilde{R}_1 & -\lambda \tilde{R}_0 \end{bmatrix} \tag{2.11}$$

Solving a linear system involving this matrix is indeed like solving m linear systems of dimensions $2n \times 2n$

3. Simulation studies

In this chapter we see some examples...

4. Applications

aa

5. Conclusions

Aaa

Appendices

A. Formulae

A.1. Woodbury decomposition

The following matrix identity holds.

Proposition A.1 (Woodbury matrix identity)**.** *Let* M *be a square* $m \times m$ *matrix which can be written as the sum* $E + UCV$*, with* E *being* $m \times m$*,* U *being* $m \times n$*,* C *being a square* $n \times n$ *matrix, and* V $n \times m$*. Then*

$$(E + UCV)^{-1} = E^{-1} - E^{-1}U\left(C^{-1} + VE^{-1}U\right)^{-1}VE^{-1} \tag{A.1}$$

This equation is exploited for faster system solving in the `fdaPDE` library. In the simple space-only problem described *e.g.* in [San21], the presence of covariates leads to a linear system involving the following matrix M:

$$M = \begin{bmatrix} \Psi^T Q \Psi & -\lambda R_1 \\ -\lambda R_1 & -\lambda R_0 \end{bmatrix}$$

Remembering that the projection matrix Q is defined as $I - H$, M can be split into the following two components, one independent from λ:

$$M = \begin{bmatrix} \Psi^T \Psi & -\lambda R_1 \\ -\lambda R_1 & -\lambda R_0 \end{bmatrix} + \begin{bmatrix} -\Psi^T H \Psi & 0 \\ 0 & 0 \end{bmatrix}$$

By calling E the matrix on the left (which is also the matrix corresponding to the problem without covariates), and remembering that $H = W\left(W^TW\right)^{-1}W^T$, Woodbury decomposition can be exploited defining the following matrices U, C, V:

$$\begin{bmatrix} -\Psi^T W\left(W^TW\right)^{-1}W^T\Psi & 0 \\ 0 & 0 \end{bmatrix} = UCV$$

$$U = \begin{bmatrix} \Psi^T W \\ 0 \end{bmatrix}$$

$$C = -\left[\left(W^TW\right)^{-1}\right]$$

$$V = U^T = \begin{bmatrix} W^T\Psi & 0 \end{bmatrix}$$

where the 0s indicate matrices of zeros of suitable dimensions.
In this way, when computing the solution of the system for different values of λ, only the factorization of the matrix E will have to be computed.

Bibliography

[MS21] M. Massardi and S. Spaziani. *fdaPDE: an efficient iterative method for spatio-temporal regression with PDE regularization*. Politecnico di Milano, APSC course, 2021 (cit. on p. 16).

[PP13] M. Pollini and L. Ponti. *Regressione spazio-temporale su domini bidimensionali non planari*. Politecnico di Milano, NAPDE course, 2013 (cit. on p. 16).

[San21] L. Sangalli. "Spatial Regression With Partial Differential Equation Regularisation". In: *International Statistical Review* (Mar. 2021). DOI: 10.1111/insr.12444 (cit. on p. 27).