

Rolling Gaussian Process Regression with Application to Regime Shifts

William Menke

Lamont-Doherty Earth Observatory of Columbia University, Palisades, NY, USA

Email: menke@ldeo.columbia.edu

How to cite this paper: Menke, W. (2022) Rolling Gaussian Process Regression with Application to Regime Shifts. *Applied Mathematics*, 13, 859-868.
<https://doi.org/10.4236/am.2022.1311054>

Received: October 13, 2022

Accepted: November 6, 2022

Published: November 9, 2022

Copyright © 2022 by author(s) and Scientific Research Publishing Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).
<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Gaussian Process Regression (GPR) can be applied to the problem of estimating a spatially-varying field on a regular grid, based on noisy observations made at irregular positions. In cases where the field has a weak time dependence, one may desire to estimate the present-time value of the field using a time window of data that rolls forward as new data become available, leading to a sequence of solution updates. We introduce “rolling GPR” (or moving window GPR) and present a procedure for implementing that is more computationally efficient than solving the full GPR problem at each update. Furthermore, regime shifts (sudden large changes in the field) can be detected by monitoring the change in posterior covariance of the predicted data during the updates, and their detrimental effect is mitigated by shortening the time window as the variance rises, and then decreasing it as it falls (but within prior bounds). A set of numerical experiments is provided that demonstrates the viability of the procedure.

Keywords

Rolling Gaussian Process Regression, Regime Shift, Moving Window Analysis, Woodbury Identity, Bordering Method

1. Introduction

Gaussian Process Regression (GPR) is a popular data assimilation method that can be used to reconstruct a field, $m(x_1, x_2, \dots) \in \mathbb{R}$, that varies with spatial coordinates, $x_i \in \mathbb{R}$ ($1 \leq i \leq K$) [1] [2] [3] [4]. Observations of the field at particular positions, together with prior information on its value at these positions and its spatial correlation function, are used to estimate the field at arbitrary positions. GPR is conceptually similar to interpolation, but is not true interpolation because the reconstructed field does not, in general, reproduce the observations

(except for the special case called Kriging [1]). This behavior is advantageous in the case of noisy data, because the reconstruction contains fewer defects caused by data outliers.

In many real-world applications, the field that is being reconstructed also has time, $t \in \mathbb{R}$, dependence, that is, $m(t, x_1, x_2, \dots)$. Furthermore, observations often are not made synchronously, but rather in a sporadic and ongoing manner. When the goal is to estimate the current state of the field, older measurements become obsolete and only the most recent observations are relevant. The choice of the length of the observation window, T , affects both the resolution of the reconstruction (for resolution improves with the number of data) and the variance of the reconstruction (for the inclusion of new data decreases variance but the inclusion of obsolete data increases it). The issue of window length is particularly acute when the field is relatively stable, except for undergoing sporadic reorganizations. These so-called regime shifts are common features of fields associated with the Earth's biosphere [5] [6] [7], climate system [8] [9] and geodynamo [10] [11].

We use a formulation of GPR [3] in which the M model parameters, $\mathbf{m} = \begin{bmatrix} \mathbf{m}^{(t)}; \mathbf{m}^{(c)} \end{bmatrix} \in \mathbb{R}^{M+N}$, and corresponding spatial coordinates, $\mathbf{x} = \begin{bmatrix} \mathbf{x}^{(t)}; \mathbf{x}^{(c)} \end{bmatrix} \in \mathbb{R}^{M+N}$ (where the semicolon implies vertical concatenation), are divided into target, t , groups of length M and control, c , groups of length N . Only the control group is observed by data, \mathbf{d} . The control points represent observations of the field at irregularly-spaced positions and the target points represent its values on a regularly-spaced grid. A prior estimate of the model parameters, $\hat{\mathbf{m}}$, their covariance, \mathbf{C}_m , and the covariance of the data, $\sigma_d^2 \mathbf{I}$, are assumed to be known. The GPR estimate of the model parameters is then:

$$\Delta \mathbf{m} \equiv \begin{bmatrix} \Delta \mathbf{m}^{(t)} \\ \Delta \mathbf{m}^{(c)} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_m^{(tc)} \\ \mathbf{C}_m^{(cc)} \end{bmatrix} \mathbf{A}^{-1} \Delta \mathbf{d} \quad \text{with } \mathbf{A} \equiv \mathbf{C}_m^{(cc)} + \sigma_d^2 \mathbf{I}$$

$$\text{and } \mathbf{m}^{est} = \Delta \mathbf{m} + \hat{\mathbf{m}} \quad \text{and } \Delta \mathbf{d} \equiv \mathbf{d} - \hat{\mathbf{m}}^{(c)} \quad \text{and } \mathbf{d}^{pre} = \begin{bmatrix} \mathbf{m}^{(c)} \end{bmatrix}^{est} \quad (1)$$

$$\mathbf{C}_{m^{est}}^{(cc)} = \sigma_d^2 \mathbf{C}_m^{(cc)} \mathbf{A}^{-2} \mathbf{C}_m^{(cc)} \quad \text{and } \mathbf{C}_{m^{est}}^{(tt)} = \sigma_d^2 \mathbf{C}_m^{(tc)} \mathbf{A}^{-2} \mathbf{C}_m^{(tc)T}$$

The formula for the solution, $\Delta \mathbf{m}$, should be evaluated from right to left for maximum computational efficiency. The matrix inverse, \mathbf{A}^{-1} , appears both in the estimated solution, \mathbf{m}^{est} , and its posterior covariance, $\mathbf{C}_{m^{est}}$ [4], which is an important diagnostic of the solution's quality. In typical cases, \mathbf{A} is far from being sparse, so when N is large, \mathbf{A}^{-1} can be computationally expensive to compute. Alternative methods that omit the calculation of \mathbf{A}^{-1} , such as solving the linear system, $\mathbf{A} \mathbf{u} = \Delta \mathbf{d}$, and then calculating $\Delta \mathbf{m} = \begin{bmatrix} \mathbf{C}_m^{(tc)}; \mathbf{C}_m^{(cc)} \end{bmatrix} \mathbf{u}$, are nearly as expensive and do not provide a (simple) pathway for computing the posterior covariance.

Suppose that we know the GPR solution when the control group has N_A element, ordered by time of observation. Our goal is to compute an updated solution after we delete the first N_1 observations from the group, leaving $N_2 = N_A - N_1$,

and then append N_3 elements, so that it now has $N_B = N_2 + N_3$ elements. This process, which we term *rolling GPR* (or moving-window GPR) is repeated many times as new data become available and old data are deemed obsolete. The process ensures that the estimate of the field is kept up-to-date.

The structure of this paper is as follows: The process of performing rolling GPR is detailed in Methodology section, and issues concerning its implementation are discussed. This process is divided into four conceptually distinct parts: discarding obsolete data from the GPR problem, appending new data, detecting and correcting the error in the solution, and detecting and responding to regime changes through changes in window length. The complete process of performing rolling GPR is outlined in Procedure section. Examples section provides a demonstration of the technique, applied to the reconstruction of a two-dimensional field with a regime shift, using an exemplary synthetic dataset and a Gaussian prior covariance function. Finally, a discussion of the efficiency of the discard-append process, together with summary remarks, are presented in Discussion and Conclusion section.

2. Methodology

Discarding Obsolete Data. The quantities $\hat{\mathbf{m}}, \mathbf{x}^{(c)}, \mathbf{d}, \mathbf{C}_m^{(tc)}, \mathbf{C}_m^{(cc)}, \mathbf{A}$ and \mathbf{A}^{-1} all need to be modified when data are discarded. The vectors, $\hat{\mathbf{m}}^{(c)}$ and \mathbf{d} are modified by retaining only their last N_2 elements:

$$\begin{aligned} \hat{\mathbf{m}}^{(c)} \equiv \begin{bmatrix} \hat{\mathbf{m}}^{(c_1)} \\ \hat{\mathbf{m}}^{(c_2)} \end{bmatrix} \text{ is replaced with } \hat{\mathbf{m}}^{(c_2)} \\ \mathbf{d} \equiv \begin{bmatrix} \mathbf{d}^{(1)} \\ \mathbf{d}^{(2)} \end{bmatrix} \text{ is replaced with } \mathbf{d}^{(2)} \end{aligned} \tag{2}$$

Here, c_1 and c_2 refer to the deleted group and retained group, respectively. The corresponding \mathbf{x} s must be modified in an identical manner. Similarly, only one block within the covariances matrices is retained:

$$\begin{aligned} \mathbf{C}_m^{(tc)} \equiv \begin{bmatrix} \mathbf{C}_m^{(tc_1)} & \mathbf{C}_m^{(tc_2)} \end{bmatrix} \text{ is replaced with } \mathbf{C}_m^{(tc_2)} \\ \mathbf{C}_m^{(cc)} \equiv \begin{bmatrix} \mathbf{C}_m^{(c_1c_1)} & \mathbf{C}_m^{(c_1c_2)} \\ \mathbf{C}_m^{(c_2c_1)} & \mathbf{C}_m^{(c_2c_2)} \end{bmatrix} \text{ is replaced with } \mathbf{C}_m^{(c_2c_2)} \end{aligned} \tag{3}$$

The updating of \mathbf{A}^{-1} requires more effort, but uses only well-known techniques. At the start of the process, the $N_A \times N_A$ matrix, \mathbf{A} , and its inverse, \mathbf{A}^{-1} , are known. These matrices are partitioned into submatrices:

$$\mathbf{A} \equiv \begin{bmatrix} \mathbf{X} & \mathbf{Z}^T \\ \mathbf{Z} & \mathbf{Y} \end{bmatrix} \text{ and } \mathbf{A}^{-1} \equiv \begin{bmatrix} \mathbf{P} & \mathbf{R}^T \\ \mathbf{R} & \mathbf{Q} \end{bmatrix} \tag{4}$$

Here, \mathbf{X} and \mathbf{P} are $N_1 \times N_1$, \mathbf{Y} and \mathbf{Q} are $N_2 \times N_2$, and \mathbf{Z} and \mathbf{R} are $N_2 \times N_1$, with $N_1 + N_2 = N_A$. We mention an identity involving $\mathbf{Z}, \mathbf{R}, \mathbf{P}$ and \mathbf{Q} that will be used later in the paper:

$$\mathbf{Z}^T \mathbf{R} \mathbf{P}^{-1} [\mathbf{I} - \mathbf{R}^T \mathbf{Z}] + \mathbf{Z}^T \mathbf{Q} \mathbf{Z} = \mathbf{Z}^T \mathbf{R} \mathbf{P}^{-1} - \mathbf{Z}^T \mathbf{R} \mathbf{P}^{-1} \mathbf{R}^T \mathbf{Z} + \mathbf{Z}^T \mathbf{Q} \mathbf{Z} = \mathbf{0} \quad (5)$$

It can be derived by multiplying out the block matrix form of $\mathbf{A}^{-1} \mathbf{A} = \mathbf{I}$, solving the diagonal elements for \mathbf{X} and \mathbf{Y} , and substituting the result into the off-diagonal elements. The process of removing the first N_1 data corresponds to replacing \mathbf{A} with \mathbf{Y} , and \mathbf{A}^{-1} with \mathbf{Y}^{-1} . An efficient method for computing \mathbf{Y}^{-1} can be designed using the Woodbury identity [12]:

$$(\mathbf{M} + \mathbf{U} \mathbf{W} \mathbf{V})^{-1} = \mathbf{M}^{-1} - \mathbf{M}^{-1} \mathbf{U} (\mathbf{W}^{-1} + \mathbf{V} \mathbf{M}^{-1} \mathbf{U})^{-1} \mathbf{V} \mathbf{M}^{-1} \quad (6)$$

Here, \mathbf{M} , \mathbf{W} , \mathbf{U} and \mathbf{V} are conformable matrices. The reader may confirm by direct substitution that:

$$\begin{aligned} \mathbf{W} = \mathbf{I} \text{ and } \mathbf{M} = \mathbf{A} \text{ and } (\mathbf{M} + \mathbf{U} \mathbf{W} \mathbf{V})^{-1} &= \begin{bmatrix} \mathbf{X}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{Y}^{-1} \end{bmatrix} \\ \mathbf{U}_{N_2, N_1} = \begin{bmatrix} \mathbf{I}_{N_1, N_1} & \mathbf{0}_{N_1, N_1} \\ \mathbf{0}_{N_2, N_1} & -\mathbf{Z}_{N_2, N_1} \end{bmatrix} \text{ and } \mathbf{V}_{N_1, N} = \begin{bmatrix} \mathbf{0}_{N_1, N_1} & -[\mathbf{Z}^T]_{N_1, N_2} \\ \mathbf{I}_{N_1, N_1} & \mathbf{0}_{N_1, N_2} \end{bmatrix} \end{aligned} \quad (7)$$

are consistent choices of the several matrices (with the subscripts indicating the sizes of selected matrices). Thus, \mathbf{Y}^{-1} can be read from the lower-right-hand block of $\mathbf{A}^{-1} - \mathbf{A}^{-1} \mathbf{U} (\mathbf{I} + \mathbf{V} \mathbf{A}^{-1} \mathbf{U})^{-1} \mathbf{V} \mathbf{A}^{-1}$. Note that the matrix, $(\mathbf{I} + \mathbf{V} \mathbf{A}^{-1} \mathbf{U})$, is $(2N_1) \times (2N_1)$, so that its inverse requires less computational effort than does the $N_2 \times N_2$ matrix, \mathbf{Y} , as long as a relatively few data are being removed. Below, we will develop an analytic formula for $(\mathbf{I} + \mathbf{V} \mathbf{A}^{-1} \mathbf{U})^{-1}$ that further reduces the size of the necessary matrices to $N_1 \times N_1$. An explicit formula for \mathbf{Y}^{-1} is obtained by manipulating the block matrix form of the Woodbury formula, starting with:

$$\begin{aligned} [\mathbf{I} + \mathbf{V} \mathbf{A}^{-1} \mathbf{U}] &= \mathbf{I} + \begin{bmatrix} \mathbf{0} & -\mathbf{Z}^T \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{P} & \mathbf{R}^T \\ \mathbf{R} & \mathbf{Q} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & -\mathbf{Z} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{N_1, N_1} - \mathbf{Z}^T \mathbf{R} & \mathbf{Z}^T \mathbf{Q} \mathbf{Z} \\ \mathbf{P} & [\mathbf{I}_{N_1, N_1} - \mathbf{R}^T \mathbf{Z}] \end{bmatrix} \end{aligned} \quad (8)$$

Note that the off-diagonal elements of this matrix are symmetric, and the off-diagonal elements are transposes of one another. Its inverse has the form:

$$\begin{aligned} [\mathbf{I} + \mathbf{V} \mathbf{A}^{-1} \mathbf{U}]^{-1} &= \begin{bmatrix} \mathbf{I}_{N_1, N_1} & \mathbf{C}_{N_1, N_1} \\ \mathbf{D}_{N_1, N_1} & \mathbf{I}_{N_1, N_1} \end{bmatrix} \\ \text{with } \mathbf{C} &= \mathbf{Z}^T \mathbf{R} \mathbf{P}^{-1} = \mathbf{Z}^T \mathbf{R} \mathbf{P}^{-1} \mathbf{R}^T \mathbf{Z} - \mathbf{Z}^T \mathbf{Q} \mathbf{Z} \\ \text{and } \mathbf{D} &= \mathbf{R}^T \mathbf{Z} [\mathbf{Z}^T \mathbf{Q} \mathbf{Z}]^{-1} = -[\mathbf{P}^{-1} - \mathbf{C}]^{-1} \end{aligned} \quad (9)$$

The formulas for \mathbf{C} and \mathbf{D} were derived by multiplying out the block diagonal form of $[\mathbf{I} + \mathbf{V} \mathbf{A}^{-1} \mathbf{U}]^{-1} [\mathbf{I} + \mathbf{V} \mathbf{A}^{-1} \mathbf{U}] = \mathbf{I}$, solving the diagonal elements for \mathbf{C} and \mathbf{D} , and applying identity Equation (5). Note that both \mathbf{C} and \mathbf{D} are symmetric matrices. The products $\mathbf{A}^{-1} \mathbf{U}$ and $\mathbf{V} \mathbf{A}^{-1}$ have the forms:

$$\begin{aligned} \mathbf{A}^{-1}\mathbf{U} &= \begin{bmatrix} \mathbf{P} & \mathbf{R}^T \\ \mathbf{R} & \mathbf{Q} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & -\mathbf{Z} \end{bmatrix} = \begin{bmatrix} \mathbf{P} & -\mathbf{R}^T\mathbf{Z} \\ \mathbf{R} & -\mathbf{QZ} \end{bmatrix} \equiv \begin{bmatrix} \mathbf{P} & \mathbf{S} \\ \mathbf{R} & \mathbf{T} \end{bmatrix} \\ \mathbf{VA}^{-1} &= \begin{bmatrix} \mathbf{0} & -\mathbf{Z}^T \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{P} & \mathbf{R}^T \\ \mathbf{R} & \mathbf{Q} \end{bmatrix} = \begin{bmatrix} -\mathbf{Z}^T\mathbf{R} & -\mathbf{Z}^T\mathbf{Q} \\ \mathbf{P} & \mathbf{R}^T \end{bmatrix} \equiv \begin{bmatrix} \mathbf{S}^T & \mathbf{T}^T \\ \mathbf{P} & \mathbf{R}^T \end{bmatrix} \end{aligned} \tag{10}$$

with $\mathbf{S} \equiv -\mathbf{R}^T\mathbf{Z}$ and $\mathbf{T} = -\mathbf{QZ}$

Thus, the expression, $\mathbf{A}^{-1}\mathbf{U}(\mathbf{I} + \mathbf{VA}^{-1}\mathbf{U})^{-1}\mathbf{VA}^{-1}$, can be simplified to:

$$\begin{aligned} \mathbf{A}^{-1}\mathbf{U}(\mathbf{I} + \mathbf{VA}^{-1}\mathbf{U})^{-1}\mathbf{VA}^{-1} &= \begin{bmatrix} \mathbf{P} & \mathbf{S} \\ \mathbf{R} & \mathbf{T} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{C} \\ \mathbf{D} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{S}^T & \mathbf{T}^T \\ \mathbf{P} & \mathbf{R}^T \end{bmatrix} \\ &= \begin{bmatrix} \left\{ \mathbf{P}(\mathbf{S}^T + \mathbf{CP}) + \mathbf{S}(\mathbf{DS}^T + \mathbf{P}) \right\} & \left\{ \mathbf{P}(\mathbf{T}^T + \mathbf{CR}^T) + \mathbf{S}(\mathbf{DT}^T + \mathbf{R}^T) \right\} \\ \left\{ \mathbf{R}(\mathbf{S}^T + \mathbf{CP}) + \mathbf{T}(\mathbf{DS}^T + \mathbf{P}) \right\} & \left\{ \mathbf{R}(\mathbf{T}^T + \mathbf{CR}^T) + \mathbf{T}(\mathbf{DT}^T + \mathbf{R}^T) \right\} \end{bmatrix} \end{aligned} \tag{11}$$

The lower right-hand block of Equation (11) becomes the new \mathbf{A}^{-1} :

$$\begin{aligned} \mathbf{A} &\text{ is replaced with } \mathbf{Y} \\ \mathbf{A}^{-1} &\text{ is replaced with } \mathbf{Y}^{-1} = \mathbf{Q} - \left((\mathbf{RT}^T + \mathbf{TR}^T) + \mathbf{RCR}^T + \mathbf{TDT}^T \right) \end{aligned} \tag{12}$$

This formula has been tested numerically, and was found to be correct to within machine precision. Accuracy and speed are improved through the use of coding techniques that utilize and enforce the symmetry of all the symmetric matrices.

Appending New Data. As before, the quantities $\hat{\mathbf{m}}^{(c_2)}$, $\mathbf{x}^{(c_2)}$, $\mathbf{d}^{(2)}$, $\mathbf{C}_m^{(tc_2)}$, $\mathbf{C}_m^{(c_2c_2)}$, \mathbf{A} and \mathbf{A}^{-1} all need to be modified when new data are appended. The vectors, $\hat{\mathbf{m}}^{(c)}$ and \mathbf{d} are modified by appending N_3 elements:

$$\hat{\mathbf{m}}^{(c_2)} \text{ is replaced with } \hat{\mathbf{m}}^{(c)} \equiv \begin{bmatrix} \hat{\mathbf{m}}^{(c_2)} \\ \hat{\mathbf{m}}^{(c_3)} \end{bmatrix} \text{ and } \mathbf{d}^{(2)} \text{ is replaced with } \mathbf{d} \equiv \begin{bmatrix} \mathbf{d}^{(2)} \\ \mathbf{d}^{(3)} \end{bmatrix} \tag{13}$$

The corresponding \mathbf{x} s must be modified in an identical manner. Similarly, new blocks are appended to the covariances matrices:

$$\begin{aligned} \mathbf{C}_m^{(tc_2)} \text{ is replaced with } \mathbf{C}_m^{(tc)} &\equiv \begin{bmatrix} \mathbf{C}_m^{(tc_2)} & \mathbf{C}_m^{(tc_3)} \end{bmatrix} \\ \mathbf{C}_m^{(c_2c_2)} \text{ is replaced with } \mathbf{C}_m^{(cc)} &\equiv \begin{bmatrix} \mathbf{C}_m^{(c_2c_2)} & \mathbf{C}_m^{(c_2c_3)} \\ \mathbf{C}_m^{(c_3c_2)} & \mathbf{C}_m^{(c_3c_3)} \end{bmatrix} \end{aligned} \tag{14}$$

As before, the updating of \mathbf{A}^{-1} requires more effort, but use a well-known technique based on the Bordering method [13] [14]. We rename the existing matrix, \mathbf{A} , to \mathbf{X} , as it will become the upper-left block of the modified \mathbf{A} . Both \mathbf{X} and its inverse, \mathbf{X}^{-1} , are known $N_2 \times N_2$ matrices. We seek to add the blocks, \mathbf{Y} and \mathbf{Z} , associated with the N_3 new data, creating an augmented $N_B \times N_B$ matrix, \mathbf{A} , with $N_B = N_2 + N_3$. The matrix, \mathbf{A} , and its inverse satisfy:

$$\mathbf{A}^{-1}\mathbf{A} \equiv \begin{bmatrix} \mathbf{P} & \mathbf{R}^T \\ \mathbf{R} & \mathbf{Q} \end{bmatrix} \begin{bmatrix} \mathbf{X} & \mathbf{Z}^T \\ \mathbf{Z} & \mathbf{Y} \end{bmatrix} = \begin{bmatrix} \mathbf{I}_{N_2N_2} & \mathbf{0}_{N_2N_3} \\ \mathbf{0}_{N_3N_2} & \mathbf{I}_{N_3N_3} \end{bmatrix} \tag{15}$$

Multiplying out the equation and solving for \mathbf{P} , \mathbf{Q} and \mathbf{R} yields:

$$\begin{aligned}
 \mathbf{P}\mathbf{X} + \mathbf{R}^T\mathbf{Z} &= \mathbf{I} \text{ so } \mathbf{P} = [\mathbf{I} - \mathbf{R}^T\mathbf{Z}] \mathbf{X}^{-1} \\
 \mathbf{P}\mathbf{Z}^T + \mathbf{R}^T\mathbf{Y} &= \mathbf{0} \text{ so } \mathbf{R}^T = -\mathbf{P}\mathbf{Z}^T\mathbf{Y}^{-1} \\
 \mathbf{R}\mathbf{X} + \mathbf{Q}\mathbf{Z} &= \mathbf{0} \text{ so } \mathbf{R} = -\mathbf{Q}\mathbf{Z}\mathbf{X}^{-1} \\
 \mathbf{R}\mathbf{Z}^T + \mathbf{Q}\mathbf{Y} &= \mathbf{I} \text{ so } -\mathbf{Q}\mathbf{Z}\mathbf{X}^{-1}\mathbf{Z}^T + \mathbf{Q}\mathbf{Y} = \mathbf{I} \text{ so } \mathbf{Q} = [\mathbf{Y} - \mathbf{Z}\mathbf{X}^{-1}\mathbf{Z}^T]^{-1}
 \end{aligned}
 \tag{16}$$

The matrices then become:

$$\begin{aligned}
 \mathbf{A} \text{ is replaced with } & \begin{bmatrix} \mathbf{X} & \mathbf{Z}^T \\ \mathbf{Z} & \mathbf{Y} \end{bmatrix} \\
 \mathbf{A}^{-1} \text{ is replaced with } & \begin{bmatrix} \mathbf{P} & \mathbf{R}^T \\ \mathbf{R} & \mathbf{Q} \end{bmatrix}
 \end{aligned}
 \tag{17}$$

These formulas have been tested numerically, and are correct to within machine precision. Accuracy and speed are improved through the use of coding techniques that utilize and enforce the symmetry of all the symmetric matrices.

Detecting and Correcting of Errors in the Solution. Numerical tests (not shown) indicate that delete/append process can be repeated many (e.g. hundreds) of times without significant loss of precision, at least for the class of matrices encountered in GPR. However, as a precaution, we recommend that the inverse be corrected every few hundred iterations, either through the direct calculation of \mathbf{A}^{-1} from \mathbf{A} , or through one application of the Schultz method [15] [16] [17]:

$$\mathbf{A}^{-1} \text{ is replaced by } \mathbf{A}^{-1} (2\mathbf{I} - \mathbf{A}\mathbf{A}^{-1})
 \tag{18}$$

One way to monitor accuracy is to compute the absolute value of just one (or a few) of the diagonal elements of the error matrix, $\Phi \equiv \mathbf{A}\mathbf{A}^{-1} - \mathbf{I}$. For fixed i , quantity $|\Phi_{ii}|$ can be computed very efficiently, because only one inner product (between the i th row of \mathbf{A} and the i th column of \mathbf{A}^{-1}) need be performed. The correction process then can be initiated when $|\Phi_{ii}|$ exceeds a threshold, chosen to be small fraction, say 10^{-8} .

Readjusting Window Length. The posterior data variance,

$$\sigma^2 = N_B^{-1} \mathbf{e}^T \mathbf{e} \text{ with } \mathbf{e} = \mathbf{d} - \mathbf{d}^{pre}
 \tag{19}$$

is a measure of how well the reconstruction fits the data. The quantity, $\chi^2 \equiv N_B \sigma^2 / \sigma_d^2$, is approximately chi-squared distributed with N_B degrees of freedom, and therefore has an expected value of N_B and variance of $2N_B$. Thus, the expected value of σ^2 is σ_d^2 , its variance is $2\sigma_d^4 / N_B$ and its 95% confidence bound is $\sigma_{95}^2 = \sigma_d^2 + 2\sigma_d^2 [2/N_B]^{1/2}$. However, σ^2 can be expected to rise well above this bound during a regime shift due to model error, that is, two distinct spatial patterns are being comingled and cannot be fit simultaneously. Shortening the window length tends to bring σ^2 closer to the bound, because obsolete data are being discarded more rapidly. This suggests the strategy of defining a threshold, guided by the value of σ_{95} , and decreasing the window length when E is above it (by setting $N_1 > N_3$ when) and increasing it once E has dropped below it (by setting $N_1 < N_3$ until N_B has reached some upper limit).

3. Procedure

Step 1. Decide upon initial values of N_1 , N_2 , N_3 . The choice $N_1 = N_3$ implies that the same number of data are appended as are discarded, leaving the window length, $N_B = N_2 + N_3$ unchanged. The rolling process is most efficient when $N_2 \gg N_1$ and $N_2 \gg N_3$.

Step 2. Solve the GPR problem for an initial group of $N_A = N_1 + N_2$ data, as in Equation (1).

Step 3. Discard N_1 data by modifying $\hat{\mathbf{m}}$ and \mathbf{d} , and their corresponding \mathbf{x} s, as in Equation (2), $\mathbf{C}_m^{(tc)}$ and $\mathbf{C}_m^{(cc)}$ as in Equation (3) and \mathbf{A} and \mathbf{A}^{-1} as in Equation (12).

Step 4. Append N_3 data by modifying $\hat{\mathbf{m}}$ and \mathbf{d} , and their corresponding \mathbf{x} s, as in Equation (13), $\mathbf{C}_m^{(tc)}$ and $\mathbf{C}_m^{(cc)}$ as in Equation (14) and \mathbf{A} and \mathbf{A}^{-1} as in Equation (17).

Step 5. (Optional) Monitor the error $|\Phi_{ii}|$, where $\Phi \equiv [\mathbf{A}\mathbf{A}^{-1} - \mathbf{I}]$, for a single index, i , and when it exceeds a threshold of say, 10^{-8} , refine \mathbf{A}^{-1} as in Equation (18).

Step 6. Solve the GPR problem for $N_B = N_2 + N_3$ data, obtaining \mathbf{m}^{est} and \mathbf{d}^{pre} as in Equation (1).

Step 7. (Optional) Compute the posterior data variance, σ^2 , as in Equation (19) and reassign the values of N_1 , N_2 and N_3 , as needed (as discussed in the *Readjusting Window Length* section).

Step 8. Once N_3 new are available, iterate the procedure, starting at Step 3.

4. Examples

Test Scenario. In these examples, the two dimensional field, $m(x, y)$, $0 \leq x \leq 30$, $0 \leq y \leq 30$ is reconstructed on a regular 30×30 grid, using data that are acquired at the steady rate of 10 observations per time interval, Δt . The observations are made at randomly chosen positions and have uncorrelated and uniform error with prior variance, $\sigma_d^2 = 10^{-4}$. The true field experiences a regime shift at time, $25\Delta t$, when it abruptly changes from a four-lobed to a three-lobed pattern:

$$m(t, x, y) = \begin{cases} \sin(2\pi x/L)\sin(2\pi y/L) & (t < 25) \\ \sin(\pi x/L)\sin(3\pi y/L) & (t \geq 25) \end{cases} \text{ with } L = 30 \quad (20)$$

The field is assumed to have zero prior value and Gaussian prior covariance:

$$\hat{\mathbf{m}} = \mathbf{0} \text{ and } C_{ij} = \gamma^2 \exp\left(-\frac{r_{ij}^2}{2s^2}\right) \quad (21)$$

$$\text{with } r_{ij}^2 = (x_i - x_j)^2 + (y_i - y_j)^2 \text{ and } \gamma^2 = \frac{1}{3} \text{ and } s = 0.22$$

Example without Window Length Readjustment. In this example (**Figure 1(A)**), the size of the rolling set of observations increases with time up to an upper bound of 90. For times, $t < 25\Delta t$, the field is correctly reconstructed, showing the correct four-lobed pattern, and the posterior variance is about equal to the

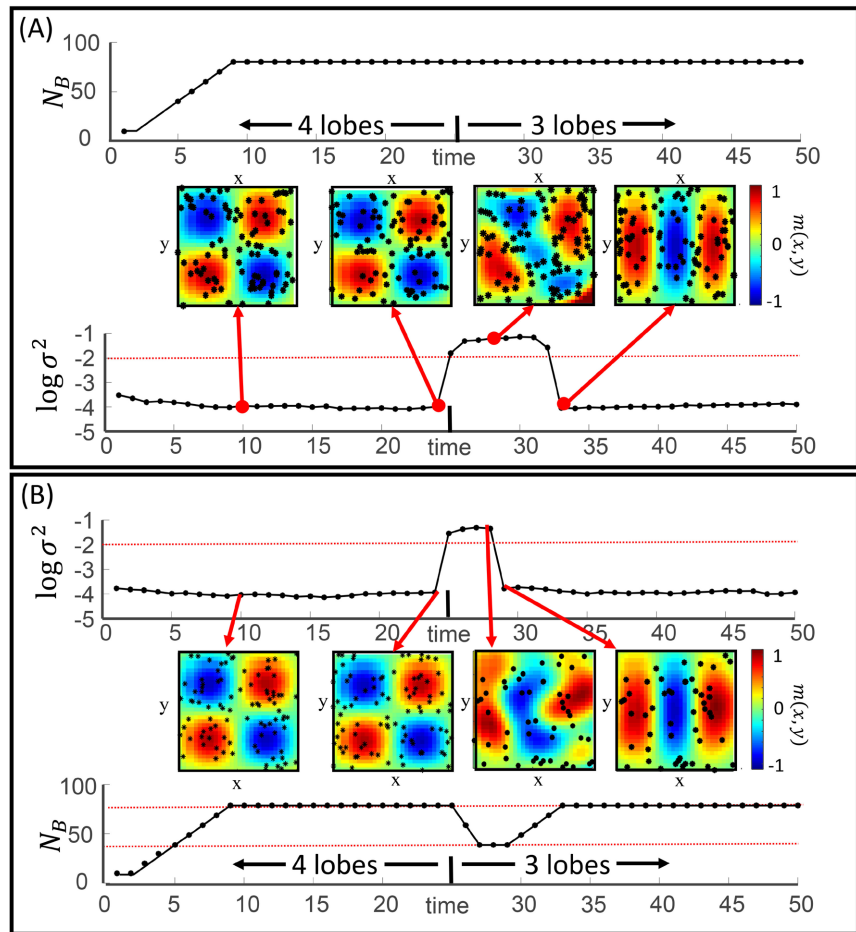


Figure 1. Example of rolling GPR. (A) Case where the window length is held constant. The window length (top curve) grows to $N_b = 90$ and then is held constant. The two-dimensional field, $m(x, y)$ (colors), abruptly changes from a four-lobed pattern to a three-lobed pattern, at time, $t = 25$. The posterior data variance, σ^2 (bottom curve), is low, except near the time of the change. (B) Case where the window length is varied. The posterior data variance, σ^2 (top black curve), and the time at which it exceeds a threshold (top red curve) is detected. The window length (bottom black curve) is decreased during the interval of high variance, and then increased afterwards, within bounds (bottom two red curves).

prior variance. The field is incorrectly reconstructed during the time interval, $25\Delta t \leq t \leq 33\Delta t$, when the window comingles the two patterns, and the posterior variance is about one thousand times higher than the prior variance. For times, $t \geq 34\Delta t$, the field is correctly reconstructed, showing the correct three-lobed pattern, and the posterior variance returns to its original level. This example confirms the ability of the method to reconstruct the field in the presence of regime shifts.

Example with Window Length Readjustment. In this example (Figure 1(B)), the posterior data variance, σ^2 (a measure of the misfit of the data), is monitored, and the size of the rolling set of observations is reduced when it increases past a threshold (but not below a lower bound of $N_B = 50$) (Figure 1(B)). Once the

error has declined below the threshold, the set size is allowed to increase, up to an upper bound of $N_B = 90$. The duration of the interval of elevated error is reduced by a factor of about two compared to the first experiment. The presence of the second three-lobed pattern is evident at an earlier time than in the first example, demonstrating the utility of the window length readjustment.

5. Discussion and Conclusions

The main advantage of the rolling Gaussian Process Regression method that we describe here is its ability to reconstruct a time-varying field without any assumptions about its dynamics. This is in contrast to other data assimilation techniques, such as Kalman filtering [18], in which the differential equation describing the time dependence is assumed to be known.

The $N \times N$ matrix, \mathbf{A} , that arises in Gaussian Process Regression is a non-sparse, symmetric positive definite matrix that takes $N_B^3/3 + O(N_B^2)$ floating point operations to invert [19]. Careful counting of operations reveals that the discard/append process described above takes about $6N_A N_B^2 + 4N_B^2 N_C$ operations. Thus, for $N_A = N_C$, it is more efficient when $N_A/N_M \approx 1/30$, that is, when just a few percent of the data are being updated.

The procedure for implementing “rolling GRP” (or moving window GPR) that we present here is more computationally efficient than solving the full GPR problem at each update, at least when the number of data that are deleted/appended is only a few percent of the total number used in the calculation. Regime shifts (sudden large changes in the field) can be detected by monitoring the posterior data variance (a measure of the misfit of the data) during the updates, and their detrimental effect is mitigated by shortening the time window as the variance rises, and then decreasing it as it falls (but within prior bounds). The numerical experiments presented here demonstrate the viability and usefulness of the procedure.

Acknowledgements

The author thanks Roger Creel of Columbia University (New York, USA) for the helpful discussion.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

- [1] Krige, D.G. (1951) A Statistical Approach to Some Basic Mine Valuation Problems on the Witwatersrand. *Journal of the Chemical, Metallurgical and Mining Society of South Africa*, **52**, 119-139.
- [2] Rasmussen, C.E. and Williams, C.K.I. (2006) Gaussian Processes for Machine Learning. The MIT Press, Cambridge, 272 p.
<https://doi.org/10.7551/mitpress/3206.001.0001>

- [3] Menke, W. and Creel, R. (2021) Gaussian Process Regression Reviewed in the Context of Inverse Theory. *Surveys in Geophysics*, **42**, 473-503. <https://doi.org/10.1007/s10712-021-09640-w>
- [4] Menke, W. (2022) Environmental Data Analysis with MATLAB or Python, Principles, Applications and Prospects. 3rd Edition, Elsevier, Amsterdam.
- [5] Scheffer, M., Carpenter, S., Foley, J.A., Folke, C. and Walker, B. (2001) Catastrophic Shifts in Ecosystems. *Nature*, **413**, 591-596. <https://doi.org/10.1038/35098000>
- [6] Beisner, B.E., Haydon, D.T. and Cuddington, K. (2003) Alternative Stable States in Ecology. *Frontiers in Ecology and the Environment*, **1**, 376-382.
- [7] Smol, J.P., Wolfe, A.P., Birks, H.J.B., Douglas, M.S.V., Jones, V.J., Korhola, A., Pietnitz, R., Rühland, K., Sorvari, S., Antoniades, D., Brooks, S.J., Fallu, M.-A., Hughes, M., Keatley, B.E., Laing, T.E., Michelutti, N., Nazarova, L., Nyman, M., Paterson, A.M., Perren, B., Quinlan, R., Rautio, M., Saulnier-Talbot, É., Siitonen, S., Solovieva, N. and Weckström, J. (2005) Climate-Driven Regime Shifts in the Biological Communities of Arctic Lakes. *Proceedings of the National Academy of Sciences of the United States of America*, **102**, 4397-4402. <https://doi.org/10.1073/pnas.0500245102>
- [8] Zhang, Q.-B. and Fang, O. (2020) Tree Rings Circle an Abrupt Shift in Climate. *Science*, **370**, 1037-1038. <https://doi.org/10.1126/science.abf1700>
- [9] Alley, R.B., Marotzke, J., Nordhaus, W.D., Overpeck, J.T., Peteet, D.M., Pielke Jr., R.A., Pierrehumbert, R.T., Rhines, P.B., Stocker, T.F., Talley, L.D. and Wallace, J.M. (2003) Abrupt Climate Change. *Science*, **299**, 2005-2010. <https://doi.org/10.1126/science.1081056>
- [10] Glatzmaier, G.A. and Roberts, P.H. (1995) A Three-Dimensional Self-Consistent Computer Simulation of a Geomagnetic Field Reversal. *Nature*, **377**, 203-209. <https://doi.org/10.1038/377203a0>
- [11] Meynadier, L. (1993) Geomagnetic Field Intensity and Reversals during the Past Four Million Years. *Nature*, **366**, 234-238. <https://doi.org/10.1038/366234a0>
- [12] Woodbury, M. (1950) Inverting Modified Matrices. Memorandum Report 42. Statistical Research Group, Princeton University, Princeton.
- [13] Householder, A.S. (1975) The Theory of Matrices in Numerical Analysis. Dover Publications, New York.
- [14] Westlake, J.R. (1968) A Handbook of Numerical Matrix Inversion and Solution of Linear Equations. John Wiley & Sons, Hoboken.
- [15] Li, R.C. (2014) Matrix Perturbation Theory. In: Hogben, L., Ed., *Handbook of Linear Algebra*, 2nd Edition, CRC Press, Boca Raton.
- [16] Petersen, K.B. and Pedersen, M.S. (2012) The Matrix Cookbook. https://www.academia.edu/42127358/The_Matrix_Cookbook
- [17] Petkovic, M.D. (2014) Generalized Schultz Iterative Methods for the Computation of Outer Inverses. *Computers & Mathematics with Applications*, **67**, 1837-1847. <https://doi.org/10.1016/j.camwa.2014.03.019>
- [18] Grewal, M.S. and Andrews, A.P. (2014) Kalman Filtering: Theory and Practice using MATLAB. 4th Edition, Wiley-IEEE Press, Hoboken.
- [19] Hunger, R. (2007) Floating Point Operations in Matrix-Vector Calculus (Version 1.3). Technical Report, Technische Universität München, Associate Institute for Signal Processing, München.