

A hierarchical Bayesian approach for modeling the evolution of the 7-day moving average of the number of deaths by COVID-19

E. F. Saraiva, L. Sauer & C. A. B. Pereira

To cite this article: E. F. Saraiva, L. Sauer & C. A. B. Pereira (2023) A hierarchical Bayesian approach for modeling the evolution of the 7-day moving average of the number of deaths by COVID-19, Journal of Applied Statistics, 50:10, 2194-2208, DOI: [10.1080/02664763.2022.2070136](https://doi.org/10.1080/02664763.2022.2070136)

To link to this article: <https://doi.org/10.1080/02664763.2022.2070136>



Published online: 06 May 2022.



Submit your article to this journal [↗](#)



Article views: 91



View related articles [↗](#)



View Crossmark data [↗](#)



Citing articles: 1 View citing articles [↗](#)



A hierarchical Bayesian approach for modeling the evolution of the 7-day moving average of the number of deaths by COVID-19

E. F. Saraiva^a, L. Sauer^b and C. A. B. Pereira^c

^aInstituto de Matemática, Universidade Federal de Mato Grosso do Sul, Campo Grande, Brazil; ^bEscola de Administração e Negócios, Universidade Federal de Mato Grosso do Sul, Campo Grande, Brazil; ^cInstituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, Brazil

ABSTRACT

In this paper, we propose a hierarchical Bayesian approach for modeling the evolution of the 7-day moving average for the number of deaths due to COVID-19 in a country, state or city. The proposed approach is based on a Gaussian process regression model. The main advantage of this model is that it assumes that a nonlinear function f used for modeling the observed data is an unknown random parameter in opposite to usual approaches that set up f as being a known mathematical function. This assumption allows the development of a Bayesian approach with a Gaussian process prior over f . In order to estimate the parameters of interest, we develop an MCMC algorithm based on the Metropolis-within-Gibbs sampling algorithm. We also present a procedure for making predictions. The proposed method is illustrated in a case study, in which, we model the 7-day moving average for the number of deaths recorded in the state of São Paulo, Brazil. Results obtained show that the proposed method is very effective in modeling and predicting the values of the 7-day moving average.

ARTICLE HISTORY

Received 24 July 2021

Accepted 19 April 2022

KEYWORDS

COVID-19; Bayesian approach; Gaussian process; predictions; MCMC algorithm

1. Introduction

Especially in the year 2020, many articles were published describing modeling procedures for the number of cases and/or deaths due to COVID-19 in many countries. The interest in this kind of modeling lies mainly in projections that these models may provide and consequently assist government agents in making decisions regarding the intensification of social isolation, the acquisition of hospital equipment, an increase in the number of intensive care units in hospitals, among others.

In general, the published works model the accumulated number of cases (or deaths) by using some nonlinear growth model. For example, Musa *et al.* [5] considered a simple exponential growth model to analyze the initial phase of the epidemic of COVID-19 in Africa, Aviv-Sharon and Aharoni [1] modeled the data from the Philippines and Taiwan using the generalized logistic model, Vasconcelos *et al.* [8] applied the Richards growth

model to the data collected in China, France, Germany, Iran, Italy, South Korea, and Spain, Wu *et al.* [10] calibrated the logistic growth model, the generalized logistic growth model, and the generalized Richards model for the number of cases recorded in China, among others.

Since the growth in the cumulative number of cases and deaths by COVID-19 has, in general, presented a heterogeneous evolution over time, this implies that the adjustment of only one of these growth models may not be adequate to explain the entire study period. This heterogeneous evolution is due to the occurrence of more than one wave of the pandemic or due to the accuracy of the statistical reports on the number of cases and deaths recorded. For example, the number of cases may present a fast growth if the number of diagnostic tests is increased. On the other hand, tests that need to be realized to include the death cases in mortality statistics may take some days. This implies a sub-notification followed by an over-notification. In addition, the number of cases and deaths may be sub-notified on the weekends and appear on the statistical reports only a few days after.

An alternative is modeling, for instance, the 7-day moving average in order to minimize the discrepancies that may be contained in the dataset. In this paper, we assume for the 7-day moving average dataset an additive model composed by a nonlinear function f plus a random error ϵ . However, in opposite to the usual approach that is based on set up f as a known mathematical function, we assume that f is an unknown random parameter. In order to estimate it from the data, we adopt a Bayesian approach putting over the unknown nonlinear function a Gaussian process prior. That is, we are assuming a probability distribution over all possible functions that fit a set of points equally well [9]. In addition, we assume a more hierarchical level putting prior distributions on the parameters of the Gaussian process. The main advantage of assuming a Gaussian process prior over f is that we are estimating f by ‘smooth functions’ obtained by the generation of values of a multivariate normal distribution with an adequate covariance matrix, and linking the generated points by lines.

To estimate all parameters of interest, we developed an MCMC algorithm based on the Metropolis-within-Gibbs sampling algorithm. We also present an MCMC algorithm for making predictions. The proposed method is illustrated in a case study, in which, we model the 7-day moving average recorded in the state of São Paulo, Brazil. Results obtained show that the proposed method is very effective in modeling the values of the 7-day moving average for the number of deaths due to COVID-19. In the course of 481 days of the pandemic, we run the estimation procedure four times, on the 100th, 180th, 280th and 465th days. In these four analyses, the mean square error of the fitted model was smaller than 0.05, indicating a very good performance of the proposed method.

The three main advantages of the proposed method are:

- (1) It is very flexible since it is not restricted to a parametric mathematical function;
- (2) Does not need the fit of a set of parametric models followed by the application of a model comparison procedure;
- (3) It is not too difficult to be implemented computationally since the estimation procedure is based on the use of a Metropolis-within-Gibbs sampling algorithm;

Although we develop the paper with a focus on the 7-day moving average for the number of deaths, the method also can be used for modeling the moving average for the number of cases. In addition, the method is not restricted to the 7-day moving average and a user may use it for modeling a d -day moving average dataset, for $d > 1$.

The remainder of the paper is organized as follows. In Section 2, we present the Bayesian approach for modeling the 7-day moving average for the number of deaths recorded in a country, state and/or city. In this section, we also describe the MCMC algorithms used to estimate the parameters of interest and make predictions. Section 3 presents an application of the proposed method to a case study. Section 4 concludes the paper with the final remarks.

2. Bayesian model for 7-day moving average

Let D_t be the number of deaths by COVID-19 recorded in a country, state or city on the t th day, for $t = 0, \dots, N$, where $t = 0$ represents the day that the first death was recorded, and N is the last day considered in the analysis. Consider $Y_t = \frac{1}{7} \sum_{m=t}^{t+6} D_m$ be the 7-day moving average of the number of deaths due to the covid-19, for $t = 1, \dots, n = N - 6$.

Assume that Y_t values are generated according to the following additive model

$$Y_t = f_t + \epsilon_t \quad (1)$$

where $f_t = f(t)$ is an unknown function and ϵ_t is a random error assumed as being generated from a normal distribution with mean 0 and variance σ^2 , $\epsilon_t \sim \mathcal{N}(0, \sigma^2)$, with $\text{cov}(\epsilon_t, \epsilon_{t'}) = 0$, for $t, t' = 1, \dots, n$ and $t \neq t'$.

At this point, it is usual to complete the model (1) by setting up f_t as a known nonlinear mathematical function. However, there may be several nonlinear parametric models that can fit the observed data points equally well. Due to this, it is common to fit a set of candidate models and then choose the best model using some model selection criteria, such as AIC or BIC. That is, the analysis stays limited to the set of models previously chosen by a user. In addition, the complexity and/or flexibility of the parametric models considered are limited by the number of parameters in the model.

In order to give flexibility to the modeling and not to be restricted to a set of parametric models, hereafter, we assume that the unknown nonlinear function is a parameter of interest. To estimate these parameters from an observed dataset, we assume a Bayesian approach. Thus, consider \mathbb{F}_f be the set of all possible functions that can explain the data. Let \mathbb{P}_f be a probability distribution defined over \mathbb{F}_f , in a way that, a finite set $\mathbf{f} = (f_1, \dots, f_n)$ of \mathbb{F}_f follows a multivariate normal distribution with mean vector \mathbf{m} of dimension $1 \times n$ and covariance Σ_f of dimension $n \times n$ and elements $\sigma(t, t') = \text{cov}(f_t, f_{t'})$, for $t, t' = 1, \dots, n$. In other words, we are assuming that *a priori* $\mathbf{f} \sim \mathcal{N}_n(\mathbf{m}, \Sigma_f)$, i.e. a Gaussian process prior over \mathbf{f} , where $\mathcal{N}_n(\cdot)$ represents the n -variate normal distribution.

Thus, setting up $\mathbf{m} = \mathbf{0}$ in order to represent our noninformative prior knowledge about the expected value of \mathbf{f} and letting Σ_f be an unknown quantity, we propose the following hierarchical Bayesian model

$$\begin{aligned} \mathbf{Y} | \mathbf{f}, \sigma_\epsilon^2 &\sim \mathcal{N}_n(\mathbf{f}, \sigma_\epsilon^2 \mathbb{I}); \\ \mathbf{f} | \Sigma_f &\sim \mathcal{N}_n(\mathbf{0}, \Sigma_f) \end{aligned}$$

$$\begin{aligned}\sigma_\epsilon^2 &\sim \mathcal{IG}\left(\frac{a}{2}, \frac{b}{2}\right); \\ \Sigma_{\mathbf{f}} | \gamma, \sigma_{\mathbf{f}}^2, \nu &\sim \mathcal{IW}(\gamma, \sigma_{\mathbf{f}}^2 \mathbb{A}_\nu). \\ \sigma_{\mathbf{f}}^2 | c, d &\sim \mathcal{G}\left(\frac{c}{2}, \frac{d}{2}\right), \\ \nu | g, h &\sim \mathcal{G}(g, h)\end{aligned}$$

where $\mathcal{IG}(\cdot)$ and $\mathcal{G}(\cdot)$ represent the inverse gamma and gamma distributions, respectively, and $\mathcal{IW}(\gamma, \sigma_{\mathbf{f}}^2 \mathbb{A}_\nu)$ represents the inverse-Wishart distribution with parameters γ and $\sigma_{\mathbf{f}}^2 \mathbb{A}_\nu$ is a matrix of dimension $n \times n$ with elements $k(t, t')$, for $t, t' = 1, \dots, n$. Each term $k(t, t')$ is calculated according to the squared exponential kernel, i.e.

$$k(t, t') = \exp\left\{-\frac{(t - t')^2}{2\nu^2}\right\}. \quad (2)$$

We also assume that a, b, c, d, g and h are known hyperparameters. We set up all of them equal to 0.1 in order to get noninformative prior distributions. Now we point out some reasons that led us to consider this structure of prior distributions. The option for the Gamma distribution for σ_ϵ^2 lies of the fact of this be a natural conjugated prior. Analogously, the inverse Wishart distribution is the natural conjugated prior for $\Sigma_{\mathbf{f}}$. Since parameters $\sigma_{\mathbf{f}}^2$ and ν assume only positive values, $\sigma_{\mathbf{f}}^2, \nu \in (0, +\infty)$, then a natural choice as prior distribution is the Gamma distribution.

Using the Bayes theorem, the joint posterior distribution for $\boldsymbol{\theta} = (\mathbf{f}, \sigma_\epsilon^2, \Sigma_{\mathbf{f}}, \sigma_{\mathbf{f}}^2, \nu)$ is

$$\pi(\boldsymbol{\theta} | \mathbf{y}, \mathbf{t}) \propto L(\boldsymbol{\theta} | \mathbf{y}, \mathbf{t}) \pi(\mathbf{f} | \Sigma_{\mathbf{f}}) \pi(\sigma_\epsilon^2 | a, b) \pi(\Sigma_{\mathbf{f}} | \gamma, \sigma_{\mathbf{f}}^2, \nu) \pi(\sigma_{\mathbf{f}}^2 | c, d) \pi(\nu | g, h) \quad (3)$$

where $L(\boldsymbol{\theta} | \mathbf{y}, \mathbf{t}) \propto (\sigma_\epsilon)^{-n/2} \exp\{-\frac{1}{2}(\mathbf{y} - \mathbf{f})^T \Sigma_\epsilon^{-1}(\mathbf{y} - \mathbf{f})\}$ is the likelihood function from a n -variate normal distribution, for $\Sigma_\epsilon = \sigma_\epsilon^2 \mathbb{I}$, being \mathbb{I} is the identity matrix of dimension $n \times n$ and the overwritten T represents the matrix transpose.

The conditional posterior distributions are given by

$$\mathbf{f} | \bullet \sim \mathcal{N}_n(\Sigma_\epsilon^{-1}(\Sigma_\epsilon^{-1} + \Sigma_{\mathbf{f}}^{-1})\mathbf{y}, (\Sigma_\epsilon^{-1} + \Sigma_{\mathbf{f}}^{-1})^{-1}) \quad (4)$$

$$\sigma_\epsilon^2 | \bullet \sim \mathcal{IG}\left(\frac{a+n}{2}, \frac{b + (\mathbf{y} - \mathbf{f})^T(\mathbf{y} - \mathbf{f})}{2}\right) \quad (5)$$

$$\Sigma_{\mathbf{f}} | \bullet \sim \mathcal{IW}(\gamma + n + 1, \mathbf{f}\mathbf{f}^T + \sigma_{\mathbf{f}}^2 \mathbb{A}_\nu) \quad (6)$$

$$\sigma_{\mathbf{f}}^2 | \bullet \sim \mathcal{G}\left(\frac{c + n\gamma}{2}, \frac{d + \text{tr}(\mathbb{A}_\nu \Sigma_{\mathbf{f}}^{-1})}{2}\right), \quad (7)$$

$$\nu | \bullet \sim \pi(\nu | \bullet) \propto \pi(\Sigma_{\mathbf{f}} | \bullet) \pi(\nu | g, h), \quad (8)$$

where the symbol \bullet represents all other parameters and the observed data.

Since Equations (4)–(7) have a known form and Equation (8) does not, then in order to get estimates for the parameters of interest, we consider a Metropolis-within-Gibbs sampling algorithm (MWGS). For each iteration of the MWGS algorithm, we update the parameters $\boldsymbol{\theta}_{-\nu} = (\mathbf{f}, \sigma_\epsilon^2, \Sigma_{\mathbf{f}}, \sigma_{\mathbf{f}}^2)$ according to the Algorithm 1.

Algorithm 1 Gibbs sampling algorithm

-
- 1: Let the state of the Markov chain consist of $\theta = (\mathbf{f}, \sigma_{\epsilon}^2, \Sigma_{\mathbf{f}}, \sigma_{\mathbf{f}}^2, \nu)$;
 - 2: initialize the algorithm with a configuration $\theta^{(0)} = (\mathbf{f}^{(0)}, \sigma_{\epsilon}^{2(0)}, \Sigma_{\mathbf{f}}^{(0)}, \sigma_{\mathbf{f}}^{2(0)}, \nu^{(0)})$;
 - 3: **procedure** For the l th iteration of the algorithm, $l = 1, \dots, L$:
 - 4: generate $\mathbf{f}^{(l)}$ from conditional distribution (4), given $\sigma_{\epsilon}^{2(l-1)}, \Sigma_{\mathbf{f}}^{(l-1)}$;
 - 5: generate $\sigma_{\epsilon}^{2(l)}$ from conditional distribution (5), given $\mathbf{f}^{(l)}$;
 - 6: generate $\Sigma_{\mathbf{f}}^{(l)}$ from conditional distribution (6), given $\mathbf{f}^{(l)}$ and $\sigma_{\mathbf{f}}^{2(l-1)}$;
 - 7: generate $\sigma_{\mathbf{f}}^{2(l)}$ from conditional distribution (7), given $\Sigma_{\mathbf{f}}^{(l)}$.
-

In order to update parameter ν via Metropolis-Hastings algorithm, let ν^* to be a candidate value generated from a candidate generating-density $q[\nu^* | \nu]$. So, the value ν^* is accepted with probability $\Psi(\nu^* | \nu) = \min(1, \mathbb{H}_{\nu})$, where

$$\mathbb{H}_{\nu} = \frac{\pi(\Sigma_{\mathbf{f}} | \gamma, \sigma_{\mathbf{f}}^2, \nu^*)}{\pi(\Sigma_{\mathbf{f}} | \gamma, \sigma_{\mathbf{f}}^2, \nu)} \frac{\pi(\nu^* | g, h)}{\pi(\nu | g, g)} \frac{q[\nu | \nu^*]}{q[\nu^* | \nu]}.$$

Now, it is necessary to specify the candidate-generating density $q[\nu^* | \nu]$. Usually $q[\cdot]$ is chosen such that it is easy to sample from it. Two common choices are:

- $q[\nu^* | \nu] = \pi[\nu]$, i.e. the candidate generating-density is given by the prior distribution. In this case, \mathbb{H}_{ν} simplifies to

$$\mathbb{H}_{\nu} = \frac{\pi(\Sigma_{\mathbf{f}} | \gamma, \sigma_{\mathbf{f}}^2, \nu^*)}{\pi(\Sigma_{\mathbf{f}} | \gamma, \sigma_{\mathbf{f}}^2, \nu)}.$$

This case is denominated in the literature by Independent Metropolis-Hastings (IMH). Although the choice of the prior distributions as the candidate generating density is mathematically attractive, this may lead to many rejections of the proposed moves and a slow convergence of the algorithm. This happens, specially, for cases in which no prior information is available and prior distribution has a large variance.

- An alternative to the IMH is to explore the neighborhood of the current value of the chain in order to propose a new value.

Thus, let $q[\nu^* | \nu] = g(|\nu - \nu^*|)$, where $g(\cdot)$ is a symmetric density, i.e. the probability of generating a move from ν to ν^* depends only on the distance between them. In this case, \mathbb{H}_{ν} simplifies to

$$\mathbb{H}_{\nu} = \frac{\pi(\Sigma_{\mathbf{f}} | \gamma, \sigma_{\mathbf{f}}^2, \nu^*)}{\pi(\Sigma_{\mathbf{f}} | \gamma, \sigma_{\mathbf{f}}^2, \nu)} \frac{\pi(\nu^* | g, h)}{\pi(\nu | g, g)}. \quad (9)$$

This case is denominated in the literature as random walk Metropolis (RWM).

In the remaining of the paper, we adopt the RWM algorithm for updating ν . But, this does not prevent the proposed method to be applied with the Metropolis step being given by the IMH. In order to implement a RWM, we set up $\nu^* = \nu + \eta$ with $\eta \sim \mathcal{N}(0, \sigma_\nu^2)$. However, as discussed by Chib and Greenberg [3] the choice of σ_ν^2 has a great influence on the efficiency of the algorithm. If σ_ν^2 is small, then random perturbations will be small in magnitude and almost all will be accepted, requiring a large number of iterations to get convergence. On the other hand, if σ_ν^2 is too large, then it will cause too many rejections of the proposed moves and a considerably slowing down convergence.

According to Bedard [2], Roberts *et al.* [6], Mattingly *et al.* [4] and Saraiva and Suzuki [7], one may fix the value of σ_ν^2 testing some values in a few pilot runs and then choosing a value in which the acceptance ratio lies between 20% and 30%. Thus, following this procedure, we run 10 pilot runs of the algorithm with $L = 10,000$ iterations for $\sigma_\nu^2 \in \mathcal{G}$, where \mathcal{G} is a grid from 0.1 to 1 with increments of size 0.1. For $\sigma_\nu^2 = 0.1$ the acceptance rate was of 75%, for $\sigma_\nu^2 = 0.2$ the acceptance rate was of 25.82%, for $\sigma_\nu^2 = 0.3$ the acceptance rate was of 24.75% for $\sigma_\nu^2 = 0.4$ the acceptance rate was of 21.12% and for the other values tested, $\sigma_\nu^2 \geq 0.5$ the acceptance rate was smaller than 20%. Thus, we fix up $\sigma_\nu^2 = 0.3$, the mean of the tested values that have led to an acceptance rate between 20% and 30%.

The RWM for updating ν is implemented according to the steps of the Algorithm 2. Using Algorithms 1 and 2, we implemented the MWGS to get estimates for parameters θ . The steps for the implementation of this algorithm are given in Algorithm 3.

Algorithm 2 Random Walk Metropolis

- 1: Let the state of the Markov chain consist of $\theta = (\mathbf{f}, \sigma_\epsilon^2, \Sigma_{\mathbf{f}}, \sigma_{\mathbf{f}}^2, \nu)$.
 - 2: Initialize the algorithm with a configuration $\theta^{(0)} = (\mathbf{f}^{(0)}, \sigma_\epsilon^{2(0)}, \Sigma_{\mathbf{f}}^{(0)}, \sigma_{\mathbf{f}}^{2(0)}, \nu^{(0)})$.
 - 3: **procedure** For the l th iteration of the algorithm, $l = 1, \dots, L$
 - 4: Generate $r \sim \mathcal{N}(0, \sigma_\nu^2)$ and set $\nu^* = \nu(l-1) + r$
 - 5: Calculate $\Psi(\nu^* | \nu) = \min(1, \mathbb{H}_\nu)$, where \mathbb{H}_ν is given by Equation (9);
 - 6: Generate $u \sim \mathcal{U}(0, 1)$. If $u \leq \Psi(\nu^* | \nu)$ accept ν^* and do $\nu^{(l)} = \nu^*$. Otherwise, reject ν^* and set $\nu^{(l)} = \nu^{(l-1)}$.
-

Algorithm 3 Metropolis-within-Gibbs algorithm (MWGS)

- 1: Let the state of the Markov chain consist of $\theta = (\mathbf{f}, \sigma_\epsilon^2, \Sigma_{\mathbf{f}}, \sigma_{\mathbf{f}}^2, \nu)$.
 - 2: Initialize the algorithm with a configuration $\theta^{(0)} = (\mathbf{f}^{(0)}, \sigma_\epsilon^{2(0)}, \Sigma_{\mathbf{f}}^{(0)}, \sigma_{\mathbf{f}}^{2(0)}, \nu^{(0)})$.
 - 3: **procedure** For the l th iteration of the algorithm, $l = 1, \dots, L$
 - 4: Update $\theta_{-\nu}$ according to Algorithm 1;
 - 5: Update ν according to Algorithm 2;
-

After running the L iterations of the Algorithm 3, we discard the first B iterations as a burn-in. We also consider jumps of size J , i.e. only 1 drawn from every J was extracted from the original sequence obtaining a subsequence of size $S = \lfloor (L - B)/J \rfloor$ to make inferences. The estimates for parameters of interest are given by the average of the generated values.

For example, the estimates for \mathbf{f} and σ_ϵ^2 are $\hat{\mathbf{f}} = \bar{\mathbf{f}} = (\bar{f}_1, \dots, \bar{f}_n)$ and $\hat{\sigma}_\epsilon^2$, where

$$\bar{f}(t) = \frac{1}{S} \sum_{l=1}^L f_t^{(M(l))}, \quad \hat{\sigma}_\epsilon^2 = \frac{1}{S} \sum_{l=1}^L \sigma_\epsilon^{2(M(l))}$$

and $f(t)^{(M(l))}$ and $\sigma_\epsilon^{2(M(l))}$ are the generated values for f_t and σ_ϵ^2 in the $M(l) = (B + 1 + l \cdot J)$ th iteration of the algorithm, respectively, for $t = 1, \dots, n$. For the other parameters, the estimates are obtained in a similar way. The credibility interval (95%) for each one of the parameters is given by the quantiles 2.5% and 97.5% of the sampled values.

2.1. Predictions

Defined the estimation procedure for the parameters $\boldsymbol{\theta}$, another interest lies in predicting the value $f_{n+1} = f(x_{n+1})$ for a new input x_{n+1} . That is, the interest is in the predictive distribution

$$\pi(y_{n+1} | \mathbf{y}) = \int \pi(y_{n+1} | \mathbf{y}, \boldsymbol{\theta}) \pi(\boldsymbol{\theta} | \mathbf{y}) d\boldsymbol{\theta}, \quad (10)$$

where $\pi(\boldsymbol{\theta} | \mathbf{y})$ is the joint posterior distribution for $\boldsymbol{\theta}$, given in Equation (3). However, these integrals do not have a known mathematical solution. Hence, we present in the sequel an MCMC algorithm to get a workable approximation for the above integral.

From model (1), the marginal distribution for \mathbf{Y} is given by a n -variate normal distribution with mean vector $\mathbf{0}$ and covariance matrix $\Sigma_{\mathbf{f}} + \Sigma_\epsilon$, $\mathbf{Y} | \Sigma_{\mathbf{f}}, \Sigma_\epsilon, \sigma_\epsilon^2 \sim \mathcal{N}_n(\mathbf{0}, \Sigma_{\mathbf{f}} + \Sigma_\epsilon)$. Since $f_{n+1} \sim \mathcal{N}(m_{n+1}, \sigma_{n+1}^2)$, the joint distribution for (\mathbf{Y}, f_{n+1}) is

$$\left[\begin{array}{c} \mathbf{Y} \\ f_{n+1} \end{array} \right] | (\mathbf{f}, \mathbf{x}, x_{n+1}, \Upsilon) \sim \mathcal{N}_{n+1} \left(\left[\begin{array}{c} \mathbf{f} \\ m(x_{n+1}) \end{array} \right], \left[\begin{array}{cc} \Sigma_{\mathbf{f}} + \Sigma_\epsilon & \mathbb{C}^T \\ \mathbb{C} & \sigma_{n+1}^2 \end{array} \right] \right),$$

where $\Upsilon = (\Sigma_{\mathbf{f}}, \sigma_\epsilon^2, \sigma_{n+1}^2, \mathbb{C})$ and $\mathbb{C} = (\sigma(1, n+1), \dots, \sigma(n, n+1))$ is a row vector, $1 \times n$, composed by the covariance among f_t and f_{n+1} , $\sigma(t, n+1) = \text{cov}(f_t, f_{n+1})$, for $t = 1, \dots, n$.

By using the properties of the multivariate normal distribution, the conditional posterior distribution for f_{n+1} is given by

$$f_{n+1} | (\mathbf{y}, \mathbf{f}, \mathbf{x}, x_{n+1}, \Upsilon) \sim \mathcal{N}(m(t_{n+1}) + \mathbb{C}(\Sigma_{\mathbf{f}} + \Sigma_\epsilon)^{-1}(\mathbf{y} - \mathbf{f}), \sigma_{n+1}^2 - \mathbb{C}(\Sigma_{\mathbf{f}} + \Sigma_\epsilon)^{-1}\mathbb{C}^T). \quad (11)$$

At this point, as we know the value of the 7-day moving average on the n th day, we fix up $m(t_{n+1}) = y_n$. In addition, we set up each $\sigma(t, n+1) = k(t, n+1)$ in order to get these values according to Equation (2). Then, a sample from conditional posterior distribution in (11) can be generated according to the implementation of the Algorithm 4.

After running the algorithm for the same L iterations, burn in B and jump J of the Algorithm 3, an approximation for the integral in (10) is given by

$$\tilde{\pi}(f_{n+1} | \mathbf{y}) = \frac{1}{S} \sum_{l=1}^L f_{n+1}^{(M(l))},$$

where $M(l)$ is the $(B + 1 + l \cdot J)$ th iteration of the algorithm.

Algorithm 4 Prediction

- 1: Let the state of the Markov chain consist of $\theta = (\mathbf{f}, \sigma_\epsilon^2, \Sigma_{\mathbf{f}}, \sigma_{\mathbf{f}}^2, \nu)$ and f_{n+1} .
- 2: Initialize the algorithm with a configuration $\theta^{(0)} = (\mathbf{f}^{(0)}, \sigma_\epsilon^{2(0)}, \Sigma_{\mathbf{f}}^{(0)}, \sigma_{\mathbf{f}}^{2(0)}, \nu^{(0)})$ and $f_{n+1}^{(0)}$.
- 3: **procedure** For the l th iteration of the algorithm, $l = 1, \dots, L$
- 4: Update θ according to Algorithm 1;
- 5: Update ν according to Algorithm 2;
- 6: Conditional on $\theta^{(l)} = \theta^{(l)} = (\mathbf{f}^{(l)}, \sigma_\epsilon^{2(l)}, \Sigma_{\mathbf{f}}^{(l)}, \sigma_{\mathbf{f}}^{2(l)}, \nu^{(l)})$, generate $f_{n+1}^{(l)}$ from conditional posterior distribution in (11).

The predictions for the next $n+j$ days, for $j = 2, \dots, r$, is obtained in a similar way by generating a sample from f_{n+j} according to Algorithm 4, where f_{n+j} is given by Equation (11), setting up $m(t_{n+j}) = \hat{f}_{n+j-1}$ and $\mathbb{C} = (k(1, n+j), \dots, k(n, n+j))$, for $j = 2, \dots, r$.

In many cases, the interest lies in predicting the value $f_{n+1} = f(x_{n+1})$ for a new input x_{n+1} conditional on the last \mathbf{y}_s recorded values and not on all past values \mathbf{y} . For example, one can have interest in the value y_{n+1} given the recorded 7-day moving average in the last $s = 30$ days, $\mathbf{y}_s = (y_{n-29}, \dots, y_n)$. This prediction is also made using Algorithm 4, just changing \mathbf{y} by \mathbf{y}_s and adapting the parameters θ for the dimension of \mathbf{y}_s . The predictions for the next $n+j$ days are done as described in the paragraph above, clearly with some differences. The advantage of this kind of prediction is the computation time that is smaller than the prediction procedure conditional on \mathbf{y} .

3. Application

In this section, we apply the proposed method to a real dataset. The dataset refers to the 7-day moving average for the number of deaths recorded in the state of São Paulo, Brazil. As an illustration of this dataset, we present in Figure 1 the number of deaths recorded in the period from 17 March 2020 (first case) to 16 July 2021 and the 7-day moving average recorded in the period from 23 March 2020 ($t = 0$) to 27 April 2021 ($n = 480$).

In order to estimate parameters of interest and make the predictions, we apply Algorithms 3 and 4 with $L = 55,000$ iterations, $B = 5000$ iterations and $J = 10$. Thus, we got a sample of size 5000 to make inferences. Using these values, we ran our first analysis on the 100th day after the recording of the first 7-day moving average. Figure 2 shows the observed value (symbols \bullet) and the credibility band of 95% (blue area) determined by the proposed method. This figure also shows the prediction band of 95% (black area) and the recorded values for the next ten days. The mean square error (MSE) of the predicted values in relation to the recorded values was 0.0056. As one can note, the prediction band contains all recorded values. Both results show a very satisfactory performance of the proposed method.

Figure 3 shows the graphic of the residuals. Figure 3(a) shows the quantile-quantile plot of the residuals in relation to the normal distribution. As one can note, the normality assumption is satisfied. Figure 3(b) shows the graphic of the predicted values *versus* the standardized residuals. As one can note, the points are uniformly distributed indicating

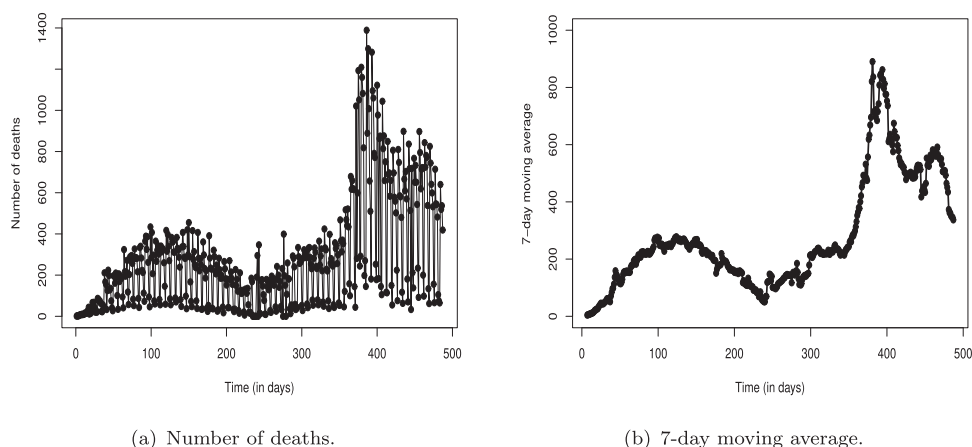


Figure 1. Number of deaths by day and 7-day moving average values. (a) Number of deaths. (b) 7-day moving average.

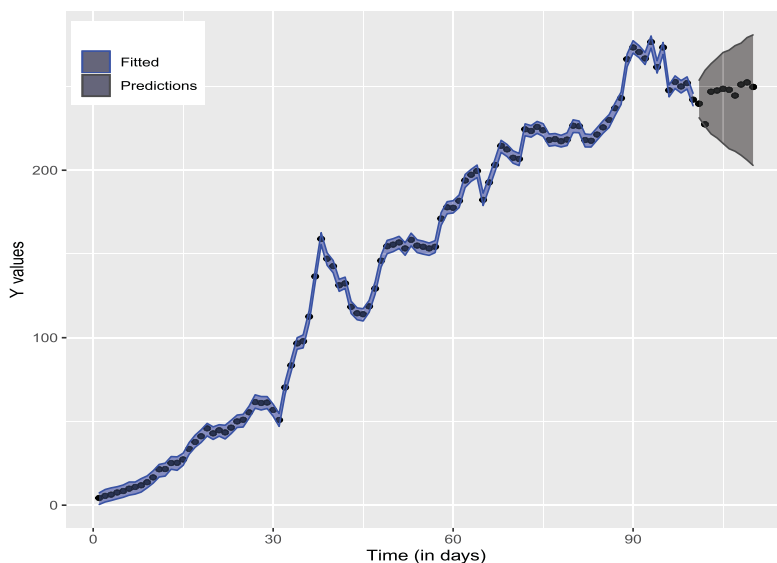


Figure 2. Confidence and prediction bands (95%).

that there is not evidence to reject the assumption of homogeneity of the variance. This graphic also shows that the residuals are not correlated. These both results also show the very satisfactory performance of the proposed method.

The estimated value for the σ_ϵ^2 is $\hat{\sigma}_\epsilon^2 = 2.4427$, with a credibility interval (95%) given by (2.0154, 2.9632). Figure 4 shows the graphics of the ergodic mean (ErM) and the estimated autocorrelation function (ACF) from the sampled values for σ_ϵ^2 . As one can note, there is no reason to doubt the convergence of the sampled values, since the ErM values present satisfactory stabilization and there is no significant ACF.

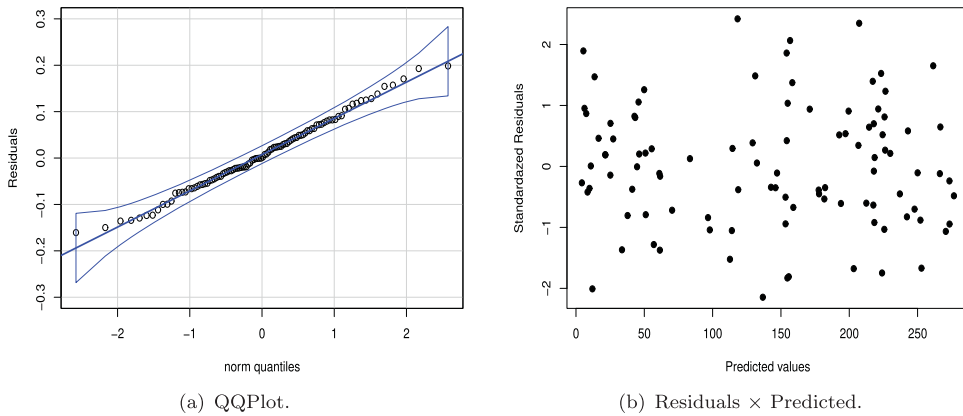


Figure 3. Residuals plot. (a) Q-QPlot, (b) residuals \times predicted.

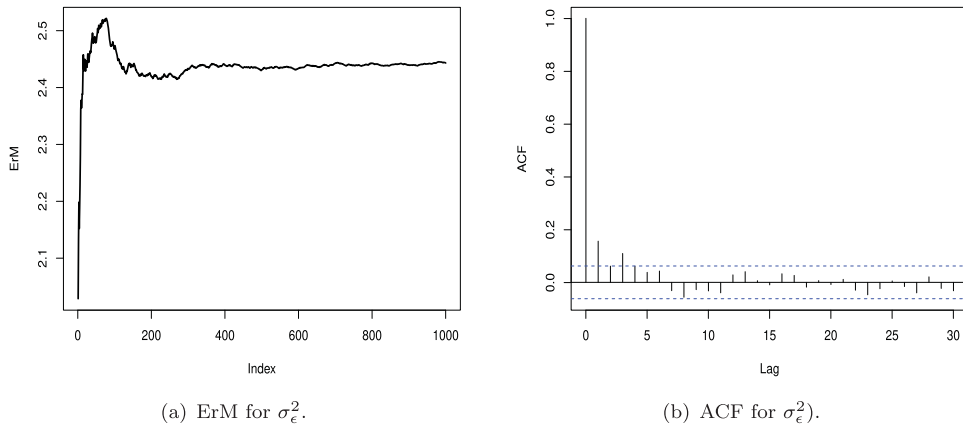


Figure 4. ErM and ACF from the sampled values for σ_ϵ^2 . (a) ErM for σ_ϵ^2 . (b) ACF for σ_ϵ^2 .

We also verify the convergence of the sampled values for \mathbf{f} . The results are similar to the presented by the sampled values for σ_ϵ^2 . As an illustration, we show in Figure 5, the ErM and the ACF for the sample valued for f_{22} . The f_{22} was chosen at random among the $f_t \in \mathbf{f}$, for $t = 1, \dots, n$. The estimate for f_{22} is $\hat{f}_{22} = 43.7603$; and the credibility interval (95%) is (40.2492, 49.2031). The recorded value was 43.43. That is an absolute percentage error of 0.7604%. Analogously to the sampled value for σ_ϵ^2 , there is no reason to doubt the convergence of the sampled values for f_{22} .

The estimate for parameter σ_f^2 is $\hat{\sigma}_f^2 = 3.8234$ with credibility interval (2.0459, 3.9189). The estimate for parameter ν is $\hat{\nu} = 0.2637$ with credibility interval (0.0153, 0.4982) and acceptance rate of 26.78%. The convergence checking for both parameters is similar to the presented for σ_ϵ^2 and f_{22} , i.e. the ergodic mean presents satisfactory stabilization and there is no significant ACF.

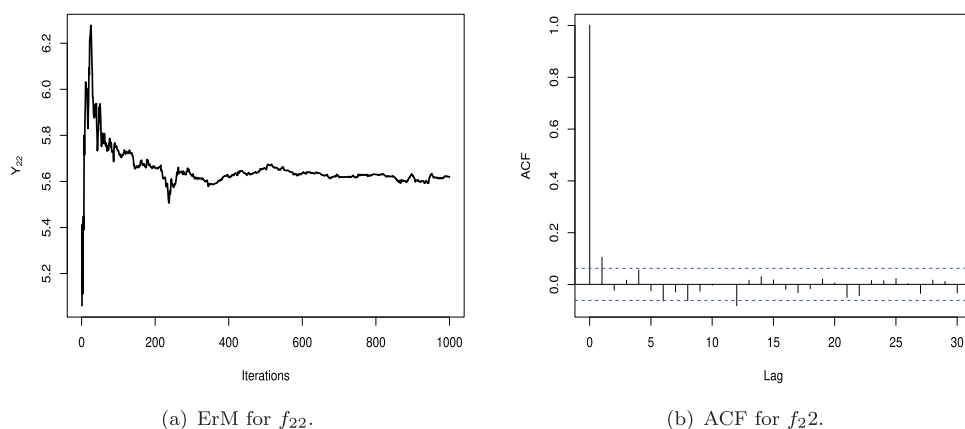


Figure 5. ErM and ACF from the sampled values for f_{22} . (a) ErM for f_{22} . (b) ACF for f_{22} .

3.1. Predictions based in the last 30 days

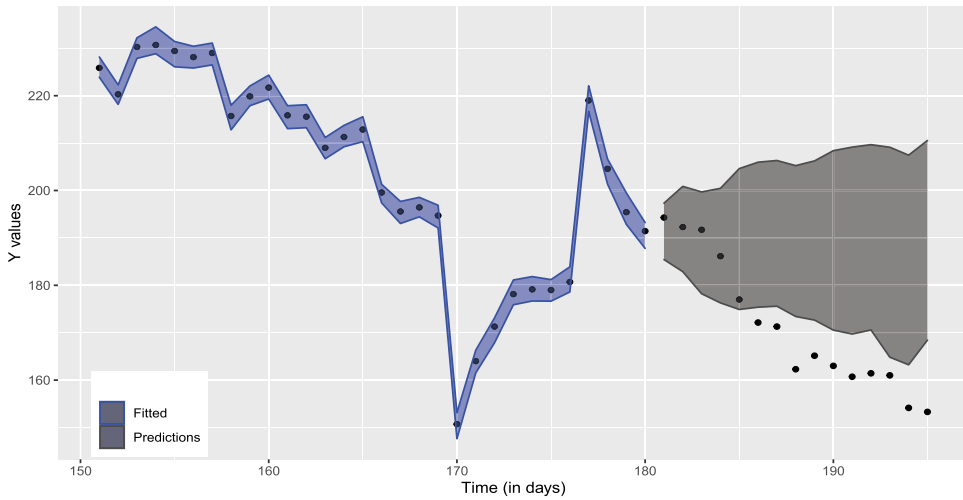
Consider now the interest in predicting the evolution of the 7-day moving average values for the next 15 days after the n th day conditional on the recorded values in the last thirty days. That is, the interest is to predict the values f_{n+j} for $j = 1, \dots, 15$ conditional on $\mathbf{y}_{30} = (y_{n-30+1}, \dots, y_n)$. In the following, we present the results from three analyses ran at 180th, 280th and 465th days.

In order to obtain these predictions, we apply Algorithm 4 just changing \mathbf{y} by \mathbf{y}_{30} and adapting the dimension of the parameters $\boldsymbol{\theta}$. We run this algorithm for the same values for L , burn-in and jump J used in Section 3. Figure 6 shows the results of the analysis carried out on the 180th day. The MSE of the fitted model is 0.0123. As one can note, the five 7-day average values recorded after the 180th day are inside the prediction band, but the values recorded after the 185th day are outside the band. This is a good news because it indicates that the reduction in the values of the 7-day moving average was greater than expected. In this period of 15 days, the moving average reduced from 191.43 (180th day) to 153.29 (195th day). A reduction of 19.42%.

Figure 7 shows the results of the analysis ran on the 280th day. The MSE of the fitted model is 0.0471. As one can note, only the two last recorded values are outside the prediction band. In this period of 15 days, the 7-day moving average value increased 44.05%, going from 119.14 (280th day) to 213 (295th day).

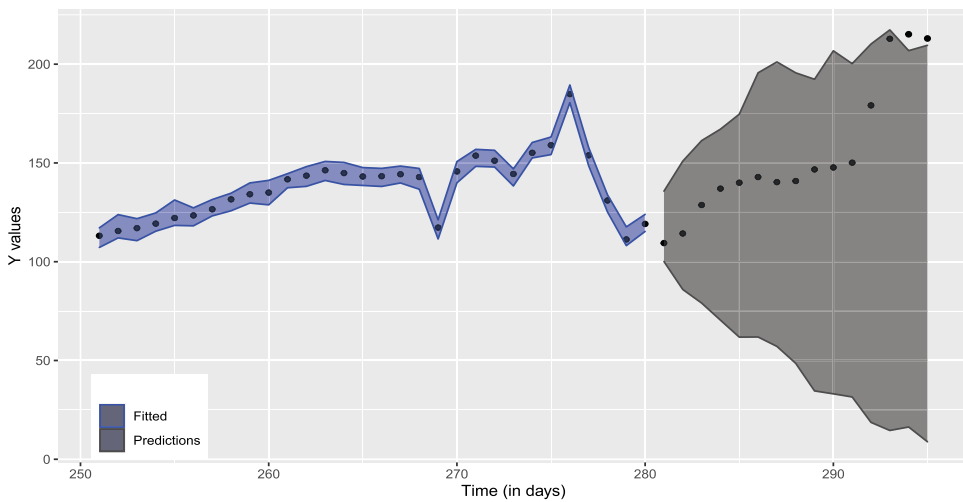
Figure 8 shows the results of the analysis ran on the 465th day. The MSE of the fitted model is 0.0437. Note that, the credibility band determined by the proposed method based on the recorded values in the last thirty days indicates a stabilization of the moving average, but the recorded values were all below this region. This is a piece of very good news because it shows that a greater reduction than expected happened. The reduction was 49.14%, going from 550.86 (365th day) to 340.14 (480th day).

Figure 9 shows the recorded values in the last 30 days and the credibility band determined by the proposed method for the next 15 days. As one can note, this graphic shows there is no expected a great increase in the values of the 7-day moving average in the next 15 days.



(a) 2-nd analysis.

Figure 6. Recorded values and confidence band determined by the proposed method, 180th day. (a) 2nd analysis.

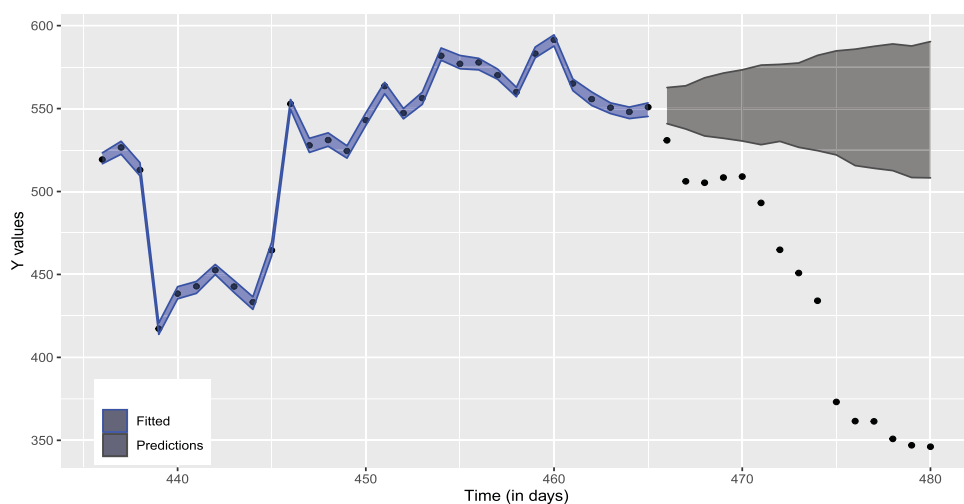


(a) 2-nd analysis.

Figure 7. Recorded values and confidence band determined by the proposed method. (a) 2nd analysis.

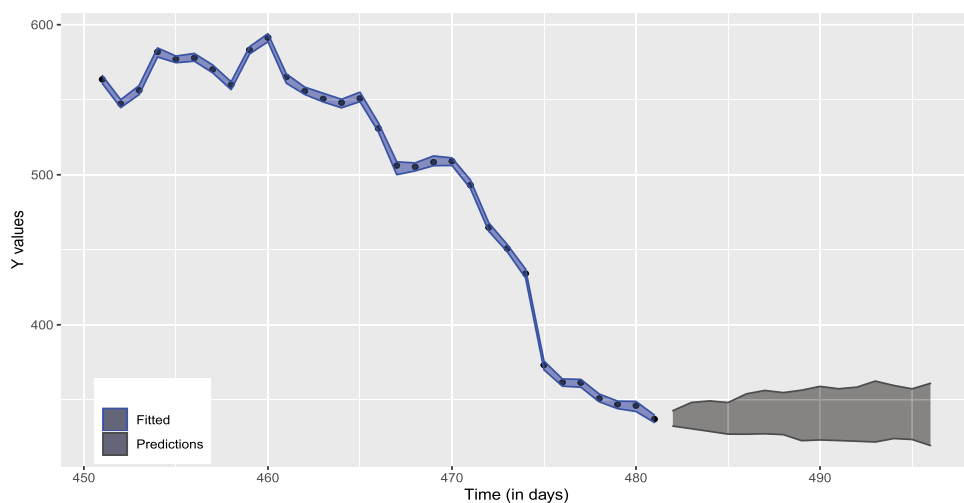
4. Final remarks

This article presented a hierarchical Bayesian methodology for modeling the evolution of the 7-day moving average for the number of deaths due to COVID-19. We opt to model the 7-day moving average in opposite to the cumulative number of deaths, as is usual, due to the moving average smooth the possible discrepancies that may have in the statistical reports of the number of deaths divulged.



(a) 2-nd analysis.

Figure 8. Recorded values and confidence band determined by the proposed method, 280th day. (a) 2nd analysis.



(a) 2-nd analysis.

Figure 9. Recorded values and confidence band determined by the proposed method, 465th day. (a) 2nd analysis.

Contrary to usual approaches, which are based on the adoption of a set of parametric models followed by a comparison using some model selection criteria, such as, AIC and BIC, the proposed approach assumes that the nonlinear mathematical function is an unknown random quantity that needs to be estimated from the observed data. Then, we adopt a Bayesian nonlinear regression approach with a Gaussian process.

The adoption of the Gaussian process prior means that we are defining a probability distribution over functions and the inference is taken directly in the space of the functions. This approach is considered nonparametric since we are using a prior distribution on the space of functions that corresponds to infinite-dimensional parameter space. Since making inferences about the infinite number of parameters is impractical, the assumption of the Gaussian process has the advantage of that conditional on a dataset a finite number of parameters (function values) can be explicitly represented by a multivariate normal distribution, making the estimation procedure simple to be computationally implemented.

In addition, since the Gaussian process is heavily influenced by the choice of the covariance function Σ_f , we assume one more hierarchical level by putting a prior distribution over Σ_f and on its hyperparameters. The inference of the parameters of interest is carried out using a Metropolis-within-Gibbs algorithm. We also present an MCMC algorithm to make predictions.

The proposed method is illustrated in a case study, in which, we model the 7-day moving average values recorded in the state of São Paulo, Brazil, in the period from 03/23/2020 to 04/27/2021. In this period, we ran the estimation procedure four times in order to verify the performance of the method. The results have shown a very good performance of the method. The MSE values were all smaller than 0.05, and there is no reason to doubt the convergence of the sampled values by the MCMC algorithm.

From a statistical data analysis point of view, the proposed method is very interesting because it does not need the assumption of a parametric model, and the inference is made by an MCMC algorithm that can be easily implemented in free software, such that, R software. The computational codes were implemented in the R language and can be obtained by e-mail to the authors.

Disclosure statement

No potential conflict of interest was reported by the author(s).

ORCID

E. F. Saraiva  <http://orcid.org/0000-0002-4698-4004>

References

- [1] E. Aviv-Sharon and A. Aharoni, *Generalized logistic growth modeling of the COVID-19 pandemic in Asia*, *Infect. Dis. Model* 5 (2020), pp. 502–509.
- [2] M. Bedard, *Weak convergence of metropolis algorithms for non-i.i.d. target distributions*, *Ann. Appl. Probab.* 17 (2007), pp. 1222–1244.
- [3] S. Chib and E. Greenberg, *Understanding the metropolis-Hastings algorithm*, *Am. Stat.* 49 (1995), pp. 327–335.
- [4] J.C. Mattingly, N.S. Pillai, and A.M. Stuart, *Diffusion limits of the random walk metropolis algorithm in high dimensions*, *Ann. Appl. Probab.* 22 (2011), pp. 881–930.
- [5] S.S. Musa, S. Zhao, and M.H. Wang, *Estimation of exponential growth rate and basic reproduction number of the coronavirus disease 2019 (COVID-19) in Africa*, *Infect. Dis. Poverty* 9 (2020), Article number: 96.
- [6] G. Roberts, A. Gelman, and W. Gilks, *Weak convergence and optimal scaling of random walk metropolis algorithms*, *Ann. Appl. Probab.* 7 (1997), pp. 110–120.

- [7] E.F. Saraiva and A.K. Suzuki, *Bayesian computational methods for estimation of two-parameters Weibull distribution in presence of right-censored data*, Chil. J. Stat. 8 (2017), pp. 25–43.
- [8] G.L. Vasconcelos, A.M.S. Macêdo, R. Ospina, F.A.G. Almeida, G.C. Duarte-Filho, A.A. Brum, and I.C.L. Souza, *Modelling fatality curves of COVID-19 and the effectiveness of intervention strategies*, PeerJ 8 (2020), Article ID e9421.
- [9] J. Wang, *An intuitive tutorial to gaussian processes regression*, preprint (2021). Available at <https://arxiv.org/pdf/2009.10862.pdf>
- [10] K. Wu, D. Darcet, and Q. Wang, *Generalized logistic growth modeling of the COVID-19 outbreak: Comparing the dynamics in the 29 provinces in China and in the rest of the world*, Nonlinear Dyn. 101 (2020), pp. 1561–1581.