

Article

A Bayesian Approach for Modeling and Forecasting Solar Photovoltaic Power Generation

Mariana Villela Flesch ¹, Carlos Alberto de Bragança Pereira ²  and Erlandson Ferreira Saraiva ^{3,*} 

¹ Faculty of Engineering, Architecture and Urbanism and Geography, Federal University of Mato Grosso do Sul, Campo Grande 79070-900, MS, Brazil; mariflesch@hotmail.com

² Institute of Mathematics and Statistics, University of São Paulo, São Paulo 05508-090, SP, Brazil; cadebp@gmail.com

³ Institute of Mathematics, Federal University of Mato Grosso do Sul, Campo Grande 79070-900, MS, Brazil

* Correspondence: erlandson.saraiva@ufms.br; Tel.: +55-67-3345-7511

Abstract: In this paper, we propose a Bayesian approach to estimate the curve of a function $f(\cdot)$ that models the solar power generated at k moments per day for n days and to forecast the curve for the $(n + 1)$ th day by using the history of recorded values. We assume that $f(\cdot)$ is an unknown function and adopt a Bayesian model with a Gaussian-process prior on the vector of values $\mathbf{f}(\mathbf{t}) = (f(1), \dots, f(k))$. An advantage of this approach is that we may estimate the curves of $f(\cdot)$ and $f_{n+1}(\cdot)$ as “smooth functions” obtained by interpolating between the points generated from a k -variate normal distribution with appropriate mean vector and covariance matrix. Since the joint posterior distribution for the parameters of interest does not have a known mathematical form, we describe how to implement a Gibbs sampling algorithm to obtain estimates for the parameters. The good performance of the proposed approach is illustrated using two simulation studies and an application to a real dataset. As performance measures, we calculate the absolute percentage error, the mean absolute percentage error (MAPE), and the root-mean-square error (RMSE). In all simulated cases and in the application to real-world data, the MAPE and RMSE values were all near 0, indicating the very good performance of the proposed approach.

Keywords: photovoltaic solar power forecasting; statistical modeling; Bayesian inference; Gaussian process; MCMC; Gibbs sampling algorithm



Citation: Flesch, M.V.; de Bragança Pereira, C.A.; Saraiva, E.F. A Bayesian Approach for Modeling and Forecasting Solar Photovoltaic Power Generation. *Entropy* **2024**, *26*, 824. <https://doi.org/10.3390/e26100824>

Academic Editors: Udo Von Toussaint and Refik Soyer

Received: 15 July 2024

Revised: 23 August 2024

Accepted: 24 September 2024

Published: 27 September 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent years, there has been a significant increase in solar energy generation, from both photovoltaic plants and residences with photovoltaic panels installed on their roofs. This has been driven by societal and governmental interest in clean and renewable energy, with an important aspect of “clean” being lower CO₂ emission compared to fossil fuels. Because of this, more photovoltaic plants are being connected to local electric-supply systems every day.

However, according to [1], this causes instability in the grid, which is one of the greatest challenges to the energy industry. Electrical operators need to know how much energy will be added to the system in order to balance it with consumption and ensure that the system is capable of meeting consumer demand. For [2], the ability to predict photovoltaic solar power output is very important for secure grid operation, scheduling, and the effectiveness of power-grid management.

In this context, statistical models emerge as important tools for modeling and predicting photovoltaic power generation. Some statistical approaches used for modeling the solar photovoltaic power generation include linear regression models [3–6], autoregressive models [7–9], and artificial-neural-network models [10–12], that is, parametric models are still commonly employed due to their ease of use.

However, parametric models have at least three limitations: (i) the analysis is limited to the function (or functions) previously chosen by the analyst; (ii) the complexity and/or flexibility of the functions considered is limited by the number of parameters in the functions; and (iii) there may exist several functions that can fit the recorded values equally well. A common solution adopted in much statistical analysis is to fit a set of candidate models and then choose the best model using some model-selection criterion, such as AIC [13] or BIC [14]. Even in those cases, issues (ii) and (iii) still remain.

In this paper, in order to give models more flexibility instead of restricting them to a function $f(\cdot)$ chosen previously, we adopt a semi-parametric Bayesian approach, in which the curve of the function $f(\cdot)$ is estimated from the observed data. For this, we assume that cumulative solar photovoltaic power generation, measured at k time instants per day, is modeled by an additive model composed of a nonlinear growth function $f(\cdot)$ evaluated at k time instants plus a random error ε . However, instead of setting up $f(\cdot)$ as a known mathematical function, we assume that $f(\cdot)$ is an unknown function whose vector of values $\mathbf{f}(\mathbf{t}) = (f(1), \dots, f(k))$ is treated as a set of parameters that must be estimated based on recorded data for $\mathbf{t} = (1, \dots, k)$. To jointly estimate $\mathbf{f}(\mathbf{t})$ and the other parameters in the proposed model, we adopt a Bayesian approach with a Gaussian-process prior over $\mathbf{f}(\mathbf{t})$. An advantage of this approach is that we may estimate the curve of function $f(\cdot)$ using “smooth functions” that are obtained by linking points generated from a k -variate normal distribution with an appropriate mean vector and covariance matrix. Additionally, we present a forecasting procedure for the value of the curve on the $(n + 1)$ th day, conditioned on the values recorded over the first n days.

Since the joint posterior distribution for the parameters and the predictive distribution of the proposed model do not have known mathematical forms, we describe how to implement a Gibbs sampling algorithm [15–17] to generate random values from these two distributions. This algorithm generates values from the distributions of interest in an indirect way, using the conditional posterior distributions, as long as those are known, which is the case for the model proposed here.

To illustrate the performance of the proposed model, we include two simulation studies. In the first one, we examine the performance of the proposed model in the estimation of the curve of $f(\cdot)$. As performance measures, we calculate the absolute percentage error (APE) and the mean absolute percentage error (MAPE). In all simulated cases, the proposed approach presents MAPE values near 0, indicating that the estimated values are close to the real values. In the second simulation study, we evaluate the performance of predictions made with the proposed approach. Analogously to the first simulation study, the proposed approach presents satisfactory performance, as indicated by MAPE values near zero. In addition to APE and MAPE values, we also evaluate the performance of predictions in terms of the root-mean-square error (RMSE). The RMSE values were all near zero, indicating a very good performance of the proposed approach. We also apply the proposed approach to a real dataset. Like in the simulation studies, the results obtained in this application were very accurate, with MAPE and RMSE values near zero.

The main novelty that we bring in this paper is in the way that we model the generation of solar photovoltaic power over time. First, we consider the photovoltaic power generated on each day as having its own behavior, and the behavior is taken to be proportional to the average behavior of the measurements over the last n days. Second, the function $f(\cdot)$ that models the generation of solar power as a function of time is considered unknown, but with its curve estimated from the observed data. We highlight the following four advantages of this approach: (i) the proposed hierarchical Bayesian model is very flexible and adapts to the number of values recorded; (ii) the inference procedure is based on a Gibbs sampling algorithm, which can be easily implemented in statistical software such as R; (iii) the predicted growth curve for day $(n + 1)$ is obtained directly using only the history of the first n measurements; and (iv) there is no need to fit a set of models and afterwards compare them using some model-selection criterion.

The remainder of the paper is organized as follows. In Section 2, we present the dataset that has motivated us to develop the proposed modeling, the hierarchical Bayesian model, and the estimation procedure for the parameters of interest. In Section 2, we also present the results of the first simulation study. In Section 3, we present the prediction procedure and the second simulation study. In Section 4, we apply the proposed approach to a real dataset. Finally, in Section 5, we conclude with some final remarks.

2. Dataset and Statistical Modeling

A critical component of any statistical analysis is the dataset used to make inferences on the parameters of interest. The dataset used in this paper was obtained from a photovoltaic plant installed on the campus of the Brazilian Federal University of Mato Grosso do Sul. This dataset is freely available on the website <https://github.com/lscad-facom-ufms/Solar2> (accessed on 3 June 2024), and more details on the experiment can be found in [18]. The dataset used to make inferences on the parameters of the proposed model contains measurements of solar photovoltaic power generation taken at $k = 74$ time instants each day over a period of $N = 19$ days. In other words, the dataset is a spreadsheet composed of 3 columns and $k \times n = 74 \times 19 = 1406$ lines. Figure 1 shows a clipping from the data spreadsheet, showing that the first column contains the day (1–19), the second column the time instant (TI, 1–74), and the third column the observed values for photovoltaic solar power (PSP) generated.

1	Day	TI	PSP
2	1	1	63.15
3	1	2	140.31
4	1	3	273.65
5	1	4	435.1

Figure 1. Clipping of the data spreadsheet.

Let W_{it} be the solar power recorded at the it th time instant of the i th day and $\mathbf{W}_i = (W_{i1}, \dots, W_{ik})$ the vector of values recorded on the i th day, for $i = 1, \dots, N$ and $t = 1, \dots, k$. Figure 2 shows the values recorded over the first two days of the experiment. As one can note, the recorded data on these two days present great variability, which makes the modeling process difficult. For the other days, the recorded values present similar behavior. Due to this, we opt to model the accumulated values.

Consider $W_{it}^{ac} = \sum_{t'=1}^t W_{it'}$ to be the accumulated values of the photovoltaic power generated through the t th time instant of the i th day and $\mathbf{W}_i^{ac} = (W_{i1}^{ac}, \dots, W_{ik}^{ac})$ to be the vector of accumulated values, for $i = 1, \dots, n$ and $t = 1, \dots, k$. Figure 3a shows the accumulated values recorded over the first four days of the experiment. As one can note, the accumulated values present more stable and predictable behavior. However, many values in the vectors \mathbf{W}_i^{ac} are on the scale of 100,000, which could cause computational problems in the inference process. To avoid this problem, we opt to model the *logarithm* of the accumulated values, denoted by $Y_{it} = \log(W_{it}^{ac})$, for $i = 1, \dots, N$ and $t = 1, \dots, k$. Additionally, let $\mathbf{Y}_i = (Y_{i1}, \dots, Y_{ik})$ be the vector of values recorded on the i th day, for $i = 1, \dots, N$.

Figure 3b, shows the graph of \mathbf{Y} values recorded over the first four days of the experiment. The symbols • in black connected by black lines represent the average values $\bar{\mathbf{y}} = (\bar{y}_1, \dots, \bar{y}_k)$. For the other days, the recorded \mathbf{Y} values present similar behavior. From this point forward, and without loss of generality, consider the modeling of $\mathbf{y} = (y_1, \dots, y_n)$, that is, the data recorded over the first n days of the experiment, for $n < N$, where the primary interest is in the prediction of the values that will be generated on day $(n + 1)$. That is, \mathbf{y} is an $n \times k$ matrix in which row i contains the y values generated on day i , for $i = 1, \dots, n$.

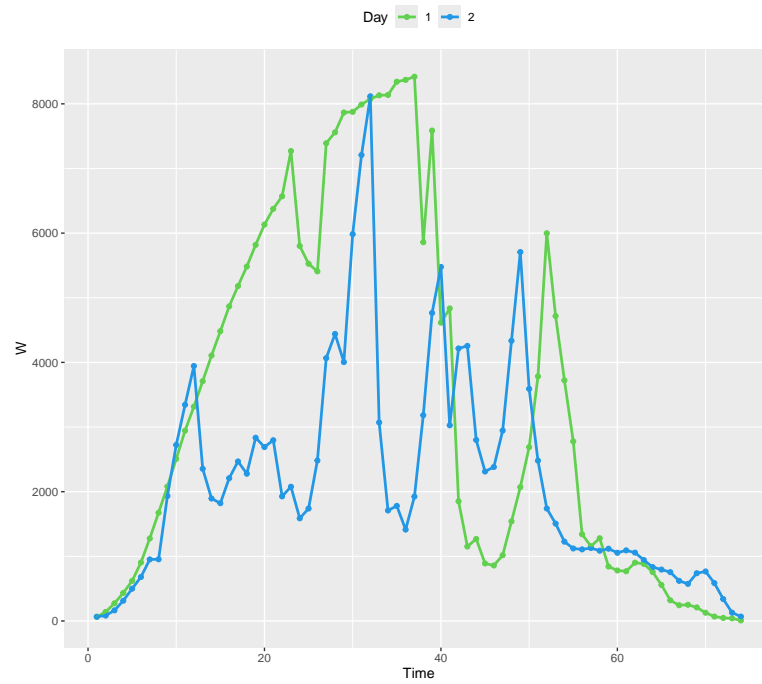


Figure 2. Solar power generated over time for days 1 and 2.

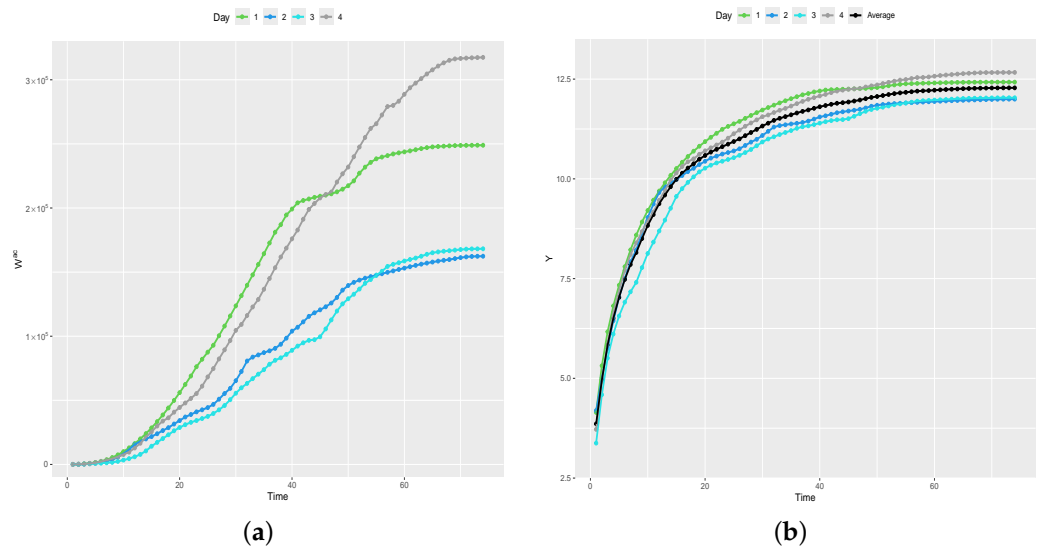


Figure 3. Solar power generated over time. (a) $W_1, W_2, W_3,$ and W_4 . (b) $W_1^{ac}, W_2^{ac}, W_3^{ac},$ and W_4^{ac} .

2.1. Hierarchical Bayesian Model

Based on Figure 3b, consider the average curve (black line) to be a nonlinear growth function $f(t)$, where $\mathbf{f}(\mathbf{t}) = (f(1), \dots, f(k))$ is a vector composed of the values of $f(\cdot)$ at k instants of time, $\mathbf{t} = (1, \dots, k)$. Assume that the growth function for the i th day is proportional to $f(\cdot)$, i.e., $f_i(\cdot) = C_i f(\cdot)$, for $C_i > 0$ and $i = 1, \dots, n$ with $n \leq N$. In other words, we are assuming that there is a growth function $f(\cdot)$ whose curve represents the average curve from n curves, with the curve on the i th day proportional to the average curve.

Consider recorded values on the i th day to be generated according to the following additive model:

$$\mathbf{Y}_i = (Y_{i1}, \dots, Y_{ik}) \sim C_i \cdot \mathbf{f}(\mathbf{t}) + \boldsymbol{\varepsilon}_i, \tag{1}$$

with $C_i > 0$, where $\boldsymbol{\varepsilon}_i = (\varepsilon_{i1}, \dots, \varepsilon_{ik})$ is a vector of random errors, for $i = 1, \dots, n$. Assume that $\boldsymbol{\varepsilon}_i$ is generated according to a k -dimensional multivariate normal distribution with

mean vector $\mathbf{0} = (0, \dots, 0)$ and covariance matrix Σ (dimension $k \times k$) composed of the elements $\sigma_\varepsilon(t, t') = \text{Cov}(\varepsilon_{it}, \varepsilon_{it'})$, for $t, t' = 1, \dots, k$ and $i = 1, \dots, n$.

To complete Model (1), we could fix $f(\cdot)$ as a known mathematical function, such as the logistic or Gompertz growth functions, among others. However, three problems with this parametric approach are (i) the analysis is limited to the function (or functions) previously chosen by the analyst; (ii) the complexity and/or flexibility of the considered functions is limited by the number of parameters in the functions; and (iii) there may exist several functions that can fit the recorded values equally well. A common solution to problem (i) is to fit a set of candidate models and then choose the best model using some model selection criterion, such as AIC [13] or BIC [14]. However, issues (ii) and (iii) still remain.

In order to avoid restricting our model to a specific parametric function, from this point onward, we assume that $f(\cdot)$ is an unknown function and that the values $\mathbf{f}(\mathbf{t}) = (f(1), \dots, f(k))$ are model parameters that need to be estimated from observed data. Under this scenario and with the model given by (1), the parameters of interest are $\theta = (\mathbf{f}(\mathbf{t}), \Sigma, \mathbf{C})$, where $\mathbf{f}(\mathbf{t}) = (f(1), \dots, f(k))$, Σ is the covariance matrix of the vector of random errors, and $\mathbf{C} = (C_1, \dots, C_n)$.

To estimate θ , we take a hierarchical Bayesian approach with a Gaussian-process prior on $\mathbf{f}(\mathbf{t})$, denoted by $\mathbf{f}(\mathbf{t})|\mathbf{m}, \Sigma_f \sim \mathcal{GP}(\mathbf{m}, \Sigma_f)$. This means that we are considering $f(\cdot)$ as an unknown function, but with the vector of values $\mathbf{f}(\mathbf{t}) = (f(1), \dots, f(k))$ generated by a k -variate normal distribution with mean vector \mathbf{m} and covariance matrix Σ_f composed of the elements $\sigma_f(t, t') = \text{Cov}(f(t), f(t'))$, for $t, t' = 1, \dots, k$. For Σ , we assume a conjugated inverse-Wishart prior distribution with parameter (δ, \mathbb{V}) , and, for C_i , we assume a prior distribution given by a truncated normal distribution (with the left-of-zero part removed) with parameters μ_c and σ_c^2 , for $i = 1, \dots, n$. The proposed model is then represented hierarchically:

$$\begin{aligned} \mathbf{Y}_i = (Y_{i1}, \dots, Y_{ik}) | \mathbf{f}(\mathbf{t}), \Sigma, \mathbf{C} &\sim \mathcal{N}_k(C_i \mathbf{f}(\mathbf{t}), \Sigma) \\ \mathbf{f}(\mathbf{t}) | \mathbf{m}, \Sigma_f &\sim \mathcal{GP}(\mathbf{m}, \Sigma_f) \\ \Sigma | \delta, \mathbb{V} &\sim \mathcal{IW}(\delta, \mathbb{V}) \\ C_i | \mu_c, \sigma_c^2 &\sim \mathcal{N}_{\text{trunc}}(0; \mu_c, \sigma_c^2), \end{aligned} \tag{2}$$

where $\mathcal{N}_k(\cdot)$, $\mathcal{GP}(\cdot)$, $\mathcal{IW}(\cdot)$, and $\mathcal{N}_{\text{trunc}}(0; \cdot)$ represent, respectively, a k -variate Gaussian distribution, the Gaussian process, the inverse-Wishart distribution, and the truncated normal distribution with values only on the right half-line; additionally, \mathbf{m} , Σ_f , δ , \mathbb{V} , μ_c , and σ_c^2 are known hyperparameters, for $i = 1, \dots, n$.

We complete the modeling by setting the following:

- (i) $\mathbf{m} = \mathbf{0}$ in order to represent our lack of informative prior knowledge about the expected value of $\mathbf{f}(\mathbf{t})$;
- (ii) $\Sigma_f = \lambda \mathbb{W}$, with $\lambda > 0$ and \mathbb{W} a matrix of dimension $k \times k$ composed of elements $\kappa(t, t')$, calculated according to the squared exponential kernel, i.e.,

$$\kappa(t, t') = \eta^2 \exp\left\{-\frac{(t - t')^2}{2\nu^2}\right\}, \tag{3}$$

with $\eta, \nu > 0$. The parameter η controls how far the generated values are from the average. Small values for η characterize functions that are close to their average value, whereas larger values allow greater variation. The parameter ν controls the smoothness of the function obtained by connecting the points. Small values of ν mean that function values may change quickly, and large values characterize functions that change more slowly (are smoother). We set $\lambda = 100$, $\eta = 1$, and $\nu = 1$ in order to obtain a weakly informative prior distribution;

- (iii) We finalize the model by setting $\delta = k$, $\mathbb{V} = 0.01 \cdot \mathbb{I}_k$, where \mathbb{I} is the identity matrix of dimension $k \times k$, and $\mu_c = \sigma_c^2 = 1$.

Applying Bayes’s theorem, the joint posterior distribution for $\theta = (\mathbf{f}(\mathbf{t}), \Sigma, \mathbf{C})$ is given by

$$\pi(\theta|\mathbf{y}, \mathbf{t}) \propto \mathcal{L}(\theta|\mathbf{y}, \mathbf{t})\pi(\mathbf{f}(\mathbf{t})|\mathbf{m}, \Sigma_{\mathbf{f}})\pi(\Sigma|\delta, \mathbb{V})\pi(\mathbf{C}|\mu_{\mathbf{c}}, \sigma_{\mathbf{c}}^2), \tag{4}$$

where $\mathcal{L}(\theta|\mathbf{y}, \mathbf{t})$ is the likelihood function of a k -variate normal distribution with parameters $\mathbf{f}(\mathbf{t})$ and Σ , and $\pi(\cdot)$ represents the probability density functions of the prior distributions, for $\mathbf{y} = (\mathbf{y}_1, \dots, \mathbf{y}_n)$ and $\mathbf{t} = (1, \dots, k)$.

However, the joint posterior distribution does not have a known mathematical form that allows us to generate random values from this distribution directly. Due to this, we need to use an algorithm that generates the random numbers of this joint distribution in an indirect way. In this paper, we opt to generate random values from $\pi(\theta|\mathbf{y}, \mathbf{t})$ using the Gibbs sampling algorithm [15,17] due to its simplicity of implementation and efficiency. This algorithm generates values from the joint posterior distribution indirectly, using the conditional posterior distributions, as long as they are known.

For the proposed hierarchical Bayesian model, the conditional posterior distributions are known and given by

$$\mathbf{f}(\mathbf{t})|\mathbf{y}, \mathbf{t}, \bullet \sim \mathcal{GP}\left(\Sigma^{-1}\left(\sum_{i=1}^n C_i \Sigma^{-1} + \Sigma_{\mathbf{f}}^{-1}\right)^{-1} \sum_{i=1}^n C_i \mathbf{y}_i, \left(\sum_{i=1}^n C_i \Sigma^{-1} + \Sigma_{\mathbf{f}}^{-1}\right)^{-1}\right) \tag{5}$$

$$\Sigma|\mathbf{y}, \mathbf{t}, \bullet \sim \mathcal{IW}\left(\delta + k + n, \mathbb{V} + \sum_{i=1}^n (\mathbf{y}_i - C_i \mathbf{f}(\mathbf{t}))^\top (\mathbf{y}_i - C_i \mathbf{f}(\mathbf{t}))\right) \tag{6}$$

$$C_i|\mathbf{y}, \mathbf{t}, \bullet \sim \mathcal{N}_{\text{trunc}}\left(0, \frac{\mathbf{f}(\mathbf{t})^\top \Sigma^{-1} \mathbf{y}_i + 1}{\mathbf{f}(\mathbf{t})^\top \Sigma^{-1} \mathbf{f}(\mathbf{t}) + 1}, \frac{\sigma_{\mathbf{c}}^2}{\mathbf{f}(\mathbf{t})^\top \Sigma^{-1} \mathbf{f}(\mathbf{t}) + \mu}\right), \tag{7}$$

where the symbol \bullet represents all other parameters.

Using the conditional posterior distributions, we implement a Gibbs sampling algorithm according to the steps described in Algorithm 1.

Algorithm 1 Gibbs sampling algorithm.

- 1: Let the state of the Markov chain consist of $\theta = (\mathbf{f}(\mathbf{t}), \Sigma, \mathbf{C})$.
 - 2: Initialize the algorithm with a configuration $\theta^{(0)} = (\mathbf{f}(\mathbf{t})^{(0)}, \Sigma^{(0)}, \mathbf{C}^{(0)})$.
 - 3: **procedure** For the l th iteration of the algorithm, $l = 1, \dots, L$:
 - 4: generate $\mathbf{f}(\mathbf{t})^{(l)}$ from conditional distribution (5), given $\Sigma^{(l-1)}$ and $\mathbf{C}^{(l-1)}$;
 - 5: generate $\Sigma^{(l)}$ from conditional distribution (6), given $\mathbf{f}(\mathbf{t})^{(l)}$ and $\mathbf{C}^{(l-1)}$;
 - 6: generate $C_i^{(l)}$ from conditional distribution (7), given $\mathbf{f}(\mathbf{t})^{(l)}$ and $\Sigma^{(l)}$, for $i = 1, \dots, n$.
-

After running L iterations of the Gibbs sampling algorithm, 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 in order to obtain a sub-sequence of size $S = [(L - B) / J]$ to make inferences. The estimates for the parameters of interest are given by the average of the generated values, i.e.,

$$\hat{\mathbf{f}}(\mathbf{t}) = \frac{1}{S} \sum_{l=1}^S \mathbf{f}(\mathbf{t})^{(M(l))}, \quad \hat{\Sigma} = \frac{1}{S} \sum_{l=1}^S \Sigma^{(M(l))} \quad \text{and} \quad \hat{C}_i = \frac{1}{S} \sum_{l=1}^S C_i^{(M(l))}$$

where $\mathbf{f}(\mathbf{t})^{(M(l))}$, $\Sigma^{(M(l))}$, and $\mathbf{C}^{(M(l))}$ are the generated values for parameters $\mathbf{f}(\mathbf{t})$, Σ , and \mathbf{C} , respectively, in the $M(l) = (B + 1 + (l - 1) \cdot J)$ th iteration of the algorithm, for $l = 1, \dots, S$. The 95% credibility interval for each of the parameters is given by the 2.5% and 97.5%

percentiles of the sampled values. The estimated curve of $f(\cdot)$ is obtained by plotting the points $(t, \hat{f}(t))$ connected by lines, for $t = 1, \dots, k$.

2.2. First Simulation Study

To illustrate the performance of the proposed approach, we develop a simulation study. To generate the dataset, we consider $f(\cdot)$ as the log-Gompertz function of parameters α_1, α_2 , and α_3 with the parametrization $f(t) = \log(\alpha_1) - \exp\{\alpha_2 - \alpha_3 t\}$, for $t > 0$. We set $\alpha_1 = 12$, $\alpha_2 = 2$, and $\alpha_3 = 0.1$. For the covariance matrix Σ , we calculate each term according to the squared-exponential kernel given in Equation (3) with $\eta^2 = 0.01$ and $\nu^2 = 10$.

The procedure to generate the artificial dataset is given by the following four steps:

- (i) Fix the number of days n and the number of time instants per day k ;
- (ii) With $\mathbf{t} = (1, \dots, k)$, calculate $\mathbf{f}(\mathbf{t}) = (f(1), \dots, f(k))$, where $f(t)$ is given by the log-Gompertz function described above;
- (iii) Fix the values $\mathbf{C} = (C_1, \dots, C_n)$, with $C_i > 0$ and $i = 1, \dots, n$;
- (iv) Generate $\mathbf{Y}_i = (Y_{i1}, \dots, Y_{ik}) \sim \mathcal{N}_k(\mathbf{f}(\mathbf{t})^\top, \Sigma)$, for $i = 1, \dots, n$

To simplify the visualization of the results, our first simulation study considers $n = 4$ and $k = 50$. We set $\mathbf{C} = (C_1, C_2, C_3, C_4) = (0.8, 0.9, 1.1, 1.2)$. Figure 4a shows the curve of $f(t)$, which we call the “average curve”, and the curve for the i th day is given by $C_i f(t)$, for $i = 1, 2, 3, 4$. Figure 4b shows the curves and actual y values generated for each day (coloured \bullet symbols), in which black \bullet symbols are the average values \bar{y} .

With the dataset generated, we apply the proposed Gibbs sampling algorithm with $L=55,000$ iterations, $B = 5000$, and $J = 10$. In this way, we obtain a posterior sample of size $S = 5000$ to make inferences. To verify how far the estimates $\hat{f}(t)$ are from the real values $f(t)$, we calculate the absolute percentage error:

$$APE(d_t) = \frac{|f(t) - \hat{f}(t)|}{f(t)} \cdot 100,$$

for $t = 1, \dots, k$.

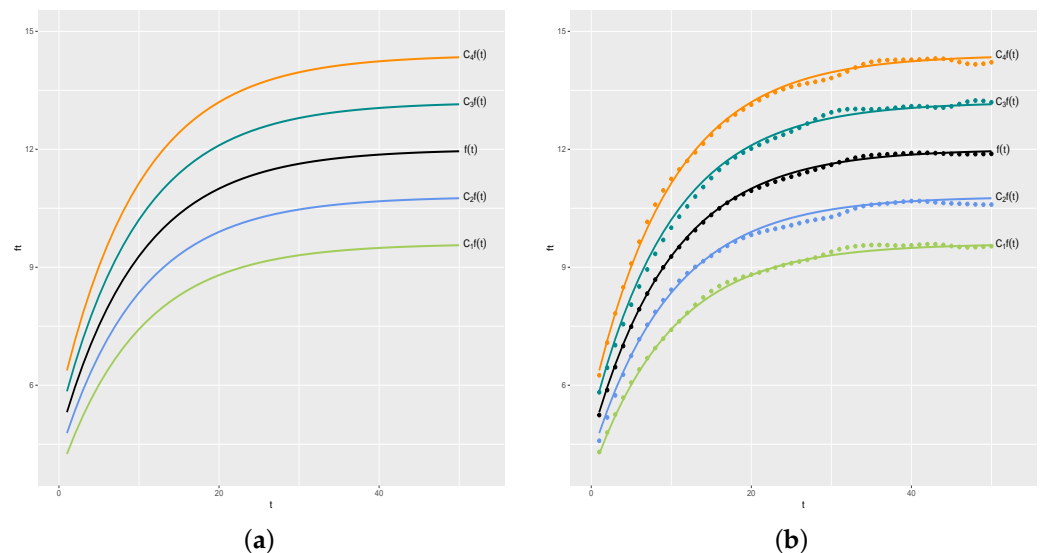


Figure 4. Real curves and generated values. (a) Real curves. (b) Generated values.

Figure 5a shows $f(t)$ (black line) and the estimated curve by the proposed method (red line) with a credibility band of 95% (red region). Figure 5b shows $APE(\mathbf{d}) = (APE(d_1), \dots, APE(d_k))$. $APE(\mathbf{d})$ values ranged from a minimum of 0.0145 to a maximum of 1.4481, with a mean absolute percentage error (MAPE) of 0.4568. The small values of APE indicate that the estimated values $\hat{f}(t)$ are very close to the real values of $f(t)$, for $t = 1, \dots, k$.

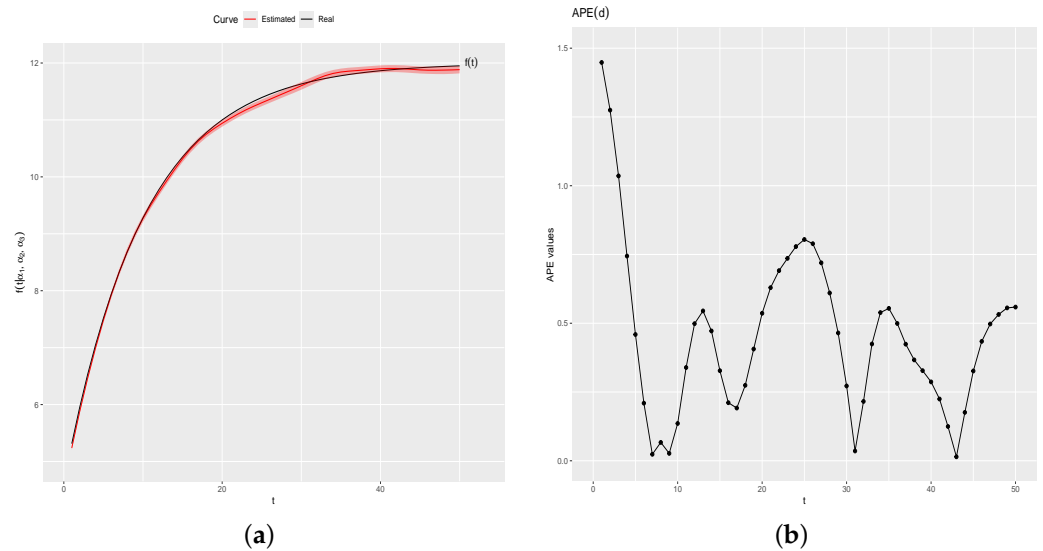


Figure 5. Real and estimated curve and APE(d) values. (a) Real and estimated $f(t)$. (b) APE(d).

Similarly, we obtained the estimated curve for the i th day by plotting the pairs (t, \hat{y}_{it}) connected by lines, for $\hat{y}_{it} = \hat{C}_i \hat{f}(t)$, $i = 1, \dots, n$, and $t = 1, \dots, k$. For this case, we calculate two different kinds of error: the APE for comparison between the real value $C_i f(t)$ and the estimated value \hat{y}_{it} ; and the APE for comparison between the generated values y_{it} and the estimated \hat{y}_{it} values. They are calculated as follows:

$$APE(d_{it}) = \frac{|C_i f(t) - \hat{y}_{it}|}{C_i f(t)} \cdot 100 \quad \text{and} \quad APE(e_{it}) = \frac{|y_{it} - \hat{y}_{it}|}{y_{it}} \cdot 100$$

for $i = 1, \dots, n$ and $t = 1, \dots, k$.

Table 1 shows the estimates and 95% credibility intervals for parameters C_i , $i = 1, 2, 3, 4$. As one can note, the estimated values are very close to the real values, and the real values are inside the credibility intervals.

Table 1. Real value, estimated value, and 95% credibility interval for C_i , $i = 1, 2, 3, 4$.

Parameter	Real Value	Estimated Value	Credibility Interval of 95%
C_1	0.8	0.8060	(0.7997, 0.8080)
C_2	0.9	0.8955	(0.8913, 0.9003)
C_3	1.1	1.0998	(1.0939, 1.1057)
C_4	1.2	1.2006	(1.1949, 1.2071)

Figure 6a shows $C_i f(t)$ (black line) and the estimated curves by the proposed method (red lines), where the symbols \bullet are the generated values for each day. Figure 6b shows the values of APE(d_i), and Table 2 shows the summary measures of APE(d_i) values, for $i = 1, \dots, n$. As one can note, the estimated curves for each one of the four days is satisfactorily close to the real curves, as indicated by APE(d_i) values near zero, $i = 1, \dots, n$.

Table 2. Summary measures of $APE(d_i)$ values, for $i = 1, 2, 3, 4$.

Measure	Min	1 ^o Q	Median	Mean	3 ^o Q	Max
$APE(d_1)$	0	0.2100	0.4450	0.5212	0.7675	1.3100
$APE(d_2)$	0	0.4825	0.8350	0.7972	1.0575	1.9400
$APE(d_3)$	0.0100	0.2400	0.4650	0.4650	0.5775	1.4600
$APE(d_4)$	0.0100	0.2375	0.4250	0.4364	1.5750	1.4000

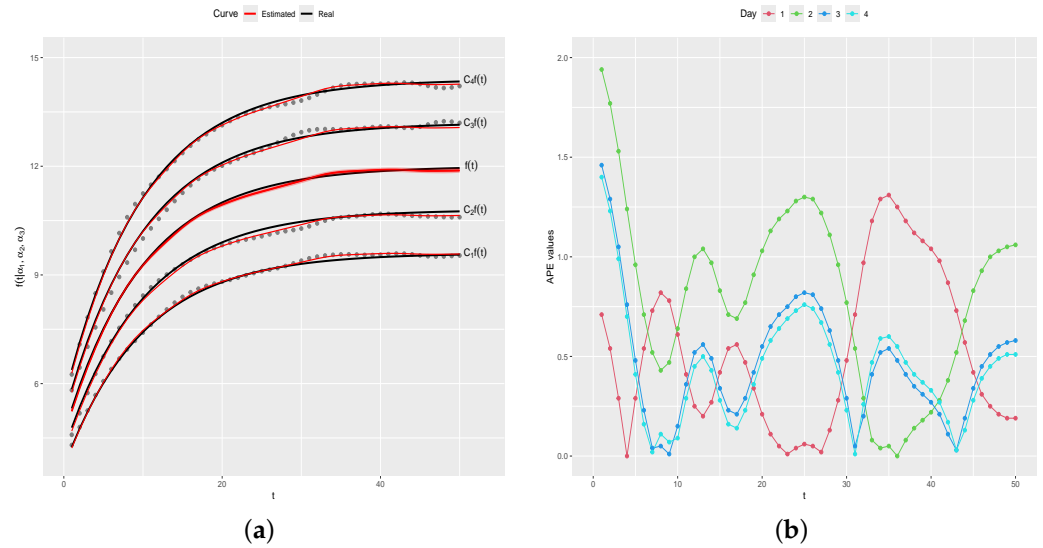


Figure 6. Real and estimated curves and $APE(d_i)$ values, for $i = 1, 2, 3, 4$. (a) Real and estimated curves. (b) $APE(d_i)$ values.

Figure 7 shows the graphic of the values of $APE(e_i) = (APE(e_1), \dots, APE(e_k))$, and Table 3 shows the summary measures of $APE(e_i)$ values, for $i = 1, \dots, n$. $APE(e_i)$ values are near zero, indicating that the estimated values \hat{y}_{it} are satisfactorily close to the generated values y_{it} , for $i = 1, \dots, n$ and $t = 1, \dots, k$.

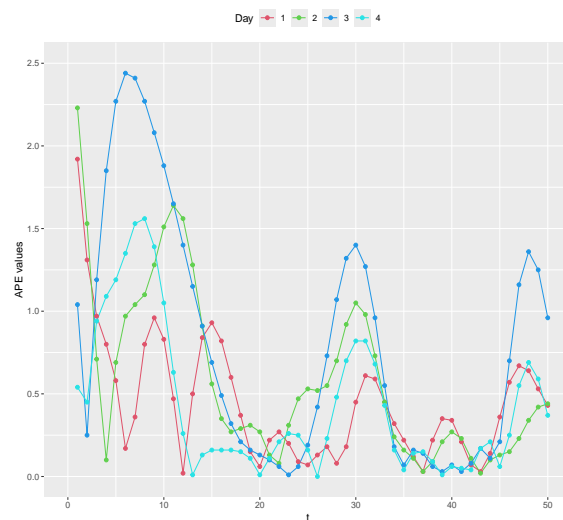


Figure 7. $APE(e_i)$ values, for $i = 1, 2, 3, 4$.

Table 3. Summary measures of $APE(e_i)$ values, for $i = 1, 2, 3, 4$.

Measure	Min	1 ^o Q	Median	Mean	3 ^o Q	Max
$APE(e_1)$	0.0200	0.1775	0.3600	0.4446	0.6075	1.9200
$APE(e_2)$	0.0200	0.2150	0.4300	0.5860	0.9175	2.2300
$APE(e_3)$	0.0100	0.1450	0.6200	0.7934	1.2650	2.4400
$APE(e_4)$	0	0.1325	0.2400	0.4330	0.3675	1.5600

Since the inferences were made from a posterior sample obtained from an MCMC algorithm, it is important to check the convergence of the sampled values. As is usual, we verify the convergence of the sampled values empirically, using the ergodic mean (ErM) of the sampled values. Figure 8 shows the graphic of the ErM for the sampled values for $f(1)$ and $f(30)$. As one can note, there is no reason to doubt the convergence of the sampled values since the ErM values present satisfactory stabilization. The graphs of ErM for the sampled values for other parameters are similar.

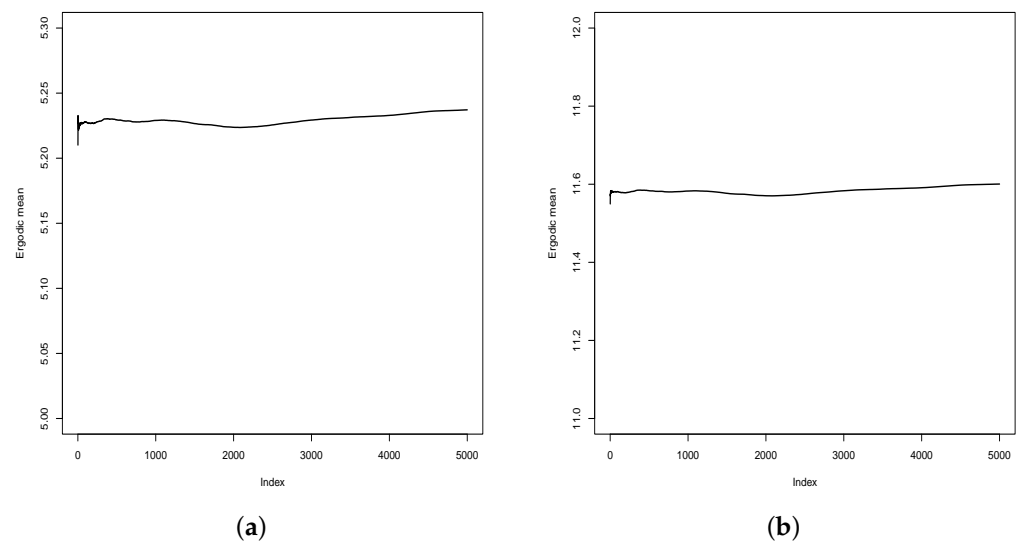


Figure 8. Ergodic mean (ErM) for sampled values for $f(1)$ and $f(30)$. (a) $f(1)$. (b) $f(30)$.

3. Predictions

In addition to obtaining the estimates $\hat{\theta} = (\hat{f}(t), \hat{\Sigma}, \hat{C})$ for the parameters $\theta = (f(t), \Sigma, C)$, the modeling used in the previous section can also be used to predict the values that will be recorded on the $(n + 1)$ th day, i.e., $Y_{n+1} = (Y_{1(n+1)}, \dots, Y_{k(n+1)})$. This can be performed by using the predictive distribution, given by

$$\pi(Y_{n+1}|y, t) = \int \pi(Y_{n+1}|y, t, \theta)\pi(\theta|y, t)d\theta, \tag{8}$$

where $\pi(\theta|y, t)$ is the joint posterior distribution for θ , given in Equation (4). However, this integral does not have a known analytic solution. Due to this, we present in the following an MCMC algorithm for obtaining an approximation for this integral.

From Model (1), the marginal distribution for Y_i is given by a k -variate normal distribution with mean vector $\mathbf{0} = (0, \dots, 0)^T$ and covariance matrix $C_i^2 \Sigma_t + \Sigma$, for $i = 1, \dots, n$. Thus,

$$Y = (Y_1, \dots, Y_n) | \Sigma_y \sim \mathcal{N}_{nk}(\mathbf{0}, \Sigma_y),$$

where $\mathcal{N}_{nk}(\cdot)$ represents an nk -variate normal distribution with mean vector $\mathbf{0}$ and a covariance matrix of dimension $nk \times nk$, given by

$$\Sigma_{\mathbf{y}} = \begin{bmatrix} C_1^2 \Sigma_{\mathbf{f}} + \Sigma & \Sigma_{12} & \Sigma_{13} & \dots & \Sigma_{1n} \\ \Sigma_{21} & C_2^2 \Sigma_{\mathbf{f}} + \Sigma & \Sigma_{23} & \dots & \Sigma_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \Sigma_{n1} & \Sigma_{n2} & \Sigma_{n3} & \dots & C_n^2 \Sigma_{\mathbf{f}} + \Sigma \end{bmatrix},$$

where $\Sigma_{ii'} = C_i C_{i'} \Sigma_{\mathbf{f}}$ are the covariance matrices (of dimension $k \times k$) among the measurements \mathbf{Y}_i and $\mathbf{Y}_{i'}$, for $i, i' = 1, \dots, n$ and $i \neq i'$. Similarly, $\mathbf{Y}_{n+1} \sim \mathcal{N}_k(\mathbf{0}, C_{n+1}^2 \Sigma_{\mathbf{f}} + \Sigma)$. Therefore, the joint distribution for $(\mathbf{Y}, \mathbf{Y}_{n+1})$ is

$$\begin{bmatrix} \mathbf{Y} \\ \mathbf{Y}_{n+1} \end{bmatrix} | \boldsymbol{\theta}, \mathbf{y}, \mathbf{t} \sim \mathcal{N}_{(n+1)k} \left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \Sigma_{\mathbf{y}} & \mathbb{B}^T \\ \mathbb{B} & C_{n+1}^2 \Sigma_{\mathbf{f}} + \Sigma \end{bmatrix} \right),$$

where $\mathbb{B} = [\Sigma_{(n+1)1} \ \Sigma_{(n+1)2} \ \dots \ \Sigma_{(n+1)n}]$ is a block of matrices, in which $\Sigma_{(n+1)i}$ are the covariance matrices among the values of \mathbf{Y}_{n+1} and \mathbf{Y}_i given by $\Sigma_{(n+1)i} = C_{n+1} C_i \Sigma_{\mathbf{f}}$, for $i = 1, \dots, n$.

Using the properties of the multivariate normal distribution, the conditional posterior distribution for \mathbf{Y}_{n+1} is given by

$$\mathbf{Y}_{n+1} | \mathbf{y}, \mathbf{t}, \boldsymbol{\theta} \sim \mathcal{N}_k \left(\mathbb{B} \Sigma_{\mathbf{y}}^{-1} \mathbf{y}, \left(C_{n+1}^2 \Sigma_{\mathbf{f}} + \Sigma \right) - \mathbb{B}^T \left(C_{n+1}^2 \Sigma_{\mathbf{f}} + \Sigma \right)^{-1} \mathbb{B} \right); \tag{9}$$

with C_{n+1} generated from

$$C_{n+1} | \mathbf{C} \sim \mathcal{Ntrunc}(0, \bar{C}, S_C^2), \tag{10}$$

where \bar{C} and S_C^2 are, respectively, the average and the variance of $\mathbf{C} = (C_1, \dots, C_n)$.

Thus, a sample from the conditional posterior distribution of $(\mathbf{Y}_{n+1}, \boldsymbol{\theta})$ can be generated according to the steps in Algorithm 2.

Algorithm 2 Prediction.

- 1: Let the state of the Markov chain consist of $\boldsymbol{\theta} = (\mathbf{f}(\mathbf{t}), \Sigma, \mathbf{C})$ and \mathbf{Y}_{n+1} .
 - 2: Initialize the algorithm with a configuration $\boldsymbol{\theta}^{(0)} = (\mathbf{f}(\mathbf{t})^{(0)}, \Sigma^{(0)}, \mathbf{C}^{(0)})$ and $C_{n+1}^{(0)}$.
 - 3: **procedure** For the l th iteration of the algorithm, $l = 1, \dots, L$:
 - 4: Update $\boldsymbol{\theta}$ according to Algorithm 1;
 - 5: Generate $Y_{n+1}^{(l)}$ from conditional posterior distribution in (9) given $C_{n+1}^{(l-1)}$;
 - 6: Generate C_{n+1} from probability distribution in (10).
-

After running the algorithm for the same L iterations, burn-in B , and jump J as we used for algorithm (1), an approximation for the integral in (8) is given by

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

where $M(l)$ is the $(B + 1 + l \cdot J)$ th iteration of the algorithm, for $l = 1, \dots, S$.

Second Simulation Study

To illustrate the performance of the prediction procedure, we present a second simulation study. Like in the first simulation study, we fix $n = 4$, $k = 50$, and $f(t)$ as the log-Gompertz function of parameters $\alpha_1 = 12$, $\alpha_2 = 2$, and $\alpha_3 = 0.1$. Here, the main objective is to predict the curve for the $(n + 1) = 5$ th day.

To obtain the curves of the first four days with the curve of $f(t)$ being the average curve, we adopt the following procedure:

- (i) Let $C_i < n$ and define $\mathbf{p} = (p_1, p_2, p_3, p_4)$ with $p_i = \frac{C_i}{n}$, for $i = 1, 2, 3, 4$;
- (ii) Generate $\mathbf{p} \sim \text{Dirichlet}(\boldsymbol{\alpha})$, where $\text{Dirichlet}(\boldsymbol{\alpha})$ is the Dirichlet distribution with parameter $\boldsymbol{\alpha}$. We set up $\boldsymbol{\alpha} = (50, 50, 50, 50)$;
- (iii) Obtain $C_i = n \cdot p_i$ and generate $\mathbf{Y}_i = (Y_{i1}, \dots, Y_{ik}) \sim \mathcal{N}_k(C_i \mathbf{f}(t), \Sigma)$, for $i = 1, \dots, n$, where Σ is obtained as described in simulation study 1.

The generated values for $\mathbf{C} = (C_1, C_2, C_3, C_4)$ were $(0.9187, 0.8767, 1.0126, 1.1919)$, respectively. Figure 9a shows the real curves for days 1 to 4, denoted by $C_i f(t)$ for $i = 1, 2, 3, 4$, and Figure 9b shows the same graphs as Figure 9a with the generated values for each day as correspondingly coloured \bullet symbols, the black \bullet symbols being the average values.

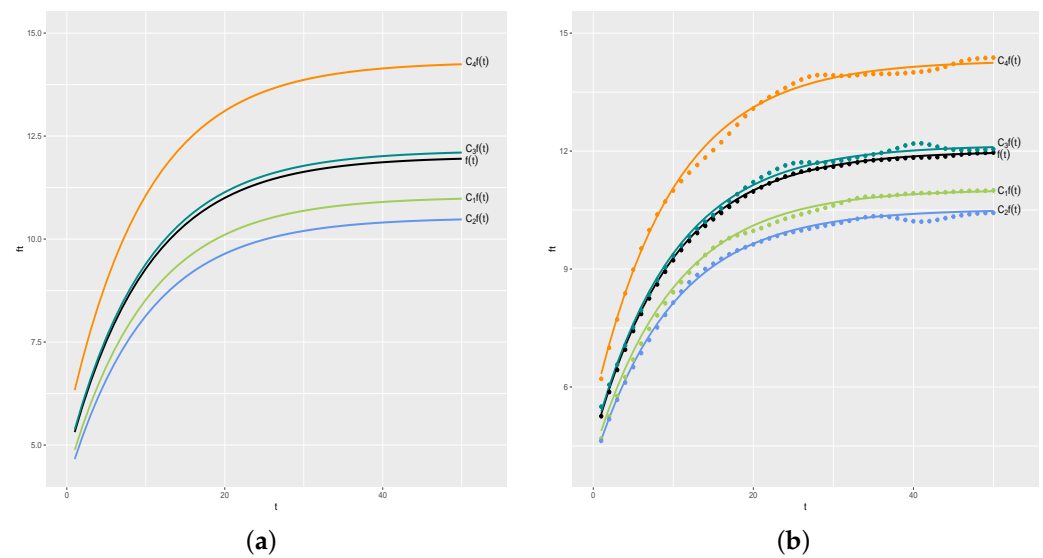


Figure 9. Real curves and generated values. (a) Real curves. (b) Generated values.

With $(\mathbf{Y}_1, \dots, \mathbf{Y}_n)$ values generated, we generate the data for the $(n + 1)$ th day as follows:

- (i) Generate $C_{n+1} \sim \mathcal{N}_{\text{trunc}}(0, \bar{C}, S_c^2)$, where $\bar{C} = \frac{1}{n} \sum_{i=1}^n C_i$ and $S_c^2 = \frac{1}{n-1} \sum_{i=1}^n (C_i - \bar{C})^2$. From the generated $\mathbf{C} = (0.9187, 0.8767, 1.0126, 1.1919)$ values, the generated value for C_{n+1} was 0.9823;
- (ii) Generate $\mathbf{Y}_{n+1} = (Y_{(n+1)1}, \dots, Y_{(n+1)k}) \sim \mathcal{N}_k(C_{n+1} \mathbf{f}(t), \Sigma)$.

We then use the generated values for $(\mathbf{Y}_1, \dots, \mathbf{Y}_n)$ in the prediction procedure (Algorithm 2) and obtain the estimates for $\hat{\mathbf{Y}}_{n+1}$. For this, we apply the prediction procedure for the same $L=55,000$ iterations with a burn-in $B = 5000$ and jump of size $J = 10$ as we used in Algorithm 1. The estimated curve for the $(n + 1)$ th day is obtained by plotting the pairs $(t, \hat{y}_{(n+1)t})$ connected by lines, where, $\hat{y}_{(n+1)t}$ is the predicted value for $Y_{(n+1)t}$, for $t = 1, \dots, k$.

Figure 10a shows $f(t)$ and the estimated curve by the proposed method, and Figure 10b shows the graph of $\text{APE}(\mathbf{d})$ values. As in the first simulation study, the results show a very satisfactory performance of the proposed method, with the estimated curve very close to the real curve of $f(t)$, as indicated by $\text{APE}(\mathbf{d})$ values all being less than 1.

Figure 11a shows $C_i f(t)$ (black line) and the estimated curves by the proposed method (red lines), with the symbols \bullet representing the generated values for each day. Figure 11b shows $\text{APE}(\mathbf{d}_i)$, for $i = 1, \dots, n$. As one can note, the estimated curves for each one of the four days are satisfactorily near the real curves, as indicated by $\text{APE}(\mathbf{d}_i)$ values near zero, for $i = 1, 2, 3, 4$. We also verify the convergence of the sampled APE values. Analogously to

results presented in the first simulation study, there is no reason to doubt the convergence of the sampled values since the ErM values present satisfactory stabilization.

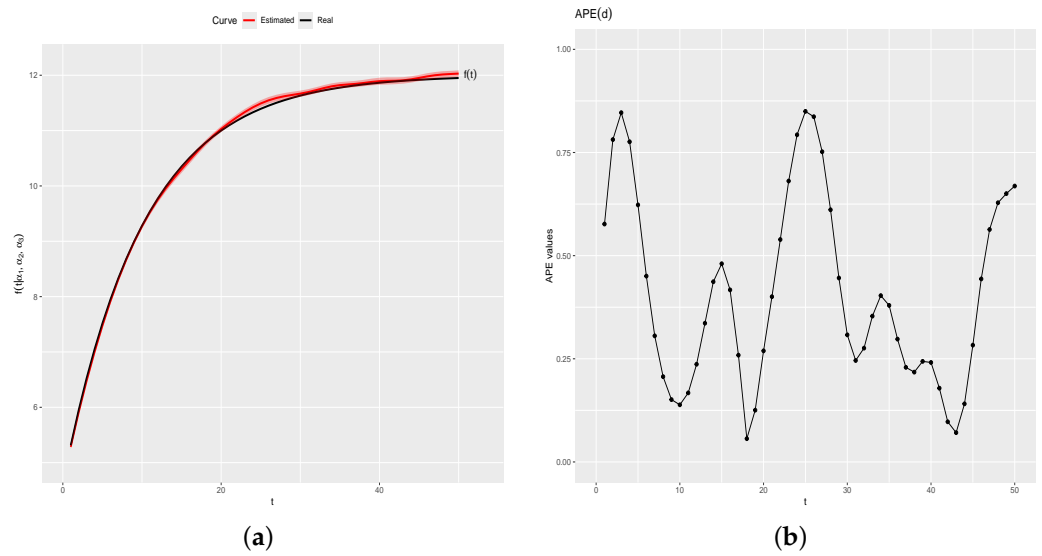


Figure 10. Real and estimated $f(t)$ and $APE(d)$ values. (a) Real and estimated curves. (b) $APE(d)$ values.

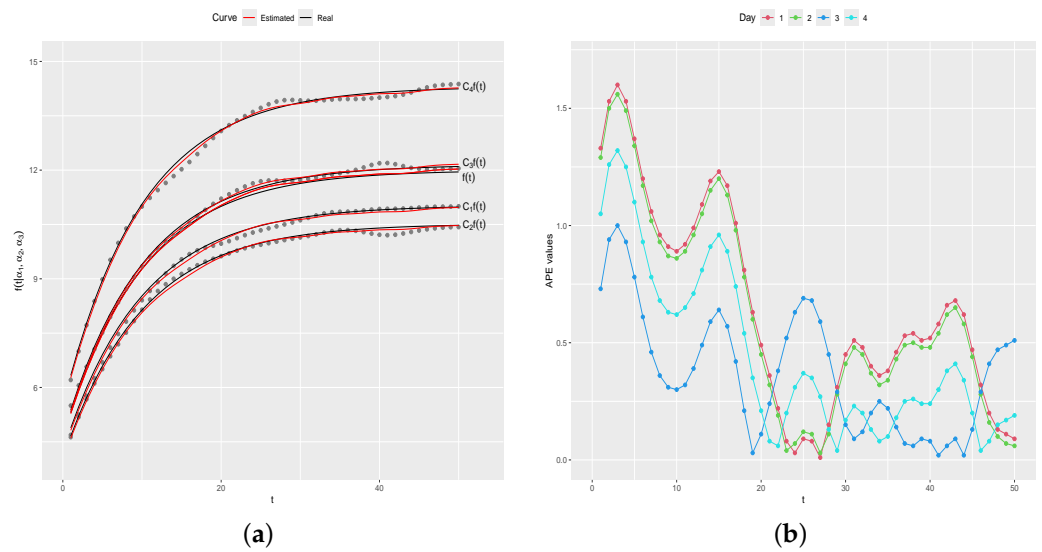


Figure 11. Real and estimated curves and $APE(d_i)$ values, for $i = 1, 2, 3, 4$. (a) Real and estimated curves. (b) $APE(d_i)$ values.

Figure 12a shows the curve for the $(n + 1)$ th day (green line), with the green \bullet symbols representing the generated data for the $(n + 1)$ th day, the predicted curve (red line), and a posterior prediction 95% credibility band (red region). The estimate for C_{n+1} is $\hat{C}_{n+1} = 0.9936$ with a 95% credibility interval given by $(0.7207, 1.2716)$, that is, the real value $C_i = 0.9823$ is inside the credibility interval. Additionally, the real curve is completely inside the 95% posterior prediction band. Figure 12b shows the graph of $APE(d)$ and $APE(e)$ in relation to predicted values. All APE values are smaller than 3, indicating that predicted values are close to real values $C_{n+1}f(t)$ and to the generated values for Y_{n+1} .

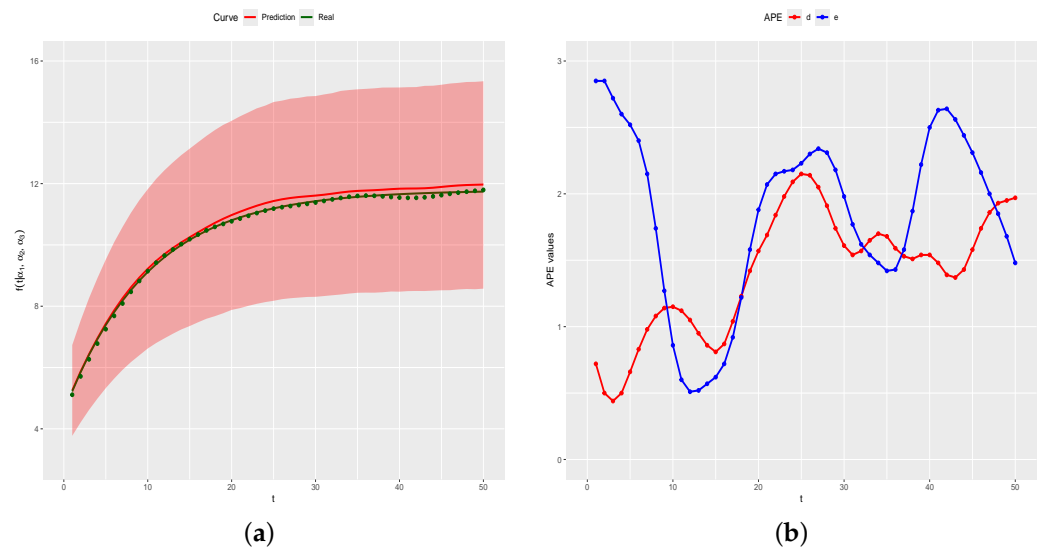


Figure 12. Real and predicted curves and APE values. (a) Real and predicted curves. (b) APE values.

In addition to APE values, we also calculate the root-mean-square error (RMSE) in order to have one more performance measure of the predictions, given by

$$RMSE = \sqrt{\frac{1}{k} \sum_{t=1}^k \left(y_{(n+1)t} - \hat{y}_{(n+1)t} \right)^2},$$

where $y_{(n+1)t}$ is the value generated for the t -th time instant of the $(n + 1)$ -th day, and $\hat{y}_{(n+1)t}$ is the respective predicted value, for $t = 1, \dots, k$. The RMSE value obtained was 0.2021, that is, similar to the APE values, the RMSE value also indicates that the predicted values are satisfactorily close to the generated values.

In order to avoid restricting the model to the results of just one artificial dataset, we repeat the second simulation study $M = 100$ times and calculate the percentage of times that the real curve for the $(n + 1)$ th day is completely inside the prediction band of 95% and the average of the APE and RMSE values. Overall, in 96% of simulated cases, the real curve of $C_{n+1}f(t)$ is completely inside the posterior prediction band, the average of the MAPE values is 0.9550, and the average of the RMSE values is 0.1534. Figure 13 shows the APE and RMSE values for the $M = 100$ simulations. Note that both results show a very satisfactory performance of the proposed method. As an illustration of the predictions results, Figure 14 shows the predicted curve with a 95% posterior prediction band (red region) and the real curve for the 18th and 27th simulations. For these two simulations, the MAPE values were 0.6115 and 4.0017, respectively; and the RMSE values were 0.0622 and 0.4996, respectively.

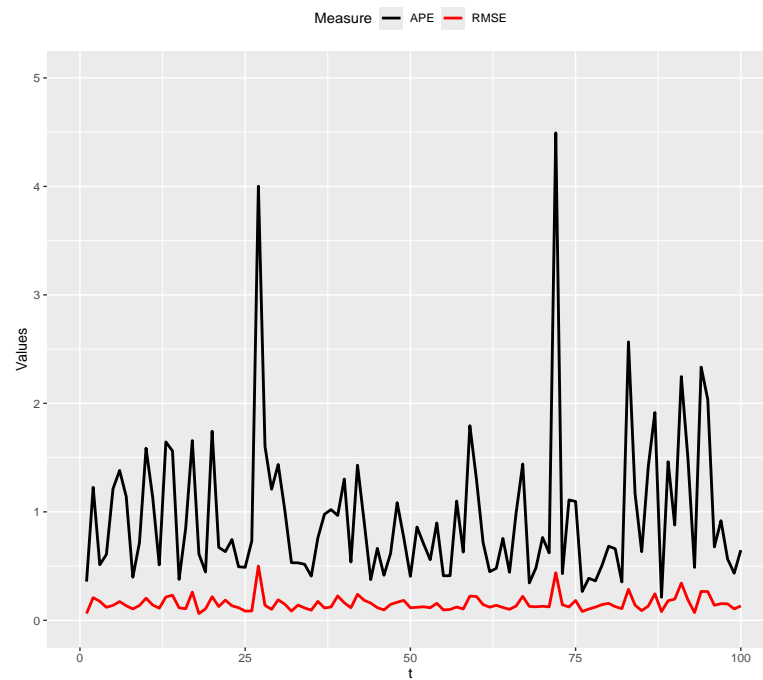


Figure 13. APE and RMSE values.

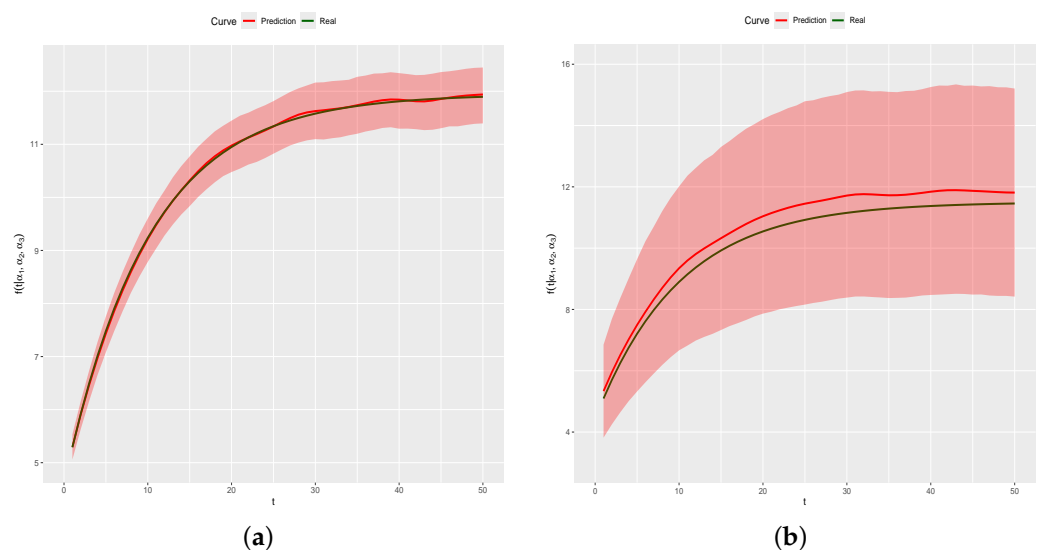


Figure 14. Real and predicted curves for the 18th and 27th simulations. (a) 18th. (b) 27th.

4. Application

We now apply the proposed approach to the real dataset described in Section 2. For this application, we use the same hyperparameter values and the same L , B , and J values used in the two simulation studies. Additionally, we use $n = 4$, i.e., we apply the proposed approach for estimating the curve of $f(t)$ using the dataset of four days, and then we predict the curve for the $(n + 1)$ th day. This procedure was applied for the data recorded over the first 19 days of the experiment, always using a window of 4 days, to obtain the estimated curve of $f(t)$ and to predict the curve for the $(n + 1)$ th day. Thus, overall, 15 analyses were carried out with predictions for days 5 to 19.

Our first application considers the recorded data on the first four days of the experiment to estimate the curve of $f(t)$, and then we predict the curve for the fifth day ($n + 1 = 5$). Figure 15a shows the average values recorded in the first four days (symbols ●), the estimated curve of $f(t)$ (red line) and a 95% credibility band (red region). Figure 15b

shows APE values for comparison between \bar{y} and the estimated values $\hat{y} = (\hat{y}_1, \dots, \hat{y}_n)$. The MAPE value is 0.2590. These results shows that the estimated values are very close to the recorded average values. In other words, the proposed approach presented a very satisfactory performance in estimating the average values recorded over the first four days on which the experiment was carried out.

Table 4 shows MAPE and RMSE values for the 15 analyses. MAPE values range from a minimum of 0.0095 for day 13 to a maximum of 0.5360 for day 11, with an average value of 0.2349. RMSE values range from 0.0011 for day 13 to a maximum of 0.0612 for day 11, with an average value of 0.0265. As one can see, all MAPE and RMSE values are near zero, indicating that the estimated values \hat{y} are close to the recorded \bar{y} for the 15 analyses.

Table 4. MAPE and RMSE values for analyses 1 to 15.

Analysis	MAPE	RMSE	Analysis	MAPE	RMSE	Analysis	MAPE	RMSE
1	0.2590	0.0292	6	0.2038	0.0228	11	0.5360	0.0612
2	0.1395	0.0155	7	0.3670	0.0393	12	0.1815	0.0208
3	0.0500	0.0061	8	0.2378	0.0273	13	0.0095	0.0011
4	0.0753	0.0085	9	0.2157	0.0243	14	0.3445	0.0394
5	0.3996	0.0450	10	0.4903	0.0542	15	0.0146	0.0017

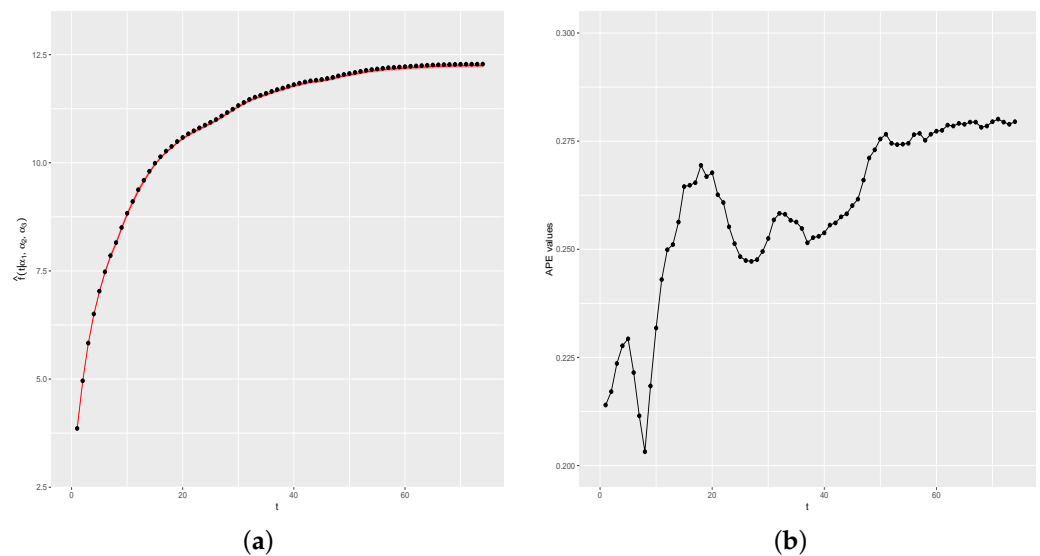


Figure 15. Estimated curve of $f(t)$ and APE(e) values. (a) Estimated curve of $f(t)$. (b) APE(e) values.

Table 5 shows MAPE and RMSE values for the 15 predictions. As one can note, MAPE values vary from a minimum of 0.3344 for day 19 to a maximum of 7.1648 for day 13. The average value of MAPE over the 15 days was 2.5719. Similarly, RMSE values range from 0.0382 for day 19 to a maximum of 0.7410 for day 13, with an average value of 0.2895. Overall, these results show that the predicted values \hat{y}_i^{pred} are close to the recorded values y_i , for $i = 5, \dots, 19$.

As an illustration of the good performance of the proposed approach in predictions, Figure 16 shows the recorded values (symbols ●), the predicted curve (red line), and a 95% prediction credibility band (red region) for the recorded values on days 9 and 19, which are the two days with the smallest MAPE and RMSE values. Figure 17 shows the prediction results for days 13 and 14, which are the two days with the greatest MAPE and RMSE values. Although predictions for days 13 and 14 present the two highest MAPE values, most of the recorded values are inside the 95% prediction credibility band. Overall, the

proposed approach presented very satisfactory performance, as indicated by the small MAPE and RMSE values.

Table 5. MAPE and RMSE values for the predictions for days 5 to 19.

Day	MAPE	RMSE	Day	MAPE	RMSE	Day	MAPE	RMSE
5	2.8659	0.3180	10	2.8855	0.3323	15	0.6193	0.07321
6	2.3021	0.2628	11	5.1500	0.5780	16	0.5880	0.0677
7	5.1190	0.5321	12	3.6303	0.4415	17	0.5753	0.0786
8	1.0693	0.1222	13	7.1648	0.7410	18	0.7374	0.1128
9	0.3507	0.0409	14	5.1875	0.6027	19	0.3344	0.0382

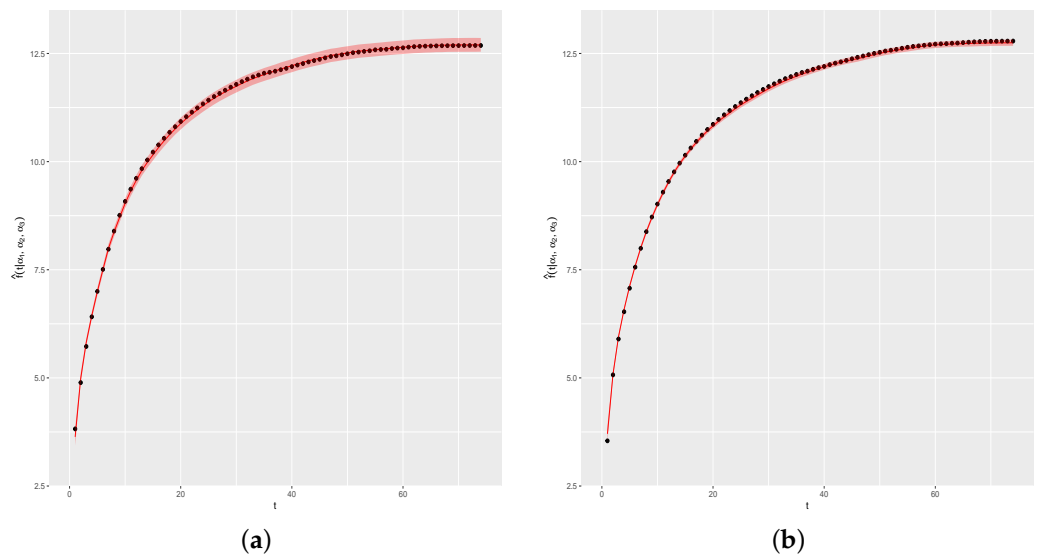


Figure 16. Recorded and predicted values for days 9 and 19. (a) Day 9. (b) Day 19.

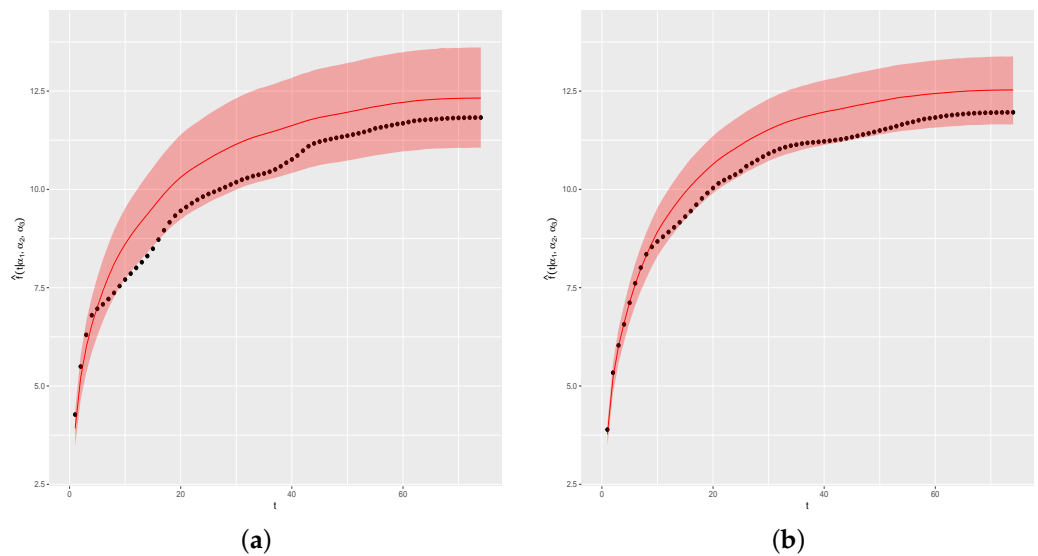


Figure 17. Recorded and predicted values for days 13 and 14. (a) Day 13. (b) Day 14.

5. Final Remarks

In this paper, we propose a Bayesian approach for modeling and forecasting photovoltaic solar power generation. For this, we assume that the growth curve of the generated

power over the time of day is proportional to an average curve whose associated function is denoted by $f(t)$. However, instead of taking a parametric approach by setting up $f(t)$ as a known mathematical function, we assume that $f(t)$ is an unknown function, but with the vector of values $\mathbf{f}(\mathbf{t}) = (f(1), \dots, f(k))$ generated a priori from a Gaussian process. To perform inference for the parameters of interest θ , we use a Gibbs sampling algorithm.

The good performance of the proposed approach and its four advantages as described in Section 1 were illustrated by means of two simulation studies and an application to a real dataset. The results obtained show that the proposed approach is an efficient alternative for modeling the solar power generated on the days considered in the study and also for forecasting next-day solar power generation.

From a practical point of view, the results show that the proposed modeling and the estimation procedure were very accurate in predicting energy generation for the next day. Although the proposed approach has been described as forecasting the growth curve for the next day, it can also be used to forecast power generation for short intervals, such as hourly power generation. An extension of the approach presented here is the inclusion of explanatory variables in the modeling since power generation is influenced by environmental variables such as temperature and irradiance. All computational implementations were carried out using the R software [19], and the code can be obtained by e-mailing the authors.

Author Contributions: Conceptualization, E.F.S. and C.A.d.B.P.; Data curation, M.V.F. and E.F.S.; Formal analysis, M.V.F. and E.F.S.; Methodology, M.V.F., E.F.S. and C.A.d.B.P.; Project administration, E.F.S.; Software, M.V.F. and E.F.S.; Supervision, E.F.S. and C.A.d.B.P.; Writing—review and editing, M.V.F., C.A.d.B.P. and E.F.S. All authors have read and agreed to the published version of the manuscript.

Funding: This study was financially supported by the Brazilian institutions CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), process number 402154/2023-1, and Fundect (Fundação de Apoio ao Desenvolvimento do Ensino, Ciência, e Tecnologia do Estado de Mato Grosso do Sul), TO number 120/2024, SIAFIC 818, process number 83/026.835/2024.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: The real dataset is freely available on the websites cited in the article. It also can be obtained by emailing the authors.

Acknowledgments: The authors acknowledge the Universidade Federal de Mato Grosso do Sul for all support.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. AlKandari, M.; Ahmad, I. Solar power generation forecasting using ensemble approach based on deep learning and statistical methods. *Appl. Comput. Inform.* **2024**, *20*, 231–250. [[CrossRef](#)]
2. Sharadga, H.; Hajimirza, S.; Balog, R.S. Time series forecasting of solar power generation for large-scale photovoltaic plants. *Renew. Energy* **2020**, *150*, 797–807. [[CrossRef](#)]
3. Ibrahim, S.; Daut, I.; Irwan, Y.M.; Irwanto, M.; Gomes, N.; Farhana, Z. Linear Regression Model in Estimating Solar Radiation in Perlis. *Energy Procedia* **2012**, *18*, 1402–1412. [[CrossRef](#)]
4. Asri, R.; Friansa, K.; Siregar, S. Predicting Solar Irradiance Using Regression Model (Case Study: ITERA Solar Power Plant). *IOP Conf. Ser. Earth Environ. Sci.* **2021**, *830*, 012080. [[CrossRef](#)]
5. Erten, M.Y.; Aydilek, H. Solar Power Prediction using Regression Models. *Int. J. Eng. Res. Dev.* **2022**, *14*, s333–s342. [[CrossRef](#)]
6. El-Aal, S.A.; Alqabli, M.A.; Naim, A.A. Forecasting solar photovoltaic energy production using linear regression-based techniques. *J. Theor. Appl. Inf. Technol.* **2023**, *101*, 3326–3337.
7. Bimenyimana, S.; Osarumwense, G.N.; Lingling, L. Output Power Prediction of Photovoltaic Module Using Nonlinear Autoregressive Neural Network. *J. Energy, Environ. Chem. Eng.* **2017**, *2*, 32–40.
8. Pamain, A.; Rao, P.V.K.; Tilya, F.N. Prediction of photovoltaic power output based on different non-linear autoregressive artificial neural network algorithms. *Glob. Energy Interconnect.* **2022**, *5*, 226–235. [[CrossRef](#)]
9. Rogier, J.K.; Nawaz, M. Forecasting Photovoltaic Power Generation via an IoT Network Using Nonlinear Autoregressive Neural Network. *Procedia Comput. Sci.* **2019**, *151*, 643–650. [[CrossRef](#)]

10. Mellit, A.; Saglam, S.; Kalogirou, S.A. Artificial neural network-based model for estimating the produced power of a photovoltaic module. *Renew. Energy* **2013**, *60*, 71–78. [[CrossRef](#)]
11. Amer, H.N.; Dahlan, N.Y.; Azmi, AMLatip, M.F.A.; Onn, M.S.; Tumian, A. Solar power prediction based on Artificial Neural Network guided by feature selection for Large-scale Solar Photovoltaic Plant. *Energy Rep.* **2023**, *9* (Suppl. 12), 262–266. [[CrossRef](#)]
12. Abdelhak, K.; Razika, I.; Ali, B.; Abdelmalek, A.; Müslüm, A.; Nacer, L.; Nabila, I. Solar photovoltaic power prediction using artificial neural network and multiple regression considering ambient and operating conditions. *Energy Convers. Manag.* **2023**, *288*, 117186. [[CrossRef](#)]
13. Akaike, H. A new look at the statistical model identification. *IEEE Trans. Autom. Control* **1974**, *19*, 716–723. [[CrossRef](#)]
14. Schwarz, G. Estimating the dimension of a model. *Ann. Stat.* **1978**, *6*, 461–464. [[CrossRef](#)]
15. Gelf, A.E.; Smith, A.F.M. Sampling-Based Approaches to Calculating Marginal Densities. *J. Am. Stat. Assoc.* **1990**, *85*, 398–409. [[CrossRef](#)]
16. Gelman, A.; Carlin, J.B.; Stern, H.S.; Dunson, D.B.; Vehtari, A.; Rubin, D.B. *Bayesian Data Analysis*; CRC Press: Boca Raton, FL, USA, 1995; Volume 2.
17. Geman, S. and Geman, D. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Trans. Pattern Anal. Mach. Intell.* **1984**, *6*, 721–741. [[CrossRef](#)] [[PubMed](#)]
18. Souza, G.; Santos, R.R.; Saraiva, E.F. A Log-logistic predictor for power generation in photovoltaic systems. *Energies* **2022**, *15*, 5973. [[CrossRef](#)]
19. R Core Team. R: A Language and Environment for Statistical Computing. Available online: <https://www.R-project.org/> (accessed on 15 January 2024).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.