mathematics

MDPI

Review

An Overview of Forecast Analysis with ARIMA Models during the COVID-19 Pandemic: Methodology and Case Study in Brazil

Raydonal Ospina [1,2], João A. M. Gondim [3], Víctor Leiva [4,*] and Cecilia Castro [5]

[1] Department of Statistics, Universidade Federal de Pernambuco, Recife 50670-901, Brazil; raydonal@de.ufpe.br
[2] Department of Statistics, Universidade Federal da Bahia, Salvador 40170-110, Brazil
[3] Department of Mathematics, Universidade Federal de Pernambuco, Recife 50670-901, Brazil; joao.gondim@ufpe.br
[4] School of Industrial Engineering, Pontificia Universidad Católica de Valparaíso, Valparaíso 2362807, Chile
[5] Centre of Mathematics, Universidade do Minho, 4710-057 Braga, Portugal; cecilia@math.uminho.pt
* Correspondence: victor.leiva@pucv.cl or victorleivasanchez@gmail.com

Abstract: This comprehensive overview focuses on the issues presented by the pandemic due to COVID-19, understanding its spread and the wide-ranging effects of government-imposed restrictions. The overview examines the utility of autoregressive integrated moving average (ARIMA) models, which are often overlooked in pandemic forecasting due to perceived limitations in handling complex and dynamic scenarios. Our work applies ARIMA models to a case study using data from Recife, the capital of Pernambuco, Brazil, collected between March and September 2020. The research provides insights into the implications and adaptability of predictive methods in the context of a global pandemic. The findings highlight the ARIMA models' strength in generating accurate short-term forecasts, crucial for an immediate response to slow down the disease's rapid spread. Accurate and timely predictions serve as the basis for evidence-based public health strategies and interventions, greatly assisting in pandemic management. Our model selection involves an automated process optimizing parameters by using autocorrelation and partial autocorrelation plots, as well as various precise measures. The performance of the chosen ARIMA model is confirmed when comparing its forecasts with real data reported after the forecast period. The study successfully forecasts both confirmed and recovered COVID-19 cases across the preventive plan phases in Recife. However, limitations in the model's performance are observed as forecasts extend into the future. By the end of the study period, the model's error substantially increased, and it failed to detect the stabilization and deceleration of cases. The research highlights challenges associated with COVID-19 data in Brazil, such as under-reporting and data recording delays. Despite these limitations, the study emphasizes the potential of ARIMA models for short-term pandemic forecasting while emphasizing the need for further research to enhance long-term predictions.

Keywords: ARIMA forecasting; epidemiological forecasting; pandemic analytics; predictive modeling; public health intelligence

MSC: 62M10

Citation: Ospina, R.; Gondim, J.A.M.; Leiva, V.; Castro, C. An Overview of Forecast Analysis with ARIMA Models during the COVID-19 Pandemic: Methodology and Case Study in Brazil. *Mathematics* **2023**, *11*, 3069. https://doi.org/10.3390/math11143069

Academic Editors: Maria Laura Manca, Hongbin Fang and Jose Luis Vicente Villardon

Received: 17 May 2023
Revised: 6 June 2023
Accepted: 6 July 2023
Published: 12 July 2023

1. Introduction

The emergence of the Sars-CoV-2 coronavirus at the end of 2019 marked the onset of a global pandemic that has since infected over 560 million people and resulted in over six million fatalities, according to data from the Worldometers platform [1]. Being detected for the first time in the Chinese city of Wuhan, the virus quickly spread across the globe, leading to the World Health Organization to officially declare a pandemic on 11 March 2020. In response to this formidable challenge, the scientific community has mobilized to understand and combat the disease, known as COVID-19 [2,3].

The research efforts have been diverse, ranging from the pursuit of effective treatments and vaccines to mathematical and statistical modeling aimed at forecasting potential infection trends [4–6]. One of the early contributions in this field is attributed to the study presented in [7], which gained international prominence for its insights.

Many researchers have drawn on the famous SIR—susceptible, infectious, recovered—model to construct deterministic structures based on differential equations [8,9]. In [10], a generalized SEIR—susceptible, exposed, infectious, recovered—model incorporating self-protection and quarantine compartments was proposed. In [11], an adjusted SEIR model that includes a quarantined population was indicated, and in [12], a further generalized version of this model for an age-structured population was presented. In [13], the ongoing trajectory of the outbreak was simulated using an age-structured SEIR model, while in [14] a SEIR model was stated to parameterize the intervention effects of the control measures. An optimal control theory [15] was applied to these epidemic models to derive optimal strategies for easing restrictive measures, as showcased in [16].

Nonpharmacological measures to combat COVID-19 have also been modeled in [17,18]. The same was carried out in [19] using branching processes. Other research [20,21] focused on the parametric modeling of growth curves, such as the use of the beta logistic model to analyze mortality curves and second waves of the pandemic. Similarly, in [22], the generalized Richards and growth models were used to analyze the COVID-19-infected cases in China. In [23], a cluster analysis of COVID-19 mortality according to sociodemographic factors at municipal level in Mexico was performed.

As it is well known, a crucial aspect of pandemic modeling is the ability to forecast the trajectory of infection and death rates. Several studies applied machine learning (ML) techniques to this end [24–26], while others [27–29] utilized deep learning and time series analysis. In [29], a combination of autoregressive integrated moving average (ARIMA) models with an association rule of mining techniques was proposed. This combination of models and techniques was designed to identify prognostic factors and to enable the efficient prediction of COVID-19 case numbers, thereby enhancing crisis management capabilities during the pandemic.

Artificial intelligence and deep learning techniques have found numerous applications across diverse domains over the years, including healthcare, manufacturing, environmental monitoring, and reliability engineering [30–32]. These techniques have also been employed for COVID-19-related tasks, such as predicting the course of the disease and enhancing diagnostic accuracy from medical imaging [33]. Another study proposed a forecasting model for COVID-19 using long short-term memory networks, a specific type of deep learning model [34]. As pointed out in [35], there is a growing recognition of the potential for the combination of artificial intelligence methods and mathematical techniques to reach precise and reliable approaches [36]. Mathematically-based models such as the modified SEIR and long short-term memory structures were considered in [37,38]. Moreover, in [39], a fusion of ML and mathematical models was assessed to enhance the precision and robustness of near-future COVID-19 pandemic predictions. This fusion involved integrating random forest, gradient boosting, k-nearest neighbors, and kernel ridge regression ML models with Bertalanffy, Gompertz, logistic, and Richards models. The analysis of such models utilized daily case data, vaccination rates, mobility statistics, and weather conditions to generate the forecasts. This innovative fusion of mathematical and ML models highlights the aim to leverage the strengths of diverse modeling techniques for more effective pandemic prediction.

Within the realm of time series data, ARIMA models have been widely studied and applied in epidemic disease prediction [40–44]. However, in [45], it was argued that these models must be suitable for dealing with complex and dynamic problems. Interestingly, despite often being overlooked in pandemic forecasting due to perceived incompatibility with complex and dynamic contexts, ARIMA models can yield promising results. The exploration of the accuracy of ARIMA model predictions compared to real data in Kuwait, as carried out in [46], exemplifies these promising results.

Recently, in [47], ML models were applied to COVID-19 data to forecast the impact of SARS-CoV-2 in South Asian Association for Regional Cooperation (SAARC) countries and globally; see [48] for the case of South American countries. The authors evaluated and compared various forecasting models, including ARIMA, GLMNet, random forest, and extreme gradient boosting (XGBoost), using selection criteria. These criteria employed performance metrics to pinpoint the most suitable and effective models. Their findings underscored the ARIMA model's superiority in accurately predicting confirmed COVID-19 cases in five out of the eight SAARC countries. Specifically, the model outperformed others in predicting cases in some of these countries [47]. In [49], the application of a hybrid forecasting model was investigated, combining both ARIMA and neural network models to predict daily COVID-19 cases, while in [50] a hybrid ARIMA-WBF, a wavelet-based forecasting model with a similar objective, was proposed. Concerning hybrid models, in [51], a multi-layer perceptron neural network was combined with ARIMA methodologies for the outbreak prediction. The COVID-19 pandemic has accelerated the study, development, and application of various disease prediction methods, often presented and used in combination, as seen in hybrid models, with the aim of achieving more accurate results. Among the various methods, ARIMA models still stand as a promising approach in this context, often demonstrating competitive performance compared to other models.

Drawing on insights from previous research, we conduct here a case study focused on the first wave of the COVID-19 pandemic in Recife, the capital city of the Brazilian state of Pernambuco. This city, home to approximately 1.6 million people according to the most recent estimates from the Brazilian Institute of Geography and Statistics (IBGE) [52], detected its first COVID-19 cases on 12 March 2020. Just over a week later, a state of public calamity was declared in Recife, as documented officially by the executive branch of its municipality [53]. As the number of cases increased, restrictions were imposed on the movement of individuals, as well as mandatory mask use in public zones of Recife and four other cities in its Metropolitan Region (Camaragibe, Olinda, Jaboatão dos Guararapes, and São Lourenço da Mata). These restrictions were lifted on 31 May 2020, but mask use remained mandatory.

Our case study analyzes the initial six months of the COVID-19 outbreak in Recife, Brazil, with a primary focus on the application and efficacy of ARIMA models in short-term forecasting. The importance of precise short-term predictions in the context of a rapidly progressing pandemic cannot be overstated. These predictions are essential to enable rapid strategic responses aimed at reducing the spread of the disease.

In the present study, we leverage ARIMA models for forecasting COVID-19 cases. Despite the increasing prevalence of such usage in recent years, the power of ARIMA models is still noteworthy, particularly in generating reliable short-term predictions. While ARIMA models may not perfectly capture the non-linearity inherent in the COVID-19 data, it has shown considerable utility in facilitating swift and informed decision-making processes within the domain of public health strategies. Moreover, we introduce a variety of predictive ML techniques. These techniques, although not directly applied in our forecasting of COVID-19 cases, were presented in this overview to expand the range of understanding regarding the available methodological tools in this area of research.

Our dataset for this study represents a highly volatile and critical phase of the pandemic. Regardless of its limitations, these data effectively demonstrated the resilience and dependability of ARIMA models in generating accurate short-term forecasts amid a rapidly evolving scenario. The obtained results emphasize the potential value of such models in the context of pandemic forecasting. Furthermore, we would like to highlight that this article not only presents a case study, but also acts as a comprehensive overview of the current state of research on pandemic modeling and prediction.

The article is organized as follows. The introduction in Section 1 includes a literature review of existing research on the topic. Section 2 serves as a mathematical background and is divided into three subsections, which provide ML techniques pertinent to conducting predictions, state ARIMA models, discuss the parameter optimization process, and present

key metrics for assessing the performance of statistical or ML models. In Section 3, we describe the dataset, conduct the data analysis, select six periods of interest, and compare the predictions made by the ARIMA model with observed data. Finally, Section 4 concludes the work, summarizing our findings and offering final remarks.

2. Mathematical Background

This section provides background of fundamental mathematical concepts and key ML techniques. The purpose of this background is to establish a comprehensive foundation of the tools and methodologies that are employed in data analysis and predictive modeling.

2.1. Predictive Machine Learning Techniques

ML techniques, including deep learning, constitute a subfield of computer science that engage in the training process by leveraging historical data. The deep learning techniques are particularly relevant given their ability to learn from large datasets, make predictions, and recognize patterns in a way that surpasses many traditional ML methods. The chosen algorithm for ML and deep learning determines an appropriate model, the selection of which is influenced by data attributes, with the ultimate goal of predicting future values. Within this ML paradigm, the algorithm employs a dataset, consisting of input instances and output values (or targets), for the purpose of model training. Once trained, the model is capable of generating predictions for a test dataset not previously found [54]. These ML strategies harness both regression and classification to facilitate predictive modeling.

Support vector machines, neural networks, logistic regression, and deep learning models have been leveraged to interpret and project the progression of COVID-19 [33,55].

While the use of generalized linear models via penalized maximum likelihood (GLMNet), random forest, XGBoost, and deep learning techniques may not be directly applicable to our specific data analysis, their brief discussion is provided due to the prominence and applicability of these techniques in the broader field of ML. A concise discussion of such techniques enriches our understanding of the diversity and capabilities of ML algorithms, even though they are used in our analysis. They provide a frame of reference and a foundation that could inspire future research directions or alternative approaches to data analysis. For example, GLMNet, which blends LASSO—least absolute shrinkage and selection operator—and ridge regression techniques, serves as an effective tool for variable selection and regularization in datasets with potentially correlated covariates, and therefore, it provides a helpful perspective on dealing with multicollinearity in ML applications. Moreover, the inclusion of deep learning, a significant branch of ML, adds depth to our overview. Deep learning has become increasingly popular due to its superior performance in handling large, high-dimensional data, and its proficiency in capturing complex patterns. It has been effectively used in various fields, including image recognition, natural language processing, and indeed, pandemic prediction and progression modeling. The exploration of deep learning methods in this background adds a crucial dimension to the understanding of available tools and methodologies for predictive modeling.

GLMNet employs the elastic net, a form of the shrinkage method. It includes ridge regression and LASSO as special cases. This property of GLMNet allows it to handle problems where the number of variables p is much greater than the size of the sample n (that is, $n << p$). The elastic net introduces a penalty term formulated as

$$P_\gamma(\boldsymbol{\alpha}) = \sum_{j=1}^{p} (0.5(1-\gamma)\alpha_j^2 + \gamma|\alpha_j|),$$

into the objective function [56], which can be expressed as

$$\min_{(\alpha_o, \boldsymbol{\alpha}) \in \mathbb{R}^{p+1}} \left\{ \frac{1}{2n} \sum_{i=1}^{n} \left(Y_i - \alpha_o - \boldsymbol{x}_i^\top \boldsymbol{\alpha} \right)^2 + \tau P_\gamma(\boldsymbol{\alpha}) \right\}. \quad (1)$$

In the objective function presented in (1), γ is a parameter that determines the type of shrinkage applied during the estimation process, while τ is a penalty parameter that manages the degree of shrinkage. The response variable is represented by $Y_i \in \mathbb{R}$, and the values of the covariate vector are denoted as $\boldsymbol{x}_i \in \mathbb{R}^p$, for $i \in \{1, \ldots, n\}$. The regression intercept is α_o, and the regression coefficients associated with the covariate vector are in the form of $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_p)$, where each α_j is an element of the vector $\boldsymbol{\alpha}$, for $j \in \{1, \ldots, p\}$. Note that p and n, respectively, represent the number of covariates and the sample size of the dataset. We can assume a linear regression structure represented by $\mathrm{E}(Y|\boldsymbol{X} = \boldsymbol{x}) = \alpha_o + \boldsymbol{x}^\top \boldsymbol{\alpha}$, where "E" signifies the expected value of Y conditional to $\boldsymbol{X} = \boldsymbol{x}$. The `glmnet` package of the `R` software provides various regression methods for variable selection and prediction, especially when $n << p$. If $\gamma = 1$, the package defaults to LASSO with l_1 penalty, enabling both parameter shrinkage and variable selection. If $\gamma = 0$, ridge regression with L2 penalty is implemented. For optimal use of the elastic net, a value of $0 < \gamma < 1$ is considered. The value of γ selectively shrinks certain coefficients to zero, enabling sparse selection. The model structure requires variable selection to form a subset of covariates. The elastic net approach is designed to handle the $p >> n$ problem, indicating that the number of parameters significantly exceeds the sample size used in modeling. It is particularly effective when groups of highly correlated covariates are formed. The model combines variable selection with its structure to enhance accuracy. If a group of covariates is highly correlated and one variable is chosen for the sample, the entire group is automatically included in the model.

As GLMNet, random forest is an ensemble learning method that operates by constructing multiple decision trees and outputs the class that represents the mode of the classes (for classification problems) or mean prediction of the individual trees (for regression problems). The use of random forest is justified due to its robustness against overfitting, and its ability to handle large datasets with high dimensionality, which makes it widely applicable in diverse ML scenarios. Random forest operates by combining two methods, bootstrap aggregating (bagging) and the random subspace method, which are used to construct a set of decision trees. Each tree in the ensemble is independently built, and the final output is obtained by aggregating the predictions of these individual decision trees. The construction of a single decision tree in a random forest involves two random selection processes. The first process randomly selects training samples from the dataset with replacement (bootstrapping), while the second process randomly selects a subset of features at each selected sample. Once the decision trees are built, classification or regression predictions are obtained through a voting method. For classification, each tree in the forest votes for a class and the class with the majority of votes is selected as the final output. For regression, the final prediction is the average of the outputs from all the trees. Formally, a random forest can be defined as a collection of decision tree functions $g(\boldsymbol{x}, \varphi_l)$, for $l \in \{1, \ldots, L\}$, where φ_l is an independent random vector parameter, and $\boldsymbol{x}$ represents the input data values. Each decision tree is characterized by a unique random vector of parameters, a distinct set of randomly chosen features, and a unique subset of bootstrapped sample data that serves as the training set. Random forest builds independent and diverse decision trees capitalizing on randomization. Consequently, random forest is considered a robust and versatile classifier, capable of handling a wide range of data scenarios. The random forest approach [57] is represented in Algorithm 1.

Algorithm 1 Structure of the random forest

Step 1: Select L decision trees, n training data that each tree links, and N bootstrap samples.
Step 2: Form L training groups with N times bootstrap samples.
Step 3: Collect m dissection features randomly on a single node from M sample features, for $m << M$.
Step 4: Calculate the best dissection feature according to m features for every decision tree node.
Step 5: Construct each decision tree entirely increasing without trimming.
Step 6: Classify data from the testing set with the random forest model formed by various decision trees.
Step 7: Evaluate the adequacy of the random forest model.
Step 8: Use the random forest model if this is adequate.

XGBoost is a gradient boosting framework that utilizes decision trees and has gained popularity for its computational efficiency and superior performance across a variety of ML tasks. An understanding of this technique provides insights into gradient boosting methods and how they can enhance model performance by reducing both bias and variance. XGBoost is a ML technique that employs an ensemble methodology known as tree boosting. XGBoost is used widely for tasks involving regression and classification, capitalizing on the power of the boosting technique. In each iteration, XGBoost generates a new 'weak' learner and incorporates it into the existing model, thereby gradually improving the overall model's predictive performance. The XGBoost model operates based on the principle of gradient boosting, specifically focusing on optimizing a specific loss function. This principle differentiates XGBoost from the random forest algorithm, where decision trees are built independently. In contrast, the gradient boosting method adds a new tree in each iteration to correct the errors made by the current ensemble of trees. This iterative process continues until a specified number of trees is reached, or the improvement in loss function falls below a threshold. This step-by-step refinement of the model is what characterizes gradient boosting methods like XGBoost. In essence, the XGBoost technique constructs new models to predict the errors of prior models, subsequently making improved predictions. This iterative process enhances the model's predictive capacity over time. Mathematically, the objective function of the XGBoost model can be formulated by the equation stated as $\Theta(F) = L(\lambda) + \Omega(F)$. In this equation, $L(\lambda)$ represents the loss function, which governs the predictive power of the model, while $\Omega(F)$ is the regularization term that helps control the model complexity and prevent overfitting. The balance between these two terms, the predictive power and the control of overfitting, helps to optimize the model's performance.

Deep learning is based on artificial neural networks with representation learning capabilities. Deep learning architectures such as deep neural networks, deep belief networks, recurrent neural networks, and convolutional neural network (CNN) have been applied across various fields, consistently demonstrating remarkable accuracy in tasks driven by pattern recognition in data. Comparatively, methods like random forest, while simpler to train and interpret, can struggle with tasks involving high-level feature interactions and sequence prediction—areas where deep learning models particularly excel. Moreover, deep learning models have the advantage of being able to automatically discover intricate structures in high-dimensional data, a task that often requires manual feature engineering in the case of random forest. Deep learning models are constructed using multiple layers of artificial neurons or nodes, designed to mimic the structure and function of the human brain. By processing data and creating patterns for decision-making, these models are the key technology behind advanced applications, including natural language processing, speech recognition, computer vision, and bioinformatics, among others. One of the most common types of deep learning models is the CNN. Primarily used in image processing, CNNs have shown promising results in medical image classification, including COVID-19 diagnosis. A CNN typically comprises an input and output layer, alongside multiple

hidden layers. These hidden layers often include convolutional layers, which are layers with activation functions such as ReLU, pooling layers, and fully connected layers.

Another significant model in deep learning is the recurrent neural network (RNN). RNNs are especially effective when dealing with sequence data, owing to their unique structure, that is, they maintain a form of memory by using their own output as an input for the next step. This makes RNNs particularly effective for tasks like time series analysis and natural language processing, where the order of data points is significant.

2.2. ARIMA Models and Parameter Estimation

ARIMA is a renowned family of time-series models that was originated for its usage in economics [58]. This family of models, capable of predicting future points in a time series dataset, are appreciated for their statistical traits, their capacity to implement a range of exponential smoothing models, and the integration of the Box–Jenkins method during the model training phase. An explanation of these models can be found in [46], and details on computational and theoretical aspects are available, for example, in [59,60].

Before diving into the ARIMA model, it is crucial to understand the backshift operator B, a helpful notational device when dealing with lags of a sequence. For a time series Y_t, the lagged series is denoted by $BY_t = Y_{t-1}$ and similarly, $B^k Y_t = Y_{t-k}$.

The ARIMA model contains three parameters (p, d, q) and can be represented as

$$(1-B)^d Y_t = \mu + \Phi(B)(1-B)^d Y_t = \mu + \Phi(B)Z(t) + \Theta(B)\varepsilon_t,$$

where $\Phi(B)$ is the autoregressive operator of order p, that is,

$$\Phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p,$$

$\Theta(B)$ is the moving average operator of order q, that is,

$$\Theta(B) = 1 + \theta_q 1B + \theta_2 B^2 + \cdots + \theta_q B^p,$$

Z_t is a white noise process with zero mean and constant variance σ^2; and ε_t is the error term.

An ARIMA model operates on the assumption of stationarity, meaning that the time series has constant mean and variance over time. Note that d in ARIMA stands for differencing, which is used to make the series stationary if it is not already. Differencing involves the transformation of the series to differences between consecutive observations. $Y_t - Y_{t-1}$ namely. Differencing can be applied more than once if the series is still not stationary, which is reflected in the value of d in the ARIMA(p, d, q) model. It is usual to represent the ARIMA(p, q, d) model in an alternative form given by

$$Y_t = c + \phi_1 Y_{t-1} + \cdots + \phi_p Y_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t,$$

where Y_t is the observation at time t, ϕ_j is the parameter of the autoregressive part of the model, θ_k is the parameter of the moving average part, and ε_t is the error term.

The autoregressive order p represents the number of lags of Y to be used as predictors. The order of integration d is the number of times the data have had past values subtracted (also known as differencing) to make the time series stationary. Then, q is the moving average order that represents the number of lagged forecast errors that should go into the ARIMA model. The autocorrelation function (ACF) and partial autocorrelation function (PACF) are essential tools in time series analysis, especially for determining the parameters (p, q) of an ARIMA model. On the one hand, the ACF measures the correlation between observations of a time series at two points in time, as a function of the time lag between the two points. The PACF, on the other hand, measures the correlation between observations at two points in time while controlling for the values at all shorter lags.

Once the model orders are identified (that is, the values of p, d, and q), the parameters of the ARMA model can be estimated. Several methods can be used to estimate these parameters, with the maximum likelihood estimation being one of the most com-

mon. The parameters are chosen in a way that maximizes the likelihood function of the observed data for the corresponding the model. Given a sample of n observations from a time series $\{y_1, \ldots, y_n\}$, the likelihood function for an ARIMA model can be written as $L(\Theta; y_1, \ldots, y_n) = f(y_1, \ldots, y_n; \Theta)$, where Θ represents the vector of parameters to be estimated, whereas f is the joint probability density function (PDF) of the observed data and the parameter Θ.

The log-likelihood function is typically used in practice due to its mathematical convenience. The log-likelihood function for an ARIMA model is given by

$$l(\Theta; y_1, \ldots, y_n) = \log(L(\Theta; y_1, \ldots, y_n)).$$

Note that the likelihood function, L, quantifies how well a particular statistical model explains the observed data. In other words, it is a measure of how likely the observed data are for the specific parameters of the model.

For an ARIMA model, the likelihood function is based on the assumption that the errors at each point in time follow a Gaussian or normal distribution. The likelihood of observing the data for the model parameters is calculated by evaluating the PDF of this normal distribution at each data point and then taking the product of these PDFs (assuming independence across time). Mathematically, for a sample of size n, if the errors are denoted by $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n$, the likelihood function is stated as

$$L(\phi, \theta, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\varepsilon_i^2}{2\sigma^2}\right).$$

The maximum likelihood estimates of the parameters are the values that maximize this log-likelihood function. For an ARMA(p, q) model, the parameters to be estimated include the autoregressive coefficients ϕ_j, for $j \in \{1, \ldots, p\}$, the moving average coefficients θ_k, for $k \in \{1, \ldots, q\}$, and the variance of the error term σ^2. The optimization process of the likelihood function for an ARIMA model involves initial values of the process, which are usually unknown. Different ways of dealing with these initial values can lead to different values of the likelihood function and hence different estimates of the parameters. In practice, the exact maximization of the likelihood function can be quite complicated due to the high dimensionality of the parameter space and the possibility of multiple local maxima. Therefore, numerical optimization methods are typically used to find the maximum likelihood estimates of the ARIMA parameters.

In ARIMA structures, model selection can be automated for optimal predictive accuracy. The model orders (p, d, q) can be chosen to minimize the Akaike information criterion (AIC), the corrected Akaike information criterion (AICc), or the Bayesian information criterion (BIC). The formulas for these criteria are given by

$$\text{AIC} = -2\log(L) + 2k, \quad \text{AICc} = \text{AIC} + \frac{2k(k+1)}{n-k-1}, \quad \text{BIC} = -2\log(L) + k\log(n),$$

where L is the likelihood function of the data under the model, n is the sample size, and k is the total number of parameters in the model, with $k = p + q + d + 1$ if the intercept term is non-zero, and $k = p + q + d$ if the intercept term is zero.

Automatic ARIMA modeling is an approach that leverages statistical methodologies and computational capabilities to automatically determine the parameters (p, d, q) of an ARIMA model. A method used for automatic ARIMA modeling is the `auto.arima()` function implemented in the `forecast` package of the `R` software [61], developed in [62]. This function performs a unit root test to decide on the order of differencing (d), and then selects the best (p, q) order based on the AIC. This is performed through a stepwise search or, optionally, through a more exhaustive search. The `auto.arima()` function is summarized in Algorithm 2, which offers an efficient and effective way of selecting the most suitable ARIMA model for a given time series dataset. Starting from an initial ARIMA

model with $(p, d, q) = (0, d, 0)$, the function automatically adjusts the values of p and q using a stepwise search, incrementally increasing or decreasing these parameters based on whether this improves the AIC score. This automated process effectively identifies the appropriate lag values for the autoregressive and moving average parts of the model, traditionally carried out through manual inspection of ACF and PACF plots.

Algorithm 2 Structure of the `auto.arima()` function

Step 1: Determine the optimal order of differencing, d, using unit root tests, such as the augmented Dickey–Fuller or the Kwiatkowski–Phillips–Schmidt–Shin test.
Step 2: Find the values of p and q that minimize the AIC, performing a stepwise search through the model space, starting with $(p, d, q) = (0, d, 0)$ and then incrementally increasing or decreasing the values of p and q based on whether the AIC decreases.
Step 3: Estimate the parameters of the ARIMA model using the maximum likelihood method, where numerical optimization algorithms are employed to find the parameters that maximize the likelihood function of the observed data for the model.
Step 4: Return the ARIMA model with the optimal parameters.
Step 5: Evaluate the adequacy of the ARIMA model.
Step 6: Use the ARIMA model if this is adequate.

The effectiveness of an ARIMA model is primarily evaluated based on forecast accuracy, which is measured using the model's forecast errors. The observed data are divided into two subsets: a training set, which aids in parameter estimation, and a test set, used to ascertain the accuracy of the forecast. Assume that the observed dataset has a size of n, and the test set has a size of m, where $m < n$. Ideally, m should be equivalent to the forecasted time-period. The forecast error is defined as the difference between the observed and forecasted values.

There are several measures utilized to quantify these forecast errors, including scale-dependent indicators like the mean absolute error (MAE), root mean squared error (RMSE), and scale-independent indicators like the mean absolute percentage error (MAPE). In addition to these measures, other error indicators such as the mean absolute scaled error (MASE), the symmetric mean absolute percentage error (SMAPE), and the coefficient of determination, also known as R-squared, can be used. These error measures are extensively utilized in time series forecasting literature for model comparison and validation [62].

2.3. Model Evaluation Metrics

Evaluation metrics are key to assessing the performance of statistical or ML models. They help us to understand how well a model is performing by comparing the predicted values with the observed values. Depending on the problem at hand, different metrics may be more or less suitable. Therefore, there are a variety of metrics to meet various needs.

The MAE is one of the simplest regression metrics that is easy to understand and compute. By calculating the average of absolute differences between predicted and observed values, it provides an intuitive measure of prediction error magnitude and is given by

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|Y_i - F_i|,$$

which states the difference between the observed (Y_i) and forecasted (F_i) values for case i.

The MAPE is used when it is important to represent the prediction error in terms of the relative size of the observed value. This can be helpful in situations where the scale of the data is relevant and it is stated as

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{Y_i - F_i}{Y_i}\right| \times 100\%.$$

The MASE was introduced [62] specifically for time-series forecasting, as it scales the prediction error of a simple benchmark. The MASE allows for meaningful comparisons across different time series, which can have different scales and seasonal patterns and it is formulated as

$$\text{MASE} = \frac{\text{MAE}}{\text{MAE}_{\text{in-sample}}} = \frac{\frac{1}{n}\sum_i |Y_i - F_i|}{\frac{1}{(T-1)}\sum_{t=2}^{T} |Y_t - X_{t-1}|}.$$

Note that the MASE is used as a measure of the accuracy of forecasts, with the denominator $\text{MAE}_{\text{in-sample}}$ being the MAE of the one-step naive forecast method on the training set (here defined as $t \in \{1, \ldots, T\}$), that uses the true value from the prior period as the forecast, that is, $F_t = Y_{t-1}$.

The SMAPE treats over-forecast and under-forecast symmetrically, making it a better choice than the MAPE when these features of forecasting are important. The SMAPE is commonly utilized in demand planning and sales forecasting, and it is expressed as

$$\text{SMAPE} = \frac{1}{n}\left(\sum_{i=1}^{n} \frac{|F_i - Y_i|}{\frac{|F_i|+|Y_i|}{2}}\right) \times 100\%,$$

which is based on the percentage error.

The RMSE is a powerful and commonly used regression metric that penalizes large errors due to squaring the differences. The RMSE makes is particularly helpful when large errors are especially undesirable and it is given by

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - F_i)^2},$$

which corresponds to the square root of the mean square error.

The R-squared is employed when we are interested in explaining the proportion of variance captured by the model. The R-squared is a key metric for understanding how much of the target variable's variability can be explained by our model and it is defined as

$$\text{R-squared} = 1 - \frac{\sum_{i=1}^{n}(Y_i - F_i)^2}{\sum_{i=1}^{n}(Y_i - \overline{Y})^2},$$

3. Case Study in Brazil

In this section, we present a detailed case study that applies the ARIMA model to forecast COVID-19 cases in the city of Recife, Brazil. We begin by describing the data used in our analysis and subsequently discuss the methodology and findings of our data analysis.

3.1. Data

The data pertaining to the COVID-19 pandemic in Recife were sourced from an IO Brazilian platform [63], which is a resource that consolidates public data presenting them in a user-friendly format. IO platforms are a generic domain that is popular in the tech world since IO or I/O means input/output in computer science.

The data under consideration cover from 12 March 2020, when the first two cases were identified in Recife, up until 15 September 2020. It was necessary to process the dataset for covering the gap of the missing days, that is, 16, 19, and 20 March, 6 May, and 6 September. This gap was bridged by linearly interpolating the last day preceding and the first day following the missing cases, and then rounding to the nearest whole number.

For model training, we selected a period of the first 60 days, from 12 March to 10 May 2020. This is because the city's restrictive measures to curb the pandemic were instituted on 11 May 2020.

The mentioned period was further extended until 31 May 2020, resulting in a total of 21 days. We adopted this duration as a time unit and chose six periods of 21 days each to

compare the model forecasts with the observed data, despite the fact that the more stringent measures were only enforced during the first of these periods. Such periods are detailed in Table 1. By the end of the last period, the state of Pernambuco was nearing the ending of the first wave, as indicated by a downward trend in the daily number of new COVID-19 cases. This trend is illustrated in Figure 1, where the solid line represents a seven-day moving average of the daily cases, a common technique used to smooth out short-term fluctuations and highlight longer-term trends in the data.

Table 1. Periods selected for forecasts covering from 12 March 2020 to 15 September 2020.

Period	Beginning	End	Duration
Training	12 March	10 May	60 days
1	11 May	31 May	21 days
2	1 June	21 June	21 days
3	22 June	12 July	21 days
4	13 July	3 August	21 days
5	4 August	24 August	21 days
6	25 August	15 September	21 days

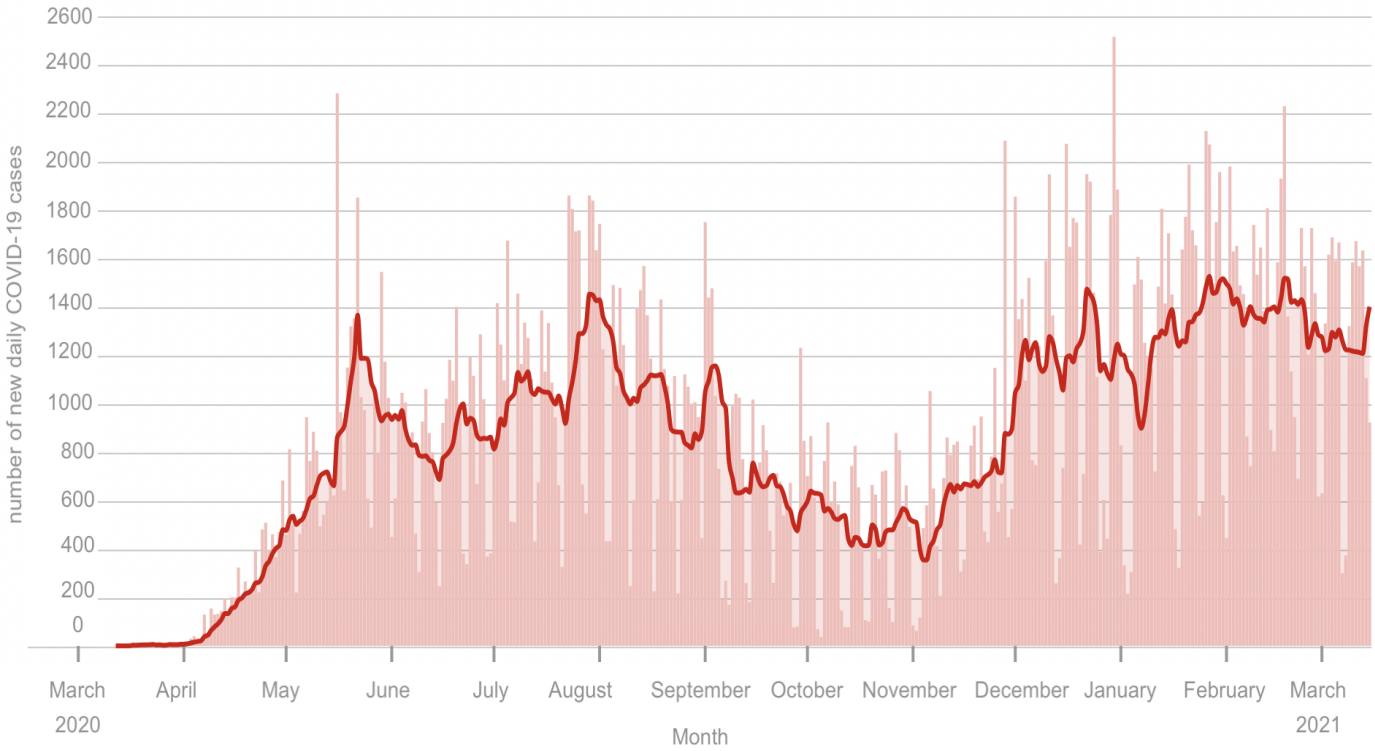

Figure 1. Time series of the number of new daily COVID-19 cases in Pernambuco, Brazil. Adapted from https://pt.wikipedia.org/wiki/Pandemia_de_COVID-19_em_Pernambuco, last updated on 15 March 2021 (accessed on 08 July 2023).

3.2. Data Analysis and Comparisons

Next, we present the analysis of the data using the ARIMA model. We discuss the methodology employed, including model selection and parameter estimation. We also present the results of the ARIMA forecasts and compare them with the observed data.

Additionally, we evaluate the accuracy of the predictions and discuss any discrepancies or limitations encountered during the analysis. The `auto.arima()` function was used to automatically select the best parameters p, d, and q for the ARIMA model based on the training data. The result was that the ARIMA(2,2,1) model provided the best fit for the training data, meaning it minimized the forecast error better than other models with different parameters. Thus, the prediction for the observed value of the time series is based on two past values, where two levels of differencing were required to make the series stationary, and the observed error (residual) is based on a past error equal to one. The residuals from this model are depicted in Figure 2. It is important to note that nearly all lags from the ACF fall within the confidence interval. Moreover, the histogram for the residuals shows a likeness to a normal distribution.

We conducted a validation of the ARIMA(2,2,1) model by comparing it with eight other selected ARIMA models. The criteria used for this comparative analysis included AIC, AICc, BIC, RMSE, MAE, and MAPE. The results of this comparison are presented in Table 2. It is noteworthy that the ARIMA(2,2,1) model, despite being selected by the `auto.arima()` function, does not possess the minimum error values across all measures. Specifically, it ranks third lowest in terms of RMSE, it has the minimum MAE, and its MAPE is the third highest. However, this model is preferred because it has the lowest values for AIC, AICc, and BIC. These information criteria consider both the model's goodness of fit and complexity. Therefore, we conclude that the ARIMA(2,2,1) model provides the best balance of fit and simplicity for the data.

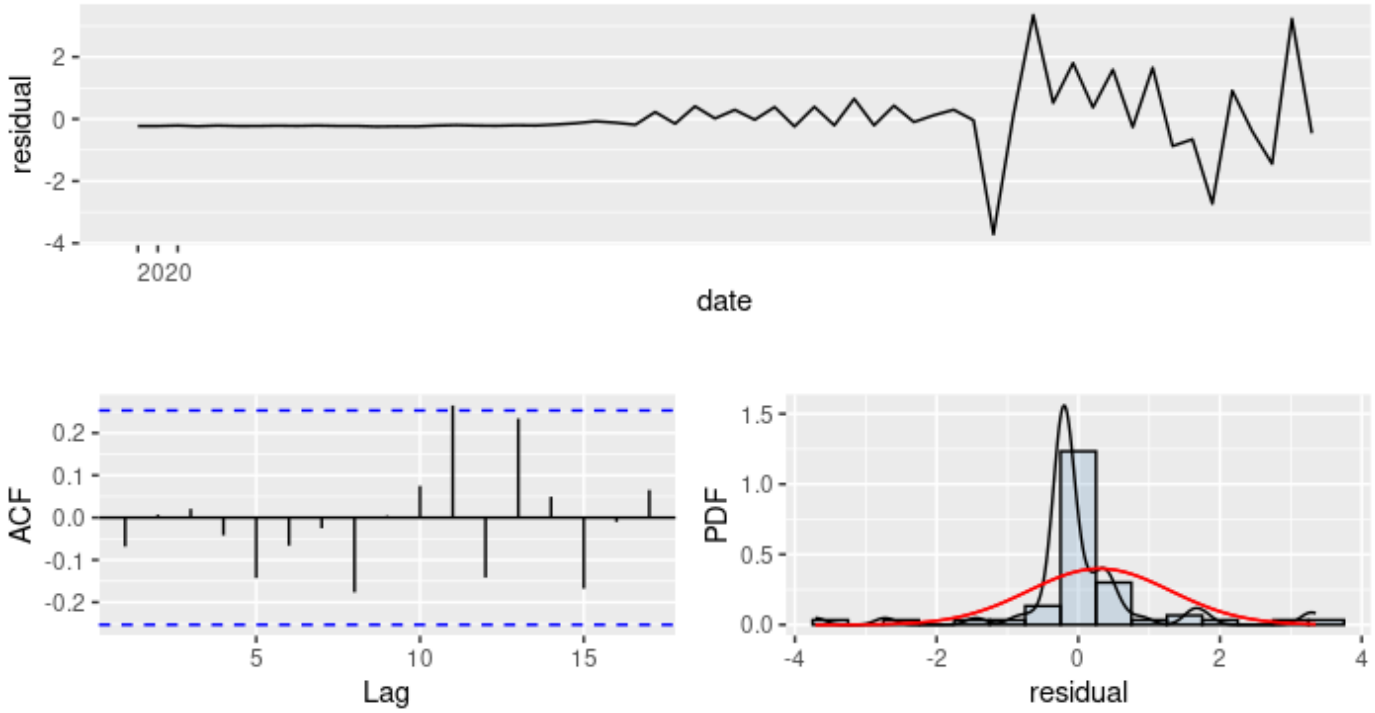

Figure 2. Plots of ARIMA(2,2,1) residuals, where the blue dashed lines indicate the statistical significance, whereas the red curve represent the normal PDF.

Table 2. Comparison of precision measures for ARIMA(p, d, q) models.

Model	AIC	AICc	BIC	RMSE	MAE	MAPE
ARIMA(2,2,2)	766.76	767.91	777.06	159.45	79.64	7.56
ARIMA(0,2,0)	800.42	800.49	802.48	232.08	109.15	9.13
ARIMA(0,2,1)	770.80	771.02	774.92	174.83	78.66	7.42
ARIMA(1,2,0)	793.85	794.07	797.97	215.25	99.23	7.88
ARIMA(2,2,1)	765.63	766.38	773.87	160.74	76.18	7.67
ARIMA(1,2,1)	772.77	773.22	778.96	174.77	78.93	7.44
ARIMA(2,2,0)	772.86	773.3	779.04	175.08	86.04	7.64
ARIMA(3,2,1)	767.26	768.41	777.56	160.22	77.83	7.62
ARIMA(3,2,0)	772.34	773.09	780.58	171.27	81.82	7.31

Now, we present a comparison between our forecasts using the ARIMA(2,2,1) model and the real data observed during the six study periods outlined previously.

In the first study period (11 May to 31 May), Figure 3 (left) shows the forecasted and observed data. The forecast predicted a value of 14,128, slightly less than the observed value of 15,474, resulting in a relative error of about 8.7%.

In the second period (1 June to 21 June), our model predicted 21,422 cases, which was greater than 19,616 cases observed, leading to a relative error of roughly 9.2%; see Figure 3 (right).

The third period (22 June to 12 July) saw our model to forecast 28,716 cases, which overestimated a number of 22,991 cases, resulting in a relative error of about 25%; see Figure 4 (left).

During the fourth period (13 July to 3 August), the model predicted 36,011 cases, which was significantly greater than 27,242 cases, leading to a relative error of around 32%; see Figure 4 (right).

In the fifth period (4 August to 24 August), our model predicted 43,305 cases, which again was a value greater than 30,737 cases observed, leading to a relative error of approximately 41%; see Figure 5 (left).

During the sixth period (25 August to 15 September), the model predicted 50,599 cases, which was considerably greater than 32,509 cases, resulting in a relative error of around 56%; see Figure 5 (right).

In all periods, 80% and 95% confidence intervals were also calculated and are shown in the corresponding figures. In summary, our ARIMA(2,2,1) model tended to overestimate the number of COVID-19 cases in each period. Future work could focus on refining this model to reduce the relative error.

We divided our forecasts into distinct periods to evaluate the performance of the ARIMA model under varying conditions over the course of the pandemic. This segmentation allowed us to analyze how the model's predictive accuracy changed in response to shifts in the pandemic's trajectory, such as during the initial outbreak, the implementation of restrictive measures, and the relaxation of these measures. The two colored confidence intervals in Figures 3–4 represent different levels of certainty in our forecasts. The darker shaded region represents a 95% confidence interval and the lighter shaded region represents an 80% confidence interval. This dual-level confidence representation allows us to communicate the inherent uncertainty of our forecasts more comprehensively.

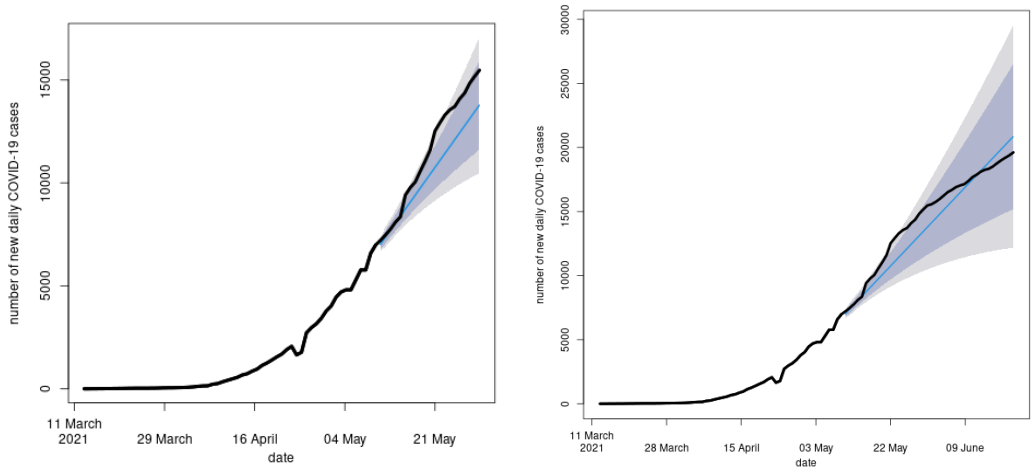

Figure 3. Forecasts of the ARIMA(2,2,1) model and number of new daily COVID-19 cases in Recife, Brazil, until the end of periods (**left**) 1 and (**right**) 2.

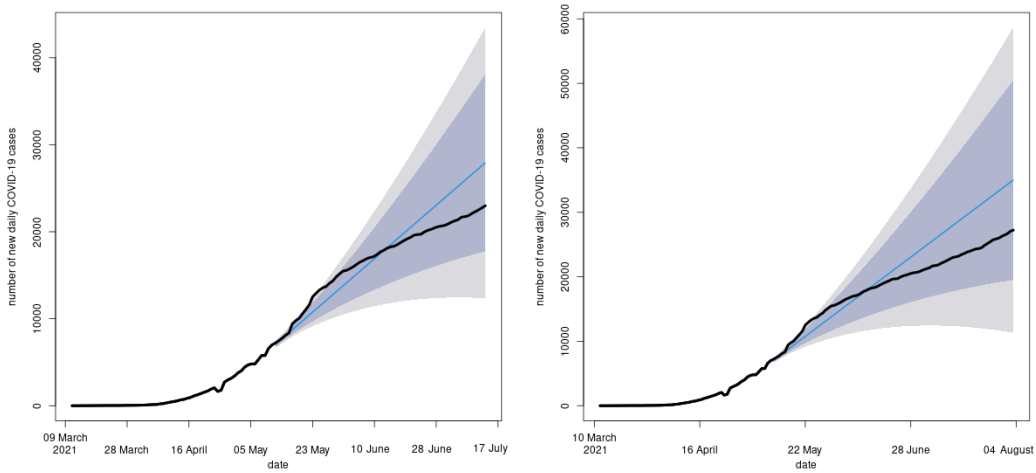

Figure 4. Forecasts of the ARIMA(2,2,1) model and observed number of new daily COVID-19 cases in Recife, Brazil, until the end of periods (**left**) 3 and (**right**) 4.

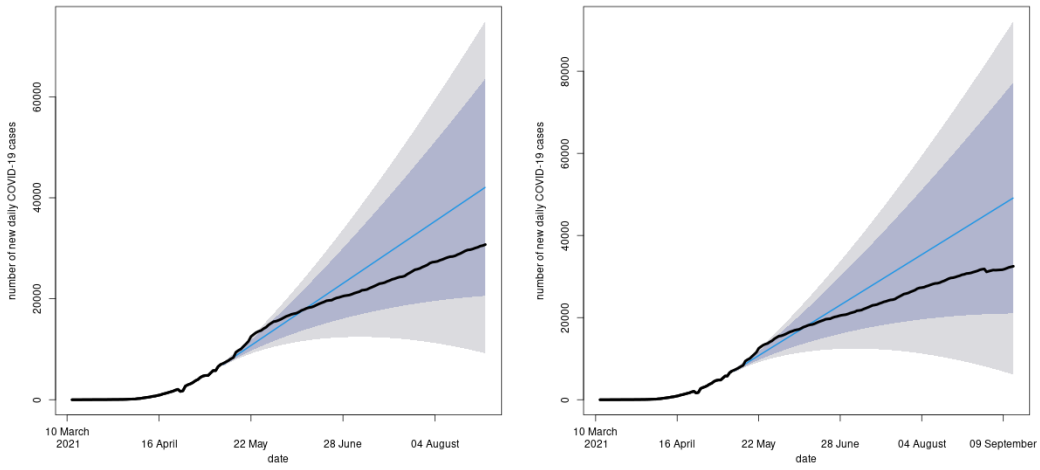

Figure 5. Forecasts of the ARIMA(2,2,1) model and observed number of new daily COVID-19 cases in Recife, Brazil, until the end of periods (**left**) 5 and (**right**) 6.

3.3. ARIMA Model: A Dual Perspective on Strengths and Weaknesses

This subsection serves as a concise mirror, reflecting both the merits and demerits of the ARIMA model. In relation to the strengths of the ARIMA model, we have that:

- It excels in the arena of short-term forecasting, as the ARIMA model offers precise prognostications for condensed time periods.
- It has a mastery in providing short-term predictions, fortifying immediate decision-making strategies and actions.
- It complements long-term forecasting mechanisms, as the ARIMA model yields valuable insights into imminent trends and near-future scenarios.

About drawbacks of the ARIMA model, we have that:

- Although it is potent in short-term forecasting, the ARIMA model may struggle to fully grasp long-term trends, especially in the presence of structural shifts in data.
- Its dependency on historical data for its predictions might limit its ability to accurately predict in situations marked by abrupt alterations or unprecedented conditions.

4. Conclusions

Our study emphasized the potential of predictive modeling in informing public health strategies during a pandemic. The COVID-19 pandemic has shown the critical importance of accurate and timely predictions in managing disease spread and mitigating its impacts. As we continue to face this unprecedented global health crisis, the further development and refinement of these predictive models will be of paramount importance.

Throughout this article, we provided an extensive review of predictive methods used for forecasting COVID-19 cases across various countries and under numerous conditions. The objective of this review was to understand the effectiveness and limitations of these methods, with a special focus on the application of ARIMA models. These models have been used to develop forecasts for many diseases. Our research evaluated the quality of these predictions by using data for confirmed cases of COVID-19 in the city of Recife, in the Brazilian state of Pernambuco, which is situated in the country's northeastern region. We showed that the prediction was always within the 80% confidence interval, except for the first analyzed period when the more restrictive measures of circulation of individuals in the city had been implemented. This caused a prediction greater than the real observed value, with a relative error of around 8.7%. However, as the end of the forecast went further into the future, the quality of the predictions fell considerably. The relative error was kept below 10% until the end of the second period of analysis, which ended 42 days after the beginning of the forecast, but reached values greater than 50% by the end of the sixth period, around four months after the analysis started. It was observed that the forecast underestimated the observed data from around the middle of the second period onwards. Therefore, this kind of modeling is effective when the predictions are for the short term. Moreover, the model was unable to detect the stabilization of the confirmed cases curve, which occurred during the study's fifth period of interest (the plateau we see in Figure 1 between May and September), or the deceleration that marked the end of the first wave of the epidemic. This is a recurrent theme when choosing ARIMA models because they would not be adequate to deal with phenomena presenting complex or dynamic characteristics.

Our study's novelty lies not in the ARIMA model itself but in its application and meticulous evaluation in the unique context of Recife city in northeastern Brazil. We illuminate the distinct challenges encountered due to sudden interventions like restrictive measures and the fluctuating phases of the pandemic. Additionally, we underlined the hurdles associated with data gathering and reporting issues, offering an original contribution to understanding the role and limitations of ARIMA models in forecasting within such intricate and dynamic environments. Our findings accentuate the necessity for continuous model scrutiny and adjustment to uphold predictive accuracy, which, in turn, can contribute towards more effective public health responses during such unstable and unforeseeable health crises.

Moving forward, it would be beneficial to explore hybrid models or models that can better account for such interventions and abrupt changes. Furthermore, continued monitoring and adaptation of these models will be essential in maintaining their predictive power and informing public health responses.

Author Contributions: Conceptualization: R.O., J.A.M.G., V.L. and C.C. data curation: R.O. and J.A.M.G. formal analysis: R.O., J.A.M.G., V.L. and C.C. investigation, R.O., J.A.M.G., V.L. and C.C. methodology: R.O., J.A.M.G., V.L. and C.C. writing—original draft: R.O., J.A.M.G. and C.C. writing—review and editing: V.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially supported by the National Council for Scientific and Technological Development (CNPq) through the grant 303192/2022-4 (R.O.), and Comissão de Aperfeiçoamento de Pessoal do Nível Superior (CAPES), from the Brazilian government; by FONDECYT, grant number 1200525 (V.L.), from the National Agency for Research and Development (ANID) of the Chilean government under the Ministry of Science and Technology, Knowledge, and Innovation; and by Portuguese funds through the CMAT—Research Centre of Mathematics of University of Minho—within projects UIDB/00013/2020 and UIDP/00013/2020 (C.C.).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data and codes used in this study are available under request.

Acknowledgments: The authors would like to thank the editors and four reviewers for their constructive comments, which led to improvement in the presentation of the manuscript.

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

References

1. Worldometers. COVID-19 Coronavirus Pandemic. 2021. Available online: https://www.worldometers.info/coronavirus/ (accessed on 8 July 2023).
2. Alkadya, W.; ElBahnasy, K.; Leiva, V.; Gad, W. Classifying COVID-19 based on amino acids encoding with machine learning algorithms. *Chemom. Intell. Lab. Syst.* **2022**, *224*, 104535. [CrossRef]
3. Ullah, A.; Malik, K.M.; Saudagar, A.K.J.; Khan, M.B.; Hasanat, M.H.A.; AlTameem, A.; Sajjad, M. COVID-19 genome sequence analysis for new variant prediction and generation. *Mathematics* **2022**, *10*, 4267. [CrossRef]
4. Alam, M.T.; Sohail, S.S.; Ubaid, S.; Ali, Z.; Hijji, M.; Saudagar, A.K.J.; Muhammad, K. It's your turn, are you ready to get vaccinated? Towards an exploration of vaccine hesitancy using sentiment analysis of Instagram posts. *Mathematics* **2022**, *10*, 4165. [CrossRef]
5. Xu, J.; Tang, Y. Bayesian framework for multi-wave COVID-19 epidemic analysis using empirical vaccination data. *Mathematics* **2021**, *10*, 21. [CrossRef]
6. Nguyen, P.H.; Tsai, J.F.; Lin, M.H.; Hu, Y.C. A hybrid model with spherical fuzzy-AHP, PLS-SEM and ANN to predict vaccination intention against COVID-19. *Mathematics* **2021**, *9*, 3075. [CrossRef]
7. Ferguson, N.; Laydon, D.; Nedjati Gilani, G.; Imai, N.; Ainslie, K.; Baguelin, M.; Perez, Z.C. *Impact of Non-Pharmaceutical Interventions (NPIs) to Reduce COVID-19 Mortality and Healthcare Demand*; Imperial College London: London, UK , 2020.
8. Brauer, F. The Kermack-McKendrick epidemic model revisited. *Math. Biosci.* **2005**, *198*, 119–131. [CrossRef]
9. Cortés-Carvajal, P.D.; Cubilla-Montilla, M.; González-Cortés, D.R. Estimation of the instantaneous reproduction number and its confidence interval for modeling the COVID-19 pandemic. *Mathematics* **2022**, *10*, 287. [CrossRef]
10. Peng, L.; Yang, W.; Zhang, D.; Zhuge, C.; Hong, L. Epidemic analysis of COVID-19 in China by dynamical modeling. *arXiv* **2020**, arXiv:2002.06563.
11. Jia, J.; Ding, J.; Liu, S.; Liao, G.; Li, J.; Duan, B.; Wang, G.; Zhang, R. Modeling the control of COVID-19: Impact of policy interventions and meteorological factors. *Electron. J. Differ. Equ.* **2020**, *23*, 1–24.
12. Castilho, C.; Gondim, J.A.M.; Marchesin, M.; Sabeti, M. Assessing the efficiency of different control strategies for the COVID-19 epidemic. *Electron. J. Differ. Equ.* **2020**, *64*, 1–17.
13. Prem, K.; Liu, Y.; Russell, T.W.; Kucharski, A.J.; Eggo, R.M.; Davies, N.; Klepac, P. The effect of control strategies to reduce social mixing on outcomes of the COVID-19 epidemic in Wuhan, China: A modelling study. *Lancet Public Health* **2020**, *5*, e261–e270. [CrossRef] [PubMed]
14. Hou, C.; Chen, J.; Zhou, Y.; Hua, L.; Yuan, J.; He, S.; Guo, Y.; Zhang, S.; Jia, Q.; Zhao, C.; et al. The effectiveness of quarantine of Wuhan city against the corona virus disease 2019 (COVID-19): A well-mixed SEIR model analysis. *J. Med. Virol.* **2020**, *92*, 841–848. [CrossRef] [PubMed]
15. Lenhart, S.; Workman, J.T. *Optimal Control Applied to Biological Models*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2007.

16. Gondim, J.A.M.; Machado, L. Optimal quarantine strategies for the COVID-19 pandemic in a population with a discrete age structure. *Chaos Solitons Fractals* **2020**, *140*, 110166. [CrossRef] [PubMed]
17. Eikenberry, S.E.; Mancuso, M.; Iboi, E.; Phan, T.; Eikenberry, K.; Kuang, Y.; Kostelich, E.; Gumel, A.B. To mask or not to mask: Modeling the potential for face mask use by the general public to curtail the COVID-19 pandemic. *Infect. Dis. Model.* **2020**, *5*, 293–308. [CrossRef]
18. Gondim, J.A.M. Preventing epidemics by wearing masks: An application to COVID-19. *Chaos Solitons Fractals* **2021**, *143*, 110599. [CrossRef]
19. Stutt, R.O.J.H.; Retkute, R.; Bradley, M.; Gilligan, C.A.; Colvin, J. A modelling framework to assess the likely effectiveness of facemasks in combination with 'lock-down' in managing the COVID-19 pandemic. *Proc. R. Soc. A* **2020**, *476*, 20200376. [CrossRef]
20. Vasconcelos, G.L.; Brum, A.A.; Almeida, F.A.G.; Macêdo, A.M.S.; Duarte-Filho, G.C.; Ospina, R. Standard and Anomalous Waves of COVID-19: A Multiple-Wave Growth Model for Epidemics. *Braz. J. Phys.* **2021**, *51*, 1867–1883. [CrossRef]
21. Vasconcelos, G.L.; Macêdo, A.M.S.; Duarte-Filho, G.C.; Brum, A.A.; Ospina, R.; Almeida, F.A.G. Power law behaviour in the saturation regime of fatality curves of the COVID-19 pandemic. *Sci. Rep.* **2021**, *11*, 4619. [CrossRef]
22. Wu, K.; Darcet, D.; Wang, Q.; Sornette, D. Generalized logistic growth modeling of the COVID-19 outbreak: Comparing the dynamics in provinces in China and in the rest of the world. *Nonlinear Dyn.* **2020**, *101*, 1561–1581. [CrossRef]
23. Pérez-Ortega, J.; Almanza-Ortega, N.N.; Torres-Poveda, K.; Martínez-González, G.; Zavala-Díaz, J.C.; Pazos-Rangel, R. Application of data science for cluster analysis of COVID-19 mortality according to sociodemographic factors at municipal level in Mexico. *Mathematics* **2022**, *10*, 2167. [CrossRef]
24. Ogundokun, R.O.; Awotunde, J.B. Machine learning prediction for COVID-19 pandemic in India. *medRxiv* **2020**, medRxiv:2020.05.20.20107847.
25. Marzouk, M.; Elshaboury, N.; Abdel-Latif, A.; Azab, S. Deep learning model for forecasting COVID-19 outbreak in Egypt. *Process. Saf. Environ. Prot.* **2021**, *153*, 363–375. [CrossRef] [PubMed]
26. Verma, H.; Mandal, S.; Gupta, A. Temporal deep learning architecture for prediction of COVID-19 cases in India. *Expert Syst. Appl.* **2022**, *195*, 116611. [CrossRef] [PubMed]
27. ArunKumar, K.E.; Kalaga, D.V.; Sai Kumar, C.M.; Chilkoor, G.; Kawaji, M.; Brenza, T.M. Forecasting the dynamics of cumulative COVID-19 cases (confirmed, recovered and deaths) for top-16 countries using statistical machine learning models: Auto-regressive integrated moving average (ARIMA) and seasonal auto-regressive integrated moving average (SARIMA). *Appl. Soft Comput.* **2021**, *103*, 107161. [PubMed]
28. Kırbaş, İ.; Sözen, A.; Tuncer, A.D.; Kazancıoğlu, F.Ş. Comparative analysis and forecasting of COVID-19 cases in various European countries with ARIMA, NARNN and LSTM approaches. *Chaos Solitons Fractals* **2020**, *138*, 110015. [CrossRef]
29. Somyanonthanakul, R.; Warin, K.; Amasiri, W.; Mairiang, K.; Mingmalairak, C.; Panichkitkosolkul, W.; Silanun, K.; Theeramunkong, T.; Nitikraipot, S.; Suebnukarn, S. Forecasting COVID-19 cases using time series modeling and association rule mining. *PLoS ONE* **2022**, *17*, e0262539. [CrossRef]
30. Zhuang, L.; Xu, A.; Wang, X.L. A prognostic driven predictive maintenance framework based on Bayesian deep learning. *Reliab. Eng. Syst. Saf.* **2023**, *234*, 109181. [CrossRef]
31. Zhao, Z.; Wu, J.; Cai, F.; Zhang, S.; Wang, Y.G. A statistical learning framework for spatial-temporal feature selection and application to air quality index forecasting. *Ecol. Indic.* **2022**, *144*, 109416. [CrossRef]
32. Luo, C.; Shen, L.; Xu, A. Modelling and estimation of system reliability under dynamic operating environments and lifetime ordering constraints. *Reliab. Eng. Syst. Saf.* **2022**, *218*, 108136. [CrossRef]
33. Shorten, C.; Khoshgoftaar, T.M.; Furht, B. Deep Learning applications for COVID-19. *J. Big Data* **2021**, *8*, 18. [CrossRef]
34. Chimmula, V.K.R.; Zhang, L. Time series forecasting of COVID-19 transmission in Canada using LSTM networks. *Chaos Solitons Fractals* **2020**, *135*, 109864. [CrossRef] [PubMed]
35. Jamshidi, M.; Roshani, S.; Daneshfar, F.; Lalbakhsh, A.; Roshani, S.; Parandin, F.; Malek, Z.; Talla, J.; Peroutka, Z.; Jamshidi, A.; et al. Hybrid Deep Learning Techniques for Predicting Complex Phenomena: A Review on COVID-19. *AI* **2022**, *3*, 416–433. [CrossRef]
36. Singh, S.; Parmar, K.S.; Kumar, J.; Makkhan, S.J.S. Development of new hybrid model of discrete wavelet decomposition and autoregressive integrated moving average (ARIMA) models in application to one month forecast the casualties cases of COVID-19. *Chaos Solitons Fractals* **2020**, *135*, 109866. [CrossRef] [PubMed]
37. Yang, Z.; Zeng, Z.; Wang, K.; Wong, S.S.; Liang, W.; Zanin, M.; He, J. Modified SEIR and AI prediction of the epidemics trend of COVID-19 in China under public health interventions. *J. Thorac. Dis.* **2020**, *12*, 165. [CrossRef] [PubMed]
38. Rangasamy, M.; Chesneau, C.; Martin-Barreiro, C.; Leiva, V. On a novel dynamics of SEIR epidemic models with a potential application to COVID-19. *Symmetry* **2022**, *14*, 1436. [CrossRef]
39. Heredia Cacha, I.; Sáinz-Pardo Díaz, J.; Castrillo, M.; López García, Á. Forecasting COVID-19 spreading through an ensemble of classical and machine learning models: Spain's case study. *Sci. Rep.* **2023**, *13*, 6750. [CrossRef]
40. Leiva, V.; Saulo, H.; Souza, R.; Aykroyd, R.G.; Vila, R. A new BISARMA time series model for forecasting mortality using weather and particulate matter data. *J. Forecast.* **2021**, *40*, 346–364. [CrossRef]
41. Jerez-Lillo, N.; Álvarez, B.L.; Gutiérrez, J.M.; Figueroa-Zúñiga, J.; Leiva, V. A statistical analysis for the epidemiological surveillance of COVID-19 in Chile. *Signa Vitae* **2022**, *18*, 19–30.

42. Ospina, R.; Leite, A.; Ferraz, C.; Magalhaes, A.; Leiva, V. Data driven tools for assessing and combating COVID-19 outbreaks based on analytics and statistical methods in Brazil. *Signa Vitae* **2022**, *18*, 18–32.
43. Yousaf, M.; Zahir, S.; Riaz, M.; Hussain, S.M.; Shah, K. Statistical analysis of forecasting COVID-19 for upcoming month in Pakistan. *Chaos, Solitons Fractals* **2020**, *138*, 109926. [CrossRef]
44. De Araújo Morais, L.R.; Da Silva Gomes, G.S. Forecasting daily COVID-19 cases in the world with a hybrid ARIMA and neural network model. *Appl. Soft Comput.* **2022**, *126*, 109315. [CrossRef] [PubMed]
45. Papastefanopoulos, V.; Linardatos, P.; Kotsiantis, S. COVID-19: A comparison of time series methods to forecast percentage of active cases per population. *Appl. Sci.* **2020**, *10*, 3880. [CrossRef]
46. Alabdulrazzaq, H.; Alenezi, M.N.; Rawajfih, Y.; Alghannam, B.A.; Al-Hassan, A.A.; Al-Anzi, F.S. On the accuracy of ARIMA based prediction of COVID-19 spread. *Results Phys.* **2021**, *27*, 104509. [CrossRef]
47. Sardar, I.; Akbar, M.A.; Leiva, V.; Alsanad, A.; Mishra, P. Machine learning and automatic ARIMA/Prophet models-based forecasting of COVID-19: Methodology, evaluation, and case study in SAARC countries. *Stoch. Environ. Res. Risk Assess.* **2023**, *37*, 345–359. [CrossRef] [PubMed]
48. Martin-Barreiro, C.; Ramirez-Figueroa, J.A.; Cabezas, X.; Leiva, V.; Galindo-Villardón, M.P. Disjoint and functional principal component analysis for infected cases and deaths due to COVID-19 in South American countries with sensor-related data. *Sensors* **2021**, *21*, 4094. [CrossRef]
49. Da Silva, C.C.; De Lima, C.L.; Da Silva, A.C.G.; Silva, E.L.; Marques, G.S.; De Araújo, L.J.B.; De Santana, M.A. COVID-19 dynamic monitoring and real-time spatio-temporal forecasting. *Front. Public Health* **2021**, *9*, 641253. [CrossRef]
50. Chakraborty, T. Ghosh, I. Real-time forecasts and risk assessment of novel coronavirus (COVID-19) cases: A data-driven analysis. *Chaos Solitons Fractals* **2020**, *135*, 109850. [CrossRef]
51. Sasikala, P.; Mary Immaculate Sheela, L. An efficient COVID-19 disease outbreak prediction using BI-SSOA-TMLPNN and ARIMA. *Int. J. Image Graph.* 2023, *in press*. [CrossRef]
52. IBGE. Estimates of Resident Population in Brazil and Federation Units with Reference Date on 1 July 2021. 2021. Available online: https://ftp.ibge.gov.br/, (accessed on 8 July 2023). (In Portuguese)
53. City Hall of Recife. Newsletters—COVID-19. 2021. Available online: https://novocoronavirus.recife.pe.gov.br/boletim/ (accessed on 8 July 2023). (In Portuguese)
54. Talabis, M.R.M.; McPherson, R.; Miyamoto, I.; Martin, J.L.; Kaye, D. Analytics defined. In *Information Security Analytics: Finding Security Insights, Patterns and Anomalies in Big Data*; Syngress Books; Elsevier: Amsterdam, The Netherlands, 2015; pp. 1–12.
55. Bustos, N.; Tello, M.; Droppelmann, G.; Garcia, N.; Feijoo, F.; Leiva, V. Machine learning techniques as an efficient alternative diagnostic tool for COVID-19 cases. *Signa Vitae* **2022**, *18*, 23–33.
56. Friedman, J.; Hastie, T.; Tibshirani, R. Regularization paths for generalized linear models via coordinate descent. *J. Stat. Softw.* **2005**, *33*, 1–22. [CrossRef]
57. Bradter, U.; Kunin, W.E.; Altringham, J.D.; Thom, T.J.; Benton, T.G. Identifying appropriate spatial scales of predictors in species distribution models with the random forest algorithm. *Methods Ecol. Evol.* **2013**, *4*, 167–174. [CrossRef]
58. Box, G.; Jenkins, G. *Time Series Analysis Forecasting and Control*; Wiley: Hoboken, NJ, USA, 2015.
59. Krispin, R. *Hands-On Time Series Analysis with R: Perform Time Series Analysis and Forecasting Using R*; Packt Publishing, Limited: Birmingham, UK, 2019.
60. Hamilton, J.D. *Time Series Analysis*; Princeton University Press: Princeton, NJ, USA, 2020.
61. R Core Team. *R: A Language and Environment for Statistical Computing*; R Foundation for Statistical Computing: Vienna, Austria, 2020.
62. Hyndman, R.J.; Khandakar, Y. Automatic time series forecasting: The forecast package for R. *J. Stat. Softw.* **2008**, *27*, 1–22. [CrossRef]
63. Brasil, I.O. Repository of Public Data Made Available in an Accessible Format. 2021. Available online: https://brasil.io/dataset/covid19/ (accessed on 8 July 2023).