© University Press 2024
ISSN 2719-6550
ISSN 2719-7417 online

"Journal of Education, Technology and Computer Science"
No. 5(35)/2024
www.eti.ur.edu.pl

Received: 25.09.2024
Accepted for printing: 11.12.2024
Published: 20.12.2024

DOI: 10.15584/jetacomps.2024.5.8

PAWEŁ DYMORA[1], **MIROSŁAW MAZUREK**[2], **PAULINA GÓRNIAK**[3]

Forecasting the Arrival of the Next Pandemic Wave – Modeling and Tools

[1] ORCID ID: 0000-0002-4473-823X, Rzeszow University of Technology, Faculty of Electrical, and Computer Engineering, Department of Complex Systems, Poland
[2] ORCID ID: 0000-0002-4366-1701, Rzeszow University of Technology, Faculty of Electrical, and Computer Engineering, Department of Complex Systems, Poland
[3] ORCID: 0009-0002-1965-5499, Rzeszow University of Technology, Faculty of Mathematics and Applied Physics, Poland

Abstract

The scope of the paper is to review the literature on data analysis and visualization in the context of the COVID-19 pandemic and to describe the different tools and methods used in this type of analysis. Examples of the use of these tools in practice and their limitations will also be presented. The paper concludes with conclusions and recommendations for the use of data analysis and visualization to better understand the COVID-19 pandemic and to predict the arrival of future pandemic waves.An important feature of the article is the possibility of a broad overview of modelling possibilities and the selection of appropriate frameworks and tools which can be used in the educational process of data analysis for students for in-depth study and prediction of trends and data, in particular of such important issues as the evolution of pandemic.

Keywords: Forecasting, Covid-19, R language, time series analysis

Introduction

The COVID-19 pandemic, which started in 2019, has changed the lives of people around the world forcing authorities and society to take many preventive measures. One of the most important challenges facing authorities and society is to anticipate the arrival of the next waves of the pandemic so that appropriate preventive measures can be taken. COVID-19 data analysis and visualization have become important tools in the fight against a pandemic, as they allow

a better understanding of the spread of the virus and the effectiveness of various prevention measures (Stubinger, 2020; Petropoulos, 2020).

Forecasting is the process of using existing knowledge and data to make a statement about an event that has not yet occurred. This can be based on past patterns and trends and the use of scientific and statistical models. Predictions tend to be more specific in their formulation because they are based on a solid foundation of evidence. An example of prediction would be predicting economic growth based on output, employment, and other macroeconomic indicators (Nielsen, 2020; Hyndman, 2008).

Forecasting, on the other hand, is the process of assessing the probability of future events based on incomplete or uncertain information. Forecasts tend to be less certain and more probabilistic, as they are based on assumptions and scenarios rather than specific data. Forecasting is often used to inform decision-making and planning because it allows a wide range of possible outcomes and their associated uncertainties to be considered. An example of forecasting would be assessing the probability of rainfall over the next few days based on a weather forecast. In this case, the forecast is based on available meteorological data such as temperature, atmospheric pressure, and humidity, but also takes into account the uncertainty associated with the unpredictability of the weather. This forecast can be useful for those planning a trip or organizing an outdoor event, as it allows them to take into account the possibility of rain and take appropriate precautions.

One of the main challenges in predicting and forecasting the next wave of COVID-19 is the dynamic nature of the virus itself. COVID-19 has proven to be highly infectious and capable of rapid mutation, making it difficult to accurately predict its future spread. Other important factors to consider include vaccination rates, potential re-infection in those already infected, and the impact of various intervention strategies such as social distancing or economic closures. Furthermore, the COVID-19 pandemic generated huge amounts of data, including case counts, hospitalizations, deaths, tests, virus mutations and health system response data. Tools such as R and Python are widely used to analyze large data sets (so-called big data), allowing this information to be processed, analyzed and visualized. This provides a more accurate understanding of epidemic trends and is crucial in the training process of students who will work as data analysts and researchers in the future and such qualifications for their professions are crucial.

Time series and data

Time series is a type of data that is related to time and that is collected at regular intervals. Examples of time series include share price data, meteorological data, demographic data, and traffic data. Time series analysis involves predicting future values based on the history of the data. To do this, various

statistical and mathematical models are used, such as linear models, ARIMA models, and neural models. These models allow the analysis of trends, seasonality, and other time series characteristics (Nielsen, 2020; Hyndman, 2008; Zagdański, Suchwałko, 2016).

One important step in time series analysis is stationarity. A time series is stationary if its statistics, such as mean and variance, are constant over time. If a time series is not stationary, it is necessary to transform it so that it is stationary before the analysis can be carried out. Another important element of time series analysis is model identification. The identification process involves finding the optimal model that best fits the data. Various methods can be used for this purpose, such as trial and error, Akaike approximation, or Bayesian approximation.

Different methods can be used to forecast the value of a time series, depending on the characteristics of the data. These methods include simple forecasting based on the mean, forecasting based on linear regression, or forecasting based on neural networks (Zagdański, Suchwałko, 2016; Dymora, Mazurek, Jucha, 2023).

The data analyzed represent cases of infection and death from COVID-19, a disease caused by the SARS-CoV-2 coronavirus. The data are for cases worldwide and cover the period from 1.01.2020 to 31.12.2022.

The publicly available dataset was downloaded from the official WHO (World Health Organization) website. Table 1 below describes the data contained in the file.

Table 1. Description of the data used in the analysis

Column name	Data tape	Description
Date_reported	Date	Date of notification to WHO
Country_code	String	ISO country code Alpha-2
Country	String	Country, territory, area
WHO_region	String	WHO regional offices
New_cases	Integer	New confirmed cases
Cumulative_cases	Integer	Total number of confirmed cases
New_deaths	Integer	New confirmed deaths
Cumaltive_deaths	Integer	Total number of confirmed deaths

	A	B	C	D	E	F	G	H
1	Date_reported	Country_code	Country	WHO_region	New_cases	Cumulative_cases	New_deaths	Cumulative_deaths
2	03.01.2020	AF	Afghanistan	EMRO	0	0	0	0
3	04.01.2020	AF	Afghanistan	EMRO	0	0	0	0
4	05.01.2020	AF	Afghanistan	EMRO	0	0	0	0
5	06.01.2020	AF	Afghanistan	EMRO	0	0	0	0
6	07.01.2020	AF	Afghanistan	EMRO	0	0	0	0
7	08.01.2020	AF	Afghanistan	EMRO	0	0	0	0
8	09.01.2020	AF	Afghanistan	EMRO	0	0	0	0
9	10.01.2020	AF	Afghanistan	EMRO	0	0	0	0
10	11.01.2020	AF	Afghanistan	EMRO	0	0	0	0
11	12.01.2020	AF	Afghanistan	EMRO	0	0	0	0
12	13.01.2020	AF	Afghanistan	EMRO	0	0	0	0
13	14.01.2020	AF	Afghanistan	EMRO	0	0	0	0

Figure 1. Preview of the first records of a data frame

Data analysis and visualization in R

To initially illustrate the behavior of the data, clear plots were made using R language packages and libraries. Libraries used in this analysis include: `library('ggplot2'); library('dplyr'); library('ggrepel'); library('tidyr'); library('shadowtext'); library('nCov2019')` (Dymora, Mazurek, Jucha, 2023; https://www.rdocumentation.org/packages/).

The world map shown in Figure 2 was generated in R shows the numbers of SARS-CoV-2 infections worldwide. The total number of cases from the first occurrence, i.e. 17.11.2019 to 31.12.2022, is more than 660 million, and the countries most affected include the United States, where the number of total cases has already exceeded 100 million, India, Brazil, France, Germany, Italy or Russia, among others. The fewest cases have been reported in African countries such as, among others: Congo, Libya, and the Democratic Republic of the Congo. However, it is worth looking at the reason for this, which is why a world map has also been drawn up with the number of tests taken, as shown in the figure below.

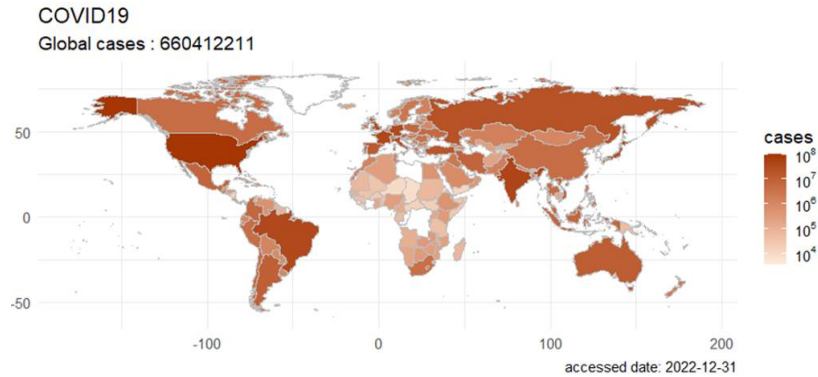

Figure 2. World map showing the number of infection cases

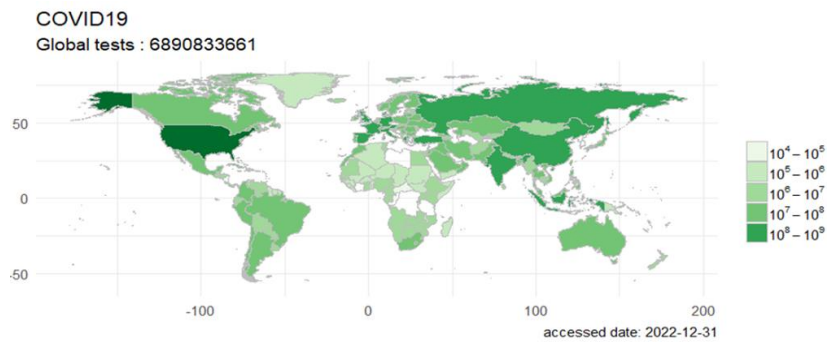

Figure 3. World map showing the number of tests performed

Figure 3 shows a world map with the number of tests performed (17.11.2019–31.12.2022). The total number of tests recorded is more than 6.8 billion, which implies that 1 in 10 people who had a COVID-19 test was positive. Similarly, however, the number of tests coincides with the number of cases detected. Therefore, here one can see the highest number of tests taken in the United States and the lowest in African countries. It can therefore be concluded that many cases have not been reported in Africa due to the negligible number of tests taken.

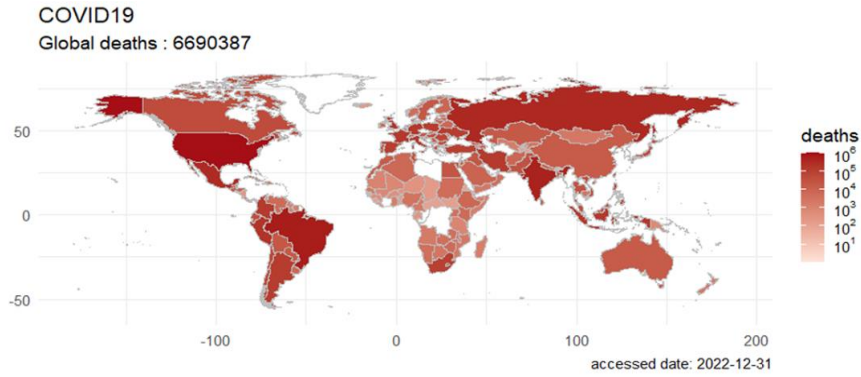

Figure 4. World map showing the number of deaths

In Figure 4, a world map shows a total number of deaths of almost 6.7 million, which, when compared to the number of cases, leads to the conclusion that 1 in about 100 people infected with SARS-CoV-2 dies from COVID-19 disease. The highest number of deaths was reported in the United States, Russia, India, and Brazil.

Time series analysis in R

Various tools and libraries for time series analysis exist in the R language, such as `'forecast', 'ts', 'timeSeries'` and `'fable'`. These tools allow for the two-rowing, visualization, and forecasting of time series data. They also allow the conversion of time series to stationary series, the identification and fitting of ARIMA models, and the determination of forecast values and their interpretation.

It was decided to present SARS-CoV-2 cases as time series. As a first step, a time series was created from the available data using the `ts()` function, which is available in the `stats` package. The series was also created in the frequency of 365 days, in order to more accurately depict the series on the graph, this setting occurs in the `frequency` parameter and `start=c()`. A time series graph was then created using the `lattice` and `ggplot` packages. Already, the seasonality of the series can be seen with an upward trend in 2020/2021 and

2021/2022. The function `xyplot()` was used to create the graphs. In Figure 5 first analysis showing the time series showing the daily number of COVID-19 infections in Poland was presented.

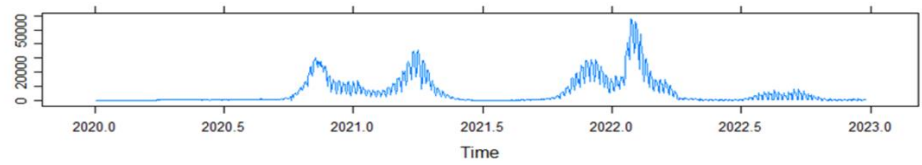

Figure 5. Time series showing the daily number of COVID-19 infections in Poland

In addition, the behavior of the data on the monthly time series is presented using the `monthplot()` (see Figure 6) and `boxplot()` functions (see Figure 7).

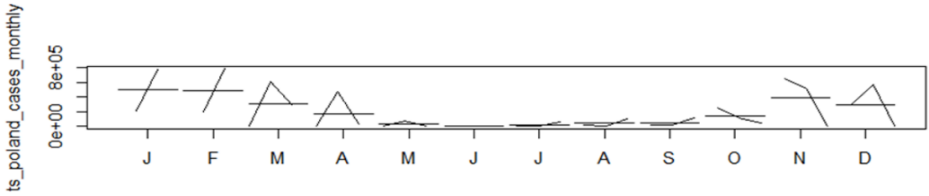

Figure 6. Monthplot chart – COVID-19 infection cases per month in Poland

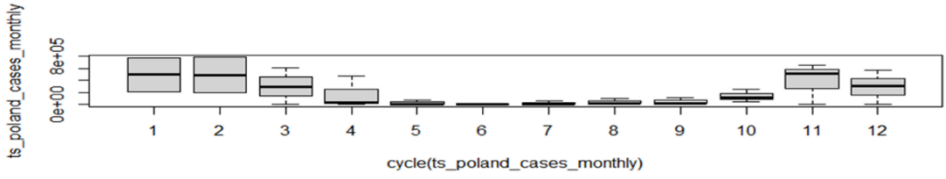

Figure 7. Boxplot chart – COVID-19 infection cases per month in Poland

Using the forecast package and the `seasonplot()` function, seasonal graphs were created by pandemic year (see. Figure 8). It can be seen that there is an increase in cases of both illness and deaths at the end of each year and a decrease at the beginning of the year (winter period).

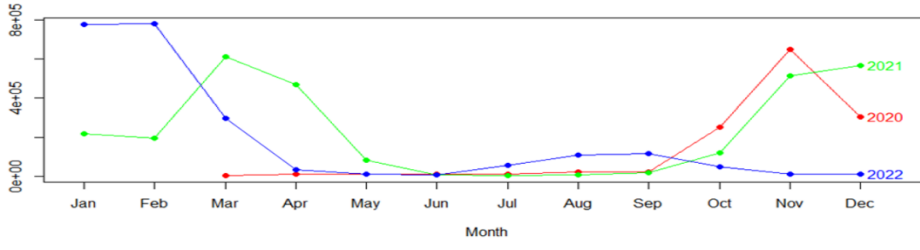

Figure 8. Seasonplot chart – infections over the years

Using the forecast package and the `tslm()` function, series plots are created, showing the trend line (blue color) and seasonality (red color), Figures 9–10. The `forecast` package implements Arima models using available functions, i.e. `Arima()` and `auto.arima()` (automatic selection of coefficients). `Tslm()` is a function in R that is used to model a time variable using linear regression. This function allows the relationship between the explanatory variable and the explanatory variables to be determined in the linear form (Dymora, Mazurek, Jucha, 2023). Analyzing the graphs in Figures 9-10, the seasonality of the data can be clearly seen; the trend overall is neither downward nor upward. Concerning deaths, the seasonality of the series is also present. On the monthly data, however, it can be seen that the trend is minimally downward.

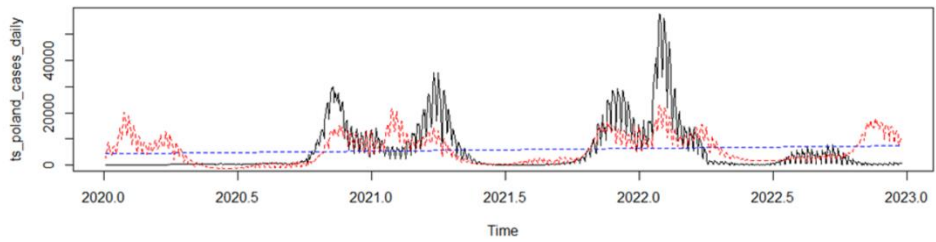

Figure 9. Graph showing seasonality and trend (daily infections) – version 1

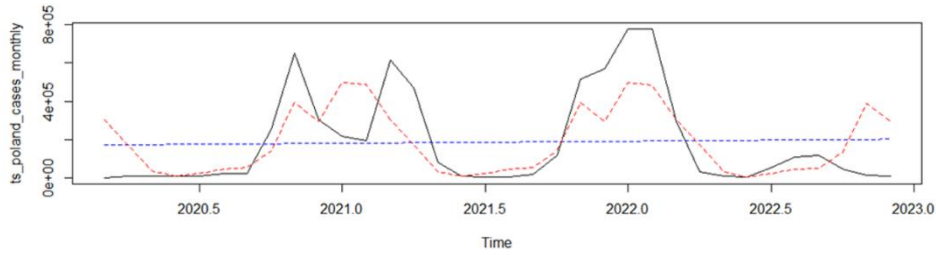

Figure 10. Graph showing seasonality and trend (daily infections) – version 2

Forecasting future values based on historical data in R

There are many different ways to forecast, depending on what the forecast is about and what data is available. Some of the popular forecasting methods are (Nielsen, 2020; Hyndman, 2008; Zagdański, Suchwałko, 2016):

1. Linear regression: this method is used when there is a linear relationship between the explanatory variable (which we want to predict) and the explanatory variables (which are used to make the prediction).
2. Multiple regression: this method is used when multiple explanatory variables are used to predict the explanatory variable.
3. Decision trees: these models are particularly useful for solving classification problems and are often used in business applications.

4. Random models: these models are used when some independent variables can affect the explanatory variable.

5. Neural networks: these models are often used to solve problems where the data are very complex and difficult to interpret using other methods.

AutoRegressive Integrated Moving Average (ARIMA) is one of the most popular time series forecasting models. It is frequently used in many fields such as business, finance and social sciences to predict future values based on past data. The ARIMA model combines three elements: autoregression (AR), differentiation (I), and moving average (MA) to provide a composite description of a time series. Autoregression is the use of previous data values to predict subsequent values. Diferentialisation is used to 'fix' the series, i.e. to remove trend and seasonality. Moving average uses the average value of the data to describe fluctuations (Nielsen, 2020; Hyndman, 2008; Zagdański, Suchwałko, 2016; Dymora *et al.*, 2023).

ARIMA coefficients are used to describe the impact of each of these three elements on the forecast. There are three ARIMA coefficients: p, d, and q. The p coefficient determines the number of previous values used for forecasting in autoregression. The coefficient d determines the number of times the series is differentiated to 'fix' it. The q coefficient determines the number of recent expected values that are used for forecasting a moving average.

The process of identifying a suitable ARIMA model involves selecting the appropriate p, d, and q coefficients that best describe the time series. This can be done using various tools and methods, such as statistical tests and visual analyses. Once a suitable model has been selected, it can be used to forecast the future values of the series

The choice of method for forecasting future values from historical data depends on a number of factors: whether trends are visible in the data, the nature of the trends, whether accuracy information is needed, and whether the computational complexity of the algorithm is important.

Arima models were used to predict future values based on historical data using the functions available in the `forecast` package, using `Arima()` and `auto.arima()` (automatic selection of coefficients). The invocation of the respective commands together with the parameters is shown in Listing 1.

Listing 1. Using `Arima()` and `auto.arima()` in R

```
# Calculation of MA coefficients
Arima(trenddiff_covid_deaths_monthly, order=c(0,0,1))
# Automatic selection of coefficients
auto.model<- auto.arima(trenddiff_covid_deaths_monthly)
summary(auto.model)
```

The most efficient model of those proposed is the one with automatically calculated parameters (`auto.arima()`). It is characterized by the lowest value of the error metric `'AIC' = 310.8`. The results obtained for the trained model are shown in Listing 2. Listing 3 presents the obtained results for both models – a comparison of the models.

Listing 2. Using `auto.arima()` results in R

```
Series: trenddiff_covid_deaths_monthly
ARIMA(0,0,1) with zero mean

Coefficients:
         ma1
      0.7970
s.e.  0.1171

sigma^2 = 638.7:  log likelihood = -153.4
AIC=310.8   AICc=311.2   BIC=313.8

Training set error measures:
                    ME     RMSE      MAE      MPE     MAPE      MASE        ACF1
Training set 0.3745432 24.8868 20.14803 120.288 201.0402 0.6003476 -0.0384035
```

Listing 3.Comparison of the Arima models

```
> model.1.0.0 <- arima(trenddiff_covid_deaths_monthly, order=c(1, 0, 0))
> model.1.0.0

Call:
arima(x = trenddiff_covid_deaths_monthly, order = c(1, 0, 0))

Coefficients:
         ar1  intercept
      0.4635     0.9997
s.e.  0.1522     8.7498

sigma^2 estimated as 763.7:  log likelihood = -156.48,  aic = 318.95
> model.0.0.1 <- arima(trenddiff_covid_deaths_monthly, order=c(0, 0, 1))
> model.0.0.1

Call:
arima(x = trenddiff_covid_deaths_monthly, order = c(0, 0, 1))

Coefficients:
         ma1  intercept
      0.7977     1.5200
s.e.  0.1168     7.6867

sigma^2 estimated as 618.6:  log likelihood = -153.38,  aic = 312.77
```

The choice of method for predicting future values from historical data depends on several factors: whether trends are visible in the data, the nature of the trends, whether accuracy information is needed, and whether the computational complexity of the algorithm is important. Other important forecasting methods have been developed are:

– Mean-based forecasting – `meanf()` function (see Figure 11),

– Naïve method – `naive()` function (see Figure 12),
– Seasonal naive method – `snaive()` function (see Figure 13),
– Drift-based forecasting – `rwf()` function (see Figure 14).

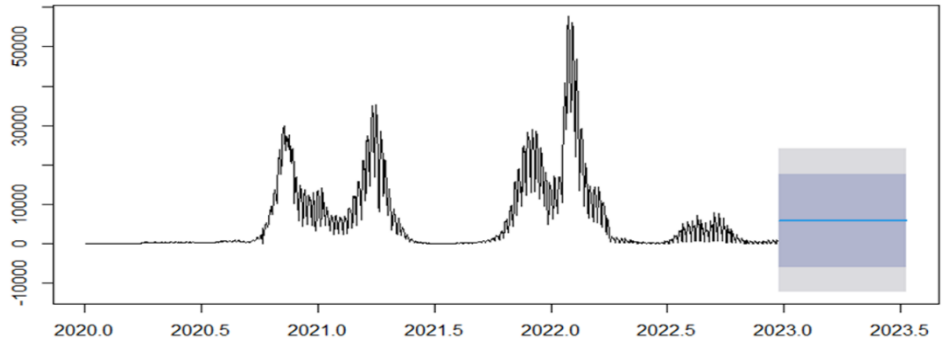

Figure 11. Mean-based forecasting

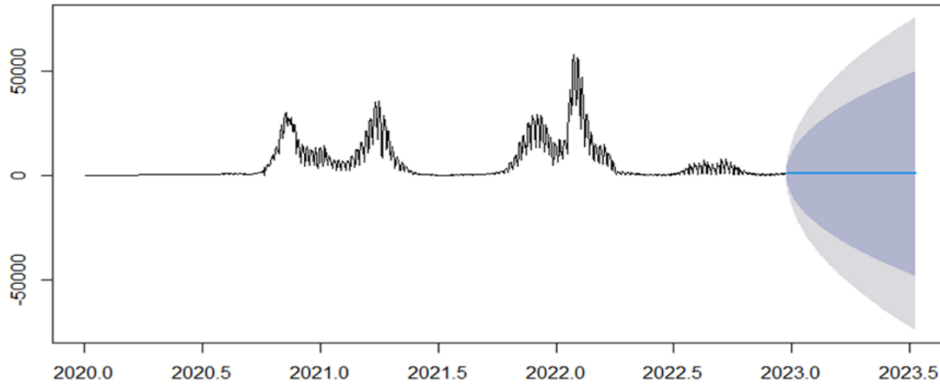

Figure 12. Naive forecasting

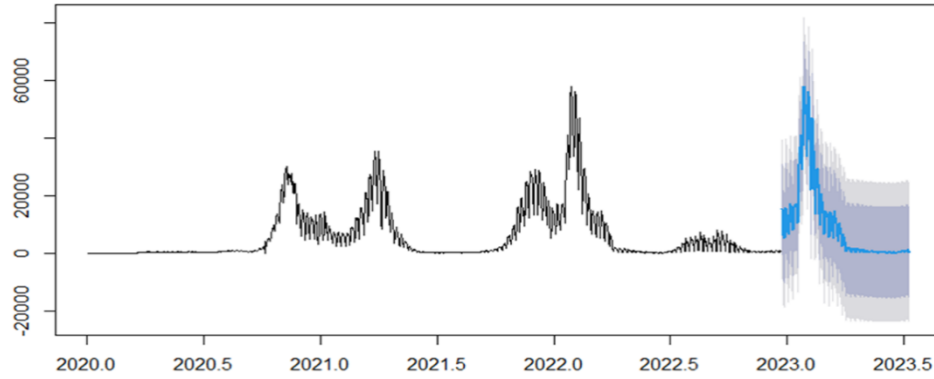

Figure 13. Seasonal naive method forecasting

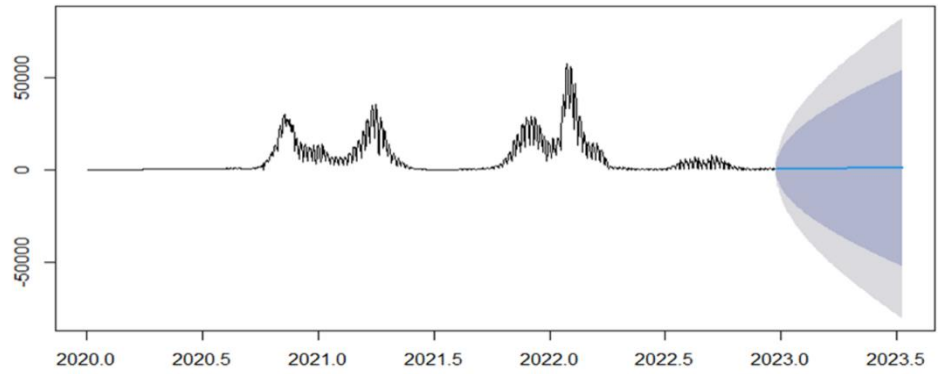

Figure 14. Drift-based forecasting

`rwf()` returns forecasts and prediction intervals for a random walk with a drift model applied to y. This is equivalent to an `ARIMA(0,1,0)` model with an optional drift coefficient. `naive()` is simply a wrapper to `rwf()` for simplicity. `snaive()` returns forecasts and prediction intervals from an `ARIMA(0,0,0)(0,1,0)m` model where m is the seasonal period. If there is no drift (as in naive), the drift parameter `c=0`. Forecast standard errors allow for uncertainty in estimating the drift parameter (unlike the corresponding forecasts obtained by fitting an ARIMA model directly).

Conclusion

The results presented in this paper can be used by governments and health services to plan and make decisions on health policy, manage resources, and prepare for future waves of pandemics. The predictions can also help with planning and coordination between different sectors such as education, transport, and industry.

An important aspect of the work carried out on COVID-19 prediction is also understanding the limitations and errors of the models and how these limitations affect the accuracy of the forecasts. The work has also required continuous adjustment and updating of the models based on new data and changing conditions.

Overall, the results of the project are important for understanding and predicting the evolution of the pandemic and helping to plan and coordinate actions to contain the spread of the virus and protect public health.However, it is important to remember that these predictions are not precise and have their limitations and that actions based on them should be considered as guidelines and not as solutions. Also, it is important to remember that the situation is dynamic and these forecasts should be constantly updated based on new data and changing conditions.

The methodologies and tools presented tools allow the building of predictive models that can predict the development of epidemics, such as waves of infections, their peaks, the simulation of the spread of a virus in a population and the effects of health policies, such as the introduction of lockdown or vaccination. This allows e.g. students to experiment with different scenarios (e.g. changes in social mobility, different vaccination strategies), providing a valuable tool for understanding pandemic dynamics. In addition, these tools enable the creation of interesting visualizations that help to communicate the results of the analyses, whether for scientific, didactic purposes, e.g. in student education, or for a wider audience.

References

Dymora, P., Mazurek, M., Jucha, M. (2023) Regression Models Evaluation of Short-Term Traffic Flow Prediction. In: W. Zamojski, J. Mazurkiewicz, J. Sugier, T. Walkowiak, J. Kacprzyk (eds.), *Dependable Computer Systems and Networks. DepCoS-RELCOMEX 2023*. Cham: Springer, 51-61.

https://www.rdocumentation.org/packages/ (1.06.2024).

Hyndman, R. (2008). *Forecasting with Exponential Smoothing: The State Space Approach*. Google Books.

Nielsen, A. (2020). *Szeregi czasowe. Praktyczna analiza i predykcja z wykorzystaniem statystyki i uczenia maszynowego*. Gliwice: Helion.

Petropoulos, F. (2020). *COVID-19: Forecasting confirmed cases and deaths with a simple time series model*. School of Management, University of Bath.

Stubinger, J. (2020). *Epidemiology of Coronavirus COVID-19: Forecasting the Future Incidence in Different Countries*. Healthcare.

Zagdański, A., Suchwałko, A. (2016) *Analiza i prognozowanie szeregów czasowych. Praktyczne wprowadzenie na podstawie środowiska R*. Warszawa: Wyd. Naukowe PWN.