FACULTY OF INFORMATION TECHNOLOGY CTU IN PRAGUE

Assignment of bachelor's thesis

Title:	Statistical modelling of Covid-19 time series
Student:	Oleh Kuznetsov
Supervisor:	Ing. Kamil Dedecius, Ph.D.
Study program:	Informatics
Branch / specialization:	Knowledge Engineering
Department:	Department of Applied Mathematics
Validity:	until the end of summer semester 2022/2023

Instructions

From the information-theoretic viewpoint, the COVID-19 pandemic is distinctive by an unprecedented amount of publically available data sets, mostly in the form of time series. The goal of this bachelor thesis is to perform statistical analyses of selected national time series. In particular:
1) study the properties of selected time series, e.g., their (non)stationarity, presence of outliers, seasonality etc.,
2) propose models that sufficiently well explain the evolution of the selected time series,
3) perform ex-post analyses of the results, evaluate predictions.
4) If possible, try to detect interesting phenomena (e.g., new waves, change points due to vaccination etc.) and connect them with government decisions or other reasons.

Electronically approved by Ing. Karel Klouda, Ph.D. on 11 February 2021 in Prague.

FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Bachelor's thesis

Statistical modeling of COVID-19 time series

Kuznetsov Oleh

Department of Applied Mathematics

Supervisor: Ing. Dedecius Kamil, Ph.D.

May 13, 2021

Acknowledgements

I express my sincere gratitude to my supervisor for this thesis Ing. Dedecius Kamil, Ph.D. It was thanks to him that I became interested in statistics and was motivated to study the topics covered in this work. The advice, guidance, and information received from him helped me deepen into this problem and achieve good results.

I am deeply grateful to my family (and especially my dear mother, father, and sister) for their support and opportunity to move to the Czech Republic and study at the Czech Technical University in Prague.

Last but not least, I would like to frankly acknowledge my friends for being with me during my studies.

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular that the Czech Technical University in Prague has the right to conclude a license agreement on the utilization of this thesis as a school work under the provisions of Article 60 (1) of the Act.

In Prague on May 13, 2021 . . . . . .

Czech Technical University in Prague

Faculty of Information Technology

Citation of this thesis

Kuznetsov, Oleh. *Statistical modeling of COVID-19 time series.* Bachelor's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2021.

Abstrakt

V letech 2020-2021 zažívá celý svět potíže způsobené globální epidemií nového viru SARS-CoV-2. Cílem této práce je statistická analýza časových řad souvisejících s pandemií COVID-19 v České republice a zkoumání využitelnosti modelů Facebook Prophet a Seasonal Autoregressive Integrated Moving Average (SARIMA) k analýze a predikci vývoje pandemických procesů. Tato práce propojuje globální změny, které nastaly ve vybraných časových řadách, s poskytnutými vládními omezeními (lockdown, nošení obličejové masky apod.).

Klíčová slova COVID-19, časové řady, prognózy časových řad, analýza časových řad, statistické modelování časových řad, Facebook Prophet, SARIMA

Abstract

In the years 2020-2021, the whole world is experiencing difficulties caused by the global epidemic of the new SARS-CoV-2 virus. This thesis aims at the statistical analysis of the time series related to the COVID-19 pandemic in the Czech Republic. It explores the usability of the Facebook Prophet and Seasonal Autoregressive Integrated Moving Average (SARIMA) models to analyze and predict the development of pandemic processes. Moreover, this thesis connects the global changes that occurred in the selected time series with the provided government restrictions (lockdown, wearing a face mask, and so on).

Keywords COVID-19, Time series, Time series forecasting, Time series analysis, Statistical time series modeling, Facebook Prophet, SARIMA

Contents

List of Figures

List of Tables

Introduction

At the beginning of 2020, the world was faced with a new disease called COVID-19. As already known, this infection is caused by the SARS-CoV-2 virus. In most cases, it occurs in the form of an acute respiratory infection. Moreover, it can cause various health complications and even death.

This virus was first detected in Wuhan, China. However, over time, it began to spread to other countries. As a result, we have witnessed a massive pandemic that has affected the entire world. Large numbers of sick people, overcrowded hospitals, and continuous government restrictive measures have become an integral part of the daily routine of countless people. Many lives depend on how the development of this outbreak will occur. Therefore, it is essential to understand this mechanism and be able to predict what will happen next [1].

Motivation

Currently, the processes related to COVID-19 can be modeled using knowledge in statistics, programming, data mining, and data analysis. A good model can help you better understand the epidemic and even predict its development. The number of new cases, deaths, people cured, the number of active cases, the reproductive index, the number of free beds in hospitals — all these are various objective metrics, data that can be used in modeling to describe the course of the pandemic.

Many experts are creating these models to improve the current situation in the world. However, in each country, the pandemic proceeds differently and the creation of models for a specific state also makes sense.

We was faced with the current world situation in the Czech Republic and everything that happened here has influenced our lives.

In light of the above, for this bachelor thesis we decided to study the modeling of processes related to the COVID-19 pandemic in this country.

Objectives

All the data that we can access can be described as a time series — a data set where all measurements are indexed with time. In 2021, there are a lot of different statistical models that can use this type of data to describe long-term processes. And for this thesis, we decided to select models named *The Facebook Prophet* and *Seasonal Autoregressive Integrated Moving Average* (SARIMA).

The Facebook Prophet is used to decompose the time series into specific components. It is also suitable for the detection of some points in time, after which the course of the pandemic was changed (changepoints). SARIMA was selected mainly for a comparison with the Facebook Prophet model. It uses a different principle based on the fact that the evolution of a time series depends on how it has developed in the past. You can find a more detailed description in the Chapter 3.

From the information-theoretic viewpoint, the COVID-19 pandemic is distinctive by an unprecedented amount of publically available data sets, mostly in the form of time series.

The goal of this bachelor thesis is to study the properties of the selected time series and use the obtained information to fit the Facebook Prophet and SARIMA models to explain the evolution of the selected time series. After that, it is necessary to detect global trend changepoints and connect them with government decisions or other reasons. Moreover, this thesis aims to study the efficiency of using data slices starting from these changepoints in the modeling of epidemic processes. The final objective is to perform ex-post analyses of the results and evaluate predictions.

Structure of the thesis

This thesis consists of 3 chapters. Chapter 1 aims to a more detailed disclosure of the theory related to time series, their properties and analysis. In Chapter 2 you can find all necessary theory behind Facebook Prophet and SARIMA models. Chapter 3 introduces practical application of selected models on COVID-19 related time series, results evaluation and discussion.

Chapter 1

Time Series. Properties and analysis

This chapter introduces the basic theory required for an understanding of time series, their properties, and the analysis approach. There also will be explained concepts like stochastic processes, mean value, variance, covariance, autocorrelation (ACF) and partial autocorrelation (PACF) functions, and time series decomposition.

All the definitions without specific source mention are common for this field of statistics and can be found in many books (Cryer et al. [2], Shumway et al. [3]).

1.1 Time series and their basic analysis

First of all, the formal definition of time series needs to be introduced.

Definition 1.1.1 (**Time series** [4])**.** *Let T be a set of time indices. The time series Y is a sequence of numerical measurements taken at time steps $t \in T$. Formally: $Y = \{Y_t \mid t \in T\}$.*

In this thesis is considered that time is discrete, all time steps with neighboring indices are equidistant, so we can say, that $t \in Z$.

1.1.1 Goals of time series analysis

To achieve the best possible results, it is necessary to establish the goals of the time series analysis. They can be formulated as:

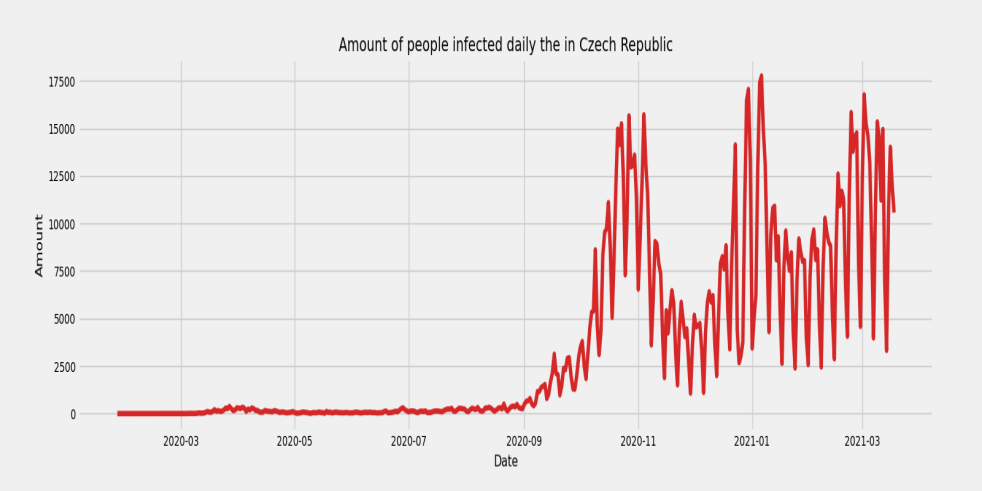


Figure 1.1: Amount of people infected daily in the Czech Republic.

- **describing the monitored process** — identifying the nature of the process represented by the selected time series.

- **forecasting** — predicting the further development of the process using historical data.

1.1.2 Real life time series illustrations

Nowadays, sequentially collected data obtained from some measurements are widespread. There are a lot of different sources of this type of information: business, meteorology, agriculture, epidemiology, and many others. In this subsection, we show some examples to give a basic idea of what such data looks like and what can be extracted from it.

All time series can be visualized as a graph. On the x-axis there are points in time, on the y-axis there are possible values of measurements taken at a given time.

For an example shown in Figure 1.1 we selected a data set that describes the number of people infected daily in the Czech Republic during the pandemic.

At first glance, we can say that this time series has seasonality and growing trend. However, there are different places where the growth rate changes, which may be caused by various cyclic processes, actions of the government, or other reasons.

1.1.3 Time series decomposition

To describe the time series demonstrated in Figure 1.2 it was necessary to use words like *"trend"*, *"seasonality"* and *"cyclic"* [4].

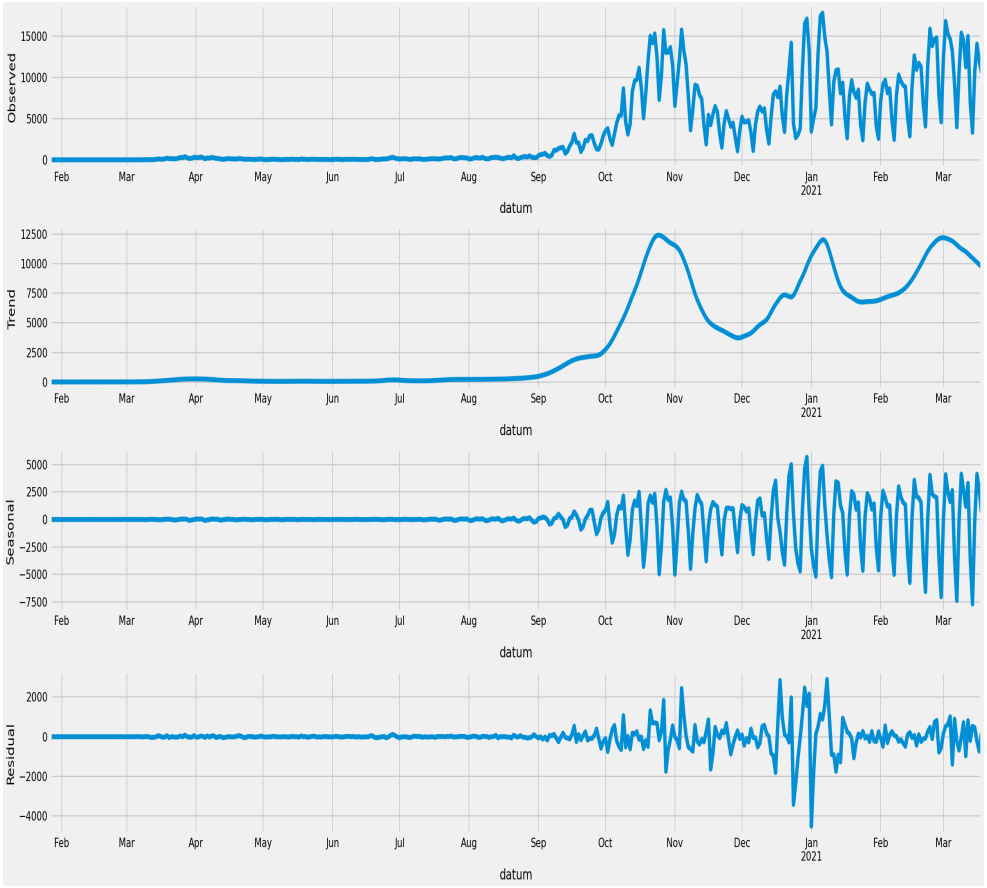


Figure 1.2: Daily number of people infected: Decomposition into trend, seasonal, and residual components.

It is important to understand that almost all time series can be decomposed into specific parts, that is why we need to introduce a bunch of new definitions.

Definition 1.1.2 (**Trend**). *The trend is the component of a time series that represents low frequency variations (long-term increase or decrease of mean value).*

Definition 1.1.3 (**Seasonal**). *The seasonal component represents periodically recurring, relatively regular and predictable (expected) development of the time series. For example, the temperature changes during the year.*

Definition 1.1.4 (**Cyclic**). *The cyclic component of a time series represents changes that do not have a specific frequency.*

Additionally, there are two main models created to describe the time series using these components (seasonal and cyclic components are denoted by one character):

- **additive**:
$$Y_t = T_t + S_t + E_t, \tag{1.1}$$
- **multiplicative**:
$$Y_t = T_t \cdot S_t \cdot E_t, \tag{1.2}$$

where Y_t is a measured value at time step t, T_t — value of the trend component at the time step t, S_t — value of the seasonal and cyclic components at the time step t, E_t — part of the process at the time step t that cannot be described by other components, also named *residual*.

It is necessary to admit that we can transform the multiplicative model into additive using properties of the logarithm:

$$\log(T_t \cdot S_t \cdot E_t) = \log T_t + \log S_t + \log E_t. \tag{1.3}$$

Figure 1.2 demonstrates a time series from the previous section and its decomposition into trend, seasonal, and residual components using the STL model [5].

1.2 Time series as a stochastic process

To perform the analysis from the statistical point of view, it is important to understand that time series can be represented as a stochastic process. Therefore, we need to give its definition and describe the basic properties.

Definition 1.2.1 (**Stochastic process** [2])**.** *Let T be a set of time indices. Then stochastic process Y is a set defined as $\{Y_t \mid t \in T\}$, where Y_t is a random variable.*

In this work, we will describe this processes in terms of mean value, variance, and covariance.

1.2.1 Mean value, Variance, Covariance

As previously stated, a stochastic process is a set of random variables. In other words, this process has a multivariate distribution and its properties depend on the distribution of each variable considered. Therefore, in this work we will specify terms that can describe this statistical distribution: mean value, variance, and covariance (first and second moments).

Definition 1.2.2 (**Mean values, Variances, Covariances** [2])**.** *Let us assume the stochastic process $Y = \{Y_t \mid t \in T\}$, where T is a set of discrete time indices. Then:*

- ***Mean value function*** *can be defined as*

$$\mu_t = E(Y_t), \quad \forall t \in T, \tag{1.4}$$

- ***Variance*** *can be defined as*

$$\sigma_t^2 = varY_t, \quad \forall t \in T, \tag{1.5}$$

- ***Autocovariance function*** *is specified as*

$$\gamma_{t_1,t_2} = cov(Y_{t_1}, Y_{t_2}), \quad \forall t_1, t_2 \in T, \tag{1.6}$$

where

$$cov(Y_{t_1}, Y_{t_2}) = E[(Y_{t_1} - \mu_{t_1})(Y_{t_2} - \mu_{t_2})]. \tag{1.7}$$

1.2.2 The Autocorrelation and Partial Autocorrelation functions

Together with the basic time series moments, we need to introduce the terms of autocorrelation and partial autocorrelation functions.

In essence, the autocorrelation of time series values denotes the linear (Pearson) correlation coefficient at different time steps. Now, it is important to introduce a more formal definition.

Definition 1.2.3 (**Autocorrelation function** [2])**.** *Let us assume atime series $Y = \{Y_t \mid t \in T\}$ with mean value μ_t and variance σ_t for each t. The autocorrelation coefficient for time steps t_1 and t_2 is given as*

$$\rho_{t_1,t_2} = corr(Y_{t_1}, Y_{t_2}) = \frac{E[(Y_{t_1} - \mu_{t_1})(Y_{t_2} - \mu_{t_2})]}{\sigma_{t_1}\sigma_{t_2}}, \quad \rho_{t_1,t_2} \in [-1, 1], \tag{1.8}$$

on condition, that mean values and variances exist and are positive.

It is necessary to understand that its value for nonneighbouring time steps is effected by values of the time series in between. That is why we need to define the partial autocorrelation function, which represents the (auto) correlation of values at timestamps t_1 and t_2 after removing the effect of the intervening variables.

Definition 1.2.4 (**Partial autocorrelation function** [2]). *Let us assume a time series $Y = \{Y_t \mid t \in T\}$. Then partial autocorrelation coefficient at lag k will be defined as:*

$$\alpha_t(k) = corr(Y_t, Y_{t-k} \mid Y_{t-1}, Y_{t-2}, \ldots, Y_{t-k+1}), \quad \alpha_t \in [-1, 1]. \tag{1.9}$$

These functions can give us an understanding of the repetitive evolution of patterns in the time series.

1.2.3 Stationarity

During the time series analysis, we might draw conclusions about its structure. Commonly, there are some reasonable simplifications and assumptions about this structure. In this thesis we will use the most important one called **stationarity**. The basic idea behind this term is that the statistical properties of the process, which generates the time series do not change over time. Specifically, we can highlight *strict* and *weak* stationary time series.

Definition 1.2.5 (**Strictly and weakly stationary time series** [6]). *Let us assume a time series $Y = \{Y_t \mid t \in T\}$, we say that this time series is:*

- ***strictly*** *stationary if joint distribution of random variables $Y_{t_1}, \ldots, Y_{t_n}$ is equal to joint distribution of $Y_{t_1+\tau}, \ldots, Y_{t_n+\tau}$ for each $t_1, \ldots, t_n$ and τ.*
- ***weakly*** *stationary if it is invariant to time shifts within distribution moments of first and second order. That means:*

  $$E(Y_t) = \mu, \tag{1.10}$$

  and

  $$cov(Y_t, Y_{t+\tau}) = \gamma(\tau), \tag{1.11}$$

  for each possible t and τ.

This thesis exploits the weekly stationarity term. Many statistical models used for time series modeling, such as ARMA (introduced in Chapter 2) assume (at least) a weekly stationary process. It is also possible to test the time series on stationarity using different statistical tests. According to Kwiatkowski et al. [7], it is possible using the combination of *Dickey-Fuller* unit root test and *Kwiatkowski–Phillips–Schmidt–Shin* (KPSS) stationarity test.

Dickey-Fuller test verifies the null hypothesis that the time series has unit roots against the alternative (no unit roots). However, sometimes it may

be not enough to exclude, for example, trend-stationarity (stationary process with a trend). KPSS test evaluates the possibility that the time series is trend-stationary (null hypothesis) against the alternative that it is nonstationary. Different combinations of the results of these tests follow the different conclusions specified in Table 1.1.

DF test result	**KPSS test result**	**Possible conclusion**
rejects null hypothesis	can not reject null hypothesis	stationarity
can not reject null hypothesis	rejects null hypothesis	nonstationarity
can not reject null hypothesis	can not reject null hypothesis	not enough data (possible trend-stationarity)
rejects null hypothesis	rejects null hypothesis	heteroscedasticity, possible structural changes

Table 1.1: Combination of the Dickey-Fuller and KPSS tests with possible conclusion.

1.2.4 Examples of basic stochastic processes

For the first example, we decided to select a process named white noise.

Definition 1.2.6 (**White noise**). *White noise is a stochastic process* $\{Y_t\}$*, where:*

$$E(Y_t) = 0, \tag{1.12}$$

$$var(Y_t) = \sigma^2 < \infty, \tag{1.13}$$

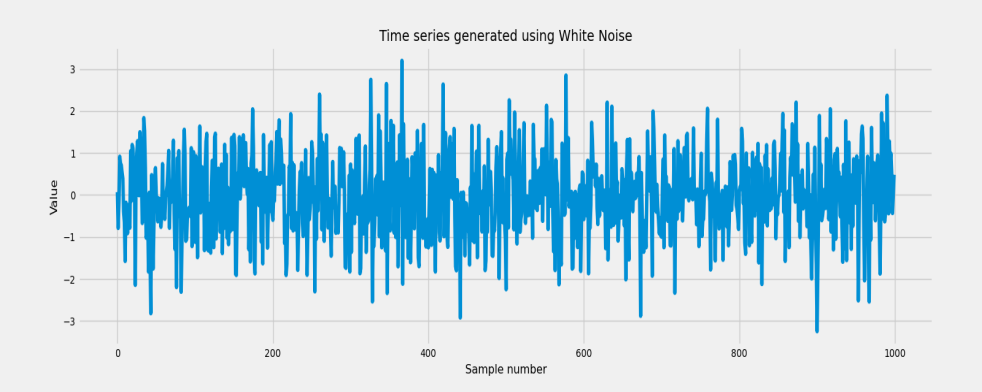
$$cov(Y_t, Y_{t+\tau}) = \gamma(\tau) = 0. \tag{1.14}$$

In other words, it is a process with a constant mean of zero, constant finite variance, and with independent and identically distributed (will be denoted as *i.i.d.*) Y_t and $Y_{t+\tau}$ for each possible t and τ. This leads to the conclusion, that this process is also *strictly stationary.*

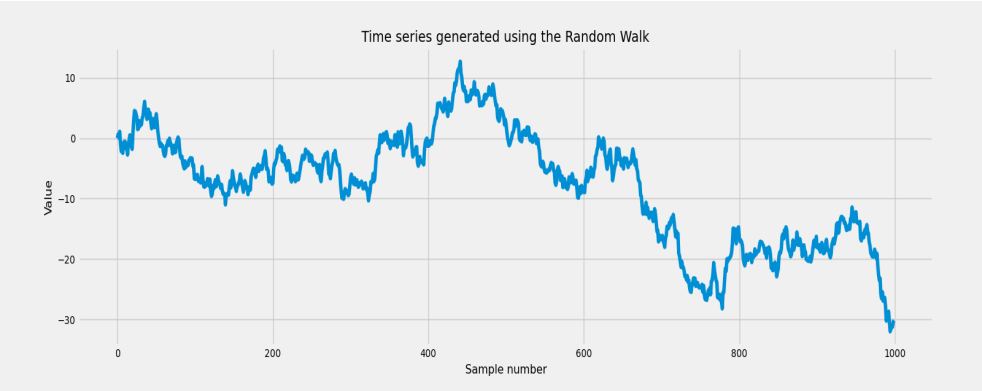
A special case of the white noise process is a normal (Gaussian) white noise, where $Y_t \sim \mathcal{N}(0, \sigma^2)$. The Figure 1.3a demonstrates an example of a time series generated using this type of process.

For the second example, we will introduce a random walk process.

Definition 1.2.7 (**Random walk**). *Let* $e_1, e_2, \ldots,$ *be a sequence of independent, identically distributed random variables with zero mean value and*



(a) Time series generated using the White noise process.



(b) Time series generated using the Random walk process.

Figure 1.3: Basic stochastic processes examples.

variance σ^2 *(White noise). Then the stochastic process* $\{Y_t \mid t \in T\}$ *will be a Random Walk if:*

$$\begin{aligned} Y_0 &= 0, \\ Y_t &= Y_{t-1} + e_t, \end{aligned} \tag{1.15}$$

can also be written as

$$Y_t = \sum_{i=1}^{t} e_i. \tag{1.16}$$

According to the attributes of the white noise and the properties of mean and variance, we can specify that Y_t has zero mean and variance $t\sigma_e^2$. Furthermore, the covariance between $Y_{t_1} = \sum_{i=1}^{t_1} e_i$ and $Y_{t_2} = \sum_{i=1}^{t_2} e_i$, where $1 \leq t_1 < t_2$ is

$t\sigma_e^2$. It follows from the covariance property (in the context of white noise):

$$cov(e_i, e_j) = \begin{cases} 0, & \text{if } i \neq j, \\ 1, & \text{if } i = j. \end{cases} \tag{1.17}$$

Based on the above, this process is nonstationary. In Figure 1.3b there can be found an example of a time series generated using the random walk process.

1.3 Summary

In this chapter, we have introduced the basic concepts behind the term "time series". We have formulated the definition of the time series, its decomposition and explored the representation of time series as a stochastic process. We have also described examples of basic processes that can generate time series.

Chapter 2

Theoretical description of selected statistical models

This chapter introduces a theoretical description of the two statistical models that were used in this thesis. They exploit fundamentally different principles based on which the process is modeled. We will describe the *Facebook Prophet* model (interchangeably called *the Prophet*), which is used for change-points detection. Then the *Seasonal Autoregressive Integrated Moving Average* (SARIMA) model will be covered.

To argue our choice from the real-world applications point of view, it is necessary to introduce other studies that aim at the same problem.

In Sengupta et al. [8] different machine learning models such as various types of regression, LSTM (Long short-term memory) deep learning forecasting model, ARIMA, and the Facebook Prophet were compered during modeling of the COVID-19 time series related to India. Results reported by the authors of this study suggest that the ARIMA and Prophet models achieved more significant prediction results (on average) than most alternatives.

In Wang et al. [9] authors used the Facebook Prophet logistic model for predicting the epidemic trend of COVID-19 in the whole world, Brazil, Russia, India, Peru, and Indonesia.

In Indhuja et al. [10], the Facebook Prophet model was used for modeling the explosive growth number of new cases in India.

In ArunKumar et al. [11] were achieved quite good results in forecasting the dynamics of COVID-19 cases (confirmed, active, and death) for 16 countries around the world using the SARIMA model.

In Ceylan et al. [12] the bunch of ARIMA models was used for forecasting the COVID-19 pandemic processes in Spain, Italy, and France. This study also refers to various researches made in the past that use (S)ARIMA models during the modeling of different diseases such as Malaria, Severe Acute Respiratory Syndrome, and so on.

Now we can move to the detailed description of the selected models.

2.1 Facebook Prophet model

The Facebook Prophet model was presented to the public and released in open source in 2017. The main objectives set forth by the creators are to simplify the modeling of processes whose evolution changes over time and to create a model that will be easily tuned during working on real-world projects.

This section mainly follows the explanation that can be found in the "Forecasting at Scale" article that introduces the Prophet model [13].

2.1.1 Main equation

In essence, the Prophet uses a decomposable additive time series model with three basic components: trend, seasonal, and holidays. More formally:

Definition 2.1.1 (**Prophet model main equation**). *Let us assume a time series $Y = \{Y_t \mid t \in T\}$ where Y_t is the value of observed variable Y at timestamp t. The basic equation behind the Prophet model is*

$$Y_t = g(t) + s(t) + h(t) + e(t), \tag{2.1}$$

where the variables are as follows:

- Y_t — *value of the observed variable in timestamp t.*
- $g(t)$ — *trend component (non-periodic changes over time).*
- $s(t)$ — *seasonal component (describes periodic changes over time).*
- $h(t)$ — *holidays component (effect of irregular schedules $\geq$ 1 day long).*
- $e(t)$ — *changes not covered by the model (error term).*

Now, let us focus on each component separately.

2.1.2 Trend component models

Prophet model has two options for trend modeling:

- **Piecewise linear model**
- **Saturation growth model**

2.1.2.1 Linear growth model

We can use a linear model for processes where the maximum or minimum value of the observed variable is not defined, and we do not need saturation near it. However, it is well known that simple linear models cannot describe the process with lots of slope changes.

For situations like this, we must introduce a linear piecewise model. This model assumes the constant growth rate with changepoints. It is the standard model and used to be the default option in Prophet.

To introduce the piecewise model equation, we need to formulate the definition of the vector of rate adjustments.

Definition 2.1.2 (**Vector of rate adjustments**). *Let us assume that we have a time series with observed value y which depends on time and has C changepoints at timestamps c_j, $j = 1, \ldots, C$. That can be described as a vector of rate adjustments*

$$\boldsymbol{\delta} \in R^C, \tag{2.2}$$

where δ_j is a change in rate of trend growth at timestamp c_j.

According to this definition, the growth rate at any timestamp t can be described as

$$b + \sum_j \delta_j, \quad \text{while } t > c_j, \tag{2.3}$$

where b is a base growth rate at timestamp t_0. This rate change can be represented like a vector $a(t)$ in $\{0,1\}^C$ such that:

$$a_j(t) = \begin{cases} 1, & \text{if } t \geq c_j, \\ 0, & \text{if } t < c_j. \end{cases} \tag{2.4}$$

The rate at timestamp t then is

$$b + \mathbf{a}(t)^T \boldsymbol{\delta}. \tag{2.5}$$

Figure 2.1: The cumulative number of people infected in Czech Republic time series modeled using the Facebook Prophet piecewise linear model with normally distributed changepoints for the first 90% of the time series.

In addition to the base growth rate b, it is possible to tune the offset parameter m (the value of the midpoint, where the slope changes its direction). This parameter is used to make the function continuous and connect the endpoints of the segments between changepoints:

$$m + \mathbf{a}(t)^T \boldsymbol{\gamma}, \tag{2.6}$$

where components of the vector $\boldsymbol{\gamma}$ *are defined as* $\gamma_j = -c_j \delta_j$.

After specifying all these parts, we can formulate the final equation of the piecewise linear model.

Definition 2.1.3 (**Peicewice linear model**). *Let* $g(t)$ *be the value of trend component* g *at timestamp* t. *Then the piecewise linear model equation can be defined as (we use extra parentheses for better visibility)*

$$g(t) = (b + \mathbf{a}(t)^T \boldsymbol{\delta})t + (m + \mathbf{a}(t)^T \boldsymbol{\gamma}), \tag{2.7}$$

where the variables of the equation are:

- b — *base trend growth rate.*
- $\mathbf{a}(t)$ — *vector that specifies changepoints used for calculation of growth rate at timestamp* t.
- $\boldsymbol{\delta}$ — *vector of rate adjustments.*

- m — *offset parameter.*
- $\boldsymbol{\gamma}$ — *vector of offset adjustments.*

According to this equation, we compute the value of the trend component g at timestamp t by multiplying the amount of time passed from the beginning of a process and the growth rate at this timestamp t. After this, we add an offset parameter m corrected using the vector of offset adjustments $\boldsymbol{\gamma}$ to make this function continuous. In this thesis, the piecewise linear model will be primary, because it correctly fits almost all selected COVID-19 time series.

Figure 2.1 introduces the example of the Facebook Prophet piecewise linear model with normally distributed changepoints for the first 90% of the time series. The vertical dashed lines pass through the timestamps where the changepoints are located.

2.1.2.2 Nonlinear Saturation growth model

Sometimes, there is a time series forecasting task where it is necessary to deal with the maximum or minimum possible value. For example, at the beginning of an epidemic process, the government cannot deal with a fast increase in the number of people infected. In this case we can see explosive (non-linear), or even nearly exponential growth. In situations like this, we can use the logistic growth model, which can be written as

$$g(t) = \frac{C}{1 + e^{-k(t-m)}}, \tag{2.8}$$

where C is the carrying capacity, k is the growth rate, and m — an offset parameter. To deal with more complex situations, there are some improvements

in the nonlinear version of the Prophet model. First, the constant carrying capacity C was replaced with time-varying $C(t)$ (better for dynamic environments, where the maximum or minimum value changes over time). Second, the growth rate is changing (the same principle as in the linear model equation 2.5). However, there is a difference associated with the offset parameter m. Its correct adjustment can be written as

$$\gamma_j = \left(c_j - m - \sum_{l<j} \gamma_l\right)\left(1 - \frac{b + \sum_{l<j} \delta_l}{b + \sum_{l \leq j} \delta_l}\right). \tag{2.9}$$

After all these corrections, we can formulate the final linear growth model equation.

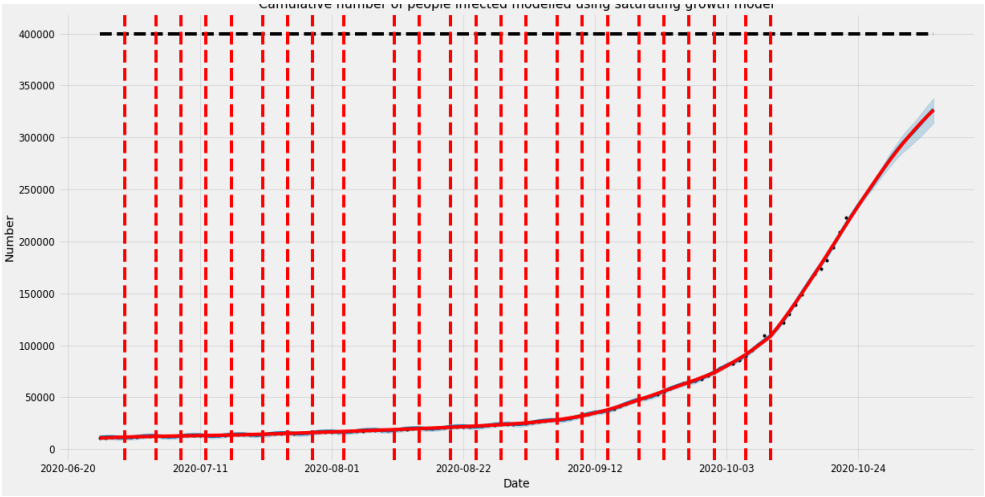


Figure 2.2: The cumulative number of people infected in Czech Republic time series (period of the explosive growth) modeled using the Facebook Prophet saturating growth model with normally distributed changepoints for the first 90% of the time series.

Definition 2.1.4 (**Logistic non-linear model**). *Let $g(t)$ be the value of trend component g at timestamp t. Then the logistic (non-linear) model equation can be defined as*

$$g(t) = \frac{C(t)}{1 + exp(-(k + \mathbf{a}(t)^T\boldsymbol{\delta})(t - (m + \mathbf{a}(t)^T\boldsymbol{\gamma})))}, \tag{2.10}$$

where the variables are as follows

- $C(t)$ — *time-varying growth rate.*
- b — *base trend growth rate.*
- $\mathbf{a}(t)$ — *vector that specifies changepoints used for calculation of growth rate at timestamp t.*
- $\boldsymbol{\delta}$ — *vector of rate adjustments.*
- m — *offset parameter.*
- $\boldsymbol{\gamma}$ — *vector of offset adjustments.*

Figure 2.2 shows the part (with exponential growth) of the time series that describes cumulative number of people infected in the Czech Republic using the Facebook Prophet saturating growth model with normally distributed changepoints for the first 90% of the time series. The capacity in the example is constant ($C(t) = 4000000$ for all t).

2.1.2.3 Automatic changepoints selection

The Prophet model supports manual changepoint selection. However, sometimes it is not available. In cases like this, for usage convenience, there is an automatic changepoint selection mechanism. It exploits a sparse prior on δ with equation 2.7 and 2.10. In practice, we might specify a large number of changepoints (e.g., one or two per month) and use the prior $\delta_j \sim Laplace(0, \tau)$.

Definition 2.1.5 (**Classical Symmetric Laplace distribution** [14])**.** *We can say that a random variable has the Classical Laplace distribution on* $(-\infty; \infty)$ *if its distribution is given by the density function*

$$f(x, \theta, \tau) = \frac{1}{2\tau} exp\Big(\frac{-|x-\theta|}{\tau}\Big), \tag{2.11}$$

where $\theta \in (-\infty; \infty)$ *is the location parameter and* $\tau > 0$ *is the scale parameter.*

The parameter τ controls the flexibility of the rate change. It has no effect on the base growth rate, so if τ goes to 0, we receive standard linear or logistic growth.

According to the official Facebook Prophet documentation [15], it specifies 25 potential normally distributed changepoints for the first 80% of the time series (by default). Subsequently, some of them are discarded by a procedure exploiting a sparse prior distribution. While fitting the trend component[1], the Prophet model minimalizes the sum of components of the vector of rate adjustments δ. In other words, it tries to specify a set of changepoints C with the biggest possible number of $\delta_j = 0$.

2.1.2.4 Trend forecast uncertainty

With just only historical data, the forecasted trend will have a constant growth rate. We need to extend the generative model forward to estimate the forecast uncertainty. In the Prophet model, we have C changepoints over a history of T time points, each of which has a rate change $\delta_j \sim Laplace(0, \tau)$. We can simulate future rate changes by replacing τ with the variance obtained from historical data. To perform this, we can use a maximum likelihood estimate of the rate scale parameter:

$$\lambda = \sum_{j=1}^{C} \delta_j. \tag{2.12}$$

[1] `https://github.com/facebook/prophet/issues/933` — explanation from a Facebook Prophet contributor on the official GitHub page

We need to save historical changepoint frequency, so future changepoints are randomly selected according to the following principle:

$$\forall j > T, \begin{cases} \delta_j = 0, & \text{with probability } \frac{T-S}{T}, \\ \delta_j \sim Laplace(0, \lambda), & \text{with probability } \frac{S}{T}. \end{cases} \tag{2.13}$$

This means an assumption that in the future we will have the same frequency and magnitude of rate changes. After receiving lambda from historical data, we use a generative model to estimate possible trends and calculate uncertainty intervals. However, this assumption is quite strong, and these intervals may have not exact coverage.

Moreover, as we already said, parameter τ controls rate change flexibility in historical data, so by increasing it, we can achieve lower training error but wider uncertainty intervals.

2.1.3 Seasonal component

A large number of time series can have multiperiod seasonality (yearly, weekly, daily, or quarterly). To handle these effects, the Prophet model contains the seasonality model, which is represented by periodic functions of t. One of the best possible options for modeling time series periodic patterns is to use the Fourier series.

Thus, let P be a constant expected time series period (for example, $P = 7$ for weekly data with the daily time scale). We can use a *standard trigonometric polynomial of order* N (in fact, it is the Fourier series [16])

$$s(t) = \sum_{n=1}^{N} \left(a_n \cos \frac{2\pi nt}{P} + b_n \sin \frac{2\pi nt}{P} \right), \tag{2.14}$$

which can approximate a wide range of seasonal patterns. There is no intercept in this equation because we already have the trend component.

According to Equation 2.14, the seasonal component fitting requires $2N$ parameters, that can be represented as a vector $\beta = [a_1, b_1, \ldots, a_N, b_N]^T$. The Prophet model does this by constructing a matrix of seasonal vectors for each timestamp t in historical and future data. For example, for weekly seasonality and $N = 5$:

$$\mathbf{X}(t) = \left[\cos \left(\frac{2\pi \cdot 1 \cdot t}{7} \right), \ldots, \sin \left(\frac{2\pi \cdot 5 \cdot t}{7} \right) \right]. \tag{2.15}$$

After defining all this, we can formulate a definition of the seasonal component of the Prophet model.

Definition 2.1.6 (**Facebook Prophet Seasonal component**). *Let us assume the Facebook Prophet model main equation* $Y_t = g(t) + s(t) + h(t) + e(t)$*, which represents the decomposed value of the observed variable* Y *at a timestamp* t*, then the seasonal component* $s(t)$ *can be defined as*

$$s(t) = X(t)\boldsymbol{\beta}, \tag{2.16}$$

where

- $\mathbf{X}(t)$ *is a matrix of seasonal vectors for each timestamp* t*.*
- $\boldsymbol{\beta} \sim \mathcal{N}(0, \sigma^2)$ *is a vector of fitting parameters that applies a smoothing prior to the seasonality.*

In practice, there are several possible values of N, and there arises the need to discriminate among them (for example, a bigger value can lead to overfitting but allows modeling of seasonal patterns that change faster). The *Akaike information criterion* (AIC) or the *Bayesian information criterion* (BIC) are two popular criteria that (loosely speaking) quantify the model quality [17].

2.1.4 Holidays component

This component is used to handle important irregular schedules that can effect forecasts.

For each holiday i, let assume D_i as a set of past and future dates of this holiday. Then, to indicate that a timestamp t takes place during holiday and to assign each holiday a parameter k_i which will indicate a corresponding change in the forecast we need to specify the matrix of regressors $\mathbf{Z}(t)$:

$$Z_i(t) = \begin{cases} 1, & \text{if } t \in D_i, \\ 0, & \text{if } t \notin D_i. \end{cases} \tag{2.17}$$

Now we can define the holidays component more formally.

Definition 2.1.7 (**Facebook Prophet Holidays component**). *Let us assume the Facebook Prophet model main equation* $Y_t = g(t) + s(t) + h(t) + e(t)$*, which represents the decomposed value of the observed variable* Y *at a timestamp* t*, then the holidays component* $h(t)$ *can be defined as*

$$h(t) = \mathbf{Z}(t) \cdot \mathbf{k}. \tag{2.18}$$

where

- $\mathbf{Z}(t)$ *is a matrix of regressors.*

- **k** *is a vector of corresponding (to selected holidays) changes in the forecast.*

In addition to the curtain holiday date, we can define a few-day window around it to handle extra weekends, and so on. Moreover, the Prophet model uses $\mathbf{k} \sim \mathcal{N}(0, \sigma^2)$ to inflict a smoothing prior on the holidays component.

2.1.5 Model fitting

Fitting the Facebook Prophet model means the estimation of different model parameters described in the previous subsections (smoothing prior, Fourier Series order, etc.). It is an optimization task, where we need to minimize the error between historical data and model [15].

2.1.6 Forecast accuracy evaluation

The Prophet model uses a time series compatible version of the cross-validation concept. It specifies the cutoff points in historical data and for each of them fits the model. Then it forecasts the following points (the amount is specified by the forecast horizon) and compares the forecasted and actual values. To measure accuracy for each forecast made during cross-validation, the Prophet can use, for example, *Mean Absolute Percentage Error* (MAPE) defined as

$$MAPE = \frac{100}{n} \sum_{t=1}^{n} \Big| \frac{Y_t - \hat{Y}_t}{Y_t} \Big|, \tag{2.19}$$

where n is an amount of forecasted points, Y_t is the real value of the observed variable Y at timestamp t, and $\hat{Y}_t$ is the forecasted value at timestamp t. This formulation is similar to the definition in Hyndman et al. [4] (Chapter 3.4, Evaluating forecast accuracy).

After calculation of the error for each forecast made during the cross-validation, the Prophet model computes the mean between all these errors.

2.1.7 Summary

The Facebook Prophet is an additive model that can be used in complex real-world tasks. It supports trend growth rate changes, multivariate seasonality, and irregular schedules. It uses complex methods for parameter estimation and forecast accuracy evaluation.

In addition to this, we found out that the Prophet model is widely used for modeling the COVID-19 pandemic processes.

All of the above properties led us to the fact that the Prophet model would be a good choice for statistical modeling of time series related to the COVID-19 pandemic.

2.2 SARIMA model

In this section, we will introduce the basic concepts behind common parametric models, created and used in practice to describe real-world processes. This category was named *Autoregressive Moving Average Models* (ARMA).

However, it is important to describe *Autoregressive* and *Moving Average* processes and models separately.

2.2.1 Autoregressive process, AR model

In essence, autoregressive processes are regressions on themselves [2]. The autoregressive process of pth-order means that value at timestamp t depends on values at p the most recent timestamps before it. For future work, we will need a formal definition for this type of a process.

Definition 2.2.1 (**Autoregressive process of order** p [3])**.** *Let us assume a stochastic process* $Y = \{Y_t \mid t \in T\}$*. This process can be named* ***the autoregressive of order*** p *if it has the following form*

$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \ldots + \phi_p Y_{t-p} + e_t, \tag{2.20}$$

where the elements of the equation are

- $\phi_1, \ldots, \phi_p$ — *autoregressive coefficients* ($\phi_p \neq 0$)*.*
- $Y_{t-1}, \ldots, Y_{t-p}$ — *most recent* p *past values.*
- $e_t \sim \mathcal{N}(0, \sigma^2)$ — *Gaussian white noise at timestamp* t*, unless otherwise stated. In fact, it is part of the* Y_t *that can not be described using* $Y_{t-1}, \ldots, Y_{t-p}$*.*

If we talk about the AR process with nonzero mean value μ*, we need to replace* Y_t *with* $Y_t - \mu$ *in Equation 2.20,*

$$Y_t - \mu = \phi_1(Y_{t-1} - \mu) + \phi_2(Y_{t-2} - \mu) + \ldots + \phi_p(Y_{t-p} - \mu) + e_t, \tag{2.21}$$

or write

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \ldots + \phi_p Y_{t-p} + e_t, \tag{2.22}$$

where $\phi_0 = \mu(1 - \phi_0 - \phi_1 - \ldots - \phi_p)$*.*

For modeling this process, we can use the autoregressive model (abbreviated as **AR**(p), where p is an order of the autoregressive process). However, this model assumes that the process is *stationary*.

2.2.1.1 Stationarity of AR process

First, we need to define the backshift operator [3], which allows a more compact and readable mathematical form of some definitions and equations that will occur in the following subsections.

Definition 2.2.2 (**Backshift operator**). *Let Y_t be a value of an observed variable Y at timestamp t. Then the **backshift operator** B is defined by*

$$BY_t = Y_{t-1},$$

and extended to powers $B^2Y_t = B(BY_t) = BY_{t-1} = Y_{t-2}$, and so on. Therefore,

$$B^kY_t = Y_{t-k}. \tag{2.23}$$

Now we can rewrite the AR model from Equation 2.20 using the backshift operator as

$$\begin{aligned} Y_t &= \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \ldots + \phi_p Y_{t-p} + e_t, \\ e_t &= Y_t - \phi_1 Y_{t-1} - \phi_2 Y_{t-2} - \ldots - \phi_p Y_{t-p} \\ &= Y_t - \phi_1 B Y_t - \phi_2 B^2 Y_t - \ldots - \phi_p B^p Y_{t-p} \\ &= (1 - \phi_1 B - \phi_2 B^2 - \ldots - \phi_p B^p) Y_t \\ &= \phi(B) Y_t, \end{aligned} \tag{2.24}$$

where $\phi(B)$ is an **autoregressive operator**. In essence, it is the **AR characteristic polynomial**, that will be introduced later in Definition 2.2.3.

This form of Equation 2.20 is widespread and used in equations related to more complex models.

Now, for understanding the stationarity of an AR process, we might also define its characteristic polynomial [2].

Definition 2.2.3 (**AR process characteristic polynomial**). *Let us assume an AR process with zero mean ($\phi_0 = 0$), $Y_t = \phi_1 Y_{t-1} + \phi_1 Y_{t-1} + \ldots + \phi_p Y_{t-p} + e_t$, then the **AR polynomial** is defined as*

$$\phi(x) = 1 - \phi_1 x - \phi_2 x^2 - \ldots - \phi_p x^p, \tag{2.25}$$

*where $\phi_1, \ldots, \phi_p$ are the regression coefficients, and $x^1, \ldots, x^p$ are the complex numbers. Therefore, the corresponding **AR characteristic equation** is*

$$1 - \phi_1 x^1 - \ldots - \phi_p x^p = 0. \tag{2.26}$$

According to the definition of the white noise process, we assume that e_t is independent from $Y_t, Y_{t-1}, \ldots, Y_{t-p}$. Thus, according to the stationarity condition of AR process (Box et al. [6]), we can now formulate the following definition.

Definition 2.2.4 (**Stationarity condition of AR(p) process**)**.** Let us assume the AR process of order p with the characteristic polynomial $\phi(x)$ and the corresponding characteristic equation $\phi(x) = 0$. This process is **stationary**

- if the roots of characteristic equation $\phi(x) = 0$ all exceeded 1 in absolute value, or in other words
- if both following conditions are satisfied:

$$\left.\begin{array}{l} \phi_1 + \phi_2 + \ldots + \phi_p < 1 \\ |\phi_p| < 1 \end{array}\right\} \tag{2.27}$$

2.2.1.2 AR process autocorrelation function

Assuming a stationary AR process with zero mean, we can now define an important recursive relationship [2] for the autocorrelation function as

$$\begin{aligned} \rho_0 &= 1, \\ \rho_{-k} &= \rho_k, \\ \rho_k &= \phi_1\rho_{k-1} + \phi_2\rho_{k-2} + \ldots + \phi_p\rho_{k-p}, \quad \text{for } k \geq 1. \end{aligned} \tag{2.28}$$

This relationship is used to compute the numerical values of ρ_k for all possible k.

2.2.1.3 Example of the AR(1) process

Figure 2.3 shows the example of AR(1) process with $\phi_1 = 0.8, Y_0 = 0.5$. It also demonstrates the related autocorrelation and partial autocorrelation functions. According to the ACF, while increasing the lag, its value decreases, which indicates that the Y_t in each timestamp t is affected by the most recent previous value. The PACF shows that we have an expressive autocorrelation at lag 1 all other lags (which are cleared from the influence of earlier lags) have values that can be considered as non-contributory.

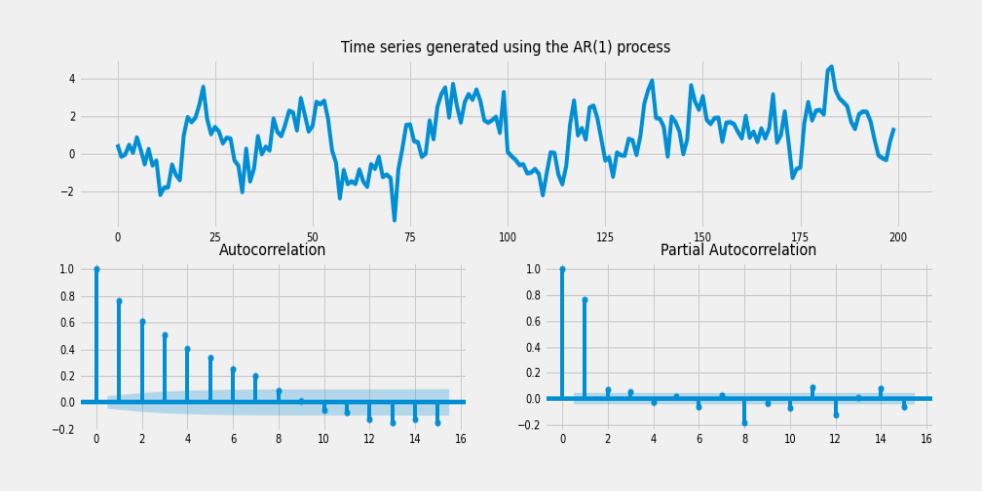


Figure 2.3: The AR(1) process example and its ACF and PACF: $\phi_1 = 0.8, Y_0 = 0.5$.

2.2.2 Moving Average process, MA model

As an alternative to an autoregressive model, where Y_t is considered as a linear combination of previous p observations, there are also exists a moving average model of order q (denoted as MA(q)) that assumes Y_t as a linear combination of the white noise at timestamp t and the white noise from q previous most recent timestamps.

Definition 2.2.5 (**Moving average process of order** q [3])**.** *Let us assume a stochastic process* $Y = \{Y_t \mid t \in T\}$*. This process can be named* ***the moving average of order*** q *if it has the following form*

$$Y_t = e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \ldots + \theta_q e_{t-q}, \tag{2.29}$$

where the elements of the equation are

- $\theta_1, \ldots, \theta_q$ — *parameters*[2] *(*$\theta_q \neq 0$*.).*
- $e_t, \ldots, e_{t-q}$ — *the Gaussian white noise with zero mean and variance* σ_e^2 *at timestamp* t *and* q *more recent timestamps before* t*, unless otherwise stated.*

The term Moving Average arises from the fact that we obtain Y_t by applying the weights $1, \theta_1, \ldots, \theta_q$ to the variables $e_t, \ldots, e_{t-q}$ and then moving them and applying to $e_{t+1}, \ldots, e_{t-q+1}$ and so on [2].

[2]Sometimes, the MA(q) model can be defined with negative coefficients: $Y_t = e_t - \theta_1 e_{t-1} - \theta_2 e_{t-2} - \ldots - \theta_q e_{t-q}$.

For a more compact representation and integration into more complex equations, we can rewrite MA(q) process in an equivalent form using the backshift operator (Equation 2.23)

$$\begin{aligned} Y_t &= e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \ldots + \theta_q e_{t-q} \\ &= e_t - + \theta_1 B e_t + \theta_2 B^2 e_t + \ldots + \theta_q B^q e_t \\ &= (1 - \theta_1 B - \theta_2 B^2 - \ldots - \theta_q B^q) e_t \\ &= \theta(B) e_t, \end{aligned} \tag{2.30}$$

where $\theta(B)$ is the **moving average operator**. In fact, $\theta(B)$ is the **MA characteristic polynomial**, that will be introduced later in Definition 2.2.6.

Towards the end of the description of the general MA process, it is necessary to admit that:

- according to Equation 2.29, the MA(q) process is **always stationary** for all values of $\theta_1, \ldots, \theta_q$,
- the autocorrelation function can be also computed using parameters $\theta_1, \ldots \theta_q$ as

$$\rho_k = \begin{cases} \frac{\theta_k + \theta_1\theta_{k+1} + \ldots + \theta_{q-k}\theta_q}{1 + \theta_1^2 + \ldots + \theta_q^2}, & \text{for } k = 1, \ldots, q, \\ 0, & \text{for } k > q. \end{cases} \tag{2.31}$$

  However, in contrast to the autocorrelation function of AR(p) processes, it is limited to lag q and can have a shape of almost anything for earlier lags [2].

2.2.2.1 Example of the MA(1) process

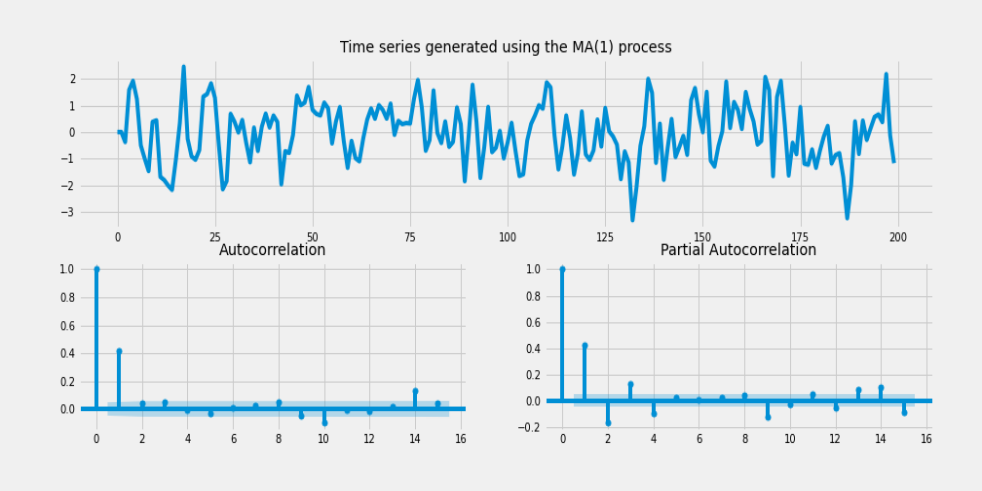
Figure 2.4 shows two examples of the MA(1) process with $\theta_1 = 0.5$ (Figure 2.4a) and $\theta_1 = 2$ (Figure 2.4b). It also demonstrates the related autocorrelation and partial autocorrelation functions.

It is visible that the ACF and PACF drop after lag 1 ($q = 1$). Moreover, we are unable to guess the curtain value of θ_1 parameter from the shape of these functions.

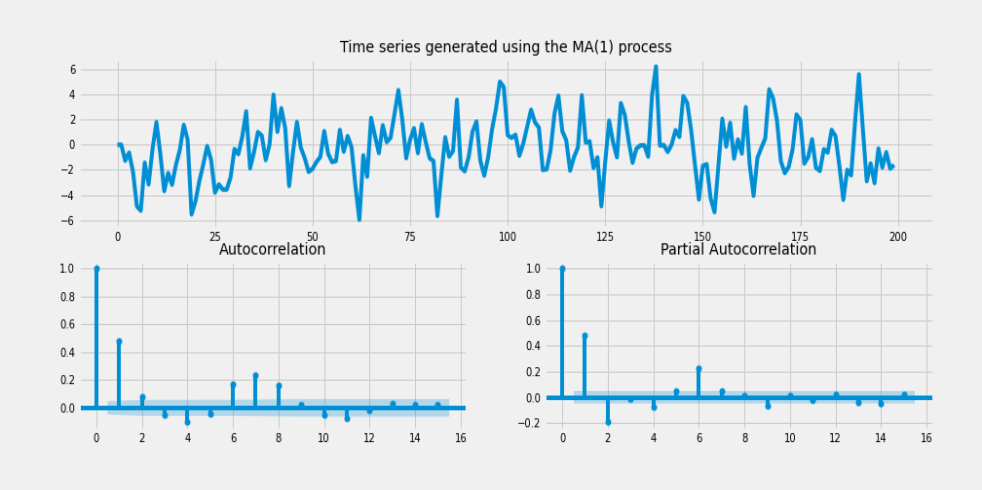
This find leads to the following property of the MA processes.

2.2.2.2 Invertibility of the MA process

According to Cryer et al. [2], some MA(q) processes can be reexpressed as an autoregression. Thus, if we consider MA(1) process (similar order as in



(a) Time series generated using the MA(1) process with $\theta_1 = 0.5$.



(b) Time series generated using the MA(1) process with $\theta_1 = 2$.

Figure 2.4: Examples of the MA(1) process with identical value of ACF and PACF at lag 1.

previous section examples)

$$Y_t = e_t + \theta_1 e_{t-1}, \tag{2.32}$$

we can rewrite it as (with the similar form of e_{t-1})

$$e_t = Y_t - \theta_1(Y_{t-1} - \theta_1 e_{t-2}) \quad = Y_t - \theta_1 Y_{t-1} + \theta_1^2 e_{t-2}. \tag{2.33}$$

If $|\theta_1| < 1$, we would not get exponential growth of Y_t dependence on previous values and could continue infinitely

$$e_t = Y_t - \theta_1 Y_{t-1} - \theta_1^2 Y_{t-2} - \dots \tag{2.34}$$

or

$$Y_t = e_t + \theta_1 Y_{t-1} + \theta_1^2 Y_{t-2} + \dots \tag{2.35}$$

It is now clear that the MA(1) model can be inverted into infinite-order autoregressive model if $|\theta_1| < 1$ and can be named **invertible**.

Before formulating the definition of the general MA(q) model invertibility, we need to define its characteristic polynomial [2].

Definition 2.2.6 (**MA characteristic polynomial**). *Let us assume an MA process, $Y_t = \phi_1 Y_{t-1} + \phi_1 Y_{t-1} + \dots + \phi_p Y_{t-p} + e_t$, then the **MA polynomial** is defined as*

$$\theta(x) = 1 - \theta_1 x - \theta_2 x^2 - \dots - \theta_q x^q, \tag{2.36}$$

*where $\theta_1, \dots, \theta_p$ are the regression coefficients, and $x^1, \dots, x^p$ are the complex numbers. Therefore, the corresponding **MA characteristic equation** is*

$$1 - \theta_1 x^1 - \dots - \theta_p x^p = 0. \tag{2.37}$$

Now we can introduce the following definition.

Definition 2.2.7 (**Invertibility of MA(q) model** [2]). *Let us assume an MA process, $Y_t = \phi_1 Y_{t-1} + \phi_1 Y_{t-1} + \dots + \phi_p Y_{t-p} + e_t$ and the model MA(q) that describes this process. This model is **invertible**, if the are coefficients π_j such that*

$$Y_t = \pi_1 Y_{t-1} + \pi_2 Y_{t-2} + \dots + e_t, \tag{2.38}$$

*which is possible **if and only if** the roots of the corresponding characteristic equation $\theta(x) = 0$ **exceeded one in absolute value.***

Therefore, if we return back to the example in previous subsection, we may now notice that even if both processes have the same ACF, only one of them (with $\theta_1 = 0.5$) is invertible.

To summarize this subsection, i would say that the **invertibility** of the MA process can be used to curtain estimation of its coefficients.

2.2.3 Autoregressive Moving Average, ARMA model

Sometimes, assuming that the process is only AR or only MA is not enough. In situations like this, we consider a mixed model named *Autoregressive Moving Average* (denoted as ARMA(p, q)) of orders p and q. This leads to the following definition.

Definition 2.2.8 (**Autoregressive Moving Average process of orders** p **and** q [3])**.** *Let us assume a stochastic process* $Y = \{Y_t \mid t \in T\}$*. This process can be named* ***the Autoregressive Moving Average of orders*** p ***and*** q *if*

$$Y_t = \phi_1 Y_{t-1} + \ldots + \phi_p Y_{t-p} + e_t + \theta_1 e_{t-1} + \ldots + \theta_q e_{t-q}, \tag{2.39}$$

where the variables of the equation are

- $\phi_1, \ldots, \phi_p$ *and* $\theta_1, \ldots, \theta_q$ *— model parameters (*$\phi_p \neq 0, \theta_q \neq 0$*).*
- $Y_{t-1}, \ldots, Y_{t-p}$ *— most recent* p *past values.*
- $e_t, \ldots, e_{t-q}$ *— the Gaussian white noise with zero mean and variance* σ_e^2 *at timestamp* t *and* q *more recent timestamps before* t*, unless otherwise stated.*

If we talk about the ARMA process with nonzero mean value μ*, we need to replace* Y_t *with* $Y_t - \mu$ *in Equation 2.39,*

$$\begin{aligned} Y_t - \mu = \phi_1(Y_{t-1} - \mu) + \ldots + \phi_p(Y_{t-p} - \mu) + \\ + e_t + \theta_1 e_{t-1} + \ldots + \theta_q e_{t-q}, \end{aligned} \tag{2.40}$$

or write

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \ldots + \phi_p Y_{t-p} + e_t + \theta_1 e_{t-1} + \ldots + \theta_q e_{t-q}, \tag{2.41}$$

where $\phi_0 = \mu(1 - \phi_0 - \phi_1 - \ldots - \phi_p)$*.*

The parameters p *and* q *are usually called the autoregressive and the moving average orders, respectively. In addition, the process introduced in Equation 2.39 can be described using the ARMA*(p, q) *model if its AR part is* ***stationary*** *(Definition 2.2.4). Moreover, for the curtain estimation of the MA coefficients, we may use the MA process* ***invertiblity*** *(Definition 2.2.7).*

It is clear that if we assume $p = 0$, then the model will be a basic moving average of order q or if we assume $q = 0$, then the model will be a basic autoregressive moving average of order p.

Exploiting the fact that ARMA(p, q) process is a mix of AR(p) and MA(q), we can rewrite Equation 2.39 in a more compact form, using the autoregressive operator introduced in Equation 2.24 and the moving average operator introduced in Equation 2.30 as

$$\phi(B)Y_t = \theta(B)e_t. \tag{2.42}$$

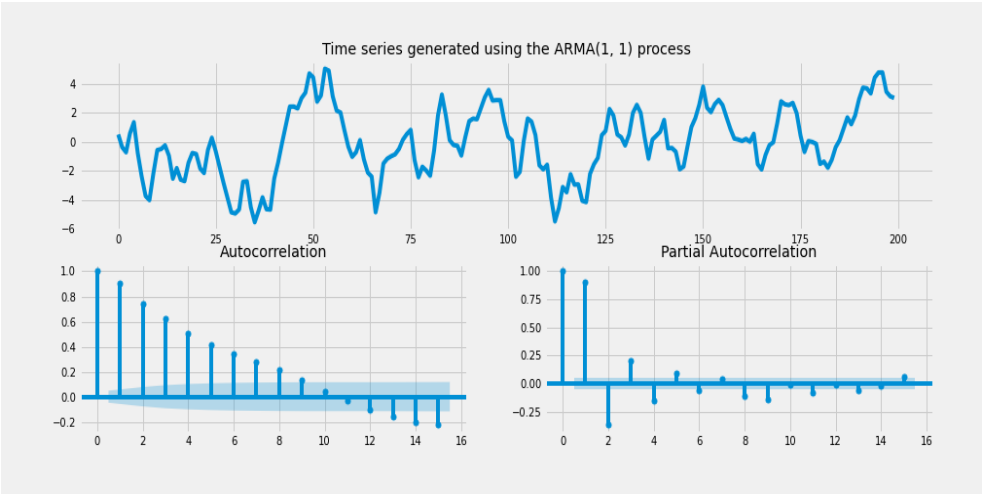


Figure 2.5: The zero mean ARMA(1, 1) process example with its ACF and PACF: $\phi_1 = 0.8, \theta_0 = 0.5, Y_0 = 0.5$.

2.2.3.1 Example of ARMA(1, 1) process

Figure 2.5 demonstrates an example of the time series generated with the ARMA(1, 1) process. The ACF and PACF indicate strong autocorrelation at lag 1.

However, If we also take to account the example of the AR(1) process in Figure 2.3, we will notice that it is hard to guess the curtain order of the mixed ARMA model. The shapes of its ACF and PACF are similar to those related to the standard AR(1) process.

2.2.4 Integrated time series: Differencing & ARIMA model

All models introduced in the previous subsections required a (weakly) stationary process. However, a large number of real-life problems consist of nonstationary time series generated by nonstationary processes. For example, the random walk process (introduced in Definition 1.2.7) has a constant mean but has no deterministic trend. It means that it can not be described using a model that assumes, for example, a stationary time series with an added time-varying mean μ.

It is clear from the definition of random walk that it is a nonstationary AR(1) process given by

$$Y_t = Y_{t-1} + e_t. \tag{2.43}$$

However, by rewriting this process using the **first difference** defined as

$$\nabla Y_t = Y_t - Y_{t-1}, \tag{2.44}$$

we will receive the following equation

$$\nabla Y_t = e_t, \tag{2.45}$$

which describes a stationary process. It is clear that this example can be easily extended to get not just the white noise.

In Cryer et al. [2] we can find an example, which assumes the process defined as

$$Y_t = M_t + X_t, \tag{2.46}$$

where M_t is an almost constant (either deterministic or stochastic) series. It is possible to approximate the value of M at timestamp t:

$$\hat{M}_t = \frac{1}{2}(Y_t + Y_{t-1}). \tag{2.47}$$

After this, we can get rid of M_t in Equation 2.46 and receive

$$Y_t - \hat{M} = Y_t - \frac{1}{2}(Y_t - Y_{t-1}) = \frac{1}{2}(Y_t - Y_{t-1}) = \frac{1}{2}\nabla Y_t, \tag{2.48}$$

which is a **constant multiple of the first difference at lag 1**.

This example assumes that M in Equation 2.46 is stochastic and its changes are driven by the random walk process (along with Y_t), such that

$$\begin{aligned} Y_t &= M_t + e_t, \\ M_t &= M_{t-1} + \varepsilon_t, \end{aligned} \tag{2.49}$$

where e_t and ε_t are independent values generated by the white noise process. Then

$$\nabla Y_t = \nabla M_t + \nabla e_t = \varepsilon_t + e_t - e_{t-1}, \tag{2.50}$$

which will have the autocorrelation function of an MA(1) process. According to Cryer et al. [2], we can study ∇Y_t from Equation 2.50 as stationary.

It is also possible to formulate assumptions which will lead to a time series that will be stationary after the second difference, and so on. This motivates the following definition.

Definition 2.2.9 (**Integrated Moving Average process ARIMA**(p, d, q) [2])**.** *Let us assume a stochastic process* $Y = \{Y_t \mid t \in T\}$*. This process can be described using* ***the Integrated Moving Average model*** *if the* d*th difference*

$$W_t = \nabla^d Y_t \tag{2.51}$$

is a ***stationary ARMA process*** *introduced in Definition 2.2.8. If* W_t *can be described using ARMA model, we can say that* Y_t *is an* ***ARIMA***(p, d, q) ***process****.*

The most common values of d that can occur in practice are 1 or 2. Therefore, we can introduce ARIMA process equation with $W_t = Y_t - Y_{t-1}$:

$$\begin{aligned} W_t = \phi_1 W_{t-1} + \phi_2 W_{t-2} + \ldots + \phi_p W_{t-p} \\ + e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \ldots + \theta_q e_{t-q}. \end{aligned} \tag{2.52}$$

It is possible to rewrite this equation in terms of the original model,

$$\begin{aligned} Y_t - Y_{t-1} = \phi_1(Y_{t-1} - Yt - 2) + \phi_2(Y_{t-2} - Yt - 3) \\ + \ldots + \phi_p(Y_{t-p} - Y_{t-p-1}) \\ + e_t + \theta_1 e_{t-1} + e_t + \theta_1 e_{t-1} + \ldots + \theta_q e_{t-q}, \end{aligned} \tag{2.53}$$

or using **the difference equation form** of the model [2] as

$$\begin{aligned} Y_t = (1+\phi_1)Y_{t-1} + (\phi_2 - \phi_1)Y_{t-2} + (\phi_3 - \phi_2)Y_{t-3} \\ + \ldots + (\phi_p - \phi_{p-1})Y_{t-p} - \phi_p Y_{t-p-1} \\ + e_t + \theta_1 e_{t-1} + e_t + \theta_1 e_{t-1} + \ldots + \theta_q e_{t-q}. \end{aligned} \tag{2.54}$$

In accordance with Equation 2.54 we now have ARMA$(p+1, q)$ process. However, the corresponding AR characteristic polynomial is

$$\begin{aligned} 1 - (1+\phi_1)x - (\phi_2 - \phi_1)x^2 - (\phi_3 - \phi_2)Y_{t-3})x^3 \\ - \ldots - (\phi_p - \phi_{p-1})x^{p\text{،}} + \phi_p x^{p+1} \\ = (1 - \phi_1 x - \phi_2 x^2 - \ldots - \phi_p x^p)(1-x), \end{aligned} \tag{2.55}$$

which shows the root $x = 1$ that applies *nonstationarity*. However, we obtain the remaining roots from the characteristic polynomial of the *stationary process* ∇Y_t.

We can also formulate the ARIMA(p, d, q) model more a more compact way using the autoregressive and moving average operators as

$$\phi(B)\nabla^d Y_t = \theta(B)e_t. \tag{2.56}$$

It is important to admit that if the process consists of only autoregressive terms, we denote the model as ARI(p, d), or if it consists of only moving average terms, it is denoted as IMA(d, q).

2.2.4.1 Example of the ARIMA(1, 1, 1) process

In Figure 2.6 we can see the time series generated with ARIMA(1, 1, 1) process (obtained using cumulative sum of the example ARMA(1, 1) process).

The ACF indicates that with lag increase, measure of the correlation decreases slowly. According to the PACF, there is a strong correlation at lag 1 (does not hints the possible order of the AR part or of the MA part).

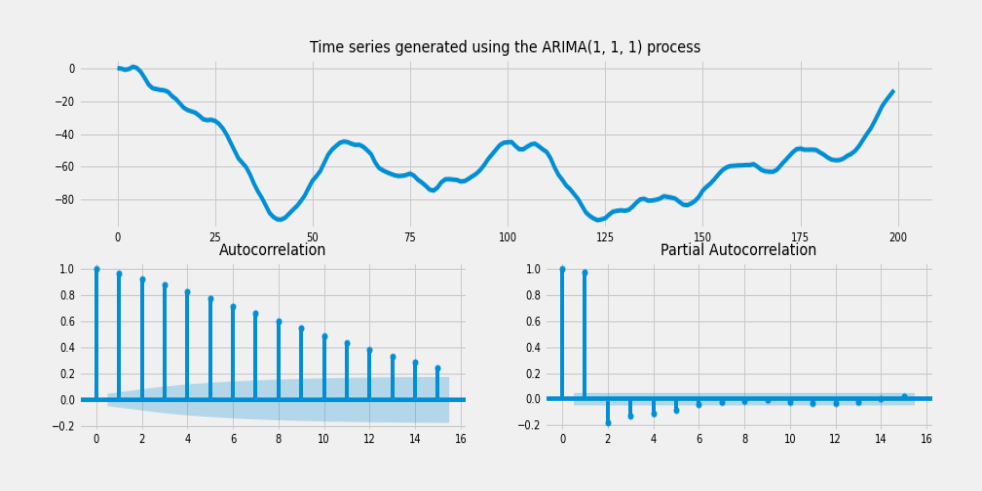


Figure 2.6: The ARIMA(1, 1, 1) process example with its ACF and PACF: $\phi_1 = 0.8, \theta_0 = 0.5, Y_0 = 0.5$.

2.2.5 Seasonality in ARIMA and ARMA processes, SARIMA model

In the previous subsections, we have introduced ARMA and ARIMA models that cover a large group of real-life time series. However, these models can not handle situations where the analyst needs to deal with (multivariate) seasonality (Definition 1.1.3). For example, we can consider a time series related to some seasonal business near the sea. During summer we will repetitively obtain a bigger amount of visitors then during other months (yearly seasonality). Or if we consider a COVID-19 related time series with information about the daily number of people infected (Figure 1.1), we would see a weekly seasonality (from Monday to Wednesday, the number reaches a peak, by Friday it gradually decreases; a sharp decrease on weekends).

These facts lead to the modification of the ARMA and ARIMA models named *Seasonal Integrated Autoregressive Moving Average* (SARIMA).

First, we need to describe the mathematical principal behind the seasonality in ARIMA model. It assumes that the value of the observed variable Y at timestamp t depends not only on the most recent previous observations but on the observations obtained at timestamps $t - s, t - 2s, \ldots$ (where s is the duration of the season). In other words, Y_t also depends on the same points of the seasonal cycle back in the history. If we consider the pure seasonality dependence of Y_t, ARMA$(P, Q)_s$ model [3] can be introduced as

$$\Phi_P(B^s)Y_t = \Theta_Q(B^s)e_t, \tag{2.57}$$

where

- $\Phi_P(B^s)$ is the **seasonal autoregressive operator of order** P and $\Theta_Q(B^s)$ is the **seasonal moving average operator of order** Q.
- s is the seasonal period.

In general, we assume that Y_t depends on a mix of seasonal and nonseasonal components. This follows a mixed ARMA$(p,q) \times (P,Q)_s$ formulated as

$$\Phi_P(B^s)\phi(B)Y_t = \Theta_Q(B^s)\theta(B)e_t. \tag{2.58}$$

Moreover, if we assume a nonstationary process and add a differencing, we will obtain the following definition.

Definition 2.2.10 (**Seasonal Integrated Autoregressive Moving Average process ARIMA$(p,d,q) \times (P,D,Q)_s$).** *Let us assume a stochastic process $Y = \{Y_t \mid t \in T\}$. This process can named as* ***Seasonal Integrated Autoregressive Moving Average*** *if it can be described using* ***Seasonal Integrated Autoregressive Moving Average model*** *[3] defined as*

$$\Phi_P(B^s)\phi(B)\nabla_s^D\nabla^d Y_t = \Theta_Q(B^s)\theta(B)e_t, \tag{2.59}$$

where the elements of the equation are:

- $\Phi_P(B^s)$ *and* $\Theta_Q(B^s)$ *— the seasonal autoregressive of operator of order P and the seasonal moving average operator of order Q, respectively.*
- $\phi(B)$ *and* $\theta(B)$ *— the ordinary autoregressive of operator and the moving average operator, respectively.*
- ∇_s^D *— seasonal difference component at lag s.*
- ∇^d *— ordinary difference component at lag 1.*
- e_t *— the Gaussian white noise with zero mean and variance σ^2.*

Definition 2.2.10 is the most important within the introduced in this section, because almost all time series considered in the practical part of this thesis can be described using this model with suitable parameters.

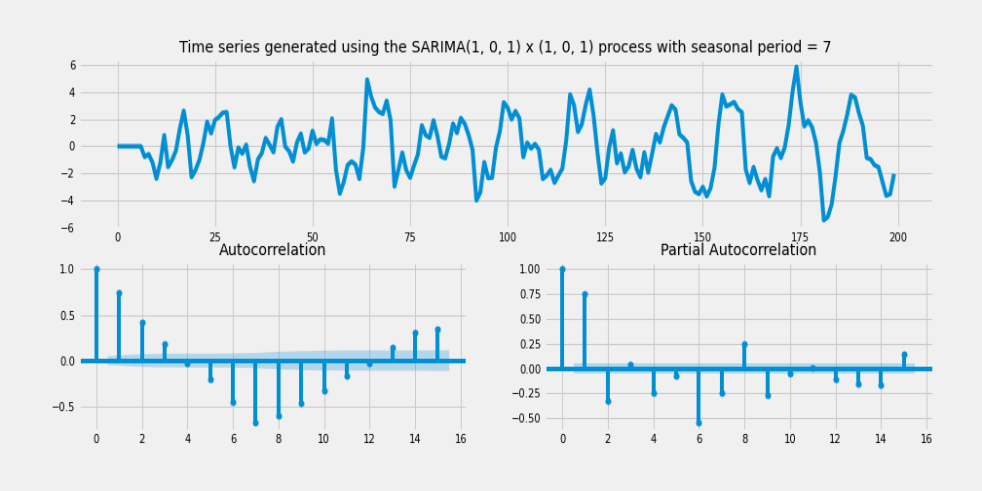


Figure 2.7: The SARIMA$(1, 0, 1) \times (1, 0, 1)_7$ process example with its ACF and PACF: $\phi_1 = 0.5, \Phi_1 = -0.4, \theta_0 = 0.4, \Theta_1 = -0.3$

2.2.5.1 Example of the seasonal ARIMA time series

Figure 2.7 demonstrates an example of the time series generated by the process denoted as SARIMA$(1, 0, 1) \times (1, 0, 1)_7$. The ACF shows the most expressive autocorrelation at lags 1 and 7. We can also see that the pattern typical for AR processes takes its place not only after lag 1 but after lag 7. The PACF demonstrates correlation at lags 1 and 6.

According to this example, we can guess the possible seasonal period of the time series. However, estimation of the orders of the AR and MA parts is not a simple task.

2.2.6 (S)ARIMA model building

The (S)ARIMA model building consists of several steps. In Shumway et al. [3] we found that it is possible to specify the following pipeline:

1. plotting the data
2. possible transforming the data
3. identifying the order of the possible model
4. parameter estimation
5. diagnostics
6. model choice

The first step is common for every data analyst task. It is necessary to visualize the original data to create a list of future actions.

The second step means that sometimes we need to apply some data transformations. For example, if the variance of data increases over time, it will need some manipulations to stabilize the process (for example, **Box-Cox transformation** or some basic such as **logarithmic** [18]).

After a suitable data transformation, it is possible to estimate the order of the model. In other words, the estimation of general p, d, q orders and seasonal P, D, Q. At the beginning of such a process, we need to estimate d and D. We can perform differencing, after which we can look at the form of the obtained time series. The ACF and PACF could also be helpful. After d and D, we can try to guess the order of the general and seasonal AR and MA components. The curtain estimation could not be possible at all, so we need to find at least approximately suitable values.

The following step implies model fitting. This can be done in a variety of ways. However, in this section we will introduce the *maximum likelihood estimation* (MLE) method [4]. In essence, we need to find parameters, which will maximize the probability of obtaining the original data with the given process. In case of (S)ARIMA it means the minimization of

$$\sum_{t=1}^{T} \varepsilon_t = Y_t - \hat{Y}_t, \tag{2.60}$$

where T is the number of timestamps in historical data, e_t is a residual (introduced, for example, in Equation 1.1 and Equation 1.2) at timestamp t, Y_t is the value of the observed variable Y at timestamp t, and $\hat{Y}_t$ is the estimated value of the Y at timestamp t. It is also possible to coordinate the estimation process using the Akaike information criterion or the Bayesian information criterion.

Finally, we need to perform *residual analysis*, that will be introduced in the practical part of this thesis. It will help us to measure the presence or absence of factors that influence our data but not described by our model (this step is also important for the Prophet model).

After doing all this, we can start forecasting using our model.

2.2.7 Forecasting using the (S)ARIMA model

In essence, the forecasting with (S)ARIMA model can be shortly explained by a short sequence of actions [4].

First, we need to rewrite the model equation for Y_{T+h}, where T is the index of the most recent timestamp in the history and $h = 1, 2, \ldots$ is the index of the forecasted timestamp. While forecasting, we need to replace future observations with their forecasts, future errors with zero, and past errors with the corresponding model residuals. The confidential intervals for the forecast are computed according to the methods used in the curtain implementation of the (S)ARIMA model.

2.2.8 Summary

During the theoretical research, we discovered that the ARIMA and SARIMA models (with suitable parameters) can be used to predict the evolution of the COVID-19 pandemic processes. Various studies and researches introduced in at the beginning of this chapter confirm that.

This find, in conjunction with the properties of this kind of statistical models, led us to the fact that it would be a strong choice for modeling pandemic processes and comparation with the Facebook Prophet model.

Chapter 3

Application of selected statistical models on COVID-19 related time series

This chapter aims at the practical application of the statistical models introduced in Chapter 2 in the context of modeling time series related to the COVID-19 pandemic in the Czech Republic.

3.1 Data

One of the most determining steps in data analysis is the searching, selection, and preparation of reliable data.

3.1.1 Data sources

The main practical goal of this thesis is to perform an analysis of time series generated by different processes related to the COVID-19 pandemic in the Czech Republic. According to this fact, the fundamental data source for this work is the official website created in association with the Ministry of Health of the Czech Republic with information about it [19].

3.1.2 Data selection

After inspection of the available data and exploring different articles that aim at the modeling of COVID-19 pandemic time series in various countries all around the world, we decided to select the time series that describe **the cumulative number of people infected**, **the cumulative number of people cured**, **the cumulative number of people dead**, and **the number of active cases**.

All selected data frames were last updated on May 4, 2021 and store measurements obtained from March 1, 2020.

3.1.3 Data preparation

All data were obtained from an official and reliable web source. Thus, there were no consistency data problems such as missing or invalid values. However, there was no time series with information about the active cases (A), so it was aggregated from time series that describe the number of people infected (I), cured (C), and dead (D) using the following formula

$$A = I - C - D. \tag{3.1}$$

After doing that, it is necessary to transform each data frame into a form compatible with the Facebook Prophet model. Table 3.1 demonstrates the time series with information about the number of people infected daily in its final form.

Cumulative number of people infected	
ds	**y**
2020-03-01	3
2020-03-02	3
2020-03-03	5
2020-03-04	6
2020-03-05	9

Table 3.1: Final form of the data frame with information about cumulative amount of people infected.

3.2 Basic analysis of the selected time series

After preparing all the selected time series, we need to perform their basic analysis to

- find some interesting phenomena, possible seasonal or cyclic patterns,
- estimate the order of future SARIMA models,
- perform basic statistical testing (stationarity tests).

Now, we can perform an analysis of each time series individually.

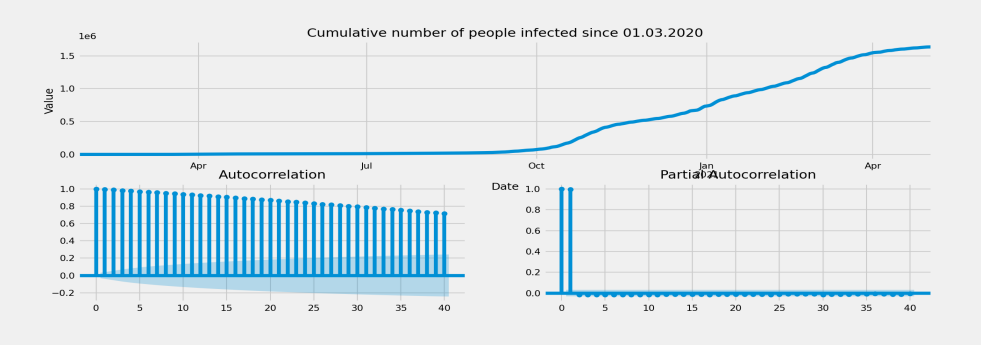


Figure 3.1: The cumulative number of people infected in Czech Republic time series with the corresponding ACF and PACF.

3.2.1 Cumulative number of people infected daily time series

First, we will inspect the cumulative number of people infected time series.

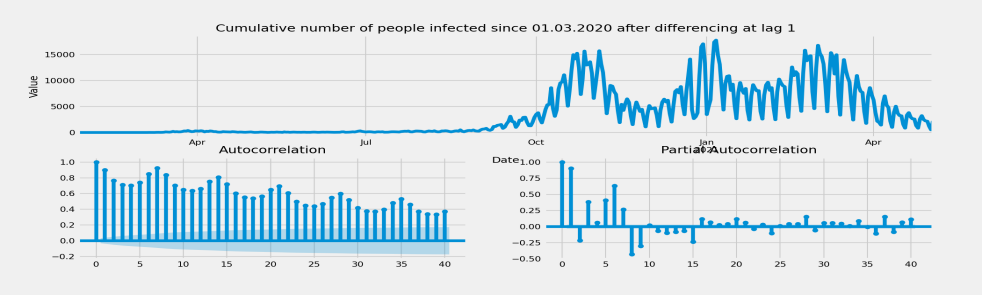
Figure 3.1 demonstrates the original time series that describes the cumulative number of people infected daily with the corresponding ACF and PACF. It is clear that this time series is a cumulative sum, so it does not give any valuable information. Thus, we need to perform differencing at lag 1.

Now we can see (Figure 3.2a) the following interesting properties of this time series:

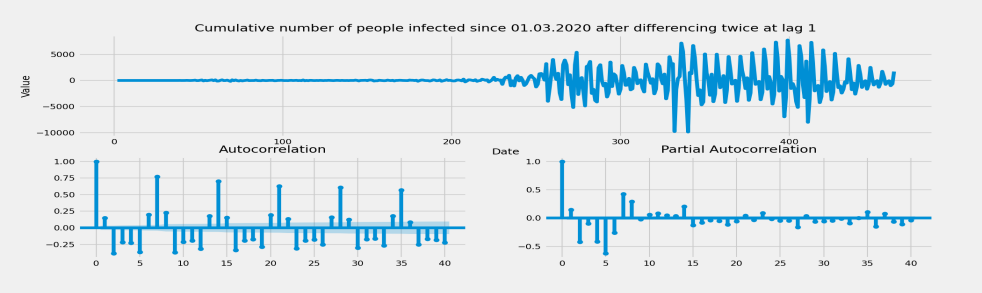
- There is no interesting activity until the beginning of June. It means that we can drop the first part of the time series during the modeling process.
- According to the ACF, there is a global trend (the correlation measure slowly decreases) and weekly seasonality (peaks at lags 7, 14, 21, and so on).
- According to the stationarity tests (Table 3.2), this is a nonstationary time series.

This leads to the fact that we need to get rid of nonstationarity. However, it is possible in two ways: one more differencing at lag 1 or seasonal differencing at lag 7.

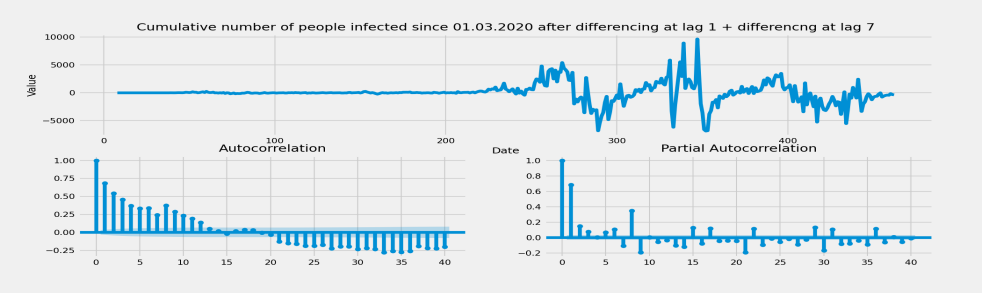
In Figure 3.2b and Figure 3.2c you can see the time series after double differencing at lag 1 and after one-time differencing at lag 1 and lag 7, respectively.



(a) The cumulative number of people infected time series after differencing at lag 1.



(b) The cumulative number of people infected time series after double differencing at lag 1.



(c) The cumulative number of people infected time series after differencing at lag 1 and 7.

Figure 3.2: The cumulative amount of people infected after different differencing sequences with the corresponding ACF and PACF.

In the first case, we got rid of the global trend, but we still have seasonality (significant correlation at lag 7, 14, and so on). Moreover, Table 3.2 indicates that the time series is now stationary. The ACF shows that the possible seasonal autoregressive order is 1.

In the second case, we got rid of the seasonality. Stationarity tests (Table 3.2) indicate that the time series is now stationary. The ACF and PACF show that the possible autoregressive order is 1.

All this information follows that the cumulative number of people infected

time series is generated by the process that can be described by SARIMA(1, 2, 0) $\times$ $(1, 0, 0)_7$ or SARIMA(1, 1, 0) $\times$ $(1, 1, 0)_7$ models.

Time series	Dikey-Fuller test p-value	KPSS test p-value
Original time series	0.993501	0.010000
1x at lag 1	0.295332	0.010000
2x at lag 1	0.001123	0.100000
1x lag 1; 1x at lag 7	0.001654	0.100000

Table 3.2: The Cumulative number of people infected: results of the stationarity tests.

3.2.2 The cumulative number of people cured time series

Figure 3.3a demonstrates the original time series that describes the cumulative number of people cured daily with the corresponding ACF and PACF. Similar to the previous time series, it is a cumulative sum. Thus, we need to perform differencing at lag 1.

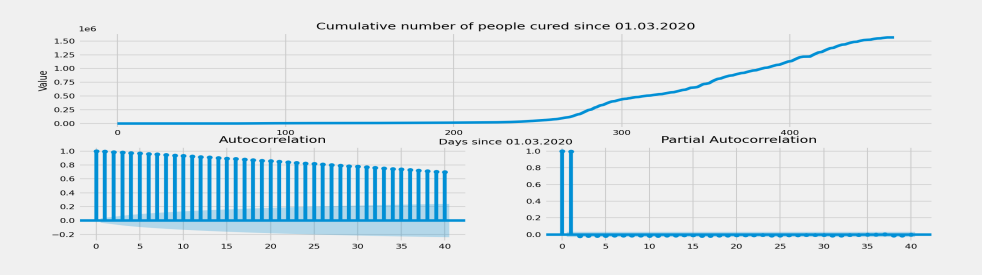
Figure 3.3b shows the time series after the first differencing. We can see the global trend and weekly seasonality. The ACF and PACF also demonstrate a slowly decreasing measure of correlation. Additionally, the ACF shows peaks at lags 7, 14, 21, and so on (seasonality). In Table 3.3 we can find out that this time series is nonstationary, and we need to do one more differencing.

Time series	Dikey-Fuller test p-value	KPSS test p-value
Original time series	0.928839	0.010000
1x at lag 1	0.120930	0.010000
2x at lag 1	0.000021	0.100000
1x lag 1; 1x at lag 7	0.000027	0.100000

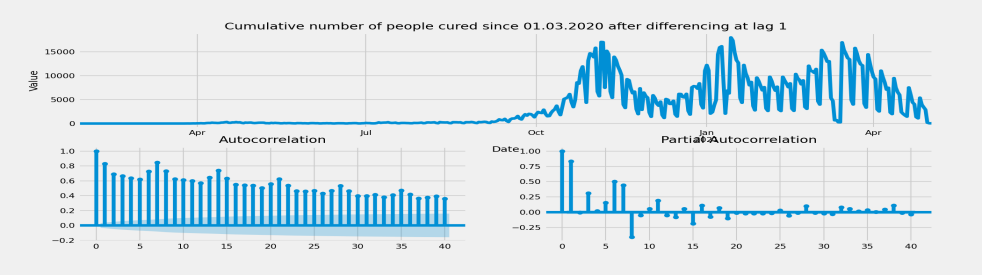
Table 3.3: The Cumulative number of people cured: results of the stationarity tests.

As in the case of the cumulative number of people infected, we can do this by one more differencing at lag 1 or lag 7 (seasonal differencing).

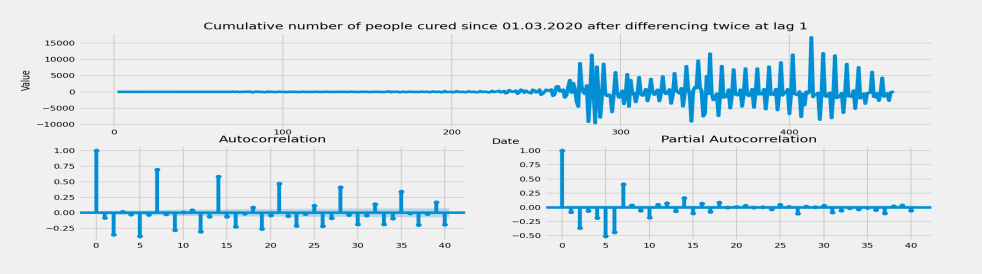
After the second differencing at lag 1 (Figure 3.3c), the time series is now stationary (Table 3.3). There is also a considerable value of the ACF at lag



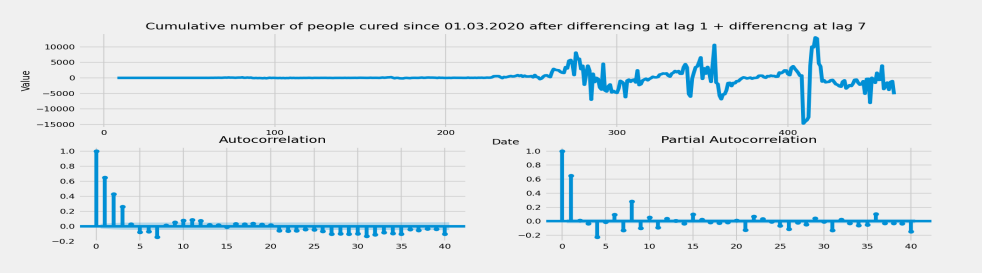
(a) The original cumulative number of people cured time series.



(b) The cumulative number of people cured time series after differencing at lag 1.



(c) The cumulative number of people cured time series after double differencing at lag 1.



(d) The cumulative number of people cured time series after differencing at lag 1 and 7.

Figure 3.3: The cumulative amount of people cured after different differencing sequences with the corresponding ACF and PACF.

7, 14, 21 (seasonality). This fact indicates that the potential order of the seasonal autoregressive part is 1.

After differencing at lag 7 (Figure 3.3d), the time series is now also stationary (Table 3.3). We got rid of seasonality. The ACF shows a fast decrease, and the PACF shows a drop after the first lag, so we can say that the order of the general autoregressive component can be equal to 1.

Based on the above, the cumulative number of people cured time series is possibly generated by the process that can be described by SARIMA(1, 2, 0) $\times$ $(1, 0, 0)_7$ or SARIMA(1, 1, 0) $\times$ $(1, 1, 0)_7$ models.

3.2.3 Cumulative number of people dead time series

Originally, the third selected time series is also a cumulative sum. After the first differencing at lag 1, it is clear that it still has the global trend, but no seasonality at all.

Figure 3.4b demonstrates that after the second difference at lag 1, we got rid of the global trend.

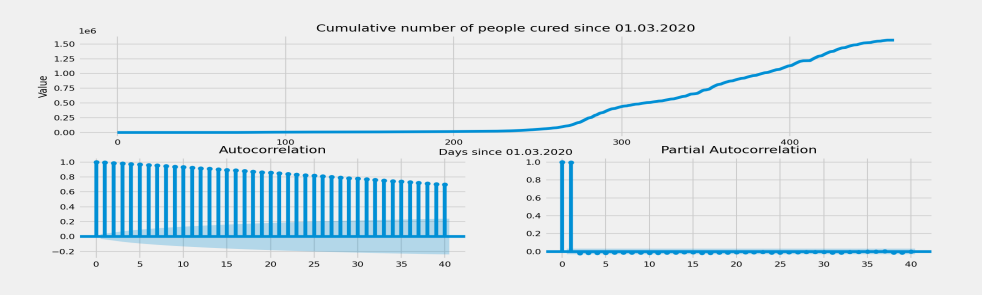
It is clear that there are no interesting phenomena in the data until the beginning of June (we can get rid of these measurements in the future). This period is also fundamentally different from the rest of the time series.

Time series	Dikey-Fuller test p-value	KPSS test p-value
Original time series	0.951460	0.010000
1x at lag 1	0.229294	0.010000
2x at lag 1	0.003183	0.030002
2x at lag 1 + Box-Cox	0.035768	0.100000
2x at lag 1 + drop	0.021909	0.061904

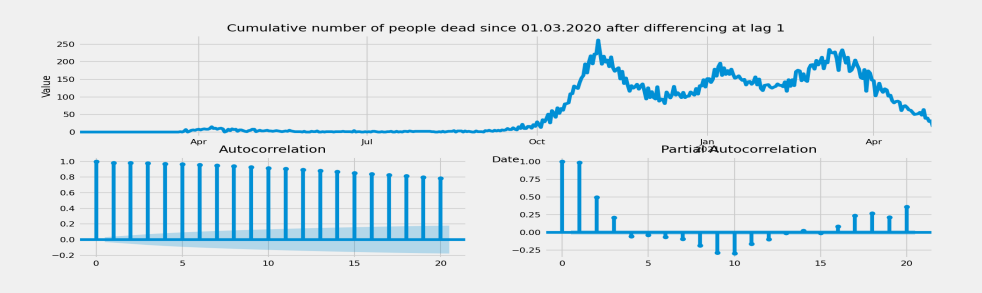
Table 3.4: The Cumulative number of people dead: results of the stationarity tests.

The ACF and PACF contain the drop after lag 1. It can indicate the absence of the autoregressive component, but the moving average order can be equal to 1. However, stationarity tests both reject the null hypothesis (Table 3.4). It means that the data are likely heteroscedastic and may have structural changes over time.

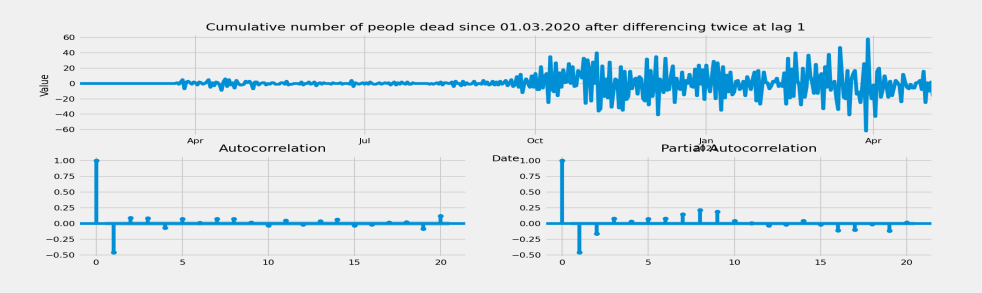
It is possible to reduce the heteroscedasticity in the data by application of, for example, Box-Cox or logarithmic transformations (according to their properties) or by dropping all measurements until June 1, 2020.



(a) The original cumulative number of people dead time series.



(b) The cumulative number of people dead time series after differencing at lag 1.



(c) The cumulative number of people dead time series after double differencing at lag 1.

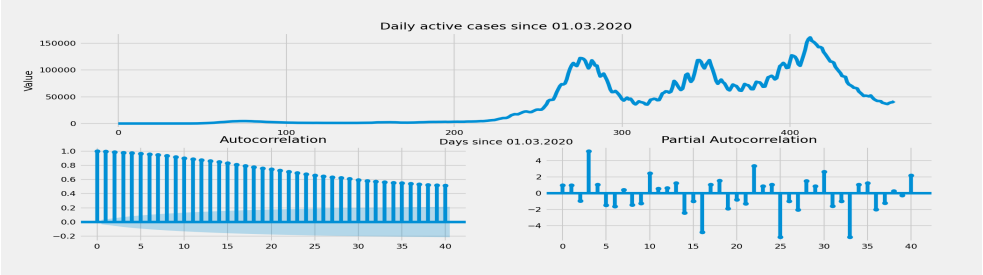
Figure 3.4: The cumulative amount of people cured after different differencing sequences with the corresponding ACF and PACF.

Table 3.4 shows that after at least one of these manipulations, the tests indicate the stationarity of the time series.

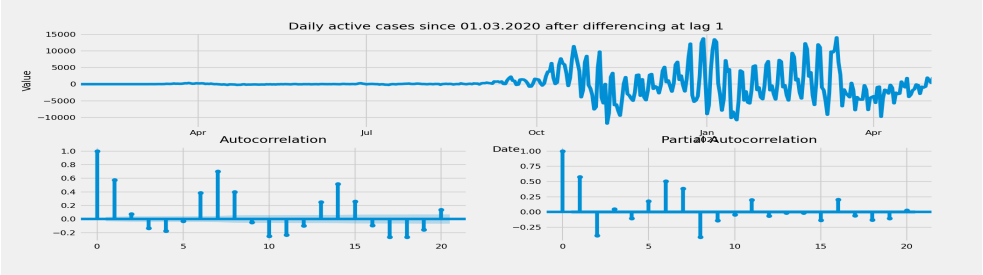
Now we can say that the cumulative number of people dead time series potentially can be modeled using ARIMA(0, 2, 1) model.

3.2.4 The number of active cases time series

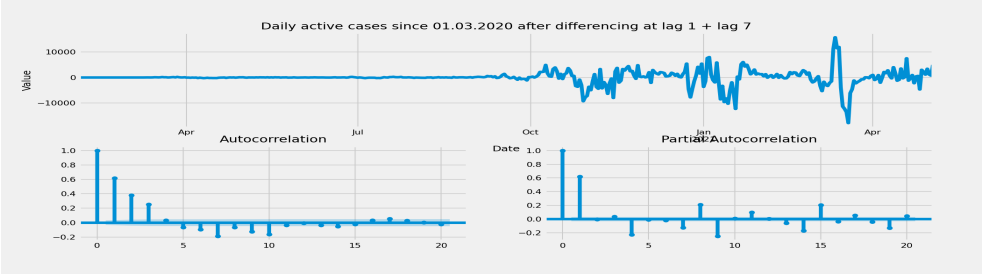
The fourth selected time series describes the number of active cases. In Figure 3.5a you can see it before any manipulation. It has a global trend, but we can



(a) The number of active cases time series



(b) The number of active cases time series after differencing at lag 1.



(c) The number of active cases time series after differencing at lag 1 and lag 7.

Figure 3.5: The number of active cases time series after different differencing sequences with the corresponding ACF and PACF.

say anything about the seasonal patterns or about the order of autoregressive and moving average components. Similar to the other time series, the period until June 1, 2020, is fundamentally different and does not contain any useful information. Stationarity testing (Table 3.5) indicates that the the time series is nonstationary. Thus, we need to perform differencing at lag 1.

Figure 3.5b shows that after the first differencing we got rid of the global trend, plus it is now clear that there is a seasonal pattern with weekly periodicity. The ACF and PACF show that the order of the seasonal autoregressive component is equal to 1 (significant correlation at lags 7, 14, and so on). According to the tests, the time series is now stationary. However, it is difficult to determine the order of the general autoregressive or moving average component.

After differencing at lag 7 (to remove seasonality), it is visible that the order of the general autoregressive component is equal to 1 (the value of the ACF slowly decreases, the value of the PACF drops after lag 1).

Time series	Dikey-Fuller test p-value	KPSS test p-value
Original time series	0.290402	0.017614
1x at lag 1	0.000158	0.100000
1x at lag 1; 1x at lag 7	0.000005	0.100000

Table 3.5: The number of active cases: results of the stationarity tests.

To summarize, this time series can be modeled using the SARIMA$(1, 1, 0) \times (1, 0, 0)_7$ or SARIMA$(1, 1, 0) \times (1, 1, 0)_7$ models.

3.3 Facebook Prophet modeling

After applying the basic analysis, the selected time series are ready for modeling using the Facebook Prophet model.

In this section, the more detailed pipeline has been formulated as

1. find the best configuration of all 3 defined components (trend, seasonal, holidays) for each selected time series.
2. perform hyperparameters tuning using the cross-validation.
3. evaluate the model using specified metrics.
4. obtain a set of changepoints for each time series individually, evaluate the possibility of using a single shared set for all selected time series.
5. create a custom set of changepoints based on the public information about restrictions provided by the Czech government and compare it with automatically detected
6. perform the forecasts
7. find some correlations between government restrictions and changes in the growth rate of the selected time series.

As we found out in the section of time series analysis, all selected time series have a part significantly distinct from the remaining time series (for example,

until June 1, 2020), where nothing special happens. We can ignore this data during the future modeling (because this period has no information about the evolution of the COVID-19 pandemic since the first wave). However, the Facebook Prophet model supports the global trend growth changes, so it will be enough to drop the period with the target variable equal to 0.

3.3.1 Parameter estimation mechanism

According to the specified pipeline, the first step is to perform hyperparameter tuning using the time series cross-validation technique.

We need to define the parameters of the cross-validation. It is also necessary to specify the parameters of the model used during the cross-validation that will can set manually. We will use the knowledge received during the basic time series analysis and the information from the official Facebook Prophet documentation [15].

The cross-validation parameters are:

- initial cutoff = 150 days ($\sim$ July 1, 2020, covers the beginning of the useful data),
- forecast horizon = 14 days (this value was selected because it is equal to 2 $\times$ length of the seasonal pattern and can be potentially useful during the pandemic analysis).

The manually specified parameters of the model are *set of holidays* and *seasonality* (for seasonal time series).

The set of holidays is specified using the inbuilt method of the Prophet model that adds a list of national holidays (depends on the country).

Table 3.6 demonstrates the curtain list of holidays specified for the Czech Republic.

For time series that contain seasonal patterns, we will also enable a *weekly seasonality*.

During the cross-validation, there is no need to compute the confidential intervals because we care only about the forecast accuracy in comparison with the historical data.

Index	Holiday name
0	Day of the Restoration of the Independent Czech State
1	Good Friday
2	Easter Monday
3	Labor Day
4	Victory Day
5	Day of the Slavic heralds Cyril and Methodius
6	Day of the burning of master Jan Hus
7	Czech Statehood Day
8	Day of the establishment of an independent Czechoslovak state
9	Day of the Struggle for Freedom and Democracy
10	Christmas Eve
11	1st Christmas holiday
12	2nd Christmas holiday

Table 3.6: The set of holidays used in the Prophet modeling.

We also will cut the last 14 days off to evaluate model performance on the unseen data.

3.3.1.1 Data transformation before modeling

For the Prophet modeling, we decided to drop the first 55 days (until the day of the first death) to get rid of measures with zero value and remain equal length of the selected time series.

3.3.1.2 Cross-validation accuracy metrics

For evaluation of the cross-validation results, we decided to use MAPE metric supported by the cross-validation function inside the Prophet diagnostics module [15]:

$$MAPE = \frac{100}{n} \sum_{t=1}^{n} \left| \frac{Y_t - \hat{Y}_t}{Y_t} \right|, \tag{3.2}$$

where n is an amount of forecasted points, Y_t is the real value of the observed variable Y at timestamp t, and $\hat{Y}_t$ is the forecasted value at timestamp t.

As we noticed earlier in Chapter 3, the resulting error is computed as the mean of the errors obtained from all forecasts made during the cross-validation process.

According to the fact that we have selected 4 different time series, we will perform 4 different cross-validations.

Table 3.7 contains information about the parameters to tune.

Parameter name	Possible values
`changepoint_prior_scale`	0.1, 0.5, 0.8, 1.0
`seasonality_prior_scale`	0.1, 1.0, 10.0
`seasonality_mode`	multiplicative, additive
`holidays_prior_scale`	0.1, 1.0, 10.0
`changepoint_range`	0.8, 0.9, 0.95

Table 3.7: The Prophet cross-validation parameter grid.

The `changepoint_prior_scale` parameter corresponds to the parameter τ of the Laplace distribution (Section 2.1.2.3). It controls the growth rate change flexibility. The `seasonality_prior_scale` parameter (for seasonal time series)regulates strength of the seasonal model. According to the official Prophet documentation [15], larger values allow larger seasonal fluctuations, smaller values mitigate the seasonality. The `seasonality_mode` parameter specifies a type of seasonal model (multiplier or addendum). The `holidays_prior_scale` parameter controls the sensitivity to the changes caused by holidays. The `changepoint_range` parameter defines the percentage of the time series (from the beginning) within which the changepoints will be distributed.

3.3.1.3 Cross-validation results

Table 3.8 contains information about the best performance parameters for each model.

It is visible that the cross-validation results are relatively satisfactory. Average MAPE varies between 5-28%.

The number of active cases time series has a larger average MAPE because it has more expressive growth changes (other time series contain a cumulative sum, this time series contains information about daily active cases). It means that, on average, the data before and after cutoffs have more distinct differences in trend directions. It can potentially affect the future forecast of the unseen data. To reduce this problem, we decided to apply *the log transformation* on this time series (according to the fact that we need to apply an inverse transform to all forecast components, we cant use Box-Cox transformation[3]).

[3]https://github.com/facebook/prophet/issues/647 — Box-Cox transformation usage with Prophet problem, the inverted sum does not equal to the sum of inverted components

	Parameter		*Error*
Time series	**Name**	**Value**	**MAPE (%)**
Cumulative number of people infected	changepoint_prior_scale	0.8	5.314
	seasonality_prior_scale	10.0	
	seasonality_mode	10.0	
	holidays_prior_scale	additive	
	changepoint_range	0.95	
Cumulative number of people cured	changepoint_prior_scale	1.0	5.381
	seasonality_prior_scale	1.0	
	seasonality_mode	additive	
	holidays_prior_scale	10.0	
	changepoint_range	0.95	
Cumulative number of people dead	changepoint_prior_scale	1.0	5.111
	holidays_prior_scale	1.0	
	changepoint_range	0.95	
Number of active cases	changepoint_prior_scale	1.0	28.957
	seasonality_prior_scale	10.0	
	seasonality_mode	mult.	
	holidays_prior_scale	0.1	
	changepoint_range	0.95	

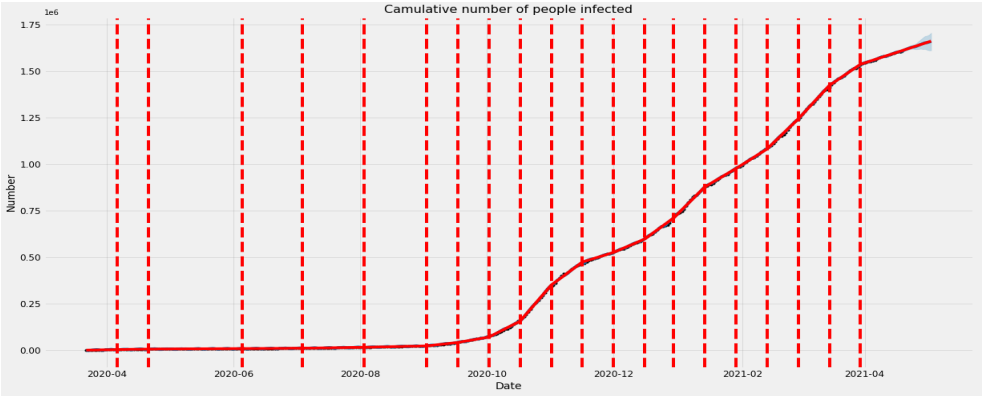
Table 3.8: The best set of the hyperparameters for each time series.

Logarithm transformation can invert the sum in such a way that it will be equal to the product of inverted components (additive model to multiplicative, because of the logarithm). Table 3.9 shows cross-validation results after this transformation. We have reduced the average MAPE from 28% to 2.5%.

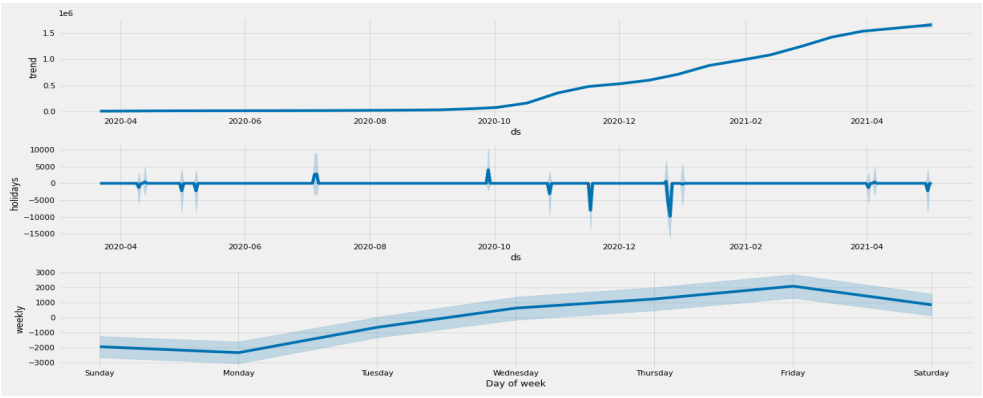
	Parameter		*Error*
Time series	**Name**	**Value**	**MAPE (%)**
Number of active cases + log. transformation	changepoint_prior_scale	1.0	2.455
	seasonality_prior_scale	10.0	
	holidays_prior_scale	10.0	
	seasonality_mode	additive	
	changepoint_range	0.95	

Table 3.9: The best set of the hyperparameters for the number of active cases time series after the logarithm transformation.

Now, we can fit new models (using tuned parameters) for each time series to analyze changepoints distribution and forecast accuracy for unseen data (the last 14 days of observations).



(a) Modeled time series.



(b) Forecast components.

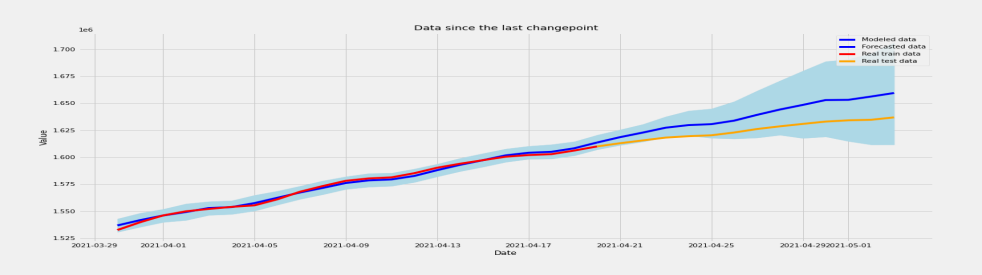
Figure 3.6: The cumulative amount of people infected modeled using the Prophet model + Forecast components.

3.3.2 Forecasting using Prophet model

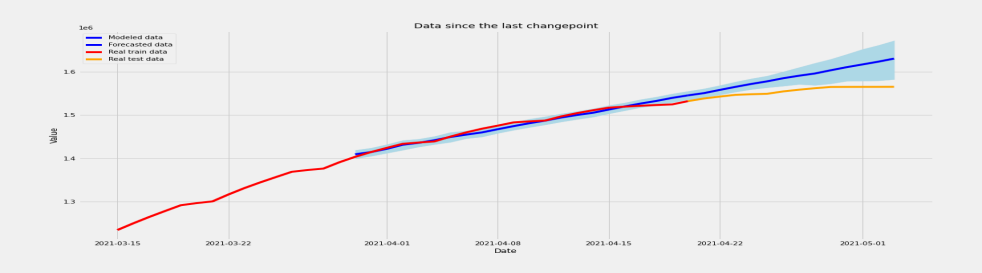
After fitting new models to the data, it is possible to perform a forecast with 14 days forecast horizon. Figure 3.6 shows an example of the resulting model plot with extra information about each forecast component individually. Additionally, in Figure 3.7a you can see a more detailed graph with the comparison of the forecasted and the original data since the last detected changepoint (May 30, 2021) for each selected time series.

In case of the number of active cases time series, we also applied an inverse transformation to compare the forecast with the original data.

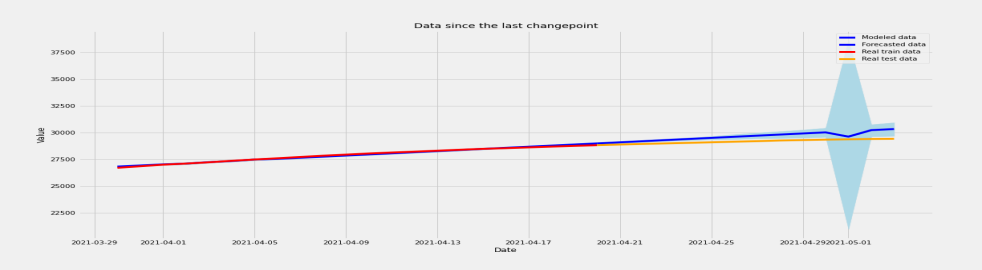
According to the graph above, the confidence intervals become dramatically wide after the first forested week. In case of the number of people dead, there is no enough data to include the influence of the Labor day (May 1, 2021),



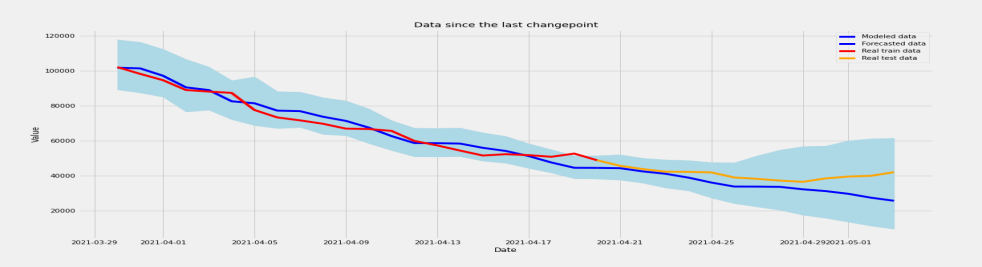
(a) Cumulative number of people infected time series forecast



(b) Cumulative number of people cured time series forecast



(c) Cumulative number of people dead time series forecast



(d) Number of active cases time series forecast

Figure 3.7: The selected time series forecasts (since the last detected changepoint May 30, 2021).

which follows extremely wide confidence intervals. This leads to the fact that only the first several forecasted days may be useful in practice.

In addition to the above visualizations, Table 3.10 contains information about forecast errors (MAPE).

Time series	Forecast MAPE (%)
Cumulative number of people infected	0.818
Cumulative number of people cured	2.143
Cumulative number of people dead	1.597
Number of active cases	14.37

Table 3.10: Results of the original time series forecast.

3.3.3 Changepoints

3.3.3.1 Automatically detected changepoints

While models fitting during the previous subsection, we have obtained a list with the changepoints. It is important to say that the changepoint lists were almost identical for all 4 time series. It is possible, because: all 25 potential changepoints are normally distributed within the first 95% of the time series (approx. twice a month).

Changepoint		Time series				Changepoint		Time series			
№	Date	I	C	D	A	№	Date	I	C	D	A
1	2020-04-06	+	+	+	+	14	2020-10-17	+	+	+	+
2	2020-04-21	+	+	+	+	15	2020-11-01	+	+	+	+
3	2020-05-06	+	+	-	+	16	2020-11-16	+	+	+	+
4	2020-05-21	+	-	+	+	17	2020-12-01	+	+	+	+
5	2020-06-05	+	-	-	+	18	2020-12-16	-	+	+	+
6	2020-06-20	+	-	-	+	19	2020-12-30	+	+	+	+
7	2020-07-04	+	+	-	+	20	2021-01-14	+	+	+	+
8	2020-07-19	-	-	+	+	21	2021-01-29	+	+	+	+
9	2020-08-03	+	+	-	+	22	2021-02-13	+	+	+	+
10	2020-08-18	+	-	-	+	23	2021-02-28	+	-	+	-
11	2020-09-02	+	+	+	+	24	2021-03-15	+	+	+	+
12	2020-09-17	+	+	+	+	25	2021-03-30	+	+	+	+
13	2020-10-02	+	+	+	+	-	-	-	-	-	-

Table 3.11: Automatically detected changepoints (date format: YYYY-MM-DD) including the information about their usage in different models.

The model tries to find the best changepoints subset with the lowest sum of growth change rate coefficients. In the selected time series, the growth rate changes frequently. and almost all potential changepoints are included in the final subset.

Table 3.11 contains information about the potential changepoints and their usage in all fitted models.

It is visible that the final subsets are similar. However, not all potential changepoints were included in every model.

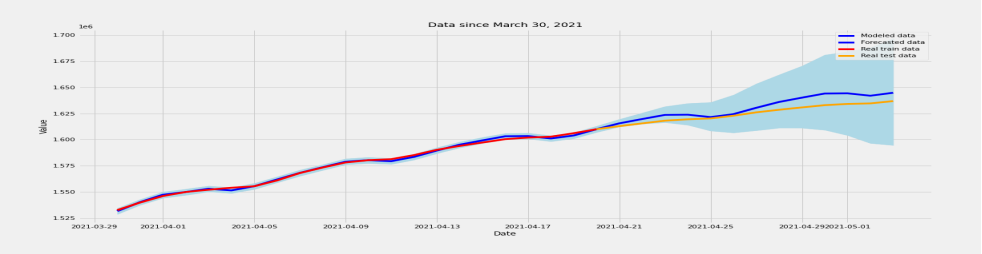
In this thesis, we will use this changepoints to measure the influence of fitting the model to the slice of the historical data since some changepoints. Thus, we have fitted multiple models to different slices of each selected time series. In Table 3.12 you can see the best forecast results for each time series.

Time series	Slice from	MAPE (%)
Cumulative number of people infected	2021-01-12	0.339
Cumulative number of people cured	2020-12-30	0.291
Cumulative number of people dead	2021-02-13	0.435
Number of active cases	2020-09-17	7.993

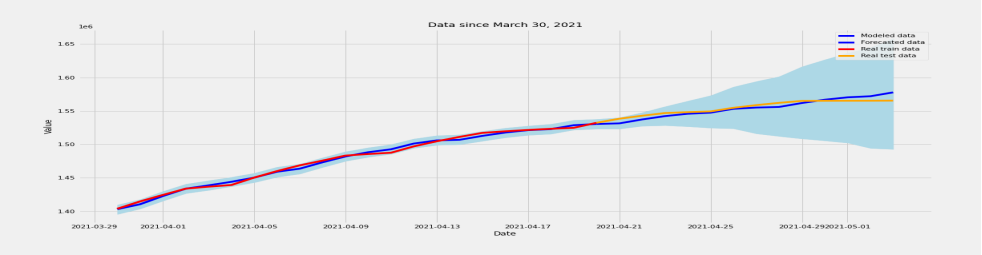
Table 3.12: Best results obtained from model fit to the slice of original data (from the changepoint).

As expected, the models that are fitted to the data slice, since some of the specified changepoints are more accurate, then the models that are fitted to all historical data. However, according to the following visualizations (Figure 3.8), the confidence intervals are increasing even faster then before.

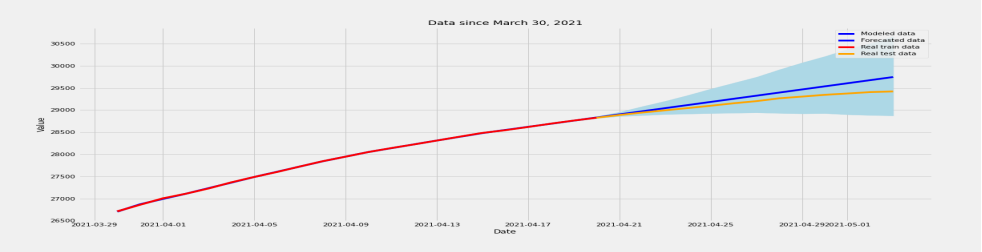
It is important to admit that we have disabled the holidays component, because it is not possible to compute their influence on the final forecast.



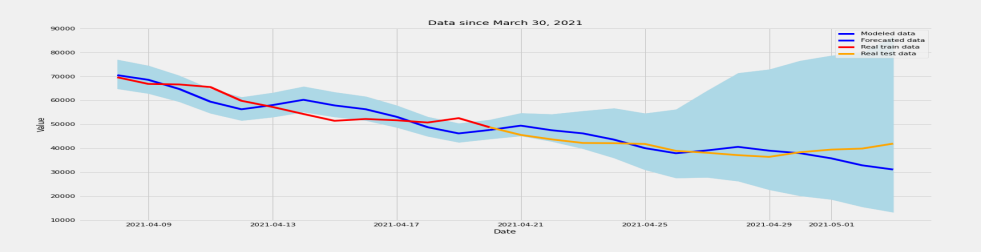
(a) Cumulative number of people infected time series forecast (fitted to the data since January 12, 2021)



(b) Cumulative number of people cured time series forecast (fitted to the data since December 30, 2020)



(c) Cumulative number of people dead time series forecast (fitted to the data since February 13, 2021)



(d) Number of active cases time series forecast (fitted to the data since September 17, 2020)

Figure 3.8: The selected time series forecasts (since March 30, 2021) fitted to the slice of data since the specified changepoint.

3.3.4 Interesting correlations between changepoints and government restrictions

One of the most interesting parts of this work is studying of the correlations between changepoints detected by the Prophet model (and their corresponding trend growth rate changes) and the list of government restrictions during the pandemic.

To perform this analysis, we have prepared the list of government restrictions since March 2, 2020. We have created our own list with 53 different restriction packs and other important events using the covid portal sponsored by the Ministry of the Interior of the Czech Republic [20].

In this section, we will introduce some important and potentially useful correlations between changes in trend growth rate and some precursory measures before them.

3.3.4.1 Correlation: Number of infected explosive growth slowdown

The following figure shows a slice of the time series that contains measurements between October and January (year 2020).

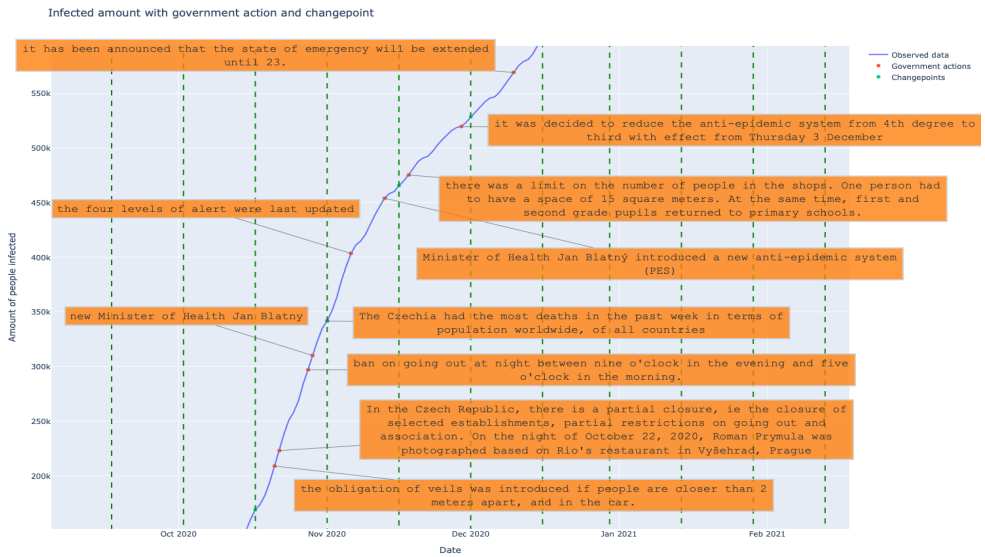


Figure 3.9: Cumulative number of infected growth slowdown October–November 2020.

During this period, the increase in the number of cases has slowed from a sharply explosive to a calmer (potentially linear) but still rapid growth. The

mean number of new cases between neighboring changepoints can be found in Table 3.13. It decreased from 11665 to 4209.7 (−227%).

Start date	End date	Mean number of active cases
2020-10-17	2020-11-01	11665.0
2020-11-01	2020-11-16	8337.3
2020-11-16	2020-12-01	4209.7

Table 3.13: Mean number of new cases daily between October 17, 2020 and December 1, 2020.

This change may indicate that the government restrictions occurred at a reasonable distance before they are potentially effective. In Figure 3.9 it is possible to see some labels that contain short descriptions of the restrictions between October 15, 2020, and December 15, 2020. **The most important are**:

- The obligation of face masks if people are closer than meters 2 apart (October 21, 2020).
- Partial closure (closure of selected establishments, October 22, 2020).
- Ban on going out at night between nine o'clock in the evening and five o'clock in the morning (October 28, 2020).

3.3.4.2 Correlation: Number of infected explosive growth slowdown 2

The second interesting slope change is depicted in Figure 3.10. It occurs during the period between February 13, 2021 and May 3, 2021.

According to Table 3.14, during February and the first half of March, the mean number of people infected daily had increased from 10169 to 11126. However, after March 15, it started to slow down to 3256 (on average) new cases daily (−341%).

Start date	End date	Mean number of active cases
2021-02-13	2021-02-28	10169.4
2021-02-28	2021-03-15	11126.5
2021-03-15	2021-03-30	8083.3
2021-03-30	2020-05-03	3256.9

Table 3.14: Mean number of new cases daily between February 13, 2021 and May 3, 2021.

This slope change may indicate that potentially effective restrictions were in place prior to this period.

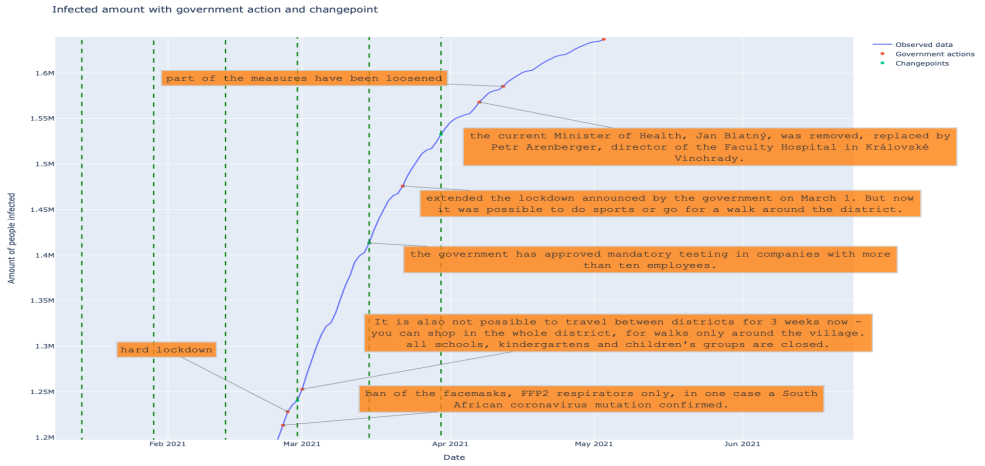


Figure 3.10: Cumulative number of infected growth slowdown February-May 2021.

We found out that before and during this time interval, the Czech government introduced the following important orders:

- Wearing face masks was prohibited, FFP2 respirators only. (February 25, 2021).
- Hard lockdown was announced (February 26, 2021).
- Travel ban between districts for 3 weeks. All schools, kindergartens, and children's groups were closed. (March 1, 2021).
- Mandatory testing in companies with more than ten employees has been approved (March 15, 2021).

3.3.4.3 Correlation: Number of infected explosive growth slowdown 3

It is important to mention a growth slowing during the spring of 2020 (Figure 3.11). At this time, the Czech Republic has faced the beginning of the COVID-19 pandemic. According to Table 3.15, mean number of people infected daily increased from 65.7 to 154.3 (+234%) and then reduced to 66.5 (−232%).

We discovered that before the beginning of the explosive increase of new cases, the government introduced the following important restrictions:

Start date	End date	Mean number of active cases
2020-03-01	2020-04-06	65.7
2020-04-06	2020-04-21	154.3
2020-04-21	2020-05-06	66.5

Table 3.15: Mean number of new cases daily between April 6, 2020 and May 6, 2020.

- All primary, secondary, higher vocational, and university schools in the Czech Republic were closed until further notice (March 11, 2020).
- A state of emergency was declared in the Czech Republic for a period of 30 days (March 12, 2020).
- The operation of restaurants and shops was banned (March 14, 2020).
- State borders have been closed (with a few exceptions, March 16, 2020).

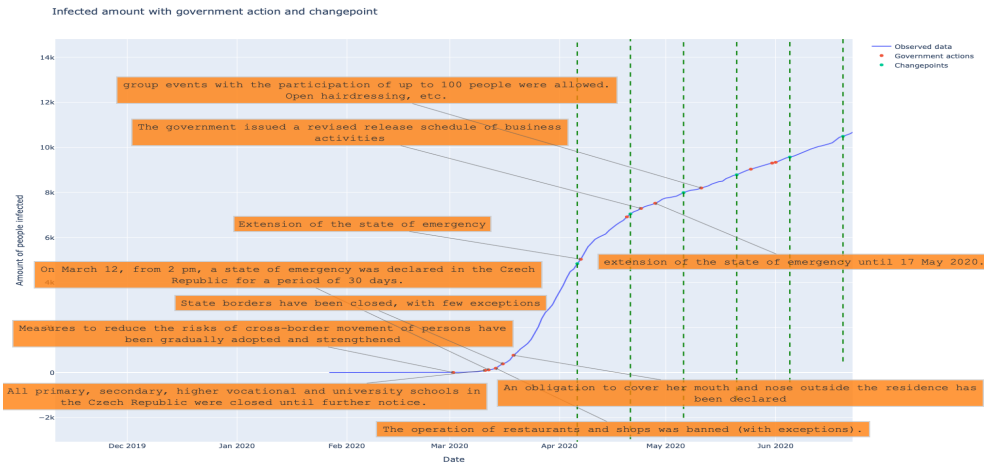
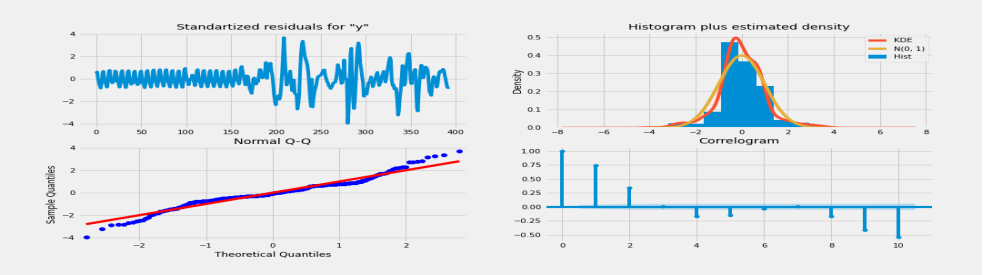


Figure 3.11: Cumulative number of infected growth slowdown April-May 2020.

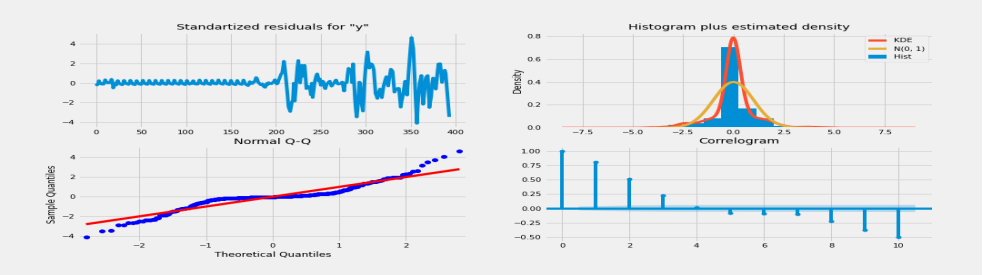
3.3.5 Residual analysis

Residual analysis is a fundamental step of the model diagnostics process. Thus, we decided to test the models introduced in Subsection 3.3.2 and Subsection 3.3.3.

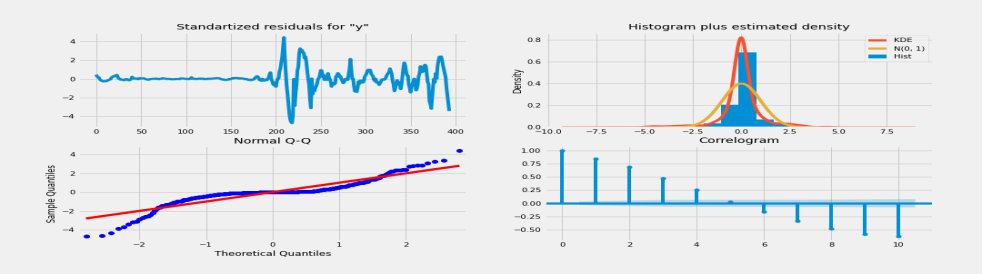
We consider models fitted to all historical data as more representative (in the context of the whole pandemic) than the models fitted to the data slices since the specific global changepoints and will perform their analysis first.



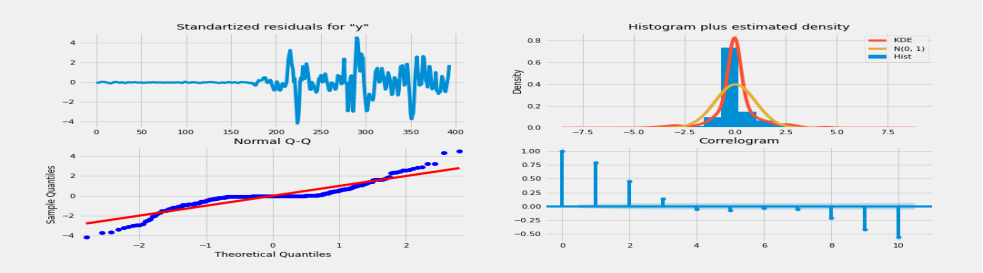
(a) Cumulative number of new cases model: Residual analysis.



(b) Cumulative number of people cured model: Residual analysis.



(c) Cumulative number of people dead model: Residual analysis.



(d) Number of active cases model: Residual analysis.

Figure 3.12: Residual analysis for the different Prophet models.

Analysis pipeline is specified, for example, in Hyndman et al. [4] and contains the statistical testing of the probability that residuals are normally distributed (Jarque-Bera test), testing of the probability of correlation between residuals,

and heteroscedasticity testing. Moreover, it implies different visualizations such as Q-Q plot, correlogram, and distribution histogram.

Figure 3.12 demonstrates the different residual diagnostic plots. It is visible that all 4 models have some seasonal correlations between their residuals[4]. This may indicate that the models do not cover some seasonal relations in the data. Q-Q plots and histograms show that all 4 models do not have a normal distribution of the residuals. Additionally, Table 3.16. contains the results of residual statistical testing[5] that approve the information received from the visualizations above.

Model	Ljung-Box test	Jarque-Bera test	Heteroscedasticity test
Cumulative amount of people infected	p-value ≈ 0; reject	p-value ≈ 0; reject	-
Cumulative amount of people cured	p-value ≈ 0; reject	p-value ≈ 0; reject	-
Cumulative amount of people dead	p-value ≈ 0; reject	p-value ≈ 0; reject	-
Active cases	p-value ≈ 0; reject	p-value ≈ 0; reject	-

Table 3.16: Statistical residual testing of the Prophet models fitted to all historical data.

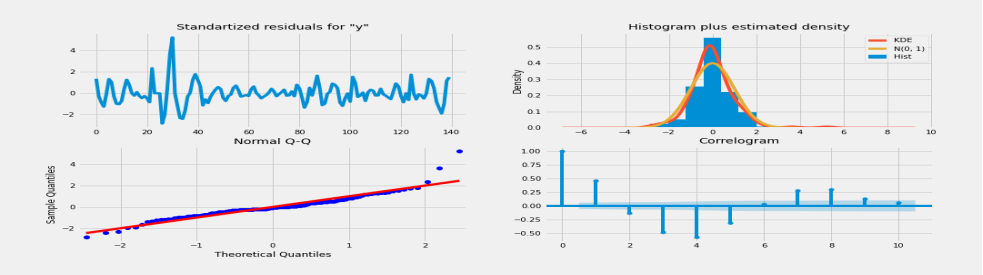
Now we can also perform the statistical residual testing of models fitted to the data slices. Table 3.17 shows that the models fitted to the slices of the data with the information about the number of people cured and dead may have normally distributed residuals. However, we still can not reject that the residuals are correlated.

Model	Ljung-Box test	Jarque-Bera test	Heteroscedasticity test
Cumulative amount of people infected	p-value ≈ 0; reject	p-value ≈ 0; reject	-
Cumulative amount of people cured	p-value ≈ 0; reject	p-value $= 0.98$; can not reject	-
Cumulative amount of people dead	p-value ≈ 0.0001; reject	p-value $= 0.21$; can not reject	-
Active cases	p-value ≈ 0; reject	p-value ≈ 0; reject	-

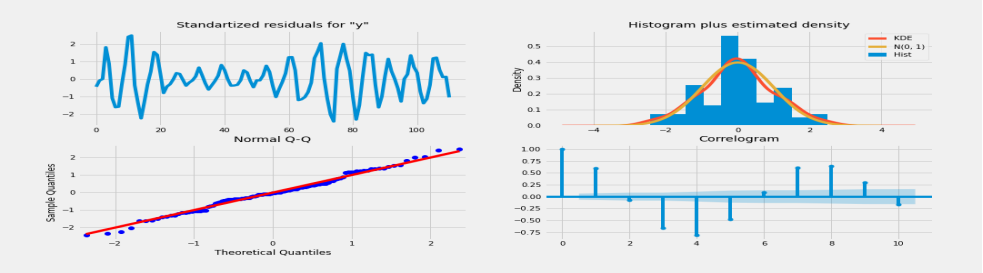
Table 3.17: Statistical residual testing of the Prophet models fitted to the slices of historical data.

[4] `https://github.com/facebook/prophet/issues/1622` — issue about residual problem in Prophet model. Almost all models have AR(1) relation in residuals. However, seasonal patters may indicate missing or not perfectly estimated seasonality.

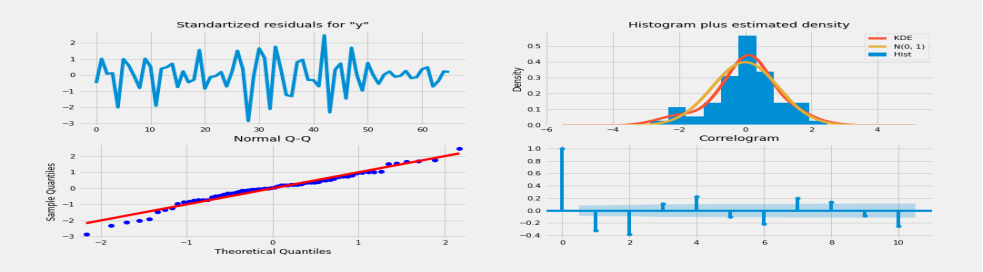
[5] The Prophet model does not support residual heteroscedasticity testing.



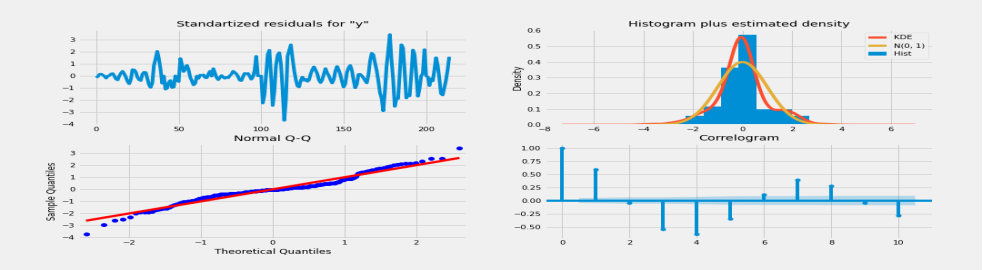
(a) Cumulative number of new cases model: Residual analysis.



(b) Cumulative number of people cured model: Residual analysis.



(c) Cumulative number of people dead model: Residual analysis.



(d) Number of active cases model: Residual analysis.

Figure 3.13: Residual analysis for the different Prophet models fitted to the slices of the historical data.

In Figure 3.13 we can also see that in all models (except for the one that describes the number of deaths) some uncovered seasonal relations occur. The

Q-Q plots and histogram indicates that all 4 models now have their residuals distributed more normally. This follows that fitting the model to the slice of the historical data reduces the number of uncovered processes that may influence this data.

3.4 SARIMA modeling

In this section, we perform modeling of the selected time series using SARIMA model implemented in Python library named *statsmodels* [21]. It has all important functionalities for model fitting, forecast, and residual analysis.

3.4.1 SARIMA model order estimation

In Section 3.2 we have already estimated the order of the (S)ARIMA models which can be used for modeling the selected time series:

- Cumulative number of people infected — SARIMA(1, 1, 0)×(1, 1, 0)$_7$.
- Cumulative number of people cured — SARIMA(1, 1, 0)×(1, 1, 0)$_7$.
- Cumulative number of people dead — ARIMA(0, 2, 1).
- Number of active cases — SARIMA(1, 1, 0)×(1, 1, 0)$_7$.

3.4.2 Forecasting using SARIMA

Similar to the Facebook Prophet forecasts, for SARIMA modeling, we decided to use the 14 days forecast horizon. Moreover, we applied the logarithm transformation to the time series with information about active cases and dropped the first 55 days of measurements (to get rid of measurements with 0 value).

In Table 3.18 you can find information about MAPE on train data and on the forecast of new unseen 14 days of data. Models that describe the cumulative sum time series have a forecast MAPE between 0.68 % and 0.80%. In difference, the number of active case forecast MAPE is equal to 41.7%.

Figure 3.14 contains visualizations of the forecasts made. It covers the period since the last changepoint detected by Prophet (March 30, 2021). It is visible that the forecasts are relatively successful. However, the forecasted number of active cases differs distinctly from the real data. Confidence intervals become extremely wide after the first few days of each forecast.

Time series	Train MAPE	Forecast MAPE (%)
Cumulative number of people infected	1.440	0.800
Cumulative number of people cured	1.580	0.680
Cumulative number of people dead	1.320	0.720
Number of active cases	4.710	41.760

Table 3.18: Results of the original time series forecast using SARIMA models.

3.4.3 Changepoints usage influence on SARIMA model

In the Prophet modeling section, we discovered that fitting the model to a slice of the historical data may improve the forecast results. This is possible because the selected COVID-19 time series change their development over time. In this subsection, we decided to study the influence of fitting the SARIMA model to the slices of data on the final forecast of the unseen data.

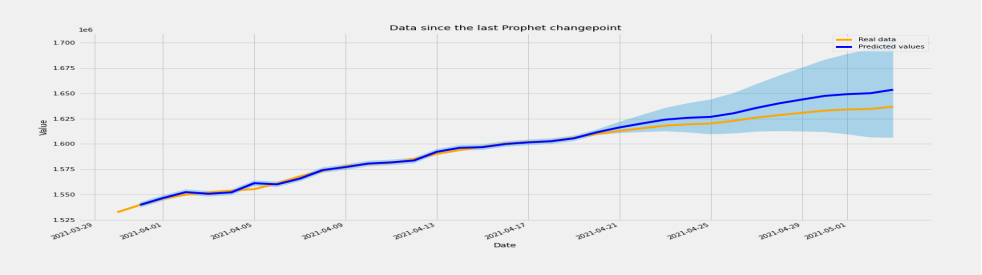
We fitted various models to the slices of the data since every changepoint detected in Section 3.3.3. Table 3.19 contains information about the accuracy of the best forecasts obtained in this way for each time series.

Time series	From date	Train MAPE (%)	Test MAPE (%)
Cumulative number of people infected	2020-10-02	1.740	0.520
Cumulative number of people cured	2020-12-30	9.590	3.780
Cumulative number of people dead	2020-10-02	2.020	0.720
Number of active cases	2021-03-15	3.620	8.950

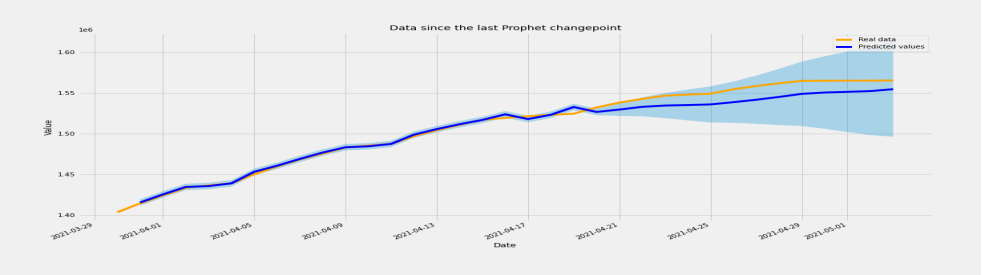
Table 3.19: Best results of the time series forecast using SARIMA models fitted to the slices of data.

According to these measurements, the prediction error of the number of new and active cases reduced from 0.8% to 0.52% and from 41.7% to 8.95%. However, the cumulative number of people dead forecast error did not change at all, and the cumulative number of people cured forecast error increased from 0.68% to 3.78%.

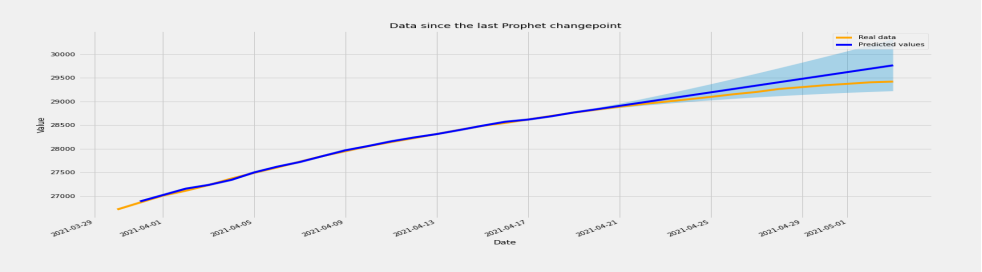
Figure 3.15 contains the corresponding forecast visualizations. On average,



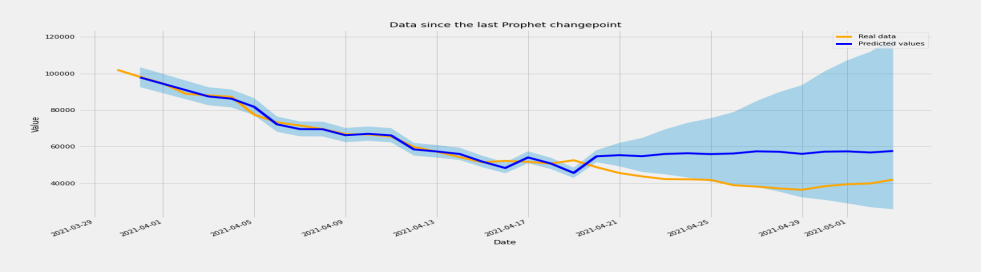
(a) Cumulative number of people infected time series forecast



(b) Cumulative number of people cured time series forecast



(c) Cumulative number of people dead time series forecast

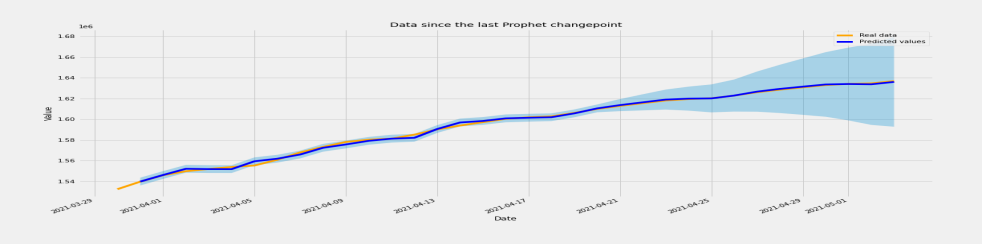


(d) Number of active cases time series forecast

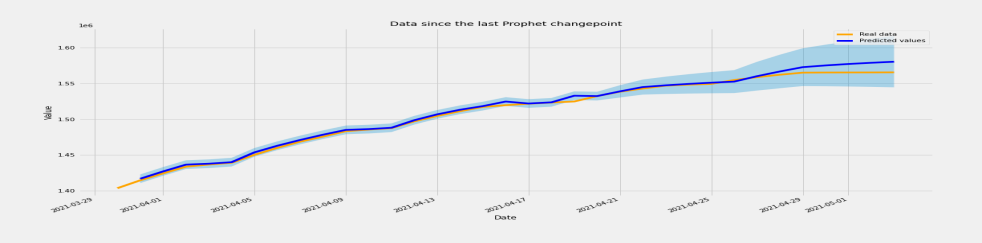
Figure 3.14: The selected time series forecasts using SARIMA model.

the confidence intervals became thinner (which indicates a reduction of the forecast uncertainty). Additionally, visualization (Figure 3.15d) of the active case forecast demonstrates significant improvements.

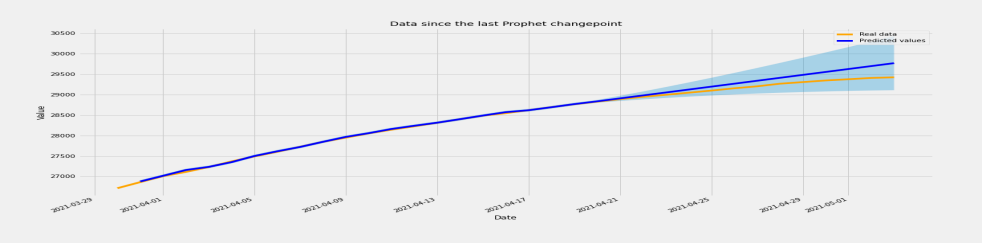
The results obtained during this experiment are mixed. Thus, we need to analyze the forecast residuals of all fitted models to conclude.



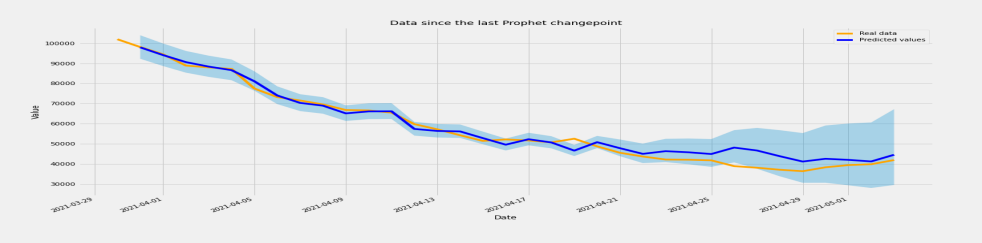
(a) Cumulative number of people infected time series forecast (data since October 2, 2020)



(b) Cumulative number of people cured time series forecast (data since December 30, 2020)



(c) Cumulative number of people dead time series forecast (data since October 2, 2020)

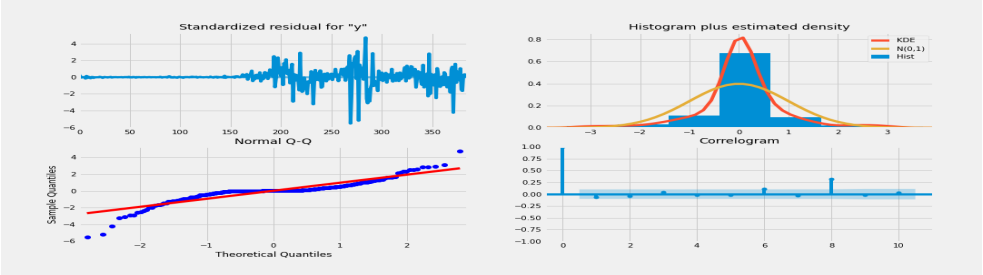


(d) Number of active cases time series forecast (data since March 15, 2021)

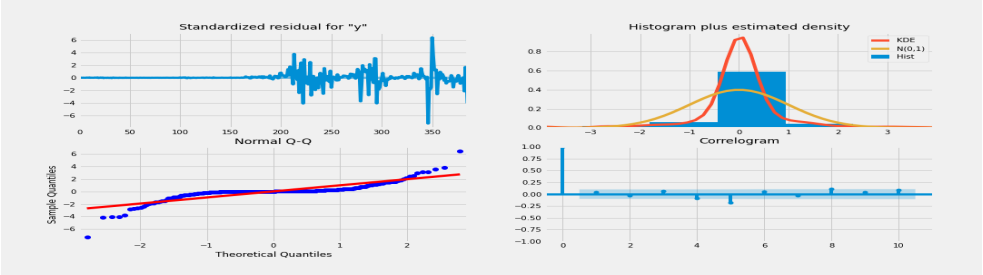
Figure 3.15: The selected time series forecasts using SARIMA (since March 30, 2021) fitted to the slice of data since the specified changepoint

3.4.4 Residual Analysis

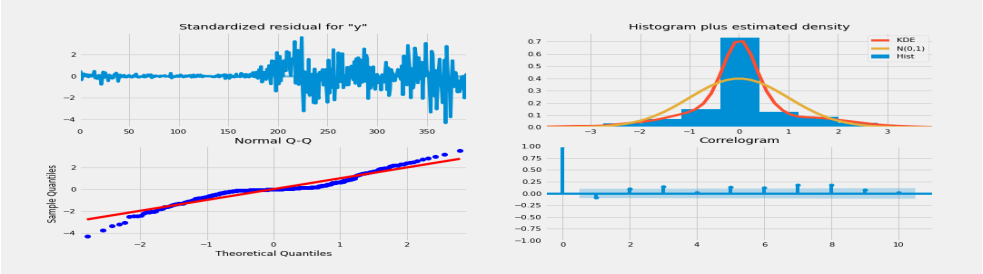
After fitting the SARIMA model, statsmodels allows us to perform residual analysis. Similar to the Prophet, we will use residual testing and visualization.



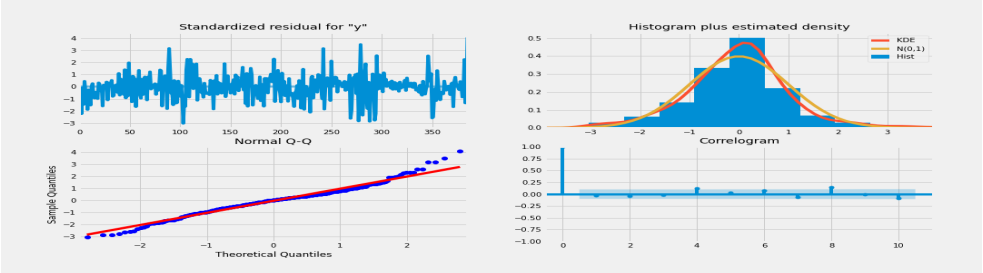
(a) Cumulative number of new cases model: Residual analysis.



(b) Cumulative number of people cured model: Residual analysis.



(c) Cumulative number of people dead: Residual analysis.



(d) Cumulative number of active cases: Residual analysis.

Figure 3.16: Residual analysis for the different SARIMA models.

First, we will analyze the models fitted to all historical data.

According to Figure 3.16, number of new cases model has a residual correlation

at lag 8, while the other 3 models may have uncorrelated residuals. However, according to the Ljung-Box test (Table 3.20) we reject the hypothesis that the residuals are not correlated only for the model that describes the cumulative number of people dead time series.

Model	Ljung-Box test	Jarque-Bera test	Heteroscedasticity test
Cumulative amount of people infected	p-value = 0.23; can not reject	p-value = 0.00; reject	p-value = 0.00; reject
Cumulative amount of people cured	p-value = 0.37; can not reject	p-value = 0.00; reject	p-value = 0.00; reject
Cumulative amount of people dead	p-value = 0.00; reject	p-value = 0.00; reject	p-value = 0.00; reject
Active cases	p-value = 0.60; can not reject	p-value = 0.00; reject	p-value = 0.36; can not reject

Table 3.20: Statistical residual testing of the SARIMA models fitted to all historical data.

Q-Q plots and histograms indicated that all 4 model residuals are not normally distributed. Jarque-Bera testing confirms it.

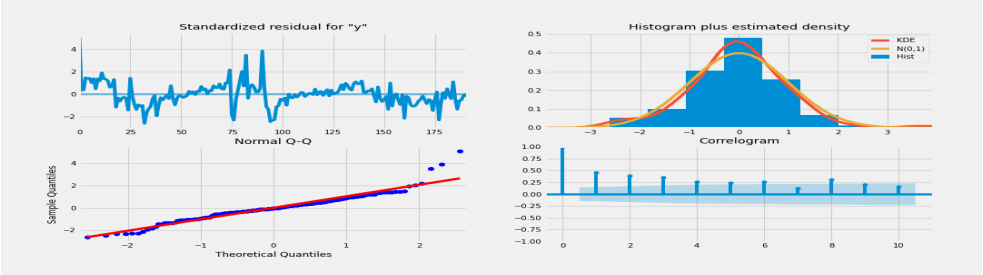
The heteroscedasticity test indicates that the model that describes the number of active cases time series has residuals that do not change their structure over time (more relevant long-term forecasts). However, this result differs from the others because we applied a logarithm transformation to this time series (which reduces the heteroscedasticity in data).

Now we can move to the residual analysis of the models fitted to the slices of the data since the specified changepoints.

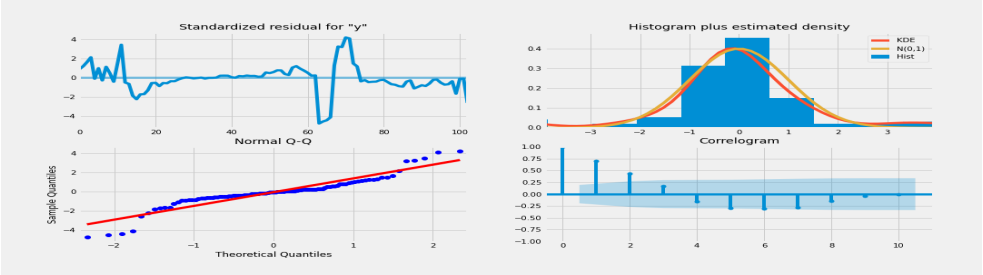
Model	Ljung-Box test	Jarque-Bera test	Heteroscedasticity test
Cumulative amount of people infected	p-value = 0.00; reject	p-value = 0.00; reject	p-value = 0.00; reject
Cumulative amount of people cured	p-value = 0.00; reject	p-value = 0.00; reject	p-value = 0.26; can not reject
Cumulative amount of people dead	p-value = 0.23; can not reject	p-value = 0.73; can not reject	p-value = 0.64; can not reject
Active cases	p-value = 0.97; can not reject	p-value = 0.00; reject	p-value = 0.00; reject

Table 3.21: Statistical residual testing of the SARIMA models fitted to slices of the historical data.

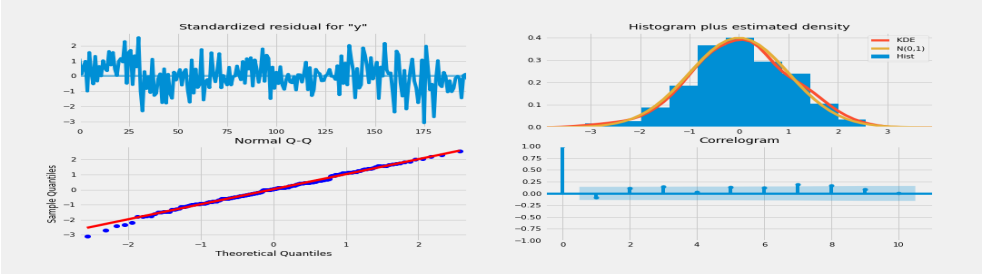
According to Figure 3.17, models that describe the cumulative number of people cured and infected now have correlated residuals. Ljung-Box test results (Table 3.21) confirm that. Their residuals also are not normally distributed.



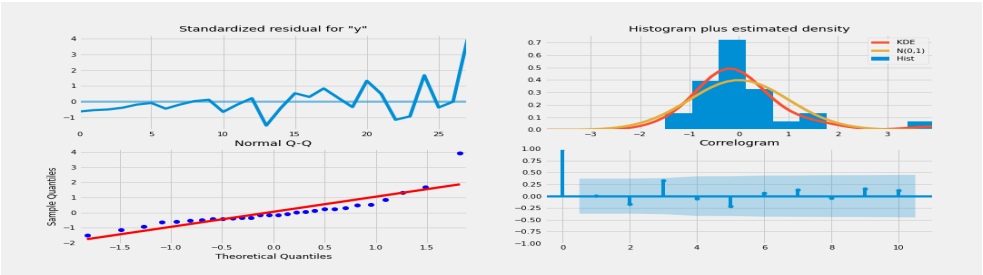
(a) Cumulative number of new cases model: Residual analysis.



(b) Cumulative number of people cured model: Residual analysis.



(c) Cumulative number of people dead: Residual analysis.



(d) Number of active cases: Residual analysis.

Figure 3.17: Residual analysis for the different SARIMA models fitted to the slices of data since specified changepoints.

However, the number of people cured time series has heteroscedastic residuals (which may be helpful for long-term forecasts).

The number of people dead model has noncorrelated, potentially normally distributed, heteroscedastic residuals and may be potentially used for long-term forecasts.

The number of active cases model also has noncorrelated residuals, but they are not heteroscedastic and not normally distributed.

3.5 Results evaluation and discussion

3.5.1 Time series analysis results evaluation

In Section 3.2 we performed the basic analysis of the selected time series. Using the theory studied during the theoretical research, we have found that:

- It is possible to separate different processes and components that influence the evolution of the COVID-19 time series, such as global trend or seasonality (using the ACF and PACF).
- It is possible to estimate the order of the SARIMA model that can be used for modeling these time series, using methods that remove global trend and seasonality (differencing).
- The selected time series are changing their behavior over time. This fact requires the usage of models that can handle these changes.

This information is helpful for future analysis steps and reduces the number of problems that may occur during time series modeling using the Facebook Prophet or SARIMA model.

3.5.2 Facebook Prophet modeling results evaluation

In Section 3.3 we studied the possibility of modeling the COVID-19 time series using the Facebook Prophet model and obtained some interesting results.

First, we discovered that it is possible to estimate proper hyperparameters using the inbuilt cross-validation technique. Models fitted using these parameters can achieve reasonable results.

Interestingly, the time series that contain the cumulative number of people infected, people cured, and people dead can be modeled without any data transformation. The number of active cases time series requires logarithm or Box-Cox transformation to reduce the measure of structural changes in the data. The forecasts made using the Prophet model fitted to all historical data

are relatively accurate: the cumulative sum time series forecast error is within 0.8% and 2.1%. However, the number of active case predictions on average have 14% error.

Unexpectedly, the Facebook Prophet model residual analysis process has some potential problems (due to an incomplete model component). However, it is possible to detect some changes not covered by our models.

As expected, using the piecewise model allowed us to create a list of global trend changepoints. We can use them to improve the forecast accuracy by fitting the model to the slice of the data starting at these changepoints. Cumulative sum forecasts now have an error between 0.3% and 0.4%, active cases prediction error reduced to 7.9%. This find indicates that the evolution of the COVID-19 processes changes over time. The measurements taken during the beginning may deter the modeling of further pandemic development.

The most striking observation to emerge from the Prophet analysis was the correlation detection between government restrictions and global slope changes. Specific examples can be found in Section 3.3.4.

All these results broaden our understanding of the COVID-19 pandemic modeling using the Facebook Prophet model.

3.5.3 SARIMA modeling results evaluation

Section 3.4 was aimed at the modeling of the COVID-19 processes using the SARIMA model. During this process, we used the information obtained from the time series analysis section to fit models of proper order.

Similar to the Prophet section, in the beginning, we used all historical data to fit the first bunch of models. The forecasts of the cumulative sums are used to have a high accuracy rate from the beginning (forecast error between 0.7-0.8%). However, the prediction of the number of active cases was poor (forecast error equal to 41%). Almost all these models (infected, cured, active) have wide confidence intervals (especially the model that describes the number of active cases). Nearly all models (except active cases) have non-heteroscedastic residuals, however, models that describe the number of active cases, people infected, and cured have their residuals noncorrelated. This find indicates that these models can be used only for short-term forecasts (few days).

We also evaluated the influence of fitting SARIMA models to the slices of the data. For this, we used the changepoints detected by the Prophet model. By doing this, we improved our prediction of active cases (error reduced from

41% to 8.9%), a cumulative number of people infected (0.8% to 0.5%). On the other side, the forecast accuracy of the number of people dead did not change at all, and the forecast of the number of people cured became less accurate (error increase from 0.7% to 3.78%). It is important to say that the confidence intervals became much thinner. The residual analysis also demonstrates mixed results.

In the case of the number of people dead time series, the residuals are noncorrelated, normally distributed, and heteroscedastic (this model may be helpful during long-term forecasts). However, the number of people cured and infected models now have a significant correlation between residuals. It may indicate the uncovered systematic processes that influence our model).

These findings confirm those of the Prophet modeling section, such as fitting models to the slices of data makes the confidence interval thinner and, on average, decreases the forecast error. Moreover, it is now clear that sometimes we may see no improvement at all. It depends on the absence of major relations before the cutoff.

Altogether, the results obtained with the SARIMA modeling are comparable to those obtained with the Prophet modeling. It is also possible to use the combination of these two models: to compare and supplement the results, or for changepoint detection with Prophet model and further usage in SARIMA modeling.

Conclusion

At the beginning of 2020, the world was faced with a new disease called COVID-19. From the information-theoretic viewpoint, the pandemic is unique due to an abnormal number of available data sets. Many of them are in the form of time series.

The goal of this bachelor thesis was to perform statistical analyses of the selected time series related to the pandemic in the Czech Republic. For this, we have selected the time series with information about:

- The cumulative number of people infected.
- The cumulative number of people cured.
- The cumulative number of people dead.
- The number of active cases.

Theoretical research

Before performing our modeling, we studied different literature aimed at theory related to the study of the time series. It allowed us to learn possible methods of time series analysis, decomposition, modeling, and further development prediction.

According to the fact that the pandemic is a world problem, we have expected to find other publications and researches dedicated to the same problem. Reading and studying these materials assured us that it is possible to analyze the COVID-19 time series using statistical models and methods.

Time series analysis

After cleaning and preparing the selected data sets, it was necessary to perform basic time series analysis, such as decomposition of the trend, seasonal and cyclic components, and estimation of seasonal period duration. We also estimated the order of the autoregressive and moving average time series components for further SARIMA modeling. To do this, we used the information obtained by time series differencing and the analysis of the (partial) autocorrelation function. We have discovered that the selected time series change their behavior over time (nonstationarity). Moreover, all of them have global trend growth rate changes, nearly all of them have weekly seasonality. This find follows the usage of models that can handle these time series features.

Model selection

For this thesis, we decided to use the Facebook Prophet and Seasonal Autoregressive Moving Average (SARIMA) models. Both of them can handle seasonal nonstationary time series that change their evolution over time.

Facebook Prophet modeling

First, we decided to model the selected time series using the Facebook Prophet. We fitted the models to all the historical data. It allowed us to detect the trend changepoints (during the whole pandemic) for each time series individually. Interestingly, the cumulative sum time series can be forecasted with a high accuracy rate even without any special data preparation. However, the daily number of active cases required the data transformation (Box-Cox or logarithm) to reduce structural changes. Then we used the changepoints to fit new models to the data starting from these changepoints. After doing it, we improved the forecast accuracy for all selected time series.

The Prophet model uses the piecewise trend concept to handle the global trend growth changes. We used this property to detect the correlations between the detected changepoints and various government restrictions that occurred during the pandemic. The most striking observation to emerge from the correlation analysis was the correlation between the reduction of the daily mean number of new cases and the government restrictions aimed at wearing face masks (respirators), social distancing, and (partial) lockdown.

SARIMA modeling

The next step was SARIMA modeling. To select the proper order of autoregressive and moving average components, we used the information obtained

during the time series analysis. Similarly to the Prophet modeling, the first bunch of models was fitted to all available historical data. In this case, the prediction accuracy was comparable to that obtained using the Prophet model, however, the number of active cases prediction was poor (MAPE = 41%), and the confidence intervals were broad (except the number of people dead forecast).

As the next step of the analysis, we explored the potential synergy between the SARIMA model and trend changepoints obtained using the Prophet model.

The outcome was not that unambiguous like in the case of the Prophet model: some forecasts became less accurate, however, the confidence intervals became thinner. It follows that fitting the model to the slice of the historical data since the specified trend changepoint may improve the forecast only if the earlier data do not contain essential information about the processes that influence time series development.

Summary

In this thesis, we discovered that it is possible to perform the COVID-19 analysis using statistical analysis and modeling. We found out that the Facebook Prophet and SARIMA models allow receiving significant results and can be used in combination or in isolation. However, because of the unstable evolution of the pandemic processes, only the first few days of the forecast can be used in practice.

Future work

This study provides the backbone for several possible extensions. First, using the judgmental forecast methods (described, for example, in Hyndman et al. [4]) in association with an epidemiology expert may improve prediction accuracy and result interpretation. Second, it is possible to extend a list of the selected time series by including vaccination, reproductive number, distribution of new cases severity, and so on. It will expand the search range for possible anomalies and interesting phenomena within the time series. Moreover, it is reasonable to test various data preparation and transformation techniques that are not discussed in this thesis.

Bibliography

[1] Hu, B.; Guo, H.; et al. Characteristics of SARS-CoV-2 and COVID-19. *Nature Reviews Microbiology*, volume 19, no. 3, Mar 2021: pp. 141–154, ISSN 1740-1534.

[2] Cryer, J. D.; Chan, K. *Time series analysis: with applications in R*. Switzerland: Springer Science & Business Media, second edition, 2008, ISBN 978-0-387-75959-3.

[3] Shumway, R. H.; Stoffe, D. S. *Time Series Analysis and Its Applications With R Examples*. Switzerland: Springer Science & Business Media, third edition, 2011, ISBN 978-1-4419-7864-6.

[4] Hyndman, R.; Athanasopoulos, G. *Forecasting: Principles and Practice [online]*. Australia: OTexts, second edition, 2018, [cit. 10 May 2021]. Available from: `https://otexts.com/fpp2/`

[5] Cleveland, R. B.; Cleveland, W. S.; et al. STL: A seasonal-trend decomposition. *Journal of official statistics*, volume 6, no. 1, 1990: pp. 3–73.

[6] Box, G.; Jenkins, G.; et al. *Time Series Analysis: Forecasting and Control*. Holden-Day series in time series analysis and digital processing, Canada: Holden-Day, 1976, ISBN 9780816211043.

[7] Kwiatkowski, D.; Phillips, P. C.; et al. Testing the null hypothesis of stationarity against the alternative of a unit root: How sure are we that economic time series have a unit root? *Journal of Econometrics*, volume 54, no. 1, 1992: pp. 159–178, ISSN 0304-4076.

[8] Sengupta, S.; Mugde, S.; et al. Covid-19 Pandemic Data Analysis and Forecasting using Machine Learning Algorithms. *medRxiv*, 2020.

[9] Wang, P.; Zheng, X.; et al. Prediction of epidemic trends in COVID-19 with logistic model and machine learning technics. *Chaos, Solitons & Fractals*, volume 139, 2020: p. 110058, ISSN 0960-0779.

[10] Indhuja, M.; Sindhuja, P. Prediction of covid-19 cases in India using prophet. *International Journal of Statistics and Applied Mathematics*, volume 5, no. 4, 2020: pp. 103–106, ISSN 2456-1452.

[11] ArunKumar, K.; Kalaga, D. V.; et al. 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). *Applied Soft Computing*, volume 103, 2021: p. 107161, ISSN 1568-4946.

[12] Ceylan, Z. Estimation of COVID-19 prevalence in Italy, Spain, and France. *Science of The Total Environment*, volume 729, 2020: p. 138817, ISSN 0048-9697.

[13] Taylor, S. J.; Letham, B. Forecasting at Scale. *The American Statistician*, volume 72, no. 1, 2018: pp. 37–45.

[14] Kotz, S.; Kozubowski, T. J.; et al. Classical Symmetric Laplace Distribution. In *The Laplace Distribution and Generalizations: A Revisit with Applications to Communications, Economics, Engineering, and Finance*, Boston, MA: Birkhäuser Boston, 2001, ISBN 978-1-4612-0173-1, pp. 15–131.

[15] Facebook. Prophet: Forecasting at Scale [online]. [cit. 22 April 2021]. Available from: `https://facebook.github.io/prophet/`

[16] Tolstov, G. P. Trigonometric Fourier Series. In *Fourier series*, New York, NY: Dover Publications Inc., 1976, ISBN 0-486-633317-9, pp. 1–40.

[17] Burnham, K. P.; Anderson, D. R. Multimodel Inference: Understanding AIC and BIC in Model Selection. *Sociological Methods & Research*, volume 33, no. 2, 2004: pp. 261–304.

[18] Osborne, J. Improving your data transformations: Applying the Box-Cox transformation. *Practical Assessment, Research, and Evaluation*, volume 15, no. 3, 2010, ISSN 1531-7714.

[19] Komenda, M.; Karolyi, M.; et al. COVID-19: Přehled aktuální situace v ČR. Onemocnění aktuálně [online]. Praha: Ministerstvo zdravotnictví ČR, 2020, [cit. 04 May 2021]. Available from: `https://onemocneni-aktualne.mzcr.cz/api/v2/covid-19`

[20] Ministerstvo vnitra ČR. Vládní opatření lidskou řečí [online]. [cit. 11 May 2021]. Available from: `https://covid.gov.cz/`

[21] Seabold, S.; Perktold, J. statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference*, 2010.

Appendix A

Acronyms

ACF Autocorrelation function

AIC Akaike information criterion

AR Autoregressive

ARMA Autoregressive Moving Average

ARIMA Autoregressive Integrated Moving Average

BIC Bayesian information criterion

KPSS Kwiatkowski–Phillips–Schmidt–Shin

MA Moving average

MAPE Mean Absolute Percentage Error

MLE Maximum Likelihood Estimation

PACF Partial autocorrelation function

Q-Q Quantile-Quantile plot

SARIMA Seasonal Autoregressive Integrated Moving Average

STL Seasonal-trend decomposition

Appendix B

Contents of enclosed CD

- `readme.txt` ..... the file with CD contents description
- `src` ..... the directory of source codes
  - `data` ..... used data sets
  - `functions` ..... used external functions
  - `notebooks` ..... Jupyter notebooks with analysis
  - `thesis` ..... the directory of LaTeX source codes of the thesis
- `texts` ..... the thesis text directory
  - `thesis_Oleh_Kuznetsov.pdf` ..... the thesis text in PDF format