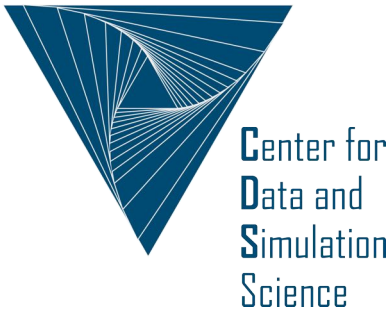


Technical Report Series
Center for Data and Simulation Science

Viktor Grimm, Alexander Heinlein, Axel Klawonn, Martin Lanser, Janine Weber

Estimating the time-dependent contact rate of SIR and SEIR models in mathematical epidemiology using physics-informed neural networks

Technical Report ID: CDS-2020-5
Available at `https://kups.ub.uni-koeln.de/id/eprint/12159`
Submitted on September 22, 2020

ESTIMATING THE TIME-DEPENDENT CONTACT RATE OF SIR AND SEIR MODELS IN MATHEMATICAL EPIDEMIOLOGY USING PHYSICS-INFORMED NEURAL NETWORKS

VIKTOR GRIMM*, ALEXANDER HEINLEIN*†, AXEL KLAWONN*†, MARTIN LANSER*†, AND JANINE WEBER*

September 22, 2020

Abstract. The course of an epidemic can be often successfully described mathematically using compartment models. These models result in a system of ordinary differential equations. Two well-known examples are the SIR and the $SEIR$ models. The transition rates between the different compartments are defined by certain parameters which are specific for the respective virus. Often, these parameters can be taken from the literature or can be determined from statistics. However, the contact rate or the related effective reproduction number are in general not constant and thus cannot easily be determined. Here, a new machine learning approach based on physics-informed neural networks is presented that can learn the contact rate from given data for the dynamical systems given by the SIR and $SEIR$ models. The new method generalizes an already known approach for the identification of constant parameters to the variable or time-dependent case. After introducing the new method, it is tested for synthetic data generated by the numerical solution of SIR and $SEIR$ models. Here, the case of exact and perturbed data is considered. In all cases, the contact rate can be learned very satisfactorily. Finally, the $SEIR$ model in combination with physics-informed neural networks is used to learn the contact rate for COVID-19 data given by the course of the epidemic in Germany. The simulation of the number of infected individuals over the course of the epidemic, using the learned contact rate, is very promising.

Key words. machine learning, physics-informed neural networks, SIR model, SEIR model, epidemic modeling, parameter estimation, COVID-19, SARS-CoV-2, scientific machine learning

AMS subject classifications. 65L09, 68T07, 68T09, 92C60, 92D30

1. Introduction.

In December 2019 the new and previously unknown coronavirus SARS-CoV-2 appeared for the first time in Wuhan, China from where it has spread all over the world. Due to the effects of globalization and a high frequency of private and business travel, the virus and COVID-19, the infectious disease caused by the coronavirus, spread rapidly within Asia and only shortly after in Europe and America. On March 11, 2020, the World Health Organization (WHO) declared the outbreak as a pandemic; see [2].

Mathematical modeling of virus epidemics is a well-known field in applied mathematics and compartment models are well-established approaches to simulate the course of an epidemic within a population. A simple but yet powerful model is the SIR model, which was introduced by Kermack and McKendrick in [14] already in 1927. In the SIR model, the population is divided into three disjoint groups denoted as compartments. The susceptible individuals are denoted by S, the infected and infectious individuals are denoted by I, and the recovered or removed individuals are denoted by R. Then, in the course of the epidemic, there is a transition of the individuals first from the compartment for S to that for I and then from there to the compartment for R; see Figure 1. The flow from one compartment to another is determined by a number of parameters which are characteristic for the modeled

*Department of Mathematics and Computer Science, University of Cologne, Weyertal 86-90, 50931 Köln, Germany, viktor.grimm@uni-koeln.de, alexander.heinlein@uni-koeln.de, axel.klawonn@uni-koeln.de, martin.lanser@uni-koeln.de, janine.weber@uni-koeln.de, url: http://www.numerik.uni-koeln.de

†Center for Data and Simulation Science, University of Cologne, Germany, url: http://www.cds.uni-koeln.de

disease. This model yields a system of three ordinary differential equations (ODEs) where the unknowns are the number of individuals in each compartment. Given the initial values in each compartment, the course of the epidemic can be simulated by numerically solving the system of ODEs. An extension of the SIR model is obtained by dividing the compartment I into two subgroups. Those who are exposed to the virus and thus infected but not yet infectious are gathered in the compartment E, and those who are already infectious are put into the compartment I; see Figure 2. Again, this results in a system of ordinary differential equations for the - now four - unknown numbers of individuals in each compartment. This model is denoted as $SEIR$ model. There exist many excellent introductions and surveys to the field of modeling infectious dieseases using compartment approaches; see, e.g., [13, 6, 7] and the references cited therein.

In practice, modeling the spread of a specific virus using a compartment model requires the knowledge of the parameters defining the flows from one compartment to another. Some of the parameters can be estimated from medical data using statistical methods, some cannot be so easily obtained this way. For instance, the parameter which is denoted as the contact rate β is time-dependent and is usually not so easily estimated directly from the available data; see, e.g., the remark in [6, Section 2.3.2]. Here, parameter indentification methods have to be applied; see, e.g., [21] where numerical methods are discussed to compute the parameters of a dynamical system by a least squares fit.

Here, we present an approach to estimate the time-dependent contact rate in SIR and $SEIR$ models using a new method based on physics-informed neural networks (PINNs). This approach has been introduced in [20] for partial differential equations and constant parameters to be learned. In the present paper, we extend this approach to time-dependent parameters and apply it to compartment models from mathematical epidemiology. The main idea of PINNs is to combine the potential of neural networks to approximate a (nonlinear) functional relation based on data with domain knowledge, here, from epidemic modeling. More precisely, we integrate a priori knowledge in form of ordinary differential equations into the loss function of deep neural networks. The construction of hybrid models, which combine black-box modeling and the use of scientific domain expertise, is one core area of scientific machine learning. In contrast to a number of existing methods, we are especially able to estimate a contact rate which is variable in time. This is achieved by decomposing the global parameter identification problem on the complete time interval into a number of shorter time intervals. For each time interval, we then apply our machine learning techniques and obtain a separate estimate of the contact rate. We verify the potential of our new method by testing it first on several different scenarios with synthetic data for both SIR and $SEIR$ models. In these cases, we generate the synthetic data by numerically solving the systems of ODEs of the SIR and $SEIR$ models, using both exact data and data with noise. Finally, as a proof of concept, we apply our new parameter identification algorithm based on PINNs to estimate the time-dependent contact rate of an $SEIR$ model for the real data of the COVID-19 pandemic in Germany.

The remainder of the paper is organized as follows. In the next section, we provide a brief introduction of the SIR and $SEIR$ compartment models. In section 3, we introduce the parameter identification problem for compartment models in a general form. In section 4, we then first introduce PINNs, describe how to use them for parameter identification, and apply them to SIR and $SEIR$ models. In section 5, we introduce our new approach for learning time-dependent parameters using PINNs by decomposing the problem in shorter time intervals and learning local contact rates.

Next, in section 6, we describe our data preprocessing and the training procedure. Finally, in section 7, we first provide numerical results for a set of experiments with synthetic data. This data has been generated by solving the SIR and $SEIR$ models, respectively, numerically. We use exact data and data with noise to prepare ourselves for realistic data from real epidemics. Then, as a proof of concept for real data, we apply our new method to an $SEIR$ model and data of the COVID-19 pandemic in Germany.

2. Compartment models in mathematical epidemiology. In this section, we provide a brief description of two compartment models common in mathematical epidemiolgy, the SIR and the $SEIR$ model; see [13, 7, 6]. Here, the entire population of a region is divided into a fixed number of suitable groups which are referred to as compartments [14]. For a more detailed introduction to compartment models in mathematical epidemiology, we refer the reader to [13, 7, 6] and the references therein.

2.1. The SIR model. For the SIR model, the population of size N is divided into three disjoint groups or compartments, respectively: susceptible (S), infectious (I), and removed (R) members of the population; see also Figure 1. The individuals are transferred between compartments as indicated in the flow diagram shown in Figure 1 with the transition rates occuring in the differential equation (2.1). In epidemiological compartment models, it is assumed that all individuals in the same group have the same characteristics and thus that all groups are homogeneous. Mathematically, the SIR model is given by the initial value problem

$$\begin{aligned} \frac{dS}{dt} &= -\beta \frac{S\,I}{N} \\ \frac{dI}{dt} &= \beta \frac{S\,I}{N} - \gamma\, I \\ \frac{dR}{dt} &= \gamma\, I \end{aligned} \tag{2.1}$$

with given initial values $S(t_0) \geq 0, I(t_0) \geq 0$, and $R(t_0) \geq 0$ at some initial time t_0. Here, the parameter γ, also denoted as mean infective period, represents the proportion of infected individuals recovering in unit time. Hence, γ can be computed as $1/D$, where D denotes the number of time units an infected person carries and can spread the disease. The parameter β is the contact rate, the number of contacts an average infectious person makes in unit time. Thus, $N \cdot \beta$ is the expected number of people an infected person infects in unit time. The model preserves the total number of individuals, and hence, the functions $S(t), I(t)$, and $R(t)$ satisfy the condition $S(t) + I(t) + R(t) = N$ at any time $t \geq 0$. Let us note that this is a feasible assumption since, in general, an epidemic has a relatively short time scale (compared to the lifespan of an individual) for which new births as well as deaths can be neglected, and additionally, travel restrictions are often enforced during an epidemic.

2.2. The $SEIR$ model. In the basic SIR model, each individual is considered to be infectious as soon as it becomes infected. However, for many epidemic infections, there is a significant exposed period during which an individual that has been infected is not yet infectious itself. In order to incorporate this exposed phase within the SIR model (see subsection 2.1), a fourth compartment for an exposed population is added, denoted by (E). More precisely, the SIR model is extended by letting infected but not yet infectious individuals first move from susceptibles (S) to exposed (E) members of the population. Only after an exposed phase, they are transferred from the exposed

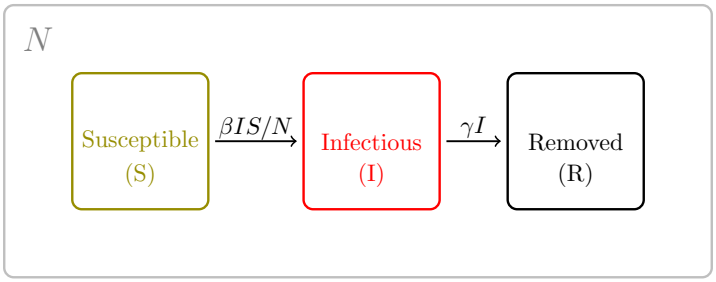


Fig. 1: Schematic representation of the SIR model.

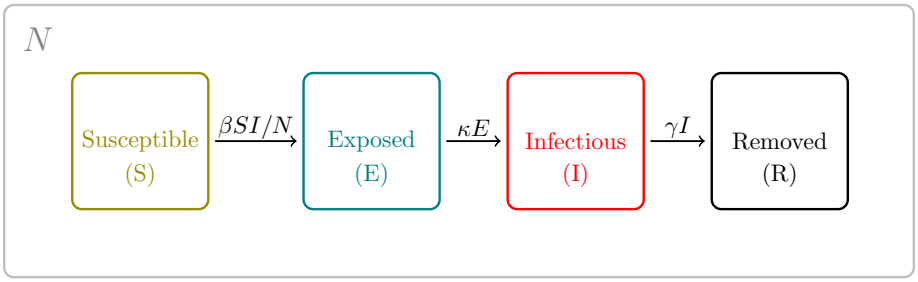


Fig. 2: Schematic representation of the SEIR model.

(E) to the infectious (I) compartment; see also Figure 2. The resulting $SEIR$ model is then given by the initial value problem

$$\begin{aligned} \frac{dS}{dt} &= -\beta \frac{S\,I}{N} \\ \frac{dE}{dt} &= \beta \frac{S\,I}{N} - \kappa\, E \\ \frac{dI}{dt} &= \kappa\, E - \gamma\, I \\ \frac{dR}{dt} &= \gamma\, I \end{aligned} \tag{2.2}$$

with given initial values $S(t_0) \geq 0, E(t_0) \geq 0, I(t_0) \geq 0$, and $R(t_0) \geq 0$. The parameters β and γ are defined as before, in the SIR model. The additional parameter $1/\kappa$ denotes the exposed period, defined as the time each individual spends in the compartment E. In analogy to the SIR model, the $SEIR$ model is also based on the assumption of a constant population size N with $S(t) + E(t) + I(t) + R(t) = N$ at any time $t \geq 0$.

3. Parameter identification problem. The SIR and $SEIR$ compartment models described in section 2 are systems of first-order ordinary differential equations of the general form

$$\frac{\partial U_p}{\partial t}(t) + F_p(U(t)) = 0, \quad t \in [t_0, T] \tag{3.1}$$

with

$$U_p(t) = \begin{pmatrix} u_p^1(t) \\ \vdots \\ u_p^n(t) \end{pmatrix} \quad \text{and} \quad F_p(U) = \begin{pmatrix} f_p^1(U) \\ \vdots \\ f_p^n(U) \end{pmatrix}$$

where $u_p^i \in \mathcal{C}^1(\mathbb{R})$ and $f_p^i \in \mathcal{C}(\mathbb{R})$, $i = 1, \dots, n$. In particular, the function F_p depends on certain parameters $p \in \mathbb{R}^k$, and therefore, the solution U_p depends on p as well. Moreover, t_0 is the initial time and T the final time.

Now, let $p = (p_t, p_f)$ be partitioned into trainable parameters $p_t \in \mathbb{R}^{k_t}$ and fixed parameters $p_f \in \mathbb{R}^{k_f}$ with $k = k_t + k_f$. Furthermore, let $\hat{U}_j$ be given data at times $t_1, \dots, t_M$. Then, we would like to solve the inverse problem

$$\arg\min_{p_t} \frac{1}{M} \sum_{j=1}^{M} \|U_p(t_j) - \hat{U}_j\|^2. \tag{3.2}$$

In other words, we seek to determine the optimal vector of trainable parameters p_t, such that the resulting solution U_p is the best fit to our given data in a least squares sense.

In the context of our epidemiological models, we obtain

$$U_{(\gamma,\beta)} = \begin{pmatrix} S \\ I \\ R \end{pmatrix} \quad \text{and} \quad F_{(\gamma,\beta)} = \begin{pmatrix} -\beta IS/N \\ \beta IS/N - \gamma I \\ \gamma I \end{pmatrix},$$

for the SIR model and

$$U_{(\kappa,\gamma,\beta)} = \begin{pmatrix} S \\ E \\ I \\ R \end{pmatrix} \quad \text{and} \quad F_{(\kappa,\gamma,\beta)} = \begin{pmatrix} -\beta IS/N \\ \beta SI/N - \kappa E \\ \kappa E - \gamma I \\ \gamma I \end{pmatrix}$$

for the $SEIR$ model. Note that the equations for S, E, and I are independent of R, and hence, the last equation can be decoupled in both models. In a post-processing step, for the SIR model, $R(t)$ can then be computed as $R(t) = N - S(t) - I(t)$ for all time points where $S(t)$ and $I(t)$ are known. Using additionally the compartment E, we can proceed analogously for the $SEIR$ model, having $R(t) = N - S(t) - E(t) - I(t)$. Hence, we only consider the reduced systems

$$U_{(\gamma,\beta)} = \begin{pmatrix} S \\ I \end{pmatrix} \quad \text{and} \quad F_{(\gamma,\beta)} = \begin{pmatrix} -\beta SI/N \\ \beta SI/N - \gamma I \end{pmatrix}, \tag{3.3}$$

and

$$U_{(\kappa,\gamma,\beta)} = \begin{pmatrix} S \\ E \\ I \end{pmatrix} \quad \text{and} \quad F_{(\kappa,\gamma,\beta)} = \begin{pmatrix} -\beta SI/N \\ \beta SI/N - \kappa E \\ \kappa E - \gamma I \end{pmatrix}, \tag{3.4}$$

respectively, in the inverse problem (3.2).

In practice, the recovery rate γ and the exposure rate κ can be assumed to be roughly constant for all individuals and in time. They can typically be estimated based on statistical data and are often available in the medical literature. On the other

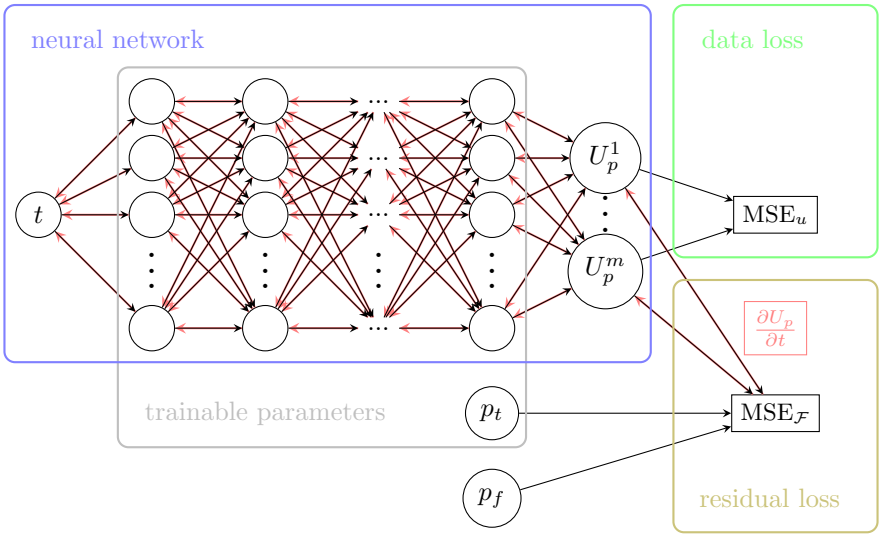


Fig. 3: Graphical representation of a PINN. The derivatives needed for the evaluation of the residual are computed using automatic differentiation in a backward propagation (red). The parameters to identify p_t are trainable parameters of the neural network, the parameters p_f are fixed.

hand, the contact rate β is subject to many external influences such as governmental policies (e.g., school/university closures, lockdown, social distancing, etc.) or changing behavior of the population (life style, hygiene standards, etc.) and thus can vary significantly over time. Hence, we will use values of γ and κ from the literature, and therefore, fix those parameters, whereas we select β as a trainable parameter. In particular, in our set of numerical experiments, we will consider both cases: a constant β and a $\beta(t)$ which varies over time in the inverse problem

$$\arg\min_{\beta} \frac{1}{M} \sum_{j=1}^{M} \|U_p(t_j) - \hat{U}_j\|^2.$$

As a discretization for the SIR and the $SEIR$ models, we will apply physics-informed neural networks.

4. Physics-informed neural networks. The basic idea of physics-informed neural networks (PINNs) is to integrate a priori knowledge in form of physical laws or domain expertise modeled by ordinary or partial differential equations into a deep learning model. In particular, this is done by differentiating neural networks with respect to their input variables and model parameters; see [20] for more details. Then, in addition to the data error, the residual of the differential equation is minimized in a least squares sense as part of the loss function.

4.1. Discretizing systems of ODEs using PINNs. In order to solve (3.2), we first discretize (3.1) using a PINN and then solve the inverse problem simultaneously while training the neural network. A PINN for discretizing (3.1) is based on a standard neural network

$$\mathcal{NN}_p^{W,b} : \mathbb{R} \to \mathbb{R}^n \tag{4.1}$$

approximating the solution

$$U_p : \mathbb{R} \to \mathbb{R}^n$$

of the system of first-order ODEs. Here, the superscript indices indicate that the neural network is determined by its weights W and biases b. In particular, we use a dense neural network with the scaled exponential linear unit (SELU) [16] activation function; this network is marked as *neural network* in Figure 3. In order to discretize (3.1) by the neural network $\mathcal{NN}_p^{W,b}$, the weights W and biases b can now be optimized such that $\mathcal{NN}_p^{W,b}$ fits the data $\hat{U}_j$, $j = 1, \dots, M$, in a least squares sense, solving

$$\arg\min_{W,b} \underbrace{\frac{1}{M} \sum_{j=1}^{M} \|\mathcal{NN}_p^{W,b}(t_j) - \hat{U}_j\|^2}_{=:\mathrm{MSE}_U^{W,b}} . \tag{4.2}$$

In order to extend $\mathcal{NN}_p^{W,b}$ to a PINN, we enhance the loss function $\mathrm{MSE}_U^{W,b}$ from (4.2) by the additional term

$$\mathcal{F}_p(\mathcal{NN}_p^{W,b}, t) = \frac{\partial \mathcal{NN}_p^{W,b}}{\partial t}(t) + F_p(\mathcal{NN}_p^{W,b}(t)),$$

which corresponds to the residual of our system of ODEs (3.1). The evaluation of the residual requires the computation of the time derivative of the neural network output $\frac{\partial \mathcal{NN}_p^{W,b}}{\partial t}$. This can be done using the backward propagation algorithm and automatic differentiation [4], which is also the standard algorithm used to compute gradients in deep neural networks in gradient-based optimization schemes.

Now, satisfying

$$\mathcal{F}_p(\mathcal{NN}_p^{W,b}, t) = 0 \quad \forall t \in [t_0, T] \tag{4.3}$$

is equivalent to $\mathcal{NN}_p^{W,b}$ solving (3.1) exactly. Since, in practice, we cannot enforce (4.3) exactly on the entire interval $[t_0, T]$, we enforce this condition at certain points in time $t^1, \dots, t^L$ in a least squares sense; we also call these *collocation points*. More precisely, we add the mean squared residual error

$$\mathrm{MSE}_{\mathcal{F}_p}^{W,b} := \frac{1}{L} \sum_{j=1}^{L} \|\mathcal{F}_p(\mathcal{NN}_p^{W,b}, t^j)\|^2 \tag{4.4}$$

to our loss function (4.2). In summary, the neural network $\mathcal{NN}_p^{W,b}$ of the form (4.1) becomes a PINN by simply adding the term (4.4) to the loss function. Hence, a PINN discretizing (3.1) is obtained solving the minimization problem

$$\arg\min_{W,b} \left(\mathrm{MSE}_U^{W,b} + \mathrm{MSE}_{\mathcal{F}_p}^{W,b}\right). \tag{4.5}$$

4.2. Choosing the data and collocation points. Even though, the points in time $t_1, \dots, t_M$, where data are available, and the collocation points $t^1, \dots, t^L$ are, in principle, independent of each other, a common choice is $\{t^1, \dots, t^L\} = \{t_1, \dots, t_M\}$. Later, we will also consider the case $\{t^1, \dots, t^L\} \supset \{t_1, \dots, t_M\}$. In order to ensure that, in a batch stochastic gradient algorithm (SGD), t_i and t^j are always in the same

batch if $t_i = t^j$, we reformulate (4.5) as follows. In particular, we choose a single set of points in time $\{t_1, \dots, t_N\} := \{t^1, \dots, t^L\} \cup \{t_1, \dots, t_M\}$ and introduce, for each specific t_i, binary data and residual flags α_i^d and α_i^r, respectively. These flags specify which loss terms are enforced for time t_i. Consequently, we use a modified loss function to obtain a minimization problem equivalent to (4.5)

$$\arg\min_{W,b} \sum_{j=1}^{N} \left(\frac{\alpha_j^d}{M_d} \|\mathcal{NN}_p^{W,b}(t_j) - \hat{U}_j\|^2 + \frac{\alpha_j^r}{M_r} \|\mathcal{F}_p(\mathcal{NN}_p^{W,b}, t_j)\|^2 \right), \tag{4.6}$$

where $M_d = \sum_{j=1}^{N} \alpha_j^d$ and $M_r = \sum_{j=1}^{N} \alpha_j^r$. Note that, in this paper, we will only consider the cases $\alpha_j^d = \alpha_j^r = 1$ or $\alpha_j^d = 0$ and $\alpha_j^r = 1$. In other words, the ODE residual (4.3) is always enforced in all points of time, even if no data fit is enforced. For the sake of brevity, we did not include a full study on the influence of α_j^d and α_j^r on the training performance of the PINN.

4.3. Parameter identification using PINNs. Now, in order to solve the inverse problem (3.2), we release the parameters p_t in the minimization problem (4.6) and obtain

$$\arg\min_{W,b,p_t} (\mathrm{MSE}_U^{W,b} + \mathrm{MSE}_{\mathcal{F}_p}^{W,b}). \tag{4.7}$$

Therefore, we do not solve the discrete inverse problem exactly but in a weak sense.

The optimization of (4.7) may be improved by introducing a weighting factor ω which balances the loss terms, yielding

$$\arg\min_{W,b,p_t} (\mathrm{MSE}_U^{W,b} + \omega \mathrm{MSE}_{\mathcal{F}_p}^{W,b}); \tag{4.8}$$

see subsection 6.3 for more details on choosing ω.

4.4. Application to SIR and $SEIR$ models. In this section, we apply PINNs in order to solve the inverse problem (3.2) for the SIR and $SEIR$ models using temporal epidemiological data for the number of susceptible, exposed, and infected individuals. Here, we consider the SIR and $SEIR$ models as systems of ODEs and neglect the removed individuals as described in section 3. In particular, we consider (3.3) and (3.4) for the SIR model and the $SEIR$ model, respectively. The respective PINN is obtained by training a neural network (4.1) on this data using the composed loss function (4.6).

SIR model. For the SIR model, we employ a dense neural network

$$\begin{aligned} \mathcal{NN}_p^{W,b} : \mathbb{R} &\to \mathbb{R}^2 \\ t &\mapsto \begin{pmatrix} \mathcal{S}_p^{W,b} \\ \mathcal{I}_p^{W,b} \end{pmatrix} =: \begin{pmatrix} \mathcal{S} \\ \mathcal{I} \end{pmatrix} \end{aligned}$$

approximating $(S(t), I(t))^T$, where S and I follow the SIR model; cf. (3.3) and subsection 2.1. In order to enforce this in a least squares sense, we minimize the mean residual error $\mathrm{MSE}_{\mathcal{F}_p}^{W,b}$ defined in (4.4), where

$$\mathcal{F}_p(\mathcal{NN}_p^{W,b}, t) = \begin{pmatrix} \frac{\partial \mathcal{S}}{\partial t}(t) & -\beta \frac{\mathcal{S}(t)\mathcal{I}(t)}{N} \\ \frac{\partial \mathcal{S}}{\partial t}(t) & +\beta \frac{\mathcal{S}(t)\mathcal{I}(t)}{N} - \gamma \mathcal{I}(t) \end{pmatrix}.$$

At the same time, we want to minimize the mean squared data error

$$\mathrm{MSE}_U = \mathrm{MSE}_S + \mathrm{MSE}_I$$

where

$$\mathrm{MSE}_S = \frac{1}{M}\sum_{i=1}^{M}\left(\mathcal{S}_p^{W,b}(t_i) - \hat{S}^i\right)^2, \quad \mathrm{MSE}_I = \frac{1}{M}\sum_{i=1}^{M}\left(\mathcal{I}_p^{W,b}(t_i) - \hat{I}^i\right)^2,$$

and $\hat{S}^i$ and $\hat{I}^i$ is given data at time t_i. Please refer to subsection 6.3 for additional information on where we minimize the residual error and how we select the collocation points. Then, the inverse problem

$$\arg\min_{W,b,\beta}\left(\mathrm{MSE}_U^{W,b} + \mathrm{MSE}_{\mathcal{F}_p}^{W,b}\right)$$

is solved, and we obtain the model parameters W and b as well as an estimate of the contact rate β.

SEIR model. The extension to the $SEIR$ model is straightforward. In particular, we consider the neural network

$$\begin{aligned}\mathcal{NN}_p^{W,b} : \mathbb{R} &\to \mathbb{R}^2\\ t &\mapsto \begin{pmatrix}\mathcal{S}_p^{W,b}\\ \mathcal{E}_p^{W,b}\\ \mathcal{I}_p^{W,b}\end{pmatrix} =: \begin{pmatrix}\mathcal{S}\\ \mathcal{E}\\ \mathcal{I}\end{pmatrix}\end{aligned}$$

approximating $(S(t), E(t), I(t))^T$; see also (3.4) and subsection 2.2. Then, we again minimize the loss function

$$\arg\min_{W,b,\beta}\left(\mathrm{MSE}_U^{W,b} + \mathrm{MSE}_{\mathcal{F}_p}^{W,b}\right),$$

where the residual is given by

$$\mathcal{F}_p(\mathcal{NN}_p^{W,b}, t) = \begin{pmatrix} \frac{\partial \mathcal{S}}{\partial t}(t) & -\beta\frac{\mathcal{S}(t)\mathcal{I}(t)}{N} \\ \frac{\partial \mathcal{S}}{\partial t}(t) & +\beta\frac{\mathcal{S}(t)\mathcal{I}(t)}{N} - \kappa\mathcal{E}(t) \\ \frac{\partial \mathcal{S}}{\partial t}(t) & +\kappa\mathcal{E}(t) - \gamma\mathcal{I}(t), \end{pmatrix}$$

and the mean squared data error is the sum of

$$\begin{aligned}\mathrm{MSE}_S &= \frac{1}{M}\sum_{i=1}^{M}\left(\mathcal{S}_p^{W,b}(t_i) - \hat{S}^i\right)^2,\\ \mathrm{MSE}_E &= \frac{1}{M}\sum_{i=1}^{M}\left(\mathcal{E}_p^{W,b}(t_i) - \hat{E}^i\right)^2, \text{ and}\\ \mathrm{MSE}_I &= \frac{1}{M}\sum_{i=1}^{M}\left(\mathcal{I}_p^{W,b}(t_i) - \hat{I}^i\right)^2.\end{aligned}$$

Let us note that solving the inverse problem (4.4) is denoted as data-driven discovery of differential equations in [20]. In this work, our aim is to use such data-driven discovery, i.e., parameter identification, and PINNs to estimate the the contact rate

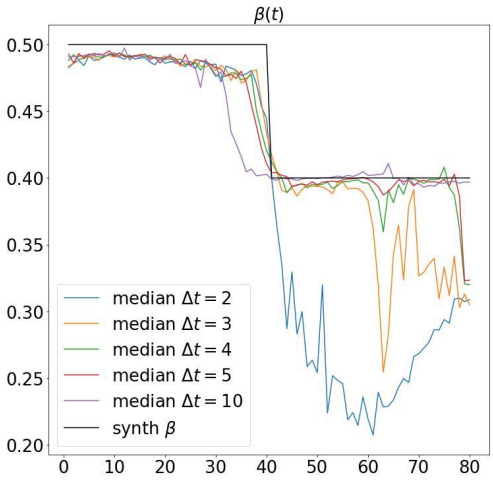


Fig. 4: Estimates for $\beta(t)$ for an SIR model for a synthetic contact rate with a sharp jump (marked in black). We show comparative results for different timeframe lengths, i.e., $\Delta t \in \{2, 3, 4, 5, 10\}$ and 10 runs per timeframe.

β of an SIR model or an $SEIR$ model for the COVID-19 pandemic in the year 2020. In particular, using PINNs for the data-driven discovery, i.e., parameter estimation, of ordinary differential equations means that we use PINNs for the solution of an inverse problem. In contrast to the more common solution of forward problems, here, we do not have a typical offline training phase and a subsequent online testing phase for the neural network. Instead, when solving an inverse problem, we use separate neural networks for each course of the parameter $\beta(t)$ which are explicitly trained using the corresponding training data for the compartments S, E, and I. Let us note that in contrast to [20], we will establish a procedure that enables us to estimate a time-dependent parameter $\beta(t)$ in (2.1). This is a clear novelty compared to [20] where the data-driven discovery of partial differential equations is used to estimate a model parameter which remains constant in time. Here, our main idea is to partition the entire interval $[t_0, T]$ for the parameter identification into a number of shorter time intervals. Within each of these time intervals, we assume a constant contact rate β for which we compute an estimate using the machine learning approach described in this section. In particular, our proposed method can trivially be executed in parallel for all time intervals and can thus be relatively quickly updated any time new data becomes available. Subsequently, we assemble the different estimates for β for all time intervals and thus obtain an estimated contact rate $\beta(t)$ which is variable in time. Please refer to section 5 for more details.

5. Learning a time-dependent contact rate. In the inverse problems introduced in subsection 4.4, we have assumed a constant contact rate β in the SIR and $SEIR$ models. This is the simplest case, which is, in reality, an inappropriate assumption most of the time. In particular, the contact rate β depends on several external factors and circumstances, such as governmental policies (e.g., school/university closures, lockdown, social distancing, etc.) or changing behavior of the population (life style, hygiene standards, etc.). Therefore, it can vary significantly over time such that using a constant β is not appropriate for modeling the entire course of the on-

going pandemic. Instead, we now consider a time-dependent contact rate $\beta(t)$. This is an extension of the parameter estimation in [20], where solely a constant model parameter is estimated using PINNs.

In order to learn a time-dependent parameter $\beta(t)$ at a given time t_i, we consider a short time interval containing t_i; we denote this short time interval as a *timeframe*. More precisely, we define the timeframe tf_i corresponding to t_i as

$$tf_i = [t_i - \Delta t, t_i + \Delta t] \cap [t_0, T], \tag{5.1}$$

where Δt determines the timeframe length $2\Delta t$; in practice, we only consider full days, i.e., $t_i \in \mathbb{Z}$ and $\Delta t \in \mathbb{N}$. Under the assumption that $\beta(t)$ does not vary drastically on each timeframe tf_i, we can approximate $\beta(t)$ on tf_i by a constant β_i and solve the corresponding inverse problem (3.2) on that timeframe. Using this approach, we can find values β_i approximating $\beta(t_i)$ for any $t_i \in [t_0, T]$, and thus, for a whole time grid discretizing $[t_0, T]$. Note that the problems on the timeframes are completely local and can be computed independently of each other.

For stability reasons, we train a batch of separate PINNs estimating $\beta(t_i)$ on each timeframe tf_i instead of just a single PINN. To finally obtain β_i, we compute the median of all estimates of all PINNs in the batch corresponding to tf_i. Let us note that we deliberately have chosen the median instead of the mean of all resulting estimations, since the median is more robust with respect to outliers and thus a more reliable approximation for $\beta(t)$. Furthermore, the median is always an actual estimate generated by one of the networks, in contrast to the mean.

Let us briefly comment on the effect of the specific value of Δt on the estimate of the contact rate β. We can observe that with an increasing Δt, the smoothing effect with respect to short-term variations in the parameter β increases. This is caused by our assumption that $\beta(t)$ is constant on each local timeframe. The importance of choosing an appropriate value for Δt is clearly visible in Figure 4. If Δt is too small, we lack sufficient data about the development of the epidemic to obtain robust predictions from training the PINNs; this is visible for the cases $\Delta t = 2, 3$, where the estimate of β is highly volatile in the second half of the course of $\beta(t)$ and the respective results are thus not feasible. On the other hand, if Δt is too large, the estimate of β is fairly inaccurate in the area of the sharp jump for $\beta(t)$ and we are unable to replicate the sudden change in the values of β; cf. the purple line corresponding to $\Delta t = 10$ in Figure 4. Additionally, an increasing timeframe length is directly related to an increasing computational effort as more training data are used for each timeframe. Therefore, in our computations, we aim at using a rather small Δt while still obtaining satisfying estimates. For the computations in Figure 4, the choices $\Delta t = 4, 5$ both provide satisfying estimates, obtaining a good trade-off between the two above mentioned aspects. For our numerical computations in section 7, we have decided to use $\Delta t = 5$.

6. Data preprocessing and training procedure. In this section, we describe the different steps of our training procedure for the parameter estimation for epidemic models using PINNs; cf. sections 4 and 5. In particular, training a PINN to estimate the contact rate β of an epidemic model is not straightforward and cannot be done out-of-the-box. It requires, amongst others, an appropriate scaling of the training data as well as of the loss terms. Moreover, since the number of data points is rather low for our timeframes, we use an additional set of collocation points.

6.1. Rescaling the data. In preliminary experimental results, we have noticed that extreme differences in the magnitude of the training data for S and I or S, E,

and I, respectively, can deteriorate the convergence in the optimization process of a PINN. Therefore, before starting the iterative training, we always apply a logarithmic scaling followed by a min-max scaling. The min-max scaling is applied separately for each compartment. As a result, all training data only take on values in $[0, 1]$, which is considered to be beneficial for the convergence of the model parameters W and b of a neural network; see, e.g, [5, Sect. 8.1] or [17].

Let us note that we reverse both types of scalings in the forward call of the PINN. Otherwise, we would have to explicitly consider the scaling of the training data when computing the derivatives for the residual terms in each step of the training procedure. To obtain standardized magnitudes of the residuals we further divide them by the size of the population N.

6.2. Additional collocation points. As described subsection 4.2, we do not necessarily use the same points of time to fit the data and the residuals. In particular, due to the rather small Δt (cf. section 5) and the resulting low number of data points within each timeframe, we use additional collocation points in order to improve the robustness of the training. More precisely, we evaluate the residual in the original data points, i.e., at each full day, and in eight additional, equidistant points in between. Hence, the number of collocation points is approximately nine times larger than the number of data points.

6.3. Balancing the loss terms. In addition to the aforementioned data preprocessing, we also incorporate certain scaling strategies in order to balance the loss terms in (4.5). Let us note that all described techniques are equally valid for both the SIR as well as the $SEIR$ model.

Weight routines. As explained in section 4, the loss function of our PINN consists of two components, the data loss $\mathrm{MSE}_U^{W,b}$ and the residual loss $\mathrm{MSE}_{\mathcal{F}_p}^{W,b}$. In order to control the ratio of these two components, we introduced a weighting factor ω for the residual-loss; cf. (4.5) or (4.8).

We divide the training process into several phases, which are defined as follows. In each phase of the training process marked with index i, we choose a specific value for the weighting factor ω_i in (4.5) or (4.8). Thus, for the first phase in our training process, we always set $\omega_1 = 0$, exclusively minimizing the loss function with respect to the data. For the subsequent phases of the training, we found the following procedure to be a good choice: we set $\omega_2 = 10^{-2}$ and then reduce ω_i recursively by a factor of 10 until we obtain $\omega_{13} = 10^{-13}$ in the last phase. In each of the resulting 13 phases, we train the neural networks for 500 epochs using an early stopping criterion ([18] or [12, Sect. 7.8]) with a patience of 10 epochs with respect to the total loss (4.5).

α-scaling. In preliminary experiments, we have noticed a strong tendency to underestimate the parameter $\beta(t)$ on "flat" parts of the curves for S, $(E,)$ and I. When examining the different components of the loss function, we observed a strong correlation between these "flat" parts and significantly low residual loss terms for the respective variables. In order to align the magnitudes of the data and the residual loss, we adaptively scale the residual loss to the same order of magnitude as the data loss. In particular, after the first phase of our training process, we compute the ratio

$$\alpha := \frac{\mathrm{MSE}_{\mathcal{F}_p}^{W,b}}{\mathrm{MSE}_U^{W,b}} \tag{6.1}$$

and multiply all subsequent weights ω_i by this ratio α. We denote this process by α-scaling for the remainder of this paper.

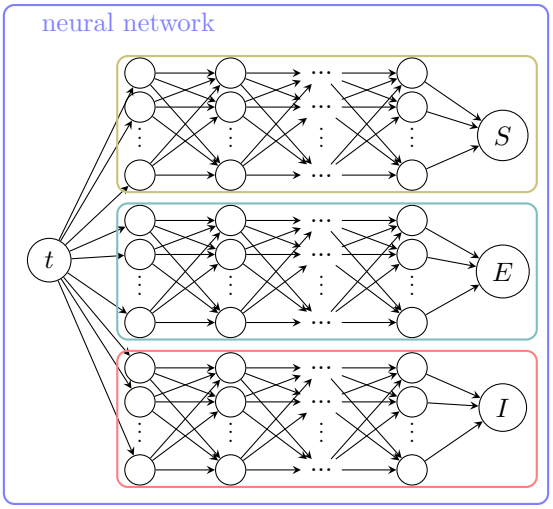


Fig. 5: PINN for the SEIR model: the neural network is partitioned into three decoupled networks discretizing S, E, and I. For the SIR model, the E is omitted.

7. Numerical results. In this section, our aim is to extract the presumably unknown parameter $\beta(t)$ from daily data for S, E, and I. As the neural network of our PINN for the SIR and $SEIR$ models, we use a neural network that is composed of two or three, respectively, decoupled neural networks discretizing S, E, and I; see Figure 5 for a graphical representation of the network structure. In particular, we use a dense feed-forward neural network with ten layers, consisting of 100 neurons per layer for each variable. As the activation function we use the scaled exponential linear unit (SELU) [16] in combination with the LeCun normal initialization [17]. Furthermore, we use a batch size of one and a learning rate of 1e-7 to minimize the loss function with the Adam optimizer [15]. For stability reasons, we always train 5 PINNs estimating β for each timeframe and use the obtained median as the final estimate β_i for the respective timeframe; cf. also section 5 and Figure 22 at the end of this section for an exemplary investigation of the robustness of our method. All computations were performed on NVIDIA V100-GPUs with CUDA 10.1 using python 3.6 and tensorflow-gpu 2.3 [3] in double precision.

7.1. Synthetic data with constant contact rate. For our first numerical experiments, we apply our proposed parameter identification procedure to the simplest case possible, which is assuming a constant contact rate β over the entire course of the epidemic. For this purpose, we have generated synthetic data as a solution of the system of ordinary differential equations (ODEs) of the SIR model (2.1) for realistic and constant values of the model parameters β and γ. We solve the ODEs using an adaptive Runge-Kutta-4-5 method provided by the python package scipy [22] with initial values $S(0)$ and $I(0)$. Specifically, we have used the initial values $I(0) = 50$, $R(0) = 0$ and the parameters $N = 80\,000\,000$, $\beta = 0.5$, and $\gamma = \frac{1}{4}$ in our simulation. The resulting curves for S and I are shown in Figure 6.

As a first sanity check to prove that our proposed method works appropriately, we consider the inverse problem with constant contact rate β as described in subsection 4.4 on the whole interval $[t_0, T]$ at once. Hence, we train a batch of five PINNs

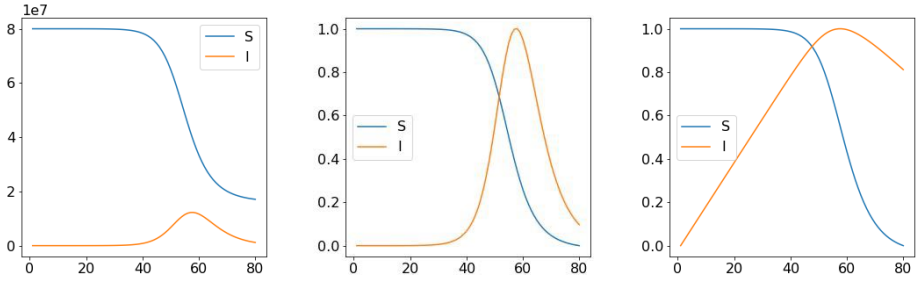
estimating β and compute the final estimate for β as the median value. We obtain the final constant estimate 0.5019 compared to the true value $\beta = 0.5$. This corresponds to a relative error of 3.8e-3 and is thus fairly accurate. The resulting curves for the obtained simulation for I using the learned estimate of β as well as the respective training data are shown in Figure 7. We additionally show the effective reproduction number R_t in Figure 7; see, e.g., [10] for a definition of R_t. For any time $t \geq t_0$, it can be computed as

$$R_t = \frac{\beta(t)}{\gamma} * \frac{S(t)}{N}.$$

As we can observe from Figure 7, our computed estimate of R_t (denoted by *learned* R_t) is almost identical to the actual values of R_t (denoted by *synth* R_t).

In the next set of experiments, we already apply our proposed procedure for learning a time dependent contact rate using timeframes of smaller length; see section 5. However, for the SIR model, we use the same training data as before; see Figure 6. To generate a second data set for S, E, and I, we perform simulations using the $SEIR$ model and the parameters $\gamma = 1/11$, $\kappa = 1/3$ and $\beta = 0.5$. This means that, for both data sets, we learn a time-dependent contact rate $\beta(t)$ which, in the training data, is actually constant in time. For all presented results, we have used $\Delta t = 5$. We show the learned estimates of β and the resulting simulated curves for I, using the estimates β, as well as the resulting estimates for R_t in Figure 8 and Figure 9 for the SIR model and the $SEIR$ model, respectively. As we can observe from both figures, our estimate of β is sufficiently good, since both the SIR and the $SEIR$ model deliver curves for I and R_t which are fairly close to the synthetic training data. However, the obtained simulation using the $SEIR$ model is more accurate compared to the SIR model. Thus, even if we allow for a time-dependent contact rate in our proposed procedure using timeframes, we are able to recover the constant contact rate β quite accurately. Let us remark that, for the first Δt days and the final Δt days, the length of the timeframe is always shorter than the usual timeframe length of $2\Delta t+1$; see (5.1) for the definition of a timeframe. Using a short timeframe, the estimate of $\beta(t)$ can sometimes be deteriorated; see also Figure 4 and the corresponding discussion in section 5 on different timeframe lengths. Hence, in some of our experiments, the etimates of $\beta(t)$ for the first and/or last few days are deteriorated due to the reduced timeframe length; see, e.g., the last few days in Figure 9 or Figure 14 for this effect. We will not further discuss this effect for all following examples explicitly to avoid redundant discussions.

As a next step, we investigate the quality of the parameter identification for synthetic data when the data are perturbed. For this purpose, we have added a normally distributed noise ε with mean zero to the original values I for the infectives for both the SIR and the $SEIR$ model as well as to the original values E for the exposed for the $SEIR$ model. More precisely, we have added the term $\varepsilon * (I(t)/15)$ and $\varepsilon * (E(t)/15)$ to each value of $I(t)$ and $E(t)$ with $t \in [t_0, T]$ respectively. The respective pertubed training data for I and the simulation results obtained when using the learned estimates of β, together with the corresponding estimated values of R_t are presented in Figure 10 for the SIR model and in Figure 11 for the $SEIR$ model, respectively. For both epidemic models, we obtain satisfactory results in terms of the parameter estimation of β. Thus, our proposed method seems to be robust for data with and without inaccuracies in the reported numbers of the infectives, when assuming a constant contact rate β. Again, the resulting estimates for β and R_t are even more accurate when using the $SEIR$ model compared to the SIR model.



(a) Raw Data (b) Min-Max-scaled Data (c) Min-Max-scaled ln(Data)

Fig. 6: Synthetic training data for the SIR model for $N = 80\,000\,000$, $I(0) = 50$, $R(0) = 0$, $\beta = 0.5$, $\gamma = \frac{1}{4}$.

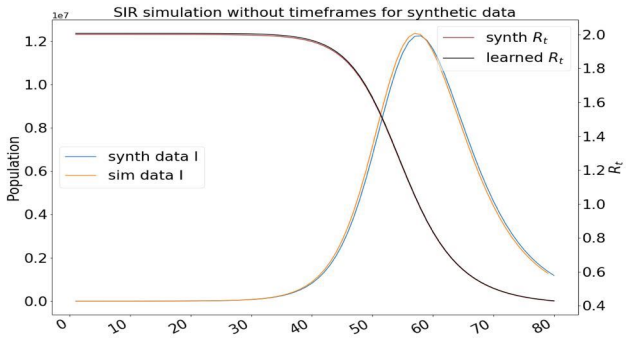


Fig. 7: Simulation result considering the SIR model using the constant learned estimate 0.5019 without timeframes ($tf_0 = [t_0, T]$) and the corresponding synthetic training data. The obtained estimate of β corresponds to a relative error of 3.8e-3.

7.2. Synthetic data with time-dependent contact rate. For the numerical experiments considered so far, the contact rate β was always constant in time. However, for real-world applications, using real data of the Corona pandemic in 2020, we need to be able to approximate values for a parameter $\beta(t)$ which varies over time. For this purpose, we explicity integrated the procedure of using timeframes into our workflow; see section 5.

7.2.1. Discontinuous contact rate with jumps. For the first set of numerical experiments with a time-dependent contact rate, we construct a synthetic training dataset that differs from the dataset in Figure 6 predominantly in the values and the course of $\beta(t)$. In particular, we now use a contact rate $\beta(t)$ with a single jump defined by

$$\beta(t) = \begin{cases} 0.5 & t \leq 40 \\ 0.4 & \text{elsewhere} \end{cases} \tag{7.1}$$

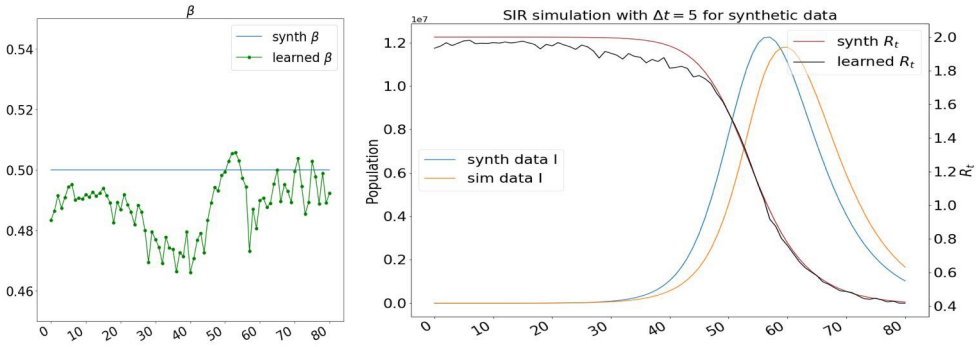


Fig. 8: Results for the SIR model with a constant contact rate $\beta = 0.5$. **Left:** Estimate of β with $\Delta t = 5$. **Right:** Comparison of simulation results for the SIR model using the learned estimates of β per timeframe (denoted by *sim data I*) or using β itself (denoted by *synth data I*), i.e., using the *learned* β versus the *synth* β from the left figure. We further show the computed estimates of R_t per timeframe (denoted by *learned* R_t) versus R_t for the training data (denoted by *synth* R_t)

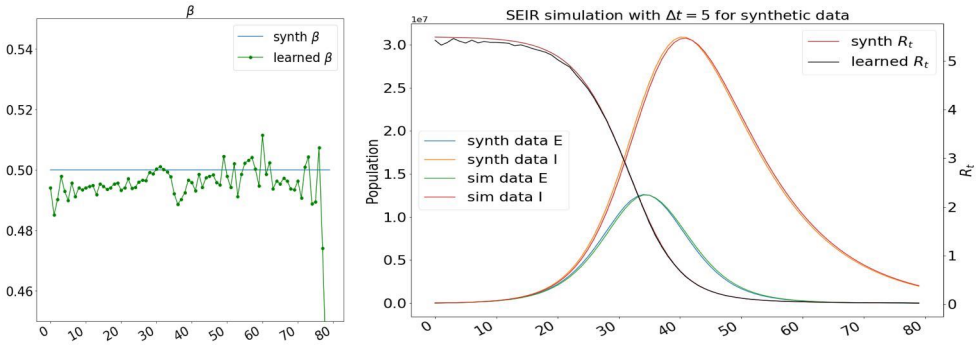


Fig. 9: Results for the $SEIR$ model with a constant contact rate $\beta = 0.5$. **Left:** Estimate of β with $\Delta t = 5$. **Right:** Comparison of simulation results for the $SEIR$ model using the learned estimates of β per timeframe or using β itself, i.e., using the *learned* β versus the *synth* β from the left figure. See Figure 8 for the labeling.

for both the SIR and the $SEIR$ model. We obtain the corresponding new training datasets by running simulations of the epidemic models (see subsection 2.1 and subsection 2.2) with the initial values $N = 80\,000\,000$, $I(0) = 50$, $R(0) = 0$, and the parameters $\gamma = \frac{1}{4}$ for the SIR model and $\gamma = 1/11$, $\kappa = 1/3$ for the $SEIR$ model, respectively. The resulting curves for S and I for the SIR model are presented in Figure 12. Let us note that the discontinuity in the contact rate $\beta(t)$ leads to a notable kink in the curves for S and I at the time interval $40 < t < 41$. This kink is particularly visible in the plot of the logarithmized and min-max-scaled data in Figure 12b. We show the learned estimates of β for all timeframes in Figure 13 (left) and in Figure 14 (left) in comparison to the true value of β (see (7.1)) for the SIR model and the $SEIR$ model, respectively. For both models we can observe that in the area around the jump, we obtain a 'smoothed' estimate of the contact rate $\beta(t)$. Please

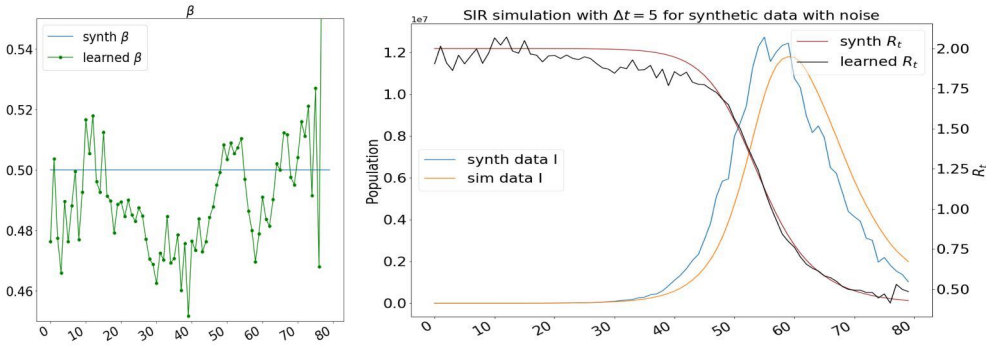


Fig. 10: Results for the SIR model with a constant contact rate $\beta = 0.5$ and normally distributed noise. **Left:** Estimate of β with $\Delta t = 5$. **Right:** Comparison of simulation results for the SIR model using the learned estimates of β per timeframe or using β itself, i.e., using the *learned* β versus the *synth* β from the left figure. See Figure 8 for the labeling.

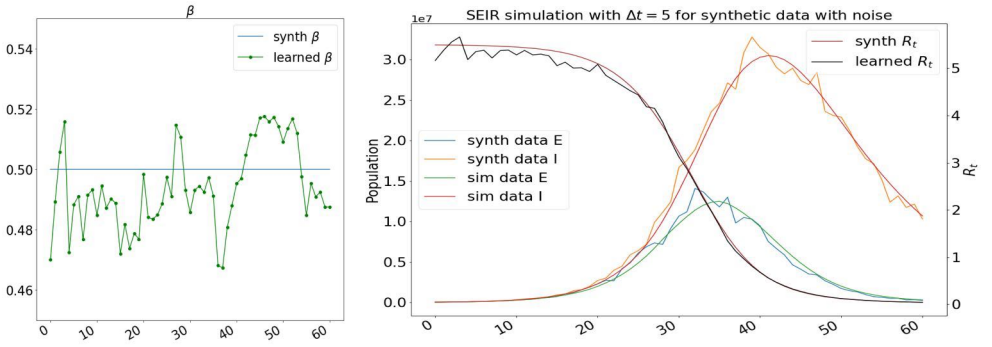
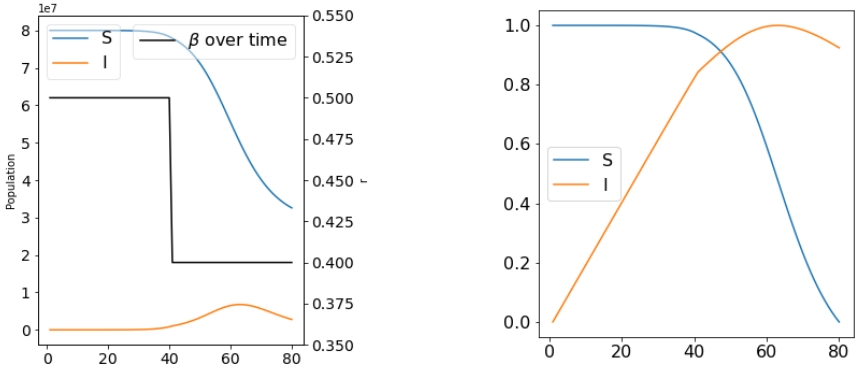


Fig. 11: Results for the $SEIR$ model with a constant contact rate $\beta = 0.5$ and normally distributed noise. **Left:** Estimate of β with $\Delta t = 5$. **Right:** Comparison of simulation results for the $SEIR$ model using the learned estimates of β per timeframe or using β itself, i.e., using the *learned* β versus the *synth* β from the left figure. See Figure 8 for the labeling.

refer to section 5 and especially to Figure 4 for a more detailed analysis of this effect. Additionally, we present the obtained simulated values for the infectives I and the exposed E in Figure 13 (right) and Figure 14 (right), respectively, when using the corresponding estimates of β as generated by the PINNs. In both cases, the resulting curves are fairly close to the true training data. Again, using the $SEIR$ model yields a more accurate parameter estimate than using the SIR model.

In correspondence to subsection 7.1, we additionally perform our parameter identification procedure on perturbed synthetic training data. In particular, we generate the noise, which is added to the number of infectives and exposed, for both the SIR and the $SEIR$ model in exactly the same way as in subsection 7.1. The results obtained by training the PINNs for the perturbed training data are shown in Figure 15 and Figure 16. For both epidemic models, our proposed approach is clearly able to

provide a satisfactory estimate of the time-dependent contact rate β, even for an perturbed training data set.



(a) Raw Data (b) Min-Max-scaled ln(Data)

Fig. 12: Synthetic training data for the SIR model for $N = 80\,000\,000$, $I(0) = 50$, $R(0) = 0$, $\gamma = \frac{1}{4}$ and discontinuous $\beta(t)$ as defined in (7.1).

7.2.2. Synthetic data for a second wave. As a more realistic but still synthetic example for the development of an epidemic disease, we tested our procedure for synthetic data which emulate a second wave; see [7, Section 9.6.2]. This means that, after a temporary decrease of the number of infectives and an intermediate rather flat part of the curve for I, the absolute number of infectives increases again. This kind of periodic behavior can be observed for real-world pandemics, e.g., as for the flu outbreak in 1918 ("Spanish flu"); see, e.g., [7, Section 9.6.2]. To model such a behavior for the infectives I, we use an extended approach to compute the time-dependent reference value for the contact rate $\beta(t)$ for the SIR and the $SEIR$ model inspired by the periodic function given in [7, Sect. 9.6.2].

Specifically, for our experiments with the SIR model, we compute the contact rate $\beta(t)$ as

$$\beta(t) = \beta\big(1 + c \cdot \cos(\pi(t + t_0)/45)\big); \tag{7.2}$$

with the parameters $N = 10\,000$, $I(0) = 150$, $t_0 = 11, c = 0.45$, and $\beta = 0.35$. We further set the parameter γ in (2.1) as $\gamma = \frac{1}{4}$ and compute the respective simulation for S and I for 120 days starting from t_0. The resulting solution for the infectives I for this specific parameter set is shown in Figure 17. When applying our parameter estimation procedure to this specific set of training data, we obtain the results shown in Figure 18. In the top row on the left, we show the obtained estimate for the contact rate β for all timeframes. As we can observe the estimation for β is fairly accurate for the entire course of the epidemic, even though it is slightly perturbed within the last part of the curve, i.e., for approximately $t \geq 75$. Moreover, the resulting simulation for the infectives when using the estimated values for $\beta(t)$ is very accurate. In particular, we are able to replicate the second increase in the number of infectives which is fairly moderate compared to the first increase at the beginning of the curve; cf. the figure in the bottom row in Figure 18.

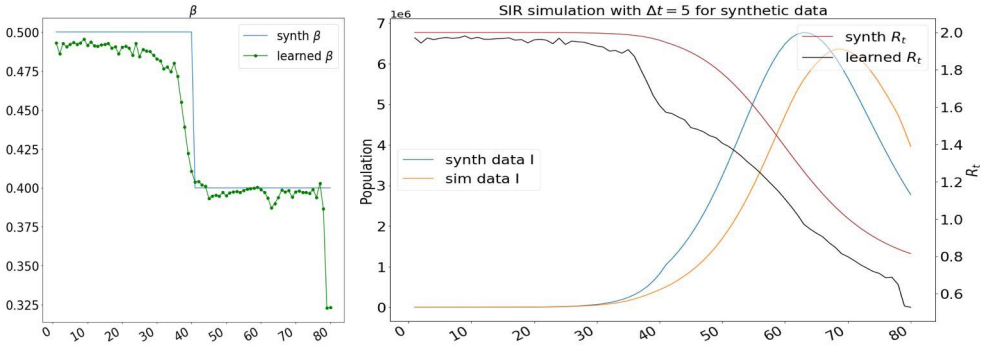


Fig. 13: Results for the SIR model and a contact rate $\beta(t)$ with a jump. **Left:** Estimate of $\beta(t)$ with $\Delta t = 5$. **Right:** Comparison of simulation results for the SIR model using the time-dependent estimate of $\beta(t)$ or using $\beta(t)$ itself, i.e., using the *learned* β versus the *synth* β from the left figure. See Figure 8 for the labeling.

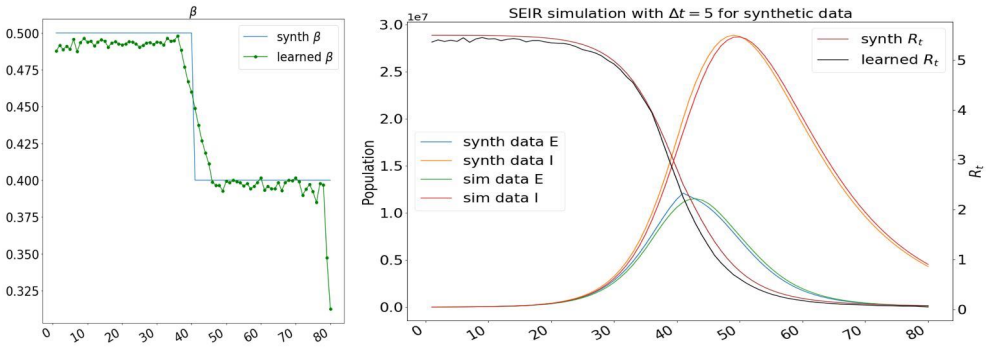


Fig. 14: Results for the $SEIR$ model and a contact rate $\beta(t)$ with a jump. **Left:** Estimate of $\beta(t)$ with $\Delta t = 5$. **Right:** Comparison of simulation results for the $SEIR$ model using the time-dependent estimate of $\beta(t)$ or using $\beta(t)$ itself, i.e., using the *learned* β versus the *synth* β from the left figure. See Figure 8 for the labeling.

Additionally, we tested our procedure for the same set of training data with additional noise added. Again, the noise, i.e., the pertubation in the data, is generated in the same way as in subsection 7.1. The corresponding estimate of $\beta(t)$ as well as the resulting simulated solution for the infectives are shown in Figure 19. As already in subsections 7.1 and 7.2.1, the proposed method is robust with respect to perturbations in the data and does still deliver satisfactory estimates for the contact rate.

For the numerical experiments obtained by using the $SEIR$ model, we have chosen a slightly different set of model parameters since using the same set of parameters as for the SIR model did not result in a course of infectives that emulates a second wave. In particular, for the simulations with the $SEIR$ model, we compute a time-dependent contact rate $\beta(t)$ by

$$\beta(t) = \beta\big(1 + c \cdot \cos(\pi(t + t_0)/35)\big) \tag{7.3}$$

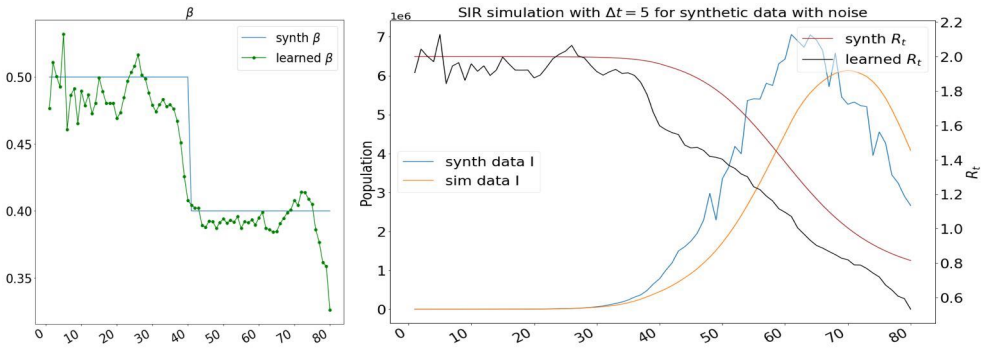


Fig. 15: Results for the SIR model, a contact rate $\beta(t)$ with a sharp jump and normally distributed noise on the training data. **Left:** Estimate of $\beta(t)$ with $\Delta t = 5$. **Right:** Comparison of simulation results for the SIR model using the time-dependent estimate of $\beta(t)$ or using $\beta(t)$ itself, i.e., using the *learned* β versus the *synth* β from the left figure. See Figure 8 for the labeling.

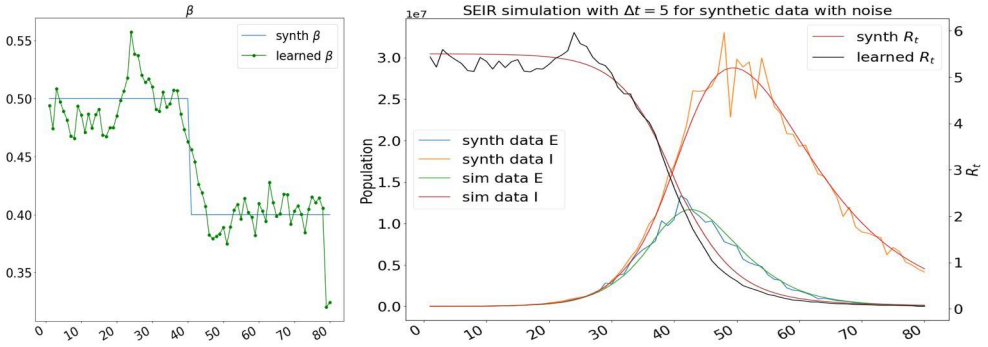


Fig. 16: Results for the $SEIR$ model, a contact rate $\beta(t)$ with a sharp jump and normally distributed noise on the training data. **Left:** Estimate of $\beta(t)$ with $\Delta t = 5$. **Right:** Comparison of simulation results for the SIER model using the time-dependent estimate of $\beta(t)$ or using $\beta(t)$ itself, i.e., using the *learned* β versus the *synth* β from the left figure. See Figure 8 for the labeling.

and set the remaining parameters as follows: $N = 10\,000$, $I(0) = 50$, $t_0 = 10, c = 0.75$, and $\beta = 0.2$. We further set the parameters γ and κ in (2.2) as $\gamma = \frac{1}{4}$ and $\kappa = \frac{1}{5}$ and compute the respective simulation for S, E, and I for 120 days starting from t_0. We show the results in Figure 20. Here, we observe that the learned β is very accurate for the complete course of the epidemic, even during the second wave with very low numbers of infected individuals.

We also present results for noisy $SEIR$ data in Figure 21; the noisy data have been generated as described in subsection 7.1. Again, we observe that our method is robust against perturbations visible both from the learned β as well as from the resulting simulations using the learned β.

Let us note that the sets of experiments in this section are the closest to reality in in the sense that they resemble a typical course of an epidemic disease. Thus, these

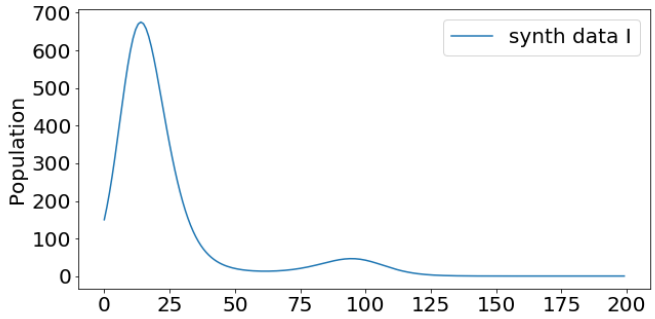


Fig. 17: Visualization for an exemplary curve for I for an SIR model with a second wave. Set of parameters for this simulation: $N = 10\,000$, $I(0) = 150$, $t_0 = 11$, $c = 0.45$, $\beta = 0.35$, and $\gamma = \frac{1}{4}$.

results serve as a proof of concept that we are able to estimate a time-dependent contact rate within epidemic models fairly accurately using PINNs as long as the data are in conformity with the underlying epidemic model. Moreover, the proposed method is also robust with respect to moderate perturbations in the reported numbers of infectives.

Finally, let us briefly comment on the robustness of our approach. As already mentioned in section 5, we always train a batch of PINNs for each timeframe and select the median of all corresponding parameter estimates as the final learned estimate of β for the respective timeframe. However, a priori it is not clear how many runs per timeframe are appropriate to obtain a fairly reliable estimate of the contact rate. To further investigate the robustness of our approach and to decide for a suitable number of runs per timeframe, we have tested our procedure for a single run as well as for 5 and 10 runs per timeframe; see Figure 22. For this set of experiments, we have trained the PINNs for an SIR model using a time-dependent contact rate $\beta(t)$ as defined in (7.2) with the same set of the parameters as before. As we can observe in Figure 22, using 5 as well as 10 runs per timeframe both deliver fairly similar results. We obtain a fairly accurate estimate of the reproduction number R_t and do not observe any extreme fluctuation within the parameter estimation. However, selecting only one randomly chosen run per timeframe results in larger perturbations of the estimate of R_t in the later course of the epidemic. Therefore, we have decided to always compute a batch of 5 runs for each timeframe for all the experiments presented in this paper.

7.3. Realistic contact rate for COVID-19 data in Germany. As a final proof of concept for our approach to estimate the contact rate $\beta(t)$ and the related effective reproduction number R_t within epidemic models, we consider the real data for the COVID-19 epidemic in Germany, based on the number of infected individuals as reported by the Johns Hopkins University; see [11]. Since individuals infected with COVID-19 are not infectious immediately and a significant exposed period exists, we consider the $SEIR$ model here. However, the parameter κ needed for the transition rate from the exposed to the infectious compartment is difficult to obtain directly from the available COVID-19 data. Thus, we first take the incubation period defined as the time from infection or exposure to the virus until the onset of symptoms and

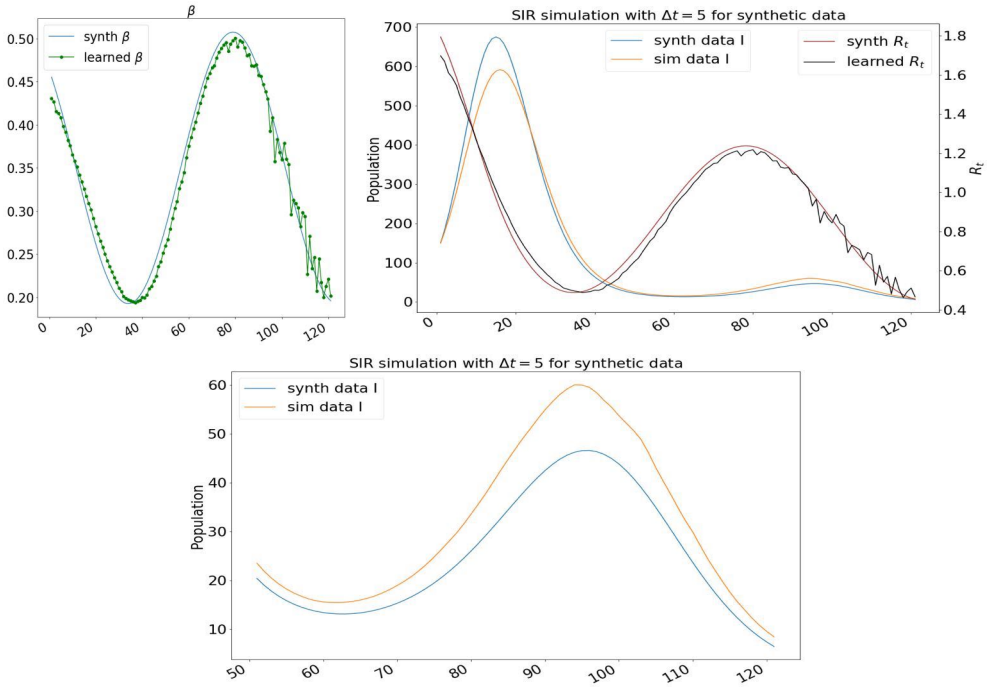


Fig. 18: Results for the SIR model and a contact rate $\beta(t)$ with a second wave. **Top, left:** Estimate of $\beta(t)$ with $\Delta t = 5$. **Top, right and bottom:** Complete simulation for the SIR model using the obtained estimate of $\beta(t)$ and a zoom to the second wave (bottom).

then subtract the presymptomatic infectious period in order to define $1/\kappa$. It has been reported in the literature [9] that the mean presymptomatic infectious period can vary between less than 1 and 4 days in different studies. As value for the incubation period, we choose 7.76 days which is reported as the median estimate for the incubation time in [19]. Combining this choice with 2.76 days as value for the presymptomatic infectious period, yields $\kappa = 1/5$. Let us note that other estimates of the incubation period are within the range of 5-6 days; see [1, Sect. 5]. Finally, we have to define γ which is given by the duration of infectiousness of an indiviual. Following [8], we assume that an individual is infectious for 8 days after the onset of symptoms. Let us note that there are also other choices possible; see [1, Sect. 10], where, e.g., 9-10 days are cited for mild to moderate courses of the disease. However, since in compartment models we mix all individuals with different courses of the disease, we have to make a certain choice within the range of published possibilities. Combining this choice with the previously selected value of 2.76 days for the presymptomatic infectious period, we decided to define $\gamma = 1/11$. Let us note that these values have been chosen to carry out some experiments with realistic data. We are aware of the fact that more experiments and a careful analysis with different sets of data are needed to make reliable predictions in the future. In fact, we have tried a few more combinations of κ and γ within the range of values given in the literature and have shown the results for the best choice in Figure 23. The selection of the best pairs of parameters κ and

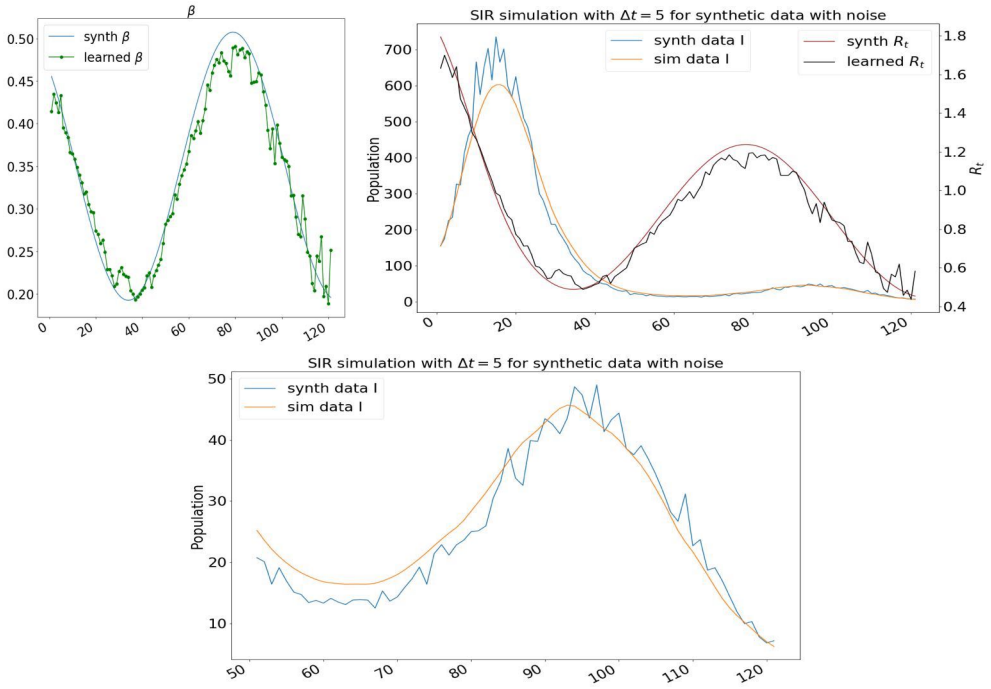


Fig. 19: Results for the SIR model, a contact rate $\beta(t)$ with a second wave and normally distributed noise on the training data. **Top, left:** Estimate of $\beta(t)$ with $\Delta t = 5$. **Top, right and bottom:** Simulation for the SIR model using the obtained estimate of $\beta(t)$ and a zoom to the second wave (bottom).

γ should be automated, e.g., by using an outer loop and a global error measure. This is left for future work.

While the data for $I(t)$ and $R(t)$ can easily be accumulated using the daily reported new infections and recoveries, we have to generate the training data for $E(t)$ artificially. We therefore assume that each reported new infection actually entered the compartment E five days earlier to compute an estimate of $E(t)$ for the training process of the PINNs. The resulting curves for $I(t)$ and $E(t)$ used as data are shown in Figure 23. Furthermore, we use $N = 83\,149\,300$ as a constant approximation of the population of Germany. Let us remark that we neglect births, deaths, and travelling individuals in our computations.

We report the learned effective reproduction number R_t as well as the simulation results for $I(t)$ and $E(t)$ using the $SEIR$ model and the learned $\beta(t)$ in Figure 23. Especially for the first large wave of infections during April and May the simulation results fit the real data very accurately. Obviously, the simulation is better for $I(t)$. Considering the data for July and August, the fit is less sufficient as well as for a short period in late June. As we have shown in the previous sections, our approach has always been able to produce quite accurate fits if the data are obtained by $SEIR$ simulations, even if they are disturbed. Hence, a likely conclusion is that the evolution of the real COVID-19 epidemic in Germany in the summer is not in good accordance with any $SEIR$ model and the data might be disturbed by, e.g., many travellers

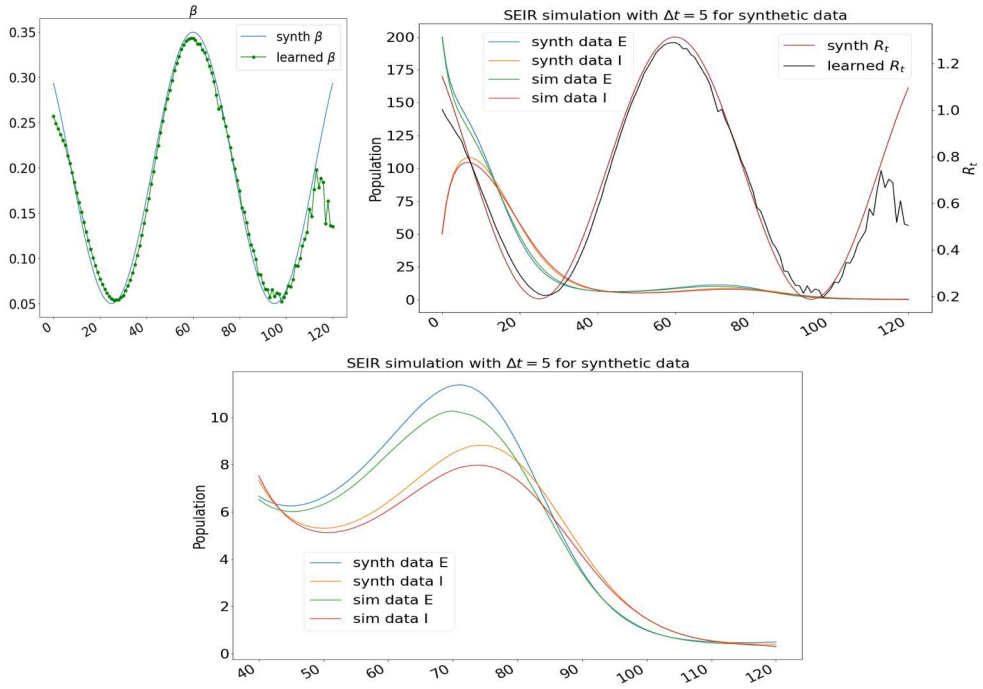


Fig. 20: Results for the $SEIR$ model and a contact rate $\beta(t)$ with a second wave. **Top, left:** Estimate of $\beta(t)$ with $\Delta t = 5$. **Top, right and bottom:** Simulation for the $SEIR$ model using the obtained estimate of β and a zoom to the second wave (bottom).

during the summer holidays, bringing infections into Germany from abroad and thus from outside of the respective compartment. This and the fact that less than 0.05% of the total population is infected at all makes it hard to fit the data. Please also note that the slight increase of infections in late June is due to a local outbreak of COVD-19 in a slaughterhouse. Such an effect is also not covered by the assumptions of the $SEIR$ compartment model that we use. Of course also our assumptions might be too strict and other parameter choices for κ and γ as well as different machine learning related parameter choices have to be considered in the future.

8. Conclusion and future work. We have considered a machine learning approach, which is based on physics-informed neural networks, to estimate parameters of dynamical systems. Previously, this method has been only applied to the case of constant parameters. In the present work, we have generalized it to the case of variable or time-dependent parameters and applied it to SIR and $SEIR$ compartment models from mathematical epidemiology. We have tested the new approach for several sets of numerical experiments with synthetic data, which have been generated by numerically solving the systems of ODEs defining the different compartment models. In our numerical experiments, we have used both the exact data and data perturbed with noise. In both cases and for different examples of contact rates (constant, discontinuous, time-dependent) as well as for the cases of a single and second wave, we were able to identify the parameters satisfactorily. From this we conclude that the

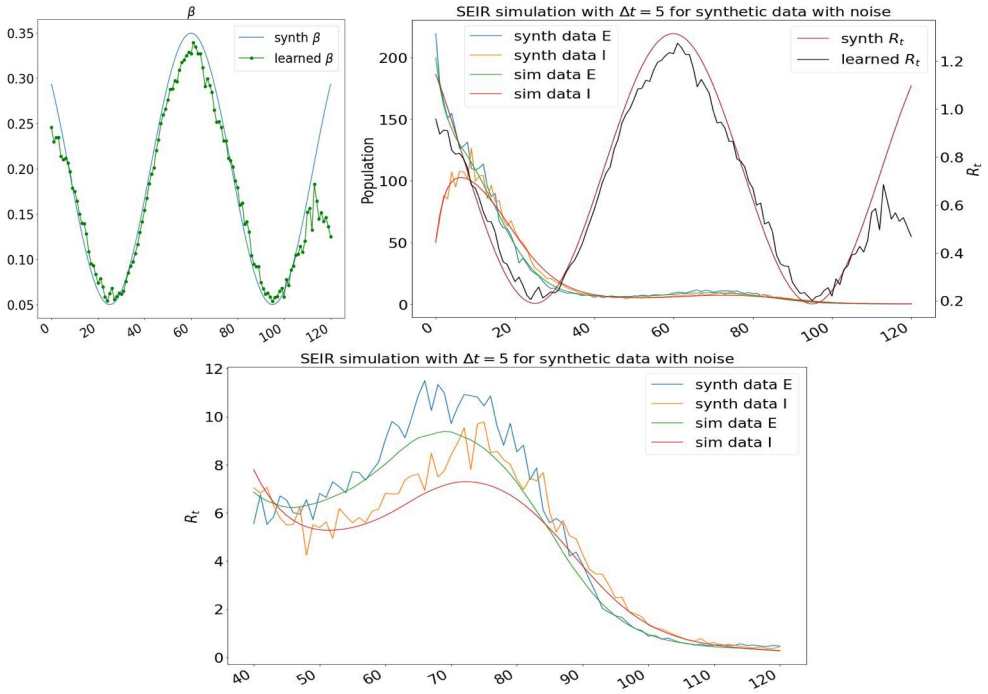


Fig. 21: Results for the $SEIR$ model, a contact rate $\beta(t)$ with a second wave and normally distributed noise on the training data. **Top, left:** Estimate of $\beta(t)$ with $\Delta t = 5$. **Top, right and bottom:** Simulation for the SIR model using the obtained estimate of $\beta(t)$ and a zoom to the second wave (bottom).

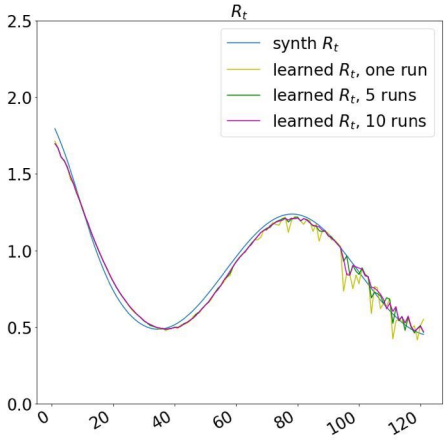


Fig. 22: Comparative results for the SIR model with a time-dependent $\beta(t)$ with a second wave. Estimations with $\Delta t = 5$ and different numbers of runs per timeframe, i.e., one, 5 and 10 runs for each timeframe tf_i.

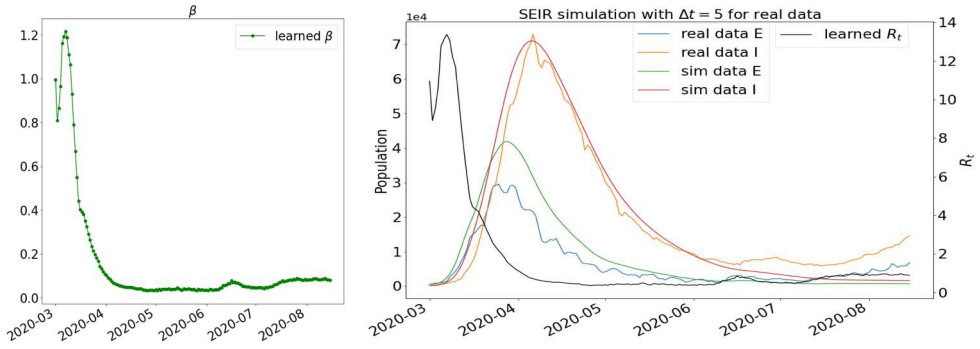


Fig. 23: Real data for $I(t)$ for the COVID-19 outbreak in Germany and generated $E(t)$ using $I(t)$ under the assumption that each individual is exposed for 5 days before it gets infectious for 11 days. The corresponding parameters are $\kappa = 1/5$ and $\gamma = 1/11$. Additionally, simulation results for the $SEIR$ model using the learned parameter β are shown as well as the resulting effective reproduction number R_t.

approach based on PINNs presented here, can identify parameters of SIR and $SEIR$ models as long as the training data are in agreement with the model assumptions. For synthetic data, we consistently obtained more accurate parameter estimates for the $SEIR$ model compared to the SIR model.

Finally, we extended our tests to real data from the COVID-19 epidemic in Germany and tested our approach using PINNs with an $SEIR$ model. Here, we are also able to identify the time-dependent contact rate and the resulting simulated number of infectious individuals I is in good agreement with the real data, at least until early or mid June. The simulated number of exposed individuals E is slightly underestimated. The local increase of infectious individuals in late June and the increase in July and August has not been captured satisfactorily although an increase of the contact rate and the related effective reproduction number can be seen in the learned data; but the latter is below one and thus the number of individuals does not increase. One possible explanation is that the real data at this stage of the course of the epidemic are not modeled well enough anymore by an $SEIR$ model since, for instance, in this period, many infections were caused abroad while traveling. Also, the slight increase in late June occurred due to a local outbreak related to a slaughterhouse and its workers. Thus, the assumption of homogeneous compartments was not satisfied anymore. Also, the dark number of unreported cases has not been considered in this model. Finally, we note that we have used the same incubation period for all age cohorts and assumed a constant degree of infectiousness - in contrast to an age of infection model. Overall, we nevertheless conclude that our new approach can be used to identify parameters of dynamical systems in compartment models. However, to obtain more accurate results for realistic epidemic data, we should extend our studies to more detailed models from mathematical epidemiology and also refine our machine learning approach. This will be the topic of future work.

Acknowledgments. We gratefully acknowledge the use of the computational facilities of the Center for Data and Simulation Science (CDS) at the University of Cologne and at the Computing Center of the Technical University of Freiberg. We also gratefully acknowledge the use of the real data for the COVID-19 epidemic in

Germany obtained from Johns Hopkins University. This work was supported in part by the Helmholtz School for Data Science in Life, Earth, and Energy (HDS-LEE), https://www.hds-lee.de.

REFERENCES

[1] *Robert-Koch-Institut (RKI), Germany. SARS-CoV-2 Steckbrief zur Coronavirus-Krankheit-2019 (COVID-19).* https://www.rki.de/DE/Content/InfAZ/N/Neuartiges_Coronavirus/Steckbrief.html#doc13776792bodyText10. Accessed: 2020-09-22.

[2] *World Health Organization (WHO) Director-General's opening remarks at the media briefing on COVID-19.* https://www.who.int/dg/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-covid-19---11-march-2020. Accessed: 2020-09-18.

[3] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, *TensorFlow: Large-scale machine learning on heterogeneous systems*, 2015, https://www.tensorflow.org/. Software available from tensorflow.org.

[4] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind, *Automatic differentiation in machine learning: a survey*, The Journal of Machine Learning Research, 18 (2017), pp. 5595–5637.

[5] C. M. Bishop et al., *Neural networks for pattern recognition*, Oxford university press, 1995.

[6] F. Brauer, *Compartmental models in epidemiology*, in Mathematical epidemiology, vol. 1945 of Lecture Notes in Math., Springer, Berlin, 2008, pp. 19–79, https://doi.org/10.1007/978-3-540-78911-6_2, https://doi.org/10.1007/978-3-540-78911-6_2.

[7] F. Brauer, C. Castillo-Chavez, and Z. Feng, *Mathematical models in epidemiology*, vol. 69 of Texts in Applied Mathematics, Springer, New York, 2019, https://doi.org/10.1007/978-1-4939-9828-9, https://doi.org/10.1007/978-1-4939-9828-9. With a foreword by Simon Levin.

[8] J. Bullard, K. Dust, D. Funk, J. Strong, D. Alexander, L. Garnett, C. Boodman, A. Bello, A. Hedley, Z. Schiffman, K. Doan, N. Bastien, Y. Li, P. Van Caeseele, and G. Poliquin, *Predicting infectious SARS-CoV-2 from diagnostic samples*, Clin. Infect. Dis., (2020), https://doi.org/10.1093/cid/ciaa638.

[9] A. W. Byrne, D. McEvoy, A. B. Collins, K. Hunt, M. Casey, B. Ann, F. Butler, J. Griffin, E. A. Lane, C. McAloon, K. O'Brien, P. Wall, K. A. Walsh, and S. J. More, *Inferred duration of infectious period of ARS-CoV-2: rapid scoping review and analysis of available evidence for asymptomatic and symptomatic COVID-19 cases*, BMJ Open, 10 (2020), https://doi.org/10.1136/bmjopen-2020-039856.

[10] P. Delamater, E. Street, T. Leslie, Y. Yang, and K. Jacobsen, *Complexity of the basic reproduction number (r0)*, Emerg Infect Dis., 25 (2019), pp. 1–4, https://doi.org/doi:10.3201/eid2501.171901, https://doi.org/10.3201/eid2501.171901.

[11] E. Dong, H. Du, and L. Gardner, *An interactive web-based dashboard to track covid-19 in real time*, The Lancet Infectious Diseases, 20 (2020), pp. 533 – 534, https://doi.org/https://doi.org/10.1016/S1473-3099(20)30120-1, http://www.sciencedirect.com/science/article/pii/S1473309920301201.

[12] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*, vol. 1, MIT press Cambridge, 2016.

[13] H. W. Hethcote, *The mathematics of infectious diseases*, SIAM Rev., 42 (2000), pp. 599–653, https://doi.org/10.1137/S0036144500371907, https://doi.org/10.1137/S0036144500371907.

[14] W. O. Kermack and A. G. McKendrick, *A contribution to the mathematical theory of epidemics*, Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character, 115 (1927), pp. 700–721.

[15] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, arXiv preprint arXiv:1412.6980, (2014).

[16] G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter, *Self-normalizing neural networks*, in Advances in neural information processing systems, 2017, pp. 971–980.

[17] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, *Efficient backprop*, in Neural networks: Tricks of the trade, Springer, 2012, pp. 9–48.

[18] L. Prechelt, *Early stopping-but when?*, in Neural Networks: Tricks of the trade, Springer, 1998, pp. 55–69.

[19] J. Qin, C. You, Q. Lin, T. Hu, S. Yu, and X.-H. Zhou, *Estimation of incubation period distribution of covid-19 using disease onset forward time: A novel cross-sectional and forward follow-up study*, Science Advances, 6 (2020), https://doi.org/10.1126/sciadv.abc1202, https://advances.sciencemag.org/content/6/33/eabc1202, https://arxiv.org/abs/https://advances.sciencemag.org/content/6/33/eabc1202.full.pdf.

[20] M. Raissi, P. Perdikaris, and G. E. Karniadakis, *Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations*, Journal of Computational Physics, 378 (2019), pp. 686–707, https://doi.org/10.1016/j.jcp.2018.10.045, https://doi.org/10.1016/j.jcp.2018.10.045.

[21] K. Schittkowski, *Numerical data fitting in dynamical systems*, vol. 77 of Applied Optimization, Kluwer Academic Publishers, Dordrecht, 2002, https://doi.org/10.1007/978-1-4419-5762-7, https://doi.org/10.1007/978-1-4419-5762-7. A practical introduction with applications and software, With 1 CD-ROM (Windows).

[22] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, *SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python*, Nature Methods, 17 (2020), pp. 261–272, https://doi.org/10.1038/s41592-019-0686-2.