Bayesian data-driven model discovery under uncertainty

Michela Ceoloni (10535585) - Federico Fatone (10530963) - Filippo Fedeli (10534669)

Mathematical Engineering - Statistical Learning Major - Politecnico di Milano

18 February 2021

Final report of Bayesian Statistics course project

A.Y. 2020/21

Prof. A. Guglielmi, Dr. R. Corradin, Dr. M. Beraha

Politecnico di Milano

Bayesian data-driven model discovery under uncertainty

Application to the first and second outbreaks of the covid-19 epidemic in Italy

Michela Ceoloni (10535585) - Federico Fatone (10530963) - Filippo Fedeli (10534669)

Supervisors: prof. A. Guglielmi, prof. A. Manzoni

Keywords: Bayesian statistics, Dynamical systems, Uncertainty quantification, Model identification, Approximate Bayesian Computation, Sequential Monte Carlo, Hamiltonian Monte Carlo, Differentiable programming.

Abstract

Model identification and parameters estimation in dynamical systems starting from raw data are of extreme importance in many fields of application and particularly in life sciences. In this context, Bayesian tools have been extensively applied with the primary purpose of quantifying the uncertainty of results. A novel method of approaching such problems, based on Hamiltonian Monte Carlo sampling, was recently proposed by Perdikaris et al.; however, some issues can arise when trying to apply this strategy to real-world data.

Our work can be cast in this framework, providing Bayesian tools to identify and select differential models and yield robust future forecasts with quantified uncertainty, even when applied to non-simulated data (specifically to Covid-19 epidemiological data). In particular, the focus of the project is on the comparison between the aforementioned technique and some methods proposed in the framework of Approximate Bayesian Computation in the last decade, in particular regarding the application of Sequential Monte Carlo introduced by Toni et al. in 2009. Thanks to the addition of a new, original, preconditioning phase and the introduction of a suitable parallelization scheme, we achieved better performances with respect to the original ABC-SMC method and the same accuracy performances of the original HMC sampling, however featuring a computational speedup of about 10 times. Furthermore, the predictive potential of the revisited ABC method has been applied to the data-driven identification of epidemiological models described in terms of nonlinear systems of ordinary differential equations, such as the well-known SIR model. In particular, we have addressed several situations, ranging from the so-called first outbreak (spring 2020) to the second outbreak (fall 2020) of the covid-19 pandemic in Italy, aiming at providing a ready-to-use emergency response tool. A Comparison between «yellow zone» and «red zone» regions has been also discussed. Thanks to the Bayesian data-driven framework we developed, several results have been achieved, such as a tight estimation of the peak of the first outbreak based on data recorded on infected and recovered compartments, and a clear distinction of the epidemic trends as a consequence of different measures taken in two Italian regions.

1 Introduction

The analysis of dynamical systems is of paramount importance in many fields of application. The development of techniques supporting scientists for this purpose is even more relevant, recently, due to the continuously increasing amount of available data and the relevance of problems at stake (climate change, epidemics...). The main goals in this context are the identification of interpretable and predictive features and, even more challenging for applications, the production of forecasts and control strategies [1].

With this aim, several techniques have been developed in the last years, taking advantage of machine learning frameworks and data-driven modeling approaches. Unfortunately, though, when model identification and parameters estimation are considered, the quality of the observations heavily influences the outcomes, and the deterministic predictive approach does not provide a sought thorough uncertainty quantification [1]. The need for a more flexible framework able to deal with noisy, sparse and irregularly sampled data, but at the same time providing quantified uncertainty of future forecasts and parameters estimates, drove the research towards the use of Bayesian approaches.

A new fully Bayesian framework formulation for robust systems identification from imperfect time-series data was proposed recently in [1]. One of the key features of the novel workflow is the construction of an accelerated Hamiltonian Monte Carlo (HMC) scheme [8], made possible by taking advantage of current developments in differentiable programming, e.g. exploiting gradient information. Good properties related to this framework are the relative computational efficiency and the end-to-end differentiability (both achieved with the *Tensorflow* platform). Furthermore, the identification of the latent dynamics is performed by choosing sparse-promoting priors, thus yielding reliable results [1].

At first, we applied the method to a simple benchmark example, dealing with Lotka-Volterra prey-predator model, as proposed by Perdikaris et al. (2020) [1] and we were able to replicate accurately the described results[1]. In this phase we noticed that the goodness of the estimate of some parameters in the proposed examples could have been improved, in particular regarding their autocorrelation. We then adopted a better Mass matrix choice for the Hamiltonian Monte Carlo, that also allowed us to sample less than in the original procedure proposed by the authors, thus lowering the required computational time.

We then extended this framework to the SIR epidemiological model [12], not considered in the initial paper, which we regarded as the starting point to develop an analysis of more complex epidemiological models, this latter representing the final goal of this project. We artificially added a noise on simulated data to emulate real-world raw data and considered two ghost variables that would be ideally discarded during model identification phase. Nevertheless, one main issue arose. Indeed, the numerical stage implemented using the Python *Tensorflow* auto-differentiation was based intrinsically on a batch-based feeding of data, that in this context meant the requirement of lots of data. Therefore, as this method was computationally inapplicable without relying on auto-differentiation, this made the method not adaptable to models fed with real-world data in data scarcity conditions. Another issue regarded the complexity of HMC method, which seemed

[1] All the results are presented in section 4 - Results

to be quite oversized for the number of parameters parameters being estimated in the examples in [1] and in the models we were planning to consider. Last, but not least, there is no reason behind the choice of the model for the error (entailing a likelihood) on fitted data w.r.t. reality, taken as Gaussian in [1].

We then turned to an alternative approach in order to achieve our final goal of applying a quantitative Bayesian uncertainty estimation framework to differential problems using real-world raw data. In particular, we considered the Approximate Bayesian Computation method using Sequential Monte Carlo (ABC-SMC) scheme, described by Toni et al. (2009), whose effectiveness for different kinds of modelling approaches was proven, especially in the life sciences field [2,3,4]. Typical of ABC methods is the fact that they rely on a simulation-based procedure, in order to skip the direct evaluation of the likelihood [4,5], as it was desirable for the problem at hand, as mentioned before; in particular, ABC-SMC is relatively computationally efficient and easily parallelizable [2,3,4].

In the application of this framework to our problems, we faced some troubles in case of a choice of priors being not diffuse enough (crucial when performing model identification, *e.g.* when applying a Lasso technique). This matter led us to the introduction of a preconditioning phase, as we actually knew a good estimate of the true centre of the posterior distribution, from the numerical estimation phase, and so we knew some information to guide the sampling with, in order to make it more efficient. To test our improvements and the goodness of this Bayesian framework, we applied our idea to the example we considered initially when applying [1], the Lotka-Volterra (LV) model, aiming at comparing the two approaches, and highlighting their most relevant features. The good news were that our framework gave similar results than the ones obtained with the approach described in [1], but much less computational time was required (without parallelization scheme we achieved a 3x speedup and with a parallelization scheme we achieved a 10x speedup on 4 cores on Intel Core i7 processor).

Thanks to the good performances of our developed framework, we eventually decided to apply it to the case study we were pursuing since the beginning of our work, namely the Covid-19 outbreak in Italy. By the usage of a model inspired by the one described in [7], we obtained really promising results in prediction relying on the prior knowledge coming from the China outbreak, both in short term (using the first 30 days data we were able to locate in time and size the peak) and in long term (using 90 days data, we were able to locate the asymptotes of the model). We then considered an application to the «second wave» and aiming in particular at highlighting possible differences among regions in *red zones* and in *yellow zones*, and finding evidence of a difference in the two cases.

Report structure

This report is organized as follows. In Section 2 we describe the several methods taken into consideration (Perdikaris et al., Toni et al., our improvements and the selected final version).

In Section 3 we present the epidemiological models used to describe the Covid-19 outbreak as well as the Bayesian framework adopted for the uncertainty quantification part.

In Section 4 we show our results, namely the original HMC sampling approach applied to Lotka-Volterra and to a SIR model, the comparison of our ABC-SMC method with the HMC one in the

Lotka-Volterra case with computational times at stake. Moreover, we show how our framework can be applied to mathematical epidemiological models, aiming at (i) identifying the model using data from the «first outbreak»in Italy and assessing its predictive capabilities, (ii) perform a robustness comparison against the case of Spain, and (iii) assessing the differences in the epidemic evolution during the «second outbreak»among Italian regions like Lombardia and Veneto, which have undergone different measures in the past months. At the end of this report, in section 5, we give some conclusions, discussing advantages and limitations of our approach and proposing possible directions for further analyses.

2 Methods

In this section, we illustrate in detail all the frameworks and methods involved in our work. First, we give a description of the Perdikaris' workflow described in [1], from which we started developing our project. We then introduce the ABC-SMC method by Toni et al. [2, 3, 4], and the development of a preconditioning step to the aforementioned framework is shown at the end of the section. We will refer to the ABC-SMC method including the preconditioning step as *empirical* ABC-SMC (eABC-SMC).

2.1 Bayesian framework using HMC scheme (Perdikaris et al.)

The workflow originally defined by Perdikaris et al. [1] is divided into two highly connected parts. The former is a numerical framework for deterministic parameters estimation, through differentiable programming. The latter is the proper Bayesian framework that makes use of Hamiltonian Monte Carlo sampling for inference and quantification of uncertainty.

2.1.1 Numerical parameters estimation

The goal of this step is to provide an estimate of the model's parameters in a deterministic flavour through the numerical optimization of a loss function. This value will in the end be used as starting point for the Hamiltonian Monte Carlo sampling. One of the key aspects of this step is the strong leverage on the propagation of gradient information through classical numerical solvers for ODEs, mixing classical adjoint methods with modern developments in automatic differentiation [1].

Consider now a general D-dimensional dynamical system of the form

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, t; \boldsymbol{\theta})$$

where $t > 0$, $\mathbf{x}(0) = \mathbf{x}_0$ initial data, $\boldsymbol{x} \in \mathbb{R}^D$ denotes the state of the system, $t \in \mathbb{R}^+$ is the time, $\boldsymbol{\theta} \in \Theta$ are the unknown parameters used in to describe the latent dynamics given by $f : \mathbb{R}^D \to \mathbb{R}^D$. Data consist of $\{\boldsymbol{x}_i\}_i$ observed at time t_i, $i = 1, ..., n$.
System identification is performed by minimizing a L^2 loss function in the variable $\boldsymbol{\theta} \in \Theta$, representing the measure of the error on the prediction with respect to the corresponding actual observation. Thus, the optimal choice for $\boldsymbol{\theta}$, denoted by $\hat{\boldsymbol{\theta}}$, is the one that minimizes, through

gradient descent, the loss function (1), i.e.

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\text{argmin}} \mathcal{L}(\boldsymbol{\theta}) \tag{1}$$

where

$$\mathcal{L}(\boldsymbol{\theta}) = \mathcal{J}(\boldsymbol{\theta}) + \beta ||\boldsymbol{\theta}||_1 \tag{2}$$

and

$$\mathcal{J}(\boldsymbol{\theta}) = \sum_{i=1}^{n} ||\boldsymbol{x}(t_i + \Delta t_i) - h_{\boldsymbol{\theta}}(\boldsymbol{x}(t_i))||_2^2 \tag{3}$$

is the proper L^2 loss function, and $h_{\boldsymbol{\theta}}(\boldsymbol{x}(t_i))$ is the output of a numerical ODE solver, which in this specific application is the fourth order Runge-Kutta method (RK4). The loss function (1) is therefore the combination of a Eucledian (2-norm) loss function used to evaluate the goodness of fit of the model and a norm-1 penalization on parameters, used to help the model identification phase by keeping the model size low.

In order to perform gradient descent, the loss (1) needs to be differentiated, and this is where the *Tensorflow* platform enters the original framework for the first time. In particular, using the adjoint system to system (0)-(2), where (0) is the dynamical system $\boldsymbol{a}(t)$ as described in [8], it can be proved that:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} = -\int_{t_1}^{t_0} \mathbf{a}(t)^T \frac{\partial f(\boldsymbol{x}(t), t, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} dt \tag{4}$$

with

$$\frac{d\boldsymbol{a}(t)}{dt} = -\mathbf{a}(t)^T \frac{\partial f(\boldsymbol{x}(t), t, \boldsymbol{\theta})}{\partial \boldsymbol{x}} dt. \tag{5}$$

Therefore the possibility to automatically derive in this phase is of paramount importance.

This workflow allows to produce only deterministic point estimates for $\boldsymbol{\theta}$, while the main purpose of the work is to produce a robust uncertainty quantification on parameters. To achieve this goal we will rely on a Bayesian estimation framework.

2.1.2 Bayesian uncertainty quantification

The core of the work by Perdikaris et al. [1] is the quantification of the uncertainty for the parameters estimation and thus for the predictions. A natural way to account for uncertainty and thus have a complete statistical characterization for all inferred model parameters in the model is the Bayesian approach. In particular, the underlying Bayesian model is the following:

$$p(\boldsymbol{x}(t + \Delta t)|\boldsymbol{x}(t), \boldsymbol{\theta}, \gamma) = \prod_{i=1}^{N} \mathcal{N}(\boldsymbol{x}(t_i + \Delta t_i); h_{\boldsymbol{\theta}}, \gamma^{-1}) \tag{6}$$

$$p(\boldsymbol{\theta}|\lambda) = Laplace(\boldsymbol{\theta}; 0, \lambda^{-1}) \tag{7}$$

$$p(log\lambda) = Gamma(log\lambda; \alpha_1, \beta_1) \tag{8}$$

$$p(log\gamma) = Gamma(log\gamma; \alpha_2, \beta_2). \tag{9}$$

It is worth to note that in the likelihood (6) we are assuming a normal distribution for the error given the prediction and that (7) and (8) are implementing a Lasso-like prior useful in performing an effective model selection, with λ being the shrinkage parameter of the model. The parameter γ considered in the likelihood and in the prior (9) represents instead the precision of the model.

Given the prior information for the unknown model parameters $\boldsymbol{\theta}$ and the expression of the likelihood function (6),the posterior distribution is computed as

$$p(\gamma, \lambda, \boldsymbol{\theta}|\mathbf{x}(t+\Delta t), \mathbf{x}(t)) \propto p(\mathbf{x}(t+\Delta t)|\mathbf{x}(t), \boldsymbol{\theta}, \gamma)p(\boldsymbol{\theta}|\lambda)p(\gamma)p(\lambda),$$

Since the posterior is not conjugate to the priors and it cannot be expressed in the form of any known distribution, a sampling method must be considered. The choice fell on Hamiltonian Monte Carlo sampling, which is normally used when the dimension of the parameter space Θ is large, as the sampling from the posterior is very difficult and computationally expensive with simpler method such as Metropolis-Hastings algorithm or the Gibbs sampler. Furthermore, empirically, the sampling showed consistent autocorrelation issues and HMC provides extremely good performances in this respect.

In Hamiltonian Monte Carlo, to generate the Markov Chain, an energy preserving leapfrog scheme is used to integrate the Hamilton's equations, which take into account the gradient ∇H of the Hamiltonian. In order to compute it, it is necessary to use an automatic derivation tool in order to achieve acceptable computational speeds, thus making the *Tensorflow* platform fundamental also for this step.

2.1.3 Limits of the presented method

Although this workflow is innovative and very flexible, as shown by our results using modified priors in order to consider Ridge regression show, it also entails also some drawbacks. First of all, regarding its original implementation, we faced issues when working with real-world data. The *Tensorflow* platform used for auto-differentiation required in fact a *batch-feeding* mechanism. This makes necessary to have lots of data available at different time steps in order to form consistent batches to feed the differentiation scheme with. This fact is not an issue when considering only simulated data as the authors did, however it becomes a real problem when trying to use it in data-scarcity situations as the one we were aiming to consider. Furthermore, there is actually no evidence to take a Gaussian error on fitted data w.r.t. reality, as the likelihood of the proposed model did, in fact there is no need to take a likelihood at all.

As the number of parameters for the cases at hand was rather limited, using Hamiltonian Monte Carlo was *de facto* unnecessary and so we decided to explore alternatives to this workflow.

2.2 ABC-SMC method (Toni et al.)

In order to replace the workflow described above, we selected the Bayesian framework developed by Toni et al. [2, 3], mainly due to the already proven efficiency with dynamical systems in life sciences and the likelihood-free approach to the problem.

Typical of ABC methods is their possibility to work with complex models such as dynamical systems without having to directly evaluate the likelihood, but just having to define the priors [2, 5]. The general scheme is the following [2, 3, 4, 5]:

Algorithm 1: Approximate Bayesian Computation - Outline

```
Result: A sample from $\pi(\boldsymbol{\theta}|x)$
Initialization: k = 0
while $k \leq K$ do
    sample $\theta$ from $\pi(\theta)$;
    compute $y = f(\cdot|\theta)$;
    if $\Delta(x, y) \leq \epsilon$ then
        save $\theta$ and $y$;
        k $\leftarrow k + 1$
    end
end
```

In Algorithm 1, $\Delta(x, y)$ is a suitable distance that replaces conceptually the role of the likelihood as it represents the confidence the statistician has in the fact that the observed sample actually comes from the underlying distribution. In our case we have that

$$\Delta(\boldsymbol{x}, \boldsymbol{y}(\boldsymbol{\theta})) = \sum_{i=1}^{n} ||\boldsymbol{x}(t_i) - h_{\boldsymbol{\theta}}(t_i)||^2. \tag{10}$$

2.2.1 ABC - Sequential Monte Carlo

As this *vanilla* ABC method is extremely inefficient for small ϵ (in the sense that the acceptance rate is usually extremely low), Toni et al. in 2009 proposed an alternative: ABC-Sequential Monte Carlo (or ABC-SMC). ABC-SMC is a population-based method, which approximates the posterior exploiting some intermediate distributions proceeding similarly to the way importance sampling works [2, 3].

The aim is to sample sequentially from the sequence of distributions $p_{\epsilon_t}(\boldsymbol{\theta}|\boldsymbol{x})$, following a decreasing tolerance sequence $\{\epsilon_t\}_{1 \leq T}$. At the first step of ABC, many particles are accepted thanks sufficiently large values chosen for ϵ_1. For the next iterations, the particles are sampled from the set of accepted particles at the previous stage, perturbed according to a perturbation kernel and weighted according to a weight function that mimics the prior's distribution. At every stage t, a sample from the posterior $p_{\epsilon_t}(\boldsymbol{\theta}|\boldsymbol{x})$ is built until the target posterior is reached [2, 3, 4]. Note that this procedure is highly parallelizable, due to the independence of the prior samplings, thus ensuring a sensible reduction of computational time if implemented. We carried out a speedup analysis of a simple parallelization scheme which is presented in the Results section. Algorithm 2 provides in detail the ABC-SMC sampling procedure as described in [2, 3]:

```
Algorithm 2: ABC - Sequential Monte Carlo
  Result: A sample from $p_\epsilon(\boldsymbol{\theta}|x)$
  Initialization: A precision schedule $\{\epsilon_t\}_{t\in 1:T}$;
  while $t \leq T$ do
      while $n \leq N$ do
          if $t = 1$ then
              sample $\tilde{\theta}$ from $\pi(\theta)$;
          else
              sample $\theta$ from the previous population $\{\theta^{(i,t-1)}\}_i$ with weights $\{\omega^{(i,t-1)}\}_i$;
              sample $\tilde{\theta}$ from $K_t(\cdot|\theta)$ s.t. $\pi(\theta) > 0$;
          end
          compute $y = f(\cdot|\tilde{\theta})$;
          if $\Delta(y, x) \leq \epsilon_t$ then
              save $\tilde{\theta}$ and $y$;
          else
      end
      compute $\{\omega^{(i,t)}\}_i$ and normalize them;
  end
```

According to [3] we used a component-wise Gaussian perturbation kernel:

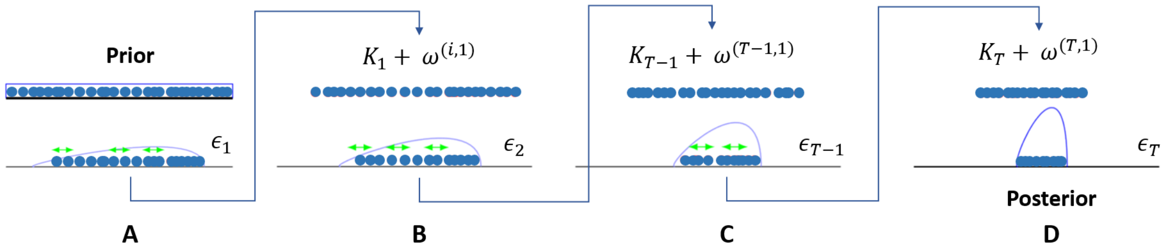
$$K_t(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t-1)}) = \prod_{j=1}^{d} \frac{1}{\sqrt{2\pi}\sigma_j^{(t)}} exp\left\{-\frac{(\theta_j^{(t)} - \theta_j^{(t-1)})^2}{2\sigma_j^{(t)2}}\right\} \tag{11}$$

with $\sigma_j^{(t)}$ chosen adaptively at each iteration (in our case equal to the standard deviation of the j^{th} component of θ at the previous timestep). Weights are instead computed according to

$$\omega^{(i,t)} \leftarrow \frac{\pi(\theta^{(i,t)})}{\sum_{j=1}^{n} \omega^{(j,t-1)} K_t(\theta^{(i,t)}|\theta^{(j,t-1)})} \tag{12}$$

with $\omega^{(i,1)} \propto 1$.

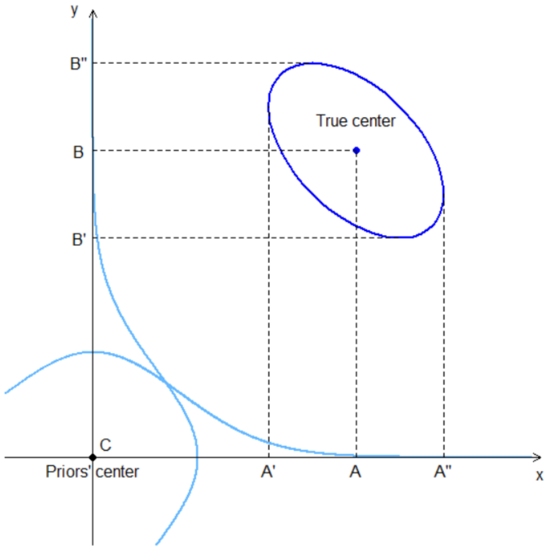
In order to provide a better understanding of the ABC-SMC algorithm, which is crucial in order to get the point of our *empirical* preconditioning framework, we report in the following a diagram that visually describes the concept behind it:



In the first step (A), a starting population is sampled directly from the prior through the acceptance-rejection condition based on the distance $\Delta(x,y)$ described before, in such a way that only samples having a distance $\Delta(x,y) \le \epsilon_1$ are accepted. In the step B, and similarly in step C and D, some weights are associated to each element of the previous population ($\omega^{(i,t)}$) and a perturbation kernel K_t is applied to it. Sampling from the previous population with probabilities proportional to the weights, and accepting or rejecting the candidates according to the distance condition $\Delta(x,y) \le \epsilon_t$, determines the formation of a new population, according to a precision sequence $\{\epsilon_t\}_t$. The last population (the one obtained with precision ϵ_T) is considered as the sampling from the posterior distribution.

2.3 Empirical ABC-SMC method (eABC-SMC)

Although the ABC-SMC framework above seems to be an effective and promising strategy, it however features some drawbacks. In particular, when priors are not diffused enough (as in the case of the Lasso-like or Ridge-Like model selection method presented before) it might show very poor performances in the initial sampling (t=1). In this scenario, by using ABC-SMC when the posterior centre is very far from the prior one, the probability of accepting a point is almost null. For this reason, we proposed an original, *empirical* preconditioning strategy to improve the acceptance rate of the original ABC-SMC algorithm, yielding what we have referred to as Empirical ABC-SMC (eABC-SMC) method.



For instance, consider the following two-dimensional case, in which we report as the blue ellipse the border of the acceptance region ($\Delta(x,y) = \epsilon_1$) and in light blue the distribution of the priors (zero-centered Gaussian distibutions).
In this case, the probability for the ABC-SMC algorithm to sample a point that would be accepted is very low ($\mathcal{O}(10^{-4})$).

The key idea behind eABC-SMC is to find an estimate for the borders of the acceptance region (A' and B' in this case) and to sample from the prior distributions truncated there.

This suggested us some preconditioning to be added to the classical ABC-SMC workflow, exploiting the numerical estimate of the true centre that comes from the deterministic estimation phase, in order to somehow «help» the sampling by the usage of appropriate truncated distributions.
This idea is justified by the fact that, due to the Theorem of Total Probabilities we find that, given the acceptance region $\mathfrak{R} = \{\boldsymbol{y} : \Delta(\boldsymbol{x},\boldsymbol{y}) \le \epsilon\}$:

$$\mathcal{L}(X \in \mathfrak{R}) \propto \mathcal{L}\left(X \in \mathfrak{R} \mid \bigcap_{j=1}^{d} \{X_j \in \Xi_j\}\right) \tag{13}$$

with Ξ_j being the projection of $\mathfrak{R}$ on j^{th} axis ($[A', A'']$ and $[B', B'']$ in the example above).

Now, after having estimated the borders of Ξ_js, it is possible to sample from the truncated prior distributions discarding completely parts of them where *a priori* we know that samples would be rejected. Regarding the borders estimation, we report here for simplicity the one dimensional version of the algorithm performing it. The extension to the multi-dimensional case is natural as it requires iterating the previous scheme to each dimension.

Algorithm 3: ABC-SMC 'empirical' preconditioning (1D)

```
Result: A: estimate of the border of $\mathfrak{R}_{\epsilon_{start}}$
Initialization: A: estimate of the posterior centre, C: the mean of $\pi$, $tol \in$ (0,1) ;
while $n \leq N$ do
    if $A > C$ then
        if $A > q_\pi(1 - tol)$ or $A < q_\pi(tol)$ then
            sample P $\sim \mathcal{U}$((C, A));
            sample $\tilde{\theta}$ from $\pi(\theta|\theta > P)$;
        else
            sample $\tilde{\theta}$ from $\pi(\theta)$;
        end
    else
        if $A > q_\pi(1 - tol)$ or $A < q_\pi(tol)$ then
            sample P $\sim \mathcal{U}$((A, C));
            sample $\tilde{\theta}$ from $\pi(\theta|\theta < P)$;
        else
            sample $\tilde{\theta}$ from $\pi(\theta)$;
        end
    end
    compute $y = f(\cdot|\tilde{\theta})$;
    if $\Delta(y, x) \leq \epsilon$ then
        save $\tilde{\theta}$;
    else
end
return $min(\tilde{\theta})I_{(A>C)} + max(\tilde{\theta})I_{(A<C)}$
```

The key idea is as follows. If the true centre of the posterior is too far from the prior's centre (so the sampled points will be likely not to be accepted), we sample a point from a Uniform distribution between the prior's centre and the true posterior's centre, and sample from the prior truncated at that point. We then compute the summary statistics, resulting in acceptance or rejection, and in the end we take the minimum (or the maximum, depending on the *true centre* location) of the accepted points as an estimate of the borders of our posterior distribution.

After the estimation of the borders we proceed sampling from the truncated distributions in order to build the first population to proceed with the ABC-Sequential Monte Carlo sampling described in Algorithm 2.

This addition to the classical ABC-SMC algorithm is an original contribution and it has been the gear that, added to the machine, made the latter finally work properly, allowing us to achieve extremely satisfying results with real-world data. In the following sections, results obtained with this method are presented and compared with the original Hamiltonian Monte Carlo sampling starting point.

3 Epidemiological Model

Before turning to the numerical results obtained with the aforementioned strategies, we introduce in this section the mathematical epidemiological model that we will consider in the following to model the covid-19 outbreak in different situations.

3.1 Differential model

The model we chose for the description of Covid-19 outbreak in Italy was a SIRD[2] model with time-varying parameters.

$$\begin{cases} \dot{S} = -\beta(t)\dfrac{SI}{N} \\ \dot{I} = \beta(t)\dfrac{SI}{N} - \gamma(t)I - \mu(t)I \\ \dot{R} = \gamma(t)I \\ \dot{D} = \mu(t)I \quad , \end{cases}$$

suitably equipped with initial conditions at time t = 0, where $S = S(t)$, $I = I(t)$, $R = R(t)$, and $D = D(t)$ are four compartments in which the total population N is split, representing susceptible, infected, recovered, or dead individuals. Here N is a given value representing the total population, and we have that for any $t \geq 0$ $S(t) + I(t) + R(t) + D(t) = N$.

In particular, we considered the recovery parameter $\gamma(t)$ as a constant, as using different functional forms was not rewarding. We then took inspiration from the recent work of Ianni and Rossi [7] for the choice of the parameterization of the contagion parameter $\beta(t)$ as a decreasing exponential. This form actually helps in the modelling of different kinds of restrictive measures and particularly the so-called «hard lockdown», as it entails a varying basic reproduction number[3]. When considering the *first wave* outbreak of March, it also makes sense to consider a decreasing trend in the lethality parameter $\mu(t)$. This choice in fact models the increase of medical expertise acquired during the first outbreak and the consequent availability of new treatments. In the following we summarize our parametric choices:

$$\gamma(t) = \gamma_0 \tag{14}$$

$$\beta(t) = \beta_0 e^{-\omega t} \tag{15}$$

$$\mu(t) = \frac{\mu_0}{t+1} \tag{16}$$

[2] *Susceptible, Infected, Recovered, Dead*

[3] The basic reproduction number in epidemiology contexts is also known as R_t and it is computed as $R_t = \frac{\beta(t)}{\mu(t)+\gamma(t)}$

When considering the *second wave* outbreak we set the lethality parameter $\mu(t)$ as constant, instead. Another parameter of interest is the initial population size N, as well as the initial condition $S(0)$, which is usually unknown, however impacting on the evolution of the epidemic described by the model.

3.2 Bayesian model

In order to perform an effective uncertainty quantification in this context we needed an appropriate hierarchical Bayesian model. As described in Section 2, we considered a likelihood-free approach and so we report here the priors structure only:

$$\beta_0 \sim Gamma(\lambda_\beta \tilde{\beta_0}, \lambda_\beta) \tag{17}$$

$$\gamma_0 \sim Gamma(\lambda_\gamma \tilde{\gamma_0}, \lambda_\gamma) \tag{18}$$

$$\mu_0 \sim Gamma(\lambda_\mu \tilde{\mu_0}, \lambda_\mu) \tag{19}$$

$$S_0 \sim Gamma(\lambda_{S_0} \tilde{S_0}, \lambda_{S_0}) \tag{20}$$

$$\omega \sim \mathcal{N}(\tilde{\omega}, \lambda_\omega) \tag{21}$$

$$\lambda_\beta \sim \mathcal{U}(\alpha_\beta, \beta_\beta) \tag{22}$$

$$\lambda_\gamma \sim \mathcal{U}(\alpha_\gamma, \beta_\gamma) \tag{23}$$

$$\lambda_\mu \sim \mathcal{U}(\alpha_\mu, \beta_\mu) \tag{24}$$

$$\lambda_{S_0} \sim \mathcal{U}(\alpha_{S_0}, \beta_{S_0}) \tag{25}$$

$$\lambda_\omega \sim \mathcal{U}(\alpha_\omega, \beta_\omega) \tag{26}$$

with $(\tilde{\beta_0}, \tilde{\gamma_0}, \tilde{\mu_0}, \tilde{S_0}, \tilde{\omega})$ being the prior means for the parameters. These values were inferred from the China outbreak for the *first wave* predictions and from previous Italian estimates for the *second wave* results.

We considered Gamma priors having as mean the just described parameters and random variance (given by the λ parameter, uniformly distributed) for strictly positive parameters and a Normal prior for the exponential rate of the contagion parameter β, as it could be in principle even negative. This model actually performed very well, giving us the necessary flexibility to obtain a coherent sampling from the posterior, while taking into account previous knowledge on the problem exploiting the power of Bayesian statistics. In section 4 more details on priors are reported.

3.3 SEIRD Model

We also tried to fit a SEIRD[4] model, with a non observed compartment (the *Exposed* one). We report here for completeness its differential form:

[4] *Susceptible, Exposed, Infected, Recovered, Dead*

$$\begin{cases} \dot{S} = -\beta(t)\dfrac{SI}{N} \\ \dot{E} = \beta(t)\dfrac{SI}{N} - \alpha(t)E \\ \dot{I} = \alpha(t)E - \gamma(t)I - \mu(t)I \\ \dot{R} = \gamma(t)I \\ \dot{D} = \mu(t)I \end{cases}$$

The time-dependent parameters in common with the SIRD model have hereby the same functional form, while the α parameter, representing the incubation rate for the transition from exposed to infected, is assumed to be constant. We used a similar set of priors as the one reported in the case of the SIRD model. In Section 4 we report a comparison between the SIRD and SEIRD results, as well as more details on priors.

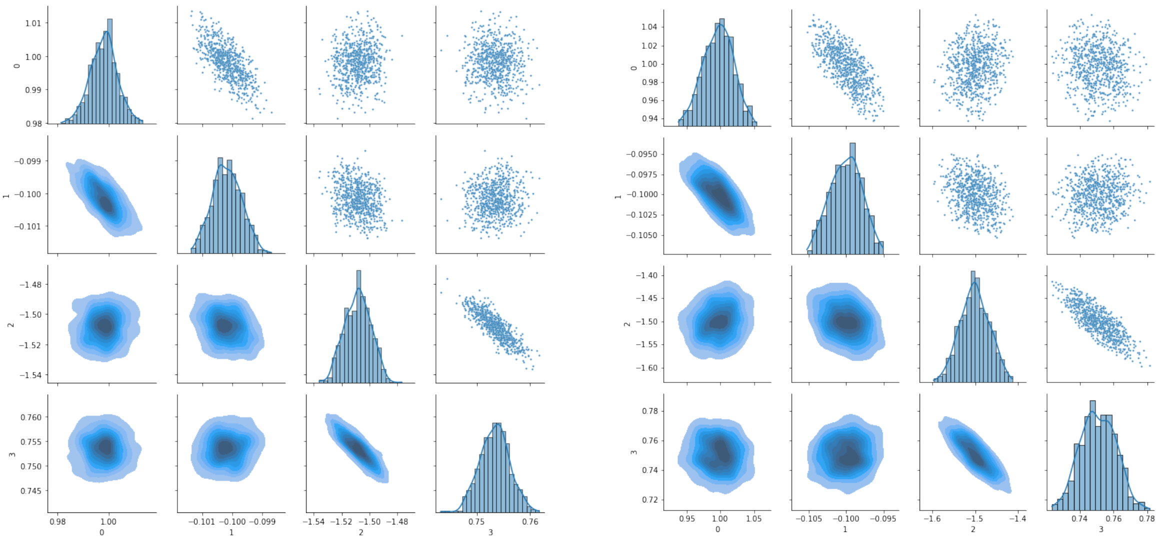
4 Results

4.1 HMC vs. ABc-SMC on a benchmark test case (Lotka-Volterra model)

First, we considered the comparison of the HMC and ABC-SMC methods in a simulated context, which was already presented in Perdikaris et al, the Lotka-Volterra model, defined as:

$$\begin{cases} \dot{x}_1 = \theta_1 x_1 - \theta_2 x_1 x_2 \\ \dot{x}_2 = \theta_3 x_1 x_2 - \theta_4 x_2 \end{cases}$$

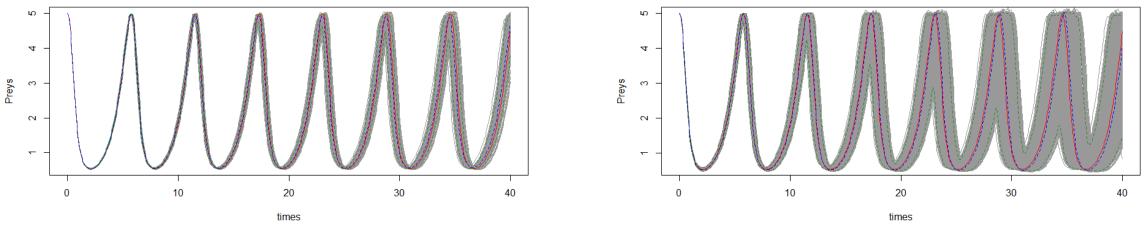
for $t > 0$, equipped with suitable initial conditions for $x_1(0)$ and $x_2(0)$. The two phases represent the number of individuals in a prey and a predator group, respectively. We assumed that both compartments are observed, and that observations are affected by a normally distributed, simulated, noise. The goal is to identify the parameters $\theta_1, \ldots, \theta_4$ starting from a set of noisy observations across a time interval $[0, T]$. Even though this is a simple toy problem, it already presents some challenges, as parameters are highly correlated between them, resulting in a more difficult sampling from the posteriors. First, we adapted the code structure presented in the original Perdikaris paper [1] to run it on our machines, updating the code from *Tensorflow*1 to *Tensorflow*2. Then, we noticed that the sampling presented in the original paper suffered of a non-negligible autocorrelation problem, so we devoted our efforts to solving it, by changing the mass matrix in Hamiltonian Monte Carlo and turning it to an adaptive scheme. After having tested its performances in model selection (with lasso-like priors) and having solved the aforementioned issues, we then compared these results with the empirical ABC-SMC framework we developed. The two posteriors look really similar from a qualitative (Figure 1) and quantitative (Table 1) point of view. While the uncertainty related to the eABC-SMC framework is higher (Figure 2), mainly due to the a priori chosen epsilon schedule and termination criterion, the eABC-SMC algorithm offers a remarkable computational speedup. In fact, the pairplots in Figure 1 were generated in 1h05 minutes of computation for HMC and 21 minutes for ABC-SMC. This speedup can be further enhanced by parallelizing the code, as shown in Section 4.3. Moreover, regarding HMC, as mentioned in the previous section, the batch feeding mechanism does not work well with how



(a) Pairplot with HMC.

(b) Pairplot with ABC-SMC.

Figure 1: A qualitative comparison of the estimated parameters with HMC and ABC-SMC.



(a) Simulation with HMC.

(b) Simulation with ABC-SMC.

Figure 2: Simulation in time for the Preys compartment with observed samples up to time t = 20.

real epidemiological data are structured, suggesting to proceed with eABC-SMC for this task.

	θ_1	θ_2	θ_3	θ_4	θ_1	θ_2	θ_3	θ_4
True	1	-0.1	-1.5	0.75	1	-0.1	-1.5	0.75
MAP	0.9979039	-0.1003933	-1.506464	0.7529937	0.996897	-0.098455	-1.506832	0.7596772
Rel. error	2.0961e-3	3.933e-3	4.309e-3	3.9916e-3	3.103e-3	1.544e-2	4.55e-3	1.290e-2

Table 1: Parameters' estimates for HMC (on the left) and ABC-SMC (on the right).

4.2 Covid-19 pandemic forecasting in Italy

Then, we proceeded with our final goal of building an accurate tool for emergency response in the pandemic outbreak. This is not an easy task because of the notorious unreliability of data and more subtle identifiability problems [10]. Therefore, for this first part, we used data from the Covid-19 epidemic in Italy from 24/02/2020 to 23/07/2020, as recorded by John Hopkins University (Source: https://github.com/ CSSEGISandData/COVID-19_Unified-Dataset). While these data can be considered accurate enough for model fitting, especially compared with other countries' data, in the first phase there are many known problems, such as under-reporting due to limited testing and different fidelity of data between weekdays and Sundays. As our method is built with an eye for parameter estimation with possibly unreliable or noisy data, we include the possibility to specify the reliability of the data at each instant, for every compartment, by selecting the relative standard deviation of the data. This precaution is taken, when needed, in the following results, for example by considering more uncertain the data coming in on Sundays, which are normally underestimated because of the lower amount of testing processed on Sundays.

4.2.1 Predictions at day 30

All taken into account, we proceed with our first objective, trying to predict the epidemic course of the disease using only data from the first 30 days (data from 24/02/2020 to 25/03/2020), with a great attention paid to the time and magnitude of the peak of infected individuals, which is of paramount importance in order to decide the harshness and duration of countermeasures and provide reliable estimation of the maximum load on hospitals and healthcare system. Looking at the literature, we decide to first fit a SIRD model for the prediction in a hard lockdown situation, adopting the functional forms described in Section 3.1. When proceeding with the numerical fitting, we found out that the equations were overparametrised, causing eABC-SMC to struggle with the sampling, but, as we found out some parameters were completely colinear between them, we were able to reduce the number of parameters, while not losing in predictive accuracy. Before working on the Italian data, though, we leveraged on the information on the pandemic in China, obtaining an estimate of the parameters for the pandemic in China, where it had started with one month earlier, to use as meaningful prior knowledge for the location of our prior distributions. The resulting fit is really accurate for all the compartments, with a total R^2 of the punctual prediction 0.96, and it manages to capture the exact location in time of the peak, estimating it at 95% confidence between 52 and 58 days. We highlight that the peak occurred, actually, at day $t = 55$, and that predictions carried out after 30 days from the outbreak start last spring had estimated the peak ten days earlier, with a much lower amount of infected individuals. If we run our code on March, 25, we would have correctly estimated the peak of the first outbreak in Italy as the center of the credibility interval. Indeed, the true value of infected individuals at peak is predicted with adequate accuracy, as the true value (108257) falls in the 95 % credibility bands (103642, 141132). Moreover, our strategy performs impressively well in estimating the asymptotic behavior of the recovered compartment, even though the variance is high in the prediction when the considered time-span increases. It is also interesting to see the estimated behavior of the often quoted R_t parameter, which is one of the main indicators of the spread of the epidemic. From Figure 5, we see how our framework provides a credible estimate for the peak's location in time, locating it with 95 % confidence to the interval (10,11), which corresponds to the exact weekend when the total lockdown started in Lombardy, however slightly overestimating the absolute value of R_t at peak.

Given the simple nature of the epidemiological model at hand, we have shown that a sound combination of prior assumptions and a thorough application of a well-designed Bayesian strategy can, in principle, provide extremely accurate predictions of the epidemic peak well in advance, by relying on data acquired until two weeks before the expected peak time. Not only, the estimated model also provides extremely good predictive capabilities.

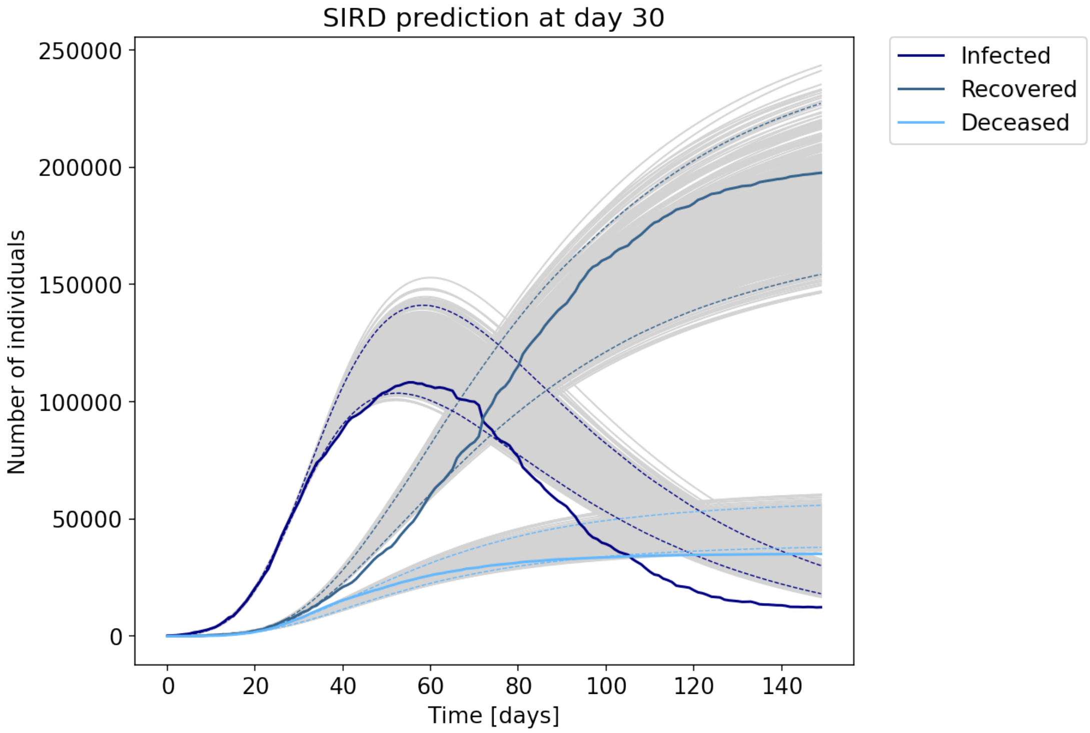


Figure 3: Simulation in time with data observed until t=30 with a SIRD model. 95 % credibility bands for each compartment are represented dashed. In gray, all the simulated trajectories.

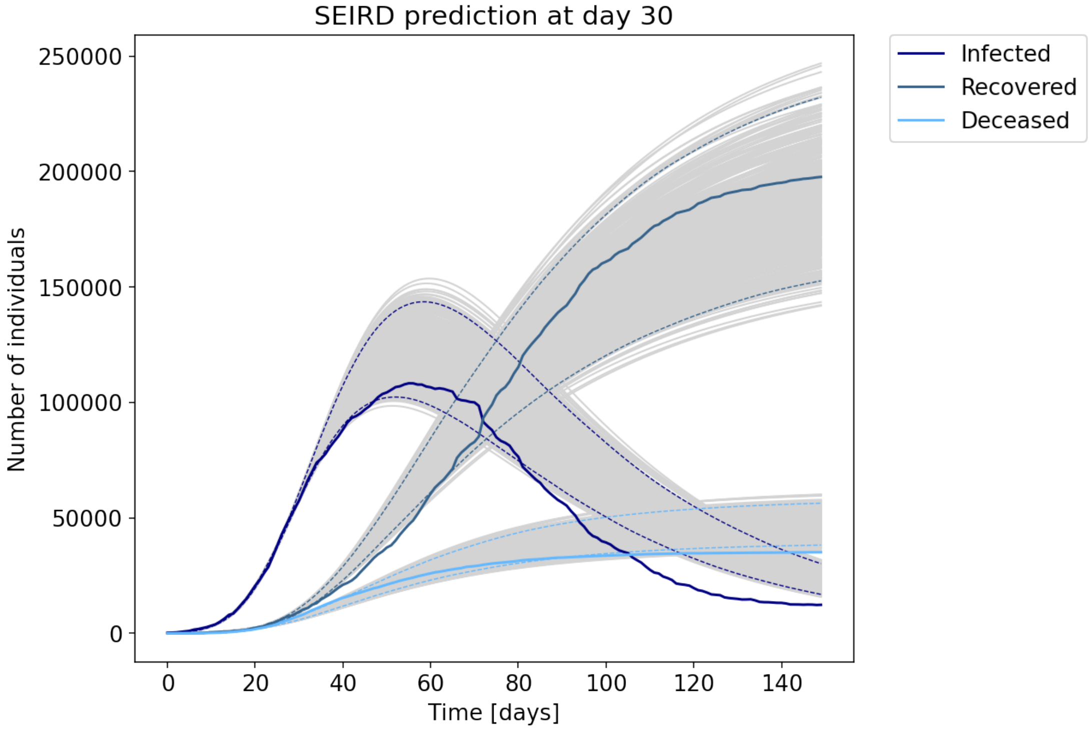


Figure 4: Simulation in time with data observed until t=30 with a SEIRD model. 95 % credibility bands for each compartment are represented dashed. In gray, all the simulated trajectories.

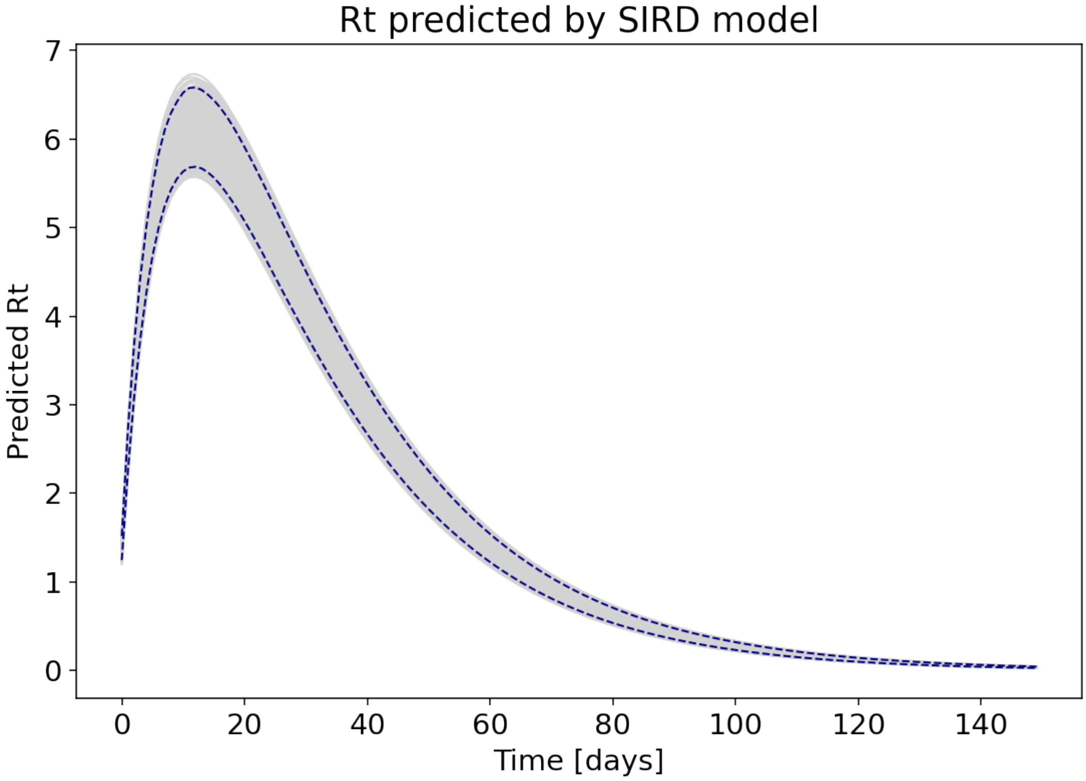


Figure 5: Simulated R_t parameter with data observed until t=30 with a SEIRD model. 95 % credibility bands are represented dashed. In gray, all the simulated trajectories.

4.2.2 SIRD vs SEIRD

We then tried to see whether a more complex model like SEIRD, which adds a compartment of exposed, not yet infected people, to account for the incubation period of the virus, could help describe better the pandemic, also testing whether our framework can deal well with non-observable states. So, after adapting the framework for the unobservable state corresponding to exposed individuals, we proceeded with the fit with the same procedure as before. While being slightly computationally heavier, the model does not outperform the simpler SIRD model and provides really similar predictions (with a marginally higher R^2 in pointwise prediction), albeit with a slightly higher variability, because of the hidden state (we did not observed the E compartment, while S, I, R and D were observed). To confirm this impression, we compute the ratio between the Predictive Bayesian Residual (PBR) for the two models for all points after t=30, taking into account the three compartments and obtain

$\frac{PBR_{SIRD}}{PBR_{SEIRD}} = 1.000019.$

A visual comparison is also displayed in Figures 3 and 4, while the estimated 95% credible intervals for the main parameters are reported in Table 2. Even SEIRD correctly estimates the peak in (52,58) and provides a similar (102312,143552) estimate for the peak of infected individuals. Because of its higher complexity without providing significant advantages, we proceeded in our analysis using the SIRD model.

	S_0	β_0	μ_0	ω	γ	α
SIRD	[355254,821657]	[0.424,0.452]	[0.260, 0.342]	[0.0178,0.0218]	[-0.0440,-0.0412]	NA
SEIRD	[414335,1394248]	[0.487,0.516]	[0.270, 0.346]	[0.0179,0.0222]	[-0.0492,-0.0464]	[2.73,2.90]

Table 2: Parameters' 95% HPD for SIRD and SEIRD. S_0 indicates the estimated susceptible population, β_0 the initial contagion parameter, μ_0 the mortality parameter, ω the exponential decay rate of β, γ the recovery parameter, α the incubation parameter, only present in the SEIRD model.

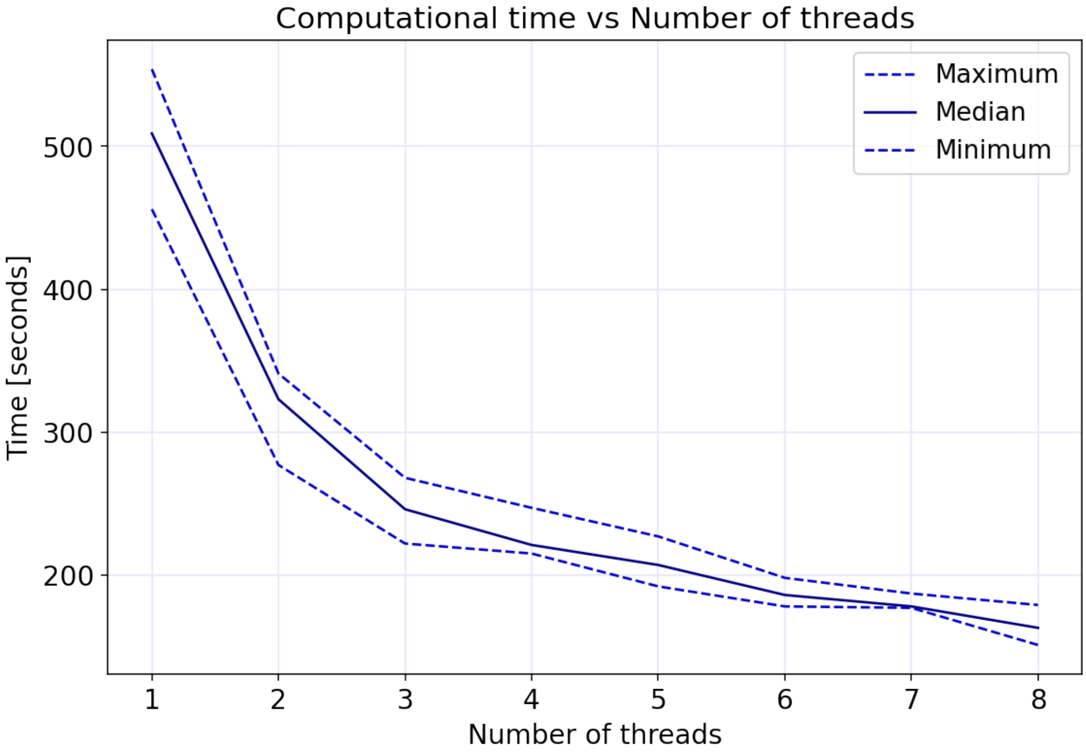


Figure 6: Computational time with the increase in number of used Python processes. Sampling run on a Intel i7 8^{th} gen, with 8 logical cores.

4.3 Computational speedup

For its nature, ABC-SMC is a highly parallelizable algorithm, because of the indipendent sampling. We therefore implemented a simple multiprocessing solution using the *Joblib* Python library, distributing the execution by starting separate Python worker processes to execute tasks concurrently on separate CPUs, using the *cloudpickle* protocol for serialization. The results show a consistent improvement in speed, as seen in Figure 6 and Table 3.

4.4 Model Robustness

In order to stress-test the robustness of our model, we tried to apply our model to countries which adopted a similar behavior to Italy during the first wave of the Covid-19 pandemic. However, the data in most countries with similar behavior (e.g. France, Spain) show several issues. We still managed to fit a model using Spain's data using the first 30 days as in the case of Italy for the fitting (Figure 7). We underline that the model the model adapts quite well to the data,

N threads	Minimum	Median	Maximum
1	456	509	554
2	277	323	341
3	222	246	268
4	215	221	247
5	192	207	227
6	178	186	198
7	177	178	187
8	151	163	179

Table 3: Computational performances based on 10 executions, with best and worst run discarded.

even in the presence of a massive fluctuation in the reported values, managing to locate the peak well, answering to the main purpose it was designed for, and simulating a plausible asymptotic behavior. However, there is there is evidence of possible troubles in reported data concerning the period $t > 80$ days, that become unreliable, as the Spanish government stopped providing numbers for the recovered patients.

4.5 Regional Differences

During the *second wave* in Fall 2020, the Italian Government introduced a colour code for the classification of risk for each Region and adopted, consequently, different restrictive measurements according to that classification. Another interesting application of our model is in the application of our framework to this data, trying to understand whether there were differences between different coloured zones, and if the different enforced restrictions indeed had different impacts. We considered in particular Lombardia (mostly been a *Red* Region) and Veneto (always been *Yellow* Region), two demographically similar regions, bordering regions. We considered the already described SIRD model with the only difference of an assumed constant mortality rate on all the available data, as we now aim at investigating the phenomenon, rather than predicting its evolution. The fitting results are extremely satisfying also in this case, showing the goodness of the proposed model. Only the dead compartments presents some issues, mainly due to the scale difference and the used priors, which come from the mortality rate of the first wave, thus resulting in an overestimation of the mortality. This is, however, not a relevant issue, as our main interest is, in this case, on the other two compartments. In particular, we focused on the $\beta(t)$ parameter with the aim of investigating whether it were increasing or decreasing, testing the ω parameter, governing the form of the exponential. Therefore, we tested for the two regions the two alternative hypotheses:

$$\begin{cases} H_0 : \beta(t) = \beta_0 e^{-\omega t} \nearrow & (\omega \leq 0) \\ H_1 : \beta(t) = \beta_0 e^{-\omega t} \searrow & (\omega > 0) \end{cases}$$

By performing the tests on our sampled posterior distributions, we obtained the following Bayes Factors for the test:

Lombardia: $2log(BF_{0,1}) : -2.54$ **Veneto**: $2log(BF_{0,1}) : 2.59$

thus confirming a general increase in the contagion speed in Veneto and a general decrease in Lombardia, while always finding a lower mean β in Lombardia than in Veneto. The presented analysis thus suggests that a classification of regions according different degrees of severity of the epidemic trend was justified, however implying extremely outcomes and medium-term evolutions, too.

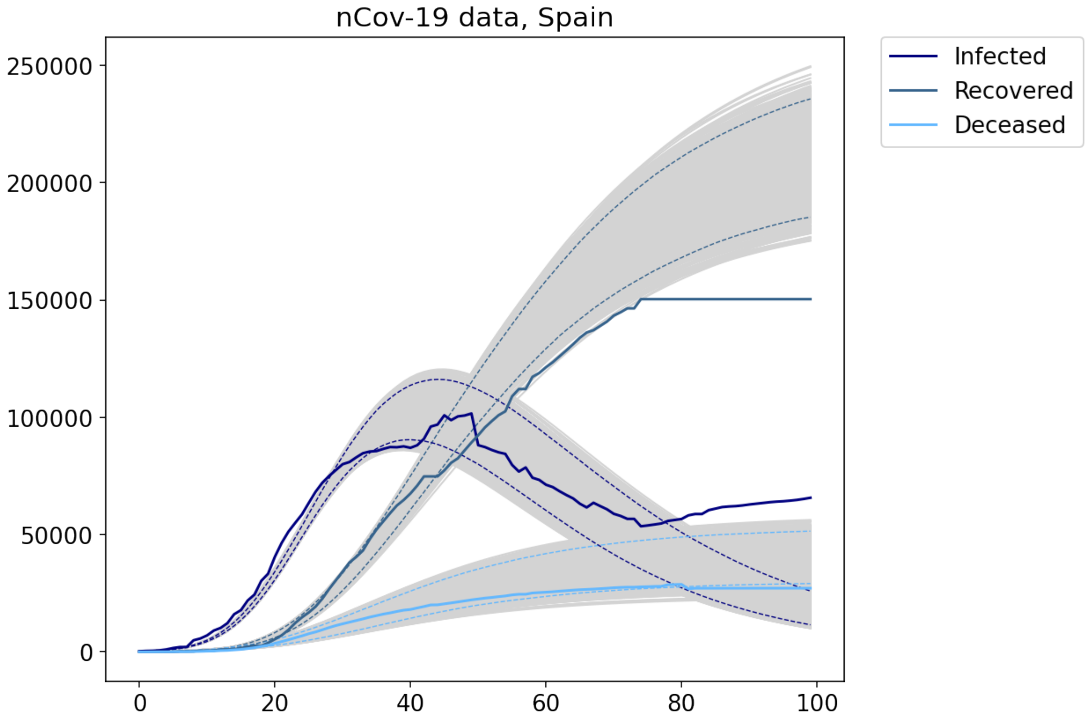
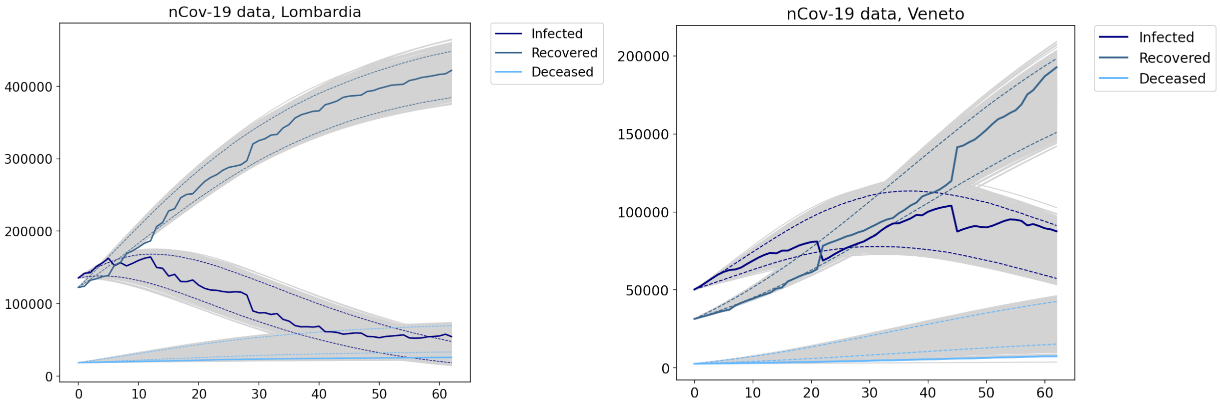


Figure 7: Simulation in time with data observed until t=30 with a SIRD model. 95 % credibility bands for each compartment are represented dashed. In gray, all the simulated trajectories.



(a) Lombardia data

(b) Veneto data

Figure 8: Simulation for Lombardia and Veneto with a SIRD model for the same temporal frame (06/11/20 - 08/01/21).

5 Conclusions

In this work, we addressed the problem of data-driven model discovery from noisy observations of nonlinear dynamical systems (or ODEs systems), in terms of both model identification and parameters estimation, exploiting two different Bayesian frameworks. A first approach [1] takes advantage of recent developments in differentiable programming to propagate gradient information through ODE solvers and Hamiltonian Monte Carlo (HMC) sampling. A second approach deals instead with Sequential Monte Carlo (SMC) sampling in the framework of Approximate Bayesian Computation for ODEs [2,3,4]. We carried out a comparison between these two approaches on a simple two-dimensional benchmark test case dealing with the prey-predator Lotka-Volterra model, finding that the ABC-SMC method outperforms the HMC-based method, and does not pose any issue when dealing with real, noisy data. Moreoever, we provided an improvement to the original ABC-SMC framework, consisting of an *empirical* preconditioning, which helps in reducing the computational effort by avoiding to sample from regions with almost null probability of acceptance. Furthermore, we built a code library implementing eABC-SMC for epidemiological models leveraging on multiprocessing, applying it successfully to a real-world complex problem.

Then, we focused on the data-driven identification of mathematical epidemiological models described in terms of nonlinear systems of ODEs, such as the well-known SIRD model. In particular, we have addressed several situations, ranging from the so-called first outbreak (spring 2020) to the second outbreak (fall 2020) of the covid-19 pandemic in Italy, aiming at providing a ready-to-use emergency response tool. A comparison between «yellow zone» and «red zone» regions has been also discussed. Thanks to the Bayesian data-driven framework we developed, several results have been achieved, such as a tight estimation of the peak of the first outbreak based on data recorded on infected and recovered compartments, and a clear distinction of the epidemic trends as a consequence of different measures taken in two Italian regions.

Possible advancements include (i) testing more complex epidemiological models like the recently proposed SUIHTER [10] or SIDARTHE [11], taking in consideration, for instance, compartments for asymptomatic or hospitalised individuals, as well as (ii) trying to perform system identification employing data on few observed compartments, or again (iii) including a measure of under-reporting on acquired observations. Moreover, the problem could be structured on a graph, including data from different regions or countries and analysing the importance of connections on the spread of the virus. Finally, the analysis could be repeated by considering age stratification between the different compartments and, including new, incoming data, assess the impact of the vaccine in the parameters' behavior.

References

[1] Y. Yang, M.A. Bhouri, P. Perdikaris (2020). Bayesian differential programming for robust systems identification under uncertainty. ArXiv pre-print, submitted to Proceedings of the Royal Society A..

[2] Toni, T., Welch, D., Strelkowa, N., Ipsen, A., Stumpf, M. P. (2009). Approximate Bayesian Computation scheme for parameter inference and model selection in dynamical systems. *Journal of the Royal Society Interface* 6(31), 187-202.

[3] Filippi S., Barnes C. P., Cornebise J., Stumpf M. P. H. (2012). On optimality of kernels for approximate Bayesian computation using sequential Monte Carlo. arXiv:1106.6280 [stat.CO] .

[4] Toni T., Stumpf M. P. H. (2009). Tutorial on ABC rejection and ABC-SMC for parameter estimation and model selection. arXiv:0910.4472 [stat.CO] .

[5] A. A. Alahmadi, J. A. Flegg, D. G. Cochrane, C. C. Drovandi, J. M. Keith (2020). A comparison of approximate versus exact techniques for Bayesian parameter inference in nonlinear ordinary differential equation models. *R. Soc. open sci.* 7: 191315.

[6] A. Look, M. Kandemir (2020). Differential Bayesian Neural Nets. arXiv:1912.00796v2 [cs.LG].

[7] Ianni, A., Rossi, N. Describing the COVID-19 outbreak during the lockdown: fitting modified SIR models to data. *Eur. Phys. J. Plus* 135, 885 (2020).

[8] Hoffman MD, Gelman A. (2014). The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research* 15, 1593–1623.

[9] C. Piazzola, L. Tamellini, and R. Tempone. A note on tools for prediction under uncertainty and identifiability of sir-like dynamical systems for epidemiology. *Mathematical Biosciences*, page 108514, 2020.

[10] N. Parolini, L. Dede, P.F. Antonietti, G. Ardenghi, A. Manzoni, E. Miglio, A.Pugliese, M. Verani, A. Quarteroni. SUIHTER: A new mathematical model for COVID-19. Application to the analysis of the second epidemic outbreak in Italy. arXiv:2101.03369 (2021).

[11] Giulia Giordano, Franco Blanchini, Raffaele Bruno, Patrizio Colaneri, Alessandro Di Filippo, Angela Di Matteo, and Marta Colaneri. Modelling the covid-19 epidemic and implementation of population-wide interventions in italy. *Nature Medicine*, pages 1–6, 2020.

[12] D. Smith, L. Moore. The SIR Model for Spread of Disease - The Differential Equation Model, *Mathematical Association of America.*