1 Introduction

Since its outbreak in late 2019 in Wuhan, China the COVID-19 virus has spread all over the world and has challenged the social and health systems of many countries. The pandemic has also triggered a huge interest from all disciplines of science and countless researchers have contributed to the better understanding of the characteristics of this novel disease. An astonishing number of more than 29000 COVID-19 related contributions is available on the WHO website and all of these articles where published after the first of January 2020.

Those contributions cover an extremely wide range of aspects of the pandemic: from the behavior of the virus on a microscopic level and it's lifespan on different surfaces to the impact of the lockdown on the psychology of the population. One very important aspect of the pandemic is obviously the understanding of it's evolutions and consequently it's modeling and forecasting. Modeling is very important because it is the only way for decision-makers to implement the right policies at the right time. A robust model can help to choose what kind of measures to implement, like school closing or travel ban and also when to stop these measures.

The goal of this work is to explore the usage of data-driven system identification tool for COVID-related data. The most common way of modeling a disease is through compartmental models. They divide the population into different compartments like susceptible (not yet infected by the disease), exposed (infected but not yet infectious), infected, recovered. The interaction processes between the different compartments are governed by a set of simple ODEs. This framework was introduced almost a century ago and is flexible enough to model most of the effects and controls of the disease. The classic epidemiological approaches will be further introduced in section 2.

In this work we deliberately choose to not base our modeling on these classic techniques and concentrate on the identification of dynamic processes in the evolution of the disease with very little a priori. The different approaches considered in this work all gravitate around an algorithm proposed in [**?**] called *system identification of nonlinear dynamics*. The details about this methodology will be explained in section 3. In a few words, this algorithm allows to identify the dynamics from observed data through a sparse regression and based on a user-defined library of candidate functions. Throughout this work, we will use this algorithm in combination with datasets collected from different sources in order to better understand both the dynamics of

the pandemic and the specificities as well as the limits of this algorithm.

This algorithm can allow us to do all kinds of interesting analyses.

2 Classic modeling methods

It was briefly mentioned in the introduction that the most classic way of modeling an infectious disease is based on dividing the population into compartments. The most basic model of this kind is the SIR model that tracks the number of susceptible, infected and recovered people in the sample. It was originally proposed in [**?**] and has become the most widely used epidemiological model discussion about micro and macro levels. This very simple model is formulated by the following set of equations:

$$\begin{aligned}\dot{S} &= -\beta SI \\ \dot{I} &= \beta SI - \gamma I \\ \dot{R} &= \gamma I\end{aligned}$$

where β and γ are the parameters of the disease. $1/\gamma$ is the time a person remains infectious while β quantifies the infectiousness of the disease. The ratio β/γ is usually denoted R_0 and called the basic reproduction value. It is a very important characteristic of the disease because it represents the average number of people a single person is susceptible to infect. All the measures introduced by the governments in order to limit the impact of the disease were targeted at reducing this R_0 value.

These models can be further elaborated in several ways. The first sophistication can be introduced by adding additional compartments like *exposed, quarantined, hospitalized, deceased* to further mimic the real interactions between these different categories of the population. It is clear that COVID-19 has a lot of delay in it's dynamics [**?**] because it can have an incubation time of up to two weeks. In order to take this into account it might be necessary to add at least an "exposed" compartment. Thus, when infected, a person is not immediately infectious and must stay a few days in the "exposed" compartment. A few examples of more sophisticated SIR models and adapted specifically to COVID-19 can be seen in [**?**] but as mentioned earlier there is a huge number of such adaptations.

Secondly, knowing that the different age groups of the population are affected in very different ways by the disease [**?**], it might be interesting to subdivide the population into the relevant age groups in order to have a more precise control over the parameters. For example the mortality rate is almost equal to 0 for children and is very high for the people above 80 years of age. Age subdivision is also a widely used practice and can be seen in [**?**] just to name a few.

Thirdly, one very classic way of making the models more complex and precise is by subdividing the region of interest into smaller sub-regions and to have a state for each of these sub-regions. For example a country can be subdivided into regions or counties. Some parameters traducing the migrations between these regions can be introduced. One can imagine that this kind of modeling can be very useful for policy-makers to adopt more precise and local measures. Few examples of such models can be found in [**?**]. These models are usually called *spatial SIRs*.

Finally, there is no limitation to the complexity of the models that can be built on the basis of the SIR models. They can be combined and adapted to a very wide range of problems. An example of such use might include adding the relevant compartments to model the usage of a tracking mobile app that facilitates the isolation of people who have interacted with another infected person.

2.1 Other modeling techniques

The modeling of infectious diseases is however not limited to the aforementioned compartmental models and lots of different approaches from different disciplines exist. Other approaches for modeling the spread of an infectious disease include:

- **Statistical models** are more inclined towards capturing the random nature of the infectious diseases. Thorough reviews of these approaches can be found in [].
- **Gravity models** can be used as standalone models or in conjunction with the compartmental approach and are called "gravity" models because they rely on considering the effect of distance and the size of donor and recipient communities [].

- **Network-based models** are built on the assumption that the spread of human disease follows its specified contact or spreading paths such as transportation or social contact networks [].
- **Agent-based models** are based on the belief that individuals and their mutual differences are the key to understanding the spread of an infection [].

Additionally, hybrid approaches combining different methods are considered in the literature [] as well as more computationally-oriented methods like cellular automata []. A thorough review about the modeling of infectious diseases can be found in [].

3 Sparse identification of nonlinear dynamics

In this section the central algorithm used in this work will be introduced. This algorithm lies on the crossroads between dynamic modeling and machine learning. On one hand this algorithm is able to fit the observed data much like more classic machine learning algorithms would do. And on the other hand it does so by finding governing ODEs in observed data. In theory this allows to model an arbitrarily complex phenomenon while having the interpretability of a set of simple ODEs. We will see that in practice, when dealing with real-world data and complex problems this technique doesn't hold up to these expectations.

Machine learning is an extremely young branch of science as opposed to dynamic modeling, because mathematicians, physicists and engineers have been modeling physical phenomena for centuries. The shift that happened in the recent years is that (1) computational power has significantly increased and now allows for computations of unprecedented complexity and (2) observation data is getting increasingly available as sensor prices go down. The combination of these two tendencies opens new horizons for modeling increasingly complex processes.

It was quickly mentioned in the introduction that this algorithm has two key features: (1) it relies on a user defined set of candidate functions and the result will only be as good as the candidate functions are with respect to the problem; (2) it uses sparse regression (some weights are gradually zeroed-out) in order to find the most parsimonious formula that still fits the data.

The whole methodology to discover the dynamics can be divided into 3 steps. Let's say that we have a series of observations $(\mathbf{x}(t_1), \mathbf{x}(t_2), \ldots, \mathbf{x}(t_m))$ where $\mathbf{x}$ is the state or the vector state of the system of interest and m is the number of observations. Our goal is to find a function f such that:

$$\dot{\mathbf{x}} = f(\mathbf{x}) \tag{1}$$

1. It is more convenient to work with matrices to do the computations so our first step will be to divide the available data into two matrices:

$$X = \begin{bmatrix} \mathbf{x}(t_1) \\ \mathbf{x}(t_2) \\ \vdots \\ \mathbf{x}(t_m) \end{bmatrix} \text{ and } \dot{X} = \begin{bmatrix} \dot{\mathbf{x}}(t_1) \\ \dot{\mathbf{x}}(t_2) \\ \vdots \\ \dot{\mathbf{x}}(t_m) \end{bmatrix}$$

   here the dot notation denotes the time-derivative and it can be either directly observed or computed numerically.

2. The second step of this algorithm is the augmentation of the state with the candidate functions. Let's say that we have a set of p candidate functions $(f_1, f_2, \ldots, f_p)$ that we want to use for this identification. What we need to do is to construct a matrix $\theta(X)$:

$$\theta(X) = \begin{bmatrix} f_1(\mathbf{x}(t_1)) & f_2(\mathbf{x}(t_1)) & \cdots & f_p(\mathbf{x}(t_1)) \\ f_1(\mathbf{x}(t_2)) & f_2(\mathbf{x}(t_2)) & \cdots & f_p(\mathbf{x}(t_2)) \\ \vdots & \vdots & \ddots & \vdots \\ f_1(\mathbf{x}(t_m)) & f_2(\mathbf{x}(t_m)) & \cdots & f_p(\mathbf{x}(t_m)) \end{bmatrix}$$

3. Finally we can run the optimization and find the best set of $\xi_1, \xi_2, \ldots, \xi_p$ that minimize the following equation in the least squares sense:

$$\dot{X} = \theta(X) \times \Xi \text{ where } \Xi = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_p \end{bmatrix}$$

   After the best $\xi_1, \xi_2, \ldots, \xi_p$ were identified, the very small ξ (those that are under a certain threshold called *cutoff value*) can be removed from

the equation alongside with the candidate function they correspond to. And then the least squares minimization can be ran again with the new subset of candidate functions and weights. After doing this thresholding a few times the algorithm converges.

This is the very general definition of the algorithm, in our case we used almost exclusively polynomial terms of the state as candidate functions. For example if our state is $\mathbf{x} = (s, i, r)$ like in the compartmental models, then our candidate function be: $\mathbf{x} \mapsto 1, \mathbf{x} \mapsto s, \mathbf{x} \mapsto i, \mathbf{x} \mapsto r, \mathbf{x} \mapsto s^2, \mathbf{x} \mapsto si, \mathbf{x} \mapsto sr, \ldots, \mathbf{x} \mapsto r^n$ where n is the maximum degree of the polynomial terms that we consider. If the state of the system is $\mathbf{x} = x$ like if we just tracked the cumulative number of cases in a single country, then the candidate functions would be: $\mathbf{x} \mapsto 1, \mathbf{x} \mapsto x, \mathbf{x} \mapsto x^2, \mathbf{x} \mapsto x^3, \ldots, \mathbf{x} \mapsto x^n$. A more thorough discussion about polynomial terms is held in section 3.1.2.

Another remark is that this algorithm works equally well with an iterative formulation as it does with the differential one like in equation 1. In the iterative case we would have a series of observations $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m)$ and the goal is still the same but we want f to express the dynamics in an iterative way:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t) \tag{2}$$

This is slightly more convenient because it prevents us from numerically computing a derivative which obviously introduces some errors.

This work was further extended in a second paper [**?**] that is more centered on PDE identification instead of working with ODEs.

3.1 Hyperparameters

If we consider this algorithm as a machine learning methodology then its hyperparameters would be:

1. the choice of the set of candidate functions;
2. the cutoff value.

Both these parameters deserve a discussion.

3.1.1 Cutoff value

We previously mentionned that the cutoff value is the parameter used for thresholding the small weights in the learning process. In fact, this parameter allows us to choose the sparsity of the resulting model and thus choose the required balance between accuracy and complexity of the model.

In practice, due to the low computational cost of this algorithm[1] TODO: run the timing we are able to compute the results for a very wide range of cutoff values and to choose the one that we prefer. Usually the one satisfying the philosophy of "Ockham's razor", in other words the lowest cutoff value that has satisfying fitting of the data. An example of such a plot can be seen on figure **??**.

further discussion

3.1.2 Candidate functions

In a simple problem, we just want our candidate function to include the real dynamics. The authors mostly use polynomial terms as candidate functions because the dynamics they identify in the examples are actually polynomial terms of the variables of the state. In the 7 examples of identification that they provide, the candidate functions always include the real dynamics.

In more general cases and more complex problems, we cannot guarantee the same thing. And instead we use polynomials as approximators and hope that they will be expressive enough to capture the evolution. Obviously, it is very important to make this distinction between making a library large enough to include the real dynamics and using polynomials as universal approximators.

Our choice of polynomial basis functions was motivated by the fact that the authors propose their usage, that they are quite simple to use and understand and finally that the study of basis functions is not completely in the scope of this work. Consequently, the question of whether they are the more relevant candidate functions in our case remains not fully answered. Indeed an elaboration was explored by using rational functions instead of polynomials. The full discussion is available in appendix **??**. Our main conclusion from this superficial experiment was that with similar maximum degrees, rational functions provided little to no advantage over the polynomial ones in

[1] Less than one second for 100 data points and 10 candidate functions on our i7 machine.

terms of mean squared deviation while requiring a less-convenient non-linear least squares fitting.

4 Data

Data plays a crucial role in our approach because our aim is to base all the modeling on the observation data. In this section we will describe what kind of data we used for our modeling.

4.1 Observing the pandemic

Because this pandemic has no precedents, the metrics people used to track the state of the disease have evolved during the course of the disease. For example, on the date the redefinition of what a "case of infection" is according to the WHO led to a huge spike in new cases in China [].

5 Methodologies

6 Remarks