Contents

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 and allows for arbitrarily complex models. 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.

The ability to identify the dynamics from data is a fundamental challenge. Even though the centuries of research have given a good understanding of surrounding physical and natural phenomena, there are plenty of more complex systems that are yet to be modeled. Examples of such systems might be the patterns in weather and climate, epidemiology, power grids, etc. [**?**]. The present work is an attempt at applying such a dynamics identification tool on real world data. However, it is important to highlight that, as mentioned earlier, the field we are targeting (epidemiology) already has some very powerful modeling frameworks. In this context it is very hard to achieve interesting results in terms of modeling.

The rest of this report is constructed as follows: in section 2 we present the classic and already existing modeling approaches for epidemiology. In section 3 we thoroughly describe the system identification algorithm on which this work is based. Then in section 4 we present and discuss about the data that we use in our modeling. In the next section 5 we talk about the different modeling approaches that we envisioned for this work and describe the results. After this, in section 6 we comment on the results and try to give some elements about the reasons our methods are challenging to use. Finally we sum-up our conclusions in section 7.

2 Classing epidemiological modeling techniques

2.1 SIR models

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. This very simple model is formulated by the following set of equations:

$$\begin{aligned} \dot{S} =& -\beta SI \\ \dot{I} =& \beta SI - \gamma I \\ \dot{R} =& \gamma I \end{aligned} \tag{1}$$

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.

2.2 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.

3.1 Algorithm

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:

$$\dot{\mathbf{x}} = f(\mathbf{x}) \tag{2}$$

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 have an absolute value 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 3.2.2.

Another remark is that this algorithm works equally well with an iterative formulation as it does with the differential one like in equation 2. 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:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t) \tag{3}$$

This is slightly more convenient because it prevents us from numerically computing a derivative which obviously introduces some errors. This is achieved by dividing the two matrices in the following way:

$$X = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_{m-1} \end{bmatrix} \text{ and } X_2 = \begin{bmatrix} \mathbf{x}_2 \\ \mathbf{x}_3 \\ \vdots \\ \mathbf{x}_m \end{bmatrix}$$

and minimizing $X_2 = \theta(X) \times \Xi$ instead.

One can notice that this algorithm is very simple and sound quite intuitive. This work was further extended in a second paper [**?**] that is more centered on PDE identification instead of working with ODEs.

3.2 Hyper-parameters

If we consider this algorithm as a machine learning methodology then its hyper-parameters would be:

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

and both these parameters deserve a discussion.

3.2.1 Cutoff value

We previously mentioned 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] 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

[1] 20.2 ms ± 359 µs for 100 data points and 10 candidate functions on our i7 machine.

3.2.2 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 1. 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 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 February 13th 2020 the redefinition of what a "case of infection" is according to the WHO led to a huge spike in new cases in China. This example highlights the fact that the observation data in not completely

reliable. Furthermore, due to the big differences in healthcare systems in different countries as well as on a regional scale, it can be unfair to compare the numbers. An example of such difference might be the testing policy. Indeed countries around the word have implemented vastly different testing approaches, from very selective testing of only symptomatic people to massive testing like in Germany and Korea. This difference in the number of tests leads to a big difference in the test positivity rate. The higher the test positivity rate the less we are confident about our real understanding of the scale of the epidemic.

In our study, we mainly used the *cumulative number of cases*. Basically, it is the total sum of people that got infected in a given region, its evolution usually looks like a logistic function. Not to be confused with the "I" in an SIR model which accounts for the number of currently infected people. Tracking the evolution of the pandemic through the cumulative number of people is convenient because (1) it is the minimum information we need to have an idea about the state of the disease in the population and (2) even if the quality of the data is not perfect it is the most widely available information (for example, the number of recovered people is missing or is not reliable for a big number of countries).

The most widely used resource for tracking the number of cases worldwide is the repository of the John Hopkins University. They have collected the numbers from all over the world and provided required APIs for all researchers and visualization dashboards to use. They also provide information about recovered cases but as mentioned earlier, the quality of the data is very doubtful.

4.2 Observing the government measures

In our modeling approach we also need to take into account the measures that were taken by the governments to prevent further spreading of the disease like school closing, travel ban, work from home recommendations and alike. For this we relied on the information gathered by a group of researchers from Oxford [**?**]. They tracked the measures that were implemented in most of the countries worldwide and provided a convenient way to retrieve this information as indicators and also they compiled all of this information into a "stringency" index. This index is a linear combination of all the indicators that they observed and it's meant to evaluate how strict are the measures implemented by the governments but it has no information about their ef-

fectiveness or performance.

4.3 Other information

Since the adopted methodology of this work heavily relies on the principles of machine learning and data science, we included additional datasets into our modeling because usually, the more relevant information we have, the better the results. For example, in one of our experiments we chose to make a model that would take into account the trajectories of all available countries (more about this in section 5.3). In order to inform the model about the present country, we added more than 40 indicators relevant for the pandemic about the country (health, hygiene and demographic descriptors). A few examples of those descriptors can be: the human development index, the total population, the number of hospital beds per 1000 people, the number of people of 65 years and above, percentage of people using at least basic sanitation services, percentage of death caused by injury and so on. This information was available to us through the World Bank API [].

5 Modeling

In this section we will introduce the different experiments that we did with the tools and data that we described earlier.

5.1 Cumulative number of cases and forecasting

The most obvious and easy way of using the system identification algorithm with COVID-19 data is to model the number of cumulative cases in a single country. If we define $\mathbf{x}_t$ as the number of cases in the country of interest at day t. Then, we want to find a function f so that:

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

When using polynomial terms as candidate functions we would look for a purely polynomial formulation of the dynamics because there is only one variable in the state. Which can be mathematically written as:

$$\mathbf{x}_{t+1} = \sum_{k=0}^{p} \xi_k \mathbf{x}_t^k$$

where p is the maximum degree of the polynomial terms.

For example we can apply this method to the evolution of the cumulative number of cases in Germany an this is the formula we will end-up with rerun:

$$\mathbf{x}_{t+1} = 4.07 \cdot 10^{-3} + 1.24 \cdot \mathbf{x}_t - 3.33 \cdot 10^{-2} \cdot \mathbf{x}_t^2 + 1.66 \cdot 10^{-3} \cdot \mathbf{x}_t^3 - 3.21 \cdot 10^{-5} \cdot \mathbf{x}_t^4$$

assuming a maximum degree of 4. The figure **??** represents the trajectory of the real data as well as the trajectory of the identified model. One can see that this model fits fairly well the observation data and that the extrapolation looks fairly plausible. Identified models in all countries are available in appendix 2.

Given that the identified model is extremely simple and has only one variable we cannot retrieve much useful information from it apart from the extrapolation of the number of cases in the next few days. For this reason, the main application of these country-wise models is forecasting. In order to have a better idea of how these very simple identified models compare to more advanced forecasting techniques we compared them to an auto-regressive integrated moving average (ARIMA) statistical model. We chose this particular model because there are examples of authors using it for COVID forecasting [**?**].

The experiment was build as follows: we identified the dynamics of the cumulative number of cases in 118 countries with maximum degrees of the polynomials ranging from 2 to 7. We kept the last two weeks of available data out of the training set in order to evaluate and compare the models on a one and two weeks forecast horizon. We trained an ARIMA$(1, 2, 1)$ check model on the same training data in each of the countries and then we compared the results. The trajectories in all countries were rescaled so that the last value of the time-series is 1. This allows the models to more easily learn the parameters and to be able to fairly aggregate the mean squared deviation to do quantitative comparisons.

Results The forecasts of the ARIMA model are fairly similar in terms of mean squared error to the result of the best performing identified system (the one that has the maximum degree that performs best on a given country). In terms of frequency of best results, the ARIMA model seems to have better results (in 55% of the countries ARIMA has the best forecast for both forecast horizons). In terms of errors both methods are also quite similar but the ARIMA model is still slightly better that the dynamic models. A more

thorough description of the results of this experiment is available in appendix 3.

Limits of the experiment The ARIMA forecasting model requires that the time-series verifies some assumptions like stationarity and that there are no other predictors verify. The statistical model is also not very easy to parametrize and requires the data to be preprocessed for better results. Given that we have done a very minimalist subset of these manipulations it would not be fair to quantitatively compare the results. This poor parametrization is due to the fact that it requires a good knowledge of these statistical tools. In other worlds: there is a big chance that the parameters of the ARIMA model are sub-optimal and thus we conclude that system identification is not likely to give better forecasting results than the statistical model because in the current parametrization it is already worse.

5.2 Control measures

In this part we will report about the second experiment: adding control measures to the equation in order to find how the dynamics are influenced by these. We rely on the information provided by [] as described in section 4.2. Since these indicators are represented as ordinal values we can directly include them into the equation as variables:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, i_1, i_2, \ldots, i_7)$$

where $i_1, i_2, \ldots, i_8$ are the government response indicators. They are encoded as integers from 0 to 4 representing the strictness of the measure. The table 1 shows a list of all those indicators.

The motivation behind this approach is that the measures implemented by most of the countries had a huge impact on the evolution of the disease. We have seen that modeling the trajectories without taking the control into account is possible but has very limited results and applications. Here we want to see if we can discover what kind of effects are introduced into the evolution when the controls are applied.

In order to do this, we augment our state with the control variables and we consider them as non-predicted variables which means that we only seek to predict the cumulative number of cases. This can be achieved by adding the controls as columns in our X matrix:

Variable	Name
i_1	School closing
i_2	Workplace closing
i_3	Cancel public events
i_4	Restrictions on gatherings
i_5	Close public transport
i_6	Stay at home requirements
i_7	Restrictions on internal movement

Table 1: Indicators of government measures

$$X = \begin{bmatrix} \mathbf{x}_1 & i_1(1) & i_2(1) & \cdots & i_7(1) \\ \mathbf{x}_2 & i_1(2) & i_2(2) & \cdots & i_7(2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_{m-1} & i_1(m-1) & i_2(m-1) & \cdots & i_7(m-1) \end{bmatrix}$$

Then, the augmented matrix $\theta(X)$ will be created with polynomial terms. Applying polynomial candidate functions in equivalent to retrieving all the combinations with replacement of r elements where r is the maximum degree of the polynomial terms and doing the product of the sample. And since the number of combinations with replacement is equal to $C^R(n,r) = \dfrac{(n+r-1)!}{r!(n-1)!}$ we are going to end-up with $C^R(9,3) = 165$ candidate functions (assuming a maximum degree $r = 3$). Here n represents the number of variables and it is equal to $9 = 1 + 1 + 7$ because we introduce a 1 constant into the variables wich allows the polynomial terms to be of all required degrees. A more thorough explanation of the implementation of the polynomial terms can be found in appendix 4.

Simplification Polynomial candidate functions include lots of terms that are constant with respect to the cumulative number of cases $\mathbf{x}$ (for example $1, i_1, i_2, \ldots, i_7, i_1^2, i_1 \times i_2, \ldots, i_7^r$) and in order to simplify the problem we can make the assumption that there is no constant part in the dynamics. In fact, we can justify this by the observation that there is no reason for the cumulative cases to augment if there were no cases on the day before. Thus, we drop all of these terms from the augmented matrix we keep only those that are not constant with respect to the number of cumulative cases

$(\mathbf{x}, \mathbf{x}i_1, \mathbf{x}i_2, \ldots, \mathbf{x}i_7, \mathbf{x}^2, \mathbf{x}^2 i_1, \ldots, \mathbf{x}^r)$. With this trick we are able to leave out some number candidate functions which makes the model considerably simpler.

Results This

5.3 Global model

The third experiment is about trying to have a model that governs the dynamics of the evolution of the number of cumulative cases in all available countries. This approach was mainly motivated by the fact that in the early stages of the epidemic it would've been very interesting to have a general idea of how the pandemic will evolve. One way of achieving this is by making a model that is able to collect knowledge from the most advanced trajectories (at that time China and Korea) and to apply the learned patterns to other countries. This global approach is an attempt at extracting common knowledge from the trajectories.

From a machine learning point of view, in order to implement this approach, we need to inform the model about the country we are currently trying to make predictions for. We chose to encode the countries as vectors of about 30 dimensions with the country indicators that we gathered from the World Bank Database as described in section 4.3. The "stringency" index was also added to inform the model about the measures. We end-up with a total of more than 40 variables and as explained in the previous sub-section this will lead to a huge number of polynomial terms. Consequently, the simplification that was already described earlier where we do not consider the terms that are constant with respect to the state will be applied here as well. To sum things up, we are once again looking for the best function f such that:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, s_t, c_1, c_2, \ldots, c_{40})$$

where s_t is the stringency of the country at day t and $c_1, c_2, \ldots, c_{40}$ are the country indicators (they are not time-dependent).

Twofold goal It is important to highlight the fact that this current global approach is very ambitious. On one hand we want the model to learn the usual sigmoidal evolutions of the number of cumulative cases. On the other hand we want the model to learn the mapping from the vector representation of the country into the characteristics of the evolution of the disease in this

country. In other words, we expect the model to identify the patterns between the descriptive indicators and the impact of the pandemic on the country. For example, the model could learn something like: the smaller the HDI the bigger the overall number of cases (or more realistically the opposite relation). We also expect the model to understand more simple things: since the total population of the country is also in the indicators we want the model to understand that the number of cumulative cases cannot surpass the total population. More importantly, the two aforementioned tasks are fundamentally different and asking a single model to handle both of them at the same time might not be completely appropriate. Indeed, one task is time-series forecasting and the other is just general regression.

We followed a standard practice for machine learning where we divide the dataset into a train and test subset. Thus we kept 20% of the countries out of the training data in order to verify the performance of our model. Our testing strategy is to give the vector representation of a country and some initial value of cumulative cases as well as the real stringency index and then we can see if the generated trajectory makes sense compared to the real one. This means that we look at the trajectories that the model predicts for countries it has never seen, allowing us to verify that the model works for our initial goal that was to predict the overall evolution from the country indicators.

Results Unfortunately, we were not able to make this approach work, similarly to the precedent experiment we observe that the model highly overfits to the training data. This mean that on the training countries the model is suspiciously good whereas on the test set the trajectories do not make sense. A more precise presentation of the results can be found in appendix 5.

Overfitting One of the ways to tackle the overfitting was to compute a "relevance" score for all of the country indicators and to choose the most relevant subset of indicators in order to decrease the number of polynomial terms and consequently the complexity of the model. After a full normalization of all the input variables we can be sure that the learned parameters fairly represent the overall weight of the candidate function in the formula. Then, for each variable, we can compute the sum of the absolute values of the coefficients of all the candidate functions where this variable is present. We end-up with a score representative of the importance of this variable in

the learned formula. It is natural to make a ranking of the variables based on this score. Interestingly, we observed that the ordering between different runs was not very consistent which strengthened our belief that the model failed to learn the underlying patterns.

5.4 Other approaches

We didn't limit ourselves to the 3 experiments explained above, we also considered the following formulations:

- Time dependent dynamics: $\mathbf{x}_{t+1} = f(\mathbf{x}_t, t)$. The idea behind this approach is that the parameters of the evolution heavily changed over the course of the pandemic when the measures were applied. The intention was to allow the model to understand this time-dependence by providing the time variable instead of explicitly providing the stringency like we did before with very limited success. The risk that we introduce with this is that the learned formula becomes completely time-dependent and is no longer a differential equation. Appendix 6 gives a better understanding of this experiment and shows some nice vector-field visualizations.

- Modeling based around SIR quantities, by having a richer state like $\mathbf{x} = (s, i, r)$ or even more elaborate. The motivation behind this approach is the analogy with the SIR models. They require at least these 3 quantities to explain the evolution which means that looking exclusively at the number of cumulative cases might not be enough to explain the dynamics. For some reason this approach didn't work at all. We will discuss a few reasons why this failed to enhance the results in section 6.1.

- Actual SIR parameter identification. During this work, it was necessary to compare the behavior of our models to the classic SIR models. For this reason we found ourselves fitting parameters for compartmental models. Both SIR and SEIR models were fitted. The encountered challenges are discussed in section 6.2.

- Neural networks were also used in this work for learning the f function in a single country setting. In this case we just checked that a simple feed-forward NN was not better than the identified models. Even

though NNs are able to approximate any function, in this case they appear overly complex compared to a simple polynomial. Their training time is vastly larger than system identification while not providing any interpretability.

The usage of a N-BEATS [**?**] model was also considered while not being completely in the scope of our work. This model was proposed a few years ago and it represents the beginning of fully deep models for time-series forecasting, as opposed to earlier models that where more inspired by statistical modeling. We didn't pursue this path very long because it would not have the same interpretability as our models had and it would be difficult to understand the patterns from the dynamic point of view. Finally, the N-BEATS model doesn't provide the possibility to add non-time-dependent data out of the box. TODO(ODE-net, Res-net)

- TODO($(n-1)(n-2)$)

6 Discussions

6.1 Identifying dynamics from SIR quantities

We briefly mentioned above that when using a state $\mathbf{x} = (s, i, r)$ the results are rather disappointing. In this section we are going to try to give a few elements on why this happens.

Generated data An easy learning setting can achieved by using generated data instead of real-world observation data from the COVID-19. For example, one can integrate the SIR ODEs with some predetermined parameters and initial conditions in order to check that we are able to recover the parameters from the fitting. The easiest integration would be with constant parameters, as opposed to a real-world scenario where the parameters are dependent on the measures and consequently they are not constant. This kind of experimentation with fake data is very useful because it allows us to have a better idea about the performance of the tools and methodologies that we use. We applied this kind of sandbox experimentation for the identification of the dynamics from generated SIR trajectories and surprisingly we never managed to recover the original ODEs. Depending on the cutoff value we can have either a very complex or very simple formula be we never

managed to converge to the original formulation as described in equation 1. Nevertheless, if the formula has more than 4 or 5 terms the system is able to closely fit the data while having a different ODE and the more complex the system is the better the fitting. This is probably due to the fact that the candidate functions have some redundant terms and that a lot of equally well fitting formulations exist. The problem with those non-original formulations is that we cannot be sure that their behavior is identical outside of the training domain. Then when the cutoff value surpasses some threshold, the system becomes too sparse and the model no longer fits the data.

Although we didn't give an exact answer to why adding the SIR quantities to the state doesn't give better results, we gave a very important element that almost excludes the possibility of doing such fitting with the methodologies of this work. One way to tackle this problem could be to find a more sophisticated way to regularize the model than the sparse regression with the cutoff value we are currently using. Indeed, we suspect that this kind of regularization might be leading the model into a local minima because of the too early exclusion of important terms.

Real data Needless to say that if the methodology doesn't work with some very simple generated scenarios there is no chance that it will work with real trajectories. However, we can use this opportunity to discuss some of the challenges that are introduced by real data. We discussed some of the limits in the observation data in section 4.1. In the case of fitting from an SIR state the results are even more influenced by the inherent biases of the data. The clearest example of such biases can be seen in the measurement of the number of recovered people that usually matches the theoretical number very loosely. This error is introduced by the fact that in most of the countries, there is no checking of whether an infected and quarantined person is still infectious or not. Consequently the numbers are updated after some time when officials decide to announce that some part of the infected population can be considered recovered. For this reason, it is not very convenient to base the modeling on such data.

The second evident challenge that is introduced with real data is the fact that that the basic reproduction value R_0 is time-dependent. When fitting the parameters of an actual SIR model we can take this time dependency into account and handle it by fitting a sliding window or a model of the basic reproduction number. But in the case of system identification with no

a priori it is much harder to do so.

6.2 Challenges with SIR fitting

The major problem when fitting parameters for SIR models is the discrepancy between real observations and the parameters of the model. The number of infected people "I" is not available in our data sources, instead we have the number of total people that were infected (we usually refer to it as the cumulative number of cases). In order to find "I", we can say that it is equal to the number of cumulative cases minus the number of recovered people, but this works only for the countries where the number of recovered people is reliable. And then the problem is that in reality if someone is tested positive then he is immediately quarantined whereas the model supposes that this person might infect other people before he recovers from the disease. That's why the actual time someone remains in the "I" compartment is smaller than the usual lifespan of the virus.

Similar problems are encountered when fitting more elaborate models like SEIR, SIRU, SEIRD, and so on, where the number of exposed people is not observed and requires assumptions. This problem can be tackled by adapting the model to the actual values that we observe in real life and adding special compartments for unobserved quantities that are later optimized to give the best possible fit on the real and observed data. This is what allows epidemiologists to give estimations about the number of asymptomatic people for example.

6.3 Different goals

Throughout this work, we had very different goals in mind. When we modeled the evolution of the number of cases in a single country we wanted our models to be as accurate as possible for forecasting purposes. Whereas when we made the "global" model we weren't as concerned by the accuracy and instead would've preferred models that would capture the real underlying patterns instead of overfitting to the observations. This distinction between the different goals that we can have in modeling is fundamental. When we develop a modeling framework, we define the scope of the effects and interactions that are taken into account by the model and the performance metrics are defined with respect to this scope. In our work, we never had an

very clearly defined scope, thus we evaluated the performance of our models by comparing them to the existing ones that have a similar scope (like forecasting for the single country modeling).

7 Conclusion

Drawing conclusions from not very successful experiments is no easy task. Throughout this report we emphasized how hard and unfruitful it was to use system identification tools for the COVID-19 data. However we can never be sure that we have tested the techniques thoroughly enough to conclude that system identification is not useful for epidemiology. In this conclusion we want to highlight some major points that are worth remembering from this exploratory work.

TODO(An actual conclusion)

Acknowledgments

The present work is not a paper but a report about my internship for my master's engineering degree at IMT Mines Ales. In this section I add some contextual information about the internship.

Context

Appendix

1. Rational basis functions `https://kipre.github.io/files/internship/reports/non-linear/nonlinear.html`
2. Simple trajectories in countries around the world `TODO`
3. Forecasting COVID-19 cases `TODO`
4. Computing polynomial terms `TODO`
5. Worldwide model `TODO`
6. Time-dependent dynamics `https://kipre.github.io/files/internship/reports/covid_time/index.html`