The inverse problem of a dynamical system solved with genetic algorithms

The inverse problem of a dynamical system solved with genetic algorithms

Ricardo Medel Esquivel[1,2], **Isidro Gómez-Vargas**[1,2], **Teodoro Rivera Montalvo**[1], **J. Alberto Vázquez**[2] **and Ricardo García-Salcedo**[1]

[1]CICATA-Legaria, Instituto Politécnico Nacional, 11500, Ciudad de México, México.

[2] ICF, Universidad Nacional Autónoma de México, 62210, Cuernavaca, Morelos, México.

E-mail: rmedele1500@alumno.ipn.mx

Abstract. In this work, we propose a complete methodology to identify the parameters of a dynamical system from a data set using genetic algorithms. Considering the search for the model parameters as an inverse problem, we numerically solve the differential equations of the dynamical system; each set of parameters is then considered as an individual of the population that evolves in the genetic algorithm. As a fitness function we use the distance L_1 between the numerical solution and the data.

1. Introduction

In many fields of science, events that cannot be repeated and it is not possible to have experimental control are studied. As a consequence, modeling these events requires the solution of an inverse problem. In science, an inverse problem is a process in which, from a set of observations, the causal factors that produced them are calculated. In adopting this strategy, the scientist must assume that the analytical model is correct; then the problem is to find the initial conditions or parameters of the model that best fit the observational data up to a certain margin of error. And it is particularly useful when observational data are a time series, that is, time-dependent data sets, such as daily records of some event.

Among others, some of the problems that we can analyze with this method are the following: coupled oscillators and systems analogous to them, such as electrical circuits or the mass flow between communicating vessels, dynamics of biological systems of the predator-prey type.

In all the previous problems, there is at least one conserved quantity: energy, mass, population, etc. The conservation of that quantity is translated mathematically to the need to model the described system by means of a set of coupled differential equations, characterized by certain parameters, which we wish to properly determine.

In this article we are interested in determining numerically the parameters of a dynamic model using a set of observations and a genetic algorithm (GA). Our intention is to propose a general methodology that allows us to use GA. Similar approaches can be found in [1] and [2].

For a description of the proposed methodology, we use the data from a problem called *Influenza in a boarding school* cited in [3] and [4], where they solve the same problem presented here, with a different method. The problem is the following: in January 1978, after a winter holiday, 763 male students returned to their boarding school. After one week, one of the students developed the flu, then two more followed, and so on. By the end of the month almost half of the students were sick. Most of the school had been affected,

and by mid-February the epidemic was over [4].

2. Problem formulation

In this section we present the bases of the SIR model and the data set used to implement the technique that we propose in this paper.

2.1. The SIR model

The SIR model consists of a system of three coupled non-linear ordinary differential equations that have no explicit analytical solution. This is the most basic model of infectious diseases, however, from its complete analysis very useful information can be extracted, which derives in public health policies [5].

In this model, developed by Ronald Ross, William Hamer and others in the early 20th century [6], a population is divided into three groups: Susceptible, $S = S(t)$, Infected, $I = I(t)$, and Removed, $R = R(t)$; hence the initials in the model name. The dynamical equations are as follows:

$$\frac{dS}{dt} = -aSI, \tag{1}$$

$$\frac{dI}{dt} = aSI - bI, \tag{2}$$

$$\frac{dR}{dt} = bI, \tag{3}$$

where a is the transmission rate and b is the average recovery rate. At any time the total population is $N = S + I + R$ and it remains constant as $\dot{N} = \dot{S} + \dot{I} + \dot{R} = 0$, where overdot implies derivative with respect to time t.

This system can be numerically integrated to obtain solutions such as those shown in the figure 1. From the beginning, the greatest interest in the study of this model arises from its

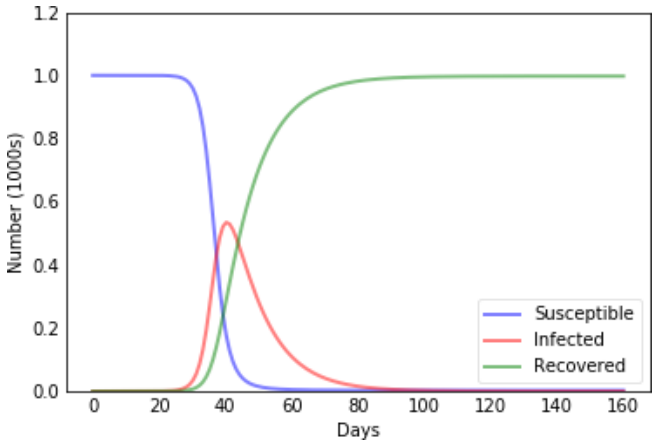

Figure 1. General form of the numerical solutions of the SIR model for an epidemic case.

ability to model epidemics. Because, even without knowing the analytical form of the solution, it is possible (using Dynamical Systems techniques) to know the behaviour of the solutions and to establish measures to control the growth of the infection.

Initially, $S(t)$ is approximately equal to N and $I(t)$ is very small. In an outbreak, typically $I(t)$ will increase every day until reaching a maximum and then tender decrease slowly, as seen in figure 1.

It is particularly useful to determine the reproductive number, defined as $R_0 = \frac{a}{b}$ since the analysis of the dynamical system indicates that [5]:

$$\frac{\mathrm{I}_{max}}{N} = 1 - \frac{1}{R_0}(1 + logR_0) \quad (4)$$

Thus, determining the parameters of the model leads to determining the maximum number of infected, a quantity that can be useful to develop public health actions. Even after the event has occurred, it is interesting to know the parameters *a* and *b* of the model, since they serve to characterize the disease. This is precisely the problem that we are interested in solving: determining the pair of parameters (*a*, *b*) of the SIR model from a data set.

2.2. Data

Marinov et. al. raises this same problem for a data set called reported in *Influenza in a boarding school* [4]. These data were first cited in [3] and it is reproduced in the table 1 and figure 2, where we also present the graph corresponding to these data, together with the solution obtained by Murray. According to [3], the values of parameters *a* and *b* are as follows: $a \approx 0.00218$ and $b \approx 0.4485$.

Table 1. Data of *Influenza in a boarding school* [3]

t	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
I(*t*)	1	3	6	25	73	222	294	258	237	191	125	69	27	11	4

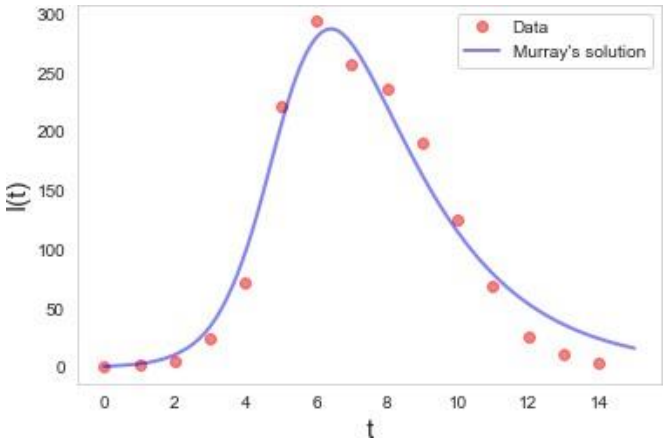

Figure 2. The numerical solution from the parameters obtained by Murray [3].

We use this data set as a preliminary test of the method. Later, we will try to apply it with more data and/or more difficult conditions. As can be seen, the data gives us the initial conditions, $I(0) = 1$, so, when determining the parameters of the system of equations, the model will be totally known.

3. Numerical methods

Genetic Algorithms (GA) are computational techniques inspired in the Darwinian evolution of biological species, and implement the biological notion of fitness, that is, the ability to survive and reproduce [7]. The figure 3 shows the flow chart of an GA.

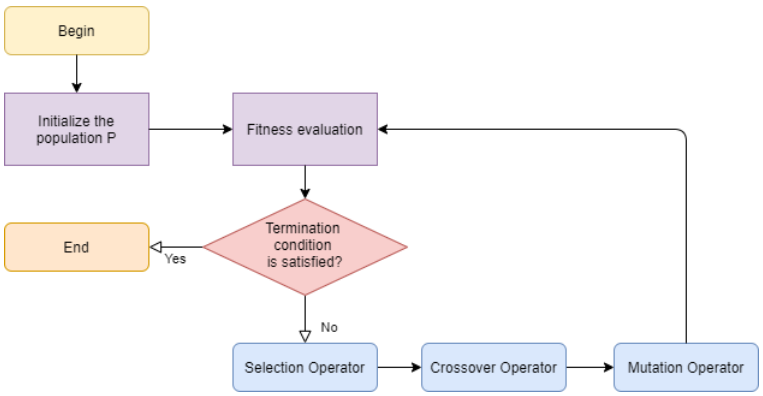

Figure 3. Flow chart of a Genetic Algorithm (GA).

In order to apply an GA, it is often necessary to code the solutions to the problem and define the search space; this coding is known as chromosome. The first step of the algorithm consists in the random generation of a population P of possible solutions (points in the search space).

The fitness function of each individual in the population is then evaluated. The fitness function must be carefully constructed to give us a direct measure of how good an individual is as a solution to the problem at hand. In this way, the evaluation of the fitness function allows us to identify and select the most promising individuals to solve the problem. The latter is done through a selection operator. There are several ways to construct such an operator. The individuals with the best fit are chosen to be passed on to the next generation and also to generate similar individuals through the crossover operator. The crossover operator generates new solutions from selected solutions. In general, from two-parent solutions it generates two child solutions. Here, there are also several ways to implement such an operator. GAs involve a third operator: the mutation operator. By analogy with the biological mutation, this operator randomly alters some individuals in the population.

The flow of GA is repeated over n generations. Some of the termination conditions may be that the number of generations reaches the n value, or that a threshold value in the accuracy of the best solution in the population is reached, or that the best solution does not change for a certain number of generations.

The chromosomes in our experiment consist of ordered pairs (a, b), corresponding to particular values of the SIR model parameters, taken randomly in the following interval: $(0, 1)$. This is established because it is known historically that in this range are the characteristic values for a and b of past epidemics. For each of these ordered pairs of the population, we numerically solve the system of differential equations of the SIR model; so that each ordered pair corresponds to a single numerical solution. Some examples of random individuals we obtained for our population are shown in figure 4. By inspection, we can see that some solutions are closer to the data than others. The GA's job will be to choose the individual that best approximates the data. The approximation to the data is quantified by evaluating the fitness function.

The fitness function can be chosen in many ways, we choose as a fitness function the Manhattan distance (or L_1 standard norm) between the numerical solution and the data at time t. Another option for the fitness function can be a Chi-square to measure the error of the numerical solution. The GA was implemented with the DEAP Python library [8] and is available in [9].

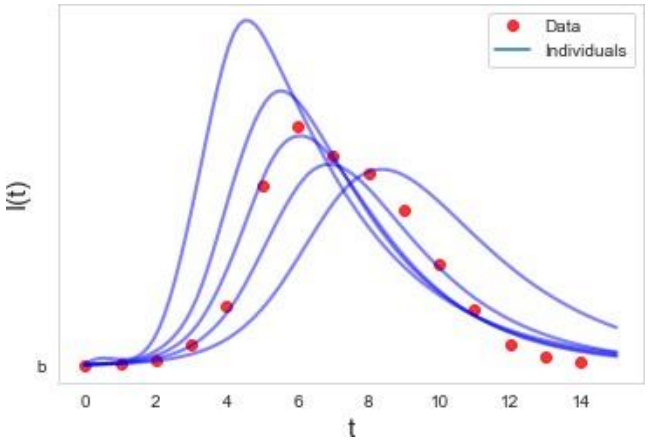

Figure 4. In our proposed procedure, each individual in the search space is coded by the parameter pair, *a* and *b*, of the differential equation system of the SIR model and is equivalent to a single numerical solution. Here are some examples of such individuals.

4. Results

A GA was implemented in which a population of 100 individuals was generated in a bidimensional search space of ordered pairs (*a*, *b*) in the interval (0, 1) over $n = 500$ generations, a crossover probability of 0.9 and a mutation probability of 0.4. We use the tournament method, which consists of choosing two individuals randomly from the population and selecting the one with the best fit.

In order to compare the results when using the Manhattan distance, the Chi-square distance was also analyzed as a fitness function, it was observed that the results do not vary significantly. Once we have implemented the GA, the values of parameters *a* and *b* that we obtained are as follows: $a \approx 0.00205$ and $b \approx 0.43552$. The figure 5 shows the best solution obtained, and also the evolution of the best fitness during the execution of the GA. As shown in right panel in figure

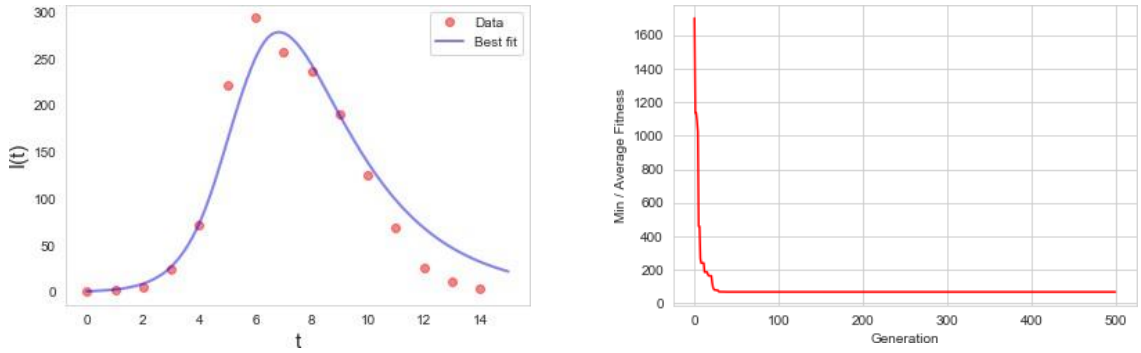

Figure 5. Left panel: Best individual (green line) obtained by the GA to adjust the data set. Right panel: Evolution of the best fitness (red) and the average fitness (green) of the population.

5, although the algorithm quickly reaches the best individual, the average fitness function of the population fluctuates, this is due to the value of the mutation operator, which is responsible for exploring new solutions.

5. Concluding remarks

In this paper we have presented a methodology that uses GA to adjust parameters in a simple dynamical system, such as the SIR epidemiological model, when a given data set is available.

The values for the parameters that we have found by this method are in agreement with those previously obtained by other statistical and mathematical techniques [4]. It is possible, for example, to perform a parameter adjustment using the Monte Carlo method, for readers interested in a basic exposition of this method we recommend [10] and [11]. However, we can comment that the relevance of the method used here is its ability to determine the model parameters without making a statistical approximation, which requires modeling the data, assuming that they follow a certain distribution. The use of GA releases us from that modelling process, and ensures that a good solution can be found when gradient-based optimisation methods fail.

The GA method presented in this paper can be applied to similar models and its use for large data-sets can only be limited by available computer resources. Precision can be improved by reducing time interval partitioning to smooth out numerical solutions.

There are aspects of the general problem raised here that are of great interest and that we will address in the future. One of them is the determination of parameters from incomplete or in-progress data. Something that is currently of great interest in certain scientific fields.

Another aspect of interest in this area is the solution of problems in which the initial conditions are unknown, something that can be extremely complicated for a system of higher order differential equations, where it is also possible that chaos can occur.

Acknowledgements

This work was partially supported by CONACyT-Mexico, SNI-CONACyT and the IPN project SIP-20200666. JAV acknowledges the support provided by FOSEC SEP-CONACYT Inves- tigación Básica A1-S-21925, and UNAM-DGAPA-PAPIIT IA102219. RGS acknowledges the support provided by CONACYT 2558591.

References

[1] González J A and Guzmán F S 2017 Inverse Problems of HIV cell dynamicsusing Genetic Algorithms IOP *Conf. Series: Journal of Physics: Conf. Series* **792** 012070

[2] Guzmán F S and González J A 2018 Use of Genetic Algorithms to solve Inverse Problems in Relativistic Hydrodynamics IOP *Conf. Series: Journal of Physics: Conf. Series* **1010** 012003

[3] Murray J D 2002 *Mathematical Biology* (USA: Springer)

[4] Marinov T T, Marinova R S, Omojola J and Jackson M 2014 Inverse problem for coefficient identification in SIR epidemic models *Comput. Math. Appl.* **67(12)** 2218–2227

[5] Weiss H 2013 The SIR model and the Foundations of Public Health *Materials Matematics* **3(17)**
[6] Anderson R M 1991 Discussion: the Kermack-McKendrick epidemic threshold theorem *Bulletin of mathematical biology* **53(1-2)** 3-32
[7] De Jong K A 2006 *Evolutionary Computation* (USA: MIT)
[8] Fortin, F A, De Rainville, F M, Gardner, M A, Parizeau, M, and Gagné, C 2012 DEAP: Evolutionary algorithms made easy *The Journal of Machine Learning Research* **13(1)** 2171-2175.
[9] Medel Esquivel R 2020 *Github repository* https://github.com/Medetl/InverseProblem
[10] Medel Esquivel R, Gomez Vargas I, Vázquez J A and García Salcedo R 2021 Accepted in *Boletín de Estadística e Investigación Operativa* 37
[11] Padilla L E, Tellez L O, Escamilla L and Vázquez J A 2020 arXiv:1903.11127