Forecast of outbreaks and calculation of the optimal flow of population: a scientific machine learning approach .

Matías Núñez
CONICET
Pandemov
Bariloche, Río Negro, Argentina
matias.nunez2@gmail.com

Florencia Grinblat
Pandemov
CABA, Argentina
florencia.grinblat@gmail.com

ABSTRACT

Humanity has been, throughout history, largely shaped, in different ways, by the outbreak of infectious diseases. The oldest way to fight against them has been social isolation, quarantine and soap. Currently, in the middle of the COVID 19 pandemic, isolation and restriction on the mobility of people continues to be a method used in various regions of the globe. Specially in countries with precarious health system which are easily overwhelmed. As the acquisition of data and information increases, so do the prevention systems We propose an approach based on ideas of scientific machine learning using neural differential equations (neural ODE) ([**?**]). This is a relatively new kind of network and among all the possible family of neural networks the reason for its selection is twofold. On the one hand we show that they serve as a detector of outbreaks and future dynamics, since due to their characteristics, they are capable of learning the dynamics of the system using relatively little data, and thus make accurate predictions. And on the other, once they have already learned the dynamics, the can serve as models to calculate the optimal flow of people between populations in such a way that the future of positive cases remain within a pre-established level. Among closed populations (states, provinces, countries, etc), which do not allow the entry or exit of people, this kind of approach can help to alleviate the measures of isolation between populations, and thus allow the movement of people and therefore activate stagnant economies.

NOTE: we heard about this challenge two weeks ago, thus we did not have much time. this is an incomplete draft, and a complete text can be found in our blog https://pandemov.blogspot.com/ and github https://github.com/pandemov/challenge .

1 APPROACH

Abstract (Briefly describe the background and overall purpose of the analytic approach) (3,500 characters max) * Please describe the background of your analytical approach and use of data set(s) (3,500 characters max)*

2 METHODS

We pre-processed the data using Data Science techniques with Pandas, Numpy and csaps written in Python . We used the data in the following link https://cmu-delphi.github.io/delphi-epidata/api/covidcast.html correspondant to the U.S. dataset(s) An example of the smoothing treatment performed to raw data before being used as inputs to the neural networks can be found in our github-link.(https://github.com/pandemov/challenge)

We used the neural ODE for forecasting future cases using previous data. The variables we used were mainly . A neural ODE learn the underlying dynamics, it considers a continuous setting and assumes that the change in is governed by an ODE

$$\frac{\partial y}{\partial t} = f(y;\theta) \tag{1}$$

to be related through some function

$$y_i = f(t_i;\theta)$$

where

$$\theta$$

are learnable parameters. The goal is to learn the underlying dynamics of change. If the dynamics do not change abruptly this has very powerful generalisation capabilities. It is a neural network inside an ordinary differential equation . The "forward pass" through a neural ODE is equivalent to solving an initial value problem, where

$$y(t_0)$$

is the input features and we replace hand-crafted equations with a neural network. A single forward pass gives us an entire trajectory in contrast to e.g. RNNs, where each forward pass through the model gives a single prediction in time.

The forward pass consist of inputting

$$y(t_0)$$

and using an ODE solver to step forward in time to solve

$$y(t) = y(t_0) + \int_{t_0}^{t} f^*(y(t))\partial t \tag{2}$$

where we use a neural network to model

$$f^*$$

In our case the parameters are learned from the data. See Fig. 1

We used the set of tools available in the Julia library DiffEqFlux (https://github.com/SciML/DiffEqFlux.jl).

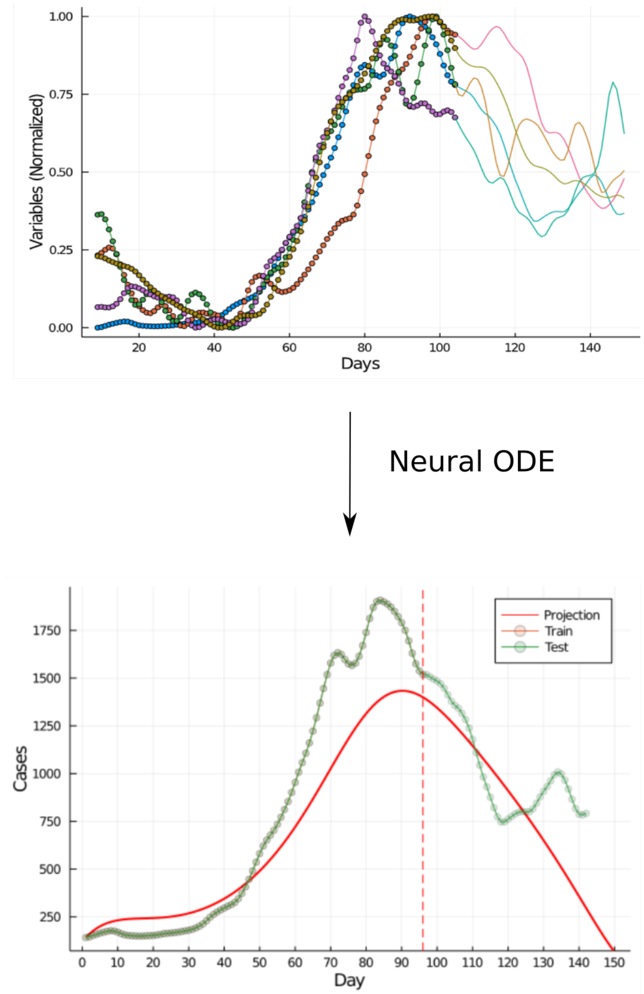
3 RESULTS

3.1 Relevant Figures and Graphs

UPLOAD graphs, charts, tables, etc. that visualize the findings of your analytical approach **??**.

3.2 LINK to Github for your analytical approach

https://github.com/pandemov/challenge



The Neural ODE learns from a set of training data correspondent to different chosen variables (above) which belong to a particular region (the state of SC in this case) and tries to learn the dynamics of the system by finding the ordinary differential equation that best describes the data. The learnt solution against the data for Active Cases in SC is shown below. The variables used, that best describe the dynamics are Hospital Admissions, Symptoms, Symptoms in Community, COVID Searches on Google, Doctor Visits and Cases.

Figure 1

4 DISCUSSION

Discussion and implications of findings (3,500 characters max)* UPLOAD a presentation (10 slides max) summarizing the analytic approach, findings, and the relevance of your approach to the COVID-19 disease control policy and practice

5 LIMITATIONS AND FURTHER WORK

Describe the limitations of the analytic approaches and discuss intended further work (3,500 characters max)

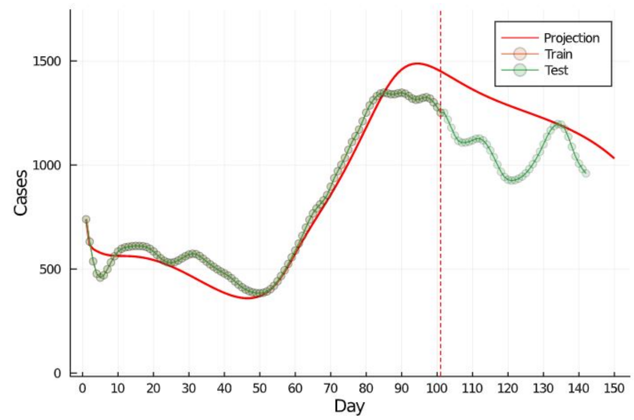


Figure 2: Neural ordinary differential equation (nODE) prediction for the active cases in the state of Ohio . The neuronal net is able to predict the next peak of cases and the average dynamics for almost forty days. The vertical line delimits the data set used to train the neuronal network and from the test set. The solid line is the solution of the neural ODE, which learned the dynamics of the outbreak. The variables used for this forecast are Cases, Hospital Admission, COVID-Like Symptoms, COVID-Like Symptoms in Community and COVID-Related Doctor Visits [?]

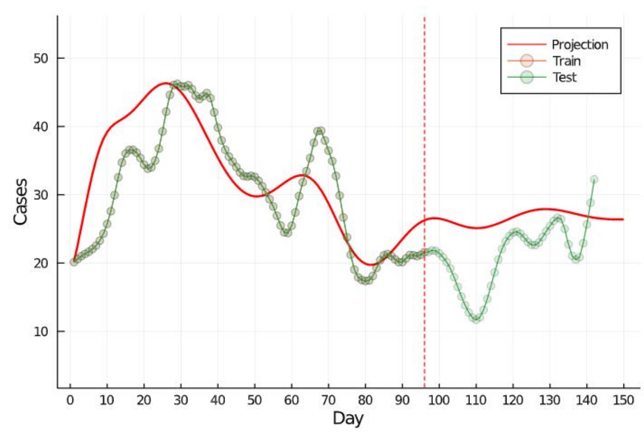


Figure 3: Prediction for the active cases in the state of ME (Maine).This is a particular case where the recorded active cases are few. The dynamic is not easily describable by an analytical model , while the neural ODE is able to learn the dynamics and predict an increase of cases for the following fifty days. This increase correlates nicely with the recorded cases not used for the learning process (test dots on the right of the vertical line).

ACKNOWLEDGMENTS

This work was supported by us. We also thank the Python and Julia developers for contributing to Data Science, Machine Learning and Artificial Intelligence.

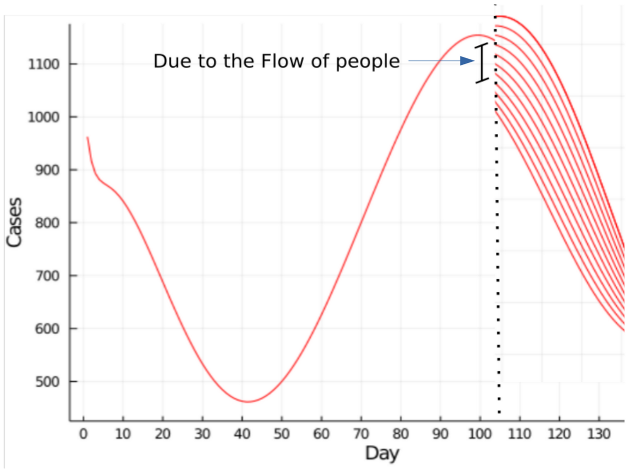


Figure 5: The trained neural ODE can extrapolate different possible scenarios depending on the variation on the number of active cases in the location . Changes in this number may be due in part due to the flow of people entering and leaving the region. This opens the possibility, for locations with strict close borders to be able to predict the effect of the flow of people on the curve of infected. (ACA mejorar la imagen).

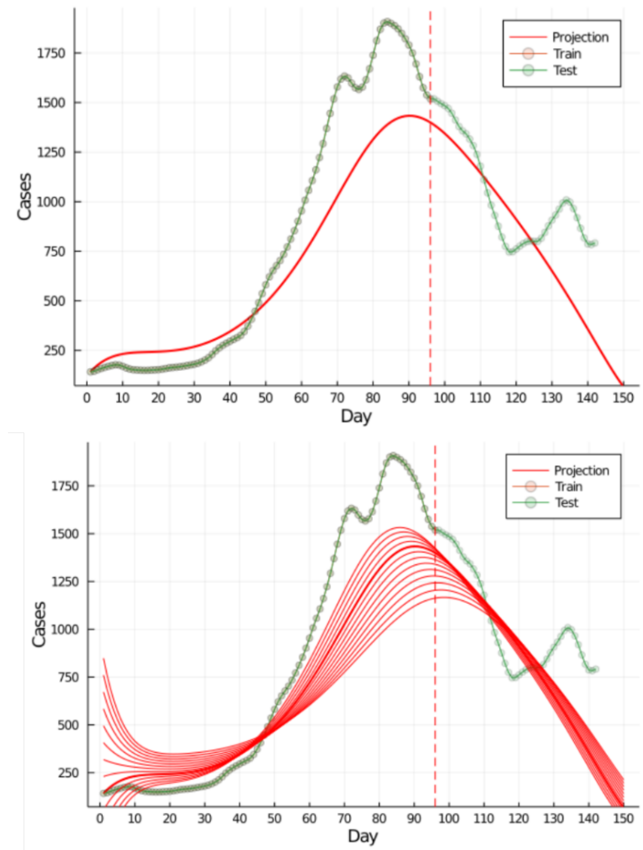


Figure 4: Once the dynamics is learnt by the neural ODE, its solution can predict the projection for different number of initial cases. This can be useful for estimating the error on the forecast, and also for inferring the effect of changes in the number of cases.

REFERENCES