Modeling COVID-19 spread in the USA using metapopulation SIR models coupled with graph convolutional neural networks

Petr Kisselev † Thomas Jefferson High School for Science & Technology, VA
Project Advisor: Padmanabhan Seshaiyer ‡ George Mason University, VA

Abstract.

Graph convolutional neural networks (GCNs) have shown tremendous promise in addressing data-intensive challenges in recent years. In particular, some attempts have been made to improve predictions of Susceptible-Infected-Recovered (SIR) models by incorporating human mobility between metapopulations and using graph approaches to estimate corresponding hyperparameters. Recently, researchers have found that a hybrid GCN-SIR approach outperformed existing methodologies when used on the data collected on a precinct level in Japan. In our work, we extend this approach to data collected from the continental US, adjusting for the differing mobility patterns and varying policy responses. We also develop the strategy for real-time continuous estimation of the reproduction number and study the accuracy of model predictions for the overall population as well as individual states. Strengths and limitations of the GCN-SIR approach are discussed as a potential candidate for modeling disease dynamics.

1. Introduction.

Compartment models are widely used in the modeling community to describe the spread of infectious diseases [1]. Standard SIR model considers three compartments: $S(t)$ - the number of susceptible individuals, $I(t)$ - the number of infected and $R(t)$-the number of recovered or deceased at time t. Representing the infection rate parameter by β and removal rate parameter by γ, the following system of equations is derived [2]:

$$\begin{cases} \dfrac{dS(t)}{dt} = -\beta \dfrac{S(t)I(t)}{N} \\ \dfrac{dI(t)}{dt} = \beta \dfrac{S(t)I(t)}{N} - \gamma I(t) \\ \dfrac{dR(t)}{dt} = \gamma I(t) \end{cases} \tag{1.1}$$

where $N = S(t) + I(t) + R(t)$ is the total population that is assumed to remain constant.

There are many modifications of the basic SIR model that have been proposed in the literature [3]. For example, the SEIR variation of the model includes another compartment for exposed individuals, while the SIRV variation incorporates a compartment vaccinated populations. It is also possible to account for individuals who end up succumbing to the disease and dying, as done in the SIRD variation and others like it. While these models are very intuitive and mathematically tractable, their predictive properties are highly dependent on the accuracy of the modeling parameters β, γ, and other parameters for the additional

†Department of Mathematical Sciences, GMU, email: peter.kisselev@gmail.com
‡Department of Mathematical Sciences, GMU, email: pseshaiy@gmu.edu

compartments. In fact, these basic parameters have been found to vary greatly between subpopulations for a variety of diseases including COVID-19 [4, 5, 6, 7]. This realization has motivated several groups to develop so-called "metapopulation SIR" or "SIR-network" models which tackle this problem by splitting the overall population into a number of subpopulations and allowing for variable infection and recovery rate parameters across these newly created "metapopulations"[2, 8].

Metapopulation ("network") SIR models for a total of M subpopulations typically have the form [2, 9, 10]:

$$\left\{\begin{array}{ll} \frac{dS_n}{dt} & = -S_n \sum\limits_{m=1}^{M} \beta_{mn} I_m \\ \frac{dI_n}{dt} & = S_n \sum\limits_{m=1}^{M} \beta_{mn} I_m - \gamma_n I_n \\ \frac{dR_n}{dt} & = \gamma_n I_n \end{array}\right. \tag{1.2}$$

where β_{mn} are the corresponding interaction parameters accounting for the movements between subpopulations. These parameters may account for the change in mobilities between different compartments and differences in vaccination policies in different regions, among other conditions. Extensive numerical validation of the network-type SIR models has been conducted in [2, 9] which demonstrated their efficiency in modeling certain disease outbreaks.

The caveat of this approach is the increase in computational complexity, the need to estimate a larger set of parameters, and work with higher dimensional data. Data-driven approaches promise to overcome the issues associated with mechanistic models [11]. In this work, we explore the benefits of coupling the metapopulation(network) SIR model with the graph convolutional neural network (GCN) methodology which has enjoyed significant advances and popularity in recent years. With the recent developments in computational power, neural networks have flourished, gaining significant popularity and enabling breakthroughs in many fields [12, 13].

One advantage of GCN parameter estimation compared to that of standard convolutional neural nets is its applicability to an arbitrary data structure as long as it may be represented by a graph. It is also better able to draw on geographical relationships. Several authors explored the GCN-SIR coupling, see the review provided in [3].

Drawing motivation from the work of Cao et al [8], we use GCNs to dynamically fit the parameters of the metapopulation SIR model using a given time series of data. Similar to [8], we focus on making predictions on the spread of COVID-19 with the GCN framework given different "horizons," and compare these forecasts with the standard SIR model. There are several distinctions in the approach presented in this work in comparison to the "mepoGNN" model of [8]. In particular: (1) the mepoGNN model was trained on Japanese precinct data, and our goal in this study is to model the spread of COVID-19 in the United States; (2) in applying the original model to US data, we found the need to change several modeling

assumptions including choosing a different form of the mobility parameter; (3) we analyzed the predictions of the model on specific subpopulations and estimated the overall reproduction number based on the metapopulation model.

More specifically, the goal of this work is to expand the scope of applicability of the model proposed in [8] by training the data on a larger dataset and modifying the mobility parameters to account for air travel between the states. In doing so, we address some of the limitations of the original approach. In addition, we mathematically derive the closed form representation of the reproduction number for the metapopulation ("network") SIR model with time-dependent parameters and show its consistency with results derived in the literature [2]. This enables us to perform state-level predictions and pave a way towards a multi-level approach to increase prediction accuracy. Finally, we develop a strategy to dynamically estimate the reproduction number at various spatial resolutions and test it against the data.

The paper is organized as follows. In section 1 we introduce the model and the research questions posed. In section 2 we have an overview of Graph Convolutional Neural Networks as a technique and its use in our specific application. Section 3 defines the model architecture more specifically, demonstrating the link between the metapopulation SIR and GCN aspects of the model. Next, in Section 4 we present the data and numerical hyperparameters used in the training and testing of the model followed by the derivation of the reproduction number in Section 5 and numerical results in Section 6. We summarize our findings and open research directions in Section 7.

2. Graph Convolutional Neural Networks.

Graph convolutional neural networks (GCNs), sometimes simply known as graph neural networks (GNNs), are an emerging technique that have shown promise in several areas [14, 3]. Fundamentally, GCNs represent an expansion of the traditional neural networks. The traditional neural network, a model inspired by functioning of a biological brain, is a computational model designed to perform tasks such as classification, regression, and pattern recognition [15]. Its structure consists of interconnected nodes, or "neurons," which are then organized into layers: an input layer, one or more hidden layers, and an output layer. Each neuron takes in information from the previous layer and outputs a weighted sum of the inputs that involves parameters including weights and biases that are constantly learned. This is followed by the application of an activation function to introduce non-linearity and enable the model to learn complex patterns. Activation functions, such as the sigmoid or ReLU (Rectified Linear Unit), play a critical role in determining the network's ability to capture intricate relationships within data. Inputs to the network represent features of the problem being modeled, while the output corresponds to predictions or classifications. Figure 2 is a schematic representation of a standard

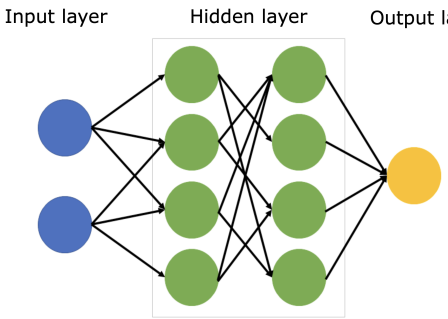

Figure 1. *Structure of a traditional neural network with circles representing individual neurons or inputs/outputs, and the arrows representing the passing of the value in the neuron to the next layer of neurons through a weight.*

neural network, illustrating the flow of information and transformations through its layers.

Graph neural networks replace the structure of the data on which a traditional neural networks performs transformations, instead, applying transformations to data which represents a graph, utilizing relationships within this structure to enhance predictions and available context [16].

More rigorously, graph neural network models may be defined as follows. Let G be defined as the graph data such that $G = (V, \mathcal{E})$. Where V is defined to represent set of nodes comprised of $|V| = M$ nodes. Similarly, we let $\mathcal{E}$ be such that $\mathcal{E} \subseteq V \times V$. Here it will be used to store connection data between the nodes. The features may also be represented by the matrix $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_M\}^T \in \mathbb{R}^{M\times D}$, where the feature vector $\mathbf{x}_i$ is associated with node v_i. Here, D is used to denote dimension of the feature. By convention, we define an adjacency matrix for G as $\mathbf{A} \subseteq R^{M\times M}$, where $\mathbf{A}_{ij} = 1$ for existing edges and $\mathbf{A}_{ij} = 0$ otherwise. Figure 2 is an illustration of embedding a graphical representation within the neural network framework that was described earlier. Specifically, the figure shows that the convolution is applied on a node-by-node basis, with the appropriate weights and biases.

In this paper we will be focusing on the subset of graph learning focused on node-level tasks. In other words, the GCN framework will be used to predict properties associated with individual nodes. As with any other neural network model, it will be necessary to train it using a subset of nodes with known properties, or the training set. This training set of data will be denoted as $\mathcal{V}_L$. The trained model will then be used to forecast the properties of unknown nodes from a separate testing set of data. The aforementioned training can be represented by the minimization of the following loss function:

$$\mathcal{L}(f_\theta(G)) = \sum_{v_i \in \mathcal{V}_L} \ell(f_\theta(\mathbf{X}, \mathbf{A})_i; y_i) \tag{2.1}$$

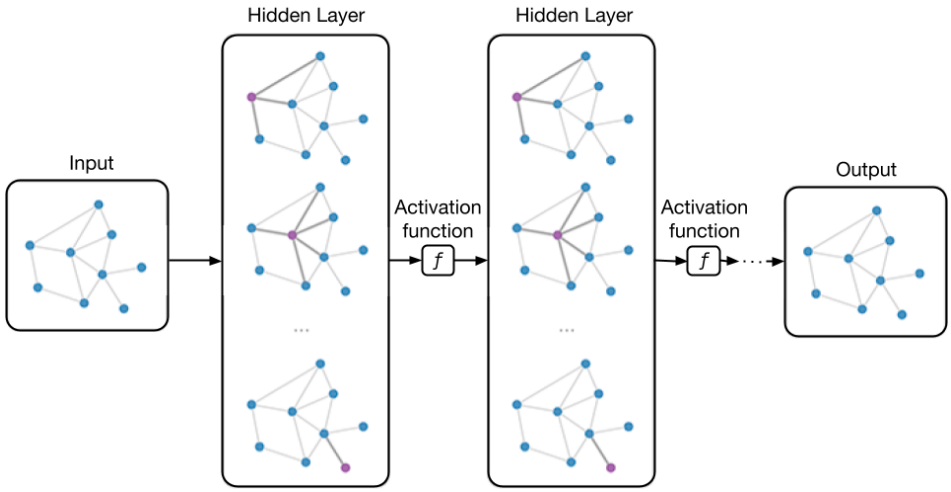

Figure 2. *Structure of a GCN: each hidden layer operates on every node individually, gaining influence from neighbors.*

Where θ is a vector containing the parameters of the model. The function $f_\theta(\mathbf{X}, \mathbf{A})$ is designated to forecast property values for each node, where y_i is defined to represent the true state of the node v_i. The difference between the predicted and true properties ($f_\theta(\cdot, \cdot)_i$ and y_i respectively) is quantified using a loss function $\ell(\cdot; \cdot)$. Examples of loss functions that can be used include RMSE (Root Mean Square Error), MAE(Mean Absolute Error), smooth L_1 loss, and others.

We will use this GNN structure to model disease dynamics of metapopulations through SIR models (1.1). The purpose of coupling the metapopulation SIR model to a convolutional neural network is to achieve better accuracy in estimating hyperparameters by taking into account communication/mobility of sub-populations between different regions. This is accomplished by the mobility parameters assigned to the edges of a graph, as described below. Section 3 explains how this hybrid approach integrates GCNs with the SIR mechanistic model which allows dynamic adjustment of the SIR model parameters. In contrast with the standard SIR paradigm where modeling parameters are kept fixed, real-time data analysis used in training the GCN informs the mechanistic model to improve its accuracy in real time.

3. GCN-SIR model description.

Consider system (1.2). Cao et al [8] chose the form $\beta_{mn} = \beta_n \left(\frac{h_{mn}}{N_m} + \frac{h_{nm}}{N_n}\right)$, where h_{mn} modeled mobility between regions m and n, and N_m, N_n represent the populations of the regions, respectively. We denote the total population of the country by $N = \sum_{m=1}^{M} N_m$. This leads to the following form of the metapopulation SIR model:

$$\begin{cases} \dfrac{dS_n}{dt} &= -S_n\beta_n \displaystyle\sum_{m=1}^{M} \alpha_{mn} I_m \\ \dfrac{dI_n}{dt} &= S_n\beta_n \displaystyle\sum_{m=1}^{M} \alpha_{mn} I_m - \gamma_n I_n \\ \dfrac{dR_n}{dt} &= \gamma_n I_n \end{cases} \tag{3.1}$$

where $\alpha_{mn} = \frac{h_{mn}}{N_m} + \frac{h_{nm}}{N_n}$ are the interaction coefficients modeling mobility. In their work, they argued that following form of mobility was best suited for this task: $h_{mn} = \alpha \frac{N_n N_m}{(dist_{mn})^d + \epsilon}$, where α, ϵ and d are the training hyperparameters and $dist_{mn}$ is the distance between the regions.

To couple the SIR model with GNN formalism described above, the SIR model was represented via a graph with the n-th node representing the n-th state with associated S_n, I_n, R_n data being part of the feature space, mobilities α_{mn} assigned to the edges and keeping γ_n, β_n to represent recovery/immunity and infection rates, respectively. In the context of the notation introduced in the previous section, $\mathbf{X}$ represents the input features of the dataset: currently susceptible population, the currently infected population, and the recovered population, and the mobility values – split across the metapopulations. More formally, for each $\mathbf{x}_i$ in $\mathbf{X}$, $\mathbf{x}_n = (S_n, I_n, R_n, H_n)$, where H_n stands for the list of h_{nm} values associated with the node n and all its neighboring states $m \neq n$. We define $\mathbf{A}$, the graph adjacency matrix, to be

$M \times M$ with $\mathbf{A}_{ij}$, for $0 \leq i,\, j < M$ and $\mathbf{A}_{ij} = 1$, as we utilize a complete graph with the mobilities acting as weights. The parameter vector, θ^{t+1}, includes the transmission rate β^{t+1} and recovery rate γ^{t+1} that are estimated by leveraging the outputs of the GCN trained on data available up to time t.

A graph convolutional neural network was trained using the somewhat complex architecture that has been claimed to be the first hybrid model that couples the metapopulation SIR model with spatiotemporal graph neural networks. The code shared by the authors [17] has been used as a basis for our investigation, where we implemented the GCN structure as shown in Figure 3. This architecture consists of three main sections: the graph learning module, the metapopulation SIR module, and the spatio-temporal module. For our model we chose to use the adaptive version of the model proposed by Cao et al [17], where the mobilities are initialized statically from an estimation based on population and distance. The spatio-temporal module is comprised of a combination of spatio-temporal (ST) layers. Each layer is created through combining a graph convolutional neural network layer with a gated temporal convolutional network layer. The results from the spatio temporal module are then passed through two fully connected layers with a ReLU (Rectified Linear) activation function and a Sigmoid activation function respectively, producing the predicted β and γ parameters [18]. These predicted parameters are fed into the final, metapopulation SIR module (see Figure 3) which is detailed above, where a simple time marching method is used to propagate the variables. The number of daily confirmed cases Y_n^{t+1} in the particular state n at a given time $t+1$ is predicted using the model (1.2) as follows:

$$Y_n^{t+1} = S_n^t \beta_n^{t+1} \sum_{m=1}^{M} \alpha_{mn} I_m^t \tag{3.2}$$

This process continues iteratively until sufficient accuracy is attained. Note that comparing to the formulation in [8], we did not drop the S_n^t term in our model and we trained the neural network accordingly.

In essence, the process consists of the GCN continually learning the dynamical patterns of of the data and adjusting the parameters of the metapopulation SIR model to minimize the cost function (2.1) and improve the accuracy of the prediction. The SIR itself does no learning, it operates entirely on the learning and output of the GCN.

4. Extending the model to US data.

In extending the original model to US data, we faced several challenges. There was a lack of an easily accessible source for recovery data. In order to gather the data necessary it was necessary to take several key steps. We started by sourcing data on daily infections from a dataset downstream from official data by the Johns Hopkins Center for Systems Science and Engineering (CSSE) [19]. This provided confirmed infection data on a county level which was binned up to the state level to alleviate computational complexity concerns. Additionally, the date standard format was reinterpreted as an integer day offset from the first day found in the dataset. We also made the decision to exclude US territories, the District of Columbia, Alaska, and Hawaii from the training and testing of our model as this would intro-

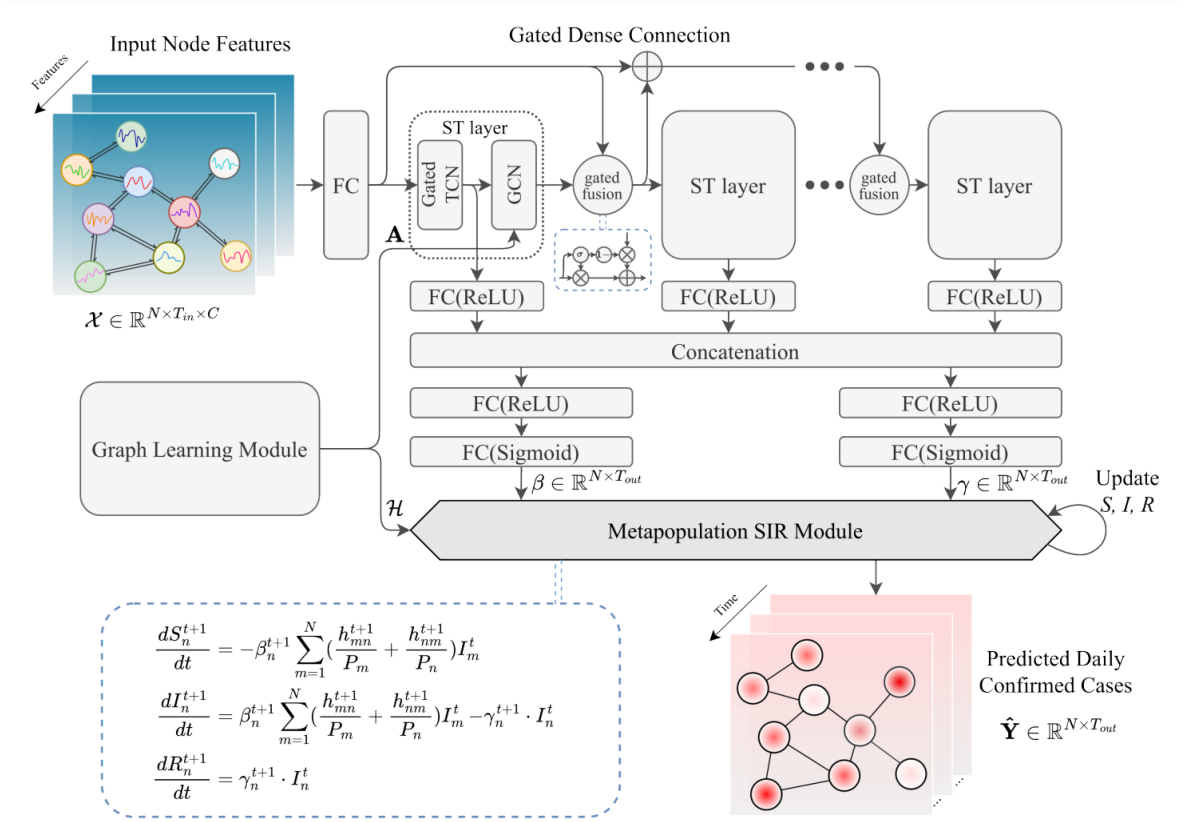

Figure 3. *Model architecture used in [8]. The predictions being made by the GCN passed through ReLU and Sigmoid Layers and then used as the parameters for the metapopulation SIR model which makes the final predictions. In our work, we use model* (3.1).

duce additional complexity into the geospatial relationships without benefiting the predictions made significantly. Several supplementary datasets were additionally used, such as data on the physical locations of state centers and US state populations [20, 21]. We were, however, unsuccessful in obtaining a dataset that could sufficiently detail recovery data in the United States. This was a challenge as the model necessitated such data as an input, data which was available in Japan but not for the US. To overcome this issue, we generated recovery parameter by numerically solving System 3.1 using an Euler approximation from an ad-hoc γ value and the known real-world infection data which we sourced. We believe that this solution is effective because while the recovery data is somewhat approximate, the model's performance in predicting infections is still compared against the ground truth.

The mobility value approximation has been improved with an additional term in the

formula to account for flight travel:

$$h_{mn} = \alpha \frac{N_n N_m}{(dist_{mn})^d + \epsilon} + \beta \max(N_n, N_m)(1 - \delta_{mn}) \tag{4.1}$$

Namely, the last term allows to have significant mobility between densely populated states even if the distance between them is large. Multiplying by the Kronecker delta function , $(1 - \delta_{mn})$, makes sure this term collapses to zero in the simple SIR model case when $M = 1$.

The following consistency analysis has been carried out. As a result of the model architecture, the mobility values are not normalized and so it is necessary to balance them out using the following:

Lemma 4.1. *The Metapopulation model* (3.1) *is consistent with the standard SIR model if and only if* $2\alpha N^2 = \epsilon$.

Proof. Taking the limiting case of $M = 1$ subpopulation and denoting $h_{mn} = h_{nm} = h$, $\beta_n = \beta, \forall n = 1, \ldots, M$, we obtain $\alpha_{nn} = \frac{2h}{N}$, where $N = N_n = N_m$ for all n, m. It is clear that since $h_{mn} = \alpha \frac{N_n N_m}{(dist_{mn})^d + \epsilon}$, $h = \alpha \frac{N^2}{\epsilon}$, so $\alpha_{nn} = \frac{2\alpha N}{\epsilon}$. Hence $S_n \beta_n \sum_{m=1}^{M} \alpha_{mn} I_m = \beta \frac{2\alpha N}{\epsilon} S_n I_m$ which is equal to $\beta SI/N$ under the condition that $2\alpha N^2 = \epsilon$ ∎

This allows to reduce the number of free parameters to α and d, simplifying the training of the mobility parameters. Parameter ϵ was fixed in accordance with Lemma 4.1. The optimized GCN hyperparameter values and training details are provided below in Table 1:

Symbol	Description	Value
α	Mobility Scaling Factor	1.12×10^{-6}
d	Distance Decay Factor	1.73
β	Flight Travel Factor	5.98×10^{-7}
λ	Learning Rate	2.5×10^{-5}
$\mathcal{L}$	Loss Function	MAE
	Optimizer	Adam
	Epoch Count	319

Table 1

Table of parameters used in the model

It must be noted that some run-to-run variance is expected in this model due to the random nature of weight initialization in the GCN training. Additionally, hardware differences may also slightly change results as using the GPU, CPU, or a dedicated accelerator will lead to having minor differences in the driver and PyTorch backend implementations. In our case, we performed our numerical analysis on a system equipped with an Intel Core i5-13600k CPU and a NVIDIA RTX 3070 GPU for training acceleration.

5. Real-time tracking of the reproduction number.

One of the critical considerations that is important to keep in mind when modeling COVID-19, as well as other infectious diseases, is the ability of the model to predict its spread.

The threshold parameter $\mathcal{R}_0$, such that the disease free equilibrium (DFE) is asymptotically stable for $\mathcal{R}_0 < 1$ and unstable otherwise, is called the basic reproduction number. A more granular parameter accounting for the changes in population susceptibility, is the so-called effective reproduction number, denoted as $\mathcal{R}_t = \mathcal{R}_0 \frac{S_t}{N}$. Both measures are important tools for the mathematical validation of epidemiological models, as well as for practical considerations. Challenges and misconceptions in estimating these metrics are well documented [22, 23].

Local parameters of the evolving epidemic change based on mobility patterns, population density and policy measures, the complexity of which creates significant difficulties for decision-making. The need for accurate continuous real-time prediction of the reproduction numbers in light of this variability has long been recognized and documented in the literature [24]. Some of the proposed real-time estimation methods include the adaptive SIR methodolgy (ASIR, [24]), where $\mathcal{R}_0$ is based on a sliding time window approach, and the introduction of an "effective contact rate" to capture incidence dynamics over a given network [25]. We argue that the graph neural network approach chosen in this work has a natural capability to capture the evolution of modeling parameters in real-time, and hence it may provide an opportunity to improve upon prior $\mathcal{R}_0$ predictions.

As noted in [26], there is a natural connection between $\mathcal{R}_0$ and $\mathcal{R}_t$ when it comes to studying SIR population models. Namely, as we look at the equation for the infected population in the standard SIR model,

$$\frac{dI}{dt} = \beta S \frac{I}{N} - \gamma I = \gamma I(\frac{\beta}{\gamma}\frac{S}{N} - 1) = \gamma(\mathcal{R}_0 \frac{S}{N} - 1)I = \gamma(\mathcal{R}_t - 1)I.$$

The role of the bifurcation parameter $\mathcal{R}_t$ is clear. It separates stable behavior of the disease-free equilibrium $I^* = 0$, for which $\frac{dI}{dt} < 1$ (for $\mathcal{R}_t < 1$) from the unstable and possibly endemic equilibrium when $\mathcal{R}_t > 1$.

For the metapopulation SIR model considered in this paper, we can take a similar approach, following the framework described in [27]. Namely,

Theorem 5.1. *Basic reproduction number for model (3.1) is given by $\mathcal{R}_0 = \rho(DA)$, where $A = \{A_{nm}\} = \{\alpha_{nm}\}$ is the mobility matrix and $D = diag(\frac{\beta_1}{\gamma_1}, \ldots, \frac{\beta_m}{\gamma_m})$ is the scaling matrix.*

Proof. It is easy to see that the Jacobian of this model, linearized around the Disease Free Equilbrium (DFE) x^*, can be represented as $DF(x^*) = F - V$, where $F_{nm} = \beta_n P_n \alpha_{nm}$ and $V = \mathrm{diag}(\gamma_1, \ldots, \gamma_m)$. Here, "$F - V$" refers to the *next generation matrix*, where "F" represents the inflow and "V" represents the outflow, and the basic reproduction number is calculated as the maximum eigenvalue of the matrix "FV^{-1}". We used the fact that the DFE is represented by $I_n^* = 0, S_n^* = N_n$ for this model. As shown in [27], the DFE is stable when $\rho(FV^{-1}) < 1$ under certain conditions on F and V that can be shown to hold in this case. Henceforth we arrive at the conclusion that for this model:

$$\begin{aligned} &\mathcal{R}_0 = \rho(DA), \text{ where} \\ &A = \{A_{nm}\} = \{\alpha_{nm}\} \text{ is the mobility matrix,} \\ &D = \mathrm{diag}(\tfrac{N_1\beta_1}{\gamma_1}, \ldots, \tfrac{N_m\beta_m}{\gamma_m}) \text{ is the scaling matrix} \end{aligned} \tag{5.1}$$

which proves the result of this theorem. ∎

From Lemma 4.1, we know that in the limiting case of $M = 1$ we have to satisfy

$$\alpha_{nn} = \frac{2\alpha N}{\epsilon} = \frac{1}{N},$$

so since $D = N\dfrac{\beta}{\gamma}$ the reproduction number of the metapopulation model $\mathcal{R}_0$ in this case converts to the well known result $\mathcal{R}_0 = \dfrac{\beta}{\gamma}$.

This result provides a method for continuous evaluation of $\mathcal{R}_0$ based on the evolving set of infection parameters estimated by the neural network. As the neural net learns and adjusts the underlying mobility and recovery rates, we can use this estimation to more accurately predict the basic reproduction number. While strictly speaking the $\mathcal{R}_0$ concerns prediction at the DFE, the idea of this calculation is to adaptively predict the steady state behavior based on the adjusted model parameters.

6. Numerical results.

Results of our numerical experiments produced by applying the modified mepoGNN model to US data are presented next. In particular, US state center-center distances and state population data was obtained from Kaggle [21]. Confirmed US COVID-19 cases were collected from the Github repository maintained by The Center for Systems Science and Engineering at Johns Hopkins University [28]. This dataset includes COVID-19 infection data from 3/1/2020 to 4/12/2020, a period of 278 days. All simulations were run on the same hardware as mentioned previously: a an Intel Core i5-13600k CPU and a NVIDIA RTX 3070 GPU.[1] In this analysis, the same configuration was used as in the original Cao et al paper: a 6:1:1, training:validation:testing split, where the training set is denoted on all figures by a dashed vertical line. The reason for including the training set data in the visualizations is that otherwise the timescale would be too small to see trends. Furthermore, this allows us to see the long-term behavior of the model in a way that is otherwise unclear.

Figure 4 shows predictions from the trained model based on a 1-day (left) and 7-day (right) horizon, respectively. The GCN-SIR model is juxtaposed with the standard SIR model trained on the same dataset. It can be seen that by taking into account variability between regions, the model improves upon the prediction provided by traditional SIR approaches. In addition, it takes advantage of the neural network's learning capabilities to effectively train model parameters. In Figure 5 we look at the accuracy of the model predictions per state, choosing Virginia, New York, California, Ohio, Rhode Island and North Dakota as a sample containing large and small subpopulations. What we see is a strong model prediction for the densely populated states (New York, Virginia, California, Ohio) and a poor prediction for the less populous states (North Dakota, Rhode Island). To quantify the performance of the model, we use the R^2 coefficient of determination as a measure of fit [29]. More specifically, it is computed with the following formula, where $\bar{y}$ is the mean of the true values, $\hat{y}$ is the

[1]The Python code is available at https://github.com/Peter-Kisselev/GCN-SIR.

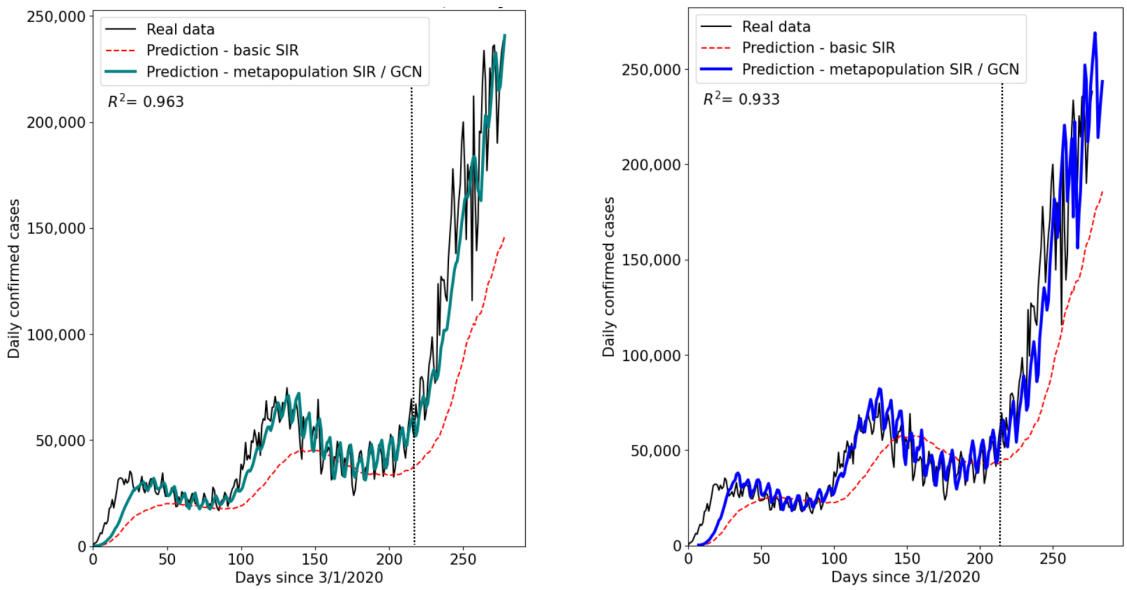

Figure 4. *Metapopulation model prediction for US data based on real COVID-19 data, compared against the standard SIR model. Left panel: SIR-GCN Predictions on US COVID-19 data, 1 day horizon; Right Panel: SIR-GCN Predictions on US COVID-19 data, 7 day horizon. Training and testing datasets are separated by the vertical dashed line.*

predicted value, and y is the true value:

$$R^2 = 1 - \frac{\sum_{i=0}^{n} (\hat{y}_i - \bar{y}_i)}{\sum_{i=0}^{n} (y_i - \bar{y}_i)} \tag{6.1}$$

To test this hypothesis, we plotted the correlation between the R^2 measure of fit and the corresponding state size in Figure 6. To account for the large variation in state populations, the populations are log-scaled. The graph clearly shows moderate correlation, confirming that large-size subpopulations enjoy a more accurate prediction by the GCN-SIR model, which is to be expected given that the size plays a critical part in the optimization algorithm used in training the GCN. It is also clear that a majority of the state-level predictions have an $R^2 > 0.6$, which indicates reasonable performance overall.

Next, we performed numerical experiments to continuously estimate the basic reproduction number ($\mathcal{R}_0$) of the entire metapopulation model using the estimate derived earlier in (5.1). The numerical results of this estimation, compared to the standard SIR reproduction number calculation, are given in Figure 7. We can see the evolution of the $\mathcal{R}_0$ value over the course of the pandemic, roughly capturing the ups and downs of the infection represented in Figure 4. The higher frequency oscillations visible in the graphs are due to the day-by-day variations in the neural net predictions and the real-world fact that people tend to travel more on certain days of the week than others. It must be noted that while the overall $\mathcal{R}_0$ values obtained by

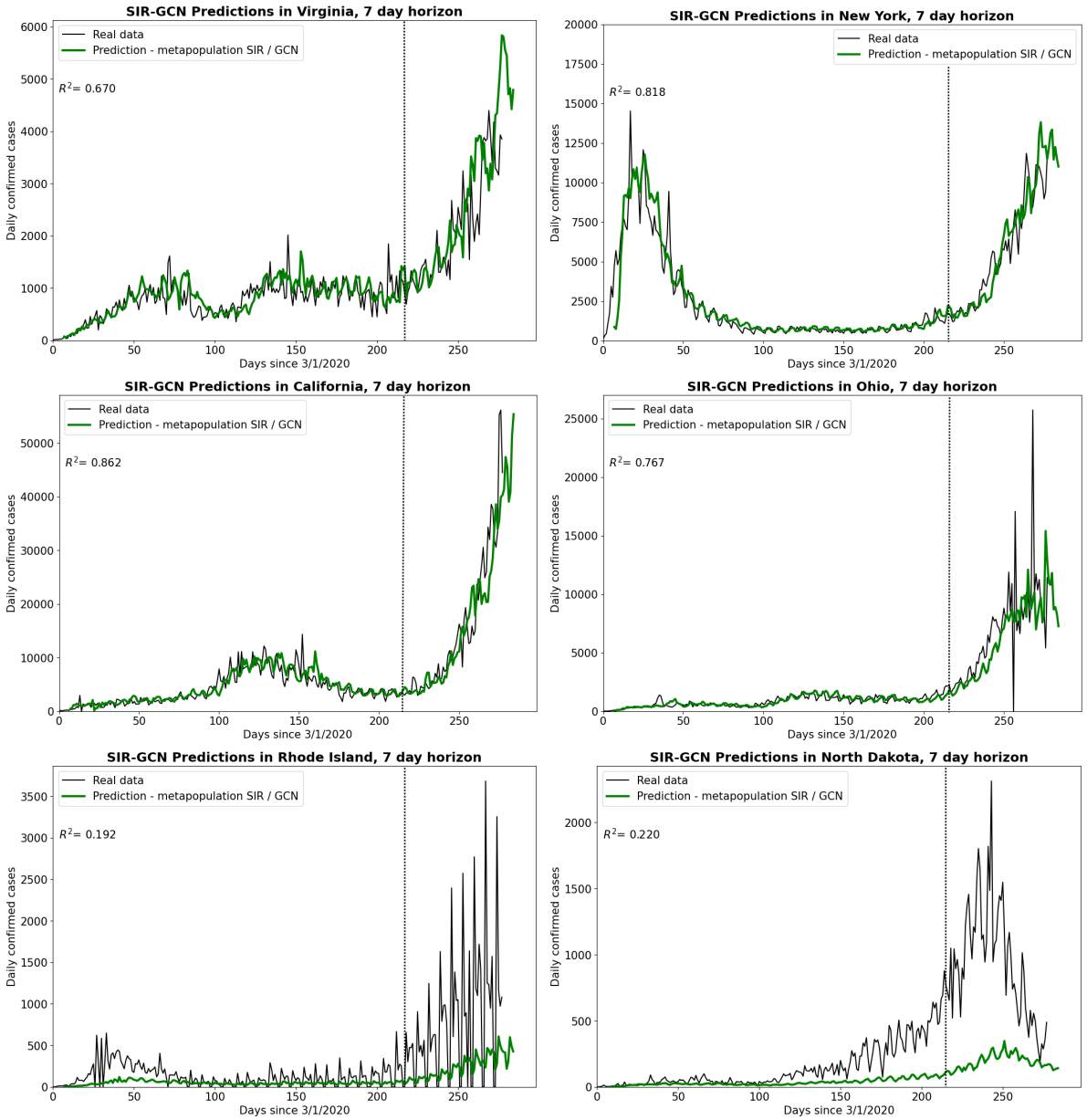

Figure 5. *Metapopulation model predictions for six US states. Training and testing datasets are separated by the vertical dashed line.*

this approach were comparable to those available in the literature, state-level predictions were far less accurate. It indicates that a more granular county-based approach might be necessary to resolve state-level estimations. As noted in Section 5, this adaptive $\mathcal{R}_0$ calculation is aimed at predicting the system behavior at the steady state and does not coincide with the effective reproduction number ($\mathcal{R}_t$) estimation which requires a more careful analysis of the S^t values.

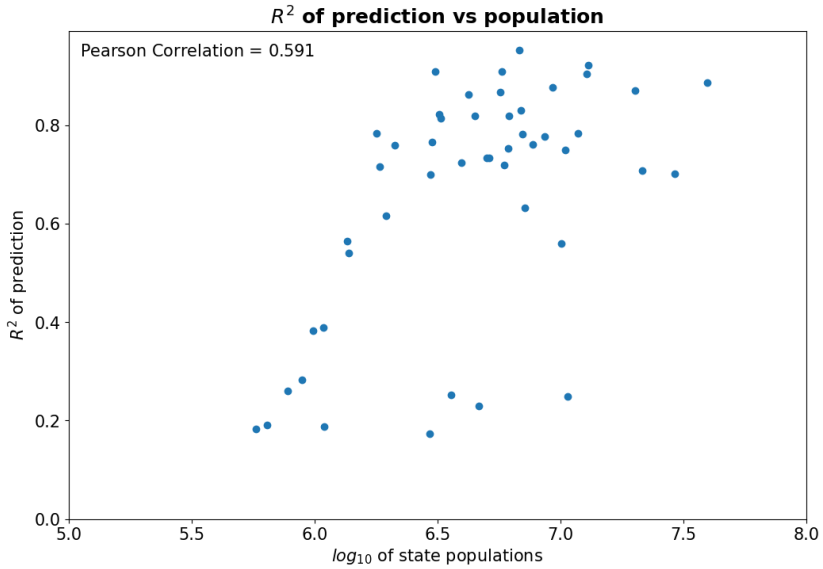

Figure 6. *Correlation between the accuracy of fit for the metatpopulation SIR model and the size of the state for 48 contiguous Unites states.*

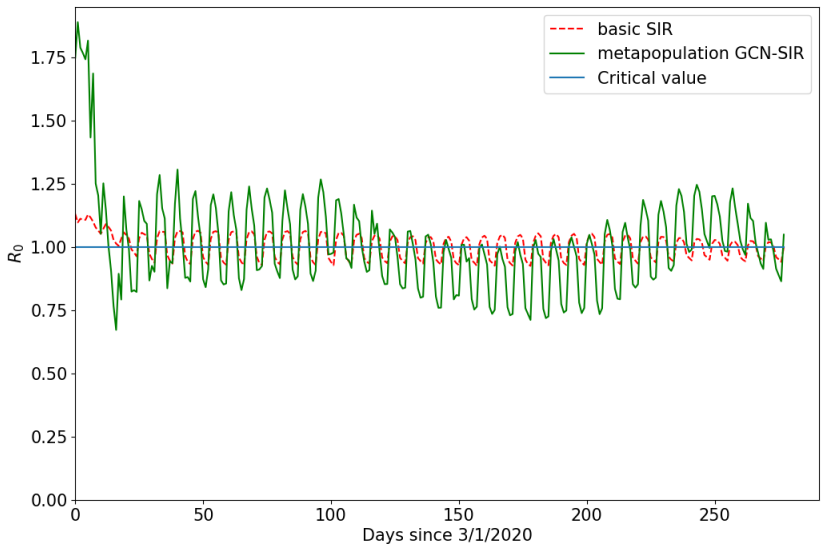

Figure 7. $\mathcal{R}_0$ *number estimation using the GCN-SIR model.*

7. Discussion and future work.

In this work we successfully adapted the hybrid GCN-SIR metapopulation model to predict the evolution of COVID-19 in the 48 continental states of the United States of America. In order to do so, we changed the formulations of the mobility parameters and derived the reproduction number formulation compatible with the standard SIR model. This allowed to streamline the process for training the hyperparameters to obtain a more robust implementation.

Upon implementing these changes, we were able to obtain a high accuracy predictions for both 7-day and 1-day horizons for the entire United States. We noticed that individual state prediction accuracy was correlated with the state population, with densely populated states enjoying a better fit.

Based on the neural network approach to learn the infection rates in real time, we developed an alternative to the adaptive SIR method for estimating the reproduction numbers. Applying this approach to the entire US population, a reasonable prediction has been obtained, giving reason to believe that further improvements may yield an even better predictive capability that would be of significant interest to policy makers and medical practitioners.

Overall, based on the results presented in this work, GCN-SIR metapopulation model seems to have a high potential for predicting improving predictions of the spread of infectious diseases based on sufficient amount of training data. To our knowledge, this is the first application of this type of a GCN-SIR coupling to real COVID-19 data collected within the USA. While these preliminary results are encouraging, we believe that additional work needs to be performed to validate the model on other types of data. High correlation of the R^2 fitting parameter with the size of the subpopulations indicates that further improvements may be made to the choice of the mobility formulation, including learning mobility matrices in real time. Additional work could include building a better mobility estimation based on a more granular county-level data. All of our current attempts at a more granular model so far have run into issues with handling the sheer size of the model.

Future work will also include deriving more accurate state-level basic and effective reproduction number estimations and improvement of parameter estimation procedures. The possibility of additionally including the effect of local policy changes into the model is also one that we will consider in the future.

Acknowledgements

This work is partially supported by the National Science Foundation grant DMS-2230117.

REFERENCES

[1] Fred Brauer and Carlos Castillo-Chavez. Mathematical models in population biology and Epidemiology. Springer, 2012.

[2] L. M. Stolerman, D. Coombs, and S. Boatto. SIR-network model and its application to dengue fever. SIAM Journal on Applied Mathematics, 75(6):2581–2609, 2015.

[3] Zewen Liu, Guancheng Wan, B. Aditya Prakash, Max S.Y. Lau, and Wei Jin. A review of graph neural networks in epidemic modeling. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD '24, page 6577–6587, New York, NY, USA, 2024. Association for Computing Machinery.

[4] Andrea L. Bertozzi, Elisa Franco, George Mohler, Martin B. Short, and Daniel Sledge. The challenges of modeling and forecasting the spread of covid-19. Proceedings of the National Academy of Sciences, 117(29):16732–16738, 2020.

[5] Mazair Raissi, Niloofar Ramezani, and Seshaiyer Padmanabhan. On parameter estimation approaches for predicting disease transmission through optimization, deep learning and statistical inference methods. Letters in Biomathematics, 6, 12 2019.

[6] Comfort Ohajunwa, Kirthi Kumar, and Seshaiyer Padmanabhan. Mathematical modeling, analysis, and simulation of the COVID-19 pandemic with explicit and implicit behavioral changes. Computational and Mathematical Biophysics, 8:216–232, 12 2020.

[7] Sagi Shaier and Maziar Raissi. Disease informed neural networks. CoRR, abs/2110.05445, 2021.

[8] Qi Cao, Renhe Jiang, Chuang Yang, Zipei Fan, Xuan Song, and Ryosuke Shibasaki. MepoGNN: Metapopulation epidemic forecasting with graph neural networks. In Massih-Reza Amini, Stéphane Canu, Asja Fischer, Tias Guns, Petra Kralj Novak, and Grigorios Tsoumakas, editors, Machine Learning and Knowledge Discovery in Databases, pages 453–468, Cham, 2023. Springer Nature Switzerland.

[9] Daniel T. Citron, Carlos A. Guerra, Andrew J. Dolgert, Sean L. Wu, John M. Henry, Héctor M. Sánchez C., and David L. Smith. Comparing metapopulation dynamics of infectious diseases under different models of human movement. Proceedings of the National Academy of Sciences, 118(18):e2007488118, 2021.

[10] J. Arino and P. van den Driessche. Disease spread in metapopulations. Nonlinear Dynamics and Evolution Equations, Fields Inst. Commun, 48:1–13, 2006.

[11] Connor et al. Shorten. Deep learning applications for COVID-19. Journal of big data, 8,1, 2021.

[12] Ibomoiye Domor Mienye, Theo G. Swart, George Obaido, Matt Jordan, and Philip Ilono. Deep convolutional neural networks in medical image analysis: A review. Information, 16(3), 2025.

[13] Wang L. Zhang Y. et al. Zhao, X. A review of convolutional neural networks in computer vision. Artif. Intell. Rev., 57, 99, 2024.

[14] Pan S. Chen F. Long G. Zhang C. Wu, Z. and P. S. Yu. A comprehensive survey on graph neural networks. IEEE Transactions on Neural Networks, 32(1):4–24, 2021.

[15] F Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review, 65 (6):386–408, 1958.

[16] Lingfei Wu, Peng Cui, Jian Pei, and Liang Zhao. Graph Neural Networks: Foundations, Frontiers, and Applications. Springer Singapore, Singapore, 2022.

[17] Qi Cao, Renhe Jiang, Chuang Yang, Zipei Fan, Xuan Song, and Ryosuke Shibasaki. MepoGNN: Metapopulation epidemic forecasting with graph neural networks. GitHub repository, 2022.

[18] Alonso Ogueda-Oliva and Seshaiyer Padmanabhan. Literate programming for motivating and teaching neural network-based approaches to solve differential equations. International Journal of Mathematical Education in Science and Technology, 55:1–34, 09 2023.

[19] Johns Hopkins University Center for Systems Science and Engineering (JHU CSSE). COVID-19 data repository by the center for systems science and engineering (csse) at johns hopkins university. GitHub repository, 2020. Archived Mar 10, 2023, https://github.com/CSSEGISandData/COVID-19.

[20] Alexandre Petit. US population by state. Kaggle dataset, 2024. https://www.kaggle.com/datasets/alexandrepetit881234/us-population-by-state.

[21] Tennerimaheshwar. US state and territory latitude and longitude data. Kaggle dataset, 2024. https://www.kaggle.com/datasets/tennerimaheshwar/us-state-and-territory-latitude-and-longitude-data.

[22] Gostic KM and et al. Practical considerations for measuring the effective reproductive number. PLoS Comput Biol., 16(12), 2020.

[23] Jing Li, Daniel Blakeley, and Robert J. Smith. The failure of r0. Computational and Mathematical Methods in Medicine, 2011(1):527610, 2011.

[24] Shapiro MB, Muscioni G. Karim F, and Augustine AS. Adaptive susceptible-infectious-removed model for continuous estimation of the COVID-19 infection rate and reproduction number in the United States: Modeling study. J. Med Internet Res., 2021.

[25] Razvan G. Romanescu, Songdi Hu, Douglas Nanton, Mahmoud Torabi, Olivier Tremblay-Savard, and Md Ashiqul Haque. The effective reproductive number: Modeling and prediction with application to the multi-wave COVID-19 pandemic. Epidemics, 44:100708, 2023.

[26] A. Cintrón-Arias, A. Castillo-Chávez, L. Bettencourt, A. Lloyd, and H.T. Banks. The estimation of the effective reproductive number from disease outbreak data. Mathematical Biosciences and Engineering, 6:261–282, 2009.

[27] P. van den Driessche and James Watmough. Reproduction numbers and sub-threshold endemic equilibria for compartmental models of disease transmission. Mathematical Biosciences, 180(1):29–48, 2002.

[28] Ensheng et al. Dong. An interactive web-based dashboard to track COVID-19 in real time. The Lancet Infectious Diseases, 20:533 – 534.

[29] Norman R. Draper and Harry Smith. Applied Regression Analysis. John Wiley & Sons, 3rd edition, 1998.