

PHYSICS-INFORMED NEURAL NETWORKS FOR INFORMED VACCINE DISTRIBUTION IN META-POPULATIONS

Alvan Caleb Arulandu¹ & Padmanabhan Seshaiyer^{2,*}

¹Thomas Jefferson High School for Science and Technology, USA

²Department of Mathematical Sciences, George Mason University, USA

*Address all correspondence to: Padmanabhan Seshaiyer, Department of Mathematical Sciences, George Mason University, USA, E-mail: pseshaiy@gmu.edu

Original Manuscript Submitted: 01/22/2023; Final Draft Received: N/A

Accurate numerical and physical models play an important role in modeling the spread of infectious disease as well as informing policy decisions. Vaccination programs rely on the estimation of disease parameters from limited, error-prone reported data. Using physics-informed neural networks (PINNs) as universal function approximators of the SIR compartmentalized differential equation model, we create a data-driven framework that uses reported data to estimate disease spread and approximate corresponding disease parameters. We apply this to data from a London boarding school, demonstrating the framework's ability to produce accurate disease and parameter estimations despite noisy data. However, real-world populations contain sub-populations, each exhibiting different levels of risk and activity. Thus, we expand our framework to model meta-populations of preferentially-mixed subgroups with various contact rates, introducing a new substitution to decrease the number of parameters. Optimal parameters are estimated through PINNs which are then used in a negative gradient approach to calculate an optimal vaccine distribution plan for informed policy decisions. We also manipulate a new hyperparameter in the loss function of the PINNs network to expedite training. Together, our work creates a data-driven tool for future infectious disease vaccination efforts in heterogeneously mixed populations.

KEY WORDS: Compartmental Models, Neural Networks, Heterogeneity

1. INTRODUCTION

In light of the millions of confirmed COVID-19 cases across the globe, understanding the transmission dynamics of infectious diseases through modeling, analysis, and simulation is essential for not only predicting disease spread but also informing effective policy interventions to combat it. After the outbreak of severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) was first identified, governments were forced to interpret the evolving pandemic landscape to create effective policies, relying on data visualization tools and statistical models to understand disease

spread both locally and globally.

To address this demand for accurate intervention simulation, [Kerr et al. \(2021b\)](#) at the Institute for Disease Modeling published *Covasim*, an open-source, accessible agent-based model (ABM) that simulates the effect of policy interventions on disease spread. Using complex contact networks to model the varied interactions individuals have throughout the day, ABMs evolve the disease state of agents over discrete time steps. By implementing rule-based interventions such as vaccination, social distancing, and contact tracing ([Kucharski et al. \(2020\)](#)), some ABMs, including *Covasim*, have been deployed by public health officials to inform policy decisions in over a dozen countries ([Kerr et al. \(2021a\)](#); [Panovska-Griffiths et al. \(2020\)](#); [Quach et al. \(2021\)](#)).

However, the computational cost of ABMs, at best, scales linearly with population size. While new techniques such as dynamic rescaling are able to maintain a constant level of precision and computation time throughout the simulation, ABMs are challenged by a constant tradeoff between computational cost and the inaccurate discretization of the population.

Compartmental models replace this discretization with mathematical theory, describing the progression of a population through disease states using differential equations. The Susceptible-Infected-Recovered (SIR) model by [Kermack and McKendrick \(1927\)](#) is one of the earliest such mathematical formulations, illustrated in Figure 1.

FIG. 1: SIR Compartmental Model Diagram

The simple SIR model assumes that the system is closed, ignoring vital dynamics such as birth and death rates. The target population, N , is then subdivided into three compartments by disease state; $S(t)$, $I(t)$, and $R(t)$ correspond to the number of susceptible, infected, and recovered individuals respectively at a given time. The flux of individuals between compartments can be described using the Law of Mass Action and two disease parameters: the transmission rate, β , and the recovery rate, α . Together with initial conditions $S(0)$, $I(0)$, and $R(0)$, the model's dynamics are fixed, as $N = S(0) + I(0) + R(0)$. The SIR model not only provides theoretical epidemiological insights but also provides a way for generating realistic outbreaks using a numerical integration technique such as Runge-Kutta 4/5 ([Butcher \(2016\)](#)).

Compared to ABMs, the versatility of compartmental models is limited, as developing a mathematical system that encapsulates the complexity of human social networks is challenging. However, recent advancements have allowed compartmental models to simulate many behaviors. For example, [Yang et al. \(2021\)](#) evaluated the effect of lockdown policy in Spain with a model that introduced compartments for exposed, asymptomatic, pre-symptomatic, mild COVID, and severe COVID. [Bouchnita et al. \(2021\)](#) used an SEIR model to predict that social distancing measures would be needed to shorten COVID-19 waves in Vietnam.

Similarly, this work uses a SIR-based model to improve vaccination programs, which help populations achieve a level of immunity that prevents infectious people from causing further outbreaks. While ABMs are able to evaluate the effects of various vaccination strategies ([Sulis and Terna \(2021\)](#)), they are unable to inform optimal policy decisions. In order to compute an optimal intervention strategy, simulations must be repetitively run, compounding the computational cost of a single simulation. Instead, our work exploits the simplicity of compartmental models to create a framework for efficient, accurate parameter estimation to drive further optimization using gradient descent.

A variety of approaches have been developed to estimate disease parameters from real-world data including non-parametric estimation [Smirnova et al. \(2019\)](#), particle swarm optimization [Akman et al. \(2018\)](#), Bayesian techniques [Akman et al. \(2016\)](#), and inverse methods. Statistical approaches such as maximum-likelihood and Poisson regression methods have also been applied [Capaldi et al. \(2012\)](#). Some of this work has shown that estimation precision increases with the amount of outbreak data used, and sensitivity analysis techniques such as Latin Hypercube Sampling have been able to rank the relative importance of model parameters.

We take the approach of simultaneously estimating disease parameters while approximating a compartmental system, and several works have also attempted to learn differential equations from data. [Ling et al. \(2016\)](#) modeled the Reynolds stress anisotropy tensor using deep neural networks, and [E et al. \(2017\)](#) solved parabolic partial differential equations (PDEs) using reinforcement learning. However, recent advancements have pioneered the use of physics-informed models to estimate solutions. In fact, physics-informed neural networks (PINNs) were

first benchmarked with the SIR model in [Raissi et al. \(2019b\)](#).

Our framework uses PINNs in conjunction with a meta-population model to estimate disease parameters which are used to construct an optimal vaccine distribution plan. The paper is sectioned as follows. In Section 2, we provide necessary background information and present the structure of our approach. Section 3 analyzes our approach through a series of computational experiments, and Section 4 addresses the limitations of our framework while outlining future research avenues.

2. BACKGROUND AND METHODS

In this section, we establish the motivation for using meta-population models and PINNs for parameter estimation. We also present an optimization technique for optimally distributing vaccines from these parameters.

2.1 Neural Networks

A deep neural network consists of multiple connected neurons, each capable of producing a real-valued activation. In a feed-forward neural network such as Figure 2, neurons are organized in sets of L connected layers. Densely-connected layers, in particular, connect each pair of neurons in neighboring layers, $x_j^{(l-1)}$ and $x_i^{(l)}$, with a weight, $w_{ij}^{(l)}$. A neuron's activation is then computed as a weighted sum of the previous layer's activations plus a bias term, b , and passed into an activation function $\sigma(t)$. In general, the network is evaluated as follows

$$\vec{x}^{(1)} = \vec{x} \text{ and } x_i^{(l)} = \sigma \left(b_i^{(l)} + \sum_{j=1}^n w_{ij}^{(l)} \right) \text{ for all } 2 \leq l \leq L$$

where $\vec{x}$ is the provided input and $\vec{x}^{(L)} = \vec{y}$ is the computed output. Treating the hidden layers as a black box, a neural network can be described as the vector-valued function $f_{net}(\vec{x}) = \vec{y}$, and by the Universal Approximation Theorem, f_{net} can approximate any continuous function $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ provided that $\sigma(t)$ is non-polynomial and the network has arbitrary width [Hornik \(1991\)](#).

FIG. 2: A Densely-Connected, Feed-Forward Neural Network Architecture

During supervised learning, the network trains on a labeled dataset of pairs $(\vec{x}, \vec{y})$, using the back-propagation algorithm to update the network's weights and biases to minimize a loss function which captures the error between the predicted $\vec{y}_{\text{pred}}$ and the true $\vec{y}$ for each sample. In the case of mean squared error (MSE),

$$\mathcal{L}(\vec{\theta}) = \frac{1}{m} \sum_i (\vec{y}_i - \vec{y}_{\text{pred}_i})^2$$

As $\mathcal{L} \rightarrow 0$ over training, $f_{\text{net}}(\vec{x}) \rightarrow f(\vec{x})$, providing a method to estimate any vector-valued system.

2.2 Physics-Informed Neural Networks

In 2017, [Raissi et al. \(2019a\)](#) introduced PINNs, a data-driven deep learning approach for approximating ordinary differential equation (ODE) and PDE systems. PINNs employ deep neural networks as universal function approximators to approximate nonlinear systems, creating a new family of data-efficient spatiotemporal function approximators. Since 2017, PINNs have been used to tackle problems in quantum mechanics, wave propagation, and fluid mechanics [Cai et al. \(2022\)](#), but by modeling infectious disease spread as an ODE system, such as the SIR model, PINNs can also be applied in the context of epidemiology.

FIG. 3: A PINNs Network Architecture for the SIR Model

For ODE problems, the network, as seen in Figure 3, consists of an input layer of one neuron corresponding to the input time t , which is normalized to the range $[-1, 1]$ during pre-processing, with $\sigma = \tanh(t)$ being the preferred activation function.

FIG. 4: Modified PINNs Training Algorithm

The network is trained to approximate the ODE system, $\vec{u}(t)$, by minimizing a loss function consisting of both differential equation error and data prediction error. For the former, each

equation in the system is rearranged and associated with an error E_i , which is then summed across the system:

$$\frac{du_i}{dt} = f(u_i, t) \Rightarrow E_i = \frac{du_i}{dt} - f(u_i, t)$$

The data prediction error is computed in standard fashion by evaluating the Mean Square Error (MSE) between the predicted compartment values of the output layer and the data provided. While [Raissi et al. \(2019a\)](#) add the data prediction error and differential equation error directly into the loss function, we linearly weight each term according to a new hyperparameter, λ .

$$\mathcal{L}(\vec{\theta}) = (1 - \lambda) \cdot \overbrace{\frac{1}{n} \sum (u_i - u_{\text{pred}_i})^2}^{\text{Data}} + \lambda \cdot \overbrace{\sum E_i}^{\text{Equation}}$$

The computed loss, $\mathcal{L}$, is then minimized using stochastic gradient descent via the Adam optimizer [Kingma and Ba \(2014\)](#). As $\mathcal{L} \rightarrow 0$ over the course of training (See Figure 4), both the data prediction error and differential equation error are minimized, resulting in a trained model that fits the training data and intended ODE structure. As $\lambda \rightarrow 0$, the network prioritizes training data over differential equation structure — and vice-versa as $\lambda \rightarrow 1$. λ provides additional tunability of the model's convergence time and accuracy, and by manipulating λ before or during training, we can optimize training dynamics.

In mechanics problems, inferring the behavior of each compartment from data, initial conditions, and equation parameters suffices. But, for infectious disease modeling, the value of PINNs lies in it's ability to infer dynamics while estimating disease parameters. Using the automatic differentiation [Baydin et al. \(2018\)](#) pipelines available in popular deep learning libraries (ex. TensorFlow v2), we create trainable variables with appropriate bounds for each equation (disease) parameter and allow the optimizer to vary both network and equation parameters during training. After sufficient training, the model will not only approximate the ODE system but the parameters of the system will also be calibrated to the training data.

2.3 Informed Policy: Vaccine Distribution

2.3.1 Preferential Mixing

The pure SIR model assumes that modeled populations are homogenous, but real-world populations are better described as heterogeneous meta-populations with each individual exhibiting differing levels of activity and risk. Person-to-person contact rates vary with population density and personal characteristics including age, gender, and location, inhibiting the generalizability of simple compartmental models to larger populations [Feng et al. \(2015\)](#). We attempt to remedy this assumption by combining PINNs with the simplest meta-population model that captures the heterogeneity of sub-population contact rates and preferential mixing between groups in the meta-population [Jacquez et al. \(1988\)](#).

We begin by dividing the meta-population, N , into n sub-populations, N_i , of individuals exhibiting similar activity levels. We then model each sub-population using SIR, partitioning N_i into compartments based on disease progression: $N_i = S_i + I_i + R_i$. However, we replace the former force of infection βI with λ_i , the per-capita force of infection for the i^{th} population. Formally, let

$$\lambda_i := \beta \sum_{j=1}^n a_i c_{ij} \frac{I_j}{N_j} = a_i \beta \sum_{j=1}^n c_{ij} \frac{I_j}{N_j}$$

where a_i is the average contact rate of the i^{th} sub-population, β is the net probability of infection from an infectious contact, and c_{ij} is the proportion of the i^{th} sub-population's contacts that are with the j^{th} sub-population. The term $a_i c_{ij} \frac{I_j}{N_j}$ can be interpreted as the average infectious contact rate of the i^{th} population with the j^{th} population, and by summing each force contribution across the sub-populations, we arrive at the net force of infection for the population of interest.

In the manner of [Jacquez et al. \(1988\)](#), the mixing matrix c_{ij} is computed by reserving a portion of each sub-population's contacts for itself and then proportionally distributing the

remaining contacts to the other sub-populations [Jacquez et al. \(1988\)](#). Formally, let

$$f_j = \frac{(1 - \epsilon_j)a_j N_j}{\sum_{k=1}^n (1 - \epsilon_k)a_k N_k}$$

$$c_{ij} := \epsilon_i \delta_{ij} + (1 - \epsilon_i)f_j = \begin{cases} \epsilon_i + (1 - \epsilon_i)f_j & i = j \\ (1 - \epsilon_i)f_j & i \neq j \end{cases}$$

where δ_{ij} is the Kronecker delta function and ϵ_i is the proportion of contacts of the i^{th} sub-population reserved for itself. δ_{ij} allows us to isolate the additional ϵ_i contribution when $i = j$, and f_j accounts for the proportional distribution of the remaining contacts by normalizing the general term $(1 - \epsilon_j)a_j N_j$.

We also account for the vital dynamics of the population with μ , merging both birth and death rates, and α , the recovery rate or the reciprocal of the mean infectious period. To avoid further complexity, we assume perfect vaccine efficacy and remove p_i of each population where p_i is the proportion of vaccinated individuals in the i^{th} population. In total, this results in the following SIR-based system of $3n$ compartmental equations and $3n + 3$ parameters.

$$\begin{aligned} \frac{dS_i}{dt} &= \mu N_i(1 - p_i) - (\lambda_i + \mu)S_i \\ \frac{dI_i}{dt} &= \lambda_i S_i - (\alpha + \mu)I_i \\ \frac{dR_i}{dt} &= \mu N_i p_i + \gamma I_i - \mu R_i \end{aligned}$$

for $1 \leq i \leq n$.

2.3.2 Optimization Approach

While PINNs is able to use differential equation structure to fix a large number of equation parameters while approximating the ODE system, we attempt to limit the number of estimated parameters to expedite convergence. We introduce the substitution $A_i = a_i \beta$ and rewrite λ_i and

f_j as follows:

$$\lambda_i = A_i \sum_{j=1}^n c_{ij} \frac{I_j}{N_j}$$

$$f_j = \frac{(1 - \epsilon_j) a_j N_j}{\sum_{k=1}^n (1 - \epsilon_k) a_k N_k} \cdot \frac{\beta}{\beta} = \frac{(1 - \epsilon_j) A_j N_j}{\sum_{k=1}^n (1 - \epsilon_k) A_k N_k}$$

This allows us to eliminate β from the system. In the context of preferential mixing, β acts as a scaling term for a_i and can be completely absorbed into $a_i \rightarrow A_i$. In practice, β is already known from approximating the pure SIR model, in which case, a_i can be directly estimated.

Further, the vaccination rate of each sub-population is typically known beforehand, eliminating n parameters of the form p_i . Additional assumptions about μ can be made, depending on the target meta-population. During training, we remove fixed parameters from the optimizer's trainable variables, allowing us to only train parameters of interest. These optimizations allow us to reduce the number of parameters to $2n + 1$.

2.3.3 Constructing a Distribution Plan

Once the preferential mixing parameters are estimated, we calculate R_v , the effective reproduction number of the meta-population or the average number of secondary infections per infectious person [Diekmann et al. \(2010\)](#). We first define the basic reproduction number for the i^{th} sub-population

$$R_{0_i} = \frac{A_i}{\alpha + \mu}$$

We then remove the proportion of vaccinated individuals to find R_{v_i} , the effective reproduction number for the sub-population.

$$R_{v_i} = (1 - p_i) R_{0_i}$$

To calculate R_v for the entire meta-population, we consider the next-generation matrix K , computed by multiplying the diagonal matrix of sub-population effective reproduction numbers with

the contact proportion matrix with elements c_{ij} .

$$K = \begin{bmatrix} R_{v_0} & & \\ & \ddots & \\ & & R_{v_n} \end{bmatrix} \times \begin{bmatrix} c_{00} & \dots & c_{0n} \\ \vdots & \ddots & \vdots \\ c_{n0} & \dots & c_{nn} \end{bmatrix}$$

The effective reproduction number for the entire meta-population is the spectral radius, or largest eigenvalue, of this matrix [Diekmann et al. \(2010\)](#).

$$R_v = \rho(K)$$

Fixing the remaining parameters, we can analyze R_v as a function of the sub-population vaccination rates p_i in the manner of [Feng et al. \(2015\)](#). We initialize $\vec{p} := \vec{p}_0$ to the estimated or known vaccination rates of the sub-populations. Then, using gradient descent, we update $\vec{p}$ to minimize $R_v(p_1, p_2, \dots, p_n)$.

$$\vec{p} := \vec{p} - \Delta t \cdot \nabla R_v$$

This provides the ideal vaccination rates over time for each sub-population, $p_i(t)$, that minimizes the spread of the infection as fast as possible.

2.4 End-To-End Framework Design

We propose PINNs as a method for directly informing infectious disease policy, particularly in the case of vaccine distribution (See [Figure 5](#)).

FIG. 5: A Directly Informed Policy Pipeline for Vaccine Distribution

Using collected data from a target meta-population, we use PINNs to approximate a modified SIR model, producing additional disease parameters. The parameters are then used to calculate and minimize the effective meta-population reproduction number, R_v , creating an optimal vaccine distribution strategy for the meta-population.

3. COMPUTATIONAL EXPERIMENTS

In this section, we substantiate the accuracy and versatility of PINNs parameter estimation on SIR-based models. We also explore the influence of model hyperparameters and the efficacy of the end-to-end framework in various mixing scenarios.

3.1 Validating PINNs Accuracy

Using Runge-Kutta 4/5, we simulate SIR dynamics for the following initial conditions and disease parameters:

$$\beta = 0.002, \alpha = 0.5, N = 1000, I(0) = 1, R(0) = 0$$

We then sample each curve at 15 random time steps to produce training data. We initialize each trainable disease parameter to 0 and then train a $1 \times 20 \times 20 \times 20 \times 3$ PINNs architecture with a learning rate of 1.0×10^{-3} and error weight of $\lambda = 0.5$ while constraining each parameter to the range $[0, \infty)$. After $\approx 10^5$ iterations, we arrive at the following approximation (See Figure 6).

FIG. 6: PINNs Approximation on Generated SIR Data

Using the simulated curves and predefined parameter values, we precisely compute the PINNs estimation error by finding the MSE between each simulation compartment and the corresponding PINNs approximation using a dense time mesh. For each disease parameter, we simply compare the simulation value with the final value of the corresponding trainable parameter.

Name	Mean Absolute Error	Mean Absolute Percent Error
Susceptible Population	0.24	0.02%
Infected Population	0.18	0.01%
Recovered Population	0.18	0.01%
Transmission Rate: β	1.6×10^{-6}	0.08%
Recovery Rate: α	9.2×10^{-5}	0.018%

TABLE 1: Error Analysis of Approximated SIR Compartments and Parameters

Across all approximations, as seen in Table 1, PINNs maintains under a 0.08% mean absolute

percent error (MAPE), indicating its effectiveness at approximating the SIR model. We also explore the effect of λ on convergence time by recording the number of training iterations needed to achieve $\mathcal{L} \leq 1.0$ while fixing the learning rate (See Table 2).

λ	Iterations ($\times 10^3$)	Training Time (min)
0.25	135	5.6
0.5	88.5	3.7
0.75	105	4.4

TABLE 2: Training Dynamics with Varying λ

We find that $\lambda = 0.5$ achieves optimal convergence time, suggesting that an equal weighting of data prediction error and differential equation error yields the fastest approximations.

3.2 Real-World Data

Having established PINNs' accuracy, we use a popular London boarding school dataset from [Anonymous \(1978\)](#) to evaluate PINNs' ability to perform parameter estimation and compartment approximations on real-world infectious disease data. The dataset records the size of the infected population in a school of $N = 763$ students over the course of 14 days. Due to the lack of data for the Susceptible and Recovered compartments, we only use the infected population to calculate data prediction error. Using $\lambda = 0.5$ as recommended by our previous results, we achieve the following estimation (See Figure 7) after 10 minutes of training on an NVIDIA RTX 2060 GPU.

FIG. 7: PINNs Approximation on Real Data

Despite being provided with only infected data, PINNs is still able to use the mathematical structure of the data to infer the dynamics of $S(t)$ and $R(t)$ as well.

FIG. 8: PINNs Loss Landscape

Analyzing the loss landscape during training (See Figure 8), we see that the disease parameters descend to one local minimum over the course of training, with a final approximation of $\beta \approx 0.0228$ and $\alpha \approx 0.454$.

3.3 Vaccine Distribution

Suppose the London Boarding School dataset had an equal number of total and infected girls and boys* at $t = 0$. Fixing $\alpha = 0.5$, $\mu = 0$, $\epsilon_1 = 0.75$, $\epsilon_2 = 0.5$, $p_i = 0$, $a_1 = 1.5$, and $a_2 = 2.5$, we can again generate and sample training data. We constrain all parameters to the range $[0, \infty)$, except ϵ_i which is further restricted to $[0, 1]$. To avoid numerical computation errors when the parameters are exactly 0, we pad each range by a small $\delta = 1.0 \times 10^{-9}$ such that $[a, b] \rightarrow [a + \delta, b - \delta]$. After $\approx 1.5 \times 10^5$ training iterations, we estimate the following dynamics (See Figure 9).

FIG. 9: PINNs on a Preferentially-Mixed Meta-Population (SIM corresponds to data and NN corresponds to the Neural Network solution)

We again compute the associated error for each approximated compartment and parameter using the simulated dynamics (See Table 3[†]).

Name	Value	Mean Absolute Percent Error
S_1	N/A	0.059%
S_2	N/A	0.094%
I_1	N/A	0.564%
I_2	N/A	0.531%
R_1	N/A	5.22%
R_2	N/A	6.56%
α	0.4999	0.022%
A_1	1.5006	0.040%
A_2	2.4978	0.089%
ϵ_1	0.7449	0.680%
ϵ_2	0.5218	4.361%

TABLE 3: Error Analysis of Preferential Mixing Estimation

While the MAPEs of R_i and ϵ_i are significantly higher, PINNs is still able to accurately estimate each compartment and parameter. As previously described, we employ gradient descent to calculate and minimize the effective reproduction number of the meta-population (See Figure 10).

*This distinction is arbitrary and can be replaced with any plausible division of the student population that results in different contact patterns and activity levels.

[†]Note: The term “value” is only applicable for parameters, so N/A has been used for compartments.

FIG. 10: Optimal Vaccine Distribution Plan (Preferential Heterogeneous Mixing)

The partial derivative of R_v with respect to each p_i represents the need for vaccines within the sub-population. Formally, the proportion of vaccines that should be allocated to the i^{th} sub-population at time t is:

$$\mathcal{P}_i(t) = \frac{\frac{\partial R_v}{\partial p_i} \cdot N_i}{\sum_{j=1}^n \left(\frac{\partial R_v}{\partial p_j} \cdot N_j \right)}$$

FIG. 11: Area Graph of Vaccine Distribution Proportion

Using this formulation, we construct a visual aid for policymakers to calculate and implement the best vaccine distribution proportion. At any given time, policymakers can use Figure 11 to look up the ideal proportions $\mathcal{P}_i$ given the vaccination rate of one of the sub-populations, such as N_1 .

Further, for specific types of mixing, PINNs estimation can be greatly simplified.

1. If $\epsilon_i = 1$, the sub-population is *isolated* and can be separately approximated from the rest of the meta-population.
2. If $\epsilon_i = 0$ for $1 \leq i \leq n$, all contacts are proportional and the $\epsilon_i \delta_{ij}$ term can be omitted from c_{ij} calculation.
3. If $\epsilon_i = \epsilon$ for $1 \leq i \leq n$, the preferential mixing is homogeneous and all trainable variables corresponding to ϵ_i can be replaced with one trainable variable ϵ . This also simplifies $f_j = \frac{a_j N_j}{\sum_{k=1}^n a_k N_k}$.
4. If $\epsilon_i = \epsilon_j$ for some $i \neq j$, the mixing is preferential and heterogeneous and no further simplifications can be made.

For example, assuming that the preferential mixing is homogeneous as described in Case 3, slightly transforms the contours of $R_v(p_1, \dots, p_n)$ (See Figure 12).

FIG. 12: Optimal Vaccine Distribution Plan (Preferential Homogeneous Mixing: $\epsilon_1 = \epsilon_2 = 0.75$)

4. DISCUSSION

4.1 Limitations

Due to the diversity of PDE and ODE models describing infectious disease spread, PINNs can be applied to estimate a large number of simulation parameters, beyond those mentioned in this work. PINNs is also able to avoid local minima and converge to valid parameter combinations despite limited, noisy data. However, as the size of the differential equation system increases, in practice, PINNs may experience convergence issues. When approximating the preferential mixing model, PINNs sometimes fails to estimate ϵ_i despite satisfying the stop loss — the loss value at which the model halts training.

FIG. 13: Grid Search of ϵ_i over PINNs Loss

A grid search over ϵ_i reveals a curve of ϵ_i parameter sets that result in low values of $\mathcal{L}$, suggesting that an additional constraint or substitution can be made involving ϵ_i that simplifies the estimation problem. Alternatively, the range of ϵ_i could be further restricted from $\in [0, 1]$ to a smaller interval (See Figure 13).

Despite achieving training times under 10 minutes on average, PINNs convergence times scale with system complexity. While additional compartments are able to account for multiple vaccination doses and other complex behavior, added differential equations and parameters also require additional training data and larger PINNs networks. PINNs are universal function approximators in theory [Raissi et al. \(2019a\)](#), but in practice, parameter estimation tasks are inherently limited to data availability.

4.2 Conclusions

This work establishes PINNs as a valuable method for informing infectious disease policy through parameter estimation. Combined with optimization approaches, PINNs is able to directly suggest optimal policy decisions, and while compartmental models are perhaps less versatile than ABMs, our coupled approach empirically requires less computational cost than traditional techniques. For example, with the vaccine distribution plan proposed, government officials are able to deploy the optimal vaccine distribution strategy that minimizes the spread of the infectious disease as fast as possible.

Reducing the number of parameters from $3n + 3$ to $2n + 1$ through substitution and assumption allowed PINNs to converge to a parameter solution, and hyperparameter experiments recommend a value of $\lambda = 0.5$ during training. Overall, PINNs is a powerful parameter estimation technique that can drive optimization, but PINNs frameworks should aim to first simplify mathematical formulations before tackling complex compartmental systems.

4.3 Future Work

For vaccine distribution, we aim to re-parameterize the preferential mixing model to elucidate additional constraints between ϵ_i , as conjectured from the previously discussed grid search. Decreasing the number of parameters would facilitate faster parameter estimation and provide additional insight into the mathematical theory behind meta-population models.

By expanding PINNs to other compartmentalized models, parameter estimations for the latent period or vaccination-specific parameters could be computed. With additional compartments, we could also revise our assumption of perfect vaccine efficacy to account for re-infection. Further, research in approximating PDE systems [Cai et al. \(2022\)](#) could allow PINNs to capture the physical mechanics of mucosalivary droplets spreading the infection. This would elucidate the mechanism behind the spread of infectious diseases beyond compartmental modeling.

However, the primary avenue of future PINNs research lies in improving the speed and accuracy of training through advanced machine-learning techniques. Future work involving transfer learning [Desai et al. \(2021\)](#) and pre-training PINNs to model ODE systems with semi-supervised

algorithms [Paticchio et al. \(2020\)](#) is of particular interest. Varied PINNs architectures and training algorithms could decrease PINNs convergence times, allowing the model to use less data to estimate larger systems. Additionally, manipulating λ over the course of training instead of fixing the hyperparameter beforehand may improve training dynamics.

Faster PINNs parameter estimation could enable daily tracking of infectious disease parameters alongside raw reported data. With greater efficiency, PINNs could become the backbone for highly-available, cloud-based, machine-learning dashboards that help policymakers track and minimize infectious disease spread.

FUNDING

This research was partly supported by the second author's grant from the National Science Foundation DMS-2031029.

AVAILABILITY OF DATA AND MATERIAL

Code and experiments are available in the following public GitHub repository. If specific additional details are needed, the authors welcome emails from the readers.

Website: <https://github.com/Claeb101/pinns-vax-dist.git>

CONFLICT OF INTEREST

The authors declare that they have no conflict interests.

The authors have no relevant financial or non-financial interests to disclose.

The authors have no competing interests to declare that are relevant to the content of this article.

REFERENCES

- Akman, D., Akman, O., and Schaefer, E., Parameter Estimation in Ordinary Differential Equations Modeling via Particle Swarm Optimization, *J. Appl. Math.*, vol. **2018**, 2018.
- Akman, O., Corby, M.R., and Schaefer, E., Examination of Models for Cholera, *Lett. Biomath.*, vol. **3**, no. 1, pp. 93–118, 2016.
- Anonymous, Influenza in a Boarding School, *BMJ*, vol. **1**, pp. 587–587, 1978.
- Baydin, A.G., Pearlmutter, B.A., Radul, A.A., and Siskind, J.M., Automatic Differentiation in Machine Learning: A Survey, *J. Mach. Learn. Res.*, vol. **18**, pp. 1–43, 2018.
- Bouchnita, A., Chekroun, A., and Jebrane, A., Mathematical Modeling Predicts That Strict Social Distancing Measures Would Be Needed to Shorten the Duration of Waves of COVID-19 Infections in Vietnam, *Front. Public Health*, vol. **8**, p. 559693, 2021.
- Butcher, J.C., *Numerical Methods for Ordinary Differential Equations*, Hoboken, NJ: John Wiley & Sons, 2016.
- Cai, S., Mao, Z., Wang, Z., Yin, M., and Karniadakis, G.E., Physics-Informed Neural Networks (PINNs) for Fluid Mechanics: A Review, *Acta Mechanica Sinica*, pp. 1–12, 2022.
- Capaldi, A., Behrend, S., Berman, B., Smith, J., Wright, J., and Lloyd, A.L., Parameter Estimation and Uncertainty Quantification for an Epidemic Model, *Math. Biosci. Eng.*, p. 553, 2012.
- Desai, S., Mattheakis, M., Joy, H., Protopapas, P., and Roberts, S., One-Shot Transfer Learning of Physics-Informed Neural Networks, arXiv:2110.11286, 2021.
- Diekmann, O., Heesterbeek, J., and Roberts, M.G., The Construction of Next-Generation Matrices for Compartmental Epidemic Models, *J. R. Soc. Interface*, vol. **7**, no. 47, pp. 873–885, 2010.

E, W., Han, J., and Jentzen, A., Deep Learning-Based Numerical Methods for High-Dimensional Parabolic Partial Differential Equations and Backward Stochastic Differential Equations, *Commun. Math. Stat.*, vol. **5**, no. 4, p. 349–380, 2017.

Feng, Z., Hill, A.N., Smith, P.J., and Glasser, J.W., An Elaboration of Theory about Preventing Outbreaks in Homogeneous Populations to Include Heterogeneity or Preferential Mixing, *J. Theor. Biol.*, vol. **386**, pp. 177–187, 2015.

Hornik, K., Approximation Capabilities of Multilayer Feedforward Networks, *Neural Networks*, vol. **4**, no. 2, pp. 251–257, 1991.

Jacquez, J.A., Simon, C.P., Koopman, J., Sattenspiel, L., and Perry, T., Modeling and Analyzing HIV Transmission: the Effect of Contact Patterns, *Math. Biosci.*, vol. **92**, no. 2, pp. 119–199, 1988.

Kermack, W.O. and McKendrick, A.G., A Contribution to the Mathematical Theory of Epidemics, *Proc. R. Soc. London Ser. A Math. Phys. Char.*, vol. **115**, no. 772, pp. 700–721, 1927.

Kerr, C.C., Mistry, D., Stuart, R.M., Rosenfeld, K., Hart, G.R., Núñez, R.C., Cohen, J.A., Selvaraj, P., Abeysuriya, R.G., and Jastrzebski, M., Controlling COVID-19 via Test-Trace-Quarantine, *Nat. Commun.*, vol. **12**, no. 1, pp. 1–12, 2021a.

Kerr, C.C., Stuart, R.M., Mistry, D., Abeysuriya, R.G., Rosenfeld, K., Hart, G.R., Núñez, R.C., Cohen, J.A., Selvaraj, P., and Hagedorn, B., Covasim: An Agent-Based Model of COVID-19 Dynamics and Interventions, *PLOS Comput. Biol.*, vol. **17**, no. 7, p. e1009149, 2021b.

Kingma, D.P. and Ba, J., Adam: A Method for Stochastic Optimization, arXiv:1412.6980, 2014.

Kucharski, A.J., Klepac, P., Conlan, A.J., Kissler, S.M., Tang, M.L., Fry, H., Gog, J.R., Edmunds, W.J., Emery, J.C., and Medley, G., Effectiveness of Isolation, Testing, Contact

Tracing, and Physical Distancing on Reducing Transmission of SARS-CoV-2 in Different Settings: a Mathematical Modelling Study, *Lancet Infect. Dis.*, vol. **20**, no. 10, pp. 1151–1160, 2020.

Ling, J., Kurzawski, A., and Templeton, J., Reynolds Averaged Turbulence Modelling using Deep Neural Networks with Embedded Invariance, *J. Fluid Mech.*, vol. **807**, 2016.

Panovska-Griffiths, J., Kerr, C.C., Stuart, R.M., Mistry, D., Klein, D.J., Viner, R.M., and Bonell, C., Determining the Optimal Strategy for Reopening Schools, the Impact of Test and Trace Interventions, and the Risk of Occurrence of a Second COVID-19 Epidemic Wave in the UK: A Modelling Study, *Lancet Child Adolescent Health*, vol. **4**, no. 11, pp. 817–827, 2020.

Paticchio, A., Scarlatti, T., Mattheakis, M., Protopapas, P., and Brambilla, M., Semi-Supervised Neural Networks Solve an Inverse Problem for Modeling Covid-19 Spread, arXiv:2010.05074, 2020.

Quach, H.L., Nguyen, K.C., Hoang, N.A., Pham, T.Q., Tran, D.N., Le, M.T.Q., Do, H.T., Vien, C.C., Phan, L.T., and Ngu, N.D., Association of Public Health Interventions and COVID-19 Incidence in Vietnam, January to December 2020, *Int. J. Infect. Dis.*, vol. **110**, pp. S28–S43, 2021.

Raissi, M., Perdikaris, P., and Karniadakis, G.E., Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations, *J. Comput. Phys.*, vol. **378**, pp. 686–707, 2019a.

Raissi, M., Ramezani, N., and Seshaiyer, P., On Parameter Estimation Approaches for Predicting Disease Transmission through Optimization, Deep Learning and Statistical Inference Methods, *Lett. Biomath.*, vol. **6**, no. 2, pp. 1–26, 2019b.

Smirnova, A., deCamp, L., and Chowell, G., Forecasting Epidemics through Nonparametric

Estimation of Time-Dependent Transmission Rates using the SEIR Model, *Bull. Math. Biol.*, vol. **81**, no. 11, pp. 4343–4365, 2019.

Sulis, E. and Terna, P., An Agent-Based Decision Support for a Vaccination Campaign, *J. Med. Syst.*, vol. **45**, no. 11, pp. 1–7, 2021.

Yang, H.M., Lombardi Junior, L.P., Castro, F.F.M., and Yang, A.C., Mathematical Modeling of the Transmission of SARS-CoV-2—Evaluating the Impact of Isolation in São Paulo State (Brazil) and Lockdown in Spain Associated with Protective Measures on the Epidemic of CoViD-19, *PLoS One*, vol. **16**, no. 6, p. e0252271, 2021.

$$N = S + I + R$$

Susceptible

β

Infected

α

Recovered

$$\frac{dS}{dt} = -\beta SI$$

$$\frac{dI}{dt} = \beta SI - \alpha I$$

$$\frac{dR}{dt} = \alpha I$$























