

## Project: Bacterial growth

### Part 1: Demonstration in MATLAB

Many dynamical systems are self-limiting because interactions within the system create negative feedback. Logistic growth captures this behavior by tying the growth rate to both the current size and the remaining capacity:

$$\frac{dx}{dt} = \lambda x(t)(1 - (x(t)/\theta)^\alpha), \quad x(0) = p_0 \quad (1)$$

where

- $\lambda$  is the intrinsic *reproduction rate*,
- $\theta$  is the *carrying capacity*, and
- $\alpha$  is an *impedance* (shape) parameter that controls how quickly the ceiling is “felt” as  $x(t)$  grows.

When  $\alpha = 1$  the model reduces to the classical logistic equation; other values reshape the trajectory’s curvature. Because resources or crowding often curb growth, this structure applies broadly—from bacterial colonies to tumors.

Given  $(\lambda, \theta, \alpha)$ , standard ODE solvers yield the trajectory  $x(t)$ . Given data  $\{(t_i, x_i)\}$ , we can estimate the parameters that best fit the observations, and then use the fitted model to predict future behavior. We shall use MATLABs built in *fminsearch* to fit the logistic growth model to synthetic noisy data.

We will demonstrate using this code briefly, materials can be found at:

<https://github.com/CalebHend/MCSB-Bootcamp-Data-Fitting/tree/main/fminsearch-MATLAB-Demo>.

**File set up Part 2** The next sections material is found at:

<https://drive.google.com/drive/folders/10YSaciWFgAvxpZKgGNzzt9tpcDoNeF4W?usp=sharing>

Have one member of your group **make a copy** of the folder in their Drive and **share and edit the copy** with the group. Work only in the copied folder.

---

## Part 2: Physics-Informed Neural Networks (PINNs)

In this section you will use Physics-Informed Neural Networks (PINNs) to estimate parameters in a biological growth model. PINNs combine data fitting with the governing differential equation by penalizing both the data misfit and the equation residual. Building on the bacterial growth model from part (a), you will infer unknown parameters from experimental measurements—going beyond a pure sum-of-squared-errors (SSE) approach.

### Problems:

- (1) Click the the hyperlink to bring you to the drive where you will see a few Colab notebooks. In this drive open *Simple NN* where our first exploration will begin. Follow the instructions provided in the notebook, and read it over carefully. Also answer the following questions when you get familiar with this code:

- (a) How well does this NN perform? What do you notice about the plots?
- (b) Can we improve its fitting capabilities by adjusting network depth and width?

- (2) For this next part you will need to open *Simple NN regularization* and *linear estimate parameters* where we will explore a simple linear fit using a PINN along with a parameter search. Again follow the instructions in the notebook and play around with the code.

Also answer the following questions when you get familiar with this code:

- (a) How well does this PINN perform? What do you notice about the plots in contrast to (1)?
- (b) Can we improve its fitting capabilities by adjusting network depth and width?
- (c) What does  $\eta_{\text{phys}}$  do to the fit when increased and decreased?
- (d) Repeat these questions for the *linear estimate parameters* note book.

- (3) Now you should be familiar enough with the structure of a PINN, such that you can now explore the *simple NN regularization Logistic Growth*. Where now we will pretend we know the parameters.

- (a) How well does this PINN perform? What do you notice about the plots?
- (b) Can we improve its fitting capabilities by adjusting network depth and width?
- (c) What does  $\eta_{\text{phys}}$  do to the fit when increased and decreased?

- (4) Now consider that we only guessed that the model we are using is the logistic growth but we have no idea what the parameters are. We now will use the PINN to extract the biologically relevant parameters. You now must open the *inverse logistic growth pinn*, where you will now answer the following questions after reading over the code:

- (a) How well does this PINN perform? What do you notice about the plots in contrast to (3)?
- (b) Can we improve its fitting capabilities by adjusting network depth and width?
- (c) Try fixing and unfixing parameters, what happens? What can you do to justify fixing a parameter?
- (d) What does  $\eta_{\text{phys}}$  do to the fit when increased and decreased?

- (5) Now we shall use your data gathered in the lab portion to extract the parameters for your experiments using *inverse logistic growth pinn*. Modify the code to do so and answer the following questions:

- (a) How well does the PINN fit your data? What observations can you make?
- (b) How does tuning the PINN, affect the performance?
- (c) How could you improve this code and or method to get better results.

**Bonus: Thinking beyond outside the  $> \epsilon$  ...**

Can anyone tell me what models would work better for your experimental data? And how would you validate this claim?