

MATH / PUBH-EPI 5421
Lab #1

The purpose of this lab two-fold: 1) to make some of the models and concepts discussed in lecture more tangible, and 2) to get you acquainted with implementing models in the computer. Specifically, we will be using Matlab, a “high-level” programming language that is very useful for scientific computing. In particular, you can use Matlab for running model simulations, visualizing data and model output, and fitting models to data. (Matlab is not the only possibility – alternatives include R and SciPy. You are welcome to use any language you like for implementing your models in this course – but Matlab is what we’ll discuss in the labs.)

You can install Matlab on your own computer using the OSU license: <https://ocio.osu.edu/software/directory>.

Always useful is the help command – typing `help <function_name>` into the Matlab command window gives information on `function_name`.

Below are some suggested exercises for exploring the SIR model and learning Matlab (not graded).

L1.1. Download the files `SIR.m` and `goSIR.m`. `SIR.m` is the Matlab *function* that specifies the right hand side of the differential equation (in this case, the scaled SIR model), and `goSIR.m` is a Matlab *script* that numerically integrates the SIR system and produces a solution trajectory.

- Add the directory where you’ve downloaded `SIR.m` and `goSIR.m` to Matlab’s path. You can do this either through the “Set Path” menu option, or by the `addpath` command: `addpath <directory name>`.
- Type `goSIR` in the command window. This runs the script `goSIR.m` – the same as if the contents of `goSIR.m` were typed into the command window. What do you see for $I(t)$ vs. t ? For $S(t)$ vs. t ?
- Open `goSIR.m` and play with the parameter values for β and γ . How do the solution trajectories $S(t)$, $I(t)$ depend upon these parameters? Are the results consistent with what we’ve learned in class about \mathcal{R}_0 ?

L1.2. (Final outbreak size.) The function `calcOutbreakSize.m` numerically solves for the final outbreak size as a function of \mathcal{R}_0 .

- Download `calcOutbreakSize.m` and `newton.m` from Carmen.

- Use `calcOutbreakSize` to compute the final outbreak size for $\mathcal{R}_0 = 2$.
- Run `goSIR.m` with $\beta = 0.5$, $\gamma = 1/4$. The third column of `x` corresponds to $R(t)$. Calculate `x(end,3) - x(1,3)`. Why is this the outbreak size? How does it compare with the predicted value from solving the final outbreak size relation using `calcOutbreakSize`?

L1.3. (Eigenvalues of the Jacobian.) The function `jacobianSIR.m` takes in the phase space values (i.e. values for S and I), and returns the Jacobian at that phase space value.

- For $\mathcal{R}_0 > 1$, recall that the endemic equilibrium for the SIR system is given by $s^* = 1/\mathcal{R}_0$, $i^* = \frac{\mu}{\mu+\gamma}(1 - 1/\mathcal{R}_0)$. Let $\mu = 1/70$, $\gamma = 10$, and $\mathcal{R}_0 = 5$. Compute the Jacobian at (s^*, i^*) for these parameter values.
- Use the `eigs` command to compute the eigenvalues of $J|_{EE}$. Based on these eigenvalues, what do you expect to see regarding disease dynamics near the endemic equilibrium?
- Using the parameter values above, run the SIR system for 100 years, from initial conditions $s_0 = 0.9999$, $i_0 = .0001$. Do you see what you expected?