Week 2 Notes (5/29/18)

During our meeting:

- Any questions about chapter 1 of Wiggins?
- Review worksheet and answers
- Review/walkthrough code and figures
- Discuss nondimensionalization of gLV equations
- Discuss Lyapunov functions for dynamical systems

For next week:

- Read chapter 2 of Wiggins
- Show that 2D gLV equations can be nondimensionalized so that

$$\begin{aligned}\frac{\mathrm{d}x_a}{\mathrm{d}t} &= x_a(\mu_a - x_a - M_{ab}x_b)\\ \frac{\mathrm{d}x_b}{\mathrm{d}t} &= x_b(\mu_b - M_{ba}x_a - x_b),\end{aligned} \tag{1}$$

  and that you may further nondimensionalize time by setting $\mu_a \to 1$.

- Show that the Lyapunov function given by Tang, Yuan and Ma in Phys. Rev. E 2013 satisfies the Lyapunov conditions given in chapter 2 of Wiggins. In our notation, the Lyapunov function they provide is

$$\begin{aligned}V(x_a, x_b) = {} & M_{ba}x_a^2/2 + M_{ab}x_b^2/2\\ & - M_{ba}\mu_a x_a - M_{ab}\mu_b x_b + M_{ab}M_{ba}x_a x_b\end{aligned} \tag{2}$$

  Some hints: 1) remember that $\hat{x}_a$ and $\hat{x}_b$ are "directions" (like $\hat{x}$ and $\hat{y}$) in a 2-dimensional space, 2) remember that $\dot{V} = \nabla V \cdot \dot{\mathbf{x}}$, where $\dot{\mathbf{x}}$ is the vector form of the dynamical system, and 3) assume $M_{ab} > 0$ and $M_{ba} > 0$ (this condition must be satisfied in order for there to be two stable steady states). Note that this equation satifies the Lyapunov conditions except for the fact that $V(\bar{x}) = 0$ for two different $\bar{x}$, corresponding to the two stable steady states. For this reason, this is called a *split Lyapunov function*.

- Generalize your code for solving the 2-dimensional gLV equations so that it can solve N-dimensional gLV equations. Hint: consider what the commands `np.dot(np.diag(mu), Y)` and `np.dot(np.diag(np.dot(M, Y)), Y)` do, and compare them to the N-dimensional gLV equations. Test your code out using the parameters from Stein et al., Plos Comp Biol 2013— I have provided code that imports these parameters and turns them into numpy arrays for you. For usage, look at `example_import_data.py` and for the implementation itself look at `barebones_CDI.py`. Compare the parameter values from the python code with the Stein paper and ensure they agree. This code also imports experimental initial conditions from the Stein paper (e.g. `ic4` in the code). Don't plot your output, but try starting from different initial conditions (0-8 are allowed; 4 is given as an example in `example_import_data.py`) and compare your obtained steady states (`y[:, -1]`) with Table B in `S1_Appendix_revision.pdf`