Homework problem 5 – Recurrent Networks

Deliverable: One document (ideally PDF, MS Word is OK too) containing the answers to all questions and the figures specified in the task assignments below. You should not include all figures produced by the code. You do not have to include the modified code.

Part 1: Wilson-Cowan (WC) model of excitatory and inhibitory populations

In the first part of this exercise, we will simulate and analyze the dynamics of a simple Wilson-Cowan model with one excitatory and one inhibitory population. The exercise follows lecture 3-5 Theoretical Primer on Nonlinear Dynamical Systems (video available on Kaltura on Canvas).

Run Section 1 of the code ("NTP640_HW5_RecurrentNetworks_Part1.m").
You should get one figure with the FI curves of the E and I population and another figure with two example trajectories that show the temporal evolution of the WC model starting from slightly different initial conditions.

Q_1.1: How does the behavior of the model differ for the two different initial conditions and why? (1 pt)

Task 1.1:
Set the initial value of the inhibitory population to 0.2 in line 59 and line 66.
Now adjust the initial value of the excitatory population in line 65 so that both populations again settle on positive activation values. Include the resulting figure.
At which value of rE_init does the switch occur (2 decimal values)? (2 pts)

Phase plane analysis

Run section 2. You should see one new figure with one blue trajectory plotted in the phase plane.

Q_1.2: What is a phase plane plot (i.e. what is plotted in the figure)? Does this plot have explicit information about time? (2 pts)

Task 1.2: Add the second trajectory from above into the figure using a different color. Include the resulting figure here. (1 pt)

Run section 3. You should see a figure with the nullclines and a vector field. The excitatory nullcline (blue) corresponds to those points in the phase plane at which there is no change in the activity of the excitatory population rE for a given value of rI (and vice versa for the inhibitory nullcline shown in red).
The Nullclines for the excitatory and inhibitory populations are defined as follows:

$$r_I = 1/w_{EI}[w_{EE}r_E - F_E^{-1}(r_E) + I_E^{ext}]$$

and

$$r_E = 1/w_{IE}[w_{II}r_I - F_I^{-1}(r_I) + I_I^{ext}].$$

Q_1.3: How do we arrive at these equations starting from the differential equations of the two populations r_E and r_I? (Explain the idea, no need to derive the equations). (1 pt)

Q_1.4: Where would a trajectory starting at rE_init = 0.5 and rI_init = 0.8 end up? Will all trajectories with rI_init = 0.9 end up at the same steady state attractor? How often do the nullclines cross? Are all of these cross points attractor states of the WC model? Why do the arrows tend to get smaller as they approach the fixed points? (5 pts)

Task 1.3: Add 3 example trajectories with very different initial conditions to the nullcline figure (ideally in green). Include the resulting figure. (2 pts)

Task 1.4: Go back to the start of the script. Let's investigate how the nullclines and thus the fixed points depend on a particular variable: wEE. Decrease wEE and run Section 3 again (you will also have to run the first part of Section 1 to change the parameters). At which value of wEE does the number of fixed points change (with one decimal accuracy)? Include nullcline figure for that value. What happens when wEE gets weaker? (3 pts)

Short pulse induced persistent activity

Set the value of wEE back to wEE = 9.0 (rerun just this part of the code to update the value).
Now run Section 4 of the code. Here we apply noisy inputs to the WC model. The same input is applied to the excitatory and inhibitory neuron population. The noise is simulated according to an Ornstein-Uhlenbeck (OU) process, which is often used for this kind of modeling and corresponds to filtered (i.e. "colored") white noise.
You should get a figure with the noise input in the top panel and the E-I activities in the bottom panel. Our initial values for the excitatory and inhibitory population were small in this case and the activity stays low but is more noisy than before.

Task 1.5: Look at the code where it says, "Add short pulse". Increase the variable SE, which corresponds to the strength of the short pulse we will add to the noise input. If you increase it enough, you should see a persistent change in the activity levels of the excitatory and inhibitory population long after the end of the pulse. Include the figure. Explain the effect of the short pulse. (2 pts)

Task 1.5: Almost done! Go back to the start of the script. Let's use a different set of parameters: wEE = 6.4, wEI = 4.8, wIE = 6, wII = 1.2, I_ext_E = 0.8. Run section 1-3 again and include the phase plane figures below. What happened to the fixed points and trajectories? (3 pts)

Task 1.6: Finally, with the new set of parameters. Decrease tau_I to 1. How do the trajectories look like now and what kind of attractor is this? Include the phase plane figure. (2 pts)

(Total points part 1: 24)

Part 2: Winner-take-all dynamics

In this part, we will increase the number of excitatory populations by one and analyze a Wilson-Cowan model with two excitatory and one inhibitory population. The code for this part of the exercise is "NTP640_HW5_RecurrentNetworks_Part2.m". Running this code may require Matlab's optimization toolbox. Please let me know immediately if you have any issues.

Task 2.1: Draw a connectivity diagram (by hand or however you want) that contains the three populations of neurons and their connectivity weights as specified in the default parameters and their external inputs and include it in your file. (2 pts)

Task 2.2: Run the whole code and include the final figure that plots the stable and unstable fixed points of the model over a range of symmetric inputs to the excitatory populations of neurons.
(Note: Running section 3 might take a minute. If it takes too long on your computer you can reduce the numInitialConditions to 100 and also use larger steps in I_ext_E_range.)
At which input values does the system transition from one dynamical regime into another? Which range of I_ext_E corresponds to winner-take-all behavior? (3 pts)

Task 2.3: Go to section 2 in the code. Based on the fixed-point analysis figure, chose at least 5 values of symmetric external input (meaning E1 and E2 get the same value) and plot the trajectories. The values should be chosen from the different dynamical regimes and closer/further from the bifurcations. Include the trajectory figures and the corresponding input values. Within the "winner-take-all" regime, can you see differences in the time it takes to reach the fixed-point values? Note: You may also want to try different random seeds for the OU noise. (3 pts)

Task 2.4: Start with symmetrical inputs of I_ext_E = 2.0 to both excitatory populations. Now increase the input to one of the populations by 0.25 and then by 0.5. What effect does this have on the trajectories. Include the figures below. (2 pts)

Task 2.5: Final task! Set the inputs back to I_ext_E = 2.0 for both excitatory populations. Now increase the input to the inhibitory population to 2.0 and run the whole code again. Include the two resulting figures below. What effect did the increase of the inhibitory input have on the fixed-points and bifurcations? How does that explain the difference in the trajectory plot compared to an input of 1.0 to the inhibitory population? (3 pts)

(Total points part 2: 13)