FIT3139: Assignment 2

(Due by 11:59pm, Sunday, 14 June 2020)

[Weight: $15 = 3 + 12$ marks.]

This assignment has two parts. The objective of the first part is to implement your own RK2 numerical method, and use it to inspect the dynamics of the replicator equation; this part is worth 3 out of 15 marks. The second part of the assignment is about using Markov Chain Theory to formulate and analyse a simple model of social segregation; this part is worth 12 of 15 marks.

Follow these procedures to submit this assignment

The assignment must be submitted *online* via Moodle, and should follow the following procedure:

- Accept the Electronic Plagiarism Statement for this Assignment. All your scripts/program will be scanned using MOSS (a plagiarism detection software). Read Monash Student Academic Integrity policy for consequences of plagiarism.
- All your scripts and reports **MUST** contain your name and student ID.
  - **You are free to program the assignment in either MATLAB or Python.**
  - Your submitted archive must extract to a directory named as your student ID.
  - This directory should contain a subdirectory for each of the two questions in the assignment, named as q1/, and q2/.
  - Within each of those subdirectories the contents include `MATLAB/Python A PDF` report with references to the scripts you used (in Python or MATLAB). You should include the scripts as well.
  - When submitting scripts and reports, choose file names that identify the subquestion. (Eg. `q1a_script.py`, or `q1b_report.pdf`, or `q2_script_driver.m`, or `q2_output.txt`)
- Submit your zipped file electronically via Moodle.

Part 1

The replicator equation is a dynamical system used in theoretical biology and social science to describe the evolution of phenotypes that interact in a population, or a process of social learning. A system with three types has a state (x, y, z) with each number of the vector describing a *frequency* of the corresponding phenotype. This means, $x + y + z = 1$, and $0 \le x \le 1$, $0 \le y \le 1$ and $0 \le z \le 1$. It is assumed that all types are present at t_0.

The following replicator dynamics arises from a simple Rock-Paper-Scissor game, with three types (rock, paper and scissor).

$$\dot{x} = x(ry + sz - \phi)$$
$$\dot{y} = y(sx + rz - \phi)$$
$$\dot{z} = z(rx + sy - \phi)$$

with $\phi = (r + s)(x(y + z) + yz)$.

Questions Part 1

(a) Using your own RK2 numerical integrator, inspect the dynamics of this system for different initial configurations when $r = -1$ and $s = 1$.

(b) Describe how the behaviour of the system changes, when you set the parameters such that:

- $r + s > 0$
- $r + s < 0$

What should you submit for this question?

You will have to submit

- Your own implementation of RK2.
- A report for the case $r = -1, s = 1$, including any code that you have used in producing the report.
- A report for the case $r + s > 0$, including any code that you have used in producing the report.
- A report for the case $r + s < 0$, including any code that you have used in producing the report.

Part 2: A simplified Schelling model

Background

The Schelling model was proposed by Thomas Schelling to explain segregation. We discussed the basic elements of the model in the first lecture of the unit. Because the model has a very large number of possible states it is hard to compute the quantities of interest exactly, mostly having to rely on Montecarlo simulations. A simple implementation of this simulation model can be found here: `http://www.natureincode.com/code/various/schelling.html`.

To make this model tractable we propose a simplified version of the model, using Markov chains. The model is described below *.

In the simplified Schelling model agents live on a cycle of finite size n. Agents can be of two types, say 0 and 1. There are no empty positions, thus, a cycle of size n also implies n agents.† In this simplified

*Please read carefully, perhaps aloud, and several times, being aware of the punctation marks

†This is a big difference with the standard Schelling model in which vacant spots are a fundamental feature.

version there are no thresholds. Instead, an agent is "happy" if at least one of her neighbours is of the same type.

Time is discrete, so $t = 1, 2, 3, \ldots$. The dynamics go as follows: At each time-step, two individuals (residing in different slots, not necessarily adjacent) are matched to *potentially* trade places. Each encounter may result in the agents trading places or retaining their position. Agents will agree to trade places if and only if at least one of the two agents benefits[‡], *and* none of the two is worse off after the swap. The *matching* procedure is randomly uniform. It is easy to see that this process gives rise to an absorbing Markov Chain.

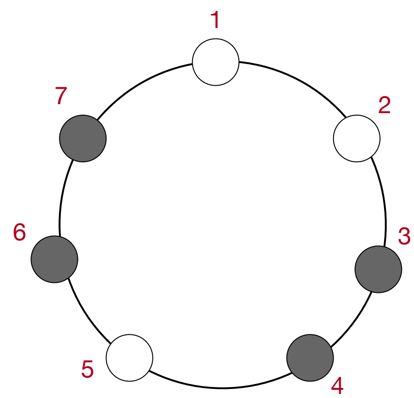

Figure 1: Example: A cycle of size 7. According to the rules above, all agents are "happy" except agent 5

Questions Part 2

(a) An absorbing Markov Chain:

- Specify explicitly the transition Matrix of the MC for $n = 4$. Explain how the transition probabilities are computed and how the states are labeled.
- Show the canonical form of the Markov chain for $n = 4$. Make sure to specify clearly how states are re-labeled or re-ordered if necessary.
- Using Montecarlo simulations show how the absorption time varies with $n = 4, 5, \ldots 10$. [§]
- Numerically approximate the absorption times for $n = 4$ and $n = 5$ and show that they agree with the Montecarlo simulations.[¶]

What should you submit for this question?

Your submission must include a report for part 2A, answering the questions and referencing any scripts used to perform the calculations/simulations. The scripts should also be included as part of your submission. Alternatively, if using Python, you can submit a Jupyter notebook containing descriptive text and code.

(b) We now turn to a model in which agent may swap places "by mistake". This means with a probability ϵ they will fail to swap places when they intend to, or will fail to stay put when they should. This small change results in a new chain that is ergodic.[‖]

[‡] i.e., goes from unhappy to happy

[§] Assume you have $n/2$ type 0's and $n/2$ type 1's for even n; or $(\lfloor n/2 \rfloor + 1)$ type 1's for odd n. Note that for this larger n you do not necessarily need to formulate the whole transition matrix of the Markov chain. You can simply simulate the events that transform one state into another.

[¶] For the numerical calculations in the case of n=5, it may not be necessary to specify a full transition matrix.

[‖] For numerical and simulation results assume a small ϵ

- Specify the full transition matrix for $n = 4$, compute the stationary distribution numerically and show that it is in agreement with Montecarlo simulations. What can you conclude from this model?
- Repeat this analysis in the case where agents do not live on a cycle, but on a simple linear structure; i.e., the agents on both ends only have one neighbour.
- Discuss reasonable extensions of this model that would allow for richer, and perhaps more realistic dynamics, while keeping tractability at hand. What can you conclude from this exercise?

What should you submit for this question?

Your submission must include a report for part 2B, answering the questions and referencing any scripts used to perform the calculations/simulations. The scripts should also be included as part of your submission. Alternatively, if using Python, you can submit a Jupyter notebook containing descriptive text and code.