Background The prisoners dilemma[1] is a two-player, non-zero-sum game that is often used as a model of the problem of cooperation. Its main feature is that the best individual actions for each player, defection, causes them to end up in a situation that leaves them worse off than what they would achieve if they cooperated.

In the standard story[2], we have two partners in crime in jail. They are kept separately and cannot communicate. They both get an offer: if they confess and witness (known as defecting) against their partner they will be released without punishment while their partner gets five years in prison, unless the partner also confesses in which case they both get three years. If none confesses (cooperates), they will both get a one year sentence.

For each individual, it is clearly best to defect: if their partner cooperates (does not witness) they will go free instead of getting a one year sentence, while if their partner defects they will get three years instead of five. However, as the game is symmetric both will follow this logic and end up in the Nash equilibrium of both defecting with both getting three years, as opposed to the one year they would get by cooperation.

In a finitely repeated game, where the number of rounds is fixed, the Nash equilibrium is still to defect from the first round. This is because you certainly should defect on the last round as this is the same situation as in the one-shot version of the game. But knowing this, you should also defect on the next-to-last round, and thus also on the one before that etc. This way of reasoning is called backward induction.

Model In this model we will use a reduced strategy space that highlights the backward induction in a N-round game. This space contains $N+1$ strategies on the form "Cooperate until round i or the other player defects, whichever comes first, and then defect for the rest of the game." We denote these strategies S_i, where in a game of N rounds i runs from 0 (always defect) to N (always cooperate).

We name the payoffs for a single game as: T (temptation) for defecting when the opponent cooperates; R (reward) when both cooperate; S (sucker)

[1]For an entertaining listening that relates to the prisoner's dilemma, check out this episode of Radiolab.

[2]...which doesn't really drive home the point as there still are a lot of extra considerations real prisoners would have towards each other, their own reputation and so on. For an example of an effort to get at the (in the author's words) "*true* prisoners dilemma", see Yudkowski's short sci-fi story Three Worlds Collide.

for cooperating with a defector; and P (punishment) when both defects. (It might be instructive to draw up the payoff-matrix). To make this a prisoner's dilemma, rather than some other two-person game like the hawk-dove game, we need to have $T > R > P > S$.[3] We don't need the full four parameters, however. In the evolutionary dynamics described below, only ranking matters; adding or multiplying each parameter with a constant gives the same dynamics. We therefore use a reduced parameter space where $R = 1$ and $S = 0$, and where $T \in [1, 2]$ and $P \in [0, 1]$.

Use a two dimensional cellular automaton (CA) approach, where each cell corresponds to one player. Each cell has a state $i \in \{0, 1, \dots N\}$, representing the player's strategy S_i. Each time-step has two stages: competition and reproduction. The competition step has all players playing the N-round iterated prisoners dilemma with their von Neumann-neighbors[4] (we also assume periodic boundary conditions). Each player obtains a total score from the four N-round games they played (the sum of $4N$ payoffs). In the reproduction step, these scores are used for selection of strategies as follows: Each player (cell) changes its strategy (state) to that of its neighbors (including itself) that received the highest total score (break ties randomly). The reproduction step is finished by mutating each cell into a random different state with some small probability ϵ, which allows for exploration of the strategy space.

Note that in a strict CA formulation this would be a range-2 interaction, as a cell's new state depends on its neighbors' score, which in turn depend on how these neighbors played against their neighbors. However, this is a side note as the simplest way to program the model is to use the above formulation with two steps.

Exercises

1. Implement the basic model and visualise it. Also plot the time evolution on the population level, see Fig. 1. For the general parameters you can use a lattice of size $L = 32$, a mutation rate $\epsilon = 1/L^2$, and a seven round game. **To report**: 1) A figure similar to Fig. 1 for $T = 1.5$,

[3] For the iterated prisoners dilemma we in general also need $2R > T + P$ so that taking alternative turns cooperating and defecting doesn't beat the pure cooperation. In our case however, this is a strategy that isn't available due to the reduced strategy space, so we disregard this requirement.

[4] The nearest von Neumann neighbors of a cell is the four cells to its immediate right, left, top, and bottom.

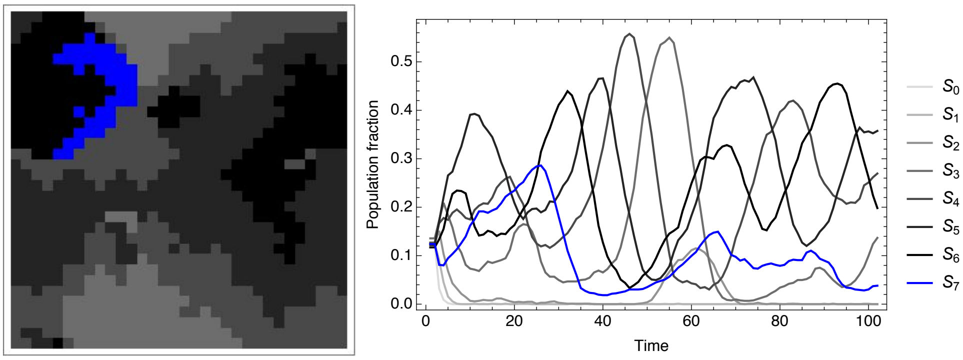

Figure 1: A snapshot of the CA after 100 time steps and the time evolution of its population fractions for $T = 1.5$ and $P = 0.5$ with $N = 7$ rounds. Blue is all cooperate (S_7), black to lighter gray is S_6 to S_0 (only S_7 to S_3 present in the snapshot).

$P = 0.5$, and 50 time steps; 2) A figure similar to Fig. 1 for $T = 1.5$, $P = 0.5$, and 1000 time steps. Explain what we see in the figures. **(7p)**

2. The model has several regimes depending on the payoff parameters T and P. Find the three main regimes. **To report**: Figures similar to Fig. 1 for each regime. Characterise the regimes and provide a description of how they differ. **(5p)**

3. Map out the phase diagram of the model. Grid the $T, P-$square with something like 10 interior points in each dimension and determine what phase (regime) the model assumes at each point. There are several ways to construct the phase plot. Here are a few suggestions: 1) Identify the phases manually by examining the CA plots and the population level plots; 2) Set up an automatic procedure to identify the phases; 3) Use the CA plots themselves as points in the phase diagram. **To report**: A figure with the phase diagram with some suitable colouring for the different phases and an explanation of what we see in the figure. **(8p)**

4. So far, the focus has been on exploring the model. Now, spend some time to mull over the implications of your findings, what the benefit or drawbacks of such a model is, and how general are your results. **To report**: A discussion of the model and your results. A few possible

things to consider: What can the model tell us about the evolution of cooperation? Are your results independent of initial conditions? Are your results expected viewed in the light of the payoff matrix for a single game? Etc. etc. **(5p)**