

## COMP3212(2023/24): Computational Biology Lab Three - Part 1

Issue	26 February 2024
Deadline	08 March 2024

### Part 1

- Implement the Fisher-Wright model for an infinite population and asexual reproduction, you will need to use an ODE integration library, e.g. `from scipy.integrate import odeint`
- Test it using the same parameters as in the slides (lecture 14 pg 19)
- Now implement the Fisher Wright model for a finite population using asexual reproduction.
- Run it a few times. what do you notice?
- Write a function to run the FW model multiple times and average the results
- Plot a graph after averaging over a number of runs.
- Does averaging over more runs for the finite model make it get closer to the infinite model?
- Now adjust your FW multiple runs model to return the take over time, i.e. when  $x/N > 0.95$ , and the proportion of mutants  $p_x = x/N$ , averaged over a given number of runs.
  - Plot graphs of how the take-over time and proportion of mutants depends on the selection strength  $s$  (assume  $P = 10000$  and  $u = v = 0.0001$ )
  - Plot graphs of how the take-over time and proportion of mutants depends on the mutation rate  $u = v = m$  (assuming  $P = 10000$  and  $s = 0.01$ )
  - Plot graphs of how the take-over time and proportion of mutants depends on the population size  $N$  (assuming  $s = 0.01$  and  $u = v = 0.0001$ )
- Comment on what conditions are necessary for the mutant to take over the population.

### Part 2

- Write a program to compute the transition matrix  $W$  for the Markov model
- Check that all the columns sum to 1
- Write a program to generate the Markov distribution for each time step.
- Calculate the average of this distribution for each time step
- Plot this for  $N = 1000$ ,  $s = 0.01$ ,  $n = m = 0.001$  after  $T = 1000$  steps
- Compare with the plot in the notes to check you have implemented this correctly.
- Use your previous code to generate a histogram of the number of mutants after the same number of time steps using the Fisher Wright model, and plot it with the Markov distribution after differing numbers of time steps (e.g. 100, 500, 1000), using the `density=True` option on `plt.hist` will help the scaling.

## Report and marking

You can ask a demonstrator or myself to mark your work in the lab, or you can submit a pdf output from your work.