Assignment 4

Y3839119

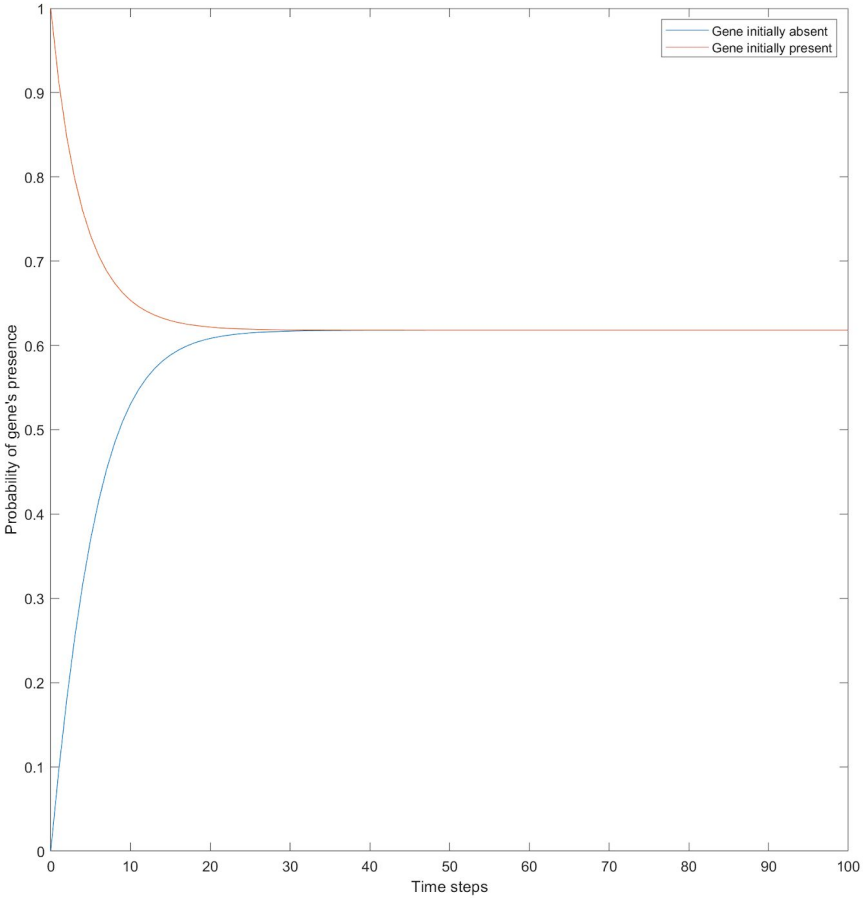
Q1

a) **`tderivs.m`** contains the function for calculating the differential equation for the frequency of a gene in population x within **`gene_dynamics.m`**.

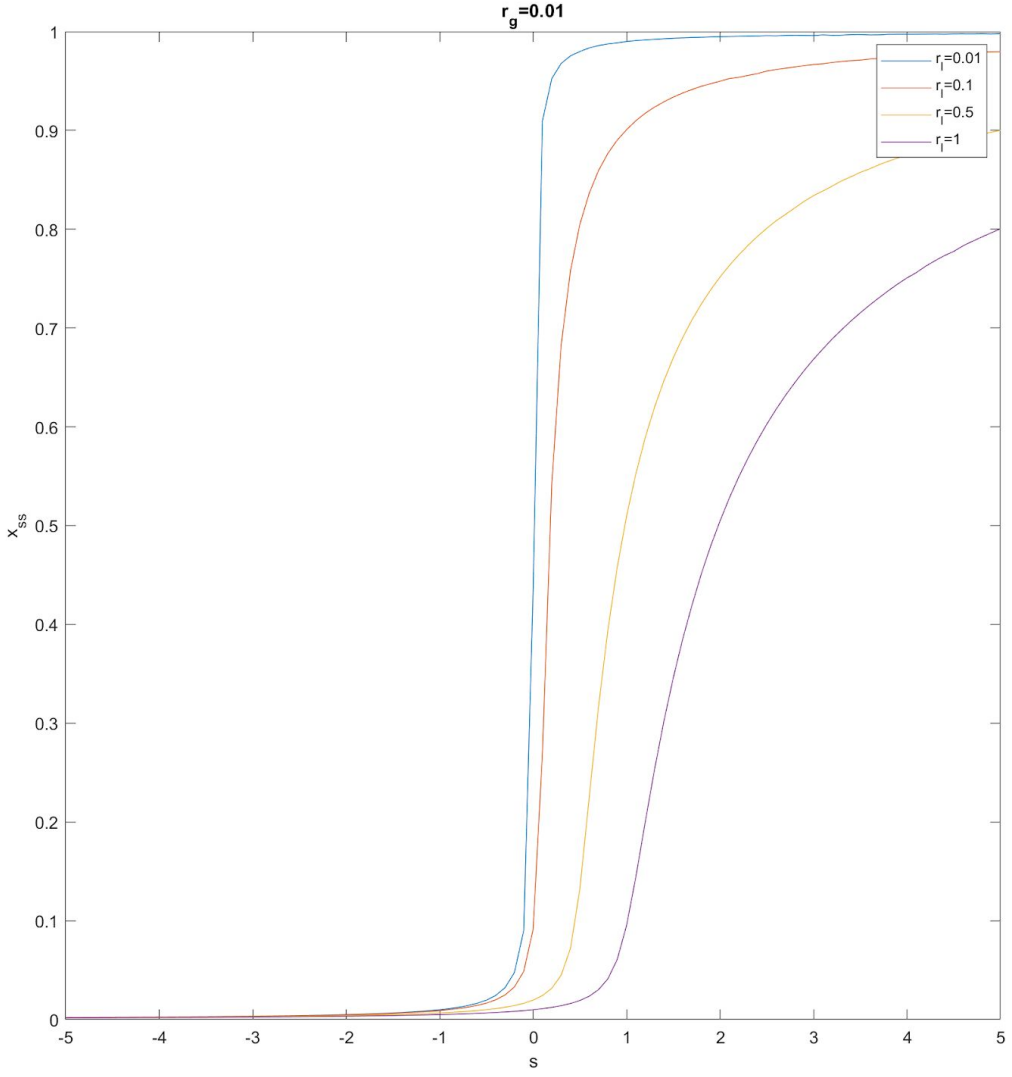
b) I have chosen for $r_l = 0.1,\ r_g = 0.1,$ and $s = 0.1$ as these are used as intermediate values by Domingo-Sananes and McInerney in figure 2 of s*election-based model of prokaryote pangenomes* (2019 preprint). These are set in **`parameters.m`**.

c) Using **`ode45`** with **`tderivs_t.m`,** I obtain 0.6180 as the stable fixed point for the system. That is, 61.8% of the population (61.8% of the single-celled organisms) will possess the beneficial gene, as it slightly improves the fitness/division rate of that organism.
When s is negative, less than half the population possess the gene (38.2% for -0.1, mirroring the result for +0.1 around the halfway point).
If its loss rate is greater than its gain rate, then fewer organisms possess the gene. The opposite is true for when $r_g > r_l$.
`fsolve` produces the same answers as **`ode45`** with function **`tderivs.m`**.

d) Using **`fsolve`** to find the system's Jacobian (for $r_l = 0.1,\ r_g = 0.1,$ and $s = 0.1$), with the command:
`[x,fval,exitflag,output,jacobian] = fsolve(@tderivs,x0)`
returns the value of the Jacobian as -0.2236. Since all (1/1) of the eigenvalues of the Jacobian are real, negative values, the fixed point must be stable.
As can be seen in the convergence plot below, the system converges to its stable state before 40 time steps have taken place, from the minimum and maximum possible values. This means 100 time steps should be plenty for ode45 to converge in future simulations. 100 time steps is enough for $r_l = 0.01,\ r_g = 0.01$ systems to converge as well (but take a bit longer, of course). It takes longer to converge for smaller magnitudes of $r_l,\ r_g,$ and s.



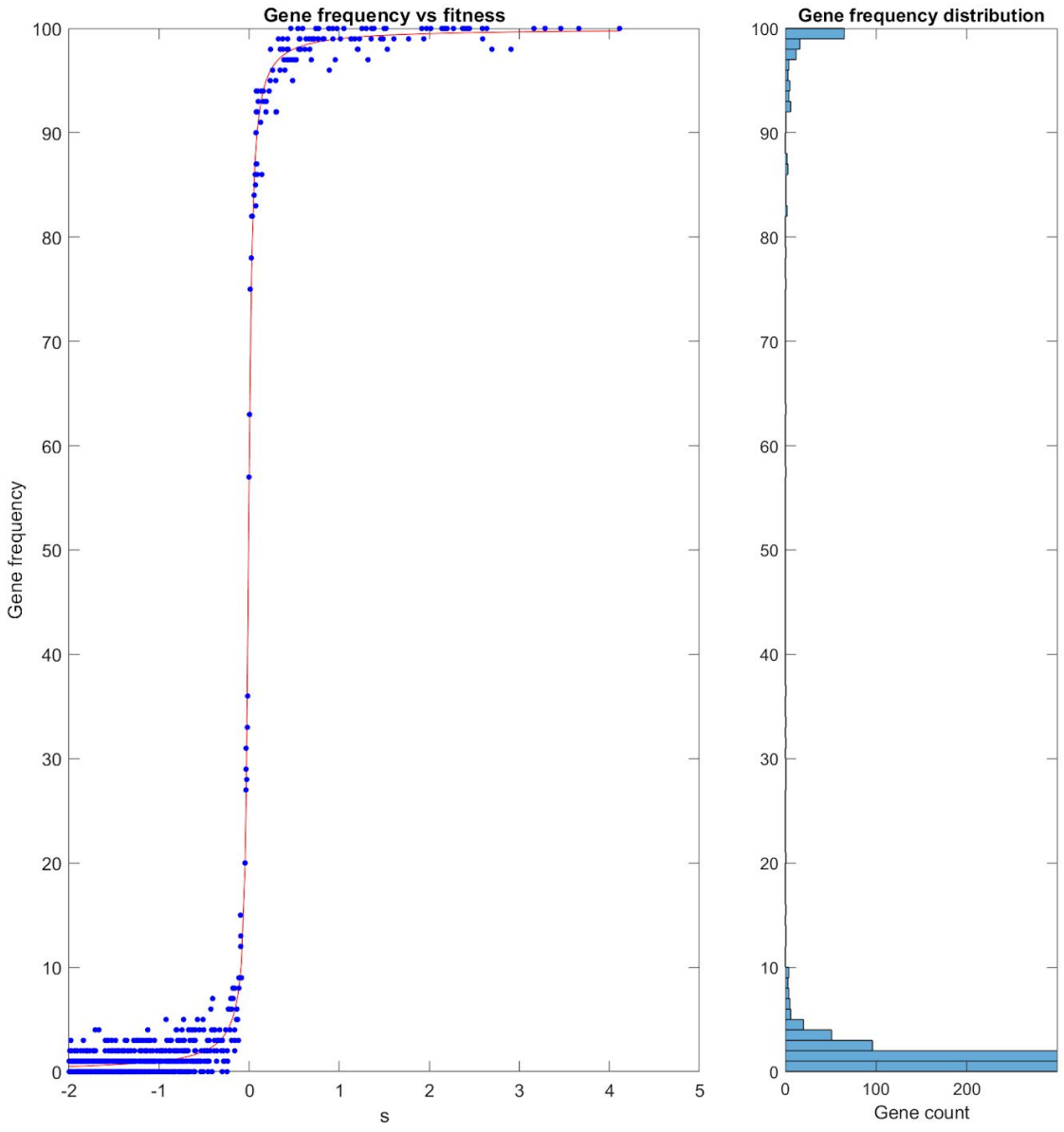
Q2



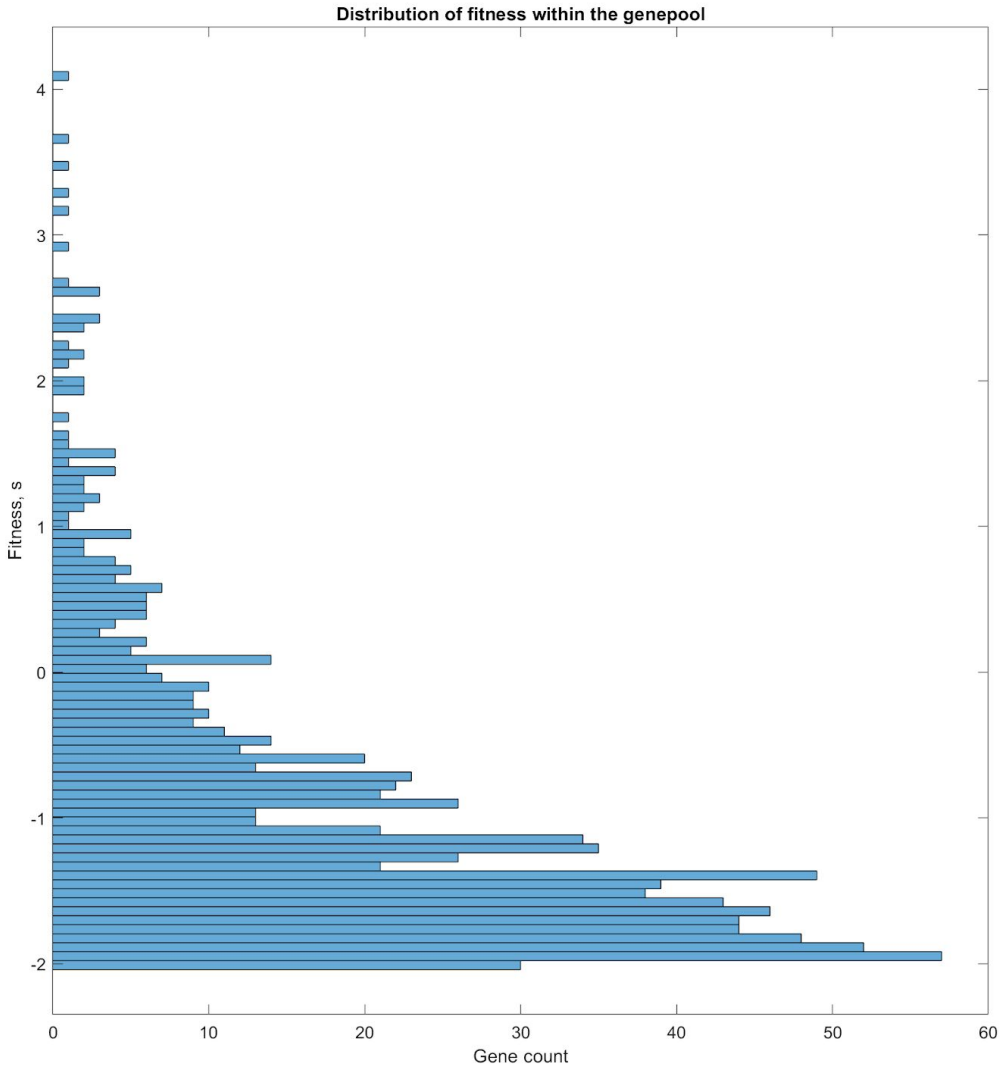
The figure above shows the reproduced plot of figure 2b, as produced by **`fitness_variation.m`**, verifying the results.

Q3

The single gene code has been adapted to describe a genome of a large number of independent genes in **`exponential_fitness_dist.m`**, where random number generation, along with the probability of a gene's presence, determines which genes are present.



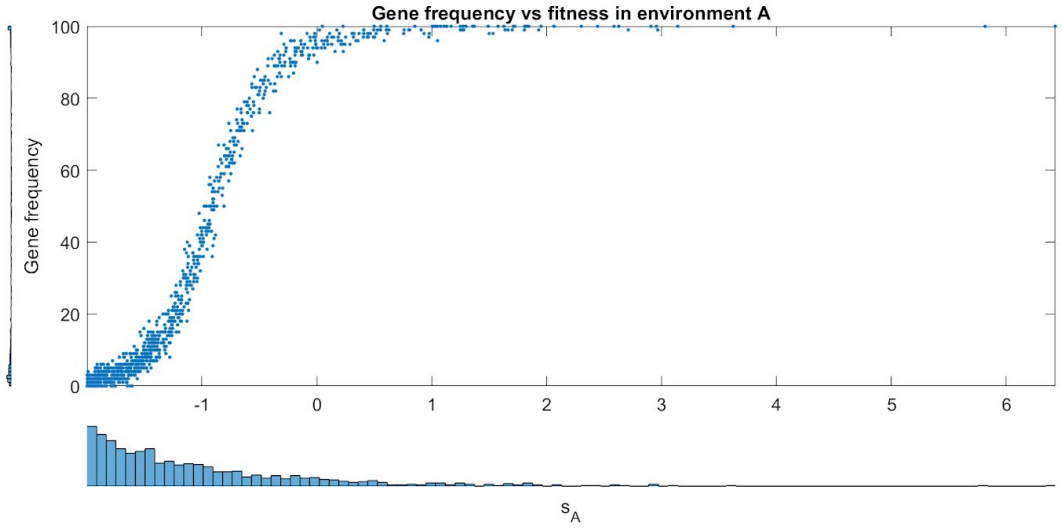
This code reproduced figures 3 III c and d, and supports their results, as seen above.



As can be seen in the plot above, the presence of genes among the population's genomes differs from the distribution of gene fitness within the genepool. Despite the sparse availability of highly beneficial genes in the genepool, the few high s genes are incredibly frequent within the population. Meanwhile, detrimental genes are infrequent despite being highly available within the genepool.

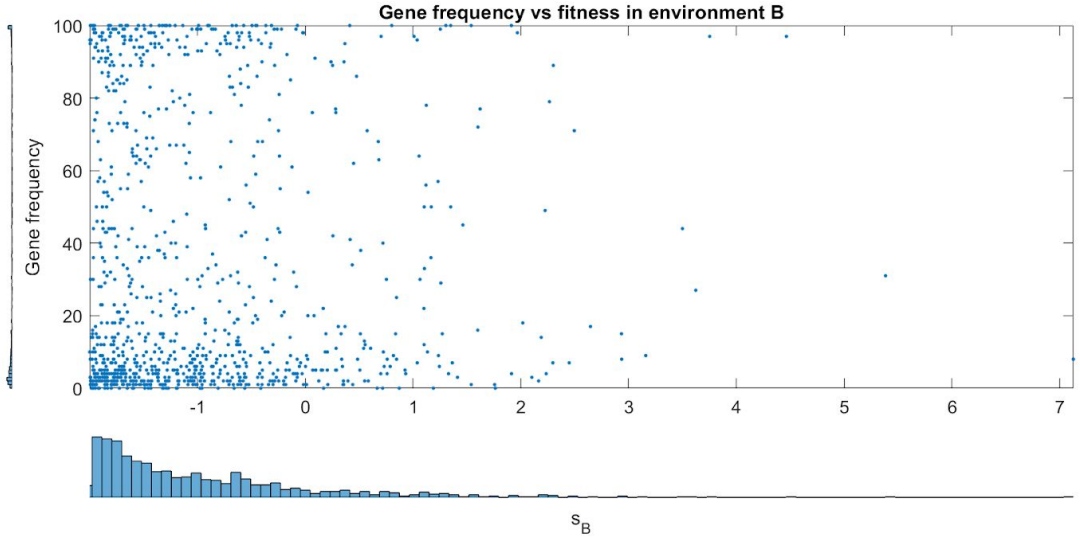
Q4

By changing the fitness of the genes periodically, to simulate changing between the two environments, the distribution of fitness changes in a notable manner. In **`environment_variation.m`**, I have simulated a population moving between environment A and B every 5 time steps ($\tau = 5$) and ending in environment A. This value of τ allows for some uptake and removal of genes with significant fitness effects, but full convergence of the system does not take place.



Environmental fitness becomes less of a determinant of gene presence, with more benign genes being much more likely to be present among the population. This can be seen in the figure above. This is because it takes far longer for relatively innocuous ($s \approx 0$) genes' presence to diminish (as mentioned in q1), and so they persist even if they are slightly detrimental. Likewise, slightly beneficial genes are not always ubiquitous. However, the genes that are highly damaging or beneficial within the current environment are sure to be excluded or included (respectively) among the population's genomes. As τ is increased, gene distribution in environment A becomes increasingly more similar to the results in q3.

Genes that are well-suited for environment B do not necessarily persist, with highly beneficial genes still being absent. Fitness in environment B has little association with gene presence in environment A. This can be seen in the figure below.



Q5

The function **`tderivs_delay.m`** contains the delay differential equation and I analysed its behaviour in the script **`delayed_fitness_effect.m`**, with the parameters set in **`parameters.m`**. The system does not appear to become unstable with delays; the probability of a gene being present is bound between 0 and 1 and the oscillations within this range gradually decay towards the fixed point. When the delay is longer, the oscillations are larger in both magnitude and time period - meaning that convergence takes longer. This makes sense, since it takes longer for the genes' fitness effect to have their effect, but eventually their prevalence among a population will be a function of its fitness effect. The delay cannot destabilise the system, since the system is attracted to a single fixed point, which it will eventually relax to. My findings can be seen in the figure below. Those not yet converged are found to converge with further time-steps.