

Evolutionary Dynamics Exercise 2

Minghang Li

October 10, 2024 (America/New_York)

This solution is organized in R Markdown, and the coding part is written in python via reticulate.

Problem 2: Sequence space and Hamming distance

Consider a binary alphabet $\mathcal{A} = \{0, 1\}$ and a DNA alphabet with $\mathcal{A} = \{A, T, C, G\}$. We are studying sequences $S \in \mathcal{A}^L$ of length L .

(a)

Assume sequences are random with a uniform distribution. What is the average Hamming distance between two random binary sequences? What is the expected Hamming distance for two random DNA sequences?

Hamming distance is defined to be the number of positions at which the corresponding symbols between two sequences are different.

For two random binary sequences, the probability of having a different symbol at a position is $1/2$, and the expected Hamming distance is $L/2$.

For two random DNA sequences, the probability of having a different symbol at a position is $3/4$, and the expected Hamming distance is $3L/4$.

(b)

Instead of a uniform distribution over the alphabet, we assume 0 has a 2 times higher probability of appearing at a position in the sequence than 1. Furthermore, the probabilities $p(A) = p(T)$ are twice the probabilities $p(C) = p(G)$. What is now the average Hamming distance between two random binary sequences? And the expected average Hamming distance for two random DNA sequences?

Binary

Let x, y be two random binary sequences. As 0 has a 2 times higher probability of appearing at a position in the sequence than 1, we have $P(x_i = 0) = 2/3$ and $P(x_i = 1) = 1/3$.

The probability of having a different symbol at a position is

$$\begin{aligned} P(x_i \neq y_i) &= P(x_i = 0, y_i = 1) + P(x_i = 1, y_i = 0) \\ &= P(x_i = 0)P(y_i = 1) + P(x_i = 1)P(y_i = 0) \\ &= \frac{2}{3} \cdot \frac{1}{3} + \frac{1}{3} \cdot \frac{2}{3} \\ &= \frac{4}{9} \end{aligned}$$

The expected Hamming distance is $4L/9$.

DNA

Let x, y be two random DNA sequences. As $p(A) = p(T)$ are twice the probabilities $p(C) = p(G)$, and $p(A) + p(T) + p(C) + p(G) = 1$, we have $p(A) = p(T) = 1/3$ and $p(C) = p(G) = 1/6$.

The probability of having a different symbol at a position is

$$\begin{aligned} P(x_i \neq y_i) &= P(x_i = A, y_i \neq A) + P(x_i = T, y_i \neq T) + P(x_i = C, y_i \neq C) + P(x_i = G, y_i \neq G) \\ &= 2 \cdot \frac{1}{3} \cdot \frac{2}{3} + 2 \cdot \frac{1}{6} \cdot \frac{5}{6} \\ &= 2 \cdot \left(\frac{2}{9} + \frac{5}{36} \right) \\ &= \frac{13}{18} \end{aligned}$$

The expected Hamming distance is $13L/18$.

(c)

Given a binary sequence of length L , how many sequences exist at a Hamming distance 2 from it? How many at distance K with $K \leq L$? Repeat the calculation for DNA sequences.

Let's denote Hamming distance as d_H .

Binary

$d_H = k$ means that there are k out of L positions that are different. At each position with different symbols, we need to choose from $|\mathcal{A}| - 1 = 2 - 1 = 1$ symbol to have a mismatch, and that results in $1^k = 1$ combinations.

$$\begin{cases} d_H = 2 : & \binom{L}{2} \cdot 1^2 \\ d_H = K : & \binom{L}{K} \cdot 1^K \end{cases}$$

DNA

The rationale is similar, except that at each position we now can choose from $|\mathcal{A}| - 1 = 4 - 1 = 3$ symbol to have a mismatch, and that results in 3^k combinations.

$$\begin{cases} d_H = 2 : & \binom{L}{2} \cdot 3^2 \\ d_H = K : & \binom{L}{K} \cdot 3^K \end{cases}$$

Problem 3: Quasispecies

Consider the quasispecies equation with two genotypes 0,1 (i.e., binary sequences of length 1). Let the fitness of genotype 0 be $f_0 > 1$, and the fitness of genotype 1 be $f_1 = 1$. Moreover, genotypes are replicated error-free with probability q . The quasispecies equation is defined as

$$\dot{x}_i = \sum_{j=1}^n \underbrace{x_j}_{\text{frequency of } j} \underbrace{\hat{f}_j}_{\text{fitness of } j} \underbrace{q_{ji}}_{\text{mutation from } i \text{ to } j} - \underbrace{\hat{\phi}}_{\text{average fitness}} x_i, \quad i = 0, \dots, n$$

In vector notation, this can be written as

$$\dot{x} = xW - \phi x$$

where $W = (w_{ij}) = (f_j q_{ji})$ is the mutation-selection matrix and $x = (x_0, \dots, x_n)$.

(a)

Write down the mutation-selection matrix W and find its eigenvalues.

$$\begin{aligned} W &= \begin{bmatrix} w_{00} & w_{01} \\ w_{10} & w_{11} \end{bmatrix} \\ &= \begin{bmatrix} f_0 q_{00} & f_1 q_{10} \\ f_0 q_{01} & f_1 q_{11} \end{bmatrix} \\ &= \begin{bmatrix} f_0 q & (1-q) \\ f_0(1-q) & q \end{bmatrix} \end{aligned}$$

The eigenvalues of W are the solutions to the characteristic equation:

$$\begin{aligned} \det(W - \lambda I) &= 0 \\ \det \left(\begin{bmatrix} f_0 q - \lambda & (1-q) \\ f_0(1-q) & q - \lambda \end{bmatrix} \right) &= 0 \\ (f_0 q - \lambda)(q - \lambda) - f_0(1-q)(1-q) &= 0 \\ f_0 q^2 - \lambda f_0 q - \lambda q + \lambda^2 - (f_0 q^2 - 2f_0 q + f_0) &= 0 \\ \lambda^2 - q(f_0 + 1)\lambda + f_0(2q - 1) &= 0 \end{aligned}$$

The eigenvalues are

$$\lambda_{1,2} = \frac{q(f_0 + 1) \pm \sqrt{q^2(f_0 + 1)^2 - 4f_0(2q - 1)}}{2}$$

(b)

What are the equilibrium points of the population? To which eigenvalue corresponds the non-trivial equilibrium point? *Hint: Perron-Frobenius theorem.*

Let's recall Perron-Frobenius theorem:

Theorem 0.1 (Perron-Frobenius theorem). *Consider a **irreducible non-negative** $n \times n$ matrix M . Matrix M has a positive eigenvalue λ_{max} , such that all other eigenvalues satisfy $|\lambda| < \lambda_{max}$. Furthermore, λ_{max} is simple and the components of the associated eigenvector w are all (strictly) positive, $w_i > 0 \forall i$.*

As we have $f_0 > 1$ and $0 \leq q \leq 1$, the matrix W is non-negative.

To test the irreducibility of $W \in M_n$, we compute:

$$(I + |W|)^{n-1} = I + W > 0 \quad (W \geq 0 \text{ and } n = 2)$$

which proves that W is irreducible. As a conclusion, we can apply Perron-Frobenius theorem.

According to the theorem, the non-trivial equilibrium point corresponds to the largest eigenvalue λ_{max} , in this case is:

$$\lambda_1 = \frac{q(f_0 + 1) + \sqrt{q^2(f_0 + 1)^2 - 4f_0(2q - 1)}}{2}$$

as we have $q \in (0, 1)$ and $f_0 > 1$.

(c)

Examine the dynamics of the quasispecies equation and confirm the results obtained in **(b)**. Assume that $q = 0.7$ and $f_0 = 1.5$, with the initial condition $(0.9, 0.1)$. Modify your simulation to track the average fitness of the population over time. How does the average fitness evolve as the population approaches equilibrium?

```
library(reticulate)
use_condaenv("bio")

import numpy as np
import matplotlib.pyplot as plt

f0 = 1.5
f1 = 1
_q = 0.7
q = np.array([[_q, 1-_q], [1-_q, _q]])
ini = np.array([0.9, 0.1])

W = np.array(
    [[f0*q[0, 0], f1*q[1, 0]],
     [f0*q[0, 1], f1*q[1, 1]]
)

eigvals, eigvecs = np.linalg.eig(W)

x_star = eigvecs[:, 0] / np.sum(eigvecs[:, 0])
phi = np.sum(x_star * np.array([f0, f1]))
ivp = np.linalg.solve(eigvecs, ini)

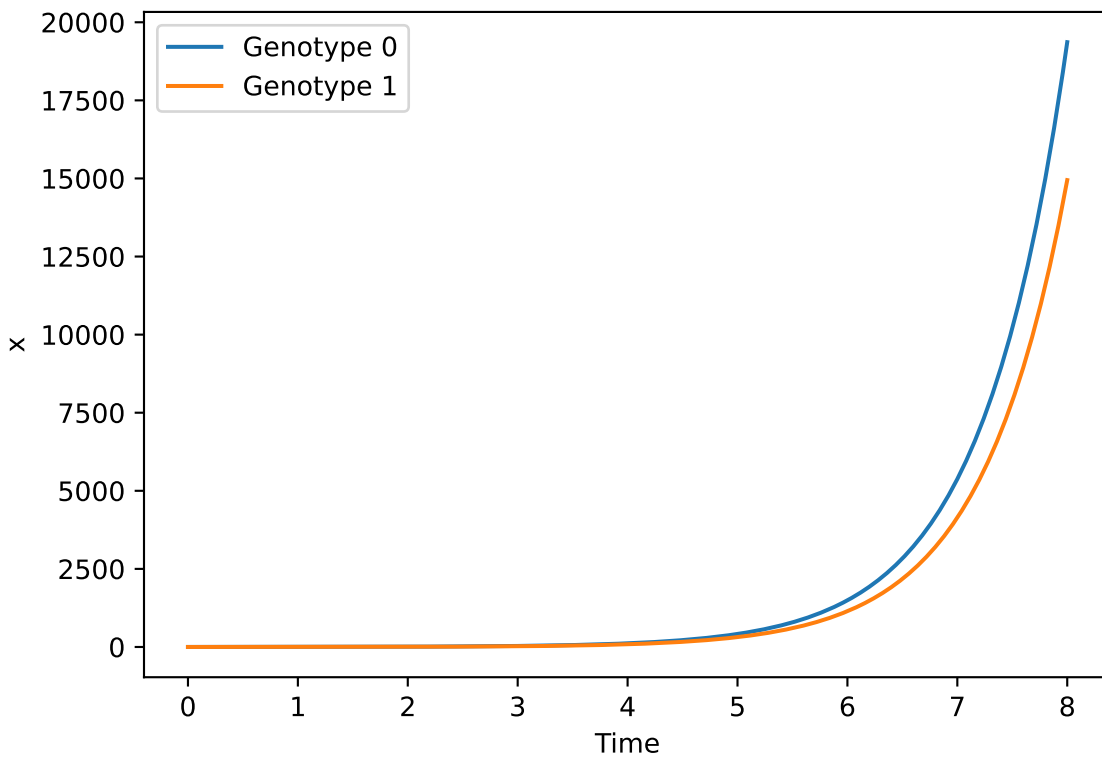
time = np.linspace(0, 8, 201)

X_0 = ivp[0] * eigvecs[0, 0] * np.exp(eigvals[0] * time) + \
      ivp[1] * eigvecs[0, 1] * np.exp(eigvals[1] * time)

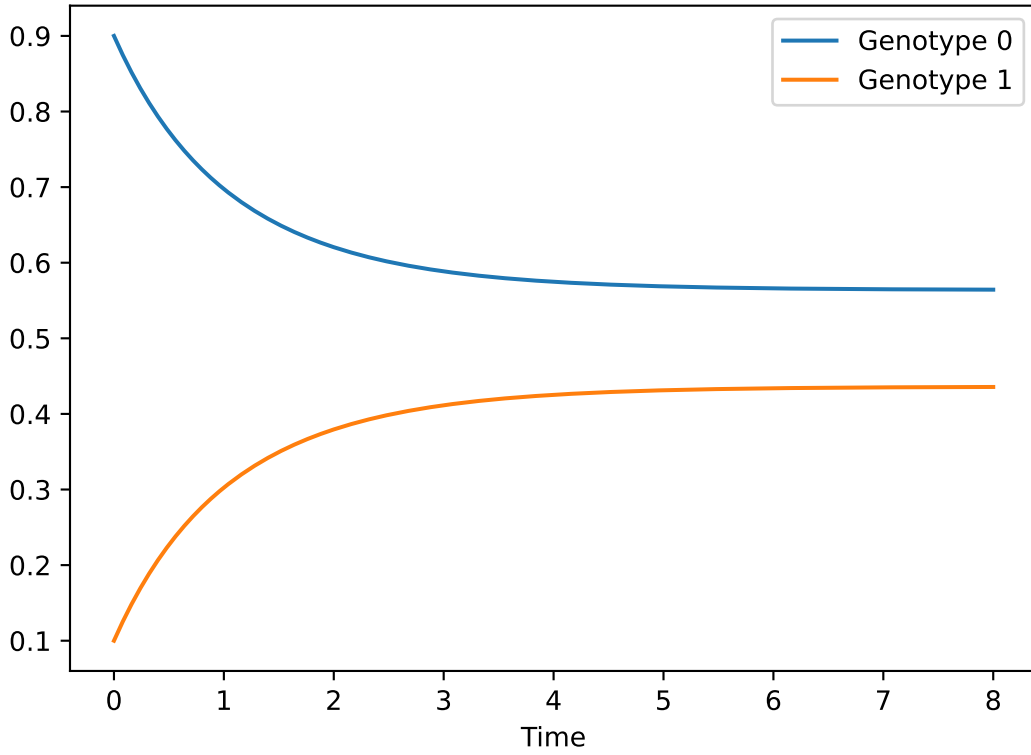
X_1 = ivp[0] * eigvecs[1, 0] * np.exp(eigvals[0] * time) + \
      ivp[1] * eigvecs[1, 1] * np.exp(eigvals[1] * time)

plt.plot(time, X_0, label='Genotype 0')
plt.plot(time, X_1, label='Genotype 1')
plt.xlabel('Time')
```

```
plt.ylabel('x')
plt.legend()
plt.show()
```



```
plt.plot(time, X_0 / (X_0 + X_1), label='Genotype 0')
plt.plot(time, X_1 / (X_0 + X_1), label='Genotype 1')
plt.xlabel('Time')
plt.legend()
plt.show()
```



Plug $q = 0.7$, $f_0 = 1.5$ into the formula, we have

$$\lambda_{1,2} = \frac{0.7 \cdot 2.5 \pm \sqrt{0.7^2 \cdot 2.5^2 - 4 \cdot 1.5 \cdot 0.4}}{2} = \frac{1.75 \pm \sqrt{0.6625}}{2}$$

We have

$$\lambda_1 \approx 1.282, \quad \lambda_2 \approx 0.468$$

And then we have the equilibrium point x^* satisfies:

$$\begin{aligned} (W - \lambda_1 I)x^* &= 0 \\ \begin{bmatrix} 1.5 \cdot 0.7 - 1.28 & 0.3 \\ 1.5 \cdot 0.3 & 0.7 - 1.28 \end{bmatrix} x^* &= 0 \\ \begin{bmatrix} -0.232 & 0.3 \\ 0.45 & -0.582 \end{bmatrix} x^* &= 0 \end{aligned}$$

Solving the system of equations, we have

$$x^* \approx \begin{bmatrix} 0.564 \\ 0.436 \end{bmatrix}$$

which exactly confirms with the solution in the numerical simulation:

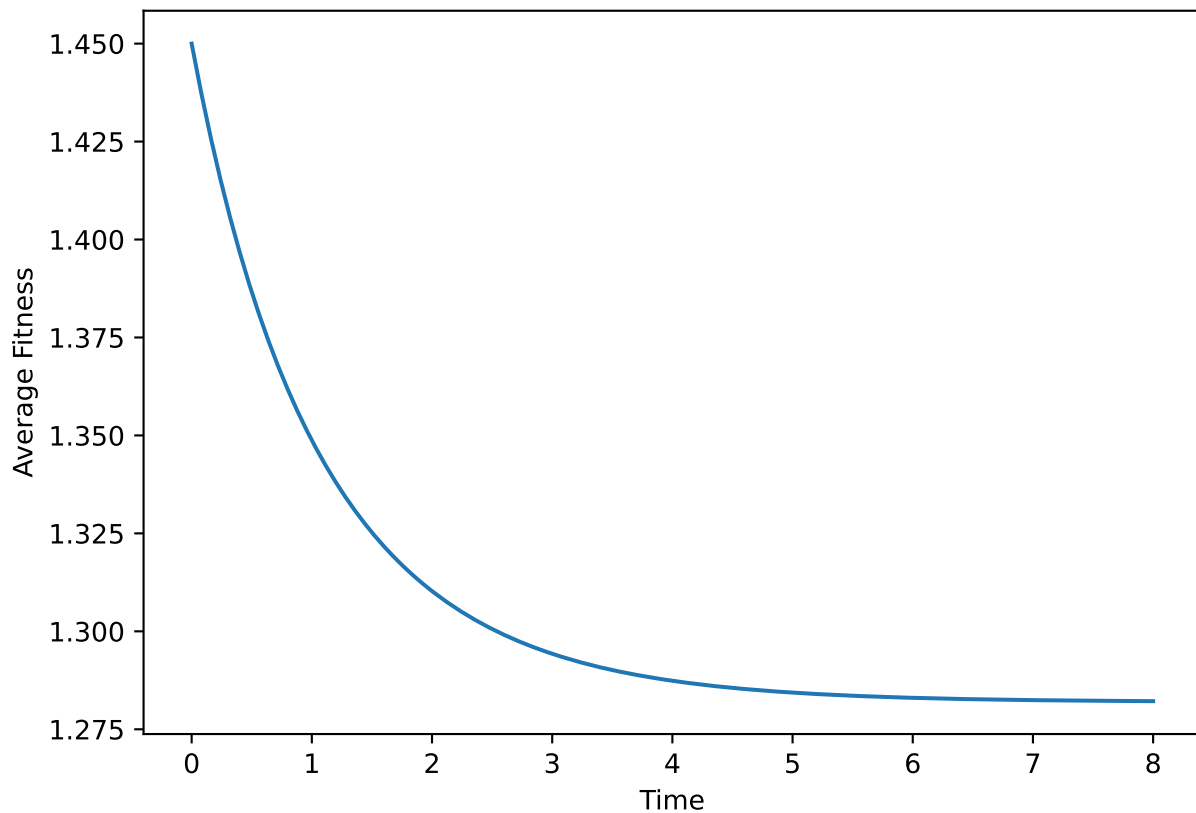
```

x_star

## array([0.56394103, 0.43605897])
phi_t = np.sum([X_0 * f0, X_1 * f1], axis=0) / (X_0 + X_1)

# figsize 10x5 inches
plt.plot(time, phi_t)
plt.xlabel('Time')
plt.ylabel('Average Fitness')
# tight layout
plt.tight_layout()
plt.show()

```



It confirms with the lecture that in general, the equilibrium x^* does not maximize the fitness ϕ . Mutation reduces population fitness.

(d)

What is the equilibrium point for $f_0 = f_1 = 1$?

If $f_0 = f_1 = 1$, then the matrix W becomes

$$W = \begin{bmatrix} q & (1-q) \\ (1-q) & q \end{bmatrix}$$

The solution to the characteristic equation:

$$\lambda_{1,2} = \frac{2q \pm \sqrt{4q^2 - 8q + 4}}{2} = q \pm \sqrt{(q-1)^2}$$

which means

$$\lambda_1 = 2q - 1, \quad \lambda_2 = 1$$

Since $q \leq 1$, $\lambda_{max} = \lambda_2 = 1$.

Plugging back to the equation, we have

$$W - I = \begin{bmatrix} q-1 & 1-q \\ 1-q & q-1 \end{bmatrix} x^* = 0$$

which yields $x^* = (0.5, 0.5)$.

(e)

In the lecture you heard about a concept called the error threshold. Assume for this subtask a wildtype genotype with fitness $f_{wt} > 1$ and all other genotypes have fitness $f_m = 1$. For a large sequence length, you can assume, that once a sequence is mutated, back-mutations are negligibly unlikely. Can you derive a condition based on q and f_{wt} for this error threshold? What happens when the threshold is crossed?

Let's denote wt as 0 for the sake of saving my fingers (after typing half of the first equation, the author complained).

The quasispecies equation is

$$\dot{x} = xW - \phi x$$

where

$$W = \begin{bmatrix} f_0 q & 0 \\ f_0(1-q) & 1 \end{bmatrix}$$

The eigenvalues of W are

$$\lambda_1 = 1, \quad \lambda_2 = f_0 q$$

If $f_0 q < 1$, then the equilibrium should satisfy:

$$W - I = \begin{bmatrix} f_0 q - 1 & 0 \\ f_0(1-q) & 0 \end{bmatrix} \begin{bmatrix} x_0^* \\ x_1^* \end{bmatrix} = 0 \Rightarrow x_0^* = 0, x_1^* = 1$$

If $f_0 q > 1$, then the equilibrium should satisfy:

$$W - f_0 q I = \begin{bmatrix} 0 & 0 \\ f_0(1-q) & 1 - f_0 q \end{bmatrix} \begin{bmatrix} x_0^* \\ x_1^* \end{bmatrix} = 0 \Rightarrow x_0^* = \frac{f_0 q - 1}{f_0 - 1}, x_1^* = 1 - x_0^*$$

The population ideally wants to populate wild type as much as possible as it has larger fitness, but it can only do so if $f_0 q > 1$ – I assume this is the error threshold. As elucidated before, if the threshold is crossed, the population will be dominated by the low fitness mutant genotype and likely eventually die out.