Modelling Biological Systems Exam 2016

Sergii Gladchuk

December 31, 2016

4. The spread of a gene.

The relative frequency p of a gene with fitness w in a population of size N has the approximate dynamics

$$p_{t+1} = \frac{w}{\overline{w}_t} p_t + \epsilon_t$$

where $\overline{w}_t$ is the population mean fitness at time t according to

$$\overline{w}_t = wp_t + 1 - p_t$$

and the random deviates ϵ_t are drawn from a Normal distribution with mean zero and the standard deviation

$$\sigma_t = \frac{1}{\overline{w}_t} \sqrt{\frac{w}{2N} p_t (1 - p_t)}$$

It is assumed that the rest of the genes in the population have fitness equal to 1. The focal gene is thus at a fitness advantage if $w > 1$.

a) Equilibria ignoring the stochasticity

Ignoring the stochasticity (setting $\epsilon_t = 0$), show that $p^* = 0$ and $p^* = 1$ are the only equilibria of the system (assuming $w \neq 1$).

This can be done by solving $p^* = f(p^*)$ in our case:

$$p = \frac{wp}{wp + 1 - p}$$

So first obvious solution is $p = 0$, the other solution:

$$1 = \frac{w}{wp + 1 - p}$$

$$p(w - 1) + 1 - w = 0$$

$$p = \frac{w - 1}{w - 1}$$

$$p = 1$$

b) Stability criteria

What are the stability criteria for $p^* = 0$?

Equilibrium point of a difference equation is stable if the slope of linear approximation is smaller than 1 and bigger then -1.

So slope of linear approximation for $p^* = 0$:

$$\frac{d}{dp}\left(\frac{wp}{wp+1-p}\right)\Bigg|_{p=0} = \frac{d}{dp}\left(\frac{wp}{p(w-1)+1}\right)\Bigg|_{p=0} = \left(\frac{f'(wp)(p(w-1)+1) - wpf'(p(w-1)+1)}{(p(w-1)+1)^2}\right)\Bigg|_{p=0} =$$

$$= \left(\frac{wp(w-1)+w-wp(w-1)}{(p(w-1)+1)^2}\right)\Bigg|_{p=0} = \left(\frac{w}{(pw-p+1)^2}\right)\Bigg|_{p=0} = w$$

So slope $= w$ and in order for equilibrium at point 0 to be stable: $w < 1$

c) Script with stochasticity

Now taking the stochasticity into account again, write an R script that runs a simulation of p_t for 1000 generations and plots the resulting time series. The starting p-value should be $1/(2N)$, corresponding to a single mutant allele of a diploid individual.

***Tip:** Break the simulation as soon as p_t is equal to or below zero. Do the same thing as soon as p_t is at or above 1. The break command may be useful.*

Since this script includes stochasticity, plot will be different each time it runs, and number of generations it will take for gene to disappear fully (frequency = 0) or be present in all population (frequency = 1) will also differ.

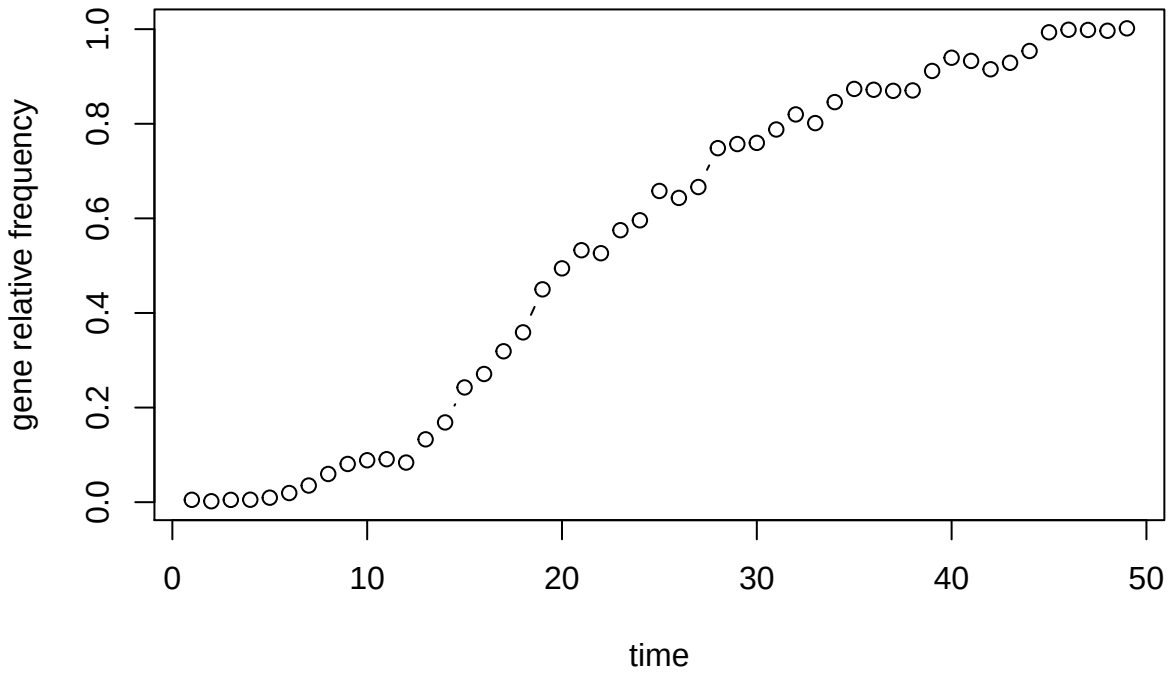
```
# parameters
w <- 1.2 #fitness
N <- 100; #population size

p <- 1/(2 * N); # starting p-val

for (t in 2:1000) {

  w.mean <- w * p[t-1] + 1 - p[t-1];
  sigma <- 1/w.mean * sqrt(w/(2*N)*p[t-1]*(1 - p[t-1]));
  p <- c(p, 0);
  #dynamix function
  p[t] <- w / (w.mean) * p[t-1] + rnorm(1,mean=0,sd = sigma);

  if(p[t] <= 0 || p[t] >= 1) {
    break;
  }
}
plot(p, type='b', xlab='time', ylab='gene relative frequency')
```



d) Function for fixation probability

Write a function that takes N and w as input parameters, runs 1000 simulations like the one above and returns the probability of fixation, i.e. the probability that p reaches 1 within 1000 generations.

Here is modified version of above scrip to run 1000 simulations and output frequency

```
q4_d_function <- function(N, w){
  simNum <- 1000
  simulations <-rep(0,simNum)

  p <- 1/(2 * N); # starting p-val
  for (s in 1:simNum) {
    for (t in 2:1000) {

      w.mean <- w * p[t-1] + 1 - p[t-1];
      sigma <- 1/w.mean * sqrt(w/(2*N)*p[t-1]*(1 - p[t-1]));
      p <- c(p, 0);
      #dynamix function
      p[t] <- w / (w.mean) * p[t-1] + rnorm(1,mean=0,sd = sigma);

      if(p[t] <= 0) {
        simulations[s] <- 0;
        break;
      }
```

```
      if(p[t] >= 1) {
        simulations[s] <- 1;
        break;
      }
    }
  }
  # output of fixation probability
  return(sum(simulations)/simNum)
}

q4_d_function(N=100,w=1.2)
```

```
## [1] 0.406
```

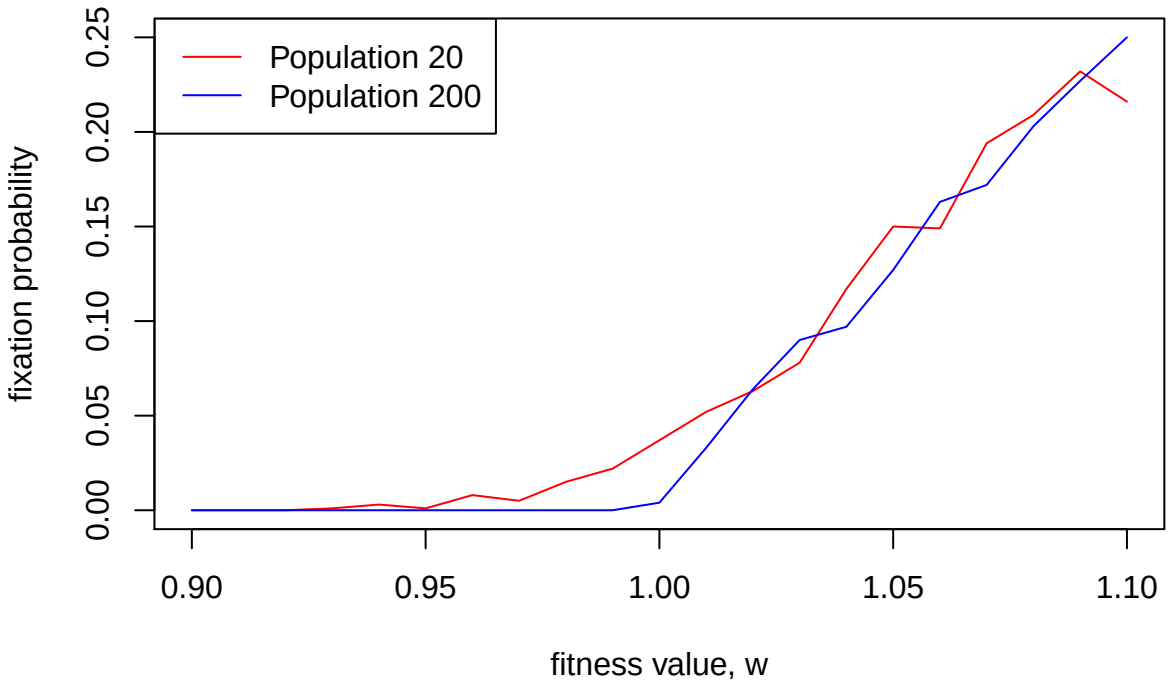
c) Script with function in d)

Write a script that uses the function in d) and plots the probability of fixation for a range of w-values from 0.9 to 1.1. Do the plot for $N = 20$ and $N = 200$. (You should be able to see that small populations are more likely to accumulate deleterious mutations than large ones.)

Final script:

```
w.vector <- seq(0.9,1.1,by=0.01)
source('~/Documents/courses/modelling/exam/q4_d_function.R')
fixationN20 <- rep(1,length(w.vector));
fixationN200 <- rep(1,length(w.vector));
for (i in 1:length(w.vector)){
  fixationN20[i] <- q4_d_function(20,w.vector[i]);
  fixationN200[i] <- q4_d_function(200,w.vector[i]);
}

#ploting
plot(fixationN20~w.vector, type='l', xlab='fitness value, w', ylab='fixation probability',
     ylim=c(0,max(fixationN20,fixationN200)), col = 'red')
lines(fixationN200~w.vector, col='blue')
legend('topleft',legend=c('Population 20','Population 200'),
       col = c('red','blue'), lty=c(1,1))
```



Indeed, small population (red line) - has higher fixation probability with fitness lower than 1 (deleterious/bad genes).