Genetic Variation

Term Definitions

Continuous time: Time that flows continuously without occurring in units.

Diffusion: A passive net movement of particles from a region of high concentration to (a) region(s) of lower concentration.

Discrete time: Time that is broken into units, such as generations.

Genetic drift: Change in allele frequency by evolutionary sampling error.

Factorial: The product of all positive integers less than or equal to a starting positive interger and represented by an ! (for example, $4! = 4 \times 3 \times 2 \times 1 = 24$).

Transition probability: A probability of changing from one state to another (or remaining in the same state). These are usually organized into a matrix of all possible states.

6.1 Genetic drift and evolutionary sampling

A key idea upon which much of computational population genetics rests is that the vast majority of expected genetic variation observed in populations will be due to seemingly random changes and not because of direct evolutionary selection. The basic idea is simple enough: Organisms sometimes die or don't reproduce by random chance, while other organisms may produce more offspring than average by chance, and that affects what

Population Genetics with R: An Introduction for Life Scientists. Áki J. Láruson and Floyd A. Reed, Oxford University Press (2021).
DOI: 10.1093/oso/9780198829539.003.0006

genetic variants pass down through generations. One of the founders of the Population Genetics discipline, Sewall Wright, referred to this back in 1929 as allele frequencies appearing to "drift" from one frequency to another over the generations, without a steady direction toward becoming fixed or extinct in the population. The term "genetic drift" has stuck. The idea that neutral variants, and not variants explicitly driven by natural selection, made up the majority of the genetic variation observed was first formalized by Motoo Kimura in the sixties and remains a key consideration for statistical analyses of observed genetic variation in populations. The significance of the theory of neutral variation is that it provides a succinct expectation for observed genetic variation, and therefore provides a concise null hypothesis against which to test alternative hypotheses. Much like in Chapter 4, when we made a series of assumptions to predict the interactions between allele and genotype frequencies, in order to quantitatively approach the random changes in allele frequencies over time there are generally a lot of assumptions that have to be made regarding several factors at play. For example, how exactly do variants get passed down through the generations? What about the occurrence of new mutations and migration bringing in new variants? There are just about an infinite amount of considerations that could be made, but in order to make this manageable we're going to assume just a few simple facts:

1. Generations of organisms do not overlap (one generation dies as the next emerges).
2. Organisms are diploid (each individual has two copies of their genome).
3. There's a fixed population size, (N), which does not change across generations.
4. Each allele is "drawn" at random, independent of all other alleles, from the previous generation to pass on to the next.

These assumptions are what make up the Wright–Fisher model of genetic drift. Using the assumptions inherent to this model, we can attempt to

predict patterns we might observe of neutrally changing variants being inherited across multiple generations. The last stipulation of the Wright–Fisher model above is that any one allele is not more likely to be drawn than any other allele. This is what defines neutrality—there is no positive or negative selection occurring. This may seem overly constrained, as we know fitness differences exist in some cases, but keep in mind that drift *always* exists and affects alleles under selection as well as neutral variants, so drift is critically important to understand. One direct computational way to approach the expected change in allele composition occurring every generation is to think of inheritance primarily in terms of probabilities.

It's generally assumed when discussing probabilities that there is an underlying distribution for how probable or improbable some event is. Using the right distribution of probabilities across different observations (quantiles) is critical. In the previous chapter we were working with the χ^2 distribution to compare observations across specific categories. Here we are going to use a binomial distribution to work with collections of binary outcomes. A binary event is an event that either occurred or didn't occur—like when flipping a coin, it's either heads or tails—and it can be quite useful to think of alleles in this manner.

Let's say we have a very small population of five diploid individuals made up of two alleles—one allele is present as three copies (black circles) and the other is present as seven copies (white circles). If we randomly draw circles to make up the next generation, the chance of drawing a black allele is 3/10 and the chance of a white allele is 7/10.

Assuming all copies have an equal chance of reproducing, what is the chance that the next generation, kept at the same size of ten, will be exactly the same as before, with just three copies of the black allele and seven copies of the white allele? This result depends on the black allele being sampled three times and the white allele being sampled seven times.

```
> 0.3^3 * 0.7^7
[1] 0.002223566
```

That's a pretty small chance: about 0.2%. However, this equation is incomplete because there are actually a lot of different ways to end up with a pattern of three black and seven white alleles. The first three could be black and the last seven white, or the first, third, and fifth black and the rest white, etc. The total number of combinations can be calculated using factorials:

$$\binom{n = 10}{k = 3} = \frac{n!}{k!(n-k)!} = \frac{10!}{3!7!} = 120.$$

So there are a total of 120 ways that we could possibly get three black and seven white alleles from sampling the original population. This operation can be referred to as "ten choose three," because we've calculated all the ways to choose three "successes" out of ten "attempts." We can calculate this in R using the `choose` function:

```
> choose(10, 3)
[1] 120
```

So now our full equation for the probability of exactly three black and seven white alleles showing up in the next generation becomes

$$P(3 \text{ and } 7) = 120 \times 0.3^3 \times 0.7^7 \approx 0.267$$

```
> 120 * 0.3^3 * 0.7^7
[1] 0.2668279
```

That suggests there's a much higher chance than we originally calculated (from 0.02% to over 26%) of keeping the same proportions from one generation to the next. What we are really talking about here is the binomial probability

$$P(\text{k out of n}) = \binom{n}{k} p^k (1-p)^{(n-k)},$$

where n is the total sample size (10 in our example), k is the number of observed variants we're interested in (3), and p is the frequency of that type (0.3).

So in this small population there is about a one-in-four chance that the allele frequency stays the same in the next generation. Turning this around, there is about a three-in-four chance that it *will* change in the next generation. In statistics we expect that when we take a sample, the frequency in the sample is often not exactly the same as in the original population we are drawing from. However, the larger the sample, the smaller we expect the deviation to be by chance. Genetic drift is essentially evolutionary sampling error, and the deviations between generations are greater in smaller populations (that is, smaller samples).

We can also readily calculate the extreme cases: that the next generation will be made up of all black or all white alleles. Intuitively we can predict that the probability of sampling all white alleles is larger than all black, because of the frequency difference. The probability of getting all white alleles in the next generation is $0.7^{10} \approx 0.028$. Note that there is only one way to get this result: all copies have to be white, so we don't have to worry about the n choose k binomial coefficient. The probability of all black alleles by random chance is $0.3^{10} \approx 5.9 \times 10^{-6}$. So from our example starting point we don't expect complete fixation or loss of one allele or the other in a single generation, but more likely than not it will change in frequency due to no forces other than random sampling.

Let's consider a single locus in a diploid organism where a mutation has occurred, giving us the possibility of two alleles existing there: A and a. We can compose a matrix containing all the possible genotypes that are possible with just two alleles. Remember that we are assuming this organism is diploid (again this is an often-used assumption, but there are plenty of exceptions), so we then expect there to be a grand total of three distinct genotypes possible: AA, Aa, and aa.

The first thing we're going to do is save the number of individuals we're dealing with as an object (`N`), and then we'll figure out how many copies of the allele A are possible, given our number of individuals. The number

of possible copies should just be a vector ranging from zero (the allele is extinct) to two times (remember, diploid) the number of individuals (2*N), at which point the allele is fixed in the population:

```
> N <- 1 #One diploid individual
> possible <- 0:(2*N) #Number of possible copies of an allele
```

We can now draw the binomial probability (P) of having a particular copy number (k), given a certain allele frequency (p) and the total copy numbers possible (n):

$$P(k|p,n) = \binom{n}{k} p^k (1-p)^{n-k}.$$

So if we start with one copy of A we have the following transition probabilities in the next generation:

Copies Next Generation	0	1	2
Binomial Probability	1/4	1/2	1/4

So the most likely outcome (50%) is that we'll stay at one copy of A, but we have a 25% chance of either losing or gaining a copy of A in the next generation.

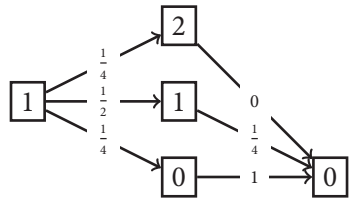
We're setting up our framework here looking at a small test case (one autogamic individual!) so that we can easily expand our code later to handle the rapidly increasing complexity that comes with adding more individuals. Let's use the R function `dbinom` to get our binomial probabilities for each of our possible allele counts, given all possible starting frequencies. We'll save these binomial probabilities as object `P` and then create a matrix containing these values that we'll call Q:

```
> P <- NULL #Vector to hold our probabilities
> for(i in possible){
    P <- c(P,dbinom(possible, size=2*N, prob=i/(2*N)))
}
> (Q <- matrix(P, ncol=2*N+1, byrow=T)) #Arrange into matrix
     [,1] [,2] [,3]
```

```
[1,] 1.00  0.0 0.00
[2,] 0.25  0.5 0.25
[3,] 0.00  0.0 1.00
```

We now have a transition probability matrix that describes what to expect, given all possible starting states. Each row shows us the probability of transitioning from zero, one, or two copies of *A* (going down the matrix), to zero, one, or two copies, respectively across the columns, in a subsequent generation. Notice that in the top row, where we're assuming that we have zero copies of *A*, we're seeing a 100% probability of staying at zero copies. And similarly, in the bottom row, we see a 100% chance of staying at two copies of *A* in the next generation. These should make sense, thinking about extinction and fixation within the confines of our model. We're assuming that no new mutations are arising that disrupt our fixed state, so once the *A* allele has saturated the population it'll never be anything but fixed. Similarly, if the *A* goes extinct it can never again arise, and will stay at zero forever. This is not very realistic, since we know that new mutations arise and even lost mutations can potentially re-emerge, but once again we're sacrificing immobilizing realism for functional reductionism.

If we assume that every generation these transition probabilities hold, then the only thing changing generation to generation is the starting frequency. So across the generations the probability of our allele *A* ending at a certain copy number is the composite probability of the transition seen in each previous generation. The probability of *A* being lost in three generations could be visualized like so:



Because we assume that what happens in each generation is independent of what happens in any other generation, we can multiply the probabilities

along any one path to get the cumulative probability of the allele following that particular path (probability multiplication rule). So the probability that the *A* allele stayed as one copy and then went extinct is just

$$\frac{1}{2} * \frac{1}{4} = \frac{1}{8} = 12.5\%.$$

But what if we want the total probability of *A* going extinct in three generations, regardless of what path it went through? Because each hypothetical path is mutually exclusive of all other paths (that is, there's no way for an allele to simultaneously experience more than one path in a single generation), we can add up all the cumulative probabilities of each path (the probability addition rule):

$$\left(\frac{1}{4} * 0\right) + \left(\frac{1}{2} * \frac{1}{4}\right) + \left(\frac{1}{4} * 1\right) = \frac{3}{8} = 37.5\%.$$

Next let's create a matrix object that gives us a starting state. In order to set it up like our transition matrix, we're going to have each column represent one of our three possible transition states (zero, one, and two copies), but we're only going to need one row in this matrix. Let's first generate the matrix object `x` with all probabilities set to zero, which we can easily update later:

```
> (x <- matrix(c(rep(0,2*N+1)), ncol=2*N+1, byrow=T))
     [,1] [,2] [,3]
[1,]    0    0    0
```

Now let's assume we're starting with a single copy of *A*, which means we'll set the probability in the "one copy" column (the second column) to 100%:

```
> x[,2] <- 1
> x
     [,1] [,2] [,3]
[1,]    0    1    0
```

We can now use matrix multiplication to get all our transition probabilities, given our starting state. This should give us the exact same probabilities we worked out earlier for a single generation transition. Refer back to Chapter 3 if you need a refresher on matrix multiplication in R.

```
> (R <- x%*%Q)
     [,1] [,2] [,3]
[1,] 0.25  0.5 0.25
```

Now let's visualize our transition probabilities over multiple generations. Let's start with assigning some color and shape variables that we'll be using consistently, set our generation variable `g` to start at one for each of our transition states, and then plot our first-generation transition probabilities with an accompanying legend. We're going to use a couple of neat features of the `legend` function by specifying our placement with `"bottom-left."` This would normally plop our legend down in the expected bottom left-hand corner of our plot, but we're going to set `xpd` to TRUE and specify an `inset` command, which should place our legend on the *top* left side of our plot. Effectively what we're doing is allowing the legend to have its own location parameters outside of the plot provided (`xpd = TRUE`), and saying that we want to move it up the y-axis by the full length of our figure margin (`inset=c(0,1)`). Finally, we'll specify that we don't want a box around our legend by specifying `bty="n."`

```
> color <- c("brown","blue","grey")

> shape <- c(15,19,17)

> #Start with generation 1 for all states
> g <- rep(1,ncol(R))

> plot(points(x=NULL, xlim=c(1,10), ylim=c(0,1),
    ylab="Probability",
    xlab="Generations")
```

```
> legend("bottomleft",
    legend=c("Extinct","One copy","Fixed"),
    col=color, pch=shape,
    xpd=TRUE, inset=c(0,1), bty="n")
```

Now that we've set up our canvas, let's fill in our transition probabilities by creating a short `while` loop, where each loop updates the probabilities and adds one to our generation time. Let's also set our exit condition to be when `g` reaches ten generations. Note that since we can't set a condition based on multiple elements in a vector, we can set the condition explicitly so that as long as the first element of `g` is less than 10 (`g[1] < 10`) our loop will continue to run:

```
> while(g[1]<=10){
    (R <- R%*%Q)
    g <- g+1
    points(g, R, col=color, pch=shape)
}
```

Taking a look at our output from the above code, it should look similar to Fig. 6.1. We can see that the probability of *A* staying as a single copy drops pretty quickly from 0.5 to practically zero within just a handful of generations. Notice that the "Fixed" and "Extinct" conditions (which perfectly overlap) pretty quickly rise to the same probability as the starting frequency of the allele. Remember that little detail; we'll revisit that a bit later.

Try changing the starting condition by manipulating `x` so that we start off with *A* being fixed in the population,

$$x = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix},$$

or with *A* being extinct:

$$x = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}.$$

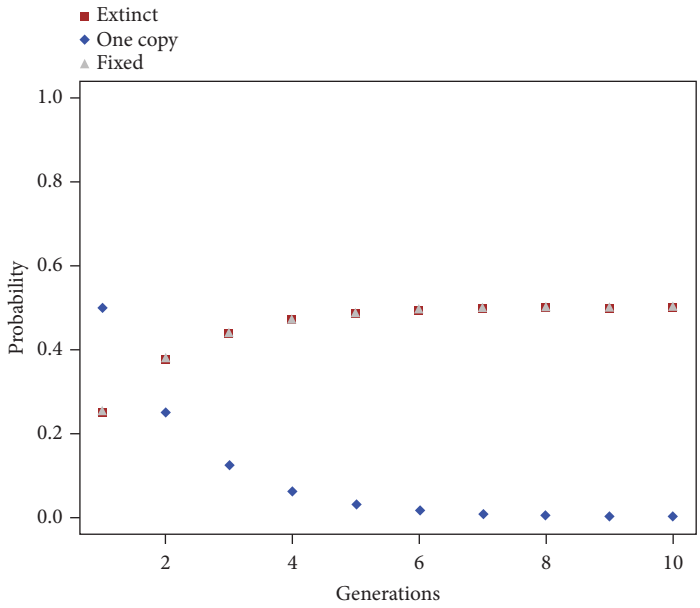


Figure 6.1 Probability of an allele existing at a certain copy number over generational time. Assumes binomial probability of a change in state, representing random fluctuation, and that the allele started as a single copy in a population of one diploid individual.

What do you see when you plot the results of these starting conditions? What you should be seeing is that if you start an allele off being fixed or extinct, it's going to stay in that condition, no matter how long you run your loop. Hence fixation and extinction are *absorbing boundaries* in this model: once you become fixed or go extinct there's no coming back.

A quick word of warning regarding `while` loops: it can be quite easy to mess up an exit condition for a `while` loop so that you never actually meet your condition. This will result in your code running indefinitely and is generally not a good use of anyone's time. If you ever find yourself in a loop limbo, you can either press the red "stop sign" that shows up in the upper right-hand corner of your terminal screen if you're using RStudio, or you can press CTRL and C on your keyboard if you're running R directly on the command line.

Now let's expand our code to something a little more interesting. Let's consider a small population of ten diploid individuals (so a total of twenty loci in the population) that again will start with a single *A* allele cropping up in one individual. We should have twenty-one possible states, since we're always including "zero copies" as our first state. Let's alter our code slightly to account for the multiple intermediate frequencies that can now be possible between the "Fixed" and "Extinct" states:

```
> N <- 10 #Ten diploid individual
> possible <- 0:(2*N) #Number of possible copies of an allele

> P <- NULL #Vector to hold our probabilities
> for(i in possible){
    P <- c(P,dbinom(possible, size=2*N, prob=i/(2*N)))
}

> #Our transition matrix should be 21 rows by 21 columns
> Q <- matrix(P, ncol=2*N+1, byrow=T)

> #Create our starting state matrix
> x <- matrix(c(rep(0,2*N+1)), ncol=2*N+1, byrow=T)

> x[2] <- 1 #Set the prob. of starting with one copy to 100%

> R <- x%*%Q #Get our first gen. transition probabilities

> #Change our color and shape parameters to include
    all the entries between our first "Extinct" state,
    and our final "Fixed" state
> color <- c("brown", rep("blue",ncol(R)-2),"grey")
> shape <- c(15, rep(19,ncol(R)-2), 17)

> #Start with generation 1 for all states
> g <- rep(1,ncol(R))

> #Let's increase our x-axis range to 100
> plot(points(x=NULL, xlim=c(1,100), ylim=c(0,1)
    ylab="Probability",
```

```
    xlab="Generations")

> #The unique() function avoids replicates for color & shape
> legend("bottomleft",
    legend=c("Extinct","Intermediate","Fixed"),
    col=unique(color), pch=unique(shape),
    inset=c(0,1), xpd=TRUE, bty="n")

> while(g[1]<100){
    R <- R%*%Q
    g <- g+1
    points(g, R, col=color, pch=shape)
}

> #Finally let's add two horizontal lines to our plot:
> #The starting allele frequency of A
> abline(h=1/(2*N), lwd=2)
> #And the starting allele frequency of a
> abline(h=1-(1/(2*N)), lwd=2, col="orange", lty=2)
```

From this you should hopefully be seeing something like Fig. 6.2. Notice that the probability of fixation (gray triangles) seems to be reaching an asymptote located at the starting allele frequency of *A*. The fixation probability appears to increase quite slowly and levels off at a probability of 5%. On the other hand, the probability of going extinct (red squares) seems to increase quite quickly and levels off at 95%! This makes some sense: we're saying that there's a random chance of increasing or decreasing in frequency every generation, and if we start with only one copy of an allele, the probability of that one allele still just drifting around the population in 100 generations is going to be pretty slim. Actually, there is an important point here hiding in plain sight. The rarer allele started off at a frequency of 1/20 and after several generations arrived at a probability of fixation of $0.05 = 1/20$. Let's see what happens when we start changing our starting allele frequency. Re-run the code where we set our initial copy number, and this time let's say that half of all our individuals have the *A* allele (an *A* allele frequency of 50%):

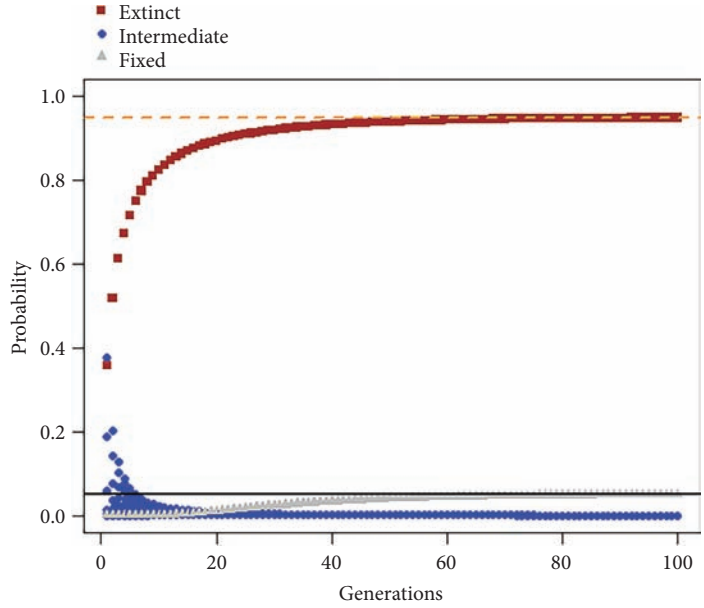


Figure 6.2 Probability of an allele state over generational time in a twenty-locus model. Starting allele frequency ($p = 5\%$) is shown with a solid black line, $1–p$ is shown with a dashed orange line.

```
> x <- matrix(c(rep(0,2*N+1)), ncol=2*N+1, byrow=T)

> x[11] <- 1 #Set the prob. of starting at ten copies to 100%
```

Now re-run the rest of the code to see our transition probabilities when half of all our loci have the *A* allele. You should be seeing something similar to Fig. 6.3. There appears to be a much more gradual increase in the probability of going extinct, as well as in the probability of becoming fixed. Once again, notice that the fixation probability levels off exactly at our starting allele frequency of 50%, and so too does our extinction probability. Under neutral genetic drift, the probability of ultimate fixation of an allele is equal to its starting frequency in the population. For brand new mutations this probability is $1/(2N)$.

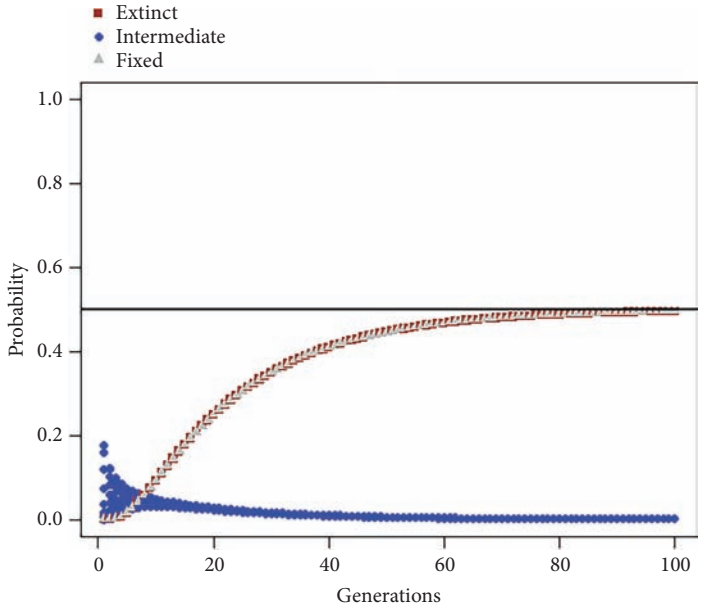


Figure 6.3 Probability of an allele state across generations when starting allele frequency is equal to 50%.

In fact, notice that this plot looks a bit similar to our first probability plot where we only had a single individual. Let's take a look at those side by side (Fig. 6.4). Both of these have the same starting allele frequency of 50%, but the difference is the total number of loci (individuals) in our model. In our first case of the single individual, our probabilities of fixing or going extinct reach their peak frequencies within the first ten generations, while in the larger population it takes almost 100 generations for those probabilities to fully flatten.

This suggests something about the effect of population size on maintenance of alleles that are subject to genetic drift. We're thinking about alleles as being sampled from one generation to make up the next. Genetic drift is the observation of small changes in the proportion of alleles that result from sampling error. Just like with statistical sampling error, large samples (large populations) have smaller sampling errors; that is, smaller

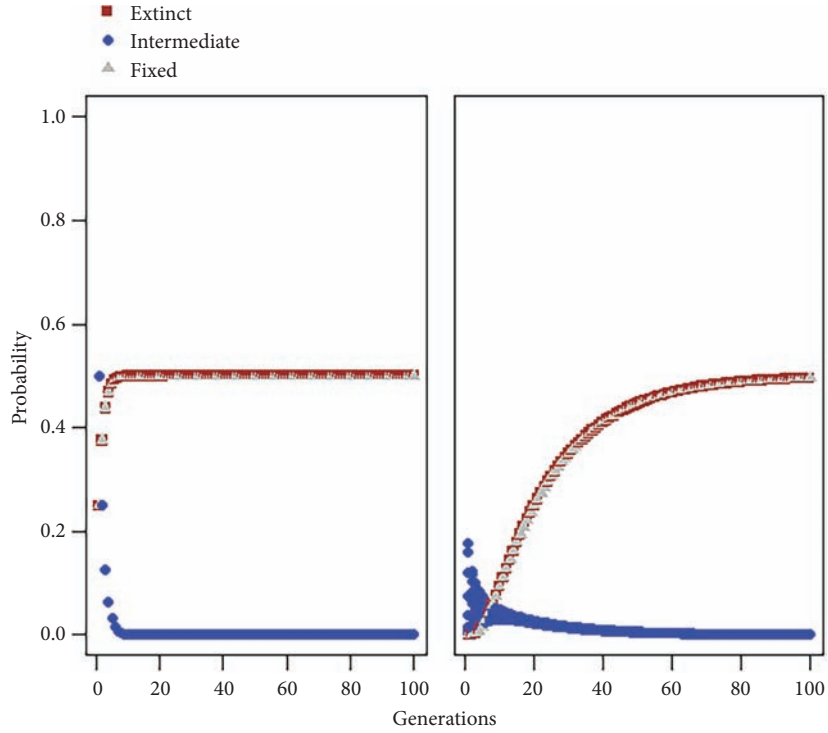


Figure 6.4 Probability of an allele state across generations when starting allele frequency is equal to 50%. Comparing single-diploid individual (left) with ten-diploid individuals (right).

overall changes in allele frequency in each generation. What makes these changes "neutral" is that the observed change is apparently random and without direction. Once again, we have to reiterate that this is a comparatively simplistic model of genetic change over time which makes a lot of simplifying assumptions about the real interplay between selection on individual genetic elements and the stochastic nature of demography across generations.

Despite this inherent randomness, theory allows us to come up with some general rules of thumb (we'll touch on the methods behind these rules in later chapters). For example, it takes about $4 \times N$ generations for a new mutation to reach fixation if we take it as a given that it will eventually fix. Also, after about N generations, we can assume that an allele has

undergone so much drift that it can essentially be at any frequency (it has "forgotten" its starting point). Finally, we assume that neutral variation will not tend to hang around for long. An allele usually reaches loss or fixation in less than $3N$ generations even when starting from an intermediate frequency (much less if close to a frequency of zero or one). While studies that quantify variation over many generations in a controlled setting are difficult, expensive, and therefore exceedingly rare, let's take a look at at least one real dataset of allele frequency changes over multiple generations.

6.2 Variation over time

One of the benchmark empirical studies of the phenomenon of genetic drift was provided by the hero of early genetics: the vinegar fly *Drosophila melanogaster* (they are most often called "fruit flies," but very pedantic entomologists will say that name should belong to the family Tephritidae).

In the fifties, Peter Buri conducted the amazingly laborious experiment of independently evolving over 100 small groups of *D. melanogaster* flies with mutant eye-color phenotypes for twenty generations. He started each group of sixteen flies at a 50% frequency for two *bw* ("brown") alleles, which gave homozygous (bw^{75}/bw^{75}, using his original notation) individuals bright red-orange eyes, while heterozygous (bw^{75}/bw) individuals were distinguishable by noticeably lighter orange eyes. Flies that were not carrying the bw^{75} allele (bw/bw homozygotes) would show up with white eyes. To keep the population size constant, in each generation a new group of sixteen flies was drawn randomly from the offspring of the previous generation to serve as the next group of parents. By looking at eye color, Buri counted the shifts in copy numbers of bw^{75} over time across over 100 replicates, and this whole thing was done twice!

Before we take a look at Buri's data, let's first visualize our earlier simulations in a new way so we can more easily compare it. Let's simulate a scenario that follows Buri's experiment by setting our population size to sixteen individuals. Then let's modify our `while` loop slightly so that we're no longer plotting points on a graph, but instead saving the transition

probabilities for each new generation in a matrix called `Prob`. We'll do this by using the function `rbind` to bind together each new output as a new row in our matrix:

```
> N <- 16 #Our 16 diploid individuals
> possible <- 0:(2*N) #32 possible sites in the population

> #Create our starting state matrix
> x <- matrix(c(rep(0,2*N+1)), ncol=2*N+1, byrow=T)

> x[N+1] <- 1 #Give half our individuals the eye color allele

> P <- NULL #Vector to hold our probabilities
> for(i in possible){
    P <- c(P,dbinom(possible, size=2*N, prob=i/(2*N)))
}

> #This time our transition matrix is 33 rows by 33 columns
> Q <- matrix(P, ncol=2*N+1, byrow=T)

> R <- x%*%Q #Get our first gen. transition probabilities

>#Start our new matrix with our first generation output
> Prob <- R

> #Start with generation 1 for all states
> g <- rep(1,ncol(R))

> #In the loop below, you can simply comment out the
    points function by putting a # in front of it
> while(g[1]<19){ #Change our generation run to 19
    R <- R%*%Q
    Prob <- rbind(Prob,R) #Update our matrix
    g <- g+1
    #points(g, R, col=color, pch=shape)
}
```

Now we have a filled in matrix, `Prob`, that has our transition probabilities across generations. Let's compare these probabilities with the actual

observed counts of allele bw^{75}. Instead of visualizing our output like before, let's visualize our output in three dimensions so that our extinct and fixed states don't overlap on our plot. To do this, we'll use the `persp` function to plot not just our generations and transition probabilities, but our number of alleles as well. In the code below we'll create a 3D plot from simulation data, then read in the observation values from Buri's 1956 paper describing the experiment and plot those data in the same way.

```
> persp(x = 1:g[1], #The range of generations
      y = possible, #Number of alleles possible in the model
      z = Prob, #Matrix of probabilities
      theta = 60, phi = 20, #Adjust to rotate the view
      xlab = "Generations", ylab = "Number of alleles",
      zlab = "Probability",
      shade = 0.3 #Adjust shading of surface
)

#Read in the data from Buri 1956
> data(fly)

> persp(x = 1:g[1],
      y = possible,
      z = fly[-1,], #Matrix of observations,
      omitting first row starting state
      theta = 60, phi = 20,
      xlab = "Generations", ylab = "Number of alleles",
      zlab = "Observations",
      shade = 0.3
)
```

If you want these two plots to be on the same figure, you can actually specify that you want R to arrange a certain number of plots either beside each other, above each other, or arranged in a multiple plot array. To do this, you can use our old friend the `par` function. *Before* running your plot generating code, you can specify how many columns and rows you want to arrange your plots into by specifying those in the argument `mfrow`. For example, if you want to plot our two earlier figures side by side, you can specify that you want upcoming plots to be arranged into a single row with two columns by running the following code before everything else:

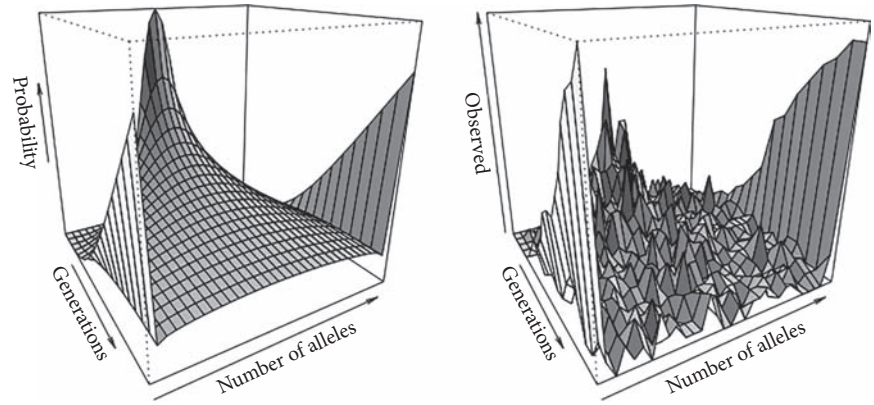


Figure 6.5 Comparison of binomial transition probabilities in a model of sixteen diploid individuals over twenty generations (left) and the observed copy numbers of the *bw*^{75} allele in populations of *D. melanogaster* kept at sixteen individuals per generation (Buri 1956, right).

```
> par(mfrow=c(1,2))
```

Looking at the two plots (Fig. 6.5), we should see that the *bw*^{75} allele behaved very similarly to the binomial sampling model for our hypothetical *A* allele. By the end of both the simulation and the experiment, we see that the highest peaks are observed in the far left "Extinct" state (zero *A*/*bw*^{75} alleles) and the far right "Fixed" state (as many *A*/*bw*^{75} alleles as there are loci). This suggests that without any other forces acting on individual alleles, over time every new variant will either disappear or no longer be a noticeable variant, as it becomes the only allele at a site. The decay of the central peak over a wide range of probabilities should bring to mind diffusion of heat in physics (inspiring the "diffusion approximation" of the neutral theory); however, the absorbing boundaries at frequencies of zero and one are ultimately where the probabilities are concentrated.

So why should we be observing any level of genetic variation within populations? Mutation rates are certainly a factor, but we've already seen that new variants are quite likely to disappear almost as soon as they arise. So let's think about some reasons for seemingly neutral alleles to stick

around in a population. We saw earlier (Fig. 6.4) that population size can influence the fixation and extinction probabilities of individual alleles. Let's visualize this effect again, but this time looking at multiple different simulations at once. We will use `rbinom` to randomly draw the number of alleles in each generation, based on the population size and their starting frequency:

```
> init_p <- 0.05 #Initial allele frequency

> gen <- 100 #Number of generations

> reps <- 10 #How many replicates to run
> colors <- rainbow(reps) #Grab some colors for our reps.

> N <- 10 #Set the population size

> #Initialize a plot that we can add lines to later
> plot(x=NULL, y=NULL, xlim=c(1, gen), ylim=c(0,1),
    xlab="Generations", ylab="Allele frequency")

> #For each replicate: draw new copy numbers of the allele
    per generation, then re-set the allele frequency 'p'
    and use that frequency for the next random draw
> for(i in 1:reps){
    p <- init_p
    for(j in 1:(gen-1)){
        a <- rbinom(n=1,size=2*N,prob=p[j])
        f <- a/(2*N)
        p <- c(p,f)
        }
    lines(x=1:gen, y=p, lwd=2, col=colors[i])
}
```

Try running the above code a few times with different population sizes (changing `N` from 10 to 100, for example). In Fig. 6.6, we show two examples from running the above code at different population sizes. You may notice that at smaller population sizes alleles seem to either go extinct or become fixed more frequently than in larger populations. And as a

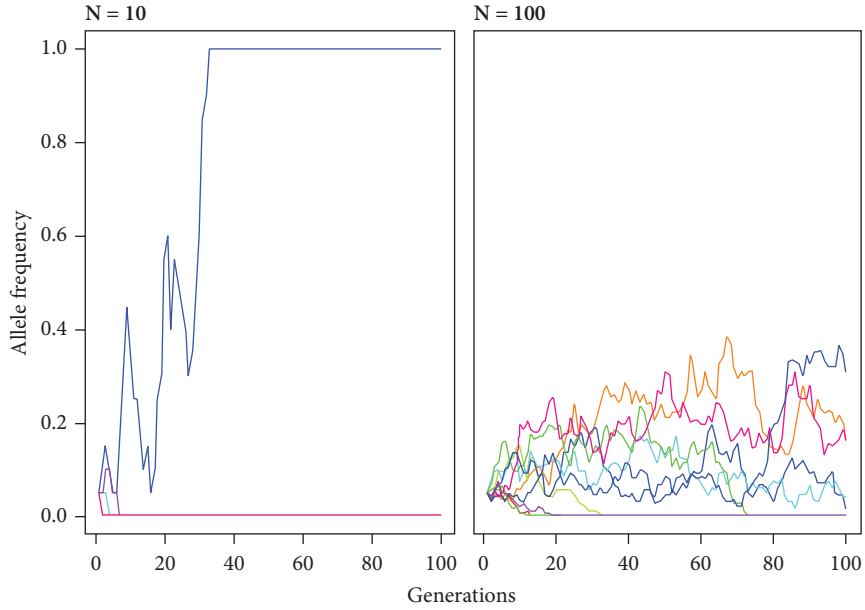


Figure 6.6 Ten replicates of random drift from a starting allele frequency of 0.05. A smaller population (left) loses all but one variant, but that one variant rises to become fixed in the population. A larger population loses half of its variants, and the remaining half stay fluctuating at moderate frequencies.

population gets larger, alleles seem to stick around for longer periods of time. However, try also changing the number of generations (`Gen`) in this simulation. Organisms can have radically different generation times: in the time between two salmon runs you might see ten, twenty, or more generations of fruit flies come and go. How do larger populations compare to smaller ones if you run them for 1,000 generations, as opposed to 100? Loss of variation over time is almost guaranteed, but factors such as population size and generation time significantly affect how rapidly we expect alleles to be lost in a population.

Finally, here is a simulation of genetic drift that keeps track of different trajectories and averages them when completed (Fig. 6.6). You can adjust starting frequencies and population sizes, and while the individual trajectories go all over the place, the average plotted at the end in the thick black line remains essentially unchanged (Fig. 6.7).

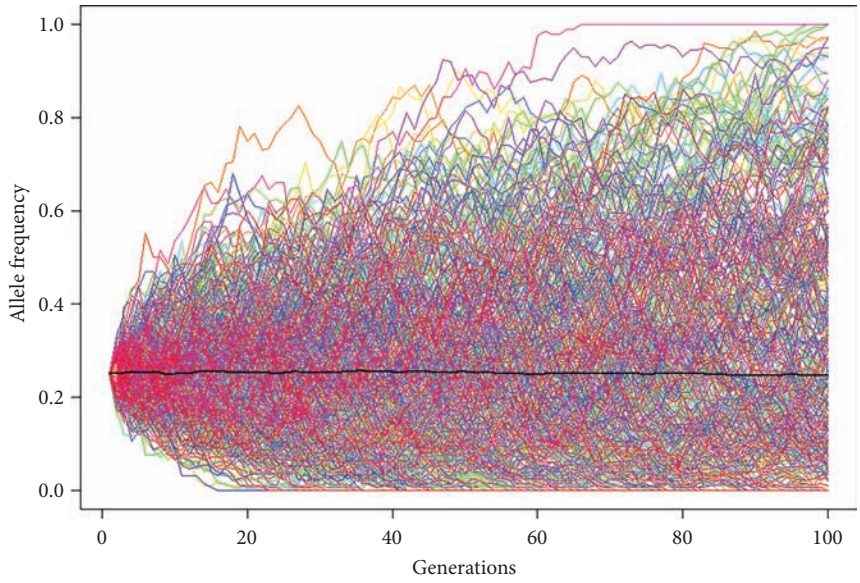


Figure 6.7 Five hundred replicates of random drift in a population of 100 individuals, starting with an allele frequency of 0.25. The average allele frequency across all replicates is plotted in black and stays close to the initial allele frequency.

```
> init_p <- 0.25 #Initial allele frequency
> gen <- 100 #Number of generations
> reps <- 500 #Lots of replicates to run
> colors <- rainbow(reps) #Grab some colors for our reps
> N <- 100 #Population size

> #Initialize a plot
> plot(x=NULL, y=NULL, xlim=c(1, gen), ylim=c(0,1),
    xlab="Generations", ylab="Allele frequency")

> Freq <- NULL #Create an object to save each replicates output

> #Iterate through the replicates
> for(i in 1:reps){
    p <- init_p
    for(j in 1:(gen-1)){
```

```
        a <- rbinom(n=1,size=2*N,prob=p[j])
        f <- a/(2*N)
        p <- c(p,f)
        }
    Freq <- rbind(Freq, p) #Save p
    lines(x=1:gen, y=p, lwd=2, col=colors[i])
}

> #Add the mean of all the replicates to the plot
> lines(1:gen, colMeans(Freq), lwd=2, col="black")
```

6.3 Quantifying variation

We've so far established that drift occurs relatively quickly in small populations, which can lead to the rapid loss of alleles over time. Take a look at Veale et al. (2015) for a real-world example of dramatic genetic drift in the stoat (*Mustela erminea*). To describe this loss of genetic variation over time, Sewall Wright (Wright 1922) came up with a summary statistic called the *fixation* index (F), which gave rise to a whole family of descriptive F-statistics. We should point out here that the summary statistics that we'll talk about in the rest of this book can be derived in multiple different ways. For example, the fixation index can be derived by looking at the percentage of heterozygotes, the rate of selfing in asexually reproducing plants, or, as we'll see here below, the probability of two alleles descending from the same copy of an allele in a pedigree. Don't be thrown when looking through the literature and seeing multiple ways to calculate the same thing (or the same symbols and similar words being used to describe different things). Population geneticists are not the best at coming up with brand new symbols for every new way to quantify phenomena.

As we know, a diploid population of size N has $2N$ possible copies of an allele. Assuming the population size is constant, what is the probability that two randomly chosen alleles in one generation both originated from the same copy of that allele in the generation before? For two alleles it

doesn't matter which allele we pick first, but for the second allele we pick to compare, we're interested in the probability that it comes from the exact same parental copy as our first allele. This constrains the possible origin of this second allele so it only has a $1/2N$ chance of coming from that same parental copy as the first allele. When two alleles have come from the exact same ancestral copy, we call them *identity-by-descent* or IBD. So our total probability that two random alleles sampled from a diploid population are the same is $F = \frac{1}{2N}$.

Conversely, the probability that two alleles were *not* from the same copy in the preceding generation is just $1 - F$ or $1 - \frac{1}{2N}$, which is a measure of genetic diversity often just called heterozygosity (H). If F is our per-generation probability of two alleles *not* being different, then H is just our per-generation probability of maintaining genetic diversity. We can use these values to define the rate of loss of heterozygosity in each generation by quantifying the expected relative change in heterozygosity from a starting value. If we know the starting amount of heterozygosity in generation g, H_g, then the remaining heterozygosity expected in the next generation is

$$H_{g+1} = H_g\left(1 - \frac{1}{2N}\right).$$

The process repeats itself in each generation, with every new generation's value of H modified by $1-\frac{1}{2N}$, so we can jump forward in time g generations from an initial heterozygosity of H_0 like so:

$$H_g = H_0\left(1 - \frac{1}{2N}\right)^g.$$

As we increase the number of generations we consider, $\left(1 - \frac{1}{2N}\right)^g$ actually converges on Euler's constant e, so we can switch from discrete time to a continuous time approximation:

$$H_g = H_0\left(1 - \frac{1}{2N}\right)^g \approx H_0 \times e^{-g/2N}.$$

Let's visualize this process across some different population sizes:

```
> N <- c(10,50,100)
> gen <- 100
> het_init <- 1.0

> line <- c(1,2,4)
> colors <- c("orange", "black", "cyan")

> het <- sapply(1:gen ,function(x) het_init*exp(-(x/(2*N))))

> plot(x=NULL, xlim=c(1,gen),ylim=c(0,1),
    xlab="Generations", ylab="Genetic diversity")

> for(i in 1:nrow(het)){
    lines(1:gen, het[i,], col=colors[i],
    lty= line[i], lwd=2)
}
> legend(x="bottomleft", legend=N, inset=c(0,1), xpd=TRUE,
    bty="n", col=colors, lty=line, lwd=2)
```

Starting with 100% of alleles being different, we see a slow and steady decay in diversity over time (Fig. 6.8). This reduction in genetic diversity over time is the generally assumed result of genetic drift, and the rate of loss is influenced by population size. The decay in heterozygosity is a non-equilibrium process. Without new variants being introduced by mutation or migration, we expect variation to eventually disappear. However, we do expect healthy populations to be able to exist at a "mutation-drift equilibrium," a point at which the population is large enough that it's not losing variants much faster than it can accumulate new ones. This equilibrium perspective is useful when thinking about a population that has become much smaller than it was previously, so that the variation that existed in the population at mutation-drift equilibrium can no longer be maintained.

One example is the Hawaiian crow or ʻalalā (*Corvus hawaiiensis, See Fig. 6.9*), which was extinct in the wild and maintained in captivity for decades as a population of approximately 100 individuals. What is the expected time for half of the genetic variation (defined as heterozygosity) averaged across the genome to be lost due to genetic drift under these

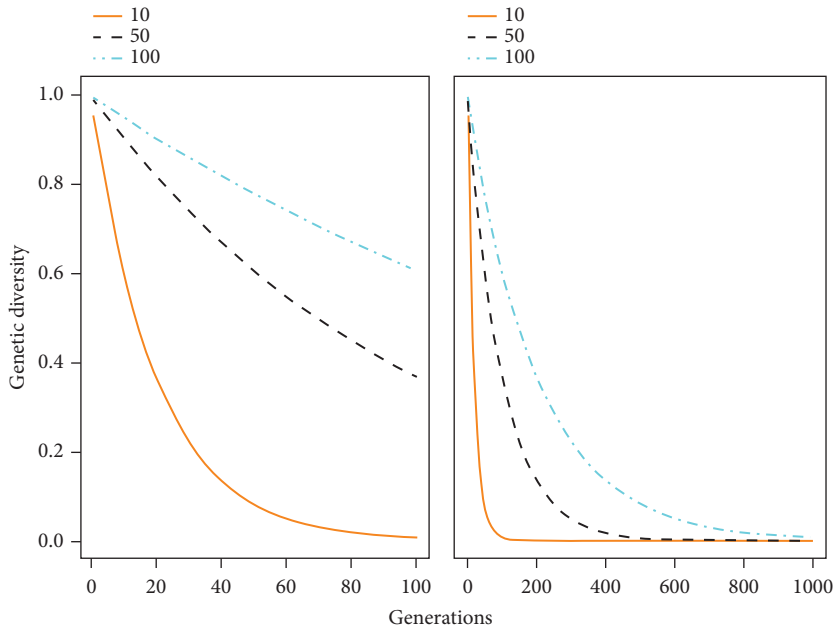


Figure 6.8 Rate of genetic diversity (*H*) loss in populations of different sizes, across two different time scales. Over longer time spans, large populations are expected to follow the same trend as smaller populations.

circumstances? To answer this, we can solve for g

$$H_g = H_0 e^{-g/(2\times N)}$$
$$g = -(2N) \times \log_e(H_g/H_0)$$

then set $H_0 = 2$, $H_g = 1$ (or any ratio of 2:1), and calculate g.

```
> -2*100*log(1/2)
[1] 138.6294
```

So half of the original variation is expected to still exist after about 138 generations (at a constant population size of 100, with no selection, and with all individuals equally likely to reproduce).

Finally, the treatment of the change in H is a deterministic expectation (or average over a large number of loci); however, it should be clear from

Figure 6.9 The Hawaiian crow or ʻalalā (*Corvus hawaiiensis*). Photo from the US Fish and Wildlife Service.

the simulations that there is a large variance around the expected change in allele frequencies for a single locus. It is important to be aware of these two perspectives (single locus and genome average) simultaneously when thinking about and interpreting genetic drift.

The variance in allele frequency between generations due to drift is

$$\sigma^2 = \frac{p(1-p)}{2N}.$$

This makes some sense, since drift of two alleles is binomial and the variance expected from binomial sampling is

$$\sigma^2 = np(1-p).$$

However, it looks like the sample size, n or $2N$, is in the wrong place. This is because the variance is rescaled to be a proportion out of a total range of $2N$ with a caveat that variance is squared, which gives $4N^2$:

$$\sigma^2 = \frac{2Np(1-p)}{4N^2} = \frac{p(1-p)}{2N}.$$

6.4 Equilibrium heterozygosity and effective population size

Earlier we talked about the decay of heterozygosity in a small population with the ʻalalā example. However, we can also talk about expected heterozygosity at mutation-drift equilibrium: the rate of input of new mutations (2μ) is counterbalanced by the rate of removal of existing genetic variation by genetic drift (both forces are considered in terms of pairs of alleles):

$$2\mu(1-H) = H \times 1/(2N).$$

Note that if H is small, $1 - H \approx 1$:

$$H = 4N\mu.$$

Another derivation is given in the last chapter when discussing the coalescent.

However, predicted levels of genetic variation often do not match what is expected from actual "census" population sizes. This has led to the concept of an effective population size, N_e; in other words, the population size that would lead to observed levels of genetic diversity in an idealized (random mating, equal probability of reproducing, constant population numbers) population.

Humans are a perfect example of this distinction. The per-nucleotide, per-generation mutation rate has been estimated as on the order of 10^{-8} (for example, Tian et al. 2019). Average nucleotide heterozygosity—the rate at which a pair of DNA sequences differ in humans and would form a heterozygote if paired together in an individual—is on the order of one

difference out of approximately 1,600 DNA base pairs or 0.0006 per base pair (for example, Stephens et al. 2001). Solving for N_e gives us $N_e = H/(4\mu)$, so given our measures of mutation rate and heterozygosity we can calculate the human N_e:

```
> H <- 0.0006
> u <- 10^-8
> H/(4*u)
[1] 15000
```

This should come as quite a shock. Our genetic diversity implies that there are only about 15,000 humans alive on the planet in any given generation! However, our census size is, at the time of writing, $N \approx 7.6 \times 10^9$; how can this be so far off?

There are a number of complexities to this, but to cut a long story short, one of the major factors is that genetic variation can quickly be lost but takes a long time (through slow mutations and genetic drift) to be regained.

To illustrate, let's have a population that switches between two sizes over time, N_1 for t_1 proportion of the time, and N_2 for the remaining t_2 part of time. Drift will be accelerated when it is at the smaller size and slow down at the larger size. How do we estimate an "effective" population of constant size that has the same overall rate of genetic drift as the oscillating population? One natural way to do this is to set the variance of the constant equal to the average variance (σ^2 from earlier) of the changing population:

$$\frac{p(1-p)}{2N_e} = t_1\frac{p(1-p)}{2N_1} + t_2\frac{p(1-p)}{2N_2}.$$

The first thing we can do is cancel out shared terms and simplify:

$$\frac{1}{N_e} = t_1\frac{1}{N_1} + t_2\frac{1}{N_2}$$
$$N_e = \frac{1}{t_1\frac{1}{N_1} + t_2\frac{1}{N_2}}.$$

This pattern, the reciprocal of the average reciprocals, is called the harmonic mean. The effective population size is expected to be equal to the harmonic mean of individual population sizes over time. Smaller numbers have larger effects on harmonic means. In population genetics, this means that accelerated drift in smaller populations can have a dominant lasting effect.

This is generalizable to an arbitrary number of population sizes and corresponding times:

$$N_e = \sum_i \left(\frac{t_i}{N_i} \right)^{-1}.$$

For our species, this means that we went through a bottleneck of less than 15,000 people in our past. Other human populations essentially went extinct (except for living on in small portions of our genome), and our numbers were dangerously low as well until we rebounded.

6.5 Overlapping generations

A final note is the issue of overlapping generations. Because of historical inertia, much of classical population genetics is based on discrete non-overlapping generations, known as a Wright–Fisher model (refer back to section 6.1), which is appropriate for annual plants or some types of invertebrates. However, many species are better modeled in continuous time as a range of ages at any given time where an individual death is replaced by an individual birth—known as a Moran model and popular in evolutionary game theory—rather than an entire generation at once. In reality, wild populations are somewhere in between these extremes; for example, even with overlapping generations individuals are more likely to reproduce in a particular age range. In general there is not much difference between essential results other than adjusting the rate of change. For example, the rate of genetic drift in a purely Moran model is twice the

rate of drift in a purely Wright–Fisher model (the allele frequency changes within a generation instead of only between generations). However, if there is extreme unevenness, such as with plant seed reservoirs in the soil, "lost" alleles can be recaptured by germination of older seeds and the effective population size is increased. We've now focused a fair bit on random change in allele frequencies.