BCOR 102 Lecture Notes

Nicholas J. Gotelli

5 November 2015

Lecture Notes For Wednesday 21 October 2015

Definitions

`evolution (general definition)` Sustained change in the phenotype (= appearance) of a system through time; includes non-biological phenomena such as the universe, culture, music

`evolution (biological)` Change in the allele frequencies of a population through time

`genotype` Underlying genetic constitution of an organism

`phenotype` Physical appearance of an organism; its observed traits

$$genotype \overset{\text{environment,development}}{\longrightarrow} phenotype$$

$$phenotype = genotype + environment$$

`gene` A section of DNA on a chromosome that codes for a specific trait (*e.g.*, flower color);

`locus` location on a chromosome where a gene occurs

`allele` One of two or more alternative states that exist for a gene (*e.g.*, red, white, purple)

`homozygote` Individual that has 2 identical alleles at a locus (rr or RR)

`heterozygote` Individual that has 2 different alleles at a locus (Rr)

`dominant allele` One whose phenotype is expressed in either homozygous or heterozygous individuals

`recessive allele` One whose phenotype is expressed only in homozygous individuals

Complications of Mendelian Inheritance

`pleiotropy` A single gene affects more than one trait

`epistasis` Gene-gene interactions; the expression of one gene affects another

`polygenic trait` Small additive effects of many genes on a single trait, such as body mass

`environmental effects` The same genotype can have a different phenotype depending on the environment in which it is raised (*e.g.* appearance of same plant clone raised in sun versus shade)

The combination of environmental effects and a polygenic trait leads to a trait that is measured on a continuous scale, and often has a normal or bell-shaped distribution. For example, body mass or height is a continuous trait. But if it were inherited as a single Mendelian gene with only 2 alleles and no environmental effects, individuals would be of only two sizes: “tall” or “short”.

Analysis of a Punnet Square

A Punnet square is a simple table to determine the genotypes and phenotypes produced from a parental cross. You need to know the genotypes of both parents to draw the square.

1. Along the top of the square, list the gamete types that could be produced by one of the parents.
2. Along the side of the square, list the gamete types that could be produced by the other parent.

Remember that each gamete type is equally likely, so divide the rows and columns of the square evenly.

3. Inside each cell of the table, write the resulting genotype and phenotype of the offspring. Each of these cells is equally probable.
4. Tabulate the relative frequencies of the different genotypes and the different phenotypes in the offspring.

Example Punnet Square for a Single Gene with Two Alleles

Alleles

R = red flower color
r = white flower color

Parental Genotypes and Phenotypes

Father's genotype: rr
Father's phenotype: white flower

Mother's genotype: Rr
Mother's phenotype: red flower

Gametes	r	r
$\boldsymbol{R}$	rR	rR
$\boldsymbol{r}$	rr	rr

Offspring genotypic frequencies: Rr:rr 0.5:0.5
Offspring phenotypic frequencies: red:white 0.5:0.5

Example Punnet Square for Two Genes with Two Alleles Each

Alleles

R = red flower color
r = white flower color
Y = smooth seed coat
y = wrinkled seed coat

Parental Genotypes and Phenotypes

Father's genotype: $RrYy$
Father's phenotype: red flower, smooth seed coat

Mother's genotype: $RrYy$
Mother's phenotype: red flower, smooth seed coat

Gametes	RY	rY	Ry	ry
RY	$RRYY$	$RrYY$	$RRYy$	$RrYy$
rY	$RrYY$	$rrYY$	$RrYy$	$rrYy$
Ry	$RRYy$	$RrYy$	$RRyy$	$Rryy$
ry	$RrYy$	$rrYy$	$Rryy$	$rryy$

Offspring genotypes: RRYY: RRYy: RRyy: RrYY: RrYy: Rryy: rrYY: rrYy: rryy
Offspring genotype counts: 1: 2: 1: 2: 4: 2: 1: 2: 1
Offspring genotypic frequencies: $\frac{1}{16}, \frac{2}{16}, \frac{1}{16}, \frac{2}{16}, \frac{4}{16}, \frac{2}{16}, \frac{1}{16}, \frac{2}{16}, \frac{1}{16}$

Offspring phenotypes: RedSmooth: Redwrinkled: whiteSmooth: whitewrinkled
Offspring phenotype counts: 9: 3: 3: 1
Offspring phenotypic frequencies: $\frac{9}{16}, \frac{3}{16}, \frac{3}{16}, \frac{1}{16}$

Lecture Notes For Friday 23 October 2015

Definitions

`gene pool` The set of all alleles in an interbreeding population

`genotypic frequencies` The proportion of different genotypes in the population

`allelic frequencies` The proportion of different alleles in the population (regardless of genotype)

Calculating Genotypic and Allelic Frequencies

The genotypic and allelic frequencies can always be calculated directly from data in which a sample of individuals from a population is genotyped (from direct sequencing, SNP analysis, or measures of protein diversity). This calculation of observed genotypic and allelic frequencies does not make any assumptions about evolution or genetic change; it is just a snapshot of genetic diversity that has been measured.

Here are some sample data in the form of counts of different genotypes for a single gene locus:

Genotype	AA	AB	BB	Sum
Number of individuals	75	20	100	200

Calculating Genotypic Frequencies

```
f(AA) = 75/200 = 0.385
f(AB) = 20/200 = 0.103
f(BB) = 100/200 = 0.512
```

Calcuating Allelic Frequencies

The allelic frequency calculation is slightly more calculated. Remember that each homozygote carries two copies of a particular allele, but a heterozygote carries only a single copy. So, we use `0.5 x f(AB)` to get the contribution of the heterozygote to the allelic frequency:

```
f(A) = f(AA) + 0.5*f(AB)
f(A) = 0.385 + 0.5*0.103 = 0.436

f(B) = F(BB) + 0.5*f(AB)
f(B) = 0.512 + 0.5*0.103 = 0.564
```

As a check on your work, make sure that the genotypic frequencies sum to 1.0 and the phenotypic frequencies sum to 1.0.

Calculating Genotypic and Allelic Frequencies When There Are More Than Two Alleles

This is slightly more complex, because you have to list out all of the possible genotypes, but the formulas are essentially the same. Here is an example of a single gene with 3 alleles J, K, and L

Genotype	JJ	JK	JL	KL	KK	LL	Sum
Number Of Individuals	10	11	0	9	2	22	54

Here are the genotype frequencies:

```
f(JJ) = 10/54 = 0.185
f(JK) = 11/54 = 0.204
f(JL) =  0/54 = 0.000
f(KL) =  9/54 = 0.167
f(KK) =  2/54 = 0.037
f(LL) = 22/54 = 0.407
```

And here are the allelic frequencies. We often use small variables p,q,r... to indicate different alleles:

```
p = f(J) = f(JJ) + 0.5*f(JK) + 0.5*f(JL)
p = f(J) = 0.185 + 0.5*0.204 + 0.5*0.000 = 0.287

q = f(K) = f(KK) + 0.5*f(JK) + 0.5*f(KL)
q = f(K) = 0.037 + 0.5*0.204 + 0.5*0.167 = 0.222

r = f(L) = f(LL) + 0.5*f(JL) + 0.5*f(KL)
r = f(L) = 0.407 + 0.5*0.000 + 0.5*0.167 = 0.491
```

Hardy-Weinberg Model

The Hardy-Weinberg equation (named after two population geneticists from the 1920s) uses simple rules of probability to generate the expected genotypic frequencies in a population that is subject only to random mating (see assumptions). It is based on the idea that, with random mating, the alleles present in the gene pool are paired up randomly in the genotypes of the offspring. We present the equation, show how we use it with data, and then list the assumptions.

The reasoning behind the Hardy-Weinberg equation is that the frequencies of alleles in the gene pool can be interpreted as probabilities of an allele being present in a single individual. Because each individual has two alleles for a gene, we end up multiplying probabilities together to get the expected frequency of a particular genotype.

If we have allele frequencies in a population p,q,r... that add up to 1.0, a simple binomial expnasion gives us the expected frequencies of each genotype:

Let p = f(A) allele and q = f(B) allele.

Because these are the only two alleles for this gene locus

$$p + q = 1.0$$

$$(p + q)^2 = 1.0^2$$

$$p^2 + 2pq + q^2 = 1.0$$

$$f(AA) + f(AB) + f(BB) = 1.0$$

Hardy Weinberg Calculation for 1-Gene, 2-Allele Example

Using the population data given above for the A and B alleles

```
Observed f(A) = 0.436
Observed f(B) = 0.564

f(AA) in Hardy-Weinberg equilibrium = p^2 = (0.436)*(0.436) = 0.1901

f(AB) in Hardy-Weinberg equilibrium = 2*p*q = 2*(0.436)*(0.564) = 0.4918

f(BB) in Hardy-Weinberg equilibrum = q^2 = (0.564)*(0.564) = 0.3181
```

Hardy Weinberg Calculation for 1-Gene, 3-Allele Example

When there are 3 alleles present in a population at a gene locus, we can use p, q, and r to represent their frequencies:

$$p + q + r = 1.0$$

$$(p + q + r)^2 = 1.0^2$$

$$p^2 + 2pq + 2qr + 2qr + q^2 + r^2 = 1.0$$

$$f(JJ) + f(JK) + f(JL) + f(KL) + f(KK) + f(LL) = 1.0$$

Using the population data given above for the J, K, and L alleles

```
p = Observed f(J) = 0.287
q = Observed f(K) = 0.222
r = Observed f(L) = 0.491

f(JJ) in Hardy-Weinberg equilibrium = p^2 = (0.287)*(0.287) = 0.0824

f(JK) in Hardy-Weinberg equilibrium = 2*p*q = 2*(0.287)*(0.222) = 0.1274

f(KL) in Hardy-Weinberg equilibrium = 2*p*r = 2*(0.287)*(0.491) = 0.2818

f(KL) in Hardy-Weinberg equilibrium = 2*q*r = 2*(0.222)*(0.491) = 0.2180

f(KK) in Hardy-Weinberg equilibrium = q^2 = (0.222)*(0.222) = 0.0493

f(LL) in Hardy-Weinberg equilibrum = r^2 = (0.491)*(0.491) = 0.2411
```

Assumptions of Hardy-Weinberg

1. No mutation
2. No migration
3. Random mating
4. No natural selection
5. Large population size
6. Random segregation of alleles

Genetic changes with Hardy-Weinberg

If the Hardy-Weinberg assumptions are met: - allelic frequencies never change

- genotypic frequencies will change in a single generation of random mating from the observed frequencies to those predicted by the Hardy-Weinberg model
- once the Hardy-Weinberg genotypic frequencies are achieved after a single generation of random mating, they will not change again in future generations

Remember that allelic frequencies can always be calculated from genotypic frequencies. This calculation involves no biological assumptions, it is just simple book-keeping.

However, in order to predict genotypic frequencies from allelic frequencies, we have to assume Hardy-Weinberg or some other kind of biological model that tell us what happens to allelic and genotypic frequencies each generation.

Lecture Notes For Monday October 26th

Mutation is the ultimate source of genetic variation in populations, but is it a strong molecular force by itself?

Varieties of mutation

With 4 possible nucleotides, there are $4^3 = 64$ possible 3-codon combinations. However, there are only 20 amino acids. Therefore, some substitutions (silent mutations) will code for an identical amino acids. Others

(neutral mutations) will change the amino acid, but not alter the performance of the protein. Some codons indicate a start/stop to protein production, and such mutations are usually detrimental. So are frame-shift mutations in which codon sequences are misread.

- point mutations (silent, neutral, beneficial, detrimental, frameshift, start/stop)
- Single Nucleotide substitutions (SNPs)
- microsatellites (repeated sequences of 2-6 nucleotides)
- gene duplications
- chrosome inversions
- polyploidy

For eukaryotes, rates of mutation are on the order of 10^{-4} to 10^{-6} mutations/gene locus/generation.

Consider an allele A, with an initial allelic frequency of p_0. Each generation A alleles mutate into B alleles at a mutation rate of u. After t generations of time, we have

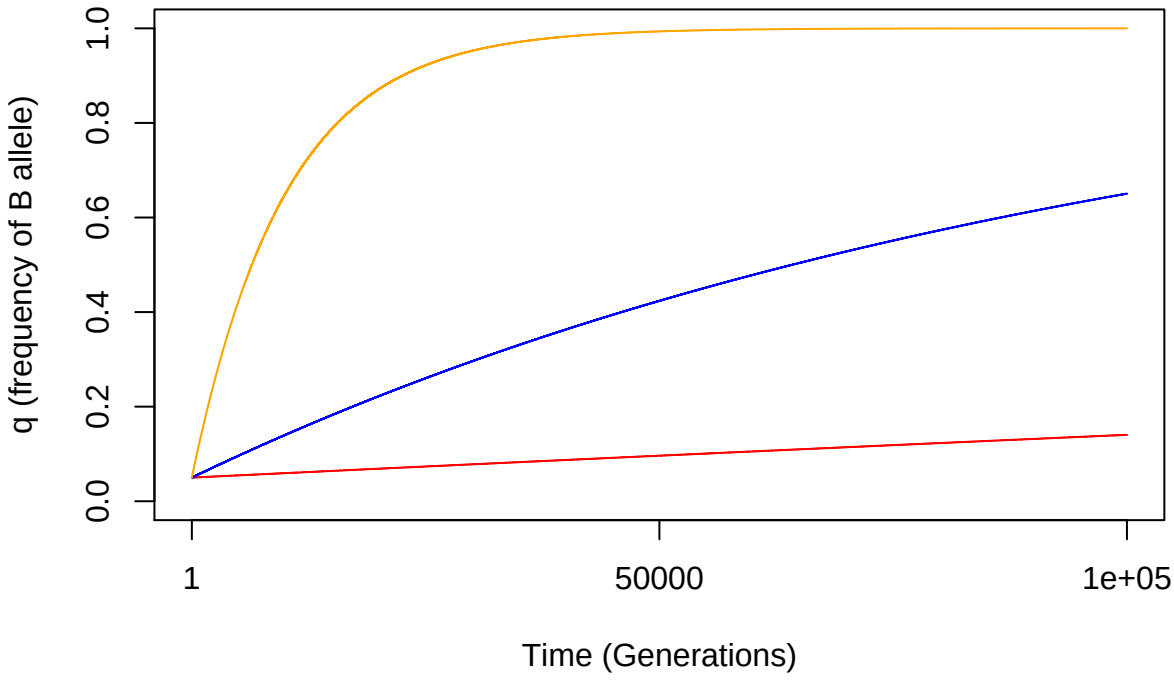
$$q_t = 1 - p_0 e^{-ut}$$

For example, suppose $p_0 = 0.95$, $u = 10^{-6}$, and $t = 100$ generations. How much of an increase will occur in the frequency of the B allele, which is starting out at f(B) $= 1 - p = 0.05$?

```
u = 10^-6
p0 = 0.95
t = 100

q(100) = 1 - p0e^{-ut}
q(100) = 1 - 0.95e^{-10^{-6}*100}
q(100) = 0.050095

Even after 1000 generations, the change is only
q(1000) = 0.0509
```

Here is a graph illustrating the change through time

```r
p0 <- 0.95
t <- 1:1e+05

marks <- c(1, 50000, 1e+05)
u <- 1e-06
qt <- 1 - p0 * exp(-u * t)
qt5 <- 1 - p0 * exp(-1e-05 * t)
qt4 <- 1 - p0 * exp(-1e-04 * t)

plot(x = t, y = qt, xlab = "Time (Generations)", ylab = "q (frequency of B allele)",
    type = "l", col = "red", xaxt = "n", ylim = c(0, 1))
axis(1, at = marks, labels = marks)
points(x = t, y = qt5, type = "l", col = "blue")
points(x = t, y = qt4, type = "l", col = "orange")
```



At this rate, it would take nearly 5 million generations (5×10^6) for the B allele to go from a frequency of 0.05 to 0.95. The increase is faster if the mutation rate is 10^{-5} (blue curve), but even at the faster rate of 10^{-4} (orange curve), it still takes almost 50,000 generations for a substantial increase in allele frequency caused only by mutation.

Finally, we note that this analysis assumes that mutation only occurs in one direction (from A to B). But if there is also back mutation from B to A occurring at rate v, then the allele will never go to fixation. Instead an equilibrium will be reached at:

$$\hat{q} = \frac{u}{u+v}$$

$$\hat{p} = \frac{v}{u+v}$$

Lecture Notes For Friday October 30th

Migration

What are the effects of migration on allelic frequency? By migration, we mean the arrival of individuals from another population.

p_0 = initial allele frequency in resident population (changing)

p_m = allelic frequency in migrant population (constant)

t = time, in number of generations

m = migrant fraction (proportion of the population that consists of new migrants each generation)

1 - m = resident fraction (proportion of population that consists of non-migrants each generation)

To calculate the allelic frequency in the resident population after one generation of migration, we have:

$$p_1 = (1 - m)p_0 + (m)p_m$$

More generally, after t generations, the frequency of the allele in the resident population (p_t)

$$p_t = (1 - m)^t(p_0 - p_m) + p_m$$

Sample Migration Problem

Given an intial resident frequency $p_0 = 0.5$, a migrant allele frequency $p_m = 0.9$, and the passage of t = 10 generations, the new allelic frequency in the resident population (p_{10}) is:

```
p10 = (1 - 0.1)^10 * (0.5 - 0.9) + 0.9 = 0.76
```

So, after only 10 generations, the allele frequency changes from $p_0 = 0.50$ to $p_{10} = 0.76$. In order for this calculation to hold, all of the other Hardy-Weinberg assumptions need to be in place (no mutation, no selection, large population size, random mating, random segregation of alleles).

Models Of Genetic Variation

If we examine a random stretch of DNA in an organisms's genome, how much variation will be present, and how will it be structured?

- `Classical model` Very low genetic variation. Most genes have two homozygous "wild type" alleles (+), with an occasional recessive allele (r) showing up that is usually deleterious. Natural selection operates mostly as purifying selection, removing recessive alleles that are deleterious. This was the view in the early 1900s that emerged from classical genetics, when the only way that "genotypes" could be scored was on the basis of major mutations (which often were deleterious).
- `Balance model` Low genetic variation, but some variants are maintained through `balancing selection` in which the fitness of the heterozygote (AB) is superior to that of either of the two homozygote genotypes (AA or BB). This can happen when the two protein variants expressed in a homozygous individual function optimally at slightly different conditions, which can increase the fitness of an individual in a variable or changing environment. Sickle-cell anemia and resistance to malaria is the classic example.
- `Neutral model` High genetic variation, with many different alleles in the population and many heterozygous loci in different individuals. These kind of alleles have no effect (good or bad) on the fitness of the organism, although in different environments or at some time in the past, they may have fitness consequences.

With modern sequencing methods revealing large amounts of genetic diversity in most species, the consensus view is that the balance and neutral models capture the typical pattern, although of course there are still many examples of deleterious recessive alleles that match the classic model.

Non-Random Mating

Hardy-Weinberg assumes random mating, but there are a number of different possibilities for how individuals choose mates:

- `random mating` Mate choice is independent of genotype or phenotype
- `positive assortative mating` More frequent matings between similar phenotypes
- `negative assortative mating` more frequent matings between dissimilar phenotypes
- `inbreeding` More frequent matings between relatives

The Inbreeding Coefficient

The degree of inbreeding can be quantified with the inbreeding coefficient, F

- `inbreeding coefficient` The fractional reduction in heterozygosity relative to a randomly mating population.

$$F = \frac{H_0 - H}{H} = 1 - \frac{H}{H_0}$$

where H is the observed heterozygosity in the population and H_0 is the expected heterozygosity in a Hardy-Weinberg population (2pq).

An equivalent definition comes from the pattern of an individuals predigree:

- `autozygous alleles` Two alleles in an individual that are identical by descent from a single ancestor
- `allozygous alleles` Two alleles in an individual that are identical by descent from two different ancestors
- `inbreeding coefficient (pedigree definition)` The probability that two alleles in an individual are identical by descent (=autozygous)

Genotype Frequencies Expected With Inbreeding

With these definitions, we can modify the Hardy-Weinberg equation to give the expected genotype frequencies with inbreeding:

Genotypes	Frequencies	F=0	F=1
AA	$p^2(1 - F) + pF$	p^2	p
AB	$2pq(1 - F)$	2pq	0
BB	$q^2(1 - F) + qF$	q^2	q

For example, suppose the allele frequencies are p = 0.2, q = 0.8, and F = 0.5. We would have:

```
f(AA) = 0.2^2(1 - 0.5) + 0.2*0.5 = 0.12
f(AB) = 2*0.2*0.8*(1 - 0.5) = 0.16
f(BB) = 0.8^2(1 - 0.5) + 0.8*0.5 = 0.72
```

So, the primary effect of inbreeding is to reduce the frequency of heterozygotes. In the extreme case of full inbreeding, there are no heterozygotes, and we end up with two inbred homozygote lines.

Costs And Benefits Of Inbreeding

The costs of inbreeding are:

- expression of deleterious recessive alleles (short-term)
- loss of heterozygosity (long-term)

These problems are especially acute for small populations (which are often highly inbred) that may be facing novel environments due to climate change and other factors.

However, there is also an argument that inbreeding could benefit a population by preserving particular genotypes that function well together (= `co-adapted gene complex`).

This effect would be most beneficial for organisms living in stable environments whose offspring do not disperse very far from their parents. Accordingly, there are many examples of restricted plant populations with little or no genetic variability that seem perfectly healthy (at least until the climate changes).

Lecture Notes For Monday November 9th

Genetic Drift

Allele frequencies in a population can change from random effects caused by the segregation of alleles into gametes during meiosis. For example, imagine a cross between two heterozygous individuals that produce a total of 400 offspring:

Genotype	Frequency	Expected Number of Offspring
AA	0.25	100
AB	0.50	200
BB	0.25	100

Allele	Expected Allele Frequency
A	0.50
B	0.50

Of course, by chance, we might not see precisely these numbers. Suppose the counts look like this:

Genotype	Observed Frequency	Observed Number of Offspring
AA	0.2525	101
AB	0.5000	200
BB	0.2475	99

This deviation has a trivial effect on the allele frequencies:

Allele	Observed Allele Frequency
A	0.5025
B	0.4975

But now imagine the same scenario for a cross that produces only 4 offspring:

Genotype	Frequency	Expected Number of Offspring
AA	0.25	1
AB	0.50	2
BB	0.25	1

Allele	Expected Allele Frequency
A	0.50
B	0.50

Look what happens this time if the genotype counts are shifted by just one individual:

Genotype	Observed Frequency	Observed Number of Offspring
AA	0.5000	2
AB	0.5000	2
BB	0.0000	0

This deviation has a huge effect on the allele frequencies:

Allele	Observed Allele Frequency
A	0.75
B	0.25

In this case, the frequency of the A allele has changed from 0.50 to 0.75 in a single generation. Remember that this change will affect all of the descendants of this cross. Even if the population size should return to 400 individuals, this random change in allele frequencies will affect the subsequent evolution of this population.

`genetic drift` random changes in allelic and genotypic frequencies caused by small population size.

You can see from this example that genotypic and allelic changes from genetic drift are much more important in small populations than in large populations. Below a size of roughly 100 individuals, genetic drift becomes very important.

The Probability of Allele Fixation

`fixed allele` a gene locus that has only a single allele in a population is "fixed" because the allele frequency is 1.00, and every individual in the population is homozygous for the same allele.

Suppose a single new allele arises in a population from mutation. The probability of fixation of this allele is its frequency in the gene pool:

$$p(\text{fixation}) = \frac{1}{2N}$$

Remember that there are 2N alleles in the gene pool for a single gene locus.

However, with mutation rate u, each generation there will be 2nu copies of the mutant produced. Thus, the probability of fixation in each generation is:

$$p(\text{fixation in one generation}) = \frac{2Nu}{2N} = u$$

Thus, with mutation and genetic drift, we have the interesting result that the probability of fixation each generation = u, the mutation rate of the allele.

As we discussed before, u is a small number, so the chances are slim. But what is the probability of fixation after t generations?

$$p(\text{fixation in one generation}) = u$$

$$p(\text{no fixation in one generation}) = 1 - u$$

$$p(\text{no fixation in t generations}) = (1 - u)^t$$

$$p(\text{fixation in at least one of t generations}) = 1 - (1 - u)^t$$

This formula applies not to just genetic drift, but to all chance events in life. Suppose for example, that you estimate that the chances of getting a ticket for speeding are 1/100, and you only speed on Friday afternoons (to get home from work quickly to start your weekend).

What are the chances of getting caught speeding at least once during a year in which you work 50 weeks?

```
p(speeding ticket during one year) =

1 - (1 - 0.1)^50 = 0.39
```

So, a 39% chance of getting caught at some time during the year, even though the chance of getting caught each individual Friday is only 0.01.

And if you commute like this for 5 years in a row?

```
p(speeding ticket at least once during 5 years) =

1 - (1 - 0.39)^5 = 0.92
```

A 92% chance you will get caught!

Back to the world of population genetics. Recall that mutation rates in the real world are on the order of 10^{-6} to 10^{-4}. What are the chances of fixation after 1000 generations?

```
p(fixation) = 1 - (1 - 0.0001)^1000 = 0.095
p(fixation) = 1 - (1 - 0.000001) ^1000 = 0.001
```

Again, these probabilities are very small. In the long-run, if we wait long enough, alleles arising by mutation will be fixed by chance. But other forces operate much more quickly to change allele frequencies. Notice that these calculations for fixation are not affected by population size. However, it is the case that allele frequencies in small populations fluctuate much more from drift than do large populations, even if those fluctuations don't necessarily lead to fixation.

Effective Population Size

There is an important distinction between the observed population size (N) and the `effective population size`(N_E):

`effective population size` The equivalent number of individuals in a randomly mating population

In general:

$$N_E < N$$

Why should the effective population size be anything less than the observed population size? There are a number of forces at work that reduce the effective population size by preventing the complete mixing of alleles that we expect in a population that is mating at random. These factors include

- `founder effect` If a population is colonized by only a few individuals (think of islands), the alleles carried by those colonizers will be a small— and often non-random— subset of the larger population they originated from.
- `bottleneck` If a population shrinks back to a small size — even for a single generation — that will reduce the effective population size more than we would expect by calculating a simple arithmetic average of the observed population sizes in consecutive generations. Thus, we can think of the founder effect as a special case of a bottleneck that occurs during colonization.
- `unbalanced sex ratio` If the ratio of males:females in a sexually reproducing population is different from 1:1, the alleles represented by the rarer sex will be disproportionately represented in the next generation.
- `limited natal dispersal` If individuals disperse only a limited distance from where they were born, they will only encounter a limited number of potential mates. Even if they mate randomly, the allelic diversity in this limited spatial "neighborhood" is less than that of the entire population.

Let's look at some simple equations for calculating N_E under these circumstances

Effective Population Size With A Bottleneck

If observed population size n_i changes in each of t consecutive generations:

$$\frac{1}{N_E} = \frac{1}{t}(\frac{1}{n_1} + \frac{1}{n_2} + \frac{1}{n_3} + \frac{1}{n_4} + \cdots + \frac{1}{n_t})$$

For example, suppose the observed population size of a population of orchids is 100,4,100,100,100, undergoing a bottleneck in generation 2, but then fully recovering in subsequent generations. For this sequence, N_E is calculated as:

```
1/N_E = 1/5(1/100 + 1/4 + 1/100 + 1/100 + 1/100)
1/N_E = 1/5(1//100 + 1/4 + 1/100 + 1/100 + 1/100)
1/N_E = 1/5(1/100 + 25/100 + 1/100 + 1/100 + 1/100)
1/N_E = 1/5(29/100)
1/N_E = (29/500)
N_E = (500/29) = 17.24 individuals
```

Notice that this number (17.24) is less than the simple average of these population sizes (85). This formula is actually a calculation of the harmonic mean of a series of numbers. The harmonic mean is affected by small outliers and is always less than the arithmetic mean of a series of numbers.

Effective Population Size With An Unbalanced Sex Ratio

Alleles will be thoroughly mixed in a randomly mating population with equal numbers of males and females. However, if the sex ratio is skewed strongly from 1:1, the allelic diversity will be limited by the rarer sex and the alleles that it is collectively carrying. If the population consists of N_M males and N_F females, the effective population size (N_E) is:

$$N_E = \frac{4N_M N_F}{N_M + N_F}$$

For example, if the population consists of 100 females and only 10 males, the effective population size is:

```
N_E = (4)(100)(10)/(100 + 10)
```

```
N_E = 4000/110 = 36.4 individuals
```

Notice that although the observed population size (N) is 110 individuals, the effective population size (N_E) is only 36.4, which is small enough for genetic drift to become important.

Effective Population Size With Limited Natal Dispersal

For complete mixing of aleles, an individual would need to be able to mate randomly with any other individual in its population. More realistically, an individual is much more likely to mate with neighboring individuals that are close by and much less likely to mate with individuals that are distant. Under these circumstances, the effective population size is calculated as:

$$N_E = 4\pi dx$$

where d is the population density (individuals/area), and x is the dispersal distance from where an individual is born to where it mates. With limited dispersal and/or a population that is at low density, individuals are likely to choose mates from only a limited "neighborhood" of nearby individuals. Even with random mating, the effect of this is to reduce the local genetic diversity in each of the neighborhoods. Limited dispersal introduces a kind of "viscosity" to the population that can make genetic drift important.

As a simple example, if the density of individuals is 10 per m^2, but the dispersal distance is only 1 m, the effective population size is:

```
N_E = 4*pi*10*1
```

```
N_E =125.6 individuals
```

In summary, genetic drift is an important force in changing allele frequencies when effective population sizes are less than 100, and there are a number of common features of populations (bottlenecks, biased sex ratios, and limited dispersal) that can lower N_E below this threshold.

Summary Of The Effects Of Four Potential Mechanisms Of Evolution

Mechanism	Change in Allele Frequency?	Change in Genotype Frequency?
Mutation	Yes (unlikely)	Yes (unlikely)
Migration	Yes	Yes
Non-random Mating	No (yes with recessive lethals)	Yes
Genetic Drift	Yes (if $N_E < 100$)	Yes (if $N_E < 100$)

Mechanism	Strength of Change?	Lead to Fixation?	Predictable?
Mutation	Weak	Yes (no with back mutation)	Yes
Migration	Strong	Yes	Yes
Non-random Mating	Weak	No (yes with recessive lethals)	Yes
Genetic Drift	Strong (if $N_E < 100$)	Yes (if $N_E < 100$)	No

Lecture Notes For Wednesday November 11th

Definitions

`natural selection (popular definition)` "survival of the fittest"

`natural selection (biological definition)` differential reproduction and/or survival of individuals with heritable traits

`tautology` a self-referencing definition

Assumptions of Natural Selection

1. Individuals exhibit variation in their traits
2. At least some of that variation has a heritable (= genetic) component
3. All individuals produce more offspring than can survive
4. Those individuals that survive better do so because of their traits

R Functions

Here is a collection of R functions that calculate each of the formulas we use in BCOR 102. You can paste them into R or use in the accompanying script file `BCOR_LECTURE_NOTES.R`. These are simple calculators that you can use to practice numerical problems and make sure you are doing your calculations correctly.

R Functions For Allelic & Hardy-Weinberg Calculations

1. `AlleleFreq_2A` takes as inputs the number or frequencies of 3 genotypes (AA AB BB) and returns the frequencies of the two alleles (A B)

2. `AlleleFreq_3A` takes as inputs the number or frequencies of 6 genotypes (JJ JK KK KL KK LL) and returns the frequencies of the three alleles (J K L)

3. `HardyWeinberg_2A` takes as inputs the frequencies of two alleles (A B) and returns the frequencies of the three genotypes at equilibrium (AA AB BB).

4. `HardyWeinberg_3A` takes as inputs the frequencies of two alleles (J K L) and returns the frequencies of the three genotypes at equilibrium (JJ JK KK KL KK LL).

```
# FUNCTION to calculate observed allele frequencies or a single gene with 2
# alleles

AlleleFreq_2A <- function(x = c(AA = 100, AB = 50, BB = 50)) {

    # Pull out counts of individual genotypes
    AA <- x[1]
    AB <- x[2]
    BB <- x[3]
    # Create a vector and divide by the sum; works for frequencies or raw counts
    # as input
    Gen_Freq <- c(AA, AB, BB)/sum(AA + AB + BB)

    # Print genotype frequencies
    cat("Observed genotypic frequencies:", "\n", "freq(AA) = ", Gen_Freq[1],
        "\n", "freq(AB) = ", Gen_Freq[2], "\n", "freq(BB) = ", Gen_Freq[3],
        "\n")
    cat("\n")

    # Create vector for allele frequencies
    Allele_Freq <- vector("numeric", 2)

    # Use genotypes to calculate allele frequencies
    Allele_Freq[1] <- Gen_Freq[1] + 0.5 * Gen_Freq[2]
    Allele_Freq[2] <- Gen_Freq[3] + 0.5 * Gen_Freq[2]

    # Print allelic frequencies
    cat("Observed allelic frequencies:", "\n", "freq(A) = ", Allele_Freq[1],
        "\n", "freq(B) = ", Allele_Freq[2], "\n")
    cat("\n")
    # Return the output vector
    return(Allele_Freq)
}
```

```
# FUNCTION to calculate observed allele frequencies or a single gene with 3
# alleles

AlleleFreq_3A <- function(x = c(JJ = 100, JK = 50, JL = 50, KL = 50, KK = 50,
    LL = 50)) {

    # Convert input vector to individual genotypes
    JJ <- x[1]
    JK <- x[2]
    JL <- x[3]
    KL <- x[4]
```

```
    KK <- x[5]
    LL <- x[6]
    # Create a vector and divide by the sum; works for frequencies or raw counts
    # as input
    Gen_Freq <- c(JJ, JK, JL, KL, KK, LL)/sum(JJ, JK, JL, KL, KK, LL)

    # Print genotype frequencies
    cat("Observed genotypic frequencies:", "\n", "freq(JJ) = ", Gen_Freq[1],
        "\n", "freq(JK) = ", Gen_Freq[2], "\n", "freq(JL) = ", Gen_Freq[3],
        "\n", "freq(KL) = ", Gen_Freq[4], "\n", "freq(KK) = ", Gen_Freq[5],
        "\n", "freq(LL) = ", Gen_Freq[6], "\n")
    cat("\n")

    # Create vector for allele frequencies
    Allele_Freq <- vector("numeric", 3)

    # Use genotypes to calculate allele frequencies
    Allele_Freq[1] <- Gen_Freq[1] + 0.5 * Gen_Freq[2] + 0.5 * Gen_Freq[3]
    Allele_Freq[2] <- Gen_Freq[5] + 0.5 * Gen_Freq[2] + 0.5 * Gen_Freq[4]
    Allele_Freq[3] <- Gen_Freq[6] + 0.5 * Gen_Freq[3] + 0.5 * Gen_Freq[4]

    # Print allelic frequencies
    cat("Observed allelic frequencies:", "\n", "freq(J) = ", Allele_Freq[1],
        "\n", "freq(K) = ", Allele_Freq[2], "\n", "freq(L) = ", Allele_Freq[3],
        "\n")
    cat("\n")

    # Return the output vector
    return(Allele_Freq)
}
```

```
# FUNCTION to calculate Hardy-Weinberg genotypic frequency for a single gene
# with 2 alleles

HardyWeinberg_2A <- function(x = c(p = 0.7, q = 0.3)) {

    # Convert input vector to individual frequencies
    p <- x[1]
    q <- x[2]
    # Create a vector for genotypic frequencies
    Genotype_Freq <- vector("numeric", 3)

    # Use Hardy-Weinberg equation to calculate genotypic frequencies from
    # allelic frequencies
    Genotype_Freq[1] <- p^2
    Genotype_Freq[2] <- 2 * p * q
    Genotype_Freq[3] <- q^2

    # Print allelic frequencies
    cat("Observed allelic frequencies:", "\n", "f(A) = ", p, "\n", "f(B) = ",
        q, "\n")
    cat("\n")
```

```
    # Print expected Hardy-Weinberg genotypic frequencies
    cat("Expected Hardy-Weinberg genotypic frequencies:", "\n", "H-W f(AA) = ",
        Genotype_Freq[1], "\n", "H-W f(AB) = ", Genotype_Freq[2], "\n", "H-W f(BB) = ",
        Genotype_Freq[3], "\n")
    cat("\n")

    # Return the output vector
    return(Genotype_Freq)
}
```

```
# FUNCTION to calculate Hardy-Weinberg genotypic frequency for a single gene
# with 3 alleles

HardyWeinberg_3A <- function(x = c(p = 0.7, q = 0.2, r = 0.1)) {

    # Convert input vector into individual allelic frequencies
    p <- x[1]
    q <- x[2]
    r <- x[3]
    # Create a vector for genotypic frequencies
    Genotype_Freq <- vector("numeric", 6)

    # Use Hardy-Weinberg equation to calculate genotypic frequencies from
    # allelic frequencies
    Genotype_Freq[1] <- p^2
    Genotype_Freq[2] <- 2 * p * q
    Genotype_Freq[3] <- 2 * p * r
    Genotype_Freq[4] <- 2 * q * r
    Genotype_Freq[5] <- q^2
    Genotype_Freq[6] <- r^2

    # Print allelic frequencies
    cat("Observed allelic frequencies:", "\n", "f(J) = ", p, "\n", "f(K) = ",
        q, "\n", "f(L) = ", r, "\n")
    cat("\n")

    # i Print expected Hardy-Weinberg genotypic frequencies
    cat("Expected Hardy-Weinberg genotypic frequencies:", "\n", "H-W f(JJ) = ",
        Genotype_Freq[1], "\n", "H-W f(JK) = ", Genotype_Freq[2], "\n", "H-W f(JL) = ",
        Genotype_Freq[3], "\n", "H-W f(KL) = ", Genotype_Freq[4], "\n", "H-W f(KK) = ",
        Genotype_Freq[5], "\n", "H-W f(LL) = ", Genotype_Freq[6], "\n")
    cat("\n")

    # Return the output vector
    return(Genotype_Freq)
}
```

Applying The Functions To Sample Data

For the two-allele example, we started with these data:

Genotype	AA	AB	BB	Sum
Number of individuals	75	20	100	200

Genotype	AA	AB	BB	Sum

First, we use `AlleleFreq_2A` to get the initial genotypic and allelic frequencies:

```
AlleleFreq_2A(x = c(AA = 75, AB = 20, BB = 100))
```

```
## Observed genotypic frequencies:
##  freq(AA) =  0.3846154
##  freq(AB) =  0.1025641
##  freq(BB) =  0.5128205
##
## Observed allelic frequencies:
##  freq(A) =  0.4358974
##  freq(B) =  0.5641026
```

```
## [1] 0.4358974 0.5641026
```

Next we use the calculated allelic frequencies to plug into `HardyWeinberg_2A` to get the expected genotypic frequencies for Hardy-Weinberg equilibrium:

```
HardyWeinberg_2A(x = c(p = 0.4358974, q = 0.5641026))
```

```
## Observed allelic frequencies:
##  f(A) =  0.4358974
##  f(B) =  0.5641026
##
## Expected Hardy-Weinberg genotypic frequencies:
##  H-W f(AA) =  0.1900065
##  H-W f(AB) =  0.4917817
##  H-W f(BB) =  0.3182117
```

```
## [1] 0.1900065 0.4917817 0.3182117
```

For the three-allele example, we started with these data:

Genotype	JJ	JK	JL	KL	KK	LL	Sum
Number Of Individuals	10	11	0	9	2	22	54

First, we use `AlleleFreq_3A` to get the initial genotypic and allelic frequencies:

```
AlleleFreq_3A(x = c(JJ = 10, JK = 11, JL = 0, KL = 9, KK = 2, LL = 22))
```

```
## Observed genotypic frequencies:
##  freq(JJ) =  0.1851852
##  freq(JK) =  0.2037037
##  freq(JL) =  0
##  freq(KL) =  0.1666667
##  freq(KK) =  0.03703704
```

```
##  freq(LL) =  0.4074074
##
## Observed allelic frequencies:
##  freq(J) =  0.287037
##  freq(K) =  0.2222222
##  freq(L) =  0.4907407

## [1] 0.2870370 0.2222222 0.4907407
```

Next we use the calculated allelic frequencies to plug into `HardyWeinberg_3A` to get the expected genotypic frequencies for Hardy-Weinberg equilibrium:

```
HardyWeinberg_3A(x = c(p = 0.287037, q = 0.2222222, r = 0.4907407))
```

```
## Observed allelic frequencies:
##  f(J) =  0.287037
##  f(K) =  0.2222222
##  f(L) =  0.4907407
##
## Expected Hardy-Weinberg genotypic frequencies:
##  H-W f(JJ) =  0.08239024
##  H-W f(JK) =  0.127572
##  H-W f(JL) =  0.2817215
##  H-W f(KL) =  0.218107
##  H-W f(KK) =  0.04938271
##  H-W f(LL) =  0.2408264

## [1] 0.08239024 0.12757199 0.28172148 0.21810696 0.04938271 0.24082643
```

To do this more elegantly and take advantage of the full power of R, we can chain these two functions together, so that the output from the allele frequency calculation forms the input for the Hardy-Weinberg calculation:

```
# Chaining functions together to get allelic frquencies and Hardy-Weinberg
# expected genotypic frequencies:

HardyWeinberg_2A(AlleleFreq_2A(x = c(AA = 75, AB = 20, BB = 100)))
```

```
## Observed genotypic frequencies:
##  freq(AA) =  0.3846154
##  freq(AB) =  0.1025641
##  freq(BB) =  0.5128205
##
## Observed allelic frequencies:
##  freq(A) =  0.4358974
##  freq(B) =  0.5641026
##
## Observed allelic frequencies:
##  f(A) =  0.4358974
##  f(B) =  0.5641026
##
## Expected Hardy-Weinberg genotypic frequencies:
##  H-W f(AA) =  0.1900066
##  H-W f(AB) =  0.4917817
##  H-W f(BB) =  0.3182117
```

```
## [1] 0.1900066 0.4917817 0.3182117
```

```
HardyWeinberg_3A(AlleleFreq_3A(x = c(JJ = 10, JK = 11, JL = 0, KL = 9, KK = 2,
    LL = 22)))
```

```
## Observed genotypic frequencies:
##  freq(JJ) =  0.1851852
##  freq(JK) =  0.2037037
##  freq(JL) =  0
##  freq(KL) =  0.1666667
##  freq(KK) =  0.03703704
##  freq(LL) =  0.4074074
##
## Observed allelic frequencies:
##  freq(J) =  0.287037
##  freq(K) =  0.2222222
##  freq(L) =  0.4907407
##
## Observed allelic frequencies:
##  f(J) =  0.287037
##  f(K) =  0.2222222
##  f(L) =  0.4907407
##
## Expected Hardy-Weinberg genotypic frequencies:
##  H-W f(JJ) =  0.08239026
##  H-W f(JK) =  0.127572
##  H-W f(JL) =  0.2817215
##  H-W f(KL) =  0.218107
##  H-W f(KK) =  0.04938272
##  H-W f(LL) =  0.2408265
```

```
## [1] 0.08239026 0.12757202 0.28172154 0.21810700 0.04938272 0.24082647
```

R Function For Calculation Of Allele Frequency With Mutation

Here is a short function that takes as input the initial frequency of the mutant allele, the mutation rate, and a vector of times. It returns a vector of the frequency of the mutant allele at each time point, which can then be used in a simple plot.

```
# FUNCTION to calculate the increase in the frequency of a mutant allele
# through time

Mutation <- function(qo = 0.5, u = 1e-06, t = 1:10) {
    qt = 1 - (1 - qo) * exp(-u * t)
    return(qt)
}
```

```
Mutation(qo = 0.5)
```

```
##  [1] 0.5000005 0.5000010 0.5000015 0.5000020 0.5000025 0.5000030 0.5000035
##  [8] 0.5000040 0.5000045 0.5000050
```

R Function For Calculation Of Changes In Allele Frequency From Migration

Here is a function that takes as input the initial frequency of the allele in the resident population, the frequency of the allele in the migrant population, the fraction of the population each generation that consists of migrants, and the number of time steps from one generation to the next. The output is the frequency of the allele in the resident population at each time step.

```
# FUNCTION to calculate the change in allele frequency from migration

Migration <- function(p0 = 0.5, pm = 0.9, m = 0.1, t = 1:10) {
    pt <- (1 - m)^t * (p0 - pm) + pm
    return(pt)
}
```

```
Migration(p0 = 0.1)
```

```
##  [1] 0.1800000 0.2520000 0.3168000 0.3751200 0.4276080 0.4748472 0.5173625
##  [8] 0.5556262 0.5900636 0.6210572
```

R Function For Calculation of Genotype Frequencies With Inbreeding

Here is a function that takes as input the initial frequency of one of the two alleles and the inbreeding coefficient F. The output is the expected frequency of the three genotypes with inbreeding.

```
# FUNCTION to calculate the change in allele frequency from inbreeding

Inbreeding <- function(p = 0.3, F = 0.5) {
    genotypes <- vector("numeric", 3)
    q <- 1 - p
    genotypes[1] <- p^2 * (1 - F) + p * F
    genotypes[2] <- 2 * p * q * (1 - F)
    genotypes[3] <- q^2 * (1 - F) + q * F

    return(genotypes)
}
```

```
Inbreeding()
```

```
## [1] 0.195 0.210 0.595
```

R Function For Calculation Of Effective Population Size With A Bottleneck

Here is a function that takes as input a series of sequential population sizes. The output is the effective population size, which in this case is the harmonic mean of the population sizes.

```
# FUNCTION to calculate effective population size with a bottleneck

Bottleneck <- function(N = 1:5) {
    Ne <- 1/((1/length(N)) * (sum(1/N)))
    return(Ne)
}
```

```
Bottleneck()
```

```
## [1] 2.189781
```

R Function For Calculation Of Effective Population Size With A Skewed Sex Ratio

Here is a function that takes as input the number of males (m) and females (f) in the population and returns the effective population size.

```
# FUNCTION to calculate effective population size with a skewed sex ratio

SexRatio <- function(m = 10, f = 12) {
    Ne <- (4 * m * f)/(m + f)
    return(Ne)
}
```

```
SexRatio()
```

```
## [1] 21.81818
```

R Function For Calculation Of Effective Population Size With Limited Dispersal

Here is a function that takes as input the population density (d) and the dispersal distance (x) and returns the effective population size.

```
# FUNCTION to calculate effective population size with limited dispersal

NatalDispersal <- function(d = 10, x = 1) {
    Ne <- 4 * pi * d * x
    return(Ne)
}
```

```
NatalDispersal()
```

```
## [1] 125.6637
```

R Function For Calculation Of Occurrence Probability With Multiple Trials

Here is a function that takes as input the probability p of a single event and the number of independent trials n. It returns the probability of at least one event occurring among the set of n trials.

```
# FUNCTION to calculate probability of at least one occurrence with
# individual probability p and number of trials n

CompoundProb <- function(p = 0.01, n = 52) {
    Prob <- 1 - (1 - p)^n
    return(Prob)
}
```

```r
CompoundProb()
```

```
## [1] 0.4070336
```