4 Inbreeding at a Single Locus

4.1 Inbred Strains

The genome of a random animal from an outbred population is polymorphic. To clarify this statement we note that in general the DNA molecule of a given chromosome of some specie are almost identical to each other. Hence, if we select two such molecules, be it the homologous copies of the same individual or a copy from each of a pair of individuals, we will find that the sequential composition of the base-pairs is essentially identical. However, differences between the two copies may emerge at some points. For example, the base-pair composition at a given locus for one of the copies may be different from the base-pair composition of the other copy at the same locus. The two variants are called *alleles*. Typically, one will not find more than two distinct alleles at the same locus. (Although, in principle, four different alleles may be observed.) If both alleles are frequent enough in the population, say the minor allele frequency is more than 1%, then the locus is considered to be a *Single Nucleotide Polymorphism* (SNP). Millions of such SNPs were mapped to date in the human genome. Millions more are being mapped in the context of genome projects of many other species. Other types of polymorphism, not SNPs, are present but are less frequent. The genetic variability among individuals has a major contribution to the variability in phenotype.

In order to investigate biological properties of a given specie it is convenient to homogenize the genetic background. For that propose special inbred strains are created. Such strains have the property that all members of the strain have identical copies of their DNA. Outside of the scientific context, inbred strains are known as pure breeds and are popular between dogs and cats breeders.

Genetically homogeneous inbred strains are created by a process of successive brother–sister mating. Random drift in finite inbred populations eventually results in the fixation of a given locus, namely in the extinction of all other alleles. Once a locus is not polymorphic, it remains so in all subsequent generations. With additional brother–sister mating, the genomes in the population become less and less polymorphic. The formal definition of an inbred strain requires at least 20 generations of strict brother–sister mating. Some of the classical inbred strains have a history of more than 100 generations of inbreeding.

In order to understand the dynamics of inbreeding, in the following two subsections we will investigate the process of forming an inbred strain via the process of selfing and strict brother-sister mating. In the next subsection, we take a closer look at a mathematical analysis of inbreeding.

4.2 Inbreeding via Selfing

There are known examples of diploid organisms that have the ability of self reproduction that involves meiosis. Most notable are some species of plants, for example Arabidopsis. Male and female germ cells are produced by the organism and merge to form offsprings. The genetic decomposition of an offspring need not be identical to that of the parent since the process of reproduction involves random sampling from the genetic material of the parent.

In this subsection we will investigate the dynamics associated with the process of repeated selfing. Starting from a founder plant a new plant is formed via self reproduction. This plant self-reproduces to form the next generation and so on. Of interest will be the genetic decomposition of random offsprings in each generation and how this decomposition varies from generation to generation.

Initially we will attempt to see what woould be the final outcome of the process of self reproduction at a given locus. We assume that the locus is on an autosome and is polymorphic with two alleles: “A” and “a”. The founder plant is sampled from an heterogenous population. At the investigated locus it may be of one of the three possible genotypes: “AA”, “Aa”, or “aa”.

Consider the selection of 10 independent copies of the chromosome locus, assuming that the frequencies in the population of the two alleles are 0.3 and 0.7, respectively:

```
> n.rep <- 10
> allele <- c("A","a")
> p.allele <- c(0.3,0.7)
> sample(allele,n.rep,rep=TRUE,prob=p.allele)
 [1] "a" "a" "a" "a" "A" "A" "a" "A" "A" "a"
```

(Read the help file of the function `sample`.)

A founder planet is composed of two copies of the locus. For the sack of this investigation we will assume that these two copies, one inherited from the father of the founder and the other from the mother of the founder, are independent.

```
> pat <- sample(allele,n.rep,rep=TRUE,prob=p.allele)
> mat <- sample(allele,n.rep,rep=TRUE,prob=p.allele)
> pat
 [1] "a" "a" "a" "a" "a" "A" "a" "a" "a" "a"
> mat
 [1] "A" "a" "A" "a" "A" "A" "a" "a" "a" "a"
> plant <- list(pat=pat,mat=mat)
> plant
$pat
 [1] "a" "a" "a" "a" "a" "A" "a" "a" "a" "a"

$mat
 [1] "A" "a" "A" "a" "A" "A" "a" "a" "a" "a"
```

A list is a special type of vector that does not restrict its components to be of the same type. They can be any type of object, and may vary from component to component. As we see in the example, the components may be refered to by name.

We use the function "`meiosis`" that we wrote before in order to produce gametes. The function is applied twice, once to produce the gamete of the male germ cell and then to produce the gamete of the female germ cell. We wrap the process of self-reproduction of an offspring plant in the function "`selfin`":

```
> selfing <- function(plant)
+ {
+     pat <- meiosis(plant$pat,plant$mat)
+     mat <- meiosis(plant$pat,plant$mat)
+     return(list(pat=pat, mat=mat))
+ }
```

Let us repeat the process of self-reproduction for 100 generation and see what type of plants it produces:

```
> for(g in 1:100) plant <- selfing(plant)
> plant
$pat
 [1] "A" "a" "A" "a" "a" "A" "a" "a" "a" "a"

$mat
 [1] "A" "a" "A" "a" "a" "A" "a" "a" "a" "a"
```

As we can see in this example after 100 generations there were no heterozyguous plants with the "Aa" genotype, only homozygotes. In order to varify that this is not a random finding let us increase the sample size of the simulation to include 100,000 independent repeates of the process of relf reproduction over 100 generations:

```
> n.rep <- 10^5
> pat <- sample(allele,n.rep,rep=TRUE,prob=p.allele)
> mat <- sample(allele,n.rep,rep=TRUE,prob=p.allele)
> plant <- list(pat=pat,mat=mat)
> table(plant$pat,plant$mat)

        a     A
  a 48973 21269
  A 20815  8943
```

As we expect, among founders the distribution of genotypes is random, with approximately $0.49 = 0.7^2$ homozygote of type "aa", $0.09 = 0.3^2$ homozygote of type "AA" and $0.42 = 2 \times 0.7 \times 0.3$ heterozygote plants.

```
> for(g in 1:100) plant <- selfing(plant)
> table(plant$pat,plant$mat)

        a     A
  a 69988     0
  A     0 30012
```

After 100 generations only homozyote plants are present. In 70% of the lines they are of type "aa" and in the other 30% they are of type "AA".

Homework Question 4.1. *Plot the frequency of homozygote as a function of the number of generations of self fertilization. Compute and save the frequency in each iteration of the "*`for`*" loop. Use the function "*`plot`*" in order to make the plot.*

We now turn to a mathematical analysis of inbreeding. Consider a self-fertilizing plant and a locus at which that plant may be heterozygous. We are interested in the probability, H_g, that it is still heterozygous after g generations of self-fertilization. We have assumed that $H_1 = 2p(1 - p)$, for $p = 0.3$. The key to our analysis is to write H_g in terms of H_{g-1}. The equation is $H_g = \frac{1}{2}H_{g-1}$, which follows from the observation that in order for the plant to be heterozygous after g generations it must be heterozygous after $g - 1$ (because we are neglecting the possibility of mutation) and whichever allele at the given locus is in the egg, the other allele must occur in the sperm. This latter event occurs with probability 1/2. By iterating this basic relation, we have $H_g = \frac{1}{2}H_{g-1} = (\frac{1}{2})^2 H_{g-2} = \cdots (\frac{1}{2})^{g-1} H_1$.

Homework Question 4.2. *Compute the theoretical probability of heterozygosity and add it as a line to the plot. You may use the function "*`line`*" to add the line.*

We compute the probability of heterozygosity in each of the 20 generations of self-fertilization and add the line to the plot:

```
> H <- 0.5^(1:20)
> lines(H,col=gray(0.5))
```

The decay is also exponential. Observe the good agreement between the simulations and the theoretical computations. The function "`lines`" is a low-level plotting function that adds lines to existing plots.

4.3 Brother-Sister Inbreeding

As we saw, when considering inbreeding by selfing, that by the 20th generation of inbreeding there is a very large probability of fixation. Our goal in this subsection is to understand the dynamics of that probability when mating is involved. As a simple example we consider a minimal population of size two, which contains one male and one female. The size of the population is maintained in each generation by selecting one brother and one sister of the current generation as parents for the next generation. We start by simulating such populations and recording the probability of heterozygosity in each generation. For the simulation we use the function "`cross`" that was written in the previous section:

```
> pat <- sample(allele,n.rep,rep=TRUE,prob=p.allele)
> mat <- sample(allele,n.rep,rep=TRUE,prob=p.allele)
> mo <- list(pat=pat,mat=mat)
> pat <- sample(allele,n.rep,rep=TRUE,prob=p.allele)
> mat <- sample(allele,n.rep,rep=TRUE,prob=p.allele)
> fa <- list(pat=pat,mat=mat)
> Fn.fa <- cross(fa,mo)
> Fn.mo <- cross(fa,mo)
> p.bs <- mean(c(Fn.fa$pat != Fn.fa$mat,Fn.mo$pat != Fn.mo$mat))
> for (g in 2:20)
+ {
+     New.fa <- cross(Fn.fa,Fn.mo)
+      New.mo <- cross(Fn.fa,Fn.mo)
+      Fn.fa <- New.fa; Fn.mo <- New.mo
+      p.bs[g] <- mean(c(Fn.fa$pat != Fn.fa$mat,Fn.mo$pat != Fn.mo$mat))
+ }
> points(p.bs,col="blue")
```

Note that the expression "`Fn.fa$pat != Fn.fa$mat`" produces a logical vector ("`TRUE`" if the animal is heterozygous and "`FALSE`" otherwise). The same holds true for the other expression, which is concatenated by the function "`c`" to form a single logical vector. The function "`mean`" expects a numerical vector as an input and computes its average value. Before computing the mean, the logical vector is converted into a numerical vector by setting "`TRUE`" equal to one and "`FALSE`" equal to zero. The result is the relative frequency of "`TRUE`"s, i.e., the relative frequency of heterozygous mice.

We add the simulated probabilities to the plot as *b*lue points with the aid of the low-level function "`points`". Observe that the probability of heterozygosity decays exponentially as a function of the number of brother–sister mating generations. The dynamics of the decay, however, is very different from the dynamics in selfing.

One can analyze the probability of heterozygosity in brother–sister mating using the same tools as in the case of self-fertilization. In this case, however, one must consider heterozygosity two generations back as well. It turns out that the recursive formula for heterozygosity is given by the equation:

$$H_g = \frac{1}{2}H_{g-1} + \frac{1}{4}H_{g-2}\ . \qquad (1)$$

Homework Question 4.3. *Prove the recursion (1), compute the sequence of probabilities, and add them to the plot.*

Instead of proving the given recursion we apply an approach that is analogous to the argument given above for selfing. It tracks the genotypes in the population in each generation and uses techniques based on Markov chains for the computation of probabilities. Although requiring more mathematical analysis than (1), it appears to be relatively easily adapted to deal with more complex problems.

The population is of size two in each generation. Each mouse in the population may either be in state 0, 1, or 2, depending on its genotype. Denote by `0-0` and `2-2` the states where both mice are homozygous for the same allele, and note that these are *absorbing states.* Once such a state is reached, the process will remain indefinitely in the same state. The other states, denoted here by 1-0, 2-0, 1-1, and 2-1 are *transient*. Fixation occurs when the process reaches an absorbing state. At initiation a distribution is set for all states. Given the state of the population at a given generation, the distribution over the states in the next generation can be described by a transition probability matrix. The distribution in generation g is given by multiplying the distribution in generation $g-1$ by the probability transition matrix.

Denote by Q the sub-matrix of transition probabilities between transient states and let π_0' be a row vector giving an initial distribution over these states. We are interested in the case

$$\pi_0' = (4 \cdot 0.3 \cdot 0.7^3, 2 \cdot 0.3^2 \cdot 0.7^2, 4 \cdot 0.3^2 \cdot 0.7^2, 4 \cdot 0.3^3 \cdot 0.7) \ ,$$

which corresponds to starting distribution over transient states which is obtained by sampling the two parents from the population independently. The distribution over the transient states after g generations is the vector $\pi_0' Q^g$, where Q^g corresponds to the multiplication of the matrix Q by itself g times. The probability of heterozygosity after g generations is $\pi_0' Q^g v$, where $v' = (1/2, 0, 1, 1/2)$, since $1/2$ of the animals in the state 1-0 are heterozygous, none of the animals in 2-0 is heterozygous, etc. In order to make the analysis more concrete let us put the computer to work:

```
> hetero.states <- paste(c(1,2,1,2),"-",c(0,0,1,1),sep="")
> hetero.states
[1] "1-0" "2-0" "1-1" "2-1"
> Q <- matrix(0,4,4)
> Q
     [,1] [,2] [,3] [,4]
[1,]    0    0    0    0
[2,]    0    0    0    0
[3,]    0    0    0    0
[4,]    0    0    0    0
> rownames(Q) <- colnames(Q) <- hetero.states

> Q["1-0",c("1-0","1-1")] <- c(0.5,0.25)
> Q["2-0","1-1"] <- 1
> Q["1-1",] <- c(0.25,0.125,0.25,0.25)
> Q["2-1",c("1-1","2-1")] <- c(0.25,0.5)
> Q
     1-0   2-0  1-1  2-1
```

```
1-0 0.50 0.000 0.25 0.00
2-0 0.00 0.000 1.00 0.00
1-1 0.25 0.125 0.25 0.25
2-1 0.00 0.000 0.25 0.50
```

Let us explain: `Q` is an object of the class "`matrix`". It was created with the aid of the function "`matrix`". Elements of the matrix are identified by the double indexing format "`[,]`", left for rows and right for columns. Submatrices can be assigned by appropriate selection of indices, with rules that follow similar patterns as the rules used for indexing vectors. As in vectors, names can be used for indexing. The first row of the matrix `Q` corresponds to the probabilities of transition from the state `1-0` to any of the four transient states. For example, if one parent has the genotype `0` and the other the genotype `1`, then there is a 50% chance that one of the offspring will have genotype `0` and the other genotype `1`. There is a 25% chance that both will have the genotype `1` and a 25% that both will have the genotype `0`. The last case produces an absorbing state. This computation gives rise to the first row of the matrix `Q`. Similar computations will lead to the other rows of the matrix.

```
> initial <- c(4*0.3*0.7^3,2*0.3^2*0.7^2,4*0.3^2*0.7^2,4*0.3^3*0.7)
> het.count <- c(0.5,0,1,0.5)
> hetero.prob <- NULL
> QQ <- Q
> for (g in 1:20)
+ {
+     hetero.prob[g] <- initial %*% QQ %*% het.count
+     QQ <- QQ %*% Q
+ }
> lines(hetero.prob,col="blue")
> legend(13.1,0.42,
+     legend=c("selfing","brother-sister"),
+     lty=c(1,1),col=c(1,4))
```

The vector "`initial`" gives the starting state; it was denoted by π_0 above. The vector "`het.count`", denoted above by v, is used in order to compute the probability of heterozygosity from the distribution of transient states. The vector "`hetero.prob`" stores the computed probabilities. Initially this vector is assigned a null value. This creates an object that can be identified and manipulated by `R`, but contains no information. Information is accumulated in each iteration of the "`for`" loop. Note that the final length of “hetero.prob” need not be preassigned. The system automatically expands the object upon request. Finally, observe that the binary operation “%*%” corresponds to matrix multiplication, unlike the operation “*”, which is applied term-by-term. (Try both “Q * Q” and “Q %*% Q” to see the difference between the two operations.)

The results of the theoretical computation are added to the plot in with the aid of the function “lines”. Note the good agreement between the simulation and the theoretical computation. For clarity, a legend is added with the low-level plotting function “legend”. The first two arguments of the function are used in order to determine the location of the upper-left corner of the legend box in the figure; and the other arguments set the text, the type, and the color of the lines to appear inside the box. The line type must be provided. In the current case the first and the third lines are of the same type: “lty = 1”, which corresponds to a solid line. A broken line, for example, corresponds to type 2. Observe that since there are two lines in the legend, the argument should be a vector of length two.

Homework Question 4.4. *Describe the dynamics of heterozygosity in brother-sister mating if the process is initiated by crossing together two inbred strains, that are homozygote within each population but are heterozygous between populations.*

Homework Question 4.5. *Investigate the dynamics of heterozygosity when a new generation is created by mating a mouse from the given population with an mouse from one of the two inbred strains.*

4.4 Kimura’s Probabilistic Approach

Attempting to extend the method of inheritance of the composition of genotypes through the generations will turn out to be very difficult. The number of states of the associated Markov process increases very rapidly as a function of the number of loci considered, the number of distinct allele, or the number of subjects in the population. Symmetries between states can be exploited in order to reduce complexity. Still, keeping track of all possibilities become too tedious even in relatively simple models. A solution to the problem was found by Motoo Kimura in his paper *A probability method for treating inbreeding systems, especially with linked genes.* In this paper Kimura proposed to keep track only of probabilities of carefully selected events in the population instead of the entire genotypic composition of the population. This allows for the more direct exploitation of symmetries of models and reduces the complexity dramatically. We will demonstrate in this subsection Kimura's approach for the determination of the dynamics of heterozygosity reduction of multi-allelic loci in larger populations, with and without self fertilization. In the next section the approach will be used in order to determine the recombination rate following inbreeding.

Let us remove the assumption that the number of alleles is two at most and denote the distinct alleles at a given locus by $A_1, \ldots, A_d$. An exact model for mating is needed when the population size is more than one in the case of selfing and more then two in the case of sexual reproduction. We will use the model of random mating, in which paring of two gametes is independent of their genotypes.

Start with a population of plants that can self fertilize. Take the population size N to be fixed and consider an autosomal locus. At each generation a plant is a result of selfing with probability $1/N$ and is a result of mating with probability $1 - 1/N$. Let us define two types of evens regarding population t: (i) Two homologous copies of a random individual share the same allele, and (ii) Two random copies from two distinct individuals share the same allele. We denote the probability of the first event by I_t and the probability of the second event by J_t. Note that the two events satisfy the recursion:

$$I_t = \frac{1}{N}(0.5 + 0.5 I_{t-1}) + \big(1 - \frac{1}{N}\big) J_{t-1} \tag{2}$$

$$J_t = \frac{1}{N}(0.5 + 0.5 I_{t-1}) + \big(1 - \frac{1}{N}\big) J_{t-1}\ , \tag{3}$$

for $t \geq 1$. It follows that for $t \geq 2$:

$$I_t = \frac{1}{N}(0.5 + 0.5 I_{t-1}) + \big(1 - \frac{1}{N}\big) I_{t-1}$$

and, upon substituting $H_t = 1 - I_t$,

$$H_t = \big(1 - \frac{1}{2N}\big) H_{t-1} = \big(1 - \frac{1}{2N}\big)^{t-1} H_1\ .$$

Observe that when $N = 1$ we get again the relation we obtained previously.

Homework Question 4.6. *Compute the probabilities I_0 and J_0 in an outbred population. Give conditions under which $I_0 = J_0$.*

Homework Question 4.7. *Plot the dynamics of the reduction of heterozygosity for various population sizes.*

Consider next a population with two distinct sexes. Assume that the population is composed of N_{M} and N_{F} and we consider again the two events and their probabilities. In the development of the recursions we need to include the generations of grand parents, since the two copies of a random individual may not originate from the same parent as in the case of selfing but may originate from the same grandparent. The probability that this is the case, the parallel of the probability $1/N$ in the case of selfing, is given by

$$\frac{1}{N_e} = \frac{1}{4}\big(\frac{1}{N_{\mathrm{M}}} + \frac{1}{N_{\mathrm{F}}}\big) .$$

The relations we obtain for this case are:

$$\begin{aligned} I_t &= \frac{1}{N_e}(0.5 + 0.5 I_{t-2}) + \big(1 - \frac{1}{N_e}\big)J_{t-2} && (4) \\ I_{t-1} &= J_{t-2} , && (5) \end{aligned}$$

hence

$$I_t = \frac{1}{2N_e} + \big(1 - \frac{1}{N_e}\big)I_{t-1} + \frac{1}{2N_e}I_{t-2} .$$

The resulting relation for proportion of heterozygosity is

$$H_t = \big(1 - \frac{1}{N_e}\big)H_{t-1} + \frac{1}{2N_e}H_{t-2} .$$

Again, when $N_{\mathrm{F}} = N_{\mathrm{M}} = 1$ this relation produces the relation we observed before.

Homework Question 4.8. *Plot the dynamics of the reduction of heterozygosity for various values for N_F and N_M. Consider, in particular, the case where $N_M = 1$ but N_F varies.*

Homework Question 4.9. *(More difficult) Write an R code that generates the process of breeding in populations of constant size and random mating. Compare between the results of simulations based on this code and the theoretical derivation.*