BIOB480/BIOE548 notes 9/05/2024

Introduction

- Complete Chi-Squared test example and discussion
- Homework questions and HWP review
- Bolnick app introduction (https://bolnicklab.shinyapps.io/HWEdemo/)
- What is an equilibrium?
- Intro to physical linkage

Linkage Disequilibrium

Hardy-Weinberg Proportions give us expectations for the frequency of genotypes at a locus. But what should we expect between different alleles at two independently assorting loci? This is the basis of concept of **linkage disequilibrium**, typically measured by the metric D.

Consider diallelic loci A and B, with alleles A_1 and A_2 and B_1 and B_2. How often should we expect the gamete (or haplotype) A_1B_1 versus A_1B_2 versus A_2B_2 versus A_1B_1? If these loci are completely independent—say, on different chromosomes—we expect a given combination to be the product of its frequencies:

$$f(A_1B_1) = f(A_1) * f(B_1)$$

To measure D, we compare this expected, *unlinked* haplotype frequency to the actual data. For example, imagine we survey a population of N=8 individuals and count the following haplotypes (at 2N = 16 chromosomes): 6 A_1B_1s (= 6/16 = 0.375), 2 A_1B_2s (= 2/16 = 0.125), 6 A_2B_2s (= 6/16 = 0.375), 2 A_2B_1s (= 2/16 = 0.125). Based on these data, we know that the overall frequency of A_1 is $f(A_1) = 0.5$. Necessarily, $f(A_2) = 0.5$. Likewise, the frequency of B_1 is 8/16=0.5 and $f(B_2)$=0.5.

D is simply the difference between the observed frequency of a particular haplotype and our expectation of it:

$$D(A_1B_1) = f(A_1B_1) - f(A_1)f(B_1) = 0.375 - (0.5 * 0.5) = 0.125$$

Contrast this with a hypothetical situation in which all four possible haplotypes are found at equal frequencies (i.e. $f(A_1B_1) = f(A_1B_2) = f(A_2B) = f(A_2B_1) = 0.25$):

$$D(A_1B_1) = f(A_1B_1) - f(A_1)f(B_1) = 0.25 - (0.5 * 0.5) = 0$$

This comparison tells us two things. First, a positive D value means that the two alleles at different loci used to calculate its value are occuring together *more often* than we would expect if assortment is truly independent. A negative value would therefore indicate two alleles occur together *less often* than we would expect. Second, a value of $D = 0$ is our expectation if there is no linkage disequilibrium.

Importantly, we will get the same *absolute* value of D no matter which pair of alleles we choose:

$$D(A_1B_1) = f(A_1B_1) - f(A_1)f(B_1) = 0.375 - (0.5 * 0.5) = 0.125$$

$$D(A_1B_2) = f(A_1B_2) - f(A_1)f(B_2) = 0.125 - (0.5 * 0.5) = -0.125$$
$$D(A_2B_2) = f(A_2B_2) - f(A_2)f(B_2) = 0.375 - (0.5 * 0.5) = 0.125$$
$$D(A_2B_1) = f(A_2B_1) - f(A_2)f(B_1) = 0.125 - (0.5 * 0.5) = -0.125$$

It follows that the maximum value of D is $|0.25|$ (when a pair of alleles are *always* or *never* inherited together).

So what causes this non-random assortment of alleles? Physical linkage—the close proximity of two loci on the same chromosome—is one possibility. But LD can also be caused by interactions between genes, natural selection, population genetic structure, and demographic history. It's therefore a bad name for a complicate phenomenon (in a field full of bad names)—neither solely due to linkage, nor really a disequilibrium (frequencies of haplotypes can be stable through time).

LD Decay

Recombination between loci will break down LD through time in most cases, a relationship described by $D^t = D_0(1 - c)^t$, where D^t is linkage disequilibrium at generation t, D_0 is the initial value of linkage disequilibrium, and c is the recombination rate. (c is usually expressed as centimorgans per megabase, where 1 centimorgan is equivalent a 1% chance two loci on a chromosome will become separated from one another as a result of recombination during meiosis. A megabase is simply one million nucleotide bases.)

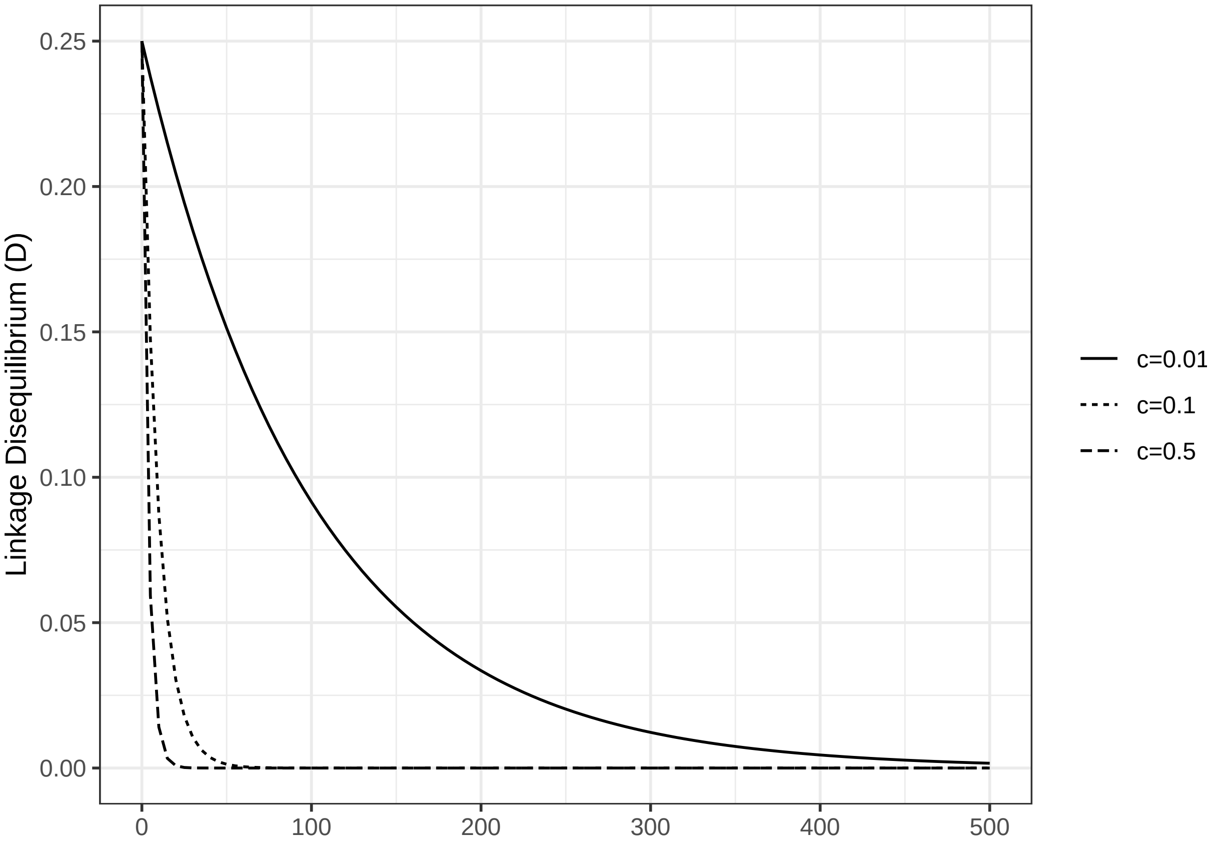
We can plot this relationship for a set of arbitrary recombination rate values as follows:

```r
library(ggplot2)

# assign different values of c and D_0
c1 <- 0.01
c2 <- 0.10
c3 <- 0.25
D_0 = 0.25

# assign functions for each value of c
ld_1 <- function(x) D_0*(1 - c1)^x
ld_2 <- function(x) D_0*(1 - c2)^x
ld_3 <- function(x) D_0*(1 - c3)^x

# generate plot
p1 <- ggplot() +
  theme_bw() +
  theme(legend.title = element_blank())+
  stat_function(fun = ld_1, aes(linetype="c=0.01")) +
  stat_function(fun = ld_2, aes(linetype="c=0.1")) +
  stat_function(fun = ld_3, aes(linetype="c=0.5")) +
  scale_linetype_discrete() +
  ylim(0,0.25) +
  xlim(0, 500) +
  ylab("Linkage Disequilibrium (D)")
p1
```



LDsim activity

CJ Battey's LDSim is a useful application to understand linakge disequilibrium more intutively: https://cjbattey.shinyapps.io/LDsim/. Instructions for a group activity using the app can be found in `05_slides.pdf`.