Analyzing the genetic structure of populations: a Bayesian approach

Introduction

Our review of Nei's G_{st} and Weir and Cockerham's θ illustrated two important principles:

1. It's essential to distinguish *parameters* from *estimates*. *Parameters* are the things we're really interested in, but since we always have to make inferences about the things we're really interested in from limited data, we have to rely on *estimates* of those parameters.

2. This means that we have to identify the possible sources of sampling error in our estimates and to find ways of accounting for them. In the particular case of Wright's F-statistics we saw that, there are two sources of sampling error: the error associated with sampling only some individuals from a larger universe of individuals within populations (*statistical sampling*) and the error associated with sampling only some populations from a larger universe of populations (*genetic sampling*).[1]

It shouldn't come as any surprise that there is a Bayesian way to do what I've just described. As I hope to convince you, there are some real advantages associated with doing so.

The Bayesian model

I'm not going to provide all of the gory details on the Bayesian model. In fact, I'm only going to describe two pieces of the model.[2] First, a little notation:

$$\begin{aligned} n_{11,i} &= \#\text{ of } A_1A_1 \text{ genotypes} \\ n_{12,i} &= \#\text{ of } A_1A_2 \text{ genotypes} \\ n_{22,i} &= \#\text{ of } A_2A_2 \text{ genotypes} \end{aligned}$$

[1] The terms "statistical sampling" and "genetic sampling" are due to Weir [4].

[2] The good news is that to do the Bayesian analyses you don't have to write any code. All you have to do is download an `R` package in a slightly strange way, but we'll get to that.

$$
\begin{aligned}
i &= \text{population index} \\
I &= \text{number of populations}
\end{aligned}
$$

These are the data we have to work with. The corresponding genotype frequencies are

$$
\begin{aligned}
x_{11,i} &= p_i^2 + f p_i (1 - p_i) \\
x_{12,i} &= 2 p_i (1 - p_i)(1 - f) \\
x_{22,i} &= (1 - p_i)^2 + f p_i (1 - p_i)
\end{aligned}
$$

So we can express the likelihood of our sample as a product of multinomial probabilities

$$
P(\mathbf{n}|\mathbf{p}, f) \propto \prod_{i=1}^{I} x_{11,i}^{n_{11,i}} x_{12,i}^{n_{12,i}} x_{22,i}^{n_{22,i}} \quad .
$$

Notice that I am assuming here that we have the same f in every population. It's easy enough to relax that assumption, but we won't worry about it for now.

The next step is to describe how allele frequencies are distributed among populations. I'll leave out the details, but broadly speaking all we do is to define a probability distribution

$$
P\left(\mathbf{p}|\bar{\mathbf{p}}, \theta\right) \quad ,
$$

where $\bar{\mathbf{p}}$ is the average allele frequency across populations, and θ is F_{ST}.[3] To complete the Bayesian model, all we need are some appropriate priors. We'll discuss them a little later, but we can now write down the complete model as

$$
P(f, \theta|\bar{\mathbf{p}}, \theta, f) \propto P(\mathbf{n}|\mathbf{p}, f) P(\mathbf{p}|\bar{\mathbf{p}}, \theta) P(\bar{\mathbf{p}}) P(\theta) P(f) \quad .
$$

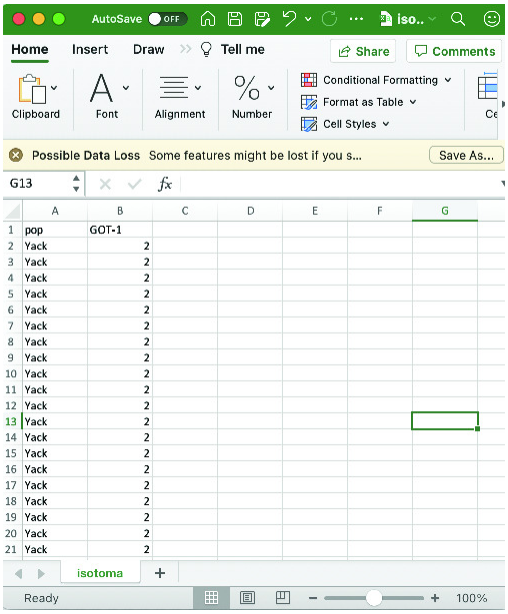
Using `Hickory` to analyze F-statistics

As I said earlier, the good news is that you don't have to write any code to run an analysis of F-statistics using a Bayesian approach. All you have to do is to download and install the package `Hickory` in `R`. Doing this isn't quite as simple as typing `install.packages("Hickory")` in `R`, but it's not too much worse.[4]

```
install.packages("devtools")
install.packages(c("bayesplot", "rstan", "tidyverse"))
devtools::install_github("kholsinger/Hickory", build_vignettes = TRUE)
```

[3] I call it θ rather than F_{ST}, because this parameter is conceptually equivalent to Weir and Cockerham's θ even though I use a different method to estimate it.

[4] One of these days I'll get around to cleaning `Hickory` up a bit more and submit it to `CRAN`. Then installing it will be as simple as `install.packages("Hickory")`



	A	B
1	pop	GOT-1
2	Yack	2
3	Yack	2
4	Yack	2
5	Yack	2
6	Yack	2
7	Yack	2
8	Yack	2
9	Yack	2
10	Yack	2
11	Yack	2
12	Yack	2
13	Yack	2
14	Yack	2
15	Yack	2
16	Yack	2
17	Yack	2
18	Yack	2
19	Yack	2
20	Yack	2
21	Yack	2

Figure 1: Selected rows of a `isotoma.csv` with data from *Isotoma petraea* [1].

Getting data into Hickory

Now you're ready to read in the data. In this case, we're going to start with the *Isotomoa petrea* example. Download the data from `http://darwin.eeb.uconn.edu/eeb348-resources/isotoma.csv`, open it up in your favorite spreadsheet editor, and you should see something similar to Figure 1.

The first row is a header row that describes the data in the columns. The first column has the heading `pop`, which indicates that the elements in the column refer to the population from which an individual was collected. The second column has the heading `GOT-1`, which indicates that this column contains the genotype of an individual at the `GOT-1` locus. Each row after the first is the genotype of one individual. I used 2 for A_1A_1, 1 for A_1A_2, and 0 for A_2A_2. I could have swapped the numbers for the two homozygotes, but the heterozygote must be given the genotype 1.

Now load `Hickory` and the `tidyverse` and take a quick look at a more complicated data set before we continue with the *Isotoma petraea* example.

```
library("Hickory")
library("tidyverse")
dat <- read_csv(system.file("extdata", "protea_repens.csv", package = "Hickory"))
```

	pop	TP113	TP154	TP178	TP188	TP210	TP231	TP243	TP249	TP260	TP297	TP
1	ALC	0	0	.	0	0	.	2	0	1	.	.
2	ALC	0	0	.	0	0	.	.	0	1	.	.
3	ALC	0	0	.	0	1	.	.	0	1	1	1
4	ALC	0	0	.	1	1	.	2	2	0	1	1
5	ALC	1	0	.	0	0	.	2	0	0	.	.
6	ALC	0	0	2	.	1	.	1	1	1	.	1
7	ALC	1	0	.	0	1	.	2	0	1	2	1
8	ALC	0	1	.	0	1	.	2	2	0	.	.
9	ALC	0	0	.	0	0	.	2	2	0	.	.
10	ALC	0	1	.	0	0	.	2	2	0	.	.
11	ALC	0	0	.	0	0	.	.	2	0	.	.
12	ALC	0	.	.	0	2	.	.	0	0	.	1
13	ANY	1	0	.	1	1	.	0	1	1	2	1
14	ANY	0	0	.	0	1	0	.	2	1	.	0
15	ANY	0	0	1	1	1	1	1	1	1	1	1
16	ANY	0	0	.	1	1	0	0	1	0	2	1
17	ANY	1	1	.	1	1	.	1	1	0	2	1
18	ANY	1	0	.	0	1	0	1	0	0	1	1
19	ANY	0	.	.	1	1	.	0	2	0	.	1
20	ANY	0	.	2	2	0	.	0	0	0	0	.
21	ANY	0	0			0			1	0		

Showing 1 to 21 of 663 entries, 21 total columns

Figure 2: Selected rows of a a data set from *Protea repens* that is distributed with `Hickory` [2].

```
view(dat)
```

Here you'll see the `pop` column again and columns for the genotype of individuals at 20 different loci (Figure 2). For now just notice how every individual has been genotyped at a number of loci, and that there are missing data (denoted by '.') for some combinations of individuals and loci.

Now that you understand something about the format of the data that `Hickory` needs, let's load it into `R` for further analysis.

```
genos <- read_marker_data("isotoma.csv")
```

Running the analysis

Now that the data are loaded, running the analysis is very straightforward.

```
fit <- analyze_codominant(genos)
```

The results are pretty easy to interpret, too.

```
Inference for Stan model: analyze_codominant.
4 chains, each with iter=2000; warmup=1000; thin=1;
post-warmup draws per chain=1000, total post-warmup draws=4000.

          mean se_mean    sd     2.5%      25%      50%      75%    97.5% n_eff  Rhat
f        0.344   0.002 0.101    0.147    0.276    0.345    0.414    0.538  3539 1.000
theta    0.075   0.001 0.046    0.017    0.042    0.065    0.096    0.197  1223 1.002
lp__  -121.464   0.150 4.035 -130.283 -124.022 -121.095 -118.549 -114.516   727 1.003

Samples were drawn using NUTS(diag_e) at Sat Jul  3 16:07:48 2021.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

The column labeled `mean` is the posterior mean for the parameter listed in the first column.[5] The column labeled `se_mean` is the standard error of the mean. It's a measure of how accurate the estimate of the posterior mean is, and we want it to be very small relative to the estimate of the posterior mean. The column labeled `sd` is the standard deviation of the posterior mean. It's a measure of our uncertainty about the mean. We expect about 95% of the posterior probability to lie within 2 standard deviations of the mean. If we compare the 2.5% and 97.5% quantiles,[6] they are very close to what we expect.

In short, there appears to be a reasonable amount of inbreeding within populations ($f = 0.344$) and a small to moderate amount of among population differentiation ($\theta = 0.075$). In contrast to the Weir and Cockerham method, we also have estimates of uncertainty associated with both f and θ.[7] Since you've probably forgotten what the other estimates look like, Table 1 compares all of the approaches we've considered.

The logic behind `Hickory` matches the logic behind Weir & Cockerham. With moderate to large sample sizes, the point estimates are reasonably close. They're somewhat different here because there is only one locus in the sample and because the sample sizes in some of the populations are very small. Notice, however, that the `Hickory` and the Weir & Cockerham estimates are similar in one very important respect. The estimate of F_{ST} is much smaller in them than in Nei's method or the direct method because they take account of genetic sampling, not just statistical sampling.

[5] Don't worry about `lp__` for the time being.

[6] Corresponding to 95% of the posterior probabiity.

[7] It's not too difficult to get estimates of uncertainty usins the Weir and Cockerham approach, but it takes some additional work.

Method	F_{is}	F_{st}
Direct	0.137	0.214
Nei	0.309	0.240
Weir & Cockerham	0.540	0.039
`Hickory`	0.344	0.075

Table 1: Comparison of different approaches for estimating population structure from genetic data.

Thinking about priors

When I introduced the Bayesian model I reminded you that we need to specify priors to complete it, so how did I get away without specifying any priors in the analysis we just completed? Because `Hickory` picks priors by default when you don't specify them. It picks priors for f and θ such that there's a 95% chance that they lie between 0.01 and 0.2. That makes sense for many organisms, since many of them are outbreeding and have low to moderate amounts of population differentiation. If we have a fair amount of data, that choice won't make much difference. What about here?

Instead of starting our analysis thinking that we have a reasonably good idea of what f and θ ought to be, let's suppose we don't have much of an idea at all. In particular, let's imagine that all we're willing to say is that there's a 95% chance that f and θ lie between 0.1 and 0.9. How do we incorporate that into the analysis?

```
fit <- analyze_codominant(genos, prior_f = list(lower = 0.1, upper = 0.9),
                          prior_theta = list(lower = 0.1, upper = 0.9))
```

As you can see, the results are quite different from those we got before.

```
Inference for Stan model: analyze_codominant.
4 chains, each with iter=2000; warmup=1000; thin=1;
post-warmup draws per chain=1000, total post-warmup draws=4000.

          mean se_mean    sd     2.5%      25%      50%      75%    97.5% n_eff  Rhat
f        0.523   0.001 0.094    0.331    0.460    0.525    0.590    0.693  4027 0.999
theta    0.263   0.005 0.131    0.068    0.163    0.243    0.342    0.571   724 1.005
lp__  -122.391   0.184 4.255 -131.796 -125.070 -121.939 -119.311 -115.408   533 1.007

Samples were drawn using NUTS(diag_e) at Sat Jul  3 16:43:51 2021.
For each parameter, n_eff is a crude measure of effective sample size,
```

```
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

The posterior mean of f is now 0.523 rather than 0.344, and the posterior mean of θ is now 0.263 rather than 0.075. If you're following along, you're probably asking yourself "Which of those estimates should I believe?" My advice is that you shouldn't believe either of them very much. Remember what a Bayesian model looks like.

$$\mathrm{P}(\phi|x) = \frac{\mathrm{P}(x|\phi)\mathrm{P}(\phi)}{\mathrm{P}(x)}$$

We get the posterior mean from the posterior distribution, $\mathrm{P}(\phi|x)$. If the posterior mean changes substantially based on different choices for the prior, $\mathrm{P}(\phi)$, it means that we don't have enough data for the likelihood, $\mathrm{P}(x|\phi)$ to dominate the result. In simpler terms, if different choices for the prior lead to markedly different conclusions, our confidence in those conclusions depends heavily on our prior beliefs, not just the data we've collected. Unless we have a lot of confidence in our prior beliefs, we shouldn't have much confidence in the conclusions.

One of the nice things about a Bayesian approach is that it gives us a straightforward way to assess how much to rely on inferences from the data we've collected. If different priors have a large influence on the posterior, as they do here, it tells us that the data we've collected don't have much information about the parameters we're interested in. If different priors don't have a large influence, then the data *do* have a fair amount of information about the parameters.[8]

There's a general lesson here: Think carefully about the prior distribution on the parameters in *any* Bayesian model you use, *and* consider exploring at least a couple of different choices of priors to see if they have a large influence on your conclusions. In addition, *pay attention to the credible intervals*. In both sets of analyses you've just seen, the credible intervals are very wide. That in itself says that the data aren't giving you a very clear idea of what the parameter is.

Assessing evidence for inbreeding and population differentiation

You've already seen that `Hickory` gives you estimates of the credible intervals for f and θ, but if you're interested in seeing whether there is evidence for inbreeding within populations or for genetic differentiation among populations, you can't just look to see whether the

[8] If it seems as if I'm repeating myself, I probably am, but I think this is a really immportant point that bears repeating.

credible intervals overlap 0. Why? because f and θ are defined to lie between 0 and 1 in `Hickory` so they can't overlap 0.[9] In some data sets the posterior mean for either or both may be substantially larger than 0, and the lower bound of the credible interval may also be substantially larger than 0. In such cases, you'd be pretty safe saying that you have evidence for inbreeding or geographical differentiation, but what if you have a situation like what you get from using the *Protea repens* data set that is distributed with `Hickory`.

```
genos <- read_marker_data(system.file("extdata", "protea_repens.csv", package = "Hickory"))
fit <- analyze_codominant(genos)

Inference for Stan model: analyze_codominant.
4 chains, each with iter=2000; warmup=1000; thin=1;
post-warmup draws per chain=1000, total post-warmup draws=4000.

           mean se_mean     sd      2.5%       25%       50%       75%     97.5% n_eff  Rhat
f         0.005   0.000  0.002     0.002     0.003     0.005     0.006     0.010  5169 0.999
theta     0.081   0.000  0.008     0.066     0.075     0.081     0.086     0.097  1599 1.001
lp__  -6242.725   0.538 17.416 -6276.812 -6254.305 -6242.781 -6230.561 -6209.327  1048 1.005

Samples were drawn using NUTS(diag_e) at Sun Jul  4 12:52:05 2021.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

The posterior means and credible intervals in these data are relatively insensitive to our choice of priors.[10] The posterior mean for θ is only 0.081, but the lower bound of the 95% credible interval is 0.066 and the credible interval is quite small, which gives us reasonably strong evidence that $\theta > 0$, i.e., that there is genetic differentiation among populations. But what about inbreeding within populations? The posterior mean of f is only 0.005, and the lower bound of the 95% credible interval is barely greater than 0, i.e., 0.002. That doesn't seem like very good evidence either way, but can we say something more?[11]

We could simply do Hardy-Weinberg tests at every locus in every population, but that could get pretty tedious. If we did that, we'd also run into problemms with multiple tests,

[9] We noted a couple of lectures ago that f can be negative when it's understood as a measure of departure from Hardy-Weinber, but for computational reasons, `Hickory` restricts it to $[0, 1]$. If you're interested in the gory details of why, feel free to ask me.

[10] Don't take my word for it. Run the analysis yourself with the second set of priors we used above or with another set of priors that strikes your fancy and compare the results.

[11] Would I be asking this question if the answer were "No"?

which inconvenient to deal with. We'll take a different approach. Namely, we'll compare the model we just fit with one that *assumes* there is no inbreeding within populations, i.e., $f = 0$. The criterion we'll use to compare the models is something known as the expected log predictive density [3]. That's a mouthful, and the mathematics is reasonably complicated, but it's easy enough to interpret the results without understanding all of those details.

First, we run the model in which we assume $f = 0$ and store the result in a different object.

```
fit_f0 <- analyze_codominant(genos, f_zero = TRUE)
```

```
Inference for Stan model: analyze_codominant.
4 chains, each with iter=2000; warmup=1000; thin=1;
post-warmup draws per chain=1000, total post-warmup draws=4000.

           mean se_mean     sd      2.5%       25%       50%       75%     97.5% n_eff
f         0.000     NaN  0.000     0.000     0.000     0.000     0.000     0.000   NaN
theta     0.081   0.000  0.008     0.067     0.076     0.080     0.086     0.097  1934 1
lp__  -6234.006   0.518 16.882 -6267.340 -6245.573 -6233.842 -6222.445 -6202.017  1060 1

Samples were drawn using NUTS(diag_e) at Sun Jul  4 13:21:39 2021.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

Notice that the estimate for f is 0, as expected. Now we can commpare the two models using `loo()`.[12]

```
loo_free <- loo(fit)
loo_f0 <- loo(f0)
compare(loo_free, loo_f0)
```

You'll see some warning messages when you run `loo()` with these data. In an ideal world, we'd do things a bit differently and get rid of them, but in this case, we don't need to worry about them. Let's focus on the table that was printed.

```
       elpd_diff se_diff
model2  0.0       0.0
model1 -1.8       3.4
```

[12] I call the first object `loo_free` because in that model f was free to vary.

Model	A_1A_1	A_1A_2	A_2A_2
$f = 0$	0.25	0.50	0.25
$f = 0.005$	0.25125	0.4975	0.25125
$f = 0.010$	0.2525	0.495	0.2525

Table 2: Comparison of genotype frequencies assuming $p = 0.5$ with $f = 0$ and f as estimated (mean and upper bound of the 95% credible interval) from the *Protea repens* data distributed with `Hickory`.

The column labeled `elpd_diff` is the difference in expected log predictive density between the model with the highest elpd and the model on the line in which the entry appears. The column labeled `se_diff` is the standard error of that difference. `model 2` refers to the second model named in the call to `compare()`, i.e., the $f = 0$ model (`loo_f0`), and `model 1` refers to the first model named in the call to `compare()`, i.e., the f "free" model. What all of this means is that

- Model 2, the $f = 0$ model, is more strongly supported than Model 2, the model in which we estimated f, *but*
- The difference between the two models, -1.8, is substantially less than twice the standard error of the difference (3.4), meaning that we don't have good evidence that one model is better than the other.

You may find it dissatisfying that we can't distinguish between these two models and that we're left saying that we don't know whether we have evidence for inbreeding within populations or not, but remember, our estimate of f is only 0.005. Table 2 shows what that means for expected genotype frequencies if $p = 0.5$. As you can see, the difference in genotype frequencies is extremely small. It's hard to believe that there would be any biologically meaningful difference between any of the scenarios that seem compatible with the data.

Extending the model

It is relatively straightforward to extend the basic model above to account for the possibility that the amount of differentiation at some loci is much greater (or much smaller) than it is as most loci. Similarly, it is relatively straightfoward to extend the model to allow some populations to be much more similar to (or much more different from) the average

population allele frequencies than others. All we need to do is to allow θ to reflect locus- and population-specific effects.

$$\begin{aligned} \theta_i &= \text{locus-specific } \theta \\ \theta_k &= \text{population-specific } \theta \\ \theta_{ik} &= \text{population- and locus-specific } \theta \\ &= (\theta_i + \theta_k)/2 \quad . \end{aligned}$$

You can read more about estimating locus- and population-specific effects in the documentation for `Hickory` if you're interested.

References

[1] S H James, A P Wylie, M S Johnson, S A Carstairs, and G A Simpson. Complex hybridity in *Isotoma petraea* V. Allozyme variation and the pursuit of hybridity. *Heredity*, 51(3):653–663, 1983.

[2] Rachel Prunier, Melis Akman, Colin T. Kremer, Nicola Aitken, Aaron Chuah, Justin Borevitz, and Kent E. Holsinger. Isolation by distance and isolation by environment contribute to population differentiation in *Protea repens* (Proteaceae L.), a widespread south african species. *American Journal of Botany*, 104(5):674–684, 2017.

[3] Aki Vehtari, Andrew Gelman, and Jonah Gabry. Practical bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and Computing*, 27(5):1413–1432, 2017.

[4] B S Weir. *Genetic Data Analysis II*. Sinauer Associates, Sunderland, MA, 1996.

Creative Commons License