Rapture selection tests progress

March 7, 2016

1 Design of the array

We designed the array last December based on SNP calls from 360 full RAD libraries. Originally, the array was aimed at allow Menna's team to determine pedigrees, and they also wanted lots of immune genes. Several sets of criteria were combined.

SNPs with high genotyping rates and MAF:

- SNPs genotyped in $\geq 1/2$ the individuals
- MAF ≥ 0.05
- SNPs dropped so that there is at least 20,000 bp between SNPs
- SNPs ≥ 50 bp away from any other non-singleton SNP
- ≤ 4 SNPs per RAD locus

SNPs near immune system genes:

- SNPs genotyped in $\geq 1/3$ the individuals
- No singletons
- ≤ 50 kb from an immune gene
- ≤ 2 non-singleton SNPs within 50 bp and
- ≤ 5 non-singleton SNPs on the same RAD locus
- The immune SNPs were arrived at by searching the annotations for keywords associated with immune function

SNPs that showed up in a preliminary GWAS run:

- top 1,000 SNPs in the preliminary run and
- ≤ 2 non-singleton SNPs within 50 bp and
- ≤ 5 non-singleton SNPs on the same RAD locus

Some of the criteria were meant to exclude loci with high diversity that might prevent probe binding. For the first set of criteria, I was trying to get a set of SNPs that would be useful for making pedigrees. We ended targeting 15,898 loci.

2 Data processing

The steps involved in processing the data:

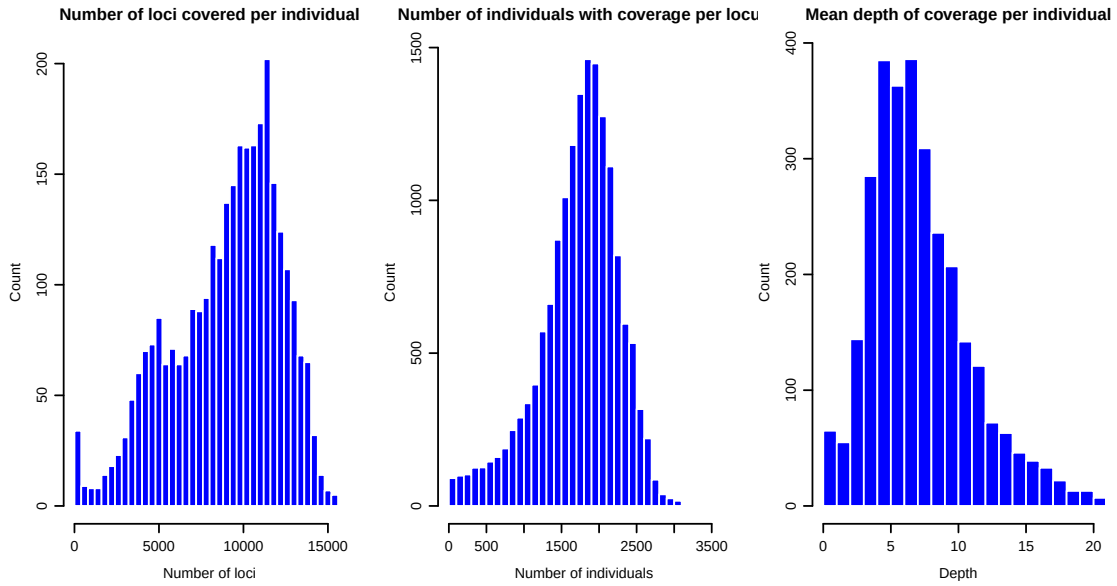
1. Flip reads so that the barcodes are all on the "forward" read
2. Run process_radtags from Stacks to de-multiplex and to remove low quality reads
3. Remove PCR duplicates using clone_filter from Stacks; default settings; this removed about 50% of the reads
4. Align to reference genome with bowtie2
5. Merge alignments from duplicate libraries
6. Filter reads:
    - Exclude reads that either did not align to a targeted cut site or are not the mate of a read that aligns to a targeted cut site
    - Exclude reads for which the mate does not map to the same scaffold

    TODO: Do another quick check on the final step - just make sure you didn't mess up the locations of cut sites or leave out reads that should be included / keep reads that shouldn't be included

3 Coverage and samples

We ran five NextSeq lanes with 3,630 libraries from 3,568 samples. Of these, 503 were pouch young, which I did not analyze and 10 were quolls, which I also did not include in any analyses.

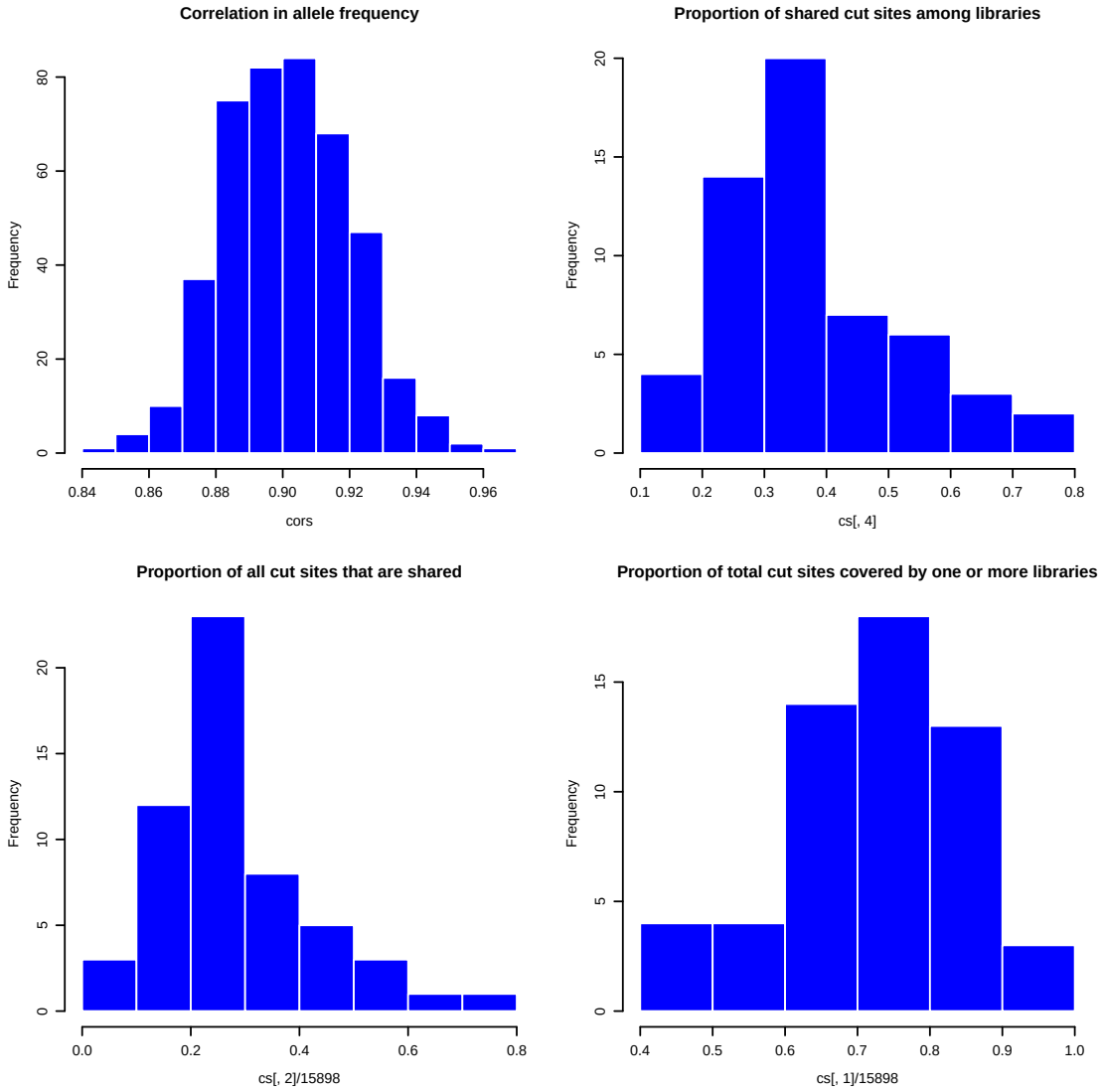
After filtering, the distribution of coverage per locus for samples that we retained looked like:



There is a wide distribution of coverage, and there are a few samples with very little coverage (although all samples had some coverage). I did not explicitly remove any samples from the analysis because ANGSD should enable us to deal with low coverage.

Repeatability

We had more than one library for several samples, so I looked at the repeatability of loci (do you get the same loci if you sequence the same sample multiple times) and of allele frequency estimates. You've seen these figures before:



This indicates that missing data is mainly due to random variation in coverage, rather than allelic dropout, and we could probably fill in the missing data with more coverage. Allele frequency estimates are fairly robust to the missing data too.

4 Sampling dates

4.1 Assignment of individuals to years

I assigned individuals to years based on the time the DNA sample was collected. If we had more than one DNA sample, I used all the time points (e.g. if we had a sample from 2006 and 2007 for an individual, I included that individual among both the 2006 and 2007 individuals). This was straightforward, and was the only way to do things before I had the database. There's probably an

argument to be made for doing this differently: we could count an individual in every year it was observed, or we could take the year it was born, etc..

4.2 Before / after DFTD infection

The first years DFTD was detected in each population:

Forestier	2004
Fentonbury	2005
Freycinet	2001
Mt William	1996
Narawntapu	2007
W Pencil Pine	2006
Woolnorth	never

4.3 Number of samples from each year and population:

47	Fentonbury	2004
52	Fentonbury	2005
58	Fentonbury	2006
63	Fentonbury	2007
37	Fentonbury	2008
11	Fentonbury	2009

131	Forestier	2004
46	Forestier	2005
168	Forestier	2006
98	Forestier	2007
93	Forestier	2008
107	Forestier	2009
13	Forestier	2010
26	Forestier	2012
1	Forestier	2013

107	Freycinet	1999
122	Freycinet	2000
71	Freycinet	2001
65	Freycinet	2002
38	Freycinet	2003
60	Freycinet	2004
54	Freycinet	2005
27	Freycinet	2006
32	Freycinet	2007
21	Freycinet	2008
7	Freycinet	2009
10	Freycinet	2010
10	Freycinet	2011
18	Freycinet	2012
12	Freycinet	2013
26	Freycinet	2014

11	$Mt_{William}$	2004
28	$Mt_{William}$	2005
20	$Mt_{William}$	2006
21	$Mt_{William}$	2007
17	$Mt_{William}$	2008
14	$Mt_{William}$	2009
8	$Mt_{William}$	2010
15	$Mt_{William}$	2011
4	$Mt_{William}$	2012
10	$Mt_{William}$	2013
8	$Mt_{William}$	2014
33	Narawntapu	1999
0	Narawntapu	2000
9	Narawntapu	2003
46	Narawntapu	2004
27	Narawntapu	2005
64	Narawntapu	2006
45	Narawntapu	2007
63	Narawntapu	2008
30	Narawntapu	2009
27	Narawntapu	2010
15	Narawntapu	2011
22	Narawntapu	2012
47	Woolnorth	2006
62	Woolnorth	2007
19	Woolnorth	2008
64	Woolnorth	2009
154	Woolnorth	2010
146	Woolnorth	2012

5 ANGSD run

ANGSD (Analysis of Next-Generation Sequencing Data) is a collection of programs for estimating genotype likelihoods/probabilities, allele frequencies, and various summary statistics without making individual genotype calls. It seems pretty flexible, but that flexibility also means there are multiple ways to achieve the same result. I used ANGSD to print out allele frequency estimates for each position in each dataset.

I ran ANGSD separately on different datasets:

- Each population before and after DFTD detection; the "before" populations were composed of individuals collected before or during the first year DFTD was detected (e.g. Freycinet individuals collected 1999-2001, inclusive). This provided the input for the allele frequency change tests.
- On non-overlapping, two year "generations," starting with 1999, each population separately. This provided the input for spatpg - Gompert's time series selection program.
- Association testing of cases and controls (see below)

Initial plans for ANGSD

I was going to follow what I think is the full ANGSD "pipeline:"

1. Estimate genotype likelihoods
2. Estimate inbreeding coefficients for each individual using another program called ngsF
3. Estimate the sample allele frequencies and site frequency spectrum using the inbreeding coefficients to improve the estimate (rather than assuming HWE).

I ran into problems with long runtimes and some datasets that just got hung up, so for the sake of actually getting something done, I decided to simplify.

Actual ANGSD run (2016-01-15 run)

Single-pass calculation of allele frequencies and genotype likelihoods. This assumes HWE, but it finished in a reasonable amount of time. As always, there are many parameters for which different settings are possible. Here are the settings I used:

- filter out reads with MAPQ < 40 (MAPQ is bowtie2's confidence that the read is aligned to the right spot; 40 is quite high, but most reads align with high confidence)
- filter out bases with Phred score < 25
- Do an base alignment quality adjustment around indels similar to the "extended" option in samtools - this helps prevent indels from causing false SNP calls (other options are a simpler baq adjustment and no adjustment)
- Use the original GATK model for genotype likelihoods (the other option available without a set of "known" SNPs is the samtools model).
- Use the reference genome allele as the "major" allele (it can produce a minor allele frequency > 0.5 if the reference is really the minor allele).
- Leave out X chromosome loci. Our capture array included 471 X loci, so we could use them in the future if we wanted to.

6 Individual Statistics

6.1 Association test (2016-01-27 run)

I used ANGSD to do an association test comparing cases and controls. The steps were:

1. Identify cases and controls
2. Calculate genotype probabilities for the cases and controls using ANGSD
3. Do a PCA using ngsCovar, which takes the ANGSD genotype probabilities as input
4. Run the association test with ANGSD

Identification of cases and controls

I identified a set of cases and controls. All were individuals born into the cohort that seemed to be strongly affected by DFTD (large change in age structure and high infection rates). The controls were individuals caught at age three or older without signs of disease, while the cases were those that were caught before age 3 with signs of disease.

The year of birth for the first cohort strongly affected by DFTD (I used Jan 1 as the cutoff day):

Fentonbury	2004
Forestier	2005
Freycinet	2003
Mt William	2000
Narawntapu	2008
W Pencil Pine	2009

For some populations, like Narawntapu and Mt William, I just picked a year or two after the initial DFTD detection because I didn't see any published infection rate or age-structure information. I didn't have any samples from Mt William before 2004 anyway.

Example SQL code to extract cases and controls from the database:

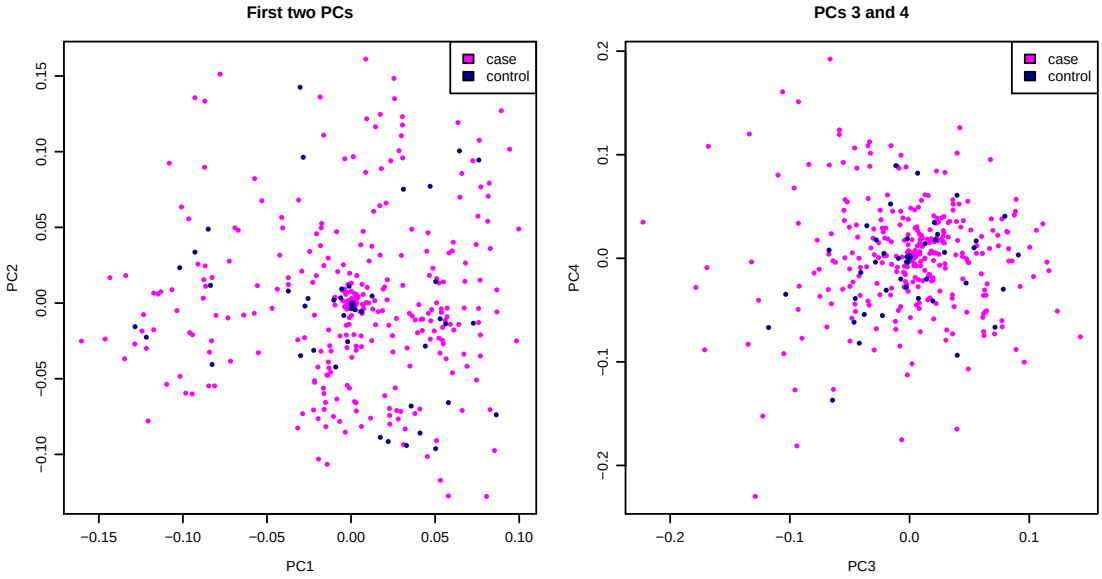
```
# Controls: more than 1000 days old (about 3 years) at collection and
# no DFTD (score < 3)
SELECT DISTINCT(f.Microchip)
    FROM fundamentals AS f
    INNER JOIN observations AS o
    ON f.Microchip = o.Microchip
    WHERE f.SiteOfFirstCapture = 'Freycinet'
    AND f.YOB > DATE('2003-01-01')
    AND DATEDIFF(o.TrappingDate, f.YOB) > 1000
    AND o.DFTDScore < 3 AND o.DFTDScore > 0;

# Cases: less than 1000 days old (about 3 years) at collection and
# clear DFTD (score > 3)
SELECT DISTINCT(f.Microchip)
    FROM fundamentals AS f
    INNER JOIN observations AS o
    ON f.Microchip = o.Microchip
    WHERE f.SiteOfFirstCapture = 'Freycinet'
    AND f.YOB > DATE('2003-01-01')
    AND DATEDIFF(o.TrappingDate, f.YOB) < 1000
    AND o.DFTDScore > 3;
```

I ended up with 45 controls and 307 cases. All populations were combined because I would not have had enough controls otherwise - I might still not really have enough controls.

Controlling for population structure

I used the ngsCovar program and kept the first two PCs to use as covariates in the association test. I'm not sure this was actually all that useful: the first two PCs only explained about 6.5% of the

variation. On the other hand, it didn't look like the cases and controls were clustering separately, in general:



Settings

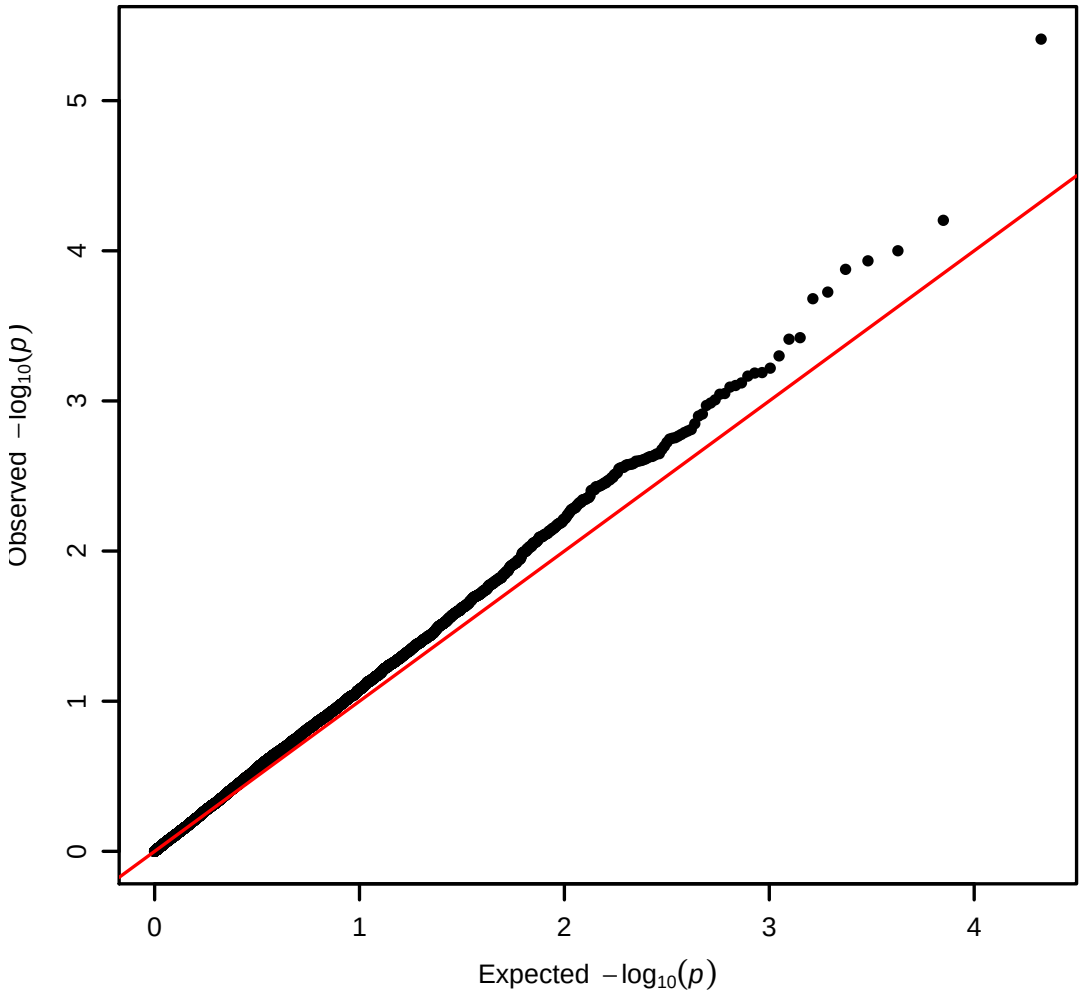
The settings are the same as for the ANGSD run to estimate allele frequencies, except that there were some additional settings:

- Only test sites where the p-value for a test of whether they are segregating is $\leq 10^{-6}$
- Minimum minor allele frequency of 0.05
- Data from ≥ 10 individuals at a site
- Use an additive model for the association
- Estimate the posterior genotype probability using the allele frequency as prior

This resulted in about 10,600 SNPs being tested

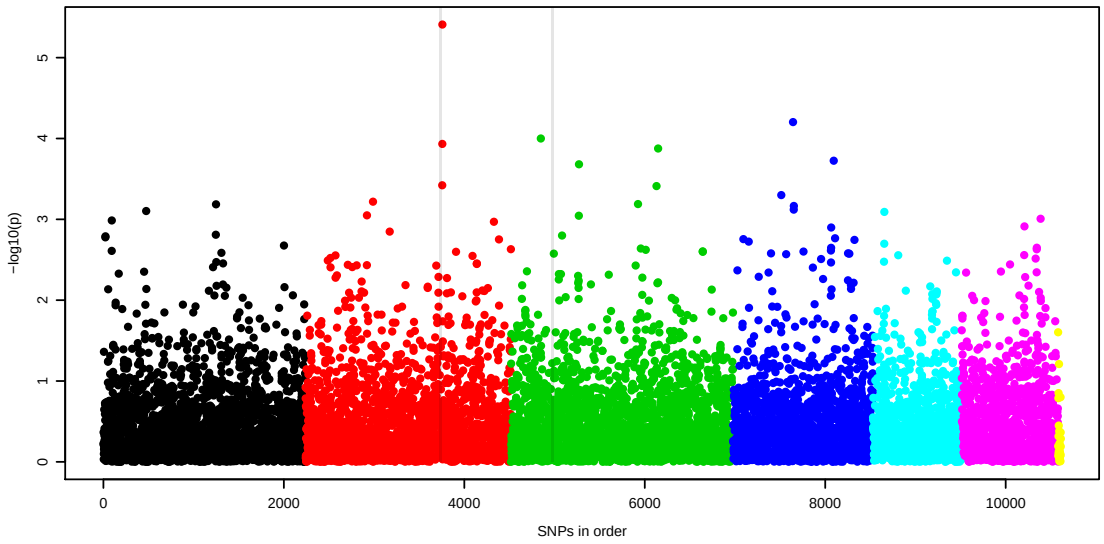
Results

First, a Q-Q plot - if the p-values are distributed as expected under the null hypothesis, there should be a straight line. I think it's generally considered not good (wrong p-values) if you don't have something close to a straight line. However, I am just ranking SNPs by strength of association, so as long as the rank order is not affected much by any problems, it's probably okay? This doesn't look



terrible:

And here is a Manhattan plot of the "p-values" (the coordinates are a little approximate - I just plotted the SNPs in order). The locations of the previously identified scaffolds are marked with gray bars. There is not a lot of precision when zoomed out this far - for example, the most significant SNP is actually on the scaffold after our chromosome 2 candidate region.



The top SNPs near the two previously identified candidate regions were about the 95th percentile ($p \approx 0.04$) and the 93rd percentile ($p \approx 0.06$) for chr. 2 and chr. 3 regions.

6.2 Change in allele frequency (2016-01-26 run)

I took the ANGSD allele frequency estimates for every position and subtracted the "before" frequency from the "after" frequency. This was done separately for each population, and is basically the same test I used for the full RAD ms, just with ANGSD allele frequency estimates.

Settings

Very similar to the association test. These criteria had to be met in either the "before" or "after" dataset and were applied to each population separately:

- MAF ≥ 0.05
- ≥ 10 individuals with data
- p-value for segregating sites $\leq 10^{-6}$

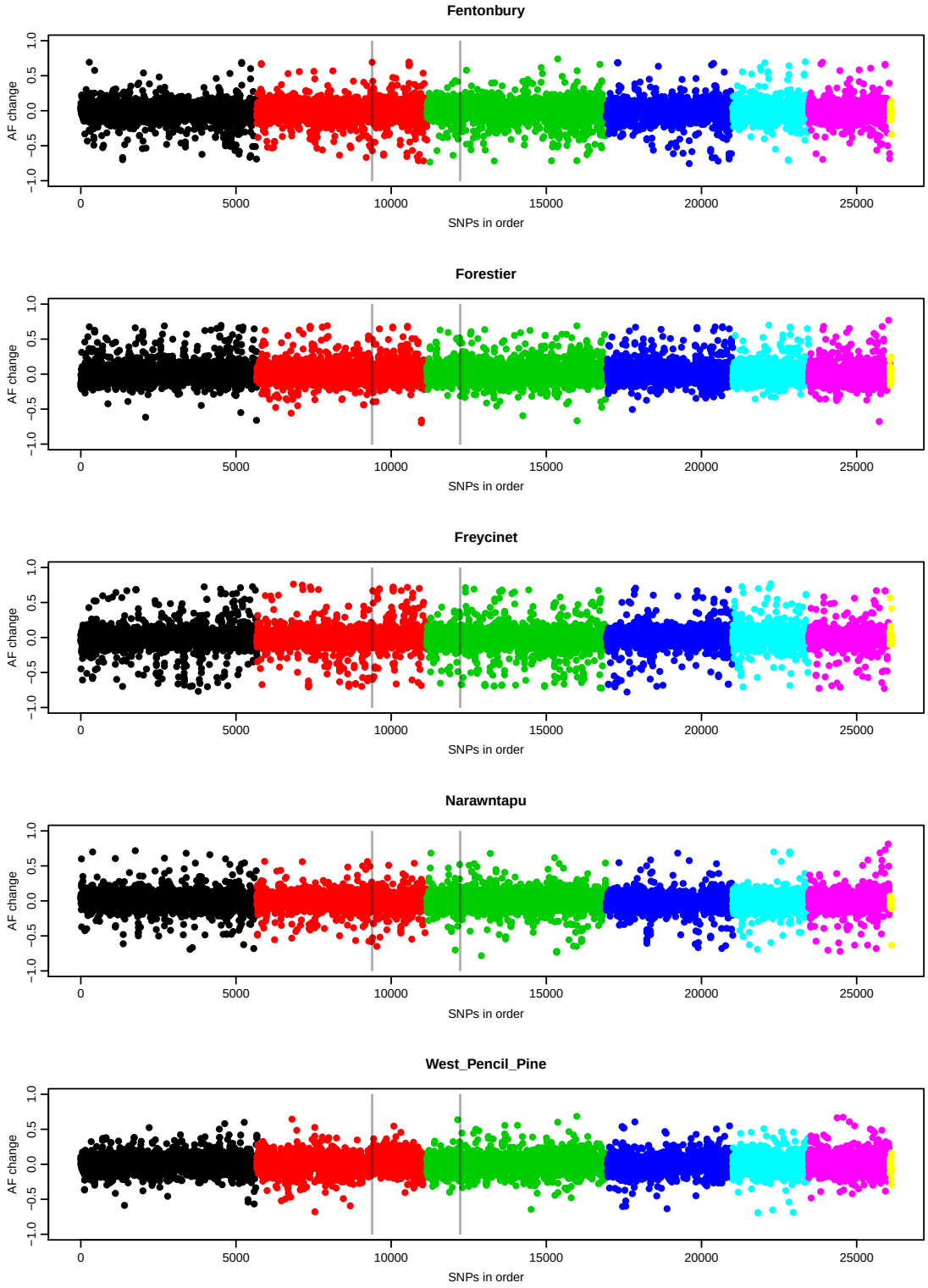
This resulted in this many SNPs:

Fentonbury	16037
Forestier	15784
Freycinet	20167
Narawntapu	18290
W Pencil Pine	16505
combined	26120

Of course, I could only apply this test to populations for which we had before and after samples, so no Woolnorth or Mt William

Results

Manhattan plots for each population. The coordinates should match up across plots



The quantiles of top SNPs near the two previously identified candidate regions were approximately:

population	chr. 2	chr.3
Fentonbury	0.99	0.95
Forestier	0.75	0.99
Freycinet	0.99	0.98
Narawntapu	0.95	0.99
W Pencil Pine	0.93	0.97

6.3 Change in variance in allele frequency

I took the ANGSD allele frequency estimates and, for every position calculated the variance in allele frequency (among populations) for the "before" and "after" datasets, including only the populations with both before and after data. Then I took the ratio of before:after.

The idea here is to try to detect conditionally neutral sites - those sites that had little effect on fitness before DFTD, but did matter after DFTD. In this case, I'd expect that different populations could differ quite a bit in allele frequency pre-DFTD due to drift, but would converge to a similar allele frequency post-DFTD due to selection. I am not aware of anyone who has tried this test before, although conditional neutrality has been discussed as a problem for landscape genomics.

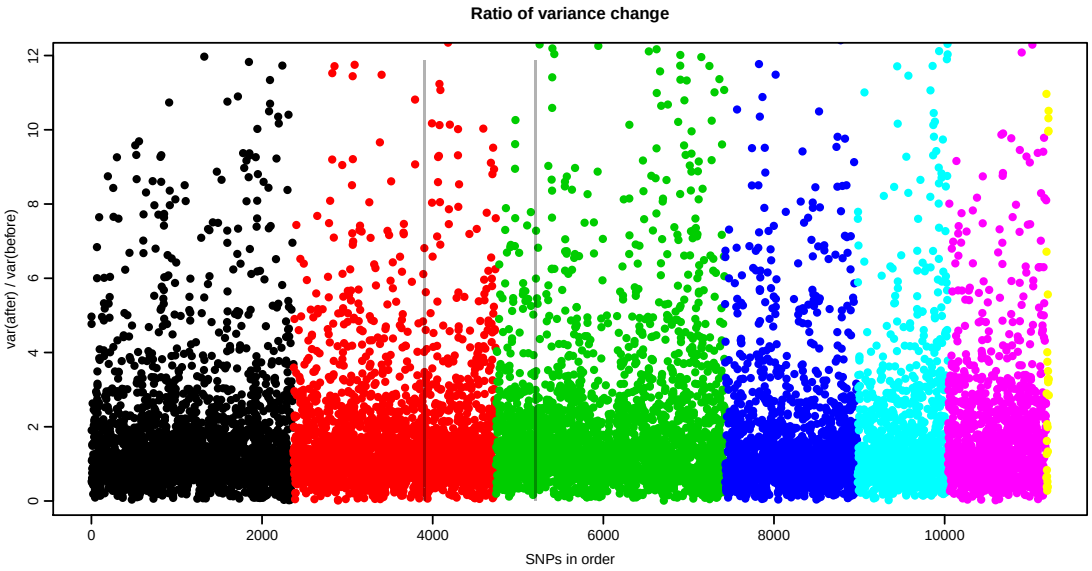
Settings

- Same as for allele frequency change, but applied to all populations at once

We ended up with 11,221 SNPs.

Results

The Manhattan plot:



The quantiles and ratios of best SNP near the previously identified regions:

region	quantile	ratio
chr. 2	0.95	6.8
chr. 3	0.95	7.3

It occurs to me now that it might be better to plot the ratios as (after / before) when after $<$ before, but as -(before / after) when after $>$ before. This shouldn't matter for ranking SNPs, however.

6.4 spatpg (2016-02-14 run)

"spatpg" is the software used to run the model from Gompert (2016). In brief, you give the program allele counts for multiple populations sampled at multiple times points and a file with environment values (in this case presence or absence of DFTD). The program then tries to estimate N_e and the strength of selection.

Details

The basic model describing selection looks like this:

$s_{i,j,k} = \alpha_i + \beta_i x_{j,k}$

where $s_{i,j,k}$ is the selection coefficient at locus i in generation k in population j, $x_{j,k}$ is the value of the environmental variable, and α and β are the intercept and coefficient of a regression describing the relationship between the environment and selection on locus i. We are most interested in β if we want to find loci that are under particularly strong selection.

It is assumed that there is no dominance, and some MCMC Bayesian magic is used to estimate s, α, and β. The effective population size is estimated based on all SNPs using Jorde and Ryman's estimator (uses data from multiple time points).

Making the input files

Unfortunately, spatpg does not directly take genotype likelihoods or probabilities (yet), so I had to produce integer counts of alleles from the ANGSD allele frequency estimates. To do this I multiplied the estimated allele frequency by twice the number of individuals with data, and then rounded to the nearest whole number.

Settings

Settings for choosing sites:

- p-value for test of segregating site $\leq 10^{-6}$
- MAF ≥ 0.05
- Minor allele *count* ≥ 3
- The above critera have to be met in at least 10 combinations of population and generation for a site to be included

These criteria resulted in 15,909 SNPs being tested.

Settings for running spatpg

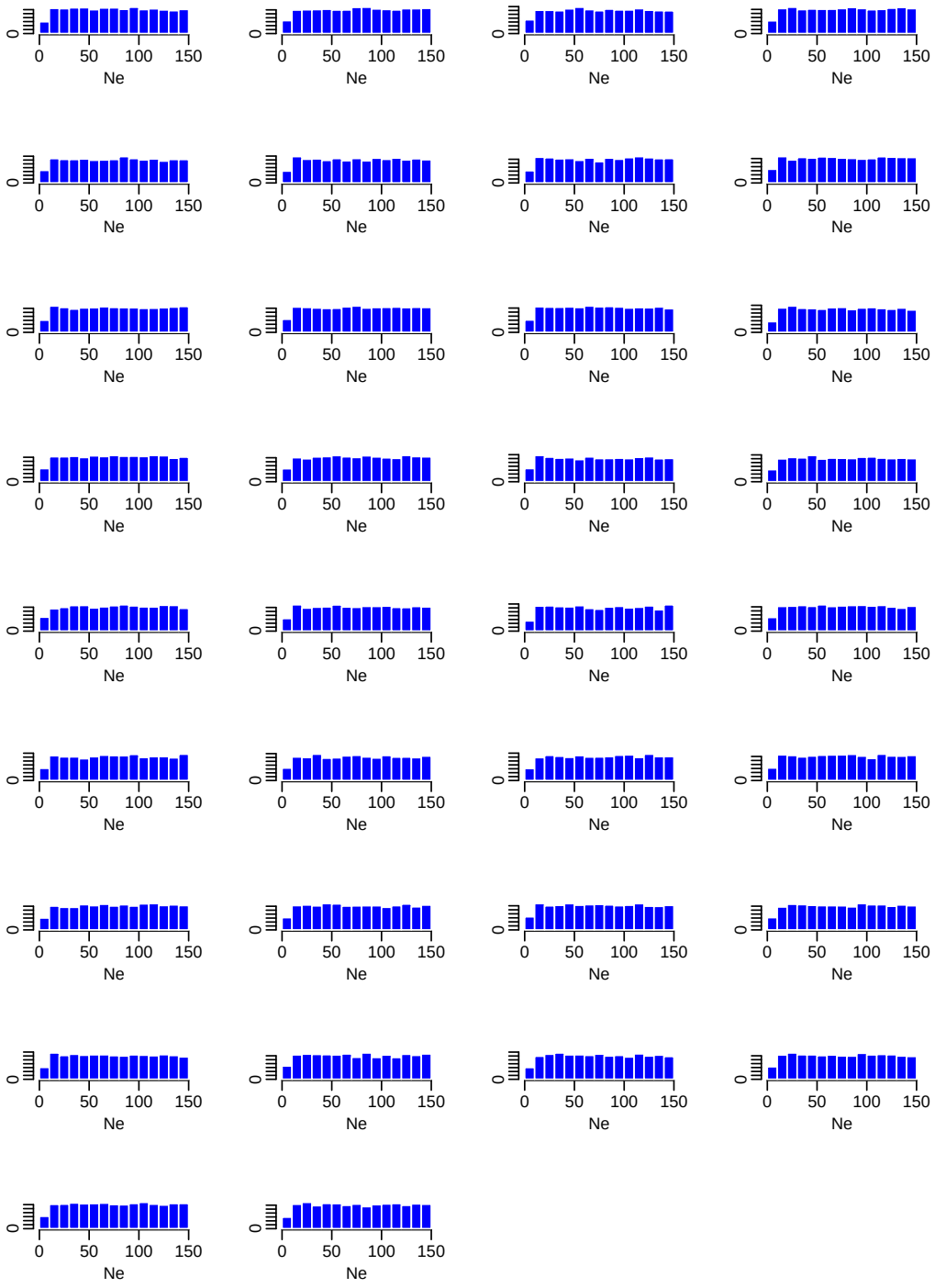
- generation time: 2 years
- 100,000 MCMC steps, sampled every 10
- 10,000 steps discarded as burnin
- $5 \leq \mathrm{N_e} \leq 150$: I thought this was reasonable

- Standard deviation on prior coefficient: 0.1. This is the default, and I left it there out of ignorance. I think that making this larger or smaller increases or decreases the posterior probability that a locus is under selection, respectively
- Standard deviation of proposal distribution: 0.05. Also the default. I think this parameter is for tuning the behavior of the MCMC sampling process

I discovered that with more than about 6,000 SNPs, spatpg would crash without an informative message. I split the SNPs into 15 chunks at random, and ran spatpg on each chunk separately. The disadvantage to this is that spatpg uses all the SNPs to estimate N_e.

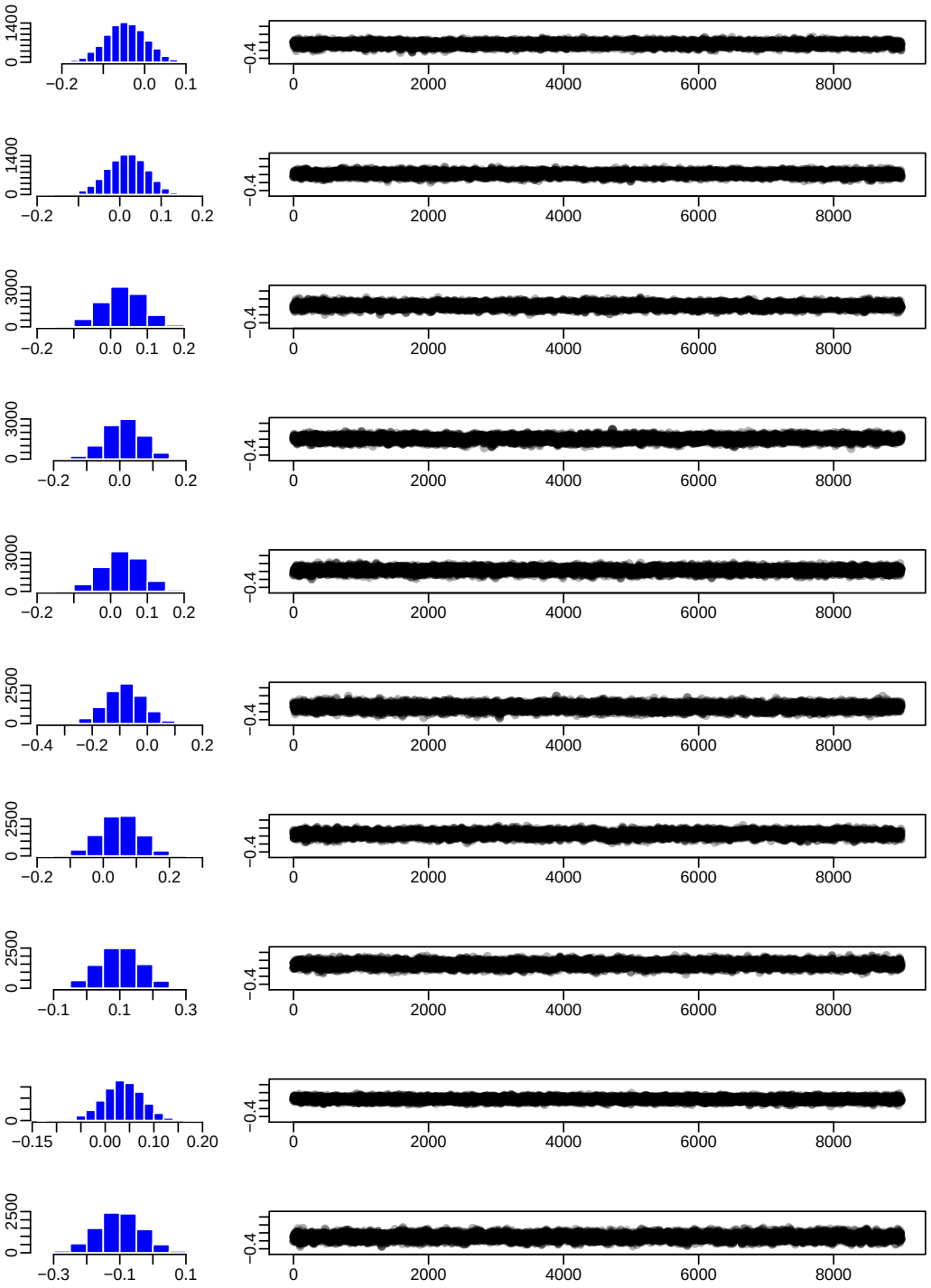
MCMC diagnostics

- N_e Estimates of N_e were all over the place - basically just random samples from the prior. With 15 chunks, 7 populations, and 8 generations, there are too many estimates of N_e to show here, but I selected a set from one chunk. Each histogram is the distribution of N_e estimates from one population/generation combination:



Because of this, I wouldn't trust the estimates of s, which are sensitive to N_e.

- Beta This is the statistic that I was most interested to look at. There are 15,938 beta estimates, so again, I can't show all of them. These are the distributions and the plots of the values over the chains for 10 randomly selected SNPs:



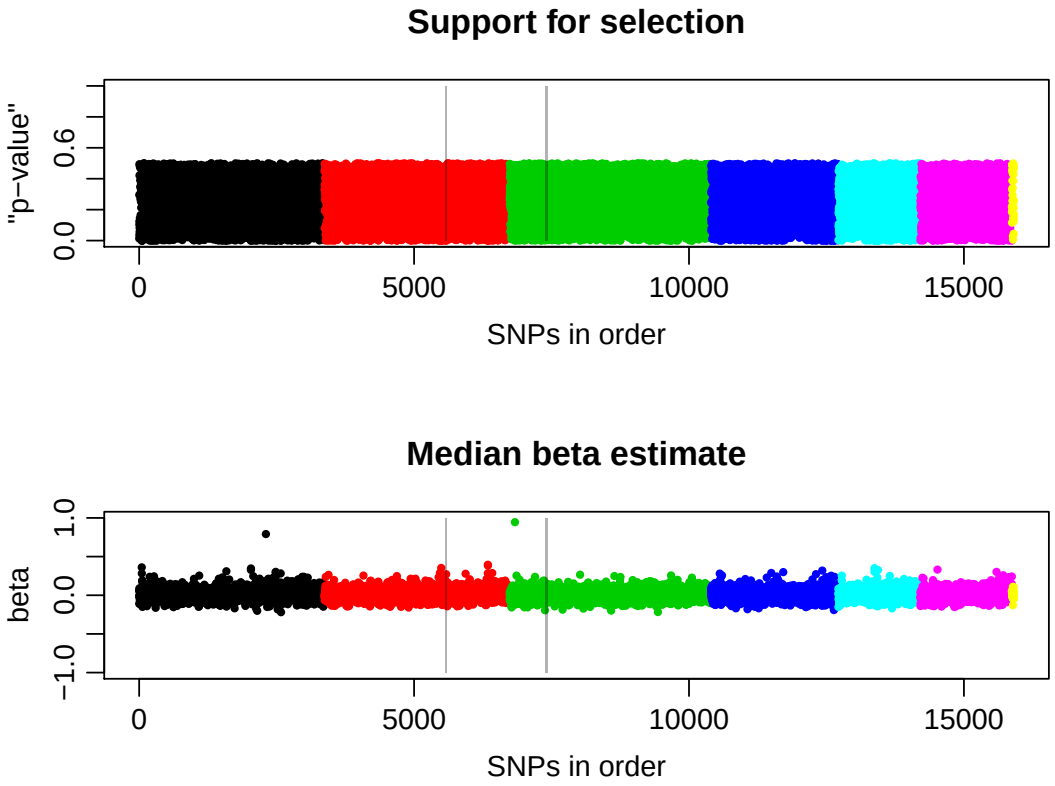
These look pretty good: there seems to be convergence, and the distributions are unimodal.

Results

The spatpg manual and paper suggest using the proportion of posterior that excludes 0 as the strength of support for selection. Thus, for each SNP, I found the proportion of MCMC samples $>$ 0 and the proportion $<$ 0, took the minimum of those two numbers, and used that as the "p-value."

I took the minimum because I don't know in advance which allele is going to be favored.

A Manhattan plot of the "p-values" and the median beta estimates:



It's interesting that there are a handful of SNPs with very large median estimates of β. I suspect these are probably wrong, but definitely worth investigating.

The most strongly supported SNPs near the candidate regions:

region	quantile	beta
chr. 2	0.99	0.27
chr. 3	0.98	0.15

The two really big outlier SNPs:

- chr3 GL849557, 148423: nearest gene is a phosphate regulating endopeptidase homolog (about 300 kb)
- chr1 GL834779, 747652: near a number of interleukin receptors and other immune genes - might be something interesting

6.5 What about F_{st} between before and after? (And LD statistics?)

I had planned to do this using the F_{st} method built into ANGSD, but it kept crashing, so I decided to abandon it for now. I think it might be better to use one of the dedicated F_{st} outlier detection methods, like Outflank instead. This means making individual genotype calls, but I think I want to do that anyway. (Update: in progress)

7 Composite statistic

This part is the shakiest. Ideally I want a method that takes into account the correlation among methods, and can deal with some missing data, and with tests that don't really produce p-values. I would like some input about improving this part.

Update: I am going to use the method recommended by François et al. 2016 to do this.

7.1 Composite statistic choices

What we did for the full RAD manuscript

For each SNP, we expressed its position in the genome-wide distribution as a quantile. So a SNP with a test statistic value (i.e. a change in allele frequency or rsb statistic) greater than 99% of all other SNPs would be 0.99. I arranged it so that larger values were always stronger evidence for selection.

We divided the genome into non-overlapping windows. Then, for each window, we found took the greatest value for each statistic, raised it to the power of the number of SNPs in the window and subtracted from 1 to get a sort of "p-value" for the window.

All the statistics were combined using Fisher's method for combining p-values:

$fischerstat = -2 \times \sum_{i=0} \log(p_i)$

where p_i is the "p-value" for the ith statistic. We compared $fischerstat$ to a chi-squared distribution with df = twice the number of statistics.

Method used by Randhawa et al.

Randhawa et al. (2014, BMC Genetics 15:34) devised a composite statistic based on ranking SNPs, rather than using the p-values directly.

First, for each statistic, the loci are ranked and the ranks are divided by the number of loci - so the best SNP out of 100 would be 0.01. This is the same as subtracting the quantile from 1. Then a Z-score is calculated from the transformed ranks, using the inverse cumulative distribution function for the standard normal distribution. I think this is the *qnorm* function in R, and is the opposite of going from a Z-score to a p-value.

Then, for each locus, they averaged the Z-scores from each statistic and calculated a p-value from the average Z-scores.

This composite statistic has some similarities to what we did before in that it is based on ranking SNPs.

Method used by Ma et al.

Ma et al. (2015, Heredity 115:246-436) came up with another composite statistic called "De-correlated Composite of Multiple Signals" that adjusts for the fact that different tests are often somewhat correlated. I ended up using a slightly modified version of this method.

For each locus:

$DCMS = \sum_{t=1}^{n} \frac{\log(\frac{1-p_t}{p_t})}{\sum_{i=1}^{n} |r_{it}|}$

where i and t are indexes over statistics and r_{it} is the correlations between statistic i and t. DCMS is weighted sum of transformed p-values, where the weights are determined by the correlation

between statistics. I think the transformation of the p-value is meant to mimic a Bayes factor from a Bayesian composite test (e.g. Grossman et al. 2013, Cell 152:703-713).

Method used here (run: compositet_stat_2016-02-18)

I modified the Ma et al. method by using "p-values" for windows instead of p-values for SNPs. I calculated window "p-values" the same way as before, and then fed them into the formula for correcting for correlations.

After doing this, it occurred to me that we probably don't have as much of a problem with missing data this time as before, so I could run the tests on individual SNPs too. In fact, most SNPs have results from multiple tests, so I think we should work with SNPs instead of window values - this avoids having to correct for the number of SNPs in a window.

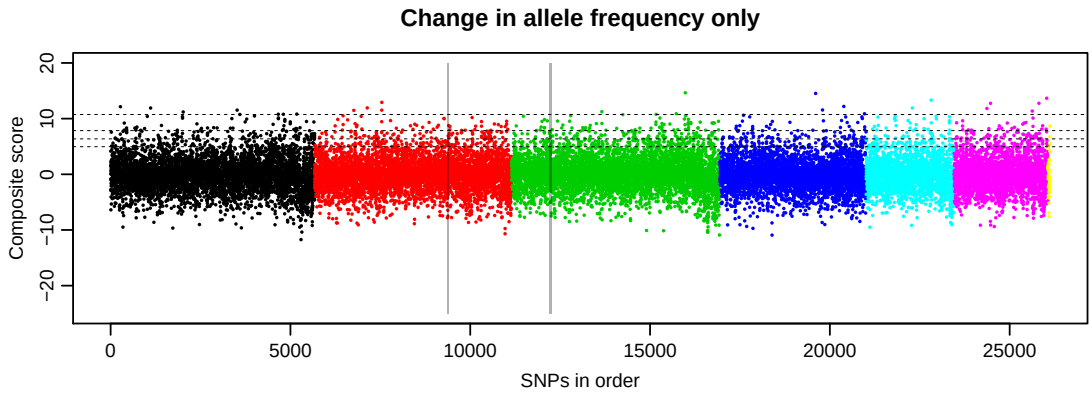
# SNPs	# tests
6899	1
2138	2
1600	3
1489	4
1844	5
1030	6
2007	7
9142	8

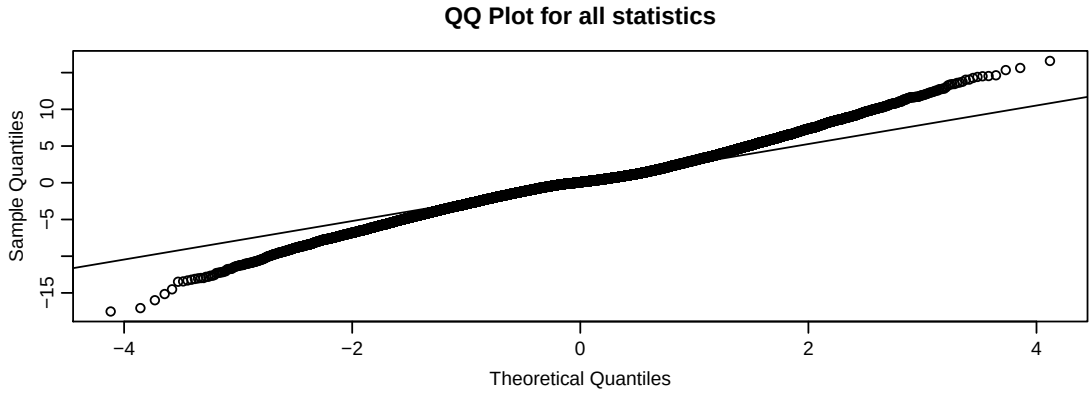
Total SNPs: 26149

As I mentioned above, I think there is room for exploring more ways to combine these tests (and there might be other tests that are useful too). I might be giving too much weight to allele frequency change tests here - each population counts just as much as the spatpg results or the association results. Also, maybe we should figure out how to take into account the direction of allele frequency change, not just the magnitude. Or maybe we use a composite statistic to combine similar tests, but then see whether there are genes near top SNPs for all methods.

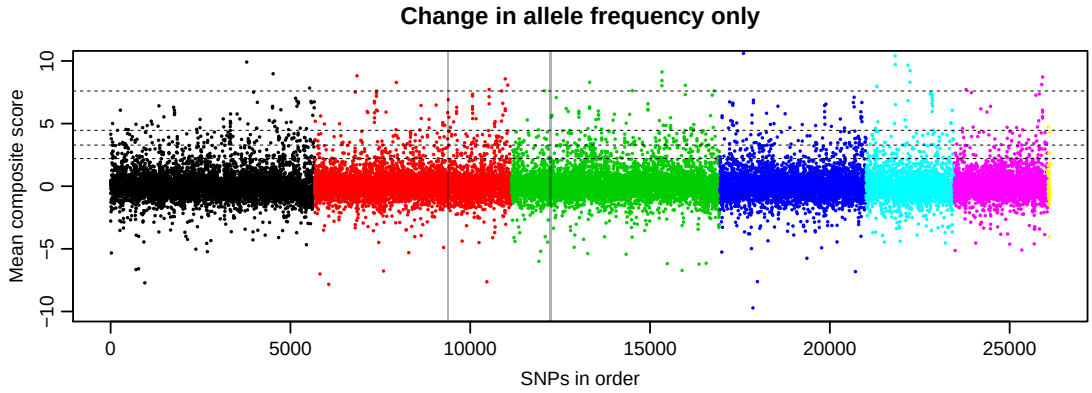
7.2 Results

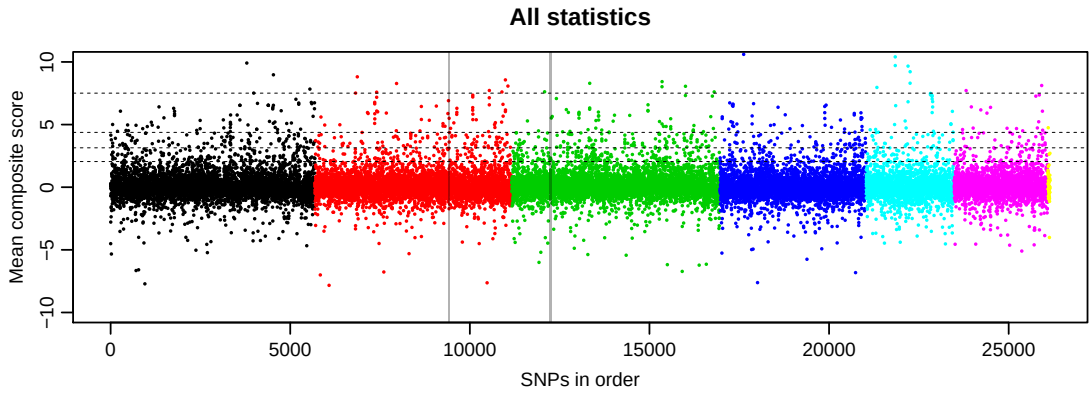
Manhattan plot for SNPs:

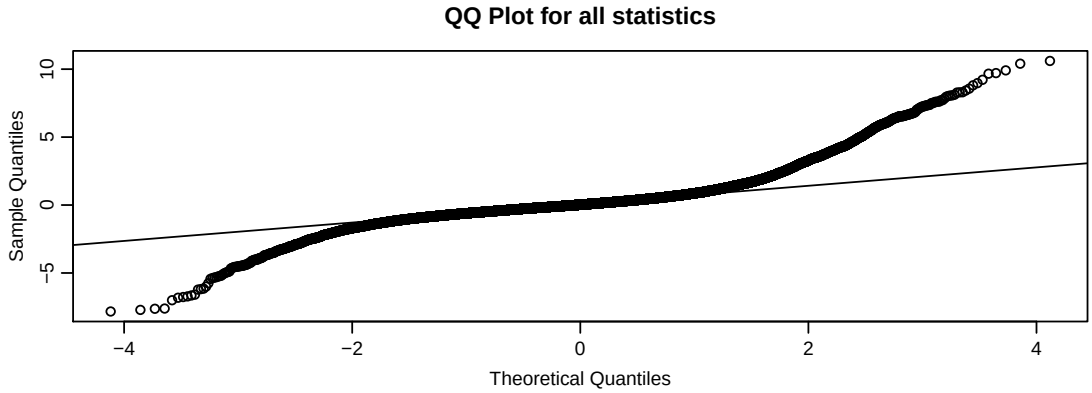




I drew in horizontal lines marking off the top 0.1%, 1%, 2.5% and 5%. The larger the score, the stronger the evidence. It seems odd that we have "peaks" of especially weak evidence, but not as much for strong evidence. One issue might be that even though we have pretty good overlap among different methods, there is still variation in the number of methods tested for each SNP. Because of this, taking a sum might not be appropriate. To compensate for that, I divided the DCMS score by the number of tests with non-missing values:





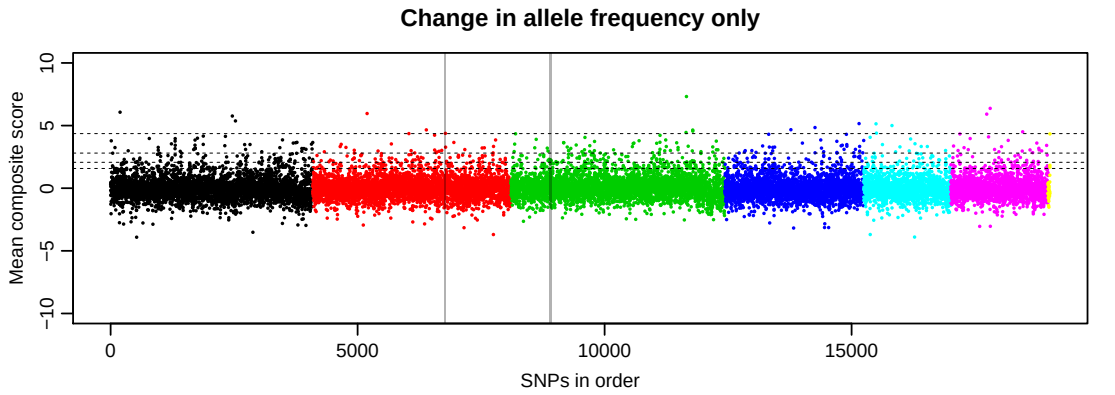


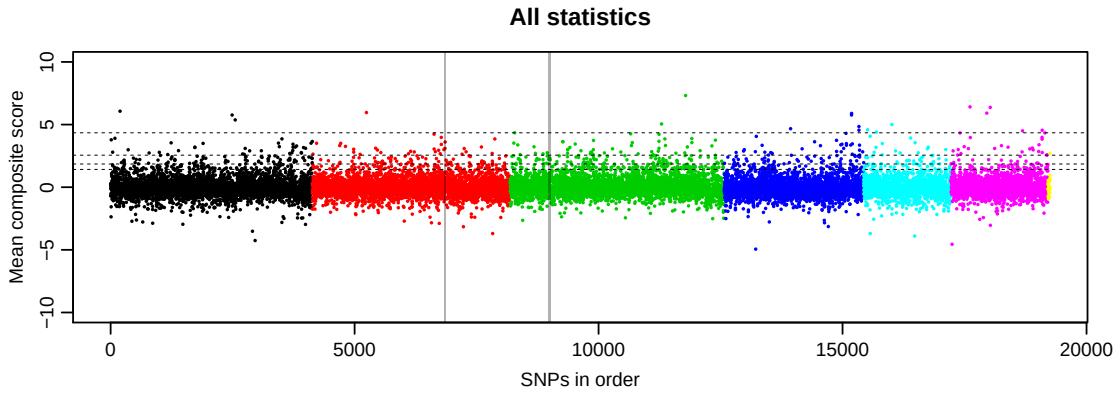
This is based more on intuition than statistical savvy, but I think the plots look much better now. The QQ plot looks just as weird, but I don't really think it needs to look normal.

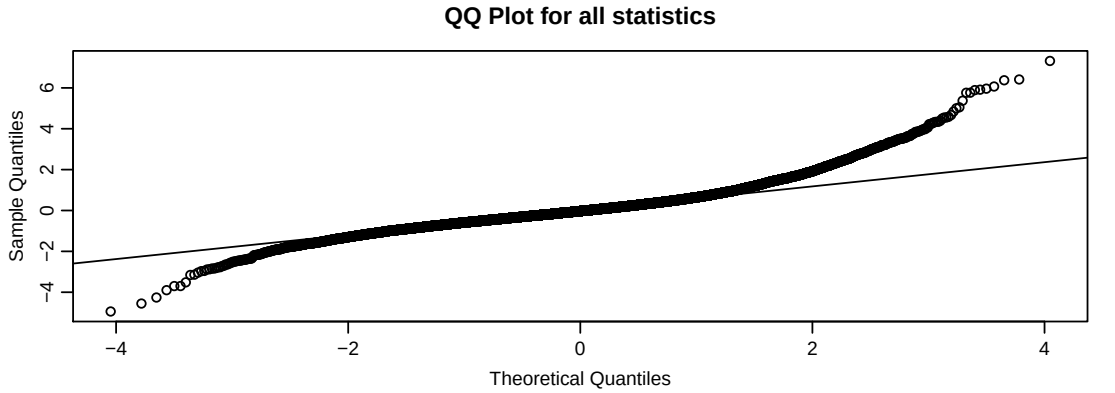
Previous candidate regions

The top SNPs near the chr. 3 candidate region end up in the top 5%, and the top SNPs near the chr. 2 candidate region end up in the top 1%. If we only consider SNPs with more results from more than one statistic, then the chr. 3 region jumps to top 1% as well. This isn't a good argument for using just the SNPs with more than one statistic though.

Just to see what it looks like, here is the Manhattan plot for the composite statistic, including only SNPs with more than one statistic:







7.3 Gene lists

Genes within 100 kb of the top 1% of SNPs for the mean composite statistic (all stats combined) - 276 genes, which does includes cereblon, but does not include the chr 3 genes (though it does include other genes on the same scaffold). Genes within 100 kb of the top 5% of the mean composite statistic - 1425 genes, including all the candidates.

Genes within 100 kb of the top 1% for allele frequency change combined across populations or top 1% for spatpg or top 1% for the association test or top 1% for variance test - four genes flagged by three tests (one has a cancer phenotype), 59 genes flagged by two tests (includes cereblon).

Genes within 100 kb of the top 5% for allele frequency change combined across populations or top 5% for spatpg or top 5% for the association test or top 5% for variance test - 29 genes flagged

by all four tests (includes cereblon), and 148 genes flagged by three tests. Most genes in the chr 3 region are flagged by the combined allele frequency test, the variance test, and spatpg, but not the association test.

So it looks like the results are not contradictory to the previous results, but also add a new set of candidates that we have to decide what to do with.

Genes flagged by all four tests:

8 What's next

There are two types of tests that I think are missing:

- F_{st} outlier tests
- LD-based tests

Both of these require individual genotype calls, so I think that's what I'll do next. I already have genotype calls, but I have changed (hopefully improved) my alignment and filtering workflow since then; it would be good to have consistency. Also, there have been some updates to sample names that changed which libraries needed to be merged.

I also need to compare the Freycinet 2014 samples to the Freycinet samples from other years. And I need to run a GO term enrichment analysis.

Individual genotype calls and phasing

- I am feeding the genotype likelihoods into BEAGLE for phasing
- Partially done - some issues with qsub
- Also need to decide how much imputation to allow

Compare Freycinet 2014 to other Freycinet samples

- Eight dropout samples
- Association test is running

However, I think it would be nice to do something more interesting than just present a list of candidate genes chosen with lots of methods. Any thoughts?

Another idea: Take the top SNPs from composite statistic and use as the predictors of susceptibility, and thin this list by taking only SNPs with allele frequency change in the same direction in every population. Of course, we need to know which allele is the susceptible allele for the SNP panel to work.

9 Locations of scripts and output

Under the Lustre file system (I copy into gluster on a regular basis, but the newest stuff will be in Lustre):

`/mnt/home/bepstein/devilsrapture`

is the top directory for everything. Scripts can be found in `scripts/batch/` and output is in `results/`.