Master Project

Lab book

Cyril Matthey-Doret

August 29, 2017

Chapter 1

Introduction

The goal of this project is to identify the locus or loci responsible for sex determination in the parasitoid wasp *Lysiphlebus fabarum*. The species is known to have CSD, which means there are one or more loci that need to be heterozygous in order to triggger female development. I will use the offspring of asexual (thelytokous) females to identify these loci. The offspring consists of haploid males, diploid males and diploid females. Using DNA sequencing data, I will exclude haploid males as they do not provide information on homozygosity/heterozygosity and compare only diploid males versus females to identify the CSD locus/loci. *Lysiphlebus fabarum* wasp specimens (generation F4) issued from thelytokous mothers (generation F3) are used. These individuals come from crossing experiments in a strongly inbred line. Here, we use restriction site associated DNA sequencing(RAD-seq) with a custom pipeline to locate the locus/loci. Samples were single-end sequenced using a ddRAD protocole and digested with ecoRI and mseI. There are 2 separate libraries with two different illumina adaptors (detailed in next section). Statistics for the raw RAD-seq data of both libraries of F4 individuals are shown in table 1.1. I also reused sequencing data for the mothers (F3) from Casper, however I don't have the raw reads statistics as they were already processed.

Data summary:

library	lib7	lib7b
raw reads	163,506,603	133,574,055
containing adaptors	23.25%	45.84%
fragment size	302bp	302bp
mean quality score	34.88	35.05
>= Q30 bases	92.62%	92.13%

Table 1.1: Summary statistics of raw reads from the 2 libraries of F4 individuals RAD-seq data.

This lab book will document my progression for mapping the CSD locus, which can be broken down into 4 main steps: preprocessing of sequencing data, STACKs analysis for RAD-seq data, association mapping and orthology search. These steps are likely to change and throughout the project and it is likely that more will be added. The general process is described visually in figure 1.1.

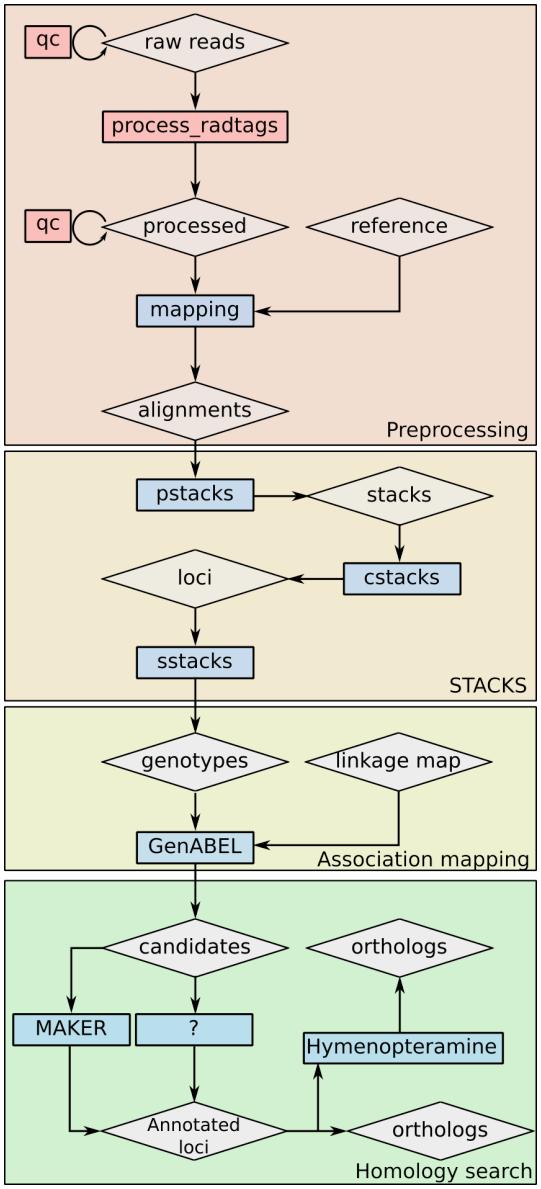

Figure 1.1: **Pipeline described in the current lab book**. Diamonds represent data, rectangles represent operations/programs. Operations/programs in blue are included in the makefile, thos in red are not.

Chapter 2

Processing reads

RAD-seq data was split into 2 separate libraries: 7 and 7b. Together, the libraries contain 173 F4 individuals from 11 different F3 mothers. There were 96 samples in library 7 (one of which was contaminated) and 77 in library 7b. In total we have 172 valid samples across 11 families. Additionally, I inherited from 28 individuals from another library (lib6) which are offspring from families A and B.

2.1 Quality control

fastqc was used for quality control, separately on each file, and on all files together in the library.

2.2 Demultiplexing

The process-radtags program from stacks was used for demultiplexing and removal of Illumina adaptors. The operation was performed separately for libraries 7 and 7b:

```
$ process_radtags -p raw/ -o processed/ -b \
/barcodes -e ecoRI --filter_illumina -E phred33 \
 -r -c -q --adapter_1 adapter --adapter_mm 2
```

lib7	Truseq adapter, index 6 [1]
lib7b	TruSeq adapter, index 12 [2]

2.3 Trimming adaptors

This step is performed by process radtags at the same time as demultiplexing. I tried different values for adapter mismatches, between 0 and 3 (i.e. reads containing sequences distant from the adapter by n mismatches are removed). This did not cause any major difference and therefore I will perform all downnstream analyses with an allowance of 2 mismatches in the adapter.

Note: Demultiplexing did not yield any read for CF4F10.

[1] GATCGGAAGAGCACACGTCTGAACTCCAGTCACGCCAATATCTCGTATGCCGTCTTCTGCTTG
[2] GATCGGAAGAGCACACGTCTGAACTCCAGTCACCTTGTAATCTCGTATGCCGTCTTCTGCTTG

Chapter 3

STACKS pipeline

The pipeline described in this chapter will map the processed reads to the reference genome and build a catalogue of loci. It will eventually implement additional features such as calculating population statistics. At each step of the pipeline, I will try different combinations of parameters and choose the one yielding the best results.

3.1 Mapping

Since the reference genome of *Lysiphlebus fabarum* was recently released, I will first map the sequencing reads to the reference, using BWA, but I may also want to use bowtie to compare the results, eventually. At the moment, a draft reference genome is available, table 3.1 displays some summary statistics of it.

3.1.1 BWA

BWA provides 3 different algorithms: MEM, backtrack (aln) and SW. MEM is normally the better for Illumina reads longer than 70bp, therefore I will be using this one here. Backtrack is preferred for short reads and SW with frequent gaps. There are also 2 algorithms for building the index: 'is' and 'bwtsw'. 'is' is used with reference ¡2GB, 'bwtsw' with larger references.

General commands:

```
$ bwa index -p <out_index_name> -a <algorithm>
$ bwa mem <index> <sample.fq> > <out.sai>
> $sample-$prefix.sam
```

When using bwa-aln, there the command running the alignment (after indexing) is:

```
$ bwa aln -n <mismatches> $index <sample.fq> > <sample.sai>
$ bwa samse -n <max_dupl> $index $sample.sai $data_dir/$sample.fq.gz
```

The first command (bwa index) constructs an index from the reference genome, whereas the second one (bwa mem/aln) actually runs the aligner. The third command (bwa samse) allows to transform the .sai into .sam files.

Statistics	Values
Assembly length (Mbp)	140.7
Largest contig (Mbp)	2.2
Mean contig size (kbp)	82.9
Median contig size (kbp)	30.4
N50 (kbp)	216.1
Number of contigs	1698

Table 3.1: Assembly statistics for the current reference genome of *Lysiphlebus fabarum*

Here is the list of different mapping parameters I may try with bwa-mem:

- -k : minimum seed length (will miss matches shorter than value)[19]
- -w : band width (gaps longer than value will not be found)[100]
- -d : maximum distance between query and reference positions before stopping seed extension. [100]
- -r : triggers reseeding for a MEM longer than min_seed_len*float. Larger values yield fewer seeds − > faster alignment but lower accuracy [1.5].

In the case we use the backtrack (aln) algorithm instead of MEM, the only parameter worth tuning is -n, the number of mismatches allowed.

Note: I did not include parameters that are not relevant to sensitivity (e.g. threads) or parameters that involve scoring (changing these naively would probably have a negative impact). Full list of parameters is on the official bwa website

Note2: About multiple hits and BWA-mem:
(https://github.com/lh3/bwa)
2. Why does a read appear multiple times in the output SAM?

BWA-SW and BWA-MEM perform local alignments. If there is a translocation, a gene fusion or a long deletion, a read bridging the break point may have two hits, occupying two lines in the SAM output. With the default setting of BWA-MEM, one and only one line is primary and is soft clipped; other lines are tagged with 0x800 SAM flag (supplementary alignment) and are hard clipped.

3.1.2 Bowtie2

General commands:

```
$ bowtie2-build reference_genome.fa L_fabarum
$ bowtie2 -x <ref_index> -U <unpaired_reads_files> -S <out_SAM_file>
```

The first command (bowtie2-build) constructs a set of index (extension: .bt2) from the reference genome, whereas the second one actually runs the aligner.

Here is the list of different mapping parameters I may try:

- –trim5 ¡n¿ : trim n bases from the 5' end (left) of each read before alignment
- –trim3 ¡n¿ : trim n bases from the 3' end (right) of each read before alignment
- -D : maximum number of seed extension that can fail in a row before stopping (increasing makes bt2 slower)
- -R : maximum number of re-seeding when attempting to align read with repetitive seeds (increasing makes bt2 slower)
- -N : number of mismatches permitted per seed (increasing reduces false negative, but makes bt2 slower)
- -L : length of seeds (decreasing makes bt2 slower but more sensitive)
- -i : interval between seeds (increasing makes bt2 slower but more sensitive)

Recommandations from the bowtie2 website to make alignment more sensitive: a) make seeds closer (reduce i) b) make seeds shorted (reduce L) c) allow more mismatches per seed

End-to-end versus local alignment (–local): end to end takes all bases in the reads into account, while local allows to trim reads to exclude the ends from the alignment. Seeds are substring of the reads which bowtie2 tries to align to narrow down the valid regions for aligning a read. There is a list of preset values for these parameters on the official bowtie2 website. Preset parameters differ between local and end-to-end modes. There are other parameters, such as score weights for gaps mismatches and allowance for 'N' ambiguous characters but changing these naively could probably have negative effect on the alignments. Procedure: try out different preset values and select the one yielding the best results. Then, eventually tweak the parameters slightly from the preset values. These simulations can be run on a subset of individuals to speed up the process.

note to self: at the end of a run, bowtie2 prints a summary to stderr such as:

20000 reads; of these:
20000 (100.00%) were unpaired; of these:
1247 (6.24%) aligned 0 times
18739 (93.69%) aligned exactly 1 time
14 (0.07%) aligned >1 times
93.77% overall alignment rate

If I use Bowtie2, I will redirect the stderr and parse it into a csv file and generate plot to visually estimate the best parameter values.

3.1.3 Mapping results

I used the aln algorith of bwa, since the mem algorithm did not report multiple alignments and has very few documentation available. I tried different mismatches values with aln (Figure 3.1), ranging from 0 to 8, and I chose to use 4 since the increase in single was quite low above this value. When the mismatch parameter is set to 4, running bwa-aln on a subset of 12 samples yielded 57% of single hit reads, 26% of multiple hits and 17% of unmapped reads.

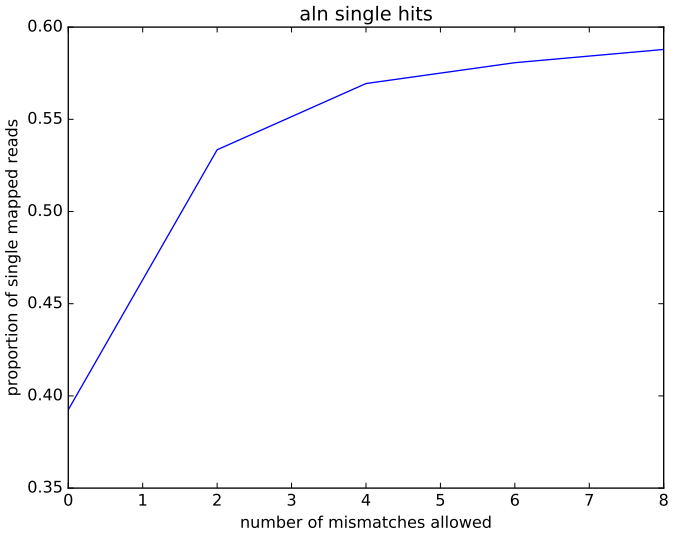

Figure 3.1: Results of the BWA mapping with different parameter values for the number of mismatches allowed during mapping. The plot shows the proportion of reads that are mapped uniquely to the reference genome.

3.2 pstacks

pstacks is a component of the STACKS suite that takes stacks of reads aligned to a reference genome as an input (typically in the SAM format) and idenify SNPs

general command:

```
$ pstacks -f <input_path> -i <sample_ID_int> -o <out_dir> \
 -m <min_depth> -p <num_threads> -t <file_type>
```

Parameters I may want to change are:

- -m : minimum depth of coverage required to call a stack [3]
- –max_clippped : alignments with more than X soft-clipped bases are discarded [15%]
- –min_mapq : minimum required quality [10]

min˙cov	nloci	mean˙cov	sd˙cov
1	5020.1	31.8	58.6
2	3659.1	42.5	65.0
3	2747.2	49.9	68.2
4	1505.3	55.6	70.1
5	781.9	60.1	71.3
6	597.9	63.9	72.3

Table 3.2: Summary statistics of stacks obtained with different parameter values for minimum coverage in pstacks.

Note: there are 3 models; SNP, bounded SNP and fixed. SNP is the default model, bounded SNPs allows to give prior expectations about the error rate, which can allow better estimations of heterozygosity and the fixed model identifies all fixed sites and masks all others.

3.2.1 pstacks results

Pstacks was run on aligned reads (BWA, 4 mismatches allowed). I tried different values for the minimum coverage required to call a stack (-m parameter), ranging from 1 to 6 (Figure 3.2). Below is the value for minimum coverage, along with the mean number of loci and alleles that was produced per sample. This table was produced including all non-empty samples (198 individuals) and the variables are averaged (arithmetic mean) over all those samples.

I will use 4 reads minimum coverage as this is already high and using lower values does not improve the output in anyway.

Note: Locus are regions formed by one or more stacks. Alleles are different stacks at the same locus.

3.3 cstacks

cstacks is a component of the STACKS suite that builds a catalog of loci with different alleles from a set of processed samples.

general command:

```
$ cstacks -s <sample_prefix> -o <out_dir> -b <catalogue_ID>\
-p <num_threads> -n <num_mm> -M <pop_map>
```

Parameters I may want to change are:

- -n : Number of mismatches allowed between sample loci when building catalogue [1]
- -g : base catalog on alignment position instead of sequence identity

Note: There are also advanced options such as gapped assembly parameters and loci matching multiple catalogue entries, but these are probably not relevant here.

3.3.1 cstacks results

I changed the number of mismatches allowed between samples between 1 and 4. Table 3.3 shows the results with a subset of 11 samples ($[A-L\&\&[\wedge B]]01$) because B01 was empty (few, low quality reads).

I tried running cstacks on all (non-empty) samples, and also modifying the script so that it will first compute the mean number of tags (reads) across all samples, and exclude those with less than 10% of this value when building the catalogue. From the 202 original samples, 4 were empty and excluding low quality ones removed 13 additional ones.

mismatch	mean˙loci	mean˙alleles
1	2696	7799
2	3129	8464
3	3198	8704
4	3263	8914

Table 3.3: Summary statistics of loci obtained with different parameter values in cstacks.

Update: 21.07.2017
When including mothers, I reach a total of 212 individuals. I use the same filtering strategy to exclude samples with low amounts of reads (Figure 3.2). From the 212 original samples, I excluded 17 low quality samples, among which 4 were totally empty.

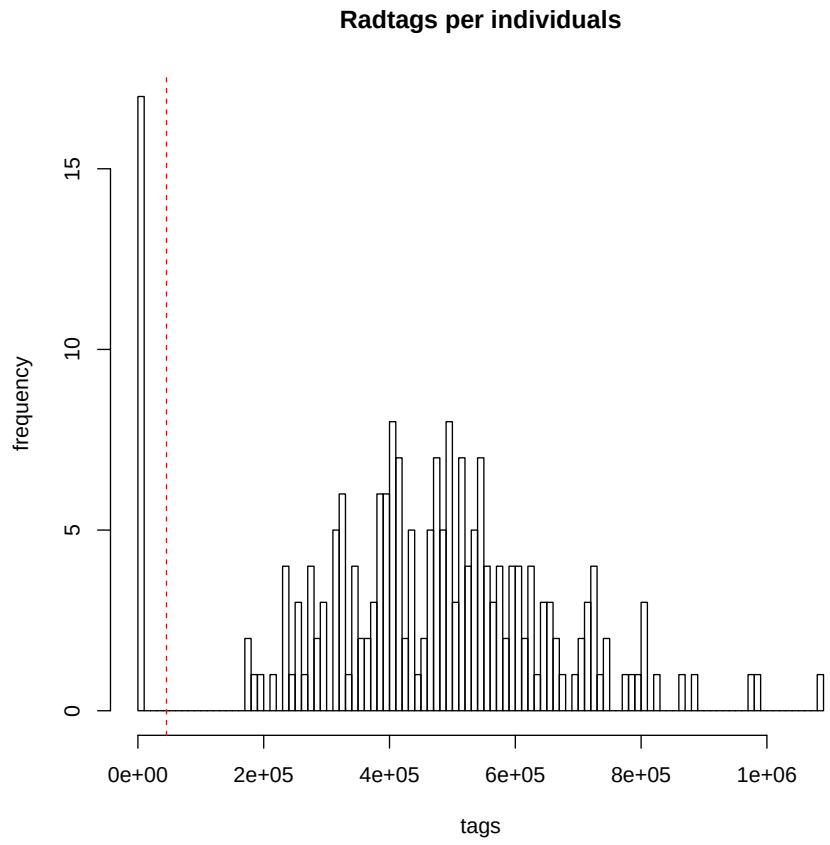

Figure 3.2: distribution of the number of radtags (reads) per sample. The red vertical dashed line indicates the cutoff value set to 10% of the mean. Samples below this value are removed from the analysis.

Summary results with n=1:

- All non-empty samples (198): 25618 alleles in catalogue, over 7062 loci.
- Excluding samples with <10% of mean tags (185): 25585 alleles in catalogue, over 7046 loci

note: following recommandations from Paris et al. 2017, Lost in parameter space: a roadmap for STACKS. *Methods in Ecology and Evolution*, I set the value of n to 3 (M-1=3)

3.4 sstacks

sstacks is used to generate one file per individual, in each file, the matching loci point to the cstacks catalogue. There is no crucial parameter to change in this program.

3.5 populations

The "populations" component of STACKS is used to compute population genetics statistics on a set of individuals. I use it here to compute FST statistics (fixation index) along the genome. It offers several features to compute different statistics, including a bootstrapping feature and a "kernel smoothing" flag, allowing to take neighbouring region into account with a decreasing weight as a function of their distance from the focus nucleotide. I will use both of these features to compute FST.

The main features to change in FST calculation are :

- -r: minimum percentage of individuals in a population required to process a locus for that population.
- -p: minimum number of populations a locus must be present in to be procesed.
- -m: minimum stack depth required for individuals at a locus.

Example populations call for FST calculation:

```
$ populations -P <stacks_files> -M <popmap> -b 1
 -k -r 0.75 -f p_value
```

3.5.1 populations results

I ran populations with the following parameters:

- -r: 0.75 - 0.85
- -p: 2
- -m: 5

Therefore, only loci with at least 5 reads of coverage were included, each loci also needs to be in at least 75% to 85% of all individuals and in both populations (males and females).

This table summarizes the first statistics I extracted from the VCF files (using vcftools) with the different values for r:

r parameter	obs.hom.	exp.hom.	n.sites	inbreed.coef.	mean depth	obs./exp. hom.
75	957.89	895.09	1171.82	0.23	121.09	1.07

plotting the inbreeding coefficient per individual with r=75 yields:

When increasing the minimum depth above 15, populations crashed. Disabling bootsrapping and kernel smoothing fixed the issue. Guess: May be caused by a contig smaller than the sliding window size during kernel smoothing. This could be the case if the sliding window has a min number of loci, in which case increasing minimum depth would cause the window to enlarge.

I tested for correlations between Mean depth of loci and homozygosity in all individuals and per family, no significant correlation was found. This means there should not be significant allele excludion caused by too stringent min depth (i.e. stacks removed because of low coverage, resulting in homozygous loci).

3.6 STACKS parameters summary

process radtags	Mis:2
bwa	Mis:4
Pstacks	MinDep:3
Cstacks	LocMis:1
Sstacks	-
populations	IndProp:0.75, MinDep:5(25)*, MaxHet:0.9

* Stringent MinDepth value used to exclude haploid males. The smaller value is used for downstream analyses to keep more loci.

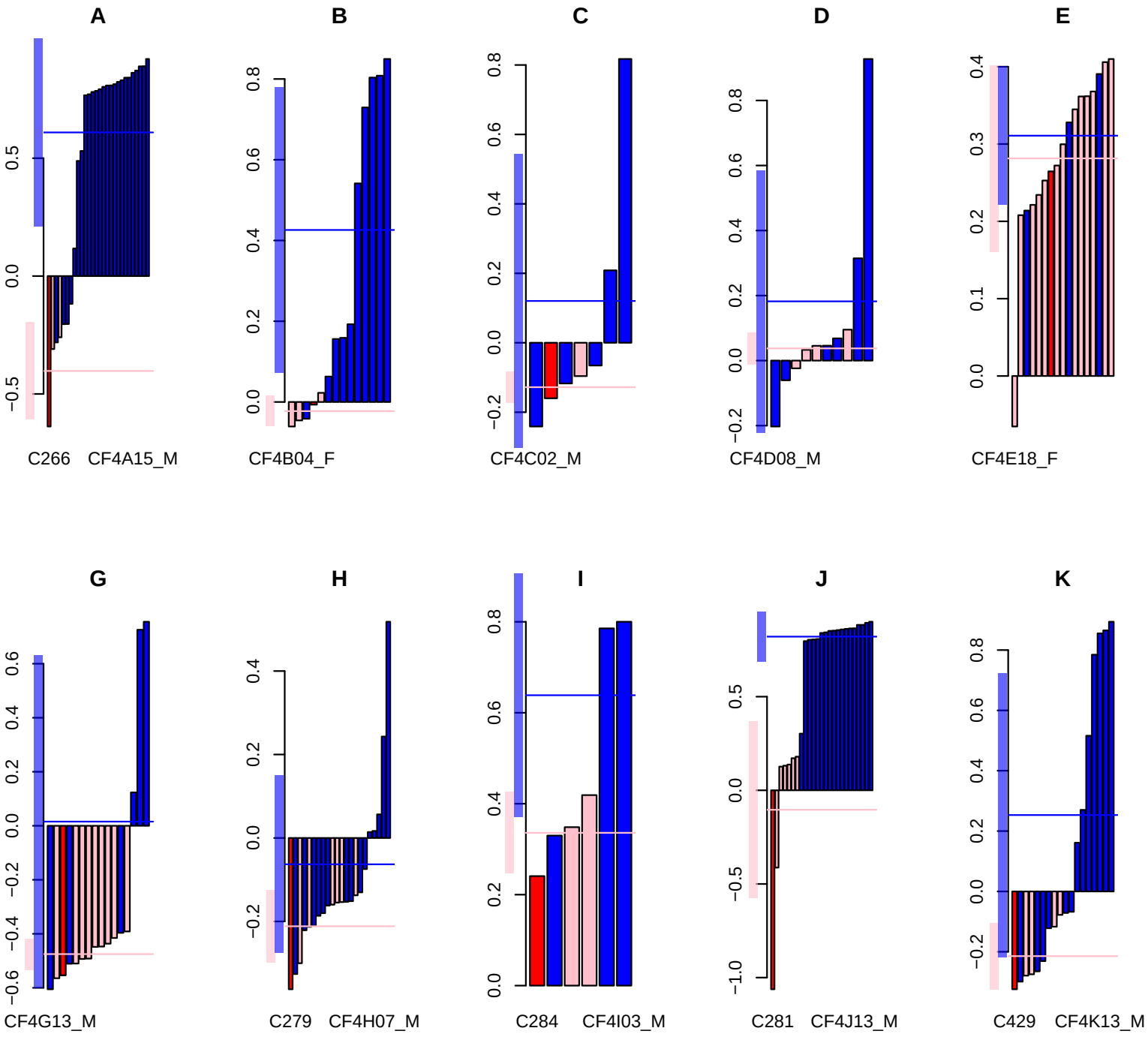

Figure 3.3: Inbreeding coefficient (F). Each plot is a family, each bar is an individual. Blue bars represent males and pink ones represent females and red ones are mothers. color bars on the y axis span the mean +- standard deviation of males and females, respectively. In theory, mothers should have the lowest inbreeding coefficient of their family (highest heterozygosity)

Chapter 4

Excluding haploids

I used the F statistics (coefficient of inbreeding) computed by vcftools to measure homozygosity levels in each individual. The F statistics used were generated with more stringent parameters (min depth: 25) to call ploidy more confidently. It is calculated as $F = \frac{O-E}{N-E}$ where O is the observed number of homozygous loci, E is the number expected by chance and N is the total number of loci.

First, I calculated the Mean μ and standard deviation σ of the inbreeding coefficient F among daughters in each family. I then classified males with:

$$F > \mu + 2\sigma$$

as haploid. This is stringent threshold should exclude all haploid males, but it can also potentially exclude diploid males from the analysis.

Update: 08.06.2017

I tried 12 different thresholds for excluding haploids. The thresholds were computed according to these rules:

- 1: All thresholds follow the formula $\mu + \tau(N\sigma)$ with $1 \leq N \leq 4$
- 2: $1 \leq \mu \leq 4$
- 3: τ is a function for transforming the inbreeding coefficient values. Either $\tau(F) = F^2$ or $\tau(F) = \sqrt{F}$

As shown above, in this report I will use the threshold named "m2" which corresponds to $F > \mu + 2\sigma$. Figure 4.1 shows a plot showing the separation of haploid and diploids using this threshold.

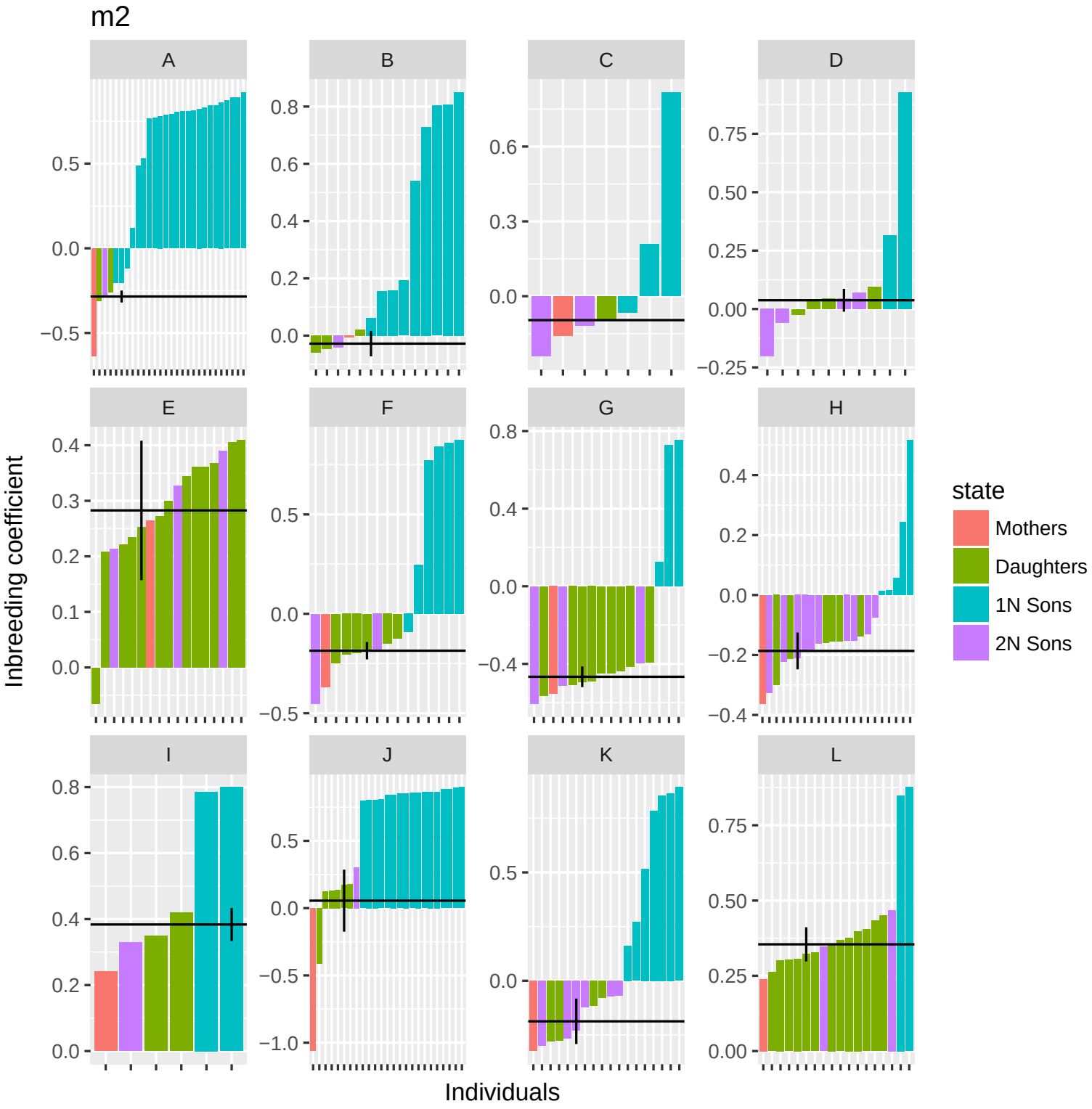

Figure 4.1: Inbreeding coefficient (F). Different colors represent the different types of individuals, as shown in the legend. Each plot is a family, each bar is an individual. Horizontal black bars show the mean inbreeding coefficient of daughters within the family +/− the standard deviation (vertical black line). In theory, mothers should have the lowest inbreeding coefficient of their family (highest heterozygosity)

After separating haploids from diploids, I performed exploratory analyses prior to association mapping. The goal of these is to assess the homozygosity rate of SNPs in diploid males and females and how many seem to fit the CSD pattern. As shown in Figure 4.2 and 4.3, there is no single SNPs thatis heterozygous in all females and homozygous in all diploid males. That can imply that the species has ml-CSD and there is no single locus that always fit the CSD pattern, or the restriction enzyme used in the RAD-seq protocol (ecoR1) did not cut in the CSD locus directly. It is very likely a combination of both explanations. It is also worth mentioning that, besides the number of SNPs, it makes little difference whether I use permissive (Figure 4.2) or stringent (Figure 4.3) parameters. The only notable change is that including more SNPs results in higher homozygosity for both sexes.

In this analysis, I excluded all consensus loci (where all individuals had the same allele) and did not consider SNPs with more than 2 genotypes in a single individuals in the heterozygosity calculation (counted as missing values), as they are likely due to contaminations.

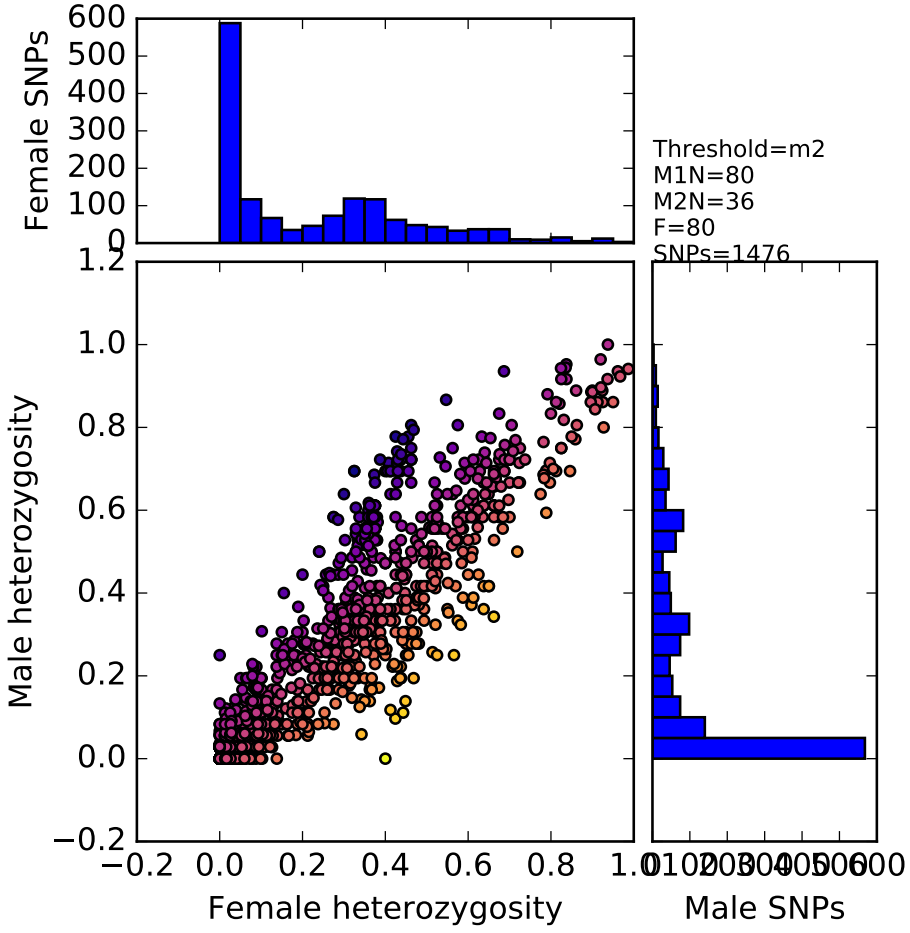

Figure 4.2: Homozygosity of female and diploid male SNPs where depth ≥ 5 (permissive parameters). Each point on the scatterplot is a SNP, and its coordinate are the proportion of females (x) and males (y) in which it is homozygous. Histograms allow to visualize the distribution of homozygosity for SNPs of each sex. The color code shows how SNPs fit the CSD pattern with lighter points being closer to it (i.e. more homozygous in males and heterozygous in females). Summary statistics on the top right show the threshold used, the number of haploid males excluded (M1N) as well as the number of diploid males (M2N), females (F) and SNPs included.

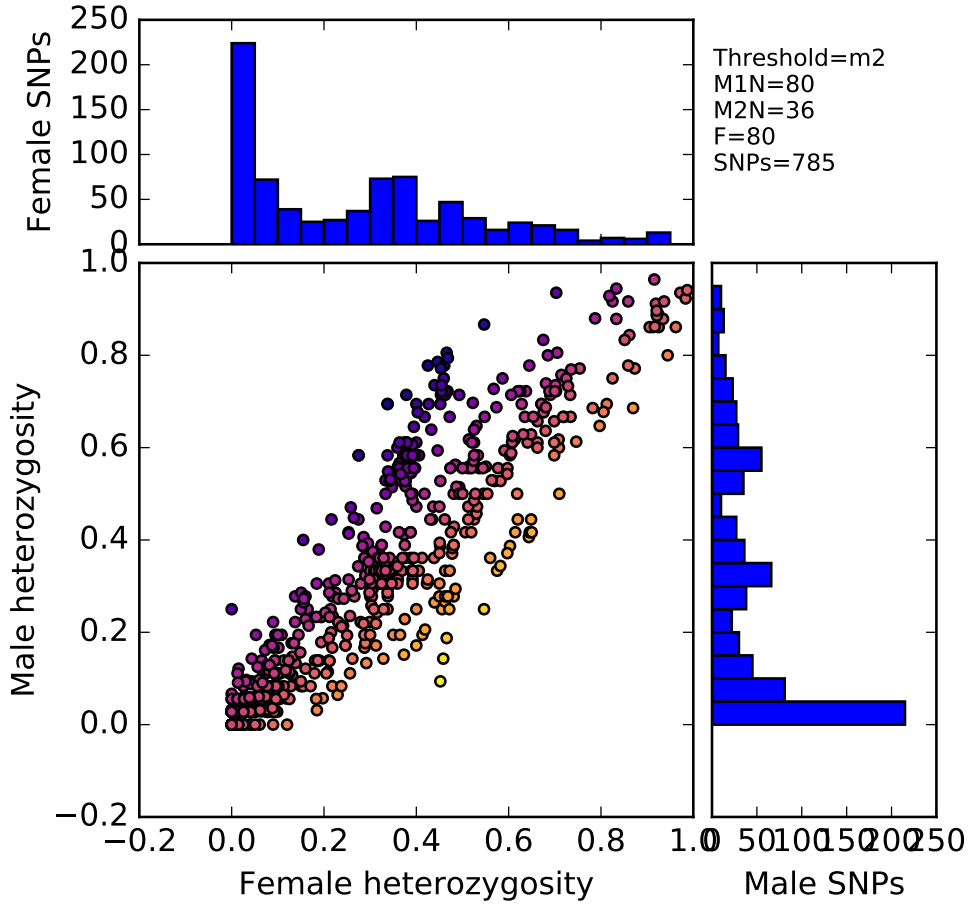

Figure 4.3: Homozygosity of female and diploid male SNPs $\geq$ 25 (stringent parameters). Each point on the scatterplot is a SNP, and its coordinate are the proportion of females (x) and males (y) in which it is homozygous. Histograms allow to visualize the distribution of homozygosity for SNPs of each sex. The color code shows how SNPs fit the CSD pattern with lighter points being closer to it (i.e. more homozygous in males and heterozygous in females). Summary statistics on the top right show the threshold used, the number of haploid males excluded (M1N) as well as the number of diploid males (M2N), females (F) and SNPs included.

Chapter 5

Cleaning genomic data

date : 23.06.2017

The first attempt at separating individuals based on inbreeding coefficient revealed several issues inherent to the data:

- The inbreeding coefficient is not clearly bimodal in all families (1 haploid mode and 1 diploid mode)
- Some loci have more than 2 haplotypes, although we are working with diploids.
- There are no loci that clearly stand out as perfectly 'CSD like' among all individuals, this should be looked at family-wise.

These issues will be solved one by one in this chapter.

5.1 Improving ploidy separation metric

The inbreeding coefficient was computed using all males/females. Also, the raw heterozygosity may be a more accurate metric than inbreeding coefficient. Both issue require the analysis to be done in a per-family fashion, either by re-running the populations program for each family, or implementing family as an additional population layer.
Date: 27.06.2017
Solution: I could switch to observed homozygosity, but I think it is also fine to keep this metric (I can always change it later). I did however, re-run the populations program separately for each family. I did not add a population layer to the popmap, because it would not have allowed for different loci blacklists between families when solving the third issue. Visualizing the distribution of inbreeding coefficients across individuals reveals a pretty good (?) separation of individuals (Figure 5.1).

Date: 12.07.2017 Update: This method of ploidy separation is flawed as it will overestimate the number of haploid males in families with few females. Indeed, with a lower standard deviation of female homozygosity, the threshold tends to the mean only.

Because the homozygosity of diploid offspring depends on its mother, but that of haploids are independent of mother background, I can probably use a universal threshold for all families. This way, the threshold can be equally conservative in all families and not biased towards certain of them. Plotting homozygosity of all individuals (Figure 5.2) revealed 2 separate meta-distributtions. The one on the left is a gaussian made up of several smaller ones (each smaller distribution is a family, with the mean depending on the mother), whereas the distribution on the right is made of a single gaussian containing all (or most) haploid males. Choosing a fixed conservative threshold in between should allow to keep more individuals and remove bias.

5.2 Loci with 3 or more haplotypes

There are two possible explanation for the existence of these loci. It could be either due to a contamination, in which case few individuals would present many loci with 3 or more haplotype. On the other hand, if few

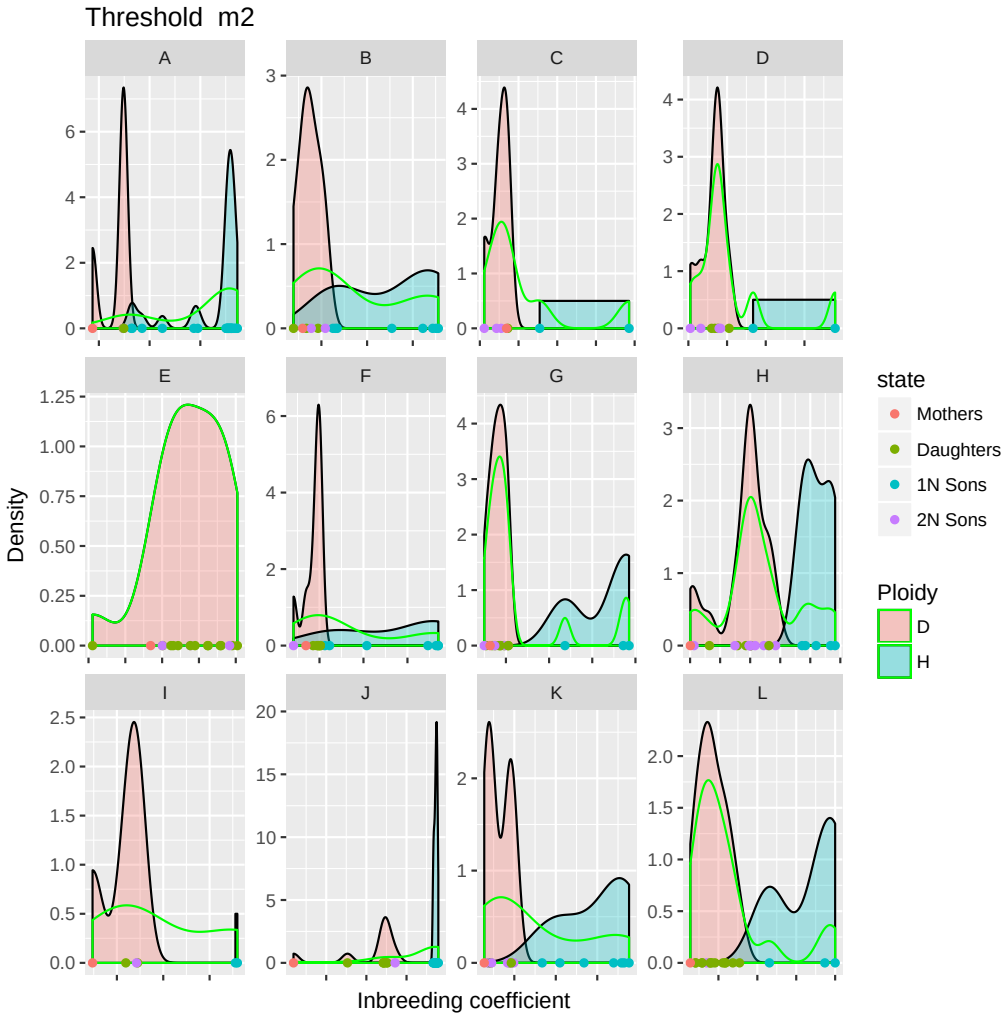

Figure 5.1: Distriution of individual inbreeding coefficient. The green curve represents the overall distribution of inbreeding coefficients within the family. The blue and pink areas represent the distribution of inbreeding coefficients for haploids and diploids, respectively, split using the m2 threshold defined at the beginning of chapter 4. Ideally, the green curve would follow a bimodal distribution clearly separating individuals by homozygosity.

loci have this issue in several individuals, it could be due to paralog merging. This can be solved by identifying which case is the most likely, and either excluding concerned individuals (first case), or loci (second case).

24.06.2017: Asked on STACKS google group about the haplotypes.tsv file. It is apparently normal to have more than 2 genotypes and depends on the parameters that were set in pstacks/ustacks. The answer I got, is that I should only work with the VCF file, which calls only 2 genotypes/site/individual. This imply figures 4.3 and 4.2 are not valid, as I misinterpreted the file content. I will re-generate these figures using the VCF file directly.

Date: 27.06.2017

Solution: I used the -012 argument in VCFtools to produce a genotype matrix from the populations output VCF file. This matrix encodes the genotype of each individual at each SNP with an integer representing the number of non-reference alleles present. These matrices were generated separately for each family and used to regenerate the figures 4.3 and 4.2.

The issue when visualizing the SNPs heterozygosity per family (Figure 5.3) is that the low number of individuals allow for few different heterozygosity values for any given SNPs, yielding a high number of 'perfectly' CSD-like candidates. Solving the next issue or grouping SNPs by contig/locus may help reduce this number, but ultimately only association mapping will allow to find the best candidate region.

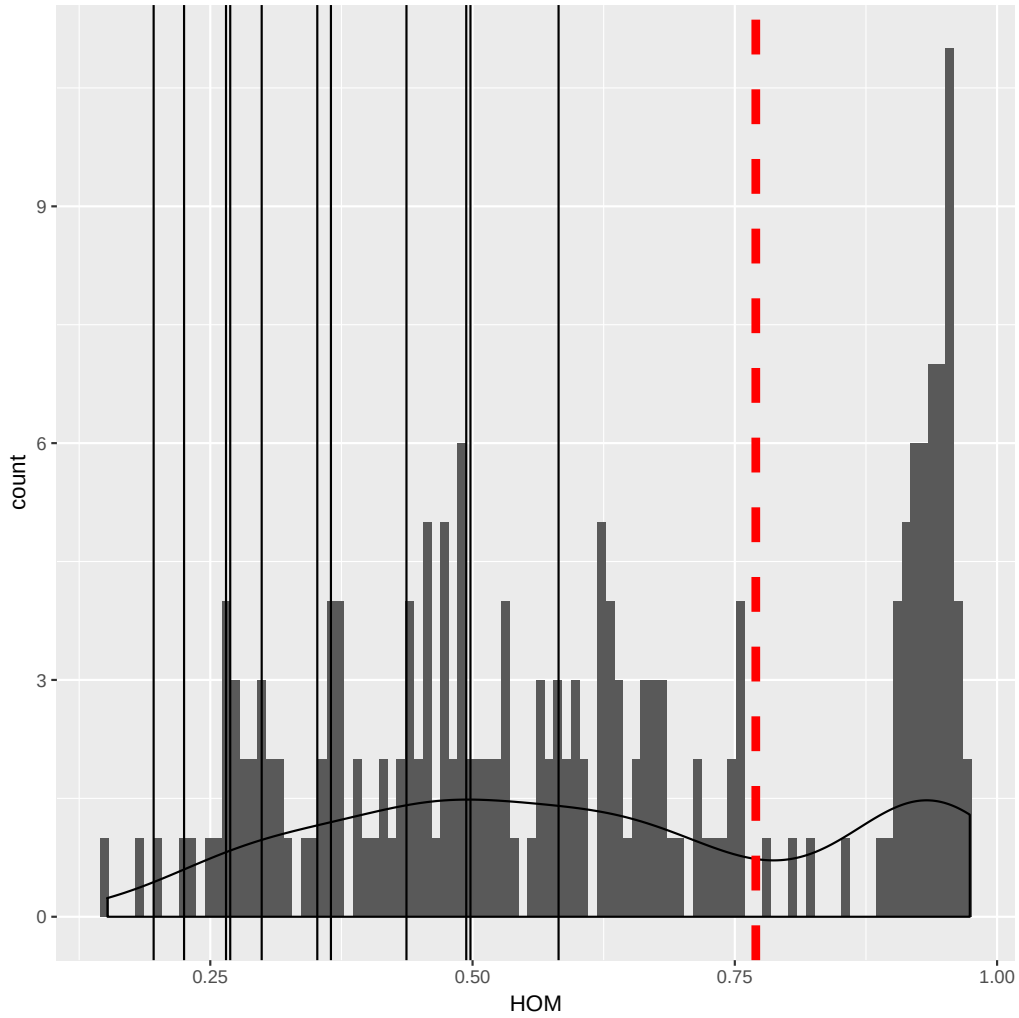

Figure 5.2: Distribution of the proportion of homozygous loci across all individuals. Continuous vertical black lines are the mothers values and the dashed vertical red line is the (visually) optimal threshold, fixed at 0.77.

5.3 No clear signal in whole population

This issue could be expected from the biological system because it is believed that there is ml-CSD in this species; there should be no single locus that will fit the CSD pattern across all individuals. This mean we need to identify different loci independently in each family. There are several things which can be done to work around this issue.

First, and most importantly, the populations program should be taking family information into account, but this was done already when solving the first issue (Improving ploidy separation metric). Second, the data should be cleaned by blacklisting loci that are homozygous in mothers in each family, as there is no way these will be heterozygous in their respective daughters and be CSD candidates.

Date: 29.06.2017

Solution: I re-ran the explo assoc script used to generate figure 5.3 and excluded all SNPs that are homozygous in the mother (Figure 5.4). It removed a significant number of SNPs from the set, but there are still way too many that fit the CSD pattern.

family	N SNPs
A	167
B	77
C	61
E	166
F	158
G	345
H	230
I	32
J	193
K	40
L	128

Table 5.1: Number of homozygous SNPs found in the mother of each family. Note family D is excluded since there was no genomic data available for this mother.

family	N SNPs
A	15
B	38
C	45
D	184
E	47
F	199
G	62
H	62
I	29
J	144
K	112
L	37

Table 5.2: Number of homozygous SNPs that fit the CSD pattern in each family.

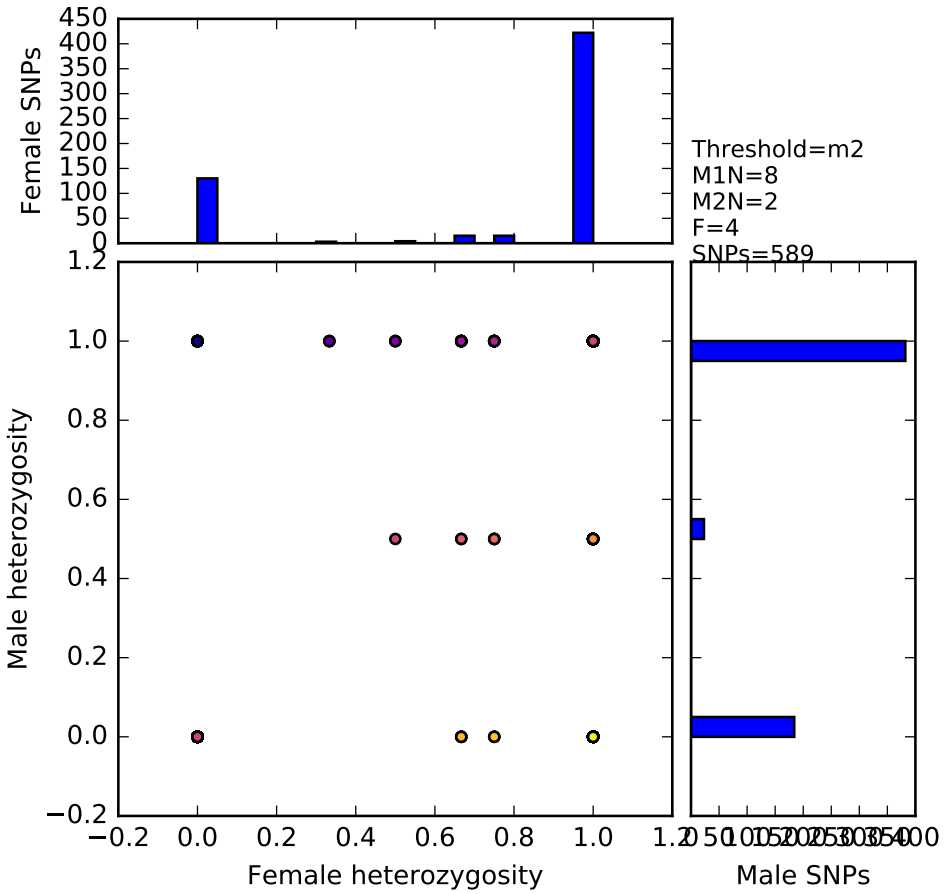

Figure 5.3: Example: SNPs heterozygosity across individuals of family B. The yellowness represents the proximity to CSD-like pattern (i.e. homozygous in all males and heterozygous in all females. Note: populations parameters set to : min depth (D)=5 and prop pop (r)=75

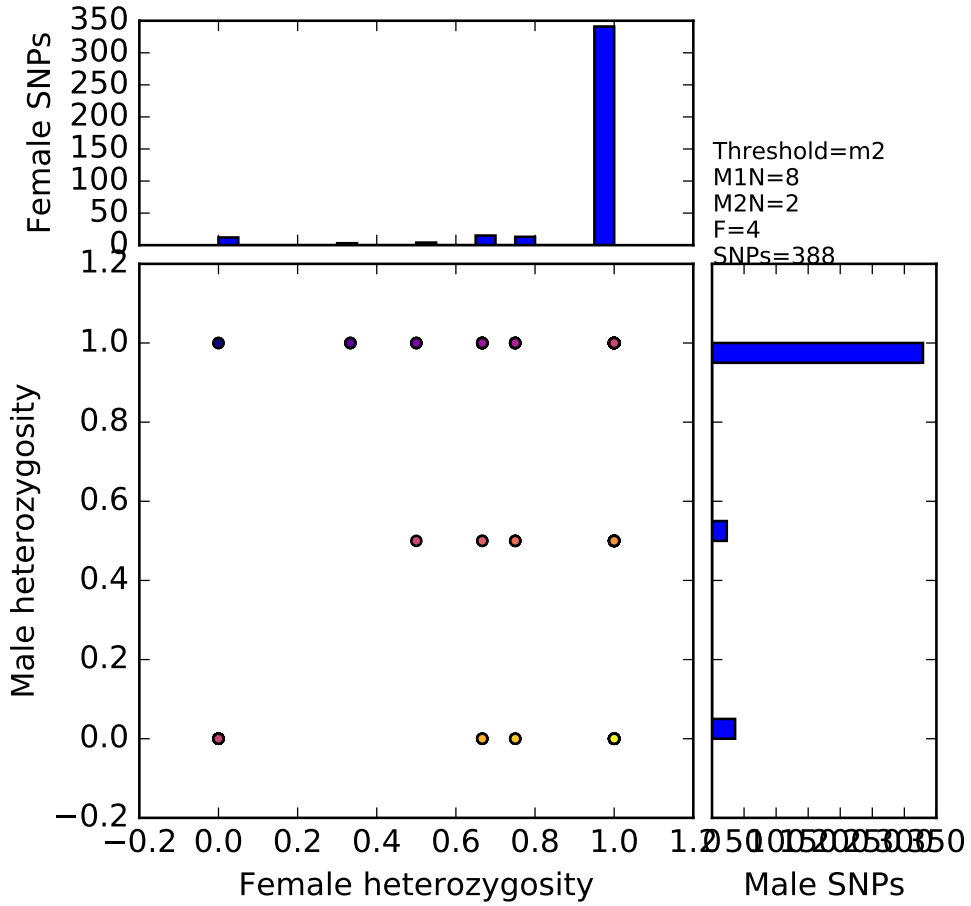

Figure 5.4: Example: SNPs heterozygosity across individuals of family B after removing SNPs that are homozygous in the mother. The yellowness represents the proximity to CSD-like pattern (i.e. homozygous in all males and heterozygous in all females. Note: populations parameters set to : min depth (D)=5 and prop pop (r)=75

Date: 24.07.2017
I removed loci that are homozygous in mother using a second technique:

- For each mother: use Pstacks snps file to find sample ID of all loci for which all position are "O" (for homozygous).
- Look up the sstacks matches file of each mother to retrieve the matching catalog ID
- exclude corresponding catalog ID for the family in populations output files in downstream analyses.

I am still not sure if this technique is safe / correct, the steps described above are currently implemented in the hom_filt function of Hom_M-F.R.
Date: 28.07.2017
I am now using a much simpler method which should be relatively correct. I use the output populations file batch_0.sumstats.tsv to remove the SNPs that have no alternative nucleotide in the female population. I am therefore assuming that if the mother is homozygous, all daughters are as well. That means I might keep some SNPs that are in fact homozygous in the mother, either due to sequencing errors or point mutations, but these events should be extremely rare.

5.3.1 Transmission bias

Here I check how frequently SNPs that are homozygous in a mother are heterozygous in its offspring and vice versa. (to be continued)

Chapter 6

Ordered genome

Date: 05.07.2017

Now that the data is cleaner, I will start looking for CSD. Casper provided me the genome with the contigs ordered by chromosomes (but not necessarily oriented correctly) he obtained with his linkage map. I will use this as the new reference genome and run the pipeline all the way from the mapping with it.

Date : 08.07.2017

I ran the whole pipeline on the ordered genome. Here are all statistics associated with all steps of the analysis previously described.

Statistics	Values
Assembly length (Mbp)	140.7
Largest scaffold (Mbp)	17.2
Mean scaffold size (kbp)	100
Median scaffold size (kbp)	25.7
N50 (kbp)	9518.5
Number of scaffolds	1408

Table 6.1: Assembly statistics for the ordered reference genome of *Lysiphlebus fabarum*

	tot	single	multi	miss
Number	170435458	94668507	44531856	31235095
proportion	1	0.555	0.261	0.183

Table 6.2: Mapping: Proportion and number of reads aligned on the ordered reference genome using BWA's aln algorithm with 4 mismatches allowed.

min˙cov	nloci	mean˙cov	sd˙cov
3	2694.0	50.0	69.4

Table 6.3: Pstacks: Summary statistics of stacks obtained with minimum coverage set to 3 in pstacks, using the ordered genome.

mismatch	mean˙loci	mean˙alleles
3	9072	34311

Table 6.4: Cstacks: Summary statistics of loci obtained with a mismatch value set to 3 in cstacks using the ordered genome.

Chapter 7

Number of CSD loci

It is possible to approximate the number of CSD loci heterozygous in a mother using the proportion of males among diploid individuals. Provided the loci are on different chromosomes, if n CSD loci are heterozygous in the mother, there should be $\frac{1}{2^n}$ of males among diploid, because as long as heterozygosity is retained at one of these loci, the offspring should be female.

7.1 Categorizing mothers

The mothers can be categorized based on this proportion of male offspring. The categories will reflect different combination of heterozygous CSD loci. Assuming the recombination rate is the same in different mothers, the different categories will depend only on the distance of the heterozygous locus/loci from the centromere. Loci further from the centromere will be undergo gene conversion more frequently.

There are different scenarii:

- 1 CSD locus: In this scenario, all mothers will be heterozygous at this locus and there should only be one category since the proportion of males among diploids will only depend on the distance of this locus from the centromere.
- 2 CSD loci, same chromosome: In this scenario, there should be a single category as only a recombination proximal to both loci would generat diploid males, therefore diploid male production only depends on the recombination rate of the locus that is closest to the centromere.
- 2 CSD loci, different chromosomes: This scenario would allow for 3 categories of mothers; those heterozygous at one, the other, or both CSD loci. Those heterozygous at the furthest locus only should have the highest production of diploid males as recombinations will be more frequent. Those with only the closest one would have a lower production. The females heterozygous at both loci would have the lowest production of diploid males (proportion ofshould be the product of the two other categories).
- 3 or more CSD loci: In case there are more than 2 loci, the number of categories would quickly increase beyond the scope of this study as we would not have enough families to detect the categories.

Note that scenario 2 is only happening if both loci are on the same side of the centromere, otherwise two separate recombination events would be required to cause male development.

Update: 02.08.2017

I need more families to build solid categories. The 12 families currently available seem to form distinct clusters, but more are needed to ensure a solid classification. Another issue is that we are not sequencing all offspring (for financial reasons) and we select the relative number of males and females to increase the power of all analysis (i.e. trying to balance the numbers of each). This will bias the numbers and we should account for it.

To address the latter problem, I propose the following strategy: If m out of M males and f out of F females are selected for sequencing, compute the proportion of diploid d among the m males. Then, compute the proportion of males p among total offspring as $p = \frac{M*d}{F+M}$, thus extrapolating the proportion of diploid males in the family to non-sequenced samples. This should not be biased as we select them irrespectively of ploidy.

However, families with small number of sequenced males are vulnerable to randomness and extrapolating might be dangerous in these.

To solve the issue of low number of families, we are preparing new libraries for sequencing, both expanding existing families and adding new ones. In total, we will add more than 200 individuals, but the exact number is subject to change.

Once the mothers have been categorized based on the proportion of males in their diploid offspring, I will use this category information to filter CSD candidates common to mothers within one category. Assuming scenario 3, for instance, there will be 3 categories. I will first filter the best hits common to the category with the lowest production of diploid males. I will then look for overlap with the two other categories separately.

Date: 24.08.2017

I used k-means clustering on proportion of males among diploid offspring to categorize families (Figure **??**). I assumed 3 clusters

Date: 25.08.2017

There may be an issue with the use of proportion of 2n males to categorize mothers; when we select individuals for sequencing, we try to balance the proportion of males and females to make sure we have enough of both sex. This will likely decrease the proportion of diploid males in families with many males as we do not use all of them. We could solve this by using the proportion of males in all offspring instead, assuming the proportion of haploidy is the same in all families. This way, we could use the total sex ratio of the family (including non-sequenced individuals) to categorize mothers. I should ask Casper if he has this information and the opinion of Tanja on that matter.

7.2 Distance to centromeres

To validate hits, I will use the information of the distance from centromere. The position of the centromere in the scaffold (i.e. metacentric vs telocentric chromosome) can be inferred from the recombination rates along the chromosome. The method I am going to use for this is to measure the proportion of offspring homozygous at each SNP that is heterozygous in the mother (i.e. proportion of recombinant offspring at that SNP).The centromere should be the region with the lowest value. The hits obtained for CSD candidates in the different categories should be coherent with this distance; the hit in the family with a low diploid male production should be closer to the centromere.

Modelling the recombination rate along chromosomes using a weighted loess (lowess) curve revealed local minima that could contain centromeres, however the position of these minima can vary strongly depending on the span (smoothing parameter) allowed for local regression when building the curve. This is likely due to the small number of individuals, or rarity of recombination events. The proportion of homozygosity also tends to decrease far from centromere, this is likely due to multiple recombination events restoring heterozygosity. Note the curve uses the proportion of recombinant offspring within a family at a given SNP weighted by the total number of offspring in that family.

I tried locating the centromere both by running populations with all individuals pooled together (Figure 7.1) and separately for each family (Figure 7.2). If I want to use the minimum estimate of local regressions objectively to locate the centromere, I will need to use some form of cross validation to estimate the span. I could also use a visually optimal value for span; the default value used in ggplot's geom_smooth() is 0.75 and seems reasonable on my data (does not seem too noisy).

Date: 12.08.2017

To obtain better result when trying to locate centromeres, I used the genomic output from populations to include fixed SNPs. I subsequently removed all SNPs that are **either missing or homozygous** in the mother from its whole family (including itself). The variation induced by the span parameter when using local regression is now much smaller. To make sure the centromeres are called correctly, I tried a second simpler method, which consists in computing means over a sliding window on each chromosome. I tried both methods with several different parameter values (Figure 7.4). The methods yielded similar results and I visually selected reasonable parameter values for span and window size to compare the centromeres positions inferred by both methods (Figure **??**). Centromeres are merely regions and have no objective precise genomic position, besides I will only be interested in relative differences in distance from centromeres to select loci, therefore the parameter

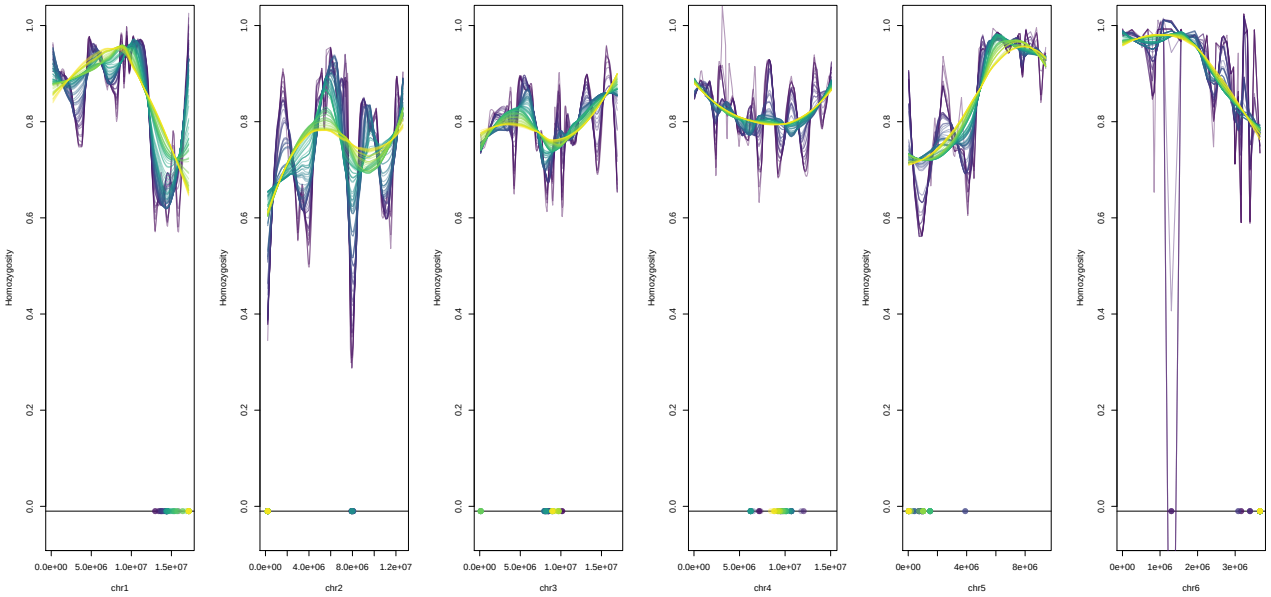

Figure 7.1: Modelling recombination rates along chromosomes using local regression with second degree polynomials on proportion of homozygous individuals as a proxy. Populations was run on all individuals pooled together (r80, mindepth:3). Colors represent different span value for the local regression and colored dots are the inferred centromere position (i.e. the minumum of the curve) with the corresponding span value.

values I chose will probably have little to no effect on the results.

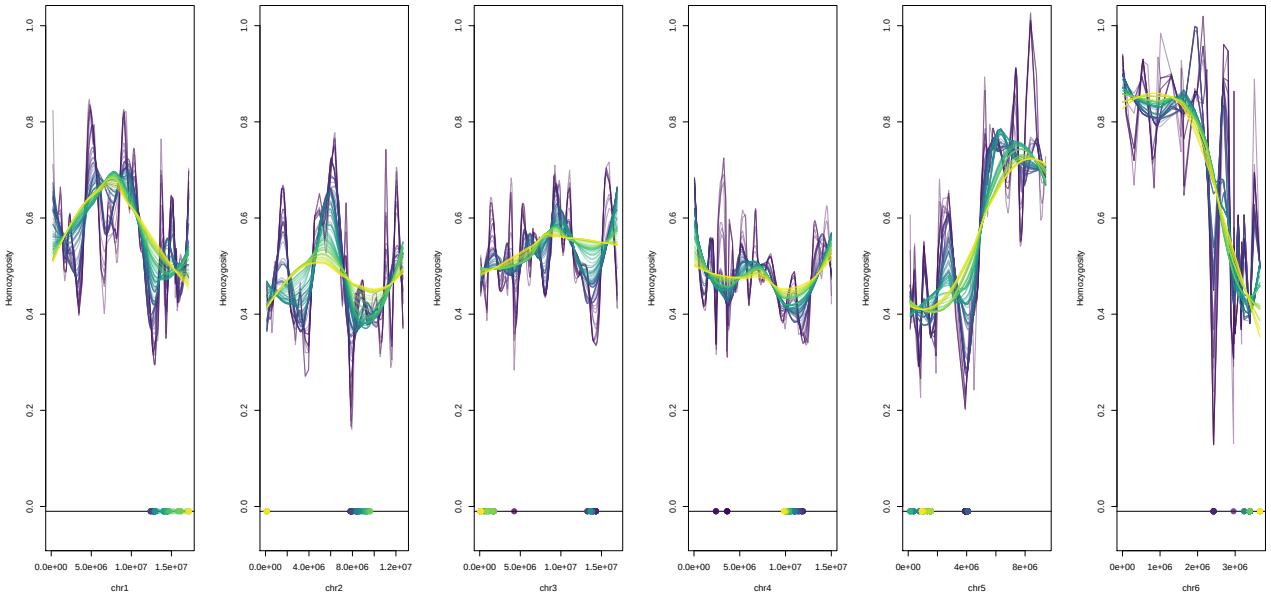

Figure 7.2: Modelling recombination rates along chromosomes using local regression with second degree polynomials on proportion of homozygous individuals as a proxy. Populations was run on separately for each family, SNPs where the mother is homozygous were removed in each family (r80, mindepth: 3).Colors represent different span value for the local regression and colored dots are the inferred centromere position with the corresponding span value.Colors represent different span value for the local regression and colored dots are the inferred centromeres position (i.e. the minumum of the curve) with the corresponding span value.

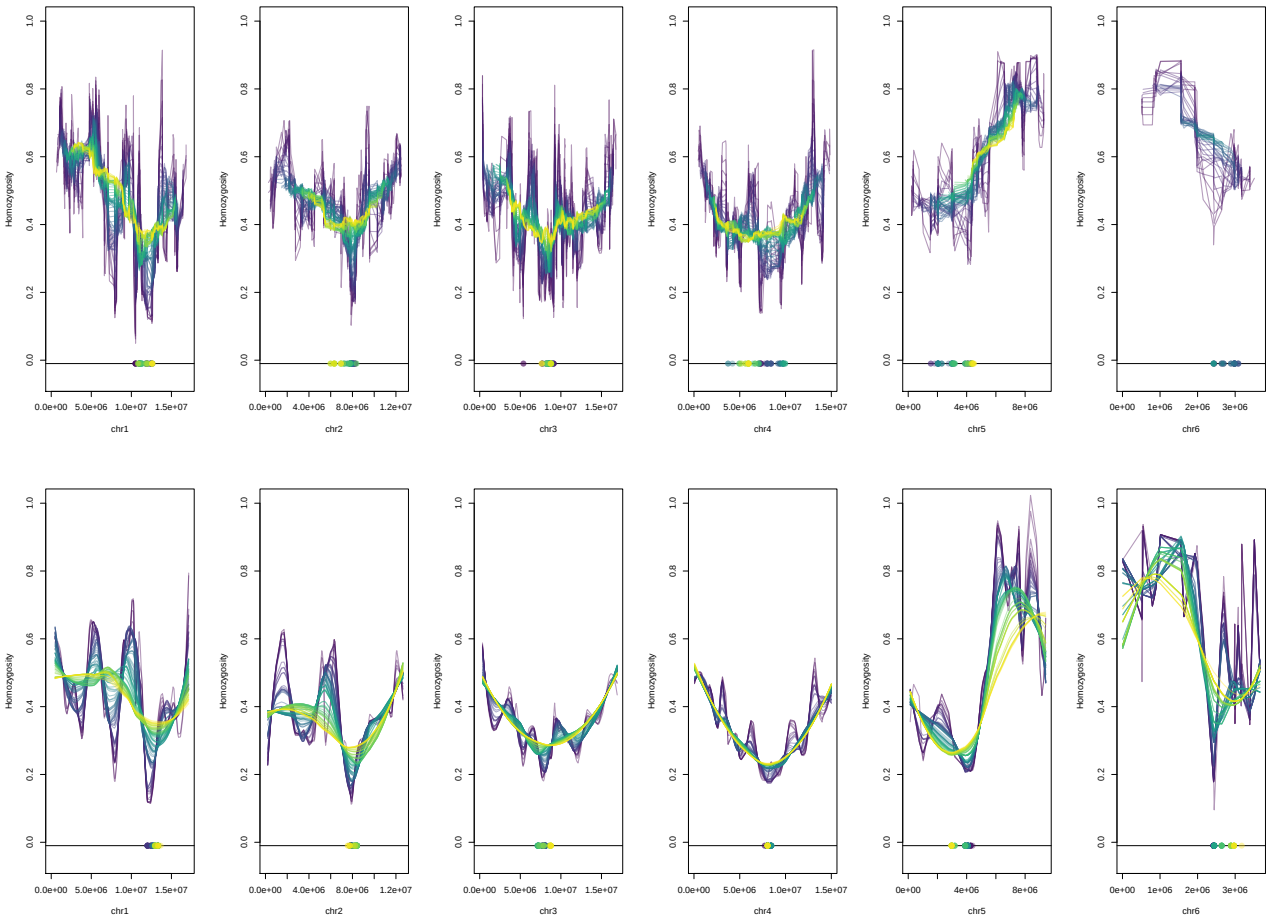

Figure 7.3: Comparing moving averages and local regression with second degree polynomials to approximate recombination rates. Populations was run on all individuals pooled together. Fixed SNPs are included and SNPs where the mother is homozygous or that were absent in the mother were removed in each family (r80, mindepth: 3). Range of parameter tested: span varied between 15% and 100% of observations by intervals of 1% for local regression and window size between 3 and 80 observations with intevals of 1 for moving averages. Colors represent the different span or window size value for the local regression and colored dots are the inferred centromeres position (i.e. the minimum of the curve) with the corresponding parameter value.

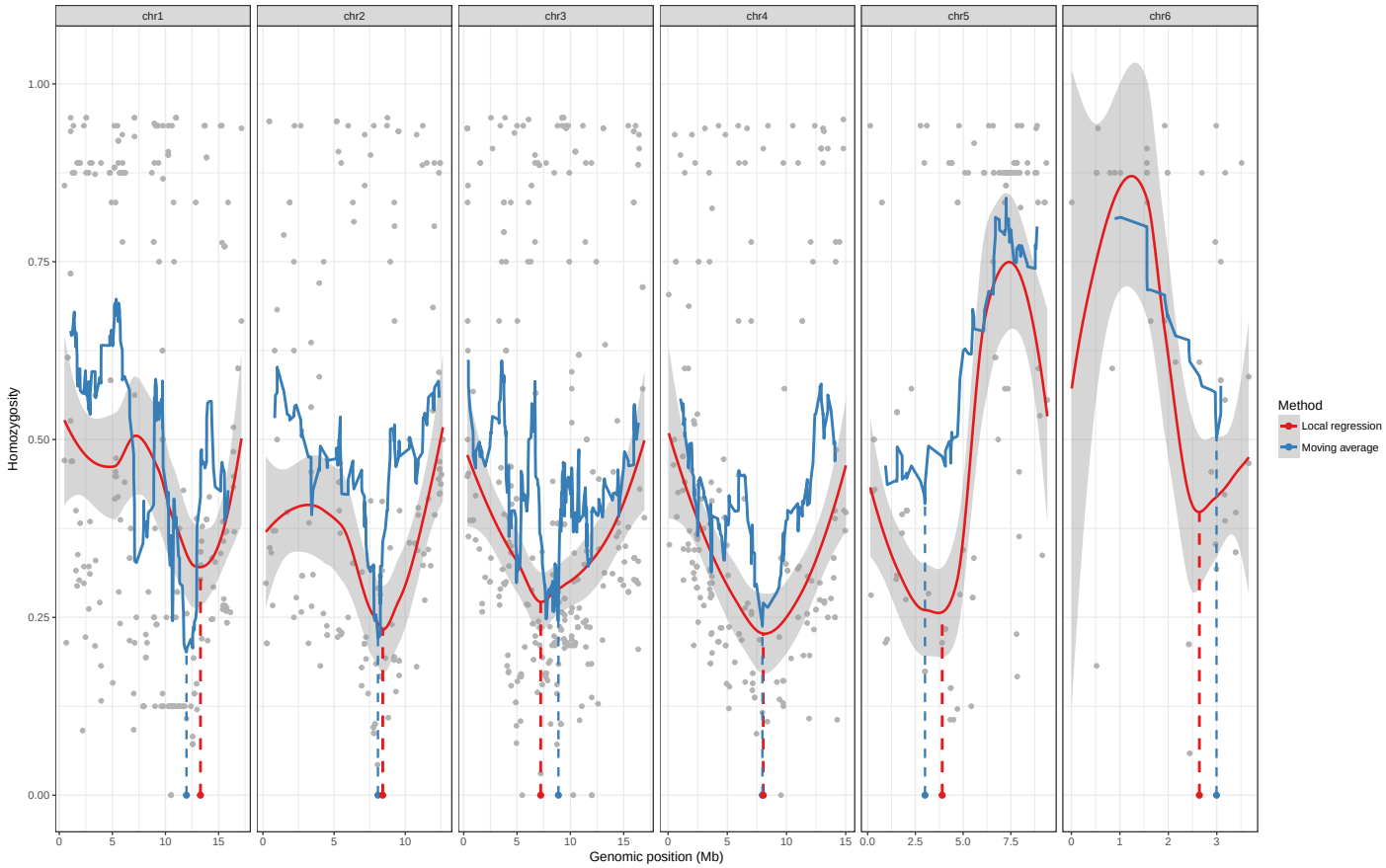

Figure 7.4: Comparing moving averages and local regression with second degree polynomials to approximate recombination rates. Populations was run on all individuals pooled together. Fixed SNPs are included and SNPs where the mother is homozygous or that were absent in the mother were removed in each family (r80, mindepth: 3). Parameters selected for comparison: window size: 20, span: 0.75. Dotted lines show the inferred centromeres position (i.e. the minimum of the curve) with both methods.

Chapter 8

Male-Female Fst

Date: 17.07.2017
During preliminary analyses, I noticed recurrent peaks in Fst between diploid males and females in different families. This could indicate the presence of male-deleterious alleles. To further investigate this possibility, I analysed Fst again using all individuals (including haploids).

I first looked at Fst over all 6 chromosomes in the assembly, averaging the Fst at each SNP across family to have an overview (Figure 8.1), and then splitting the families to identify recurrent peaks (Figure 8.2).

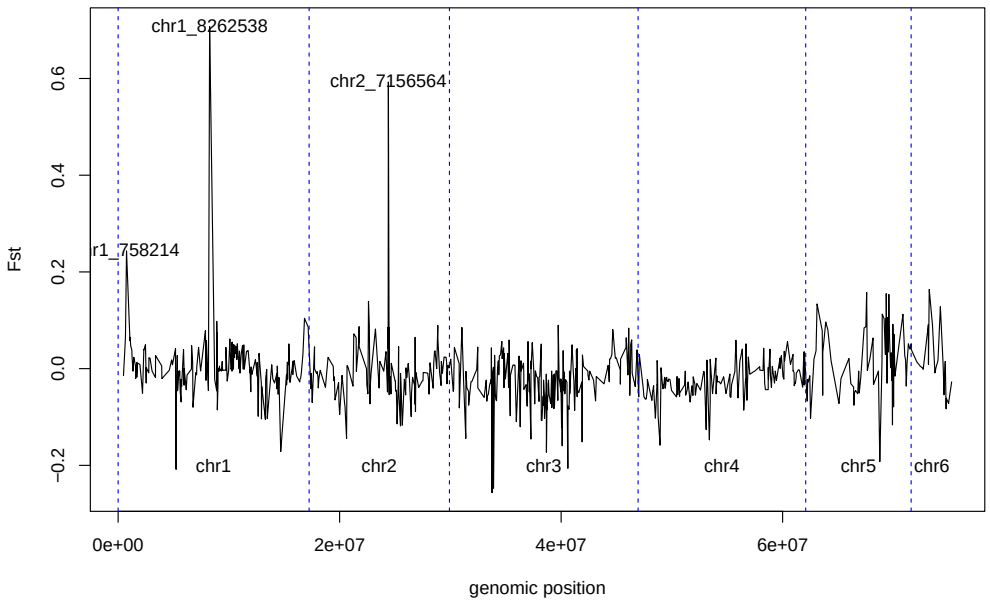

Figure 8.1: Fst values averaged at each SNP across all families. All individuals, including haploids and mothers are included in the analysis and top scoring peaks are labelled with the SNP position in the format "chromosome_basepair" where basepair is the position within the chromosome. Note: populations parameters set to : min depth (D)=20 and prop pop (r)=80

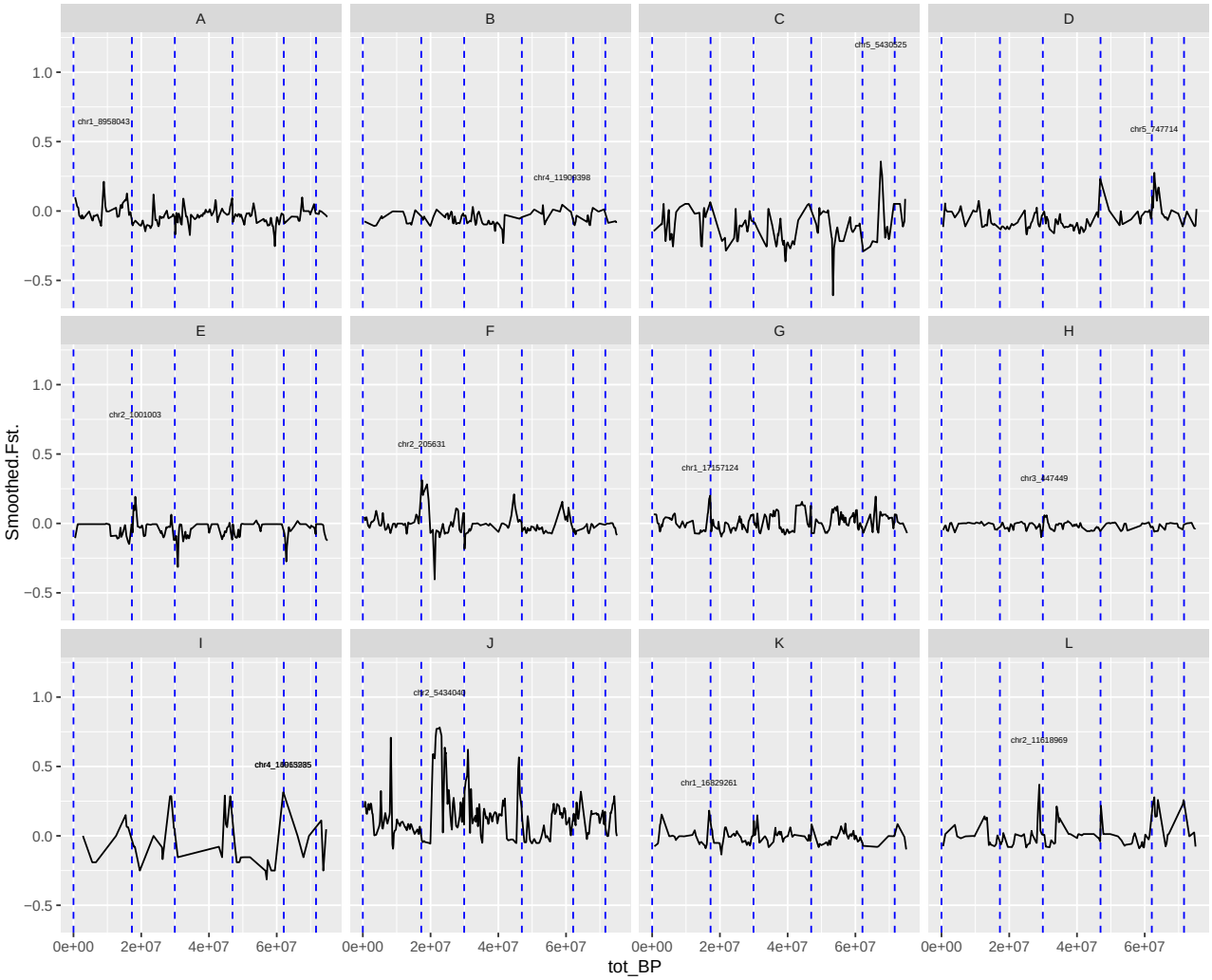

Figure 8.2: Fst values at each SNP by family. All individuals, including haploids and mothers are included in the analysis and top scoring peaks are labelled with the SNP position in the format "chromosome_basepair" where basepair is the position within the chromosome. Note: populations parameters set to : min depth (D)=20 and prop pop (r)=80

The issue with averaging the Fst over families as in figure 8.1 is that some SNPs can be sequenced in only one family, therefore the averaged value will only depend on that family. If we merge all families' curve into a single plot, it appears there is no single peak common to all families. This can either mean that there is no interesting signal, or if that the signal depends on the family's genetic background, possibly due to an epistatic effect. Here, however, I do not have the sample size to test for it.

This issue will be dropped from the analysis as the effect is much weaker than what we expected at first and probably just random noise.

Chapter 9

Association mapping

9.1 Coarse analyses

Before running any association mapping analysis, I will scan the genome for region that are frequently homozygous in males and heterozygous in females. This is done only after removing SNPs that are homozygous in mothers and individuals that are defined as haploids with fixed homozygosity threshold from the populations analysis.

Date: 21.08.2017

The statistic I use when scanning the genome for CSD is $CSD = \frac{Hom_m + Het_f}{2}$ where Hom_m and Het_f are the proportions of homozygous males and heterozygous females respectively at each SNP. If the value is 1, all individuals respect the CSD pattern (all males are homozygous and all females are heterozygous) if it is 0, no individuals respect it (all males are heterozygous and all females homozygous). This metric should not be biased by the gradual decrease in heterozygosity along the chromosome arms as the same weight is given to males and females regardless of numbers.

9.2 Case-control association test

Date: 18.08.2017 I will use a simple case-control odds ratio to identify CSD hits. I will first do it without taking families into account. I will then perform this test separately for each category of mother and finally I will use distance to centromeres to refine the lists of hits (c.f. Chapter: 7 Number of CSD loci").

The odds ratio are computed fo each SNP using the following observed (O) contingency table where each cell is the number of genotypes corresponding to the category:

	Homozygous	Heterozygous	Total
Male	Mo	Me	Mt
Female	Fo	Fe	Ft
Total	To	Te	Tt

Accordingly, the expected (E) numbers are given by:

	Homozygous	Heterozygous	Total
Male	$\frac{Mt*To)}{Tt}$	$\frac{Mt*Te)}{Tt}$	Mt
Female	$\frac{Ft*To)}{Tt}$	$\frac{Ft*Te)}{Tt}$	Ft
Total	To	Te	Tt

The χ^2 can then be measured as follows:

$\chi^2 = \frac{(O(Mo)-E(Mo))^2}{E(Mo)} + \frac{(O(Me)-E(Me))^2}{E(Me)} + \frac{(O(Fo)-E(Fo))^2}{E(Fo)} + \frac{(O(Fe)-E(Fe))^2}{E(Fe)}$

I applied this approach on grouped (i.e. single population run with all families) pooled (proportion of homozygous individuals calculated using all families for each SNP) data to have a first insight on the potentially interesting regions (Figure 9.1). If I want to get reliable results however, I will need to use a different association mapping technique that takes family information into account and to perform the test separately on each category of family (see section: 7.1, "Categorizing mothers").

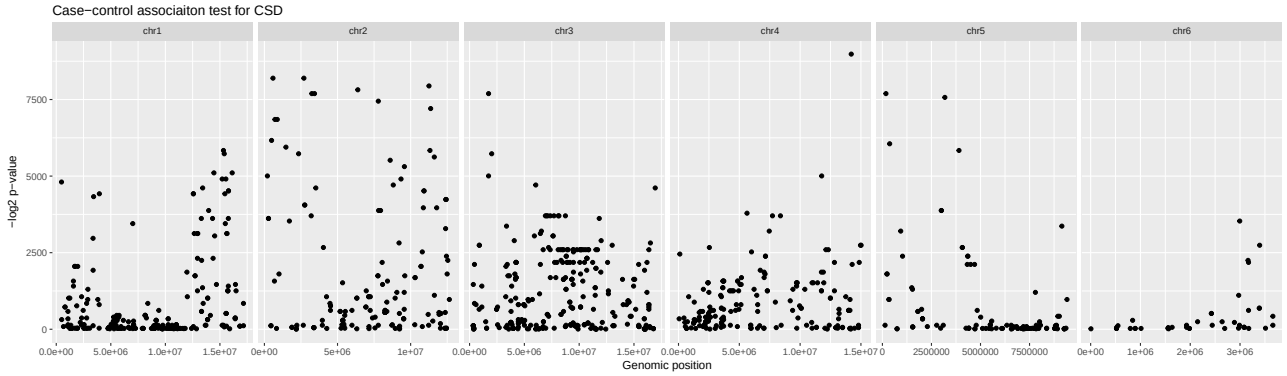

Figure 9.1: Manhattan plots with log2 p-values for the case-control association test on CSD. Populations was run on all families together and for each SNP, all individuals were pooled to compute the proportion of homozygous individuals

9.2.1 Case-control on categories

Here, I apply the test after subsetting individuals by categories, inferred using diploid male production as detailed in section 7.1 "Categorizing mothers".

I will do it considering different number of categories, corresponding to different scenari regarding the number of CSD loci. Figure 9.1 already addresses the unlikely scenario of single-locus CSD. Here, I will deal with 2 and 3 loci scenari, with 3 and

9.2.2 Refining hits using centromeres

9.3 Family-based association mapping

I will try to adapt the "generalized family-based association test for dichotomous traits" (GDT) published by Chen W.-M., Manichaikul A. and Rich S.S. in The American Journal of Human Genetics (2009). This test should work both in inbred and non-inbred families and takes family information into account.

Here, I will modify the test presented in the publication to test for association of homozygosity with male phenotype instead of a particular allele with a disease. I will also need to make it work with a single parent.

Date 23.08.2017

After investigating the proofs of the score used in the family based test mentioned above, I realized adapting it would not make sense; it uses pedigree information and relies on the differences in kinship coefficients or proportion of IBD alleles. Since I have asexuals, all kinship coefficient in the families are 1, they are all siblings and it does not make much sense to include pedigree information. I will instead focus on a better implementation of the case-control test.

Chapter 10

Coverage analysis

In this section, I check wether there are individuals or genomic regions that have abnormally high or low coverage, I identify the potential reasons for these abnormalities and eventually, I will remove the concerned elements from the analysis.

10.1 Individual stats

Prior to coverage, I had a look at the raw reads statistics. On average,each sample has 803951 reads and the (ordered) reference genome is 140.7 Mb long. This makes for an average of $5.7 * 10^{-3}$ reads per basepair. In comparison, the 3 datasets used in the Paris 2017 paper (Roadmap to STACKS) have $9.6 * 10^{-4}$, $2.7 * 10^{-3}$ and $8.6 * 10^{-3}$.

I first analysed the coverage on a per individual basis. That means for each individual, it has been averaged over all polymorphic SNPs kept throughout the STACKS pipeline. The mean sample coverage is relatively high and follows a normal distribution centered around 119X with a standard deviation of 33. There are two outliers with exceptionally high coverage (at 228X and 270X, respectively) which I might consider removing. The coverage is not different across families (Figure **??**). Coverage is not affected by the family of an individual (it is not affected by ploidy or sex either).

Analyzing the number of sites (SNPs) per sample revealed that it was strongly affected by the family (Figure 10.2). This could be caused by storage duration, reagent quality or manipulations. If so, I cannot account for this and will either need to get rid of lowest quality families (such as I) or get more individuals.

Update: 06.08.2017

To investigate whether the inter-family variation in the number of sequenced sites was due to differing amounts of sequenced DNA, I compared the number of sequencing reads per family (Figure 10.3). There is interfamily variation, but it seems relatively low, compared to the number of sites in figure 10.2. This variation is likely due to the low number of samples and is probably not meaningful. Moreover, the order of families, when sorted from highest to lowest is different for number of sequenced sites and number of reads.

10.2 Genome coverage

To identify regions of high or low coverage, I extracted information about the mean coverage per family at each polymorphic site that passed the STACKS populations filters. I then binned SNPs by 1kb regions (Figure 10.4). Some SNPs have extreme coverage values (¿600X) and might be caused by repetitive regions or paralogs that have been merged as a single locus.

10.3 Lowering the filters

Date: 09.08.2017

The main issues I am facing in this chapter are 1) the abnormally high coverage across my samples and 2) the high variation in the number of sites between families.

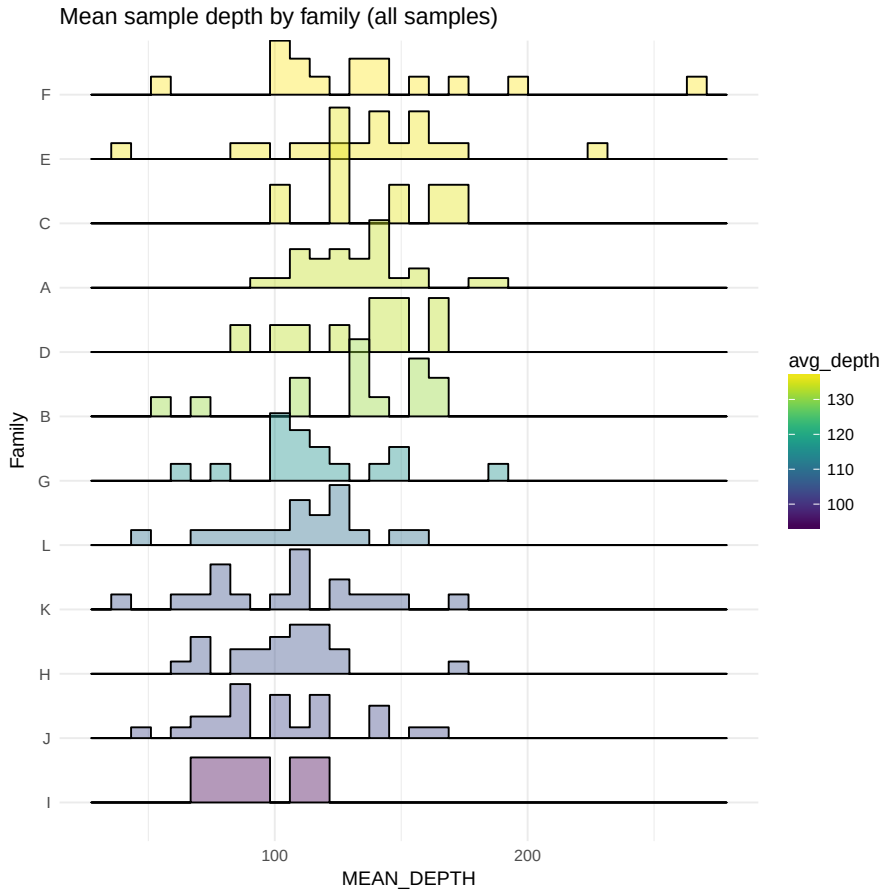

Figure 10.1: Sample depth, averaged across all polymorphic sites which passed STACKS populations filters. Individuals of each family are plotted together and histograms are colored according to the family's mean depth.

The reason the high mean coverage per sample was worrying is because I have about the same amount of reads per sample as the 3 datasets described in the roadmap to stacks paper (Paris et al, 2017), yet i have at least twice more coverage per sample. After re-reading the paper, I found out this was simply a consequence of my stringent filters at the populations stage, which they do not use (see citation below).

"Despite controlling for the mini- mum read depth of alleles at the ustacks phase of the pipe- line, many studies also incorporate a minimum stack depth required for individuals at a locus in the populations mod- ule of STACKS (e.g. Gaither et al. 2015; Ivy et al. 2016; Kjeldsen et al. 2016). Such a method is undesirable, as read depth has already been accounted for by the SNP model. Once the SNP model has made a determination, its evaluation should be trusted and using further non-statistically based limits on depth of coverage is ill advised and will result in the arbitrary dropping of loci."

I will follow their advice since my coverage is absurdly high, and I would benefit a sharp increase in the number of loci by removing this filter.

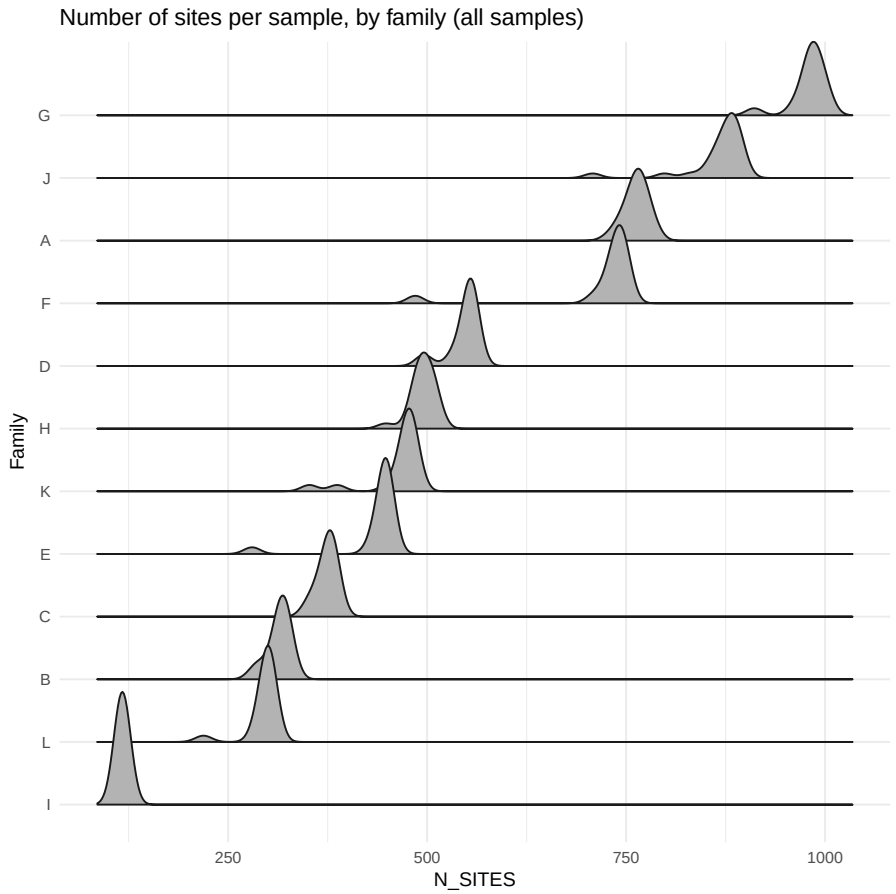

Figure 10.2: Number of polymorphic sites (SNPs) per individual that passed the STACKS populations filters. Distribution of individuals is shown by family. There is a strong clustering of families.

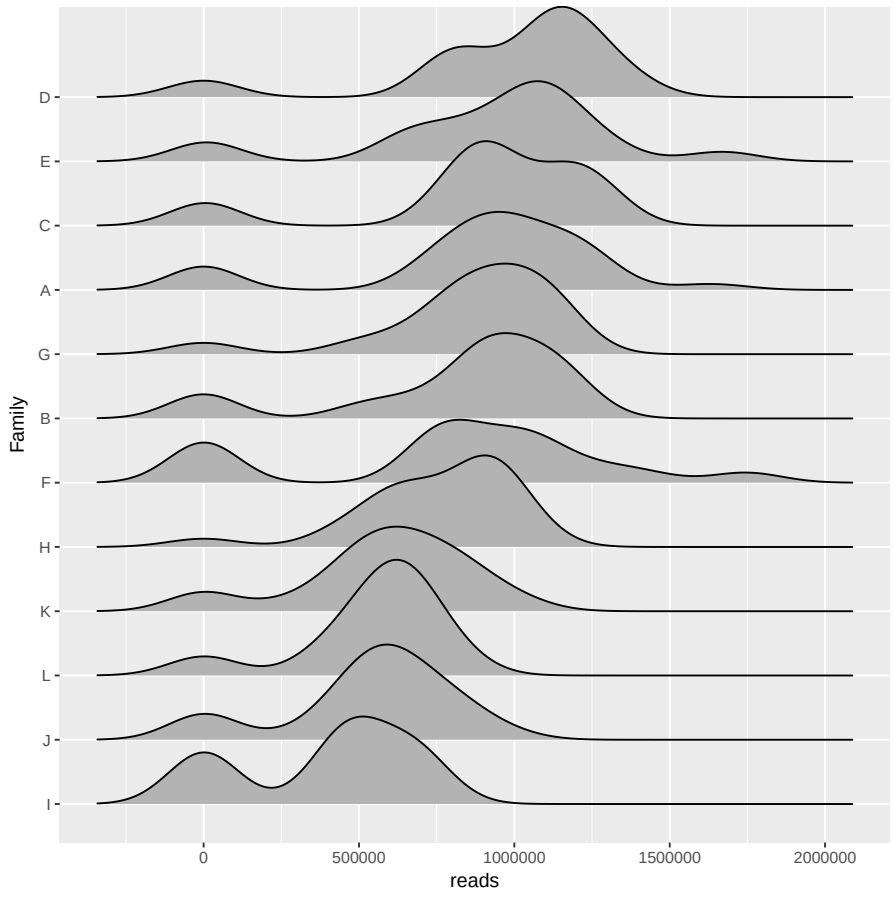

Figure 10.3: Number of sequencing reads per individual, before mapping or any filter. Distribution of individuals is shown by family. The inter-family variation is not particularly high relative to intra-family variation.

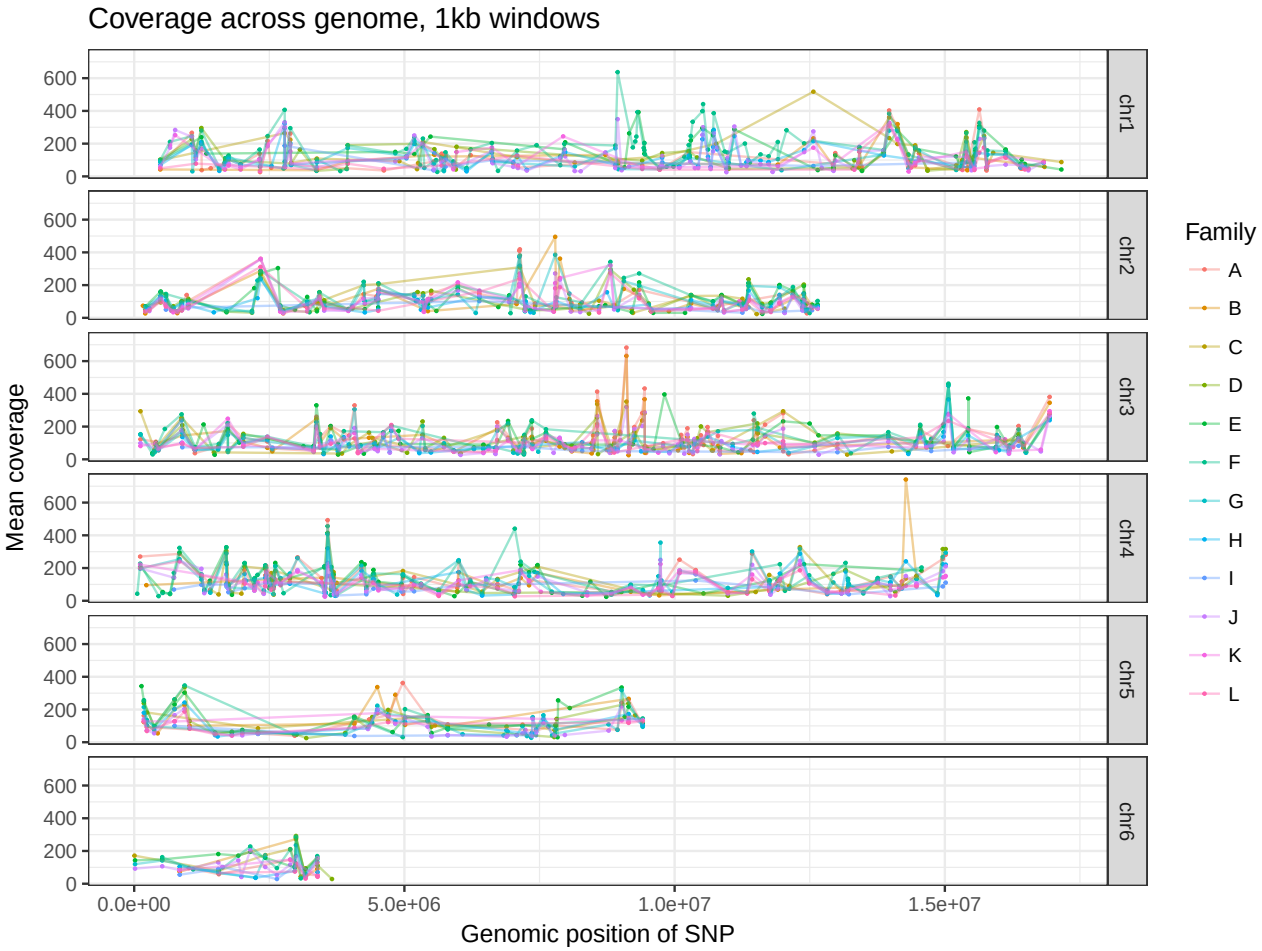

Figure 10.4: Mean depth per family at every SNP that passed the populations filters.

Chapter 11

Additional samples (lib10 + lib10b)

Date: 26.08.2017
Libraries 10 and 10b got back from the samples and I have processed them through the pipeline. Library10 suffered a dramatic loss in read count due to an unidentified technical problem (10x less reads than lib10b).
IMPORTANT: W10 appeared twice in the name list: once in library 10b and once in what will later be library11. Therefore I shifted the numbering by 1 for family W in library11. I will need to make sure this is the case in the barcode file as well.

Date: 28.08.2017
After removing all samples from library10 (see subsection 11.1.2 "Coverage", below). I still have NNN diploid individuals, distributed across FFF families.

11.1 Data description

11.1.1 Family composition

With the new samples, I now have 239 diploids across 32 families. Many families are relatively small, however (Figure 11.1) and relying on sex ratio among diploid offspring to categorize families could be unreliable.

11.1.2 Coverage

The coverage across genome (Figure 11.3) did not change much by adding the new libraries, however looking at the coverage per individuals(Figure 11.2) and number of site per individuals (Figure 11.4) confirms the poor quality of library 10. These individuals barely passed the filters i have put in place, but overall they still have very low read counts and including them exposes my whole analysis to technical biases. I will therefore remove all samples from that library. Note that I'll also remove W10 since it is the only healthy sample left in the family and all females are lost (they were in lib10).

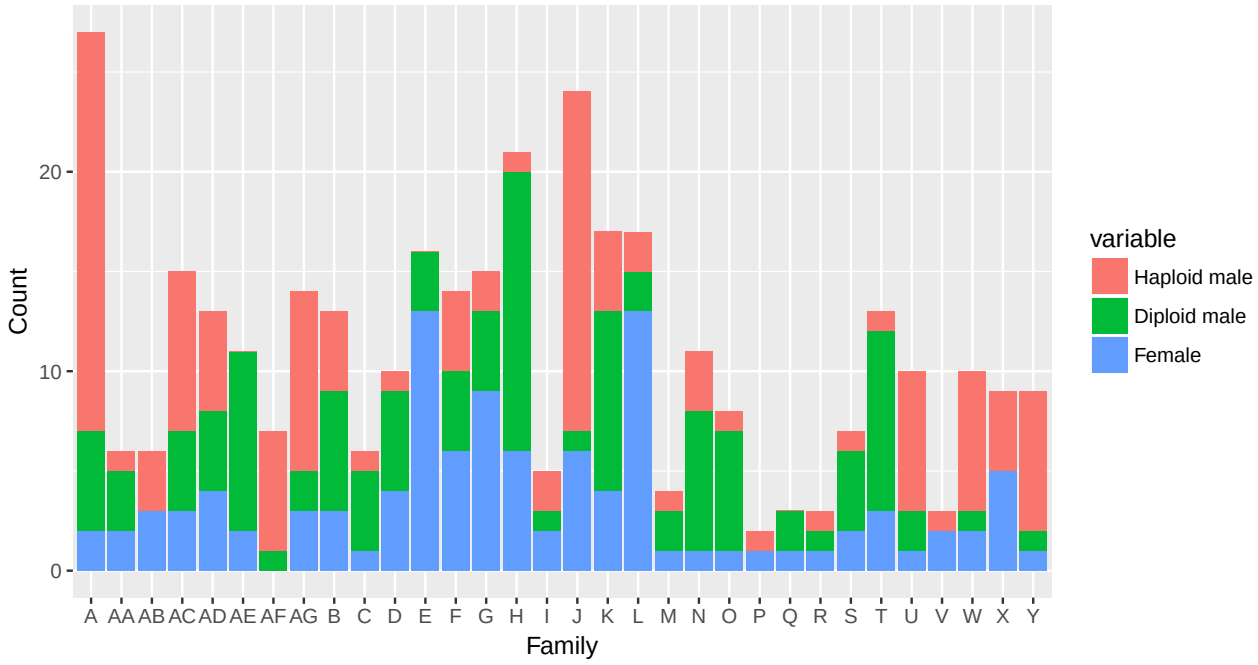

Figure 11.1: Composition of offspring per family when including new samples from library 10 and 10b.

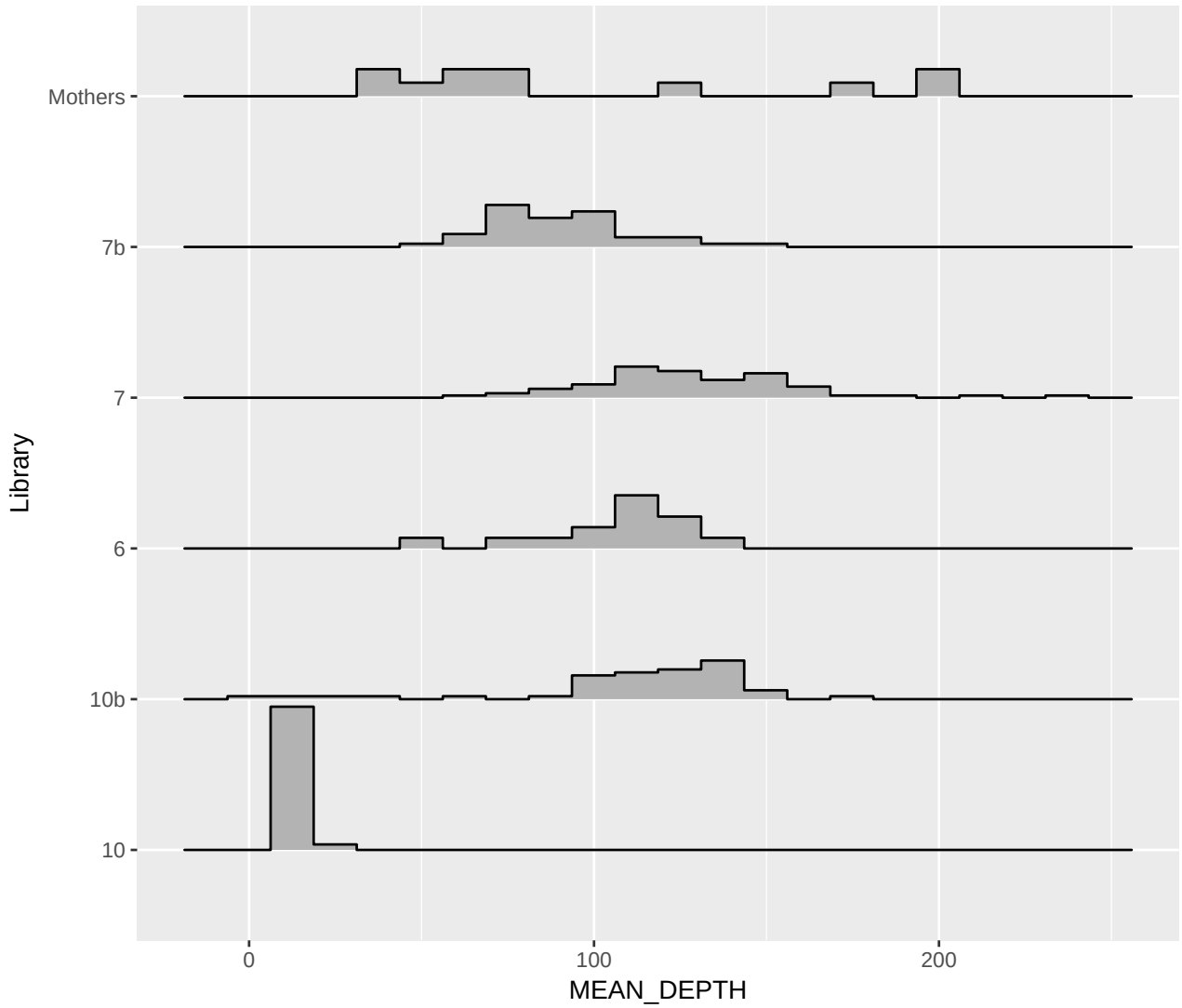

Figure 11.2: Distribution of the mean sequencing depth per individuals in each library including the new ones (10 and 10b).

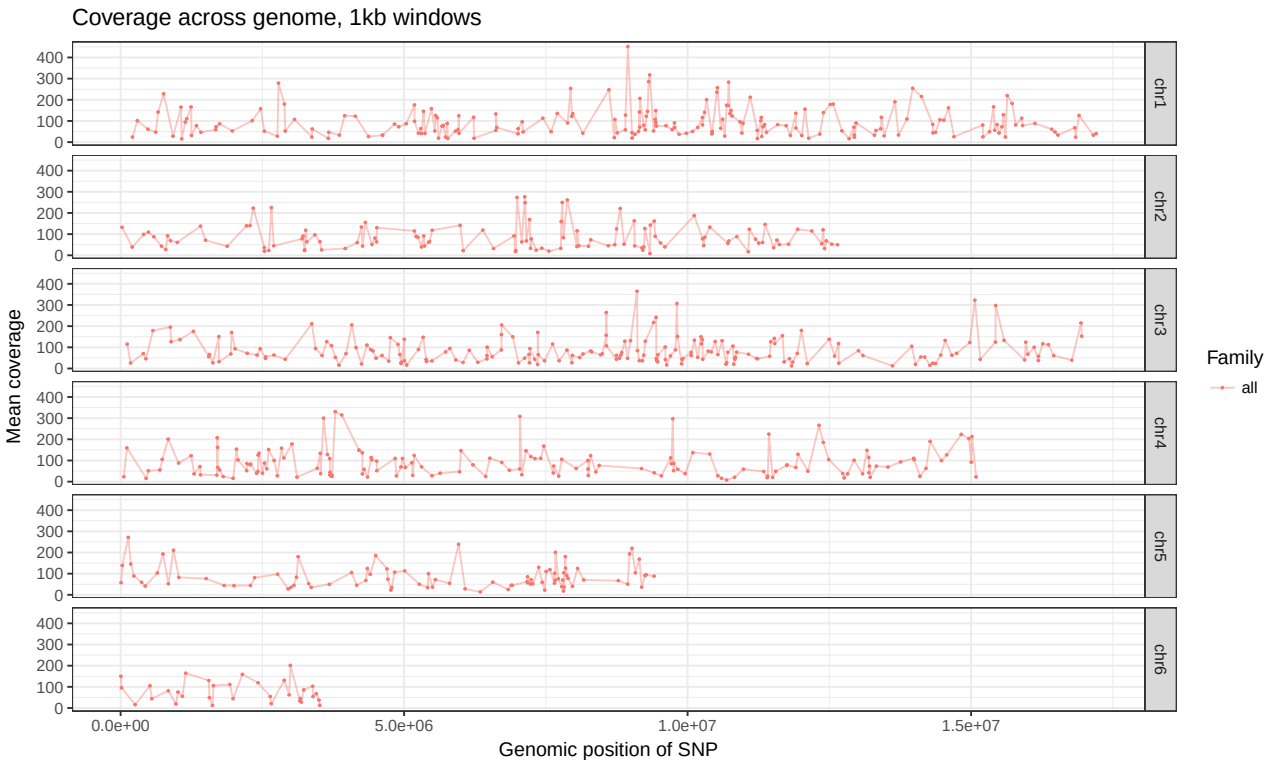

Figure 11.3: Sequencing depth across genome. Depth is averaged over all samples at each sites and values are averaged by 1kb windows. New samples from libraries 10 and 10b are included

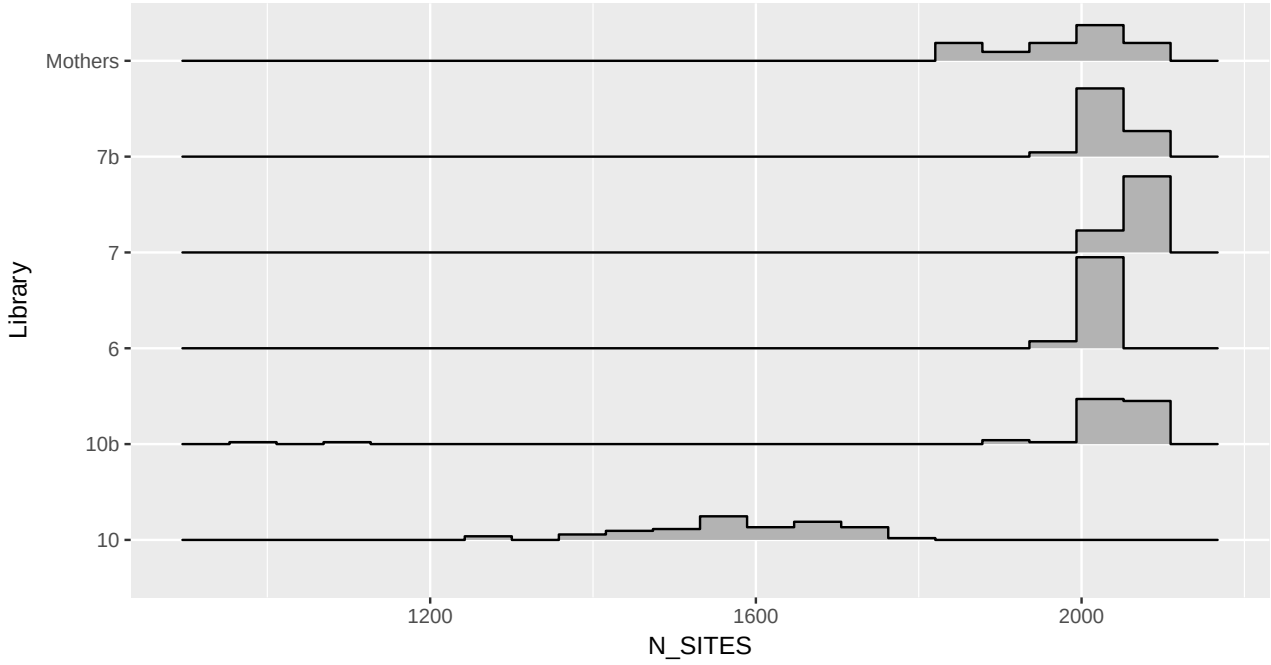

Figure 11.4: Distribution of the number of sites per individuals in each library including the new ones (10 and 10b).

11.2 CSD

After removing all samples from library 10, I performed the downstream analyses to find CSD again.

11.2.1 Grouping mothers

11.2.2 Finding centromeres

11.2.3 CSD scan

11.2.4 Association mapping

Chapter 12

TO DO

- Categorizing mothers: Use total number of males and infer number of diploid males from known proportion of haploids in each family.
- Centromere finding: look at percentage of heterozygous lociin the centromeric region of each individual separately $->$ isolate potential cases of terminal fusion automixis that could interfere with centromere localization.
- use Fisher exact test instead of Chi2 for association mapping: need to test for independence of rows/-columns.
- if n CSD loci on different chromosomes are het. in a mother, the proportion of males in diploid offspring will be $\frac{1}{2^n}$.
- Compute coverage along genome for all loci including monomorphic ones.
- Isolate regions of abnodmally high coverage, eventually exclude them. Watch out for overmerging of paralogs and repetitive sequences.
- Visualize average heterozygosity (or allele frequency) at each SNPs in offpring versus in mother to assess transmission bias (hom SNPs in mothers more often het in offspring or reversse).
- Validate haploid definition using haplotypes deduced from linkage map.
- convert VCF file into PED format using vcftools
- import data into genABEL and perform association mapping
- make snp calling less reliant on correct mother calls and exclude SNPs only if mother is hom/missing AND less than N offspring are heterozygous (small N)
- automate threshold selection for inferring ploidy