Workshop on Genomics 2020

Population genetics - Lab02: Selection scan

Selection

...or how different alleles at a locus may affect survival and reproduction

In this lab we will focus on positive selection, and in particular on how to find regions of a chromosome that could have undergone a (complete) **selective sweep**. Wait, do we all know what a selective sweep is?

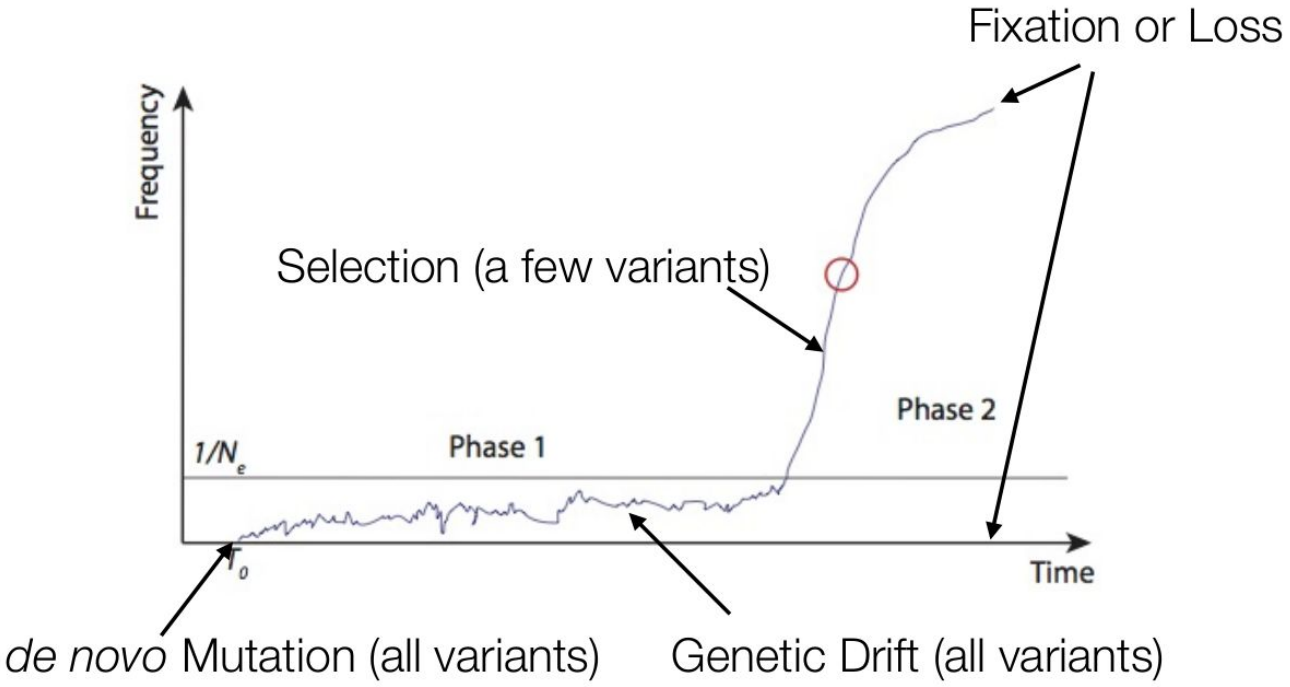
Selection

...or how different alleles at a locus may affect survival and reproduction

In this lab we will focus on positive selection, and in particular on how to find regions of a chromosome that could have undergone a (complete) **selective sweep**. Wait, do we all know what a selective sweep is? What about **genetic hitchhiking**?

Positive selection

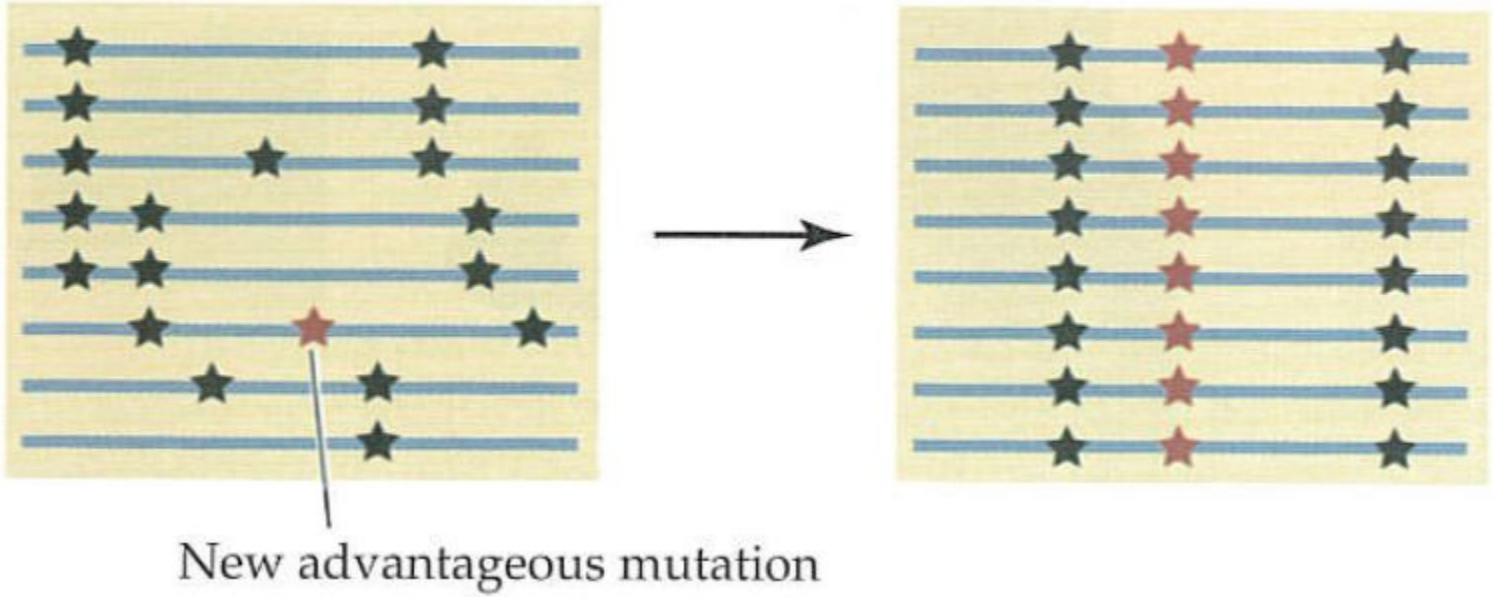
When an advantageous allele, after escaping loss by drift when at low frequency, rapidly reaches fixation in the population.



from Kelley Harris, Genome561_lecture1.pdf

Selective sweep

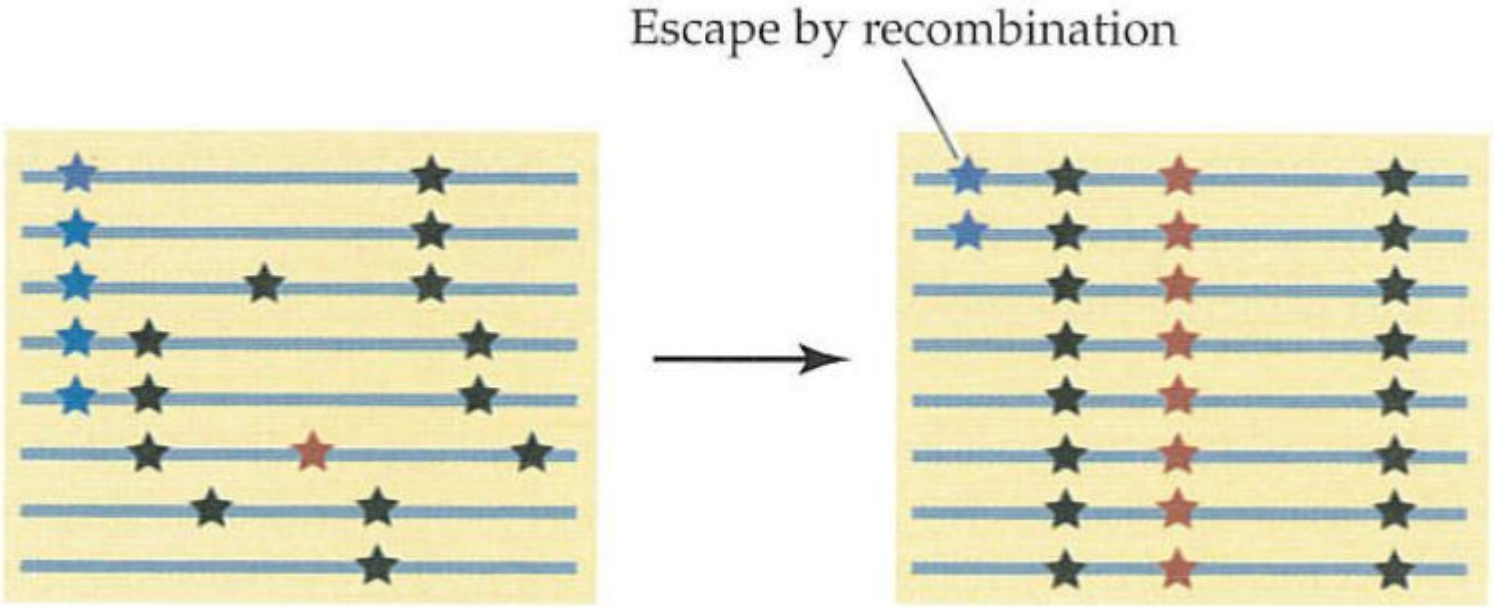
Complete (hard) sweep



from Nielsen and Slatkin 2005

Selective sweep

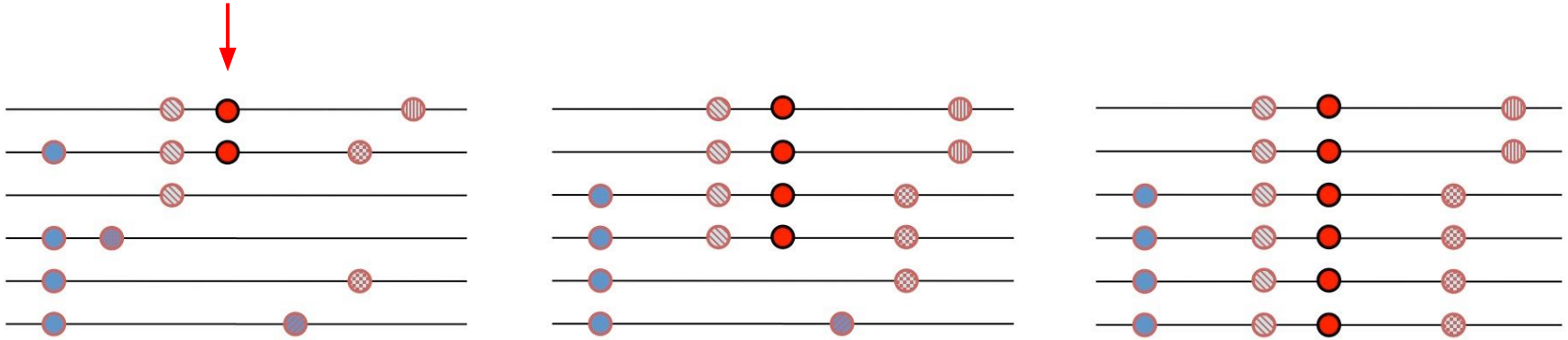
Complete (hard) sweep with recombination



from Nielsen and Slatkin 2005

Selective sweep

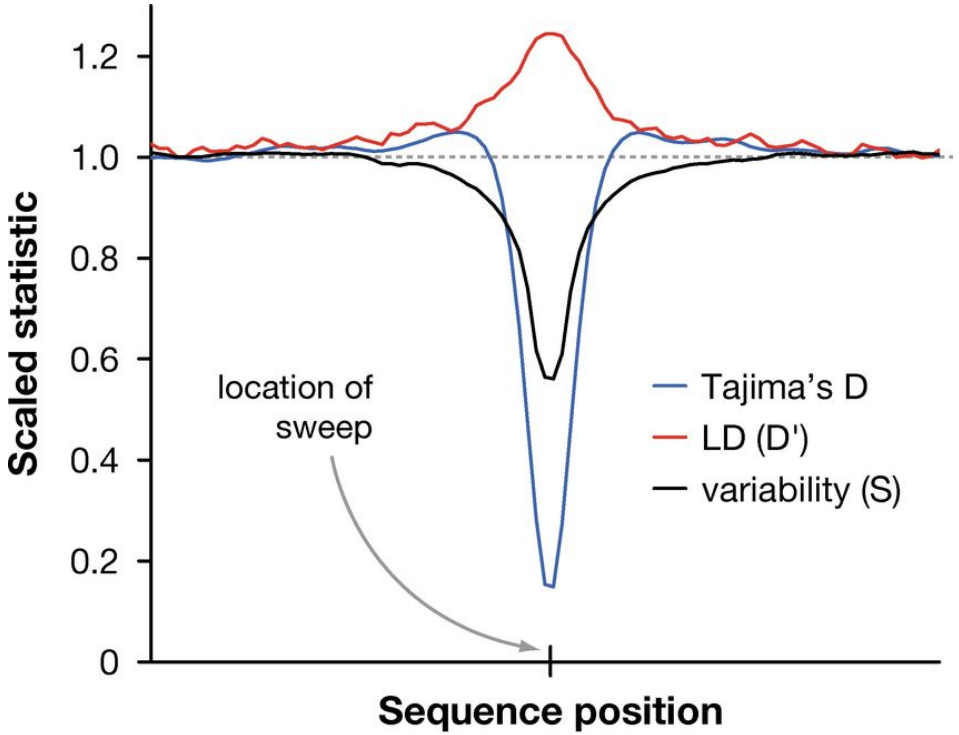
Soft sweep from standing variation



from Pleuni Pennings, wikimedia

Selective sweep

Effects on nearby neutral variation due to hitchhiking

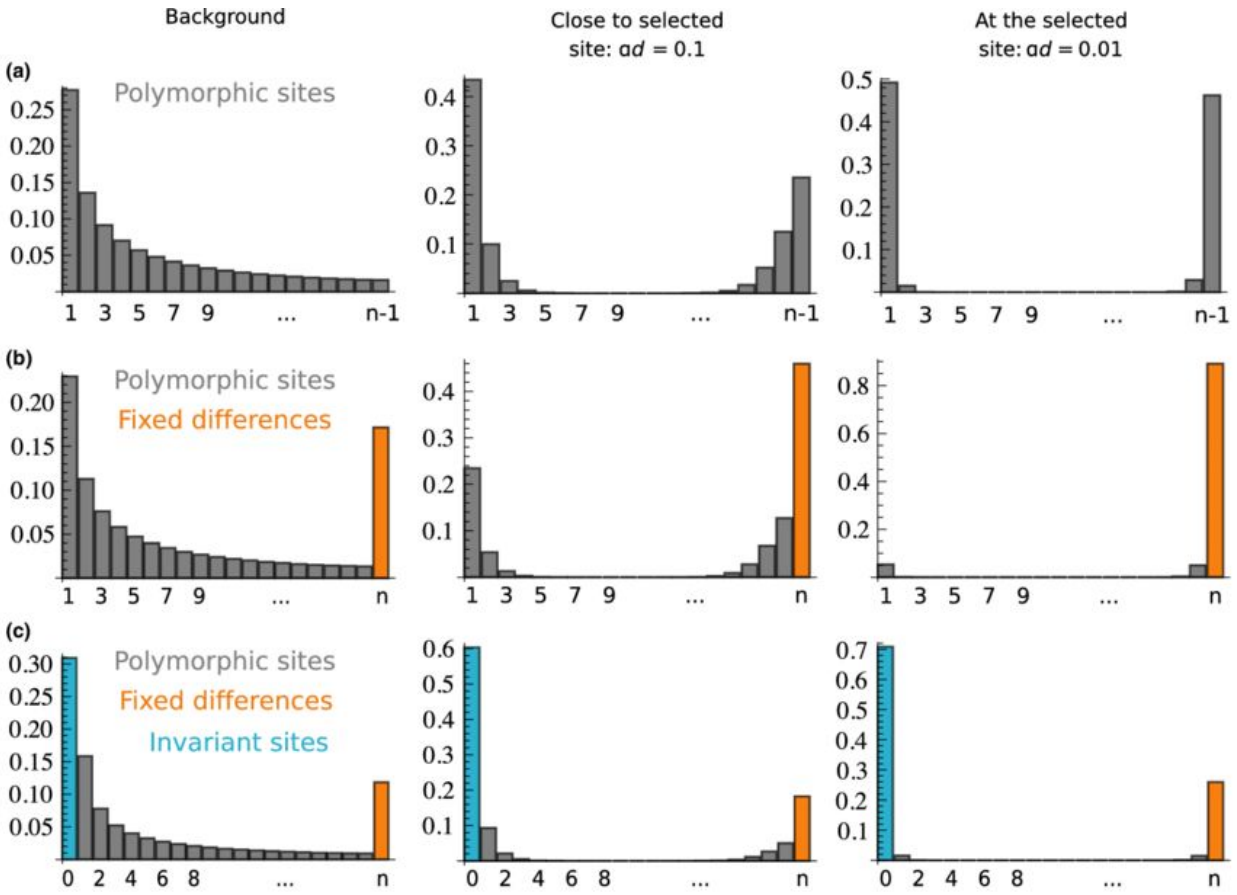


Nielsen R. 2005.
Annu. Rev. Genet. 39:197–218

Selective sweep

Effects on nearby neutral variation due to hitchhiking

Local from global SFS deviation can be detected using a Composite Likelihood Ratio (CLR) test in **SweepFinder2**



Huber et al 2016

Selection scan using SFS as summary stats

We are going to be using the SFS to detect signatures of positive selection

First, we are going to use the SFS generated from some simulated data using forward-simulations in SLiM3

Next, we are going to use the SFS generated from some empirical data (penguins!)

Selection scan using SFS as summary stats

0. Access your Amazon EC2 instance
Get your public domain address and login using Guacamole

1. Simulate a positively selected mutation using SLiM3 (https://messerlab.org/slim)

SLiM3 is a powerful evolution simulation framework that allows us to model realistic evolutionary scenarios and predict their expected patterns in genomic data. We're not going to dive into the specifics behind SLiM today, but you can read more about it in the link above (or ask us!)

Today, we will use it to simulate 1 Mb of data from a population of 2000 diploid individuals. We will introduce a mutation under positive selection, and use **Sweepfinder2** to detect the signature of this hard sweep in the simulated chromosome. We will then explore how recombination affects the signature of a hard sweep.

Shout out to Jim Whiting (Uni. Exeter) for assisting with this exercise!

Selection scan using SFS as summary stats

0. Access your Amazon EC2 instance
Get your public domain address and login using Guacamole

1. Simulate a positively selected mutation using SLiM3 (https://messerlab.org/slim)

SLiM3 is a powerful evolution simulation framework that allows us to model realistic evolutionary scenarios and predict their expected patterns in genomic data. We're not going to dive into all of the specifics behind SLiM today, but you can read more about it in the link above (or ask us!)

Today, we will use it to simulate 1 Mb of data from a population of 2000 diploid individuals. We will introduce a mutation under positive selection, and use **Sweepfinder2** to detect the signature of this hard sweep in the simulated chromosome. We will then explore how recombination affects the signature of a hard sweep.

Shout out to Jim Whiting (Uni. Exeter) for assisting with this exercise!

Selection scan using SFS as summary stats

```
// Keywords: conditional sweep

initialize() {

initializeTreeSeq(simplificationInterval=5000,checkCoalescence=T);

        defineConstant("theta",0.004);
        //defineConstant("rho",0.0008);
        defineConstant("popN",2000);
        //defineConstant("sweep_locus",300000);
        //defineConstant("time_post_sweep",10);
        //defineConstant("selection",0.5);

        initializeMutationRate(0);
        initializeMutationType("m1", 0.5, "f", 0.0);
        initializeMutationType("m2", 1.0, "f", selection);  // introduced mutation

        m1.convertToSubstitution = T;
        m2.convertToSubstitution = T;

        initializeGenomicElementType("g1", m1, 1.0);
        initializeGenomicElement(g1, 0, 999999);
        initializeRecombinationRate(rho/(4*popN));
}
1 {
        // save this run's identifier, used to save and restore
        defineConstant("simID", getSeed());

        sim.addSubpop("p1", popN);
}
```

Shout out to Jim Whiting (Uni. Exeter) for assisting with this exercise!

Selection scan using SFS as summary stats

Open a terminal and navigate to the slim3 directory:

```
cd ~/workshop_materials/pop_gen/sweepfinder2/slim3
```

Let's run slim3 to simulate our population

```
slim -d rho=0.0008 -d sweep_locus=300000 -d time_post_sweep=10 -d selection=0.5
./scripts/basic_sweep_trees.slim
```

Here, we use a rho of 0.0008, introducing a mutation under positive selection at 0.3Mb, sampling the population 10 generations post-sweep, at a selection coefficient of 0.5

Wait for the tree sequences to be simulated. Then convert the output trees to a vcf file:

```
python3 ./scripts/mutate_outputs.py ./outputs/positive_sweep.trees
./outputs/sweep.rho0.0008.vcf 0.0008
```

NB Tree sequencing is essentially a method of tracking the true local ancestry of every chromosome position in every individual as a SLiM model runs. Such ancestry information can be saved out to files called .trees files. See page 37 of the SLiM3 manual for more info

Shout out to Jim Whiting (Uni. Exeter) for assisting with this exercise!

Selection scan using SFS as summary stats

2. Prepare the input file for SweepFinder2 (http://www.personal.psu.edu/mxd60/sf2.html)

Sweepfinder requires a specific input file with SNPs as rows and four columns. Depending on what you want to do, pre-written scripts are available to convert vcf to sweepfinder2 format (we will use one later with the empirical data), but here we will use vcftools and some unix!

position	x	n	folded
43	0	48	0
57	0	48	0
58	0	48	0
88	46	46	0
99	0	46	0
101	0	46	0
118	0	44	0
135	21	44	0
156	0	46	0
157	0	46	0

Selection scan using SFS as summary stats

2. Prepare the input file for SweepFinder2 (http://www.personal.psu.edu/mxd60/sf2.html)

```
vcftools --vcf ./outputs/sweep.rho0.0008.vcf --counts2 --out ./outputs/SF_rho0.0008_tmp

tail -n+2 ./outputs/SF_rho0.0008_tmp.frq.count | awk -v OFS="\t" '{print $2,$6,$4,"1"}' > ./outputs/SF_rho0.0008.in

echo -e 'position\tx\tn\tfolded' | cat - ./outputs/SF_rho0.0008.in > temp && mv temp ./outputs/SF_rho0.0008.in
```

If the prompt asks if you want to overwrite, say yes! We now have a file formatted ready for use in Sweepfinder2

Selection scan using SFS as summary stats

3. Run SweepFinder2 (http://www.personal.psu.edu/mxd60/sf2.html)

SweepFinder2 -sg 5000 ./outputs/SF_rho0.0008.in ./outputs/Sweepfinder_rho0.0008.out

And plot the result in R:
Rscript ./scripts/plot_rho0.0008.R

Does the mutation introduced at 0.3 Mb show a signature of selection??

Ok, now we're going to change rho (the population-scaled recombination rate) and see how this affects our ability to detect a hard sweep

Selection scan using SFS as summary stats

4. Let's run slim3 to simulate our population with a higher recombination rate of 0.008 (rho)

```
slim -d rho=0.008 -d sweep_locus=300000 -d time_post_sweep=10 -d selection=0.5
./scripts/basic_sweep_trees.slim
```

Convert the output trees to a vcf file:

```
python3 ./scripts/mutate_outputs.py ./outputs/positive_sweep.trees
./outputs/sweep.rho0.008.vcf 0.008
```

Convert to Sweepfinder2 format:

```
vcftools --vcf ./outputs/sweep.rho0.008.vcf --counts2 --out ./outputs/SF_rho0.008_tmp
```

```
tail -n+2 ./outputs/SF_rho0.008_tmp.frq.count | awk -v OFS="\t" '{print $2,$6,$4,"1"}' >
./outputs/SF_rho0.008.in
```

```
echo -e 'position\tx\tn\tfolded' | cat - ./outputs/SF_rho0.008.in > temp && mv temp
./outputs/SF_rho0.008.in
```

Selection scan using SFS as summary stats

4. Let's run slim3 to simulate our population with a higher recombination rate of 0.008 (rho)

Run sweepfinder2:
SweepFinder2 -sg 5000 ./outputs/SF_rho0.008.in ./outputs/Sweepfinder_rho0.008.out

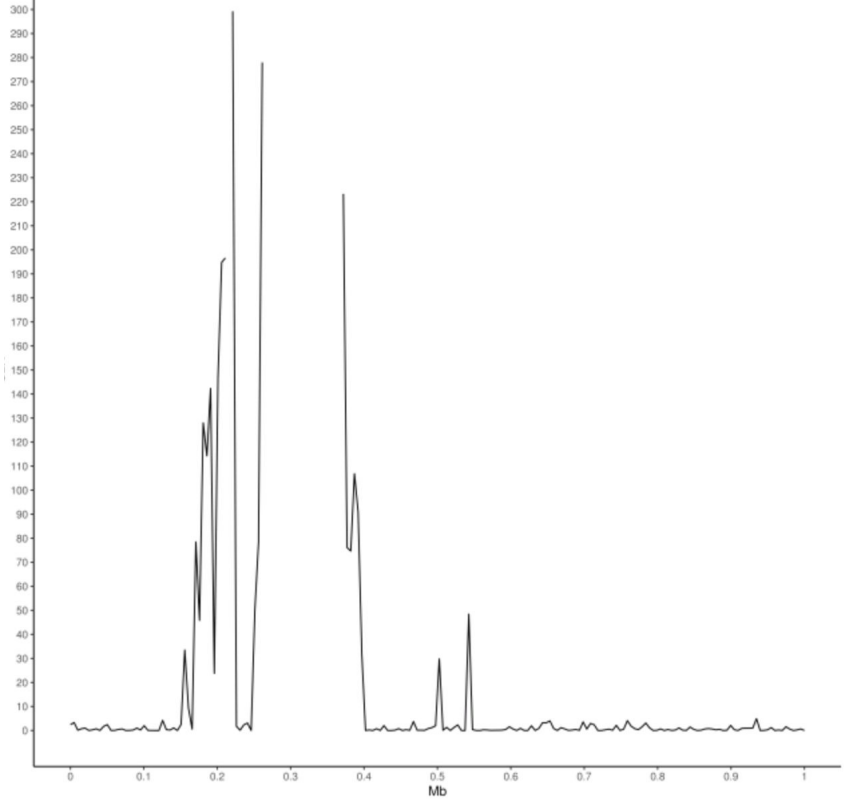
And plot the result in R:
Rscript ./scripts/plot_rho0.008.R

Compare this plot with the previous one using a rho of 0.0008. How does the signature of the hard sweep differ with a higher recombination rate? Does it show what you expect given how recombination affects hitchhiking?

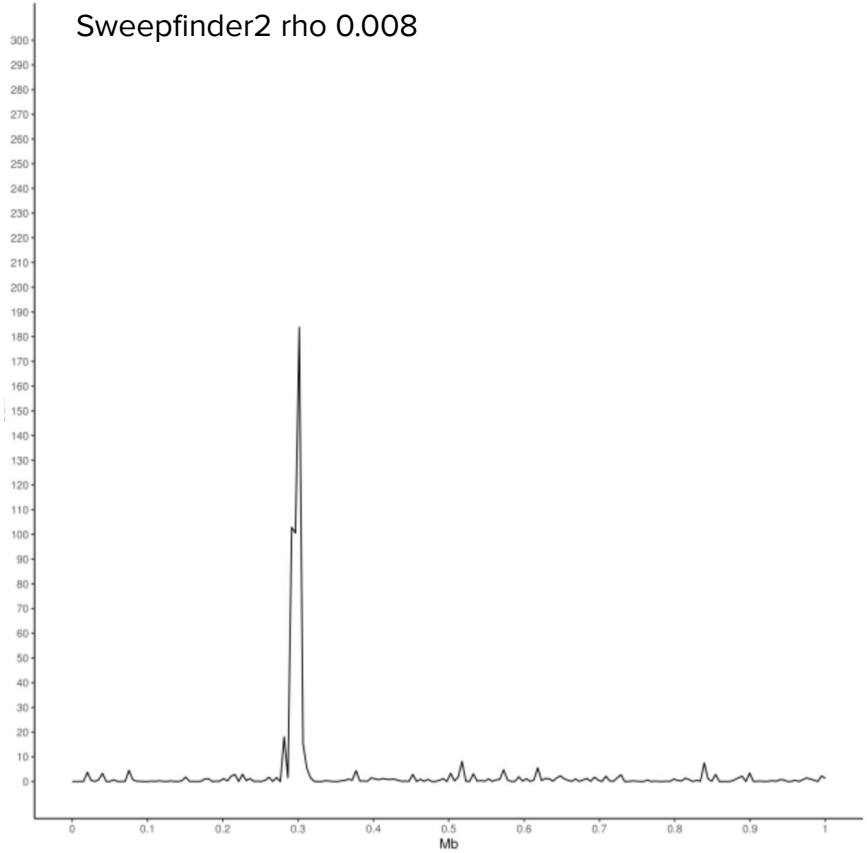
If you wanted, you could edit other parameters in a SLiM simulation, for example theta, effective population size and the selection coefficient. You could then run Sweepfinder2 to see how these affect the ability to detect a hard sweep, but for now, we will move on to detecting sweeps in some empirical data.

Selection scan using SFS as summary stats

Sweepfinder2 rho 0.0008



Sweepfinder2 rho 0.008



Selection scan using SFS as summary stats

5. Ok, enough of simulating data!! Let's try detecting sweeps in some empirical data
Move into the calc_sfs directory:
cd ~/workshop_materials/pop_gen/calc_sfs

Run the script vcf2SW2.py using the Emperor as ingroup 1 (-p1 emp), the King and input 2 (-p2 king) and the other species as outgroup (-p3 pop_out) to convert the three vcf files in this folder (exclude NW_008794753.1.filt.biall.recode.vcf)

python3 vcf2SW2.py -f NW_008794115.1.filt.biall.recode.vcf -p1 emp -p2 king -p3 pop_out

The script outputs a new file with the same stem name and the extension .emp.sw

Run the script again for the three vcf files but this time use the King as ingroup 1 and the Emperor as ingroup 2.

Move all six .sw files to the folder *sweepfinder2/penguins* in *pop_gen*

mv *.sw /home/genomics/workshop_materials/pop_gen/sweepfinder2/penguins
and change to this directory for the following steps

Selection scan using SFS as summary stats

6.1. (Prepare the global SFS)

The model implemented in *Sweepfinder2* looks for deviations in local SFS from the global one. This means we need to provide an SFS estimated on the whole genome or produced by simulations if we are really sure about the demographic history of the investigated population. In this exercise, we will use the whole-genome SFS as background for the Composite Likelihood Ratio test.
To make the global SFS, we have
i) converted all vcf files of all scaffolds into *Sweepfinder2* input files
ii) concatenated them
iii) used the concatenated file as input in this *Sweepfinder2* command:

SweepFinder2 –f concat.emp.sw wholegenome.emp.sw.spect

The background SFS (one per species) are already in the penguins folder. Can you see them?

Selection scan using SFS as summary stats

6.2. Run SweepFinder2

Now run SweepFinder2 using 5 kb windows to estimate the local SFS (-lg 5000) on the six files we have just prepared (three scaffolds per species). Below, is an example of the command line with one scaffold .sw file as input, the whole-genome SFS for the Emperor penguin and a .sf2 file as output

SweepFinder2 -lg 5000 NW_008794115.1.emp.sw wholegenome.emp.sw.spect NW_008794115.1.emp.sf2

NB the NW_008795129.1.filt.biall.recode.vcf is a large scaffold! So run it with a 50 kb window size (this should take about 15 minutes)

Selection scan using SFS as summary stats

6.3. Check outputs drawing the CLR distribution along the scaffolds
Open the output files we just created (.sf2). What are the different columns?
Can you see the Composite Likelihood Ratio for each windows?
For each file, plot the CLR along the genome by editing and using the Rscript plot_sf2.R. Are there any signatures of selective sweeps in any of the scaffolds for either of the species?

EXTRA. Check the annotation file for genes in regions of high CLR
One of the first things you want to do when finding signature of selection in some genomic regions is to see if a gene is annotated in that region. Well, you should first be really sure about your results and do additional testing but curiosity usually wins.

A gene annotation file for the Emperor penguin reference genome (extension .gff) is available in the folder. Using a simple bash command, it is possible to see the list of genes annotated in a scaffold and scroll to the region of interest

```
grep NW_008794115 GCF_000699145.1_ASM69914v1_genomic.gff | grep -e "\tmRNA\t" | less -S
```

Selection scan using SFS as summary stats

A bit more to say about SweepFinder

1. More powerful if you **simulate** the background **SFS** according to the (correct) demographic model

2. It is better to use a **recombination map**

3. It is also useful to make the analysis in the context of **background selection**

4. At the very least, you should change and test the effect of the **window size** for the grid