Haploscope Documentation

F. A. San Lucas*, N. A. Rosenberg, P. Scheet*
Haploscope version 0.4.1
January 4, 2012
http://scheet.org/software

*If you have questions, or if you would like to request a feature or report a bug, please contact:
Anthony San Lucas at sanlucas@gmail.com or Paul Scheet at pscheet@alum.wustl.edu.

Contents

Haploscope Description

Haploscope is a software package that facilitates flexible rendering of images to aid interpretation of model-based summaries of population haplotypes. Haploscope is designed to accept haplotype frequency input directly from output files of fastPHASE (Scheet & Stephens, 2006), though output from other cluster-based models for population haplotypes could be adapted for input to Haploscope. To see an example of how to run fastPHASE to obtain Haploscope input, see Appendix A.

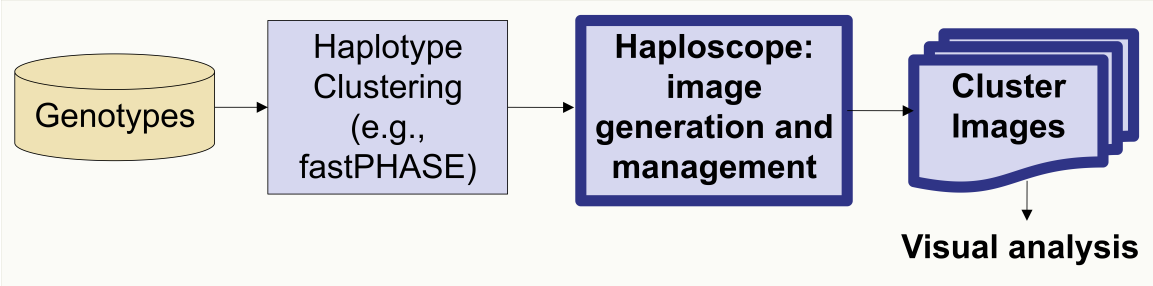


Figure 1. Haploscope accepts haplotype clustering information provided by a clustering program and generates images that make it possible to perform high-level visual interpretation of the cluster data.

Haploscope generates three types of images providing different visualizations of population haplotypes:

1. **Population haplotype cluster plots:** "consensus haplotypes" frequencies within individual populations
2. **Cluster allele heat maps:** model details, e.g., cluster-specific allele frequencies
3. **Summary haplotype plots:** haplotype cluster frequencies composed of multiple subpopulations in a single plot

Haploscope can be run interactively or through the command-line. It is a Java program and can be installed on most operating systems. Sample input files are provided with the Haploscope distribution and this tutorial shows how to generate images from these input files.

Installation

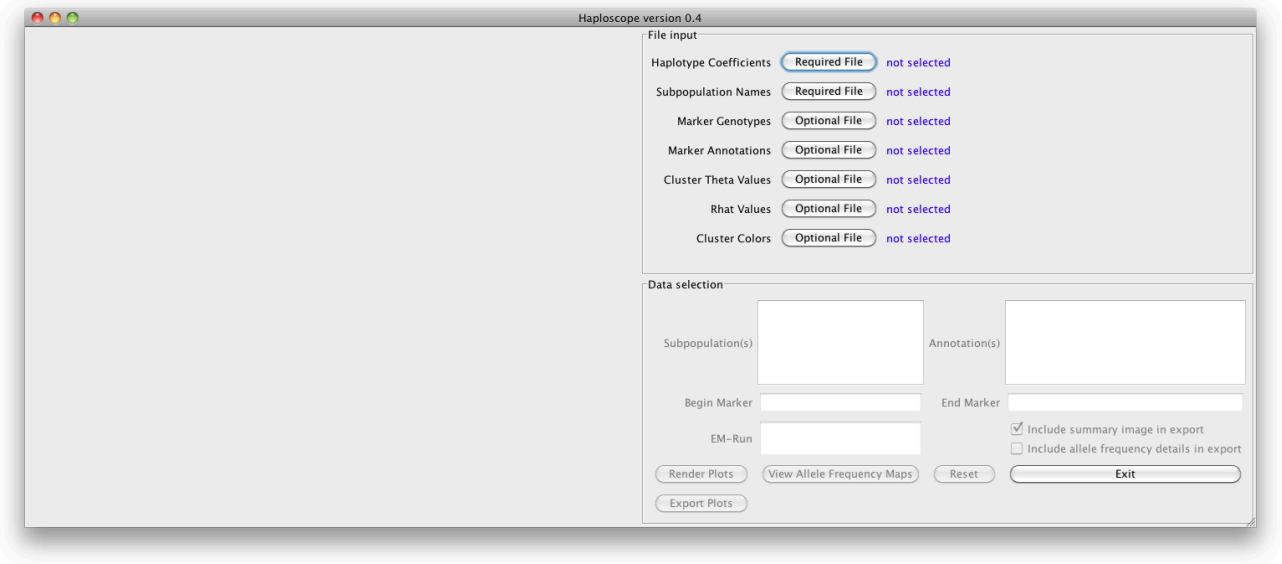
1. Requirements: Java 1.5 or higher. Most operating systems already have this installed. One way to verify this is with the following command:
   `java -version`
2. Download `haploscope-bin.tar.gz` from https://cge.mdanderson.org/~pascheet/software.html
3. Extract the compressed file: `tar -xzf haploscope-bin.tar.gz`
4. The location of the extracted `haploscope` directory will be referenced with `$HAPLOSCOPE_HOME` in the remainder of the tutorial.

GUI Interface

To launch the GUI interface, run (from `$HAPLOSCOPE_HOME`):

```
1. cd bin
2. ./haploscope.sh --interactive=true
```

There are two main input panels on the right side of the GUI application: a file input panel and a data selection panel. The left side of the GUI is reserved for the cluster plots.



Below is a description of the Haploscope input files. Examples of these files are packaged with the Haploscope distribution and can be found in `$HAPLOSCOPE_HOME/examples/LCT`. And a description of the file formats can be found in the "Input file format descriptions" section of this tutorial.

There are 2 files that are required for generating haplotype cluster plots. The rest of the files are optional and are used for additional markup of the plots and for viewing details of the haplotype clusters.

1. *Haplotype Coefficients* (required file). This file contains all of the population cluster allele frequencies for all markers analyzed by fastPHASE. This file is generated by fastPHASE as `prefix_pop_cluster_freqs`, where the "prefix" may be supplied by the fastPHASE user.
2. *Subpopulation Names* (required file). This file needs to be generated by the user. It maps subpopulation IDs (indicated in `prefix_pop_cluster_freqs`) to names to be used by Haploscope.
3. *Marker Annotations* (optional). This file is generated by the user and provides a mechanism for marking and labeling markers of interest onto the haplotype cluster plots.
4. *Cluster Theta Values* (optional – but required if you want to generate an allele heatmap). This file provides allele frequencies within each cluster. This data can be displayed in grayscale on a heatmap for cluster allele interpretation. This file is named `prefix_thetahat.txt` by fastPHASE.
5. *Marker Genotypes* (optional, but required for viewing the allele heatmap). This file is named `prefix_origchars` by fastPHASE and this file indicates the nucleotides for allele 1 and allele 2 at each marker. These nucleotides are displayed on the allele heatmap for interpretation of cluster alleles.
6. *r-hat* (optional). Jump probabilities for the HMM (akin to recombination parameters) can be displayed under the haplotype cluster plots to give a visual indication of genomic areas for where there may have been haplotype switching between consecutive markers (or indirectly this could be thought of as possible recombination spots between markers). This file is named `prefix_rhat.txt` by fastPHASE.

Data Selection

1. *Subpopulation(s)* – the user must select at least one subpopulation to display.
2. *Annotation(s)* – if a marker annotation file was specified, there will be a list of possible annotations that the user can select for markup onto the cluster plots.
3. *Begin/End Markers* – the user can zoom in on an area of interest on the plot by specifying the begin and end markers to display on the cluster plots. The markers are numbered from 1 to n, where n markers were originally input into fastPHASE.
4. *EM-Start (e.g., EM-Run)* – normally the user can leave this at its default. If multiple parameter solutions/ EM runs were produced by fastPHASE (by default there are multiple runs of the EM; modifiable with `-T<number>`), the user will need to specify which start number to use.
5. *Print Summary Image* – if this is checked, a "Summary Haplotypes" image will be exported (in addition to individual population cluster plots) whenever the "Export Plots" button is pressed.
6. *Render Plots* – this will invoke the rendering of individual population haplotype cluster plots onto the Haploscope GUI.
7. *Export Plots* – this will export the plots on the Haploscope GUI to the user's file system. If "Print Summary Image" was checked, a "Summary Haplotypes" image will also be exported.
8. *View Allele Frequency Maps* – this will pop up a heatmap detailing cluster and population allele frequencies. This image currently is not exportable, but the user can use a "print screen" utility if this is desired.

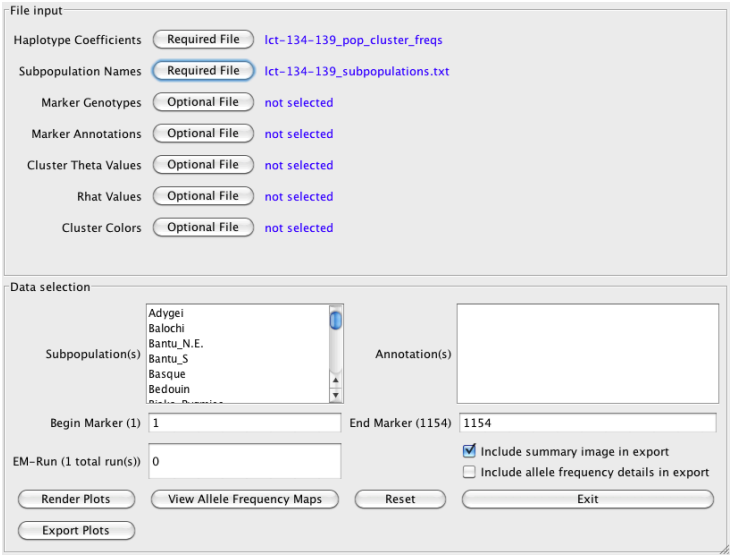
Interactive Tutorial: An example using genotypes surrounding the LCT region

The input data for this tutorial is provided with the distribution, and this tutorial details step-by-step instructions for how to use Haploscope to visualize haplotype clustering inferred from an analysis of the LCT region across multiple populations.

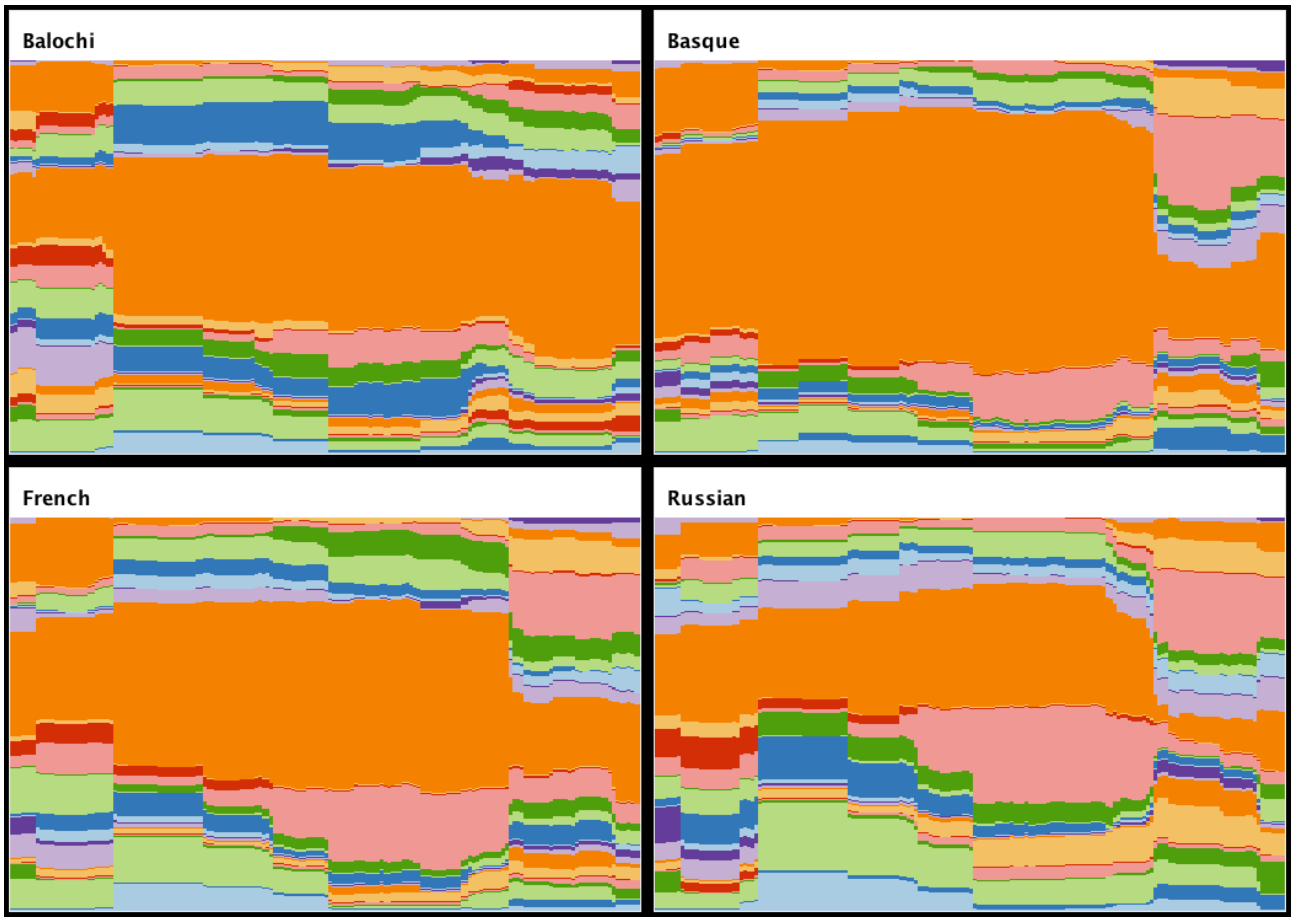
Interactive steps required to generate plots

Sample data files are provided in `$HAPLOSCOPE_HOME/examples`:

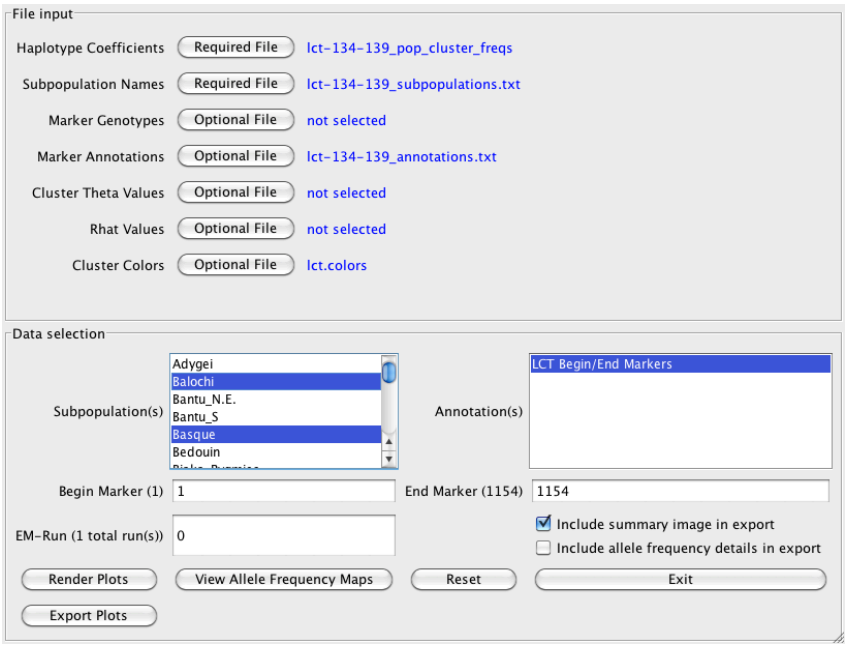
1. Select a haplotype coefficients file (`$HAPLOSCOPE_HOME/examples/LCT/lct-134-139_pop_cluster_freqs`).
2. Select a sub-population names file (`$HAPLOSCOPE_HOME/examples/LCT/lct-134-139_subpopulations.txt`).



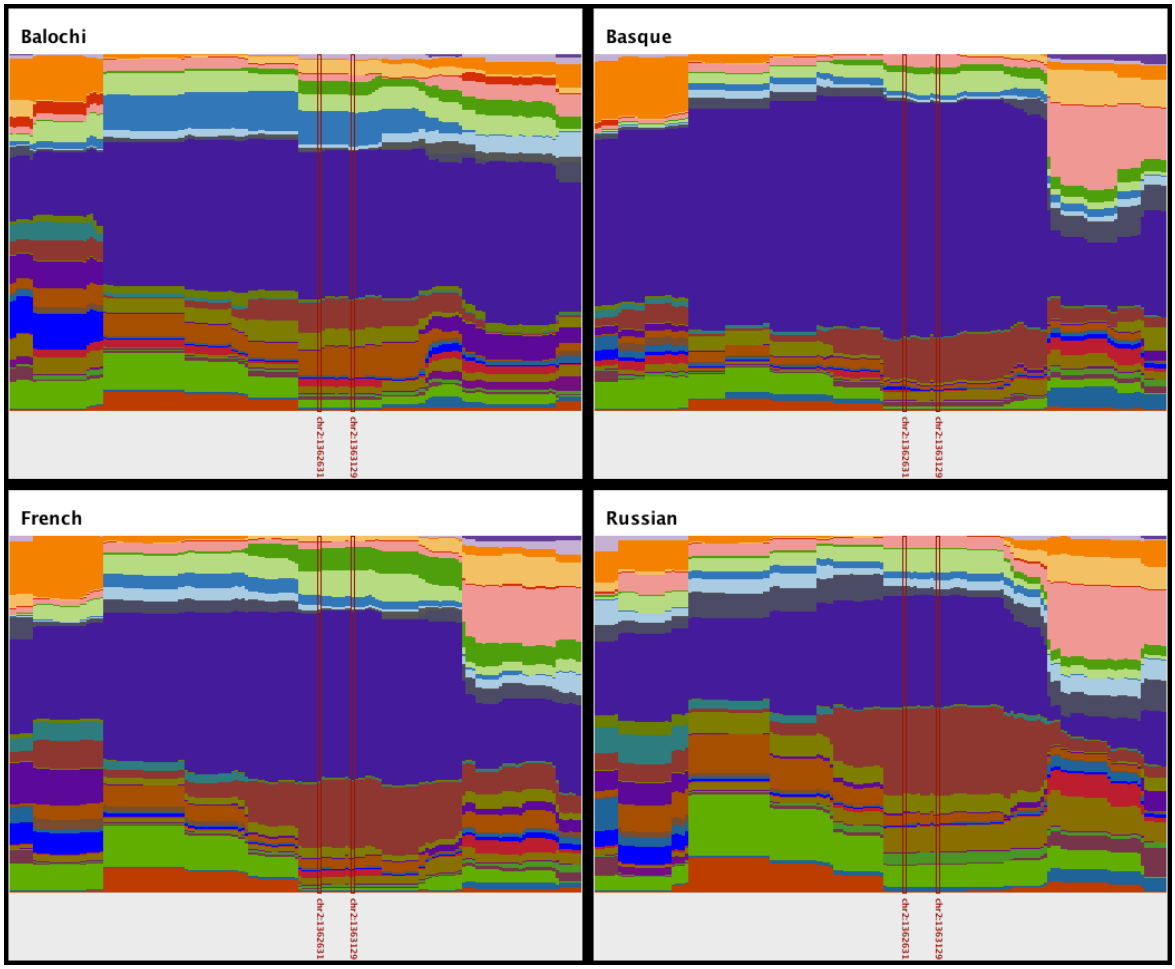
3. After providing these two required files, Haploscope will fill in the `Data Selection` panel with some default values. Select a few subpopulations by clicking one or more subpopulation names from the `Subpopulation(s)` list. Hold down the ctrl key to select more than one subpopulation. To deselect a subpopulation, just select it again. For this tutorial, select the following subpopulations:
    a. `Balochi`
    b. `French`
    c. `French_Basque`
    d. `Russian`
4. There are 1154 markers that could be displayed. For `Begin Marker` enter in `430` and for `End Marker` enter `600`. This will tell Haploscope to generate images for the region defined by markers 430 to 600 only.
5. Press `Render Plots`. The following plots will be generated in the GUI. Each plot shows cluster haplotype frequencies for an individual subpopulation.



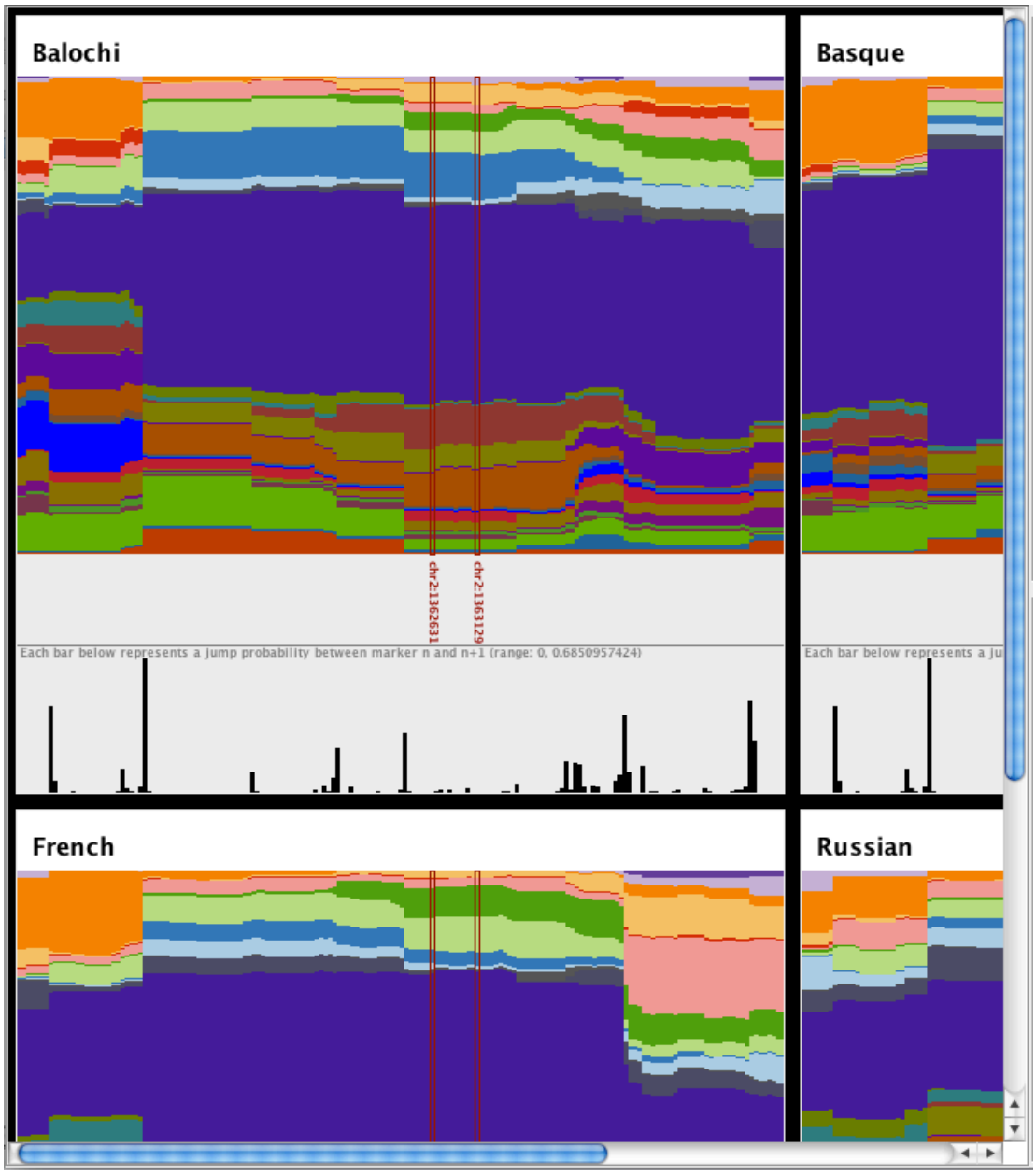
6. You can specify annotations to be displayed on the plots: specify a marker annotations file (`$HAPLOSCOPE_HOME/examples/LCT/lct-134-139_annotations.txt`).
7. Select an annotation group. In this tutorial example, only one annotation group will be listed: `LCT Begin/End Markers`.
8. Select a color file to change the cluster colors (`$HAPLOSCOPE_HOME/examples/LCT/lct.colors`).



9. Press `Render Plots` to update the plot colors and to see the annotations.

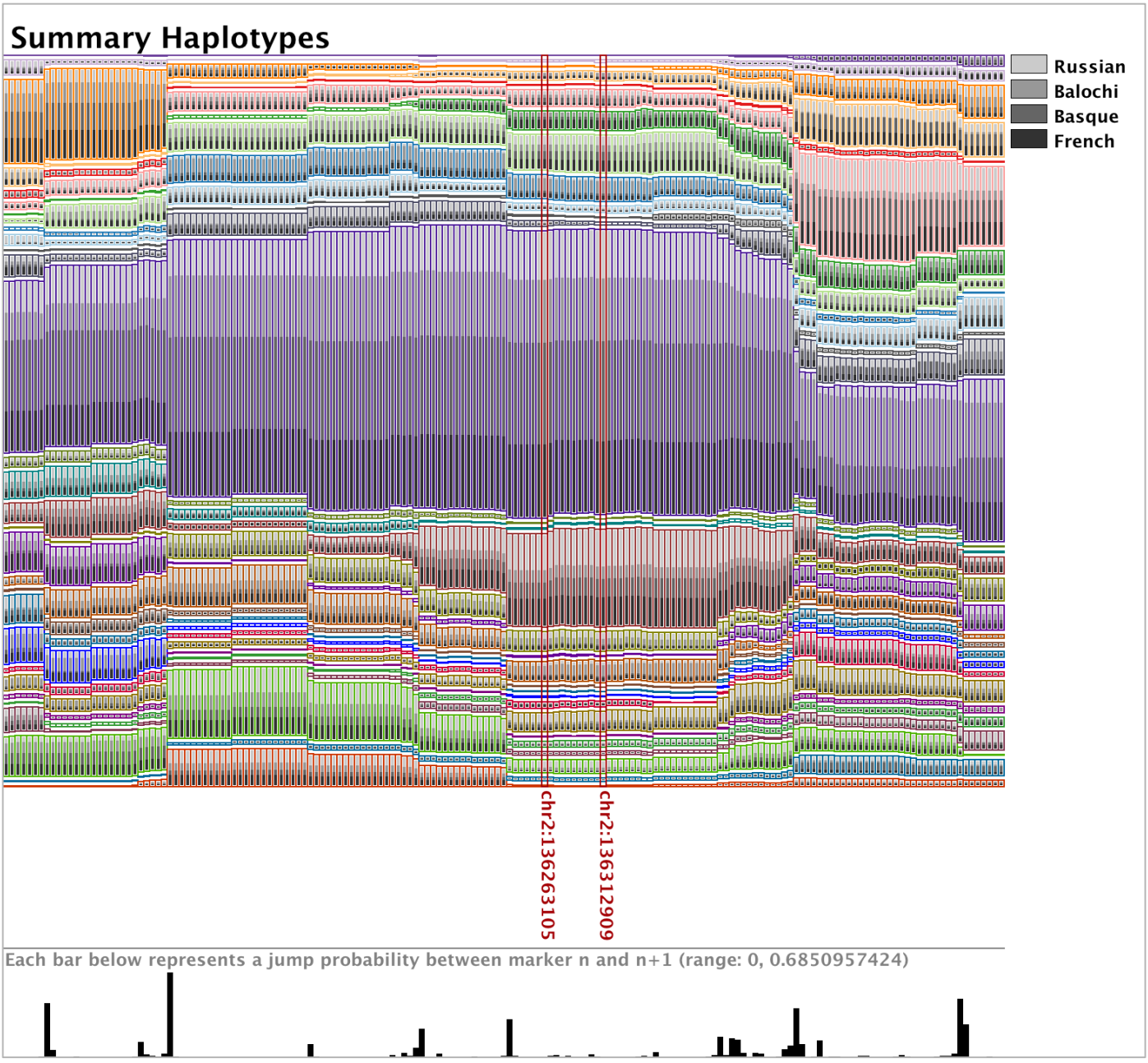


10. Estimated jump probabilities ("r-hat" values) can be displayed under the cluster plots by specifying an rhat file (output from fastPHASE). Select `$HAPLOSCOPE_HOME/examples/LCT/lct-134-139_rhat.txt`.
11. Press `Render Plots` to update the plots with rhat values. The rhat values give an indication of the probability of cluster switching (indicative of historical recombination) between two consecutive markers.

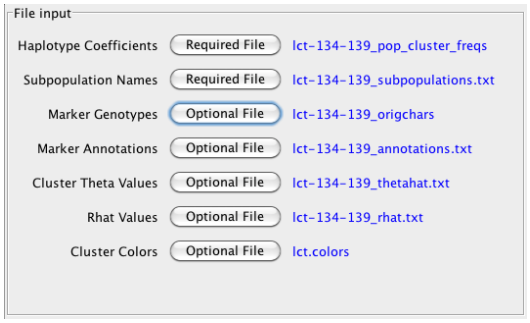


12. Click or unclick `print summary image` (if checked, this will create a summary image when the plots are exported).
13. Press `Export Plots` (to export the rendered plots and a summary image). High-quality postscript files will be generated in `$HAPLOSCOPE_HOME/images`. The summary image illustrates the subpopulation differences in each cluster. The frequencies for each cluster at each marker are averaged across selected subpopulations (where each subpopulation is

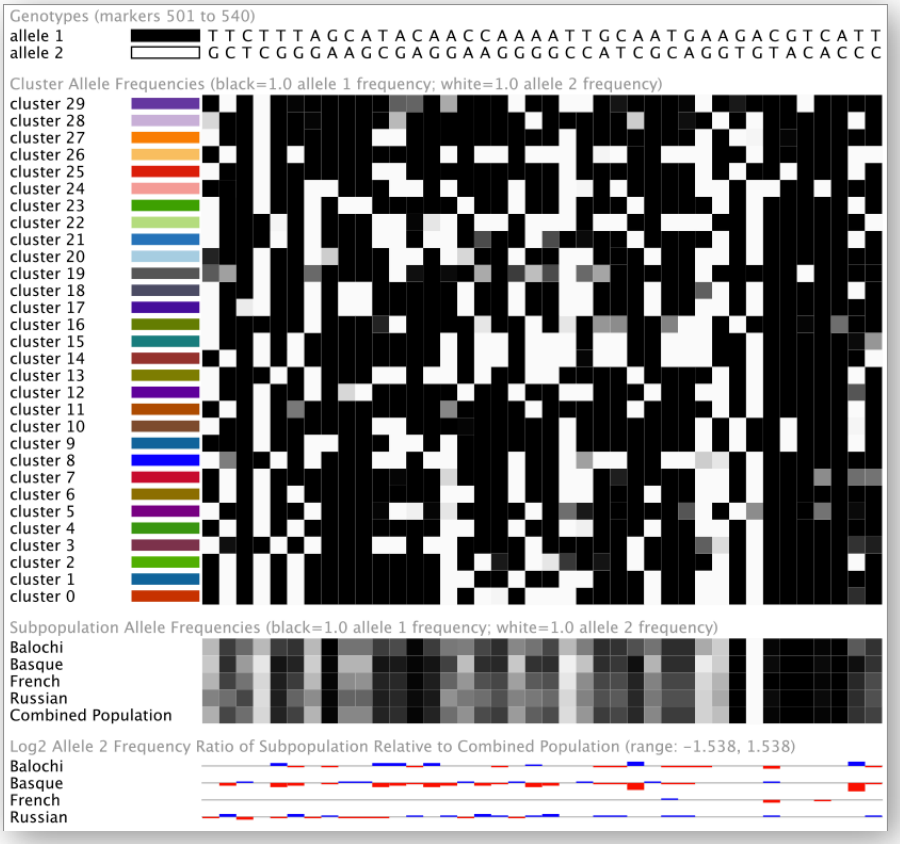
equally weighted). The contribution that each subpopulation makes to this average is depicted with different shades of gray. Images providing similar information can be found in Browning and Weir (2010).



14. Select a cluster theta (i.e., cluster allele frequency) file (`$HAPLOSCOPE_HOME/examples/LCT/lct-134-139_thetahat.txt`).
15. Select a genotype file (`$HAPLOSCOPE_HOME/examples/LCT/lct-134-139_origchars`).



16. To limit the genomic range a bit more (i.e., to zoom in) set `Begin Marker` to `501` and `End Marker` to `540`. These 40 markers contain the LCT SNPs.
17. Press `View Allele Frequency Maps` to view a window that illustrates these allele frequencies in a heatmap. The allele genotypes and the allele frequencies are shown for every marker in each cluster using a grayscale color. Black indicates a 100% allele 1 frequency, and white indicates a 100% allele 2 frequency. In addition summary allele frequencies for populations are shown below the cluster data, and relative frequency graphs are shown at the bottom. These relative frequency graphs relate the allele 2 frequencies of each population vs the combined (total) population. As a note, Haploscope will likely have trouble trying to build this heatmap if you try to view more than 1000 markers at once. To export this image, check the `Include allele frequency details in export` checkbox on the application form. Then when `Export Plots` is clicked, this image will be generated along with the other Haploscope plots.



Command-Line Tutorial: An example using genotypes surrounding the LCT region

If a user wants to automate the generation of haplotype cluster images, this can be done through the command line. Within the `$HAPLOSCOPE_HOME/bin` directory there are 3 example scripts for generating images from the command line. Below, we show 2 of them.

Generating simple haplotype cluster plots

This command will generate a simple haplotype cluster plot for each subpopulation specified through the `subpopulations` flag. The user should provide a comma-separated list of subpopulation IDs that correspond to the IDs in the `names.file`.

```
./haploscope.sh --interactive=false \
--input.file=../examples/LCT/lct-134-139_pop_cluster_freqs  \
--names.file=../examples/LCT/lct-134-139_subpopulations.txt  \
--subpopulations=2,14,15,39 \
--output.file.prefix=ex1 \
--begin.marker=400 \
--end.marker=650 \
--image.legend.show=false \
--print.title=false
```

Generating fully annotated images

This command will generate the same images but they will be annotated with additional information, and this script specifies additional input files that allow the creation of the allele-frequency-details image.

```
./haploscope.sh --interactive=false \
--input.file=../examples/LCT/lct-134-139_pop_cluster_freqs  \
--names.file=../examples/LCT/lct-134-139_subpopulations.txt  \
--rhat.file=../examples/LCT/lct-134-139_rhat.txt \
--thetahat.file=../examples/LCT/lct-134-139_thetahat.txt \
--orig.char.file=../examples/LCT/lct-134-139_origchars \
--color.file=../examples/LCT/lct.colors \
--marker.annotation.file=../examples/LCT/lct-134-139_annotations.txt \
--subpopulations=2,14,15,39 \
--output.file.prefix=ex2 \
--em.run=0 \
--begin.marker=400 \
--end.marker=650 \
--details.begin.marker=501 \
--details.end.marker=540 \
--print.allele.frequency.details.image=true
```

This example and another more detailed example command that generates annotated images can be found in the `$HAPLOSCOPE_HOME/bin` directory. By default, postscript generated files are stored in the `$HAPLOSCOPE_HOME/images` directory.

Input file descriptions

This section will describe the format of the input files.

Population cluster frequencies file

(Example: `$HAPLOSCOPE_HOME/examples/LCT/lct-134-139_pop_cluster_freqs`)

This example file was output by fastPHASE. So if you used fastPHASE to generate this file, you can just use it. If you are using some other clustering model, you can alter your file to include the following:

- (Line 1) number of markers
- (Line 2) subpopulation IDs (space separated)
- (Lines 3 through end of file) marker cluster frequencies:
    - Each line represents the cluster frequencies for an individual marker for an individual population.
    - There are n rows of frequencies for each subpopulation (where n is the number of markers).
    - There are m sets of these rows (where m is the number of subpopulations) where each set corresponds to the subpopulation IDs in the corresponding order listed on line 2.

r-hat file

(Example: `$HAPLOSCOPE_HOME/examples/LCT/lct-134-139_rhat.txt`)
This file specifies a single r-hat value for each marker (for each start of fastPHASE).

```
> start 0
1.0000000000
0.0003450966
0.0001149218
0.0002132104
0.0000161409
0.0000038008
0.0214172685
0.1167785842
...
```

Cluster theta (allele frequencies) file

(Example: `$HAPLOSCOPE_HOME/examples/LCT/lct-134-139_thetahat.txt`)
This file specifies cluster allele frequencies for each marker (for each start of fastPHASE). Each row specifies the cluster allele frequencies for an individual marker. So the number of data rows equals the number of markers analyzed. The number of columns is the number of clusters assumed in the model (e.g., from `-K` in fastPHASE)

```
> start 0
0.001000  0.001000  0.001085  0.002345  0.376112  0.001000  0.002698
0.001000  0.001000  0.001000  0.001000  0.001000  0.001000  0.001000
0.021508  0.001000  0.001000  0.001000  0.001000  0.001000  0.890638
0.001000  0.001000  0.001000  0.001000  0.001000  0.001000  0.001000
0.188586  0.350345
0.011827  0.001000  0.934022  0.999000  0.561355  0.223648  0.932174
0.001000  0.395925  0.096991  0.999000  0.001000  0.454967  0.111182
0.999000  0.001000  0.006892  0.033075  0.001000  0.008131  0.999000
0.001000  0.001000  0.001000  0.001000  0.213054  0.207405  0.001000
0.836858  0.999000
```

```
...
```

Subpopulation names file

(Example: `$HAPLOSCOPE_HOME/examples/LCT/lct-134-139_annotations.txt`)

This file maps the subpopulation IDs from Line 2 of the cluster frequencies file, to names displayed by Haploscope. This is a tab-separated list:

```
1	Adygei
2	Balochi
3	Bantu_N.E.
4	Bantu_S
5	Bedouin
6	Biaka_Pygmies
7	Brahui
8	Burusho
...
```

Cluster color file

(Example: `$HAPLOSCOPE_HOME/examples/LCT/lct.colors`)

This file specifies the RGB hexadecimal color to use for each cluster. Clusters in Haploscope are drawn from bottom to top – so cluster 1 is at the bottom of the haplotype cluster plots and cluster n at the top. If a color file is not specified, a default coloring scheme is used – this scheme can be found (and edited) at `$HAPLOSCOPE_HOME/resources/colors/color-brewer-10.colors`

```
1	cc3300
2	006698
3	4cb200
4	7f334c
5	329932
6	7f007f
7	8d7100
8	cc0033
...
```

Marker annotation file

(Example: `$HAPLOSCOPE_HOME/examples/LCT/lct-134-139_pop_cluster_freqs`)

Below are two annotation groups specified through the annotation file. The title of the first annotation group is `LCT Begin/End Markers` and the color to be used for the corresponding annotations is `#A40004` (an RGB hexadecimal number). There are two markers annotated in this group: markers `500` and `504`. Remember that markers are numbered from 1 to n where n is the number of markers input into Haploscope. The labels for these markers (tab separated from the marker number) are `LCT Begin` and `LCT End` respectively. A second group of `Random Additional Markers` is also shown. If these annotations are input into Haploscope, you could display none, one or both of these annotation groups at any given time by selecting (or deselecting the annotation groups) and re-rendering the plots.

```
> LCT Begin/End Markers; A40004
500     LCT Begin
504     LCT End
> Random Additional Markers; 00AA00
434     Random Marker 1
612     Random Marker 2
```

APPENDIX A: Running fastPHASE to obtain Haploscope input data files

Haploscope is designed to accept input that is generated by fastPHASE. This is the simplest way to use Haploscope. For instructions on downloading and installing fastPHASE, see `http://depts.washington.edu/uwc4c/express-licenses/assets/fastphase`.

A sample command to run fastPHASE and generate the cluster frequency, rhat and cluster theta files is the following:

`fastPHASE -Xu<subpoplabels.inp> -Pp -T1 -K15 <fastphase.inp>`

where `<subpoplabels.inp>` and `<fastphase.inp>` are filenames (the user would exclude the "<" marks). "subpoplabels.inp" is a file with *one row* of integers (1 per individual in the input file), separated by spaces; each integer is a label, indicating the subpopulation from which each individual was sampled. (The integer "-99" is reserved for missing/ unknown.)

See the "Input File Descriptions" section of this tutorial for a description of these files. Haploscope is also bundled with example fastPHASE output files so that you can modify your own datasets to these formats if you choose not to use fastPHASE.

APPENDIX B: Configuring Display Parameters

Haploscope parameters that control font sizes and dimensions of the GUI-based displays and exported images are configurable through a properties file. There is a `haploscope.properties` file, which has some display properties that can be configured. The `haploscope.properties.original` file is read-only by default, so that if you make changes to the `haploscope.properties` file and you want to revert your changes back to the original settings, simply copy the `haploscope.properties.original` file to your `haploscope.properties` file.

Some useful parameters to adjust are the following:

- GUI-display plot properties:
  - Font size of the title: `haploscope.plot.title-font-size=16`
  - Font size of annotation labels: `haploscope.plot.label-font-size=8`
  - Height of header area: `haploscope.plot.header-height=40`
  - Height of drawing area: `haploscope.plot.drawing-height=320`
  - Height of annotation area: `haploscope.plot.label-height=60`
  - Minimum width of individual plots: `haploscope.plot.minimum-width=350`

- Corresponding exported image properties:
  - Font size of the title: `haploscope.image.title-font-size=40`
  - Font size of annotation labels: `haploscope.image.label-font-size=20`
  - Font size of legend text in the summary image: `haploscope.image.legend-font-size=20`
  - Height of header area: `haploscope.image.header-height=60`
  - Height of drawing area: `haploscope.image.drawing-height=1000`

    - Height of annotation area: `haploscope.image.label-height=220`
    - Minimum width of image (this is a suggested width, which Haploscope may enlarge to fit the graphics more evenly): `haploscope.image.minimum-width=1200`

- Additional properties for configuration of the allele-frequency-detail heatmap images:
    - Font size of labels: `haploscope.heatmap.label-font-size=11`
    - Width of a heatmap cell: `haploscope.heatmap.col-width=12`
    - Height of a heatmap cell: `haploscope.heatmap.row-height=12`

As a note, some combinations of property values will not work well in the display, and Haploscope will not try to correct for such values. For example, if you make a font size larger than the display area, Haploscope will not try to correct for this. There are several additional parameters that can be modified to more fine-tune the layout of plots. See the `haploscope.properties` file to see what else might be configurable.

References

1. Scheet and Stephens. Am J Hum Genet. 2006; 78:629-644.
2. Browning and Weir. Genetics. 2010; 185:1337-1344.