eSPiGA

Statistical Population Genomic Analyses

Jordi Leno-Colorado[1], Luca Ferretti[2], Emanuele Raineri[3],
Giacomo Marmorini[4], Joan Jene[1], Gonzalo Vera[1],
Julio Rozas[5], Sebastian E. Ramos-Onsins[1]

[1]Centre for Research in Agricultural Genomics (CRAG),
CSIC-IRTA-UAB-UB Consortium, Bellaterra, Spain.
[2]Big Data Institute (BDI), University of Oxford, United Kingdom.
[3]Centro Nacional de Anlisis Genmico (CNAG-CRG), Barcelona, Spain.
[4]Yukawa Institute for Theoretical Physics (YITP), Kyoto, Japan.
[5]Dept. Genètica, Microbiologia i Estadística
& Institut de Recerca de la Biodiversitat (IRBio),
Universitat de Barcelona, Barcelona, Spain.

May 7, 2020

STILL UNDER DEVELOPMENT:

1. We need to compile some functins for macos and linux OS, (gVCF2tFasta, indexingtFasta and mergetFasta are C++ programs that now only compile for Linux, we need also to compile for macos).

2. We need that the graphical interface works just by clicking the icon. Nowadays only works when is executed from the Terminal.

3. We may improve a little bit the Interface. The most conservative way is to make transparent the options not chosen.

4. We will change the logo by a spigot.

Contents

1 eSPIGA overview

eSPIGA is a software package designed for the analysis of genome variability of multiple populations, specifically focused on the analyses of variant frequencies. The package can work with multiple format alignment (gVCF, fasta, tFasta and ms), annotation files (GFF/GTF) and additional filter files. A common problem using High-Throughput Sequence data is to deal with large quantity of missing data, which can substantially restrict the analysis of variability to few regions, or perform it with a reduced number of statistics. Here, the core of the package, *mstatspop*, calculates an extended number of population and variability-related statistics and neutrality test accounting for positions including missing data. In addition, optimal neutrality tests are for first time implemented. A number of file outputs are available, including simple text tables, Site Frequency Spectrum for one or several populations, output files to be used in other softwares, file showing sliding window stats and fully extended formats. Finally, filtering options are available to obtain the desired output.

eSPIGA provides a user-friendly graphical user interface (GUI), which helps to the user to chose the options and necessary flags for the analysis of sequence variability. The final result is a file with all the statistics obtained from the performed analysis. These results can be showed in a tab-formatted table in the interface. The interface is divided in three main sections: (i) pre-processing: where sequence format converters, annotation files and additional filtered options are managed to be ready for the analysis with *mstatspop* program, (ii) analysis: calculation of all statistics given the desired options of the user, including sequence and annotation files, and (iii) post-processing: the obtained statistics are showed and can be filtered from the whole output file(s) and released in text tab separated tables. A window including all the selected commands are visualised at the bottom of the interface. these commands can be copied and included in a file or directly run on a terminal. This framework allows a fast automation of pipelines for posterior analysis.

2 Software URL location, License and Disclaimer

:

2.1 URL

http://www.github.com/CRAGENOMICA/GSAW
(Provisional name, now the package is called GSAW)

2.2 License

eSPIGA graphical interface is a free software, under any of the following licenses: LGPL version 3, GPL version 2 and GPL version 3. All users have the rights to obtain, modify and redistribute the full source code of your application. Your users are granted rights founded on the four freedoms of the GNU General Public License. Any modification to a Qt component covered by the GNU Lesser General Public License must be contributed back to the community. This is the primary open source Qt license, which covers the majority of Qt modules.

All C, C++ and perl packages included in eSPIGA are free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

All packages included in eSPIGA s distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public Licenses (http://www.gnu.org/licenses/) for more details.

3 Installation

3.1 Github

The code is uploaded in github (http://www.github.com). To download the package, there are two options, either clone the repository, that is, copying the git command to execute on a terminal:

git clone https://github.com/CRAGENOMICA/GSAW.git
(Provisional name, now the package is called GSAW)
or just download the code directly into your computer.

3.1.1 Packages included

eSPIGA calls a number of packages already uploaded in github. The packagers can be cloned with the following commands:

-mstatspop: *git clone https://github.com/CRAGENOMICA/mstatspop.git*
-fastaconvtr: *git clone https://github.com/CRAGENOMICA/fastaconvtr.git*
-weight4tfa: *git clone https://github.com/CRAGENOMICA/weight4tfa.git*
-gVCF2tFasta: *git clone https://github.com/CRAGENOMICA/gVCF2tFasta.git*

-indexingtFasta: *git clone https://github.com/CRAGENOMICA/indexingtFasta.git*
-mergetFasta: *git clone https://github.com/CRAGENOMICA/mergetFasta.git*
-concatenate_tFasta: *git clone https://github.com/CRAGENOMICA/concatenate_tFasta.git*

3.2 Compilation

The repository contains several folders and files. Run first ***sh ./compile-GStatsPop.sh***, from this folder to download and compile all the necessary programs related to G-StatsPop. finally. click twice on the executable icon named G-StatsPop to automatically open the graphical interface.
(Compilation only works for linux for some functions. Need also to compile for macos.
More, the executable DOES NOT WORK just by clicking, it is necessary to execute from the Terminal.
The interface needs to be somewhat modified to be more clear (perhaps the easiest way is to make much more transparent the window options that are not chosen.)

3.3 Additional features

An additional program designed to visualize T-fasta files (gzip and indexed) is also available and can be opened by clicking the icon Tfasta-Viewer.

4 Getting started

4.1 Simulation analysis

A number of simulator programs, such as the coalescent Hudson simulator textitms (Hudson, 2002) *mlcoalsim* (Ramos-Onsins and Mitchell-Olds, 2007), *slim* (Haller and Messer, 2019) and others have the option to output in ms format (Hudson, 2002). This format includes the matrix of variants for a number of desired replicates. The core of eSPIGA, *mstatspop*, can perform simulation analysis for each of the replicates using the ms, format, and to obtain descriptors and variability-related statistics for single or multiple populations and including or not outgroup species populations.

4.2 Single-locus analysis

eSPIGA can perform single-locus analysis using a fasta file as input. The analysis could be done using all the sequence of the input fasta file. Additional filters can be included to analyze specifically a type of regions (*e.g.*, intron, exon, 3'UTR, etc...) or positions (*e.g.*, silent, synonymous, four times degenerated...), which can be detailed in a tab text file and/or including an annotation file (in GFF/GTF format).

4.3 Genomic analysis using sliding windows

eSPIGA perform genomic calculations using sliding windows. For this purpose, the software uses a tFasta as input (transposed Fasta), where each row corresponds to a sequence position and each column corresponds to a different sample, therefore each column contains the whole sequence of each sample. This format is created in order to include the information about missing data and monomorphic positions in a easy-to-read format and less heavy than gVCF format file. Tfasta must be indexed and gzipped to perform the analysis. Several options are allowed for genomic analysis, including filtering for specific fragments and/or annotation file. The sliding window analysis can be overlapped or not.

4.4 Conversors and other utilities

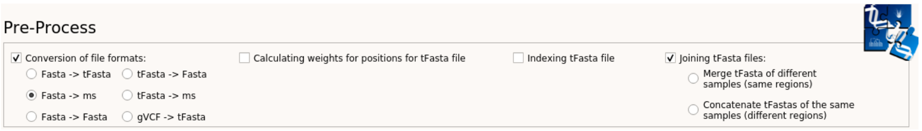
eSPIGA contains a number of format conversors. With these conversors it is possible to obtain fasta, tFasta -section 5.1.1- or ms,-format files from Fasta or tFasta) files as well as with gVCF. Besides, eSPIGA contains other utilities like the generation of a weight file for the tFasta files, indexation and compression and merge different tFasta files (both to join different chromosomes of the same sample and to join the same sequence of different samples)

5 Workflow using eSPIGA

The graphical interface of eSPIGA counts with three graphical interfaces: (i) Pre-processing, to perform format conversions if necessary or generate needed files for the analysis; (ii) *mstatspop* analysis, where the user would select the parameters for the desired analysis; and (iii) Post-processing, where the results will be showed in table format and the user can filter this table by the desired parameters and download it.

5.1 FIRST SCREEN: Pre-processing

In addition to performing format conversions, the user can also index tFasta files, join tFasta files (horizontally or vertically) or generate a file with the weights for the positions of the Tfasta files.



Pre-Process

☑ Conversion of file formats:
○ Fasta -> tFasta ○ tFasta -> Fasta
● Fasta -> ms ○ tFasta -> ms
○ Fasta -> Fasta ○ gVCF -> tFasta

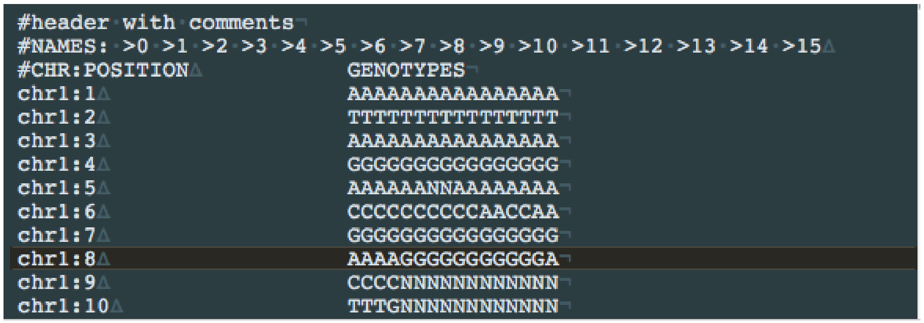
□ Calculating weights for positions for tFasta file

□ Indexing tFasta file

☑ Joining tFasta files:
○ Merge tFasta of different samples (same regions)
○ Concatenate tFastas of the same samples (different regions)

5.1.1 The tFasta format

The tFasta format is essentially a transposed fasta format. An example of a (unzipped) tFasta format is the following:



```
#header with comments
#NAMES: >0 >1 >2 >3 >4 >5 >6 >7 >8 >9 >10 >11 >12 >13 >14 >15
#CHR:POSITION                GENOTYPES
chr1:1                       AAAAAAAAAAAAAAAA
chr1:2                       TTTTTTTTTTTTTTTT
chr1:3                       AAAAAAAAAAAAAAAA
chr1:4                       GGGGGGGGGGGGGGGG
chr1:5                       AAAAAANNAAAAAAAA
chr1:6                       CCCCCCCCCCAACCAA
chr1:7                       GGGGGGGGGGGGGGGG
chr1:8                       AAAAGGGGGGGGGGGA
chr1:9                       CCCCNNNNNNNNNNNN
chr1:10                      TTTGNNNNNNNNNNNN
```

The first rows can contain comments (precede by a '#' character. Then, the names of the sequences are defined by a row starting with '@NAMES:': each lineage separated at least by one space/tab and starting with the character '¿'. A header comment can be included, indicating that the first column contains the information of the chromosome/scaffold name and the position and the next column contains the nucleotides for all the lineages, in order. tFasta is indexed and zipped to be used with *mstatspop*.

5.1.2 File format conversion

eSPIGA can convert among different formats (ms, fasta, tFasta and gVCF). In case of generate simulations with a given distribution of valid positions, the conversion of tFasta or fasta to ms files is useful to generate mask files. The conversion from gVCF or Fasta files to tFasta is useful to perform the analysis in sliding windows mode.

If the option 'Calculating weights for positions' is selected, the conversion from Fasta to tFasta and viceversa generates a weighting file based in the alignment and in the annotation file (General Feature Format, GFF), if the user does not indicate the contrary. This weighting file have three columns: (i) the physical position, (ii) weight score for positions and (iii) boolean weight for the variant, if the position is being taken into account (1) or not (0), which depends on the type of mutation that the user is analyzing. Therefore, weighting files are something like this:

Chromosome:Position	Weight	WeightVar
1:1	0.000	1.000
1:2	1.000	1.000
1:3	0.000	1.000
1:4	0.000	1.000
1:5	0.333	0.000
1:6	0.000	1.000
1:7	0.306	1.000
1:8	0.333	1.000
1:9	0.000	1.000
...	...	...

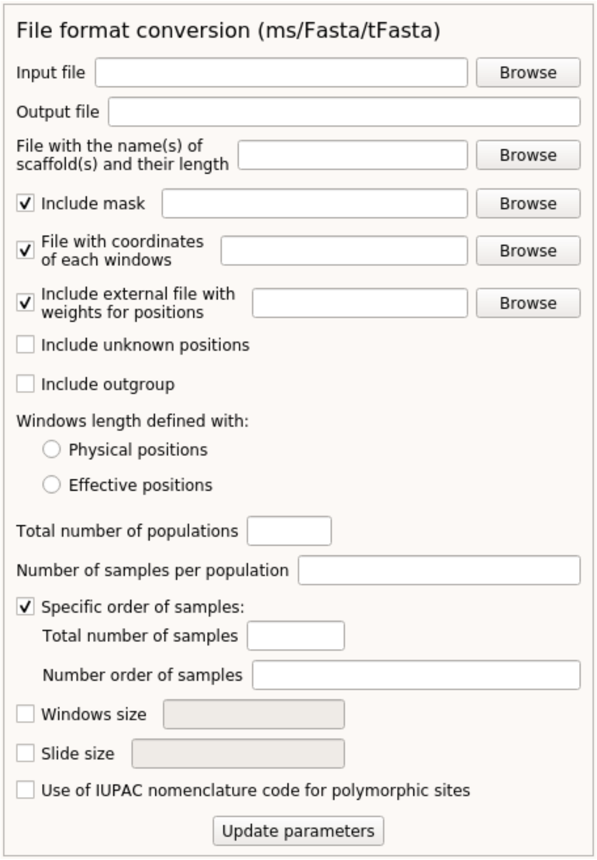
For the file format conversions among ms, fasta and tFasta, it is mandatory to include the input file, the output file (including the extension with .gz), the input and outputs formats and a file with the name of scaffolds and their lengths (separated by tab), one example of this type of file is:

1	315321322
2	162569375
3	144787322
4	143465943
5	111506441
...	...

As optional parameters, the user can include an outgroup (would be the last population), take into account unknown positions (missing data), define the window lengths in 'physical' or 'effective' positions and specify the order of samples. In case to include an outgroup population, is necessary indicate the total number of populations and the number of samples per population. Furthermore, the user may indicate a file to mask regions, with the start and the end of regions (it will mask these positions with Ns) like in this example:

Chromosome	Initial	Final
10	1	10000
10	20001	30000
10	40001	50000
10	60001	70000
10	80001	90000

If the output file is ms or Fasta, the user can include a file with the coordinates of each window that he wants to analyze (with the same structure than the previous one, scaffold-init-end). In the case of outputing ms format, the user also can define the window and the slide size. Furthermore, if the input is a Fasta file, and each sequence defines a genotype, it is possible to use the IUPAC nomenclature code for the polymorphic sites (unphased data).



File format conversion (ms/Fasta/tFasta)

Input file

Output file

File with the name(s) of scaffold(s) and their length

☑ Include mask

☑ File with coordinates of each windows

☑ Include external file with weights for positions

□ Include unknown positions

□ Include outgroup

Windows length defined with:

○ Physical positions

○ Effective positions

Total number of populations

Number of samples per population

☑ Specific order of samples:

Total number of samples

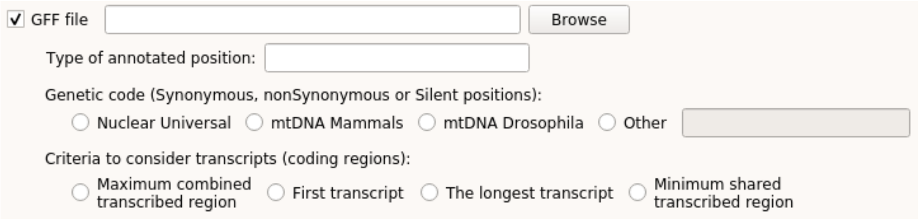
Number order of samples

□ Windows size

□ Slide size

□ Use of IUPAC nomenclature code for polymorphic sites

The user can include a GFF file and define the type of position that wants to analyze (coding, noncoding, synonymous, nonsynonymous, silent, etc.). If synonymous, nonsynonymous or silent is selected, the user have to add the desired genetic code. Defined codes are: Nuclear universal, mtDNA of Drosophila and mtDNA of Mammals. In case to use another genetic code, the option ”Other” must be selected. In such case, the user have to introduce the single letter code for the 64 triplets in the order UUU UUC UUA UUG, etc... In case of using coding regions, it is mandatory to select the criteria for multiple splicing. The transcript to consider can be one of the given choices: the maximum combined transcribed region, the minimum shared transcribed region, the first transcript in the annotation file, or the longest transcript.



☑ GFF file Browse

Type of annotated position:

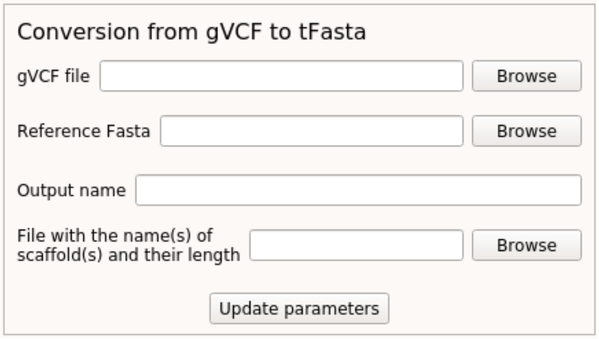
Genetic code (Synonymous, nonSynonymous or Silent positions):

○ Nuclear Universal ○ mtDNA Mammals ○ mtDNA Drosophila ○ Other

Criteria to consider transcripts (coding regions):

○ Maximum combined transcribed region ○ First transcript ○ The longest transcript ○ Minimum shared transcribed region

In case of converting a file from gVCF to tFasta format, the user have to select the input gVCF file, the reference Fasta file, the output name (without extension) and a file with the name of scaffolds and their lengths (as described above). In this conversion a weighting file is not generated.



Conversion from gVCF to tFasta

gVCF file Browse

Reference Fasta Browse

Output name

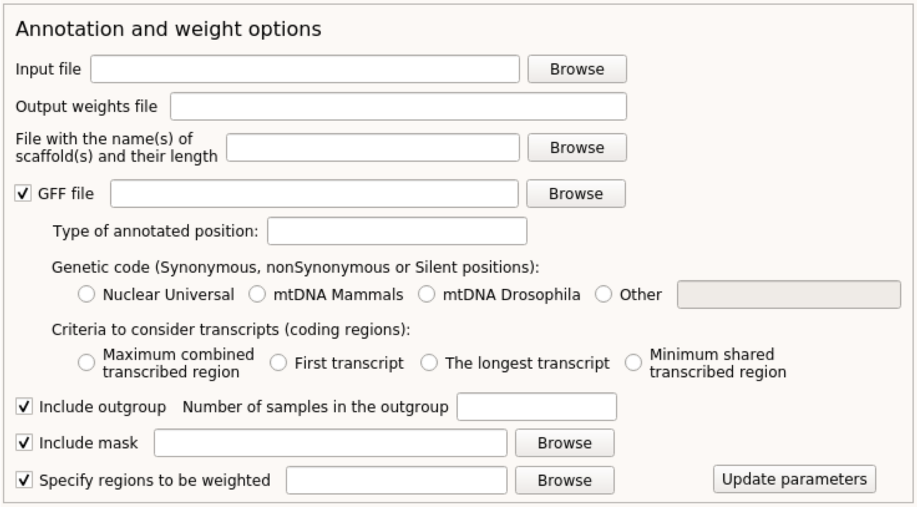
File with the name(s) of scaffold(s) and their length Browse

Update parameters

5.1.3 Annotation and weighting options

eSPIGA can also calculate the weights for positions for tFasta files without any format conversion. In this case, the user have to select the input and output (with .gz extension) files, the file with the name of scaffolds and their lengths (as described previously), and a GFF file to define the type of position to analyze (coding, noncoding, synonymous, nonsynonymous, silent, etc.) and the genetic code plus the criteria for multiple splicing, if necessary (see subsection above).

As optional parameters, the user can specify the existence of an outgroup species, plus its number of samples, mask regions, including a file indicating the start and the end of regions to be weighted, and specify the coordinates of regions to analyze, including a file with the start and end positions of regions to be weighted, rest would be weighted as 0.



Annotation and weight options

Input file [ ] Browse

Output weights file [ ]

File with the name(s) of scaffold(s) and their length [ ] Browse

☑ GFF file [ ] Browse

Type of annotated position: [ ]

Genetic code (Synonymous, nonSynonymous or Silent positions):

○ Nuclear Universal ○ mtDNA Mammals ○ mtDNA Drosophila ○ Other [ ]

Criteria to consider transcripts (coding regions):

○ Maximum combined transcribed region ○ First transcript ○ The longest transcript ○ Minimum shared transcribed region

☑ Include outgroup Number of samples in the outgroup [ ]

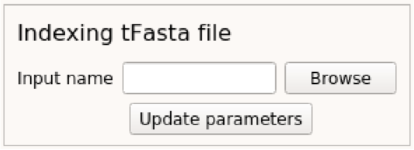
☑ Include mask [ ] Browse

☑ Specify regions to be weighted [ ] Browse Update parameters

5.1.4 Indexing tFasta file

An available option allows to index tFasta unindexed files (which is a requirement to analyze the samples with *mstatspop*)This option is useful when converting gVCF to Tfasta files.

For this purpose, the user only have to select the tFasta file to index. An additional file will appear after indexing.



Indexing tFasta file

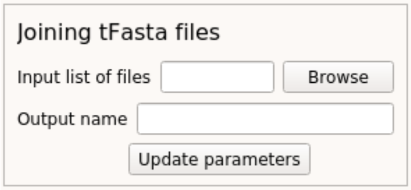
Input name [ ] Browse

Update parameters

5.1.5 Joining tFasta files

Joining different tFasta files can be horizontally or vertically, i.e. merge the same region from tFasta files of different samples or concatenate different regions tFasta files of the same sample.

To join two equal tFasta fragments/chromosomes from different samples/populations (horizontal join), the user have to upload a file of a list with the paths and names of the tFasta files, one per line, and introduce a name for the output.



Joining tFasta files

Input list of files [ ] Browse

Output name [ ]

Update parameters

In case joining two different fragments from the same samples, note that the option just concatenate two unzipped (text format) and unindexed files. To do that, use concatenate_tFasta window.

5.2 SECOND SCREEN: *mstatspop* analyses

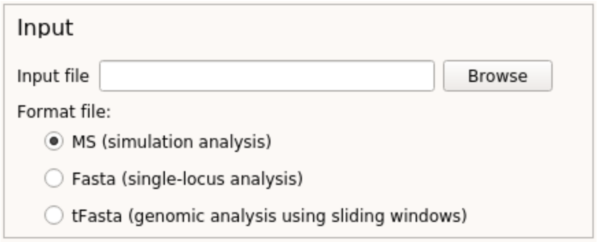
The second interface is the more important section of the package, and constitutes the core of the analyses. The package performs the main analysis of genome variability using *mstatspop* program. *mstatspop* calculates statistics of variability using multiple population in tFasta (transposed Fasta), Fasta or ms format files.

The program offers multiple options, missing values are allowed, IUPAC code for diploid individuals can be processed, Fst comparisons and permutation test can be performed among all populations, etc. *mstatspop* can calculate the variability and neutrality tests based on frequency spectrum in data considering positions with missing values. To perform the analysis, the user have to select all the required parameters, in addition to certain parameters that are optional according to each analysis.

When all necessary parameters of the analysis are filled, the user can execute the program through the 'Run' button at the bottom of the interface or, if the execution spent presumably long time, copy the command to execute externally. Note that the actions selected in the pre-process, analysis and post process are shown in the 'Run' window, which can be executed at any time. All the commands in the 'Run' window will be executed sequentially when the button will be press.

5.2.1 Input and Output parameters

In a first step, the user selects the input file and its file format: ms for simulation analysis, Fasta for single-locus analysis and tFasta for genomic analysis using sliding windows. Depending on the selected input format, eSPIGA displays specific parameters of each type.



Input

Input file

Format file:

● MS (simulation analysis)

○ Fasta (single-locus analysis)

○ tFasta (genomic analysis using sliding windows)

The output path and name will be specified and the user have to select among different options of output format file:

- Extended output: Output in an extended list format with header for each set of statistics.

- Single line/window: All statistics in a single line per window.

- Single line SFS/window: Site frequency spectrum (SFS) in a single line per window.

- dadi-like format: File format to use as input in δaδi software.

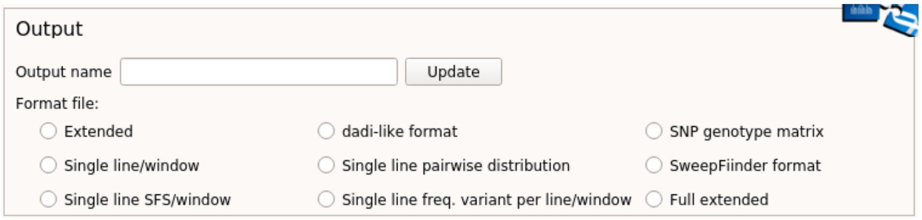
- Single line pairwise distribution: All pairwise comparisons (mismatch distribution) in a single line

- Single line freq. variant per line/window: Frequency of variants for each line, one windows per line.

- SNP genotype matrix

- SweepFiinder format: File format to use as input in SweepFiinder software.

- Full extended: Extended format plus δaδi-like format, all pairwise comparisons and frequency of SNPs of each line.



Output

Output name [   ] Update

Format file:

○ Extended ○ dadi-like format ○ SNP genotype matrix

○ Single line/window ○ Single line pairwise distribution ○ SweepFiinder format

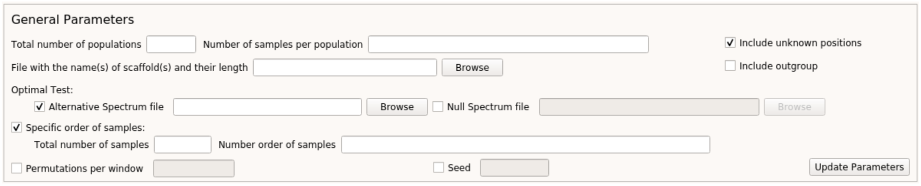
○ Single line SFS/window ○ Single line freq. variant per line/window ○ Full extended

5.2.2 General parameters

Other required parameters for any analysis with *mstatspop* are the number of populations and the number of samples per each population and a file with the name of scaffolds and their lengths (as the described file in the **File format conversion** section). As optional and general parameters for all analysis the user can include an outgroup species (would be the last population), take into account unknown positions (missing data) and include an alternative site sprectrum file with an alternative spectrum for each population, except outgroup (this option is only to perform Optimal test). If the alternative spectrum file is included, the user can also include a null spectrum file with the null spectrum of each population, if not included, the standard neutral model (SNM) is considered as default. These spectrum files have a file format like this:

fr[0,1]	fr[0,2]	fr[0,3]	fr[0,4]	fr[0,5]	fr[0,6]	fr[0,7]	fr[0,8]	...
2	1	0.6667	0.5	0.4	0.3333	0.2857	0.25	...
3	1.5	1	0.75	0.6	0.5	0.4286	0.375	...
...	...	...	...	...	...	...	...	...

If the input file is a Fasta or tFasta file, the user can specify the specific order of the samples. For this purpose the user must introduce the number of total samples and the number order of each sample (0 is the first sample) separated by a white space. Moreover, with Fasta and tFasta as input files, the user should specify a seed number and the number of permutations per window to make Fst confident intervals.



General Parameters

Total number of populations Number of samples per population ☑ Include unknown positions

File with the name(s) of scaffold(s) and their length □ Include outgroup

Optimal Test:

☑ Alternative Spectrum file □ Null Spectrum file

☑ Specific order of samples:

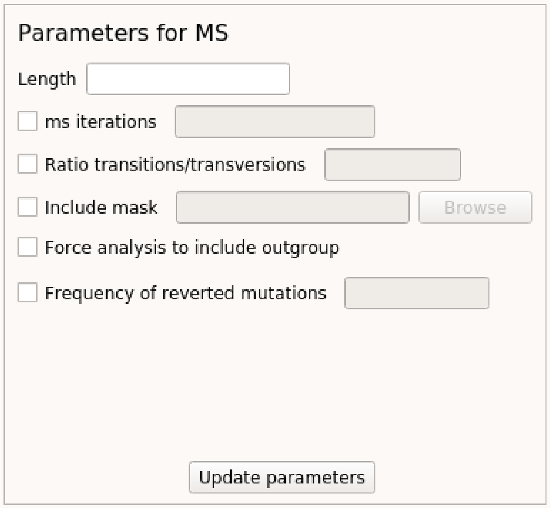
Total number of samples Number order of samples

□ Permutations per window □ Seed

5.2.3 MS parameters

For the simulation analysis of a single region, using ms format as input file, the user have to specify the length of the sequence as an additional required parameter. Additional optional parameters are the number of ms iterations, the ratio of transitions/transversions (the default is 0.5), include an outgroup and, if the outgroup is included, the frequency of the reverted mutation. The user also can include a mask file to exclude part(s) of the region, the format of this file is the first row with length weights and the next samples rows with 0 if the position is missing and 1 if it is sequenced, something like this:

```
1.000 1.000 1.000 1.000 1.00 1.000 1.000 1.000 1.00 1.000 1.000 1.000 ...
111100001111...
111100111111...
111101101111...
```



Parameters for MS

Length

□ ms iterations

□ Ratio transitions/transversions

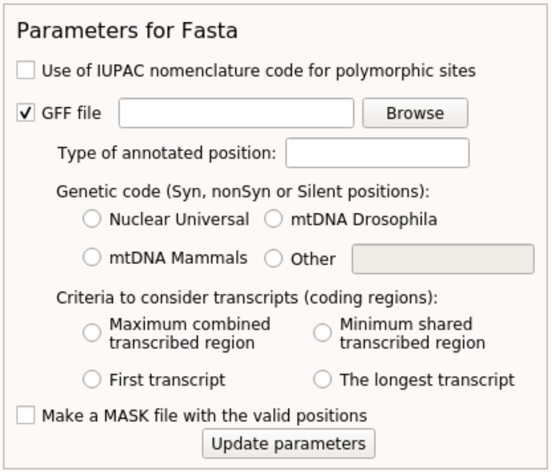
□ Include mask

□ Force analysis to include outgroup

□ Frequency of reverted mutations

5.2.4 Fasta parameters

For single-locus analysis, using a Fasta file as input, there are some optional parameters like the use of IUPAC nomenclature code (that is, in case assuming 2 nucleotides per lineage -diploid-), the number of lineages per sequence (1 or 2) or the possibility of making a mask file with the valid positions of the input Fasta file, which is useful for running subsequent ms simulations. Furthermore, the user can include a GFF file to define the type of position to analyze (coding, noncoding, synonymous, nonsynonymous, silent, etc.). If synonymous, nonsynonymous or silent is selected, the user have to add the genetic code: Nuclear universal, mtDNA of Drosophila, mtDNA of Mammals or Other (in this case the user have to introduce the single letter code for the 64 triplets in the order UUU UUC UUA UUG ...). In the case of using coding regions, select the criteria to consider alternative splicing: the maximum combined transcribed region, the minimum shared transcribed region, the first transcript or the longest transcript (by default).



Parameters for Fasta

□ Use of IUPAC nomenclature code for polymorphic sites

☑ GFF file [ ] Browse

Type of annotated position: [ ]

Genetic code (Syn, nonSyn or Silent positions):

○ Nuclear Universal ○ mtDNA Drosophila

○ mtDNA Mammals ○ Other [ ]

Criteria to consider transcripts (coding regions):

○ Maximum combined transcribed region ○ Minimum shared transcribed region

○ First transcript ○ The longest transcript

□ Make a MASK file with the valid positions

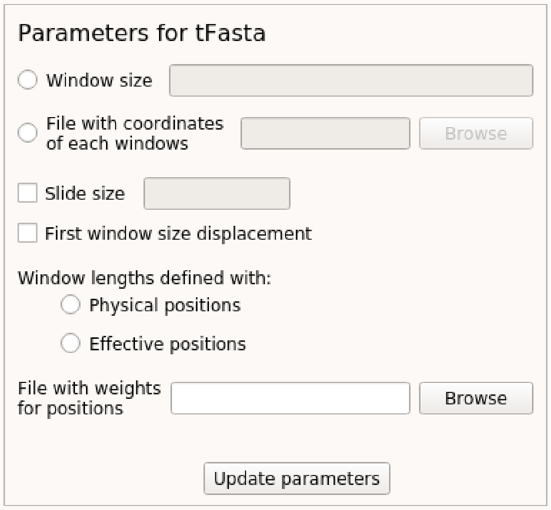
Update parameters

5.2.5 tFasta parameters: Analysis on Genome datasets by using Sliding Windows or Genomic Regions

For genomic analysis using sliding windows, the input will be a tFasta file and the user have to select if the analysis will be done through windows of a determined size (in this case the user also can select the distance between windows, the slide size) or using an external file with the coordinates of each window, like this example:

Chromosome	Initial	Final
10	1	10000
10	20001	30000
10	40001	50000
10	60001	70000
10	80001	90000

It is also possible to move the first window to compare overlapped windows and define window lengths in physical or effective positions. The user also can select a file with weights for positions, as the created in the **Pre-process** step.



Parameters for tFasta

○ Window size

○ File with coordinates of each windows [Browse]

□ Slide size

□ First window size displacement

Window lengths defined with:

○ Physical positions

○ Effective positions

File with weights for positions [Browse]

[Update parameters]

5.3 THIRD SCREEN: Post-processing

If the output format of the previous analysis is the ”single line/window”, in the last step, **Post-processing**, will be displayed the results in a table format, where each column corresponds to a statistic or parameter and each row corresponds to a different window. In this table, the user can select the desired columns by clicking in the header and then it will be possible to download the filtered table with only these selected columns through the ’Download Table’ button at the bottom of the interface.

5.3.1 Output statistics and parameters

The statistics and neutrality tests calculated in the analysis and displayed in this step are the following:

- General information:
  - infile: Path and name of the input file
  - scaffold_name: Scaffold name
  - start_window: Start position of the window
  - end_window: End position of the window
  - missing: include unknown positions
  - iteration:
  - npermutations:
  - seed: Seed number
  - Length:
  - Lengtht:

  - mh:
  - Ratio_S/V: Ratio transitions/transversions
  - Ratio_Missing: Ratio of missing data in the sequence
  - Variants: Number of variants
  - npops: Number of populations
  - nsam[i]: Number of samples at population i
  - Eff_length1_pop_outg[i]: Effective length for population i with at least one sequence
  - Eff_length2_pop_outg[i]: Effective length for population i with at least two sequences
  - Eff_length3_pop_outg[i]: Effective length for population i with at least three sequences

- Estimates of variability for each population:
  - S[i]: Number of biallelic segregating sites at population i.
  - Theta(Wat)[i]: Watterson's estimate of nucleotide variation (θ) Watterson (1975)) at population i.
  - Theta(Taj)[i]: Tajima's estimate of nucleotide variation per locus (nucleotide diversity, π) Tajima (1983) at population i.
  - Theta(Fu&Li)[i]: Fu and Li's estimate of variability, based on singleton Fu and Li (1993)) at population i.
  - Theta(Fay&Wu)[i]: Fay and Wu's estimate of nucleotide variation Fay and Wu (2000) at population i.. Useful for detection of high frequency variants produced by selective process.
  - Theta(Zeng)[i]: Zeng variability estimate Zeng *et al.* (2006) at population i.
  - Theta(Achaz,Wat)[i]: Watterson estimate without considering singletons Achaz (2008) at population i.
  - Theta(Achaz,Taj)[i]: Tajima estimate without considering singletons Achaz (2008) at population i.
  - Theta/nt(Taj)HKY[i]: Tajima estimate considering the evolutionary model HKY (Hasegawa *et al.* (1985)
  - an_xo[i]: $\sum_{j=1}^{nsam-1}(\frac{1}{j})$ at population i.
  - bn_xo[i]: $\sum_{j=1}^{nsam-1}(\frac{1}{j^2})$ at population i.
  - Divergence[i]: Divergence estimate between the population i and the outgtroup
  - Divergence/nt_HKY[i]: Divergence estimate using the evolutionary model HKY (Hasegawa *et al.* (1985)between the population i and the outgtroup

- Neutrality tests for each population:
  - TajimaD[i]: Tajima's D test Tajima (1989). This test basically looks at the differences between Watterson's estimate θ, and Tajima's estimate π (Tajima, 1983). Significant negative values indicate an excess of low frequency variants while positive values indicate an excess of intermediate frequency variants.
  - Fu&LiD[i]: Fu and Li's D test with outgroup Fu and Li (1993). Fu and Li's tests are similar to Tajima's D test, but using different statistics related to the level of diversity.
  - Fu&LiF[i]: Fu and Li's F* test without outgroup Fu and Li (1993).
  - Fay&WunormH[i]: Fay and Wu's H normalized test Fay and Wu (2000).
  - ZengE[i]: Zeng *et al.* neutraulity test Zeng *et al.* (2006).
  - AchazY[i]: Like Tajima's test but not considering singletons Achaz (2008).
  - FerrettiL[i]: Ferretti test, similar to Tajima's D or Fay and Wu's F test, which considers the difference between Watterson versus Fay and Wu theta estimators Ferretti *et al.* (2018).
  - R2[i]: Ramos and Rozas' R2 test Ramos-Onsins and Rozas (2002). Test to detect demographic expansion using small sample sizes in any recombinant or non-recombinant environment.

- Variants assigned to different types of positions:
  - Sx[i]: Number of exclusive variants for each population.
  - Sf[i]: Number of fixed variants for each population.
  - Sxf[i,rest]: Number of polymorphic variants at population i but fixed in the rest of populations.
  - Ss[i]: Number of shared variants for each population.

- Mismatch distribution statistics:
  - MD_SDev[i]: Standard deviation of the pairwise distribution (π).
  - MD_Skewness[i]: Skewness of the pairwise distribution.
  - MD_Kurtosis[i]: Kurtosis of the pairwise distribution.

- Fst: Genetic differentiation among populations F$_{st}$ Hudson *et al.* (1992), calculated as:

$$1 - PiW/PiA \tag{1}$$

  - Fst and P-value: Average differentiation among all populations and P-value.

  - Fst1[i,rest] and P-value: Differentiation of each population across all the rest (except outgroup) and P-value.
  - Fst[i,j] and P-value: Differentiation between populations *i* and *j*.

- Nucleotide and haplotype diversity:
  - PiW[i]: Average nucleotide diversity within populations per locus (see for example Hudson *et al.*, 1992)
  - PiA[i,j]: Nucleotide diversity among populations per locus (see for example Hudson *et al.*, 1992)
  - PiT[i,j]:
  - len[i,j]:

- Frequency of variants for each population:
  - fr[i,l]: Frequency of variants for population *i* and lineage *l*

6 Authors Contribution

Jordi Leno-Colorado constructed the Graphical interface and the included functions of the packages mergetFasta, indexingTFasta, concatenateTFasta and the gvcf2TFasta codes in C++. Luca Ferretti, Emanuele Raineri and Giacomo Marmorini developed the functions for optimal tests and contributed to the missing data functions. Joan Jene developed the index and compress functions for reading and writing input and output files ,developed the program tfaviewer for visualizing tfasta files and contributed to many debugging hours. Gonzalo Vera modified a number of sections in mstatspop code and contributed to the software idevelopment of the project. Julio Rozas contributed to the development and the final steps of the application. Sebastian E. Ramos-Onsins developed most of the project and wrote mstatspop, fasta_convrtr and weight4tfasta codes in C.

7 Bibliography

Achaz, G., 2008 Testing for neutrality in samples with sequencing errors. Genetics **179**: 1409.

Fay, J., and C.-I. Wu, 2000 Hitchhiking under positive Darwinian selection. Genetics **155**: 1405.

Ferretti, L., A. Klassmann, E. Raineri, S. E. Ramos-Onsins, T. Wiehe, *et al.*, 2018 The neutral frequency spectrum of linked sites. Theor Popul Biol **123**: 70–79.

Fu, Y., and W.-H. Li, 1993 Statistical tests of neutrality of mutations. Genetics **133**: 693.

Haller, B. C., and P. W. Messer, 2019 Slim 3: Forward genetic simulations beyond the wright-fisher model. Mol Biol Evol **36**: 632–637.

Hasegawa, M., H. Kishino, and T. Yano, 1985 Dating of the human-ape splitting by a molecular clock of mitochondrial dna. J Mol Evol **22**: 160–74.

Hudson, R. R., 2002 Generating samples under a wright-fisher neutral model of genetic variation. Bioinformatics **18**: 337–8.

Hudson, R. R., D. D. Boos, and N. L. Kaplan, 1992 A statistical test for detecting geographic subdivision. Mol Biol Evol **9**: 138–51.

Ramos-Onsins, S. E., and T. Mitchell-Olds, 2007 Mlcoalsim: multilocus coalescent simulations. Evol Bioinform Online **3**: 41–44.

Ramos-Onsins, S. E., and J. Rozas, 2002 Statistical properties of new neutrality tests against population growth. Mol Biol Evol **19**: 2092–2100.

Tajima, F., 1983 Evolutionary relationship of DNA sequences in finite populations. Genetics **105**: 437.

Tajima, F., 1989 Statistical method for testing the neutral mutation hypothesis by DNA polymorphism. Genetics **123**: 585.

Watterson, G., 1975 On the number of segregating sites in genetical models without recombination. Theoretical population biology **7**: 256.

Zeng, K., Y.-X. Fu, S. Shi, and C.-I. Wu, 2006 Statistical tests for detecting positive selection by utilizing high-frequency variants. Genetics **174**: 1431–1439.