KLinterSel Manual v 0.3

Antonio Carvajal-Rodríguez

Centro de Investigación Mariña (CIM)

Departamento de Bioquímica, Genética e Inmunología

Facultad de Biología

Universidad de Vigo, Vigo 36310, Spain

Email: acraaj@uvigo.es

Web: http://acraaj.webs.uvigo.es

Table of Contents

Versions

Version KLinterSel 0.3 (December 2025):

- A new parametric test (HGkI) based on a sequentially conditioned hypergeometric distribution.

- A rank-based measure to assess deviations from uniformity in the spatial distribution of SNPs along the chromosome.

Version KLinterSel 0.2 (November 2025):

- Major speed-up of the KL computation thanks to a more efficient RelEntr implementation.

- New command-line options:

    --permissive to disable the strict redundancy filter between candidate files.

    --uniform perform resampling assuming a uniform distribution of SNPs.

- Sorted intersection output: elements within each intersection are now returned in sorted order for consistent and cleaner reporting.

- Bug fix: corrected an issue in the resampling test where the comparison of medians introduced a slight conservative bias.

Version KLinterSel 0.1 (August 2025):

- Improvements for figure customization. Multiprocessor options.

Version KLinterSel 0.0 (July 2025):

- This is the first version.

Introduction

KLinterSel is a Python project designed to assess the agreement among different methods for detecting candidate sites under selection. It implements a Monte Carlo–based test (T_{KL}) and a hypergeometric k-way intersection test (HGkI) to evaluate whether observed overlaps are closer than expected by chance. The program also computes intersections among candidates and provides complementary plotting and statistical summaries.

Pre-built Binaries Requiring No Installation

Pre-built KLinterSel binaries are available at https://github.com/noosdev0/KLinterSel/releases/tag/v0.3. Binaries are provided for Windows, Linux, and macOS (arm64) and should work on most versions of these operating systems. To run them (assuming that the files totsnps.txt, sigsmethod1.norm, sigsmethod2.tsv and sigsmethod3.norm exist and are in the same KLinterSel folder as the executable) type in the command-line interface (by default, the Monte Carlo TKL test is run; to use the hypergeometric test, simply add the --HG flag):

Linux: ./KLinterSel_U totsnps.txt sigsmethod1.norm sigsmethod2.tsv sigsmethod3.norm --path ./ --dist 10000

macOS: ./KLinterSel_OS totsnps.txt sigsmethod1.norm sigsmethod2.tsv sigsmethod3.norm --path ./ --perm 10000

Windows: Double click just for running the program under default options. The program will ask for the names of the files. Alternatively, you can go to the command prompt (cmd.exe) and type

C:\KLinterSel\KLinterSel.exe totsnps.txt sigsmethod1.norm sigsmethod2.tsv sigsmethod3.norm --path ./ --dist 100

or you can also access the Run command by pressing the Windows logo key +r

then drag and drop the .exe file from your folder and add the desired arguments, e.g.

```
C:\KLinterSel\KLinterSel_Win.exe totsnps.txt sigsmethod1.norm
sigsmethod2.tsv sigsmethod3.norm --path ./ --perm 1000 --dist 10
```

Installation

Clone the KLinterSel repository or download the files. To use the KLinterSel script (KLinterSel.py), you need to have Python installed (version 3.7 or higher). To install the necessary dependencies, navigate to the folder containing the KLinterSel.py script where the requirements.txt file should also be located, and run the following command in the terminal:

```
pip install -r requirements.txt
```

This command will install all required libraries as specified in the requirements.txt file, including numpy, pandas, matplotlib, seaborn, scipy and psutil, along with any other dependencies listed.

To run the .py script type in the command-line interface (assuming that the files totsnps.txt, sigsmethod1.norm, sigsmethod2.tsv and sigsmethod3.norm exist and are in the same folder as the script):

```
python3 KLinterSel.py totsnps.txt sigsmethod1.norm
sigsmethod2.tsv sigsmethod3.norm
```

Input (Data Format)

The script requires two types of input files:

1.- Original Positions Data File: The first file contains the positions of all analyzed SNPs. A CSV, TSV or text (.txt) file with the following structure:

```
CHR POS

1 12345

1 12367

.

.
```

The first column identifies the chromosome, and the second column lists the position of the SNP within the chromosome.

2.- Candidate Site Results Files: Files containing the candidate positions for each method. These files can have the same format as the original data file, but in addition files with a norm extension are also accepted. These files correspond to the output files from the norm-selscan program. In this case, a single chromosome is assumed, and physical positions with a value of one in the last column are selected.

Command Line Arguments

A full description of all available command-line options can be displayed by running:

```
python3 KLinterSel.py --help
```

Below is a breakdown of the various command line arguments that can be used with the script:

- **Basic Command**

  python3 KLinterSel.py totsnps.txt sigsmethod1.norm sigsmethod2.tsv sigsmethod3.norm

  This command processes the data without additional options it is equivalent to

  python3 KLinterSel.py totsnps.txt sigsmethod1.norm sigsmethod2.tsv sigsmethod3.norm --path . --perm 10000 --dist 10000 --SL 0.05.

- **Distance option**

  --dist 10000 Specify the distance threshold (default 10,000) for the intersection computation.

- **Hypergeometric k-way intersection test**

  --chr-id By default, when the --HG flag is used, the program computes the HGkI test for all chromosomes. If a

valid chromosome number is specified using --chr-id (any integer from 1 to the last chromosome), the HGkI test and the corresponding intersections are computed only for that chromosome.

--HG Run the hypergeometric k-way intersection test. When this option is enabled, the standard distance-based KLinterSel analysis is skipped. The window size for the hypergeometric test is defined by --W (or W = 1 if --W is not provided).

--W It defines the size of the disjoint units used by the hypergeometric test, independently of the distance scale used by KLinterSel. When W = 1 (default), units correspond to individual SNPs (exact overlap). When W > 1, units correspond to disjoint genomic windows of size W that contain at least one SNP. Note that when --W is large, multiple candidate SNPs within the same window are collapsed into a single event for the hypergeometric test.

- **Intersections**

  --chr-id By default, the program calculates intersections for all chromosomes. If a valid chromosome number is specified using--chr-id (any integer from 1 to the last chromosome), intersections are computed only for that chromosome. When the HGkI test is used, this flag also restricts the test itself to the specified chromosome. For the T_{KL} test, all chromosomes are still analyzed, but the intersections are reported only for the specified chromosome.

  --skip-intersection Skip SNP intersection computation.

- **Kmax option**

  --Kmax undefined If defined and the number of candidate SNPs in any file is greater than Kmax, it filters SNP positions to group those that are at distance ≤ D, leaving only the first and last of each group.

- **No test option**

  --notest Compute SNP intersections only. No statistical tests (T_{KL} or HGkI) are performed.

- **Number of permutations**

  --perm 10000 Number of permutations used to compute the expected distance profile (default: 10,000). Ignored when --HG is set.

- **Path Specification**

  --path ./home/KLinterSel/results/data Specify the directory where the input files are located.

- **Permissive with redundant methods**

  --permissive Disables the new default rule that removes candidate files showing high redundancy (i.e., when all or most candidate sites are identical across methods).

- **Plotting**

  --paint Only for T_{KL} test (ignored when --HG is set). Enable the generation of plots. In this case, intersection computations are skipped. The program draws two histograms: the one on the left shows the observed distances, and the one on the right shows the expected distance profile if the matches between methods were generated at random given the genomic locations of SNPs in the input data. If there are multiple chromosomes, the program displays the graph for each chromosome and, after closing it, continues with the next chromosome. If you only want a graph for a specific chromosome, for example chromosome 7, you can specify --paint 7.

  The scale of the X and Y axes can be controlled with the --max-xvalue and --max-yvalue arguments, respectively. The first is the Mb scale (--max-xvalue 20) and the second is a frequency between 0 and 1 (--max-yvalue 0.25).

- **Significance level**

  --SL 0.05 Set the significance level for the statistical tests (default 0.05).

- **Stats**

  --stats Generates statistics for each chromosome in each file, including minimum, maximum, mean, standard deviation, and median values. No other calculations are performed in this case.

- **Uniform**

  --uniform For the T_{KL} test, perform resampling assuming a uniform SNP distribution..

Output

The KLinterSel.py script generates three types of output files.

HGkI Test Results Output File (HGkItest_)

- This file reports the results of the HGkI test for each chromosome. It includes the total number of SNPs and occupied windows, the number of candidate SNPs and windows identified by each method, the observed number of overlapping windows (k_{obs}), the number expected under the null model E[k_{obs}], the associated p-value, and a rank-based measure of deviation from uniformity in the spatial distribution of SNPs along the chromosome (0 = lowest rank of deviation, 1 = highest rank of deviation), together with an independent, test-based classification derived from Kolmogorov–Smirnov, Cramér–von Mises, and cv_counts (coefficient of variation of SNP counts) statistics (close to uniform, medium deviation, or strong deviation).

  Under the null hypothesis of independent and uniformly distributed candidates across windows, the expected number of k-way overlapping windows is

$$E[k_{obs}] = \frac{\prod_{i=1}^{k} n_i}{N^{k-1}}$$

where n_i is the number of windows occupied by method i and N is the total number of windows in the given chromosome. This expectation ranges from 0 to the smallest number of windows reported by any method.

T_{KL} Test Results Output File (TKLtest_)

- This file reports the results of the T_{KL} test for each chromosome. It includes the total number of SNPs and the number of candidate SNPs identified by each method, a Kullback-Leibler-like discrepancy measure between the observed and expected ordered distance profiles, the associated p-value, the observed (oQ2) and expected (eQ2) medians, and a rank-based measure of deviation from uniformity in the spatial distribution of SNPs along the chromosome (0 = lowest rank of deviation, 1 = highest rank of deviation), together with an independent, test-based classification derived from Kolmogorov–Smirnov, Cramér–von Mises, and cv_counts (coefficient of variation of SNP counts) statistics (close to uniform, medium deviation, or strong deviation).

Intersections Result File (INTERSEC_test_D1E4_)

- This file contains, for each chromosome, the intersections among the different methods. The distance threshold used to define intersections is indicated in the file name; for example, _D1E4_ denotes a distance of 10 kb. Intersections are reported for all combinations of methods. For instance, when three methods are used (K = 3), both the pairwise intersections and the three-way intersection are provided. The last column lists the genomic sites involved in the intersection across all methods. Intersections are reported only for chromosomes that are significant under the selected statistical test (T_{KL} or HGkI). If intersections for all chromosomes are desired, the --

notest flag can be used, or equivalently the --SL 1 option.

Plots (if enabled)

- If the plotting option is enabled, the program generates graphical representations of the observed and expected distance profiles that can be saved to the user's computer.