The `MareyMap` package version 1.3

Aurélie Siberchicot, Clément Rezvoy, Delphine Charif,
Laurent Guéguen and Gabriel Marais

April 7, 2025

`MareyMap` is an `R` package to estimate local recombination rates along the genome. `MareyMap` relies on comparing the genetic and the physical maps of a given chromosome to estimate local recombination rates (given by the slope of the curve describing the relationship between both variables), a graphical method called the Marey map method introduced by A. Chakravarti in 1991[1]. `MareyMap` accepts Marey map data as input (genetic and physical positions of markers for a set of chromosomes of a species) and will return local recombination rate estimates.

`MareyMap` has many features and possible options (detailled in the present user guideline document) including:

- taking Marey map data from any species, including some Marey map data for a few species provided with the package

- estimating local recombination rates using different interpolation methods

- providing in an automatic way local recombination rates for any given gene (or set of genes) in the genome

If you use `MareyMap`, please cite:
Rezvoy C, Charif D, Guéguen L, Marais GAB. (2007) MareyMap: an R-based tool with graphical interface for estimating recombination rates. *Bioinformatics* 23(16):2188-9.
https://doi.org/10.1093/bioinformatics/btm315

If you use `MareyMapOnline`, please cite:
Siberchicot A, Bessy A, Guéguen L, Marais G (2017) MareyMap Online: A User-Friendly Web Application and Database Service for Estimating Recombination Rates Using Physical and Genetic Maps. *Genome Biology and Evolution* 9(10):2506-2509. https://doi.org/10.1093/gbe/evx178

Contents

---

[1]Chakravarti A. (1991) A graphical representation of genetic and physical maps: the Marey map. Genomics 11(1):219-22.

1 Installing and starting `MareyMap`

1.1 Initial installation

`MareyMap` is a package developed under the `R` software; sources are available on `http://cran.r-project.org/`. The `R` software must be installed in such a way that graphical interfaces can work. On Windows and Mac OS, this is automatically done when the `R` software is installed. On Linux, the two libraries *tcl* and *tk* must be installed, which is done by installing `R` with the *--with-tcltk* option.

When `R` is installed, the package `MareyMap` and its dependencies `tcltk`, `tkrplot` and `tools` must be installed, using the commands

```
install.packages(MareyMap)
install.packages(tcltk)
install.packages(tkrplot)
install.packages(tools)
```

on a `R` console.

1.2 Starting MareyMap

In a `R` console, first load the package:

```
library(MareyMap)
```

Then, open a graphical interface with the command:

```
startMareyMapGUI()
```

A new window, as shown in Figure 1 should open. If not, close your `R` console, re-load and re-start the package.
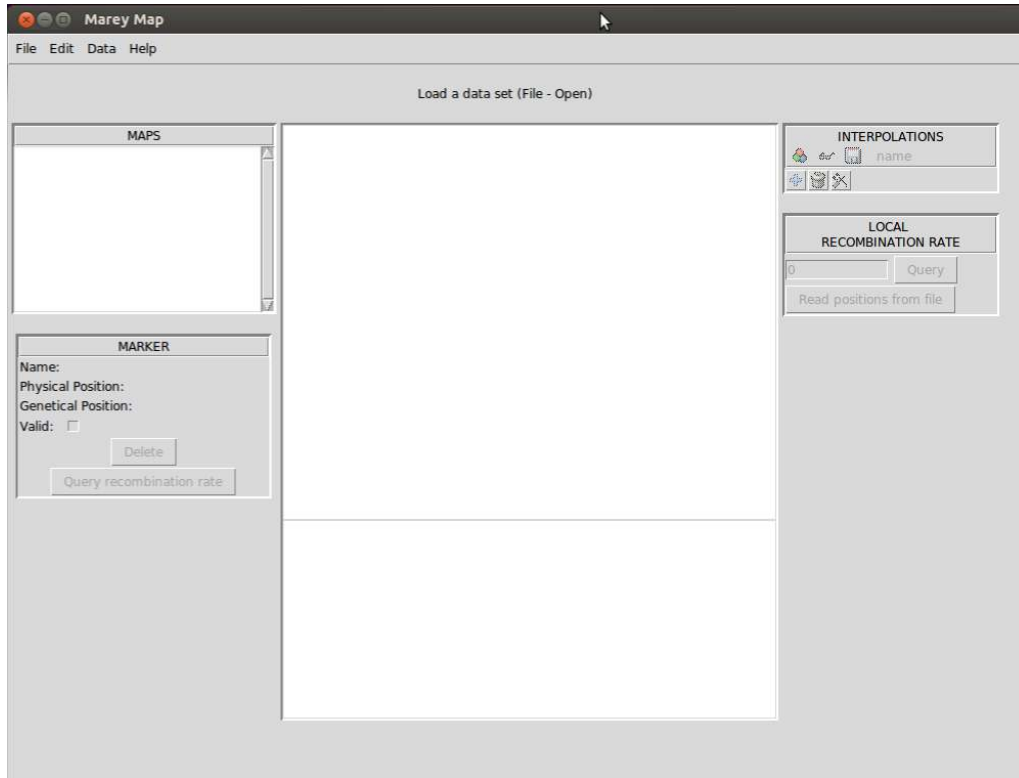
Figure 1: The `MareyMap` graphical interface

2 Data

2.1 Loading data

The user must either choose at least one dataset among those available in the `MareyMap` package or import his/her own dataset.

6 ready-to-use datasets are provided along with the package. This includes Marey maps for: *Arabidopsis thaliana*[2], *Caenorhabditis elegans*[3], *Drosophila melanogaster*[4] and *Homo sapiens*[5] (male, female and sex-averaged).

When the input dataset does not come from the package, the extension of the data file must be *txt*, *rda*, *Rda*, *rdata* or *Rdata*. When using a text file, the input data must be a data frame with the columns "set", "map", "mkr", "phys" and "gen". If missing, an additional column "vld" (indicating if the marker is valid or not) is added with `TRUE` value by default. The "set" colum corresponds to the organism, "map" corresponds to chromosomes, "mkr" corresponds to markers of genes, "phys" corresponds to physical position of markers, "gen" corresponds to genetic distances between each marker and "vld" corresponds to valid markers (this column is not mandatory).

Column names must be in the first row and values must be separated by a white space and each character

---

[2]Wright SI, Agrawal N, Bureau TE. Effects of recombination rate and gene density on transposable element distributions in Arabidopsis thaliana. Genome Res. 2003. 13:1897-1903.

[3]Wormbase Release WS160 `https://wormbase.org//`. see Rizzon C, Martin E, Marais G, Duret L, Segalat L, Biemont C. Patterns of selection against transposons inferred from the distribution of Tc1, Tc3 and Tc5 insertions in the mut-7 line of the nematode Caenorhabditis elegans. Genetics. 2003. 165:1127-1135.

[4]Marais G, Piganeau G. Hill-Robertson interference is a minor determinant of variations in codon bias across Drosophila melanogaster and Caenorhabditis elegans genomes. Mol Biol Evol. 2002. 19:1399-1406.

[5]Rutgers Combined Linkage-Physical Maps, version 2.0 (Build 35). Xiangyang Kong and Tara Matise 12/08/2004. see Kong et al. A high-resolution recombination map of the human genome. Nat Genet. 2002. 31:241-247.

string must be between double quotes (including column names), as in the example below. The rows which contains NA entries are removed.

```
"set" "map" "mkr" "phys" "gen" "vld"
"Arabidopsis thaliana" "Chromosome 1" "GST1" 663291 3.99 TRUE
"Arabidopsis thaliana" "Chromosome 1" "SGCSNP151" 1148355 3.35 TRUE
"Arabidopsis thaliana" "Chromosome 1" "AtEAT1" 1435872 3.87 TRUE
"Arabidopsis thaliana" "Chromosome 1" "ve002" 1521308 7.15 TRUE
"Arabidopsis thaliana" "Chromosome 1" "SGCSNP388" 1526933 7.66 TRUE
"Arabidopsis thaliana" "Chromosome 1" "SGCSNP170" 1642565 7.66 TRUE
"Arabidopsis thaliana" "Chromosome 1" "ve003" 2032443 7.76 TRUE
"Arabidopsis thaliana" "Chromosome 1" "SGCSNP308" 2664435 0.89 TRUE
```

Choose and open a dataset with the "**File**" and "**Open**" menus. The "**Data**" menu lists all the dataset opened. When one dataset is selected, the "**MAPS**" left frame is updated and shows the Marey maps (one for each chromosome) available in the dataset.

In the "**MAPS**" frame, the user selects one map (*i.e.* one chromosome) by clicking on it. The selected map is displayed (the physical positions on *x*-axis and the genetic distances on *y*-axis) in the central part of the interface. The "**INTERPOLATIONS**" right frame becomes active and the user can perform interpolations. Figure 2 shows the *Arabidopsis_thaliana Chromosome 1* Marey map as example.
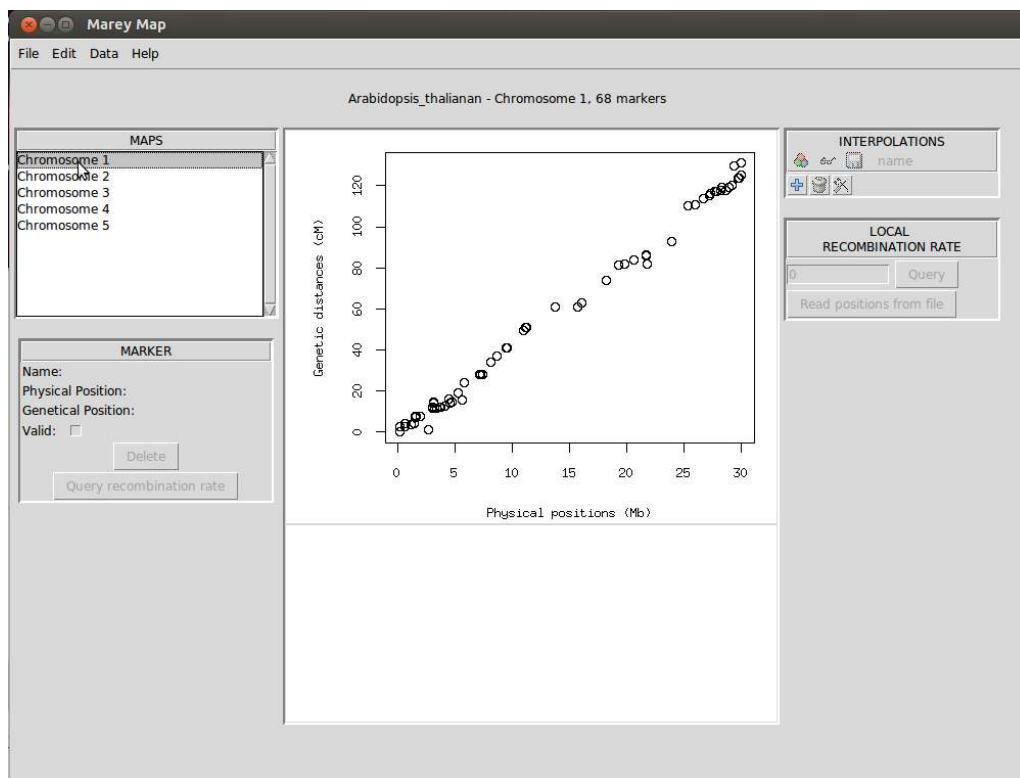

Figure 2: Displayed Marey map of a chromosome

2.2 Map cleaning

Physical or genetic maps occasionally include errors. Those will appear as outliers in a Marey map of a chromosome, disrupting the monotonically increasing behaviour expected from a Marey map function. Clicking on a marker on the map (the point becomes filled red) will display information about this marker in the "**MARKER**" left frame. If you un-select the "**Valid**" option (a red cross covers the point), this marker will not be included in the interpolations. This operation is reversible.

4

Deleting the marker is also possible. The marker will be removed from the rest of the analysis, but not from the raw data. The marker will be included again if the dataset is re-uploaded.

3 Interpolation methods

3.1 Selecting and running an interpolation method

3.1.1 Selecting a method

To run an interpolation method on a Marey map, click on the ✚ icon in the "**INTERPOLATIONS**" right frame and select an interpolation method from the list (see Figure 3).

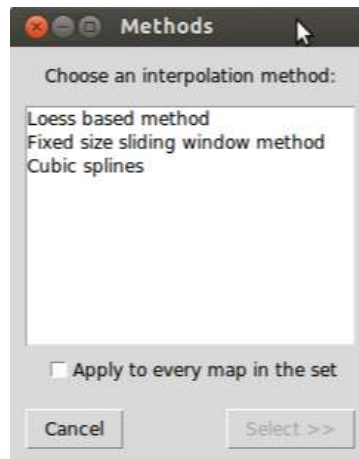After interpolation is done, the results are displayed in the central frame.

Figure 3: List of interpolation methods.

3.1.2 Changing and deleting interpolations

You can change the parameters of an interpolation by clicking on the ⚒ icon in the "**INTERPOLATIONS**" frame and delete an interpolation by clicking on the 🗑 icon. Interpolations can be shown in the central displaying frame, using the 👓 checkbox. The 💾 checkbox indicates whether the interpolation results should be included when saving the results into a text file.

3.1.3 Common parameters

Some parameters are common for all interpolation methods. By default, a name ("**Name**" parameter) is given to an interpolation (can be changed by the user), the interpolation results will be saved ("**Saved**" parameter) and displayed ("**Displayed**" parameter), and a line color ("**Line color**" parameter) is automatically chosen. These parameters can be changed at any time.

See 3.2 for specific parameters to each interpolation method.

3.1.4 Running a method to every map in a set

It is possible to run the same interpolation method (with the same parameters) on all the Marey maps (all the chromosomes) of a dataset. Just click on the "**Apply to every map in the set**" checkbox in the window that opens when a new interpolation is being set (see Figure 3). In this case, the interpolation will have the same name for all the Marey maps.

Similarly, changing or deleting an interpolation will affect all the maps if you use the "**Apply to every map in the set**" checkbox.

3.2 Available interpolation methods

The `MareyMap` package currently provides three interpolation methods: Loess, Sliding Windows and Cubic Splines.

3.2.1 Loess

Loess (or Lowess for LOcally WEighted Scatterplot Smoothing) estimates the recombination rates by locally adjusting a polynomial curve (1st or 2nd degree). The size of the window is defined as a percentage of the total number of markers and therefore can adapt to the variation of the density of markers across the map. Inside of a given window, each marker is attributed a weight depending on how far they are from the center of the window. The parameters β of the curves are those that minimize the mean squared deviation between the data and the curve:

$$Q = \sum_{i=1}^{n} \omega_i [y_i - f(x_i, \hat{\beta})]^2$$

where (x_i, y_i) are the observed data and ω_i is the weight of each marker calculated by:

$$\omega(u) = (1 - u^3)^3$$

with:

$$u = \frac{|x_0 - x_i|}{max_N(x_0)|x_0 - x_i|}$$

For this method, you can select the degree of the fitted curve ("**Degree**" parameter) and the size of the window ("**Span**" parameter). The span parameter is the percentage of the total number of points to take into account for computing the local polynomial at the vicinity of a marker. Span controls the degree of smoothing. The same span value is applied to all the maps, which may not be optimal if the error variance or the curvature of the underlying function f varies.

This method is based on the `R loess` function. For more information about this method, write `?loess` in a `R` console.
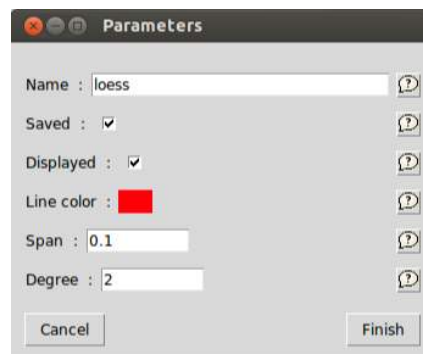Selecting this method will open a window as shown in Figure 4.

Figure 4: The Loess method

3.2.2 Sliding window

This method estimates the local recombination rates by carrying out linear regressions within a sliding window of a given physical size. You may adjust the size of the window ("**Size**" parameter), the distance between two successive windows ("**Shift**" parameter), as well the minimum number of marker per window for the interpolation to be carried out ("**Threshold**" parameter).
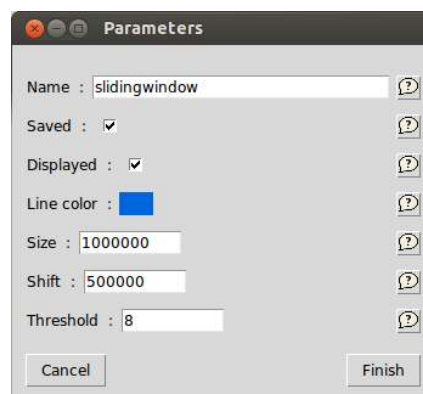Selecting this method will open a window as shown in Figure 5.

Figure 5: The sliding window method

3.2.3 Cubic splines

A cubic smoothing spline behaves approximately like a kernel smoother, but it corresponds to the function $\hat{f}$ that minimizes the penalized residual sum of squares given by:

$$PRSS = \sum_{i=1}^{n}(y_i - f(x_i))^2 + \lambda \int (f''(t))^2 dt$$

λ is the smoothing parameter, corresponding to the span in loess. A different λ can be specified using the "**Spar**" parameter.

The "**Degree of freedom**" parameter controls the amount of smoothing and corresponds to the trace of the smoothing matrix. It is also estimated automatically using spar or by cross-validation.

These two parameters will be estimated automatically under R either by locally or generalized cross-validation.

The generalized cross-validation is performed using this function:

$$CV(\lambda) = \frac{1}{n} \sum_{i=1}^{n} (y_i^* - \hat{f}_\lambda^{-i}(x_i))$$

Here $\hat{f}_\lambda^{-i}(x_i)$ is the leave-one-out smooth at x_i, that is constructed using all the data except for (x_i, y_i) and then the resulting least squares line is evaluated at x_i. CV is calculated for different values of λ and the λ that minimizes this criterion is chosen. The "**Generalized cross-validation**" method should be used when there are several markers with identical physical position.

In the graphical interface, you must fill the parameter chosen in the "**Type**" list.

This method is directly based on the function `smooth.spline` of R. To get more information about this method you can type `?smooth.spline` in a R console.
Selecting this method will open a window as shown in Figure 6.

Figure 6: The cubic splines method

4 Queries

Once an interpolation method has been run on a map, you can make queries about local recombination rates using the "**LOCAL RECOMBINATION RATE**" right frame. There are four different ways of using this frame.

1. You may want to know the recombination rate at a given physical position on the currently displayed map. The position must be specified in base pair (ex. 31564623) but can also be expressed using Mb or Kb (ex. 31Mb, 564Kb or even 31Mb+564Kb+623). The local recombination rate from each interpolation available will be provided for this position. This is done when the "**Query**" button is pressed, and shown in the updated "**LOCAL RECOMBINATION RATE**" window. A vertical red line is then displayed on the two central graphics, at the physical position of interest.

2. You may want to know the recombination rate at several positions on the currently displayed map. Just list them separating them by ":" (ex. 31Mb:12287456:44Kb+564). When clicking on the "**Query**" button, results will be displayed in a separate window (see Figure 7) and can be saved into a text file. The results will include one column per interpolation available for the displayed map.
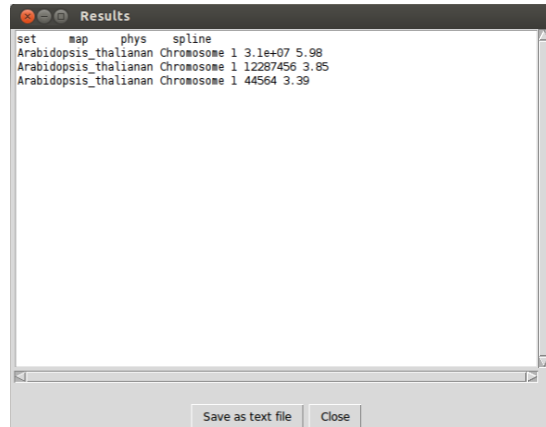

Figure 7: Example of output for several queries.

3. You may want to know the recombination rate at many positions (for instance all the genes of a genome). This can be done by up-loading a text file including all the positions. To do this, you can

- enter the path of the above mentioned file and click on the "**Query**" button,

- or click on "**Read positions from file**" and select the file using the file selector dialog window.

The input file must be a text file (*txt* extension) containing at least a "map" column and a "phys" column indicating respectively the map and the physical position of each gene. An example file *test_query.txt* is provided along with the package. This file may also include a "set" column if there are genes from several genomes for instance (if this column is not present all the genes are considered from the same genome, *i.e.* the same query). Any other column will be ignored by the program but will be kept in the output file.

4. It is also possible to know the recombination rate at a position in an interactive way. When one marker is selected (by clicking) on the displayed map (in the top central frame), some details are updated in the "**MARKERS**" left frame. You will be able to click on the "**Query recombination rate**" button. As before, results are shown in the updated "**LOCAL RECOMBINATION RATE**" frame and a vertical red line is displayed on the two central graphics, at the physical position of interest (see Figure 8).
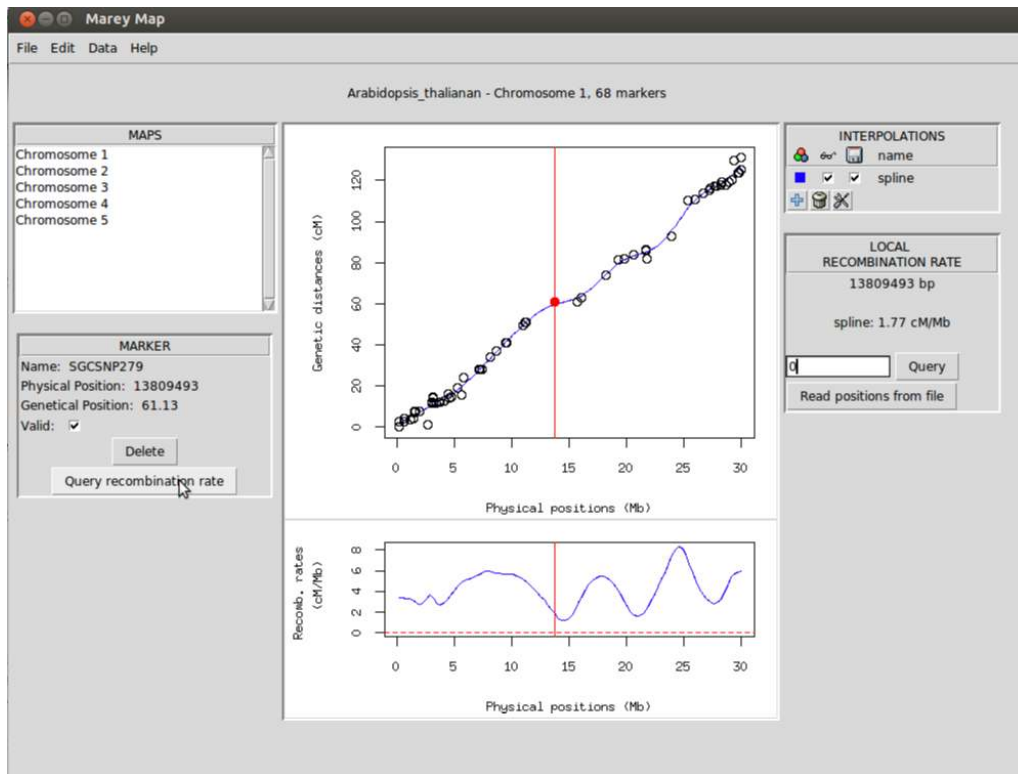
Figure 8: Example of an interactive query.

5 Saving your results

5.1 Saving data

Maps can be saved to `R` data files (*rda*, *Rda*, *rdata* or *Rdata*) or to text files (*txt*).
All interpolation methods created (applied on either a map or a set of maps) in the current `R` console are saved in the file.

If the file is a text file, it will include a line per marker with columns "set" (for the dataset name), "map" (for the map name, ie. the chromosome name), "phys" (for the physical position of the marker), "gen" (for the genetic position of the marker) and "vld" (indicating if the marker is valid or not). If interpolation methods are included (those for which the ⊟ checkbox is checked), the file also contains a column per interpolation (the column name is the interpolation method name) with the local recombination rate computed for each marker. Functions used to build the interpolations are also saved as comments at the beginning of the text file.

5.2 Exporting pictures

Maps can also be graphically exported in *jpeg*, *png*, *pdf* or *eps* formats. Only the currently diplayed map is exported, with only the interpolations which are checked as "**Displayed**". You can choose to export either the Marey map (on the top), or the recombination rate display (on the bottom), or both.

5.3 Loading previous analyses

You may want to resume work on a dataset.
If the work was saved in a *txt* format, you can re-run interpolation methods using the `R` commands previously used, which can be found at the top of the *txt* file.
If it has been saved in a *rda* format, the "**Open**" command loads all previously saved interpolations.