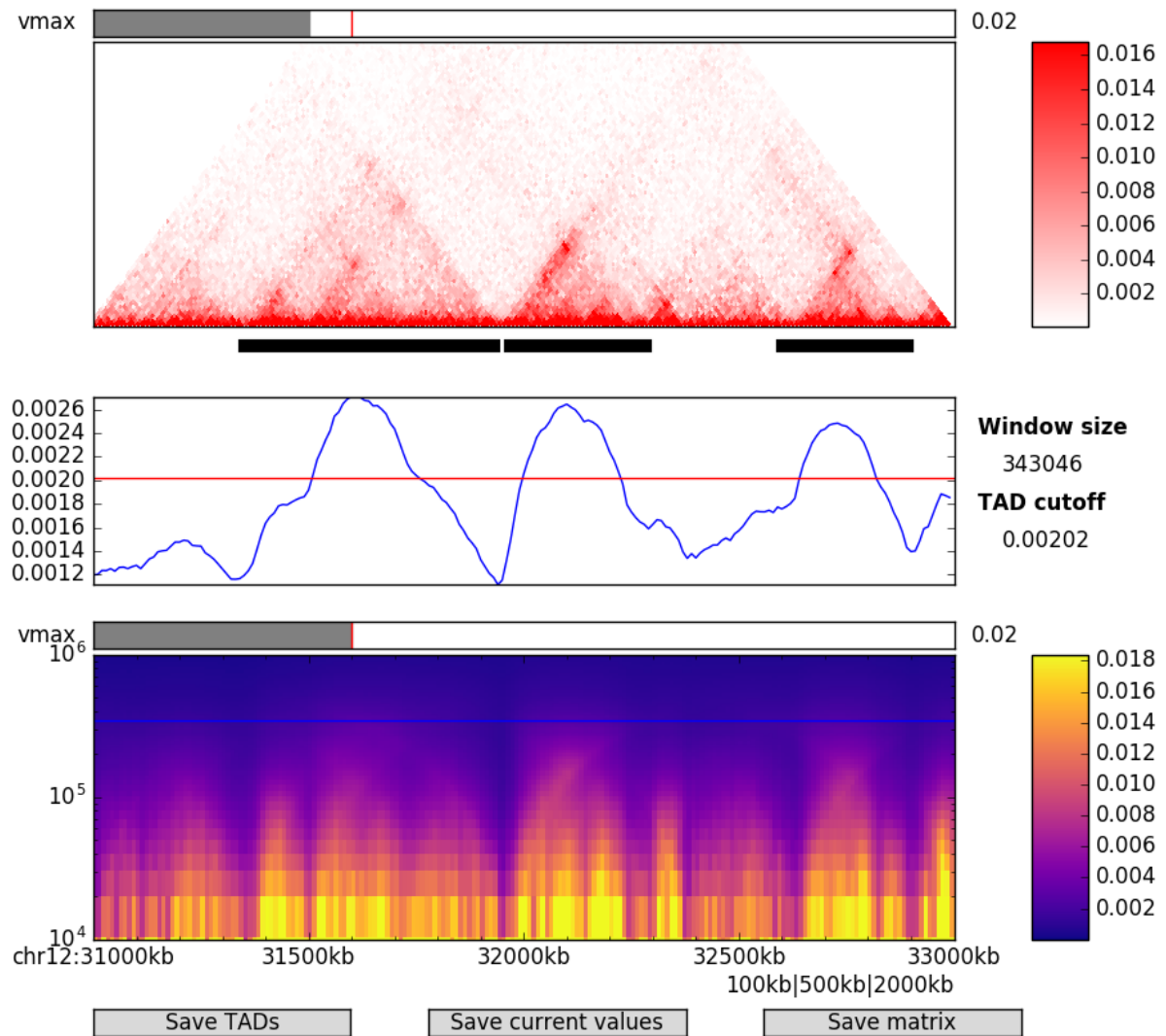


TADtool

TADtool is an interactive tool for the identification of meaningful parameters in topologically-associating domains (TADs) algorithms for Hi-C data.



- [TADtool](#)
 - [Quick start](#)
 - [Installation](#)
 - [Usage](#)
 - [plot](#)
 - [Launching from the command line](#)
 - [Interactive plotting window](#)
 - [Hi-C plot](#)
 - [Index plot](#)
 - [Heatmap](#)
 - [Export buttons](#)
 - [tads](#)

Quick start

Installation:

```
pip install tadtool
```

Run sample data from [GitHub repo](#):

```
tadtool plot examples/chr12_20-35Mb.matrix.txt examples/chr12_20-35Mb_regions.bed chr12:31000000-33000000
```

This should open the interactive plotting window (see above). Start exploring by clicking in plots - the effects should be self-explanatory.

Installation

You can install TADtool from the command line using PyPI

```
pip install tadtool
```

or download the source from our [GitHub repo](#) and install manually

```
python setup.py install
```

This should install both the Python 2 package and a command-line executable called **tadtool**.

Test the installation running

```
tadtool -h
```

and you should see a brief help message.

Usage

TADtool has two basic commands: **plot**, which invokes the plotting window, and **tads**, which can be used to call TADs with predefined parameters.

plot

Launching from the command line

Here is the help output from **tadtool plot -h**:

```
usage: tadtool plot [-h] [-w WINDOW_SIZES [WINDOW_SIZES ...]] [-a ALGORITHM]
                  [-m MAX_DIST] [-d DATA]
                  matrix regions plotting_region

Main interactive TADtool plotting window

positional arguments:
  matrix                Square Hi-C Matrix as tab-delimited or .npy file
                        (created with numpy.save)
  regions               BED3 file (no header) with regions corresponding to
                        the number of rows in the provided matrix.
  plotting_region       Region of the Hi-C matrix to display in plot. Format:
                        <chromosome>:<start>-<end>, e.g.
                        chr12:31000000-33000000

optional arguments:
  -h, --help            show this help message and exit
  -w WINDOW_SIZES [WINDOW_SIZES ...], --window-sizes WINDOW_SIZES [WINDOW_SIZES ...]
                        Window sizes in base pairs used for TAD calculation.
                        You can pass (1) a filename with whitespace-delimited
                        window sizes, (2) three integers denoting start, stop,
                        and step size to generate a range of window sizes, or
                        (3) more than three integers to define window sizes
                        directly. If left at default, window sizes will be
                        logarithmically spaced between 10**4 and 10**6, or
                        10**6.5 for the insulation and directionality index,
                        respectively.
  -a ALGORITHM, --algorithm ALGORITHM
                        TAD-calling algorithm. Options: insulation,
                        directionality. Default: insulation.
  -m MAX_DIST, --max-distance MAX_DIST
                        Maximum distance in base-pairs away from the diagonal
                        to be shown in Hi-C plot. Default: 3000000
  -d DATA, --data DATA
                        Matrix with index data. Rows correspond to window
                        sizes, columns to Hi-C matrix bins. If provided,
                        suppresses inbuilt index calculation.
```

plot takes three mandatory (positional) arguments:

- A (square) Hi-C matrix file, which can be either simply a tab-delimited text file, or a numpy **.npy** file. The latter is for the tech-savvy and can be created with the **numpy.save** method in Python. It is possible to load large matrices at high resolution, but bear in

mind that a large matrix will consume more memory and may slow down TAD calculations. We recommend using intra-chromosomal matrices of a single chromosome for the best experience.

- A BED3 file with region information for the Hi-C matrix, i.e. a tab-delimited file where each row contains chromosome name, start, and end coordinates (inclusive) of the region. This file must not contain any headers.
- A region selector string to inform TADtool about the region it is supposed to plot. The string must be of the format **<chromosome>:<start>-<end>**, where **<end>** is inclusive. Examples: **chr5:34000000-37000000**, **chrIX:300000-1000000**

When called with only these three arguments

```
tadtool plot /path/to/matrix.txt /path/to/regions.bed chr1:1000000-3000000
```

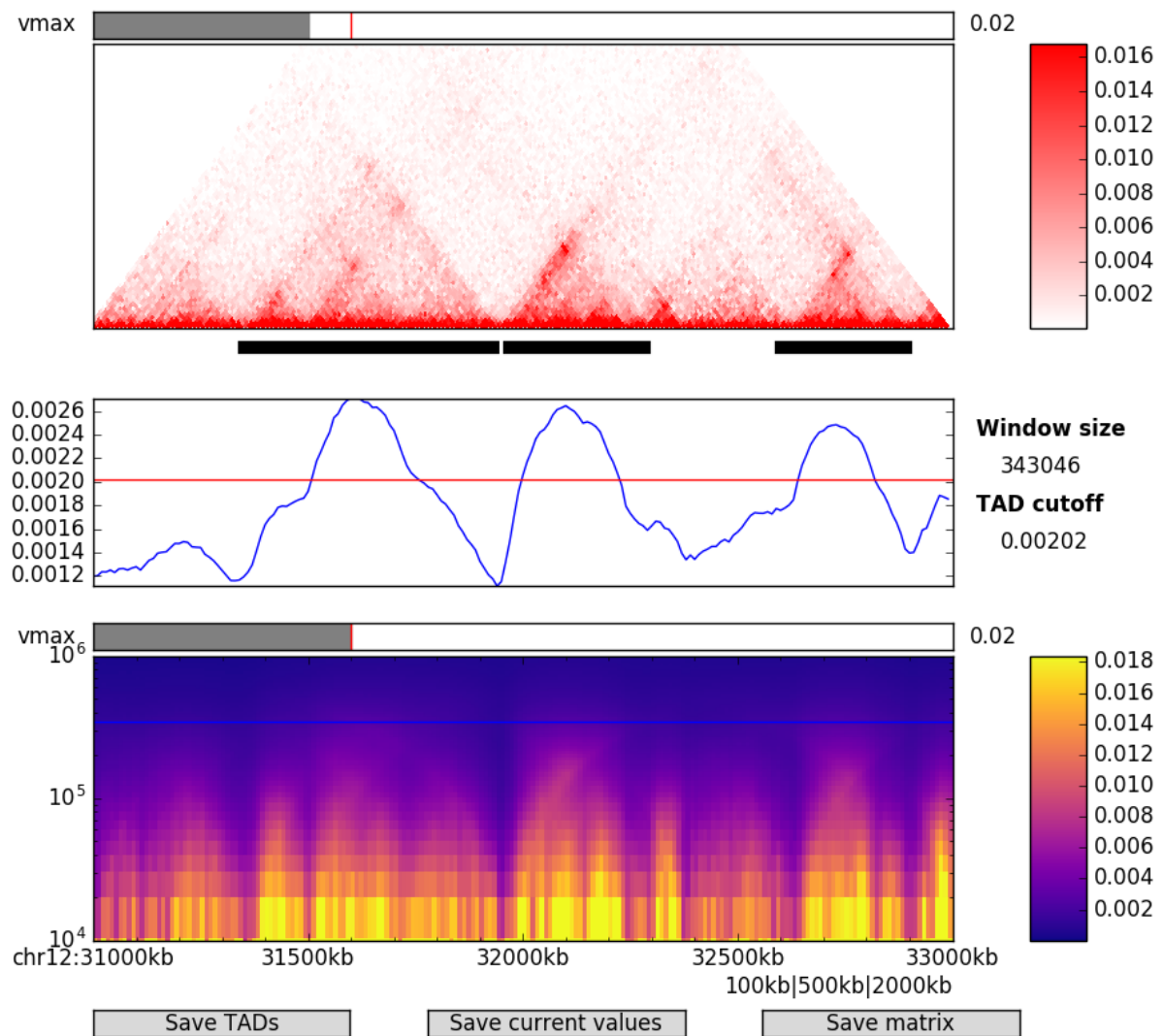
TADtool will calculate the insulation index for a range of automatically selected window sizes.

Optional arguments give you more control about the data that is generated:

- **-w** lets you define a custom range of window sizes, either as a file (**txt** or **npz**), a range, or specific values.
- **-a** lets you select the directionality index instead of the insulation index.
- **-m** lets you specify a distance away from the Hi-C matrix diagonal that should be omitted from the plot.
- **-d** lets you bypass the TAD-calling calculation completely by providing a file with precomputed index data.

Interactive plotting window

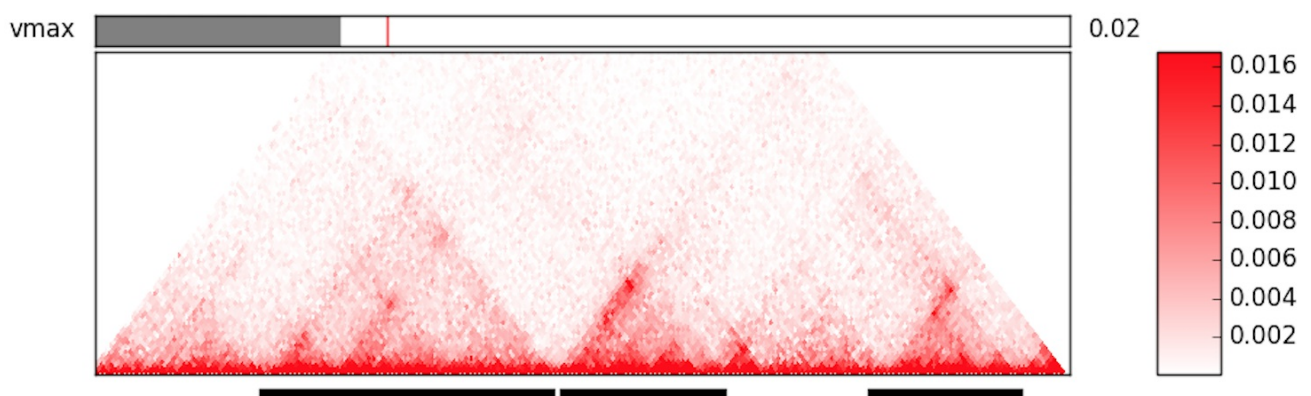
After starting TADtool with the **plot** option as instructed above, the interactive plotting window will open.



The window consists of four parts: the Hi-C plot with bars indicating the currently called TADs, a line plot with the insulation or directionality index at the current window size, a heatmap with the insulation or directionality index for all specified window sizes, and a row of buttons to save the currently displayed data.

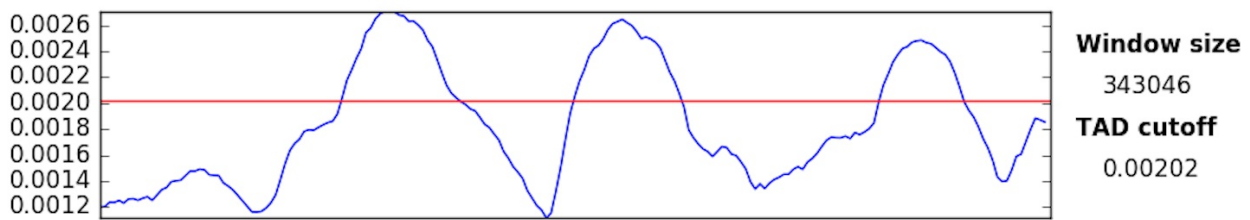
You can use the toolbar at the very bottom of the window to zoom and pan each plot (they will be synchronized) or to save the figure window in its current state.

Hi-C plot



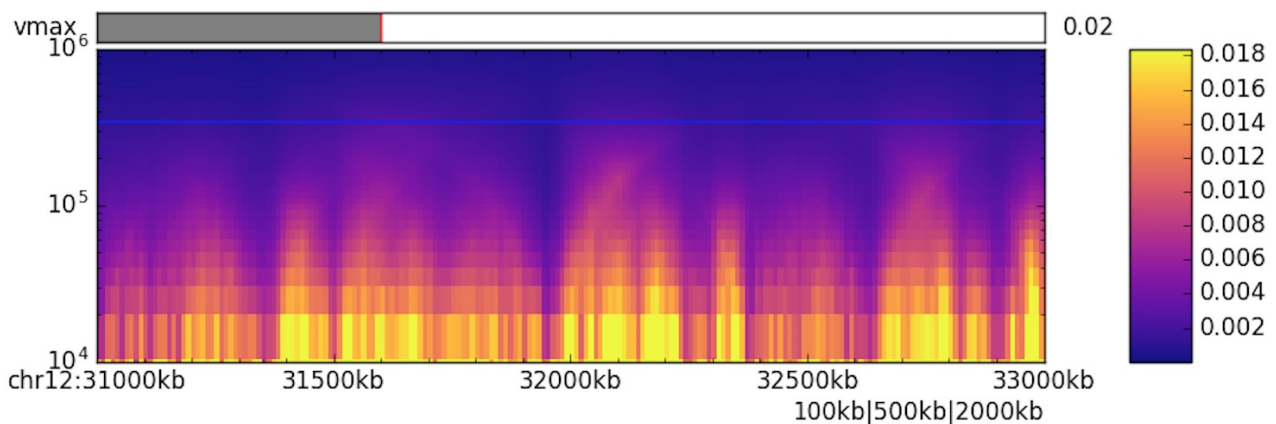
This triangular plot shows a Hi-C map in the selected region from the command line. Use the slider on the top to adjust the color intensity as appropriate. On the bottom you will find bars that indicate the currently called TADs as calculated with the selected parameters.

Index plot



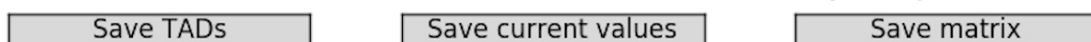
This plot shows the insulation or directionality index for each region in the Hi-C matrix at the currently selected window size. Red lines indicate the cutoff that is used to call TADs in this plot. Clicking within the plotting area moves the cutoff and simultaneously recalculates the TADs with the new cutoff.

Heatmap



The heatmap shows all insulation/directionality indexes for each specified window size simultaneously in a condensed form. Every row corresponds to one window size. A blue bar indicates the currently chosen window size. Clicking within the plotting area changes the current window size and updates the index plot and TAD indicators.

Export buttons



At the bottom of the plotting window you will find three buttons that allow you to export TADs, index data, and the index matrix, respectively.

tads

This command allows the calculation of TADs directly using one of the provided algorithms. This may be useful for quick TAD calculations when parameters are already known, or to fine-tune parameters estimated from the main TADtool plot. Here is the help output from

tadtool tads -h:

```
usage: tadtool tads [-h] [-a ALGORITHM]
                    matrix regions window_size cutoff [output]

Call TADs with pre-defined parameters

positional arguments:
  matrix      Square Hi-C Matrix as tab-delimited or .npy file
              (Created with numpy.save)
  regions     BED3 file (no header) with regions corresponding to
              the number of rows in the provided matrix.
  window_size Window size in base pairs
  cutoff      Cutoff for TAD-calling algorithm at given window size.
  output      Optional output file to save TADs.

optional arguments:
  -h, --help  show this help message and exit
```

```
-a ALGORITHM, --algorithm ALGORITHM
                        TAD-calling algorithm. Options: insulation,
                        directionality. Default: insulation.
```

As **plot**, **tads** needs a Hi-C matrix and a regions BED file as input. In addition, it requires a window size in base pairs and a cutoff (floating point). Optionally, the user can specify an output file to save TAd regions, otherwise TADs will be written to the command line.