

User guide of MSTD

MSTD is a generic and efficient method to identify multi-scale topological domains (MSTD) from symmetric Hi-C and other high resolution asymmetric promoter capture Hi-C datasets.

Program Download

Download page – [Get the latest version of MSTD](#)

Using MSTD

Installation:

1. It is recommended to install MSTD in python3.6 or above environment.
2. Installing requiring packages, including numpy, pandas, seaborn and colormap using pip or conda.
3. Extracting the downloaded installation package to the specified directory and run the command: `python setup.py install`

Getting started:

MSTD currently contains two versions of the detection of multi-scale topological domains. Version 1 is to detect domains whose boundaries may exist overlapping, while version 2 is to detect domain without overlapping. The version 2 is more generic for most analyses. Both versions of MSTD are included in the modules “MSTD_v1” and “MSTD_v2”, respectively. Both versions contain the following functions.

Function 1:

`MSTD(Matrix_file,Output_file,MDHD=5,symmetry=1>window=5,visualization=0)`

Description: Identifying multi-scale topological domains from Hi-C maps or promoter capture Hi-C maps.

Parameters:

Matrix_file: Input file address, the format of the file is N*N matrix file without row and column names for Hi-C maps and the format of the file is N*M matrix file without row and column names for promoter capture Hi-C maps.

Output_file: Output file address, each line in the file is triple containing boundaries and centers of detected domains.

MDHD: integer, the threshold for the minimum distance of the elements that have higher density than the element k.

symmetry: 1/0, 1 represents the detecting of TADs and 0 represents the detecting PADs

windows: The size of window used to compute the average contact frequency of one element. Defaults=5. According to the different resolution of the data, it can be adjusted between 5-10, which has little effect on the results.

visualization: if visualization=1, Visualization of results can be showed.

Function 2:

```
MSTD2(Matrix_file,Output_file,MDHD=5,symmetry=1>window=5,visualization=0)
```

Description: Identifying multi-scale topological domains from Hi-C maps or promoter capture Hi-C maps.

Parameters:

Matrix_file: Input file address, the format of the file is triple for promoter capture Hi-C maps, the format of the file is N*N matrix file without row and column names for Hi-C maps.

Output_file: Output file address, each line in the file is triple containing boundaries and centers of detected domains.

MDHD: integer, the threshold for the minimum distance of the elements that have higher density than the element k.

symmetry: 1/0, 1 represents the detecting of TADs and 0 represents the detecting PADs

windows: The size of window used to compute the average contact frequency of one element. Defaults=5. According to the different resolution of the data, it can be adjusted between 5-10, which has little effect on the results.

visualization: if visualization=1, Visualization of results can be showed.

Function 2:

```
example(symmetry=1)
```

Description: showing the examples for Identifying multi-scale topological domains from Hi-C maps or promoter capture Hi-C maps, respectively.

Parameters:

symmetry: 1/0, 1 represents the detecting of TADs and 0 represents the detecting PADs

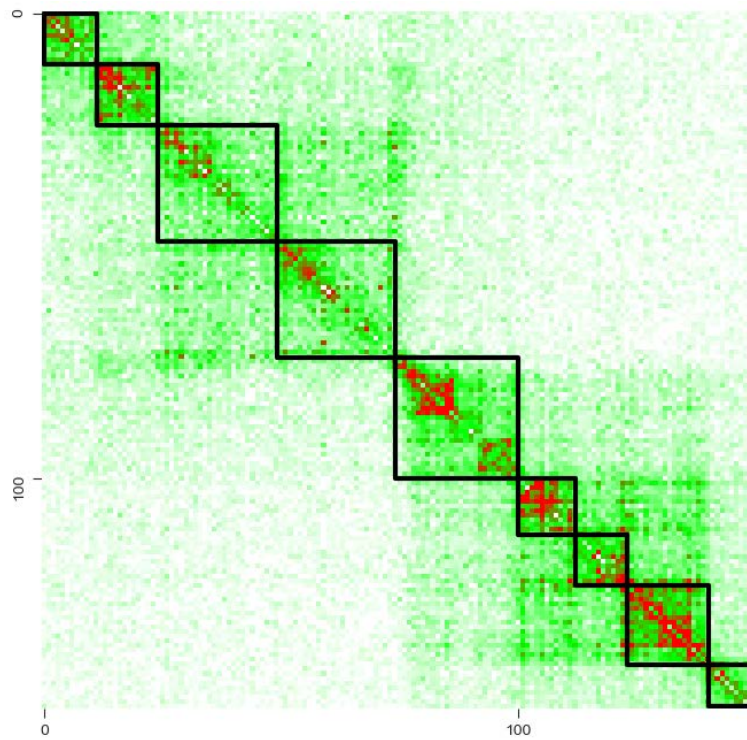
Examples:

```
import os
import MSTDlib.MSTD_v2 as MSTD_v2
os.chdir("E:\\Users\\lyusen\\Project\\Project3\\Python code\\MSTD-master")
```

```
#1. run the example in the program
```

```
MSTD_v2.example(symmetry=1)
```

```
# Visualizing the result 1
```



#2. run example using symmetric Hi-C dataset in the MSTD packages

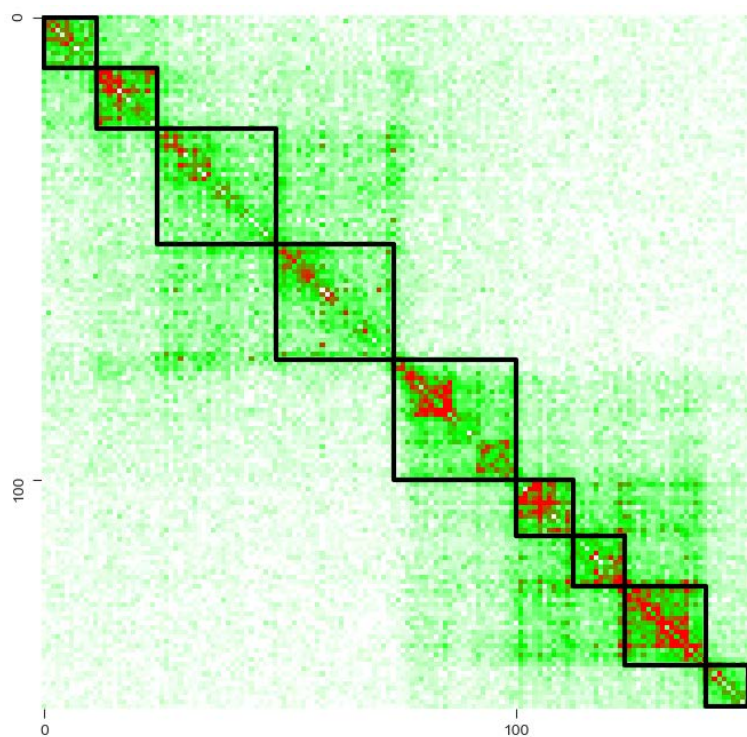
Dir='./src/MSTDlib'

Matrix_file=Dir+'\\data\\cortex_chr6_2350-2500_HiC'

Output_file=Dir+'\\data\\cortex_chr6_output'

MSTD_v2.MSTD(Matrix_file,Output_file,MDHD=10,symmetry=1>window=10,visualization=1)

Visualizing the result 2



#3. run example using asymmetric promoter capture Hi-C dataset in the MSTD packages

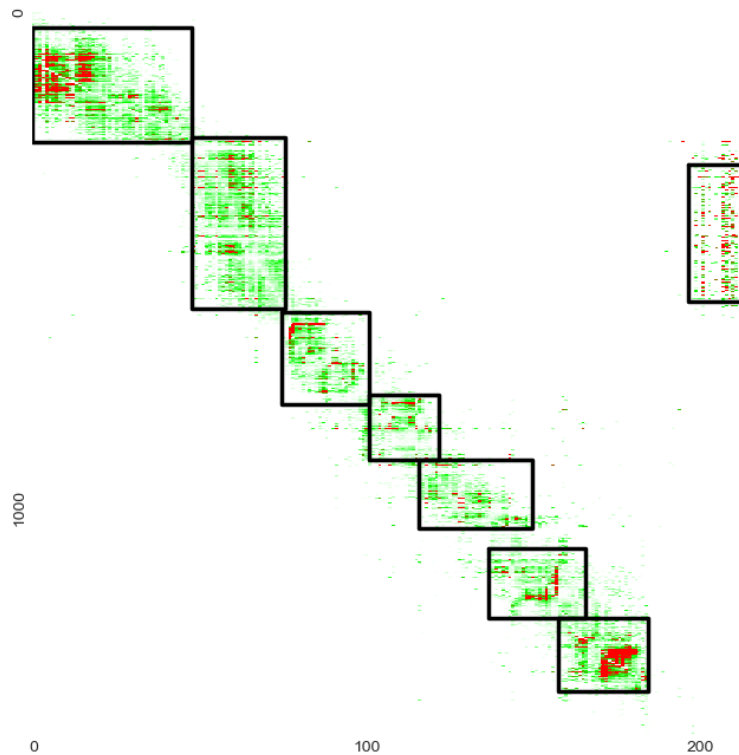
Dir='./src/MSTDlib'

Matrix_file=Dir+'\\data\\nB_chr19_480-700_CHiC'

Output_file=Dir+'\\data\\nB_chr19_480-700_CHiC_output'

MSTD_v2.MSTD(Matrix_file,Output_file,MDHD=100,symmetry=2>window=5,visualization=1)

Visualizing the result 3



Contact us:

ccyeyusen@sina.com