**GenomeFlow User Manual**

Contents

[1. Convert mapped Hi-C reads to hic format file 4](#_Toc504661693)

[a) Purpose 4](#_Toc504661694)

[b) Input file format 4](#_Toc504661695)

[c) Output 6](#_Toc504661696)

[d) Running 6](#_Toc504661697)

[2. Extract contact matrices from a hic format 8](#_Toc504661698)

[a) Purpose 8](#_Toc504661699)

[b) Input 8](#_Toc504661700)

[c) Output 8](#_Toc504661701)

[d) Running 8](#_Toc504661702)

[3. Normalize HiC contact matrices 9](#_Toc504661703)

[a) Purpose 9](#_Toc504661704)

[b) Input 9](#_Toc504661705)

[c) Output 9](#_Toc504661706)

[d) Running: 9](#_Toc504661707)

[4. Visualizing Dataset in 2D format 10](#_Toc504661708)

[a) Purpose 10](#_Toc504661709)

[b) Input 10](#_Toc504661710)

[c) Output 10](#_Toc504661711)

[d) Running 10](#_Toc504661712)

[e) Display Controls 11](#_Toc504661713)

[f) TAD Annotation 14](#_Toc504661714)

[g) Demonstration 15](#_Toc504661715)

[5. Identify TAD 17](#_Toc504661716)

[a) Purpose 17](#_Toc504661717)

[b) Input 17](#_Toc504661718)

[c) Output 17](#_Toc504661719)

[d) Running 17](#_Toc504661720)

[6. Check TAD consistency between two TADs from different methods 19](#_Toc504661721)

[a) Purpose 19](#_Toc504661722)

[b) Input 19](#_Toc504661723)

[c) Output 19](#_Toc504661724)

[d) Running 19](#_Toc504661725)

[7. 3D model reconstruction by LorDG 20](#_Toc504661726)

[a) Purpose 20](#_Toc504661727)

[b) Input 20](#_Toc504661728)

[c) Output 21](#_Toc504661729)

[d) Running 21](#_Toc504661730)

[8. 3D model reconstruction by 3DMax 22](#_Toc504661731)

[a) Purpose 23](#_Toc504661732)

[b) Input 23](#_Toc504661733)

[c) Output 23](#_Toc504661734)

[d) Running 23](#_Toc504661735)

[9. Chromatin loop identification 25](#_Toc504661736)

[a) Purpose 25](#_Toc504661737)

[b) Input 25](#_Toc504661738)

[c) Output 25](#_Toc504661739)

[d) Running 25](#_Toc504661740)

[10. Model annotation 26](#_Toc504661741)

[a) Purpose 26](#_Toc504661742)

[b) Input 26](#_Toc504661743)

[c) Output 26](#_Toc504661744)

[d) Running 26](#_Toc504661745)

[11. Gene expression data visualization (a special case of model annotation) 27](#_Toc504661746)

[a) Purpose 27](#_Toc504661747)

[b) Input 27](#_Toc504661748)

[c) Output 28](#_Toc504661749)

[d) Running 28](#_Toc504661750)

[12. Comparing 2 models 29](#_Toc504661751)

[a) Purpose 29](#_Toc504661752)

[b) Input 29](#_Toc504661753)

[c) Output 29](#_Toc504661754)

[d) Running 29](#_Toc504661755)

**List of Figures**

[Figure 1: Convert to HiC function 7](#_Toc504661604)

[Figure 2: Extract Contact Matrices from a hic file 8](#_Toc504661605)

[Figure 3: Normalize HiC contact matrices 10](#_Toc504661606)

[Figure 4: Visualize Dataset in 2D Format 11](#_Toc504661607)

[Figure 5: Demonstration of TAD Annotation on 2D Heatmap 17](#_Toc504661608)

[Figure 6: Identifying TADs on a contact matrix 18](#_Toc504661609)

[Figure 7: Comparing two TADs for a consistency check 20](#_Toc504661610)

[Figure 8: 3D Model reconstruction by LorDG 21](#_Toc504661611)

[Figure 9: 3D Model reconstruction by 3DMax 23](#_Toc504661612)

[Figure 9: Chromatin loops 25](#_Toc504661613)

[Figure 11: Coordinate of a point in the model 27](#_Toc504661614)

[Figure 12: Gene expression visualization demonstration 29](#_Toc504661615)

[Figure 13: Comparing two constructed models 30](#_Toc504661616)

1. **Convert mapped Hi-C reads to hic format file**
2. **Purpose**

To create a binary hic format file containing contact matrices at different resolutions and normalized by different methods from a text file describing mapped Hi-C reads

1. **Input** **file format**

Five formats are acceptable: short format, short format with score, medium format, long format and 4DN DCIC format. A sample file is executable/sample\_data/GSM1551688\_HIC143\_merged\_nodups.zip (unzip it before use)

* + **Short format**

A whitespace separated file that contains, on each line

<str1> <chr1> <pos1> <frag1> <str2> <chr2> <pos2> <frag2>

* str = strand (0 for forward, anything else for reverse)
* chr = chromosome (must be a chromosome in the genome)
* pos = position
* frag = restriction site fragment

If not using the restriction site file option, frag will be ignored, but please see above note on dummy values. readname and strand are also not currently stored within *.hic* files.

* + **Short with score format**

This format is useful for reading in already processed files, e.g. those that have been already binned and/or normalized; this format can be easily used in conjunction with the -r flag to create a *.hic*file that contains a single resolution.

A whitespace separated file that contains, on each line

<str1> <chr1> <pos1> <frag1> <str2> <chr2> <pos2> <frag2> <score>

* str = strand (0 for forward, anything else for reverse)
* chr = chromosome (must be a chromosome in the genome)
* pos = position
* frag = restriction site fragment
* score = the score imputed to this read

If not using the restriction site file option, frag will be ignored, but please see above note on dummy values. readname and strand are also not currently stored within *.hic* files.

* + **Medium format**

A whitespace separated file that contains, on each line

<readname> <str1> <chr1> <pos1> <frag1> <str2> <chr2> <pos2> <frag2> <mapq1> <mapq2>

* str = strand (0 for forward, anything else for reverse)
* chr = chromosome (must be a chromosome in the genome)
* pos = position
* frag = restriction site fragment
* mapq = mapping quality score

If not using the restriction site file option, frag will be ignored, but please see above note on dummy values. If not using mapping quality filter, mapq will be ignored. readname and strand are also not currently stored within .hic files.

* + **Long format**

The long format is used by [Juicer](https://github.com/theaidenlab/juicer) and takes in directly the *merged\_nodups.txt* file.

A whitespace separated file that contains, on each line

<str1> <chr1> <pos1> <frag1> <str2> <chr2> <pos2> <frag2> <mapq1> <cigar1> <sequence1> <mapq2> <cigar2> <sequence2> <readname1> <readname2>

* str = strand (0 for forward, anything else for reverse)
* chr = chromosome (must be a chromosome in the genome)
* pos = position
* frag = restriction site fragment
* mapq = mapping quality score
* cigar = cigar string as reported by aligner
* sequence = DNA sequence

If not using the restriction site file option, frag will be ignored, but please see above note on dummy values. If not using mapping quality filter, mapq will be ignored. readname, strand, cigar, and sequence are also not currently stored within *.hic* files.

* + **4DN DCIC format**

A file that follows the 4DN DCIC format specification ([the 4DN DCIC format specification](https://github.com/4dn-dcic/pairix/blob/master/pairs_format_specification.md)).  
See the link for more information. Briefly, there should be a header with the first seven columns reserved:

## pairs format v1.0

#columns: readID chr1 position1 chr2 position2 strand1 strand2

If the columns line contains (in any field after field 7) both frag1 and frag2, those will also be read in; otherwise Pre will set frag1=0 and frag2=1, so that no reads are discarded. Other fields are ignored.

1. **Output**

A binary .hic file containing contact matrices

1. **Running**

Access the function from the menu toolbar: 2D-Functions/Convert to HiC

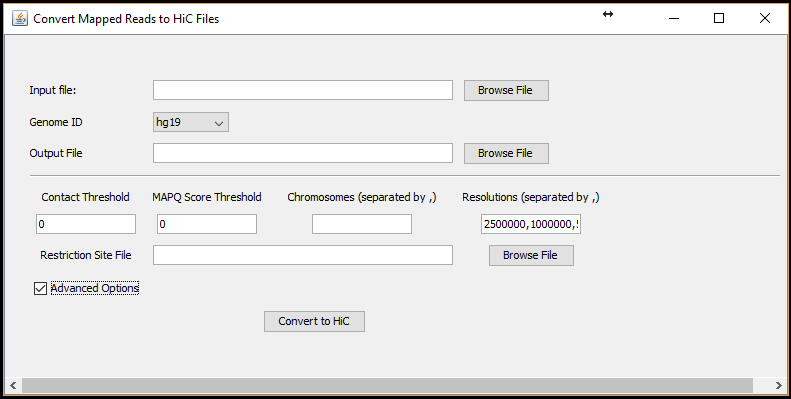


Figure 1: Convert to HiC function

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Default** |
| Input file | A text file describes mapped Hi-C reads (format described above) | NA |
| Genome ID | Version genome of Hi-C data | hg19 |
| Output File | A name of the output hic format file | NA |
| Contact Threshold | Number of interaction threshold for contacts to be used in creating contact matrices. | 0 |
| MAPQ Score Threshold | Mapping quality score threshold for reads to be considered in creating contact matrices. | 0 |
| Chromosomes | Chromosomes for which their contact matrices need to be created. When left blank, all chromosomes will be considered. Chromosomes must be separated by a comma (,). | All (when left blank) |
| Resolutions | List of resolutions of contact matrices to be created. Resolutions are separated by a comma (,) | 2500000, 1000000, 500000, 250000, 100000, 50000, 25000,10000,5000 |
| Restriction Site File | Each line starts with a chromosome number followed by positions of restriction sites on that chromosome, in numeric order, and ending with the size of the chromosome. When provided, 8 additional fragment-delimited resolutions are added: 500f, 250f, 100f, 50f, 20f, 5f, 2f, 1f | blank |

1. **Extract contact matrices from a hic format**
   1. **Purpose**

To extract a contact matrix from a hic format into a sparse matrix format in a text file

* 1. **Input**

A local path to a hic format or an online link to a hic format. A link to a hic file: <https://www.encodeproject.org/files/ENCFF219YOB/@@download/ENCFF219YOB.hic>

* 1. **Output**

A contact matrix in sparse matrix format (each line represents a contact by three numbers separated by whitespaces: position1 postion2 interaction\_frequency)

* 1. **Running**

Access the function from the menu toolbar: 2D-Functions/Extract HiC

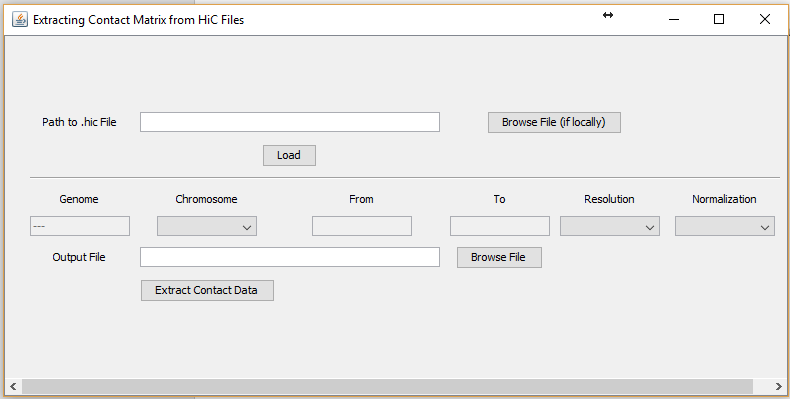


Figure 2: Extract Contact Matrices from a hic file

|  |  |  |
| --- | --- | --- |
| **Field / Button** | **Description** | **Default** |
| Path to .hic File | An online link or local path to a hic format file | NA |
| Load | Click this button to fetch information from the header of the hic file. | NA |
| Genome | Genome version of the hic file | NA |
| Chromosomes | List of resolutions of contact matrices in the hic file | NA |
| From | Start of a fragment (to extract its contact matrix). When From and To are left blank, the whole chromosome is considered. | Blank |
| To | End of a fragment (to extract its contact matrix). When From and To are left blank, the whole chromosome is considered. | Blank |
| Resolution | List of resolutions of contact matrices in the hic file | NA |
| Normalization | List of normalization methods used to normalize contact matrices | NA |
| Extract Contac Data | Click this button to initiate extracting contact data | NA |

1. **Normalize HiC contact matrices**
   1. **Purpose**

To normalize contact matrices in sparse matrix format.

* 1. **Input**

A contact matrix in sparse matrix format (each line represents a contact by three numbers separated by whitespaces: position1 position2 interaction\_frequency)

* 1. **Output**

A normalized contact matrix in sparse matrix format. The matrix is normalized by the Iterative Correction and Eigenvector decomposition (ICE) method

* 1. **Running**:

Access the function from the menu toolbar: 2D-Functions/Normalized HiC Data

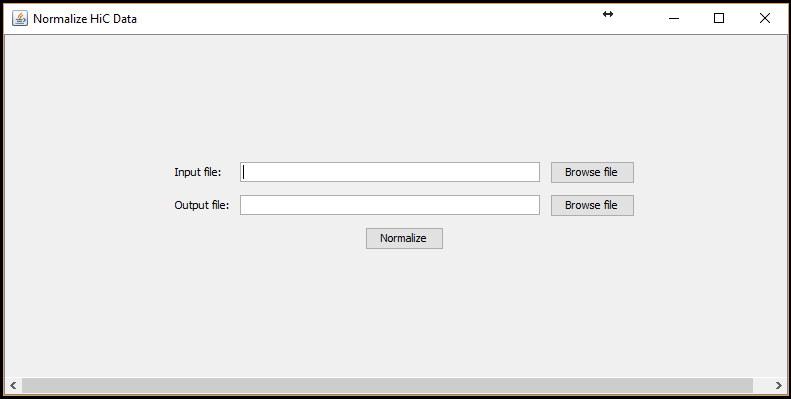


Figure 3: Normalize HiC contact matrices

1. **Visualizing Dataset in 2D format**
   1. **Purpose**

To create a two dimensional (2D) graphical representation of a contact matrix from an input file.

* 1. **Input**

A sparse matrix format (each line represents a contact by three numbers separated by whitespaces: position1 postion2 interaction\_frequency(IF)) or an input file in square matrix format (a full matrix representing all the contact regions). Mark the *Is Square Matrix?* box if the input is a square matrix.

An example sparse matrix file can be found here:

/executable/sample\_data/ contact\_matrices/ chr11\_10kb\_gm12878\_list\_125mb\_135mb.txt

Examples of square matrix files can be found here:

/executable/sample\_data/ contact\_matrices/square\_matrices/

Note: Resolution for square matrices = 40000

* 1. **Output**

A Heatmap which is a graphical representation of contact data where numeric values in the input contact matrix are represented as colors based according to a selected color gradient.

* 1. **Running**

Access the function from the menu toolbar: 2D-Functions/Visualize Dataset.

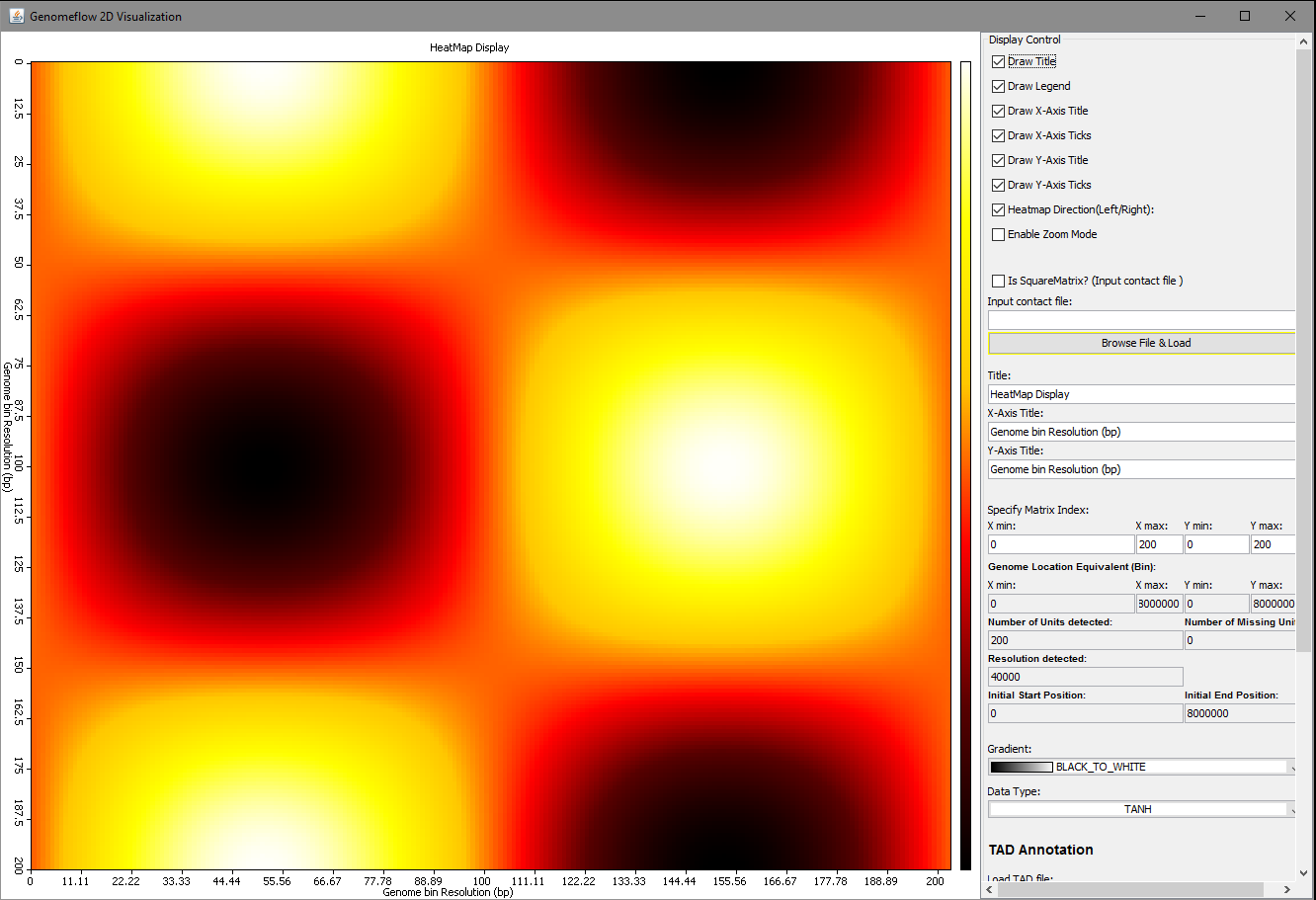


Figure 4: Visualize Dataset in 2D Format

* 1. **Display Controls**

The description of the display controls on the display window is given below.

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Default** |
| Draw Title | It shows or hides the Heatmap title | checked |
| Draw Legend | It shows or hides the color legend | checked |
| Draw X-Axis Title | It shows or hides the X-axis title label on the 2D display window | checked |
| Draw X-Axis Ticks | It shows or hides the X-axis ticks label on the 2D display window | checked |
| Draw Y-Axis Title | It shows or hides the Y-axis title label on the 2D display window | checked |
| Draw Y-Axis Ticks | It shows or hides the Y-axis ticks label on the 2D display window | checked |
| Heatmap Direction(Left/Right) | It changes the Y-axis origin of the heatmap matrix from the Bottom-Left to Top-Left and vice versa | checked |
| Enable Zoom Mode | It allows the user to zoom in/out of the heat map matrix | unchecked |
| Is Square Matrix?(Input contact file) | Allows the user to specify if the input is a Square matrix (a full matrix) or a sparse matrix. If checked, it displays a textbox for the user to specify the matrix resolution. | unchecked |
| Specify Resolution | It is visible only if *Is SquareMatrix?* is checked. It allows user specify resolution for the input matrix. | NA |
| Input contact file | A text file containing a contact matrix in any of the format described above. | NA |
| Title | Allows user to specify the title of the heatmap | Heatmap Display |
| X-Axis Title | Allows user to specify the X-Axis title for the heatmap | Genome bin Resolution (bp) |
| Y-Axis Title | Allows user to specify the Y-Axis title for the heatmap | Genome bin Resolution (bp) |
| X min | It allows the user to specify the minimum X-axis Tick for the heatmap | 0 |
| X max | It allows the user to specify the maximum X-axis Tick for the heatmap | 200 |
| Y min | It allows the user to specify the minimum Y-axis Tick for the heatmap | 0 |
| Y max | It allows the user to specify the maximum Y-axis Tick for the heatmap | 200 |
| X min [Genome Location Equivalent] | It shows the genomic position equivalent for the minimum X-axis Tick for the heatmap | 0 |
| X max[Genome Location Equivalent] | It shows the genomic position equivalent for the maximum X-axis Tick for the heatmap | 8000000 |
| Y min [Genome Location Equivalent] | It shows the genomic position equivalent for the minimum Y-axis Tick for the heatmap | 0 |
| Y [Genome Location Equivalent] | It shows the genomic position equivalent for the maximum Y-axis Tick for the heatmap | 8000000 |
| Number of Units detected | It shows the number of regions found in the input matrix | 200 |
| Number of Missing Units | It shows the number of gaps or missing regions noted from the input matrix | 0 |
| Resolution detected | It displays the resolution of the input matrix | 40000 |
| Initial Start Position | It shows the minimum genome position observed from the input matrix | 0 |
| Initial End Position | It shows the maximum genome position observed from the input matrix | 8000000 |
| Gradient | An array of Color used as a gradient. One color is used as the bottom gradient and another color is used as the top gradient. Hence, it produces a gradient from one color to the other. The Gradient Colors are explained below | HOT |
| GRADIENT\_BLACK\_TO\_WHITE | Produces a gradient from black (low) to white (high) |  |
| GRADIENT\_BLUE\_TO\_RED | Produces a gradient from blue (low) to red (high) |  |
| GRADIENT\_HEAT | Produces a different gradient for hot things (black, brown, orange, white) |  |
| GRADIENT\_HOT | Produces a gradient for hot things (black, red, orange, yellow, white) |  |
| GRADIENT\_MAROON\_TO\_GOLD | Produces a gradient from maroon (low) to gold (high) |  |
| GRADIENT\_RAINBOW | Produces a gradient through the rainbow: violet, blue, green, yellow, orange, red |  |
| GRADIENT\_RED\_TO\_GREEN | Produces a gradient from red (low) to green (high) |  |
| GRADIENT\_ROY | Produces a gradient through red, orange, yellow |  |
| Data Type | It determines the type of data to be displayed. The types available are the raw input data, a Tanh of input data, a Pearson correlation of input data, and a Spearman correlation of the input data. | TANH |

* 1. **TAD Annotation**

The description of the display controls on the display window for TAD annotation is given below.

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Default** |
| Load TAD file | Browse and Load a .bed format file containing the TADs identified for the input matrix | NA |
| Identified TAD | It shows the TADs in the input file | NA |
| Show TAD on Heatmap | It marks the boundary of the TADs identified on the displayed heatmap |  |
| Display Multiple TADs | Once checked, allows TADs from different method to be overlapped on the same display window. This function is useful for comparing TADs identified by different methods for a dataset. | unchecked |
| Choose Display Color | Choose the color for the TAD boundary marks | Color 1 |

## Demonstration

Figure 5 below shows the TAD annotation for the TADs identified by two TAD identification algorithms (ClusterTAD and DI) for mESC Chromosome 17 from [Ren Lab](http://chromosome.sdsc.edu/mouse/hi-c/download.html).

**Step 1:**

To run this demonstration, load a sample square matrix as the input contact file.

The example file can be found here: /executable/sample\_data/ contact\_matrices/square\_matrices/mESC\_nij.chr17. Resolution for the square matrix = 40000

Load the contact file as instructed in part 4: [Visualizing Dataset in 2D format](#Visualizing_Dataset)

**Step 2:**

Modify the highlighted fields on the display window. The table below shows the values set for each field in the display control.

|  |  |
| --- | --- |
| **Field** | **Value** |
| Draw Title | checked |
| Draw Legend | checked |
| Draw X-Axis Title | checked |
| Draw X-Axis Ticks | checked |
| Draw Y-Axis Title | checked |
| Draw Y-Axis Ticks | checked |
| Heatmap Direction(Left/Right) | checked |
| Enable Zoom Mode | unchecked |
| Is SquareMatrix?(Input contact file) | checked |
| Specify Resolution | 40000 |
| Input contact file | Path/to/chr17/inputfile |
| Title | HeatMap Display |
| X-Axis Title | Number of Bins |
| Y-Axis Title | Number of Bins |
| X min | 500 |
| X max | 700 |
| Y min | 500 |
| Y max | 700 |
| X min [Genome Location Equivalent] | 20000000 |
| X max[Genome Location Equivalent] | 28000000 |
| Y min [Genome Location Equivalent] | 20000000 |
| Y [Genome Location Equivalent] | 28000000 |
| Number of Units detected | 2382 |
| Number of Missing Units | 0 |
| Resolution detected | 40000 |
| Initial Start Position | 0 |
| Initial End Position | 95240000 |
| Gradient | HOT |
| Data Type | TANH |

**Step 3:**

* 1. *Browse & Load* the ClusterTAD file found here:

ClusterTAD: /executable/sample\_data/TAD\_annotation/mESC\_TAD\_bed/ ClusterTAD /chr17.bed.

* 1. Select a Unique from Color 1 to 4. (Ex: Color 1 for ClusterTAD and Color 2 for DI)
  2. Click the *Show TAD on Heatmap* button.

**Step 4:**

To display multiple TADs on the Heatmap, Mark/Check the *Display Multiple TADs* then Repeat [Step 3](#Step3) with the DI file found here:

DI: /executable/sample\_data/TAD\_annotation/mESC\_TAD\_bed/ DI /chr17.bed

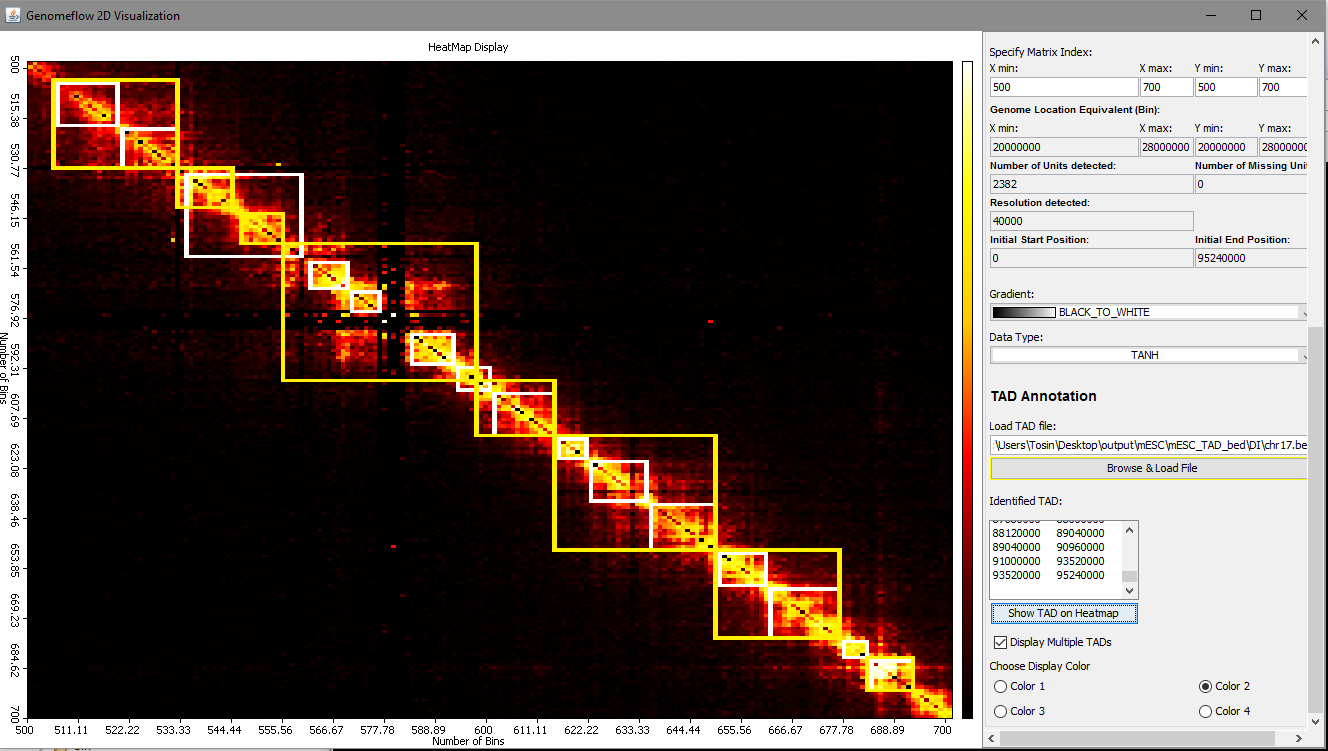


Figure 5: Demonstration of TAD Annotation on 2D Heatmap

1. **Identify TAD**
   1. **Purpose**

To identify Topological Associated domains from input contact matrix.

* 1. **Input**

An input file in square matrix format (a full matrix representing all the contact regions) or a sparse matrix format (each line represents a contact by three numbers separated by whitespaces: position1 postion2 interaction\_frequency(IF)).

An example sparse matrix file can be found here:

/executable/sample\_data/ contact\_matrices/ chr11\_10kb\_gm12878\_list\_125mb\_135mb.txt

* 1. **Output**

A TAD with the best quality will be generated prefixed with ***BestTAD***\_ in *bed* format. This file will be found here: /*Selected\_output\_directory\_from\_GUI*/Output/TADs/*.*

* 1. **Running**

Access the function from the menu toolbar: 2D-Functions/Identify TAD.

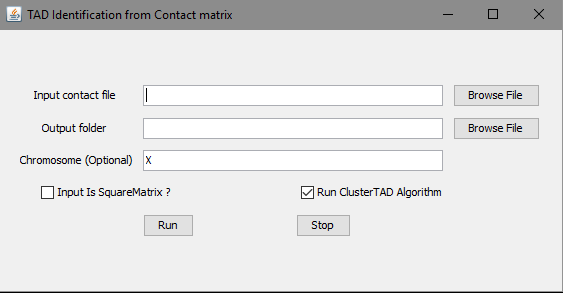


Figure 6: Identifying TADs on a contact matrix

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Default** |
| Input contact file | An input file in any of the format described above | NA |
| Output folder | Directory to output the comparison report | NA |
| Is SquareMatrix?(Input contact file) | Allows the user to specify if the input is a Square matrix (a full matrix) or a sparse matrix. If checked, it displays a textbox for the user to specify the matrix resolution. | unchecked |
| Data Resolution | It is visible only if *Is SquareMatrix?* is checked. It allows user specify resolution for the input matrix. | 40000 |
| Chromosome (optional) | Allows user to specify the chromosome data | X |
| Run ClusterTAD Algorithm | The default algorithm used for TAD identification from the input contact Matrix | checked |
| Run | To start the identification process. A progress bar is displayed to show the steps taken by the TAD identification algorithm,. | NA |
| Stop | During the identification, if this button is pressed, the program will stop. | NA |

1. **Check TAD consistency between two TADs from different methods**
   1. **Purpose**

To compare two TADs from two different Topological Associated domains identification method.

* 1. **Input**

A file containing TADs in .bed format. The method whose TADs consistency is to be checked is termed Method-1, and the methods whose TADs is to be compared with is termed Method-2. Choose the same chromosome for different methods. For example, to compare TAD from ClusterTAD with DI for chromosome 17,

Method-1 = /executable/sample\_data/TAD\_annotation/mESC\_TAD\_bed/ ClusterTAD /chr17.bed.

Method-2 = /executable/sample\_data/TAD\_annotation/mESC\_TAD\_bed/ DI /chr17.bed

* 1. **Output**

A report of the consistency of the Method-1 with Method-2. The output reports the following cases:

|  |  |
| --- | --- |
| **Case** | **Description** |
| Case 1 | The number of Exact TADs found in both Method-1 and Method-2 |
| Case 2 | The number of Sub-TADs that exist between Method-1 and Method-2 |
| Case 3 | The number of Conflicting TADs. |
| Case 4 | The number of TADs in Method-1 but not found in Method-2 |

* 1. **Running**

Access the function from the menu toolbar: 2D-Functions/Check TAD Consistency.

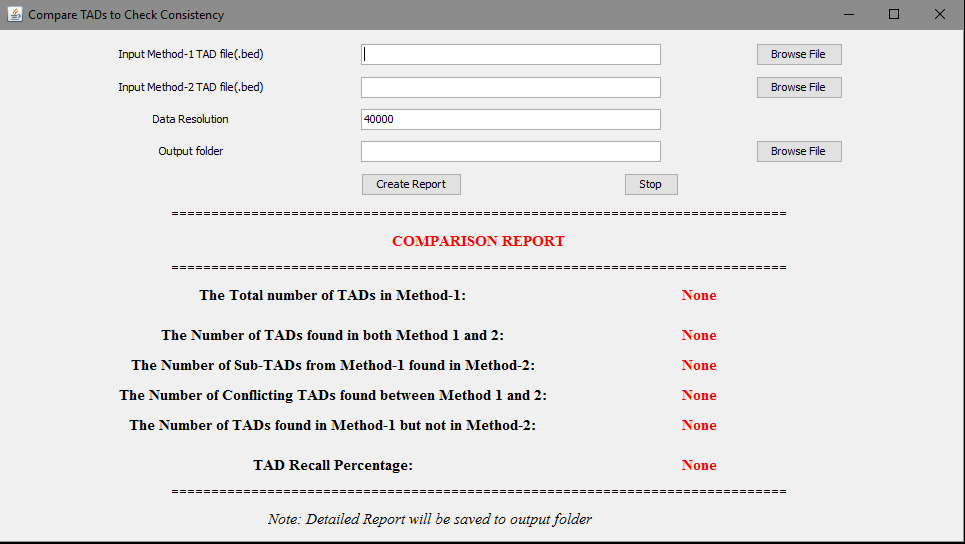


Figure 7: Comparing two TADs for a consistency check

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Default** |
| Input Method-1 TAD file(.bed) | Browse the .bed format file containing the TADs identified by Method-1 | NA |
| Input Method-2 TAD file(.bed) | Browse the .bed format file containing the TADs identified by Method-2 | NA |
| Data Resolution | The Resolution of the dataset the TADs were identified from. | 40000 |
| Output folder | Directory to output the comparison report | NA |
| Create Report | Once this button is pressed, a progress bar is displayed to show the steps taken by the TAD identification algorithm,. | NA |
| Stop | During the check, if this button is pressed, the program will stop. | NA |

1. **3D model reconstruction by LorDG**
   1. **Purpose**

To build 3D chromosomes and genome models

* 1. **Input**

A contact matrix in sparse matrix format

* 1. **Output**

3D models in. gss format file and .pdb format file

* 1. **Running**

Access the function from the menu toolbar: 3D-Functions/LorDG-3D Modeller

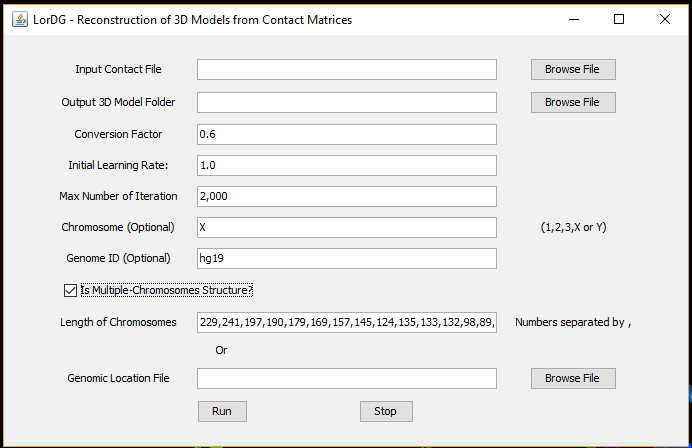


Figure 8: 3D Model reconstruction by LorDG

|  |  |  |
| --- | --- | --- |
| **Field /Button** | **Description** | **Default** |
| Conversion Factor | in the formula , where is interaction frequency between . When the field is left blank, the program will search for the best value in the range [0.1-3.0] with a step size of 0.1. Users can also specify a range to search by put 2 numbers separated by a hyphen (e.g. 0.5-1.0). During the searching, the right-top corner of the main screen displays information about the current value being tested. | 1.0 |
| Initial Learning Rate | Initial learning rate of the optimization. Higher learning rate can speed up the reconstruction process but can cause the process to fail as well | 1.0 |
| Max number of Iteration | Maximum number of iterations for the optimization | 1000 |
| Chromosome | Chromosome name of the contact matrix in the input. If the input contains contact matrix of the whole genome, leave this field blank. | X |
| Genome ID | Genome version of the contact matrix in the input. | hg19 |
| Is Multiple-Chromosomes Structure? | if the input contains both inter-and intra-chromosomal contacts data, this checkbox should be checked. | unchecked |
| Length of Chromosomes | This field contains a list of lengths of chromosomes in increasing order of chromosome names and separated by commas, if “Is Multiple-Chromosomes Structure” is checked. Please note that these lengths should not contain omitted regions (e.g. centromeres) in the input of chromosomes. |  |
| Run | To start the reconstruction process. The main screen displays how models are being formed from initially random models. The information about the reconstruction is displayed in the top-right corner of the main screen. The conversion factor is being used to build model and the current value of the objective function (higher is better). After the reconstruction is finished, the score of the model is displayed in the top-right corner of the main screen (the lower the value is, the better the model is). | NA |
| Stop | During the reconstruction, if this button is pressed, the program will stop and output the currently best structure. If the program is searching for the best conversion factor, it will stop the searching and use the best-found conversion factor to build models. | NA |

1. **3D model reconstruction by 3DMax**
   1. **Purpose**

To build 3D chromosomes and genome models

* 1. **Input**

A contact matrix in sparse matrix format

* 1. **Output**

3D models in .gss format file and .pdb format file

* 1. **Running**

Access the function from the menu toolbar: 3D-Functions/3DMax-3D Modeller

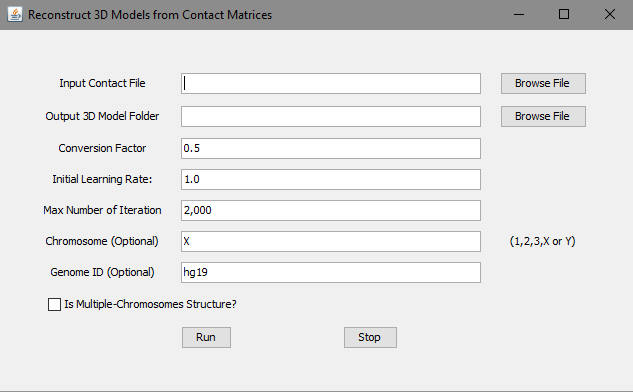


Figure 9: 3D Model reconstruction by 3DMax

|  |  |  |
| --- | --- | --- |
| **Field /Button** | **Description** | **Default** |
| Conversion Factor | in the formula , where is interaction frequency between . When the field is left blank, the program will search for the best value in the range [0.1-2.0] with a step size of 0.1. Users can also specify a range to search by put 2 numbers separated by a hyphen (e.g. 0.5-1.0). During the searching, the right-top corner of the main screen displays information about the current value being tested. | 0.5 |
| Initial Learning Rate | Initial learning rate of the optimization. Higher learning rate can speed up the reconstruction process but can cause the process to fail as well | 1.0 |
| Max number of Iteration | Maximum number of iterations for the optimization | 2000 |
| Chromosome | Chromosome name of the contact matrix in the input. If the input contains contact matrix of the whole genome, leave this field blank. | X |
| Genome ID | Genome version of the contact matrix in the input. | hg19 |
| Is Multiple-Chromosomes Structure? | if the input contains both inter-and intra-chromosomal contacts data, this checkbox should be checked. | unchecked |
| Length of Chromosomes | This field contains a list of lengths of chromosomes in increasing order of chromosome names and separated by commas, if “Is Multiple-Chromosomes Structure” is checked. Please note that these lengths should not contain omitted regions (e.g. centromeres) in the input of chromosomes. |  |
| Run | To start the reconstruction process. The main screen displays how models are being formed from initially random models. The information about the reconstruction is displayed in the top-right corner of the main screen. The conversion factor is being used to build model and the current value of the objective function (higher is better). After the reconstruction is finished, the score of the model is displayed in the top-right corner of the main screen (the lower the value is, the better the model is). | NA |
| Stop | During the reconstruction, if this button is pressed, the program will stop and output the currently best structure. If the program is searching for the best conversion factor, it will stop the searching and use the best-found conversion factor to build models. | NA |

1. **Chromatin loop identification**
   1. **Purpose**

To identify chromatin loop in 3D models

* 1. **Input**

A 3D model to visualize

* 1. **Output**

A list of chromatin loops in a bed format file (optional) and highlighted in the 3D model

* 1. **Running**

Access the function from the menu toolbar: 3D-Functions/Loop Detection

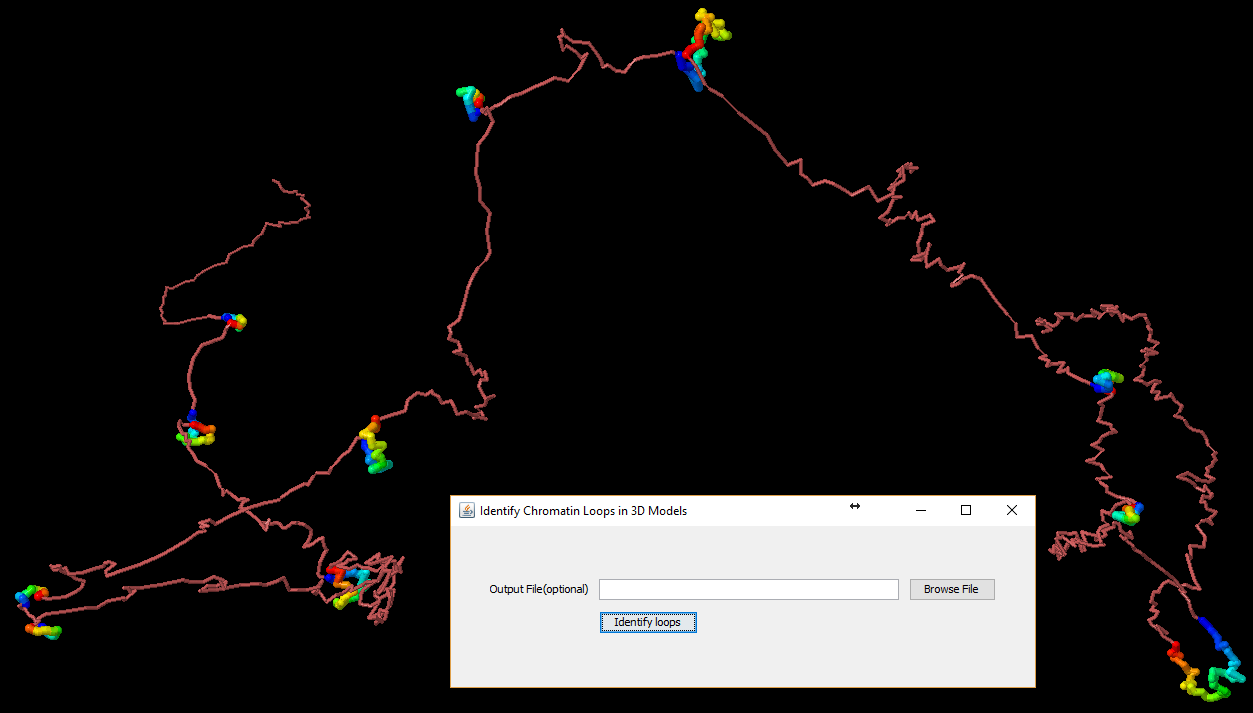


Figure 9: Chromatin loops

The function identifies chromatin loops and highlights them in the 3D model. The loops can also be outputted into a bed format file specified in the Output File field. The top-right corner of the main screen displays the number of chromatin loops identified.

Loops are colored in spectrum (from blue to red). To highlight loops better, color the model by a single color (right-click on the main screen, choose color/structure/chain)

1. **Model annotation**
   1. **Purpose**

To annotate 3D models with genomic elements

* 1. **Input**

A 3D model (e.g. in executable/sample\_data/models) and genomic elements in bed format files (e.g. in executable/track\_files)

* 1. **Output**

3D model is annotated with data from bed format files

* 1. **Running**

Access the function from the menu toolbar: 3D-Functions/Model Annotation

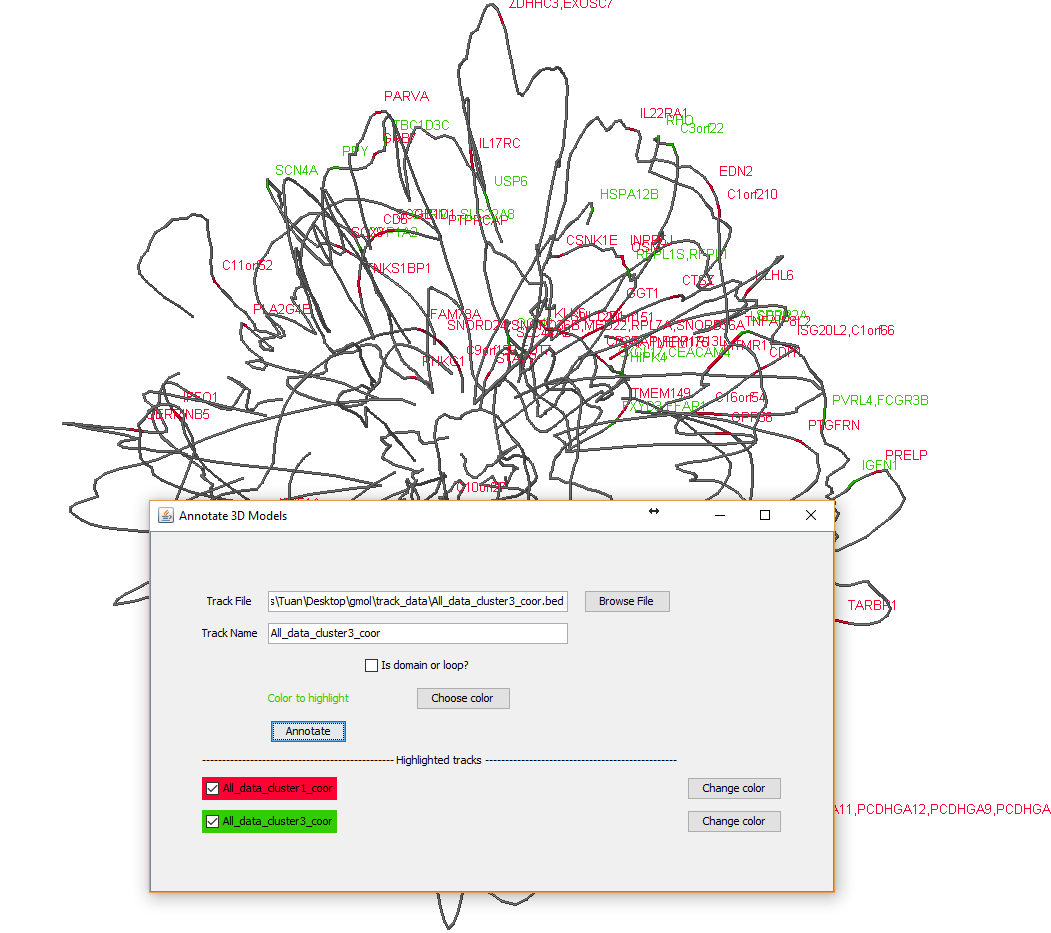


Figure 10: Function to annotate 3D models

To better highlight track data, change the color of the model to a sing color (right-click on the main screen, Color/Structure/Reset). The background can be changed to white to (Color/Background/White)

|  |  |  |
| --- | --- | --- |
| **Field / Button** | **Description** | **Default** |
| Track file | A file in bed format (see executable/track\_files for example) to annotate the model | NA |
| Track name | A unique name associated with the above input file | Name of track file |
| Is domain or loop? | Indicate if the track file contains domains or loops. Adjacent domains/loops will be colored in red/blue alternatively. | Unchecked |
| Choose color | To pick a color to label annotation and points overlapped by genomic elements in the track file. | Random |
| Change color | To change color of the corresponding track | NA |
|  | Checking corresponding track names will display or hidden the content of tracks. |  |

To get the genomic coordinate of a point, left-click or mouse-over to the point as shown in **Figure .**

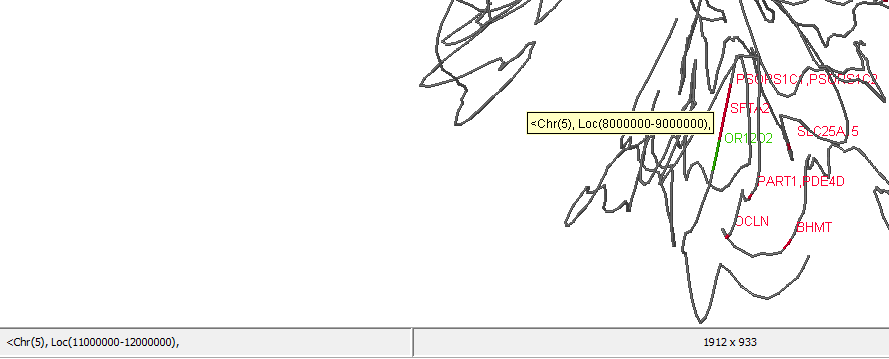


Figure 11: Coordinate of a point in the model

1. **Gene expression data visualization (a special case of model annotation)**
   1. **Purpose**

To display gene expression level along a 3D model

* 1. **Input**
  + A 3D model in GSS format (e.g. in executable/sample\_data/models/ chr11\_10kb\_gm12878\_list\_60mb\_70mb\_1514493462531.gss) to visualize,
  + A gene expression data file in GCT format ( <http://software.broadinstitute.org/cancer/software/genepattern/file-formats-guide#GCT> ), an example file is executable/sample\_data/gene\_expression/allaml.dataset.gct .
  + And a text file to specify genomic coordinates of probes/genes in the GCT format file (each line consists of 4 elements separated by space or tab, e.g.: probe\_or\_gene\_name chr\_number start end). A sample is executable/sample\_data/gene\_expression/probe\_coordinates.txt

These 3 following files are prepared for demo: executable/sample\_data/models/ chr11\_10kb\_gm12878\_list\_60mb\_70mb\_1514493462531.gss, executable/sample\_data/gene\_expression/allaml.dataset.gct and executable/sample\_data/gene\_expression/ probe\_coordinates.txt.

* 1. **Output**

Expression levels of genes/probes are annotated in the 3D model. Usually, the GCT file contains several samples and therefore, the median value (across all samples) together with minimum and maximum values (in brackets) are displayed next to probe/gene names.

If the 3D model and the gene expression data file have no overlap, no annotation will be added to the 3D model.

* 1. **Running**

Access the function from the menu toolbar: 3D-Functions/Model Annotation. A GCT file must be filled in the “Track File” field.

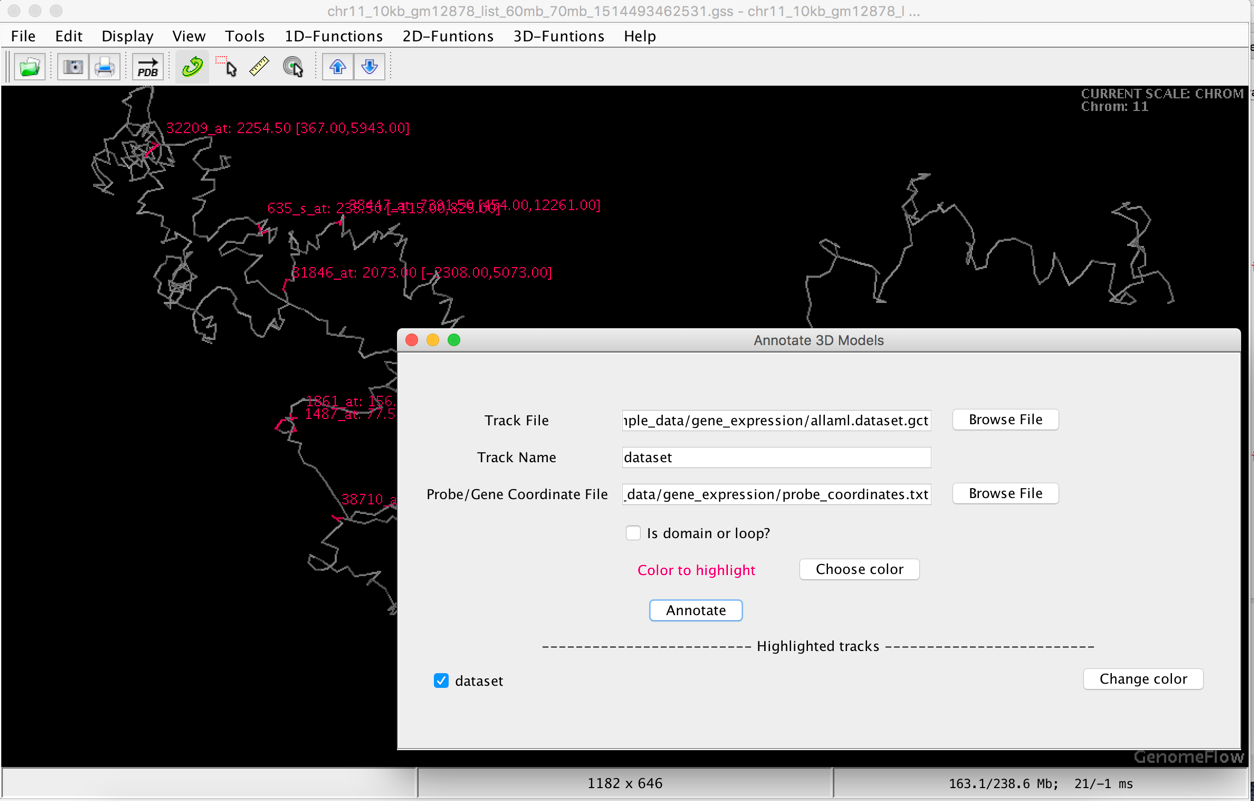


Figure 12: Gene expression visualization demonstration

1. **Comparing 2 models**
   1. **Purpose**

To superimpose and compare two 3D-models in GSS format.

* 1. **Input**

Two chromosome models in GSS format.

* 1. **Output**

The two models are scaled, superimposed and visualized. Spearman’s correlation and RMSE between pairwise distances of the two models are calculated.

* 1. **Running**

Access the function from the menu toolbar: 3D-Functions/Compare Models

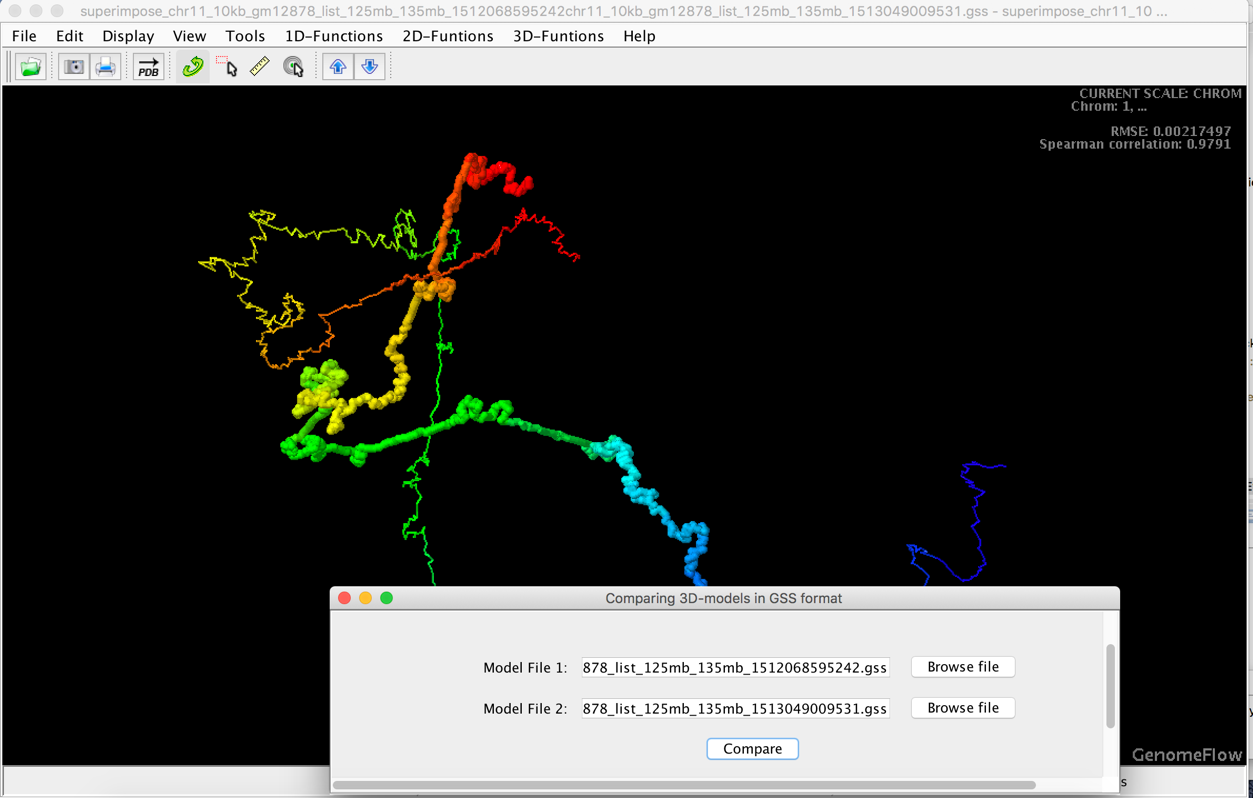


Figure 13: Comparing two constructed models