

DSMEditor User Manual

Table of Contents

Types of DSMs	2
Feature Comparison of DSM Types.....	2
Basic Functionality	2
1. Creating A Matrix	2
2. Modifying the Matrix	4
Adding Rows	4
Adding Connections.....	5
Sorting Items	6
Interfaces	7
Groupings	7
3. Visualizing the Matrix	8
View Modes.....	8
Search.....	8
Cross Highlight.....	9
Cell Marking.....	9
4. Matrix Analysis	9
Validate Symmetry.....	9
Propagation Analysis.....	9
Example Propagation Analysis	9
Cluster Analysis	10
Cluster Algorithm	11
5. Importing/Exporting Matrices.....	11
Importing.....	11
Exporting.....	12
6. Miscellaneous.....	12
Transpose.....	12
Undo/Redo.....	12
Save on Exception	12
7. Keyboard Shortcuts	13

Types of DSMs

The DSMEditor application supports three types of design structure matrices (DSM): Symmetric, Asymmetric/Non-Symmetric, and Multi-Domain (MDM). The symmetric matrix is the most common type of DSM because it is the simplest and most intuitive.

Feature Comparison of DSM Types

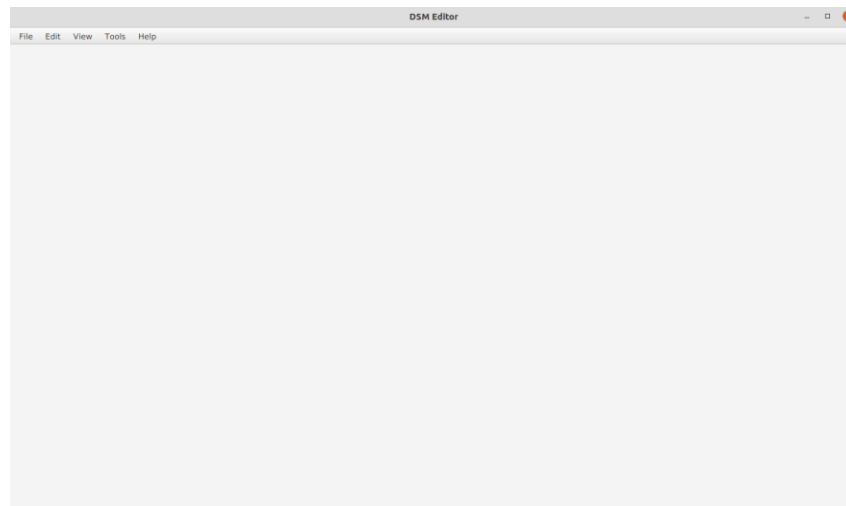
Feature	Symmetric	Asymmetric (Non-Symmetric)	Multi-Domain
Basic Functionality: Adding/deleting items, connections, groupings, and interfaces; sorting, searching, transpose; export CSV, Adjacency Matrix (CSV), Excel, PNG; propagation analysis	X	X	X
Clustering Algorithms (Thebeau, ART-1)	X		
Export Thebeau Matlab File	X		
Hierarchical Grouping			X
Thebeau Cluster Analysis	X		
Validate Symmetry	X		X

Basic Functionality

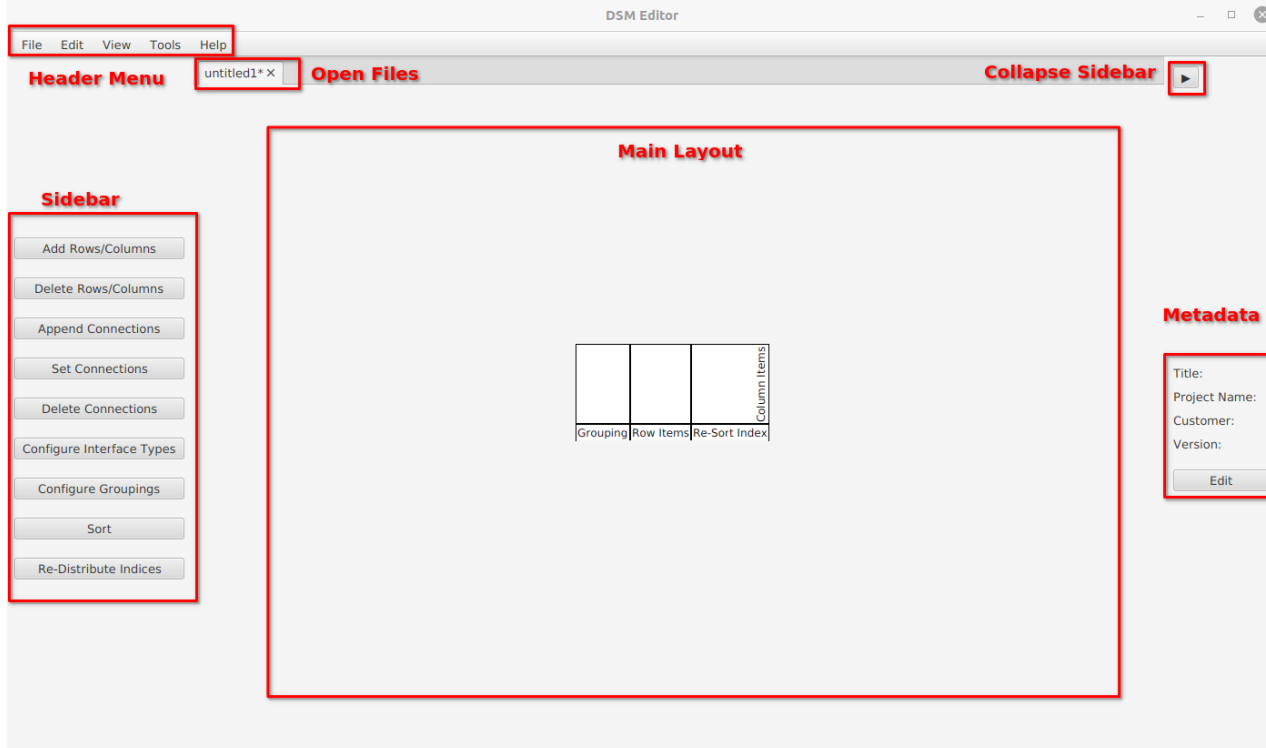
1. Creating A Matrix

Open up the DSM application either from IntelliJ as a developer or by launching the executable. The latest versions of DSMEditor contain a .exe file for convenience that can be directly executed on the Windows operating system by double-clicking the file. The Mac and Linux versions contain a shell script that launches the DSMEditor by invoking the Java runtime. This will require the use of the terminal to run the shell script. It is assumed that if you are using one of these operating systems that you will know how to do this.

When the application starts, it will look like the following:



This tutorial will show how to create and edit a Symmetric DSM, but the basic functionality is similar between all matrix types. Click File > New > Symmetric Matrix. You will then see a matrix appear:



A brief explanation of each highlighted section is given below:

Header Menu:

- File: contains options for importing and exporting files, saving, and opening new files
- Edit: undo/redo functionality, high level matrix modifications
- View: zoom, view mode (names, weights, interfaces, fast-render)
- Tools: various options to help in design and analysis
- Help: Application info and submit bug report

Open Files: Contains a list of open files. These tabs can be rearranged like in a standard file editor. An asterisk will appear when the file has been modified but not saved. There is no auto-save. The asterisk may appear spuriously when undo and redo functionality are used.

Collapse Sidebar: Allows pane with metadata to be hidden so that more of the matrix can be viewed

Sidebar: This contains different buttons used to modify the matrix. This is the main way to interact with the matrix.

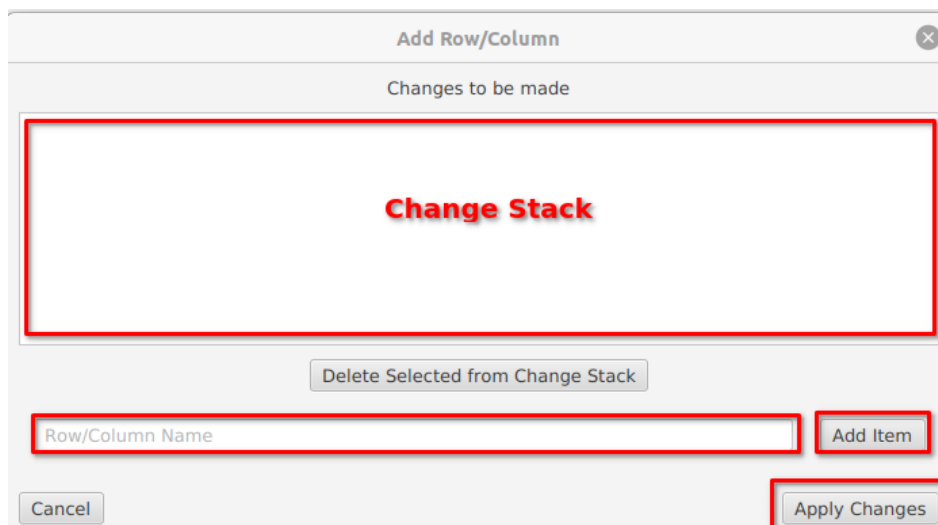
Main Layout: this shows the current state of the matrix.

Metadata: this contains various text entries that can be used to save data with the matrix.

2. Modifying the Matrix

Adding Rows

To start adding rows, click the “Add Rows/Columns” button on the left sidebar. A window will appear:



To start, type the name of the row/column. Because this is a symmetric matrix. The row and column name will be the same. The application knows that the matrix is symmetric, so the user does not need to add two items, one for the row and one for the column. Once the text is entered, either press the enter key or click add item. The enter key is considered a keyboard shortcut. The operation will appear in the change stack. When an item is in the change stack, it can be removed by selecting it (multiple operations can be selected by either ctrl or shift clicking) and clicking the “Delete Selected from Change Stack” button. When all the desired row names are entered, click the “Apply Changes” button. For MDMs, the domain must be selected before creating the item and it cannot be changed once the item is created. An example of what it will look like is below:

Add Row/Column

Changes to be made

- a (Row/Column)
- b (Row/Column)
- c (Row/Column)
- d (Row/Column)
- e (Row/Column)
- f (Row/Column)

Delete Selected from Change Stack

f Add Item

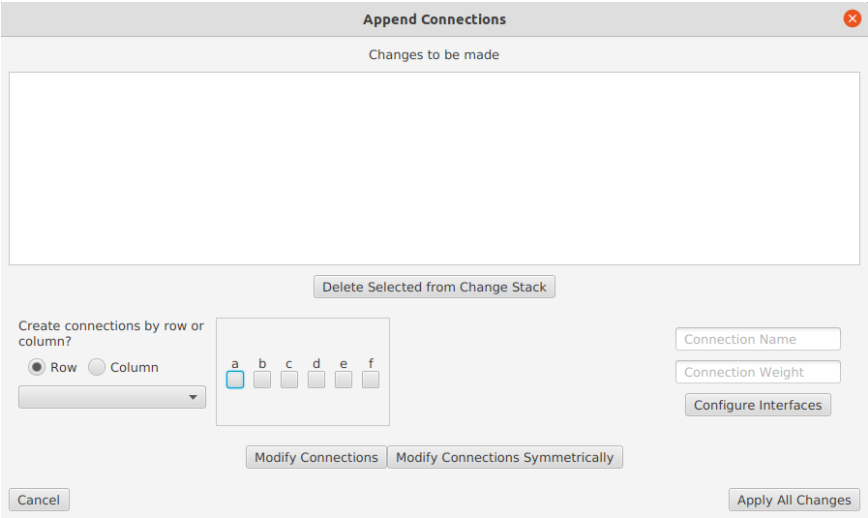
Cancel Apply Changes

Rows can be renamed by clicking on them in the matrix view. They can only be deleted through the Delete Rows/Columns user interface.

Adding Connections

There are several ways to add connections to the matrix. (1) Through the Append Connections button, (2) through the Set Connections button, and (3) by clicking on an individual cell. All 3 options allow for the same configuration of matrix connections by giving ability to set the name/type, weight, and interfaces. The name window is useful of a single character identifier, weight is for associating the connection to a number (think weighted graph), and the interfaces are for defining more complex relationships. Note: interfaces should typically be configured before creating connections, but they can be added later too. Append Connections allows adding connections to a given row or column. It will not delete connections that already exist in that row or column. Set Connections will set the row or column connections exactly as shown by deleting existing connections if they are deselected. Connections are considered non-existent or deleted if the connection name is blank/an empty string. This is typically useful for directly modifying individual connections through method 3.

As with the Add Rows/Columns interface, the Append Connections and Set Connections also have a change stack window with the same functionality. The Append Connections interface is given below:

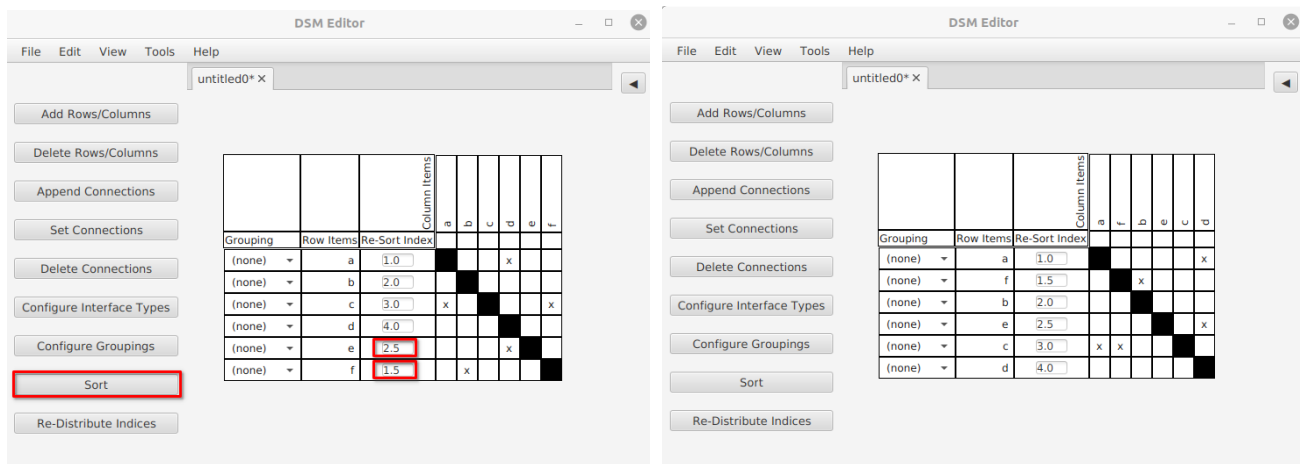


The image shows a software window titled "Append Connections" with a close button in the top right corner. Below the title bar is a section labeled "Changes to be made" containing a large empty rectangular area. Underneath this is a button labeled "Delete Selected from Change Stack". Below that, there is a section titled "Create connections by row or column?" with two radio buttons: "Row" (which is selected) and "Column". To the right of these radio buttons is a 1x6 grid of checkboxes labeled "a", "b", "c", "d", "e", and "f". The checkbox for "a" is checked. To the right of the grid are two text input fields: "Connection Name" and "Connection Weight". Below these fields is a button labeled "Configure Interfaces". At the bottom of the dialog are three buttons: "Cancel" on the left, "Modify Connections" in the center, and "Modify Connections Symmetrically" on the right. In the bottom right corner, there is a button labeled "Apply All Changes".

To start, select if you want to modify connections by the row or by the column (horizontal vs vertical respectively). Next, select either the row or the column you desire to add connections to. Click each checkbox to add a connection between these two elements. Give the connection a name and a weight. Note: you can only add connections using the append interface, so the name and weight cannot be left blank. A good default value for weight if weights are not being used is 1. Finally click Modify Connections to add the connections as specified, or Modify Connections Symmetrically to add the symmetric connection as well. As in the Add Rows/Columns interface the change stack shows all the changes in the order they will be performed. Click cancel to not perform the changes (or close out the window) or Apply All Changes to add the connections to the matrix.

Sorting Items

Items can be sorted after they are added to the matrix by using the Re-Sort Index numeric text entry. The items will be sorted in ascending order based on the index specified in the text entry. Decimal values of arbitrary precision can be entered. Clicking the Sort button will move all the elements to the new positions. To update the numbering to be from 1.0 to N (number of elements), click the Re-Distribute Indices button. An example of sorting is below:



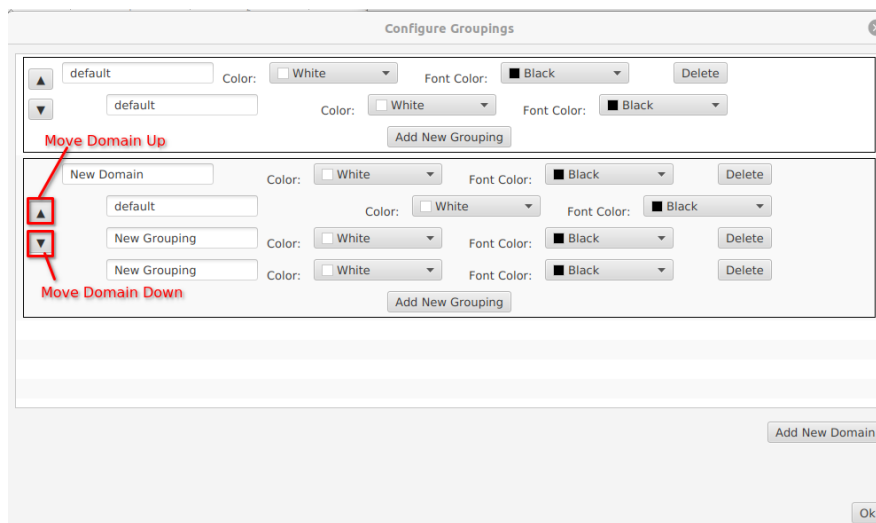
Interfaces

Interfaces are a property of the matrix connections. They allow for hierarchically adding properties to matrix connections. The hierarchy is two levels denoted as interface groupings and interfaces. Each interface belongs to an interface group. The interface group only contains a name and is primarily used for grouping similar interfaces to enable easier searching when configuring connection interfaces. The interface has both a name and an abbreviation. When using the interface view mode (see View Modes below), the abbreviation will be shown on the matrix view. The abbreviation should ideally be a single letter. In the interface view mode, you will need to go to the View drop-down in the Header Menu to select the visible interfaces in the matrix view. Interfaces are not very well supported except for this basic viewing functionality.

Groupings

Groupings are used to group similar elements in the DSM. Read about DSM Clustering for design philosophy. Groupings need to be configured by clicking the Configure Groupings button. Symmetric DSM groupings are a flat structure where rows and columns have the same list of groups. Non-Symmetric DSM groupings are a flat structure, but rows and columns have a different list of groups. Multi-Domain groupings are hierarchical, they are broken into domain and domain-groupings. Each domain-grouping belongs to a single domain. Groupings/domain-groupings have a default group that cannot be deleted which is semantically equivalent to “no group” or “default group.” This default group does not have to be used as such, but it will always exist, so it is recommended to use it as such.

A special note on domains is that they will appear in the matrix view in the order they appear in the groupings interface. In the image below, the “default” domain will appear before the “New Domain” domain. They can be rearranged by using the arrow buttons on the left side:



3. Visualizing the Matrix

View Modes

The application supports four different view modes that can be selected by choosing View > View Mode in the Header Menu:

Names: shows matrix connections by their given name

Weights: shows matrix connections by their given weight

Interfaces: shows matrix connections by their given interfaces. Note: which interfaces are visible must be configured by going to View > Configure Visible Interfaces. (This option will appear only in Interfaces view mode). This is to ensure the boxes are not too cluttered if interfaces are used extensively.

Fast Render: shows a simplified visual of the matrix where connections are represented only by a dot if the connection exists. Removes the ability to interact with the matrix by clicking in the matrix view. This mode is good for large matrices which slow down the application and are difficult to see in the view-port without scrolling.

Search

The application contains a search feature for highlighting connections which match a criteria. This is opened by choosing Tools > Find Connections in the Header Menu or using the Ctrl-F keyboard shortcut. Users can search for exact match of the name, sub string contained in the name, or by weights. There is not search for interfaces, but users can utilize the Interfaces view mode to achieve similar behavior. Found connections are highlighted in a light blue. This is only a visual property and is not saved to the file in any way.

Cross Highlight

The application contains a cross highlight feature in the matrix layout. This highlights the row and column where the cell the mouse currently hovers over in green. Toggle with Tools > Toggle Cross Highlight or by clicking the F key. This is only a visual property and is not saved to the file in any way.

Cell Marking

Individual cells can be marked temporarily by right-clicking a cell. This will highlight the cell in yellow until the highlight is turned off by right-clicking on the cell again. This is only a visual property and is not saved to the file in any way.

4. Matrix Analysis

Validate Symmetry

This feature is only for Symmetric matrices and MDMs (which are symmetric). This will highlight in orange connections that are not symmetric i.e. they are one-way. Toggle this with Tools > Validate Symmetry.

Propagation Analysis

Given a start item, each level counts the connections to items in the previous level. For example, level 0 contains the start item, level 1 contains all the connections to the start item, level 2 contains all the connections to the connections to the start item, etc. Items that are excluded are added to the count, but not propagated through. For example, if item “b” is found as a connection in level 1, then the count increments by one, but “b” is not considered for level 2 (no connections to “b” will be searched for in level 2). Users can count by occurrence (1 if connection exists, 0 else) or by connection weight.

Once the configuration is set up, the analysis can be run. Choose Run > Run Propagation analysis. The output is given as a bar chart of the propagation count for each element as well as a table output which can be copied and pasted in CSV format.

Example Propagation Analysis

An example propagation analysis is performed on the following DSM (this DSM can be found in the repository under the verification folder as DSMED-10_dsm_file.dsm).

Simple Symmetric DSM

Project Name: DSM Editor Tool QA
Customer: Acme
Version: 0.1.5

Grouping	Row Items	Re-Sort Index	Column Items								
			A	B	C	G	H	D	E	I	J
G1	A	0.5	X	X			X	X			X
G1	B	0.6	X	X				X			
G1	C	0.7	X	X							
G3	G	0.81				X					
G3	H	0.82					X				
G2	D	1.0	X	X				X	X	X	
G2	E	2.0						X			
G2	I	3.0						X		X	
G2	J	4.0	X					X			X

Propagation Analysis performed with: Start item = A Number of Levels to Run = 3 Minimum Weight = 1 Count by Occurrence				
Propagated Element	Propagation Level 1: From Row A Counts	Propagation Level 2: From Row B, H, D, J Counts	Propagation Level 3: From Row B, D, J, C, E, I (A is excluded because it was the starting element) Counts	Total Propagation Count
B	1	1	1+1	4
H	1			1
D	1	1+1	1+1+1+1	7
J	1	1	1	3
A		1+1+1	1+1+1	6
C		1	1	2
E		1	1	2
I		1	1	2
G				0

Cluster Analysis

Runs a cluster analysis on the matrix. This feature is only for Symmetric matrices (including breakout views of MDMs). See Ronnie Thebeau's paper for a description of the algorithm (https://dsmweb.org/wp-content/uploads/2019/05/msc_thebeau.pdf). It is strongly suggested to read at least the first part of the paper to understand at a high level how the algorithm works. The implementation differs slightly in that DSM elements can only belong to a single group by design, so there is no need to penalize items that belong to multiple groups. This behavior can be simulated by explicitly adding duplicate DSM elements and assigning them to multiple groups. The downside is that the number of duplicates is predetermined. The cluster analysis only calculates the cost function (the cluster bids) and does not run the clustering algorithm. See below for the clustering algorithm implementation. A second difference is that this cost function includes an Optimal Cluster Size parameter. This modifies the equation such that

$$\text{IntraClusterCost} = (\text{DSM}(j,k) + \text{DSM}(k,j)) * |\text{ClusterSize}(y) - \text{opt_cluster_size}|^{\text{powcc}}$$

instead of

$$\text{IntraClusterCost} = (\text{DSM}(j,k) + \text{DSM}(k,j)) * \text{ClusterSize}(y)^{\text{powcc}}$$

The optimal cluster size can be set to 0 to use the original equation.

The output shows green if the group the element belongs to has the highest bid using Thebeau's cost function. It shows red if the group it belongs to had the smallest bid. It shows white if it is in the middle.

Cluster Algorithm

The cluster algorithms are supported only for the Symmetric DSMs. These are mainly for research purposes or to give a starting point for clustering the matrix. There are currently 2 algorithm options: the simulated annealing algorithm by Thebeau and an Adaptive Resonance Theory 1 (ART1) algorithm. The Thebeau algorithm has not been rigorously validated according to the paper, but visual inspection of the clustered output for the DSM used in the paper suggests that the provided implementation is at least similar. One slight difference is that this implementation allows exclusions, meaning that these elements will not be considered when running the clustering algorithm and will exist in the matrix output as they appeared in the input. The ART1 algorithm was an experiment in an alternative clustering method. It groups elements that are similar in the sense that they have similar connections. However, this provides poor performance in terms of the traditional definition of clustering.

The user interface for the cluster algorithms is accessed by choosing Tools > Cluster Algorithms. The algorithm to be run is on the left side. Users can choose between two provides options for clustering. The right-hand side has parameters to run a cluster analysis using Thebeau's algorithm (see Cluster Analysis section above). This allows for comparing different outputs using the same cost function.

5. Importing/Exporting Matrices

Importing

Matrices can be imported into the DSMEditor by choosing File > Import and then selecting the type of file to import. There are two options: Thebeau Matlab file and Adjacency Matrix. The Thebeau Matlab file is only for Symmetric matrices, however.

Thebeau Matlab File: File format used in Thebeau's Matlab scripts. His script represents a DSM in a Matlab file and supports connections and groups. See the downloadable files at <https://dsmweb.org/matlab-macro-for-clustering-dsms/> for more information.

Adjacency Matrix: As the name implies, this exports an adjacency matrix in CSV format where the value is 0 for connections that don't exist and the connection weight for connections that do exist. The file format has a few extra features to allow for the multiple DSM types and groups. This format is recommended for research where is desired to pass matrices to script(s) that operate on them (such as a Python script to cluster a matrix). An example of what the file looks like for each of the DSM types when opened with Excel is below. These do not necessary have to be exported from the DSMEditor tool, but can be custom generated.

Symmetric

symmetric						
<group>	a	f	b	e	c	d
(none)	0	0	0	0	0	1
(none)	0	0	1	0	0	0
(none)	0	0	0	0	0	0
(none)	1	0	0	0	0	1
(none)	1	1	0	0	0	0

(none)	0	0	0	0	0	0
--------	---	---	---	---	---	---

Asymmetric/Non-Symmetric

asymmetric					
		(c_none)	col_g1	col_g1	col_g1
<group>	<row>	e	f	g	h
(r_none)	a	0	0	0	1
(r_none)	b	0	1	0	0
row_g1	c	0	0	0	0
row_g1	d	1	0	0	1

Multi-Domain

multi-domain								
<domain>	<group>	a	b	c	d	e	f	g
domain1	d1-none	0	0	0	0	1	0	0
domain1	d1g1	0	0	0	0	0	0	0
domain1	d1g1	0	0	0	0	0	0	1
domain1	d1g1	1	0	0	0	0	0	0
domain2	d2-none	0	0	0	0	0	0	0
domain2	d2-none	0	0	0	1	0	0	0
domain2	d2g1	1	0	0	0	0	0	0

Exporting

All matrix types have several export options: CSV, Adjacency Matrix (CSV), Excel, PNG. CSV will export connection names, Adjacency Matrix will export weights (see above), Excel will export connection names, and PNG can export either names or weights. It is currently not supported to be able to export interfaces.

Symmetric matrices can additionally be exported as Thebeau Matlab files. See above for more information.

6. Miscellaneous

Transpose

Matrices have a transpose feature. This equivalent to the traditional matrix transpose operation where the rows and the columns are interchanged. Use Edit > Transpose Matrix.

Undo/Redo

Changes made can be undone and redone by either using the keyboard shortcuts or by going to Edit > Undo or Edit > Redo

Save on Exception

When an exception occurs in the DSMEditor, a recovery file of the matrix is saved. It is sometimes difficult to tell when exception occurs because the application is designed to handle them gracefully (a little too gracefully at the moment). If some behavior was expected after user interaction, and it doesn't happen, an exception likely occurred.

There is also a log file generated with the exception stack trace. The files are created in a directory labeled .recovery and .log respectively. It is located wherever the executable file (either .exe or .jar) is located. They may be hidden however due to naming conventions. The contents of the log file will be very helpful when submitting a bug report.

7. Keyboard Shortcuts

F	Toggle Cross-Highlight
Ctrl+F	Connection Search
Ctrl+I	Zoom In
Ctrl+O	Zoom Out
Ctrl+S	Save
Ctrl+Shift+S	Save As
Ctrl+Y	Redo
Ctrl+Z	Undo