BiGGEsTS



BiclusterinG Gene Expression Time Series Quickstart Guide for v1.0.4

BiGGEsTS is a software tool for time series gene expression data analysis, based on biclustering algorithms particularly suited for this kind of data. It is open source and freely available at: <u>http://kdbio.inesc-id.pt/software/biggests/</u>. The purpose of this quickstart document is to provide simple instructions on how to install and use BiGGEsTS. It is written under the assumption that the researcher knows what microarrays and expression data are. However, anyone who wants to try or start using BiGGEsTS without any previous knowledge on these concepts will be able to run the software using this guide. For obtaining help, sending feedback and improvements and/or reporting issues related to BiGGEsTS software, contact Joana P. Gonçalves at: jpg@kdbio.inesc-id.pt.

Note that BiGGEsTS is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or any later version. BiGGEsTS is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details. A copy of the GNU General Public License is included with BiGGEsTS. For detailed information about the license see the GNU licenses at http://www.gnu.org/licenses/.

Pre-Requisites

If you want to install BiGGEsTS, please check if you have a JVM (Java Virtual Machine) with JDK/JRE 1.5 or higher installed on your system.

We recommend a minimum of 1024 MB of RAM and 300MB of available disk space for installing and running the BiGGEsTS software.





Index

1. Installation	3
2. Running BiGGEsTS	
2.1. Menu bar	6
2.2. Loading a dataset	7
2.3. Analyzing matrices	
2.4. Preprocessing time series gene expression data	
2.4.1. Filtering of missing values	
2.4.2. Filling of remaining missings	
2.4.3. Normalization	
2.4.4. Smoothing	
2.4.5. Discretization	
2.5. Biclustering time series expression data	
2.5.1. CCC-Biclustering	
2.5.2. <i>e</i> -CCC-Biclustering	
2.5.3. CCC-Biclustering and <i>e</i> -CCC-Biclustering extensions	
2.5.4. CC-TSB	17
2.6. Visualizing biclustering results	
2.7. Analyzing GO terms	23
2.8. Post-processing groups of biclusters	25
3. Formal Definitions	
3.1. Expression matrix	
3.2. Bicluster	
3.3. Coherent biclusters with contiguous columns	
3.4. Contiguous column coherent biclusters with exact patterns	
3.5. Contiguous column coherent biclusters with approximate patterns	
3.6. Sign-changes (anticorrelated patterns)	29
3.7. Gene-shifts (scaled patterns)	29
3.8. Time-lags (time lagged patterns)	30
3.9. Post-processing metrics	



1. Installation

On Windows, using the installer (the following images were taken with BiGGEsTS 1.0.3 installer and differ visually from the ones in BiGGEsTS 1.0.4, but the installation steps are exactly the same):

1. Double-click on the installer file and click **Next**.



2. After reading the terms of the license, select the *l accept the agreement* option and click *Next*.

📑 Setup		×				
License Agreement	Bi C Es 1	G S				
Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.						
	GNU GENERAL PUBLIC LICENSE					
	Version 3, 29 June 2007					
Copyright (C) 2007 Free Software Foundation, Inc. <http: fsf.org=""></http:> . Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.						
	Preamble	~				
Do you accept this license?	I accept the agreement					
	I do not accept the agreement					
BitRock Installer						
	< <u>B</u> ack <u>N</u> ext > Cancel)				

3. Specify the directory where BiGGEsTS should be installed and click *Next*.

Installation Directory Please specify the directory where BIGGESTS will be installed. Installation Directory C:\Program Files\BIGGESTS	🕉 Setup				>
Please specify the directory where BIGGESTS will be installed. Installation Directory C:\Program Files\BIGGESTS BifRock Installer	Installation Directo	гу			Bi GG Es <mark>TS</mark>
Installation Directory C:\Program Files\BiGGEsTS 6	Please specify the dire	ctory where BiGGEsTS will b	e installed.		
BìRock Installer	Installation Directory	C:\Program Files\BiGGEsTS		6	
BitFlock Installer					
BiRock Installer					
BitRock Installer					
BitRock Installer					
BitRock Installer					
Rack Next > Cancel	BitRock Installer		< Back	Nevt	Cancel

4. After a successful installation you will be prompted to finish the process. You can do this by clicking the *Finish* button. BiGGEsTS is now ready to be used. You will find shortcuts to BiGGEsTS on both the **Desktop** and the **Start** menu.

	Completing the BiGGEsTS Setup Wizard
	Setup has finished installing BiGGEsTS on your computer.
Bi GG	
Es <mark>TS</mark>	
	< Back Finish Cancel

On Windows using the multi-platform distribution:

- 1. After downloading the zip or tar.gz file from BiGGEsTS website, decompress it to a suitable location.
- 2. You can execute the installer file contained in the resulting installation directory (double-click the install.bat file). This will install and set the paths to Graphviz and GO files. If you skip this step, it will be automatically performed the first time you run BiGGEsTS. Wait for the installation to conclude.
- 3. You may then execute BiGGEsTS anytime by calling the biggests script file (double-click the biggests.bat file). We strongly advise you to run BiGGEsTS through this script, rather than executing the jar directly, unless you are acquainted with JVM and its parameters and want to tune them yourself.

On MacOS and other UNIX systems (Linux, etc), using the multi-platform distribution:

- 1. Install Graphviz on your system. You'll find the source code and binaries, as well as documentation, at http://www.graphviz.org/. From Graphviz installation we will need only the **dot** command-line program. On Linux, Graphviz dot gets installed by default at /usr/bin/dot. On MacOS, you'll find it at /usr/local/bin/dot.
- 2. Download the zip or tar.gz file from BiGGEsTS website and decompress it to a suitable location.
- 3. If you chose to install in a custom directory, you can edit BiGGEsTS installer file (install.sh), append the path to the dot binary file (default is /usr/bin/dot on Linux and /usr/local/bin/dot on MacOS) preceded by a space, to the last line (e.g. "java -classpath biggests.jar biggests/utils/BiggestsInstall /usr/bin/dot"). Alternatively, you can leave the install.sh script untouched and you'll be prompted by BiGGEsTS to indicate the dot file location and allowed to browse the file system to look for it. Execute the install.sh script file to automatically set the path to the dot file and install the GO Files. You may skip this step, but in that case it will be automatically performed the first time you run BiGGEsTS.
- 4. You may now execute BiGGEsTS whenever by simply running biggests.sh script.



Important notice regarding the initialization of the JVM:

When executing BiGGEsTS through biggest.bat or biggest.sh, the parameters of the Java Virtual Machine (JVM) are automatically tuned according to the characteristics of the system in which the application is being executed. In some cases, however, it is not possible to automatically figure a good combination of parameters.

A common problem concerns the object heap size. Specifically, when an error occurs, a message similar to the following is displayed (the values after –Xms and –Xmx may be different):

java -Xss5M -Xms<mark>1073204</mark>k -Xmx<mark>1073204</mark>k -jar biggests.jar Error occurred during initialization of VM Could not reserve enough space for object heap Could not create the Java virtual machine.

The most direct solution to overcome this is to edit the biggest.bat (Windows) or biggest.sh (UNIX) file and replace the Xms and Xmx parameters by values lower than the automatically determined ones (highlighted in green). You should also guarantee that the new values do not exceed the amount of physical memory available in your system. In 32-bit Windows machines with more than 2GB of RAM, these values should not exceed 1500000KB, that is, approximately 1.5GB.

The biggest.bat and biggest.sh files are also available in the version of BiGGEsTS for Windows provided with the installer. They are usually contained in the installation directory (typically C:\Program Files\BiGGEsTS\).



2. Running BiGGEsTS

2.1. Menu bar

The menu bar contains three menus: *Session, Settings* and *Help*. Functionalities accessible via this menu bar can always be performed, regardless of the node and/or the tab selected in the dataset tree and the tabbed panes.

The *Session* menu contains the following items:

- Load dataset: it basically selects the Loading tab.
- **Load session**: enables loading a session previously saved using the functionality **Save session**, also available in this menu. The user is always prompted if the current session should be saved before loading a new one.
- **Save session**: saves a session. The user is prompted for specifying the location for the resulting session file, compressed in a zip file.
- **Exit**: Prompts the user if the current session should be stored and terminates the application.

The *Settings* menu contains the following item:

 Color scheme: enables the selection of one of two color schemes for heatmaps (tables of colors and symbols). Available options are *Red and green* and *Yellow* and blue. The red and yellow and the green and blue colors are used for the higher and lower expression values, respectively, in the corresponding schemes.

The *Help* menu contains the *About...* item, which displays a popup with information about the application, such as the authors and the version.



2.2. Loading a dataset

When you start BiGGEsTS, the main window looks like this (program starts on the *Loading* functionality):

BiGGEsTS - Biclustering Gene Expression Time Series		
Session Settings Help		
Datasets	oading Preprocessing Biclustering Analyzing Post-processing Dataset file Browse_	<u> </u>
Information This box displays information about matrices, algorithm applications, etc.	Organism Pseudomonas fluorescens Pf-5 Pseudomonas syringae DC3000 Preudomonas syringae DC3000 Probeset IDs in column number Probeset IDs in column number Probeset IDs in column number Browse Sichactor promoty IDS-3 Solanaceae Typanosoma brucei Typanosoma brucei CDMA U COMA U COM U C	=
	Load a dataset	×

The BiGGEsTS window.

You may want to **load a dataset** from a character delimited text file. Type the **path** to the file to load (or click *Browse...* button and select the dataset file in file system instead; sample files are also available in the Datasets directory, the default location for raw time series expression data, within the directory where BiGGEsTS is installed; see the readme.txt file in the same directory for specific details on the contents of these sample files). You'll be presented a preview of the data in your file.

The dataset text file must be structured like this:

	C ₁	C ₂	C ₃	C ₄	C ₅	C ₆	
G_1	E ₁₁	E ₁₂	E ₁₃	E ₁₄	E ₁₅	E ₁₆	
G ₂	E ₂₁	E ₂₂	E ₂₃	E ₂₄	E ₂₅	E ₂₆	
G ₃	E ₃₁	E ₃₂	E ₃₃	E ₃₄	E ₃₅	E ₃₆	

In which each G_x is a name of a gene, each C_y is a name of a condition (time-point) and each E_{xy} is the expression value of gene G_x in condition C_y . Values must be delimited by a specific character, which may be a **tab**, regular **space** or a **semicolon** (;). The gene expression data may also have some missing values (blank gaps). BiGGEsTS will deal with them later.



BiGGEsTS - Biclustering Gene Expression Time Series				_ - X
Session Settings Help				
Datasets	Loading Preprocessing Bicluster	ng (Analyzing (Post-processing)	Browse	^
	Organism Pseudomonas fluorescens Pf-5 Pseudomonas syringae pt-2000 Pseudomonas syringae pt-phaseoli Raftus norvegicus Saccharomyces cerevisian Schizosaccharomyces pombe	Open Look In: Decal Disk (C:) MSOffice Peri Ver Program Files Strawberry Swsetup SYSTEM.SAV File kame: Files of Type: TXT Files (*txt)	TEMP Users WINDOWS heat_stress.txt yeast_heat_stress.txt	=
BIGGESTS Information Box	Shiekkaneta oneoensis mr.1 Silicibacter pomeroyi DSS-3 Solanaceae Trypanosoma brucei Trypanosoma brucei chr 2 Vibrio cholerae		Open Cancel	_
		Load Load a	a dataset.	v

BIGGESTS - Biclustering Gene Expression Time Series									L	_ = ×	j
Session Settings Help											Ĩ.
Datasets	Loading	Preprocessing	Biclusteri	ng Analyzing	Post-processing						1
											Ĩ
	Dataset	file									1
	C:\heat_	_stress.txt					Brows	e			l
											l
		0	1	2	3	4	5				L
	GENES	Avera	ge 000 min	Heat Shock 005	. Heat Shock 015	Heat Shock 030i	Heat Shock 060) 🔺			L
	TFC3	-0.56		-0.14	0.38	0.34	0.23	=			L
	VPS8	-0.62		-0.13	0.77	0.77	-0.57				L
	EFB1	0.6		0.33	0.81	-0.12	-0.12				L
	YAL004	W -0.1		0.09	-0.97	0.19	0.28				L
	SSA1	-0.71		-0.03	1.41	0.8	0.0	_			L
	ERP2	0.24		-0.27	0.12	-0.15	0.43	_			L
	PONT4	-0.43		-0.20	0.02	0.04	0.00	_			L
	MDM10	-0.15		0.14	0.78	0.14	0.19				L
	014/00	0.00		0.04	0.70	0.04	0.07	-			L
Information This box displays information about matrices, algorithm applications, etc.	Organis Pseudo Pseudo Rattus r Saccha Schizos Shewar Silicibar Solanac Trypanc Vibrio c	m monas fluoresce monas syringae norvegicus romyces cerevis accharomyces p tella oneidensis h cter pomeroyi DS ceae usoma brucei chr holerae	ns Pf.5 DC3000 pv. phaseolico iae oombe dR-1 S-3 2	ola 1448A	Conditions' name: File contains g Probeset IDs in cc IDs conversion file Datatype CDNA	s in row number ene IDs in column lumn number	number 0 v	Browse			
					Load						
						Load a dataset.					Î

The input of time series gene expression data.

If the names of the genes in your file comply with an identification nomenclature system other than HGNC (HUGO Gene Nomenclature Committee), then you must uncheck the *File contains gene IDs in column number* checkbox, specify which column contains the names of the genes and provide an ID conversion file. This should be a character delimited text file (allowed delimiters are the same as the ones for dataset files) containing two columns: the first with the probeset IDs (the names of the genes used in the experiment) and the second with the corresponding HGNC names. You may also use BiGGEsTS with probeset IDs without converting them by leaving the *File contains gene IDs in column number* checkbox checked. Just be aware that, if you do this, function analysis won't produce valid results, since such names can't be matched with the ones used in the Gene Ontology files.



Select the number of the row which contains the **names of the experimental conditions** and the correct **data type**.

To perform a **log transformation** on the loaded data, check the **Log Transformation** checkbox. In that case, both **Original** and **Preprocessed** matrices containing the original and the log transformed data matrix, respectively, will be added to the tree.

Finally, click the *Load* button. BiGGEsTS will check if the Gene Ontology files are available for the organism that you have selected and if that is the case, it retrieves the GO terms that annotate the genes of the loaded dataset. This may take a few seconds but once it is done, one is able to check which GO terms annotate a given gene, by clicking on the corresponding row in the matrix. Note that the GO terms retrieved correspond to the most specific GO terms annotating the gene (before applying the true path rule).



2.3. Analyzing matrices

After loading the sample dataset, a new dataset is created in the dataset tree (on the left side of the window). An **Original** matrix is added to this new dataset, containing the loaded gene expression data. By default, BiGGEsTS shows the colored expression matrix (*Colors* tab), but you may also visualize the matrix with the original data by clicking on the *Values* tab (bottom of the window).



Table of colors for the original matrix. Lower values are colored green or blue, higher values are colored red or yellow and missing values are set to yellow or white, depending on whether the *Red and green* or the *Yellow and blue* color scheme is in use.

All kinds of matrices are **sortable**. This means that you can have the genes (rows) of a matrix sorted by their expression values/symbols for a given condition (column) in increasing or decreasing order. This is done by clicking the cell that corresponds to the condition, that you want the rows of the table to be sorted by, in the header of the table. A first click sorts the values in increasing order, a second one does it in decreasing order and a third causes the rows to return to their original order (resets).

Moving your mouse over one of the cells of the matrix displays a specific **text tip** according to the content of the cell.

Matrices can be **exported as image files**. The access to this functionality is provided via a click on the right button of the mouse on any area occupied by the table, except the header. When exporting a matrix as an image, you will be prompted to provide a proper file name, location and format (PNG and JPG available).

As mentioned before, BiGGEsTS is able to **extract the GO terms that annotate the genes** of the dataset. If such information is available, it can be obtained by clicking on a row of the table (left button of the mouse). The GO terms that annotate the corresponding gene are then displayed in a popup window, which additionally allows for selecting and copying the text content to paste on other documents.



BiGGEsTS - Biclustering Gene Expression Time Serie	s								- 0 x
Session Settings Help									
Datasets	Loading	Preprocessin	a Biclustering	Analyzing	Post-processing]			
Dataset 0	Constant V		5			1			
D Original 0	Matrix	xpression	Function						
riepiocessed matrices									
 Discretized Matrices 	Gen	s Names Ave	erage 000 mi He	at Shock 005	Heat Shock 015H	eat Shock 030H	eat Shock 060		
 Biclusters 	PPH2	2	0.16000000	-0.07000000	0.33000001	0.28000000	0.04000000		-
Post-Processed Biclusters	RBS1		0.38999999	-0.50999999	0.00000000	-0.15000001	-0.47000000		
	UFD2		-0.14000000	0.04000000	0.37000000	0.23999999	0.04000000		
	RPL3	ja	0.67000002	-0.04000000	-0.56000000	-1.36000001	-0.49000001		
	ARF1		0.82999998	0.15000001	-0.01000000	-0.10000000	0.16000000		
	NUS1		-0.47000000	-0.47000000	0.37000000	0.14000000	-0.14000000		
	SNF3		-0.64999998	0.19000000	0.88000000	0.49000001	-0.17000000		
	SEC3		-0.30000001	-0.31999999	0.44999999	-0.12000000	-0.17000000		
	YDL19	6w	-0.79000002	0.38000000	0.16000000	0.44000000	-0.27000001		
	ASF2		-0.19000000	0.63000000	0.50999999	-0.03000000	0.37000000		
	<u>66C1</u>		0.56000000	-0.28999999	-0.18000001	0.70999998	0.61000001		
	YDL19	90	-1.32000005	0.07000000	1.00000000	0.31999999	-0.49000001		
	MGT1		-0.12000000	0.0800000	0.0800000	0.47999999	0.51999998		
	TRM8		0.95999998	-0.62000000	-2.11999989	-1.09000003	0.36000001		
	MRPL	11	0.15000001	-0.34000000	0.04000000	0.15000001	0.30000001		
	ACK1		0.00000000	-0.14000000	0.01000000	-0.04000000	0.20999999		
	RTN2		-2.76999998	-1.74000001	1.55999994	0.28000000	-1.60000002		
	HEM3		0.08000000	-0.09000000	-0.69000000	-0.41999999	0.14000000		
	YDL20	6w	-0.67000002	-0.27000001	0.00000000	0.00000000	0.00000000		
	GLE1		0.00000000	-1.00000000		-1.60000002	0.00000000		
	NHP2		0.00000000						
	CWC2		0.23000000	1.00000000	2.57999992		1.00000000		
	UGA4		0.00000000	0.81000000		-1.78999996			
Original 0	YDL21	1c	-0.33000001		-0.31999999	0.00000000	1.74000001		
originaro	SHR3		-0.76999998	-0.30000001	-1.00000000	-0.15000001			
Organism: Saccharomyces cerevisiae	NOP6		0.00000000		-1.00000000	-1.00000000	-1.00000000		
Datatime: cDNA	PRR2		0.00000000		1.58000004		0.57999998		
butugper corer	GDH2		-0.30000001	-0.17000000	-0.36000001	-0.81000000	-0.61000001		
Number of genes: 6142	RRI1		0.00000000	-1.00000000	-0.57999998		1.52999997		
Number of time nointe: 5	TIM22		0.33000001	-0.31999999	1.13999999	-0.50999999	0.74000001		
reaction of three points, 5	YDL21	8w	-0.50999999	0.36000001	0.31000000	0.69999999	0.30000001		
Missing values: true	DTD1		-0.02000000	-0.15000001	-0.28999999	-0.20000000	0.49000001		-
Missings marked with value: -2.14748365E9					0.2000000	0.20000000			
	Values	Colors Sym	nbols Dendrog	gram					
					Analyz	e expression matr	ices.		

Table of values for the original matrix. Missing values are colored in dark blue.

After having loaded the time series expression data, BiGGEsTS is quite intuitive to use. For every operation there are some basic steps to follow: select some data matrix or bicluster in the dataset tree and then choose the functionality that you want to perform.

Functionality selection on the tree

A click with the right button of the mouse on the node in the tree will allow you to export the corresponding matrix, bicluster or group of biclusters as text. For matrices, only one text format is available, which includes a header explaining the contents (where each line is preceded by #) and then the tab separated information (basically the expression matrix with real or discrete values). In the case of biclusters or groups of biclusters, you may opt for a compact representation, where biclusters will be denoted by the subsets of genes and time points they contain, and their pattern when discrete values are used (Export as subset(s) of rows and columns). This format again includes a header explaining the contents and the data itself, and results in a single file for either an individual bicluster or a group. Alternatively, biclusters can be output as expression matrices, where each file corresponds to a bicluster and contains a tab separated table-like expression matrix: genes' names on the first column, time points' names on the first row and corresponding expression values (real or discrete, Export as real-valued matrix and Export as discrete-valued matrix). Another option is available to export each bicluster as a set of two files, one containing the names of the genes (one per line) and the other holding the expression values for all time points of each gene per line, separated by blank space characters (Export for Regulatory Snapshots).

Functionality selection on the tabbed pane

Additional functionalities can be selected on the tabs at the top (and bottom) of the window. The **input of gene expression data is always available** for every selected node in the dataset tree. Below is a summary of the additional main features, available



upon the selection of a given type of node in the dataset tree together with a given set of tabs:

Original or **Preprocessed** matrices – (i) Preprocessing; (ii) Biclustering (CC-TSB, only available if the matrix has no missing values); (iii) Analyzing -> Matrix -> Values; (iv) Analyzing -> Matrix -> Colors.

Discretized matrices – (i) Biclustering (CCC-Biclustering and e-CCC-Biclustering); (ii) Analyzing -> Matrix -> Values; (iii) Analyzing -> Matrix -> Colors; (iv) Analyzing -> Matrix -> Symbols.

Biclusters Group or **Post-Processed Biclusters Group** – (i) Analyzing -> Matrix -> Values; (ii) Analyzing -> Matrix -> Colors; (iii) Analyzing -> Matrix -> Symbols (only for groups of biclusters obtained from discretized matrices); (iv) Analyzing -> Expression -> Bicluster time-points; (v) Analyzing -> Expression -> Bicluster pattern (only for groups of biclusters obtained from discretized matrices); (vi) Analyzing -> Function -> Table; (vii) Post-Processing.

Bicluster – (i) Analyzing -> Matrix -> Values; (ii) Analyzing – Matrix -> Colors; (iii) Analyzing -> Matrix -> Symbols (only for biclusters obtained from discretized matrices); (iv) Analyzing -> Expression -> Bicluster time-points; (v) Analyzing -> Expression -> All time-point; (vi) Analyzing -> Expression -> Bicluster pattern (only for biclusters obtained from discretized matrices); (vii) Analyzing -> Function -> Table.

2.4. Preprocessing time series gene expression data

The next step we want to exemplify involves the preprocessing of the **Original** expression matrix. Select the **Preprocessing** tab (on the top of the window), after having the matrix that you want to preprocess selected on the dataset tree. Preprocessing includes the following techniques: (i) **gene filtering**, for filtering genes with missing values and only available for matrices with missing values; (ii) **missing values filling**, for filling missing data with real values; (iii) **data normalization**, to compensate for systematical differences between data measured by the several microarrays/conditions; (iv) **smoothing**, for reducing the impact of the noise in the analysis; and (v) **discretization**, for reducing the infinite set of real gene expression values to an adequate range of discrete values. Note that the data normalization, smoothing and discretization techniques can be applied to matrices with missing values with no previous treatment, because they use an appropriate approach to minimize the impact of the missing values. Once you have selected the preprocessing techniques and set their corresponding parameters, you can click the **Apply** button.

Upon the selection of any or several of the first four preprocessing options (filter, fill, normalize, smooth) and disabling of the discretization one, only a **Preprocessed** node is added to the dataset tree. If the discretization is enabled, an additional **Discretized** node is also created and added to the dataset tree. When selecting more than one preprocessing option, the several options are applied one at a time by BiGGEsTS to the gene expression data following the order of the options displayed, from top to bottom, in the preprocessing panel.

Session Settings Help	
Datasets Biclustering Analyzing Post-processing	
P C Dataset 0	
Original 0	
Preprocessed Matrices	
Discretized Matrices Filter	
Biclusters O all genes with missing values	
Post-Processed Biclusters	
G genes with more than 20 % of missing values	
Fill missings	
with the average over the K-neighbors 4 (k value)	
O with the average over the same row	
	=
✓ Normalize	
by overall values	
The burgers of the magn is and still in	
Smooth with window	
Original 0	
Organism: Saccharomyces cerevisiae	
Discretization Technique (Variation Detween une points (5 symbols)	
Number of genes: 6142	
Number of time points: 5	
Missing values: true	1
Missings marked with value: -2.14748365E9 Apply Apply	
	-

Preprocessing panel displaying the preprocessing options.

The names of the preprocessing steps are quite explicit, but we also list them here, including a brief description of their parameters:

Filtering of missing values

Includes two options for removing genes (rows of the matrix) with missing values: (i) all genes with missing values eliminates all rows which contain missing values; (ii)



genes with more than x % of missing values eliminates all rows whose percentage of missing values exceeds the value of x.

Filling of remaining missings

Provides three alternatives to fill the values missing from the expression matrix with: (i) a given value, which has to be typed in the corresponding text field; (ii) the average of the values of the k-neighbor cells of the same gene (row); (iii) the average over all the values of the same gene (row).

Normalization

Normalizes the expression values. They can be normalized altogether using the (i) **by overall values** option to mean 0 and standard deviation 1 or to a given mean and standard deviation, which have to additionally be specified in the corresponding text fields; or by row using the (ii) **by gene** option to mean 0 and standard deviation 1.

Smoothing

Acts like a low-pass filter for attenuating the negative effect of outliers; requires a parameter: the **length of a window** of neighbors to consider when computing the new value for substituting an outlier (note that you must provide an odd value, since the window includes the outlier value).

Discretization

Provides a number of different discretization techniques for replacing each absolute expression value by a symbol of a given alphabet. Alphabets of two or three symbols are the most common, containing the symbols {D, U} and {D, N, U}, respectively, where D means down-regulation, N is no-regulation and U means up-regulation. A brief description of the available methods follows. For more details see [1].

A first group of methods may be applied to the **overall values** of the matrix or **by gene** and computes the corresponding discrete value for each real element based on its absolute expression value:

- (i) Expression average is a binary discretization (alphabet {D, U}). Each element of the expression matrix is transformed into a D, if its absolute value is lower than the mean, or into a U otherwise. The mean value can be computed using all the expression values in the matrix or by gene and is calculated by summing all the values and dividing by their number.
- (ii) Mid-range is also a binary discretization, similar to expression average, except for the fact that the mid-range expression value is used as the criterion for choosing the symbol, instead of the mean. The mid-range is obtained by subtracting the minimum expression value to the maximum and dividing by two.
- (iii) **Max-minus percent-max** is also a binary discretization, similar to the previous two. In this case, the criterion for choosing the discrete symbol is computed as the difference between the maximum expression value and a given percentage of it. Such percentage must be specified by the researcher in the application.
- (iv) **Equal frequency** can be applied with an alphabet containing an arbitrary number of symbols. It selects the symbols in such a way that, after discretization, each symbol contains the same number of occurrences in the matrix.



- (v) Equal width can also be applied to an alphabet containing an arbitrary number of symbols. It divides the range of expression values, between the minimum and the maximum values in the matrix, into as many equal width intervals as the number of symbols in the alphabet. Then, every expression value in the matrix is substituted by the corresponding symbol.
- (vi) Expression mean and standard deviation uses an alphabet of three symbols {D, N, U} and a parameter alpha defined by the researcher. Symbol D is used to replace the expression values below the difference between the mean value and the product of alpha and the standard deviation. U is used for expression levels higher than the sum of the mean value and the product of alpha and the standard deviation and the product of alpha and the standard deviation. N is used for the remaining expression values.

The second group of methods computes the discrete values based on the variation of the expression level between every pair of consecutive conditions:

- (vii) **Transitional state discrimination** uses a binary alphabet {D, U}. In a first step, each value in the matrix is normalized to a z-score, computed as the difference between the expression value and the mean then divided by the standard deviation value. Each element in the matrix is then replaced by an U if the difference between its z-score and the z-score in the same row and previous condition exceeds 0. Otherwise, it gets the symbol D.
- (viii) **Variation between time points** can be used with an alphabet of two or three symbols, {D, U} or {D, N, U}, respectively. In the binary case, a parameter alpha must be specified by the researcher. A threshold is calculated as the product of alpha and the standard deviation of the expression values of all genes in time point 0. Then, each element in the matrix is replaced by a U if the difference between its expression value and the value in the same row and previous condition exceeds the computed threshold. Otherwise, it gets the symbol D. In the case of the three-letter alphabet, the threshold is directly chosen by the researcher in the application. Each element in the matrix is then replaced by a U if the difference between its expression value and the value and the value in the same gene and previous time point are higher than the threshold, by a D if such difference is lower than the symmetric of the threshold value, and N otherwise.

4 BiGGEsTS - Biclustering Gene Expression Time S	Series	×
Session Settings Help		
🗂 Datasets	Loading Preprocessing Bickustering Analyzing Post-processing	
P C Dataset 0	Matrix Expression Conction	
- Original 0	MININ LAPICEMENT INFERENCE	
Preprocessed Matrices		-
 Discretized Matrices 		
Discretized 0		
Diclusters	22 EL #2 EL	
Best Brassesed Biskusters	And Market B. B. B.	
- PostProcessed Diclusiers		
	VP38	
	who law who law who law y	
	SSAI SSAI	
	ERP2	
	F UNDA	
	0F0/0	
	SWC3	
	CYSS CYSS	
	SYNG SYNG	
	NTGI	
	P5x1	
	74210	
	4751	
	CCR4 CCR4	
Discretized 0	FUN26	
	9,472	
Parent node: Preprocessed 0		
Number of genes: 6089	M408	
Number of time points: 5	DR12	
	YALICINE	
Organism: Saccharomyces cerevisiae	PRI2	
Discratization technique:	and the second s	
variations between time points		
parameter: threshold - 1.0	PDPAt Contraction of the second	*
Alphabet: (D. N. U)	Values Colors Symbols Dendrogram	

Heatmap of a discretized matrix (obtained from the discrete symbols instead of the real values). You may check which symbol is actually contained in each cell by moving the mouse over it (text tip).



2.5. Biclustering time series expression data

Accessing the **Biclustering** tab, you are able to choose the biclustering algorithm and parameterize it.

🛎 BiGGEsTS - Biclustering Gene Expression Time Serie	s	×
Session Settings Help		
Detaset Dataset Original Orig	Loading Preprocessing Biclustering Analyzing Post-processing Biclustering Algorithms CCC-Biclustering e-CCC-Biclustering CCC-Biclustering e-CCC-Biclustering isign changes time lags [] (maximum time lag) gene shifts [] (m of shift(s))	
Discretized 0 Parent node: Preprocessed 0 Number of genes: 6089 Number of time points: 5 Organism: Saccharomyces cerevisiae Discretization technique: variations between time points parameter. threshold: -1.0 Abbabet: ON 10.0	Αρρίγ	
	Editation	÷

CCC-Biclustering with the default parameterization.

BiGGEsTS integrates three biclustering algorithms: one for real data with no missing values, CC-TSB-Biclustering; and two for discretized matrices that are also able to compute in the presence of missing values, CCC-Biclustering and e-CCC-Biclustering. In our example, we will apply the CCC-Biclustering algorithm to the previously mentioned **Discretized** matrix. We select the CCC-Biclustering algorithm, take the default parameterization and click the **Apply** button. Next, we also apply the CCC-Biclustering algorithm, but this time enabling the sign-changes option.

The available biclustering algorithms and parameterizations are:

CCC-Biclustering – for finding biclusters with exact expression patterns in discrete data; when in the presence of missing values, the algorithm uses an appropriate approach for disregarding them from the analysis of the gene expression values.

e-CCC-Biclustering – for finding biclusters with approximate expression patterns (maximum of *e* errors) in discrete data; required parameters: the **maximum number of errors** allowed per pattern; when activated, the **restricted errors** variation considers as valid errors only the substitutions of symbols which are on a given neighborhood in the alphabet of discretization. When in the presence of missing values, the algorithm may follow one of two approaches: ignore them, as in the case of the CCC-Biclustering algorithm, or consider them as errors. The default behavior is to ignore the missing elements.

Common variations of both CCC-Biclustering and *e*-CCC-Biclustering:



- gene-shifts, for grouping in biclusters genes with similar expression evolutions, but at different expression levels; parameter: the number of shifts, that is, of different expression levels to consider;
- (ii) **sign-changes**, for including in biclusters genes with symmetric expression patterns;
- (iii) time-lags, for considering in the same biclusters genes with similar expression evolutions, but starting at different points in time, in a predefined order as generated by temporal programs; parameter: the maximum time lag allowed between expression patterns;

Note: gene-shifts cannot be combined with sign-changes or time-lags.

CC-TSB – for finding biclusters with approximate expression patterns in real data; parameters: (i) **delta**, the threshold for the MSR of each bicluster; (ii) **alpha**, the ratio between the MSR of a row and the MSR of the matrix; (iii) the **number of biclusters to extract**; (iv) the **maximum number of iterations**.

Note that the computation may take time, especially when a large matrix is involved. Sometimes, if the dimension of the matrix is really demanding, computation may also end up aborting, usually due to memory issues.



2.6. Visualizing biclustering results

Upon a successful application of a biclustering algorithm, a group of biclusters is added to the tree, the *Analyzing*, *Matrix* and *Colors* tabs are selected and the colored matrices of the first 5 biclusters of the group are displayed by default. This number can be changed using the combo box available in the *Colors* panel. The matrices of values and symbols are accessed by selecting the *Values* or *Symbols* tab, respectively.

BiGGEsTS - Biclustering Gene Expression Time Series								1 X
Session Settings Help								
🗖 Datasets	Loading	Preprocessing Biclusteri	ng Analyzing	Post-processing				
🕈 🔚 Dataset 0	Matrix	xpression Function						
Original 0								
Preprocessed Matrices								E
Biclusters	-							
← □ Biclusters Group 0	service a service of the servic							
Post-Processed Biclusters								
_	Bicluster 1							
		2 P						
		as h						
		0 mi						
		8 8						
		\$ \$						
	Gene	lamas II I						
	ERP2							
	SWC							
	CYS3 DEP1							
	CCR4							
	MAK1							
	MTW1							
	RBG1							
Biclusters Group 0	YAL0							
Parent node: Discretized 0	ERV4							
Number of biclusters: 167	YALD-							
	YALD							
Datatype: discrete symbols	YALD	-b						
Organism: Saccharomyces cerevisiae	YALD							
	ERP1							
Biclustering algorithm: CCC-Biclustering	SEN3							Ţ
	Values	Colore Sambole Dendr	ogram					
	values	COIDES COMING COMMENT	ogram					

Heatmap (matrix of colors) of the first 5 CCC-biclusters (only the first matrix is visible).



BiGGEsTS enables the display of miniaturized expression charts of biclusters in a group.

Miniatures of the time-points charts of the first 24 CCC-biclusters identified using the default CCC-Biclustering algorithm.



The expression charts and pattern charts are available upon selection of the *Analyzing*, *Expression*, *Bicluster time-points* tabs (in this order) and the *Analyzing*, *Expression*, *Bicluster pattern* tabs (in this order), respectively. Note that when biclusters are large, that is, composed of many genes and/or conditions, or when you are trying to display the charts of a considerable number of biclusters in a group, BiGGEsTS may take some time to draw all these data. By default, only the first 12 biclusters are displayed. You may change this number in the corresponding combo box on the top of the panel.



Miniatures of the pattern charts of the first 24 biclusters identified using the default CCC-Biclustering algorithm.



Miniatures of the pattern charts of the first 24 biclusters identified by CCC-Biclustering with sign-changes (anticorrelated patterns).



To access the individual biclusters you must open the folder of the group of biclusters in the dataset tree (try clicking the "key" on the left of the group folder). Once the group is opened you may select a specific bicluster and view its corresponding information. Matrices of values, colors and symbols are displayed in a similar fashion to the one used for original, preprocessed and discretized matrices. Bicluster expression charts, all time-points expression charts and bicluster pattern charts are upon selection of the Bicluster time-points, All time-points and Bicluster pattern tabs, respectively (after selecting *Analyzing* and *Expression* tabs).

🕌 BiGGEsTS - Biclustering Gene Expression T	ime Series							x
Session Settings Help								
Datasets	Loading Preproce	ssing Biclustering	Analyzing	Post-processing				
P 🗖 Dataset 0	Contra Vermini	- Kronstein						
C Original 0	Matrix Expression	n Function						
Proprocessed Matrices								
Discretized Matrices								
Distretized matrices	Genes Names	Heat Shock 005 Hea	t Shock 015	Heat Shock 030 He	at Shock 060			_
Biclusters	VPS8	-0.28209308	1.17700911	1.17700911	-0.99543190			^
P Biclusters Group u	SSA1	-0.44083309	1.51842499	0.68846160	-0.40001521			
Bicluster 1	MDM10	-0.05809365	1.32721603	0.47815531	-0.01340622			
 Bicluster 2 	SYN8	0.52739310	1.21/06104	0.48682433	-1.622/4814			
 Bicluster 3 	PM12	-1.45668566	1.67597187	0.10964306	-0.12530626			
- Bicluster 4	FRIZ	0.42893973	1.48370981	0.32346278	-1.04773808			
Disluster 5	RNC4	-0.22070412	0.94776732	1.22220652	-0.39013524			
- Bicluster 5	DDD45	-0.20164933	1.06500450	0.76000540	-0.48443839			
Bicluster 6	POP5	-1.04112308	1.00556695	1 25607494	1 21100545			
 Bicluster 7 	FLINIA	-0.07037003	1.48561788	0.76997239	-0.43656316			
 Bicluster 8 	CLN3	-0.0000006	0.49157578	0.92853200	-1.91168368			
- Bicluster 9	GEM1	0.25491968	0.71185118	1.04853749	-0.20201179			
Disluster 10	YAL049c	-0.94648015	1.37076426	0.97911727	-0.32637247			
Dicluster ru	OAF1	0.08951437	1.34271538	0.80562931	-0.98465800			
- Bicluster 11	ACS1	0.08392486	0.79546154	1.28806388	-0.66410089			
 Bicluster 12 	PEX22	0.55820847	1.31254423	0.44505811	-0.95046300			
 Bicluster 13 	GPB2	-0.75064123	1.60674429	0.73823386	-0.93675047			
Bicluster 14	CNE1	-0.67815799	1.67544889	0.31913313	-0.03989172			
Bicluster 15	YAL061w	0.58798593	1.28332210	0.45989767	-0.96737105			
Dictaster 15	GDH3	-0.16746682	1.76484275	0.28340539	-1.00480092			
Biciuster 16	RFA1	-1.79641092	1.09780669	0.69860417	-0.19960120			
	PAU7	0.45626888	1.01119053	0.45626888	-0.03699478			
Bicluster 14	YAR023c	0.26632267	1.19369614	0.40899557	-0.04280184			
	UIP3	0.07106446	1.52619386	0.52790743	-0.80878121			
Parent node: Biclusters Group 0	YAR029w	0.27251071	1.25818777	0.59140617	-0.45225182			
Number of genes: 1091	YAR070c	0.25663120	0.60089254	1.19552600	-0.30670562			
Number of time points: 4	SCIT	-1.02560937	0.81728250	1.53841424	-0.78523225			
SIZE: 4364	YBLU120	-1.21430119	1.72507117	0.31950881	-0.191/4124			
Detetmen discusts such als		-0.59791627	1.20137260	0.72494200	-0.64705091			
Organiamy Casebaramyses estavisias	DED1	-0.47088000	1.26000175	0.72104390	-0.20722701			-
organism. Saccharomyces cerevisiae	1611	-0.202201111	1.555091751	0.756566671	-0.257257011			
	Values Colors	Symbols Dendroar	am					
	8			A	the contraction	an advise a s		

Analyze expression matrices

Table of values of the bicluster 14 identified using the default CCC-Biclustering algorithm.



Table of colors of the bicluster 14 identified by the default CCC-Biclustering algorithm.





Table of symbols of the bicluster 14 identified using the default CCC-Biclustering algorithm.



Expression chart showing the evolution of the expression level of the genes in the CCC-bicluster 16 (obtained using the CCC-Biclustering on another matrix resulting from the discretization of unnormalized expression data) along the corresponding conditions of the bicluster. It is possible to normalize the expression levels by checking the *Normalize to mean 0 and std 1* checkbox.





Expression chart showing the evolution of the expression level of the genes in the CCC-bicluster 16 along all the conditions of the dataset. It is possible to normalize the expression level by checking the *Normalize to mean 0 and std 1* checkbox.



Pattern chart displaying the expression pattern of the CCC-bicluster 16.

All charts, except the miniature ones, allow **saving as image**, **printing** and **zooming**. These options can be accessed by clicking the right button of the mouse on the chart.



2.7. Analyzing GO terms

BiGGEsTS automatically extracts the GO terms that annotate the genes of the dataset when the Gene Ontology files are available. For biclusters and groups of biclusters, it is additionally possible to perform functional enrichment using the term-for-term analysis, which computes the statistical significance of each GO term that annotates the genes in the biclusters. The term-for-term analysis is available by selecting the *Analyzing*, *Function* and *Table* tabs in this order.

🕌 BiGGEsTS - Biclustering Gene Expression Ti	ime Series	×						
Session Settings Help								
Datasets	Loading Preprocessing Biclustering Analyzing Post-processing Matrix Expression Function							
Original Original	Function Analysis							
	P-Value 0.01							
	Organism: Saccharomyces cerevisiae							
	☑ Use available GO frontend							
	GO files to use in Function Analysis							
	Gene Ontology File							
	C:\Program Files\BIGGEsTS\GO_Files\gene_ontology.obo							
	Browse Download							
	Gene Association File							
	C1Program Files\BIGGEsTSIGO_Files\gene_association.sgd							
	Browse Download							
Biclusters Group 0								
	Apply							
Parent node: Discretized 0 Number of biclusters: 167								
Detetimer disersts symbols								
Organism: Saccharomyces cerevisiae								
Biclustering algorithm:								
CCC-Biclustering	Table Significance Graph							
	Analyze GO terms and functional enrichment							

Parameters for the term-for-term analysis.

The required parameters for term-for-term analysis are: a **p-value**, the threshold below which the Bonferroni corrected p-values computed for the GO terms are considered statistically significant; the general **ontology** and specific organism **annotation files**. If these files are available at the proper location (the GO_Files directory within the BiGGEsTS installation directory), their corresponding file paths are displayed in the text boxes. Also note that when this is the case, most likely BiGGEsTS has already used these files to extract the GO terms that annotate the genes in a previous step. When such happens, the **Use available GO frontend** check box appears selected, which means that the annotations have already been retrieved and the term-for-term analysis can use them, avoiding to repeat the parsing of the Gene Ontology files. This accelerates the computation process of the term-for-term analysis.

Additionally, BiGGEsTS enables to download the files from the Gene Ontology repository by clicking the **Download** button below the corresponding file path text box.

The results of the term-for-term analysis are displayed in a table. Significant and highly significant terms are highlighted in green. The threshold for statistical high significance is always 0.01. The one for statistical significance is also 0.01 by default, but can be changed to a different value.



BiGGEsTS - Biclustering Gene Expression Time	ne Series								×
Session Settings Help									
Datasets Dataset 0 Original 0	Loading Preprocessing Matrix Expression Fun	Biclustering /	Analyzing Pos	t-processing					
Preprocessed Matrices Discretized Mat	Values calculated usi	ing p-value = 0.01	Recalculate						
	Bicluster ID	# Genes #	Time points	Best p-value	Best corrected # Sign	ificant ter # Highl	v sia ter Sia	nificance thr# Sig	terms (thr
	1	2224 1488	2	0.00000000	0.00000000	6.0 2.0	30.0 3.0	0.01000000	30 🔺
	3	295	3	0.00004855	0.07679824	0.0	0.0	0.01000000	0 ==
	4	582	3	0.00000002	0.00004160	1.0	7.0	0.01000000	7
	5	611	3	0.00025290	0.50099784	0.0	0.0	0.01000000	0
	6	2525	2	0.00000000	0.00000000	25.0	45.0	0.01000000	45
	7	296	3	0.00000072	0.00118183	1.0	1.0	0.01000000	1
	8	13	4	0.00426579	0.68679166	0.0	0.0	0.01000000	0
	9	100	4	0.00002007	0.03213314	1.0	11.0	0.01000000	11
	11	1515	*	0.000000000	0.00000107	12.0	45.0	0.01000000	45
	12	104	4	0.00031231	0.28545365	0.0	0.0	0.01000000	0
	13	320	4	0.00001919	0.03188263	1.0	0.0	0.01000000	0
	14	1091	4	0.00000000	0.00000000	17.0	35.0	0.01000000	35
	15	714	3	0.00000000	0.00000239	10.0	15.0	0.01000000	15
	16	387	4	0.00003550	0.06208146	0.0	0.0	0.01000000	0
	17	217	4	0.00000752	0.01062131	3.0	0.0	0.01000000	0
Biclusters Group 0	18	110	4	0.00002265	0.02210875	1.0	0.0	0.01000000	0
	19	1727	2	0.00001050	0.02876052	2.0	0.0	0.01000000	0
Parent node: Discretized 0	20	431	3	0.00015353	0.25239581	0.0	0.0	0.01000000	0
Number of biclusters: 167	21	15	4	0.00345512	0.92597109	0.0	0.0	0.01000000	0
	22	3	5	0.00491964	0.30501750	0.0	0.0	0.01000000	0
Datatype: discrete symbols Organism: Saccharomyces cerevisiae	23	12	5	0.00393797	0.96480340	0.0	0.0	0.01000000	
Biclustering algorithm: CCC-Biclustering									
	Table Significance Graph								
				Analyze GO	terms and functional enr	ichment			

Results of the term-for-term analysis applied to the genes in the group of biclusters. It's possible to recalculate the significance of the GO terms using different Gene Ontology files or p-value threshold by clicking the **Recalculate** button. Clicking a row of this table will select the corresponding bicluster in the dataset tree and redirect the content of the panel to the results of the following selection of tabs: Analyzing, Matrix, Colors.

BIGGESTS - Biclustering Gene Expression T	ime Series					_ = ×					
Session Settings Help											
Datasets	Loading Preprocessing	Biclustering Analyzing Post-processing									
🕈 🗂 Dataset 0	Matrix Exproscion	unction									
- Original 0	Matrix Expression 1	uncuon									
Preprocessed Matrices											
Discretized Matrices											
- Biclusters											
C Richusters Group 0											
Pictusters of oup of	Values calculated using p-value = 0.01 Recalculate										
Bicluster 1	Number of Genes i	Number of Genes in Deputation: 6099									
- Bicluster 2											
- Dicluster 3	Number of Genes i	n Bicluster: 1091									
 Bicluster 4 		1									
- Bicluster 5	GO Term ID	GO Term Name	#A. Population G.	#A. Set G.	P-Value	Corrected P-Value					
Bicluster 6	00:0006085	acetyl-CoA biosynthetic process	4	2	0.14964238	1.0000000					
Distustor 7	60.0006090	pyruvate metabolic process	41	14	0.00933387	0.0000000					
- Bicluster /	60:0006091	lastate metabolic process	509	130	0.04210202	1.00000000					
- D Bicluster 8	00:0000000	aluconeogenesis	31	12	0.04510205	1.00000000					
- 🗋 Bicluster 9	00:0006139	nucleobase nucleoside nucleotide and nucleic acid m	1441	183	1 0000000	1.00000000					
 Bicluster 10 	GO:0006140	regulation of nucleotide metabolic process	3	2	0.08476067	1.00000000					
- Bicluster 11	GO:0006112	energy reserve metabolic process	38	23	0.00000001	0.00001431					
Bicluster 12	GO:0006113	fermentation	16	3	0.56667113	1.0000000					
Dictuster 12	GO:0006116	NADH oxidation	10	4	0.08694948	1.0000000					
Bicluster 13	GO:0006118	electron transport	87	30	0.00014698	0.39171028					
Bicluster 14	GO:0006119	oxidative phosphorylation	38	24	0.0000000	0.0000189					
 Bicluster 15 	GO:0006120	mitochondrial electron transport, NADH to ubiquinone	1	1	0.17917556	1.0000000					
Bicluster 16	GO:0006121	mitochondrial electron transport, succinate to ubiquino	4	4	0.00102601	1.0000000					
	GO:0006122	mitochondrial electron transport, ubiquinol to cytochro	8	7	0.00003940	0.10500936					
Bicluster 14	G0:0006123	mitochondrial electron transport, cytochrome c to oxygen	5	3	0.04310203	1.0000000					
	00:0016925	protein sumoyiation	10	3	0.26031396	1.0000000					
Parent node: Biclusters Group 0	00:0006038	cell wall chitin protobalia process	12	3	0.30724958	1.0000000					
Number of genes: 1091	60:0000037	response to cold	interation processor 7	2	0.42001414	1.00000000					
Number of time points: 4	00:0003403	chitin catabolic process	siment process 7	1	0.50557215	1.00000000					
Size: 4364	60:0009408	response to heat	23	6	0.21851493	1 00000000					
			201	0.							
Datatype: discrete symbols											
Organism: Saccharomyces cerevisiae											
	Table Simificance Cra	nh									
	Table Significance Gra	hu									
	Analyze bicluster										

Results of the term-for-term analysis applied to the genes in the bicluster 14.

When the term-for-term analysis reveals significant functions (highlighted in green), the *Significance Graph* tab is enabled and the corresponding panel displays a graph with the ontology of the GO terms. The significant terms are highlighted in green, yellow or purple depending on which of the main GO term they specialize from. You may further **zoom** the graph or **save** it as a **PNG** or **SVG** image.





Graph of significant GO terms extracted by the term-for-term analysis for the genes in the bicluster 14.

2.8. Post-processing groups of biclusters

The groups of biclusters can be post-processed (*Post-Processing* tab). This operation consists in filtering and/or sorting the biclusters of a group of biclusters based on specific criteria, available for selection in the post-processing panel. A short description of each filtering/sorting criterion and parameters is provided below the corresponding list in the panel, upon the selection of a given item. To apply the post-processing operations to the data click on the *Apply* button.

BiGGEsTS - Biclustering Gene Expression Ti	ime Series	۲
Session Settings Help		
Datasets Datasets Dataset Dataset Dataset Dataset Proprocessed Matrices Discretized Matrices Discretized Matrices Discretized Matrices Discretized Matrices Packaters Packaters Packaters Packaters Post-Processed Biclusters Discretized Bicluster Discretized	Loading Preprocessing Biclustering Analyzing Post-processing Image: State of the points genes, time points genes, time points trivial, ACV Sorting Criteria ACV ARV Image: State of trivial genes, time points trivial, ACV MSR ARV. ACV Image: State of trivial, ACV ARV. ASR MSR Image: State of trivial, ACV ARV. ASR NSR Image: State of trivial, ACV NSR NSR Image: State of trivial, ACV NUMBER of genes NSR Image: State of trivial, ACV NUMBER of genes NSR Image: State of trivial, State Variance Variance Image: State of trivial, State Variance Variance Image: State of trivial, State Sorting Criteria trivial Variance Image: State of trivial, State Variance Variance Image: State of trivial trivial, State Variance Variance Image: State of trivial	
Biclusters Group 0	Filtering technique description and parameters filter biclusters by pattern w/ columns:	
Parent node: Discretized 0 Number of biclusters: 167	Markov chain order 1 p-value threshold 0.01	
Datatype: discrete symbols Organism: Saccharomyces cerevisiae		
Biclustering algorithm: CCC-Biclustering	Аррју	

Post-Processing panel: available filtering and sorting options.

Below is a list of the available post-processing criteria for filtering and/or sorting the biclusters in a group of biclusters (if you need a formal definition of the metrics used in



post-processing techniques please see the section with formal definitions, in the end of this document):

Filtering

Filtering processes the group of biclusters and removes biclusters which: (i) are **trivial**, that is, are composed by a single row or a single column; contain (ii) a **number of genes** less than a given threshold, (iii) a **number of conditions** less than a given threshold, (iv) **a number of genes** and **a number of conditions** less than two corresponding thresholds; have (v) an **average column variance (ACV)** greater than a given threshold, (vi) an **average row variance (ARV)** less than a given threshold, (vii) an **ACV** greater than and an ARV less than two given thresholds, (viii) a **MSR** greater than a given threshold.

BiGGEsTS can also eliminate biclusters: whose (ix) **size**, the number of genes times the number of conditions, is less than a given threshold; which satisfy a combination of the previously mentioned criteria, such as the thresholds for their (x) **size**, **ARV** and **ACV**, (xi) **numbers of conditions and genes**, **ARV** and **ACV**, (xii) **numbers of conditions and genes**, **ARV** and **ACV**, (xii) **numbers of conditions and genes**, **ARV**, and **ACV**, (xii) **numbers of conditions and genes**, **ARV**, **ACV** and **MSR**; which are very similar to other biclusters in the group, according to a given **percentage of similarity** on the dimension(s) of (xiii) the **genes** or (xiv) both **genes and conditions**; which (xv) are **constant**, that is, have no variation of the level of expression from one condition to another.

Two additional options for removing biclusters with **non significant expression patterns** are available for biclusters with discrete data, both using Markov chains to assess the statistical significance of the pattern and computing a p-value based on the (xvi) **overall** or (xvii) **column-wise** background probability of the occurrence of the pattern.

Sorting

Sorting the biclusters in a group reorganizes the biclusters according to: their values of (i) ACV, (ii) ARV, (iii) MSR; the difference between their (iv) ARV and ACV, (v) ARV and MSR. You can also sort biclusters by their number of (vi) genes or (vii) conditions, (viii) size or (ix) variance. These criteria allow biclusters to be sorted either in decreasing or decreasing order of the considered value(s).

Additional **sorting by the best p-value** obtained for the GO terms that annotate the genes of each bicluster in the group, both (x) **standard** and (xi) **corrected** for multiple testing, is also available. Groups of biclusters with discrete data can further be sorted based on the **significance of their expression pattern**, measured by a p-value computed using Markov models and based on the (xii) **overall** or (xiii) **column-wise** background probability of the occurrence of the pattern.

After post-processing a group of biclusters, a new group of biclusters is generated, which we call a **Post-Processed Biclusters Group**, and added to the dataset tree. This group is similar to a group of biclusters and allows for the very same functionalities. The resulting biclusters in a post-processed group can also be found in the original group. Because biclusters are just filtered and/or sorted, their data does not change in



relation to the original data. However, as a consequence of the filtering operation, the post-processed group may not contain all the original biclusters. Moreover, as a consequence of the sorting operation, the post-processed biclusters may also not be in the same order as in the original group. To address this, the post-processed biclusters are given new identifiers, following their order in the post-processed group, but they also maintain the original identifiers in parenthesis, (), for easier tracking the bicluster in the original group.



3. Formal Definitions

3.1. Expression matrix Let A be a |R| row by |C| column gene expression matrix defined by its set of rows (genes), R, and its set of columns (conditions), C. Let A_{ij} represent the expression level of gene i under condition j. Let A_{iC} and A_{Rj} denote row i and column j of matrix A, respectively.

3.2. Bicluster A bicluster is a submatrix A_{IJ} defined by $I \subseteq R$, a subset of rows, and $J \subseteq C$, a subset of columns. A bicluster with only one row or one column is called trivial.

3.3. COHERENT BICLUSTERS WITH CONTIGUOUS COLUMNS

CC-TSB algorithm Given an expression matrix A with absolute expression values, the number of biclusters to extract, an empirical threshold α for the ratio between the MSR of each row and the MSR of all the rows in the bicluster, an upper limit δ for the MSR of a bicluster and a maximum number of iterations, the goal is to identify and report a predefined number of biclusters with a good MSR value. Due to its heuristic nature, this approach is not guaranteed to find the optimal set of biclusters.

3.4. CONTIGUOUS COLUMN COHERENT BICLUSTERS WITH EXACT PATTERNS

CCC-Bicluster A contiguous column coherent bicluster (CCC-Bicluster) A_{IJ} is a subset of rows $I = \{i_1, ..., i_k\}$ and a subset of contiguous columns $J = \{r, r + 1 ..., s - 1, s\}$, such that $A_{ij} = A_{lj}$ for all rows $i, l \in I$ and columns $j \in J$. Each CCC-Bicluster defines a string S that is common to every row in I for the columns in J: the bicluster pattern.

Maximal CCC-Bicluster A CCC-Bicluster A_{IJ} is maximal if no other CCC-Bicluster exists that properly contains it, that is, for every other CCC-Bicluster A_{LM} , $I \subseteq L \land J \subseteq M \Rightarrow I = L \land J = M$. In other words, A_{IJ} is maximal if it cannot be extended with more genes or more contiguous columns.

CCC-Biclustering algorithm Given a discretized expression matrix A, the goal is to identify and report all maximal CCC-Biclusters.

3.5. <u>CONTIGUOUS COLUMN COHERENT BICLUSTERS WITH APPROXIMATE</u> <u>PATTERNS</u>

e-Neighborhood The *e*-Neighborhood of a string *S* of length |S|, defined over the alphabet Σ with $|\Sigma|$ symbols, N(e, S) is the set of strings S_i such that $|S| = |S_i|$ and $Hamming(S, S_i) < e$, where *e* is an integer such that $e \ge 0$. This means that the Hamming distance between *S* and S_i is no more than *e*, that is, we need at most *e* substitutions to obtain *S* from S_i .



e-CCC-Bicluster A contiguous column coherent bicluster with *e* errors per gene (*e*-CCC-Bicluster) is a CCC-Bicluster A_{IJ} where all the strings S_i that define the expression pattern of each gene in *I* are in the *e*-Neighborhood of an expression pattern *S* that defines the *e*-CCC-Bicluster: $S_i \in N(e, S), i \in I$. The definition of 0-CCC-Bicluster is equivalent to that of a CCC-Bicluster.

Maximal *e*-**CCC**-**Bicluster** An *e*-CCC-Bicluster A_{IJ} is maximal if it is row-maximal, leftmaximal and right-maximal. This means that no more rows or contiguous columns can be added to *I* or *J*, respectively, maintaining the coherence property in the definition of *e*-CCC-Bicluster (above).

e-CCC-Biclustering algorithm Given a discretized expression matrix A and the integer $e \ge 0$, the goal is to identify and report all maximal *e*-CCC-Biclusters $B_k = A_{I_k I_k}$.

3.6. <u>SIGN-CHANGES (ANTICORRELATED PATTERNS)</u>

Opposite expression pattern Given the alphabet Σ with $|\Sigma|$ sorted in lexicographic order and the expression pattern $S = [S[1] \dots S[|S|]]$, we define its opposite pattern S^{-1} as follows: $S^{-1} = [S[1]^{-1} \dots S[|S|]^{-1}]$, where $S[j]^{-1} \in \Sigma, j \in \{1, \dots, |S|\}$ is the opposite symbol of the symbol $S[j] \in \Sigma$. Assuming that $S[j]^{-1}$ corresponds to the symbol at position p in Σ , $\Sigma[p]$, we compute $\Sigma[p]^{-1}$ as follows: $\Sigma[p]^{-1} = \Sigma[|\Sigma| - p + 1], \forall p \in \{1, \dots, |\Sigma|\}$. When $|\Sigma|$ is odd, $\Sigma[|\Sigma|/2]^{-1} = \Sigma[|\Sigma|/2]$.

CCC-Bicluster with sign-changes A CCC-Bicluster with sign-changes A_{IJ} is a CCC-Bicluster such that $A_{ij} = A_{lj}$ or $A_{ij} = A_{lj}^{-1}$ for all rows $i, l \in I$ and columns $j \in J$, where A_{lj}^{-1} is the opposite symbol (in Σ) of that in A_{lj} .

e-CCC-Bicluster with sign-changes An *e*-CCC-Bicluster with sign-changes A_{IJ} is an *e*-CCC-Bicluster where all the strings S_i that define the expression pattern of each of the genes in I are wither in the *e*-Neighborhood of the expression pattern S that defines the *e*-CCC-Bicluster, or in the *e*-Neighborhood of its opposite expression pattern S^{-1} : $S_i \in N(e, S)$ or $S_i \in N(e, S^{-1}), i \in I$.

3.7. <u>GENE-SHIFTS (SCALED PATTERNS)</u>

CCC-Bicluster with gene-shifts A CCC-Bicluster with gene-shifts A_{IJ} is a CCC-Bicluster where all the strings S_i that define the expression pattern S that defines the CCC-Bicluster, or one of the patterns resulting from shifting the expression pattern S K symbols up or K symbols down, where K is an integer and $K \in [1, ..., |\Sigma| - 1]$. This means $S = S_i \lor S_i \in S^{\uparrow} = \{S^{\uparrow 1}, ..., S^{\uparrow K}\} \lor S_i \in S^{\downarrow} = \{S^{\downarrow 1}, ..., S^{\downarrow K}\}, i \in I$.

e-CCC-Bicluster with gene-shifts An *e*-CCC-Bicluster with gene-shifts A_{IJ} is an *e*-CCC-Bicluster where all the strings S_i that define the expression pattern of each of the genes in I are either in the *e*-Neighborhood of the expression pattern S that defines



the *e*-CCC-Bicluster, or in the *e*-Neighborhood of the patterns resulting from shifting its expression pattern *S K* symbols up $S^{\uparrow} = \{S^{\uparrow 1}, ..., S^{\uparrow K}\}$ or *K* symbols down $S^{\downarrow} = \{S^{\downarrow 1}, ..., S^{\downarrow K}\}$, where *K* is an integer and $K \in [1, ..., |\Sigma| - 1]$. This means $S_i = N(e, S) \lor S_i = N(e, S^{\uparrow}) \lor S_i = N(e, S^{\downarrow}), i \in I$.

3.8. TIME-LAGS (TIME LAGGED PATTERNS)

CCC-Bicluster with time-lags A CCC-Bicluster with time-lags A_{IJ} is a CCC-Bicluster such that $A_{ij} = A_{lj_l}$ for all rows $i, l \in I$ and columns $j_l \in J_l$. J_l is the set of contiguous columns corresponding to one occurrence of pattern S in row l, S_l , such that $j_l = j + lag_l$ and $lag_l \in \{0, ..., |C| - 1\}$ is the time lag between S_l and the starting pattern.

Starting pattern (in CCC-Biclusters with time-lags) Given a pattern S and a set of rows I where it occurs at different time-lags, each of these occurrences is specified by a set of contiguous columns J_i . We consider the first (left most) occurrence of pattern S within all these occurrences specified by the contiguous columns in J_i , for all rows $i \in I$ as the starting pattern. This starting pattern is specified by a set of contiguous columns, $j \in J$, which corresponds to the set of contiguous columns where S occurs with time lag zero.

e-CCC-Bicluster with time-lags An *e*-CCC-Bicluster with time-lags A_{IJ} is an *e*-CCC-Bicluster defined by the expression pattern *S* such that $A_{ij} = A_{lj_l}$, where $A_{lj_l} = S$ or $A_{lj_l} \in N(e,S)$ for all rows $i, l \in I$ and columns $j_l \in J_l$. J_l is the set of contiguous columns corresponding to one occurrence of pattern *S*, or one pattern in N(e,S), in row l, S_l , such that $j_l = j + lag_l$ and $lag_l \in \{0, ..., |C| - 1\}$ is the time lag between S_l and the starting pattern.

Starting pattern (in *e*-**CCC-Biclusters with time-lags)** Given a pattern *S* in N(e, S) and a set of rows *I* where these patterns occur at different time-lags, each of these occurrences is specified by a set of contiguous columns J_i . We consider the first (left most) occurrence of pattern *S*, or a pattern in N(e, S), within all these occurrences specified by the contiguous columns in J_i , for all rows $i \in I$ as the starting pattern. This starting pattern is specified by a set of contiguous columns, $j \in J$, which corresponds to the set of contiguous columns where *S*, or a pattern in N(e, S), occurs with time lag zero.



3.9. POST-PROCESSING METRICS

Size

 $|R| \times |C|$

Variance

$$\frac{\sum_{i=1}^{|R|} \sum_{j=1}^{|C|} (A_{ij} - m_A)^2}{|R| \times |C|}, \qquad m_A = \frac{\sum_{i=1}^{|R|} \sum_{j=1}^{|C|} A_{ij}}{|R| \times |C|}$$

Average Column Variance (ACV)



Average Row Variance (ARV)



Mean Squared Residue (MSR)

$$\frac{\sum_{i=1}^{|R|} \sum_{j=1}^{|C|} \left(A_{ij} - \left(\frac{\sum_{k=1}^{|R|} A_{kj}}{|R|} \right) - \left(\frac{\sum_{l=1}^{|C|} A_{ll}}{|C|} \right) + m_A \right)^2}{|R| \times |C|}$$

Bicluster similarity

In order to compute the similarity measure between two biclusters, $B_1 = (I_1, J_1)$ and $B_2 = (I_2, J_2)$ we use the Jaccard Index. This score is used to measure the degree of overlap between two biclusters both in terms and conditions and is defined as follows:

$$\begin{split} S(B_1, B_2) &= S\big((I_1, J_1), (I_2, J_2)\big) = \frac{|B_1 \cap B_2|}{|B_1 \cup B_2|} = \frac{|B_{11}|}{|B_{01}| + |B_{10}| - |B_{11}|}, \text{ where } B_{11} = \{(i, j): (i, j) \in B_1 \land (i, j) \in B_2\}, \quad B_{10} &= \{(i, j): (i, j) \in B_1 \land (i, j) \notin B_2\} \text{ and } B_{01} = \{(i, j): (i, j) \notin B_1 \land (i, j) \in B_2\}, \text{ for the genes } i \in I_1 \cup I_2 \text{ and the conditions } j \in J \cup J_2. \text{ Similarly, the gene and condition similarities can be computed, respectively, as follows: } S(I_1, I_2) = \frac{|I_1 \cap I_2|}{|I_1 \cup I_2|} \text{ and } S(J_1, J_2) = \frac{|J_1 \cap J_2|}{|J_1 \cup J_2|}. \end{split}$$



Significance of a bicluster pattern

This is a superficial description of the computation of the significance of a bicluster pattern. For more details see [1].

The statistical significance of a CCC-Bicluster *B* is *p*-value(*B*), which is computed by obtaining the probability $P(p_B)$ of a random occurrence of the pattern of the bicluster, p_B , under the null hypothesis *k* times in *n* independent trials, where *k* is the number of genes in the bicluster minus 1 and *n* is the number of genes in the expression matrix minus 1. The null hypothesis assumes that the expression levels of the genes evolve independently.

The value of this probability is obtained by computing the *tail of the binomial distribution*, which gives the probability of an event with probability p_B occurring k or more times in n independent trials.

We use the simplifying assumption that the probability of occurrence of a specific expression pattern p_B , $P(p_B)$, is adequately modeled by a first order Markov Chain, with state transition probabilities obtained from the values in the corresponding columns in the matrix.

For a CCC-Bicluster with sign-changes, P is defined as: $P(p_B \cup p_B^{-1}) = P(p_B) + P(p_B^{-1})$, where p_B^{-1} is the opposite (symmetric) pattern of p_B .

For a CCC-Bicluster with time-lags, P is defined as: $P((p_B)_{\rightarrow}^{LAG}) = \sum_{lag|B} P((p_B)_{\rightarrow}^{lag})$, where $lag \in \{0, ..., |C| - 1\}$ and the values of lag are restricted to those occurring in B, lag|B.

For a CCC-Bicluster with time-lags, P is defined as: $P((p_B)^{LAG}_{\rightarrow} \cup (p_B^{-1})^{LAG}_{\rightarrow}) = \sum_{lag|B} P((p_B)^{lag}_{\rightarrow}) + \sum_{lag|B} P((p_B^{-1})^{lag}_{\rightarrow})$, where $lag \in \{0, ..., |C| - 1\}$ and the values of lag are restricted to those occurring in B, lag|B.

For a CCC-Bicluster with gene-shifts (scaled patterns), P is defined as: $P(p_B \cup p_B^{\uparrow} \cup p_B^{\downarrow}) = P(p_B) + \sum_{shift} P(p_B^{\uparrow shift}) + \sum_{shift} P(p_B^{\downarrow shift})$, where $shift \in \{1, ..., K\}$ and K is the value used in the CCC-Biclustering algorithm with gene-shifts to shift the expression pattern K symbols up and down.



In the case of an *e*-CCC-Bicluster, the statistical significance or the value of *p*-value(*B*), is computed by obtaining the probability of a random occurrence under the null hypothesis of the expression patterns in the *e*-Neighborhood of the expression pattern p_B defining the *e*-CCC-Bicluster, $N(e, p_B)$, *k* times in *n* independent trials, where *k* is the number of genes in the bicluster minus 1 and *n* is the number of genes in the expression matrix minus 1.

This is performed using the simplifying assumption that the probability of occurrence of a specific expression pattern in the *e*-Neighborhood of the pattern p_B , $N(e, p_B)$, is adequately modeled by a first order Markov Chain, with state transition probabilities obtained from the values in the corresponding columns in the matrix. In the general case, $P(N(e, p_B)) = \sum_{i=1}^{|N(e, p_B)|} P(N(e, p_B)[i])$, where $N(e, p_B)$ and $N(e, p_B)[i]$ are, respectively, the number of patterns and the *i*th pattern in the *e*-Neighborhood of the pattern p_B .

When missing values are considered as valid errors, $N(e, p_B)$ is computed using the alphabet $\sum' \cup mv'$, where mv is the symbol used for missing value and each element mv' is obtained by concatenating m and one number in the range $\{1, ..., |C|\}$, that is, $mv' = \{mv\} \times \{1, ..., |C|\}$.

When only restricted errors are allowed, $N(e, p_B)$ is not computed using all the symbols in Σ' . The allowed substitutions for each symbol in p_B are the z neighbors, both to the left and to the right of $\Sigma'[p]$ that are considered as valid errors, where p is the position of the symbol $p_B[k]$ in Σ' .

For an *e*-CCC-Bicluster with sign-changes, *P* is computed as $P(N(e, p_B) \cup N(e, p_B^{-1})) = P(N(e, p_B)) + P(N(e, p_B^{-1}))$, where $P(N(e, p_B)) = \sum_{i=1}^{|N(e, p_B)|} P(N(e, p_B)[i])$, $P(N(e, p_B^{-1})) = \sum_{i=1}^{|N(e, p_B^{-1})|} P(N(e, p_B^{-1})[i])$. $|N(e, p_B)|$, $|N(e, p_B^{-1})|$, $N(e, p_B)[i]$ and $N(e, p_B^{-1})[i]$ are, respectively, the number of patterns and the *i*th pattern in the e-Neighborhood of the pattern p_B and p_B^{-1} , respectively.

For an *e*-CCC-Bicluster with time-lags, *P* is computed as $P(N(e, (p_B)_{\rightarrow}^{LAG})) = \sum_{lag|B} P(N(e, (p_B)_{\rightarrow}^{lag})).$

For an e-CCC-Bicluster with sign-changes and time-lags, P is computed as $P(N(e, (p_B)^{LAG}) \cup N(e, (p_B^{-1})^{LAG})) = \sum_{lag|B} P(N(e, (p_B)^{lag})) + P(N(e, (p_B^{-1})^{lag})).$



For an *e*-CCC-Blcluster with gene-shifts (scaled patterns), *P* is computed as $P\left(N(e, p_B) \cup N\left(e, p_B^{\uparrow}\right) \cup N\left(e, p_B^{\downarrow}\right)\right) = P\left(N(e, p_B)\right) + \sum_{shift} P\left(N\left(e, p_B^{\uparrow shift}\right)\right) + \sum_{shift} P\left(N\left(e, p_B^{\downarrow shift}\right)\right).$

References

[1] Sara C. Madeira, Efficient Biclustering Algorithms for Time Series Gene Expression Data Analysis, PhD Thesis, Instituto Superior Técnico, Technical University of Lisbon, Dec 2008.