User guide

Jasmin Straube

Version 1.0

Contents

1 TimeOmics in a Nutshell

TimeOmics is a user-friendly application to filter, visualize and analyse high dimensional time course ‘omics’ data. TimeOmics enables a variety of functions (Fig. 1), for molecule expression experiments measured on multiple biological replicates over multiple time points to enable:

- Filtering of not expressed or noisy molecules.
- Modelling of time course expression profiles.
- Clustering of modelled expression profiles.
- Analysing differential expression over time (between groups and time and group interaction, if two groups are available).

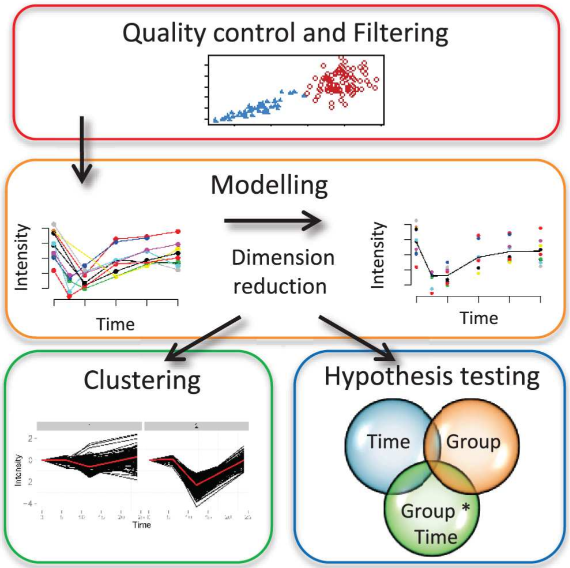


Figure 1: **Overview of the TimeOmics analysis framework.** The proposed framework consists of three stages: quality control and filtering; serial modelling of profiles; and analysis with clustering to identify similarities between profiles or with hypothesis testing to identify differences over time, between groups, and/or in group and time interactions.

1.1 Data requirements

TimeOmics does not provide tools for normalizing and transforming the data. You should make sure your data is appropriately normalized and transformed prior to analysis. Also make sure that you do not observe any batch effects that may effect the analysis.
Access to the full functionality of TimeOmics requires the data to have:

- A minimum of four time points.
- A minimum of three biological replicated measurements per time point.

1.2 Citing TimeOmics

The TimeOmics software can be cited as:

> Straube J, Bernard A, Huang BE and Lê Cao K-A (2015)
> TimeOmics: Web application for interactive time course 'omics' data analysis
> Submitted

The R package `lmms` can be cited as:

> Straube J, Lê Cao K-A, and Huang BE (2015)
> lmms: Linear Mixed Effect Model Splines for Modelling and Analysis of Time Course Data.
> R package version 1.3

The statistical method can be cited as:

> Straube J, Gorse A-D, Huang BE and Lê Cao K-A (2015)
> A linear mixed model spline framework for analyzing time course 'omics' data.
> PLoS ONE 10(8): e0134540. doi: 10.1371/journal.pone.0134540

1.3 How to get help?

The user guide and references will hopefully answer most questions about TimeOmics. However, additional questions can be directed to j.straube[at]qfab.org. We appreciate bug reports in the software or R functions and welcome any suggestions or comments for improvements.

2 Quick start

2.1 Downloading R, RStudio and TimeOmics

TimeOmics version 1.0 functionality was tested on R version 3.2 and RStudio version 0.99. Follow the subsequent steps to install R (≥ 3.2), RStudio (≥ 0.99) and TimeOmics:

1. Download and install the latest version of **R** for your machine from here.
2. Download and install the latest version of **RStudio Desktop** for your machine from here.
3. Once you have R and RStudio running, download TimeOmics from GitHub here.
4. You will also need to install the R package `shiny` in order to run TimeOmics. Open RStudio and type into the RStudio console:

```
> install.packages('shiny')
```

2.2 Run TimeOmics

Run TimeOmics by typing the following commands in the console:

```
> library(shiny)
> runApp('C:/filepath/to/TimeOmics')
```

Note: The first time you launch TimeOmics may take time as many package dependencies need to be automatically installed.

2.3 Workflow

TimeOmics can analyse data with either a single biological sample or multiple biological samples. Figure 2 displays the different analysis workflows.

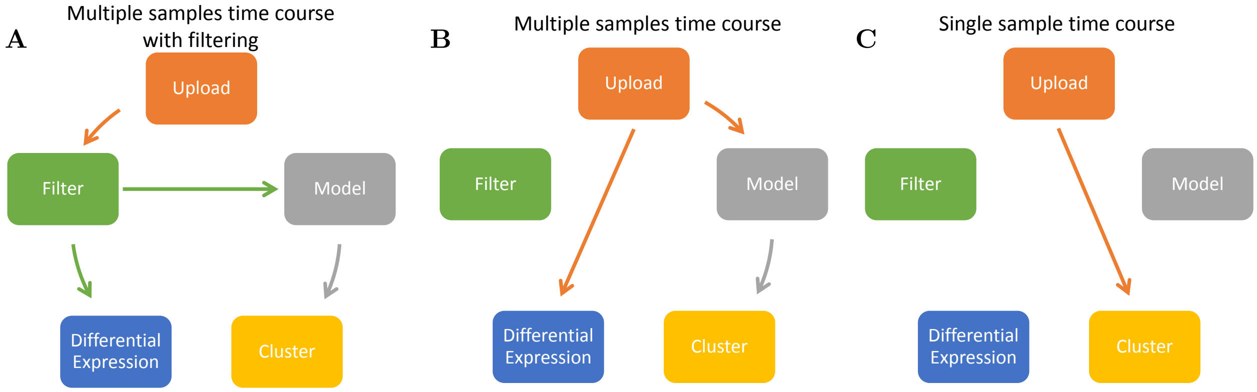


Figure 2: **Workflows.** Presented are the possible workflows using TimeOmics. Workflow **A** and **B** are for multiple samples time course experiments. In workflow **A** after uploading the data, you perform filtering prior to modelling the expression profiles and clustering or analysing for differential expression. In workflow **B** you can perform modelling, clustering and analysing for differential expression without filtering. Workflow **C** is for single sample time course experiments, where only the upload and the clustering functions are provided.

2.3.1 Multiple samples time course experiment

In time course studies multiple biological replicates are measured at multiple time points and the experimental design looks similar to table 1. With these kind of experiments all functions of TimeOmics are enabled. You can analyse your data following workflow A or B from Figure 2.

Sample ID	Time	Group
1	1	1
1	2	1
1	3	1
1	4	1
2	1	1
2	2	1
2	3	1
2	4	1
3	1	1
3	2	1
3	3	1
3	4	1

Table 1: **Workflow A or B:** Multiple samples experimental design.

2.3.2 Single Sample time course experiment

In the case where you only measure one source or use a different approach to summarize and model the time trajectories using the median or the mean, the experimental design looks like table 2. You can only use the functionality provided in the 'Upload' tab and 'Cluster' tab.

Sample ID	Time	Group
1	1	1
1	2	1
1	3	1
1	4	1

Table 2: **Workflow C:** Single sample experimental design.

2.4 Download Images and Tables

All images generated in TimeOmics can be easily saved with a right click on the image -> save image. Tables can be downloaded through the 'Download' button in a .csv format.

2.5 Run an Example

You can run TimeOmics with data provided in two ways, either by uploading the example data available from the ExampleData folder in the downloaded TimeOmics application or on the user interface itself by going to the 'Example and Help' tab and check the 'Run example' check box. You can now follow workflow A or B from Figure 2 to get familiar with TimeOmics functionality.

3 Upload

The ‘Upload’ tab provides an interface to upload data and assess the quality via data distribution plots. Moreover you will select what groups in your data you further want to analyse. Example data is available in the downloaded TimeOmics application folder ‘ExampleData’.

3.1 Input, Format and Workflow

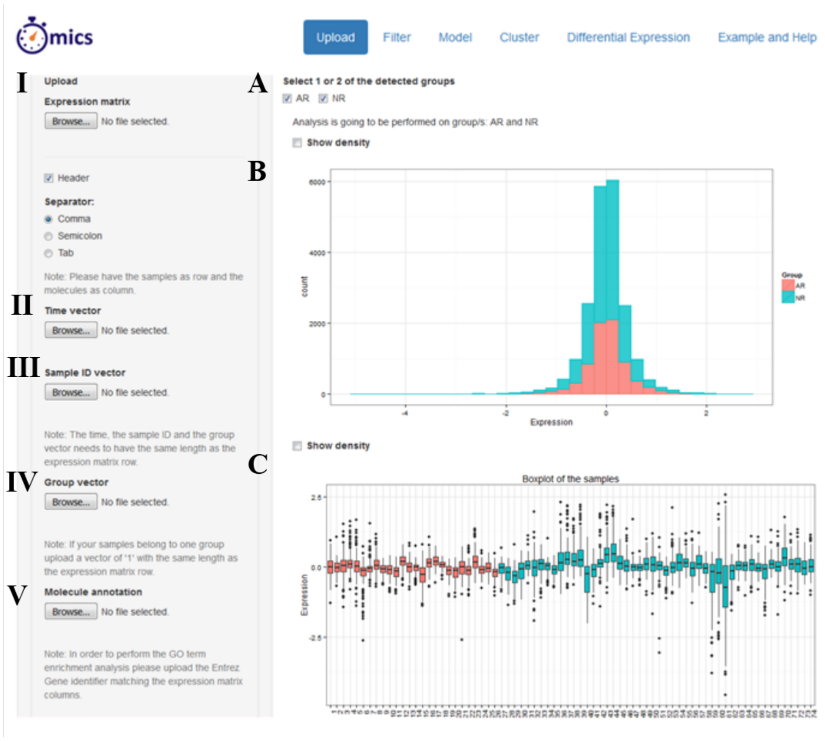
The Upload step requires four files:

- An **expression data set** in a matrix with samples in rows and molecules in columns (Fig. 3 I). The matrix can contain a header and the delimiter in your file can be either comma, semicolon or tab.
- A numeric vector indicating the **time** point of each sample (Fig. 3 II). The vector is comma separated and does not have a header. The time vector needs to match the samples from the expression matrix.
- A character or numeric vector indicating the **source of the sample** (e.g. sample ID) (Fig. 3 III). The vector is comma separated and does not have a header. The sampleID vector needs to match the expression matrix rows.
- A numeric or character vector indicating the **sample groups** (Fig. 3 IV). The vector is comma separated and does not include a header. The sample group vector needs to match the samples from the expression matrix.
- Optional, a character vector of **molecule annotation** (Fig. 3 V). The molecule annotation file can either contain gene symbols or protein identifier (IPI). The vector is comma separated and does not include a header. The molecule annotation vector needs to match the molecules from the expression matrix.

1. Upload the file molecule expression matrix by clicking on the ’Choose File’ button and select your file.
2. Indicate whether your data file has a header by selecting the ‘Header’ check box.
3. Indicate the delimiter in your file either: comma, semicolon or tab.
4. Upload the files containing information about the sampled time point (Time vector), the source of the sample (Sample ID vector) and the according group (Sample Group vector).
5. Select the groups you further want to investigate.
6. Choose to visualize the density distribution of the histogram or boxplot by ticking the ‘Show density’ check box.

3.2 Graphical outputs

The plots generated when uploading the files aid the user to check the quality of the uploaded data. The first plot generated is a histogram plot of the whole uploaded expression matrix (combining all samples and molecules, Fig. 3B). Optionally, the data can be visualized as density distribution by ticking the check box above the histogram plot ‘Show density’. Data should appear bell shaped (normally distributed) since samples are assumed to be normalized. If you observe strong deviations from a normal distribution, we recommend to normalize the data prior to using TimeOmics. If the data is not normally distributed TimeOmics may not produce valid results.



Upload

Expression matrix

Browse... No file selected.

☑ Header

Separator:

● Comma

○ Semicolon

○ Tab

Note: Please have the samples as row and the molecules as column.

Time vector

Browse... No file selected.

Sample ID vector

Browse... No file selected.

Note: The time, the sample ID and the group vector needs to have the same length as the expression matrix row.

Group vector

Browse... No file selected.

Note: If your samples belong to one group upload a vector of '1' with the same length as the expression matrix row.

Molecule annotation

Browse... No file selected.

Note: In order to perform the GO term enrichment analysis please upload the Entrez Gene identifier matching the expression matrix columns.

Select 1 or 2 of the detected groups

☑ AR ☑ NR

Analysis is going to be performed on group/s: AR and NR

□ Show density

□ Show density

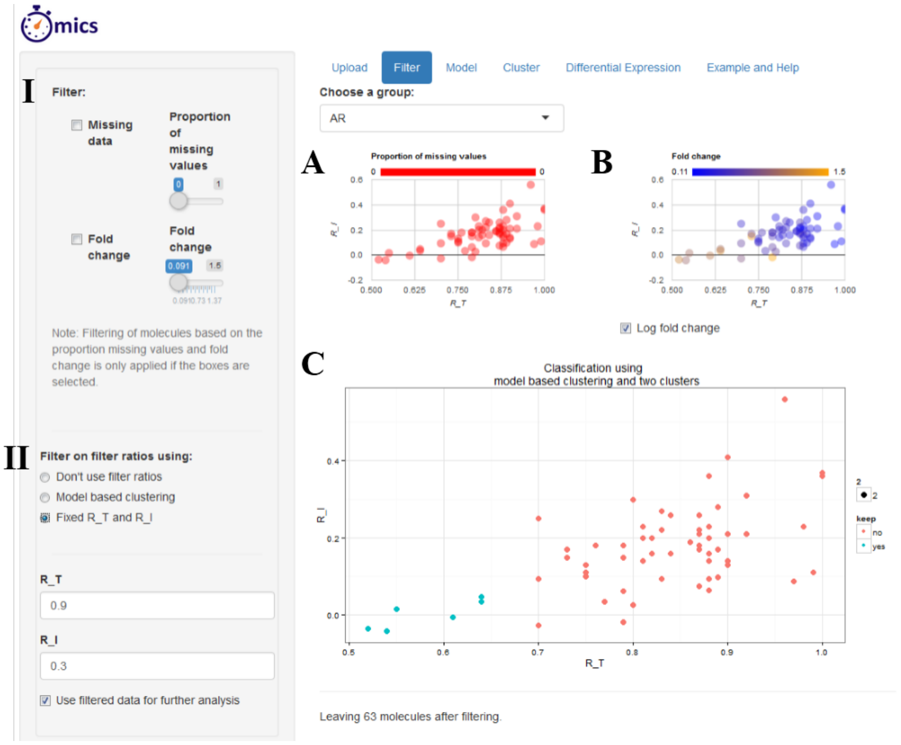
Figure 3: **Upload.** The 'Upload' tab is the user interface to upload the data to be analysed. TimeOmics requires an expression matrix **I**, a vector indicating the sample time point **II**, a vector indicating the sample source **II**, a vector indicating the grouping **IV** and optionally the molecule annotation **V**. Select the groups you want to analyse **A**. The data histogram **B** and sample boxplot **C** are visualized upon uploading the data.

On uploading group information the histogram plot will change appearance if more than one group is available. The histogram and density will display multiple histograms coloured by the according groups in the sample.

The second plot displays the sample wise boxplot (Fig. 3C). Optionally, the density for each sample can be displayed by ticking the 'Show density' check box above the sample boxplot. The data distribution should be roughly the same for all samples (homogeneously distributed). If sample boxes show strong deviations it may indicate technical difficulties or batch effects and analysis should proceed with caution.

4 Filter

‘Omics’ experiments measure the intensity of thousands of molecules at the same time. However, only a few molecules will respond to treatment or perturbation. Molecules that do not change expression over time or are noisy (non-informative) can be removed prior to analysis.
The method provided is specifically developed for time course data. It is based on the time to molecule standard deviation ratio (R_T) and the subject to molecule standard deviation ratio (R_I) as described in [1] and implemented in the `lmms` R package [2]. High filter ratios indicate non-informative molecules. However, filter ratios can be affected by high number of missing values. Therefore, the filter ratios are visualized along with the proportion of missing values and enable to filter molecules with a lot of missing data. We suggest to first remove molecules with a high number of missing values prior to filtering on the filter ratios.



I Filter:

□ Missing data — Proportion of missing values: 0 – 1

□ Fold change — Fold change: 0.091 – 1.5

Note: Filtering of molecules based on the proportion missing values and fold change is only applied if the boxes are selected.

II Filter on filter ratios using:

○ Don't use filter ratios
○ Model based clustering
● Fixed R_T and R_I

R_T: 0.9

R_I: 0.3

☑ Use filtered data for further analysis

Choose a group: AR

☑ Log fold change

Leaving 63 molecules after filtering.

Figure 4: **Filter.** The ‘Filter’ tab allows you to filter for proportion of missing values and fold changes **I**. Moreover, you can filter molecules based on their filter rations **II**. The filter ratio plots help you to identify appropriate threshold for the proportion of missing values **A** the fold changes **B**. Filter ratio plot **C** shows the classification using a model based clustering algorithm and helps to discriminate between informative and non-informative profiles.

4.1 Graphical outputs

After uploading your data click on the ‘Filter’ tab. TimeOmics automatically visualizes the filter ratios R_T and R_I for each molecule. The filter ratios should guide the user to identify molecules that are not expressed or do not have a consistent response across subjects over time. The ratios are visualized with three different types of information:

- Proportion of missing values (Fig. 4 A)

- Fold change (Fig. 4 B)
- Model based clustering results with two clusters (Fig. 4 C)

If multiple groups are available the data of only one group will be visualized. You can switch between groups by selecting 'Choose a group' drop down menu.

4.2 Filter Workflow

4.2.1 Missing Values/Fold Change

You can filter the data by proportion of missing values or fold change (Fig. 4 I).

1. Select the fold change or proportion of missing values cut-off using the slider.
2. Select the check box of the variables you wish to filter for.
3. Check the 'Use filtered data for further analysis' check box to remove all molecules that do not meet the criteria.

Note: the data is by default assumed to be log transformed for the fold change calculation. Hence, the fold change is the difference between the maximum average time point value and the minimum average time point value. If your data is not log transformed, deselect the check box below Figure 4 B. The fold change will then be updated to the ratio of the maximum average time point value and the minimum average time point value.

4.2.2 Filter Ratios

You can remove molecules with no expression changes over time or noisy (non-informative) profiles by filtering molecules with high filter ratios (R_T, R_I) using two approaches.

- Use the classification of a model based clustering algorithm [8] to discriminate informative from non-informative profiles.
    1. Select the radio button 'Model based clustering'.
    2. Molecules are removed that are in the cluster with the high R_T and R_I coloured red in the plot of Figure 4 C.
    3. Check the 'Use filtered data for further analysis' check box to keep working on the filtered data set.
- Alternatively, use fixed R_T and R_I to filter molecules.
    1. Define R_T and R_I (Default values are $R_T = 0.9$ and $R_I = 0.3$).
    2. Select the radio button 'Fixed R_T and R_I'.
    3. Molecules are removed that have higher R_T and R_I values as defined.
    4. Check the 'Use filtered data for further analysis' check box to keep working on the filtered data set.

4.2.3 Filtering data with two groups

You can not select different criteria (proportion of missing values, fold change or classification of model based clustering) for different groups. Moreover, molecules are only filtered if they do not pass the defined criteria for both groups.

5 Model

Modelling a representative expression profile for molecules from multiple subjects measured over time is not a trivial task. Subjects can have the same expression for some molecules, while for other molecules they have different levels of expression or even change expression at different rates. Using the mean or median to obtain a representative expression profile for all subjects is simple but often sensitive to outliers and missing data. A better solution is to model expression profiles as smooth function of time. To model this representative expression profile for each molecule we used our data driven method Linear Mixed effect Model Splines (LMMS) [1]. LMMS has many advantages. It can handle unbalanced designs and missing data. It also takes into account the variance structure of the subjects, which was shown to prevent over or under smoothing of the profiles.
TimeOmics offers a range of splines to be selected by the user namely 'cubic', 'p-spline' and 'cubic p-spline'. The difference between these splines is the basis function and the way they select the knots or break points. The 'cubic' spline basis as proposed by [3], uses all inner time points of the measured time interval as knots. The 'p-spline' and the a 'cubic p-spline' basis use the quantiles of the measured time interval as knots as proposed by [4]. Cubic splines are preferred if the number of time points are small. However, with increasing number of time points this basis can get very computational expensive. The alternative is to use either the 'p-spline' or the 'cubic p-spline'. The models applied to the data are a linear model (0), a LMMS (1), a LMMS with subject-specific random intercept (2) and a LMMS with subject-specific random intercept and slope (3).

5.1 Model Workflow

After you uploaded and optionally filtered the data you can model the representative molecule expression profiles (Fig. 5 I). If you selected more than one group in the 'Upload' tab the molecules will be modelled for each group individually.

1. Select the spline basis.
2. Click on the 'Model' button. When the modelling is completed, a table of each molecule with the model used (linear model (0), a LMMS (1), a LMMS with random intercept (2) and LMMS with random intercept and slope (3)) appears.

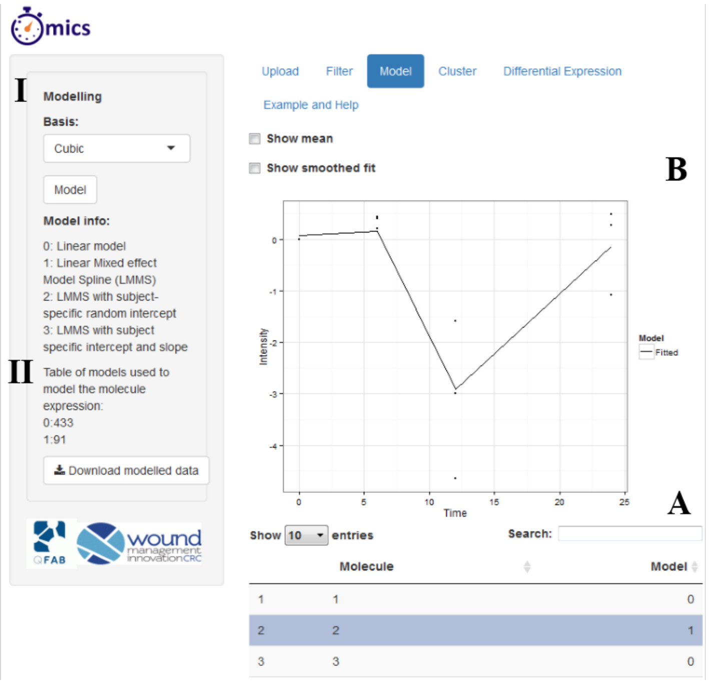
Then several options are available:

- Sort the table by molecule name or by model.
- Search for a molecule name or model by typing into to the 'Search' field at the top right of the table.
- Select a molecule in the table to visualize the modelled profile along with the expression data.
- Download the LMMS modelled profiles (provided as a comma separated file) or proceed with the clustering of the modelled molecular profiles.

By default TimeOmics outputs the fitted time points observed in the experiment. The `lmms` R package provides further functionality, that allows the prediction of more time points within the measured time interval.

5.2 Graphical outputs

Once the modelling step is completed a table appears (Fig. 5A) and you can visualize the expression of each molecule with respect to time (Fig. 5B) by clicking on the molecule in the provided table. The fitted profile is the default visualization. You can smooth the fitted profile or add the mean profile to the plot by selecting the corresponding check boxes.



Modelling

Basis: Cubic

Model

Model info:

0: Linear model
1: Linear Mixed effect Model Spline (LMMS)
2: LMMS with subject-specific random intercept
3: LMMS with subject specific intercept and slope

Table of models used to model the molecule expression:
0:433
1:91

□ Show mean

□ Show smoothed fit

Show 10 entries Search:

	Molecule	Model
1	1	0
2	2	1
3	3	0

Figure 5: **Model.** The ‘Model’ tab (Section 5) enables the modelling of multiple samples time course experiments. You can select the spline basis to be used to model the data **I**. After modelling is completed you obtain a summary with the number of models used to model your data **II**. Moreover, you obtain a table with each molecule and the model used **A**. You can also visualize the modelled profiles together with the measured data and the mean **B**.

6 Cluster

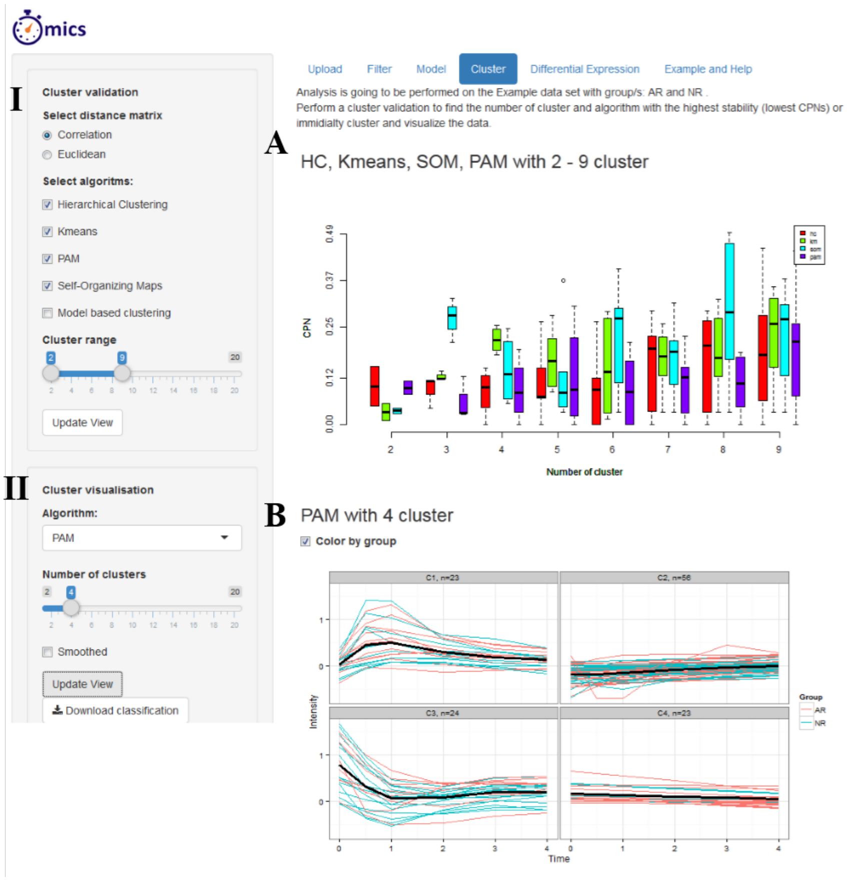
Clustering of the molecular profiles can give valuable insights into a systems molecular behaviour. Molecule profiles with the same pattern are believed to be co-regulated, have similar functions or are involved in the same biological processes. The analysis of the individual clusters for biological enrichment can therefore be more informative than analysing all differentially expressed molecules over time. We provide the choice of cluster analysis on the LMMS modelled profiles euclidean distance or on the Pearson correlation matrix.
Clustering on the Pearson correlation may perform better if you wish to cluster profiles according to their shape. Clustering on the euclidean distance will rather lead to clustering on the distance and same shaped profiles with different expression levels may not be clustered together.
The data used for the clustering input is either modelled profiles using the ‘Model’ tab provided functions, alternatively it can be a single sample experimental design.

The ‘Cluster’ analysis consist of three parts:

1. Cluster stability assessment (Fig. 6 I).
2. Expression profile cluster visualisation (Fig. 6 II).
3. GO enrichment analysis of clusters (Fig. 6 III).

The following clustering algorithms are provided:

- Hierarchical Clustering (HC) with Ward agglomeration method [5]
- Kmeans [5]
- Partitioning Around Medoids (PAM) [6]
- Self-Organizing Maps (SOM) [7]
- Model based clustering (Mclust) [8]



Analysis is going to be performed on the Example data set with group/s: AR and NR .
Perform a cluster validation to find the number of cluster and algorithm with the highest stability (lowest CPNs) or immidialty cluster and visualize the data.

Cluster validation

Select distance matrix

● Correlation
○ Euclidean

Select algoritms:

☑ Hierarchical Clustering
☑ Kmeans
☑ PAM
☑ Self-Organizing Maps
□ Model based clustering

Cluster range: 2 – 9

Cluster visualisation

Algorithm: PAM

Number of clusters: 4

□ Smoothed

☑ Color by group

Figure 6: **Cluster.** The 'Cluster' tab enables the stability assessment of clusters given different algorithms and number of clusters **I**. A boxplot helps you to choose the algorithm and number of clusters that predicts the most stable clustering **A**. Then you can choose the cluster algorithm and the number of clusters **II** to visualize the clustered profiles **B**.

6.1 Cluster Stability Assessment Workflow

In order to perform the cluster stability assessment do the following steps:

1. Select if you wish to perform the assessment on the correlation matrix.
2. Select the cluster algorithms you wish to assess.
3. Choose the range of numbers of clusters you wish to assess.

4. Finally, press the 'Update View' button to perform the analysis.
5. You can download a table with the molecules and the cluster they fell into by clicking on the 'Download classification' button.

6.1.1 Cluster Proportion of Non-overlap

The Cluster Proportion of Non-overlap (CPN) is provided to assess cluster stability via leave-one-out cross-validation. We modified the implementation of the CPN as proposed by [9] to obtain a CPN per cluster. The CPN measures the proportion of observations not placed in the same cluster with a clustering based on the full data and a clustering based on the data with a single time point removed. Let C^i represent the cluster containing observation i using all of the available data and $C^{i,-t}$ represent the cluster containing observation i, where the clustering is based on the dataset with time point t removed. Then T is the number of different time points, and N is the number of different observations in the data, the CPN is given by:

$$CPN = \frac{1}{TN} \sum_{i=1}^{N} \sum_{t=1}^{T} 1 - \frac{C^{i,-t} \cap C^i}{C^i}$$

The CPN is in the interval [0; 1], with values close to zero corresponding to highly robust clustering results.

6.1.2 Graphical outputs

The stability measurement is calculated as the Cluster Proportion of Non-overlap (CPN). The CPN for each algorithm and number of clusters is visualized as a boxplot (Fig. 6 A). The algorithm and number of clusters with the smallest and lowest box should be chosen.

6.2 Expression Profile Clustering Workflow

In order to cluster the expression profiles:

1. Select the clustering algorithm.
2. Select the number of clusters.
3. Click the 'Update View' button to perform the analysis. The clustered profiles will be visualized as soon as the analysis was successfully performed.
4. You can choose to color the profiles by group.

6.2.1 Graphical outputs

The cluster plot visualizes the modelled profiles grouped by the predicted cluster (Fig. 6 B). The black line is the mean profile of the predicted cluster. The header gives the information about the cluster number and the number of profiles that were predicted in that particular cluster.

6.3 GO Enrichment Analysis

To validate the biological relevance of the clustered profiles, you can perform a size based enrichment analysis. To use this function you need to

1. Provide gene symbols of protein identifier (IPI) in the 'Upload' tab.

2. Cluster your profiles as described in the 'Expression Profile Clustering Workflow' section. Once the analysis was successful, you receive a table with the following information:

    - Cluster of the enriched GO term (**Cluster**)
    - Number of molecules in your data in the GO category (**Counts**)
    - Overall number of molecules in the GO category (**Num.Mols**)
    - Size of the cluster (**Cluster.size**)
    - GO ID (**GO**)
    - GO description (**GODescrip**)
    - Ontology (**Ontology**)
    - P-value of the hypergeometric distribution (**p.value**)
    - FDR adjusted p-value (**adj.p**)

3. Sort the table by any of the above mentioned table entries either alphabetically or numerically decreasing/increasing by clicking on the according table name.

4. Search for anything in the table by typing into to the 'Search' field at the top right of the table.

Note: The GO enrichment analysis is currently only available for human using the *org.Hs.eg.db* R package [10].

How is the GO enrichment analysis performed? The enrichment is computed using the hypergeometric distribution based on the number of molecules in the set of interest. Let p be the total number of unique molecules observed in the experiment, p_f the total number of molecules that are in the GO category of interest, and g_c the number of molecules assigned to a given cluster C. Based on the hypergeometric distribution the p-value of seeing p_{GO} molecules in the intersection of the category of interest and cluster C can be computed as

$$\sum^{min(g_c,p_f)} \frac{\binom{p_f}{i}\binom{p-p_f}{g_c-i}}{\binom{p}{g_c}}$$

7 Differential Expression

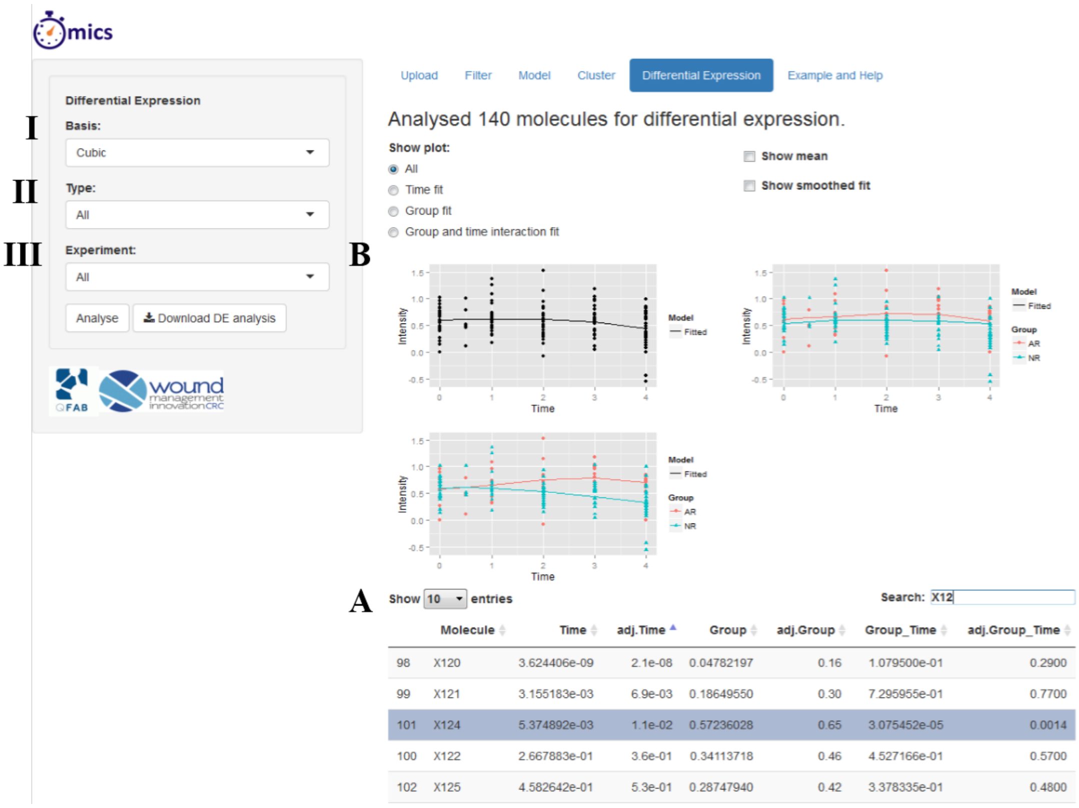
Differential expression analysis is a mean to identify differentially expressed molecules between groups and/or over time. Here we use the Linear Mixed effect Model Spline (LMMS) approach as described in [1] and implemented in the `lmms` R package.
You can select the spline basis as described in Section 5. Given two groups you can define the 'Type' of differential expression analysis to be performed either over time, between groups or time and group interaction. You can also perform all three of them at the same time. If you only have one group you can only test for differential expression over time.
Moreover, you can define the type of 'Experiment' to restrict the modelling to one model for each measured molecule. We discriminate between 'Time course', 'Longitudinal 1' and 'Longitudinal 2' models. 'Time course' models each molecule with a LMMS. You select this model if you don't expect much variation across biological replicates (e.g. cell cultures). 'Longitudinal 1' models a LMMS subject-specific intercept. Select this model if you expect subject-specific expression levels. 'Longitudinal 2' models each molecule with a LMMS and a random intercept and slope. Select this model if you expect your the subject-specific molecule expression to vary in expression levels and speed of expression change. However, each molecule expression can be differently affected by environmental or genetic factors. Choosing one model for all molecules may lead to over- or under-smoothing of the data. A solution is to let the data drive the modelling process. TimeOmics provides this functionality if you select 'All' in the 'Experiment' drop down menu. All three models are considered and for each molecule the model that has the best goodness of fit is selected. The analysis returns p-values as well as False Discovery Rate (FDR) adjusted p-values as proposed by [11].

7.1 Differential Expression Workflow

1. Select the basis for modelling (Fig. 7 I).
2. Define the 'Type' of the differential expression analysis by selecting in the drop down menu (Fig. 7 II).
3. Define the 'Experiment' performed (Fig. 7 III).
4. Click the 'Analyse' Button to perform the analysis.
5. Once the analysis is done a table appears for each molecule with the p-value for the test performed and the FDR adjusted p-value (Fig. 7 A). The table can be sorted by each of the above mentioned table entries either alphabetically or numerically decreasing/increasing by clicking on the table name of choice. You can also search for anything in the table by typing into the 'Search' field at the top right of the table.
6. Click on the molecule in the table to visualize the expression data along with the modelled profiles (Fig. 7 B).
7. Download the results by clicking on the 'Download DE analysis' button.

7.2 Graphical outputs

After the analysis is completed you can click on it in the provided table (Fig. 7 A) and visualize the expression of each molecule depending on time (Fig. 7 B). Given you performed all differential expression types you can visualize all models at once by clicking the 'Show plot' radio button 'All'. Alternatively you can visualize only the 'Time', 'Group' or 'Group and time interaction' fit. Moreover, you can visualize the expression data and the LMMS fitted data along with the mean profile or a smoothed fit by selecting the according check boxes.



Differential Expression

Basis: Cubic

Type: All

Experiment: All

Analyse | Download DE analysis

Analysed 140 molecules for differential expression.

Show plot:
- ● All
- ○ Time fit
- ○ Group fit
- ○ Group and time interaction fit

□ Show mean

□ Show smoothed fit

Show 10 entries   Search: X12

	Molecule	Time	adj.Time	Group	adj.Group	Group_Time	adj.Group_Time
98	X120	3.624406e-09	2.1e-08	0.04782197	0.16	1.079500e-01	0.2900
99	X121	3.155183e-03	6.9e-03	0.18649550	0.30	7.295955e-01	0.7700
101	X124	5.374892e-03	1.1e-02	0.57236028	0.65	3.075452e-05	0.0014
100	X122	2.667883e-01	3.6e-01	0.34113718	0.46	4.527166e-01	0.5700
102	X125	4.582642e-01	5.3e-01	0.28747940	0.42	3.378335e-01	0.4800

Figure 7: **Differential expression.** The 'Differential Expression' tab enables the differential expression analysis. You can select the basis of the spline **I**, define the type of differential expression analysis **II** and the kind of experiment performed **III**. Once the analysis is done a table appears for each molecule with p-value and FDR adjusted p-value **A**. By clicking on the molecule in the table you can visualize the expression data with the modelled profiles and the mean expression profile **B**.

References

[1] Straube J, Gorse A-D, Huang BE and Lê Cao K-A. A linear mixed model spline framework for analyzing time course 'omics' data. PLoS ONE 2015;10(8):e0134540.

[2] Straube J, Lê Cao K-A, and Huang BE. *lmms: Linear Mixed Effect Model Splines for Modelling and Analysis of Time Course Data*, 2015. R package version 1.3.

[3] Verbyla AP, Cullis BR, Kenward MG, Welham SJ. The analysis of designed experiments and longitudinal data by using smoothing splines. Appl Statist. 1999;48(3):269–311.

[4] Ruppert D. Selecting the Number of Knots for Penalized Splines. J Comput Graph Statist. 2002;11(4):735–757.

[5] R Core Team. R: A language and environment for statistical computing, 2015. R Foundation for Statistical Computing.

[6] Maechler M, Rousseeuw P, Struyf A, Hubert M, Hornik K. *cluster: Cluster Analysis Basics and Extensions*, 2013. R package version 1.14.4.

[7] Wehrens R, Buydens L. *Self- and Super-organising Maps in R: the kohonen package*, 2007. R package version 2.0.14.

[8] Auder B, Lebret R, Iovleff S, Langrognet F. *Rmixmod: An interface for MIXMOD*, 2014. R package version 2.0.2.

[9] Brock G, Pihur V, Datta S, Datta S. clValid: *Validation of Clustering Results*, 2011. R package version 0.6-4.

[10] Carlson M. *org.Hs.eg.db: Genome wide annotation for Human*, 2015. R package version 3.1.2

[11] Benjamini Y, Hochberg Y. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. J R Stat Soc Ser B Stat Soc. 1995;57(1):289–300.