ProStaR Tutorial

Florence Combes*, Wieczorek Samuel*, Burger Thomas*

April 6, 2017

Abstract

This document is a tutorial to *ProStaR*. It uses the data stored in the 'Exp1_R25_pept.txt' file, from the *DAPARdata* package, to illustrate on a walkthrough example the use of *ProStaR* to conduct a proteomics quantitative analysis.

*firstname.lastname@cea.fr

Contents

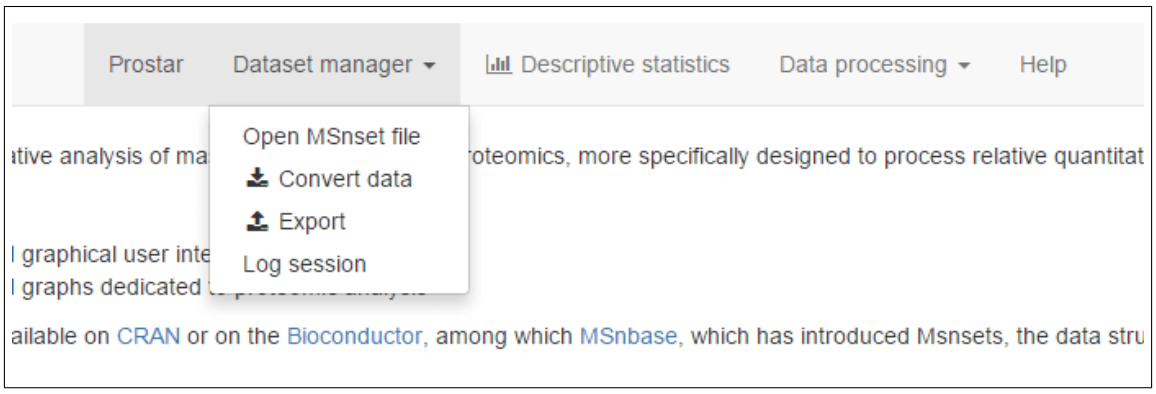

Figure 1: Dataset Manager Menu

1 Preambule

This document is only a step by step illustration of the use of *ProStaR*. For a comprehensive description of *ProStaR* functionalities, we refer the readers to the "*DAPAR* and *ProStaR* user manual", available on the Bioconductor webpage of *ProStaR*.

2 Loading data

The text-format file of 'Exp1_R25_pept.txt' is part of the *DAPARdata* package. For a precise understanding of the dataset, please refer to the document "Description of the UPS{pep/prot}25 datasets" available on the bioconductor webpage of *DAPARdata*.

In a nutshell, it gathers the peptide intensities of 6 samples, containing each the same yeast lysate. In addition, several human proteins have been spiked in, with different concentrations: three samples theoretically contains 2.5 more humain proteins than the three others. At the end of the quantitative analysis, all and only these proteins should be selected as significantly differentially abundant.

The text file containing the data must be of tabular form, as those produced by MaxQuant software, for instance. It is to be uploaded by clicking on the "Convert data" of the "Dataset manager" submenu (see Fig. 1). This opens the page captured on Fig. 2 and composed of five panels: Select File, Data Id, Exp and Feat data, Samples and Convert.

In the panel **Select File**: Indicate where to find the text file containing the data. As in this example, the file contains peptide intensities that are not depicted on a logarithm scale, the corresponding options must be manually selected (see Fig. 2).

In the panel **Data Id**: Indicate the column that contains the IDs of the peptides. These IDs have to be unique across all the rows. In the example dataset, it is simply the column referred to as "id". This is why "id" is selected in the drop down menu (see Fig. 3).

In the panel **Exp. and Feat. Data**: indicate the title of the columns that contains quantitative data of the peptides (See Fig. 4).

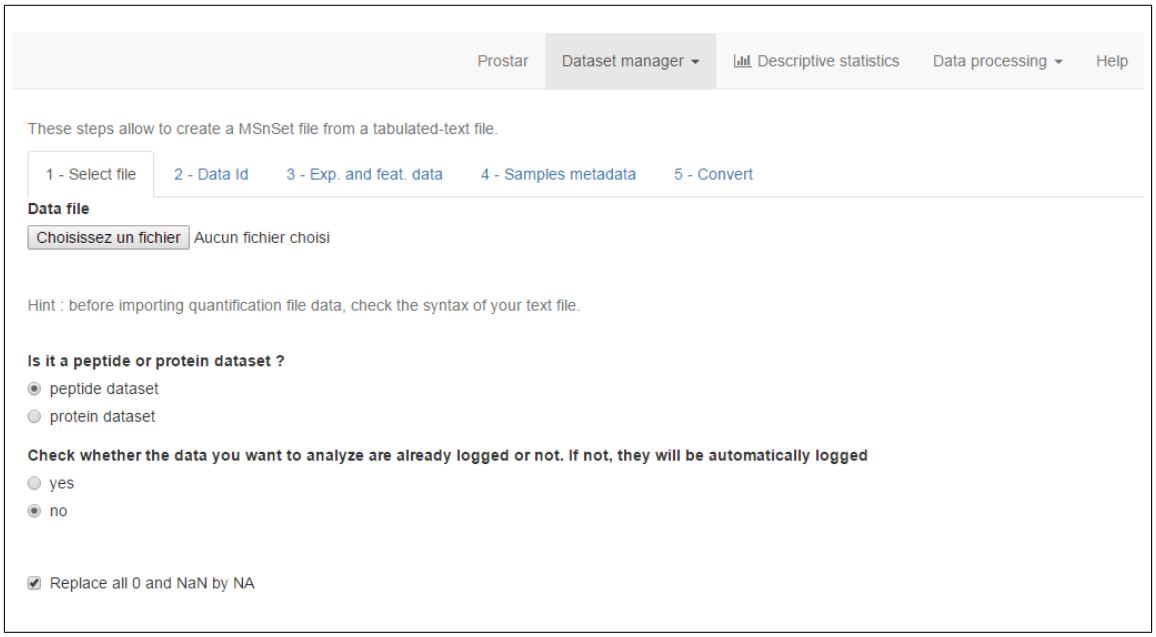

Figure 2: Conversion Text Data Page

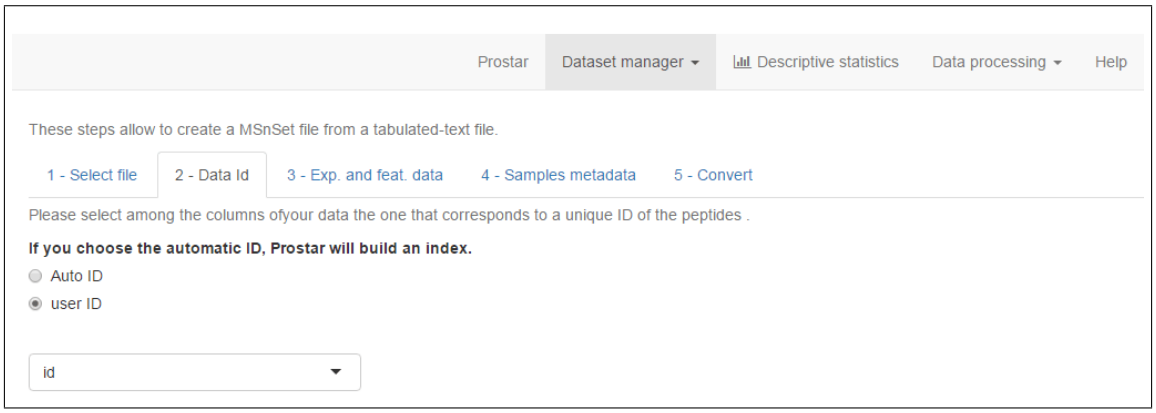

Figure 3: Using user-column 'id' as an ID in ProStaR

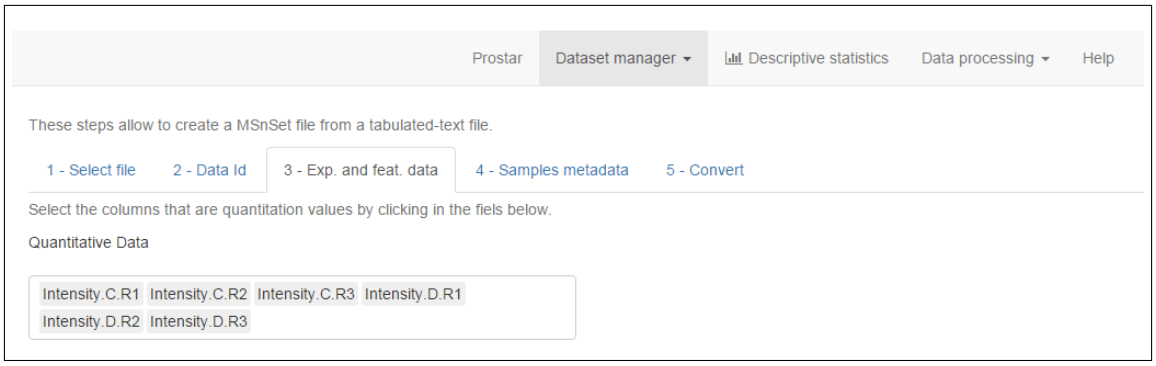

Figure 4: Exp. and Feat. Data panel

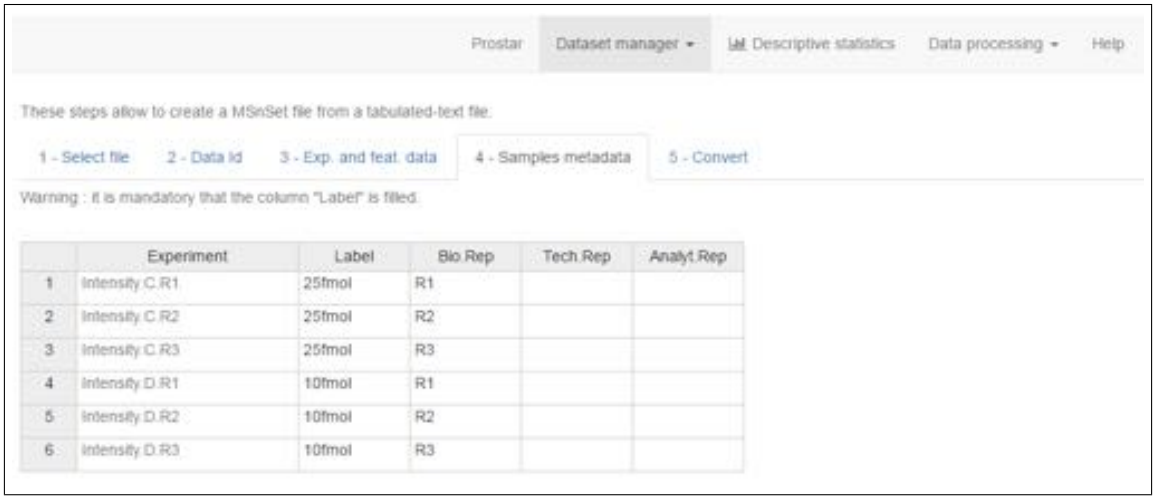

Figure 5: Metadata table to fill in (Samples metadata panel)

Figure 6: The conversion of text file to MSnSet format is done.

In the panel **Samples metadata**: Fill in the table according to the properties of the experimental design (see Fig. 5). In the example case, the first 3 samples correspond to 25 fmol of human proteins, while the others to 10fmol; the concentration is used to refer to the conditions. What matters here is to have different name for the two conditions, and to have a similar name within each condition. Because of the simple experimental design, the biological, technical and analytical replicates do not need to be filled.

Finally, in the panel **Convert**, enter the name of the project and click on "Convert data" to build an MSnSet object containing the data. A message indicates when the conversion is over (see fig. 6).

Please also notice in the screenshot that the value in the top-right box has changed from 'None' to 'Original - peptide'. All along the *ProStaR* pipeline, this box indicates the current state of the dataset. Here "original" means that no processing has been performed so far, and "peptide" means that the dataset contains peptide intensities.

3 Descriptive statistics

Once the data are loaded, their exploration/visualisation is the natural following step. By clicking on "Descriptive statistics" in the menu, one finds a series of 9 panels, that are helpful to understand the data:

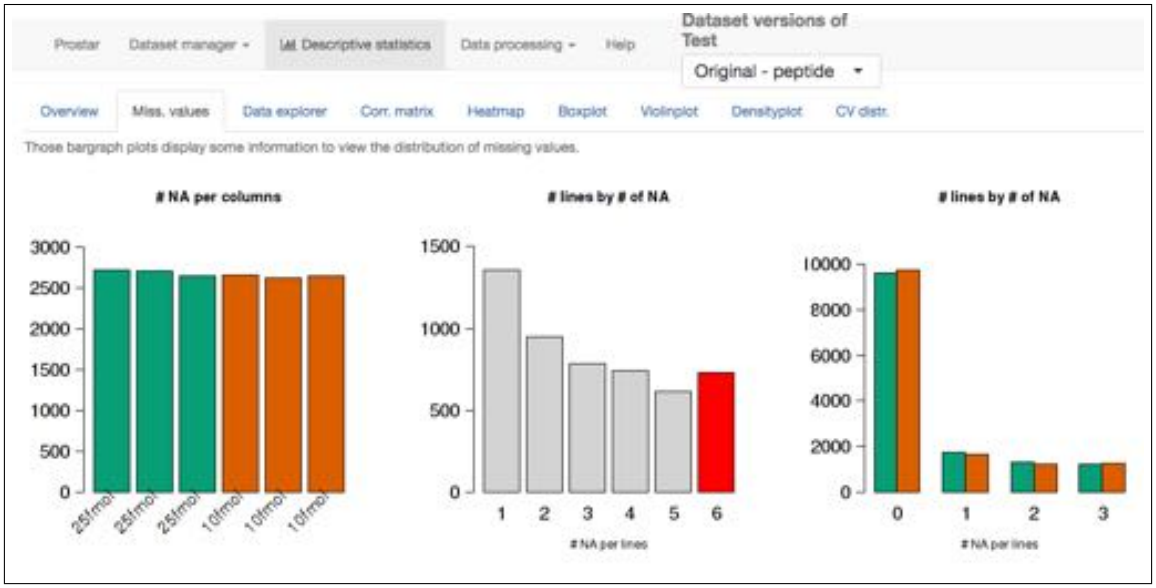

Figure 7: Missing Values repartition in the dataset

The first one, referred to as **Overview**, shows very basics and general informations about your dataset. For our example we can see we have: 6 samples, 13919 peptides, 19.19% of missing values and 733 lines with only NA values (missing values).

The second one, referred to as **Missing values**, summarizes the missing values distributions with 3 plots (see fig. 7 from left to right), that are also use to monitor the imputation of missing values (see Section 6):

- **'# NA per columns'** i.e. the number of missing values per sample.
- **'# of lines by # of NA'**: how many lines (peptides in our example) have 1, 2, .., up to 6 NA. The red bar bar of the barplot (6 here) corresponds to peptides that have only missing values, i.e. that have been identified but not quantified. Notice that this corresponds to the 733 lines already pointed on the previous paragraph.
- **'# of lines by # of NA per condition'**: It is the same plot as the second one, yet condition-wise, rather than entire dataset-wise. Notice that in this third plot, the number of lines with 0 missing values is presented, contrarily to the second one (where it would shrink too much the Y-axis).

The third one, referred to as **Data explorer**, allows the practitioner to navigate through the MSnSet containing the data. Several tables are available: quantitative data, peptides metadata, replicate metadata.

The other panels produce graphical representations of the dataset: **Boxplot**, **Densityplot**, **Correlation matrix**, **Heatmap**, **Violinplot** and **CV distribution** (see some of them on Fig. 8 to 10).

In particular:

- The correlation matrix (Fig. 8) shows well the replicates correlate better within conditions (25fmol together or 10fmol together). This confirms the qualilty of the dataset.

Figure 8: Correlation Matrix (for the 6 samples)

- On the density plots (Fig. 10) the distributions are Gaussian-like. It is a pre-requisite for most next coming statistical processing.

4 Filtering

Clicking on 'Filter data' in the drop down submenu 'Data processing' allows the practitioner to filter out some lines of the dataset according to various criteria.

In the first panel, one filters out peptides with too many missing values. The same plots as those from the 'Descriptive statistics' panel are displayed (see Fig. 11). For this dataset, we propose to filter out peptides that have at maximum 1 missing value among the 3 peptide intensities of each condition (but other filtering option would also make sense, depending on the dataset, and on the next coming processing). This results in the options selection that is illustrated on Fig. 12, where one sees that the 3 plots have been automatically updated, as soon as the 'Perform filtering MV' button has been clicked on.

Once this filtering has been performed, it is possible to move to the string-based filtering panel, which proposes to remove lines on the basis of some meta-data (i.e. peptide sequences that are known to be contaminants ou reverse sequences). This panels works as the previous one and is shown on Fig. 13.

Finally, one goes on the third panel, named 'Visualise and validate', that aims at performing the final validation of the filtering steps. To do so, one simply clicks on "Save filtered dataset" (Fig. 14). From

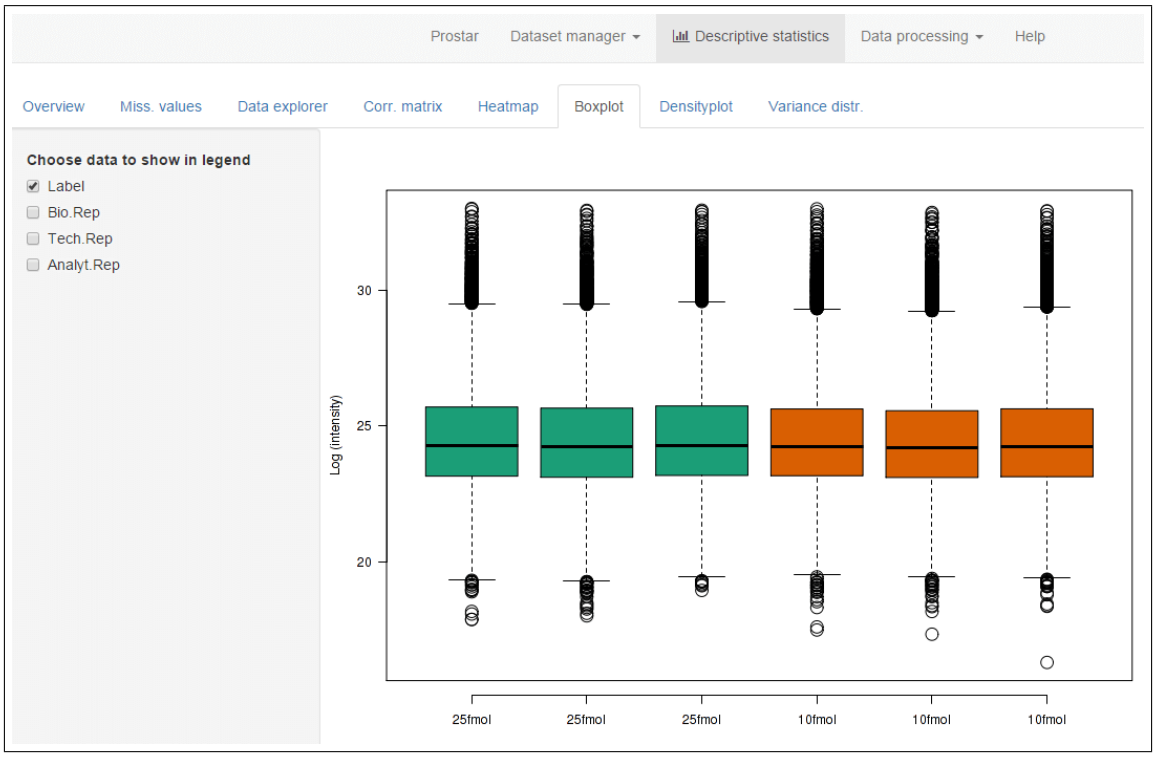

Figure 9: Boxplot (raw data)

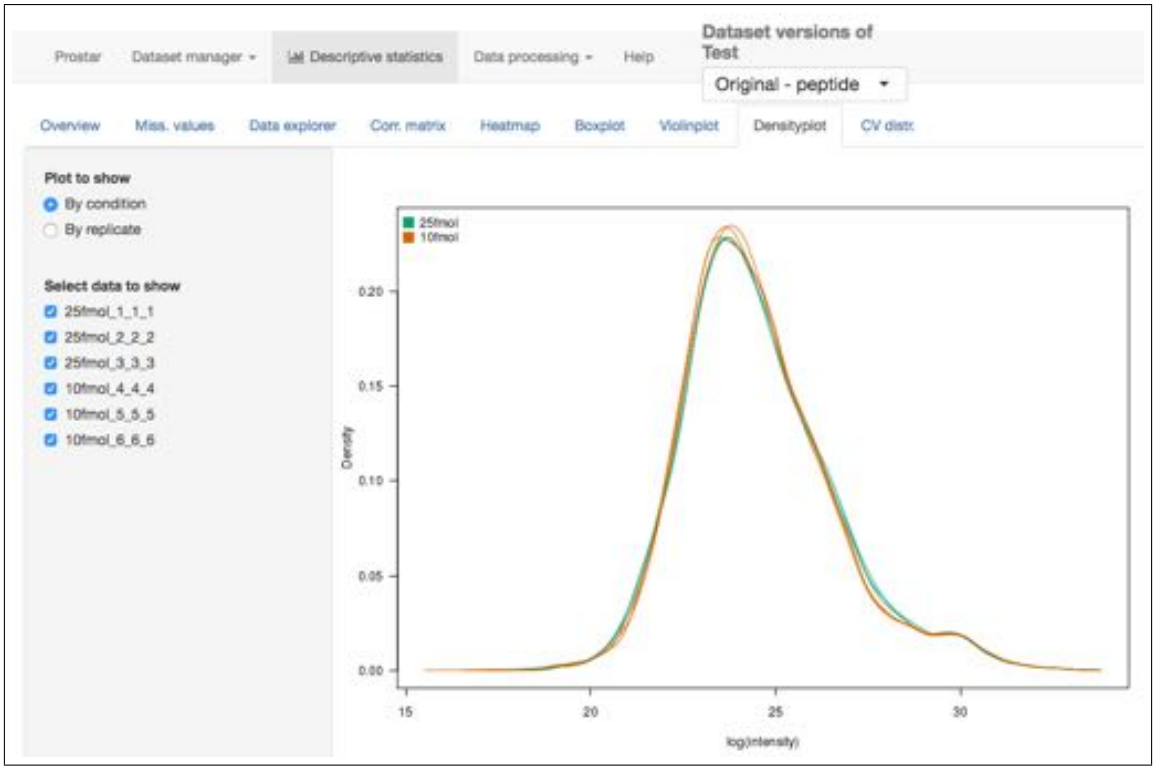

Figure 10: Density plots of the log2 raw data

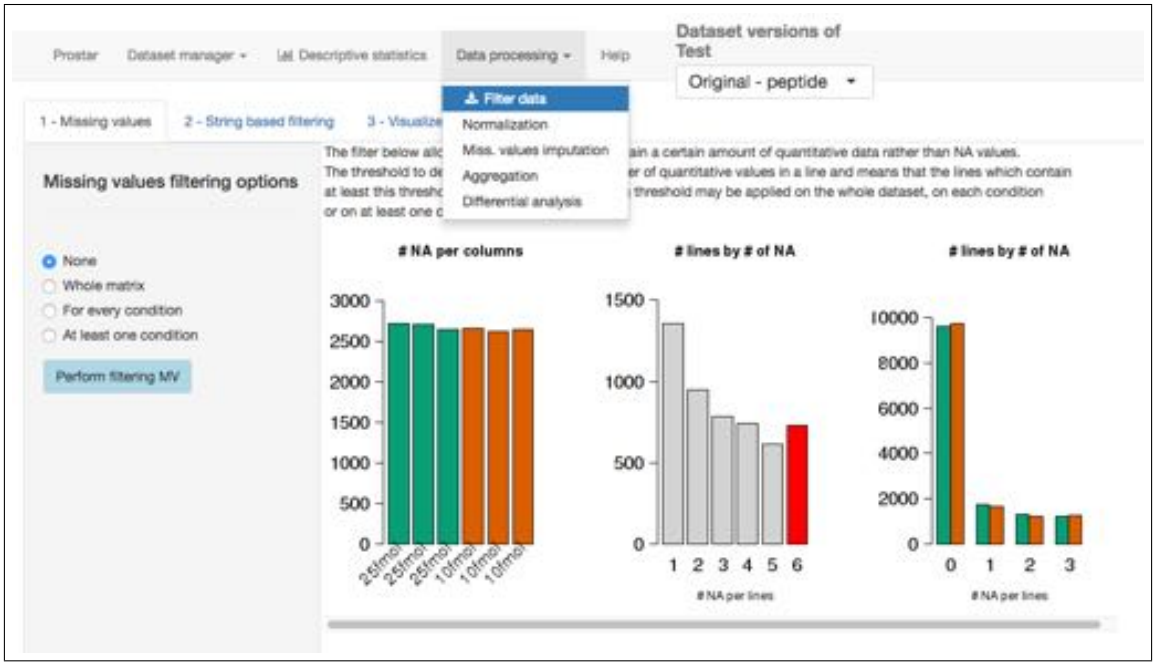

Figure 11: Screenshot of the missing value filtering panel. Graphics are interactively updated.

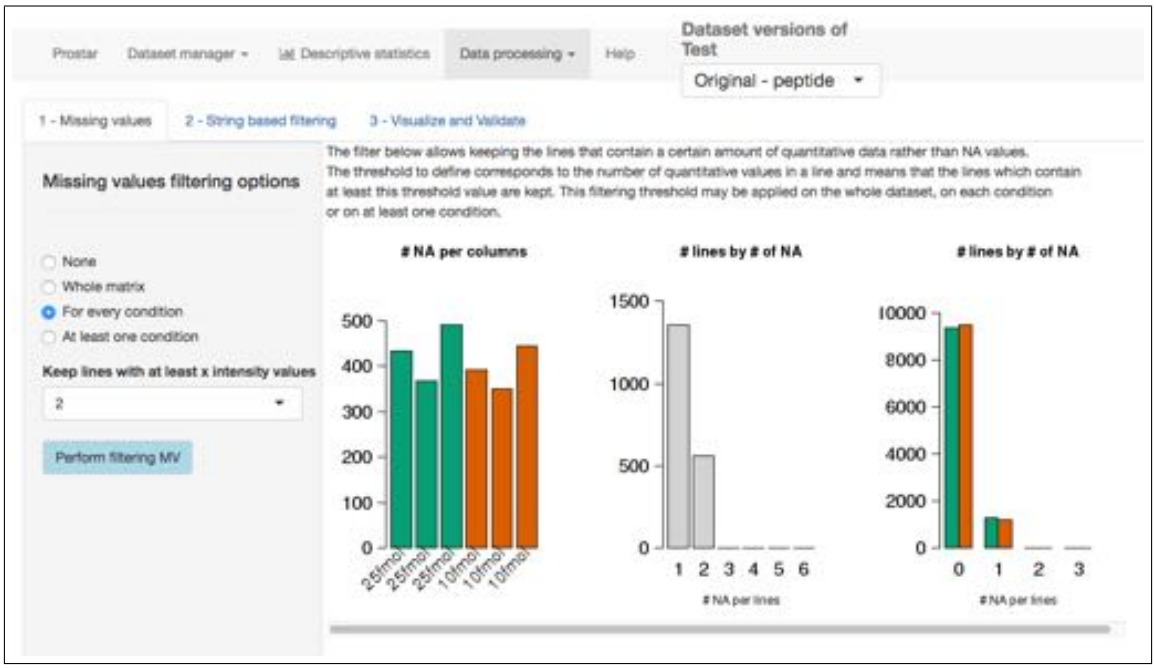

Figure 12: Screenshot of the missing value filtering panel. Graphics are interactively updated.

that point on, the field in the top-right box is updated to "**Filtered** - peptide", indicating that *ProStaR* will work on this filtered version of the dataset. As a result, all the panels are interactively updated through *ProStaR*. Hence, if one goes back to the 'Descriptive statistics' panels, all the plots will display this "**Filtered** - peptide" dataset. For example in the 'Overview' panel, it now appears that the filtered data consist of 10660 peptides (instead of 13919), with 3.88% of missing values.

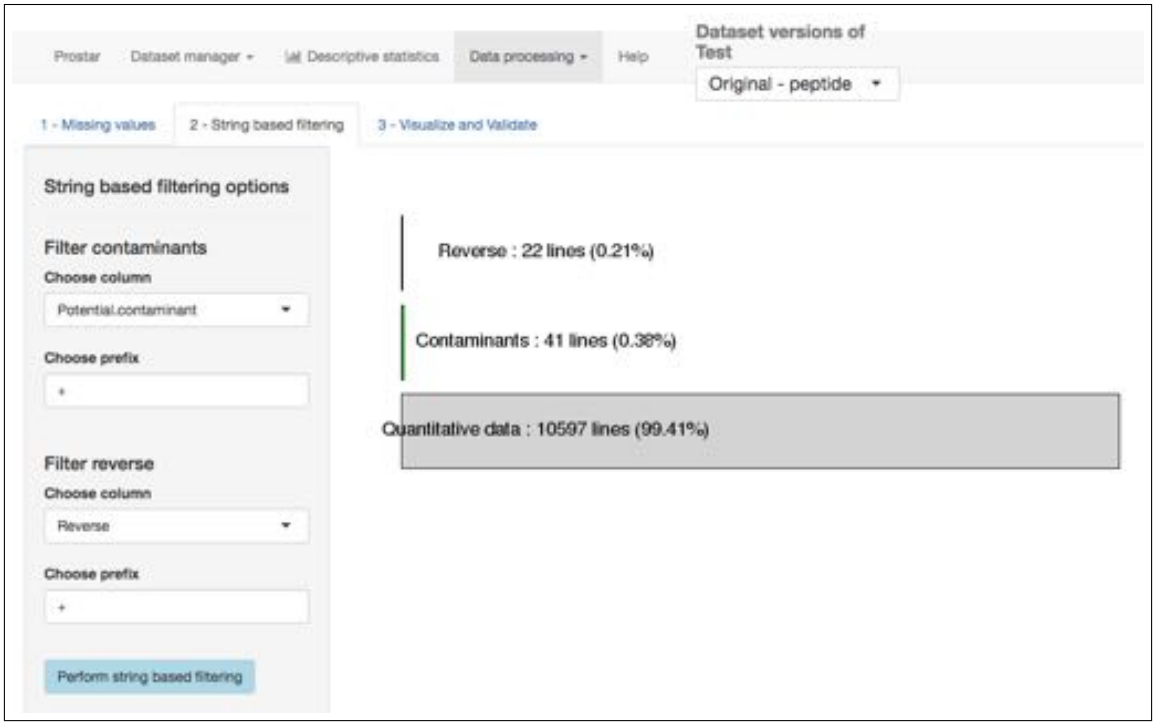

Figure 13: Screenshot of the string based filtering panel. Graphics are interactively updated.

5 Normalization

Once the data are filtered, the next step is to normalize the data (in the "Data processing" dropdown menu). It works as the previous processing steps, yet the navigation is even simpler. First, chose the adequate normalization method (here we have chosen "Quantile Centering - within conditions with quantile=0.5"); Second perfom the normalization; and finally, save it (see Fig. 16). As usual, notice that once the data are normalized, the top-right box indicates "Normalized - peptide".

6 Missing value imputation

The next step is to impute the missing values. On this panel (Fig. 17) two plots are of help to visualize the distribution of the missing values in the dataset, and to choose the adequate imputation method.

On the left hand side (Fig. 17) is displayed the number of missing values as a function of the intensity of the peptides. The missing values heatmap (on the right hand side) allows to see once-at-a-glance the possible difference of distribution of the missing values between the conditions.

On this example, we impute the missing values with the imp4p method (with 1 iteration, with imputation of LAPALA, the upper distribution bound is set to 2.5 and the distribution type used is beta). Fig. 18 represents the automatically updated plots. Once again, it is necessary to save the imputation before going on.

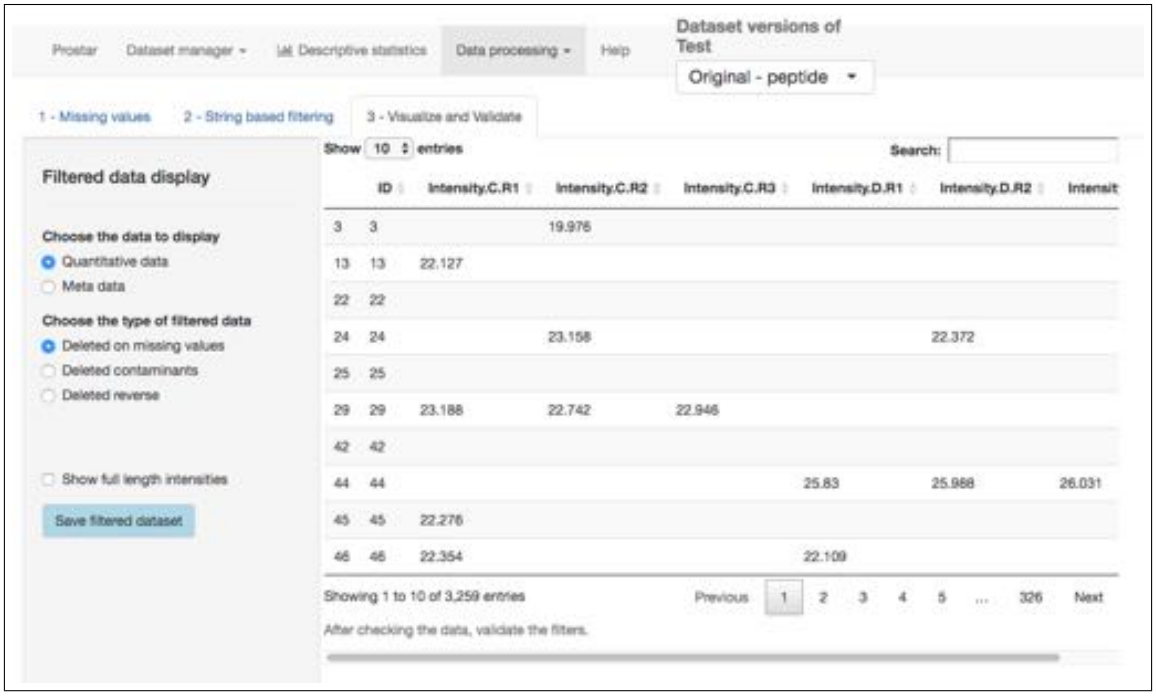

Figure 14: Panel for filter data validation

7 Protein-wise aggregation

The dataset we work on is quantified at the peptide-level. Nevertheless, the aim of a proteomics quantitative analysis is to conclude at protein-level. Thus, we aggregate the peptide intensities into protein abundance values (the available aggregation methods are described in the 'DAPAR and ProStaR user manual').

On the panel "Aggregation" of the "Data processing" drop-down menu (Fig. 19), one has first to indicate in the "Aggregate peptides" tab, the protein ID to consider (here 'Protein.group.IDs'). As soon as the protein ID is indicated, some plots and statistics provide evidences on the number of (specific or shared) peptides per proteins (on the right hand side of the panel), so as to help with the tuning of the two other parameters (Fig. 19).

After the tuning of the other parameters, namely, the aggregation method (here 'mean'), and whether shared peptides are considered or not (here 'yes'), the aggregation can be performed. A message shows up indicating the aggregation was successfully conducted (see Fig. 20):

As usual with *ProStaR*, it is necessary to save the results. This is done on the next panel "Configure protein dataset". Here, one indicates among the peptide metadata of the dataset which one will be kept and aggregated to define the protein metadata (here, we consider the column "Leading.razor.protein" - Fig. 21). Once the aggregation is saved, please check that the top right box indicates now "Aggregated - proteins". From that point on, it is possible to go back to the "Descriptive statistics" tabs, and to investigate the new protein-wise dataset. For instance, in the "Overview" tab, it appears that one has the following information:

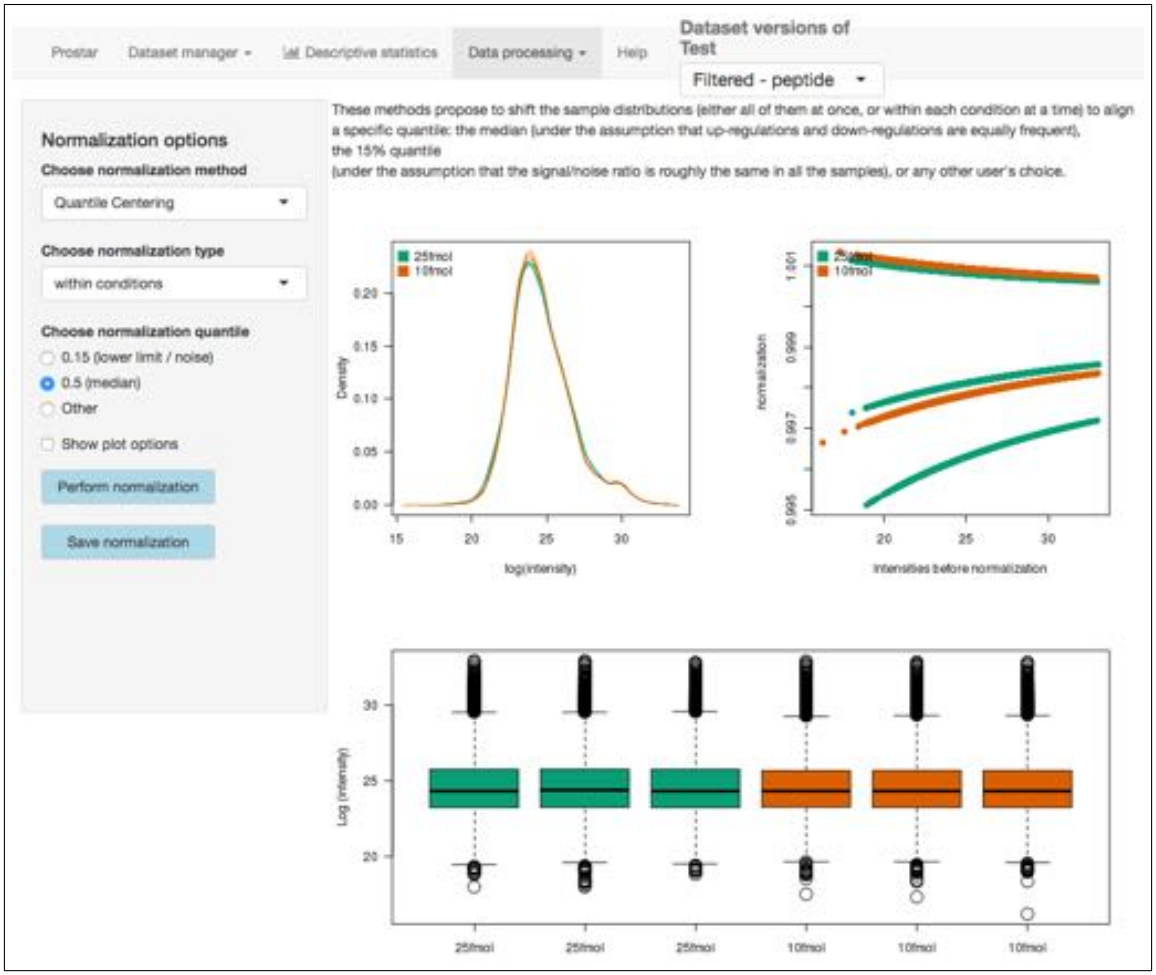

Figure 15: The normalization panel - before performing normalization

```
There is 6 samples in your data.
There is 2082 lines in your data.
Percentage of missing values: 0 %
```

8 Differential analysis

Differential analysis is the final step to identify differentially abundant proteins. This is performed on the corresponding panel of the "Data processing" drop-down menu.

The first step of the differential analysis is to build a volcano plot. This requires conducting a statistical test on each protein, so as to fit them with a p-value each (which will be used as a complement to the fold-change to construct the plot). To do so, we advise to use Limma moderated t-test. Concretely, one has to define which are the two conditions to compare (select "condition 1" and "condition 2" - in the example case, it is obvious as there are only 2 different conditions) and to select "Limma" on the drop down menu, on the left hand side of the panel. Then, the volcano plot is displayed (Fig. 22).

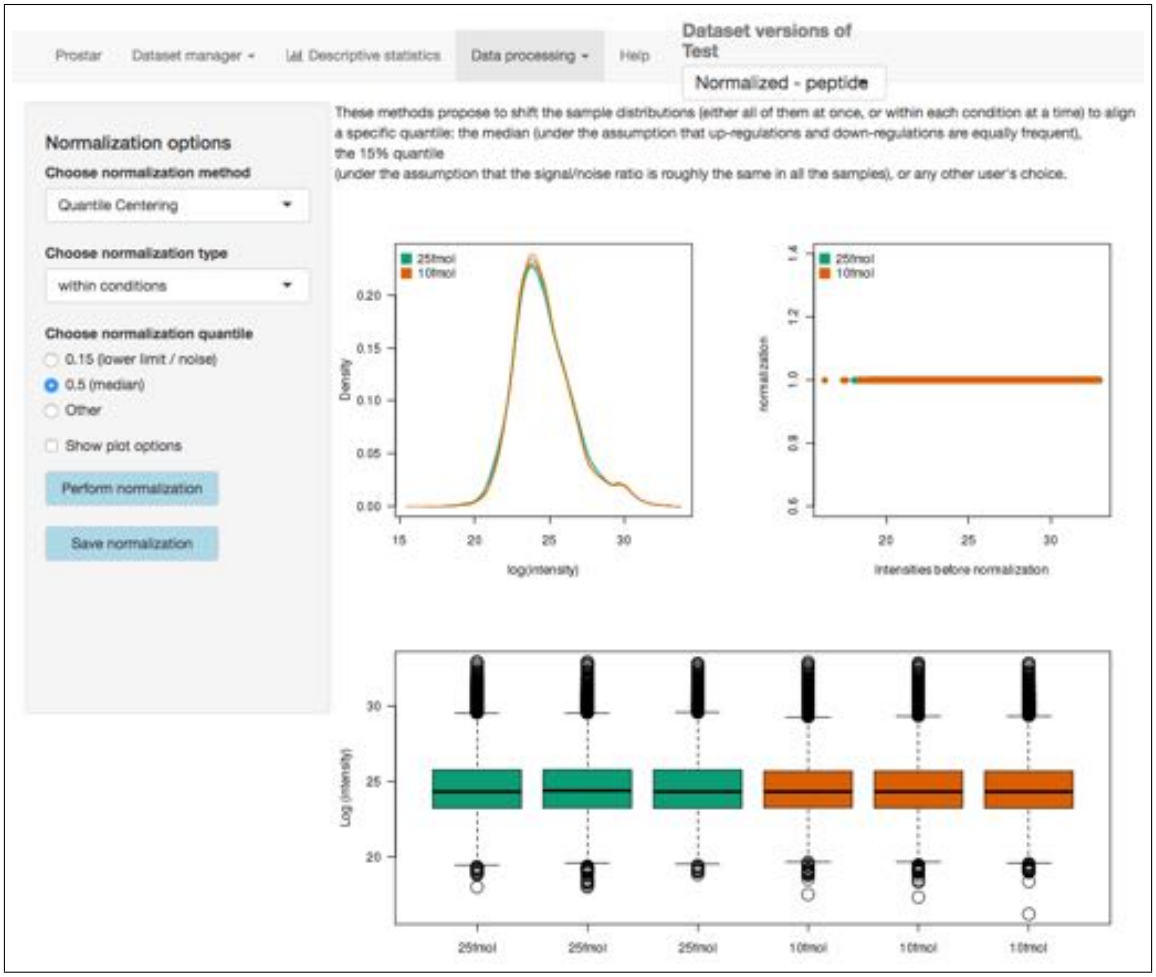

Figure 16: The normalization panel - after the normalization has been performed

Then, the practitioner is allowed to define a threshold on the fold-change, so as to filter out proteins which do not vary enough between the conditions (Fig. 23).

When the user puts the pointer of his mouse over a point of the plot, a tooltip window appears and shows some informations about that point. He can select the items to show in the Select widget where the different choices correspond to the columns of the feature meta-data table. The tooltip window is automatically updated. When the user clicks on a point, a table is displayed above the volcanoplot. It shows the values of intensities for all the samples related to the selected point. The cells colored in blue indicate that the corresponding value was a missing value in the original dataset and has been imputed, The user can click and draw a rectangle on the plot to zoom in. By clicking on the button named "Reset zoom", the user can return to the entire plot.

The **Calibrate Ana Diff** panel (fig. 24) helps the practitioner to check if the ditribution of the p-values is well-calibrated. This calls the cp4p R package functions, and we refer to the corresponding tutorial: https://sites.google.com/site/thomasburgerswebpage/download/tutorial-CP4P-4.pdf. Here, we simply use the default setting with "pounds" method.

On the **FDR** panel (fig. 25), one adjustes the p-value threshold (here to 0.0001), and apply the adaptive

Figure 17: Panel Missing Values Imputation

Benjamini-Hochberg procedure to adjust the p-values (corrected by pounds methods, see previous tab), which leads to an estimator of an upper bound of the proportion of false discoveries (referred to as False Discovery Rate, or FDR - in the example, it is estimated to 0%).

Finally, on the **Validate and Save** panel, one visualizes the selected 45 proteins and save a new dataset with only the latter ones.

9 Export

At the end of the quantitative analysis, or whenever an intermediate step is achieved along the processing pipeline, it is possible export the dataset. To do so, an "Export" panel is available in the "Dataset Manager" drop-down menu (see Fig. 26).

Let us note that the dataset which will be exported is the one that is indicated in the "Dataset versions" box. On the left of the panel ones specifies: the export format (MsnSet or Excel), and of course the name of the file exported. In the case of an Excel export, it is necessary to specify the protein or peptide

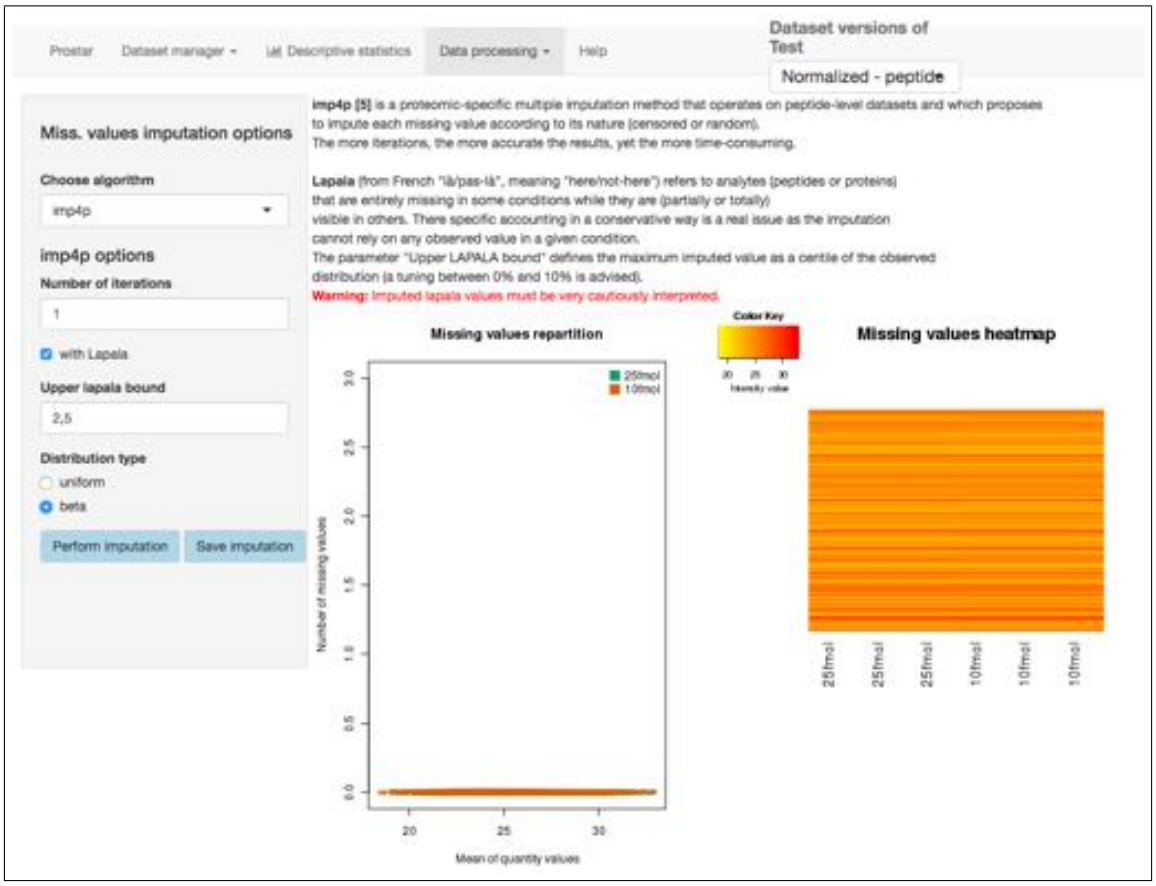

Figure 18: Panel Missing Values Imputation

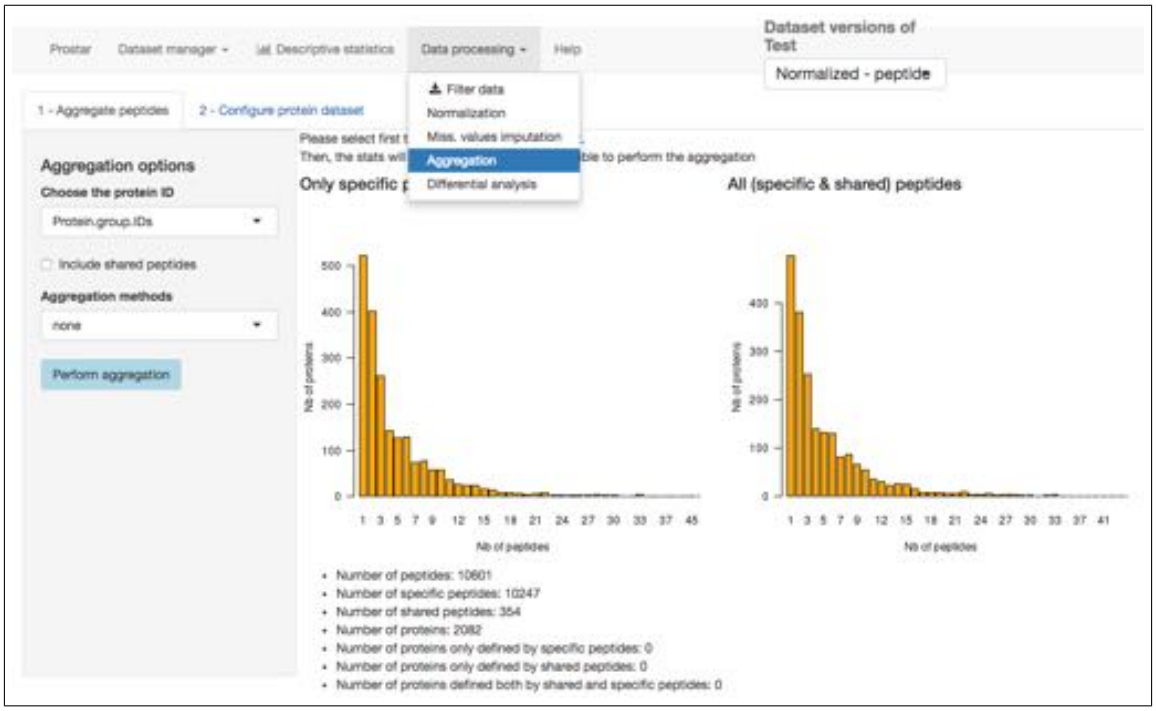

Figure 19: Aggregation step panel before aggregation

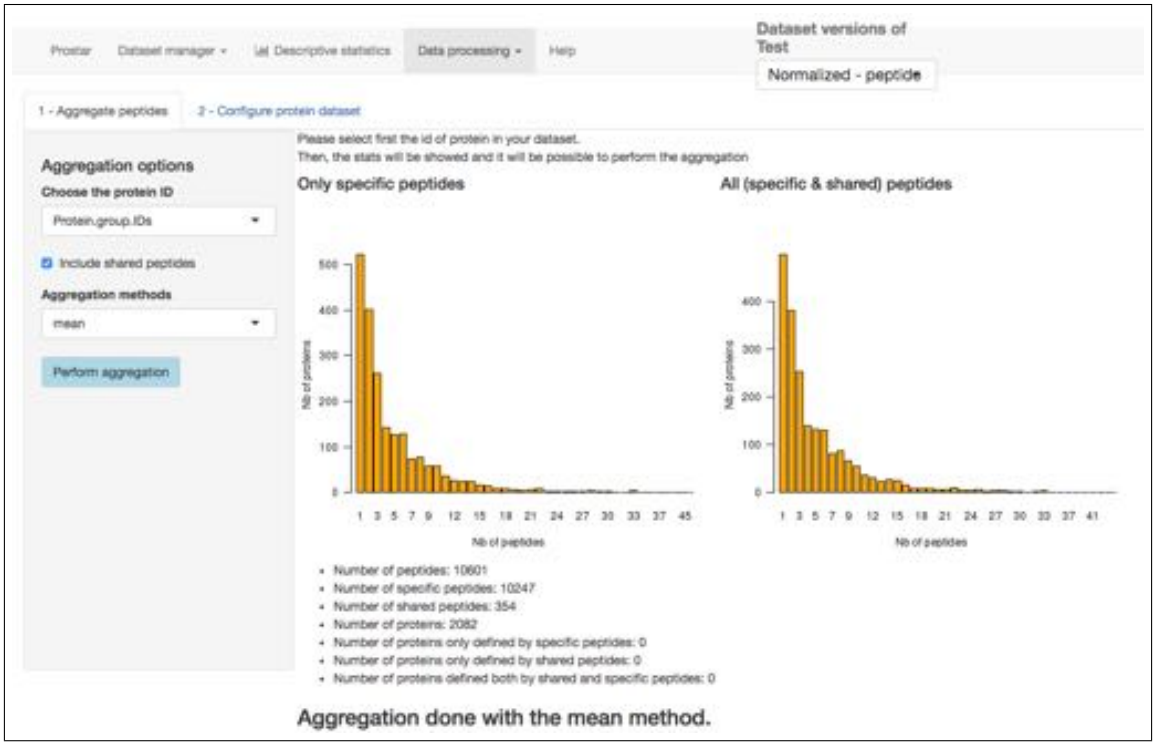

Figure 20: Aggregation step panel after aggregation

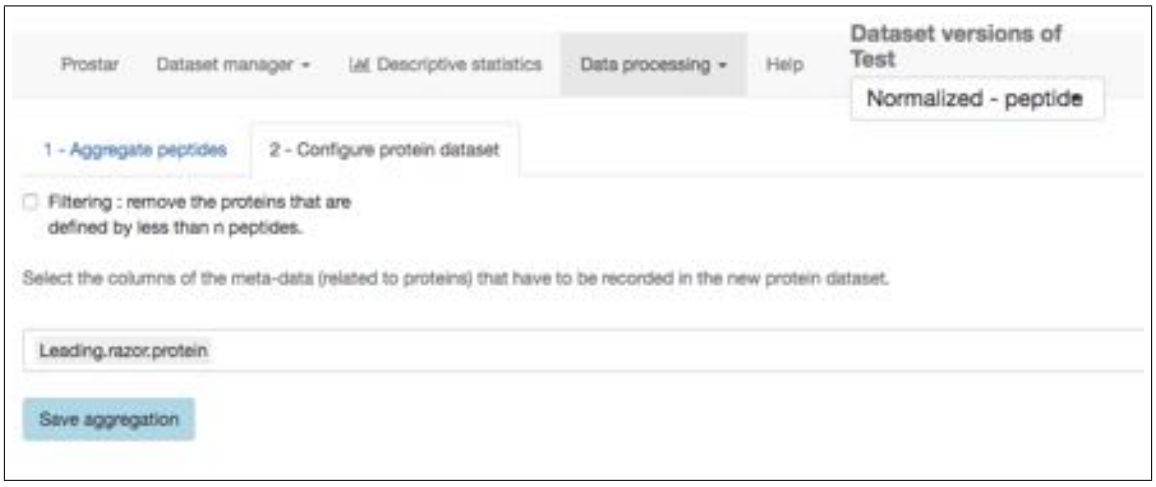

Figure 21: Aggregation step panel after aggregation

ID that is used to map the data to the meta data (the file contains 3 sheets for the quantitative data, peptide/protein metadata and the replicate metadata).

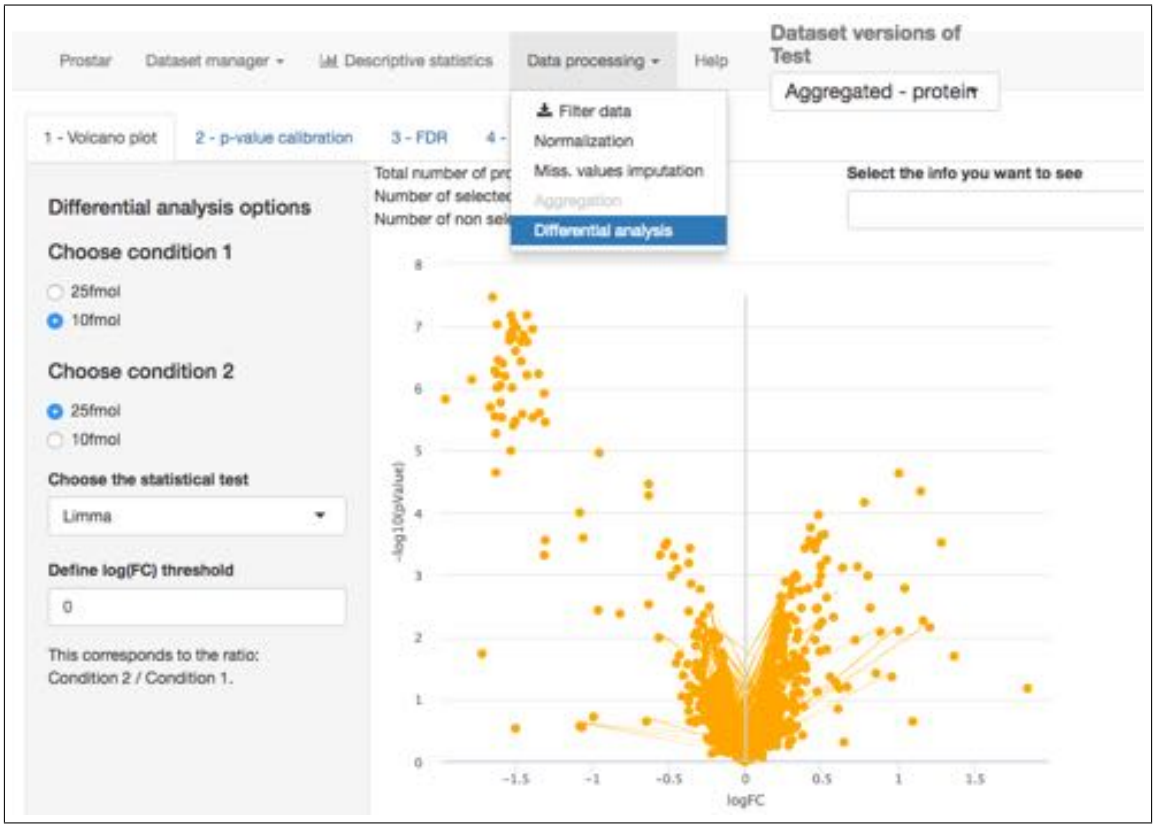

Figure 22: Vocano plot panel (Differential analysis)

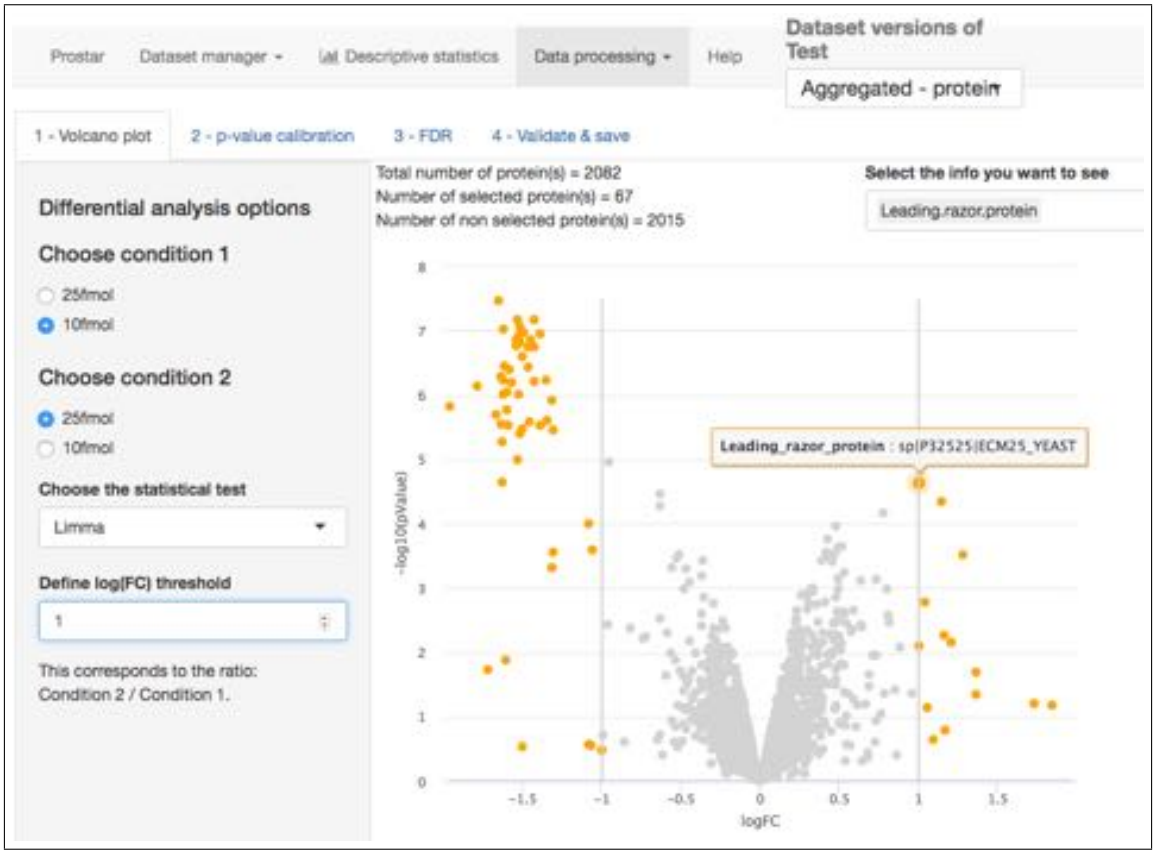

Figure 23: Vocanoplot panel (Differential analysis)

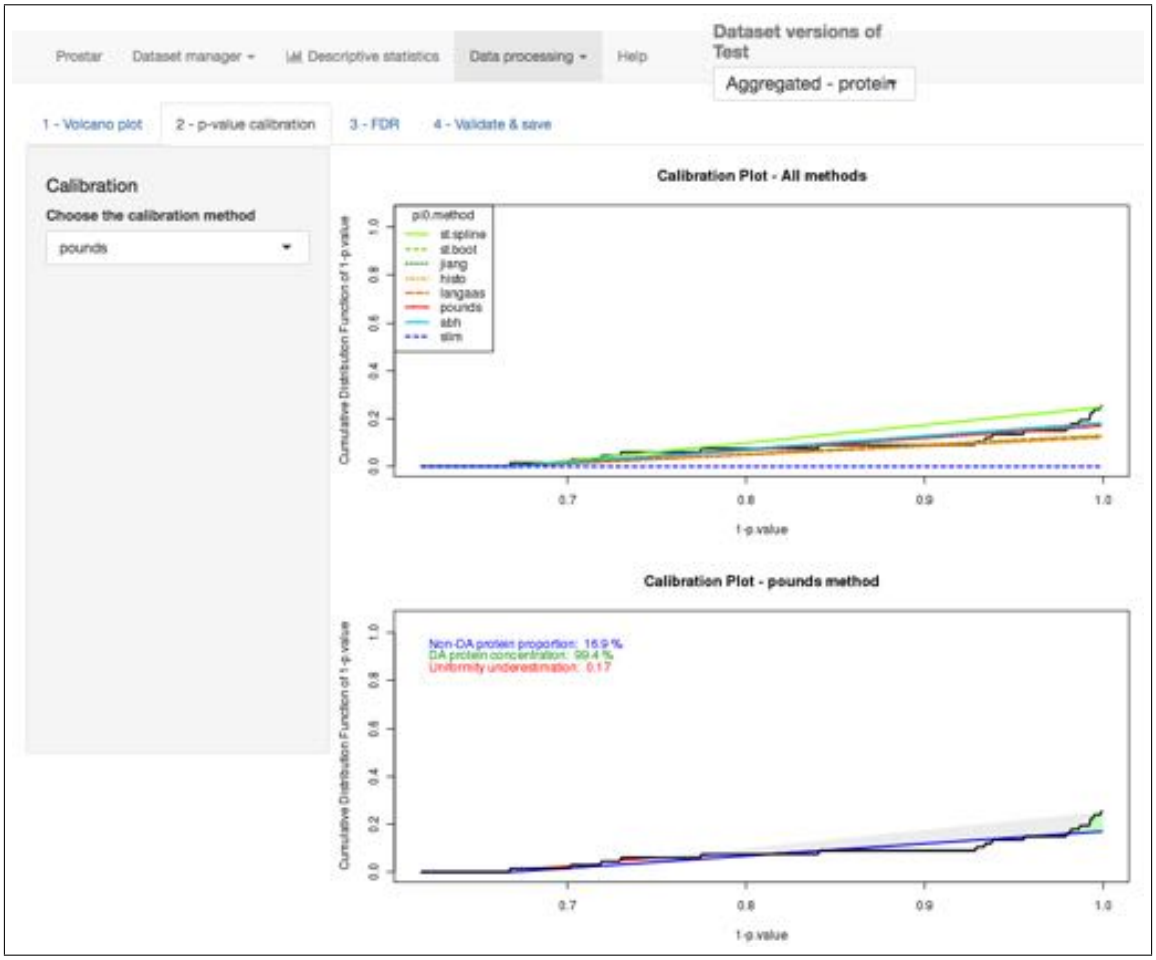

Figure 24: Calibration panel for the differential analysis)

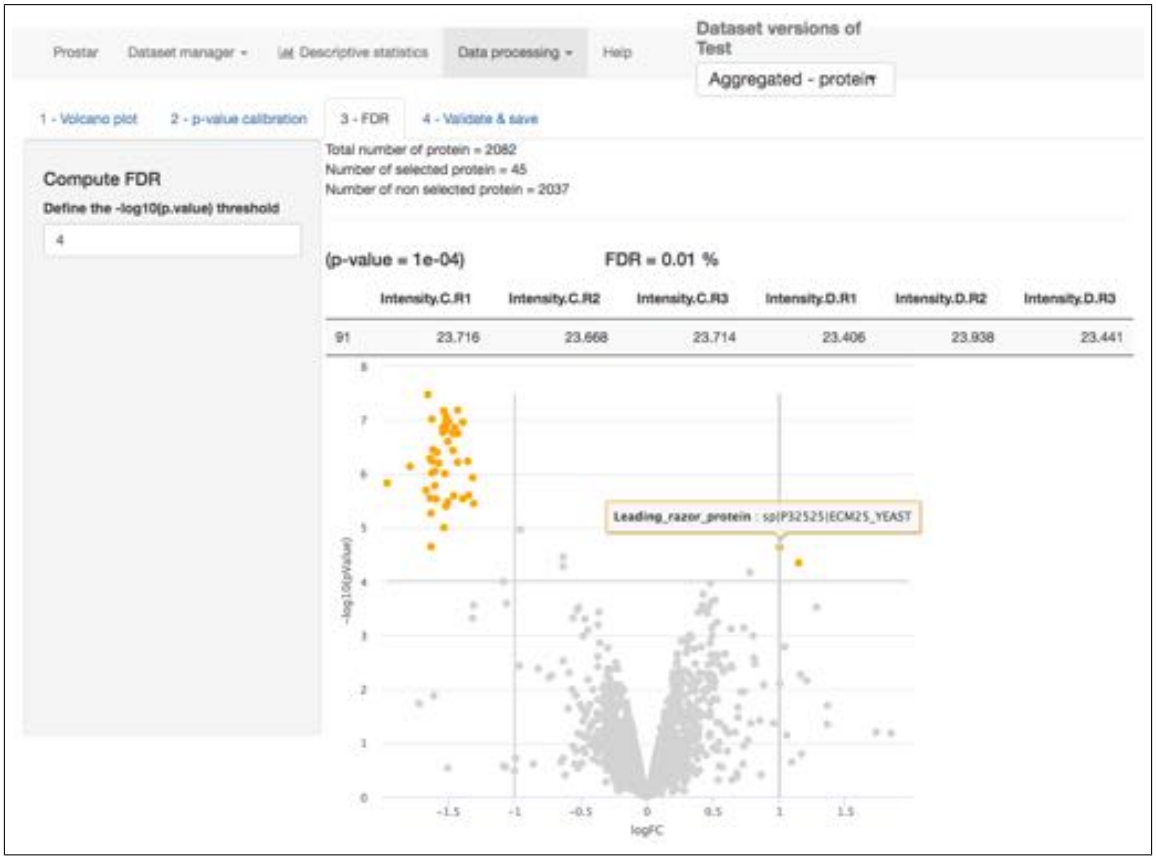

Figure 25: FDR visualization (Differential Analysis)

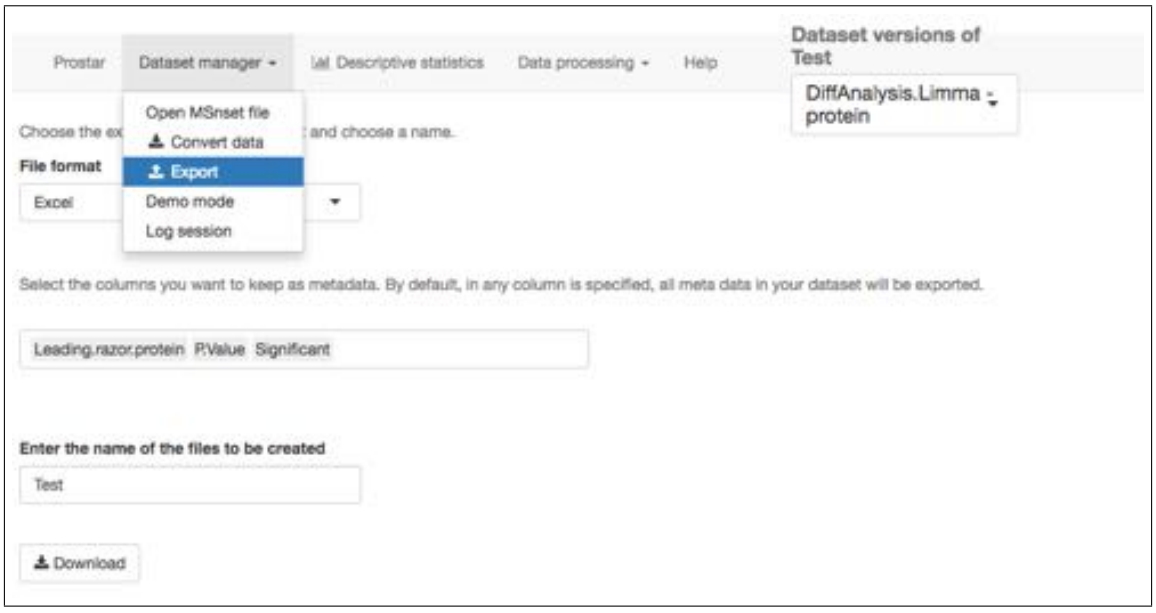

Figure 26: Export page of *ProStaR*