MetaMass: tools for mass spectrometry data meta-analysis

Jan Stuchlý, Fridtjof Lund-Johansen

August 17, 2016

jan.stuchly@lfmotol.cuni.cz

1 Introduction

The package provides tools for meta-analysis of subcellular proteomics data as described in Lund-Johansen et al Nature Methods 2016. The input is a tab-delimited text file with HGNC official human gene symbols as protein identifiers and normalized MS signal values measured in sub-cellular fractions from a single or multiple experiments. The program performs K-means clustering of the data, and the clusters are classified, scored and sorted on basis of their content of markers for subcellular localization. The program automatically generates three types of output files: 1. A cdt file for visualization of the classified dataset as a heatmap in JavaTreeView, 2. A tab-delimited text file listing all protein identifiers with their assigned subcellular location together with annotations from the Human Protein Atlas, Uniprot and GO. 3. Precision-recall curves that provide information about the fit between the dataset and the marker set.

The package is useful for the following applications: a) Mapping the subcellular location of proteins. b) Assess the performance of subcellular fractionation methods. c) Assess the fit between subcellular proteomics data and information in annotation databases.
Required software:

- R (https://cran.r-project.org)
- R Studio (https://www.rstudio.com) - for more user-friendly R-front-end
- Install MetaMass - Within R-studio use the command: install_github(”stuchly/MetaMass”)
- Open user manual from within R: vignette(”MetaMass”)
- JavaTreeView (http://jtreeview.sourceforge.net/) - for visualisation of heatmaps (the .cdt files - see below)

```
> install.packages("devtools") ##  Install devtools from R
> library(devtools) ## load devtools
> install_github("stuchly/MetaMass") ## Install MetaMass
> library(MetaMass) ## load MetaMass
> vignette("MetaMass") ## see more detailed vignette
```

2 Data input

On input users provide a tab-delimited text file containing the MS data (data.frame). For detailed instructions on data formatting see Lund-Johansen et al. Nature Methods 2016. Briefly:

- The first column must contain official gene symbols (HGNC)
- All other columns should contain numerical data only
- Values for individual sets of fractions should be normalized
- MS data from different experiments should be separated by a blank column.

3 Output

To generate output files, the user must specify the parameter output="name" and inspect the results outside R. In this case four files will be created in the working directory.

- name_table.txt - spreadsheet with separate columns for protein identifiers, markers used for the analysis, full text annotations from the Human Protein Atlas, Uniprot and GO, protein overlap in the groups, cluster assignment, assigned location, precision of subcellular mapping, and finally, the MS normalized signal values. For an example, see supplementary Table 1.15 in Lund-Johansen et al. Nature Methods 2016.
- name_pr.pdf - precision-recall curves for the markers.
- name_pr_abs.pdf - number of assigned proteins versus the precision of the subcellular mapping.
- name_javatree.cdt - heatmap to be visualized in the Java TreeView application. Each line is annotated by the protein ID, marker/annotation (Annot=;if present for this protein) and assigned location (assign=;assigned location for cluster containing this protein)

4 Metadata

The package is distributed with MS data that can be used to reproduce the results in the paper or serve as a reference for user-supplied data. These data are referred to as Metadata, and they correspond to those in supplementary Table 1.3 in the article. Metadata are not (by default) clustered or classified, but used as reference in heatmaps and the mapping table. To see a list of the meta-data files, use the command `?Metadata` Their location on the computer can be found via the system.file function. The Metadata can be used as `data.frame` or if the user want to open then in a spreadsheet editor their location on the computer can be found via `system.file` function.

```
> filename<-system.file("extdata","Bileck.txt",package="MetaMass")
> filename

[1] "/Library/Frameworks/R.framework/Versions/3.3/Resources/library/MetaMass/extdata/Bileck.txt"
```

5 Walk-through

In this section we provide a detailed description how to analyze MS data files. Users should first retrieve supplementary table 1 from Lund-Johansen et al. Nature Methods 2016 and follow instructions in the table to retrieve the following datasets as described in text box within the tables: Data_Fig2a (Table 1.13), Datasets 4, 9 and 10 (Table 1.3, use the filtering option explained in the text box in columns in columns BO:BY). The files should be saved in a new folder (e.g. named "Test") as separate tab-delimited text files and named "Data_Fig2a.txt" "study4.txt", "study9.txt" and "study10.txt", respectively. Set the folder "Test" as the working directory in R-studio. (type ctrl shift H and navigate to the folder).

5.1 Analysis option 1

Most users are likely to prefer to use the default option, since the command is very simple. The command below will generate a heatmap similar to that in Fig.2a in the article, a data table similar to that in supplementary Table 1.15 and a precision recall curves for the fit between the default marker set and the data. Users only need to modify the names for the input and output files to work with other datasets.

```
> analyze.MSfile(MSfile = "Data_Fig2a.txt", overlap=2, output = "Fig2a")
```

Explanation:
analyze.MSfile : the function used in all analyses (no modifications, allowed)
MSfile: input file (to modify : "my_input_file.txt")
Overlap: minimum protein overlap in datasets (use 1 if only one dataset is to be analyzed)
Output: prefix to name output files. (to modify "my_file_name")

Default settings:
Metadata: The "Christoforou" dataset was chosen as default because of the high resolution and coverage of proteins in cytoplasmic organelles.
Cluster size: The program automatically adjusts the numbers of clusters to obtain an average of five proteins per cluster.
Marker set: The marker set "Study 1+ Uniprot/GOoverlap" described in the article provided the best fit with MS data. This set was therefore selected as default.
The output files are found in the working directory. All output files have the prefix "Fig2a". The Fig2a.cdt file is a heatmap file (JavaTreeView), the Fig2a_table.txt is the classification result table (e.g. Excel), and the Fig2a_pr.pdf and Fig2a_pr_abs.pdf contain precision-recall curves (e.g. Acrobat Reader).

5.2 Analysis option 2

Generate recall precision curves for multiple different marker sets (i.e similar to those in Fig. 2b in the article, but with traces for all available marker sets).

```
> analyze.MSfile(MSfile = "Data_Fig2a.txt", overlap=2, output = "Fig2acurves", markers = c(3:8))
```

Explanation:
The command *markers = c(3:8)* specifies use of all marker sets that are included in the package. The marker sets are numbered as follows: 3= Christoforou+UniprotGO_overlap, 4= UniprotGO_overlap, 5 UniprotGO_sum, 6= HPA_Single_supportive, 7= HPA_Single_uncertain, Set 8 is a slim version of 3, where cytoskeleton is referred to as cytosol, and proteins in cytoplasmic organelles and membranes are referred to as membrane. This set is useful to assess precision of methods used to separate cytoplasm, membranes and nuclei. (article Fig. 2c).

Tip. *Most of the text is similar to the command used in 5.1. Use the arrow up key in R-studio to bring this command back. Simply modify the last part: output= "Fig2acurves", markers = c(3:7)) and type enter.*

Result: The Recall-Precision curves show results obtained with all the marker sets for the data in Fig 2a. The table contains columns with the mapping result obtained with all marker sets. The function generates a single heatmap corresponding to the first marker set, which in this case is the default marker set. The heatmap is therefore the same as that obtained using the default option.

5.3 Analysis option 3

Generate heatmaps to compare the mapping result obtained using different marker sets.

```
> analyze.MSfile(MSfile = "Data_Fig2a.txt", overlap=2, output = "Fig2aUniGOoverlap", markers = 4)
```

Tip. *Type arrow-up to bring back the command from example 5.2, and modify the text after output as indicated in the command. Repeat this approach with all marker sets up to 7. With this approach it takes very little time to make heatmaps for all marker sets.*

Result:
The heatmap generated using the Uniprot/GO overlap set marker set is rather similar to that obtained using the Study 1 + Uniprot/GO marker set (Fig2a article, example 5.1). With markers from the Human Protein Atlas, the area in the map assigned to the nucleus is much larger, and many of the proteins assigned to the nucleus are found in the cytoplasmic fractions. The heatmap obtained with the "uncertain" annotations from the HPA has very little structure. Thus, there is very little correspondence with the MS data. In the output table, the mapped locations are listed alongside annotations from Uniprot, GO, and the HPA.

5.4 Analysis option 4

Generate recall-precision curves for single datasets (i.e. similar to Fig. 2c in article)

```
> analyze.MSfile(MSfile = "study4.txt", overlap=1, output = "study4", markers = 8)
```

Result: The recall response curves for cytosol and nucleus are similar to those for study 4 in Fig2a.

5.5 Analysis option 5

Perform a meta-analysis of datasets 4, 9 and 10.

```
> analyze.MSfile(MSfile = c("study4.txt","study9.txt", "study10.txt"),  overlap=2, output = "study4910")
```

Explanation:
c("study4.txt","study9.txt", "study10.txt") is used to merge data from multiple files into one analysis.

Result:
The fractionation methods used in studies 4, 9 and 10 have limited resolution. However, combined they have high resolution. Thus, study 4 has high resolution of mitochondria, but poor separation of ER and nuclei. Study 9 has good separation of cytoplasmic organelles and nuclei, but no resolution of ER, mitochondria or cytosol. Study 9 has good resolution of cytosol, membranes and nuclei, but does not discriminate ER from mitochondria. When the datasets are combined, they complement each other.

5.6 Troubleshooting

The common challenge for new users of R is data import. The function `analyze.MSfile` expects tab-delimited file which could be read via function `read.table()`

```
> data_table<-read.table(filename,header=TRUE,sep="\t")
```

If there is an error concerning reading the input file, the user can try this function to check if the file is in correct format. The other possible issue is the grouping of the data columns. As mentioned above each contiguous sequence of numerical columns (flanked by non-numeric column) is considered as one group - the user can check if all (and only) the data he wants to analyze would be considered as MS data as follows

```
> colnames(data_table)[sapply(data.table,is.numeric)]
```

5.7 Custom Annotation file

By default the marker sets in data.frame `AnnotationAM` are used however the user can provide a custom Annotation file which meets the following conventions

- the file is tab-delimited (by default)
- the first column contains the Uniprot IDs
- the localizations identifiers must be empty string or syntactically valid names (i.e. a string which consists of letters, numbers, and the dot and (for versions of R at least 1.9.0) underscore characters, and starts with either a letter or a dot not followed by a number. Reserved words are not syntactic names)

The localizations will be sorted in the following order

```
> data(levelsC)
> levelsC

 [1] "CYTOSOL"              "CS"                   "RIBOSOME"
 [4] "ENDOSOME"             "LYSOSOME"             "PM"
 [7] "MEMBRANE"             "ER"                   "GOLGI"
[10] "MITOCHONDRION"        "NUCLEUS"              "EXTRACELLULAR_MATRIX"
```

and the localizations not present in `levelsC` will be sorted in lexicographical order after those present in `levelsC`.

5.8 Additional user-selectable variables

Number of groups in K-means clustering : The default option is 5 proteins per cluster. The command clusters= 250 specifies 250 clusters. Comparison metrics: The default option is Euclidean distance. The command... metric ="correlation" specifies Pearson correlation (see `?analyze.MSfile`).

6 Analysis within R

Although the main purpose of this package is to create annotated lookup tables and heatmaps which can be conveniently analyzed outside R the results can be naturally treated as any R object. The function `analyze.MSfile` returns named list containing the original data, annotation(s) and cluster assignments. Two function can be used to extract it's contents - see `?get.data` and `?get.clusters`.

```
> file2<-system.file("extdata","Data_Fig_1b.txt",package="MetaMass")
> ##cluster with respect MSfile only (cluster.metadata=FALSE by default)
> res2<-analyze.MSfile(MSfile=file2,Metadata=c("Christoforou"),output="res2",markers=c(3:5))
> data2<-get.data(res2,data.only=TRUE)
> cls2_1<-get.clusters(res2,rID=1) #rID=1 annotation with respect to markers[1]; default
> head(cls2_1)

  cluster GOLGI CYTOSOL ER MITOCHONDRION NUCLEUS PM LYSOSOME PEROXISOME CS
1       1     0       1  0             0       0  0        0          0  0
2       2     0       0  1             0       0  0        0          0  0
3       3     0       0  0             0       0  2        0          0  0
4       4     0       0  8             0       0  0        0          0  0
5       5     0       0  0             0       1  0        0          0  0
6       6     0       0  0             0       1  0        0          0  0
  ENDOSOME RIBOSOME Nb_of_annotations precision_main_component main_component
1        0        0                 1                         1        CYTOSOL
2        0        0                 1                         1             ER
3        0        0                 2                         1             PM
4        0        0                 8                         1             ER
5        0        0                 1                         1        NUCLEUS
6        0        0                 1                         1        NUCLEUS
  GOLGI_ratio CYTOSOL_ratio ER_ratio MITOCHONDRION_ratio NUCLEUS_ratio PM_ratio
1           0             1        0                   0             0        0
2           0             0        1                   0             0        0
3           0             0        0                   0             0        1
4           0             0        1                   0             0        0
5           0             0        0                   0             1        0
6           0             0        0                   0             1        0
  LYSOSOME_ratio PEROXISOME_ratio CS_ratio ENDOSOME_ratio RIBOSOME_ratio
1              0                0        0              0              0
2              0                0        0              0              0
3              0                0        0              0              0
4              0                0        0              0              0
5              0                0        0              0              0
6              0                0        0              0              0
  assigned_location Nb_main_component Nb_assigned_location
1           CYTOSOL                 1                    1
2                ER                 1                    1
3                PM                 2                    2
4                ER                 8                    8
5           NUCLEUS                 1                    1
6           NUCLEUS                 1                    1
  precision_assigned_location updated_order
1                           1            82
2                           1           357
3                           1           221
4                           1           288
5                           1           617
6                           1           618
```

Here we have extracted the data accompanied only by the protein ID and the cluster ID together with the analysis results with respect to the first marker set. As the the clusters in the `cls2_1 data.frame` are ordered by the cluster ID we can add any information to the data e.g.

```
> data2<-data.frame(data2,main_component1=cls2_1$main_component[data2$cluster])
```

7 Number of clusters

It is obvious that the classification recall depends strongly on the number of clusters - the recall grows as the average cluster size decreases (with 100% agreement with the annotation when each cluster contains just 1 protein) however the number of unclassified proteins grows as well. To asses this fact we analyzed the data in examples 1-3 with different number of clusters (starting with average cluster size around 2000 - the exact value depends on the size of dataset - and decreasing the size to 2) - figure 1 The solid red line (example 1) shows almost perfect reconstruction of classification with as much as 100 proteins per cluster (left panel) and since in this example only annotated proteins were used no unclassified proteins. The results for the data in examples 2 and 3 show consistently no loss of resolution for cluster size above 25.

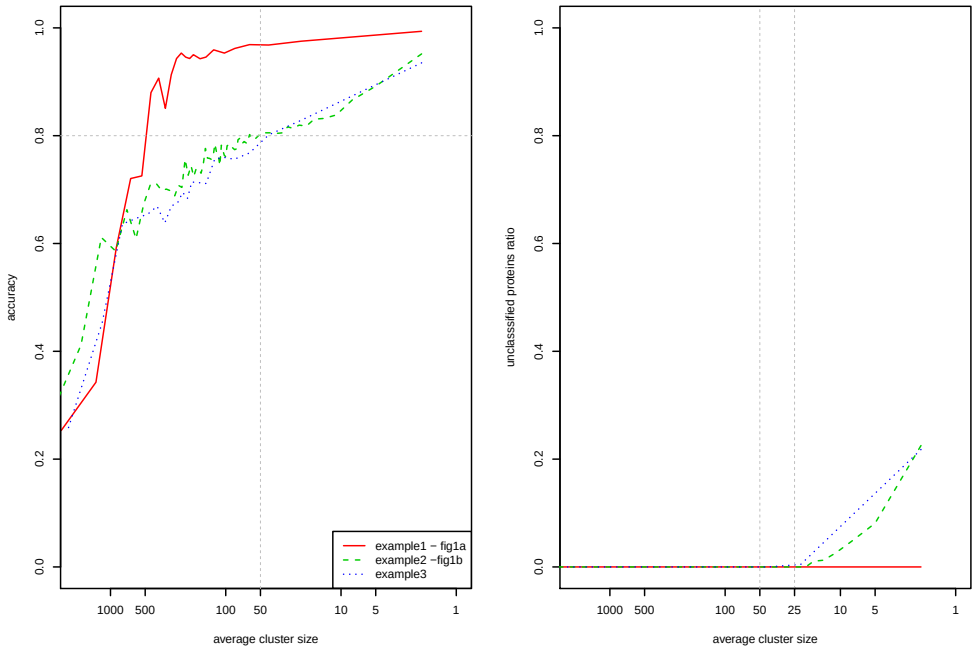

Figure 1: Performance against average cluster size