MetaMass: tools for mass spectrometry data meta-analysis

Jan Stuchlý, Fridtjof Lund-Johansen

April 25, 2016

jan.stuchly@lfmotol.cuni.cz

1 Introduction

The presented package provides tools for meta-analysis of human proteome mass spectrometry (MS) data as described in (ref Lund-Johansen et al 2016). The purpose is to provide stand-alone tool for analyzing text files or `data.frame` with an integration with R for further analysis.
Area of usage: MetaMass is a tool for meta-analysis of sub-cellular proteomics data (ref Lund-Johansen et al 2016). Users can analyze mass spectrometry (MS) data within the context of published datasets and sets of markers identified by mining of MS datasets or curated annotations from Uniprot, GO and the Human Protein Atlas. The input is text files with official human gene symbols as protein identifiers and normalized MS signal values measured in sub-cellular fractions. The output is a cdt file for visualization of the classified datasets as heatmaps in JavaTreeView, a table with with gene names, annotations, assigned locations and purity scores and precision-recall curves that provide information about the fit between the dataset and the marker set.
Required software:

R (https://cran.r-project.org)

R Studio (https://www.rstudio.com) - for more user-friendly R-front-end

Rtools (https://cran.r-project.org/bin/windows/Rtools/) necessary on MS Windows operating system only

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 this vignette
```

2 Input

On input user provides a `data.frame` containing the MS data or path to text file(s) with the data. Data are then processed with respect to following conventions:

- All columns containing numerical values only are used as MS data
- MS data could be divided into different groups" if separated by blank/non-numeric column i.e. numeric columns flanked by non-numeric columns are understood as a single "group". With more file on input each file is considered as separate set of groups.
- `data.frame` must contain a column containing protein ID which match the annotation file (by default the genename in the first column;see below)

- by default tab-delimited text file is assumed

An annotation file is provided in the package

```
> library(MetaMass)
> data(AnnotationAM)
> head(AnnotationAM)

  Uniprot.Accession   Gene Christoforou Uniprot   G.O HPA.supportive
1            P04217   A1BG
2            Q9NQ94   A1CF                                  Nucleus
3            P01023    A2M
4            A8K2U0  A2ML1
5            Q9NPC4 A4GALT               Golgi Golgi
6            Q9UNA3  A4GNT               Golgi Golgi
  HPA.uncertain
1
2
3
4
5
6
```

2.1 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)]
```

3 Output

By default (with parameter `output=NULL`) the user level function `analyze.MSfile` does not create any files in the working directory and returns named list to be analyzed within R (see below). However most of the users are expected to specify the parameter `output="name"` and inspect the results outside R. In this case three files will be created in the working directory.

- name_table.txt - spreadsheet containing the analyzed data and all annotations used (see below) for the analysis together the cluster assignment and purity with respect to the most abundant component
- name_pr.pdf - precision-recall curves of the cluster assignment with respect to all used annotations
- name_javatree.cdt - heatmap to be visualized in the Java TreeView application. Each line is annotated by the protein ID, annotation (Annot=;if present for this protein) and assignment (assign=;assignment of the cluster containing this protein)

4 Metadata

The package is distributed with MS data which could be used to reproduce the results in the paper or as a reference for user supplied data - these data are called Metadata - see `?Metadata.` 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] "/Users/grigorij/Library/R/3.2/library/MetaMass/extdata/Bileck.txt"
```

5 Analysis options

See `?analyze.MSfile` for detailed information.

5.1 Analysis option 1

Map proteins in the dataset with unknown locations and assess the precision of the mapping. The test dataset is clustered and classified with the Christoforou markers (default option). The metadata is not clustered/classified but aligned "passively" to serve as a reference (default option).

```
> Res1= analyze.MSfile(MSfile="filename.txt", Metadata= "Christoforou", output="myfile2")
```

Explanation: This is the simplest command for data analysis. The program applies a number of default options. Users therefore only have to specify the test file, the metadata and the prefix for the outputfiles. Below we explain how users can write commands to vary a number of parameters. MS file: test dataset, name of text file in working directory Metadata: MetaMass contains data from studies analyzed in Lund-Johansen et al. 2016 (see `?Metadata`). The Christoforou dataset was obtained using density gradient centrifugation of intact organelles, and this set has the highest resolution. Output: user-defined prefix for the output files (if output= "myfile" the output files will be named "myfile_table.txt", "myfile_javatree.cdt", myfile_pr.pdf")

5.2 Analysis option 2

Compare the information in the dataset to that in annotation databases such as Uniprot and the Human Protein Atlas. This option is primarily useful to obtain a table with information about how the proteins are annotated in Uniprot, GO and the HPA.

```
> Res2= analyze.MSfile(MSfile="filename.txt", Metadata= "Christoforou", output="myfile2",markers=c(3:7))
```

Explanation: Users can select to analyze their data in context of marker sets other than the Christoforou markers. The marker sets are specified by numbers, 3= Christoforou, 4= Uniprot, 5= GO, 6= Human Protein Atlas (HPA) supportive, 7= HPA uncertain. (see)

```
> ?AnnotationAM
```

By specifying markers as c(3:7), the output table will contain columns for annotations from Uniprot, GO and the Human Protein Atlas. The heatmap will still be organized according to the mapping result using the Christoforou marker set.

5.3 Analysis option 3

Visualize how proteins with well-defined locations are distributed in the subcellular fractions in the test dataset. The Christoforou dataset is organized according to the localizations mapped in the original publication. The test dataset is "passively" aligned.

```
> Res3= analyze.MSfile(MSfile="filename.txt", Metadata= "Christoforou", output="myfile3",group=0,cluster.metadata=TRUE)
```

group=0 specifies that the test dataset will not be clustered, cluster.metadata= TRUE indicates that the metadataset will be clustered (default=FALSE)

5.4 Analysis option 4

Compare the partitioning in the dataset to results obtained using different types of fractionation methods.

```
> Res4= analyze.MSfile(MSfile="filename.txt", Metadata= "Larance", output="myfile4")
```

Explanation: The "Larance" dataset was obtained using the popular Pierce kit for subcellular fractionation. The "Thakar" dataset was obtained using a similar method (differential detergent fractionation). The "Bileck" dataset was obtained by lysing cells in the presence of 0.5% Triton-X100. The "Carvalho" dataset is relevant for results obtained with detergent-free separation of cytosol, nuclei and mitochondria. Note that a single dataset with results obtained with a method to separate cytosol, nucleus and membranes is not suitable since the number of clusters will be too low.

5.5 Additional user-selectable variables

Number of groups in K-means clustering : The default option is 500 groups. The optimal number of proteins per group is 10-20. If the dataset has more than 5000 proteins or less than 3000 proteins users may specify the number of groups accordingly. The command clusters= 250 specifies 250 clusters. Comparison metrics: The default option is Eucledian 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 lokup 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 CYTOSOL MITOCHONDRION NUCLEUS PM LYSOSOME PEROXISOME ER CS
1       1       0             5       0  0        0          0  0  0
2       2       0             0       0  1        0          0  0  0
3       3       0             0       0  3        0          0  0  0
4       4       0             0       5  0        0          0  0  0
5       5       0             1       0  0        0          0  1  0
6       6       0             7       0  0        0          0  0  0
  EXTRACELLULAR_MATRIX RIBOSOME ENDOSOME Nb_of_annotations
1                    0        0        0                 5
2                    0        0        0                 1
3                    0        0        0                 3
4                    0        0        0                 5
5                    0        0        0                 2
6                    0        0        0                 7
  purity_main_component main_component CYTOSOL_ratio MITOCHONDRION_ratio
1                   1.0  MITOCHONDRION             0                 1.0
2                   1.0             PM             0                 0.0
3                   1.0             PM             0                 0.0
4                   1.0        NUCLEUS             0                 0.0
5                   0.5  MITOCHONDRION             0                 0.5
6                   1.0  MITOCHONDRION             0                 1.0
  NUCLEUS_ratio PM_ratio LYSOSOME_ratio PEROXISOME_ratio ER_ratio CS_ratio
1             0        0              0                0      0.0        0
2             0        1              0                0      0.0        0
3             0        1              0                0      0.0        0
4             1        0              0                0      0.0        0
5             0        0              0                0      0.5        0
6             0        0              0                0      0.0        0
  EXTRACELLULAR_MATRIX_ratio RIBOSOME_ratio ENDOSOME_ratio assigned_location
1                          0              0              0     MITOCHONDRION
2                          0              0              0                PM
3                          0              0              0                PM
4                          0              0              0           NUCLEUS
5                          0              0              0                ER
6                          0              0              0     MITOCHONDRION
  Nb_main_component Nb_assigned_location purity_assigned_location updated_order
1                 5                    5                      1.0           297
2                 1                    1                      1.0           176
3                 3                    3                      1.0           147
4                 5                    5                      1.0           396
5                 1                    1                      0.5           255
6                 7                    7                      1.0           284
```

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 Examples

The simplest way to use this package is the wrapper function `analyze.MSfile` (see `?analyze.MSfile`). This function reads and process your the tab-delimited text file with MS data and stores the results in the working directory. Set working directory: In R-studio go to menu: Session, select set working directory, identify the folder where your data are stored. The output files will also appear in this folder. After setting the working directory, it is necessary to provide the path to the tab-delimited file with MSdata. As the first example we use the data from figure 1 in the paper without the first (reference;metadata) columns.

```
> file1<-system.file("extdata","Data_Fig_1a.txt",package="MetaMass")
```

The function MetaMass recognizes two types of data - `MSfile` and `Metadata`. The `MSfile` are the actual data to be analyzed whereas the Metadata stand for internally stored MSdata which can be used as reference (see `?Metadata`).

```
> ##proteins identified by gene-name -> annotation.ID=2 (see ?AnnotationAM)
> ##cluster with respect metadata only (group=0)
> res1<-analyze.MSfile(MSfile=file1,Metadata=c("Christoforou"),output="res1",group=0,cluster.metadata=TRUE)
```

In this case (compare the command to the `usage` section in `?analyze.MSfile`) we use the default annotation, as Metadata we use the Christoforou, Mulvey, Breckels, et al. (2016) dataset and the output will be stored in files starting with the string 'res'. As a toy example (to get the similar result as in figure 1a) we want to cluster only the metadata and align the `MSfile` - do this we set `group=0` which says that none of the data in MSfile should be clustered and we have to specify that want cluster the metadata `cluster.metadata=TRUE`). As the second example let us reconstruct the figure 1b in the paper. Here we want to cluster the MSfile (which is default and we need not specify the `group` parameter) and align the metadata to it.

```
> 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")
```

The results are stored in the working directory in files starting with 'res2'. As the third example we show how to use multiple files as input.

```
> ##compare multiple files component fractionation with Metadata
> files1<-system.file("extdata",c("Bileck.txt","Thakar.txt","Carvalho.txt"),package="MetaMass")
> res3<-analyze.MSfile(MSfile=files1,Metadata=c("Christoforou"),output="res3")
>
```

As the last example let us show how to use multiple annotations (one can use custom annotation file if it corresponds the appropriate format). Let us use all but the 3rd annotation i.e. columns 3,4,6,7. And as another example of option we use 480 cluster.

```
> res4<-analyze.MSfile(MSfile=file2,Metadata=c("Christoforou"),output="res2_4annot",clusters=480,markers=c(3,4,6,7))
```

User can plot the precision-recall curves directly by (figure 1)

```
> par(mfrow=c(5,3),mar=c(1, 4, 2.2, 1) + 0.1,cex=0.45)
> plot.prAM(res4) #plot in  5 rows and 3 columns
```

And the heatmaps as seen in the Jva TreeView application are shown in the figure 2

8 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 3. 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.

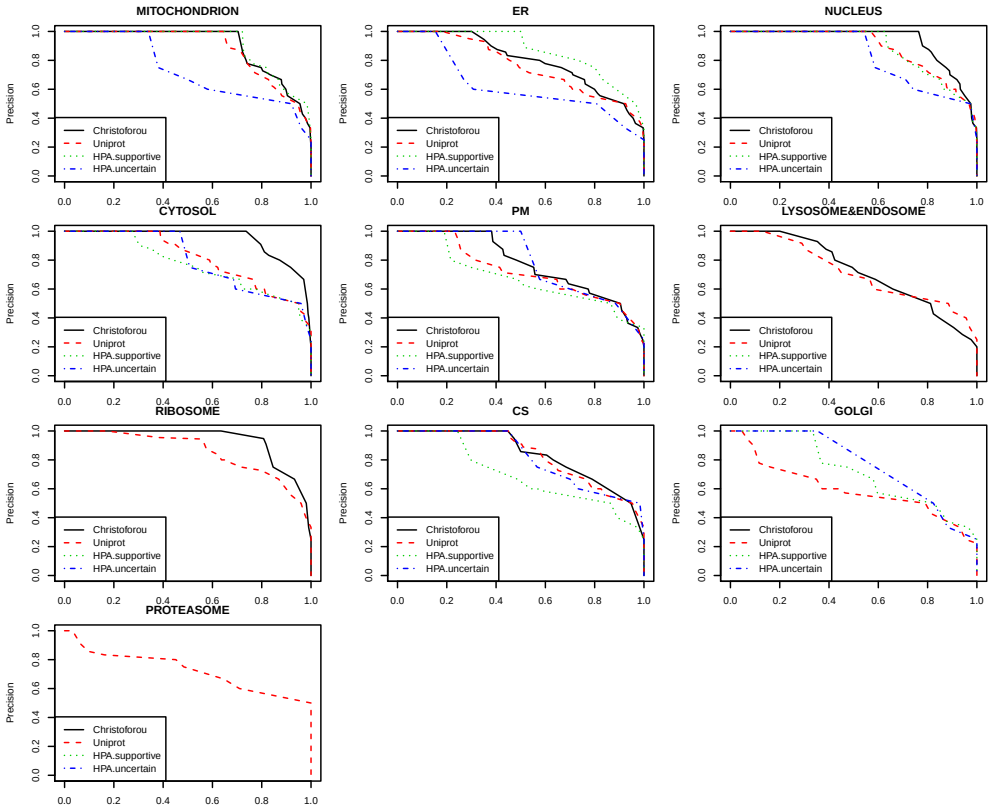
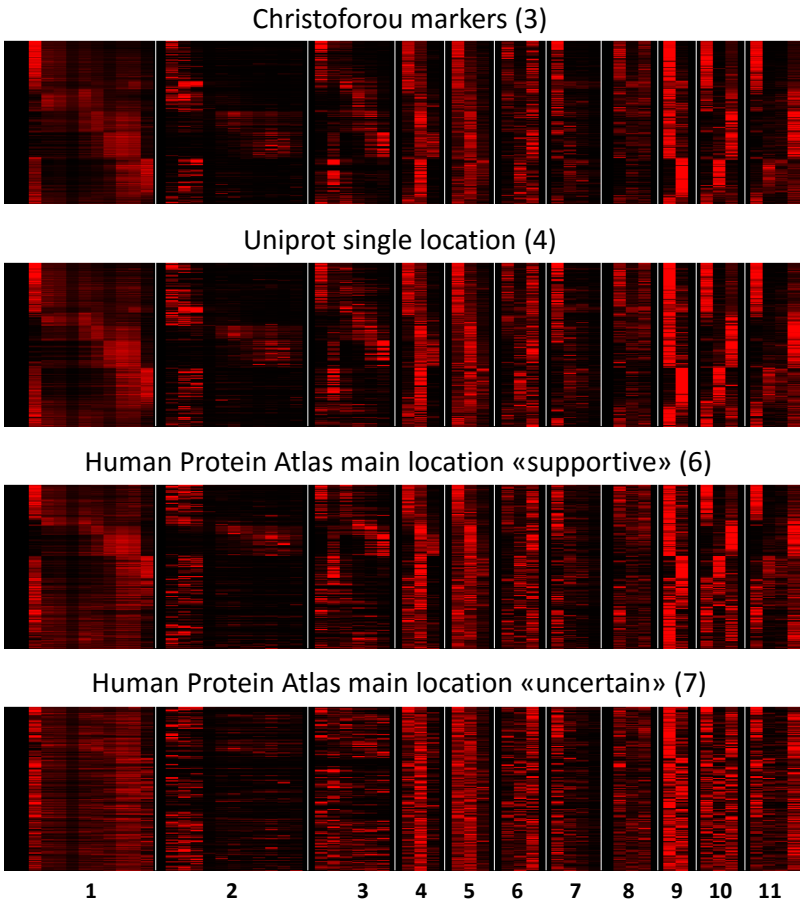


Figure 1: Precision-recall curves



The heatmaps show results obtained when datasets from studies 2-11 were classified on basis of indicated marker sets. Study 1 (*Christoforou et al.*) represents the metadata. This dataset was not used for the analysis, but aligned afterwards (default option in MetaMS). The numbers in parenthesis represent the corresponding options for the markers = argument in MetaMS. Results obtained with the Uniprot annotations are similar to those obtained with the Christoforou markers. The «supportive» annotations from the Human Protein Atlas (HPA) classify a larger number of proteins as nuclear, while the HPA «uncertain» annotations yield a heatmap with no obvious pattern. This demonstrates that these annotations have a very poor fit with the MS data and the other marker sets.

Figure 2: heatmaps as seen in JavaTree view

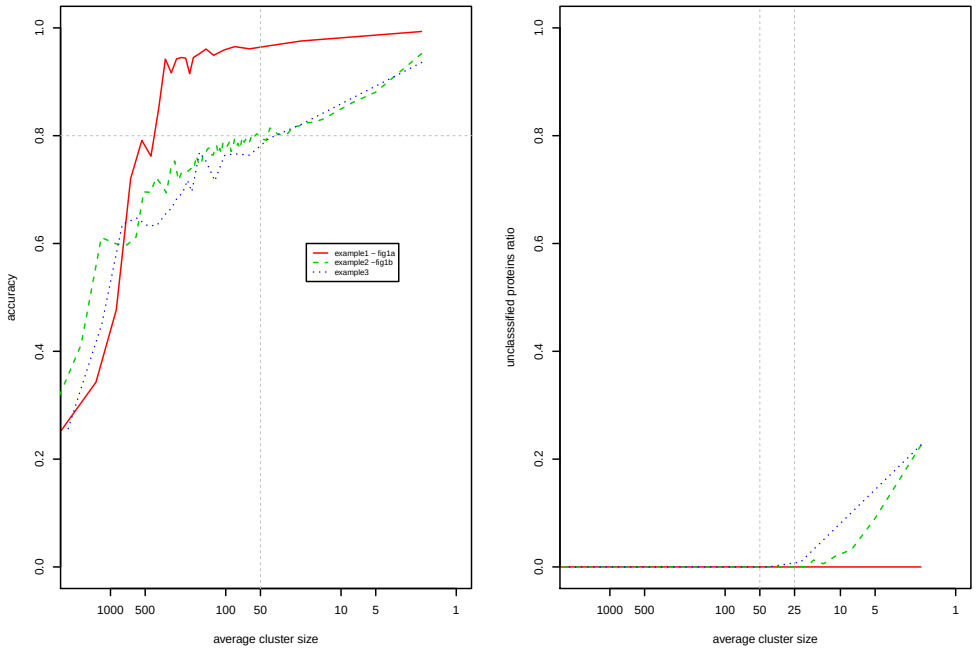


Figure 3: Performance against average cluster size