MetaMass: tools for mass spectrometry data meta-analysis

Jan Stuchlý, Fridtjof Lund-Johansen

May 18, 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 precision 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

- trailing spaces in character columns are removed

An annotation file is provided in the package

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

  Uniprot   Gene Christoforou_Uniprot_G.O_overlap Christoforou_Uniprot_G.O_sum
1 P04217    A1BG
2 Q9NQ94    A1CF
3 P01023     A2M
4 A8K2U0   A2ML1
5 Q9NPC4  A4GALT                            Golgi                        Golgi
6 Q9UNA3   A4GNT                            Golgi                        Golgi
  HPA_Single_supportive HPA_Single_uncertain HPA_main_supportive
1
2                            Nucleus                                Nucleus
3
4
5
6
  HPA_main_uncertain ChristoforouUniprot_G.Ooverlap_CMN
1
2
3
4
5                                               membrane
6                                               membrane
```

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)]
```

2.2 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`.

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 precision with respect to the most abundant component

- name_pr.pdf - precision-recall curves of the cluster assignment with respect to all used annotations

- name_pr_abs.pdf - number of assigned proteins versus the precision of clusters with respect given localization

- 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] "/Library/Frameworks/R.framework/Versions/3.2/Resources/library/MetaMass/extdata/Bileck.txt"
```

5 Walk-through

In this section we give detailed description how create and analyze the MS data files. This section is intended for users with minimal experience with R. If the reader wants to avoid downloading a creating this files he can find the same examples in the section Examples where the same internally stored datasets are used. See `?analyze.MSfile` for detailed information.

Preparation of datasets and marker set for testing of the tool.

1. Copy contents of supplementary table 1.13 (Data Fig2a) into new spreadsheet. Delete columns B:L (Christoforou data) . Save as tab-delimited text in folder named e.g. Test, name the file Data_Fig.2a 2. Copy contents of supplementary table 1.14 (study4_9_10) into new spreadsheet. Save as tab-delimited text in folder named e.g. Test, name the file study4_9_10.txt 3. Copy contents in supplementary Table 1.11, (Annotation Table R-version) . Copy data into new spreadsheet. Save as tab-delimited text in folder named e.g. Test, name the file MyMarkers.txt

5.1 Analysis option 1

see Example 2 Generate the heatmap and the classification output files corresponding to Fig. 2a in the article: (typed commands are in script font, those below can be copied and pasted into R-studio, complete commands by pressing the enter key.)

```
> library(MetaMass)
> analyze.MSfile(MSfile = "Data_Fig2a.txt",Metadata = "Christoforou",overlap = 2, clusters = 1400, Annotation = "MyMarkers.txt",
+ markers =c(3,4,7), output = "Fig2a")
```

Tip. *Typing in R-Studio is greatly simplified using the tab key to auto-complete the command.. Thus, the command above can be typed as follows: ana [tab], msf [tab], "dat [tab], meta [tab], "Christoforou"), out [tab], "Fig2a")*

Explanation: analyze.MSfile: name of function used to perform all the analyses. MSfile: dataset to be analyzed, Metadata (see `?Metadata`): The "Christoforou" dataset is one of several stored in MetaMass as reference data. A complete list is found in supplementary Table 1.10. The meta-data is not clustered or used for the classification, but simply aligned to the processed test dataset as reference. Overlap=2 . The analysis is restricted to proteins that are identified both in the dataset and the metadata (Edit: the behavior has changed in current version. The `overlap` parameter affects now only data used for clustering i.e. the presence of protein in Metadata is not checked unless `cluster.metdata=TRUE`. For desired behavior set `overlap=NULL` which is default - see Example 2 Clusters: number of clusters to generate in K-means clustering. (an average cluster size of 5 is recommended which is default) Annotation: Text file with marker lists, number corresponds to column number in the file. Markers: c(3,4,7) use marker sets in columns 3,4, and 7 in the annotation list. output: file-prefix for the output files.

Result: The output files are found in the working directory. All output files have the prefix "Fig2a". Fig2a.cdt file is a heatmap file (JavaTreeView), The algorithm generates a single heatmap corresponding to the first of the marker sets, in this case marker set 3. (Uniprot/GO overlap + study 1). Fig.2a_table.txt is the classification result table (open with e.g Excel). The table contains protein identifiers, annotations from HPA, Uniprot and GO, the location assigned by MetaMass and the precision. The table also contains the MS signal values. Fig2a_pr-pdf contains precision-recall curves with relative numbers on the y-axis(e.g. Acrobat Reader). Fig2a_pr_abs.pdf contain precision recall curves with absolute numbers on the y-axis. These numbers correspond to the number of mapped proteins.

5.2 Analysis option 2

See Example 4 Compare results obtained when datasets 4, 9 and 10 are analyzed, and study 1 is used as metadata, and vice versa. (supplementary Fig. 14). Analyze dataset1, use dataset 4,9 and10, as Metadata.

```
> analyze.MSfile(MSfile = "study1.txt", clusters=432, Metadata = c("Carvalho", "Bileck", "Thakar"), Annotation = "MyMarkers.txt",
+ markers=c(3,4,6,7,9), output = "study1_MyMark")
```

Analyze dataset 4,9 and 10, use dataset1 as Metadata

```
> analyze.MSfile(MSfile = "study4_9_10.txt",clusters = 473,Annotation = "MyMarkers.txt", markers=c(3,4,6,7,9),output = "study4_9_10_MyMark")
```

The heatmaps are shown in supplementary Fig. 14. The pie chart in the figure shows the frequency of proteins that were mapped to the same location, different locations, not mapped, and those that were mapped using one of the datasets only. Even though studies 4 9 and 11 used low resolution methods, 80% of proteins that were mapped to a location in both analyses are mapped to the same locations as in study 1 when the data are combined.

5.3 Analysis option 3

See Example 7 Compare heatmaps obtained when the data in Fig. 2a are analyzed using a wide range of marker sets from Uniprot, GO and the Human Protein Atlas (supplementary Fig. 16)

```
> analyze.MSfile(MSfile = "Data_Fig2a.txt", Annotation = "MyMarkers.txt", markers=9, clusters=1400, output = "Fig2aSVM")
> analyze.MSfile(MSfile = "Data_Fig2a.txt", Annotation = "MyMarkers.txt", markers=4, clusters=1400, output = "Fig2aUGOovl")
> analyze.MSfile(MSfile = "Data_Fig2a.txt", Annotation = "MyMarkers.txt", markers=3, clusters=1400, output = "Fig2aStudy1UGOovl")
> analyze.MSfile(MSfile = "Data_Fig2a.txt", Annotation = "MyMarkers.txt", markers=6, clusters=1400, output = "Fig2aUniGOsum")
> analyze.MSfile(MSfile = "Data_Fig2a.txt", Annotation = "MyMarkers.txt", markers=7, clusters=1400, output = "Fig2aHPAsupportive")
> analyze.MSfile(MSfile = "Data_Fig2a.txt", Annotation = "MyMarkers.txt", markers=8, clusters=1400, output = "Fig2aHPAuncertain")
```

5.4 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 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	0	0	0	1	0	0	0	0
2	2	0	0	0	0	0	0	0	0	0
3	3	0	0	0	0	0	0	0	0	0
4	4	0	0	1	0	3	2	0	1	0
5	5	0	0	0	0	2	2	0	0	0
6	6	0	0	0	0	0	1	0	0	0

	ENDOSOME	EXTRACELLULAR	RIBOSOME	Nb_of_annotations	precision_main_component
1	0	0	1	2	0.5000000
2	0	0	0	0	NA
3	0	0	0	0	NA
4	0	0	0	7	0.4285714
5	0	0	0	4	0.5000000
6	0	0	0	1	1.0000000

	main_component	GOLGI_ratio	CYTOSOL_ratio	ER_ratio	MITOCHONDRION_ratio
1	NUCLEUS	0	0	0.0000000	0
2	<NA>	NaN	NaN	NaN	NaN
3	<NA>	NaN	NaN	NaN	NaN
4	NUCLEUS	0	0	0.1428571	0
5	NUCLEUS	0	0	0.0000000	0
6	PM	0	0	0.0000000	0

	NUCLEUS_ratio	PM_ratio	LYSOSOME_ratio	PEROXISOME_ratio	CS_ratio
1	0.5000000	0.0000000	0	0.0000000	0
2	NaN	NaN	NaN	NaN	NaN
3	NaN	NaN	NaN	NaN	NaN
4	0.4285714	0.2857143	0	0.1428571	0
5	0.5000000	0.5000000	0	0.0000000	0
6	0.0000000	1.0000000	0	0.0000000	0

	ENDOSOME_ratio	EXTRACELLULAR_ratio	RIBOSOME_ratio	assigned_location
1	0	0	0.5	NUCLEUS
2	NaN	NaN	NaN	<NA>
3	NaN	NaN	NaN	<NA>
4	0	0	0.0	NUCLEUS
5	0	0	0.0	NUCLEUS
6	0	0	0.0	PM

	Nb_main_component	Nb_assigned_location	precision_assigned_location
1	1	1	0.5000000
2	NA	NA	NA
3	NA	NA	NA
4	3	3	0.4285714
5	2	2	0.5000000
6	1	1	1.0000000

	updated_order
1	652
2	663
3	664
4	582
5	622
6	249

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 2a in the paper without the first (reference;metadata) columns.

7.1 Example 1

```
> 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 'res1'. 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`).

7.2 Example 2

As the second example let us reconstruct the figure 2a 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_Fig2a.txt",package="MetaMass")
> MyMarkers<-system.file("extdata","MyMarkers.txt",package="MetaMass")
> ##cluster with respect MSfile only (cluster.metadata=FALSE by default)
> library(MetaMass)
> analyze.MSfile(MSfile = file2,Metadata = "Christoforou", clusters = 1400, Annotation = MyMarkers,
+ markers =c(3,4,7), output = "Fig2a")
```

The results are stored in the working directory in files starting with 'Fig2a'.

7.3 Example 3

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")
```

See figure **??**

7.4 Example 4

Analyze dataset1, use dataset 4,9 and 10, as Metadata.

```
> file1<-system.file("extdata","Christoforou.txt",package="MetaMass")
> MyMarkers<-system.file("extdata","MyMarkers.txt",package="MetaMass")
> analyze.MSfile(MSfile = file1, clusters=432, Metadata = c("Carvalho", "Bileck", "Thakar"), Annotation = MyMarkers,
+ markers=c(3,4,6,7,9), output = "study1_MyMark")
```

Analyze dataset 4,9 and 10, use dataset1 as Metadata

```
> files1<-system.file("extdata",c("Bileck.txt","Thakar.txt","Carvalho.txt"),package="MetaMass")
> analyze.MSfile(MSfile = files1,clusters = 473,Annotation = MyMarkers, markers=c(3,4,6,7,9),output = "study4_9_10_MyMark")
```

7.5 Example 5

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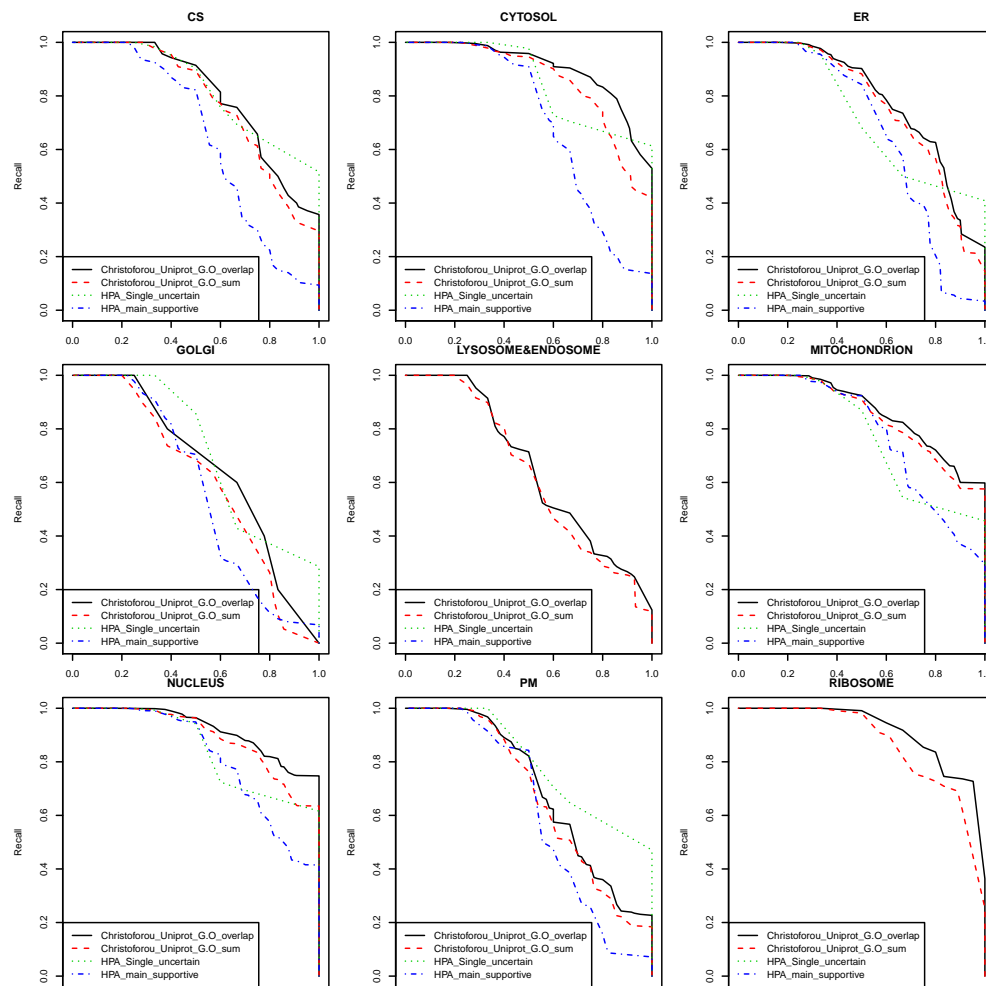

Figure 1: Precision-recall curves

7.6 Example 6

By default only proteins which were detected in all studies (groups) and Metadata are used for clustering. This could be bypassed by setting the parameter `overlap=`k - in this case missing values are replaced by 1s and only the proteins which were detected in at least k studies are used for clustering. The other proteins would remain in the table and heatmap as unassigned.

```
> files2<-system.file("extdata",c("Bileck.txt","Thakar.txt","Carvalho.txt","Andreyev.txt","Rodriguez.txt"),package="MetaMass")
> res3<-analyze.MSfile(MSfile=files2,Metadata=c("Christoforou"),output="res3intersect")
```

Here only 1280 proteins present in all studies are exported.

```
> files2<-system.file("extdata",c("Bileck.txt","Thakar.txt","Carvalho.txt","Andreyev.txt","Rodriguez.txt"),package="MetaMass")
> res3<-analyze.MSfile(MSfile=files2,Metadata=c("Christoforou"),output="res3_3",overlap=3)
```

Here 8548 proteins present in at least 3 studies are analyzed.

```
> files2<-system.file("extdata",c("Bileck.txt","Thakar.txt","Carvalho.txt","Andreyev.txt","Rodriguez.txt"),package="MetaMass")
> res3<-analyze.MSfile(MSfile=files2,Metadata=c("Christoforou"),output="res3_5",overlap=5,markers=c(3:7))
```

An finally 3315 proteins detected in at least 5 studies. In this case of course we leave large proportion of proteins unclassified see figure 2.

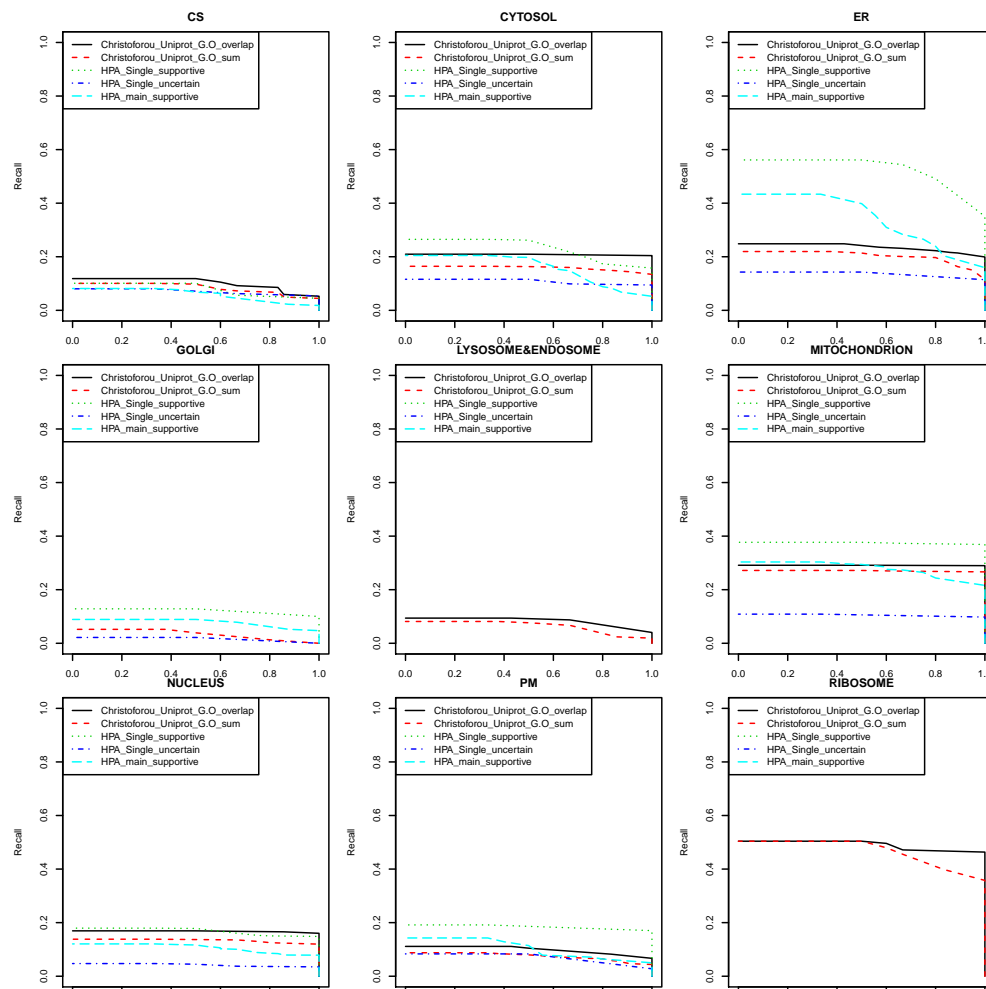Figure 2: Precision-recall curves

7.7 Example 7

```
> file2<-system.file("extdata","Data_Fig2a.txt",package="MetaMass")
> MyMarkers<-system.file("extdata","MyMarkers.txt",package="MetaMass")
> analyze.MSfile(MSfile = file2, Annotation = MyMarkers, markers=9, clusters=1400, output = "Fig2aSVM")
> analyze.MSfile(MSfile = file2, Annotation = MyMarkers, markers=4, clusters=1400, output = "Fig2aUGOovl")
> analyze.MSfile(MSfile = file2, Annotation = MyMarkers, markers=3, clusters=1400, output = "Fig2aStudy1UGOovl")
> analyze.MSfile(MSfile = file2, Annotation = MyMarkers, markers=6, clusters=1400, output = "Fig2aUniGOsum")
> analyze.MSfile(MSfile = file2, Annotation = MyMarkers, markers=7, clusters=1400, output = "Fig2aHPAsupportive")
> analyze.MSfile(MSfile =file2, Annotation = MyMarkers, markers=8, clusters=1400, output = "Fig2aHPAuncertain")
```

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.
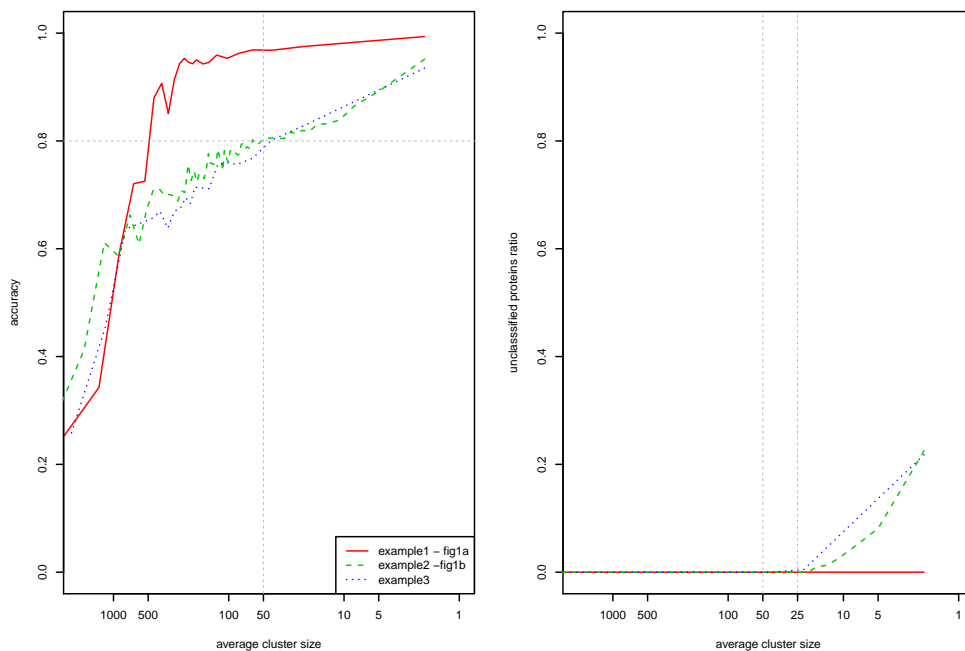

Figure 3: Performance against average cluster size