

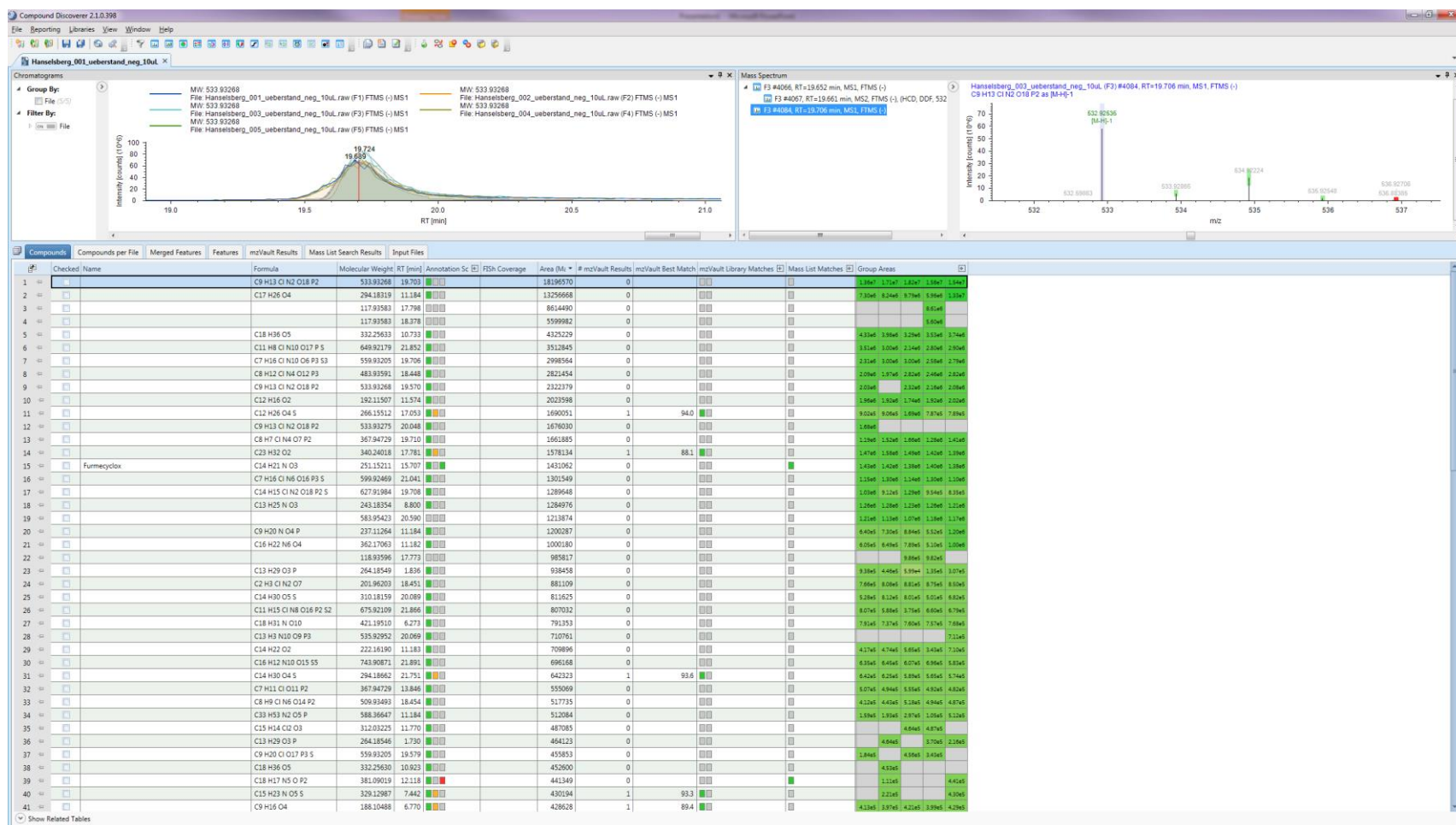
Tutorial to use R with the export data of Compound Discoverer to obtain input files for the FOR-IDENT platform

Uwe Kunkel

2018/01/10

1. Step: export data from CD

- Go to the results page of your CD study



1. Step: export data from CD

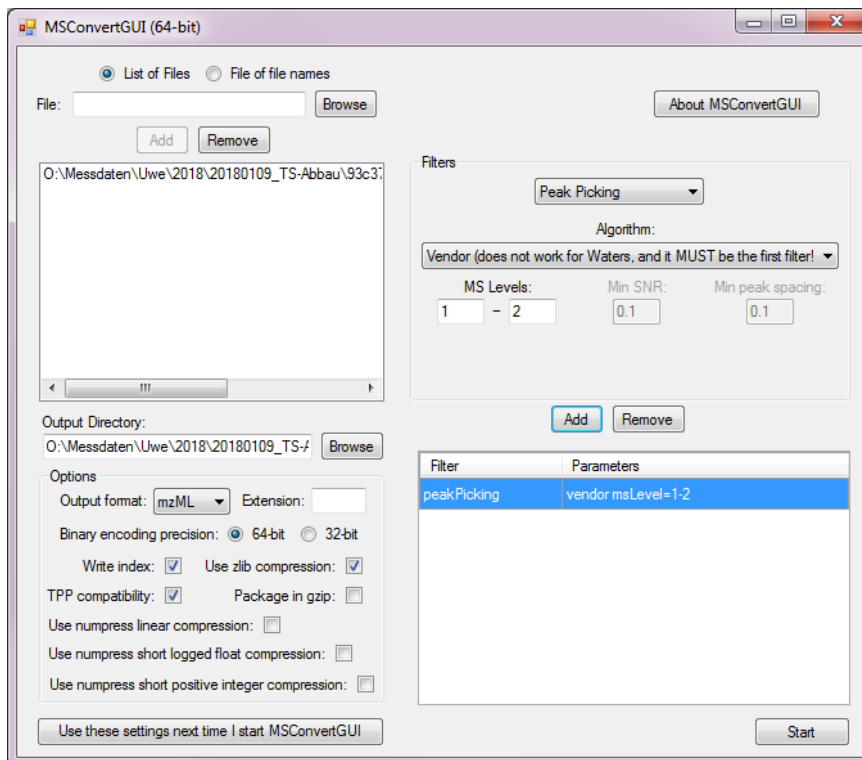
- Export the compounds table to a text-file (e.g. Compounds_CD_neg.csv)

The screenshot displays the Compound Discoverer 2.10.398 software interface. The top section shows three panels: Chromatograms, Mass Spectrum, and a third panel for peak data. The Chromatograms panel displays a chromatogram with peaks labeled with retention times (e.g., 19.724, 19.659). The Mass Spectrum panel shows a mass spectrum with peaks labeled with m/z values (e.g., 532.02536, 532.02536, 534.02224). The bottom panel is a table of compounds, with columns for Checked, Name, Formula, Molecular Weight, RT [min], Annotation Sc, Fish Coverage, Area (M), # m/zVault Results, m/zVault Best Match, m/zVault Library Matches, Mass List Matches, and Group Areas. The table lists various compounds, including C9 H13 O N2 O18 P2, C17 H26 O4, C18 H36 O5, C11 H8 O N10 O17 P 5, C7 H16 O N10 O6 P3 S3, C8 H12 O N4 O12 P3, C9 H13 O N2 O18 P2, C12 H16 O2, C12 H26 O4 S, C9 H13 O N2 O18 P2, C8 H7 O N4 O7 P2, C23 H32 O2, C14 H21 N O3, C7 H16 O N10 O6 P3 S, C14 H15 C N2 O18 P2 S, C13 H25 N O3, C9 H20 N O4 P, C16 H22 N6 O4, C13 H29 O3 P, C2 H3 C1 N2 O7, C14 H30 O5 S, C11 H15 C1 N8 O16 P2 S2, and C18 H31 N O10. The table also includes a 'Furmycyclo' label for some compounds. A context menu is open over the table, showing options like 'Copy With Headers', 'Copy', 'Clear Selection', 'Enable Column Fixing', 'Cell Selection Mode', 'Collapse All Column Headers', 'Expand All Column Headers', 'Check Selected', 'Check All', 'Uncheck Selected', 'Uncheck All', 'Edit Compound Annotation', 'Clear Compound Annotation', 'Apply FISH Scoring', and 'Export'. The 'Export' option is highlighted, and a sub-menu is visible with options: 'Export to Text File...', 'Export to Excel...', 'Export to Xcalibur Inclusion/Exclusion List...', and 'Export to TraceFinder...'.

be sure that the columns Formula, Molecular weight and RT are columns number 3 to 5 in the compounds table, otherwise you have to adjust the R-script

2. Step: Convert raw-file to mzML-formate

- This can e.g. be done with Proteowizard (<http://proteowizard.sourceforge.net/downloads.shtml>)



Activate the pick picking algorithm for MS levels 1 and 2 during the file conversion

This will convert your data from “profile” to “centroid”

This centroidization is not absolutely necessary but recommend.

3. Step: Install R (and R-studio)

- Install the latest version of R
 - <https://cran.r-project.org/bin/windows/base/>
- You may optionally also install R-Studio, by this you get a nicer GUI for R
 - <https://www.rstudio.com/products/rstudio/download/>



4. Preparing R for the work

- At first you have to install an extra package of R that can handle the mzML-data
- Thereto paste in the R console

```
source("https://bioconductor.org/biocLite.R")  
biocLite("mzR")
```



This only works if you have an internet connection, otherwise there are also options to install the mzR package if you do not have internet access on the

- You may optionally also install R-Studio, by this you get a nicer GUI for R and also directly an editor which you need to write and adapt R-code
 - In the following I used R-Studio and there workflow is described using R-Studio

4. Preparing R for the work

- Open the respective R-script (20180110_CD R-Addin_neg. mode.R or 20180110_CD R-Addin_pos. mode.R) for pos or neg. mode

The screenshot displays the RStudio environment with three main panes:

- R-console to edit the script:** The top-left pane shows the R script being edited. The code includes comments and R commands for loading data, defining variables, and performing calculations. The console at the bottom shows the execution of the script.
- Overview over your data:** The top-right pane, titled 'Environment', lists the objects in the global environment. It shows variables like 'ms_data', 'ms_scan', 'ms2', 'ms2_peak_max', and 'ms2_suspect' with their respective dimensions. Below this, the 'Values' pane displays the structure of the 'data.ms' object, showing columns like 'file_cd', 'file_ms', 'formula_corr', 'mass_cd', 'mass_electron', 'mass_proton', 'mass_tol', 'mass_tol_ppm', 'masses', 'ms2_data', 'polarity', 'RT_cd', 'RT_tol', and 'RTs'.
- Helpfiles and plots:** The bottom-right pane shows the 'R Documentation' for the 'Pattern Matching and Replacement' package. It provides a detailed description of the functions 'grep', 'grepl', 'gregexpr', 'gregexec', 'sub', and 'gsub', along with their usage and arguments.

R-console to paste and run the script

```
1 source("https://bioconductor.org/biocLite.R")
2 biocLite("mzr")
3
4 # test to load mzML data into R
5 library(mzr)
6
7 # documentation
8 # browse vignettes("mzr")
9
10 setwd("E:/CD addin") ### Folder where your data is in
11
12 ##### Tested and written for CD v.2.1
13
14 ##### Definitions of files to work with
15
16 file_cd <- "compounds_cd_neg.csv" # your Compound-Discoverer output-file
17 file_ms <- "test_neg.mzml" # your MS files after conversion to mzML
18
19 ##### Definitions of mass, retention time tolerances and polarity
20
21 mass_tol_ppm <- 5
22 rt_tol_sec <- 10
23 polarity <- "neg" ##### please chose polarity, accepted values: "pos" or "neg"
24 mass_proton <- 1.007276
25
26
27 ##### Loading and manipulation of files
28
29 ##### Compound discoverer file
30
31
32 cd_output <- read.csv(file_cd, sep = "\t", dec = ".", stringsAsFactors = F, na.strings = "")
33
34 ## head(cd_output) ### control, if import worked successfully
35
36 cd_output <- cd_output[,1:5] ## reduction to necessary columns of cd output file
37
38 cd_output[,2] <- cd_output[,2] - mass_proton ## subtracts the mass of one proton to the mass in the cd_output_file since this was automatically added for the output file to obtain t
39
40 colnames(cd_output) <- c("sum_formula", "Mz", "RT") ## set column names to sensible names
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

Environment

Object	Class	Attributes
ms_data	data.frame	5628 obs. of 5 variables
ms_scan	data.frame	5628 obs. of 21 variables
ms2	data.frame	3742 obs. of 5 variables
ms2_peak_max	data.frame	1 obs. of 5 variables
ms2_suspect	data.frame	1 obs. of 5 variables

Values

Variable	Value
data.ms	Formal class 'mzramp'
file_cd	"compounds_cd_neg.csv"
file_ms	"test_neg.mzml"
formula_corr	"C16H12N10O15S"
formula_un_corr	"C16H12N10O15S"
mass_cd	1.007276
mass_electron	0.000549
mass_proton	1.007276
mass_tol	0.00371450717
mass_tol_ppm	5
masses	rum [1:108] 263 263 233
ms2_data	rum [1:40] 53.7 5378 54.
polarity	"neg"
RT_cd	21.891
RT_tol	0.166666666666667
rt_tol_sec	10
RTs	rum [1:108] 1.73 1.84 1.84 1.85 1.85 ...

Pattern Matching and Replacement

grep, grepl, gregexpr, grepgrep and gregexec search for matches to argument pattern within each element of a character vector; they differ in the format of and amount of detail in the results.

sub and gsub perform replacement of the first and all matches respectively.

Usage

```
grep(pattern, x, ignore.case = FALSE, perl = FALSE, value = FALSE,
      fixed = FALSE, useBytes = FALSE, invert = FALSE)
grepl(pattern, x, ignore.case = FALSE, perl = FALSE,
       fixed = FALSE, useBytes = FALSE)
gregexpr(pattern, text, ignore.case = FALSE, perl = FALSE,
          fixed = FALSE, useBytes = FALSE)
sub(pattern, replacement, x, ignore.case = FALSE, perl = FALSE,
     fixed = FALSE, useBytes = FALSE)
gsub(pattern, replacement, x, ignore.case = FALSE, perl = FALSE,
     fixed = FALSE, useBytes = FALSE)
regexpr(pattern, text, ignore.case = FALSE, perl = FALSE,
         fixed = FALSE, useBytes = FALSE)
gregexpr(pattern, text, ignore.case = FALSE, perl = FALSE,
          fixed = FALSE, useBytes = FALSE)
regexec(pattern, text, ignore.case = FALSE, perl = FALSE,
         fixed = FALSE, useBytes = FALSE)
```

Arguments

pattern character string containing a **regular expression** (or character string for fixed = TRUE) to be matched in the given character vector. Coerced by **as.character** to a character string if possible. If a character vector of length 2 or more is supplied, the first element is used with a warning. Missing values are allowed except for regexpr and gregexpr.

x, text a character vector where matches are sought, or an object which can be coerced by **as.character** to a character vector. **Long vectors** are supported.

ignore.case if FALSE, the pattern matching is case sensitive and if TRUE, case is ignored during matching.

perl logical. Should Perl-compatible regexps be used?

5. Adjust the R-script to your data

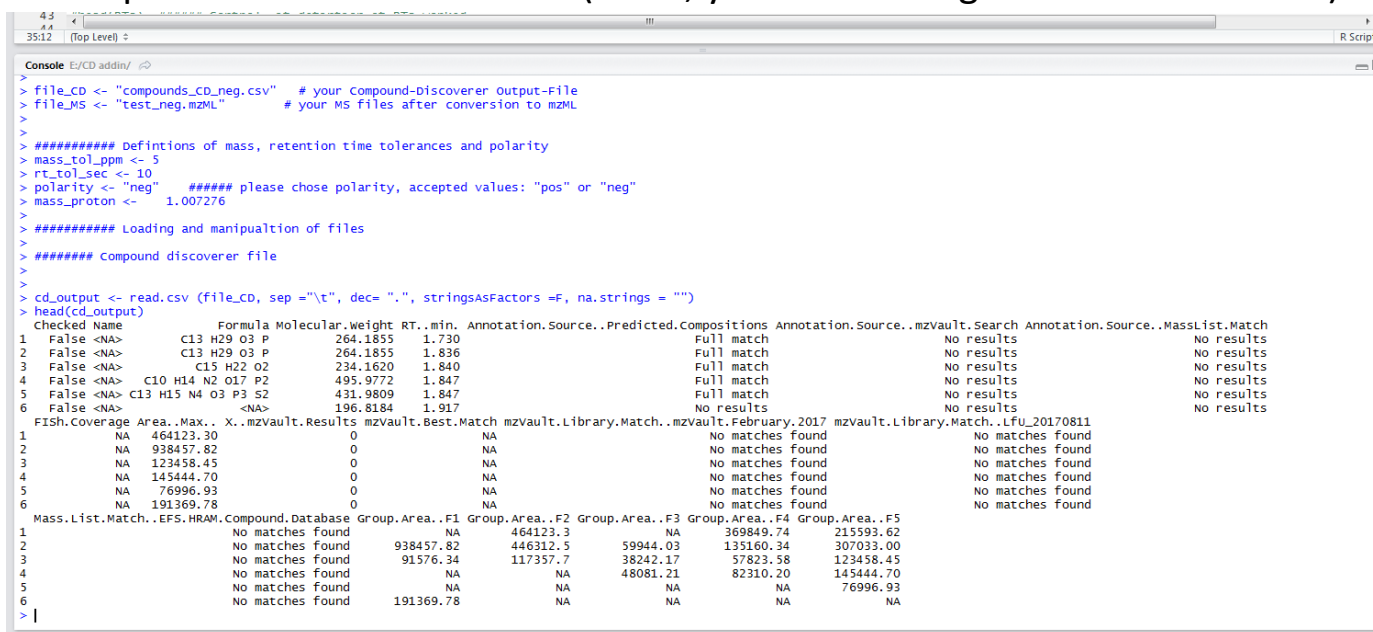
- Change the necessary things in the R-code to import and run your data
 - The path where your data is located
In script for neg mode: line 10: `setwd("E:/CD addin")`
 - The name of your Compound discover output file
In script for neg mode: line 19: `file_MS <- "test_neg.mzML"`
 - The name of the mzML-file which you want to process and look for MS/MS data
In script for neg mode: line 18: `file_CD <- "compounds_CD_neg.csv"`
The name of the mzML-file which you want to process and look for MS/MS data
 - The info on the polarity of your data
In script for neg. mode: line 25: `polarity <- "neg"`
 - The RT (in seconds) and mass tolerances (in ppm) which the script uses for finding MS2-information based on the information in the CD export file. I think 5 ppm and 10 s are a good value
In script for neg. mode: line 23: `mass_tol_ppm <- 5`
In script for neg. mode: line 24: `rt_tol_sec <- 10`

5. Adjust the R-script to your data(2)

- Change the necessary things in the R-code to import and run your data
 - When importing the CD export file into R, you have to specify the format of the csv-file, i.e. what is used to separate the columns and what is used for decimal points

If you use English language settings, you normally have a “,” as column separator and “.” as decimal point

In script for neg. mode: line 33: `cd_output <- read.csv (file_CD, sep=".", dec=".", stringsAsFactors =F, na.strings = "")`
 - To check the correct importing of the data from the CD output file you can use line 34: `head(cd_output)`
 - The output should look as follows (if not, you must change the code in line 33)



```
> file_CD <- "compounds_CD_neg.csv" # your Compound-Discoverer output-File
> file_MS <- "test_neg.mzML" # your MS files after conversion to mzML
>
> ##### Definitions of mass, retention time tolerances and polarity
> mass_tol_ppm <- 5
> rt_tol_sec <- 10
> polarity <- "neg" ##### please chose polarity, accepted values: "pos" or "neg"
> mass_proton <- 1.007276
>
> ##### Loading and manipulation of files
> ##### Compound discoverer file
>
> cd_output <- read.csv (file_CD, sep = "\t", dec= ".", stringsAsFactors =F, na.strings = "")
> head(cd_output)
  Checked Name      Formula Molecular.weight  RT..min. Annotation.Source..Predicted.Compositions Annotation.Source..mzVault.Search Annotation.Source..MassList.Match
1  False <NA>      C13 H29 O3 P      264.1855    1.730      NA      Full match      No results      No results
2  False <NA>      C13 H29 O3 P      264.1855    1.836      NA      Full match      No results      No results
3  False <NA>      C13 H22 O2      234.1620    1.840      NA      Full match      No results      No results
4  False <NA>      C10 H14 N2 O17 P2    495.9772    1.847      NA      Full match      No results      No results
5  False <NA>      C13 H15 N4 O3 P3 S2    431.9809    1.847      NA      Full match      No results      No results
6  False <NA>      <NA>      196.8184    1.917      NA      No results      No results      No results
  Fish.Coverage Area..Max.. X..mzVault.Results mzVault.Best.Match mzVault.Library.Match..mzVault.February.2017 mzVault.Library.Match..LfU_20170811
1      NA      464123.30      0      NA      No matches found      No matches found
2      NA      938457.82      0      NA      No matches found      No matches found
3      NA      123458.45      0      NA      No matches found      No matches found
4      NA      145444.70      0      NA      No matches found      No matches found
5      NA      76996.93      0      NA      No matches found      No matches found
6      NA      191369.78      0      NA      No matches found      No matches found
  Mass.List.Match..EFS.HRAM.Compound.Database Group.Area..F1 Group.Area..F2 Group.Area..F3 Group.Area..F4 Group.Area..F5
1      No matches found      NA      464123.3      NA      369849.74      215593.62
2      No matches found      938457.82      446312.5      59944.03      135160.34      307033.00
3      No matches found      91576.34      117357.7      38242.17      57823.58      123458.45
4      No matches found      NA      NA      48081.21      82310.20      145444.70
5      No matches found      NA      NA      NA      NA      76996.93
6      No matches found      191369.78      NA      NA      NA      NA
```

6. Work in R-Studio and run the script

- To work in R-Studio, just copy and paste parts of the script from the editor window to the console window. The script then runs automatically
- In the editor, you can make some comment to explain the code and data manipulation steps. This is done by using “#”. All that is written in a line in the editor after the “#” sign is not interpreted in the R-console.
- If all the adjustments worked, you can copy and paste all things from the editor window and you will automatically get a txt-file in the FOR-IDENT format. The file will be saved in the folder where all your data is located
- The file name will be automatically chosen. It will be:
“Name_of_the_mzML file”_”polarity”_”export_CD”.txt

So if you file is “test.mzml” and you measured in neg. mode, the file will get the name
“test_neg_pos_export_CD.txt”