Metabolomics data analysis with Bioconductor

Johannes Rainer (Eurac Research, Italy)[1]

June 12, 2017 @CSAMA2017

[1] email: johannes.rainer@eurac.edu, github/twitter: jotsetung

Talk content

- Focus on pre-processing of LCMS data.
- Focus on the `xcms` package (*new* user interface), but other exist too (e.g. `yamss`).

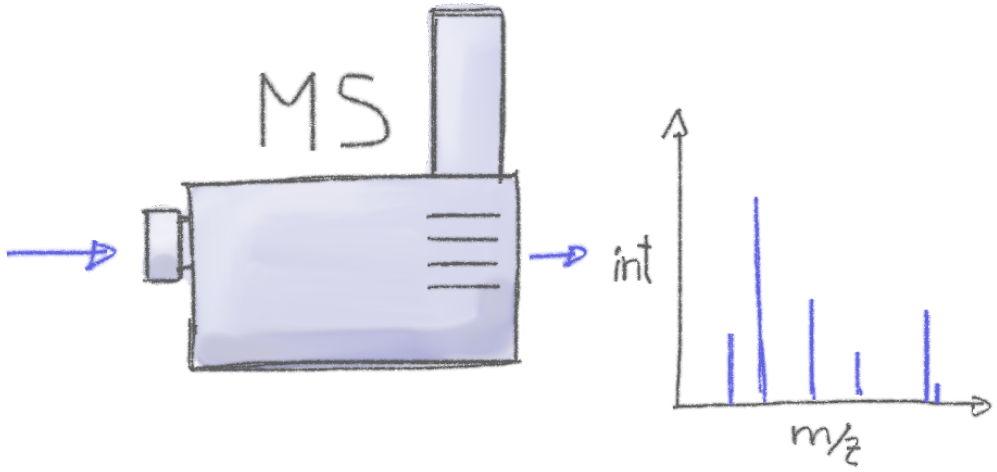
Metabolomics?

- Is the large-scale study of small molecules (metabolites) in a system (cell, tissue or organism).
- Metabolites are intermediates and products of cellular processes (metabolism).
- From Genome to Metabolome:
  - Genome: what can happen.
  - Transcriptome: what appears to be happening.
  - Proteome: what makes it happen.
  - Metabolome: what actually happened. Influenced by genetic and environmental factors.

How are we measuring that?

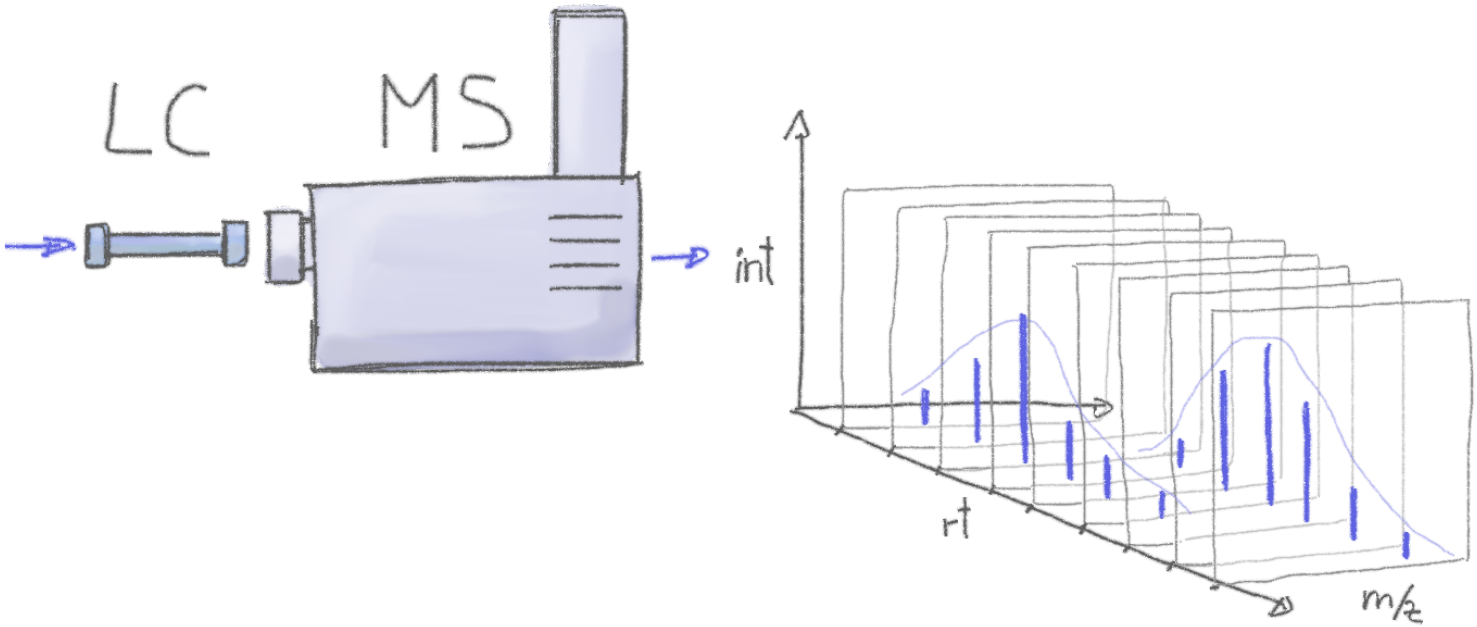
- Nuclear magnetic Resonance (NMR) - not covered here.
- Mass spec (MS)-based metabolomics

Mass Spectrometry (MS)



- Problem: unable to distinguish between metabolites with the same mass-to-charge ratio (m/z).

Liquid Chromatography Mass Spectrometry (LCMS)



- Combines physical separation via LC with MS for mass analysis.
- Additional time dimension to separate different ions with same m/z.
- Also used: Gas-chromatography (GC) instead of LC.
- Additional complication: targeted/untargeted metabolomics.

LCMS-based metabolomics data pre-processing

- Input: mzML or netCDF files with multiple MS spectra per sample.
- Output: matrix of abundances, rows being *features*, columns samples.
- feature: ion with a unique mass-to-charge ratio (m/z) and retention time.
- Example: load files from the `faahKO` data packages, process using `xcms`.

```
library(xcms)
library(faahKO)
library(RColorBrewer)

cdf_files <- dir(system.file("cdf", package = "faahKO"), recursive = TRUE,
                 full.names = TRUE)[c(1, 2, 7, 8)]
## Read the data
faahKO <- readMSData2(cdf_files)
```

- `OnDiskMSnExp`: small memory size, loads data on-demand.

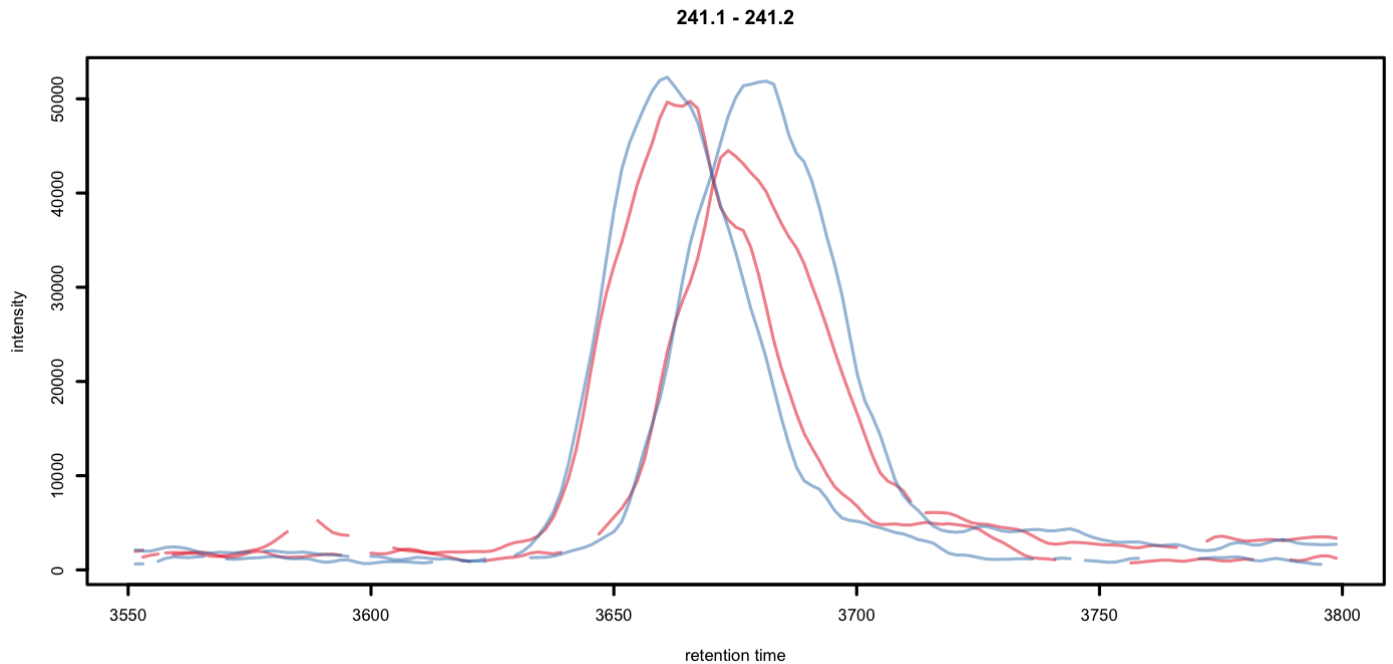
LCMS-based metabolomics data pre-processing

- Chromatographic peak detection.
- Sample alignment.
- Correspondence.

LCMS pre-processing: Peak detection

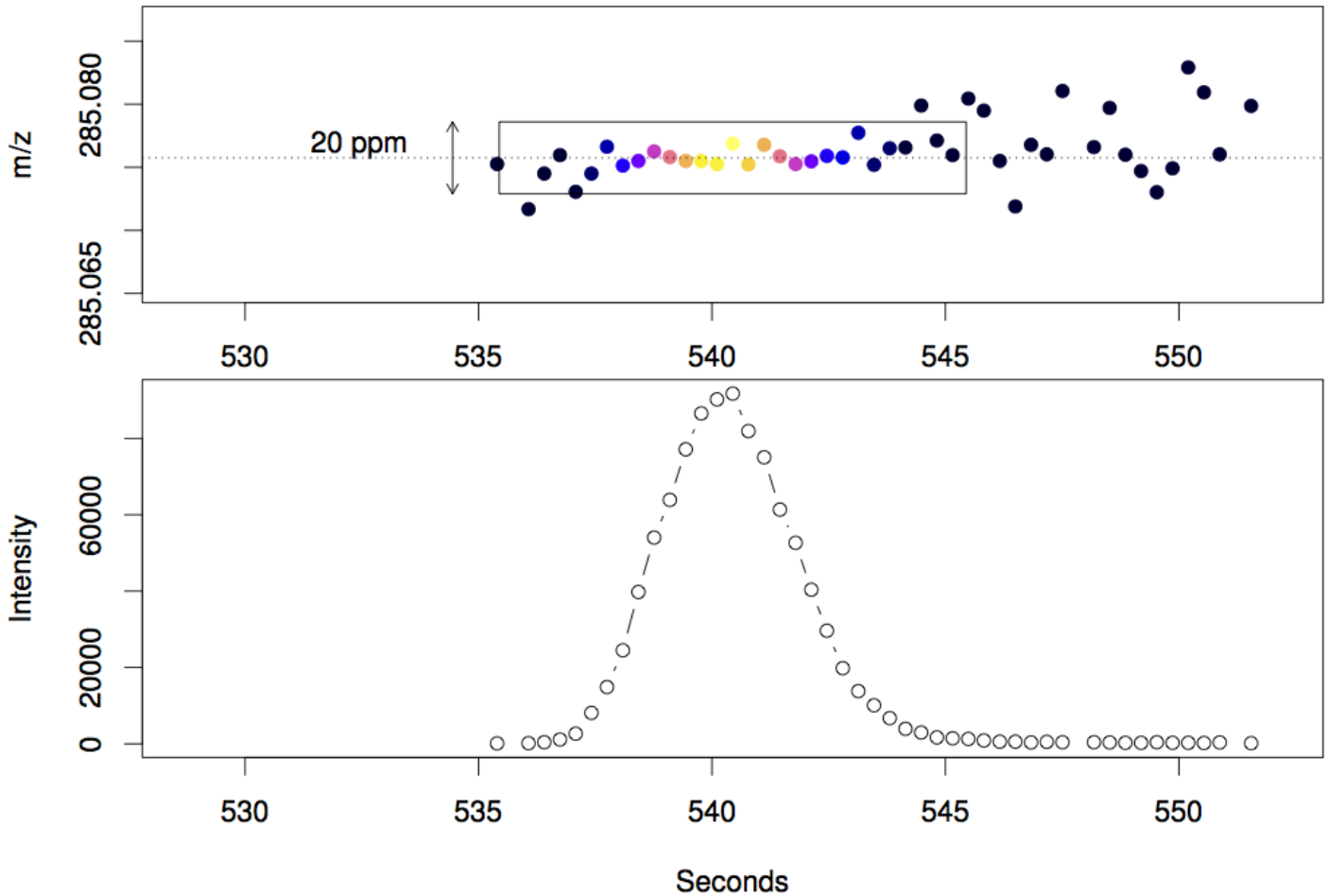
- Goal: Identify chromatographic peaks within slices along mz dimension.
- What type of peaks have to be detected?

```
mzr <- c(241.1, 241.2)
chrs <- extractChromatograms(faahKO, mz = mzr, rt = c(3550, 3800))

cols <- brewer.pal(3, "Set1")[c(1, 1, 2, 2)]
plotChromatogram(chrs, col = paste0(cols, 80))
```



LCMS pre-processing: Peak detection

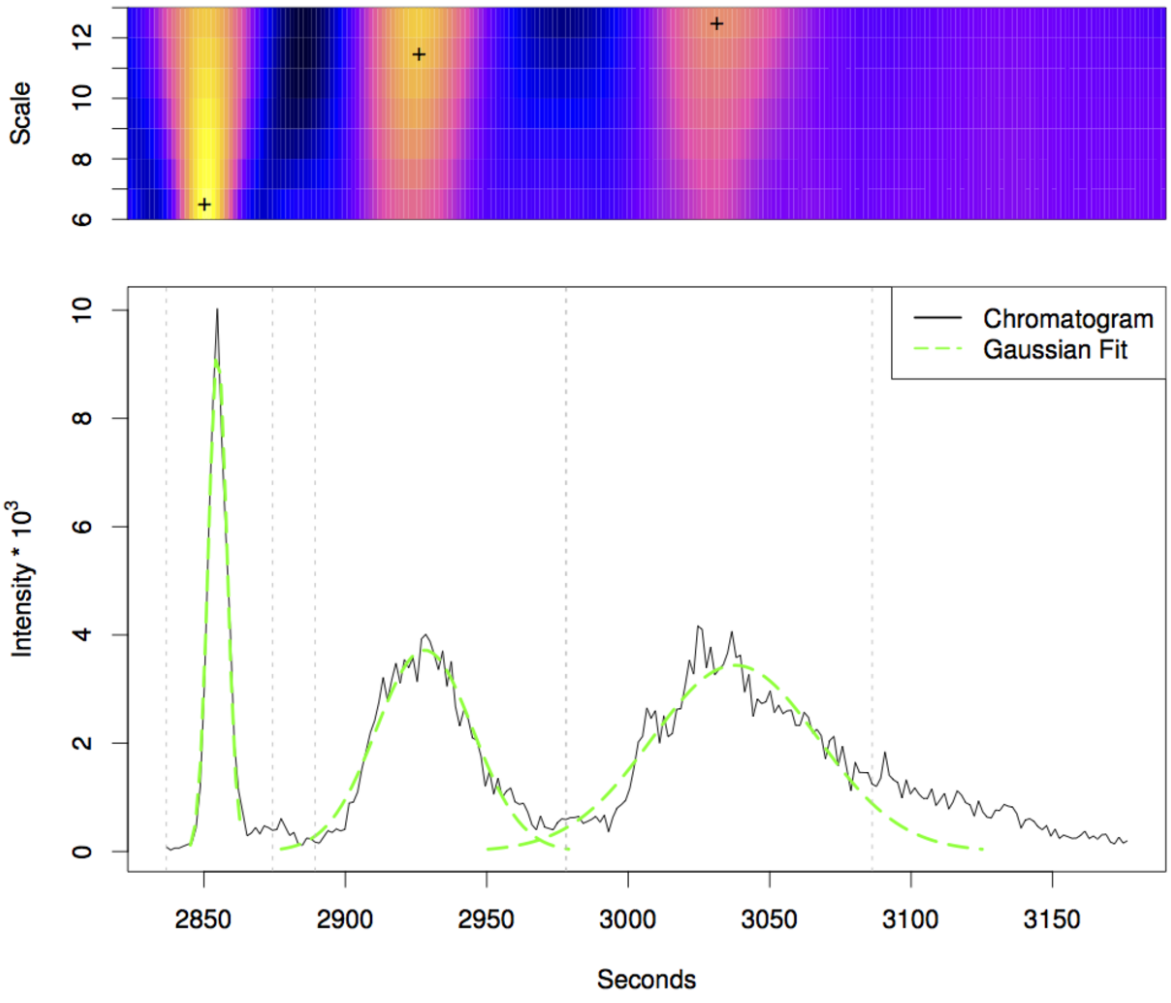
- centWave (Tautenhahn et al. *BMC Bioinformatics*, 2008):
- Step 1: Detection of regions of interest



- mz-rt regions with low mz-variance.

LCMS pre-processing: Peak detection

- Step 2: Peak detection using continuous wavelet transform (CWT)



- Equivalent to multiple Gaussian fits and choosing the best.

LCMS pre-processing: Peak detection

- Example: centWave-based peak detection:

```
faahKO <- findChromPeaks(faahKO, param = CentWaveParam())
```

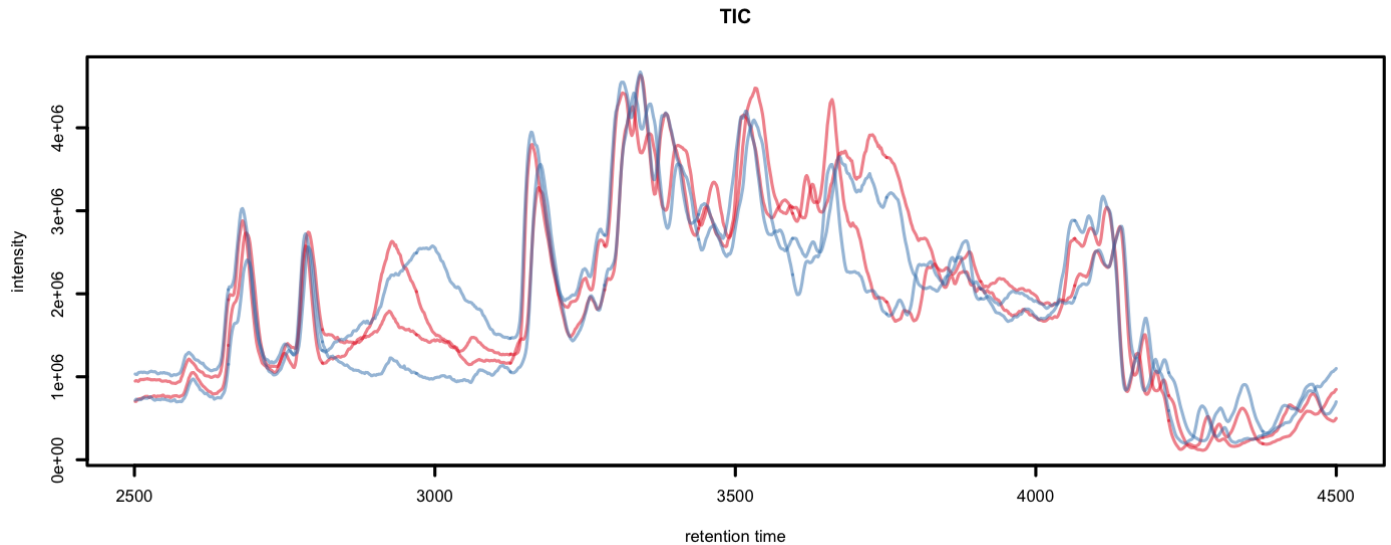
- Result: `XCMSnExp`, container for LC/GC-MS results, extends `OnDiskMSnExp`.

```
head(chromPeaks(faahKO))
```

```
mz mzmin mzmax       rt    rtmin    rtmax        into        intb  maxo
[1,] 425.9 425.9 425.9 2520.158 2510.768 2527.982    9999.769    9984.120   741
[2,] 464.3 464.3 464.3 2518.593 2504.508 2532.677   32103.270   32082.926  1993
[3,] 499.1 499.1 499.1 2524.852 2520.158 2527.982    4979.194    4904.709   883
[4,] 572.7 572.7 572.7 2524.852 2520.158 2527.982    2727.446    2721.187   559
[5,] 579.8 579.8 579.8 2524.852 2520.158 2527.982    2450.477    2444.218   468
[6,] 453.2 453.2 453.2 2506.073 2501.378 2527.982 1007408.973 1007380.804 38152
sn sample is_filled
[1,]   740      1         0
[2,]  1992      1         0
[3,]    13      1         0
[4,]   558      1         0
[5,]   467      1         0
[6,] 38151      1         0
```

LCMS pre-processing: Alignment

- Goal: Adjust retention time differences/shifts between samples.
- Total ion chromatogram (TIC) representing the sum of intensities across a spectrum.



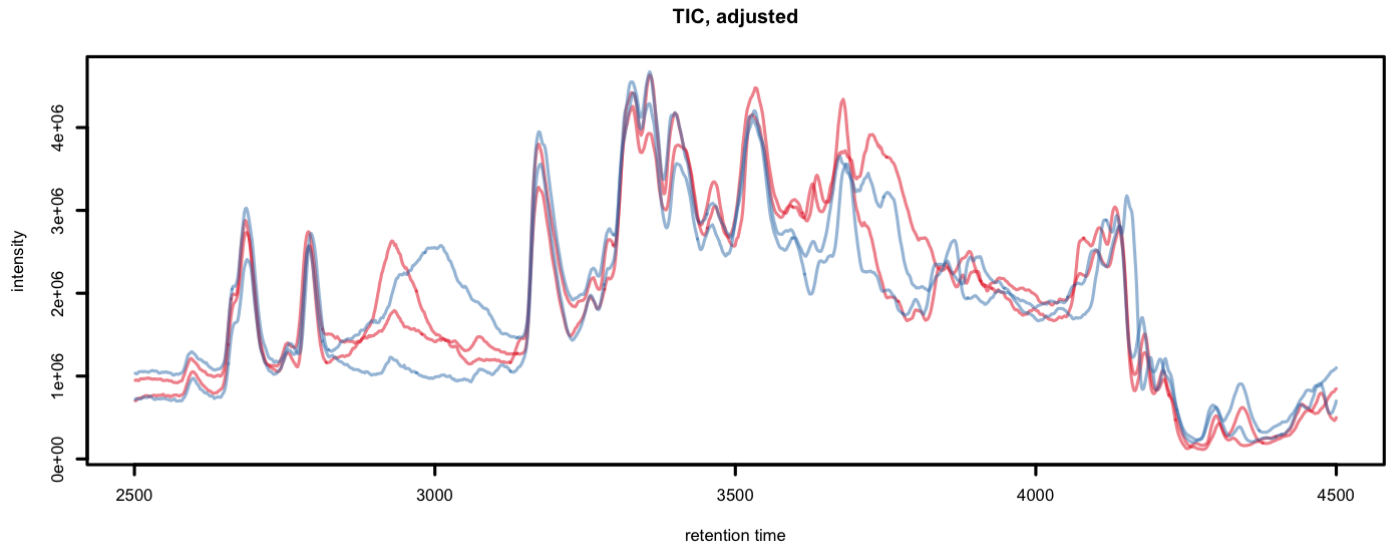
- Overview of algorithms: (Smith et al. *Brief Bioinformatics* 2013).
- `xcms`: *peak groups* (Smith et. al *Anal Chem* 2006), obiwarp (Prince et al. *Anal Chem*, 2006),

LCMS pre-processing: Alignment

- Example: use obiwarp to align samples.

```
faahKO <- adjustRtime(faahKO, param = ObiwarpParam())
```

- TIC after adjustment:

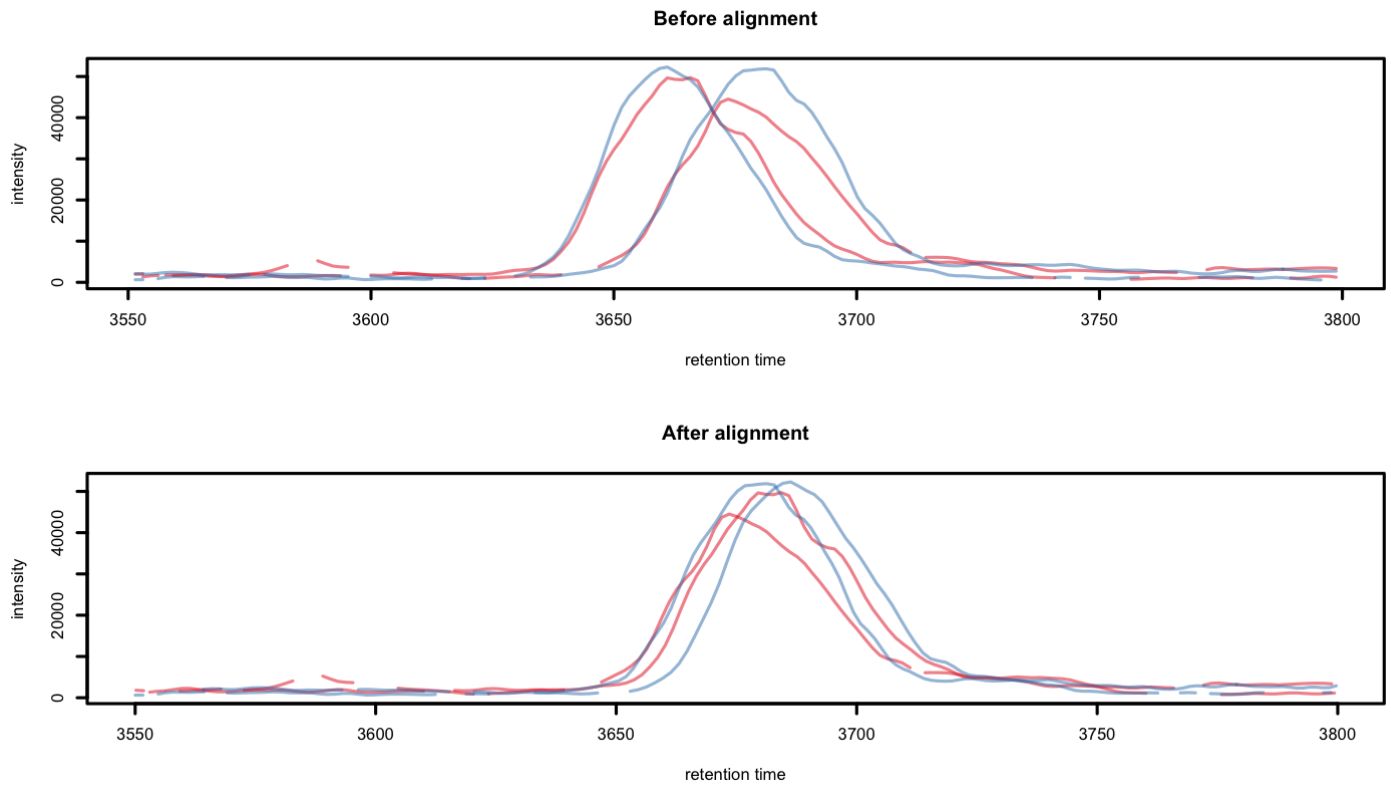


- Assumptions:
  - Samples relatively similar (either similar chromatograms or a set of common metabolites present in all).
  - Warping methods: analyte elution order is same in all samples.

LCMS pre-processing: Alignment

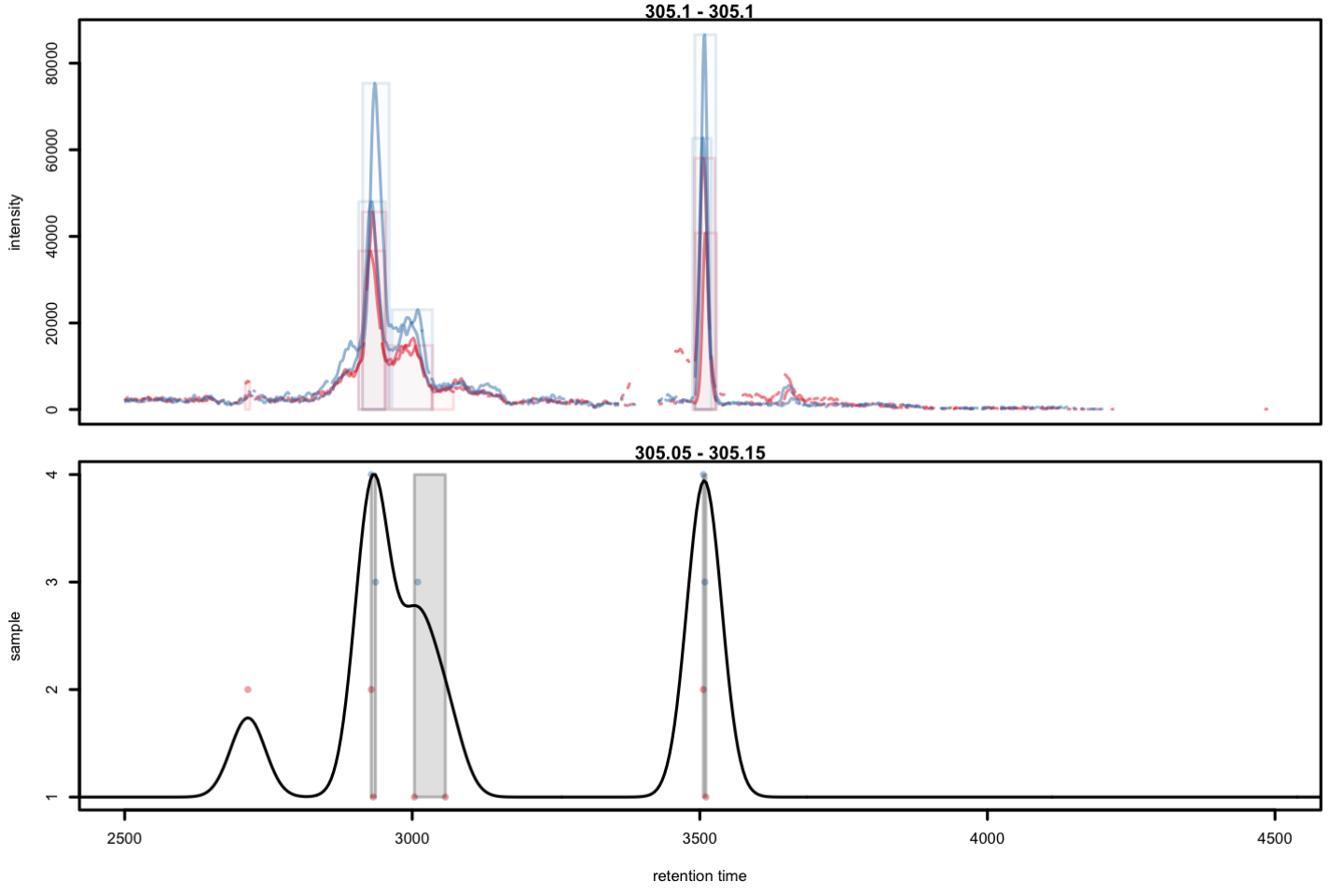
- Example: effect of alignment on example peak.

```
chrs_adj <- extractChromatograms(faahKO, mz = mzr, rt = c(3550, 3800))

par(mfrow = c(2, 1))
plotChromatogram(chrs, col = paste0(cols, 80), main = "Before alignment")
plotChromatogram(chrs_adj, col = paste0(cols, 80), main = "After alignment")
```



LCMS pre-processing: Correspondence

- Goal: Group detected chromatographic peaks across samples.
- `xcms`: *peak density* method:



- Peaks that are close in rt are grouped to a *feature*.

LCMS pre-processing: Correspondence

- Example: peak grouping.

```
faahKO <- groupChromPeaks(faahKO, param = PeakDensityParam())
```

- Extract results: `featureDefinitions`: extract the definition of features.

```
## Definitions of the features:
featureDefinitions(faahKO)
```

```
DataFrame with 1678 rows and 9 columns
   mzmed     mzmin     mzmax     rtmed     rtmin     rtmax    npeaks
       <numeric> <numeric> <numeric> <numeric> <numeric> <numeric> <numeric>
FT0001    200.20     200.2     200.2  3517.121  3496.681  3537.561         2
FT0002    200.25     200.2     200.3  3865.494  3838.392  3892.595         2
FT0003    201.25     201.2     201.3  4134.665  4112.232  4157.099         2
...          ...       ...       ...       ...       ...       ...       ...
FT1676    599.45     599.4     599.5  2989.643  2973.994  3005.293         2
FT1677    599.85     599.8     599.9  2583.538  2524.809  2611.715         4
FT1678    600.00     600.0     600.0  3456.187  3445.045  3467.329         2
      X1          peakidx
       <numeric>           <list>
FT0001         2        1089,5859
FT0002         2        1677,6249
FT0003         2        4557,6593
...          ...              ...
FT1676         2        2885,7528
FT1677         4 75,2316,4673,...
FT1678         2        1049,3394
```

LCMS pre-processing: Correspondence

- `featureValues`: extract *values* for each feature from each sample.

```
## Access feature intensities
head(featureValues(faahKO, value = "into"))
```

```
       ko15.CDF ko16.CDF wt15.CDF wt16.CDF
FT0001 6029.945       NA 4586.527       NA
FT0002 1144.015       NA 1018.815       NA
FT0003       NA  774.576 1275.475       NA
FT0004       NA       NA 1284.728   1220.7
FT0005 2759.095 3872.963       NA       NA
FT0006 7682.585 3806.080       NA       NA
```

LCMS pre-processing

- Final note: XCMSnExp object tracks all analysis steps.

```
## Extract the "processing history"
processHistory(faahKO)
```

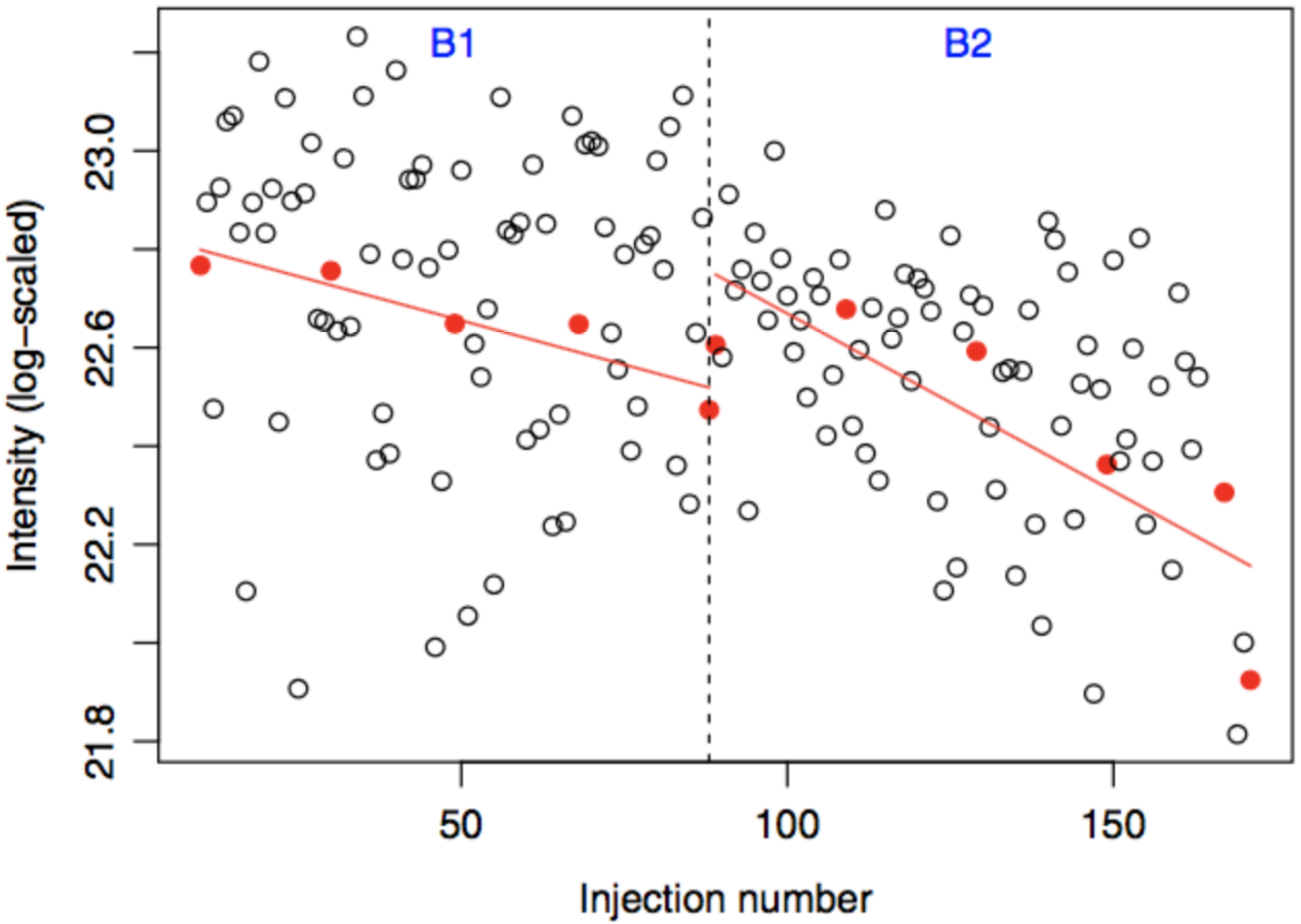
```
[[1]]
Object of class "XProcessHistory"
 type: Peak detection
 date: Thu Jun  8 13:16:56 2017
 info:
 fileIndex: 1,2,3,4
 Parameter class: CentWaveParam

[[2]]
Object of class "XProcessHistory"
 type: Retention time correction
 date: Thu Jun  8 13:17:13 2017
 info:
 fileIndex: 1,2,3,4
 Parameter class: ObiwarpParam

[[3]]
Object of class "XProcessHistory"
 type: Peak grouping
 date: Thu Jun  8 13:17:20 2017
 info:
 fileIndex: 1,2,3,4
 Parameter class: PeakDensityParam
```

What next? Missing values

- `xcms` provides the possibility to read data from raw files to fill-in missing peaks (`fillChromPeaks`).
- Data imputation. Be aware of introduced correlations.

What next? Data normalization

- Adjust within batch and between batch differences.
- Injection order dependent signal drift (Wehrens et al. *Metabolomics* 2016).



What next? Identification

- Annotate features to metabolites.
- Features are not chemical compounds.
- Features from the same compound are co-eluting and can be related (isotopes, adducts).
- Starting point: `CAMERA` package.
- On-line spectra databases (e.g. `MassBank`).

Finally. . .

- Hands on in the afternoon workshop.

thank you for your attention!