Analysis of MS Data in R

Investigating Mass Spectra and Neutral Loss Trigger

Sebastian Gibb

2017-11-27

Availability

https://github.com/sgibb/talk-odense-20171127/

github: @sgibb / twitter: @sebastiangibb

Loading Data with `MSnbase` (Gatto and Lilley 2011)

```
library("MSnbase")

msfile <- file.path(
    "data",
    "neutral-loss",
    "LUM2_01470_KS_L1-5-2_EC17-123-150In2_55oC_NLtrig.mzML.gz"
)

ms <- readMSData(msfile, mode="onDisk")
```

The `MSnExp` object

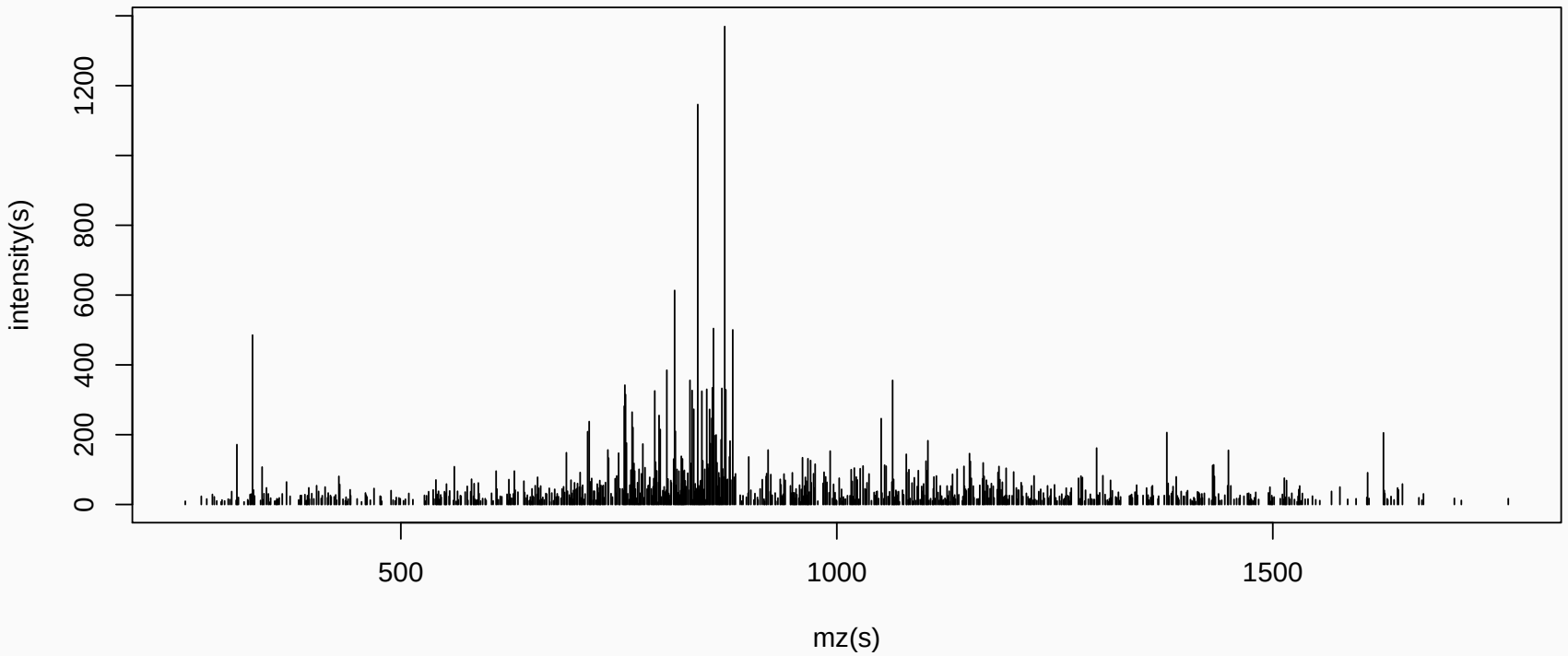
```
ms
# MSn experiment data ("OnDiskMSnExp")
# Object size in memory: 50.52 Mb
# - - - Spectra data - - -
#  MS level(s): 1 2 3
#  Number of spectra: 108844
#  MSn retention times: 0:0 - 120:0 minutes
# - - - Processing information - - -
# Data loaded [Fri Nov 24 14:09:00 2017]
#  MSnbase version: 2.5.0
# - - - Meta data  - - -
# phenoData
#   rowNames:
#     LUM2_01470_KS_L1-5-2_EC17-123-150In2_55oC_NLtrig.mzML.gz
#   varLabels: sampleNames
#   varMetadata: labelDescription
# Loaded from:
#   LUM2_01470_KS_L1-5-2_EC17-123-150In2_55oC_NLtrig.mzML.gz
# protocolData: none
# featureData
#   featureNames: F1.S000001 F1.S000002 ... F1.S108844 (108844
#     total)
#   fvarLabels: fileIdx spIdx ... spectrum (29 total)
#   fvarMetadata: labelDescription
# experimentData: use 'experimentData(object)'
```

The MSnExp object 2

```
head(featureNames(ms))
# [1] "F1.S000001" "F1.S000002" "F1.S000003" "F1.S000004" "F1.S000005"
# [6] "F1.S000006"

fData(ms)[1:2, 1:6]
#           fileIdx spIdx centroided smoothed seqNum acquisitionNum
# F1.S000001       1     1          NA       NA      1              1
# F1.S000002       1     2          NA       NA      2              2

ms[[1]]
# Object of class "Spectrum1"
#  Retention time: 0:0
#  MSn level: 1
#  Total ion count: 32197
#  Polarity: 1
```

Plot a Spectrum of an `MSnExp` object

```
s <- ms[[95]]
plot(mz(s), intensity(s), type="h")
```



Filter by MS level

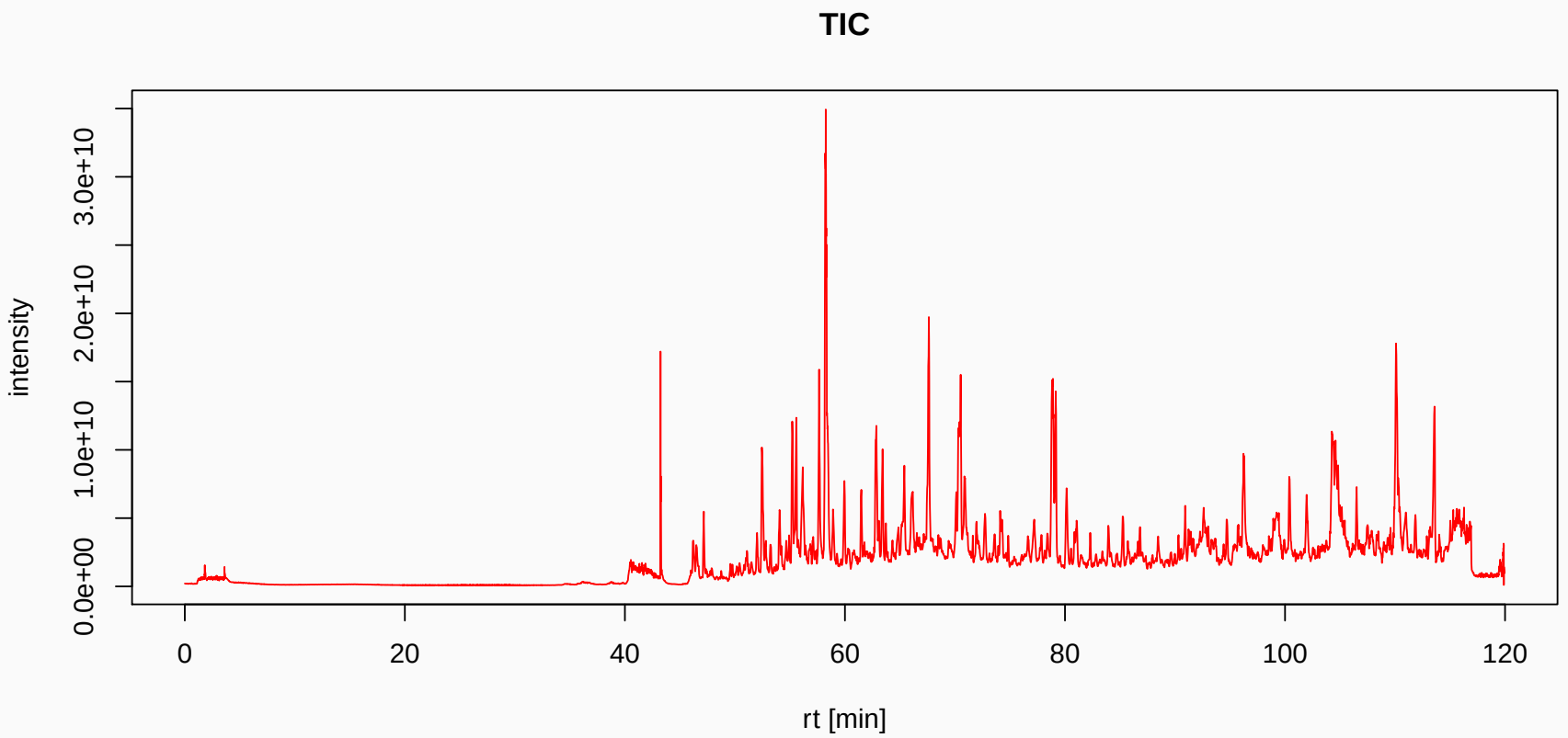
```
ms1 <- ms[fData(ms)$msLevel == 1]

ms1 <- filterMsLevel(ms, 1)
ms2 <- filterMsLevel(ms, 2)
ms3 <- filterMsLevel(ms, 3)
```

The MSnExp object 3

```
ms1
# MSn experiment data ("OnDiskMSnExp")
# Object size in memory: 1.81 Mb
# - - - Spectra data - - -
#  MS level(s): 1
#  Number of spectra: 5565
#  MSn retention times: 0:0 - 119:58 minutes
# - - - Processing information - - -
# Data loaded [Fri Nov 24 14:09:00 2017]
# Filter: select MS level(s) 1 [Sat Nov 25 17:29:30 2017]
#  MSnbase version: 2.5.0
# - - - Meta data  - - -
# phenoData
#   rowNames:
#     LUM2_01470_KS_L1-5-2_EC17-123-150In2_55oC_NLtrig.mzML.gz
#   varLabels: sampleNames
#   varMetadata: labelDescription
# Loaded from:
#   LUM2_01470_KS_L1-5-2_EC17-123-150In2_55oC_NLtrig.mzML.gz
# protocolData: none
# featureData
#   featureNames: F1.S000001 F1.S000002 ... F1.S108817 (5565 total)
#   fvarLabels: fileIdx spIdx ... spectrum (29 total)
#   fvarMetadata: labelDescription
# experimentData: use 'experimentData(object)'
```

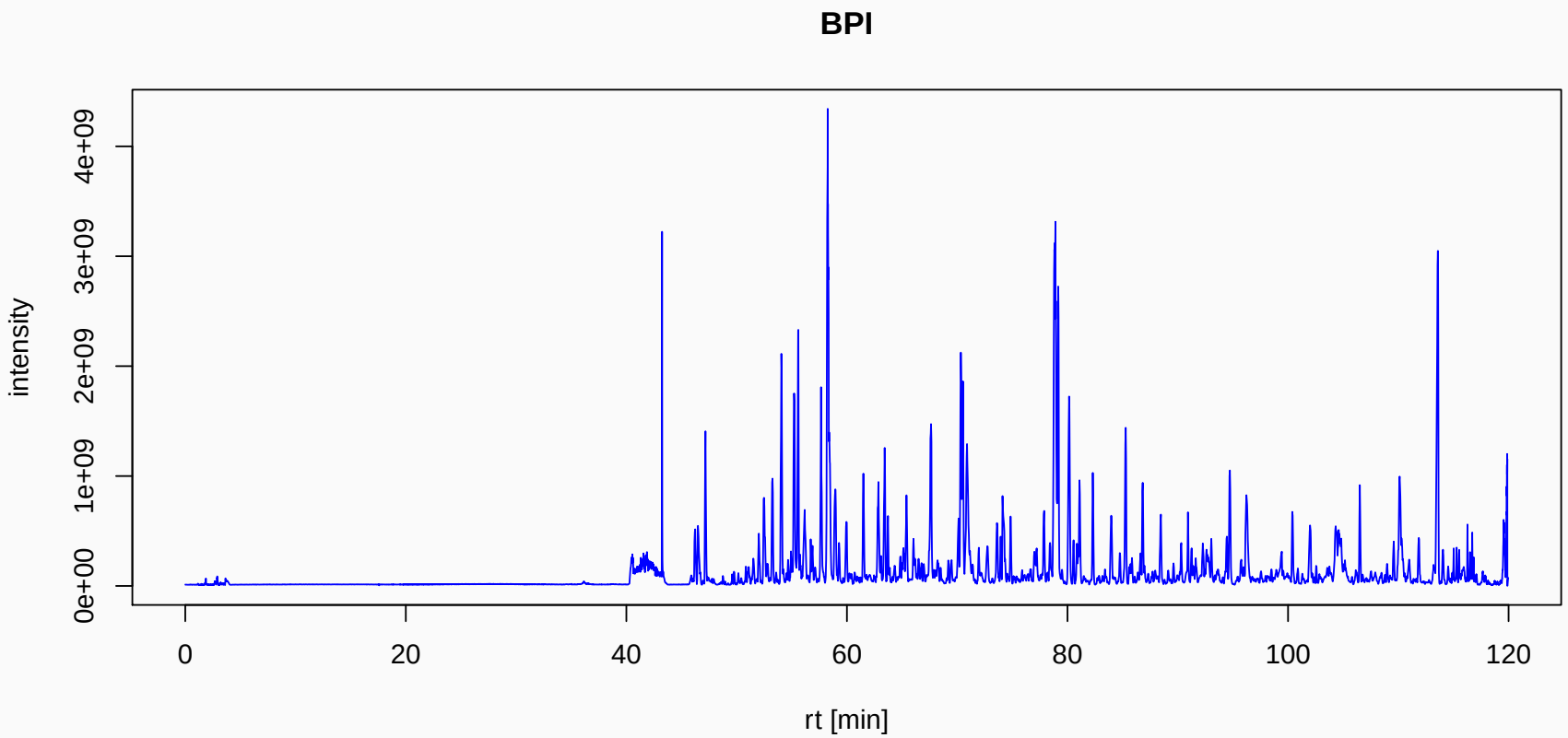
TIC

```
plot(rtime(ms1) / 60, tic(ms1),
     type="l", col="red",
     main="TIC", xlab="rt [min]", ylab="intensity")
```



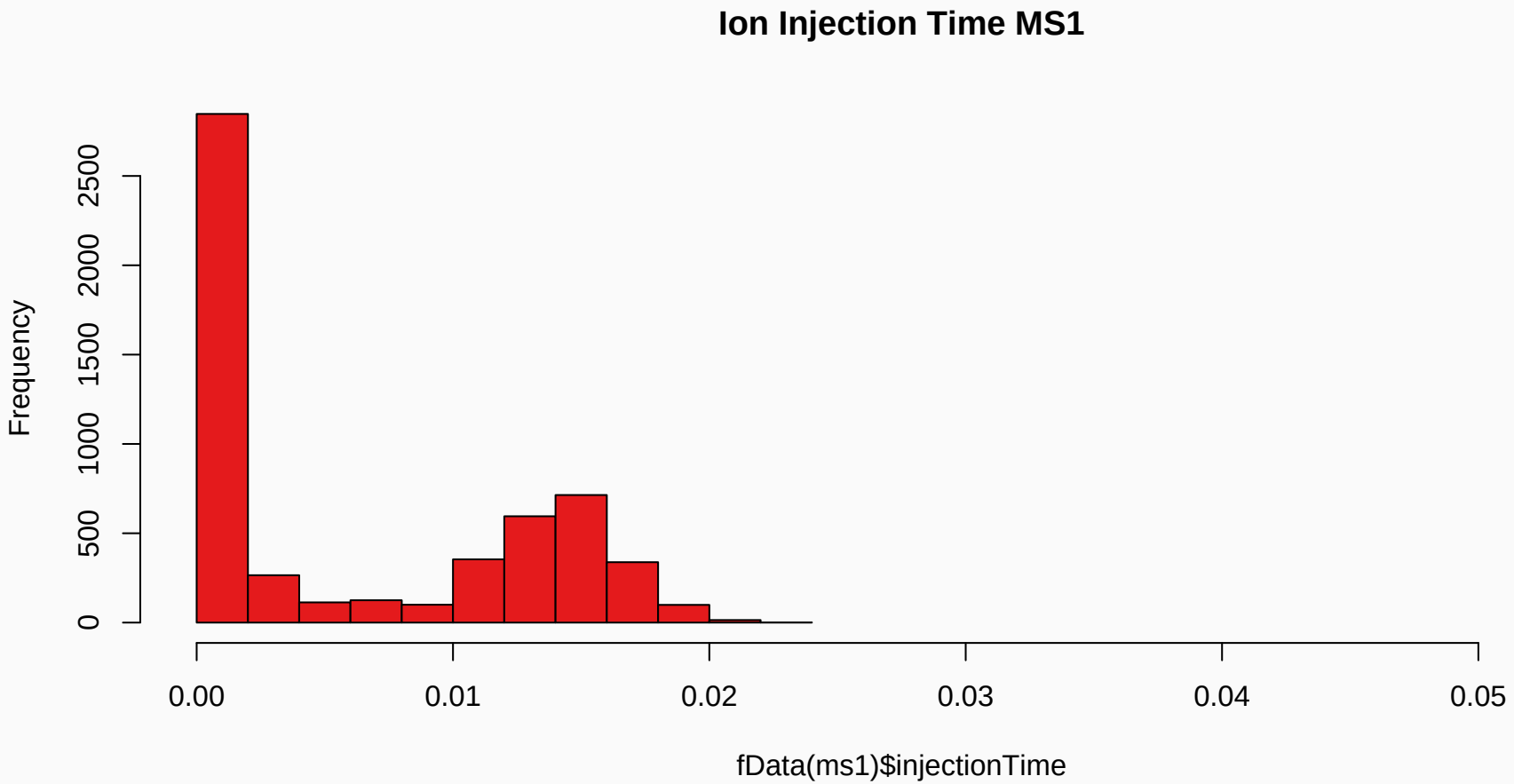
BPI

```
plot(rtime(ms1) / 60, fData(ms1)$basePeakIntensity,
     type="l", col="blue",
     main="BPI", xlab="rt [min]", ylab="intensity")
```



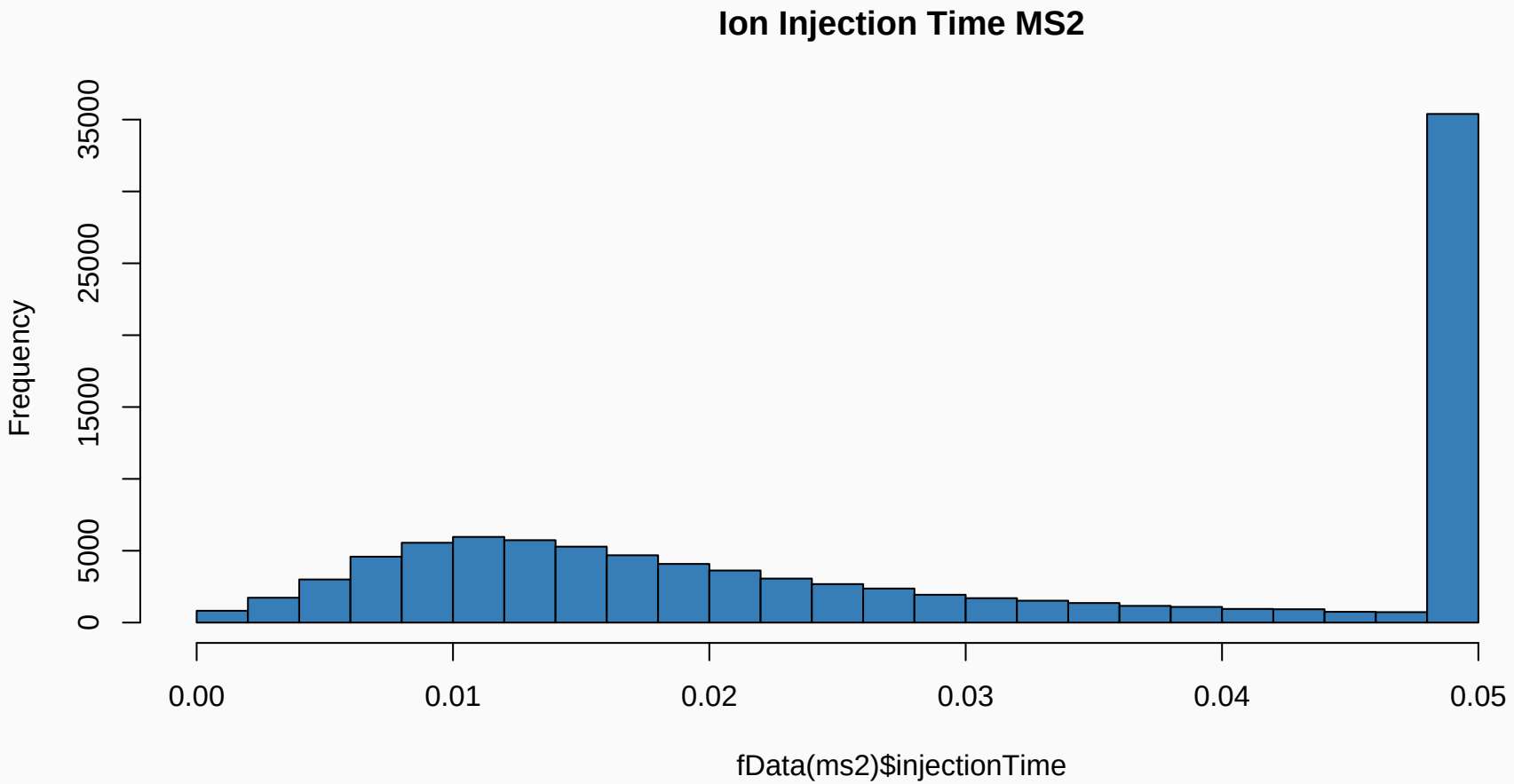
IIT MS1

```
hist(fData(ms1)$injectionTime,
     col=col["MS1"],
     main="Ion Injection Time MS1", xlim=c(0, 0.055))
```



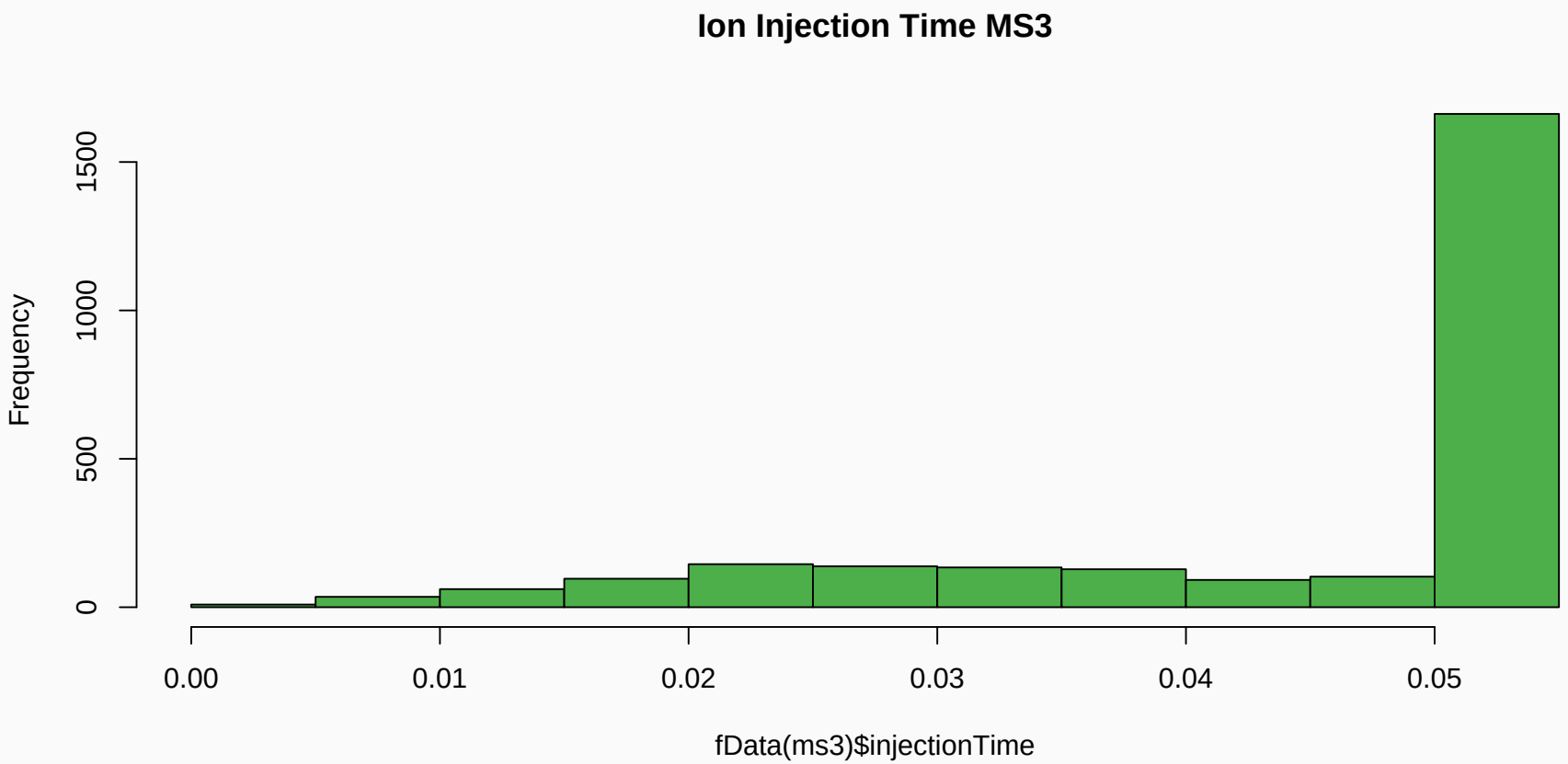
IIT MS2

```
hist(fData(ms2)$injectionTime,
     col=col["MS2"],
     main="Ion Injection Time MS2", xlim=c(0, 0.055))
```



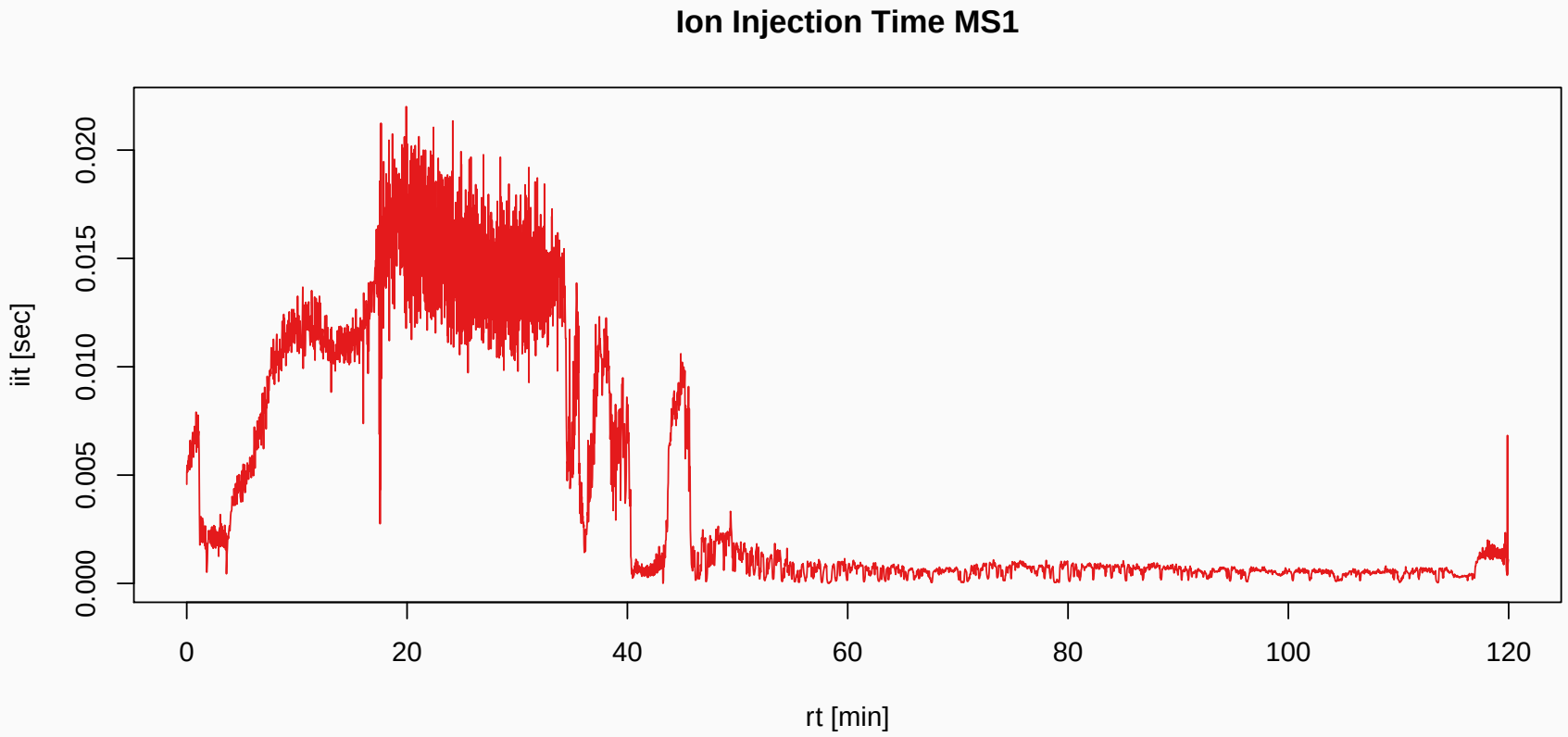
IIT MS3

```
hist(fData(ms3)$injectionTime,
     col=col["MS3"],
     main="Ion Injection Time MS3", xlim=c(0, 0.055))
```



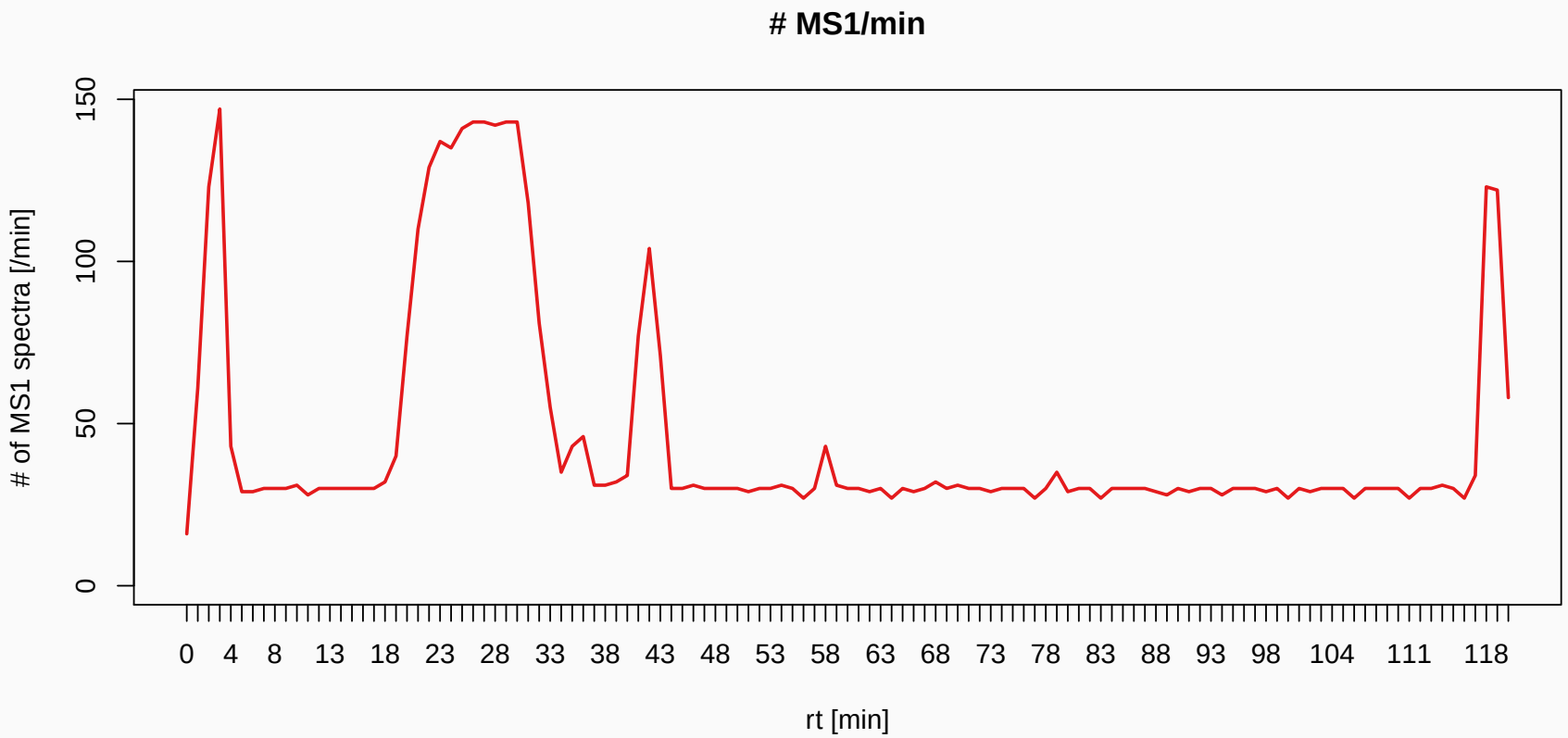
IIT vs RT MS1

```
plot(rtime(ms1) / 60, fData(ms1)$injectionTime,
     type="l", col=col["MS1"],
     main="Ion Injection Time MS1",
     xlab="rt [min]", ylab="iit [sec]")
```



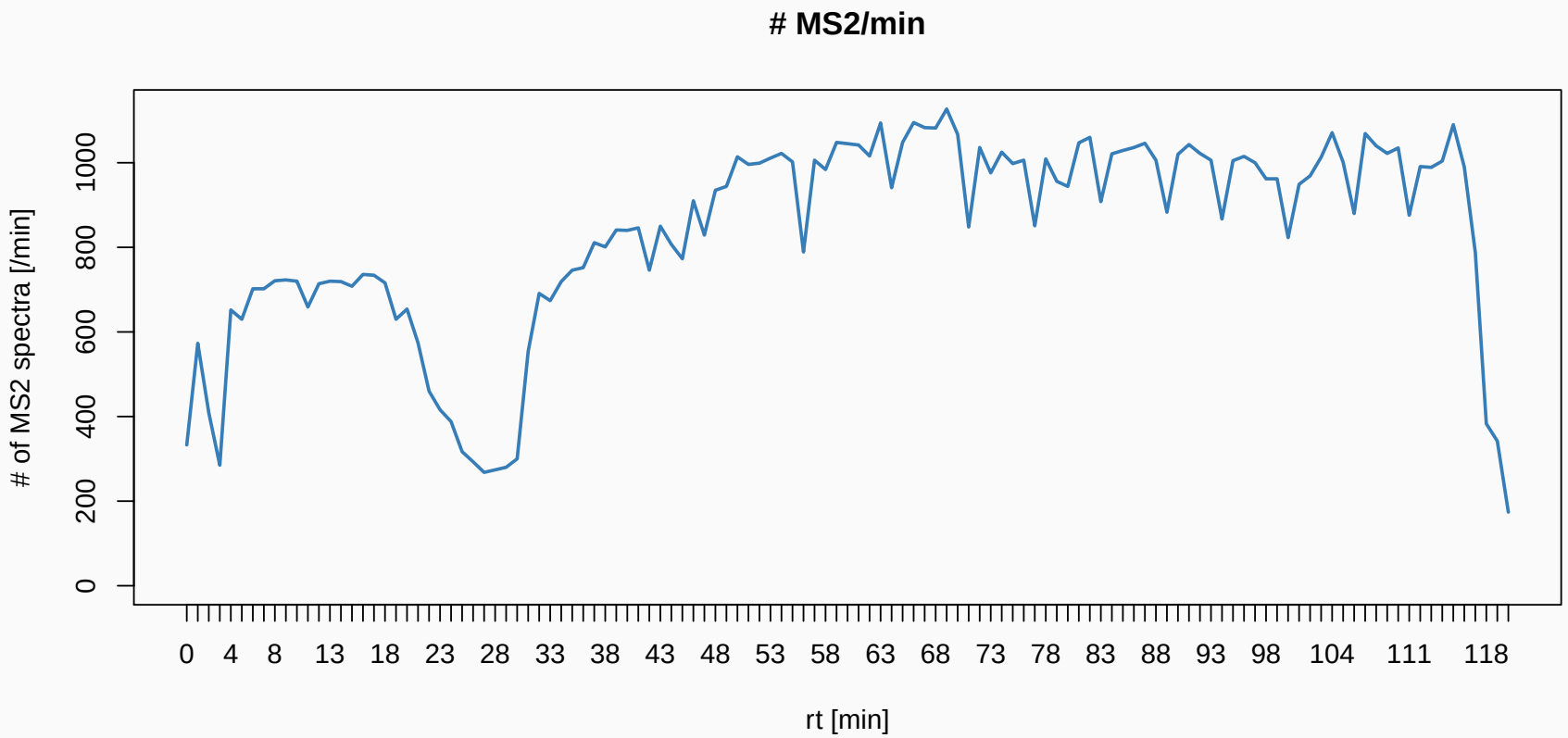
MS1 per minute

```
plot(table(round(rtime(ms1) / 60)),
     type="l", col=col["MS1"], main="# MS1/min",
     xlab="rt [min]", ylab="# of MS1 spectra [/min]")
```



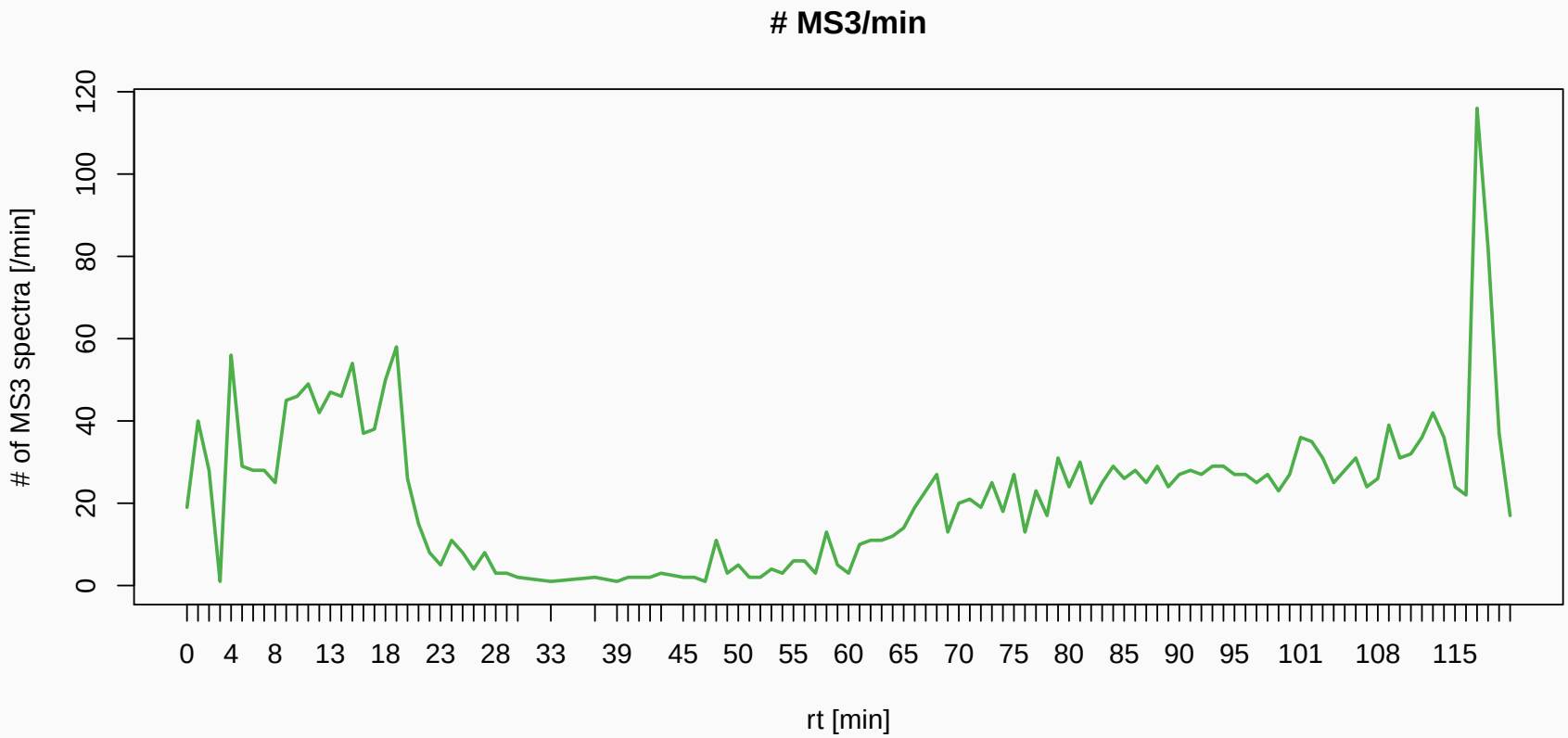
MS2 per minute

```
plot(table(round(rtime(ms2) / 60)),
     type="l", col=col["MS2"], main="# MS2/min",
     xlab="rt [min]", ylab="# of MS2 spectra [/min]")
```



MS3 per minute

```
plot(table(round(rtime(ms3) / 60)),
     type="l", col=col["MS3"], main="# MS3/min",
     xlab="rt [min]", ylab="# of MS3 spectra [/min]")
```



NL trigger (Jiang et al. 2017)

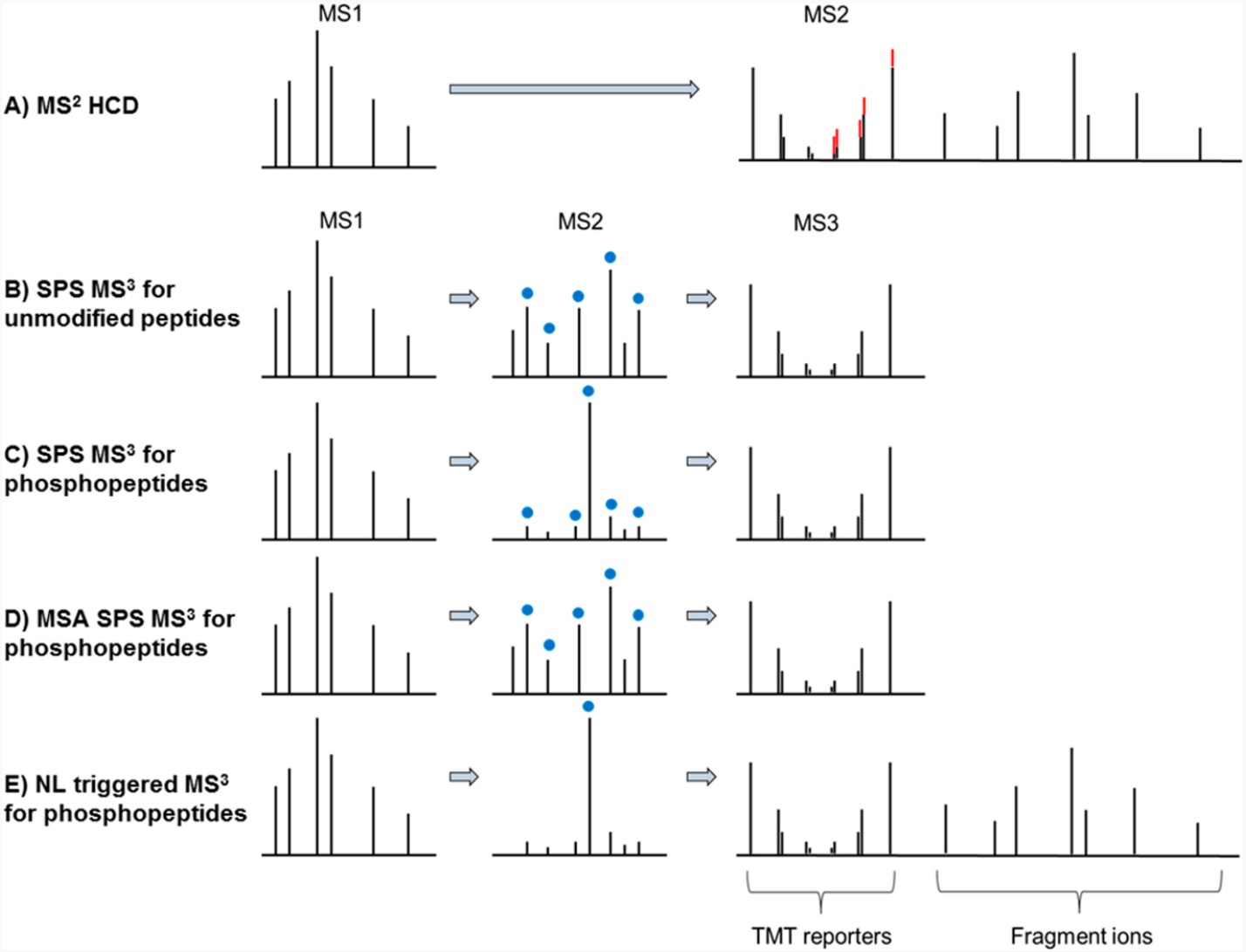


Figure 1: (Fig. 1 in Jiang et al. 2017)

NL trigger/Thermo Orbitrap Lumos

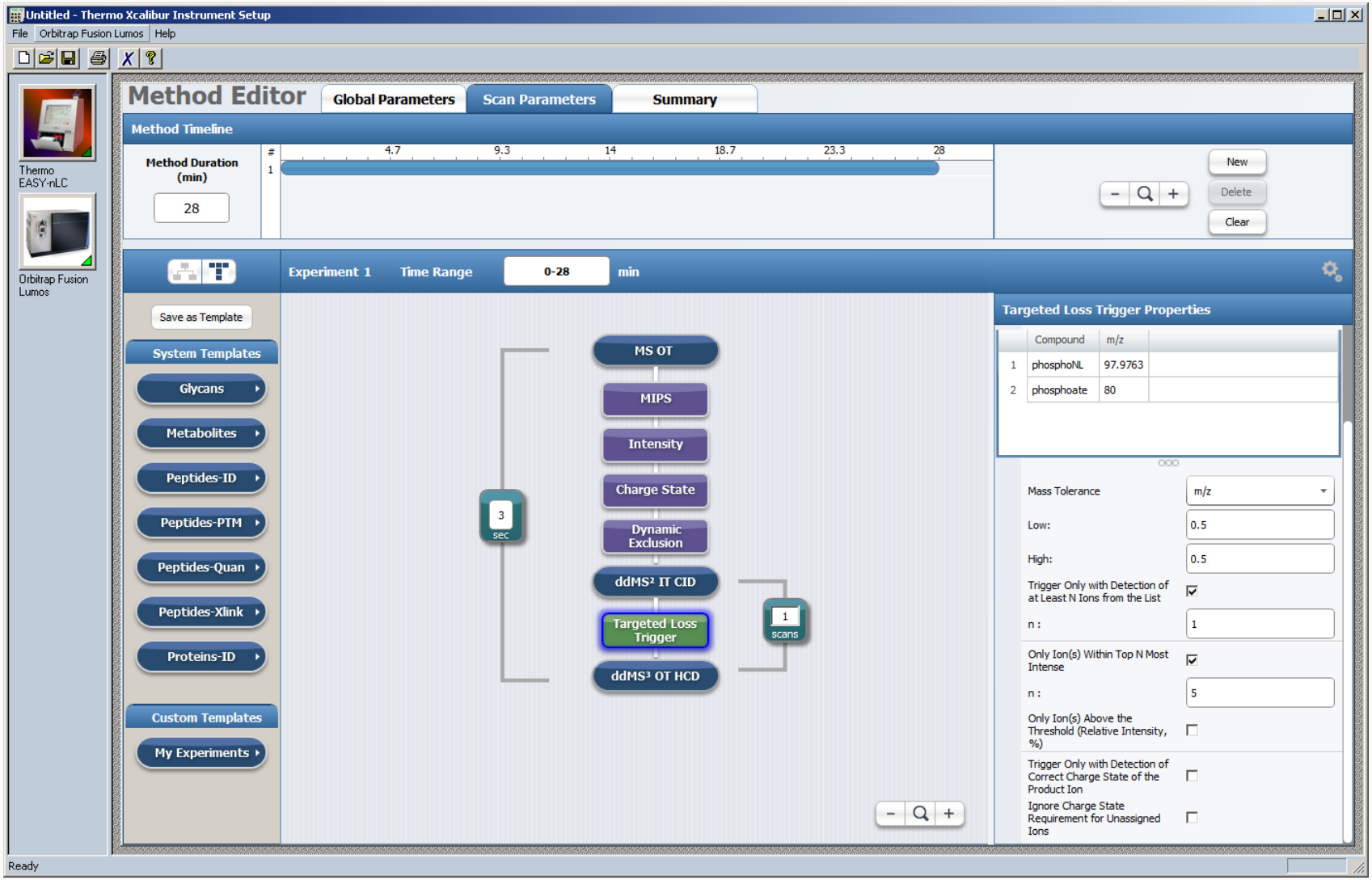


Figure 2

NL trigger

Own NL trigger

MS2: precursor mz - mz of top 5 peaks $\approx$ 80/97 [Da]

Thermo NL trigger

MS2 $\leftrightarrow$ MS3

Verification

Comparison of MS2 scan numbers

NL trigger/How to find our own trigger? 1

```
findTop5mz <- function(x) {
    mz(x)[order(intensity(x), decreasing=TRUE)[1:5]]
}
top5mz <- spectrapply(ms2, findTop5mz)
top5mz <- do.call(rbind, top5mz)
top5mz[1:2, ]
#                [,1]     [,2]     [,3]    [,4]    [,5]
# F1.S000003 507.9660 619.1350 427.9571 699.134 601.090
# F1.S000004 676.1671 427.9181 756.1360 508.006 409.902

deltamz <- precursorMz(ms2) - top5mz
deltamz[1:2, ]
#                 [,1]      [,2]      [,3]       [,4]      [,5]
# F1.S000003  55.58902 -55.57996  135.5980 -135.57898 -37.53497
# F1.S000004 -84.10120 164.14780 -164.0702   84.05984 182.16382
```

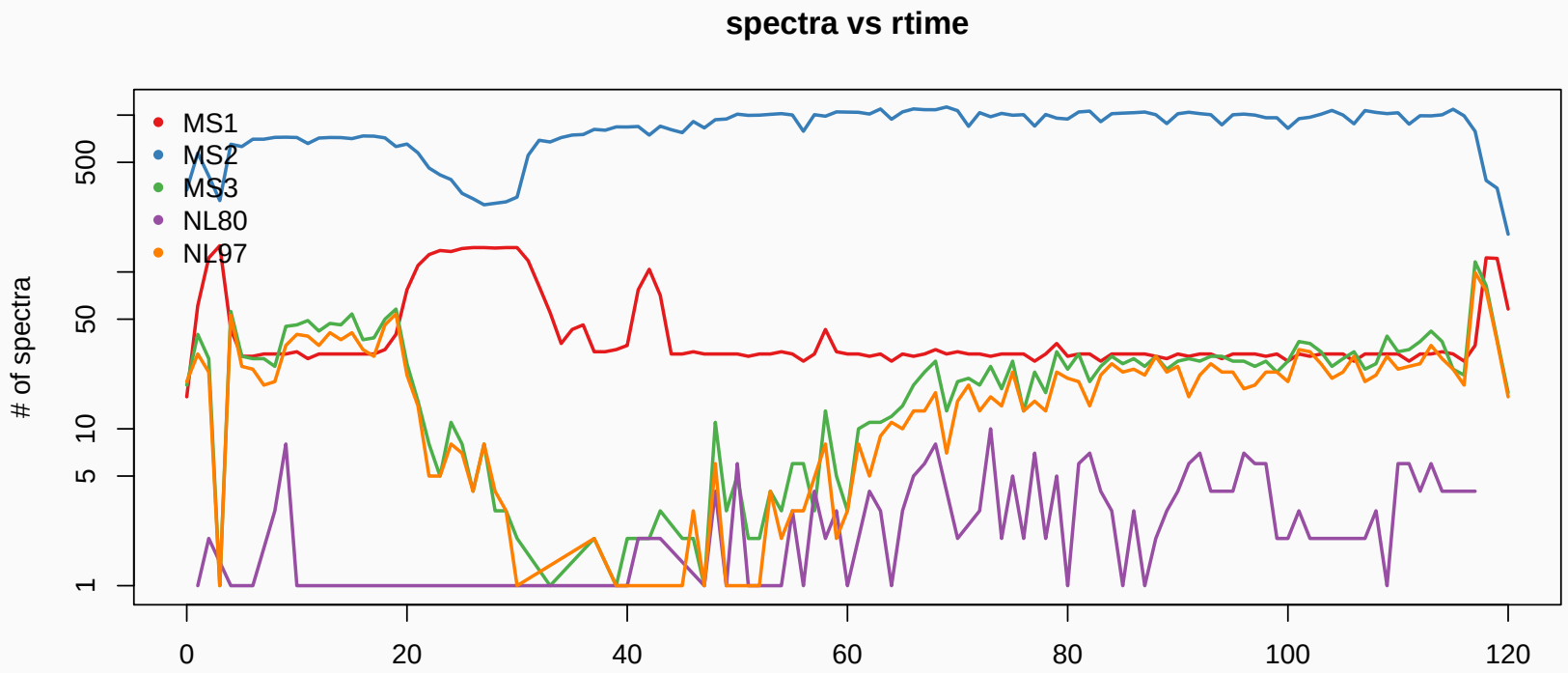
NL trigger/How to find our own trigger? 2

```
nl <- c(phosphate=80, phosphoNL=97.9763)
tolerance <- 0.5 # [Da]

# How many phosphates?
nl80 <- as.logical(rowSums(
    abs(deltamz - nl["phosphate"]) <= tolerance)
)
sum(nl80)
# [1] 276

# How many phosphoNL?
nl97 <- as.logical(rowSums(
    abs(deltamz - nl["phosphoNL"]) <= tolerance)
)
sum(nl97)
# [1] 2124
```

->

NL trigger/Spectra + NL vs RT log

```
plot(NA, ylim=c(1, 1100), xlim=c(0, 120), log="y",
     main="spectra vs rtime", xlab="rt [min]", ylab="# of spectra")
l <- list(ms1, ms2, ms3, ms2[nl80], ms2[nl97])
for (i in seq(along=l)) {
    lines(table(round(rtime(l[[i]]) / 60)), col=col[i], type="l")
}
legend("topleft", legend=names(col), col=col, pch=20, bty="n")
```



NL trigger/Find device trigger; the tricky part 1

```
fData(ms2)$filterString[1:2]
# [1] "ITMS + c NSI t d Full ms2 563.5551@cid35.00 [150.0000-1138.0000]"
# [2] "ITMS + c NSI t d Full ms2 592.0658@cid35.00 [158.0000-1195.0000]"
fData(ms3)$filterString[1:2]
# [1] "FTMS + p NSI d Full ms3 933.4402@cid35.00 915.4110@hcd30.00 [100.0000-2000.0000]"
# [2] "FTMS + p NSI d Full ms3 850.0895@cid35.00 752.1000@hcd30.00 [100.0000-2000.0000]"

ms2df <- data.frame(scanId=scanIndex(ms2),
                    pcCid=gsub(".*ms2 *([0-9.]+@cid[0-9.]+) .*", "\\1",
                               fData(ms2)$filterString),
                    stringsAsFactors=FALSE)
```

	scanId	pcCid
F1.S000003	3	563.5551@cid35.00
F1.S000004	4	592.0658@cid35.00

```
ms3df <- data.frame(scanId=scanIndex(ms3),
                    pcCid=gsub(".*ms3 *([0-9.]+@cid[0-9.]+) .*", "\\1",
                               fData(ms3)$filterString),
                    stringsAsFactors=FALSE)
```

	scanId	pcCid
F1.S000040	40	933.4402@cid35.00
F1.S000054	54	850.0895@cid35.00

NL trigger/Find device trigger; the tricky part 2

```
nldf <- merge(ms3df, ms2df, by="pcCid",
              all.x=TRUE, all.y=FALSE,
              sort=FALSE, suffixes=c(".ms3", ".ms2"))
nldf$delta <- nldf$scanId.ms3 - nldf$scanId.ms2
```

pcCid	scanId.ms3	scanId.ms2	delta
933.4402@cid35.00	40	38	2
850.0895@cid35.00	54	52	2
644.5992@cid35.00	67	11506	-11439
644.5992@cid35.00	67	11081	-11014
644.5992@cid35.00	67	4763	-4696
644.5992@cid35.00	67	4364	-4297

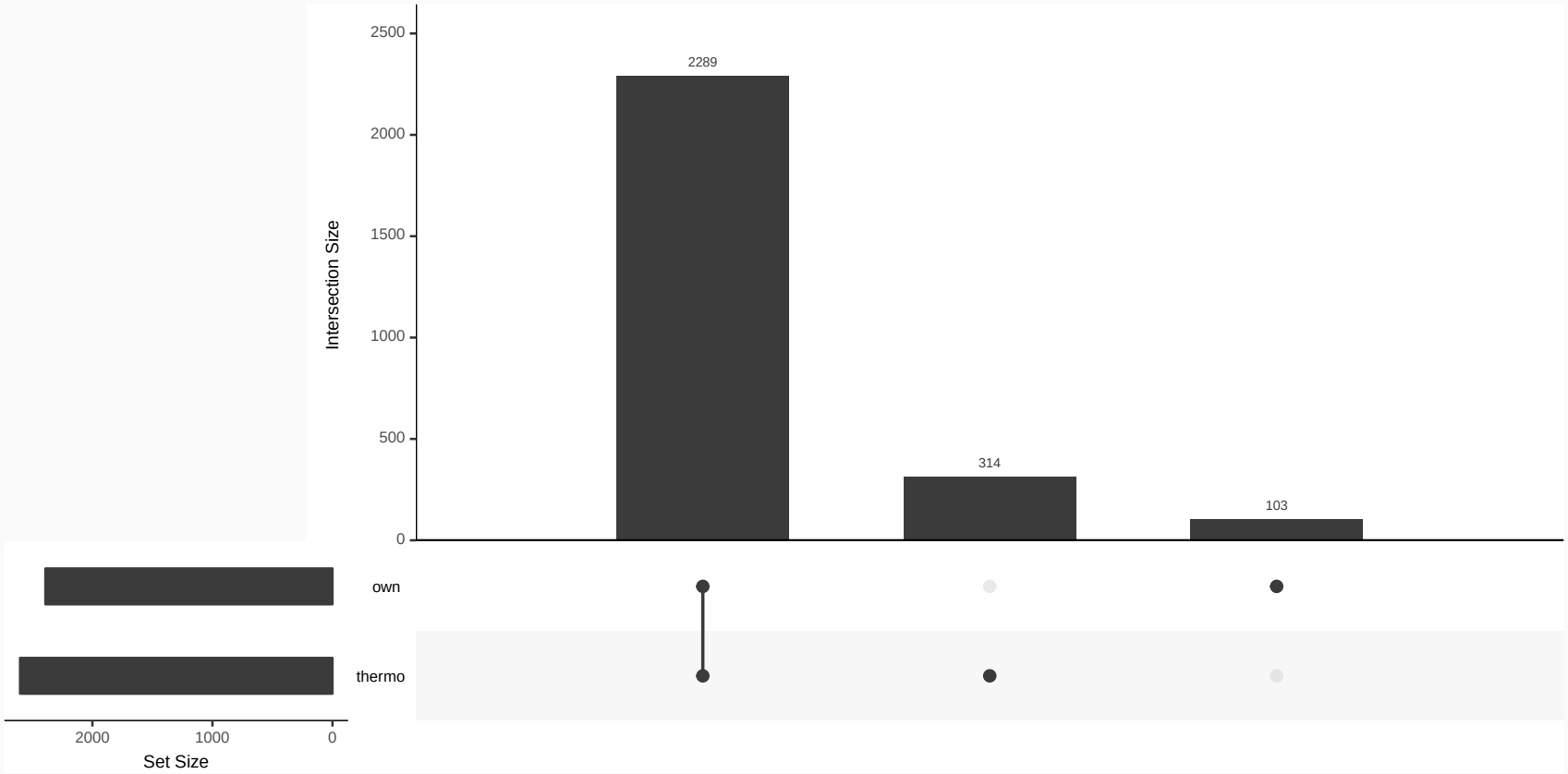
NL trigger/Find device trigger; the tricky part 3

```
nldf <- nldf[nldf$delta > 0,]

nldf <- nldf[order(nldf$scanId.ms3, nldf$delta),]

nldf <- nldf[!duplicated(nldf$scanId.ms3),]
```

	pcCid	scanId.ms3	scanId.ms2	delta
1	933.4402@cid35.00	40	38	2
2	850.0895@cid35.00	54	52	2
10	644.5992@cid35.00	67	65	2
12	699.0558@cid35.00	82	80	2
15	636.8463@cid35.00	83	81	2
16	757.3283@cid35.00	108	106	2

NL trigger/Find common trigger/differences 1

```
ms2idNL <- list(own=scanIndex(ms2)[nl80 | nl97],
                thermo=nldf$scanId.ms2)

library("UpSetR")
upset(fromList(ms2idNL), order.by="freq")
```

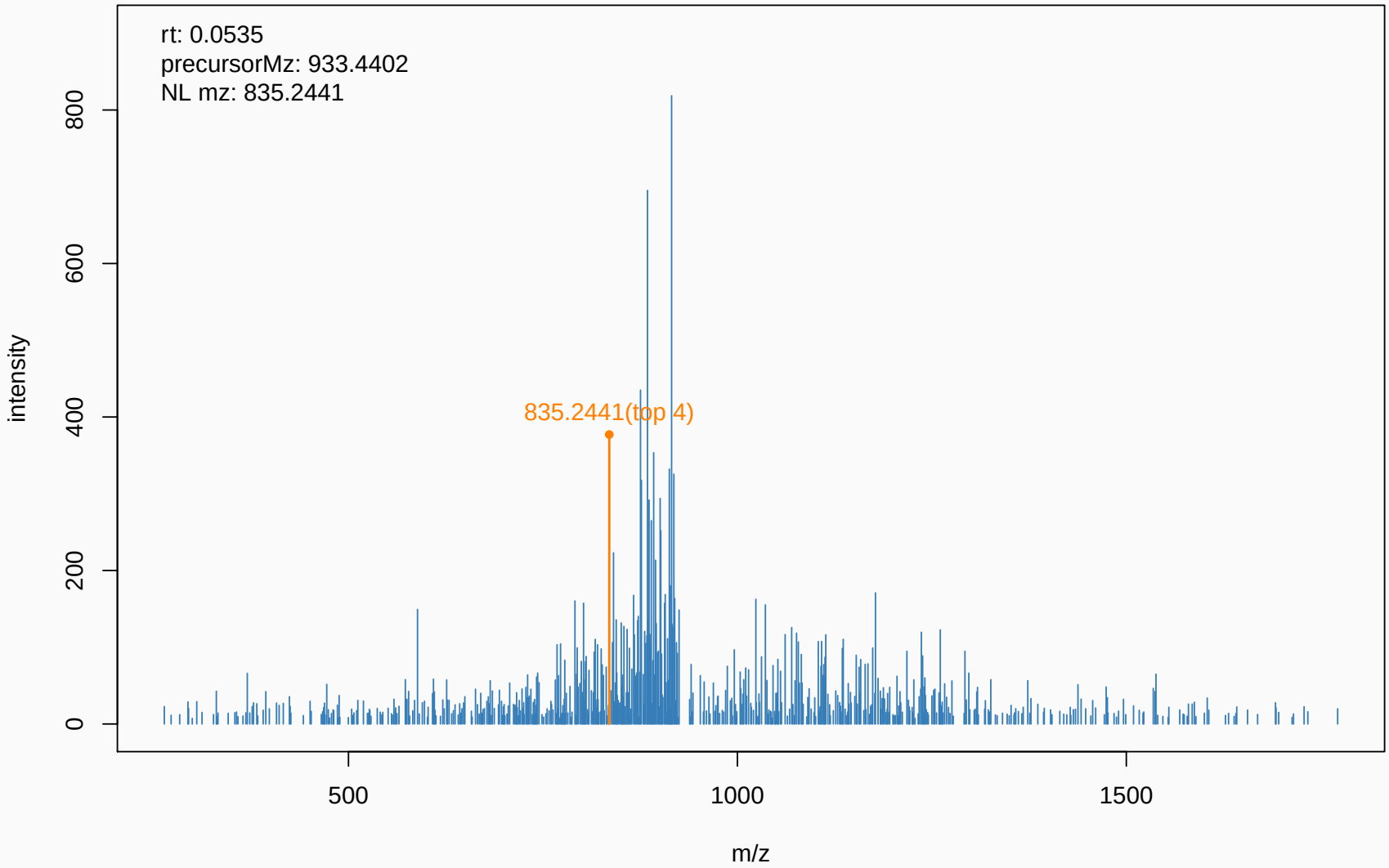


NL trigger/Find common trigger/differences 2

```
common <- intersect(ms2idNL$own, ms2idNL$thermo)
uniqueOwn <- setdiff(ms2idNL$own, ms2idNL$thermo)
uniqueThermo <- setdiff(ms2idNL$thermo, ms2idNL$own)
```

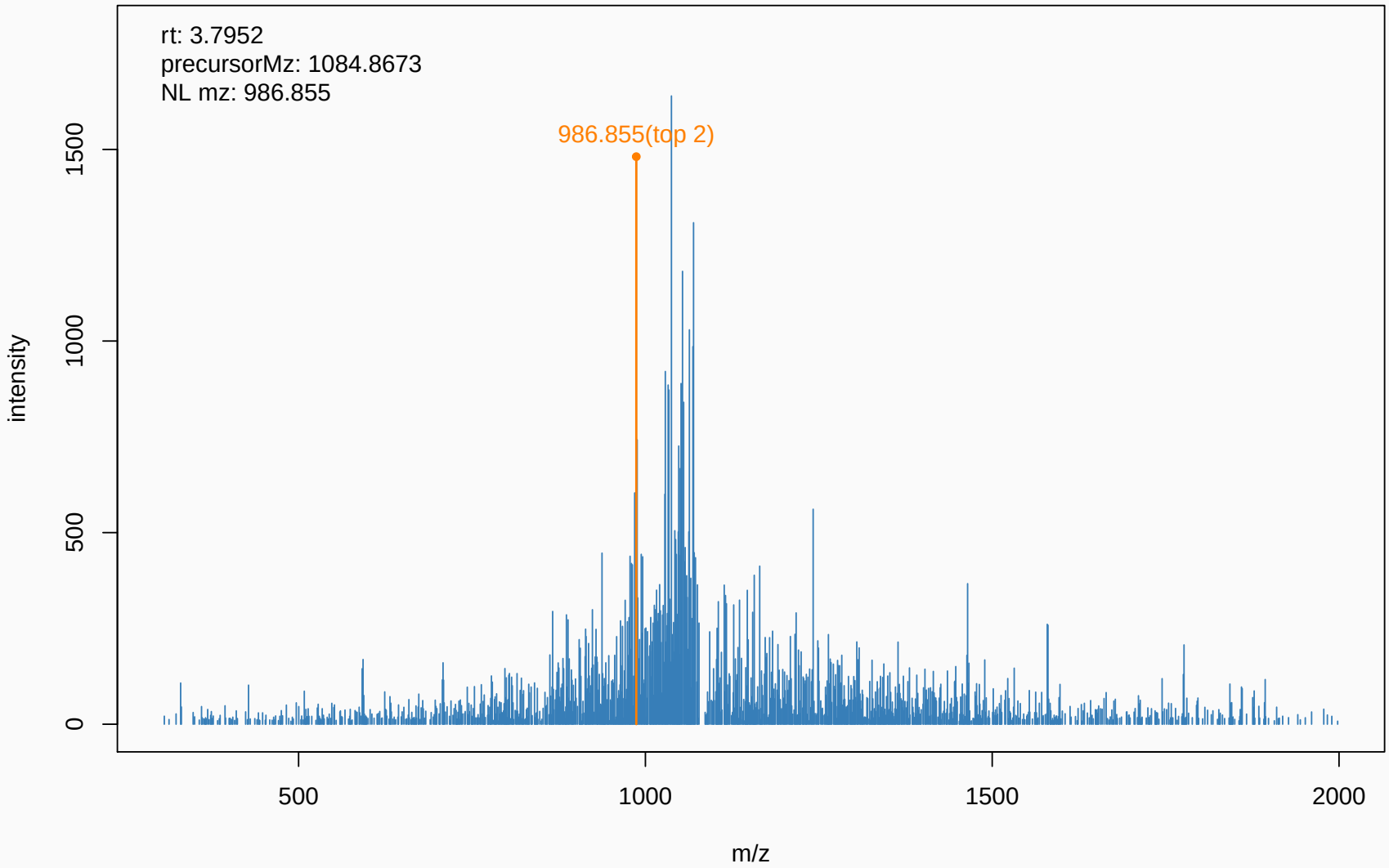
NL trigger/Common trigger

```
plotMs2(ms[[common[1]]])
```



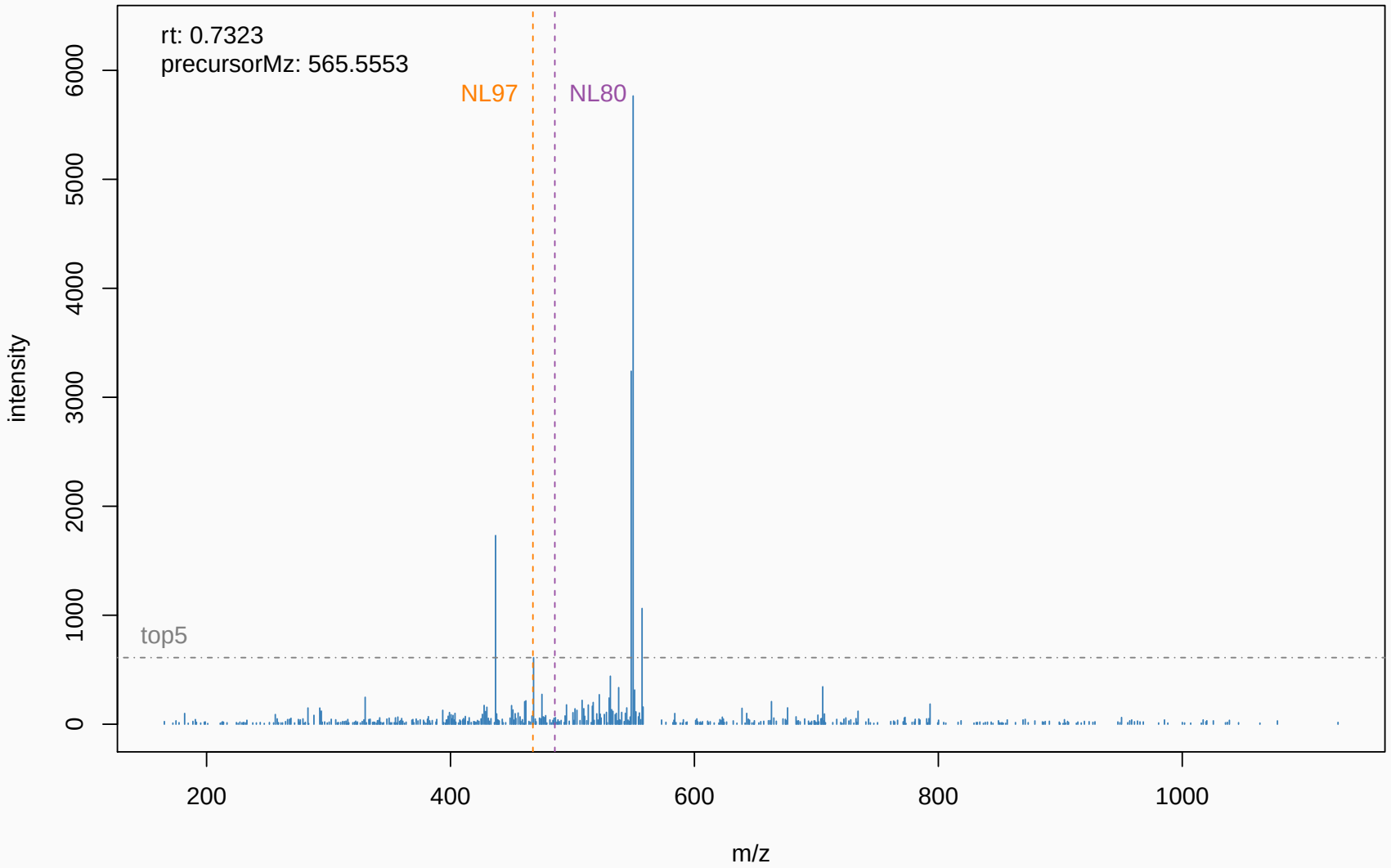
NL trigger/Should have been triggered

```
plotMs2(ms[[uniqueOwn[1]]])
```



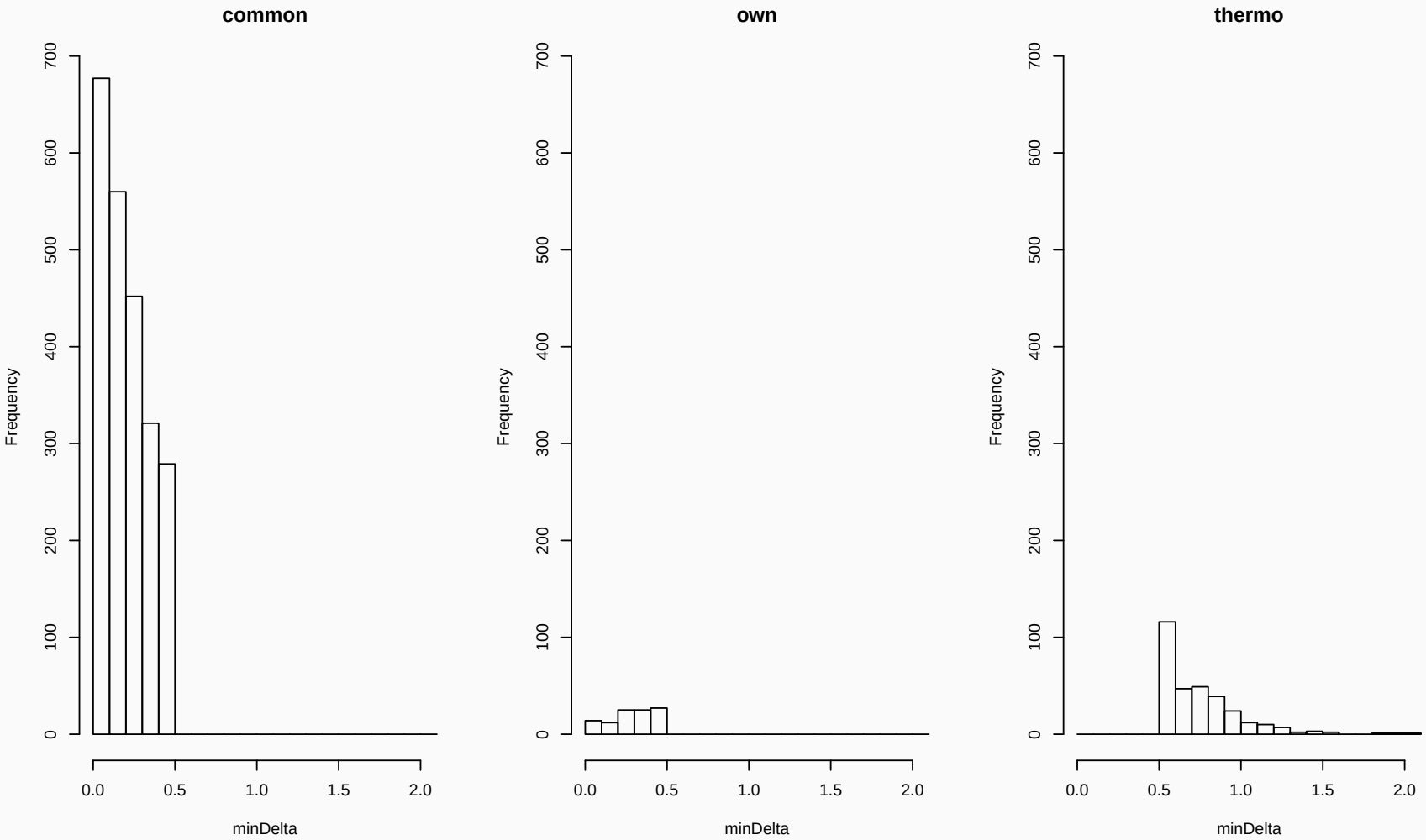
NL trigger/Was triggered but shouldn't

```
plotMs2(ms[[uniqueThermo[1]]])
```



NL trigger/Trigger tolerance

```
l <- list(common=common, own=uniqueOwn, thermo=uniqueThermo)
i <- lapply(l, match, scanIndex(ms2))
for (j in seq(along=l)) {
    minDelta <- apply(deltamz[i[[j]], ], 1, function(r)min(abs(r - rep(nl, each=length(r)))))
    hist(minDelta, breaks=seq(0, 2.1, by=0.1), ylim=c(0, 675), main=names(l)[j])
}
```



Thanks

- MSnbase: @lgatto, @jotsetung
- mzR: @sneumann, @lgatto, @jotsetung

Availability

https://github.com/sgibb/talk-odense-20171127/

github: @sgibb / twitter: @sebastiangibb

References

Gatto, Laurent, and Kathryn S. Lilley. 2011. "MSnbase-an R/Bioconductor Package for Isobaric Tagged Mass Spectrometry Data Visualization, Processing and Quantitation." *Bioinformatics* 28 (2). Oxford University Press (OUP): 288–89. doi:10.1093/bioinformatics/btr645.

Jiang, Xiaoyue, Ryan Bomgarden, Joseph Brown, Devin L. Drew, Aaron M. Robitaille, Rosa Viner, and Andreas R. Huhmer. 2017. "Sensitive and Accurate Quantitation of Phosphopeptides Using TMT Isobaric Labeling Technique." *Journal of Proteome Research* 16 (11). American Chemical Society (ACS): 4244–52. doi:10.1021/acs.jproteome.7b00610.