02−MathDAMP−Elements

This notebook introduces the basic functionality of the *Math*DAMP package. The core features of the package are described and the usage of individual functions is demonstrated with multiple options. For examples of performing common types of differential analysis of metabolite profiles, please refer to the notebooks: 03−MathDAMP−TwoDatasets.nb, 04−MathDAMP−Outliers.nb, 05−MathDAMP−TwoGroups.nb, and 06−MathDAMP−MultipleGroups.nb.

First, the *Math*DAMP package has to be loaded. Please assign the path leading to the *Math*DAMP files to the `MathDAMPPath` variable.

```
MathDAMPPath = "/home/baran/math/ms/MathDAMP.1.0.0/";
<< (MathDAMPPath <> "MathDAMP.m")
```

```
MathDAMP version 1.0.0 loaded (2006/04/26)

This program is distributed in the hope that it will be useful, but WITHOUT
  ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

Usage reference information for any of the functions from the *Math*DAMP package can be displayed by executing ? followed by the function's name. For details regarding the implementation of the functions please refer to the MathDAMP.nb notebook.

```
?DAMPPlotChromatogram
```

```
DAMPPlotChromatogram[{msdata,msdata,...},mz,options] plots multiple overlaid chromatograms/electropherograms from the list of msdatas
  corresponding to m/z mz (mz may be a list of m/z values as well). The DAMPGenColors function is used by default to assign colors to
  individual chromatograms/electropherograms. Custom colors may be specified via the PlotOptions option (and the enclosed PlotStyle option).
Options:
PlotOptions - list of options for the MultipleListPlot function
  which is used internally for plotting (default: see the output of Options[DAMPPlotChromatogram]),
AnnotationTable - annotation table for the labeling of the chromatogram/electropherogram (default: {})
Resolution - resolution to which the msdata were binned along the m/z dimension. The annotation table passed through the AnnotationTable option
  will be rounded accordingly to ensure the appropriate appearance of annotation labels on the chromatogram/electropherogram (default: 1)
LegendData - a list of legend elements {{color,label},...}. If Automatic is specified, the
  SampleName from each msdata is used as a label (default: Automatic)
```

Data Import

*Math*DAMP functions operate on a relatively simple structure of datasets acquired by hyphenated mass spectrometry techniques. Several functions for the import of different data formats were implemented (please refer to the MathDAMP.nb notebook for details). Custom import functions for different data formats may be implemented by converting the data to MathDAMP's internal format (described in the MathDAMP.nb notebook) upon loading.
In this notebook, most of the operations will be performed on two datasets acquired by capillary electrophoresis coupled to a quadrupole mass spectrometer (CE−QMS) operated in a selected ion monitoring (SIM) mode. The two datafiles are among the sample data provided with the MathDAMP package. The data are stored in an Agilent MS format. `DAMPImportMS` function is used for importing Agilent MS files.

```
{ctrl, smpl} = DAMPImportMS[MathDAMPPath <> "/data/" <> #] & /@ {"control.ms", "sample.ms"};
```

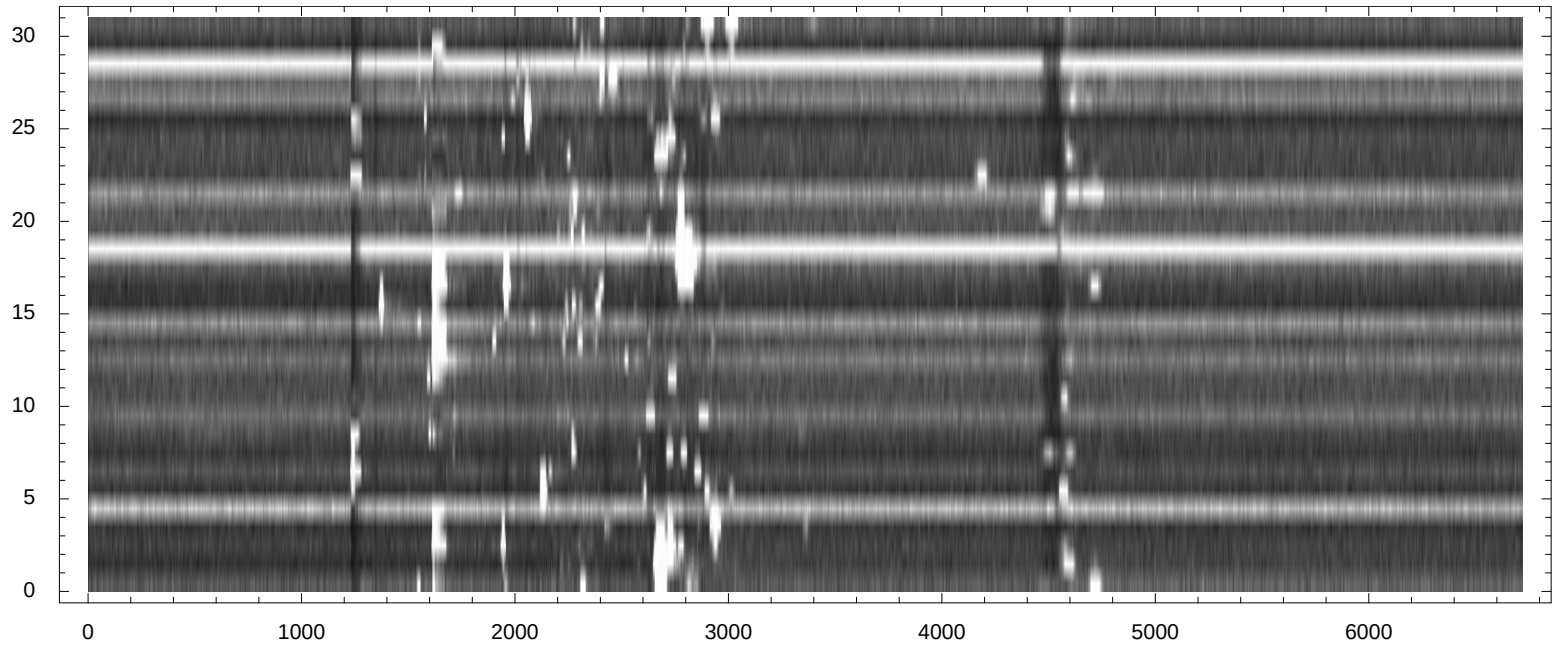
The datasets are represented as lists with 4 elements in *Mathematica*.

```
Length[ctrl]
```

```
4
```

The first element contains a matrix of signal intensities (rows being lists of signal intensities corresponding to individual chromatograms/electropherograms), the second element contains a list of m/z values, the third element contains a list of timepoints (in minutes), and the fourth element contains additional information about the dataset (as a list of rules). The dimensions of the signal intensity matrix are determined by the length of the m/z value list and the length of the list of timepoints.

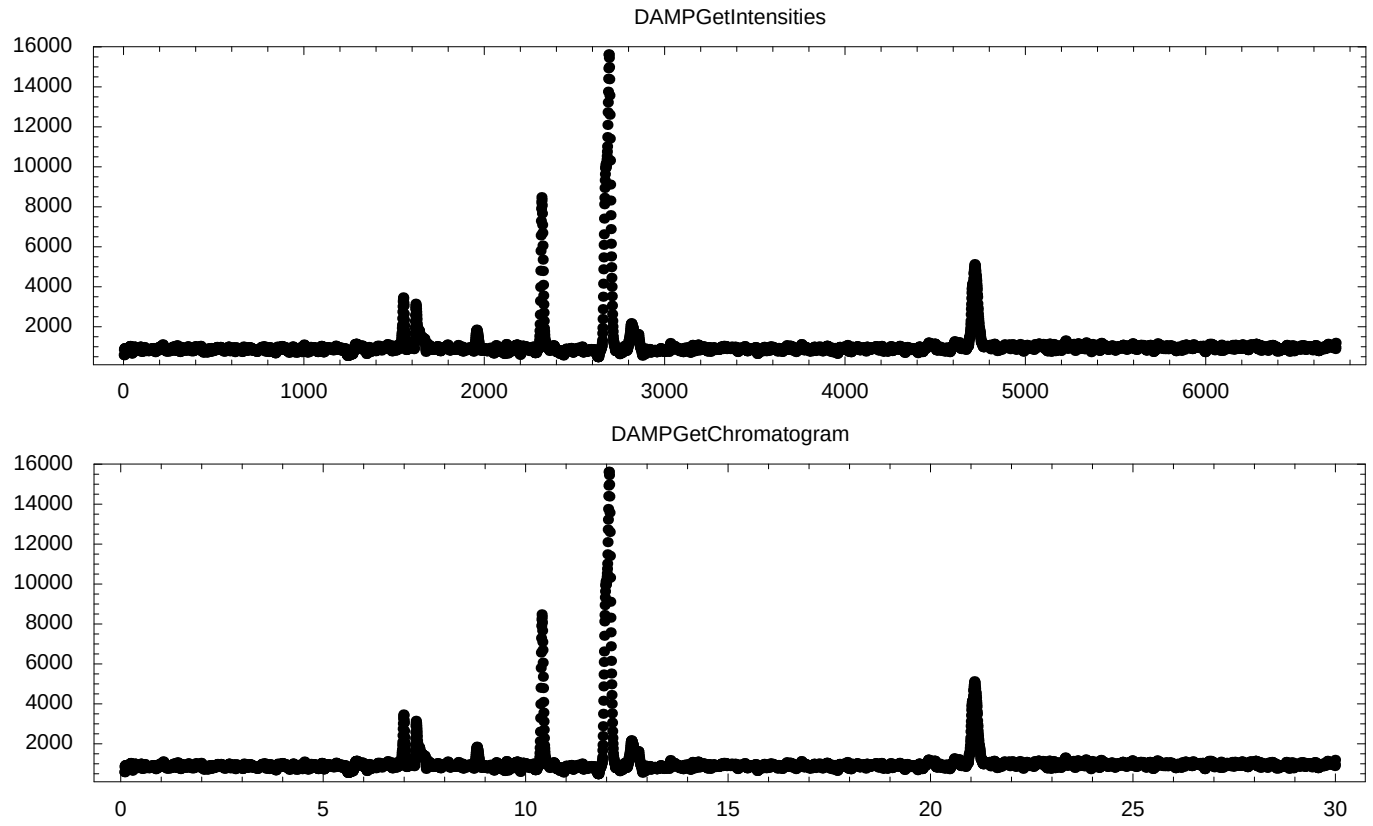
```
ListDensityPlot[ctrl[[1]], Mesh → False, ImageSize → 750, AspectRatio → .4, TextStyle → DAMPTextStyle];
ctrl[[2]]
Take[ctrl[[3]], 40] // N
ctrl[[4]]
```




```
{131., 132., 133., 134., 135., 136., 137., 138., 139., 140., 141., 142., 143., 144.,
 145., 146., 147., 148., 149., 150., 151., 152., 153., 154., 155., 156., 157., 158., 159., 160., 182.}

{0.08745, 0.0919, 0.09635, 0.1008, 0.10525, 0.1097, 0.11415, 0.118617, 0.123067, 0.127517, 0.131967, 0.136417, 0.140867, 0.145317,
 0.149767, 0.154217, 0.158667, 0.163117, 0.167567, 0.172017, 0.176467, 0.180917, 0.185367, 0.189817, 0.194267, 0.198717, 0.203167,
 0.207617, 0.212067, 0.216517, 0.220967, 0.225417, 0.229867, 0.234317, 0.238767, 0.243217, 0.247667, 0.252117, 0.256567, 0.261033}

{SampleName → control}
```

The functions `DAMPGetIntensities` and `DAMPGetChromatogram` can be used to retrieve the list of signal intensities or the chromatogram/electropherogram corresponding to a specific m/z value. Note that the latter function returns a list of `{timepoint,signal intensity}` elements.

```
ListPlot[#[ctrl, 131], Frame → True, PlotRange → All, ImageSize → 750, AspectRatio → .25, TextStyle → DAMPTextStyle, PlotLabel → #] & /@
 {DAMPGetIntensities, DAMPGetChromatogram};
```



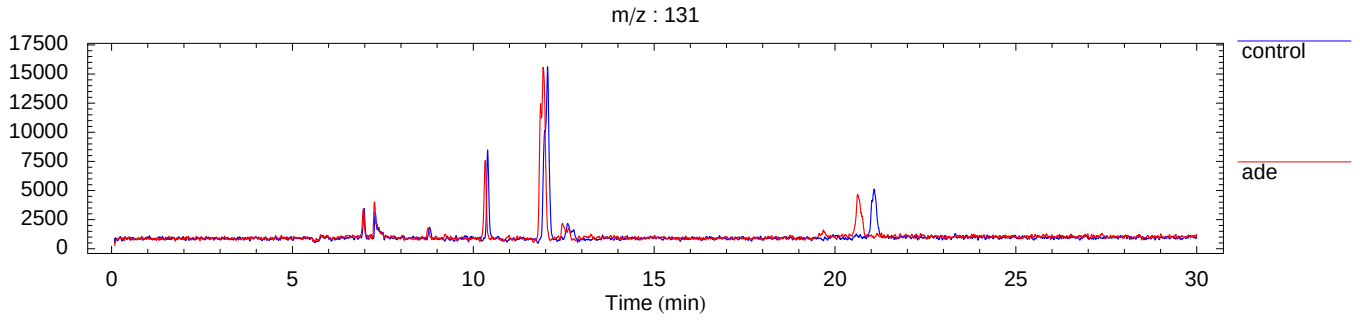
Additional functions for data import include `DAMPImportCSV`, DAMPImportMZXML, `DAMPImportCDF`, and `DAMPImportBDT`. Please execute `?FunctionName` or refer to the MathDAMP.nb notebook for details.

```
?DAMPImportMZXML

DAMPImportMZXML[filename,samplename,options] reads and processes the first level MS scans from an
  mzXML data file specified by filename into a MathDAMP format {matrix of signal intensities, list of m/z values, list of
  timepoints, additional information (list of rules)}. The sample name is specified via the second parameter (samplename).
Options:
Resolution - specifies the resolution to which the data will be binned along the m/z axis (default: 1)
ImportMode - determines the m/z dimension elements. If set to Sequential, the m/z dimension is represented by a discrete range of values determined by the
  smallest and the largest m/z value in the imported dataset (rounded according to the resolution). Resolution determines the stepsize of the
  discrete range. If set to Selective, the m/z dimension is represented by only those m/z values which are present in the imported dataset (rounded
  according to the resolution). Sequential mode is recommended for scan data, Selective mode is recommended for SIM data. (default: Sequential)
```

Plotting Chromatograms/Electropherograms

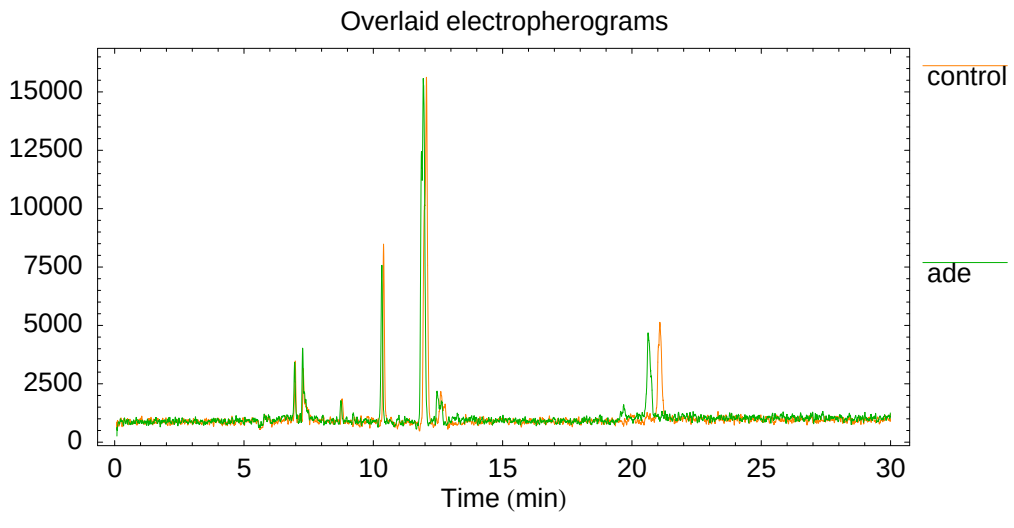
`DAMPPlotChromatogram` plots overlaid chromatograms/electropherograms from a list of datasets (passed as the first parameter) corresponding to a specific m/z value (passed as the second parameter).

```
DAMPPlotChromatogram[{ctrl, smpl}, 131];
```



The appearance of the plot may be modified by additional options (these can be listed by executing ?`DAMPPlotChromatogram`).

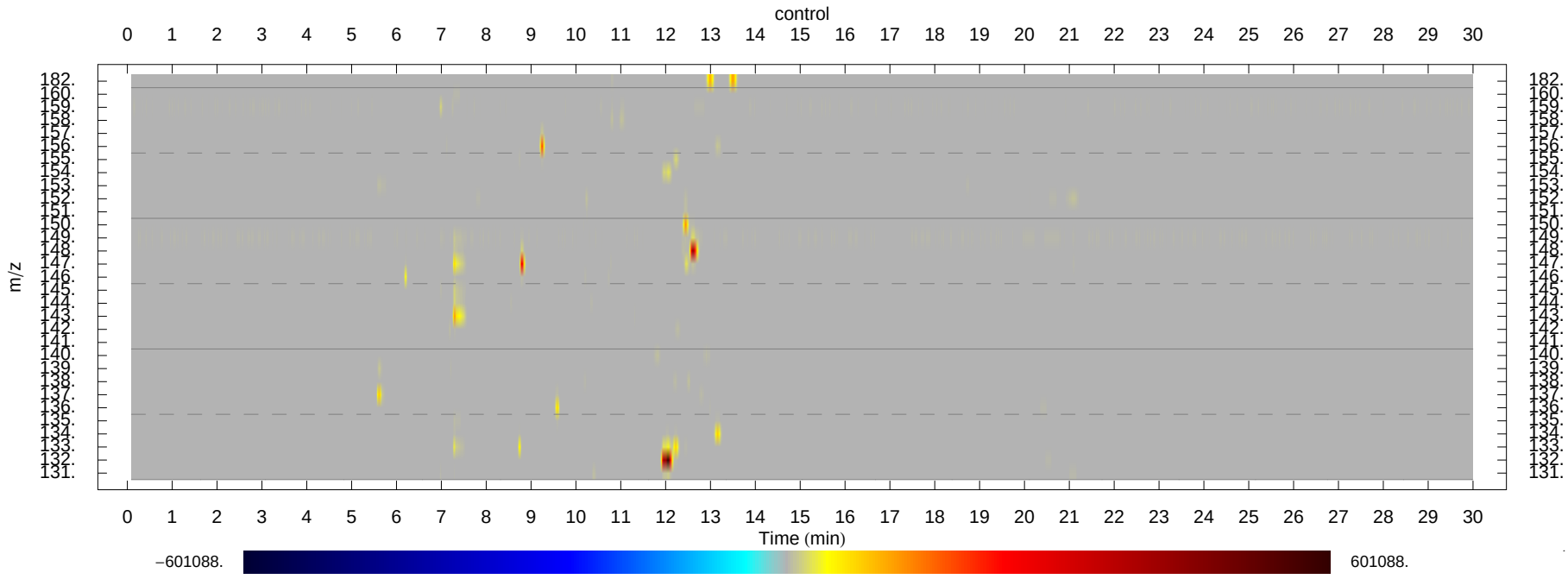
```
DAMPPlotChromatogram[{ctrl, smpl}, 131, PlotOptions → {AspectRatio → .5, ImageSize → 500,
  PlotLabel → "Overlaid electropherograms", PlotStyle → {Hue[1/12], Hue[1/3, 1, .7]}, TextStyle → {FontFamily → "Helvetica", FontSize → 10}}];
```



Plotting the Datasets on Density Plots

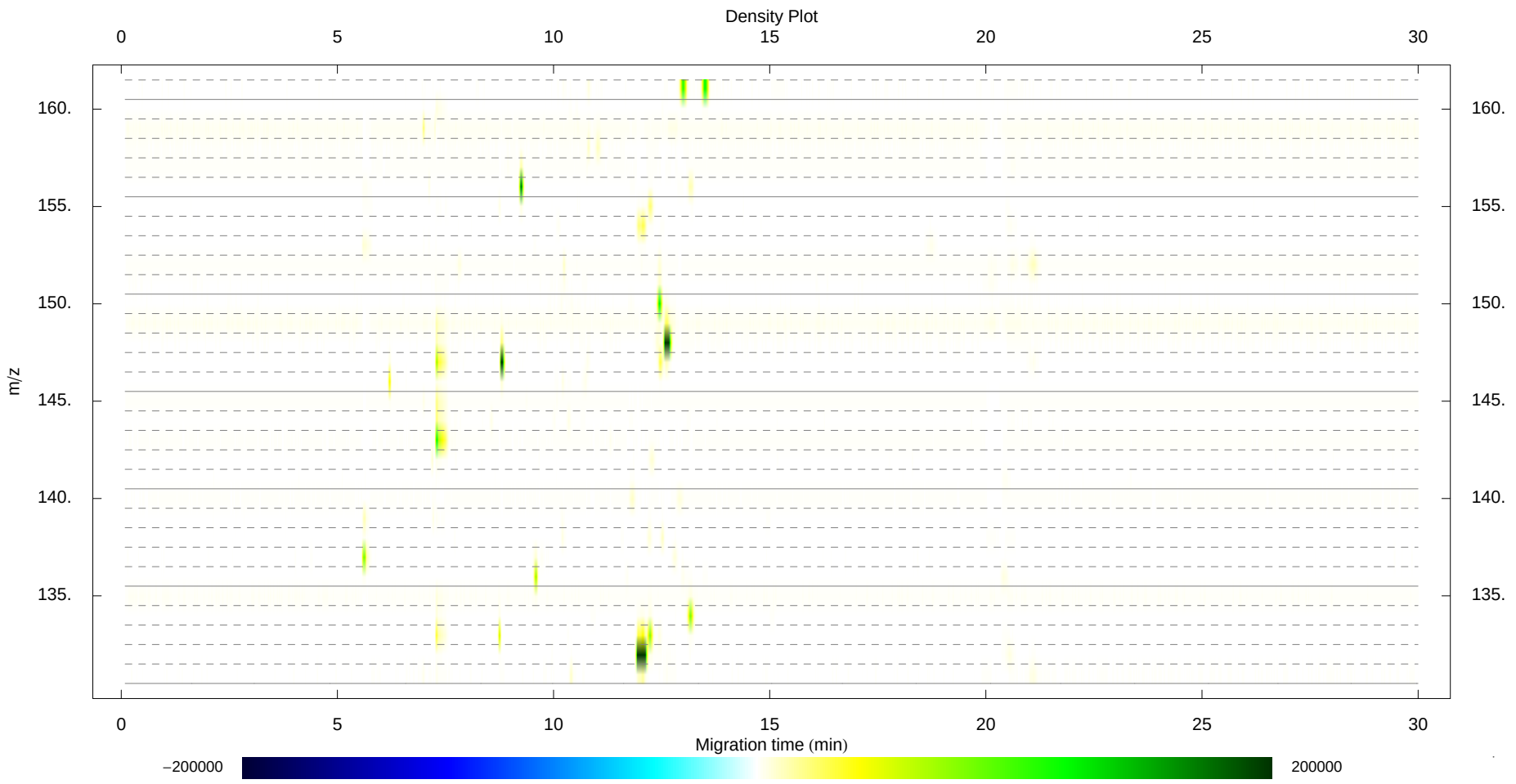
The raw datasets may be plotted on density plots using the `DAMPDensityPlot` function. The axes represent the retention/migration time and m/z values. The peaks appear as colored bands or spots.

```
DAMPDensityPlot[ctrl];
```



The appearance of the density plot may be modified by options.

```
DAMPDensityPlot[ctrl, MaxScale → 200000, FrameTickFreqs → {5, 5}, FrameTickOffsets → {0, 4}, Palette →
   DAMPGradientPalette[ColorPositions → {.2, .6}, PositiveColors → {1 / 6, 1 / 3}, BackgroundGrayLevel → 1], mzGridLineFreq → 1, mzGridLineStyle →
  Join[{{AbsoluteThickness[.25], GrayLevel[.5]}}, Table[{AbsoluteThickness[.25], GrayLevel[.5], Dashing[{.005, .005}]}, {4}]],
 PlotOptions → {PlotLabel → "Density Plot", FrameLabel → {"Migration time (min)", "m/z"}, AspectRatio → .5}];
```



To list the available options, execute `?DAMPDensityPlot`. The plots may be further annotated to allow easier identification of the peaks. Plot annotations will be demonstrated further below.

```
?DAMPDensityPlot
```

```
DAMPDensityPlot[msdata,options] plots the msdata using the ListDensityPlot function. A gradient palette
  is used for representing the signal intensities and the plot may be annotated to allow easier identification of peaks.
Options:
MaxScale – determines the extent of the signal intensity scale (default: Automatic)
LogScale – determines whether the signal intensities should be displayed using a logarithmic scale (default: False)
FrameTickFreqs – frequencies at which tickmarks are placed on the time axis (frequency
  in minutes) and on the m/z axis (in terms of the number of elements in the msdata's mz list) (default: {1,1})
FrameTickOffsets – determines the positions with respect to the origin of msdata's dimensions
  at which the tickmarks are started to be placed (default: {0,0})
mzTickShift – shift of tickmark positioning on the m/z axis (default: −0.5)
mzFrameTicks – a list of custom frame tickmarks to be shown on the m/z axis (default: Automatic)
Palette – list of color specifications to be used for representing the signal intensity values (default: DAMPGradientPalette[])
mzGridLineFreq – frequency of horizontal gridlines in terms of the number of elements in the
  msdata's m/z value list. Enter a list of values to place the gridlines at certain specified positions (default: 5)
mzGridLineStyle – style options for horizontal gridlines (default: {{AbsoluteThickness[0.25],
  GrayLevel[0.5]},{AbsoluteThickness[0.25],GrayLevel[0.5],Dashing[{0.01,0.01}]}})
AnnotationTables – list of annotation tables to be overlaid on the plot (default: None)
PlotOptions – options for the ListDensityPlot function (default:
  {Mesh->False,ImageSize->930,AspectRatio->0.35,FrameLabel->{"Time (min)","m/z"},TextStyle->DAMPTextStyle})
AnnotationOptions – lists of options for the DAMPDrawAnnotation function for each annotation table. The number
  of lists of options must correspond to the number of passed annotation tables (default: Automatic)
```

■ Dataset cropping and range selection

The functions `DAMPCrop`, `DAMPSelectMZs`, and `DAMPDropMZs` can be used to select parts of interest from the datasets.

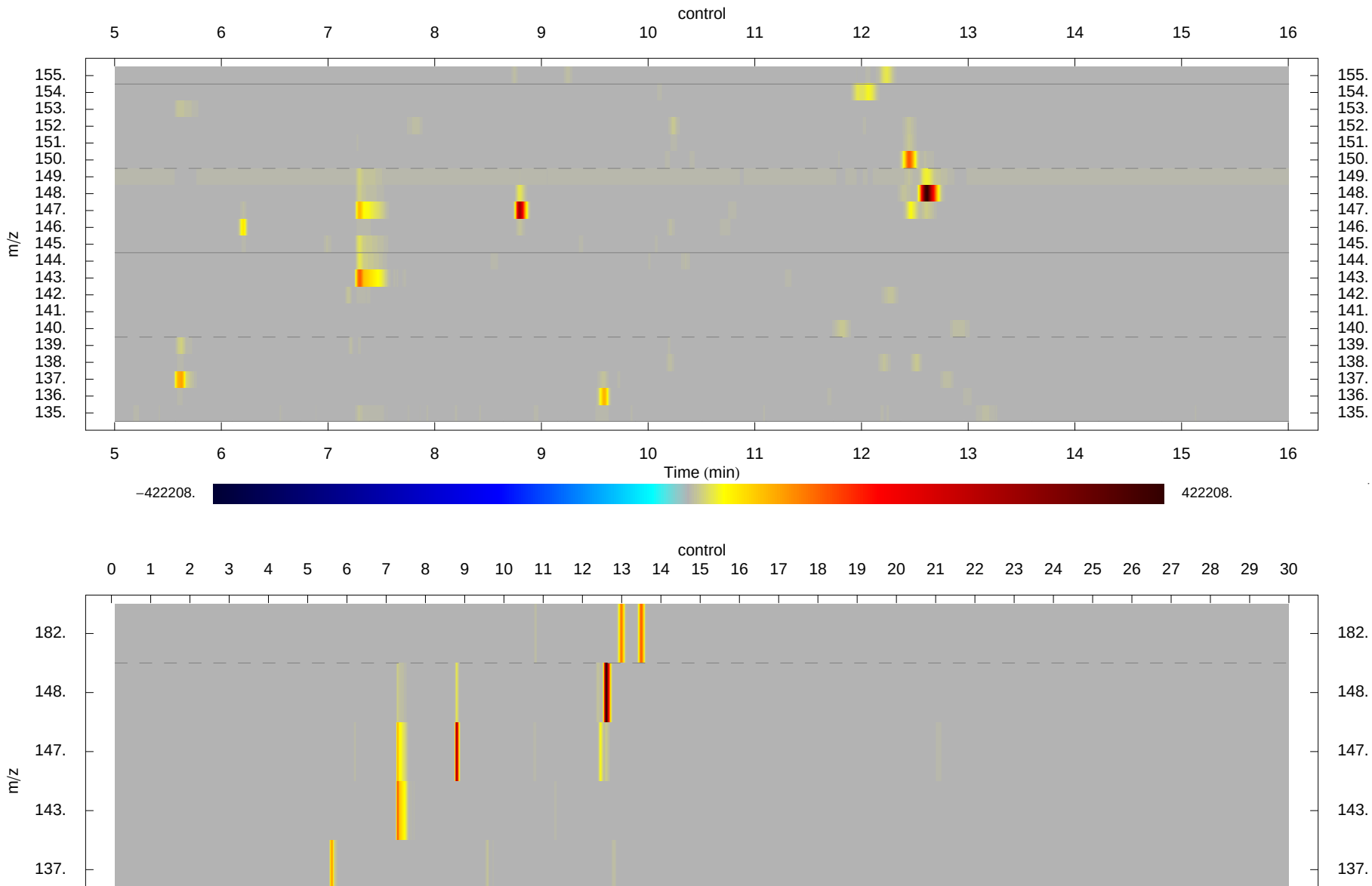
```
?DAMPCrop
?DAMPSelectMZs
?DAMPDropMZs
```

DAMPCrop[msdata,options] reduces the msdata dataset to datapoints falling within the timerange and m/z value range specified by options:
mzRange – two element list specifying the cropping m/z value range (default: All)
TimeRange – two element list specifying the cropping time range (default: All).

DAMPSelectMZs[msdata,mzs] reduces the msdata datased to datapoints corresponding to m/z values specified in the mzs list.

DAMPDropMZs[msdata,mzs] reduces the msdata datased by eliminating datapoints corresponding to m/z values specified in the mzs list.

```
DAMPDensityPlot[DAMPCrop[ctrl, mzRange → {135, 155}, TimeRange → {5, 16}]];
DAMPDensityPlot[DAMPSelectMZs[ctrl, {136, 137, 143, 147, 148, 182}]];
DAMPDensityPlot[DAMPDropMZs[ctrl, {135, 145, 155}]];
```



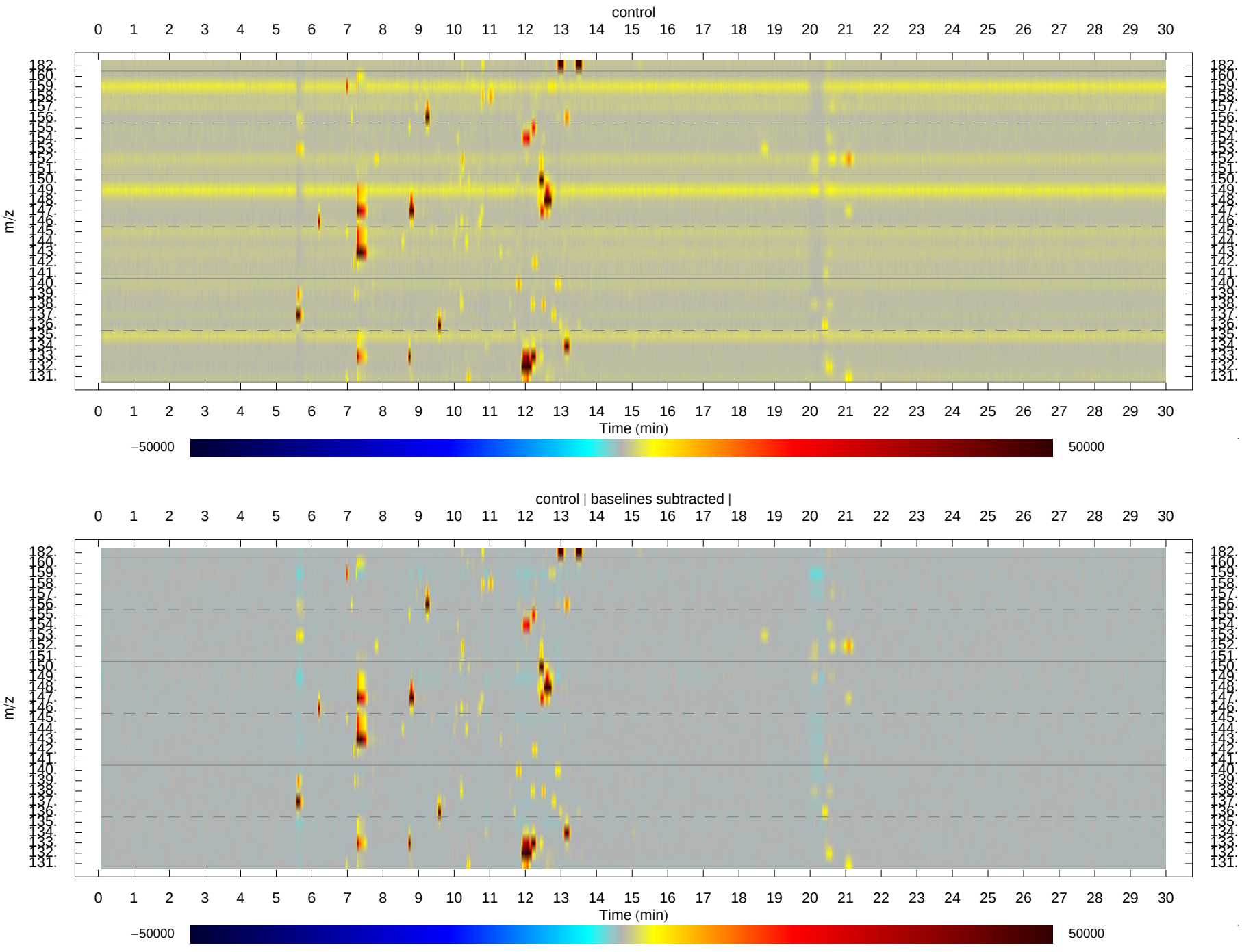
■ Baseline subtraction, noise removal, and smoothing

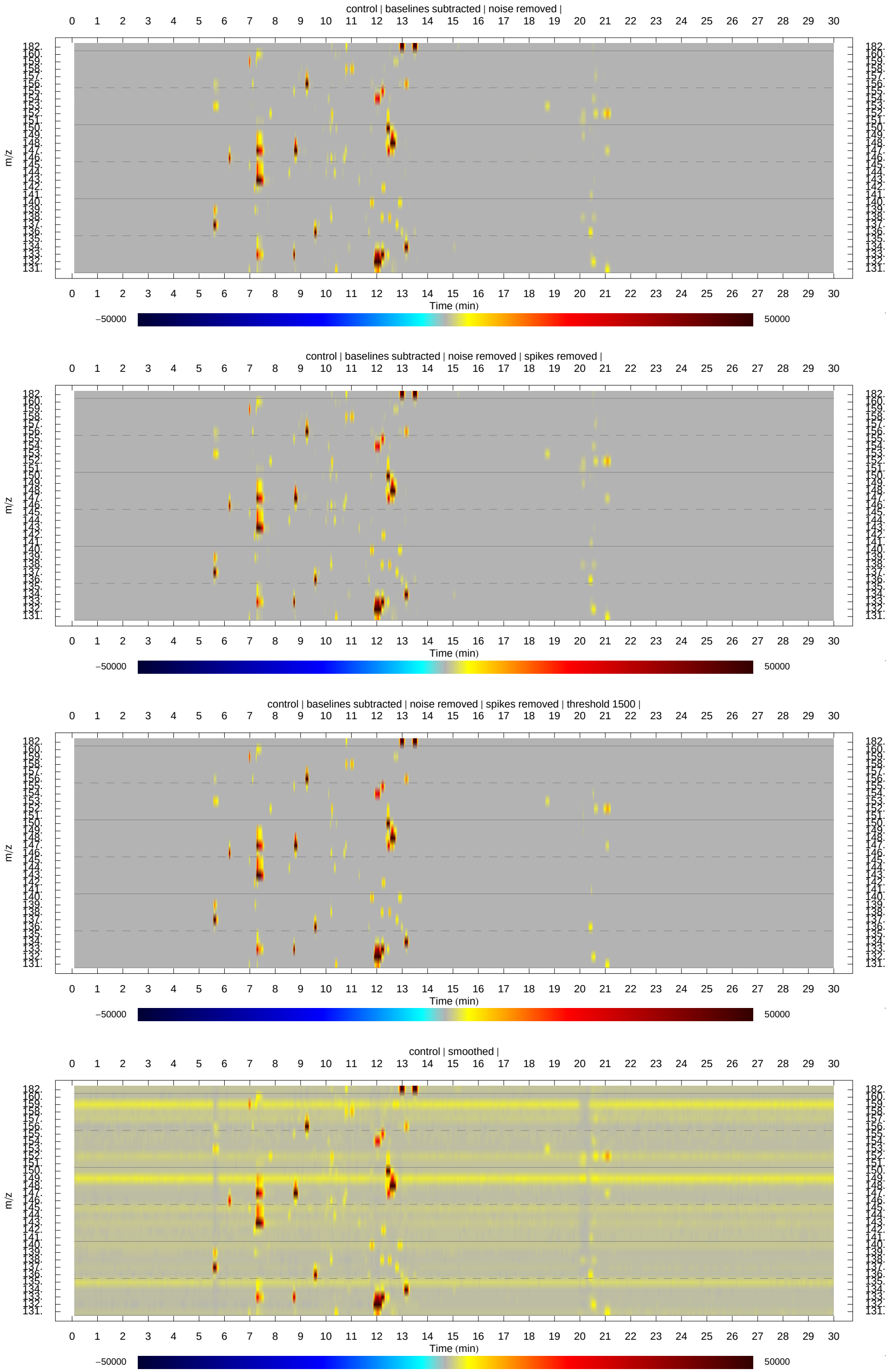
DAMPSubtractBaselines performs baseline subtraction from all chromatograms/electropherograms in a *Math*DAMP dataset. Baselines are fitted to a polynomial by robust nonlinear regression. Execute ?DAMPSubtractBaselines for the description of options, or refer to the MathDAMP.nb notebook for details regarding the implementation. The usage reference information for addi−tional functions related to noise removal and smoothing is shown below.

```
?DAMPSubtractBaselines
?DAMPRemoveNoise
?DAMPRemoveSpikes
?DAMPThreshold
?DAMPSmooth
```

```
DAMPSubtractBaselines[msdata,options] subtracts baselines from all chromatograms/electropherograms in msdata dataset and returns the new processed dataset.
Options:
BaselineFittingFunction - pure function which is expected to return the baseline points
  for every timepoint in a chromatogram/electropherogram passed as an argument (default: DAMPRobustPolynomialFit)
SampleNameSuffix - string to be added to the SampleName from the msdata to keep the track of modifications performed on the dataset (default: "bs")

DAMPRemoveNoise[msdata,options] removes noise from every chromatogram/electropherogram in msdata by leveling to
  0 all signal intensities, absolute values of which are smaller than a threshold. The threshold is calculated as a specific
  multiple of the standard deviation of signal intensities from a specified time range of every chromatogram/electropherogram.
Options:
TimeRange - time range (in minutes) from which to calculate the standard deviation of signal intensities (default: {1,3})
SDThreshold - specifies the multiple of the standard deviation of signal
  intensities from the selected time range to be used as the noise discrimination threshold (default: 5)
LevelNegativeSignals - if set to True, all negative signal intensities in every chromatogram/electropherogram will be leveled to 0 (default: True)
SampleNameSuffix - string to be added to the SampleName from the msdata to keep the track of modifications performed on the dataset (default: "nr")

DAMPRemoveSpikes[msdata,options] levels to 0 all signal intensity values
  both neighbours of which have 0 signal intensity value in every chromatogram/electropherogram in msdata.
Options:
SampleNameSuffix - string to be added to the SampleName from the msdata to keep the track of modifications performed on the dataset (default: "sr")

DAMPThreshold[msdata,threshold,options] levels to 0 all signal intensity values in msdata which are within ±threshold.
Options:
SampleNameSuffix - string to be added to the SampleName from the msdata to keep track of modifications performed on the dataset (default: "t")

DAMPSmooth[msdata,options] applies a smoothing function to all chromatograms/electropherograms in msdata.
Options:
SmoothingFunction - pure function to be applied to the list of signal intensities from a chromatogram/electropherogram (default: DAMPMovingAverageFast[#,9]&)
SampleNameSuffix - string to be added to the SampleName from the msdata to keep track of modifications performed on the dataset (default: "s")
```

```
DAMPSubtractBaselines[ctrl, SampleNameSuffix → " baselines subtracted "];
DAMPRemoveNoise[%, SampleNameSuffix → " noise removed "];
DAMPRemoveSpikes[%, SampleNameSuffix → " spikes removed "];
DAMPThreshold[%, 1500, SampleNameSuffix → " threshold 1500 "];
DAMPSmooth[ctrl, SmoothingFunction → (DAMPMovingAverageFast[#, 20] &), SampleNameSuffix → " smoothed "];
DAMPDensityPlot[#, MaxScale → 50000] & /@ {ctrl, %%%%%, %%%%, %%%, %%, %};
```





The datasets will be preprocessed for subsequent use by subtracting the baselines and by noise removal (with default options).

```
{ppctrl, ppsmpl} = DAMPRemoveNoise[DAMPSubtractBaselines[#]] & /@ {ctrl, smpl};
```

■ Peak picking

The `DAMPPickPeaks` function finds peaks in a *Math*DAMP dataset. Currently, the peak lists are used for the sole purpose of dataset alignmnet. Given the robustness of the alignment method, the peak lists do not have to contain all peaks from every dataset and may contain erroneously picked peaks as well (noise or jumping baseline related). The requirements on the quality of peak picking results are therefore not high and the algorithm is rather simple. For details regarding the implementation, please refer to the MathDAMP.nb notebook. However, the returned peak lists seem to be quite accurate (as can be seen on the next density plot below).

```
?DAMPPickPeaks
```

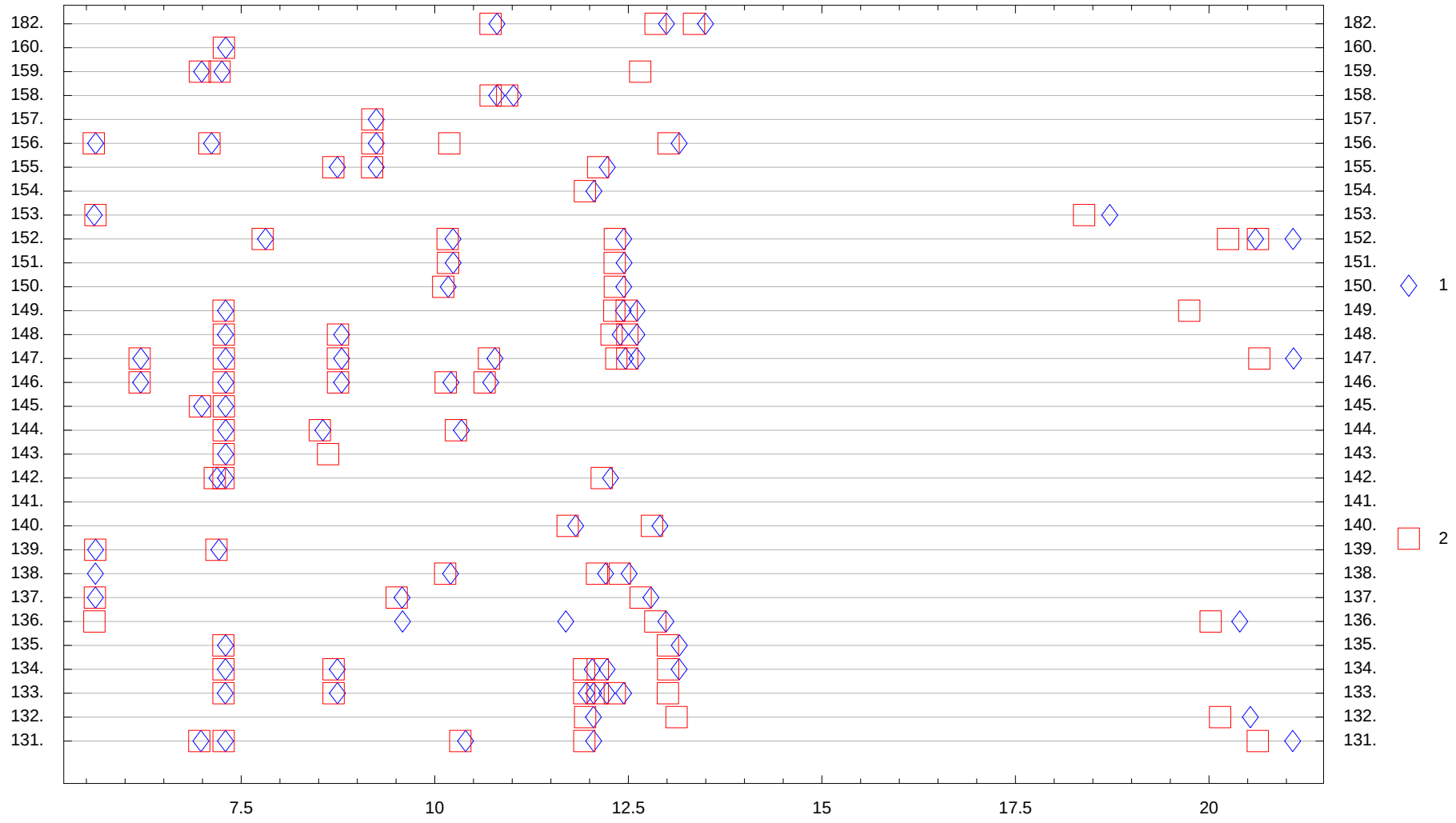
```
DAMPPickPeaks[msdata,options] picks peaks from all chromatograms/electropherograms in msdata and returns them in a list {{m/z value,peaklist},
  {m/z value,peaklist},...}. The function does not have any default options, the options passed to the DAMPPickPeaks function are passed
  further to the DAMPPickChromatogramPeaks function which is used internally to pick peaks from individual chromatograms/electropherograms.
```

```
peaklists = DAMPPickPeaks[#, Threshold → 2000] & /@ {ppctrl, ppsmpl};
TableForm[Take[peaklists[[1]], 5], TableDepth → 1]
```

```
{131., {{6.97967, 2586.76}, {7.29654, 2268.56}, {10.3982, 7586.41}, {12.0525, 14725.9}, {21.0809, 4195.62}}}
{132., {{12.0491, 600509.}, {20.5342, 4717.08}}}
{133., {{7.29417, 33397.4}, {8.74295, 58108.2}, {11.9588, 30033.4}, {12.0526, 40593.3}, {12.2233, 78201.2}, {12.4397, 4673.98}}}
{134., {{7.29557, 5466.44}, {8.74117, 3078.31}, {12.0338, 2902.04}, {12.2261, 4072.02}, {13.1565, 81834.9}}}
{135., {{7.2983, 4707.63}, {13.1584, 3588.7}}}
```

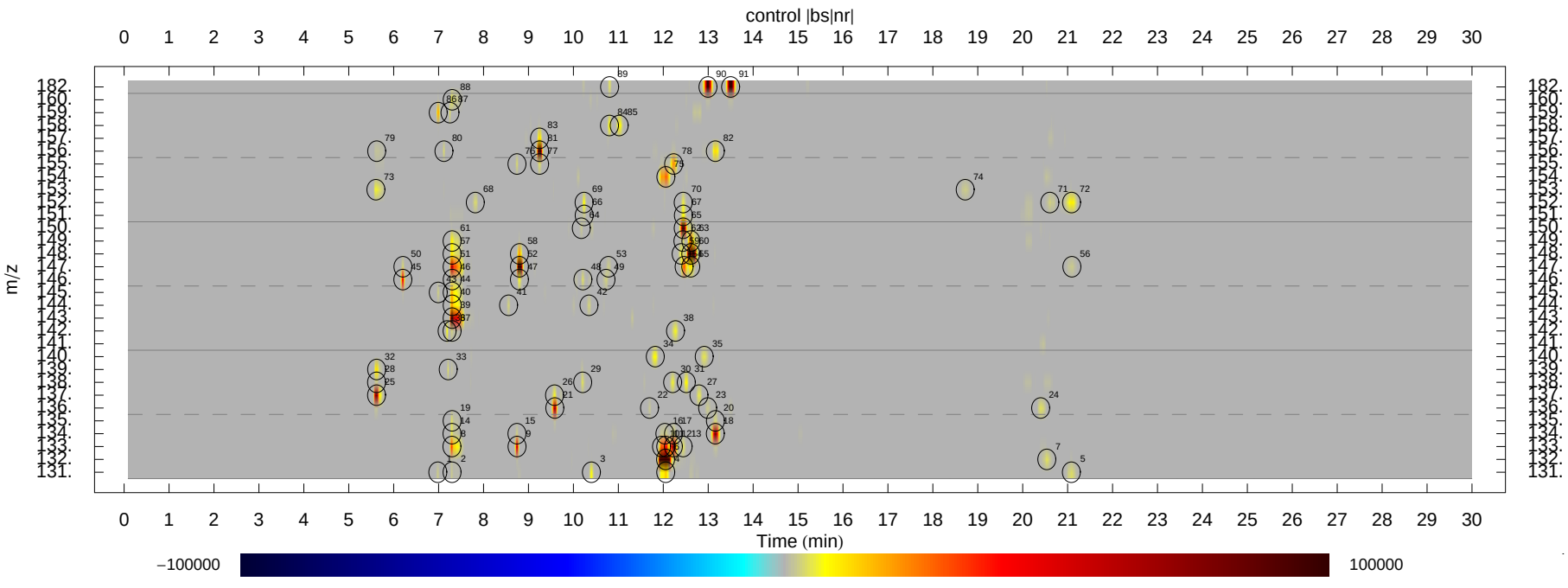
Peaks found in the first 5 electropherograms from the `ctrl` dataset are listed above. The layouts of picked peaks may be displayed on peak layout plots for comparison or for visual inspection of the alignment quality.

```
DAMPPlotPeakLayout[peaklists];
```



The peak lists may also be converted to an annotation table format for display on chromatograms or on density plots.

```
DAMPDensityPlot[ppctrl, MaxScale → 100000, AnnotationTables → {DAMPPeakListToAnnotationTable[peaklists[[1]]]};
```

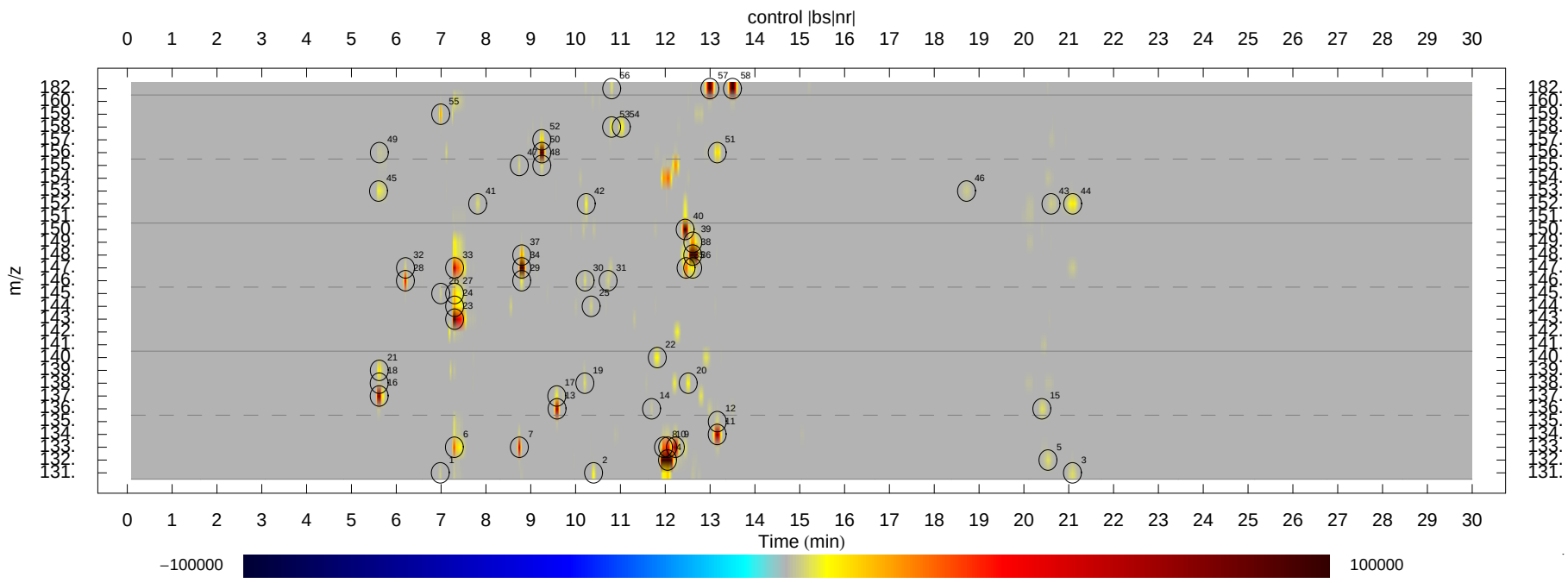


Sometimes, a large number of peaks may be erroneously picked from a single chromatogram/electropherogram (noise above the peak−picking threshold) or an overwhelming number of redundant signals is present along the m/z dimension at a certain retention/migration time. The presence of these signals may bias the alignment and it could be desirable to select a representative set of peaks from a peak list. Also, the alignment procedure needs more running time for peak lists with large numbers of peaks. The function `DAMPSelectRepresentativePeaks` performs the representa−tive peak selection.

```
?DAMPSelectRepresentativePeaks
```

```
DAMPSelectRepresentativePeaks[peaklist,options] reduces the peaklist returned by the DAMPPickPeaks function to a selected number of
  highest peaks in every chromatogram/electropherogram and a selected number of highest peaks in every time interval of selected size.
Options:
PeaksPerChromatogram - number of highest peaks to select from every chromatogram/electropherogram (default: All)
PeaksPerInterval - number of highest peaks to select from every time interval (default: All)
IntervalSize - size of the time interval (in minutes) for selection of highest peaks when PeaksPerInterval is not set to All (default: 1)
TimeRange - select peaks from this retention/migration time range only (default: All)
```

```
DAMPDensityPlot[ppctrl, MaxScale → 100000, AnnotationTables → {DAMPPeakListToAnnotationTable[
    DAMPSelectRepresentativePeaks[peaklists[[1]], PeaksPerChromatogram → 5, PeaksPerInterval → 5, IntervalSize → .5]]}];
```



■ Annotation table manipulation

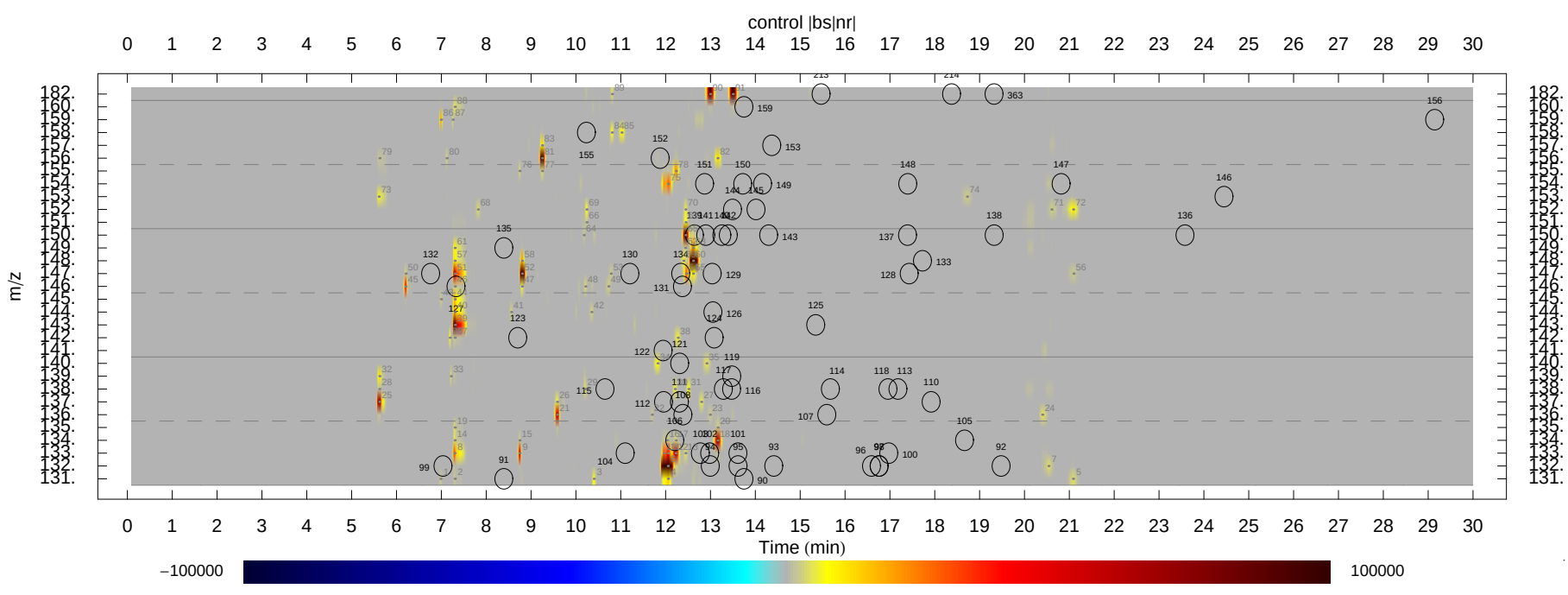
Annotation tables are intended to facilitate easier identification of peaks. The annotation table may be constructed according to an analysis of a mixture of standard compounds. The table is expected to consist of 5 columns: m/z, retention/migration time, short compound name/id, full compound name, and relative position of the text with respect to the label on the density plots (1 − right, 2 − top, 3 − left, 4 − bottom, 1.5 − top right, etc).

```
annottab = DAMPLoadAnnotationTable[MathDAMPPath <> "/iab_cems_cation.csv"];
TableForm[Take[%, 5]]
```

71.0637	10.351	1	3−Aminopropionitrile monofumarate salt	4
75.0944	7.137	2	1,3−Diaminopropane dihydrochloride	4
76.0425	13.51	3	Glycine	2
76.0537	10.898	4	Semicarbazide hydrochloride	2
76.0793	11.41	5	Isopropanolamine	4

Multiple annotation tables can easily be displayed on a single density plot. Their appearance may be modified as well. The following example shows the above loaded (and unaligned) annotation table along with the labels for picked peaks (shown as gray dots). The alignment of annotation tables to *Math*DAMP datasets is demonstrated in the next section.

```
DAMPDensityPlot[ppctrl, MaxScale → 100000, AnnotationTables → {annottab, DAMPPeakListToAnnotationTable[peaklists[[1]]]},
  AnnotationOptions → {{}, {LabelShape → (Point[#1] &), LabelSize → {.1, .1},
     LabelStyle → {GrayLevel[.5], AbsolutePointSize[1]}, TextStyle → {FontColor → GrayLevel[.5]}}}];
```



■ Dataset alignment

To align two datasets, parameters of a (custom) function describing the time shifts of corresponding peaks in two datasets are optimized. A combination of global optimization and dynamic programming is used for this purpose. The function is then used to rescale the timescale on one of the datasets, interpolate the chromatograms/electropherograms, and timepoints identical to timepoints in the reference datasets are selected. For details regarding the dataset alignment procedure, please refer to the MathDAMP.nb notebook.
`DAMPFitShiftFunction` performs the parameter optimization for the retention/migration time shift function.

```
?DAMPFitShiftFunction
```

```
DAMPFitShiftFunction[peaklist1,peaklist2,options] optimizes the parameters of a retention/migration
  time shift function so that when applied to peaklist2 the optimum peak alignment (as measured by DAMPDPScore) is achieved.
Options:
ShiftFunction - pure function to be used as a retention/migration time shift function (default: (1/(1/(α #)+γ/2)&))
ShiftFunctionParameters - list of parameters for optimization. If Automatic is specified, these are extracted automatically
  from ShiftFunction. The parameters may be also specified explicitly with seek ranges (default: {{α,0.8,1.2},{γ,-0.04,0.04}})
GapPenalty - gap penalty value to be passed to the DAMPDPScore function for the scoring of the alignment. A list of gap penalty
  values may be passed to perform the fitting of the retention/migration time shift function iteratively (default: {3,0.5})
NMinimizeOptions - list of options to be passed to the NMinimize function used for optimization (default: {MaxIterations->1000})
TimeRange - specifies the selection time range of peaks from peaklist1 to be used for scoring (default: {0,∞})
```

```
fsrslt = DAMPFitShiftFunction[Sequence @@
   (DAMPSelectRepresentativePeaks[#, PeaksPerChromatogram → 5, PeaksPerInterval → 5, IntervalSize → .5] & /@ peaklists), GapPenalty → {4, .5}]
```

$$\left\{\text{Score} \to 8.38966,\ \text{BestFitFunc} \to \left(\frac{1}{\frac{1}{0.99447\,\#1} - \frac{0.00245727}{2}}\ \&\right),\ \text{BestFitPars} \to \{\alpha \to 0.99447,\ \gamma \to -0.00245727\}\right\}$$

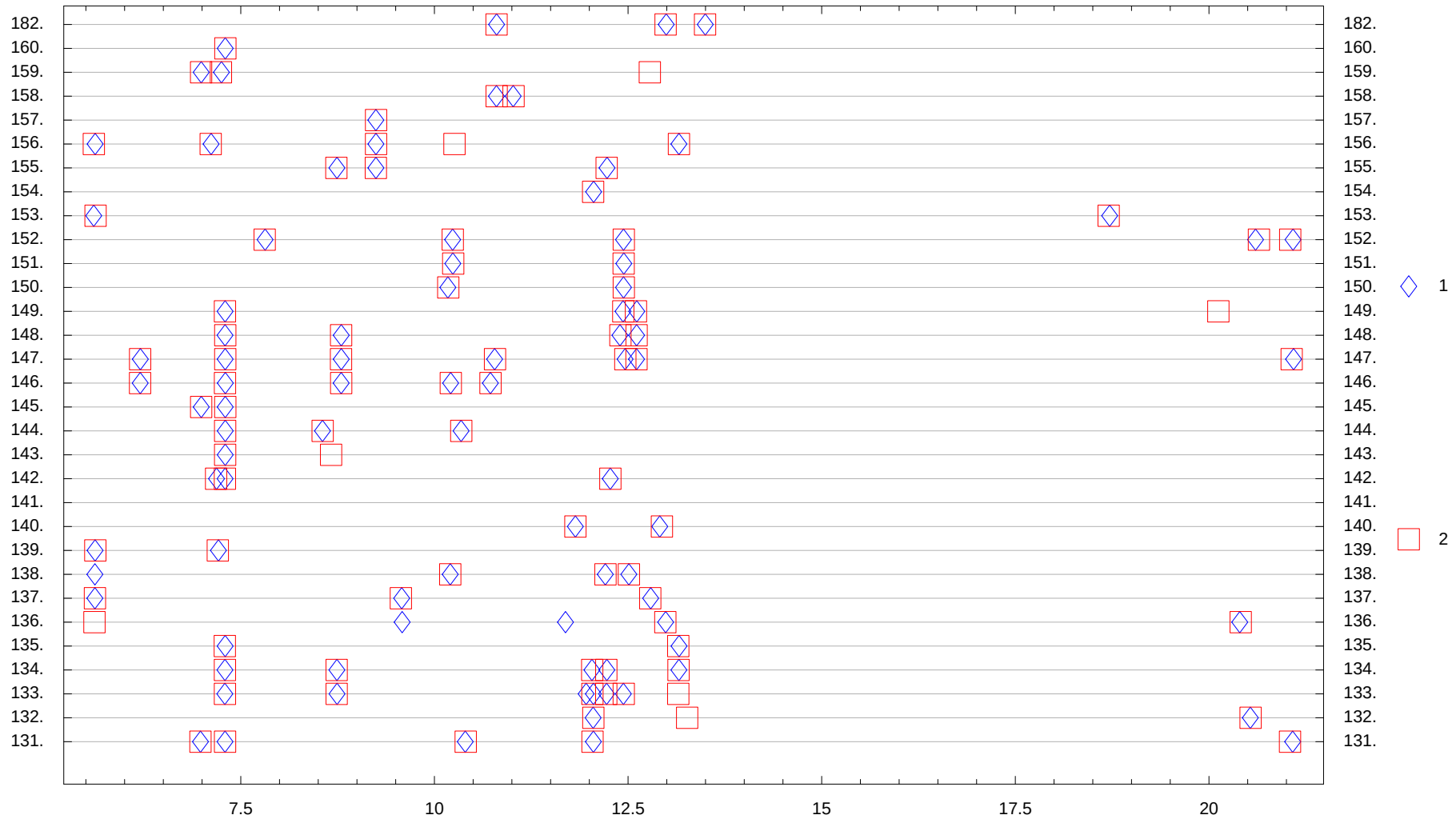
Only a subset of peaks (selected by the `DAMPSelectRepresentativePeaks`) was used to achieve faster alignment. A function derived by Reijenga *et al.* (see the MathDAMP.nb notebook for details) is used by default as a retention/migration time shift function (due to the predominant use of capillary electrophoresis based techniques in the authors' institution − Institute for Advanced Biosciences, Keio University). Any function may be passed to `DAMPFitShiftFunction` as a retention/migration time shift function as demonstrated below with a second order polynomial.

```
DAMPFitShiftFunction[
 Sequence @@ (DAMPSelectRepresentativePeaks[#, PeaksPerChromatogram → 5, PeaksPerInterval → 5, IntervalSize → .5] & /@ peaklists),
 ShiftFunction → (# + a + b # + c #^2 &), ShiftFunctionParameters → Automatic, GapPenalty → {4, .5}]
```

$$\{\text{Score} \to 8.38822,\ \text{BestFitFunc} \to (\#1 + 0.0140676 - 0.00849169\,\#1 + 0.00138294\,\#1^2\ \&),\ \text{BestFitPars} \to \{a \to 0.0140676,\ b \to -0.00849169,\ c \to 0.00138294\}\}$$

The peak layouts may be shown for visual confirmation of the alignment (alignment done with the default time shift function). The layout of peaks prior to the alignment is shown in the previous section.

```
DAMPPlotPeakLayout[{peaklists[[1]], DAMPAlignPeakList[peaklists[[2]], BestFitFunc /. fsrslt]}];
```

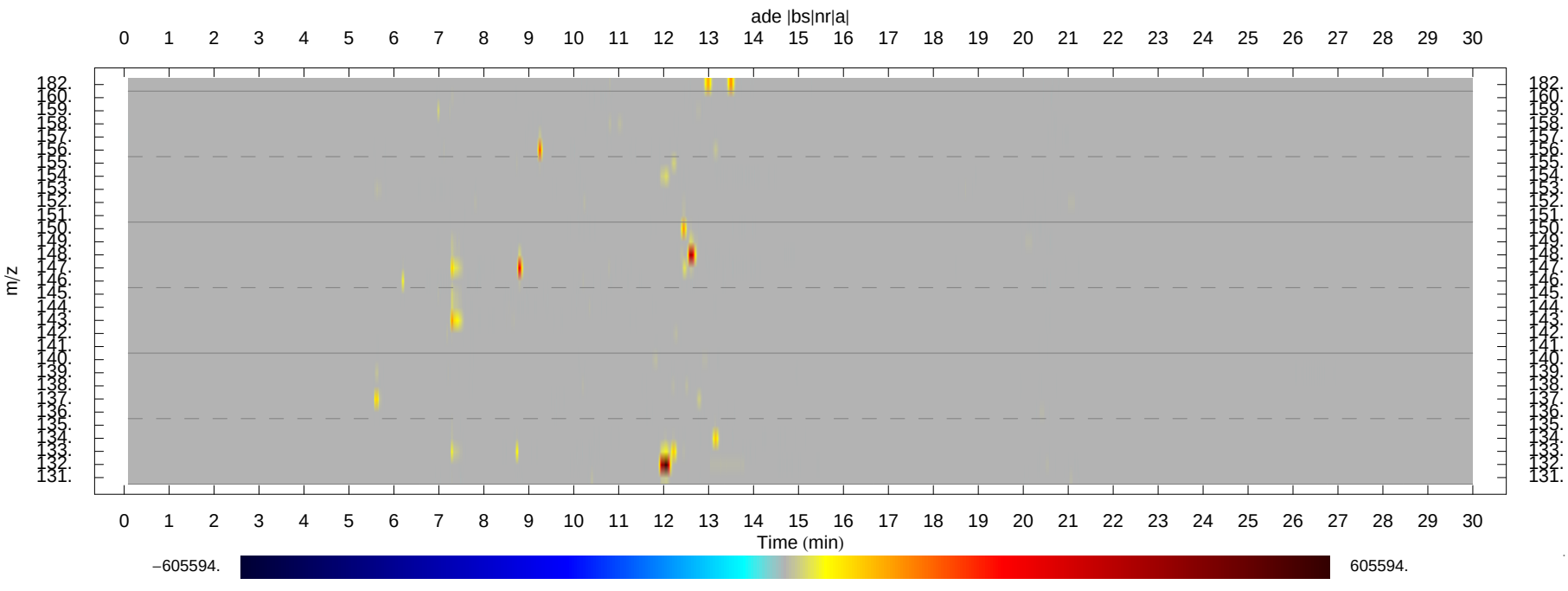


After finding the time shift function, the aligned dataset is created using the function `DAMPAlign`.

```
?DAMPAlign
```

```
DAMPAlign[msdata,shiftfunction,timepoints,options] aligns msdata according to timeshift
  function shiftfunction and selects (by interpolation) time points identical to the ones in the timepoints list.
Options:
SampleNameSuffix - string to be added to the SampleName from the msdata to keep track of modifications performed on the dataset (default: "a")
```

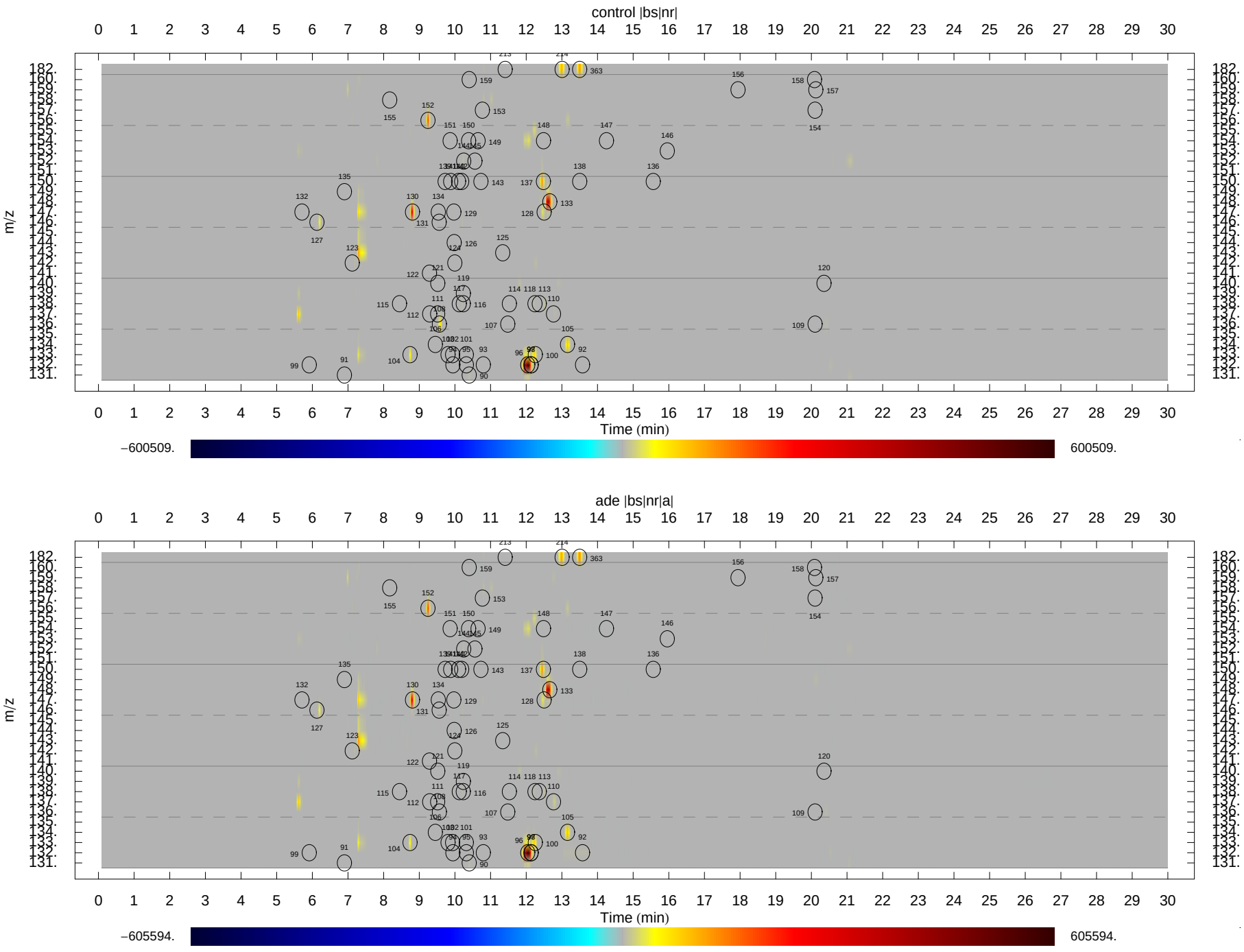
```
alignedsmpl = DAMPAlign[ppsmpl, BestFitFunc /. fsrslt, ctrl[[3]]];
DAMPDensityPlot[alignedsmpl];
```



Annotation tables may be aligned in a similar fashion. The step below demonstrates the robustness of the alignment procedure. Even a relatively small number of corresponding peaks is sufficient for finding the optimal alignment. The timeshifts of unaligned annotation labels were quite significant (over 5 min) when the density plots below are compared to the one at the end of the previous section.

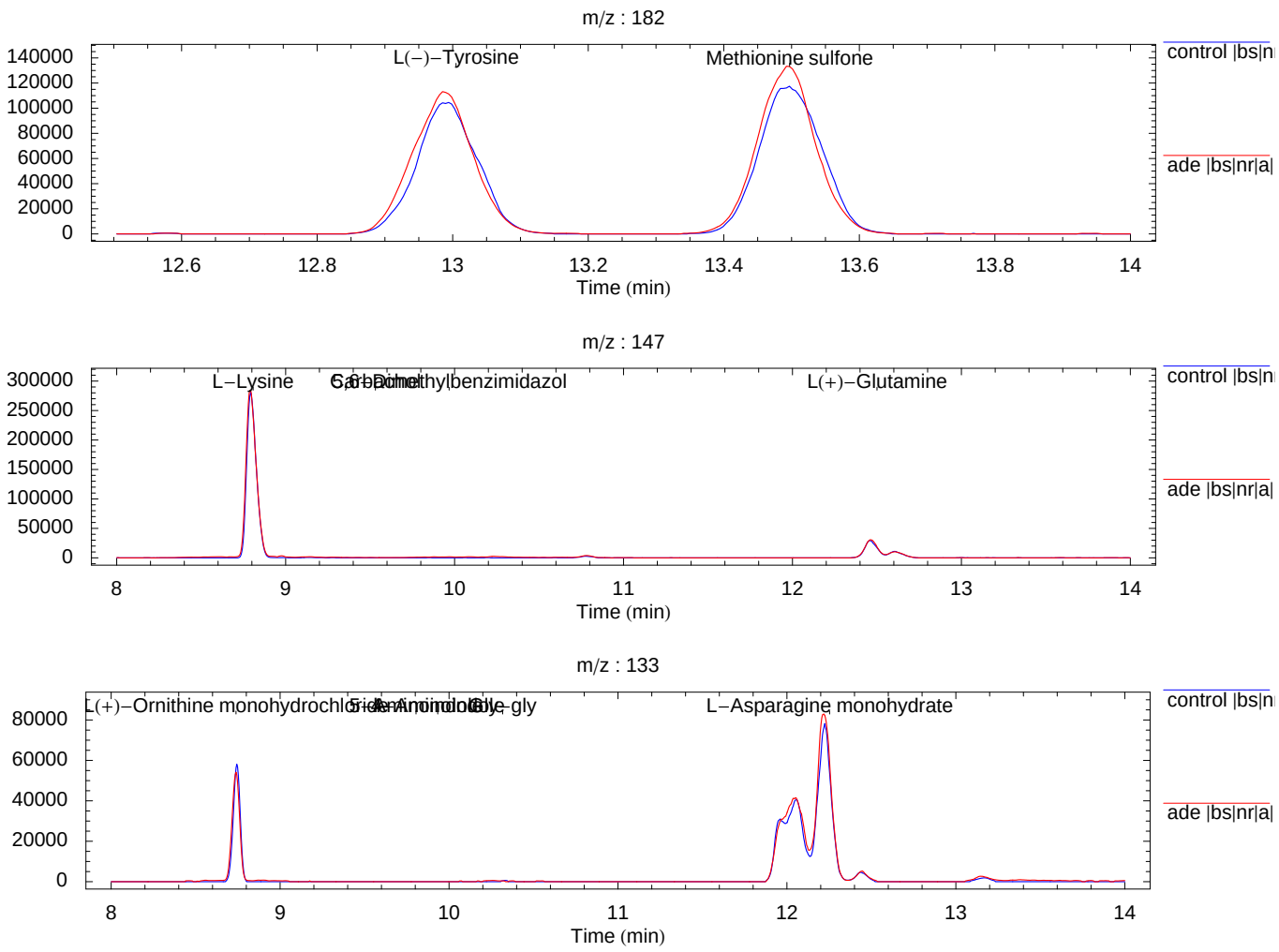
```
?DAMPAlignAnnotationTable

DAMPAlignAnnotationTable[peaklist,annotationtable,options] aligns annotationtable to peaklist and returns the new annotation
  table. Options for DAMPFitShiftFunction, which is used internally, may be passed directly via options. Additional option Resolution
  determines the rounding of m/z values in the annotation table. If the resulting annotation table is intended to be used on a density
  plot where the underlying data were binned to 0.1 m/z resolution, the resolution option should be set to 0.1 as well (default: 1)
```

```
alignedannottab = DAMPAlignAnnotationTable[
   DAMPSelectRepresentativePeaks[peaklists[[1]], PeaksPerChromatogram → 5, PeaksPerInterval → 5, IntervalSize → .5], annottab];
DAMPDensityPlot[#, AnnotationTables → {alignedannottab}] & /@ {ppctrl, alignedsmpl};
```



Plots of chromatograms/electropherograms may be annotated as well. Full compound names are used instead of short names/ids in this case.

```
DAMPPlotChromatogram[{ppctrl, alignedsmpl}, #[[1]], AnnotationTable → alignedannottab, PlotOptions → {PlotRange → {#[[2]], All}}] & /@
  {{182, {12.5, 14}}, {147, {8, 14}}, {133, {8, 14}}};
```

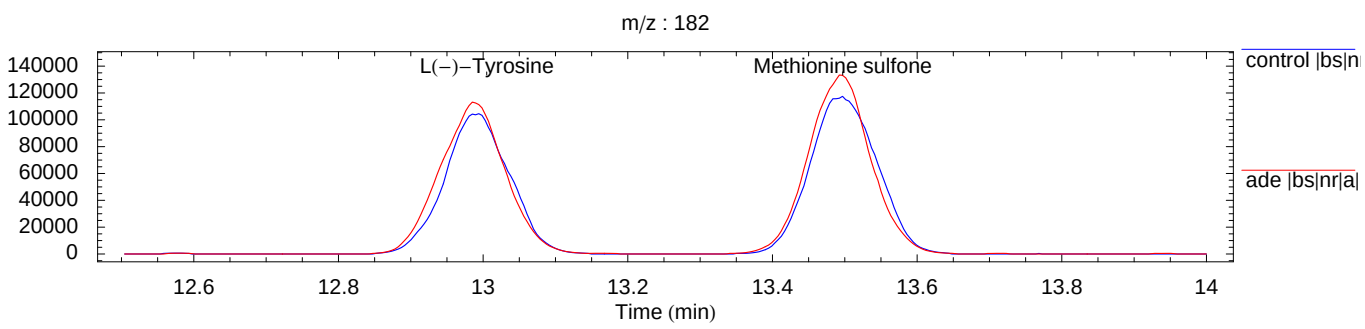


`DAMPNormalizeGroup` function aligns and normalizes multiple datasets to a selected reference dataset. This function assembles the steps described in this section along with an intensity normalization step described in the next section. The `DAMPNormalizeGroup` function is used by the functions for common types of differential analysis of metabolite profiles demonstrated in the notebooks 03−MathDAMP−TwoDatasets.nb, 04−MathDAMP−Outliers.nb, 05−MathDAMP−TwoGroups, and 06−MathDAMP−MultipleGroups.nb.

■ Dataset normalization

Often the datasets' signal intensity values have to be normalized according to the peak of the internal standard. *Math*DAMP implements very simple peak integration functionality. A specified range of a chromatogram/electropherogram is integrated blindly. When normalizing multiple datasets using the `DAMPNormalizeGroup` function (mentioned at the end of the previous section), the location of the peak of the internal standard in the reference dataset has to be either specified explicitly or can be extrapolated from the aligned annotation table. In the latter case, only the short name/id of the internal standard is specified and the peak is located automatically. For more details about the `DAMPNormalizeGroup` function, please refer to the MathDAMP.nb notebook and the 03−MathDAMP−TwoDatasets.nb, 04−MathDAMP−Outliers.nb, 05−MathDAMP−TwoGroups, and 06−MathDAMP−MultipleGroups.nb notebooks.
Signal intensity normalization of the `alignedsmpl` dataset to the `ppctrl` dataset according to the area of the Methioninesulfone peak is shown below. The location of the peak is specified explicitly.

```
DAMPPlotChromatogram[{ppctrl, alignedsmpl}, 182, AnnotationTable → alignedannottab, PlotOptions → {PlotRange → {{12.5, 14}, All}}];
```




```
?DAMPIntegrate
```

```
DAMPIntegrate[chromatogram,options] calculates the area below the signal intensities of a chromatogram/electropherogram within the retention/migration
  time range specified by the option TimeRange (default: {0,∞}). Baseline may be calculated as an average of signal intensities within the
  timerange specified by the options BaselineFromTimeRange (default: None). If set to None, baseline is set to signal intensity value 0.
```

```
DAMPIntegrate[DAMPGetChromatogram[#, 182], TimeRange → {13.3, 13.8}] & /@ {ppctrl, alignedsmpl}
normcoefs = %[[1]] / %
```

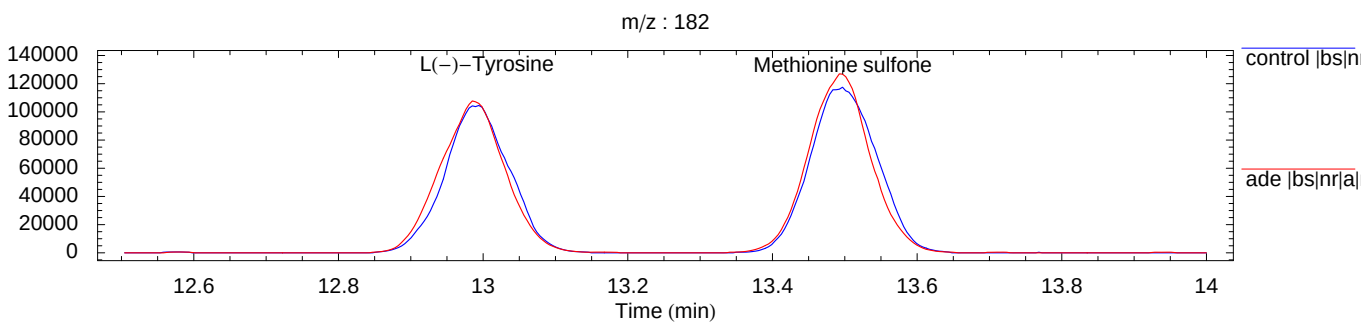
```
{12793.3, 13437.9}
```

```
{1., 0.952029}
```

```
?DAMPNormalize
```

```
DAMPNormalize[msdata,coefficient,options] multiplies the signal intensities in msdata (msdata[[1]]) with coefficient.
Options:
SampleNameSuffix – string to be added to the SampleName from the msdata to keep the track of modifications performed on the dataset (default: "n")
```

```
normasmpl = DAMPNormalize[alignedsmpl, normcoefs[[2]]];
DAMPPlotChromatogram[{ppctrl, normasmpl}, 182, AnnotationTable → alignedannottab, PlotOptions → {PlotRange → {{12.5, 14}, All}}];
```



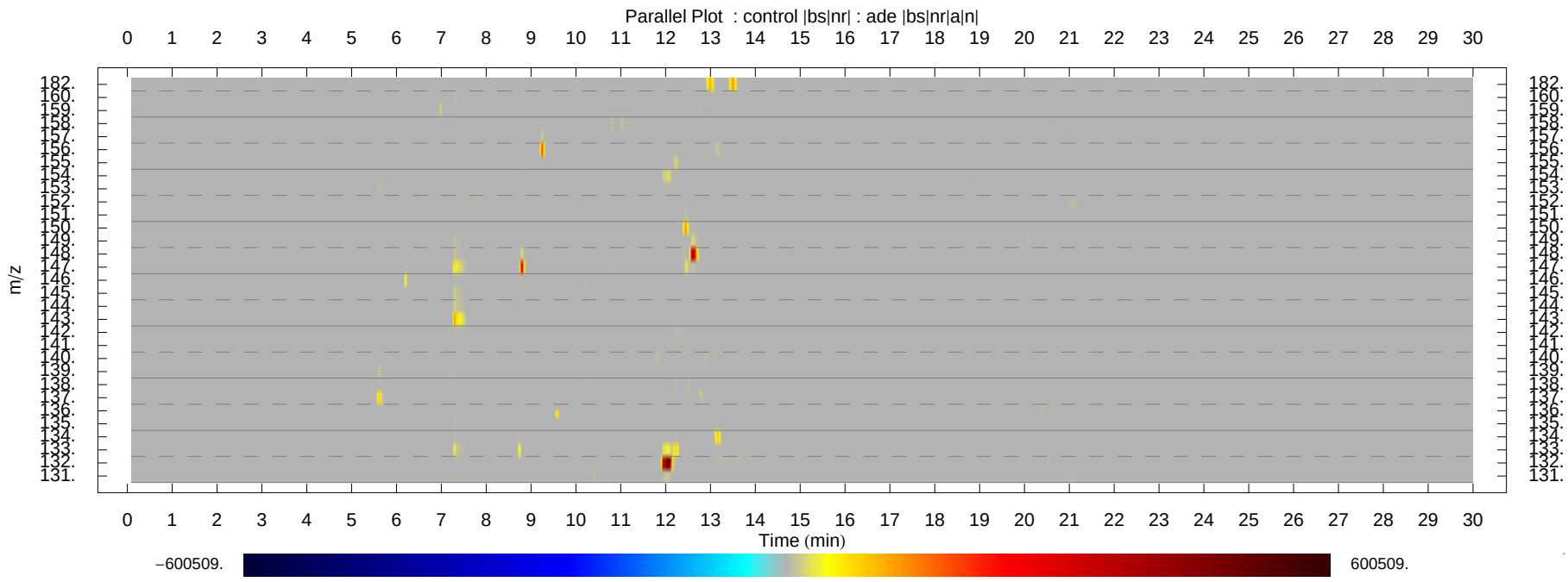
■ Comparing normalized datasets

One way to compare the normalized datasets is to interlace their chromatograms/electropherograms into each other and plot the resulting dataset on a parallel plot. Here, the electropherograms from the datasets corresponding to identical m/z values are plotted next to each other. Differences would appear as half−bands (like for m/z 136 at about 9.5 min or for m/z 137 between 12 and 13 min).

```
?DAMPParallelPlot
```

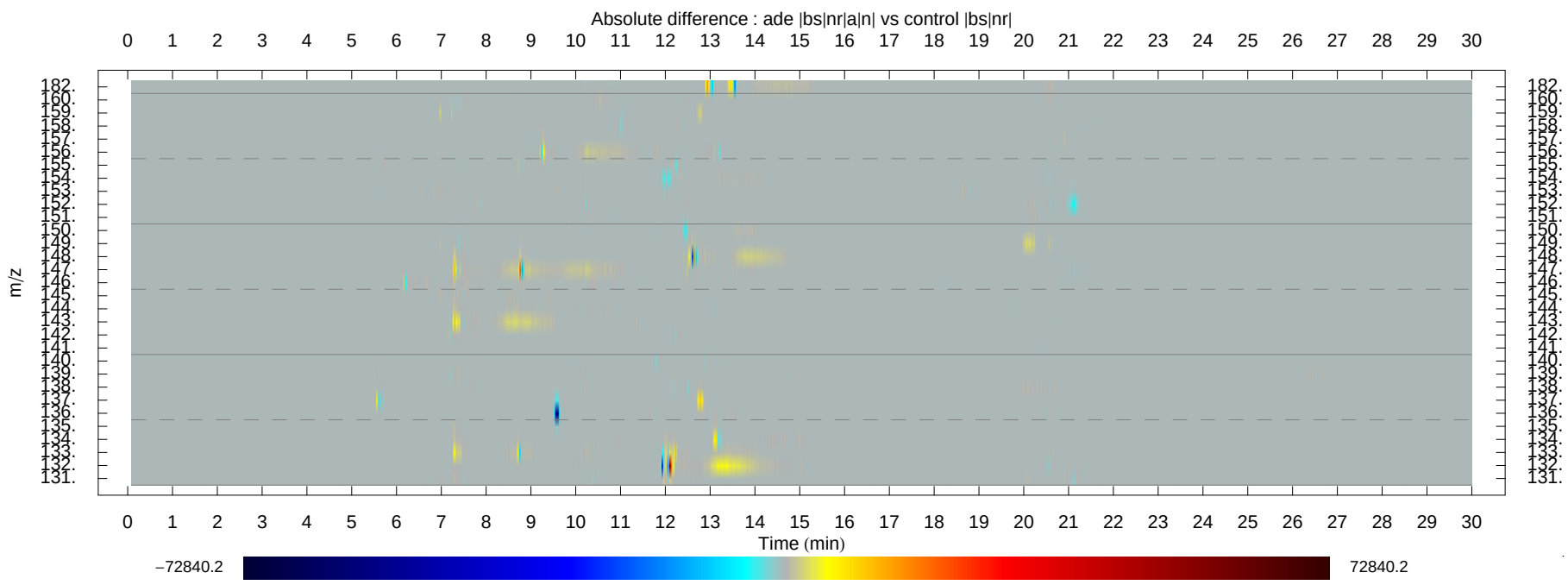
```
DAMPParallelPlot[msdatas,options] plots msdatas (a list of datasets) on a density plot in a parallel format, so that chromatograms/
  electropherograms from the datasets corresponding to the same m/z value appear next to each other. The datasets do not have to have
  chromatograms corresponding to an identical set of m/z values, neither do the datasets have to be aligned. In the latter case, the
  time axis is labeled according to the first dataset in msdatas (please note that the time axis may be misleading for the remaining
  datasets in this case). The options are passed directly to the DAMPDensityPlot function which is used internally for plotting the data.
```

```
DAMPParallelPlot[{ppctrl, normasmpl}, mzGridLineFreq → 4];
```



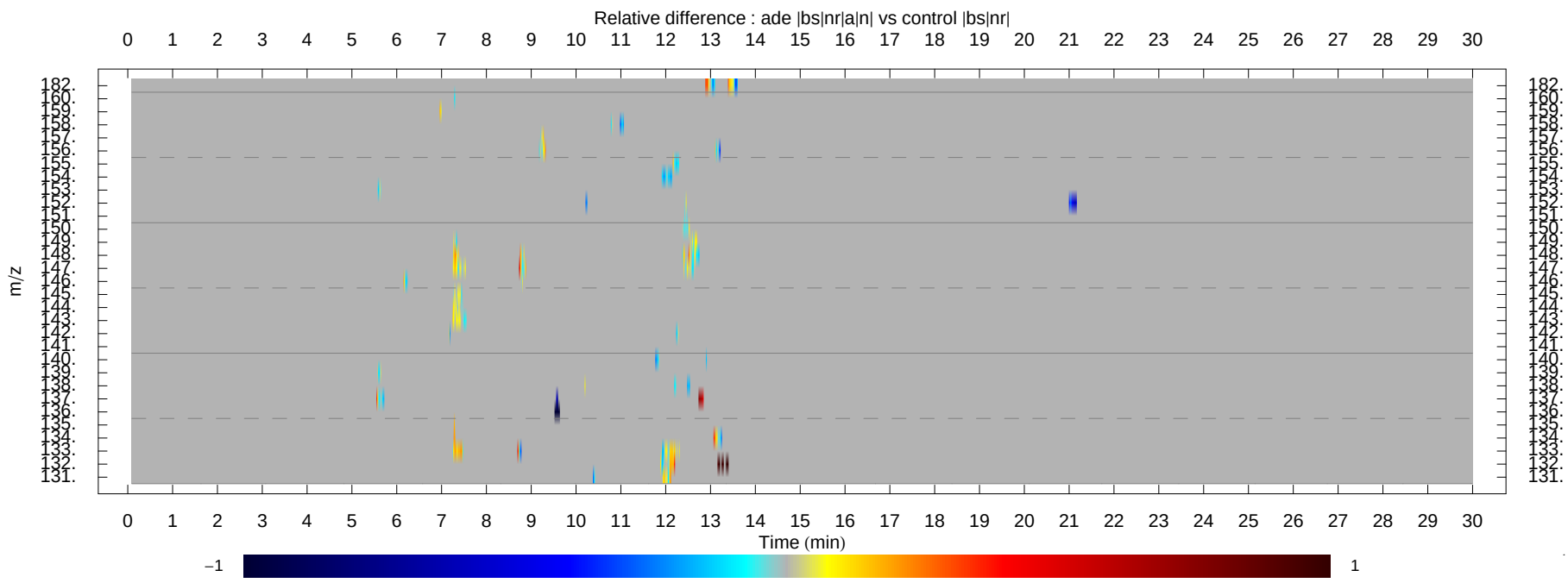
Additionally, simple arithmetic operations may be performed on the signal intensity matrices of the two normalized datasets to highlight differences between them. Subtraction provides a dataset representing the difference in signal intensities. The `normasmpl` dataset contains identical timepoints to the `ctrl` dataset (via the `DAMPAlign` function) so the m/z value list as well as the list of timepoints is taken from the `ctrl` dataset.

```
absdif = {normasmpl[[1]] - ppctrl[[1]], ppctrl[[2]], ppctrl[[3]],
   {SampleName → "Absolute difference : " <> (SampleName /. normasmpl[[4]]) <> " vs " <> (SampleName /. ppctrl[[4]])}};
DAMPDensityPlot[
  absdif];
```



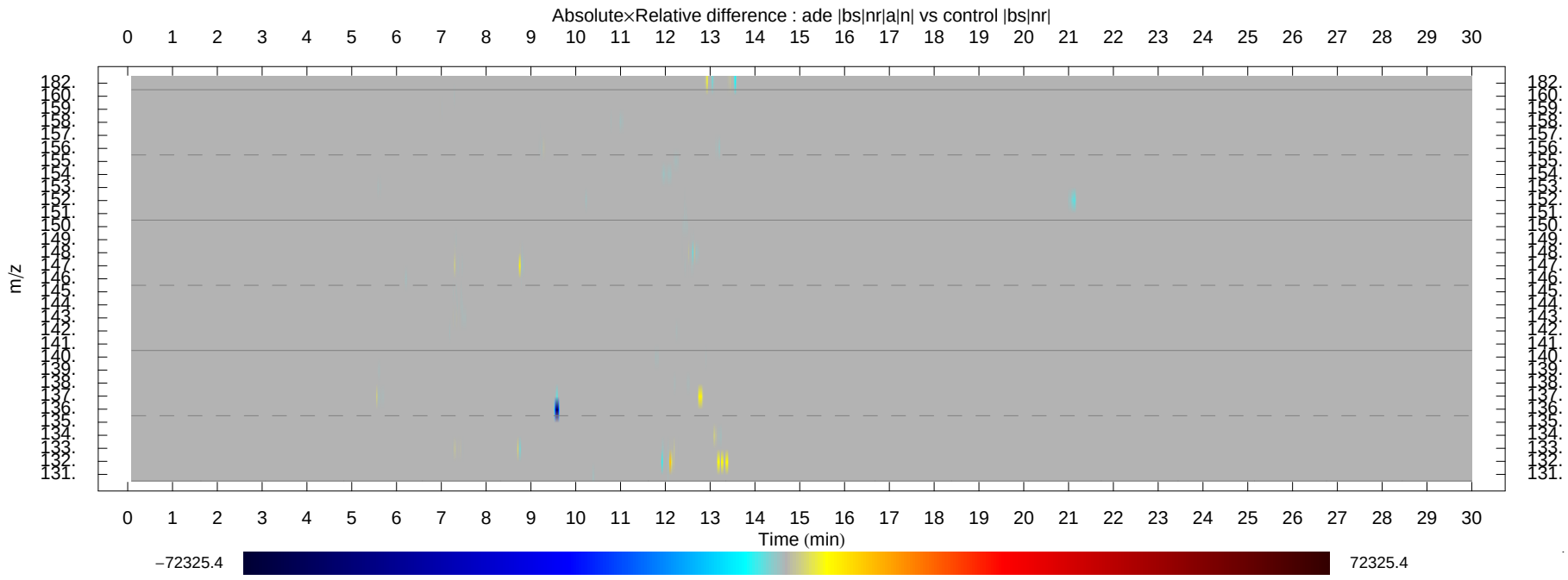
The result contains some signals indicating either positive (yellow/red) or negative (cyan/blue) difference between the datasets. Some ambiguous signals (red and blue in close proximity) appear on the result as well. These may be caused by partial misalignments of the corresponding peaks or small relative differences in significant peaks (the small relative difference is significant in absolute terms). For instance, there are two ambiguous signals in the topmost lane (m/z 182) around migration time of 13 min on the density plot above. These correspond to the peaks shown on the last electropherogram in the previous section. The signals on the density plot are due to an imperfect overlap of the corresponding peaks. The ambiguous signals may be ruled out as false positives either after the visual inspection of overlaid chromatograms/electropherograms (which can be generated automatically in a ranked order as described in the next section) or the presence of these signals may be suppressed using different kinds of visualization approaches described below.
In a way similar to the absolute difference, a relative difference between the two datasets can be calculated. In this case, the difference between the corresponding signal intensities is divided by the larger of the two (or an absolute value of the difference between the two signal intensities, if one of them is negative). The signal intensities in the resulting dataset fall within the range −1 to 1.

```
reldif = {(normasmpl[[1]] - ppctrl[[1]]) /
    (Chop[Max[Join[Abs[#], {Abs[#[[1]] - #[[2]]}]]] & /@ Transpose[#] & /@ Transpose[{normasmpl[[1]], ppctrl[[1]]}], 5000] /. {0 → ∞, 0. → ∞}),
   ppctrl[[2]], ppctrl[[3]], {SampleName → "Relative difference : " <> (SampleName /. normasmpl[[4]]) <> " vs " <> (SampleName /. ppctrl[[4]])}};
DAMPDensityPlot[reldif, MaxScale → 1];
```



Tiny peaks (often noise−related which evaded preprocessing) may provide significant signals on the relative result. A signal intensity threshold of 5000 suppresses these influences in the previous result. In spite of this, numerous misleading signals still remain in the result. A simple way to suppress signals originating from small relative changes in huge peaks (scoring high on the absolute difference plot) and significant relative differences in tiny noise−related peaks (scoring high on the relative difference plot) is to multiply the absolute and relative difference results (below). Differences significant in both absolute and relative terms tend to be highlighted. As shown below, ambiguous signals become suppressed and signals coming from actual differences acquire better visibility. This holds even if the threshold for the relative difference is set to 0.

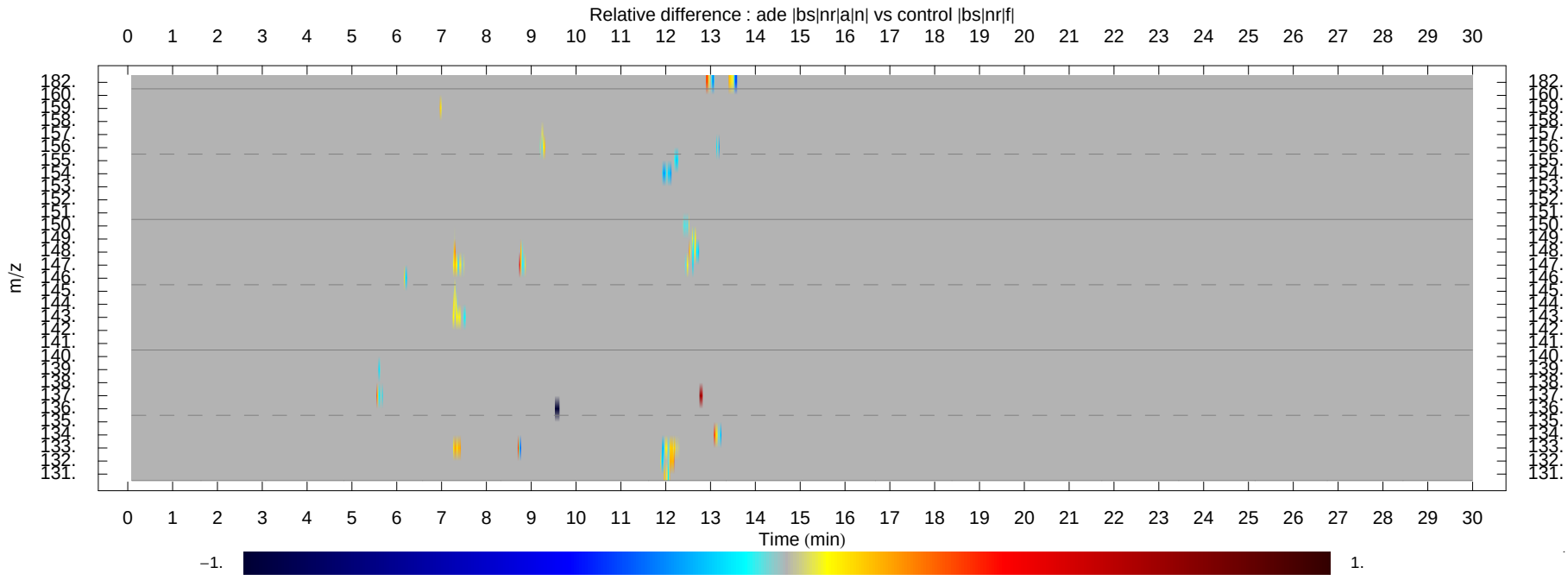
```
absreldif = {absdif[[1]] Abs[reldif[[1]]], ppctrl[[2]], ppctrl[[3]],
   {SampleName → "Absolute×Relative difference : " <> (SampleName /. normasmpl[[4]]) <> " vs " <> (SampleName /. ppctrl[[4]])}};
DAMPDensityPlot[
  absreldif];
```



Availability of replicate datasets allows the application of statistical tests to all corresponding signal intensities. Examples may be found in notebooks 04−MathDAMP−Outliers.nb (looking for outliers within multiple datasets using *z*−scores and by analyzing quartiles), 05−MathDAMP−TwoGroups (comparison of two groups of replicates includes the *t*−test), and 06−MathDAMP−MultipleGroups.nb (comparing multiple groups of replicates using *F* ratio). Noise removal proves not to be necessary when using these approaches. However, the resulting datasets are usually smoothed (by applying a moving average filter) to suppress strong signals originating from 'lucky' constellations of a particular set of corresponding noise related signal intensities (without any strong signals in their neighborhood).

Any resulting datasets may be further combined (in a way similar to the absolute×relative result) or used as a filtration criteria for other results. Below is an example of selecting only those datapoints from the relative difference, where at least one of the corresponding signal intensities in the `ctrl` and `smpl` datasets exceeds a threshold (10000). Using the `DAMPFilter` function may prove especially useful when a result of a statistical test (like the *t*−test for two groups of replicates) is used as the criteria dataset (to filter the absolute×relative result between the averaged groups for instance).

```
DAMPDensityPlot[DAMPFilter[reldif, DAMPApplyFunctionToGroup[{ppctrl, normasmpl}, Max], 10000]];
```



The `DAMPApplyFunctionToGroup` applies a specified pure function to corresponding signal intensities in the group of datasets. Above, the function was used to create a dataset containing maxima from corresponding signal intensities in the `ctrl` and the `smpl` datasets (by using the `Max` function as the specified pure function).

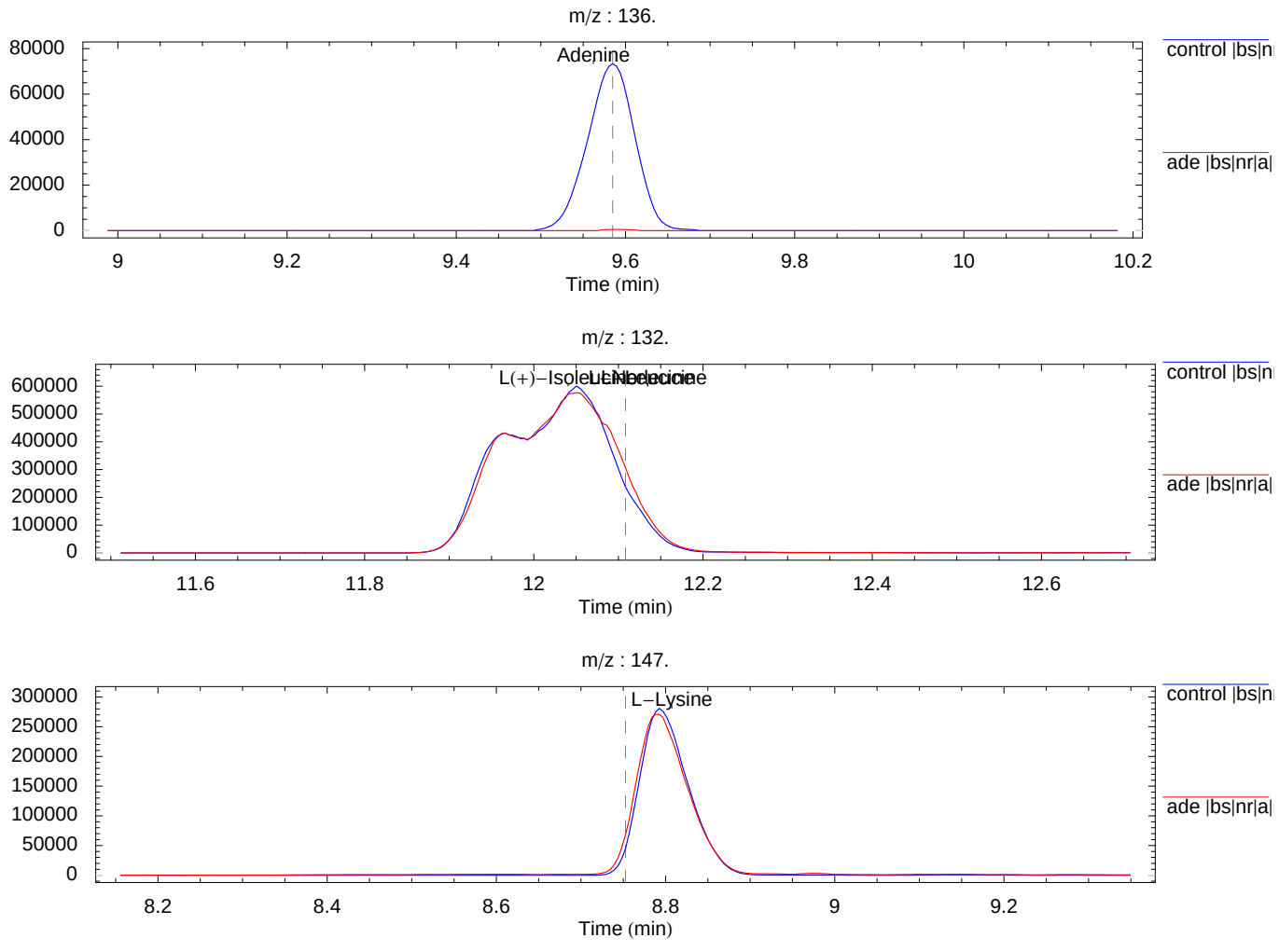
```
?DAMPApplyFunctionToGroup

DAMPApplyFunctionToGroup[msdatas,function,options] applies function
  to either all corresponding signal intensities in the datasets or to the msdatas (decided by option).
Options:
ApplyToIntensitiesOnly - if set to true, the function is applied to the
  corresponding signal intentisites in the msdatas (instead of to the whole msdatas list) (default: True)
ResultSampleName - string to be set as the SampleName of the resulting dataset. If set to Automatic,
  the SampleName of the first dataset from msdatas is used (default: Automatic)
SampleNameSuffix - string to be added to the SampleName to keep the track of the modifications performed on the dataset (default: "")
```

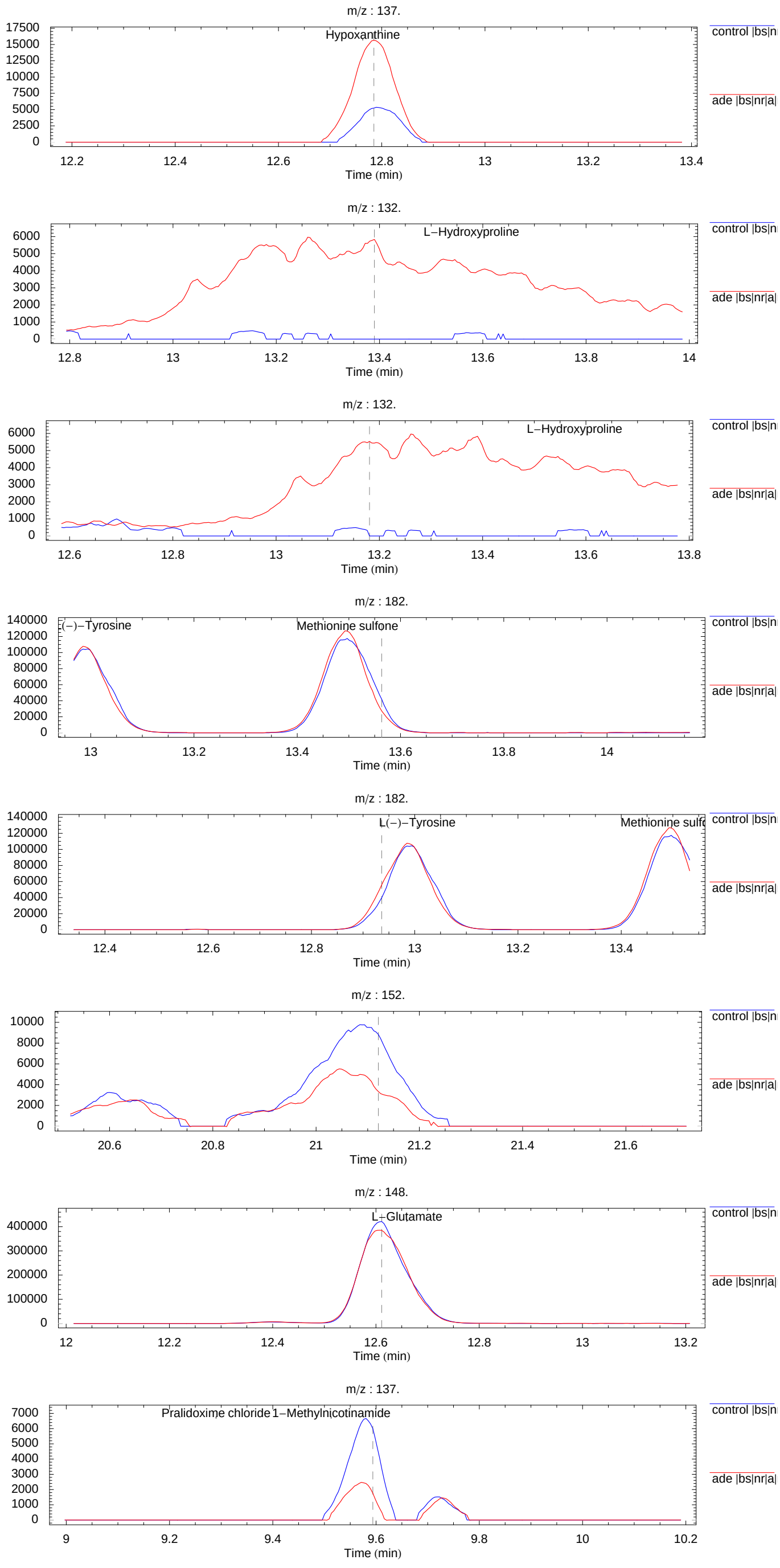
```
?DAMPFilter

DAMPFilter[msdata,criteriamsdata,threshold,options] sets to 0 those signal intensities in msdata for which the
  aboslute values of corresponding signal intensities in the criteriamsdata dataset are not equal to or greater than threshold.
Options:
SampleNameSuffix - string to be added to the SampleName from the msdata to keep track of modifications performed on the dataset (default: "f")

DAMPFilter[msdata,criteriamsdata,filterfunction,options] pure function filterfunction is applied to the signal intensity matrix of
  criteriamsdata. For zero signal intensities in the result, the corresponding signal intensities in the msdata dataset are set to 0 as well.
Options:
SampleNameSuffix - string to be added to the SampleName from the msdata to keep the track of modifications performed on the dataset (default: "f")
```

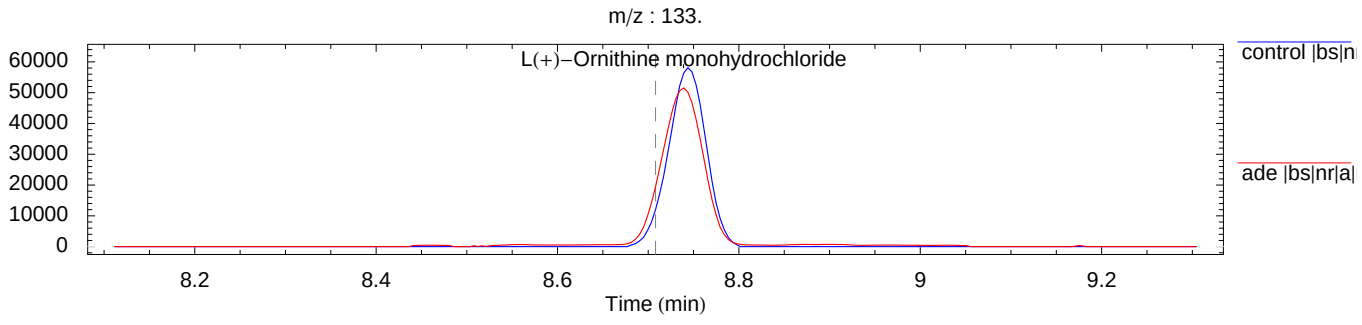
■ Listing the overlaid chromatograms/electropherograms in the vicinities of the most significant differences

For the visual confirmation of significant differences between the datasets (and for the rejection of false positives), overlaid chromatograms/electropherograms are plotted in descending order of significance. Below are the electropherograms of the top 12 differences from the absolute×relative difference result from above. The vertical dashed line indicates the position of the most signifi−cant difference according to the selected criteria.

```
DAMPPlotCandidates[{ppctrl, normasmpl}, absreldif, PlotCount → 12, PlotChromatogramOptions → {AnnotationTable → alignedannottab}];
```




m/z : 137.
Hypoxanthine
Time (min)
m/z : 132.
L−Hydroxyproline
Time (min)
m/z : 132.
L−Hydroxyproline
Time (min)
m/z : 182.
(−)−Tyrosine
Methionine sulfone
Time (min)
m/z : 182.
L(−)−Tyrosine
Time (min)
m/z : 152.
Time (min)
m/z : 148.
L+Glutamate
Time (min)
m/z : 137.
Pralidoxime chloride
1−Methylnicotinamide
Time (min)
control |bs|n
ade |bs|nr|a|



This notebook demonstrated the basic core functionality of the *Math*DAMP package. For a more convenient usage, the core functions are assembled into modules for common types of differential analysis of metabolite profiles. Examples can be found in the additional notebooks (03−MathDAMP−TwoDatasets.nb, 04−MathDAMP−Outliers.nb, 05−MathDAMP−TwoGroups, and 06−Math−DAMP−MultipleGroups.nb) from the *Math*DAMP package.