HowTo Use the Bioconductor PROcess package

April 25, 2006

Contents

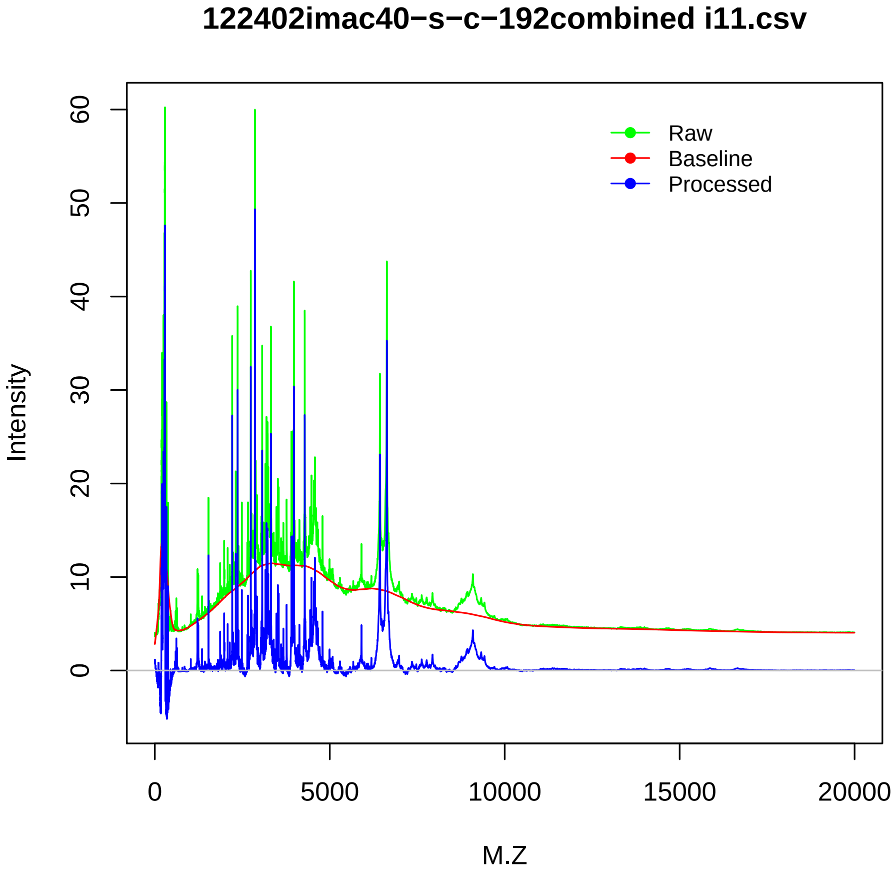
1 Introduction

The `PROcess` package contains a collection of functions for processing spectra to remove baseline drifts if any, detect peaks and align them to a set of protobiomarkers. This document serves as a quick tutorial for using the `PROcess` package.

2 Baseline subtraction

Our first observation of a raw spectrum is that it exhibits elevated baseline, more so at smaller m/z values than at larger values. This elevated baseline is mostly caused by the chemical noises in the EAM and ion overload. Ideally a spectrum should rest more or less on the zero horizontal line. This baseline needs to be subtracted from each raw spectrum. The following example shows the result of a spectrum with its baseline removed.

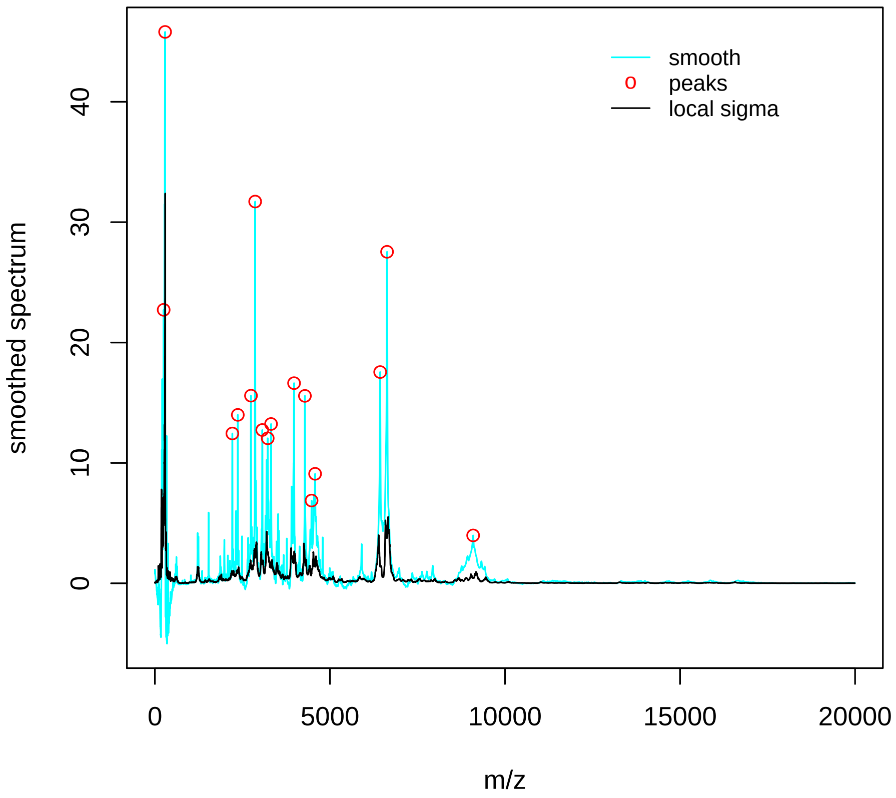
```
> library(PROcess)

Loading required package: Icens
Loading required package: survival
Loading required package: splines

> fdat <- system.file("Test", package = "PROcess")
> fs <- list.files(fdat, pattern = "\\.*csv\\.*", full.names = TRUE)
> f1 <- read.files(fs[1])
> fcut <- f1[f1[, 1] > 0, ]
> bseoff <- bslnoff(fcut, method = "loess", plot = TRUE, bw = 0.1)
> title(basename(fs[1]))
```



3 Peak detection

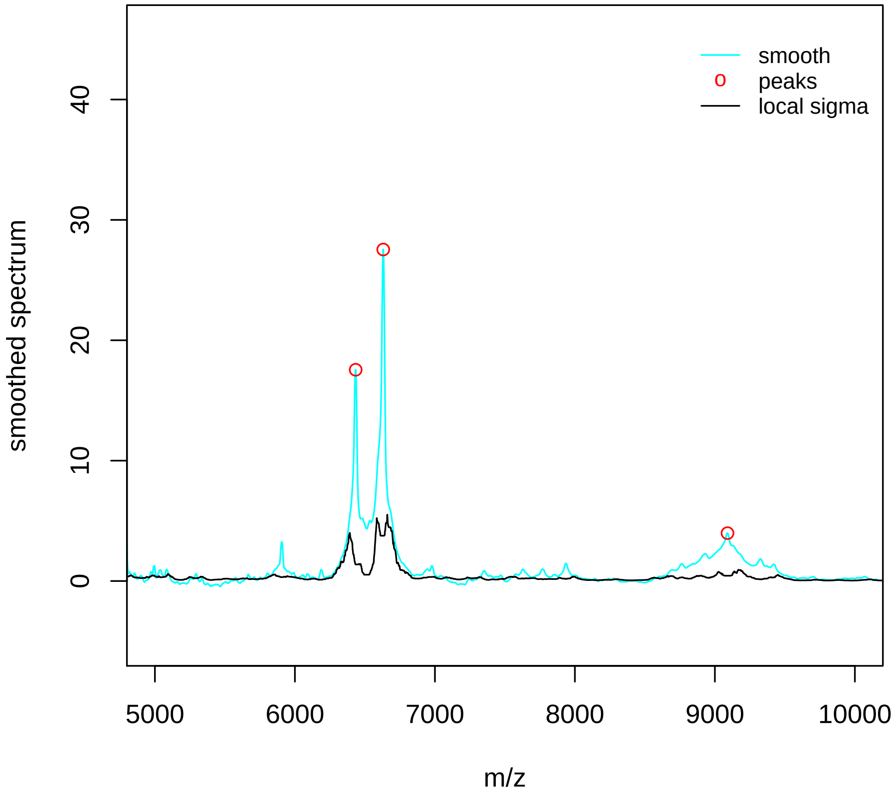
After baseline is removed, peaks can be located by using `isPeak`. A spectrum is smoothed first using moving average of the k nearest neighbours. Smoothing helps to enhance peaks and get rid of spurious peaks. However, we do not recommend large amount of smoothing (controlled by parameter `sm.span`) in this step because we do not wish to smooth away too many short and wide peaks and also we need the precision in peak locations. As a first step we do not mind getting more potential features.

```
> pkgobj <- isPeak(bseoff, span = 81, sm.span = 11, plot = TRUE)
```



We can also zoom in to inspect peaks in a particular range of m/z values.

```
> specZoom(pkgobj, xlim = c(5000, 10000))
```



4 Batch operation

We demonstrate the batch functionality of this package using a set of 2 spectra.

4.1 Apply baseline subtraction to a set of spectra

```
> testdir <- system.file("Test", package = "PROcess")
> testM <- rmBaseline(testdir)
```

4.2 Renormalize spectra

Suppose we want to normalize a set of spectra to their median AUC (Area Under the Curve), where an AUC is calculated for m/z values greater than a cutoff point, 1500.

```
> rtM <- renorm(testM, cutoff = 1500)
```

4.3 Identify peaks of spectra

```
> peakfile <- paste(tempdir(), "testpeakinfo.csv", sep = "/")
> getPeaks(rtM, peakfile)
```

4.4 Quality assessment

Quality assessment is necessary because some spectra are very noisy and have hardly any peaks. Function `quality` computes three parameters `Quality`, `Retain` and `peak` for assessing a set of spectra.

```
> qualRes <- quality(testM, peakfile, cutoff = 1500)
> print(qualRes)
```

```
                                         Quality    Retain      peak
122402imac40-s-c-192combined i11.csv 0.5419812 0.3825606 1.2727273
122402imac40-s-c-192combined i12.csv 0.6234705 0.3548255 0.7272727
```

A spectrum is deemed of poor quality and should be removed from subsequent analyses if it meets the following 3 conditions simultaneously:

1. `Quality` $<$ 0.4;

2. `Retain` $<$ 0.1;

3. `peak` $<$ 1/2 of the mean peak number in the chip.

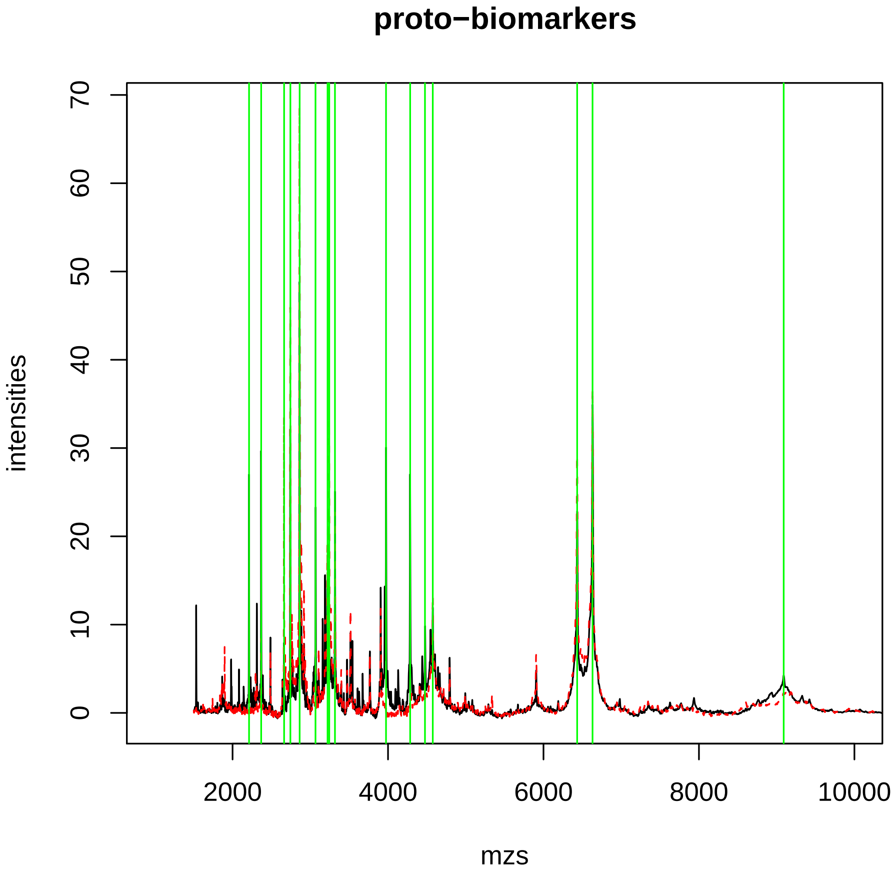
4.5 Get protobiomarkers

One challenge in MS data is that not only they exhibit variation vertically but also do they horizontally. This horizontal variation is not simply a constant shift but associated with value of m/z. Currently the accuracy in the m/z position is believed to be within 0.3% o f the m/z value. Once the peaks are detected, we align the peaks by first generating an interval of size 0.3% of the m/z value which centers at m/z for each m/z where a peak is detected. We treat those m/z intervals as interval censored data. We treat those intervals as a partially ordered set and use the locations of the maximal cliques to define the locations of the peaks **?**. We call these aligned peaks (across spectra) proto-biomarkers and use the centers of the resulting intervals as the locations of the aligned peaks. For each spectrum we determine which actual peaks are represented by an aligned peak (proto-biomarker) and use the maximum of those as the height of the proto-biomarker. If there were no peaks then we use the maximum value within the resulting interval.

```
> bmkfile <- paste(tempdir(), "testbiomarker.csv", sep = "/")
> testBio <- pk2bmkr(peakfile, rtM, bmkfile)
> mzs <- as.numeric(rownames(rtM))
> matplot(mzs, rtM, type = "l", xlim = c(1000, 10000), ylab = "intensities",
+     main = "proto-biomarkers")
> bks <- getMzs(testBio)
> print(round(bks))
```

```
 [1] 2212 2368 2663 2744 2863 3067 3222 3244 3317 3974 4285 4475 4576 6433 6631
[16] 9090
```

```
> abline(v = bks, col = "green")
```



5 An alternative way to obtain proto-biomarkers

? propose an alternative way for peak detection using the average spectrum of all spectra of a given experiment. Their algorithm is comprised of the following steps, (1) compute the mean of all raw spectra, (2) de-noise, baseline correct and find peaks by locating all local maxima in the mean spectrum, and (3) quantify the identified peaks in the individual spectra. The major advantage of this approach is the simplicity and the speed to arrive at a set of proto-biomarkers in comparison to the approach described in the previous sections to select peaks from individual spectra.

In PROcess we adopt the general idea of using mean spectrum to locate peaks, but leave the decision to users whether they should compute the mean spectrum of the raw spectra, or of the smoothed (de-noised), baseline corrected and normalized spectra for their experiment at hand. If the steps of baseline-subtraction and normalization are done properly, there may be improvement in peak detection using the mean spectrum computed from the pre-processed spectra.

Our approach comprises the following steps.

- Compute the mean of all raw spectra using aveSpec, or the mean of all pre-processed spectra using standard R function rowMeans.
- Detect peaks of the mean spectrum by isPeak.
- Align peaks by align if there seems to be peak clusters.
- Quantify peaks in individual spectra that have been smoothed, baseline-removed and normalized, by getPeaks2 that locates for each peak the maximum intensity in a neighbourhood of the peak defined by a user-specified precision of peaks.

We now demonstrate this approach to the test data set using PROcess.

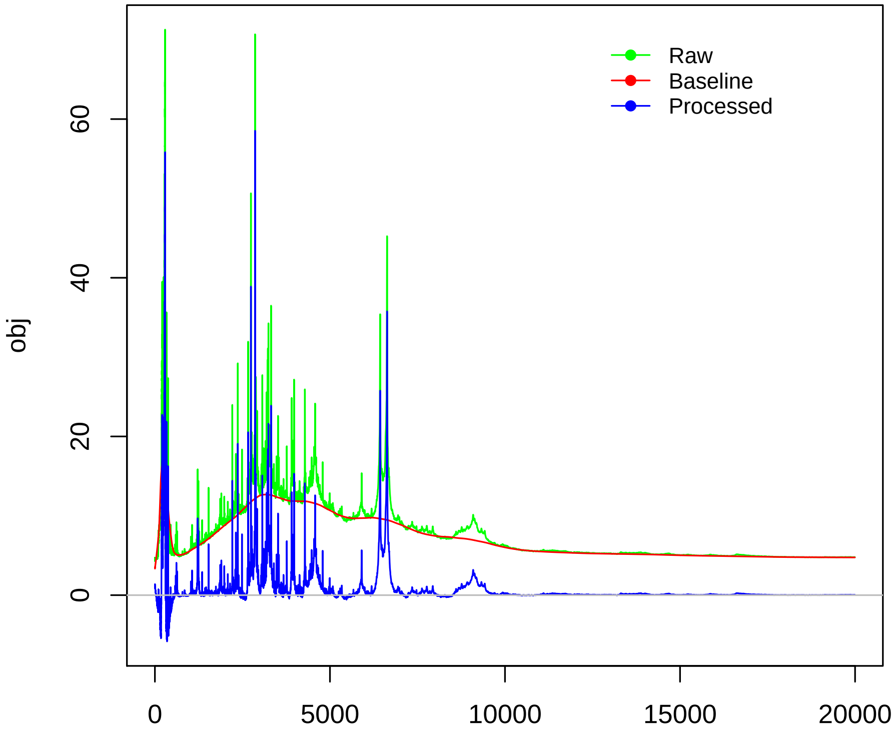
We execute the following code to compute the mean spectrum.

```
> grandAve <- aveSpec(fs)
> mzs <- grandAve[, 1]
```

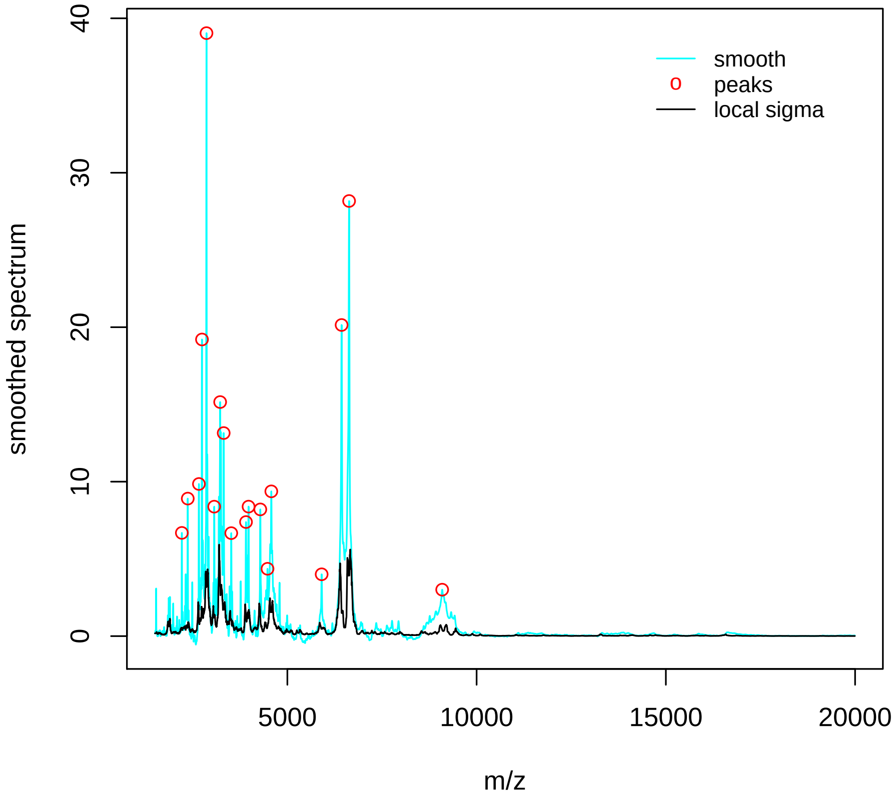
Should you wish to compute the mean spectrum of the pre-processed spectra, you can do rowMeans(rtM) instead, then skip the baseline correction step and proceed to the peak detection step using isPeak .

We execute the following code to remove baseline, detect peaks in the overall mean spectrum and quantify them in individual spectra.

```
> grandOff <- bslnoff(grandAve[mzs > 0, ], method = "loess", plot = T,
+     bw = 0.1)
```



```
> grandPkg <- isPeak(grandOff[grandOff[, 1] > 1500, ], zerothrsh = 1,
+     plot = T, ratio = 0.1)
> grandpvec <- round(grandPkg[grandPkg$peak, "mz"])
> print(as.vector(grandpvec))

 [1] 2212 2368 2663 2744 2863 3067 3224 3317 3518 3907 3974 4285 4475 4576 5906
[16] 6432 6631 9090
```



We can then quantify the peaks by running the following code.

```
> grandBmk <- getPeaks2(rtM, grandpvec)
```

References