Chemometric Analysis of Spectroscopic Data in R: hyperSpec

C. Beleites (cbeleites@units.it) and V. Sergo

CENMAT, Materials and Natural Resources Dept., University of Trieste, Via Valerio 6/a, Trieste/Italy

Motivation

We present **hyperSpec**, a new software that greatly facilitates the analysis of spectra using the statistical software R [1, http://www.r-project.org]. Our needs for biospectroscopic data analysis are best met in a progamming environment supplying tools for both chemometrics and handling of spectra.

Standard and specialized statistical procedures are available in R. This is a big advantage as programming the handling of spectra is far less error prone than progamming statistical routines. The correctness of statistical software is a concern, and R was assessed by Keeling and Pavur[2].

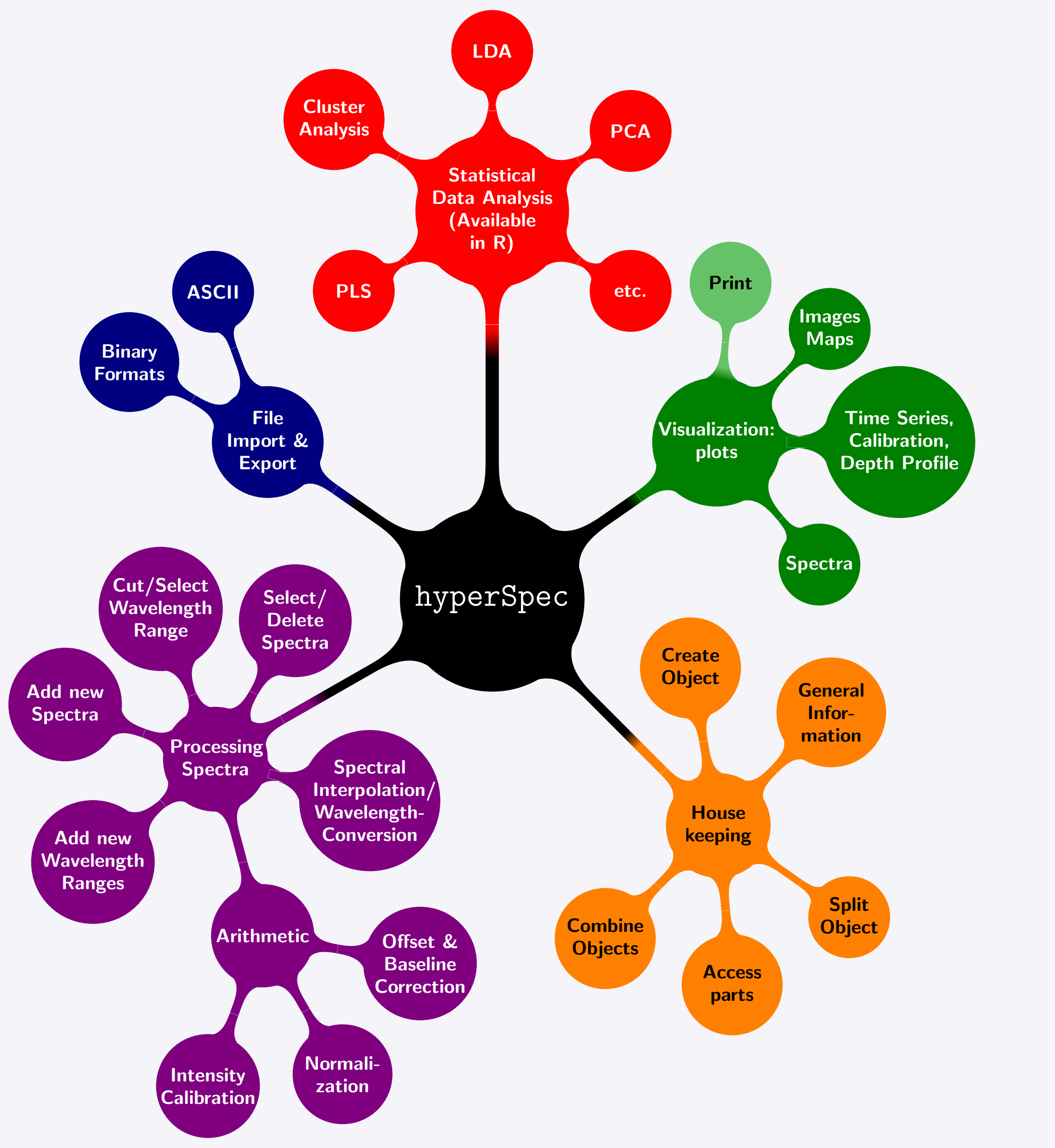
hyperSpec makes R a convenient platform for the analysis of spectral data sets, including spectral images and maps.

Requirements

Our key scenarios for chemometric analysis in biomedical spectroscopy are:

1. We do spectral preprocessing, and use chemometric methods like regression, cluster analysis, classification, etc.
2. We acquire spectral maps of arbitrary shape.
3. The data sets can be large (20 000 spectra and more), so batch processing/scripting should be possible.
4. We combine maps with single spectra (e. g. reference substances).
5. We customize/extend standard procedures if our needs are not met. See e. g. the “Centering” in the chondrocyte example below, or think of developing a diagnostic test: How to ensure statistical independence at patient level with varying numbers of spectra per patient?
6. We build Graphical User Interfaces (GUIs) for specific tasks.

Features



hyperSpec provides . . .

- Functions to import spectra into R.
- Means to attach any amount of non-spectral data to each of the spectra, such as time, position, concentrations, diagnoses, etc.
- Several plot funcions to display spectra, false-colour maps, calibration lines etc., and basic interaction like obtianing the spectrum and wavelength you click at.
- Functions to work with the spectra and do your preferred preprocessing.
- Functions to ease the interaction with statistical data analysis methods.
  - Hand over the spectra matrix e. g. to ***dist*** in the chondrocyte example,
  - Hand over the appropriate data.frame e. g. to ***lm*** in the calibration example.
  - Re-importing the results of e. g. PCA or MSC preprocessing is also possible.

In R, you can . . .

- Add and change functions and methods at runtime, e. g. add functions to im-/export your spectrometer’s proprietary data format.
- Do batch processing, remote calculations, and external calculations in Matlab[3]
- Use *literate programming* to include calculations in (text) documents.
- Write a GUI tailored to your specific data analysis requirements.
- Use advanced statistical procedures.

Example work flow: Raman Map of Chondrocytes in Cartilage

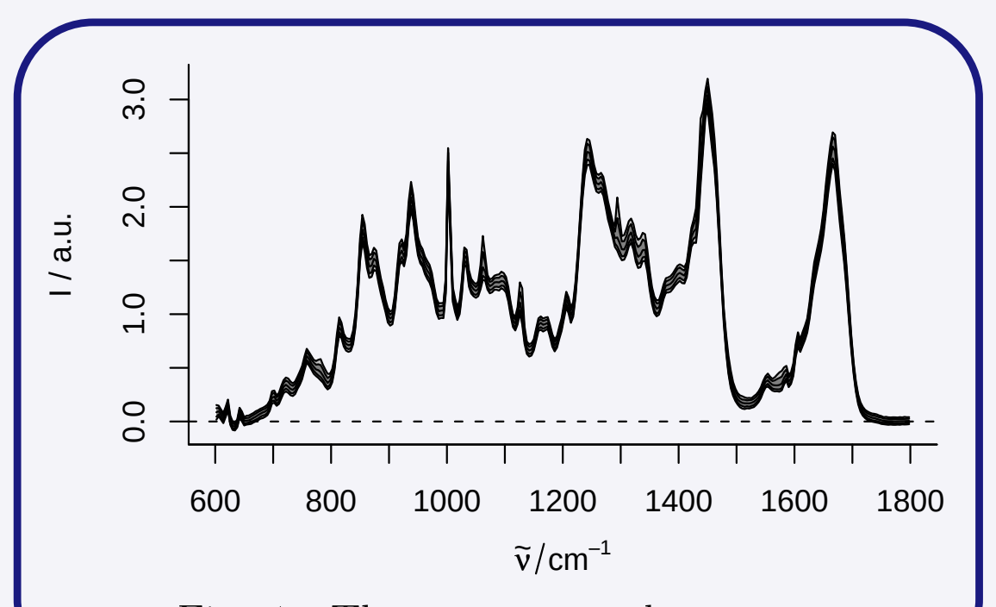


Fig. 1 The preprocessed spectra.

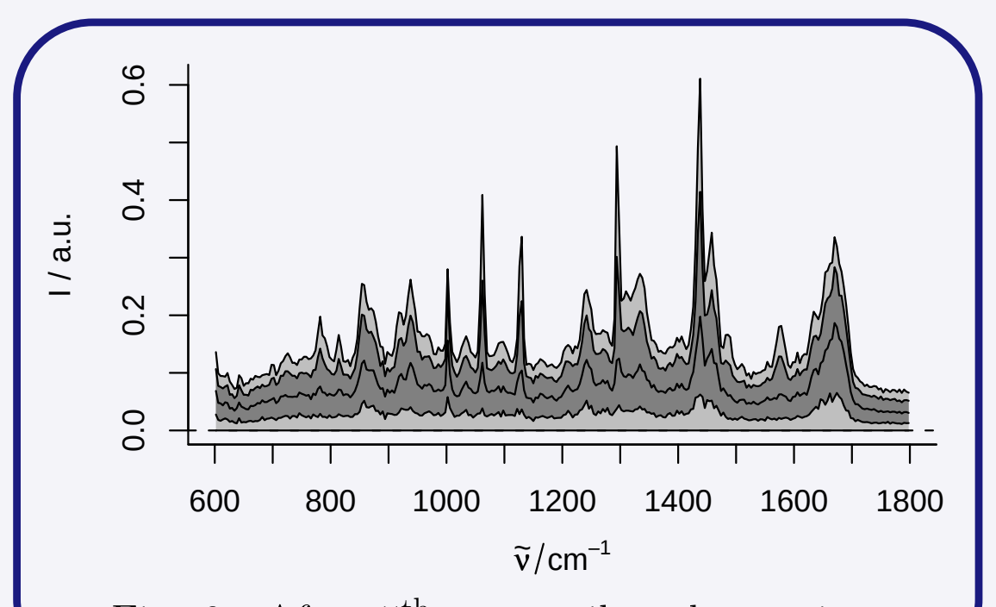


Fig. 2 After 5th percentile substraction.

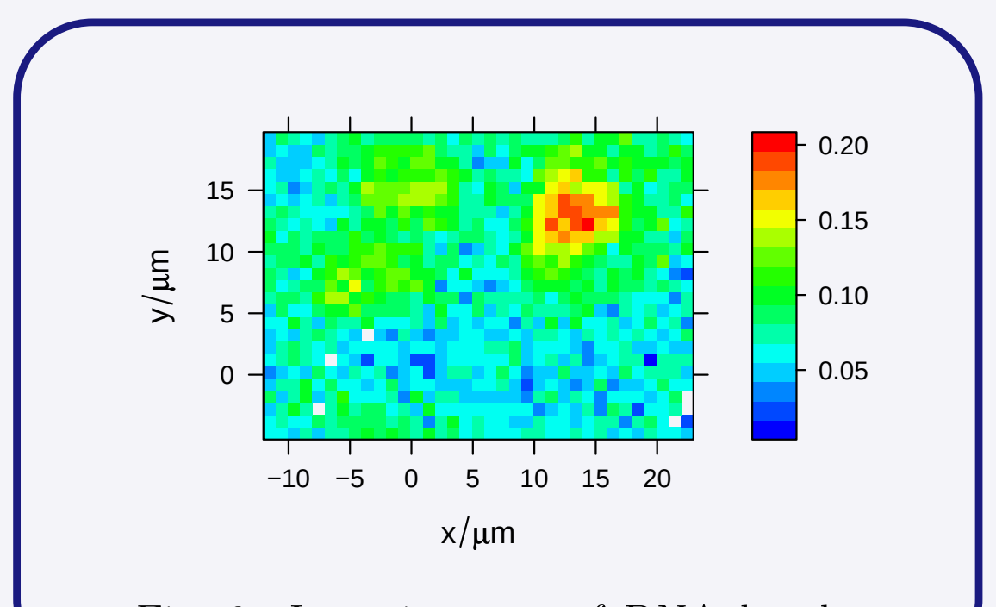


Fig. 3 Intensity map of DNA bands.

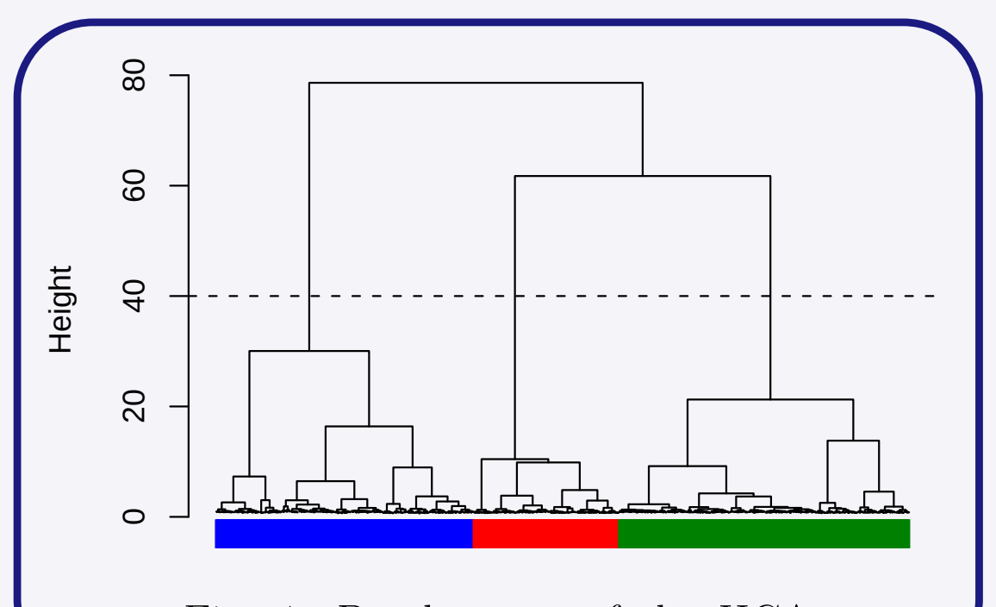


Fig. 4 Dendrogram of the HCA

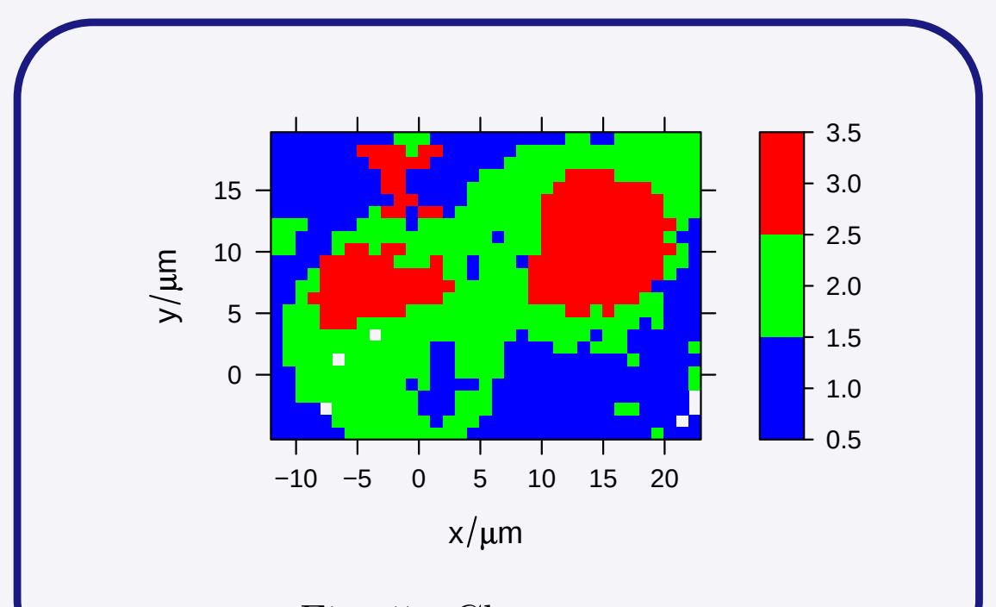


Fig. 5 Cluster map.

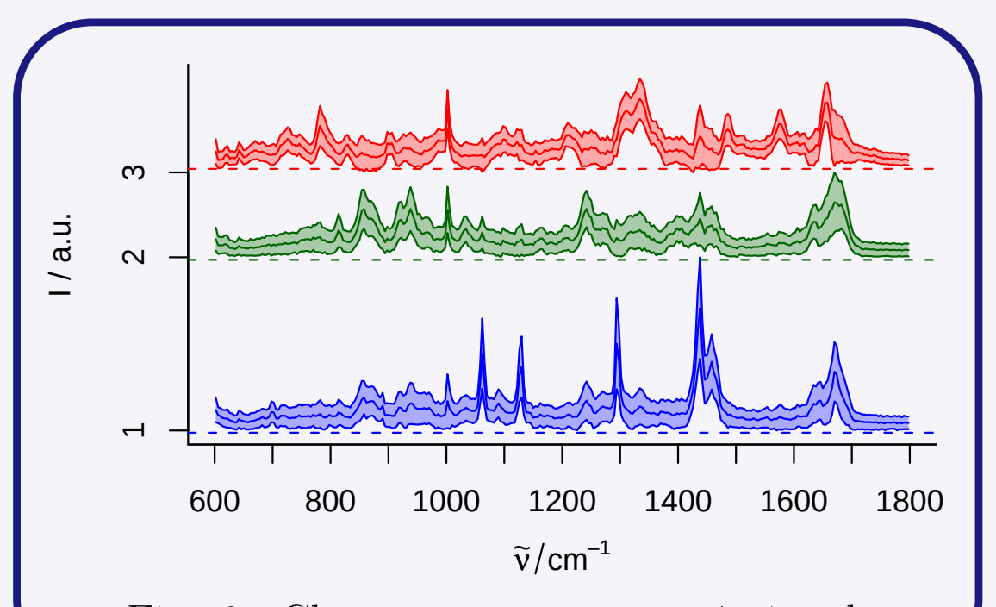
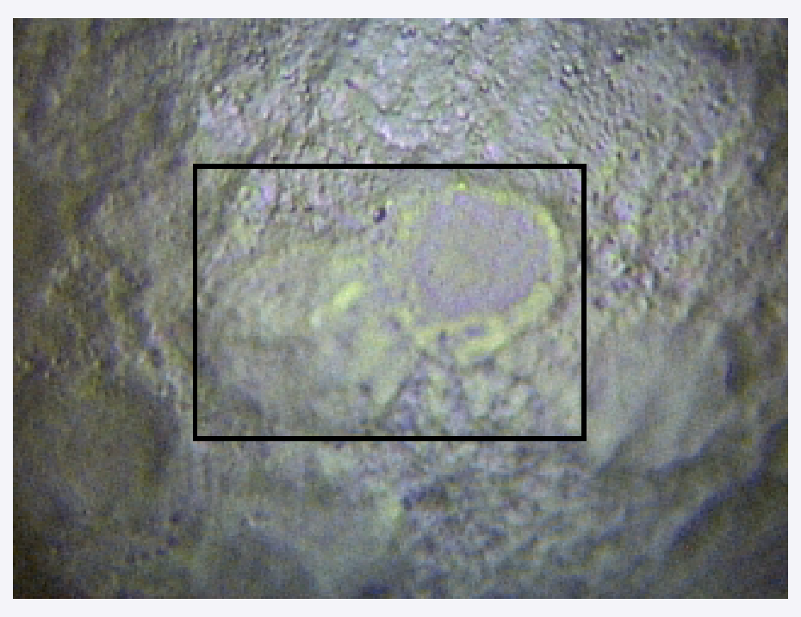


Fig. 6 Cluster mean spectra ± 1 std.



Sample: thick section of pig cartilage
Spectrometer: Renishaw InVia
Excitation: 633 nm, 10 s / spectrum
Objective: 100×, NA 0.85
Measurement Area: 35 × 21 µm
Grid: 1 µm step size

Loading the library

```
> library (hyperSpec)
```

Data import

```
> chondro <- scan.txt.Renishaw ("chondro.txt", data = "xyspc")
> print (chondro)
hyperSpec object
   875 spectra
   3 data columns
   1272 data points / spectrum
wavelength: tilde(nu)/cm^-1 [numeric 1272]  601.622 602.664 ... 1802.15
data:  (875 rows x 3 columns)
   (1) y: y/(mu * m) [numeric 875] range -4.77 -3.77 ... 19.23
   (2) x: x/(mu * m) [numeric 875] range -11.55 -10.55 ... 22.45
   (3) spc: I / a.u. [AsIs matrix 875 x 1272] range 52.2573 52.5012 ... 1884.25 + NA
```

Preprocessing

Smoothing interpolation of $\tilde{\nu}$

```
> chondro <- spc.loess (chondro, seq (602, 1800, 4))
```

Linear baseline correction

```
> chondro <- chondro - spc.fit.poly.below (chondro)
Fitting with npts.min =  15
```

Normalization

```
> chondro <- sweep (chondro, 1, apply (chondro, 1, mean), "/")
```

Outlier Removal (see the complete work flow in **hyperSpec**’s documentation).

```
> chondro <- chondro [- c(105, 140, 216, 289, 75, 69)]
```

Plotting spectra (fig. 1)

```
> plot (chondro, "spcprctl5")
```

Instead of centering, subtract the 5th percentile spectrum (fig. 2).

```
> chondro <- sweep (chondro, 2, apply (chondro, 2, quantile, 0.05), "-")
> plot (chondro, "spcprctl5")
```

Intensity map: Looking at Nucleic Acid Bands

The requested wavelengths can be entered directly:

```
> print (plotmap (chondro[, , c(728, 782, 1098, 1240, 1482, 1577)]))
```

Hierarchical cluster analysis

Calculate dendrogram with EUCLIDean distance and WARD’s method:

```
> dendrogram <- hclust (dist (chondro [[]]), method = "ward")
> plot (dendrogram)
```

Cut the dendrogram into 3 clusters and plot the cluster map (fig. 5)

```
> clusters <- cutree (dendrogram, k = 3)
> print (plotmap (chondro, z = as.factor (clusters)))
```

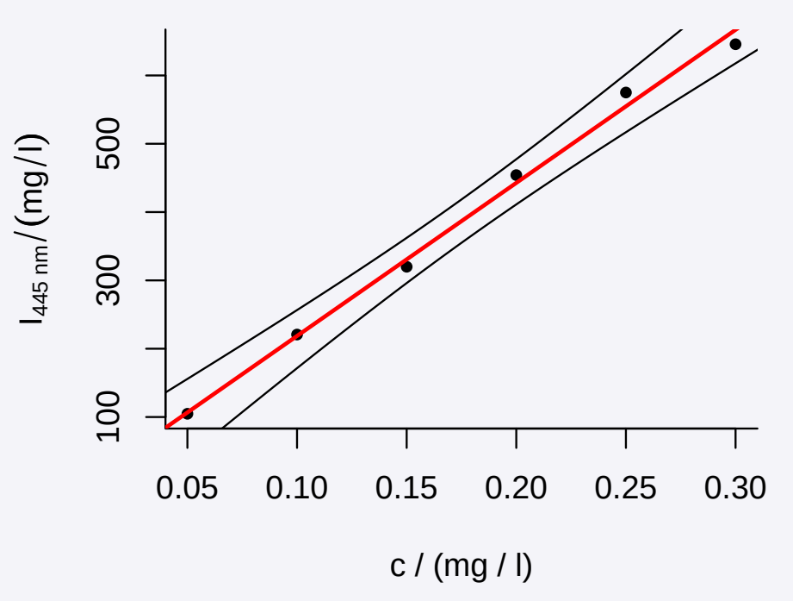
Decorate the dendrogram with colors and cut level (fig. 4):

```
> plot (dendrogram, labels = FALSE, hang = 0, main = NULL)
> abline (h = 40, lty = 2)
> col.clust <- c ("blue", "darkgreen", "red")
> points (seq (nrow (chondro)), rep (-3, nrow (chondro)),
+         col = col.clust [clusters [dendrogram$order]], pch = "|")
```

Calculate cluster mean and standard deviation spectra (fig. 6)

```
> cm <- aggregate (chondro, clusters, mean_pm_sd)
> plot(cm, stacked = ".aggregate", fill = ".aggregate",
+      col = col.clust)
```

Calibration Plot: Quinine Fluorescence



Concentrations: 0.05 – 0.30 mg/l.
Spectrometer: PE LS50-B
Excitation: 350 nm
Spectra: Fluorescence emission 405 – 495 nm
Noise added to enlarge confidence interval.

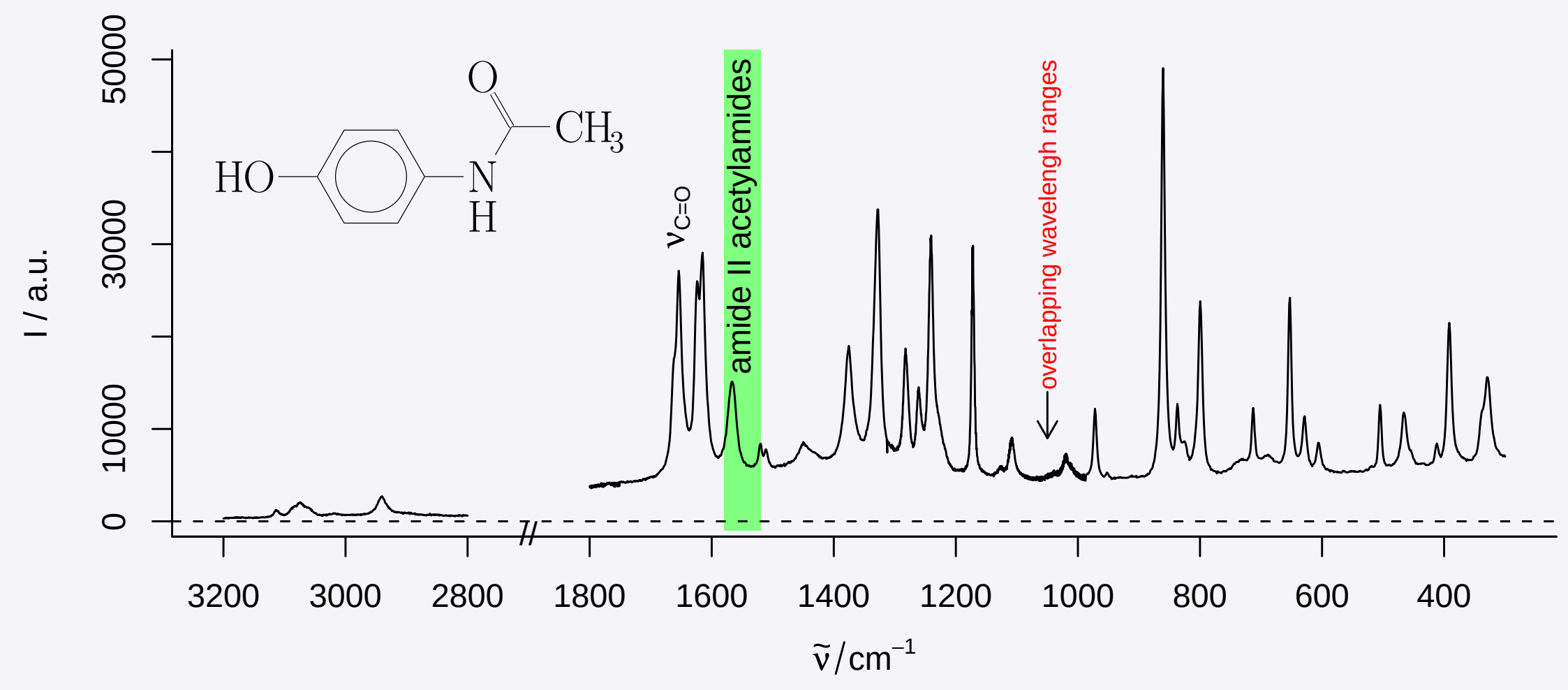
```
> calibration <- lm (c ~ spc, data = flu [,, 445]$.)
> summary (calibration)
Call:
lm(formula = c ~ spc, data = flu[, , 445]$.)

Residuals:
        1          2          3          4          5          6 
 0.0008302 -0.0009316  0.0047873 -0.0050864 -0.0090314  0.0094319 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.0024111  0.0068549   0.352    0.743
spc         0.0004463  0.0000159  28.077 9.57e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.007431 on 4 degrees of freedom
Multiple R-squared: 0.995,        Adjusted R-squared: 0.9937
F-statistic: 788.3 on 1 and 4 DF,  p-value: 9.574e-06
```

Advanced Spectra Plotting: Paracetamol



```
> plot (paracetamol, wl.range = c (300 ~ 1800, 2800 ~ max), xoffset = 800,
+      wl.reverse = TRUE)
```

Text annotations support greek letters, subscripts, etc.

```
> text (1654, paracetamol [[,,1654]] + 2500, expression (nu["C=O"]),
+      adj = c(0, 0.5), srt = 90)
```

Annotation with arrows on clicked location

```
> pt <- locator ()
> arrows (pt$x, pt$y + 5000, pt$x, pt$y, 0.1)
> text (pt$x, pt$y + 6000, c ("overlapping wavelengh ranges", col = "red"
+      adj = c(0,0.5), srt = 90, cex = 0.75)
```

Region annotation

```
> rect (1580, -1000, 1520, 100000, col = "#00FF0080", border = NA)
> text (1550, 33000, "amide II acetylamides", srt = 90)
```

Obtaining hyperSpec and Terms of Use

Homepage: http://r-forge.r-project.org/projects/hyperspec/.

Installation in R:

```
> install.packages("hyperSpec", repos="http://R-Forge.R-project.org")
```

hyperSpec is distributed under GPL >= 3.

- You are welcome to use **hyperSpec** for your data analyses.
- Properly cite the use of the package as given by: `> citation ("hyperSpec")`
- If you adapt or extend the code to your needs, you are kindly asked to make it available to the public (e. g. submit to **hyperSpec**).

Conclusions

A software package was developed to ease the analysis of hyperspectral data sets, i. e. spectra together with further information such as spatial coordinates, time series, concentrations etc., in the statistical environment R.

hyperSpec provides data im- and export, convenient plotting functions, and methods to handle and preprocess the spectra. It is easily extensible by the user, and works smoothly with other R libraries that provide specialized statistical tools.

Literature

[1] R Development Core Team. *R: A Language and Environment for Statistical Computing.* ISBN 3-900051-07-0.

[2] Kellie B. Keeling and Robert J. Pavur. “A comparative study of the reliability of nine statistical software packages”. In: *Computational Statistics & Data Analysis* 51.8 (2007), pp. 3811–3831.

[3] Henrik Bengtsson and Jason Riedy. *R.matlab: Read and write of MAT files together with R-to-Matlab connectivity.* R package version 1.2.4. 2008. URL: http://www.braju.com/R/.

Acknowledgements

The authors thank S. Paoletti and coworkers, and A. Bonifacio for kindly providing the chondrocyte data.

M. Kammer (TU Dresden/Germany) contributed the fluorescence spectra.

C. Beleites gratefully acknowledges funding by Associazione per i Bambini Chirurgici del Burlo onlus.