Package: EMSC (via r-universe)

September 7, 2024

Encoding UTF-8

Type Package

Title Extended Multiplicative Signal Correction

Version 0.9.4

Date 2024-03-11

Description Background correction of spectral like data. Handles variations in scaling, polynomial baselines, interferents, constituents and replicate variation. Parameters for corrections are stored for further analysis, and spectra are corrected accordingly.

License GPL-2

LazyData TRUE

URL https://github.com/khliland/EMSC/

BugReports https://github.com/khliland/EMSC/issues/

Depends R (>= 2.10)

Imports pracma

RoxygenNote 7.3.1

Repository https://khliland.r-universe.dev

RemoteUrl https://github.com/khliland/emsc

RemoteRef HEAD

RemoteSha 986ba17a61402bed4994b1d531f9f7d5a049a54a

Contents

EMSC *Extended multiplicative signal correction (EMSC)*

Description

Performs model-based background correction and normalisation of spectra. EMSC handles variations in scaling, polynomial baselines and interferents. Parameters for corrections are stored for further analysis, and spectra are corrected accordingly.

Usage

```
EMSC(X, model = NULL, ...)
```

Arguments

`X`	`matrix` containing spectra as rows.
`model`	an EMSC model to use instead of the other parameters.
`...`	named model parameters for EMSC_model.

Details

This is the main EMSC function performing all calculations. It can be run with no parameters (defaults are used), with a predefined EMSC model object or with parameters that are passed on to the EMSC model building function `EMSC_model`.

Value

An object of class EMSC is returned. This contains:

- `corrected`: `matrix` of corrected spectra.
- `parameters`: `matrix` of fitted parameter values.
- `model`: object containing input all input parameters.
- `X`: original data.

References

H. Martens, E. Stark, Extended multiplicative signal correction and spectral interference subtraction: new preprocessing methods for near infrared spectroscopy. J Pharm Biomed Anal. 1991; 9(8):625-35.

Joakim Skogholt, Kristian Hovde Liland, Ulf Geir Indahl, Pre-processing of spectral data in the extended multiplicative signal correction framework using multiple reference spectra Journal of Raman Spectroscopy 50(3), (2019), pp. 407-417.

See Also

`EMSC_model` `predict.EMSC` `plot.EMSC`

Examples

```
data(fishoil)
Raman      <- fishoil$Raman[, 850:3300]
EMSC.basic <- EMSC(Raman)
EMSC.poly6 <- EMSC(Raman, degree = 6)
EMSC.rep   <- EMSC(Raman, degree = 6, reference = Raman[30, ],
                   replicates = fishoil$replicates)

old.par  <- par(mfrow = c(2,2), mar = c(4,4,1,1))
xlim     <- rev(as.numeric(range(colnames(Raman))))
matplot(colnames(Raman), t(Raman), type = 'l', xlim = xlim,
        ylab = 'Relative intensity', xlab = 'Raw spectra')
matplot(colnames(Raman), t(EMSC.basic$corrected), type = 'l', xlim = xlim,
        ylab = 'Relative intensity', xlab = 'Corrected (basic)')
matplot(colnames(Raman), t(EMSC.poly6$corrected), type = 'l', xlim = xlim,
        ylab = 'Relative intensity', xlab = 'Corrected (6th degree polynomial)')
matplot(colnames(Raman), t(EMSC.rep$corrected),   type = 'l', xlim = xlim,
        ylab = 'Relative intensity',
        xlab = 'Corrected (reference = spec. #30, replicate correction (90%))')
par(old.par)
```

EMSC_model *Model object for extended multiplicative signal correction (EMSC)*

Description

Sets up an EMSC model to be applied to one or more set of spectra.

Usage

```
EMSC_model(
  x,
  reference = NA,
  degree = 2,
  interferent = NULL,
  constituent = NULL,
  weights = NULL,
  replicates = NULL,
  rep_corr = 0.9
)
```

Arguments

x	numeric vector containing abcissas of spectra to be corrected or matrix to be corrected with/without names colnames.
reference	numeric vector containing the reference spectrum.
degree	integer giving the polynomial degree of the baseline; 0 or higher, default = 2.

interferent	numeric vector containing a spectral component to remove.
constituent	numeric vector containing a spectral component to include.
weights	numeric vector of abcissas weights.
replicates	optional vector which identifies replicates. Default = NULL, meaning no replicate correctio will be performed.
rep_corr	proportion of variance or number of subspace components in replicate space (default = 0.9).

Value

An EMSC model is returned containing all parameters.

See Also

EMSC predict.EMSC plot.EMSC

---

fishoil *Raman spectra of fish oil*

Description

42 fish oil samples with 3 replicates each constitute a Raman data set of 126 samples over 3471 Raman shifts. There is a large fluorescence signal in the spectra that needs to be removed before any data analysis can be performed.

Usage

```
data(fishoil)
```

Format

A data.frame consisting of three parts. The estimated Iodine concentrations of the fish oil samples, Raman spectra (matrix) and a replicate vector.

References

K.H. Liland, A. Kohler, N.K. Afseth. Model-based pre-processing in Raman spectroscopy of biological samples. Journal of Raman Spectroscopy (2016).

milk — *Raman spectra of milk (wrong specification, kept for backward compatibility)*

Description

42 milk (should be fish oil) samples with 3 replicates each constitute a Raman data set of 126 samples over 3471 Raman shifts. There is a large fluorescence signal in the spectra that needs to be removed before any data analysis can be performed.

Usage

```
data(milk)
```

Format

A `data.frame` consisting of three parts. The estimated Iodine concentrations of the milk samples, Raman spectra (`matrix`) and a replicate vector.

References

K.H. Liland, A. Kohler, N.K. Afseth. Model-based pre-processing in Raman spectroscopy of biological samples. Journal of Raman Spectroscopy (2016).

orthogonalVectors — *Orthogonal vectors*

Description

Create orthogonal set of vectors that resemble the original input vectors.

Usage

```
orthogonalVectors(X, dim = 1, re = TRUE)
```

Arguments

`X`	a `matrix` with vectors as rows (default) or columns (see below).
`dim`	an integer specifying which dimension is the object dimension.
`re`	a logical indicating if vectors should be norm-scaled before orthogonalization and rescaled afterwards (default = TRUE).

Details

The input vectors are orthgonalized using singular value decomposition. To make the resulting vectors similar to the input vectors (not just any base for the same space) they are re-oriented towards the original vectors using Procrustes rotations.

To force the procedure to handle vectors of unequal magnitudes similarilly they are by default rescaled to norm vectors before orthogonalization and rescaled afterwards. This can be overridden using the `re` paramter.

See Also

`EMSC` `EMSC_model` `plot.EMSC`

`plot.EMSC` *Plot, print and summary methods for EMSC*

Description

Plotting routine for `EMSC` ojects. The default behaviour is to plot raw spectra, reference spectrum, polynomials, interferents, constituents, replicate model and corrected spectra. This can be tweaked by changing the parameters.

Usage

```
## S3 method for class 'EMSC'
plot(
  x,
  y,
  spec = "all",
  what = c("raw", "reference", "polynomials", "interferents", "constituents",
    "replicates", "corrected"),
  where = c(1, 1, 2, 3, 3, 4, 5),
  revX = FALSE,
  labels,
  type,
  lty,
  lwd = NULL,
  pch,
  cex = NULL,
  col,
  xlab,
  ylab,
  pretty.xlabels = TRUE,
  xlim,
  ...
)
```

```
## S3 method for class 'EMSC'
print(x, ...)

## S3 method for class 'EMSC'
summary(object, ...)
```

Arguments

x	An object fitted by the EMSC function.
y	Unused parameter to conform to generic plot.
spec	Parameter specifying if all spectra should be plotted (default) or a subset (numeric vector).
what	character vector defining what to plot.
where	integer vector defining which elements should be plotted in which subplot.
revX	Reverse x axis (default = FALSE).
labels	"names" or "numbers" uses column names for x axis labelling.
type	plotting type (line, points, ...).
lty	line type.
lwd	line width.
pch	plot character.
cex	symbol/line scaling.
col	symbol/line colour.
xlab	x label.
ylab	y label.
pretty.xlabels	Use pretty x labels (default = TRUE).
xlim	x limits.
...	Additional arguments to matplot.
object	An object fitted by the EMSC function.

Details

The parameters what and where must match so that the parts of the EMSC model end up in the correct subplot. There are limits to the freedom of this function.

print and summary return minimal information on the EMSC object.

Value

No return.

Author(s)

Kristian Hovde Liland

See Also

EMSC EMSC_model plot.EMSC

Examples

```
data(fishoil, package = "EMSC")
Raman      <- fishoil$Raman[, 850:3300]
EMSC.rep   <- EMSC(Raman, degree = 6, reference = Raman[30, ],
                   replicates = fishoil$replicates)
plot(EMSC.rep)
```

---

predict.EMSC *Predict Method for EMSC*

Description

Prediction for EMSC ojects. Corrections are calculated for the new matrix based on the EMSC model used in the input object.

Usage

```
## S3 method for class 'EMSC'
predict(object, newdata = NULL, ...)
```

Arguments

object	An object fitted by the EMSC function.
newdata	A matrix or object convertable to a matrix containing observations as rows.
...	unused.

See Also

EMSC EMSC_model

Examples

```
data(fishoil)
Raman.cal <- fishoil$Raman[  1:90,  850:3300]
Raman.val <- fishoil$Raman[-(1:90), 850:3300]
EMSC.cal  <- EMSC(Raman.cal)
EMSC.val  <- predict(EMSC.cal, Raman.val)
identical(EMSC.cal$model, EMSC.val$model) # Same model, reference spectrum, etc.

matplot(t(EMSC.cal$corrected), type = 'l', col = 'black', lty = 1, ylab = 'Intensity')
matplot(t(EMSC.val$corrected), type = 'l', col = 'red', lty = 2, add = TRUE)
legend('topleft', legend = c('Calibration','Validation'), lty = 1:2, col = 1:2)
```

SavitzkyGolay *Savitzky-Golay filtering and derivatives*

Description

Savitzky-Golay filtering and derivatives

Usage

```
SavitzkyGolay(
  X,
  poly = 3,
  width = 11,
  deriv = 2,
  ends = c("cut", "extrapolate", "zeros")
)
```

Arguments

X	matrix containing spectra as rows.
poly	Polynomial degree of smoother.
width	Window width of smoother, default = 11, must be an odd number.
deriv	Derivative degree, can be 0, default = 2.
ends	Handling of spectrum ends, i.e. first and last (width-1)/2 points. Default is "cut", i.e. remove ends, "extrapolate" copies the first/last estimable point, while "zeros" is included for backward compatilibity (fill with 0).

Value

A matrix of filtered spectra (possibly with derivatives)

Examples

```
data(fishoil)
Raman    <- fishoil$Raman[, 850:3300]
SavGol   <- SavitzkyGolay(Raman)
old.par  <- par(mfrow = c(2,1), mar = c(4,4,1,1))
matplot(colnames(Raman), t(Raman), type = 'l',
        ylab = 'Relative intensity', xlab = 'Raw spectra')
matplot(colnames(SavGol), t(SavGol), type = 'l',
        ylab = 'Relative intensity', xlab = 'Smoothed 2nd derivative')
par(old.par)
```

Index