Package 'alkahest'

July 26, 2024

Title Pre-Processing XY Data from Experimental Methods

Version 1.2.0

Maintainer Nicolas Frerebeau <nicolas.frerebeau@u-bordeaux-montaigne.fr>

Description A lightweight, dependency-free toolbox for pre-processing XY data from experimental methods (i.e. any signal that can be measured along a continuous variable). This package provides methods for baseline estimation and correction, smoothing, normalization, integration and peaks detection. Baseline correction methods includes polynomial fitting as described in Lieber and Mahadevan-Jansen (2003) <doi:10.1366/000370203322554518>, Rolling Ball algorithm after Kneen and Annegarn (1996) <doi:10.1016/0168-583X(95)00908-6>, SNIP algorithm after Ryan et al. (1988) <doi:10.1016/0168-583X(88)90063-8>, 4S Peak Filling after Liland (2015) <doi:10.1016/j.mex.2015.02.009> and more.

License GPL (>= 3)

URL https://packages.tesselle.org/alkahest/,
 https://github.com/tesselle/alkahest

BugReports https://github.com/tesselle/alkahest/issues

Depends R (>= 3.5.0)

Imports grDevices, methods, stats, utils

Suggests knitr, Matrix, markdown, tinytest

VignetteBuilder knitr

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Collate 'AllGenerics.R' 'alkahest-internal.R' 'alkahest-package.R' 'baseline.R' 'data.R' 'integrate.R' 'peaks.R' 'replace.R' 'resample.R' 'rescale.R' 'signal.R' 'smooth.R' 'windows.R' 'xrd.R'

NeedsCompilation no

2 Contents

Author Nicolas Frerebeau [aut, cre] (https://orcid.org/0000-0001-5759-4944	>,
Université Bordeaux Montaigne),	
Brice Lebrun [ctb] (https://orcid.org/0000-0001-7503-8685 >, Logo designer).	
Université Bordeaux Montaigne [fnd], CNRS [fnd]	

Repository CRAN

Date/Publication 2024-07-26 12:40:06 UTC

Contents

baseline_asls
baseline_linear
baseline_peakfilling
baseline_polynomial
baseline_rollingball
baseline_rubberband
baseline_snip
BEGe
integrate_rectangle
integrate_trapezoid
ka2_strip_penalized
LaBr
peaks_find
peaks_fwhm
Raman
replace_negative
replace_threshold
resample_bin
resample_down
resample_interpolate
rescale_area
rescale_range
rescale_snv
rescale_total
rescale_transform
signal_bind
signal_correct
signal_drift
signal_mean
signal_shift
smooth_likelihood
smooth_loess
smooth_rectangular
smooth_savitzky
smooth_triangular
smooth whittaker

baseline_asls 3

subset		 •						 •	•	•	 •	•		•	•	•		
window_sliding								 										
window_tumbling	g							 										
XRD								 										

Index 56

baseline_asls

Asymmetric Least Squares Smoothing

Description

Baseline estimation with asymmetric least squares smoothing.

Usage

```
baseline_asls(x, y, ...)
## S4 method for signature 'numeric,numeric'
baseline_asls(x, y, p = 0.01, lambda = 10^4, stop = 100)
## S4 method for signature 'ANY,missing'
baseline_asls(x, p = 0.01, lambda = 10^4, stop = 100)
```

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
	Currently not used.
p	A length-one numeric vector giving the asymmetry (0.001 $ is a good choice for a signal with positive peaks).$
lambda	A length-one numeric vector giving the smoothing parameter.
stop	An integer giving the stopping rule (i.e. maximum number of iterations).

Value

Returns a list with two components x and y.

Author(s)

P. H. C. Eilers and H. F. M. Boelens (original Matlab code)

References

Eilers, P. H. C. & Boelens, H. F. M. (2005). *Baseline Correction with Asymmetric Least Squares Smoothing*.

4 baseline_linear

See Also

```
signal_correct()
```

Other baseline estimation methods: baseline_linear(), baseline_peakfilling(), baseline_polynomial(), baseline_rollingball(), baseline_rubberband(), baseline_snip()

Examples

```
## X-ray diffraction
data("XRD")

## Subset from 20 to 70 degrees
XRD <- signal_select(XRD, from = 20, to = 70)

## Plot spectrum
plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")

## Polynomial baseline
baseline <- baseline_asls(XRD, p = 0.005, lambda = 10^7)

lines(baseline, type = "1", col = "red")</pre>
```

baseline_linear

Linear Baseline Estimation

Description

Linear Baseline Estimation

Usage

```
baseline_linear(x, y, ...)
## S4 method for signature 'numeric,numeric'
baseline_linear(x, y, points = range(x))
## S4 method for signature 'ANY,missing'
baseline_linear(x, points = range(x))
```

Arguments

x, y A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).

... Currently not used.

points A numeric vector specifying the data points to be used in the fitting process (in x unit).

baseline_peakfilling 5

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

See Also

```
signal_correct()
```

Other baseline estimation methods: baseline_asls(), baseline_peakfilling(), baseline_polynomial(), baseline_rollingball(), baseline_rubberband(), baseline_snip()

Examples

```
## X-ray diffraction
data("XRD")

## Plot spectrum
plot(XRD, type = "l", xlab = expression(2*theta), ylab = "Count")

## Linear baseline
baseline <- baseline_linear(XRD, points = c(25, 34))

plot(XRD, type = "l", xlab = expression(2*theta), ylab = "Count")
lines(baseline, type = "l", col = "red")

## Correct baseline
XRD$count <- XRD$count - baseline$y

plot(XRD, type = "l", xlab = expression(2*theta), ylab = "Count")</pre>
```

baseline_peakfilling 4S Peak Filling

Description

Baseline estimation by iterative mean suppression.

Usage

```
baseline_peakfilling(x, y, ...)
## S4 method for signature 'numeric,numeric'
baseline_peakfilling(x, y, n, m, by = 10, lambda = 1600, d = 2, sparse = FALSE)
## S4 method for signature 'ANY,missing'
baseline_peakfilling(x, n, m, by = 10, lambda = 1600, d = 2, sparse = FALSE)
```

6 baseline_peakfilling

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
	Currently not used.
n	An integer value giving the number of iterations.
m	An odd integer giving the half window size.
by	A length-one numeric vector giving the umber of buckets to divide x into.
lambda	An integer giving the smoothing parameter. The larger lambda is, the smoother the curve (see smooth_whittaker()).
d	An integer specifying the order of the penalty (see smooth_whittaker()).
sparse	A logical scalar: should sparse matrices be used for computation (see smooth_whittaker())? If TRUE, Matrix is required.

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

References

Liland, K. H. (2015). 4S Peak Filling - baseline estimation by iterative mean suppression. *MethodsX*, 2, 135-140. doi:10.1016/j.mex.2015.02.009.

See Also

```
signal_correct(), smooth_whittaker()
Other baseline estimation methods: baseline_asls(), baseline_linear(), baseline_polynomial(),
baseline_rollingball(), baseline_rubberband(), baseline_snip()
```

```
## X-ray diffraction
data("XRD")

## 4S Peak Filling baseline
baseline <- baseline_peakfilling(XRD, n = 10, m = 5, by = 10, sparse = TRUE)

plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")
lines(baseline, type = "1", col = "red")</pre>
```

baseline_polynomial 7

baseline_polynomial Polynomial Baseline Estimation

Description

Polynomial Baseline Estimation

Usage

```
baseline_polynomial(x, y, ...)
## S4 method for signature 'numeric, numeric'
baseline_polynomial(x, y, d = 3, tolerance = 0.001, stop = 100)
## S4 method for signature 'ANY, missing'
baseline_polynomial(x, d = 3, tolerance = 0.001, stop = 100)
```

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
• • •	Currently not used.
d	An integer giving the degree of the polynomial. Must be less than the number of unique points.
tolerance	A numeric scalar giving the tolerance of difference between iterations.
stop	An integer giving the stopping rule (i.e. maximum number of iterations).

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

References

Lieber, C. A. and Mahadevan-Jansen, A. (2003). Automated Method for Subtraction of Fluorescence from Biological Raman Spectra. *Applied Spectroscopy*, 57(11): 1363-67. doi:10.1366/000370203322554518.

See Also

```
signal_correct()
```

Other baseline estimation methods: baseline_asls(), baseline_linear(), baseline_peakfilling(), baseline_rollingball(), baseline_rubberband(), baseline_snip()

8 baseline_rollingball

Examples

```
## X-ray diffraction
data("XRD")

## Subset from 20 to 70 degrees

XRD <- signal_select(XRD, from = 20, to = 70)

## Plot spectrum
plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")

## Polynomial baseline
baseline <- baseline_polynomial(XRD, d = 4, tolerance = 0.02, stop = 1000)

plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")
lines(baseline, type = "1", col = "red")</pre>
```

baseline_rollingball Rolling Ball Baseline Estimation

Description

Rolling Ball Baseline Estimation

Usage

```
baseline_rollingball(x, y, ...)
## S4 method for signature 'numeric,numeric'
baseline_rollingball(x, y, m, s)
## S4 method for signature 'ANY,missing'
baseline_rollingball(x, m, s)
```

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
	Currently not used.
m	An odd integer giving the window size (i.e. the number of adjacent points to be used; see window_sliding()) for minimization/maximization.
S	An odd integer giving the window size (i.e. the number of adjacent points to be used; see window_sliding()) for smoothing.

Value

Returns a list with two components x and y.

baseline_rubberband 9

Note

There will be (m-1)/2 points both at the beginning and at the end of the data series for which a complete m-width window cannot be obtained. To prevent data loss, progressively wider/narrower windows are used at both ends of the data series.

Author(s)

N. Frerebeau

References

Kneen, M. A. and Annegarn, H. J. (1996). Algorithm for Fitting XRF, SEM and PIXE X-Ray Spectra Backgrounds. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, 109/110: 209-213. doi:10.1016/0168583X(95)009086.

See Also

```
signal_correct()
```

Other baseline estimation methods: baseline_asls(), baseline_linear(), baseline_peakfilling(), baseline_polynomial(), baseline_rubberband(), baseline_snip()

Examples

```
## X-ray diffraction
data("XRD")

## Subset from 20 to 70 degrees

XRD <- signal_select(XRD, from = 20, to = 70)

## Plot spectrum
plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")

## Rolling Ball baseline
baseline <- baseline_rollingball(XRD, m = 201, s = 151)

plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")
lines(baseline, type = "1", col = "red")</pre>
```

baseline_rubberband

Rubberband Baseline Estimation

Description

Rubberband Baseline Estimation

10 baseline_rubberband

Usage

```
baseline_rubberband(x, y, ...)
## S4 method for signature 'numeric,numeric'
baseline_rubberband(x, y, noise = 0, spline = TRUE, ...)
## S4 method for signature 'ANY,missing'
baseline_rubberband(x, noise = 0, spline = TRUE, ...)
```

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
• • •	Extra arguments to be passed to stats::smooth.spline().
noise	A length-one numeric vector giving the noise level. Only used if method is "rubberband".
spline	A logical scalar: should spline interpolation through the support points be used instead of linear interpolation? Only used if method is "rubberband".

Details

A convex envelope of the spectrum is determined and the baseline is estimated as the part of the convex envelope lying below the spectrum. Note that the rubber band does not enter the concave regions (if any) of the spectrum.

Value

Returns a list with two components x and y.

Note

baseline_rubberband() is slightly modified from C. Beleites' hyperSpec::spc.rubberband().

Author(s)

N. Frerebeau

See Also

```
signal_correct()
```

```
Other baseline estimation methods: baseline_asls(), baseline_linear(), baseline_peakfilling(), baseline_polynomial(), baseline_rollingball(), baseline_snip()
```

baseline_snip 11

Examples

```
## gamma-ray spectrometry
data("BEGe")

## Subset from 2.75 to 200 keV

BEGe <- signal_select(BEGe, from = 3, to = 200)

## Plot spectrum
plot(BEGe, type = "1", xlab = "Energy (keV)", ylab = "Count")

## Rubberband baseline
baseline <- baseline_rubberband(BEGe)

plot(BEGe, type = "1", xlab = "Energy (keV)", ylab = "Count")
lines(baseline, type = "1", col = "red")</pre>
```

baseline_snip

SNIP Baseline Estimation

Description

Sensitive Nonlinear Iterative Peak clipping algorithm.

Usage

```
baseline_snip(x, y, ...)
## S4 method for signature 'numeric,numeric'
baseline_snip(x, y, LLS = FALSE, decreasing = FALSE, n = 100)
## S4 method for signature 'ANY,missing'
baseline_snip(x, LLS = FALSE, decreasing = FALSE, n = 100)
```

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
	Currently not used.
LLS	A logical scalar: should the LLS operator be applied on x before employing SNIP algorithm? Only used if method is "SNIP".
decreasing	A logical scalar: should a decreasing clipping window be used?
n	An integer value giving the number of iterations.

Value

Returns a list with two components x and y.

12 BEGe

Author(s)

N. Frerebeau

References

Morháč, M., Kliman, J., Matoušek, V., Veselský, M. & Turzo, I. (1997). Background elimination methods for multidimensional gamma-ray spectra. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 401(1), p. 113-132. doi:10.1016/S01689002(97)010231

Morháč, M. & Matoušek, V. (2008). Peak Clipping Algorithms for Background Estimation in Spectroscopic Data. *Applied Spectroscopy*, 62(1), p. 91-106. doi:10.1366/000370208783412762

Ryan, C. G., Clayton, E., Griffin, W. L., Sie, S. H. & Cousens, D. R. (1988). SNIP, a statistics-sensitive background treatment for the quantitative analysis of PIXE spectra in geoscience applications. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, 34(3), p. 396-402. doi:10.1016/0168583X(88)900638

See Also

```
signal_correct()
```

Other baseline estimation methods: baseline_asls(), baseline_linear(), baseline_peakfilling(), baseline_polynomial(), baseline_rollingball(), baseline_rubberband()

Examples

```
## gamma-ray spectrometry
data("BEGe")

## Subset from 2.75 to 200 keV
BEGe <- signal_select(BEGe, from = 3, to = 200)

## Plot spectrum
plot(BEGe, type = "1", xlab = "Energy (keV)", ylab = "Count")

## SNIP baseline
baseline <- baseline_snip(BEGe, LLS = FALSE, decreasing = FALSE, n = 100)

plot(BEGe, type = "1", xlab = "Energy (keV)", ylab = "Count")
lines(baseline, type = "1", col = "red")</pre>
```

BEGe

Gamma-Ray Spectrometry

Description

Gamma-Ray Spectrometry

integrate_rectangle 13

Usage

BEGe

Format

```
A data.frame with 8192 \text{ rows (channels)} and 2 \text{ variables.} energy (keV) count
```

See Also

```
Other datasets: LaBr, Raman, XRD
```

Examples

```
data("BEGe")
plot(BEGe, type = "1", xlab = "Energy (keV)", ylab = "Count")
```

Description

Approximates the definite integral by using the rectangle rule.

Usage

```
integrate_rectangle(x, y, ...)
## S4 method for signature 'numeric,numeric'
integrate_rectangle(x, y, right = FALSE)
## S4 method for signature 'ANY,missing'
integrate_rectangle(x, right = FALSE)
```

Arguments

```
x, y A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
... Currently not used.
right A logical scalar: should the right rule be used instead of the left rule?
```

Value

Returns a list with two components x and y.

14 integrate_trapezoid

Author(s)

N. Frerebeau

See Also

Other integration methods: integrate_trapezoid()

Examples

```
## Calculate the area under the sine curve from 0 to pi
# integrate(f = function(x) x^3, lower = 0, upper = 2)
x <- seq(0, 2, len = 101)
y <- x^3

plot(x, y, type = "l")

integrate_rectangle(x, y, right = FALSE) # 3.9204
integrate_rectangle(x, y, right = TRUE) # 4.0804
integrate_trapezoid(x, y) # 4.0004</pre>
```

integrate_trapezoid

Trapezoidal Rule

Description

Approximates the definite integral by using the trapezoidal rule.

Usage

```
integrate_trapezoid(x, y, ...)
## S4 method for signature 'numeric, numeric'
integrate_trapezoid(x, y)
## S4 method for signature 'ANY, missing'
integrate_trapezoid(x)
```

Arguments

```
x, y A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).... Currently not used.
```

Value

Returns a list with two components x and y.

ka2_strip_penalized 15

Author(s)

N. Frerebeau

See Also

Other integration methods: integrate_rectangle()

Examples

```
## Calculate the area under the sine curve from 0 to pi
# integrate(f = function(x) x^3, lower = 0, upper = 2)
x <- seq(0, 2, len = 101)
y <- x^3

plot(x, y, type = "l")

integrate_rectangle(x, y, right = FALSE) # 3.9204
integrate_rectangle(x, y, right = TRUE) # 4.0804
integrate_trapezoid(x, y) # 4.0004</pre>
```

ka2_strip_penalized

Strip XRD ka2

Description

Strip XRD ka2

Usage

```
ka2_strip_penalized(x, y, ...)
## S4 method for signature 'numeric, numeric'
ka2_strip_penalized(
 Х,
 у,
 lambda,
 wave = c(1.5406, 1.54443),
  tau = 0.5,
  nseg = 1,
  progress = interactive()
)
## S4 method for signature 'ANY, missing'
ka2_strip_penalized(
  Х,
  lambda,
 wave = c(1.5406, 1.54443),
  tau = 0.5,
```

16 ka2_strip_penalized

```
nseg = 1,
progress = interactive()
)
```

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
	Currently not used.
lambda	An integer giving the smoothing parameter. The larger lambda is, the smoother the curve.
wave	A length-two numeric vector giving the characteristic wavelengths of the anode material (defaults to copper).
tau	A length-one numeric vector giving the ratio between $\alpha 1$ and $\alpha 2$ line intensities (defaults to 1/2).
nseg	A length-one numeric vector specifying the number of equally sized segments for B-spline basis matrix computation.
progress	A logical scalar: should a progress bar be displayed?

Value

Returns a list with two components x and y.

Note

Matrix is required.

Author(s)

J. J. de Rooi et al. (original R code).

References

de Rooi, J. J., van der Pers, N. M., Hendrikx, R. W. A., Delhez, R., Böttger A. J. and Eilers, P. H. C. (2014). Smoothing of X-ray diffraction data and Ka2 elimination using penalized likelihood and the composite link model. *Journal of Applied Crystallography*, 47: 852-860. doi:10.1107/S1600576714005809

```
## Not run:
## X-ray diffraction
data("XRD")

## Subset from 20 to 40 degrees

XRD <- signal_select(XRD, from = 20, to = 40)

## Plot diffractogram
plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")</pre>
```

LaBr 17

```
## Penalized likelihood smoothing
lambda <- seq(from = 1, to = 8, length.out = 40)
lambda <- 10^lambda

likelihood <- smooth_likelihood(XRD, lambda = lambda, d = 3)
lines(likelihood, col = "red")

## Strip ka2
ka2 <- ka2_strip_penalized(XRD, lambda = lambda, tau = 0.5, nseg = 1)
lines(ka2, col = "blue")

## End(Not run)</pre>
```

LaBr

Gamma-Ray Spectrometry

Description

Gamma-Ray Spectrometry

Usage

LaBr

Format

```
A data.frame with 1024 rows (channels) and 2 variables.
```

```
energy (keV)
```

count

See Also

```
Other datasets: BEGe, Raman, XRD
```

```
data("LaBr")
plot(LaBr, type = "1", xlab = "Energy (keV)", ylab = "Count")
```

18 peaks_find

peaks_find	Find Peaks		
------------	------------	--	--

Description

Finds local maxima in sequential data.

Usage

```
peaks_find(x, y, ...)
## S4 method for signature 'numeric,numeric'
peaks_find(x, y, method = "MAD", SNR = 2, m = NULL, ...)
## S4 method for signature 'ANY,missing'
peaks_find(x, method = "MAD", SNR = 2, m = NULL, ...)
```

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
	Extra parameters to be passed to internal methods.
method	A character string specifying the method to be used for background noise estimation (see below).
SNR	An integer giving the signal-to-noise-ratio for peak detection (see below).
m	An odd integer giving the window size (i.e. the number of adjacent points to be used). If NULL, 5% of the data points is used as the half window size.

Details

A local maximum has to be the highest one in the given window and has to be higher than $SNR \times noise$ to be recognized as peak.

The following methods are available for noise estimation:

MAD Median Absolute Deviation.

Note that to improve peak detection, it may be helpful to smooth the data and remove the baseline beforehand.

Value

Returns a list with two components x and y.

peaks_fwhm 19

Note

There will be (m-1)/2 points both at the beginning and at the end of the data series for which a complete m-width window cannot be obtained. To prevent data loss, progressively wider/narrower windows are used at both ends of the data series.

Adapted from Stasia Grinberg's findPeaks function.

Author(s)

N. Frerebeau

See Also

Other peaks detection methods: peaks_fwhm()

Examples

```
## X-ray diffraction
data("XRD")
## 4S Peak Filling baseline
baseline <- baseline_peakfilling(XRD, n = 10, m = 5, by = 10, sparse = TRUE)
plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")
lines(baseline, type = "l", col = "red")
## Correct baseline
XRD <- signal_drift(XRD, lag = baseline, subtract = TRUE)</pre>
## Find peaks
peaks <- peaks_find(XRD, SNR = 3, m = 11)</pre>
plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")
lines(peaks, type = "p", pch = 16, col = "red")
abline(h = attr(peaks, "noise"), lty = 2) # noise threshold
## Half-Width at Half-Maximum
x <- seq(-4, 4, length = 1000)
y \leftarrow dnorm(x)
peaks_fwhm(x, y, center = 0) # Expected: 2 * sqrt(2 * log(2))
```

peaks_fwhm

Half-Width at Half-Maximum

Description

Estimates the Half-Width at Half-Maximum (FWHM) for a given peak.

20 peaks_fwhm

Usage

```
peaks_fwhm(x, y, ...)
## S4 method for signature 'numeric,numeric'
peaks_fwhm(x, y, center)
## S4 method for signature 'ANY,missing'
peaks_fwhm(x, center)
```

Arguments

x, y A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
... Currently not used.
center A numeric value giving the peak position in x units.

Details

It tries to get the smallest possible estimate.

Value

A numeric value.

Author(s)

N. Frerebeau

See Also

Other peaks detection methods: peaks_find()

```
## X-ray diffraction
data("XRD")

## 4S Peak Filling baseline
baseline <- baseline_peakfilling(XRD, n = 10, m = 5, by = 10, sparse = TRUE)

plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")
lines(baseline, type = "1", col = "red")

## Correct baseline
XRD <- signal_drift(XRD, lag = baseline, subtract = TRUE)

## Find peaks
peaks <- peaks_find(XRD, SNR = 3, m = 11)

plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")</pre>
```

Raman 21

```
lines(peaks, type = "p", pch = 16, col = "red")
abline(h = attr(peaks, "noise"), lty = 2) # noise threshold

## Half-Width at Half-Maximum
x <- seq(-4, 4, length = 1000)
y <- dnorm(x)

peaks_fwhm(x, y, center = 0) # Expected: 2 * sqrt(2 * log(2))</pre>
```

Raman

Raman Spectroscopy

Description

Raman Spectroscopy

Usage

Raman

Format

A data. frame with 1182 rows and 2 variables.

shift Raman shift.

intensity

See Also

```
Other datasets: BEGe, LaBr, XRD
```

```
data("Raman")
plot(Raman, type = "1", xlab = "Shift", ylab = "Intensity")
```

22 replace_negative

replace_negative

Replace Negative Values

Description

Replace Negative Values

Usage

```
replace_negative(x, y, ...)
## S4 method for signature 'numeric,numeric'
replace_negative(x, y, value = 0)
## S4 method for signature 'ANY,missing'
replace_negative(x, value = 0)
```

Arguments

x, y
A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).

Extra parameters to be passed to threshold.
value
A numeric value to replace negative values.

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

See Also

Other replacement methods: replace_threshold()

replace_threshold 23

replace_threshold

Replace Values Below a Given Threshold

Description

Replace Values Below a Given Threshold

Usage

```
replace_threshold(x, y, threshold, ...)
## S4 method for signature 'numeric,numeric,function'
replace_threshold(x, y, threshold, value = 0, ...)
## S4 method for signature 'ANY,missing,function'
replace_threshold(x, threshold, value = 0, ...)
## S4 method for signature 'numeric,numeric,numeric'
replace_threshold(x, y, threshold, value = 0, ...)
## S4 method for signature 'ANY,missing,numeric'
replace_threshold(x, threshold, value = 0, ...)
```

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
threshold	A numeric value or a function that takes a numeric vector as argument and returns a single numeric value.
	Extra parameters to be passed to threshold.
value	A numeric value to replace values below threshold.

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

See Also

Other replacement methods: replace_negative()

24 resample_bin

resample_bin

Bin

Description

Averages x values and applies a function to the corresponding y values.

Usage

```
resample_bin(x, y, ...)
## S4 method for signature 'numeric,numeric'
resample_bin(x, y, by, f = mean, ...)
## S4 method for signature 'ANY,missing'
resample_bin(x, y, by, f = sum)
```

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
	Extra parameters to be passed to f.
by	An integer specifying the binning ratio (i.e. the number of points to be grouped together; see window_tumbling()).
f	A function that takes a numeric vector of intensities as argument and returns

a single numeric vector. Used to estimate the local representative value in each bin (defaults to sum(); see examples).

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

See Also

Other resampling methods: resample_down(), resample_interpolate()

```
## X-ray diffraction
data("XRD")

## Plot spectrum
plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")
```

resample_down 25

```
## Bin by 3
XRD_bin_mean <- resample_bin(XRD, by = 3, f = mean)</pre>
XRD_bin_min <- resample_bin(XRD, by = 3, f = min)</pre>
plot(XRD, type = "l", xlim = c(25, 35),
     xlab = expression(2*theta), ylab = "Count")
lines(XRD_bin_mean, type = "1", col = "red")
lines(XRD_bin_min, type = "1", col = "green")
## Downsample by 10
XRD_down <- resample_down(XRD, by = 10)</pre>
plot(XRD, type = "l", xlim = c(20, 40),
     xlab = expression(2*theta), ylab = "Count")
lines(XRD_down, type = "1", col = "red")
## Linearly interpolate
XRD_approx <- resample_interpolate(XRD, from = 20, to = 40, by = 0.02)</pre>
plot(XRD, type = "l", xlim = c(20, 40),
     xlab = expression(2*theta), ylab = "Count")
lines(XRD_approx, type = "1", col = "red")
```

resample_down

Downsample

Description

Downsample

Usage

```
resample_down(x, y, ...)
## S4 method for signature 'numeric,numeric'
resample_down(x, y, by)
## S4 method for signature 'ANY,missing'
resample_down(x, y, by)
```

Arguments

x, y A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
... Currently not used.
by An integer specifying the downsampling factor.

26 resample_interpolate

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

See Also

Other resampling methods: resample_bin(), resample_interpolate()

Examples

```
## X-ray diffraction
data("XRD")
## Plot spectrum
plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")
## Bin by 3
XRD_bin_mean <- resample_bin(XRD, by = 3, f = mean)</pre>
XRD_bin_min <- resample_bin(XRD, by = 3, f = min)</pre>
plot(XRD, type = "l", xlim = c(25, 35),
     xlab = expression(2*theta), ylab = "Count")
lines(XRD_bin_mean, type = "1", col = "red")
lines(XRD_bin_min, type = "1", col = "green")
## Downsample by 10
XRD_down <- resample_down(XRD, by = 10)</pre>
plot(XRD, type = "l", xlim = c(20, 40),
     xlab = expression(2*theta), ylab = "Count")
lines(XRD_down, type = "1", col = "red")
## Linearly interpolate
XRD_approx \leftarrow resample_interpolate(XRD, from = 20, to = 40, by = 0.02)
plot(XRD, type = "l", xlim = c(20, 40),
     xlab = expression(2*theta), ylab = "Count")
lines(XRD_approx, type = "1", col = "red")
```

Description

Linearly Interpolate

resample_interpolate 27

Usage

```
resample_interpolate(x, y, ...)
## S4 method for signature 'numeric, numeric'
resample_interpolate(x, y, from, to, by, ...)
## S4 method for signature 'ANY, missing'
resample_interpolate(x, y, from, to, by, ...)
```

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
	Extra arguments to be passed to stats::approx().
from	A length-one numeric vector giving the starting value of the sequence where interpolation is to take place.
to	A length-one numeric vector giving the end value of the sequence where interpolation is to take place.
by	A length-one numeric vector specifying the increment of the sequence.

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

See Also

Other resampling methods: resample_bin(), resample_down()

28 rescale_area

rescale_area

Normalize intensities by AUC

Description

Rescales intensities so that the area under the curve (AUC) is equal to 1.

Usage

```
rescale_area(x, y, ...)
## S4 method for signature 'numeric, numeric'
rescale_area(x, y, method = c("rectangle", "trapezoid"), ...)
## S4 method for signature 'ANY, missing'
rescale_area(x, method = c("rectangle", "trapezoid"), ...)
```

Arguments

x, y A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
 ... Currently not used.
 method A character string specifying the method for integration. It must be one of "rectangle" or "trapezoid". Any unambiguous substring can be given.

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

rescale_range 29

See Also

Other normalization methods: rescale_range(), rescale_snv(), rescale_total(), rescale_transform()

Examples

```
## gamma-ray spectrometry
data("BEGe")
## Subset from 2.75 to 200 keV
BEGe <- signal_select(BEGe, from = 3, to = 200)
## Plot spectrum
plot(BEGe, type = "1", xlab = "Energy (keV)", ylab = "Count")
## Normalize by area under the curve
BEGe_area <- rescale_area(BEGe)</pre>
plot(BEGe_area, type = "1", xlab = "Energy (keV)", ylab = "Count")
integrate_rectangle(BEGe)
integrate_rectangle(BEGe_area)
## Rescale so that intensities sum to 1
BEGe_total <- rescale_total(BEGe, total = 1)</pre>
plot(BEGe_total, type = "1", xlab = "Energy (keV)", ylab = "Count")
## Rescale intensities to 0-1
BEGe_range <- rescale_range(BEGe, min = 0, max = 1)
plot(BEGe_range, type = "1", xlab = "Energy (keV)", ylab = "Count")
```

rescale_range

Rescales intensities to have specified minimum and maximum

Description

Rescales intensities to have specified minimum and maximum.

Usage

```
rescale_range(x, y, ...)
rescale_min(x, y, ...)
rescale_max(x, y, ...)
## S4 method for signature 'numeric,numeric'
rescale_range(x, y, min = 0, max = 1)
## S4 method for signature 'ANY,missing'
rescale_range(x, min = 0, max = 1)
```

30 rescale_range

```
## S4 method for signature 'numeric,numeric'
rescale_min(x, y, min = 0)

## S4 method for signature 'ANY,missing'
rescale_min(x, min = 0)

## S4 method for signature 'numeric,numeric'
rescale_max(x, y, max = 1)

## S4 method for signature 'ANY,missing'
rescale_max(x, max = 1)
```

Arguments

x, y A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
 ... Currently not used.
 min A legnth-one numeric vector specifying the output minimum.
 max A legnth-one numeric vector specifying the output maximum.

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

See Also

Other normalization methods: rescale_area(), rescale_snv(), rescale_total(), rescale_transform()

```
## gamma-ray spectrometry
data("BEGe")

## Subset from 2.75 to 200 keV
BEGe <- signal_select(BEGe, from = 3, to = 200)

## Plot spectrum
plot(BEGe, type = "1", xlab = "Energy (keV)", ylab = "Count")

## Normalize by area under the curve
BEGe_area <- rescale_area(BEGe)
plot(BEGe_area, type = "1", xlab = "Energy (keV)", ylab = "Count")
integrate_rectangle(BEGe)
integrate_rectangle(BEGe)
integrate_rectangle(BEGe_area)</pre>
```

rescale_snv 31

```
## Rescale so that intensities sum to 1
BEGe_total <- rescale_total(BEGe, total = 1)
plot(BEGe_total, type = "1", xlab = "Energy (keV)", ylab = "Count")
## Rescale intensities to 0-1
BEGe_range <- rescale_range(BEGe, min = 0, max = 1)
plot(BEGe_range, type = "1", xlab = "Energy (keV)", ylab = "Count")</pre>
```

rescale_snv

Standard Normal Variate (SNV) Transformation

Description

Subtracts the mean and scales to unit variance.

Usage

```
rescale_snv(x, y, ...)
## S4 method for signature 'numeric,numeric'
rescale_snv(x, y, ...)
## S4 method for signature 'ANY,missing'
rescale_snv(x, y, ...)
```

Arguments

x, y A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).... Currently not used.

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

References

Barnes, R. J., Dhanoa, M. S. & Lister, S. J. (1989). Standard Normal Variate Transformation and De-Trending of Near-Infrared Diffuse Reflectance Spectra. *Applied Spectroscopy*, 43(5): 772-777. doi:10.1366/0003702894202201.

See Also

Other normalization methods: rescale_area(), rescale_range(), rescale_total(), rescale_transform()

32 rescale_total

Examples

```
## Raman spectrometry
data("Raman")

## Subset from 200 to 800 1/cm
Raman <- signal_select(Raman, from = 200, to = 800)

## Plot spectrum
plot(Raman, type = "1", xlab = "Raman shift", ylab = "Intensity")

## Normalize SNV
Raman_snv <- rescale_snv(Raman)
plot(Raman_snv, type = "1", xlab = "Raman shift", ylab = "Intensity")</pre>
```

rescale_total

Rescale intensities to sum to a specified value

Description

Rescales intensities to sum to a specified value.

Usage

```
rescale_total(x, y, ...)
## S4 method for signature 'numeric, numeric'
rescale_total(x, y, total = 1)
## S4 method for signature 'ANY, missing'
rescale_total(x, total = 1)
```

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
	Currently not used.
total	A legnth-one numeric vector specifying the output total. Defaults to 1, i.e. normalizes by total intensity.

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

rescale_transform 33

See Also

Other normalization methods: rescale_area(), rescale_range(), rescale_snv(), rescale_transform()

Examples

```
## gamma-ray spectrometry
data("BEGe")
## Subset from 2.75 to 200 keV
BEGe <- signal_select(BEGe, from = 3, to = 200)
## Plot spectrum
plot(BEGe, type = "1", xlab = "Energy (keV)", ylab = "Count")
## Normalize by area under the curve
BEGe_area <- rescale_area(BEGe)</pre>
plot(BEGe_area, type = "1", xlab = "Energy (keV)", ylab = "Count")
integrate_rectangle(BEGe)
integrate_rectangle(BEGe_area)
## Rescale so that intensities sum to 1
BEGe_total <- rescale_total(BEGe, total = 1)</pre>
plot(BEGe_total, type = "1", xlab = "Energy (keV)", ylab = "Count")
## Rescale intensities to 0-1
BEGe_range <- rescale_range(BEGe, min = 0, max = 1)</pre>
plot(BEGe_range, type = "l", xlab = "Energy (keV)", ylab = "Count")
```

rescale_transform

Transform Intensities

Description

Transform Intensities

Usage

```
rescale_transform(x, y, ...)
## S4 method for signature 'numeric, numeric'
rescale_transform(x, y, f, ...)
## S4 method for signature 'ANY, missing'
rescale_transform(x, f, ...)
```

34 signal_bind

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
	Extra arguments to be passed to f.
f	A function that takes a numeric vector of intensities as argument and returns a numeric vector.

Details

Transformation of intensities can be used to improve the identification of peaks with a low signal-to-noise ratio.

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

See Also

Other normalization methods: rescale_area(), rescale_range(), rescale_snv(), rescale_total()

Examples

```
## gamma-ray spectrometry
data("BEGe")

## Subset from 2.75 to 200 keV

BEGe <- signal_select(BEGe, from = 3, to = 200)

## Plot spectrum
plot(BEGe, type = "1", xlab = "Energy (keV)", ylab = "Count")

## Transform intensities

BEGe_trans <- rescale_transform(BEGe, f = sqrt)
plot(BEGe_trans, type = "1", xlab = "Energy (keV)", ylab = "sqrt(Count)")</pre>
```

signal_bind

Bind

Description

Combines XY objects.

signal_bind 35

Usage

```
signal_bind(...)
## S4 method for signature 'ANY'
signal_bind(...)
```

Arguments

... Any object that can be interpreted in a suitable way (see grDevices::xy.coords()).

Value

Returns a matrix of intensities.

Author(s)

N. Frerebeau

See Also

```
Other signal processing methods: signal_correct(), signal_drift(), signal_mean(), signal_shift(), subset()
```

36 signal_correct

signal_correct

Baseline Correction

Description

Baseline Correction

Usage

```
signal_correct(x, y, ...)

## S4 method for signature 'numeric,numeric'
signal_correct(
    x,
    y,
    method = c("linear", "polynomial", "asls", "rollingball", "rubberband", "SNIP", "4S"),
    ...
)

## S4 method for signature 'ANY,missing'
signal_correct(
    x,
    method = c("linear", "polynomial", "asls", "rollingball", "rubberband", "SNIP", "4S"),
    ...
)
```

Arguments

x, y A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
... Extra arguments to be passed to baseline_*() (see details).
method A character string specifying the method for baseline estimation (see details).
Any unambiguous substring can be given.

Details

Available methods for baseline estimation:

```
asls Asymmetric Least Squares Smoothing (see baseline_asls()).

linear Linear baseline estimation (see baseline_linear()).

polynomial Polynomial baseline estimation (see baseline_polynomial()).

rollingball Rolling ball baseline estimation (see baseline_rollingball()).

rubberband Rubberband baseline estimation (see baseline_rubberband()).

SNIP Sensitive Nonlinear Iterative Peak clipping algorithm (see baseline_snip()).

4S 4S Peak Filling (see baseline_peakfilling()).
```

signal_drift 37

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

See Also

```
Other signal processing methods: signal_bind(), signal_drift(), signal_mean(), signal_shift(), subset()
```

Examples

```
## gamma-ray spectrometry
data("BEGe")

## Subset from 2.75 to 200 keV
BEGe <- signal_select(BEGe, from = 3, to = 200)

## Drift
baseline <- baseline_snip(BEGe)
BEGe_drif <- signal_drift(BEGe, lag = baseline, subtract = TRUE)

plot(BEGe, type = "1", xlab = "Energy (keV)", ylab = "Count")
lines(BEGe_drif, type = "1", col = "red")

## Correct
BEGe_corr <- signal_correct(BEGe, method = "SNIP")

plot(BEGe, type = "1", xlab = "Energy (keV)", ylab = "Count")
lines(BEGe_corr, type = "1", col = "red")</pre>
```

signal_drift

Drift Intensities

Description

Drift Intensities

Usage

```
signal_drift(x, y, lag, ...)
## S4 method for signature 'numeric, numeric, numeric'
signal_drift(x, y, lag)
## S4 method for signature 'ANY, missing, ANY'
signal_drift(x, lag, subtract = FALSE)
```

38 signal_mean

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
lag	A numeric vector specifying the offset or any object that can be interpreted in a suitable way (see grDevices::xy.coords())
	Currently not used.
subtract	A logical scalar: should lag be subtracted to y?

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

See Also

```
Other signal processing methods: signal_bind(), signal_correct(), signal_mean(), signal_shift(), subset()
```

Examples

```
## gamma-ray spectrometry
data("BEGe")

## Subset from 2.75 to 200 keV
BEGe <- signal_select(BEGe, from = 3, to = 200)

## Drift
BEGe_plus <- signal_drift(BEGe, lag = 250)
BEGe_minus <- signal_drift(BEGe, lag = 250, subtract = TRUE)

plot(BEGe, type = "l", xlab = "Energy (keV)", ylab = "Count")
lines(BEGe_plus, type = "l", col = "red")
lines(BEGe_minus, type = "l", col = "green")</pre>
```

signal_mean

Mean Intensities

Description

Mean Intensities

signal_mean 39

Usage

```
signal_mean(...)
## S4 method for signature 'ANY'
signal_mean(...)
```

Arguments

... Any object that can be interpreted in a suitable way (see grDevices::xy.coords()).

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

See Also

```
Other signal processing methods: signal_bind(), signal_correct(), signal_drift(), signal_shift(), subset()
```

40 signal_shift

signal_shift

Shift the X Scale

Description

Shifts the x scale by a given value.

Usage

```
signal_shift(x, y, lag, ...)
## S4 method for signature 'numeric,numeric'
signal_shift(x, y, lag)
## S4 method for signature 'ANY,missing'
signal_shift(x, lag)
```

Arguments

x, y A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
 lag A numeric vector specifying the offset.
 ... Currently not used.

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

See Also

```
Other signal processing methods: signal_bind(), signal_correct(), signal_drift(), signal_mean(), subset()
```

```
## X-ray diffraction
data("XRD")

## Plot spectrum
plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")

## Shift by one degree

XRD_offset <- signal_shift(XRD, lag = 1)
lines(XRD_offset, type = "1", col = "red")</pre>
```

smooth_likelihood 41

smooth_likelihood	Penalized Likelihood Smoothing
SINGO CII_TINCTINOOG	1 changea Bikennood Smoothing

Description

Penalized Likelihood Smoothing

Usage

```
smooth_likelihood(x, y, ...)
## S4 method for signature 'numeric, numeric'
smooth_likelihood(x, y, lambda, d = 2, SE = FALSE, progress = interactive())
## S4 method for signature 'ANY, missing'
smooth_likelihood(x, lambda, d = 2, SE = FALSE, progress = interactive())
```

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
	Currently not used.
lambda	An integer giving the smoothing parameter. The larger lambda is, the smoother the curve.
d	An integer specifying the order of the penalty.
SE	A logical scalar: should standard errors be returned?
progress	A logical scalar: should a progress bar be displayed?

Value

Returns a list with two components x and y.

Note

Matrix is required.

Author(s)

J. J. de Rooi et al. (original R code).

References

de Rooi, J. J., van der Pers, N. M., Hendrikx, R. W. A., Delhez, R., Böttger A. J. and Eilers, P. H. C. (2014). Smoothing of X-ray diffraction data and Ka2 elimination using penalized likelihood and the composite link model. *Journal of Applied Crystallography*, 47: 852-860. doi:10.1107/S1600576714005809

42 smooth_loess

See Also

Other smoothing methods: smooth_loess(), smooth_rectangular(), smooth_savitzky(), smooth_triangular(), smooth_whittaker()

Examples

```
## Not run:
## X-ray diffraction
data("XRD")
## Subset from 20 to 40 degrees
XRD <- signal_select(XRD, from = 20, to = 40)</pre>
## Plot diffractogram
plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")
## Penalized likelihood smoothing
lambda <- seq(from = 1, to = 8, length.out = 40)</pre>
lambda <- 10^lambda
likelihood <- smooth_likelihood(XRD, lambda = lambda, d = 3)</pre>
lines(likelihood, col = "red")
## Strip ka2
ka2 <- ka2_strip_penalized(XRD, lambda = lambda, tau = 0.5, nseg = 1)</pre>
lines(ka2, col = "blue")
## End(Not run)
```

smooth_loess

Loess Smoothing

Description

Smoothes intensities by loess fitting.

Usage

```
smooth_loess(x, y, ...)
## S4 method for signature 'numeric, numeric'
smooth_loess(x, y, span = 0.75, ...)
## S4 method for signature 'ANY, missing'
smooth_loess(x, span = 0.75, ...)
```

smooth_loess 43

Arguments

```
x, y A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
... Extra arguments to be passed to stats::loess().
span An integer specifying the degree of smoothing (see stats::loess()).
```

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

See Also

```
Other smoothing methods: smooth_likelihood(), smooth_rectangular(), smooth_savitzky(), smooth_triangular(), smooth_whittaker()
```

```
## Simulate data with some noise
x < - seq(-4, 4, length = 100)
y <- dnorm(x) + rnorm(100, mean = 0, sd = 0.01)
## Plot spectrum
plot(x, y, type = "1", xlab = "", ylab = "")
## Rectangular smoothing
unweighted <- smooth_rectangular(x, y, m = 3)</pre>
plot(unweighted, type = "1", xlab = "", ylab = "")
## Triangular smoothing
weighted <- smooth_triangular(x, y, m = 5)</pre>
plot(weighted, type = "1", xlab = "", ylab = "")
## Loess smoothing
loess <- smooth_loess(x, y, span = 0.75)
plot(loess, type = "l", xlab = "", ylab = "")
## Savitzky-Golay filter
savitzky \leftarrow smooth_savitzky(x, y, m = 21, p = 2)
plot(savitzky, type = "1", xlab = "", ylab = "")
## Whittaker smoothing
whittaker <- smooth_whittaker(x, y, lambda = 1600, d = 2)
plot(whittaker, type = "1", xlab = "", ylab = "")
```

44 smooth_rectangular

smooth_rectangular

Rectangular Smoothing

Description

Unweighted sliding-average or rectangular Smoothing.

Usage

```
smooth_rectangular(x, y, ...)
## S4 method for signature 'numeric, numeric'
smooth_rectangular(x, y, m = 3)
## S4 method for signature 'ANY, missing'
smooth_rectangular(x, m)
```

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
	Currently not used.
m	An odd integer giving the window size (i.e. the number of adjacent points to be used; see window_sliding()).

Details

It replaces each point in the signal with the average of m adjacent points.

Value

Returns a list with two components x and y.

Note

There will be (m-1)/2 points both at the beginning and at the end of the data series for which a complete m-width window cannot be obtained. To prevent data loss, progressively wider/narrower windows are used at both ends of the data series.

Author(s)

N. Frerebeau

See Also

```
Other smoothing methods: smooth_likelihood(), smooth_loess(), smooth_savitzky(), smooth_triangular(), smooth_whittaker()
```

smooth_savitzky 45

Examples

```
## Simulate data with some noise
x < - seq(-4, 4, length = 100)
y <- dnorm(x) + rnorm(100, mean = 0, sd = 0.01)
## Plot spectrum
plot(x, y, type = "1", xlab = "", ylab = "")
## Rectangular smoothing
unweighted <- smooth_rectangular(x, y, m = 3)</pre>
plot(unweighted, type = "1", xlab = "", ylab = "")
## Triangular smoothing
weighted <- smooth_triangular(x, y, m = 5)</pre>
plot(weighted, type = "1", xlab = "", ylab = "")
## Loess smoothing
loess <- smooth_loess(x, y, span = 0.75)
plot(loess, type = "1", xlab = "", ylab = "")
## Savitzky-Golay filter
savitzky \leftarrow smooth_savitzky(x, y, m = 21, p = 2)
plot(savitzky, type = "1", xlab = "", ylab = "")
## Whittaker smoothing
whittaker <- smooth_whittaker(x, y, lambda = 1600, d = 2)</pre>
plot(whittaker, type = "1", xlab = "", ylab = "")
```

smooth_savitzky

Savitzky-Golay Filter

Description

Savitzky-Golay Filter

Usage

```
smooth_savitzky(x, y, ...)
## S4 method for signature 'numeric,numeric'
smooth_savitzky(x, y, m = 3, p = 2)
## S4 method for signature 'ANY,missing'
smooth_savitzky(x, m, p)
```

Arguments

x, y A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).

46 smooth_savitzky

```
    Currently not used.
    An odd integer giving the window size (i.e. the number of adjacent points to be used).
    An integer giving the degree of the polynomial to be used.
```

Details

This method is based on the least-squares fitting of polynomials to segments of m adjacent points.

Value

Returns a list with two components x and y.

Note

There will be (m-1)/2 points both at the beginning and at the end of the data series for which a complete m-width window cannot be obtained. To prevent data loss, the original (m-1)/2 points at both ends of the data series are preserved.

Author(s)

N. Frerebeau

References

Gorry, P. A. (1990). General Least-Squares Smoothing and Differentiation by the Convolution (Savitzky-Golay) Method. *Analytical Chemistry*, 62(6), p. 570-573. doi:10.1021/ac00205a007.

Savitzky, A. & Golay, M. J. E. (1964). Smoothing and Differentiation of Data by Simplified Least Squares Procedures. *Analytical Chemistry*, 36(8), p. 1627-1639. doi:10.1021/ac60214a047.

See Also

Other smoothing methods: smooth_likelihood(), smooth_loess(), smooth_rectangular(), smooth_triangular(), smooth_whittaker()

```
## Simulate data with some noise
x <- seq(-4, 4, length = 100)
y <- dnorm(x) + rnorm(100, mean = 0, sd = 0.01)

## Plot spectrum
plot(x, y, type = "1", xlab = "", ylab = "")

## Rectangular smoothing
unweighted <- smooth_rectangular(x, y, m = 3)
plot(unweighted, type = "1", xlab = "", ylab = "")

## Triangular smoothing
weighted <- smooth_triangular(x, y, m = 5)</pre>
```

smooth_triangular 47

```
plot(weighted, type = "l", xlab = "", ylab = "")

## Loess smoothing
loess <- smooth_loess(x, y, span = 0.75)
plot(loess, type = "l", xlab = "", ylab = "")

## Savitzky-Golay filter
savitzky <- smooth_savitzky(x, y, m = 21, p = 2)
plot(savitzky, type = "l", xlab = "", ylab = "")

## Whittaker smoothing
whittaker <- smooth_whittaker(x, y, lambda = 1600, d = 2)
plot(whittaker, type = "l", xlab = "", ylab = "")</pre>
```

smooth_triangular

Triangular Smoothing

Description

Weighted sliding-average or triangular smoothing.

Usage

```
smooth_triangular(x, y, ...)
## S4 method for signature 'numeric,numeric'
smooth_triangular(x, y, m = 3)
## S4 method for signature 'ANY,missing'
smooth_triangular(x, m)
```

Arguments

x, y	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
	Currently not used.
m	An odd integer giving the window size (i.e. the number of adjacent points to be used; see window_sliding()).

Details

It replaces each point in the signal with the weighted mean of m adjacent points.

Value

Returns a list with two components x and y.

48 smooth_whittaker

Note

There will be (m-1)/2 points both at the beginning and at the end of the data series for which a complete m-width window cannot be obtained. To prevent data loss, progressively wider/narrower windows are used at both ends of the data series.

Author(s)

N. Frerebeau

See Also

```
Other smoothing methods: smooth_likelihood(), smooth_loess(), smooth_rectangular(), smooth_savitzky(), smooth_whittaker()
```

Examples

```
## Simulate data with some noise
x < - seq(-4, 4, length = 100)
y <- dnorm(x) + rnorm(100, mean = 0, sd = 0.01)
## Plot spectrum
plot(x, y, type = "1", xlab = "", ylab = "")
## Rectangular smoothing
unweighted <- smooth_rectangular(x, y, m = 3)</pre>
plot(unweighted, type = "l", xlab = "", ylab = "")
## Triangular smoothing
weighted <- smooth_triangular(x, y, m = 5)</pre>
plot(weighted, type = "1", xlab = "", ylab = "")
## Loess smoothing
loess \leftarrow smooth_loess(x, y, span = 0.75)
plot(loess, type = "l", xlab = "", ylab = "")
## Savitzky-Golay filter
savitzky \leftarrow smooth_savitzky(x, y, m = 21, p = 2)
plot(savitzky, type = "l", xlab = "", ylab = "")
## Whittaker smoothing
whittaker <- smooth_whittaker(x, y, lambda = 1600, d = 2)
plot(whittaker, type = "1", xlab = "", ylab = "")
```

smooth_whittaker

Whittaker Smoothing

Description

Whittaker Smoothing

smooth_whittaker 49

Usage

```
smooth_whittaker(x, y, ...)
## S4 method for signature 'numeric,numeric'
smooth_whittaker(x, y, lambda = 1600, d = 2, sparse = FALSE)
## S4 method for signature 'ANY,missing'
smooth_whittaker(x, lambda = 1600, d = 2, sparse = FALSE)
```

Arguments

х, у	A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).
	Currently not used.
lambda	An integer giving the smoothing parameter. The larger lambda is, the smoother the curve.
d	An integer specifying the order of the penalty.
sparse	A logical scalar: should sparse matrices be used for computation? If TRUE, Matrix is required.

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

References

```
Eilers, P. H. C. (2003). A Perfect Smoother. Analytical Chemistry, 75(14): 3631-36. doi:10.1021/ac034173t.
```

See Also

```
Other smoothing methods: smooth_likelihood(), smooth_loess(), smooth_rectangular(), smooth_savitzky(), smooth_triangular()
```

```
## Simulate data with some noise
x <- seq(-4, 4, length = 100)
y <- dnorm(x) + rnorm(100, mean = 0, sd = 0.01)
## Plot spectrum
plot(x, y, type = "l", xlab = "", ylab = "")
## Rectangular smoothing
unweighted <- smooth_rectangular(x, y, m = 3)</pre>
```

50 subset

```
plot(unweighted, type = "l", xlab = "", ylab = "")
## Triangular smoothing
weighted <- smooth_triangular(x, y, m = 5)
plot(weighted, type = "l", xlab = "", ylab = "")

## Loess smoothing
loess <- smooth_loess(x, y, span = 0.75)
plot(loess, type = "l", xlab = "", ylab = "")

## Savitzky-Golay filter
savitzky <- smooth_savitzky(x, y, m = 21, p = 2)
plot(savitzky, type = "l", xlab = "", ylab = "")

## Whittaker smoothing
whittaker <- smooth_whittaker(x, y, lambda = 1600, d = 2)
plot(whittaker, type = "l", xlab = "", ylab = "")</pre>
```

subset

Subset

Description

- signal_select() allows to subset by values of x.
- signal_slice() allows to subset by position along x.

Usage

```
signal_select(x, y, ...)
signal_slice(x, y, ...)
## S4 method for signature 'numeric, numeric'
signal_select(x, y, from, to)
## S4 method for signature 'ANY, missing'
signal_select(x, from, to)
## S4 method for signature 'numeric, numeric'
signal_slice(x, y, subset)
## S4 method for signature 'ANY, missing'
signal_slice(x, subset)
```

Arguments

x, y A numeric vector. If y is missing, an attempt is made to interpret x in a suitable way (see grDevices::xy.coords()).

window_sliding 51

... Currently not used.

from, to A numeric value giving the first and last value (in x unit) to be selected.

subset An integer vector giving either positive values to keep, or negative values to

drop. The values provided must be either all positive or all negative (coerced to

integer as by as.integer()).

Value

Returns a list with two components x and y.

Author(s)

N. Frerebeau

See Also

```
Other signal processing methods: signal_bind(), signal_correct(), signal_drift(), signal_mean(), signal_shift()
```

Examples

```
## gamma-ray spectrometry
data("BEGe")

## Plot spectrum
plot(BEGe, type = "l", xlab = "Energy (keV)", ylab = "Count")

## Subset from 2.75 keV to 200 keV
BEGe_1 <- signal_select(BEGe, from = 3, to = 200)

## Plot spectrum
plot(BEGe_1, type = "l", xlab = "Energy (keV)", ylab = "Count")

## Subset from the 20th to the 1250th value
BEGe_2 <- signal_slice(BEGe, subset = 20:1250)

## Plot spectrum
plot(BEGe_2, type = "l", xlab = "Energy (keV)", ylab = "Count")</pre>
```

window_sliding

Sliding Windows

Description

There will be (m-1)/2 points both at the beginning and at the end of the data series for which a complete m-width window cannot be obtained. To prevent data loss, progressively wider/narrower windows are evaluated at both ends of the data series.

52 window_sliding

Usage

```
window_sliding(n, m, ...)
## S4 method for signature 'integer,integer'
window_sliding(n, m, i = NULL)
## S4 method for signature 'numeric,numeric'
window_sliding(n, m, i = NULL)
```

Arguments

n	An integer giving the length of the data series (will be coerced with as . integer() and hence truncated toward zero).
m	An odd integer giving the window size, i.e. the number of adjacent points to be used (will be coerced with as.integer() and hence truncated toward zero).
	Currently not used.
i	A vector integer specifying the indices of the data points for which windows should be returned. If NULL (the default), windows are evaluated for each data point.

Value

Returns a length-*n* list of integer vectors (indices of the data points in each window).

Author(s)

N. Frerebeau

See Also

Other moving windows: window_tumbling()

```
## Length of the data series
n <- 10

## Progressive sliding windows
sliding <- window_sliding(n = n, m = 5)

plot(NULL, xlim = c(1, n), ylim = c(1, 10.5), xlab = "Index", ylab = "Window")
for (i in seq_along(sliding)) {
    w <- sliding[[i]]
    text(x = w, y = rep(i, length(w)), labels = w, pos = 3)
    lines(w, rep(i, length(w)), type = "1", lwd = 2)
}

## Tumbling windows
## (compare with drop = TRUE)</pre>
```

window_tumbling 53

```
tumbling <- window_tumbling(n = n, m = 3, drop = FALSE)

plot(NULL, xlim = c(1, n), ylim = c(1, 5.5), xlab = "Index", ylab = "Window")
for (i in seq_along(tumbling)) {
  w <- tumbling[[i]]
  text(x = w, y = rep(i, length(w)), labels = w, pos = 3)
  lines(w, rep(i, length(w)), type = "l", lwd = 2)
}</pre>
```

window_tumbling

Tumbling Windows

Description

Tumbling Windows

Usage

```
window_tumbling(n, m, ...)
## S4 method for signature 'integer,integer'
window_tumbling(n, m, drop = FALSE)
## S4 method for signature 'numeric,numeric'
window_tumbling(n, m, drop = FALSE)
```

Arguments

n	An integer giving the length of the data series (will be coerced with as.integer() and hence truncated toward zero).
m	An integer giving the window size, i.e. the number of adjacent points to be used (will be coerced with as.integer() and hence truncated toward zero).
	Currently not used.
drop	A logical scalar: if m is not a multiple of n, should the last data points be removed so that all windows have the same length?

Value

Returns a list of integer vectors (indices of the data points in each window).

Author(s)

N. Frerebeau

See Also

Other moving windows: window_sliding()

54 XRD

Examples

```
## Length of the data series
n <- 10
## Progressive sliding windows
sliding <- window_sliding(n = n, m = 5)</pre>
plot(NULL, xlim = c(1, n), ylim = c(1, 10.5), xlab = "Index", ylab = "Window")
for (i in seq_along(sliding)) {
 w <- sliding[[i]]</pre>
  text(x = w, y = rep(i, length(w)), labels = w, pos = 3)
  lines(w, rep(i, length(w)), type = "1", lwd = 2)
## Tumbling windows
## (compare with drop = TRUE)
tumbling <- window_tumbling(n = n, m = 3, drop = FALSE)</pre>
plot(NULL, xlim = c(1, n), ylim = c(1, 5.5), xlab = "Index", ylab = "Window")
for (i in seq_along(tumbling)) {
  w <- tumbling[[i]]</pre>
  text(x = w, y = rep(i, length(w)), labels = w, pos = 3)
 lines(w, rep(i, length(w)), type = "1", lwd = 2)
}
```

XRD

Powder X-ray Diffraction

Description

Powder X-ray Diffraction

Usage

XRD

Format

A data. frame with 2989 rows and 2 variables.

theta

count

See Also

Other datasets: BEGe, LaBr, Raman

XRD 55

```
data("XRD")
plot(XRD, type = "1", xlab = expression(2*theta), ylab = "Count")
```

Index

baseline_asls, 3 baseline_linear, 4 baseline_peakfilling, 5 baseline_polynomial, 7 baseline_rollingball, 8 baseline_rubberband, 9 baseline_snip, 11 * datasets BEGe, 12 LaBr, 17 Raman, 21 XRD, 54 * integration methods integrate_trapezoid, 14 * moving windows window_sliding, 51 window_tumbling, 53 * normalization methods rescale_area, 28 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replace_negative, 22 replace_threshold, 23 * resampling methods signal_shift, 40 subset, 50 * subset, 50 * smooth_likelihood, 41 smooth_loess, 42 smooth_rectangular, 47 smooth_avitzky, 45 smooth_loess, 42 smooth_loess, 42 smooth_rectangular, 47 smoth_loess, 42 smooth_rectangular, 47 smooth_avitzky, 45 smooth_loess, 42 smooth_rectangular, 47 smooth_loess, 42 smooth_rectangular, 49 smooth_crangular, 47 smooth_loess, 42 smooth_crangular, 47 smoth_loes, 42 smooth_rectangular, 49 smooth_gealine_late, 48 * specialized tools ka2_strip_penalized, 15 xaleseline_asls, 3, 5-7, 9, 10, 12 baseline_asls, and integrate transform, 4, 4, 6, 7, 9, 10, 12 baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, 1, 4 baseline_line
baseline_peakfilling, 5 baseline_polynomial, 7 baseline_rollingball, 8 baseline_rubberband, 9 baseline_snip, 11 * datasets BEGe, 12 LaBr, 17 Raman, 21 XRD, 54 * integrate_rectangle, 13 integrate_trapezoid, 14 * moving windows window_sliding, 51 window_sliding, 51 window_tumbling, 53 * normalization methods rescale_area, 28 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replace_negative, 22 replace_engentered * resampling methods * methods smooth_likelihood, 41 smooth_loess, 42 smooth_rectangular, 44 smooth_savitzky, 45 smooth_whittaker, 48 * smooth_likelihood, 41 smooth_loess, 42 smooth_loess, 42 smooth_likelihood, 41 smooth_loess, 42 smooth_loes, 42 smooth_loes, 42 smooth_loes, 42 smooth_likelihood, 41 smooth_loes, 42 smooth_loes, 42 smooth_likelihood, 41 smooth_loes, 42 smooth_loes, 42 smooth_loes, 42 smooth_likelihood, 41 smooth_loes, 42 smooth_loes, 4
baseline_polynomial, 7 baseline_rollingball, 8 baseline_rubberband, 9 baseline_snip, 11 * datasets BEGe, 12 LaBr, 17 Raman, 21 XRD, 54 * integration methods integrate_trapezoid, 14 * moving windows window_sliding, 51 window_tumbling, 53 * normalization methods rescale_area, 28 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods * measule_trapeloid, 12 * smooth_loess, 42 * smooth_savitzky, 45 * smooth_triangular, 47 * smooth_savitzky, 45 * smooth_savitzky, 45 * smooth_triangular, 47 * smooth_savitzky, 45 * smooth_triangular, 47 * smooth_savitzky, 45 * smooth_trectangular, 44 * smooth_savitzky, 45 * smooth_savitzky, 45 * smooth_trectangular, 44 * smooth_savitzky, 45 * smooth_trectangular, 47 * smooth_savitzky, 45 * smooth_savitzky, 45 * smooth_savitzky, 45 * smooth_savitzky, 45 * smooth_trectangular, 47 * smooth_savitzky, 45 * smooth_
baseline_rollingball, 8 baseline_rubberband, 9 baseline_snip, 11 * datasets BEGe, 12 LaBr, 17 Raman, 21 XRD, 54 * integration methods integrate_rectangle, 13 integrate_trapezoid, 14 * moving windows window_sliding, 51 window_tumbling, 53 * normalization methods rescale_area, 28 rescale_range, 29 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replace_negative, 22 replace_negative, 22 replace_treshold, 23 * resampling methods * resampling methods * smooth_loess, 42 * smooth_rectangular, 44 * smooth_triangular, 47 * smooth_twittaker, 48 * * specialized tools ka2_strip_penalized, 15 * sas.integer(), 51–53 * sas.integer(), 51–53 * baseline_asls, 3, 5–7, 9, 10, 12 baseline_asls, ANY, missing-method (baseline_asls), 3 baseline_asls,, and (baseline_linear), 4, 6, 7, 9, 10, 12 baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, ANY, missing-method (baseline_linear), 4 baseline_peakfilling(), 36 baseline_peakfilling(), 36 baseline_peakfilling(), 36 baseline_peakfilling, ANY, missing-method
baseline_rollingball, 8 baseline_rubberband, 9 baseline_snip, 11 * datasets BEGe, 12 LaBr, 17 Raman, 21 XRD, 54 * integration methods integrate_rectangle, 13 integration methods integrate_rectangle, 13 integration methods integrate_rectangle, 13 integration methods integrate_rectangle, 13 integrate rectangle, 13 integrate rectangle, 13 integrate rectangle, 13 integrate rectangle, 14 *smooth_rreingular, 47 smooth_triangular, 47 smooth_svitzky, 45 smooth_svitzky, 45 smooth_rectangular, 44 smooth_svitzky, 45 smooth_rreingular, 47 smooth_svitzky, 45 smooth_rreingular, 47 smooth_vhittaker, 48 *specialized tools as.integer(), 51–53 baseline_asls, 3, 5–7, 9, 10, 12 baseline_asls, (), 36 baseline_asls, (), 36 baseline_asls, ANY, missing-method (baseline_asls), 3 baseline_asls, ANY, missing-method (baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, ANY, missing-method (baseline_linear), 4 baseline_linear, numeric_numeric_method (baseline_linear), 4 baseline_linear, numeric_numeric_method (baseline_linear), 4 baseline_peakfilling(), 36 baseline_peakfilling(), 36 baseline_peakfilling, ANY, missing-method
baseline_snip, 11 * datasets BEGe, 12 LaBr, 17 Raman, 21 XRD, 54 * integration methods integrate_rectangle, 13 integrate_trapezoid, 14 * moving windows window_sliding, 51 window_tumbling, 53 * normalization methods rescale_range, 29 rescale_snv, 31 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_treshold, 23 * resampling methods * mesonth_savitzky, 45 smooth_savitzky, 45 smooth_triangular, 47 smooth_whittaker, 48 * specialized tools ka2_strip_penalized, 15 ka2_strip_penalized, 15 ka2_strip_penalized, 15 ka2_strip_penalized, 15 sas.integer(), 51–53 baseline_asls, 3, 5–7, 9, 10, 12 baseline_asls, 3, 36 baseline_asls, ANY, missing-method (baseline_asls), 3 baseline_asls, Nay, missing-method (baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, ANY, missing-method (baseline_linear), 4 baseline_linear, Nay, missing-method (baseline_linear), 4 baseline_peakfilling(), 36 baseline_peakfilling(), 36 baseline_peakfilling(), 36 baseline_peakfilling(), 36 baseline_peakfilling, ANY, missing-method
* datasets BEGe, 12 LaBr, 17 Raman, 21 XRD, 54 * integration methods integrate_rectangle, 13 integrate_trapezoid, 14 * moving windows window_sliding, 51 window_tumbling, 53 * normalization methods rescale_area, 28 rescale_snv, 31 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_treshold, 23 * resampling methods BEGe, 12 smooth_whittaker, 48 * specialized tools ka2_strip_penalized, 15 ka2_strip_easls, 3, 3-7, 9, 10, 12 baseline_asls, 3, 5-7, 9, 10, 12 baseline_asls, 3, 5-7, 9, 10, 12 baseline_asls, 3, 6-7, 9, 10, 12 baseline_asls, 36 baseline_asls, anmeric, numeric-method (baseline_asls, anueric, numeric, numeric-method (baseline_linear, 4, 4, 6, 7, 9, 10, 12 b
BEGe, 12 LaBr, 17 Raman, 21 XRD, 54 * integration methods integrate_rectangle, 13 integrate_trapezoid, 14 * moving windows window_sliding, 51 window_tumbling, 53 * normalization methods rescale_area, 28 rescale_range, 29 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods * speaks detection methods rescampling methods * resampling methods * speaks detection methods resplace_peakfilling, 4, 5, 5, 7, 9, 10, 12 baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, ANY, missing-method (baseline_linear), 4 baseline_linear, 4, 4 baseline_linear, 5, 5, 7, 9, 10, 12 baseline_peakfilling, 4, 5, 5, 7, 9, 10, 12 baseline_peakfilling, 9, NY, missing-method
LaBr, 17 Raman, 21 XRD, 54 * integration methods integrate_rectangle, 13 integrate_trapezoid, 14 * moving windows window_sliding, 51 window_tumbling, 53 * normalization methods rescale_area, 28 rescale_range, 29 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods * resampling methods * specialized tools * ka2_strip_penalized, 15 * specialized tools * as.integer(), 51-53 * baseline_asls, 3, 5-7, 9, 10, 12 baseline_asls, anumeric, numeric-method (baseline_asls, 3, 5-7, 9, 10, 12 baseline_asls, anumeric, numeric-method (baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_asls, anumeric, numeric-method (baseline_linear), 4 baseline_asls, anumeric, numeric-method (baseline_linear), 4 baseline_asls, anumeric, numeric-method (baseline_linear), 4 baseline_asls, 3 baseline_asls, anumeric, numeric-method (baseline_linear), 4 ba
LaBr, 17 Raman, 21 XRD, 54 * integration methods integrate_rectangle, 13 integrate_trapezoid, 14 * moving windows window_sliding, 51 window_tumbling, 53 * normalization methods rescale_area, 28 rescale_range, 29 rescale_snv, 31 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods * ka2_strip_penalized, 15 * sas.integer(), 51-53 * baseline_asls, 3, 5-7, 9, 10, 12 * baseline_asls, 3, 5-7
Raman, 21 XRD, 54 * integration methods integrate_rectangle, 13 integrate_trapezoid, 14 * moving windows window_sliding, 51 window_tumbling, 53 * normalization methods rescale_area, 28 rescale_range, 29 rescale_snv, 31 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods * resampling methods * resampling methods * resampling methods * integration methods baseline_asls, 3, 5-7, 9, 10, 12 baseline_asls, 3, 3 baseline_asls, 3, 3 baseline_asls, numeric, numeric-method (baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, ANY, missing-method (baseline_linear), 4 baseline_linear, 4 baseline_peakfilling(), 36 baseline_peakfilling, ANY, missing-method
<pre>* integration methods integrate_rectangle, 13 integrate_trapezoid, 14 * moving windows window_sliding, 51 window_tumbling, 53 * normalization methods rescale_area, 28 rescale_range, 29 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods as.integer(), 51–53 baseline_asls, 3, 5–7, 9, 10, 12 baseline_asls), 3 baseline_asls, Numeric, numeric-method (baseline_asls), 3 baseline_asls-method (baseline_asls), 3 baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, ANY, missing-method (baseline_linear), 4 baseline_linear, numeric, numeric-method (baseline_linear), 4 baseline_linear, 4 baseline_peakfilling(), 36 baseline_peakfilling(), 36 baseline_peakfilling, ANY, missing-method</pre>
<pre>* integration methods integrate_rectangle, 13 integrate_trapezoid, 14 * moving windows window_sliding, 51 window_tumbling, 53 * normalization methods rescale_area, 28 rescale_range, 29 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods integrate_rectangle, 13 baseline_asls, 3, 5-7, 9, 10, 12 baseline_asls, ANY, missing-method (baseline_asls), 3 baseline_asls, numeric, numeric-method (baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear,, ANY, missing-method (baseline_linear), 4 baseline_linear, numeric, numeric-method (baseline_linear), 4 baseline_peakfilling, 4, 5, 5, 7, 9, 10, 12 baseline_peakfilling, ANY, missing-method</pre>
integrate_rectangle, 13 integrate_trapezoid, 14 * moving windows window_sliding, 51 window_tumbling, 53 * normalization methods rescale_area, 28 rescale_range, 29 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods integrate_trapezoid, 14 baseline_asls, 3, 5–7, 9, 10, 12 baseline_asls, ANY, missing-method (baseline_asls), 3 baseline_asls, numeric, numeric, numeric-method (baseline_linear(), 36 baseline_linear, ANY, missing-method (baseline_linear), 4 baseline_linear, numeric, numeric-method (baseline_linear), 4 baseline_linear-method (baseline_linear), 4 baseline_peakfilling(), 36 * resampling methods baseline_peakfilling, ANY, missing-method
integrate_trapezoid, 14 * moving windows window_sliding, 51 window_tumbling, 53 * normalization methods rescale_area, 28 rescale_range, 29 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods asseline_asls, 3, 3-7, 9, 10, 12 baseline_asls, ANY, missing-method (baseline_asls), 3 baseline_asls, numeric, numeric-method (baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, ANY, missing-method (baseline_linear), 4 baseline_linear, numeric, numeric-method (baseline_linear), 4 baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, ANY, missing-method (baseline_linear), 4 baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, ANY, missing-method baseline_linear, 0, 4, 0, 10, 12 baseline_linear,
<pre>* moving windows window_sliding, 51 window_tumbling, 53 * normalization methods rescale_area, 28 rescale_range, 29 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods * resampling methods window_sliding, 51 (baseline_asls, ANY, missing-method (baseline_asls), 3 baseline_asls-method (baseline_asls), 3 baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, ANY, missing-method (baseline_linear), 4 baseline_linear, numeric, numeric-method (baseline_linear), 4 baseline_linear, 4, 4, 5, 5, 7, 9, 10, 12 baseline_peakfilling(), 36 baseline_peakfilling, ANY, missing-method</pre>
<pre>window_sliding, 51 window_tumbling, 53 * normalization methods rescale_area, 28 rescale_range, 29 rescale_snv, 31 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods window_tumbling, 53 (baseline_asls), 3 baseline_asls, numeric, numeric-method (baseline_asls), 3 baseline_asls, numeric, numeric-method (baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, ANY, missing-method (baseline_linear), 4 baseline_linear, numeric, numeric-method (baseline_linear), 4 baseline_linear), 4 baseline_linear), 4 baseline_peakfilling, 4, 5, 5, 7, 9, 10, 12 baseline_peakfilling(), 36 baseline_peakfilling, ANY, missing-method</pre>
<pre>window_tumbling, 53 * normalization methods rescale_area, 28 rescale_range, 29 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods * resampling methods * resampling methods * rescale_tumbling, 53 baseline_asls, numeric, numeric-method (baseline_asls), 3 baseline_asls, 3 baseline_asls, 3 baseline_asls, numeric, numeric</pre>
* normalization methods rescale_area, 28 rescale_range, 29 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods mathematics baseline_asls, numeric, numeric leasts, numer ic, numer asls), 3 baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, ANY, missing-method (baseline_linear), 4 baseline_linear, numeric, numeric-method (baseline_linear), 4 baseline_linear, 4 baselin
rescale_area, 28 rescale_range, 29 rescale_snv, 31 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods rescale_area, 28 baseline_asls, 3 baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear(), 36 baseline_linear, numeric, numeric-method (baseline_linear), 4 baseline_linear), 4 baseline_linear), 4 baseline_peakfilling, 4, 5, 5, 7, 9, 10, 12 baseline_peakfilling(), 36 baseline_peakfilling, ANY, missing-method
rescale_range, 29 rescale_snv, 31 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods baseline_linear, 4, 4, 6, 7, 9, 10, 12 baseline_linear, ANY, missing-method (baseline_linear), 4 baseline_linear, numeric, numeric-method (baseline_linear), 4 baseline_linear), 4 baseline_linear), 4 baseline_peakfilling, 4, 5, 5, 7, 9, 10, 12 baseline_peakfilling(), 36 baseline_peakfilling, ANY, missing-method
rescale_snv, 31 rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods baseline_linear, ANY, missing-method (baseline_linear), 4 baseline_linear-numeric, numeric-method (baseline_linear), 4 baseline_linear), 4 baseline_peakfilling, 4, 5, 5, 7, 9, 10, 12 baseline_peakfilling(), 36 baseline_peakfilling, ANY, missing-method
rescale_total, 32 rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods baseline_linear, ANY, missing-method (baseline_linear), 4 baseline_linear-method (baseline_linear), 4 baseline_linear), 4 baseline_linear), 4 baseline_linear), 4 baseline_linear), 4 baseline_peakfilling, 4, 5, 5, 7, 9, 10, 12 baseline_peakfilling(), 36 baseline_peakfilling, ANY, missing-method
rescale_transform, 33 * peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods baseline_linear, numeric, numeric-method (baseline_linear), 4 baseline_linear), 4 baseline_peakfilling, 4, 5, 5, 7, 9, 10, 12 baseline_peakfilling(), 36 baseline_peakfilling, ANY, missing-method
* peaks detection methods peaks_find, 18 peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods (baseline_linear, numeric, numeric-method (baseline_linear), 4 baseline_linear), 4 baseline_peakfilling, 4, 5, 5, 7, 9, 10, 12 baseline_peakfilling(), 36 baseline_peakfilling, ANY, missing-method
peaks_find, 18 (baseline_linear), 4 peaks_fwhm, 19 baseline_linear-method * replacement methods (baseline_linear), 4 replace_negative, 22 baseline_peakfilling, 4, 5, 5, 7, 9, 10, 12 replace_threshold, 23 baseline_peakfilling(), 36 * resampling methods baseline_peakfilling, ANY, missing-method
peaks_fwhm, 19 * replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods (baseline_linear), 4 baseline_peakfilling, 4, 5, 5, 7, 9, 10, 12 baseline_peakfilling(), 36 baseline_peakfilling, ANY, missing-method
* replacement methods replace_negative, 22 replace_threshold, 23 * resampling methods (baseline_linear), 4 baseline_peakfilling, 4, 5, 5, 7, 9, 10, 12 baseline_peakfilling(), 36 baseline_peakfilling, ANY, missing-method
replace_negative, 22 baseline_peakfilling, 4, 5, 5, 7, 9, 10, 12 replace_threshold, 23 baseline_peakfilling(), 36 * resampling methods baseline_peakfilling, ANY, missing-method
replace_threshold, 23 baseline_peakfilling(), 36 * resampling methods baseline_peakfilling, ANY, missing-method
* resampling methods baseline_peakfilling,ANY,missing-method
resample_bin, 24 (baseline_peakfilling), 5
resample_down, 25 baseline_peakfilling, numeric-method
resample_interpolate, 26 (baseline_peakfilling), 5
* signal processing methods baseline_peakfilling-method
signal_bind, 34 (baseline_peakfilling), 5
signal_correct, 36 baseline_polynomial, 4-6, 7, 9, 10, 12
signal drift. 37 baseline polynomial(). 36

INDEX 57

baseline_polynomial,ANY,missing-method	<pre>integrate_rectangle,numeric,numeric-method</pre>
(baseline_polynomial), 7	(integrate_rectangle), 13
<pre>baseline_polynomial,numeric,numeric-method</pre>	<pre>integrate_rectangle-method</pre>
<pre>(baseline_polynomial), 7</pre>	(integrate_rectangle), 13
baseline_polynomial-method	integrate_trapezoid, <i>14</i> , 14
(baseline_polynomial), 7	<pre>integrate_trapezoid, ANY, missing-method</pre>
baseline_rollingball, 4-7, 8, 10, 12	(integrate_trapezoid), 14
baseline_rollingball(), 36	integrate_trapezoid, numeric, numeric-method
- · · · · · · · · · · · · · · · · · · ·	
baseline_rollingball,ANY,missing-method	(integrate_trapezoid), 14
(baseline_rollingball), 8	<pre>integrate_trapezoid-method</pre>
<pre>baseline_rollingball,numeric,numeric-method</pre>	(integrate_trapezoid), 14
(baseline_rollingball), 8	
baseline_rollingball-method	ka2_strip_penalized, 15
(baseline_rollingball), 8	ka2_strip_penalized,ANY,missing-method
baseline_rubberband, 4-7, 9, 9, 12	<pre>(ka2_strip_penalized), 15</pre>
baseline_rubberband(), 36	ka2_strip_penalized,numeric,numeric-method
baseline_rubberband,ANY,missing-method	(ka2_strip_penalized), 15
(baseline_rubberband), 9	ka2_strip_penalized-method
baseline_rubberband,numeric,numeric-method	(ka2_strip_penalized), 15
•	(kaz_strip_penalized), 13
(baseline_rubberband), 9	LaDa 12 17 21 54
baseline_rubberband-method	LaBr, 13, 17, 21, 54
(baseline_rubberband), 9	list, 3, 5–8, 10, 11, 13, 14, 16, 18, 22–24,
baseline_snip, <i>4</i> – <i>7</i> , <i>9</i> , <i>10</i> , 11	26–28, 30–32, 34, 37–41, 43, 44, 46,
baseline_snip(), 36	47, 49, 51–53
baseline_snip, ANY, missing-method	logical, 6, 10, 11, 13, 16, 38, 41, 49, 53
(baseline_snip), 11	
baseline_snip,numeric,numeric-method	matrix, <i>35</i>
(baseline_snip), 11	
baseline_snip-method (baseline_snip), 11	numeric, 3, 4, 6-8, 10, 11, 13, 14, 16, 18, 20,
BEGe, 12, 17, 21, 54	22–25, 27, 28, 30–32, 34, 36, 38, 40,
blue, 12, 17, 21, 34	41, 43–45, 47, 49–51
phomostom 10 20 26	71, 10 10, 17, 15 01
character, 18, 28, 36	peaks_find, 18, 20
12 17 21 54	peaks_find, ANY, missing-method
data.frame, 13, 17, 21, 54	-
	(peaks_find), 18
function, 23, 24, 34	peaks_find, numeric, numeric-method
	(peaks_find), 18
grDevices::xy.coords(), 3, 4, 6–8, 10, 11,	<pre>peaks_find-method (peaks_find), 18</pre>
13, 14, 16, 18, 20, 22–25, 27, 28,	peaks_fwhm, <i>19</i> , 19
30-32, 34-36, 38-41, 43-45, 47, 49,	<pre>peaks_fwhm,ANY,missing-method</pre>
50	(peaks_fwhm), 19
	peaks_fwhm,numeric,numeric-method
hyperSpec::spc.rubberband(), 10	(peaks_fwhm), 19
, регорог торот аллоглана (у, то	peaks_fwhm-method (peaks_fwhm), 19
integer, 3, 6-8, 11, 16, 18, 24, 25, 41, 43, 44,	pearso_1 will inceriou (pearso_1 will), 17
46, 47, 49, 51–53	Raman, 13, 17, 21, 54
integrate_rectangle, 13, 15	replace_negative, 22, 23
integrate_rectangle, ANY, missing-method	replace_negative,ANY,missing-method
(integrate_rectangle), 13	(replace_negative), 22

58 INDEX

replace_negative,numeric,numeric-method	(rescale_range), 29
<pre>(replace_negative), 22</pre>	rescale_min,numeric,numeric-method
replace_negative-method	(rescale_range), 29
(replace_negative), 22	rescale_min-method(rescale_range), 29
replace_threshold, 22, 23	rescale_range, 29, 29, 31, 33, 34
replace_threshold, ANY, missing, function-metho	drescale_range,ANY,missing-method
(replace_threshold), 23	(rescale_range), 29
replace_threshold,ANY,missing,numeric-method	rescale_range,numeric,numeric-method
(replace_threshold), 23	(rescale_range), 29
replace_threshold,numeric,numeric,function-m	e ተዋ ልgale_range-method(rescale_range), 29
(replace_threshold), 23	rescale_snv, 29, 30, 31, 33, 34
replace_threshold, numeric, numeric, numeric-me	
(replace_threshold), 23	(rescale_snv), 31
replace_threshold-method	rescale_snv,numeric,numeric-method
(replace_threshold), 23	(rescale_snv), 31
resample_bin, 24, 26, 27	rescale_snv-method (rescale_snv), 31
resample_bin,ANY,missing-method	rescale_total, 29-31, 32, 34
(resample_bin), 24	rescale_total, ANY, missing-method
resample_bin, numeric, numeric-method	(rescale_total), 32
(resample_bin), 24	rescale_total,numeric,numeric-method
	(rescale_total), 32
resample_bin-method (resample_bin), 24	rescale_total-method(rescale_total), 32
resample_down, 24, 25, 27	rescale_transform, 29-31, 33, 33
resample_down, ANY, missing-method	rescale_transform, ANY, missing-method
(resample_down), 25	(rescale_transform), 33
resample_down, numeric, numeric-method	rescale_transform, numeric, numeric-method
(resample_down), 25	(rescale_transform), 33
resample_down-method (resample_down), 25	rescale_transform-method
resample_interpolate, 24, 26, 26	<pre>(rescale_transform), 33</pre>
resample_interpolate, ANY, missing-method	(
(resample_interpolate), 26	signal_bind, 34, 37-40, 51
resample_interpolate, numeric, numeric-method	signal_bind, ANY-method(signal_bind), 34
(resample_interpolate), 26	signal_bind-method(signal_bind), 34
resample_interpolate-method	signal_correct, 35, 36, 38-40, 51
<pre>(resample_interpolate), 26</pre>	signal_correct(), 4-7, 9, 10, 12
rescale_area, 28, 30, 31, 33, 34	signal_correct,ANY,missing-method
rescale_area,ANY,missing-method	(signal_correct), 36
(rescale_area), 28	signal_correct,numeric,numeric-method
rescale_area,numeric,numeric-method	(signal_correct), 36
(rescale_area), 28	<pre>signal_correct-method(signal_correct),</pre>
rescale_area-method(rescale_area),28	36
rescale_max (rescale_range), 29	signal_drift, 35, 37, 37, 39, 40, 51
rescale_max,ANY,missing-method	signal_drift,ANY,missing,ANY-method
(rescale_range), 29	(signal_drift), 37
rescale_max,numeric,numeric-method	signal_drift,numeric,numeric,numeric-method
(rescale_range), 29	(signal_drift), 37
rescale_max-method(rescale_range), 29	<pre>signal_drift-method(signal_drift), 37</pre>
rescale_min(rescale_range), 29	signal_mean, 35, 37, 38, 38, 40, 51
rescale_min,ANY,missing-method	signal_mean, ANY-method(signal_mean), 38

INDEX 59

signal_mean-method(signal_mean), 38	(smooth_triangular), 47
signal_select(subset), 50	smooth_triangular,numeric,numeric-method
signal_select,ANY,missing-method	(smooth_triangular),47
(subset), 50	smooth_triangular-method
signal_select,numeric,numeric-method	(smooth_triangular),47
(subset), 50	smooth_whittaker, <i>42–44</i> , <i>46</i> , <i>48</i> , 48
signal_select-method(subset), 50	$smooth_whittaker(), 6$
signal_shift, <i>35</i> , <i>37</i> – <i>39</i> , 40, <i>51</i>	<pre>smooth_whittaker,ANY,missing-method</pre>
signal_shift,ANY,missing-method	(smooth_whittaker), 48
(signal_shift), 40	<pre>smooth_whittaker,numeric,numeric-method</pre>
signal_shift,numeric,numeric-method	(smooth_whittaker), 48
(signal_shift), 40	smooth_whittaker-method
signal_shift-method(signal_shift), 40	(smooth_whittaker), 48
	stats::approx(), 27
signal_slice(subset), 50	stats::loess(), 43
signal_slice,ANY,missing-method	stats::smooth.spline(), 10
(subset), 50	subset, 35, 37–40, 50
signal_slice,numeric,numeric-method	sum(), 24
(subset), 50	3um(), 27
$signal_slice-method(subset), 50$	window_sliding, 51, 53
smooth_likelihood, 41, 43, 44, 46, 48, 49	window_sliding(), 8, 44, 47
<pre>smooth_likelihood,ANY,missing-method</pre>	window_sliding,integer,integer-method
(smooth_likelihood), 41	(window_sliding), 51
<pre>smooth_likelihood,numeric,numeric-method</pre>	window_sliding,numeric,numeric-method
(smooth_likelihood), 41	(window_sliding), 51
smooth_likelihood-method	
(smooth_likelihood), 41	<pre>window_sliding-method(window_sliding), 51</pre>
smooth_loess, 42, 42, 44, 46, 48, 49	
smooth_loess,ANY,missing-method	window_tumbling, 52, 53
(smooth_loess), 42	window_tumbling(), 24
smooth_loess,numeric,numeric-method	window_tumbling,integer,integer-method
(smooth_loess), 42	(window_tumbling), 53
smooth_loess-method (smooth_loess), 42	window_tumbling,numeric,numeric-method
	(window_tumbling), 53
smooth_rectangular, 42, 43, 44, 46, 48, 49	window_tumbling-method
smooth_rectangular,ANY,missing-method	<pre>(window_tumbling), 53</pre>
(smooth_rectangular), 44	
smooth_rectangular,numeric,numeric-method	XRD, 13, 17, 21, 54
(smooth_rectangular), 44	
smooth_rectangular-method	
(smooth_rectangular), 44	
smooth_savitzky, <i>42–44</i> , 45, <i>48</i> , <i>49</i>	
<pre>smooth_savitzky,ANY,missing-method</pre>	
(smooth_savitzky), 45	
<pre>smooth_savitzky,numeric,numeric-method</pre>	
(smooth_savitzky), 45	
smooth_savitzky-method	
(smooth_savitzky),45	
smooth_triangular, <i>42-44</i> , <i>46</i> , 47, <i>49</i>	
<pre>smooth_triangular,ANY,missing-method</pre>	