Package 'breakage'

February 19, 2015

Type Package

Title SICM pipette tip geometry estimation

Version 1.1-1

Date 2014-12-08

Author Matthew Caldwell <m.caldwell@ucl.ac.uk>

Maintainer Matthew Caldwell <m.caldwell@ucl.ac.uk>

Description

Estimates geometry of SICM pipette tips by fitting a physical model to recorded breakage-current data.

License Artistic-2.0

LazyLoad yes

LazyData yes

Depends Imap

NeedsCompilation no

Repository CRAN

Date/Publication 2014-12-08 21:39:22

R topics documented:

`breakage-package` *Estimating SICM pipette tip geometry from breakage data*

Description

Functions to estimate the interior half-cone angle and tip radius of a scanning ion conductance microscope (SICM) pipette by fitting recorded changes in pipette conductance as the pipette tip is broken. See the referenced paper for a full discussion of this technique.

Details

Package:	breakage
Type:	Package
Version:	1.1
Date:	2014-12-08
License:	Artistic 2.0
LazyLoad:	yes

Author(s)

Matthew Caldwell <m.caldwell@ucl.ac.uk>

References

Caldwell, M., Del Linz, S. J. L., Smart, T. G. S. and Moss, G. W. J. 2012 Method for estimating the tip geometry of scanning ion conductance microscope pipets. *Anal. Chem.* 84(21):8980–8984

Examples

```
# load the example data set
data("break.data")

# get the bottom points
raw.points <- breakage.plot(break.data, mV=100)
## Not run:
# select the clusters
clustered <- break.clust(raw.points)

# fit the tip geometry to the clustered points
fit <- fit.breakage(clustered, rho=64)

## End(Not run)
```

apply.breaks *Apply a function to regions of a signal*

Description

Applies a specified function to sections of an input signal in between specified index points. The indices can be supplied explicitly in the `idx` parameter, or as the logical vector `btm` in which the divider points are set to `TRUE`. By default, the `link{median}` of each segment is calculated.

Usage

```
apply.breaks(x, btm = NULL, f = median, idx = (1:length(btm))[btm])
```

Arguments

`x`	The data vector whose segments are to be processed.
`btm`	A logical vector containing `TRUE` at the points where the data in `x` should be split into segments. Either this or `idx` must be provided.
`f`	A function object to apply to each segment of data. Should return a single value. The default is to use the `median` function.
`idx`	A vector of indices at which the data vector should be split into segments. The data at the index is not included in the segment. Either this or `btm` must be provided.

Details

Unbounded stretches at each end are dropped, as are empty regions (i.e. regions between adjacent dividers). Thus, if there are `N` divider points there will be at most `N-1` returned values (and possibly fewer).

Value

A vector containing the function results for each non-empty segment.

Author(s)

Matthew Caldwell

See Also

`find.bottom`

Examples

```
x <- rnorm(500)
apply.breaks(x, idx=50 * 1:10)
```

`break.clust` *Manually select point clusters from breakage data*

Description

Plots the supplied position-resistance data and allows individual clusters to be selected graphically. A data frame of the median breakage distances and resistances for the selected clusters is returned.

Usage

```
break.clust(data, zero.invert = TRUE)
```

Arguments

`data`	A data frame including `Z` and `Mohm` columns, as returned by `breakage.plot`.
`zero.invert`	Whether to convert to the `Z` positional data to breakage distance. See Details for more information.

Details

Point clusters are chosen using the Imap `select.pts` function. A cluster is selected by clicking points around it to draw a polygon. Right-click to close the polygon and define the cluster. To finish selecting, draw a polygon that encloses no points.

The recorded positional information is typically in terms of the piezo used to position the pipette. Model fitting, on the other hand, is done in terms of the breakage distance, i.e. measuring away from the unbroken pipette tip. We can convert from one to the other by taking the most extended surface position as that corresponding to the unbroken tip and substracting subsequent surface positions to determine the length of the broken off region, and this is done by default.

Where data have been collected in more than one pass, with possible positional changes in between, the results from each pass must be aligned together. It may be more convenient to do this in the context of the full position, and then convert to breakage distance afterward.

Value

`Z`	The median position for each cluster, in microns. Values are optionally offset from the maximum tip position, as described above.
`Mohm`	The median resistance for each cluster, in megohms.

Author(s)

Matthew Caldwell

References

Caldwell, M., Del Linz, S. J. L., Smart, T. G. S. and Moss, G. W. J. 2012 Method for estimating the tip geometry of scanning ion conductance microscope pipets. *Anal. Chem.* 84(21):8980–8984

See Also

breakage.plot, fit.breakage

Examples

```
# load the example data set
data("break.data")

# get the bottom points
raw.points <- breakage.plot( break.data, mV=100 )
## Not run:
# select the clusters
clustered <- break.clust( raw.points )

## End(Not run)
```

---

break.data *Sample recording of SICM pipette breakage*

Description

Position and current data recorded using a scanning ion conductance microscope in hopping mode. An elevated fall rate drove the pipette to overshoot its detection range and collide with the cover slip surface and break, altering its resistance.

Usage

```
data(break.data)
```

Format

A data frame with 9999 observations on the following 3 variables.

- s The time of each data point, in milliseconds from the start of the recording.
- pA The ion current passing through the pipette, in picoamps.
- Z The pipette position, in microns, recorded within the frame of reference of the SICM piezo actuator.

Source

Caldwell, M., Del Linz, S. J. L., Smart, T. G. S. and Moss, G. W. J. 2012 Method for estimating the tip geometry of scanning ion conductance microscope pipets. *Anal. Chem.* (in press)

Examples

```
data(break.data)
plot(pA ~ s, data=break.data, type="l")
plot(Z ~ s, data=break.data, type="l")
```

`breakage.plot` *Plot a pipette breakage current or resistance relationship*

Description

Calculates and plots a current-distance or resistance-distance relation from recorded current-position data during pipette tip breakage.

Usage

```
breakage.plot(x, time.limits = NULL, btm = find.bottom(x$Z),
    pch = 16, col = 1, lty = "solid", col.l = 2,
    plot.line = FALSE, f = median, mV = NA, ...)
```

Arguments

`x`	A data frame containing the experimental current and position traces. The frame should include a column named `Z`, containing the position in microns, and a column named `pA`, containing the recorded current in picoamps. If the `time.limits` parameter is specified, there must also be a column named `s` containing time information (typically in seconds, although the units are not used).
`time.limits`	An optional 2-element vector specifying lower and upper bounds in the `s` column of data to which the processing should be constrained.
`btm`	An optional vector of division points breaking the signal into segments. Typically this is obtained from the `Z` data by calling `find.bottom`.
`pch`	Symbol to use for plotting individual current/resistance measurements.
`col`	Colour to use for plotting the current/resistance points.
`lty`	Line type to use for plotting a regression line.
`col.l`	Colour to use for plotting a regression line.
`plot.line`	Whether to plot a regression line. By default this is not done, as a linear regression line corresponds poorly to the resistance behaviour. It may be a useful rough indicator when plotting only current data.
`f`	Function to use for determining the current between hops. By default the `median` is used, but in some cases it may make sense to use a different measure.
`mV`	The pipette potential at which the data were recorded. If not supplied, resistance cannot be calculated and the current data are graphed directly.
`...`	Other graphical parameters to be passed to the `plot` function.

Value

`Z`	The identified bottom of hop positions, in microns.
`pA`	The corresponding median (or other) current through the withdrawn pipette, in picoamps.
`Mohm`	The calculated pipette resistance in megohms. (Only present if `mV` was supplied.)

Author(s)

Matthew Caldwell

References

Caldwell, M., Del Linz, S. J. L., Smart, T. G. S. and Moss, G. W. J. 2012 Method for estimating the tip geometry of scanning ion conductance microscope pipets. *Anal. Chem.* 84(21):8980–8984

See Also

`find.bottom`

Examples

```
# load the example data set
data("break.data")
breakage.plot(break.data, mV=100)
```

`err.breakage` *Error function for estimated tip geometry*

Description

Calculates the residual sum of squares error between measured resistance data and the predicted resistances for a given set of model parameters.

Usage

```
err.breakage(params, data, rho = 51, l = 1000)
```

Arguments

`params`	A vector of the 2 tip geometry parameters, where `params[1]` is the half-cone angle in radians, and `params[2]` is the initial inner radius in microns.
`data`	The recorded data as an XY list or data frame, where `data$x` contains the breakage distances in microns and `data$y` holds the corresponding resistances in megohms.
`rho`	Resistivity of the filler solution, in ohm centimetres. The default value of 51 ohm cm corresponds to a 150 mM solution of KCl.
`l`	The initial length of the pipette tip. Because the resistance is dominated by the narrower tip region, the precise value of this is not usually important provided it is much greater than the breakage distance.

Details

This function quantifies the error between a posited model of the pipette tip geometry and the results actually recorded. It is organised for minimisation by the standard `optim` function, and is used as such within `fit.breakage`. It is not expected to be much used outside that context.

Value

The calculated error value.

Author(s)

Matthew Caldwell

See Also

fit.breakage

Examples

```
# estimate breakage data for one geometry
brks <- 0.2 * 0:10
res <- resist.breakage(brks, theta=4*pi/180, r=0.05, rho=64)
dat <- list(x=brks, y=res)

# compare it to different geometry
err.breakage(c(2*pi/180, 0.1), dat, rho=64)
err.breakage(c(6*pi/180, 0.02), dat, rho=64)
```

---

find.bottom *Identify the bottom points in a hopping-mode position trace*

Description

Identifies the bottom points of each approach in a positional trace from a scanning ion conductance probe operating in hopping mode.

Usage

```
find.bottom(x, window = 50, box.size = 9, clip.ends=TRUE)
```

Arguments

x	A vector of positions recorded over time. Positions are taken to be measured *upward*, i.e. a higher value corresponds to a higher probe position further from the sample. Units are unspecified.
window	Minimum number of previous descending points before a bottom point is accepted. This allows positional wobbles at the bottom and top of a hop to be ignored.
box.size	Width (in samples) of a simple box filter to apply in order to smooth the signal before searching for minima.
clip.ends	Whether to omit bottom points discovered within window points from the end of the signal. This avoids mis-identifying a *de facto* minimum at the end of the signal as being an actual bottom point. TRUE by default.

Value

A logical vector the same length as the input vector `x`, with the bottom points set to `TRUE` and all others `FALSE`.

Author(s)

Matthew Caldwell

Examples

```
# use a simple sine wave as an example signal
t <- 1:1000
hops <- sin(t * pi/100)

# find the bottom points
bots <- find.bottom(hops)

# plot the result
plot(hops, type="l")
abline(v=t[bots], col=2)
```

`fit.breakage` *Fit a tip geometry model to breakage resistance data*

Description

Estimates pipette tip half-cone angle and initial inner radius from the resistance vs breakage distance data. Fitting is performed by minimisation of the residual sum of squares calculated by `err.breakage`. By default, a plot of the data and the fitted model is drawn.

Usage

```
fit.breakage(data, start = list(theta = 3 * pi/180, r = 0.05),
    rho = 51, l = 1000, do.plot = TRUE, ...)
```

Arguments

`data`	Measured breakage data as an XY list or data frame, with `data$x` holding the breakage distances in microns, and `data$y` the corresponding resistance in megohms. For compatibility with `breakage.plot` and `break.clust`, these columns may instead by named `Z` and `Mohm` respectively.
`start`	List of initial values for the optimisation parameters. Must contain values for `theta` and `r`. The error function typically has a well-behaved minimum and the default start values should converge in most cases.
`rho`	Resistivity of the filler solution, in ohm centimetres. The default value of 51 ohm cm corresponds to a 150 mM solution of KCl.

l	The initial length of the pipette tip, in microns. Because the resistance is dominated by the narrower tip region, the precise value of this is not usually important provided it is much greater than the breakage distance.
do.plot	Whether to plot a graph of the data and fit results if successful.
...	Additional parameters passed to the plotting functions.

Details

Optimisation is performed using the L-BFGS-B method of optim, constraining the possible values of r and theta to physically plausible ranges.

Value

theta	The fitted estimate of the half-cone angle, in radians; or NA if the optimisation failed.
r	The fitted estimate of the tip inner radius, in microns; or NA if the optimisation failed.
degrees	The fitted estimate of the half-cone angle, in degrees; or NA if the optimisation failed.
theta.sd	A crude estimate of the standard error in theta, calculated from the Hessian returned by optim; or NA if optimisation failed.
r.sd	A crude estimate of the standard error in r, calculated from the Hessian returned by optim; or NA if optimisation failed.
degrees.sd	A crude estimate of the standard error in degrees, calculated from the Hessian returned by optim; or NA if optimisation failed.
err	The residual sum of squares error between the fitted model predictions and the data; or NA if the optimisation failed.
opt	The result structure returned by optim. If the optimisation failed, this may provide some useful information as to why.

Author(s)

Matthew Caldwell

References

Caldwell, M., Del Linz, S. J. L., Smart, T. G. S. and Moss, G. W. J. 2012 Method for estimating the tip geometry of scanning ion conductance microscope pipets. *Anal. Chem.* 84(21):8980–8984

See Also

fit.breakage

Examples

```
# fake up some breakage data
brks <- sort(abs(0.5 + rnorm(n=15, sd=0.5) * 1:15))
res <- resist.breakage(brks, theta=3*pi/180, r=0.04, rho=64) + rnorm(15)

# fit it
fit.breakage(list(x=brks, y=res))
```

`fit.sensitivity.plot` *Visualise fit sensitivity for a breakage-resistance model*

Description

Plots a contour map of the sensitivity of the residual error of a breakage resistance model to variation of the fitted parameters. This should give some idea of the goodness of the fit.

Usage

```
fit.sensitivity.plot(data, fit, rho = 51, l = 1000,
    r.range = 0.015, theta.range = pi/360, steps = 100,
    nlevels = 200, r.squared = TRUE, bound.at = 0.99, ...)
```

Arguments

`data`	Measured breakage data as an XY list or data frame, with `data$x` holding the breakage distances in microns, and `data$y` the corresponding resistance in megohms. For compatibility with `breakage.plot` and `break.clust`, these columns may instead by named `Z` and `Mohm` respectively.
`fit`	The fitting results list returned by `fit.breakage`
`rho`	Resistivity of the filler solution, in ohm centimetres. Must be the same value used for fitting or the plot will be incorrect.
`l`	The initial length of the pipette tip, in microns. Must be the same value used for fitting or the plot will be incorrect.
`r.range`	The range above and below the fitted inner radius result, `r`, to test and plot, in microns.
`theta.range`	The range above and below the fitted half-cone angle result, `theta`, to test and plot, in radians.
`steps`	The total number of steps to evaluate in both directions.
`nlevels`	The number of contour lines to draw.
`r.squared`	Whether to plot the error value directly, or transform it to plot the coefficient of determination (or R-squared) instead. The latter is plotted by default. It contains the same information in marginally more convenient form, but its use might be considered misleading if taken as implying a linear model.
`bound.at`	A level at which to plot a significance contour. This is on top of the `nlevels` contours automatically positioned by the main plot, and plotted in a different colour.
`...`	Any additional parameters to be passed to the main `contour` plot.

Value

x	The x (theta) values of the plotted surface.
y	The y (radius) values of the plotted surface.
z	The z (error) values of the plotted surface.

Author(s)

Matthew Caldwell

See Also

fit.breakage,

Examples

```
# fake up some breakage data
brks <- sort(abs(0.5 + rnorm(n=15, sd=0.5) * 1:15))
res <- resist.breakage(brks, theta=3*pi/180, r=0.04, rho=64) + rnorm(15)
dat <- list(x=brks, y=res)

# fit it
fit <- fit.breakage(dat, do.plot=FALSE)

# plot the sensitivity surface
fit.sensitivity.plot(dat, fit)
```

---

resist.access *Calculate access resistance for a pipette tip*

Description

Calculates the access resistance for the open pore at the tip of a SICM pipette, using the approximation due to Hall (1975).

Usage

```
resist.access(r, rho = 51)
```

Arguments

r	Radius of the pipette tip opening.
rho	Resistivity of the surrounding conductive material, in ohm centimetres. The default value of 51 ohm cm corresponds to a 150 mM solution of KCl.

Value

The calculated access resistance, in ohms.

Author(s)

Matthew Caldwell

References

Hall, James E. 1975 Access resistance of a small circular pore. *J. Gen. Physiol.*, **66**(4), 531–532

See Also

`resist.access`, `resist.total`, `resist.breakage`

Examples

```
# access resistance to a 50nm pipette tip in solution of resistivity 64 ohm cm
# converting result from ohms to megohms for readability
resist.access(r=0.05, rho=64) * 1e-6
```

`resist.breakage` *Calculate resistance of a pipette tip after breakage*

Description

Calculates the total resistance of a pipette tip, given its initial geometry and the length of a segment broken off the tip.

Usage

```
resist.breakage(x, theta, r, rho = 51, l = 1000)
```

Arguments

`x`	The length of the broken off segment, in microns.
`theta`	Pipette internal half-cone angle, in radians.
`r`	Pipette initial tip radius, in microns.
`rho`	Resistivity of the pipette filler medium, in ohm centimetres. The default value of 51 ohm cm corresponds to a 150 mM solution of KCl.
`l`	The initial length of the pipette tip. Because the resistance is dominated by the narrower tip region, the precise value of this is not usually important provided it is much greater than the breakage distance.

Details

Pipette resistance is calculated exactly as for `resist.total`, after first adjusting the geometry for the broken region. This rearrangement in terms of breakage distance is convenient for fitting resistance changes resulting from a sequence of breaks caused by overshooting the detection range in hopping mode SICM. (See referenced paper for more details.)

Value

The calculated resistance, in megohms. Note that conversion from ohms is done automatically here.

Author(s)

Matthew Caldwell

References

Caldwell, M., Del Linz, S. J. L., Smart, T. G. S. and Moss, G. W. J. 2012 Method for estimating the tip geometry of scanning ion conductance microscope pipets. *Anal. Chem.* 84(21):8980–8984

See Also

resist.access, resist.cone, resist.total

Examples

```
# specify a range of breakage distances in 100 nm steps
brks <- 0.1 * 0:150

# calculate corresponding resistances for a pipette with initial
# tip radius 50nm, half-cone angle 4 degrees, length 1mm
# and filler resistivity 64 ohm cm
res <- resist.breakage(brks, theta=4*pi/180, r=0.05, rho=64)

# plot the predicted curve of resistance against breakage
plot(res ~ brks, type="l", ylim=c(0, max(res)),
     ylab="Resistance (Mohm)", xlab="Breakage Distance (um)")
```

---

resist.cone *Calculate resistance of a truncated conical conductor*

Description

Calculates the resistance of a truncated conical volume of material of known resistivity.

Usage

```
resist.cone(l, r1, r2, rho = 51)
```

Arguments

l	Length of the truncated cone, in microns.
r1	Radius of one end of the truncated cone, in microns.
r2	Radius of the other end of the truncated cone, in microns.
rho	Resistivity of the conductive material, in ohm centimetres. The default value of 51 ohm cm corresponds to a 150 mM solution of KCl.

Value

Calculated resistance, in ohms.

Note

A NaN result will be produced if r1==r2.

Author(s)

Matthew Caldwell

See Also

resist.access, resist.total, resist.breakage

Examples

```
# resistance of truncated conical segment 30 microns long, with end radii 50nm and 500nm
# filled with solution of resistivity 64 ohm cm
# converting result from ohms to megohms for readability
resist.cone(l=30, r1=0.05, r2=0.5, rho=64) * 1e-6
```

---

resist.total *Calculate overall pipette tip resistance*

Description

Calculates the resistance of a pipette tip, as the sum of its internal conical resistance and external access resistance. Pipette geometry is specified in terms of length, tip radius and half-cone angle.

Usage

```
resist.total(r, l, theta, rho = 51)
```

Arguments

r	Pipette tip (inner) radius, in microns.
l	Pipette length, in microns. Note that resistance contribution falls of markedly away from the tip, so the exact length may not be crucially important.
theta	Pipette internal half-cone angle, in radians.
rho	Resistivity of the pipette filler medium, in ohm centimetres. The default value of 51 ohm cm corresponds to a 150 mM solution of KCl.

Value

The calculated total resistance, in ohms.

Author(s)

Matthew Caldwell

See Also

resist.access, resist.cone, resist.breakage

Examples

```
# total resistance for a 30 micron long pipette tip with a half-cone angle of 4 degrees
# and a tip radius of 50nm, with a filler/bath resistivity of 64 ohm cm
# converting result from ohms to megohms for readability
resist.total(r=0.05, l=30, theta=4*pi/180, rho=64) * 1e-6
```

Index