Package 'IsoplotR'

June 22, 2016

Title Statistical Toolbox for Radiometric Geochronology

Version 0.4

Description An R implementation of Ken Ludwig's popular Isoplot add-in to Microsoft Excel. Currently plots U-Pb data on Wetherill and Tera-Wasserburg concordia diagrams. Calculates concordia and discordia ages. Performs linear regression of measurements with correlated errors using the 'York' approach. Generates Kernel Density Estimates (KDEs) and Cumulative Age Distributions (CADs). Future versions will include functionality for the Ar-Ar, Rb-Sr, Sm-Nd, Re-Os, U-Th-He, fission track and cosmogenic nuclide methods, including isochrons, age spectra, ternary diagrams, radial plots, banana diagrams and multidimensional scaling plots. A graphical user interface is provided as an RStudio Shiny app at http://isoplotr.london-geochron.com. Offline access to this interface will be provided at a later point in time.

Author Pieter Vermeesch [aut, cre]

Maintainer Pieter Vermeesch <p.vermeesch@ucl.ac.uk>

Depends R (>= 3.0.0)

Imports methods

License GPL-2

LazyData true

RoxygenNote 5.0.1

R topics documented:

---

age *Calculate isotopic ages*

Description

Calculates U-Pb ages and propagates their analytical uncertainties. Evaluates the equivalence of multiple (^{206}Pb/^{238}U-^{207}Pb/^{235}U or ^{207}Pb/^{206}Pb-^{206}Pb/^{238}U) compositions, computes the weighted mean isotopic composition and the corresponding concordia age using the method of maximum likelihood, computes the mswd of equivalence and concordance and their respective Chi-squared p-values. Performs linear regression of U-Pb data on Wetherill and Tera-Wasserburg concordia diagrams. Computes the upper and lower intercept ages (for Wetherill) or the lower intercept age and the ^{207}Pb/^{206}Pb intercept (for Tera-Wasserburg), taking into account error correlations and decay constant uncertainties.

Usage

```
age(x, ...)

## Default S3 method:
age(x, method = "Pb206U238", dcu = TRUE, ...)

## S3 method for class 'UPb'
age(x, concordia = 1, wetherill = TRUE, dcu = TRUE,
  i = NA, ...)

## S3 method for class 'detritals'
age(x, ...)
```

Arguments

x	a scalar containing an isotopic ratio, a two element vector containing an isotopic ratio and its standard error, or an object of class UPb or detritals.
...	optional arguments
method	one of either 'Pb206U238', 'Pb207U235', or 'Pb207Pb206'
dcu	propagate the decay constant uncertainties?
concordia	scalar flag indicating whether each U-Pb analysis should be considered separately (concordia=1), a concordia age should be calculated from all U-Pb analyses together (concordia=2), or a discordia line should be fit through all the U-Pb analyses (concordia=2).
wetherill	boolean flag to indicate whether the data should be evaluated in Wetherill (TRUE) or Tera-Wasserburg (FALSE) space. This option is only used when concordia=2
i	(optional) index of a particular aliquot

Value

if x is a scalar or a vector, returns the age using the geochronometer given by method and its standard error.

if x has class UPb and concordia=1, returns a table with the following columns: 't.75', 'err[t.75]', 't.68', 'err[t.68]', 't.76','err[t.76]', 't.conc', 'err[t.conc]', containing the 207Pb/235U-age and standard error, the ^{206}Pb/^{238}U-age and standard error, the ^{207}Pb/^{206} Pb-age and standard error, and the concordia age and standard error, respectively.

if x has class UPb and concordia=2, returns a list with the following items:

x a named vector with the (weighted mean) U-Pb composition

cov the covariance matrix of the (mean) U-Pb composition

age the concordia age (in Ma)

age.err the standard error of the concordia age

mswd a list with two items (equivalence and concordance) containing the MSWD (Mean of the Squared Weighted Deviates, a.k.a the reduced Chi-squared statistic outside of geochronology) of isotopic equivalence and age concordance, respectively.

p.value a list with two items (equivalence and concordance) containing the p-value of the Chi-square test for isotopic equivalence and age concordance, respectively.

if x has class UPb and concordia=3, returns a list with the following items:

x a two element vector with the upper and lower intercept ages (if wetherill==TRUE) or the lower intercept age and ^{207}Pb/^{206}Pb intercept (for Tera-Wasserburg)

cov the covariance matrix of the elements in x

Examples

```
data(examples)
print(age(examples$UPb))
print(age(examples$UPb,concordia=1))
print(age(examples$UPb,concordia=2))
```

---

botev *Compute the optimal kernel bandwidth*

Description

Uses the diffusion algorithm of Zdravko Botev (2011) to calculate the bandwidth for kernel density estimation

Usage

```
botev(x)
```

Arguments

x	a vector of ordinal data

Value

a scalar value with the optimal bandwidth

Author(s)

Dzdravko Botev

References

Botev, Z. I., J. F. Grotowski, and D. P. Kroese. "Kernel density estimation via diffusion." The Annals of Statistics 38.5 (2010): 2916-2957.

Examples

```
data(examples)
samp <- examples$DZ[['N1']]
bw <- botev(samp)
print(bw)
```

---

cad *Plot continuous data as cumulative age distributions*

Description

Plot a dataset as a Cumulative Age Distribution (CAD), also known as a 'empirical cumulative distribution function'.

Usage

```
cad(x, method = NA, pch = NA, verticals = TRUE, xlab = "age [Ma]",
  colmap = "heat.colors", col = "black", ...)
```

Arguments

x	an object of class UPb or detritals
method	a string indicating what kind of age should be plotted. If x has class UPb, type could be one of either t.75, t.68 (default), t.76 or t.conc
pch	(optional) plot character
verticals	boolean flag indicating if the horizontal lines of the CAD should be connected by vertical lines
xlab	x-axis label

colmap	an optional string with the name of one of R's built-in colour palettes (e.g., heat.colors, terrain.colors, topo.colors, cm.colors), which are to be used for plotting data of class detritals.
col	colour to give to single sample datasets (i.e. not of class detritals)
...	optional arguments to the generic plot function

Examples

```
data(examples)
cad(examples$DZ)
```

---

concordia *Concordia diagram*

Description

Plot U-Pb data on Wetherill and Tera-Wasserburg concordia diagrams, calculate concordia ages and compositions, evaluates the equivalence of multiple ($^{206}Pb/^{238}U$-$^{207}Pb/^{235}U$ or $^{207}Pb/^{206}Pb$-$^{206}Pb/^{238}U$) compositions, computes the weighted mean isotopic composition and the corresponding concordia age using the method of maximum likelihood, computes the mswd of equivalence and concordance and their respective Chi-squared p-values. Performs linear regression and computes the upper and lower intercept ages (for Wetherill) or the lower intercept age and the $^{207}Pb/^{206}Pb$ intercept (for Tera-Wasserburg), taking into account error correlations and decay constant uncertainties.

Usage

```
concordia(x, limits = NULL, alpha = 0.05, wetherill = TRUE,
  show.numbers = FALSE, ellipse.col = rgb(0, 1, 0, 0.5),
  concordia.col = "darksalmon", dcu = TRUE, show.age = 1)
```

Arguments

x	an object of class UPb
limits	age limits of the concordia line
alpha	confidence cutoff for the error ellipses
wetherill	boolean flag (FALSE for Tera-Wasserburg)
show.numbers	boolean flag (TRUE to show grain numbers)
ellipse.col	background colour of the error ellipses
concordia.col	colour of the concordia line
dcu	show decay constant uncertainty?
show.age	one of either 1: don't show the age 2: calculate the concordia age 3: fit a discordia line

Examples

```
data(examples)
concordia(examples$UPb)
```

`ellipse` *Get coordinates of error ellipse for plotting*

Description

Construct an error ellipse age a given confidence level from its centre and covariance matrix

Usage

```
ellipse(x, y, covmat, alpha = 0.05)
```

Arguments

`x`	x-coordinate (scalar) for the centre of the ellipse
`y`	y-coordinate (scalar) for the centre of the ellipse
`covmat`	covariance matrix of the x-y coordinates
`alpha`	the probability cutoff for the error ellipses

Value

a [50x2] matrix of plot coordinates

Examples

```
x = 99; y = 101;
covmat <- matrix(c(1,0.9,0.9,1),nrow=2)
ell <- ellipse(x,y,covmat)
plot(c(90,110),c(90,110),type='l')
polygon(ell,col=rgb(0,1,0,0.5))
points(x,y,pch=21,bg='black')
```

examples *Example datasets for testing* IsoplotR

Description

U-Pb and detrital zircon datasets

Details

`examples` is a list with two items

`UPb`: an object of class `'UPb'` containing a high precision U-Pb dataset packaged with Ken Ludwig's `Isoplot` program.

`DZ`: an object of class `'detrital'` containing a detrital zircon U-Pb dataset from Namibia.

Author(s)

Ken Ludwig and Pieter Vermeesch

References

Ludwig, K. R. User's manual for Isoplot 3.00: a geochronological toolkit for Microsoft Excel. No. 4. Kenneth R. Ludwig, 2003.

Vermeesch, Pieter, and Eduardo Garzanti. "Making geological sense of 'Big Data' in sedimentary provenance analysis." Chemical Geology 409 (2015): 20-27.

Examples

```
data(examples)
concordia(examples$UPb)
dev.new()
kde(examples$DZ)
```

iratio *Isotopic ratios*

Description

Gets or sets natural isotopic ratios.

Usage

```
iratio(ratio, x = NULL, e = NULL)
```

Arguments

ratio	one of either 'U238U235', 'Ar40Ar36', 'Ar38Ar36', 'Rb85Rb87', 'Sr88Sr86', 'Sr87Sr86', 'Sr84Sr86', 'Re185Re187', 'Os184Os192', 'Os186Os192', 'Os187Os192', 'Os188Os192', 'Os189Os192'
x	new value for ratio
e	new value for its standard error

Value

if x=e=NULL, returns a two-item vector containing the mean value of the requested ratio and its standard error, respectively.

References

Ar: Lee, Jee-Yon, et al. "A redetermination of the isotopic abundances of atmospheric Ar." Geochimica et Cosmochimica Acta 70.17 (2006): 4507-4512.

Rb: Catanzaro, E. J., et al. "Absolute isotopic abundance ratio and atomic weight of terrestrial rubidium." J. Res. Natl. Bur. Stand. A 73 (1969): 511-516.

Sr: Moore, L. J., et al. "Absolute isotopic abundance ratios and atomic weight of a reference sample of strontium." J. Res. Natl.Bur. Stand. 87.1 (1982): 1-8.

Re: Gramlich, John W., et al. "Absolute isotopic abundance ratio and atomic weight of a reference sample of rhenium." J. Res. Natl. Bur. Stand. A 77 (1973): 691-698.

Os: Voelkening, Joachim, Thomas Walczyk, and Klaus G. Heumann. "Osmium isotope ratio determinations by negative thermal ionization mass spectrometry." Int. J. Mass Spect. Ion Proc. 105.2 (1991): 147-159.

U: Hiess, Joe, et al. "$^{238}U/^{235}U$ systematics in terrestrial uranium-bearing minerals." Science 335.6076 (2012): 1610-1614.

Examples

```
# returns the 238U/235U ratio of Hiess et al. (2012):
print(iratio('U238U235'))
# use the 238U/235U ratio of Steiger and Jaeger (1977):
iratio('U238U235',138.88,0)
print(iratio('U238U235'))
```

---

kde *Create (a) kernel density estimate(s)*

Description

Creates one or more kernel density estimates using a combination of the Botev (2010) bandwidth selector and the Abramson (1982) adaptive kernel bandwidth modifier.

Usage

```
kde(x, ...)

## Default S3 method:
kde(x, from = NA, to = NA, bw = NA, adaptive = TRUE,
  log = FALSE, n = 512, plot = TRUE, pch = NA, xlab = "age [Ma]",
  ylab = "", kde.col = rgb(1, 0, 1, 0.6), hist.col = rgb(0, 1, 0, 0.2),
  show.hist = TRUE, bty = "n", binwidth = NA, ncol = NA, ...)

## S3 method for class 'UPb'
kde(x, from = NA, to = NA, bw = NA, adaptive = TRUE,
  log = FALSE, n = 512, plot = TRUE, pch = NA, xlab = "age [Ma]",
  ylab = "", kde.col = rgb(1, 0, 1, 0.6), hist.col = rgb(0, 1, 0, 0.2),
  show.hist = TRUE, bty = "n", binwidth = NA, ncol = NA, type = 4,
  cutoff.76 = 1100, cutoff.disc = c(-15, 5), ...)

## S3 method for class 'detritals'
kde(x, from = NA, to = NA, bw = NA, adaptive = TRUE,
  log = FALSE, n = 512, plot = TRUE, pch = NA, xlab = "age [Ma]",
  ylab = "", kde.col = rgb(1, 0, 1, 0.6), hist.col = rgb(0, 1, 0, 0.2),
  show.hist = TRUE, bty = "n", binwidth = NA, ncol = NA,
  samebandwidth = TRUE, normalise = TRUE, ...)
```

Arguments

x	a vector of numbers or an object of class UPb or detrital
...	optional arguments to be passed on to density
from	minimum age of the time axis. If NULL, this is set automatically
to	maximum age of the time axis. If NULL, this is set automatically
bw	the bandwidth of the KDE. If NULL, bw will be calculated automatically using botev()
adaptive	boolean flag controlling if the adaptive KDE modifier of Abramson (1982) is used
log	transform the ages to a log scale if TRUE
n	horizontal resolution of the density estimate
plot	show the KDE as a plot
pch	the symbol used to show the samples. May be a vector. Set pch = NA to turn them off.
xlab	the label of the x-axis
ylab	the label of the y-axis
kde.col	the fill colour of the KDE specified as a four element vector of r, g, b, alpha values
hist.col	the fill colour of the histogram specified as a four element vector of r, g, b, alpha values

show.hist	boolean flag indicating whether a histogram should be added to the KDE
bty	change to "o", "l", "7", "c", "u", or "]" if you want to draw a box around the plot
binwidth	scalar width of the histogram bins, in Myr if x$log==FALSE, or as a fractional value if x$log==TRUE. Sturges' Rule is used if binwidth==NA
ncol	scalar value indicating the number of columns over which the KDEs should be divided. This option is only used if x is of class detritals.
type	scalar indicating whether to plot the $^{207}Pb/^{235}U$ age (type=1), the $^{206}Pb/^{238}U$ age (type=2), the $^{207}Pb/^{206}Pb$ age (type=3), the $^{207}Pb/^{206}Pb$-$^{206}Pb/^{238}U$ age (type=4), or the (Wetherill) concordia age (type=5)
cutoff.76	the age (in Ma) below which the $^{206}Pb/^{238}U$ and above which the $^{207}Pb/^{206}Pb$ age is used. This parameter is only used if type=4.
cutoff.disc	two element vector with the maximum and minimum percentage discordance allowed between the $^{207}Pb/^{235}U$ and $^{206}Pb/^{238}U$ age (if $^{206}Pb/^{238}U$ < cutoff.76) or between the $^{206}Pb/^{238}U$ and $^{207}Pb/^{206}Pb$ age (if $^{206}Pb/^{238}U$ > cutoff.76). Set cutoff.disc=NA if you do not want to use this filter.
samebandwidth	boolean flag indicating whether the same bandwidth should be used for all samples. If samebandwidth = TRUE and bw = NULL, then the function will use the median bandwidth of all the samples.
normalise	boolean flag indicating whether or not the KDEs should all integrate to the same value.

Value

if plot==TRUE, returns an object of class KDE, i.e. a list containing the following items:

x horizontal plot coordinates

y vertical plot coordinates

bw the base bandwidth of the density estimate

ages the data values from the input to the KDE function

or, if class(x)=='detritals', an object of class KDEs, i.e. a list containing the following items:

kdes a named list with objects of class KDE

from the beginning of the common time scale

to the end of the common time scale

themax the maximum probability density of all the KDEs

xlabel the x-axis label to be used by plot.KDEs

Examples

```
data(examples)
kde(examples$DZ[['N1']],kernel="epanechnikov")
kde(examples$DZ,from=0,to=3000)
```

lambda *Decay constants*

Description

Gets or sets the decay constants of radioactive istopes

Usage

```
lambda(nuclide, x = NULL, e = NULL)
```

Arguments

nuclide	the nuclide name
x	new value for the decay constant
e	new value for the decay constant uncertainty

Value

if x==e==NULL, returns a two-item vector containing the decay constant [in Ma-1] and its standard error, respectively.

Examples

```
print(lambda('U238'))
# use the decay constant of Kovarik and Adams (1932)
lambda('U238',0.0001537,0.0000068)
print(lambda('U238'))
```

read.data *Read geochronology data*

Description

Cast a .csv file or a matrix into one of IsoplotR's data classes

Usage

```
read.data(x, ...)

## Default S3 method:
read.data(x, method = "Pb206U238", format = 1, ...)

## S3 method for class 'matrix'
read.data(x, method = "U-Pb", format = 1, ...)

## S3 method for class 'data.frame'
read.data(x, method = "U-Pb", format = 1, ...)
```

Arguments

x	a file name (.csv format) or matrix
...	optional arguments to the read.csv function
method	one of 'U-Pb', 'Ar-Ar', 'Rb-Sr', 'Sm-Nd', 'Re-Os', 'U-Th-He', 'fission tracks', 'cosmogenic nuclides' or 'other'
format	formatting option, depends on the value of method. If method = 'U-Pb', then format is one of either: 1: 7/6, s[7/6], 6/8, s[6/8], 7/5, s[7/5] (other formats will be added later)

Value

an object of class 'UPb', 'ArAr', 'RbSr', 'SmNd', 'ReOs', 'UThHe', 'fission', 'cosmogenics', or 'other'

Examples

```
# load one of the built-in .csv files:
fname <- system.file("UPb.csv",package="IsoplotR")
UPb <- read.data(fname,'U-Pb')
concordia(UPb)
```

---

settings *Load settings to and from json*

Description

Get and set preferred values for decay constants and isotopic abundances from and to a .json file format

Usage

```
settings(fname = NULL)
```

Arguments

fname	the path of a .json file

Value

if fname==NULL, returns a .json string

Examples

```
json <- system.file("defaults.json",package="IsoplotR")
settings(json)
print(settings())
```

yorkfit *Linear regression of X,Y-variables with correlated errors*

Description

Implements the unified regression algorithm of York et al. (2004) which, although based on least squares, yields results that are consistent with maximum likelihood estimates of Ludwig and Titterington (1994)

Usage

```
yorkfit(X, Y, sX, sY, rXY)
```

Arguments

X	vector of measurements
Y	vector of measurements
sX	standard errors of X
sY	standard errors of Y
rXY	correlation coefficients between X and Y

Value

a five element list containing

a the intercept of the straight line fit

b the slope of the fit

sa the standard error of the intercept

sb the standard error of the slope

References

Ludwig, K. R., and D. M. Titterington. "Calculation of 230ThU isochrons, ages, and errors." Geochimica et Cosmochimica Acta 58.22 (1994): 5031-5042.

York, Derek, et al. "Unified equations for the slope, intercept, and standard errors of the best straight line." American Journal of Physics 72.3 (2004): 367-375.

Examples

```
X <- c(1.550,12.395,20.445,20.435,20.610,24.900,
       28.530,50.540,51.595,86.51,106.40,157.35)
Y <- c(.7268,.7849,.8200,.8156,.8160,.8322,
       .8642,.9584,.9617,1.135,1.230,1.490)
n <- length(X)
sX <- X*0.01
sY <- Y*0.005
```

```
rXY <- rep(0.8,n)
fit <- yorkfit(X,Y,sX,sY,rXY)
covmat <- matrix(0,2,2)
plot(range(X),fit$a+fit$b*range(X),type='l',ylim=range(Y))
for (i in 1:n){
    covmat[1,1] <- sX[i]^2
    covmat[2,2] <- sY[i]^2
    covmat[1,2] <- rXY[i]*sX[i]*sY[i]
    covmat[2,1] <- covmat[1,2]
    ell <- ellipse(X[i],Y[i],covmat,alpha=0.05)
    polygon(ell)
}
```

Index