Package 'IsoplotR'

May 24, 2016

Title Statistical Toolbox for Radiometric Geochronology

Version 0.3

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, taking into account error correlations. 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, kernel density estimates, radial plots, banana diagrams and multidimensional scaling plots. A graphical user interface is provided as an RStudio Shiny app.

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:

`concordia.age` *Calculate U-Pb concordia ages*

Description

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.

Usage

```
concordia.age(x, wetherill = TRUE, dcu = TRUE)
```

Arguments

`x`	an object of class `UPb`
`wetherill`	boolean flag to indicate whether the data should be evaluated in Wetherill (`TRUE`) or Tera-Wasserburg (`FALSE`) space
`dcu`	propagate the decay constant uncertainties?

Value

a list with the following items:

`x`: a named vector with the weighted mean U-Pb composition

`x.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.

Examples

```
data(UPb)
fit <- concordia.age(UPb)
print(paste('age = ',fit$age,'+/-',fit$age.err,'Ma, MSWD = ',fit$mswd))
```

concordia.plot *Concordia diagram*

Description

Wetherill and Tera-Wasserburg concordia diagrams

Usage

```
concordia.plot(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 = 0)
```

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 0: don't show the age 1: calculate the concordia age 2: fit a discordia line

Examples

```
data(UPb)
concordia.plot(UPb)
```

discordia.age *Linear regression on a U-Pb concordia diagram*

Description

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

```
discordia.age(x, wetherill = TRUE, dcu = TRUE)
```

Arguments

x	an object of class UPb
wetherill	boolean flag to indicate whether the data should be evaluated in Wetherill (TRUE) or Tera-Wasserburg (FALSE) space
dcu	propagate the decay constant uncertainties?

Value

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(UPb)
fit <- discordia.age(UPb)
print(paste('lower intercept = ',fit$x[1],'+/-',sqrt(fit$cov[1,1]),'Ma'))
```

---

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')
```

I.R *Isotopic ratios*

Description

Gets or sets natural isotopic ratios.

Usage

```
I.R(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. "238U/235U systematics in terrestrial uranium-bearing minerals." Science 335.6076 (2012): 1610-1614.

Examples

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

`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 into one of IsoplotR's data classes

Usage

```
read.data(fname, method = "U-Pb", format = 1, ...)
```

Arguments

fname	file name (.csv format)
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]
...	optional arguments to the read.csv function

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.plot(UPb)
```

read.matrix *Read geochronology data*

Description

Cast a matrix into one of IsoplotR's data classes

Usage

```
read.matrix(x, method = "U-Pb", format = 1)
```

Arguments

x	a matrix
method	see read.data for details
format	see read.data for details

Value

see read.data for details

Examples

```
# load one of the built-in .csv files:
fname <- system.file("UPb.csv",package="IsoplotR")
dat <- read.csv(fname,header=TRUE)
UPb <- read.matrix(dat,method='U-Pb',format=1)
concordia.plot(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())
```

UPb *An example U-Pb dataset*

Description

An example U-Pb dataset provided with Ludwig's Isoplot add-in

Details

`UPb` is an object of class `UPb`, i.e. a list with two items

`x`: a matrix formatted according to `format`

`format`: an integer defining the format of `x`. Options are:

`1`: 7/6, s[7/6], 6/8, s[6/8], 7/5, s[7/5]

Author(s)

Ken Ludwig and Pieter Vermeesch

Examples

```
data(UPb)
concordia.plot(UPb)
```

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