Package 'isogeochem'

September 3, 2021

Type Package

Title Tools For Carbonate Isotope Geochemistry

Version 0.0.0.9200

Author David Bajnai

Description This collection of functions makes working with stable oxygen and clumped isotope data simpler and more reproducible.

License CC BY 4.0

URL https://github.com/davidbajnai/isogeochem

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

Suggests knitr,
rmarkdown

VignetteBuilder knitr

Imports shades,
graphics,
reconPlots (>= 0.1.0),
grDevices

Remotes andrewheiss/reconPlots

R topics documented:

`a18c_w` *Calculate the 18O/16O fractionation factor between carbonate and water*

Description

`a18c_w()` calculates the equilibrium 18O/16O oxygen isotope fractionation factor between carbonate and water for a given temperature.

Usage

```
a18c_w(temp, min = "calcite", eq = "Daeron19")
```

Arguments

`temp`	Crystallization temperature in degrees Celsius.
`min`	Mineralogy. Options are `"calcite"` (default), `"aragonite"`, and `"dolomite"`.
`eq`	Equation used to calculate the equilibrium 18O/16O oxygen isotope fractionation factor between calcium carbonate and water. For calcite choose from `"Daeron19"` (default), `"Watkins13"`, `"Coplen07"`, `"KO97"` , and `"FO77"`. For aragonite and dolomite eq need not be specified.

Value

Returns the equilibrium 18O/16O oxygen isotope fractionation factor "alpha"

References

Calcite:

O'Neil, J. R., Clayton, R. N., & Mayeda, T. K. (1969). Oxygen isotope fractionation in divalent metal carbonates. The Journal of Chemical Physics, 51(12), 5547-5558. https://doi.org/10.1063/1.1671982

Kim, S.-T., & O'Neil, J. R. (1997). Equilibrium and nonequilibrium oxygen isotope effects in synthetic carbonates. Geochimica et Cosmochimica Acta, 61(16), 3461-3475. https://doi.org/10.1016/S0016-7037(97)00169-5

Coplen, T. B. (2007). Calibration of the calcite–water oxygen-isotope geothermometer at Devils Hole, Nevada, a natural laboratory. Geochimica et Cosmochimica Acta, 71(16), 3948-3957. https://doi.org/10.1016/j.gca.2007.05.028

Watkins, J. M., Nielsen, L. C., Ryerson, F. J., & DePaolo, D. J. (2013). The influence of kinetics on the oxygen isotope composition of calcium carbonate. Earth and Planetary Science Letters, 375, 349-360. https://doi.org/10.1016/j.epsl.2013.05.054

Daëron, M., Drysdale, R. N., Peral, M., Huyghe, D., Blamart, D., Coplen, T. B., et al. (2019). Most Earth-surface calcites precipitate out of isotopic equilibrium. Nature Communications, 10, 429. https://doi.org/10.1038/s41467-019-08336-5

Aragonite:

Dettman, D. L., Reische, A. K., & Lohmann, K. C. (1999). Controls on the stable isotope composition of seasonal growth bands in aragonitic fresh-water bivalves (unionidae). Geochimica et Cosmochimica Acta, 63(7-8), 1049-1057. https://doi.org/10.1016/s0016-7037(99)00020-4

Dolomite:

Vasconcelos, C., McKenzie, J. A., Warthmann, R., & Bernasconi, S. M. (2005). Calibration of the d18O paleothermometer for dolomite precipitated in microbial cultures and natural environments. Geology, 33(4), 317-320. https://www.doi.org/10.1130/g20992.1

Examples

```
a18c_w(25, "calcite", "Coplen07")   # Returns 1.030249 and 30.25
a18c_w(25, "aragonite") # Returns 1.000913 and 0.91
a18c_w(25, "dolomite")  # Returns 1.031456 and 31.46
```

`d17Oc` *Triple oxygen isotope values*

Description

`d17Oc()` calculates equilibrium calcite d18O, d17O, and D17O values for a given temperature.

Usage

```
d17Oc(temp, d18Ow_VSMOW, eq18 = "Daeron19", lambda = 0.528)
```

Arguments

`temp`	Calcite growth temperature in degrees Celsius.
`d18Ow_VSMOW`	Water d18O value expressed on the VSMOW scale (parts per mille).
`eq18`	Equation used to calculate the equilibrium 18O/16O oxygen isotope fractionation factor between calcite and water. Options are `"Daeron19"` (default), `"Watkins13"`, `"Coplen07"`, `"KO97"` , and `"FO77"`.
`lambda`	Triple oxygen isotope reference slope. Default is `0.528`.

Details

$$\theta_{A/B} = \frac{\alpha^{17}_{A/B}}{\alpha^{18}_{A/B}}$$

$$\delta'^{17}O_{w,VSMOW} = \beta \times \delta'^{18}O_{w,VSMOW} + \gamma \text{ where } \beta = 0.528 \text{ and } \gamma = 0$$

$$\Delta^{17}O = \delta'^{17}O_{c,VSMOW} - \lambda \times \delta'^{18}O_{c,VSMOW}$$

Value

Returns a data frame with the carbonate d18O, d17O, and D17O values expressed on the VSMOW scale (all in parts per mille).

References

Guo, W., & Zhou, C. (2019). Triple oxygen isotope fractionation in the DIC-H2O-CO2 system: A numerical framework and its implications. Geochimica et Cosmochimica Acta, 246, 541-564. https://www.doi.org/10.1016/j.gca.2018.11.018

See Also

Other equilibrium_carbonate: D47c(), D48c(), d18Oc()

Examples

```
d17Oc(10,-1) # Returns d18Oc = 32.44, d17Oc = 16.91, D17O = -0.084
d17Oc(10,-1)[,3] # Returns D17O = -0.084
prime(d17Oc(10,-1)[,2]) # Returns d'17O = 16.77
```

---

d18Oc *Equilibrium carbonate d18O value*

Description

d18Oc() calculates the equilibrium d18O value of a carbonate grown at a given temperature.

Usage

```
d18Oc(temp, d18Ow_VSMOW, min = "calcite", eq = "Daeron19")
```

Arguments

temp	Crystallization temperature in degrees Celsius.
d18Ow_VSMOW	Water d18O value expressed on the VSMOW scale (parts per mille).
min	Mineralogy. Options are "calcite" (default), "aragonite", and "dolomite".
eq	Equation used to calculate the equilibrium 18O/16O oxygen isotope fractionation factor between calcium carbonate and water. Options depend on mineralogy. For calcite choose from "Daeron19" (default), "Watkins13", "Coplen07", "KO97" , and "FO77". For aragonite and dolomite the eq need not be specified.

Value

Returns the equilibrium carbonate d18O value expressed on the VSMOW scale (parts per mille).

Note

Use to_vsmow() and to_vpdb() to convert between the VSMOW and VPDB scales.

References

References are listed in the description of a18c_w().

See Also

d18Ow() calculates the d18O value of the ambient water from the d18O value of a carbonate and its growth temperature.

Other equilibrium_carbonate: D47c(), D48c(), d17Oc()

Examples

```
d18Oc(33.7, -13.54, eq="Coplen07") # Returns 14.58
to_vpdb(d18Oc(12, -6.94, min="aragonite")) # Returns -5.21
d18Oc(25, -10.96, min="dolomite") # Returns 20.15
```

---

d18Ow *Calculate water d18O value*

Description

d18Ow() calculates the d18O value of the ambient water. from the d18O value of a carbonate and its growth temperature.

Usage

```
d18Ow(temp, d18Oc_VSMOW, min = "calcite", eq = "Daeron19")
```

Arguments

temp	Crystallization temperature, in degrees Celsius.
d18Oc_VSMOW	Carbonate d18O value expressed on the VSMOW scale (parts per mille).
min	Mineralogy. Options are "calcite" (default), "aragonite", and "dolomite".
eq	Equation used to calculate the equilibrium 18O/16O oxygen isotope fractionation factor between calcium carbonate and water. Options depend on mineralogy. For calcite choose from "Daeron19" (default), "Watkins13", "Coplen07", "KO97" , and "FO77". For aragonite and dolomite the eq need not be specified.

Value

Returns the water d18O value expressed on the VSMOW scale (parts per mille).

Note

Use to_vsmow() and to_vpdb() to convert between the VSMOW and VPDB scales.

References

References are listed in the description of a18c_w().

See Also

d18Oc() calculates the equilibrium d18O value of a carbonate grown at a given temperature.

temp_d18O() calculates growth temperatures from oxygen isotope data.

Examples

```
d18Ow(33.7, 14.58, "calcite", "Coplen07") # Returns -13.54
d18Ow(25, to_vsmow(-7.47), "aragonite") # Returns -6.53
d18Ow(25, 20.43, "dolomite") # Returns -10.69
```

D47c *Carbonate D47 for a given temperature*

Description

`D47c()` calculates the equilibrium carbonate D47 value for a given temperature.

Usage

```
D47c(temp, eq = "Fiebig21")
```

Arguments

`temp`	Carbonate growth temperature in degrees Celsius.
`eq`	Equation used for the calculation. Options are `"Fiebig21"` (default) and `"Petersen19"`. `"Fiebig21"` refers to the CDES90 calibration in Fiebig et al. (2021). `"Petersen19"` refers to the synthetic-only D47-RFACBr,WD "UNICAL" calibration of Petersen et al. (2019).

Value

Returns the carbonate D47 value expressed on the CDES90 scale.

References

Petersen, S. V., Defliese, W. F., Saenger, C., Daëron, M., Huntington, K. W., John, C. M., et al. (2019). Effects of improved 17O correction on interlaboratory agreement in clumped isotope calibrations, estimates of mineral-specific offsets, and temperature dependence of acid digestion fractionation. Geochemistry, Geophysics, Geosystems, 20(7), 3495-3519. https://www.doi.org/10.1029/2018GC008127

Fiebig, J., Daëron, M., Bernecker, M., Guo, W., Schneider, G., Boch, R., et al. (2021). Calibration of the dual clumped isotope thermometer for carbonates. Geochimica et Cosmochimica Acta. https://www.doi.org/10.1016/j.gca.2021.07.012

See Also

`temp_D47()`

Other equilibrium_carbonate: `D48c()`, `d17Oc()`, `d18Oc()`

Examples

```
D47c(33.7) # Returns 0.5713
```

D48c *Carbonate D48 for a given temperature*

Description

`D48c()` calculates the equilibrium carbonate D48 value for a given temperature.

Usage

```
D48c(temperature, eq = "Fiebig21")
```

Arguments

`temperature`	Carbonate growth temperature in degrees Celsius.
`eq`	Equation used for the calculation. Options are `"Fiebig21"` (default) and `"Swart21"`. `"Fiebig21"` refers to the CDES90 calibration in Fiebig et al. (2021). `"Swart21"` refers to the CDES90 "PBLM1" calibration in Swart et al. (2021).

Value

Returns the carbonate D48 value expressed on the CDES90 scale.

References

Fiebig, J., Daëron, M., Bernecker, M., Guo, W., Schneider, G., Boch, R., et al. (2021). Calibration of the dual clumped isotope thermometer for carbonates. Geochimica et Cosmochimica Acta. https://www.doi.org/10.1016/j.gca.2021.07.012

Swart, P. K., Lu, C., Moore, E., Smith, M., Murray, S. T., & Staudigel, P. T. (2021). A calibration equation between D48 values of carbonate and temperature. Rapid Communications in Mass Spectrometry, 35(17), e9147. https://www.doi.org/10.1002/rcm.9147

See Also

Other equilibrium_carbonate: `D47c()`, `d17Oc()`, `d18Oc()`

Examples

```
D48c(33.7) # Returns 0.237
D48c(33.7,"Swart21") # Returns 0.239
```

mix_d17O *Mixing curves in triple oxygen isotope space*

Description

`mix_d17O()` produces mixing curves in in triple oxygen isotope space (d18O vs. D17O).

Usage

```
mix_d17O(d18O_A, d17O_A, d18O_B, d17O_B, lambda = 0.528)
```

Arguments

`d18O_A`	d18O value of component A (parts per mille).
`d17O_A`	d17O value of component A (parts per mille).
`d18O_B`	d18O value of component B (parts per mille).
`d17O_B`	d17O value of component B (parts per mille).
`lambda`	Triple oxygen isotope reference slope. Default `0.528`.

Value

Returns a data frame:

- d18O value of the mixture at % mixing
- d18O value of the mixture % mixing
- % mixing: from 100% A and 0% B to 0% A and 100% B.

See Also

`d17Oc()` calculates equilibrium calcite d18O, d17O, and D17O values for a given temperature.

Examples

```
# Mixing between a Mesozoic marine carbonate and a diagentic carbonate
mix_d17O(d17Oc(10,-1)[1],d17Oc(10,-1)[2],d17Oc(100,0)[1],d17Oc(100,0)[2])
```

prime *Converting delta to delta prime*

Description

`prime()` converts "classical delta" values to "delta prime" values.

Usage

```
prime(classical)
```

Arguments

classical	"Classical delta" values to be converted (parts per mille).

Details

$$\delta'^{17}O = 1000 \times \ln(\frac{\delta^{17}O}{1000} + 1)$$

Value

Returns the "delta prime" value (parts per mille).

See Also

unprime() converts "delta prime" values to "classical delta" values.

Examples

```
prime(10)
```

---

temp_d18O *Oxygen isotope thermometry*

Description

temp_d18O() calculates carbonate growth temperature from oxygen isotope data.

Usage

```
temp_d18O(d18Oc_VSMOW, d18Ow_VSMOW, eq = "Daeron19")
```

Arguments

d18Oc_VSMOW	Carbonate d18O value expressed on the VSMOW scale (parts per mille).
d18Ow_VSMOW	Water d18O value expressed on the VSMOW scale (parts per mille).
eq	Equation used to calculate the equilibrium 18O/16O oxygen isotope fractionation factor between calcite and water. For calcite choose from "Daeron19" (default), "Watkins13", "Coplen07", "KO97" , and "FO77".

Value

Returns the carbonate growth temperature in degrees Celsius.

Note

Use to_vsmow() and to_vpdb() to convert between the VSMOW and VPDB scales.

References

References are listed in the description of a18c_w().

See Also

d18Oc() calculates the equilibrium d18O value of a carbonate grown at a given temperature.

d18Ow() calculates the d18O value of the ambient water from the d18O value of a carbonate and its growth temperature.

Other thermometry: temp_D47(), temp_D48()

Examples

```
temp_d18O(14.58, -13.54, "Coplen07") # Returns 33.7
```

---

temp_D47 *Clumped isotope thermometry*

Description

temp_D47() calculates carbonate growth temperature from D47 value.

Usage

```
temp_D47(D47_CDES90, D47_error, eq = "Petersen19")
```

Arguments

D47_CDES90	Carbonate D47 values expressed on the CDES90 scale: referenced to 90°C acid digestion.
D47_error	Error on the D47 value. Optional.
eq	Equation used for the calculation. Options are "Petersen19" (default) and "Kele15". "Petersen19" refers to the synthetic-only IUPAC-reprocessed "Br "UNICAL" calibration of Petersen et al. (2019). "Kele14" refers to the Kele et al. (2015) calibration reprocessed by Bernasconi et al. (2020) using the IUPAC parameters.

Value

Returns the carbonate growth temperature in degrees Celsius, and — if D47_error is specified — also the error.

References

References are listed at D47c().

See Also

D47c() calculates the equilibrium carbonate D47 value for a given temperature.

Other thermometry: temp_D48(), temp_d18O()

Examples

```
temp_D47(0.580) # Returns 33.7
temp_D47(0.580, 0.004) # Returns 33.7 and 1.9
```

temp_D48 *Dual clumped isotope thermometry*

Description

`temp_D48()` calculates carbonate growth temperature from D47 and D48 values.

Usage

```
temp_D48(
  D47_CDES90,
  D48_CDES90,
  D47_error,
  D48_error,
  ks,
  add = FALSE,
  col,
  pch
)
```

Arguments

`D47_CDES90`	Carbonate D47 values expressed on the CDES90 scale.
`D48_CDES90`	Carbonate D48 values expressed on the CDES90 scale.
`D47_error`	Error on the D47 value. Optional.
`D48_error`	Error on the D48 value. Optional.
`ks`	Kinetic slope. Has to be negative!
`add`	Add graphics to an already existing plot? Default `FALSE`.
`col`	Graphical parameter. Optional.
`pch`	Graphical parameter. Optional.

Details

The function calculates a D47 value as an intersect of two curves:

- the equilibrium D47 vs D48 curve from Fiebig et al. (2021)
- the kinetic slope

The resulting D47 value is then converted to temperature using the `D47c()` function, i.e., by default the equation of Petersen et al. (2019). This is not consistent and I will fix it in a later version. In any case, the resulting discrepancy is smaller than the temperature error.

Value

Returns the carbonate growth temperature in degrees Celsius.

References

References are listed at `D48c()` and `D47c()`.

See Also

D47c() calculates the equilibrium carbonate D47 value for a given temperature.

D48c() calculates the equilibrium carbonate D48 value for a given temperature.

Other thermometry: temp_D47(), temp_d18O()

Examples

```
temp_D48(0.617, 0.139, ks = -0.6) # Returns 44
temp_D48(0.546, 0.277, ks = -1)   # Returns 33
```

---

to_vpdb *Converting between VSMOW and VPDB scales*

Description

to_vpdb() converts d18O values expressed on the VSMOW scale to the VPDB scale.

Usage

```
to_vpdb(d18O_VSMOW, eq = "IUPAC")
```

Arguments

d18O_VSMOW	d18O values expressed on the VSMOW scale (parts per mille).
eq	Equation used for the conversion. Options are "IUPAC", and "Coplen83". The default is "IUPAC": the IUPAC recommended equation listed in Brand et al. (2014) and Kim et al. (2015). To use the equation listed in Coplen et al. (1983) and the Hoefs book, set the parameter to "Coplen83".

Details

The IUPAC recommended equation to convert between the scales is:

$$\delta^{18}O_{VPDB} = 0.97001 \times \delta^{18}O_{VSMOW} - 29.99$$

Value

Returns the d18O value expressed on the VPDB scale (parts per mille).

References

References are listed at to_vsmow().

See Also

to_vsmow() converts d18O values expressed on the VPDB scale to the VSMOW scale.

Examples

```
to_vpdb(0) # Returns -29.99
to_vpdb(0, eq = "Coplen83") # Returns -29.98
```

to_vsmow *Converting between VPDB and VSMOW scales*

Description

`to_vsmow()` converts d18O values expressed on the VPDB scale to the VSMOW scale.

Usage

```
to_vsmow(d18O_VPDB, eq = "IUPAC")
```

Arguments

`d18O_VPDB`	d18O values expressed on the VPDB scale (parts per mille).
`eq`	Equation used for the conversion. Options are `"IUPAC"`, and `"Coplen83"`. The default is `"IUPAC"`: the IUPAC recommended equation listed in Brand et al. (2014) and Kim et al. (2015). To use the equation listed in Coplen et al. (1983) and the Hoefs book, set the parameter to `"Coplen83"`.

Details

The IUPAC recommended equation to convert between the scales is:

$$\delta^{18}O_{VSMOW} = 1.03092 \times \delta^{18}O_{VPDB} + 30.92$$

Value

Returns the d18O value expressed on the VSMOW scale (parts per mille).

References

Coplen, T. B., Kendall, C., & Hopple, J. (1983). Comparison of stable isotope reference samples. Nature, 302, 236-238. https://doi.org/10.1038/302236a0

Brand, W. A., Coplen, T. B., Vogl, J., Rosner, M., & Prohaska, T. (2014). Assessment of international reference materials for isotope-ratio analysis (IUPAC Technical Report). Pure and Applied Chemistry, 86(3), 425-467. https://doi.org/10.1515/pac-2013-1023

Kim, S.-T., Coplen, T. B., & Horita, J. (2015). Normalization of stable isotope data for carbonate minerals: Implementation of IUPAC guidelines. Geochimica et Cosmochimica Acta, 158, 276-289. https://doi.org/10.1016/j.gca.2015.02.011

See Also

`to_vpdb()` converts d18O values expressed on the VSMOW scale to the VPDB scale.

Examples

```
to_vsmow(0) # Returns 30.92
to_vsmow(0, eq = "Coplen83") # Returns 30.91
```

unprime *Converting delta prime to delta*

Description

`unprime()` converts "delta prime" values to "classical delta" values.

Usage

```
unprime(prime)
```

Arguments

`prime`	"Delta prime" values to be converted (parts per mille).

Details

$$\delta^{17}O = 1000 \times e^{(\frac{\delta'^{17}O}{1000}+1)}$$

Value

Returns the "classical delta" value (parts per mille).

See Also

`prime()` converts "classical delta" values to "delta prime" values.

Examples

```
unprime(9.950331)
```

xdic *Dissolved inorganic carbon species*

Description

`xdic()` calculates the relative abundance of the DIC species as a function of solution temperature, pH, and salinity.

Usage

```
xdic(temp, pH, S)
```

Arguments

`temp`	The temperature of the solution in degrees Celsius.
`pH`	The pH of the solution.
`S`	The salinity of the solution in parts per mille.

Value

Returns a data frame with the relative abundance of the DIC species.

Index