Package ‘bayclumpr’

March 29, 2023

Type Package

Title Bayesian Analysis of Clumped Isotope Datasets

Version 0.1.0

URL https://bayclump.tripatilab.epss.ucla.edu/,
      https://tripati-lab.github.io/bayclumpr/

BugReports https://github.com/Tripati-Lab/bayclumpr/issues

Description Simulating synthetic clumped isotope dataset, fitting
    linear regression models under Bayesian and non-Bayesian frameworks,
    and generating temperature reconstructions for the same two approaches.
    Please note that models implemented in this package are described
    in Roman-Palacios et al. (2021) <doi:10.1002/essoar.10507995.1>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.0

Imports parallel, loo, deming, IsoplotR, rstan, stats

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation no

Author Cristian Roman Palacios [aut, cre]
      (<https://orcid.org/0000-0003-1696-4886>),
      Hannah M. Carroll [aut] (<https://orcid.org/0000-0003-3343-3358>),
      Aradhna Tripati [aut] (<https://orcid.org/0000-0002-1695-1754>)

Maintainer Cristian Roman Palacios <cromanpa94@arizona.edu>

Repository CRAN

Date/Publication 2023-03-29 15:30:06 UTC
Bayesian regressions to calibrate the clumped isotopes paleothermometer using stan.

Description

Bayesian regressions to calibrate the clumped isotopes paleothermometer using stan.

Usage

```r
cal.bayesian(
  calibrationData,
  numSavedSteps = 3000,
  priors = "Informative",
  MC = TRUE
)
```

Arguments

- `calibrationData` The target calibration dataset.
- `numSavedSteps` Number of MCMC iterations to save.
- `priors` Either Informative, Weak, or Uninformative on the slope and intercept.
- `MC` Multicore (TRUE/FALSE)

Value

A list the raw models fit in stan and a multi-model comparison based on l00.
cal.ci  

This function is used to generate CI estimates at given intervals. It is currently used for plotting in BayClump.

Description

This function is used to generate CI estimates at given intervals. It is currently used for plotting in BayClump.

Usage

cal.ci(data, from, to, length.out = 100)

Arguments

data A data.frame with two columns named as beta and alpha. This should be the result of bootstrapping or the posterior distribution for a given calibration set.
from the lower limit in x.
to the upper limit in x.
length.out the number of breaks.

Value

A data.frame or list of data.frames with the confidence interval for a given model in a given range of X.

cal.dataset  

Generate a synthetic dataset for clumped isotopes calibrations

Description

Generate a synthetic dataset for clumped isotopes calibrations

Usage

cal.dataset(error = "S1", nobs = 1000)

Arguments

everror Error scenario: low (S1), Intermediate (S2), or High (S3)
nobs Number of observations in the simulated dataset

Value

A data.frame with true and measured values as well as their uncertainties.
cal.deming

*Fit Deming regression models on a given calibration dataset*

**Description**

Fit Deming regression models on a given calibration dataset

**Usage**

```r
cal.deming(data, replicates, samples = NULL)
```

**Arguments**

- `data`: The calibration dataset
- `replicates`: Number of bootstrap replicates
- `samples`: Number of samples per bootstrap replicate

**Value**

A `data.frame` with replicate-level regression parameters

---

cal.ols

*Fit OLS regression models on a given calibration dataset*

**Description**

Fit OLS regression models on a given calibration dataset

**Usage**

```r
cal.ols(data, replicates, samples = NULL)
```

**Arguments**

- `data`: The calibration dataset
- `replicates`: Number of bootstrap replicates
- `samples`: Number of samples per bootstrap replicate

**Value**

A `data.frame` with replicate-level regression parameters
cal.prior

Generate a dataset reflecting the priors used to run the analyses

Description
Generate a dataset reflecting the priors used to run the analyses

Usage
```r
cal.prior(prior, n = 1000)
```

Arguments
- `prior`  Informative or not
- `n` number of observations to simulate

Value
A `data.frame` with prior distributions.

cal.wols

Fit weighted OLS regression models on a given calibration dataset

Description
Fit weighted OLS regression models on a given calibration dataset

Usage
```r
cal.wols(data, replicates, samples = NULL)
```

Arguments
- `data` The calibration dataset
- `replicates` Number of bootstrap replicates
- `samples` Number of samples per bootstrap replicate

Value
A `data.frame` with replicate-level regression parameters
**cal.york**  
*Fit York regression models on a given calibration dataset*

**Description**  
Fit York regression models on a given calibration dataset

**Usage**  
```r
cal.york(data, replicates, samples = NULL)
```

**Arguments**
- `data`  
The calibration dataset
- `replicates`  
Number of bootstrap replicates
- `samples`  
Number of samples per bootstrap replicate

**Value**  
a `data.frame` with replicate-level regression parameters

---

**rec.bayesian**  
*This function generate temperature predictions (in $10^6/T^2$) based on a calibration dataset and target D47. Note that this approach additionally accounts for measured error in the target D47. This approach is congruent with the one used in McClelland et al. (2022).*

**Description**  
This function generate temperature predictions (in $10^6/T^2$) based on a calibration dataset and target D47. Note that this approach additionally accounts for measured error in the target D47. This approach is congruent with the one used in McClelland et al. (2022).

**Usage**  
```r
rec.bayesian(  
calModel,  
recData,  
iter = 1000,  
mixed = FALSE,  
postcalsamples = NULL,  
MC = TRUE  
)
```
**rec.clumped**

**Arguments**

- `calModel` The stan model to be analyzed.
- `recData` The reconstruction dataset.
- `iter` Number of replicates to retain.
- `mixed` whether the model `calModel` is mixed or not.
- `postcalsamples` Number of posterior samples to analyze from the calibration step.
- `MC` Multicore (TRUE/FALSE)

**Value**

A `data.frame` with temperature reconstructions and the original values used in the reconstruction.

---

**Description**

This function performs temp reconstruction ($10^6/T^2$ with $T$ in K) for multiple replicates of the same target.

**Usage**

```
rec.clumped(recData, obCal)
```

**Arguments**

- `recData` Reconstruction dataset
- `obCal` A `data.frame` summarizing the distribution of slopes and intercepts

**Value**

A `data.frame` with temperature reconstructions and the original values used in the reconstruction.
Generate a dataset reflecting the priors used to run the analyses

Usage

    rec.prior(prior, n = 1000)

Arguments

    prior | Informative or not
    n    | number of observations to simulate

Value

    A data.frame with prior distributions.
Index

cal.bayesian, 2
rec.bayesian, 6

cal.ci, 3
rec.clumped, 7

cal.dataset, 3
rec.clumped, 7

cal.deming, 4
rec.prior, 8

cal.ols, 4

cal.prior, 5

cal.wols, 5

cal.york, 6
rec.prior, 8