Package ‘assignR’

February 24, 2022

Type Package
Title Infer Geographic Origin from Isotopic Data
Version 2.2.0
Author Chao Ma, Gabe Bowen
Maintainer Gabe Bowen <gabe.bowen@utah.edu>
Imports raster, sp, rgdal, maptools, mvnfast, rlang, geosphere
Depends R (>= 3.5)
Suggests knitr, rmarkdown, testthat, covr
VignetteBuilder knitr
License GPL-3
Language en-US
Encoding UTF-8
LazyData true
RoxygenNote 7.1.1
NeedsCompilation no
Repository CRAN
Date/Publication 2022-02-24 10:20:05 UTC

R topics documented:

assignR ................................................................. 2
c.wDist ................................................................. 3
calRaster ............................................................... 4
d2h_LrNA ............................................................... 5
getIsoscapes .......................................................... 6
**assignR**

Tools for inferring geographic origin from isotopic data

**Description**

Routines for rescaling isoscapes using known-origin tissue isotope data, assigning origin of unknown samples, and summarizing and assessing assignment results.

**Author(s)**

**Maintainer:** Gabe Bowen <gabe.bowen@utah.edu>  **Authors:** Chao Ma, Gabe Bowen

**See Also**

[https://spatial-lab.github.io/assignR/](https://spatial-lab.github.io/assignR/)
Description
Combine statistics from one or more \texttt{wDist} objects in a single data frame.

Usage
\begin{verbatim}
## S3 method for class 'wDist'
c(...)
\end{verbatim}

Arguments
\begin{verbatim}
... One or more wDist objects
\end{verbatim}

Value
data.frame containing sample IDs, distance, and bearing statistics for each sample in ...

See Also
\texttt{wDist}

Examples
\begin{verbatim}
library(sp)

# load North America boundary and global isoscape
data("naMap")
data("d2h_lRNA")

# load hydrogen isotope data for human hair in North America
d = subOrigData(group = "Modern human", mask = naMap, niter = 100)

# rescale from environmental isoscape to tissue isoscape
r = calRaster(known = d, isoscape = d2h_lRNA, mask = naMap)

# four unknown-origin examples
id = c("A", "B", "C", "D")
d2H = c(-110, -90, -105, -102)
un = data.frame(id,d2H)

# posterior probabilities
pp = pdRaster(r, unknown = un, mask = naMap)

# random collection locations
sites = d$data[sample(seq(length(d$data)), 4),]
\end{verbatim}
# generate a wDist object
wd = wDist(pp, sites)

# combine stats and print
c(wd)

---

calRaster  
Rescale isoscape using linear regression

Description

This function uses known-origin tissue data to rescale a map of environmental isotope values to a map of tissue value (and associated uncertainty) using a linear regression model.

Usage

calRaster(known, isoscape, mask = NULL, interpMethod = 2, NA.value = NA, ignore.NA = TRUE, genplot = TRUE, outDir = NULL, verboseLM = TRUE)

Arguments

- **known**: subOrigData or SpatialPointsDataFrame. Known-origin tissue isotope data from the subOrigData function or provided by user. User-provided data must be formatted as a subOrigData object (see `subOrigData`) or a SpatialPointsDataFrame in which the first data field contains the measured tissue isotope value and the second the 1 standard deviation uncertainty on that value.
- **isoscape**: raster. Isoscape raster with two layers. The first one is the mean and the second is one standard deviation.
- **mask**: SpatialPolygonsDataFrame. Constrains the area of the output rasters. If this is not provided, the entire area of `isoscape` is returned.
- **interpMethod**: numeric. 1 or 2. Designate one of two methods for extracting values from `isoscape`. If 1, values for the cell in which a point falls are returned. If 2 the returned values are interpolated (bilinear interpolation) from the values of the four adjacent grid cells.
- **NA.value**: NA or numeric. Value representing the absence of data in `isoscape`. Commonly used values include NA, -9999, and 9999.
- **ignore.NA**: logical. If NA values are extracted from `isoscape` at the location of samples in known, ignore these values and proceed with calRaster.
- **genplot**: logical. Plot the results.
- **outDir**: character string. Directory path to which output will be saved. If NULL no files are written.
- **verboseLM**: logical. Print out the linear regression results.
Value

Returns an object of class “rescale”.

isoscape.rescale

RasterBrick. isoscape rescaled using a linear regression model between values of known and isoscape. The first layer is the mean prediction and the second the standard deviation.

lm.data
data.frame. Known origin data and extracted isoscape values used for linear regression modeling.

lm.model
list. Linear regression model.

See Also

pdRaster

Examples

# load North America boundary and global isoscape
data(“naMap”)
data(“d2h_lrNA”)

# load hydrogen isotope data for human hair in North America
d = subOrigData(group = “Modern human”, mask = naMap, niter = 100)

# rescale from environmental isoscape to tissue isoscape
r = calRaster(known = d, isoscape = d2h_lrNA, mask = naMap)

----------------------

d2h_lrNA

data: low resolution North American growing season H isoscape

Description

Interpolated growing season precipitation H isoscape from waterisotopes.org

Usage

data(“d2h_lrNA”)

Format

RasterBrick with two layers. The first layer is the mean prediction and the second is 1 standard deviation

References

Examples

```r
library(raster)
data("d2h_lrNA")
plot(d2h_lrNA)
```

getIsoscapes  

**Description**

This function retrieves gridded isotope maps from waterisotopes.org, unpacks the zip archives, and bundles the map layers as a RasterStack.

**Usage**

```r
getIsoscapes(isoType = "GlobalPrecipGS", timeout = 1200)
```

**Arguments**

- `isoType` character string indicating which isoscapes are requested: see 'Details'.
- `timeout` integer. Maximum allowed file download time, in seconds. Some isoscape archives exceed 2 GB in size and may require long download times on slow connections. This option may not work on all system configurations.

**Details**

Accepted `isoType` values are:

- "GlobalPrecipGS" Global growing-season precipitation H and O isotope values
- "GlobalPrecipMA" Global mean-annual precipitation H and O isotope values
- "GlobalPrecipMO" Global monthly precipitation H and O isotope values
- "GlobalPrecipALL" Global mean-annual and monthly precipitation H and O isotope values
- "USPrecipMA" High-resolution contiguous USA mean-annual precipitation H and O isotope values
- "USPrecipMO" High-resolution contiguous USA monthly precipitation H and O isotope values
- "USPrecipALL" High-resolution contiguous USA mean-annual and monthly precipitation H and O isotope values
- "USSurf" High-resolution contiguous USA surface water H and O isotope values
- "USTap" High-resolution contiguous USA surface water H and O isotope values
- "USSr" High-resolution contiguous USA Sr isotope ratios
- "CaribSr" High-resolution Sr isotope ratios for the circum-Caribbean region
Value

RasterStack containing the requested isoscape layers.

References

https://wateriso.utah.edu/waterisotopes/pages/data_access/ArcGrids.html

Examples

```r
## Not run: iso = getIsoscapes("CaribSr")
```

### isoStack

<table>
<thead>
<tr>
<th>isoStack</th>
<th>Stack isoscapes</th>
</tr>
</thead>
</table>

**Description**

Combine multiple isoscapes into a single data object, including optional reconciliation of raster properties.

**Usage**

```r
isoStack(..., clean = TRUE)
```

**Arguments**

- `...` Two or more RasterStack or RasterBrick isoscapes, each with two layers, or rescale objects as produced by `calRaster`. For objects other than `rescale`, the first layer in each should contain the tissue-specific isoscape and the second the isoscape prediction uncertainty (1 standard deviation).
- `clean` logical. Reconcile differences in raster properties within `...`?

**Details**

If `clean = TRUE` all rasters are projected to the projection of the first object in `...` and then resampled to the highest spatial resolution and smallest common spatial extent within `...`. Finally, cells containing NA in any layer within `...` are masked across all layers.

If `clean = FALSE` any differences in raster properties between isoscapes will produce an error.

**Value**

Returns an object of class “isoStack”, a list containing the isoscapes objects in `...` after any cleaning.
Examples

# load two isoscapes
data("d2h_lrNA")
data("sr_MI")

# stack them
h_s = isoStack(list(d2h_lrNA, sr_MI))

# plot isoStack
plot(h_s)

---

jointP  Joint probability of origin

Description

Joint probability for individuals of common origin (product of probabilities)

Usage

jointP(pdR)

Arguments

pdR  RasterStack or RasterBrick of probability density maps, e.g., as produced by pdRaster. All layers must have common spatial extent and projection.

Value

RasterLayer.

Examples

# load North America boundary and isoscape
data("naMap")
data("d2h_lrNA")

# load hydrogen isotope data for human hair in North America
d = subOrigData(group = "Modern human", mask = naMap, niter = 100)

# rescale from environmental isoscape to tissue isoscape
r = calRaster(known = d, isoscape = d2h_lrNA, mask = naMap)

# four unknown-origin examples
id = c("A", "B", "C", "D")
d2H = c(-110, -90, -105, -102)
un = data.frame(id, d2H)

# posterior probabilities
knownOrig

\[
\text{pp} = \text{pdRaster}(r, \text{unknown} = \text{un}, \text{mask} = \text{naMap})
\]

# joint probability for individuals of common origin
\text{jointP(pp)}

---

**knownOrig**  
*data: database of H and O isotope data for tissues of known origin*

**Description**

This dataset consists of hydrogen and oxygen isotope values and metadata for human hair, insect wings, and bird feathers of known geographic origin.

**Usage**

\text{data("knownOrig")}

**Format**

list.

- **sites**  
  SpatialPointsDataFrame with 5 fields. WGS84 unprojected geometry.

  - [, 1 ] Site_ID: Unique ID
  - [, 2 ] Site_name: Site name or descriptor
  - [, 3 ] State: State or province of collection site, where recorded
  - [, 4 ] Country: Country of collection site, where recorded
  - [, 5 ] Site_comments: Site comments

- **samples**  
  data.frame with 15 fields.

  - [, 1 ] Sample_ID: Unique ID
  - [, 2 ] Sample_ID_orig: ID used in original data report
  - [, 3 ] Site_ID: ID for sample collection site
  - [, 4 ] Dataset_ID: ID for dataset from which sample is derived
  - [, 5 ] Taxon: Genus and species name
  - [, 6 ] Group: Biological group (informal)
  - [, 7 ] Source_quality: Code indicating level of certainty in geographic origin
  - [, 8 ] Age_class: Code for age of individual
  - [, 9 ] Material_type: Tissue sampled, e.g., “Hair”
  - [, 10 ] Matrix: Compound measured, e.g., “Keratin”
  - [, 11 ] d2H: Hydrogen isotope value (permil)
  - [, 12 ] d2H.sd: Reported analytical uncertainty for hydrogen isotope value (permil)
  - [, 13 ] d18O: Oxygen isotope value (permil)
  - [, 14 ] d18O.sd: Reported analytical uncertainty for oxygen isotope value (permil)
  - [, 15 ] Sample_comments: Sample comments

- **sources**  
  data.frame with 17 fields.
[, 1 ] Dataset_ID: Unique ID
[, 2 ] Dataset_name: Short name or descriptor
[, 3 ] Citation: Bibliographic citation for study
[, 4 ] Sampling_method: How material was subsampled for analysis, if reported
[, 5 ] Sample_powdered: Was sample powdered prior to analysis (Y/N/NA)?
[, 6 ] Lipid_extraction: Were lipids chemically extracted prior to analysis (Y/N/NA)?
[, 7 ] Lipid_extraction_method: Solvent used to extract lipids
[, 8 ] Exchange: Was a correction for exchangeable H made (Y/N/NA)?
[, 9 ] Exchange_method: Method used to correct for exchangeable H
[, 10 ] Exchange_T: Was H exchange carried out at ambient or high temperature (Ambient/High/NA)?
[, 11 ] H_cal: Reference scale used to calibrate H isotope data, see stds object hstds
[, 12 ] O_cal: Reference scale used to calibrate O isotope data, see stds object ostds
[, 13 ] Std_powdered: Were calibration standards powdered (Y/N/NA)?
[, 14 ] Drying: Did the study document how samples were fully dried and transferred dry to instrument (Y/N/NA)?
[, 15 ] Analysis_method: Instrument configuration used for analysis
[, 16 ] Analysis_type: What elements were analyzed for stable isotope ratios (H/O/H_O)?
[, 17 ] Source_comments: Data source comments

See Also

subOrigData

Examples

library(sp)
data("knownOrig")
class(knownOrig$sites)
class(knownOrig$samples); class(knownOrig$sources)
summary(knownOrig$samples)
print(knownOrig$sources[, 1:2])
plot,assignR:::wrld_simpl, border = "grey")
plot(knownOrig$sites, add = TRUE)

naMap data: naMap

Description

Simplified North American boundaries polygon

Usage

data("naMap")
oddsRatio

Format

SpatialPolygonsDataFrame

Examples

library(sp)

data("naMap")
plot(naMap)

oddsRatio  Odds ratio of points or regions

Description

Calculate ratio of odds for two locations (points or polygons)

Usage

oddsRatio(pdR, inputP)

Arguments

pdR  RasterStack or RasterBrick of probability density maps, e.g., as produced by pdRaster.

inputP  SpatialPoints* or SpatialPolygons* object of length 2

Examples

library(sp)

# load North America boundary and global isoscape
data("naMap")
data("d2h_lrNA")

# load hydrogen isotope data for human hair in North America
d = subOrigData(group = "Modern human", mask = naMap, niter = 100)

# rescale from environmental isoscape to tissue isoscape
r = calRaster(known = d, isoscape = d2h_lrNA, mask = naMap)

# four unknown-origin examples
id = c("A", "B", "C", "D")
d2H = c(-110, -90, -105, -102)
un = data.frame(id,d2H)

# posterior probabilities
pp = pdRaster(r, unknown = un, mask = naMap)
# SpatialPolygons for two regions of interest

data("states")
s1 = states[states$STATE_ABBR == "UT",]
s2 = states[states$STATE_ABBR == "NM",]
plot(naMap)
plot(s1, border = "red", add=TRUE)
plot(s2, border = "blue", add=TRUE)

# Get odds ratio for two regions using SpatialPolygon method
s12 = rbind(s1, s2)
oddsRatio(pp, s12)

# Create SpatialPoints for two points of interest
pp1 = c(-112,40)
pp2 = c(-105,33)
pp12 = SpatialPoints(coords = rbind(pp1,pp2),
                      proj4string=CRS("+proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0"))
points(pp12, pch = 21, bg = "light blue")

# Get odds ratio for two points using SpatialPoints method
oddsRatio(pp, pp12)

---

**pdRaster**  
*Probability of origin surfaces*

**Description**

Calculate posterior probabilities of origin for a sample based on its isotope ratio.

**Usage**

```r
pdRaster(r, unknown, prior = NULL, mask = NULL, genplot = TRUE, outDir = NULL)
```

**Arguments**

- **r**: RasterStack or RasterBrick with two layers, rescale object (see `calRaster`), or `isoStack` object. For user-generated raster objects, the first layer must be the substrate-specific isoscape (mean prediction) and the second the isoscape prediction uncertainty (1 standard deviation).
- **unknown**: data.frame, `refTrans` object, or list of two or more `refTrans` objects. For user-created data.frame, first column should contain unique sample IDs, and subsequent columns should contain sample isotope values for one or more isotopes to be used in assignment.
- **prior**: raster. Optional raster layer with prior probabilities, which has the same projection, resolution and extent as `r`.
- **mask**: SpatialPolygonsDataFrame. This mask will constrain the assignment area. If this is not provided, a default of mask of the extent of `r` is used.
- **genplot**: logical. Plot results in R.
outDir character string. Directory path to which output will be saved. If NULL no files are written.

Details

If more than one isotope marker is to be used for multivariate assignment, \( r \) must be an `isoStack` object and the number of isoscapes in that object must be equal to the number of isotope-value columns or `refTrans` objects included in `unknown`. Isoscapes and unknown sample values will be matched based on order, so it is critical that the values appear in the same order in these two input objects.

Value

RasterStack or RasterLayer including a probability density surface for each individual in `unknown`. If `outDir` is not NULL, writes individual rasters in GeoTIFF format and a single PDF file with images for each probability density raster to the designated directory.

See Also

calRaster refTrans isoStack

Examples

```r
# load North America boundary and global isoscape
data("naMap")
data("d2h_lrNA")

# load hydrogen isotope data for human hair in North America
d = subOrigData(group = "Modern human", mask = naMap)

# rescale from environmental isoscape to tissue isoscape
r = calRaster(known = d, isoscape = d2h_lrNA, mask = naMap)

# sample to assign
id = "smile"
d2H = -80
un = data.frame(id, d2H)

# posterior probability surface
asn = pdRaster(r, unknown = un, mask = naMap)
```

---

**plot.isoStack**  
*Plot method for stacked isoscapes*

**Description**

Plot the output from `isoStack`.
Usage

```r
## S3 method for class 'isoStack'
plot(x, ...)
```

Arguments

- `x`: An isoStack object
- `...`: Other arguments to be passed to plot

See Also

- `isoStack`

Examples

```r
# load two isoscapes
data("d2h_lrNA")
data("sr_MI")

# stack them
h_s = isoStack(list(d2h_lrNA, sr_MI))

# plot isoStack
plot(h_s)
```

---

**plot.QA**

Plot quality metrics for geographic assignments

### Description

Plot the output from QA, including spatial precision, bias, sensitivity and odds ratio of known locations for validation samples.

### Usage

```r
## S3 method for class 'QA'
plot(x, ..., outDir = NULL)
```

### Arguments

- `x`: One or more QA objects
- `...`: Other arguments to be passed to plot
- `outDir`: character string. Directory path to which output will be saved. If NULL no files are written.
plot.wDist

References


See Also

QA

Examples

```r
# load data
data("naMap") # North America
data("d2h_lrNA") # precipitation hydrogen isoscape of North America
data("knownOrig") # isotopes of known-origin samples

# extract some known-origin data
d1 = subOrigData(taxon = "Buteo lagopus")

# run quality assessment based on precipitation hydrogen isotopes and
# known-origin samples; small values of valiStation and valiTime
# are used in example to reduce run time

# first with one example
# gives warning because a small number of samples are available
qa1 = QA(isoscape = d2h_lrNA, known = d1, valiStation = 1,
          valiTime = 2, by = 10, mask = naMap, name = "Buteo")

# plot the qa result
plot(qa1)

# now compare with a second data set
qa2 = QA(isoscape = d2h_lrNA, known = d2, valiStation = 1,
          valiTime = 2, by = 10, mask = naMap, name = "Charadrius")
plot(qa1, qa2)
```

plot.wDist

*Plot weighted distance and bearing distributions*

Description

Plot the output from `wDist`, including weighted kernel density distributions for distance and bearing of travel.
Usage

```
## S3 method for class 'wDist'
plot(x, ..., bin = 20, pty = "both", index = c(1:5))
```

Arguments

- `x` A wDist object
- `...` Other arguments to be passed to plot
- `bin` numeric. Bin width used to generate rose plot of travel bearings, in degrees. Must be a factor of 360.
- `pty` character. Type of plot to produce. Must be one of “dist”, “bear”, or “both”.
- `index` numeric. Which items in x to plot? Numeric vector of up to 5 integers. Values in excess of 5 or exceeding the length of x will be ignored.

Details

For the default `pty`, two plot panels will be printed to the active graphical device showing the distance and bearing distributions for (up to) the first five samples in `wd`. If more than five items exist in `wd`, those beyond the fifth will be ignored and a message returned.

See Also

- `wDist`

Examples

```r
library(sp)

# load North America boundary and global isoscape
data("naMap")
data("d2h_lrNA")

# load hydrogen isotope data for human hair in North America
d = subOrigData(group = "Modern human", mask = naMap, niter = 100)

# rescale from environmental isoscape to tissue isoscape
r = calRaster(known = d, isoscape = d2h_lrNA, mask = naMap)

# four unknown-origin examples
id = c("A", "B", "C", "D")
d2H = c(-110, -90, -105, -102)
un = data.frame(id, d2H)

# posterior probabilities
pp = pdRaster(r, unknown = un, mask = naMap)

# random collection locations
sites = d$data[sample(seq(length(d$data)), 4),]

# generate a wDist object
```
wd = wDist(pp, sites)

# plot distributions
plot(wd)

# plot bearing distribution for sample B with a finer bin size
plot(wd, bin = 5, pty = "bear", index = 2)

---

**Quality assessment of geographic assignments**

**Description**

How well does a given isoscape and/or known origin data set constrain the geographic origin of samples? Uses iterative re-sampling of known origin data to evaluate sample assignments and reports a suite of quality metrics.

**Usage**

```r
QA(known, isoscape, bySite = TRUE, valiStation = 1, valiTime = 50, recal = TRUE, by = 2, prior = NULL, mask = NULL, setSeed = TRUE, name = NULL)
```

**Arguments**

- `known`: subOrigData, list of subOrigData, or SpatialPointsDataFrame. Known-origin tissue isotope data from the `subOrigData` function or provided by user. User-provided data must be formatted as subOrigData objects (see `subOrigData`) or a SpatialPointsDataFrame (see Details).

- `isoscape`: RasterStack or RasterBrick with two layers or `isoStack` object. For user-generated raster objects, the first layer must be the isoscape (mean prediction) and the second the isoscape prediction uncertainty (1 standard deviation).

- `bySite`: logical. Resample known by site (TRUE) or by sample (FALSE)?

- `valiStation`: numeric. How many sites or samples from known are withheld for validation? Must be two or more smaller than the length of `known`.

- `valiTime`: numeric. How many times do you want to randomly draw validation samples and run the validation? Must be an integer greater than one.

- `recal`: logical. Recalibrate the isoscape(s) using the known-origin data? If FALSE, isoscape should be a calibrated product appropriate to the samples, and a single iteration is run for each sample in known; parameters `bySite`, `valiStation`, and `valiTime` are ignored.

- `by`: integer. Threshold increment to use in evaluating assignment performance. Must be between 1 and 25.

- `prior`: raster. Optional raster layer with prior probabilities, which has the same projection, resolution and extent as `isoscape`. 


mask SpatialPolygonsDataFrame. Constrains the area of the output rasters. If this is not provided, the entire area of isoscape is returned.

setSeed logical. Do you want to set.seed() when you randomly draw validation stations? “TRUE” gives the same sequence of random draws each time the function is called.

name character. Useful for identifying the QA output in subsequent plotting.

Details

If known is a user-provided SpatialPointsDataFrame, the first field in @data must include the measured value for the first (or only) isotope marker and the second the one standard deviation uncertainty on that value. Subsequent fields must include the same information for all other isotope markers included in the analysis, and these markers must appear in the same order as in isoscape. A user-provided SpatialPointsDataFrame must include a field named “Site_ID” containing unique values for each sampling site to support the “bySite” option, otherwise use bySite = FALSE.

Value

Returns an object of class “QA”.

val_stations numeric. An X*Y data.frame of validation station IDs for all valiTime. X = valiTime and Y = valiStation.

pd_val numeric. An X*Y data.frame containing the posterior probability density for the validation stations. X = valiTime and Y = valiStation.

prption_byArea numeric. An X*Y data.frame showing the proportion of validation individuals for which the known origin is contained within the top 0.00 to 1.00 area quantile (with increment of by / 100; Y = ceiling(100 / by) + 1). X = valiTime.

prption_byProb numeric. An X*Y data.frame showing the proportion of validation individuals for which the known origin is contained within the top 0.00 to 1.00 probability quantile (with increment of by / 100; Y = ceiling(100 / by) + 1). X = valiTime.

precision list. The length of the list is valiTime. Each element is an X*Y matrix showing the proportional area of the total assignment surface covered by the assignment region at a given probability quantile from 0.00 to 1.00 (with increment of by / 100; X = ceiling(100 / by) + 1) for each validation sample (Y = valiStation).

random_prob_density

Random probability of assignment to any given grid cell on the assignment surface(i.e. 1 divided by the total number of grid cells).

name character. Name assigned to the QA object.

by integer. Value of by used.

Note

See Ma et al. (2020) for methodological details.
References


See Also

plot.QA

Examples

```r
# load data
data("naMap") # North America
data("d2h_lrNA") # precipitation hydrogen isoscape for North America
data("knownOrig") # hydrogen isotopes of known-origin samples

# extract some known-origin data
d1 = subOrigData(taxon = "Buteo lagopus")

# run quality assessment based on precipitation hydrogen isotopes and # known-origin samples; small values of valiStation and valiTime # are used in example to reduce run time

# first with one example
# gives warning because a small number of samples are available
qa1 = QA(known = d1, isoscape = d2h_lrNA, valiTime = 2, by = 10, mask = naMap, name = "Buteo")

# plot the qa result
plot(qa1)

# now compare with a second data set
# second with a second data set

d2 = subOrigData(taxon = "Charadrius montanus")
qa2 = QA(known = d2, isoscape = d2h_lrNA, valiTime = 2, by = 10, mask = naMap, name = "Charadrius")
plot(qa1, qa2)
```

**qttRater**  
*Sample assignment using thresholds*

**Description**

Selects the grid cells of probability density rasters with the highest probability and returns rasters with these cell values set to 1. Cells are selected based on the user-specified quantile threshold so that the most-probable cells representing a given fraction of the assignment area or posterior probability are returned.
**qtlRaster**

**Usage**

```r
qtlRaster(pdR, threshold, thresholdType = "area", genplot = TRUE, outDir = NULL)
```

**Arguments**

- `pdR` RasterLayer, RasterStack, or RasterBrick. Probability density maps for individual samples, e.g., as output by `pdRaster`.
- `threshold` numeric from 0 to 1. Quantile to be selected.
- `thresholdType` character. Either "area" (default) or "prob". If "area", the most probable cells constituting `threshold` percent of the assignment area are selected. If "prob", the most probable cells constituting `threshold` percent of the posterior probability are selected.
- `genplot` logical. Plot results in R.
- `outDir` character string. Directory path to which output will be saved. If NULL no files are written.

**Value**

RasterStack or RasterLayer including a binary assignment surface for each individual in `pdR`. If `outDir` is not NULL, writes individual rasters in GeoTIFF format and a single PDF file with images for each raster to the designated directory.

**Examples**

```r
# load North America boundary and global isoscape
data("naMap")
data("d2h_lrNA")

# load hydrogen isotope data for human hair in North America
d = subOrigData(group = "Modern human", mask = naMap, niter = 100)

# rescale from environmental isoscape to tissue isoscape
r = calRaster(known = d, isoscape = d2h_lrNA, mask = naMap)

# four unknown-origin examples
id = c("A", "B", "C", "D")
d2H = c(-110, -90, -105, -102)
un = data.frame(id, d2H)

# posterior probability surface
asn = pdRaster(r, unknown=un, mask=naMap)

# assign to most probable 10 percent of area
## Not run: qtlRaster(asn, threshold = 0.1)

# assign to most probable 10 percent of probability distribution
qtlRaster(asn, threshold = 0.1, thresholdType = "prob")
```
refTrans

Transform reference scale of data

Description

This function conducts transformations to convert isotope measurements between reference scales.

Usage

refTrans(samples, marker = "d2H", ref_scale = "VSMOW_H", niter = 5000)

Arguments

samples data.frame. Must include a field with data to be transformed, analytical reproducibility of sample data (1 standard deviation), and original reference scale for calibration of data. These fields must be named marker, marker.sd, and marker.cal, respectively, where marker is “d2H” or “d18O”. Values for the cal field should correspond to Calibration codes found in stds tables hstds and ostds.

marker character string. Column name for isotopic data to be extracted, either “d2H” or “d18O”.

ref_scale character string. Text identifier for reference scale to which all isotope values will be transformed. See stds.

niter integer. Number of random samples used to propagate uncertainty in calibration hierarchy transformations.

Value

Returns an object of class “refTrans”.

data data.frame. Formatted identically to input object samples, with values for the data and data uncertainty fields replaced with transformed values.

chains list. Each item is a character string vector containing the hierarchy of calibrations used in the transformation for a set of selected samples. See stds.

References


Examples

# Some fake sample data
s = data.frame("d2H" = seq(-100, -10, by=10), "d2H.sd" = rep(2), "d2H_cal" = rep("OldUT_H_1"))

# Transform to VSMOW-SLAP scale using default arguments
d1 = refTrans(s)
# Transformed values
d1$data$d2H

# error - target scale not valid for marker
## Not run: d2 = refTrans(s, ref_scale = "VSMOW_0")

---

**sr_MI**

*data: low resolution locally weathered Sr isoscape for Michigan*

---

**Description**

Modeled 87Sr/86Sr value of the local rock weathering flux, obtained from waterisotopes.org and aggregated to 10 km resolution.

**Usage**

```r
data("sr_MI")
```

**Format**

RasterBrick with two layers. The first layer is the mean prediction and the second is 1 standard deviation (here estimated as 1 percent of the modeled mean)

**References**


**Examples**

```r
library(raster)
data("sr_MI")
plot(sr_MI)
```

---

**states**

*data: states*

---

**Description**

Spatial Polygon Data Frame of lower 48 U.S. states

**Usage**

```r
data("states")
```
**stds**

**Format**

SpatialPolygonsDataFrame

**Examples**

```r
library(sp)

data("states")
plot(states)
```

---

**stds**

*data: documentation of keratin H and O isotope standard calibrations*

**Description**

This data object contains information on keratin H and O isotope standard materials and calibrations used across multiple laboratories since the year 2000.

**Usage**

```r
data("stds")
```

**Format**

list.

**hstds** data.frame with 18 fields.

- **[, 1 ] Calibration**: Calibration code
- **[, 2 ] High_ID**: Identifier for high-value standard
- **[, 3 ] High_material**: Description of high-value standard material
- **[, 4 ] High**: Mean hydrogen isotope value of high-value standard
- **[, 5 ] High_sd**: Standard deviation of calibration data for high-value standard
- **[, 6 ] High_n**: Number of calibration data for high-value standard
- **[, 7 ] High_se**: Standard error of the calibrated mean for high-value standard
- **[, 8 ] Low_ID**: Identifier for low-value standard
- **[, 9 ] Low_material**: Description of low-value standard material
- **[, 10 ] Low**: Mean hydrogen isotope value of low-value standard
- **[, 11 ] Low_sd**: Standard deviation of calibration data for low-value standard
- **[, 12 ] Low_n**: Number of calibration data for low-value standard
- **[, 13 ] Low_se**: Standard error of the calibrated mean for low-value standard
- **[, 14 ] Ref_scale**: Calibration scale against which the values for this calibration are anchored
- **[, 15 ] Citation_val**: Source for the calibrated values
- **[, 16 ] Citation_cal**: Source for the methodology used for this calibration
- **[, 17 ] Treatment**: Description of calibration procedure
- **[, 18 ] H_calibration_comments**: Comments
ostds data.frame with 18 fields.

[, 1 ] Calibration: Calibration code
[, 2 ] High_ID: Identifier for high-value standard
[, 3 ] High_material: Description of high-value standard material
[, 4 ] High: Mean oxygen isotope value of high-value standard
[, 5 ] High_sd: Standard deviation of calibration data for high-value standard
[, 6 ] High_n: Number of calibration data for high-value standard
[, 7 ] High_se: Standard error of the calibrated mean for high-value standard
[, 8 ] Low_ID: Identifier for low-value standard
[, 9 ] Low_material: Description of low-value standard material
[, 10 ] Low: Mean oxygen isotope value of low-value standard
[, 11 ] Low_sd: Standard deviation of calibration data for low-value standard
[, 12 ] Low_n: Number of calibration data for low-value standard
[, 13 ] Low_se: Standard error of the calibrated mean for low-value standard
[, 14 ] Ref_scale: Calibration scale against which the values for this calibration are anchored
[, 15 ] Citation_val: Source for the calibrated values
[, 16 ] Citation_cal: Source for the methodology used for this calibration
[, 17 ] Treatment: Description of calibration procedure
[, 18 ] O_calibration_comments: Comments

ham matrix. n x n symmetric, where n is the number of calibrations represented here and in stds$hstds.

oam matrix. n x n symmetric, where n is the number of calibrations represented here and in stds$ostds.

References


Examples

library(raster)
data("stds")

print(stds$hstds[, 1:5])
print(stds$ostds[, 1:5])
image(stds$ham)
image(stds$oam)
subOrigData

Extract known origin sample data

Description

This function subsets the known-origin isotope dataset included in this package and conducts optional transformations to convert isotope measurements to a common reference scale.

Usage

```
subOrigData(marker = "d2H", taxon = NULL, group = NULL, dataset = NULL,
            age_code = NULL, mask = NULL, ref_scale = "VSMOW_H", niter = 5000)
```

Arguments

- **marker**: character string. Column name for isotopic data to be extracted, either “d2H” or “d18O”.
- **taxon**: character string or string vector. Species name(s) for data to be extracted.
- **group**: character string or string vector. Taxonomic groups for data to be extracted.
- **dataset**: integer or integer vector. Dataset ID(s) for data to be extracted. See `knownOrig` feature `sources`.
- **age_code**: character string or string vector. Animal age code for data to be extracted.
- **mask**: SpatialPolygons or SpatialPolygonsDataFrame. Constrains the geographic area from which data are extracted. If not provided, global.
- **ref_scale**: character string. Text identifier for reference scale to which all isotope values will be transformed. See `stds`. Use “NULL” to obtain untransformed values on the originally reported scale.
- **niter**: integer. Number of random samples used to propagate uncertainty in calibration hierarchy transformations.

Value

Returns an object of class “subOrigData”, formatted for use in `calRaster` or `QA` functions.

- **data**: SpatialPointsDataFrame including one feature for each selected sample. data.frame fields are described in `knownOrig` feature `samples`.
- **sources**: data.frame. Information for all data sources for the selected samples. Fields are described in `knownOrig` feature `sources`.
- **chains**: list. Each item is a character string vector containing the hierarchy of calibrations used in the transformation for a set of selected samples. See `stds`.
- **marker**: character string. The isotopic marker specified in the call to `subOrigData`.

References

Examples

```r
## WITHOUT mask
# extract d2H data for Jackdaw, Partridge and Willow Grouse, transformed
# to the VSMOW/SLAP H reference scale by default
d1 = subOrigData(taxon = c("Danaus plexippus", "Setophaga ruticilla",
                      "Turdus migratorius"), niter = 100)
summary(d1)

# extract d2H data for insects and passerine birds without transformation
d2 = subOrigData(group = c("Insect","Passerine"), ref_scale = NULL)
summary(d2)

# extract d18O data for all humans, transformed to the VSMOW/SLAP O reference scale
d3 = subOrigData(marker = "d18O",
                      group = c("Modern human", "Indigenous human"), ref_scale = "VSMOW_O",
                      niter = 100)
summary(d3)

# extract d2H data for humans using taxon, transformed to the VSMOW/SLAP H reference scale
d4 = subOrigData(marker = "d2H", taxon = "Homo sapiens", ref_scale = "VSMOW_H",
                      niter = 100)
summary(d4)

## WITH mask
# error - no samples found
## Not run: d5 = subOrigData(taxon = "Turdus philomelos", mask = naMap)
# this works OK
d6 = subOrigData(taxon = c("Danaus plexippus", "Setophaga ruticilla",
                       "Turdus migratorius"), mask = naMap)
```

unionP

### Union probability of origin

**Description**

Probabilities that at least one individual came from each location in the assignment area (union of probabilities)

**Usage**

```r
unionP(pdR)
```

**Arguments**

- `pdR` RasterStack or RasterBrick of probability density maps, e.g., as produced by `pdRaster`.

**Value**

RasterLayer.
Examples

```r
# load North America boundary and isoscape
data("naMap")
data("d2h_lrNA")

# load hydrogen isotope data for human hair in North America
d = subOrigData(group = "Modern human", mask = naMap, niter = 100)

# rescale from environmental isoscape to tissue isoscape
r = calRaster(known = d, isoscape = d2h_lrNA, mask = naMap)

# four unknown-origin examples
id = c("A", "B", "C", "D")
d2H = c(-110, -90, -105, -102)
un = data.frame(id, d2H)

# posterior probabilities
pp = pdRaster(r, unknown = un, mask = naMap)

# probability that one or more individuals are from a given location
unionP(pp)
```

---

**wDist**

*Probability weighted distances and bearings*

**Description**

Calculate the distance and bearing of migration for one or more samples, weighted by probabilities from a `pdRaster` analysis.

**Usage**

```r
wDist(pdR, sites, maxpts = 1e5)
```

**Arguments**

- `pdR`  
  RasterStack or RasterBrick of n probability density maps, e.g., as produced by `pdRaster`.
- `sites`  
  SpatialPoints* object containing the collection locations for the n samples represented in `pdR`.
- `maxpts`  
  numeric. Maximum number of grid cells at which to calculate bearing and distance.
Details

`pdR` and `sites` must be of equal length and corresponding order. Names in the output object are taken from the names of the layers in `pdR`.

Distances and bearings are calculated on the WGS84 geoid using functions from the `geosphere` package. These calculations can take a long time for large rasters. If `maxpts` is less than the number of grid cells in each `pdR` layer, calculations are carried out for `maxpts` randomly selected cells.

Bearing values correspond to the initial bearing from source to collection location, and are reported on a scale of -180 to +180 degrees. The statistical metrics are rectified so that values for distributions spanning due south are reported correctly. Both weighted bearing and distance distributions are often multimodal, and it is recommended to review the distribution densities to assess the representativeness of the statistics (e.g., using `plot.wDist`).

Value

Returns an object of class “wDist”, a list of length n. Each item contains three named objects:

- `stats`: named number. Statistics for the distance (dist, meters) and bearing (bear, degrees) between source and collection locations, including the weighted mean (wMean) and quantile (wXX) values.
- `d.dens`: density. Weighted kernel density for the distance between source and collection locations (meters). See `density`.
- `b.dens`: density. Weighted kernel density for the bearing between source and collection locations (degrees). See `density`.

Examples

```r
library(sp)

# load North America boundary and global isoscape
data("naMap")
data("d2h_lrNA")

# load hydrogen isotope data for human hair in North America
d = subOrigData(group = "Modern human", mask = naMap, niter = 100)

# rescale from environmental isoscape to tissue isoscape
r = calRaster(known = d, isoscape = d2h_lrNA, mask = naMap)

# four unknown-origin examples
id = c("A", "B", "C", "D")
d2H = c(-110, -90, -105, -102)
un = data.frame(id,d2H)

# posterior probabilities
pp = pdRaster(r, unknown = un, mask = naMap)

# random collection locations
sites = d$data[sample(seq(length(d$data)), 4),]
```
# generate a wDist object
wd = wDist(pp, sites)

# structure of the wDist object
str(wd, 2)
Index

* datasets
  knownOrig, 9
  stds, 23
  assignR, 2
  c.wDist, 3
  calRaster, 4, 7, 12, 13, 25
  d2h_lrNA, 5
  density, 28
  getIsoscapes, 6
  isoStack, 7, 12–14, 17
  jointP, 8
  knownOrig, 9, 25
  naMap, 10
  oddsRatio, 11
  pdRaster, 5, 12
  plot.isoStack, 13
  plot.QA, 14, 19
  plot.wDist, 15, 28
  QA, 14, 15, 17, 25
  qtlRaster, 19
  refTrans, 12, 13, 21
  sr_MI, 22
  states, 22
  stds, 10, 21, 23, 25
  subOrigData, 4, 10, 17, 25
  unionP, 26
  wDist, 3, 15, 16, 27