Documentation on paleoLibrary functions

latest update by: Michael Schröder, AWI

May 26, 2011

Use of this file

This file provides elementary information on the tools that are available in the paleoLibrary. For each function, its use, input and output values are briefly described. Also, hyperlinks are given in order to show interconnections among the approximately 220 functions (12/05/11). [Note that the latter are not ordered alphabetically, but categorized according to the task they perform.] There is an alphabetically ordered index at the end of this file, though.

Notation

The following graphical features are used to keep the information boxes clearly arranged:

- ▷ input and output types appear colored:
  - · Green: variable is a pField object
  - · Blue: variable is a pTs object
  - · Orange: variable is a character string
  - · Cyan: variable is a vector
  - · Purple: variable is boolean (TRUE or FALSE)
  - · Magenta: variable is a scalar
  - · LimeGreen: variable is a matrix or an array
  - · Yellow: variable is a function
  - · Black: variable is a list
  - · Brown: output is a plot
  - · Gray: . . . (other formats; e.g., ts object (see R help for ts))
- ▷ variable names are highlighted, e.g., `variablename`
- ▷ optional input is marked by the ⋆ symbol

Still to be done

Each function needs to be reviewed by corresponding code authors. Apart from this, there are still many gaps in the descriptions (see the comment lines). Also, it would be nice to employ external links more systematically to provide the user with detailed information on a subject of interest. Last but not least, an EXAMPLE item is yet to be added to each information box.

paleoLibrary functions

"[.pField" defines the output of a pField object pfield when the user requests pfield[p1], pfield[p1,], pfield[,p2] or pfield[p1,p2]

USES: pField, c1t2
IS USED BY: pField

INPUT:

- pField object (x)
- scalar / vector (p1) and/or scalar / vector (p2)

OUTPUT: depends on the choice of p1 and p2:

pfield[p1]	=	scalar / vector ($p1^{th}$ value(s) of pfield)
pfield[p1,]	=	pField object (pfield at times p1)
pfield[,p2]	=	pTs object (time series at $p1^{th}$ gridpoint(s))
pfield[p1,p2]	=	scalar / vector (pfield at time(s) p1 and gridpoint(s) p2)

COMMENT: correct?

"[.pTs" defines the output of a pTs object pts when the user requests pts[p1], pts[p1,], pts[,p2] or pts[p1,p2]

USES: pTs, c1t2
IS USED BY: pTs

INPUT:

- pTs object (x)
- scalar / vector (p1) and/or scalar / vector (p2)

OUTPUT: depends on the choice of p1 and p2:

pts[p1]	=	pTs object (pts at times p1)
pts[p1,]	=	? pTs object (pts at times p1) if pts multidimensional?
pts[,p2]	=	? pTs object ($p2^{th}$ time series of pts) if pts multidimensional?
pts[p1,p2]	=	? pTs object ($p2^{th}$ time series of pts at time(s) p1?)

COMMENT: output format unclear

addhistory adds a history (e.g., a comment) to a pTs or a pField object

USES: -
IS USED BY: e.g., scale.pTs, Ops.pField, detrend.pField, , prcompO.pTs

INPUT:

- pTs / pField object (`x`)
- vector (`newhist`; e.g., a character string)

OUTPUT:

- pTs / pField object (with updated history attribute)

COMMENT: correct?

addland adds land contours to a plot (requires the R package clim.pact)

USES: -
IS USED BY: plotmap.pField, plotmap.pFieldb, plotmapc.pField addland, ploteof, plotwind, plotcont.pField, myfun1, myfun2

INPUT: -
⋆ OPTIONAL INPUT:

- character string (`col="grey50"`; color of the contour lines)
- scalar (`lwd=1`; thickness of the contour lines)

OUTPUT:

- inserted into the current plot

COMMENT: correct?

addpoints adds points to a field plot. The colors of the points are chosen according to the color scale applied to the 2D field

USES: -
IS USED BY: -

INPUT:

- scalar / vector (`lat`; latitude(s))
- scalar / vector (`lon`; longitude(s))
- scalar / vector (`value`; values belonging to the longitude/latitude pair(s))

⋆ OPTIONAL INPUT:

- scalar (`pch=19`; type of points, see R help for points)
- scalar (`lwd=7`; thickness of the point contours)
- 2D pField object (`data=NULL`; for adjusting the color scale to the input values)
- vector (`zlim=NULL`; range of data values to be plotted; if not specified (`zlim`=NULL, the default) , `zlim` is defined by the observed data set range)
- scalar (`nlevels=20`; number of levels taken for the partition in levels)
- vector (`levels=pretty(zlim,nlevels)`; a set of levels which are used to partition the range of the data; the default has nlevels steps on the `zlim` range)
- function (`palette=distinct.col`; color palette used for the plotting in filled.contour.own)

OUTPUT:

- inserted into the current plot

COMMENT: correct?

addwind adds velocities to a 2D field (for convenience, multiple times as in addland)

USES: plot.preparation, quiver
IS USED BY: -

INPUT:

- pField object (`dataU`)
- pField object (`dataV`)

⋆ OPTIONAL INPUT:

- boolean (`shift=F`; see plot.preparation)

OUTPUT:

- inserted into the current plot

COMMENT: correct?

annual_insolation calculates the mean annual insolation for certain years at a certain latitude

USES: daily_insolation
IS USED BY: -

INPUT:

- scalar / vector (`kyear`; kyr BP of interest)
- scalar (`lat`; latitude)

OUTPUT:

- scalar / vector (insolation in W/m^2)

COMMENT: correct?

applyData applies a function to a pTs/pField object

USES: -
IS USED BY: rollmean.pTs

INPUT:

- pTs / pField object (`x`)
- function (`fun`; e.g., rollmean)

OUTPUT:

- pTs / pField object

COMMENT: correct?

applyspace applies a function to a pField object by fixing the time steps. That is, the function is applied to 2D fields, where each 2D field belongs to a certain time step)

USES: -
IS USED BY: getname

INPUT:

- pField object (`data`)
- function (`FUN`)

OUTPUT:

- pTs object (function output=scalar for each time step)

COMMENT: correct?

applytime applies a function to a pField object by fixing the grid points. That is, the function is applied to time series, where each time series belongs to a certain grid point)

USES: getlon, getlat, getname
IS USED BY: makefilm

INPUT:

- pField object (`data`)
- function (`FUN`)

⋆ OPTIONAL INPUT:

- scalar (`newtime=NULL`; defines the time step assigned to the 2D pField output; the default extracts the mean time from the input field)

OUTPUT:

- pField object (function output=scalar for each grid point)

COMMENT: correct?

ar1fit ?

USES: -
IS USED BY: specConf

INPUT: ?

OUTPUT: ?

COMMENT: description incomplete

areamean computes a weighted mean for each time step of a pField object

USES: pTs
IS USED BY: millenium.anomalies, millenium.temperatures, regional.ts

INPUT:

- pField object (`data`)

OUTPUT:

- pTs object (weighted average for each time step)

COMMENT: correct? weighting unclear

axes.type labels the axes of a 2D field plot

USES: -
IS USED BY: plotmap.pField

INPUT:

- vector (`lon`; longitudes of the 2D pField)
- vector (`lat`; latitudes of the 2D pField)

⋆ OPTIONAL INPUT:

- character string (`lonlatlabel="360"`; type of longitude labeling; "180" and "180EW" are also possible)
- scalar (`number.labels=5`; number of labels per longitude/latitude axis)

OUTPUT:

- inserted into the current plot

COMMENT: -

bandpass generates a bandpass filter that has both a low-pass and a high-pass component

USES: lowpass
IS USED BY: indirectly: filter.pField, filter.pTs, filter.pTs1, critval.cor

INPUT:

- scalar (`omega.lower`; lower cutoff-frequency)
- scalar (`omega.upper`; upper cutoff-frequency)

⋆ OPTIONAL INPUT:

- scalar (`n=9`; length of the filter?)
- scalar (`sample=1`; ?)
- boolean (`convergence=T`; ?)

OUTPUT:

- vector (filter?)

COMMENT: correct? input and output incomplete

blocksample is used in the rampfit function to generate bootstrap surrogates (permutation of blocks of residuals)

USES: -
IS USED BY: rampfit

INPUT:

- vector (`data`; rampfit residuals)

⋆ OPTIONAL INPUT:

- scalar (`blocklength=10`; length of the blocks to be permuted)

OUTPUT:

- vector (permuted residuals)

COMMENT: correct?

c1t2 helps transforming a 1D object into a 2D object (e.g., assigning longitudes and latitudes to a data vector) (inverts the c2t1 function)

USES: -
IS USED BY: maxpoint, minpoint, "[.pField", "[.pTs"

INPUT:

- vector (`x`; data)
- scalar (`nLon`; number of longitudes)

OUTPUT:

- list: vector (latitude indices) , vector (longitude indices)

COMMENT: correct?

c2t1 inverts the c1t2 function

USES: -
IS USED BY: selspace

INPUT:

- vector (`lat`; latitude indices)
- vector (`lon`; longitude indices)
- scalar (`nLon`; number of longitudes)

OUTPUT:

- vector (data indices)

COMMENT: correct?

cbind.pTs merges pTs objects (removed as problems arise with newer R versions, probably because the R function cbind.ts does not exist anymore)

USES: -
IS USED BY: -

INPUT: -

OUTPUT: -

COMMENT: function update possible?

ccf.pTs exists in two forms: one calculates the cross-correlation/cross-covariance between two pTs objects, the other the cross correlation/cross covariance between a pTs and a pField object (? see myccf) (both seem to be "experimental unfinished functions")

USES: pField, (myccf, getlon, getlat)
IS USED BY: -

INPUT:

- pTs object (`pTs1`)
- pTs object /pField object (`pTs2`)

⋆ OPTIONAL INPUT:

- boolean (`debug=F`; debugging variable)

OUTPUT:

- vector (ccf output)
- (plot)

COMMENT: description incomplete; versions are not consistent

change.cor.test tests if the changes of correlation values are significant over time (see Sterl et al. (2007)) using the R regression tool lm; the observed difference between the maximum and minimum correlation is compared with the distribution of correlation differences obtained with surrogate time-series

USES: roll.2, cor.pTs, fisher, filter.pTs1
IS USED BY: -

INPUT:

- pTs object (`ts1`)
- pTs object (`ts2`)

⋆ OPTIONAL INPUT:

- scalar (`width=30`; width of the rolling time window the cor.pTs function is applied to)
- scalar (`N.R=1000`; number of surrogate time series)
- scalar (`p=0.05`; p-value)
- boolean (`detrend=FALSE`; determines whether the data are to be detrended)
- boolean (`bootstrap=FALSE`; defines the way the surrogates are generated; if `bootstrap`=TRUE, the data (=residuals originating from the linear regression/correlation) are used, else realizations are drawn from a normal distribution)
- boolean (`fisher=FALSE`; ?)
- vector (`filt=NULL`; filter applied to the data (=residuals originating from the linear regression/correlation); `filt`=NULL means that no filtering is carried through)

OUTPUT:

- list: ?

COMMENT: description incomplete; see input (e.g., fisher variable) and output

composite ? takes a time series and a 3D field and builds composite maps (maximum and minimum)

USES: pField
IS USED BY: -

INPUT:

- pTs object (`ts`)

- pField object (`field`)

⋆ OPTIONAL INPUT:

- ...

OUTPUT:

- list: pField object , pField object

COMMENT: description incomplete

composite.pTs ? takes a time series and a 3D field and builds composite maps (maximum and minimum)

USES: pField
IS USED BY: -

INPUT:

- pTs object (`ts`)
- pField object (`field`)

⋆ OPTIONAL INPUT:

- ...

OUTPUT:

- list: multiple pField objects

COMMENT: description and use unclear

copyattr passes on attributes from one object to another

USES: -
IS USED BY: cortest.pTs, cor.pTs, cor.pTsM, lmSlope.pTs,

INPUT:

- object (`data`; e.g., pTs/pField object ("new" data))
- object (`source`; e.g., pTs/pField object ("old" data))

⋆ OPTIONAL INPUT:

- character string (`newhistory=''`; new history)
- boolean (`cclass=T`; indicates whether the class attribute shall also be passed on)

OUTPUT:

- pTs / pField object (updated "new" data)

COMMENT: correct?

cor.pTs computes the correlation between a pTs object and a pField object/pTs object

USES: copyattr
IS USED BY: -

INPUT:

- pTs object (`pTs`)
- pTs / pField object (`field`)

⋆ OPTIONAL INPUT:

- character string (`use="complete.obs"`; type of use of the observations, see R help for cor)
- scalar (`min.obs=30`; minimum number of observations used for correlation)
- boolean (`debug=F`; debugging variable)

OUTPUT:

- scalar (correlation between 2x pTs)) / vector (correlation between pTs and pField)

COMMENT: correct?

cor.pTsM computes the correlation between a pTs object and a pField object/pTs object, discarding time steps for which the pField object is not available (NA)

USES: copyattr
IS USED BY: -

INPUT:

- pTs object (`pTsS`)
- pTs / pField object (`pFieldS`)

⋆ OPTIONAL INPUT:

- boolean (`debug=F`; debugging variable)

OUTPUT:

- scalar (correlation between 2x pTs)) / pField object (correlation between pTs and pField)

COMMENT: correct?

cor.sb determines overlapping time steps of two pTs objects and carries through a correlation test (using the R tool cor.test)

USES: sb
IS USED BY: -

INPUT:

- pTs object (`a`)
- pTs object (`b`)

OUTPUT:

- cor.test output (e.g., correlation of the sample, p-value)

COMMENT: correct?

cor.sig computes the correlation between two vectors (e.g., pTs objects) and returns NA value if sample correlation is not significant (uses the R tool cor.test)

USES: -
IS USED BY: -

INPUT:

- vector (`ts1`; e.g., pTs object)
- vector (`ts2`; e.g., pTs object)

⋆ OPTIONAL INPUT:

- scalar (`pval=0.05`; p-value)

OUTPUT:

- scalar (correlation estimate if the correlation is significant, else NA)

COMMENT: correct?

corcontour adds correlation contour lines to a 2D field plot (using the R tool contour), so that positive correlations are visualized by solid and negative correlations by dashed lines

USES: -
IS USED BY: -

INPUT:

- vector (`lon`; longitudes)
- vector (`lat`; latitudes)
- matrix (`data`; 2D correlation field)

OUTPUT:

- inserted into the current plot

COMMENT: correct?

cortest.pTs carries through a correlation test between a pTs and a pField object

USES: copyattr
IS USED BY: pcor, sigcor

INPUT:

- pTs object (`pTs`)
- pField object (`field`)

⋆ OPTIONAL INPUT:

- scalar (`min.obs=30`; minimum number of observations used for correlation)

OUTPUT:

- two pField objects saved in one variable (observed correlations and corresponding p-values), can be extracted row-wise

COMMENT: correct?

cost ?

USES: specred, rmse
IS USED BY: ar1fit

INPUT: ?

OUTPUT:

- scalar (RMSE)

COMMENT: description incomplete

critval.cor computes Monte Carlo significance boundaries for the correlation (considering the null hypothesis cor=0) using iid realizations from a normal distribution

USES: -
IS USED BY: -

INPUT:

- scalar (`N`; samplesize (e.g., number of time steps))

- vector (`filt`; filter to be used)

⋆ OPTIONAL INPUT:

- scalar (`pval=0.05`; p-value)
- character string (`one_sided=F`; type of hypothesis test: either `one_sided`=F (the default=one-sided correlation test) or `one_sided`=T (two-sided correlation test))
- scalar (`N.R=5000`; number of random realizations; a high number of realizations leads to more exact significance boundaries)

OUTPUT:

- scalar (if one_sided=T) / vector (if one_sided=F) (critical values)

COMMENT: correct?

cwind adds wind (?) contour lines to a 2D field plot (using the R tool contour)

USES: -
IS USED BY: -

INPUT:

- vector (`lon`; longitudes)
- vector (`lat`; latitudes)
- matrix (`data`; 2D wind (?) field)

OUTPUT:

- inserted into the current plot

COMMENT: correct? what does wind stand for?

daily_insolation computes daily average insolation as a function of day and latitude for any year of the past five million years

USES: orbital_parameters, orbital_parameters_fast

IS USED BY: ins.march21, ins.dec21, annual_insolation

INPUT:

- scalar (`kyear`; kyr BP)
- scalar (`lat`; latitude)
- scalar / vector (`day`; day/day(s) of the year)

⋆ OPTIONAL INPUT:

- scalar (`day_type=1`; day format: `day_type`=1: day input is calendar day (1-365.24), where day 1 is January first; `day_type`=2: day input is solar longitude (0-360 degrees))
- boolean (`fast=T`; fast=T (orbital_parameters_fast is called) or =F (orbital_parameters is called))

OUTPUT:

- list: scalar (daily average solar radiation in W/m^2) , 3x scalar (orbital parameters?)

COMMENT: correct? output description incomplete; also: What is the difference between daily_insolation and daily_insolation_param?

daily_insolation_param computes daily average insolation as a function of day and latitude for any year of the past five million years

USES: -
IS USED BY: ins.dec21.param

INPUT:

- ? scalar (??? kyr BP)
- scalar (`lat`; latitude)
- scalar / vector (`day`; day(s) of the year)
- scalar (`ecc`; eccentricity)
- scalar (`obliquity`; obliquity)
- scalar (`long_perh`; precession?)

⋆ OPTIONAL INPUT:

- scalar (`day_type=1`; day format: `day_type`=1 (default): day input is calendar day (1-365.24), where day 1 is January first; `day_type`=2: day input is solar longitude (0-360 degrees))

OUTPUT:

- list: scalar (daily average solar radiation in W/m^2) , 3x scalar (orbital parameters?)

COMMENT: correct? input and output description incomplete; also: What is the difference between daily_insolation and daily_insolation?

deming carries through a total linear regression using the R optimize tool

USES: -
IS USED BY: -

INPUT:

- vector (`x`; data vector no.1)
- vector (`y`; data vector no.2)
- vector (`xstd`; standard deviations of data vector no.1?)
- vector (`ystd`; standard deviations of data vector no.2?)

⋆ OPTIONAL INPUT:

- boolean (`jackknife=T`; ?)
- boolean (`dfbeta=F`; ?)
- boolean (`scale=T`; ?)

OUTPUT:

- list: ? coefficients etc

COMMENT: description incomplete

detrend ? is used to remove trends observed in the data

USES: ? is detrend just forwarding to detrend.pTs/detrend.pField?
IS USED BY: pcor, roll.1, roll.2, index.nao, index.ao

INPUT: ?

OUTPUT: ?

COMMENT: description incomplete

detrend.default ? see detrend

USES: ?
IS USED BY: -

INPUT: ?

OUTPUT: ?

COMMENT: description incomplete

detrend.pField removes the linear trend a pField object (using the linear regression tool lm implemented in R) and gives residual time series for each gridpoint

USES: addhistory
IS USED BY: -

INPUT:

- pField object (`x`)

OUTPUT:

- pField object

COMMENT: correct?

detrend.pTs removes the linear trend a pTs object (using the linear regression tool lm implemented in R) and outputs the residual time series

USES: addhistory
IS USED BY: -

INPUT:

· pTs object (`y`)

OUTPUT:

· pTs object

COMMENT: correct?

distinct.col produces a color palette with easily distinguishable colors (using the R tool colorRampPalette)

USES: -
IS USED BY: -

INPUT:

· scalar (`n`; number of colors to be generates)

OUTPUT:

· vector of character strings (color names)

COMMENT: correct?

drunif generates random samples out of the finite set {0,1,. . . ,n} (discrete version of the R function runif)

USES: -
IS USED BY: -

INPUT:

· scalar (`n`; sample size)

⋆ OPTIONAL INPUT:

· scalar (`min=0`; start value)
· scalar (`max=1`; end value)

OUTPUT:

· vector (random sample)

COMMENT: correct? isn't is easier to use the R tool sample.int?

end.own extracts the last time point from a time series/pTs object/pField

USES: -
IS USED BY: season.pTs

INPUT:

- ts/pTs object /pField object (`n`)

OUTPUT:

- scalar (end time)

COMMENT: correct?

enoise.pTs generates a random sample that has approximately the same distribution as an observed time series

USES: -
IS USED BY: -

INPUT:

- pTs object (`ts`)

OUTPUT:

- pTs object

COMMENT: correct? is enoise.pTs not just a simple permutation?

eval.region ? "extracts regional temperature and plots a summary to pdf"

USES: regional.ts, list2pTs, applyspace, pTs, snoise.pTs
IS USED BY: -

INPUT:

- scalar (`lat1`; latitude no. 1)
- scalar (`lat2`; latitude no. 2)
- scalar (`lon1`; longitude no. 1)
- scalar (`lon2`; longitude no. 2)
- scalar (`mean.window`; (width of the rolling mean window, contrary to medsmooth interpreted as total window size)

⋆ OPTIONAL INPUT:

- scalar (`from=1000`; start time of considered time window)
- scalar (`to=2000`; end time of considered time window)
- character string (`region="global"`; description of the region selected)
- boolean (`PLOT=TRUE`; plots data if desired and saves plot to pdf)

OUTPUT:

- pTs object
- (plot)

COMMENT: description incomplete; unclear if function works properly, as obviously no input data are used; variable `FILENAME` not used

fastcor.pTs computes the correlation between a pTs and a pField object

USES: pField
IS USED BY: -

INPUT:

- pTs object (`pTs`)
- pField object (`pField`)

OUTPUT:

- pField object

COMMENT: correct?

filled.contour.own plots a 2D field and is essentially the same as the R function filled.contour (one advantage of filled.contour.own is that colored backgrounds are allowed)

USES: -
IS USED BY: plotmap.pField, hovmoeller

INPUT:

- matrix (`z`; data, e.g., ordered with respect to longitude and latitude)

⋆ OPTIONAL INPUT:

- vector (`x=seq(0,1,len=nrow(z))`; x-values, e.g., longitudes)
- vector (`y=seq(0,1,len=ncol(z))`; y-values, e.g., latitudes)
- vector (`xlim=range(x,finite=TRUE)`; defines the x-axis limits of the plot window)
- vector (`ylim=range(y,finite=TRUE)`; defines the y-axis limits of the plot window)
- vector (`zlim=range(z,finite=TRUE)`; range of data values to be plotted; if not specified, `zlim` is defined by the observed data set range)
- vector (`levels=pretty(zlim,nlevels)`; a set of levels which are used to partition the range of the data in terms of color; the default has `nlevels` steps on the `zlim` range)
- scalar (`nlevels=20`; number of levels used for the partition in levels)
- function (`color.palette = cm.colors`; color palette used for plotting)
- vector of character strings (`col=color.palette(length(levels)-1)`; colors used for plotting, the default draws length(levels)-1) color names from the color palette)
- character string (`set.bg=NULL`; background color, the default being white/transparent)
- graphic variables in plot.title (e.g., other data to be plotted (see plotmap.pField))
- ...

OUTPUT:

- plot

COMMENT: correct?

filter.pField filters a pField object (e.g., filter.pField removes frequencies from each gridpoint's time series that are higher, lower or both higher and lower than some specified boundaries)

USES: getlat, getlon, getname, gethistory
IS USED BY: -

INPUT:

- pField object (`field`)
- vector (`Filter`; filter)
- scalar (`f.time`; cuts off `f.time` time units at the start and at the end of the pField object, respectively)

⋆ OPTIONAL INPUT:

- ...

OUTPUT:

- pField object (filtered pField object)

COMMENT: correct?

filter.pTs filters a pTs object (e.g., filter.pTs removes frequencies from a time series that are higher, lower or both higher and lower than some specified boundaries)

USES: getlat, getlon, getname, gethistory
IS USED BY: -

INPUT:

- pTs object (`data`)
- vector (`filter`; filter)

⋆ OPTIONAL INPUT:

- ...

OUTPUT:

- pTs object (filtered pTs object)

COMMENT: correct?

filter.pTs1 filters a pTs object (e.g., filter.pTs removes frequencies from a time series that are higher, lower or both higher and lower than some specified boundaries); as filter.pTs, but with modified boundary conditions

USES: getlat, getlon, getname, gethistory
IS USED BY: -

INPUT:

- pTs object (`data`)
- vector (`filter`; filter)

⋆ OPTIONAL INPUT:

- scalar (`method=1`; ? `method`=1 (the default) or =2 or =3)
- ...

OUTPUT:

- pTs object (filtered pTs object)

COMMENT: input description incomplete (what does method stand for?)

find.var searches in ncdf file for variable names

USES: -
IS USED BY: read_data

INPUT:

- ncdf data (`data.nc`; open ncdf file)
- vector of character strings (`searched_vars`; possible variable names)

OUTPUT:

- vector (variable name, index of variable)

COMMENT: correct?

first returns first element of a vector/pTs object/pField object

USES: -
IS USED BY: -

INPUT:

- vector /pTs object /pField object (`x`)

OUTPUT:

- scalar /element (first element of the vector/pTs object/pField object)

COMMENT: correct?

fisher ?

USES: -
IS USED BY: change.cor.test

INPUT: ?

OUTPUT:

- scalar (RMSE)

COMMENT: description incomplete

get.a1 computes the autocorrelation at lag 1 of a vector/pTs object using the R autocorrelation function acf

USES: -
IS USED BY: -

INPUT:

- scalar /vector (`x`; ?)

OUTPUT:

- scalar /vector (?)

COMMENT: description incomplete

get.transfer computes the transfer function given a filter

USES: -
IS USED BY: -

INPUT: ?

OUTPUT:

- ? list: ?
- (plot)

COMMENT: description incomplete

gethistory returns the "history" attribute of a pTs/pField object

USES: -
IS USED BY: e.g., filter.pTs, filter.pTs1, scale.pTs, prcompO.pTs

INPUT:

- pTs / pField object (`data`; more general: an object with a "history" attribute)

OUTPUT:

- character string ("history" attribute)

COMMENT: correct?

getilat ?

USES: -
IS USED BY: -

INPUT:

- vector (`lat`; latitudes)

OUTPUT:

- vector (modified latitudes)

COMMENT: description incomplete

getlat returns the "lat" (latitude) attribute of a pTs/pField object

USES: -
IS USED BY: e.g., maxpoint, minpoint, latlonField, list2pTs, filter.pTs

INPUT:

- pTs / pField object (`data`; more general: an object with a "lat" attribute)

OUTPUT:

- vector (latitudes)

COMMENT: correct?

getlon returns the "lon" (longitude) attribute of a pTs/pField object

USES: -
IS USED BY: e.g., maxpoint, minpoint, latlonField, list2pTs, filter.pTs

INPUT:

- pTs / pField object (`data`; more general: an object with a "lon" attribute)

OUTPUT:

- vector (latitudes)

COMMENT: correct?

getname returns the "name" attribute of a pTs/pField object

USES: -
IS USED BY: e.g., applyspace, applytime, index.nino1.2, list2pTs, filter.pTs

INPUT:

- pTs / pField object (`data`; more general: an object with a "name" attribute)

OUTPUT:

- character string (name) / vector of character strings (names)

COMMENT: correct?

getS0 ? computes the variance of an AR(1) process for given innovation variance and AR(1) coefficient?

USES: -
IS USED BY: -

INPUT:

- scalar (`sigma`; innovation variance?)
- scalar (`a0`; AR(1) coefficient?)

OUTPUT: scalar (process variance?)

COMMENT: description unclear

gphcontour adds gph (geopotential height) contour lines to a 2D field plot (using the R tool contour), so that positive correlations are visualized by solid and negative correlations by dashed lines

USES: -
IS USED BY: -

INPUT:

- vector (`lon`; longitudes)
- vector (`lat`; latitudes)
- matrix (`data`; 2D gph field)

OUTPUT:

- inserted into the current plot

COMMENT: correct?

highpass derives the smoothed least square high-pass filter (given the cut-off frequency and the length of the filter)

USES: lowpass
IS USED BY: -

INPUT:

- scalar (`omega.c`; cutoff frequency)

⋆ OPTIONAL INPUT:

- scalar (`n=9`; length of the filter)
- scalar (`sample=1`; ?)
- boolean (`convergence=T`; ?)

OUTPUT:

- vector (highpass filter)

COMMENT: description incomplete

hovmoeller plots latmean output anomalies (by averaging over time)

USES: latmean, filled.contour.own, getlat
IS USED BY: -

INPUT:

- pField object (`data`)

⋆ OPTIONAL INPUT:

- vector (`refperiod=c(start(data)[1],end(data)[1])`; time range of interest)
- character string (`xlab="time"`; x-axis label)
- character string (`ylab="latitude"`; y-axis label)
- function (`FUN=contour(time(data),getlat(data),zmeans.anomaly,add=T)`; function to be plotted)
- ...

OUTPUT:

- plot

COMMENT: correct?

icecontour adds ??? a contour line to a 2D field plot (using the R tool contour)

USES: -
IS USED BY: -

INPUT:

- vector (`lon`; longitudes)
- vector (`lat`; latitudes)
- matrix (`data`; 2D ??? field)

OUTPUT:

- inserted into the current plot

COMMENT: description incomplete - what does this contour line have to do with ice?

identifyPch helps identify coordinates in a plot using the R tools xy.coords and identify

USES: -
IS USED BY: PickN

INPUT:

- vector (`x`; x-axis coordinates)

⋆ OPTIONAL INPUT:

- vector (`y=NULL`; y-axis coordinates)
- scalar (`n=6`; number of coordinate pairs to be determined)
- scalar (`pch=19`; type of plotted points, see R help for par)
- vector of character strings (`colors=rep(c("blue","green","red"),2)`; defining the colors of the points)

OUTPUT:

- vector (indices of identified coordinate pairs)
- inserted into the current plot

COMMENT: correct?

index.ao ?

USES: selspace, prcompO.pField, detrend, getname
IS USED BY: -

INPUT:

- pField object (`slp`; ?)

OUTPUT:

- pField object (?)

COMMENT: description incomplete

index.pna ?

USES: selspace, getname
IS USED BY: -

INPUT:

- pField object (`gph`; ?)

OUTPUT:

- pField object (?)

COMMENT: description incomplete

index.nao ?

USES: selspace, prcompO.pField, scale.pField, detrend, lmSlope.pTs, getname
IS USED BY: -

INPUT:

- pField object (`slp`; ?)

⋆ OPTIONAL INPUT:

- boolean (`plot=F`; indicates if a plot should be made)
- boolean (`pattern=F`; ?)
- boolean (`scale=F`; ?)

OUTPUT:

- list: pField object (?)
- . . . ?

COMMENT: description incomplete

index.nino.tni ?

USES: index.nino1.2, index.nino4, getname

IS USED BY: -

INPUT:

- pField object (`sst`; ?)

OUTPUT:

- pTs object (?)

COMMENT: description incomplete

index.nino1.2 ?

USES: selspace, applyspace, getname
IS USED BY: index.nino.tni

INPUT:

- pField object (`sst`; ?)

OUTPUT:

- pTs object (?)

COMMENT: description incomplete

index.nino3 ?

USES: selspace, applyspace, getname
IS USED BY: -

INPUT:

- pField object (`sst`; ?)

OUTPUT:

- pTs object (?)

COMMENT: description incomplete

index.nino3.4 ?

USES: selspace, scale.pField, getname
IS USED BY: -

INPUT:

- pField object (`sst`; ?)

OUTPUT:

- pTs object (?)

COMMENT: description incomplete

index.nino4 ?

USES: selspace, applyspace, getname
IS USED BY: index.nino.tni

INPUT:

- pField object (`sst`; ?)

OUTPUT:

- pTs object (?)

COMMENT: description incomplete

index.soi ?

USES: selspace, getname
IS USED BY: -

INPUT:

- pField object (`slp`; ?)

OUTPUT:

- pField object (?)

COMMENT: description incomplete

index.soimodel ?

USES: selspace, getname
IS USED BY: -

INPUT:

- pField object (`slp`; ?)

OUTPUT:

- pField object (?)

COMMENT: description incomplete

ins.dec21 ?

USES: daily_insolation, tlag
IS USED BY: -

INPUT:

- scalar (`kyear`; kyr before present)
- scalar (`LAT`; latitude)

OUTPUT: ?

COMMENT: description incomplete

ins.dec21.param ?

USES: daily_insolation_param, tlag
IS USED BY: -

INPUT:

- scalar (`ecc`; eccentricity)
- scalar (`obliquity`; obliquity)

- scalar (`long_perh`; precession?)
- scalar (`LAT`; latitude)

OUTPUT: ?

COMMENT: description incomplete

ins.march21 ?

USES: daily_insolation
IS USED BY: -

INPUT:

- scalar (`kyear`; kyr before present)
- scalar (`LAT`; latitude)

OUTPUT: ?

COMMENT: description incomplete

is_pField checks if an object has the pField format

USES: -
IS USED BY: Ops.pField

INPUT:

- object (`data`; potentially pField object)

OUTPUT:

- boolean (T (is a pField object) or F (is not a pField object))

COMMENT: correct?

is_pTs checks if an object has the pTs format

USES: -
IS USED BY: Ops.pTs

INPUT:

- object (`data`; potentially pTs object)

OUTPUT:

- boolean (T (is a pTs object) or F (is not a pTs object))

COMMENT: correct?

julday.own computes Julian days from ? using the R tool julday

USES: -
IS USED BY: read_data

INPUT: ?

OUTPUT:

- scalar (julian days)

COMMENT: description incomplete - input format unclear

last returns last element of a vector/pTs object/pField object

USES: -
IS USED BY: -

INPUT:

- vector /pTs object /pField object (`x`)

OUTPUT:

- scalar /element (last element of the vector/pTs object/pField object)

COMMENT: correct?

latlonField extracts longitudes and latitudes from a pField object and returns vectors that assign latitudes and longitudes to each element of a 2D field displayed as a vector; see output information

USES: getlon, getlat
IS USED BY: schwerpunkt

INPUT:

- pField object (`data`)

OUTPUT:

- list: vector (vector[i]=longitude that belongs to c(2D field)[i]) , vector (vector[i]=latitude that belongs to c(2D field)[i])

COMMENT: correct?

latmean averages a pField object over longitudes so that values exist for each pair of latitude and time step

USES: -
IS USED BY: hovmoeller, zonalmean

INPUT:

- pField object (`data`)

OUTPUT:

- matrix (pField averaged over longitudes; matrix [i,j]=i^{th} time step, j^{th} latitude)

COMMENT: correct?

list2pTs converts a list of single pTs timeseries (having the same length) into one pTs object

USES: getname, getlon, getlat, pTs
IS USED BY: millenium.anomalies, millenium.temperatures, eval.region

INPUT:

- list (`x`; with pTs objects as its entries)

OUTPUT:

- pTs object (in which all pTs objects are stored)

COMMENT: correct?

lmSlope.pTs carries through a linear regression between a pTs object (independent variable) and a pTs/pField object (dependent variable) and saves out the slope estimate (using the R regression tool lm)

USES: copyattr
IS USED BY: index.nao

INPUT:

- pTs object (`pTsS`; independent variable)
- pTs / pField object (`pFieldS`; dependent variable)

⋆ OPTIONAL INPUT:

- boolean (`debug=F`; debugging variable)

OUTPUT:

- scalar / vector (slope estimates)

COMMENT: correct?

load_huascara ? reads in huascara data and generates a pTs objects

USES: pTs
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)

OUTPUT:

- pTs object (?)

COMMENT: description incomplete

lopt ?

USES: -
IS USED BY: rampfit

INPUT:

- scalar (`a1`; ?)
- scalar (`n`; ?)

OUTPUT:

- scalar (?)

COMMENT: description incomplete

lowpass derives the smoothed least square low-pass filter (given the cutoff frequency and the length of the filter)

USES: -
IS USED BY: highpass, bandpass

INPUT:

- scalar (cutoff frequency)

⋆ OPTIONAL INPUT:

- scalar (`n=9`; length of the filter)
- scalar (`sample=1`; ?)
- boolean (`convergence=T`; ?)

OUTPUT:

- vector (lowpass filter)

COMMENT: description incomplete

makefilm averages a pField object over different time intervals (similar to the roll1 and roll2 functions) using the R tool rollmean (R library zoo) and saves out 2D field plots (jpeg format) for each considered interval

USES: applyspace, applytime
IS USED BY: -

INPUT:

- pField object (`data`)
- scalar (`startdate`; start time)
- scalar (`enddate`; end time)

⋆ OPTIONAL INPUT:

- scalar (`step=5`; distance between intervals)
- scalar (`avrg=11`; number of time steps to be averaged over)
- character string (`prefix="ani_"`; prefix of the jpeg files to be generated)
- boolean (`anomaly=FALSE`; work with absolute (anomaly=F) or anomaly values (`anomaly`=T, the default)
- ...

OUTPUT:

- jpeg files

COMMENT: correct?

maxpoint searches for the maximum element of a pTs/pField object and returns corresponding latitude and longitude values (as well as the maximum element itself)

USES: getlat, getlon, c1t2
IS USED BY: -

INPUT:

- pTs / pField object (`data`)

OUTPUT:

- list: scalar (latitude where the pTs/pField object has its maximum) , scalar (longitude where the pTs/pField object has its maximum) , scalar (maximum value)

COMMENT: correct?

medsmooth computes the running mean, running median and running median absolute deviation (see also CLIM-X-DETECT: A Fortran 90 program for robust detection of extremes against a time-dependent background in climate records (Mudelsee, 2006)) of a vector/pTs object

USES: -
IS USED BY: smoothspec

INPUT:

- vector /pTs object (`x`)
- scalar (`m`; defining the running mean window size, i.e., `m` adjacent elements are considered in either direction)

OUTPUT:

- list: vector (running mean) , vector (running median) , vector (running median absolute deviation)

COMMENT: correct?

mergeattr merges attributes of different sources (similar to copyattr)

USES: -
IS USED BY: -

INPUT:

- data object (`data`; e.g., pTs/pField object)
- object with attributes (`source1`; source no.1, e.g., pTs/pField object)
- object with attributes (`source2`; source no.2, e.g., pTs/pField object)

⋆ OPTIONAL INPUT:

- character string (`newhistory=''`; new history to be added to the data object (e.g., a comment))

OUTPUT:

- data object with attributes (e.g., pTs/pField object)

COMMENT: correct?

millenium.anomalies extracts the temperature anomalies as averages from the MPI COSMOS Milleniums data using the R tool rollmean (R library zoo) (the milleniums data folder must be in the root directory)

USES: read_data, selspace, areamean, list2pTs
IS USED BY: -

INPUT: -
⋆ OPTIONAL INPUT:

- scalar (`lat1=-90`; latitude no.1)
- scalar (`lat2=90`; latitude no.2)
- scalar (`lon1=0`; longitude no.1)
- scalar (`lon2=360`; longitude no.2)
- scalar (`from=1000`; start time)
- scalar (`to=2000`; end time)
- scalar (`mean.window=1`; width of the rolling mean window, contrary to medsmooth interpreted as total window size)
- scalar (`ref.min=1961`; ? reference time no.1)
- scalar (`ref.max=1990`; ? reference time no.2)

OUTPUT:

- pTs object (?)

COMMENT: description incomplete; difference between from/to and ref.min/ref.max as well as output format unclear

millenium.aod ? using the R tool rollmean (R library zoo) (the milleniums data folder must be in the root directory)

USES: pTs
IS USED BY: -

INPUT: -
⋆ OPTIONAL INPUT:

- scalar (`from=1000`; start time)
- scalar (`to=2000`; end time)
- scalar (`mean.window=1`; width of the rolling mean window, contrary to medsmooth interpreted as total window size)

OUTPUT:

- list: ?

COMMENT: description incomplete

millenium.temperatures millenium.temperatures() extracts the temperature as average from the MPI COSMOS Milleniums data

USES:
IS USED BY:

INPUT: -
⋆ OPTIONAL INPUT:

- scalar (`lat1=-90`; latitude no.1)
- scalar (`lat2=90`; latitude no.2)
- scalar (`lon1=0`; longitude no.1)
- scalar (`lon2=360`; longitude no.2)
- scalar (`from=1000`; start time)
- scalar (`to=2005`; end time)
- scalar (`rolling.mean=1`; width of the rolling mean window, contrary to medsmooth interpreted as total window size)

OUTPUT:

- pTs object (?)

COMMENT: description incomplete

millenium.tsi ?

USES: pTs
IS USED BY: -

INPUT: -
⋆ OPTIONAL INPUT:

- scalar (`from=1000`; start time)
- scalar (`to=2005`; end time)
- scalar (`mean.window=1`; width of the rolling mean window, contrary to medsmooth interpreted as total window size)

OUTPUT:

- pTs object (?)

COMMENT: description incomplete

minpoint searches for the minimum element of a pTs/pField object and returns corresponding latitude and longitude values (as well as the minimum element itself)

USES: getlat, getlon, c1t2
IS USED BY: -

INPUT:

- pTs / pField object (`data`)

OUTPUT:

- list: scalar (latitude where the pTs/pField object has its minimum) , scalar (longitude where the pTs/pField object has its minimum) , scalar (minimum value)

COMMENT: correct?

myccf ? calls the R tool ccf in order to compute cross-correlations/cross-covariances between two vectors/ts objects/pTs objects

USES: -
IS USED BY: ccf.pTs

INPUT:

- vector / ts object / pTs object (`ts1`)
- vector / ts object / pTs object (`ts2`)

OUTPUT:

- scalar (?)

COMMENT: description incomplete

mycor.test computes the correlation of two vectors/ts objects/pTs objects and produces a p-value of the correlation (calling the R tool cor.test)

USES: -
IS USED BY: cortest.pTs

INPUT:

- vector / ts object / pTs object (`v1`)
- vector / ts object / pTs object (`v2`)

⋆ OPTIONAL INPUT:

- ...

OUTPUT:

- vector (correlation and p-value)

COMMENT: correct?

myfun1 adds a grid, land contours and a main as well as a subtitle to a plot

USES: addland

IS USED BY: -

INPUT:

- character string (`sTitle`; main title)
- character string (`sSub`; subtitle)

OUTPUT:

- inserted into the current plot

COMMENT: correct?

myfun2 adds a grid, land contours and a main as well as a subtitle to a plot. Also, contour lines are drawn for a 2D field (differing colors for negative and positive field values)

USES: addland

IS USED BY: -

INPUT:

- character string (`sTitle`; main title)
- character string (`sSub`; subtitle)
- vector (`lat`; latitudes)
- vector (`lon`; longitudes)
- matrix (`plotdata`; data)

OUTPUT:

- inserted into the current plot

COMMENT: correct?

na.apply applies a function to time series of a pField object that do not contain NA values

USES: -
IS USED BY: -

INPUT:

- pField object (`x`)
- function (`FUN`)

⋆ OPTIONAL INPUT:

- ...

OUTPUT:

- pField object (with function output included)

COMMENT: correct?

ngt ? ngt returns all possible ngt measurements as a list of pTs objects

USES: pTs
IS USED BY: -

INPUT: -
⋆ OPTIONAL INPUT:

- scalar (`from=1500`; start time)
- scalar (`to=1991`; end time)
- scalar (`mean.window=1`; ? is not made use of)

OUTPUT:

- pTs object

COMMENT: description incomplete, function seems to be faulty

normcontour0 adds ??? contour lines to a 2D field plot (using the R tool contour), so that positive correlations are visualized by solid and negative correlations by dashed lines

USES: -
IS USED BY: -

INPUT:

- vector (`lon`; longitudes)
- vector (`lat`; latitudes)
- matrix (`data`; 2D ??? field)

OUTPUT:

- inserted into the current plot

COMMENT: description incomplete

normcontour1 adds ??? contour lines to a 2D field plot (using the R tool contour), so that positive correlations are visualized by solid and negative correlations by dashed lines

USES: -
IS USED BY: -

INPUT:

- vector (`lon`; longitudes)
- vector (`lat`; latitudes)
- matrix (`data`; 2D ??? field)

OUTPUT:

- inserted into the current plot

COMMENT: description incomplete

Ops.pField ?

USES: is_pField, pField, addhistory
IS USED BY: -

INPUT: ?

OUTPUT:

- pField object (?)

COMMENT: description incomplete

Ops.pTs ?

USES: is_pTs, pTs, addhistory
IS USED BY: -

INPUT: ?

OUTPUT:

- pTs object (?)

COMMENT: description incomplete

orbital_parameters loads insolation data (downloaded as ORBIT91 from ncdc.noaa.gov) and returns orbital parameters for a requested kiloyear in the past (the data covers the time span 0-5 million yr BP) using the R tool spline

USES: unwrap
IS USED BY: daily_insolation

INPUT:

- scalar (`kyear`; kyr of interest)
- character string (`FILEDATA`; path of the file)

OUTPUT:

- list: ? vector (eccentricity) , ? vector (obliquity) , ? vector (? precession)

COMMENT: description incomplete

orbital_parameters_fast ?

USES: ?
IS USED BY: daily_insolation

INPUT:

- scalar (`kyear`; kyr of interest)

OUTPUT:

- list: ? vector (eccentricity) , ? vector (obliquity) , ? vector (? precession)

COMMENT: description incomplete; does this function work, by the way? is there no data input?!

paleo.symbols adds typical symbols to a plot (e.g., "Temperature [°C]" as a y-label: plot(1:10,ylab=paleo.symbols()))

USES: -
IS USED BY: -

INPUT: -
⋆ OPTIONAL INPUT:

- character string (`paramet="Temperature"`; symbol type: either "Temperature" (the default) or "Latitude")

OUTPUT: ?

COMMENT: description incomplete - @output: how can a "substitute"-variable described best?

paleo.symbols.temperature adds a "Temperature [°C]" symbol to a plot (e.g., as a y-label: plot(1:10,ylab=paleo.symbols.temperature()))

USES: -
IS USED BY: eval.region

INPUT: -
⋆ OPTIONAL INPUT:

- character string (`paramet="Temperature"`; symbol type: "Temperature" (the default), no other options (see paleo.symbols))

OUTPUT: ?

COMMENT: description incomplete - @output: how can a "substitute"-variable described best?

par.uin ? "this is just a definition of a function to plot vectorplots"

USES: -
IS USED BY: quiver

INPUT: -

OUTPUT:

- vector (?)

COMMENT: description incomplete

pcor detrends a pField object and carries out a correlation test for a time series extracted from the field and the field itself

USES: detrend, cortest.pTs, selspace
IS USED BY: -

INPUT:

- pField object (`data`)
- scalar (`p1`; latitude of the time series to be extracted from the pField object)
- scalar (`p2`; longitude of the time series to be extracted from the pField object)
- scalar (`t1`; start time)
- scalar (`t2`; end time)

OUTPUT:

- two pField objects saved in one variable (observed correlations and corresponding p-values), can be extracted row-wise

COMMENT:

pField adds attributes such as longitudes and latitudes to a gridded data array (which in most cases is three-dimensional (where the dimensions stand in general for longitude, latitude and time)) and assigns the class "pField" to the resulting object

USES: -
IS USED BY: e.g., Ops.pField, "[.pField", read_data, season.pTs, selspace

INPUT:

- array (`data`; data, mostly 3D)
- vector (`time`; time)

⋆ OPTIONAL INPUT:

- vector (`lat=0`; latitudes)
- vector (`lon=0`; longitudes)
- character string (`name=" "`; name of the pField object to be generated)
- character string (`history=" "`; history, helps to keep track of possible changes made to the pField object)
- boolean (`date=T`; adds the date of creation/modification in the format "Thu May 19 09:48:10 2011" to the object if `date`=T)

OUTPUT:

- pField object (gridded 3D field with class "pField")

COMMENT: correct? btw: what happens if lon=0 and lat=0?

PickN identifies coordinates using the identifyPch routine in the case of multiple screens

USES: identifyPch
IS USED BY: -

INPUT:

- list / vector (`x`; x-axis coordinates)

- list (`y`; y-axis coordinates)
- scalar (`i`; screen number)

⋆ OPTIONAL INPUT:

- scalar (`NPick=6`; numbers of points to be identified)

OUTPUT:

- vector (indices of identified coordinate pairs)
- inserted into the current plot

COMMENT: correct?

plot.pField plots a 2D pField object pfield calling the plotmap routine; note that with class(pfield)="pField", one writes plot(pfield) instead of plot.pField(pfield)

USES: plotmap, plotmap.pField
IS USED BY: e.g., plot.sig (implicitly, see above)

INPUT:

- 2D pField object (`x`; data)

⋆ OPTIONAL INPUT:

- ... (see plotmap.pField)

OUTPUT:

- plot

COMMENT: correct? connection between plotmap and plotmap.pField unclear

plot.Polygon adds polygons (e.g., contour lines that were calculated with the R tool contourLines) to a plot using the R function polygon (the same function as plotSig)

USES: -
IS USED BY: plot.sig

INPUT:

- list (`sigline`; contourLines output, specifying the polygon vertices)

OUTPUT:

- inserted into the current plot

COMMENT: correct? the same function as plotSig

plot.preparation arranges a 2D pField object for plotting (e.g., puts Europe in the center of the 2D land contour plot if desired, reverses the latitudes if they are stored in the (90,-90) format instead of (-90,90))

USES: -
IS USED BY: plotmap.pField, plot.sig, plotmap.square, addwind, plotmap.pFieldb

INPUT:

- 2D pField object (`plotdata`; data)

⋆ OPTIONAL INPUT:

- boolean (`shift=F`; puts Europe in the plot window center if `shift`=T)
- boolean (`long=F`; `long`=F (the default) and `long`=T (if the 2D field is originally separated by the default land contour plot, extends the plot window so that a region of interest appears both on the left and on the right side of the plot to give a continuous display; otherwise, the plotting spares the original part on the left hand and only shows the continuous field part on the right hand)); note that `shift` and `long` cannot be set to TRUE at the same time

OUTPUT:

- matrix (adjusted 2D field)
- vector (adjusted latitudes)
- vector (adjusted longitudes)

COMMENT: correct?

plot.pTs plots a pTs time series object; note that with class(pts)="pTs", one writes plot(pts) instead of plot.pTs(pts)

USES: getname
IS USED BY: e.g., plot.unicor (implicitly, see above)

INPUT:

- pTs object (`x`)

⋆ OPTIONAL INPUT:

- character string (`plot.type="multiple"`; `plot.type`="multiple" (the default?) or ="single"; decides if a pTs object with multiple time series but same time basis is plotted into multiple ("multiple") or one ("single") plot window(s))
- ... (see R help for plot.ts)

OUTPUT:

- plot

COMMENT: correct? what is the plot.type default?

plot.sig visualizes the output of a pField/pTs-pTs correlation (used with plot.unicor), highlighting significant areas

USES: rbow.col.nonsigarea, sigline.preparation, plot.Polygon, plot.preparation, corcontour, plot.pField
IS USED BY: plot.unicor

INPUT:

- 2D pField object (`plotmap`; correlations)
- 2D pField object (`sigmap`; p-values)

⋆ OPTIONAL INPUT:

- character string (`plot_sig="nocol.nonsigarea"`; per default, contour lines are computed for the entire area, while color is used for significant correlation values only; if `plot_sig`="col.nonsigarea", all correlation values appear in color, and significant areas are highlighted (shaded in blue))

- scalar (`crit_val=0.05`; critical value)
- function (`FUN=NULL`; e.g., for adding specified points to the plot)
- function (`palette=NULL`; color palette to be used; per default, the rbow.col.nonsigarea palette is selected))
- ...

OUTPUT:

- plot

COMMENT: correct?

plotcont.pField plots the contours of a 2D pField object (similar to plotmapc.pField, but without grid and land contours)

USES: -
IS USED BY: -

INPUT:

- 2D pField object (`plotdata`)

⋆ OPTIONAL INPUT:

- character string (`sTitle=NULL`; plot title)
- ...

OUTPUT:

- plot

COMMENT: correct? seems to be an old function that is no longer in use (e.g., the plot.preparation function is not called but entirely implemented)

ploteof ?

USES: getlat, getlon, getname, addland
IS USED BY: -

INPUT: -

OUTPUT:

- plot

COMMENT: description incomplete

plotindex plots a pTs time series object in the histogram format so that positive values appear in red and negative values in blue color (using the R function plot.ts); also computes and adds a running mean to the plot (using the R tool rollmean)

USES: getname
IS USED BY: -

INPUT:

- pTs object (`data`)

⋆ OPTIONAL INPUT:

- scalar (`r.intervall=5`; width of the rolling mean window, contrary to medsmooth interpreted as total window size)
- scalar (`lwd=3`; width of the histogram bars)
- scalar (`lwd1=2`; width of the running mean line)
- character string (`main=NULL`; plot title)

OUTPUT:

- plot

COMMENT: correct?

plotmap is used to plot a 2D pField object

USES: ? plotmap.pField ?
IS USED BY: plot.pField

INPUT:

- 2D pField object (`plotdata`)

⋆ OPTIONAL INPUT:

- ... (see plotmap.pField)

OUTPUT:

- plot

COMMENT: description incomplete; connection between plotmap and plotmap.pField unclear

plotmap.pField plots a 2D pField object using filled.contour.own (and thereby the R function filled.contour.own)

USES: plot.preparation, filled.contour.own, addland, axes.type, rbow
IS USED BY: ? plotmap ?

INPUT:

- 2D pField object (`plotdata`)

⋆ OPTIONAL INPUT:

- character string (`main=NULL`; title of the plot; if not provided, the pField name is used)
- vector (`zlim=range(plotdata,finite=TRUE)`; range of data values to be plotted; if not specified, `zlim` is defined by the observed data set range)
- scalar (`nlevels=20`; number of levels taken for the partition in levels)
- vector (`levels=pretty(zlim,nlevels)`; a set of levels which are used to partition the range of the data in terms of color; the default has `nlevels` steps on the `zlim` range)
- scalar (`nlevels=20`; number of levels used for the partition in levels)
- function (`palette=NULL`; color palette used for plotting; the default is a rainbow scale given through the rbow function)
- function (`FUN=NULL`; additional function to be included in the plot)
- boolean (`shift=F`; puts Europe in the plot center if desired (`shift`=T); else, the default contours are displayed (`shift`=F); see also plot.preparation)
- boolean (`long=F`; see plot.preparation)

- vector (`xlim=NULL`; defines the x-axis limits of the plot window; the default takes the longitudes of the input field)
- vector (`ylim=NULL`; defines the y-axis limits of the plot window; the default takes the latitudes of the input field)
- character string (`sSub=NULL`; subtitle of the plot; if not provided, no subtitle is produced)
- character string (`set.bg=NULL`; background color appearing in the plot; white is the default)
- character string (`gridcolor="lightgray"`; color of the grid)

OUTPUT:

- plot

COMMENT: correct?

plotmap.pFieldb plots two 2D pField objects in one plot using the R function filled.contour.own

USES: rbow, addland, plot.preparation
IS USED BY: -

INPUT:

- 2D pField object (`plotdata`)
- 2D pField object (`plotdata2`)

⋆ OPTIONAL INPUT:

- function (`palette=rbow`; color palette to be used for plotting)
- ... (for the remaining arguments see plotmap.pField)

OUTPUT:

- plot

COMMENT: correct?

plotmap.square plots a 2D pField object by calling the plotsquare routine (which uses the R function image)

USES: addland, plot.preparation, plotsquare
IS USED BY: -

INPUT:

- 2D pField object (`plotdata`)

⋆ OPTIONAL INPUT:

- function (`palette=rbow`; color palette to be used for plotting)
- character string (`sSub=""`; plot subtitle)
- ... (for the remaining arguments see plotmap.pField)

OUTPUT:

- plot

COMMENT: correct?

plotmapc.pField plots the contours of a 2D pField object (similar to plotcont.pField, but with grid and land contours)

USES: -
IS USED BY: -

INPUT:

- 2D pField object (`plotdata`)

⋆ OPTIONAL INPUT:

- character string (`sTitle=NULL`; plot title)
- ...

OUTPUT:

- plot

COMMENT: correct? seems to be an old function that is no longer in

use (e.g., the plot.preparation function is not called but entirely implemented)

PlotMultiple plots data with multiple dimensions (e.g., lists); more general than the R function plot.ts

USES: -
IS USED BY: -

INPUT:

- list (`x`; values for the x-axes)
- list (`y`; values for the y-axes)

⋆ OPTIONAL INPUT:

- character string (`type="b"`; line type)
- character string (`xlab="time"`; x-axis label)
- vector /list (`xlim=range(x)`; the x-axis boundaries; either globally (vector) or specifically defined for each y component (list), the default being the global definition `xlim`=range(x))
- boolean (`bPoints=TRUE`; indicates if data should be additionally highlighted by the R function points)
- scalar (`pch=20`; number that specifies the type of point plotting if `bPoints`=TRUE, see R help for points)

OUTPUT:

- plot

COMMENT: correct?

plotSig adds polygons (e.g., contour lines that were calculated with the R tool contourLines) to a plot using the R function polygon (the same function as plot.Polygon)

USES: -
IS USED BY: plot.sig

INPUT:

- list (`sigline`; contourLines output, specifying the polygon vertices)

OUTPUT:

- inserted into the current plot

COMMENT: correct? the same function as plot.Polygon

plotsquare plots a 2D field and is similar to the R function filled.contour, but uses the R function image instead of .Internal(filledcontour(...))

USES: -
IS USED BY: plotmap.square

INPUT:

- matrix (`z`; data, e.g., ordered with respect to longitude and latitude)

⋆ OPTIONAL INPUT:

- vector (`x=seq(0,1,len=nrow(z))`; x-values, e.g., longitudes)
- vector (`y=seq(0,1,len=ncol(z))`; y-values, e.g., latitudes)
- vector (`xlim=range(x,finite=TRUE)`; defines the x-axis limits of the plot window)
- vector (`ylim=range(y,finite=TRUE)`; defines the y-axis limits of the plot window)
- vector (`zlim=range(z,finite=TRUE)`; range of data values to be plotted; if not specified, `zlim` is defined by the observed data set range) ,
- vector (`levels=pretty(zlim,nlevels)`; a set of levels which are used to partition the range of the data in terms of color; the default has `nlevels` steps on the `zlim` range)
- scalar (`nlevels=20`; number of levels used for the partition in levels)
- function (`color.palette = cm.colors`; color palette used for plotting)
- vector of character strings (`col=color.palette(length(levels)-1)`; colors used for plotting, the default draws length(levels)-1) color names from the color palette)

- character string (`set.bg=NULL`; background color, the default being white/transparent) , graphic variables in plot.title (e.g., other data to be plotted (see plotmap.pField))
- ...

OUTPUT:

- plot

COMMENT: correct?

plot.unicor function that correlates a 3D field with a time series, carries out a significance test if desired and plots the results

USES: getlon, getlat, selspace, cor.filter, cortest.pTs, cor.pTs, plot.sig
IS USED BY: -

INPUT:

- pField object (`field`)
- pTs object (`ts`)

⋆ OPTIONAL INPUT:

- vector (`area=NULL`; area over which the correlation shall take place; must be given as c(lon1,lon2,lat1,lat2); if `area`=NULL (the default), the entire field is used)
- character string (`plot_sig=NULL`; specifies whether a significance test is to be carried through; `plot_sig` must be set to NULL (the default), "col.nonsigarea" or "nocol.nonsigarea"; NULL: correlations are computed without performing significance analysis; "col.nonsigarea": all correlation values appear in color. significant areas are highlighted (shaded in blue); "nocol.nonsigarea": contour lines are computed for the entire area, while color is used for significant correlation values only)
- vector (`zlim=NULL`; defines the range of correlation values shown in the plot; per default, the whole range of computed correlation values is taken (before `plot_sig` is evaluated)) , list (`filt=NULL`; if `filt` is of the format `filt`=list(filter="lowpass",x=1/10, y=21,method=1), the data are filtered and an modified correlation test is applied; otherwise (`filt`=NULL, the default), the data are not filtered)
- ...

OUTPUT:

- plot

COMMENT: correct?

plotwind ? plots a 2D pField object using the R tool filled.contour

USES: getlon, getlat, addland
IS USED BY: -

INPUT:

- 2D pField object (`data`)

⋆ OPTIONAL INPUT:

- function (`palette=rbow`; color palette to be used for plotting)
- character string (`title=NULL`; plot title)
- ... (for the remaining arguments see plotmap.pField)

OUTPUT:

- plot

COMMENT: description incomplete; Why should there be any such function next to plotmap.pField?

pnacontour adds ??? a contour line to a 2D field plot (using the R tool contour)

USES: -
IS USED BY: -

INPUT:

- vector (`lon`; longitudes)
- vector (`lat`; latitudes)
- matrix (`data`; 2D ??? field)

OUTPUT:

- inserted into the current plot

COMMENT: description incomplete

prcompNA.pField ? principal component analysis of a pField object (?) using the R tool prcomp

USES: pTs, pField
IS USED BY: -

INPUT:

- pField object (`data`; ?)

⋆ OPTIONAL INPUT:

- ...?

OUTPUT:

- list: ?

COMMENT: description incomplete

prcompO.pField ? principal component analysis of a pField object (?) using the R tool prcomp

USES: pField, gethistory, addhistory
IS USED BY: -

INPUT:

- pField object (`data`; ?)

⋆ OPTIONAL INPUT:

- ...?

OUTPUT:

- list: ?

COMMENT: description incomplete

prcompO.pTs ? principal component analysis of a pTs object (?) using the R tool prcomp

USES: pTs, gethistory, addhistory
IS USED BY: -

INPUT:

- pTs object (`data`; ?)

⋆ OPTIONAL INPUT:

- ...?

OUTPUT:

- list: ?

COMMENT: description incomplete

pTs adds attributes such as longitude and latitude to a time series vector/time series vectors (having the same time basis) and assigns the class "pTs" to the resulting object

USES: -
IS USED BY: e.g., Ops.pField, "[.pField", read_data, season.pTs, selspace

INPUT:

- vector /matrix /ts object (`data`; data)
- vector (`time`; time)

⋆ OPTIONAL INPUT:

- scalar (`lat=0`; latitude)
- vector (`lon=0`; longitude;)
- character string (`name=""`; name of the pTs object to be generated)

- character string (`history=""`; history, helps to keep track of possible changes made to the pTs object (e.g., a comment))
- boolean (`date=T` ;adds the date of creation/modification in the format "Thu May 19 09:48:10 2011" to the object if `date`=T)

OUTPUT:

- pTs object (time series object with class "pTs")

COMMENT: correct? btw: what happens if lon=0 and lat=0?

pval.cor computes the two-sided p-value of an observed correlation coefficient using the t-distribution

USES: -
IS USED BY: -

INPUT:

- scalar (`r`; observed correlation)
- scalar (`N`; sample size)

OUTPUT:

- scalar (two-sided p-value)

COMMENT: correct?

quiver produces a vectorplot using the R tool arrows (see online R help); requires an already opened plot window

USES: -
IS USED BY: addwind

INPUT:

- vector (`lon`; longitudes)
- vector (`lat`; latitudes)
- matrix (`u`; x-axis vector components)
- matrix (`v`; y-axis vector components)

⋆ OPTIONAL INPUT:

- scalar (`scale=1`; regulates the vector lengths by scaling)
- scalar (`length=0.2`; length of the vector head)
- scalar (`maxv=max(abs(na.omit(u)),abs(na.omit(v)))`; also regulating the vector lengths by scaling)

OUTPUT:

- inserted into the current plot

COMMENT: correct?

ramp computes the modeled data values for given ramp parameters and levels (see see Mudelsee's paper: "Ramp function regression: a tool for quantifying climate transitions" (2000) and rampfit_xt)

USES: -
IS USED BY: rampfit_xt, rampfit.val

INPUT:

- vector (`t`; time)
- scalar (`i0`; index of the start of the flat part of the ramp)
- scalar (`i1`; index of the start of the ramp)
- scalar (`i2`; index of the end of the ramp)
- scalar (`i3`; index of the end of the second flat part)
- scalar (`x1`; level of the start of the ramp)
- scalar (`x2`; level of the end of the ramp)

OUTPUT:

- vector ("predicted" data values)

COMMENT: correct?

ramp_xfit computes the ramp levels for given data and ramp parameters (see see Mudelsee's paper: "Ramp function regression: a tool for quantifying climate transitions" (2000) and rampfit_xt)

USES: -
IS USED BY: rampfit_xt

INPUT:

- vector (`x`; data)
- vector (`t`; time)
- vector (`sigma2`; variances associated with the data)
- scalar (`i0`; index of the start of flat part of the ramp)
- scalar (`i1`; index of the start of the ramp)
- scalar (`i2`; index of the end of the ramp)
- scalar (`i3`; index of the end of the second flat part)

OUTPUT:

- list: scalar (level of the first flat part) , scalar (level of the second flat part)

COMMENT: correct?

rampfit computes optimal ramp parameters (in terms of the MSE) for a given data vector (see Mudelsee's paper: "Ramp function regression: a tool for quantifying climate transitions" (2000)) and carries out a robustness test on surrogate data

USES: rampfit_xt, rampfit.val, lopt, blocksample
IS USED BY: -

INPUT:

- vector (`x`; data)
- scalar (`i1_min`; lower index limit of the start of the ramp)
- scalar (`i1_max`; upper index limit of the start of the ramp)
- scalar (`i2_min`; lower index limit of the end of the ramp)
- scalar (`i2_max`; upper index limit of the end of the ramp)
- scalar (`tc1`; ? width of the first flat part (in time units))
- scalar (`tc2`; ? width of the second flat part (in time units))

⋆ OPTIONAL INPUT:

- vector (`t=seq(x)`; time)
- vector (`sigma2=rep(1,length(x))`; uncertainty (variance) for every point in time)
- scalar (`N.R=10`; number of surrogates)

OUTPUT:

- list: rampfit_xt output for data, rampfit_xt output for surrogates

COMMENT: description incomplete: what is meant by "width of the flat part"?

rampfit.val computes the residuals of a ramp fit for given data and parameters and plots if desired(see Mudelsee's paper: "Ramp function regression: a tool for quantifying climate transitions" (2000))

USES: ramp
IS USED BY: rampfit

INPUT:

- vector (`x`; data)
- vector (`t`; time)
- list (`param`; scalar (`i0`; index of the start of flat part of the ramp) , scalar (`i1`; index of the start of the ramp) , scalar (`i2`; index of the end of the ramp) , scalar (`i3`; index of the end of the second flat part) , scalar (`x1`; level of the start of the ramp) , scalar (`x2`; level of the end of the ramp))
- boolean (`bPlot=F`; plots if `bPlot`=T)

OUTPUT:

- list (vector (residuals) , vector (ramp output))
- (plot)

COMMENT: correct?

rampfit_xt computes optimal ramp parameters (in terms of the MSE) for a given data vector (see Mudelsee's paper: "Ramp function regression: a tool for quantifying climate transitions" (2000))

USES: ramp_xfit, ramp
IS USED BY: rampfit

INPUT:

- vector (`x`; data)
- vector (`t`; time)
- scalar (`i1_min`; lower index limit of the start of the ramp)
- scalar (`i1_max`; upper index limit of the start of the ramp)
- scalar (`i2_min`; lower index limit of the end of the ramp)
- scalar (`i2_max`; upper index limit of the end of the ramp)
- scalar (`tc1`; ? width of the first flat part (in time units))
- scalar (`tc2`; ? width of the second flat part (in time units))
- vector (`sigma2`; uncertainty (variance) for every point in time)

OUTPUT:

- list: Ramp parameters as scalars (see Mudelsee's paper)

COMMENT: description incomplete: what is meant by "width of the flat part"?

rbow generates a rainbow color palette (is exactly the same function as the R function rainbow)

USES: -
IS USED BY: plotmap.pField, plotmap.square, plotwind, plotmap.pFieldb,

INPUT:

- ... (see R help for rainbow)

OUTPUT:

- vector of character strings (colors)

COMMENT: correct? why is there an extra definition of the rainbow function?

rbow.col.nonsigarea generates a specific rainbow color palette adjusted to the plot.sig function (similar to rbow and the R function rainbow)

USES: -
IS USED BY: plot.sig

INPUT:

- ... (see R help for rainbow)

OUTPUT:

- vector of character strings (colors)

COMMENT: correct?

read.accum reads in accumulation data (which is for instance stored in .txt format) and outputs a pTs time series object

USES: pTs
IS USED BY: -

INPUT:

- character string (`filename`; path of the file)
- character string (`name`; name of the data set)
- scalar (`lat`; latitude)
- scalar (`lon`; longitude)

OUTPUT:

- pTs object (data)

COMMENT: correct?

read.angsmalik ? reads in Angsmallik (?) data set

USES: pTs
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)

OUTPUT:

- pTs object (data)

COMMENT: description incomplete

read.clim ? reads in climate (?) data set

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)
- character string (`varname`; name of the variable of interest)

⋆ OPTIONAL INPUT:

- character string (`lonname="lon"`; name of longitude variable)

OUTPUT:

- pField object

COMMENT: description incomplete

read.had.annual ? reads in HAD (?) data set and outputs a pField object with annual resolution

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)

⋆ OPTIONAL INPUT:

- character string (`varname="temp"`; name of the variable of interest)
- character string (`latname="latitude"`; name of the latitude variable)
- character string (`lonname="longitude`; name of longitude variable)

OUTPUT:

- pField object

COMMENT: description incomplete

read.had.monthly ? reads in HAD (?) data set and outputs a pField object with monthly resolution

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)

⋆ OPTIONAL INPUT:

- character string (`varname="temp"`; name of the variable of interest)

OUTPUT:

- pField object

COMMENT: description incomplete

read.ice.had.annual ? reads in ice HAD (?) data set and outputs a pField object with annual resolution

USES: pField

IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)
- character string (`varname=""`; name of the variable of interest)

OUTPUT:

- pField object

COMMENT: description incomplete

read.ice.had.sd ? reads in ice HAD (?) data set and outputs a pField object

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)
- character string (`varname=""`; name of the variable of interest)

OUTPUT:

- pField object

COMMENT: description incomplete

read.mld.kara ? reads in Kara (?) data set and outputs a pField object

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)

⋆ OPTIONAL INPUT:

- character string (`varname="MIXED_LAYER_DEPTH"`; name of the variable of interest)
- character string (`name=""`; name of the data set)

OUTPUT:

- pField object

COMMENT: description incomplete

read.mon.ecmwf ? reads in a geopotential height data set from the www.ecmwf.int/ (European Centre for Medium-Range Weather Forecasts) and outputs a pField object with monthly resolution

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)

⋆ OPTIONAL INPUT:

- character string (`varname="z"`; name of the variable of interest)

OUTPUT:

- pField object

COMMENT: description incomplete

read.rodgers.nao ? reads in a NAO (North Atlantic Oscillations) data set and outputs a pField object

USES: pTs
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)

OUTPUT:

- pTs object

COMMENT: description incomplete

read_dailyinsol ? reads in an insolation data set

USES: -
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)

OUTPUT:

- matrix

COMMENT: description incomplete

read_data reads in data from ncdf (netcdf) file (universal tool)

USES: find.var, julday.own
IS USED BY: millenium.anomalies, millenium.temperatures

INPUT:

- character string (`FILENAME`; path of the file)

⋆ OPTIONAL INPUT:

- character string (`varname=NULL`; name of the variable of interest)
- character string (`name=""`; name of the data set)
- character string (`lonname=NULL`; name of the longitude variable)
- character string (`latname=NULL`; name of the latitude variable)
- vector (`missVal=c(-1e+20,1e+20)`; assigns NA values to data values that exceed some specified boundaries)

OUTPUT:

- pField object

COMMENT: correct?

read_er ? reads in ncdf file (similar to read_er.clim) and outputs a pField object

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)
- character string (`varname`; name of the variable of interest)

⋆ OPTIONAL INPUT:

- character string (`name=""`; name of the data set)

OUTPUT:

- pField object

COMMENT: description incomplete; use unclear

read_er.clim ? reads in ncdf file (similar to read_er) and outputs a pField object

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)
- character string (`varname`; name of the variable of interest)

⋆ OPTIONAL INPUT:

- character string (`name=""`; name of the data set)

OUTPUT:

- pField object

COMMENT: description incomplete; use unclear

read_gasforcing ? reads in gas forcing (?) data set and outputs a pTs object

USES: pTs
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)

OUTPUT:

- pTs object

COMMENT: description incomplete

read_ipcc.mon ? reads in NCEP (National Centers for Environmental Prediction) data set(s) (?) and outputs a pField object with monthly resolution (similar to read_ncep.mon

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)

⋆ OPTIONAL INPUT:

- character string (`varname="slp"`; name of the variable of interest)
- scalar (`sy=1850`; ?)

OUTPUT:

- pField object

COMMENT: description incomplete; what does the variable sy stand for?

read_kaplan ? reads in Kaplan data set (e.g., SST (Sea Surface Temperature) data) and outputs a pField object

USES: pField
IS USED BY: -

INPUT:

· character string (`FILENAME`; path of the file)

⋆ OPTIONAL INPUT:

· character string (`varname="ssta"`; name of the variable)

OUTPUT:

· pField object

COMMENT: description incomplete

read_kaplan_monthly ? reads in Kaplan data set (e.g., SST (Sea Surface Temperature) data) and returns a pField object with monthly resolution

USES: pField
IS USED BY: -

INPUT:

· character string (`FILENAME`; path of the file)

⋆ OPTIONAL INPUT:

· character string (`varname="sst"`; name of the variable of interest)

OUTPUT:

· pField object

COMMENT: description incomplete

read_mann ? reads in Mann SOI (Southern Oscillation Index) data set and returns a pTs object

USES: pTs
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)

⋆ OPTIONAL INPUT:

- scalar (`index=2`; ?)

OUTPUT:

- pTs object

COMMENT: description incomplete

read_ncep ? reads in NCEP (National Centers for Environmental Prediction) data set (?) and outputs a pField object with monthly resolution (similar to read_ncep.clim

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)
- character string (`varname`; name of the variable of interest)

⋆ OPTIONAL INPUT:

- character string (`name=""`; name of the data set)

OUTPUT:

- pField object

COMMENT: description incomplete; what is the difference between read_ncep and read_ncep.clim?

read_ncep.clim ? reads in NCEP (National Centers for Environmental Prediction) data set (?) and outputs a pField object with monthly resolution (similar to read_ncep

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)

⋆ OPTIONAL INPUT:

- character string (`varname="slp"`; name of the variable of interest)
- character string (`name=""`; name of the data set)
- character string (`lonname="lon"`; name of the longitude variable)
- character string (`latname="lat"`; name of the latitude variable)

OUTPUT:

- pField object

COMMENT: description incomplete; what is the difference between read_ncep.clim and read_ncep?

read_ncep.clim.day ? reads in NCEP (National Centers for Environmental Prediction) data set (?) and outputs a pField object with daily resolution (similar to read_ncep_day

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)
- character string (`varname`; name of the variable)

OUTPUT:

- pField object

COMMENT: description incomplete; what is the difference between read_ncep.clim.day and read_ncep_day?

read_ncep.mon ? reads in NCEP (National Centers for Environmental Prediction) data set (?) and outputs a pField object with daily resolution (similar to read_ipcc.mon

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)

⋆ OPTIONAL INPUT:

- character string (`varname="slp"`; name of the variable of interest)

OUTPUT:

- pField object

COMMENT: description incomplete

read_ncep_day ? reads in NCEP (National Centers for Environmental Prediction) data set (?) and outputs a pField object with daily resolution (similar to read_ncep.clim

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)
- character string (`varname`; name of the variable)

OUTPUT:

- pField object

COMMENT: description incomplete; what is the difference between read_ncep.clim.day and read_ncep_day?

read_ncep_yr ? reads in NCEP (National Centers for Environmental Prediction) data set (?) and outputs a pField object with annual resolution

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)
- character string (`varname`; name of the variable)

OUTPUT:

- pField object

COMMENT: description incomplete

read_one ? reads in ncdf (netcdf) file and outputs a pField object

USES: pField
IS USED BY:

INPUT:

- character string (`FILENAME`; path of the file)

⋆ OPTIONAL INPUT:

- character string (`varname="slp"`; name of the variable of interest)
- character string (`name=""`; name of the data set)

OUTPUT:

- pField object

COMMENT: description incomplete

read_precipdata ? reads in a precipitation data set (from a .txt file) and returns a pTs object

USES: pTs
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)

⋆ OPTIONAL INPUT:

- scalar (`skip=20`; lines to be skipped in the .txt file)
- scalar (`lat=69.21`; latitude associated with the data)
- scalar (`lon=308.9`; longitude associated with the data)
- character string (`name="prec Illuisat"`; name of the data set)
- boolean (`na.pad=TRUE`; decides if NA data values are set to zero (`na.pad`=TRUE, the default))

OUTPUT:

- pTs object

COMMENT: description incomplete

read_solarforcing ? reads in a solar forcing (?) data set and returns a pTs object

USES: pTs
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the file)

OUTPUT:

- pTs object

COMMENT: description incomplete

read_sonne ? reads in two data sets: Sonne data (?) and Martin Stendel's model output (?)

USES: pField
IS USED BY: -

INPUT:

- character string (`FILENAME`; path of the sonne data set file ?)
- character string (`GRIDNAME`; path of the model output file ?)

⋆ OPTIONAL INPUT:

- character string (`varname="var139"`; name of the variable of interest)
- character string (`name=NULL`; name of the pField object to be generated; if `name`=NULL, `FILENAME` is returned)

OUTPUT:

- pField object

COMMENT: description incomplete

red simulates an AR(1) process using the R routine arima.sim

USES: -
IS USED BY: -

INPUT:

- scalar (`a1`; AR(1) coefficient)
- scalar (`n`; sample size)

OUTPUT:

- vector (simulated AR(1) realizations)

COMMENT: correct?

regional.ts processes a set of pField objects: selects a region and applies areamean and the R tool rollmean

USES: selspace, areamean
IS USED BY: eval.region

INPUT:

- list (`pField.list`; pField objects)
- scalar (`lat1`; latitude no.1 for selspace)
- scalar (`lat2`; latitude no.2 for selspace)
- scalar (`lon1`; longitude no.1 for selspace)
- scalar (`lon2`; longitude no.2 for selspace)
- scalar (`mean.window`; width of the rolling mean window, contrary to medsmooth interpreted as total window size)

OUTPUT:

- list: pField objects

COMMENT: correct?

rm_season.pTs ? removes seasonal mean (?) from a pTs object (with the entire seascycle.pTs function implemented)

USES: -
IS USED BY: -

INPUT:

- ts object / pTs object (`ts`)

OUTPUT:

- ts object / pTs object

COMMENT: description incomplete

rmse calculates the RMSE (Root Mean Squared Error) between two data vectors

USES: -
IS USED BY: cost

INPUT:

- vector (`data1`; data)
- vector (`data2`; data)

OUTPUT:

- scalar (RMSE)

COMMENT: correct?

roll.1 checks if an object has "pTs" or "pField" format and applies roll.1ts or roll.1field, respectively (e.g., to derive a rolling mean)

USES: roll.1ts, roll.1field
IS USED BY: -

INPUT:

- pTs / pField object (`data`; data)
- scalar (`width`; width of the rolling time window the function is applied to, e.g., for calculating a rolling mean)
- function (`FUN`; e.g., `FUN`=mean when a rolling mean shall be computed)

⋆ OPTIONAL INPUT:

- scalar (`by=1`; distance between the start points of the rolling intervals)
- character string (`name=NULL`; name of the resulting pTs/pField object; if `name`=NULL, `FILENAME` is returned)
- boolean (`detrend=F`; if `detrend`=T, the data are detrended before the function is applied)
- boolean (`scale=F`; if `scale`=T, the data are scaled (see R help for scale) before the function is applied)
- ...

OUTPUT:

- pTs / pField object (processed data)

COMMENT: correct?

roll.1field applies a rolling function (e.g., mean) to each time series of a pField object

USES: getname, pField, gethistory, detrend, addhistory
IS USED BY: roll.1

INPUT:

- pField object (`field`; data)
- scalar (`width`; width of the rolling time window the function is applied to, e.g., for calculating a rolling mean)
- function (`FUN`; e.g., `FUN`=mean when a rolling mean shall be computed)

⋆ OPTIONAL INPUT:

- scalar (`by=1`; distance between the start points of the rolling intervals)
- character string (`name=NULL`; name of the resulting pField object; if `name`=NULL, `FILENAME` is returned)
- boolean (`detrend=T`; if `detrend`=T, the data are detrended before the function is applied)
- boolean (`scale=T`; if `scale`=T, the data are scaled (see R help for scale) before the function is applied)
- ...

OUTPUT:

- pField object (processed data)

COMMENT: correct?

roll.1ts applies a rolling function (e.g., mean) to a pTs time series object

USES: getname, pTs, gethistory, detrend, addhistory
IS USED BY: roll.1

INPUT:

- pTs object (`ts1`; data)
- scalar (`width`; width of the rolling time window the function is applied to, e.g., for calculating a rolling mean)
- function (`FUN`; e.g., `FUN`=mean when a rolling mean shall be computed)

⋆ OPTIONAL INPUT:

- scalar (`by=1`; distance between the start points of the rolling intervals)
- character string (`name=NULL`; name of the resulting pTs object; if `name`=NULL, `FILENAME` is returned)
- boolean (`detrend=T`; if `detrend`=T, the data are detrended before the function is applied)
- boolean (`scale=T`; if `scale`=T, the data are scaled (see R help for scale) before the function is applied)
- ...

OUTPUT:

- pTs object (processed data)

COMMENT: correct?

roll.2 checks if two objects have "pTs" and "pField" or "pTs" and "pTs" format and applies roll.2ts or roll.2field, respectively (e.g., to derive a rolling correlation)

USES: roll.2ts, roll.2field
IS USED BY: change.cor.test

INPUT:

- pTs object (`data1`)
- pTs / pField object (`data2`)

- scalar (`width`; width of the rolling time window the function is applied to, e.g., for calculating a rolling correlation)
- function (`FUN`; e.g., `FUN`=cor when a rolling correlation shall be computed)

⋆ OPTIONAL INPUT:

- scalar (`by=1`; distance between the start points of the rolling intervals)
- character string (`name=NULL`; name of the resulting pTs/pField object; if `name`=NULL, `FILENAME` is returned)
- boolean (`detrend=F`; if `detrend`=T, the data are detrended before the function is applied)
- boolean (`scale=F`; if `scale`=T, the data are scaled (see R help for scale) before the function is applied)
- ...

OUTPUT:

- pTs / pField object

COMMENT: correct?

roll.2field applies a rolling function (e.g., correlation) to a pTs object and each time series of a pField object

USES: getname, pField, gethistory, detrend, addhistory
IS USED BY: roll.2

INPUT:

- pField object (`field`)
- pTs object (`ts`)
- scalar (width of the rolling time window the function is applied to, e.g., for calculating a rolling correlation)
- function (e.g., `FUN`=cor when a rolling correlation shall be computed)

⋆ OPTIONAL INPUT:

- scalar (`by=1`; distance between the start points of the rolling intervals)

- character string (`name=NULL`; name of the resulting pTs/pField object; if `name`=NULL, `FILENAME` is returned)
- boolean (`detrend=F`; if `detrend`=T, the data are detrended before the function is applied)
- boolean (`scale=F`; if `scale`=T, the data are scaled (see R help for scale) before the function is applied)
- ...

OUTPUT:

- pField object

COMMENT: correct?

roll.2ts applies a rolling function (e.g., correlation) to a pTs time series object and another pTs time series object

USES: getname, pTs, gethistory, detrend, addhistory
IS USED BY: roll.2

INPUT:

- pTs object (`ts1`)
- pTs object (`ts2`)
- scalar (width of the rolling time window the function is applied to, e.g., for calculating a rolling correlation)
- function (e.g., `FUN`=cor when a rolling correlation shall be computed)

⋆ OPTIONAL INPUT:

- scalar (`by=1`; distance between the start points of the rolling intervals)
- character string (`name=NULL`; name of the resulting pTs/pField object; if `name`=NULL, `FILENAME` is returned)
- boolean (`detrend=F`; if `detrend`=T, the data are detrended before the function is applied)
- boolean (`scale=F`; if `scale`=T, the data are scaled (see R help for scale) before the function is applied)
- ...

OUTPUT: pTs object

COMMENT: correct?

roll.plot plots for each time step the roll.1field / roll.2field output

USES: plot.pField (implicitly)
IS USED BY: -

INPUT:

- pField object (`field`; data processed with roll.1field or roll.2field)

⋆ OPTIONAL INPUT:

- scalar (`width=0`; width of the rolling time window the function was applied to in roll.1field / roll.2field)

OUTPUT:

- multiple plots

COMMENT: correct?

rollmean.k computes a running mean for a pTs time series object

USES: pTs, getlat, getlon, getname
IS USED BY: -

INPUT:

- pTs object (`x`; data)
- scalar (`k`; width of the rolling mean time window; also: distance between the start points of the rolling intervals)

⋆ OPTIONAL INPUT:

- boolean (`na.pad=FALSE`; decides if NA data values are set to zero (`na.pad`=TRUE))
- ...

OUTPUT:

- pTs object (processed data)

COMMENT: correct? where is the rollmean.default function defined?

rollmean.pField ? computes a running mean for a pField object (but is the same function as rollmean.pTs)

USES: addhistory
IS USED BY: -

INPUT:

- pField object (`x`; data)
- scalar (`k`; width of the rolling mean time window)

⋆ OPTIONAL INPUT:

- boolean (`na.pad=FALSE`; decides if NA data values are set to zero (`na.pad`=TRUE))
- ...

OUTPUT:

- ? pField object (processed data)

COMMENT: description incomplete; is the same function as rollmean.pTs (is that correct?)

rollmean.pTs computes a running mean for a pTs time series object (similar to rollmean.k, but with the distance between the start points of the rolling intervals set to 1)

USES: addhistory
IS USED BY: -

INPUT:

- pTs object (`x`; data)
- scalar (`k`; width of the rolling mean time window)

⋆ OPTIONAL INPUT:

- boolean (`na.pad=FALSE`; decides if NA data values are set to zero (`na.pad`=TRUE))
- ...

OUTPUT:

- pTs object (processed data)

COMMENT: correct? Note that there is another version of rollmean.pTs in the paleoLibrary

rval.cor derives the correlation coefficient from the two-sided p-value and the sample size

USES: -
IS USED BY: -

INPUT:

- scalar (`p`; two-sided p-value)
- scalar (`n`; sample size)

OUTPUT:

- scalar (correlation coefficient)

COMMENT: correct?

sb determines the common time window of two pTs/pField objects and outputs the respective sections

USES: -
IS USED BY: cor.sb

INPUT:

- pTs / pField object (`a`)
- pTs / pField object (`b`)

⋆ OPTIONAL INPUT:

- boolean (`debug=FALSE`; debugging variable)

OUTPUT:

- list: pTs / pField object , pTs / pField object

COMMENT: correct?

scale.pField scales a pField object with the R tool scale

USES: -
IS USED BY: index.nino3.4, index.nao

INPUT:

- pField object (`x`; data)

⋆ OPTIONAL INPUT:

- boolean /vector (`center=TRUE`; see R help for scale)
- boolean /vector (`scale=TRUE`; see R help for scale)

OUTPUT:

- pField object (scaled/centered data)

COMMENT: correct?

scale.pTs scales a pTs time series object with the R tool scale

USES: gethistory, pTs, getlat, getlon, addhistory
IS USED BY: -

INPUT:

- pTs object (`x`; data)

⋆ OPTIONAL INPUT:

- boolean /vector (`center=TRUE`; see R help for scale)
- boolean /vector (`scale=TRUE`; see R help for scale)

OUTPUT:

- pTs object (scaled/centered data)

COMMENT: correct?

scale_space ? scales a pField object (?)

USES: -
IS USED BY: -

INPUT:

- ? pField object (`data`; data)

OUTPUT:

- ? pField object (scaled/centered data)

COMMENT: description incomplete; if scale_space is for pField objects: What is the difference between this function and scale.pField?

schwerpunkt calculates the barycenter of a 2D pField object using the R tool weighted.mean

USES: latlonField, weighted.mean,
IS USED BY: -

INPUT:

- 2D pField object (`data`; data)

OUTPUT:

- list: scalar (latitude of barycenter) , scalar (longitude of barycenter)

COMMENT: correct?

seascycle.pTs ? does not work, problem is with cbind.ts

USES: -
IS USED BY: -

INPUT:

- ts object / pTs object (`seascycle`

OUTPUT: ?

COMMENT: description incomplete; does not work, problem is with cbind.ts

season.pTs takes a pTs/pField object whose resolution is higher than annual and outputs seasonal averages (e.g., December/January/February, annual resolution)

USES: pField, pTs, getlat, getlon, getname
IS USED BY: -

INPUT:

- pTs / pField object (`ts`)

⋆ OPTIONAL INPUT:

- character string (???)/ vector (`timewindow=c(1,11)/12`; defines the season to be considered: the default is `timewindow`=c(1,11)/12, focussing on the time span from January (1/12) to November (11/12))
- scalar (`TOL=1/350`; ?)
- boolean (`debug=FALSE`; debugging variable)

OUTPUT:

- pTs / pField object (with annual resolution)

COMMENT: description incomplete; is a character string allowed for the timewindow input?

selspace takes a pField object and outputs a section of interest (specified by latitude and longitude values; if only one latitude and one longitude are given, the output is a pTs object (otherwise a pField object))

USES: pTs, pField, getname, gethistory, addhistory
IS USED BY: e.g., index.nino3, index.nao, index.pna, pcor, index.soi

INPUT:

- pField object (`data`)
- scalar (`lat1`; latitude no.1)
- scalar (`lon1`; longitude no.1)

⋆ OPTIONAL INPUT:

- scalar (`lat2=NULL`; latitude no.2)
- scalar (`lon2=NULL`; longitude no.2)
- scalar (`tolLat=NULL`; ?)
- scalar (`tolLon=NULL`; ?)

OUTPUT:

- pTs / pField object (data section of interest)

COMMENT: description incomplete; what exactly are the tolerance values for?

selspace.interpolate interpolates a 2D pField object to a given point (if the adjancents points are missing, the nearest neighbour is returned; otherwise, bilinear interpolation is performed using the linearInterpp function from the R package fUtilities

USES: -
IS USED BY: -

INPUT:

- 2D pField object (`data`)
- scalar (`lat1`; latitude of interest)
- scalar (`lon1`; longitude of interest)

⋆ OPTIONAL INPUT:

- scalar (`SBOX=5`; ? if one of the lat/lon neighbours is NA, the area around the lat/lon pair is extended, an extension controlled by the `SBOX` variable)

OUTPUT:

- scalar ((interpolated) data value)

COMMENT: description incomplete; characterization of SBOX variable not precise enough

sigcor carries out a correlation test between a pTs and a pField object using the cortest.pTs function and sets non-significant correlation values to NA

USES: cortest.pTs
IS USED BY: -

INPUT:

- pTs object (`ts`)
- pField object (`data`)

⋆ OPTIONAL INPUT:

- scalar (`p=0.05`; p-value)

OUTPUT:

- pField object (correlation coefficients with non-significant values set to NA)

COMMENT: correct?

sigline.preparation computes contour lines for a pField object using the R tool contourLines; serves for instance to highlight areas with significant correlation coefficients (requiring only one contour level in order so separate correlation values with p-values >0.05 from correlation values with p-values $<=0.05$)

USES: plot.preparation

IS USED BY: plot.sig

INPUT:

- 2D pField object (`sigmap`; e.g., p-values corresponding to a correlation field)
- scalar / vector (`levels`; contour levels, e.g., `levels`=1-p_val)

OUTPUT:

- list: see R help for contourLines

COMMENT: correct?

sim.coh simulates two vectors/time series with the same coherence (see Huybers (2008))

USES: -
IS USED BY: -

INPUT:

- scalar (`cb`; coherence, has to be a scalar or a vector with half the length of `N` (the time series length) containing the coherences corresponding to the frequencies)

⋆ OPTIONAL INPUT:

- scalar (`N=1000`; length of the vector/time series)

OUTPUT:

- matrix (with one vector/time series in each column)

COMMENT: correct?

slp.diff.contour adds ??? sea level pressure (?) contour lines to a 2D field plot (using the R tool contour), so that positive correlations are visualized by solid and neg- ative correlations by dashed lines

USES: -
IS USED BY: -

INPUT:

- vector (`lon`; longitudes)
- vector (`lat`; latitudes)
- matrix (`data`; 2D ??? field)

OUTPUT:

- inserted into the current plot

COMMENT: description incomplete

slpcontour adds ??? sea level pressure (?) contour lines to a 2D field plot (using the R tool contour)

USES: -
IS USED BY: -

INPUT:

- vector (`lon`; longitudes)
- vector (`lat`; latitudes) matrix (`data`; 2D ??? field)

OUTPUT:

- inserted into the current plot

COMMENT: description incomplete

smoothspec smooths a spectrum with the medsmooth function (considering the running median)

USES: medsmooth
IS USED BY: specConf

INPUT:

- object of class "spec" (`spec`; =list with `spec$spec` being a vector)
- scalar (`m`; defining the running mean window size, i.e., `m` adjacent elements are considered in either direction)

OUTPUT:

- object of class "spec" (=list with `spec$spec` updated)

COMMENT: correct?

snoise.pTs generates surrogate time series that have the same autoregressive coefficient(s) as some input time series (if not provided, the order of the model and the coefficient(s) themselves are estimated with the R function ar and the Akaike_information_criterion)

USES: pTs, getlat, getlon
IS USED BY: eval.region

INPUT:

- vector / ts object / pTs object (`ts`; input time series)

⋆ OPTIONAL INPUT:

- scalar / vector (`a1=NULL`; AR coefficients whose number defines the order of the AR surrogates model)
- scalar (`order.max=1`; if `a1`=NULL, the AR(p) process order p of the input time series has to be estimated; `order.max` imposes a constraint on the maximum order; the default is `order.max=1`, leading to an AR(1) model)

OUTPUT:

- pTs object (surrogate time series)

COMMENT: correct?

specConf ? computes and smoothes the spectrum of a vector/time series using the smoothspec function and the R tool spectrum and plots if desired ?

USES: smoothspec, ar1fit, specred
IS USED BY: -

INPUT:

- ? time series / pTs object (`x`; data input)

⋆ OPTIONAL INPUT:

- ...

OUTPUT: ?

COMMENT: description incomplete

specLocalConf ? computes the spectrum of a vector/time series using the R tool spectrum and plots if desired ?

USES: -
IS USED BY: -

INPUT:

- ? time series / pTs object (`x`; data input)

⋆ OPTIONAL INPUT:

- ...?

OUTPUT: ?

COMMENT: description incomplete

specred ? computes the theoretical spectrum of an AR(1) model ?

USES:
IS USED BY: specConf, cost

INPUT: ?

OUTPUT: ?

COMMENT: description incomplete

splot plots two time series in one graphics device (taking advantage of the R tool scale)

USES: -

IS USED BY: -

INPUT:

- ts object / pTs object (`x1`)
- ts object / pTs object (`x2`)

⋆ OPTIONAL INPUT:

- vector (`xlim=NULL`; x-axis limits; per default, the x-axis range of the first input object is used)
- character string (`col1="blue"`; color of the first input object)
- character string (`col1="red"`; color of the second input object)
- boolean (`zeroline=FALSE`; draws a horizontal line at ??? if `zeroline`=TRUE)
- character string (`xlab=""`; x-axis label)
- scalar (`lwd=2`; thickness of the plot lines)
- scalar (`scale=1`; scales the second input object)
- scalar (`expand=1`; scales the y-axis)
- character string (`ylab=""`; left y-axis label)
- character string (`ylab2=""`; does not work yet; right y-axis label)
- vector (`x1.at=NULL` ; positions of the y-axis tickmarks on the left hand side, the default standard marking)
- vector (`x2.at=NULL`; positions of the y-axis tickmarks on the right hand side, the default being standard marking)
- character string (`main=""`; plot title)
- scalar (`line2=NULL`; draws a horizontal line at the specified position (???))
- scalar (`padj=2.5`; for y-axis labeling, see R help for mtext)

OUTPUT:

- plot

COMMENT: description incomplete; see input variables zeroline and line2

SSA2 ? using the R function eigen

USES: pTs, getname

IS USED BY: -

INPUT:

- ? ts object / pTs object (`ts`; ?)

⋆ OPTIONAL INPUT:

- ...?

OUTPUT:

- pTs object (?)
- (plot)

COMMENT: description incomplete

start.own extracts the last time point from a time series/pTs object/pField

USES: -
IS USED BY: season.pTs

INPUT:

- ts object/pTs object /pField object (`ts`)

OUTPUT:

- scalar (start time)

COMMENT: correct?

summary.pField extracts and prints general information (e.g., name, history, time range, latitude/longitude range) from a pField object

USES: -
IS USED BY: -

INPUT:

- pField object (`x`)

OUTPUT:

- print output

COMMENT: correct?

summary.pTs extracts and prints general information (e.g., name, history, time range) from a pField object

USES: -
IS USED BY: -

INPUT:

- pTs object (`x`)

OUTPUT:

- print output

COMMENT: correct?

sur.cholesky generates surrogates that have the same autocovariance matrix as some input time series (based on a Cholesky decomposition; see Haam and Huybers, 2010)

USES: -
IS USED BY: -

INPUT:

- vector / ts object / pTs object (`ts_in`; data input)
- scalar (`N.R`; number of surrogates to be generated)

OUTPUT:

- matrix (one surrogate time series per column)

COMMENT: correct?

tlag ?

USES: -
IS USED BY: ins.dec21, ins.dec21.param

INPUT: ?

OUTPUT:

- vector (?)

COMMENT: description incomplete

unwrap corrects the radian phase angles in an array by adding multiples of $\pm 2\pi$ when absolute jumps between consecutive array elements are greater than π radians; based on this Matlab implementation

USES: -
IS USED BY: orbital_parameters

INPUT: ?

OUTPUT:

- vector (?)

COMMENT: description incomplete

zonalmean applies the latmean to a pField object and saves out corresponding time steps as well as latitudes

USES: latmean, getlat
IS USED BY: -

INPUT:

- pField object (`data`; data)

OUTPUT:

- list: matrix (one data value for each time step and latitude) , vector (latitudes) , vector (time steps)

COMMENT: correct?

Index

corcontour

cortest.pTs

cost

critval.cor

cwind

daily_insolation

daily_insolation_param

deming

detrend

detrend.default

detrend.pField

detrend.pTs

distinct.col

drunif

end.own

enoise.pTs

eval.region

fastcor.pTs

filled.contour.own

filter.pField

filter.pTs

filter.pTs1

find.var

first

fisher

get.a1

get.transfer

gethistory

getilat

getlat

getlon

getname

getS0

gphcontour

highpass

hovmoeller

icecontour

identifyPch

index.ao

index.pna

index.nao

index.nino.tni

index.nino1.2

index.nino3

index.nino4

index.nino4

index.soi

index.soimodel

ins.dec21

ins.dec21.param

ins.march21

is_pField

is_pTs

julday.own

last

latlonField

latmean

list2pTs

lmSlope.pTs

load_huascara

lopt

lowpass

makefilm

maxpoint

medsmooth

mergeattr

millenium.anomalies

millenium.aod

millenium.temperatures

millenium.tsi

minpoint

myccf

mycor.test

myfun1

myfun2

na.apply

ngt

normcontour0

normcontour1

Ops.pField

Ops.pTs

orbital_parameters

orbital_parameters_fast

paleo.symbols

paleo.symbols.temperature

par.uin

pcor

pField

PickN

plot.pField

plot.Polygon

plot.preparation

plot.pTs

plot.sig

plotcont.pField

ploteof

plotindex

plotmap

plotmap.pField

plotmap.pFieldb

plotmap.pTs

plotmap.square

plotmapc.pField

PlotMultiple

plotSig

plotsquare

plot.unicor

plotwind

pnacontour

prcompNA.pField

prcompO.pField

prcompO.pTs

pTs

pval.cor

quiver

ramp

ramp_xfit

rampfit

rampfit.val

rampfit_xt

rbow

rbow.col.nonsigarea

read.accum

read.angsmalik

read.clim

read.had.annual

read.had.monthly

read.ice.had.annual

read.ice.had.sd

read.mld.kara

read.mon.ecmwf

read.rodgers.nao

read_dailyinsol

read_data

read_er

read_er.clim

read_gasforcing

read_ipcc.mon

read_kaplan

read_kaplan_monthly

read_mann

read_ncep

read_ncep.clim

read_ncep.clim.day

read_ncep.mon

read_ncep_day

read_ncep_yr

read_one

read_precipdata

read_solarforcing

read_sonne

red

regional.ts

rm_season.pTs

rmse

roll.1

roll.1field

roll.1ts

roll.2

roll.2field

roll.2ts

roll.plot

rollmean.k

rollmean.pField

rollmean.pTs

rval.cor

sb

scale.pField

scale.pTs

scale_space

schwerpunkt

seascycle.pTs

season.pTs

selspace

selspace.interpolate

sigcor

sigline.preparation

sim.coh

slp.diff.contour

slpcontour

smoothspec

snoise.pTs

specConf

specLocalConf

specred

splot

SSA2

start.own

summary.pField

summary.pTs

sur.cholesky

tlag

unwrap

zonalmean