Package 'amber'

August 28, 2020

Type Package

Title Automated Model Benchmarking R Package

Version 1.0.3

Author Christian Seiler [cre, aut]

Maintainer Christian Seiler <christian.seiler@canada.ca>

Description Functions that quantify how well the Canadian Land Surface Scheme Including Biogeochemical Cycles (CLASSIC) reproduces land surface processes when compared against reference data. To summarize model performance across different statistical metrics, this package employs a skill score system that was originally developed by Collier et. al., (2018) <doi:10.1029/2018MS001354> .

License GPL-3

Encoding UTF-8

LazyData true

Depends R (>= 3.4.0)

Imports classInt, doParallel, foreach, Hmisc, latex2exp, ncdf4, parallel, raster, rgdal, rgeos, scico, sp, stats, utils, viridis, xtable

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

RoxygenNote 6.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2020-08-28 11:20:02 UTC

R topics documented:

`amber-package` *Overview of AMBER functions*

Description

The Canadian Land Surface Scheme Including Biogeochemical Cycles (CLASSIC) is the land surface component of the Canadian Earth System Model (CanESM) (Melton et al, 2020). The Automated Model Benchmarking R package (AMBER) evaluates the ability of CLASSIC to reproduce land surface processes by comparing model outputs against quasi-observational data sets derived from remote sensing products, eddy covariance flux tower measurements, and stream flow measurements. To summarize model performance across different statistical metrics AMBER employs a skill score system that was originally developed by the International Land Model Benchmarking (ILAMB) framework (Collier et al., 2018). AMBER was created to tailor the ILAMB skill score approach for CLASSIC model outputs. While AMBER was tested for CLASSIC only it may also work for other models.

amber functions

The functions provided by AMBER can be grouped into three categories:

- Functions that compute skill scores and other metrics for a single variable. This includes: `scores.fluxnet.csv`, `scores.fluxnet.nc`, `scores.fluxnet.site`, `scores.functional.response`, `scores.grid.notime`, `scores.grid.time`, `scores.runoff`, and `scores.site.notime`.
- Functions that visualize model output for a single variable. This includes: `plotEnsembleHovmoeller`, `plotEnsembleMean`, `plotGrid`, `plotHovmoeller`, `plotNcIrreg`, `plotNc`, `plotZonalMeans`, `plotZonalMeanStats`, `seasonalCycleIrreg`, `seasonalCycle`, `zonalMeanIrreg`, `zonalMean`, and `zonalMeanStats`.
- Functions that visualize summary statistics across multiple variables. This includes: `scores.compare.benchmarks`, `scores.compare.ensemble`, `scores.compare`, `scores.tables`, `scores.tables.tweak`, `correlationMatrixDiff`, `correlationMatrixFluxnet`, `correlationMatrix`, `globalSumsTable`, `metrics.compare`, `plotBars`, and `plotFluxnetStats`.

  The suffix *Irreg* indicates that this function may be applied to data that is on an irregular grid. In the case of CLASSIC, this applies to high-resolution simulations that are conducted for the Canadian domain. The string *ensemble* implies that the function is designed to handle multiple model runs.

Skill scores

The performance of a model is expressed through scores that range from zero to one, where increasing values imply better performance. These scores are usually computed in five steps:

- (i) computation of a statistical metric,
- (ii) nondimensionalization,
- (iii) conversion to unit interval,
- (iv) spatial integration, and
- (v) averaging scores computed from different statistical metrics.

The latter includes the bias, root-mean-square error, phase shift, inter-annual variability, and spatial distribution. The equations for computing the bias score (S_{bias}) are:

$(i)\ bias(\lambda,\phi) = \overline{v_{mod}}(\lambda,\phi) - \overline{v_{ref}}(\lambda,\phi)$

$(ii)\ \varepsilon_{bias} = |bias(\lambda,\phi)|/\sigma_{ref}(\lambda,\phi)$

$(iii)\ s_{bias}(\lambda,\phi) = e^{-\varepsilon_{bias}(\lambda,\phi)}$

$(iv)\ S_{bias} = \overline{\overline{s_{bias}}}$

The equations for computing the root mean square error score (S_{rmse}) are:

$(i)\ crmse(\lambda,\phi) = \sqrt{\frac{1}{t_f - t_0}\int_{t_0}^{t_f}[(v_{mod}(t,\lambda,\phi) - \overline{v_{mod}}(\lambda,\phi)) - (v_{ref}(t,\lambda,\phi) - \overline{v_{ref}}(\lambda,\phi))]^2 dt}$

$(ii)\ \varepsilon_{rmse}(\lambda,\phi) = crmse(\lambda,\phi)/\sigma_{ref}(\lambda,\phi)$

$(iii)\ s_{rmse}(\lambda,\phi) = e^{-\varepsilon_{rmse}(\lambda,\phi)}$

$(iv)\ S_{rmse} = \overline{\overline{s_{rmse}}}$

The equations for computing the phase score (S_{phase}) are:

$(i)\ \theta(\lambda,\phi) = \max(c_{mod}(t,\lambda,\phi)) - \max(c_{ref}(t,\lambda,\phi))$

(ii) not applicable, as units are consistent across all variables

(iii) $s_{phase}(\lambda,\phi) = \frac{1}{2}[1+\cos(\frac{2\pi\theta(\lambda,\phi)}{365})]$

(iv) $S_{phase} = \overline{\overline{s_{phase}}}$

The equations for computing the inter-annual variability score (S_{iav}) are:

(i) $iav_{ref}(\lambda,\phi) = \sqrt{\frac{1}{t_f - t_0}\int_{t_0}^{t_f}(v_{ref}(t,\lambda,\phi) - c_{ref}(t,\lambda,\phi))^2 dt}$

(i) $iav_{mod}(\lambda,\phi) = \sqrt{\frac{1}{t_f - t_0}\int_{t_0}^{t_f}(v_{mod}(t,\lambda,\phi) - c_{mod}(t,\lambda,\phi))^2 dt}$

(ii) $\varepsilon_{iav} = |(iav_{mod}(\lambda,\phi) - iav_{ref}(\lambda,\phi))|/iav_{ref}(\lambda,\phi)$

(iii) $s_{iav}(\lambda,\phi) = e^{-\varepsilon_{iav}(\lambda,\phi)}$

(iv) $S_{iav} = \overline{\overline{s_{iav}}}$

The equations for computing the spatial distribution score (S_{dist}) are:

(i) $\sigma = \sigma_{\overline{v_{mod}}}/\sigma_{\overline{v_{ref}}}$

(ii) and (iii) not applicable

(iv) $S_{dist} = 2(1+R)/(\sigma + \frac{1}{\sigma})^2$

where $\overline{v_{mod}}(\lambda,\phi)$ and $\overline{v_{ref}}(\lambda,\phi)$ are the mean values in time t of a variable v as a function of longitude λ and latitude ϕ for model and reference data, respectively, t_0 and t_f are the initial and final time step, $\sigma_{\overline{v_{mod}}}$ and $\sigma_{\overline{v_{ref}}}$ are the standard deviation of the time mean values from the model and reference data, and R is the spatial correlation coefficient of $\overline{v_{ref}}(\lambda,\phi)$ and $\overline{v_{mod}}(\lambda,\phi)$.

Score values are then combined to derive a single overall score for each output variable:

(v) $S_{overall} = \frac{S_{bias}+2S_{rmse}+S_{phase}+S_{iav}+S_{dist}}{1+2+1+1+1}$.

Note that S_{rmse} is weighted by a factor of two, which emphasizes its importance.

References

Collier, Nathan, Forrest M. Hoffman, David M. Lawrence, Gretchen Keppel-Aleks, Charles D. Koven, William J. Riley, Mingquan Mu, and James T. Randerson. 2018. "The International Land Model Benchmarking (ILAMB) System: Design, Theory, and Implementation." Journal of Advances in Modeling Earth Systems 10 (11): 2731–54.

Melton, Joe R., Vivek K. Arora, Eduard Wisernig-Cojoc, Christian Seiler, Matthew Fortier, Ed Chan, and Lina Teckentrup. 2020. "CLASSIC v1.0: The Open-Source Community Successor to the Canadian Land Surface Scheme (CLASS) and the Canadian Terrestrial Ecosystem Model (CTEM) - Part 1: Model Framework and Site-Level Performance." https://doi.org/10.5194/gmd-13-2825-2020.

`correlationMatrix` *Correlation matrix for statistical metrics computed by AMBER*

Description

This function produces a correlation matrices for mean values, $bias$, $rmse$, $crmse$, $phase$, and corresponding scores. The input data consist of netCDF files produced by scores.grid.time and scores.grid.notime.

Usage

```
correlationMatrix(metric, fileList = FALSE, myFiles = NA,
  myRows = NA, myColumns = NA, inputDir, outputDir = FALSE,
  significanceLevel = 0.01, plot.width = 8, plot.height = 6.8,
  plot.margin = c(10, 10, 1, 4))
```

Arguments

metric	A string that indicates for what statistical metric the correlation matrix should be computed. Options are 'mod-mean', ref-mean', bias', 'crmse', 'phase', 'bias-score', 'crmse-score', 'phase-score', or 'iav-score'.
fileList	Logical, FALSE by default. Should be set to TRUE if user defines a list of files that shall be included in the analysis. If set to FALSE, the function will search for all files whose names include the string of the metric defined above.
myFiles	Names of NetCDF files that shall be included in the analysis. Only relevant if fileList is TRUE.
myRows	Optional: the user can highlight relations between variables by specifying variable names along the rows and columns of a matrix. Those relations are then highlighted by plotting a polygon around the corresponding grid cells and by plotting the corresponding value.
myColumns	Optional: Same as myRows but for variable names listed along the columns of the matrix.
inputDir	A string that gives the location of NetCDF files produced by scores.grid.time, e.g. '/home/project/study'.
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
significanceLevel	A number that gives the desired significance level of a correlation, e.g. 0.01
plot.width	A number that gives the plot width, e.g. 8
plot.height	A number that gives the plot height, e.g. 6.8
plot.margin	An R object that gives the plot margin, e.g. c(10, 10, 1, 4)

Value

A list with the Spearman correlation coefficient and corresponding p-values, and a Figure of the correlation matrix

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
```

```
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

inputDir <- paste(system.file('extdata', package = 'amber'), 'zonalMeanStats', sep = '/')
correlationMatrix(metric = 'bias', inputDir = inputDir)

# Alternatively, use a list of file names
myFiles <- c('GPP-FluxCom-bias.nc', 'LAI-AVHRR-bias.nc', 'ALBS-CERES-bias.nc')

# You can specify certain relationships to highlight them in your correlation matrix
myRows <- c('LAI.AVHRR', 'ALBS.CERES')
myColumns <- c('GPP.FluxCom', 'GPP.FluxCom')

correlationMatrix(metric = 'bias', fileList = TRUE, myFiles = myFiles,
myRows = myRows, myColumns = myColumns, inputDir = inputDir)
 #donttest
```

`correlationMatrixDiff` *Plot a matrix that shows the difference between two correlation coefficient matrices computed by correlationMatrix.*

Description

This function plots a matrix that shows the difference between two correlation coefficient matrices computed by correlationMatrix. This is useful for assessing how well model data reproduces correlations that are evident in reference data. The difference is computed as the absolute value of the first correlation matrix minus the absolute value of the second correlation matrix.

Usage

```
correlationMatrixDiff(cm.one, cm.two, myRows = NA, myColumns = NA,
  inputDir, outputDir = FALSE, ofileName = "correlationMatrixDiff.pdf",
  plot.width = 8, plot.height = 6.8, plot.margin = c(10, 10, 1, 4))
```

Arguments

`cm.one`	An R objective that gives a correlation matrix computed by correlationMatrix.
`cm.two`	An R objective that gives another correlation matrix computed by correlationMatrix.

myRows	Optional: the user can highlight relations between variables by specifying variable names along the rows and columns of a matrix. Those relations are then highlighted by plotting a polygon around the corresponding grid cells and by plotting the corresponding value.
myColumns	Optional: Same as myRows but for variable names listed along the columns of the matrix.
inputDir	A string that gives the location of NetCDF files produced by scores.grid.time, e.g. '/home/project/study'.
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
ofileName	A string that gives the output file name, e.g. 'myOutput.pdf'
plot.width	A number that gives the plot width, e.g. 8
plot.height	A number that gives the plot height, e.g. 6.8
plot.margin	An R object that gives the plot margin, e.g. c(10, 10, 1, 4)

Value

A Figure of a matrix that shows the difference between two correlation coefficient matrices.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

inputDir <- paste(system.file('extdata', package = 'amber'), 'zonalMeanStats', sep = '/')
cm.one <- correlationMatrix(metric = 'bias', inputDir = inputDir)
cm.two <- correlationMatrix(metric = 'bias', inputDir = inputDir)

correlationMatrixDiff(cm.one, cm.two, inputDir = inputDir)

# You can specify certain relationships to highlight them in your correlation matrix
myRows <- c('LAI.AVHRR', 'LAI.AVHRR')
myColumns <- c('ALBS.CERES', 'GPP.FluxCom')
```

```
correlationMatrixDiff(cm.one, cm.two, inputDir = inputDir, myRows = myRows, myColumns = myColumns)
```

`correlationMatrixFluxnet` *Correlation matrix for statistical metrics computed by AMBER for FLUXNET data*

Description

This function produces a correlation matrices for mean values, $bias$, $crmse$, $phase$, and corresponding scores. The input data consist of text files produced by scores.fluxnet.csv.

Usage

```
correlationMatrixFluxnet(metric, inputDir, outputDir = FALSE,
  fileNames = c("GPP_FLUXNET", "HFLS_FLUXNET", "HFSS_FLUXNET",
  "NEE_FLUXNET", "RECO_FLUXNET", "RNS_FLUXNET"),
  significanceLevel = 0.01, plot.width = 8, plot.height = 6.8,
  plot.margin = c(10, 10, 1, 4))
```

Arguments

`metric`	A string that indicates for what statistical metric the correlation matrix should be computed. Options are 'mod.mean', ref.mean', bias', 'crmse', 'phase', 'bias.score', 'crmse.score', 'phase.score', or 'iav.score'.
`inputDir`	A string that gives the location of text files produced by scores.fluxnet.csv, e.g. '/home/project/study'.
`outputDir`	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
`fileNames`	An object of strings that give the filenames that should be included. The default is c('GPP_FLUXNET', 'HFLS_FLUXNET', 'HFSS_FLUXNET', 'NEE_FLUXNET', 'RECO_FLUXNET', 'RNS_FLUXNET')
`significanceLevel`	A number that gives the desired significance level of a correlation, e.g. 0.01
`plot.width`	A number that gives the plot width, e.g. 8
`plot.height`	A number that gives the plot height, e.g. 6.8
`plot.margin`	An R object that gives the plot margin, e.g. c(10, 10, 1, 4)

Value

A list with the Spearman correlation coefficient and corresponding p-values, and a Figure of the correlation matrix

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

inputDir <- paste(system.file('extdata', package = 'amber'), 'scores', sep = '/')
correlationMatrixFluxnet(metric = 'bias', inputDir)
```

---

globalSumsTable *Table of globally summed values and corresponding biases.*

Description

This function produces a table of globally summed values and corresponding biases when comparing model and reference data. The table provides the variable name, the reference data name, globally summed values from the model and reference data, the absolute bias, the relative bias, units, and time period. The inputs consist of netCDF files produced by scores.grid.time and scores.grid.notime. The global sums are based on grid cells that model and reference data have in common.

Usage

```
globalSumsTable(mod.path.list, modelIDs, variableNames, unitLaTeX,
  conversionFactor, outputDir = FALSE)
```

Arguments

mod.path.list	A list with paths for each model run, e.g. mod.path.list <- list(mod01.path, mod02.path, mod03.path).
modelIDs	An R object with the different model run IDs, e.g. c('CLASSIC.CRUJRAv2', 'CLASSIC.GSWP3W5E5', 'CLASSIC.CRUNCEP')
variableNames	Variable names, e.g. c('GPP', 'CSOIL', 'BURNT')
unitLaTeX	Desired output units in LaTeX format, e.g. c('PgC yr$^-1$', 'PgC', '10^6 km^2 yr^-1')

conversionFactor
: Factors that convert the unit of the input files into the desired output unit, e.g. c((10^(15) / 365.25)^(-1), 10^(-12), 12*10^(-14)). Examples of typical conversions include (i) carbon fluxes from gC/day to PgC/yr, (ii) carbon stocks from kgC to PgC, and fractional area burnt from percentage per month to 10^6 km2 per year. Conversion from gC/day to PgC/yr: 1 PgC/yr = 10^15 gC/yr = 10^15 / 365.25 gC/day. Taking the reciprocal leads to the conversion factor, i.e. (10^(15) / 365.25)^(-1). Conversion from kgC to PgC: 1 PgC = 10^15 gC = 10^12 kgC. The resulting conversion factor is 10^(-12). Conversion from percentage of grid cell per month to 10^6 km2 per year: 10^6 km2 per year = 10^12 m2 per year = 1/12 * 10^12 m2 per month. To account for the conversion from percentage to fraction, we need to multiply by 100. The resulting conversion factor is 12*10^(-14).

outputDir
: A string that gives the output directory, e.g. '/home/project/study'. The output table in LaTeX format will only be written if the user specifies an output directory.

Value

A table with globally summed values and corresponding biases.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

mod01.path <- paste(system.file('extdata', package = 'amber'), 'model01', sep = '/')
mod02.path <- paste(system.file('extdata', package = 'amber'), 'model02', sep = '/')

mod.path.list <- list(mod01.path, mod02.path)
modelIDs <- c('CLASSIC.CRUJRAv2', 'CLASSIC.GSWP3W5E5')

variableNames <- c('GPP', 'BURNT')
unitLaTeX <- c('PgC yr$^{-1}$', '10$^6$ km$^2$ yr^{-1}')
conversionFactor <- c((10^(15) / 365.25)^(-1), 12*10^(-14))
```

```
globalSumsTable(mod.path.list, modelIDs, variableNames, unitLaTeX, conversionFactor,
outputDir = FALSE)
```

`metrics.compare` *Compares global mean statistical meterics for multiple model runs*

Description

This function plots statistical meterics for multiple model runs. This is useful for comparing the impact of changes in model settings or input data.

Usage

```
metrics.compare(mod.path.list = mod.path.list, plot.width = 13,
  plot.height = 14, outputDir = FALSE, defineVariableOrder = TRUE,
  myVariables = myVariables)
```

Arguments

`mod.path.list`	A list with paths for each model run, e.g. mod.path.list <- list(mod01.path, mod02.path, mod03.path)
`plot.width`	Number that gives the plot width, e.g. 6
`plot.height`	Number that gives the plot height, e.g. 5
`outputDir`	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
`defineVariableOrder`	Logical. If TRUE, variables are sorted according to the parameter myVariables defined below. Default setting is FALSE.
`myVariables`	An R object with variable names of variables that should be included in table, e.g. c('GPP', 'RECO', 'NEE')

Value

A figure in PDF format that shows statistical metrics for multiple model runs.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
```

```
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

mod01.path <- system.file('extdata/SIMod01', package = 'amber')
mod02.path <- system.file('extdata/SIMod02', package = 'amber')

mod.path.list <- list(mod01.path, mod02.path)
myVariables <- c('RNS', 'RSS', 'RLS', 'ALBS', 'HFLS', 'HFSS', 'HFG', 'GPP',
'RECO', 'NEE', 'FIRE', 'AGB', 'CVEG', 'CSOIL', 'LAI', 'BURNT', 'SNW', 'MRSLL', 'MRRO')

myVariables <- c('ALBS', 'GPP', 'LAI')

metrics.compare(mod.path.list = mod.path.list, plot.width = 5, plot.height = 15,
outputDir = FALSE, defineVariableOrder = TRUE, myVariables = myVariables)
```

---

plotBars *Barplots of reference and model data with same unit*

Description

This function produces barplots of annual mean values for a selection of variables and reference data. The user may choose to plot both, absolute and fractional values. A typical application is a barplot for variables related to the surface energy balance.

Usage

```
plotBars(mod.path.list, modelIDs, variableNames, referenceNames,
  showFractions = FALSE, targetVariable = "RNS",
  ofileName = "barplot.pdf", plot.width = 12, plot.height = 8,
  outputDir = FALSE, subcaption = FALSE)
```

Arguments

mod.path.list	A list with paths for each model run, e.g. mod.path.list <- list(mod01.path, mod02.path, mod03.path). The respective folders must each contain the file 'metricsTable' produced by the function scores.tables.
modelIDs	An R object with the different model run IDs, e.g. c('CLASSIC.CRUJRAv2', 'CLASSIC.GSWP3W5E5', 'CLASSIC.CRUNCEP')
variableNames	Names of variables that should be plotted, e.g. c('HFG', 'HFLS', 'HFSS', 'RNS')

referenceNames	Names of reference data that should be plotted, e.g. c('FLUXNET', 'CLASS')
showFractions	Logical. If set to TRUE, the Figure will show fractional values, such as the fractions of net surface radiation that are converted into heat fluxes.
targetVariable	A string that gives the variable for which fractions are computed, e.g. 'RNS' for net surface radiation. Only relevant if showFractions = TRUE.
ofileName	A string that gives the output file name, e.g. 'barplot.pdf'
plot.width	Number that gives the plot width, e.g. 12
plot.height	Number that gives the plot height, e.g. 8
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
subcaption	Logical. If TRUE then the subcaptions (a) and (b) are added.

Value

Barplots of annual mean values annual mean values for a selection of variables and reference data.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

mod01.path <- system.file('extdata/SIMod01', package = 'amber')
mod02.path <- system.file('extdata/SIMod02', package = 'amber')

mod.path.list <- list(mod01.path, mod02.path)
modelIDs <- c('CLASSIC.CRUJRAv2', 'CLASSIC.GSWP3W5E5')
variableNames <- c('HFG', 'HFLS', 'HFSS', 'RNS')
referenceNames <- c('FLUXNET', 'CLASS')

plotBars(mod.path.list, modelIDs, variableNames, referenceNames,
showFractions = TRUE, targetVariable = 'RNS', ofileName = 'barplot.pdf',
plot.width = 5, plot.height = 5, outputDir = FALSE)
```

plotEnsembleHovmoeller

Hovmoeller Diagram for model ensemble

Description

This function plots Hovmoeller diagrams of monthly climatological mean values and biases computed by scores.grid.time for multiple ensemble members

Usage

```
plotEnsembleHovmoeller(mod.path.list = mod.path.list,
  modelIDs = modelIDs, myVariables = myVariables, myBin = 20,
  gridCellWidth = 2, my.ylim = c(-100, 100), plot.width = 8.4,
  plot.height = 5, outputDir = FALSE)
```

Arguments

mod.path.list	A List of directories where AMBER output is stored for different model runs, e.g. list(mod01.path, mod02.path, mod03.path)
modelIDs	An R object with the different model run IDs, e.g. c('CLASSIC.CRUJRAv2', 'CLASSIC.GSWP3W5E5', 'CLASSIC.CRUNCEP')
myVariables	An R object with the variable names of interest, e.g. c('GPP.FluxCom', 'RECO.FluxCom').
myBin	An integer number that defines the latitudinal range used for computing the zonal mean. For instance, a value of 10 implies that a zonal mean is computed for every 10 degrees latitude.
gridCellWidth	A number that is used as a factor to adjust the width of grid cells, e.g. 1.
my.ylim	An R object that gives the longitude range that you wish to plot, e.g. c(-90, 90)
plot.width	Number that gives the plot width, e.g. 8
plot.height	Number that gives the plot height, e.g. 4
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.

Value

Figures in PDF format.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
```

```
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

mod01.path <- paste(system.file('extdata', package = 'amber'), 'model01', sep = '/')
mod02.path <- paste(system.file('extdata', package = 'amber'), 'model02', sep = '/')
mod.path.list <- list(mod01.path, mod02.path)
modelIDs <- c('CLASSIC.CRUJRAv2', 'CLASSIC.GSWP3W5E5')

myVariables <- c('GPP-MODIS', 'GPP-GOSIF')

plotEnsembleHovmoeller(mod.path.list = mod.path.list,
modelIDs = modelIDs, myVariables = myVariables, myBin = 20, gridCellWidth = 2,
my.ylim = c(-100, 100), plot.width = 8.4, plot.height = 5.0)
 #donttest
```

`plotEnsembleMean` *Ensemble mean plots of AMBER results (bias, bias scores, etc)*

Description

This function plots ensemble mean, minimum, and maximum values of a statistical metric computed by scores.grid.time and scores.grid.notime.

Usage

```
plotEnsembleMean(long.name, metric, mod.path.list, modelIDs, myVariables,
  shp.filename = system.file("extdata/ne_110m_land/ne_110m_land.shp",
  package = "amber"), my.xlim = c(-180, 180), my.ylim = c(-60, 85),
  plot.width = 5, plot.height = 7, outputDir = FALSE,
  subcaption = "")
```

Arguments

`long.name`	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
`metric`	A string that specifies what statistical metric should be plotted. This includes for instance 'bias', 'crmse', 'phase', 'iav', 'bias-score', 'rmse-score', 'phase-score', and 'iav-score'.

mod.path.list	A List of directories where AMBER output is stored for different model runs, e.g. list(mod01.path, mod02.path, mod03.path)
modelIDs	An R object with the different model run IDs, e.g. c('CLASSIC.CRUJRAv2', 'CLASSIC.GSWP3W5E5', 'CLASSIC.CRUNCEP')
myVariables	An R object with the variable names of interest, e.g. c('GPP.FluxCom', 'RECO.FluxCom').
shp.filename	A string that gives the coastline shapefile
my.xlim	An R object that gives the longitude range that you wish to plot, e.g. c(-180, 180)
my.ylim	An R object that gives the longitude range that you wish to plot, e.g. c(-90, 90)
plot.width	Number that gives the plot width, e.g. 8
plot.height	Number that gives the plot height, e.g. 4
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
subcaption	A string that defines the subcaption of the figure, e.g. '(a)'.

Value

Figures in PDF format.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

long.name <- 'Gross Primary Productivity'
metric <- 'mod-mean'

mod01.path <- paste(system.file('extdata', package = 'amber'), 'model01', sep = '/')
mod02.path <- paste(system.file('extdata', package = 'amber'), 'model02', sep = '/')
mod.path.list <- list(mod01.path, mod02.path)
modelIDs <- c('CLASSIC.CRUJRAv2', 'CLASSIC.GSWP3W5E5')

myVariables <- c('GPP-GOSIF', 'GPP-MODIS')
```

```
plotEnsembleMean(long.name, metric, mod.path.list, modelIDs, myVariables,
 plot.width = 5, plot.height = 5.5)
```

`plotFluxnetStats` *Plots that show statistical metrics for FLUXNET sites*

Description

This function plots statistical metrics for the comparison against FLUXNET data.

Usage

```
plotFluxnetStats(inputDir, outputDir = FALSE, mod.id = "CLASSIC",
  variableNames = c("GPP", "RECO", "NEE", "RNS", "HFLS", "HFSS"),
  plot.width = 12, plot.height = 8)
```

Arguments

`inputDir`	A string that gives the location of text files produced by scores.fluxnet.csv, e.g. '/home/project/study'.
`outputDir`	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
`mod.id`	A string that identifies the source of the reference data set, e.g. 'CLASSIC'
`variableNames`	A string of six variables that should be plotted. Default is set to c('GPP', 'RECO', 'NEE', 'RNS', 'HFLS', 'HFSS').
`plot.width`	Number that gives the plot width, e.g. 12
`plot.height`	Number that gives the plot height, e.g. 8

Value

Figures that show statistical metrics produced by the function scores.fluxnet.csv. The function expects six input variables.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
```

```
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

inputDir <- paste(system.file('extdata', package = 'amber'), 'scores', sep = '/')
plotFluxnetStats(inputDir, outputDir = FALSE, mod.id = 'CLASSIC.CRUJRAv2')
```

`plotGrid` *Plots raster layers of a raster stack object*

Description

This function plots the results from scores.grid.time and scores.grid.notime.

Usage

```
plotGrid(long.name, plot.me, irregular = FALSE,
  my.projection = "+proj=longlat +ellps=WGS84",
  shp.filename = system.file("extdata/ne_110m_land/ne_110m_land.shp",
  package = "amber"), my.xlim = c(-180, 180), my.ylim = c(-60, 85),
  plot.width = 8, plot.height = 3.8, outputDir = FALSE)
```

Arguments

`long.name`	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
`plot.me`	A list that is produced by scores.grid.time or scores.grid.notime.
`irregular`	logical: TRUE if data is on an irregular grid and FALSE if data is on a regular grid
`my.projection`	A string that gives the projection of the irregular grid
`shp.filename`	A string that gives the coastline shapefile
`my.xlim`	An R object that gives the longitude range that you wish to plot, e.g. c(-180, 180)
`my.ylim`	An R object that gives the longitude range that you wish to plot, e.g. c(-90, 90)
`plot.width`	Number that gives the plot width, e.g. 8
`plot.height`	Number that gives the plot height, e.g. 4
`outputDir`	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.

Value

Figures in PDF format. This may include the model data (mean, $mod.mean$; standard deviation; interannual-variability, $mod.iav$; month of annual cycle maximum, $mod.max.month$), the reference data (mean, $ref.mean$; standard deviation; interannual-variability, $ref.iav$; month of annual cycle maximum, $ref.max.month$), statistical metrics (bias, $bias$; root mean square error, $rmse$; time difference of the annual cycle maximum, $phase$), and scores (bias score, $bias.score$; root mean square error score, $rmse.score$; inter-annual variability score $iav.score$; annual cycle score ($phase.score$).

Examples

```
# (1) Global plots on a regular grid

library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

long.name <- 'Gross primary productivity'
nc.mod <- system.file('extdata/modelRegular', 'gpp_monthly.nc', package = 'amber')
nc.ref <- system.file('extdata/referenceRegular', 'gpp_GBAF_128x64.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'GBAF' # give reference dataset a name
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
unit.conv.ref <- 86400*1000 # optional unit conversion for reference data
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)

# Short version using default settings:
plot.me <- scores.grid.time(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit)
plotGrid(long.name, plot.me)

plot.me <- scores.grid.time(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit, score.weights = c(1,2,1,1,1), outlier.factor = 1,
irregular = FALSE)
plotGrid(long.name, plot.me)
```

```
# (2) Regional plots on a rotated grid
long.name <- 'Gross primary productivity'
nc.mod <- system.file('extdata/modelRotated', 'gpp_monthly.nc', package = 'amber')
nc.ref <- system.file('extdata/referenceRotated', 'gpp_GBAF_rotated.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'GBAF' # give reference dataset a name
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
unit.conv.ref <- 86400*1000 # optional unit conversion for reference data
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)
my.projection <- '+proj=ob_tran +o_proj=longlat +o_lon_p=83. +o_lat_p=42.5 +lon_0=263.'

plot.me <- scores.grid.time(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit, score.weights = c(1,2,1,1,1), outlier.factor = 10,
irregular = TRUE, my.projection = my.projection)

# Plot results:
irregular <- TRUE # data is on an irregular grid
my.projection <- '+proj=ob_tran +o_proj=longlat +o_lon_p=83. +o_lat_p=42.5 +lon_0=263.'
# shp.filename <- system.file('extdata/ne_50m_admin_0_countries/ne_50m_admin_0_countries.shp',
# package = 'amber')
shp.filename <- system.file("extdata/ne_110m_land/ne_110m_land.shp", package = "amber")
my.xlim <- c(-171, 0) # longitude range that you wish to plot
my.ylim <- c(32, 78) # latitude range that you wish to plot
plot.width <- 7 # plot width
plot.height <- 3.8 # plot height

plotGrid(long.name, plot.me, irregular, my.projection,
shp.filename, my.xlim, my.ylim, plot.width, plot.height)
 #donttest
```

`plotHovmoeller` *Plot Hovmoeller diagrams that show monthly climatological mean values and biases*

Description

This function plots Hovmoeller diagrams of monthly climatological mean values and biases computed by scores.grid.time.

Usage

```
plotHovmoeller(plot.me, long.name, mod.id, ref.id, myBin = 20,
  gridCellWidth = 2, plot.width = 4, plot.height = 5.2,
  my.ylim = c(-100, 100), outputDir = FALSE)
```

Arguments

`plot.me`	A list that is produced by scores.grid.time

long.name	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
mod.id	A string that identifies the source of the reference data set, e.g. 'CLASSIC'
ref.id	A string that identifies the source of the reference data set, e.g. 'MODIS'
myBin	An integer number that defines the latitudinal range used for computing the zonal mean. For instance, a value of 10 implies that a zonal mean is computed for every 10 degrees latitude.
gridCellWidth	A number that is used as a factor to adjust the width of grid cells, e.g. 1.
plot.width	Number that gives the plot width, e.g. 8
plot.height	Number that gives the plot height, e.g. 8
my.ylim	An R object with the latitudinal range that should be plotted, e.g. c(-40, 65).
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.

Value

Figures in PDF format.

Examples

```
# Global plots on a regular grid

library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

long.name <- 'Gross primary productivity'
nc.mod <- system.file('extdata/modelRegular', 'gpp_monthly.nc', package = 'amber')
nc.ref <- system.file('extdata/referenceRegular', 'gpp_GBAF_128x64.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'GBAF' # give reference dataset a name
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
unit.conv.ref <- 86400*1000 # optional unit conversion for reference data
```

```
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)

# Short version using default settings:
plot.me <- scores.grid.time(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit)
plotHovmoeller(plot.me, long.name, mod.id, ref.id)

 #donttest
```

---

plotNc *Plots the time-mean of a variable stored in NetCDF model output on a regular grid*

Description

This function plots the time-mean, spatial-mean, zonal mean, and seasonal cycle of variable stored in NetCDF model output. The function expects model data to be on a regular grid.

Usage

```
plotNc(long.name, nc.mod, mod.id, unit.conv.mod, variable.unit, timePeriod,
  outlier.factor = 1000, my.xlim = c(-180, 180), my.ylim = c(-60,
  85), plot.width = 8, plot.height = 3.8, outputDir = FALSE)
```

Arguments

long.name	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
nc.mod	A string that gives the path and name of the netcdf file that contains the model output, e.g. '/home/model_gpp.nc'
mod.id	A string that identifies the source of the reference data set, e.g. 'CLASSIC'
unit.conv.mod	A number that is used as a factor to convert the unit of the model data, e.g. 86400
variable.unit	A string that gives the final units using LaTeX notation, e.g. 'gC m$^-2$ day$^-1$'
timePeriod	A string that gives the time period over which to average the data, e.g. c('1980-01', '2017-12')
outlier.factor	A number that is used to define outliers, e.g. 10. Plotting raster objects that contain extreme outliers lead to figures where most grid cells are presented by a single color since the color legend covers the entire range of values. To avoid this, the user may define outliers that will be masked out and marked with a red dot. Outliers are all values that exceed the interquartile range multiplied by the outlier factor defined here.
my.xlim	An R object that gives the longitude range that you wish to plot, e.g. c(-180, 180)

my.ylim	An R object that gives the longitude range that you wish to plot, e.g. c(-90, 90)
plot.width	Number that gives the plot width, e.g. 8
plot.height	Number that gives the plot height, e.g. 4
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory

Value

Figures in PDF format that show the time-mean, spatial-mean, zonal mean, and seasonal cycle.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

long.name <- 'Gross primary productivity'
nc.mod <- system.file('extdata/modelRegular', 'gpp_monthly.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)
timePeriod <- c('1980-01', '2017-12')
outlier.factor <- 1

plotNc(long.name, nc.mod, mod.id, unit.conv.mod, variable.unit, timePeriod, outlier.factor)
```

---

plotNcIrreg *Plots the time-mean of a variable stored in NetCDF model output on an irregular grid*

Description

This function plots the time-mean, spatial-mean, zonal mean, and seasonal cycle of variable stored in NetCDF model output. The function expects data to be on an irregular grid.

Usage

```
plotNcIrreg(long.name, nc.mod, mod.id, unit.conv.mod, variable.unit,
  timePeriod, outlier.factor = 1000,
 my.projection = "+proj=ob_tran +o_proj=longlat +o_lon_p=83. +o_lat_p=42.5 +lon_0=263.",
  my.xlim = c(-171, 0), my.ylim = c(32, 78), plot.width = 7,
  plot.height = 3.8, numCores = 2, timeInt = "month",
  outputDir = FALSE)
```

Arguments

long.name	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
nc.mod	A string that gives the path and name of the netcdf file that contains the model output, e.g. '/home/model_gpp.nc'
mod.id	A string that identifies the source of the reference data set, e.g. 'CLASSIC'
unit.conv.mod	A number that is used as a factor to convert the unit of the model data, e.g. 86400
variable.unit	A string that gives the final units using LaTeX notation, e.g. 'gC m$^-2$ day$^-1$'
timePeriod	A string that gies the time period over which to average the data, e.g. c('1980-01', '2017-12')
outlier.factor	A number that is used to define outliers, e.g. 10. Plotting raster objects that contain extreme outliers lead to figures where most grid cells are presented by a single color since the color legend covers the entire range of values. To avoid this, the user may define outliers that will be masked out and marked with a red dot. Outliers are all values that exceed the interquartile range multiplied by the outlier factor defined here.
my.projection	A string that defines the projection of the irregular grid
my.xlim	An R object that gives the longitude range that you wish to plot, e.g. c(-180, 180)
my.ylim	An R object that gives the longitude range that you wish to plot, e.g. c(-90, 90)
plot.width	Number that gives the plot width, e.g. 8
plot.height	Number that gives the plot height, e.g. 4
numCores	An integer that defines the number of cores, e.g. 2
timeInt	A string that gives the time interval of the model data, e.g. 'month' or 'year'
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.

Value

Figures in PDF format that show the time-mean, spatial-mean, zonal mean, and seasonal cycle.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

long.name <- 'Gross primary productivity'
nc.mod <- system.file('extdata/modelRotated', 'gpp_monthly.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)
timePeriod <- c('1980-01', '2017-12')
outlier.factor <- 1

my.xlim <- c(-171, 0) # longitude range that you wish to plot
my.ylim <- c(32, 78) # latitude range that you wish to plot
plot.width <- 7 # plot width
plot.height <- 3.8 # plot height

plotNcIrreg(long.name, nc.mod, mod.id, unit.conv.mod, variable.unit, timePeriod, outlier.factor)
 #donttest
```

---

plotZonalMeans *Plot zonal mean values of model and reference data in a single figure*

Description

This function plots zonal mean values of model and reference data, computed from zonalMeanStats, for a selection of variables.

Usage

```
plotZonalMeans(mod.path.list, modelIDs, myVariables, metric = "mean",
  lat.range = c(-50, 80), plot.width = 6, plot.height = 10,
  outputDir = FALSE, myFilename = "zonalMeans.pdf",
  legendLocation = "topright", plotRef = TRUE, subcaption = "")
```

Arguments

mod.path.list	A List of directories where AMBER output is stored for different model runs, e.g. list(mod01.path, mod02.path, mod03.path)
modelIDs	An R object with the different model run IDs, e.g. c('CLASSIC.CRUJRAv2', 'CLASSIC.GSWP3W5E5', 'CLASSIC.CRUNCEP')
myVariables	An R object with the variable names of interest, e.g. c('GPP.FluxCom', 'RECO.FluxCom').
metric	Specify for what statistical metric you wish to plot zonal means. Current options are 'mean', 'bias', 'crmse', 'phase', 'iav', 'sd', 'bias.score', 'rmse.score', 'phase.score', and 'iav.score'.
lat.range	Latitudinal range of ticks and labels on the horizontal axis, e.g. c(-50, 80).
plot.width	A number that gives the desired plot width, e.g. 6.
plot.height	A number that gives the desired plot height, e.g. 10.
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
myFilename	A string that gives name of the Figure, e.g. 'zonalMeans.pdf'.
legendLocation	A string that specifies the location of the model ID legend, e.g. 'topright'.
plotRef	Logical. If FALSE, reference data is not plotted. Default is TRUE.
subcaption	A string that defines the subcaption of the figure, e.g. '(a)'.

Value

A figure that shows zonal mean values of model and reference data.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

mod01.path <- system.file('extdata/SIMod01', package = 'amber')
mod02.path <- system.file('extdata/SIMod02', package = 'amber')
mod.path.list <- list(mod01.path, mod02.path)
modelIDs <- c('CLASSIC.CRUJRAv2', 'CLASSIC.GSWP3W5E5')
```

```
myVariables <- c('GPP.MODIS', 'LAI.AVHRR')
plotZonalMeans(mod.path.list, modelIDs, myVariables, metric = 'mean')
```

`plotZonalMeanStats` *Plot zonal mean plots of AMBER results (bias, bias scores, etc)*

Description

This function plots results from zonalMeanStats, i.e. zonal mean values of model and reference data and the zonal mean bias, centralized root-mean-square error, phase, inter-annual variability, and corresponding scores.

Usage

```
plotZonalMeanStats(zonalMeanStats, zonalMeanStatsUnits,
  lat.range = c(-50, 80), outputDir = FALSE)
```

Arguments

`zonalMeanStats`	A string that gives the name and location of the zonalMeanStats file produced by zonalMeanStats, e.g. '/home/project/study/zonalMeanStats'.
`zonalMeanStatsUnits`	A string that gives the name and location of the zonalMeanStatsUnits file produced by zonalMeanStats, e.g. '/home/project/study/zonalMeanStatsUnits'.
`lat.range`	Latitudinal range of ticks and labels on the horizontal axis, e.g. c(-50, 80).
`outputDir`	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.

Value

Figures that show zonal mean values of model and reference data, centralized root-mean-square error, phase, inter-annual variability, and corresponding scores for each variable and globally gridded reference data set.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
```

```
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

zonalMeanStats <- system.file('extdata/zonalMeanStats', 'zonalMeanStats', package = 'amber')
zonalMeanStatsUnits <- system.file('extdata/zonalMeanStats',
 'zonalMeanStatsUnits', package = 'amber')
plotZonalMeanStats(zonalMeanStats, zonalMeanStatsUnits, outputDir = FALSE)
```

`scores.compare` *Compares scores from two model runs*

Description

This function compares scores from two different model runs.

Usage

```
scores.compare(mod01.path, mod02.path, mod01.id, mod02.id,
  score.weights = c(1, 2, 1, 1, 1), plot.width = 7.3,
  plot.height = 5, outputDir = FALSE, defineVariableOrder = FALSE,
  myVariables = myVariables)
```

Arguments

`mod01.path`	A string that gives the path where the output from scores.tables is stored (model 1)
`mod02.path`	A string that gives the path where the output from scores.tables is stored (model 2)
`mod01.id`	A string that gives the name of a model, e.g. 'CanESM2'
`mod02.id`	A string that gives the name of a model, e.g. 'CanESM5'
`score.weights`	R object that gives the weights of each score (S_{bias}, S_{rmse}, S_{phase}, S_{iav}, S_{dist}) that are used for computing the overall score, e.g. c(1,2,1,1,1)
`plot.width`	Number that gives the plot width, e.g. 6
`plot.height`	Number that gives the plot height, e.g. 5
`outputDir`	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
`defineVariableOrder`	Logical. If TRUE, variables are sorted according to the parameter myVariables defined below. Default setting is FALSE.
`myVariables`	An R object with variable names of variables that should be included in table, e.g. c('GPP', 'RECO', 'NEE')

Value

A figure in PDF format that shows the (a) skill score and (b) skill score difference when compared against a different model run. White numbers indicate that score differences are not statistically significant at the 5 percent level using the two-sided Wilcox significance test.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

mod01.path <- system.file('extdata/model01', package = 'amber')
mod02.path <- system.file('extdata/model02', package = 'amber')
mod01.id <- 'Model01'
mod02.id <- 'Model02'
score.weights <- c(1,2,1,1,1)
scores.compare(mod01.path, mod02.path, mod01.id, mod02.id,  score.weights)
```

`scores.compare.benchmarks`

Compares model scores against scores reference scores.

Description

Interpreting scores is challenging as reference data are subject to uncertainty. This function compares two types of scores. The first set of scores expresses model performance and is based on comparing model output against reference data. The second set of scores is based on a comparison of two independent reference data (e.g. remotely sensed GPP against FLUXNET). The difference between both scores reflect the uncertainty of reference data and indicates how well a model could perform given this uncertainty. Scores that are based on reference data only are here referred to as benchmark scores.

Usage

```
scores.compare.benchmarks(bench.path, model.path, model.id,
  plot.width = 8, plot.height = 4.8, outputDir = FALSE,
  defineVariableOrder = FALSE, myVariables = myVariables)
```

Arguments

bench.path	A string that gives the path where benchmarks (i.e. reference vs. reference data) are stored
model.path	A string that gives the path where the output from scores.tables is stored (model)
model.id	A string that gives the name of a model, e.g. 'CLASSIC'
plot.width	Number that gives the plot width, e.g. 7.3
plot.height	Number that gives the plot height, e.g. 6.5
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
defineVariableOrder	Logical. If TRUE, variables are sorted according to the parameter myVariables defined below. Default setting is FALSE.
myVariables	An R object that defines the variables and their order in the score table, e.g. c('GPP', 'RECO', 'NEE').

Value

A figure in PDF format that shows the (a) benchmark skill score, (b) model skill score, and (c) score difference.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

bench.path <- system.file('extdata/scoresBenchmarks', package = 'amber')
model.path <- system.file('extdata/scores', package = 'amber')
```

```
model.id <- 'CLASSIC'
myVariables <- c('ALBS', 'GPP', 'LAI')

scores.compare.benchmarks(bench.path, model.path, model.id, plot.width = 8, plot.height = 4,
defineVariableOrder = TRUE, myVariables = myVariables)
```

`scores.compare.ensemble` *Summarize scores from multiple model runs in single figure.*

Description

This function produces a figure that summarizes score values from multiple model runs. The figure has four columns, which give the multi-model mean scores, the total score range, the model with the lowest score and the model with the highest score. The respective inputs are created by the functions scores.fluxnet.csv or scores.fluxnet.nc, scores.grid.notime, scores.grid.time, and scores.runoff.

Usage

```
scores.compare.ensemble(mod.path.list = mod.path.list,
  modelIDs = modelIDs, myVariables = myVariables, plot.width = 10,
  plot.height = 10, myMargin = c(12, 0, 3, 0), outputDir = FALSE)
```

Arguments

`mod.path.list`	A list with paths for each model run, e.g. mod.path.list <- list(mod01.path, mod02.path, mod03.path)
`modelIDs`	An R object with the different model run IDs, e.g. c('CLASSIC.CRUJRAv2', 'CLASSIC.GSWP3W5E5', 'CLASSIC.CRUNCEP')
`myVariables`	An R object with variable names of variables that should be included in table, e.g. c('GPP', 'RECO', 'NEE')
`plot.width`	Number that gives the plot width, e.g. 6
`plot.height`	Number that gives the plot height, e.g. 5
`myMargin`	An R object that gives the figure margins, e.g. c(4, 13, 3, 4)
`outputDir`	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.

Value

A figure in PDF format that shows the ensemble scores, the total score range, the model with the lowest score and the model with the highest score.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

mod01.path <- system.file('extdata/model01', package = 'amber')
mod02.path <- system.file('extdata/model02', package = 'amber')
mod.path.list <- list(mod01.path, mod02.path)
modelIDs <- c('CLASSIC.CRUJRAv2', 'CLASSIC.GSWP3W5E5')
myVariables <- c('GPP', 'BURNT')
#myVariables <- c('RNS', 'RSS', 'RLS', 'ALBS', 'HFLS', 'HFSS', 'HFG', 'GPP', 'RECO',
#'NEE', 'FIRE', 'AGB', 'CVEG', 'CSOIL', 'LAI', 'BURNT', 'SNW', 'MRSLL', 'MRRO')

scores.compare.ensemble(mod.path.list = mod.path.list, modelIDs = modelIDs,
myVariables = myVariables, plot.width = 9.3, plot.height = 10,
myMargin = c(12, 0, 3, 0), outputDir = FALSE)
```

---

scores.fluxnet.csv *Scores for FLUXNET reference data in CSV format*

Description

This function compares model output against FLUXNET measurements in CSV format. The performance of a model is expressed through scores that range from zero to one, where increasing values imply better performance. These scores are computed in five steps: (i) computation of a statistical metric, (ii) nondimensionalization, (iii) conversion to unit interval, (iv) spatial integration, and (v) averaging scores computed from different statistical metrics. The latter includes the bias, root-mean-square error, phase shift, inter-annual variability, and spatial distribution. The corresponding equations are documented in amber-package.

Usage

```
scores.fluxnet.csv(long.name, nc.mod, ref.csv, mod.id, ref.id,
  unit.conv.mod, unit.conv.ref, variable.unit, score.weights = c(1, 2, 1,
```

```
1, 1), rotate.me = TRUE, irregular = FALSE,
my.projection = "+proj=longlat +ellps=WGS84",
shp.filename = system.file("extdata/ne_110m_land/ne_110m_land.shp",
package = "amber"), my.xlim = c(-180, 180), my.ylim = c(-60, 85),
plot.width = 8, plot.height = 3.8, numCores = 2,
outputDir = FALSE, variable.name = FALSE, numberOfMonths = 36,
phaseMinMax = "phaseMax", meanPerGridCell = TRUE, myCex = 0.5,
subcaption = "")
```

Arguments

long.name	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
nc.mod	A string that gives the path and name of the netcdf file that contains the model output, e.g. '/home/model_gpp.nc'
ref.csv	A string that gives the path and name of the csv file that contains the reference data output, e.g. '/home/reference_gpp.csv'. The columns of this file should contain latitude, longitude, date, variable of interest, and site name.
mod.id	A string that identifies the source of the reference data set, e.g. 'CanESM2'
ref.id	A string that identifies the source of the reference data set, e.g. 'MODIS'
unit.conv.mod	A number that is used as a factor to convert the unit of the model data, e.g. 86400
unit.conv.ref	A number that is used as a factor to convert the unit of the reference data, e.g. 86400
variable.unit	A string that gives the final units using LaTeX notation, e.g. 'gC m$^-2$ day$^-1$'
score.weights	R object that gives the weights of each score (S_{bias}, S_{rmse}, S_{phase}, S_{iav}, S_{dist}) that are used for computing the overall score, e.g. c(1,2,1,1,1)
rotate.me	logical: TRUE if you want longitudes to range from -180 to 180 degrees and FALSE if you want longitudes to range from 0 to 360 degrees
irregular	logical: TRUE if data is on an irregular grid and FALSE if data is on a regular grid
my.projection	A string that gives the projection of the irregular grid
shp.filename	A string that gives the coastline shapefile
my.xlim	An R object that gives the longitude range that you wish to plot, e.g. c(-180, 180)
my.ylim	An R object that gives the longitude range that you wish to plot, e.g. c(-90, 90)
plot.width	Number that gives the plot width, e.g. 8
plot.height	Number that gives the plot height, e.g. 4
numCores	An integer that defines the number of cores, e.g. 2
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.

`variable.name` A string with the variable name, e.g. 'GPP'. If FALSE, the variable name stored in the NetCDF file will be used instead. Default is FALSE.

`numberOfMonths` An integer that gives the mimimum number of months that each site should have, e.g. 60. All sites with fewer months will be excluded.

`phaseMinMax` A string (either 'phaseMax' or 'phaseMin') that determines whether to assess the seasonal peak as a maximum or a minimum. The latter may be appropriate for variables that tend to be negative, such as net longwave radiation or net ecosystem exchange.

`meanPerGridCell` Logical. If TRUE, then values from different sites that are located in the same grid cell are averaged. Default is set to TRUE.

`myCex` A number that determines the size of the dots in the Figure.

`subcaption` A string that defines the subcaption of the figure, e.g. '(a)'.

Value

(1) Figures in PDF format that show maps of the model data at the location of FLUXNET sites (mean, $mod.mean$; interannual-variability, $mod.iav$; month of annual cycle maximum, $mod.max.month$), the reference data (mean, $ref.mean$; interannual-variability, $ref.iav$; month of annual cycle maximum, $ref.max.month$), statistical metrics (bias, $bias$; centralized root mean square error, $crmse$; time difference of the annual cycle maximum, $phase$), and scores (bias score, $bias.score$; root mean square error score, $rmse.score$; inter-annual variability score $iav.score$; annual cycle score ($phase.score$).

(2) Four text files: (i) score values and (ii) score inputs for each individual site, and (iii) score values and (iv) score inputs averaged across sites.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

# (1) Global plots on a regular grid
long.name <- 'Gross primary productivity'
```

```
nc.mod <- system.file('extdata/modelRegular', 'gpp_monthly.nc', package = 'amber')
ref.csv <- system.file('extdata/referenceRegular', 'gpp_monthly_fluxnet.csv', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'FLUXNET' # give reference dataset a name
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
unit.conv.ref <- 1 # optional unit conversion for reference data
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)

# Short version using default settings:
scores.fluxnet.csv(long.name, nc.mod, ref.csv, mod.id, ref.id,
unit.conv.mod, unit.conv.ref, variable.unit)

# To zoom into a particular region:
scores.fluxnet.csv(long.name, nc.mod, ref.csv, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit, score.weights = c(1, 2, 1, 1, 1),
my.xlim = c(-150, -60), my.ylim = c(20, 60), plot.width = 6, plot.height = 3.8)

# (2) Regional plots on a rotated grid
nc.mod <- system.file('extdata/modelRotated', 'gpp_monthly.nc', package = 'amber')
ref.csv <- system.file('extdata/referenceRegular', 'gpp_monthly_fluxnet.csv', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'FLUXNET' # give reference dataset a name
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
unit.conv.ref <- 1 # optional unit conversion for reference data
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)
score.weights <- c(1,2,1,1,1) # score weights of S_bias, S_rmse, S_phase, S_iav, S_dist
rotate.me <- FALSE
irregular <- TRUE
my.projection <-'+proj=ob_tran +o_proj=longlat +o_lon_p=83. +o_lat_p=42.5 +lon_0=263.'
# shp.filename <- system.file('extdata/ne_50m_admin_0_countries/ne_50m_admin_0_countries.shp',
# package = 'amber')
shp.filename <- system.file("extdata/ne_110m_land/ne_110m_land.shp", package = "amber")
my.xlim <- c(-171, 0) # longitude range that you wish to plot
my.ylim <- c(32, 78) # latitude range that you wish to plot
plot.width <- 7
plot.height <- 3.8
numCores <- 2

scores.fluxnet.csv(long.name, nc.mod, ref.csv, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit, score.weights, rotate.me, irregular, my.projection,
shp.filename, my.xlim, my.ylim)
 #donttest
```

`scores.fluxnet.nc` *Scores for FLUXNET reference data in NetCDF format*

Description

This function compares model output against measurements from FLUXNET or from the Baseline Surface Radiation Networking in NetCDF format. The performance of a model is expressed

through scores that range from zero to one, where increasing values imply better performance. These scores are computed in five steps: (i) computation of a statistical metric, (ii) nondimensionalization, (iii) conversion to unit interval, (iv) spatial integration, and (v) averaging scores computed from different statistical metrics. The latter includes the bias, root-mean-square error, phase shift, inter-annual variability, and spatial distribution. The corresponding equations are documented in amber-package.

Usage

```
scores.fluxnet.nc(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
  unit.conv.ref, variable.unit, score.weights, rotate.me = TRUE,
  irregular = FALSE,
 my.projection = "+proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0",
  shp.filename = system.file("extdata/ne_110m_land/ne_110m_land.shp",
  package = "amber"), my.xlim = c(-180, 180), my.ylim = c(-60, 85),
  plot.width = 8, plot.height = 3.8, numCores = 2,
  outputDir = FALSE, phaseMinMax = "phaseMax")
```

Arguments

long.name	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
nc.mod	A string that gives the path and name of the netcdf file that contains the model output, e.g. '/home/model_gpp.nc'
nc.ref	A string that gives the path and name of the netcdf file that contains the reference data output, e.g. '/home/reference_gpp.nc'
mod.id	A string that identifies the source of the reference data set, e.g. 'CanESM2'
ref.id	A string that identifies the source of the reference data set, e.g. 'MODIS'
unit.conv.mod	A number that is used as a factor to convert the unit of the model data, e.g. 86400
unit.conv.ref	A number that is used as a factor to convert the unit of the reference data, e.g. 86400
variable.unit	A string that gives the final units using LaTeX notation, e.g. 'gC m$^-2$ day$^-1$'
score.weights	R object that gives the weights of each score (S_{bias}, S_{rmse}, S_{phase}, S_{iav}, S_{dist}) that are used for computing the overall score, e.g. c(1,2,1,1,1)
rotate.me	logical: TRUE if you want longitudes to range from -180 to 180 degrees and FALSE if you want longitudes to range from 0 to 360 degrees
irregular	logical: TRUE if data is on an irregular grid and FALSE if data is on a regular grid
my.projection	A string that gives the projection of the irregular grid
shp.filename	A string that gives the coastline shapefile
my.xlim	An R object that gives the longitude range that you wish to plot, e.g. c(-180, 180)
my.ylim	An R object that gives the longitude range that you wish to plot, e.g. c(-90, 90)

plot.width	Number that gives the plot width, e.g. 8
plot.height	Number that gives the plot height, e.g. 4
numCores	An integer that defines the number of cores, e.g. 2
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
phaseMinMax	A string (either 'phaseMax' or 'phaseMin') that determines whether to assess the seasonal peak as a maximum or a minimum. The latter may be appropriate for variables that tend to be negative, such as net longwave radiation or net ecosystem exchange.

Value

(1) Figures in PDF format that show global maps of the model data at the location of FLUXNET sites (mean, $mod.mean$; interannual-variability, $mod.iav$; month of annual cycle maximum, $mod.max.month$), the reference data (mean, $ref.mean$; interannual-variability, $ref.iav$; month of annual cycle maximum, $ref.max.month$), statistical metrics (bias, $bias$; centralized root mean square error, $crmse$; time difference of the annual cycle maximum, $phase$), and scores (bias score, $bias.score$; root mean square error score, $rmse.score$; inter-annual variability score $iav.score$; annual cycle score ($phase.score$).

(2) Four text files: (i) score values and (ii) score inputs for each individual site, and (iii) score values and (iv) score inputs averaged across sites. when averaging over all station.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

long.name <- 'Gross primary productivity'
nc.mod <- system.file('extdata/modelRegular', 'gpp_monthly.nc', package = 'amber')
nc.ref <- system.file('extdata/referenceRegular', 'gpp_FLUXNET.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'FLUXNET' # give reference dataset a name
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
unit.conv.ref <- 86400*1000 # optional unit conversion for reference data
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)
```

```
score.weights <- c(1,2,1,1,1) # score weights of S_bias, S_rmse, S_phase, S_iav, S_dist
# global plot
scores.fluxnet.nc(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
 unit.conv.ref, variable.unit, score.weights)


# regional plot
scores.fluxnet.nc(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit, score.weights, my.xlim = c(-150, -60), my.ylim = c(20, 60),
plot.width = 6, plot.height = 3.8)
scores.fluxnet.nc(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit, score.weights)

# (2) Example for data on a rotated grid
nc.mod <- system.file('extdata/modelRotated', 'gpp_monthly.nc', package = 'amber')
nc.ref <- system.file('extdata/referenceRegular', 'gpp_FLUXNET.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'FLUXNET' # give reference dataset a name
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
unit.conv.ref <- 86400*1000 # optional unit conversion for reference data
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)
score.weights <- c(1,2,1,1,1) # score weights of S_bias, S_rmse, S_phase, S_iav, S_dist
rotate.me <- FALSE
irregular <- TRUE
my.projection <-'+proj=ob_tran +o_proj=longlat +o_lon_p=83. +o_lat_p=42.5 +lon_0=263.'
# shp.filename <- system.file('extdata/ne_50m_admin_0_countries/ne_50m_admin_0_countries.shp',
# package = 'amber')
shp.filename <- system.file("extdata/ne_110m_land/ne_110m_land.shp", package = "amber")
my.xlim <- c(-171, 0) # longitude range that you wish to plot
my.ylim <- c(32, 78) # latitude range that you wish to plot
plot.width <- 7 # plot width
plot.height <- 3.8 # plot height
numCores = 2

scores.fluxnet.nc(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit, score.weights, rotate.me, irregular,
my.projection,
shp.filename, my.xlim, my.ylim)
 #donttest
```

`scores.fluxnet.site` *Scores for FLUXNET reference data when model run at FLUXNET site*

Description

This function compares model output in CSV format against FLUXNET measurements in CSV format. Use this function when running your model at each FLUXNET site individually. The performance of a model is expressed through scores that range from zero to one, where increasing values imply better performance. These scores are computed in five steps: (i) computation of

a statistical metric, (ii) nondimensionalization, (iii) conversion to unit interval, (iv) spatial integration, and (v) averaging scores computed from different statistical metrics. The latter includes the bias, root-mean-square error, phase shift, inter-annual variability, and spatial distribution. The corresponding equations are documented in amber-package.

Usage

```
scores.fluxnet.site(long.name, mod.csv, mod.csv.path, ref.csv, mod.id,
  ref.id, unit.conv.mod, unit.conv.ref, variable.unit, sites,
  score.weights = c(1, 2, 1, 1, 1), my.xlim = c(-180, 180),
  my.ylim = c(-60, 85), plot.width = 8, plot.height = 3.8,
  numCores = 2, outputDir = FALSE, phaseMinMax = "phaseMax",
  myCex = 0.7)
```

Arguments

long.name	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
mod.csv	A string that gives the name of the model output in csv format, e.g. 'gpp_monthly.csv'
mod.csv.path	A string that gives the path to the model output for site-level runs
ref.csv	A string that gives the path and name of the csv file that contains the reference data output, e.g. '/home/reference_gpp.csv'. The columns of this file should contain latitude, longitude, date, variable of interest, and site name.
mod.id	A string that identifies the source of the reference data set, e.g. 'CanESM2'
ref.id	A string that identifies the source of the reference data set, e.g. 'MODIS'
unit.conv.mod	A number that is used as a factor to convert the unit of the model data, e.g. 86400
unit.conv.ref	A number that is used as a factor to convert the unit of the reference data, e.g. 86400
variable.unit	A string that gives the final units using LaTeX notation, e.g. 'gC m$^-2$ day$^-1$'
sites	A vector of strings that give the fluxnet site names, e.g. c('AU-Tum','BR-Sa1','CA-Qfo')
score.weights	R object that gives the weights of each score (S_{bias}, S_{rmse}, S_{phase}, S_{iav}, S_{dist}) that are used for computing the overall score, e.g. c(1,2,1,1,1)
my.xlim	An R object that gives the longitude range that you wish to plot, e.g. c(-180, 180)
my.ylim	An R object that gives the longitude range that you wish to plot, e.g. c(-90, 90)
plot.width	Number that gives the plot width, e.g. 8
plot.height	Number that gives the plot height, e.g. 4
numCores	An integer that defines the number of cores, e.g. 2
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.

phaseMinMax	A string (either 'phaseMax' or 'phaseMin') that determines whether to assess the seasonal peak as a maximum or a minimum. The latter may be appropriate for variables that tend to be negative, such as net longwave radiation or net ecosystem exchange.
myCex	A number that determines the size of the dots in the Figure. Default is set to 0.7.

Value

(1) Figures in PDF format that show maps of the model data at the location of FLUXNET sites (mean, $mod.mean$; interannual-variability, $mod.iav$; month of annual cycle maximum, $mod.max.month$), the reference data (mean, $ref.mean$; interannual-variability, $ref.iav$; month of annual cycle maximum, $ref.max.month$), statistical metrics (bias, $bias$; centralized root mean square error, $crmse$; time difference of the annual cycle maximum, $phase$), and scores (bias score, $bias.score$; root mean square error score, $rmse.score$; inter-annual variability score $iav.score$; annual cycle score ($phase.score$).

(2) Four text files: (i) score values and (ii) score inputs for each individual site, and (iii) score values and (iv) score inputs averaged across sites. when averaging over all station.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

long.name <- 'Gross primary productivity'
mod.csv <- 'gpp_monthly.csv'
mod.csv.path <- system.file('extdata/siteLevelRun', package = 'amber')
ref.csv <- system.file('extdata/referenceRegular', 'gpp_monthly_fluxnet.csv', package = 'amber')
mod.id <- 'CLASSIC-Sitelevel' # define a model experiment ID
ref.id <- 'FLUXNET' # give reference dataset a name
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
unit.conv.ref <- 1 # optional unit conversion for reference data
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)

sites <- c('AU-Tum','CA-TPD', 'US-WCr')
```

```
# Short version using default settings:
scores.fluxnet.site(long.name, mod.csv, mod.csv.path, ref.csv, mod.id, ref.id,
unit.conv.mod, unit.conv.ref, variable.unit, sites)

# Additional parameters:
score.weights <- c(1,2,1,1,1) # score weights of S_bias, S_rmse, S_phase, S_iav, S_dist
my.xlim <- c(-180, 180)
my.ylim <- c(-60, 85)
plot.width <- 8
plot.height <- 3.8
numCores <- 2

scores.fluxnet.site(long.name, mod.csv, mod.csv.path, ref.csv, mod.id, ref.id,
unit.conv.mod, unit.conv.ref, variable.unit, sites, score.weights,
my.xlim, my.ylim, plot.height, numCores)

# To zoom into a particular region:
scores.fluxnet.site(long.name, mod.csv, mod.csv.path, ref.csv, mod.id, ref.id,
unit.conv.mod, unit.conv.ref, variable.unit, sites,
my.xlim = c(-150, -60), my.ylim = c(20, 60), plot.width = 6, plot.height = 3.8)
 #donttest
```

`scores.functional.response`

Response of a variable to its forcing

Description

This function conducts a relationship analysis by assessing the functional response of two variables. The variables consist of a dependent variable y (e.g. GPP) and an independent variable x (e.g. temperature). Usually, there are 4 datasets involved: (1) modeled x, (2) modeled y, (3) reference x, (4) reference y, where $reference$ refers to observation-based datasets. The time period of analysis should cover the period that all data sets have in common. When the model is forced with observations, (1) and (3) are identical. The common time period is then defined by (2) and (4). The performance of a model is expressed through a score value that ranges from zero to one, where increasing values imply better performance:

$$\varepsilon^{u}_{func} = \sqrt{\frac{\int (f_{mod}(u) - f_{ref}(u))^2 du}{\int f_{ref}(u)^2 du}}$$

where $f_{mod}(u)$ and $f_{ref}(u)$ are the binned time means of the model and reference data, respectively.

Usage

```
scores.functional.response(nc.mod.x, nc.mod.y, nc.ref.x, nc.ref.y, mod.id,
  ref.id, unit.conv.mod.x, unit.conv.mod.y, unit.conv.ref.x,
  unit.conv.ref.y, x.bin, y.bin, my.xlab, my.ylab, legendA = "topleft",
  legendB = "topleft", outputDir = FALSE)
```

Arguments

nc.mod.x	A string that gives the path and name of the netcdf file that contains the model forcing data, e.g. '/home/model_tas.nc'
nc.mod.y	A string that gives the path and name of the netcdf file that contains the model output data, e.g. '/home/model_gpp.nc'
nc.ref.x	A string that gives the path and name of the netcdf file that contains the reference forcing data, e.g. '/home/ref_tas.nc'
nc.ref.y	A string that gives the path and name of the netcdf file that contains the reference data, e.g. '/home/ref_gpp.nc'
mod.id	A string that identifies the source of the reference data set, e.g. 'CLASSIC'
ref.id	A string that identifies the source of the reference data set, e.g. 'MODIS'
unit.conv.mod.x	A number that is used as a factor to convert the unit of the model forcing data, e.g. 86400
unit.conv.mod.y	A number that is used as a factor to convert the unit of the model output data, e.g. 86400
unit.conv.ref.x	A number that is used as a factor to convert the unit of the reference forcing data, e.g. 86400
unit.conv.ref.y	A number that is used as a factor to convert the unit of the reference data, e.g. 86400
x.bin	A number that gives the size of the x-bin, e.g. 2
y.bin	A number that gives the size of the y-bin, e.g. 2
my.xlab	A string that serves as the label for the x-axis, e.g. expression(paste('near surface air temperature (',degree,'C)')) # (R notation)
my.ylab	A string that serves as the label for the y-axis, e.g. expression('GPP (g C m'^-2~'day'^-1~')') # (R notation)
legendA	A string that defines the legend location for the upper subplot. The default value is 'topleft'.
legendB	A string that defines the legend location for the lower subplot. The default value is 'topleft'.
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.

Value

A figure in PDF format that shows the functional relation ship between two variables, the frequency of grid cells for each bin, and the score value.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

nc.mod.x <- system.file('extdata/modelRegular', 'tas_monthly.nc', package = 'amber')
nc.mod.y <- system.file('extdata/modelRegular', 'gpp_monthly.nc', package = 'amber')
nc.ref.x <- system.file('extdata/modelRegular', 'tas_monthly.nc', package = 'amber')
nc.ref.y <- system.file('extdata/referenceRegular', 'gpp_GBAF_128x64.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'GBAF' # give reference dataset a name
unit.conv.mod.x <- 1
unit.conv.mod.y <- 86400*1000
unit.conv.ref.x <- 1
unit.conv.ref.y <- 86400*1000
x.bin <- 2.5 # adjust as required
y.bin <- 1 # adjust as required
my.xlab <- expression(paste('near surface air temperature (',degree,'C)')) # (R notation)
my.ylab <- expression('GPP (g C m'^{-2}~'day'^{-1}~')') # (R notation)

scores.functional.response(nc.mod.x, nc.mod.y, nc.ref.x, nc.ref.y, mod.id,
ref.id, unit.conv.mod.x, unit.conv.mod.y, unit.conv.ref.x, unit.conv.ref.y,
x.bin, y.bin, my.xlab, my.ylab)
```

`scores.grid.notime` *Scores for gridded reference data that do not have a varying time dimension*

Description

This function compares model output against remote-sensing based reference data that do not vary in time. The performance of a model is expressed through a score that ranges from zero to one, where increasing values imply better performance. Contrary to the function scores.grid.time, only two scores are computed (bias score S_{bias} and spatial distribution score, S_{dist}) since the reference

data do not vary with time. Contrary to scores.grid.time, the bias is relative to the absolute reference mean value rather than the reference standard deviation. Again, this is because the reference data do not vary with time:

$(i)\ bias(\lambda,\phi) = \overline{v_{mod}}(\lambda,\phi) - \overline{v_{ref}}(\lambda,\phi)$

$(ii)\ \varepsilon_{bias} = |bias(\lambda,\phi)|/|\overline{v_{ref}}(\lambda,\phi)|$

$(iii)\ s_{bias}(\lambda,\phi) = e^{-\varepsilon_{bias}(\lambda,\phi)}$

$(iv)\ S_{bias} = \overline{\overline{s_{bias}}}$

Usage

```
scores.grid.notime(long.name, nc.mod, nc.ref, mod.id, ref.id,
  unit.conv.mod, unit.conv.ref, variable.unit, score.weights = c(1, 2, 1,
  1, 1), outlier.factor = 1000, irregular = FALSE,
 my.projection = "+proj=ob_tran +o_proj=longlat +o_lon_p=83. +o_lat_p=42.5 +lon_0=263.",
  numCores = 2, period = c("1980-01", "2017-12"), timeInt = "month",
  outputDir = FALSE, variable.name = FALSE)
```

Arguments

long.name	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
nc.mod	A string that gives the path and name of the netcdf file that contains the model output, e.g. '/home/model_gpp.nc'
nc.ref	A string that gives the path and name of the netcdf file that contains the reference data output, e.g. '/home/reference_gpp.nc'
mod.id	A string that identifies the source of the reference data set, e.g. 'CanESM2'
ref.id	A string that identifies the source of the reference data set, e.g. 'MODIS'
unit.conv.mod	A number that is used as a factor to convert the unit of the model data, e.g. 86400
unit.conv.ref	A number that is used as a factor to convert the unit of the reference data, e.g. 86400
variable.unit	A string that gives the final units using LaTeX notation, e.g. 'gC m$^-2$ day$^-1$'
score.weights	R object that gives the weights of each score (S_{bias}, S_{rmse}, S_{phase}, S_{iav}, S_{dist}) that are used for computing the overall score, e.g. c(1,2,1,1,1)
outlier.factor	A number that is used to define outliers, e.g. 10. Plotting raster objects that contain extreme outliers lead to figures where most grid cells are presented by a single color since the color legend covers the entire range of values. To avoid this, the user may define outliers that will be masked out and marked with a red dot. Outliers are all values that exceed the interquartile range multiplied by the outlier factor defined here.
irregular	Logical. If TRUE the data are converted from an irregular to a regular grid. Default is FALSE.
my.projection	A string that defines the projection of the irregular grid

numCores	An integer that defines the number of cores, e.g. 2
period	An R object that gives the period over which to average the model data, e.g. c('1980-01', '2017-12')
timeInt	A string that gives the time interval of the model data, e.g. 'month' or 'year'
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
variable.name	A string with the variable name, e.g. 'GPP'. If FALSE, the variable name stored in the NetCDF file will be used instead. Default is FALSE.

Value

(1) A list that contains three elements. The first element is a raster stack with model data (mean, $mod.mean$), reference data (mean, $ref.mean$), and the corresponding bias (bias, $bias$). The second and third element of the list are spatial point data frames that give the model and reference outliers, respectively. The content of the list can be plotted using plotGrid.

(2) NetCDF files for each of the statistical variables listed above. (3) Three text files: (i) score values and (ii) score inputs averaged across the entire grid, and (iii) score values for each individual grid cell.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

# (1) Global plots on a regular grid
long.name <- 'Soil Carbon'
nc.mod <- system.file('extdata/modelRegular', 'cSoil_monthly.nc', package = 'amber')
nc.ref <- system.file('extdata/referenceRegular', 'soilc_HWSD_128x64.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'HWSD' # give reference dataset a name
unit.conv.mod <- 1 # optional unit conversion for model data
unit.conv.ref <- 1 # optional unit conversion for reference data
variable.unit <- 'kgC m$^{-2}$' # unit after conversion (LaTeX notation)

# Short version using default settings:
```

```
plot.me <- scores.grid.notime(long.name, nc.mod, nc.ref, mod.id, ref.id,
unit.conv.mod, unit.conv.ref, variable.unit)
plotGrid(long.name, plot.me)


# (2) Regional plots on a rotated grid
long.name <- 'Soil Carbon'
nc.mod <- system.file('extdata/modelRotated', 'cSoil_monthly.nc', package = 'amber')
nc.ref <- system.file('extdata/referenceRotated', 'soilc_HWSD_rotated.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'HWSD' # give reference dataset a name
unit.conv.mod <- 1 # optional unit conversion for model data
unit.conv.ref <- 1 # optional unit conversion for reference data
variable.unit <- 'kgC m$^{-2}$' # unit after conversion (LaTeX notation)
score.weights <- c(1,2,1,1,1) # score weights of S_bias, S_rmse, S_phase, S_iav, S_dist
outlier.factor <- 1000
irregular <- TRUE
my.projection <- '+proj=ob_tran +o_proj=longlat +o_lon_p=83. +o_lat_p=42.5 +lon_0=263.'
numCores <- 2
period <- c('1980-01', '2017-12') # period over which to average the model data

plot.me <- scores.grid.notime(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit, score.weights, outlier.factor, irregular, my.projection,
numCores, period)

# Plot results:
irregular <- TRUE # data is on an irregular grid
my.projection <- '+proj=ob_tran +o_proj=longlat +o_lon_p=83. +o_lat_p=42.5 +lon_0=263.'
# shp.filename <- system.file('extdata/ne_50m_admin_0_countries/ne_50m_admin_0_countries.shp',
#  package = 'amber')
shp.filename <- system.file("extdata/ne_110m_land/ne_110m_land.shp", package = "amber")
my.xlim <- c(-171, 0) # longitude range that you wish to plot
my.ylim <- c(32, 78) # latitude range that you wish to plot
plot.width <- 7 # plot width
plot.height <- 3.8 # plot height

plotGrid(long.name, plot.me, irregular, my.projection,
shp.filename, my.xlim, my.ylim, plot.width, plot.height)
 #donttest
```

`scores.grid.time` *Scores for gridded reference data with a varying time dimension*

Description

This function compares model output against remote-sensing based reference data that vary in time. The performance of a model is expressed through scores that range from zero to one, where increasing values imply better performance. These scores are computed in five steps: (i) computation of a statistical metric, (ii) nondimensionalization, (iii) conversion to unit interval, (iv) spatial integration, and (v) averaging scores computed from different statistical metrics. The latter includes

the bias, root-mean-square error, phase shift, inter-annual variability, and spatial distribution. The corresponding equations are documented in amber-package.

Usage

```
scores.grid.time(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
  unit.conv.ref, variable.unit, score.weights = c(1, 2, 1, 1, 1),
  outlier.factor = 1000, irregular = FALSE,
 my.projection = "+proj=ob_tran +o_proj=longlat +o_lon_p=83. +o_lat_p=42.5 +lon_0=263.",
  numCores = 2, timeInt = "month", outputDir = FALSE, myLevel = 1,
  variable.name = FALSE, phaseMinMax = "phaseMax")
```

Arguments

long.name	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
nc.mod	A string that gives the path and name of the netcdf file that contains the model output, e.g. '/home/model_gpp.nc'
nc.ref	A string that gives the path and name of the netcdf file that contains the reference data output, e.g. '/home/reference_gpp.nc'
mod.id	A string that identifies the source of the reference data set, e.g. 'CLASSIC'
ref.id	A string that identifies the source of the reference data set, e.g. 'MODIS'
unit.conv.mod	A number that is used as a factor to convert the unit of the model data, e.g. 86400
unit.conv.ref	A number that is used as a factor to convert the unit of the reference data, e.g. 86400
variable.unit	A string that gives the final units using LaTeX notation, e.g. 'gC m$^-2$ day$^-1$'
score.weights	R object that gives the weights of each score (S_{bias}, S_{rmse}, S_{phase}, S_{iav}, S_{dist}) that are used for computing the overall score, e.g. c(1,2,1,1,1)
outlier.factor	A number that is used to define outliers, e.g. 10. Plotting raster objects that contain extreme outliers lead to figures where most grid cells are presented by a single color since the color legend covers the entire range of values. To avoid this, the user may define outliers that will be masked out and marked with a red dot. Outliers are all values that exceed the interquartile range multiplied by the outlier factor defined here.
irregular	Logical. If TRUE the data is converted from an irregular to a regular grid. Default is FALSE.
my.projection	A string that defines the projection of the irregular grid
numCores	An integer that defines the number of cores, e.g. 2
timeInt	A string that gives the time interval of the model data, e.g. 'month' or 'year'
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.

myLevel	A number that determines what level of the output netCDF file to use. This is relevant for files with multiple levels, which applies to soil data. By default, myLevel is set to 1.
variable.name	A string with the variable name, e.g. 'GPP'. If FALSE, the variable name stored in the NetCDF file will be used instead. Default is FALSE.
phaseMinMax	A string (either 'phaseMax' or 'phaseMin') that determines whether to assess the seasonal peak as a maximum or a minimum. The latter may be appropriate for variables that tend to be negative, such as net longwave radiation or net ecosystem exchange.

Value

(1) A list that contains three elements. The first element is a raster stack with model data (mean, $mod.mean$; standard deviation; interannual-variability, $mod.iav$; monthly mean climatology; month of annual cycle maximum, $mod.max.month$), the reference data (mean, $ref.mean$; standard deviation; interannual-variability, $ref.iav$; monthly mean climatology; month of annual cycle maximum, $ref.max.month$), statistical metrics (bias, $bias$; centralized root mean square error, $crmse$; time difference of the annual cycle maximum, $phase$), and scores (bias score, $bias.score$; root mean square error score, $rmse.score$; inter-annual variability score $iav.score$; annual cycle score ($phase.score$). The second and third element of the list are spatial point data frames that give the model and reference outliers, respectively. Most of the content of the list can be plotted using plotGrid. The only exception is monthly mean climatology.

(2) NetCDF files for each of the statistical variables listed above.

(3) Three text files: (i) score values and (ii) score inputs averaged across the entire grid, and (iii) score values for each individual grid cell.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

# (1) Global plots on a regular grid
long.name <- 'Gross primary productivity'
nc.mod <- system.file('extdata/modelRegular', 'gpp_monthly.nc', package = 'amber')
```

```
nc.ref <- system.file('extdata/referenceRegular', 'gpp_GBAF_128x64.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'GBAF' # give reference dataset a name
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
unit.conv.ref <- 86400*1000 # optional unit conversion for reference data
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)

plot.me <- scores.grid.time(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit)
plotGrid(long.name, plot.me)

# (2) Regional plots on a rotated grid
long.name <- 'Gross primary productivity'
nc.mod <- system.file('extdata/modelRotated', 'gpp_monthly.nc', package = 'amber')
nc.ref <- system.file('extdata/referenceRotated', 'gpp_GBAF_rotated.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'GBAF' # give reference dataset a name
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
unit.conv.ref <- 86400*1000 # optional unit conversion for reference data
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)
my.projection <- '+proj=ob_tran +o_proj=longlat +o_lon_p=83. +o_lat_p=42.5 +lon_0=263.'

plot.me <- scores.grid.time(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit, score.weights = c(1, 2, 1, 1, 1), outlier.factor = 1000,
irregular = TRUE, my.projection = my.projection, numCores = 2, timeInt = 'month',
outputDir = FALSE, myLevel = 1, variable.name = FALSE)

# Plot results:
irregular <- TRUE # data is on an irregular grid
my.projection <- '+proj=ob_tran +o_proj=longlat +o_lon_p=83. +o_lat_p=42.5 +lon_0=263.'
# shp.filename <- system.file('extdata/ne_50m_admin_0_countries/ne_50m_admin_0_countries.shp',
#  package = 'amber')
shp.filename <- system.file("extdata/ne_110m_land/ne_110m_land.shp", package = "amber")
my.xlim <- c(-171, 0) # longitude range that you wish to plot
my.ylim <- c(32, 78) # latitude range that you wish to plot
plot.width <- 7 # plot width
plot.height <- 3.8 # plot height

plotGrid(long.name, plot.me, irregular, my.projection,
shp.filename, my.xlim, my.ylim, plot.width, plot.height)
 #donttest
```

`scores.runoff` *Scores for runoff*

Description

This function compares modelled runoff and measured streamflow on an annual basis for 50 river basins. Modelled runoff is converted to streamflow by spatially integrating annual runoff across

each river basins. The performance of a model is expressed through scores that range from zero to one, where increasing values imply better performance. These scores are computed in five steps: (i) computation of a statistical metric, (ii) nondimensionalization, (iii) conversion to unit interval, (iv) spatial integration, and (v) averaging scores computed from different statistical metrics. The latter includes the bias, root-mean-square error, inter-annual variability, and spatial distribution. The corresponding equations are documented in amber-package. Contrary to the function scores.grid.time, no phase score is computed since data is evaluated on an annual basis. Assessing annual rather than monthly values causes the standard deviation $sigma_{ref}(\lambda, \phi)$ to be very small. For this reason, the relative errors for runoff are computed with repect to the mean rather than the standard deviation:

$(ii)\ \varepsilon_{bias} = |bias(\lambda, \phi)|/\overline{v_{ref}}(\lambda, \phi)\ (ii)\ \varepsilon_{rmse}(\lambda, \phi) = crmse(\lambda, \phi)/\overline{v_{ref}}(\lambda, \phi).$

Usage

```
scores.runoff(long.name, nc.mod, nc.ref, nc.basins, mod.id, ref.id,
  unit.conv.mod, unit.conv.ref, variable.unit, score.weights = c(1, 2, 1,
  1, 1),
  shp.filename = system.file("extdata/ne_110m_land/ne_110m_land.shp",
  package = "amber"), my.xlim = c(-180, 180), my.ylim = c(-60, 85),
  plot.width = 8, plot.height = 3.8, outputDir = FALSE,
  subcaption = "")
```

Arguments

long.name	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
nc.mod	A string that gives the path and name of the netcdf file that contains the model output, e.g. '/home/model_gpp.nc'
nc.ref	A string that gives the path and name of the netcdf file that contains the reference data output, e.g. '/home/reference_gpp.nc'
nc.basins	A string that gives the path and name of the netcdf file that contains the river basins, e.g. '/home/basins.nc'
mod.id	A string that identifies the source of the reference data set, e.g. 'CanESM2'
ref.id	A string that identifies the source of the reference data set, e.g. 'MODIS'
unit.conv.mod	A number that is used as a factor to convert the unit of the model data, e.g. 86400
unit.conv.ref	A number that is used as a factor to convert the unit of the reference data, e.g. 86400
variable.unit	A string that gives the final units using LaTeX notation, e.g. 'gC m$^-2$ day$^-1$'
score.weights	R object that gives the weights of each score (S_{bias}, S_{rmse}, S_{phase}, S_{iav}, S_{dist}) that are used for computing the overall score, e.g. c(1,2,1,1,1)
shp.filename	A string that gives the coastline shapefile
my.xlim	An R object that gives the longitude range that you wish to plot, e.g. c(-180, 180)

my.ylim	An R object that gives the longitude range that you wish to plot, e.g. c(-90, 90)
plot.width	Number that gives the plot width, e.g. 8
plot.height	Number that gives the plot height, e.g. 4
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory
subcaption	A string that defines the subcaption of the figure, e.g. '(a)'.

Value

(1) Figures in PDF format that show global maps of the model data (mean, $mod.mean$; interannual-variability, $mod.iav$), the reference data (mean, $ref.mean$; interannual-variability, $ref.iav$), statistical metrics (bias, $bias$; centralized root mean square error, $crmse$), and scores (bias score, $bias.score$; root mean square error score, $rmse.score$; inter-annual variability score $iav.score$).

(2) Three text files: (i) score values and (ii) score inputs averaged across the entire grid, and (iii) score values for each individual river basin.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

long.name <- 'Streamflow'
nc.mod <- system.file('extdata/modelRegular', 'mrro_monthly.nc', package = 'amber')
nc.ref <- system.file('extdata/referenceRegular', 'runoff.nc', package = 'amber')
nc.basins <- system.file('extdata/referenceRegular', 'basins.nc', package = 'amber')
mod.id <- 'CLASSIC' # model name
ref.id <- 'GRDC' # give reference dataset a name
unit.conv.mod <- 86400 # optional unit conversion for model data
unit.conv.ref <- 86400 # optional unit conversion for reference data
variable.unit <- 'kg m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)
score.weights <- c(1,2,1,1,1) # define score weights

scores.runoff(long.name, nc.mod, nc.ref, nc.basins, mod.id, ref.id, unit.conv.mod,
```

```
unit.conv.ref, variable.unit, score.weights)
 #donttest
```

`scores.site.notime` *Scores for site-level reference data that do not vary in time*

Description

This function compares model output against site-level measurements such as carbon stocks. The performance of a model is expressed through scores that range from zero to one, where increasing values imply better performance. Contrary to the function scores.grid.time, only two scores are computed (bias score S_{bias} and spatial distribution score, S_{dist}) since the reference data do not vary with time. Contrary to scores.grid.time, the bias is relative to the absolute reference mean value rather than the reference standard deviation. Again, this is because the reference data do not vary with time:

(i) $bias(\lambda, \phi) = \overline{v_{mod}}(\lambda, \phi) - \overline{v_{ref}}(\lambda, \phi)$

(ii) $\varepsilon_{bias} = |bias(\lambda, \phi)|/|\overline{v_{ref}}(\lambda, \phi)|$

(iii) $s_{bias}(\lambda, \phi) = e^{-\varepsilon_{bias}(\lambda, \phi)}$

(iv) $S_{bias} = \overline{\overline{s_{bias}}}$

Usage

```
scores.site.notime(long.name, nc.mod, ref.csv, mod.id, ref.id,
  unit.conv.mod, unit.conv.ref, variable.unit, score.weights = c(1, 2, 1,
  1, 1), rotate.me = TRUE, irregular = FALSE,
  my.projection = "+proj=longlat +ellps=WGS84",
  shp.filename = system.file("extdata/ne_110m_land/ne_110m_land.shp",
  package = "amber"), my.xlim = c(-180, 180), my.ylim = c(-60, 85),
  plot.width = 8, plot.height = 3.8, numCores = 2,
  period = c("1980-01", "2017-12"), outputDir = FALSE,
  variable.name = FALSE, meanPerGridCell = TRUE, myCex = 0.5,
  subcaption = "")
```

Arguments

`long.name`	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
`nc.mod`	A string that gives the path and name of the netcdf file that contains the model output, e.g. '/home/model_gpp.nc'
`ref.csv`	A string that gives the path and name of the csv file that contains the reference data output, e.g. '/home/reference_biomass.csv'. The columns must be in the following order: Plot ID, longitude, latitude, data values.
`mod.id`	A string that identifies the source of the reference data set, e.g. 'CanESM2'
`ref.id`	A string that identifies the source of the reference data set, e.g. 'MODIS'

unit.conv.mod	A number that is used as a factor to convert the unit of the model data, e.g. 86400
unit.conv.ref	A number that is used as a factor to convert the unit of the reference data, e.g. 86400
variable.unit	A string that gives the final units using LaTeX notation, e.g. 'gC m$^-2$ day$^-1$'
score.weights	R object that gives the weights of each score (S_{bias}, S_{rmse}, S_{phase}, S_{iav}, S_{dist}) that are used for computing the overall score, e.g. c(1,2,1,1,1)
rotate.me	logical: TRUE if you want longitudes to range from -180 to 180 degrees and FALSE if you want longitudes to range from 0 to 360 degrees
irregular	logical: TRUE if data is on an irregular grid and FALSE if data is on a regular grid
my.projection	A string that gives the projection of the irregular grid
shp.filename	A string that gives the coastline shapefile
my.xlim	An R object that gives the longitude range that you wish to plot, e.g. c(-180, 180)
my.ylim	An R object that gives the longitude range that you wish to plot, e.g. c(-90, 90)
plot.width	Number that gives the plot width, e.g. 8
plot.height	Number that gives the plot height, e.g. 4
numCores	An integer that defines the number of cores, e.g. 2
period	An R obect that gives the period over which to average the model data, e.g. c('1980-01', '2017-12')
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
variable.name	A string with the variable name, e.g. 'GPP'. If FALSE, the variable name stored in the NetCDF file will be used instead. Default is FALSE.
meanPerGridCell	Logical. If TRUE, then values from different sites that are located in the same grid cell are averaged. Default is set to TRUE.
myCex	A number that determines the size of the dots in the Figure. Default is set to 0.7.
subcaption	A string that defines the subcaption of the figure, e.g. '(a)'.

Value

(1) Figures in PDF format that show maps of the model mean, reference mean, and bias. (2) Four text files: (i) score values and (ii) score inputs for each individual site, and (iii) score values and (iv) score inputs averaged across sites.

Examples

```
library(amber)
library(classInt)
library(doParallel)
```

```
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

# (1) Global plots on a regular grid
long.name <- 'soil carbon'
nc.mod <- system.file('extdata/modelRegular', 'cSoil_monthly.nc', package = 'amber')
ref.csv <- system.file('extdata/siteLevelRefData', 'siteLevelDataNoTime.csv', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'ABC' # give reference dataset a name
unit.conv.mod <- 1 # optional unit conversion for model data
unit.conv.ref <- 1 # optional unit conversion for reference data
variable.unit <- 'kgC m$^{-2}$' # unit after conversion (LaTeX notation)

# Short version using default settings:
scores.site.notime(long.name, nc.mod, ref.csv, mod.id, ref.id,
unit.conv.mod, unit.conv.ref, variable.unit)

# To zoom into a particular region:
scores.site.notime(long.name, nc.mod, ref.csv, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit, score.weights = c(1, 2, 1, 1, 1),
my.xlim = c(-150, -60), my.ylim = c(20, 60), plot.width = 6, plot.height = 3.8)

# (2) Regional plots on a rotated grid
nc.mod <- system.file('extdata/modelRotated', 'cSoil_monthly.nc', package = 'amber')
ref.csv <- system.file('extdata/siteLevelRefData', 'siteLevelDataNoTime.csv', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'ABC' # give reference dataset a name
unit.conv.mod <- 1 # optional unit conversion for model data
unit.conv.ref <- 1 # optional unit conversion for reference data
variable.unit <- 'kgC m$^{-2}$' # unit after conversion (LaTeX notation)
rotate.me <- FALSE
irregular <- TRUE
my.projection <-'+proj=ob_tran +o_proj=longlat +o_lon_p=83. +o_lat_p=42.5 +lon_0=263.'
# shp.filename <- system.file('extdata/ne_50m_admin_0_countries/ne_50m_admin_0_countries.shp',
#  package = 'amber')
shp.filename <- system.file("extdata/ne_110m_land/ne_110m_land.shp", package = "amber")
my.xlim <- c(-171, 0) # longitude range that you wish to plot
my.ylim <- c(32, 78) # latitude range that you wish to plot
plot.width <- 7
plot.height <- 3.8
numCores <- 2
```

```
scores.site.notime(long.name, nc.mod, ref.csv, mod.id, ref.id,
unit.conv.mod, unit.conv.ref, variable.unit, rotate.me = TRUE, irregular = TRUE,
my.projection = my.projection, shp.filename = shp.filename,
my.xlim = my.xlim,  my.ylim = my.ylim, plot.width = plot.width, plot.height = plot.height)
 #donttest
```

`scores.tables` *Summarize results in a table and a plot*

Description

This function merges all tables that have been created by the functions scores.fluxnet.csv or scores.fluxnet.nc, scores.grid.notime, scores.grid.time, and scores.runoff.

Usage

```
scores.tables(plot.width = 6, plot.height = 5, myMargin = c(4, 13, 3,
  4), inputDir = getwd(), outputDir = FALSE, sortByScore = FALSE,
  defineVariableOrder = FALSE, myVariables = myVariables)
```

Arguments

`plot.width`	Number that gives the plot width, e.g. 6
`plot.height`	Number that gives the plot height, e.g. 5
`myMargin`	An R object that gives the figure margins, e.g. c(4, 13, 3, 4)
`inputDir`	A string that gives the input directory, e.g. '/home/project/study'.
`outputDir`	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
`sortByScore`	logical. If TRUE, score table is sorted by overall score. FALSE by default.
`defineVariableOrder`	Logical. If TRUE, variables are sorted according to the parameter myVariables defined below. Default setting is FALSE.
`myVariables`	An R object with variable names of variables that should be included in table, e.g. c('GPP', 'RECO', 'NEE')

Value

(1) Two tables in LaTeX format that shows the scores of all variables that were assessed (with and without mass weighting). (2) Two figures in PDF format that show the same information as (1). (3) Two NetCDF files that show the same information as (1). (4) Five tables in LaTeX format that show the globally averaged statistical metrics for calculating a score (without mass weighting). (S_{bias}, S_{rmse}, S_{phase}, S_{iav}, S_{dist}).

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

myInputDir <- paste(system.file('extdata', package = 'amber'), 'scores', sep = '/')
scores.tables(inputDir = myInputDir, sortByScore = TRUE)
```

---

scores.tables.tweak *Tweak summary table*

Description

This function allows the user to tweak the summary table computed by scores.tables. Contrary to scores.tables, this function can be used to create a single summary table that includes the most important metrics only. The user can specify what variables to include and in what order they should appear.

Usage

```
scores.tables.tweak(myVariables,
  myCaption = "Globally averaged statistical metrics",
  inputDir = getwd(), outputDir = FALSE)
```

Arguments

myVariables	An R object with variable names of variables that should be included in table, e.g. c('GPP', 'RECO', 'NEE')
myCaption	A string that is used as table caption, e.g. 'Globally averaged statistical metrics'.
inputDir	A string that gives the input directory, e.g. '/home/project/study'.
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.

Value

One table in LaTeX format that shows a subset of statistical metrics

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

myInputDir <- paste(system.file('extdata', package = 'amber'), 'scores', sep = '/')
myVariables <- c('GPP', 'LAI', 'ALBS')
scores.tables.tweak(myVariables = myVariables, inputDir = myInputDir)
```

---

seasonalCycle *Zonal mean plots of model and reference data*

Description

This function plots the mean seasonal cycle and corresponding inter-quartile range of model and reference data.

Usage

```
seasonalCycle(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
  unit.conv.ref, variable.unit, outlier.factor = 1000, plot.width = 6,
  plot.height = 5, outputDir = FALSE, myLevel = 1,
  globalValues = "globalMean")
```

Arguments

long.name	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
nc.mod	A string that gives the path and name of the netcdf file that contains the model output, e.g. '/home/model_gpp.nc'

nc.ref	A string that gives the path and name of the netcdf file that contains the reference data output, e.g. '/home/reference_gpp.nc'
mod.id	A string that identifies the source of the reference data set, e.g. 'CLASSIC'
ref.id	A string that identifies the source of the reference data set, e.g. 'MODIS'
unit.conv.mod	A number that is used as a factor to convert the unit of the model data, e.g. 86400
unit.conv.ref	A number that is used as a factor to convert the unit of the reference data, e.g. 86400
variable.unit	A string that gives the final units using LaTeX notation, e.g. 'gC m$^-2$ day$^-1$'
outlier.factor	A number that is used to define outliers, e.g. 10. Outliers are all values that exceed the interquartile range multiplied by the outlier factor defined here. 180 degrees and FALSE if you want longitudes to range from 0 to 360 degrees
plot.width	Number that gives the plot width, e.g. 8
plot.height	Number that gives the plot height, e.g. 4
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
myLevel	A number that determines what level of the output netCDF file to use. This is relevant for files with multiple levels, which applies to soil data. By default, myLevel is set to 1.
globalValues	Either 'globalMean' or 'globalSum'. If set to 'globalMean', values are averaged across all grid cells. If set to 'globalSum', values are summed up. The sum is weighted by grid cell area.

Value

Three plots in PDF format that give the (a) monthly time series, (b) annual time series, and (c) the climatological mean seasonal cycle of model and reference data. Concerning the latter, thee bold line presents the mean and the shaded area gives the total range caused by interannual variability.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
```

```
library(viridis)
library(xtable)

long.name <- 'Gross primary productivity'
nc.mod <- system.file('extdata/modelRegular', 'gpp_monthly.nc', package = 'amber')
nc.ref <- system.file('extdata/referenceRegular', 'gpp_GBAF_128x64.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'GBAF' # give reference dataset a name
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
unit.conv.ref <- 86400*1000 # optional unit conversion for reference data
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)
outlier.factor <- 1000

seasonalCycle(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit)
```

`seasonalCycleIrreg` *Zonal mean plots of model and reference data on an irregular grid*

Description

This function plots the mean seasonal cycle and corresponding inter-quartile range of model and reference data that are stored on an irregular grid.

Usage

```
seasonalCycleIrreg(long.name, nc.mod, nc.ref, mod.id, ref.id,
  unit.conv.mod, unit.conv.ref, variable.unit, outlier.factor = 1000,
 my.projection = "+proj=ob_tran +o_proj=longlat +o_lon_p=83. +o_lat_p=42.5 +lon_0=263.",
  plot.width = 8, plot.height = 3.8, numCores = 2,
  timeInt = "month", outputDir = FALSE)
```

Arguments

`long.name`	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
`nc.mod`	A string that gives the path and name of the netcdf file that contains the model output, e.g. '/home/model_gpp.nc'
`nc.ref`	A string that gives the path and name of the netcdf file that contains the reference data output, e.g. '/home/reference_gpp.nc'
`mod.id`	A string that identifies the source of the reference data set, e.g. 'CLASSIC'
`ref.id`	A string that identifies the source of the reference data set, e.g. 'MODIS'
`unit.conv.mod`	A number that is used as a factor to convert the unit of the model data, e.g. 86400
`unit.conv.ref`	A number that is used as a factor to convert the unit of the reference data, e.g. 86400

variable.unit	A string that gives the final units using LaTeX notation, e.g. 'gC m$^-2$ day$^-1$'
outlier.factor	A number that is used to define outliers, e.g. 10. Plotting raster objects that contain extreme outliers lead to figures where most grid cells are presented by a single color since the color legend covers the entire range of values. To avoid this, the user may define outliers that will be masked out and marked with a red dot. Outliers are all values that exceed the interquartile range multiplied by the outlier factor defined here.
my.projection	A string that defines the projection of the irregular grid
plot.width	Number that gives the plot width, e.g. 8
plot.height	Number that gives the plot height, e.g. 4
numCores	An integer that defines the number of cores, e.g. 2
timeInt	A string that gives the time interval of the model data, e.g. 'month' or 'year'
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.

Value

A plot in PDF format that gives the (a) global mean time series and (b) the climatological mean seasonal cycle of model and reference data. The bold line presents the spatial mean values and the shaded area gives the total range caused by interannual variability.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

long.name <- 'Gross primary productivity'
nc.mod <- system.file('extdata/modelRotated', 'gpp_monthly.nc', package = 'amber')
nc.ref <- system.file('extdata/referenceRotated', 'gpp_GBAF_rotated.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'GBAF' # give reference dataset a name
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
```

```
unit.conv.ref <- 86400*1000 # optional unit conversion for reference data
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)

seasonalCycleIrreg(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit)
 #donttest
```

---

zonalMean *Zonal mean plots of model and reference data*

Description

This function plots zonal mean annual means and corresponding total range for model and reference data.

Usage

```
zonalMean(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
  unit.conv.ref, variable.unit, outlier.factor = 1000, plot.width = 6,
  plot.height = 5, outputDir = FALSE, myLevel = 1, subcaption = "")
```

Arguments

long.name	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
nc.mod	A string that gives the path and name of the netcdf file that contains the model output, e.g. '/home/model_gpp.nc'
nc.ref	A string that gives the path and name of the netcdf file that contains the reference data output, e.g. '/home/reference_gpp.nc'
mod.id	A string that identifies the source of the reference data set, e.g. 'CLASSIC'
ref.id	A string that identifies the source of the reference data set, e.g. 'MODIS'
unit.conv.mod	A number that is used as a factor to convert the unit of the model data, e.g. 86400
unit.conv.ref	A number that is used as a factor to convert the unit of the reference data, e.g. 86400
variable.unit	A string that gives the final units using LaTeX notation, e.g. 'gC m$^-2$ day$^-1$'
outlier.factor	A number that is used to define outliers, e.g. 10. Plotting raster objects that contain extreme outliers lead to figures where most grid cells are presented by a single color since the color legend covers the entire range of values. To avoid this, the user may define outliers that will be masked out and marked with a red dot. Outliers are all values that exceed the interquartile range multiplied by the outlier factor defined here.
plot.width	Number that gives the plot width, e.g. 8

plot.height	Number that gives the plot height, e.g. 4
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.
myLevel	A number that determines what level of the output netCDF file to use. This is relevant for files with multiple levels, which applies to soil data. By default, myLevel is set to 1.
subcaption	A string that defines the subcaption of the figure, e.g. '(a)'.

Value

A figure in PDF format that gives the zonal mean values of model and reference data. The bold line presents the mean and the shaded area the corresponding interquartile range (IQR). The IQR presents the inter-annual variability and longitudinal variability, combined.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

long.name <- 'Gross primary productivity'
nc.mod <- system.file('extdata/modelRegular', 'gpp_monthly.nc', package = 'amber')
nc.ref <- system.file('extdata/referenceRegular', 'gpp_GBAF_128x64.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'GBAF' # give reference dataset a name
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
unit.conv.ref <- 86400*1000 # optional unit conversion for reference data
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)
outlier.factor <- 1000

zonalMean(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit, outlier.factor)
```

zonalMeanIrreg *Zonal mean plots of model and reference data on an irregular grid*

Description

This function plots zonal mean values and corresponding inter-quartile ranges of model and reference data. The function expects data to be on an irregular grid.

Usage

```
zonalMeanIrreg(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
  unit.conv.ref, variable.unit, outlier.factor = 1000,
 my.projection = "+proj=ob_tran +o_proj=longlat +o_lon_p=83. +o_lat_p=42.5 +lon_0=263.",
  plot.width = 8, plot.height = 3.8, numCores = 2,
  timeInt = "month", outputDir = FALSE)
```

Arguments

long.name	A string that gives the full name of the variable, e.g. 'Gross primary productivity'
nc.mod	A string that gives the path and name of the netcdf file that contains the model output, e.g. '/home/model_gpp.nc'
nc.ref	A string that gives the path and name of the netcdf file that contains the reference data output, e.g. '/home/reference_gpp.nc'
mod.id	A string that identifies the source of the reference data set, e.g. 'CLASSIC'
ref.id	A string that identifies the source of the reference data set, e.g. 'MODIS'
unit.conv.mod	A number that is used as a factor to convert the unit of the model data, e.g. 86400
unit.conv.ref	A number that is used as a factor to convert the unit of the reference data, e.g. 86400
variable.unit	A string that gives the final units using LaTeX notation, e.g. 'gC m$^-2$ day$^-1$'
outlier.factor	A number that is used to define outliers, e.g. 10. Plotting raster objects that contain extreme outliers lead to figures where most grid cells are presented by a single color since the color legend covers the entire range of values. To avoid this, the user may define outliers that will be masked out and marked with a red dot. Outliers are all values that exceed the interquartile range multiplied by the outlier factor defined here.
my.projection	A string that defines the projection of the irregular grid
plot.width	Number that gives the plot width, e.g. 8
plot.height	Number that gives the plot height, e.g. 4
numCores	An integer that defines the number of cores, e.g. 2
timeInt	A string that gives the time interval of the model data, e.g. 'month' or 'year'
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.

Value

A figure in PDF format that gives the zonal mean values of model and reference data. The bold line presents the mean and the shaded area the corresponding interquartile range (IQR). The IQR presents the inter-annual variability and longitudinal variability, combined.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

long.name <- 'Gross primary productivity'
nc.mod <- system.file('extdata/modelRotated', 'gpp_monthly.nc', package = 'amber')
nc.ref <- system.file('extdata/referenceRotated', 'gpp_GBAF_rotated.nc', package = 'amber')
mod.id <- 'CLASSIC' # define a model experiment ID
ref.id <- 'GBAF' # give reference dataset a name
unit.conv.mod <- 86400*1000 # optional unit conversion for model data
unit.conv.ref <- 86400*1000 # optional unit conversion for reference data
variable.unit <- 'gC m$^{-2}$ day$^{-1}$' # unit after conversion (LaTeX notation)
outlier.factor <- 1000

zonalMeanIrreg(long.name, nc.mod, nc.ref, mod.id, ref.id, unit.conv.mod,
unit.conv.ref, variable.unit, outlier.factor, numCores = 2)
 #donttest
```

`zonalMeanStats` *Zonal mean plots of AMBER results (bias, bias scores, etc)*

Description

This function computes zonal mean values of model and reference data and the zonal mean bias, centralized root-mean-squre error, phase, inter-annual variability, and corresponding scores. The computation is based on the NetCDF files produced by scores.grid.time.

Usage

```
zonalMeanStats(inputDir, outputDir = FALSE)
```

Arguments

inputDir	A string that gives the location of NetCDF files produced by scores.grid.time and scores.grid.notime, e.g. '/home/project/study'.
outputDir	A string that gives the output directory, e.g. '/home/project/study'. The output will only be written if the user specifies an output directory.

Value

A list with two tables. The first table gives the zonal mean values of the mean, bias, standard deviation, centralized root-mean-squre error, month of seasonal peak, phase, inter-annual variability, and corresponding scores for each variable and globally gridded reference data set. The second table gives the physical units of each variable in LaTeX notation (e.g. 'W m$^-2$'). Both tables are written to two text files (zonalMeanStats and zonalMeanStatsUnits) if the user specifies an output directory.

Examples

```
library(amber)
library(classInt)
library(doParallel)
library(foreach)
library(Hmisc)
library(latex2exp)
library(ncdf4)
library(parallel)
library(raster)
library(rgdal)
library(rgeos)
library(scico)
library(sp)
library(stats)
library(utils)
library(viridis)
library(xtable)

inputDir <- paste(system.file('extdata', package = 'amber'), 'zonalMeanStats', sep = '/')
zonalMeanStats(inputDir, outputDir = FALSE)
```

Index