Package 'sdmpredictors'

August 23, 2023

Type Package

Title Species Distribution Modelling Predictor Datasets

Version 0.2.15

Date 2023-08-11

Depends R (>= 3.2.5)

Imports R.utils (>= 2.4.0), stats, utils, raster, terra

Description Terrestrial and marine predictors for species distribution modelling
from multiple sources, including WorldClim <https://www.worldclim.org/>,,
ENVIREM <https://envirem.github.io/>, Bio-ORACLE <https://bio-oracle.org/>
and MARSPEC <http://www.marspec.org/>.

Encoding UTF-8

License MIT + file LICENSE

URL http://lifewatch.github.io/sdmpredictors/

BugReports https://github.com/lifewatch/sdmpredictors/issues

Suggests ggplot2, reshape2, testthat, knitr, rmarkdown, httr, httr2

RoxygenNote 7.2.3

VignetteBuilder knitr

NeedsCompilation no

Author Samuel Bosch [aut],
Lennert Tyberghein [ctb],
Olivier De Clerck [ctb],
Salvador Fernandez [aut, cre],
Lennert Schepers [ctb],
LifeWatch Belgium [fnd] (https://lifewatch.be)

Maintainer Salvador Fernandez <salvador.fernandez@vliz.be>

Repository CRAN

Date/Publication 2023-08-23 15:50:09 UTC

R topics documented:

---

calculate_statistics   *Calculate statistics for a given raster.*

Description

Method used to calculate the statistics of all layers. It can be re-used to calculate statistics for a cropped version of the rasters.

Usage

```
calculate_statistics(layercode, raster)
```

Arguments

layercode	character. Name of the layer.
raster	RasterLayer. The raster you want to calculate statistics for.

Value

A dataframe with the layercode and all basic statistics.

See Also

[layer_stats](#)

Examples

```
## Not run:
# calculate statistics of the SST and salinity in the Baltic Sea

# warning using tempdir() implies that data will be downloaded again in the
# next R session
x <- load_layers(c("BO_sstmax", "BO_salinity"), datadir = tempdir())
e <- extent(13, 31, 52, 66)
baltics <- crop(x, e)
View(rbind(calculate_statistics("SST Baltic Sea", raster(x, layer = 1)))
            calculate_statistics("Salinity Baltic Sea", raster(x, layer = 2)))

## End(Not run)
```

---

correlation_groups   *Groups layers based on the Pearson correlation*

Description

`correlation_groups` returns groups of layer codes such as each layer from one group has an absolute Pearson product-moment correlation coefficient (Pearson's r) that is smaller than the maximum_correlation (default 0.7) with each variable in any other group. The correlation values of quadratic layers are used for creating the groups but only non quadratic layer codes are returned.

Usage

```
correlation_groups(layers_correlation, max_correlation=0.7)
```

Arguments

layers_correlation

matrix or dataframe. A square matrix with the layers correlations you want to group.

max_correlation

number. The maximum correlation 2 layers may have before they are put in the same correlation group.

Value

A list of vectors with each vector containing the layer codes of one correlation group.

References

Dormann, C. F., Elith, J., Bacher, S., Buchmann, C., Carl, G., Carre, G., ... Lautenbach, S. (2013). Collinearity: a review of methods to deal with it and a simulation study evaluating their performance. Ecography, 36(1), 027-046. doi:10.1111/j.1600-0587.2012.07348.x Barbet-Massin, M. & Jetz, W. (2014). A 40-year, continent-wide, multispecies assessment of relevant climate predictors for species distribution modelling. Diversity and Distributions, 20(11), 1285-1295. doi:10.1111/ddi.12229

See Also

    layers_correlation list_layers layer_stats

Examples

```
corr <- layers_correlation(c("BO_calcite", "BO_damin", "MS_bathy_5m"))
print(corr)
print(correlation_groups(corr, max_correlation=0.6))
```

---

  dataset_citations   *Generate dataset citations*

Description

    dataset_citations returns dataset citations as text or as "bibentry" objects.

Usage

```
dataset_citations(datasets = c(), astext = TRUE)
```

Arguments

datasets	character vector. Code of the datasets. When no datasets are provided (default), then all citations are returned.
astext	logical. When TRUE (default), then citations are returned as text otherwise they are returned as "bibentry" objects.

Details

    Note that in order to generate a full list of citations it is preferable to run the layer_citations function.

Value

    Either a character vector or a list of "bibentry" objects.

See Also

    layer_citations, bibentry, list_datasets

Examples

```
# print the Bio-ORACLE citation
print(dataset_citations("Bio-ORACLE"))

# print all citations as Bibtex
print(lapply(dataset_citations(astext = FALSE), toBibtex))
```

---

equalareaproj	*World Behrmann equal area coordinate reference system (ESRI:54017), used when using load_layers with equal_area = TRUE*

Description

World Behrmann equal area coordinate reference system (ESRI:54017), used when using load_layers with equal_area = TRUE

Usage

```
equalareaproj
```

Format

An object of class CRS of length 1.

---

get_future_layers	*Get the name of future climate layer(s) based on the current climate layer(s)*

Description

`get_future_layers` returns information on the future climate layers for the matching current climate layers.

Usage

```
get_future_layers(current_layer_codes, scenario, year)
```

Arguments

current_layer_codes

> character vector. Code(s) of the current climate layers either as a character vector or as the dataframe provided by list_layers.

scenario   character vector. Climate change scenario, e.g. ″B1″,″A1B″, ″A2″.

year   integer. Year for which you want the climate change prediction, e.g. 2100, 2200.

Details

Stops with an exception if no matching future climate layer was found for one or more of the provided current climate layer codes.

Value

A dataframe with information on the future layer(s) matching the provided current layer(s).

See Also

[list_layers_future](), [list_layers](), [load_layers]()

Examples

```
future_layers <- get_future_layers(c("BO_salinity", "BO_sstmean"),
                                    scenario = "B1", year = 2100)
future_layers$layer_code
```

---

get_layers_info *Layer info for specific layer codes*

Description

`get_layers_info` returns all detailed information on the current or future climate layers of one or more datasets.

Usage

```
get_layers_info(layer_codes = c())
```

Arguments

layer_codes   character vector. Vector with the layer codes of the layers you want the full
               information for. This can also be a dataframe with as column `layer_code`.

Value

A list with four dataframes `common`, `current`, `future` and `paleo`, the `common` dataframe contains data for all shared columns in the other three dataframes. The other dataframes contain all detailed information on the layer(s) matching the layer codes. By default information for all layers is returned.

See Also

[list_layers](), [list_layers_future](), [list_layers_paleo](), [load_layers]()

Examples

```
info <- get_layers_info(c("BO_salinity", "BO_B1_2100_salinity"))
info$common
info$current
info$future
info$paleo
```

get_paleo_layers	*Get the name of paleo climate layer(s) based on the current climate layer(s)*

Description

get_paleo_layers returns information on the future climate layers for the matching current climate layers.

Usage

```
get_paleo_layers(current_layer_codes, model_name = NA, epoch = NA,
  years_ago = NA)
```

Arguments

current_layer_codes

        character vector. Code(s) of the current climate layers either as a character vector or as the dataframe provided by list_layers.

model_name   character vector. Paleo climate model name see the model_name column in the result from list_layers_paleo.

epoch   character vector. Epoch for which you want the paleo layer, e.g. "mid-Holocene", "Last Glacial Maximum".

years_ago   integer. Years for which you want the paleo layer, e.g. 6000, 21000.

Details

Stops with an exception if no matching paleo layer was found for one or more of the provided current climate layer codes.

Value

A dataframe with information on the paleo layer(s) matching the provided current layer(s).

See Also

list_layers_paleo, list_layers, load_layers

Examples

```
paleo_layers <- get_paleo_layers("MS_biogeo08_sss_mean_5m", years_ago = 6000)
paleo_layers$layer_code
```

---

layers_correlation   *Gives the Pearson correlation between layers*

Description

`layers_correlations` returns the Pearson product-moment correlation coefficient (Pearson's r) for every combination of the give layercodes. The correlation between a terrestrial and a marine layer has been set to NA.

Usage

```
layers_correlation(layercodes = c())
```

Arguments

layercodes   character vector or dataframe. Codes of the layers, you want the correlation
                matrix of, as a character vector or a dataframe with a "layer_code" column.
                With the default empty vector the correlation between all layers is returned.

Value

A dataframe with the Pearson product-moment correlation coefficients.

See Also

[list_layers](#) [layer_stats](#) [correlation_groups](#) [plot_correlation](#)

Examples

```
# correlation of the first 10 layers
layers_correlation()[1:10,1:10]
layers_correlation(c("BO_calcite", "MS_bathy_5m"))
layers_correlation(c("BO_calcite", "MS_bathy_5m"))
```

---

layer_citations   *Generate citations for all layers*

Description

`layer_citations` returns layer citations as text or as ["bibentry"](#) objects.

Usage

```
layer_citations(layers = c(), astext = TRUE)
```

Arguments

layers	character vector. Code of the layers from past, current and future climate layers. When no layers are provided (default), then all citations are returned.
astext	logical. When TRUE (default), then citations are returned as text otherwise they are returned as "bibentry" objects.

Details

Note that for some layers multiple references are returned as some of the predictors have been published separately.

Value

Either a character vector or a list of "bibentry" objects.

See Also

layer_citations, bibentry, list_datasets

Examples

```
# print the citation for the Bio-ORACLE salinity layer
print(layer_citations("BO_salinity"))

# print the citation for a MARSPEC paleo layer
print(layer_citations("MS_biogeo02_aspect_NS_21kya"))

# print all citations as Bibtex
print(lapply(layer_citations(astext = FALSE), toBibtex))
```

---

layer_stats   *Gives basic layer statistics*

Description

layer_stats returns basic statistics (minimum, q1, median, q3, maximum, median absolute deviation (mad), mean, standard deviation (sd)) for each given layercode.

Usage

```
layer_stats(layercodes = c())
```

Arguments

layercodes	character vector or dataframe. Codes of the layers you want the basic statistics of as a character vector or a dataframe with a "layer_code" column. With the default empty vector all statistics are returned.

Value

A dataframe with basic statistics about each given layercode.

See Also

[list_layers](#) [layers_correlation](#)

Examples

```
# layer stats for the first 10 layers
layer_stats()[1:10,]
layer_stats(c("BO_calcite", "MS_bathy_5m"))
```

---

list_datasets   *Lists the supported datasets*

Description

list_datasets returns information on the supported datasets.

Usage

```
list_datasets(terrestrial = NA, marine = NA, freshwater = NA)
```

Arguments

terrestrial	logical. When TRUE, then datasets that only have terrestrial data (seamasked) are returned.
marine	logical. When TRUE, then datasets that only have marine data (landmasked) are returned.
freshwater	logical. When TRUE, then datasets that only have freshwater data are returned.

Details

By default it returns all datasets, when both marine, freshwater and terrestrial are FALSE then only datasets without land- nor seamasks are returned.

Value

A dataframe with information on the supported datasets.

See Also

[list_layers](#), [list_layers_future](#), [list_layers_paleo](#)

Examples

```
list_datasets()
list_datasets(marine=TRUE)
list_datasets(terrestrial=TRUE)
```

---

list_layers   *List the current climate layers provided by one or more datasets*

Description

list_layers returns information on the layers of one or more datasets.

Usage

```
list_layers(datasets=c(), terrestrial = NA, marine = NA, freshwater =
  NA, monthly = TRUE, version = NULL)
```

Arguments

datasets	character vector. Code of the datasets.
terrestrial	logical. When TRUE (default), then datasets that only have terrestrial data (sea-masked) are returned.
marine	logical. When TRUE (default), then datasets that only have marine data (land-masked) are returned.
freshwater	logical. When TRUE, then datasets that only have freshwater data are returned.
monthly	logical. When FALSE, then no monthly layers are returned. All annual and monthly layers are returned by default.
version	numeric vector. When NULL then layers from all versions of datasets are returned (default) else layers are filtered by version number.

Details

By default it returns all layers from all datasets, when both marine and terrestrial are FALSE then only layers from datasets without land- nor seamasks are returned. Layers for paleo and future climatic conditions can be listed with list_layers_paleo and list_layers_future. Available paleo and future climate layers for a current climate layer code can be listed with the functions get_paleo_layers and get_future_layers.

Value

A dataframe with information on the supported current climate layers.

See Also

load_layers, list_datasets, list_layers_future, list_layers_paleo, get_future_layers, get_paleo_layers

Examples

```
# list the first 5 layers
list_layers()[1:5,]
# list the layercodes all monthly layers from WorldClim
worldclim <- list_layers("WorldClim")
worldclim[!is.na(worldclim$month),]$layer_code
# list layer codes for Bio-ORACLE and MARSPEC
list_layers(c("Bio-ORACLE","MARSPEC"))$layer_code
# list the first 5 terrestrial layers
list_layers(marine=FALSE)[1:5,]
# list the first 5 marine layers
list_layers(terrestrial=FALSE)[1:5,]
# list all annual MARSPEC layers (remove monthly layers)
list_layers("MARSPEC", monthly = FALSE)
```

---

list_layers_future   *List the future climate layers provided by one or more datasets*

Description

list_layers_future returns information on the future climate layers of one or more datasets.

Usage

```
list_layers_future(datasets = c(), scenario = NA, year = NA,
  terrestrial = NA, marine = NA, freshwater = NA, monthly = TRUE, version =
  NULL)
```

Arguments

datasets	character vector. Code of the datasets.
scenario	character vector. Climate change scenario, e.g. "B1","A1B", "A2".
year	integer. Year for which you want the climate change prediction, e.g. 2100, 2200.
terrestrial	logical. When TRUE (default), then datasets that only have terrestrial data (sea-masked) are returned.
marine	logical. When TRUE (default), then datasets that only have marine data (land-masked) are returned.
freshwater	logical. When TRUE, then datasets that only have freshwater data are returned.
monthly	logical. When FALSE, then no monthly layers are returned. All annual and monthly layers are returned by default.
version	numeric vector. When NULL then layers from all versions of datasets are returned (default) else layers are filtered by version number.

Details

By default it returns all layers from all datasets, when both marine and terrestrial are FALSE then only layers without land- nor seamasks are returned.

Value

A dataframe with information on the supported future climate layers.

See Also

list_layers, list_layers_paleo, list_datasets, load_layers

Examples

```
# list the first 5 layers
list_layers_future()[1:5,]
# list layer codes for Bio-ORACLE with scenario B1 and year 2100
list_layers_future("Bio-ORACLE", scenario = "B1", year = 2100)$layer_code
```

---

list_layers_paleo   *List the paleo climate layers provided by one or more datasets*

Description

list_layers_paleo returns information on the paleo climate layers of one or more datasets.

Usage

```
list_layers_paleo(datasets = c(), model_name = NA, epoch = NA,
  years_ago = NA, terrestrial = NA, marine = NA, freshwater = NA, monthly =
  TRUE, version = NULL)
```

Arguments

datasets	character vector. Code of the datasets.
model_name	character vector. Paleo climate model name see the model_name column in the result.
epoch	character vector. Epoch for which you want the paleo layer, e.g. "mid-Holocene", "Last Glacial Maximum".
years_ago	integer. Years for which you want the paleo layer, e.g. 6000, 21000.
terrestrial	logical. When TRUE (default), then datasets that only have terrestrial data (sea-masked) are returned.
marine	logical. When TRUE (default), then datasets that only have marine data (land-masked) are returned.
freshwater	logical. When TRUE, then datasets that only have freshwater data are returned.

monthly	logical. When `FALSE`, then no monthly layers are returned. All annual and monthly layers are returned by default.
version	numeric vector. When `NULL` then layers from all versions of datasets are returned (default) else layers are filtered by version number.

Details

By default it returns all layers from all datasets, when both marine and terrestrial are `FALSE` then only layers without land- nor seamasks are returned.

Value

A dataframe with information on the supported paleo climate layers.

See Also

[list_layers](#), [list_layers_future](#), [list_datasets](#), [load_layers](#)

Examples

```
# list the first 5 layers
list_layers_paleo()[1:5,]
# list layer codes for MARSPEC for the mid-Holocene
list_layers_paleo("MARSPEC", epoch = "mid-Holocene")$layer_code
```

---

load_layers   *Load layers*

Description

Method to load rasters from disk or from the internet. By default a RasterStack is returned but this is only possible When all rasters have the same spatial extent and resolution.

Usage

```
load_layers(layercodes, equalarea=FALSE, rasterstack=TRUE,
  datadir=NULL)
```

Arguments

layercodes	character vector or dataframe. Layer_codes of the layers to be loaded or dataframe with a "layer_code" column.
equalarea	logical. If `TRUE` then layers are loaded with a Behrmann cylindrical equal-area projection ([equalareaproj](#)), otherwise unprojected ([lonlatproj](#)). Default is `FALSE`.
rasterstack	logical. If `TRUE` (default value) then the result is a [stack](#) otherwise a list of rasters is returned.

datadir   character. Directory where you want to store the data. If `NULL` is passed (default)
                 then the sdmpredictors_datadir option is read. To set this run `options(sdmpredictors_datadir="<`
                 `preferred directory>")` in every session or add it to your .RProfile.

Value

RasterStack or list of raster

See Also

[list_layers](), [layer_stats](), [layers_correlation]()

Examples

```
## Not run:
# warning using tempdir() implies that data will be downloaded again in the
# next R session
env <- load_layers("BO_calcite", datadir = tempdir())

## End(Not run)
```

---

lonlatproj   *Longitude/latitude coordinate reference system (EPSG:4326), used*
                          when using load_layers with equal_area = FALSE

Description

Longitude/latitude coordinate reference system (EPSG:4326), used when using load_layers with
equal_area = FALSE

Usage

```
lonlatproj
```

Format

An object of class `CRS` of length 1.

---

pearson_correlation_matrix

Calculate the Pearson correlation coefficient matrix for a rasterstack

---

Description

Calculate the Pearson correlation coefficient matrix for a rasterstack

Usage

```
pearson_correlation_matrix(x, cachesize = 20, same_mask = FALSE)
```

Arguments

x	RasterStack. The stack of rasters you want to calculate the Pearson correlation coefficient matrix for. This can be obtained by calling load_layers.
cachesize	integer. For how many rasters should the values be kept in local memory. By default this is set to 20, a parameter which works reasonably well on a windows computer with 8GB RAM.
same_mask	logical. Whether we can assume that the mask is the same for all layers (same NA values), default is FALSE.

Value

A correlation matrix.

See Also

layers_correlation plot_correlation load_layers

Examples

```
## Not run:
# calculate correlation between SST and salinity in the Baltic Sea

# warning using tempdir() implies that data will be downloaded again in the
# next R session
x <- load_layers(c("BO_sstmax", "BO_salinity"), datadir = tempdir())
e <- extent(13, 31, 52, 66)
baltics <- crop(x, e)
print(pearson_correlation_matrix(baltics))

## End(Not run)
```

---

plot_correlation	*Plot the correlation matrix for the provided layercodes*

Description

#' plot_correlation creates a plot of the correlation between different layers

Usage

```
plot_correlation(layers_correlation, prettynames = list(),
  palette = c("#2c7bb6", "#abd9e9", "#ffffbf", "#fdae61", "#d7191c"))
```

Arguments

layers_correlation

> matrix or dataframe. A square matrix with the layers correlations you want to plot as returned by layers_correlation or pearson_correlation_matrix.

prettynames
> list. Optional list with as names the layercodes and as values the name of the layer to be used in the plot.

palette
> character vector. optional vector with 5 entries for the range of colors to be used for the correlation matrix plot.

Details

This function requires ggplot2 and plots the correlations for the layers in the same order as the layercodes are provided to this function.

Value

A ggplot object that can be printed or saved.

See Also

layers_correlation pearson_correlation_matrix list_layers layer_stats correlation_groups

Examples

```
correlation <- layers_correlation(c("BO_calcite", "BO_damin", "MS_bathy_5m"))
p <- plot_correlation(correlation)
print(p)
```

---

sdmpredictors   *sdmpredictors: Species Distribution Modeling Predictor Datasets*

Description

Terrestrial and marine predictors for species distribution modeling from multiple sources, including WorldClim (<https://www.worldclim.org/>), Bio-ORACLE (<https://bio-oracle.org/>) and MARSPEC (<http://www.marspec.org/>).

References

WorldClim: Hijmans, R.J., S.E. Cameron, J.L. Parra, P.G. Jones and A. Jarvis (2005). Very high resolution interpolated climate surfaces for global land areas. International Journal of Climatology 25: 1965-1978. (doi:10.1002/joc.1276)

ENVIREM: Title, P. O. and Bemmels, J. B. (2017). envirem: An expanded set of bioclimatic and topographic variables increases flexibility and improves performance of ecological niche modeling. Ecography (Cop.). in press. (doi:10.1111/ecog.02880)

Bio-ORACLE: Tyberghein L., Verbruggen H., Pauly K., Troupin C., Mineur F. & De Clerck O. (2012) Bio-ORACLE: a global environmental dataset for marine species distribution modeling. Global Ecology and Biogeography (doi:10.1111/j.14668238.2011.00656.x).

Assis, J., Tyberghein, L., Bosh, S., Verbruggen, H., Serrao, E. A., & De Clerck, O. (2017). Bio-ORACLE v2.0: Extending marine data layers for bioclimatic modelling. Global Ecology and Biogeography. (doi:10.1111/geb.12693)

MARSPEC: Sbrocco, EJ and Barber, PH (2013) MARSPEC: Ocean climate layers for marine spatial ecology. Ecology 94: 979. (doi:10.1890/121358.1)

---

sdm_to_bo   *Translate file name from sdmpredictors to bio-oracle*

Description

Translate file name from sdmpredictors to bio-oracle

Usage

```
sdm_to_bo(sdm)
```

Arguments

sdm   the name of the sdm string

Value

A string with the name of the same file in bio-oracle

Examples

```
sdm_to_bo("BO2_tempmean_ss.tif")
sdm_to_bo("BO21_RCP26_2050_curvelltmax_bdmax.tif")
```

Index