Package ‘aws.wrfsmn’

March 9, 2024

Type Package

Title Data Processing of SMN Hi-Res Weather Forecast from 'AWS'

Version 0.0.3

Description Exploration of Weather Research & Forecasting ('WRF') Model data of Servicio Meteorologico Nacional (SMN) from Amazon Web Services (<https://registry.opendata.aws/smn-ar-wrf-dataset/>) cloud. The package provides the possibility of data downloading, processing and correction methods. It also has map management and series exploration of available meteorological variables of 'WRF' forecast.

License GPL (>= 3)

Depends R (>= 4.1.0)

Imports aws.s3 (>= 0.3.21), lubridate (>= 1.9.3), terra (>= 1.7-65), dplyr (>= 1.1.4), ggplot2 (>= 3.4.4), hydroGOF (>= 0.5-4), stats (>= 4.1.2), magrittr (>= 2.0.3)

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation no

Author Gonzalo Diaz [cre, aut]

Maintainer Gonzalo Diaz <gonzalomartindiaz22@gmail.com>

Repository CRAN

Date/Publication 2024-03-09 01:20:07 UTC

R topics documented:

`eva` *Evaporation data (observation and model)*

Description

Data of evaporation from in-situ observation and several soil model outputs

Usage

```
data(eva)
```

Format

An object of class `"data.frame"`.

Dates 1st column with dates

evapo_obs 2nd column with evaporation observation

OUT_PREC Precipitation

OUT_EVAP Evaporation

OUT_RUNOFF Runoff

OUT_BASEFLOW Baseflow

OUT_SOIL_MOIST_lyr_1 Soil moisture from 1st layer

OUT_EVAP_CANOP Evaporation from canopy

OUT_SURF_TEMP Surface temperature

References

Diaz et al. (2024) AAGG 2024 Not yet published

Examples

```
data(eva)
```

`find.nearest.point` *Temporal series of closest location*

Description

Location of nearest point to lon/lat and temporal serie of location

Usage

```
find.nearest.point(data.spat.raster = data.spat.raster, lon = lon, lat = lat)
```

Arguments

`data.spat.raster`	Spat Raster of WRF SMN (only one or several)
`lon`	Longitude location of nearest point to find
`lat`	Latitude location of nearest point to find

Value

a vector with the nearest location (lon/lat) and time serie of that location

`get.wrf.files` *List of available files for downloading*

Description

Character string with the filenames of WRF SMN located in AWS Bucket

Usage

```
get.wrf.files(year = year, month = month, day = day, cycle = cycle, time = time)
```

Arguments

`year`	character or numeric of year
`month`	character or numeric of month
`day`	character or numeric of day
`cycle`	cycle of forecast, "00", "06", "12" or "18"
`time`	selection of datasets, "01H", "24H" or "10M"

Value

string of the list of elements in the Bucket

load.by.variable *Load and projection of data*

Description

Open of netcdf files of WRF SMN drom AWS and optional projection

Usage

```
load.by.variable(nc.filenames, variable, transform, method)
```

Arguments

nc.filenames	netcdf files
variable	name of variable from https://odp-aws-smn.github.io/documentation_wrf_det/Formato_de_datos/ as character
transform	TRUE to project data to longlat datum=WGS84
method	if transform is set TRUE define projection method taken from project function of terra package

Value

Spat Raster

mg.evaluation *Evaluation of regression*

Description

Evaluation of the linear multiple regression obtained from the multiple.guidance function

Usage

```
mg.evaluation(
  input.data = input.data,
  predictand = predictand,
  predictors = predictors,
  var.model = var.model,
  lmodel = lmodel
)
```

Arguments

`input.data` Data frame with first column as a "POSIXct" class and one or more columns with data values. The predictand and predictors variables should be located in these columns

`predictand` Character with column name of the predictand variable

`predictors` Character array with one or more elements of the predictors chosen by the user

`var.model` Character with column name of the modeled predicting variable

`lmodel` Element of class lm obtained from multiple.guidance output function

Value

List of two elements. First element is a matrix with the columns of observed data, modeled data and corrected data. Second element is a data frame of the statistical results of the modeled and corrected data versus observed data

`multiple.guidance` *Multiple lineal regression of data*

Description

Definition of linear multiple regression adjustment based on predictor variables that fit a predicting variable

Usage

```
multiple.guidance(
  input.data = input.data,
  predictand = predictand,
  predictors = predictors
)
```

Arguments

`input.data` Data frame with first column as a "POSIXct" class and one or more columns with data values. The predictand and predictors variables should be located in these columns

`predictand` Character with column name of the predictand variable

`predictors` Character array with one or more elements of the predictors chosen by the user

Value

an element of class lm

`ploting` *Plot of data*

Description

Plot of observed, modeled and corrected guidance series

Usage

```
ploting(data = data)
```

Arguments

`data`	Data frame from daily2monthly output function or any other temporal series

Value

ggplot element

`wrf.download` *Download of wrf files*

Description

Download of WRF SMN data from AWS

Usage

```
wrf.download(wrf.name = wrf.name)
```

Arguments

`wrf.name`	list of names to download from get.wrf.files. e.g.: "DATA/WRF/DET/2024/01/01/18/WRFDETAR_24H_

Value

downloaded netcdf files

%>% *Daily data to monthly*

Description

Data transformation from daily to monthly scale

Usage

```
daily2monthly(data = data)
```

Arguments

`data`	matrix with daily data from mg.evaluation output function

Value

Data frame with monthly data

Index