Package: ShapleyOutlier (via r-universe)

September 6, 2024

Type Package

Title Multivariate Outlier Explanations using Shapley Values and Mahalanobis Distances

Version 0.1.1

Depends R (>= 4.0.0)

Maintainer Marcus Mayrhofer <marcus.mayrhofer@tuwien.ac.at>

Description Based on Shapley values to explain multivariate outlyingness and to detect and impute cellwise outliers. Includes implementations of methods described in Mayrhofer and Filzmoser (2022) <doi:10.48550/ARXIV.2210.10063>.

License GPL-3

Imports dplyr, Rdpack, stats, tibble, tidyr, robustbase, forcats, egg, ggplot2, gridExtra, RColorBrewer, magrittr

Suggests grDevices, cellWise, robustHD, knitr, MASS, rmarkdown

RdMacros Rdpack

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

VignetteBuilder knitr

Repository CRAN

NeedsCompilation no

Author Marcus Mayrhofer [aut, cre], Peter Filzmoser [aut]

Date/Publication 2023-02-20 10:30:02 UTC

Contents

`MOE` *Detecting cellwise outliers using Shapley values based on local outlyingness.*

Description

The `MOE` function indicates outlying cells for a data vector with p entries or data matrix with $n \times p$ entries containing only numeric entries `x` for a given center `mu` and covariance matrix `Sigma` using the Shapley value. It is a more sophisticated alternative to the `SCD` algorithm, which uses the information of the regular cells to derive an alternative reference point (Mayrhofer and Filzmoser 2022).

Usage

```
MOE(
  x,
  mu,
  Sigma,
  Sigma_inv = NULL,
  step_size = 0.1,
  min_deviation = 0,
  max_step = NULL,
  local = TRUE,
  max_iter = 1000,
  q = 0.99,
  check_outlyingness = FALSE,
  check = TRUE,
  cells = NULL,
  method = "cellMCD"
)
```

Arguments

`x`	Data vector with p entries or data matrix with $n \times p$ entries containing only numeric entries.

mu	Either NULL (default) or mean vector of x. If NULL, method is used for parameter estimation.
Sigma	Either NULL (default) or covariance matrix $p \times p$ of x. If NULL, method is used for parameter estimation.
Sigma_inv	Either NULL (default) or Sigma's inverse $p \times p$ matrix. If NULL, the inverse of Sigma is computed using solve(Sigma).
step_size	Numeric. Step size for the imputation of outlying cells, with step_size $\in [0, 1]$. Defaults to 0.1.
min_deviation	Numeric. Detection threshold, with min_deviation $\in [0, 1]$. Defaults to 0.2
max_step	Either NULL (default) or an integer. The maximum number of steps in each iteration. If NULL, max_step $= p$.
local	Logical. If TRUE (default), the non-central Chi-Squared distribution is used to determine the cutoff value based on mu_tilde.
max_iter	Integer. The maximum number of iterations.
q	Numeric. The quantile of the Chi-squared distribution for detection and imputation of outliers. Defaults to 0.99.
check_outlyingness	Logical. If TRUE (default), the outlyingness is rechecked after applying min_deviation.
check	Logical. If TRUE (default), inputs are checked before running the function and an error message is returned if one of the inputs is not as expected.
cells	Either NULL (default) or a vector/matrix of the same dimension as x, indicating the outlying cells. The matrix must contain only zeros and ones, or TRUE/FALSE.
method	Either "cellMCD" (default) or "MCD". Specifies the method used for parameter estimation if mu and/or Sigma are not provided.

Value

A list of class shapley_algorithm (new_shapley_algorithm) containing the following:

x	A p-dimensional vector (or a $n \times p$ matrix) containing the imputed data.
phi	A p-dimensional vector (or a $n \times p$ matrix) containing the Shapley values (outlyingness-scores) of x; see shapley.
mu_tilde	A p-dimensional vector (or a $n \times p$ matrix) containing the alternative reference points based on the regular cells of the original observations.
x_original	A p-dimensional vector (or a $n \times p$ matrix) containing the original data.
x_original	The non-centrality parameters for the Chi-Squared distribution
x_history	A list with n elements, each containing the path of how the original data vector was modified.
phi_history	A list with n elements, each containing the Shapley values corresponding to x_history.
mu_tilde_history	A list with n elements, each containing the alternative reference points corresponding to x_history.
S_history	A list with n elements, each containing the indices of the outlying cells in each iteration.

References

Mayrhofer M, Filzmoser P (2022). "Multivariate outlier explanations using Shapley values and Mahalanobis distances." doi:10.48550/ARXIV.2210.10063.

Examples

```
p <- 5
mu <- rep(0,p)
Sigma <- matrix(0.9, p, p); diag(Sigma) = 1
Sigma_inv <- solve(Sigma)
x <- c(0,1,2,2.3,2.5)
MOE_x <- MOE(x = x, mu = mu, Sigma = Sigma)
plot(MOE_x)

library(MASS)
set.seed(1)
n <- 100; p <- 10
mu <- rep(0,p)
Sigma <- matrix(0.9, p, p); diag(Sigma) = 1
X <- mvrnorm(n, mu, Sigma)
X[sample(1:(n*p), 100, FALSE)] <- rep(c(-5,5),50)
MOE_X <- MOE(X, mu, Sigma)
plot(MOE_X, subset = 20)
```

---

new_shapley *Class constructor for class* shapley.

Description

This function creates an object of class shapley that is returned by the shapley function.

Usage

```
new_shapley(phi = numeric(), mu_tilde = NULL, non_centrality = NULL)
```

Arguments

phi	A p-dimensional vector (or a $n \times p$ matrix) containing the Shapley values (outlyingness-scores) of a p-dimensional data vector (or a $n \times p$ data matrix).
mu_tilde	Optional. A p-dimensional vector (or a $n \times p$ matrix) containing the alternative reference points based on the regular cells of the original observations.
non_centrality	Optional. The non-centrality parameters for the Chi-Squared distribution, which are given by mahlanobis(mu_tilde, mu, Sigma).

Value

Named list of class shapley, containing the input parameters.

`new_shapley_algorithm` *Class constructor for class* `shapley_algorithm`.

Description

This function creates an object of class `shapley_algorithm` that is returned by the `SCD` and `MOE` functions.

Usage

```
new_shapley_algorithm(
  x = numeric(),
  phi = numeric(),
  x_original = numeric(),
  mu_tilde = NULL,
  non_centrality = NULL,
  x_history = NULL,
  phi_history = NULL,
  mu_tilde_history = NULL,
  S_history = NULL
)
```

Arguments

`x`	A p-dimensional vector (or a $n \times p$ matrix) containing the imputed data.
`phi`	A p-dimensional vector (or a $n \times p$ matrix) containing the Shapley values (outlyingness-scores) of a p-dimensional data vector (or a $n \times p$ data matrix).
`x_original`	A p-dimensional vector (or a $n \times p$ matrix) containing the original data.
`mu_tilde`	Optional. A p-dimensional vector (or a $n \times p$ matrix) containing the alternative reference points based on the regular cells of the original observations.
`non_centrality`	Optional. The non-centrality parameters for the Chi-Squared distribution, which are given by `mahlanobis(mu_tilde, mu, Sigma)`.
`x_history`	Optional. A list with n elements, each containing the path of how the original data vector was modified.
`phi_history`	Optional. A list with n elements, each containing the Shapley values corresponding to `x_history`.
`mu_tilde_history`	Optional. A list with n elements, each containing the alternative reference points corresponding to `x_history`.
`S_history`	Optional. A list with n elements, each containing the indices of the outlying cells in each iteration.

Value

Named list of class `shapley_algorithm`, containing the input parameters.

new_shapley_interaction *Class constructor for class* shapley_interaction.

Description

This function creates an object of class shapley_interaction that is returned by the shapley_interaction function.

Usage

```
new_shapley_interaction(PHI = numeric())
```

Arguments

PHI	A $p \times p$ matrix containing the decomposition of the squared Mahalanobis distance of a p-dimensional numeric vector into outlyingness scores for pairs of variables.

Value

Matrix of class shapley_interaction, containing input matrix PHI.

plot.shapley *Barplot of Shapley values*

Description

Barplot of Shapley values

Usage

```
## S3 method for class 'shapley'
plot(
  x,
  subset = NULL,
  chi2.q = 0.99,
  abbrev.var = 3,
  abbrev.obs = 10,
  sort.var = FALSE,
  sort.obs = FALSE,
  plot_md = TRUE,
  md_squared = TRUE,
  rotate_x = TRUE,
  ...
)
```

Arguments

x	A list of class shapley.
subset	Either an integer, "chi2", or NULL (default) to select which rows of phi should be displayed. If NULL, all n rows are displayed, for a single integer the subset rows with the highest Mahalanobis distance are displayed, for an integer vector the subset selected rows are displayed, and for "chi2" all outlying rows are displayed (Mahalanobis distance greater than $\sqrt{}$qchisq(chi2.q,p)).
chi2.q	Quantile, only used if subset == "chi2".
abbrev.var	Integer. If abbrev.var > 0, column names are abbreviated using abbreviate with minlenght = abrev.var.
abbrev.obs	Integer. If abbrev.obs > 0, row names are abbreviated using abbreviate with minlenght = abrev.obs.
sort.var	Logical. If TRUE (default), variables are sorted according to the distance
sort.obs	Logical. If TRUE (default), observations are sorted according to their Mahalanobis distance.
plot_md	Logical. If TRUE (default), the Mahalanobis distance will be included in the plot.
md_squared	Logical. If TRUE (default), the squared Mahalanobis distance is plotted otherwise the (not-squared) Mahalanobis distance.
rotate_x	Logical. If TRUE (default), the x-axis labels are rotated.
...	Optional arguments passed to methods.

Value

Returns a barplot that displays the Shapley values (shapley)for each observation and optionally (plot_md = TRUE) includes the squared Mahalanobis distance (black bar) and the corresponding (non-)central chi-square quantile (dotted line).

Examples

```
library(MASS)
set.seed(1)
n <- 100; p <- 10
mu <- rep(0,p)
Sigma <- matrix(0.9, p, p); diag(Sigma) = 1
X <- mvrnorm(n, mu, Sigma)
X_clean <- X
X[sample(1:(n*p), 100, FALSE)] <- rep(c(-5,5),50)
call_shapley <- shapley(X, mu, Sigma)
plot(call_shapley, subset = 1:20)
plot(call_shapley, subset = 5, rotate_x = FALSE)
plot(call_shapley, subset = 5, md_squared = FALSE, rotate_x = FALSE)
```

`plot.shapley_algorithm` *Barplot and tileplot of Shapley values.*

Description

Barplot and tileplot of Shapley values.

Usage

```
## S3 method for class 'shapley_algorithm'
plot(
  x,
  type = "both",
  subset = NULL,
  abbrev.var = FALSE,
  abbrev.obs = FALSE,
  sort.var = FALSE,
  sort.obs = FALSE,
  n_digits = 2,
  rotate_x = TRUE,
  continuous_rowname = FALSE,
  ...
)
```

Arguments

`x`	A list of class `shapley_algorithm`.
`type`	Either `"both"` (default), `"bar"`, or `"cell"`. If `"both"` (default) a barplot and a tileplot are created, otherwise only the selected plot is created.
`subset`	Either an integer, `"chi2"`, or `NULL` (default) to select which rows of `phi` should be displayed. If `NULL`, all n rows are displayed, for a single integer the `subset` rows with the highest Mahalanobis distance are displayed, for an integer vector the `subset` selected rows are displayed, and for `"chi2"` all outlying rows are displayed (Mahalanobis distance greater than $\sqrt{}$`qchisq(chi2.q,p)`).
`abbrev.var`	Integer. If `abbrev.var` > 0, column names are abbreviated using abbreviate with `minlenght = abrev.var`.
`abbrev.obs`	Integer. If `abbrev.obs` > 0, row names are abbreviated using abbreviate with `minlenght = abrev.obs`.
`sort.var`	Logical. If `TRUE` (default), variables are sorted according to the distance
`sort.obs`	Logical. If `TRUE` (default), observations are sorted according to their Mahalanobis distance.
`n_digits`	Integer. If `n_digits`> 0, the original values of the variables are given in each cell with `n_digits` decimals places.

`rotate_x` Logical. If `TRUE` (default), the x-axis labels are rotated.

`continuous_rowname` Logical. If `TRUE`, the rownames are converted to a numeric vector.

`...` Arguments passed on to `plot.shapley`.

Value

Returns plots for a list of class `shapley_algorithm`. If `type` is `"bar"`, a barplot is generated. It displays the Shapley values (`shapley`) for each observation and optionally (`plot_md = TRUE`) includes the squared Mahalanobis distance (black bar) and the corresponding (non-)central chi-square quantile (dotted line). If `type` is `"cell"` a tileplot is generated. It displays each cells of the dataset and shows the original value from the observations, color coding indicates whether those values were higher (red) or lower (blue) than the imputed values, and the color intensity is based on the magnitude of the Shapley value. If `type` is `"both"`, the barplot and the tileplot are generated.

Examples

```
library(MASS)
set.seed(1)
n <- 100; p <- 10
mu <- rep(0,p)
Sigma <- matrix(0.9, p, p); diag(Sigma) = 1
X <- mvrnorm(n, mu, Sigma)
X[sample(1:(n*p), 100, FALSE)] <- rep(c(-5,5),50)
MOE_X <- MOE(X, mu, Sigma)
plot(MOE_X, subset = 20, n_digits = 0)
```

---

plot.shapley_interaction

Plot of Shapley interaction indices

Description

Plot of Shapley interaction indices

Usage

```
## S3 method for class 'shapley_interaction'
plot(
  x,
  abbrev = 4,
  title = "Shapley Interaction",
  legend = TRUE,
  text_size = 22,
  ...
)
```

Arguments

x	A $p \times p$ matrix containing the Shapley interaction indices (shapley_interaction) of a single observation.
abbrev	Integer. If abbrev.var > 0, variable names are abbreviated using abbreviate with minlenght = abrev.
title	Character. Title of the plot.
legend	Logical. If TRUE (default), a legend is plotted.
text_size	Integer. Size of the text in the plot
...	Optional arguments passed to methods.

Value

Returns a figure consisting of two panels. The upper panel shows the Shapley values, and the lower panel the Shapley interaction indices.

Examples

```
p <- 5
mu <- rep(0,p)
Sigma <- matrix(0.9, p, p); diag(Sigma) = 1
Sigma_inv <- solve(Sigma)
x <- c(0,1,2,2.3,2.5)
PHI <- shapley_interaction(x, mu, Sigma)
plot(PHI)
```

---

print.shapley *Print function for class* shapley.

Description

Print function for class shapley.

Usage

```
## S3 method for class 'shapley'
print(x, ...)
```

Arguments

x	List of class shapley.
...	Optional arguments passed to methods.

Value

Prints the list entries of x that are not NULL.

print.shapley_algorithm *Print function for class* shapley_algorithm.

Description

Print function for class shapley_algorithm.

Usage

```
## S3 method for class 'shapley_algorithm'
print(x, ...)
```

Arguments

x	List of class shapley_algorithm.
...	Optional arguments passed to methods.

Value

Prints the imputed data and the Shapley values.

print.shapley_interaction *Print function for class* shapley_interaction.

Description

Print function for class shapley_interaction.

Usage

```
## S3 method for class 'shapley_interaction'
print(x, ...)
```

Arguments

x	Matrix of class shapley_interaction.
...	Optional arguments passed to methods.

Value

Prints the Shapley interaction indices.

SCD *Detecting cellwise outliers using Shapley values.*

Description

The SCD function indicates outlying cells for a data vector with p entries or data matrix with $n \times p$ entries containing only numeric entries x for a given center mu and covariance matrix Sigma using the Shapley value (Mayrhofer and Filzmoser 2022).

Usage

```
SCD(
  x,
  mu,
  Sigma,
  Sigma_inv = NULL,
  step_size = 0.1,
  min_deviation = 0,
  max_step = NULL,
  max_iter = 1000,
  q = 0.99,
  method = "cellMCD",
  check = TRUE,
  cells = NULL
)
```

Arguments

x	Data vector with p entries or data matrix with $n \times p$ entries containing only numeric entries.
mu	Either NULL (default) or mean vector of x. If NULL, method is used for parameter estimation.
Sigma	Either NULL (default) or covariance matrix $p \times p$ of x. If NULL, method is used for parameter estimation.
Sigma_inv	Either NULL (default) or Sigma's inverse $p \times p$ matrix. If NULL, the inverse of Sigma is computed using solve(Sigma).
step_size	Numeric. Step size for the imputation of outlying cells, with step_size $\in [0, 1]$. Defaults to 0.1.
min_deviation	Numeric. Detection threshold, with min_deviation $\in [0, 1]$. Defaults to 0.2
max_step	Either NULL (default) or an integer. The maximum number of steps in each iteration. If NULL, max_step $= p$.
max_iter	Integer. The maximum number of iterations.
q	Numeric. The quantile of the Chi-squared distribution for detection and imputation of outliers. Defaults to 0.99.

method	Either "cellMCD" (default) or "MCD". Specifies the method used for parameter estimation if mu and/or Sigma are not provided.
check	Logical. If TRUE (default), inputs are checked before running the function and an error message is returned if one of the inputs is not as expected.
cells	Either NULL (default) or a vector/matrix of the same dimension as x, indicating the outlying cells. The matrix must contain only zeros and ones, or TRUE/FALSE.

Value

A list of class shapley_algorithm (new_shapley_algorithm) containing the following:

x	A p-dimensional vector (or a $n \times p$ matrix) containing the imputed data.
phi	A p-dimensional vector (or a $n \times p$ matrix) containing the Shapley values (outlyingness-scores) of x; see shapley.
x_original	A p-dimensional vector (or a $n \times p$ matrix) containing the original data.
x_history	The path of how the original data vector was modified.
phi_history	The Shapley values corresponding to x_history.
S_history	The indices of the outlying cells in each iteration.

References

Mayrhofer M, Filzmoser P (2022). "Multivariate outlier explanations using Shapley values and Mahalanobis distances." doi:10.48550/ARXIV.2210.10063.

Examples

```
p <- 5
mu <- rep(0,p)
Sigma <- matrix(0.9, p, p); diag(Sigma) = 1
Sigma_inv <- solve(Sigma)
x <- c(0,1,2,2.3,2.5)
SCD_x <- SCD(x = x, mu = mu, Sigma = Sigma)
plot(SCD_x)

library(MASS)
set.seed(1)
n <- 100; p <- 10
mu <- rep(0,p)
Sigma <- matrix(0.9, p, p); diag(Sigma) = 1
X <- mvrnorm(n, mu, Sigma)
X[sample(1:(n*p), 100, FALSE)] <- rep(c(-5,5),50)
SCD_X <- SCD(X, mu, Sigma)
plot(SCD_X, subset = 20)
```

shapley *Decomposition of squared Mahalanobis distance using Shapley values.*

Description

The shapley function computes a p-dimensional vector containing the decomposition of the squared Mahalanobis distance of x (with respect to mu and Sigma) into outlyingness contributions of the individual variables (Mayrhofer and Filzmoser 2022). The value of the j-th coordinate of this vector represents the average marginal contribution of the j-th variable to the squared Mahalanobis distance of the individual observation x.
If cells is provided, Shapley values of x are computed with respect to a local reference point, that is based on a cellwise prediction of each coordinate, using the information of the regular cells of x, see (Mayrhofer and Filzmoser 2022).
If x is a $n \times p$ matrix, a $n \times p$ matrix is returned, containing the decomposition for each row.

Usage

```
shapley(
  x,
  mu = NULL,
  Sigma = NULL,
  inverted = FALSE,
  method = "cellMCD",
  check = TRUE,
  cells = NULL
)
```

Arguments

x	Data vector with p entries or data matrix with $n \times p$ entries containing only numeric entries.
mu	Either NULL (default) or mean vector of x. If NULL, method is used for parameter estimation.
Sigma	Either NULL (default) or covariance matrix $p \times p$ of x. If NULL, method is used for parameter estimation.
inverted	Logical. If TRUE, Sigma is supposed to contain the inverse of the covariance matrix.
method	Either "cellMCD" (default) or "MCD". Specifies the method used for parameter estimation if mu and/or Sigma are not provided.
check	Logical. If TRUE (default), inputs are checked before running the function and an error message is returned if one of the inputs is not as expected.
cells	Either NULL (default) or a vector/matrix of the same dimension as x, indicating the outlying cells. The matrix must contain only zeros and ones, or TRUE/FALSE.

Value

phi	A p-dimensional vector (or a $n \times p$ matrix) containing the Shapley values (outlyingness-scores) of x.
mu_tilde	A p-dimensional vector (or a $n \times p$ matrix) containing the alternative reference points based on the regular cells of the original observations.
non_centrality	The non-centrality parameters for the Chi-Squared distribution, given by mahlanobis(mu_tilde, mu, Sigma)

References

Mayrhofer M, Filzmoser P (2022). "Multivariate outlier explanations using Shapley values and Mahalanobis distances." doi:10.48550/ARXIV.2210.10063.

Examples

```
## Without outlying cells as input in the 'cells' argument#'
# Single observation
p <- 5
mu <- rep(0,p)
Sigma <- matrix(0.9, p, p); diag(Sigma) = 1
Sigma_inv <- solve(Sigma)
x <- c(0,1,2,2.3,2.5)
shapley(x, mu, Sigma)
phi <- shapley(x, mu, Sigma_inv, inverted = TRUE)
plot(phi)

# Multiple observations
library(MASS)
set.seed(1)
n <- 100; p <- 10
mu <- rep(0,p)
Sigma <- matrix(0.9, p, p); diag(Sigma) = 1
X <- mvrnorm(n, mu, Sigma)
X_clean <- X
X[sample(1:(n*p), 100, FALSE)] <- rep(c(-5,5),50)
call_shapley <- shapley(X, mu, Sigma)
plot(call_shapley, subset = 20)


## Giving outlying cells as input in the 'cells' argument
# Single observation
p <- 5
mu <- rep(0,p)
Sigma <- matrix(0.9, p, p); diag(Sigma) = 1
Sigma_inv <- solve(Sigma)
x <- c(0,1,2,2.3,2.5)
call_shapley <- shapley(x, mu, Sigma_inv, inverted = TRUE,
method = "cellMCD", check = TRUE, cells = c(1,1,0,0,0))
plot(call_shapley)

# Multiple observations
```

```
library(MASS)
set.seed(1)
n <- 100; p <- 10
mu <- rep(0,p)
Sigma <- matrix(0.9, p, p); diag(Sigma) = 1
X <- mvrnorm(n, mu, Sigma)
X_clean <- X
X[sample(1:(n*p), 100, FALSE)] <- rep(c(-5,5),50)
call_shapley <- shapley(X, mu, Sigma, cells = (X_clean - X)!=0)
plot(call_shapley, subset = 20)
```

`shapley_interaction` *Decomposition of squared Mahalanobis distance using Shapley interaction indices.*

Description

The `shapley_interaction` function computes a $p \times p$ matrix containing pairwise outlyingness scores based on Shapley interaction indices. It decomposes the squared Mahalanobis distance of `x` (with respect to `mu` and `Sigma`) into outlyingness contributions of pairs of variables (Mayrhofer and Filzmoser 2022).

Usage

```
shapley_interaction(x, mu, Sigma, inverted = FALSE)
```

Arguments

`x`	Data vector with p entries containing only numeric entries.
`mu`	Either `NULL` (default) or mean vector of `x`. If NULL, `method` is used for parameter estimation.
`Sigma`	Either `NULL` (default) or covariance matrix $p \times p$ of `x`. If NULL, `method` is used for parameter estimation.
`inverted`	Logical. If `TRUE`, `Sigma` is supposed to contain the inverse of the covariance matrix.

Value

A $p \times p$ matrix containing the decomposition of the squared Mahalanobis distance of `x` into outlyingness scores for pairs of variables with respect to `mu` and `Sigma`.

References

Mayrhofer M, Filzmoser P (2022). "Multivariate outlier explanations using Shapley values and Mahalanobis distances." doi:10.48550/ARXIV.2210.10063.

Examples

```
p <- 5
mu <- rep(0,p)
Sigma <- matrix(0.9, p, p); diag(Sigma) = 1
Sigma_inv <- solve(Sigma)
x <- c(0,1,2,2.3,2.5)
shapley_interaction(x, mu, Sigma)
PHI <- shapley_interaction(x, mu, Sigma_inv, inverted = TRUE)
plot(PHI)
```

WeatherVienna *Weather data from Vienna*

Description

Monthly data from the weather station Hohe Warte since April 1872 - Vienna (Stadt Wien 2022).

Usage

```
WeatherVienna
```

Format

A data frame with 1,804 rows and 25 columns:

`year` Year

`month` Month

`t` Daily mean air temperature in °C, (t7 mean + t19 mean + tmax mean + tmin mean)/4; before 1971: t7 mean + t14 mean + 2 x t21 mean)

`t_max` Absolute maximum air temperature in °C

`t_min` Absolute air temperature minimum in °C

`avg_t_max` Mean daily maximum air temperature in °C

`avg_t_min` Mean daily minimum air temperature in °C

`num_frost` Number of frost days (days with a temperature maximum tmin < 0.0 °C)

`num_ice` Number of ice days (days with a temperature maximum tmax < 0.0 °C)

`num_summer` Number of summer days (days with a temperature maximum tmax >= 25.0 °C)

`num_heat` Number of hot days (days with a temperature maximum tmax >= 30.0 °C)

`p` Daily mean air pressure in hPa (mean of all measurements at 7 a.m., 2 p.m., 7 p.m. CET; before 1971 9 p.m. instead of 7 p.m.)

`p_max` Maximum air pressure in hPa (maximum of all measurements7 am, 2 pm, 7 pm CET; before 1971 9 pm instead of 7 pm)

`p_min` Minimum air pressure in hPa (minimum of all measurements7 am, 2 pm, 7 pm CET; before 1971 9 pm instead of 7 pm)

`sun_h` Monthly total sunshine duration in hours

`num_clear` Number of clear days (daily mean cloudiness < 20/100)

`num_cloud` Number of cloudy days (daily mean cloudiness > 80/100)

`rel_hum` Daily mean relative humidity in percent (2 x RH7 mean + RH14 mean + RH19 mean)/4; before 1971 9 p.m. instead of 7 p.m.)

`rel_hum_max` Relative humidity maximum in percent

`rel_hum_min` Relative humidity minimum in percent

`wind_v` Monthly average wind speed in km/h

`num_wind_v60` Number of days with wind peaks >= 60 km/h

`wind_v_max` Maximum wind speed in km/h

`precp_sum` Monthly total precipitation in mm

`num_precp_01` Number of days with precipitation >= 0.1 mm

Source

The data were downloaded from `https://www.data.gv.at/katalog/dataset/wetter-seit-1872-hohe-warte-wien` in September 2022.

References

Stadt Wien (2022). "Monthly data from the weather station Hohe Warte since April 1872 - Vienna." https://www.data.gv.at/katalog/dataset/wetter-seit-1872-hohe-warte-wien.

Examples

```
data("WeatherVienna")
summary(WeatherVienna)
```

Index