Package ‘litteR’

October 10, 2020

Title  Litter Analysis
Version  0.8.2
Date  2020-10-09
Description  Data sets on various litter types like beach litter, riverain litter, floating litter, and seafloor litter are rapidly growing. This package offers a simple user interface to analyse these litter data in a consistent and reproducible way. It also provides functions to facilitate several kinds of litter analysis, e.g., trend analysis, power analysis, and baseline analysis. Under the hood, these functions are also used by the user interface. See Schulz et al. (2019) <doi:10.1016/j.envpol.2019.02.030> for details. MS-Windows users are advised to run ‘litteR’ in 'RStudio'. See our vignette: Installation manual for 'RStudio' and 'litteR'.

Depends  R (>= 4.0.0)
Imports  readr (>= 1.3.1), stringr (>= 1.4.0), dplyr (>= 1.0.0), tidyselect (>= 1.1.0), tidyr (>= 1.1.0), fs (>= 1.4.1), ggplot2 (>= 3.3.1), purrr (>= 0.3.4), rlang (>= 2.2.1), rmarkdown (>= 2.2), tcltk
Suggests  knitr, kableExtra, testthat (>= 2.3.2)
License  GPL (>= 3)
Encoding  UTF-8
LazyData  true
VignetteBuilder  knitr
RoxygenNote  7.1.1
NeedsCompilation  no
Author  Dennis Walvoort [aut, cre, cph], Willem van Loon [aut, cph]. Rijkswaterstaat - The Netherlands [cph, fnd, dtc]
Maintainer  Dennis Walvoort <dennis.Walvoort@wur.nl>
Description

A tool for the analysis of various litter types, e.g., beach litter, riverain litter, floating litter, and seafloor litter.
adj_boxplot_stats

Details

The easiest way to get convenient with litteR is to create an empty project directory and fill it with example files by calling the function `create_litter_project`. The workhorse function in litteR is called `litter`. This function will start a simple user interface and lets you select an input file (*.csv) and a settings file (*.yaml). It will produce an HTML-report with litter analysis results according to the selected options in the settings file. See the package vignette for more details.

Author(s)

Maintainer: Dennis Walvoort <dennis.Walvoort@wur.nl> [copyright holder]

Authors:

- Willem van Loon <willem.van.loon@rws.nl> [copyright holder]

Other contributors:

- Rijkswaterstaat - The Netherlands [copyright holder, funder, data contributor]

References


----

adj_boxplot_stats  Adjusted Boxplot Statistics

Description

Adjusted boxplot statistics according to Hubert & Vandervieren (2008). The upper whisker extends from the hinge to the largest value no further than the upper fence. Similarly, the lower whisker extends from the hinge to the smallest value no further than the lower fence. See Hubert & Vandervieren (2008, p.5191, Eq.5).

Usage

```r
adj_boxplot_stats(x, ...)
```

## Default S3 method:
```r
adj_boxplot_stats(x, ...)
```
Arguments

- `x` numeric vector
- `...` further arguments passed to or from other methods.

Value

Numeric vector consisting of respectively the lower whisker/fence, the first quartile/hinge, the median, the third quartile/hinge, and the upper whisker/fence.

Methods (by class)

- default: Adjusted Boxplot Statistics

References


See Also

- `stat_adj_boxplot`

Examples

adj_boxplot_stats(rlnorm(100))

create_litter_project  Create Project Directory

Description

Fills an empty directory (path) with example files. If the path argument is missing or NULL, a Tcl/Tk dialogue will be started.

Usage

create_litter_project(path = NULL)

Arguments

- `path` (Existing) directory name
create_logger

**Simple Logger**

**Description**

Logger, in the spirit of loggers like log4j. Implemented logging levels are DEBUG, INFO, WARN, ERROR (in increasing order of specificity). Logging events can be filtered to show only events with a minimum specificity.

**Usage**

```r
create_logger(con = stdout(), level = c("DEBUG", "INFO", "WARN", "ERROR"))
```

**Arguments**

- `con`: connection to write logging data to
- `level`: log only events of this level and those that are more specific (see details)

**Value**

Anonymous logging functions

**Examples**

```r
logger <- create_logger(level = "INFO")
logger$info("starting specific computation")
logger$info("Today is {Sys.Date()}")
```

---

cv

**Coefficient of Variation**

**Description**

Coefficient of Variation

**Usage**

```r
cv(x, na.rm = FALSE)
```

**Arguments**

- `x`: a numeric vector
- `na.rm`: logical. Should missing values be removed?
Value

coefficient of variation (numeric vector of length 1).

References


---

**enumerate** | **Enumerate Objects**

**Description**

Generic function for enumerating objects

**Usage**

```
enumerate(x, ...)
```

```r
## S3 method for class 'numeric'
enumerate(x, ...)
```

**Arguments**

- `x` object to enumerate
- `...` further arguments passed to or from other methods.

**Methods (by class)**

- numeric: enumerate numeric vector.

**See Also**

`enumerate.character`
**enumerate.character**

**Enumerate Character Vector**

**Description**

Collapsing a character vector of length n, to a character vector of length 1.

**Usage**

```
## S3 method for class 'character'
enumerate(x, ...)
```

**Arguments**

- `x` character vector
- `...` further arguments passed to or from other methods.

**Value**

character vector of length 1, with elements separated by a comma except for the last element which is prepended by "and".

**Examples**

```
enumerate("apples")
enumerate(c("apples", "oranges"))
enumerate(c("apples", "oranges", "pears"))
```

**enumerate.sequenized**

**Convert Sequenized Output to Character String**

**Description**

Convert Sequenized Output to Character String

**Usage**

```
## S3 method for class 'sequenized'
enumerate(x, ...)
```

**Arguments**

- `x` object of class sequenized.
- `...` further arguments passed to or from other methods.
intercept

Value

string representation (character vector of length 1) of a sequenized object

See Also

sequenize.integer

has_write_access

Check Write Permission

Description

Simple wrapper for file.access with mode=2

Usage

has_write_access(path)

Arguments

path filename

Value

TRUE if write access, FALSE if not

intercept

Intercept

Description

Extract the intercept from object x.

Usage

intercept(x, ...)

Arguments

x object

... further arguments passed to or from other methods.

Value

estimate of the intercept (numeric vector of length 1).
iod

Index of Dispersion

Description

A normalized measure of the dispersion of a probability distribution.

Usage

iod(x, na.rm = FALSE)

Arguments

x a numeric vector
na.rm logical. Should missing values be removed?

Value

index of dispersion (numeric vector of length 1).

References


is_date_format

Check Date Format

Description

Checks if the data format x complies with format.

Usage

is_date_format(x, format = "%Y-%m-%d")

Arguments

x object of class character or Date
format required date format (see strftime)

Value

TRUE if x complies with format, and FALSE otherwise.

Examples

is_date_format("2019-05-14", "%Y-%m-%d")
is_natural_number  Test for Natural Numbers

Description

Test for natural numbers according to ISO 80000-2, that is the set 0, 1, 2, ...

Usage

is_natural_number(x)

Arguments

x numeric vector

Value

TRUE in case x is a natural number, FALSE otherwise.

Examples

stopifnot(!is_natural_number(3.1))
stopifnot(!is_natural_number(2.99))
stopifnot(is_natural_number(3))
stopifnot(all(is_natural_number(0:9)))
stopifnot(sum(is_natural_number(c(1, 2.5, 3))) == 2)

list_duplicates  List Duplicates

Description

Lists all duplicates as a list of tuples.

Usage

list_duplicates(x, ...)

## S3 method for class 'character'
list_duplicates(x, ...)

## S3 method for class 'tbl'
list_duplicates(x, ...)

## S3 method for class 'data.frame'
list_duplicates(x, ...)
Arguments

x  object of class character, tibble or data.frame
...

Value

list of row numbers with duplicates

Methods (by class)

- character: list duplicates for a character vector.
- tbl: lists duplicates for a tibble.
- data.frame: lists duplicates for a data.frame.

Examples

list_duplicates(c("a", "b", "c")) # list()
list_duplicates(c("a", "b", "a", "c")) # list(c(1, 3))

Description

Starts a graphical user interface for analysing litter data. A Tcl/Tk-dialogue will be started if one or more arguments are missing.

Usage

litter(filename = NULL)

Arguments

filename  name of file containing settings (see vignette for details)

Details

For details, see our vignette by typing: vignette("litter-manual")

Value

directory name (invisibly) where all results are stored.
mann_kendall

Description
Performs Mann-Kendall non-parametric trend test.

Usage
mann_kendall(x, type = c("both", "increasing", "decreasing"))

## S3 method for class 'mann_kendall'
  test_statistic(x, ...)

## S3 method for class 'mann_kendall'
  p_value(x, ...)

Arguments
x numeric vector representing a time-series.
type direction to test (both, increasing, or decreasing).
... further arguments passed to or from other methods.

Value
object of class Mann-Kendall.

Methods (by generic)
  • test_statistic: Extracts Mann Kendall tau
  • p_value: Extract p-value

See Also
test_statistic, p_value, cor.test

Examples

# create mann_kendall object
mk <- mann_kendall(c(9, 4, 7, 5, 3), type = "decreasing")

# get test statistic tau
test_statistic(mk)

# get p-value
p_value(mk)
**medcouple**

### Description

Robust statistic that quantifies the skewness of univariate distributions.

### Usage

```r
medcouple(x, ...)  
```

### Arguments

- `x`: numeric vector
- `...`: further arguments passed to or from other methods.

### Value

`medcouple` (numeric vector of length 1).

### Methods (by class)

- default: default method

### Note

This is a naive, but robust and simple implementation. For a more efficient implementation see package `robustbase` and the references section below.

### References

### `p_value` function

**Description**

Extract p-value.

**Usage**

```r
p_value(x, ...)  
```

**Arguments**

- `x`: object
- `...`: further arguments passed to or from other methods.

**Value**

p-value of a test (numeric vector of length 1).

---

### `read_litter` function

**Description**

Reads litter data from various formats. Currently only the OSPAR data snapshot format, and a wide format are supported. See the package vignette for more details.

**Usage**

```r
read_litter(filename, logger = create_logger(level = "INFO"), type_names)  
```

**Arguments**

- `filename`: name of litter file
- `logger`: optional logger object (see `create_logger`)
- `type_names`: character vector of allowed type_names

**Value**

tibble with litter data in long format
**read_litter_types**  
*Read Type Names*

**Description**
Read the file that links type names to group codes. See the package vignette for more details.

**Usage**
```
read_litter_types(filename, logger = create_logger(level = "INFO"))
```

**Arguments**
- `filename`: name of type file
- `logger`: optional logger object (see `create_logger`)

**Value**
tibble with look-up-table of type names and group codes

---

**read_settings**  
*Read Settings File*

**Description**
Reads settings file. See tutorial for its format.

**Usage**
```
read_settings(filename, logger = create_logger(level = "INFO"))
```

**Arguments**
- `filename`: name of litter file
- `logger`: optional logger object (see `create_logger`)

**Value**
validated settings file
Sample From an ECDF

Description
Type stable implementation of an Empirical Cumulative Distribution Function (ECDF) sampler.

Usage
recdf(x, n)

Arguments
- x: numeric vector
- n: number of draws

Value
vector of n elements of the same type as x

See Also
ecdf

Examples
recdf(1:5, 10)

Relative Median Absolute Deviation

Description
This is the Median Absolute Deviation divided by the median and is similar to the coefficient of variation.

Usage
rmad(x, na.rm = FALSE)

Arguments
- x: a numeric vector
- na.rm: logical. Should missing values be removed?
Value

Relative median absolute deviation (numeric vector of length 1).

References

https://en.wikipedia.org/wiki/Median_absolute_deviation

---

**Rolling Statistics**

**Description**

Applies function `fun` within a rolling (moving) window of size `w` to vector numeric vector `x`.

**Usage**

`roll(x, w = 3, fun = mean)`

**Arguments**

- `x` numeric vector (time-series)
- `w` width of moving window
- `fun` function to be applied

**Value**

vector of length `length(x)-w`

---

**Sequenize Objects**

**Description**

Generic function for sequenizing objects

**Usage**

`sequenize(x, ...)`

**Arguments**

- `x` object to sequenize
- `...` further arguments passed to or from other methods.

**See Also**

`sequenize.integer`
sequenize.integer  Sequenize Integer Sequence

Description
Compression of integer sequences to 'start-end' notation. For instance c(1:5, 8:9) becomes "1-5, 8-9".

Usage
```r
## S3 method for class 'integer'
sequenize(x, ...)
```

Arguments
- `x` vector of integers.
- `...` further arguments passed to or from other methods.

Value
object of class sequenized

Note
The elements of `x` should be unique and in ascending order.

Examples
```r
sequenize(c(1:4, 8:9))
```

slope  Slope

Description
Extract slope.

Usage
```r
slope(x, ...)
```

Arguments
- `x` object
- `...` further arguments passed to or from other methods.

Value
estimate of the slope (numeric vector of length 1).
**stat_adj_boxplot**

*Adjusted Boxplot Statistics for ggplot2*

**Description**

Computes adjusted boxplot statistics to be used by ggplot2. See Hubert & Vandervieren (2008, p.5191, Eq.5).

**Usage**

```r
stat_adj_boxplot()
stat_adj_boxplot_outlier()
```

**Functions**

- `stat_adj_boxplot_outlier`: add outliers to adjusted boxplot

**References**


**See Also**

`adj_boxplot_stats`, `stat_adj_boxplot_outlier`

**Examples**

```r
library(ggplot2)
d <- data.frame(x = gl(2, 50), y = rnorm(100))
ggplot(data = d, mapping = aes(x = x, y = y)) +
  stat_adj_boxplot()
```

---

**test_statistic**

*Test Statistic*

**Description**

Extract test_statistic.

**Usage**

```r
test_statistic(x, ...)
```
theil_sen

Arguments

x object
...

further arguments passed to or from other methods.

Value
test statistic of a test (numeric vector of length 1).

See Also
test_statistic.wilcoxon, test_statistic.mann_kendall

theil_sen Theil Sen Slope Estimator

Description
Theil Sen Slope Estimator

Usage

theil_sen(x, y, ...)

## S3 method for class 'theil_sen'
slope(x, ...)

## S3 method for class 'theil_sen'
intercept(x, ...)

Arguments

x time vector (numeric, or Date).
y numeric value.
...

further arguments passed to or from other methods.

Value
object of class Theil_Sen.

Methods (by generic)

• slope: Extract slope.
• intercept: Extract intercept.

References
trimean

Examples

# create theil_sen object
ts <- theil_sen(1:5, c(1, 2, 3, 5, 9))

# get slope
slope(ts)

# get intercept
intercept(ts)

trimean  Tukey's Trimean

Description

Robust centrality measure estimated as the weighted average of the three quartiles: \((Q_1 + 2Q_2 + Q_3)/4\), where \(Q_1\), \(Q_2\) and \(Q_3\) are the first, second and third quartiles respectively.

Usage

trimean(x, ...)

## Default S3 method:
trimean(x, ...)

Arguments

x  numeric vector
...

further arguments passed to or from other methods.

Value

trimean (numeric value of length 1).

Methods (by class)

• default: Tukey's trimean

References

https://en.wikipedia.org/wiki/Trimean

Examples

stopifnot(trimean(0:100) == mean(0:100))
stopifnot(trimean(0:100) == median(0:100))
validate  
Validation of LitteR File Formats

Description

Generic function for validation of file formats.

Usage

validate(x, ...)

## S3 method for class 'litter'
validate(x, type_names, logger = create_logger(level = "INFO"), ...)

## S3 method for class 'litter_types'
validate(x, logger = create_logger(level = "INFO"), ...)

## S3 method for class 'settings'
validate(x, logger = create_logger(level = "INFO"), ...)

Arguments

x          object to validate
...
...        further arguments passed to or from other methods.
type_names character vector of permissible types
logger     optional logger object (see create_logger)

Value

validated object of class wide
validated object of class litter_types
validated settings (list)

Methods (by class)

- litter: validate litter data.
- litter_types: validate litter_types file
- settings: validate settings file
**Description**

Constructor for a Wilcoxon test (simple wrapper for `wilcox.test`).

**Usage**

```r
wilcoxon(x, type = c("both", "greater", "less"), mu = 0)
```

```r
## S3 method for class 'wilcoxon'
test_statistic(x, ...)
```

```r
## S3 method for class 'wilcoxon'
p_value(x, ...)
```

**Arguments**

- `x` numeric vector representing a time-series.
- `type` direction to test (both, increasing, or decreasing).
- `mu` baseline value (null hypothesis)
- `...` further arguments passed to or from other methods.

**Value**

object of class `wilcoxon`.

**Methods (by generic)**

- `test_statistic`: Extract test statistic V
- `p_value`: Extract p-value

**See Also**

- `wilcox.test`, `p_value`, `test_statistic`

**Examples**

```r
# create wilcoxon object
w <- wilcoxon(c(9, 4, 7, 5, 3), type = "less")

# get test statistic V
test_statistic(w)

# get p-value
p_value(w)
```
Index

adj_boxplot_stats, 3, 19
character, 11
connection, 5
cor.test, 12
create_litter_project, 3, 4
create_logger, 5, 14, 15, 22
cv, 5
data.frame, 11
ecdf, 16
enumerate, 6
enumerate.character, 6, 7
enumerate.sequenized, 7
file.access, 8
has_write_access, 8
intercept, 8
intercept.theil_sen (theil_sen), 20
iod, 9
is_date_format, 9
is_natural_number, 10
list, 11
list_duplicates, 10
litteR (litteR-package), 2
litter, 3, 11
litteR-package, 2
mann_kendall, 12
medcouple, 13
p_value, 12, 14, 23
p_value.mann_kendall (mann_kendall), 12
p_value.wilcoxon (wilcoxon), 23
read_litter, 14
read_litter_types, 15
read_settings, 15
recdf, 16
rmad, 16
roll, 17
sequenize, 17
sequenize.integer, 8, 17, 18
slope, 18
slope.theil_sen (theil_sen), 20
stat_adj_boxplot, 4, 19
stat_adj_boxplot_outlier, 19
stat_adj_boxplot_outlier (stat_adj_boxplot), 19
strptime, 9
test_statistic, 12, 19, 23
test_statistic.mann_kendall (mann_kendall), 12
test_statistic.wilcoxon, 20
test_statistic.wilcoxon (wilcoxon), 23
theil_sen, 20
tibble, 11
trimean, 21
validate, 22
wilcox.test, 23
wilcoxon, 23