

# Package ‘zonator’

May 18, 2020

**Type** Package

**Title** Utilities for Zonation Spatial Conservation Prioritization  
Software

**Version** 0.6.0

**Date** 2020-05-16

**License** FreeBSD

**Encoding** UTF-8

**Maintainer** Joona Lehtomaki <joona.lehtomaki@gmail.com>

**Description** Create new analysis setups and deal with results of  
Zonation conservation prioritization software <<https://github.com/cbig/zonation-core>>.  
This package uses data available in the 'zdat' (7.7 MB) package  
for building the vignettes.

**Imports** ggplot2 (>= 2.0.0), methods, RColorBrewer, raster, reshape2,  
rgdal

**Depends** R (>= 2.15.2)

**Suggests** knitr, rasterVis, rmarkdown, testthat, zdat (>= 0.1.0)

**StagedInstall** yes

**Additional\_repositories** <https://jlehtoma.github.io/drat>

**URL** <https://cbig.github.io/zonator/>

**BugReports** <https://github.com/cbig/zonator/issues>

**VignetteBuilder** knitr

**RoxygenNote** 7.1.0

**NeedsCompilation** no

**Author** Joona Lehtomaki [aut, cre]

**Repository** CRAN

**Date/Publication** 2020-05-18 08:50:02 UTC

**R topics documented:**

check_names	3
check_path	4
check_zonation	4
clean_str	5
comp	6
copy_zvariant	7
correlation	8
cost	8
create_spp	9
create_zproject	11
cross_jaccard	12
curves	13
decimalplaces	14
ds_alpha	15
featurenames	16
features_info	17
file_path_relative_to	18
file_path_sans_ext	19
get_dat_param	20
get_tutorialdir	20
get_variant	21
groupnames	22
groups	23
has_results	24
jaccard	25
leaf_tags	26
load_zproject	27
map_indexes	27
names,Zproject-method	28
nfeatures	28
nvariants	29
opendir	30
outdir	30
parse_bat	31
performance	32
plot,ZCurvesDataFrame,missing-method	33
plot,ZGroupCurvesDataFrame,missing-method	34
plot_hist	35
ppa_lsm	36
print,Zvariant-method	36
rank_raster	37
rank_rasters	38
read_bat	39
read_curves	39
read_dat	40
read_features_info	41

read_groups . . . . .	41
read_grp_curves . . . . .	42
read_ppa_lsm . . . . .	42
read_result_rasters . . . . .	43
read_spp . . . . .	44
regroup_curves . . . . .	44
require_package . . . . .	45
results . . . . .	46
run_bat . . . . .	46
save_zvariant . . . . .	47
selection_coverage . . . . .	48
set_dat_param . . . . .	48
show,Zvariant-method . . . . .	49
sppdata . . . . .	50
sppweights . . . . .	51
variants . . . . .	52
write_dat . . . . .	52
ZCurvesDataFrame . . . . .	53
ZGroupCurvesDataFrame . . . . .	54
zlegend . . . . .	54
zparameters . . . . .	55
Zproject . . . . .	56
Zresults . . . . .	56
Zvariant . . . . .	57

## Index 59

---

check_names	<i>A function check feature/group names.</i>
-------------	--

---

### Description

Checks a vector of names only contains unique items and if they're not, unique names will be created. Also, the items must be suitable for columns names. Function is strict so that if the vector is not valid or it cannot be coerced to be one an error is induced.

### Usage

```
check_names(x)
```

### Arguments

x                      Character or numeric vector.

### Value

Valid vector of the original size.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

check_path	<i>A function to deal with potentially relative paths.</i>
------------	--

---

**Description**

Checks if a path can be resolved (i.e. whether it exists). An additional parameter `parent.path` can be provided, in which case `x` is appended to it and the concatenated path is checked for existence. If the path cannot be resolved, raise an error.

**Usage**

```
check_path(x, parent.path = NULL, require.file = FALSE)
```

**Arguments**

<code>x</code>	Character string path.
<code>parent.path</code>	Character string root path.
<code>require.file</code>	Logical indicating if a file is required for return or if an existing parent folder is enough

**Value**

A cleaned character string

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

check_zonation	<i>Check if Zonation is installed.</i>
----------------	--

---

**Description**

Check if Zonation is installed.

**Usage**

```
check_zonation(exe = "zig3")
```

**Arguments**

<code>exe</code>	Character string for overriding the default Zonation executable (default: zig3).
------------------	--

**Value**

A logical indicating whether requested Zonation executable is found.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**Examples**

```
## Not run:  
  check_zonation("zig4")  
  
## End(Not run)
```

---

clean_str	<i>Clean leading and trailing whitespaces from a given string. Additionally, all occurrences of multiple whitespaces are replaced with a single whitespace.</i>
-----------	---

---

**Description**

Clean leading and trailing whitespaces from a given string. Additionally, all occurrences of multiple whitespaces are replaced with a single whitespace.

**Usage**

```
clean_str(x)
```

**Arguments**

x                    Character string.

**Value**

An absolute path to a file of NULL if the path does not exist.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

`comp`*Compare matrices in various ways.*

---

**Description**

Function can be used to compare two Zonation output rasters with one of the following functions (part of zonator package):

- correlation
- subtraction
- frequency(NOT IMPLEMENTED)
- coverage

**Usage**

```
comp(x, y, fun = "correlation", ...)
```

**Arguments**

<code>x</code>	Numeric matrix.
<code>y</code>	Numeric matrix.
<code>fun</code>	Function used for the numeric comparison.
<code>...</code>	Further arguments passed on to selected comparison function.

**Value**

A DataFrame with each row containing columns title, count, and catid

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

correlation subtraction frequency coverage

---

copy\_zvariant                    *Copy existing variant as a new Zvariant object.*

---

## Description

Corresponding files on the file system are immediately created. In order to modify the variant, manipulate the returned Zvariant object and use [save\\_zvariant](#) method.

## Usage

```
copy_zvariant(x, name, dir)
```

```
## S4 method for signature 'Zvariant,character,character'  
copy_zvariant(x, name, dir)
```

## Arguments

x	Zvariant object.
name	Character string naming the copied variant.
dir	Character string directory where the new variant is created on file system.

## Details

If the variant being copied has results, they are not copied to the new variant.

## Value

Zvariant object

## Note

Relative paths in spp-file are translated into absolute paths as otherwise dealing with them might be tricky.

## Author(s)

Joona Lehtomaki <[joona.lehtomaki@gmail.com](mailto:joona.lehtomaki@gmail.com)>

## See Also

[Zvariant-class save\\_zvariant](#)

---

correlation	<i>Correlation between two matrices.</i>
-------------	--

---

### Description

Calculate correlation between two matrices using [cor](#). A group of specific threshold can be set, in which case the correlations are calculated incrementally for values above the thresholds.

### Usage

```
correlation(x, y, method = "kendall", thresholds = c(0))
```

### Arguments

x	Numeric matrix.
y	Numeric matrix.
method	Character string correlation method used (default: 'kendall').
thresholds	Numeric vector of thresholds used (default: c(0)).

### Value

A list with 2 items:

thresholds	Correlations between 2 matrices with values above a given threshold.
total	Overall correlation between the 2 matrices.

### Author(s)

Joona Lehtomaki <joona.lehtomaki@gmail.com>

### See Also

[cor](#)

---

cost	<i>Get cost data of a Z* object.</i>
------	--------------------------------------

---

### Description

Returns the "cost\_needed\_for\_top\_fraction" column from Zonation curves file. Note that the cost is the same in curves and grp\_curves files. pr\_lost is always included in the returned data, but no other columns are included.



**Usage**

```
cost(x)

## S4 method for signature 'Zresults'
cost(x)

## S4 method for signature 'Zvariant'
cost(x)
```

**Arguments**

x                    Z\* object.

**Details**

Method implementation in class `Zvariant` is just a thin wrapper for passing the arguments to variant's code `Zresults` object.

**Value**

data.frame object with two columns:

- `pr_lost` Proportion of landscape lost.
- `cost` Cost of a given fraction of the solution.

If no results are available, return NA.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[Zresults-class](#) [Zvariant-class](#)

---

create\_spp

*Generate spp\_file based on a directory of input rasters.*

---

**Description**

Generate spp\_file based on a directory of input rasters.

**Usage**

```
create_spp(  
  filename = "filelist.spp",  
  weight = 1,  
  alpha = 1,  
  bqp = 1,  
  bqp_p = 1,  
  cellrem = 0.25,  
  spp_file_dir,  
  recursive = FALSE,  
  spp_file_pattern = ".+\\.\\.(tif|img)$",  
  override_path = NULL  
)
```

**Arguments**

filename	character string defining the name of the spp file created.
weight	numeric template value for weights.
alpha	numeric template value for alpha values.
bqp	numeric template value for bqp values.
bqp_p	numeric template value for bqp_p values.
cellrem	numeric template value for cellrem values.
spp_file_dir	character path or a vector of paths to target dir.
recursive	Logical defining whether files in spp_file_dir should be listed recursively.
spp_file_pattern	pattern used to match raster files.
override_path	character path used to override the dirpath in input raster file paths. In case recursive = TRUE, then there can be an arbitrary number of subdirectories and override path is used only up until the spp_file_dir. This way the correct subdirectory structure is retained.

**Value**

invisible(TRUE), function is used for side effects.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

create_zproject	<i>Create a new Zonation project on file system.</i>
-----------------	--

---

### Description

Based on a set of input arguments, creates a new Zonation project on the file system following a particular folder and file layout.

### Usage

```
create_zproject(
  name,
  dir,
  variants,
  dat_template_file = NULL,
  spp_template_file = NULL,
  spp_template_dir = NULL,
  overwrite = FALSE,
  debug = FALSE,
  ...
)
```

### Arguments

name	Character string name for the project. A new directory named by name will be created in a location specified by dir.
dir	Character string path pointing to a location to be created.
variants	Character vector of names for new variants. Ignored if using an existing project.
dat_template_file	Character path to a dat file template. If no template is specified, uses the template distributed with zonator. Ignored if using an existing project.
spp_template_file	Character path to a spp file template. If this or spp_template_dir are not specified, uses the template distributed with zonator. Ignored if using an existing project.
spp_template_dir	Character path to directory containing biodiversity feature rasters. If this or spp_template_file are not specified, uses the template distributed with zonator. If both are defined, then spp_template_dir overrides. Ignored if using an existing project.
overwrite	logical should existing project be overwritten (default: FALSE).
debug	logical defining if debugging level for logging should be used (default: FALSE).
...	additional arguments passed to <a href="#">create_spp</a> .

**Value**

Invisible(NULL) .

**Note**

This function is used only for the intended side-effect of creating a new Zonation project. To load the project as an instance of [Zproject-class](#), see [load\\_zproject](#).

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[load\\_zproject](#) and [create\\_spp](#).

---

cross_jaccard	<i>Calculate Jaccard coefficients bewteen all the RasterLayers within a single RasterStack.</i>
---------------	---

---

**Description**

This method is a utility method that is intended to be used to compare top-fractions of the landscape. Thus, x.max and y.max for [jaccard](#) are fixed to 1.0.

**Usage**

```
cross_jaccard(stack, thresholds, ...)
```

**Arguments**

stack	RasterStack-object.
thresholds	Numeric vector values of thresholds.
...	additional arguments passed on to <a href="#">jaccard</a> .

**Value**

Dataframe with Jaccard coefficients between all the RasterLayers.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[jaccard](#)

---

curves *Get curves results data of a Z\* object.*

---

### Description

`pr_lost` is always included in the returned data, but other columns can be specified using `cols` argument.

### Usage

```
curves(x, cols = NULL, groups = FALSE, lost.lower = 0, lost.upper = 1)

## S4 method for signature 'ZCurvesDataFrame'
curves(x, cols = NULL)

## S4 method for signature 'ZGroupCurvesDataFrame'
curves(x, cols = NULL)

## S4 method for signature 'Zresults'
curves(x, cols = NULL, groups = FALSE, lost.lower = 0, lost.upper = 1)

## S4 method for signature 'Zvariant'
curves(x, cols = NULL, groups = FALSE, lost.lower = 0, lost.upper = 1)
```

### Arguments

<code>x</code>	<code>Z*</code> object.
<code>cols</code>	numeric or character vector of columns to be returned
<code>groups</code>	logical indicating whether group curves data should be returned.
<code>lost.lower</code>	numeric defining the lower limit of <code>pr_lost</code> to be included [0.0, 0.99] (default: 0.0)
<code>lost.upper</code>	numeric defining the upper limit of <code>pr_lost</code> to be included [0.01, 1.0] (default: 1.0)

### Details

Arguments `upper` and `lower` can be used to define a specific range of `pr_lost` to be returned.

Method implementation in class `Zvariant` is just a thin wrapper for passing the arguments to `variant`'s code `Zresults` object.

### Value

`ZCurvesDataFrame` or `ZGroupCurvesDataFrame` containing the (selected) curves file data. If column names are provided, but none are found, return `NA`.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[Zresults-class](#) [Zvariant-class](#) [read\\_curves](#) [read\\_grp\\_curves](#)

---

decimalplaces

*Find out the number of decimal places in a number.*

---

**Description**

Original implementation from <https://stackoverflow.com/questions/5173692/how-to-return-number-of-decimal-places-in-r>

**Usage**

```
decimalplaces(x, true_number = FALSE)
```

**Arguments**

`x` Float or double numeric number.  
`true_number` Logical setting whether the true number (see notes) of decimal places.

**Value**

Integer number of decimal places. Maximum

**Note**

R usually restricts the number of decimal to 9 in printing etc. Unless `true_number = TRUE`, return 9 and give a warning.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

`ds_alpha`*Calculate alpha value for distribution smoothing.*

---

**Description**

alpha-value of biodiversity feature-specific scale of landscape use. The value indicates the range of connectivity of biodiversity features. For example, it may refer to how a species uses the surrounding landscape. This value can be calculated based on, for example, the dispersal capability or the home range sizes of the species.

**Usage**

```
ds_alpha(landscape.use, ratio)
```

**Arguments**

<code>landscape.use</code>	Use of landscape (in relation to connectivity) in given map units. Note that if the units used here differ from the real map units of the biodiversity feature the ratio between two must be set using <code>ratio</code> argument.
<code>ratio</code>	Defines the ratio between units used in <code>landscape.use</code> and the actual map units in the biodiversity feature. E.g. if the map unit of a feature is m and use of landscape is defined as 1.5 km, then <code>ratio</code> should be set to 1000.

**Value**

numerical alpha.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

Zonation manual.

**Examples**

```
ds_alpha(1.5, 1000)
```

---

featurenames	<i>Feature names of Zonation variant.</i>
--------------	---

---

### Description

Get and set names for analysis features used a given Zonation variant.

### Usage

```
featurenames(x)

featurenames(x) <- value

## S4 method for signature 'ZCurvesDataFrame'
featurenames(x)

## S4 replacement method for signature 'ZCurvesDataFrame,character'
featurenames(x) <- value

## S4 method for signature 'Zresults'
featurenames(x)

## S4 method for signature 'Zvariant'
featurenames(x)

## S4 replacement method for signature 'Zvariant,character'
featurenames(x) <- value
```

### Arguments

x	Z* object.
value	character vector of feature names to be assigned. Can be named or not.

### Details

Argument x can be an instance of one the following Z\* classes:

- Zvariant
- Zresults

### Value

Character vector of spp feature names.

### Note

spp features have by default names that are derived from the feature raster file path.



**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[Zvariant-class Zresults-class groupnames groups](#)

**Examples**

```
## Not run:
setup.dir <- system.file("extdata/tutorial/basic", package="zonator")
tutorial.project <- create_zproject(setup.dir)
variant.caz <- get_variant(tutorial.project, "01")

# Feature names for a Zvariant object
featurenames(variant.caz)

# Feature names for a Zresults object
results.caz <- results(variant.caz)
featurenames(results.caz)

## End(Not run)
```

---

features\_info

*Get the features info component of Zresults.*

---

**Description**

Returns the data in \*.features\_info.txt results standard output of Zonation if present.

**Usage**

```
features_info(x)

## S4 method for signature 'Zresults'
features_info(x)

## S4 method for signature 'Zvariant'
features_info(x)
```

**Arguments**

x                    Z\* object.

**Details**

Argument `x` can be an instance of one the following `Z*` classes:

- `Zvariant`
- `Zresults`

**Value**

data.frame containing the features info data.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[Zvariant-class](#) [Zresults-class](#) [groupnames](#) [groups](#)

**Examples**

```
## Not run:
setup.dir <- system.file("extdata/tutorial/basic", package="zonator")
tutorial.project <- create_zproject(setup.dir)
variant.caz <- get_variant(tutorial.project, "01")

# Feature names for a Zvariant object
features_info(variant_caz)

## End(Not run)
```

---

`file_path_relative_to` *Transform an absolute path to relative path in relation to given location*

---

**Description**

Transform an absolute path to relative path in relation to given location

**Usage**

```
file_path_relative_to(path, org_relative_to, new_relative_to)
```

**Arguments**

`path` Character string path.

`org_relative_to` Character string path to which path is originally relative to.

`new_relative_to` Character string path to which path is supposed to be relative to.

**Value**

Character string relative file path.

**Note**

Both path and relative\_to must be in absolute form.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

file_path_sans_ext	<i>Re-implementation of <a href="#">file_path_sans_ext</a> in tools. This version can handle "." just before the file extension, unlike the original implementation.</i>
--------------------	--

---

**Description**

Re-implementation of [file\\_path\\_sans\\_ext](#) in tools. This version can handle "." just before the file extension, unlike the original implementation.

**Usage**

```
file_path_sans_ext(x, compression = FALSE)
```

**Arguments**

x	Character vector giving file paths.
compression	Logical: should compression extension '.gz', '.bz2' or '.xz' be removed first?

**Value**

File path without the file extension.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

get\_dat\_param                    *Get a specified run setting parameter value*

---

### Description

Essentially these are parameter values corresponding to Zonation dat-files. This function is used to get values of given parameter. In the current implementation, sections have no meaning.

### Usage

```
get_dat_param(x, parameter, warn_missing = TRUE)
```

```
## S4 method for signature 'Zvariant'  
get_dat_param(x, parameter, warn_missing = TRUE)
```

### Arguments

x	Zvariant object.
parameter	Character string name of the parameter.
warn_missing	Logical indicating a warning should be raised if the parameter is not used.

### Value

Character string value of the parameter. If requested parameter is a valid Zonation parameter but not used currently, returns NA.

### Author(s)

Joona Lehtomaki <joona.lehtomaki@gmail.com>

### See Also

[Zvariant-class](#) and [set\\_dat\\_param](#).

---

get\_tutorialdir                    *Get the directory of Zonation tutorial.*

---

### Description

Get the directory of Zonation tutorial.

### Usage

```
get_tutorialdir()
```

**Value**

path Character path to Zonation tutorial directory.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

`get_variant`                      *Get a specified variant in a Zonation project*

---

**Description**

Get a specified variant in a Zonation project

**Usage**

```
get_variant(x, index)

## S4 method for signature 'Zproject'
get_variant(x, index)
```

**Arguments**

<code>x</code>	Zproject object.
<code>index</code>	int or string index defining the variant required.

**Value**

Zvariant object

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[Zproject-class](#) and [Zvariant-class](#)

---

groupnames	<i>Get group names for a class Zvariant instance.</i>
------------	---

---

### Description

Group names can be assigned to a [Zvariant](#) or [Zresults](#) object. This is a replacement function for variant group names. If the particular variant doesn't use groups the gives a warning.

### Usage

```
groupnames(x)

groupnames(x) <- value

## S4 method for signature 'ZGroupCurvesDataFrame'
groupnames(x)

## S4 method for signature 'Zresults'
groupnames(x)

## S4 method for signature 'Zvariant'
groupnames(x)

## S4 replacement method for signature 'Zvariant,character'
groupnames(x) <- value
```

### Arguments

x	Zvariant object.
value	named character vector.

### Details

All current group codes must be found in the keys, i.e. there can't be missing values. Argument value can, however, contain keys that are not in the current group codes.

### Value

A character vector containing the group names. If there are no groups, return NA.

### Author(s)

Joona Lehtomaki <joona.lehtomaki@gmail.com>

### See Also

[Zvariant-class groupnames groups](#)  
[Zvariant-class groupnames Zresults groups](#)

---

groups	<i>Get group codes of a class Zvariant instance.</i>
--------	--

---

### Description

If the particular variant doesn't use groups or doesn't have them assigned, return NA. Note that here 'groups' means the first column in Zonation groups file ('output group').

### Usage

```
groups(x)

groups(x) <- value

## S4 method for signature 'Zvariant'
groups(x)

## S4 replacement method for signature 'Zvariant,numeric'
groups(x) <- value
```

### Arguments

x	Zvariant object.
value	numeric vector of group ids. Vector length must match to the number of features, no recycling is done.

### Value

A numeric vector containing the group ids. If there are no groups, return NA.

### Author(s)

Joona Lehtomaki <joona.lehtomaki@gmail.com>

### See Also

[Zvariant-class](#)

---

has_results	<i>Check which results a Z* object has.</i>
-------------	---

---

### Description

If the results are available (i.e. variants have been run) then the variant should have a list object containing the results.

### Usage

```
has_results(x)

## S4 method for signature 'Zresults'
has_results(x)

## S4 method for signature 'Zvariant'
has_results(x)
```

### Arguments

x [Zvariant-class](#) or [Zresults](#) object.

### Details

The value returned is a list of logical where key of each element corresponds to a specific type of results.

### Value

list of logical values

### Author(s)

Joona Lehtomaki <joona.lehtomaki@gmail.com>

### See Also

[Zvariant-class](#)



---

`jaccard`*Calculate the Jaccard coefficient.*

---

### Description

The Jaccard coefficient measures similarity between sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets. The Jaccard coefficient can be calculated for a subset of rasters provided by using the threshold argument.

### Usage

```
jaccard(  
  x,  
  y,  
  x.min = 0,  
  x.max = 1,  
  y.min = 0,  
  y.max = 1,  
  warn.uneven = FALSE,  
  limit.tolerance = 4,  
  disable.checks = FALSE  
)
```

### Arguments

<code>x</code>	raster object.
<code>y</code>	raster object.
<code>x.min</code>	Numeric minimum threshold value for x to be used (default 0.0).
<code>x.max</code>	Numeric maximum threshold value for x to be used (default 1.0).
<code>y.min</code>	Numeric minimum threshold value for y to be used (default 0.0).
<code>y.max</code>	Numeric maximum threshold value for y to be used (default 1.0).
<code>warn.uneven</code>	Logical indicating whether a warning is raised if the compared raster coverages are very (>20x) uneven.
<code>limit.tolerance</code>	integer values that defines to which precision x and y limits are rounded to. This helps e.g. with values that close to 0 but not quite 0 (default: 4, i.e. <code>round(x, 4)</code> ).
<code>disable.checks</code>	logical indicating if the input limit values are checked against the actual raster values in x and y.

### Details

Min and max values must be provided for both RasterLayer objects x and y. Method can be used with RasterLayers of any value range, but the defaults [0.0, 1.0] are geared towards comparing Zonation rank priority rasters. Limits provided are inclusive.

**Value**

A numeric value [0, 1]

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

leaf_tags	<i>Find all the leaf tags in a potentially nested list. The generic form of a list is tag = value; find all the tags in a list.</i>
-----------	---

---

**Description**

Find all the leaf tags in a potentially nested list. The generic form of a list is tag = value; find all the tags in a list.

**Usage**

```
leaf_tags(x, omit_sections = FALSE)
```

**Arguments**

x                   List to be searched.  
omit\_sections   Logical indicating if sections should be omitted from vector names.

**Value**

Character vector of tags.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**Examples**

```
l <- list("a" = 1, "b" = list("c" = 3, "d" = 4), "e" = 5)  
leaf_tags(l)
```

---

load_zproject	<i>Load a Zonation project.</i>
---------------	---------------------------------

---

**Description**

Loads an existing Zonation project as an object of [Zproject-class](#). Individual variants within the Zonation project are parsed into [Zvariant-class](#) objects and potential results into [Zresults-class](#) objects.

**Usage**

```
load_zproject(root, debug = FALSE)
```

**Arguments**

root	Character string path pointing to an existing directory (with potentially bat-files in it) or to a new directory to be created.
debug	logical defining if debugging level for logging should be used.

**Value**

A Zproject object.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[Zproject-class](#), [Zvariant-class](#) and [create\\_zproject](#)

---

map_indexes	<i>Map vector to actual column indexes.</i>
-------------	---

---

**Description**

Compare a vector of column names or indexes against another vector which is known to be true.

**Usage**

```
map_indexes(x, y)
```

**Arguments**

x	Character or numeric vector of possible matches.
y	Character or numeric vector of true values. x and y must be of the same length.

**Value**

A numeric vector of the same length of x and y containing matched column indexes.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

names,Zproject-method *Names of variants in Zproject*

---

**Description**

Get the names of all the variants within a given [Zproject](#).

**Usage**

```
## S4 method for signature 'Zproject'
names(x)
```

**Arguments**

x                    Zproject object.

---

nfeatures                    *Get the number of feature included in a Zonation variant*

---

**Description**

Get the number of feature included in a Zonation variant

**Usage**

```
nfeatures(x)

## S4 method for signature 'Zvariant'
nfeatures(x)
```

**Arguments**

x                    Zvariant object.

**Value**

int number of variants

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[Zvariant-class](#)

---

nvariants

*Get the number of variants included in a Zonation project*

---

**Description**

Get the number of variants included in a Zonation project

**Usage**

```
nvariants(x)
```

```
## S4 method for signature 'Zproject'  
nvariants(x)
```

**Arguments**

x                   Zproject object.

**Value**

int number of variants

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[Zproject-class](#) and [Zvariant-class](#)

---

opendir	<i>Open the directory of a Zproject using the system file browser.</i>
---------	--

---

**Description**

Currently support Windows Explorer (Windows) and Dolphin (Linux/KDE).

**Usage**

```
opendir(x)

## S4 method for signature 'Zproject'
opendir(x)
```

**Arguments**

x                    a [Zproject](#) object.

**Value**

invisible

**Author(s)**

Joona Lehtomaki <[joona.lehtomaki@gmail.com](mailto:joona.lehtomaki@gmail.com)>

**See Also**

[Zproject-class](#) and [Zvariant-class](#)

---

outdir	<i>Get path to output directory.</i>
--------	--------------------------------------

---

**Description**

Zonation variant has an output directory defined in project bat-file. This is of course the same path as in the results of the particular variant.

**Usage**

```
outdir(x)

## S4 method for signature 'Zresults'
outdir(x)

## S4 method for signature 'Zvariant'
outdir(x)
```

**Arguments**

x                    [Zvariant-class](#) or [Zresults](#) object.

**Value**

character string path to output directory location.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[Zvariant-class Zresults-class](#)

---

parse_bat	<i>Parse the content of a Zonation batch (bat) file and make OS specific adjustments if needed.</i>
-----------	---

---

**Description**

The main issues faced between different platforms are the name of the executable, ways of calling it, and path separators. Relative file paths need to be expanded into full absolute paths.

**Usage**

```
parse_bat(bat.file, exe = NULL)
```

**Arguments**

bat.file	Character string path to a Zonation batch (bat) file.
exe	Character string for overriding the Zonation executable specified in the bat-file.

**Value**

A character string command sequence suitable for execution.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

performance                      *Get performance levels.*

---

### Description

Method returns performance levels at specified levels of cell removal for particular features (either for individual features or groups from a Zresults object).

### Usage

```
performance(x, pr.lost, features = "all", groups = FALSE, melted = FALSE)

## S4 method for signature 'Zresults'
performance(x, pr.lost, features = NULL, groups = FALSE, melted = FALSE)
```

### Arguments

x	Zresults object.
pr.lost	numeric vector containing the fractions of landscape lost for which the feature/group performance values are wanted (default: 'all').
features	character vector of features names to be extracted. Must match with feature names in curves data
groups	logical indicating whether group curves data should be used (default: FALSE).
melted	logical indicating whether the data.frame returned should be in melted format (default: FALSE)

### Value

Data frame containing the curves file data for selected fractions of landscape lost. First column is pr\_lost. If feature names are provided and none are viable, return NA.

### Note

In the current implementation, only average performance levels for groups are returned.

### Author(s)

Joona Lehtomaki <joona.lehtomaki@gmail.com>

### See Also

[Zresults-class read\\_curves read\\_grp\\_curves](#)



---

`plot,ZCurvesDataFrame,missing-method`*Plot Zonation performance curves for individual features.*

---

### Description

Generic plotting function for plotting feature performance curves. The method does some data pre-processing specific to `ZCurvesDataFrame` object before passing the data and arguments for `plot_curves`.

### Usage

```
## S4 method for signature 'ZCurvesDataFrame,missing'  
plot(  
  x,  
  min = FALSE,  
  mean = FALSE,  
  w.mean = FALSE,  
  ext = FALSE,  
  subs = NULL,  
  ...  
)
```

### Arguments

<code>x</code>	<code>ZCurvesDataFrame</code> object.
<code>min</code>	logical plot the minimum feature performance of a group (default: FALSE).
<code>mean</code>	logical plot the minimum feature performance of a group (default: FALSE). If no other statistic is used, mean will always be plotted. If other statistic(s) are plotted and mean is to be disabled, this will have to be done by setting mean explicitly to FALSE.
<code>w.mean</code>	logical plot the weighted mean feature performance of a group (default: FALSE).
<code>ext</code>	logical plot extinction risk of a group (default: FALSE).
<code>subs</code>	character vector defining the names of features (subset of all features) to be plotted.
<code>...</code>	Additional arguments passed on to <code>plot_curves</code> .

### Note

If no other statistic is selected, mean will be set to TRUE and plotted.

### Author(s)

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[read\\_curves](#) and [plot\\_curves](#).

---

plot,ZGroupCurvesDataFrame,missing-method

*Plot Zonation performance curves for groups.*

---

**Description**

Generic plotting function for plotting group performance curves. The method does some data pre-processing specific to [ZGroupCurvesDataFrame](#) object before passing the data and arguments for [plot\\_curves](#).

**Usage**

```
## S4 method for signature 'ZGroupCurvesDataFrame,missing'
plot(
  x,
  min = FALSE,
  mean = FALSE,
  w.mean = FALSE,
  max = FALSE,
  ext = FALSE,
  subs = NULL,
  ...
)
```

**Arguments**

x	<a href="#">ZGroupCurvesDataFrame</a> object.
min	logical plot the minimum feature performance of a group (default: FALSE).
mean	logical plot the minimum feature performance of a group (default: FALSE). If no other statistic is used, mean will always be plotted. If other statistic(s) are plotted and mean is to be disabled, this will have to be done by setting mean explicitly to FALSE.
w.mean	logical plot the weighted mean feature performance of a group (default: FALSE).
max	logical plot the maximum feature performance of a group (default: FALSE).
ext	logical plot extinction risk of a group (default: FALSE).
subs	character vector defining the names of groups (subset of all groups) to be plotted.
...	Additional arguments passed on to <a href="#">plot_curves</a> .

**Note**

If no other statistic is selected, mean will be set to TRUE and plotted.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[read\\_curves](#) and [plot\\_curves](#).

---

plot\_hist

*Create a ggplot2 histogram of a RasterLayer.*

---

**Description**

Create a ggplot2 histogram of a RasterLayer.

**Usage**

```
plot_hist(  
  x,  
  mask.obj = NULL,  
  add.mean = FALSE,  
  add.median = FALSE,  
  save.dir = "",  
  binwidth = 0.05,  
  title = NULL  
)
```

**Arguments**

x	RasterLayer object containing the spatial data.
mask.obj	RasterLayer object optionally used for masking only specific parts of x.
add.mean	Boolean whether a vertical blue line is added to the plot indicating the mean value of x.
add.median	Boolean whether a vertical red line is added to the plot indicating the median value of x.
save.dir	Character path (folder) for saving the plot as an image.
binwidth	Double value of binwidth for <a href="#">geom_bar</a> .
title	Character string title of the plot.

**Value**

a ggplot object containing the plot.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[geom\\_bar](#).

---

ppa_lsm	<i>Get ppa.lsm results data of a Z* object.</i>
---------	---

---

**Description**

Simple getter-method for ppa.lsm information (if used) contained in Zonation results

**Usage**

```
ppa_lsm(x)

## S4 method for signature 'Zresults'
ppa_lsm(x)
```

**Arguments**

x                    Z\* object.

**Value**

Data frame containing PPA LSM data items 1 and 3 combined (See [Zresults-class](#) for more details). If no results are available, give a warning.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[Zresults-class read\\_curves read\\_grp\\_curves](#)

---

print,Zvariant-method	<i>Print Zvariant information.</i>
-----------------------	------------------------------------

---

**Description**

Generic printing function

**Usage**

```
## S4 method for signature 'Zvariant'
print(x)
```

**Arguments**

x                    ZVariant object.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

rank\_raster

*Get Zonation result rank raster.*

---

**Description**

Getter method for rank priority raster included in Zonation results. Rank raster is one of the main outputs of Zonation.

**Usage**

```
rank_raster(x)
```

```
## S4 method for signature 'Zresults'  
rank_raster(x)
```

```
## S4 method for signature 'Zvariant'  
rank_raster(x)
```

**Arguments**

x                    [Zresults](#) or [Zvariant](#) object.

**Details**

Since a given Zvariant object can only have 1 rank priority raster, this method only calls the rank\_raster method of a Zresults object associated with the Zvariant object.

**Value**

[RasterLayer](#) object. If no results are available, give a warning.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[Zresults-class](#)

---

rank_rasters	<i>Get Zonation result rank rasters of a project.</i>
--------------	---

---

### Description

Each [Zproject](#) object has a set of variants and their results associated with it. This method will get the selected available rank rasters (1 per variant) and create a [RasterStack](#) object.

### Usage

```
rank_rasters(x, variants = NULL)

## S4 method for signature 'Zproject'
rank_rasters(x, variants = NULL)
```

### Arguments

x	<a href="#">Zproject</a> object.
variants	a numeric (IDs) or character (name) vector defining which variants are included in the returned stack (default: NULL means all).

### Details

Argument variants can be used to the define which variants are included, the default is to return all. Method will give a warning if a variant doesn't have a rank raster associated with it. If none of the variants have a rank raster, then a NA is returned.

### Value

[RasterStack](#) object. If no rank rasters are available at all, return NA.

### Author(s)

Joona Lehtomaki <joona.lehtomaki@gmail.com>

### See Also

[rank\\_raster](#) [get\\_variant](#)

---

read_bat	<i>Read Zonation-specific (MS Windows) batch file.</i>
----------	--

---

**Description**

Batch files include calls to Zonation core and look like following: call zig3.exe -r [INPUT\_PATH].dat [INPUT\_PATH].spp [OUTPUT\_PATH].txt 0.0 0 1.0 0

**Usage**

```
read_bat(infile)
```

**Arguments**

infile	Character string input file path.
--------	-----------------------------------

**Value**

List of parsed bat-parameters.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

read_curves	<i>Read in performance curves produced by Zonation.</i>
-------------	---

---

**Description**

Header is automatically generated based on the number of features in the file. If you need to read in grouped curves files, use [read\\_grp\\_curves](#) instead.

**Usage**

```
read_curves(infile)
```

**Arguments**

infile	Character file path to .curves.txt file.
--------	--

**Value**

Curves object with all the information in the curves file. If the requested file does not exist, return NA.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[read\\_grp\\_curves](#)

---

read\_dat

*Read a dat file (Windows-style ini-file) for configuration information.*

---

**Description**

Read a dat file (Windows-style ini-file) for configuration information.

**Usage**

```
read_dat(infile)
```

**Arguments**

infile            Character string input file path.

**Value**

List of parsed ini-parameters.

**Note**

Adapted from <http://bit.ly/11e4Jh0>

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>



---

read\_features\_info      *Read a features info file.*

---

**Description**

Read a features info file.

**Usage**

```
read_features_info(infile)
```

**Arguments**

infile                  Character string input file path.

**Value**

Data frame of parsed features info data.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

read\_groups              *Read a groups file.*

---

**Description**

Read a groups file.

**Usage**

```
read_groups(infile)
```

**Arguments**

infile                  Character string input file path.

**Value**

Data frame of parsed groups parameters.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

read_grp_curves	<i>Read in performance curves for grouped features produced by Zonation.</i>
-----------------	--

---

**Description**

Header is automatically generated based on the number of groups in the file. If you need to read in individual curves files, use [read\\_curves](#) instead.

**Usage**

```
read_grp_curves(infile)
```

**Arguments**

infile            Character file path to .curves.txt file.

**Value**

A DataFrame with all the information in the curves file. If the requested file does not exist, return NA.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[read\\_curves](#)

---

read_ppa_lsm	<i>Read Zonation post-processing analysis (ppa) result file.</i>
--------------	--

---

**Description**

Read Zonation post-processing analysis (ppa) result file.

**Usage**

```
read_ppa_lsm(x)
```

**Arguments**

x                Character string input file path.

**Value**

List of 3 data frames: 1. Most important species in units x 2. Average proportion remaining over all spp in units 3. Data fractions in units

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

read\_result\_rasters    *Read Zonation-specific raster result files.*

---

**Description**

Input rasters are given as raster names (i.e. without the raster file extension). Additional root (folder) path and file extension can be provided to construct the full paths.

**Usage**

```
read_result_rasters(rasters, path = NULL, format = NULL)
```

**Arguments**

rasters	Character vector of raster names.
path	Character string indication an optional root path that is prepended to each rasters names.
format	Character string indicating the raster format used (i.e. the file extension).

**Value**

A [RasterStack](#) object of result rasters.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

read_spp	<i>Read Zonation variant specific spp-file.</i>
----------	---

---

**Description**

Read Zonation variant specific spp-file.

**Usage**

```
read_spp(infile)
```

**Arguments**

infile	Character string input file path.
--------	-----------------------------------

**Value**

data.frame of parsed spp data.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

regroup_curves	<i>Re-calculate group curves data.</i>
----------------	--

---

**Description**

When results grouping is changed group-specific curves data has to be re-calculated. Normally group curves file is produced by Zonation based on the groupings provided by the user. Same information can almost completely (except for ext-values) be calculated afterwards from the feature-specific curves files.

**Usage**

```
regroup_curves(x, weights, group.ids)
```

**Arguments**

x	Data frame of feature specific representation levels.
weights	numeric vector for feature specific weights
group.ids	numeric vector of new group codes. Number of groups must match with columns in x.

**Details**

This function calculates the following stats for `Zvariant` object based on a vector of new group IDs:

**min:** Minimum value of representation on each iteration among features within a group.

**mean:** Mean value of representation on each iteration among features within a group.

**max:** Maximum value of representation on each iteration among features within a group.

**w. mean:** Weighted (based on feature weight) mean value of representation on each iteration among features within a group.

**Value**

`ZGroupCurvesDataFrame` with new group statistics.

**Note**

Current implementation does not calculate values for `ext2` (extinction risk). Column `ext2` is retained in the returned data frame for compatibility, but column will be populated with NAs.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

<code>require_package</code>	<i>Requires a given package and if not present installs and loads it.</i>
------------------------------	---

---

**Description**

Requires a given package and if not present installs and loads it.

**Usage**

```
require_package(package, ...)
```

**Arguments**

<code>package</code>	Character name of a package.
<code>...</code>	Additional arguments passed on to <code>install.packages</code> .

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

results	<i>Getter method for results (Zresults) in a class Zvariant object.</i>
---------	---

---

### Description

Since not all changes to Zvariant are reflected to its Zresults (e.g. feature and group names) there may quite a lot runtime patching going on.

### Usage

```
results(x)

## S4 method for signature 'Zvariant'
results(x)
```

### Arguments

x                   Zvariant object.

### Details

Results are returned even if only part of them are available.

### Value

Zresults object. If variant doesn't have any results return NA.

### Author(s)

Joona Lehtomaki <joona.lehtomaki@gmail.com>

### See Also

[Zresults-class](#)

---

run_bat	<i>Try to run a given batch (bat) files.</i>
---------	--

---

### Description

Try to run a given batch (bat) files.

### Usage

```
run_bat(bat.file, exe = "zig3")
```

**Arguments**

bat.file	Character string path to a Zonation batch (bat) file.
exe	Character string for overriding the default Zonation executable (default: zig3).

**Value**

A logical indicating whether running the batch file was successful.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

save_zvariant	<i>Saves the current state of an instance of <a href="#">Zvariant-class</a> on the file system. Zvariant object tracks the location of relevant files, but the root path can be changed. If it is not changed, then the current files can be overwritten.</i>
---------------	---

---

**Description**

Saves the current state of an instance of [Zvariant-class](#) on the file system. Zvariant object tracks the location of relevant files, but the root path can be changed. If it is not changed, then the current files can be overwritten.

**Usage**

```
save_zvariant(x, dir = "", overwrite = FALSE, debug_msg = FALSE)
```

```
## S4 method for signature 'Zvariant'
```

```
save_zvariant(x, dir = "", overwrite = FALSE, debug_msg = FALSE)
```

**Arguments**

x	Zvariant object.
dir	Character string path to the root directory where the variant is created.
overwrite	Logical indicating whether files should be overwritten if they exist.
debug_msg	Logical setting whether extra debugging information should be printed.

**Value**

Invisible NULL. This method is used only for its side effects.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[Zvariant-class](#)

---

selection_coverage	<i>Intersection of two coverages.</i>
--------------------	---------------------------------------

---

**Description**

Calculate how much two coverages (as defined by values greater than a given threshold in two numeric matrices) overlap.

**Usage**

```
selection_coverage(x, y, thresholds)
```

**Arguments**

x	Numeric matrix.
y	Numeric matrix.
thresholds	Numeric vector of thresholds used.

**Value**

A list with 2 items:

thresholds	Correlations between 2 matrices with values above a given threshold.
total	Overall correlation between the 2 matrices.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

set_dat_param	<i>Set a specified run setting parameter value</i>
---------------	--

---

**Description**

Essentially these are parameter values corresponding to Zonation dat-files. This function is used to set values of given parameter. In the current implementation, sections have no meaning.

**Usage**

```
set_dat_param(x, parameter, value)

## S4 method for signature 'Zvariant'
set_dat_param(x, parameter, value)
```



### Arguments

x	Zvariant object.
parameter	Character string name of the parameter.
value	Character string or numeric value of the parameter.

### Value

x Zvariant object.

### Note

Method does not check for the sanity of the values provided.

### Author(s)

Joona Lehtomaki <joona.lehtomaki@gmail.com>

### See Also

[Zvariant-class](#) and [get\\_dat\\_param](#).

---

show,Zvariant-method *Print Zvariant information.*

---

### Description

Generic printing function

### Usage

```
## S4 method for signature 'Zvariant'  
show(object)
```

### Arguments

object            ZVariant object.

### Author(s)

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

`sppdata`*Simple getter method for spp data in a class Zvariantobject.*

---

**Description**

Method will also return group column with spp data if it exists.

Data can be assigned independent of whether groups are used or not. Since groups information is stored separately in groups slot, groups information must also be updated independently.

**Usage**

```
sppdata(x, group.names = FALSE)

sppdata(x) <- value

## S4 method for signature 'Zvariant'
sppdata(x, group.names = FALSE)

## S4 replacement method for signature 'Zvariant,data.frame'
sppdata(x) <- value
```

**Arguments**

<code>x</code>	Zvariant object.
<code>group.names</code>	boolean indicating whether group codes (FALSE) or names (TRUE) are used to indicate group. (default: FALSE)
<code>value</code>	data frame that must match the number and names of columns in sppdata (see <a href="#">sppdata</a> .

**Value**

Data frame (object's spp.data)

**Note**

Everytime spp data is set, groups information is deleted as there is now straightforward way of preserving and/or imputing group information.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[Zvariant-class](#)

---

`sppweights`*Get biodiversity feature weights of a Zonation variant*

---

**Description**

Get biodiversity feature weights of a Zonation variant

Set biodiversity feature weights of a Zonation variant

**Usage**

```
sppweights(x)
```

```
sppweights(x) <- value
```

```
## S4 method for signature 'Zvariant'  
sppweights(x)
```

```
## S4 replacement method for signature 'Zvariant,numeric'  
sppweights(x) <- value
```

**Arguments**

`x` Zvariant object.

`value` numeric vector with the length equal to the number of features in the variant

**Value**

A numeric vector of weights

**Note**

The weight vector must be exactly the correct length, no recycling is done. Vector elements must be coercible to numeric.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[Zvariant-class](#)

---

variants	<i>Get all variants in a Zonation project</i>
----------	---

---

**Description**

Get all variants in a Zonation project

**Usage**

```
variants(x)

## S4 method for signature 'Zproject'
variants(x)
```

**Arguments**

x                    Zproject object.

**Value**

A list of Zvariant objects

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**See Also**

[Zproject-class](#) and [Zvariant-class](#)

---

write_dat	<i>Write a Zonation run configuration (dat) file.</i>
-----------	---

---

**Description**

The function takes a nested list of values and writes it to a dat file (a Windows style .ini-file).

**Usage**

```
write_dat(x, filename, overwrite = FALSE)
```

**Arguments**

x                    List containing the data to be written.  
filename            String file path.  
overwrite           Logical indicating whether the file should be overwritten.

**Value**

Invisible null.

**Note**

Only 1 level of nestedness is accepted.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**Examples**

```
## Not run:
  dat <- list("Settings" = list("removal_rule" = 1, use_groups = 1))
  write_dat(dat, "settings.dat")

## End(Not run)
```

---

ZCurvesDataFrame

*The ZCurvesDataFrame class*

---

**Description**

ZCurvesDataFrame class inherits class data.frame.

**Details**

Class does not implement new methods, but it is used to override some behaviour such as plot. Usually ZCurvesDataFrame object belongs to a [Zresults](#) object.

**Slots**

is.feature: Logical indicating whether column is actually a feature.

**Note**

If user modifies or subsets ZCurvesDataFrame in a function it is up to the user to update the indexes in slot is.feature

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

ZGroupCurvesDataFrame *The ZGroupCurvesDataFrame class*

---

**Description**

ZGroupCurvesDataFrame class inherits class data.frame.

**Details**

Class does not implement new methods, but it is used to override some behaviour such as plot. Usually ZGroupCurvesDataFrame object belongs to a [Zresults](#) object.

**Slots**

is.group: Logical indicating whether column is actually a group.

**Note**

If user modifies or subsets ZGroupCurvesDataFrame in a function it is up to the user to update the indexes in slot is.group

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

zlegend *Get various Zonation legends*

---

**Description**

Zonation result rank rasters can be displayed in various color schemes.

**Usage**

```
zlegend(x)
```

**Arguments**

x String character name for the color scheme.

**Details**

Each color scheme is a list with following item:

values: Value breaks in the rank priority map

labels: Labels to be used in the map legend

colors: Colors used for the value classes

Following color schemes are available:

1. "spectral"

**Value**

A list color scheme.

**Note**

Color schemes are stored in env .options.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

**Examples**

```
zlegend("spectral")
```

---

zparameters

*Get all Zonation run configuration parameters.*

---

**Description**

This set of parameters is all that is accepted by Zonation.

**Usage**

```
zparameters(just_names = FALSE)
```

**Arguments**

just\_names      Logical indicating if only the parameter names should be returned.

**Value**

Character vector of parameter names or a list of (parameter = section) structure.

**Note**

Parameters are hard-coded to this package and know nothing of potential future developments with Zonation.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

Zproject

*The Zproject class*

---

**Description**

Zproject class represents a Zonation project project, i.e. all the input and output files and folders.

**Details**

A project contains one or more variants of particular Zonation analysis setup. A single variant is represented as an instance of [Zvariant-class](#).

**Slots**

**root:** Character string path pointing to the root (dir) of the project.

**variants:** List of objects of class [Zvariant-class](#).

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

Zresults

*The Zresults class*

---

**Description**

Zresults class represents a full set of Zonation results associated with a single variant (instance of class [Zvariant-class](#)).

**Details**

Slots of class Zresults can be queried using the \$-operator.



**Slots**

**root:** Character string path pointing to the root (dir) of the results.  
**modified:** Character timestamp for when results were last modified.  
**run.info:** Character file path for run info file.  
**curves:** Data frame of curve (performance) results.  
**grp.curves:** Data frame of group curve (performance) results.  
**rank:** RasterLayer object of rank priority.  
**wrscr:** RasterLayer object of weighted range-size corrected richness.  
**prop:** RasterLayer object of the proportional loss of distribution.  
**ppa.lsm:** Data frame containing PPA LSM data items 1 and 3.  
**features.info:** Data frame containing features info data.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

---

Zvariant

*The Zvariant class*

---

**Description**

Zvariant class represents a Zonation analysis variant with the associated parameters.

**Details**

Currently Zvariant must be instantiated based on an existing Zonation batch file. If the variant has been run, then the results are also associated with the instance of Zvariant-class.

**Slots**

**name:** Character string name of the variant.  
**bat.file:** Character string path to a Zonation-style batch file.  
**dat.data:** List holding the parsed data from Zonation dat-file  
**spp.data:** Data frame holding the parsed data from Zonation spp-file  
**output.dir:** Character string path to the output directory.  
**groups:** Data frame holding the parsed data from Zonation groups-file  
**call.params:** List of parsed call parameters from the batch file.  
**condition.layers:** Data frame holding the parsed data from condition file.  
**results:** List holding the results (data frames).  
**results\_dirty:** Logical indicating if the current object data (dat.data and spp.data) has been changed when results are present. If TRUE, data has changed and results may have been produced using different data.

**Author(s)**

Joona Lehtomaki <joona.lehtomaki@gmail.com>

# Index

- \*Topic **post-processing**
  - comp, 6
  - correlation, 8
  - cross\_jaccard, 12
  - jaccard, 25
  - selection\_coverage, 48
- \*Topic **ppa**
  - comp, 6
  - correlation, 8
  - cross\_jaccard, 12
  - jaccard, 25
  - selection\_coverage, 48
- \*Topic **results**
  - cost, 8
  - curves, 13
  - map\_indexes, 27
  - ppa\_lsm, 36
  - read\_curves, 39
  - read\_grp\_curves, 42
  - regroup\_curves, 44
- \*Topic **zonation**
  - map\_indexes, 27
  - read\_curves, 39
  - read\_grp\_curves, 42
  - regroup\_curves, 44
- check\_names, 3
- check\_path, 4
- check\_zonation, 4
- clean\_str, 5
- comp, 6
- copy\_zvariant, 7
- copy\_zvariant, Zvariant, character, character-method (copy\_zvariant), 7
- cor, 8
- correlation, 8
- cost, 8
- cost, Zresults-method (cost), 8
- cost, Zvariant-method (cost), 8
- create\_spp, 9, 11, 12
- create\_zproject, 11, 27
- cross\_jaccard, 12
- curves, 13
- curves, ZCurvesDataFrame-method (curves), 13
- curves, ZGroupCurvesDataFrame-method (curves), 13
- curves, Zresults-method (curves), 13
- curves, Zvariant-method (curves), 13
- decimalplaces, 14
- ds\_alpha, 15
- featurenames, 16
- featurenames, ZCurvesDataFrame-method (featurenames), 16
- featurenames, Zresults-method (featurenames), 16
- featurenames, Zvariant-method (featurenames), 16
- featurenames<- (featurenames), 16
- featurenames<- , ZCurvesDataFrame, character-method (featurenames), 16
- featurenames<- , Zvariant, character-method (featurenames), 16
- features\_info, 17
- features\_info, Zresults-method (features\_info), 17
- features\_info, Zvariant-method (features\_info), 17
- file\_path\_relative\_to, 18
- file\_path\_sans\_ext, 19, 19
- geom\_bar, 35, 36
- get\_dat\_param, 20, 49
- get\_dat\_param, Zvariant-method (get\_dat\_param), 20
- get\_tutorialdir, 20
- get\_variant, 21, 38

- get\_variant, Zproject-method  
(get\_variant), 21
- groupnames, 17, 18, 22, 22
- groupnames, ZGroupCurvesDataFrame-method  
(groupnames), 22
- groupnames, Zresults-method  
(groupnames), 22
- groupnames, Zvariant-method  
(groupnames), 22
- groupnames<- (groupnames), 22
- groupnames<-, Zvariant, character-method  
(groupnames), 22
- groups, 17, 18, 22, 23
- groups, Zvariant-method (groups), 23
- groups<- (groups), 23
- groups<-, Zvariant, numeric-method  
(groups), 23
  
- has\_results, 24
- has\_results, Zresults-method  
(has\_results), 24
- has\_results, Zvariant-method  
(has\_results), 24
  
- install.packages, 45
  
- jaccard, 12, 25
  
- leaf\_tags, 26
- load\_zproject, 12, 27
  
- map\_indexes, 27
  
- names, Zproject-method, 28
- nfeatures, 28
- nfeatures, Zvariant-method (nfeatures),  
28
- nvariants, 29
- nvariants, Zproject-method (nvariants),  
29
  
- opendir, 30
- opendir, Zproject-method (opendir), 30
- outdir, 30
- outdir, Zresults-method (outdir), 30
- outdir, Zvariant-method (outdir), 30
  
- parse\_bat, 31
- performance, 32
- performance, Zresults-method  
(performance), 32
- plot, ZCurvesDataFrame, missing-method,  
33
- plot, ZGroupCurvesDataFrame, missing-method,  
34
- plot\_curves, 33–35
- plot\_hist, 35
- ppa\_lsm, 36
- ppa\_lsm, Zresults-method (ppa\_lsm), 36
- print, Zvariant-method, 36
  
- rank\_raster, 37, 38
- rank\_raster, Zresults-method  
(rank\_raster), 37
- rank\_raster, Zvariant-method  
(rank\_raster), 37
- rank\_rasters, 38
- rank\_rasters, Zproject-method  
(rank\_rasters), 38
- RasterLayer, 37
- RasterStack, 38, 43
- read\_bat, 39
- read\_curves, 14, 32, 34–36, 39, 42
- read\_dat, 40
- read\_features\_info, 41
- read\_groups, 41
- read\_grp\_curves, 14, 32, 36, 39, 40, 42
- read\_ppa\_lsm, 42
- read\_result\_rasters, 43
- read\_spp, 44
- regroup\_curves, 44
- require\_package, 45
- results, 46
- results, Zvariant-method (results), 46
- run\_bat, 46
  
- save\_zvariant, 7, 47
- save\_zvariant, Zvariant-method  
(save\_zvariant), 47
- selection\_coverage, 48
- set\_dat\_param, 20, 48
- set\_dat\_param, Zvariant-method  
(set\_dat\_param), 48
- show, Zvariant-method, 49
- sppdata, 50, 50
- sppdata, Zvariant-method (sppdata), 50
- sppdata<- (sppdata), 50

sppdata<- ,Zvariant,data.frame-method  
    (sppdata), 50  
sppweights, 51  
sppweights,Zvariant-method  
    (sppweights), 51  
sppweights<- (sppweights), 51  
sppweights<- ,Zvariant,numeric-method  
    (sppweights), 51  
  
variants, 52  
variants,Zproject-method (variants), 52  
  
write\_dat, 52  
  
ZCurvesDataFrame, 13, 33, 53  
ZCurvesDataFrame-class  
    (ZCurvesDataFrame), 53  
ZGroupCurvesDataFrame, 13, 34, 54  
ZGroupCurvesDataFrame-class  
    (ZGroupCurvesDataFrame), 54  
zlegend, 54  
zparameters, 55  
Zproject, 28, 30, 38, 56  
Zproject-class (Zproject), 56  
Zresults, 9, 13, 22, 24, 31, 37, 53, 54, 56  
Zresults-class (Zresults), 56  
Zvariant, 9, 13, 22, 37, 45, 57  
Zvariant-class, 47  
Zvariant-class (Zvariant), 57