Package ‘treePlotArea’

June 13, 2023

**Title**  Correction Factors for Tree Plot Areas Intersected by Stand Boundaries

**Version**  1.4.1

**Description**  The German national forest inventory uses angle count sampling, a sampling method first published as ‘Bitterlich, W.: Die Winkelzählmessung. Allgemeine Forst- und Holzwirtschaftliche Zeitung, 58. Jahrg., Folge 11/12 vom Juni 1947’ and extended by Grosenbaugh (<https://academic.oup.com/jof/article-abstract/50/1/32/4684174>) as probability proportional to size sampling. When plots are located near stand boundaries, their sizes and hence their probabilities need to be corrected.

**License**  BSD_2_clause + file LICENSE

**URL**  https://gitlab.com/fvafrcu/treeplotarea.git

**Depends**  R (&gt;= 4.0.0)

**Imports**  fritools, graphics, sf, stats

**Suggests**  checkmate, pkgload, plotrix, rmarkdown, rprojroot, RUnit, testthat, tinytest

**VignetteBuilder**  utils

**Encoding**  UTF-8

**Language**  en-US

**LazyData**  true

**RoxygenNote**  7.2.3

**NeedsCompilation**  no

**Author**  Andreas Dominik Cullmann [aut, cre], Bernhard Bösch [ctb], Christoph Fischer [ctb], Gerald Kändler [ctb]

**Maintainer**  Andreas Dominik Cullmann &lt;fvafrcu@mailbox.org&gt;

**Repository**  CRAN

**Date/Publication**  2023-06-13 10:00:02 UTC
**R topics documented:**

- `treePlotArea-package` .......................... 2
- `boundaries` .................................. 3
- `bw2bwi2022de` ................................. 4
- `check_boundaries` .............................. 5
- `get_boundary_polygons` ...................... 6
- `get_correction_factors` ..................... 7
- `get_defaults` ................................ 9
- `plot_tree_plot_area` ......................... 10
- `select_valid_angle_count_trees` .......... 11
- `set_options` .................................. 12
- `trees` ........................................ 13
- `validate_data` ............................... 14

**Index**

```
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>treePlotArea-package</td>
<td>2</td>
</tr>
<tr>
<td>boundaries</td>
<td>3</td>
</tr>
<tr>
<td>bw2bwi2022de</td>
<td>4</td>
</tr>
<tr>
<td>check_boundaries</td>
<td>5</td>
</tr>
<tr>
<td>get_boundary_polygons</td>
<td>6</td>
</tr>
<tr>
<td>get_correction_factors</td>
<td>7</td>
</tr>
<tr>
<td>get_defaults</td>
<td>9</td>
</tr>
<tr>
<td>plot_tree_plot_area</td>
<td>10</td>
</tr>
<tr>
<td>select_valid_angle_count_trees</td>
<td>11</td>
</tr>
<tr>
<td>set_options</td>
<td>12</td>
</tr>
<tr>
<td>trees</td>
<td>13</td>
</tr>
<tr>
<td>validate_data</td>
<td>14</td>
</tr>
</tbody>
</table>

```

---

**treePlotArea-package**

*Correction Factors for Tree Plot Areas Intersected by Stand Boundaries*

**Description**

The German national forest inventory uses angle count sampling, a sampling method first published by Bitterlich (1947) and extended by Grosenbaugh (1952) as probability proportional to size sampling. When plots are located near stand boundaries, their sizes and hence their probabilities need to be corrected.

**Details**

You will find the details in

vignette("An_Introduction_to_treePlotArea", package = "treePlotArea").

**References**


Description

An extract from the federal database. Refer to *Aufnahmeanweisung für die vierte Bundeswaldinventur (2021 - 2022)*.

Usage

```r
data("boundaries", package = "treePlotArea")
```

Format

A data frame with 148 observations on the following 13 variables. Variables not needed with the package are marked with an asterisk.

- `tnr` The tract id.
- `enr` The corner id. A tract may have up to 4 corners on wooden floor.
- `vbl` * An indicator giving the country. 804 denotes Baden-Württemberg.
- `rnr` * The boundary id.
- `rk` An indicator giving the validity of the boundary. Values of 9 or higher indicate that this boundary is not valid (any more).
- `rart` An indicator giving the type of the boundary (stand or forest boundary, for example).
- `rterrain` * An indicator giving the type of terrain behind the border.
- `spa_gon` The azimuth in gon of the starting point of the boundary.
- `spa_m` The distance to the starting point of the boundary in centimeter
- `spk_gon` As above, for the boundary’s flexing point.
- `spk_m` As above, for the boundary’s flexing point.
- `spe_gon` As above, for the boundary’s stopping point.
- `spe_m` As above, for the boundary’s stopping point.

References


Examples

```r
boundaries <- get(data("boundaries", package = "treePlotArea"))
```
Description

The data tree coming with this package was processed by Gerald Kaendler for the country of Baden-Wuerttemberg, and is the reference for testing as he adjusted diameter measurements to breast height where they had been measured in diverging heights (due to deformations of trees at breast height). Which we really need to do. But he did some other things we need to revert if we want to follow the standards from the federal database. He

1. converted the diameter at breast height from millimeter to centimeter and renamed it,
2. converted horizontal distance from centimeter to meter and renamed it.

So we add two variables holding the diameter in millimeter and the horizontal distance in centimeter, named by the output of

getOption("treePlotArea")[["angle_counts"]][["dbh"]]
and
getOption("treePlotArea")[["angle_counts"]][["distance"]]
respectively.

Usage

bw2bwi2022de(x)

Arguments

x A tree data set, typically get(data("trees", package = "treePlotArea").

Value

A tree data set prepared to work with the package.

See Also

Other data functions: select_valid_angle_count_trees()

Examples

trees <- get(data("trees", package = "treePlotArea"))
summary(trees)
angle_counts <- bw2bwi2022de(trees)
summary(angle_counts)
check_boundaries

Description

There is a boundary (tract 6878, corner 1, boundary 1) in the federal database for the 2012 survey that runs exactly through the plot. If that boundary would be valid, at that corner the term "stand" is not defined.

Usage

check_boundaries(x, stop_on_error = TRUE, clean_data = FALSE)

Arguments

- **x**: A data.frame containing boundaries. It has to have columns named by the contents of either 
  - get_defaults()[["boundaries"]]
  - getOption("treePlotArea")[["boundaries"]).
  Could be get(data("boundaries", package = "treePlotArea")).

- **stop_on_error**: Throw an error if invalid boundaries are found?

- **clean_data**: Get rid of invalid boundaries?

Details

So we check for such boundaries. These are straight boundaries with identical azimuth values for start and end, and flexed boundaries where azimuth values for either start or end and the azimuth value for the nook are identical and the nook is farther away from the plot than the corresponding start or end.

Value

A (possibly cleansed) data.frame containing boundaries.

See Also

Other boundary functions: get_boundary_polygons()
get_boundary_polygons  
Convert Boundaries to Polygons

Description

Used by get_correction_factors to convert a boundary table to polygons. You may want to see the polygons, that is why we exported this function.

Usage

get_boundary_polygons(boundaries, stop_on_error = TRUE, clean_data = FALSE)

Arguments

boundaries  A data.frame containing boundaries. It has to have columns named by the contents of either get_defaults()[["boundaries"]]
getOption("treePlotArea")[["boundaries"]]. Could be get(data("boundaries", package = "treePlotArea")) or the output of get_boundary_polygons.

stop_on_error  Throw an error if invalid boundaries are encountered? (There was tract 6878, corner 1, boundary 1 in the federal database for the 2012 survey, runs through the plot. There is no stand defined that way!).

clean_data  Omit invalid boundaries in any further calculations?

Value

A list with all boundary polygons for each corner for each tract.

See Also

Other boundary functions: check_boundaries()

Examples

boundaries <- get(data("boundaries", package = "treePlotArea"))
boundary_polygons <- get_boundary_polygons(boundaries)
get_correction_factors

Correction Factors for Tree Plot Areas Intersected by Stand Boundaries

Description

Get correction factors for an angle count table (i.e. a data.frame) and a corresponding boundary table (i.e. a data.frame).

Usage

get_correction_factors(
  angle_counts,
  boundaries,
  verbose = TRUE,
  stop_on_error = FALSE,
  skip_check = FALSE,
  counting_factor = 4
)

Arguments

angle_counts A data.frame containing angle counts. It has to have columns named by the contents of either
get_defaults()[["angle_counts"]]
getOption("treePlotArea")[["angle_counts"]].
Could be bw2bwi2022de(get(data("trees", package = "treePlotArea"))).

boundaries A data.frame containing boundaries. It has to have columns named by the contents of either
get_defaults()[["boundaries"]]
getOption("treePlotArea")[["boundaries"]].
Could be get(data("boundaries", package = "treePlotArea")) or the output of get_boundary_polygons.

verbose Be verbose?

stop_on_error Passed to get_boundary_polygons.

skip_check We usually check if the angle counts are suitable, (for example whether a diameter at breast height, a horizontal distance and an azimuth measurement are given). Skip this check? This might be of interest if you want to check whether a deadwood plot (radius 5 m, centered around the corner) is intersected by a boundary.

counting_factor The basal area factor used in counting the trees. For tally trees in the German national forest inventory its value is 4 [m^2].
Details

The columns in the names have to be named according to the values of \texttt{getOption("treePlotArea")}. If they do not: you can either rename the columns or set the option accordingly, probably using \texttt{set_options}.

Value

A \texttt{data.frame} containing the correction factors and a status giving information on possibly errors.

See Also

\texttt{set_options}

Examples

data("trees", "boundaries", package = "treePlotArea")

# For CRAN's sake: draw a subset
tracts <- c(sample(boundaries[["tnr"]], 20), 10056)

trees <- subset(trees, tnr %in% tracts)
boundary_polygons <- get_boundary_polygons(boundaries)
correction_factors <- get_correction_factors(angle_counts, boundary_polygons)
summary(correction_factors$status)

# Select valid angle count trees only
valid_angle_counts <- select_valid_angle_count_trees(angle_counts)
correction_factors <- get_correction_factors(valid_angle_counts,
boundary_polygons)
summary(correction_factors$status)

# Select a single tree

# Dead wood plots:
dead_wood_plots <- unique(trees[TRUE, c("tnr", "enr")])
dead_wood_plots["bnr"] <- 0
dead_wood_plots["hori"] <- 0
dead_wood_plots["azi"] <- 0
dead_wood_plots["bhd"] <- 200
get_correction_factors(dead_wood_plots, boundary_polygons,  
  skip_check = TRUE)

# Set the deadwood plot's radius to 500 mm
dead_wood_plots["bhd"] <- 5000
# The counting factor has unit square meters per area.
# Area is hardcoded to 10000 [square meters], so to get a plot radius that's
# equal to the dbh, we need 2 * sqrt(counting_factor) / sqrt(10000) to be
# equal to 1.
get_correction_factors(dead_wood_plots, boundary_polygons,  
  skip_check = TRUE,  
  counting_factor = 2500)

---

get_defaults

Get Default Options for treePlotArea

Description

Used to see (not set) the default options set by treePlotArea. Use set_options to change these
default values.

Usage

get_defaults()

Value

A named list. It has the following entries giving the column names of the angle count or boundary
data that hold information on:

angle_counts tract_id  The tract id.
  corner_id  The corner id.
  tree_id    The tree id.
  distance   The distance from the center of the tract’s corner.
  azimuth    The azimuth from North.
  dbh        The diameter at breast height.

boundaries tract_id    The tract id.
  corner_id   The corner id.
  boundary_type Type of boundary.
  boundary_status Validity of the boundary.
  distance_start The starting point’s distance.
  distance_flexing The flexing point’s distance.
  distance_end  The ending point’s distance.
  azimuth_start The starting point’s azimuth.
  azimuth_flexing The flexing point’s azimuth.
  azimuth_end   The ending point’s azimuth.
plot_tree_plot_area

See Also
Other option functions: set_options()

Examples
get_defaults()

plot_tree_plot_area

Description
Visualize a tree, its plot area and its corner’s boundaries.

Usage
plot_tree_plot_area(
  angle_counts, boundaries, tnr, enr, bnr,
  frame_factor = 4, use_sub = TRUE
)

Arguments
angle_counts  A data.frame containing angle counts. It has to have columns named by the contents of either get_defaults()[['angle_counts']] or getOption("treePlotArea")['angle_counts']. Could be bw2bwi2022de(get(data("trees", package = "treePlotArea"))).

boundaries  A data.frame containing boundaries. It has to have columns named by the contents of either get_defaults()[['boundaries']] or getOption("treePlotArea")['boundaries']. Could be get(data("boundaries", package = "treePlotArea")) or the output of get_boundary_polygons.

tnr  Number of the tract.

enr  Number of the tract’s corner.

bnr  Number of the corner’s tree.

frame_factor  Plotting from as a factor of the tree plot area. Stick with the default.

use_sub  Use the subtitle (or leave it blank)? Stick with the default.
select_valid_angle_count_trees

Value

The corrections factor for the tree’s plot area

Examples

tnr <- 166
enr <- 2
bnr <- 7
angle_counts <- bw2bwi2022de(get(data("trees", package = "treePlotArea")))
plot_tree_plot_area(angle_counts = angle_counts,
        boundaries = get(data("boundaries",
        package = "treePlotArea"),
        tnr = tnr, enr = enr, bnr = bnr, frame_factor = 4)

select_valid_angle_count_trees

Select Valid Angle Counts Only

Description

The tree data coming with this package was processed by Gerald Kaendler for the country of Baden-Wuerttemberg, and is the reference for testing as he adjusted diameter measurements to breast height where they had been measured in diverging heights (due to deformations of trees at breast height). Which we really need to do. But he also added trees that are not part of the angle count sampling, which this function removes. We need that mainly to run tests against the reference values computed by grenzkreis because we would not be able to easily find the keys to merge the data. So this function is probably of no use to you. And we remove trees with a diameter at breast height greater than zero and a distance of 0, for these tree should not be there.

Usage

select_valid_angle_count_trees(x, sample_type = "stp", tree_status = "pk")

Arguments

x A tree data set, typically get(data(trees, package = "treePlotArea")).
sample_type An indicator giving the type of sample the tree was in. 0 marks the angle count sample with counting factor 4.
tree_status An indicator giving the status of a tree in the German national forest inventory. 0 marks ingrowth, 1 marks ongrowth.

Value

A tree data containing valid angle count trees only.

See Also

Other data functions: bw2bwi2022de()
Examples

```r
trees <- get(data("trees", package = "treePlotArea"))
subset(trees, entf == 0 & bhd2 > 0 & stp == 0)
angle_counts <- select_valid_angle_count_trees(trees)
subset(angle_counts, entf == 0 & bhd2 > 0 & stp == 0)
```

set_options

**Set Default Options for** treePlotArea

Description

Just convenience function for options. treePlotArea has a set of default options to define the columns of the data.frames that are passed to get_correction_factors. See get_defaults for a description of these options.

Usage

```r
set_options(...)  
```

Arguments

... See options. Leave empty to initialize the defaults if need be.

Value

Invisibly TRUE.

See Also

Other option functions: get_defaults()

Examples

```r
# Set the default
set_options()
getOption("treePlotArea")
# Overwrite some
option_list <- list(angle_counts = list(dbh = "diameter"),
                 boundaries = list(boundary_status = "boundart_stat"))
set_options(angle_counts = option_list["angle_counts"],
            boundaries = option_list["boundaries"])
getOption("treePlotArea")$angle_counts$dbh
# restore default
option_list <- get_defaults()
set_options(angle_counts = option_list["angle_counts"],
            boundaries = option_list["boundaries"])
getOption("treePlotArea")$angle_counts$dbh
```
Description

This is an extract form a data set prepared by Gerald Kaendler. He

1. added trees that are not part of the angle count sampling,
2. converted the diameter at breast height from millimeter to centimeter and renamed it,
3. converted horizontal distance from centimeter to meter and renamed it,
4. computed correction factors using grenzkreis.

Usage

```r
data("trees", package = "treePlotArea")
```

Format

A data frame with 1121 observations on the following 9 variables. Variables not needed with the package are marked with an asterisk.

- **tnr**: The tract id.
- **enr**: The corner id. A tract may have up to 4 corners on wooden floor.
- **bnr**: The tree id.
- **bhd2**: The diameter at breast height, given in centimeter.
- **kf2**: The correction factor given by grenzkreis.
- **entf**: The trees’ distance from the center of the tract’s corner, given in meter.
- **azi**: The azimuth from North, measured in gon (or gradian).
- **pk**: An indicator giving the type of a tree in the German national forest inventory. 0 marks ingrowth, 1 marks ongrowth.
- **stp**: An indicator giving the type of sample the tree was in. 0 marks the angle count sample with counting factor 4.

Examples

```r
trees <- get(data("trees", package = "treePlotArea"))
summary(trees)
angle_counts <- bw2bwi2022de(trees)
summary(angle_counts)
```
validate_data

Validate Data and Optionally Delete Missing Data

Description

The 2012 data of the federal database contains tract 18063, corner 2. There are boundaries recorded for that corner, nevertheless tree 14 has no azimuth measurement. This function therefore checks for the data sets not having missing data in the columns needed by `get_correction_factors` and optionally removes affected observations. It does not cross check whether missing data is really needed (azimuth is not when there is no boundary recorded for that tracts corner).

Usage

`validate_data(x, type = c(NA, "angle_counts", "boundaries"), clean = FALSE)`

Arguments

- **x**: A tree or angle count data set.
- **type**: The type of data, stick with the default to let us guess.
- **clean**: Omit missing data? If the input contains missing data in the columns needed by `get_correction_factors`, the affected observations may be deleted. Otherwise an error is thrown.

Value

A tree data set. The input, if that was valid data, the cleaned input otherwise. Throws an error if columns are missing.

Examples

```r
cartography <- get(data("boundaries", package = "treePlotArea")))
nrow(cartography)
nrow(validate_data(x = cartography))
boundaries[1, "enr"] <- NA
try(validate_data(boundaries))
nrow(validate_data(boundaries, clean = TRUE))
```
Index

* boundary functions
  check_boundaries, 5
  get_boundary_polygons, 6
* data functions
  bw2bwi2022de, 4
  select_valid_angle_count_trees, 11
* datasets
  boundaries, 3
  trees, 13
* option functions
  get_defaults, 9
  set_options, 12
* package
  treePlotArea-package, 2

boundaries, 3
bw2bwi2022de, 4, 11

check_boundaries, 5, 6
data.frame, 5–8, 10, 12

get_boundary_polygons, 5, 6, 6, 7, 10
get_correction_factors, 6, 7, 12, 14
get_defaults, 5–7, 9, 10, 12

Invisibly, 12

options, 12

plot_tree_plot_area, 10

select_valid_angle_count_trees, 4, 11
set_options, 8–10, 12

treePlotArea-package, 2
trees, 13
TRUE, 12

validate_data, 14