Package: kwb.CrowdWater (via r-universe)

June 14, 2026

Title Lake Indicators from CrowdWater Observations

Version 0.0.0.9000

Description This package uses observations data of lakes collected by the CrowdWater App to derive semi-quantitative indictors for nutrients (trophic sate), biotop value, use of the lakes and water availability characteristics.

License MIT + file LICENSE

URL https://github.com/KWB-R/kwb.CrowdWater

BugReports https://github.com/KWB-R/kwb.CrowdWater/issues

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Suggests covr, htmltools

Depends R (>= 2.10)

LazyData true

Imports htmlwidgets, leaflet, leaflet.extras, magrittr

Config/pak/sysreqs libabsl-dev cmake libgdal-dev gdal-bin libgeos-dev make libicu-dev libpng-dev libuv1-dev libssl-dev libproj-dev libsqlite3-dev libudunits2-dev

Repository https://kwb-r.r-universe.dev

Date/Publication 2025-09-29 14:21:20 UTC

RemoteUrl https://github.com/KWB-R/kwb.CrowdWater

RemoteRef HEAD

RemoteSha b962f46243718924e53d4e9aec968dc5ed504399

Contents

`add_ObservationIndicators` *Add all the indicators to the lake properties table*

Description

Add all the indicators to the lake properties table

Usage

```
add_ObservationIndicators(df_cw)
```

Arguments

`df_cw`	Dataframe of the CrowdWater lake properties, prepared by `prepare_data()`

Value

df_cw extended by 4 indicator columns

assign_points_MC *Assign Points of Multiple choice answer*

Description

Used within the indicator functions

Usage

```
assign_points_MC(df, property, categories, points, NA_answers = "No answer")
```

Arguments

df	Prepared CrowdWater data frame
property	Character Value. One of the column names of df
categories	Character vector that lists the catetories of the property for which points are assigned
points	Numeric vector of the same length and order of the categories vector with the assigned points
NA_answers	Character vector defining one or more answers that are treated as NA values

Value

a list of

- points: Numeric vector of the category points
- valid_answer: Logical vector if all the required properties were observed
- scoring_system: The points assigned to the categories

assign_points_SC *Assign Points of Single choice answer*

Description

Used within the indicator functions

Usage

```
assign_points_SC(df, property, categories, points, NA_answers = "No answer")
```

Arguments

df	Prepared CrowdWater data frame
property	Character Value. One of the column names of df
categories	Character vector that lists the catetories of the property for which points are assigned
points	Numeric vector of the same length and order of the categories vector with the assigned points
NA_answers	Character vector defining one or more answers that are treated as NA values

Value

a list of

- points: Numeric vector of the category points
- valid_answer: Logical vector if all the required properties were observed
- scoring_system: The points assigned to the categories

---

base_map *the Basemap used for all leaflet maps*

Description

the Basemap used for all leaflet maps

Usage

```
base_map(
  longitude = 13.3987,
  latitude = 52.5048,
  zoom = 11,
  berlin_districts = TRUE
)
```

Arguments

longitude	Numeric value defining the longitude if WGS84
latitude	Numeric value defining the latitude if WGS84
zoom	The initial and minimum zoom level
berlin_districts	Logical value if berlin districts should be added

Value

A leaflet map

bboxes *Regions Bounding Boxes*

Description

Regions Bounding Boxes

Usage

```
bboxes
```

Format

A named list of regions defined by bounding boxes. The bounding boxes are numeric vectors of coordinates which define the top, bottom, left, and right coordinates of the box.(in WGS84)

berlin_district_shp *Simple feature of the Berlin district boundaries*

Description

Simple feature of the Berlin district boundaries

Usage

```
berlin_district_shp
```

Format

Besides the geometry, the Name of the district and its ID is given

categories_sum *This function returns the sum of categories per property of the given dataframe*

Description

This function returns the sum of categories per property of the given dataframe

Usage

```
categories_sum(df, property, language = "english", includeNA = FALSE)
```

Arguments

df	Dataframe of the crowdwater lake properties prepared by prepare_data(), optionally filtered for one ROOT_ID
property	The property
language	A character string, defining the language of df. If unknown the function identify_language() can be used to find out.
includeNA	Logical if NA (No answer) data should be included

Value

A dataframe with the columns ROOT_ID, LATITUED, LONGITUDE and all the available categories per defined property. Besides the percentage of categories per site, the number of observations is given.

---

create_legend_labels *Adds additional information to legend values*

Description

Adds additional information to legend values

Usage

```
create_legend_labels(
  lv,
  low_label = NULL,
  high_label = NULL,
  increase_bounds = TRUE
)
```

Arguments

lv	legend values as returned by indicators_to_colors() in the legend_table
low_label	Additional lable for the lowest value
high_label	Additional lable for the highest value
increase_bounds	Add greater or equal / less or equal than sign to the highest /lowest legend value

Value

extended legend labels

get_LakeIndicators *Calculate all indicators for all recorded lakes*

Description

Calculate all indicators for all recorded lakes

Usage

```
get_LakeIndicators(df_cw)
```

Arguments

df_cw	Dataframe of the crowdwater lake properties prepared by prepare_data()

Value

A dataframe that contains columns of the root id, coordinates, the indicator values and number of observations as well as the date of the most recent observation

identify_language *Function identifies the language based on category names*

Description

Function identifies the language based on category names

Usage

```
identify_language(df_cw)
```

Arguments

df_cw	Dataframe of the crowdwater lake properties prepared by prepare_data()

Value

A character string of the language

indicator_biotope *Calculation of the CrowdWater Biotope Indicator*

Description

The indicator is calculated per observation

Usage

```
indicator_biotope(df_cw, return_scoring_system = FALSE)
```

Arguments

df_cw	Dataframe of the crowdwater lake properties prepared by prepare_data()
return_scoring_system	If TRUE the scoring system is returned instead of the indicator vector.

Value

Either a numeric vector of the indicator points, or a description of the scoring system. This is a list of properties and categories which contribute to the indicator.

indicator_nutrients *Calculation of the CrowdWater Nutrient Indicator*

Description

The indicator is calculated per observation

Usage

```
indicator_nutrients(df_cw, return_scoring_system = FALSE)
```

Arguments

df_cw	Dataframe of the prepared crowdwater data
return_scoring_system	If TRUE the scoring system is returned instead of the indicator vector.

Value

Either a numeric vector of the indicator points, or a description of the scoring system. This is a list of properties and categories which contribute to the indicator.

indicator_usage *Calculation of the CrowdWater Usage Indicator*

Description

The indicator is calculated per observation

Usage

```
indicator_usage(df_cw, return_scoring_system = FALSE)
```

Arguments

df_cw	Dataframe of the crowdwater lake properties prepared by prepare_data()
return_scoring_system	If TRUE the scoring system is returned instead of the indicator vector.

Value

Either a numeric vector of the indicator points, or a description of the scoring system. This is a list of properties and categories which contribute to the indicator.

indicator_waterStress *Calculation of the CrowdWater Biotope Indicator*

Description

The indicator is calculated per observation

Usage

```
indicator_waterStress(df_cw, return_scoring_system = FALSE)
```

Arguments

df_cw	Dataframe of the crowdwater lake properties prepared by prepare_data()
return_scoring_system	If TRUE the scoring system is returned instead of the indicator vector.

Details

While most citizens are not able to tell if the lake experiences water stress by the observation at one point of time, there might be citizens who know the water body for a long time. The more observations include this information the more certain it is. Some citizen may only know about water level changes but cannot answer if the lake dries out. This is why in contrast to the other indicators, the points are counted even if one question remains unknown or unanswered.

Value

Either a numeric vector of the indicator points, or a description of the scoring system. This is a list of properties and categories which contribute to the indicator.

`indicators_to_colors` *Transformation of numeric values into colors*

Description

Transformation of numeric values into colors

Usage

```
indicators_to_colors(
  v_indicator,
  v_worst,
  v_neutral,
  v_best,
  col_worst = "#E6550D",
  col_neutral = "#EFF3FB",
  col_best = "#08306B",
  col_NA = "#00000000",
  resolution_bad = 0.1,
  resolution_good = 0.1,
  resolution_legend = 1
)
```

Arguments

`v_indicator`	Numeric Vector of indicator data
`v_worst`	Worst possible value
`v_neutral`	Neutral value
`v_best`	Best possible value
`col_worst`	Color for worst value as hexadecimal string of the form "#rrggbb"
`col_neutral`	Color for worst value as hexadecimal string of the form "#rrggbb"
`col_best`	Color for worst value as hexadecimal string of the form "#rrggbb"
`col_NA`	Color for NA values (Transparent by default)
`resolution_bad`	the resolution between neutral and worst value
`resolution_good`	The resolution between netral and best value
`resolution_legend`	The resolution of the legend values

Value

Vector of colors corresponding to the indicator values

lake_indicator *From single observation to average site indicators*

Description

From single observation to average site indicators

Usage

```
lake_indicator(df_cw, indicator_type)
```

Arguments

df_cw	Dataframe of the crowdwater lake properties prepared by prepare_data()
indicator_type	A character defining the indicator. Either "nutrients" "biotope", "usage" or "waterstress"

Value

A data frame of 5 columns (ROOT_ID, LATITUDE, LONGITUDE and the indicator value and the number of observations)

latest_observations *Returnes only the latest observations per lake*

Description

Returnes only the latest observations per lake

Usage

```
latest_observations(df_cw)
```

Arguments

df_cw	Dataframe of the crowdwater lake properties prepared by prepare_data()

Value

A dataframe in the same shape as the lake properties dataframe with all historical observations removed

---

`leaflet_indicator` *Create an interactive leaflet map which contains all the information of the last observation*

Description

The leaflet map has two layers. 1) all the observered properties 2) the corresponding image of that observation

Usage

```
leaflet_indicator(cw_indicators, language = "german")
```

Arguments

`cw_indicators`	Dataframe of the crowdwater indicators per lake created by `get_LakeIndicators()`
`language`	A character string, defining the language if the map

Value

A leflet map

`leaflet_last_observation_days` *Create an interactive leaflet map which contains the days passed since the last observations*

Description

Create an interactive leaflet map which contains the days passed since the last observations

Usage

```
leaflet_last_observation_days(df_cw, language = "german")
```

Arguments

`df_cw`	Dataframe of the crowdwater lake properties prepared by `prepare_data()`
`language`	A character string, defining the language if the map

Value

A leaflet map

leaflet_last_observation_info

Create an interactive leaflet map which contains all the information of the last observation

Description

The leaflet map has two layers. 1) all the observered properties 2) the corresponding image of that observation

Usage

```
leaflet_last_observation_info(df_cw, language = "german")
```

Arguments

df_cw	Dataframe of the crowdwater lake properties prepared by prepare_data()
language	A character string, defining the language if the map

Value

A leflet map

merge_LakeIndicators *Merge two lake indicator tables*

Description

Merge two lake indicator tables

Usage

```
merge_LakeIndicators(x, y)
```

Arguments

x	indicator table created by lake_indicator()
y	indicator table created by lake_indicator()

Value

merged table

prepare_data *CrowdWater lake data preparation*

Description

The downloaded csv file is filtered for lake specific columns, for regions and for validated observations. The observed properties are named correctly in English.

Usage

```
prepare_data(path, file, region = NULL, validated_only = TRUE)
```

Arguments

path	The filepath
file	The filename including the ".csv" ending
region	Either a character defining a region which is available in or a numeric vector of coordinates in WGS84 shaping a rectangle by "top", "bottom", "left" and "right" coordinate.
validated_only	Logical if only checked observations are selected

Value

A list of two data frames. The first one contains data on the observered proberties, the second data frame contains data of the phyiscale category.

props_and_cats *List of CrowdWater Properties and Categories*

Description

List of CrowdWater Properties and Categories

Usage

```
props_and_cats
```

Format

A named list where list names correspond to the crowdwater properties (column names of the table). For each property a data frame of categories is given, which includes the translation into German or English, the order of categories if applicable (if not -> NA) and the type of answers

renameMultipleChoice *Replaces multiple separated category names by new names*

Description

Multiple choice answers per observation are combined a one cell of the table and seperated by a comma.

Usage

```
renameMultipleChoice(v, oldNames, newNames, sep = ", ")
```

Arguments

v	Character vector of categories
oldNames	Character vector listing the old names of the categories
newNames	Characteri vector listing the new names of the categories in the same order as the oldNames vector
sep	The seperator that is used in a cell between categories of the same observation

Value

Character vector of renamed categories

renameSingleChoice *Replaces multiple category names by new names*

Description

Replaces multiple category names by new names

Usage

```
renameSingleChoice(v, oldNames, newNames)
```

Arguments

v	Character vector of categories
oldNames	Character vector listing the old names of the categories
newNames	Characteri vector listing the new names of the categories in the same order as the oldNames vector

Value

Character vector of renamed categories

save_leaflet *Save a leaflet as html*

Description

By adding content to the html it is assured that icons are scaled according to the device. In this way the map can be used by mobile devices.

Usage

```
save_leaflet(x, file)
```

Arguments

x	Leaflet object to be saved
file	Filename including path

site_category_percentage *Returnes only the latest observations per lake*

Description

Returnes only the latest observations per lake

Usage

```
site_category_percentage(df_cw, includeNA = FALSE)
```

Arguments

df_cw	Dataframe of the crowdwater lake properties prepared by prepare_data()
includeNA	Logical if NA (No answer) data should be included

Value

A dataframe in the same shape as the lake properties dataframe with all historical observations removed

`tranlsate_categories` *From single observation to average site indicators*

Description

From single observation to average site indicators

Usage

```
tranlsate_categories(df_cw, from, to)
```

Arguments

`df_cw`	Dataframe of the crowdwater lake properties prepared by `prepare_data()`
`from`	Character string, defining the language of df_cw
`to`	Character string, defining the language to translate in

Value

df_cw translated

Index