

standardizeSnapshot

December 15, 2022

<code>create_logger</code>	<i>Create logger</i>
----------------------------	----------------------

Description

Initializes a logger with a given logfile.

Usage

```
create_logger(my_logfile)
```

Arguments

`my_logfile` path to the log file (character).

Value

Returns a logger and creates a logfile at the given path. If the path given does not exist, also creates this path.

<code>digikam</code>	<i>Digikam sample data</i>
----------------------	----------------------------

Description

A dataset mimicking typical Digikam data (randomized rows)

Usage

```
digikam
```

Format

A data frame with 100 rows and 22 variables:

X integer Row names (read by R when reading the csv file)

Station character Camera

Species character Species

DateTimeOriginal character Date and time

Date character Date

Time character Time

delta.time.secs integer Time elapsed since this species was last seen at this camera (seconds).
Not relevant because rows were permuted.

delta.time.mins double Time elapsed since this species was last seen at this camera (minutes).
Not relevant because rows were permuted.

delta.time.hours double Time elapsed since this species was last seen at this camera (hours).
Not relevant because rows were permuted.

delta.time.days double Time elapsed since this species was last seen at this camera (days). Not
relevant because rows were permuted.

Directory character Local directory where the original photo is

FileName character Name of the original photo on the local storage

EXIF.Model character Exif info

EXIF.Make character Exif info

metadata_Species character Species (other column)

metadata_Number character Species count

metadata_Behaviour character Tagged behaviors

metadata_Sex character Tagged sex

n_images integer Number of pictures associated to this event?

metadata_young_present character Tagged youngs (Yes/No)

metadata_Numberofindividuals logical Tagged number of individuals on the picture

HierarchicalSubject character Summary column for all metadata_...

<code>get_final_filename</code>	<i>Get final filemane</i>
---------------------------------	---------------------------

Description

Return the filename from the file columns.

Usage

```
get_final_filename(df)
```

Arguments

df The dataframe to be copied. Must have columns locationID, season, roll.

Value

The filename for this file in the format locationID_Sseason_Rroll.csv It there are several locationID, seasons or rolls, they are separated by a dash in the filename: locationID1-locationID2...

Examples

```
zooniverse_std <- standardize_snapshot_df(df = zooniverse, standard = standard)
get_final_filename(zooniverse_std)
```

get_logfile_name	<i>Name logfile</i>
------------------	---------------------

Description

Create a name for the logfile from an input file/folder.

Usage

```
get_logfile_name()
```

Value

A string with format "log__YYYY-MM-DD_HH:MM:SS.log" where YYYY-MM-DD_HH:MM:SS is the current date/time.

guess_classifier	<i>Guess classifier</i>
------------------	-------------------------

Description

Guesses the classifier used to annotate the data based on the column names given in colnames_df.

Usage

```
guess_classifier(colnames_df)
```

Arguments

colnames_df A character vector of column names.

Value

The classifier: either zooniverse, digikam or traptagger.

Examples

```
guess_classifier(colnames(zooniverse))
guess_classifier(colnames(traptagger))
guess_classifier(colnames(digikam))
```

read_snapshot_files	<i>Read Snapshot files</i>
---------------------	----------------------------

Description

Reads files from a vector of folders (and optionnally ignores some file/folders) into a list of dataframes.

Usage

```
read_snapshot_files(input, except, basepath)
```

Arguments

input	a character vector of valid paths: can be files or folders, or a mix of both
except	files to ignore (optional)
basepath	the part of the path that should be ignored when copying final files (i.e. absolute path inside one's computer that should not be copied in final file.)

Value

A named list of dataframe. Each element of the list is a dataframe containing the contents of a file read from the files list given in input. The names of the list are the file names from the root of input: If the input is a file, it is a filename. If input is a folder, it is the relative path from input to the file inside input.

Examples

```
## Not run:
read_snapshot_files("/home/lnicvert/Documents/PhD/Snapshot/data/1_raw_data/DHP",
                    basepath = "/home/lnicvert/Documents/PhD/Snapshot/data/1_raw_data",
                    except = "DHP/DHP+OVE_same_file/*")

## End(Not run)
```

rename_standard	<i>Rename columns to match standard names</i>
-----------------	---

Description

This function renames the columns in a dataframe to match the standard names, given in the dataframe standard.

Usage

```
rename_standard(
  df,
  classifier = c("zooniverse", "traptagger", "digikam"),
  standard_colnames
)
```

Arguments

df	The dataframe with the columns to rename
classifier	The classifier used to create the dataframe df. Can be 'zooniverse', 'traptagger', 'digikam'.
standard_colnames	A dataframe with 2 columns (at least) named like the classifier and 'new'. The column named like the classifier contains column names that are expected in the initial file. These names will be matched in the column names of df using partial matching (case insensitive and removing blanks). The column 'new' contains the column names for the final file. Columns in the classifier column will be renamed as the corresponding value in 'new'. If no pre-existing column corresponds to 'new' (indicated with a NA) then the column will be created and filled with NAs.

Value

Returns a dataframe for which the columns have been renamed

standardize_snapshot_df	<i>Standardize dataframe</i>
-------------------------	------------------------------

Description

Standardizes a dataframe to the Snapshot standard.

Usage

```
standardize_snapshot_df(df, standard_df, locationID_digikam, classifier)
```

Arguments

<code>df</code>	The dataframe to standardize. It is expected to match the data format for either Zooniverse, TrapTagger or Digikam processed data (i.e. have column names defined in <code>standfard_df</code>).
<code>standard_df</code>	The standard dataframe to match column names to the new standard. A dataframe with ≥ 2 columns, one of which must be named 'zooniverse', 'digikam' or 'traptagger' and another one must be named 'new'.
<code>locationID_digikam</code>	Optional locationID to be used for Digikam data (will display a warning if not provided for Digikam data.)
<code>classifier</code>	Optional character for the classifier.

Value

The standardized dataframe: it has the same columns as specified in `standard_df$new`, dates and times are standardized to "YYYY-MM-DD" and "HH:MM:SS", capture info (locationID, cameraID, roll, capture and season if classifier is zooniverse) is filled. Columns are also in the same order as provided in `standard_df$new` and the rows are ordered by camera, date and time.

Examples

```
standardize_snapshot_df(zooniverse, standard)
standardize_snapshot_df(traptagger, standard)
standardize_snapshot_df(digikam, standard, locationID_digikam = "MOK")
```

```
standardize_snapshot_list
```

Standardize a list of dataframes

Description

Standardizes a list of dataframes to the Snapshot standard.

Usage

```
standardize_snapshot_list(df_list, standard_df, classifier)
```

Arguments

<code>df_list</code>	a list of dataframes.
<code>standard_df</code>	The standard dataframe to match column names to the new standard. A dataframe with ≥ 2 columns, one of which must be named 'zooniverse', 'digikam' or 'traptagger' and another one must be named 'new'.
<code>classifier</code>	Optional character or vector of characters for the classifier.

Value

The list of standardized dataframes: each dataframe has the same columns as specified in `standard_df$new`, dates and times are standardized to "YYYY-MM-DD" and "HH:MM:SS", capture info (locationID, cameraID, roll, capture and season if classifier is `zooniverse`) is filled. Columns are also in the same order as provided in `standard_df$new` and the rows are ordered by camera, date and time.

Examples

```
df_list <- list(zooniverse, digikam, traptagger)
names(df_list)[2] <- "MOK_record_table_0min_deltaT_2021-05-07.csv"
standardize_snapshot_list(df_list, standard)
```

<code>standardizeSnapshot</code>	<i>Standardize and clean Snapshot data</i>
----------------------------------	--

Description

`standardizeSnapshot` is a R package to standardize camera trap records files from the Snapshot Safari project. Snapshot Safari data comes in different standards, following the method that was used to classify pictures (`Zooniverse`, `TrapTagger` or `Digikam`). This package allows to standardize all data sources to a unique file format and then cleans the files to homogenize records.

References

Snapshot Safari: a large-scale collaborative to monitor Africa's remarkable biodiversity (Pardo et al, 2021) (<https://www.sajs.co.za/article/view/8134>)

<code>standardize_species</code>	<i>Standardize species</i>
----------------------------------	----------------------------

Description

Eliminate species duplicate names (things like 'birdofprey' and 'birdsofprey')

Usage

```
standardize_species(species)
```

Arguments

`species` vector of species names

Value

the vector of species names with names standardized

standard	<i>Standard column names</i>
----------	------------------------------

Description

A dataframe listing the standard column names for the different Snapshot data formats.

Usage

```
standard
```

Format

A data frame with 39 rows and 4 variables:

```
zooniverse character Expected columns names for Zooniverse data
traptagger character Expected columns names for TrapTagger data
digikam character Expected columns names for Digikam data
new character Columns names for the output standardized data
```

traptagger	<i>TrapTagger sample data</i>
------------	-------------------------------

Description

A dataset mimicking typical TrapTagger data (randomized rows)

Usage

```
traptagger
```

Format

A data frame with 100 rows and 9 variables:

```
Capture_ID character ID for the capture composed of locationcapture#roll#Cam.Site
Cam.Site character Camera
id integer Picture ID
latitude double Camera latitude (mock data)
longitude double Camera longitude (mock data)
timestamp character Date time
capture_labels character Species seen on the picture
capture_sighting_count integer Species count
capture_url character Picture url (mock data)
```

write_log_message	<i>Write log message</i>
-------------------	--------------------------

Description

Writes a log message. If a logger is provided, writes to that logger; if it is NA, displays a message.

Usage

```
write_log_message(message, logger = NA, level = "info")
```

Arguments

message	The message to display/write to the logger
logger	Logger to write to (log4r object of class "logger") (defaults to NA)
level	Logging level: either 'info', 'warn', 'debug' or 'error'.

Value

Either a message or writes a log (with the logger parameters)

write_standardized_df	<i>Write the standardized file</i>
-----------------------	------------------------------------

Description

Writes a file to a given location. If to does not exist, it is created, and if filename is not provided, a default standardized name is chosen.

Usage

```
write_standardized_df(  
  df,  
  to,  
  filename,  
  write = TRUE,  
  return_path = ifelse(write, FALSE, TRUE),  
  verbose = TRUE  
)
```

Arguments

df	The standardized file
to	The target folder to copy data in. If it does not exist, will be created.
filename	The name to give to the file.
write	if TRUE (default), will write the df in the to folder, and df will be named filename.
return_path	Should the path be returned?
verbose	Should messages be displayed when creating a folder/file?

Value

Writes the file to the folder to/filename. Also returns the path to/filename if return_path == TRUE.

```
write_standardized_list
```

Write the standardized files

Description

Writes a list of files to a given location. If to does not exist, it is created, and if filename is not provided, a default standardized name is chosen.

Usage

```
write_standardized_list(
  df_list,
  filenames,
  to,
  write = TRUE,
  return_path = ifelse(write, FALSE, TRUE),
  verbose = TRUE
)
```

Arguments

df_list	The standardized files list to write. If the list is named, the subdirectory structure in the names will be used to replicate the subdirectory structure in the destination.
filenames	(Optional) vector of customized file names.
to	Destination folder to write to.
write	Should the result be written or only the path returned?
return_path	Should the path be returned?
verbose	Should messages be displayed when creating a folder/file?

Value

Writes the files to the folder to/filename1, to/filename2.... Also returns the paths to/filename1, to/filename2... if return_path == TRUE.

zooniverse	<i>Zooniverse sample data</i>
------------	-------------------------------

Description

A dataset mimicking typical Zooniverse data (randomized rows)

Usage

```
zooniverse
```

Format

A data frame with 100 rows and 24 variables:

```
capture_id character ID for the capture composed of season#site#roll#capture
season character Zooniverse season code
site character Camera ID
roll integer Roll (index for the camera service)
capture integer ID for the capture per camera/season/roll
capture_date_local character Date
capture_time_local character Time
zooniverse_url_0 character Url for first photo (mock URL)
zooniverse_url_1 character Url for second photo (mock URL)
zooniverse_url_2 character Url for third photo (mock URL)
subject_id integer Internal Zooniverse ID for the capture event (mock ID)
question__species character Species
question__count_max character Maximum count from the volunteers
question__count_median character Median count from the volunteers
question__count_min character Minimum count from the volunteers
question__standing double Proportion of users who declared a standing behavior
question__resting double Proportion of users who declared a resting behavior
question__moving double Proportion of users who declared a moving behaviour
question__eating double Proportion of users who declared a eating behaviour
question__interacting double Proportion of users who declared an interacting behaviour
question__young_present double Proportion of users who declared young presents
question__horns_visible double Proportion of users who declared horns on the picture
p_users_identified_this_species double Proportion of users who identified the consensus
species
pielous_evenness_index double Pielou evenness index
```

Index

- * **datasets**
 - digikam, [1](#)
 - standard, [8](#)
 - traptagger, [8](#)
 - zooniverse, [11](#)
- create_logger, [1](#)
- digikam, [1](#)
- get_final_filename, [2](#)
- get_logfile_name, [3](#)
- guess_classifier, [3](#)
- read_snapshot_files, [4](#)
- rename_standard, [5](#)
- standard, [8](#)
- standardize_snapshot_df, [5](#)
- standardize_snapshot_list, [6](#)
- standardize_species, [7](#)
- standardizeSnapshot, [7](#)
- traptagger, [8](#)
- write_log_message, [9](#)
- write_standardized_df, [9](#)
- write_standardized_list, [10](#)
- zooniverse, [11](#)