Package 'mpathr'

October 9, 2024

Title Easily Handling Data from the 'm-Path' Platform

Version 1.0.0

Description Provides tools for importing and cleaning Experience Sampling Method ('ESM') data collected via the 'm-Path' platform. The goal is to provide with a few utility functions to be able to read and perform some common operations in 'ESM' data collected through the 'm-Path' platform (<https://m-path.io/landing/>). Functions include raw data handling, format standardization, and basic data checks, as well as to calculate the response rate in data from 'ESM' studies.

License GPL (>= 3)

URL https://m-path.io, https://github.com/m-path-io/mpathr

Encoding UTF-8

Suggests knitr, rmarkdown, data.table, spelling, testthat (>= 3.0.0)

Config/testthat/edition 3

Imports cli, dplyr, lifecycle, readr, tidyr, rlang, ggplot2, jsonlite

Language en-US

BugReports https://github.com/m-path-io/mpathr/issues

Depends R (>= 4.0.0)

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

LazyData true

VignetteBuilder knitr

NeedsCompilation no

Author Merijn Mestdagh [aut, cre] (<https://orcid.org/0000-0001-5077-861X>), Lara Navarrete [aut], Koen Niemeijer [aut] (<https://orcid.org/0000-0002-0816-534X>), m-Path Software [cph]

Maintainer Merijn Mestdagh <merijn.mestdagh@m-path.io>

R topics documented:

`mpathr-package` *mpathr: Easily Handling Data from the 'm-Path' Platform*

Description

Provides tools for importing and cleaning Experience Sampling Method ('ESM') data collected via the 'm-Path' platform. The goal is to provide with a few utility functions to be able to read and perform some common operations in ESM data collected through the 'm-Path' platform (https://m-path.io/landing/). Functions include raw data handling, format standardization, and basic data checks, as well as to calculate the response rate in data from ESM studies.

Author(s)

Maintainer: Merijn Mestdagh <merijn.mestdagh@m-path.io> (ORCID)

Authors:

- Lara Navarrete <larann901@gmail.com>
- Koen Niemeijer <koen.niemeijer@kuleuven.be> (ORCID)

Other contributors:

- m-Path Software [copyright holder]

See Also

Useful links:

- https://m-path.io
- https://github.com/m-path-io/mpathr
- Report bugs at https://github.com/m-path-io/mpathr/issues

.mpath_locale *Locale to be used for m-Path data*

Description

Hard coded locale to be used for 'm-Path' data

Usage

```
.mpath_locale
```

Format

An object of class `locale` of length 7.

Value

Return a locale to be used in `readr::read_delim()` or friends.

example_data *Example m-path data*

Description

Contains the preprocessed example data for an m-path research study.

In the study, 20 participants completed 11 beeps over the course of 10 days. The study consisted of:

- An intake questionnaire, that participants answered at the study's start.
- A main questionnaire (10 times per day), where participants answered questions about their emotions and context at the time.
- An evening questionnaire (once, at the end of the day), about their emotions and activities throughout the day.

Each row corresponds to one beep sent during the study.

Usage

```
example_data
```

Format

A data frame with 1980 rows and 47 columns:

participant Participant identifier.

code Code the participants used to sign up for the study.

questionnaire The questionnaire that participants answered in that beep (it can be the main or the evening questionnaire).

scheduled Time stamp for when the notification was scheduled for, in unix time.

sent Time stamp for when the notification was sent, in unix time.

start Time stamp for when the notification was answered, in unix time. If the notification was never answered, this value is an NA.

stop Time stamp for when the notification was completed, in unix time. If the notification was never answered, this value is an NA.

phone_server_offset The difference between the phone time and the server time.

obs_n Observation number for each participant. Goes from 1 (first observation), to 110 (last observation of the study).

day_n Day number of the study, for the participant. Goes from 1 to 10.

obs_n_day Observation number within the day (for each participant). Goes from 1 to 11.

answered Logical, whether the beep was answered or not.

bpm_day Average heart rate per day. Note that unlike the rest of the variables, this corresponds to simulated data.

gender Participant's gender. 1 means 'Male', 2 means 'Female', 3 'Other'.

gender_string Participant's gender, as a string.

age Participant's age in years.

life_satisfaction Composite variable corresponding to participant's life satisfaction according to the Satisfaction With Life Scale (SWLS).

neuroticism Composite variable corresponding to participant's neuroticism according to the Big Five Inventory (BFI).

slider_happy Participants' self-reported happiness at the time of the beep. From 0 (not happy at all) to 100 (very happy).

slider_sad Participants' self-reported sadness at the time of the beep. From 0 (not sad at all) to 100 (very sad).

slider_angry Participants' self-reported anger at the time of the beep. From 0 (not angry at all) to 100 (very angry).

slider_relaxed Participants' self-reported relaxation at the time of the beep. From 0 (not relaxed at all) to 100 (very relaxed).

slider_anxious Participants' self-reported anxiety at the time of the beep. From 0 (not anxious at all) to 100 (very anxious).

slider_energetic Participants' self-reported energy at the time of the beep. From 0 (not energetic at all) to 100 (very energetic).

slider_tired Participants' self-reported tiredness at the time of the beep. From 0 (not tired at all) to 100 (very tired).

location_index Index corresponding to the participant's answer to the question "Where are you now?", from a list of multiple options.

location_string Text corresponding to the participant's selected location at the time of the beep.

company_index Index corresponding to the participant's answer to the question "With whom are you right now?", from a list of multiple options.

company_string Text corresponding to the participant's selected company at the time of the beep.

activity_index Index corresponding to the participant's answer to the question "What are you doing now?", from a list of multiple options.

activity_string Text corresponding to the participant's selected activity at the time of the beep.

step_count Step count between the previous answered beep and the current beep

evening_slider_happy Participants' happiness during the day, from 0 (not happy at all) to 100 (very happy).

evening_slider_sad Participants' sadness during the day, from 0 (not sad at all) to 100 (very sad).

evening_slider_angry Participants' anger during the day, from 0 (not angry at all) to 100 (very angry).

evening_slider_relaxed Participants' relaxation during the day, from 0 (not relaxed at all) to 100 (very relaxed).

evening_slider_anxious Participants' anxiety during the day, from 0 (not anxious at all) to 100 (very anxious).

evening_slider_energetic Participants' energy during the day, from 0 (not energetic at all) to 100 (very energetic).

evening_slider_tired Participants' tiredness during the day, from 0 (not tired at all) to 100 (very tired).

evening_stressful Participant's answer to whether something stressful had happened during the day. 1 means 'yes', 0 means 'no'.

evening_positive Participant's answer to whether something positive had happened during the day. 1 means 'yes', 0 means 'no'.

positive_description Explanation of the positive event (if participants responded 'yes' to the previous question).

stressful_description Explanation of the stressful event (if participants responded 'yes' to the previous question).

evening_activity_index Index corresponding to the participant's answer(s) to the question "What activities did you do today?", from a list of multiple options.

evening_activity_string Text corresponding to the participant's selected activities during the day.

delay_start_min Delay in minutes between the scheduled beep and the time the participants started the beep.

delay_end_min Time in minutes the participants took to fill in the beep (difference between the columns start and stop).

`is_opened_in_excel` *Check if an m-Path CSV file was opened in Excel*

Description

This function checks if an m-Path data file has previously been opened in Excel, in which case the whole file is wrapped in quotation marks. Actual quotation marks will then also be quoted, which is why we can't simply remove the outer quotes. Also, this function takes a single string as input (the first line of the file) instead of the file itself, because this would mean the file would have to be read twice. One time for this function, and then another time to get the column names.

Usage

```
is_opened_in_excel(line, call = rlang::caller_env())
```

Arguments

`line`	The first line of the file to check if it was opened in Excel.
`call`	The environment from which the function was called to display in the error message.

Value

Returns TRUE if the line is opened by Excel, otherwise an error informing the user of this problem.

`mpath_example` *Get path to m-Path example data*

Description

This function provides an easy way to access the m-Path example files.

Usage

```
mpath_example(file = NULL)
```

Arguments

`file`	the name of the file to be accessed. If NULL, the function will return a list of all the example files.

Value

a character string with the path to the m-Path example data

Examples

```
# Example 1: access 'example_basic.csv' data

mpath_example('example_basic.csv') # returns the full path to the file
'example_basic.csv'

# Example 2: list all the example files

mpath_example() # returns the example files as a vector
```

`plot_response_rate` *Plots response rate per day (and per participant)*

Description

This function returns a ggplot object with the response rate per day (x axis) and participant (color). Note that instead of using calendar dates, the function returns a plot grouped by the day inside the study for the participant.

Usage

```
plot_response_rate(data, valid_col, participant_col, time_col)
```

Arguments

data	data frame with data
valid_col	name of the column that stores whether the beep was answered or not
participant_col	name of the column that stores the participant id (or equivalent)
time_col	name of the column that stores the time of the beep

Value

a ggplot object with the response rate per day (x axis) and participant (color)

Examples

```
# load data
data(example_data)

# make plot with plot_response_rate
plot_response_rate(data = example_data,
time_col = sent,
participant_col = participant,
valid_col = answered)
# The resulting ggplot object can be formatted using ggplot2 functions (see ggplot2
# documentation).
```

---

read_meta_data *Read m-Path meta data*

Description

Internal function to read the meta data file for an m-Path file.

Usage

```
read_meta_data(meta_data)
```

Arguments

meta_data	A string with the path to the meta data file

Value

A tibble with the contents of the meta data file.

read_mpath *Read m-Path data*

Description

[Experimental] This function reads an 'm-Path' file into a dataframe.

Usage

```
read_mpath(file, meta_data)
```

Arguments

file	A string with the path to the m-Path file
meta_data	A string with the path to the meta data file

Details

Note that this function has been tested with the meta data version v.1.1. So it is advised to use that version of the meta data. (In 'm-Path', change the version in 'Export data' > "export version").

Value

A tibble with the 'm-Path' data.

Examples

```
# We can use the function mpath_examples to get the path to the example data
basic_path <- mpath_example(file ="example_basic.csv")
meta_path <- mpath_example("example_meta.csv")

data <- read_mpath(file = basic_path,
                meta_data = meta_path)
```

response_rate *Calculate response rate*

Description

Calculate response rate

Usage

```
response_rate(
  data,
  valid_col,
  participant_col,
  time_col = NULL,
  period_start = NULL,
  period_end = NULL
)
```

Arguments

data	data frame with data
valid_col	name of the column that stores whether the beep was answered or not
participant_col	name of the column that stores the participant id (or equivalent)
time_col	optional: name of the column that stores the time of the beep, as a 'POSIXct' object.
period_start	string representing the starting date to calculate response rates (optional). Accepts dates in the following formats: `yyyy-mm-dd` or`yyyy/mm/dd`.
period_end	period end to calculate response rates (optional).

Value

a data frame with the response rate for each participant, and the number of beeps used to calculate the response rate

Examples

```
# Example 1: calculate response rates for the whole study
# Get example data
data(example_data)

# Calculate response rate for each participant

# We don't specify time_col, period_start or period_end.
# Response rates will be based on all the participant's data
response_rate <- response_rate(data = example_data,
                               valid_col = answered,
                               participant_col = participant)

# Example 2: calculate response rates for a specific time period
data(example_data)

# Calculate response rate for each participant between dates
response_rate <- response_rate(data = example_data,
                               valid_col = answered,
                               participant_col = participant,
                               time_col = sent,
                               period_start = '2024-05-15',
                               period_end = '2024-05-31')

# Get participants with a response rate below 0.5
response_rate[response_rate$response_rate < 0.5,]
```

---

write_mpath *Write m-Path data to a CSV file*

Description

Write m-Path data to a CSV file

Usage

```
write_mpath(x, file)
```

Arguments

x	A data frame or tibble to write to disk.
file	File or connection to write to.

Value

Returns x invisibly.

Examples

```
data <- read_mpath(
  mpath_example("example_basic.csv"),
  mpath_example("example_meta.csv")
)

## Not run:
  write_mpath(data, "data.csv")

## End(Not run)
```

Index