

Package ‘suddengains’

May 23, 2020

Title Identify Sudden Gains in Longitudinal Data

Date 2020-05-20

Version 0.4.4

Description Identify sudden gains based on the three criteria outlined by Tang and DeRubeis (1999) <doi:10.1037/0022-006X.67.6.894> to a selection of repeated measures. Sudden losses, defined as the opposite of sudden gains can also be identified. Two different datasets can be created, one including all sudden gains/losses and one including one selected sudden gain/loss for each case. It can extract scores around sudden gains/losses. It can plot the average change around sudden gains/losses and trajectories of individual cases.

Depends R (>= 3.5.0)

License GPL-3

Encoding UTF-8

URL <https://milanwiedemann.github.io/suddengains/>

BugReports <https://github.com/milanwiedemann/suddengains/issues>

LazyData true

Imports dplyr (>= 0.8.0), tibble (>= 2.1.1), magrittr (>= 1.5), rlang (>= 0.3.4), stringr (>= 1.4.0), ggplot2 (>= 3.1.1), psych (>= 1.8.12), readr (>= 1.3.1), tidyr (>= 0.8.2), ggrepel (>= 0.8.0), patchwork (>= 1.0.0), forcats, naniar, scales

Suggests haven (>= 2.1.0), writexl (>= 1.1.0), knitr (>= 1.21), DT (>= 0.5), rmarkdown (>= 1.11), spelling (>= 2.1)

RoxygenNote 7.1.0

VignetteBuilder knitr

Language en-US

NeedsCompilation no

Author Milan Wiedemann [aut, cre] (<<https://orcid.org/0000-0003-1991-282X>>),
Graham M Thew [ctb] (<<https://orcid.org/0000-0003-2851-1315>>),
Richard Stott [ctb] (<<https://orcid.org/0000-0003-2533-5504>>),
Anke Ehlers [ctb, ths] (<<https://orcid.org/0000-0002-8742-0192>>),
Mental Health Research UK [fnd],
Wellcome Trust [fnd]

Maintainer Milan Wiedemann <milan.wiedemann@gmail.com>

Repository CRAN

Date/Publication 2020-05-22 22:40:03 UTC

R topics documented:

check_interval	2
count_intervals	4
create_byperson	5
create_bysg	8
define_crit1_cutoff	10
describe_sg	12
extract_values	13
identify_sg	14
identify_sl	16
plot_sg	19
plot_sg_intervals	21
plot_sg_trajectories	22
rename_sg_vars	24
select_cases	25
sgdata	26
sgdata_bad	28
write_byperson	29
write_bysg	32

Index	35
--------------	-----------

check_interval	<i>Check if a given interval is a sudden gain/loss</i>
----------------	--

Description

Checks if a specific session to session interval is a sudden gain/loss.

Usage

```
check_interval(
  pre_values,
  post_values,
  sg_crit1_cutoff,
  sg_crit2_pct = 0.25,
  sg_crit3 = TRUE,
  sg_crit3_alpha = 0.05,
  sg_crit3_adjust = TRUE,
  sg_crit3_critical_value = 2.776,
  identify = c("sg", "sl"),
  details = TRUE
)
```

Arguments

pre_values	Vector, three pre gain/loss values to be checked for a sudden gain/loss (n-2, n-1, n)
post_values	Vector, three post gain/loss values to be checked for a sudden gain/loss (n+1, n+2, n+3)
sg_crit1_cutoff	Numeric, specifying the cut-off value to be used for the first sudden gains criterion. The function <code>define_crit1_cutoff</code> can be used to calculate a cutoff value based on the Reliable Change Index (RCI; Jacobson & Truax, 1991). If set to NULL the first criterion wont be applied.
sg_crit2_pct	Numeric, specifying the percentage change to be used for the second sudden gains/losses criterion. If set to NULL the second criterion wont be applied.
sg_crit3	If set to TRUE the third criterion will be applied automatically adjusting the critical value for missingness. If set to FALSE the third criterion wont be applied.
sg_crit3_alpha	Numeric, alpha for the student t-test (two-tailed) to determine the critical value to be used for the third criterion. Degrees of freedom are based on the number of available data in the three sessions preceding the gain and the three sessions following the gain.
sg_crit3_adjust	Logical, specify whether critical value gets adjusted for missingness, see Lutz et al. (2013) and the documentation of this R package for further details. This argument is set to TRUE by default adjusting the critical value for missingness as described in the package documentation and Lutz et al. (2013): A critical value of 2.776 is used when all three data points before and after a potential gain are available, where one datapoint is missing either before or after a potential gain a critical value of 3.182 is used, and where one datapoint is missing both before and after the gain a critical value of 4.303 is used. If set to FALSE a critical value of 2.776 will instead be used for all comparisons, regardless of missingnes in the sequence of data points that are investigated for sudden gains.
sg_crit3_critical_value	Numeric, specifying the critical value to instead be used for all comparisons, regardless of missingnes in the sequence of data points that are investigated for potential sudden gains.
identify	String, specifying whether to identify sudden gains ("sg") or sudden losses ("sl").
details	Logical, details yes no?

Value

Information on whether a given interval is a sudden gain/loss

Examples

```
# Check interval for sudden gain using all 3 criteria
# No missing data, alpha = 0.05
check_interval(pre_values = c(32, 31, 33),
```

```

    post_values = c(5, 6, 7),
    sg_crit1_cutoff = 7,
    sg_crit2_pct = .25,
    sg_crit3 = TRUE,
    sg_crit3_alpha = .05,
    identify = "sg")

# No missing data, alpha = 0.01
check_interval(pre_values = c(32, 31, 33),
  post_values = c(5, 6, 7),
  sg_crit1_cutoff = 7,
  sg_crit2_pct = .25,
  sg_crit3 = TRUE,
  sg_crit3_alpha = .01,
  identify = "sg")

# Check intervall for sudden gain using only third criterion
# Some missing data, alpha = 0.01
check_interval(pre_values = c(NA,31,33),
  post_values = c(5, NA, 7),
  sg_crit1_cutoff = NULL,
  sg_crit2_pct = NULL,
  sg_crit3 = TRUE,
  sg_crit3_alpha = .01,
  identify = "sg")

# Check intervall for sudden loss using all three criteria
# Some missing data, alpha = 0.05
check_interval(pre_values = c(5, NA, 7),
  post_values = c(16, 12, 14),
  sg_crit1_cutoff = -7,
  sg_crit2_pct = .25,
  sg_crit3 = TRUE,
  sg_crit3_alpha = .05,
  identify = "sl")

```

count_intervals	<i>Count number of between-session intervals available to identify sudden gains</i>
-----------------	---

Description

Calculates the number of total between-session intervals present in the data set and the number of between-session intervals that are available to identify sudden gains talking into account the pattern of missing data.

Usage

```
count_intervals(data, id_var_name, sg_var_list, identify_sg_1to2 = FALSE)
```

Arguments

<code>data</code>	A data set in wide format including an ID variable and variables for each measurement point.
<code>id_var_name</code>	String, specifying the name of the ID variable. Each row should have a unique value.
<code>sg_var_list</code>	Vector, specifying the variable names of each measurement point sequentially.
<code>identify_sg_1to2</code>	Logical, indicating whether to identify sudden losses from measurement point 1 to 2. If set to TRUE, this implies that the first variable specified in <code>sg_var_list</code> represents a baseline measurement point, e.g. pre-intervention assessment.

Value

List with values for:

- `total_between_sess_intervals`: The total number of between-session intervals present in the data set, NAs are also included here. This multiplies the number of cases (rows) with the number of specified between-session intervals: $nrows * (length(sg_var_list) - 1)$.
- `total_between_sess_intervals_sg`: The total number of between-session intervals where sudden gains can theoretically, NAs are also included here. This multiplies the number of cases (rows) with the number of between-session intervals where sudden gains can be identified using the 3 original criteria: $nrows * (length(sg_var_list) - 3)$.
- `available_between_sess_intervals_sg`: The total number of between-session intervals that can be analysed for sudden gains taking into account the pattern of missing data.
- `not_available_between_sess_intervals_sg`: The total number of between-session intervals that can not be analysed for sudden gains due to the pattern of missing data.

Examples

```
# Count between session intervals in "sgdata"
count_intervals(data = sgdata,
               id_var_name = "id",
               sg_var_list = c("bdi_s1", "bdi_s2", "bdi_s3", "bdi_s4",
                              "bdi_s5", "bdi_s6", "bdi_s7", "bdi_s8",
                              "bdi_s9", "bdi_s10", "bdi_s11", "bdi_s12"))
```

<code>create_byperson</code>	<i>Create a data set with one gain per person</i>
------------------------------	---

Description

This function returns a wide data set with one row for each case in `data`. The data set includes variables indicating whether each case experienced a sudden gain/loss or not, values around the period of each gain/loss, and descriptives. For cases with no sudden gain/loss the descriptive variables are coded as missing (NA).

Usage

```

create_byperson(
  data,
  sg_crit1_cutoff,
  id_var_name,
  sg_var_list,
  tx_start_var_name,
  tx_end_var_name,
  sg_measure_name,
  multiple_sg_select = c("first", "last", "smallest", "largest"),
  data_is_bysg = FALSE,
  identify = c("sg", "sl"),
  sg_crit2_pct = 0.25,
  sg_crit3 = TRUE,
  sg_crit3_alpha = 0.05,
  sg_crit3_adjust = TRUE,
  sg_crit3_critical_value = 2.776,
  identify_sg_1to2 = FALSE
)

```

Arguments

<code>data</code>	A data set in wide format including an ID variable and variables for each measurement point.
<code>sg_crit1_cutoff</code>	Numeric, specifying the cut-off value to be used for the first sudden gains criterion. The function <code>define_crit1_cutoff</code> can be used to calculate a cutoff value based on the Reliable Change Index (RCI; Jacobson & Truax, 1991). If set to NULL the first criterion wont be applied.
<code>id_var_name</code>	String, specifying the name of the ID variable.
<code>sg_var_list</code>	Vector, specifying the variable names of each measurement point sequentially.
<code>tx_start_var_name</code>	String, specifying the variable name of the first measurement point of the intervention.
<code>tx_end_var_name</code>	String, specifying the variable name of the last measurement point of the intervention.
<code>sg_measure_name</code>	String, specifying the name of the measure used to identify sudden gains/losses.
<code>multiple_sg_select</code>	String, specifying which sudden gain/loss to select for this data set if more than one gain/loss was identified per case. Options are: "first", "last", "smallest", or "largest". The default is to select the first sudden gain ("first") if someone experienced multiple gains.
<code>data_is_bysg</code>	Logical, specifying whether the data set in the data argument is a bysg datasets created using the <code>create_bysg</code> function.

identify	String, specifying whether to identify sudden gains ("sg") using <code>identify_sg</code> or sudden losses ("sl") using <code>identify_sl</code> . The default is to identify sudden gains ("sg").
sg_crit2_pct	Numeric, specifying the percentage change to be used for the second sudden gains/losses criterion. If set to NULL the second criterion wont be applied.
sg_crit3	If set to TRUE the third criterion will be applied automatically adjusting the critical value for missingness. If set to FALSE the third criterion wont be applied.
sg_crit3_alpha	Numeric, alpha for the student t-test (two-tailed) to determine the critical value to be used for the third criterion. Degrees of freedom are based on the number of available data in the three sessions preceding the gain and the three sessions following the gain.
sg_crit3_adjust	Logical, specify whether critical value gets adjusted for missingness, see Lutz et al. (2013) and the documentation of this R package for further details. This argument is set to TRUE by default adjusting the critical value for missingness as described in the package documentation and Lutz et al. (2013): A critical value of 2.776 is used when all three data points before and after a potential gain are available, where one datapoint is missing either before or after a potential gain a critical value of 3.182 is used, and where one datapoint is missing both before and after the gain a critical value of 4.303 is used (for <code>sg_crit3_alpha = 0.05</code>). If set to FALSE the critical value setin <code>sg_crit3_critical_value</code> will instead be used for all comparisons, regardless of missingnes in the sequence of data points that are investigated for potential sudden gains.
sg_crit3_critical_value	Numeric, specifying the critical value to instead be used for all comparisons, regardless of missingnes in the sequence of data points that are investigated for potential sudden gains.
identify_sg_1to2	Logical, indicating whether to identify sudden losses from measurement point 1 to 2. If set to TRUE, this implies that the first variable specified in <code>sg_var_list</code> represents a baseline measurement point, e.g. pre-intervention assessment.

Value

A wide data set with one row per case (`id_var_name`) in data.

References

Tang, T. Z., & DeRubeis, R. J. (1999). Sudden gains and critical sessions in cognitive-behavioral therapy for depression. *Journal of Consulting and Clinical Psychology*, 67(6), 894–904. <https://doi.org/10.1037/0022-006X.67.6.894>.

Examples

```
# Create byperson data set, selecting the largest gain in case of muliple gains
create_byperson(data = sgdata,
                sg_crit1_cutoff = 7,
                id_var_name = "id",
```

```

tx_start_var_name = "bdi_s1",
tx_end_var_name = "bdi_s12",
sg_var_list = c("bdi_s1", "bdi_s2", "bdi_s3",
               "bdi_s4", "bdi_s5", "bdi_s6",
               "bdi_s7", "bdi_s8", "bdi_s9",
               "bdi_s10", "bdi_s11", "bdi_s12"),
sg_measure_name = "bdi",
multiple_sg_select = "largest")

```

create_bysg

Create a data set with one row for each sudden gain/loss

Description

This function returns a wide data set with one row for each sudden gain/loss and assigns a unique identifier to each. The data set includes variables indicating values around the period of each gain/loss, and calculates descriptives of each gain/loss.

Usage

```

create_bysg(
  data,
  sg_crit1_cutoff,
  id_var_name,
  sg_var_list,
  tx_start_var_name,
  tx_end_var_name,
  sg_measure_name,
  sg_crit2_pct = 0.25,
  sg_crit3 = TRUE,
  sg_crit3_alpha = 0.05,
  sg_crit3_adjust = TRUE,
  sg_crit3_critical_value = 2.776,
  identify = c("sg", "s1"),
  identify_sg_1to2 = FALSE
)

```

Arguments

data	A data set in wide format including an ID variable and variables for each measurement point.
sg_crit1_cutoff	Numeric, specifying the cut-off value to be used for the first sudden gains criterion. The function <code>define_crit1_cutoff</code> can be used to calculate a cutoff value based on the Reliable Change Index (RCI; Jacobson & Truax, 1991). If set to NULL the first criterion wont be applied.
id_var_name	String, specifying the name of the ID variable.

<code>sg_var_list</code>	Vector, specifying the variable names of each measurement point sequentially.
<code>tx_start_var_name</code>	String, specifying the variable name of the first measurement point of the intervention.
<code>tx_end_var_name</code>	String, specifying the variable name of the last measurement point of the intervention.
<code>sg_measure_name</code>	String, specifying the name of the measure used to identify sudden gains/losses.
<code>sg_crit2_pct</code>	Numeric, specifying the percentage change to be used for the second sudden gains/losses criterion. If set to NULL the second criterion wont be applied.
<code>sg_crit3</code>	If set to TRUE the third criterion will be applied automatically adjusting the critical value for missingness. If set to FALSE the third criterion wont be applied.
<code>sg_crit3_alpha</code>	Numeric, alpha for the student t-test (two-tailed) to determine the critical value to be used for the third criterion. Degrees of freedom are based on the number of available data in the three sessions preceding the gain and the three sessions following the gain.
<code>sg_crit3_adjust</code>	Logical, specify whether critical value gets adjusted for missingness, see Lutz et al. (2013) and the documentation of this R package for further details. This argument is set to TRUE by default adjusting the critical value for missingness as described in the package documentation and Lutz et al. (2013): A critical value of 2.776 is used when all three data points before and after a potential gain are available, where one datapoint is missing either before or after a potential gain a critical value of 3.182 is used, and where one datapoint is missing both before and after the gain a critical value of 4.303 is used (for <code>sg_crit3_alpha = 0.05</code>). If set to FALSE the critical value set in <code>sg_crit3_critical_value</code> will instead be used for all comparisons, regardless of missingnes in the sequence of data points that are investigated for potential sudden gains.
<code>sg_crit3_critical_value</code>	Numeric, specifying the critical value to instead be used for all comparisons, regardless of missingnes in the sequence of data points that are investigated for potential sudden gains.#'
<code>identify</code>	String, specifying whether to identify sudden gains ("sg") using <code>identify_sg</code> or sudden losses ("sl") using <code>identify_sl</code> . The default is to identify sudden gains ("sg").
<code>identify_sg_1to2</code>	Logical, indicating whether to identify sudden losses from measurement point 1 to 2. If set to TRUE, this implies that the first variable specified in <code>sg_var_list</code> represents a baseline measurement point, e.g. pre-intervention assessment.

Value

A wide data set with one row per sudden gain/loss.

References

Tang, T. Z., & DeRubeis, R. J. (1999). Sudden gains and critical sessions in cognitive-behavioral therapy for depression. *Journal of Consulting and Clinical Psychology*, 67(6), 894–904. <https://doi.org/10.1037/0022-006X.67.6.894>.

Examples

```
# Create bypsg data set
create_bysg(data = sgdata,
  sg_crit1_cutoff = 7,
  id_var_name = "id",
  tx_start_var_name = "bdi_s1",
  tx_end_var_name = "bdi_s12",
  sg_var_list = c("bdi_s1", "bdi_s2", "bdi_s3",
    "bdi_s4", "bdi_s5", "bdi_s6",
    "bdi_s7", "bdi_s8", "bdi_s9",
    "bdi_s10", "bdi_s11", "bdi_s12"),
  sg_measure_name = "bdi")
```

define_crit1_cutoff *Define cut-off value for first SG criterion*

Description

Define a cut-off value for the first sudden gains criterion based on the Reliable Change Index (RCI; Jacobson & Truax, 1991) using an estimate for the standard deviation (sd) of the normal population and the reliability of the scale. These values can be entered manually using the arguments `sd` and `reliability` or extracted from data using the arguments `data_sd` and `data_reliability`. This function calculates the standard error of measurement (`se`), the standard error of the difference (`sdiff`) and a value that classifies as reliable change (`reliable_change_value`) based on the Reliable Change Index (RCI; Jacobson & Truax, 1991).

$$se = sd * \sqrt{(1 - reliability)}$$

$$sdiff = \sqrt{(2 * se^2)}$$

$$reliablechangevalue = 1.96 * sdiff$$

Usage

```
define_crit1_cutoff(
  sd = NULL,
  reliability = NULL,
  data_sd = NULL,
  data_reliability = NULL
)
```

Arguments

sd	Numeric, standard deviation of normal population or standard deviation at baseline. This argument is not needed if a vector with pretreatment scores is specified in the data_sd argument.
reliability	Numeric, between 0 and 1 indicating reliability of the scale. This argument is not needed if item-by-item data is specified in the data_reliability argument.
data_sd	A vector with pretreatment values. This argument is not needed if the standard deviation is specified in the sd argument.
data_reliability	A dataset in wide format (one row for each individual and one column for each item) including only the item-by-item scores of the SG measure (no ID variable). According to Jacobson & Truax (1991) the test-retest reliability should be used. Martinovich et al. (1996) suggest that the internal consistency (Cronbach's alpha) can be used instead of the test-retest reliability and may be more appropriate for estimating the standard error in some cases. This argument is not needed if the reliability is specified in the reliability argument.

Value

A list with estimates the for standard error of measurement (se), the standard error of the difference (sdiff) and a value that classifies as reliable change (reliable_change_value).

References

- Jacobson, N. S., & Truax, P. A. (1991). Clinical significance: A statistical approach to defining meaningful change in psychotherapy research. *Journal of Consulting and Clinical Psychology*, 59 (1), 12-19. <https://doi.org/10.1037/0022-006X.59.1.12>.
- Martinovich, Z., Saunders, S., & Howard, K. (1996). Some Comments on "Assessing Clinical Significance". *Psychotherapy Research*, 6(2), 124–132. <https://doi.org/10.1080/10503309612331331648>.
- Stiles et al. (2003). Early sudden gains in psychotherapy under routine clinic conditions: Practice-based evidence. *Journal of Consulting and Clinical Psychology*, 71 (1), 14-21. <https://doi.org/10.1037/0022-006X.71.1.14>.

Examples

```
# Define cut-off value for first SG criterion
# In this example the standard deviation and the reliability are specified manually
define_crit1_cutoff(sd = 10.5,
                    reliability = 0.931)

# In this example the reliability is specified manually
# The standard deviation of the variable "bdi_s0" in the dataset "sgdata" gets calculated
define_crit1_cutoff(data_sd = sgdata$bdi_s0,
                    reliability = 0.931)
```

`describe_sg`*Show descriptives for the sudden gains datasets*

Description

Descriptives might differ between the `bysg` and `byperson` data sets depending on whether multiple gains are present.

Usage

```
describe_sg(data, sg_data_structure = c("bysg", "byperson"))
```

Arguments

`data` A `bysg` or `byperson` dataset created using the function `create_bysg` or `create_byperson`.
`sg_data_structure` String, indicating whether the input data is a `bysg` or `byperson` dataset.

Value

A list, showing basic descriptive statistics for sudden gains within the dataset specified. Note that some numbers (e.g. percentages) will be different depending which dataset is selected, because where a participant has multiple gains, only one is selected for the `byperson` dataset. The list includes values for:

- `total_n`: number of rows in input dataset
- `sg_total_n`: total number of sudden gains
- `sg_n`: number of people who experienced a sudden gain (`byperson` dataset only)
- `sg_pct`: percentage of people in the input dataset who experienced a sudden gain
- `sg_multiple_n`: number of people who experienced a sudden gain (`byperson` dataset only)
- `sg_multiple_pct`: percentage of people in the input dataset who experienced more than one sudden gain
- `sg_reversal_n`: number of sudden gains that later meet the criteria for a reversal
- `sg_reversal_pct`: percentage of sudden gains that later meet the criteria for a reversal
- `sg_magnitude_m`: mean magnitude of the sudden gains observed
- `sg_magnitude_sd`: standard deviation of the magnitude of the sudden gains observed

Examples

```
# Create bysg dataset
bysg <- create_bysg(data = sgdata,
  sg_crit1_cutoff = 7,
  id_var_name = "id",
  tx_start_var_name = "bdi_s1",
  tx_end_var_name = "bdi_s12",
```

```

sg_var_list = c("bdi_s1", "bdi_s2", "bdi_s3",
               "bdi_s4", "bdi_s5", "bdi_s6",
               "bdi_s7", "bdi_s8", "bdi_s9",
               "bdi_s10", "bdi_s11", "bdi_s12"),
sg_measure_name = "bdi")

# Describe bysg dataset
describe_sg(data = bysg,
            sg_data_structure = "bysg")

```

extract_values	<i>Extract values around the sudden gain</i>
----------------	--

Description

Extract values of measures around the time of a sudden gain.

Usage

```

extract_values(
  data,
  id_var_name,
  extract_var_list,
  sg_session_n_var_name = "sg_session_n",
  extract_measure_name = "x",
  start_numbering = 1,
  add_to_data = TRUE
)

```

Arguments

data	A bysg or byperson data set in wide format with the variable <code>sg_session_n</code> and all variables specified in <code>extract_var_list</code> .
id_var_name	String, specifying the name of the ID variable.
extract_var_list	List or vector, specifying the variable names of session to session scores to extract from. If this is a list, the name of each element will be used when creating new variables. Note that each element of this list must have the same number of repeated measures as specified in <code>sg_var_list</code> when creating the sudden gains data set.
sg_session_n_var_name	String, specifying variable name that contains information about the pregain session number. If the sudden gains data set was created using the <code>suddengains</code> R package, the default argument <code>"sg_session_n"</code> should be used.
extract_measure_name	String, if <code>extract_var_list</code> is a vector, this string will be used as the when creating new variables of the extracted measures.

start_numbering	Numeric, set to by default 1. Change to 0 if a pre-treatment (e.g. baseline assessment) measurement point is included in <code>extract_var_list</code> .
add_to_data	Logical, if set to TRUE, the extracted values are added as new variables to the input dataset. If set to false, only the ID variable and all extracted values will be returned.

Value

A wide dataset with values for `extract_measure_name` around the sudden gain.

Examples

```
# Create bysg dataset
bysg <- create_bysg(data = sgdata,
  sg_crit1_cutoff = 7,
  id_var_name = "id",
  tx_start_var_name = "bdi_s1",
  tx_end_var_name = "bdi_s12",
  sg_var_list = c("bdi_s1", "bdi_s2", "bdi_s3",
    "bdi_s4", "bdi_s5", "bdi_s6",
    "bdi_s7", "bdi_s8", "bdi_s9",
    "bdi_s10", "bdi_s11", "bdi_s12"),
  sg_measure_name = "bdi")

# For bysg dataset select "id" and "rq" variables first
sgdata_rq <- sgdata %>%
  dplyr::select(id, rq_s0:rq_s12)

# Join them with the sudden gains data set, here "bysg"
bysg_rq <- bysg %>%
  dplyr::left_join(sgdata_rq, by = "id")

# Extract "rq" scores around sudden gains on "bdi" in the bysg dataset
bysg_rq <- extract_values(data = bysg_rq,
  id_var_name = "id_sg",
  extract_var_list = c("rq_s1", "rq_s2", "rq_s3", "rq_s4",
    "rq_s5", "rq_s6", "rq_s7", "rq_s8",
    "rq_s9", "rq_s10", "rq_s11", "rq_s12"),
  extract_measure_name = "rq",
  add_to_data = TRUE)
```

identify_sg

Identify sudden gains.

Description

Function to identify sudden gains in longitudinal data structured in wide format.

Usage

```

identify_sg(
  data,
  id_var_name,
  sg_var_list,
  sg_crit1_cutoff,
  sg_crit2_pct = 0.25,
  sg_crit3 = TRUE,
  sg_crit3_alpha = 0.05,
  sg_crit3_adjust = TRUE,
  sg_crit3_critical_value = 2.776,
  identify_sg_1to2 = FALSE,
  crit123_details = FALSE
)

```

Arguments

<code>data</code>	A data set in wide format including an ID variable and variables for each measurement point.
<code>id_var_name</code>	String, specifying the name of the ID variable. Each row should have a unique value.
<code>sg_var_list</code>	Vector, specifying the variable names of each measurement point sequentially.
<code>sg_crit1_cutoff</code>	Numeric, specifying the cut-off value to be used for the first sudden gains criterion. The function <code>define_crit1_cutoff</code> can be used to calculate a cutoff value based on the Reliable Change Index (RCI; Jacobson & Truax, 1991). If set to NULL the first criterion wont be applied.
<code>sg_crit2_pct</code>	Numeric, specifying the percentage change to be used for the second sudden gains criterion. If set to NULL the second criterion wont be applied.
<code>sg_crit3</code>	If set to TRUE the third criterion will be applied automatically adjusting the critical value for missingness. If set to FALSE the third criterion wont be applied.
<code>sg_crit3_alpha</code>	Numeric, alpha for the two-tailed student t-test to determine the critical value to be used for the third criterion. Degrees of freedom are based on the number of available data in the three sessions preceding the gain and the three sessions following the gain.
<code>sg_crit3_adjust</code>	Logical, specify whether critical value gets adjusted for missingness, see Lutz et al. (2013) and the documentation of this R package for further details. This argument is set to TRUE by default adjusting the critical value for missingness as described in the package documentation and Lutz et al. (2013): A critical value of 2.776 is used when all three data points before and after a potential gain are available, where one datapoint is missing either before or after a potential gain a critical value of 3.182 is used, and where one datapoint is missing both before and after the gain a critical value of 4.303 is used (for <code>sg_crit3_alpha = 0.05</code>). If set to FALSE the critical value set in <code>sg_crit3_critical_value</code> will instead be used for all comparisons, regardless of missingnes in the sequence of data points that are investigated for potential sudden gains.

sg_crit3_critical_value

Numeric, specifying the critical value to instead be used for all comparisons, regardless of missingness in the sequence of data points that are investigated for potential sudden gains.

identify_sg_1to2

Logical, indicating whether to identify sudden gains from measurement point 1 to 2. If set to TRUE, this implies that the first variable specified in `sg_var_list` represents a baseline measurement point, e.g. pre-intervention assessment.

crit123_details

Logical, if set to TRUE this function returns information about which of the three criteria (e.g. "sg_crit1_2to3", "sg_crit2_2to3", and "sg_crit3_2to3") are met for each session to session interval for all cases. Variables named "sg_2to3", "sg_3to4" summarise all criteria that were selected to identify sudden gains.

Value

A wide data set indicating whether sudden gains are present for each session to session interval for all cases in data.

References

Lutz, W., Ehrlich, T., Rubel, J., Hallwachs, N., Röttger, M.-A., Jorasz, C., ... Tschitsaz-Stucki, A. (2013). The ups and downs of psychotherapy: Sudden gains and sudden losses identified with session reports. *Psychotherapy Research*, 23(1), 14–24. <https://doi.org/10.1080/10503307.2012.693837>.

Tang, T. Z., & DeRubeis, R. J. (1999). Sudden gains and critical sessions in cognitive-behavioral therapy for depression. *Journal of Consulting and Clinical Psychology*, 67(6), 894–904. <https://doi.org/10.1037/0022-006X.67.6.894>.

Examples

```
# Identify sudden gains
identify_sg(data = sgdata,
            sg_crit1_cutoff = 7,
            id_var_name = "id",
            sg_var_list = c("bdi_s1", "bdi_s2", "bdi_s3",
                           "bdi_s4", "bdi_s5", "bdi_s6",
                           "bdi_s7", "bdi_s8", "bdi_s9",
                           "bdi_s10", "bdi_s11", "bdi_s12"))
```

identify_sl

Identify sudden losses.

Description

Function to identify sudden losses in longitudinal data structured in wide format.

Usage

```

identify_sl(
  data,
  id_var_name,
  sg_var_list,
  sg_crit1_cutoff,
  sg_crit2_pct = 0.25,
  sg_crit3 = TRUE,
  sg_crit3_alpha = 0.05,
  sg_crit3_adjust = TRUE,
  sg_crit3_critical_value = 2.776,
  identify_sg_1to2 = FALSE,
  crit123_details = FALSE
)

```

Arguments

<code>data</code>	A data set in wide format including an ID variable and variables for each measurement point.
<code>id_var_name</code>	String, specifying the name of the ID variable. Each row should have a unique value.
<code>sg_var_list</code>	Vector, specifying the variable names of each measurement point sequentially.
<code>sg_crit1_cutoff</code>	Numeric, specifying the negative cut-off value to be used for the first sudden losses criterion, see examples below. The function <code>define_crit1_cutoff</code> can be used to calculate a cutoff value based on the Reliable Change Index (RCI; Jacobson & Truax, 1991). If set to NULL the first criterion won't be applied.
<code>sg_crit2_pct</code>	Numeric, specifying the percentage change to be used for the second sudden losses criterion. If set to NULL the first criterion won't be applied.
<code>sg_crit3</code>	Logical, if set to TRUE the third criteria will be applied automatically adjusting the critical value for missingness. If set to FALSE the third criterion won't be applied.
<code>sg_crit3_alpha</code>	Numeric, alpha for the student t-test (two-tailed) to determine the critical value to be used for the third criterion. Degrees of freedom are based on the number of available data in the three sessions preceding the loss and the three sessions following the loss.
<code>sg_crit3_adjust</code>	Logical, specify whether critical value gets adjusted for missingness, see Lutz et al. (2013) and the documentation of this R package for further details. This argument is set to TRUE by default adjusting the critical value for missingness as described in the package documentation and Lutz et al. (2013): A critical value of 2.776 is used when all three data points before and after a potential gain are available, where one datapoint is missing either before or after a potential gain a critical value of 3.182 is used, and where one datapoint is missing both before and after the gain a critical value of 4.303 is used (for <code>sg_crit3_alpha = 0.05</code>). If set to FALSE the critical value set in <code>sg_crit3_critical_value</code> will instead

be used for all comparisons, regardless of missingnes in the sequence of data points that are investigated for potential sudden gains.

`sg_crit3_critical_value`

Numeric, if the argument `sg_crit3_adjust = FALSE`, specifying the critical value to instead be used for all comparisons, regardless of missingnes in the sequence of data points that are investigated for potential sudden gains.

`identify_sg_1to2`

Logical, indicating whether to identify sudden losses from measurement point 1 to 2. If set to `TRUE`, this implies that the first variable specified in `sg_var_list` represents a baseline measurement point, e.g. pre-intervention assessment.

`crit123_details`

Logical, if set to `TRUE` this function returns information about which of the three criteria (e.g. `"sg_crit1_2to3"`, `"sg_crit2_2to3"`, and `"sg_crit3_2to3"`) are met for each session to session interval for all cases. Variables named `"sg_2to3"`, `"sg_3to4"` summarise all criteria that were selected to identify sudden gains.

Value

A wide data set indicating whether sudden losses are present for each session to session interval for all cases in data.

References

Lutz, W., Ehrlich, T., Rubel, J., Hallwachs, N., Röttger, M.-A., Jorasz, C., ... Tschitsaz-Stucki, A. (2013). The ups and downs of psychotherapy: Sudden gains and sudden losses identified with session reports. *Psychotherapy Research*, 23(1), 14–24. <https://doi.org/10.1080/10503307.2012.693837>.

Tang, T. Z., & DeRubeis, R. J. (1999). Sudden gains and critical sessions in cognitive-behavioral therapy for depression. *Journal of Consulting and Clinical Psychology*, 67(6), 894–904. <https://doi.org/10.1037/0022-006X.67.6.894>.

Examples

```
# Identify sudden losses
identify_sl(data = sgdata,
  # Negative cut-off value to identify sudden losses
  sg_crit1_cutoff = -7,
  id_var_name = "id",
  sg_var_list = c("bdi_s1", "bdi_s2", "bdi_s3",
    "bdi_s4", "bdi_s5", "bdi_s6",
    "bdi_s7", "bdi_s8", "bdi_s9",
    "bdi_s10", "bdi_s11", "bdi_s12"))
```

plot_sg	<i>Plot average change in variables around the sudden gain</i>
---------	--

Description

Generates a plot of the mean values around the sudden gain using `ggplot`. This can be used to plot the primary outcome or secondary measures. The parameters starting with "group" allow to plot the average gain magnitude by group. Further `ggplot2` components can be added using `+` following this function.

Usage

```
plot_sg(
  data,
  id_var_name,
  tx_start_var_name,
  tx_end_var_name,
  sg_pre_post_var_list,
  ylab,
  xlab = "Session",
  colour_single = "#239b89ff",
  colour_group = c("viridis", "ggplot", "grey"),
  viridis_option = c("D", "A", "B", "C"),
  viridis_begin = 0,
  viridis_end = 1,
  group_var_name = NULL,
  group_levels = NULL,
  group_labels = NULL,
  group_title = NULL,
  apaish = TRUE
)
```

Arguments

<code>data</code>	A <code>bysg</code> or <code>byperson</code> dataset created using the function create_bysg or create_byperson .
<code>id_var_name</code>	String, specifying the name of the ID variable.
<code>tx_start_var_name</code>	String, specifying the variable name of the first measurement point of the intervention.
<code>tx_end_var_name</code>	String, specifying the variable name of the last measurement point of the intervention.
<code>sg_pre_post_var_list</code>	Vector, specifying the variable names of the 3 measurement points before, and the 3 after the sudden gain, for the measure being plotted.
<code>ylab</code>	String, specifying the label for the y axis i.e. the name of the measure being plotted.

xlab	String, specifying the label for the x axis, e.g. "Session".
colour_single	String, specifying the colour of the plot for one group.
colour_group	String, specifying the discrete colour palette to be used for the groups.
viridis_option	String specifying the colour option for discrete viridis palette, see scale_fill_viridis_d .
viridis_begin	Numeric, specifying hue between 0 and 1 at which the viridis colormap begins, see scale_fill_viridis_d .
viridis_end	Numeric, specifying hue between 0 and 1 at which the viridis colormap ends, see scale_fill_viridis_d .
group_var_name	String, specifying the variable name of the group variable.
group_levels	Vector, specifying the levels as numeric for the groups in group_var_name.
group_labels	Vector, specifying the label names as strings for the groups in group_var_name.
group_title	String, specifying the title that will be used for the groups specified in group_labels.
apaish	Logical, make plot APA publishable.

Value

A plot of the mean values around the sudden gain, for the measure specified.

Examples

```
# First create a bysg (or byperson) dataset
bysg <- create_bysg(data = sgdata,
  sg_crit1_cutoff = 7,
  id_var_name = "id",
  tx_start_var_name = "bdi_s1",
  tx_end_var_name = "bdi_s12",
  sg_var_list = c("bdi_s1", "bdi_s2", "bdi_s3",
    "bdi_s4", "bdi_s5", "bdi_s6",
    "bdi_s7", "bdi_s8", "bdi_s9",
    "bdi_s10", "bdi_s11", "bdi_s12"),
  sg_measure_name = "bdi")

# Plot average change of BDI values around the period of the sudden gain
plot_sg(data = bysg,
  id_var_name = "id",
  tx_start_var_name = "bdi_s1",
  tx_end_var_name = "bdi_s12",
  sg_pre_post_var_list = c("sg_bdi_2n", "sg_bdi_1n", "sg_bdi_n",
    "sg_bdi_n1", "sg_bdi_n2", "sg_bdi_n3"),
  ylab = "BDI", xlab = "Session")
```

plot_sg_intervals	<i>Plot summary of available data per time point and analysed session to session intervals</i>
-------------------	--

Description

Plot summary of available data per time point and analysed session to session intervals

Usage

```
plot_sg_intervals(data, id_var_name, sg_var_list, identify_sg_1to2 = FALSE)
```

Arguments

data	A data set in wide format including an ID variable and variables for each measurement point.
id_var_name	String, specifying the name of the ID variable. Each row should have a unique value.
sg_var_list	Vector, specifying the variable names of each measurement point sequentially.
identify_sg_1to2	Logical, indicating whether to identify sudden losses from measurement point 1 to 2. If set to TRUE, this implies that the first variable specified in <code>sg_var_list</code> represents a baseline measurement point, e.g. pre-intervention assessment.

Value

Plot showing percentage of available data per time point and percentage of session to session intervals that were analysed for sudden gains.

Examples

```
# Create plot
plot_sg_intervals(data = sgdata,
  id_var_name = "id",
  sg_var_list = c("bdi_s1", "bdi_s2", "bdi_s3",
    "bdi_s4", "bdi_s5", "bdi_s6",
    "bdi_s7", "bdi_s8", "bdi_s9",
    "bdi_s10", "bdi_s11", "bdi_s12"))
```

plot_sg_trajectories *Plot individual trajectories*

Description

Plot individual trajectories of selected cases using `ggplot`. This function can be combined with a filter command to explore the trajectories of individual or groups of cases. Further `ggplot2` components can be added using `+` following this function.

Usage

```
plot_sg_trajectories(  
  data,  
  id_var,  
  var_list,  
  select_id_list = NULL,  
  select_n = NULL,  
  show_id = TRUE,  
  show_legend = TRUE,  
  legend_title = "ID",  
  id_label_size = 2,  
  connect_missing = TRUE,  
  colour = c("viridis", "ggplot", "grey"),  
  viridis_option = c("D", "A", "B", "C"),  
  viridis_begin = 0,  
  viridis_end = 1,  
  line_alpha = 1,  
  point_alpha = 1,  
  xlab = "X",  
  ylab = "Y",  
  scale_x_num = FALSE,  
  scale_x_num_start = 1,  
  apaish = TRUE,  
  ...  
)
```

Arguments

<code>data</code>	Dataset in wide format.
<code>id_var</code>	String, specifying ID variable.
<code>var_list</code>	Vector, specifying variable names to be plotted in sequential order.
<code>select_id_list</code>	Vector, specifying case IDs to be plotted.
<code>select_n</code>	Numeric, specifying number of randomly selected cases to be plotted.
<code>show_id</code>	Logical, specifying whether or not to show ID variables inside the plot near the first measurement point.

show_legend	Logical, specifying whether or not a legend of all IDs.
legend_title	String, specifying the title of legend, by default the variable name of id_var will be shown.
id_label_size	Numeric, specifying the size of the ID label, if show_id = TRUE.
connect_missing	Logical, specifying whether to connect points across missing values.
colour	String, specifying the discrete colour palette to be used.
viridis_option	String, specifying the colour option for discrete viridis palette, if colour = "viridis". See scale_fill_viridis_d for more details.
viridis_begin	Numeric, specifying hue between 0 and 1 at which the viridis colormap begins, if colour = "viridis". See scale_fill_viridis_d for more details.
viridis_end	Numeric, specifying hue between 0 and 1 at which the viridis colormap ends, if colour = "viridis". See scale_fill_viridis_d for more details.
line_alpha	Numeric, specifying alpha (transparency) of lines.
point_alpha	Numeric, specifying alpha (transparency) of points.
xlab	String for x axis label.
ylab	String for y axis label.
scale_x_num	Logical, if TRUE print sequential numbers starting from 1 as x axis labels, if FALSE use variable names.
scale_x_num_start	Numeric, specifying the starting value of the x axis, if scale_x_num = TRUE.
apaish	Logical, if TRUE aligns plot with APA guidelines.
...	Further arguments to be passed on to geom_label_repel .

Value

ggplot2 object

Examples

```
# Plot individual trajectories of IDs 2, 4, 5, and 9
plot_sg_trajectories(data = sgdata,
  id_var = "id",
  select_id_list = c("2", "4", "5", "9"),
  var_list = c("bdi_s1", "bdi_s2", "bdi_s3", "bdi_s4",
    "bdi_s5", "bdi_s6", "bdi_s7", "bdi_s8",
    "bdi_s9", "bdi_s10", "bdi_s11", "bdi_s12"),
  show_id = TRUE,
  id_label_size = 4,
  label.padding = .2,
  show_legend = FALSE,
  colour = "viridis",
  viridis_option = "D",
  viridis_begin = 0,
  viridis_end = .8,
  connect_missing = FALSE,
```

```

        scale_x_num = TRUE,
        scale_x_num_start = 1,
        apaish = TRUE,
        xlab = "Session",
        ylab = "BDI")

# Create byperson dataset to use for plotting
byperson <- create_byperson(data = sgdata,
  sg_crit1_cutoff = 7,
  id_var_name = "id",
  tx_start_var_name = "bdi_s1",
  tx_end_var_name = "bdi_s12",
  sg_var_list = c("bdi_s1", "bdi_s2", "bdi_s3",
    "bdi_s4", "bdi_s5", "bdi_s6",
    "bdi_s7", "bdi_s8", "bdi_s9",
    "bdi_s10", "bdi_s11", "bdi_s12"),
  sg_measure_name = "bdi")

# First, filter byperson dataset to only include cases with more than one sudden gain
# Next, plot BDI trajectory of 3 randomly selected cases with with more than one sudden gain
byperson %>%
  dplyr::filter(sg_freq_byperson > 1) %>%
  plot_sg_trajectories(id_var = "id_sg",
    var_list = c("bdi_s1", "bdi_s2", "bdi_s3", "bdi_s4",
      "bdi_s5", "bdi_s6", "bdi_s7", "bdi_s8",
      "bdi_s9", "bdi_s10", "bdi_s11", "bdi_s12"),
    select_n = 3,
    show_id = TRUE,
    show_legend = TRUE,
    scale_x_num = TRUE,
    scale_x_num_start = 1,
    xlab = "Session",
    ylab = "BDI")

```

 rename_sg_vars

Rename variable names to a generic and consistent format

Description

Rename variable names to a generic and consistent format that can be used by other functions of the suddengains package.

Usage

```

rename_sg_vars(
  data,
  rename_var_list,
  new_var_str = "temp_var_",
  start_numbering = 1
)

```


Arguments

data	Dataframe in wide format.
rename_var_list	Vector of variables to be renamed.
new_var_str	String, new name for variables.
start_numbering	Numeric, first number to be used as suffix for renaming variables specified in "rename_var_list".

Value

Dataframe in wide format with renamed variables.

select_cases	<i>Select sample providing enough data to identify sudden gains</i>
--------------	---

Description

Select sample for further sudden gains analyses depending on specified methods.

The following table shows the different data patterns that get selected when method = "pattern". This function goes through the data and selects all cases with at least one of the following data patterns.

Pattern	x1	x2	x3	x4	x5	x6
1.	x	N	x	x	.	.
2.	x	N	x	.	x	.
3.	x	.	N	x	x	.
4.	x	.	N	x	.	x

Note. x1 to x6 are consecutive data points of the primary outcome measure. 'x' = Available data; '.' = Missing data. 'N' represents available data to be examined as a possible pregain session.

Usage

```
select_cases(
  data,
  id_var_name,
  sg_var_list,
  method = c("pattern", "min_sess"),
  min_sess_num = NULL,
  return_id_lgl = FALSE
)
```

Arguments

data	A dataset in wide format with an id variable and the sudden gains variables.
------	--

<code>id_var_name</code>	String, specifying the name of the ID variable. Each row should have a unique value.
<code>sg_var_list</code>	Vector, specifying the variable names of each measurement point sequentially.
<code>method</code>	String, specifying the method used to select cases: <code>pattern</code> or <code>min_sess</code> .
<code>min_sess_num</code>	Numeric, minimum number of available sessions to be selected. This argument needs to be specified if <code>method = min_sess</code> .
<code>return_id_lgl</code>	Logical, if TRUE the function returns the ID variable and a new variable <code>sg_select</code> indicating whether there is enough data available to identify sudden gains. If set to FALSE this function returns the input data together with the new variable <code>sg_select</code> .

Value

A wide dataset indicating with all cases and a variable indicating whether each cases provides enough data to identify sudden gains.

Examples

```
# 1. method = "pattern"
select_cases(data = sgdata,
             id_var_name = "id",
             sg_var_list = c("bdi_s1", "bdi_s2", "bdi_s3", "bdi_s4",
                             "bdi_s5", "bdi_s6", "bdi_s7", "bdi_s8",
                             "bdi_s9", "bdi_s10", "bdi_s11", "bdi_s12"),
             method = "pattern",
             return_id_lgl = FALSE)

# 2. method = "min_sess"
select_cases(data = sgdata,
             id_var_name = "id",
             sg_var_list = c("bdi_s1", "bdi_s2", "bdi_s3", "bdi_s4",
                             "bdi_s5", "bdi_s6", "bdi_s7", "bdi_s8",
                             "bdi_s9", "bdi_s10", "bdi_s11", "bdi_s12"),
             method = "min_sess",
             min_sess_num = 9,
             return_id_lgl = TRUE)
```

sgdata

Longitudinal dataset with repeated measures of depression and rumination

Description

Example dataset with a measure of depression symptoms (BDI) and a secondary process measure (RQ; Rumination Questionnaire) to illustrate how the package works.

Usage

```
data(sgdata)
```

Format

A longitudinal dataset in wide format, i.e one row per person, one column per variable.

- id: ID variable, unique identifier for each person
- bdi_s0: BDI value, baseline assessment
- bdi_s1: BDI value, session 1
- bdi_s2: BDI value, session 2
- bdi_s3: BDI value, session 3
- bdi_s4: BDI value, session 4
- bdi_s5: BDI value, session 5
- bdi_s6: BDI value, session 6
- bdi_s7: BDI value, session 7
- bdi_s8: BDI value, session 8
- bdi_s9: BDI value, session 9
- bdi_s10: BDI value, session 10
- bdi_s11: BDI value, session 11
- bdi_s12: BDI value, session 12
- bdi_fu1: BDI value, follow-up measure 1
- bdi_fu2: BDI value, follow-up measure 2
- rq_s0: RQ value, baseline assessment
- rq_s1: RQ value, session 1
- rq_s2: RQ value, session 2
- rq_s3: RQ value, session 3
- rq_s4: RQ value, session 4
- rq_s5: RQ value, session 5
- rq_s6: RQ value, session 6
- rq_s7: RQ value, session 7
- rq_s8: RQ value, session 8
- rq_s9: RQ value, session 9
- rq_s10: RQ value, session 10
- rq_s11: RQ value, session 11
- rq_s12: RQ value, session 12
- rq_fu1: RQ value, follow-up measure 1
- rq_fu2: RQ value, follow-up measure 2

Examples

```
# Load data into global environment  
data(sgdata)
```

sgdata_bad	<i>Longitudinal dataset with repeated measures of depression and rumination (bad variable names)</i>
------------	--

Description

Same as sgdata but with bad variable names to illustrate that the package can also work with inconsistent names.

Usage

```
data(sgdata_bad)
```

Format

A longitudinal dataset in wide format, i.e one row per person, one column per variable.

- id: ID variable, unique identifier for each person
- bdi_intake: BDI value, baseline assessment
- bdi1: BDI value, session 1
- wk2bdi: BDI value, session 2
- bdi.S3: BDI value, session 3
- bdis4: BDI value, session 4
- bdi_session5: BDI value, session 5
- bdi_weekSix: BDI value, session 6
- beck.dep.inv_sess7: BDI value, session 7
- weeeek8bdi: BDI value, session 8
- bdi_nine: BDI value, session 9
- bdii10: BDI value, session 10
- bDi11: BDI value, session 11
- bdi_s12_end: BDI value, session 12
- bdi_fu1: BDI value, follow-up measure 1
- bdi_followup2: BDI value, follow-up measure 2
- rq_intake: RQ value, baseline assessment
- rqi1: RQ value, session 1
- wk2rq: RQ value, session 2
- rq.S3: RQ value, session 3
- rq_s4: RQ value, session 4
- rq_session5: RQ value, session 5
- rq_weekSix: RQ value, session 6

- rq_sess7: RQ value, session 7
- weeeek8rq: RQ value, session 8
- rqnine: RQ value, session 9
- rq10: RQ value, session 10
- rqi11: RQ value, session 11
- rq_s12_end: RQ value, session 12
- prq_fu1: RQ value, follow-up measure 1
- rqq_followup2: RQ value, follow-up measure 2

Examples

```
# Load data into global environment
data(sgdata_bad)
```

write_byperson	<i>Write a sudden gains data frame (bysg) to CSV, SPSS, STATA or Excel files</i>
----------------	--

Description

Writes a data frame as a specified file type.

Usage

```
write_byperson(
  data,
  sg_crit1_cutoff,
  id_var_name,
  sg_var_list,
  tx_start_var_name,
  tx_end_var_name,
  sg_measure_name,
  sg_crit2_pct = 0.25,
  sg_crit3 = TRUE,
  sg_crit3_alpha = 0.05,
  sg_crit3_adjust = TRUE,
  sg_crit3_critical_value = 2.776,
  identify = c("sg", "s1"),
  identify_sg_1to2 = FALSE,
  multiple_sg_select = c("first", "last", "smallest", "largest"),
  data_is_bysg = FALSE,
  format = c("CSV", "SPSS", "STATA", "Excel"),
  path,
  stata_version = 14,
  ...
)
```

Arguments

<code>data</code>	A data set in wide format including an ID variable and variables for each measurement point.
<code>sg_crit1_cutoff</code>	Numeric, specifying the cut-off value to be used for the first sudden gains criterion. The function <code>define_crit1_cutoff</code> can be used to calculate a cutoff value based on the Reliable Change Index (RCI; Jacobson & Truax, 1991). If set to NULL the first criterion wont be applied.
<code>id_var_name</code>	String, specifying the name of the ID variable. Each row should have a unique value.
<code>sg_var_list</code>	Vector, specifying the variable names of each measurement point sequentially.
<code>tx_start_var_name</code>	String, specifying the variable name of the first measurement point of the intervention.
<code>tx_end_var_name</code>	String, specifying the variable name of the last measurement point of the intervention.
<code>sg_measure_name</code>	String, specifying the name of the measure used to identify sudden gains/losses.
<code>sg_crit2_pct</code>	Numeric, specifying the percentage change to be used for the second sudden gains/losses criterion. If set to NULL the second criterion wont be applied.
<code>sg_crit3</code>	If set to TRUE the third criterion will be applied automatically adjusting the critical value for missingness. If set to FALSE the third criterion wont be applied.
<code>sg_crit3_alpha</code>	Numeric, alpha for the student t-test (two-tailed) to determine the critical value to be used for the third criterion. Degrees of freedom are based on the number of available data in the three sessions preceding the gain and the three sessions following the gain.
<code>sg_crit3_adjust</code>	Logical, specify whether critical value gets adjusted for missingness, see Lutz et al. (2013) and the documentation of this R package for further details. This argument is set to TRUE by default adjusting the critical value for missingness as described in the package documentation and Lutz et al. (2013): A critical value of 2.776 is used when all three data points before and after a potential gain are available, where one datapoint is missing either before or after a potential gain a critical value of 3.182 is used, and where one datapoint is missing both before and after the gain a critical value of 4.303 is used (for <code>sg_crit3_alpha = 0.05</code>). If set to FALSE the critical value set in <code>sg_crit3_critical_value</code> will instead be used for all comparisons, regardless of missingnes in the sequence of data points that are investigated for potential sudden gains.
<code>sg_crit3_critical_value</code>	Numeric, specifying the critical value to instead be used for all comparisons, regardless of missingnes in the sequence of data points that are investigated for potential sudden gains.#'
<code>identify</code>	String, specifying whether to identify sudden gains ("sg") using <code>identify_sg</code> or sudden losses ("sl") using <code>identify_sl</code> . The default is to identify sudden gains ("sg").

identify_sg_1to2	Logical, indicating whether to identify sudden losses from measurement point 1 to 2.
multiple_sg_select	String, specifying which sudden gain/loss to select for this data set if more than one gain/loss was identified per case. Options are: "first", "last", "smallest", or "largest", create_byperson .
data_is_bysg	Logical, specifying whether the data set in the data argument is a bysg data set created using the create_bysg function.
format	String, specifying the format of the data file, "CSV", "SPSS", "STATA" or "Excel".
path	String, specifying the file name ending with the matching file extension, ".csv", ".sav", ".dta" or ".xlsx".
stata_version	Numeric, specifying STATA version number.
...	Additional parameters to be passed on to the specified write function, see write_csv for "CSV", write_sav for "SPSS", write_dta for "STATA" or write_xlsx for "Excel" for more information.

References

Tang, T. Z., & DeRubeis, R. J. (1999). Sudden gains and critical sessions in cognitive-behavioral therapy for depression. *Journal of Consulting and Clinical Psychology*, 67(6), 894–904. <https://doi.org/10.1037/0022-006X.67.6.894>.

Examples

```
# Adjust "path" argument before running
# Create character string name for temporary "byperson.csv" file
temp <- tempfile(pattern = "byperson", fileext = ".csv")

# Write byperson dataset (CSV file)
# To write a different format change the 'format' argument ...
# ... as well as the file extension in the 'path' argument
write_byperson(data = sgdata,
               sg_crit1_cutoff = 7,
               id_var_name = "id",
               tx_start_var_name = "bdi_s1",
               tx_end_var_name = "bdi_s12",
               sg_var_list = c("bdi_s1", "bdi_s2", "bdi_s3", "bdi_s4",
                              "bdi_s5", "bdi_s6", "bdi_s7", "bdi_s8",
                              "bdi_s9", "bdi_s10", "bdi_s11", "bdi_s12"),
               sg_measure_name = "bdi",
               identify_sg_1to2 = FALSE,
               multiple_sg_select = "largest",
               format = "CSV",
               path = temp)
```

write_bysg	<i>Write a sudden gains data frame (bysg) to CSV, SPSS, STATA or Excel files</i>
------------	--

Description

Writes a data frame as a specified file type.

Usage

```
write_bysg(
  data,
  sg_crit1_cutoff,
  id_var_name,
  sg_var_list,
  tx_start_var_name,
  tx_end_var_name,
  sg_measure_name,
  sg_crit2_pct = 0.25,
  sg_crit3 = TRUE,
  sg_crit3_alpha = 0.05,
  sg_crit3_adjust = TRUE,
  sg_crit3_critical_value = 2.776,
  identify = c("sg", "sl"),
  identify_sg_1to2 = FALSE,
  format = c("CSV", "SPSS", "STATA", "Excel"),
  path,
  stata_version = 14,
  ...
)
```

Arguments

data	A data set in wide format including an ID variable and variables for each measurement point.
sg_crit1_cutoff	Numeric, specifying the cut-off value to be used for the first sudden gains criterion. The function <code>define_crit1_cutoff</code> can be used to calculate a cutoff value based on the Reliable Change Index (RCI; Jacobson & Truax, 1991). If set to NULL the first criterion wont be applied.
id_var_name	String, specifying the name of the ID variable. Each row should have a unique value.
sg_var_list	Vector, specifying the variable names of each measurement point sequentially.
tx_start_var_name	String, specifying the variable name of the first measurement point of the intervention.

tx_end_var_name	String, specifying the variable name of the last measurement point of the intervention.
sg_measure_name	String, specifying the name of the measure used to identify sudden gains/losses.
sg_crit2_pct	Numeric, specifying the percentage change to be used for the second sudden gains/losses criterion. If set to NULL the second criterion wont be applied.
sg_crit3	If set to TRUE the third criterion will be applied automatically adjusting the critical value for missingness. If set to FALSE the third criterion wont be applied.
sg_crit3_alpha	Numeric, alpha for the student t-test (two-tailed) to determine the critical value to be used for the third criterion. Degrees of freedom are based on the number of available data in the three sessions preceding the gain and the three sessions following the gain.
sg_crit3_adjust	Logical, specify whether critical value gets adjusted for missingness, see Lutz et al. (2013) and the documentation of this R package for further details. This argument is set to TRUE by default adjusting the critical value for missingness as described in the package documentation and Lutz et al. (2013): A critical value of 2.776 is used when all three data points before and after a potential gain are available, where one datapoint is missing either before or after a potential gain a critical value of 3.182 is used, and where one datapoint is missing both before and after the gain a critical value of 4.303 is used (for sg_crit3_alpha = 0.05). If set to FALSE the critical value set in sg_crit3_critical_value will instead be used for all comparisons, regardless of missingnes in the sequence of data points that are investigated for potential sudden gains.
sg_crit3_critical_value	Numeric, specifying the critical value to instead be used for all comparisons, regardless of missingnes in the sequence of data points that are investigated for potential sudden gains.
identify	String, specifying whether to identify sudden gains ("sg") using <code>identify_sg</code> or sudden losses ("sl") using <code>identify_sl</code> . The default is to identify sudden gains ("sg").
identify_sg_1to2	Logical, indicating whether to identify sudden losses from measurement point 1 to 2.
format	String, specifying the format of the data file, "CSV", "SPSS", "STATA" or "Excel".
path	String, specifying the file name ending with the matching file extension, ".csv", ".sav", ".dta" or ".xlsx".
stata_version	Numeric, specifying STATA version number.
...	Additional parameters to be passed on to the specified write function, see <code>write_csv</code> for "CSV", <code>write_sav</code> for "SPSS", <code>write_dta</code> for "STATA" or <code>write_xlsx</code> for "Excel" for more information.

References

Tang, T. Z., & DeRubeis, R. J. (1999). Sudden gains and critical sessions in cognitive-behavioral therapy for depression. *Journal of Consulting and Clinical Psychology*, 67(6), 894–904. <https://doi.org/10.1037/0022-006X.67.6.894>.

Examples

```
# Adjust "path" argument before running
# Create character string name for temporary "bysg.csv" file
temp <- tempfile(pattern = "bysg", fileext = ".csv")

# Write bysg dataset (CSV file)
# To write a different format change the 'format' argument
# as well as the file extension in the 'path' argument
write_bysg(data = sgdata,
            sg_crit1_cutoff = 7,
            id_var_name = "id",
            tx_start_var_name = "bdi_s1",
            tx_end_var_name = "bdi_s12",
            sg_var_list = c("bdi_s1", "bdi_s2", "bdi_s3", "bdi_s4",
                           "bdi_s5", "bdi_s6", "bdi_s7", "bdi_s8",
                           "bdi_s9", "bdi_s10", "bdi_s11", "bdi_s12"),
            sg_measure_name = "bdi",
            identify_sg_1to2 = FALSE,
            format = "CSV",
            path = temp)
```

Index

*Topic **dataset**

sgdata, [26](#)

sgdata_bad, [28](#)

check_interval, [2](#)

count_intervals, [4](#)

create_byperson, [5](#), [12](#), [19](#), [31](#)

create_bysg, [6](#), [8](#), [12](#), [19](#), [31](#)

define_crit1_cutoff, [3](#), [6](#), [8](#), [10](#), [15](#), [17](#), [30](#),
[32](#)

describe_sg, [12](#)

extract_values, [13](#)

geom_label_repel, [23](#)

ggplot, [19](#), [22](#)

identify_sg, [7](#), [9](#), [14](#), [30](#), [33](#)

identify_sl, [7](#), [9](#), [16](#), [30](#), [33](#)

plot_sg, [19](#)

plot_sg_intervals, [21](#)

plot_sg_trajectories, [22](#)

rename_sg_vars, [24](#)

scale_fill_viridis_d, [20](#), [23](#)

select_cases, [25](#)

sgdata, [26](#)

sgdata_bad, [28](#)

write_byperson, [29](#)

write_bysg, [32](#)

write_csv, [31](#), [33](#)

write_dta, [31](#), [33](#)

write_sav, [31](#), [33](#)

write_xlsx, [31](#), [33](#)