Package 'sdcLog'

February 16, 2021

Title Tools for Statistical Disclosure Control in Research Data Centers

Version 0.2.0

Description Tools for researchers to explicitly show that their results comply to rules for statistical disclosure control imposed by research data centers. These tools help in checking descriptive statistics and models and in calculating extreme values that are not individual data. Also included is a simple function to create log files. The methods used here are described in the ``Guidelines for the checking of output based on microdata research'' by Bond, Brandt, and de Wolf (2015) <https://ec.europa.eu/eurostat/cros/system/files/dwb_standalone-document_output-checking-guidelines.pdf>.

License GPL-3

URL https://github.com/matthiasgomolka/sdcLog

BugReports https://github.com/matthiasgomolka/sdcLog/issues

Depends R (>= 3.5)

Imports broom (>= 0.5.5), checkmate (>= 2.0.0), crayon (>= 1.3.4), data.table (>= 1.12.8), methods

Suggests covr (>= 3.5.0), devtools, here, knitr, rmarkdown, skimr, spelling, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

Language en-US

LazyData true

RoxygenNote 7.1.1

NeedsCompilation no

Author Matthias Gomolka [aut, cre], Tim Becker [aut]

Maintainer Matthias Gomolka <matthias.gomolka@posteo.de>

Repository CRAN

Date/Publication 2021-02-16 21:10:02 UTC

R topics documented:

`check_distinct_ids` *Internal function which creates cross-tables with number of distinct id's*

Description

Internal function which creates cross-tables with number of distinct id's

Usage

```
check_distinct_ids(data, id_var, val_var = NULL, by = NULL)
```

Arguments

`data`	data.frame from which the descriptive statistics are calculated.
`id_var`	character The name of the id variable. Defaults to `getOption("sdc.id_var")` so that you can provide `options(sdc.id_var = "my_id_var")` at the top of your script.
`val_var`	character vector of value variables on which descriptive statistics are computed.
`by`	character vector of grouping variables.

check_dominance *Internal function which creates cross-tables with number of distinct id's*

Description

Internal function which creates cross-tables with number of distinct id's

Usage

```
check_dominance(data, id_var, val_var = NULL, by = NULL)
```

Arguments

data	data.frame from which the descriptive statistics are calculated.
id_var	character The name of the id variable. Defaults to `getOption("sdc.id_var")` so that you can provide `options(sdc.id_var = "my_id_var")` at the top of your script.
val_var	character vector of value variables on which descriptive statistics are computed.
by	character vector of grouping variables.

common_arguments *arguments*

Description

arguments

Arguments

data	data.frame from which the descriptive statistics are calculated.
id_var	character The name of the id variable. Defaults to `getOption("sdc.id_var")` so that you can provide `options(sdc.id_var = "my_id_var")` at the top of your script.
val_var	character vector of value variables on which descriptive statistics are computed.
by	character vector of grouping variables.
zero_as_NA	logical If TRUE, zeros in 'val_var' are treated as NA.
model	The estimated model object. Can be a model type like lm, glm and various others (anything which can be handled by `broom::augment()`).
min_obs	integer The minimum number of observations used to calculate the minimum and maximum. Defaults to `getOption("sdc.n_ids",5L)`. *This is not the number of distinct entities.*
max_obs	integer The maximum number of observations used to calculate the minimum and maximum. Defaults to `nrow(data)`. *This is not the number of distinct entities.*

`sdc_descriptives` *Disclosure control for descriptive statistics*

Description

Checks if your descriptive statistics comply to statistical disclosure control. Checks for number of distinct entities and dominance.

Usage

```
sdc_descriptives(
  data,
  id_var = getOption("sdc.id_var"),
  val_var = NULL,
  by = NULL,
  zero_as_NA = NULL
)
```

Arguments

`data`	data.frame from which the descriptive statistics are calculated.
`id_var`	character The name of the id variable. Defaults to `getOption("sdc.id_var")` so that you can provide `options(sdc.id_var = "my_id_var")` at the top of your script.
`val_var`	character vector of value variables on which descriptive statistics are computed.
`by`	character vector of grouping variables.
`zero_as_NA`	logical If TRUE, zeros in 'val_var' are treated as NA.

Value

A list of class `sdc_descriptives` with detailed information about options, settings, and compliance with the criteria distinct entities and dominance.

Examples

```
sdc_descriptives(
  data = sdc_descriptives_DT,
  id_var = "id",
  val_var = "val_1"
)

sdc_descriptives(
  data = sdc_descriptives_DT,
  id_var = "id",
  val_var = "val_1",
  by = "sector"
)
```

```
sdc_descriptives(
  data = sdc_descriptives_DT,
  id_var = "id",
  val_var = "val_1",
  by = c("sector", "year")
)

sdc_descriptives(
  data = sdc_descriptives_DT,
  id_var = "id",
  val_var = "val_2",
  by = c("sector", "year")
)

sdc_descriptives(
  data = sdc_descriptives_DT,
  id_var = "id",
  val_var = "val_2",
  by = c("sector", "year"),
  zero_as_NA = FALSE
)
```

`sdc_descriptives_DT` *Example data for* `sdc_descriptives()`

Description

Utilized in the vignette.

Usage

```
data("sdc_descriptives_DT")
```

Format

A data.table with 20 rows and 5 columns.

Details

The data.table contains the following columns:

- id factor random identifier
- sector factor economic sector
- year integer time variable
- val_1, val_2 numeric value variables

sdc_log *Create Stata-like log files from R Scripts*

Description

This function creates Stata-like log files from R Scripts. It can handle several files (in a character vector) at once.

Usage

```
sdc_log(r_script, destination, replace = FALSE, append = FALSE)
```

Arguments

r_script	character Path of the R script to be run with logging.
destination	One of: • character Path of the log file to be used. • file connection to which the log should be written. This is especially useful, when you have nested calls to sdc_log() and want to write everything into the same log file. Then, create a single file connection and provide this connection to all calls to sdc_log() (and close it afterwards).
replace	logical Indicates whether to replace an existing log file.
append	logical Indicates whether to append an existing log file.

Value

character vector holding the path(s) of the written log file(s).

sdc_min_max *Calculate RDC rule-compliant extreme values*

Description

Checks if calculation of extreme values comply to RDC rules. If so, function returns average min and max values according to RDC rules.

Usage

```
sdc_min_max(
  data,
  id_var = getOption("sdc.id_var"),
  val_var,
  by = NULL,
  max_obs = nrow(data)
)
```

Arguments

data	data.frame from which the descriptive statistics are calculated.
id_var	character The name of the id variable. Defaults to `getOption("sdc.id_var")` so that you can provide `options(sdc.id_var = "my_id_var")` at the top of your script.
val_var	character vector of value variables on which descriptive statistics are computed.
by	character vector of grouping variables.
max_obs	integer The maximum number of observations used to calculate the minimum and maximum. Defaults to `nrow(data)`. *This is not the number of distinct entities.*

Value

A list list of class `sdc_min_max` with detailed information about options, settings and the calculated extreme values (if possible).

Examples

```
sdc_min_max(sdc_min_max_DT, id_var = "id", val_var = "val_1")
sdc_min_max(sdc_min_max_DT, id_var = "id", val_var = "val_2")
sdc_min_max(sdc_min_max_DT, id_var = "id", val_var = "val_3", max_obs = 10)
sdc_min_max(sdc_min_max_DT, id_var = "id", val_var = "val_1", by = "year")
sdc_min_max(
  sdc_min_max_DT, id_var = "id", val_var = "val_1", by = c("sector", "year")
)
```

---

sdc_min_max_DT *Example data for* `sdc_min_max()`

Description

Utilized in the vignette

Usage

```
data("sdc_min_max_DT")
```

Format

A data.table with 20 rows and 6 columns.

Details

The data.table contains the following columns:

- id factor random identifier
- sector factor economic sector
- year integer time variable
- val_1 - val_3 numeric value variables

`sdc_model` *Disclosure control for models*

Description

Checks if your model complies to RDC rules. Checks for overall number of entities and number of entities for each level of dummy variables.

Usage

```
sdc_model(data, model, id_var = getOption("sdc.id_var"))
```

Arguments

`data`	data.frame which was used to build the model.
`model`	The estimated model object. Can be a model type like lm, glm and various others (anything which can be handled by `broom::augment()`).
`id_var`	character The name of the id variable. Defaults to `getOption("sdc.id_var")` so that you can provide `options(sdc.id_var = "my_id_var")` at the top of your script.

Value

A list of class `sdc_model` with detailed information about options, settings, and compliance with the distinct entities criterion.

Examples

```
# Check simple models
model_1 <- lm(y ~ x_1 + x_2, data = sdc_model_DT)
sdc_model(data = sdc_model_DT, model = model_1, id_var = "id")

model_2 <- lm(y ~ x_1 + x_2 + x_3, data = sdc_model_DT)
sdc_model(data = sdc_model_DT, model = model_2, id_var = "id")

model_3 <- lm(y ~ x_1 + x_2 + dummy_3, data = sdc_model_DT)
sdc_model(data = sdc_model_DT, model = model_3, id_var = "id")
```

sdc_model_DT *Example data for* sdc_model()

Description

Utilized in the vignette

Usage

```
data("sdc_model_DT")
```

Format

A data.table with 80 rows and 9 columns.

Details

The data.table contains the following columns:

- id factor random identifier
- y - x_4 numeric value variables
- dummy_1 - dummy_3 factor dummy variables

warn_distinct_ids *Throw only a single warning about insufficient distinct ID's*

Description

Checks if any check on distinct ID's was problematic and throws a single warning.

Usage

```
warn_distinct_ids(list)
```

Arguments

list	list of elements of class sdc_distinct_ids.

Index