Package: verdata (via r-universe)

October 31, 2026

Title Analyze Data from the Truth Commission in Colombia

Version 1.0.2

Maintainer Maria Gargiulo <mariag@hrdag.org>

Description Facilitates use and analysis of data about the armed conflict in Colombia resulting from the joint project between La Jurisdicción Especial para la Paz (JEP), La Comisión para el Esclarecimiento de la Verdad, la Convivencia y la No repetición (CEV), and the Human Rights Data Analysis Group (HRDAG). The data are 100 replicates from a multiple imputation through chained equations as described in Van Buuren and Groothuis-Oudshoorn (2011) <doi:10.18637/jss.v045.i03>. With the replicates the user can examine four human rights violations that occurred in the Colombian conflict accounting for the impact of missing fields and fully missing observations.

License GPL-2

URL https://github.com/HRDAG/verdata

BugReports https://github.com/HRDAG/verdata/issues

Depends R (>= 3.5)

Imports arrow, assertr, base, digest, dplyr, glue, LCMCR, logger, magrittr, purrr, Rdpack, readr, rjson, rlang, stats, stringr, tibble, tidyr, tidyselect, tools

Suggests covr, spelling, testthat (>= 3.0.0)

RdMacros Rdpack

Config/testthat/edition 3

Encoding UTF-8

Language en-US

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Config/pak/sysreqs cmake libgsl0-dev libicu-dev libxml2-dev libssl-dev libx11-dev

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

Date/Publication 2026-02-03 17:04:57 UTC

RemoteUrl https://github.com/hrdag/verdata

RemoteRef HEAD

RemoteSha 6d62ea38062ec4f7c0ecc2260b6e74904165044a

Contents

combine_estimates . . . 2
combine_replicates . . . 4
confirm_files . . . 5
diccionario_replicas . . . 6
diccionario_vars_adicional . . . 7
estimates_exist . . . 7
estratificacion . . . 8
filter_standard_cev . . . 9
get_valid_sources . . . 9
lookup_estimates . . . 10
mse . . . 11
proportions_imputed . . . 13
proportions_observed . . . 14
read_replicates . . . 14
run_lcmcr . . . 15
stratification . . . 17
summary_observed . . . 18

Index **20**

`combine_estimates` *Combine MSE estimation results for a given stratum calculated using multiple replicate files created using multiple imputation. Combination is done using the standard approach that makes use of the laws of total expectation and total variance.*

Description

Combine MSE estimation results for a given stratum calculated using multiple replicate files created using multiple imputation. Combination is done using the standard approach that makes use of the laws of total expectation and total variance.

Usage

```
combine_estimates(stratum_estimates)
```

Arguments

`stratum_estimates`
: A data frame of estimates for a stratum of interest calculated using `mse` for all replicates being used for the analysis. The data frame should have columns `N` and `n_obs` from the `mse` function and an additional column `replicate` indicating which replicate the estimates were calculated on.

Value

A data frame row with the point estimate (`N_mean`) and the associated 95% uncertainty interval (lower bound is `N_025`, upper bound is `N_975`).

References

Gelman A, Carlin JB, Stern HS, Dunson DB, Vehtari A, Rubin DB (2013). *Bayesian Data Analysis*, 0 edition. Chapman and Hall/CRC. ISBN 978-0-429-11307-9. doi:10.1201/b16018.

Examples

```
set.seed(19481210)

library(dplyr)
library(purrr)
library(glue)


simulate_estimates <- function(stratum_data, replicate_num) {

    # simulate an imputed stratification variable to determine whether a record
    # should be considered part of the stratum for estimation
    stratification_var <- sample(c(0, 1), size = 100,
                                 replace = TRUE, prob = c(0.1, 0.9))

    my_stratum <- bind_cols(my_stratum, tibble::tibble(stratification_var)) %>%
        filter(stratification_var == 1)

    results <- mse(my_stratum, "my_stratum", K = 4) %>%
        mutate(replicate = replicate_num)

    return(results)

}


in_A <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.45, 0.65))
in_B <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.5, 0.5))
in_C <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.75, 0.25))

my_stratum <- tibble::tibble(in_A, in_B, in_C)

replicate_nums <- glue("R{1:20}")
```

```
estimates <- map_dfr(.x = replicate_nums,
             .f = ~simulate_estimates(stratum_data = my_stratum, replicate_num = .x))

combine_estimates(estimates)
```

`combine_replicates` *Combine imputed replicates according to calculate totals. Combination is done using the standard approach that makes use of the laws of total expectation and total variance.*

Description

Combine imputed replicates according to calculate totals. Combination is done using the standard approach that makes use of the laws of total expectation and total variance.

Usage

```
combine_replicates(
  violation,
  replicates_obs_data,
  replicates_data,
  strata_vars = NULL,
  conflict_filter = TRUE,
  forced_dis_filter = FALSE,
  edad_minors_filter = FALSE,
  include_props = FALSE,
  digits = 2
)
```

Arguments

`violation`	Violation to be analyzed. Options are "homicidio", "secuestro", "reclutamiento" and "desaparicion".
`replicates_obs_data`	The data frame that results from applying `summary_observed`.
`replicates_data`	A data frame containing replicates data.
`strata_vars`	Variable with all observations (without missing values).
`conflict_filter`	Filter that indicates if the data is filtered using the "is_conflict" rule.
`forced_dis_filter`	Filter that indicates if the data is filtered using the "is_forced_dis" rule.
`edad_minors_filter`	Optional filter by age (`edad`) < 18.

include_props A logical value indicating whether or not to include the proportions from the calculations before merging with summary_observed's output.

digits Number of decimal places to round the results to. Default value is 2.

Value

A data frame with 5 or more columns: name of variable(s), observed the number of observations in each category for every variable, imp_lo the lower bound of the 95% confidence interval, imp_hi the upper bound of the 95% confidence interval, and imp_mean the point estimate of the mean value.

Examples

```
local_dir <- system.file("extdata", "right", package = "verdata")
replicates_data <- read_replicates(local_dir, "reclutamiento", c(1, 2),
version = "v1")
replicates_obs_data <- summary_observed("reclutamiento", replicates_data,
strata_vars = "sexo", conflict_filter = FALSE, forced_dis_filter = FALSE,
edad_minors_filter = FALSE, include_props = FALSE, digits = 2)
tab_combine <- combine_replicates("reclutamiento", replicates_obs_data,
replicates_data, strata_vars = 'sexo', conflict_filter = TRUE,
forced_dis_filter = FALSE, edad_minors_filter = FALSE, include_props = FALSE,
digits = 2)
```

---

confirm_files *Confirm files are identical to the ones published.*

Description

Confirm files are identical to the ones published.

Usage

```
confirm_files(replicates_dir, violation, replicate_nums, version)
```

Arguments

replicates_dir Directory containing the replicates. The name of the files must include the violation in Spanish and lower case letters (homicidio, secuestro, reclutamiento, desaparicion).

violation Violation being analyzed. Options are "homicidio", "secuestro", "reclutamiento", and "desaparicion".

replicate_nums A numeric vector containing the replicates to be analyzed. Values in the vector should be between 1 and 100 inclusive.

version Version of the data being read in. Options are "v1" or "v2". "v1" is appropriate for replicating the replicating the results of the joint JEP-CEV-HRDAG project. "v2" is appropriate for conducting your new analyses of the conflict in Colombia.

Value

A data frame row with `replicate_num` rows and two columns: `replicate_path`, a string indicating the path to the replicate checked and `confirmed`, a boolean values indicating whether the replicate contents match the published version.

Examples

```
local_dir <- system.file("extdata", "right", package = "verdata")
confirm_files(local_dir, "reclutamiento", c(1, 2), version = "v1")
```

---

diccionario_replicas *Diccionario de datos para las variables que aparecen en los archivos de las réplicas.*

Description

Diccionario de datos para las variables que aparecen en los archivos de las réplicas.

Usage

```
data(diccionario_replicas)
```

Format

Un data frame con 55 filas y 4 variables.

nombre_variable nombre de la variable

tipo tipo de la variable: caracter, numérico, lógico

detalle_variable explicación detallada de la variable

categorias_variable valores posibles de la variable

Source

Proyecto conjunto JEP-CEV-HRDAG.

`diccionario_vars_adicional` *Variables adicionales que pueden ser útiles para analizar los datos.*

Description

Variables adicionales que pueden ser útiles para analizar los datos.

Usage

```
data(diccionario_vars_adicional)
```

Format

Un data frame con 11 filas y 4 variables.

nombre_variable nombre de la variable

tipo tipo de la variable: caracter, numérico, lógico

detalle_variable explicación detallada de la variable

categorias_variable valores posibles de la variable

Source

Proyecto conjunto JEP-CEV-HRDAG.

`estimates_exist` *Check whether stratum estimates already exist in pre-calculated files.*

Description

Check whether stratum estimates already exist in pre-calculated files.

Usage

```
estimates_exist(stratum_data_prepped, estimates_dir)
```

Arguments

`stratum_data_prepped`	A data frame including all records in a stratum of interest. The data frame should only include the source columns prefixed with `in_` and all columns should only contain 1's and 0's.
`estimates_dir`	Directory containing pre-calculated estimates, if you would like to use pre-calculated results.

Value

A list with two entries, `estimates_exist` and `estimates_path`. `estimates_exist` is a logical value indicating whether calculations for the stratum of interest are available in the directory containing the pre-calculated estimates. If `estimates_exist` is `TRUE`, `estimates_path` will contain the full file path to the JSON file containing the estimates, otherwise it will be `NA`.

Examples

```
in_A <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.45, 0.65))
in_B <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.5, 0.5))
in_C <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.75, 0.25))
in_D <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(1, 0))

my_stratum <- tibble::tibble(in_A, in_B, in_C, in_D) %>%
    dplyr::mutate(rs = rowSums(.)) %>%
    dplyr::filter(rs >= 1) %>%
    dplyr::select(-rs)

estimates_exist(stratum_data_prepped = my_stratum, estimates_dir = "path_to_estimates")
```

---

estratificacion

Datos que documentan las estratificaciones necesarias para replicar los resultados del informe metodológico del proyecto conjunto CEV-HRDAG-JEP (versión en español).

Description

Datos que documentan las estratificaciones necesarias para replicar los resultados del informe metodológico del proyecto conjunto CEV-HRDAG-JEP (versión en español).

Usage

```
data(estratificacion)
```

Format

Un data frame con 31 filas y 4 variables.

violacion el hecho de violencia al analizar

estimacion el tipo de análisis que utiliza la estratificación (p.ej., patrones de violencia por año, sexo, etc.)

estratificacion las variables utilizadas para estratificar las estimaciones

notas notas adicionales sobre la estratificación; NA si no hay notas

Source

Proyecto conjunto JEP-CEV-HRDAG.

filter_standard_cev *Filter records to replicate results presented in the CEV methodology report.*

Description

Filter records to replicate results presented in the CEV methodology report.

Usage

```
filter_standard_cev(replicates_data, violation, perp_change = TRUE)
```

Arguments

replicates_data	A data frame with data from all replicates to be filtered.
violation	Violation to be analyzed. Options are "homicidio", "secuestro", "reclutamiento", and "desaparicion".
perp_change	A logical value indicating whether victims in years after 2016 with perpetrator values (indicated by p_str) of the FARC-EP ("GUE-FARC") should be reassigned to other guerrilla groups (p_str value "GUE-OTRO").

Value

A filtered data frame.

Examples

```
local_dir <- system.file("extdata", "right", package = "verdata")
replicates_data <- read_replicates(local_dir, "reclutamiento", c(1, 2), version = "v1")
filter_standard_cev(replicates_data, "reclutamiento", perp_change = TRUE)
```

get_valid_sources *Determine valid sources for estimation of a stratum of interest.*

Description

Determine valid sources for estimation of a stratum of interest.

Usage

```
get_valid_sources(stratum_data_prepped, min_n = 1)
```

Arguments

`stratum_data_prepped`
: A data frame with all records in a stratum of interest. Columns indicating sources should be prefixed with `in_` and should be numeric with 1 indicating that an individual was documented in the source and 0 indicating that an individual was not documented in the source.

`min_n`
: The minimum number of records that must appear in a source to be considered valid for estimation. `min_n` should never be less than or equal to 0; the default value is 1.

Value

A character vector containing the names of the valid sources.

Examples

```
set.seed(19481210)
in_A <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.45, 0.65))
in_B <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.5, 0.5))
in_C <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.75, 0.25))
in_D <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(1, 0))

my_stratum <- tibble::tibble(in_A, in_B, in_C, in_D)
get_valid_sources(my_stratum)
```

`lookup_estimates` *lookup_estimates*

Description

Look up and read in existing estimates from pre-calculated files.

Usage

```
lookup_estimates(stratum_data_prepped, estimates_dir)
```

Arguments

`stratum_data_prepped`
: A data frame including all records in a stratum of interest. The data frame should only include the source columns prefixed with `in_` and all columns should only contain 1's and 0's.

`estimates_dir`
: Directory containing pre-calculated estimates, if you would like to use pre-calculated results. Note, setting this option forces the model specification parameters to be identical to those used to calculate the pre-calculated estimates. Do not specify a file path If you would like to use a custom model specification.

Value

A data frame with one column, N, indicating the results. If the stratum was not found in the pre-calculated files, N will be NA and the data frame will have one row. If the stratum was found in the pre-calculated files, N will contain draws from the posterior distribution of the model and the data frame will contain 1,000 rows.

Examples

```
in_A <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.45, 0.65))
in_B <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.5, 0.5))
in_C <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.75, 0.25))
in_D <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(1, 0))

my_stratum <- tibble::tibble(in_A, in_B, in_C, in_D) %>%
    dplyr::mutate(rs = rowSums(.)) %>%
    dplyr::filter(rs >= 1) %>%
    dplyr::select(-rs)

lookup_estimates(stratum_data_prepped = my_stratum, estimates_dir = "path_to_estimates")
```

---

mse *mse*

Description

Prepare data for estimation and calculate estimates using run_lcmcr.

Usage

```
mse(
  stratum_data,
  stratum_name,
  estimates_dir = NULL,
  min_n = 1,
  K = NULL,
  buffer_size = 10000,
  sampler_thinning = 1000,
  seed = 19481210,
  burnin = 10000,
  n_samples = 10000,
  posterior_thinning = 500
)
```

Arguments

stratum_data	A data frame including all records in a stratum of interest. Columns indicating sources should be prefixed with in_ and should be numeric.
stratum_name	An identifier for the stratum.
estimates_dir	File path for the folder containing pre-calculated estimates, if you would like to use pre-calculated results. Note, setting this option forces the model specification parameters to be identical to those used to calculate the pre-calculated estimates. Do not specify a file path If you would like to use a custom model specification.
min_n	The minimum number of records that must appear in a source to be considered valid for estimation. min_n should never be less than or equal to 0; the default value is 1.
K	The maximum number of latent classes to fit. By default the function will calculate K as the minimum value of 2 raised to the number of valid sources - 1 or 15.
buffer_size	Size of the tracing buffer. Default value is 10,000.
sampler_thinning	Thinning interval for the tracing buffer. Default value is 1,000.
seed	Integer seed for the internal random number generator. Default value is 19481210.
burnin	Number of burn in iterations. Default value is 10,000.
n_samples	Number of samples to be generated. Samples are taken one every posterior_thinning iterations of the sampler. Default value is 10,000. The final number of samples from the posterior is n_samples divided by 1,000.
posterior_thinning	Thinning interval for the sampler. Default value is 500.

Value

A data frame with five columns. validated is a logical value indicating whether the stratum is estimable, N is the draws from the posterior distribution (NA if the stratum is not estimable), valid_sources is a string indicating which sources were used in the estimation, n_obs is the number of observations on valid lists in the stratum of interest (NA if the stratum is not estimable), and stratum_name is a stratum identifier. If the stratum is estimable the return will consist of n_samples divided by 1,000 rows.

Examples

```
set.seed(19481210)
in_A <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.45, 0.65))
in_B <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.5, 0.5))
in_C <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.75, 0.25))
in_D <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(1, 0))

my_stratum <- tibble::tibble(in_A, in_B, in_C, in_D)
mse(stratum_data = my_stratum, stratum_name = "my_stratum")
```

proportions_imputed *Calculate the proportions of each level of a variable after applying* combine_replicates *to completed data (includes imputed values).*

Description

Calculate the proportions of each level of a variable after applying combine_replicates to completed data (includes imputed values).

Usage

```
proportions_imputed(complete_data, strata_vars, digits = 2)
```

Arguments

complete_data	A data frame containing the output from combine_replicates.
strata_vars	A vector of column names identifying the variables to be used for stratification.
digits	Number of decimal places to round the results to. Default value is 2.

Value

A data frame that contains the proportions after applying combine_replicates.

Examples

```
local_dir <- system.file("extdata", "right", package = "verdata")
replicates_data <- read_replicates(replicates_dir = local_dir,
violation = "reclutamiento", replicate_nums = c(1, 2), version = "v1",
crash = TRUE)
replicates_obs_data <- summary_observed("reclutamiento", replicates_data,
strata_vars = "sexo", conflict_filter = FALSE, forced_dis_filter = FALSE,
edad_minors_filter = FALSE, include_props = FALSE)
tab_combine <- combine_replicates("reclutamiento", replicates_obs_data,
replicates_data, strata_vars = 'sexo', conflict_filter = TRUE,
forced_dis_filter = FALSE, edad_minors_filter = FALSE, include_props = FALSE)
prop_data_complete <- proportions_imputed(tab_combine, strata_vars = "sexo",
digits = 2)
```

proportions_observed *Calculate the proportions of each level of a variable after applying* summary_observed *to observed values.*

Description

Calculate the proportions of each level of a variable after applying summary_observed to observed values.

Usage

```
proportions_observed(obs_data, strata_vars, digits = 2)
```

Arguments

obs_data	A data frame containing the output from summary_observed.
strata_vars	A vector of column names identifying the variables to be used for stratification.
digits	Number of decimal places to round the results to. Default is 2.

Value

A data frame that contains the proportions after applying summary_observed.

Examples

```
local_dir <- system.file("extdata", "right", package = "verdata")
replicates_data <- read_replicates(local_dir, "reclutamiento", c(1, 2), version = "v1")
tab_observed <- summary_observed("reclutamiento", replicates_data,
strata_vars = "sexo", conflict_filter = TRUE, forced_dis_filter = FALSE,
edad_minors_filter = TRUE, include_props = TRUE)
prop_data <- proportions_observed(tab_observed, strata_vars = "sexo",
digits = 2)
```

read_replicates *Read replicates in a directory and verify they are identical to the ones published.*

Description

Read replicates in a directory and verify they are identical to the ones published.

Usage

```
read_replicates(
  replicates_dir,
  violation,
  replicate_nums,
  version,
  crash = TRUE
)
```

Arguments

replicates_dir	A path to the directory containing the replicates. Then file name of each replicate must contain at least the name of the violation in Spanish and lower case letters (homicidio, secuestro, reclutamiento, desaparicion), and the replicate number preceded by "R", (e.g., "R1" for replicate 1).
violation	A string indicating the violation being analyzed. Options are "homicidio", "secuestro", "reclutamiento", and "desaparicion".
replicate_nums	A numeric vector containing the replicates to be analyzed. Values in the vector should be between 1 and 100 inclusive.
version	Version of the data being read in. Options are "v1" or "v2". "v1" is appropriate for replicating the replicating the results of the joint JEP-CEV-HRDAG project. "v2" is appropriate for conducting your new analyses of the conflict in Colombia.
crash	A parameter to define whether the function should crash if the content of the file is not identical to the one published. If crash = TRUE (default), it will return error and not read the data, if crash = FALSE, the function will return a warning but still read the data.

Value

A data frame with the data from all indicated replicates.

Examples

```
local_dir <- system.file("extdata", "right", package = "verdata")
read_replicates(local_dir, "reclutamiento", 1, 2, version = "v1")
```

---

run_lcmcr *Calculate multiple systems estimation estimates using the Bayesian Non-Parametric Latent-Class Capture-Recapture model developed by Daniel Manrique-Vallier (2016).*

Description

Calculate multiple systems estimation estimates using the Bayesian Non-Parametric Latent-Class Capture-Recapture model developed by Daniel Manrique-Vallier (2016).

Usage

```
run_lcmcr(
  stratum_data_prepped,
  stratum_name,
  min_n = 1,
  K,
  buffer_size,
  sampler_thinning,
  seed,
  burnin,
  n_samples,
  posterior_thinning
)
```

Arguments

stratum_data_prepped	A data frame with all records in the stratum of interest documented by sources considered valid for estimation (i.e., there should be no rows with all 0's). Columns indicating sources should be prefixed with in_ and should be numeric with 1 indicating that an individual was documented in the source and 0 indicating that an individual was not documented in the source.
stratum_name	An identifier for the stratum.
min_n	The minimum number of records that must appear in a source to be considered valid for estimation. min_n should never be less than or equal to 0; the default value is 1.
K	The maximum number of latent classes to fit.
buffer_size	Size of the tracing buffer.
sampler_thinning	Thinning interval for the tracing buffer.
seed	Integer seed for the internal random number generator.
burnin	Number of burn in iterations.
n_samples	Number of samples to be generated. Samples are taken one every posterior_thinning iterations of the sampler. Final number of samples from the posterior is n_samples divided by 1,000.
posterior_thinning	Thinning interval for the sampler.

Value

A data frame with four columns and n_samples divided by 1,000 rows. N is the draws from the posterior distribution, valid_sources is a string indicating which sources were used in the estimation, n_obs is the number of observations in the stratum of interest, and stratum_name is the stratum identifier.

References

Manrique-Vallier D (2016). "Bayesian population size estimation using Dirichlet process mixtures." *Biometrics*, **72**(4), 1246–1254. doi:10.1111/biom.12502.

Examples

```
set.seed(19481210)
library(dplyr)

in_A <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.45, 0.65))
in_B <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.5, 0.5))
in_C <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(0.75, 0.25))
in_D <- sample(c(0, 1), size = 100, replace = TRUE, prob = c(1, 0))

my_stratum <- tibble::tibble(in_A, in_B, in_C, in_D) %>%
    dplyr::mutate(rs = rowSums(.)) %>%
    dplyr::filter(rs >= 1) %>%
    dplyr::select(-rs)
run_lcmcr(stratum_data_prepped = my_stratum, stratum_name = "my_stratum",
          K = 4, buffer_size = 10000, sampler_thinning = 1000, seed = 19481210,
          burnin = 10000, n_samples = 10000, posterior_thinning = 500)
```

`stratification` *Data documenting the stratifications used to replicate the results of the methodological report of the joint JEP-CEV-HRDAG project (version in English).*

Description

Data documenting the stratifications used to replicate the results of the methodological report of the joint JEP-CEV-HRDAG project (version in English).

Usage

```
data(stratification)
```

Format

A data frame with 31 rows and 4 variables.

violation the human rights violation being analyzed

estimation the type of analysis the stratification was used for (e.g., patterns of violence by year, sex, etc.)

stratification the variables used to stratify the estimates

notes additional notes about the stratification; NA if no notes

Source

Joint JEP-CEV-HRDAG project.

`summary_observed` *Summary statistics for observed data.*

Description

Summary statistics for observed data.

Usage

```
summary_observed(
  violation,
  replicates_data,
  strata_vars = NULL,
  conflict_filter = FALSE,
  forced_dis_filter = FALSE,
  edad_minors_filter = FALSE,
  include_props = FALSE,
  digits = 2
)
```

Arguments

`violation`	Violation to be analyzed. Options are "homicidio", "secuestro", "reclutamiento", and "desaparicion".
`replicates_data`	Data frame containing replicate data.
`strata_vars`	Variable to be analyzed. Before imputation this variable may have missing values.
`conflict_filter`	Filter that indicates if the data is filtered by the rule "is_conflict" or not.
`forced_dis_filter`	Filter that indicates if the data is filter by the rule "is_forced_dis" or not.
`edad_minors_filter`	Optional filter by age ("edad") < 18.
`include_props`	A logical value indicating whether or not to include the proportions from the calculations.
`digits`	Number of decimal places to round the results to. Default is 2.

Value

A data frame with two or more columns, (1) name of variable(s) and (2) the number of observations in each of the variable's categories.

Examples

```
local_dir <- system.file("extdata", "right", package = "verdata")
replicates_data <- read_replicates(local_dir, "reclutamiento", c(1, 2), version = "v1")
tab_observed <- summary_observed("reclutamiento", replicates_data,
strata_vars = "sexo", conflict_filter = FALSE, forced_dis_filter = FALSE,
edad_minors_filter = FALSE, include_props = FALSE, digits = 2)
```

Index

∗ **datasets**
diccionario_replicas, 6
diccionario_vars_adicional, 7
estratificacion, 8
stratification, 17

combine_estimates, 2
combine_replicates, 4
confirm_files, 5

diccionario_replicas, 6
diccionario_vars_adicional, 7

estimates_exist, 7
estratificacion, 8

filter_standard_cev, 9

get_valid_sources, 9

lookup_estimates, 10

mse, 11

proportions_imputed, 13
proportions_observed, 14

read_replicates, 14
run_lcmcr, 15

stratification, 17
summary_observed, 18