

Package ‘retroharmonize’

November 2, 2021

Type Package

Title Ex Post Survey Data Harmonization

Version 0.2.0

Date 2021-11-02

Maintainer Daniel Antal <daniel.antal@ceemid.eu>

Description Assist in reproducible retrospective (ex-post) harmonization of data, particularly individual level survey data, by providing tools for organizing metadata, standardizing the coding of variables, and variable names and value labels, including missing values, and documenting the data transformations, with the help of comprehensive s3 classes.

License GPL-3

URL <https://retroharmonize.dataobservatory.eu/>,
<https://ropengov.github.io/retroharmonize/>,
<https://github.com/rOpenGov/retroharmonize>

BugReports <https://github.com/rOpenGov/retroharmonize/issues>

Depends R (>= 3.5.0)

Imports assertthat, dplyr (>= 1.0.0), fs, glue, haven, here, labelled, magrittr, methods, pillar, purrr, rlang, snakecase, stats, stringr, tibble, tidyr, tidyselect, utils, vctrs

Suggests covr, ggplot2, knitr, markdown, png, rmarkdown, spelling, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

Language en-US

RoxygenNote 7.1.2

X-schema.org-isPartOf <http://ropengov.org/>

X-schema.org-keywords ropengov

NeedsCompilation no

Author Daniel Antal [aut, cre] (<<https://orcid.org/0000-0001-7513-6760>>),
 Marta Kolczynska [ctb] (<<https://orcid.org/0000-0003-4981-0437>>),
 Pyry Kantanen [ctb] (<<https://orcid.org/0000-0003-2853-2765>>),
 Diego Hernangómez Herrero [ctb]
 (<<https://orcid.org/0000-0001-8457-4658>>)

Repository CRAN

Date/Publication 2021-11-02 22:20:12 UTC

R topics documented:

as_factor	3
as_labelled_spss_survey	3
collect_val_labels	4
concatenate	5
create_codebook	6
document_survey_item	7
document_waves	8
harmonize_na_values	9
harmonize_values	10
harmonize_var_names	11
harmonize_waves	13
labelled_spss_survey	14
label_normalize	16
merge_waves	17
metadata_create	18
na_range_to_values	19
pull_survey	20
read_dta	21
read_rds	22
read_spss	23
read_surveys	24
retroharmonize	25
subset_save_surveys	26
subset_waves	27
suggest_permanent_names	28
suggest_var_names	29
survey	30

Index

32

as_factor	<i>Convert labelled_spss_survey vector To Factor</i>
-----------	--

Description

Convert a `labelled_spss_survey` vector to a type of factor. Keeps only the levels and class attributes.

Usage

```
as_factor(x, levels = "default", ordered = FALSE)
```

Arguments

x	Object to coerce to a factor.
levels	How to create the levels of the generated factor: <ul style="list-style-type: none"> • "default": uses labels where available, otherwise the values. Labels are sorted by value. • "both": like "default", but pastes together the level and value • "label": use only the labels; unlabelled values become NA • "values": use only the values
ordered	If TRUE create an ordered (ordinal) factor, if FALSE (the default) create a regular (nominal) factor.

See Also

as_factor is imported from haven: [:as_factor](#)

as_labelled_spss_survey	<i>Labelled to labelled_spss_survey</i>
-------------------------	---

Description

Labelled to labelled_spss_survey

Usage

```
as_labelled_spss_survey(x, id)
```

Arguments

x	A vector of class haven_labelled or haven_labelled_spss.
id	The survey identifier.

Value

A vector of `labelled_spss_survey`

See Also

Other type conversion functions: [labelled_spss_survey\(\)](#)

`collect_val_labels` *Collect labels from metadata file*

Description

Collect labels from metadata file

Usage

```
collect_val_labels(metadata)
```

```
collect_na_labels(metadata)
```

Arguments

`metadata` A metadata data frame created by [metadata_create](#).

Value

The unique valid labels or the user-defined missing labels found in all the files analyzed in metadata.

See Also

Other harmonization functions: [harmonize_na_values\(\)](#), [harmonize_values\(\)](#), [harmonize_var_names\(\)](#), [label_normalize\(\)](#), [suggest_permanent_names\(\)](#), [suggest_var_names\(\)](#)

Examples

```
test_survey <- retroharmonize::read_rds (
  file = system.file("examples", "ZA7576.rds",
    package = "retroharmonize"),
  id = "test"
)
example_metadata <- metadata_create (test_survey)

collect_val_labels (metadata = example_metadata )
collect_na_labels ( metadata = example_metadata )
```

`concatenate`*Concatenate haven_labelled_spss vectors*

Description

Concatenate `haven_labelled_spss` vectors

Usage

```
concatenate(x, y)
```

Arguments

`x` A `haven_labelled_spss` vector.
`y` A `haven_labelled_spss` vector.

Value

A concatenated `haven_labelled_spss` vector. Returns an error if the attributes do not match. Gives a warning when only the variable label do not match.

Examples

```
v1 <- labelled::labelled(  
  c(3,4,4,3,8, 9),  
  c(YES = 3, NO = 4, `WRONG LABEL` = 8, REFUSED = 9)  
)  
v2 <- labelled::labelled(  
  c(4,3,3,9),  
  c(YES = 3, NO = 4, `WRONG LABEL` = 8, REFUSED = 9)  
)  
s1 <- haven::labelled_spss(  
  x = unclass(v1),                    # remove labels from earlier defined  
  labels = labelled::val_labels(v1), # use the labels from earlier defined  
  na_values = NULL,  
  na_range = 8:9,  
  label = "Variable Example"  
)  
  
s2 <- haven::labelled_spss(  
  x = unclass(v2),                    # remove labels from earlier defined  
  labels = labelled::val_labels(v2), # use the labels from earlier defined  
  na_values = NULL,  
  na_range = 8:9,  
  label = "Variable Example"  
)  
concatenate (s1,s2)
```

create_codebook	<i>Create a codebook</i>
-----------------	--------------------------

Description

Create a codebook from one or more survey data files.

Usage

```
create_codebook(metadata = NULL, survey = NULL)
```

```
codebook_waves_create(waves)
```

Arguments

metadata	A metadata table created by metadata_create . Defaults to NULL.
survey	A survey data frame, defaults to NULL. If the survey is given as parameter, the metadata will be set to the metadata of this particular survey by metadata_create .
waves	A list of surveys.

Details

For a list of survey waves, use `codebook_waves_create`. The returned codebook contains only labelled variables, i.e., numeric and character types are not included, because they do not require coding.

Value

A codebook for the survey as a data frame, including the metadata, and all found SPSS-type valid or missing labels.

See Also

Other metadata functions: [metadata_create\(\)](#)

Other metadata functions: [metadata_create\(\)](#)

Examples

```
create_codebook (  
  survey = read_rds (  
    system.file("examples", "ZA7576.rds",  
               package = "retroharmonize")  
  )  
)  
  
examples_dir <- system.file("examples", package = "retroharmonize")  
survey_list <- dir(examples_dir)[grepl("\\.rds", dir(examples_dir))]
```

```
example_surveys <- read_surveys(  
  file.path( examples_dir, survey_list),  
  save_to_rds = FALSE)  
  
codebook_waves_create (example_surveys)
```

document_survey_item *Document survey item harmonization*

Description

Document the current and historic coding and labelling of the variable.

Usage

```
document_survey_item(x)
```

Arguments

x A labelled_spss_survey vector from a single survey or concatenated from several surveys.

Value

Returns a list of the current and historic coding, labelling of the valid range and missing values or range, the history of the variable names and the history of the survey IDs.

See Also

Other documentation functions: [document_waves\(\)](#)

Examples

```
var1 <- labelled::labelled_spss(  
  x = c(1,0,1,1,0,8,9),  
  labels = c("TRUST" = 1,  
            "NOT TRUST" = 0,  
            "DON'T KNOW" = 8,  
            "INAP. HERE" = 9),  
  na_values = c(8,9))  
  
var2 <- labelled::labelled_spss(  
  x = c(2,2,8,9,1,1 ),  
  labels = c("Tend to trust" = 1,  
            "Tend not to trust" = 2,  
            "DK" = 8,  
            "Inap" = 9),  
  na_values = c(8,9))
```

```

h1 <- harmonize_values (
  x = var1,
  harmonize_label = "Do you trust the European Union?",
  harmonize_labels = list (
    from = c("^tend\\sto|^trust", "^tend\\snot|not\\s\\strust", "^dk|^don", "^inap"),
    to = c("trust", "not_trust", "do_not_know", "inap"),
    numeric_values = c(1,0,99997, 99999)),
  na_values = c("do_not_know" = 99997,
               "inap" = 99999),
  id = "survey1",
)

h2 <- harmonize_values (
  x = var2,
  harmonize_label = "Do you trust the European Union?",
  harmonize_labels = list (
    from = c("^tend\\sto|^trust", "^tend\\snot|not\\s\\strust", "^dk|^don", "^inap"),
    to = c("trust", "not_trust", "do_not_know", "inap"),
    numeric_values = c(1,0,99997, 99999)),
  na_values = c("do_not_know" = 99997,
               "inap" = 99999),
  id = "survey2"
)

h3 <- concatenate(h1, h2)
document_survey_item(h3)

```

document_waves

Document survey lists

Description

Document the key attributes surveys in a survey list.

Usage

```
document_waves(survey_list)
```

Arguments

survey_list A list of [survey](#) objects.

Value

Returns a data frame with the key attributes of the surveys in a survey list: the name of the data file, the number of rows and columns, and the size of the object as stored in memory.

See Also

Other documentation functions: [document_survey_item\(\)](#)

Examples

```
examples_dir <- system.file( "examples", package = "retroharmonize")

my_rds_files <- dir( examples_dir)[grepl(".rds",
                                         dir(examples_dir))]

example_surveys <- read_surveys(file.path(examples_dir, my_rds_files))

waves_document <- document_waves(example_surveys)

attr(waves_document, "original_list" )
waves_document
```

harmonize_na_values *Harmonize na_values in haven_labelled_spss*

Description

Harmonize na_values in haven_labelled_spss

Usage

```
harmonize_na_values(df)
```

Arguments

df A data frame that contains haven_labelled_spss vectors.

Value

A tibble where the na_values are consistent

See Also

Other harmonization functions: [collect_val_labels\(\)](#), [harmonize_values\(\)](#), [harmonize_var_names\(\)](#), [label_normalize\(\)](#), [suggest_permanent_names\(\)](#), [suggest_var_names\(\)](#)

Examples

```
examples_dir <- system.file(
  "examples", package = "retroharmonize"
)

test_read <- read_rds (
  file.path(examples_dir, "ZA7576.rds"),
  id = "ZA7576",
  doi = "test_doi")
```

```
harmonize_na_values(test_read)
```

harmonize_values *Harmonize the values and labels of labelled vectors*

Description

Harmonize the values and labels of labelled vectors

Usage

```
harmonize_values(
  x,
  harmonize_label = NULL,
  harmonize_labels = NULL,
  na_values = c(do_not_know = 99997, declined = 99998, inap = 99999),
  na_range = NULL,
  id = "survey_id",
  name_orig = NULL,
  remove = NULL,
  perl = FALSE
)
```

Arguments

x	A labelled vector
harmonize_label	A character vector of 1L containing the new, harmonize variable label. Defaults to NULL, in which case it uses the variable label of x, unless it is also NULL.
harmonize_labels	A list of harmonization values
na_values	A named vector of na_values, the observations that are defined to be treated as missing in the SPSS-style coding.
na_range	A min, max range of na_range, the continuous missing value range. In most surveys this should be left NULL.
id	A survey ID, defaults to survey_id
name_orig	The original name of the variable. If left NULL it uses the latest name of the object x.
remove	Defaults to NULL. A character or regex that will be removed from all old value labels, like "\\(\\)" for (and).
perl	Use perl-like regex? Defaults to FALSE.

Value

A labelled vector that contains in its metadata attributes the original labelling, the original numeric coding and the current labelling, with the numerical values representing the harmonized coding.

See Also

Other variable label harmonization functions: [harmonize_waves\(\)](#), [label_normalize\(\)](#), [na_range_to_values\(\)](#)

Other harmonization functions: [collect_val_labels\(\)](#), [harmonize_na_values\(\)](#), [harmonize_var_names\(\)](#), [label_normalize\(\)](#), [suggest_permanent_names\(\)](#), [suggest_var_names\(\)](#)

Examples

```
var1 <- labelled::labelled_spss(
  x = c(1,0,1,1,0,8,9),
  labels = c("TRUST" = 1,
            "NOT TRUST" = 0,
            "DON'T KNOW" = 8,
            "INAP. HERE" = 9),
  na_values = c(8,9))

harmonize_values (
  var1,
  harmonize_labels = list (
    from = c("^tend\\sto|^trust", "^tend\\snot|not\\strust", "^dk|^don", "^inap"),
    to = c("trust", "not_trust", "do_not_know", "inap"),
    numeric_values = c(1,0,99997, 99999)),
  na_values = c("do_not_know" = 99997,
               "inap" = 99999),
  id = "survey_id"
)
```

harmonize_var_names *Harmonize the variable names of surveys*

Description

The function harmonizes the variable names of surveys (of class `survey`) that are imported from an external file as a wave.

Usage

```
harmonize_var_names(
  waves,
  metadata,
  old = "var_name_orig",
  new = "var_name_suggested",
  rowids = TRUE
)
```

Arguments

waves	A list of surveys imported with read_surveys .
metadata	A metadata table created by <code>metadata_create</code> and binded together for all surveys in waves.
old	The column name in metadata that contains the old, not harmonized variable names.
new	The column name in metadata that contains the new, harmonized variable names.
rowids	Rename var labels of original vars rowid to simply uniqid?

Details

If the metadata that contains subsetting information is subsetted, then it will subset the surveys in waves.

Value

The list of surveys with harmonized variable names.

See Also

Other harmonization functions: [collect_val_labels\(\)](#), [harmonize_na_values\(\)](#), [harmonize_values\(\)](#), [label_normalize\(\)](#), [suggest_permanent_names\(\)](#), [suggest_var_names\(\)](#)

Examples

```
examples_dir <- system.file("examples", package = "retroharmonize")
survey_list <- dir(examples_dir)[grepl("\\.rds", dir(examples_dir))]

example_surveys <- read_surveys(
  file.path( examples_dir, survey_list),
  save_to_rds = FALSE)
metadata <- lapply ( X = example_surveys, FUN = metadata_create )
metadata <- do.call(rbind, metadata)

metadata$var_name_suggested <- label_normalize(metadata$var_name)

metadata$var_name_suggested[metadata$label_orig == "age education"] <- "age_education"

harmonize_var_names(waves = example_surveys,
  metadata = metadata )
```

harmonize_waves	<i>Harmonize waves</i>
-----------------	------------------------

Description

Harmonize the values of surveys.

Usage

```
harmonize_waves(waves, .f, status_message = FALSE)
```

Arguments

`waves` A list of surveys

`.f` A function to apply for the harmonization.

`status_message` Defaults to FALSE. If set to TRUE it shows the id of the survey that is being joined.

Details

The functions binds together variables that are all present in the surveys, and applies a harmonization function `.f` on them.

Value

A natural full join of all surveys in a single data frame.

See Also

Other variable label harmonization functions: [harmonize_values\(\)](#), [label_normalize\(\)](#), [na_range_to_values\(\)](#)

Examples

```
examples_dir <- system.file("examples", package = "retroharmonize")
survey_list <- dir(examples_dir)[grepl("\\.rds", dir(examples_dir))]

example_surveys <- read_surveys(
  file.path( examples_dir, survey_list),
  save_to_rds = FALSE)

metadata <- lapply ( X = example_surveys, FUN = metadata_create )
metadata <- do.call(rbind, metadata)

to_harmonize <- metadata %>%
  dplyr::filter ( var_name_orig %in%
    c("rowid", "w1") |
    grepl("trust ", label_orig ) ) %>%
  dplyr::mutate ( var_label = var_label_normalize(label_orig)) %>%
  dplyr::mutate ( var_name = val_label_normalize(var_label))
```

```

harmonize_eb_trust <- function(x) {
  label_list <- list(
    from = c("^tend\\snot", "^cannot", "^tend\\sto", "^can\\srely",
             "^dk", "^inap", "na"),
    to = c("not_trust", "not_trust", "trust", "trust",
           "do_not_know", "inap", "inap"),
    numeric_values = c(0,0,1,1, 99997,99999,99999)
  )

  harmonize_values(x,
    harmonize_labels = label_list,
    na_values = c("do_not_know"=99997,
                  "declined"=99998,
                  "inap"=99999)
  )
}

merged_surveys <- merge_waves ( example_surveys, var_harmonization = to_harmonize )

harmonized <- harmonize_waves(waves = merged_surveys,
  .f = harmonize_eb_trust,
  status_message = FALSE)

# For details see Afrobarometer and Eurobarometer Case Study vignettes.

```

labelled_spss_survey *Labelled vectors for multiple SPSS surveys*

Description

This class is amending `haven::labelled_spss` with a unique object identifier `id` to make later binding or joining reproducible and well-documented.

Usage

```

labelled_spss_survey(
  x = double(),
  labels = NULL,
  na_values = NULL,
  na_range = NULL,
  label = NULL,
  id = NULL,
  name_orig = NULL
)

as_character(x)

```

```
is.labelled_spss_survey(x)
```

```
as_numeric(x)
```

Arguments

x	A vector to label. Must be either numeric (integer or double) or character.
labels	A named vector or NULL. The vector should be the same type as x. Unlike factors, labels don't need to be exhaustive: only a fraction of the values might be labelled.
na_values	A vector of values that should also be considered as missing.
na_range	A numeric vector of length two giving the (inclusive) extents of the range. Use -Inf and Inf if you want the range to be open ended.
label	A short, human-readable description of the vector.
id	Survey ID
name_orig	The original name of the variable. If left NULL it uses the latest name of the object x.

Details

It inherits many methods from labelled, but uses more strict coercion and validation rules.

See Also

`as_factor`

Other type conversion functions: [as_labelled_spss_survey\(\)](#)

Other type conversion functions: [as_labelled_spss_survey\(\)](#)

Examples

```
x1 <- labelled_spss_survey(  
  1:10, c(Good = 1, Bad = 8),  
  na_values = c(9, 10),  
  id = "survey1")  
  
is.na(x1)  
  
# Print data and metadata  
print(x1)  
  
x2 <- labelled_spss_survey( 1:10,  
  labels = c(Good = 1, Bad = 8),  
  na_range = c(9, Inf),  
  label = "Quality rating",  
  id = "survey1")  
  
is.na(x2)
```

```
# Print data and metadata
x2
```

label_normalize	<i>Normalize value and variable labels</i>
-----------------	--

Description

label_normalize removes special characters, whitespace, and other typical typing errors.

Usage

```
label_normalize(x)

var_label_normalize(x)

val_label_normalize(x)
```

Arguments

x A character vector of labels to be normalized.

Details

var_label_normalize changes the vector to snake_case. val_label_normalize removes possible chunks from question identifiers.

The functions var_label_normalize and val_label_normalize may be differently implemented for various survey series.

See Also

Other variable label harmonization functions: [harmonize_values\(\)](#), [harmonize_waves\(\)](#), [na_range_to_values\(\)](#)

Other harmonization functions: [collect_val_labels\(\)](#), [harmonize_na_values\(\)](#), [harmonize_values\(\)](#), [harmonize_var_names\(\)](#), [suggest_permanent_names\(\)](#), [suggest_var_names\(\)](#)

Other harmonization functions: [collect_val_labels\(\)](#), [harmonize_na_values\(\)](#), [harmonize_values\(\)](#), [harmonize_var_names\(\)](#), [suggest_permanent_names\(\)](#), [suggest_var_names\(\)](#)

Examples

```
label_normalize (
  c("Don't know", " TRUST", "DO NOT TRUST",
    "inap in Q.3", "Not 100%", "TRUST < 50%",
    "TRUST >=90%", "Verify & Check", "TRUST 99%+")
)

var_label_normalize (
  c("Q1_Do you trust the national government?",
    " Do you trust the European Commission")
)
```



```

    )

    val_label_normalize (
      c("Q1_Do you trust the national government?",
        " Do you trust the European Commission")
    )

```

merge_waves	<i>Merge waves</i>
-------------	--------------------

Description

Merge a list of surveys into a list with harmonized variable names, variable labels and survey identifiers.

Usage

```
merge_waves(waves, var_harmonization)
```

Arguments

waves	A list of surveys
var_harmonization	Metadata of surveys, including at least filename, var_name_orig, var_name, var_label.

Value

A list of surveys with harmonized names and variable labels.

See Also

survey

Examples

```

examples_dir <- system.file("examples", package = "retroharmonize")
survey_list <- dir(examples_dir)[grepl("\\.rds", dir(examples_dir))]

example_surveys <- read_surveys(
  file.path( examples_dir, survey_list),
  save_to_rds = FALSE)

metadata <- metadata_waves_create(example_surveys)

to_harmonize <- metadata %>%
  dplyr::filter ( var_name_orig %in%
    c("rowid", "w1") |

```

```

      grepl("trust ", label_orig ) ) %>%
dplyr::mutate ( var_label = var_label_normalize(label_orig) ) %>%
dplyr::mutate ( var_name = val_label_normalize(var_label) )

merge_waves ( example_surveys, to_harmonize )

```

metadata_create	<i>Create a metadata table</i>
-----------------	--------------------------------

Description

Create a metadata table from the survey data files.

Usage

```

metadata_create(survey)

metadata_waves_create(survey_list)

```

Arguments

survey	A survey data frame.
survey_list	A list containing surveys of class survey.

Details

A data frame like tibble object is returned. In case you are working with a list of surveys (waves), call [metadata_waves_create](#), which is a wrapper around a list of [metadata_create](#) calls.

The structure of the returned tibble:

filename The original file name; if present; missing, if a non-[survey](#) data frame is used as input survey.

id The ID of the survey, if present; missing, if a non-[survey](#) data frame is used as input survey.

var_name_orig The original variable name in SPSS.

class_orig The original variable class after importing with [read_spss](#).

label_orig The original variable label in SPSS.

labels A list of the value labels.

valid_labels A list of the value labels that are not marked as missing values.

na_labels A list of the value labels that refer to user-defined missing values.

na_range An optional range of a continuous missing range, if present in the vector.

n_labels Number of categories or unique levels, which may be different from the sum of missing and category labels.

n_valid_labels Number of categories in the non-missing range.

n_na_labels Number of categories of the variable, should be the sum of the former two.

na_levels A list of the user-defined missing values.

Value

A nested data frame with metadata and the range of labels, na_values and the na_range itself.

See Also

Other metadata functions: [create_codebook\(\)](#)

Other metadata functions: [create_codebook\(\)](#)

Examples

```
metadata_create (
  survey = read_rds (
    system.file("examples", "ZA7576.rds",
               package = "retroharmonize")
  )
)
examples_dir <- system.file( "examples", package = "retroharmonize")

my_rds_files <- dir( examples_dir)[grepl(".rds",
                                       dir(examples_dir))]

example_surveys <- read_surveys(file.path(examples_dir, my_rds_files))
metadata_waves_create (example_surveys)
```

na_range_to_values *Harmonize user-defined missing value ranges*

Description

Harmonize the na_values attribute with na_range, if the latter is present.

Usage

```
na_range_to_values(x)
```

```
is.na_range_to_values(x)
```

Arguments

x A labelled_spss or labelled_spss_survey vector

Details

na_range_to_values() tests if the function needs to be called for na_values harmonization. The na_range is often missing and less likely to cause logical problems when joining survey answers.

Value

A `x` with harmonized `na_values` and `na_range` attributes. If `min(na_values)` or `max(na_values)` than the left- and right-hand value of `na_range`, it gives a warning and adjusts the original `na_range`.

See Also

Other variable label harmonization functions: [harmonize_values\(\)](#), [harmonize_waves\(\)](#), [label_normalize\(\)](#)

Examples

```
var1 <- labelled::labelled_spss(
  x = c(1,0,1,1,0,8,9),
  labels = c("TRUST" = 1,
             "NOT TRUST" = 0,
             "DON'T KNOW" = 8,
             "INAP. HERE" = 9),
  na_range = c(8,12))

na_range_to_values(var1)
as_numeric(na_range_to_values(var1))
as_character(na_range_to_values(var1))
```

pull_survey

Pull a survey from a survey list

Description

Pull a survey by survey code or id.

Usage

```
pull_survey(survey_list, id = NULL, filename = NULL)
```

Arguments

<code>survey_list</code>	A list of surveys
<code>id</code>	The id of the requested survey. If NULL use filename
<code>filename</code>	The filename of the requested survey.

Value

A single survey identified by id or filename.

See Also

Other import functions: [read_dta\(\)](#), [read_rds\(\)](#), [read_spss\(\)](#), [read_surveys\(\)](#), [subset_save_surveys\(\)](#)

Examples

```

examples_dir <- system.file( "examples", package = "retroharmonize")

my_rds_files <- dir( examples_dir)[grepl(".rds",
                                         dir(examples_dir))]

example_surveys <- read_surveys(
  file.path(examples_dir, my_rds_files) )

pull_survey(example_surveys, id = "ZA5913")

```

read_dta	<i>Read Stata DTA files (‘.dta’) files</i>
----------	--

Description

This is a wrapper around `haven::read_dta` with some exception handling.

Usage

```
read_dta(file, id = NULL, filename = NULL, doi = NULL, .name_repair = "unique")
```

Arguments

<code>file</code>	A STATA file.
<code>id</code>	An identifier of the tibble, if omitted, defaults to the file name.
<code>filename</code>	An import file name.
<code>doi</code>	An optional document object identifier.
<code>.name_repair</code>	Defaults to "unique" See <code>tibble::as_tibble</code> for details.

Details

`read_dta()` reads both `‘.dta’` files.

The function is not yet tested.

Value

A tibble.

Variable labels are stored in the "label" attribute of each variable. It is not printed on the console, but the RStudio viewer will show it.

`write_sav()` returns the input `‘data’` invisibly.

See Also

Other import functions: [pull_survey\(\)](#), [read_rds\(\)](#), [read_spss\(\)](#), [read_surveys\(\)](#), [subset_save_surveys\(\)](#)

Examples

```
path <- system.file("examples", "iris.dta", package = "haven")
read_dta(path)
```

read_rds	<i>Read survey from rds file</i>
----------	----------------------------------

Description

Read survey from rds file

Usage

```
read_rds(file, id = NULL, filename = NULL, doi = NULL)
```

Arguments

file	A re-saved survey, imported with <code>haven::read_spss</code>
id	An identifier of the tibble, if omitted, defaults to the file name.
filename	An import file name.
doi	An optional document object identifier.

Value

A tibble, data frame variant with survey attributes.

See Also

Other import functions: [pull_survey\(\)](#), [read_dta\(\)](#), [read_spss\(\)](#), [read_surveys\(\)](#), [subset_save_surveys\(\)](#)

Examples

```
path <- system.file("examples", "ZA7576.rds", package = "retroharmonize")
read_survey <- read_rds(path)
attr(read_survey, "id")
attr(read_survey, "filename")
attr(read_survey, "doi")
```

read_spss	<i>Read SPSS ('.sav', '.zsav', '.por') files. Write '.sav' and '.zsav' files.</i>
-----------	---

Description

This is a wrapper around `haven::read_spss` with some exception handling.

Usage

```
read_spss(  
  file,  
  user_na = TRUE,  
  id = NULL,  
  filename = NULL,  
  doi = NULL,  
  .name_repair = "unique"  
)
```

Arguments

<code>file</code>	An SPSS file.
<code>user_na</code>	Should user-defined <code>na_values</code> be imported? Defaults to <code>TRUE</code> .
<code>id</code>	An identifier of the tibble, if omitted, defaults to the file name.
<code>filename</code>	An import file name.
<code>doi</code>	An optional document object identifier.
<code>.name_repair</code>	Defaults to "unique" See <code>tibble::as_tibble</code> for details.

Details

`'read_sav()'` reads both `'sav'` and `'zsav'` files; `'write_sav()'` creates `'zsav'` files when `'compress = TRUE'`. `'read_por()'` reads `'por'` files. `'read_spss()'` uses either `'read_por()'` or `'read_sav()'` based on the file extension.

When the SPSS file has columns which are of class labelled, but have no labels, they are read as numeric or character vectors.

Value

A tibble.

Variable labels are stored in the "label" attribute of each variable. It is not printed on the console, but the RStudio viewer will show it.

`'write_sav()'` returns the input `'data'` invisibly.

See Also

Other import functions: [pull_survey\(\)](#), [read_dta\(\)](#), [read_rds\(\)](#), [read_surveys\(\)](#), [subset_save_surveys\(\)](#)

Examples

```

path <- system.file("examples", "iris.sav", package = "haven")
haven::read_sav(path)

tmp <- tempfile(fileext = ".sav")
haven::write_sav(mtcars, tmp)
haven::read_sav(tmp)

```

read_surveys

Read Survey Files

Description

Import surveys into a list. Adds filename as a constant to each element of the list.

Usage

```
read_surveys(import_file_names, .f = "read_rds", save_to_rds = FALSE)
```

Arguments

import_file_names	A vector of file names to import.
.f	A function to import the surveys with. Defaults to 'read_rds'. For SPSS files, read_spss is recommended, which is a well-parameterized version of read_spss that saves some metadata, too.
save_to_rds	Should it save the imported survey to .rds? Defaults to FALSE.

Details

The functions handle exceptions with wrong filenames and not readable files. If a file cannot be read, a warning is given, and empty survey is added to the the list in the place of this file.

Value

A list of the surveys. Each element of the list is a data frame-like [survey](#) type object where some metadata, such as the original file name, doi identifier if present, and other information is recorded for a reproducible workflow.

See Also

[survey](#)

Other import functions: [pull_survey\(\)](#), [read_dta\(\)](#), [read_rds\(\)](#), [read_spss\(\)](#), [subset_save_surveys\(\)](#)

Examples

```
file1 <- system.file(
  "examples", "ZA7576.rds", package = "retroharmonize")
file2 <- system.file(
  "examples", "ZA5913.rds", package = "retroharmonize")

read_surveys (c(file1,file2), .f = 'read_rds' )
```

retroharmonize

retroharmonize: Retrospective harmonization of survey data files

Description

The goal of retroharmonize is to facilitate retrospective (ex-post) harmonization of data, particularly survey data, in a reproducible manner. The package provides tools for organizing the metadata, standardizing the coding of variables, variable names and value labels, including missing values, and for documenting all transformations, with the help of comprehensive S3 classes.

import functions

Read data stored in formats with rich metadata, such as SPSS (.sav) files, and make them usable in a programmatic context.

[read_spss](#): read an SPSS file and record metadata for reproducibility

[read_rds](#): read an rds file and record metadata for reproducibility

[read_surveys](#): programmatically read a list of surveys

[subset_save_surveys](#): programmatically read a list of surveys, and subset them (pre-harmonize the same variables.)

[pull_survey](#): pull a single survey from a survey list.

variable name harmonization functions

[label_normalize](#) removes special characters, whitespace, and other typical typing errors and helps the uniformization of labels and variable names.

[suggest_permanent_names](#): Suggest the use of variable naming conventions.

variable label harmonization functions

Create consistent coding and labelling.

[create_codebook](#): Create a codebook from the original SPSS variable codes and labels.

[harmonize_values](#): Harmonize the label list across surveys.

[harmonize_waves](#): Create a list of surveys with harmonized value labels.

[na_range_to_values](#): Make the na_range attributes, as imported from SPSS, consistent with the na_values attributes.

survey harmonization functions

[merge_waves](#): Create a list of surveys with harmonized names and variable labels.

documentation functions

[metadata_create](#) and [metadata_waves_create](#)
[create_codebook](#) and [codebook_waves_create](#)

Make the workflow reproducible by recording the harmonization process. [document_survey_item](#): Returns a list of the current and historic coding, labelling of the valid range and missing values or range, the history of the variable names and the history of the survey IDs. [document_waves](#): Document the key attributes surveys in a survey list.

type conversion functions

Consistently treat labels and SPSS-style user-defined missing values in the R language. [survey](#) helps constructing a valid survey data frame, and [labelled_spss_survey](#) helps creating a vector for a questionnaire item. [as_numeric](#): convert to numeric values.
[as_factor](#): convert to labels to factor levels.
[as_character](#): convert to labels to characters.
[as_labelled_spss_survey](#): convert labelled and labelled_spss vectors to labelled_spss_survey vectors.

subset_save_surveys *Subset and Save Surveys*

Description

Read a predefined survey list and variables.

Usage

```
subset_save_surveys(  
  var_harmonization,  
  selection_name = "trust",  
  import_path = "",  
  export_path = "working"  
)
```

Arguments

var_harmonization	Metadata of surveys, including at least filename, var_name_orig, var_name, var_label.
selection_name	An identifier for the survey subset.
import_path	The path to the survey files.
export_path	The path where the subsets should be saved.

Value

The function does not return a value. It saves the subsetted surveys into .rds files.

See Also

Other import functions: [pull_survey\(\)](#), [read_dta\(\)](#), [read_rds\(\)](#), [read_spss\(\)](#), [read_surveys\(\)](#)

Examples

```
test_survey <- read_rds (
  file = system.file("examples", "ZA7576.rds",
                    package = "retroharmonize")
)

test_metadata <- metadata_create ( test_survey )
test_metadata <- test_metadata[c(18:37),]
test_metadata$var_name <- var_label_normalize (test_metadata$var_name_orig)
test_metadata$var_label <- test_metadata$label_orig

saveRDS(test_survey, file.path(tempdir(),
                              "ZA7576.rds"),
        version = 2)

subset_save_surveys ( var_harmonization = test_metadata,
                    selection_name = "tested",
                    import_path = tempdir(),
                    export_path = tempdir())

file.exists ( file.path(tempdir(), "ZA7576_tested.rds"))
```

subset_waves

Subset all surveys in a wave

Description

The function harmonizes the variable names of surveys (of class survey) that are imported from an external file as a wave with with [read_surveys](#).

Usage

```
subset_waves(waves, subset_names = NULL)
```

Arguments

waves A list of surveys imported with [read_surveys](#).

subset_names The names of the variables that should be kept from all surveys in the list that contains the wave of surveys. Defaults to NULL in which case it returns all variables without subsetting.

Details

It is likely that you want to harmonize the variable names with `harmonize_var_names` first.

Value

The list of surveys with harmonized variable names.

Examples

```
examples_dir <- system.file("examples", package = "retroharmonize")
survey_list <- dir(examples_dir)[grepl("\\.rds", dir(examples_dir))]

example_surveys <- read_surveys(
  file.path( examples_dir, survey_list),
  save_to_rds = FALSE)
metadata <- metadata_waves_create(example_surveys)

metadata$var_name_suggested <- label_normalize(metadata$var_name)

metadata$var_name_suggested[metadata$label_orig == "age education"] <- "age_education"

hnw <- harmonize_var_names(waves = example_surveys,
  metadata = metadata )

subset_waves (hnw, subset_names = c("uniqid", "w1", "age_education"))
```

suggest_permanent_names

Suggest permanent names

Description

Suggest the use of established naming conventions.

Usage

```
suggest_permanent_names(survey_program = "eurobarometer")
```

Arguments

`survey_program` Suggest permanent names for the survey program "eurobarometer"

Details

Established survey programs usually have their own variable name conventions. The suggested constant names keep these variable names constant.

Value

A character vector with suggested permanent names.

See Also

Other harmonization functions: [collect_val_labels\(\)](#), [harmonize_na_values\(\)](#), [harmonize_values\(\)](#), [harmonize_var_names\(\)](#), [label_normalize\(\)](#), [suggest_var_names\(\)](#)

Examples

```
suggest_permanent_names ( "eurobarometer" )
```

suggest_var_names	<i>Suggest variable names</i>
-------------------	-------------------------------

Description

The function harmonizes the variable names of surveys (of class `survey`) that are imported from an external file as a wave.

Usage

```
suggest_var_names(
  metadata,
  permanent_names = NULL,
  survey_program = NULL,
  case = "snake"
)
```

Arguments

metadata	A metadata table created by <code>metadata_create</code> and binded together for all surveys in waves.
permanent_names	A character vector of names to keep.
survey_program	If <code>permanent_names = NULL</code> then suggest_permanent_names is called with this parameter, unless it is also NULL
case	Unless it is set to NULL it will standardize the suggested variable name with to_any_case . The default is "snake".

Value

A metadata tibble augmented with `$var_name_suggested`

See Also

Other harmonization functions: [collect_val_labels\(\)](#), [harmonize_na_values\(\)](#), [harmonize_values\(\)](#), [harmonize_var_names\(\)](#), [label_normalize\(\)](#), [suggest_permanent_names\(\)](#)

Examples

```

examples_dir <- system.file("examples", package = "retroharmonize")
survey_list <- dir(examples_dir)[grepl("\\.rds", dir(examples_dir))]

example_surveys <- read_surveys(
  file.path(examples_dir, survey_list),
  save_to_rds = FALSE)
metadata <- lapply ( X = example_surveys, FUN = metadata_create )
metadata <- do.call(rbind, metadata)

utils::head(
  suggest_var_names(metadata, survey_program = "eurobarometer" )
)

```

 survey

Survey data frame

Description

Store the data of a survey in a tibble (data frame) with a unique survey identifier, import filename, and optional doi.

Usage

```

survey(
  object = data.frame(),
  id = character(),
  filename = character(),
  doi = character()
)

is.survey(object)

## S3 method for class 'survey'
summary(object, ...)

```

Arguments

object	A tibble or data frame that contains the survey data.
id	A mandatory identifier for the survey
filename	The import file name.
doi	Optional doi, can be omitted.
...	Arguments passed to summary method.

Value

A tibble with id, filename, doi metadata information.

Examples

```
example_survey <- survey(  
  object =data.frame (  
    rowid = 1:6,  
    observations = runif(6)),  
  id = 'example',  
  filename = "no_file"  
)
```

Index

- * **documentation functions**
 - document_survey_item, 7
 - document_waves, 8
 - * **harmonization functions**
 - collect_val_labels, 4
 - harmonize_na_values, 9
 - harmonize_values, 10
 - harmonize_var_names, 11
 - label_normalize, 16
 - suggest_permanent_names, 28
 - suggest_var_names, 29
 - * **import functions**
 - pull_survey, 20
 - read_dta, 21
 - read_rds, 22
 - read_spss, 23
 - read_surveys, 24
 - subset_save_surveys, 26
 - * **joining functions**
 - concatenate, 5
 - * **metadata functions**
 - create_codebook, 6
 - metadata_create, 18
 - * **survey harmonization functions**
 - merge_waves, 17
 - * **type conversion functions**
 - as_labelled_spss_survey, 3
 - labelled_spss_survey, 14
 - * **variable label harmonization functions**
 - harmonize_values, 10
 - harmonize_waves, 13
 - label_normalize, 16
 - na_range_to_values, 19
- as_character, 26
- as_character (labelled_spss_survey), 14
- as_factor, 3, 3, 26
- as_labelled_spss_survey, 3, 15, 26
- as_numeric, 26
- as_numeric (labelled_spss_survey), 14
- as_tibble, 21, 23
- codebook_waves_create, 26
- codebook_waves_create
(create_codebook), 6
- collect_na_labels (collect_val_labels),
4
- collect_val_labels, 4, 9, 11, 12, 16, 29
- concatenate, 5
- create_codebook, 6, 19, 25, 26
- document_survey_item, 7, 8, 26
- document_waves, 7, 8, 26
- harmonize_na_values, 4, 9, 11, 12, 16, 29
- harmonize_values, 4, 9, 10, 12, 13, 16, 20,
25, 29
- harmonize_var_names, 4, 9, 11, 11, 16, 28, 29
- harmonize_waves, 11, 13, 16, 20, 25
- is.labelled_spss_survey
(labelled_spss_survey), 14
- is.na_range_to_values
(na_range_to_values), 19
- is.survey (survey), 30
- label_normalize, 4, 9, 11–13, 16, 20, 25, 29
- labelled_spss, 14
- labelled_spss_survey, 3, 4, 14, 26
- merge_waves, 17, 26
- metadata_create, 4, 6, 18, 18, 26
- metadata_waves_create, 18, 26
- metadata_waves_create
(metadata_create), 18
- na_range_to_values, 11, 13, 16, 19, 25
- pull_survey, 20, 21–25, 27
- read_dta, 20, 21, 21, 22–24, 27

read_rds, [20](#), [21](#), [22](#), [23–25](#), [27](#)
read_spss, [18](#), [20–23](#), [23](#), [24](#), [25](#), [27](#)
read_surveys, [12](#), [20–23](#), [24](#), [25](#), [27](#)
retroharmonize, [25](#)

subset_save_surveys, [20–25](#), [26](#)
subset_waves, [27](#)
suggest_permanent_names, [4](#), [9](#), [11](#), [12](#), [16](#),
[25](#), [28](#), [29](#)
suggest_var_names, [4](#), [9](#), [11](#), [12](#), [16](#), [29](#), [29](#)
summary.survey (survey), [30](#)
survey, [8](#), [18](#), [24](#), [26](#), [30](#)

to_any_case, [29](#)

val_label_normalize (label_normalize),
[16](#)
var_label_normalize (label_normalize),
[16](#)