Package ‘ILSAmerge’

June 14, 2026

Type Package

Title Merge and Download International Large-Scale Assessments (ILSA) Data

Version 1.4.2

Maintainer Andrés Christiansen `<andres.christiansen@iea-hamburg.de>`

Description Merges and downloads 'SPSS' data from different International Large-Scale Assessments (ILSA), including: Trends in International Mathematics and Science Study (TIMSS), Progress in International Reading Literacy Study (PIRLS), and other studies from <`https://www.iea.nl/`>.

License GPL (>= 3.0)

URL `https://github.com/dopatendo/ILSAmerge`, `https://dopatendo.github.io/ILSAmerge/`

Imports haven, httr2

Encoding UTF-8

Suggests knitr, rmarkdown

VignetteBuilder knitr

Depends R (>= 4.1)

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author Andrés Christiansen [aut, cre]

Repository CRAN

Date/Publication 2026-06-14 19:30:02 UTC

Contents

addSchools 2
asthistibble 3
availableILSA 4
combineStudents 4
get.atr 5

ILSAdownload 6
ILSAfile.info 7
ILSAmerge 8
ILSAready 9
ILSArename 11
justload 12
mistoNAs 13
readILSA 14
spss.syntax 15
untibble 16
whichcol 16

Index **18**

`addSchools` *Add school data*

Description

Add school data to student and teacher files merged by `ILSAmerge`. It will run `combineStudents` internally. To see which ILSA are available for adding school data use `availableILSA`.

Usage

```
addSchools(inputdir = getwd(), outputdir = getwd(), quiet = FALSE)
```

Arguments

`inputdir`	a string indicating the path where `ILSAmerge` files are stored.
`outputdir`	a string indicating where the combined data will be saved.
`quiet`	a logical value indicating if progress status should be shown. Default is `FALSE`.

Value

Saves combined student data and teacher data with added school data.

Examples

```
# Path where raw 'SPSS' files are
input <- system.file("extdata/timssadv", package = "ILSAmerge")

# Path where merged files will be saved
dir.create(file.path(tempdir(),"addSchools"))
output <- file.path(tempdir(),"addSchools")

# Merging 'TIMSS' Advanced 1995, as .rds file
ILSAmerge(inputdir = input, outputdir = output, filetype = "rds", quiet = FALSE)

# Check file names
```

```
list.files(output,pattern = ".rds")

# Add school data
addSchools(inputdir = output, outputdir = output)

# Check file names
list.files(output,pattern = ".rds")
```

`asthistibble` *Copy tibble attributes*

Description

Converts a data frame into a tibble copying all attributes.

Usage

```
asthistibble(tibble, x)
```

Arguments

`tibble`	a tibble object.
`x`	a data frame with the same columns of `tibble`.

Value

A tibble.

Examples

```
# tibble generated by haven
input <- system.file("extdata/reds", package = "ILSAmerge")
tib <- do.call(rbind,justload(inputdir = input,population = "BCGV1"))

# Remove all tibble attributes
x <- tib
class(x) <- "data.frame"
for(i in 1:ncol(x)){
  attributes(x[,1]) <- NULL
}

# Mix variables
set.seed(1919)
x <- x[,sample(ncol(x))]
head(x)[,1:10]
tib

asthistibble(tibble = tib, x = x)
```

`availableILSA` *Check available ILSA data*

Description

Checks which 'SPSS' data from different International Large-Scale Assessments (ILSA). are available.

Usage

```
availableILSA(
  print = TRUE,
  FOR = c("download", "combine.students", "add.schools", "ILSAready")
)
```

Arguments

`print`	a logical value indicating if results should be printed or not.
`FOR`	a string indicating the availability of ILSA data for different purposes. Valid strings are `"download"`, `"combine.students"`, `add.schools`, and `"ILSAready"`.

Value

A list with the names of the ILSA and the available years.

Examples

```
availableILSA(print = TRUE)
```

`combineStudents` *Combine student data*

Description

Combines achievement and background student data merged by `ILSAmerge`. To see which ILSA are available for combining use `availableILSA`.

Usage

```
combineStudents(inputdir = getwd(), outputdir = getwd(), quiet = FALSE)
```

Arguments

`inputdir`	a string indicating the path where `ILSAmerge` files are stored.
`outputdir`	a string indicating where the combined data will be saved.
`quiet`	a logical value indicating if progress status should be shown. Default is `FALSE`.

Value

Saves combined student data produced by ILSAmerge.

Examples

```
# Path where raw 'SPSS' files are
input <- system.file("extdata/timssadv", package = "ILSAmerge")

# Path where merged files will be saved
dir.create(file.path(tempdir(),"combineStudents"))
output <- file.path(tempdir(),"combineStudents")

# Merging 'TIMSS' Advanced 1995, as .rds file
ILSAmerge(inputdir = input, outputdir = output, filetype = "rds", quiet = FALSE)

# Rename files
ILSArename(output)

# Check file names
list.files(output,pattern = ".rds")

# Combine student files
combineStudents(inputdir = output, outputdir = output)

# Check file names
list.files(output,pattern = ".rds")
```

`get.atr` *Retrieve tibble attributes*

Description

Retrieves attributes from a tibble object created by haven.

Usage

```
get.atr(tibble, which, NULLasNA = TRUE, exact = FALSE)

get.nas(tibble, aslist = TRUE)

get.varlab(tibble)
```

Arguments

`tibble`	a tibble object.
`which`	a non-empty character string specifying which attribute is to be accessed.

`NULLasNA`	a logical value indicating if `NULL` attributes should be listed as NA. Default is `TRUE`.
`exact`	logical: should `which` be matched exactly?
`aslist`	a logical value indicating if output should be a list.

Value

A list or a data frame.

Examples

```
# tibble generated by haven
input <- system.file("extdata/reds", package = "ILSAmerge")
x <- do.call(rbind,justload(inputdir = input,population = "BCGV1"))
x

# Get an attribute
get.atr(tibble = x, which = "label")[1:3]
get.atr(tibble = x, which = "labels")[1:3]
get.atr(tibble = x, which = "format.spss")[1:3]

# Get NAs
get.nas(tibble = x,aslist = TRUE)[10:20]
get.nas(tibble = x,aslist = FALSE)[10:20,]

# Get variable labels
get.varlab(tibble = x)[10:20,]
```

`ILSAdownload` *Download ILSA data*

Description

Downloads 'SPSS' data from different International Large-Scale Assessments (ILSA). This functions supports the following ILSA: 'TIMSS', 'TIMSS Advanced', 'PIRLS', 'ICCS', 'ICILS', 'LANA', 'CIVED', 'REDS', 'RLII', and 'SITES.' Depending on the study, you will need to decide which data to download, and read and accept its terms and conditions to proceed with the download. Data is retrieved from `https://www.iea.nl`.

Usage

```
ILSAdownload(
  study,
  year,
  outputdir = getwd(),
  unzip = FALSE,
  maxtime = 999,
```

```
  quiet = FALSE,
  agreeLicense = FALSE
)
```

Arguments

`study`	a string indicating the name of the study. For available studies check the description of this function.
`year`	a numeric value indicating the year of the study.
`outputdir`	the directory where data will be downloaded.
`unzip`	a logical value indicating if files should be unzipped. Default is `FALSE`.
`maxtime`	a numeric value indicating the maximum time allowed for downloading a file. Default is `999`.
`quiet`	a logical value indicating if progress status should be shown. Default is `FALSE`.
`agreeLicense`	a logical value indicating if you agree with the Disclaimer and License Agreement file from `https://www.iea.nl`. If `FALSE`, you will be prompted to agree with it or else data will not be downloaded. Default is `FALSE`.

Value

Saves 'SPSS' ILSA data locally.

Examples

```
# For example, to download 'RLII' 1991 data:

# Path where files will be saved
dir.create(file.path(tempdir(),"ILSAdownload"))
output <- file.path(tempdir(),"ILSAdownload")


# Downloading 'RLII' 1991 and unzipping files
ILSAdownload(study = "RLII", year = 1991, outputdir = output, unzip = TRUE, agreeLicense = TRUE)
```

`ILSAfile.info` *ILSA data files information*

Description

Aggregates International Large-Scale Assessments (ILSA) data files information by population.

Usage

```
ILSAfile.info(inputdir = getwd())
```

Arguments

`inputdir`	a string indicating the path where ILSA 'SPSS' files are stored.

Value

A data frame with the number of files and MBs per population.

Examples

```
# Path where raw 'SPSS' files are
input <- system.file("extdata/reds", package = "ILSAmerge")

# Get file information
ILSAfile.info(inputdir = input)
```

`ILSAmerge` *Merge ILSA data*

Description

Merges 'SPSS' data from different International Large-Scale Assessments (ILSA). This function has been tested to behave correctly for: 'TIMSS', 'TIMSS Advanced', 'PIRLS', 'ICCS', 'ICILS', 'LANA', CIVED', 'REDS', 'RLII', and 'SITES' (2006).

Usage

```
ILSAmerge(
  inputdir = getwd(),
  outputdir = getwd(),
  population = NULL,
  filetype = c("rds", "zsav", "sav"),
  MBlimit = NULL,
  MBlistlimit = 200,
  SPSSlimit = 50,
  quiet = FALSE
)
```

Arguments

`inputdir`	a string indicating the path where ILSA 'SPSS' files are stored.
`outputdir`	the directory where the merged data will be saved.
`population`	a character vector indicating which files should be merged. If `NULL` (the default), all files will be merged. For more information on available populations, run `ILSAfile.info()` first.
`filetype`	a string indicating the type of file to be saved, it can be `"rds"`, `"zsav"`, or `"sav"`.

`MBlimit`	a numerical value indicating the allowed limit of the combined storage of the files of one type (see `ILSAfile.info()`). For type files that go over the limit, files will not be merged in R, but an 'SPSS' syntax will be produced via `spss.syntax()`. If set to `NULL`, no limit will be used and all files will be merged within R. If speed is a problem, we recommend that this number should not be over `200` and merge the rest in 'SPSS'. Beware that some ILSA will have files with different columns and this could cause some 'SPSS' syntaxes to fail. If this happens, merge through `R`.
`MBlistlimit`	a numerical value indicating the allowed limit of the combined storage of the files of one type for merging through a list. Values over the limit will be merged through a matrix, which will be slower but uses less memory. Default is `200`.
`SPSSlimit`	a numerical value indicating the limit of files per command of 'SPSS', typically `50`.
`quiet`	a logical value indicating if progress status should be shown. Default is `FALSE`.

Details

For files merged within R it will also add country information where needed. Country information will be retrieved from 'GitHub' if possible. If not, it will use the package internal data.

Value

Saves merged ILSA data or `.sps` syntax for merging ILSA data.

Examples

```
# Path where raw 'SPSS' files are
input <- system.file("extdata/reds", package = "ILSAmerge")

# Path where merged files will be saved
dir.create(file.path(tempdir(),"ILSAmerge"))
output <- file.path(tempdir(),"ILSAmerge")

# Merging 'REDS' 2021, as .rds file
ILSAmerge(inputdir = input, outputdir = output, filetype = "rds", quiet = FALSE)
```

`ILSAready` *Download and prepare ILSA data*

Description

Downloads ILSA data, merges it, combines students and adds school information. This function is a wrapper for `ILSAdownload`, `ILSAmerge`, `ILSArename`, `combineStudents`, and `addSchools`. To see which ILSA are available for this function use `availableILSA`. If data is already downloaded you can use `ILSAreadylocal`.

Usage

```
ILSAready(
  study,
  year,
  outputdir = getwd(),
  filetype = c("rds", "zsav", "sav"),
  MBlistlimit = 200,
  quiet = FALSE,
  agreeLicense = FALSE,
  maxtime = 999
)

ILSAreadylocal(
  inputdir = getwd(),
  outputdir = getwd(),
  filetype = c("rds", "zsav", "sav"),
  quiet = FALSE,
  MBlistlimit = 200
)
```

Arguments

`study`	a string indicating the name of the study. For available studies check the description of this function.
`year`	a numeric value indicating the year of the study.
`outputdir`	the directory where the merged data will be saved.
`filetype`	a string indicating the type of file to be saved, it can be `"rds"`, `"zsav"`, or `"sav"`.
`MBlistlimit`	a numerical value indicating the allowed limit of the combined storage of the files of one type for merging through a list. Values over the limit will be merged through a matrix, which will be slower but uses less memory. Default is `200`.
`quiet`	a logical value indicating if progress status should be shown. Default is `FALSE`.
`agreeLicense`	a logical value indicating if you agree with the Disclaimer and License Agreement file from `https://www.iea.nl`. If `FALSE`, you will be prompted to agree with it or else data will not be downloaded. Default is `FALSE`.
`maxtime`	a numeric value indicating the maximum time allowed for downloading a file. Default is `999`.
`inputdir`	a string indicating the path where ILSA 'SPSS' files are stored.

Value

Saves merged and renamed ILSA data.

Examples

```
dir.create(file.path(tempdir(),"timssadv"),showWarnings = FALSE)
output <- file.path(tempdir(),"timssadv")
```

```
input <- system.file("extdata/timssadv", package = "ILSAmerge")

ILSAreadylocal(inputdir = input, outputdir = output, filetype = "zsav")
```

`ILSArename` *Rename* `ILSAmerge` *files*

Description

Renames files produced by `ILSAmerge` from name codes to comprehensible names including the study name, year and respondent. This function has been tested to behave correctly for: 'TIMSS', 'TIMSS Advanced', 'PIRLS', 'ICCS', 'ICILS', 'LANA', 'CIVED', 'REDS', 'RLII', and 'SITES' (2006).

Usage

```
ILSArename(
  inputdir = getwd(),
  codeTOname = TRUE,
  overwrite = TRUE,
  quiet = FALSE
)
```

Arguments

`inputdir`	a string indicating the path where ILSA 'SPSS' files are stored.
`codeTOname`	a logical value indicating if files should be renamed from codes to names (`TRUE`) or from names to codes (`FALSE`). Default is `TRUE`.
`overwrite`	a logical value indicating if files should be overwritten. If `FALSE`, files will be copied with the new names. Default is `TRUE`.
`quiet`	a logical value indicating if progress status should be shown. Default is `FALSE`.

Value

Renames or copies files produced by `ILSAmerge`.

Examples

```
# Merge files
dir.create(file.path(tempdir(),"REDS2021"),showWarnings = FALSE)
ILSAmerge(inputdir = system.file("extdata/reds", package = "ILSAmerge"),
outputdir = file.path(tempdir(),"REDS2021"))

# Show files with raw names
list.files(file.path(tempdir(),"REDS2021"))
```

```
# Rename files
ILSArename(inputdir = file.path(tempdir(),"REDS2021"))

# Show files new names
list.files(file.path(tempdir(),"REDS2021"))
```

`justload` *Loading ILSA data into a list*

Description

Load ’SPSS’ data from different International Large-Scale Assessments (ILSA), including: ’TIMSS’, ’TIMSS Advanced’, ’PIRLS’, ’ICCS’, ’ICILS’, ’LANA’, ’CIVED’, ’REDS’, ’RLII’, and ’SITES’ (2006) into a list.

Usage

```
justload(
  inputdir = getwd(),
  population,
  justattributes = FALSE,
  addcountries = FALSE
)
```

Arguments

`inputdir`	a string indicating the path where ILSA ’SPSS’ files are stored.
`population`	a character value indicating which files should be loaded. For more information on available populations, run `ILSAfile.info()` first.
`justattributes`	a logical value indicating if 0 rows should be loaded. This can be used when we just need to check column attributes. Default is `FALSE`.
`addcountries`	a logical value indicating if country information should be added to the elements of the list. This means adding the variable `CNTRY` where needed and adding labels for `IDCNTRY` where needed. If `FALSE` (the default), data will be loaded as is. Country information will be retrieved from ’GitHub’ if possible. If not, it will use the package internal data.

Value

A list of tibbles.

Examples

```
# Path where raw 'SPSS' files are
input <- system.file("extdata/reds", package = "ILSAmerge")

# Load only attributes
emptylist <- justload(inputdir = input, population = "BCGV1", justattributes = TRUE)

# Load complete data
fulllist <- justload(inputdir = input, population = "BCGV1", justattributes = FALSE)

# Load complete data and add country labels
withcou <- justload(inputdir = input, population = "BCGV1", addcountries = TRUE)
```

`mistoNAs` *Missing to NAs*

Description

Converts values labelled as missings to `NA`.

Usage

```
mistoNAs(tibble)
```

Arguments

`tibble`	a tibble object.

Value

A tibble.

Examples

```
# tibble generated by haven
input <- system.file("extdata/reds", package = "ILSAmerge")
tib <- do.call(rbind,justload(inputdir = input,population = "BCGV1"))

tib

mistoNAs(tib)
```

`readILSA` *Read ILSA data*

Description

Reads files created with ILSAmerge().

Usage

```
readILSA(file, mistoNAs = FALSE, untibble = FALSE)
```

Arguments

`file`	a path to an '.rds', '.sav', or '.zsav' file.
`mistoNAs`	a logical value indicating if missing values should be converted into NAs. Default is `FALSE`.
`untibble`	a logical value indicating if data should be converted into a plain data frame with no column attributes.

Value

A tibble or a data frame.

Examples

```
# Path where raw 'SPSS' files are
input <- system.file("extdata/reds", package = "ILSAmerge")

# Path where merged files will be saved
unlink(file.path(tempdir(),"ILSAmerge"),recursive = TRUE)
dir.create(file.path(tempdir(),"ILSAmerge"))
output <- file.path(tempdir(),"ILSAmerge")

# Merging 'REDS' 2021, as .rds file
ILSAmerge(inputdir = input, outputdir = output, filetype = "rds", quiet = FALSE)

# Read student file
readILSA(file = file.path(output,"BSGV1.rds"))
```

`spss.syntax` *'SPSS' merge syntax*

Description

Produces and saves an 'SPSS' merge syntax given a list of files.

Usage

```
spss.syntax(filelist, name, outputdir = getwd(), zsav = TRUE, SPSSlimit = 50)
```

Arguments

`filelist`	a character vector with the list of files to be merged.
`name`	a string with the name of the merged file (without any extension).
`outputdir`	the directory where the `.sps` file and the merged file will be saved.
`zsav`	a logical value indicating if the the merged file should be compressed with zsav. Default is `TRUE`.
`SPSSlimit`	a numerical value indicating the limit of files per command of 'SPSS', typically `50`.

Value

Saves an `.sps` file with the 'SPSS' syntax for merging the desired files.

Examples

```
# Path where raw 'SPSS' files are
input <- system.file("extdata/reds", package = "ILSAmerge")

# Path where merged files will be saved
dir.create(file.path(tempdir(),"spsssyntax"))
output <- file.path(tempdir(),"spsssyntax")

# List of BCGV1 files to be merged
files <- list.files(path = input, pattern = "BCG.+V1|bcg.+v1")

# Create 'SPSS' syntax
spss.syntax(filelist = files, name = "BCGV1", outputdir = output, zsav = TRUE)
```

`untibble` *Untibble*

Description

Converts a tibble into a plain data frame with no column attributes.

Usage

```
untibble(tibble, mistoNAs = FALSE)
```

Arguments

`tibble`	a tibble object or a list of tibbles.
`mistoNAs`	a logical value indicating if missing values should be converted into NAs. Default is `FALSE`.

Value

A tibble.

Examples

```
# Path where raw 'SPSS' files are
input <- system.file("extdata/reds", package = "ILSAmerge")

# Load complete data
fulllist <- justload(inputdir = input, population = "BCGV1", justattributes = FALSE)

# Untibble first element
unt1 <- untibble(fulllist[[1]])

# Untibble all list
unt2 <- untibble(fulllist)
```

`whichcol` *Which column*

Description

Retrieves column names and labels that matches a general expression via grepl.

Usage

```
whichcol(pattern, tibble, label = TRUE, ignore.case = TRUE, ...)
```

Arguments

`pattern`	character string containing a regular expression (or character string for `fixed = TRUE`) to be matched in the given character vector. Coerced by `as.character` to a character string if possible. If a character vector of length 2 or more is supplied, the first element is used with a warning. Missing values are allowed except for `regexpr`, `gregexpr` and `regexec`.
`tibble`	a tibble object.
`label`	a logical value indicating if pattern should be searched in variable label, instead of variable name. Default is `TRUE`.
`ignore.case`	logical. if `FALSE`, the pattern matching is *case sensitive* and if `TRUE`, case is ignored during matching.
`...`	Arguments passed on to `base::grepl` `x,text` a character vector where matches are sought, or an object which can be coerced by `as.character` to a character vector. Long vectors are supported. `perl` logical. Should Perl-compatible regexps be used? `value` logical. If `FALSE`, a vector containing the (`integer`) indices of the matches determined by `grep` is returned, and if `TRUE`, a vector containing the matching elements themselves is returned. `fixed` logical. If `TRUE`, `pattern` is a string to be matched as is. Overrides all conflicting arguments. `useBytes` logical. If `TRUE` the matching is done byte-by-byte rather than character-by-character. See ‘Details’. `invert` logical. If `TRUE` return indices or values for elements that do *not* match. `replacement` a replacement for the matched pattern in `sub` and `gsub`. Coerced to character if possible. For `fixed = FALSE` this can include backreferences `"\1"` to `"\9"` to parenthesized subexpressions of `pattern`. For `perl = TRUE` only, it can also contain `"\U"` or `"\L"` to convert the rest of the replacement to upper or lower case and `"\E"` to end case conversion. If a character vector of length 2 or more is supplied, the first element is used with a warning. If `NA`, all elements in the result corresponding to matches will be set to `NA`.

Value

A data frame.

Examples

```
# tibble generated by haven
input <- system.file("extdata/reds", package = "ILSAmerge")
x <- do.call(rbind,justload(inputdir = input,population = "BCGV1"))
x

whichcol("weight",x)
```

Index

∗ **tibblemanagement**
    asthistibble, 3
    get.atr, 5
    mistoNAs, 13
    untibble, 16
    whichcol, 16

addSchools, 2, *9*
as.character, *17*
asthistibble, 3
availableILSA, *2*, *4*, 4, *9*

base::grepl, *17*

combineStudents, *2*, 4, *9*

get.atr, 5
get.nas *(*get.atr*)*, 5
get.varlab *(*get.atr*)*, 5
grepl, *16*

ILSAdownload, 6, *9*
ILSAfile.info, 7
ILSAmerge, *2*, *4*, *5*, 8, *9*, *11*
ILSAready, 9
ILSAreadylocal, *9*
ILSAreadylocal *(*ILSAready*)*, 9
ILSArename, *9*, 11

justload, 12

Long vectors, *17*

mistoNAs, 13

readILSA, 14
regular expression, *17*

spss.syntax, 15

untibble, 16

whichcol, 16