

Package ‘Rilostat’

October 12, 2022

Type Package

Title ILO Open Data via Ilostat Bulk Download Facility or SDMX Web Service

Date 2021-07-29

Version 1.1.8

Encoding UTF-8

Maintainer David Bescond <bescond@ilo.org>

Description Tools to download data from the ilostat database <<https://ilostat.ilo.org>> together with search and manipulation utilities.

License BSD_2_clause + file LICENSE

Depends R (>= 3.5.0)

Imports plyr, dplyr (>= 1.0.4), stringr (>= 1.4.0), readr (>= 1.4.0), tibble (>= 3.0.6), tidyr, haven, xml2, data.table, RCurl, DT

Suggests knitr, rmarkdown, devtools, plotly, ggplot2 (>= 3.3.3), shiny, curl

LazyData false

URL <https://ilostat.github.io/Rilostat/>

BugReports <https://github.com/ilostat/Rilostat/issues>

Repository CRAN

RoxygenNote 7.1.1

NeedsCompilation no

Author ILO Department of Statistics [cph, fnd],
David Bescond [aut, cre] (<<https://orcid.org/0000-0002-1642-0304>>),
Mabelin Villarreal-Fuentes [ctb]
(<<https://orcid.org/0000-0001-5644-5760>>)

Date/Publication 2021-07-29 06:40:02 UTC

R topics documented:

clean_ilstat_cache	2
distribution_ilstat	3
get_ilstat	4
get_ilstat_dic	8
get_ilstat_toc	9
label_ilstat	11
Rilostat	12
runExplorer	13
sdmx_ilstat	14

Index	17
--------------	-----------

clean_ilstat_cache	<i>Clean Iloststat Cache</i>
--------------------	------------------------------

Description

Deletes all cache files from the your ilostat cache directory. See [get_ilstat](#) for more on cache.

Usage

```
clean_ilstat_cache(
  cache_dir = getOption("ilostat_cache_dir", file.path(tempdir(), "ilostat")),
  cache_update = getOption("ilostat_cache_update", FALSE),
  quiet = getOption("ilostat_quiet", FALSE)
)
```

Arguments

cache_dir	A character, path to a cache directory. The directory has to exist. The NULL (default) uses and creates 'ilostat' directory in the temporary directory from tempdir . The directory can also be set with option <code>ilostat_cache_dir</code> ,
cache_update	a logical whether to delete only out of date cache files. Useful when <code>cache_dir</code> is set as keep only update datasets. Can be set also with <code>options(ilostat_update = TRUE)</code> . Default is FALSE.
quiet	a logical, if TRUE , don't return message from processing, FALSE (default). Can be set also with <code>options(ilostat_quiet = TRUE)</code> ,

Author(s)

David Bescond <bescond@ilo.org>

References

See citation("Rilostat") ilostat bulk download facility user guidelines https://www.ilo.org/ilostat-files/Documents/ILOSTAT_BulkDownload_Guidelines.pdf

Examples

```
## Not run:  
clean_iloostat_cache()  
  
## End(Not run)
```

distribution_iloostat *Switch iloostat to distribution*

Description

Gets distribution for iloostat number of persons only.

Usage

```
distribution_iloostat(x, var)
```

Arguments

x	dataset to transform into distribution.
var	String variable name use for the distribution default "no", could be "sex", "classif1", "classif2".

Details

this function use the max of the corresponding grouping so it is important to not filter any subset of the corresponding variable selected for the distribution at this level, ie. if you remove SEX_T, the distribution by sex will only have SEX_F or SEX_M / $\max(\text{SEX_M}, \text{SEX_F}) * 100$, which is no longer a distribution.

In addition, distribution is only applicable for indicators with Number of persons (usually in thousands), So plse do not distribute ratios, earnings, hours of works, CPI, GDP etc ... no warning will prevent for that if doubts use distribution from get_iloostat() instead of, warnings will help you.

Value

a data_frame. obs_status will no longer be a number of persons but a percentage.

Author(s)

David Bescond <bescond@ilo.org>

References

See citation("Rilostat") iloostat bulk download facility user guidelines https://www.ilo.org/iloostat-files/Documents/ILOSTAT_BulkDownload_Guidelines.pdf

Examples

```
## Not run:
dat <- get_ilstat("EMP_TEMP_SEX_STE_GEO_NB_A", cache = FALSE)
dat_dist <- distribution_ilstat(dat, "classif1")
head(dat_dist)
clean_ilstat_cache()

## End(Not run)
```

get_ilstat

Read Ilost Data

Description

Download datasets from ilostat <https://ilstat.ilo.org> via bulk download facility <https://ilstat.ilo.org/data/bulk/>.

Usage

```
get_ilstat(
  id,
  segment = getOption("ilstat_segment", "indicator"),
  type = getOption("ilstat_type", "code"),
  lang = getOption("ilstat_lang", "en"),
  time_format = getOption("ilstat_time_format", "raw"),
  filters = getOption("ilstat_filter", "none"),
  fixed = getOption("ilstat_fixed", TRUE),
  detail = getOption("ilstat_detail", "full"),
  cache = getOption("ilstat_cache", TRUE),
  cache_update = getOption("ilstat_cache_update", TRUE),
  cache_dir = getOption("ilstat_cache_dir", NULL),
  cache_format = getOption("ilstat_cache_format", "rds"),
  back = getOption("ilstat_back", TRUE),
  distribution = getOption("ilstat_distribution", "no"),
  cmd = getOption("ilstat_cmd", "none"),
  quiet = getOption("ilstat_quiet", FALSE)
)
```

Arguments

id	A code name for the dataset of interest. See get_ilstat_toc or details for how to get code.
segment	A character, way to get datasets by: "indicator" (default) or "ref_area". Can be set also with options(ilstat_segment = 'ref_area'),
type	a character, type of variables, "code" (default), "label" or "both". Can be set also with options(ilstat_type = 'both'),

lang	a character, code for language. Available are "en" (default), "fr" and "es". Can be set also with options(ilstat_lang = 'fr'),
time_format	a string giving a type of the conversion of the time column from the ilostat format. "raw" (default) does not do conversion and return time as character (ie. '2017', '2017Q1', '2017M01'). A "date" converted to a Date with a first date of the period. A "date_last" converted to a Date with a last date of the period and "num" converted to a numeric. Can be set also with options(ilstat_time_format = 'date'),
filters	a list; "none" (default) to get a whole dataset or a named list of filters to get just part of the table. Names of list objects are ilostat variable codes and values are vectors of observation codes. filters detect on variables, so could be partial, ie. list(sex = 'T') is enough but equivalent to list(sex = 'SEX_T'). Additional options: <ul style="list-style-type: none"> • timefrom : starting year of the return dataset, • timeto : ending year of the return dataset.
fixed	a logical, if TRUE (default), filters arguments pattern is a string to be matched as is, Change to FALSE if more complex regex matching is needed.
detail	a character, 'full' (default), whether 'serieskeyonly', return only key, no data, no notes or dataonly return key and data, no notes or 'bestsourceonly' return best source only at the series key levels, see details. Can be set also with options(ilstat_detail = 'serieskeyonly'),
cache	a logical whether to do caching. Default is TRUE. Affects only queries from the ilostat bulk download facility. Can be set also with options(ilstat_cache = FALSE),
cache_update	a logical whether to update cache. Check cache update with last.update attribute store on the cache file name and the one from the table of contents. Can be set also with options(ilstat_cache_update = FALSE). Default is TRUE,
cache_dir	a path to a cache directory. The directory has to exist. The NULL (default) uses and creates 'ilstat' directory in the temporary directory from tempdir . The directory can also be set with option ilostat_cache_dir,
cache_format	a character, format to store on the cache "rds" (default), but also "csv", "dta", "sav", "sas7bdat". useful for getting ilostat dataset directly on the cache_dir without R. Can be set also with options(ilstat_cache_format = 'dta'),
back	a logical, TRUE return dataframe on R or not FALSE, useful for just saving file in specific cache_format,
distribution	a character, transform number of persons into distribution by sex or classif1 or classif2, "sex", "classif1" or "classif2". Can be set also with options(ilstat_distribution = 'classif1'), default is 'no',
cmd	a character, R expression use for manipulate internal data frame dat object applied to each datasets retrieved If use, cache is set to FALSE. see examples.
quiet	a logical, if TRUE , don't return message from processing, FALSE (default). Can be set also with options(ilstat_quiet = TRUE),

Value

a tibble. One column for each dimension in the data and the values column for numerical values, as well as the metadata columns. The time column for a time dimension.

others

Data sets are downloaded from the ilostat bulk download facility. If only the table id is given, the whole table is downloaded from the bulk download facility.

The bulk download facility is the fastest method to download whole datasets. It is also often the only way as the sdmx API has limitation of maximum 300 000 records at the same time and whole datasets usually exceeds that.

By default datasets from the bulk download facility are cached as they are often rather large.

Cache files are stored in a temporary directory by default or in a named directory if `cache_dir` or option `ilstat_cache_dir` is defined. The cache can be emptied with `clean_ilstat_cache`.

The id, a code, for the dataset can be searched with the `get_ilstat_toc` or from the [bulk download facility](<https://ilstat.ilo.org/data/bulk/>).

Author(s)

David Bescond <bescond@ilo.org>

References

See citation("Rilostat") ilostat bulk download facility user guidelines <https://ilstat.ilo.org/data/bulk/>

See Also

`get_ilstat_toc`, `label_ilstat`

Examples

```
## Not run:
##### get simple dataset
dat <- get_ilstat("UNE_2UNE_SEX_AGE_NB_A")
head(dat)
dat <- get_ilstat("NZL_Q", segment = "ref_area")
head(dat)

dir.create(file.path(tempdir(), "r_cache"))
dat <- get_ilstat("UNE_2UNE_SEX_AGE_NB_A",
                cache_dir = file.path(tempdir(), "r_cache"))
head(dat)

clean_ilstat_cache(cache_dir = file.path(tempdir(), "r_cache"))

options(ilstat_update = TRUE)
dat <- get_ilstat("UNE_2UNE_SEX_AGE_NB_A")
head(dat)
```

```
options(ilstat_update = FALSE)
options(ilstat_cache_dir = file.path(tempdir(), "r_cache"))
dat <- get_ilstat("UNE_2UNE_SEX_AGE_NB_A")

clean_ilstat_cache()

##### get multiple datasets
dat <- get_ilstat(c("CPI_ACPI_COI_RT_M", 'CPI_ACPI_COI_RT_Q'), cache = FALSE)
head(dat)
toc <- get_ilstat_toc(search = 'CPI_')
head(toc)
dat <- get_ilstat(toc, cache = FALSE) #id as a tibble

##### get datasets with filters
dat <- get_ilstat(id = c("UNE_2UNE_SEX_AGE_NB_A", 'EMP_2EMP_SEX_AGE_NB_A'),
  filters = list( ref_area = "FRA",
    classif1 = "AGE_YTHADULT_YGE15",
    time = "2016",
    sex = c("T", 'SEX_F')), quiet = TRUE)
head(dat)
clean_ilstat_cache()

##### store in other format
dir.create(file.path(tempdir(), "ilstat"))

dat <- get_ilstat("UNE_2UNE_SEX_AGE_NB_A",
  cache_dir = file.path(tempdir(), "r_cache"), cache_format = 'csv')
dat <- get_ilstat("UNE_2UNE_SEX_AGE_NB_A",
  cache_dir = file.path(tempdir(), "r_cache"), cache_format = 'dta')

##### use distribution

dat <- get_ilstat("EMP_TEMP_SEX_STE_GEO_NB_A", distribution = 'classif1')

# obs_status and note_classif could not be pivot as store at value level

select(dat, -obs_status, -note_classif) %>% spread(classif1, obs_value)

# use label

get_ilstat(dat, "EMP_TEMP_SEX_STE_GEO_NB_A", distribution = 'classif1', type = 'label')

select(dat, -obs_status.label, -note_classif.label) %>% spread(classif1.label, obs_value)

clean_ilstat_cache()

##### advanced manipulation

dat <- get_ilstat("UNE_2UNE_SEX_AGE_NB_A", cmd = "dat %>% count(ref_area)", quiet = TRUE)
label_ilstat(dat, code = 'ref_area')

clean_ilstat_cache()
```

```
## End(Not run)
```

get_ilstat_dic	<i>Read Ilostat Dictionary</i>
----------------	--------------------------------

Description

Downloads one ilostat dictionary from ilostat <https://ilostat.ilo.org> via bulk download facility <https://ilostat.ilo.org/data/bulk/>.

Usage

```
get_ilstat_dic(dic, lang = getOption("ilostat_lang", "en"))
```

Arguments

dic	A character, dictionary for the variable to be downloaded,
lang	a character, code for language. Available are "en" (default), "fr" and "es". Can be set also with options(ilostat_lang = 'fr'),

Details

For a given coded variable from ilostat <https://ilostat.ilo.org/>. The dictionaries link codes with human-readable labels. To translate codes to labels, use [label_ilstat](#).

Value

tibble with two columns: code names and full names.

Author(s)

David Bescond <bescond@ilo.org>

References

See citation("Rilostat") ilostat bulk download facility user guidelines https://www.ilo.org/ilostat-files/WEB_bulk_download/ILOSTAT_BulkDownload_Guidelines.pdf ilostat bulk download facility main page <https://ilostat.ilo.org/data/bulk/>

See Also

[label_ilstat](#), [get_ilstat](#).

Examples

```
## Not run:
tmp <- get_ilstat_dic("indicator")
head(tmp)
tmp <- get_ilstat_dic("classif1", lang = "fr")
head(tmp)

## End(Not run)
```

get_ilstat_toc	<i>Read Ilostat Table of Contents</i>
----------------	---------------------------------------

Description

Download one table of contents from ilostat <https://ilostat.ilo.org> via bulk download facility <https://ilostat.ilo.org/data/bulk/>.

Usage

```
get_ilstat_toc(
  segment = getOption("ilstat_segment", "indicator"),
  lang = getOption("ilstat_lang", "en"),
  search = getOption("ilstat_search", "none"),
  filters = getOption("ilstat_filter", "none"),
  fixed = getOption("ilstat_fixed", TRUE)
)
```

Arguments

segment	A character, way to get datasets by: "indicator" (default) or "ref_area", Can be set also with options(ilstat_segment = 'ref_area'),
lang	a character, code for language. Available are "en" (default), "fr" and "es". Can be set also with options(ilstat_lang = 'fr'),
search	a character vector, "none" (default), datasets with this pattern in the description will be returned, characters vector will be use as AND, Character with ' ' as OR, see example, options(ilstat_time_format = 'date'),
filters	a list; "none" (default) to get a whole toc or a named list of filters to get just part of the table. Names of list objects are ilostat toc variable codes and values are vectors of observation codes. filters detect on variables.
fixed	a logical, if TRUE (default), pattern is a string to be matched as is, Change to FALSE if more complex regex matching is needed.

Details

The TOC in English by indicator is downloaded from https://www.ilo.org/iloostat-files/WEB_bulk_download/indicator/table_of_contents_en.csv. The values in column 'id' should be used to download a selected dataset.

The TOC in English by ref_area is downloaded from https://www.ilo.org/iloostat-files/WEB_bulk_download/ref_area/table_of_contents_en.csv. The values in column 'id' should be used to download a selected dataset.

Value

A tibble with ten columns depending of the segment: indicator or ref_area

- id : The codename of dataset of theme, will be used by the get_iloostat and get_iloostat_raw functions,
- indicator or ref_area : The indicator or ref_area code of dataset,
- indicator.label or ref_area.label : The indicator or ref_area name of dataset,
- freq : The frequency code of dataset,
- freq.label : Is freq name of dataset,
- size : Size of the csv.gz files,
- data.start : First time period of the dataset,
- data.end : Last time period of the dataset,
- last.update : Last update of the dataset,
- ... : Others relevant information

Author(s)

David Bescond <bescond@ilo.org>

References

See citation("Rilostat") iloostat bulk download facility user guidelines https://www.ilo.org/iloostat-files/WEB_bulk_download/ILOSTAT_BulkDownload_Guidelines.pdf

See Also

[get_iloostat](#).

Examples

```
## Not run:
## default segment by indicator, default lang English
toc <- get_iloostat_toc()
head(toc)
toc <- get_iloostat_toc(segment = 'ref_area', lang = 'fr')
head(toc)
##
## search on toc
```

```

toc <- get_ilstat_toc(search = 'education')
head(toc)
toc <- get_ilstat_toc(lang = 'fr', search = 'Ã©ducation')
head(toc)
toc <- get_ilstat_toc(segment = 'ref_area', lang = 'fr', search = 'Albanie')
toc
toc <- get_ilstat_toc(segment = 'ref_area', lang = 'es', search = 'Trimestral')
head(toc)
##
## search multi on toc
toc <- get_ilstat_toc(segment = 'ref_area', lang = 'fr',
  search = 'Albanie|France', fixed = FALSE)
head(toc)
toc <- get_ilstat_toc(search = 'youth|adult', fixed = FALSE)
head(toc)
toc <- get_ilstat_toc(search = c('youth','adult'), fixed = FALSE)
head(toc)
##
## End(Not run)

```

label_ilstat

Switch Iloststat codes and labels

Description

Gets definitions/labels for ilostat codes from ilostat dictionaries.

Usage

```

label_ilstat(
  x,
  dic = NULL,
  code = NULL,
  lang = getOption("ilostat_lang", "en")
)

```

Arguments

x	A character or a factor vector or a data_frame to labelled.
dic	A string (vector) naming ilostat dictionary or dictionaries. If NULL (default) dictionary names are taken from column names of the data_frame. A character or a factor vector or a data_frame to labelled,
code	a vector of names of the column for which code columns should be retained. Set to "all", keep all the code.
lang	a character, code for language. Available are "en" (default), "fr" and "es". Can be set also with options(ilostat_lang = 'fr'),

Details

A character or a factor vector of codes returns a corresponding vector of definitions. `label_ilstat` labels also `data_frame`s from `get_ilstat`. For vectors a dictionary "time" and "values" columns are returned as they were, so you can supply `data_frame` from `get_ilstat` and get `data_frame` with definitions instead of codes.

Value

a vector or a `data_frame`. The suffix ".label" is added to code column names.

Author(s)

David Bescond <bescond@ilo.org>

References

See citation("Rilostat") `ilstat` bulk download facility user guidelines https://www.ilo.org/ilstat-files/Documents/ILOSTAT_BulkDownload_Guidelines.pdf

Examples

```
## Not run:
dat <- get_ilstat("UNE_2UNE_SEX_AGE_NB_A", cache = FALSE)
dat_lab <- label_ilstat(dat)
head(dat_lab)
clean_ilstat_cache()

## End(Not run)
```

Rilostat

Get more Support

Description

On regular basis new tutorial and examples are building and available through this function.

Details

brief description of the package

Package:	Rilostat
Type:	Package
Version:	See <code>sessionInfo()</code> or DESCRIPTION file
Date:	2020-2025
License:	BSD_2_clause + LICENSE
LazyLoad:	yes

R Tools for ilostat Open Data

Author(s)

David Bescond <bescond@ilo.org>

References

See citation("Rilostat")

ilostat bulk download facility user guidelines https://www.ilo.org/ilostat-files/Documents/ILOSTAT_BulkDownload_Guidelines.pdf

Examples

```
## Not run:  
  
# check which documentation have been recently added:  
  
# help(Rilostat)  
  
# https://ilostat.github.io/Rilostat/  
  
## End(Not run)
```

runExplorer

Play with ilostat explorer

Description

open ilostat explorer app on your computer.

Usage

```
runExplorer(  
  port = getOption("shiny.port", 3838),  
  launch.browser = getOption("shiny.launch.browser", interactive()),  
  host = getOption("shiny.host", "127.0.0.1"),  
  display.mode = "normal"  
)
```

Arguments

port	The TCP port that the application should listen on. If the 'port' is not specified, and the 'shiny.port' option is set (with 'options(shiny.port = XX)'), then that port will be used. Otherwise, use the 3838 port.
launch.browser	If true, the system's default web browser will be launched automatically after the app is started. Defaults to true in interactive sessions only. This value of this parameter can also be a function to call with the application's URL.

host	The IPv4 address that the application should listen on. Defaults to the ‘shiny.host’ option, if set, or ‘"127.0.0.1"’ if not. See Details.
display.mode	The mode in which to display the application. If set to the value "showcase", shows application code and metadata from a DESCRIPTION file in the application directory alongside the application. If set to "normal", displays the application normally. Defaults to "auto", which displays the application in the mode given in its DESCRIPTION file, if any.

Author(s)

David Bescond <bescond@ilo.org>

References

See citation("Rilostat")

Examples

```
## Not run:
runExplorer()

## End(Not run)
```

sdmx_ilstat

Read Ilostat Data, Metadata via ILO SDMX api

Description

query codelist, data and metadata via ilo sdmx api

Usage

```
sdmx_ilstat(
  dsd,
  sdmx_resource = getOption("ilstat_sdmx_resource", "codelist"),
  sdmx_format = "csv",
  lang = getOption("ilstat_lang", "en"),
  count = getOption("ilstat_sdmx_count", FALSE),
  quiet = getOption("ilstat_quiet", FALSE)
)
```

Arguments

dsd	A datastructure definition, see examples section,
sdmx_resource	: a character, type of info to be returned from the sdmx api: 'codelist' (default), 'data', 'dataflow', 'conceptref',
sdmx_format	: for data only, a character, format of info to be returned from the sdmx api: 'csv' (default) others no longer available.
lang	a character for language. Available are "en" (default), "fr" and "es". Can be set also with options(ilstat_lang = 'fr'),
count	a logical, count data records only if resource = 'data', FALSE (default),
quiet	a logical, if TRUE , don't return message from processing, FALSE (default). Can be set also with options(ilstat_quiet = TRUE).

Author(s)

David Bescond <bescond@ilo.org>

References

See citation("Rilostat")

ilstat sdmx user guidelines: "https://www.ilo.org/ilstat-files/Documents/SDMX_User_Guide.pdf"

Examples

```
## Not run:
##### get codelist
# fetch indicator define on ILOSTAT
dic <- sdmx_ilstat(dsd = "CL_INDICATOR", lang ="en")
head(dic)

# fetch country available on ILOSTAT
dic <- sdmx_ilstat(dsd = "CL_AREA", lang ="es")
head(dic)

# fetch classif ECO version available on ILOSTAT
dic <- sdmx_ilstat(dsd = "CL_ECO", lang ="en")
head(dic)

# fetch note type available on ILOSTAT
dic <- sdmx_ilstat(dsd = "CL_NOTE_TYPE", lang ="en")
head(dic)

# fetch note "Repository" available on ILOSTAT
dic <- sdmx_ilstat(dsd = "CL_NOTE_R1", lang ="en")
head(dic)

##### get data

### with attribute
dat <- sdmx_ilstat(dsd = 'STI_ALB_EMP_TEMP_SEX_AGE_NB',
```

```
                                sdmx_resource = 'data')
head(dat)

# without attribute
dat <- sdmx_ilstat(dsd = "STI_DEU_EMP_TEMP_SEX_AGE_NB?detail=dataonly",
                  sdmx_resource = 'data')
head(dat)

##### with multi country and advanced filters

# to get the order of the filter first get the conceptref of the DSD

filter_position <- sdmx_ilstat(dsd = 'STI_ALL_EMP_TEMP_SEX_AGE_NB',
                              sdmx_resource = 'conceptref')
filter_position

# COUNTRY and FREQ are in second and third position of the filters

dat <- sdmx_ilstat(dsd = "STI_ALL_EMP_TEMP_SEX_AGE_NB/.FRA+DEU.Q...",
                  sdmx_resource = 'data')
head(dat)

##### dataflow available

flow <- sdmx_ilstat("STI_TTO_MULTI", sdmx_resource = 'dataflow')

flow <- sdmx_ilstat("KI_ALL_EMP_MULTI", sdmx_resource = 'dataflow')

flow <- sdmx_ilstat("YI_FRA_UNE_MULTI", sdmx_resource = 'dataflow')

##### count data available

sdmx_ilstat("STI_FRA_UNE_TUNE_SEX_AGE_NB/STI.FRA....",
           sdmx_resource = 'data', count = TRUE)

## End(Not run)
```


Index

- * **database**

- get_ilstat_dic, 8

- get_ilstat_toc, 9

- * **package**

- Rilostat, 12

- * **utilities**

- get_ilstat_dic, 8

- get_ilstat_toc, 9

clean_ilstat_cache, 2, 6

Date, 5

distribution_ilstat, 3

get_ilstat, 2, 4, 8, 10, 12

get_ilstat_dic, 8

get_ilstat_toc, 4, 6, 9

label_ilstat, 6, 8, 11

Rilostat, 12

runExplorer, 13

sdmx_ilstat, 14

tempdir, 2, 5