Package ‘ech’

May 11, 2023

Title  Downloading and Processing Microdata from ECH-INE (Uruguay)

Version  0.1.3

Maintainer  Gabriela Mathieu <calcita@gmx.li>

Description  A consistent tool for downloading ECH data, processing them and generating new indicators: poverty, education, employment, etc. All data are downloaded from the official site of the National Institute of Statistics at https://www.gub.uy/instituto-nacional-estadistica/datos-y-estadisticas/encuestas/encuesta-continua-hogares.

License  GPL-3

Depends  R (>= 3.5.0)

Imports  assertthat, curl, dplyr (>= 1.0.0), fs, geouy, glue, haven (>= 2.3.0), janitor, labelled, laeken, rlang, srvyr (>= 0.4.0), statar, stringr, survey, tidyr, utils

Suggests  knitr, rmarkdown, testthat (>= 2.1.0)

VignetteBuilder  knitr

ByteCompile  true

Encoding  UTF-8

LazyData  true

RooxygenNote  7.2.3

SystemRequirements  ‘unrar’ (Linux/macOS) or ‘7-Zip’ (Windows) to work with ‘.rar’ files, GDAL (>= 3.0.2), GEOS (>= 3.8.0), PROJ (>= 6.2.1)

NeedsCompilation  no

Author  Gabriela Mathieu [aut, cre, cph]
  (<https://orcid.org/0000-0003-3965-9024>),
  Richard Detomasi [aut] (<https://orcid.org/0000-0002-6725-0261>),
  Tati Micheletti [ctb] (<https://orcid.org/0000-0003-4838-8342>),
  Instituto Nacional de Estadistica, Uruguay (INE) [dtc]

Repository  CRAN

Date/Publication  2023-05-11 08:50:17 UTC
R topics documented:

age_groups .................................................. 3
archive_extract ........................................... 4
basket_goods .................................................. 4
branch_ciiu .................................................. 5
cba_cbna_int .................................................. 6
cba_cbna_mdeo ............................................. 7
cba_cbna_rur ............................................... 7
dates_ech ..................................................... 8
deflate .......................................................... 9
dic ............................................................ 10
ech ............................................................ 11
employment ................................................... 11
employment_restrictions ..................................... 12
enrolled_school .............................................. 13
get_estimation_gini .......................................... 14
get_estimation_gpg .......................................... 15
get_estimation_mean ......................................... 16
get_estimation_median ........................................ 17
get_estimation_qsr .......................................... 19
get_estimation_ratio ......................................... 20
get_estimation_total ......................................... 21
get_microdata ............................................... 22
household_type .............................................. 23
housing_conditions ......................................... 24
housing_deprivation ......................................... 25
housing_situation ........................................... 26
housing_tenure ............................................... 27
income_constant_prices ..................................... 28
income_quantiles .......................................... 29
integrated_poverty_measure ................................ 30
ipab_base2010 ............................................... 31
ipab_base2010_int ............................................ 32
ipab_base2010_mdeo .......................................... 32
ipc_base2010 .................................................. 33
ipc_base2010_int ............................................. 34
ipc_base2010_mdeo ........................................... 35
labor_income_per_capita ..................................... 35
labor_income_per_hour ....................................... 38
level_completion ............................................. 39
level_education ............................................. 41
organize_educ ............................................... 42
organize_ht11 ................................................ 43
organize_names ............................................. 43
overcrowding ................................................ 44
poverty ....................................................... 45
read_microdata .............................................. 46
age_groups

This function allows you to calculate age groups

Description
This function allows you to calculate age groups

Usage
```r
age_groups(data = ech::toy_ech_2018, cut = c(0, 4, 11, 17, 24), e27 = "e27")
```

Arguments

- `data` : data.frame
- `cut` : breaks points to cut a numeric variable
- `e27` : Variable name of age

Value

- data.frame

See Also

Other demographic: `household_type()`

Examples

```r
# toy_ech_2018 <- age_groups(data = ech::toy_ech_2018, cut = c(0, 4, 11, 17, 24))
```
archive_extract  

*Extract compressed archives*

**Description**
Extract compressed archives

**Usage**
```r
archive_extract(archive.path = NULL, dest.path = NULL)
```

**Arguments**
- **archive.path**  
  Ruta de origen del archivo comprimido
- **dest.path**  
  Ruta destino del archivo descomprimido

**Details**
Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

**Value**
No return value, called for side effects

**See Also**
Other utils: `dates_ech()`, `ech`, `unlabelled()`, `unrarPath`

---

basket_goods  

*This function allows you to get the Basket goods*

**Description**
This function allows you to get the Basket goods

**Usage**
```r
basket_goods(data = ech::cba_cbna_mdeo, year = NULL)
```

**Arguments**
- **data**  
  data.frame with the price of the basket of goods from Montevideo, Interior or Rural region
- **year**  
  the ECH year
branch_ciiu

Details
Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value
data.frame

See Also
Other income: deflate(), income_constant_prices(), income_quantiles(), labor_income_per_capita(), labor_income_per_hour(), organize_ht11()

Examples
    df <- basket_goods(data = ech::cba_cbnia_mdeo, year = 2018)

branch_ciiu  This function allows you to identify activity branches

Description
This function allows you to identify activity branches

Usage
branch_ciiu(
    data = ech::toy_ech_2018,
    f72_2 = "f72_2",
    group = TRUE,
    disaggregated = FALSE
)

Arguments
  data  data.frame
  f72_2  Variable name of ciu code rev.4
  group  logical to define 12 or 18 categories, if FALSE code 18. Default: TRUE
  disaggregated  logical to define disaggregated branches or not. Default: FALSE

Details
Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.
**Value**

data.frame

**See Also**

Other employment: `employment_restrictions()`, `employment()`, `underemployment()`

**Examples**

toy_ech_2018 <- branch_ciiu(data = ech::toy_ech_2018)

cba_cbna_int

---

**cba_cbna_int**  
* A dataset containing the CBA and CBNA for the Interior Urbano region

**Description**

A dataset containing the CBA and CBNA for the Interior Urbano region

**Usage**

cba_cbna_int

**Format**

A data frame with 234 rows and 4 variables:

- **fecha** date from 2001 to 2020
- **cba_li** CBA
- **cbna** CBNA
- **cbt_lp** CBT

**Details**

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

**Source**

[https://www.gub.uy/instituto-nacional-estadistica/](https://www.gub.uy/instituto-nacional-estadistica/)

**See Also**

**cba_cbna_mdeo**

A dataset containing the CBA and CBNA for the Montevideo region

**Description**
A dataset containing the CBA and CBNA for the Montevideo region

**Usage**
cba_cbna_mdeo

**Format**
A data frame with 234 rows and 4 variables:

- **fecha** date from 2001 to 2020
- **cba_lí** CBA
- **cbna** CBNA
- **cbt_lp** CBT

**Details**
Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

**Source**
[https://www.gub.uy/instituto-nacional-estadistica/](https://www.gub.uy/instituto-nacional-estadistica/)

**See Also**

**cba_cbna_rur**

A dataset containing the CBA and CBNA for the Interior Rural region

**Description**
A dataset containing the CBA and CBNA for the Interior Rural region

**Usage**
cba_cbna_rur
dates_ech

Format
A data frame with 234 rows and 4 variables:

fecha  date from 2001 to 2020
cba_li  CBA
cbna  CBNA
cbt_lp  CBT

Details
Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Source
https://www.gub.uy/instituto-nacional-estadistica/

See Also

---

dates_ech  This function allows you to organize dates

Description
This function allows you to organize dates

Usage
dates_ech(data)

Arguments
data  data frame with an ’yy’ variable for the year, and a ’mm’ variable for the month

Value
data.frame

See Also
Other utils: archive_extract(), ech, unlabelled(), unrarPath
This function allows you to calculate a deflator coefficient.

Usage

deflate(
    base_month = NULL,
    base_year = NULL,
    index = "IPC",
    level = "G",
    df_year = NULL
)

Arguments

- base_month: baseline month
- base_year: baseline year
- index: IPC or IPAB
- level: General index ('G'), Montevideo index ('M') or Interior index ('I')
- df_year: ECH year

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value

vector

See Also

Other income: `basket_goods()`, `income_constant_prices()`, `income_quantiles()`, `labor_income_per_capita()`, `labor_income_per_hour()`, `organize_html()`
A dataset containing variables names change of the ECH 2006-2018

Description
A dataset containing variables names change of the ECH 2006-2018

Usage
dic

Format
A data frame with 976 rows and 21 variables:

- **codos**  Code oh label
- **descripcion**  Description of label
- **modulo**  Module in the form 2017
- **obs**  Observations
- **unidad**  Level of variable household (H) individual (P) or general (G)
- **var06**  ECH variables names 2006
- **var07**  ECH variables names 2007
- **var08**  ECH variables names 2008
- **var09**  ECH variables names 2009
- **var10**  ECH variables names 2010
- **var11**  ECH variables names 2011
- **var12**  ECH variables names 2012
- **var13**  ECH variables names 2013
- **var14**  ECH variables names 2014
- **var15**  ECH variables names 2015
- **var16**  ECH variables names 2016
- **var17**  ECH variables names 2017
- **var18**  ECH variables names 2018
- **var19**  ECH variables names 2019
- **var21**  ECH variables names 2021 segunda semestre
- **var22**  ECH variables names 2022 primer semestre

Details
Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.
ech

Description
Toolbox for Downloading and Processing Microdata from the Continuous Household Survey of Uruguay (ECH)
See the README on Github

See Also
Other utils: archive_extract(), dates_ech(), unlabelled(), unrarPath

employment

Description
This function allows you to calculate the variables: PEA, PET, PO, PD

Usage
employment(data = ech::toy_ech_2018, pobpcoac = "pobpcoac")

Arguments
data data.frame with microdata
pobpcoac Variable name of definition of population by activity status. Default: "pobpcoac"

Details
Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.
employment_restrictions

Value
data.frame, tbl and tbl_df object

See Also
Other employment: branch_ciiu(), employment_restrictions(), underemployment()

Examples
toy_ech_2018 <- employment(data = ech::toy_ech_2018, pobpcoac = "pobpcoac")

employment_restrictions

This function allows you to identify workers with employment restrictions

Description
This function allows you to identify workers with employment restrictions

Usage

employment_restrictions(
data = ech::toy_ech_2018,
f82 = "f82",
underemployment = "underemployment"
)

Arguments
data data.frame
f82 Variable name of contribution to the pension fund
underemployment Variable name of underemployment

Details
Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value
data.frame
enrolled_school

See Also

Other employment: branch_ciiu(), employment(), underemployment()

Examples

```
toy_ech_2018 <- underemployment(data = ech::toy_ech_2018)
toy_ech_2018 <- employment_restrictions(data = toy_ech_2018)
```

enrolled_school

This function allows you to calculate the people enrolled in school

Description

This function allows you to calculate the people enrolled in school

Usage

```
enrolled_school(
  data = ech::toy_ech_2018,
  e27 = "e27",
  e193 = "e193",
  e197 = "e197",
  e201 = "e201",
  e212 = "e212",
  e215 = "e215",
  e218 = "e218",
  e221 = "e221",
  e224 = "e224"
)
```

Arguments

data data.frame with necessary variables Defaults to ech.
e27 Variable name of age
e193 Variable name of attendance school
e197 Variable name of attendance primary
e201 Variable name of attendance secondary
e212 Variable name of attendance technical school (non-university)
e215 Variable name of attendance magisterio
e218 Variable name of attendance university
e221 Variable name of attendance tertiary
e224 Variable name of attendance postgrade
Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value
data.frame

See Also

Other education: level_completion(), level_education(), organize_educ(), years_of_schooling()

Examples

```
toy_ech_2018 <- enrolled_school(data = ech::toy_ech_2018)
```

---

**get_estimation_gini**  
This function allows you to estimate the Gini coefficient

Description

This function allows you to estimate the Gini coefficient

Usage

```
get_estimation_gini(
  data = ech::toy_ech_2018,
  variable = NULL,
  by = NULL,
  level = NULL,
  ids = NULL,
  numero = "numero",
  estrato = NULL,
  pesoano = "pesoano",
  bootstrap = FALSE,
  r = NULL
)
```

Arguments

data  ech data frame
variable  Variable name of income without rental value per capita deflated
by  data frame column
level  is household ("h") or individual ("i").
This function allows you to estimate the Gender Pay Wage Gap (GPG)

Usage

```r
get_estimation_gpg(
  data = ech::toy_ech_2018,
  variable = "total_income_per_hour",
  e26 = "e26",
  by = NULL,
  ids = NULL,
  estrato = NULL,
  pesoano = "pesoano",
  stat = "media"
)
```
get_estimation_mean

Arguments

data: data.frame
variable: Variable name of total income per hour
e26: Variable name of sex
by: data frame column
ids: Variable name of cluster
estrato: Variable name of strata
pesoano: Variable name of weights
stat: Media or Median

Value

table

See Also

Other estimation: get_estimation_gini(), get_estimation_mean(), get_estimation_median(),
get_estimation_qsr(), get_estimation_ratio(), get_estimation_total(), set_design()

Examples

toy_ech_2018 <- labor_income_per_hour(data = ech::toy_ech_2018, base_month = 6, base_year = 2018)
get_estimation_gpg(data = toy_ech_2018, variable = "total_income_per_hour", e26 = "e26")

generate_estimation_mean <- function(data = ech::toy_ech_2018, variable = "total_income_per_hour", e26 = "e26")
get_estimation_gpg(data = toy_ech_2018, variable = "total_income_per_hour", e26 = "e26")

Description

This function allows you to estimate mean variable at universe level.

Usage

generate_estimation_mean(
  data = ech::toy_ech_2018,
  variable = NULL,
  by.x = NULL,
  by.y = NULL,
  domain = NULL,
  level = NULL,
  ids = NULL,
  numero = "numero",
  estrato = NULL,
  pesoano = "pesoano",
  name = "estimacion"
)
Arguments

data          data frame with ECH microdata
variable      data frame column to estimate
by.x          data frame column
by.y          data frame column
domain        subpopulation reference setted as character expresion of logical evaluation
level         is household ("h") or individual ("i").
ids           ids
numero        household id
estrato       strata
pesoano       weights
name          name for the estimation new column

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value

table

See Also

Other estimation: get_estimation_gini(), get_estimation_gpg(), get_estimation_median(), get_estimation_qsr(), get_estimation_ratio(), get_estimation_total(), set_design()

Examples

generate_estimation_mean(data = ech::toy_ech_2018, variable = "pobre06", by.x = "dpto", level = "h")

This function allows you to estimate median variable at universe level.
Usage

get_estimation_median(
    data = ech::toy_ech_2018,
    variable = NULL,
    by.x = NULL,
    by.y = NULL,
    domain = NULL,
    level = NULL,
    ids = NULL,
    numero = "numero",
    estrato = NULL,
    pesoano = "pesoano",
    name = "estimacion"
)

Arguments

data data frame with ECH microdata
variable data frame column to estimate
by.x data frame column
by.y data frame column
domain subpopulation reference setted as character expresion of logical evaluation
level is household ("h") or individual ("i").
ids ids
numero household id
estrato strata
pesoano weights
name name for the estimation new column

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value

table

See Also

Other estimation: get_estimation_gini(), get_estimation_gpg(), get_estimation_mean(),
get_estimation_qsr(), get_estimation_ratio(), get_estimation_total(), set_design()
get_estimation_qsr

Examples

get_estimation_median(data = ech::toy_ech_2018, variable = "ht11", by.x = "dpto", level = "h")

get_estimation_qsr

This function allows you to estimate de Income Quintile Share Ratio

Description

This function allows you to estimate de Income Quintile Share Ratio

Usage

get_estimation_qsr(
  data = ech::toy_ech_2018,
  variable = "y_pc_d_r",
  by = NULL,
  ids = NULL,
  estrato = NULL,
  pesoano = "pesoano"
)

Arguments

data  data.frame
variable Variable name of total income per hour
by data frame column
ids Variable name of cluster
estrato Variable name of strata
pesoano Variable name of weights

Value

table

See Also

Other estimation: get_estimation_gini(), get_estimation_gpg(), get_estimation_mean(),
get_estimation_median(), get_estimation_ratio(), get_estimation_total(), set_design()

Examples

toy_ech_2018 <- income_constant_prices(data = ech::toy_ech_2018, index = "IPC", level = "R",
  base_month = "01", base_year = "2005")
get_estimation_qsr(data = toy_ech_2018, variable = "y_pc_d_r", pesoano = "pesoano")
get_estimation_ratio  

This function allows you to estimate ratio variables at universe level.

Description

This function allows you to estimate ratio variables at universe level.

Usage

```r
get_estimation_ratio(
  data = ech::toy_ech_2018,
  variable.x = NULL,
  variable.y = NULL,
  by.x = NULL,
  by.y = NULL,
  domain = NULL,
  level = NULL,
  ids = NULL,
  numero = "numero",
  estrato = NULL,
  pesoano = "pesoano",
  name = "estimacion"
)
```

Arguments

- `data`  
  data frame with ECH microdata
- `variable.x`  
  data frame column to estimate
- `variable.y`  
  data frame column to estimate
- `by.x`  
  data frame column
- `by.y`  
  data frame column
- `domain`  
  subpopulation reference setted as character expresion of logical evaluation
- `level`  
  is household ("h") or individual ("i")
- `ids`  
  Variable name of cluster
- `numero`  
  Variable name of household id
- `estrato`  
  Variable name of strata
- `pesoano`  
  Variable name of weights
- `name`  
  name for the estimation new column

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.
get_estimation_total

Value

table

See Also

Other estimation: get_estimation_gini(), get_estimation_gpg(), get_estimation_mean(), get_estimation_median(), get_estimation_qsr(), get_estimation_total(), set_design()

Examples

toy_ech_2018 <- employment(data = ech::toy_ech_2018, pobpcoac = "pobpcoac")
get_estimation_ratio(data = toy_ech_2018, variable.x = "po", variable.y = "pea", level = "i")

get_estimation_total  This function allows you to estimate total variable at universe level.

Description

This function allows you to estimate total variable at universe level.

Usage

get_estimation_total(
  data = ech::toy_ech_2018,
  variable = NULL,
  by.x = NULL,
  by.y = NULL,
  domain = NULL,
  level = NULL,
  ids = NULL,
  numero = "numero",
  estrato = NULL,
  pesoano = "pesoano",
  name = "estimacion"
)

Arguments

data  data frame with ECH microdata
variable  data frame column to estimate
by.x  data frame column
by.y  data frame column
domain  subpopulation reference setted as character expresion of logical evaluation
level  is household ("h") or individual ("i").
get_microdata

```r
ids      ids
numero   household id
estrato  strata
pesoano  weights
name     name for the estimation new column
```

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value

table

See Also

Other estimation: `get_estimation_gini()`, `get_estimation_gpg()`, `get_estimation_mean()`, `get_estimation_median()`, `get_estimation_qsr()`, `get_estimation_ratio()`, `set_design()`

Examples

```r
get_estimation_total(variable = "pobre06", by.x = "dpto", level = "h")
```

Description

This function allows you to download and read ECH from INE website

Usage

```r
get_microdata(year = NULL, folder = tempdir(), toR = TRUE)
```

Arguments

- `year` allows download data from 2011 to 2019. Default the last year
- `folder` Folder where are the files or be download
- `toR` write data frame in R format and delete download file and unpack files

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.
household_type

Value
unrar files from INE web and the respective data frame in tibble format

See Also
Other dwnd_read: read_microdata()

household_type

This function allows you to calculate the household type for each household in the survey. A household is composed of one or more people who occupy a housing unit.

Description
This function allows you to calculate the household type for each household in the survey. A household is composed of one or more people who occupy a housing unit.

Usage
household_type(
  data = ech::toy_ech_2018,
  numero = "numero",
  e26 = "e26",
  e27 = "e27",
  e30 = "e30"
)

Arguments
data data frame with ECH microdata
numero Variable name of household id
e26 Variable name of sex
e27 Variable name of age
e30 Variable name of householder

Details
Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value
data.frame

See Also
Other demographic: age_groups()
Examples

```r
toy_ech_2018 <- household_type(data = ech::toy_ech_2018)
```

---

**housing_conditions**

This function allows you to calculate the housing conditions

**Description**

This function allows you to calculate the housing conditions

**Usage**

```r
housing_conditions(data = ech::toy_ech_2018, c2 = "c2", c3 = "c3", c4 = "c4")
```

**Arguments**

- `data` : data.frame
- `c2` : Variable name of predominant material on external walls
- `c3` : Variable name of predominant roofing material
- `c4` : Variable name of predominant flooring material

**Details**

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

**Value**

- data.frame

**See Also**

Other dwelling: `housing_deprivation()`, `housing_situation()`, `housing_tenure()`, `overcrowding()`

**Examples**

```r
toy_ech_2018 <- housing_conditions(data = ech::toy_ech_2018)
```
**housing_deprivation**  
*This function allows you to calculate the housing status*

**Description**

This function allows you to calculate the housing status

**Usage**

```r
housing_deprivation(
  data = ech::toy_ech_2018,
  n = 1,
  ht19 = "ht19",
  d9 = "d9",
  d10 = "d10",
  d11 = "d11",
  d12 = "d12",
  d13 = "d13",
  d16 = "d16",
  d18 = "d18",
  d19 = "d19",
  c2 = "c2",
  c3 = "c3",
  c4 = "c4",
  quintil = "quintil",
  region_4 = "region_4"
)
```

**Arguments**

- `data`  
  data.frame
- `n`  
  number of deprivations to consider. Default 1
- `ht19`  
  Variable name of number of individuals in the household
- `d9`  
  Variable name of number of rooms
- `d10`  
  Variable name of number of rooms to sleep
- `d11`  
  Variable name of principal source of potable water
- `d12`  
  Variable name of water supply network / water access
- `d13`  
  Variable name of sanitary facilities
- `d16`  
  Variable name of sewerage facilities
- `d18`  
  Variable name of energy source for lighting
- `d19`  
  Variable name of cooking space
- `c2`  
  Variable name of predominant material on external walls
- `c3`  
  Variable name of predominant roofing material
housing_situation

c4 Variable name of predominant flooring material
quintil Variable name of income quintil
region_4 Variable name of region

Details
Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value
data.frame

See Also
Other dwelling: housing_conditions(), housing_situation(), housing_tenure(), overcrowding()

Examples

```r
toy_ech_2018 <- income_constant_prices(data = ech::toy_ech_2018)
toy_ech_2018 <- income_quantiles(data = toy_ech_2018)
toy_ech_2018 <- housing_deprivation(data = toy_ech_2018)
```

---

housing_situation This function allows you to calculate the housing situation

Description
This function allows you to calculate the housing situation

Usage

```r
housing_situation(
  data = ech::toy_ech_2018,
  c5_1 = "c5_1",
  c5_2 = "c5_2",
  c5_3 = "c5_3",
  c5_4 = "c5_4",
  c5_5 = "c5_5",
  c5_6 = "c5_6",
  c5_7 = "c5_7",
  c5_8 = "c5_8",
  c5_9 = "c5_9",
  c5_10 = "c5_10",
  c5_11 = "c5_11",
  c5_12 = "c5_12"
)
```
housing_tenure

Arguments

data	data.frame
c5_1	Variable name of roof condensation
c5_2	Variable name of roof drips
c5_3	Variable name of walls cracks
c5_4	Variable name of broken doors or windows
c5_5	Variable name of floors cracks
c5_6	Variable name of plaster drop on walls
c5_7	Variable name of detached ceilings
c5_8	Variable name of poor sunlight
c5_9	Variable name of poor ventilation
c5_10	Variable name of floods when it rains
c5_11	Variable name of in danger of collapse
c5_12	Variable name of dampness in the foundations

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value

data.frame

See Also

Other dwelling: housing_conditions(), housing_deprivation(), housing_tenure(), overcrowding()

Examples

    toy_ech_2018 <- housing_situation(data = ech::toy_ech_2018)

    housing_tenure

    housing_tenure(data = ech::toy_ech_2018, d8_1 = "d8_1")

Description

This function allows you to calculate the housing tenure

Usage

    housing_tenure(data = ech::toy_ech_2018, d8_1 = "d8_1")
# income_constant_prices

## Arguments

- **data**: data.frame
- **d8_1**: Variable name of housing_tenure (owner, renter, rent-free occupancy, etc.)

## Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

## Value

data.frame

## See Also

Other dwelling: `housing_conditions()`, `housing_deprivation()`, `housing_situation()`, `overcrowding()`

## Examples

```r
toy_ech_2018 <- housing_tenure(data = ech::toy_ech_2018)
```

```r
# income_constant_prices

This function allows you to calculate the household income constant prices

## Description

This function allows you to calculate the household income constant prices

## Usage

```r
income_constant_prices(
  data = ech::toy_ech_2018,
  base_month = 6,
  base_year = 2018,
  index = "IPC",
  level = "G",
  mes = "mes",
  ht11 = "ht11",
  ht13 = "ht13",
  ht19 = "ht19"
)
```
income_quantiles

Arguments

- **data**: data.frame with ECH microdata
- **base_month**: baseline month
- **base_year**: baseline year
- **index**: IPC or IPAB
- **level**: General ("G") or Regional ("R")
- **mes**: month
- **ht11**: Variable name of income. Default: ht11
- **ht13**: Variable name of rental value. Default: ht13
- **ht19**: Variable name of number of individuals in the household. Default: ht19

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value
data.frame

See Also

Other income: `basket_goods()`, `deflate()`, `income_quantiles()`, `labor_income_per_capita()`, `labor_income_per_hour()`, `organize_ht11()`

Examples

toy_ech_2018 <- income_constant_prices(data = ech::toy_ech_2018)

income_quantiles

This function allows you to calculate the Household Income Quantiles

Description

This function allows you to calculate the Household Income Quantiles

Usage

```
income_quantiles(
  data = ech::toy_ech_2018,
  quantile = 5,
  weights = "pesoano",
  income = "y_pc_d"
)
```
integrated_poverty_measure

Arguments

data data.frame
quantile Variable name of quintil (5) or decil (10). Default: 5
weights Variable name of ponderation variable. Default: "pesoano"
income Variable name of income constant price. Default: "y_pc_d"

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value
data.frame

See Also

Other income: basket_goods(), deflate(), income_constant_prices(), labor_income_per_capita(), labor_income_per_hour(), organize_ht11()

Examples

toy_ech_2018 <- income_constant_prices(data = ech::toy_ech_2018)
toy_ech_2018 <- income_quantiles(data = toy_ech_2018)

integrated_poverty_measure

This function allows you to calculate an integrated poverty measure

Description

This function allows you to calculate an integrated poverty measure

Usage

integrated_poverty_measure(
  data = ech::toy_ech_2018,
  pobre06 = "pobre06",
  UBN_q = "UBN_q"
)

Arguments

data data.frame
pobre06 Variable name of poverty
UBN_q Variable name of UBN
Value
data.frame

See Also
Other poverty: poverty(), unsatisfied_basic_needs()

Examples

```r
toy_ech_18 <- enrolled_school(data = ech::toy_ech_2018)
toy_ech_18 <- years_of_schooling(toy_ech_18)
toy_ech_18 <- unsatisfied_basic_needs(toy_ech_18)
toy_ech_18 <- integrated_poverty_measure(data = toy_ech_18)
```

ipab_base2010

A dataset containing the IPAB

Description
A dataset containing the IPAB

Usage
ipab_base2010

Format
A data frame with 286 rows and 2 variables:

- **fecha** date from 1997 to 2020
- **indice** IPAB

Details
Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Source
https://www.gub.uy/instituto-nacional-estadistica/

See Also
Description

A dataset containing the IPAB for the Interior region

Usage

ipab_base2010_int

Format

A data frame with 108 rows and 2 variables:

fecha  date from 2011 to 2019
indice  IPAB

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Source

https://www.gub.uy/instituto-nacional-estadistica/

See Also


Description

A dataset containing the IPAB for the Montevideo region

Usage

ipab_base2010_mdeo
A dataset containing the IPC base 2010

Description
A dataset containing the IPC base 2010

Usage
ipc_base2010

Format
A data frame with 990 rows and 5 variables:

fecha date from 1937 to 2019
indice IPC
mensual mensual value of IPC
trimestre three-month period value of IPC
cuatrimestre four-month period value of IPC
semestre six-month period value of IPC
acum_ano acumulated IPC
acum_12_meses acumulated IPC last 12 month
Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Source

https://www.gub.uy/instituto-nacional-estadistica/

See Also


---

ipc_base2010_int     A dataset containing the IPC base 2010 only for the Interior region

Description

A dataset containing the IPC base 2010 only for the Interior region

Usage

ipc_base2010_int

Format

A data frame with 120 rows and 2 variables:

fecha  date from 2011 to 2019
indice  IPC

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Source

https://www.gub.uy/instituto-nacional-estadistica/

See Also

A dataset containing the IPC base 2010 only for the Montevideo region

Description
A dataset containing the IPC base 2010 only for the Montevideo region

Usage
ipc_base2010_mdeo

Format
A data frame with 120 rows and 2 variables:

fecha  date from 2011 to 2019
indice  IPC

Details
Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Source
https://www.gub.uy/instituto-nacional-estadistica/

See Also

This function allows you to calculate the labor income per capita

Description
This function allows you to calculate the labor income per capita
Usage

labor_income_per_capita(
  data = ech::toy_ech_2018,
  numero = "numero",
  pobpcoac = "pobpcoac",
  g126_1 = "g126_1",
  g126_2 = "g126_2",
  g126_3 = "g126_3",
  g126_4 = "g126_4",
  g126_5 = "g126_5",
  g126_6 = "g126_6",
  g126_7 = "g126_7",
  g126_8 = "g126_8",
  g127_3 = "g127_3",
  g128_1 = "g128_1",
  g129_2 = "g129_2",
  g130_1 = "g130_1",
  g131_1 = "g131_1",
  g133_1 = "g133_1",
  g133_2 = "g133_2",
  g134_1 = "g134_1",
  g134_2 = "g134_2",
  g134_3 = "g134_3",
  g134_4 = "g134_4",
  g134_5 = "g134_5",
  g134_6 = "g134_6",
  g134_7 = "g134_7",
  g134_8 = "g134_8",
  g135_3 = "g135_3",
  g136_1 = "g136_1",
  g137_2 = "g137_2",
  g138_1 = "g138_1",
  g139_1 = "g139_1",
  g141_1 = "g141_1",
  g141_2 = "g141_2",
  g142 = "g142",
  g144_1 = "g144_1",
  g144_2_1 = "g144_2_1",
  g144_2_3 = "g144_2_3",
  g144_2_4 = "g144_2_4",
  g144_2_5 = "g144_2_5"
)

Arguments

data data frame
numero Variable name of household id
pobpcoac Variable name of definition of population by activity status
labor_income_per_capita

\[ g_{126}_1 \quad \text{Variable name of net salary} \]
\[ g_{126}_2 \quad \text{Variable name of commissions, incentives, overtime payment, fringe benefits} \]
\[ g_{126}_3 \quad \text{Variable name of non-surrendering expenses} \]
\[ g_{126}_4 \quad \text{Variable name of tips} \]
\[ g_{126}_5 \quad \text{Variable name of annual complementary salary} \]
\[ g_{126}_6 \quad \text{Variable name of vacation pay} \]
\[ g_{126}_7 \quad \text{Variable name of delayed payments} \]
\[ g_{126}_8 \quad \text{Variable name of transportation tickets} \]
\[ g_{127}_3 \quad \text{Variable name of received food or drink} \]
\[ g_{128}_1 \quad \text{Variable name of received food tickets} \]
\[ g_{129}_2 \quad \text{Variable name of received housing or accommodation} \]
\[ g_{130}_1 \quad \text{Variable name of another type of compensation} \]
\[ g_{131}_1 \quad \text{Variable name of received another type of supplement paid by the employer} \]
\[ g_{133}_1 \quad \text{Variable name of the right to cultivate goods for own-consumption} \]
\[ g_{133}_2 \quad \text{Variable name of the right to cultivate goods for own-consumption (amount received from the sale)} \]
\[ g_{134}_1 \quad \text{Variable name of net salary} \]
\[ g_{134}_2 \quad \text{Variable name of commissions, incentives, overtime payment, fringe benefits} \]
\[ g_{134}_3 \quad \text{Variable name of non-surrendering expenses} \]
\[ g_{134}_4 \quad \text{Variable name of tips} \]
\[ g_{134}_5 \quad \text{Variable name of annual complementary salary} \]
\[ g_{134}_6 \quad \text{Variable name of vacation pay} \]
\[ g_{134}_7 \quad \text{Variable name of delayed payments} \]
\[ g_{134}_8 \quad \text{Variable name of transportation tickets} \]
\[ g_{135}_3 \quad \text{Variable name of received food or drink} \]
\[ g_{136}_1 \quad \text{Variable name of received food tickets} \]
\[ g_{137}_2 \quad \text{Variable name of received housing or accommodation} \]
\[ g_{138}_1 \quad \text{Variable name of another type of compensation} \]
\[ g_{139}_1 \quad \text{Variable name of received another type of supplement paid by the employer} \]
\[ g_{141}_1 \quad \text{Variable name of the right to cultivate goods for own-consumption} \]
\[ g_{141}_2 \quad \text{Variable name of the right to cultivate goods for own-consumption (amount received from the sale)} \]
\[ g_{142} \quad \text{Variable name of withdrawals for business household expenses you have or had} \]
\[ g_{144}_1 \quad \text{Variable name of collected products for own consumption (non-agricultural worker)} \]
\[ g_{144}_2_1 \quad \text{Variable name of collected products for own consumption (non-agricultural worker)} \]
\[ g_{144}_2_3 \quad \text{Variable name of collected products for own consumption (non-agricultural worker)} \]
\[ g_{144}_2_4 \quad \text{Variable name of collected products for own consumption (non-agricultural worker)} \]
\[ g_{144}_2_5 \quad \text{Variable name of collected products for own consumption (non-agricultural worker)} \]
labor_income_per_hour

Details
Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value
data.frame

See Also
Other income: basket_goods(), deflate(), income_constant_prices(), income_quantiles(), labor_income_per_hour(), organize_ht11()

Examples

```r
toy_ech_2018 <- labor_income_per_capita(data = ech::toy_ech_2018)
```

---
labor_income_per_hour  This function allows you to calculate the labor income per hour

Description
This function allows you to calculate the labor income per hour

Usage

```r
labor_income_per_hour(
  data = ech::toy_ech_2018,
  numero = "numero",
  f85 = "f85",
  pobpcoac = "pobpcoac",
  pt4 = "pt4",
  base_month = 6,
  base_year = 2018,
  mes = "mes"
)
```

Arguments

data  data frame
numero  Variable name of household id
f85  Variable name of hours worked per week
pobpcoac  Variable name of definition of population by activity status
pt4  Variable name of total employment income
**level_completion**

base_month  baseline month
base_year    baseline year
mes          month

**Details**

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

**Value**

data.frame

**See Also**

Other income: `basket_goods()`, `deflate()`, `income_constant_prices()`, `income_quantiles()`, `labor_income_per_capita()`, `organize_ht11()`

**Examples**

```r
toy_ech_2018 <- ech::toy_ech_2018
toy_ech_2018 <- labor_income_per_hour(data = toy_ech_2018, base_month = "06", base_year = "2018")
```

---

**level_completion**  This function allows you to calculate the level of school completion

**Description**

This function allows you to calculate the level of school completion

**Usage**

```r
level_completion(
  data = ech::toy_ech_2018,
  e197 = "e197",
  e197_1 = "e197_1",
  e201 = "e201",
  e51_4 = "e51_4",
  e51_5 = "e51_5",
  e51_6 = "e51_6",
  e51_7_1 = "e51_7_1",
  e51_7 = "e51_7",
  e51_8 = "e51_8",
  e51_9 = "e51_9",
  e51_10 = "e51_10",
)```
level_completion

e212 = "e212",
e215 = "e215",
e218 = "e218",
e221 = "e221",
n = 4
}

Arguments

data data.frame
e197 Variable name of attends primary school
e197_1 Variable name of completed primary
e201 Variable name of attends secondary
e51_4 Variable name of years passed in lower secondary
e51_5 Variable name of years passed in upper secondary
e51_6 Variable name of years passed in technical upper secondary
e51_7_1 Variable name of technical education requirements
e51_7 Variable name of years passed in technical education
e51_8 Variable name of years passed in magisterio/profesorado
e51_9 Variable name of years passed in university or similar
e51_10 Variable name of years passed in tertiary (non-university)
e212 Variable name of attendance technical school (non-university)
e215 Variable name of attendance magisterio
e218 Variable name of attendance university
e221 Variable name of attendance tertiary
n years of tertiary

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value

data.frame

See Also

Other education: enrolled_school(), level_education(), organize_edu(), years_of_schooling()

Examples

toy_ech_2018 <- level_completion(data = ech::toy_ech_2018)
level_education allows you to calculate the highest level of education achieved.

**Description**
This function allows you to calculate the highest level of education achieved.

**Usage**
```r
level_education(
  data = ech::toy_ech_2018,
  e51_2 = "e51_2",
  e51_3 = "e51_3",
  e51_4 = "e51_4",
  e51_5 = "e51_5",
  e51_6 = "e51_6",
  e51_7 = "e51_7",
  e51_7_1 = "e51_7_1",
  e51_8 = "e51_8",
  e51_9 = "e51_9",
  e51_10 = "e51_10",
  e51_11 = "e51_11",
  e193 = "e193",
  e49 = "e49"
)
```

**Arguments**
- `data`: data.frame
- `e51_2`: Variable name of years passed in primary
- `e51_3`: Variable name of years passed in special primary
- `e51_4`: Variable name of years passed in lower secondary
- `e51_5`: Variable name of years passed in upper secondary
- `e51_6`: Variable name of years passed in technical upper secondary
- `e51_7`: Variable name of years passed in technical school
- `e51_7_1`: Variable name of technical school requirements
- `e51_8`: Variable name of years passed in magisterio/profesorado
- `e51_9`: Variable name of years passed in university or similar
- `e51_10`: Variable name of years passed in tertiary (non-university)
- `e51_11`: Variable name of years passed in postgrade
- `e193`: Variable name of attendance school
- `e49`: Variable name of attendance school ever
organize_educ

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value
data.frame

See Also
Other education: enrolled_school(), level_completion(), organize_educ(), years_of_schooling()

Examples
toy_ech_2018 <- level_education(data = ech::toy_ech_2018)

organize_educ(data, year, e49 = "e49", e579 = "e579", numero = "numero")

Description
This function allows you to fix education variables from 2021

Usage
organize_educ(data, year, e49 = "e49", e579 = "e579", numero = "numero")

Arguments
data: data.frame
year: survey year
e49: Variable name of e49
e579: Variable name of e579
numero: Variable name of numero

Value
data.frame

See Also
Other education: enrolled_school(), level_completion(), level_education(), years_of_schooling()
organize_ht11

This function allows you to fix ht11 from 2013 to 2015

Description

This function allows you to fix ht11 from 2013 to 2015

Usage

organize_ht11(data, year, ht11 = "ht11", numero = "numero")

Arguments

data: data.frame
year: survey year
ht11: Variable name of ht11
numero: Variable name of numero

Value

data.frame

See Also

Other income: basket_goods(), deflate(), income_constant_prices(), income_quantiles(), labor_income_per_capita(), labor_income_per_hour()

Examples


organize_names

This function allows you to organize the variables names of ECH with reference in 2017.

Description

This function allows you to organize the variables names of ECH with reference in 2017.

Usage

organize_names(data, year, level = "hyp")
overcrowding

Arguments

- **data**: data.frame contains the ECH microdata
- **year**: numeric reference year of the data. Available from 2011 to 2019
- **level**: (string) indicates whether the base to be labelled is of the type "household", "h", "individual", "i" or both, "hyp". Default "hyp"

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value

data.frame

See Also

Other organize: to_ascii()

Examples

toy_ech_2018 <- organize_names(data = ech::toy_ech_2018, year = 2018, level = "h")

overcrowding

This function allows you to calculate overcrowding in the household

Description

This function allows you to calculate overcrowding in the household

Usage

overcrowding(data = ech::toy_ech_2018, ht19 = "ht19", d10 = "d10")

Arguments

- **data**: data.frame
- **ht19**: Variable name ofumber of individuals in the household
- **d10**: Variable name of number of rooms to sleep

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.
This function allows you to calculate poor and indigent people or household

Usage

```r
poverty(data = ech::toy_ech_2018, scale = 0.8, region_4 = "region_4", dpto = "dpto", ht11 = "ht11", ht19 = "ht19", numero = "numero")
```

Arguments

- `data`: data.frame
- `scale`: equivalency scale
- `region_4`: Variable name of region. Default: region_4
- `dpto`: Variable name of departamento. Default: dpto
- `ht11`: Variable name of income. Default: ht11
- `ht19`: Variable name of number of individuals in the household. Default: ht19
- `numero`: household id

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.
Value

data.frame

See Also

Other poverty: integrated_poverty_measure(), unsatisfied_basic_needs()

Examples

toy_ech_2018 <- poverty(data = ech::toy_ech_2018)

Description

This function allows you to read ECH from a local folder

Usage

read_microdata(path = NULL)

Arguments

path Folder where are the files or be download

Details

Disclaimer: El script no es un producto oficial de INE.

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value

an object called df

See Also

Other dwnld_read: get_microdata()
This function allows you to set the survey design

Usage

```r
set_design(
  data = ech::toy_ech_2018,
  level = "i",
  numero = "numero",
  ids = NULL,
  estrato = NULL,
  pesoano = "pesoano"
)
```

Arguments

- `data` data frame with ECH microdata
- `level` is household ("h") or individual ("i")
- `numero` variables specifying the householder ids
- `ids` variables specifying the unit primary sampling (it's not a public variable)
- `estrato` variable specifying strata
- `pesoano` variable specifying weights

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value

a list

See Also

Other estimation: `get_estimation_gini()`, `get_estimation_gpg()`, `get_estimation_mean()`,
`get_estimation_median()`, `get_estimation_qsr()`, `get_estimation_ratio()`, `get_estimation_total()`

Examples

```r
set_design(data = ech::toy_ech_2018, level = "h")
```
toy_ech_2018  A dataset containing only 1000 rows of the ECH 2018

Description

A dataset containing only 1000 rows of the ECH 2018

Usage

toy_ech_2018

Format

A data frame with 1000 rows and 579 variables:

- **numero**: household id
- **nper**: 
- **anio**: 
- **mes**: 
- **dpto**: 
- **nomdpto**: 
- **secc**: 
- **segm**: 
- **loc_agr_13**: 
- **nom_loc_agr_13**: 
- **ccz**: 
- **barrio**: 
- **nombarrio**: 
- **estred13**: 
- **region_3**: 
- **region_4**: 
- **pesoano**: 
- **pesotri**: 
- **pesomen**: 
- **c1**: 
- **c2**: 
- **c3**: 
- **c4**: 
- **c5_1**: 
- **c5_2**: 
g126_5
g126_6
g126_7
g126_8
g250_1
g250_2
g250_5
g250_3
g250_4
g127
g127_1
g127_2
g127_3
g128
g128_1
g129
g129_1
g129_2
g130
g130_1
g131
g131_1
g132
g132_1
g132_2
g132_3
g133
g133_1
g133_2
g_st_1
g134_1
g134_2
g134_3
g134_4
g134_5
g134_6
g134_7
g255
g256
g152
g151_5
g151_1
g151_2
g151_3
g151_3_1
g151_4
g257
g153
g153_1
g153_2
g258
g258_1
g154
g154_1
pobpcoac
subempleo
mto_cuota
mto_emer
mto_hogcon
mto_desay
mto_almue
mto_vacas
mto_oveja
mto_caball
lecheenpol
indaceliac
indaucc
indaemer
pt1
pt2
pt4
pobre06
indigente06
upm_id
estrato ...
toy_ech_2018_income

Details
Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Source
https://www.gub.uy/instituto-nacional-estadistica/

See Also

<table>
<thead>
<tr>
<th>toy_ech_2018_income</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A dataset containing only 1000 rows of the ECH 2018 income variables</td>
</tr>
</tbody>
</table>

Description
A dataset containing only 1000 rows of the ECH 2018 income variables

Usage
toy_ech_2018_income

Format
A data frame with 1000 rows and 9 variables:

- **numero** household id
- **mes**
- **yvsl**
- **ht11**
- **ht13**
- **ht19**
- **dpto**
- **pesoano**
- **estred13**
- **anio**
- **region_4** ...

Details
Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.
to_ascii

Source

https://www.gub.uy/instituto-nacional-estadistica/

See Also


<table>
<thead>
<tr>
<th>to_ascii</th>
<th>to_ascii</th>
</tr>
</thead>
</table>

Description

to_ascii

Usage

to_ascii(x, upper = TRUE)

Arguments

x a column

upper logic. Default TRUE

Value

vector

See Also

Other organize: organize_names()

Examples

d <- lapply(dic, to_ascii)
This function allows you to identify underemployed people

Usage

underemployment(
  data = ech::toy_ech_2018,
  pobpcoac = "pobpcoac",
  f85 = "f85",
  f98 = "f98",
  f101 = "f101",
  f102 = "f102",
  f103 = "f103",
  f104 = "f104"
)

Arguments

data data.frame
pobpcoac Variable name of definition of population by activity status. Default: "pobpcoac"
f85 Variable name of number of hours worked in the main job
f98 Variable name of Number of hours worked at the secondary job
f101 Variable name of reasons why you want another job
f102 Variable name of want to work more hours
f103 Variable name of are available to work more hours at this time
f104 Variable name of reasons why you dont work more hours

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value

data.frame

See Also

Other employment: branch_ciiu(), employment_restrictions(), employment()

Examples

toy_ech_2018 <- underemployment(data = ech::toy_ech_2018)
unlabelled

Description
This function allows you to labelled variables

Usage
unlabelled(data = NULL)

Arguments
data data frame

Value
data.frame

See Also
Other utils: archive_extract(), dates_ech(), ech, unrarPath

Examples
df <- unlabelled(data = ech::toy_ech_2018)

unrarPath

Description
The known path for unrar or 7z

Usage
.unrarPath

Format
An object of class NULL of length 0.

See Also
Other utils: archive_extract(), dates_ech(), ech, unlabelled()
unsatisfied_basic_needs

This function allows you to calculate de Unsatisfied Basic Needs

Description
This function allows you to calculate de Unsatisfied Basic Needs

Usage

unsatisfied_basic_needs(
  data = ech::toy_ech_2018,
  c2 = "c2",
  c3 = "c3",
  c4 = "c4",
  d9 = "d9",
  d11 = "d11",
  d12 = "d12",
  d13 = "d13",
  d14 = "d14",
  d15 = "d15",
  d16 = "d16",
  d18 = "d18",
  d19 = "d19",
  d21_1 = "d21_1",
  d21_2 = "d21_2",
  d21_3 = "d21_3",
  d260 = "d260",
  ht19 = "ht19",
  pobre06 = "pobre06",
  e27 = "e27",
  school_enrollment = "school_enrollment",
  years_schooling = "years_schooling",
  e238 = "e238",
  anio = "anio"
)

Arguments

data          data.frame
  c2            Variable name of predominant material on external walls
  c3            Variable name of predominant roofing material
  c4            Variable name of predominant flooring material
  d9            Variable name of number of rooms
  d11           Variable name of principal source of potable water
unsatisfied_basic_needs

d12     Variable name of water supply network / water access
d13     Variable name of sanitary facilities
d14     Variable name of bathroom presence
d15     Variable name of private bathroom use
d16     Variable name of sewerage facilities
d18     Variable name of energy source for lighting
d19     Variable name of cooking space
d21_1   Variable name of heater or termophon presence
d21_2   Variable name of instantaneous water heater presence
d21_3   Variable name of fridge presence
d260    Variable name of energy source for heating
ht19    Variable name of number of individuals in the household
pobre06 Variable name of poverty
e27     Variable name of age
school_enrollment Variable name of school_enrollment
years_schooling  Variable name of years_schooling
e238    Variable name of attendance to initial education
anio    Variable name of survey year

Details

Based on [Fascículo I: Las Necesidades Básicas Insatisfechas a partir de los Censos 2011](https://www5.ine.gub.uy/documents/Demograf)

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value
data.frame

See Also

Other poverty: `integrated_poverty_measure()`, `poverty()`

Examples

toy_ech_18 <- enrolled_school(data = ech::toy_ech_2018)
toy_ech_18 <- years_of_schooling(toy_ech_18)
toy_ech_18 <- unsatisfied_basic_needs(toy_ech_18)
urls_ine

A dataset containing the urls of INE datasets and diccionaries

Description

A dataset containing the urls of INE datasets and diccionaries

Usage

urls_ine

Format

A data frame with 9 rows and 4 variables:

yy  date from 2011 to 2019
md_sav  url for microdata download
upm_sav  url for upm download
dic  url for dictionary download

Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Source

https://www.gub.uy/instituto-nacional-estadistica/

See Also

This function allows you to calculate the years of schooling

Usage

```
years_of_schooling(
  data = ech::toy_ech_2018,
  e193 = "e193",
  e51_2 = "e51_2",
  e51_3 = "e51_3",
  e51_4 = "e51_4",
  e51_5 = "e51_5",
  e51_6 = "e51_6",
  e51_7 = "e51_7",
  e51_7_1 = "e51_7_1",
  e51_8 = "e51_8",
  e51_9 = "e51_9",
  e51_10 = "e51_10",
  e51_11 = "e51_11",
  max_years = 22
)
```

Arguments

data: data.frame
e193: Variable name of attendance school
e51_2: Variable name of years passed in primary
e51_3: Variable name of years passed in special primary
e51_4: Variable name of years passed in lower secondary
e51_5: Variable name of years passed in upper secondary
e51_6: Variable name of years passed in bachillerato tecnologico
e51_7: Variable name of years passed in technical education
e51_7_1: Variable name of technical education requirements
e51_8: Variable name of years passed in magisterio/profesorado
e51_9: Variable name of years passed in university or similar
e51_10: Variable name of years passed in tertiary (non-university)
e51_11: Variable name of years passed in postgrade
max_years: Maximum years of schooling
Details

Disclaimer: This script is not an official INE product. Aviso: El script no es un producto oficial de INE.

Value
data.frame

See Also

Other education: `enrolled_school()`, `level_completion()`, `level_education()`, `organize_educ()`

Examples

toy_ech_2018 <- years_of_schooling(data = ech::toy_ech_2018)
Index

* datasets
  cba_cbna_int, 6
  cba_cbna_mdeo, 7
  cba_cbna_rur, 7
dic, 10
ipab_base2010, 31
ipab_base2010_int, 32
ipab_base2010_mdeo, 32
ipc_base2010, 33
ipc_base2010_int, 34
ipc_base2010_mdeo, 35
toy_ech_2018, 48
toy_ech_2018_income, 64
unrarPath, 67
urls_ine, 70

* dataset
  cba_cbna_int, 6
  cba_cbna_mdeo, 7
cba_cbna_rur, 7
dic, 10
ipab_base2010, 31
ipab_base2010_int, 32
ipab_base2010_mdeo, 32
ipc_base2010, 33
ipc_base2010_int, 34
ipc_base2010_mdeo, 35
toy_ech_2018, 48
toy_ech_2018_income, 64
urls_ine, 70

* demographic
  age_groups, 3
  household_type, 23

* design
  set_design, 47

* dwelling
  housing_conditions, 24
  housing_deprivation, 25
  housing_situation, 26
  housing_tenure, 27

* overcrowding, 44

* dwnld_read
  get_microdata, 22
  read_microdata, 46

* education
  enrolled_school, 13
  level_completion, 39
  level_education, 41
  organize_educ, 42
  years_of_schooling, 71

* employment
  branch_ciui, 5
  employment, 11
  employment_restrictions, 12
  underemployment, 66

* estimation
  get_estimation_gini, 14
  get_estimation_gpg, 15
  get_estimation_mean, 16
  get_estimation_median, 17
  get_estimation_qsr, 19
  get_estimation_ratio, 20
  get_estimation_total, 21
  set_design, 47

* household_type
  household_type, 23

* income
  basket_goods, 4
  deflate, 9
  income_constant_prices, 28
  income_quantiles, 29
  labor_income_per_capita, 35
  labor_income_per_hour, 38
  organize_ht11, 43

* inference
  get_estimation_mean, 16
  get_estimation_median, 17
  get_estimation_ratio, 20
  get_estimation_total, 21
* organize
  organize_names, 43
to_ascii, 65
* poverty
  integrated_poverty_measure, 30
  poverty, 45
  unsatisfied_basic_needs, 68
* utils
  archive_extract, 4
dates_ech, 8
ech, 11
unlabelled, 67
unrarPath, 67
.unrarPath (unrarPath), 67
age_groups, 3, 23
archive_extract, 4, 8, 11, 67
basket_goods, 4, 9, 29, 30, 38, 39, 43
branch_ciui, 5, 12, 13, 66
cba_cbna_int, 6, 7, 8, 11, 31–35, 64, 65, 70
cba_cbna_mdeo, 6, 7, 8, 11, 31–35, 64, 65, 70
cba_cbna_rur, 6, 7, 7, 11, 31–35, 64, 65, 70
dates_ech, 4, 8, 11, 67
deflate, 5, 9, 29, 30, 38, 39, 43
dic, 6–8, 10, 31–35, 64, 65, 70
ech, 4, 8, 11, 67
employment, 6, 11, 13, 66
employment_restrictions, 6, 12, 12, 66
enrolled_school, 13, 40, 42, 72
get_estimation_gini, 14, 16–19, 21, 22, 47
get_estimation_gpg, 15, 15, 17–19, 21, 22, 47
get_estimation_mean, 15, 16, 16, 18, 19, 21, 22, 47
get_estimation_median, 15–17, 17, 19, 21, 22, 47
get_estimation_qsr, 15–18, 19, 21, 22, 47
get_estimation_ratio, 15–19, 20, 22, 47
get_estimation_total, 15–19, 21, 21, 47
get_microdata, 22, 46
household_type, 3, 23
housing_conditions, 24, 26–28, 45
housing_deprivation, 24, 25, 27, 28, 45
housing_situation, 24, 26, 26, 28, 45
housing_tenure, 24, 26, 27, 27, 45
income_constant_prices, 5, 9, 28, 30, 38, 39, 43
income_quantiles, 5, 9, 29, 38, 39, 43
integrated_poverty_measure, 30, 46, 69
ipab_base2010, 6–8, 11, 31, 32–35, 64, 65, 70
ipab_base2010_int, 6–8, 11, 31, 32, 33–35, 64, 65, 70
ipab_base2010_mdeo, 6–8, 11, 31, 32, 34, 35, 64, 65, 70
ipc_base2010, 6–8, 11, 31–33, 33, 34, 35, 64, 65, 70
ipc_base2010_int, 6–8, 11, 31–34, 35, 64, 65, 70
ipc_base2010_mdeo, 6–8, 11, 31–34, 35, 64, 65, 70
labor_income_per_capita, 5, 9, 29, 30, 35, 39, 43
labor_income_per_hour, 5, 9, 29, 30, 38, 39, 43
level_completion, 14, 39, 42, 72
level_education, 14, 40, 41, 42, 72
organize_educ, 14, 40, 42, 42, 72
organize_ht11, 5, 9, 29, 30, 38, 39, 43
organize_names, 43, 65
overcrowding, 24, 26–28, 44
poverty, 31, 45, 69
read_microdata, 23, 46
set_design, 15–19, 21, 22, 47
to_ascii, 44, 65
toy_ech_2018, 6–8, 11, 31–35, 48, 65, 70
toy_ech_2018_income, 6–8, 11, 31–35, 64, 65, 70
underemployment, 6, 12, 13, 66
unlabelled, 4, 8, 11, 67, 67
unrarPath, 4, 8, 11, 67, 67
unsatisfied_basic_needs, 31, 46, 68
urls_ine, 6–8, 11, 31–35, 64, 65, 70
years_of_schooling, 14, 40, 42, 71