Package 'gtaptools'

April 27, 2023

Type Package

Title A set of tools to improve the productivity of CGE modelers.

Version 0.1.0

Description The gtaptools is a package under development that aims to offer a set of functions designed for supporting simulation exercises with CGE (Computable General Equilibrium) models in R language. The primary goal of this package is to facilitate and improve file management, increase the analytical potential of the database, and provide graphical visualizations of simulation results.

Encoding UTF-8

LazyData true

Depends R (>= 3.1)

Imports data.table (>= 1.14.8),
HARr

Remotes github::USDA-ERS/MTED-HARr

RoxygenNote 7.2.3

License GPL (>= 3)

R topics documented:

`agg_har` *Aggregates headers of data in .har structure.*

Description

It aggregates variables from a .har file on disk or an object with the structure exported by the read_har function. It is possible to adopt customized weights and functions to calculate aggregations. The specification of GTAP models (...) through the *model* parameter is supported so that the respective weight variables are automatically detected according to the model being analyzed.

Usage

```
agg_har(
  input_data,
  model = NULL,
  correspondences,
  vars_weighted_mean,
  output_har_file = NULL
)
```

Arguments

input_data	It can indicate a path to a .har file or an existing object in the R environment that has the output structure of the read_har function.
model	Indicates the CGE model being worked on (Supports only GTAP, ...). For supported models, this information is sufficient to define the variables that must be aggregated through weighted mean.
correspondences	A list indicating the original sets and new aggregated sets that will be exported. It can indicate a path to a .csv file or an existing object in the R environment that has the correspondences between the *input_data* sets and the new aggregated sets. The first line will be considered as header and identifier, and must necessarily contain the same name as the set that must be aggregated from *input_data*.
vars_weighted_mean	Vector of characters relating the variables that must be grouped with a weighted average of their respective weight variables, in the format c("var" = "weight"). Please note the example section. The sets of the variable and its weight will be made compatible through the aggregation by sum of the weight variable, if necessary.
output_har_file	Output .har file name.

---

har_shape *Bind data bases and generate/change headers.*

Description

Allows the combination of different databases in data.frame or array format. Generate new variables flexibly from custom functions. Calculations can be performed between headers/variables of different dimensions/sets.

Usage

```
har_shape(
  input_data,
  new_calculated_vars = NULL,
  del_headers = NULL,
  export_sets = T,
  output_har_file = NULL
)
```

Arguments

input_data	It must consist of one or more input databases, which must be separated from each other by sublists (see example). In the case of multiple databases, all will be combined for the final output.Arrays and data.frames must be inside sublists (list(....)) as indicated in the examples section. Aggregations on input data can only be performed on single array and data.frame inputs.
new_calculated_vars	New variables resulting from custom calculations between the headers contained in input_data. The header_name[c("its sets")] format must be adopted. The new header generated by the calculation will be aggregated by sum in the sets indicated for it. Please check the examples section and the package's online manual for more details.
del_headers	Vector of characters with the names of headers that must be excluded from the output.
export_sets	If a name for a .har file is indicated, all sets will exported to that .har file. If TRUE the sets will included in output_har_file, if FALSE the sets will not be written anywhere. (default = TRUE)
output_har_file	Output .har file name.

Note

1. The calculations indicated in the new_calculated_vars variable are processed sequentially. Therefore, if the calculation for generating a new header depends on another header that will also be generated within new_calculated_vars, the second one must be defined first.

2. Ensure that the .har files adopted as input_data have an adequate structure, including the declaration of sets for each file header. It prevents the output_har_file from being recorded with errors that make it impossible for the file to be opened by the Viewhar software later.

Examples

```
# example code

# - The list_df is composed by list_df, a list of input data
#    (path to a .har database(1), data.frame(2), list of arrays(3), array(4)).

path_to_har <- gtaptools::templates("oranig_example.har")

list_db <- list(
  path_to_har, # 1 - path to .har database
  list(
    input_data = gtaptools::example_df, # 2 - data.frame
    header = quote(`1MAR`[c("COM", "SRC", "IND", "MAR")])
  ),
  gtaptools::example_arrays_har, # 3 - list of arrays
  list(
    input_data = gtaptools::example_arrays_har$XPLH, # 4 - array
    header = quote(`XPLH`[c("COM", "HOU")])
  )
)

# - calcs defines the calculations that aggregate (1),
#    solve a matrix (2) and create a header (3).
```

```
calcs <- list(
  quote(MARC["COM"] := `1MAR`), # Sums to set COM
  quote(MULT[c("REG", "HOU")] := solve(MAKE)), # Solves the matrix
  quote(NSET := c("Comm1", "Comm2")) # Creates sets
)


# - new_binded_db is a list object that combines the databases
#    contained in list_df and the calculations described in calcs.
#   Also, the "3PUR" header will not be included in the data output
#    to "gtaptools_shape_example.har", while the sets are being
#    written to "gtaptools_shape_example_sets.har"


new_binded_db <-
  gtaptools::har_shape(
    input_data = list_db,
    new_calculated_vars = calcs,
    del_headers = c("3PUR"),
    export_sets = "gtaptools_shape_example_sets.har",
    output_har_file = "gtaptools_shape_example.har"
  )
```

`squeeze_sim` *Squeeze the simulation files into a .zip file.*

Description

Scans the .cmf file and selects just essential files for the simulation and compresses them in a .zip file. It also creates a .bat file that makes it easy to run the simulation later. The files that are included are those specified in the .cmf file and that have the extension .tab, .cmf, .sti, .bat, .har, .prm, .shk, .cls, and in the case output = T, .sl4, .upd, .slc. (This function does not support dynamic simulations operationalized in the RunDynam software.)

Usage

```
squeeze_sim(cmf_file, zip_file, add_files = NULL, output = F, bat = T)
```

Arguments

cmf_file	Path to .cmf file which manages the simulation.
zip_file	Name of the .zip file that will be created.
add_files	Vector with the names or extensions of the files that will also be included in the .zip file in addition to the files mentioned in the description.
output	Includes simulation output files (default = F).
bat	Create a batch file to compile (if necessary) and run the simulation (default = T). For this functionality it is necessary to have Gempack installed.

---

summarise_header *Aggregates headers of data in .har structure.*

Description

Summarizes a single database to an array compatible format for writing to .har files.

Usage

```
summarise_header(
  input_data,
  header,
  fun = function(x) sum(x, na.rm = T),
  export_sets = T,
  output_har_file = NULL,
  output_csv_file = NULL,
  ...
)
```

Arguments

input_data	An array that has the output structure of the read_har function or a data.frame.
header	Must be adopted the format header_name[c("its sets")], where the header_name must be the same name of the numeric values column in case of a data.frame input_data, and the "its sets" must be the same name of the categorical columns of that data.frame. Please check the examples section and the package's online manual for more details.
fun	Function used for aggregation in case of non-unique values in sets (default = sum).
export_sets	If TRUE, the vectors with the set elements are incorporated with the output. If an output .har file is indicated, it will be created and exported to that .har file. If FALSE, they will not be exported.
output_har_file	Output .har file name.

`output_csv_file`
: Output .csv file name.

`...`
: Any additional arguments to be used to write the .csv file through data.table::fwrite, such as separator character (sep = ","), the character for decimal points (dec = "."), etc.

Value

An array or vector of characters (sets) structured in a compatible way to compose a .har file.

Examples

```
# example code
```

`templates` *Templates and examples*

Description

Provides the path or direct access to built-in templates and examples.

Usage

```
templates(file)
```

Arguments

`file`
: Template/example file name. Please check the file list and description below.

Note

It is important to point out that the databases and simulations included as an example in the package are only intended to support the understanding and application of the package's functionalities. Therefore, the numerical structure contained in these bases should not be used for applied research.

Below is the list of files and the description of their contents:

-"oranig_example.har" - ORANIG CGE model database for the 2015 Brazilian economy.

Examples

```
# example code

path_to_oranig <- gtaptools::templates("oranig_example.har")
path_to_oranig
```

Index