Package: panelcleaner (via r-universe)

October 12, 2024

Title An interactive interface to homogenize messy panel data into a long format

Version 0.0.5

Description Panel data structure can sometimes change over the course of data collection, which can be a real nuisance when cleaning data. This package aims to document the state of all waves of data in a comprehensive manner and then homogenize the data into a long dataset ready for data transformations.

License MIT + file LICENSE

Suggests testthat, covr, tibble, kableExtra, knitr, rmarkdown

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.1

URL `https://github.com/Global-TIES-for-Children/panelcleaner`

BugReports `https://github.com/Global-TIES-for-Children/panelcleaner/issues`

Depends R (>= 3.1)

Imports dplyr, rcoder, rlang (>= 0.4.6), glue, tidyr

VignetteBuilder knitr

Remotes nyuglobalties/rcoder

Repository https://nyuglobalties.r-universe.dev

RemoteUrl https://github.com/nyuglobalties/panelcleaner

RemoteRef HEAD

RemoteSha 4a4391dd55d1ee71b7802ee36b7bb47d893b4352

Contents

add_mapping *Add a variable mapping to a panel*

Description

Add a variable mapping to a panel

Usage

```
add_mapping(panel, mapping, ...)

## Default S3 method:
add_mapping(panel, mapping, ...)

## S3 method for class 'unhomogenized_panel'
add_mapping(panel, mapping, ...)
```

Arguments

`panel`	A panel object
`mapping`	A `data.frame` to be used to map out variables over each wave
`...`	Parameters passed onto other methods

Value

The input panel with a new mapping

Methods (by class)

- `add_mapping(default)`: The default method
- `add_mapping(unhomogenized_panel)`: Method for unhomogenized panels

`add_wave` *Add a wave of data to a panel*

Description

Adds a wave of data to a panel, assigned to wave `wave`

Usage

```
add_wave(panel, data, wave, ...)
```

Arguments

`panel`	A panel object
`data`	A `data.frame`-like object
`wave`	The wave tag associated with this dataset
`...`	Other arguments passed to methods

`add_waves` *Add multiple waves of data to a panel*

Description

Add multiple waves of data to a panel

Usage

```
add_waves(panel, data, ...)
```

Arguments

`panel`	A panel object
`data`	A list of `data.frame`-like objects, with the list names being wave tags for their respective datasets
`...`	Other arguments passed to methods

`amend_wave` *Modify a wave's dataset*

Description

Modify a wave's dataset

Usage

```
amend_wave(x, wave_id, wave_db, ...)
```

Arguments

`x`	A panel, currently an unhomogenized panel
`wave_id`	The desired wave tag
`wave_db`	The amended dataset
`...`	Other parameters passed to methods

`default_panel_mapping_schema` *View the default panel mapping schema*

Description

This schema outlines the column names used by default during panel mapping creation, when not overridden with the `.schema` parameter in `panel_mapping()`.

Usage

```
default_panel_mapping_schema()
```

`enpanel` *Collect multiple waves of data into a panel*

Description

Collect multiple waves of data into a panel

Usage

```
enpanel(name, ..., .id_col = "id", .waves_col = "wave")
```

Arguments

`name`	The name of the panel data, e.g. an assessment name
`...`	The waves of the panel. If unnamed arguments are used, the parameter order will be used as the wave tags. If named parameters are used, the names of the parameters will be used as the wave tags.
`.id_col`	The name of the column that has the observed participant IDs. Multiple names can be used if multiple columns uniquely identify participants.
`.waves_col`	The column name for the wave tags once the panel is homogenized and bound together into a single `data.frame`

Value

An `unhomogenized_panel` object

`homogenize_panel` *Homogenize all waves to consistent structure*

Description

Once all waves are collected into a single `unhomogenized_panel` object, this will homogenize variable names and, where applicable, categorical codings according to a panel mapping.

Usage

```
homogenize_panel(panel, mapping = NULL, ...)

bind_waves(panel, allow_issues = FALSE, ...)
```

Arguments

`panel`	An unhomogenized panel
`mapping`	A panel mapping. If NULL, a panel mapping must be attached to the `panel` object using `add_mapping()`
`...`	Parameters to be used for context, usually for defining a panel schema
`allow_issues`	If `TRUE`, will allow waves to be bound together even if there are identified issues. Use caution with this!

Value

An `unhomogenized_panel` that is ready to be homogenized using `bind_waves()`

Functions

- `bind_waves()`: Bind waves into a homogenized panel after successful homogenization

Homogenization Steps

The first part of the homogenization process is to harmonize wave variable names to the homogenized name. If either there are missing wave variable names and a provided homogenized name *or* provided variable names and a missing homogenized name, an error will be thrown. The original version of panelcleaner included a notion of "issues" that would allow harmonization with errors, but after continued practice of using panelcleaner, this behavior is deprecated in favor of halting the harmonization process altogether.

The next step is to harmonize the codings for categorical data. As panelcleaner was intended to be used in a data processing pipeline before analysis was conducted in Stata, the desired behavior of panelcleaner is to separate values from labels, unlike R's `factor` class. Codings are written using `rcoder::coding()`. The harmonization process is similar to names: errors will be thrown if wave codings and homogenized codings aren't both present or missing, and the codings in all waves will be recoded to the homogenized coding.

Descriptions:

The last step is to harmonize variable descriptions. This part is optional. It will only happen if the `homogenized_description` (or custom name specified with a custom panelcleaner schema) is present. The same types will occur for descriptions. The only thing different about harmonizing descriptions is that it doesn't affect the data: it operates by assigning the `bpr.description` attribute for variables. This feature is really only useful if you intend you data to be used in a blueprintr project.

Extra Parameters

In some cases the default behavior of panelcleaner is too restrictive, especially during the beginning of data collection. Often, APIs or general data exports don't include variables that don't have any submissions yet, but you still want to keep those variables in your input data. These parameters lift some restrictions on panelcleaner's behavior:

- `drop_na_homogenized`: If `TRUE`, any NA entries in the homogenized_name column will be ignored, as if the row in the panel mapping doesn't exist.

- `ignored_missing_codings`: If `TRUE`, waves with NA codings but with non-NA homogenized codings will not have their values homogenized.
- `ignored_missing_homogenized_codings`: If `TRUE`, any variables that have defined wave codings but no homogenized coding will not have their codings homogenized.
- `error_missing_raw_variables`: If `FALSE`, raw variables that should be present in the data, given the panel mapping, but aren't will not throw an error. Instead, they'll be added to the list of issues.
- `replace_missing_with_na`: If `TRUE`, raw_variables that should be present in the data, given the panel mapping, but are not will be created and filled with NA values. A message will be displayed of all the variables where this action was applied. This value supersedes `error_missing_raw_variables`.

`issues` *View homogenization issues*

Description

Some issues may be uncovered over the course of homogenization. Rather than halting execution immediately upon encountering these problems, these issues are stored within the panel object. Use this function to view the issues with a panel.

Usage

```
issues(x)
```

Arguments

`x`	A panel object

`panel_mapping` *Create a panel mapping*

Description

Create a panel mapping

Usage

```
panel_mapping(df, waves, .schema = list())
```

Arguments

`df`	A `data.frame` that has variable mapping information
`waves`	A vector of all known waves recorded in the mapping file
`.schema`	A list of column names that outline the structure of the mapping file. `default_panel_mapping_schema()` has the default settings.

wave — *Extract a wave dataset from a panel object*

Description

Gets the `data.frame` assigned to `wave_id` from a panel.

Usage

```
wave(x, wave_id, ...)
```

Arguments

`x`	A panel, currently an unhomogenized panel
`wave_id`	The wave tag to which the dataset is assigned
`...`	Other parameters passed to other methods

Index