# pyreadstat Documentation

***Release 1.2.3***

**Otto Fajardo**

**Aug 21, 2023**

# Contents:


1 Metadata Object Description 1

2 Functions Documentation 3

3 Indices and tables 5

CHAPTER 1

# Metadata Object Description

Each parsing function returns a metadata object in addition to a pandas dataframe. That object contains the following fields:

- notes: notes or documents (text annotations) attached to the file if any (spss and stata).
- column_names : a list with the names of the columns.
- column_labels : a list with the column labels, if any.
- column_names_to_labels : a dictionary with column_names as keys and column_labels as values
- file_encoding : a string with the file encoding, may be empty
- number_columns : an int with the number of columns
- number_rows : an int with the number of rows. If metadataonly option was used, it may be None if the number of rows could not be determined. If you need the number of rows in this case you need to parse the whole file. This happens for xport and por files.
- variable_value_labels : a dict with keys being variable names, and values being a dict with values as keys and labels as values. It may be empty if the dataset did not contain such labels. For sas7bdat files it will be empty unless a sas7bcat was given. It is a combination of value_labels and variable_to_label.
- value_labels : a dict with label name as key and a dict as value, with values as keys and labels as values. In the case of parsing a sas7bcat file this is where the formats are.
- variable_to_label : A dict with variable name as key and label name as value. Label names are those described in value_labels. Sas7bdat files may have this member populated and its information can be used to match the information in the value_labels coming from the sas7bcat file.
- original_variable_types : a dict of variable name to variable format in the original file. For debugging purposes.
- readstat_variable_types : a dict of variable name to variable type in the original file as extracted by Readstat.i For debugging purposes. In SAS and SPSS variables will be either double (numeric in the original app) or string (character). Stata has in addition int8, int32 and float types.
- table_name : table name (string)
- file_label : file label (SAS) (string)

- missing_ranges: a dict with keys being variable names. Values are a list of dicts. Each dict contains two keys, 'lo' and 'hi' being the lower boundary and higher boundary for the missing range. Even if the value in both lo and hi are the same, the two elements will always be present. This appears for SPSS (sav) files when using the option user_missing=True: user defined missing values appear not as nan but as their true value and this dictionary stores the information about which values are to be considered missing.
- missing_user_values: a dict with keys being variable names. Values are a list of character values (A to Z and _ for SAS, a to z for SATA) representing user defined missing values in SAS and STATA. This appears when using user_missing=True in read_sas7bdat or read_dta if user defined missing values are present.
- variable_alignment: a dict with keys being variable names and values being the display alignment: left, center, right or unknown
- variable_storage_width: a dict with keys being variable names and values being the storage width
- variable_display_width: a dict with keys being variable names and values being the display width
- variable_measure: a dict with keys being variable names and values being the measure: nominal, ordinal, scale or unknown

There are two functions to deal with value labels: set_value_labels and set_catalog_to_sas. You can read about them in the next section.

CHAPTER 2

# Functions Documentation

CHAPTER 3

# Indices and tables

- genindex
- modindex
- search