

# Importing Data

## *Getting Data into RStudio*

by Martin Frigaard

Written: October 03 2022

Updated: April 14 2023

# Materials

The slides are in the [slides.pdf](#) file

The materials for this training are in the [worksheets](#) folder:

```
worksheets/  
└─ import.Rmd
```

# Import Data

Open `import.Rmd` to follow along

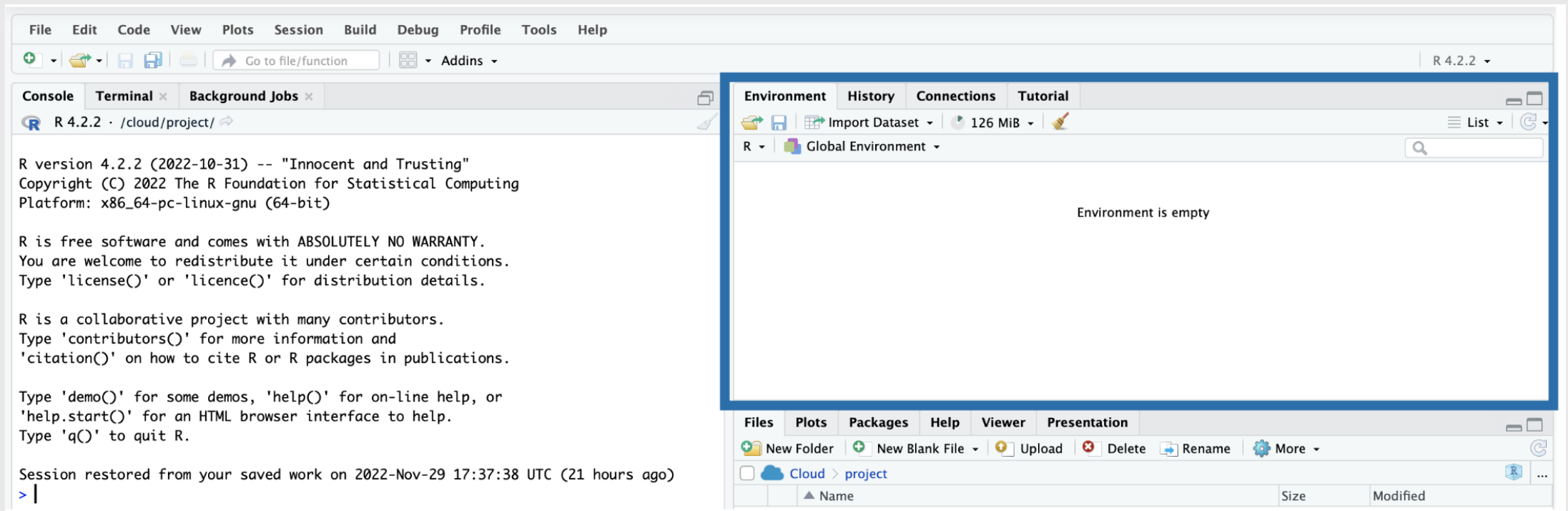
# Importing Data

## Packages for importing data:

File type	Package
SAS ( <code>.sas7bdat</code> )	<code>haven</code>
Excel ( <code>.xlsx</code> , <code>.xls</code> )	<code>readxl</code> , <code>openxlsx</code>
Plain Text ( <code>.csv</code> , <code>.tsv</code> , <code>.txt</code> )	<code>readr</code> , <code>data.table</code>

# Importing Data (*Environment*)

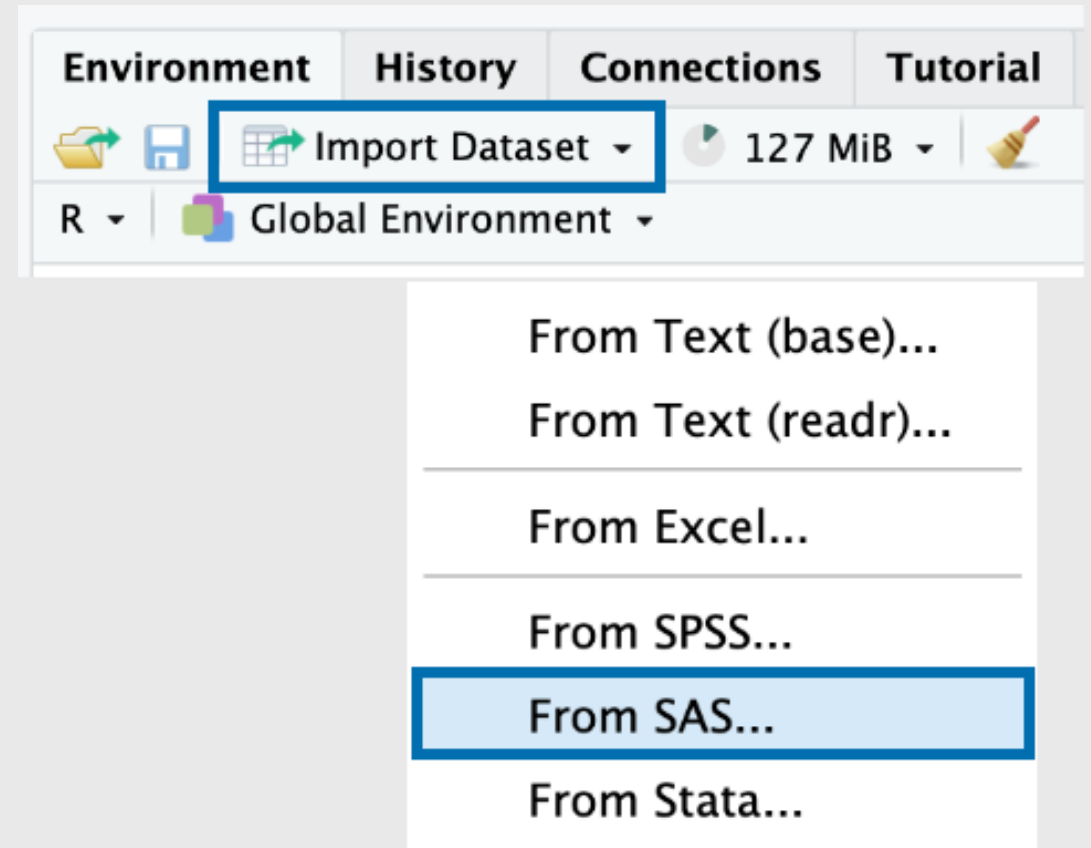
## The **Environment** Pane



# Importing Data (*Import Dataset*)

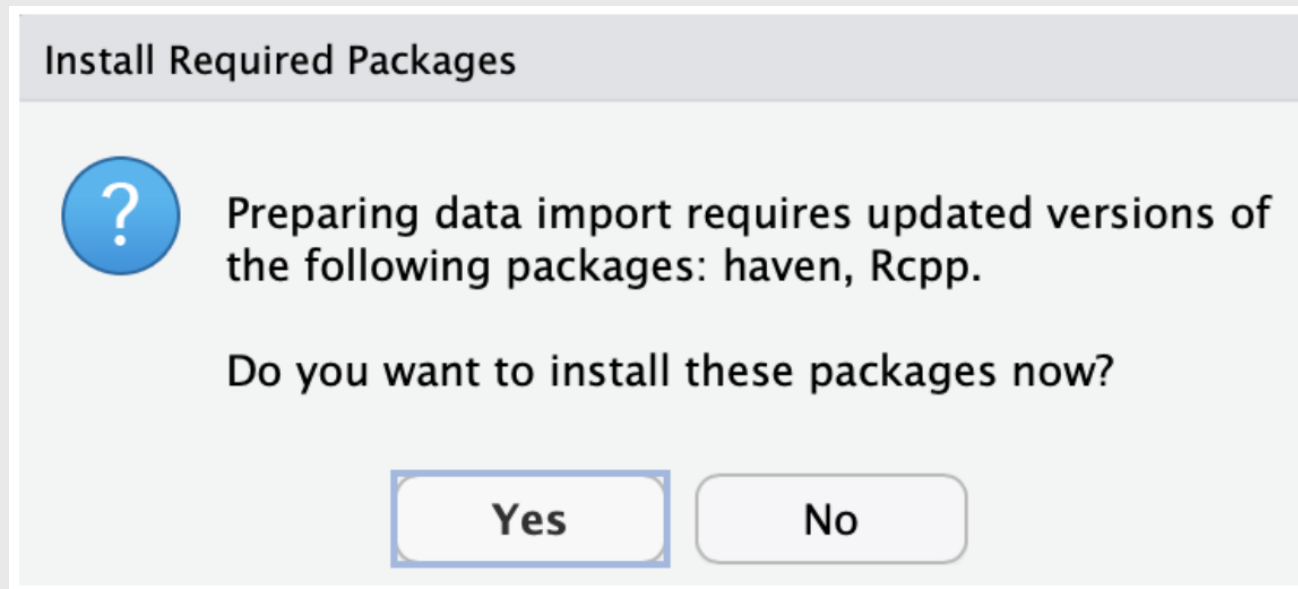
Click **Import Dataset**

Click **From SAS**



# Importing Data (*Required Packages*)

If you see a prompt to install required packages, click **Yes**



# Importing Data (*Dialogue Box*)

You will see the  
**Import Statistical  
Data Dialogue Box**

Click **Browse** and  
navigate to the  
`data/medical.sas7bdat`  
file

Import Statistical Data

File/URL:  Browse...

Data Preview:

Import Options:

Name:

Model:  Browse...

Format:  ☒ Open Data Viewer

Code Preview:

```
library(haven)
dataset <- read_sas(NULL, NULL)
View(dataset)
```

[? Reading data using haven](#) Import Cancel



# Importing Data (*Dialogue Box*)

You will see the  
path in **File/URL**

A preview of the  
data will appear in  
**Data Preview**

File/URL:

/cloud/project/data/medical.sas7bdat

Data Preview:

ID = person identifier	YEAR = year index	MEDEXP = annual medical expenditure in hundreds of dollars	INC = annual income in thousands of dollars	AGE = age in years	INSUR = 1 if individual i has private health insurance in year t and ...
1	1	9	49	51	1
1	2	9	51	52	1
1	3	9	55	53	1
1	4	10	58	54	1
1	5	11	61	55	1
2	1	6	48	62	1
2	2	7	48	63	1
2	3	7	58	64	1
2	4	7	59	65	1
2	5	7	63	66	1
3	1	4	46	57	0
3	2	3	51	58	0
3	3	5	55	59	0
3	4	4	58	60	0
3	5	4	63	61	0
4	1	5	68	48	1
4	2	3	70	49	1
4	3	6	75	50	1

Previewing first 50 entries.

# Importing Data (*Dialogue Box*)

You see we have additional **Import Options**



The screenshot shows the 'Import Options' dialog box. The 'Import Options' section on the left is highlighted with a blue border. It contains the following fields and controls:


- Name:** A text box containing the word 'medical'.
- Model:** An empty text box with a 'Browse...' button to its right.
- Format:** A dropdown menu set to 'SAS' with a small gear icon to its right.
- Open Data Viewer:** A checked checkbox.

Below these fields is a link that says '? Reading data using haven'. To the right of the 'Import Options' section is the 'Code Preview' section, which contains the following R code:

```
library(haven)
medical <- read_sas("data/medical.sas7bdat",
  NULL)
View(medical)
```

At the top right of the 'Code Preview' section is a small copy icon. At the bottom right of the dialog are 'Import' and 'Cancel' buttons.

We also see a **Code Preview**. Click on the small copy icon, then click **Import**



This screenshot is identical to the one above, but the 'Code Preview' section on the right is highlighted with a blue border. A blue arrow points from the bottom right towards the small copy icon in the top right corner of the 'Code Preview' text area.

# Importing Data (*Data Viewer*)

RStudio imports the data and opens it in the **Data Viewer**

The screenshot displays the RStudio interface with the 'medical' dataset loaded. The Data Viewer on the left shows a table with 15 rows and 4 columns: ID, YEAR, MEDEXP, and INC. The Environment pane on the right shows the 'medical' dataset with 1000 observations and 6 variables.

ID	YEAR	MEDEXP	INC
1	1	1	9
2	1	2	9
3	1	3	9
4	1	4	10
5	1	5	11
6	2	1	6
7	2	2	7
8	2	3	7
9	2	4	7
10	2	5	7
11	3	1	4
12	3	2	3
13	3	3	5
14	3	4	4
15	3	5	4

Showing 1 to 15 of 1,000 entries, 6 total columns.

# Importing Data (*Data Viewer*)

We can also see **medical** has been added to our **Environment** pane

The screenshot shows the R Studio interface. The main window displays a data table with 15 rows and 5 columns. The columns are labeled: ID (person identifier), YEAR (year index), MEDEXP (annual medical expenditure in hundreds of dollars), and INC (annual income in thousands of dollars). The first 15 rows of data are visible. The Environment pane on the right shows the 'medical' dataset, which contains 1000 observations and 6 variables. The 'medical' dataset is highlighted with a blue box.

ID	YEAR	MEDEXP	INC
1	1	1	9
2	1	2	9
3	1	3	9
4	1	4	10
5	1	5	11
6	2	1	6
7	2	2	7
8	2	3	7
9	2	4	7
10	2	5	7
11	3	1	4
12	3	2	3
13	3	3	5
14	3	4	4
15	3	5	4

Showing 1 to 15 of 1,000 entries, 5 total columns

# Importing Data

Is what we did reproducible?

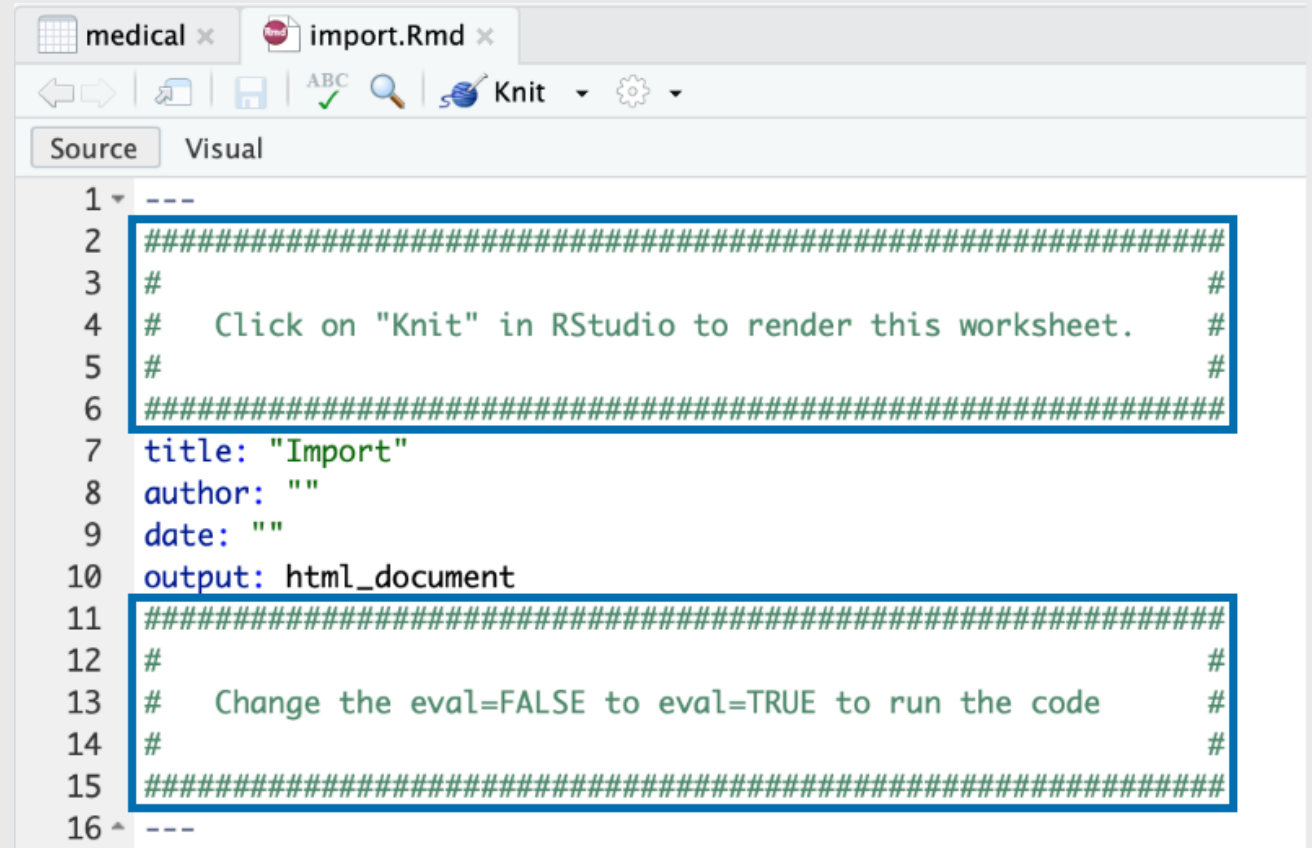
***No, but it can be!***

Open `import.Rmd` from the `worksheets` folder

# Importing Data

In `Import.Rmd`

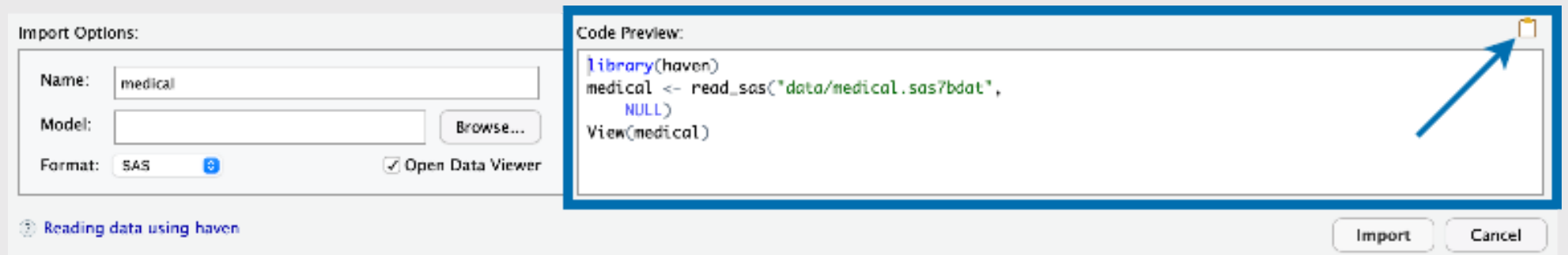
- Instructions inside `#` boxes won't run
- Fill in `author` and `date` (inside quotes)



```
1 ---
2 #####
3 #                                                                    #
4 #   Click on "Knit" in RStudio to render this worksheet.           #
5 #                                                                    #
6 #####
7 title: "Import"
8 author: ""
9 date: ""
10 output: html_document
11 #####
12 #                                                                    #
13 #   Change the eval=FALSE to eval=TRUE to run the code             #
14 #                                                                    #
15 #####
16 ---
```

# Importing Data (from local)

We already have the code to import `medical.sas7bdat` from local



Import Options:

Name:

Model:

Format:  ☒ Open Data Viewer

[Reading data using haven](#)

Code Preview:

```
library(haven)
medical <- read_sas("data/medical.sas7bdat",
  NULL)
View(medical)
```

We need to adjust the file path to `../data/medical.sas7bdat`

```
. # importing with dialogue
└─ data/
   └─ medical.sas7bdat
```

```
. # importing from file
└─ data/
   └─ medical.sas7bdat
└─ worksheets/
   └─ import.Rmd
```

# Importing Data (download and import)

We can also download the file from a `url`

```
download.file(  
  url = "http://www.principlesofeconometrics.com/sas/medical.sas7bdat",  
)
```

And save this to a local `destfile`

```
download.file(  
  url = "http://www.principlesofeconometrics.com/sas/medical.sas7bdat",  
  destfile = "../data/downloads/medical.sas7bdat")
```



# Importing Data (download and import)

Now we can import the file from our `downloads/` folder

```
. # importing from downloads folder
├── data/
│   ├── medical.sas7bdat
│   └── downloads/
│       └── medical.sas7bdat
└── worksheets/
    └── import.Rmd
```

```
medical <- read_sas("../data/downloads/medical.sas7bdat")
```

# Importing Data (parameters)

For a more permanent solution, we can use parameters in our R Markdown file to store file location (or other metadata)

```
title: "May Report"
author: "Joe Smith"
date: "2022-11-30"
output: html_document

params:
  sas_data_url: !r file.path("http://www.principlesofeconometrics.com/sas/medical.sas7bdat")
  sas_data_dir: !r c("../data/sas/")
```

```
download.file(url = params$sas_data_url,
)
```

```
download.file(url = params$sas_data_url,
  destfile = params$sas_data_dir)
```

# Importing Data (multiple files)

If we have a folder with multiple files, we can reduce duplicated code with iteration.

```
. # importing multiple files
├── data/sas/
│   ├── elemapi-2000.sas7bdat
│   ├── elemapi2-2000.sas7bdat
│   ├── hsb2.sas7bdat
│   └── nations.sas7bdat
└── worksheets/
    └── import.Rmd
```

```
# create vector of files
sas_filenames <- list.files(
  path = "../data/sas",
  full.names = TRUE)
all_sas_data <- sas_filenames |>
  # give this vector names
  purrr::set_names() |>
  # use read_sas() on all files
  purrr::map(.x = , .f = read_sas)
```

`all_sas_data` is a list of datasets

# Importing Data (multiple files)

Each named according to their path in `data/sas/`

```
str(all_sas_data)
# $ ../data/sas/elemap1-2000.sas7bdat : tibble [400 × 21] (S3: tbl_df/tbl/data.frame)
#   ..$ snum      : num [1:400] 906 889 887 876 888 ...
#   .. ..- attr(*, "label")= chr "school number"
#   ..$ dnum      : num [1:400] 41 41 41 41 41 98 98 108 108 108 ...
#   .. ..- attr(*, "label")= chr "district number"
#   .. [list output truncated]
# $ ../data/sas/elemap2-2000.sas7bdat: tibble [400 × 22] (S3: tbl_df/tbl/data.frame)
#   ..$ snum      : num [1:400] 906 889 887 876 888 ...
#   .. ..- attr(*, "label")= chr "school number"
#   ..$ dnum      : num [1:400] 41 41 41 41 41 98 98 108 108 108 ...
#   .. ..- attr(*, "label")= chr "district number"
#   .. [list output truncated]
# $ ../data/sas/hsb2.sas7bdat       : tibble [200 × 11] (S3: tbl_df/tbl/data.frame)
#   ..$ id        : num [1:200] 3 5 16 35 8 19 6 1 4 22 ...
#   ..$ female    : num [1:200] 0 0 0 1 1 1 1 1 1 0 ...
#   .. [list output truncated]
# $ ../data/sas/nations.sas7bdat   : tibble [109 × 15] (S3: tbl_df/tbl/data.frame)
#   ..$ country   : chr [1:109] "Algeria" "Argentin" "Australi" "Austria" ...
#   .. ..- attr(*, "label")= chr "Country"
#   ..$ pop       : num [1:109] 21.9 30.5 15.8 7.6 100.6 ...
#   .. ..- attr(*, "label")= chr "1985 population in millions"
#   .. [list output truncated]
```