

Package: nhanesdata (via r-universe)

May 12, 2026

Title Harmonized Access to NHANES Survey Data

Version 0.2.2

Description Instant access to harmonized National Health and Nutrition Examination Survey (NHANES) data spanning 1999-2023. Retrieve pre-processed datasets from reliable cloud storage with automatic type reconciliation and integrated search tools for variables and datasets. Simplifies NHANES data workflows by handling cycle management and maintaining data consistency across survey waves. Data is sourced from <https://www.cdc.gov/nchs/nhanes/>.

License MIT + file LICENSE

URL <https://github.com/kyleGrealis/nhanesdata>,
<https://www.kylegrealis.com/nhanesdata/>

BugReports <https://github.com/kyleGrealis/nhanesdata/issues>

Language en-US

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Depends R (>= 4.1.0)

Imports arrow, dplyr, nhanesA, rlang, scales, stringr, srvyr, tibble

Suggests cli, curl, devtools, fs, ggplot2, httptest2, janitor,
jsonlite, knitr, paws.storage, purrr, reactable, rmarkdown,
roxygen2, testthat (>= 3.0.0), tools, withr, yaml

Config/testthat/edition 3

VignetteBuilder knitr

Config/pak/sysreqs cmake make libicu-dev libxml2-dev libssl-dev

Repository <https://kylegrealis.r-universe.dev>

Date/Publication 2026-05-12 16:04:44 UTC

RemoteUrl <https://github.com/kylegrealis/nhanesdata>

RemoteRef HEAD

RemoteSha 4788f8ed73be443018597d1a24a30f1429dd6855

Contents

create_design	2
get_url	4
read_nhanes	5
term_search	6
var_search	7

Index	8
--------------	----------

create_design	<i>Calculate survey design weight within a NHANES dataset</i>
---------------	---

Description

Input an NHANES dataset and apply the proper weight calculation. There are 3 categories of weights:

1. Interview weight
2. Mobile Exam Center (MEC) weight
3. Fasting weight

The probability of being sampled for each type of NHANES category decreases from interview to fasting samples. Therefore, when selecting the proper weight, the practitioner should use the weight with the lowest probability when combining variables across categories. For example, when performing an analysis using demographics (interview), diabetes information (interview), and DEXA scanning (MEC), the associated MEC weight is the proper weight variable to use.

It is also important to select the proper year grouping for the cycle. NHANES cycles for 1999 and 2001 use 4-year sample weights, while all subsequent cycles use 2-year sample weights. This type of combination requires careful attention to:

1. The variables used, to determine weight category (interview, MEC, fasting).
2. The cycles (years) used, to select proper year grouping variable.

This function will allow the user to input a dataset, select analysis start & end years, and specify the type of weight category. The resulting survey design will calculate the proper weight and apply that when creating the design object.

NOTE: It is **not** required to specify variables for this function and it is **highly recommended** to perform preprocessing of variables *before* creating a complex design object.

See also [as_survey_design](#)

Usage

```
create_design(
  dsn,
  start_yr,
  end_yr,
  wt_type = c("interview", "mec", "fasting")
)
```

Arguments

dsn	Tibble or data-frame.
start_yr	Numeric. Lower bound for year filtering (inclusive). Must be an odd year representing a valid NHANES cycle start: 1999, 2001, 2003, ..., 2019, 2021. For example, use 2007 for the 2007-2008 cycle. Data will be filtered to include years between start_yr and end_yr.
end_yr	Numeric. Upper bound for year filtering (inclusive). Must be an odd year \geq start_yr. Weight calculations are based on the number of cycles actually present in the filtered data, so it is valid to have gaps (e.g., start_yr=1999, end_yr=2017 with 2007-2010 missing).
wt_type	Character. Category of weight to be used. Use the weight category with the lowest probability of selection, but only if at least one variable from that category is to be used. Accepts full names ("interview", "mec", "fasting") or abbreviations ("int", "mec", "fast").

Details

Weight Calculation for Combined Cycles

NHANES provides 4-year weights for the 1999-2000 and 2001-2002 cycles, while all subsequent cycles provide only 2-year weights. When combining multiple cycles:

- If 1999 or 2001 cycles are included: Use the 4-year weight variable multiplied by $2/n$ where n is the total number of cycles. The numerator is 2 because the 4-year weight represents two 2-year cycles.
- For cycles 2003 and beyond: Use the 2-year weight variable multiplied by $1/n$.
- The denominator n is always the total number of cycles in the analysis.

Example: Combining 4 cycles (1999, 2001, 2003, 2005):

- 1999 & 2001: $\text{wtmec4yr} * 2/4$
- 2003 & 2005: $\text{wtmec2yr} * 1/4$

Fasting weights follow the same rules: $\text{wtsaf4yr} * 2/n$ for 1999/2001 and $\text{wtsaf2yr} * 1/n$ for 2003+.

Fasting Subsample Weights

For fasting subsample analyses combining 1999-2002 cycles, the 4-year fasting weight (WTSAF4YR) exists in laboratory files (e.g., LAB10AM, LAB13AM) but is typically NOT in demographic files obtained via nhanesA. If your dataset includes 1999 or 2001 cycles with fasting data, ensure WTSAF4YR is present in the merged dataset. The function will validate its presence and apply the same $2/n$ scaling used for other 4-year weights.

Value

A survey design object of class `tbl_svy` (from `srvyr` package) containing the calculated design weights and survey design metadata (PSUs, strata). Participants without valid weights for the specified weight type are automatically filtered out before design object creation. Participants with zero weights are retained in the design object but will be automatically excluded from most survey analyses.

Examples

```
# Load demographics data
demo <- read_nhanes("demo")

# Create design object with interview weights
design <- create_design(
  dsn = demo,
  start_yr = 1999,
  end_yr = 2011,
  wt_type = "interview"
)

# Combine with examination data and use MEC weights
bmx <- read_nhanes("bmx")
combined <- demo |>
  dplyr::left_join(bmx, by = c("seqn", "year"))

design_mec <- create_design(
  dsn = combined,
  start_yr = 2007,
  end_yr = 2017,
  wt_type = "mec"
)
```

get_url

Get CDC Documentation URL for NHANES Table

Description

Constructs and returns the full CDC documentation URL for a given NHANES table. The function handles table names with or without cycle suffixes (e.g., "DEMO_J" for 2017-2018 or "DEMO" for 1999-2000) and automatically maps the suffix to the appropriate survey cycle year.

Usage

```
get_url(table)
```

Arguments

table	Character. The table where variable information is needed. Can include cycle suffix (e.g., "DEMO_J") or not (e.g., "DEMO"). Not case-sensitive.
-------	---

Value

Character string (invisibly). Full URL to CDC data documentation, codebook, and frequencies is returned invisibly and also printed to the console via message() for interactive use.

See Also

[term_search](#), [var_search](#)

Other search and lookup functions: [term_search\(\)](#), [var_search\(\)](#)

Examples

```
# These examples will run and display URLs
get_url("DEMO_J") # Demographics 2017-2018
get_url("diq_j") # Case-insensitive: Diabetes 2017-2018
get_url("DIQ") # No suffix = 1999-2000 cycle
```

read_nhanes

Read NHANES Data from Cloud Storage

Description

Downloads pre-processed NHANES data files from cloud storage. Data includes all survey cycles (1999-2023) automatically merged and harmonized, with annual updates.

Usage

```
read_nhanes(dataset)
```

Arguments

dataset Character. NHANES dataset base name (e.g., "trigly", "demo"). **Case-insensitive** - use 'demo', 'DEMO', or 'Demo' interchangeably. Must be a single string (length 1). Leading/trailing whitespace is automatically trimmed.

Details

This function downloads NHANES datasets from cloud storage (hosted at nhanes.kylegrealis.com). All datasets combine multiple survey cycles with automatic type harmonization. Data is updated annually via automated workflows that pull fresh data from CDC servers.

Dataset names are case-insensitive throughout this package. Use uppercase (matches CDC documentation) or lowercase (easier to type) - both work identically.

Error handling: The function validates inputs and provides informative error messages if the dataset fails to load (e.g., network issues, non-existent datasets, misspelled names). Error messages include the attempted URL and suggestions for troubleshooting.

Value

A tibble containing the requested NHANES dataset across all available survey cycles. Always includes year and seqn columns plus dataset-specific variables.

Examples

```
# All case variations work identically:
trigly <- read_nhanes("trigly") # Lowercase
demo <- read_nhanes("DEMO") # Uppercase
acq <- read_nhanes("Acq") # Mixed case
```

term_search	<i>Search NHANES Variables by Term or Phrase</i>
-------------	--

Description

A convenience wrapper around `nhanesA::nhanesSearch` that returns a simplified, concise output focused on variable names, descriptions, and survey years. Results are sorted by year (most recent first) and then by variable name.

Usage

```
term_search(var)
```

Arguments

var	Character. Search term or phrase to find in variable names or descriptions. Case-insensitive. Special regex characters are automatically escaped for literal matching.
-----	--

Value

A data.frame with 4 columns:

- `Variable.Name`: NHANES variable code
- `Variable.Description`: Description of the variable
- `Data.File.Name`: Name of the data file containing the variable
- `Begin.Year`: Starting year of the survey cycle (numeric)

Results are sorted by `Begin.Year` (descending) then `Variable.Name`. Returns an empty data.frame with correct structure if no matches found.

See Also

[var_search](#) for searching by exact variable name, [get_url](#) for getting documentation URLs, [nhanesSearch](#) for the underlying search function

Other search and lookup functions: [get_url\(\)](#), [var_search\(\)](#)

Examples

```
# Search for diabetes-related variables (showing first 5 results)
term_search("diabetes") |> head(5)

# Search for blood pressure measurements (showing first 5 results)
term_search("blood pressure") |> head(5)
```

var_search	<i>Search for NHANES Variable by Exact Name</i>
------------	---

Description

A convenience wrapper around `nhanesA::nhanesSearchVarName` that searches for variables by exact variable name match. The function automatically converts input to uppercase to match NHANES naming conventions. Use this when you know the variable code; use `term_search()` for text-based searches.

Usage

```
var_search(var)
```

Arguments

`var` Character. Variable name to search for. Will be automatically converted to uppercase. Not case-sensitive.

Value

A character vector of CDC table names containing the variable (e.g., "DEMO", "DEMO_B", "DEMO_C"). Returns character(0) if the variable is not found.

See Also

[term_search](#) for text-based searches, [get_url](#) for documentation URLs, [nhanesSearchVarName](#) for the underlying function

Other search and lookup functions: [get_url\(\)](#), [term_search\(\)](#)

Examples

```
# Search for specific variable (case-insensitive)
var_search("RIDAGEYR") # Age variable across all DEMO cycles
var_search("BPXSY1") # Systolic blood pressure
```

Index

* search and lookup functions

[get_url](#), 4

[term_search](#), 6

[var_search](#), 7

[as_survey_design](#), 2

[create_design](#), 2

[get_url](#), 4, 6, 7

[nhanesSearch](#), 6

[nhanesSearchVarName](#), 7

[read_nhanes](#), 5

[term_search](#), 5, 6, 7

[var_search](#), 5, 6, 7