

Accessing NHANES Data

Below you can find instructions for installing and running the Docker container for NHANES data, created by the CCB.

Note that these instructions go into how to:

1. Install Docker/Docker Desktop.
2. Download the docker image.
3. Create and run the docker container.
4. Connect to the running docker container.

However, they do not go into how to actually access the NHANES data once connected to the container. We will be learning how to do this in class.

Through the Docker container (reccomended)

These steps will allow you to access the NHANES data via a builtin

Connect to RStudio in your browser as shown in the [Accessing NHANES with Docker](#) page.

Installing Phonto

In the R console, run the command:

```
devtools::install_github("ccb-hms/phonto")
```

If successful you should see output similar to:

```
Downloading GitHub repo ccb-hms/phonto@HEAD
```

```
checking for file '/tmp/RtmpjKcsPx/remotes8977d6e909/ccb-hms-phonto-121d255/DESCRIPTION' .
```

```
preparing 'phonto':
```

```
checking DESCRIPTION meta-information
```

```
checking for LF line-endings in source and make files and shell scripts
```

```
checking for empty or unneeded directories
```

```
building 'phonto_0.0.0.0069.tar.gz'
```

```
* installing *source* package 'phonto' ...
```

```
** using staged installation
```

```
** R
```

```
** data
```

```
*** moving datasets to lazyload DB
```

```
** inst
** byte-compile and prepare package for lazy loading
EpiConductor Container Version: v0.0.4
Data Collection Date: 2023-06-28
** help
*** installing help indices
** building package indices
** installing vignettes
** testing if installed package can be loaded from temporary location
** testing if installed package can be loaded from final location
** testing if installed package keeps a record of temporary installation path
* DONE (phonto)
```

Timeout Errors

You may get a timeout error such as this:

```
> devtools::install_github("ccb-hms/phonto")
Error: Failed to install 'phonto' from GitHub:
  Timeout was reached: [api.github.com] Resolving timed out after 10000 milliseconds
```

Or a longer error related to login timing out like this:

```

> devtools::install_github("ccb-hms/phonto")
Downloading GitHub repo ccb-hms/phonto@HEAD
  checking for file '/tmp/RtmpjKcsPx/remotes895e1b4f91/ccb-hms-phonto-121d255/DESCRIPTION'
  preparing 'phonto':
  checking DESCRIPTION meta-information ...
  checking for LF line-endings in source and make files and shell scripts
  checking for empty or unneeded directories
  building 'phonto_0.0.0.0069.tar.gz'

* installing *source* package 'phonto' ...
** using staged installation
** R
** data
*** moving datasets to lazyload DB
** inst
** byte-compile and prepare package for lazy loading
Error : nanodbc/nanodbc.cpp:1021: 00000: [Microsoft][ODBC Driver 17 for SQL Server]Login
Error: unable to load R code in package 'phonto'
Execution halted
ERROR: lazy loading failed for package 'phonto'
* removing '/usr/local/lib/R/library/phonto'
Warning message:
In i.p(...) :
  installation of package '/tmp/RtmpjKcsPx/file8930842adb/phonto_0.0.0.0069.tar.gz' had

```

If you get one of these errors, please retry the installation command. It may take 2-3 tries for the docker container to establish a connection with Github.

Through nhanesA

With `nhanesA`, we can easily download entire tables from NHANES. However, there are some extra processing steps we'll have to perform compared to using the dockerized database. You can learn more about using `nhanesA` [here](#).

```

library(nhanesA)
library(tidyverse)

```

```

-- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
v dplyr      1.1.2      v readr      2.1.4

```

```

v forcats    1.0.0      v stringr    1.5.0
v ggplot2    3.4.2      v tibble     3.2.1
v lubridate  1.9.2      v tidyr      1.3.0
v purrr      1.0.1
-- Conflicts ----- tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()     masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become

```

```

library(DT)
# Get data with nahnesA
DEMO_H = nhanes('DEMO_H')
DEMO_I = nhanes('DEMO_I')
DPQ_H = nhanes('DPQ_H')
DPQ_I = nhanes('DPQ_I')

# Append Files
DEMO <- bind_rows(DEMO_H, DEMO_I)
DPQ <- bind_rows(DPQ_H, DPQ_I)

datatable(head(DEMO))

```

PhantomJS not found. You can install it with `webshot::install_phantomjs()`. If it is installed

Show entries

Search:

	SEQN ⚙	SDDSRVYR ⚙	RIDSTATR ⚙	RIAGENDR ⚙	RIDAGEYR ⚙	RIDAGEMN ⚙	RIDRETH1 ⚙	RIDRETH3 ⚙	RIDEXMC
1	73557	8	2	1	69		4	4	
2	73558	8	2	1	54		3	3	
3	73559	8	2	1	72		3	3	
4	73560	8	2	1	9		3	3	
5	73561	8	2	2	73		3	3	
6	73562	8	2	1	56		1	1	

Showing 1 to 6 of 6 entries

Previous Next

```
datatable(head(DPQ))
```

Show entries

Search:

	SEQN	DPQ010	DPQ020	DPQ030	DPQ040	DPQ050	DPQ060	DPQ070	DPQ080	DPQ090	DPQ100
1	73557	1	0	0	0	0	0	0	0	0	1
2	73558	2	0	0	0	0	0	0	0	0	0
3	73559	0	0	0	0	0	0	0	0	0	
4	73561	2	1	0	3	3	0	0	0	0	1
5	73562	3	3	3	3	3	1	2	1	0	3
6	73564	0	1	0	1	0	0	0	0	0	0

Showing 1 to 6 of 6 entries

Previous Next

There are a few differences between this data and the processed data we've been using. First, we have to join the DEMO and DPQ tables. We'll learn more about joining or merging data in the last week of class.

Second, the values in the raw NHANES tables are numeric encodings for each variable. With `nhanesA` we can lookup the code using `nhanesCodebook` and convert numeric codes using `nhanesTranslate`.

```
nhanesCodebook('DEMO_H', 'RIAGENDR')
```

```
$`Variable Name:`
```

```
[1] "RIAGENDR"
```

```
$`SAS Label:`
```

```
[1] "Gender"
```

```
$`English Text:`
```

```
[1] "Gender of the participant."
```

```
$`Target:`
```

```
[1] "Both males and females 0 YEARS -\r 150 YEARS"
```

```
$RIAGENDR
```

```
# A tibble: 3 x 5
```

	`Code or Value`	`Value Description`	Count	Cumulative	`Skip to Item`
	<chr>	<chr>	<int>	<int>	<lgl>
1	1	Male	5003	5003	NA
2	2	Female	5172	10175	NA
3	.	Missing	0	10175	NA

```
nhanesTranslate(DEMO_H)
```

```
Column name is required
```

```
NULL
```

Downloading individual files

If all else fails, individual files can be downloaded from the CDC website and read into R using the [foreign](#) package. This example is taken from an example analysis put out by the CDC [here](#).


```

#' Prevalence of Depression Among Adults Aged 20 and Over: United States, 2013-2016
#' Brody DJ, Pratt LA, Hughes JP. Prevalence of Depression Among Adults Aged 20 and Over:
#' States, 2013-2016. NCHS Data Brief. No 303. Hyattsville, MD: National Center for Health

#' # Data preparation
# Download & Read SAS Transport Files
# Demographic (DEMO)
download.file("https://wwwn.cdc.gov/nchs/nhanes/2013-2014/DEMO_H.XPT", tf <- tempfile(), m
DEMO_H <- foreign::read.xport(tf)[,c("SEQN", "RIAGENDR", "RIDAGEYR", "SDMVSTRA", "SDMVPSU", "WT
download.file("https://wwwn.cdc.gov/nchs/nhanes/2015-2016/DEMO_I.XPT", tf <- tempfile(), m
DEMO_I <- foreign::read.xport(tf)[,c("SEQN", "RIAGENDR", "RIDAGEYR", "SDMVSTRA", "SDMVPSU", "WT

# Mental Health - Depression Screener (DPQ)
download.file("http://wwwn.cdc.gov/nchs/nhanes/2013-2014/DPQ_H.XPT", tf <- tempfile(), mod
DPQ_H <- foreign::read.xport(tf)
download.file("http://wwwn.cdc.gov/nchs/nhanes/2015-2016/DPQ_I.XPT", tf <- tempfile(), mod
DPQ_I <- foreign::read.xport(tf)

```