Accessing NHGIS Data in R

Finn Roberts

Senior Data Analyst, IPUMS NHGIS

2024-03-05

IPUMS NHGIS

Zoom Logistics

- Webinar is being recorded and will be posted
- Real-time closed captions are being auto-generated
  - Turn on/off by clicking “CC” button in Zoom controls
- Send questions about Zoom directly to host (IPUMS)
- Submit content questions with Q&A tool
- Will post written Q&A document following webinar

Roadmap

1. Overview of IPUMS NHGIS

2. ipumsr + IPUMS API setup

3. A motivating example

4. Query NHGIS metadata

5. Define, submit, and download an extract

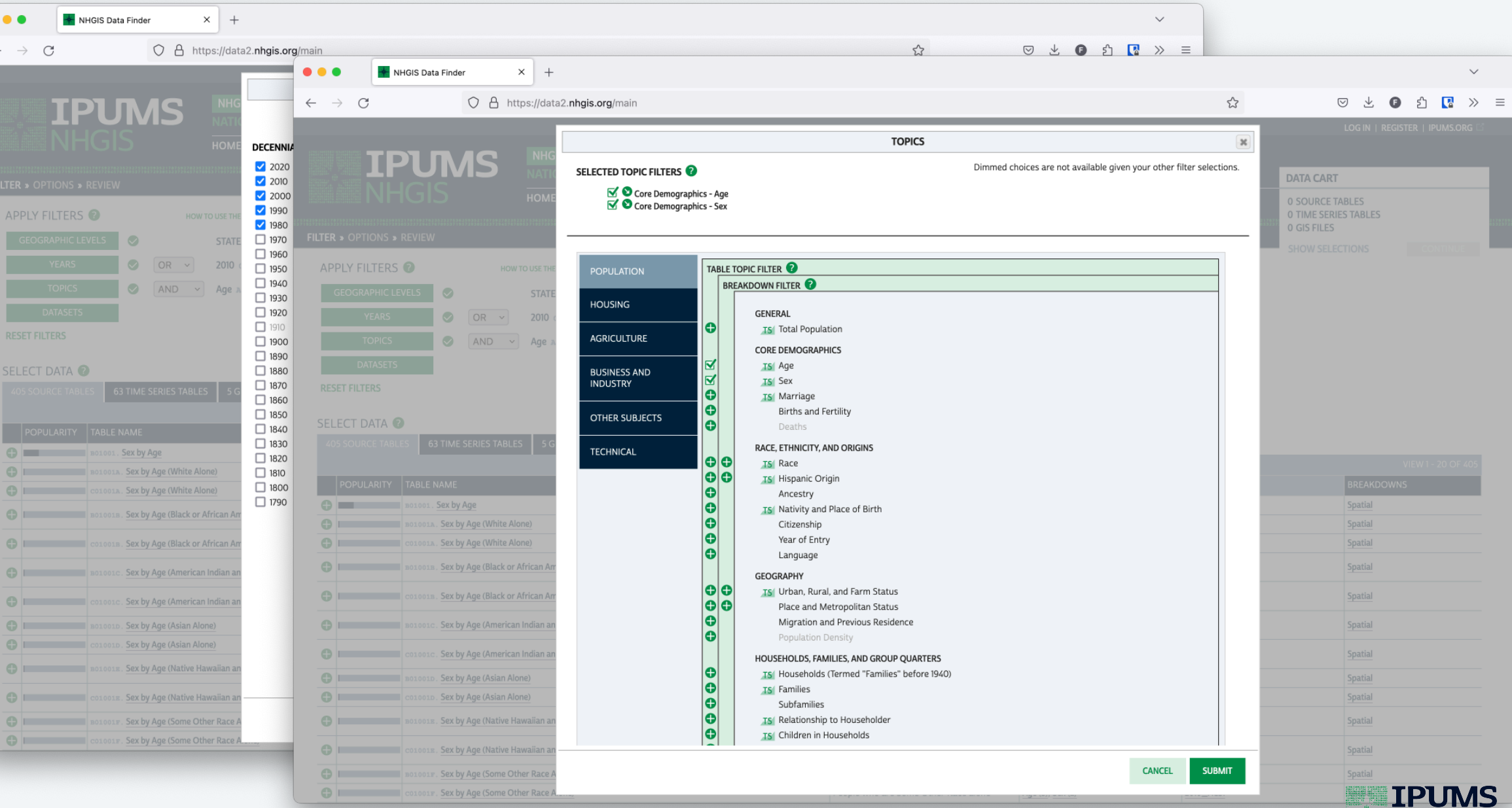
6. An iterative workflow

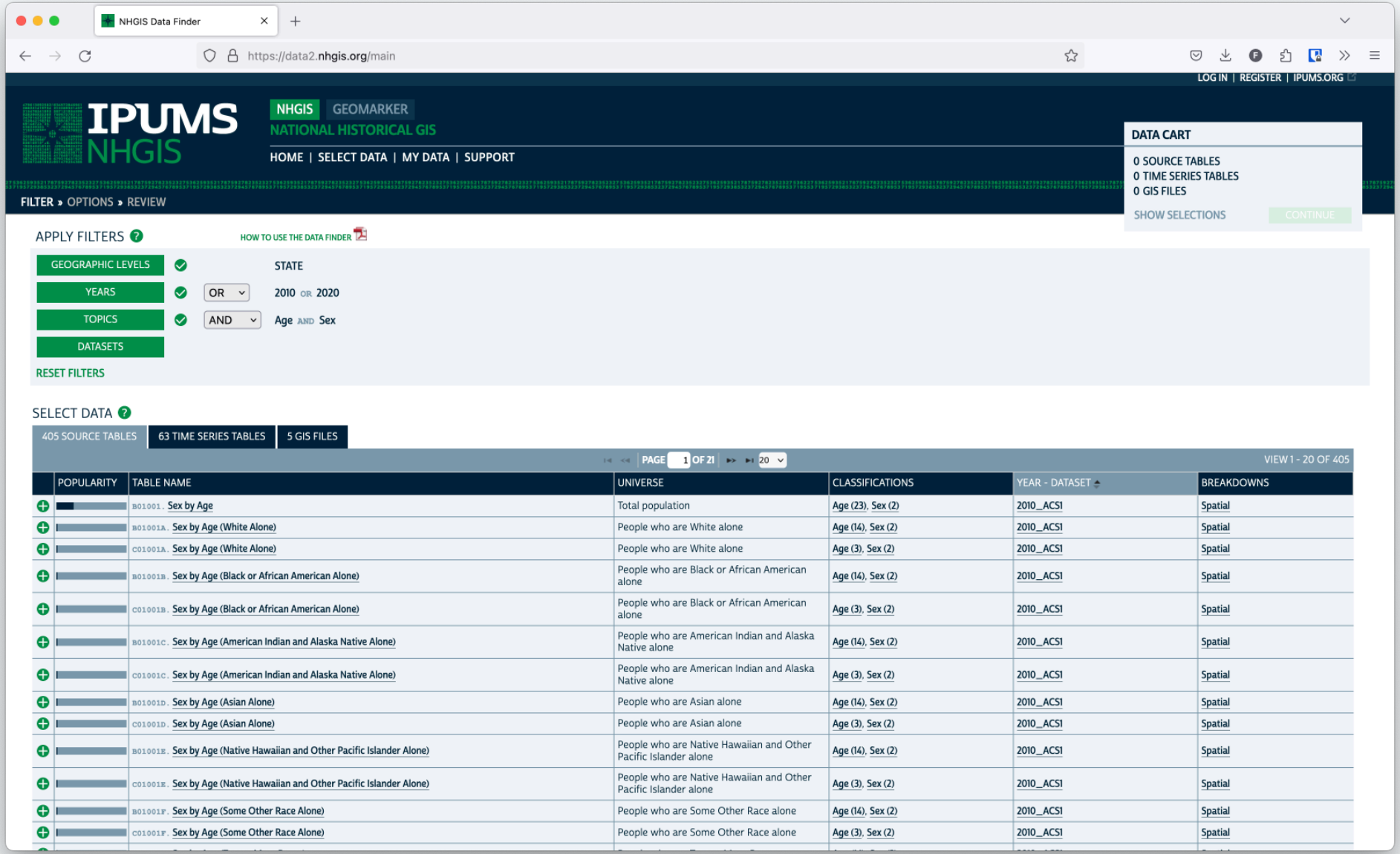
IPUMS NHGIS

- Statistical summary tables for the U.S.
- Data from 1790-present
- Data sources:
  - Datasets + data tables
  - Time series tables
  - Shapefiles

IPUMS NHGIS

Accessing NHGIS Data: Traditionally




NHGIS Data Finder
https://data2.nhgis.org/main
LOG IN | REGISTER | IPUMS.ORG
IPUMS NHGIS
NHGIS
GEOMARKER
NATIONAL HISTORICAL GIS
HOME | SELECT DATA | MY DATA | SUPPORT
DATA CART
0 SOURCE TABLES
0 TIME SERIES TABLES
0 GIS FILES
SHOW SELECTIONS
CONTINUE
FILTER » OPTIONS » REVIEW
APPLY FILTERS
HOW TO USE THE DATA FINDER
GEOGRAPHIC LEVELS
STATE
YEARS
OR
2010 OR 2020
TOPICS
AND
Age AND Sex
DATASETS
RESET FILTERS
SELECT DATA
405 SOURCE TABLES
63 TIME SERIES TABLES
5 GIS FILES
PAGE 1 OF 21
20
VIEW 1 - 20 OF 405
POPULARITY
TABLE NAME
UNIVERSE
CLASSIFICATIONS
YEAR - DATASET
BREAKDOWNS
B01001. Sex by Age
Total population
Age (23), Sex (2)
2010_ACS1
Spatial
B01001A. Sex by Age (White Alone)
People who are White alone
Age (14), Sex (2)
2010_ACS1
Spatial
C01001A. Sex by Age (White Alone)
People who are White alone
Age (3), Sex (2)
2010_ACS1
Spatial
B01001B. Sex by Age (Black or African American Alone)
People who are Black or African American alone
Age (14), Sex (2)
2010_ACS1
Spatial
C01001B. Sex by Age (Black or African American Alone)
People who are Black or African American alone
Age (3), Sex (2)
2010_ACS1
Spatial
B01001C. Sex by Age (American Indian and Alaska Native Alone)
People who are American Indian and Alaska Native alone
Age (14), Sex (2)
2010_ACS1
Spatial
C01001C. Sex by Age (American Indian and Alaska Native Alone)
People who are American Indian and Alaska Native alone
Age (3), Sex (2)
2010_ACS1
Spatial
B01001D. Sex by Age (Asian Alone)
People who are Asian alone
Age (14), Sex (2)
2010_ACS1
Spatial
C01001D. Sex by Age (Asian Alone)
People who are Asian alone
Age (3), Sex (2)
2010_ACS1
Spatial
B01001E. Sex by Age (Native Hawaiian and Other Pacific Islander Alone)
People who are Native Hawaiian and Other Pacific Islander alone
Age (14), Sex (2)
2010_ACS1
Spatial
C01001E. Sex by Age (Native Hawaiian and Other Pacific Islander Alone)
People who are Native Hawaiian and Other Pacific Islander alone
Age (3), Sex (2)
2010_ACS1
Spatial
B01001F. Sex by Age (Some Other Race Alone)
People who are Some Other Race alone
Age (14), Sex (2)
2010_ACS1
Spatial
C01001F. Sex by Age (Some Other Race Alone)
People who are Some Other Race alone
Age (3), Sex (2)
2010_ACS1
Spatial

A new way: ipumsr

- ipumsr v0.6.0+ (July 2023)
- New interface to access NHGIS data **entirely within R**
  - Browse NHGIS metadata
  - Define an extract request
  - Submit request to the IPUMS servers
  - Download extract data
  - Load data into R*

Why use ipumsr?

1. **Reproducibility**
    - Easily regenerate identical extract definitions
    - Share extract definitions with collaborators + reviewers

2. **Flexibility**
    - Quickly update extract definitions to add new data sources
    - Comprehensive access to NHGIS metadata

3. **Automation**
    - Build data retrieval workflows directly into your analysis
    - For example: query metadata to identify when new datasets become available

First-time setup

First, install ipumsr if you haven't already:

```
install.packages("ipumsr")
```

Then, load the library:

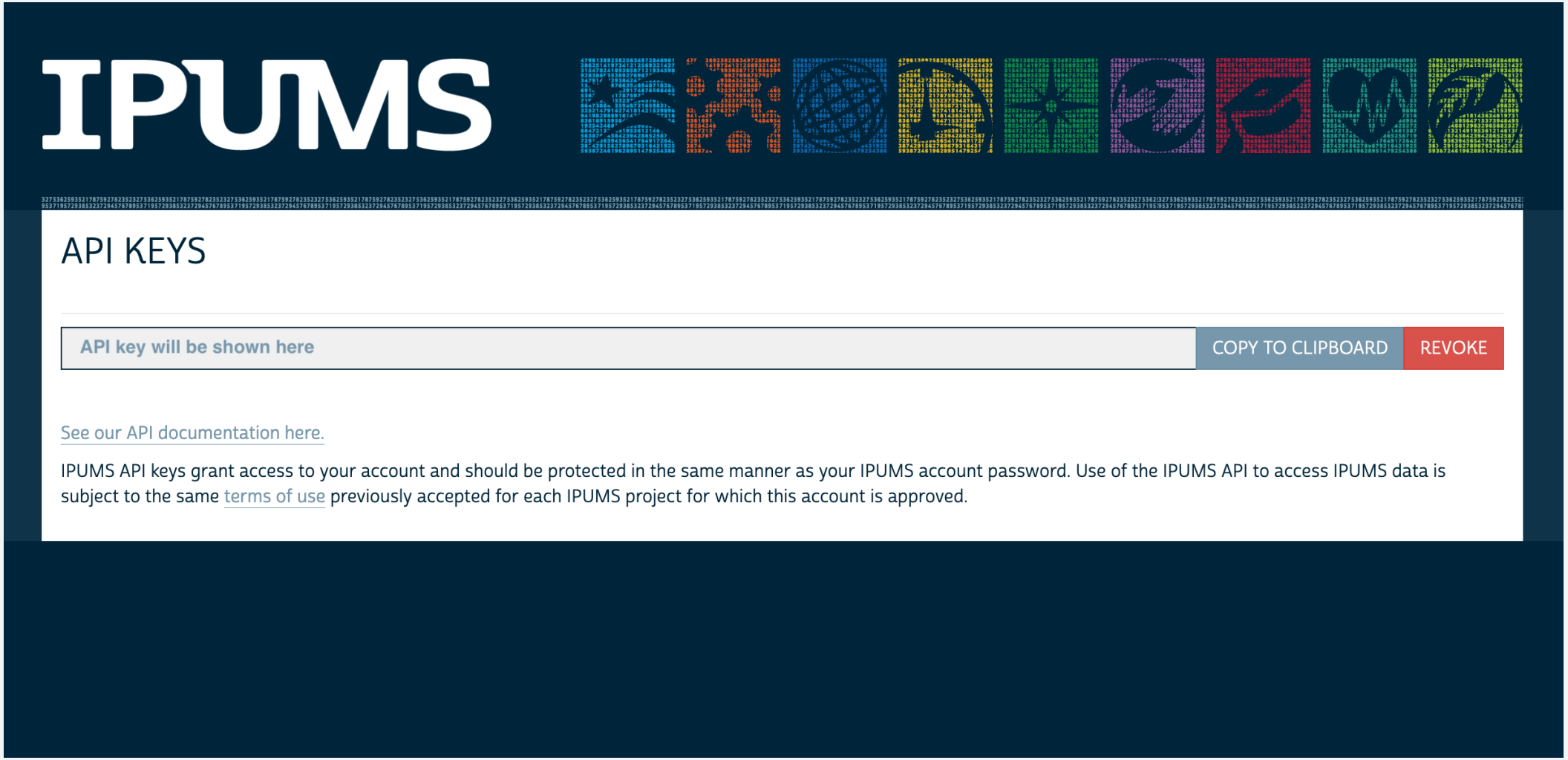
```
library(ipumsr)
```

Get an IPUMS API Key

- ipumsr extract request/metadata functionality is built on top of the **IPUMS API**

- You'll need an API Key to get started
  - Log into your IPUMS NHGIS account
  - https://account.ipums.org/api_keys

Get an IPUMS API Key



Get an IPUMS API Key

- Save your key in your R environment:

```
set_ipums_api_key("paste-your-api-key-here", save = TRUE)
```

IPUMS NHGIS

Let’s get started!

Motivating example

HSR Health Services Research

METHODS ARTICLE | Open Access

Multidimensional structural racism predicts birth outcomes for Black and White Minnesotans

Tongtan Chantarat PhD, MPH, David C. Van Riper MA, Rachel R. Hardeman PhD, MPH

First published: 25 April 2022 | https://doi.org/10.1111/1475-6773.13976 | Citations: 10

See related debate-commentary by **Brown et al**.

Funding information: Eunice Kennedy Shriver National Institute of Child Health and Human Development, Grant/Award Number: P2C HD041023; Robert J. Jones Urban Research and Outreach-Engagement Center, University of Minnesota

IPUMS NHGIS

Motivating example

- **Goal:** examine racial inequities in homeownership rates
- **Data:** 2017 ACS 5-year
- **Geography:** County
- Method:
  - Calculate proportion of owner-occupied households to total households for each race/ethnicity group
  - Calculate ratio of homeowner proportions between non-Hispanic white households and other race/ethnicity households

Motivating example

- Recall the three different data products NHGIS provides:
  - Datasets + data tables
  - Time series tables
  - Shapefiles

Motivating example

- Recall the three different data products NHGIS provides:
  - **Datasets + data tables**
  - Time series tables
  - Shapefiles
- Need to find relevant data tables for homeownership inequity

Search metadata

- NHGIS provides summary metadata and detailed metadata
- *Summary* metadata contains general information about all available datasets, data tables, time series tables, and shapefiles

```
nhgis_ds <- get_metadata_nhgis("datasets")
```

Search metadata

- NHGIS provides summary metadata and detailed metadata
- *Summary* metadata contains general information about all available datasets, data tables, time series tables, and shapefiles

```
nhgis_ds <- get_metadata_nhgis("datasets")

nhgis_ds
#> # A tibble: 261 × 4
#>    name       group       description                                    sequence
#>    <chr>      <chr>       <chr>                                             <int>
#>  1 1790_cPop  1790 Census Population Data [US, States & Counties]             101
#>  2 1800_cPop  1800 Census Population Data [US, States & Counties]             201
#>  3 1810_cPop  1810 Census Population Data [US, States & Counties]             301
#>  4 1820_cPop  1820 Census Population Data [US, States & Counties]             401
#>  5 1830_cPop  1830 Census Population Data [US, States & Counties]             501
#>  6 1840_cAg   1840 Census Agriculture Data [US, States & Counties]            601
#>  7 1840_cMfg  1840 Census Manufacturing Data [US, States & Counties]          602
#>  8 1840_cPopX 1840 Census Population & Other Data [US, States & Counti…      603
#>  9 1850_cAg   1850 Census Agriculture Data [US, States & Counties]            701
#> 10 1850_cPAX  1850 Census Population, Agriculture & Other Data [US, St…      702
#> # ℹ 251 more rows
```

Search metadata

- We know we're working with the 2017 ACS 5-year
- We can filter to identify datasets of interest

```
library(dplyr)

nhgis_ds |>
  filter(group == "2017 American Community Survey") |>
  select(name, description)
```

Search metadata

- We know we're working with the 2017 ACS 5-year
- We can filter to identify datasets of interest

```
library(dplyr)

nhgis_ds |>
  filter(group == "2017 American Community Survey") |>
  select(name, description)
#> # A tibble: 5 × 2
#>   name            description
#>   <chr>           <chr>
#> 1 2017_ACS1       1-Year Data
#> 2 2013_2017_ACS5a 5-Year Data [2013-2017, Block Groups & Larger Areas]
#> 3 2013_2017_ACS5b 5-Year Data [2013-2017, Tracts & Larger Areas]
#> 4 2013_2017_ACS5c 5-Year Data [2013-2017, Summary by Residence 1 Year Ago]
#> 5 2013_2017_ACS5d 5-Year Data [2013-2017, Summary by Place of Work]
```

Search metadata

- Each dataset is associated with multiple data tables
- We can view *detailed* metadata for an individual dataset, data table, or time series table
- Detailed metadata contains information about the available options for that data source

```
ds_meta <- get_metadata_nhgis(dataset = "2013_2017_ACS5a")
```

Search metadata

- Each dataset is associated with multiple data tables
- We can view *detailed* metadata for an individual dataset, data table, or time series table
- Detailed metadata contains information about the available options for that data source

```
ds_meta <- get_metadata_nhgis(dataset = "2013_2017_ACS5a")

str(ds_meta, 1)
#> List of 10
#>  $ name                   : chr "2013_2017_ACS5a"
#>  $ nhgis_id               : chr "ds233"
#>  $ group                  : chr "2017 American Community Survey"
#>  $ description            : chr "5-Year Data [2013-2017, Block Groups & Larger Areas]"
#>  $ sequence               : int 5502
#>  $ has_multiple_data_types: logi TRUE
#>  $ data_tables            : tibble [347 × 4] (S3: tbl_df/tbl/data.frame)
#>  $ geog_levels            : tibble [87 × 4] (S3: tbl_df/tbl/data.frame)
#>  $ geographic_instances   : tibble [52 × 2] (S3: tbl_df/tbl/data.frame)
#>  $ breakdowns             : tibble [1 × 4] (S3: tbl_df/tbl/data.frame)
```

Search metadata

- Each dataset is associated with multiple data tables
- We can view *detailed* metadata for an individual dataset, data table, or time series table
- Detailed metadata contains information about the available options for that data source

```
ds_meta$data_tables
#> # A tibble: 347 × 4
#>    name    nhgis_code description                                    sequence
#>    <chr>   <chr>      <chr>                                             <int>
#>  1 B00001  AHYO       Unweighted Sample Count of the Population             1
#>  2 B00002  AHYP       Unweighted Sample Housing Units                       2
#>  3 B01001  AHYQ       Sex by Age                                            3
#>  4 B01002  AHYR       Median Age by Sex                                     4
#>  5 B01002A AHYS       Median Age by Sex (White Alone)                       5
#>  6 B01002B AHYT       Median Age by Sex (Black or African American Alo…     6
#>  7 B01002C AHYU       Median Age by Sex (American Indian and Alaska Na…     7
#>  8 B01002D AHYV       Median Age by Sex (Asian Alone)                       8
#>  9 B01002E AHYW       Median Age by Sex (Native Hawaiian and Other Pac…     9
#> 10 B01002F AHYX       Median Age by Sex (Some Other Race Alone)             10
#> # ℹ 337 more rows
```

Search metadata

- Each dataset is associated with multiple data tables
- We can view *detailed* metadata for an individual dataset, data table, or time series table
- Detailed metadata contains information about the available options for that data source

```
library(stringr)

ds_meta$data_tables |>
  filter(str_detect(description, "Tenure"))
```

Search metadata

- Each dataset is associated with multiple data tables
- We can view *detailed* metadata for an individual dataset, data table, or time series table
- Detailed metadata contains information about the available options for that data source

```
library(stringr)

ds_meta$data_tables |>
  filter(str_detect(description, "Tenure"))
#> # A tibble: 34 × 4
#>    name    nhgis_code description                                    sequence
#>    <chr>   <chr>      <chr>                                            <int>
#>  1 B25003  AH37       Tenure                                             200
#>  2 B25003A AH38       Tenure (White Alone Householder)                   201
#>  3 B25003B AH39       Tenure (Black or African American Alone Househol…  202
#>  4 B25003C AH4A       Tenure (American Indian and Alaska Native Alone …  203
#>  5 B25003D AH4B       Tenure (Asian Alone Householder)                   204
#>  6 B25003E AH4C       Tenure (Native Hawaiian and Other Pacific Island…  205
#>  7 B25003F AH4D       Tenure (Some Other Race Alone Householder)         206
#>  8 B25003G AH4E       Tenure (Two or More Races Householder)             207
#>  9 B25003H AH4F       Tenure (White Alone, Not Hispanic or Latino Hous…  208
#> 10 B25003I AH4G       Tenure (Hispanic or Latino Householder)            209
#> # ℹ 24 more rows
```

Define an extract request

```
nhgis_ext <- define_extract_nhgis()
```

IPUMS NHGIS

Define an extract request

```
nhgis_ext <- define_extract_nhgis(
  description = "2017 ACS Tenure by race and ethnicity",
  datasets = ds_spec(
    "2013_2017_ACS5a",
    data_tables = c("B25003B", "B25003D", "B25003H", "B25003I"),
    geog_levels = "county"
  )
)
```

Define an extract request

```
nhgis_ext <- define_extract_nhgis(
  description = "2017 ACS Tenure by race and ethnicity",
  datasets = ds_spec(
    "2013_2017_ACS5a",
    data_tables = c("B25003B", "B25003D", "B25003H", "B25003I"),
    geog_levels = "county"
  )
)

nhgis_ext
#> Unsubmitted IPUMS NHGIS extract
#> Description: 2017 ACS Tenure by race and ethnicity
#>
#> Dataset: 2013_2017_ACS5a
#>   Tables: B25003B, B25003D, B25003H, B25003I
#>   Geog Levels: county
```

IPUMS NHGIS

Submit an extract request

```
nhgis_ext <- submit_extract(nhgis_ext)
```

IPUMS NHGIS

Submit an extract request

```
nhgis_ext <- submit_extract(nhgis_ext)

nhgis_ext
#> Submitted IPUMS NHGIS extract number 1327
#> Description: 2017 ACS Tenure by race and ethnicity
#>
#> Dataset: 2013_2017_ACS5a
#>   Tables: B25003B, B25003D, B25003H, B25003I
#>   Geog Levels: county
```

IPUMS NHGIS

Download your extract

- It may take time for the servers to process your request
- Use `wait_for_extract()` to ensure request has completed:

```
nhgis_ext <- wait_for_extract(nhgis_ext)
```

- Once it's complete, you can download the extract file(s):

```
nhgis_files <- download_extract(nhgis_ext)
```

Load an NHGIS extract

- NHGIS distributes files in zip archives:

```
basename(nhgis_files)
#> [1] "nhgis1327_csv.zip"
```

Load an NHGIS extract

- Read NHGIS zip archives directly with `read_nhgis()`

```
nhgis_data <- read_nhgis(nhgis_files)
```

IPUMS NHGIS

Load an NHGIS extract

- Read NHGIS zip archives directly with `read_nhgis()`

```
nhgis_data <- read_nhgis(nhgis_files)

nhgis_data
#> # A tibble: 3,220 × 66
#>    GISJOIN  YEAR    STUSAB REGIONA DIVISIONA STATE STATEA COUNTY COUNTYA COUSUBA
#>    <chr>    <chr>   <chr>  <lgl>   <lgl>     <chr> <chr>  <chr>  <chr>   <lgl>
#>  1 G0100010 2013-2… AL     NA      NA        Alab… 01     Autau… 001     NA
#>  2 G0100030 2013-2… AL     NA      NA        Alab… 01     Baldw… 003     NA
#>  3 G0100050 2013-2… AL     NA      NA        Alab… 01     Barbo… 005     NA
#>  4 G0100070 2013-2… AL     NA      NA        Alab… 01     Bibb … 007     NA
#>  5 G0100090 2013-2… AL     NA      NA        Alab… 01     Bloun… 009     NA
#>  6 G0100110 2013-2… AL     NA      NA        Alab… 01     Bullo… 011     NA
#>  7 G0100130 2013-2… AL     NA      NA        Alab… 01     Butle… 013     NA
#>  8 G0100150 2013-2… AL     NA      NA        Alab… 01     Calho… 015     NA
#>  9 G0100170 2013-2… AL     NA      NA        Alab… 01     Chamb… 017     NA
#> 10 G0100190 2013-2… AL     NA      NA        Alab… 01     Chero… 019     NA
#> # ℹ 3,210 more rows
#> # ℹ 56 more variables: PLACEA <lgl>, TRACTA <lgl>, BLKGRPA <lgl>,
#> #   CONCITA <lgl>, AIANHHA <lgl>, RES_ONLYA <lgl>, TRUSTA <lgl>, AIHHTLI <lgl>,
#> #   AITSCEA <lgl>, ANRCA <lgl>, CBSAA <lgl>, CSAA <lgl>, METDIVA <lgl>,
#> #   NECTAA <lgl>, CNECTAA <lgl>, NECTADIVA <lgl>, UAA <lgl>, CDCURRA <lgl>,
#> #   SLDUA <lgl>, SLDLA <lgl>, ZCTA5A <lgl>, SUBMCDA <lgl>, SDELMA <lgl>,
#> #   SDSECA <lgl>, SDUNIA <lgl>, PCI <lgl>, PUMAA <lgl>, GEOID <chr>, …
```

IPUMS NHGIS

Calculate homeownership disparities

How do we interpret variable codes?

```
colnames(nhgis_data)
#>  [1] "GISJOIN"   "YEAR"      "STUSAB"    "REGIONA"   "DIVISIONA" "STATE"
#>  [7] "STATEA"    "COUNTY"    "COUNTYA"   "COUSUBA"   "PLACEA"    "TRACTA"
#> [13] "BLKGRPA"   "CONCITA"   "AIANHHA"   "RES_ONLYA" "TRUSTA"    "AIHHTLI"
#> [19] "AITSCEA"   "ANRCA"     "CBSAA"     "CSAA"      "METDIVA"   "NECTAA"
#> [25] "CNECTAA"   "NECTADIVA" "UAA"       "CDCURRA"   "SLDUA"     "SLDLA"
#> [31] "ZCTA5A"    "SUBMCDA"   "SDELMA"    "SDSECA"    "SDUNIA"    "PCI"
#> [37] "PUMAA"     "GEOID"     "BTTRA"     "BTBGA"     "NAME_E"    "AH39E001"
#> [43] "AH39E002"  "AH39E003"  "AH4BE001"  "AH4BE002"  "AH4BE003"  "AH4FE001"
#> [49] "AH4FE002"  "AH4FE003"  "AH4GE001"  "AH4GE002"  "AH4GE003"  "NAME_M"
#> [55] "AH39M001"  "AH39M002"  "AH39M003"  "AH4BM001"  "AH4BM002"  "AH4BM003"
#> [61] "AH4FM001"  "AH4FM002"  "AH4FM003"  "AH4GM001"  "AH4GM002"  "AH4GM003"
```

Calculate homeownership disparities

Option 1: View metadata in loaded data

```
ipums_var_info(nhgis_data$AH39E001)
#> # A tibble: 1 × 3
#>   var_label        var_desc                                          val_labels
#>   <chr>            <chr>                                             <list>
#> 1 Estimates: Total Table AH39: Tenure (Black or African American Alo… <tibble>
```

Calculate homeownership disparities

Option 2: Check detailed metadata

```
get_metadata_nhgis(dataset = "2013_2017_ACS5a", data_table = "B25003H")
#> $name
#> [1] "B25003H"
#>
#> $description
#> [1] "Tenure (White Alone, Not Hispanic or Latino Householder)"
#>
#> $universe
#> [1] "Occupied housing units with a householder who is White alone, not Hispanic or Latino"
#>
#> $nhgis_code
#> [1] "AH4F"
#>
#> $sequence
#> [1] 208
#>
#> $dataset_name
#> [1] "2013_2017_ACS5a"
#>
#> $variables
#> # A tibble: 3 × 2
#>   description     nhgis_code
#>   <chr>           <chr>
#> 1 Total           AH4F001
#> 2 Owner occupied  AH4F002
#> 3 Renter occupied AH4F003
```

Calculate homeownership disparities

1. Calculate proportion of homeowners for each race category

```
nhgis_data <- nhgis_data |>
  mutate(
    ho_prop_wanh = AH4FE002 / AH4FE001,
    ho_prop_ba   = AH39E002 / AH39E001,
    ho_prop_aa   = AH4BE002 / AH4BE001,
    ho_prop_h    = AH4GE002 / AH4GE001
  )
```

Calculate homeownership disparities

2. Calculate ratio of homeowner proportions across race categories

```
nhgis_data <- nhgis_data |>
  mutate(
    ho_prop_wanh = AH4FE002 / AH4FE001,
    ho_prop_ba   = AH39E002 / AH39E001,
    ho_prop_aa   = AH4BE002 / AH4BE001,
    ho_prop_h    = AH4GE002 / AH4GE001
  ) |>
  mutate(
    ho_ratio_wanh_ba = ho_prop_wanh / ho_prop_ba,
    ho_ratio_wanh_h  = ho_prop_wanh / ho_prop_h,
    ho_ratio_wanh_aa = ho_prop_wanh / ho_prop_aa
  )
```

Calculate homeownership disparities

3. Select columns of interest

```
nhgis_data <- nhgis_data |>
  mutate(
    ho_prop_wanh = AH4FE002 / AH4FE001,
    ho_prop_ba   = AH39E002 / AH39E001,
    ho_prop_aa   = AH4BE002 / AH4BE001,
    ho_prop_h    = AH4GE002 / AH4GE001
  ) |>
  mutate(
    ho_ratio_wanh_ba = ho_prop_wanh / ho_prop_ba,
    ho_ratio_wanh_h  = ho_prop_wanh / ho_prop_h,
    ho_ratio_wanh_aa = ho_prop_wanh / ho_prop_aa
  ) |>
  select(YEAR, STATEA, STATE, COUNTYA, COUNTY, starts_with("ho_ratio"))

nhgis_data
#> # A tibble: 3,220 × 8
#>    YEAR      STATEA STATE   COUNTYA COUNTY      ho_ratio_wanh_ba ho_ratio_wanh_h
#>    <chr>     <chr>  <chr>   <chr>   <chr>                  <dbl>           <dbl>
#>  1 2013-2017 01     Alabama 001     Autauga Co…             1.43            1.00
#>  2 2013-2017 01     Alabama 003     Baldwin Co…             1.64            1.52
#>  3 2013-2017 01     Alabama 005     Barbour Co…             1.62            2.00
#>  4 2013-2017 01     Alabama 007     Bibb County             1.36            1.52
#>  5 2013-2017 01     Alabama 009     Blount Cou…             1.62            1.15
#>  6 2013-2017 01     Alabama 011     Bullock Co…             1.41          Inf
#>  7 2013-2017 01     Alabama 013     Butler Cou…             1.34            1.96
#>  8 2013-2017 01     Alabama 015     Calhoun Co…             1.61            1.47
```

Full workflow

```
nhgis_data <- define_extract_nhgis(
  description = "2017 ACS Tenure by race and ethnicity",
  datasets = ds_spec(
    "2013_2017_ACS5a",
    data_tables = c("B25003B", "B25003D", "B25003H", "B25003I"),
    geog_levels = "county"
  )
) |>
  submit_extract() |>
  wait_for_extract() |>
  download_extract() |>
  read_nhgis() |>
  mutate(
    ho_prop_wanh = AH4FE002 / AH4FE001,
    ho_prop_ba   = AH39E002 / AH39E001,
    ho_prop_aa   = AH4BE002 / AH4BE001,
    ho_prop_h    = AH4GE002 / AH4GE001
  ) |>
  mutate(
    ho_ratio_wanh_ba = ho_prop_wanh / ho_prop_ba,
    ho_ratio_wanh_h  = ho_prop_wanh / ho_prop_h,
    ho_ratio_wanh_aa = ho_prop_wanh / ho_prop_aa
  ) |>
  select(YEAR, STATEA, STATE, COUNTYA, COUNTY, starts_with("ho_ratio"))
```

Full workflow (more flexible)

ho_extract.R

```
define_extract_nhgis(
  description = "2017 ACS Tenure by race and ethnicity",
  datasets = ds_spec(
    "2013_2017_ACS5a",
    data_tables = c("B25003B", "B25003D", "B25003H", "B25003I"),
    geog_levels = "county"
  )
) |>
  submit_extract() |>
  wait_for_extract() |>
  download_extract("data/nhgis")
```

ho_ineq.R

```
file <- "data/nhgis/nhgis1307_csv.zip" # careful here!

read_nhgis(file) |>
  mutate(
    ho_prop_wanh = AH4FE002 / AH4FE001,
    ho_prop_ba   = AH39E002 / AH39E001,
    ho_prop_aa   = AH4BE002 / AH4BE001,
    ho_prop_h    = AH4GE002 / AH4GE001
  ) |>
  mutate(
    ho_ratio_wanh_ba = ho_prop_wanh / ho_prop_ba,
    ho_ratio_wanh_h  = ho_prop_wanh / ho_prop_h,
    ho_ratio_wanh_aa = ho_prop_wanh / ho_prop_aa
  ) |>
  select(YEAR, STATEA, STATE, COUNTYA, COUNTY, starts_with("ho_ratio"))
```

IPUMS NHGIS

Full workflow (more flexible)

ho_extract.R

```
define_extract_nhgis(
  description = "2017 ACS Tenure by race and ethnicity",
  datasets = ds_spec(
    "2013_2017_ACS5a",
    data_tables = c("B25003B", "B25003D", "B25003H", "B25003I"),
    geog_levels = "county"
  )
) |>
  submit_extract() |>
  wait_for_extract() |>
  download_extract("data/nhgis")
```

ho_ineq.R

```
file <- "data/nhgis/nhgis1307_csv.zip" # careful here!

read_nhgis(file) |>
  mutate(
    ho_prop_wanh = AH4FE002 / AH4FE001,
    ho_prop_ba   = AH39E002 / AH39E001,
    ho_prop_aa   = AH4BE002 / AH4BE001,
    ho_prop_h    = AH4GE002 / AH4GE001
  ) |>
  mutate(
    ho_ratio_wanh_ba = ho_prop_wanh / ho_prop_ba,
    ho_ratio_wanh_h  = ho_prop_wanh / ho_prop_h,
    ho_ratio_wanh_aa = ho_prop_wanh / ho_prop_aa
  ) |>
  select(YEAR, STATEA, STATE, COUNTYA, COUNTY, starts_with("ho_ratio"))
```

IPUMS NHGIS

Scaling up: An iterative workflow

- What if we want to make the same calculation for multiple years?
- Conveniently, ACS5a datasets contain *consistent data table codes* across years
- Can iteratively build dataset specifications for an extract definition

Scaling up: An iterative workflow

- Identify datasets of interest:

```
acs_ds <- get_metadata_nhgis("datasets") |>
  filter(str_detect(name, "ACS5a")) |>
  pull(name)

acs_ds
#>  [1] "2005_2009_ACS5a" "2006_2010_ACS5a" "2007_2011_ACS5a" "2008_2012_ACS5a"
#>  [5] "2009_2013_ACS5a" "2010_2014_ACS5a" "2011_2015_ACS5a" "2012_2016_ACS5a"
#>  [9] "2013_2017_ACS5a" "2014_2018_ACS5a" "2015_2019_ACS5a" "2016_2020_ACS5a"
#> [13] "2017_2021_ACS5a" "2018_2022_ACS5a"
```

Scaling up: An iterative workflow

- Each dataset needs its own specification in our extract (`ds_spec()`)
- We need to *iterate*

```
library(purrr)

# For each dataset in `acs_ds`, generate a `ds_spec` for the desired tables
ds <- map(
  acs_ds,
  function(x) {
    ds_spec(
      x,
      data_tables = c("B25003B", "B25003D", "B25003H", "B25003I"),
      geog_levels = "county"
    )
  }
)
```

Scaling up: An iterative workflow

- Next, just pass your list of dataset specifications to your extract definition

```
nhgis_ext <- define_extract_nhgis(
  description = "Homeownership data for 2017-2022 ACS",
  datasets = ds
)
```

Scaling up: An iterative workflow

- Next, just pass your list of dataset specifications to your extract definition

```
nhgis_ext <- define_extract_nhgis(
  description = "Homeownership data for 2017-2022 ACS",
  datasets = ds
)

nhgis_ext
#> Unsubmitted IPUMS NHGIS extract
#> Description: Homeownership data for 2017-2022 ACS
#>
#> Dataset: 2005_2009_ACS5a
#>   Tables: B25003B, B25003D, B25003H, B25003I
#>   Geog Levels: county
#>
#> Dataset: 2006_2010_ACS5a
#>   Tables: B25003B, B25003D, B25003H, B25003I
#>   Geog Levels: county
#>
#> Dataset: 2007_2011_ACS5a
#>   Tables: B25003B, B25003D, B25003H, B25003I
#>   Geog Levels: county
#>
#> Dataset: 2008_2012_ACS5a
#>   Tables: B25003B, B25003D, B25003H, B25003I
#>   Geog Levels: county
#>
#> Dataset: 2009_2013_ACS5a
```

Scaling up: An iterative workflow

- …and proceed with the extract submission process

```
nhgis_ext <- define_extract_nhgis(
  description = "Homeownership data for 2017-2022 ACS",
  datasets = ds
)

nhgis_files <- nhgis_ext |>
  submit_extract() |>
  wait_for_extract() |>
  download_extract()
```

Scaling up: An iterative workflow

- In this case, you'll have multiple files in your extract

```
ipums_list_files(nhgis_files)
#> # A tibble: 14 × 2
#>    type  file
#>    <chr> <chr>
#>  1 data  nhgis1328_csv/nhgis1328_ds176_20105_county.csv
#>  2 data  nhgis1328_csv/nhgis1328_ds184_20115_county.csv
#>  3 data  nhgis1328_csv/nhgis1328_ds191_20125_county.csv
#>  4 data  nhgis1328_csv/nhgis1328_ds195_20095_county.csv
#>  5 data  nhgis1328_csv/nhgis1328_ds201_20135_county.csv
#>  6 data  nhgis1328_csv/nhgis1328_ds206_20145_county.csv
#>  7 data  nhgis1328_csv/nhgis1328_ds215_20155_county.csv
#>  8 data  nhgis1328_csv/nhgis1328_ds225_20165_county.csv
#>  9 data  nhgis1328_csv/nhgis1328_ds233_20175_county.csv
#> 10 data  nhgis1328_csv/nhgis1328_ds239_20185_county.csv
#> 11 data  nhgis1328_csv/nhgis1328_ds244_20195_county.csv
#> 12 data  nhgis1328_csv/nhgis1328_ds249_20205_county.csv
#> 13 data  nhgis1328_csv/nhgis1328_ds254_20215_county.csv
#> 14 data  nhgis1328_csv/nhgis1328_ds262_20225_county.csv
```

Scaling up: An iterative workflow

- In this case, you'll have multiple files in your extract

```
# Match by keyword for 2018 data
nhgis_data <- read_nhgis(nhgis_files, file_select = contains("2018"))

# Match by index
nhgis_data <- read_nhgis(nhgis_files, file_select = 10)
```

Additional workflow ideas

- Automatically detect and add data for latest ACS upon release
- Recode variable names for extracts with multiple tables
- Build personal functions for common data prep processes
- Check out our previous webinar on reproducible workflows!
  - https://www.youtube.com/watch?v=xa9saWAga2M

Other NHGIS extract options

- Similar workflow for time series tables
  - `tst_spec()`
- Shapefiles available by name
  - `read_ipums_sf()`
- Other extract-wide parameters available
  - Geographic extents, file format, etc.

Final notes

- API + ipumsr support for several IPUMS microdata projects
- Save + share extract definitions in JSON format
- More examples on the ipumsr website:
  - https://tech.popdata.org/ipumsr
- Problems? Reach out on the IPUMS Forum or submit a GitHub issue

Useful links

- NHGIS: https://www.nhgis.org/
- ipumsr: https://tech.popdata.org/ipumsr
- NHGIS API requests in ipumsr: https://tech.popdata.org/ipumsr/articles/ipums-api-nhgis.html
- ipumsr GitHub issues: https://github.com/ipums/ipumsr/issues
- API Keys: https://account.ipums.org/api_keys
- IPUMS API: https://developer.ipums.org/docs/v2/apiprogram/
- IPUMS Forum: https://forum.ipums.org/
- Reproducible workflows webinar: https://www.youtube.com/watch?v=xa9saWAga2M
- ipumsr microdata webinar: https://www.youtube.com/watch?v=OT6upQ1dBgU

Thank you!

IPUMS NHGIS