IHIS Extraction and Analysis

Exercise 2

OBJECTIVE: Gain an understanding of how the IHIS dataset is structured and how it can be leveraged to explore your research interests. This exercise will use the IHIS dataset to explore associations among BMI, poverty, health status, sleep, and frequency of exercise.

11/13/2017

Research Questions

Is there a relationship between hours of sleep and health status? What is the relationship between health and poverty status, controlling for physiology?

Objectives

- Create and download an IHIS data extract
- Decompress data file and read data into R
- Analyze the data using sample code
- Validate data analysis work using answer key

IHIS Variables

- AGE: Age
- SEX: Sex
- POORYN: Above or Below Poverty threshold
- HEALTH: Health status
- BMI: Body Mass Index
- HRSLEEP: Usual hours of sleep per day
- VIG10FWK: Frequency of vigorous activity (10+min) per week

R Code to Review

This tutorial's sample code and answers use the so-called "tidyverse" style, but R has the blessing (and curse) that there are many different ways to do almost everything. If you prefer another programming style, please feel free to use it. But, for your reference, these are some quick explanations for commands that this tutorial will use:

- **%>%** - The pipe operator which helps make code with nested function calls easier to read. When reading code, it can be read as "and then". The pipe makes it so that code like `ingredients %>% stir() %>% cook()` is equivalent to `cook(stir(ingredients))` (read as "take *ingredients* and then *stir* and then *cook*").
- **as_factor** - Converts the value labels provide for IPUMS data into a factor variable for R
- **summarize** - Summarize a datasets observations to one or more groups
- **group_by** - Set the groups for the summarize function to group by
- **filter** - Filter the dataset so that it only contains these values
- **mutate** - Add on a new variable to a dataset
- **ggplot** - Make graphs using ggplot2

- **weighted.mean** - Get the weighted mean of the a variable

Review Answer Key (At end of document)

Common Mistakes to Avoid

1) Not changing the working directory to the folder where your data is stored
2) Mixing up = and == ; To assign a value in generating a variable, use "<-" (or "="). Use "==" to test for equality.

Note: In this exercise, for simplicity we will use "weighted.mean". For analysis where variance estimates are needed, use the survey or srvyr package instead.

Registering with IHIS

- Go to http://www.ihis.us, click on User Registration and Login and Apply for access. Log in if you are a registered user. If you are a first time user, enter an email address and password, then submit your user information so you can create IHIS data extracts.

Step 1: Download the Data

- Return to the homepage and click on Browse and Select Data.
- Click the Select Samples box, and check the boxes for the 2004 through 2010 samples.
- Click the Submit sample selections box. Using the drop down menu or search feature, select the following variables and add them to your data cart using the plus symbols to the left of the variable:
    - AGE: Age
    - SEX: Sex
    - POORYN: Above or Below Poverty threshold
    - HEALTH: Health status
    - BMI: Body Mass Index
    - HRSLEEP: Usual hours of sleep per day
    - VIG10FWK: Frequency of vigorous activity (10+min) per week

Step 2: Request the Data

- Click the green VIEW CART button under your data cart
- Review variable selection. Note that additional variables are in your data cart. The data extract system automatically supplies variables that indicate the sample (YEAR), are needed for variance estimation (SERIAL, PERNUM), and are used for

weighting the variables and years selected. Click the green Create Data Extract button.

- Review the â€˜Extract Request Summaryâ€™ screen, describe your extract, and click Submit Extract.
- You will receive an email when the data is available to download.
- To access the page to download the data, follow the link in the email, or click on the Download or Revise Extracts link on the homepage.

Getting the data into your statistics software

The following instructions are for R If you would like to use a different stats package, see: https://www.ihis.us/ihis/extract_instructions.shtml

Step 1: Download the Data
- Go to http://www.ihis.us/ and click on Download or Revise Extracts
- Right-click on the data link next to extract you created
- Choose "Save Target As..." (or "Save Link As...")
- Save into "Documents" (that should pop up as the default location)
- Do the same thing for the DDI link next to the extract
- (Optional) Do the same thing for the R script
- You do not need to decompress the data to use it in R

Step 2: Install the ipumsr package
- Open R from the Start menu
- If you haven't already installed the ipumsr package, in the command prompt, type the following command:

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

Step 3: Read in the data
- Set your working directory to where you saved the data above by adapting the following command (Rstudio users can also use the "Project" feature to set the working directory. In the menubar, select File -> New Project -> Existing Directory and then navigate to the folder):

```
setwd("~/") # "~/" goes to your Documents directory on most computers
```

- Run the following command from the console, adapting it so it refers to the extract you just created (note the number may not be the same depending on how many extracts you've already made):

```
library(ipumsr)
ddi <- read_ipums_ddi("ihis_00001.xml")
data <- read_ipums_micro(ddi)
```

```
# Or, if you downloaded the R script, the following is equivalent:
#    source("ihis_00001.R")
```

- This tutorial will also rely on the dplyr package, so if you want to run the same code, run the following command (but if you know other ways better, feel free to use them):

```
library(dplyr)
library(ggplot2)
```

- To stay consistent with the exercises for other statistical packages, this exercise does not spend much time on the helpers to allow for translation of the way IPUMS uses labelled values to the way base R does. You can learn more about these in the value-labes vignette in the R package. From R run command: `vignette("value-labels", package = "ipumsr")`

Analyze the Sample – Part I Group Means

Section 1: Analyze the Data

A) On the website, find the codes page for the HRSLEEP and HEALTH variables. What code values for HRSLEEP should be excluded to avoid skewing the average number of hours slept? How would you restrict the code values for HEALTH to eliminate unknown responses?

_____

B) Suppose you wanted to study the relationship between hours of sleep and health status. Determine the average reported hours of sleep per night by health status. On average, how many hours does an individual with excellent health in this sample sleep per night?

_____

```
data %>%
  filter(HEALTH < 6 & HRSLEEP > 0 & HRSLEEP < 25) %>%
  group_by(HEALTH = as_factor(HEALTH)) %>%
  summarize(HRSLEEP = mean(HRSLEEP))
```

C) Is there a noticeable trend between health status and hours of sleep using this sample? _____

D) Does the trend change for people under 60 in this sample?

_____

```
data %>%
  filter(HEALTH < 6 & HRSLEEP > 0 & HRSLEEP < 25 & AGE < 60) %>%
  group_by(HEALTH = as_factor(HEALTH)) %>%
  summarize(HRSLEEP = mean(HRSLEEP))
```

Analyze the Sample â€" Part II Weighting the Data

Section 1: Weight the Data

To get a more accurate estimation of demographic patterns from the sample, you will have to use the person weight.

A) Without weights, what proportion of people in this sample was below the poverty threshold in 2010? _____

```
data %>%
  group_by(YEAR, POORYN = as_factor(POORYN)) %>%
  summarize(n = n()) %>%
  mutate(pct = n / sum(n)) %>%
  filter(YEAR == 2010)
```

B) Using weights, what proportion of the population was below the poverty threshold in 2010? _____

```
data %>%
  group_by(YEAR, POORYN = as_factor(POORYN)) %>%
  summarize(n = sum(PERWEIGHT)) %>%
  mutate(pct = n / sum(n)) %>%
  filter(YEAR == 2010)
```

C) Using the household weight (and you must exclude all but one individual from a household), what proportion of households was below the poverty threshold in 2010? _____

```
data %>%
  filter(PERNUM == 1) %>%
  group_by(YEAR, POORYN = as_factor(POORYN)) %>%
  summarize(n = sum(HHWEIGHT)) %>%
  mutate(pct = n / sum(n)) %>%
  filter(YEAR == 2010)
```

Analyze the Sample â€" Part III Generating Variables

Section 1: Analyze the Data

Generate a variable that is 0 when an individual exercises less than 3 times a week, and 1 when an individual exercises 3 or more times a week.

A) Use `ipums_val_labels` to find the codes for VIG10FWK. Which code means "Never"? _____
Note: Youâ€™ll have to exclude codes above 28 when defining when exer3 is greater than 3 times a week.

B) What is the average difference in BMI for an individual in this sample who exercises at least 3 times a week compared to someone who exercises fewer than 3 times per week? _____

Remember to restrict the codes for BMI so unknown and missing codes are excluded.

```r
data <- data %>%
  mutate(EXER3 = VIG10FWK >= 3 & VIG10FWK <= 28)

data %>%
  filter(BMI > 0 & BMI < 99) %>%
  group_by(EXER3) %>%
  summarize(BMI = mean(BMI))
```

C) What percent of more frequent exercisers report excellent health? _____ Less frequent exercisers? _____

```r
data %>%
  filter(HEALTH < 6) %>%
  group_by(EXER3, HEALTH = as_factor(HEALTH)) %>%
  summarize(n = n()) %>%
  mutate(pct = n / sum(n))
```

Analyze the Sample – Part IV Graphing

Section 1: Analyze the Data

Create a graph to show the mean BMI over age for males and females.

A) How does the universe for BMI appear on this graph?

_____

B) Approximately at what age does BMI peak for women? For men?

_____

```r
data_summary <- data %>%
  filter(BMI > 0 & BMI < 99) %>%
  group_by(SEX = as_factor(SEX), AGE) %>%
  summarize(BMI = mean(BMI))

ggplot(data_summary, aes(x = AGE, y = BMI, color = SEX)) +
  geom_line() +
  scale_color_manual(values = c(Male = "#7570b3", Female = "#e6ab02"))
```

Section 2: Include a New Variable

Introduce the Variable (POORYN)

C) Create a graph to show how an associated effect of poverty status on BMI differs with gender, controlling for frequent exercise. Comment on three apparent trends.

_____

ANSWERS Analyze the Sample â€" Part I Group Means

Section 1: Analyze the Data

A) On the website, find the codes page for the HRSLEEP and HEALTH variables. What code values for HRSLEEP should be excluded to avoid skewing the average number of hours slept? How would you restrict the code values for HEALTH to eliminate unknown responses?

HRSLEEP: 00 NIU; 25 Less than 1 hour; 97 Unknown-refused; 98 Unknown-not ascertained; 99 Unknown-don't know HEALTH: 7 Unknown-refused; 8 Unknown-not ascertained; 9 Unknown-don't know

B) Suppose you wanted to study the relationship between hours of sleep and health status. Determine the average reported hours of sleep per night by health status. On average, how many hours does an individual with excellent health in this sample sleep per night?

7.2 hours

```
data %>%
  filter(HEALTH < 6 & HRSLEEP > 0 & HRSLEEP < 25) %>%
  group_by(HEALTH = as_factor(HEALTH)) %>%
  summarize(HRSLEEP = mean(HRSLEEP))
#> # A tibble: 5 x 2
#>      HEALTH  HRSLEEP
#>      <fctr>    <dbl>
#> 1 Excellent 7.212197
#> 2 Very Good 7.138609
#> 3      Good 7.164440
#> 4      Fair 7.173898
#> 5      Poor 7.297630
```

C) Is there a noticeable trend between health status and hours of sleep using this sample?

There seems to be no trend at all, except perhaps Excellent and Poor health have slightly higher averages, which could indicate people in poor health sleep to improve and people with excellent health are associated with getting more sleep.

D) Does the trend change for people under 60 in this sample?

When excluding the older population (perhaps with a higher incidence of poor health), better health is associated with more hours of sleep, though the differences between averages is small.

```
data %>%
  filter(HEALTH < 6 & HRSLEEP > 0 & HRSLEEP < 25 & AGE < 60) %>%
  group_by(HEALTH = as_factor(HEALTH)) %>%
  summarize(HRSLEEP = mean(HRSLEEP))
#> # A tibble: 5 x 2
#>      HEALTH  HRSLEEP
#>      <fctr>    <dbl>
#> 1 Excellent 7.184819
#> 2 Very Good 7.067901
#> 3      Good 7.051870
#> 4      Fair 6.967255
#> 5      Poor 6.943560
```

ANSWERS Analyze the Sample – Part II Weighting the Data

Section 1: Weight the Data

To get a more accurate estimation of demographic patterns from the sample, you will have to use the person weight.

A) Without weights, what proportion of people in this sample was below the poverty threshold in 2010?

   16.48% of the sample

```
data %>%
  group_by(YEAR, POORYN = as_factor(POORYN)) %>%
  summarize(n = n()) %>%
  mutate(pct = n / sum(n)) %>%
  filter(YEAR == 2010)
#> # A tibble: 3 x 4
#> # Groups:   YEAR [1]
#>    YEAR                          POORYN     n       pct
#>   <int>                          <fctr> <int>     <dbl>
#> 1  2010 At or above poverty threshold 64190 0.7134125
#> 2  2010      Below poverty threshold 14830 0.1648217
#> 3  2010  Unk (1997+: incl. Undefined) 10956 0.1217658
```

B) Using weights, what proportion of the population was below the poverty threshold in 2010?

   13.76% of the sample

```
data %>%
  group_by(YEAR, POORYN = as_factor(POORYN)) %>%
  summarize(n = sum(PERWEIGHT)) %>%
  mutate(pct = n / sum(n)) %>%
  filter(YEAR == 2010)
#> # A tibble: 3 x 4
#> # Groups:   YEAR [1]
#>    YEAR                          POORYN      n       pct
#>   <int>                          <fctr>  <dbl>     <dbl>
```

```
#> 1  2010 At or above poverty threshold 225482265 0.7414105
#> 2  2010      Below poverty threshold  41851235 0.1376115
#> 3  2010  Unk (1997+: incl. Undefined)  36792586 0.1209781
```

C) Using the household weight (and you must exclude all but one individual from a household), what proportion of households was below the poverty threshold in 2010?

12.91% of the sample

```
data %>%
  filter(PERNUM == 1) %>%
  group_by(YEAR, POORYN = as_factor(POORYN)) %>%
  summarize(n = sum(HHWEIGHT)) %>%
  mutate(pct = n / sum(n)) %>%
  filter(YEAR == 2010)
#> # A tibble: 3 x 4
#> # Groups:   YEAR [1]
#>    YEAR                          POORYN        n       pct
#>   <int>                          <fctr>    <dbl>     <dbl>
#> 1  2010 At or above poverty threshold 75120443 0.7383860
#> 2  2010      Below poverty threshold 13137035 0.1291287
#> 3  2010  Unk (1997+: incl. Undefined) 13478533 0.1324854
```

ANSWERS Analyze the Sample â€" Part III Generating Variables

Section 1: Analyze the Data

Generate a variable that is 0 when an individual exercises less than 3 times a week, and 1 when an individual exercises 3 or more times a week.

A) Use `ipums_val_labels` to find the codes for VIG10FWK. Which code means â€œNeverâ€•?

95 Never

Note: Youâ€™ll have to exclude codes above 28 when defining when exer3 is greater than 3 times a week.

```
ipums_val_labels(data$VIG10FWK)
#> # A tibble: 7 x 2
#>     val                               lbl
#>   <dbl>                             <chr>
#> # 1     0                    Not in Universe
#> # 2    94             Less than once per week
#> # 3    95                             Never
#> # 4    96 Unable to do moderate/vigorous activity
#> # 5    97                   Unknown-refused
#> # 6    98             Unknown-not ascertained
#> # 7    99                Unknown-don't know
```

B) What is the average difference in BMI for an individual in this sample who exercises at least 3 times a week compared to someone who exercises fewer than 3 times per week?

1.2 BMI (27.7-26.5) Remember to restrict the codes for BMI so unknown and missing codes are excluded.

```
data <- data %>%
  mutate(EXER3 = VIG10FWK >= 3 & VIG10FWK <= 28)

data %>%
  filter(BMI > 0 & BMI < 99) %>%
  group_by(EXER3) %>%
  summarize(BMI = mean(BMI))
#> # A tibble: 2 x 2
#>   EXER3      BMI
#>   <lgl>    <dbl>
#> 1 FALSE 27.72161
#> 2  TRUE 26.51532
```

C) What percent of more frequent exercisers report excellent health?

Exercise 3+ per week: 41.37% excellent health

Less frequent exercisers?

Fewer than 3 per week: 34.19% excellent health

```
data %>%
  filter(HEALTH < 6) %>%
  group_by(EXER3, HEALTH = as_factor(HEALTH)) %>%
  summarize(n = n()) %>%
  mutate(pct = n / sum(n))
#> # A tibble: 10 x 4
#> # Groups:   EXER3 [2]
#>    EXER3   HEALTH        n        pct
#>    <lgl>    <fctr> <int>       <dbl>
#>  1 FALSE Excellent 188948 0.341937200
#>  2 FALSE Very Good 163784 0.296398175
#>  3 FALSE      Good 142071 0.257104388
#>  4 FALSE      Fair  43852 0.079358501
#>  5 FALSE      Poor  13926 0.025201735
#>  6  TRUE Excellent  17900 0.413681535
#>  7  TRUE Very Good  14955 0.345620522
#>  8  TRUE      Good   8354 0.193066790
#>  9  TRUE      Fair   1812 0.041876589
#> 10  TRUE      Poor    249 0.005754564
```

ANSWERS Analyze the Sample â€" Part IV Graphing

Section 1: Analyze the Data

Create a graph to show the mean BMI over age for males and females.

A) How does the universe for BMI appear on this graph?
   There appears to be no BMI for individuals below 18, because the universe for BMI is only for adults older than 18.

B) Approximately at what age does BMI peak for women? For men?
   Women: ~ 61 years old; Men: ~50 years old

```r
data_summary <- data %>%
  filter(BMI > 0 & BMI < 99) %>%
  group_by(SEX = as_factor(SEX), AGE) %>%
  summarize(BMI = mean(BMI))

ggplot(data_summary, aes(x = AGE, y = BMI, color = SEX)) +
  geom_line() +
  scale_color_manual(values = c(Male = "#7570b3", Female = "#e6ab02"))
```

Section 2: Include a New Variable

Introduce the Variable (POORYN)

C) Create a graph to show how an associated effect of poverty status on BMI differs with gender, controlling for frequent exercise. Comment on three apparent trends.
   Women under the poverty threshold are more likely to have a higher BMI on average whether or not they exercise. Frequent exercisers have lower BMI's on average in each category. Men under the poverty threshold seem to have a lower BMI on average controlling for exercise.

```r
data_summary <- data %>%
  filter(BMI > 0 & BMI < 99 & POORYN != 9) %>%
  group_by( EXER3, SEX = as_factor(SEX), POORYN = as_factor(POORYN)) %>%
  summarize(BMI = mean(BMI))

ggplot(data_summary, aes(x = interaction(EXER3, SEX), y = BMI, fill =
POORYN)) +
  geom_col(position = "dodge") +
  scale_fill_manual(values = c("#7570b3", "#e6ab02")) +
  theme(legend.position = "bottom")
```