Lecture 6 problem set

INSERT YOUR NAME HERE

November 9, 2018

Contents

Required reading and instructions

Required reading before next class

- Work through slides from lecture 6 that we don't get to in class
  - [REQUIRED] slides from section 5 "Missing data"
- [REQUIRED] R Pivot Blog
  - https://tidyr.tidyverse.org/dev/articles/pivot.html
- [OPTIONAL] GW chapter 12 (tidy data)
  - Lecture 8 covers this material pretty closely, so read chapter if you can, but I get it if you don't have time
- [OPTIONAL] Wickham, H. (2014). Tidy Data. *Journal of Statistical Software*, 59(10), 1-23. doi: 10.18637/jss.v059.i10
  - This is the journal article that introduced the data concepts covered in GW chapter 12 and created the packages related to tidying data

Mid-semester evaluation

- Please take 10 minutes to complete the anonymous mid-quarter evaluation Here

---

Overview

This problem set has three parts.

1. I'll ask you some definitional/conceptual questions about the concepts introduced in lecture
2. Tidying untidy data: reshaping from long to wide

    - e.g., dataset has one row for each combination of university ID and enrollment age group, but you want a dataset with one row per university ID and one enrollment variable for each age group
    - for these questions we'll use fall enrollment data from the Integrated Postsecondary Data System (IPEDS), specifically the fall enrollment sub-survey that focuses on enrollment by age group
3. Tidying untidy data: reshaping from wide to long
    - for these questions we'll use data from the NCES digest of education statistics that contains data about the total number of teachers in each state

Load library and data

In order to use the `pivot_wider` and `pivot_longer` functions, you need to install the developer version of tidyr

```
#install.packages("devtools") #uncomment if you have not installed these packages
#devtools::install_github("tidyverse/tidyr")
library(tidyverse)
#> -- Attaching packages ---------------------------------------------------------------------------
#> v ggplot2 3.2.1          v purrr   0.3.2
#> v tibble  2.1.3          v dplyr   0.8.3
#> v tidyr   1.0.0.9000     v stringr 1.4.0
#> v readr   1.3.1          v forcats 0.4.0
#> -- Conflicts ------------------------------------------------------------------------------------
#> x dplyr::filter() masks stats::filter()
#> x dplyr::lag()    masks stats::lag()
library(haven)
library(labelled)
```

Part I: Conceptual questions

- What is the difference between the terms "unit of analysis" [our term; not necessarily used outside this class] and "observational level" [A Wickham term]?
    - ANSWER:
- What are the three rules of tidy data?
    - ANSWER:

Part II: Questions about reshaping long to wide

Description of the data

For these questions, we'll be using data from the Fall Enrollment survey component of the Integrated Postsecondary Education Data System (IPEDS)

- Specifically, we'll be using data from the survey sub-component that focuses on enrollment by age-group.
- The dataset we'll be using data from Fall 2016 (i.e., Fall of the 2016-17 academic year)
- Here is a link to a data dictionary (an excel file) for the enrollment by age dataset: LINK
- In the dataset you load below:
    - I've dropped a few of the variables from the raw enrollment by age data
    - I've added a few variables from the "institutional characteristics" survey (e.g., institution name, state, sector) that should be pretty self explanatory if you examine the variable labels and/or value labels
- the variable `unitid` is the ID variable for each college/university
- the dataset has one observation for each combination of the variables unitid-efbage-lstudy

Overview of the reshaping long to wide tasks

- Load the data frame and assign it the name `age_f16_allvars_allobs`
- Create two different data frame objects based on the data frame `age_f16_allvars_allobs`
  - A dataframe `agegroup1_obs` that has fewer variables than `age_f16_allvars_allobs` and keeps observations where age-group equals `1` (1. All age categories total)
    - this data frame has the simplist structure; we'll reshape this one first
  - A dataframe `levstudy1_obs` that has fewer variables than `age_f16_allvars_allobs` and keeps observations where "level of study" equals `1` (1. All Students total)
    - we'll reshape this one second
- Questions related to reshaping `agegroup1_obs`
- Questions related to reshaping `levstudy1_obs`

Load data and create three new data frames

- Load IPEDS data that contains fall enrollment by age

NOTE: IN THIS QUESTION, WE GIVE YOU THE ANSWERS; ALL YOU HAVE TO DO IS RUN THE BELOW CODE CHUNK

```
rm(list = ls()) # remove all objects
#getwd()
#list.files("../../../documents/rclass/data/ipeds/ef/age") # list files in directory w/ NLS data

#Read Stata data into R using read_data() function from haven package
age_f16_allvars_allobs <- read_dta(file="https://github.com/ozanj/rclass/raw/master/data/ipeds/ef/age/ef

#rename a couple variables
age_f16_allvars_allobs <- age_f16_allvars_allobs %>% rename(agegroup=efbage, levstudy=lstudy)

#list variables and variable labels
names(age_f16_allvars_allobs)
#>  [1] "unitid"       "agegroup"     "levstudy"     "efage01"
#>  [5] "efage02"      "efage03"      "efage04"      "efage05"
#>  [9] "efage06"      "efage07"      "efage08"      "efage09"
#> [13] "fullname"     "stabbr"       "sector"       "iclevel"
#> [17] "control"      "hloffer"      "locale"       "merge_age_ic"
age_f16_allvars_allobs %>% var_label()
#> $unitid
#> [1] "Unique identification number of the institution"
#>
#> $agegroup
#> [1] "Age category"
#>
#> $levstudy
#> [1] "Level of student"
#>
#> $efage01
#> [1] "Full time men"
#>
#> $efage02
#> [1] "Full time women"
#>
#> $efage03
#> [1] "Part time men"
```

```
#>
#> $efage04
#> [1] "Part time women"
#>
#> $efage05
#> [1] "Full time total"
#>
#> $efage06
#> [1] "Part time total"
#>
#> $efage07
#> [1] "Total men"
#>
#> $efage08
#> [1] "Total women"
#>
#> $efage09
#> [1] "Grand total"
#>
#> $fullname
#> [1] "Institution (entity) name"
#>
#> $stabbr
#> [1] "State abbreviation"
#>
#> $sector
#> [1] "Sector of institution"
#>
#> $iclevel
#> [1] "Level of institution"
#>
#> $control
#> [1] "Control of institution"
#>
#> $hloffer
#> [1] "Highest level of offering"
#>
#> $locale
#> [1] "Degree of urbanization (Urban-centric locale)"
#>
#> $merge_age_ic
#> NULL
```

- Create two new data frames based on `age_f16_allvars_allobs`

NOTE: IN THIS QUESTION, WE GIVE YOU THE ANSWERS; ALL YOU HAVE TO DO IS RUN THE BELOW CODE CHUNK

```
#Create dataframe that keeps observations where age-group equals `1` (1. All age categories total)
agegroup1_obs <- age_f16_allvars_allobs %>%
  select(fullname,unitid,agegroup,levstudy,efage09,stabbr,locale) %>%
  filter(agegroup==1) %>%
  select(-agegroup)
```

```
glimpse(agegroup1_obs)
#> Observations: 7,019
#> Variables: 6
#> $ fullname <chr> "Amridge University", "Amridge University", "Amridge ...
#> $ unitid   <dbl> 100690, 100690, 100690, 100724, 100724, 100724, 10075...
#> $ levstudy <dbl+lbl> 1, 2, 5, 1, 2, 5, 1, 2, 5, 1, 2, 1, 2, 5, 1, 2, 5...
#> $ efage09  <dbl> 597, 294, 303, 5318, 4727, 591, 37663, 32563, 5100, 1...
#> $ stabbr   <chr> "AL", "AL", "AL", "AL", "AL", "AL", "AL", "AL", "AL",...
#> $ locale   <dbl+lbl> 12, 12, 12, 12, 12, 12, 13, 13, 13, 32, 32, 12, 1...

#Create dataframe keeps observations where "level of study" equals `1` (1. All Students total)
levstudy1_obs <- age_f16_allvars_allobs %>%
  select(fullname,unitid,agegroup,levstudy,efage09,stabbr,locale) %>%
  filter(levstudy==1) %>%
  select(-levstudy)

glimpse(levstudy1_obs)
#> Observations: 36,703
#> Variables: 6
#> $ fullname <chr> "Amridge University", "Amridge University", "Amridge ...
#> $ unitid   <dbl> 100690, 100690, 100690, 100690, 100690, 100690, 10069...
#> $ agegroup <dbl+lbl> 1, 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 1, 2, 3, ...
#> $ efage09  <dbl> 597, 57, 7, 16, 34, 540, 88, 97, 110, 158, 78, 9, 531...
#> $ stabbr   <chr> "AL", "AL", "AL", "AL", "AL", "AL", "AL", "AL", "AL",...
#> $ locale   <dbl+lbl> 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 12, 1...
```

Questions related to reshaping the dataset agegroup1_obs from long to wide

- Run whatever investigations seem helpful to you to get to know the data (e.g., list variable names, list variable variable labels, list variable values, tabulations). You may decide to comment out some of these investigations before you knit and submit the problem set so that your pdf doesn't get too long.

Sort and print a few obs

Run some frequencies

- Run the following code, which confirms that there is one row per each combination of unitid-levstudy

NOTE: IN THIS QUESTION, WE GIVE YOU THE ANSWERS; BUT TRY TO UNDERSTAND WHAT EACH PART OF THE CODE IS DOING

```
agegroup1_obs %>% group_by(unitid,levstudy) %>% # group by vars
  summarise(n_per_group=n()) %>% # create a measure of number of observations per group
  ungroup %>% # ungroup (otherwise frequency table [next step] created) separately for each group
  count(n_per_group) # frequency of number of observations per group
#> # A tibble: 1 x 2
#>   n_per_group     n
#>         <int> <int>
#> 1           1  7019
```

Using code from previous question as a guide, confirm that the object agegroup1_obs has more than one observation for each value of unitid

- Diagnose whether the data frame agegroup1_obs meets each of the three criteria for tidy data
  - YOUR ANSWERS HERE:
    - Each variable must have its own column:

    * Each observation must have its own row:

    * Each value must have its own cell:
- What changes need to be made to `age_all` to make it tidy?
  - YOUR ANSWER HERE:
- With respect to "reshaping long to wide" to tidy a dataset, define the "names_to" parameter.
  - YOUR ANSWER HERE:
- What should the "names_to" column be in the data frame `agegroup1_obs`?
  - YOUR ANSWER HERE:
- With respect to "reshaping long to wide" to tidy a dataset, define the "values_to" paramenter.
  - YOUR ANSWER HERE:
- What should the "value_to" column be in the data frame `agegroup1_obs`?
  - YOUR ANSWER HERE:

Tidy the data frame `agegroup1_obs` and create a new object `agegroup1_obs_tidy`, then print a few observations

Confirm that the new object `agegroup1_obs_tidy` contains one observation for each value of unitid

Create a new object `agegroup1_obs_tidy_v2` from the object `agegroup1_obs` by performing the following steps in one line of code with multiple pipes:

- Create a variable `level` that is a character version of the variable 'levstudy'
- Drop the original variable `levstudy`
- Tidy the dataset

Print a few observations of `agegroup1_obs_tidy_v2`; Why is this data frame preferable over `agegroup1_obs_tidy`?

`- YOUR ANSWER HERE:`

Questions related to reshaping the dataset levstudy1_obs from long to wide

- Run whatever investigations seem helpful to you to get to know the data frame `levstudy1_obs` (e.g., list variable names, list variable variable labels, list variable values, tabulations). You may decide to comment out some of these investigations before you knit and submit the problem set so that your pdf doesn't get too long.

Sort and print a few obs

Run some frequencies

- Confirm that there is one row per each combination of unitid-agegroup

Using code from previous question as a guide, confirm that the object `levstudy1_obs` has more than observation for each value of unitid

- Why is the data frame `levstudy1_obs` not tidy?
  - YOUR ANSWER HERE:
- What changes need to be made to `levstudy1_obs` to make it tidy?
  - YOUR ANSWER HERE:

Tidy the data frame `levstudy1_obs` and create a new object `levstudy1_obs_tidy` (it is up to you whether you want to create character version of the variable `agegroup` prior to tidying) then print a few observations

Confirm that the new object `levstudy1_obs_tidy` contains one observation for each value of unitid

Part III: Questions about reshaping wide to long

Here, we load a table from NCES digest of education statistics that contains data about the total number of teachers in each state for particular years.

```
load(url("https://github.com/ozanj/rclass/raw/master/data/nces_digest/nces_digest_table_208_30.RData"))

#covert character variables for teacher totals to integers
table208_30[2:6] <- data.frame(lapply(table208_30[2:6],as.integer))

table208_30
#> # A tibble: 51 x 6
#>    state tot_fall_2000 tot_fall_2005 tot_fall_2009 tot_fall_2010
#>    <chr>         <int>         <int>         <int>         <int>
#>  1 Alab~         48194         57757         47492         49363
#>  2 Alas~          7880          7912          8083          8170
#>  3 Ariz~         44438         51376         51947         50030
#>  4 Arka~         31947         32997         37240         34272
#>  5 Cali~        298021        309222        316298        260806
#>  6 Colo~         41983         45841         49060         48542
#>  7 Conn~         41044         39687         43592         42951
#>  8 Dela~          7469          7998          8639          8933
#>  9 Dist~          4949          5481          5854          5925
#> 10 Flor~        132030        158962        183827        175609
#> # ... with 41 more rows, and 1 more variable: tot_fall_2011 <int>
```

- Why is the data frame `table208_30` not tidy?
  - YOUR ANSWER HERE:
- What changes need to be made to `table208_30` to make it tidy?
  - YOUR ANSWER HERE:

Tidy the data frame `table208_30` and create a new object `table208_30_tidy`:

- hint: use the
- after you tidy the data, print a few observations

Once finished, knit to (pdf) and upload both .Rmd and pdf files to class website under the week 6 tab
Remeber to use this naming convention "lastname_firstname_ps6"