Tutorial 1: Answers

Leah Brooks

2/1/2021

Problem Set 1 Answers

You are welcome and encouraged to work with others on the homework. However, each of you must turn in your own homework, in your own words. All duplicate versions of a homework receive a grade of zero.

1. Why do we do `table(was.counties$statefips)` and `summary(was.counties$cv1)` and not vice-versa?

We use `table()` for categorical or integer variables, and `summary()` for continuous variables. As a matter of practice, you can take a mean of a numeric categorical variable – it just won't mean anything!

2. Why does the first summary in part G.3. yield 11 observations, but the second 44?

Load the data:

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------
```

```
## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.4
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
```

```
## -- Conflicts ------------------------------------------------------------------------------------ tidy
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
# load csv data
was.counties <- read.csv("h:/pppa_data_viz/2019/tutorial_data/was_msas_1910_2010_20190107.csv")
str(was.counties)
```

```
## 'data.frame':    246 obs. of  5 variables:
##  $ statefips : int  11 24 24 24 24 24 51 51 51 51 ...
##  $ countyfips: int  1 9 17 21 31 33 13 43 47 59 ...
##  $ cv1       : int  331069 10325 16386 52673 32089 36147 10231 7468 13472 20536 ...
##  $ year      : int  1910 1910 1910 1910 1910 1910 1910 1910 1910 1910 ...
##  $ cv28      : int  NA NA NA NA NA NA NA NA NA NA ...
```

Here is the first summary:

```
# summarize by year w/o missings
print("find average by year w/o missing values")
```

```
## [1] "find average by year w/o missing values"
```

```
was.counties.grp.yr <- group_by(.data = was.counties, year)
was.by.year <- summarize(.data = was.counties.grp.yr, cv1.yr=mean(cv1, na.rm = TRUE))
was.by.year
```

```
## # A tibble: 11 x 2
##     year  cv1.yr
##    <int>   <dbl>
##  1  1910  32892.
##  2  1920  39282.
##  3  1930  44222.
##  4  1940  59891.
##  5  1950  81966.
##  6  1960 110812.
##  7  1970 143834.
##  8  1980 142777
##  9  1990 173222.
## 10  2000 201560.
## 11  2010 234843
```

And here is the second:

```
# summarize by state and year
print("find info by state and year")
```

```
## [1] "find info by state and year"
```

```
was.counties.grp.st <- group_by(.data = was.counties,year,statefips)
was.by.state.yr <- summarize(.data = was.counties.grp.st,
                             cv.st.total = sum(cv1, na.rm = TRUE))
was.by.state.yr
```

```
## # A tibble: 44 x 3
## # Groups:   year [11]
##     year statefips cv.st.total
##    <int>     <int>       <int>
##  1  1910        11      331069
##  2  1910        24      147620
##  3  1910        51      163267
##  4  1910        54       15889
##  5  1920        11      437571
##  6  1920        24      158258
##  7  1920        51      174085
##  8  1920        54       15729
##  9  1930        11      486869
## 10  1930        24      189435
## # ... with 34 more rows
```

The first reports one observation by year, and there are 11 years in the data (1910-2010). The second summarize reports data by year and state, so that there are 11 years x 4 states (DC, MD, VA, WV) observations, or 44.

3. Find and report the average population in DC for the entire period 1910-2010

```
# keep dc only
was.counties.dc <- was.counties[which(was.counties$statefips == 11),]
# mean population for all years
mean(was.counties.dc$cv1, na.rm = TRUE)
```

```
## [1] 605478.1
```

4. Find state-level (or the part of the state we observe) average population over the entire period. Put a table with this information in your final output. Describe the results in a sentence or two.

```r
# find state-level population by year
str(was.counties)
```

```
## 'data.frame':    246 obs. of  5 variables:
##  $ statefips : int  11 24 24 24 24 24 51 51 51 51 ...
##  $ countyfips: int  1 9 17 21 31 33 13 43 47 59 ...
##  $ cv1       : int  331069 10325 16386 52673 32089 36147 10231 7468 13472 20536 ...
##  $ year      : int  1910 1910 1910 1910 1910 1910 1910 1910 1910 1910 ...
##  $ cv28      : int  NA NA NA NA NA NA NA NA NA NA ...
```

```r
was.counties.st <- group_by(.data = was.counties, statefips, year)
# add up to state-year level
state.year <- summarize(.data = was.counties.st, state_pop = sum(cv1, na.rm = TRUE))
# now take a state-level average
state.year <- group_by(.data = state.year, statefips)
state.overall <- summarize(.data = state.year, state_pop_all = mean(state_pop, na.rm = TRUE))
state.overall
```

```
## # A tibble: 4 x 2
##   statefips state_pop_all
##       <int>         <dbl>
## 1        11       605478.
## 2        24       999115.
## 3        51       987486.
## 4        54        25746.
```

5. For each of the four states, are there more or fewer jurisdictions in this dataset now than in 1910? (Hint: `sum(!is.na(variable.name))` tells you the total number of non-missing observations.)

Here I count jurisdictions by year for all years

```r
# group at state-year level
was.counties.st <- group_by(.data = was.counties, statefips, year)
# count jurisdictions by state-year
state.year <- summarize(.data = was.counties.st, no_jurisdictions = sum(!is.na(cv1), na.rm = TRUE))
# just print 1910 and 2010
state.year[which(state.year$year %in% c(1910,2010)),]
```

```
## # A tibble: 8 x 3
## # Groups:   statefips [4]
##   statefips  year no_jurisdictions
##       <int> <int>            <int>
## 1        11  1910                1
## 2        11  2010                1
## 3        24  1910                5
## 4        24  2010                5
## 5        51  1910               13
## 6        51  2010               17
## 7        54  1910                1
## 8        54  2010                1
```

6. What is the most populous jurisdiction in the DC area in 2010?

```r
# just limit to 2010 counties
was.counties.2010 <- was.counties[which(was.counties$year == 2010),]
# print the maximum population
max.pop <- max(was.counties.2010$cv1, na.rm = TRUE)
```

```r
# list all counties
was.counties.2010[,c("statefips","countyfips","cv1")]
```

```
##     statefips countyfips     cv1
## 223        11          1  601723
## 224        24          9   88737
## 225        24         17  146551
## 226        24         21  233385
## 227        24         31  971777
## 228        24         33  863420
## 229        51         13  207627
## 230        51         43   14034
## 231        51         47   46689
## 232        51         59 1081726
## 233        51         61   65203
## 234        51        107  312311
## 235        51        153  402002
## 236        51        157    7373
## 237        51        177  122397
## 238        51        179  128961
## 239        51        187   37575
## 240        51        510  139966
## 241        51        600   22565
## 242        51        610   12332
## 243        51        630   24286
## 244        51        683   37821
## 245        51        685   14273
## 246        54         37   53498
```

The maximum matches to Fairfax County, VA: state 51, county 59.