The basics: 04 grouped analysis

Ari Anisfeld

9/8/2020

Questions

`group_by` and `summarize`

1. `midwest` is a data set that comes bundled with `tidyverse`. In an earlier lab you calculated the population of Ohio in the following way.

```
midwest %>%
  filter(state == "OH")
  summarize(total_population = sum(poptotal))
```

   With `group_by` you can calculate the total population of all the states at once!

```
midwest %>%
  group_by(...) %>%
  summarize(total_population = sum(poptotal))
```

2. For each state in the `midwest` data, calculate total `area`.
3. For each state in the `midwest` data, calculate the proportion of counties that are in a metro area (`inmetro`).[1]
4. For each state, calculate the proportion of people with a college degree and also with high school degrees.
   - First, use `mutate` to calculate the number of people with the degree type.
   - Then, use `group_by` and `summarize` to calculate the proportions.

`group_by` and `mutate`

1. Add a column to midwest called `pop_state` that equals the state population. Compare your result to what you calculated early.

```
# fill in the ... with approriate code
midwest %>%
  group_by( ... ) %>%
  mutate(pop_state = ... )
```

2. Building off the previous question, create a column that shows the number of people living below the poverty line (`percbelowpoverty`) in each county. Look at your results to make sure they make sense.

[1] Recall that the `mean()` of a column of `0` and `1`s tell you the proportion of `1`s.

`count`

1. Reproduce this table using `count()`.

```
## # A tibble: 2 x 2
##   inmetro     n
##     <int> <int>
## 1       0   287
## 2       1   150
```

2. Reproduce this table using `add_count()`.

```
## # A tibble: 6 x 3
## # Groups:   inmetro [2]
##   state inmetro     n
##   <chr>   <int> <int>
## 1 IL          0   287
## 2 IL          0   287
## 3 IL          0   287
## 4 IL          1   150
## 5 IL          0   287
## 6 IL          0   287
```

```
# fill in the ... with the appropriate code.
 midwest %>%
    select(state, inmetro) %>%
    ... %>%
    head()
```

1. Reproduce the following table

```
## # A tibble: 10 x 3
##    state inmetro     n
##    <chr>   <int> <int>
##  1 IL          0    74
##  2 IL          1    28
##  3 IN          0    55
##  4 IN          1    37
##  5 MI          0    58
##  6 MI          1    25
##  7 OH          0    48
##  8 OH          1    40
##  9 WI          0    52
## 10 WI          1    20
```

Solutions

1.
```
midwest %>%
  group_by(state) %>%
  summarize(total_population = sum(poptotal))
```

2.
```
midwest %>%
  group_by(state) %>%
  summarize(total_area = sum(area))
```

3.
```
midwest %>%
  group_by(state) %>%
  summarize(prop_in_metro = mean(inmetro))
```

4.
```
 midwest %>%
  mutate(pop_with_hs = perchsd * poptotal,
         pop_with_college = percollege * poptotal) %>%
  group_by(state) %>%
  summarize(total_population = sum(poptotal),
            perc_with_hs = sum(pop_with_hs)/total_population,
            perc_with_college = sum(pop_with_college)/total_population,)
```

You might have been tempted to do it in the following way, but this underestimates the statewide ra

```
  midwest %>%
    group_by(state) %>%
    summarise(perc_with_hs = mean(perchsd))
```

group_by and mutate

1.
```
midwest %>%
  group_by(state) %>%
  mutate(pop_state = sum(poptotal))
```

2. A careful analyst would say this is wrong, because we do not know the poverty status of each and every person in the counties (see `percpovertyknown`). A challenge problem is to find the lower and upper bound on the number of people with poverty per county.

```
midwest %>%
  group_by(state) %>%
  mutate(pop_state = sum(poptotal),
         pop_below_poverty = pop_state * percbelowpoverty/100)
```

count

1.
```
midwest %>%
  count(inmetro)
```

```
## # A tibble: 2 x 2
##   inmetro     n
##     <int> <int>
## 1       0   287
## 2       1   150
```

2.
```
# fill in the ... with the appropriate code.
 midwest %>%
    select(state, inmetro) %>%
```

```
    add_count(inmetro) %>%
    head()
```

3.
```
## # A tibble: 10 x 3
##    state inmetro     n
##    <chr>   <int> <int>
##  1 IL          0    74
##  2 IL          1    28
##  3 IN          0    55
##  4 IN          1    37
##  5 MI          0    58
##  6 MI          1    25
##  7 OH          0    48
##  8 OH          1    40
##  9 WI          0    52
## 10 WI          1    20
```