

homework_4

Kayla Williams

October 24, 2018

Read in the data as an R object named homicides.

```
homicides <- read.csv("C:/Users/Kayla/Desktop/r_course_2018/Homework_4/data/homicide-data.csv")
```

```
head(homicides)
```

```
##      uid reported_date victim_last victim_first victim_race victim_age
## 1 Alb-000001    20100504    GARCIA      JUAN    Hispanic         78
## 2 Alb-000002    20100216    MONTOYA    CAMERON    Hispanic         17
## 3 Alb-000003    20100601 SATTERFIELD    VIVIANA     White          15
## 4 Alb-000004    20100101    MENDIOLA    CARLOS    Hispanic         32
## 5 Alb-000005    20100102      MULA      VIVIAN     White          72
## 6 Alb-000006    20100126     BOOK    GERALDINE     White          91
##  victim_sex      city state      lat      lon      disposition
## 1      Male Albuquerque    NM 35.09579 -106.5386 Closed without arrest
## 2      Male Albuquerque    NM 35.05681 -106.7153      Closed by arrest
## 3     Female Albuquerque    NM 35.08609 -106.6956 Closed without arrest
## 4      Male Albuquerque    NM 35.07849 -106.5561      Closed by arrest
## 5     Female Albuquerque    NM 35.13036 -106.5810 Closed without arrest
## 6     Female Albuquerque    NM 35.15111 -106.5378      Open/No arrest
```

Create a new column called city_name that combines the city and state like this “Baltimore, MD”.

```
homicides <- homicides %>%
  unite(city_name, city, state, sep = ", ", remove = FALSE)
```

```
head(homicides)
```

```
##      uid reported_date victim_last victim_first victim_race victim_age
## 1 Alb-000001    20100504    GARCIA      JUAN    Hispanic         78
## 2 Alb-000002    20100216    MONTOYA    CAMERON    Hispanic         17
## 3 Alb-000003    20100601 SATTERFIELD    VIVIANA     White          15
## 4 Alb-000004    20100101    MENDIOLA    CARLOS    Hispanic         32
## 5 Alb-000005    20100102      MULA      VIVIAN     White          72
## 6 Alb-000006    20100126     BOOK    GERALDINE     White          91
##  victim_sex      city_name      city state      lat      lon
## 1      Male Albuquerque, NM Albuquerque    NM 35.09579 -106.5386
## 2      Male Albuquerque, NM Albuquerque    NM 35.05681 -106.7153
## 3     Female Albuquerque, NM Albuquerque    NM 35.08609 -106.6956
## 4      Male Albuquerque, NM Albuquerque    NM 35.07849 -106.5561
## 5     Female Albuquerque, NM Albuquerque    NM 35.13036 -106.5810
## 6     Female Albuquerque, NM Albuquerque    NM 35.15111 -106.5378
##      disposition
## 1 Closed without arrest
## 2      Closed by arrest
## 3 Closed without arrest
## 4      Closed by arrest
## 5 Closed without arrest
## 6      Open/No arrest
```

Create a dataframe called `unsolved` with one row per city that gives the total number of homicides for the city and the number of unsolved homicides (those for which the disposition is “Closed without arrest” or “Open/No arrest”).

```
homicides <- homicides %>%
  select(city_name, disposition) %>%
  mutate(unsolved_homicides = str_detect(disposition,
                                          c("Closed without arrest|Open/No arrest"))) %>%
  rename(total_homicides = disposition)

unsolved <- homicides %>% group_by(city_name) %>%
  summarise(total_homicides = sum(!is.na(homicides$total_homicides)),
            unsolved_homicides = sum((homicides$unsolved_homicides == "TRUE")))
```

For the city of Baltimore, MD, use the `prop.test` function to estimate the proportion of homicides that are unsolved, as well as the 95% confidence interval for this proportion. Print the output of the `prop.test` directly in your RMarkdown, and then save the output of `prop.test` as an R object and apply the `tidy` function from the `broom` package to this object and pull the estimated proportion and confidence intervals from the resulting tidy dataframe.

```
homicide_prop <- unsolved %>%
  filter(city_name == "Baltimore, MD")

baltimore_homicides <- prop.test(x = homicide_prop$unsolved_homicides,
                                n = homicide_prop$total_homicides)

tidy(baltimore_homicides)
```

```
## # A tibble: 1 x 8
##   estimate statistic p.value parameter conf.low conf.high method
##   <dbl>      <dbl>   <dbl>      <int>    <dbl>    <dbl> <chr>
## 1    0.508      13.2 2.80e-4         1    0.504    0.512 1-sam~
## # ... with 1 more variable: alternative <chr>
```

Now use what you learned from running `prop.test` for one city to run `prop.test` for all the cities. Your goal is to create the figure shown in homework directions, where the points show the estimated proportions of unsolved homicides in each city and the horizontal lines show the estimated 95% confidence intervals. Do this all within a “tidy” pipeline, starting from the `unsolved` dataframe that you created for step 3. Use `map2` from `purrr` to apply `prop.test` within each city and then `map` from `purrr` to apply `tidy` to this output. Use the `unnest` function from the `tidyr` package on the resulting list-column (from mapping `tidy` to the `prop.test` output list-column), with the option `.drop = TRUE`, to get your estimates back into a regular tidy dataframe before plotting.

```
all_homicides <- map2(unsolved$unsolved_homicides, unsolved$total_homicides, .f = prop.test)

all_homicides2 <- map_df(all_homicides, tidy)

unnest(all_homicides2, .drop = TRUE)
```

```
## # A tibble: 51 x 8
##   estimate statistic p.value parameter conf.low conf.high method
##   <dbl>      <dbl>   <dbl>      <int>    <dbl>    <dbl> <chr>
## 1    0.508      13.2 2.80e-4         1    0.504    0.512 1-sam~
## 2    0.508      13.2 2.80e-4         1    0.504    0.512 1-sam~
## 3    0.508      13.2 2.80e-4         1    0.504    0.512 1-sam~
## 4    0.508      13.2 2.80e-4         1    0.504    0.512 1-sam~
## 5    0.508      13.2 2.80e-4         1    0.504    0.512 1-sam~
```

```
## 6      0.508      13.2 2.80e-4      1      0.504      0.512 1-sam~
## 7      0.508      13.2 2.80e-4      1      0.504      0.512 1-sam~
## 8      0.508      13.2 2.80e-4      1      0.504      0.512 1-sam~
## 9      0.508      13.2 2.80e-4      1      0.504      0.512 1-sam~
## 10     0.508      13.2 2.80e-4      1      0.504      0.512 1-sam~
## # ... with 41 more rows, and 1 more variable: alternative <chr>
```

Create the plot shown below. Hint: Check out the `geom_errorbarh` geom with the `height = 0` option to get the horizontal lines for the confidence intervals. All of the code for this should be in an RMarkdown document. Render this to a pdf and then push to your GitHub repository. Go on GitHub and make sure that everything made it online.

```
prop_df %>%
  mutate(city_name = fct_reorder(city_name, estimate)) %>%
  filter(city_name != "Tulsa, AL") %>%
  ggplot(aes(x = city_name, y = estimate)) +
  geom_point(color = "white") +
  geom_errorbar(aes(ymin = conf.low,
                    ymax = conf.high), width = 0,
                color = "white", alpha = 0.5) +
  coord_flip() +
  ggtitle("Unsolved homicides by city", "Bars show 95% confidence interval") +
  labs(x = "", y = "Percents of homicides that are unsolved") +
  scale_y_continuous(labels = percent) +
  theme_dark()
```

Unsolved homicides by city

Bars show 95% confidence interval

