

# Homework\_Four\_Final

Anastasia Ratcliff

11/7/2018

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##   filter, lag
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages ----- tidyverse 1.2.1 --
## v tibble  1.4.2      v purrr   0.2.5
## v tidyr   0.8.1      v stringr 1.3.1
## v readr   1.1.1      v forcats 0.3.0
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
## The following object is masked from 'package:base':
##
##   date
```

```
library(readr)
library(stringr)
library(purrr)
library(broom)
library(tibble)
```

```
# Read in the data as an R Object
```

```
homicides <- read_csv("/Users/anastasiaratchiff/Desktop/R- Programing /Homework_Four/Data/homicide-data
```

```
## Parsed with column specification:
## cols(
##   uid = col_character(),
##   reported_date = col_integer(),
##   victim_last = col_character(),
##   victim_first = col_character(),
##   victim_race = col_character(),
##   victim_age = col_character(),
```

```

## victim_sex = col_character(),
## city = col_character(),
## state = col_character(),
## lat = col_double(),
## lon = col_double(),
## disposition = col_character()
## )

#create new column called city_name
homicides <- homicides %>%
  unite(city_name, city, state, sep = ",", remove =FALSE )

#new dataframe
#one row per city that gives total homicides

#one row number of unsolved homicides
unsolved <- homicides %>%
  select(city_name, disposition)%>%
  mutate(disposition = factor(disposition))%>%
  mutate(disposition = fct_recode(disposition, '1' = 'Closed without arrest',
                                '1' = 'Open/No arrest',
                                '0' = 'Closed by arrest '))%>%

  group_by(city_name)%>%
  mutate(total_homicides = n()) %>%
  mutate(tote_unsolved = disposition == '1') %>%
  mutate(tote_unsolved = sum(tote_unsolved))%>%
  group_by(city_name, tote_unsolved)%>%
  count()%>%
  ungroup () %>%
  rename(total_homicides = n)

#city of Baltimore use prop.test

baltimore <- unsolved %>%
  filter(city_name == 'Baltimore,MD')

#prop.test
prop.test(baltimore$tote_unsolved,
          baltimore$total_homicides)

##
## 1-sample proportions test with continuity correction
##
## data:  baltimore$tote_unsolved out of baltimore$total_homicides, null probability 0.5
## X-squared = 239.01, df = 1, p-value < 2.2e-16
## alternative hypothesis: true p is not equal to 0.5
## 95 percent confidence interval:
##  0.6275625 0.6631599
## sample estimates:
##           p
## 0.6455607

```

```

#object from prop test
baltimore_pt <- prop.test(baltimore$tote_unsolved,
                          baltimore$total_homicides)

#tidy function from broom
baltimore_pt %>%
  tidy() %>%
  select(estimate, conf.low, conf.high)

## # A tibble: 1 x 3
##   estimate conf.low conf.high
##   <dbl>     <dbl>     <dbl>
## 1    0.646    0.628    0.663

#running prop.test for all of them

test1 <- purrr::map2(unsolved$tote_unsolved,unsolved$total_homicides,
                     ~ prop.test(.x, n = .y))

unsolved <- unsolved %>%
  mutate( sts_col = purrr::map2(unsolved$tote_unsolved,unsolved$total_homicides,
                                ~ prop.test(.x, n = .y)%>%
                                  {tibble(estimate = .[["estimate"]],
                                           CI_lower = .[["conf.int"]][[1]],
                                           CI_upper = .[["conf.int"]][[2]]})}%>%
                                unnest())%>%
  mutate(city_name = factor (city_name, levels = city_name[order(estimate)]))

#plot data

unsolved_plot <- ggplot(unsolved)+
  geom_errorbarh(aes(xmin = CI_lower,
                    xmax = CI_upper,
                    y = reorder(city_name, estimate)),height = 0, color = "white")+
  geom_point(mapping = aes(x = estimate, y = reorder(city_name, estimate)),
            color = "white", size = 2) +
  ggtitle("Unsolved homicides by city",
          subtitle = "Bars show 95% confidence interval") +
  labs(x = "Percent of homicides that are unsolved",
       y = "") +
  scale_x_continuous(breaks = c(0.2, 0.3, 0.4, 0.5, 0.6, 0.7),
                    limits = c(.2,.8),
                    label = c("20.0%", "30.0%",
                              "40.0%", "50.0%",
                              "60.0%", "70.0%")) +
  theme_dark()

unsolved_plot

```

## Unsolved homicides by city

Bars show 95% confidence interval

