727_HW1_AngelinaLu

Angelina Lu

2024-09-17

1) Provide the link to the GitHub repo that you used to practice git from Week 1.

https://github.com/angelinalu61/test-727-0828

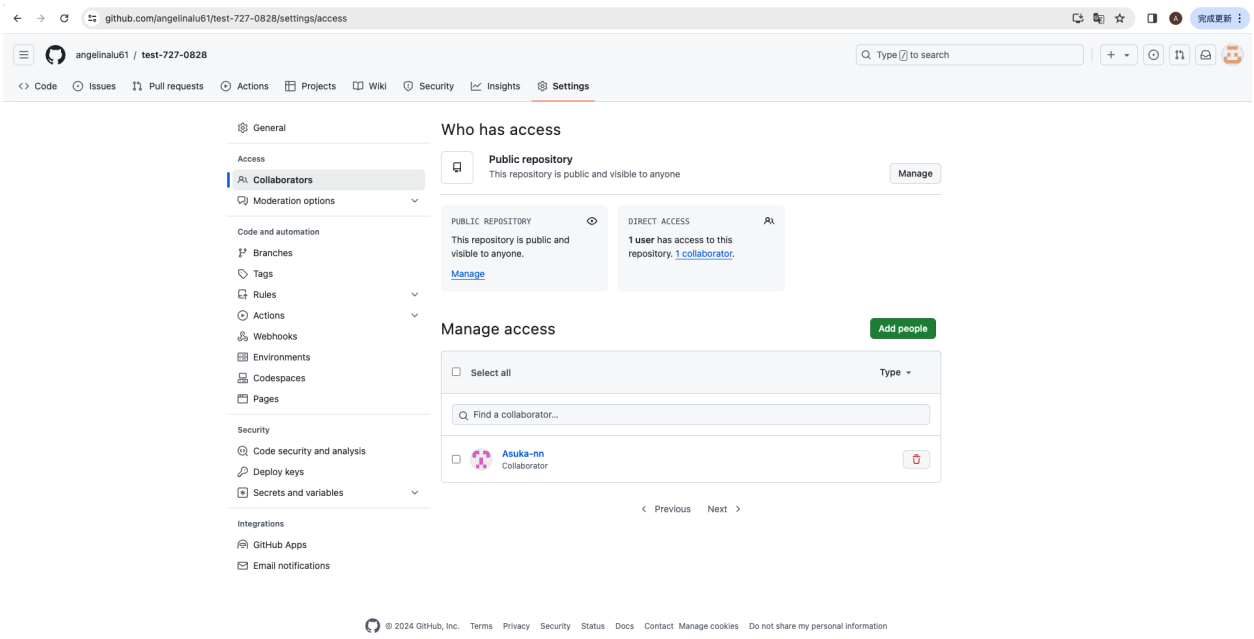


Figure 1: Collaborator

2) Read in the .dta version and store in an object called angell_stata.

3) Read in the .txt version and store it in an object called angell_txt.

```
options(repos = c(CRAN = "http://cran.r-project.org"))
install.packages("haven")
```

```
##
## The downloaded binary packages are in
##  /var/folders/8_/p0sgn35n70bcqqr1rc6jxwyr0000gn/T//RtmpFPQ3S0/downloaded_packages
```

```
install.packages("readr")
```

```
##
## The downloaded binary packages are in
##  /var/folders/8_/p0sgn35n70bcqqr1rc6jxwyr0000gn/T//RtmpFPQ3S0/downloaded_packages
```

```
library(haven)
library(readr)
knitr::opts_chunk$set(echo = TRUE, cache = TRUE,
                      autodep = TRUE, cache.comments = FALSE,
                      message = FALSE, warning = FALSE,
                      fig.width = 4.5, fig.height = 3.5)
angell_stata <- read_dta("/Users/angelina/Desktop/UMich/727_Fundamentals of Computing and Data Display/
angell_txt <- read_table("/Users/angelina/Desktop/UMich/727_Fundamentals of Computing and Data Display/
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   city = col_character(),
##   morint = col_double(),
##   ethhet = col_double(),
##   geomob = col_double(),
##   region = col_character()
## )
```

```
## Warning: 29 parsing failures.
## row col  expected    actual
##   1  -- 5 columns 6 columns '/Users/angelina/Desktop/UMich/727_Fundamentals of Computing and Data Dis
##   2  -- 5 columns 6 columns '/Users/angelina/Desktop/UMich/727_Fundamentals of Computing and Data Dis
##   3  -- 5 columns 6 columns '/Users/angelina/Desktop/UMich/727_Fundamentals of Computing and Data Dis
##   4  -- 5 columns 6 columns '/Users/angelina/Desktop/UMich/727_Fundamentals of Computing and Data Dis
##   6  -- 5 columns 6 columns '/Users/angelina/Desktop/UMich/727_Fundamentals of Computing and Data Dis
## ... ... ......... ......... ...............................................................................
## See problems(...) for more details.
```

```
head(angell_stata)
```

```
## # A tibble: 6 x 5
##   city       morint ethhet geomob region
##   <chr>       <dbl>  <dbl>  <dbl> <chr>
## 1 Rochester    19     20.6   15    E
## 2 Syracuse     17     15.6   20.2  E
## 3 Worcester    16.4   22.1   13.6  E
## 4 Erie         16.2   14     14.8  E
## 5 Milwaukee    15.8   17.4   17.6  MW
## 6 Bridgeport   15.3   27.9   17.5  E
```

```
head(angell_txt)
```

```
## # A tibble: 6 x 5
```

```
##   city       morint ethhet geomob region
##   <chr>       <dbl>  <dbl>  <dbl> <chr>
## 1 Rochester    19     20.6   15   E
## 2 Syracuse     17     15.6   20.2 E
## 3 Worcester    16.4   22.1   13.6 E
## 4 Erie         16.2   14     14.8 E
## 5 Milwaukee    15.8   17.4   17.6 MW
## 6 Bridgeport   15.3   27.9   17.5 E
```

4) What are the differences between angell_stata and angell_txt? Are there differences in the classes of the individual columns?

```
str(angell_stata)
```

```
## tibble [43 x 5] (S3: tbl_df/tbl/data.frame)
##  $ city  : chr [1:43] "Rochester" "Syracuse" "Worcester" "Erie" ...
##   ..- attr(*, "format.stata")= chr "%15s"
##  $ morint: num [1:43] 19 17 16.4 16.2 15.8 ...
##   ..- attr(*, "format.stata")= chr "%9.0g"
##  $ ethhet: num [1:43] 20.6 15.6 22.1 14 17.4 ...
##   ..- attr(*, "format.stata")= chr "%9.0g"
##  $ geomob: num [1:43] 15 20.2 13.6 14.8 17.6 ...
##   ..- attr(*, "format.stata")= chr "%9.0g"
##  $ region: chr [1:43] "E" "E" "E" "E" ...
##   ..- attr(*, "format.stata")= chr "%9s"
```

```
str(angell_txt)
```

```
## spc_tbl_ [43 x 5] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ city  : chr [1:43] "Rochester" "Syracuse" "Worcester" "Erie" ...
##  $ morint: num [1:43] 19 17 16.4 16.2 15.8 15.3 15.2 14.3 14.2 14.1 ...
##  $ ethhet: num [1:43] 20.6 15.6 22.1 14 17.4 27.9 22.3 23.7 10.6 12.7 ...
##  $ geomob: num [1:43] 15 20.2 13.6 14.8 17.6 17.5 14.7 23.8 19.4 31.9 ...
##  $ region: chr [1:43] "E" "E" "E" "E" ...
##  - attr(*, "problems")= tibble [29 x 5] (S3: tbl_df/tbl/data.frame)
##   ..$ row     : int [1:29] 1 2 3 4 6 7 9 12 15 18 ...
##   ..$ col     : chr [1:29] NA NA NA NA ...
##   ..$ expected: chr [1:29] "5 columns" "5 columns" "5 columns" "5 columns" ...
##   ..$ actual  : chr [1:29] "6 columns" "6 columns" "6 columns" "6 columns" ...
##   ..$ file    : chr [1:29] "'/Users/angelina/Desktop/UMich/727_Fundamentals of Computing and Data Di
##  - attr(*, "spec")=
##   .. cols(
##   ..   city = col_character(),
##   ..   morint = col_double(),
##   ..   ethhet = col_double(),
##   ..   geomob = col_double(),
##   ..   region = col_character()
##   .. )
```

```r
stata_classes <- sapply(angell_stata, class)
txt_classes <- sapply(angell_txt, class)
comparison <- data.frame(
  Column = names(stata_classes),
  Stata_Class = stata_classes,
  Txt_Class = txt_classes
)
comparison
```

```
##        Column Stata_Class Txt_Class
## city     city   character character
## morint morint     numeric   numeric
## ethhet ethhet     numeric   numeric
## geomob geomob     numeric   numeric
## region region   character character
```

The table shows that city and region are classified as character in both datasets, while morint, ethhet, and geomob are classified as numeric in both datasets. Therefore, there are no significant differences between angell_stata and angell_txt in terms of column classes. However, in the raw document, the column names in angell_txt which are X0 to X6 are different from those in angell_stata which are "city", "morint", "ethhet", "geomob", "region".

5) Make any updates necessary so that angell_txt is the same as angell_stata.

When I was reading the txt document, I added col_names = c("city", "morint", "ethhet", "geomob", "region") in line 24 to ensure that angell_txt has the same column names as angell_stata.

6) Describe the Ethnic Heterogeneity variable. Use descriptive statistics such as mean, median, standard deviation, etc. How does it differ by region?

```r
library(dplyr)
ethhet_stats <- angell_stata %>%
  summarise(
    mean = mean(ethhet, na.rm = TRUE),
    median = median(ethhet, na.rm = TRUE),
    sd = sd(ethhet, na.rm = TRUE),
    min = min(ethhet, na.rm = TRUE),
    max = max(ethhet, na.rm = TRUE)
  )
ethhet_stats
```

```
## # A tibble: 1 x 5
##    mean median    sd   min   max
##   <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1  31.4   23.7  20.4  10.6  84.5
```

```r
ethhet_by_region <- angell_stata %>%
  group_by(region) %>%
  summarise(
```

```
    Mean = mean(ethhet, na.rm = TRUE),
    Median = median(ethhet, na.rm = TRUE),
    SD = sd(ethhet, na.rm = TRUE),
    Min = min(ethhet, na.rm = TRUE),
    Max = max(ethhet, na.rm = TRUE)
  )
ethhet_by_region
```

```
## # A tibble: 4 x 6
##   region  Mean Median    SD   Min   Max
##   <chr>  <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 E       23.5   22.1 10.8   10.6  45.8
## 2 MW      21.7   19.2  9.08  10.7  39.7
## 3 S       52.5   53.8 21.4   20.7  84.5
## 4 W       16.5   16.1  4.16  12.3  23.9
```

There are significant differences in Ethnic Heterogeneity values across different regions, especially in the South(S), where the median and range are noticeably higher than in other regions, which might indicate the South has the more diverse ethnic cities. Furthermore, the variability across regions which measured by standard deviation also differs, showing that ethnic heterogeneity is more concentrated in some regions and more spread out in others. For example, the West(W) has the lowest level of Ethnic Heterogeneity, and its values are relatively stable (standard deviation of 4.16).

7) Install the "MASS" package, load the package. Then, load the Boston dataset.

```
install.packages("MASS")
```

```
##
## The downloaded binary packages are in
##  /var/folders/8_/p0sgn35n70bcqqr1rc6jxwyr0000gn/T//Rtmp7MIxHh/downloaded_packages
```

```
library(MASS)
data("Boston")
```

8) What is the type of the Boston object?

```
typeof(Boston)
```

```
## [1] "list"
```

9) What is the class of the Boston object?

```
class(Boston)
```

```
## [1] "data.frame"
```

10) How many of the suburbs in the Boston data set bound the Charles river?

```
table(Boston$chas)
```

```
##
##   0   1
## 471  35
```

```
sum(Boston$chas == 1)
```

```
## [1] 35
```

There are 35 suburbs in the Boston dataset that bound the Charles River, as indicated by the chas variable having a value of 1.

11) Do any of the suburbs of Boston appear to have particularly high crime rates? Tax rates? Pupil-teacher ratios? Comment on the range of each variable.

```
summary(Boston$crim)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##   0.00632   0.08204   0.25651   3.61352   3.67708 88.97620
```

```
summary(Boston$tax)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   187.0   279.0   330.0   408.2   666.0   711.0
```

```
summary(Boston$ptratio)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   12.60   17.40   19.05   18.46   20.20   22.00
```

Crime rates: The range of crime rates is quite large, from a low of 0.00632 to a high of 88.97620, indicating a significant variation in crime rates across the suburbs of Boston. Additionally, most suburbs have lower crime rates, as shown by the median of 0.25651.

Tax rates: Most suburbs' tax rates are concentrated between 279 and 330. However, the third quartile (Q3) is 666, which means that 25% of the suburbs have tax rates close to the maximum of 711 and may have higher fiscal demands.

Pupil-teacher ratios: There is a smaller range in pupil-teacher ratios, with most suburbs' ratios close to 18 to 20.

12) Describe the distribution of pupil-teacher ratio among the towns in this data set that have a per capita crime rate larger than 1. How does it differ from towns that have a per capita crime rate smaller than 1?

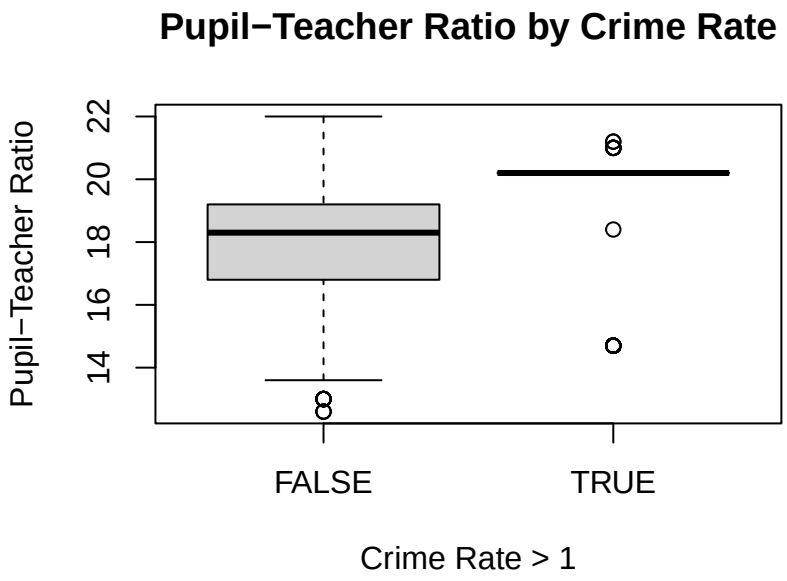
```r
high_crime_towns <- Boston[Boston$crim > 1, ]
low_crime_towns <- Boston[Boston$crim < 1, ]
summary(high_crime_towns$ptratio)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   14.70   20.20   20.20   19.29   20.20   21.20
```

```r
summary(low_crime_towns$ptratio)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   12.60   16.80   18.30   18.02   19.20   22.00
```

```r
boxplot(ptratio ~ crim > 1, data = Boston,
        main = "Pupil-Teacher Ratio by Crime Rate",
        xlab = "Crime Rate > 1",
        ylab = "Pupil-Teacher Ratio")
```



First, I used a boxplot to visualize the result. The pupil-teacher ratio in suburbs with a crime rate greater than 1 is relatively high, and the data is very concentrated around 20.20, indicating that the pupil-teacher ratio in these high-crime areas is high, and most of the suburbs have relatively scarce educational resources. On the other hand, the minimum value of 12.60 and the Q1 of 16.80 indicate that some suburbs with a low crime rate have a relatively low pupil-teacher ratio, which shows that these suburbs have more adequate educational resources.

13) Write a function that calculates 95% confidence intervals for a point estimate. The function should be called my_CI. When called with my_CI(2, 0.2), the function should print out "The 95% CI upper bound of point estimate 2 with standard error 0.2 is 2.392. The lower bound is 1.608."

```
my_CI <- function(point_estimate, standard_error) {
  upper_bound <- point_estimate + 1.96 * standard_error
  lower_bound <- point_estimate - 1.96 * standard_error

  message <- paste0("The 95% CI upper bound of point estimate ", point_estimate,
                    " with standard error ", standard_error,
                    " is ", round(upper_bound, 3), ". The lower bound is ",
                    round(lower_bound, 3), ".")
  print(message)
}

my_CI(2, 0.2)
```

```
## [1] "The 95% CI upper bound of point estimate 2 with standard error 0.2 is 2.392. The lower bound is
```

14) Create a new function called my_CI2 that does that same thing as the my_CI function but outputs a vector of length 2 with the lower and upper bound of the confidence interval instead of printing out the text. Use this to find the 95% confidence interval for a point estimate of 0 and standard error 0.4.

```
my_CI2 <- function(point_estimate, standard_error) {
  lower_bound <- point_estimate - 1.96 * standard_error
  upper_bound <- point_estimate + 1.96 * standard_error
  return(c(lower_bound, upper_bound))
}

ci <- my_CI2(0, 0.4)
print(ci)
```

```
## [1] -0.784  0.784
```

15) Update the my_CI2 function to take any confidence level instead of only 95%. Call the new function my_CI3. You should add an argument to your function for confidence level.

```
my_CI3 <- function(point_estimate, standard_error, conf_level = 0.95) {
  z_value <- qnorm(1 - (1 - conf_level) / 2)

  lower_bound <- point_estimate - z_value * standard_error
  upper_bound <- point_estimate + z_value * standard_error

  return(c(lower_bound, upper_bound))
}
# Test 95% confidence interval
my_CI3(0, 0.4, 0.95)
```

```
## [1] -0.7839856  0.7839856
```

```
# Test 90% confidence interval
my_CI3(0, 0.4, 0.90)
```

```
## [1] -0.6579415  0.6579415
```

16) Without hardcoding any numbers in the code, find a 99% confidence interval for Ethnic Heterogeneity in the Angell dataset. Find the standard error by dividing the standard deviation by the square root of the sample size.

```
n <- length(angell_stata$ethhet)
sd_ethhet <- sd(angell_stata$ethhet, na.rm = TRUE)
se_ethhet <- sd_ethhet / sqrt(n)
mean_ethhet <- mean(angell_stata$ethhet, na.rm = TRUE)
ci_ethhet_99 <- my_CI3(mean_ethhet, se_ethhet, conf_level = 0.99)
ci_ethhet_99
```

```
## [1] 23.35425 39.38993
```

17) Write a function that you can apply to the Angell dataset to get 95% confidence intervals. The function should take one argument: a vector. Use if-else statements to output NA and avoid error messages if the column in the data frame is not numeric or logical.

```
CI_95 <- function(x) {
  if (is.numeric(x) || is.logical(x)) {
    mean_value <- mean(x, na.rm = TRUE)
    SE <- sd(x, na.rm = TRUE) / sqrt(sum(!is.na(x)))

    # qnorm(0.975) gives approximately 1.96, which is the Z-value for a 95% confidence interval.
    Z <- 1.96

    lower_CI <- mean_value - Z * SE
    upper_CI <- mean_value + Z * SE

    return(c(lower_CI, upper_CI))
  } else {

    return(NA)
  }
}

CI_95(angell_stata$ethhet)
```

```
## [1] 25.27116 37.47303
```