Social Sciences Intro to Statistics

Pset 3: Due MONTH, DATE, YEAR at 11:59pm

Belle Lee

2024-06-15

Overview

Welcome to your third pset of the course. This problem set is intended to give you some practice becoming familiar with descriptive statistics. In this problem set, we are asking you to: create an R project, render your file, load and investigate an R data frame that is stored on the web, and apply some basic functions to better understand distributions.

- Note: Change the values of the YAML header above to your name and the date.

Question 1: Creating an R project

Create an R project

- Create a folder where you want to save files associated with problem set 3. Let's call that folder "problemset3", but you can name it whatever you want.
  - For instance, it could be SSS » problem_sets » problemset3.
- In RStudio, click on "File" » "New Project" » "Existing Directory" » "Browse".
- Browse to find and select your problem set 3 folder.
- Click on "Create Project".
  - An R project file has the extension ".Rproj".
  - The name of the file should be "problemset3.Rproj", or whatever you named the folder.

Save this problemset2.qmd file anywhere in the folder named problemset3.

- At the top of this .qmd file, type in your first and last name in the appropriate place in the YAML header (e.g. "Belle Lee").

- in the date field of the YAML header, insert the date within quotations (any date format is fine).
- Now click the “Render” button near the top of your RStudio window (icon with blue arrow sign) or drop down “File” and select “Render Document”.
    - Alternatively you can use the shortcut: **Cmd/Ctrl + Shift + k**.
    - *Note*: One goal of this assignment is to make sure you are able to render without running into errors.

Question 2: Hypothesis Testing

1. If you do not know the true value of a population parameter, can you still make a hypothesis?

ANSWER:

2. Explain what an alternative hypothesis is. You may provide an example of an alternative hypothesis in relation to a null hypothesis.

ANSWER:

Question 3: Components of a T-test

1. Explain the logic of a test statistic, what would it calculate?

ANSWER:

2. What are the components of a t-test?

ANSWER:

3. Load the ipeds data below from the course website.

```
library(tidyverse)
library(ggplot2)
library(labelled)
library(patchwork)

# Load ipeds dataset from course website
load(url('https://raw.githubusercontent.com/bcl96/Social-Sciences-Stats/main/data/ipeds/outpu

# Create ipeds data frame with fewer variables/observations
df_ipeds_pop <- panel_data %>%
  # keep data from fall 2022
  filter(year == 2022) %>%
  # which universities to keep:
    # 2015 carnegie classification: keep research universities (15,16,17) and master's univer
  filter(c15basic %in% c(15,16,17,18,19,20)) %>%
  # which variables to keep
  select(instnm,unitid,opeid6,opeid,control,c15basic,stabbr,city,zip,locale,obereg, # basic i
         tuition6,fee6,tuition7,fee7, # avg tuition and fees for full-time grad, in-state and
         isprof3,ispfee3,osprof3,ospfee3, # avg tuition and fees for MD, in-state and out-of-
         isprof9,ispfee9,osprof9,ospfee9, # avg tuition and fees for Law, in-state and out-of
         chg4ay3,chg7ay3,chg8ay3) %>% # [undergraduate] books+supplies; off-campus (not with
  # rename variables; syntax <new_name> = <old_name>
  rename(region = obereg, # revion
         tuit_grad_res = tuition6, fee_grad_res = fee6, tuit_grad_nres = tuition7, fee_grad_n
         tuit_md_res = isprof3, fee_md_res = ispfee3, tuit_md_nres = osprof3, fee_md_nres = o
         tuit_law_res = isprof9, fee_law_res = ispfee9, tuit_law_nres = osprof9, fee_law_nres
         books_supplies = chg4ay3, roomboard_off = chg7ay3, oth_expense_off = chg8ay3) %>% #
  # create measures of tuition+fees
  mutate(
    tuitfee_grad_res = tuit_grad_res + fee_grad_res, # graduate, state resident
    tuitfee_grad_nres = tuit_grad_nres + fee_grad_nres, # graduate, non-resident
    tuitfee_md_res = tuit_md_res + fee_md_res, # MD, state resident
    tuitfee_md_nres = tuit_md_nres + fee_md_nres, # MD, non-resident
    tuitfee_law_res = tuit_law_res + fee_law_res, # Law, state resident
    tuitfee_law_nres = tuit_law_nres + fee_law_nres) %>% # Law, non-resident
  # create measures of cost-of-attendance (COA) as the sum of tuition, fees, book, living exp
  mutate(
    coa_grad_res = tuit_grad_res + fee_grad_res + books_supplies + roomboard_off + oth_expens
    coa_grad_nres = tuit_grad_nres + fee_grad_nres + books_supplies + roomboard_off + oth_exp
    coa_md_res = tuit_md_res + fee_md_res + books_supplies + roomboard_off + oth_expense_off
    coa_md_nres = tuit_md_nres + fee_md_nres + books_supplies + roomboard_off + oth_expense_o
    coa_law_res = tuit_law_res + fee_law_res + books_supplies + roomboard_off + oth_expense_o
    coa_law_nres = tuit_law_nres + fee_law_nres + books_supplies + roomboard_off + oth_expens
  # keep only observations that have non-missing values for the variable coa_grad_res
    # this does cause us to lose some interesting universities, but doing this will eliminate
  filter(!is.na(coa_grad_res))

# Add variable labels to the tuit+fees variables and coa variables
  # tuition + fees variables
    var_label(df_ipeds_pop[['tuitfee_grad_res']]) <- 'graduate, full-time, resident; avg tuit
    var_label(df_ipeds_pop[['tuitfee_grad_nres']]) <- 'graduate, full-time, non-resident; avg
    var_label(df_ipeds_pop[['tuitfee_md_res']]) <- 'MD, full-time, state resident; avg tuitio
    var_label(df_ipeds_pop[['tuitfee_md_nres']]) <- 'MD, full-time, non-resident; avg tuition
    var_label(df_ipeds_pop[['tuitfee_law_res']]) <- 'Law, full-time, state resident; avg tui
    var_label(df_ipeds_pop[['tuitfee_law_nres']]) <- 'Law, full-time, non-resident; avg tuiti

  # COA variables
    var_label(df_ipeds_pop[['coa_grad_res']]) <- 'graduate, full-time, state resident COA; ==
    var_label(df_ipeds_pop[['coa_grad_nres']]) <- 'graduate, full-time, non-resident COA; ==
    var_label(df_ipeds_pop[['coa_md_res']]) <- 'MD, full-time, state resident COA; == tuition
    var_label(df_ipeds_pop[['coa_md_nres']]) <- 'MD, full-time, non-resident COA; == tuition
    var_label(df_ipeds_pop[['coa_law_res']]) <- 'Law, full-time, state resident COA; == tuiti
    var_label(df_ipeds_pop[['coa_law_nres']]) <- 'Law, full-time, non-resident COA; == tuitio

df_ipeds_pop %>% glimpse()
#> Rows: 965
#> Columns: 38
#> $ instnm           <chr> "Alabama A & M University", "University of Alabama a~
#> $ unitid           <dbl> 100654, 100663, 100706, 100724, 100751, 100830, 1008~
#> $ opeid6           <chr> "001002", "001052", "001055", "001005", "001051", "0~
#> $ opeid            <chr> "00100200", "00105200", "00105500", "00100500", "001~
#> $ control          <dbl+lbl> 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 2, 1, 1, 2, ~
#> $ c15basic         <dbl+lbl> 18, 15, 16, 19, 16, 18, 16, 20, 18, 18, 19, 18, ~
#> $ stabbr           <chr+lbl> "AL", "AL", "AL", "AL", "AL", "AL", "AL", "AL", ~
#> $ city             <chr> "Normal", "Birmingham", "Huntsville", "Montgomery", ~
#> $ zip              <chr> "35762", "35294-0110", "35899", "36104-0271", "35487~
#> $ locale           <dbl+lbl> 12, 12, 12, 12, 13, 12, 13, 12, 23, 43, 21, 13, ~
#> $ region           <dbl+lbl> 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, ~
#> $ tuit_grad_res    <dbl> 10128, 8424, 10632, 7416, 11100, 7812, 10386, 15325,~
#> $ fee_grad_res     <dbl> 1414, 0, 1054, 2740, 690, 766, 1784, 900, 1000, 190,~
#> $ tuit_grad_nres   <dbl> 20160, 19962, 24430, 14832, 31460, 17550, 31158, 153~
#> $ fee_grad_nres    <dbl> 1414, 0, 1054, 2740, 690, 766, 1784, 900, 1000, 190,~
#> $ tuit_md_res      <dbl> NA, 31198, NA, NA, 31198, NA, NA, NA, NA, NA, NA, NA~
#> $ fee_md_res       <dbl> NA, 3464, NA, NA, 0, NA, NA, NA, NA, NA, NA, NA, NA,~
#> $ tuit_md_nres     <dbl> NA, 62714, NA, NA, 62714, NA, NA, NA, NA, NA, NA, NA~
#> $ fee_md_nres      <dbl> NA, 3464, NA, NA, 0, NA, NA, NA, NA, NA, NA, NA, NA,~
#> $ tuit_law_res     <dbl> NA, NA, NA, NA, 24080, NA, NA, 39000, NA, NA, NA, NA~
#> $ fee_law_res      <dbl> NA, NA, NA, NA, 300, NA, NA, 325, NA, NA, NA, NA, 65~
#> $ tuit_law_nres    <dbl> NA, NA, NA, NA, 44470, NA, NA, 39000, NA, NA, NA, NA~
#> $ fee_law_nres     <dbl> NA, NA, NA, NA, 300, NA, NA, 325, NA, NA, NA, NA, 65~
#> $ books_supplies   <dbl> 1600, 1200, 2416, 1600, 800, 1200, 1200, 1800, 998, ~
#> $ roomboard_off    <dbl> 9520, 14330, 11122, 7320, 14426, 10485, 14998, 8020,~
#> $ oth_expense_off  <dbl> 3090, 6007, 4462, 5130, 4858, 4030, 6028, 4600, 3318~
#> $ tuitfee_grad_res <dbl> 11542, 8424, 11686, 10156, 11790, 8578, 12170, 16225~
#> $ tuitfee_grad_nres <dbl> 21574, 19962, 25484, 17572, 32150, 18316, 32942, 162~
#> $ tuitfee_md_res    <dbl> NA, 34662, NA, NA, 31198, NA, NA, NA, NA, NA, NA, NA~
#> $ tuitfee_md_nres   <dbl> NA, 66178, NA, NA, 62714, NA, NA, NA, NA, NA, NA, NA~
#> $ tuitfee_law_res   <dbl> NA, NA, NA, NA, 24380, NA, NA, 39325, NA, NA, NA, NA~
#> $ tuitfee_law_nres  <dbl> NA, NA, NA, NA, 44770, NA, NA, 39325, NA, NA, NA, NA~
#> $ coa_grad_res      <dbl> 25752, 29961, 29686, 24206, 31874, 24293, 34396, 306~
#> $ coa_grad_nres     <dbl> 35784, 41499, 43484, 31622, 52234, 34031, 55168, 306~
#> $ coa_md_res        <dbl> NA, 56199, NA, NA, 51282, NA, NA, NA, NA, NA, NA, NA~
#> $ coa_md_nres       <dbl> NA, 87715, NA, NA, 82798, NA, NA, NA, NA, NA, NA, NA~
#> $ coa_law_res       <dbl> NA, NA, NA, NA, 44464, NA, NA, 53745, NA, NA, NA, NA~
#> $ coa_law_nres      <dbl> NA, NA, NA, NA, 64854, NA, NA, 53745, NA, NA, NA, NA~

##########
########## Create data frame of generated variables, with each variable meant to represent th
##########


num_obs <- 10000

# Generate normal distribution w/ custom mean and sd
set.seed(124)
norm_dist <- rnorm(n = num_obs, mean = 50, sd = 5)

# Generate right-skewed distribution
set.seed(124)
rskew_dist <- rbeta(n = num_obs, shape1 = 2, shape2 = 5)

# Generate left-skewed distribution
set.seed(124)
lskew_dist <- rbeta(n = num_obs, shape1 = 5, shape2 = 2)

# Generate standard normal distribution (default is mean = 0 and sd = 1)
set.seed(124)
stdnorm_dist <- rnorm(n = num_obs, mean = 0, sd = 1)  # equivalent to rnorm(10)

# Create dataframe
df_generated_pop <- data.frame(norm_dist, rskew_dist, lskew_dist, stdnorm_dist)

# drop individual objects associated with each variable
rm(norm_dist,rskew_dist,lskew_dist,stdnorm_dist)
rm(num_obs)
```

```
##########
########## Create sample versions of generated population data frame and IPEDS population da
##########

# create sample version of our generated data
  set.seed(124) # set seed so that everyone ends up with the same random sample

  df_generated_sample <- df_generated_pop %>% sample_n(size = 200)
  df_generated_sample %>% glimpse()
#> Rows: 200
#> Columns: 4
#> $ norm_dist    <dbl> 42.70513, 50.24400, 61.29008, 45.47494, 44.74406, 47.9912~
#> $ rskew_dist   <dbl> 0.34451771, 0.31359906, 0.09375337, 0.05581678, 0.0744584~
#> $ lskew_dist   <dbl> 0.6554823, 0.6864009, 0.9062466, 0.9441832, 0.9255415, 0.~
#> $ stdnorm_dist <dbl> -1.45897348, 0.04880097, 2.25801577, -0.90501164, -1.0511~


# create sample version of our ipeds data

  set.seed(124) # set seed so that everyone ends up with the same random sample

  df_ipeds_sample <- df_ipeds_pop %>% sample_n(size = 200)

  # compare mean of coa_grad_res between population and sample
  mean(df_ipeds_pop$coa_grad_res, na.rm = TRUE)
#> [1] 32528.35
  mean(df_ipeds_sample$coa_grad_res, na.rm = TRUE)
#> [1] 31620.8
```

4. Let's investigate the in-state fees for full-time graduate students in `df_ipeds_sample`. If the following is our null hypothesis, please state in words the alternative hypothesis

Null hypothesis H_0 - (in words) H_0 : the population mean cost of in-state fees for full-time (resident) graduate students, μ_Y, is $800.

ANSWER:

5. Calculate the t-test value by hand

6. Calculate the t-test value by function

7. Did you get the same t-test value by hand and by function? What does this t-test value tell us? What would we need to know if this is statistically significant?

ANSWER:

Question 4: Comparing Two Groups

1. What does a p-value represent?

ANSWER:

2. What happens when we have a p-value that is less than an alpha level of 0.05?

ANSWER:

Render to pdf and submit problem set

Render to pdf by clicking the "Render" button near the top of your RStudio window (icon with blue arrow) or drop down "File" and select "Render to PDF"

- Go to the [class website] (Need to fill in classwebsite) and under the "Readings & Assignments" » "Week 1" tab, click on the "Problem set 1 submission link"

- Submit both .qmd and pdf files

- Use this naming convention "lastname_firstname_ps#" for your .qmd and pdf files (e.g. lee_belle_ps1.qmd & lee_belle_ps1.pdf)