STAT 231: Problem Set 3B

Alastair Poole

due by 5 PM on Friday, March 12

This homework assignment is designed to help you futher ingest, practice, and expand upon the material covered in class over the past week(s). You are encouraged to work with other students, but all code and text must be written by you, and you must indicate below who you discussed the assignment with (if anyone).

Steps to proceed:

1. In RStudio, go to File > Open Project, navigate to the folder with the course-content repo, select the course-content project (course-content.Rproj), and click "Open"

2. Pull the course-content repo (e.g. using the blue-ish down arrow in the Git tab in upper right window)

3. Copy ps3B.Rmd from the course repo to your repo (see page 6 of the GitHub Classroom Guide for Stat231 if needed)

4. Close the course-content repo project in RStudio

5. Open YOUR repo project in RStudio

6. In the ps3B.Rmd file in YOUR repo, replace "YOUR NAME HERE" with your name

7. Add in your responses, committing and pushing to YOUR repo in appropriate places along the way

8. Run "Knit PDF"

9. Upload the pdf to Gradescope. Don't forget to select which of your pages are associated with each problem. *You will not get credit for work on unassigned pages (e.g., if you only selected the first page but your solution spans two pages, you would lose points for any part on the second page that the grader can't see).*

If you discussed this assignment with any of your peers, please list who here:

ANSWER: Ethan Samuels, Ethan Gilman

Shiny app

1. Finish your app from Lab04b and add your app code to the R code chunk below:

(1) update the Lab04b app to still explore the `electric_skateboards` dataset, but with different app functionality (e.g. different widgets, variables, layout, theme. . .); OR
(2) use it as a template to create a Shiny app for a different dataset, choosing from:

- `candy_rankings` (candy characteristics and popularity)
- `hate_crimes` (hate crimes in US states, 2010-2015)
- `mad_men` (tv performers and their post-show career)
- `ncaa_w_bball_tourney` (women's NCAA div 1 basketball tournament, 1982-2018)
- `nfl_suspensions` (NFL suspensions, 1946-2014)

These five datasets are part of the `fivethirtyeight` package and their variable definitions are included in a pdf posted to the Moodle course page.

If using the `electric_skateboards` dataset, be sure to update:

- at least 2 different widgets; and
- the layout (e.g. not in tabs or different page layout) or the theme

  – check out: https://rstudio.github.io/shinythemes/

- like a challenge? incorporate one of the click, hover or brush features

  – check out: https://shiny.rstudio.com/articles/plot-interaction.html

```r
## keep eval = FALSE in this code chunk option so your app doesn't
## try to run when knitting the document

## add your app code here (including any packages and datasets loaded,
## the ui call, the server call, and the shinyApp call)

library(shiny)
library(shinythemes)
library(tidyverse)
library(DT)
library(ggrepel)
library(fivethirtyeight)


###############
# import data #
###############
nfl <- fivethirtyeight::nfl_suspensions
nfl2 <- nfl_suspensions %>%
  filter(is.na(category) == FALSE) %>%
  mutate(numericvar = parse_number(games)) %>%
  filter(is.na(numericvar) == FALSE)
###############################################################
# define choice values and labels for widgets (user inputs) #
###############################################################
# define vectors for choice values and labels
# can then refer to them in server as well (not just in defining widgets)
```

```r
# for TAB 1 (HISTOGRAM) widgets:
# for selectInput, 'choices' object should be a NAMED LIST
hist_choice_values <- c("name","team","games", "year")
hist_choice_names <- c("Name","Team","Games", "Year")
names(hist_choice_values) <- hist_choice_names

# for checkboxGroupInput
category_choices <-  unique(nfl_suspensions$category)

# for TAB 2 (SCATTERPLOT) widgets:

# for selectizeInput choices for skateboard name, pull directly from data
name_choices <- unique(nfl_suspensions$name)

# for TAB 3 (TABLE) widgets:
# for selectizeInput choices for company name, pull directly from data
cmpy_choices <- unique(nfl_suspensions$team)

############
#    ui    #
############
ui <- navbarPage(

  title="NFL Suspensions",

  tabPanel(
    title = "Histogram",
    sidebarLayout(
      sidebarPanel(
        selectInput(inputId = "histvar"
                    , label = "Choose a variable of interest to plot:"
                    , choices = hist_choice_values
                    , selected = "team"),
        checkboxGroupInput(inputId = "category"
                           , label = "Include Category:"
                           , choices = category_choices
                           , selected = "Personal conduct")
      ),
      mainPanel(
        plotOutput(outputId = "hist")
      )
    )
  ),

  tabPanel(
    title = "Scatterplot",

    sidebarLayout(

      sidebarPanel(
        selectizeInput(inputId = "id_name"
                       , label = "Identify players(s) in the scatterplot:"
                       , choices = name_choices
```

```
                              , selected = NULL
                              , multiple = TRUE)
      ),
      mainPanel(
        plotOutput(outputId = "scatter")
      )
    )
  ),

  tabPanel(
    title = "Table",

    sidebarLayout(
      sidebarPanel(
        selectizeInput(inputId = "team"
                       , label = "Choose one or more teams:"
                       , choices = cmpy_choices
                       , selected = "DIYElectric"
                       , multiple = TRUE)
      ),
      mainPanel(
        DT::dataTableOutput(outputId = "table")
      )
    )
  ),
  tabPanel(
    title = "Original Graph",

    sidebarLayout(
      sidebarPanel(
        tags$div(
          HTML(paste("Original figure was presented by "
                     , tags$a(href="https://fivethirtyeight.com/features/nfl-domestic-violence-policy-s
                              "538 NFL Suspensions")
                     , sep = ""))
        )
      ),
      mainPanel(
        h3("Information overload!"),
        plotOutput(outputId = "original")
      )
    )
  )
)


############
# server   #
############
server <- function(input,output){

  # TAB 1: HISTOGRAM
  data_for_hist <- reactive({
    data <- filter(nfl_suspensions, category %in% input$category)
```

```r
  })

  output$hist <- renderPlot({
    ggplot(data = data_for_hist(), aes_string(x = input$histvar)) +
      geom_bar(color = "#2c7fb8", fill = "#7fcdbb", alpha = 0.7) +
      labs(x = hist_choice_names[hist_choice_values == input$histvar]
           , y = "Number of Suspensions") +
      theme(axis.text.x = element_text(angle = 90, hjust = 1))
  })

  # TAB 2: INTERACTIVE SCATTERPLOT
  output$scatter <- renderPlot({
      ggplot(data = nfl2, aes_string(x="year", y="numericvar")) +
      geom_point(color = "#2c7fb8") +
      labs(x = "Year", y = "Games", title = "NFL Suspensions") +
      geom_label_repel(data = filter(nfl2, name %in% input$id_name)
                       , aes(label = name), show.legend = FALSE) +
      facet_wrap(~category, ncol = 3)
  })

  # TAB 3: TABLE
  data_for_table <- reactive({
    data <- filter(nfl_suspensions, team %in% input$team)
  })

  output$table <- DT::renderDataTable({
    data_for_table()
  })

  # TAB 4: RE-CREATION OF ORIGINAL FIGURE (STATIC)
  output$original <- renderPlot({
    ggplot(data = nfl_suspensions, aes(x=year, y=games, color=team
                                       , shape = category)) +
      geom_point() +
      geom_text(aes(label = team), hjust = 0, nudge_x = 0.05, size=3) +
      labs(x = "Year", y = "Games"
           , title = "NFL Suspensions"
           , shape = "Suspension Category") +
      guides(color = FALSE)
  })
}

####################
# call to shinyApp #
####################
shinyApp(ui = ui, server = server)

# Your turn.  Copy this code as a template into a new app.R file (WITHIN A FOLDER
# named something different than your other Shiny app folders).  Then, either
# (1) update this template to still explore the skateboards dataset, but with
#     different app functionality (e.g. different widgets, variables, layout, theme...);
#   OR
# (2) use this as a template to create a Shiny app for a different dataset
```

```
#       from the fivethirtyeight package:
#       either candy_rankings (candy characteristics and popularity)
#               hate_crimes (hate crimes in US states, 2010-2015)
#               mad_men (tv performers and their post-show career),
#               ncaa_w_bball_tourney (women's NCAA div 1 basketball tournament, 1982-2018),
#           or nfl_suspensions (NFL suspensions, 1946-2014)
#         these five datasets are part of the fivethirtyeight package
#          and their variable definitions are included in pdfs posted to the Moodle course page
```

2. Publish your app. Then, go to the Google group conversation "PS3B: Shiny Apps" and reply to the message with (1) the URL to your published Shiny app; and (2) a paragraph explaining what story your Shiny app is telling, and how the interactivity you created enhances the telling of that story.

   ANSWER: Do not include anything here. The link to your app and the paragraph should be posted to the "PS3B: Shiny Apps" Google conversation thread.