Tarea 3

Ángela Vieyto

02/07/2021

Ejercicio 1

1. Crear dos pestañas una con el nombre Bivariado y la otra Univariado.

```
ui <- fluidPage(
        titlePanel("Datos de Propina"),
        sidebarLayout(
          sidebarPanel(
            selectInput(
              "varcolor",
              "Variable en color",
              c("sexo", "fuma", "dia", "momento")
            ),
            mainPanel(
              tabsetPanel(
                tabPanel("Bivariado",
                         h2("Diagrama de dispersión", align = "center"),
                         plotOutput("scat")
                         ),
                tabPanel("Univariado")
              )
            )
          )
        )
      )
```

2. Crea en la pestaña Bivariado una tabla debajo de la figura que contenga la media y el desvío de las variables propina y total agrupando por la variable de color (la tabla debe ser reactiva a la variable seleccionada), esto implica que la tabla debe modificarse cuando se selecciona una variable de color. Usar el paquete DT para hacer la tabla.

```
ui <- fluidPage(
        sidebarLayout(
          mainPanel(
            tabsetPanel(
              tabPanel("Bivariado",
                       h2("Diagrama de dispersión", align = "center"),
                       plotOutput("scat"),
                       dataTableOutput("tabla")
              )
```

```
            )
          )
        )
      )

server <- function(input, output) {

tabla <- reactive({
          propinas %>%
            group_by(.data[[input$varcolor]]) %>%
              summarise(
                mean_prop = mean(propina),
                sd_propina = sd(propina),
                mean_total = mean(total),
                sd_total = sd(total)
              ) %>% datatable()
        })

output$tabla <- renderDT({tabla()})

}
```

2.1. Genera un selectInput llamado "digitos" que defina la cantidad de decimales a mostrar en la tabla anterior (0, 1 ó 2 decimales a mostrar). Para esto, agrega una linea extra en el calculo de la tabla que use la función across. Recordá que across solo puede usarse dentro de los verbos de dplyr, es decir, summarise(across(. . .)) sería la forma correcta. Tip: fijate que solo podes redondear variables númericas, la función where() es tu amiga (where y algo más. . .).

```
ui <- fluidPage(
        sidebarLayout(
          sidebarPanel(
            selectInput("digitos", "Cantidad de decimales",
                        c(0, 1, 2)
            )
          )
        )
      )

server <- function(input, output) {

tabla <- reactive({
          propinas %>%
            group_by(.data[[input$varcolor]]) %>%
              summarise(
                mean_prop = mean(propina),
                sd_propina = sd(propina),
                mean_total = mean(total),
                sd_total = sd(total)
              ) %>%
                mutate(
                  across(
                    where(is.double),
                    function (x) {
```

```
                        round(x,
                              digits = 2 # .data[[input$digitos]] no me funcionó, así que no quedó reacti
                      )}
                    )
                  ) %>% datatable()
        })

output$tabla <- renderDT({tabla()})

}
```

3. En la pestaña Univariado hacé un gráfico de barras que sea reactivo a la selección de las variables sexo, fuma, dia y momento.

```
ui <- fluidPage(
        mainPanel(
          tabsetPanel(
            tabPanel("Univariado",
                     h2("Gráfico de Barras", align = "center"),
                     plotOutput("bar")
            )
          )
        )
      )

server <- function(input, output) {

  output$bar <- renderPlot({

                  ggplot(propinas,
                         aes(x=.data[[input$varcolor]])) +
                  geom_bar() +
                  labs(y = "Cantidad")
                })
          }
```

4. Generá dos botones de acciones (actionButton), el primero genera que se muestren los gráficos y tabla, el segundo que se borren los gráficos y tabla.

```
ui <- fluidPage(
        titlePanel("Datos de Propina"),
        sidebarLayout(
          sidebarPanel(
            actionButton("runif", "Mostrar"),
            actionButton("reset", "Ocultar")
          )
        )
      )


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

```
react <- reactiveValues(data = NULL)

observeEvent(input$runif, {

  react$scat <- ggplot(data = propinas,
                       aes(x = total,
                           y = propina,
                           colour = .data[[input$varcolor]])) +
                geom_point() +
                theme(aspect.ratio = 1) +
                scale_x_continuous(name ="Total de la cuenta") +
                scale_y_continuous(name = "Propina")


  react$tabla <- reactive({

                 propinas %>%
                  group_by(.data[[input$varcolor]]) %>%
                  summarise(
                    mean_prop = mean(propina),
                    sd_propina = sd(propina),
                    mean_total = mean(total),
                    sd_total = sd(total)
                  ) %>%
                  mutate(
                    across(
                      where(is.double),
                      function (x) {round(x,
                                          digits = 2 # .data[[input$digitos]]
                      )}
                    )) %>% datatable()
              })

  react$bar <- ggplot(data = propinas,
                      aes(x = .data[[input$varcolor]])) +
               geom_bar() +
               labs(y = "Cantidad")
  })


  observeEvent(input$reset, {
    react$scat <- NULL
    react$tabla <- NULL
    react$bar <- NULL
  })

  output$scat <- renderPlot({

    if (is.null(react$scat))
      return()
    plot(react$scat)

  })
```

```
    output$tabla <- renderDT({

      if (is.null(react$tabla))
        return()
      print(react$tabla())

    })


    output$bar <- renderPlot({

      if (is.null(react$bar))
        return()
      plot(react$bar)

    })
}
```

5. Subí tu aplicación a shinyiapps.io y compartí el link.

https://angelavieyto.shinyapps.io/tarea_3/