STAT 408 - R Shiny

March 6, 2018

Overview of R Shiny

R Shiny provides a way to create interactive visualizations and web applets

There are two key components of an R Shiny Script:

1. The ui (user interface) provides a way for the user to interactive with the visualization and for the program to capture input values.
2. The server piece takes those outputs and applies them to R code.

Server Code

Under the hood: Server Code

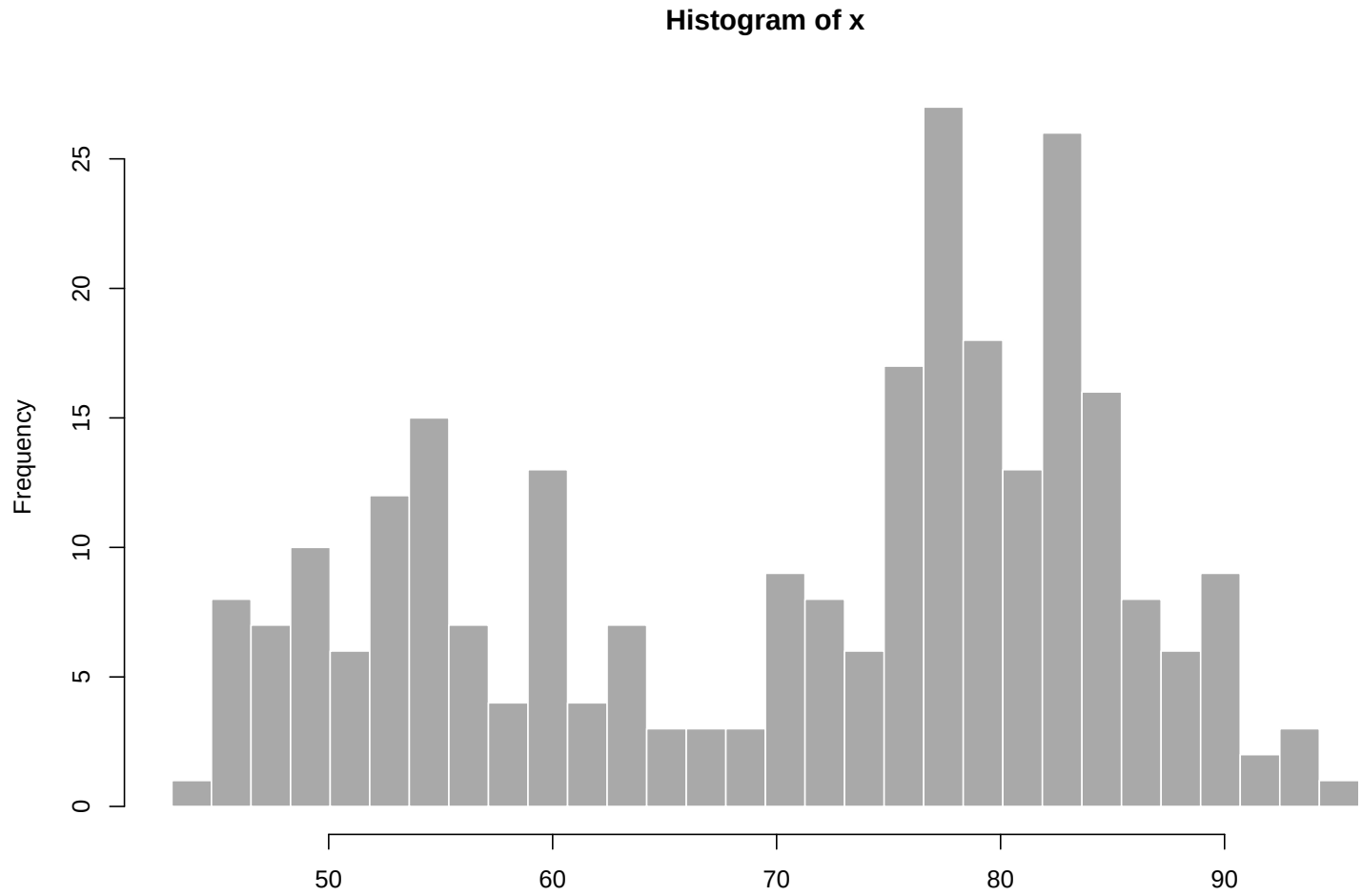

```
# Define server logic required to draw a histogram
server <- function(input, output) {

  output$distPlot <- renderPlot({
    # generate bins based on input$bins from ui.R
    x    <- faithful[, 2]
    bins <- seq(min(x), max(x), length.out = input$bins + 1)

    # draw the histogram with the specified number of bins
    hist(x, breaks = bins, col = 'darkgray', border = 'white')
  })
}
```

Under the hood: Server Code


```
    input.bins <- 30 # number of bins
    x    <- faithful[, 2]
    bins <- seq(min(x), max(x), length.out = input.bins + 1)
      # draw the histogram with the specified number of bins
    hist(x, breaks = bins, col = 'darkgray', border = 'white')
```

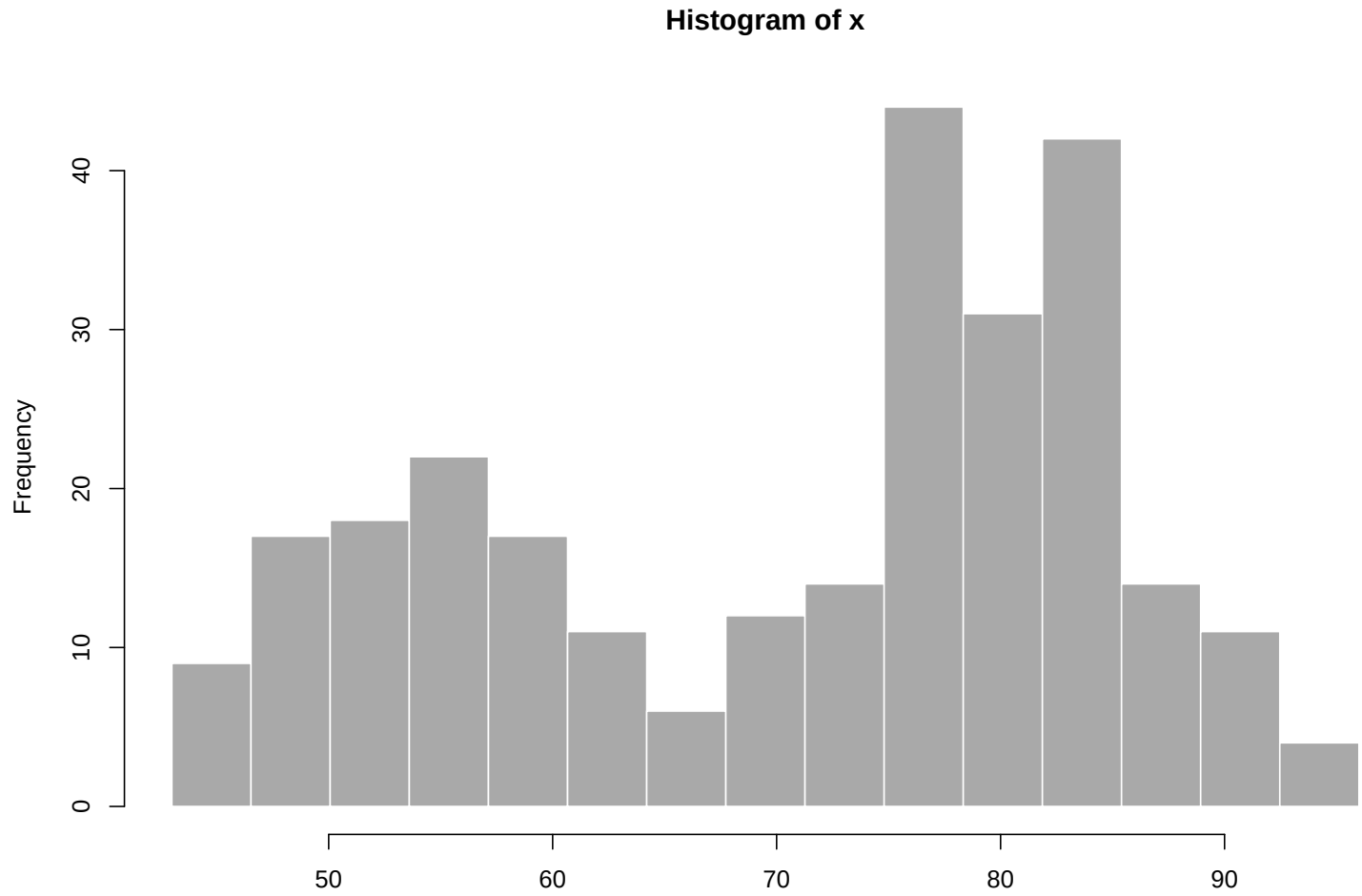
Under the hood: Server Code


```
input.bins <- 30 # number of bins
x    <- faithful[, 2]
bins <- seq(min(x), max(x), length.out = input.bins + 1)
  # draw the histogram with the specified number of bins
hist(x, breaks = bins, col = 'darkgray', border = 'white')
```



Under the hood: Server Code


```
input.bins <- 15 # number of bins
x    <- faithful[, 2]
bins <- seq(min(x), max(x), length.out = input.bins + 1)
  # draw the histogram with the specified number of bins
hist(x, breaks = bins, col = 'darkgray', border = 'white')
```



UI Code

Under the Hood: UI Code


```
# Define UI for application that draws a histogram
ui <- fluidPage(
   # Application title
   titlePanel("Old Faithful Geyser Data"),
   # Sidebar with a slider input for number of bins
   sidebarLayout(
      sidebarPanel(
         sliderInput("bins",
                     "Number of bins:",
                     min = 1,
                     max = 50,
                     value = 30)
      ),
      # Show a plot of the generated distribution
      mainPanel(
         plotOutput("distPlot")
      )
   )
)
```

Shiny Code: Running the Application


```
# Define server logic required to draw a histogram
server <- function(input, output) {

   output$distPlot <- renderPlot({
      # generate bins based on input$bins from ui.R
      x    <- faithful[, 2]
      bins <- seq(min(x), max(x), length.out = input$bins + 1)

      # draw the histogram with the specified number of bins
      hist(x, breaks = bins, col = 'darkgray', border = 'white')
   })
}

# Run the application
shinyApp(ui = ui, server = server)
```

Shiny Output

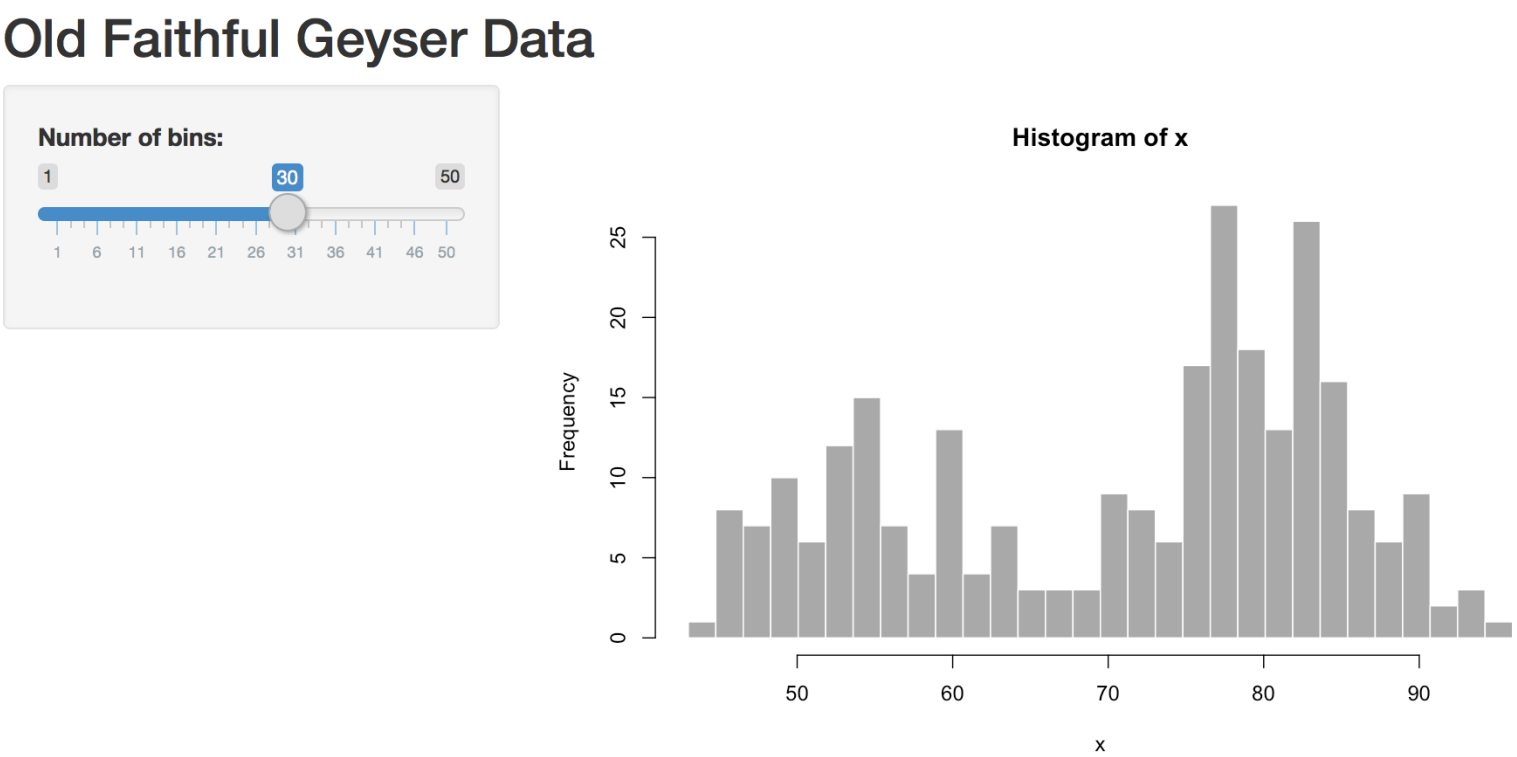


Figure 1: Snapshot of Shiny applet

Shiny Lab