ReadMe_myAPP

Ioannis Petridis

December, 2018

Dataset

The USArrests dataset was used for this application. The dataset shows violent crime rates for each state in United States

```
# Head of dataset
head(USArrests,5)
```

```
##            Murder Assault UrbanPop Rape
## Alabama      13.2     236       58 21.2
## Alaska       10.0     263       48 44.5
## Arizona       8.1     294       80 31.0
## Arkansas      8.8     190       50 19.5
## California    9.0     276       91 40.6
```

UI Code

```
#
# This is the user-interface definition of a Shiny web application. You can
# run the application by clicking 'Run App' above.
#
# Find out more about building applications with Shiny here:
#
#    http://shiny.rstudio.com/
#

library(shiny)
library(leaflet)
library(plotly)
data("USArrests")

# Define UI for application that draws a histogram
shinyUI(fluidPage(


  # Application title
  titlePanel("Crime Rates in US"),

  # Sidebar with a slider input for number of bins
  sidebarLayout(
    sidebarPanel(
      radioButtons("typeCrime", "Arrest Type:",
                   c("Murder" = "Murder",
                     "Assault" = "Assault",
                     "UrbanPop" = "UrbanPop",
                     "Rape" = "Rape"),selected = colnames(USArrests)[1]),
```

```
      submitButton("Submit")
    ),

    # Show a plot of the generated distribution
    mainPanel(
      h5("Graph of Crimes per State"),
      plotOutput("plot")
    )
  )
))
```

Server Code

```
#
# This is the server logic of a Shiny web application. You can run the
# application by clicking 'Run App' above.
#
# Find out more about building applications with Shiny here:
#
#    http://shiny.rstudio.com/
#

library(shiny)

# Define server logic required to draw a histogram
shinyServer(function(input, output) {
  output$plot<-renderPlot({
    crimes <- data.frame(state=tolower(rownames(USArrests)), USArrests)
    gg <- ggplot(crimes, aes(map_id=state, fill=USArrests[,input$typeCrime]))
    gg <- gg + guides(fill=guide_legend(title=as.character(input$typeCrime)))
    gg <- gg + geom_map(map=map_data("state"))
    gg <- gg + expand_limits(x=map_data("state")$long, y=map_data("state")$lat)
    gg
  })

})
```