Data 608 Module 3

Md Jalal Uddin

March 9, 2019

```
library(shiny)
library(DT)
```

```
## Warning: package 'DT' was built under R version 3.5.2
```

```
##
## Attaching package: 'DT'
```

```
## The following objects are masked from 'package:shiny':
##
##     dataTableOutput, renderDataTable
```

```
library(plotly)
```

```
## Warning: package 'plotly' was built under R version 3.5.2
```

```
## Loading required package: ggplot2
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

```
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(googleVis)
```

```
## Warning: package 'googleVis' was built under R version 3.5.2
```

```
## Creating a generic function for 'toJSON' from package 'jsonlite' in package 'googleVis'
```

```
##
## Welcome to googleVis version 0.6.3
##
## Please read Google's Terms of Use
## before you start using the package:
## https://developers.google.com/terms/
##
## Note, the plot method of googleVis will by default use
## the standard browser to display its output.
##
## See the googleVis package vignettes for more details,
## or visit https://github.com/mages/googleVis.
##
## To suppress this message use:
## suppressPackageStartupMessages(library(googleVis))
```

```
library(rsconnect)
```

```
## Warning: package 'rsconnect' was built under R version 3.5.2
```

```
##
## Attaching package: 'rsconnect'
```

```
## The following object is masked from 'package:shiny':
##
##     serverInfo
```

Question 1 As a researcher, you frequently compare mortality rates from particular causes across different States. You need a visualization that will let you see (for 2010 only) the crude mortality rate, across all States, from one cause (for example, Neoplasms, which are effectively cancers). Create a visualization that allows you to rank States by crude mortality for each cause of death.

```
# Load data
df <- read.csv("https://raw.githubusercontent.com/charleyferrari/CUNY_DATA608/master/lecture3/data/clea

# Subset for 2010 only
df2010 <- subset(df, Year==2010)

# Unique causes of death for 2010
allcauses <- unique(df2010$ICD.Chapter)
```

```
# Create server logic
ui1 <- fluidPage(
  titlePanel("2010 CDC Mortality Rates"),
  fluidRow(selectInput("cause", "Cause of Death:", choices=sort(allcauses))),
  plotOutput("plot1", height = 800)
)
```

```
# Unique causes of death for 2010
allcauses <- as.data.frame(unique(df2010$ICD.Chapter))

# Create user-interface definition
server1 <- function(input, output) {
  showrates <- reactive({showrates <- subset(df2010, ICD.Chapter==input$cause)})
  output$plot1 <- renderPlot({
    ggplot(showrates(), aes(x=Crude.Rate, y=reorder(State, -Crude.Rate)))+
      scale_x_continuous(limits=c(0, max(showrates()$Crude.Rate))+5, expand = c(0,0))+
      geom_segment(aes(yend=State), xend=0, color="blue")+
      geom_point(size=3, color = "red") +
      theme_bw()+
      theme(panel.grid.major.y = element_blank(), axis.title=element_text(size=14))+
      xlab("2010 CDC Mortality Rate") +
      ylab("State") +
      ggtitle(input$cause)
  })
}
```

```
shinyApp(ui = ui1, server = server1)
```

Shiny applications not supported in static R Markdown documents

Question 2 Often you are asked whether particular States are improving their mortality rates (per cause) faster than, or slower than, the national average. Create a visualization that lets your clients see this for themselves for one cause of death at the time. Keep in mind that the national average should be weighted by the national population.

```
# Load data
df3 <- read.csv("https://raw.githubusercontent.com/charleyferrari/CUNY_DATA608/master/lecture3/data/clea

# Get unique lists of inputs
allcauses<-unique(df3$ICD.Chapter)
allstates<-unique(df3$State)

# Create UI script
ui2 <- fluidPage(
  title = "State Mortality Rates Over Time",
  fluidRow(
    column(6, selectInput('causes', 'Cause of Death', choices=sort(allcauses)) ),
    column(6, selectInput('states', 'State', choices=sort(allstates)) )
  ),
  fluidRow(
    plotOutput('myplot')
  )
)
```

```
# Create server script
server2 <- function(input, output) {
  output$myplot <- renderPlot({
    data <- df3 %>%
      filter(State==input$states, ICD.Chapter==input$causes)
    usavg <- df3 %>%
      filter(ICD.Chapter==input$causes) %>%
      group_by(Year) %>%
      summarise(rateyr=(sum(as.numeric(Deaths))/sum(as.numeric(Population))*100000))
    ggplot(data, aes(x=Year, y=Crude.Rate, color='purple')) +
      geom_line(size=3) +
      geom_line(aes(x=usavg$Year, y=usavg$rateyr, color='green'),size=2) +
      scale_color_manual(
        name='Legend',
        values=c('green'='green', 'purple'='purple'),
        labels=c('National Average', 'State'))
  })
}
```

```
shinyApp(ui = ui2, server = server2)
```

Shiny applications not supported in static R Markdown documents