

Yexuan Gu  
UP 494  
Lab Exercise 4  
11/8/16

## Creating Graphics and Dashboards with R

### Tools

- Csv files of daily jail log data from <http://dart.ncsa.uiuc.edu/stuffed/bpni/>
- R and its packages: shiny, shinydashboard, highcharter, dplyr, tidyr, rdrop2, lubridate, RCurl

### Procedures

```
1 install.packages("shiny")
2 install.packages("shinydashboard")
3 install.packages("highcharter")
4 install.packages("dplyr")
5 install.packages("tidyr")
6 install.packages("rdrop2")
7 install.packages("lubridate")
8 install.packages("RCurl")
9
10
11 library(shiny)
12 library(shinydashboard)
13 library(highcharter)
14 library(dplyr)
15 library(tidyr)
16 library(rdrop2)
17 library(lubridate)
18 library(RCurl)
```

- 1) Install and library packages
- 2) Set work directory to my work folder

```
20 setwd("C:\\Users\\ygu23\\Desktop\\Academic\\UP494\\LabExercise4")
```

- 3) Read and organize data

```
22 data <- read.csv('djlsummary.all_LE4.csv')
23 data$Date <- as.Date(data$custDate, format="%m/%d/%Y")
24 y.limit <- max(data$popTot)
25 output.table <- data
26 chg.Tot.Pop <- diff(data$popTot)
27 daye <- lubridate::wday(data$custDate, label=T, abbr = F)
28
29 k <- list()
30 for (i in 1:(length(daye) - 1)) {
31   k[i] <- (paste0(daye[i], " to ", daye[i+1], ""))
32 }
33 chg.Tot.Pop.Labels <- as.character(k)
34 change.data <- cbind.data.frame(chg.Tot.Pop.Labels, chg.Tot.Pop)
35 colnames(change.data) <- c("Period", "Change in Total Population")
36
37 lval <- dim(data)[1]
38 lval <- lval * -1
39 data.sorted <- data[order(-1:lval),]
40
```

- 4) Set up user interface
  - a) Dashboard page includes dashboard header, sidebar, sidebar menu. The first step is to set up

```
1 ui <- dashboardPage(
2   dashboardHeader(title = "Champaign County Jail Population Summary"),
3   dashboardSidebar(
4     sidebarMenu(
5       menuItem("Dashboard", tabName = "Dashboard"),
6       menuItem("Raw Data", tabName = "Rawdata")
7     )
8   ),
9   mainContentArea()
10 )
```

these elements

- b) Still in the “dashboardPage” command, we need to set up the dashboard body based on the csv file that we collected. There are two items within the dashboard body: “dashboard” and “raw data”. Within the first tab item, dashboard, there is the majority of our data visualization. We use fluidRow to visualize the data into boxes. The data includes total jail population today, total inmates sentenced, total inmates end of sentence, total jail population distributed by location, satellite and downtown, change in jail population, total bookings yesterday, total inmates transferred, total inmates released yesterday, total inmates with appointments and total inmates with appointments / bond.

```

9 dashboardBody(
10   tabItems(
11     tabItem("Dashboard",
12       fluidRow(
13         valueBoxOutput("total"),
14         valueBoxOutput("sentenced"),
15         valueBoxOutput("unsentenced"),
16         valueBoxOutput("end-of-sentence")
17       ),
18       fluidRow(
19         box(
20           width = 8, status = "info", solidHeader = TRUE,
21           title = "Distribution by Location",
22           # plotOutput("plot1")
23           highchartOutput("plot1")
24         ),
25         box(
26           width = 4, status = "info",
27           title = "Change in Jail Population",
28           tableOutput("table1")
29         )
30       ),
31       fluidRow(
32         valueBoxOutput("bookings"),
33         valueBoxOutput("transfers"),
34         valueBoxOutput("releases"),
35         valueBoxOutput("appts"),
36         valueBoxOutput("apBond")
37       )
38     )
39   )

```

- c) In Raw Data, we also enable users to download all data as csv.

```

39   tabItem("Rawdata",
40     numericInput("maxrows", "Rows to show", 25, min=1, max=dim(data)[1]),
41     verbatimTextOutput("rawtable"),
42     downloadButton("downloadCsv", "Download All Data as CSV")
43   )

```

- 5) After setting up the user interface, we need to script the server.

- a) Using Highchart to visualize the total jail population distributed in downtown and satellite.

```

1 server <- function(input, output) {
2   output$plot1 <- renderHighchart({
3     downtown <- subset(data, select=c(popDtn, Date))
4     satellite <- subset(data, select=c(popsat, Date))
5     highchart() %>%
6       hc_chart(type="area") %>%
7       hc_xAxis(categories = unique(as.Date(downtown$Date, "%m/%d/%Y")),
8         tickmarkPlacement = 'on') %>%
9       hc_yAxis(title = list(text = "Persons")) %>%
10      hc_add_series(data=satellite$popsat, name = "Satellite Facility", color="orange") %>%
11      hc_add_series(data=downtown$popDtn, name = "Downtown Facility", color = "dodgerblue") %>%
12      hc_add_theme(hc_theme_gridlight())
13   })
14 }

```

- b) Render the chart we just made with specific attributes

```

15 output$total <- renderValueBox({
16   valueBox(
17     value = data.sorted$popTot[1],
18     subtitle = "Total Jail Population Today",
19     icon = icon("area-chart"),
20     color = "yellow"
21   )
22 })

```

- c) Set the box for other information, such as total jail population, total inmates sentenced, etc. Give different colors to boxes with different information.

```

15- output$total <- renderValueBox({
16-   valueBox(
17-     value = data.sorted$popTot[1],
18-     subtitle = "Total Jail Population Today",
19-     icon = icon("area-chart"),
20-     color = "yellow"
21-   )
22- })

32- output$unsentenced <- renderValueBox({
33-   valueBox(
34-     value = data.sorted$custUnsent[1],
35-     subtitle = "Total Inmates Unsented",
36-     icon = icon("clock-o"),
37-     color = "aqua"
38-   )
39- })

51- output$bookings <- renderValueBox({
52-   valueBox(
53-     value = data.sorted$bookings[1],
54-     subtitle = "Total Bookings Yesterday",
55-     icon = icon("book"),
56-     color = "yellow"
57-   )
58- })

68- output$releases <- renderValueBox({
69-   valueBox(
70-     value = data.sorted$releases[1],
71-     subtitle = "Total Inmates Released Yesterday",
72-     icon = icon("sign-out"),
73-     color = "aqua"
74-   )
75- })

86- output$apBond <- renderValueBox({
87-   valueBox(
88-     value = data.sorted$apBond[1],
89-     subtitle = "Total Inmates With Appointments/Bond",
90-     icon = icon("sign-out"),
91-     color = "olive"
92-   )
93- })

23- output$sentenced <- renderValueBox({
24-   valueBox(
25-     value = data.sorted$custSent[1],
26-     subtitle = "Total Inmates Sentenced",
27-     icon = icon("balance-scale"),
28-     color = "light-blue"
29-   )
30- })

41- output$unsentenced <- renderValueBox({
42-   valueBox(
43-     value = data.sorted$custEndSent[1],
44-     subtitle = "Total Inmates End of Sentence",
45-     icon = icon("clock-o"),
46-     color = "blue"
47-   )
48- })

59- output$transfers <- renderValueBox({
60-   valueBox(
61-     value = data.sorted$transfers[1],
62-     subtitle = "Total Inmates Transferred",
63-     icon = icon("exchange"),
64-     color = "light-blue"
65-   )
66- })

77- output$appts <- renderValueBox({
78-   valueBox(
79-     value = data.sorted$appts[1],
80-     subtitle = "Total Inmates With Appointments",
81-     icon = icon("sign-out"),
82-     color = "teal"
83-   )
84- })

```

d) Create the table for change in jail population

```

94-
95- output$table1 <- renderTable({
96-   lval <- dim(change.data)[1]
97-   lval <- lval * -1
98-   change.data.display <- change.data[order(-1:lval),]
99-   change.data.display <- head(change.data.display, 11)
100- }, digits = 1, include.rownames=FALSE)
101-
102- output$downloadCsv <- downloadHandler(
103-   filename = "djlsummary.all_LE4.csv",
104-   content = function(file) {
105-     write.csv(data, file)
106-   },
107-   contentType = "text/csv"
108- )
109-

```

e) Output the raw data into a table in the raw data tab.

```

110- output$rawtable <- renderPrint({
111-   orig <- options(width = 1000)
112-   print((tail(data, input$maxrows)), row.names=FALSE)
113-   options(orig)
114- })
115- }
116-

```

## Results and Analysis

Link on shinyapp.io: <http://127.0.0.1:5759>

Champaign County

Dashboard

Row Data

184

Total Jail Population Today

8

Total Inmates Sentenced

1

Total Inmates End of Sentence

Distribution by Location

Change in Jail Population

Period	Change in Total Population
NA to NA	-2
NA to NA	0
NA to NA	-11
NA to NA	7
NA to NA	3
NA to NA	6
NA to NA	-7
NA to NA	3
NA to NA	-1
NA to NA	-13
NA to NA	-4

10

Total Bookings Yesterday

0

Total Inmates Transferred

12

Total Inmates Released Yesterday

5

Total Inmates With Appointments

4

Total Inmates With Appointments/Bond

[illegible]

Figure 2. Raw data page for daily jail population in Champaign County

## **Discussion**

In my opinion, the shiny dashboard package is tremendously useful and applicable for urban planning. Firstly, government data are always in pdf or excel files. Even though government encourages and makes efforts to make the data transparent to residents, the complicated forms of data still impede the users' reading and understanding. The shiny dashboard increases the readability of the data from diverse professional institutions not only for residents, but also for planners, designers, engineers, etc. Secondly, shiny dashboard not only simply visualize the data in its original formats, but it also helps to organize the data into specific form for different purposes of use. For example, the figure made by Highchart visualizes the data into a comparison between different items. Thirdly, there will be new data coming in the system fast every day. The data can be updated fast by these tools to meet changes and uncertainties for residents' uses and urban planning.