

Introduction to Shiny

Augustin Luna
21 January, 2016

Research Fellow
Department of Biostatistics and Computational Biology
Dana-Farber Cancer Institute

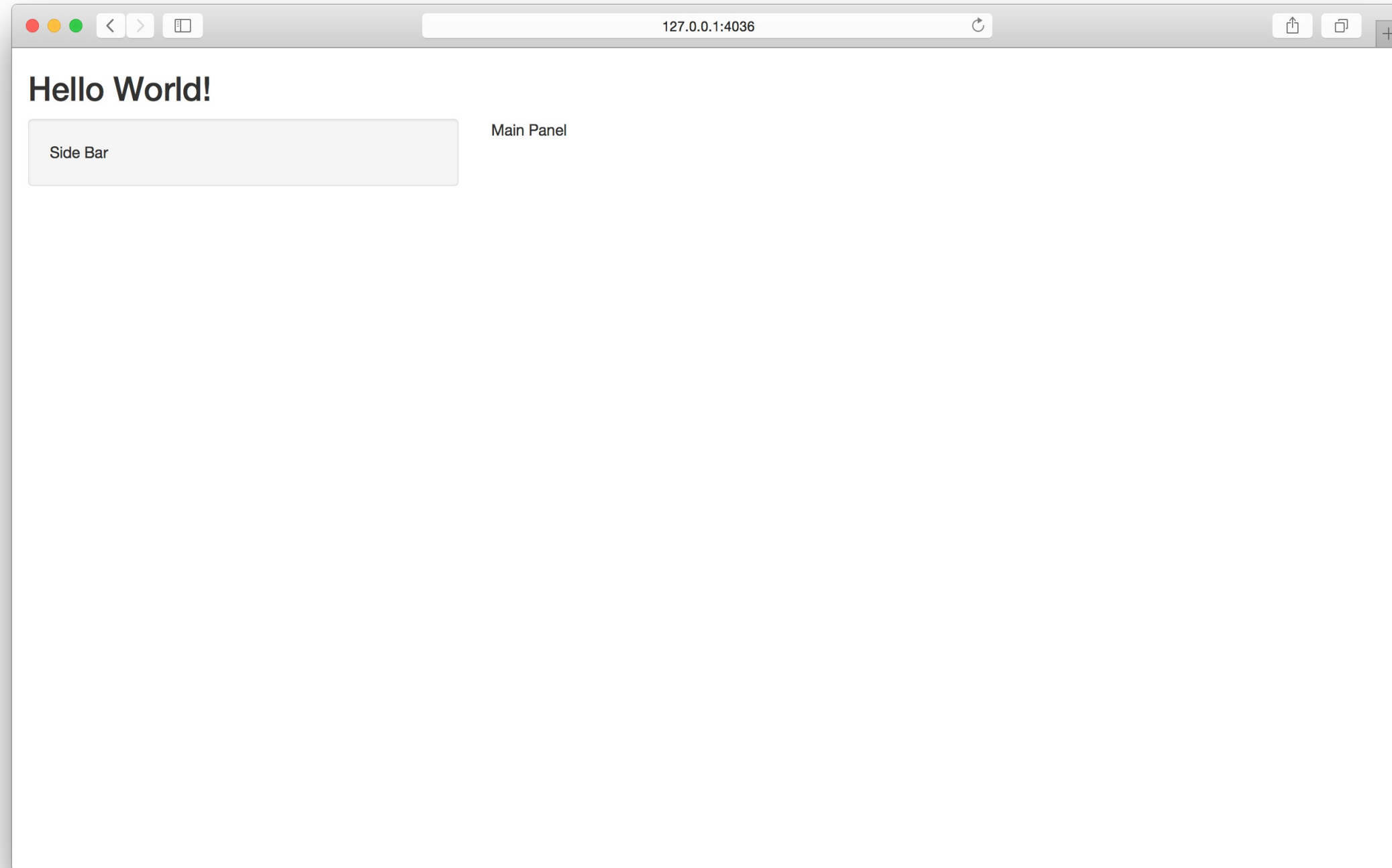
What is Shiny?

- Provides an easy and quick way of presenting data and engaging with users
 - Share your analyses with users that do not know R
- An R-based web application framework for interactive R scripts and visualizations
- Developed by RStudio
- Web applications can be run locally, on a web server, or hosted by RStudio at <http://shinyapps.io>
- Can handle complex situations
 - File uploads for data processing
 - Interactive plots

Shiny App Structure

- Two components
 - User interface script: controls layout and appearance
 - Server script: contains code to run any analyses visualized in the app
- Conversion of R code to an HTML (website) is handled by the `shiny` R package
 - Knowledge of web development is not necessary, but custom Javascript, HTML and CSS can be included

Hello World! Shiny App Screenshot



Hello World! Shiny App Code

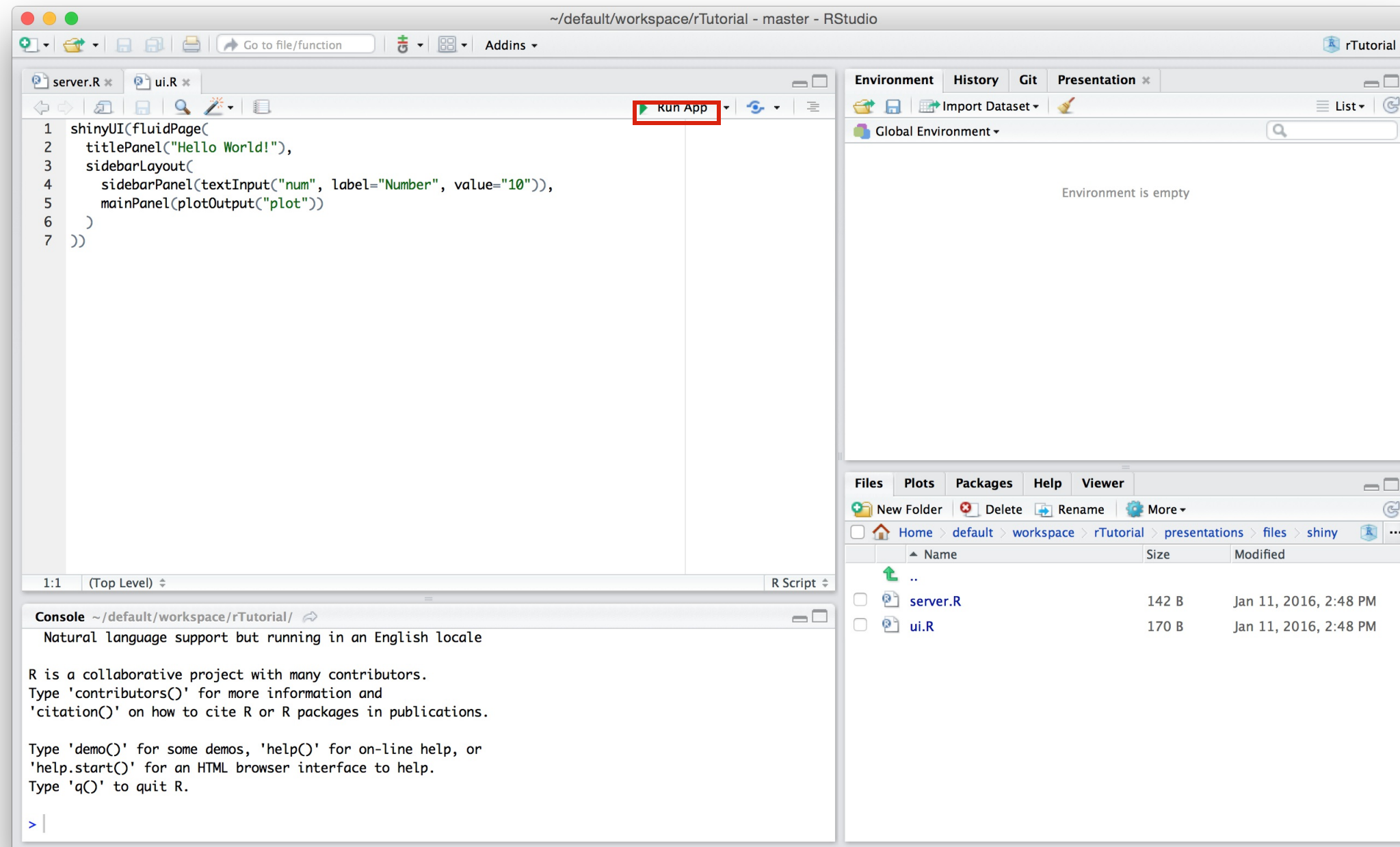
- `fluidPage`: Expand components to fill page
- `titlePanel`: Adds page title
- `sidebarLayout`: Sets page layout
- `sidebarPanel`: Sidebar contents
- `mainPanel`: Main page contents

```
# ui.R
shinyUI(fluidPage(
  titlePanel("Hello
World!"),
  sidebarLayout(
    sidebarPanel("Side
Bar"),
    mainPanel("Main Panel")
  )
))

# server.R
shinyServer(function(input,
output) {
  # NOTHING
})
```

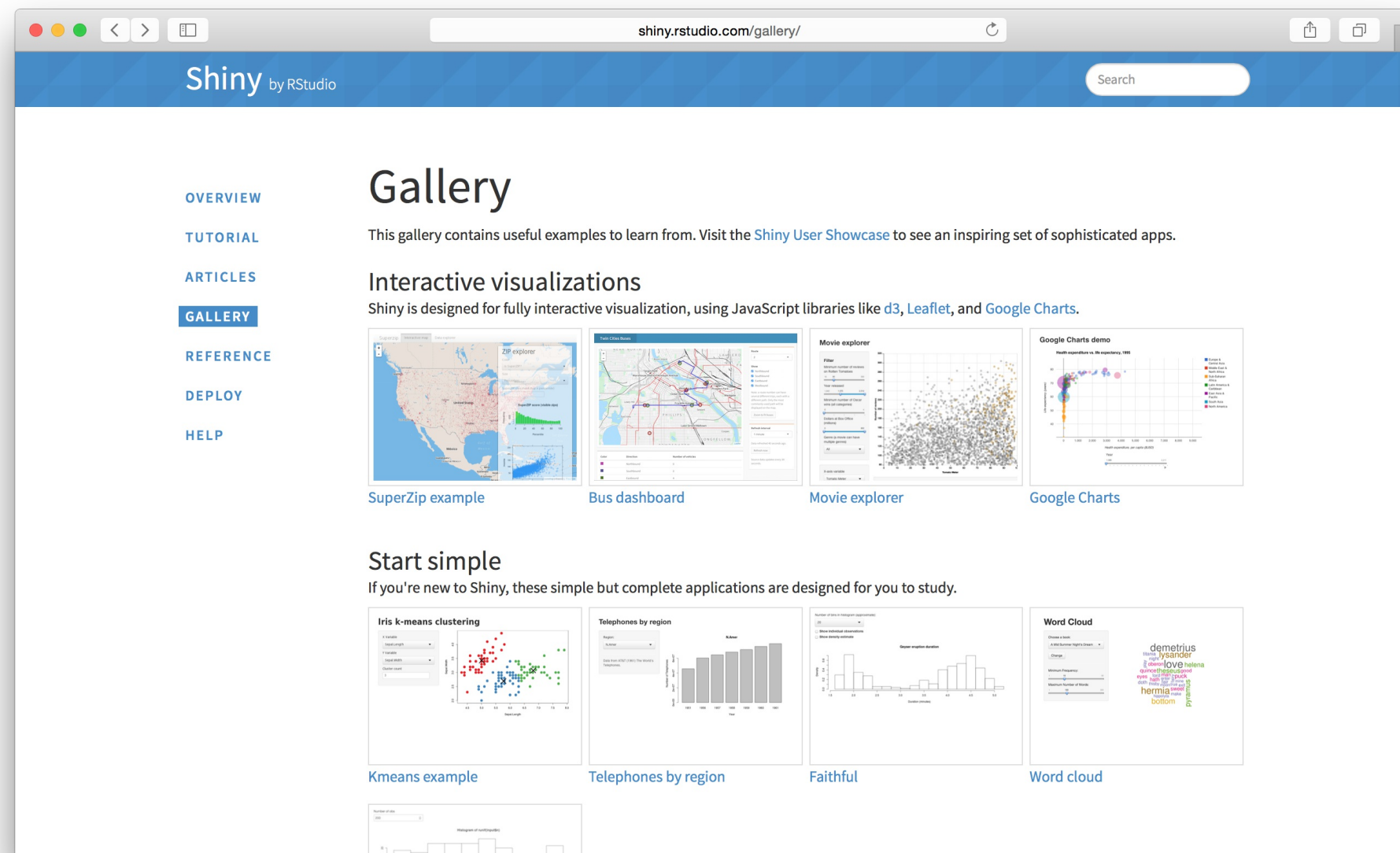
Run Shiny App in RStudio

- You can also call `runApp()` in the folder with app files



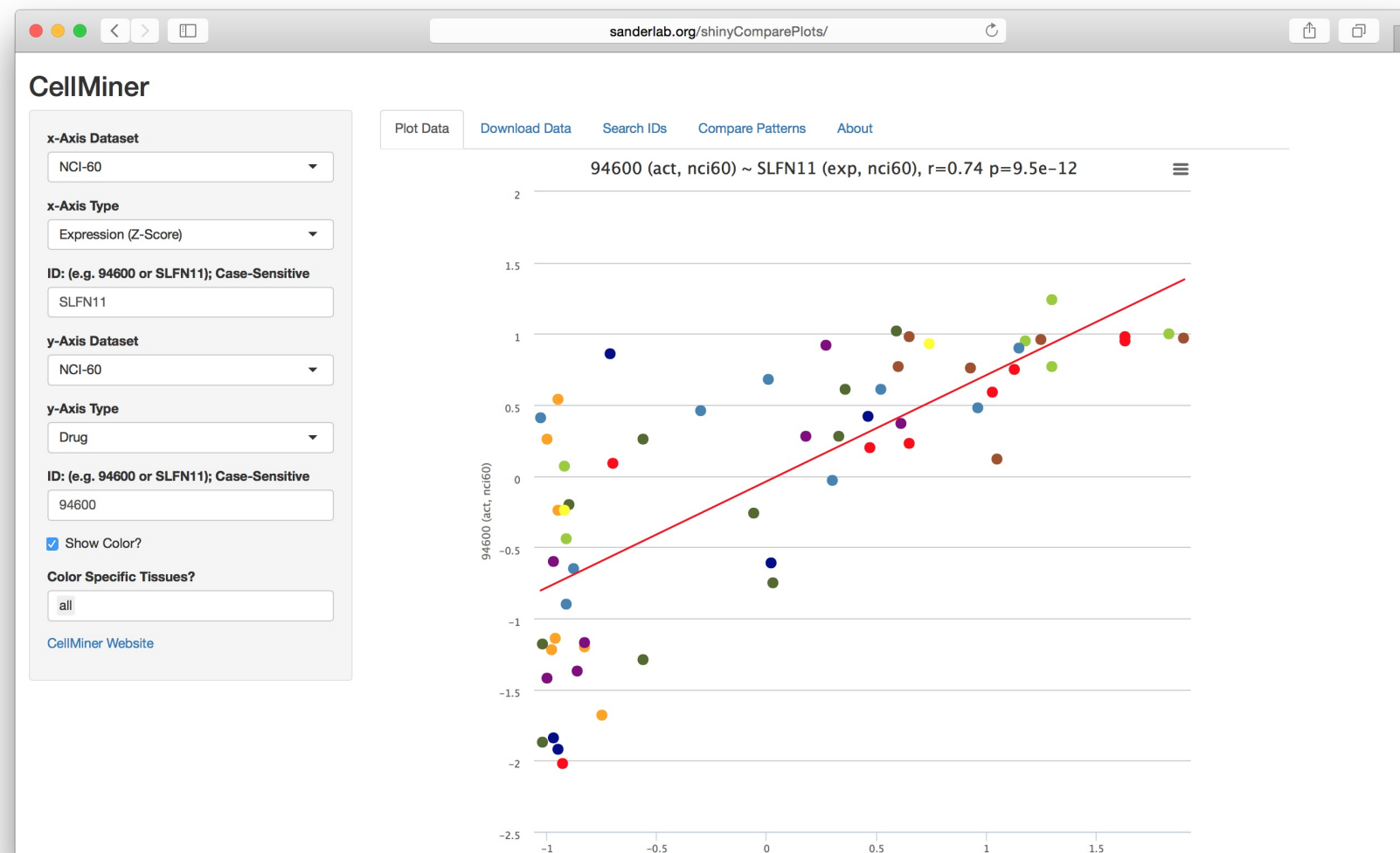
Shiny App Gallery

- RStudio provides a large gallery of Shiny apps with code
 - <http://shiny.rstudio.com/gallery/>



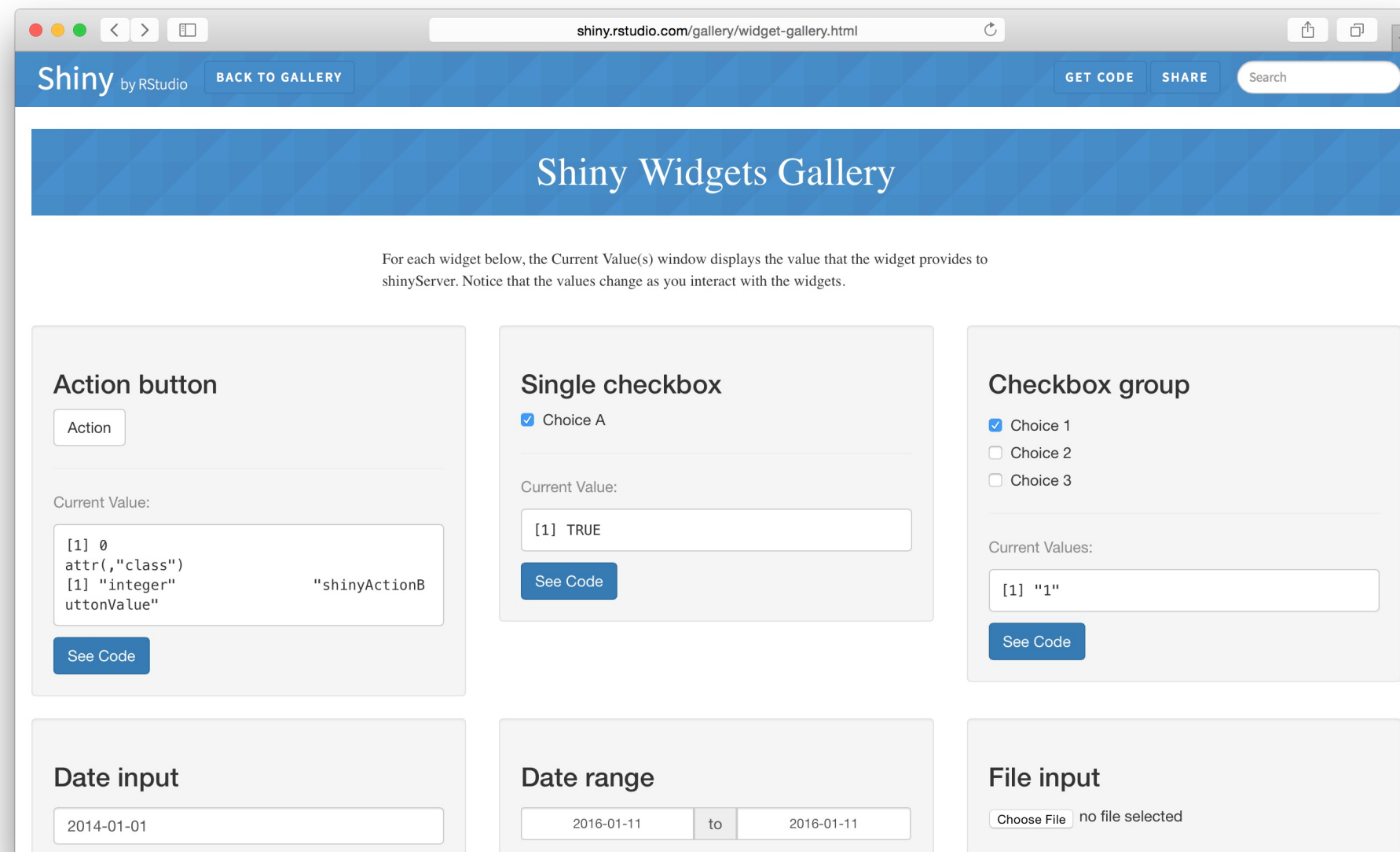
rcellminer Shiny Apps

- Complex Shiny apps are provided with rcellminer
- Features with interactive plots, progress bars, tabs, etc.
 - <http://sanderlab.org/shinyComparePlots/>



Shiny App Inputs (Widgets)

- Most inputs found on websites are available in Shiny
 - <http://shiny.rstudio.com/gallery/widget-gallery.html>



Shiny Widgets to be Covered

- Inputs
 - `textInput`: Provide a text field
- Outputs
 - `verbatimTextOutput`: Print R messages “as is”
 - `plotOutput`: Display an R plot image
- `includeMarkdown`: Displays Markdown-formatted text files

Displaying Formatted Text using Markdown

- Text formatting organizes content with headers, bullets, etc.
- Markdown is a text-to-HTML conversion tool that allows you to write using an easy-to-read, easy-to-write plain text format
 - <https://www.rstudio.com/resources/cheatsheets/>

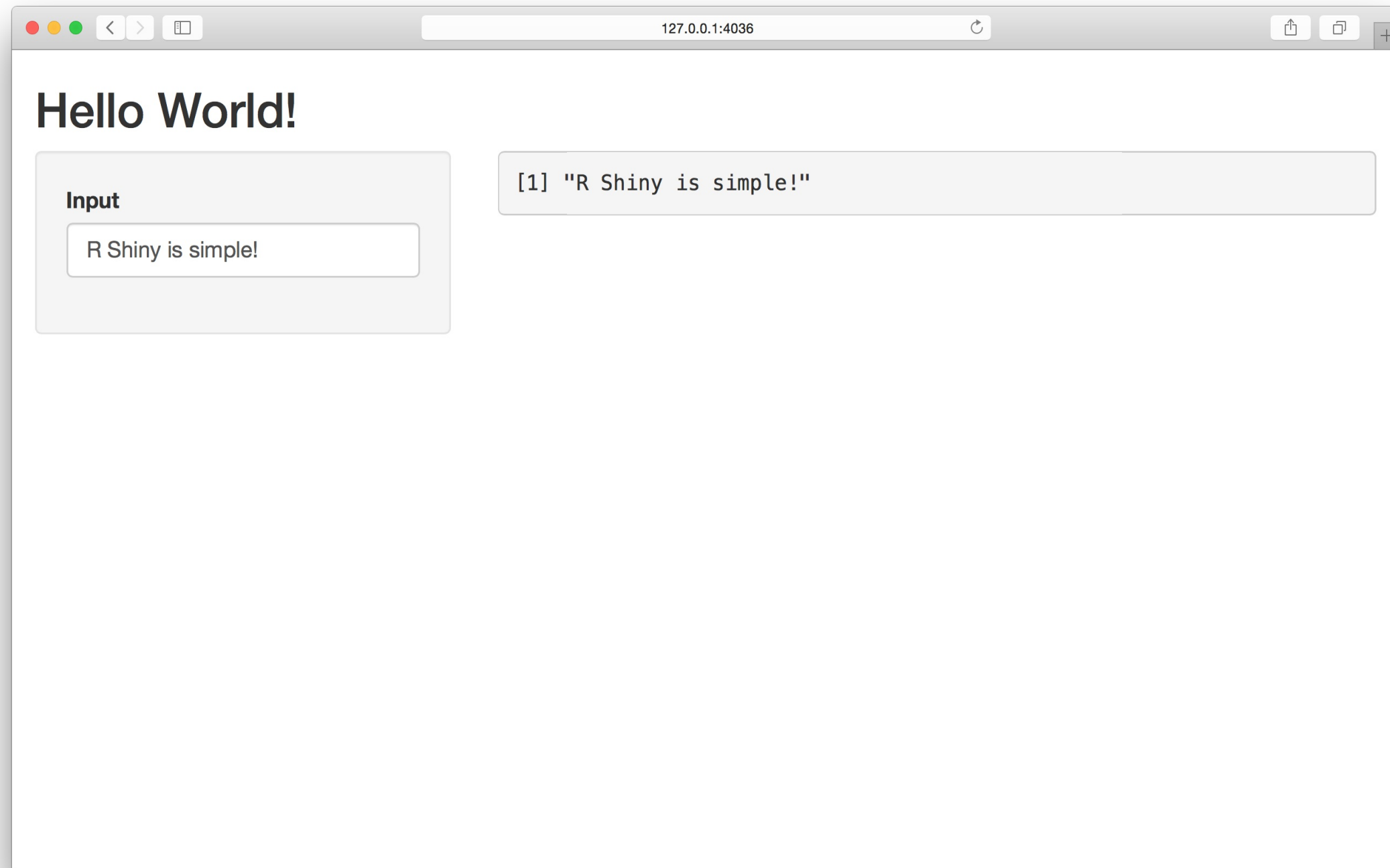
Shiny Text Input

- `textInput` example that changes text as the user types; a reactive input
- `textInput("text", ...)` label matches `input$text`
- `verbatimTextOutput("value")` label matches `output$value`

```
# ui.R
shinyUI(fluidPage(
  titlePanel("Hello World!"),
  sidebarLayout(
    sidebarPanel(textInput("text", label="Input",
value="Type here...")),
    mainPanel(verbatimTextOutput("value"))
  )
))

# server.R
shinyServer(function(input, output) {
  output$value <- renderPrint({ input$text })
})
```

Text Input Shiny App Example



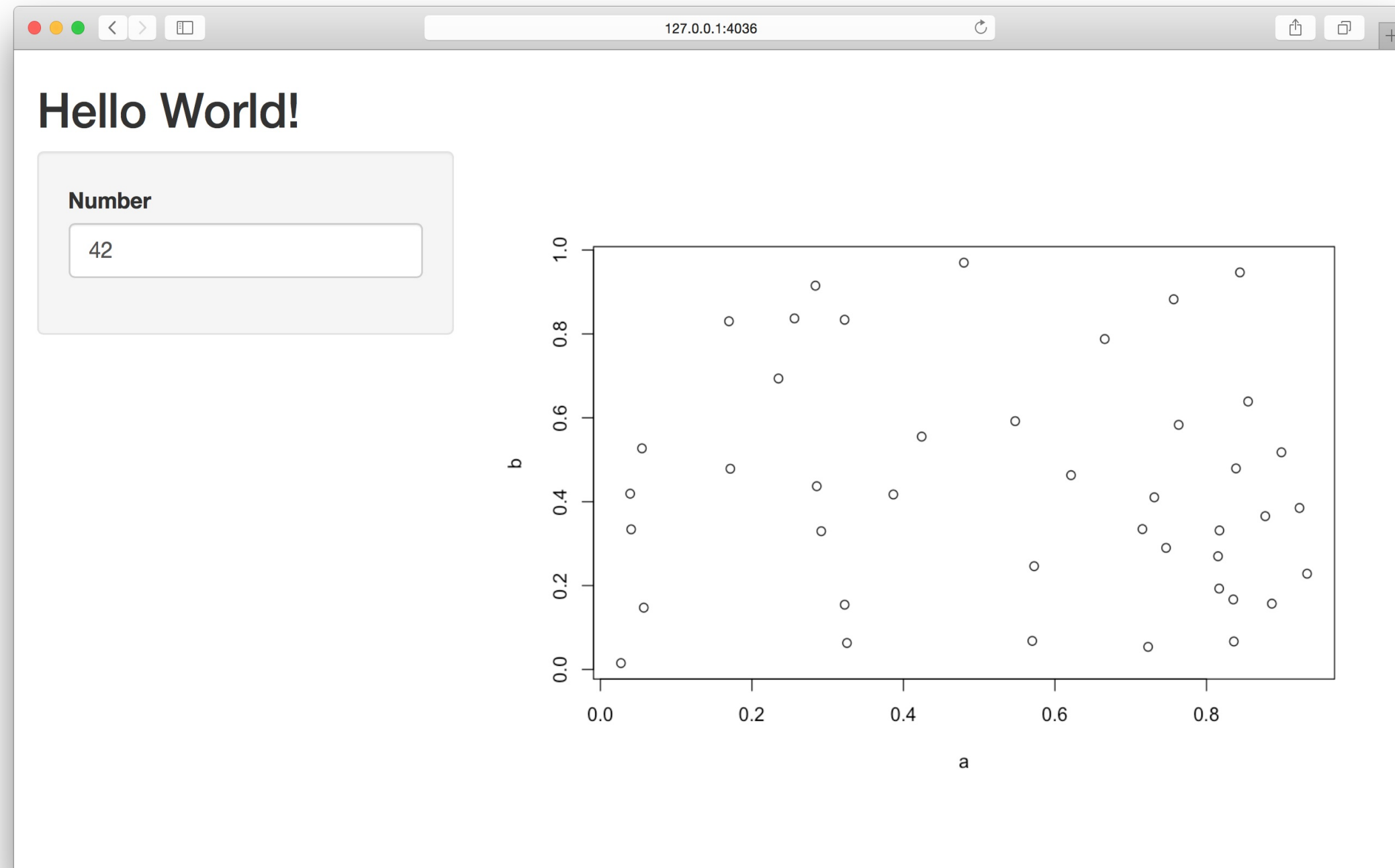
Shiny Plot Output

- Any R code that produces a plot can be included in `renderPlot`

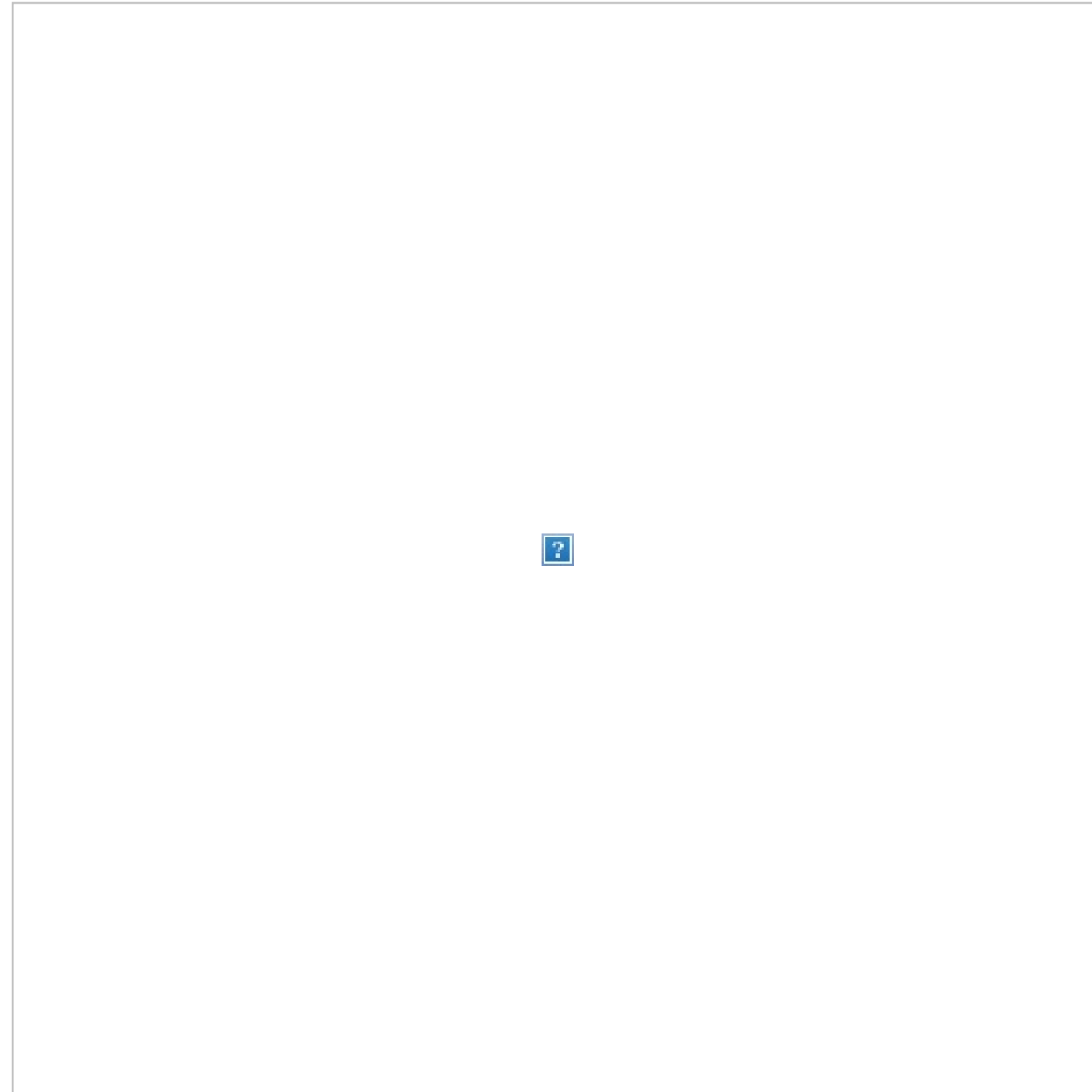
```
# ui.R
shinyUI(fluidPage(
  titlePanel("Hello World!"),
  sidebarLayout(
    sidebarPanel(textInput("num", label="Number",
value="10")),
    mainPanel(plotOutput("plot"))
  )
))

# server.R
shinyServer(function(input, output) {
  output$plot <- renderPlot({
    a <- runif(input$num)
    b <- runif(input$num)
    plot(a, b)
  })
})
```

Plot Output Shiny App Example



CellMiner Heatmap Shiny App



Install Heatmap Shiny Component

```
if (!require("devtools")) {  
  install.packages("devtools")  
}  
  
devtools::install_github("rstudio/d3heatmap")
```

CellMiner Heatmap: ui.R

```
library(shiny)
library(d3heatmap)

shinyUI(fluidPage(
  titlePanel("CellMiner Heatmap"),
  sidebarLayout(
    sidebarPanel(
      textInput("geneList", "Gene List:", "TP53 BRAF
PTEN")
    ),
    mainPanel(
      d3heatmapOutput("heatmap")
    )
  )
))
```

CellMiner Heatmap: server.R

```
# server.R
library(shiny)
library(rcellminer)
library(d3heatmap)

shinyServer(
  function(input, output){
    output$heatmap <- renderD3heatmap({
      genes <- unlist(strsplit(input$geneList, " "))
      expData <-
        getAllFeatureData(rcellminerData::molData)[["exp"]]
      d3heatmap(expData[genes, 1:20], scale="column",
        colors="YlOrRd")
    })
  }
)
```

CellMiner Formulas Shiny App

- Drug activity can result from the interaction of many genes
- Source code
 - https://bitbucket.org/cbio_mskcc/rcellminer/src/
 - Located under rcellminer/inst/shinyTutorial/
 - Files: www/help.md, server.R, ui.R
- Notes
 - Code in `reactive()` runs when an input changes

Code Location in server.R

- Location of R code in server.R affects how frequently it is run

```
# server.R

# Run once when the app is loaded

shinyServer(function(input, output) {

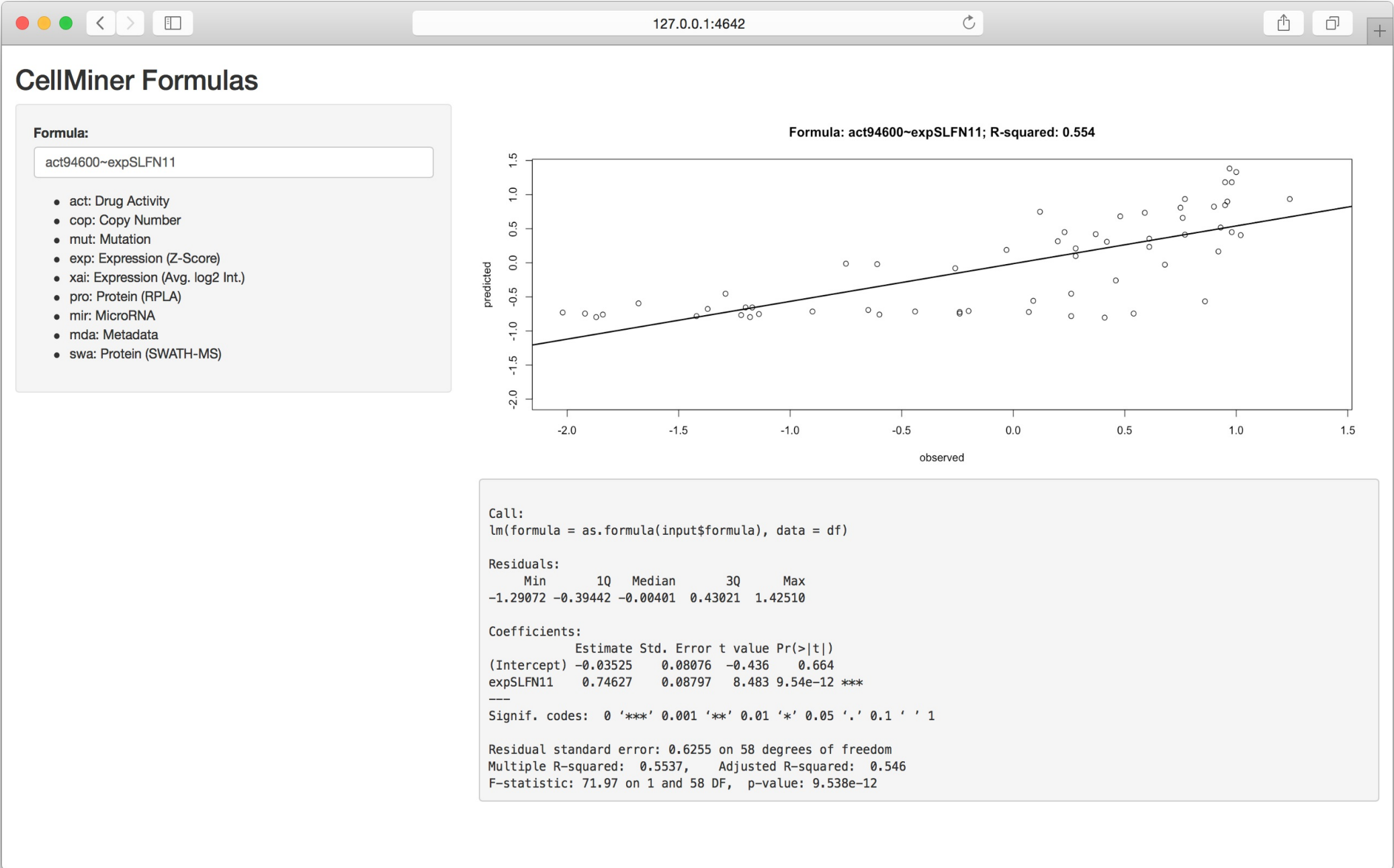
  # Run once when a new user visits

  output$plot <- renderPlot({

    # Run every time a user makes input
    changes

  })
})
```

CellMiner Formulas Shiny App

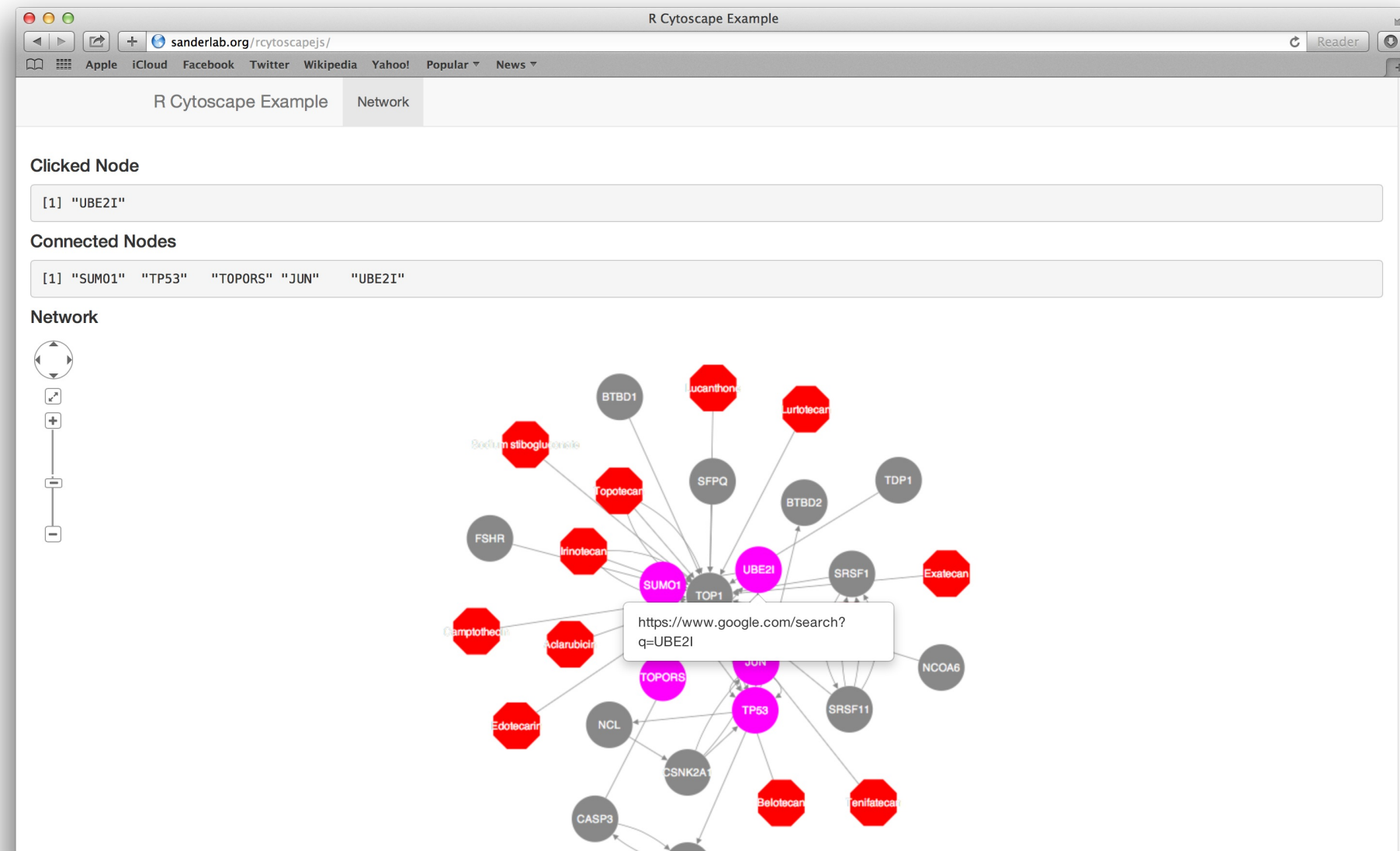


Package Recommendations for Advanced Functionality

- `rcharts`: display of interactive plots using existing Javascript libraries
 - Highcharts is included in `rcharts` and is highly customizable
 - <http://ramnathv.github.io/rCharts/>
- `DT`: interactive tables using the DataTables library
 - <https://rstudio.github.io/DT/>
- `htmlwidgets`: allows any Javascript library to be used into Shiny apps
- `rcytoscape.js`: built with `htmlwidgets` is for network views
 - <https://github.com/cytoscape/r-cytoscape.js>

r-cytoscape.js

- Provides interactive network views that communicate user actions back to R



Hosting Apps on shinyapps.io

- shinyapps.io by RStudio
- Easy to update Shiny apps from RStudio
- Free tier can be used quickly with moderate use
- Authentication is not free

FREE	STARTER	BASIC	STANDARD	PROFESSIONAL
\$0 /month	\$9 /month (or \$100/year)	\$39 /month (or \$440/year)	\$99 /month (or \$1,100/year)	\$299 /month (or \$3,300/year)
New to Shiny? Deploy your applications for FREE.	More applications. More active hours!	Take your users' experience to the next level!	Need password protection? Authenticate your users!	Professional has it all! Personalize your domains.
5 Applications	25 Applications	Unlimited Applications	Unlimited Applications	Unlimited Applications
25 <u>Active Hours</u>	100 <u>Active Hours</u>	500 <u>Active Hours</u>	2,000 <u>Active Hours</u>	10,000 <u>Active Hours</u>
✔ Community Support	✔ Email Support	✔ <u>Performance Boost</u>	✔ Authentication	✔ Authentication

Hosting Apps on using Shiny Server

- Shiny Server allows self-hosting apps, simple to set up
 - <https://www.rstudio.com/products/shiny/shiny-server/>
- Authentication is not free
 - Alternative, set authentication at the web server level
 - <https://wiki.apache.org/httpd/PasswordBasicAuth>

Embedding Shiny Apps in R Packages

- Shiny apps can be embedded as part of R packages
- Benefits
 - Longevity: Even if projects websites are taken down, users will continue to have access
 - Privacy: Users may be reluctant to upload confidential data
 - Speed: Users locally may have access to more powerful machines for processing data
- `rcellminer` includes examples of embedding Shiny apps in R packages

Getting Help

- Shiny tutorials
 - <http://shiny.rstudio.com>
- Cheatsheet (summary of main functions)
 - <https://www.rstudio.com/resources/cheatsheets/>
- Stack Overflow
 - <http://stackoverflow.com>

Summary

- The Shiny tutorials are excellent
- Almost any website features can be included
- Provides a easy and quick way of presenting data and engaging with users
- Excellent for apps connected with publications
- Complex functionality can be tricky to debug