

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

Data Visualization: Shiny Application



Outline

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

1. Introduction to web applications
2. Reactive Shiny and R framework
3. Shiny application demo - ITMS
4. Practice



Imperative Web Frameworks

Data Visualization: Shiny Application

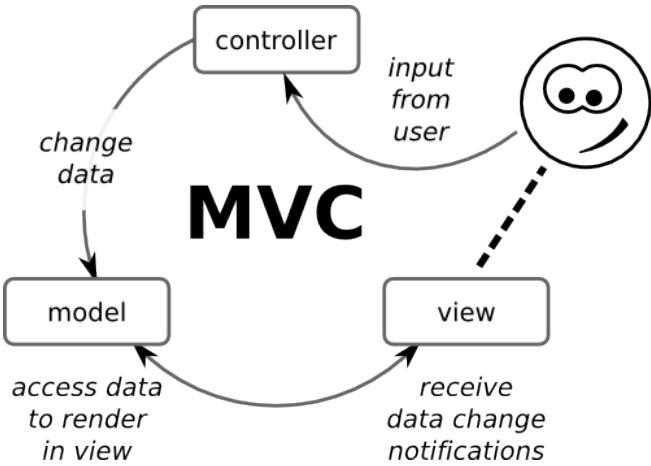
Olga Scrivner

Web Framework

Shiny App

Practice Demo

Traditional Model-View-Controller



The Model-View-Controller (**MVC**) pattern was invented at Xerox Parc by Trygve Reenskaug in 1979



Imperative Web Frameworks

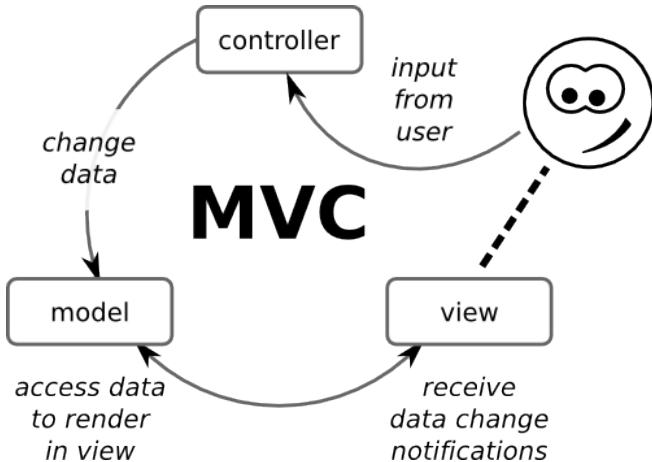
Data Visualization: Shiny Application

Olga Scrivner

Shiny App

Practice Demo

Traditional Model-View-Controller



The Model-View-Controller (**MVC**) pattern was invented at Xerox Parc by Trygve Reenskaug in 1979



“The controller is **essential** and **explicit**: you have to specify what to do when you receive user requests and what resources you are going to mobilize to carry out the necessary tasks outlined in the model” (Ribeiro 2016)

Reactive Web Framework

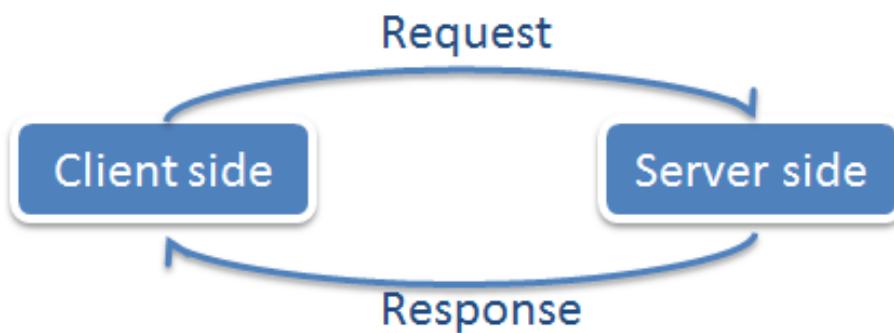
Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo



“Reactive Systems are highly responsive, giving users effective interactive feedback”

<http://www.reactivemanifesto.org/>
<http://littleactuary.github.io/blog/Web-application-framework-with-Shiny/>

Reactive Architecture

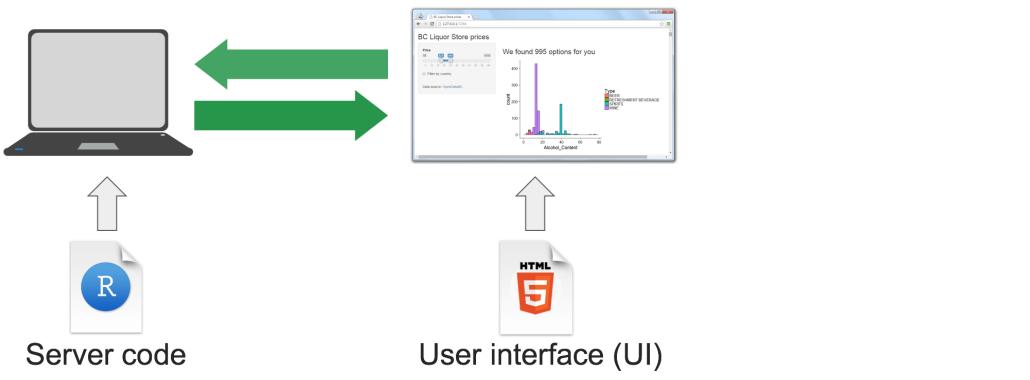
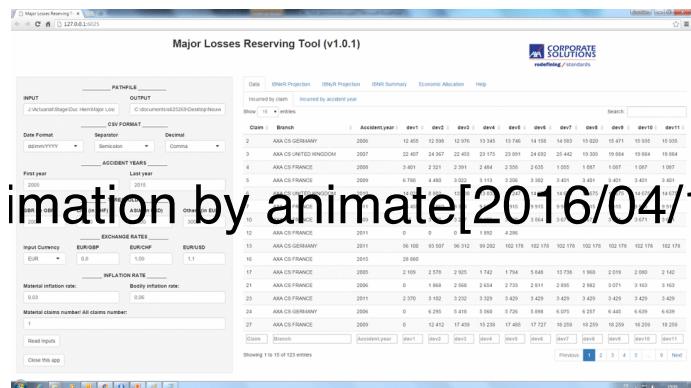
Data Visualization: Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo



Reactive Framework and Data Science

Data Visualization:
Shiny Application

Olga Scrivner

[Web Framework](#)

Shiny App

Practice Demo

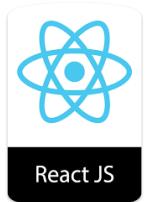
“The impact of data scientists’ work depends on how well others can understand their insights to take further actions”

Benefit 1: Interactive display and manipulation of data

Benefit 2: No installation required

Benefit 3: Easy to develop and share with clients and project teams

Benefit 4: Open source library



<http://datascience.ibm.com/blog/shiny-a-data-scientist-best-friend/>

Shiny Application

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

1. Shiny is an R package for building interactive web applications
2. Open-Sourced by RStudio 11/2012 on CRAN
3. Uses web sockets (new HTTP):
 - ▶ Interactive communication sessions between the user's browser and a server without having to poll the server for a reply
4. Entirely extensible - custom input/output

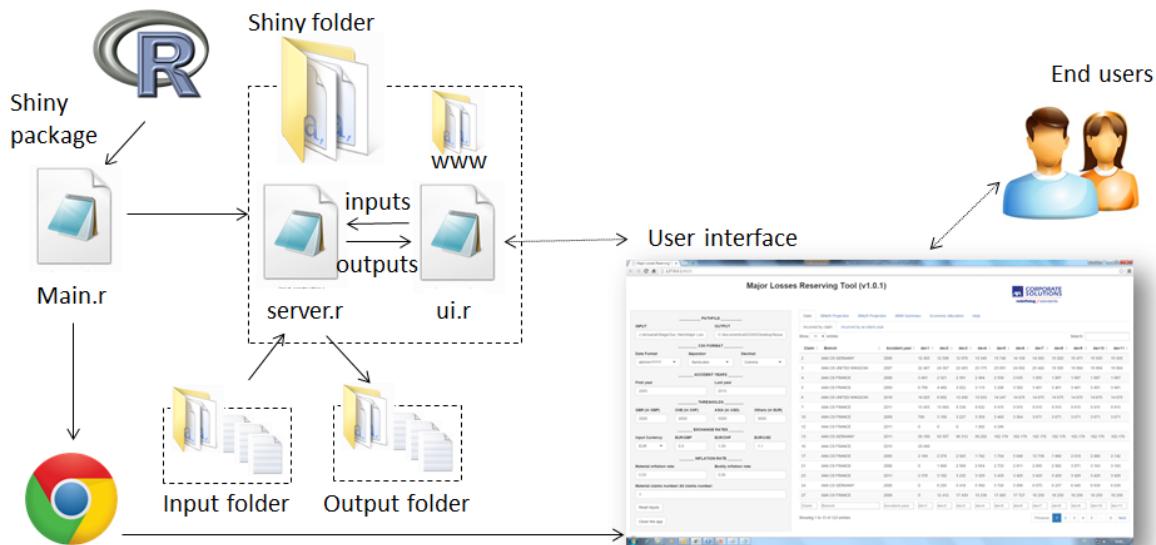
R: Historical Background

R - an implementation of S programming language

- ▶ In 1970s John Chambers, Rick Becker, and Allan Wilks develop S and S+ at Bell Labs
 - ▶ Bell System monopoly was broken up in 1982
 - ▶ Late 80s some attempt to commercialize S/S+ but already too many non-commercial implementations
 - ▶ Ross Ihaka and Robert Gentleman produce R in early 1990s



Shiny Library



Data Visualization: Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

<http://littleactuary.github.io/blog/Web-application-framework-with-Shiny/>

Shiny Gallery - Get Inspired

Data Visualization:
Shiny Application

Olga Scrivner

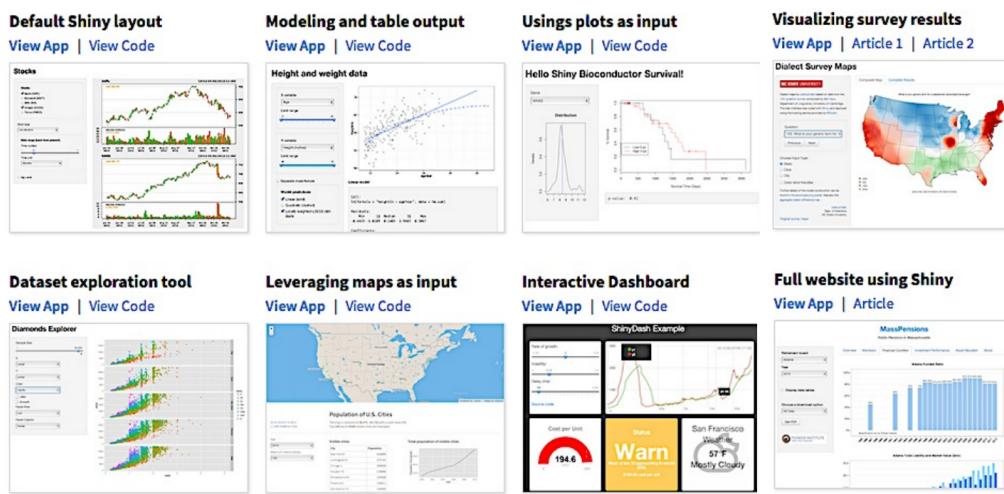
Web Framework

Shiny App

Practice Demo

Users can choose
input parameters with
sliders, drop-downs,
and text fields.

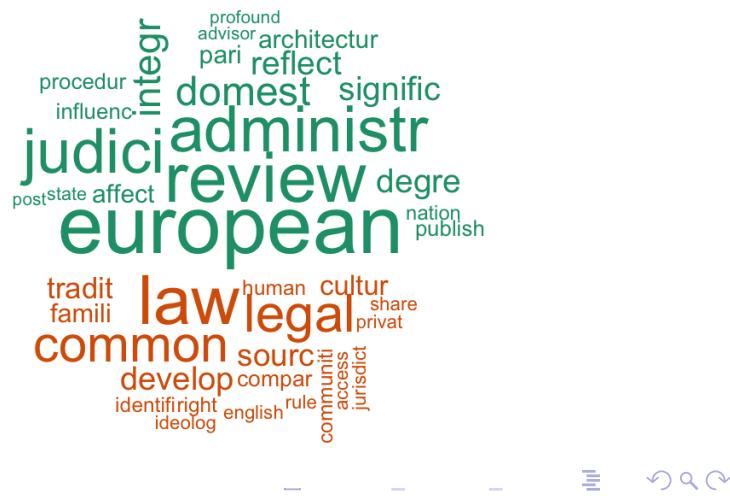
HTML/JavaScript
knowledge not
required.



<https://www.rstudio.com/products/shiny/shiny-user-showcase/>

Interactive Text Mining Suite (Scrivner et al. 2016)

1. Web application for text processing and mining
 2. Interactive natural language processing techniques
 - ▶ Wordstops, stemming, text-preprocessing
 3. High customization



Accessibility

- ▶ PC, Mac, Linux, Smart Phones
 - ▶ Chrome, Firefox, Safari
 - ▶ ITMS Web site:

<http://www.interactivetextminingsuite.com>



ITMS Structure

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo



1. File Uploads

- ▶ Upload files (txt, pdf, rdf, Google books API)

2. Data Preparation

- ▶ Data preprocessing (stopwords, stemming, metadata)

3. Data Visualization

- ▶ Word frequencies, cluster analysis, topic modeling

Stopwords

Stopwords (e.g. **the, and**):

Data Cleaning

Stopwords

Stemming

Metadata

Select Default or Upload

- None
- Default 
- Upload

Default is the list from tm package:
stopwords('SMART')

```
[1] "a"           "a's"  
[5] "above"       "according"  
[9] "actually"    "after"  
[13] "against"    "ain't"  
[17] "allows"     "almost"  
[21] "already"    "also"  
[25] "am"          "among"  
[29] "and"         "another"  
[33] "anyhow"     "anyone"  
[37] "anyways"    "anywhere"  
[41] "appreciate" "appropriate"  
[45] "around"      "as"  
[49] "asking"      "associate"  
[53] "away"        "awfully"
```

Manual Removal of Stopwords

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

Manual Removal

Select one or multiple words (hold shift key down)

Select words to be removed

made written | ←

- subject
- supported
- systems
- textbooks
- training
- ultimately
- union
- voluntary

judiciary jurisprudence led legislative made
marriage member needed notes operation
organization parent perspective polity
possess practice preference procedures
produce progeny proper provided qualify

Viewer

Apply Stopwords or None (no changes)

- apply ←
- none

private law field legislative intervention
rendering eu law part national legal system
requiring courts account jurisprudence
europen society polity proper organization

Stemming

Data Visualization: Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

Stems - tm package

Choose Language

- none
 - English
 - Spanish
 - Danish
 - Dutch
 - Finnish

Stem Viewer

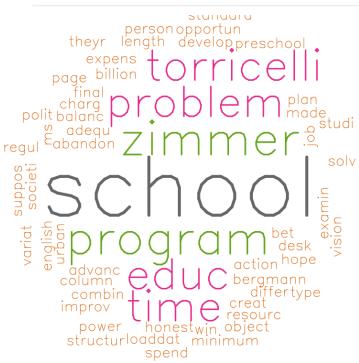
privat law field legisl intervent render eu law part nation legal system requir court account jurisprud european societi politi proper organ oper legal system law appli studi sourc led reconfigur common law legal famili parent legal system enter marriag give rise progeni privat european communiti act european union law legal effect nation legal system basi nation court requir appli eu law remain conceptu strike heart domest legal system hold state court subject eu law requir note prefer australia canada zealand legal system analysi support earlier studi year train need qualifi practic favour legal system possess subconsci bias system procedure vis mi member state embed legal system conceiv state complianc eu law voluntar act reli extent law part uk legal system dealt textbook uk academ general access english ultim produc judgment higher qualiti give judiciari perspect legal system court benefit fulli insight provid compar law



Word Cloud Representation

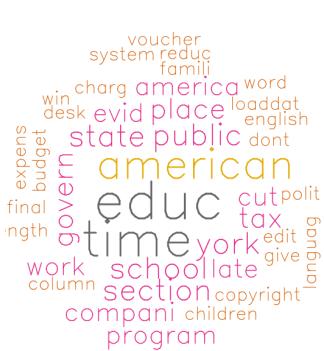
Cloud Type

- Word Cloud
- Commonality Cloud
- Comparison of two or more docs



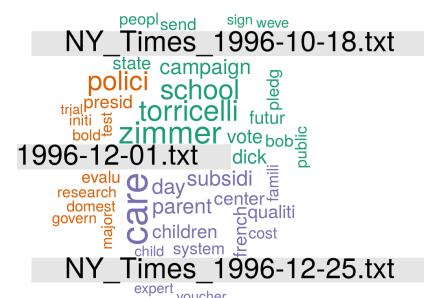
Cloud Type

- Word Cloud
- Commonality Cloud
- Comparison of two or more docs



Cloud Type

- Word Cloud
- Commonality Cloud
- Comparison of two or more docs



Data Visualization: Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

Customization

Data Visualization: Shiny Application

Olga Scrivner

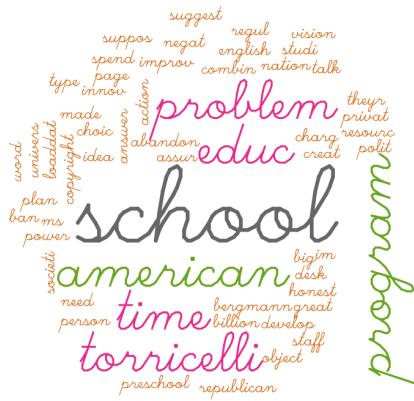
Web Framework

Shiny App

Practice Demo

Select Font

- Sans Serif
 - Script
 - Gothic



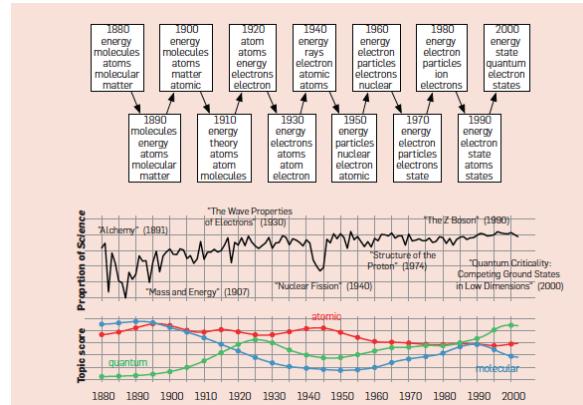
Select Color Palette

- black
 - green
 - multi



Topic Modeling

Discovering underlying theme of collection from *Science* magazine
1990-2000 (Blei 2012)



- ▶ **LDA** (Latent Dirichlet allocation)
- ▶ **STM** (Structural Topic model)
- ▶ Chronological topic visualization (lda): requires metadata

Topic Modeling Tuning

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

- ▶ Selection of topics (how many different themes)
- ▶ Selection of words per theme (how many words per topic)
- ▶ Selection of iteration

Run LDA Analysis

- none
 run

Selected Topics LDA (`lda.collapsed.gibbs.sampler` package)

V1	V2	V3
policy	children	public
evidence	care	zimmer
president	vouchers	schools

Using Google Books API

Select your search terms and submit

Choose file format

- XML
- JSON
- Google Books Search

Enter your search terms for Google Books,
separated by spaces

social science

Submit

Current limitation is 40 books

Show 25 entries

Search:

titles	authors	dates	corpus
Readings in the Philosophy of Social Science	Michael Martin, Lee C. McIntyre	1994	Readings in the Philosophy of Social Science the first comprehensive anthology in the philosophy of social science to appear since the late 1960s
Dictionary of the Social	Craig Calhoun	2002	Dictionary of the Social Sciences Defines key



Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

Workshop Materials

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

1. Rstudio
2. R
3. Shiny library
4. Some ideas are from <http://deanattali.com/blog/building-shiny-apps-tutorial/>

R software

R is a **free** software for data analysis, text mining and visualization.

To install R on Window:

1. Download the binary file for R

[https://cran.r-project.org/bin/windows/base/
R-3.3.1-win.exe](https://cran.r-project.org/bin/windows/base/R-3.3.1-win.exe)

2. Open the downloaded .exe file and Install R

To install R on Mac:

1. Download the appropriate version of .pkg file

<https://cran.r-project.org/bin/macosx/>

2. Open the downloaded .pkg file and Install R

R Studio

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

RStudio is a free user interface for R.

1. Install the appropriate RStudio version <https://www.rstudio.com/products/rstudio/download/>
2. Run it to install R-studio

R Studio Structure

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

RStudio screen

The screenshot shows the RStudio interface with the following sections:

- Console:** Displays R code and its output. The code includes matrix operations and file creation.
- Workspace:** Shows two objects: A (4x2 double matrix) and B (4x2 double matrix).
- Files:** Shows a single file named "History".
- Plots:** No plots are currently displayed.

Text Labels:

- The workspace tab shows all the active objects (see next slide). The history tab shows a list of commands used so far.
- The files tab shows all the files and folders in your default workspace as if you were on a PC/Mac window. The plots tab will show all your graphs. The packages tab will list a series of packages or add-ons needed to run certain processes. For additional info see the help tab

The console is where you can type commands and see output

Installing Packages

Data Visualization:
Shiny Application

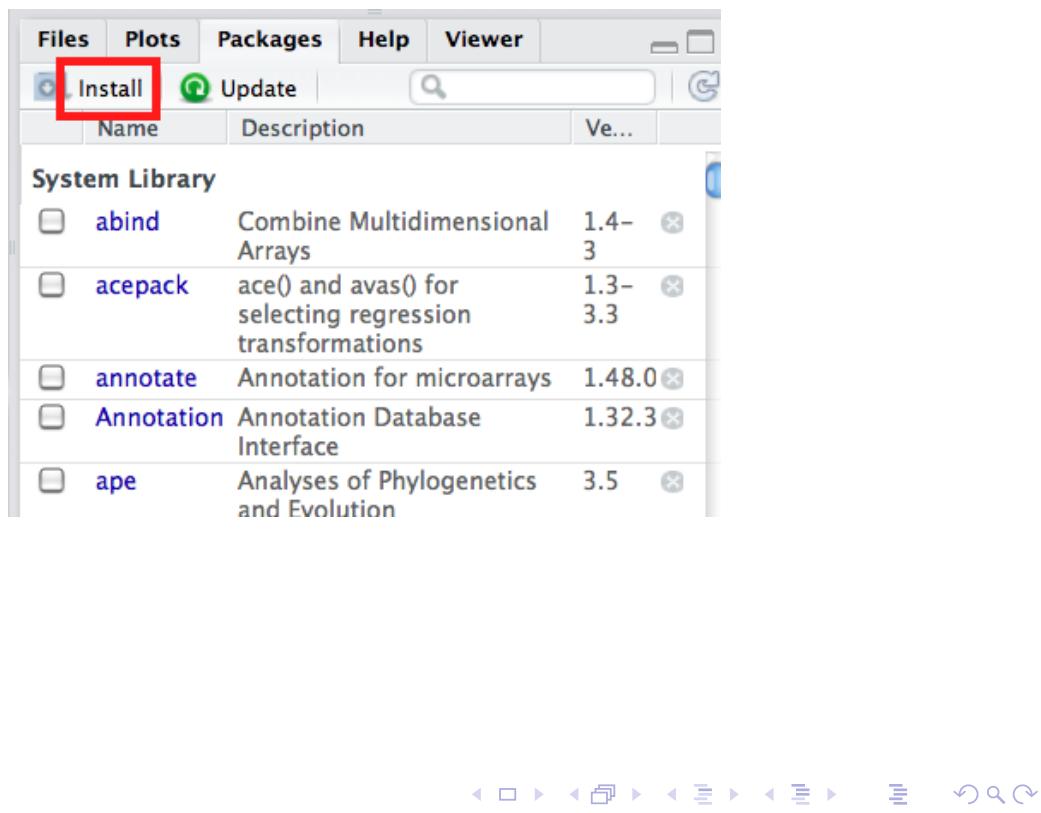
Olga Scrivner

Web Framework

Shiny App

Practice Demo

In your bottom left window - go to **Packages**



Selecting Packages - shiny

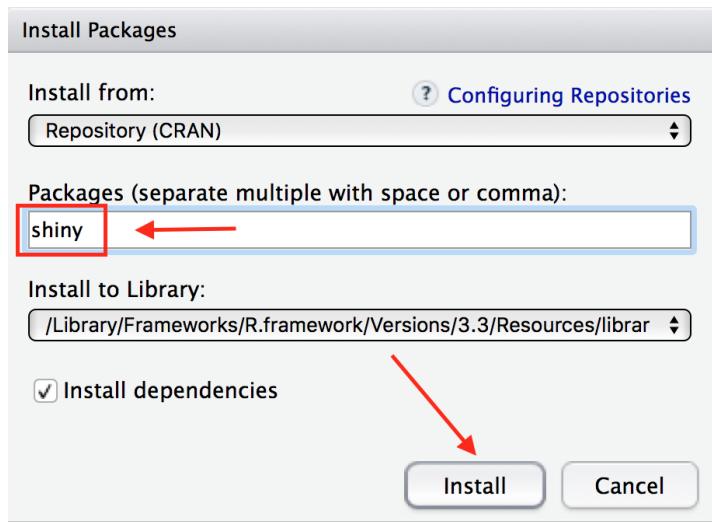
Data Visualization: Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo



Creating Scripts

Data Visualization: Shiny Application

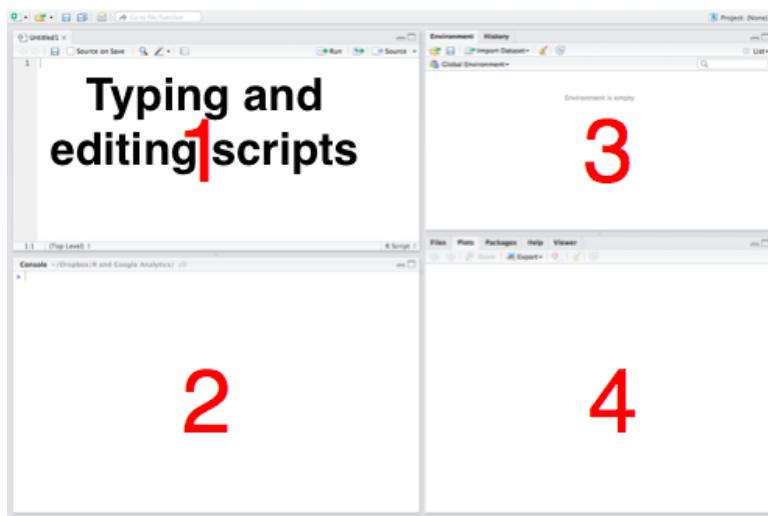
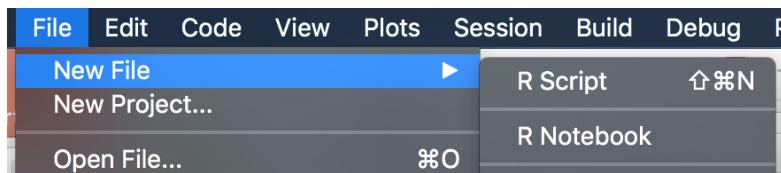
Olga Scrivner

Web Framework

Shiny App

Practice Demo

Create R File: **File** → **New File** → **R Script**



Execution - RUN

Data Visualization:
Shiny Application

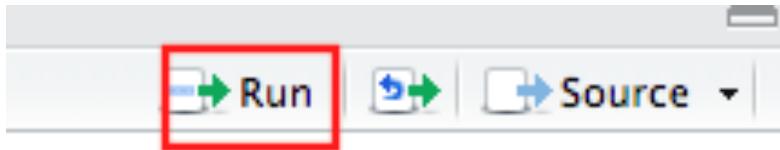
Olga Scrivner

Web Framework

Shiny App

Practice Demo

To execute your commands you need to click **run**



Type in the script and run:

```
library(shiny)
```

Shiny Demo

Data Visualization: Shiny Application

Olga Scrivner

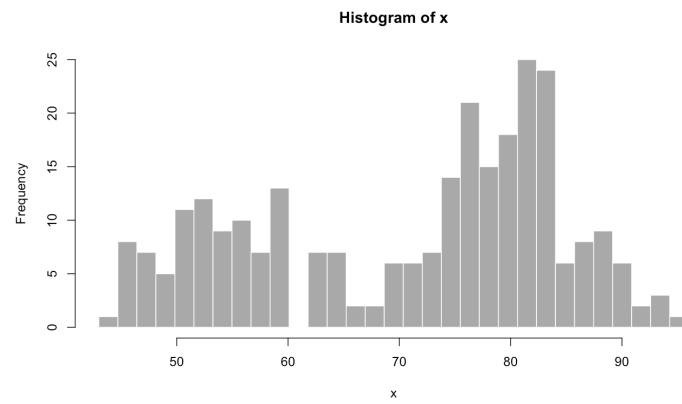
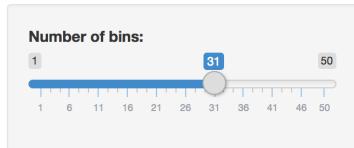
Web Framework

Shiny App

Practice Demo

```
runExample("01_hello")
```

Hello Shiny!



Shiny Demo

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

Hello Shiny!
by RStudio, Inc.

This small Shiny application demonstrates Shiny's automatic UI updates. Move the *Number of bins* slider and notice how the `renderPlot` expression is automatically re-evaluated when its dependant, `input$bins`, changes, causing a histogram with a new number of bins to be rendered.

server.R ui.R

show with app

```
library(shiny)

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

  # Expression that generates a histogram. The expression is
  # wrapped in a call to renderPlot to indicate that:
  #
  # 1) It is "reactive" and therefore should be automatically
  #    re-executed when inputs change
  # 2) Its output type is a plot

  output$distPlot <- renderPlot({
    x   <- faithful[, 2] # Old Faithful Geyser data
    bins <- seq(min(x), max(x), length.out = input$bins + 1)

    # draw the histogram with the specified number of bins
  })
}
```

Shiny Demo

Data Visualization:
Shiny Application

Olga Scrivner

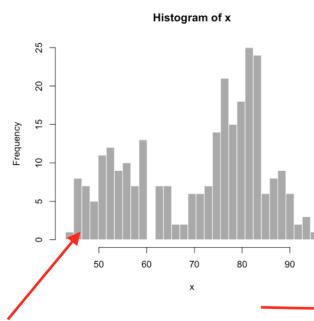
Web Framework

Shiny App

Practice Demo

Hello Shiny!

Number of bins:



server.R ui.R ↴ show below

```
library(shiny)

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

  # Expression that generates a histogram. The expression is
  # wrapped in a call to renderPlot to indicate that:
  #
  # 1) It is "reactive" and therefore should be automatically
  #    re-executed when inputs change
  # 2) Its output type is a plot

  output$distPlot <- renderPlot({
    x     <- faithful[, 2] # Old Faithful Geyser data
    bins <- seq(min(x), max(x), length.out = input$bins + 1)

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

UI.R

```
server.R ui.R

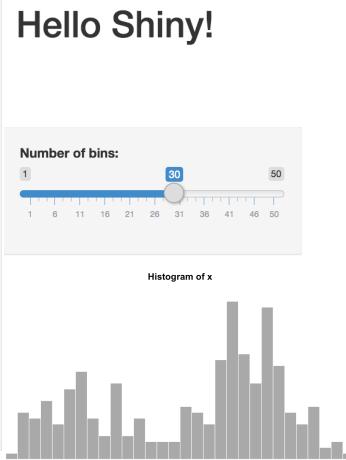
library(shiny)

# Define UI for application that draws a histogram
fluidPage(


# Application title
titlePanel("Hello Shiny!"),

# Sidebar with a slider input for the number of bins
sidebarLayout(
  sidebarPanel(
    sliderInput("bins",
               "Number of bins:",
               min = 1,
               max = 50,
               value = 30)
  ),
  mainPanel(
    plotOutput("distPlot")
  )
)
```

↓ show below



Reactive Input/Output

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

```
server.R ui.R
```

```
library(shiny)

# Define UI for application that draws a histogram
fluidPage(
  # Application title
  titlePanel("Hello Shiny!"),

  # Sidebar with a slider input for the 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")
    )
  )
)
```

```
server.R ui.R
```

```
library(shiny)

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

  # Expression that generates a histogram. The expression is
  # wrapped in a call to renderPlot to indicate that:
  #
  # 1) It is "reactive" and therefore should be automatically
  #    re-executed when inputs change
  # 2) Its output type is a plot

  output$distPlot <- renderPlot({
    x <- faithful[, 2] # Old Faithful Geyser data
    bins <- seq(min(x), max(x), length.out = input$bins + 1)

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

- ▶ Input - things user can toggle
- ▶ Output - R objects that user can see, often depend on inputs

Closing App

Data Visualization: Shiny Application

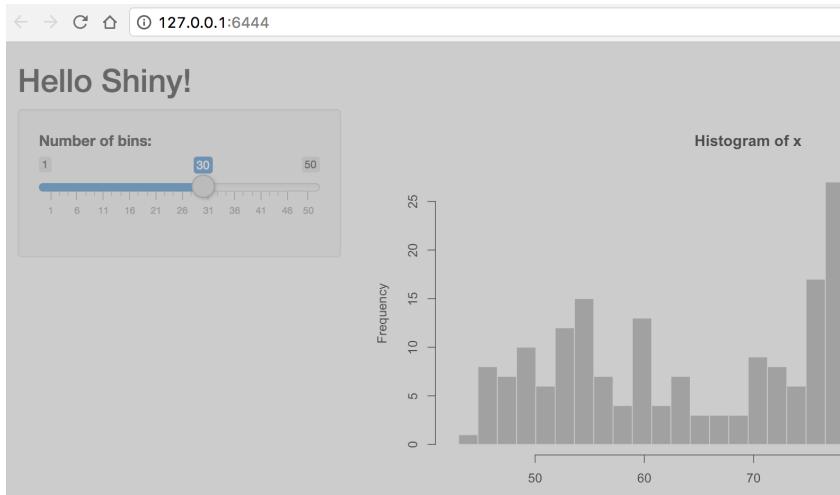
Olga Scrivner

Web Framework

Shiny App

Practice Demo

```
Console ~ /     
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.  
  
> library("shiny", lib.loc="/Library/Frameworks/R.framework/Versions/3.3/Resources/library")  
Warning message:  
package 'shiny' was built under R version 3.3.2  
> runExample("01_hello")  
  
Listening on http://127.0.0.1:6444
```



Step 1 - Create New Web Shiny App

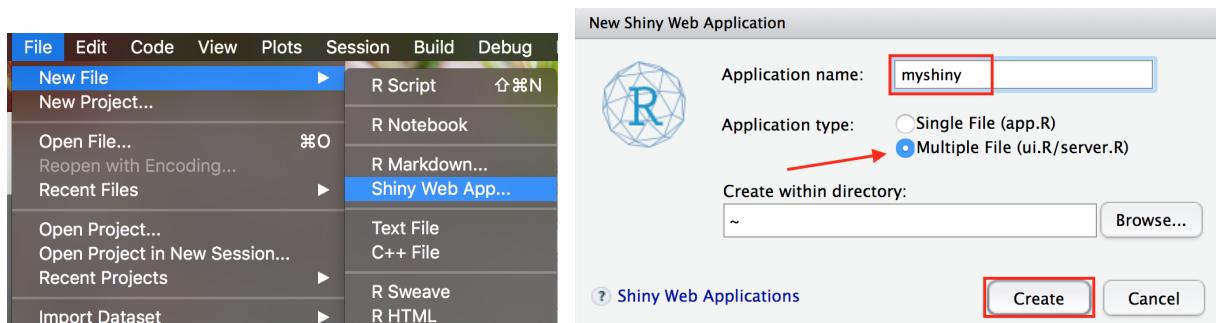
Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo



Step 2 - Run App

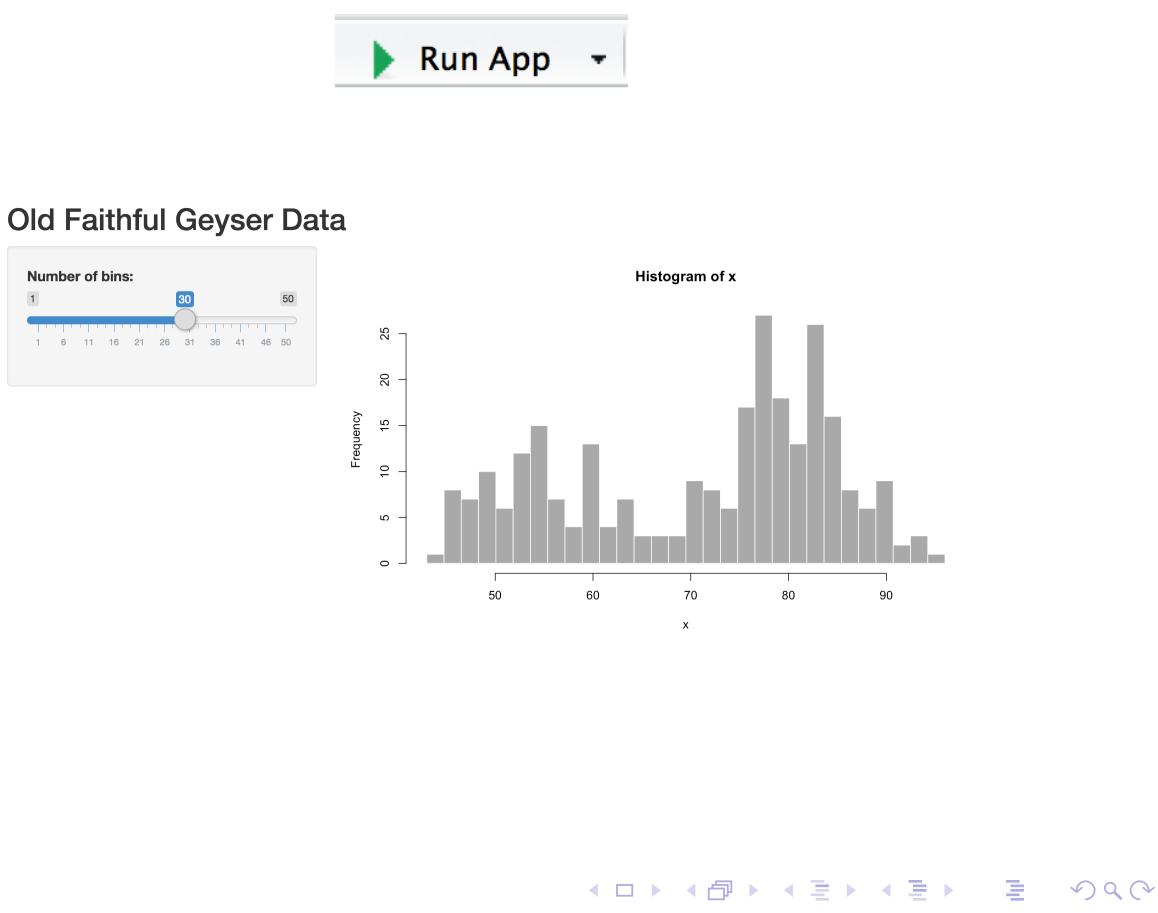
Data Visualization: Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo



UI

```
shinyUI(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")  
    )  
  )
```

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

Modifying UI - Practice

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

1. Change Title
2. Change Bins values: min, max and value
3. Save
4. RunApp

HTML

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

The UI script simply creates HTML

```
<body>
  <div class="container-fluid">
    <h2>Old Faithful Geyser Data</h2>
    <div class="row">
      <div class="col-sm-4">
        <form class="well">
          <div class="form-group shiny-input-container">
            <label class="control-label" for="bins">Number of bins:</label>
            <input class="js-range-slider" id="bins" data-min="1" data-max="50" data-from="30"
data-step="1" data-grid="true" data-grid-num="9.8" data-grid-snap="false" data-keyboard="true"
data-keyboard-step="2.04081632653061" data-drag-interval="true" data-data-type="number" data-
pretty-separator=","/>
          </div>
        </form>
      </div>
      <div class="col-sm-8">
        <div id="distPlot" class="shiny-plot-output" style="width: 100% ; height: 400px">
      </div>
    </div>
  </div>
</body>
```

HTML

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

HTML tags:

`http://shiny.rstudio.com/articles/tag-glossary.html`

- ▶ `h1()` = header1
- ▶ `br()` = line break
- ▶ `p()` = paragraph
- ▶ `hr()` = line

Adding HTML Tag to UI.R - Practice

```
shinyUI(fluidPage(  
# Application title  
titlePanel("My Title"),  
h3("My subtitle"),  
p("This is my first app!"),  
br(),  
hr(),
```

Adding HTML Tag to UI.R - Practice

```
shinyUI(fluidPage(  
# Application title  
titlePanel("My Title"),  
h3("My subtitle"),  
p("This is my first app!"),  
br(),  
hr(),
```

RunApp

My Title

My subtitle

This is my first app!

Layout

Data Visualization: Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

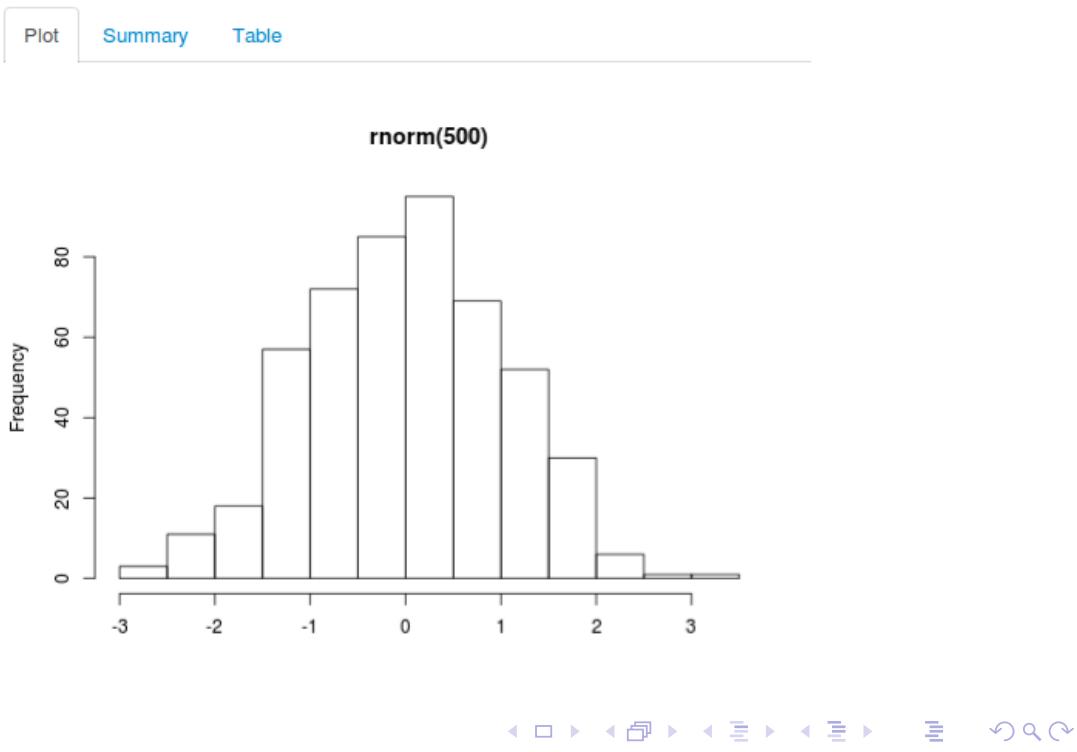


`http://shiny.rstudio.com/articles/layout-guide.html`

mainPanel

```
mainPanel( plotOutput("distPlot"))
```

Let's add 3 tab panels: **Plot, Summary, Table**



Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

tabsetPanel

```
mainPanel(  
  tabsetPanel(  
    tabPanel("Plot", plotOutput("distPlot")),  
    tabPanel("Summary"),  
    tabPanel("Table")  
)  
)
```

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

tabsetPanel

```
mainPanel(  
  tabsetPanel(  
    tabPanel("Plot", plotOutput("distPlot")),  
    tabPanel("Summary"),  
    tabPanel("Table")  
)  
)
```

My Title

My subtitle

This is my first app!



Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

Input Data

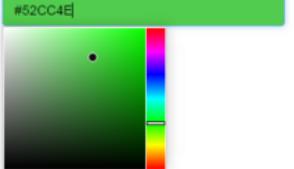
Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

Buttons <code>Action</code> <code>Submit</code> <code>actionButton()</code> <code>submitButton()</code>	Single checkbox <code>checkboxInput()</code> <code>checkboxGroupInput()</code>	Checkbox group <code>checkboxInput()</code> <code>checkboxGroupInput()</code>	Date input <code>dateInput()</code>	Colour input <code>#52CC4E</code>  <code>shinyjs::colourInput()</code>
Date range <code>dateRangeInput()</code>	File input <code>fileInput()</code>	Numeric input <code>numericInput()</code>	Password Input <code>passwordInput()</code>	
Radio buttons <code>radioButtons()</code>	Select box <code>selectInput()</code>	Sliders <code>sliderInput()</code>	Text input <code>textInput()</code>	

Input Data

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

1. Blog <https://languagevariationsuite.wordpress.com/>
2. Download csv file - movie_metadata.csv
3. Place this file into the directory **myshiny**

New Input in UI.R

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

We will add **fileInput** function inside **sidebarPanel**:

NB: commas are important!

```
sidebarPanel(  
  sliderInput(.....),  
  fileInput()  
)
```

New Input in UI.R

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

```
fileInput('file1', 'Choose CSV File',  
multiple=FALSE,  
accept=c('text/csv',  
'text/comma-separated-values,text/plain',  
'.csv')  
)  
  
sidebarLayout(  
  sidebarPanel(  
    sliderInput("bins",  
      "Number of bins:",  
      min = 1,  
      max = 50,  
      value = 30),  
    fileInput('file1', 'Choose CSV File', multiple=FALSE,  
      accept=c('text/csv',  
              'text/comma-separated-values,text/plain',  
              '.csv'))  
,  
)
```

Reactive Function in Server.R

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

```
myfile <- reactive({  
  if (is.null(input$file1)){return()}  
  mydata <- read.csv(input$file1$datapath,  
  header=TRUE, sep=',')  
  return(mydata)  
})  
  
shinyServer(function(input, output) {  
  
  myfile <- reactive({  
    if (is.null(input$file1)){return()}  
    mydata <- read.csv(input$file1$datapath, header=TRUE, sep=',')  
    return(mydata)  
})
```

To use reactive data we will pass **myfile()** content to output functions

New Output Function **summary** in Server.R

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

```
output$summary <- renderPrint({  
  summary(myfile())  
})
```

New Output Function **table** in Server.R

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

```
output$table <- renderDataTable({  
  myfile()  
})
```

Linking **summary** and **table** with UI.R

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

```
tabsetPanel(  
  
  tabPanel("Plot", plotOutput("distPlot")),  
  
  tabPanel("Summary", verbatimTextOutput("summary")),  
  
  tabPanel("Table", dataTableOutput("table"))  
)
```

Linking **summary** and **table** with UI.R

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

```
tabsetPanel(  
  
  tabPanel("Plot", plotOutput("distPlot")),  
  
  tabPanel("Summary", verbatimTextOutput("summary")),  
  
  tabPanel("Table", dataTableOutput("table"))  
)
```

RunApp

Testing CSV Upload

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

My Title

My subtitle

This is my first app!

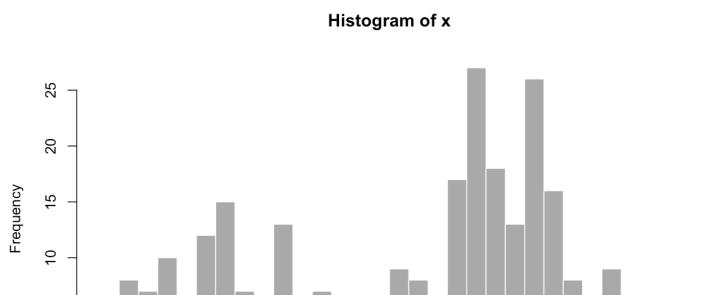
Number of bins:

1 30 50

Choose CSV File

Browse... No file selected

Plot Summary Table



One more IF-Statement - Server.R

We want to do a histogram for csv file:

```
if (condition) {do..} else {do...}
```

One more IF-Statement - Server.R

We want to do a histogram for csv file:

```
if (condition) {do..} else {do...}  
  
output$distPlot <- renderPlot({  
  if (is.null(input$file1)) {  
    ....  
    hist(x, breaks = bins, col = 'darkgray', border =  
      'white')  
  }  
})
```

One more IF-Statement - Server.R

We want to do a histogram for csv file:

```
if (condition) {do..} else {do...}

output$distPlot <- renderPlot({
  if (is.null(input$file1)) {
    ....
    hist(x, breaks = bins, col = 'darkgray', border =
      'white')
  }

  else{
    x <- myfile()$budget
    bins <- seq(min(x), max(x), length.out = input$bins
      + 1)
    hist(x, breaks = bins, col='red',
      main = 'My New Histogram')
  }
})
```

RunApp

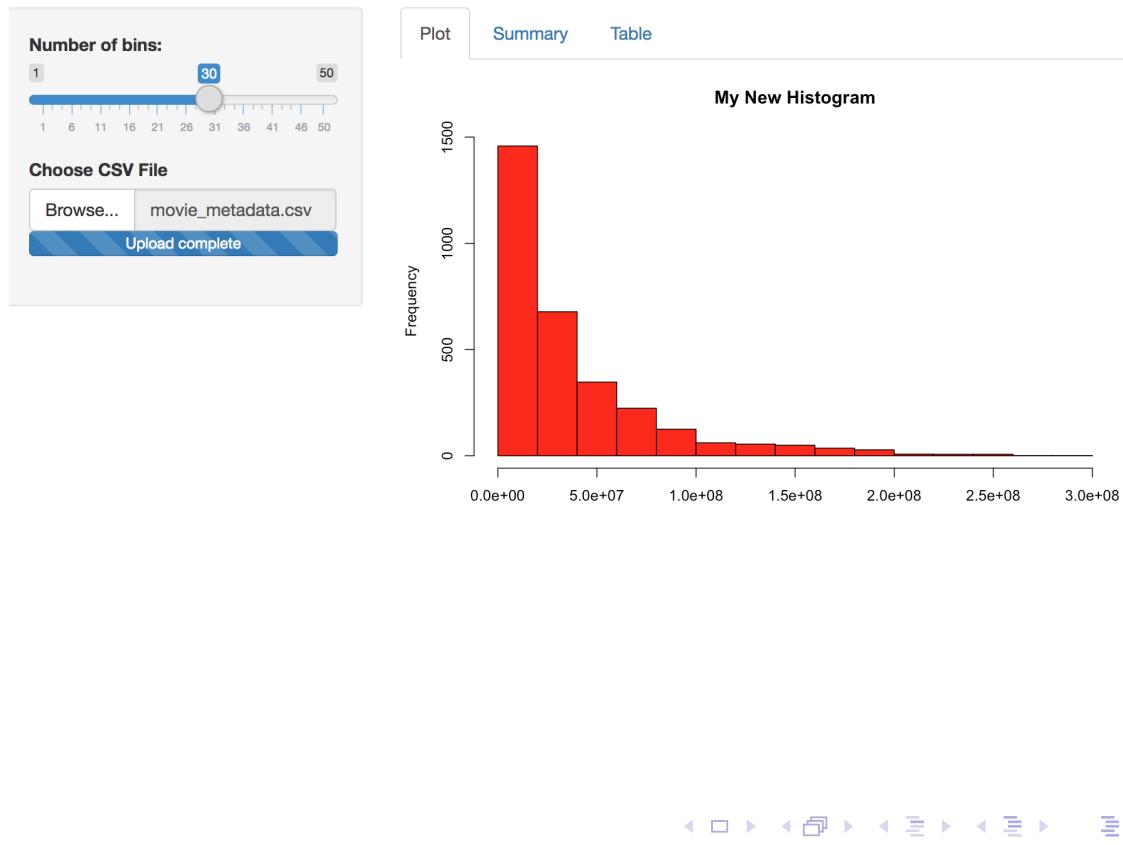
Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo



Deployment Options

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

1. Share server.r and ui.r
2. Host on shinyapps.io
3. Host on Shiny server



Deploy with shinyapps.io

Data Visualization:
Shiny Application

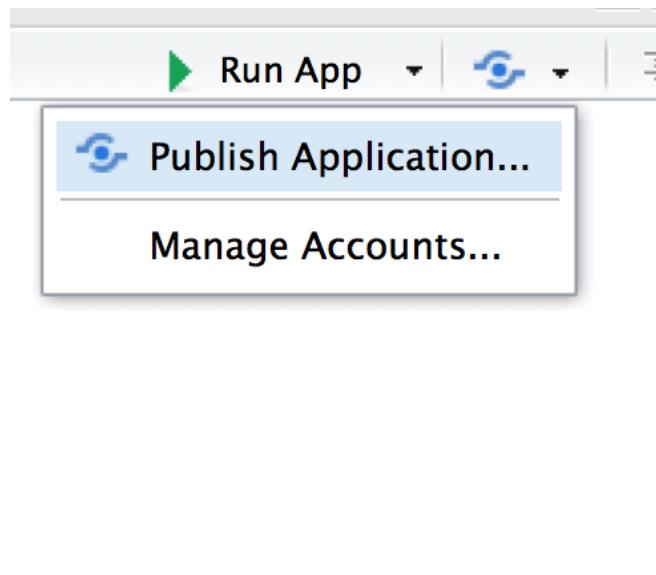
Olga Scrivner

Web Framework

Shiny App

Practice Demo

- ▶ www.shinyapps.io
- ▶ sign up for an account.
- ▶ **Publish Application** button in RStudio and follow instructions



Deploy with shinyapps.io

Data Visualization: Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

Publish to Server



Publish Files From: ~/myshiny

- movie_metadata.csv
 - server.R
 - ui.R



Destination Account: [Add New Account](#)

Add New Account



 languagevariationsuite: shinyapps.io

Title:

myshiny

Uncheck All

Launch browser

Publish

Cancel

My shinyapps.io

Data Visualization: Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

The screenshot shows the shinyapps.io dashboard interface. On the left, a dark sidebar contains links for Dashboard, Applications (with a count of 2), and Account. The main area features a "WHAT'S NEW?" section with a large blue box titled "APPLICATIONS ONLINE" showing 2 applications, one of which is "Running". Below this, a "RECENT APPLICATIONS" table lists three entries:

ID	Name	Status
74932	Pages	Running
82881	TextMining	Sleeping
143129	TextMiningBeta	Archived

At the bottom, there is a navigation bar with icons for back, forward, search, and other application management functions.

My shinyapps.io

Data Visualization: Shiny Application

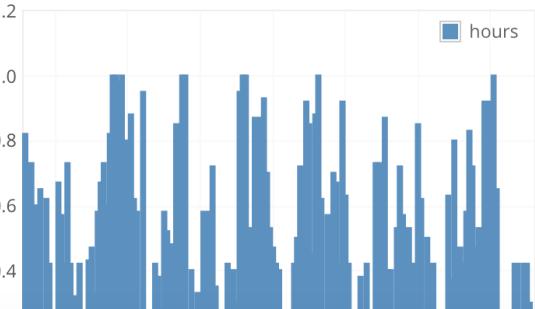
Olga Scrivner

Web Framework

Shiny App

Practice Demo

APPLICATION 74932 – PAGES

 Overview	 Metrics	 URLs	 Settings	 Users	 Logs	 Restart	 Archive	 Delete
OVERVIEW			INSTANCES					
Id	74932		Id: 656444					
Name	Pages							
URL	https://languagevariationsuite.shinyapps.io/Pages/							
Status	Running							
Size	xlarge							
Deployed	Dec 21, 2016							
Updated	Feb 20, 2017							
APPLICATION USAGE								
Total: 78.67 hours								
								

Useful resources

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

1. Shiny official tutorial -

<http://shiny.rstudio.com/tutorial>

2. Cheat sheet - <http://shiny.rstudio.com/images/shiny-cheatsheet.pdf>

3. Publish your app free - <http://www.shinyapps.io>

4. Examples -<http://www.showmeshirey.com/>

5. Tutorial by Dean Attali - <http://deanattali.com/blog/building-shiny-apps-tutorial/>

Thank you!

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

My email: obscrivn@indiana.edu

Credits

Data Visualization:
Shiny Application

Olga Scrivner

Web Framework

Shiny App

Practice Demo

<https://github.com/IBMDatascience/dsx-shiny-apps>

<http://www.slideshare.net/SarahAerni/>

data-science-as-a-commodity-use-madlib-r-other-oss-tools-for-data-

[http://www.unixstickers.com/image/data/stickers/
react/badge/React-JS.sh.png](http://www.unixstickers.com/image/data/stickers/react/badge/React-JS.sh.png)

<https://github.com/rstudio/shiny/issues/250>

[http://www.slideshare.net/ilios-catallo/
spring-mvc-the-basics](http://www.slideshare.net/ilios-catallo/spring-mvc-the-basics)