

How to create applications using Shiny

ISGlobal

Barcelona, February 13-14th 2018

Part I: Introduction



Outline

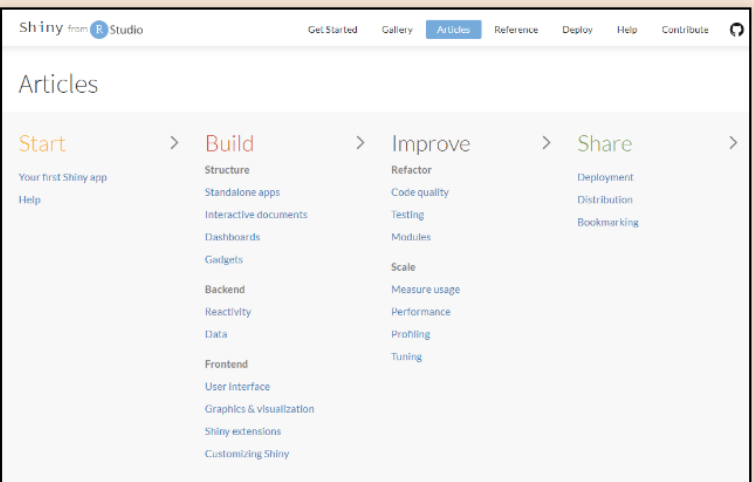
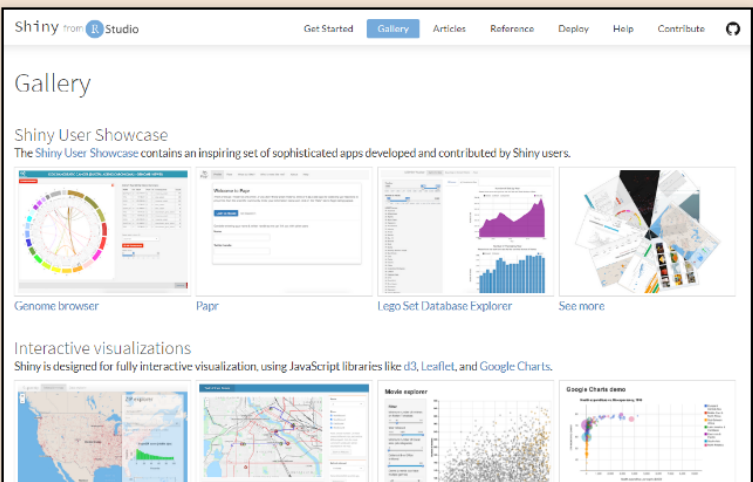
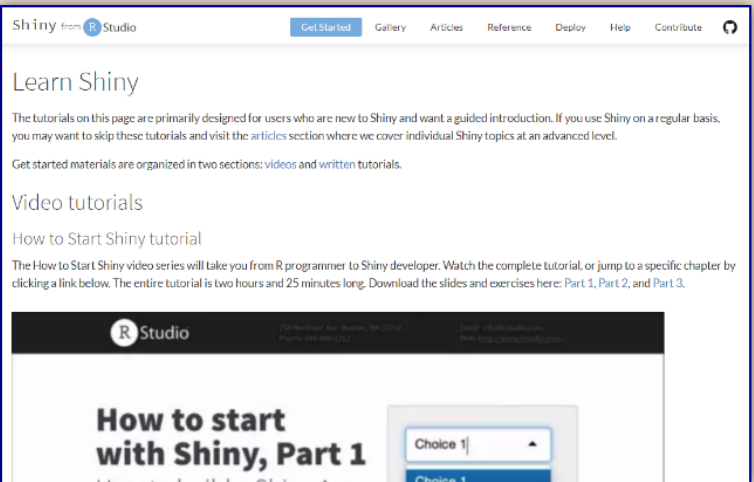
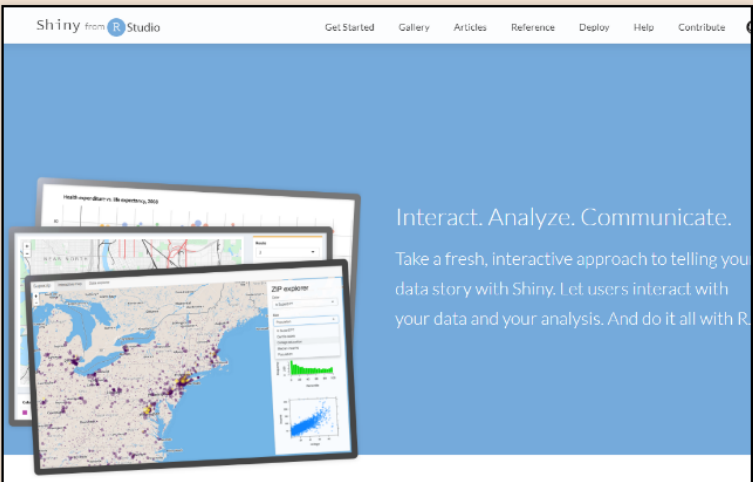
- **Part II:** Form design
 - Input elements
 - Output elements
 - Layout
 - Conditional panels
- **Part III:** Logic of Shiny
 - How Shiny works
 - Isolate
 - Reactive objects
 - Upload data
 - Download files
 - Validate inputs
 - Rendering elements
- **Part IV:** Ways to improve the application
 - HTML and CSS
 - Pop-ups and Modals
 - Collapse panels
 - Carrouselles
 - Themes (app appearance)
 - Sizeable
 - Input alerts
- **Part V:** Advanced issues
 - observe and observeEvent functions.
 - Updating elements
 - Reactive variables
 - hide, show, toggle and disable functions

Introduction

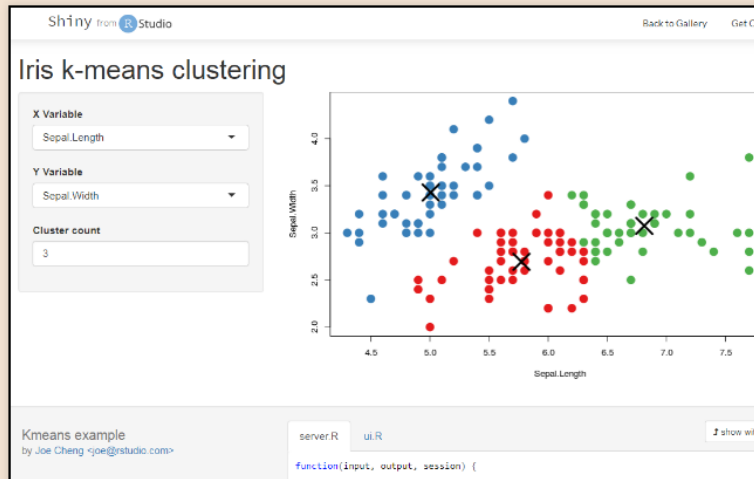
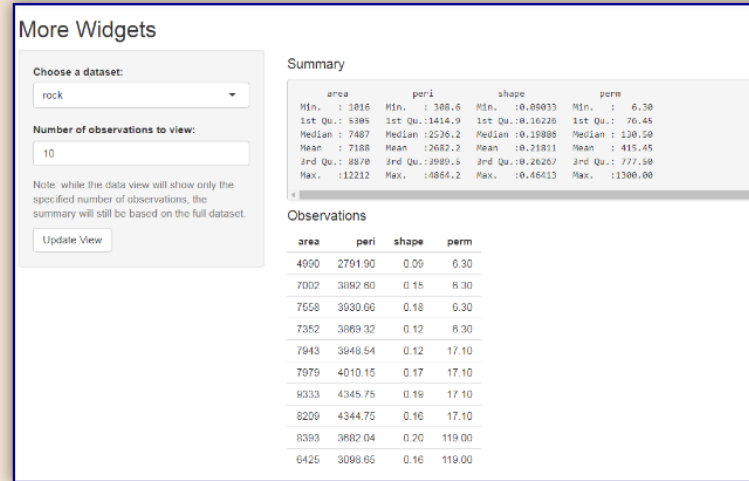
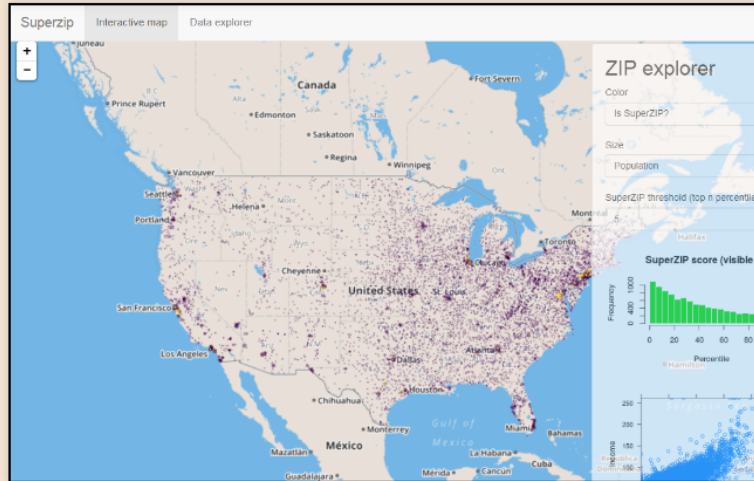
What is Shiny?

- **Shiny** is an **R** package meant to build web based interactive apps.
- It has been created by **RStudio** team. (**Joe Chen**)
- No knowledge of web languages such as **HTML**, **JavaScript** or **PHP** is required. It only uses **R** code.
- Very flexible and powerful apps can be designed by writing small pieces of code.
- Useful to make your R written functions or packages be used by many users not familiarized with R who prefers “click” than “type”.
- See [Shiny web page](#) with lots of examples and extensive help documents.

Shiny website pages



Shiny website examples



MAVIS: Meta Analysis by Shiny

[Main Menu](#)
[Input Examples](#)
[Inter-rater reliability](#)
[Model Options and Settings -](#)
[Publication Bias -](#)
[Effect Size Calculator -](#)
[About](#)

Data Analysis and Input Options:

- ☒ Mean Differences (*n*, *M*, *SD*)
- ☐ Mean Differences (*n*, *SE*, *Size d*)
- ☐ Correlations (*n*, *r*)
- ☐ Dichotomous Models

See Dichotomous Model Options, default is set to log odds ratio

☒ The data contains a categorical moderator (subgroup) variable.

Update View

Click here to update your results, you need to do this after you change the data, model, or any of the settings

Note: Input values must be separated by tabs. Copy and paste from Excel

Your data examples will have exactly the same header (variable names) in the first row

For details of how this data should look click on the Input Examples tab

1	Studdy Moderator	N1	M1	SD1	N2	M2	SD2
2	Burkeham et al (2011)	INTERNET	30	55.84	18.23	38	43.58
3	Fleiderman et al (2012)	BOOK	125	134.73	36.17	156	115.93
4	Ruck & Mewster (2013)	INTERNET	25	30.27	5.30	71	48.64
5	Hesser et al (2012)	INTERNET	35	46.27	9.69	32	36.01
6	Joffe Hayes (2012)	BOOK	180	125.01	10.72	199	122.15
7	Johnson et al (2012)	BOOK	5	77.4	1.9	6	62.3
8	Lappalainen et al (2013)	INTERNET	12	55.73	6.45	12	53.67
9	Marling et al (2011)	INTERNET	154	3.88	8.62	154	3.65
10	Muto et al (2011)	BOOK	38	44.3	6.67	51	43.48
11	Tinswail et al (2011)	BOOK	52	82.5	20.91	56	50.11

Data for this example is from the



How to use Shiny: (1) locally

Requirements:

R and **shiny** package must be installed in your PC or Mac.

Steps

1. Write the app code in two files named “**ui.R**” and “**server.R**” placed in the same folder.
2. Call `runApp()`.

Alternative

1. Write all code in a single file file named **app.R**.
2. call `runApp(list(ui, server))` or `shinyApp(ui, server)`.

How to use Shiny: (2) remote server

Own server

- Advantages
 - Accessible from any device with internet explorer.
 - No need to have **R**, **shiny** or other packages/software.
- Requirements
 - **R**, **shiny** and other required packages/software installed in the server.
 - Install required software. More info [here](#).
 - OS must be Linux.

Shiny server

- Advantages
 - Very simple: only need to install shinyapps and registrar to www.shinyapps.io/.
 - Accessible from any device with internet explorer.
 - **Shiny server** offers use statistics.
- Inconveniences
 - Free up to a maximum number of hours and users.

Shiny extensions: used in this course

Packages available on CRAN to improve both app appearance and functionality.

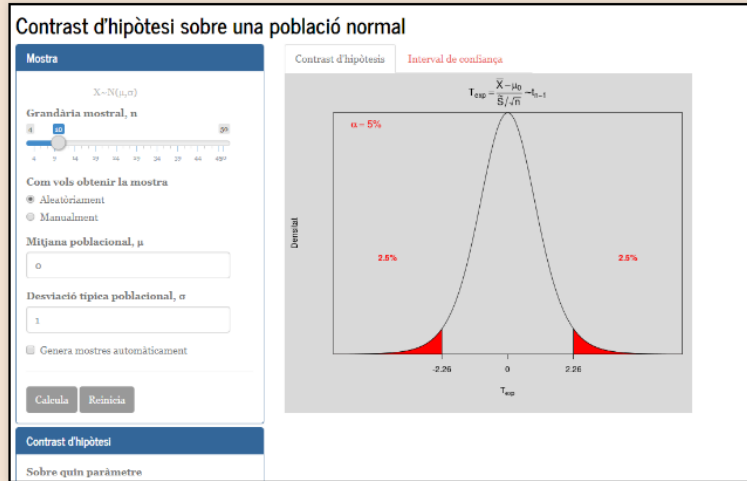
The ones we will see in the course:

- **shinyBS**: to create pop-ups, modals,... Visit its website [here](#)
- **bsplus**: complementary to shinyBS. Also, it creates carrouselles. Visit its website [here](#)
- **shinyjs**: to create toggles, hide or show elements, etc. Visit its website [here](#)
- **shinyjqui**: to make plots, tables, panels, etc. resizable. Visit its website [here](#)
- **shinyFeedback**: create alerts on input widgets when values out of range are introduced. Visit its website [here](#)
- **shinythemes**: change the app appearance. Visit its website [here](#)

Shiny extensions: available on CRAN

shiny	Web Application Framework for R
shiny.semantic	Semantic UI Support for Shiny
shinyAce	Ace Editor Bindings for Shiny
shinyaframe	'WebVR' Data Visualizations with 'RStudio Shiny ' and 'Mozilla A-Frame'
shinybootstrap2	Bootstrap 2 Web Components for Use with Shiny
shinyBS	Twitter Bootstrap Components for Shiny
shinycssloaders	Add CSS Loading Animations to ' shiny ' Outputs
shinydashboard	Create Dashboards with ' Shiny '
shinyDND	Shiny Drag-n-Drop
shinyFeedback	Displays User Feedback Next to Shiny Inputs
shinyFiles	A Server-Side File System Viewer for Shiny
shinyHeatmaply	Deploy 'heatmaply' using ' shiny '
ShinyImage	Image Manipulation, with an Emphasis on Journaling
ShinyItemAnalysis	Test and Item Analysis via Shiny
shinyjquery	'jQuery UI' Interactions and Effects for Shiny
shinyjs	Easily Improve the User Experience of Your Shiny Apps in Seconds
shinyKCode	An Interactive Application for ODE Parameter Inference Using Gradient Matching
shinyLP	Bootstrap Landing Home Pages for Shiny Applications
shinymaterial	Implement Material Design in Shiny Applications
shinyRGL	Shiny Wrappers for RGL
shinyShortcut	Creates an Executable Shortcut for Shiny Applications
shinystan	Interactive Visual and Numerical Diagnostics and Posterior Analysis for Bayesian Models
ShinyTester	Functions to Minimize Bonehead Moves While Working with ' shiny '
shinythemes	Themes for Shiny
shinyTime	A Time Input Widget for Shiny
shinytoastr	Notifications from ' Shiny '

"homemade" examples



Inicio / Ayuda

Distribución según grupo de riesgo

Riesgos y NNT según percentiles

Percentiles y NNT según Riesgos

Introduce la palabra clave

Objetivos:

Este aplicativo permite comparar los percentiles de población correspondientes cualquier punto de corte en el nivel de riesgo a 10 años estimado con las ecuaciones para el riesgo coronario y cardiovascular más utilizadas en nuestro entorno. Para ello se aplica y compara la distribución de cada una de las ecuaciones en una muestra amplia y razonablemente representativa de la población Española [Ref1].

En pantalla se muestran distintas figuras que ilustran cada uno de los aspectos (percentiles, grupos de riesgo, etc.). El usuario puede fácilmente interactuar cambiando la población (hombres, mujeres, todos) y los puntos de corte en el nivel de riesgo, etc. y automáticamente los resultados se actualizan.

Esta herramienta facilita la elección de los puntos de corte más apropiados en cada ecuación para optimizar su sensibilidad y especificidad.

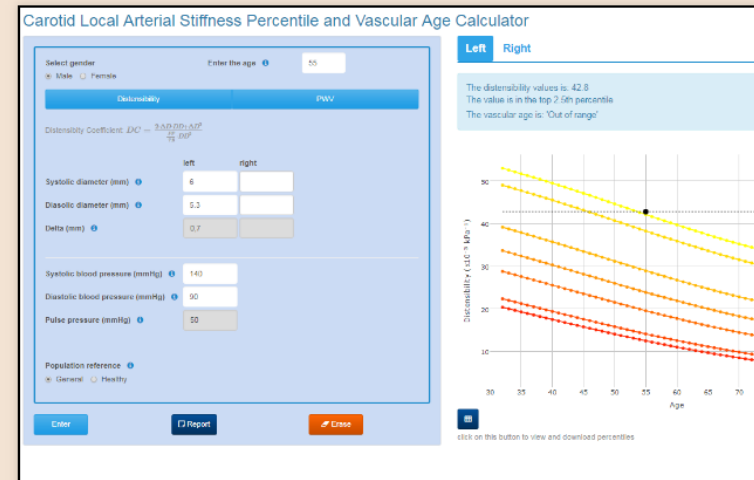
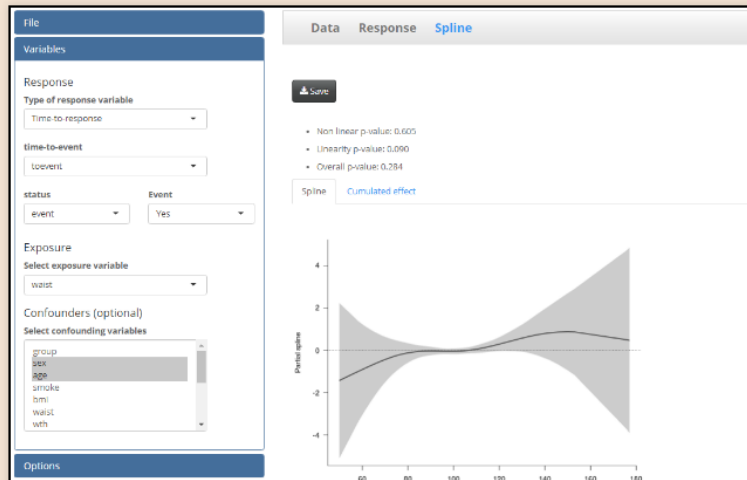
Población:

Los resultados de este aplicativo están calculados sobre la muestra razonablemente representativa de la población española (estudio FRESKO), sobre la derivó y validó una ecuación para el cálculo del riesgo coronario y que se presenta en este aplicativo como ecuación FRESKO [Ref1].

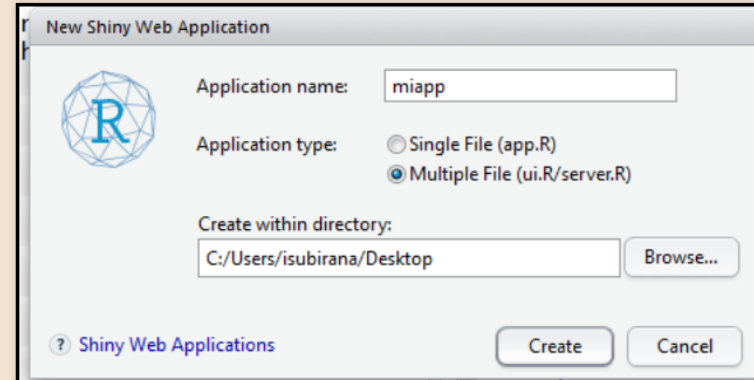
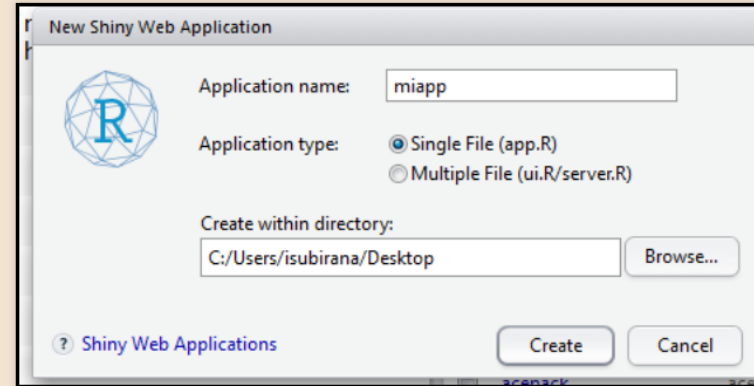
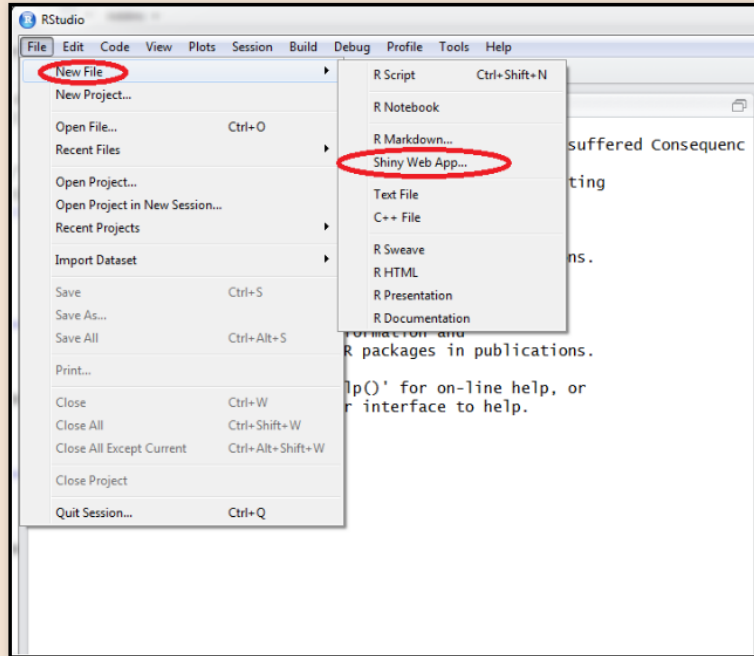
Los participantes fueron individuos sin antecedentes de enfermedad cardiovascular, de 35 a 80 años sin tratamiento para el colesterol. En esta aplicación los participantes del estudio FRESKO de 35 a 74 años.

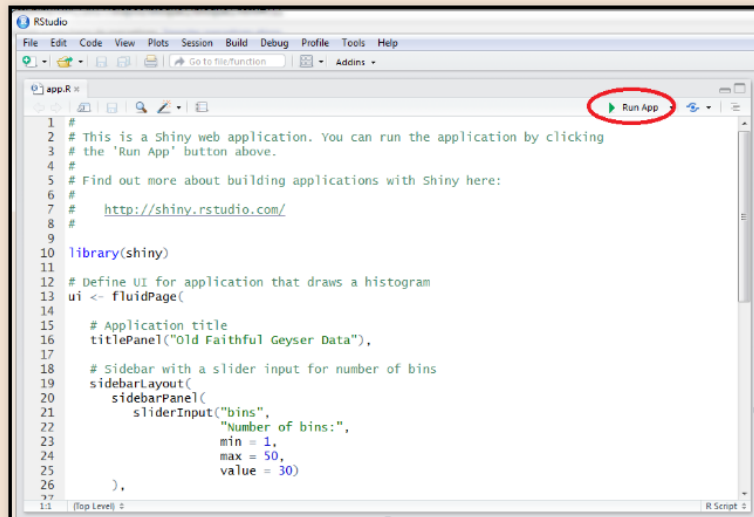
Ecuaciones:

Las ecuaciones que se comparan en este aplicativo son:



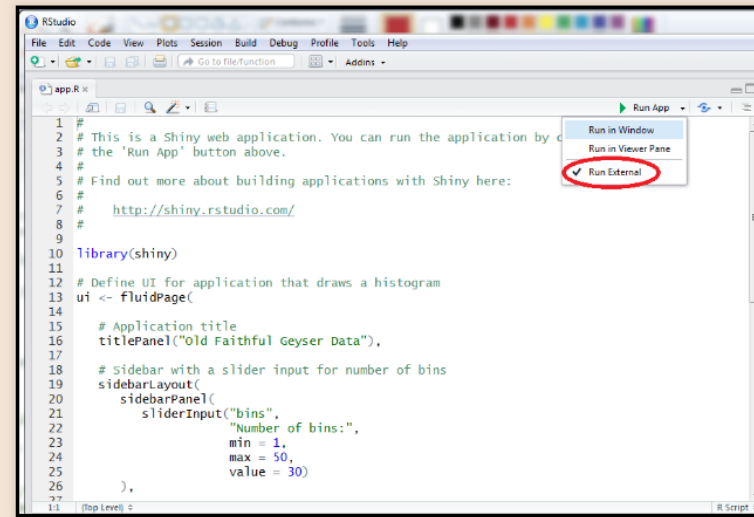
My first app (RStudio)





The screenshot shows the RStudio interface with a script editor containing R code for a Shiny application. The 'Run App' button in the top right corner of the script editor is circled in red.

```
1 #  
2 # This is a Shiny web application. You can run the application by clicking  
3 # the 'Run App' button above.  
4 #  
5 # Find out more about building applications with Shiny here:  
6 #  
7 # http://shiny.rstudio.com/  
8 #  
9  
10 library(shiny)  
11  
12 # Define UI for application that draws a histogram  
13 ui <- fluidPage(  
14  
15   # Application title  
16   titlePanel("Old Faithful Geyser Data"),  
17  
18   # Sidebar with a slider input for number of bins  
19   sidebarLayout(  
20     sidebarPanel(  
21       sliderInput("bins",  
22         "Number of bins:",  
23         min = 1,  
24         max = 50,  
25         value = 30)  
26     ),  
27  
28   )  
29 )
```



The screenshot shows the RStudio interface with the same script editor. The 'Run App' button has been clicked, and a dropdown menu is visible. The 'Run External' option in the dropdown menu is circled in red.

```
1 #  
2 # This is a Shiny web application. You can run the application by clicking  
3 # the 'Run App' button above.  
4 #  
5 # Find out more about building applications with Shiny here:  
6 #  
7 # http://shiny.rstudio.com/  
8 #  
9  
10 library(shiny)  
11  
12 # Define UI for application that draws a histogram  
13 ui <- fluidPage(  
14  
15   # Application title  
16   titlePanel("Old Faithful Geyser Data"),  
17  
18   # Sidebar with a slider input for number of bins  
19   sidebarLayout(  
20     sidebarPanel(  
21       sliderInput("bins",  
22         "Number of bins:",  
23         min = 1,  
24         max = 50,  
25         value = 30)  
26     ),  
27  
28   )  
29 )
```