Shiny

Sesión II: Extensiones a Shiny

Hèctor Perpiñán Fabuel - Unitat de Bioestadística, IRBLLEIDA

07 de Junio, 2017

Contenido

1. Customizar apariencia
    - HTML
    - CSS
2. Extensiones a Shiny (mediante paquetes)
3. Compartir las apps por internet
    - local
    - global
4. MapEs (una app desarrollada por FISABIO - DG Salud Pública)

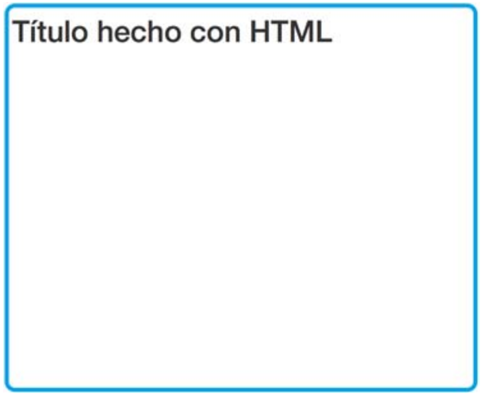
1. Customizar apariencia

HTML & CSS

Comandos HTML

Shiny acepta **código HTML**

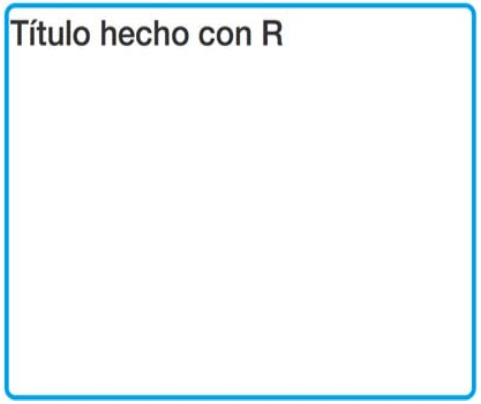
```
ui <- fluidPage(
  HTML("<h1>Título hecho con HTML</h1>")
)

server <- function(input, output) {}

shinyApp(ui, server)
```



Título hecho con HTML

R permite añadir contenido a una página con las funciones **tags**

```
ui <- fluidPage(
  tags$h1("Título hecho con R")
)

server <- function(input, output) {}

shinyApp(ui, server)
```

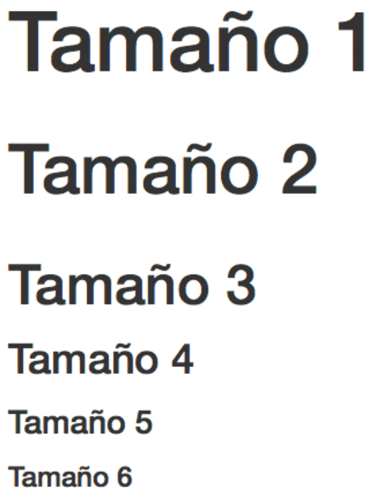


Título hecho con R

tags: **h1() - h6()**

Tamaño del texto introducido

```
ui <- fluidPage(
  h1("Tamaño 1"),
  h2("Tamaño 2"),
  h3("Tamaño 3"),
  h4("Tamaño 4"),
  h5("Tamaño 5"),
  h6("Tamaño 6")
)

server <- function(input, output) {}

shinyApp(ui, server)
```



Tamaño 1

Tamaño 2

Tamaño 3

Tamaño 4

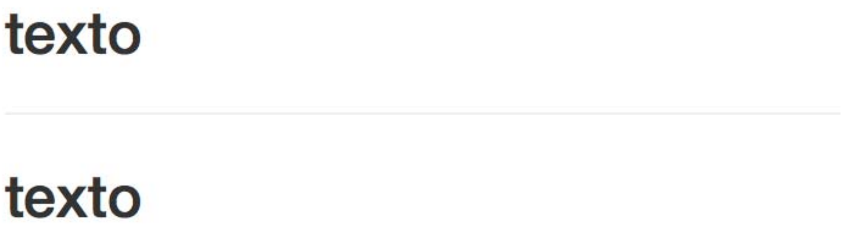
Tamaño 5

Tamaño 6

tags: **hr()**

Línea horizontal

```
ui <- fluidPage(
  h3("texto"),
  hr(),
  h3("texto")
)

server <- function(input, output) {}

shinyApp(ui, server)
```



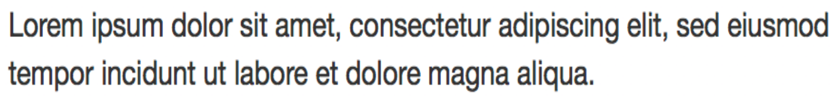
texto

texto

tags: "**Texto**"

Texto normal. El texto plano sin modificadores no necesita .

```r
ui <- fluidPage(
  "Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed
)

server <- function(input, output) {}

shinyApp(ui, server)
```

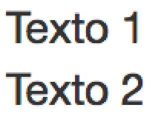


Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua.

tags: **br()**

Salto de línea

```
ui <- fluidPage(
  "Texto 1",
  br(),
  "Texto 2"
)

server <- function(input, output) {}

shinyApp(ui, server)
```



Texto 1
Texto 2

tags: **p()**

Párrafo

```
ui <- fluidPage(
  p("Párrafo 1"),
  p("Párrafo 2")
)

server <- function(input, output) {}

shinyApp(ui, server)
```

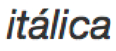
Párrafo 1

Párrafo 2

tags: **em()**

Letra en **itálica/cursiva**

```
ui <- fluidPage(
  em("itálica")
)

server <- function(input, output) {}

shinyApp(ui, server)
```

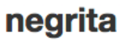


itálica

tags: **strong()**

Letra en **negrita**

```
ui <- fluidPage(
  strong("negrita")
)

server <- function(input, output) {}

shinyApp(ui, server)
```



negrita

tags: **code()**

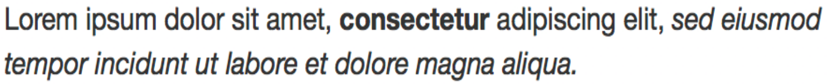
Texto monoespaciado. Típicamente se emplea este comando para introducir **código**

```
ui <- fluidPage(
  code("código")
)

server <- function(input, output) {}

shinyApp(ui, server)
```

`código`

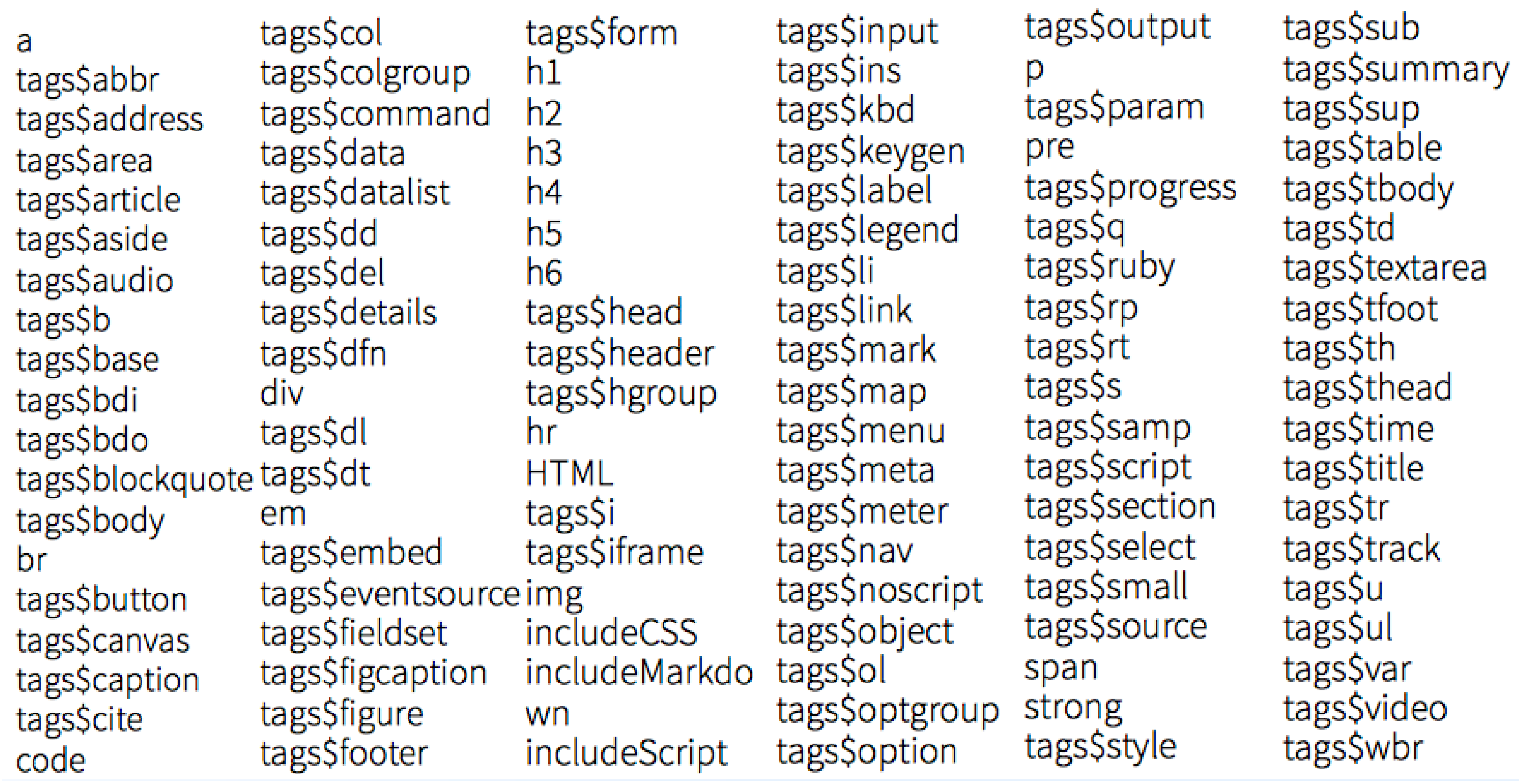
Los se pueden **anidar** unos dentro de otros.

```
ui <- fluidPage(
  p("Lorem ipsum dolor sit amet, ", strong("consectetur"), " ad
)

server <- function(input, output) {}

shinyApp(ui, server)
```



Lorem ipsum dolor sit amet, **consectetur** adipiscing elit, *sed eiusmod tempor incidunt ut labore et dolore magna aliqua.*

tags que soporta Shiny



a
tags$abbr
tags$address
tags$area
tags$article
tags$aside
tags$audio
tags$b
tags$base
tags$bdi
tags$bdo
tags$blockquote
tags$body
br
tags$button
tags$canvas
tags$caption
tags$cite
code
tags$col
tags$colgroup
tags$command
tags$data
tags$datalist
tags$dd
tags$del
tags$details
tags$dfn
div
tags$dl
tags$dt
em
tags$embed
tags$eventsource
tags$fieldset
tags$figcaption
tags$figure
tags$footer
tags$form
h1
h2
h3
h4
h5
h6
tags$head
tags$header
tags$hgroup
hr
HTML
tags$i
tags$iframe
img
includeCSS
includeMarkdown
includeScript
tags$input
tags$ins
tags$kbd
tags$keygen
tags$label
tags$legend
tags$li
tags$link
tags$mark
tags$map
tags$menu
tags$meta
tags$meter
tags$nav
tags$noscript
tags$object
tags$ol
tags$optgroup
tags$option
tags$output
p
tags$param
pre
tags$progress
tags$q
tags$ruby
tags$rp
tags$rt
tags$s
tags$samp
tags$script
tags$section
tags$select
tags$small
tags$source
span
strong
tags$style
tags$sub
tags$summary
tags$sup
tags$table
tags$tbody
tags$td
tags$textarea
tags$tfoot
tags$th
tags$thead
tags$time
tags$title
tags$tr
tags$track
tags$u
tags$ul
tags$var
tags$video
tags$wbr

Insertar un CSS

Las hojas de estilo en cascada (CSS) son un marco para personalizar la apariencia de elementos en una página web.

```r
ui <- fluidPage(
  theme = "bootstrap.css",
  sidebarLayout(
    sidebarPanel(),
    mainPanel()
  )
)

server <- function(input, output) {}

shinyApp(ui, server)
```

```r
ui <- fluidPage(
  includeCSS("bootstrap.css"),
  sidebarLayout(
    sidebarPanel(),
    mainPanel()
  )
)
```

2. Extensiones a Shiny (mediante paquetes)

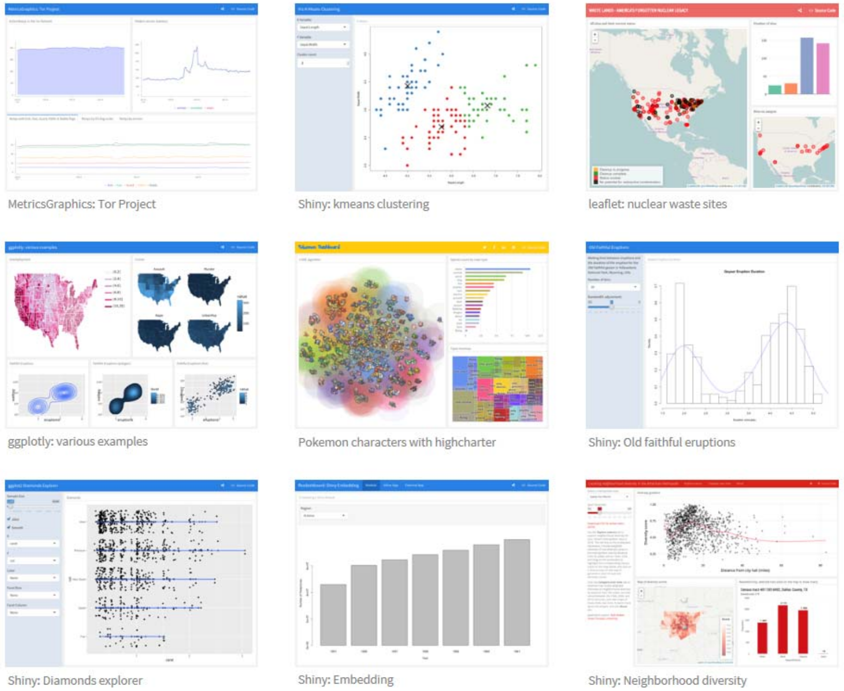
2. Extensiones a Shiny (mediante paquetes)

- Extensiones de formato
  - **flexdashboard**: Tableros interactivos fáciles para R (RMarkdown)
  - **shinythemes**: Temas CSS listos para usar con Shiny
  - **shinydashboard**: Tableros para Shiny
  - **shinyjqui**: Interacciones y efectos de animación para Shiny
- Extensiones para cálculos/gráficos interactivos
  - **htmlwidgets**: Un marco para embeber visualizaciones de JavaScript en R

flexdashboard (http://rmarkdown.rstudio.com/flexdashboard/index.htm

- Formalmente es un **RMarkdown** (documento interactivo) con elementos de **Shiny**
- Muy buena combinación con **Github**
- Redes sociales (Twitter, Facebook, Google+, LinkedIn and Pinterest)
- Posibilidad de incrustar código

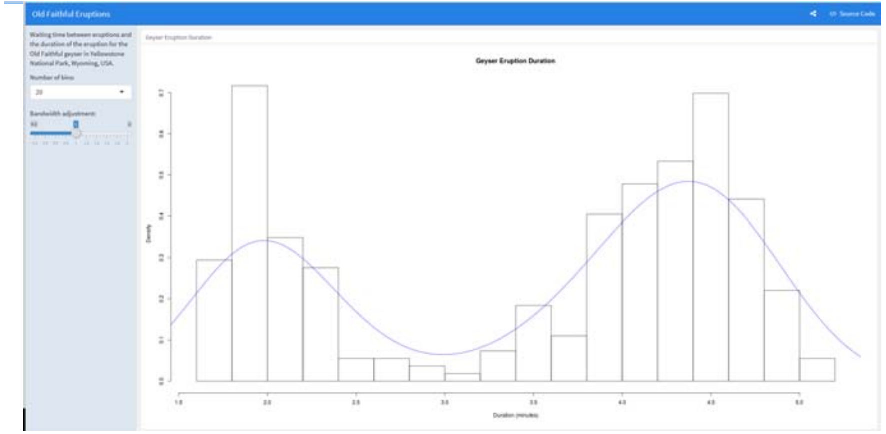
(https://beta.rstudioconnect.com/jjallaire/l showcase-storyboard/htmlwidgets-showcase-storyboard.html)


MetricsGraphics: Tor Project
Shiny: kmeans clustering
leaflet: nuclear waste sites
ggplotly: various examples
Pokemon characters with highcharter
Shiny: Old faithful eruptions
Shiny: Diamonds explorer
Shiny: Embedding
Shiny: Neighborhood diversity

(https://beta.rstudioconnect.com/jjallaire/htm showcase-storyboard/htmlwidgets-showcase-storyboard.html)

flexdashboard (http://rmarkdown.rstudio.com/flexdashboard/index.htm

(https://beta.rstudioconnect.com/jjallaire/l showcase-storyboard/htmlwidgets-showcase-storyboard.html) (https://beta.rstudioconnect.com/jjallaire/l showcase-storyboard/htmlwidgets-showcase-storyboard.html)



Old Faithful Eruptions

Waiting time between eruptions and the duration of the eruption for the Old Faithful geyser in Yellowstone National Park, Wyoming, USA.

Number of bins: 20

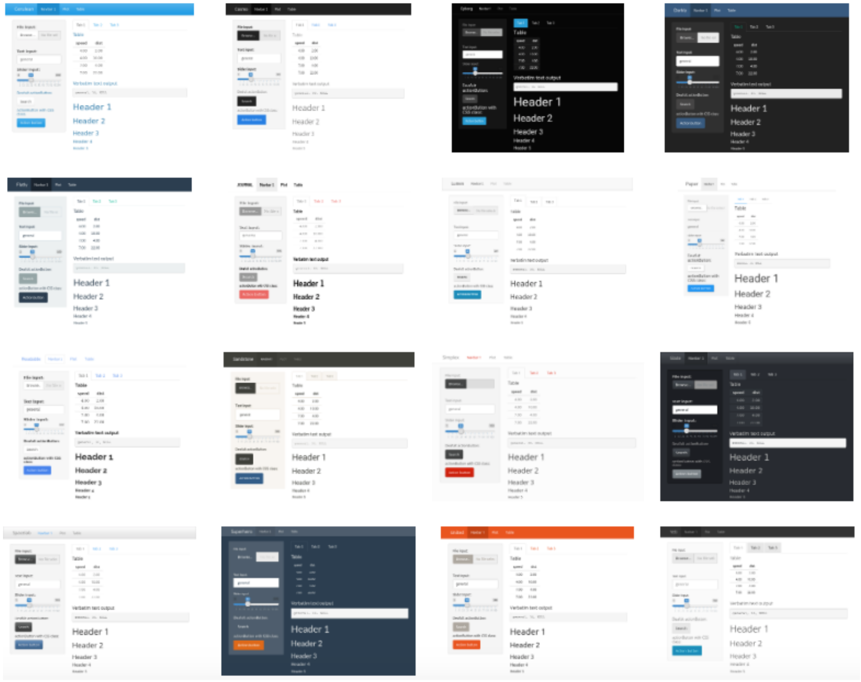
Bandwidth adjustment:

Geyser Eruption Duration

(https://beta.rstudioconnect.com/jjallaire/l showcase-storyboard/htmlwidgets-showcase-storyboard.html)

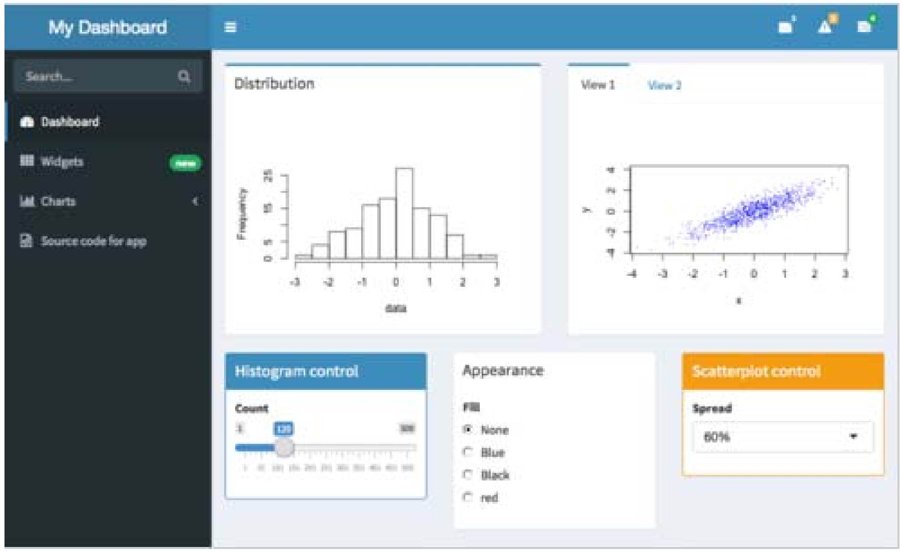
(https://beta.rstudioconnect.com/jjallaire/shir eruptions/)

shinythemes (https://rstudio.github.io/shinythemes/)



```
library(shiny); library(shinythemes)


shinyApp(
  ui = fluidPage(theme = shinytheme("cyborg"),
    ...
  ),
  server = function(input, output) { }
)
```

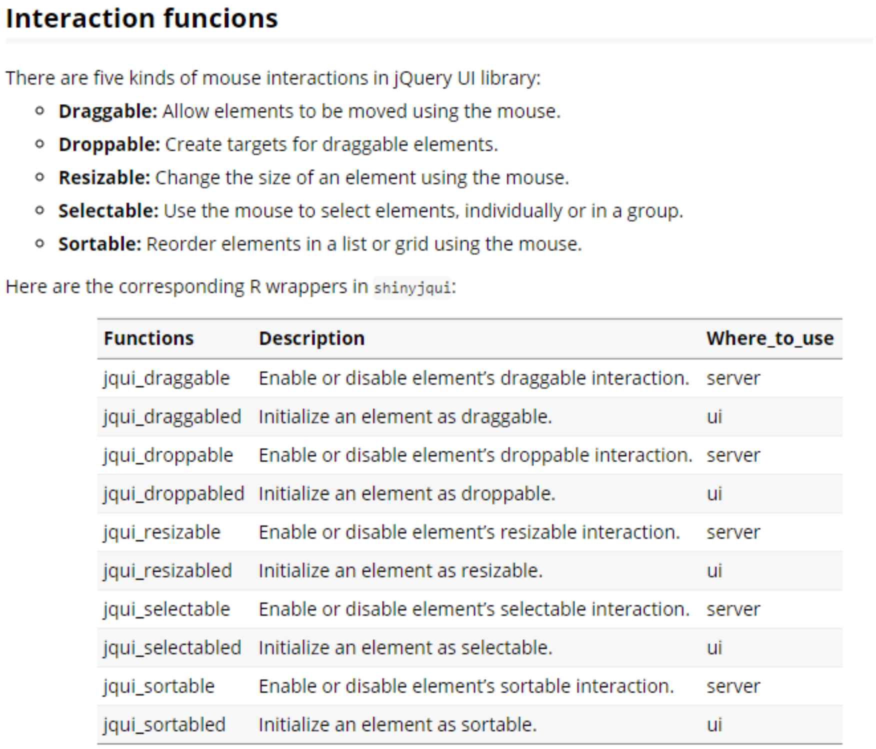
shinydashboard (https://rstudio.github.io/shinydashboard/index.html)



```
library(shiny); library(shinydashboard)
```

shinyjqui (https://yang-tang.github.io/shinyjqui/)

- Acciones de ratón (https://yang-tang.github.io/shinyjqui/articles/introdu interactions)
- Efectos de animación (https://yang-tang.github.io/shinyjqui/articles/introdu effects)
- Animación de 'classes' (https://yang-tang.github.io/shinyjqui/articles/introdu animation)



Interaction funcions

There are five kinds of mouse interactions in jQuery UI library:

- **Draggable:** Allow elements to be moved using the mouse.
- **Droppable:** Create targets for draggable elements.
- **Resizable:** Change the size of an element using the mouse.
- **Selectable:** Use the mouse to select elements, individually or in a group.
- **Sortable:** Reorder elements in a list or grid using the mouse.

Here are the corresponding R wrappers in `shinyjqui`:

Functions	Description	Where_to_use
jqui_draggable	Enable or disable element's draggable interaction.	server
jqui_draggabled	Initialize an element as draggable.	ui
jqui_droppable	Enable or disable element's droppable interaction.	server
jqui_droppabled	Initialize an element as droppable.	ui
jqui_resizable	Enable or disable element's resizable interaction.	server
jqui_resizabled	Initialize an element as resizable.	ui
jqui_selectable	Enable or disable element's selectable interaction.	server
jqui_selectabled	Initialize an element as selectable.	ui
jqui_sortable	Enable or disable element's sortable interaction.	server
jqui_sortabled	Initialize an element as sortable.	ui

htmlwidgets (http://www.htmlwidgets.org/)

87 widgets registrados actualmente ( ):

- : Mapeado geoespacial interactivo
- : Creación de gráficos interactivos
- : Graficado de series temporales
- : Visualización gráfica de datos con D3 (https://d3js.org/)
- : Visualización de datos tabulares
- **d3heatmap**: Mapas de calor
- : Grafos y diagramas de flujo
- : Renderiza escenas creadas con rgl (http://rgl.neoscientists.org/about.shtml)

Leaflet (https://rstudio.github.io/leaflet/): mapas interactivos

- Llamadas en Shiny: y
- Tiles (http://leaflet-extras.github.io/leaflet-providers/preview/index.html)

```
library(leaflet)
leaflet() %>% addTiles() %>% addMarkers(lng=-0.3531, lat=39.4815, popup="FISABIO")
```

+

−

Plotly (https://plot.ly/r/): Navaja suiza de gráficos interactivos

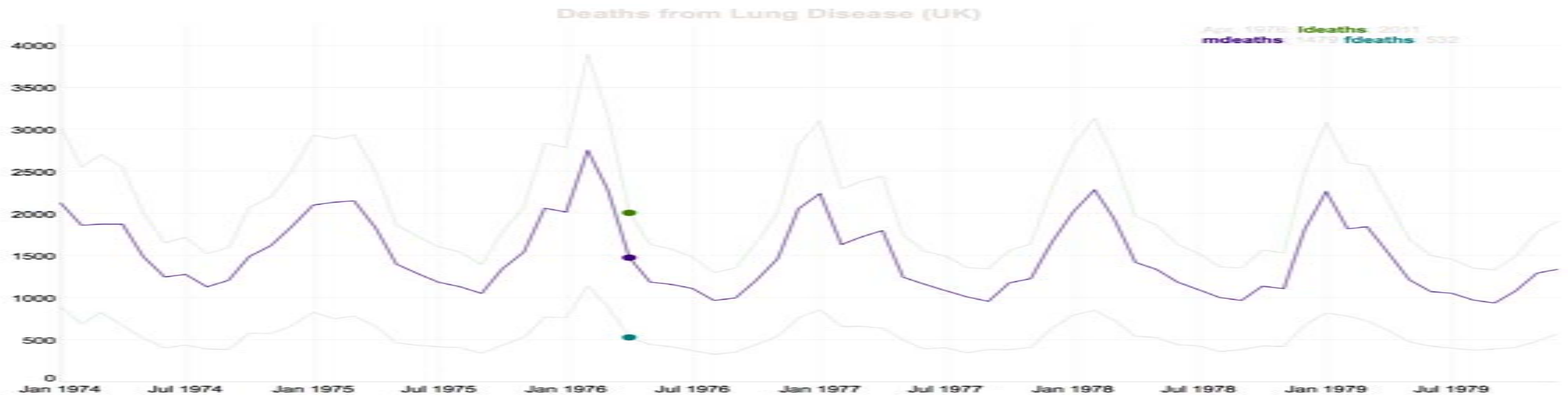
- Llamadas en Shiny: y

```
library(ggplot2, plotly)
p <- ggplot(data = diamonds, aes(x = cut, fill = clarity)) + geom_bar(position = "dodge")
ggplotly(p)
```

dygraphs (https://rstudio.github.io/dygraphs/): Gráficos interactivos para series temporales

- Llamadas en Shiny: y

```
library(dygraphs)
lungDeaths <- cbind(ldeaths, mdeaths, fdeaths)
dygraph(lungDeaths, main = "Deaths from Lung Disease (UK)") %>%
  dyHighlight(highlightCircleSize = 5,
              highlightSeriesBackgroundAlpha = 0.2,
              hideOnMouseOut = FALSE)
```



networkD3 (http://christophergandrud.github.io/networkD3/): Grafos interactivos

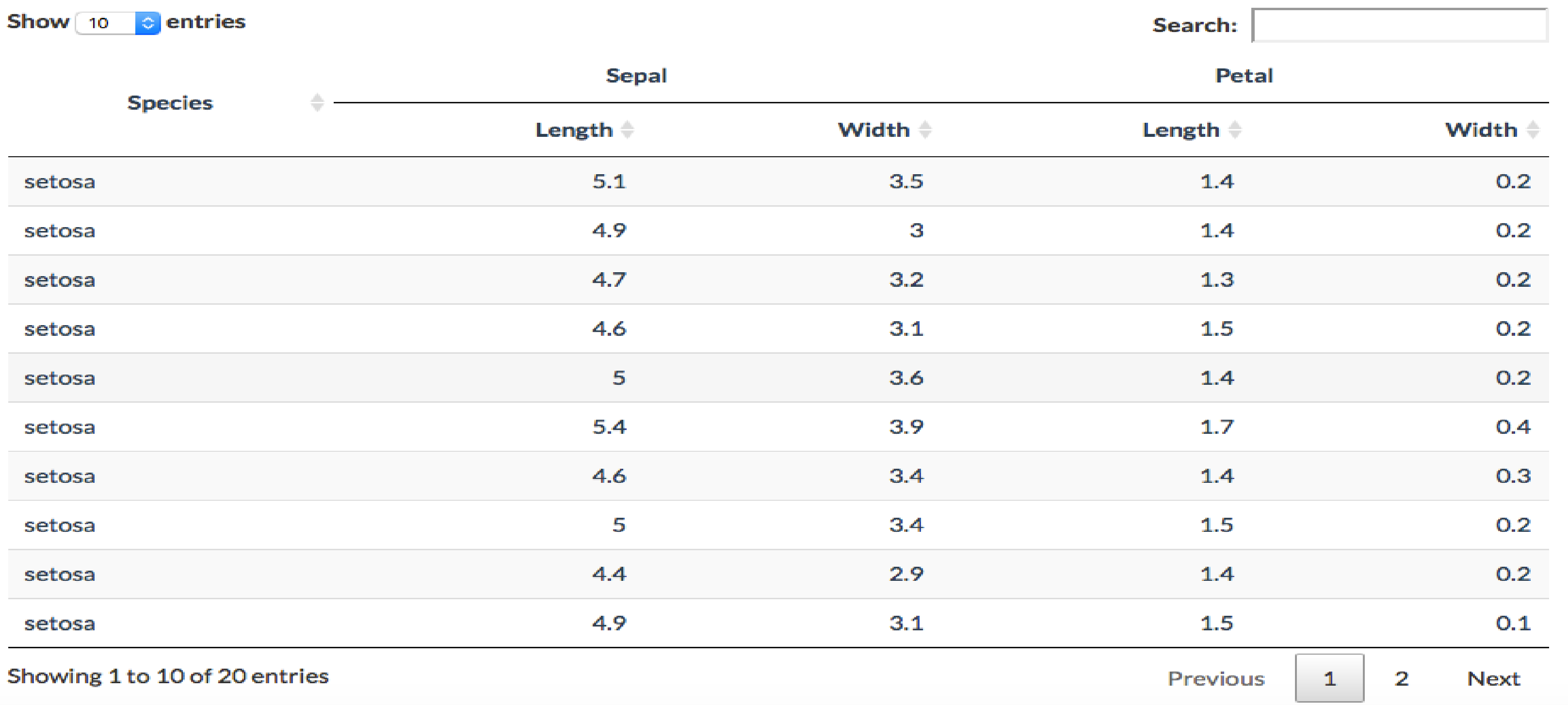
- Llamadas en Shiny: y

```
library(networkD3)

data(MisLinks, MisNodes)

forceNetwork(Links = MisLinks, Nodes = MisNodes, Source = "source", Target = "target", Value = "value", NodeID = "name", Group = "grou
```

DataTable (http://rstudio.github.io/DT/): Tablas dinámicas (paquete DT)

- Llamadas en Shiny: y

```
DT::datatable(iris, options = list(pageLength = 3, dom = "pt", language = list(url = '//cdn.datatables.net/plug-ins/1.10.11/i18n/Spani
```



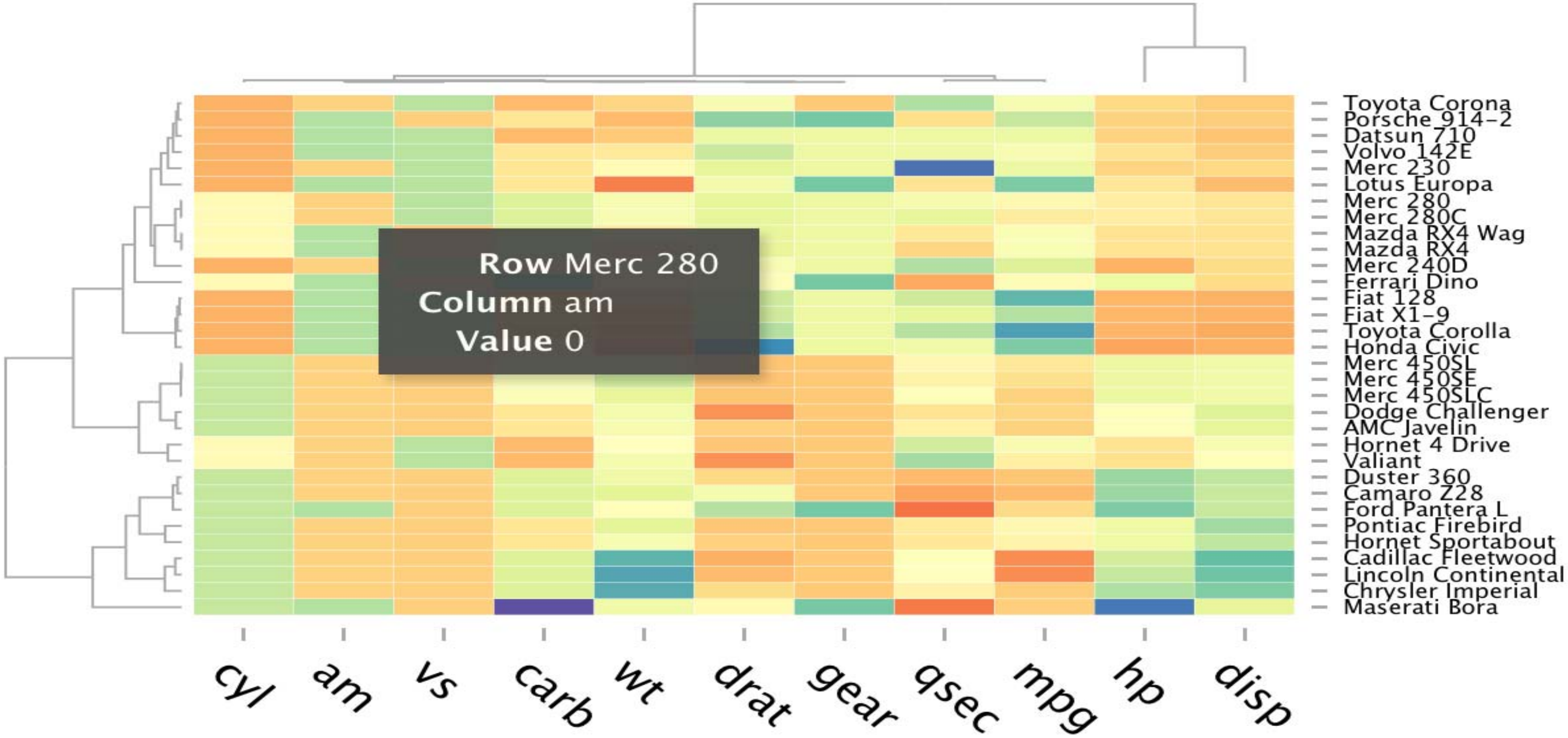
Species	Sepal		Petal	
	Length	Width	Length	Width
setosa	5.1	3.5	1.4	0.2
setosa	4.9	3	1.4	0.2
setosa	4.7	3.2	1.3	0.2
setosa	4.6	3.1	1.5	0.2
setosa	5	3.6	1.4	0.2
setosa	5.4	3.9	1.7	0.4
setosa	4.6	3.4	1.4	0.3
setosa	5	3.4	1.5	0.2
setosa	4.4	2.9	1.4	0.2
setosa	4.9	3.1	1.5	0.1

Showing 1 to 10 of 20 entries

d3heatmap (https://github.com/rstudio/d3heatmap)

- Llamadas en Shiny: y

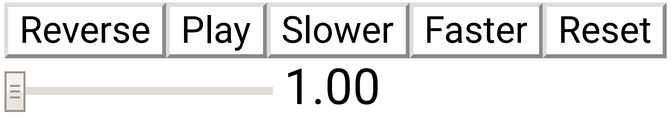
```
library(d3heatmap)

d3heatmap(mtcars, scale = "column", colors = "Spectral")
```



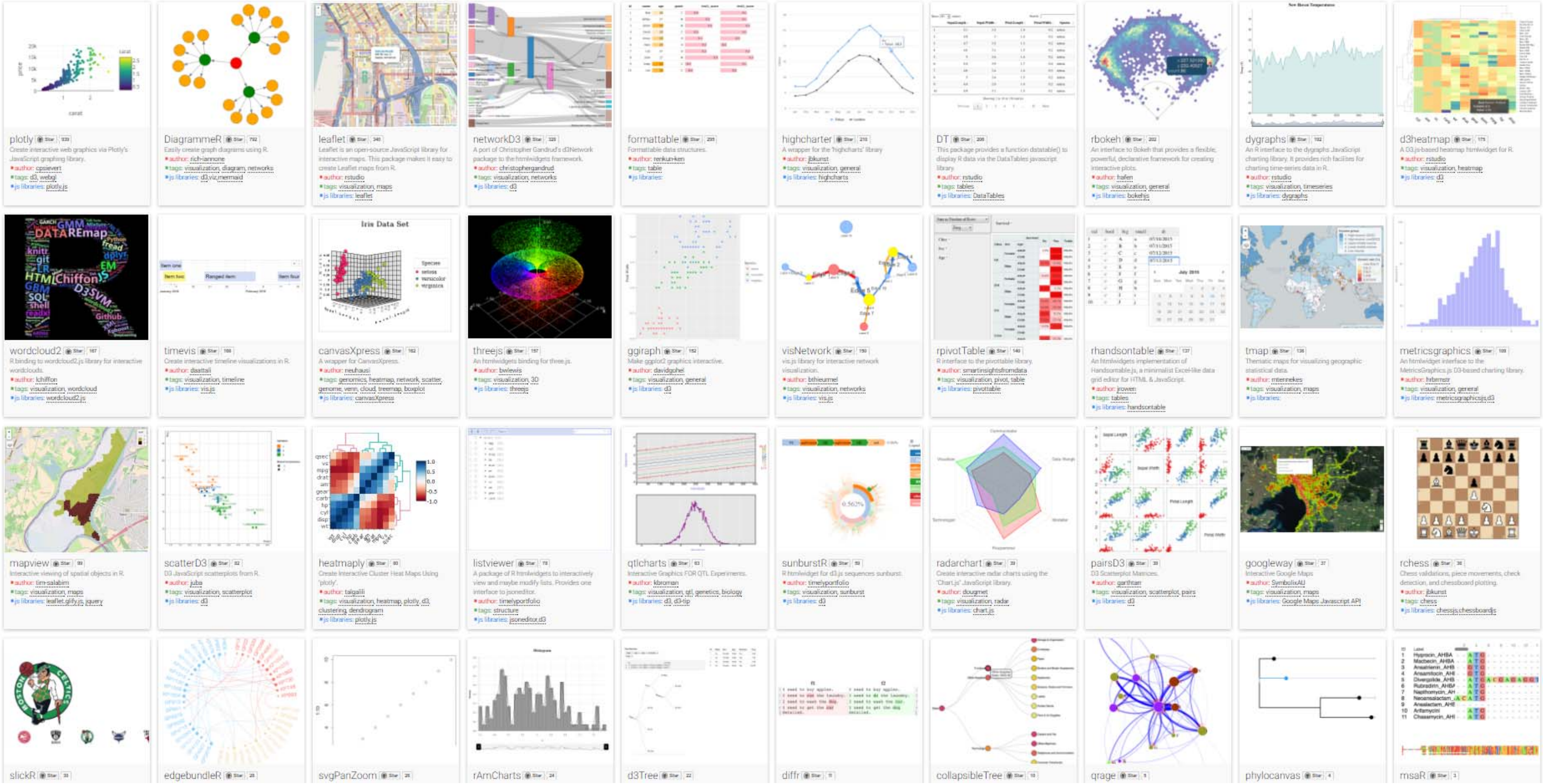
DiagrammeR (http://rich-iannone.github.io/DiagrammeR/): Grafos y diagramas de flujo

rglwidget (http://www.htmlwidgets.org/showcase_rglwidget.html)

```
library(rgl); library(rglwidget); library(htmltools)

theta <- seq(0, 6*pi, len=100)
xyz <- cbind(sin(theta), cos(theta), theta)
lineid <- plot3d(xyz, type="l", alpha = 1:0, lwd = 5, col = "bl

browsable(tagList(
  rglwidget(elementId = "example", width = 500, height = 400,
            controllers = "player"),
  playwidget("example",
             ageControl(births = theta, ages = c(0, 0, 1),
                        objids = lineid, alpha = c(0, 1, 0)),
                        start = 1, stop = 6*pi, step = 0.1,
                        rate = 6,elementId = "player")))
```



y muchos más en http://gallery.htmlwidgets.org/ (http://gallery.htmlwidgets.org/)



plotly Star 939
Create interactive web graphics via Plotly's JavaScript graphing library.
- author: cpsievert
- tags: d3, webgl
- js libraries: plotly.js

DiagrammeR Star 792
Easily create graph diagrams using R.
- author: rich-iannone
- tags: visualization, diagram, networks
- js libraries: d3,viz,mermaid

leaflet Star 340
Leaflet is an open-source JavaScript library for interactive maps. This package makes it easy to create Leaflet maps from R.
- author: rstudio
- tags: visualization, maps
- js libraries: leaflet

networkD3 Star 328
A port of Christopher Gandrud's d3Network package to the htmlwidgets framework.
- author: christophergandrud
- tags: visualization, networks
- js libraries: d3

formattable Star 299
Formattable data structures.
- author: renkun-ken
- tags: table
- js libraries:

highcharter Star 210
A wrapper for the 'highcharts' library
- author: jbkunst
- tags: visualization, general
- js libraries: highcharts

DT Star 206
This package provides a function datatable() to display R data via the DataTables javascript library.
- author: rstudio
- tags: tables
- js libraries: DataTables

rbokeh Star 202
An interface to Bokeh that provides a flexible, powerful, declarative framework for creating interactive plots.
- author: hafen
- tags: visualization, general
- js libraries: bokehjs

dygraphs Star 192
An R interface to the dygraphs JavaScript charting library. It provides rich facilites for charting time-series data in R.
- author: rstudio
- tags: visualization, timeseries
- js libraries: dygraphs

d3heatmap Star 175
A D3.js-based heatmap htmlwidget for R.
- author: rstudio
- tags: visualization, heatmap
- js libraries: d3

wordcloud2 Star 167
R binding to wordcloud2.js library for interactive wordclouds.
- author: lchiffon
- tags: visualization, wordcloud
- js libraries: wordcloud2.js

timevis Star 166
Create interactive timeline visualizations in R.
- author: daattali
- tags: visualization, timeline
- js libraries: vis.js

canvasXpress Star 162
A wrapper for CanvasXpress.
- author: neuhausi
- tags: genomics, heatmap, network, scatter, genome, venn, cloud, treemap, boxplot
- js libraries: canvasXpress

threejs Star 157
An htmlwidgets binding for three.js.
- author: bwlewis
- tags: visualization, 3D
- js libraries: threejs

ggiraph Star 152
Make ggplot2 graphics interactive.
- author: davidgohel
- tags: visualization, general
- js libraries: d3

visNetwork Star 150
vis.js library for interactive network visualization.
- author: bthieurmel
- tags: visualization, networks
- js libraries: vis.js

rpivotTable Star 140
R interface to the pivottable library.
- author: smartinsightsfromdata
- tags: visualization, pivot, table
- js libraries: pivottable

rhandsontable Star 137
An htmlwidgets implementation of Handsontable.js, a minimalist Excel-like data grid editor for HTML & JavaScript.
- author: jrowen
- tags: tables
- js libraries: handsontable

tmap Star 136
Thematic maps for visualizing geographic statistical data.
- author: mtennekes
- tags: visualization, maps
- js libraries:

metricsgraphics Star 109
An htmlwidget interface to the MetricsGraphics.js D3-based charting library.
- author: hrbrmstr
- tags: visualization, general
- js libraries: metricsgraphicsjs,d3

mapview Star 99
Interactive viewing of spatial objects in R.
- author: tim-salabim
- tags: visualization, maps
- js libraries: leaflet.glify.js, jquery

scatterD3 Star 92
D3 JavaScript scatterplots from R.
- author: juba
- tags: visualization, scatterplot
- js libraries: d3

heatmaply Star 90
Create Interactive Cluster Heat Maps Using 'plotly'.
- author: talgalili
- tags: visualization, heatmap, plotly, d3, clustering, dendrogram
- js libraries: plotly.js

listviewer Star 79
A package of R htmlwidgets to interactively view and maybe modify lists. Provides one interface to jsoneditor.
- author: timelyportfolio
- tags: structure
- js libraries: jsoneditor,d3

qtlcharts Star 63
Interactive Graphics FOR QTL Experiments.
- author: kbroman
- tags: visualization, qtl, genetics, biology
- js libraries: d3, d3-tip

sunburstR Star 59
R htmlwidget for d3.js sequences sunburst.
- author: timelyportfolio
- tags: visualization, sunburst
- js libraries: d3

radarchart Star 39
Create interactive radar charts using the 'Chart.js' JavaScript library.
- author: dougmet
- tags: visualization, radar
- js libraries: chart.js

pairsD3 Star 39
D3 Scatterplot Matrices.
- author: garthtarr
- tags: visualization, scatterplot, pairs
- js libraries: d3

googleway Star 37
Interactive Google Maps
- author: SymbolixAU
- tags: visualization, maps
- js libraries: Google Maps Javascript API

rchess Star 36
Chess validations, piece movements, check detection, and chessboard plotting.
- author: jbkunst
- tags: chess
- js libraries: chessjs,chessboardjs

slickR Star 30

edgebundleR Star 26

svgPanZoom Star 26

rAmCharts Star 24

d3Tree Star 22

diffr Star 11

collapsibleTree Star 10

qrage Star 5

phylocanvas Star 4

msaR Star 3

3. Compartir las apps creadas con Shiny

3. Compartir las apps creadas con Shiny…

de forma **local**, con alguien que tiene R en su ordenador.

- , o

de forma **global**, con todo el mundo (sin necesidad de tener R).

- shinyapps.io
- Shiny Server
- RStudio Connect

Compartir apps Shiny

local

runUrl()

- Comprimir la carpeta de la app en un zip (https://es.wikipedia.org/wiki/Formato_de_compresi%C3%B3n_ZIP) y enlazar el archivo en una web

```
runUrl( "<link a la web>")
```

runGitHub()

- Alojar tu app en tu repositorio libre de GitHub (https://github.com/)

```
runGitHub( "<nombre de tu repositorio>", "<tu nombre de usuario>")
```

runGist()

- Alojar tu app en tu repositorio libre de GitHub (https://github.com/), manteniendo tu anonimato con Gist (https://gist.github.com/)

- Subir los archivos a Gist (https://gist.github.com/)

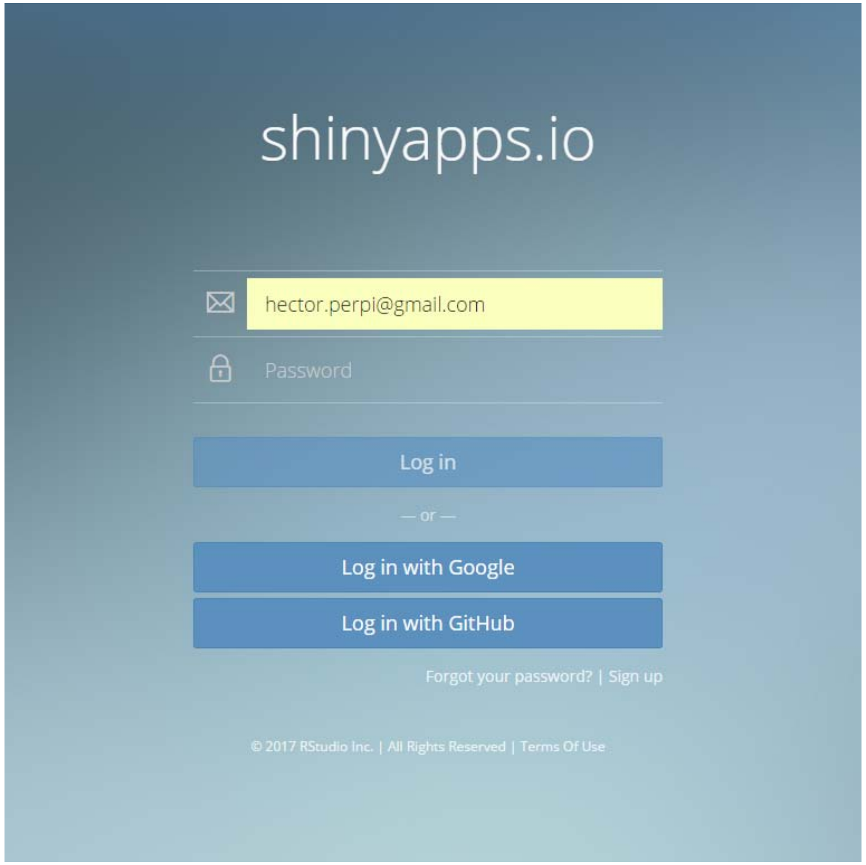
1. Gist (https://gist.github.com/) nos dará una Url
2. Los números finales de la Url son el **código gist**

```
runGist("código gist")
```

Compartir apps Shiny

global

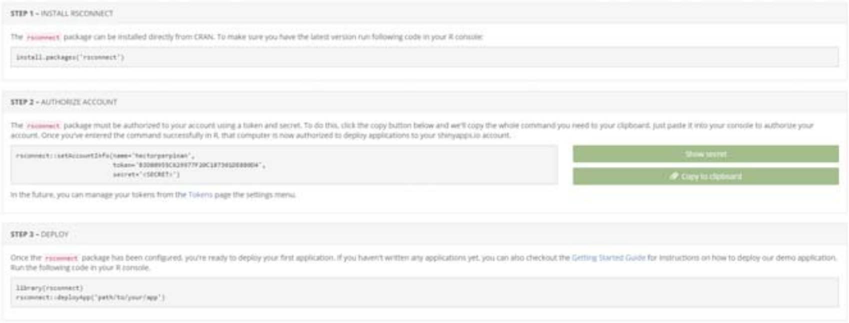
shinyapps.io (https://www.shinyapps.io/)



hector.perpi@gmail.com

Password

© 2017 RStudio Inc. | All Rights Reserved | Terms Of Use



STEP 1 - INSTALL RSCONNECT

The `rsconnect` package can be installed directly from CRAN. To make sure you have the latest version run following code in your R console:

```
install.packages('rsconnect')
```

STEP 2 - AUTHORIZE ACCOUNT

The `rsconnect` package must be authorized to your account using a token and secret. To do this, click the copy button below and we'll copy the whole command you need to your clipboard. Just paste it into your console to authorize your account. Once you've entered the command successfully in R, that computer is now authorized to deploy applications to your shinyapps.io account.

```
rsconnect::setAccountInfo(name='hectorperpinan',
                          token='83D80955C629977F20C187301DE880D4',
                          secret='<SECRET>')
```

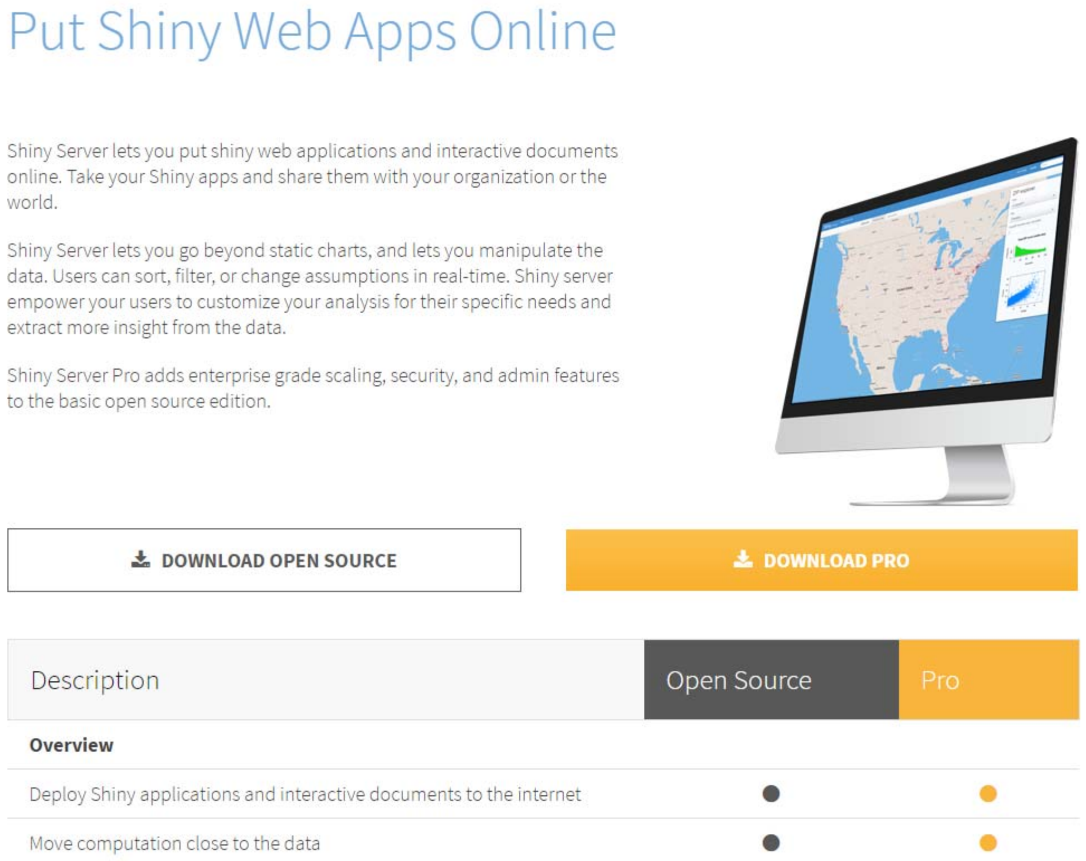
In the future, you can manage your tokens from the Tokens page the settings menu.

STEP 3 - DEPLOY

Once the `rsconnect` package has been configured, you're ready to deploy your first application. If you haven't written any applications yet, you can also checkout the Getting Started Guide for instructions on how to deploy our demo application. Run the following code in your R console.

```
library(rsconnect)
rsconnect::deployApp('path/to/your/app')
```

Shiny Server (https://www.rstudio.com/products/shiny/shiny-server/)



Put Shiny Web Apps Online

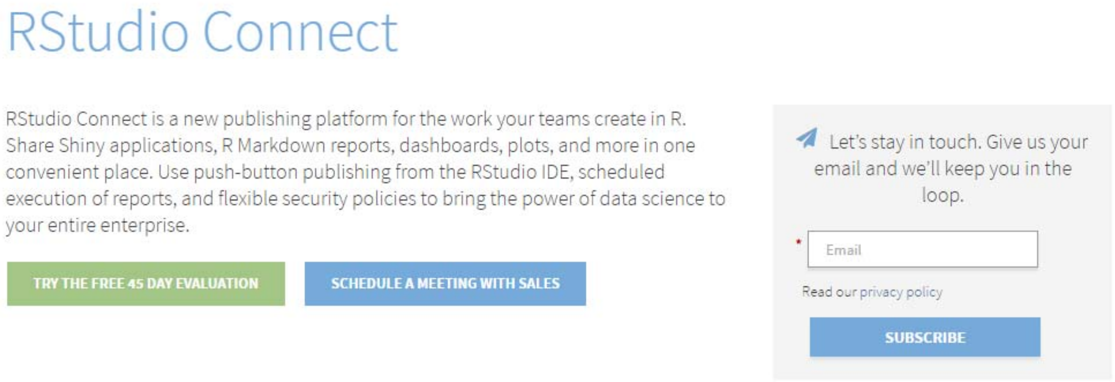
Shiny Server lets you put shiny web applications and interactive documents online. Take your Shiny apps and share them with your organization or the world.

Shiny Server lets you go beyond static charts, and lets you manipulate the data. Users can sort, filter, or change assumptions in real-time. Shiny server empower your users to customize your analysis for their specific needs and extract more insight from the data.

Shiny Server Pro adds enterprise grade scaling, security, and admin features to the basic open source edition.

Description	Open Source	Pro
Overview		
Deploy Shiny applications and interactive documents to the internet	●	●
Move computation close to the data	●	●

RStudio Connect (https://www.rstudio.com/products/connect/)



RStudio Connect

RStudio Connect is a new publishing platform for the work your teams create in R. Share Shiny applications, R Markdown reports, dashboards, plots, and more in one convenient place. Use push-button publishing from the RStudio IDE, scheduled execution of reports, and flexible security policies to bring the power of data science to your entire enterprise.

TRY THE FREE 45 DAY EVALUATION

SCHEDULE A MEETING WITH SALES

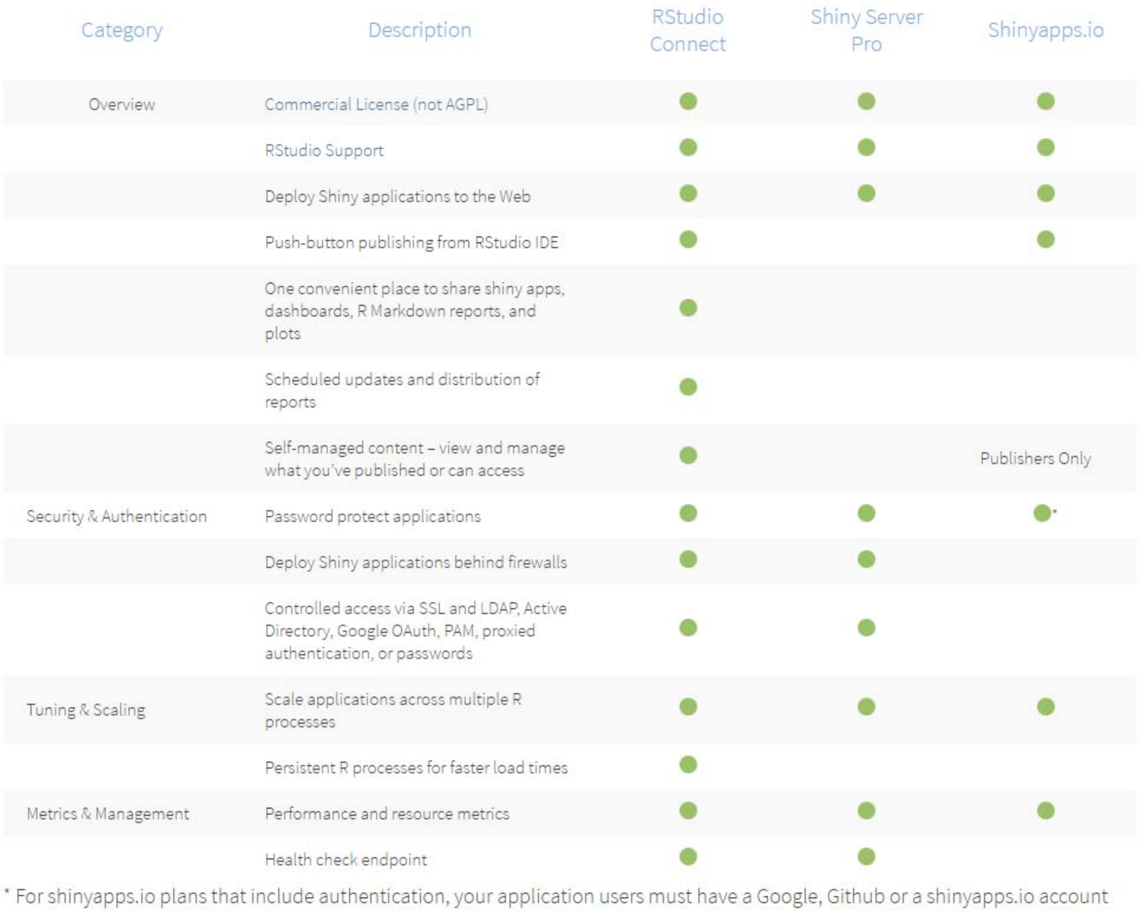
Let's stay in touch. Give us your email and we'll keep you in the loop.

* Email

Read our privacy policy

SUBSCRIBE

Comparativa características

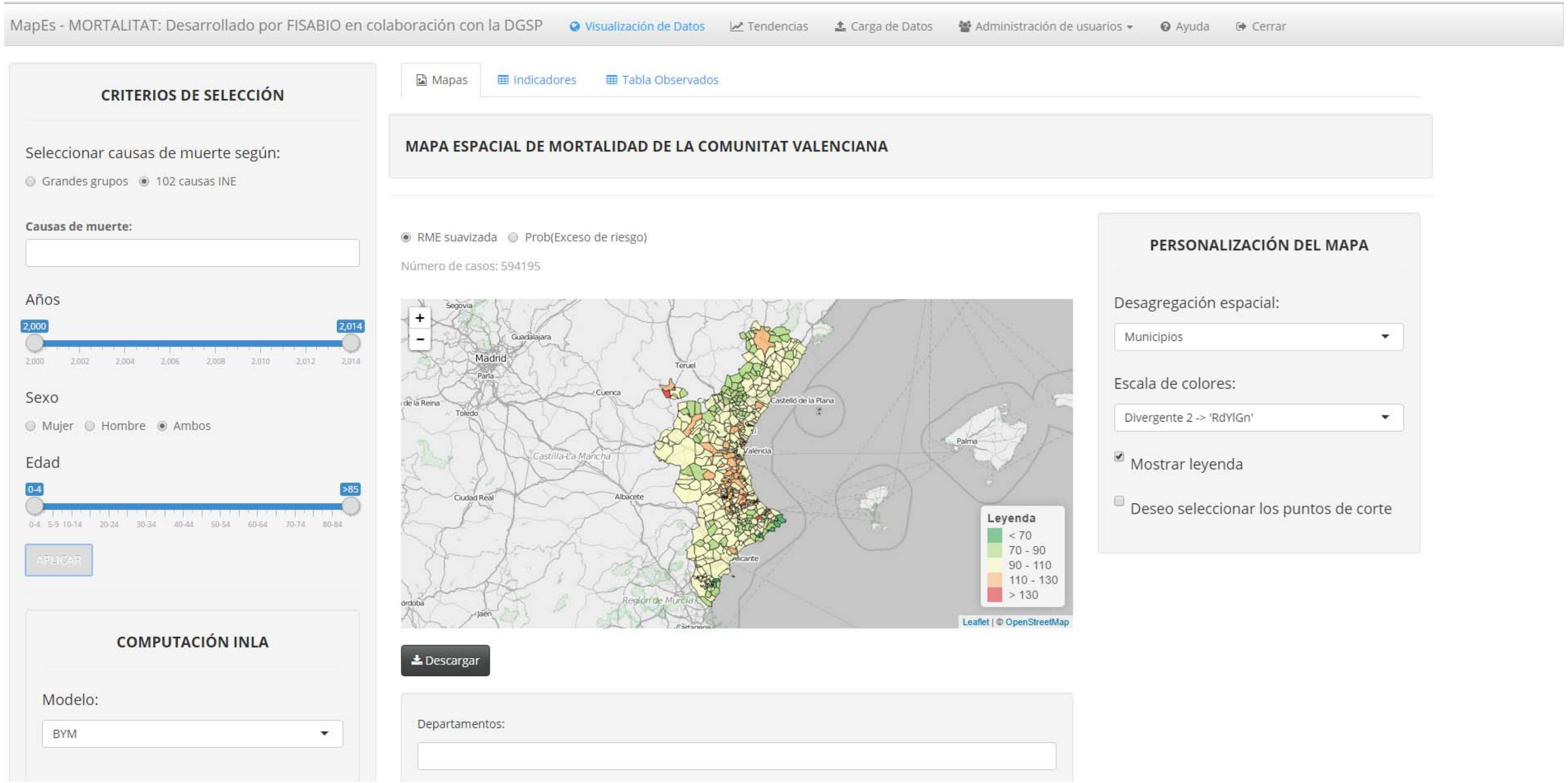


Category	Description	RStudio Connect	Shiny Server Pro	Shinyapps.io
Overview	Commercial License (not AGPL)	●	●	●
	RStudio Support	●	●	●
	Deploy Shiny applications to the Web	●	●	●
	Push-button publishing from RStudio IDE	●		●
	One convenient place to share shiny apps, dashboards, R Markdown reports, and plots	●		
	Scheduled updates and distribution of reports	●		
	Self-managed content – view and manage what you've published or can access	●		Publishers Only
Security & Authentication	Password protect applications	●	●	●*
	Deploy Shiny applications behind firewalls	●	●	
	Controlled access via SSL and LDAP, Active Directory, Google OAuth, PAM, proxied authentication, or passwords	●	●	
Tuning & Scaling	Scale applications across multiple R processes	●	●	●
	Persistent R processes for faster load times	●		
Metrics & Management	Performance and resource metrics	●	●	●
	Health check endpoint	●	●	

* For shinyapps.io plans that include authentication, your application users must have a Google, Github or a shinyapps.io account

4. MapEs (una app desarrollada por FISABIO - DG Salud Pública)

MapEs (http://mapes.fisabio.san.gva.es/MapEs/)



CRITERIOS DE SELECCIÓN

Seleccionar causas de muerte según:

○ Grandes grupos ● 102 causas INE

Causas de muerte:

Años: 2,000 – 2,014

Sexo

○ Mujer ○ Hombre ● Ambos

Edad: 0-4 – >85

COMPUTACIÓN INLA

Modelo: BYM

MAPA ESPACIAL DE MORTALIDAD DE LA COMUNITAT VALENCIANA

● RME suavizada ○ Prob(Exceso de riesgo)

Número de casos: 594195

Leyenda

- 🟩 < 70
- 🟩 70 - 90
- 🟨 90 - 110
- 🟧 110 - 130
- 🟥 > 130

Departamentos:

PERSONALIZACIÓN DEL MAPA

Desagregación espacial: Municipios

Escala de colores: Divergente 2 -> 'RdYlGn'

☑ Mostrar leyenda

☐ Deseo seleccionar los puntos de corte

Bibliografía y recursos

RStudio. 2016a. “Hoja de referencia de Shiny.” https://www.rstudio.com/wp-content/uploads/2015/03/shiny-spanish.pdf (https://www.rstudio.com/wp-content/uploads/2015/03/shiny-spanish.pdf).

———. 2016b. “Shiny - Tutorial.” https://shiny.rstudio.com/tutorial/lesson1/ (https://shiny.rstudio.com/tutorial/lesson1/).

RStudio Team. 2016. . Boston, MA: RStudio, Inc. http://www.rstudio.com/ (http://www.rstudio.com/).

Shi, Kejia. 2017. “Leaflet Cheat Sheet: Leaflet for R.” https://github.com/rstudio/cheatsheets/raw/master/source/pdfs/leaflet cheat sheet.pdf (https://github.com/rstudio/cheatsheets/raw/master/source/pdfs/leaflet cheat sheet.pdf).

Wickham, H. 2015. . Boca Raton, FL: CRC.