

# Introducción a Rmarkdown

Ronny A. Hernández Mora.



@RonnyHdezM



ronnyhdez



<http://ronnyhdez.rbind.io/>



¡Grabar!

# Material del curso

README.md

## ¡Bienvenido al curso de ciencia de datos con R!

Este es un curso libre y gratuito que puede usar para dar sus primeros pasos con el lenguaje de programación R.



Artwork by @allison\_horst

### Material del curso

Sesión	Presentación	Video
0-Preparación	No hay	<a href="https://www.youtube.com/watch?v=NCvJwJSMq60">https://www.youtube.com/watch?v=NCvJwJSMq60</a>
1- Introducción a las herramientas	Por subir	Por subir



[https://github.com/ronnyhdez/curso\\_ciencia\\_datos\\_r](https://github.com/ronnyhdez/curso_ciencia_datos_r)

main 4 branches 0 tags

Go to file Add file **Code**

ronnyhdez Merge pull request #11 from ronnyhdez/T8

- img Ref #1 estructura del curso
- presentaciones Ref #8 material presentacion
- sesion\_01 Ref 1 material sesion 1
- sesion\_02 Ref #8 orden en script segunda ses
- .gitignore Ref #2 materiales sesion 2

Clone

HTTPS SSH GitHub CLI

[https://github.com/ronnyhdez/curso\\_cie](https://github.com/ronnyhdez/curso_cie)

Use Git or checkout with SVN using the web URL.

**Download ZIP**

10 days ago

# ¿Qué queremos de la sesión de hoy?

- Cómo utilizar un archivo .Rmd
- Comprender las ventajas de combinar texto y código
- Entender los elementos básicos de Rmarkdown







Archivos .Rmd

# ¿Qué soluciona?

```
113
114- # Date ranges data visualization
115
116- ```{r date range visualization}
117 # Latent Heat Flux in time
118 towers %>%
119   mutate(date = ymd_hms(date_time)) %>%
120   ggplot(aes(x = date_time, y = latent_heat_flux, color = tower)) +
121   geom_jitter(alpha = 0.5, size = 0.7) +
122   scale_y_continuous(breaks = seq(-4000, 3000, by = 500)) + #
123   scale_x_datetime(date_labels = "%b%Y", breaks = "months") +
124   scale_color_manual(name = "Tower", labels = c("Carbon tower",
125   "Principe tower"),
126   values = c("#1C7636", "#FFCC37")) +
127   theme_light(base_size = 16) +
128   theme(axis.text.x = element_text(angle = 75, h = 1)) +
129   labs(title = "Latent Heat Flux for all observations available from both towers",
130   subtitle = "Without filtering outliers",
131   x = "Date",
132   y = "Latent Heat Flux (W m-2)",
133   color = "Tower")
134
135 # Temperature in time
136 temperature_sensors %>%
137   mutate(date = ymd_hms(date_time)) %>%
138   ggplot(aes(x = date_time, y = temperature)) +
139   geom_point(alpha = 0.5) +
140   scale_y_continuous(breaks = seq(10, 40, by = 2)) +
141   scale_x_datetime(date_labels = "%b%Y", breaks = "months") +
142   theme_light(base_size = 16) +
143   theme(axis.text.x = element_text(angle = 75, h = 1)) +
144   labs(title = "Temperature (C)",
145   subtitle = "All the datapoints available from sensor dataset",
146   x = "Date", y = "Temperature (C)")
```

## ¡Comunicar!



- Con evapotranspiración calculado, sacar conclusiones

### 1 Consideraciones:

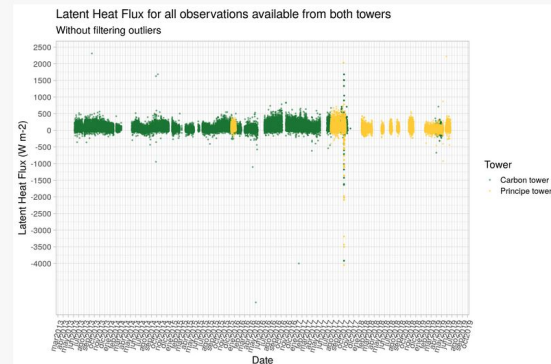
2 Date ranges data visualization

3 Evapotranspiration calculation



### 2 Date ranges data visualization

Code



Code

# ¿Qué soluciona?

Text using Markdown syntax	Corresponding HTML produced by a Markdown processor	Text viewed in a browser
<p><b>Heading</b></p> <p><b>Sub-heading</b></p> <p>Paragraphs are separated by a blank line.</p> <p>Two spaces at the end of a line produce a line break.</p>	<pre>&lt;h1&gt;Heading&lt;/h1&gt; &lt;h2&gt;Sub-heading&lt;/h2&gt; &lt;p&gt;Paragraphs are separated by a blank line.&lt;/p&gt; &lt;p&gt;Two spaces at the end of a line&lt;br /&gt; produce a line break.&lt;/p&gt;</pre>	<p>Heading</p> <p>Sub-heading</p> <p>Paragraphs are separated by a blank line.</p> <p>Two spaces at the end of a line produce a line break.</p>
<p>Text attributes <i>italic</i>, <b>bold</b>, <code>monospace</code>. Some implementations may use <i>single-asterisks</i> for italic text.</p> <p>Horizontal rule:</p> <p>---</p> <p>Strikethrough:</p> <p><del>strikethrough</del></p>	<pre>&lt;p&gt;Text attributes &lt;em&gt;italic&lt;/em&gt;, &lt;strong&gt;bold&lt;/strong&gt;, &lt;code&gt;monospace&lt;/code&gt;. Some implementations may use &lt;i&gt;single-asterisks&lt;/i&gt; for italic text.&lt;/p&gt; &lt;p&gt;Horizontal rule:&lt;/p&gt; &lt;hr /&gt; &lt;p&gt;Strikethrough:&lt;/p&gt; &lt;del&gt;strikethrough&lt;/del&gt;</pre>	<p>Text attributes <i>italic</i>, <b>bold</b>, <code>monospace</code>. Some implementations may use <i>single-asterisks</i> for italic text.</p> <p>Horizontal rule:</p> <p>Strikethrough:</p> <p><del>strikethrough</del></p>
<p>Bullet list:</p> <ul style="list-style-type: none"><li>* apples</li><li>* oranges</li><li>* pears</li></ul> <p>Numbered list:</p> <ol style="list-style-type: none"><li>1. lather</li><li>2. rinse</li><li>3. repeat</li></ol>	<pre>&lt;p&gt;Bullet list:&lt;/p&gt; &lt;ul&gt; &lt;li&gt;apples&lt;/li&gt; &lt;li&gt;oranges&lt;/li&gt; &lt;li&gt;pears&lt;/li&gt; &lt;/ul&gt; &lt;p&gt;Numbered list:&lt;/p&gt; &lt;ol&gt; &lt;li&gt;lather&lt;/li&gt; &lt;li&gt;rinse&lt;/li&gt; &lt;li&gt;repeat&lt;/li&gt; &lt;/ol&gt;</pre>	<p>Bullet list:</p> <ul style="list-style-type: none"><li>• apples</li><li>• oranges</li><li>• pears</li></ul> <p>Numbered list:</p> <ol style="list-style-type: none"><li>1. lather</li><li>2. rinse</li><li>3. repeat</li></ol>



Aaron Swartz

Tomado de: <https://en.wikipedia.org/wiki/Markdown>



John Gruber

Recomendación: [https://www.youtube.com/watch?v=3Q6Fzbgs\\_Lg](https://www.youtube.com/watch?v=3Q6Fzbgs_Lg)



# Elementos de documento Rmarkdown

- YAML
- Chunks
- Índice
- Texto
- Código



# Hoja de referencia

## R Markdown : : GUÍA RÁPIDA

### ¿Qué es R Markdown?

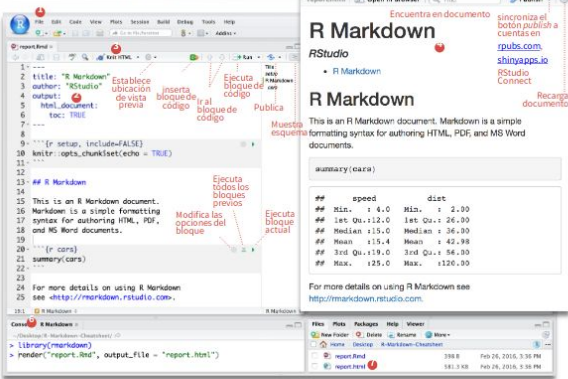
**Rmd** - Un archivo R Markdown (.Rmd) es un registro de tu investigación. Contiene el código que un científico necesita para reproducir el trabajo junto con la narración que el lector requiere para comprenderlo.

**Investigación reproducible** - Con el click de un botón, o el tipo de un comando, puedes volver a ejecutar el código en un archivo R Markdown para reproducir tu trabajo y exportar los resultados como un informe terminado.

**Documentos dinámicos** - Puedes optar por exportar el informe terminado en una variedad de formatos, incluidos documentos html, pdf, MS Word o RTF, dispositivos, cuadernos y más basados en html o pdf.

### Flujo de trabajo

1. **Abre un nuevo archivo .Rmd** en File > New File > R Markdown. Utiliza el asistente que se abre para rellenar previamente el archivo con una plantilla.
2. **Escribe el documento** editando la plantilla.
3. **Compila (Knit)** el documento para crear un reporte: usa el botón **Knit** o **render()** para hacerlo.
4. **Vista previa del resultado** en ventana de IDE.
5. **Publica** (opcional) al servidor web.
6. **Examina el registro de construcción** en la consola R Markdown.
7. **Usa el archivo de salida** que se guarda junto a .Rmd.



### Renderizar

Usa `markdown::render()` para renderizar/knitr desde línea de comando. Argumentos importantes:

<b>input</b> - archivo a compilar	<b>output_options</b> - lista de opciones de renderizado (como en YAML)	<b>output_file</b> - output_dir	<b>params</b> - lista de parámetros a usar	<b>envir</b> - entorno en el cual evaluar bloques de código	<b>encoding</b> - formato de codificación del documento a compilar (input)
-----------------------------------	---	---------------------------------	--	---	--

### Insertar código con knitr syntax

**CÓDIGO INCrustado EN TEXTO**  
Insertarlo con `"r"<code>`. Los resultados aparecen en el texto sin el código  
Built with 'rgetVersion()' Built with 3.2.3

**BLOQUES DE CÓDIGO**  
Usa 6 más líneas rodeadas con `"{r}"` y `"{r}"`. Ubica las opciones del bloque entre las llaves luego de `r`. Insert with `getVersion()`  
`"{r}"` (echo=TRUE)  
`getVersion()`  
`"{r}"` (echo=TRUE)  
`getVersion()`

**OPCIONES GLOBALES**  
Definir con `knitr::opts_chunk$set()`, ej:  
`"{r}"` (include=FALSE)  
`knitr::opts_chunk$set(echo = TRUE)`

### OPCIONES IMPORTANTES DE LOS BLOQUES DE CÓDIGO

- cache** - resultados almacenados en caché para futuras compilaciones / knits (default = FALSE)
- cache.path** - directorio en donde guardar los resultados en caché (default = "cache")
- child** - archivos para compilar luego incluir (default = NULL)
- engine** - lenguaje de programación utilizado en el bloque de código (default = "R")
- error** - mostrar mensaje de error en el doc (TRUE) o detener la compilación cuando hay un error (FALSE) (default = FALSE)
- eval** - ejecutar el código dentro del bloque (default = TRUE)
- fig.align** - 'left', 'right', o 'center' (default = 'center')
- fig.cap** - título del gráfico como cadena de caracteres (default = NULL)
- fig.height**, **fig.width** - dimensiones de los gráficos en pulgadas.
- highlight** - resaltar código fuente (default = TRUE) ejecutarlos (default = TRUE)
- message** - mostrar mensajes del código en el documento (default = TRUE)
- results** (default = "markup")
  - asis** - incluir los resultados en el output
  - hide** - no mostrar resultados
  - hold** - incluir los resultados al final de todo el código
  - tidy** - empujar el código para mostrar (default = FALSE)
- warning** - mostrar las advertencias del código en el documento (default = TRUE)

Opciones no mencionadas: R.options, ani.opts, autopdf, background, cache.comments, cache.lazy, cache.rebuild, cache.warnings, dev, dev.args, dpi, engine.opts, engine.paths, fig.asis, fig.env, fig.height, fig.width, fig.margin, fig.process, fig.retina, fig.scap, fig.show, fig.showenv, fig.subcap, internal, out.extra, out.height, out.width, prompt, print, ret.latex, render.size, split, tidy.opts



### Estructura

**Encabezado YAML**  
Sección opcional del render (ej. pandoc)  
Opciones escritas como pares clave-valor (YAML).  
Al comienzo del archivo  
Entre líneas de ---  
**Texto**  
Narración formateada con markdown, mezclada con:  
**Bloques de código**  
Fragmentos de código incluido. Cada bloque:  
Comienza con `"{r}"`  
Termina con `"{r}"`  
R Markdown ejecuta el código y agrega los resultados al documento.  
Utiliza la ubicación del archivo .Rmd como el **directorio de trabajo**.

### Parámetros

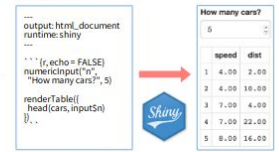
Parametriza tus documentos para reutilizarlos con nuevos inputs (ej. datos, valores, etc.)

1. **Agrega parámetros**: Establece parámetros en el encabezado como subvalores de parámetros.
2. **Llama parámetros**: Llama a los parámetros en el código como `params$nombre`.
3. **Establece parámetros**: con `knitr::opts_chunk$set()` con `params` y `params$nombre` o con el argumento `params` del render().  
`render("doc.Rmd", params = list(n = 1, d = as.Date("2015-01-01")))`

### Documentos interactivos

Convierte tu reporte en un documento interactivo de Shiny en 4 pasos:

1. Añade `runtime::shiny` al encabezado YAML.
2. Utiliza funciones input de Shiny para agregar objetos de input.
3. Utiliza funciones render de Shiny para incluir salidas reactivas.
4. Renderiza con `markdown::run` o clickeando Run Document en el IDE de RStudio.



Incorpora una app completa a tu documento con `shiny::shinyAppDir()`

**Publica en RStudio Connect**, para compartir documentos R Markdown de forma segura, agendar actualizaciones automáticas e interactuar con parámetros en tiempo real [www.rstudio.com/products/connect/](http://www.rstudio.com/products/connect/)



RStudio® es una marca registrada de RStudio, Inc. • CC BY SA RStudio • [info@rstudio.com](mailto:info@rstudio.com) • 844-448-1212 • [rstudio.com](http://rstudio.com) • Obtén más información en [markdown.rstudio.com](http://markdown.rstudio.com) • markdown 1.6 • Actualizado: 2019-12



Café 19:50 a 20:00

¡Manos a la obra!

Abrir un archivo .Rmd en el proyecto



# ¡Manos a la obra!

## Práctica

Ronny Hernández-mora

6/8/2021

## Presentación

Yo soy **Ronny Hernández Mora** y me gusta trabajar con *datos*

Si querés seguirme en redes sociales, podés hacerlo por mi cuenta de [twitter](#)

## Actividades que me gustan

Cuando no estoy en la computadora, me gusta:

- Leer libros
- Andar en bicicleta
- Jugar con arduinos (aunque estoy implica estar en la compu)

## Cursos que he tomado

Durante mi carrera profesional, he tomado los siguientes cursos:

Curso	Del 1 al 5 qué tanto me gustó
Estadística	4
Administración de servidores linux	5
Fisiología vegetal	5
Ecología vegetal	5

Programación en C++  
Algoritmos, estructuras de datos y objetos

L. Joyanes Aguilar

EI385

# Learning Python

FIFTH EDITION

Lutz

LINUX BASICS FOR HACKERS

O'REILLY

Fundamentals of Data Visualization

Wilke

O'REILLY

Mastering Ubuntu Server  
Third Edition

Jay LaCroix

<|

R for Data Science

John Fox  
John Fox

O'REILLY

THINK LIKE A PROGRAMMER

SPRAT



The Pragmatic  
Programmer

20<sup>th</sup> ANNIVERSARY  
EDITION



Zuur • Ieno • Walker  
Saveliev • Smith



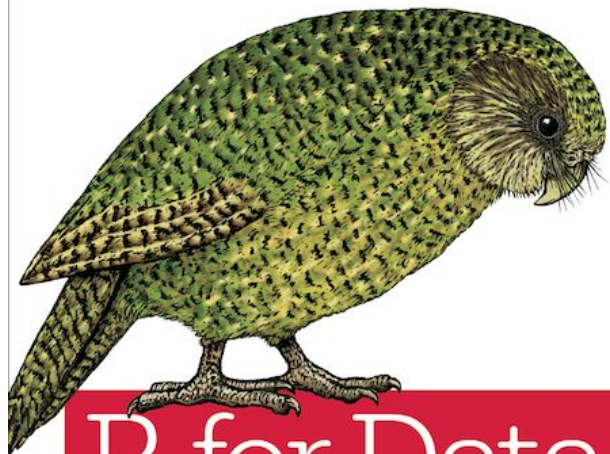
Mixed Effects Models  
Extensions in Ecology with R

Wichham

Advanced R

# Recursos

O'REILLY®



# R for Data Science

VISUALIZE, MODEL, TRANSFORM, TIDY, AND IMPORT DATA


Hadley Wickham &  
Garrett Grolemund

<https://es.r4ds.hadley.nz/>



# ¡Gracias !

Ronny A. Hernández Mora.

 @RonnyHdezM

 ronnyhdez

 <http://ronnyhdez.rbind.io/>

ronny.hernandezm@gmail.com