

Clase 5.0

Output

Marcos Rosetti y Luis Pacheco-Cobos

Estadística y Manejo de Datos con R (EMDR) — Virtual

broom



broom

- Convertir resultados estadísticos en tablas exportables.
- `tidy()` construye un df que resume la información contenida en un modelo.
- `augment()` añade columnas originales a los datos modelados, como predicciones.
- `glance()` construye un resumen de una línea del modelo.

broom

```
library(broom)
lmfit <- lm(mpg ~ wt, mtcars)
summary(lmfit)
```

```
##
## Call:
## lm(formula = mpg ~ wt, data = mtcars)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4.5432 -2.3647 -0.1252  1.4096  6.8727
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  37.2851     1.8776  19.858 < 2e-16 ***
## wt          -5.3445     0.5591  -9.559 1.29e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.046 on 30 degrees of freedom
## Multiple R-squared:  0.7528, Adjusted R-squared:  0.7446
## F-statistic: 91.38 on 1 and 30 DF,  p-value: 1.294e-10
```


broom

```
tidy(lmfit)
```

```
## # A tibble: 2 × 5  
##   term      estimate std.error statistic  p.value  
##   <chr>      <dbl>     <dbl>     <dbl>    <dbl>  
## 1 (Intercept) 37.3       1.88      19.9 8.24e-19  
## 2 wt          -5.34      0.559     -9.56 1.29e-10
```

broom

```
head(augment(lmfit))
```

```
## # A tibble: 6 × 9
##   .rownames      mpg    wt .fitted .resid .std.resid   .hat .sigma .cooks
##   <chr>         <dbl> <dbl> <dbl> <dbl>   <dbl> <dbl> <dbl> <dbl>
## 1 Mazda RX4      21    2.62  23.3 -2.28   -0.766  0.0433  3.07  1.33e-2
## 2 Mazda RX4 Wag  21    2.88  21.9 -0.920  -0.307  0.0352  3.09  1.72e-3
## 3 Datsun 710     22.8  2.32  24.9 -2.09   -0.706  0.0584  3.07  1.54e-2
## 4 Hornet 4 Drive  21.4  3.22  20.1  1.30    0.433  0.0313  3.09  3.02e-3
## 5 Hornet Sportabout 18.7  3.44  18.9 -0.200  -0.0668  0.0329  3.10  7.60e-5
## 6 Valiant        18.1  3.46  18.8 -0.693  -0.231  0.0332  3.10  9.21e-4
```

broom

```
glance(lmfit)
```

```
## # A tibble: 1 × 12
##   r.squared adj.r.squared sigma statistic p.value    df logLik  AIC  BIC
##   <dbl>      <dbl> <dbl>    <dbl>    <dbl> <dbl> <dbl> <dbl> <dbl>
## 1    0.753      0.745  3.05     91.4 1.29e-10     1 -80.0 166. 170.
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

broom

```
lmfit1 <- lm(mpg ~ wt, mtcars)
lmfit2 <- lm(mpg ~ wt + drat, mtcars)
lmfit3 <- lm(mpg ~ wt + drat + hp, mtcars)
```

```
# Resumen eficiente
```

```
all_models <- rbind.data.frame(
  tidy(lmfit1) %>% mutate(model = 1),
  tidy(lmfit2) %>% mutate(model = 2),
  tidy(lmfit3) %>% mutate(model = 3))
all_models
```

```
## # A tibble: 9 × 6
##   term          estimate std.error statistic  p.value model
##   <chr>          <dbl>     <dbl>    <dbl>   <dbl> <dbl>
## 1 (Intercept)  37.3        1.88     19.9  8.24e-19    1
## 2 wt           -5.34       0.559    -9.56  1.29e-10    1
## 3 (Intercept)  30.3        7.32     4.14  2.74e- 4    2
## 4 wt           -4.78       0.797    -6.00  1.59e- 6    2
## 5 drat          1.44       1.46     0.989  3.31e- 1    2
## 6 (Intercept)  29.4        6.16     4.77  5.13e- 5    3
## 7 wt           -3.23       0.796    -4.05  3.64e- 4    3
## 8 drat          1.62       1.23     1.32  1.99e- 1    3
## 9 hp           -0.0322     0.00892  -3.61  1.18e- 3    3
```

formattable & kableExtra

- Paquetes para crear tablas coloridas y visualmente atractivas.

```
library(formattable)  
library(kableExtra)
```

formattable & kableExtra

- Manejan múltiples formatos numéricos.

```
percent(c(0.1, 0.02, 0.03, 0.12))
```

```
## [1] 10.00% 2.00% 3.00% 12.00%
```

```
accounting(c(1000, 500, 200, -150, 0, 1200))
```

```
## [1] 1,000.00 500.00 200.00 (150.00) 0.00 1,200.00
```

- Otros como `comma()`, `currency()`, `scientific()`

formattable & kableExtra

```
mtcars[1:5, 1:4] %>%
  mutate(
    car = row.names(.),
    mpg = color_tile("white", "orange")(mpg),
    cyl = cell_spec(cyl, angle = (1:5)*60,
                    background = "red", color = "white", align = "center"),
    disp = ifelse(disp > 200,
                  cell_spec(disp, color = "red", bold = T),
                  cell_spec(disp, color = "green", italic = T)),
    hp = color_bar("lightgreen")(hp)) %>%
  select(car, everything()) %>%
  kable(escape = F) %>%
  kable_styling("hover", full_width = F) %>%
  column_spec(5, width = "3cm") %>%
  add_header_above(c(" ", "Hello" = 2, "World" = 2))
```

formattable & kableExtra

		Hello		World	
car		mpg	cyl	disp	hp
Mazda RX4	Mazda RX4	21.0	6	160	110
Mazda RX4 Wag	Mazda RX4 Wag	21.0	9	160	110
Datsun 710	Datsun 710	22.8	7	108	93
Hornet 4 Drive	Hornet 4 Drive	21.4	9	258	110
Hornet Sportabout	Hornet Sportabout	18.7	8	360	175

formattable & kableExtra

```
df<-data.frame(  
  id = 1:10,  
  name = c("Bob", "Ashley", "James", "David", "Jenny", "Hans",  
           "Leo", "John", "Emily", "Lee"),  
  age = c(28, 27, 30, 28, 29, 29, 27, 27, 31, 30),  
  grade = c("C", "A", "A", "C", "B", "B", "B", "A", "C", "C"),  
  test1_score = c(8.9, 9.5, 9.6, 8.9, 9.1, 9.3, 9.3, 9.9, 8.5, 8.6),  
  test2_score = c(9.1, 9.1, 9.2, 9.1, 8.9, 8.5, 9.2, 9.3, 9.1, 8.8),  
  final_score = c(9, 9.3, 9.4, 9, 9, 8.9, 9.25, 9.6, 8.8, 8.7),  
  registered = c(TRUE, FALSE, TRUE, FALSE, TRUE, TRUE, TRUE, FALSE, FALSE, FALSE),  
  stringsAsFactors=FALSE)
```

formattable & kableExtra

```
formattable(df,
  list(age = color_tile("white", "orange"),
    grade = formatter("span", style = x ~ ifelse(x == "A",
      style(color = "green", font.weight = "bold"), NA)),
    area(col = c(test1_score, test2_score)) ~ normalize_bar("pink", 0.2),
    final_score = formatter("span",
      style = x ~ style(color = ifelse(rank(-x) <= 3, "green",
        "gray")), x ~ sprintf("%.2f(rank:%02d)", x, rank(-x))),
    registered = formatter("span",
      style = x ~ style(color = ifelse(x, "green", "red")),
      x ~ icontext(ifelse(x, "ok", "remove"),
        ifelse(x, "Yes", "No")))
  ))
```

formattable & kableExtra

id	name	age	grade	test1_score	test2_score	final_score	registered
1	Bob	28	C	8.9	9.1	9.00(rank:06)	Yes
2	Ashley	27	A	9.5	9.1	9.30(rank:03)	No
3	James	30	A	9.6	9.2	9.40(rank:02)	Yes
4	David	28	C	8.9	9.1	9.00(rank:06)	No
5	Jenny	29	B	9.1	8.9	9.00(rank:06)	Yes
6	Hans	29	B	9.3	8.5	8.90(rank:08)	Yes
7	Leo	27	B	9.3	9.2	9.25(rank:04)	Yes
8	John	27	A	9.9	9.3	9.60(rank:01)	No
9	Emily	31	C	8.5	9.1	8.80(rank:09)	No
10	Lee	30	C	8.6	8.8	8.70(rank:10)	No

Rmarkdown & knitr



Rmarkdown & knitr

- Paquetes para producir documentos y presentaciones que incluyen elementos *seamlessly*.
- El **output** es en forma de html o pdf.
- Puede incluir el código con el formato de R y produce los gráficos *on the fly* (al vuelo).

Rmarkdown & knitr



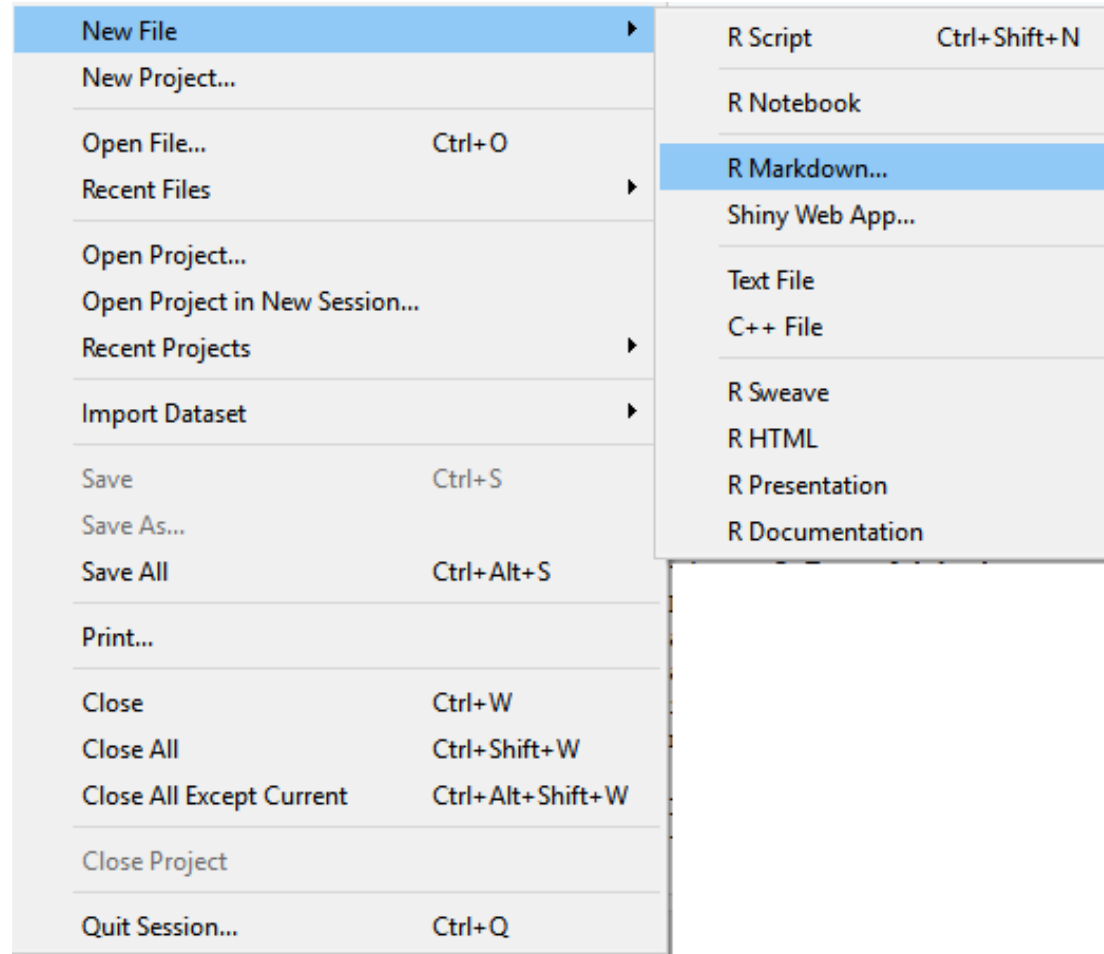
Rmarkdown & knitr

Rmarkdown
TEXT. CODE. OUTPUT.
(GET IT TOGETHER, PEOPLE.)

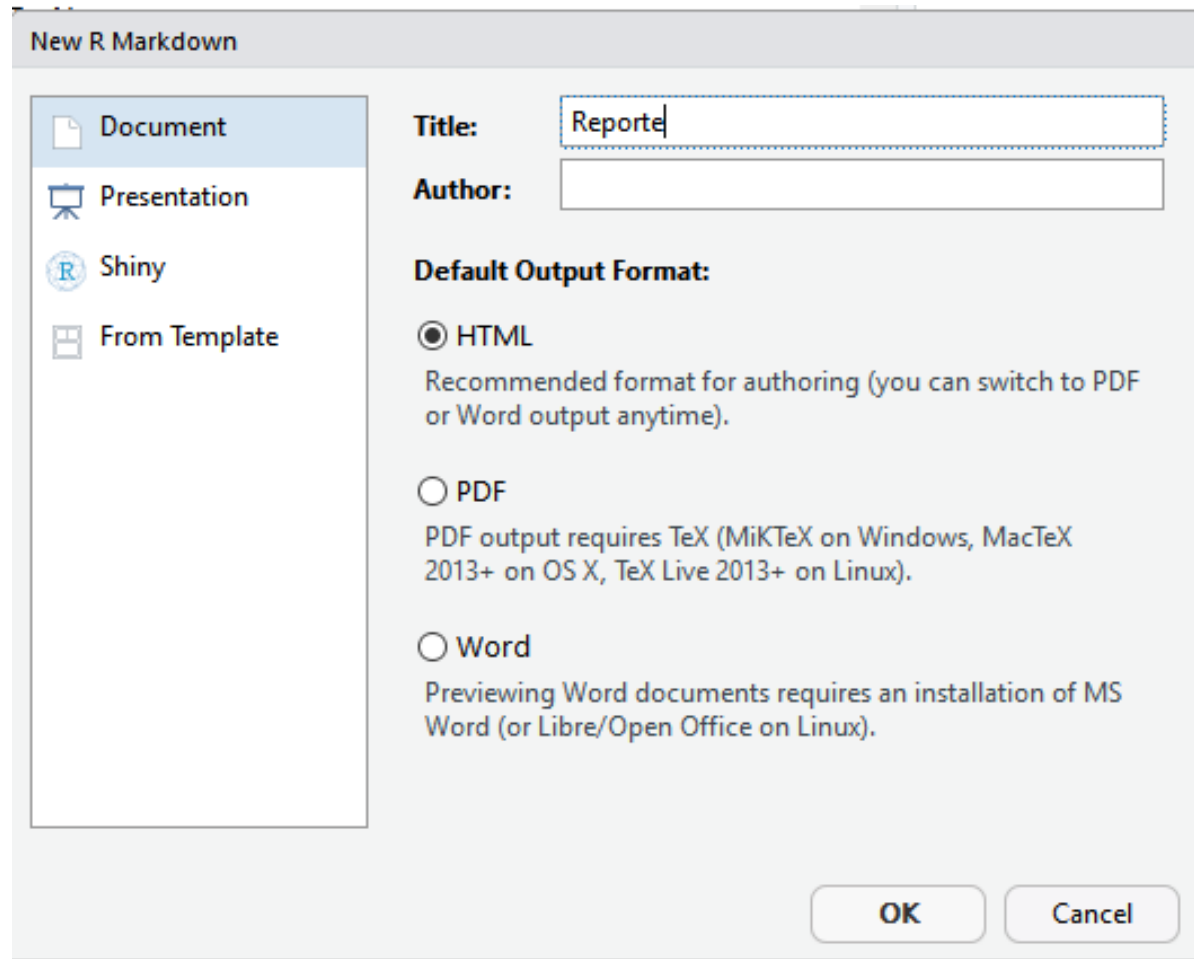


Allison Horst
@allison_horst

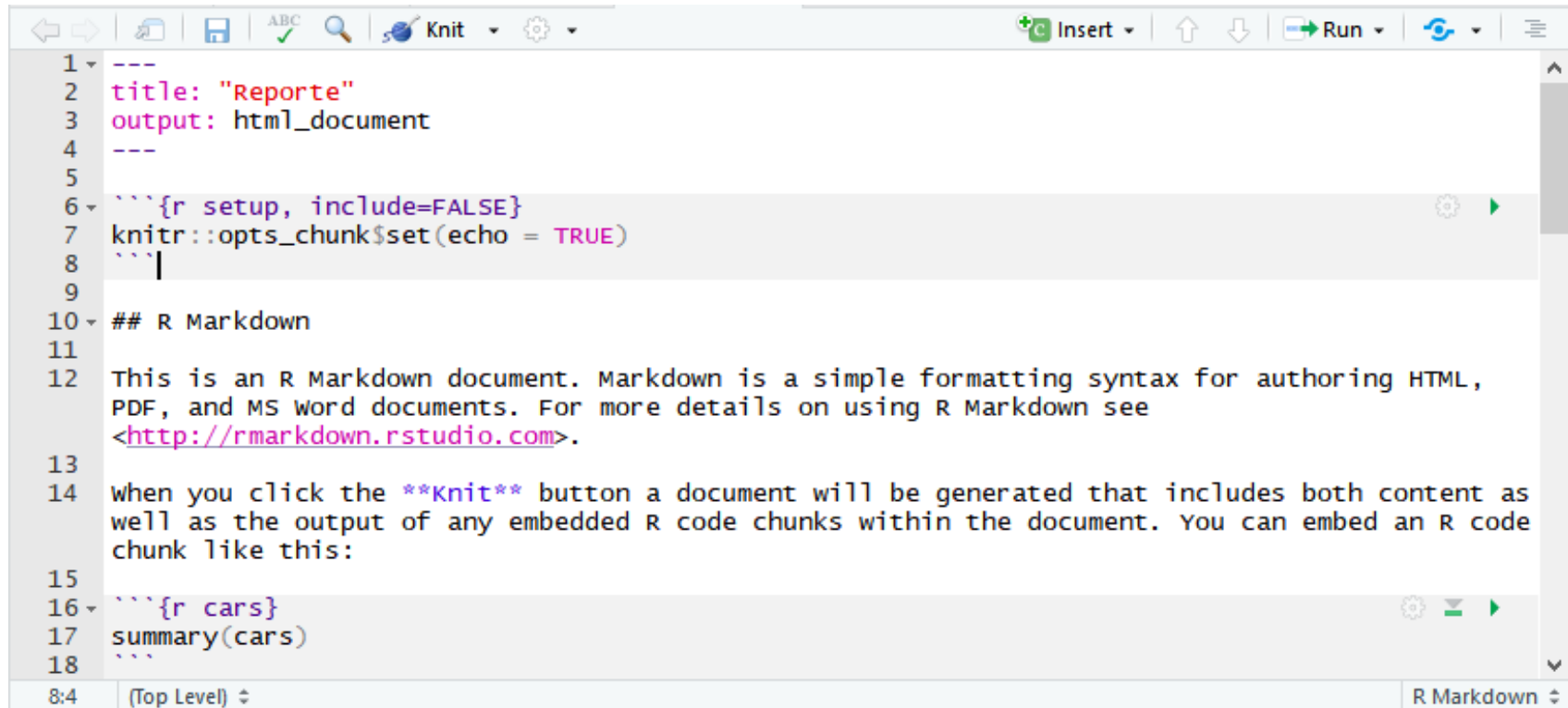
Rmarkdown & knitr



Rmarkdown & knitr



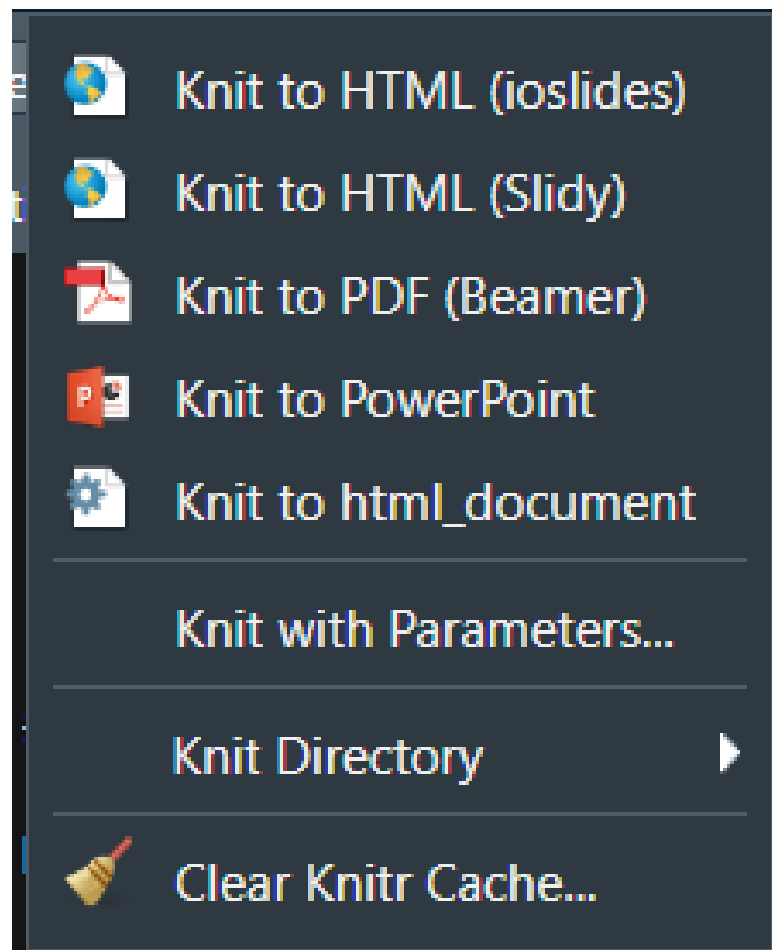
Rmarkdown & knitr



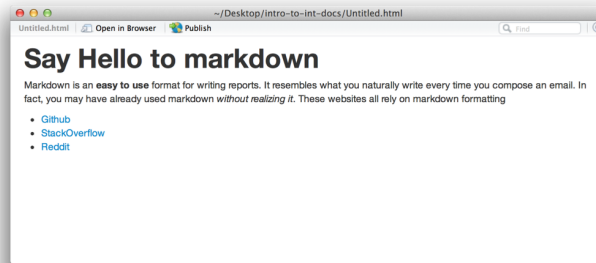
```
1 ---
2 title: "Reporte"
3 output: html_document
4 ---
5
6 ```{r setup, include=FALSE}
7 knitr::opts_chunk$set(echo = TRUE)
8 ```
9
10 ## R Markdown
11
12 This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML,
13 PDF, and MS Word documents. For more details on using R Markdown see
14 <http://rmarkdown.rstudio.com>.
15
16 when you click the knit button a document will be generated that includes both content as
17 well as the output of any embedded R code chunks within the document. You can embed an R code
18 chunk like this:
19
20 ```{r cars}
21 summary(cars)
22 ```
```

8:4 (Top Level) R Markdown

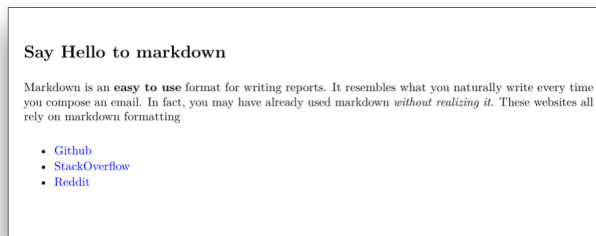
Rmarkdown & knitr



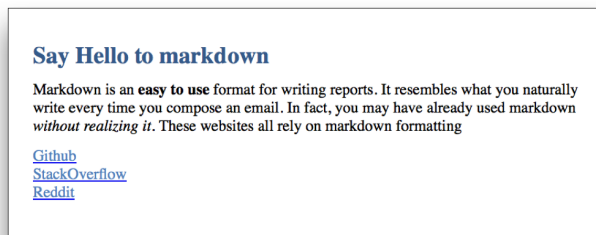
Rmarkdown & knitr



HTML



PDF



MS Word

Texto en Rmarkdown

- Cabezales

H1

H2

H3

H4

H5

H6

Texto en **R**markdown

- También funciona usar líneas.

=====

Texto en **Rmarkdown**

- Énfasis

`*italics*` o `_italics_` produce *italics* or *italics*.

`**bold**` o `__bold__` produce **bold** or **bold**.

`~~Strikethrough~~` produce ~~Strikethrough~~

Texto en **R**markdown

- Listas

1. First ordered list item

2. Another item

- * Unordered list can use asterisks

- Or minuses

- + Or pluses

Texto en Rmarkdown

- Ligas

[Ir a google](https://www.google.com)

```
[Ir a google](https://www.google.com)
```

El código genérico es:

```
[El texto va aquí](https://www.el-enlace-va-aqui.com)
```

Texto en **R**markdown

- Imágenes

```
</center> ![alt text]("image") </center>
```

Texto en **R**markdown

Fórmulas

- Esto:

`$-b \pm \sqrt{b^2 - 4ac} \over 2a$`

- Imprime esto:

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Code chunks (pedazos de código)

```
Here's some code  
```\r}  
dim(iris)
```
```



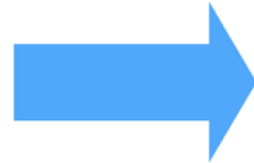
Here's some code

```
dim(iris)
```

```
## [1] 150 5
```

Code chunks (pedazos de código)

```
Here's some code  
```${r echo=FALSE}  
dim(iris)
```
```



Here's some code

```
## [1] 150 5
```

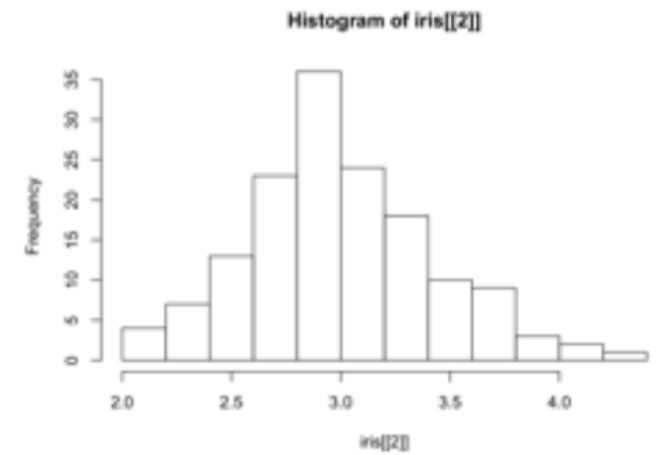
Code chunks (pedazos de código)

Here's a plot

```
```{r echo=FALSE}  
hist(iris[[2]])
```
```



Here's a plot



Code chunks (pedazos de código)

```
Here's some code  
`` `{r eval=FALSE}  
dim(iris)  
`` `
```



Here's some code

```
dim(iris)
```

Shiny

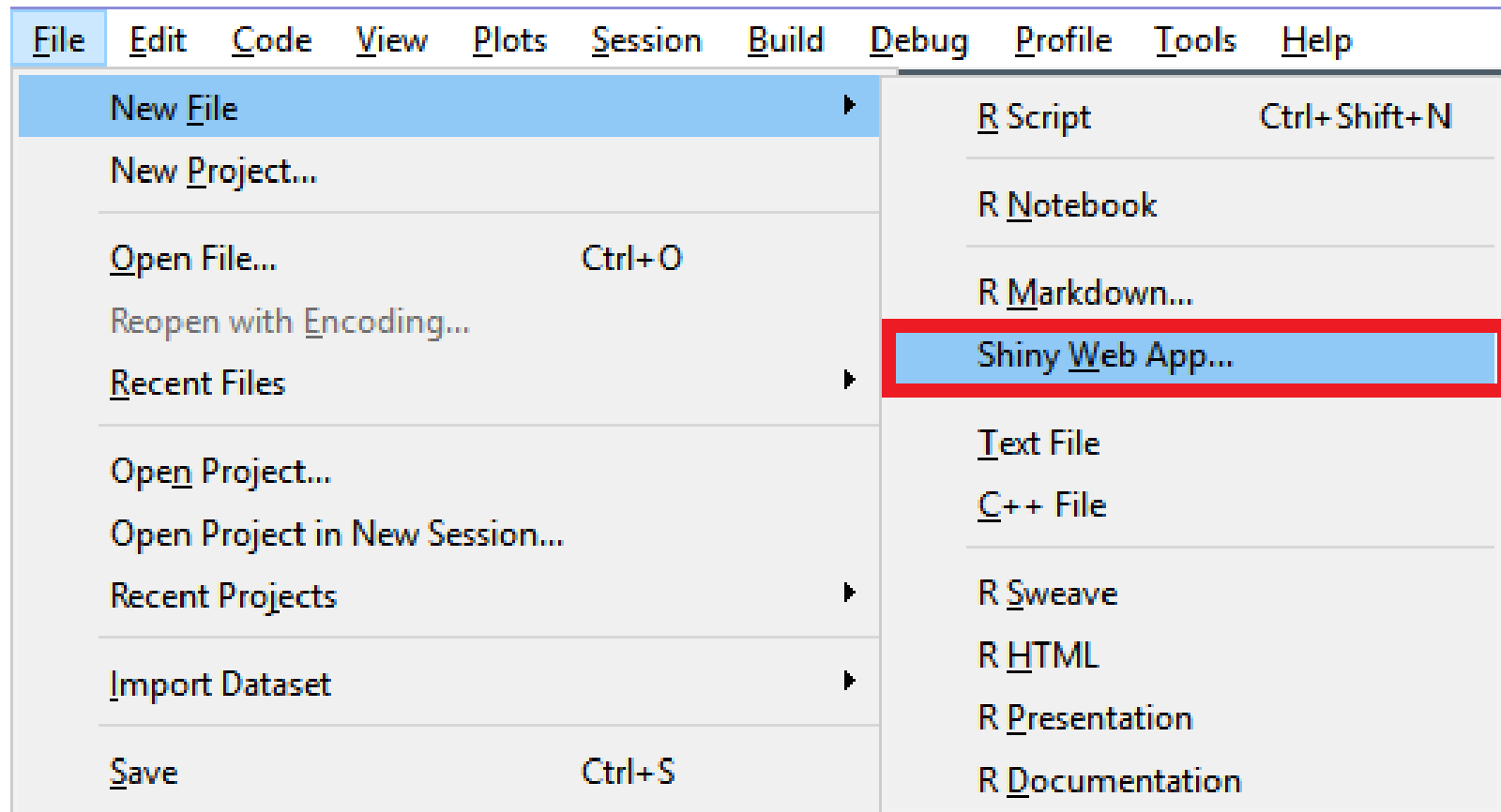


Shiny

Para construir web apps interactivas en R.


```
library(shiny)  
runExample("01_hello")
```

Shiny



Shiny

New Shiny Web Application



Application name:

Application type: Single File (app.R)
 Multiple File (ui.R/server.R)

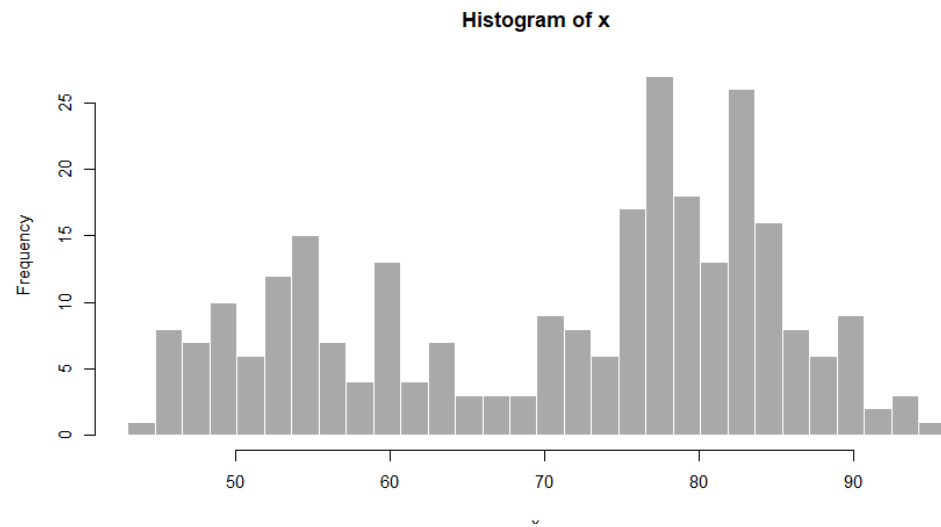
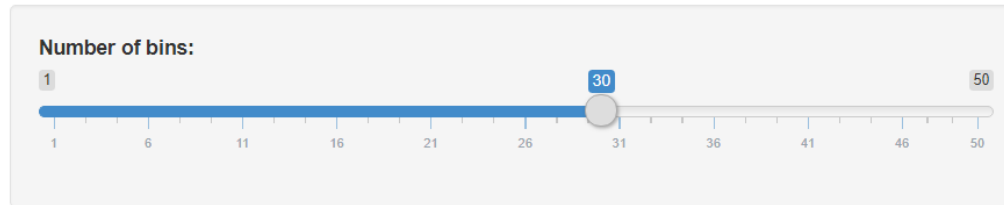
Create within directory:

[? Shiny Web Applications](#)

Shiny

- ¿Cómo generar la siguiente web app?

Old Faithful Geyser Data



Shiny

- `ui` se encarga de crear la parte interactiva de la aplicación.

```
ui <- fluidPage(  
  
  # Título de la aplicación  
  titlePanel("Old Faithful Geyser Data"),  
  
  # Barra lateral con deslizador para seleccionar el número de compartimientos  
  sidebarLayout(  
    sidebarPanel(  
      sliderInput("bins",  
                  "Number of bins:",  
                  min = 1,  
                  max = 50,  
                  value = 30)  
    ),  
  
    # Muestra un gráfico con la distribución generada  
    mainPanel(  
      plotOutput("distPlot")  
    )  
  )  
)
```

Shiny

- server se encarga de crear el servidor y mostrar los resultados.

```
# Define el servidor lógico requerido para dibujar un histograma
server <- function(input, output) {
  output$distPlot <- renderPlot({

    # Genera compartimientos según la entrada input$bins indicada en ui.R
    x <- faithful[, 2]
    bins <- seq(min(x), max(x), length.out = input$bins + 1)

    # Dibuja el histograma con el número especificado de compartimientos
    hist(x, breaks = bins, col = 'darkgray', border = 'white')
  })
}
```

Shiny

- Se combinan en una función que ejecuta la app.

```
# Corre la aplicación  
shinyApp(ui = ui, server = server)
```

Fin del curso



Licencia CC BY



Estadística y Manejo de Datos con R (EMDR) por Marcos F. Rosetti S. y Luis Pacheco-Cobos se distribuye bajo una [Licencia Creative Commons Atribución 4.0 Internacional](https://creativecommons.org/licenses/by/4.0/).