

Presentacion

Construyendo Presentacion

R Markdown

2024-07-09

Table of contents

- 1 Seccion #1 - Distribucion
- 2 Seccion #2 - Math Format
- 3 Chunks/Celdas de codigo
- 4 Multimedia

Section 1

Secccion #1 - Distribucion

Subseccion #1 - Encabezados

Se utilizan los encabezados:

- # Para secciones
- ## Para subsecciones
- ### Para bloques de texto
- #### para bloques de text dobles

Subseccion #2 - Bloques de texto

Bloque de texto simple

Contenido del bloque de texto

Bloque de codigo doble

Contenido del bloque de texto doble

Section 2

Seccion #2 - Math Format

Inicializacion

En la cabecera YAML del archivo .Rmd, establece el tipo de salida a beamer_presentation y especifica el motor LaTeX que deseas usar. Por ejemplo, si prefieres usar pdflatex, tu cabecera YAML comenzaría así:

```
---  
title: "Título de mi Presentación"  
author: "Tu Nombre"  
date: "2024-07-09"  
output:  
  beamer_presentation:  
    latex_engine: pdflatex  
---
```

Inicializacion

Luego de esto se podra operar perfectamente el modo matematico de la siguiente forma:

- Para formato matematico en linea se abre con `\(expresion matematica\)`
- Para formato matematico en salto d linea con tabulacion se abre con `\[expresion matematica\]`
- Para los modificadores de texto funciona de la forma usual de latex

Ejemplo de Modificadores de Texto

Este es un **ejemplo** de cómo se pueden aplicar los modificadores de texto en LaTeX dentro de un documento R Markdown para Beamer.

- **Negrita:** `\textbf{Texto en negrita}`
- *Cursiva:* `\textit{Texto en cursiva}`
- Subrayado: `\underline{Texto subrayado}`
- ~Tamaño de fuente~: `\fontsize{12}{14}\selectfont` Texto de tamaño 12
- +Color de texto+: `\textcolor{blue}{Texto azul}`

Enumerate

Al igual que en \LaTeX :

```
\begin{enumerate}  
  \item Item 1  
\end{enumerate}
```

Y se veria asi:

- 1 Item 1

Tablas

Ejemplo de tabla usando la sintaxis de Markdown:

```
|Esto es |  una |  tabla  |
|-----|-----|-----|
|lo juro |es una| tabla   |
```

Y se debería ver así:

Esto es	una	tabla
lo juro	es una	tabla

Section 3

Chunks/Celdas de código

Chunks de R

Chunk. Bloque de código.

Los bloques de código de R dentro de un documento R Markdown se indican de la manera siguiente

```
{r}
```

```
x = 1+1
```

```
x
```

que resulta en

```
x = 1+1
```

```
x
```

Chunks de R

Hay diversas opciones de crear un bloque de código de R:

- Ir al menú desplegable de “Chunks” y seleccionar el de R
- Introducir manualmente
- Alt + Command + I (para Mac) o Alt + Control + I (para Windows)

Chunks de R

A los chunks se les puede poner etiqueta, para así localizarlos de manera más fácil. Por ejemplo

```
{r PrimerChunk}
```

```
x = 1+2+3
```

```
{r SegundoChunk}
```

```
y = 1*2*3
```

Parámetros de los chunks

La parte entre llaves también puede contener diversos parámetros, separados por comas entre ellos y separados de la etiqueta (o de r, si hemos decidido no poner ninguna).

Estos parámetros determinan el comportamiento del bloque al compilar el documento pulsando el botón Knit situado en la barra superior del área de trabajo.

Parámetros de los chunks

Código	Significado
<code>echo</code>	Si lo igualamos a <code>TRUE</code> , que es el valor por defecto, estaremos diciendo que queremos que se muestre el código fuente del chunk. En cambio, igualado a <code>FALSE</code> , no se mostrará
<code>eval</code>	Si lo igualamos a <code>TRUE</code> , que es el valor por defecto, estaremos diciendo que queremos que se evalúe el código. En cambio, igualado a <code>FALSE</code> , no se evaluará
<code>message</code>	Nos permite indicar si queremos que se muestren los mensajes que R produce al ejecutar código. Igualado a <code>TRUE</code> se muestran, igualado a <code>FALSE</code> no
<code>warning</code>	Nos permite indicar si queremos que se muestren los mensajes de advertencia que producen algunas funciones al ejecutarse. Igualado a <code>TRUE</code> se muestran, igualado a <code>FALSE</code> no
<code>include</code>	Nos permite indicar si queremos incluir el código base en el documento final

Parámetros de los chunks

```
{r, echo=FALSE}
```

```
sec = 10:20
```

```
sec
```

```
cumsum(sec)
```

No aparece

Parámetros de los chunks

```
{r, echo=TRUE, message = TRUE}
```

```
library(datasets)
```

```
head(cars,3)
```

```
library(datasets)
```

```
head(cars,3)
```

```
##      speed dist
```

```
## 1         4     2
```

```
## 2         4    10
```

```
## 3         7     4
```

Parámetros de los chunks

```
{r, echo=TRUE, message = FALSE, comment = NA}
```

```
library(datasets)
```

```
head(cars,3)
```

```
library(datasets)
```

```
head(cars,3)
```

```
      speed dist
1         4    2
2         4   10
3         7    4
```

PD: comment=NA evita que aparezcan los ##

Parámetros de los chunks

Significado	Código	Resultado
results	markup	Valor por defecto. Nos muestra los resultados en el documento final línea a línea, encabezados por ##
results	hide	No se nos muestra el resultado en el documento final
results	asis	Nos devuelve los resultados línea a línea de manera literal en el documento final y el programa con el que se abre el documento final los interpreta como texto y formatea adecuadamente
results	hold	Muestra todos los resultados al final del bloque de código

Parámetros de los chunks

```
{r, echo=TRUE, results = "markup"}
```

```
sec = 10:20`
```

```
sec
```

```
cumsum(sec)
```

```
sec = 10:20
```

```
sec
```

```
## [1] 10 11 12 13 14 15 16 17 18 19 20
```

```
cumsum(sec)
```

```
## [1] 10 21 33 46 60 75 91 108 126 145 165
```

Parámetros de los chunks

Código

```
{r, echo=TRUE, results = "hide"}
```

```
sec = 10:20
```

```
sec
```

```
cumsum(sec)
```

```
sec = 10:20
```

```
sec
```

```
cumsum(sec)
```

Parámetros de los chunks

Código

```
{r, echo=TRUE, results = "asis"}
```

```
sec = 10:20
```

```
sec
```

```
cumsum(sec)
```

```
sec = 10:20
```

```
sec
```

```
[1] 10 11 12 13 14 15 16 17 18 19 20
```

```
cumsum(sec)
```

```
[1] 10 21 33 46 60 75 91 108 126 145 165
```


Parámetros de los chunks

Código

```
{r, echo=TRUE, results = "hold"}
```

```
s
```

```
sec = 10:20
```

```
sec
```

```
cumsum(sec)
```

```
sec = 10:20
```

```
sec
```

```
cumsum(sec)
```

```
## [1] 10 11 12 13 14 15 16 17 18 19 20
```

```
## [1] 10 21 33 46 60 75 91 108 126 145 165
```

Section 4

Multimedia

Imagenes

```
\centering  
\includegraphics[scale=0.2]{img/LogoRStudio.png}
```

Imágenes

