

# Introdução ao tidyverse

## 6 R Markdown

### xaringan [presentation ninja]

---

Maurício Vancine

26/04/2019



# 6 R Markdown

## Conteúdo

6.1 Markdown

6.2 R Markdown

6.3 Instalar pacotes

6.4 Criar um arquivo R Markdown

6.5 Estrutura do RMarkdown

6.6 Sintaxe do Markdown

6.7 Renderizar um arquivo R Markdown

6.8 Principais formatos de saída

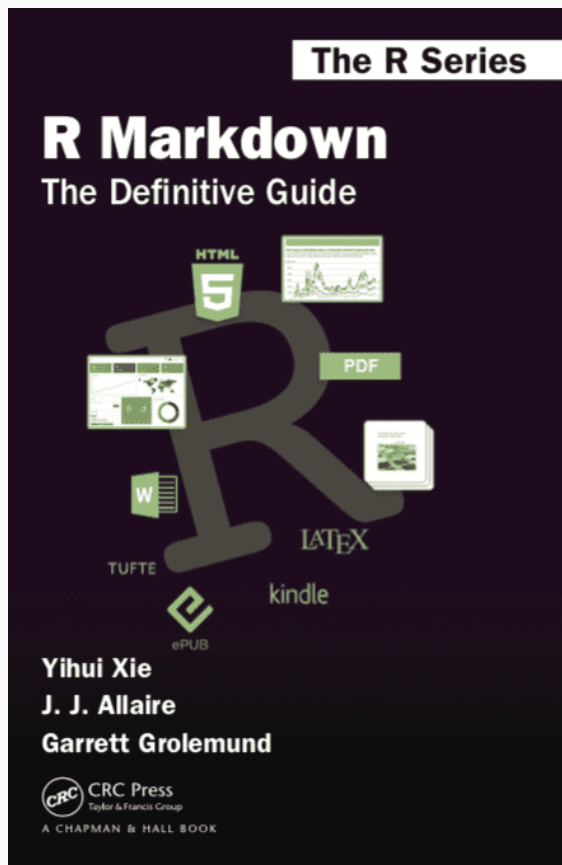
# 6 R Markdown

## Ciência de Dados com R: introdução (2018)



# 6 R Markdown

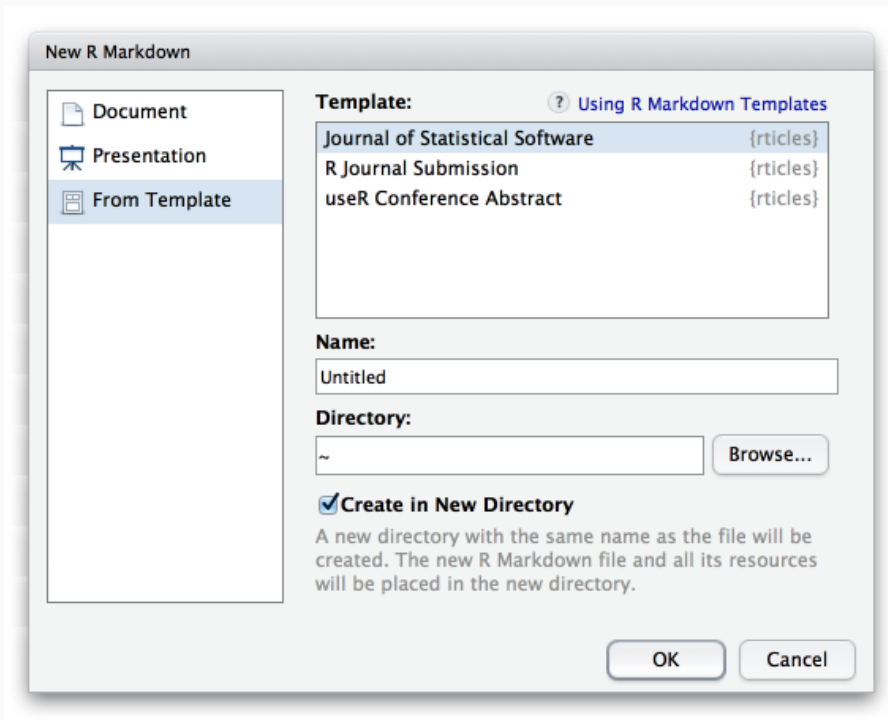
## R Markdown: The Definitive Guide (2018)



# 6 R Markdown

## Pacote para escrever artigos

rticles

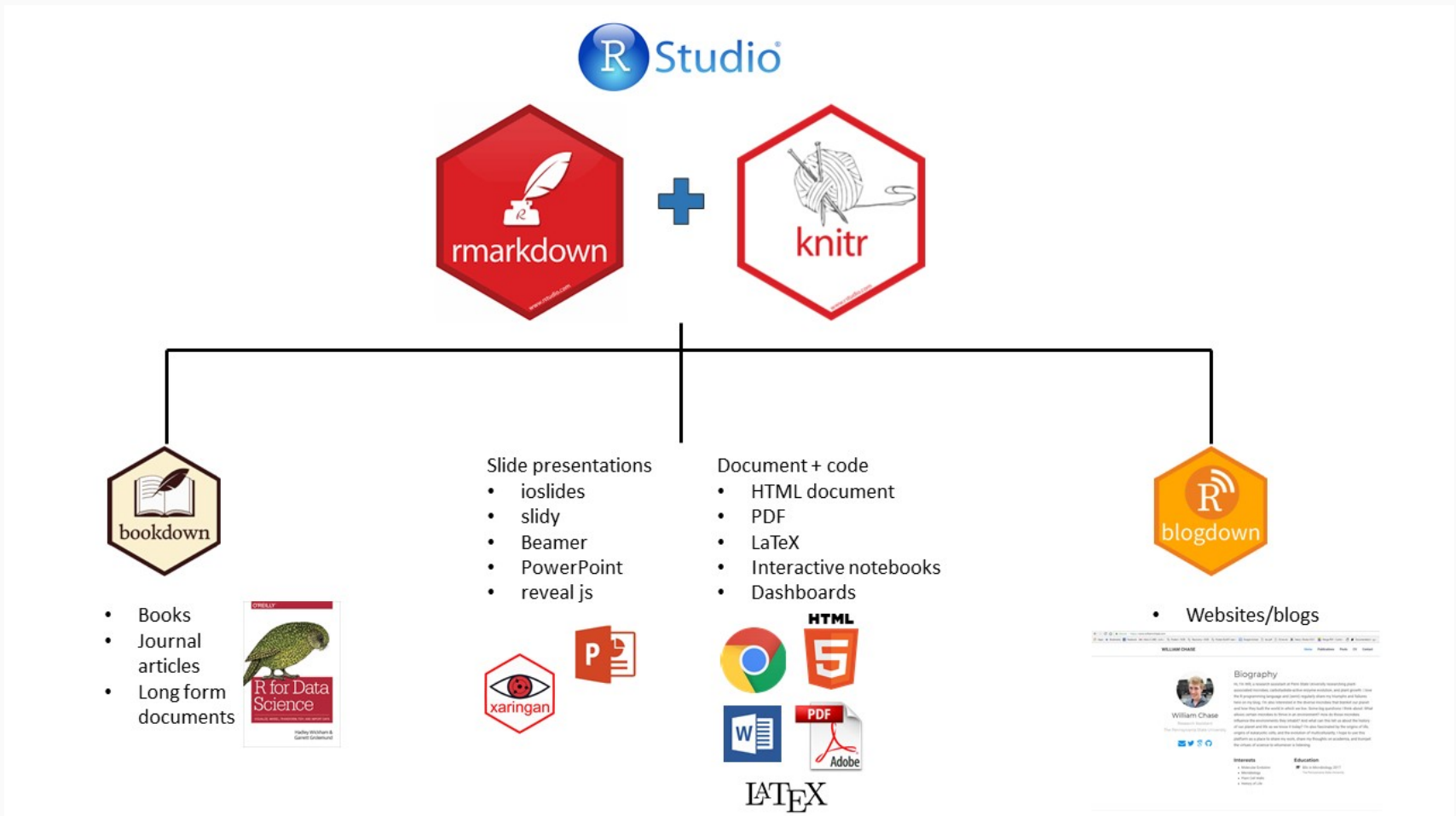


[\*] <https://github.com/rstudio/rticles>

[\*] <https://bookdown.org/yihui/rmarkdown/journals.html>

# 6 R Markdown

## Possibilidades de criação de várias saídas



# 6.1 Markdown

Markdown é uma **linguagem de marcação** que diz como algo deve ser **entendido**, mas não têm capacidade de **processamento ou execução de funções**

A **vantagem do Markdown** é a sua **simplicidade** e a **possibilidade de utilização** de uma linguagem comum para a criação de **vários tipos de documentos**





# 6.2 rmarkdown

## R Markdown Cheatsheet

### R Markdown : : CHEAT SHEET

#### What is R Markdown?

**.Rmd files** - An R Markdown (.Rmd) file is a record of your research. It contains the code that a scientist needs to reproduce your work along with the narration that a reader needs to understand your work.

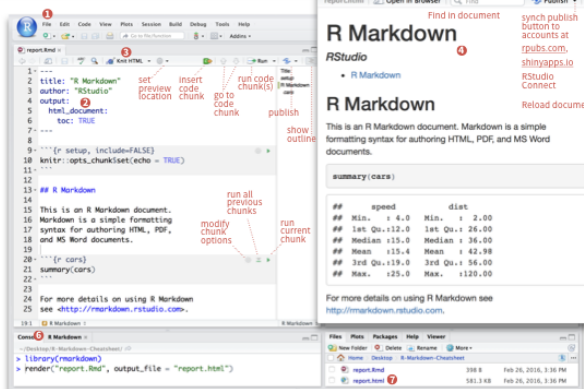
**Reproducible Research** - At the click of a button, or the type of a command, you can run the code in an R Markdown file to reproduce your work and export the results as a finished report.

**Dynamic Documents** - You can choose to export the finished report in a variety of formats, including HTML, PDF, MS Word, or RTF documents, HTML or PDF based slides, Notebooks, and more.

#### Workflow



1. Open a new .Rmd file at File > New File > R Markdown. Use the wizard that opens to pre-populate the file with a template.
2. Write document by editing template.
3. Knit document to create report: use knit button or render() to knit.
4. Preview Output in IDE window.
5. Publish (optional) to web server.
6. Examine build log in R Markdown console.
7. Use output file that is saved along side .Rmd.



#### render

Use `markdown::render()` to render/knit at cmd line. Important args:

**output\_options** - List of render options (as in YAML)

**output\_file** - output file

**output\_dir** - output dir

**params** - list of params to use

**envir** - environment to evaluate code chunks in

**encoding** - of input file

#### Embed code with knitr syntax

**INLINE CODE**  
Insert with `"r<code>".` Results appear as text without code.  
Built with `"r<code>getVersion()"` Built with `"r<code>3.2.3"`

**CODE CHUNKS**  
One or more lines surrounded with `"{r}"` and `"{r}"`. Place chunk options within curly braces, after `"{r}"`. Insert with `"{r<code>getVersion()</code>}"`

**GLOBAL OPTIONS**  
Set with knitr: `opts_chunk$set()`, e.g.

`knitr::opts_chunk$set(echo = TRUE)`

#### IMPORTANT CHUNK OPTIONS

**cache** - cache results for future knits (default = FALSE)

**cache.path** - directory to save cached results in (default = ".cache/")

**child** - files to knit and then include (default = NULL)

**collapse** - collapse all output into single block (default = FALSE)

**comment** - prefix for each line of results (default = "#")

**dependson** - chunk dependencies for caching (default = NULL)

**echo** - Display code in output document (default = TRUE)

**engine** - code language used in chunk (default = "R")

**error** - Display error messages in doc (TRUE) or stop render when errors occur (FALSE) (default = FALSE)

**eval** - Run code in chunk (default = TRUE)

**fig.align** - 'left', 'right', or 'center' (default = 'center')

**fig.cap** - figure caption as character string (default = NULL)

**fig.height**, **fig.width** - Dimensions of plots in inches

**highlight** - highlight source code (default = TRUE)

**include** - include chunk in doc after running (default = TRUE)

**message** - display code messages in document (default = TRUE)

**results** - 'asis' - passthrough results 'hide' - do not display results 'hold' - put all results below all code

**tidy** - tidy code for display (default = FALSE)

**warning** - display code warnings in document (default = TRUE)



RStudio® is a trademark of RStudio, Inc. • CC BY SA RStudio • info@rstudio.com • 844-448-1212 • rstudio.com • Learn more at [rmarkdown.rstudio.com](http://rmarkdown.rstudio.com) • rmarkdown 1.6 • Updated: 2016-02

#### .rmd Structure



**YAML Header**  
Optional section of render (e.g. pandoc) options written as key-value pairs (YAML).  
At start of file  
Between lines of ...  
**Text**  
Narration formatted with markdown, mixed with:  
**Code Chunks**  
Chunks of embedded code. Each chunk:  
Begins with `"{r}"`  
ends with `"{r}"`  
R Markdown will run the code and append the results to the doc.  
It will use the location of the .Rmd file as the **working directory**

#### Parameters

Parameterize your documents to reuse with different inputs (e.g. data, values, etc.)

1. **Add parameters** - Create and set parameters in the header as sub-values of params

```
... params:
  n: 100
  d: r Sys.Date()
... 
```

2. **Call parameters** - Call parameter values in code as `params$<name>`

```
Today's date
is: r params$d
```

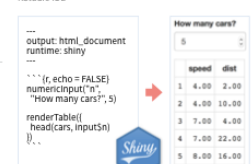
3. **Set parameters** - Set values with knitr with parameters or the params argument of `render()`:

```
render("doc.Rmd", params = list(n = 1,
  d = as.Date("2015-01-01"))
```

#### Interactive Documents

Turn your report into an interactive Shiny document in 4 steps:


1. Add runtime: shiny to the YAML header.
2. Call Shiny input functions to embed input objects.
3. Call Shiny render functions to embed reactive output.
4. Render with markdown: run or click Run Document in RStudio IDE



Embed a complete app into your document with `shiny::shinyAppDir()`  
NOTE: Your report will be rendered as a Shiny app, which means you must choose an HTML output format, like `html_document`, and serve it with an active R session.

# 6.2 rmarkdown

## R Markdown Reference Guide



### R Markdown Reference Guide

Learn more about R Markdown at [rmarkdown.rstudio.com](http://rmarkdown.rstudio.com)  
Learn more about Interactive Docs at [shiny.rstudio.com/articles](http://shiny.rstudio.com/articles)


Contents:

1. Markdown Syntax
2. Knitr chunk options
3. Pandoc options

Syntax	Becomes
<p>Make a code chunk with three back ticks followed by an <code>r</code> in braces. End the chunk with three back ticks:</p> <pre>```r paste("Hello", "World!") ```</pre>	<p>Make a code chunk with three back ticks followed by an <code>r</code> in braces. End the chunk with three back ticks:</p> <pre>paste("Hello", "World!")  ## [1] "Hello World!"</pre>
<p>Place code inline with a single back ticks. The first back tick must be followed by an <code>R</code>, like this <code>`r paste("Hello", "World!")`</code>.</p>	<p>Place code inline with a single back ticks. The first back tick must be followed by an <code>R</code>, like this <code>Hello World!</code>.</p>
<p>Add chunk options within braces. For example, <code>echo=FALSE</code> will prevent source code from being displayed:</p> <pre>```r eval=TRUE, echo=FALSE paste("Hello", "World!") ```</pre>	<p>Add chunk options within braces. For example, <code>echo=FALSE</code> will prevent source code from being displayed:</p> <pre>## [1] "Hello World!"</pre>

Learn more about chunk options at <http://yihui.name/knitr/options>

Chunk options		
option	default value	description
<strong>Code evaluation</strong>		
<code>child</code>	NULL	A character vector of filenames. Knitr will knit the files and place them into the main document.
<code>code</code>	NULL	Set to <code>R</code> code. Knitr will replace the code in the chunk with the code in the code option.
<code>engine</code>	<code>"r"</code>	Knitr will evaluate the chunk in the named language, e.g. <code>engine = "python"</code> . Run <code>names(knitr::knit_engines\$get())</code> to see supported languages.
<code>eval</code>	TRUE	If FALSE, knitr will not run the code in the code chunk.
<code>include</code>	TRUE	If FALSE, knitr will run the chunk but not include the chunk in the final document.
<code>purrr</code>	TRUE	If FALSE, knitr will not include the chunk when running <code>purrr</code> to extract the source code.
<strong>Results</strong>		
<code>collapse</code>	FALSE	If TRUE, knitr will collapse all the source and output blocks created by the chunk into a single block.
<code>echo</code>	TRUE	If FALSE, knitr will not display the code in the code chunk above it's results in the final document.
<code>results</code>	<code>"markup"</code>	If <code>"hide"</code> , knitr will not display the code's results in the final document. If <code>"hold"</code> , knitr will delay displaying all output pieces until the end of the chunk. If <code>"asis"</code> , knitr will pass through results without reformatting them (useful if results return raw HTML, etc.).
<code>error</code>	TRUE	If FALSE, knitr will not display any error messages generated by the code.
<code>message</code>	TRUE	If FALSE, knitr will not display any messages generated by the code.
<code>warning</code>	TRUE	If FALSE, knitr will not display any warning messages generated by the code.
<strong>Code Decorations</strong>		
<code>background</code>	<code>"#FFFFFF"</code>	A background color for chunks in LaTeX output.
<code>comment</code>	<code>"#"</code>	A character string. Knitr will append the string to the start of each line of results in the final document.
<code>highlight</code>	TRUE	If TRUE, knitr will highlight the source code in the final output.
<code>prompt</code>	FALSE	If TRUE, knitr will add <code>&gt;</code> to the start of each line of code displayed in the final document.
<code>size</code>	<code>"normalize"</code>	Fontsize for LaTeX output.
<code>strip.white</code>	TRUE	If TRUE, knitr will remove white spaces that appear at the beginning or end of a code chunk.
<code>tidy</code>	FALSE	If TRUE, knitr will tidy code chunks for display with the <code>tidy_source()</code> function in the <code>formattest</code> package.

 Updated 10/30/2014 © 2014 RStudio, Inc. All rights reserved.

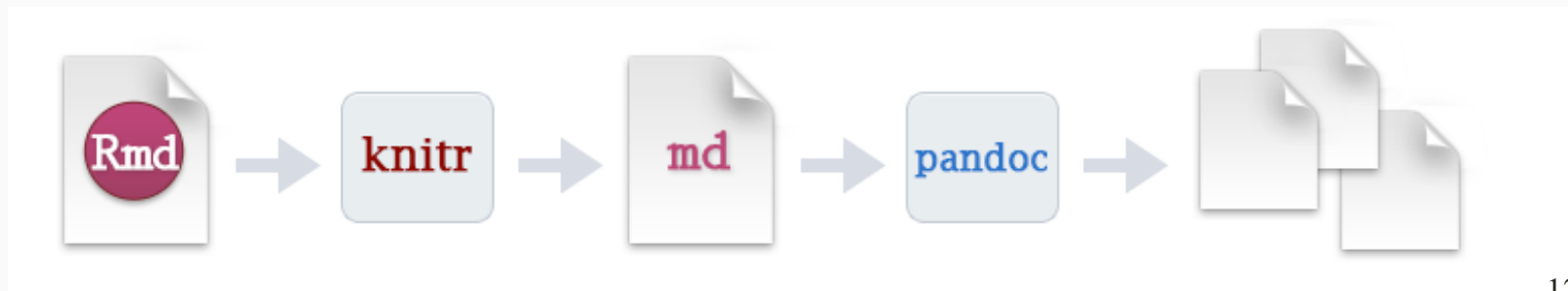
[\*] <https://www.rstudio.com/wp-content/uploads/2015/03/rmarkdown-reference.pdf>

## 6.2 R Markdown

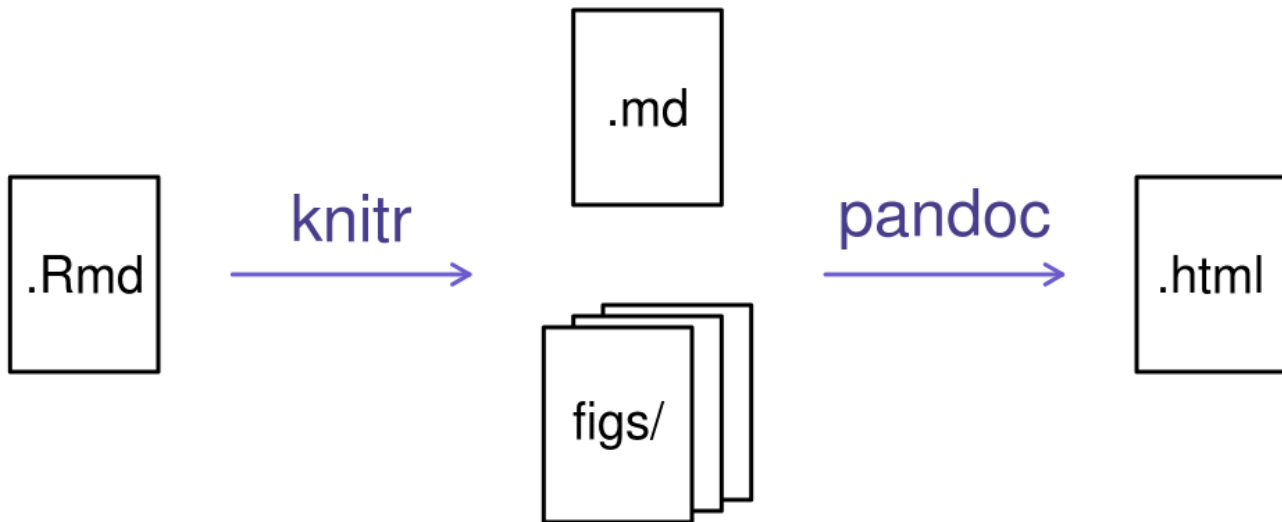
O **R Markdown** executa **scripts em R** e os incorpora a um arquivo Markdown (extensão .md)

O pacote **knitr** executa “pedaços” (**chunks**) de códigos e gera um arquivo .md com os códigos e resultados

O **pandoc** converte para a linguagem desejada, gerando os arquivos nos formatos escolhidos (.html, .docx, .pdf, .odt)



## 6.2 R Markdown



# 6.3 Instalar pacotes

## Instalar do CRAN

```
install.packages("rmarkdown")  
library(markdown)
```

## Instalar do CRAN

```
install.packages("knitr")  
library(knitr)
```

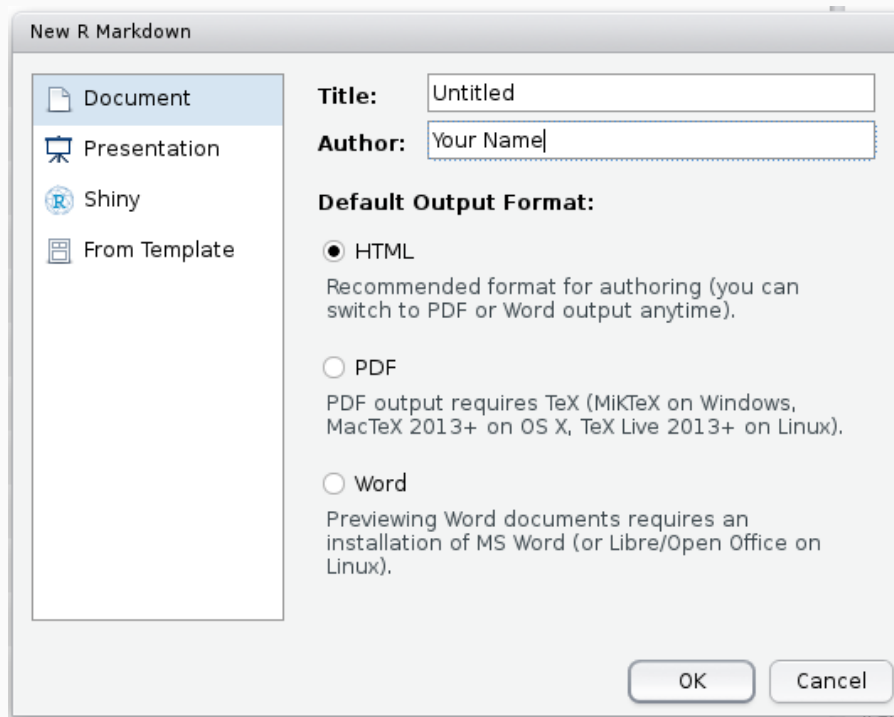
## Para gerar pdfs

```
install.packages("tinytex")  
library(tinytex)  
  
# install TinyTeX  
install_tinytex() # pode demorar
```

# 6.4 Criar um arquivo R Markdown

É um arquivo **diferente** do script (.R), com o formato (.Rmd) = R Markdown

File -> New File -> R Markdown



# 6.5 Estrutura do RMarkdown

The image shows a screenshot of an R Markdown document in RStudio, titled "Untitled1". The document is structured as follows:

- Header (YAML front-matter):** Lines 2-5 contain metadata: `title: "notes"`, `author: "Kieran healy"`, `date: "9/13/2017"`, and `output: html_document`. This section is annotated with "Information about the document".
- Code chunk:** Lines 8-10 contain an R code chunk: `{r setup, include=FALSE}`, `knitr::opts_chunk$set(echo = TRUE)`, and `{r}`. This section is annotated with "Code chunk".
- Text:** Lines 12-13 contain the heading `## R Markdown`. Lines 14-15 contain a paragraph: "This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <<http://rmarkdown.rstudio.com>>." Lines 16-17 contain another paragraph: "When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:". This section is annotated with "Notes and discussion, with formatting instructions".
- Code chunk:** Lines 18-20 contain an R code chunk: `{r cars}`, `summary(cars)`, and `{r}`. This section is annotated with "Run all chunks up to this point".
- Text:** Line 22 contains the heading `## Including Plots`. Line 24 contains a paragraph: "You can also embed plots, for example:".
- Code chunk:** Lines 26-28 contain an R code chunk: `{r pressure, echo=FALSE}`, `plot(pressure)`, and `{r}`. This section is annotated with "Set options for this code chunk".
- Text:** Line 30 contains a paragraph: "Note that the `'echo = FALSE'` parameter was added to the code chunk to prevent printing of the R code that generated the plot."

Annotations in the image include:

- A green circle around the **Knit** button in the toolbar, with the text "Process the whole document".
- A green bracket pointing to the header section, with the text "Information about the document".
- A green bracket pointing to the first code chunk, with the text "Code chunk".
- A green bracket pointing to the text section, with the text "Notes and discussion, with formatting instructions".
- A green circle around the **Run** button in the toolbar, with the text "Run just this chunk".
- A green circle around the **Run** button in the toolbar, with the text "Run all chunks up to this point".
- A green circle around the **Run** button in the toolbar, with the text "Set options for this code chunk".



# 6.5 Estrutura do RMarkdown

## 1 YAML

Responsável pelas **configurações dos documentos**, onde informamos o título do documento, a data de criação, o nome do autor e o tipo de output que desejamos

```
---  
title: "Exemplo"  
author: "Seu nome"  
date: "15-01-2019"  
output:  
  html_document: default  
  pdf_document:  
    fig_caption: yes  
    fig_height: 3.5  
    fig_width: 7  
    number_sections: yes  
lang: pt-br  
---
```

# 6.5 Estrutura do RMarkdown

## 2 Code Chunks

Pedaços de código em R para serem executados e gerar resultados que serão incorporados ao documento

Atalho: `ctrl + alt + I`

```
```${r cars}  
summary(cars)  
```
```



# 6.5 Estrutura do RMarkdown

## 2 Code Chunks

Valores são acrescentados aqui: `{r, eval = FALSE}`

| Opção      | Valor Padrão | Descrição  |
|------------|--------------|--|
| eval       | TRUE         | Execução do código   |
| include    | TRUE         | Exibição do código. Resultados não aparecem  |
| collapse   | FALSE        | Colapsar os resultados num bloco   |
| echo       | FALSE        | Exibição do código. Resultados aparecem  |
| results    | markup       | hide: sem resultados; hold: resultados formatados; asis: resultados não formatados |
| error      | TRUE         | Mostrar erros  |
| message    | TRUE         | Mostrar mensagens  |
| warning    | TRUE         | Mostrar warnings   |
| fig.height | 7            | Altura do gráficos em polegadas  |
| fig.width  | 7            | Largura do gráficos em polegadas   |

# 6.5 Estrutura do RMarkdown

## 3 Markdown

Formato de texto que será exibido quando o arquivo for "renderizado"

```
## Título  
Texto normal  
  
Mais uma linha  
  
Outros formatos de texto Markdown serão explicados...
```

# 6.6 Sintaxe do Markdown

## Cabeçalhos

```
# Cabeçalho de primeiro nível  
## Cabeçalho de segundo nível  
### Cabeçalho de terceiro nível
```

## Estilo de texto

```
# *Itálico* e **Negrito**
```

## Citações

```
> Aqui vai um texto para citação
```

## Código no texto

```
`mean(x)`
```

## 6.6 Sintaxe do Markdown

### Código processado no texto

```
` r mean(c(2, 3, 4)) `
```

Imagens - tem de estar na mesma pasta **images**, no mesmo diretório que o arquivo . Rmd

```

```

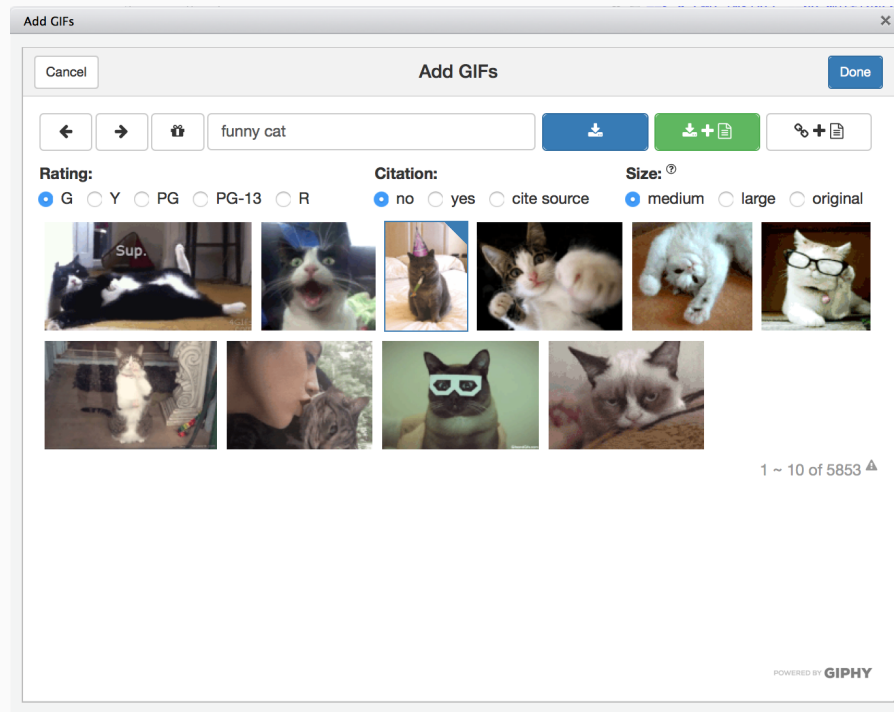
```
![Título Opcional](images/code_chunk.png)
```

# 6.6 Sintaxe do Markdown

## GIFs

O Addin [giphyr](#) procura Gifs diretamente no RStudio

```
install.packages("giphyr")  
devtools::install_github("haozhu233/giphyr")
```



# 6.6 Sintaxe do Markdown

## Listas não ordenadas

```
# * Item a
# * Item b
    + Subitem b1
    + Subitem b2
```

## Listas ordenadas

```
1. Item 1
2. Item 2
3. Item 3
    i. Item 3a
    ii. Item 3b
```



# 6.6 Sintaxe do Markdown

## Tabelas

```
Coluna 1	Coluna 2
10	Brasil
20	China
```

Fonte: MDIC.

## Gerador de tabelas para o Markdown

[\*] [https://www.tablesgenerator.com/markdown\\_tables](https://www.tablesgenerator.com/markdown_tables)

# 6.6 Sintaxe do Markdown

## Tabelas

```
x <- letters[1:3]
y <- LETTERS[1:3]
knitr::kable(data.frame(x, y), align = 'cc')
```

## Sites

```
Site do [MDIC] (www.mdic.gov.br)
```

## Linha horizontal ou Quebra de Página

```
# ***
---
```

# 6.6 Sintaxe do Markdown

## Equação

Modelo linear simples

```
$y_i = \alpha + \beta x_i + e_i$
```

$$y_i = \alpha + \beta x_i + e_i$$

## Equação em Bloco

Modelo linear simples

```
$$y_i = \alpha + \beta x_i + e_i$$
```

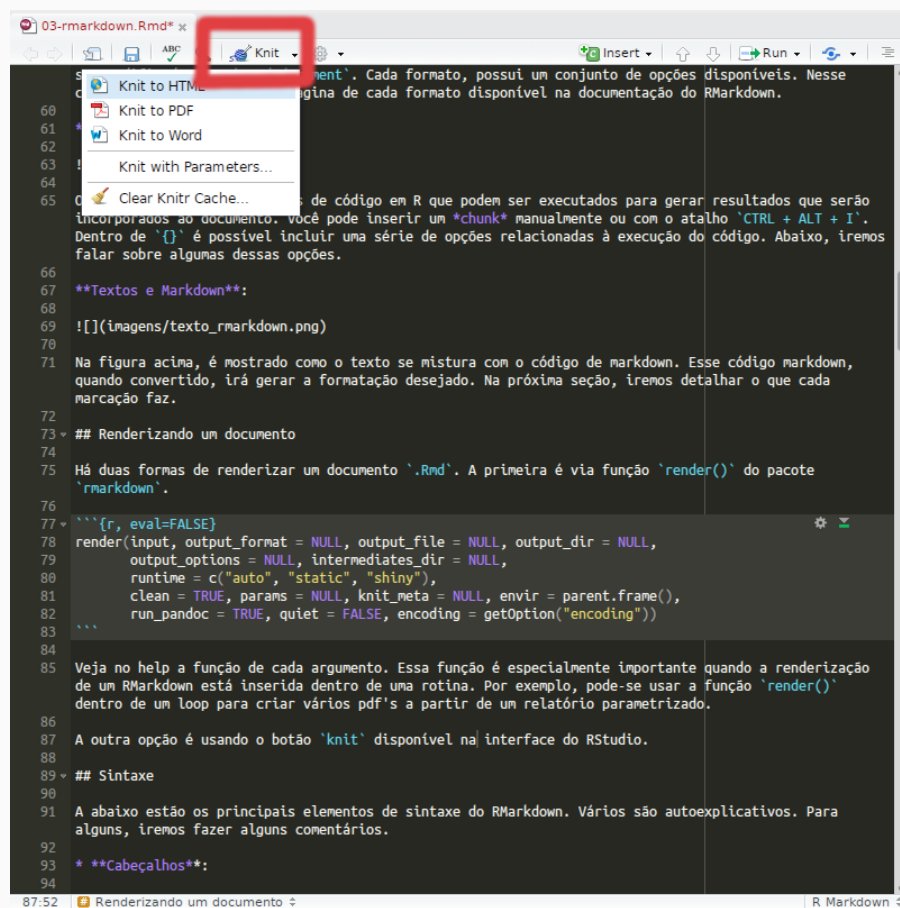
```
$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2b}$$
```

$$y_i = \alpha + \beta x_i + e_i$$

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2b}$$

# 6.7 Renderizar um R Markdown

Usando o botão `knit`, disponível na interface do RStudio



# 6.8 Principais formatos de saída

R Markdown  
Text + markdown + R code

```
1 # Example Markdown Document
2 # Author: Nathaniel Phillips
3 # Date: 30 November 2016
4 # Title: Introduction
5 # Keywords: R, Markdown, Quarto
6 # Abstract: This document is a placeholder for the abstract.
7 # Introduction
8 # Method
9 # Participants
10 # Results
11 # Discussion
12 # Conclusion
13 # Bibliography
14 # Appendix
15 # References
16 # Figures
17 # Tables
18 # Code
19 # Tables
20 # Figures
21 # Tables
22 # Figures
23 # Tables
24 # Figures
25 # Tables
26 # Figures
27 # Tables
28 # Figures
29 # Tables
30 # Figures
31 # Tables
32 # Figures
33 # Tables
34 # Figures
35 # Tables
36 # Figures
37 # Tables
38 # Figures
39 # Tables
40 # Figures
41 # Tables
42 # Figures
43 # Tables
44 # Figures
45 # Tables
46 # Figures
47 # Tables
48 # Figures
49 # Tables
50 # Figures
51 # Tables
52 # Figures
53 # Tables
54 # Figures
55 # Tables
56 # Figures
57 # Tables
58 # Figures
59 # Tables
60 # Figures
61 # Tables
62 # Figures
63 # Tables
64 # Figures
65 # Tables
66 # Figures
67 # Tables
68 # Figures
69 # Tables
70 # Figures
71 # Tables
72 # Figures
73 # Tables
74 # Figures
75 # Tables
76 # Figures
77 # Tables
78 # Figures
79 # Tables
80 # Figures
81 # Tables
82 # Figures
83 # Tables
84 # Figures
85 # Tables
86 # Figures
87 # Tables
88 # Figures
89 # Tables
90 # Figures
91 # Tables
92 # Figures
93 # Tables
94 # Figures
95 # Tables
96 # Figures
97 # Tables
98 # Figures
99 # Tables
100 # Figures
```

“Knit” to PDF

“Knit” to HTML

“Knit” to Word

Example Markdown Document

Nathaniel Phillips

30 November 2016

### Introduction

The purpose of this research is to understand how people evaluate the desirability of people based on their social media profiles. To answer this, we had participants view social media profiles of single people and, based on those profiles, evaluate how desirable the people were. We had two hypotheses based on previous research:

1. Women would receive higher desirability ratings than men.
2. People with rare hair colors would have higher desirability ratings than those with common hair colors.

### Method

#### Participants

We collected data from 1000 participants recruited from campus. The mean age was 22.33 (479 female and 521 male). Data were collected over the course of 50 experimental sessions.

### Results

Table 1 presents the first few rows of the data:

First few rows of the survey data:

| session | sex | age | haircolor | university   | education    | atbless | intelligence | attractiveness | desirability |
|---------|-----|-----|-----------|--------------|--------------|---------|--------------|----------------|--------------|
| 1       | m   | 22  | brwn      | 3.Georgetown | 3.Master     | 3.Yes   | 1.Low        | 3.High         | 15           |
| 1       | m   | 19  | blond     | 2.Durham     | 1.HighSchool | 1.No    | 2.medium     | 2.medium       | 40           |

Example Markdown Document

Nathaniel Phillips

30 November 2016

### Introduction

The purpose of this research is to understand how people evaluate the desirability of people based on their social media profiles. To answer this, we had participants view social media profiles of single people and, based on those profiles, evaluate how desirable the people were. We had two hypotheses based on previous research:

1. Women would receive higher desirability ratings than men.
2. People with rare hair colors would have higher desirability ratings than those with common hair colors.

### Method

#### Participants

We collected data from 1000 participants recruited from campus. The mean age was 22.33 (479 female and 521 male). Data were collected over the course of 50 experimental sessions.

### Results

Table 1 presents the first few rows of the data:

First few rows of the survey data:

| session | sex | age | haircolor | university   | education    | atbless | intelligence | attractiveness | desirability |
|---------|-----|-----|-----------|--------------|--------------|---------|--------------|----------------|--------------|
| 1       | m   | 22  | brwn      | 3.Georgetown | 3.Master     | 3.Yes   | 1.Low        | 3.High         | 15           |
| 1       | m   | 19  | blond     | 2.Durham     | 1.HighSchool | 1.No    | 2.medium     | 2.medium       | 40           |

Example Markdown Document

Nathaniel Phillips

30 November 2016

### Introduction

The purpose of this research is to understand how people evaluate the desirability of people based on their social media profiles. To answer this, we had participants view social media profiles of single people and, based on those profiles, evaluate how desirable the people were. We had two hypotheses based on previous research:

1. Women would receive higher desirability ratings than men.
2. People with rare hair colors would have higher desirability ratings than those with common hair colors.

### Method

#### Participants

We collected data from 1000 participants recruited from campus. The mean age was 22.33 (479 female and 521 male). Data were collected over the course of 50 experimental sessions.

### Results

Table 1 presents the first few rows of the data:

First few rows of the survey data:

| session | sex | age | haircolor | university   | education    | atbless | intelligence | attractiveness | desirability |
|---------|-----|-----|-----------|--------------|--------------|---------|--------------|----------------|--------------|
| 1       | m   | 22  | brwn      | 3.Georgetown | 3.Master     | 3.Yes   | 1.Low        | 3.High         | 15           |
| 1       | m   | 19  | blond     | 2.Durham     | 1.HighSchool | 1.No    | 2.medium     | 2.medium       | 40           |

# 6.8 Principais formatos de saída

## html

```
output: html_document
```

Linguagem de marcação para construção de páginas web

## pdf

```
output: pdf_document
```

PDFs criados pelo RMarkdown utiliza LaTeX (pronuncia-se: Lah-tech or Lay-tech)

## Word (.xmls)

```
output: word_document
```

## OpenDocument Text(.odt)

```
output: odt_document
```

Dúvidas?

Muito obrigado pela atenção!



# Maurício Vancine

Contatos:

✉ [mauricio.vancine@gmail.com](mailto:mauricio.vancine@gmail.com)

🐦 [mauriciovancine](https://twitter.com/mauriciovancine)

🔗 [mauriciovancine.netlify.com](https://mauriciovancine.netlify.com)

Slides criados via pacote [xaringan](#) e tema [Metropolis](#)