

Comunicando seus resultados: criando apresentações com R

Meetup R-Ladies SP - N. 3

Beatriz Milz

21/11/2018



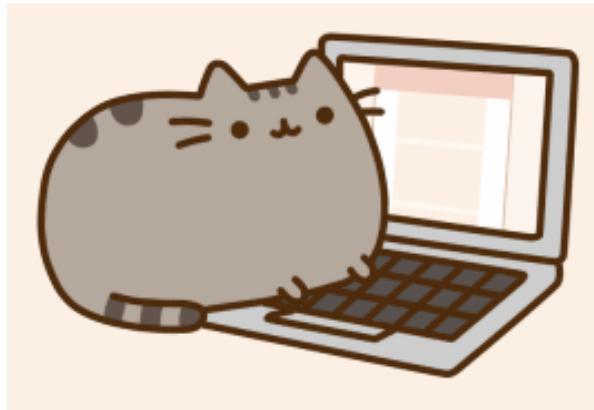
Sobre mim



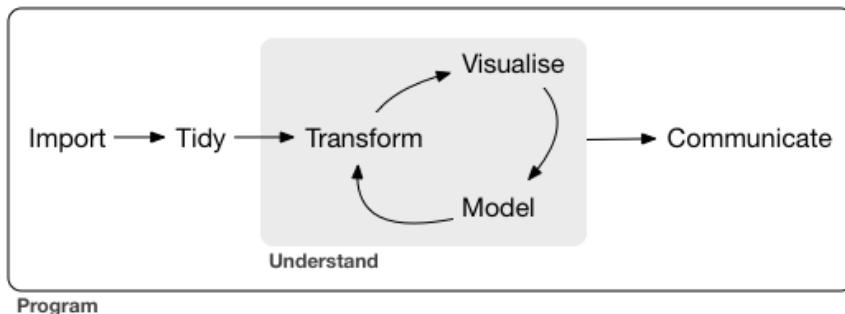
- Beatriz Milz, 25 anos, Diadema.
- Bacharel em Gestão Ambiental (EACH/USP).
- Mestre em Ciências no PPG-Análise Ambiental Integrada (UNIFESP/Diadema).
- Atualmente candidata ao Doutorado em Ciência Ambiental no PROCAM/IEE/USP.
- Aprendendo R desde agosto de 2018.
- Participa da comunidade R-Ladies São Paulo desde agosto de 2018.
- Email: beatriz.milz@hotmail.com
- Página: <https://beatrizmilz.github.io/>
- Github: @beatrizmilz (repositório de código)
- Twitter: @BeaMilz

Comunicando seus resultados: criando apresentações com R

- 1) Introdução sobre a linguagem de marcação markdown e sobre arquivos do tipo RMarkdown.
- 2) Prática sobre como criar uma apresentação utilizando o pacote Xaringan.
- 3) Hospedagem da apresentação utilizando o Github Pages.



Por que apresentações são importantes?



Fonte: <http://r4ds.had.co.nz/>

- Uma das formas de comunicar os resultados!
- Hoje iremos praticar como criar uma apresentação utilizando o R e o pacote Xaringan.

Pré-requisitos:

- R
- RStudio
- Pacote `Rmarkdown` * O R instala esse pacote automaticamente quando necessário.
- Pacote `Xaringan`
- Pacote `knitr`

Introdução sobre o Xaringan

- O xaringan é um pacote do R para criar apresentações usando o [remark.js](#) através do Rmarkdown.



- Mais sobre o xaringan:
 - [Apresentação da documentação](#)
 - [Livro R Markdown \(Capítulo 7\)](#)
 - [Remark.js Wiki](#)
 - [Xaringan Wiki](#)

Instalar o Xaringan

- Existem duas formas de instalar:

CRAN:

```
install.packages("xaringan")
```

Github:

```
devtools::install_github("yihui/xaringan")
```

Outras formas de criar apresentações:

- output: **beamer_presentation** -> beamer (pdf)
- output: **ioslides_presentation** -> ioslides (HTML)
- output: **slidy_presentation** -> slidy (HTML)

Obs: para separar entre um slide e outro, usar três *

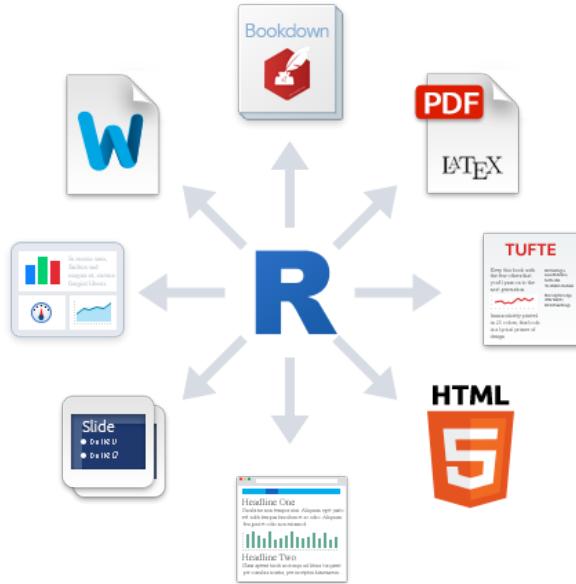
RMarkdown

Rmarkdown

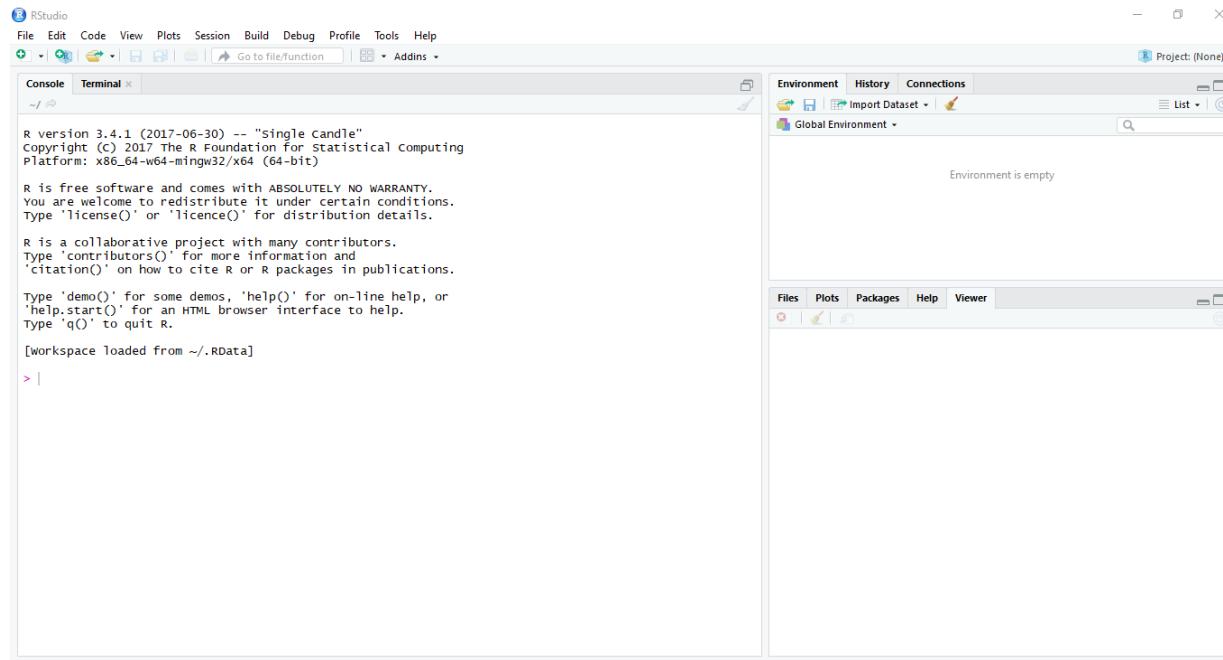
- É um tipo de arquivo que suporta códigos em R, texto, markdown e outros formatos.
- Apresenta os seus códigos, o resultado, e também os seus comentários.
- Arquivos **.Rmd** são reproduzíveis.
- São interessantes para trabalhar em equipe, para comunicar seus resultados, e para desenvolver seus códigos em R.
- O markdown é uma linguagem de marcação simples.
- [Rmarkdown Cheatsheet](#)

Rmarkdown

- Possibilita exportar diferentes tipos de arquivos.



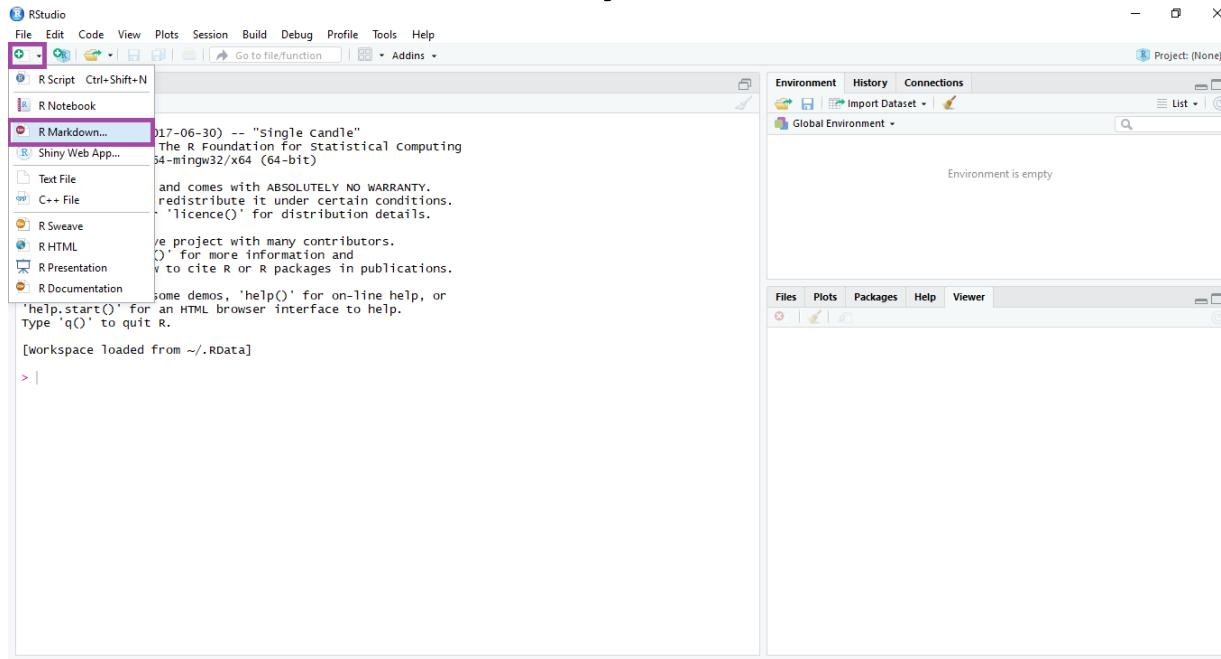
Inicie o RSTUDIO:



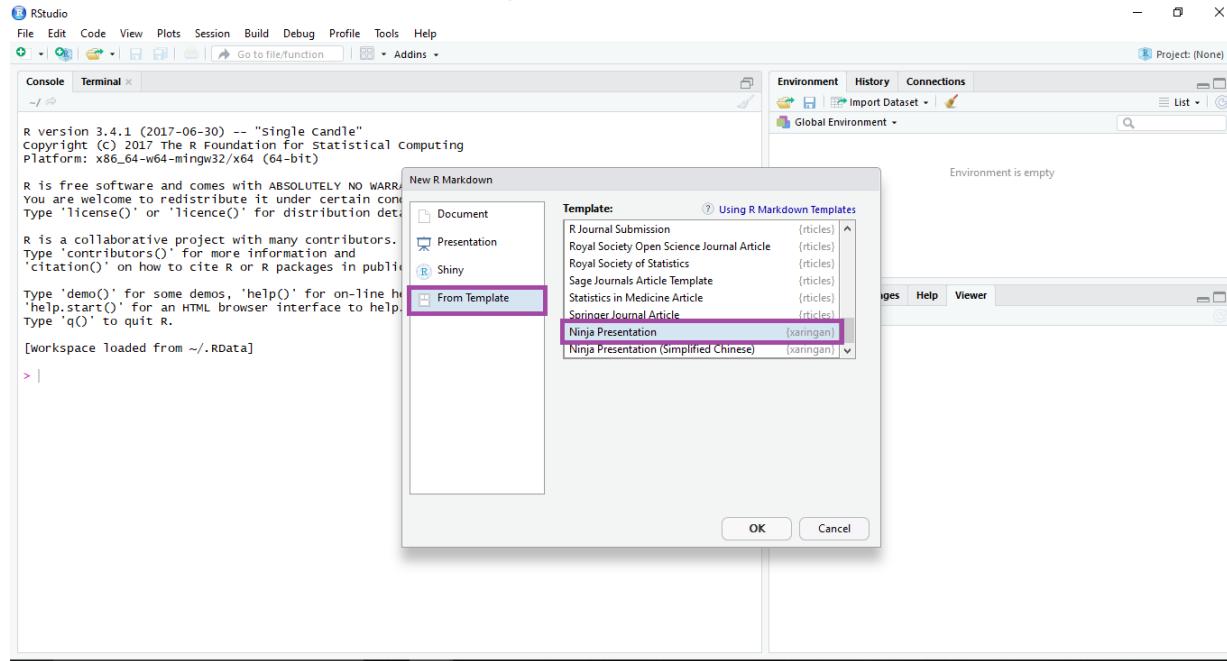
- Instale os pacotes:

```
install.packages("rmarkdown")
install.packages("knitr")
install.packages("xaringan")
```

Crie um novo arquivo Rmarkdown:

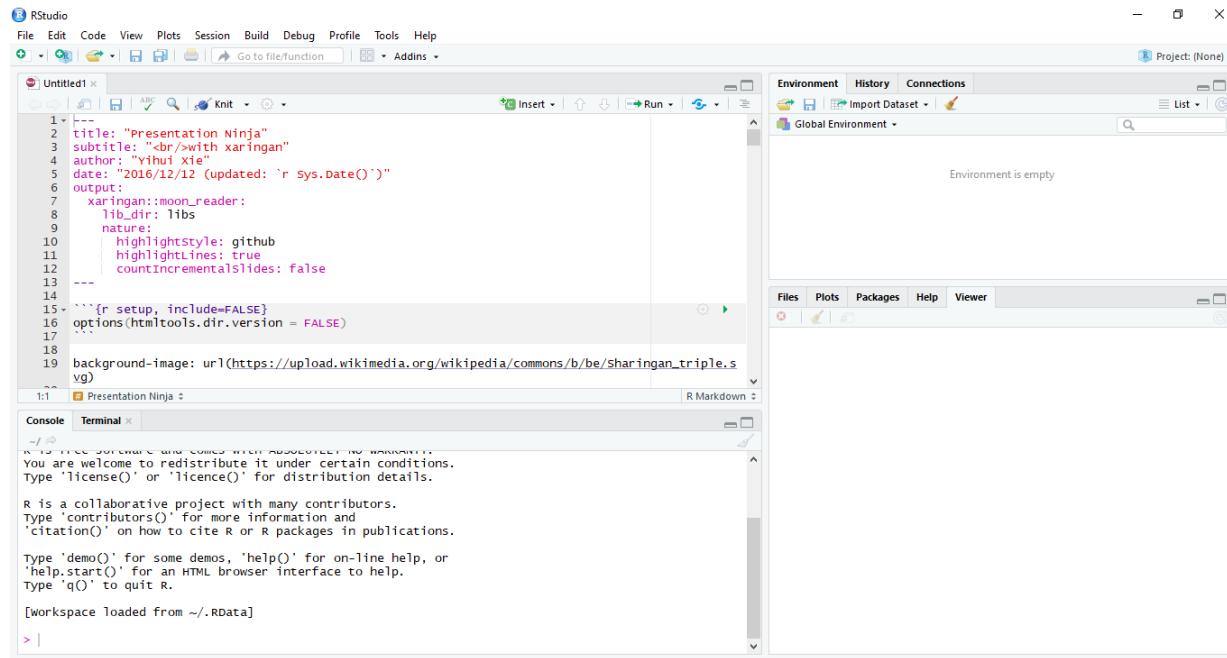


Selecione o template Ninja Presentation:



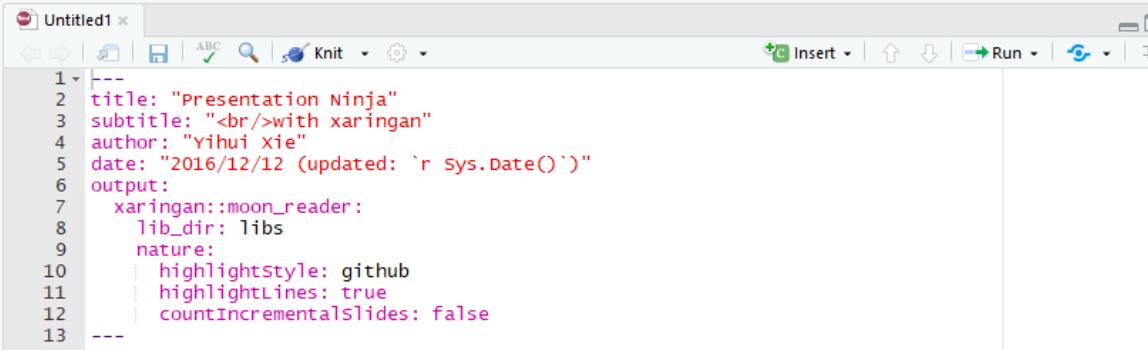
- Dica: para facilitar posteriormente, crie a sua apresentação com o nome de **index.Rmd**.

Será criado um RMarkdown com o Ninja Presentation:



yaml - metadados do documento:

- yet another markup language
- Controla muitas das configurações do documento.



The screenshot shows the RStudio interface with a code editor window titled "Untitled1". The code is a YAML document with the following content:

```
1 ---  
2 title: "Presentation Ninja"  
3 subtitle: "<br/>with xaringan"  
4 author: "yihui xie"  
5 date: "2016/12/12 (updated: `r sys.date()`)"  
6 output:  
7   xaringan::moon_reader:  
8     lib_dir: libs  
9     nature:  
10    | highlightstyle: github  
11    | highlightLines: true  
12    | countIncrementalSlides: false  
13 ---
```

- title: título
- subtitle: subtítulo
- author: autor
- date: data
- output: formato (Ex: html_document, pdf_document, etc). No caso do Xaringan, o output é **xaringan::moon_reader**
- Cuidado com a **identação**

Leia também: [Output Formats - R Markdown: The Definitive Guide](#)

Quais são os temas disponíveis?

```
names(xaringan:::list_css())
```

```
## [1] "chocolate-fonts"    "chocolate"          "default-fonts"  
## [4] "default"            "duke-blue"          "hygge-duke"  
## [7] "hygge"              "kunoichi"           "lucy-fonts"  
## [10] "lucy"               "metropolis-fonts" "metropolis"  
## [13] "middlebury-fonts"   "middlebury"         "ninjutsu"  
## [16] "rladies-fonts"      "rladies"            "robot-fonts"  
## [19] "robot"              "rutgers-fonts"     "rutgers"  
## [22] "shinobi"            "tamu-fonts"         "tamu"  
## [25] "uo-fonts"           "uo"
```

Alguns exemplos de disponíveis: [xaringan themes](#)

Colocar o tema RLadies

```
output:  
  xaringan::moon_reader:  
    css: ["rladies", "rladies-fonts"]
```

- Criado por/Made by: Alison Presmanes Hill
- Se você entende de CSS, é possível adaptar os temas e também criá-los :)

Exemplo do yaml:

The screenshot shows the RStudio interface with a YAML configuration file open. The file contains the following content:

```
1 ---  
2 title: "Meet up R ladies"  
3 subtitle: "testando o xaringan"  
4 author: "bia"  
5 date: "28/10/2018"  
6 output:  
7   xaringan::moon_reader:  
8     css: ["rladies", "rladies-fonts"]  
9     lib_dir: libs  
10    nature:  
11      highlightStyle: github  
12      highlightLines: true  
13      countIncrementalSlides: false  
14 ---  
15 ---  
16 ```{r setup, include=FALSE}  
17 options(htmltools.dir.version = FALSE)  
18 ---  
19 Hello world!
```

The 'Knit' button in the toolbar is highlighted with a purple box.

- Aperte o Knit e...

Resultado:

Meet up R ladies
testando o xaringan
bia
28/10/2018

Hello world!

1 / 2 2 / 2

- O knit exportará um arquivo do tipo html.
- Agora o tema da apresentação está configurado!
- Aperte H ou ? para conferir a lista de atalhos do sharingan.

DEU ERRO NO KNITR!

Calma! É bem comum dar erro ao compilar o arquivo pois **qualquer** código errado causa isso. Algumas coisas pra se observar:

- Yaml header - se algo estiver errado, vai dar erro ao compilar. Muito cuidado com a identação!
- Código nos chunks - teste o Run All Chunks e ver se dá erro em algum específico. Se descobrir, tente verificar o que há de errado no código (pode ser a falta de um parênteses, vírgula, entre outros).



Markdown

- O markdown é uma linguagem de marcação simples.
- [Rmarkdown Cheatsheet](#)
- Elaborada para ser fácil de ler, fácil de escrever.

Markdown

Sintaxe Markdown

Adicionar títulos

```
# Título 1  
## Título 2  
### Título 3
```

Título 1

Título 2

Título 3

- Rmarkdown Cheatsheet

- Inserir imagens

```
# !Logo do R](imagens/Rlogo.png)
```



- Rmarkdown Cheatsheet

- **Formatar textos**

Negrito: **colocar o texto entre dois asteriscos**.

Itálico: *colocar o texto entre astericos*.

Riscado: ~~colocar o texto entre dois tis~~

Itálico: *colocar o texto entre astericos*.

Negrito: **colocar o texto entre dois asteriscos**.

Riscado: ~~colocar o texto entre dois tis~~

- **Criar links**

[R](<https://www.r-project.org/>)

R

- Rmarkdown Cheatsheet

- Adicionar campos de código

```
`escreva um código aqui`
```

escreva um código aqui

- *Fórmulas**

```
$A = \pi * r^2$
```

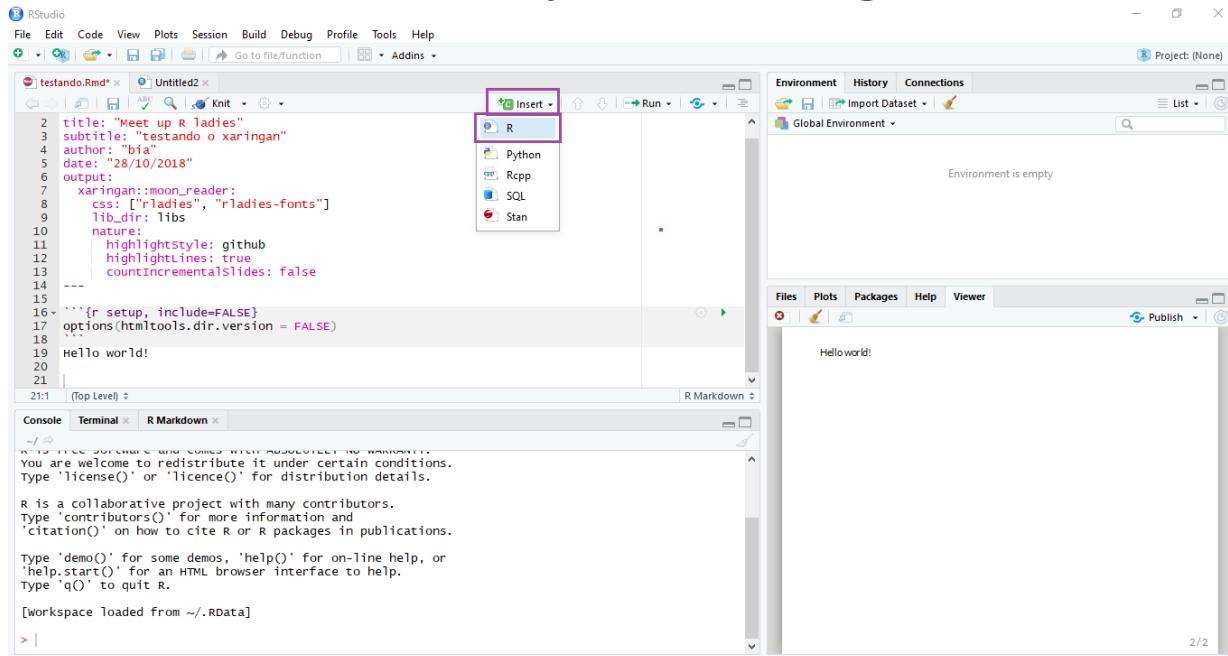
$$A = \pi * r^2$$

Elaborando a apresentação E explorando a cheatsheet

Quebra de slides



Adicionando campos de código - chunks



- Aceita código em R, Python, SQL.

Dica: Atalho para criar chunk: **Ctrl + Alt + I**.

Campos de código - chunks

- Crie um chunk para cada tarefa que for executar no R.
- O chunk tem várias opções interessantes referentes ao documento gerado:
 - `eval = FALSE` - Apresenta o código mas não "roda" ele. É útil para quando queremos apenas mostrar o código.
 - `include = FALSE` - "roda" o código, porém não mostra o código e nem o resultado no documento final. Ex: `library(tidyverse)` em apresentações que não apresentam os códigos.
 - `echo = FALSE` - não apresenta o código, mas apresenta o resultado.
 - `message = FALSE` e `warning = FALSE` - não mostra as mensagens de erro e warning no arquivo gerado.
 - `cache = TRUE` - mantém uma cópia do resultado do seu código no computador. É prático para usar quando o código

Consulta: [Knitr Options](#)

Adicionando campos de código - chunks

Exemplo: Adicionar um gráfico

```
```{r}
attach(mtcars)
plot(wt, mpg, main="Scatterplot Example",
xlab="Car Weight ", ylab="Miles Per Gallon ", pch=19)
```
```

Dica de atalho: Ctrl-Shift-Enter - O Rstudio executa o código e disponibiliza os resultados abaixo do código.

Adicionando um chunk de configuração geral

- O que for adicionado nessas configurações irá ser aplicado em todos os chunks do seu arquivo!

```
{r setup, include=FALSE}
```

```
knitr::opts_chunk$set(echo = TRUE, message = FALSE, warning = FALSE, fig.align =  
'center')
```

- São exemplos, você pode consultar quais [KnitR Options](#) são mais adequadas para seu projeto :)

Adicionando código R fora do chunk

```
`r`
```

Exemplo:

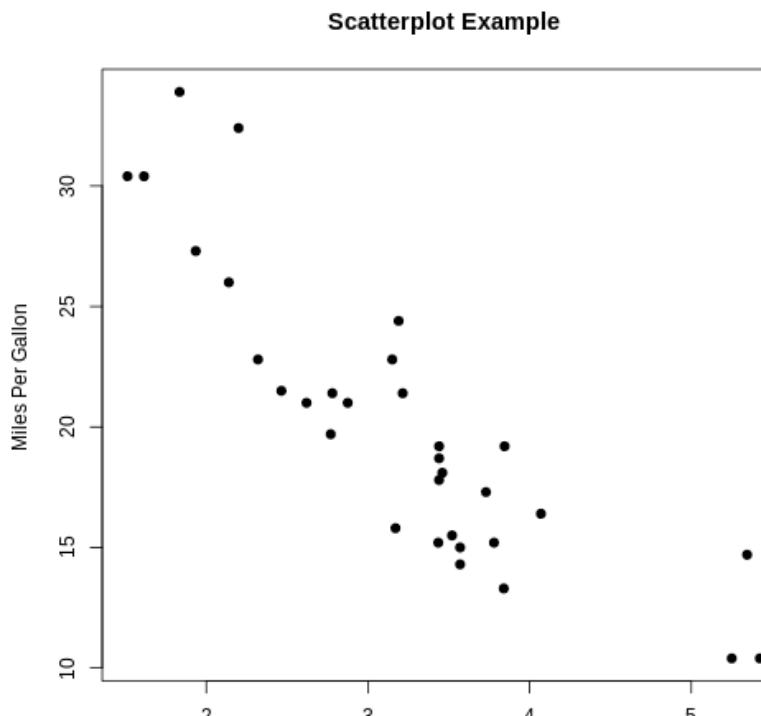
```
# O banco de dados mtcars possui `r nrow(mtcars)` carros cadastrados.
```

Exemplo: O banco de dados mtcars possui 32 carros cadastrados.

- Útil para relatórios!

Exemplo: Adicionar um gráfico

```
attach(mtcars)
plot(wt, mpg, main="Scatterplot Example",
xlab="Car Weight ", ylab="Miles Per Gallon ", pch=19)
```



Exemplo: Adicionar uma tabela

```
head(mtcars)
```

```
##          mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4     21.0   6 160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag 21.0   6 160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710    22.8   4 108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive 21.4   6 258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8 360 175 3.15 3.440 17.02  0  0    3    2
## Valiant       18.1   6 225 105 2.76 3.460 20.22  1  0    3    1
```

Exemplo: Adicionar uma tabela

```
DT::datatable(head(mtcars, 10),  
  fillContainer = FALSE, options = list(pageLength = 5))
```

Adicionar imagens usando o knitrR

Função: `knitr::include_graphics()`

Exemplo:

```
knitr::include_graphics("https://rladies.org/wp-content/uploads/2016/12/R
```



Adicionar imagens usando o knitR: Opções do chunk

- `out.width = "50%"` - Porcentagem do tamanho da imagem



KnitR Options

Adicionar figuras usando o knitr: Opções do chunk

- Alinhamento da figuras
- `fig.align = 'center', 'default', **'left'**, 'right'`



- `fig.align = 'center', 'default', 'left', **'right' **`



Adicionar figuras usando o knitr: Opções do chunk

- Alinhamento da figuras
- `fig.cap="plotting example"`



plotting example

Adicionar Fórmulas

Para escrever fórmulas, utilizar LaTeX. Existe duas formas:

- Centralizado - usar \$\$

```
$$celsius = kelvin - 273.15$$
```

$$celsius = kelvin - 273.15$$

- "Inline" - usar \$

```
$$ fahrenheit = celsius \times \frac{9}{5} + 32 $$
```

$$fahrenheit = celsius \times \frac{9}{5} + 32$$

Para outros símbolos matemáticos, [consulte aqui](#).

Exercício

1. Crie um arquivo Rmarkdown do tipo *presentation ninja*;
 2. Aplique o estilo R-ladies;
 3. Altere dados de autor, dê um título para o slide.
 4. Usando a cheatsheet, use sua imaginação: adicione títulos, textos, imagens, links, etc.
 5. Usando os chunks (campos de código), teste como ele funciona, e use algum código nele (gráfico, tabela, função, etc).
 6. Exporte a apresentação usando o Knitr.
 7. Abra o arquivo da sua apresentação em um navegador.
 8. Aprecie sua apresentação feita no R :D
- A melhor forma de aprender é ir testando!



Compartilhando a apresentação

Salvar em PDF

Usando o navegador Chrome ou no Chromium, imprima para arquivo (Ctrl + P).

Não dá certo com o Mozilla Firefox :(.

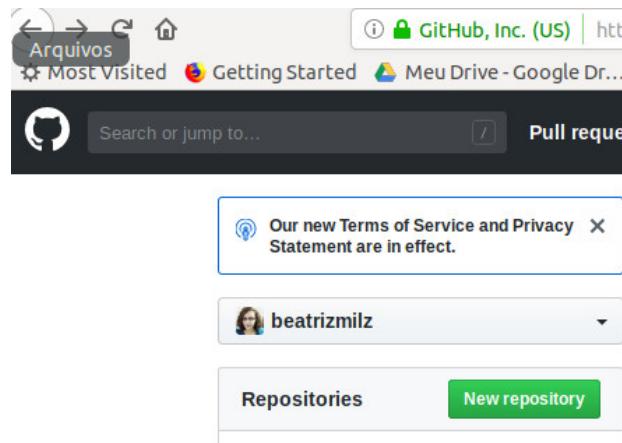
Consulta: <https://github.com/yihui/xaringan/wiki/Export-Slides-to-PDF>

Salvar em HTML

- O formato que utilizamos para exportar uma apresentação elaborada com o xaringan é **.html**. Para isso, é necessário apertar o botão Knit (mostrado anteriormente).
- É possível hospedar o seu arquivo de apresentação em sites como o [Github Pages](#) e [Netlify](#).

Compartilhando a apresentação via Github Pages

- 1) Crie uma conta gratuita no github: <https://github.com/>
- 2) Na sua conta do github, crie um novo repositório (New repository)



- OBS: Esse post do [@w_oliveiras](#) é sobre como usar git e github utilizando o terminal/linha de comando e pode ajudar!

Compartilhando a apresentação via Github Pages

3) Coloque um nome e descrição no novo repositório. Deixe como público.

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner



beatrizmilz

Repository name

teste



Great repository names are short and memorable. Need inspiration? How about [probable-telegram](#).

Description (optional)

teste



Public

Anyone can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with a README

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None

Add a license: None



Create repository

Compartilhando a apresentação via Github Pages

4) Repositório criado!

beatrilmilz / teste

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

teste Edit

Manage topics

1 commit 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

beatrilmilz Initial commit Latest commit 2abd3eb 3 minutes ago

README.md Initial commit 3 minutes ago

README.md

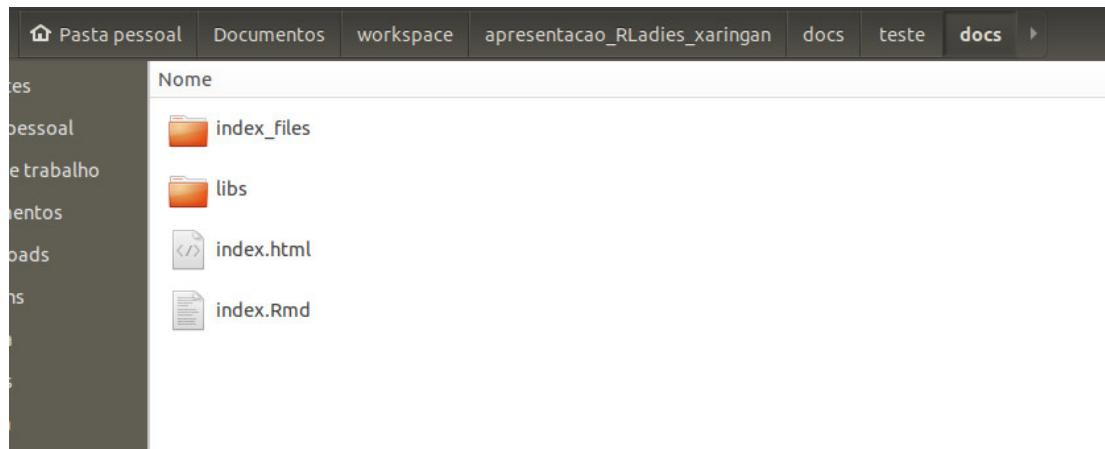
teste

teste

Compartilhando a apresentação via Github Pages

5) Githubpages e pasta DOCS.

- Para o github pages funcionar, você precisa adicionar o conteúdo em uma pasta chamada `docs`. Então, antes de fazer upload, no seu computador/workingdirectory/projeto, crie uma pasta com o nome `docs` e coloque todos os arquivos da sua apresentação nessa pasta.
- De preferência, a sua apresentação deve se chamar `index.html` (após compilado).



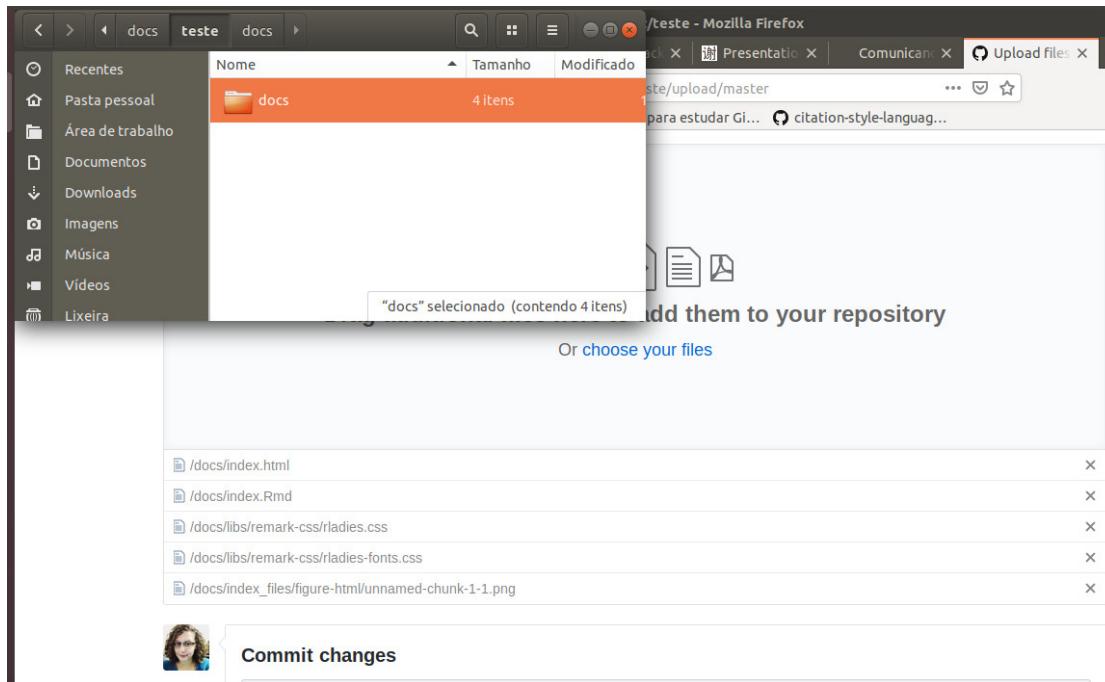
Compartilhando a apresentação via Github Pages

- Área de upload do seu repositório do github (essa não é a única forma de realizar upload, também é possível através do github desktop e pelo terminal de comando).

The screenshot shows a GitHub repository interface. At the top, there is a navigation bar with links for Code, Issues (0), Pull requests (0), Projects (0), Wiki, Insights, and Settings. Below the navigation bar, the repository name "teste /" is displayed. The main area features a large button with icons for file types (document, chart, code, etc.) and the text "Drag files here to add them to your repository" followed by "Or choose your files". At the bottom, there is a "Commit changes" section with a user profile picture, a "Commit changes" button, an "Add files via upload" input field, and a text area for "Add an optional extended description...".

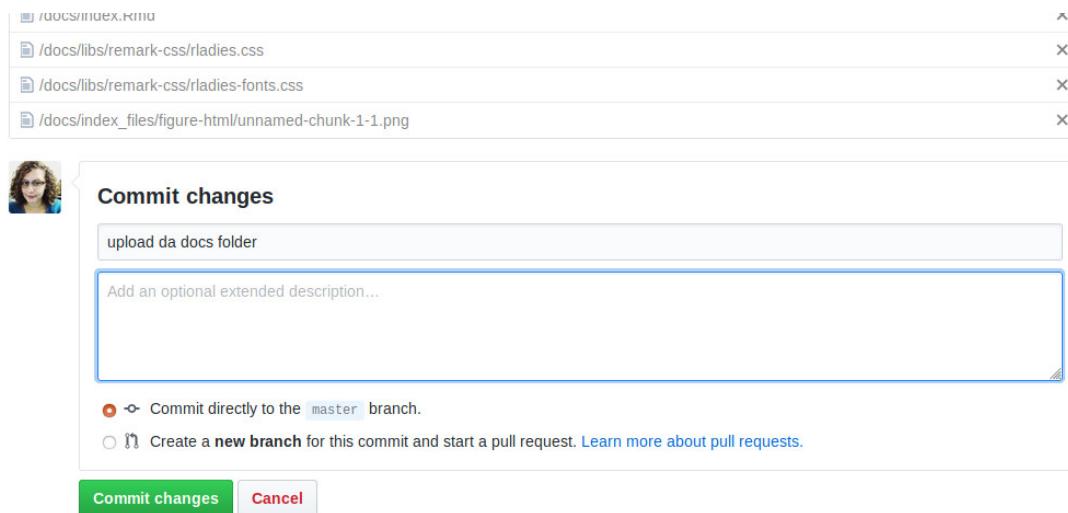
Compartilhando a apresentação via Github Pages

6) Faça o upload da pasta docs INTEIRA no seu repositório do github.



Compartilhando a apresentação via Github Pages

7) Coloque uma mensagem no commit, para deixar registrado que alteração foi feita.



The screenshot shows a GitHub commit dialog. At the top, there is a list of staged files:

- /docs/index.html
- /docs/libs/remark-css/radies.css
- /docs/libs/remark-css/radies-fonts.css
- /docs/index_files/figure-html/unnamed-chunk-1-1.png

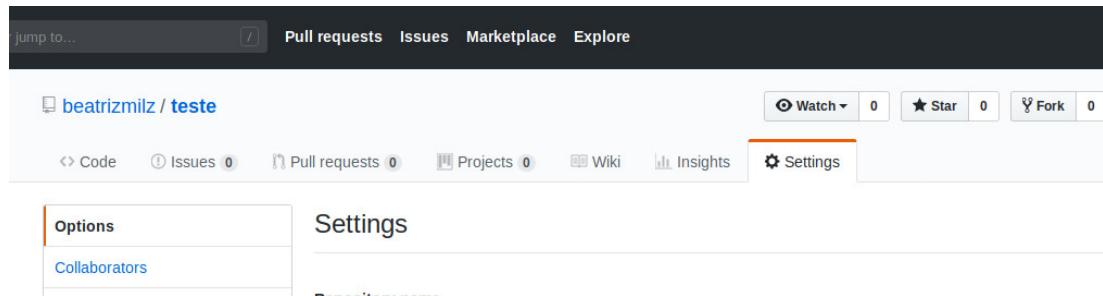
Below the file list is the "Commit changes" form. It includes a user profile picture, a title field ("upload da docs folder"), a large text area for an optional extended description, and two radio button options for choosing the branch:

- Commit directly to the `master` branch.
- Create a **new branch** for this commit and start a pull request. [Learn more about pull requests.](#)

At the bottom are two buttons: a green "Commit changes" button and a red "Cancel" button.

Compartilhando a apresentação via Github Pages

8) No seu repositório, clique em Settings...



Compartilhando a apresentação via Github Pages

8) ... e na seção Github Pages, selecione **master branch /docs folder**:

GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Source

GitHub Pages is currently disabled. Select a source below to enable GitHub Pages for this repository. [Learn more.](#)

master branch /docs folder ▾

Save

Theme Chooser

Select a theme to publish your site with a Jekyll theme using the `master branch /docs folder`. [Learn more.](#)

Choose a theme

Resultado: <https://beatrizmilz.github.io/teste/>

Agora é só enviar o link das suas apresentações lindas!

Dica: Criar um URL mais curto com o [Bitly](#)

bit.ly/RLadiesSP3



Referências

- R for Data Science - Hadley Wickham & Garret Grolemund - Gratuito, em inglês.
- Xaringan
 - Xaringan no Github
 - Documentação
- Themes:
- Rladies:
 - Alison Presmanes Hill
- Kunoichi:
 - Documentação Kunoichi
 - Kunoichi no Github
 - Emi Tanaka!
- Material de estudo:
 - R Markdown: The Definitive Guide



Obrigada!

Apresentação elaborada com [Xaringan](#) e [Kunoichi](#).

Agradecimentos especiais:

[Bruna Wundervald](#) pela revisão e sugestões.

[Julio Trecenti](#), [Curso-R](#) e [13 aMostra de Estatística - IME USP](#) pelo mini-curso de Rmarkdown.

[DataCamp](#) - Reporting with R Markdown Course.

[Maria Marinho](#) pelas sugestões e ajuda com a versão em pdf.

