



Estatística Computacional

Universidade Federal da Bahia

Gilberto Pereira Sassi

Tópico 4

Pacotes que iremos usar na semana 6

```
library(readxl)
library(readODS)
library(writexl)
library(ggthemes)
library(lvplot)
library(glue)
library(tidyverse)
```



Geração de documentos no R

rmarkdown

(bookdown, blogdown, pkgdown)

Sobre o markdown

- Criado em 2004 por [John Gruber](#)
- Criado originalmente para formatar textos em web
- **Filosofia:** fácil de escrever, fácil de ler o código e permitir edição em forma de prosa – foco no conteúdo e não nos detalhes da linguagem (como pode ser o caso do HTML)
- **Inspiração:** formatação permitida ao escrever e-mails
- **Principais vantagens:**
 - portátil
 - permite estilizar arquivos de texto e não depende de versões de programas como power point da Microsoft
 - extremamente popular e usados em diversos aplicativos como [notion.so](#), [Hugo – Static Site Generator\(SSG\)](#) e [Gatsby – SSG](#)



markdown na linguagem R

- **rmarkdown**: permite gerar documentos de alta qualidade incluindo automaticamente tabelas, gráficos e sintaxe de forma simples usando `markdown`, além disso podemos incluir fórmulas matemáticas usando código latex. Permite gerar documentos no formato `.docx`, `.html`, `.ppt`, apresentação beamer e `.pdf` (usando código latex). Referências:
 - introdução ao **rmarkdown**: [rmarkdown](#)
 - referência ao **rmarkdown**: [documentação rmarkdown](#)
- **pkgdown**: extensão do pacote `rmarkdown` que permite criar um site com a documentação de pacotes no R. Referências:
 - referência ao **pkgdown**: [documentação pkgdown](#)
- **blogdown**: extensão do pacote `rmarkdown` que usa o SSG [Hugo](#) que permite usar a linguagem R para criar blogs e websites
 - referências ao **blogdown** [documentação blogdown](#)
- **bookdown**: extensão do pacote `rmarkdown` que permite construir artigos e livros acadêmicos.
 - referência ao **bookdown**: [documentação bookdown](#)

Aqui vamos focar apenas no `rmarkdown`.



markdown: sintaxe básica

Para detalhes da linguagem markdown, consulte [documentação markdown](#).

Definindo seções, subseções e afins

Tabela *de-para* entre markdown, HTML e latex.

markdown	html	latex
# texto	<h1>texto</h1>	\section{texto}
## texto	<h2>texto</h2>	\subsection{texto}
### texto	<h3>texto</h3>	\subsubsection{texto}
#### texto	<h4>texto</h4>	\paragraph{texto}
##### texto	<h5>texto</h5>	\subparagh{texto}
##### texto	<h6>texto</h6>	



markdown: sintaxe básica

formatação de texto

- texto em negrito: **texto em negrito** ou texto em negrito
- *texto em itálico*: *texto em itálico* ou **texto em itálico**
- ***texto em itálico e negrito***: ***texto em itálico e negrito*** ou ***texto em itálico e negrito***
- código inline: `código inline` (crase)`
- citação em bloco (equivalente a `<blockquote></blockquote>`)

```
> ##### Texto
>
> Uma citação qualquer.
>
>> Gilberto Pereira Sassi
```

Texto

Uma citação qualquer.

Gilberto Pereira Sassi



markdown: sintaxe básica

listas

- **listas não ordenadas:** equivalente a `\begin{itemize} \end{itemize}` no latex e a tag ``. Podemos usar `*`, `+` e `-`

Sintaxe markdown

```
* primeiro item  
* segundo item
```

Resultado

- primeiro item
- segundo item

- **listas ordenadas:** equivalente a `\begin{enumerate} \end{enumerate}` no latex e a tag ``. Podemos usar `1.`, `i.` e `a.`

Sintaxe markdown

```
1. primeiro item  
1. segundo item
```

Resultado

1. primeiro item
2. segundo item



markdown: sintaxe básica

Listas incrementais em apresentações:

- lista não ordenada incremental: >-
- lista ordenada incremental: 1.

endereços eletrônicos, imagem e quebra de linha

`\href{}` é uma função do pacote `hyperref` do latex.

markdown	html	latex
<code>![texto alternativo]</code> <code>(/caminho/até/arquivo.png)</code>	<code></code>	<code>\includegraphics[]{}</code>
<code><email@exemplo.com></code>	<code>email@exemplo.com</code>	<code>\href{mailto:email@exemplo.com}{email@exemplo.com}</code>
<code>
</code>	<code>
</code>	<code>*</code>
<code>[texto]</code> <code>(https://endereço/index.html)</code>	<code> texto </code>	<code>\href{https://endereço/index.html}{texto}</code>



markdown: sintaxe básica

Tabela

- cabeçalho de tabela é separado por três ou mais hífens - - - - e as colunas são separadas por pipe |
 - Alinhamentos: : - - - - alinhamento à esquerda; - - - - : alinhamento à direita; : - - - - : alinhamento ao centro

Sintaxe markdown

```
|coluna 1|coluna 2|coluna 3|
|:----|:----:|-----:|
|texto 1|texto 2|texto 3|
```

Resultado

coluna 1	coluna 2	coluna 3
texto 1	texto 2	texto 3

Subescrito, superescrito, *riscado* e link para seções

- Rótulo para seções: considere `### Seção 1 {#rotulo-secao-1}`. Podemos referenciar esta seção com `[seção 1](#rotulo-secao-1)`
- Texto sobrescrito `^texto^`. Exemplo: `x^2^`. Resultado: x^2
- Texto subescrito `~texto~`. Exemplo: `H~2~O`. Resultado: H_2O
- Texto *riscado* `~~texto~~`. Exemplo: `~~exemplo~~`. Resultado: ~~exemplo~~
- Nota de rodapé (não funciona para latex-beamer): `^[nota de rodapé]`



markdown no R: rmarkdown

- **Objetivo:** simplificar a elaboração de relatórios, artigos e livros
- Principais vantagens:
 - Documentos facilmente reprodutível
 - Código fácil de ler (markdown)
 - Permite gerar documentos dinamicamente com o argumento `params`
- O código fonte dos documentos em `rmarkdown` tem extensão `.Rmd` inclui as seguintes partes:
 - Cabeçaho **YAML**: primeira parte do documento delimitada por três hífen, e include as formatações do documento
 - **chunk**: lugar onde colocamos o nosso código R com a seguinte sintaxe:

```
```${r}  
1 + 1
log2(10)
```
```

- **inline chunk**: permite colocar código R no meio de uma linha. ``r 1 + 1``
- texto escrito e formatado usando `markdown`
- Equações usando código latex



YAML (Yeat Another Markup Language)

- O cabeçalho **YAML** começa na primeira linha e está demilitada entre --- e ---

```
---  
yaml header  
---
```

- Opções de **.Rmd** no cabeçalho **YAML**:
 - **title**: string indicando o título do documento
 - **date**: string indicando a data de criação do documento
 - **font-size**: tamanho da fonte
 - **bibliography**: arquivo bibtex **.bib** com as referências
 - **biblio-style**: estilo bibtex
 - **output**: formato do documento gerado. Os mais usados: **bookdown::pdf_document2**, **bookdown::word_document2**, **bookdown::beamer_presentation2** e **bookdown::ioslides_presentation2**



YAML (Yeat Another Markup Language)

Opções para output

- **highlight**: coloração e destaque do código. Valores possíveis: *tango*, *pygments*, *kate*, *monochrome*, *espresso*, *zenburn*, and *haddock*
- **fig_caption**: true ou false. true indica que as figuras serão numeradas
- **toc**: true ou false. true indica que o documento terá sumário
- **toc_depth**: qual a profundidade para incluir no sumário. Por ex: se **toc_depth**: 3, então o `rmarkdown` inclui no sumário todos os textos em #, ## e ###

Apenas para beamer

- **theme**: tema para apresentação em beamer. Consulte a matriz de temas no endereço eletrônico: [matriz de temas](#)
- **colortheme**: cor do tema para apresentação em beamer. Consulte a matriz de temas no endereço eletrônico: [matriz de temas](#)

Apenas para .docx

- **reference_docx**: documento .docx que serve de template para o `rmarkdown`



YAML (Yeat Another Markup Language)

Opções para output

opções para arquivos gerados pelo latex

- `includes:`
 - `in_header`: inclui código entre `\documentclass[]{} e \begin{document}`
 - `before_body`: inclui código imediatamente **depois** do `\begin{document}`
 - `after_body`: inclui código imediatamente **antes** do `\end{document}`
- `keep_text`: `true` ou `false`. `true` indica que o código latex será salvo em arquivo separado



opções apenas bookdown::pdf_document2

- documentclass: tipo de documento – por exemplo article, memoir ou IEEEtran
- classoptions: opções para o comando \documentclass[opções]{} – por exemplo oneside



Fórmulas matemáticas: `rmarkdown`

- fórmulas *inline*: envolva as fórmulas em código latex com `\(\)` ou `$ $`. **Exemplo:** `\(\frac{x}{y}\)`. **Resultado:** $\frac{x}{y}$.
- fórmula centralizada: envolva as fórmulas em código latex com `$$ $$` (ou use `\begin{equation} \end{equation}`)

Trechos de código latex que você pode vir a usar

- fração: `\frac{x}{y}`. **Exemplo:** `\(\frac{x}{y}\)`. **Resultado:** $\frac{x}{y}$
- somatório: `\sum_{k=a}^b`. **Exemplo:** `\(\sum_{k=a}^b x_k\)`. **Resultado:** $\sum_{k=a}^b x_k$
- potenciação: `x^a`. **Exemplo:** `x^a`. **Resultado:** x^a
- subescrito: `y_a`. **Exemplo:** `y_a`. **Resultado:** y_a
- integração: `\int_a^b`. **Exemplo:** `\int_a^b f(u)du`. **Resultado:** $\int_a^b f(u)du$
- Para mais fórmulas, consulte: [referência latex](#)
- Você usar o aplicativo `destexify`: [detexify](#)



Parâmetros para bloco de códigos (ou **chunk**)

- Bloco de código: começa com três crases e termina com três crases
 - O `rmarkdown` usa `label` para salvar localmente figuras e arquivos temporários dentro do projeto

```
```${r label, <opções>}  
bloco de códigos
```
```

- `eval`: parâmetro lógico. `TRUE` indica que o código não será processado
- `echo`: parâmetro lógico. `TRUE` indica que o código do *chunk* será incluído no documento
- `error`: parâmetro lógico. `TRUE` indica que o código (com erro de programação) será incluído no documento e a mensagem de erro também será incluída no documento
- `message`: parâmetro lógico. `FALSE` indica que mensagens geradas pelo código não serão incluídas no documento
- `warnings`: parâmetro lógico. `FALSE` indica que mensagens de `warnings` geradas pelo código não serão incluídas no documento
- `cache`: parâmetro lógico. `TRUE` indica que o `rmarkdown` vai salvar o resultado do processamento em uma pasta dentro do projeto para evitar o processamento desse código toda vez que você construir o documento com `rmarkdown::render()` (ou `ctrl+shift+k` no *rmarkdown*)



Parâmetros para bloco de códigos (ou **chunk**)

- `fig.align`: alinhamento da figura. Uma das opções: `left`, `right` ou `center`
- `fig.cap`: legenda da figura
- `out.height`: altura da figura (sugiro usar porcentagem)
- `out.width`: largura da figura (sugiro usar porcentagem)



Referência cruzada (usando **bookdown**)

Equações (não funciona bem com arquivos .html)

- Colocamos (`\#eq:label`) dentro de `\begin{equation}` `\end{equation}` ou `\begin{align}` `\end{align}`
- Para referenciar no texto, use `\@ref(eq:label)`

Figuras

- Cabeçalho no chunk: `{r label, fig.cap = "Legenda da figura."}`
- Para referência no texto, use `\@ref(fig:label)`

Teoremas, definições e afins

- Criamos um chunk substituindo o `r` por algum ambiente no cabeçalho: `{ambiente, label = "label", name = "Nome do ambiente"}`
- Para referencia no texto, use `\@ref(prefixo:label)`



Referência cruzada (usando **bookdown**)

Teoremas, definições e afins

| ambiente | Nome impresso no texto | prefixo para usar em \@ref() |
|--------------------|------------------------|------------------------------|
| theorem | Teorema | thm |
| lemma | Lema | lem |
| corollary | Corolário | cor |
| proposition | Propósito | prp |
| conjecture | Conjuntura | cnj |
| definition | Definição | def |
| example | Exemplo | exm |
| exercise | Exercício | exr |

