

O que é Markdown?

Sintaxe do Markdown

Cabeçalhos

Itálico

Negrito

Listas não-ordenadas

Listas ordenadas

Sub-listas

Links

Imagens

Parágrafo

Códigos

Tabelas

Equações matemáticas

Escrevendo um documento em Markdown

Pandoc

Markdown e Pandoc

Code ▼

Fernando Mayer & Walmes Zeviani

O que é Markdown?

Segundo o próprio criador da linguagem:

Markdown is a text-to-HTML conversion tool for web writers. Markdown allows you to write using an easy-to-read, easy-to-write plain text format, then convert it to structurally valid XHTML (or HTML).

John Gruber

- Markdown (<http://daringfireball.net/projects/markdown>) é uma linguagem de marcação (https://pt.wikipedia.org/wiki/Linguagem_de_marcação/) simples para escrever textos
- O texto pode ser lido sem nenhum processamento, ou seja, da maneira como está escrito

- Outras linguagens de marcação como HTML e $LATEX$ requerem um grande número de tags para formatar o texto, muitas vezes dificultando a leitura do código-fonte
- A proposta do Markdown é que o escritor se concentre no texto e não na formatação
- Pode ser convertido para **vários outros formatos** além de HTML

Sintaxe do Markdown

A sintaxe do Markdown é muito simples, e pode ser resumida da seguinte forma:

Cabeçalhos

```
# Título
## Sub-título
### Sub-sub-título
```

Itálico

```
*Este texto aparecerá em itálico.*
```

Este texto aparecerá em itálico.

Negrito

```
**Este texto aparecerá em negrito.**
```

Este texto aparecerá em negrito.

Listas não-ordenadas

```
- Primeiro item
- Segundo item
- Terceiro item
```

- Primeiro item
- Segundo item
- Terceiro item

Listas ordenadas

```
1. Primeiro item
2. Segundo item
3. Terceiro item
```

1. Primeiro item
2. Segundo item
3. Terceiro item

Sub-listas

Utilize 4 espaços para criar uma sub-lista:

- ```
1. Primeiro item
 - Um sub-item
 - Outro sub-item
2. Segundo item
3. Terceiro item
```

- ```
1. Primeiro item
  o Um sub-item
  o Outro sub-item
2. Segundo item
3. Terceiro item
```

Links

Links para endereços Web podem ser inseridos com `[texto](link)` :

```
O criador do conceito de "literate programming" foi
[Donald Knuth](https://en.wikipedia.org/wiki/Donald_Knuth).
```

O criador do conceito de "literate programming" foi Donald Knuth
(https://en.wikipedia.org/wiki/Donald_Knuth).

```
Devemos instalar o pacote [knitr](http://yihui.name/knitr) para poder
usar o R Markdown.
```

Devemos instalar o pacote knitr (<http://yihui.name/knitr>) para poder usar o R Markdown.

Imagens

Para inserir uma imagem, a sintaxe é a mesma de inserir um link, mas com uma exclamação (!) na frente: `![texto](imagem)` .

O link para a imagem pode ser um endereço Web:

```
![Logo do R](http://developer.r-project.org/Logo/Rlogo-5.png)
```



Logo do R

Ou um endereço local:

```
![Logo do Markdown](img/markdown.png)
```



Logo do Markdown

Parágrafo

Para criar parágrafos basta pular uma linha:

```
O criador do conceito de "literate programming" foi  
[Donald Knuth](https://en.wikipedia.org/wiki/Donald_Knuth).
```

```
Devemos instalar o pacote [knitr](http://yihui.name/knitr) para poder  
usar o R Markdown.
```

O criador do conceito de “literate programming” foi Donald Knuth
(https://en.wikipedia.org/wiki/Donald_Knuth).

Devemos instalar o pacote knitr (<http://yihui.name/knitr>) para poder usar o R Markdown.

Códigos

Para apresentar códigos na própria linha, colocamos o texto entre duas crases (` `):

```
Para gerar números aleatórios de uma distribuição normal no R, use a  
função `rnorm()`.
```

Para gerar números aleatórios de uma distribuição normal no R, use a função `rnorm()` .

Para apresentar blocos de código, coloque o texto entre três crases seguidas (```) no início e no final. O bloco

```
```\n\nx <- rnorm(n = 10, mean = 100, sd = 5)\nhist(x, main = "")\n```
```

Irá gerar

```
x <- rnorm(n = 10, mean = 100, sd = 5)\nhist(x, main = "")
```

Note que esse código não será interpretado, ele apenas será mostrado no texto. Esse será o papel do R aqui mais adiante!

# Tabelas

Tabelas podem ser escritas da seguinte forma:

Caracter	Permissão
-----	-----
`r`	Permissão de leitura (*read*)
`w`	Permissão de escrita (*write*)
`x`	Permissão de execução (*execute*)
`-`	Permissão desabilitada

Para gerar o seguinte resultado:

Caracter	Permissão
r	Permissão de leitura ( <i>read</i> )
w	Permissão de escrita ( <i>write</i> )
x	Permissão de execução ( <i>execute</i> )
-	Permissão desabilitada

## Equações matemáticas

Equações matemáticas podem ser escritas em formato *L<sup>A</sup>T<sub>E</sub>X*. A página HTML resultante irá renderizar as equações através do MathJax (<http://www.mathjax.org>).

Equações na própria linha podem ser inseridas entre `$` :

Um modelo de regressão linear simples: `$Y = \beta_0 + \beta_1 x + \epsilon_n$`.

Um modelo de regressão linear simples:  $Y = \beta_0 + \beta_1 x + \epsilon$ .

Equações podem ser exibidas entre `$$` para ficarem centralizadas em uma linha própria:

```
$$
f(x;\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}}
e^{\left\{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right\}}
$$
```

$$f(x; \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

## Escrevendo um documento em Markdown

Um documento Markdown possui a extensão `.md` (embora não seja a única possível).

Veja o arquivo de exemplo Exemplo1.md (exemplos/Exemplo1.md):

```
Um documento em Markdown
```

```
Sobre o Markdown
```

O Markdown é uma linguagem de marcação muito simples, desenvolvida por John Gruber.

A ideia básica por trás da linguagem é fazer com que o escritor se preocupe mais com o **conteúdo** do texto do que com a **formatação**.

```
Mais um título
```

Aqui vamos tentar descrever uma análise.

```
Simulando variáveis aleatórias
```

No R podemos simular valores de uma distribuição normal padrão através da função `rmnorm()`.

Seja  $X \sim \text{N}(0,1)$ , então para gerar 30 valores dessa variável aleatório normal, fazemos

```
```\n(x <- rmnorm(30))\n```
```

Com o resultado dessa simulação, podemos calcular a média e a variância dessa VA X para conferir se o resultado fica próximo de 0 e 1, respectivamente.

Também podemos fazer um histograma dessa VA X simulada

```
```\nhist(x)\n```
```

Para converter um documento Markdown em HTML (ou outro formato) é necessário um **conversor**. O conversor padrão do Markdown é escrito em Perl, e pode ser integrado em diversas ferramentas, mas não é apropriado para usuários comuns. Para testar a conversão do documento, copie e cole na página do Dingus (<http://daringfireball.net/projects/markdown/dingus>).

## Pandoc

O Pandoc (<http://pandoc.org/>) é um conversor extremamente versátil, capaz de converter diversos formatos, incluindo Markdown para HTML.

Se o Pandoc estiver instalado no seu sistema (Unix) é possível converter o documento na linha de comando (shell) com

```
pandoc -f markdown -t html Exemplo1.md -o Exemplo1.html
```

O pacote `knitr` possui a função `pandoc()` que é um *wrapper* para executar o programa `pandoc` no sistema.

[Hide](#)

```
pandoc(input = "exemplos/Exemplo1.md", format = "html")
```

```
executing pandoc -t html -o exemplos/Exemplo1.html 'exemplos/Exemplo1_utf8.md'
```

```
[1] "exemplos/Exemplo1.html"
```

Em ambos os casos, o resultado pode ser visualizado ao abrir o arquivo `Exemplo1.html` (`exemplos/Exemplo1.html`) no navegador.



([https://creativecommons.org/licenses/by-nc-sa/4.0/deed.pt\\_BR](https://creativecommons.org/licenses/by-nc-sa/4.0/deed.pt_BR))

Este conteúdo está disponível por meio da Licença Creative Commons 4.0