

Miscelânea



Gráficos interativos e estáticos

Gráficos interativos têm o poder de engajar mais, por conta do efeito *voosh*. Todo ser humano gosta de interagir com aquilo que está analisando.

No entanto, isso vem com um custo: não são todos os documentos capazes de processar visualizações dinâmicas. Em particular, PDF, Word e PPT não rodam esses conteúdos.

Nessa aula, vamos visitar tanto o mundo estático quanto dinâmico, para que você saiba por onde começar quando receber um novo desafio.

Atividade

Vamos construir htmlwidgets nos scripts!



Ao RStudio:

htmlwidgets

htmlwidgets são bibliotecas de visualização JavaScript encapsuladas em pacotes de R. Elas nos permitem usar diversas ferramentas JavaScript diretamente do R, adicionando algumas poucas linhas de código em nosso script.

Usando htmlwidgets, conseguimos construir tabelas, gráficos, mapas e muito outras visualizações interativas e naturalmente bonitas.

[Clique aqui](#) para acessar uma lista completa de todos os htmlwidgets disponíveis.

Tabelas com reactable

O pacote `reactable` nos permite criar tabelas interativas baseadas na biblioteca [React Table](#).

[Clique aqui](#) para acessar o tutorial completo do pacote `{reactable}`.

A interatividade dos `htmlwidgets` não depende de uma sessão R rodando por trás. Você pode utilizá-los em qualquer documento `.html`.

```
reactable::reactable(mtcars, compact = TRUE, defaultPageSize = 4,  
                      striped = TRUE)
```

	mpg	cyl	disp	hp	drat	wt					
Mazda RX4	21	6	160	110	3.9	2.62					
Mazda RX4 Wag	21	6	160	110	3.9	2.875					
Datsun 710	22.8	4	108	93	3.85	2.32					
Hornet 4 Drive	21.4	6	258	110	3.08	3.215					
1–4 of 32 rows			Previous	1	2	3	4	5	...	8	Next

Pacotes alternativos para tabelas

A seguir, uma lista de pacotes/ funções alternativos que trazem soluções para visualização de tabelas.

- `knitr::kable()`: não é um `htmlwidget` (não possui interatividade), mas é uma solução para formatar tabelas quando não precisamos que elas sejam interativas. Funciona em conjunto com o pacote `{kableExtra}`.
- `DT::datatable()`: outro `htmlwidget` para criar tabelas interativas. Funciona tal como o `reactable()`, mas um pouco mais burocrático para formatar as tabelas. Baseado na biblioteca JavaScript [DataTables](#).

Tutoriais

- [Tutorial kable e kableExtra](#)
- [Tutorial DT](#)

Gráficos com plotly

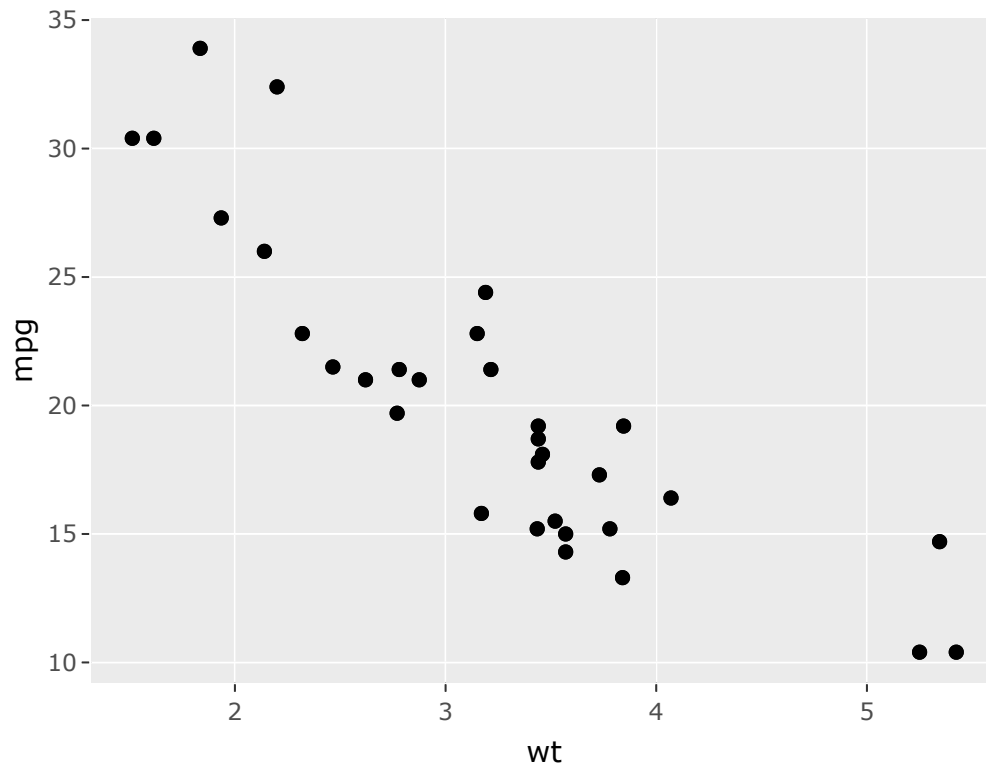
O pacote `plotly` nos permite criar gráficos interativos baseados na biblioteca `Plotly` (construída em `D3`).

Embora seja possível criar um `plotly` do zero usando a função `plot_ly()`, um jeito muito eficiente de utilizar essa biblioteca é criar um `ggplot` e então utilizar a função `ggplotly()`. Veja o exemplo a seguir.

Tutoriais

- [Tutorial plotly](#)
- [Interactive web-based data visualization with R, plotly, and shiny](#)


```
p <- ggplot(mtcars) +  
  geom_point(aes(x = wt, y = mpg))  
plotly::ggplotly(p, height = 400)
```



Pacotes alternativos

A seguir, uma lista de pacotes/ funções alternativos que trazem soluções para visualização de gráficos.

- `highcharter::highcharter()`: pacote gráfico baseado na biblioteca JavaScript [Highcharts](#). A biblioteca Highcharts é gratuita apenas para fins educacionais e não lucrativos (exceto órgãos governamentais). Para outros usos, você pode precisar de uma licença. [\[avançado\]](#)
- Procure por pacotes para tipos específicos de gráficos na [galeria de htmlwidgets](#).

Tutoriais

- [Tutorial highcharter](#)
- [Documentação Highcharts](#)

Mapas com leaflet

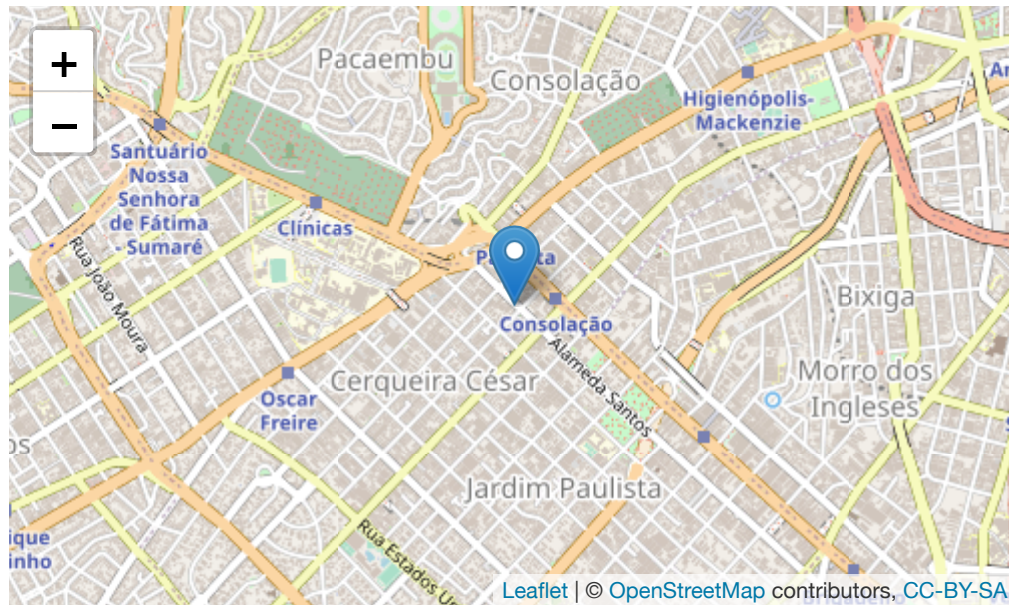
O pacote `{leaflet}` nos permite criar mapas interativos baseados na biblioteca JavaScript open-source [Leaflet](#).

Para criar um mapa leaflet, utilizamos a função `leaflet::leaflet()` e diversas funções auxiliares para caracterizar nosso mapa. Um tutorial de como utilizar o `leaflet` se encontra [aqui](#).

A seguir, mostramos um exemplo simples de como criar um mapa leaflet.

```
library(leaflet)
```

```
leaflet(height = 300) %>%  
  addTiles() %>% # Adiciona a camada gráfica do OpenStreetMap (padrão)  
  addMarkers(  
    lng = -46.6623969, lat = -23.5581664,  
    popup = "A Curso-R morava aqui antes da pandemia :'"  
  )
```



Pacotes alternativos

- `highcharter::hcmmap()`: variação do `highcharter` para mapas, baseada na biblioteca JavaScript [Highcharts](#). [\[avançado\]](#)
- `{tmap}`: Pacote focado em mapas temáticos.

Tutoriais

- [Construindo mapas com o highcharter](#)
- [Documentação Highmaps](#)
- [Documentação do tmap](#)

Referências e material extra

htmlwidgets

- [Galeria htmlwidgets](#)

reactable

- [A biblioteca React Table](#)
- [Tutorial reactable](#)

DT

- [Tutorial DT](#)

plotly

- [Tutorial plotly](#)
- [Interactive web-based data visualization with R, plotly, and shiny](#)

highcharter/highcharts

- [Tutorial highcharter](#)
- [Biblioteca Highcharts](#)
- [Galeria Highcharts](#)
- [Documentação Highcharts](#)

leaflet

- [Biblioteca Leaflet](#)
- [Tutorial Leaflet](#)

highmaps

- [Galeria Highmaps](#)
- [Documentação Highmaps](#)