

# Exemplo histórico

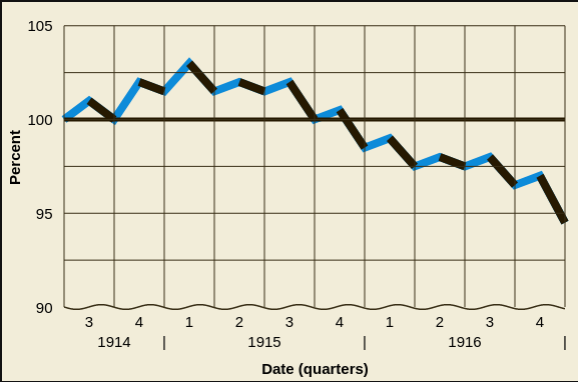
## Introdução

Para este trabalho, foi escolhido reproduzir um gráfico de Willard Cope Brinton, o qual é possível ver em seu livro Graphic Presentation, de 1880, na [página 303](#). O gráfico em questão é um gráfico de linhas, como é possível ver no livro, ele representa um método desenvolvido em Washington D. C. para acompanhar a construção naval britânica, onde a linha azul mostra o que foi construído no trimestre, e a linha preta o que foi afundado ao fim do mesmo trimestre. Além disso, também podemos extrair do livro que cem por cento equivalia a quantidade de navios a vapor britânicos no início da guerra.

Após essa contextualização, devemos explicar qual é a importância histórica desse gráfico: a linha ondulada no eixo X. Nesse sentido, podemos ver que o gráfico não começa com a escala do eixo Y em zero, mas sim em 90 e, por esse motivo, o autor decidiu chamar a atenção do leitor a esse fato reproduzindo a linha ondulada que vemos no eixo X, simulando um papel rasgado, o que indica que haviam mais dados abaixo daquela linha e que devemos nos atentar a isso.

Nesse contexto, sabemos que hoje em dia é considerado má prática marcar as linhas de grade que não sejam as dos eixos X e Y (em 0) de forma diferente das demais, pois não devemos dar a entender que as linhas de grade têm importâncias diferentes de acordo com a marca gráfica. Além disso, não é recomendado começar com a escala diferente de 0, pois com isso podemos estar omitindo dados relevantes, como por exemplo em um gráfico de barras, se começarmos a escala em outra posição podemos causar a impressão errada na diferença entre os dados das diferentes barras. Entretanto, vemos que no caso onde desejamos recuperar dados de valores absolutos e compará-los (como no exemplo do gráfico de barras), realmente isso pode ser uma má prática, entretanto quando se deseja apenas mostrar variações nos valores, pode ser realmente útil e em certos casos necessário omitir parte do "domínio" dos valores como é feito no gráfico de Brinton, e dessa maneira o método que ele utilizou para indicar que isso foi feito é muito interessante e pode ser bem vindo.

Dito isso, podemos ver abaixo o gráfico reproduzido em Python, utilizando a biblioteca Altair:



## Considerações finais

Como é possível ver, o resultado final atingido ficou muito parecido com o gráfico original, sendo bem satisfatório. Entretanto, o processo de criação do gráfico fica omitido ao só ver o resultado final. Nesse sentido, é importante ressaltar que ao longo do processo diversos ajustes foram feitos, como: o *grid* do gráfico preciso ser removido (dos valores padrões) pois o Altair não deixa defini-lo precisamente, ele cria modificações como julga melhor para o resultado, deixando diferente do desejado; a linha ondulada embaixo do gráfico também foi feita "manualmente" (com uma função senooidal, gerando os dados dos pontos da senoide e plotando um gráfico de linha); as *labels* do eixo X também foram feitas "manualmente", pois criar labels multilinha é um ainda problema conhecido e não resolvido do Altair e, além disso, acontecia o mesmo problema do *grid*. Ainda buscando reproduzir o gráfico da forma mais fiel possível, a paleta de cores foi extraída da imagem original utilizando o [Adobe Color](#) (sabemos que a cor do fundo se deve a cor da página onde foi feito o gráfico, mas mesmo assim a cor dela também foi considerada na paleta).

Mesmo assim, podemos notar algumas diferenças do gráfico original, como a alternância entre a linha azul e preta, pois foi necessário fazer toda a linha azul e após isso sobrepor com as linhas pretas a cada vez que os dados indicavam uma descida. Também podemos ver que foram modificadas algumas anotações textuais, mas isso foi feito propositalmente, pois não era o mais importante a ser destacado nessa representação gráfica, por esse motivo a modificação foi feita para ser mais legível (foram alteradas as posições dos títulos das *labels* e removida a anotação de "100%", reforçando a linha horizontal que já estava presente no gráfico).

Dessa forma, embora tenha sido complicado reproduzir o gráfico, o processo para chegar ao resultado final foi importante para entender melhor o funcionamento da biblioteca utilizada para a criação dos gráficos, compreendendo suas limitações e descobrindo meios de contorná-las. Por fim, novamente, esta é mais uma experiência que nos faz dar valor a documentação da biblioteca utilizada, que por mais que ela não seja suficiente para resolver todos os problemas (até porque nem todos os problemas têm solução "da forma esperada" - de maneira que escala bem com outros dados), ela ainda é uma grande ferramenta de auxílio para o desenvolvimento dos projetos.

## Observações

O código para gerar os gráficos e esta página HTML foram feitos em um notebook python, que pode ser encontrado em um documento separado. Tanto essa página HTML quanto o notebook python estão disponíveis no [repositório do GitHub](#).

## Bibliografia

ALTAIR. Vega-Altair: Declarative Visualization in Python. Disponível em: [altair-viz.github.io/](https://altair-viz.github.io/). Acessado em 8 de setembro de 2022.

VANDERPLAS, Jake. Multiple Charts in one HTML. Disponível em: [github.com/](https://jakevdp.github.io/). Acessado em 1 de outubro de 2022.

MATERIAL. Dark theme. Disponível em: [material.io/](https://material.io/). Acessado em 8 de outubro de 2022.

ADOBE. Adobe Color. Disponível em: [color.adobe.com/pt/create/](https://color.adobe.com/pt/create/). Acessado em 8 de outubro de 2022.

ALTAIR. Top-Level Chart Configuration. Disponível em: [altair-viz.github.io/](https://altair-viz.github.io/). Acessado em 8 de outubro de 2022.

ANDREWS, R. J.. Design secrets we can learn from historic visualizations. Disponível em: [www.tableau.com/](https://www.tableau.com/). Acessado em 8 de outubro de 2022.

ANDREWS, R. J.. Tear Up Your Baseline. Disponível em: [medium.com/](https://medium.com/). Acessado em 8 de outubro de 2022.

BRINTON, Willard Cope. Graphic presentation. Disponível em: [archive.org/](https://archive.org/). Acessado em 8 de outubro de 2022.