

Ciência Reprodutível para Experimentos em Computação de Alto Desempenho

Pedro Bruel, Lucas Schnorr, Alfredo Goldman

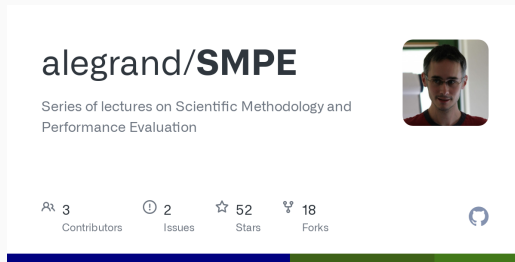
phrb@ime.usp.br

3 de maio de 2021

Introdução

Agradecimentos e Crédito

A **Arnaud Legrand** e seu curso:



The screenshot shows the GitHub repository page for 'alegrand/SMPE'. The repository name is displayed in a large, bold font. Below it, the description reads 'Series of lectures on Scientific Methodology and Performance Evaluation'. To the right of the text is a profile picture of a man with glasses. Below the repository name, there are statistics: 3 Contributors, 2 Issues, 52 Stars, and 18 Forks. The GitHub logo is also visible. The page has a blue and green header bar.

alegrand/**SMPE**

Series of lectures on Scientific Methodology and Performance Evaluation

3 Contributors 2 Issues 52 Stars 18 Forks

<https://github.com/alegrand/SMPE>



Dependências e outros Recursos

- Site com **instruções** e mais **recursos**:

Ciência Reprodutível - ERADSP 2021

Ciência Reprodutível para Experimentos em Computação de Alto Desempenho (ERADSP 2021)

[View the Project on GitHub](#)
phrb/reprodutibilidade-eradsp-2021

Ciência Reprodutível para Experimentos em Computação de Alto Desempenho

Este é o site do minicurso apresentado na ERADSP 2021.

Conteúdo

- [Exercício Prático](#)
- [Recursos](#)

<https://phrb.github.io/reprodutibilidade-eradsp-2021>

- Temos uma imagem **Docker** com Jupyter Notebook, R, pacotes, e dados:

```
git clone https://github.com/phrb/reprodutibilidade-eradsp-2021.git
cd reprodutibilidade-eradsp-2021/exercicio_pratico && ./build.sh -b
```

Roteiro

O que é Ciência Reprodutível?

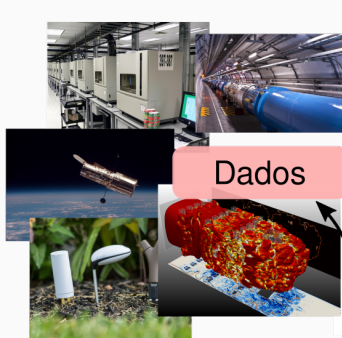
Desafios e Abordagens para se fazer Ciência Reprodutível

Mão na Massa: Ferramentas para Reprodutibilidade

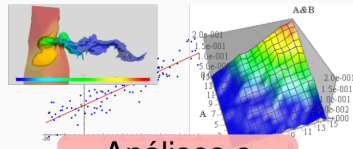
O que é Ciência Reprodutível?

Provocação: O que Sobrevive do Trabalho Científico?

Quem Produz



Dados



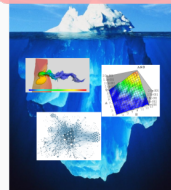
Análises e Visualizações



Quem Lê

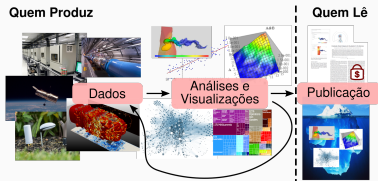


Publicação



O que é Ciência Reprodutível?

Trabalhar de forma **transparente** para diminuir a distância entre **quem produz** e **quem lê**



Trabalhar de forma transparente?

- Caderno de laboratório e metodologia
- Ambientes de software, controle de versão
- Plataformas de compartilhamento, colaboração, e arquivamento

Definições

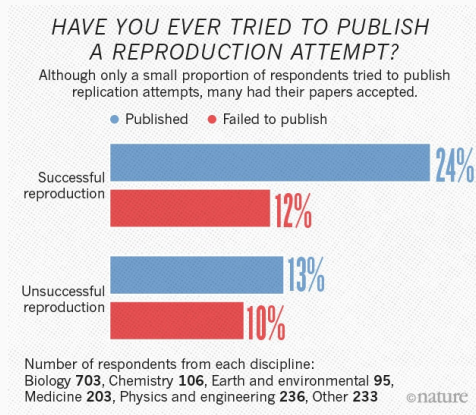
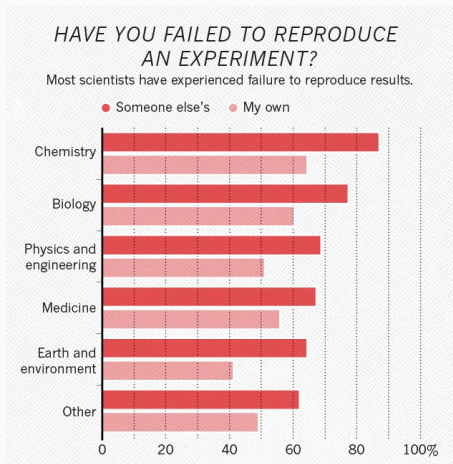
- **Vocabulário Internacional de Metrologia (VIM)**

Distingue entre **resultados** e **conclusões** que podem ser reproduzidos:

- Pela mesma equipe, nas mesmas condições experimentais: *Repetibilidade*
- Por uma equipe diferente, nas mesmas condições experimentais: *Replicabilidade*
- Por uma equipe diferente, em condições experimentais diferentes: **Reprodutibilidade**

Há uma Crise de Reprodutibilidade?

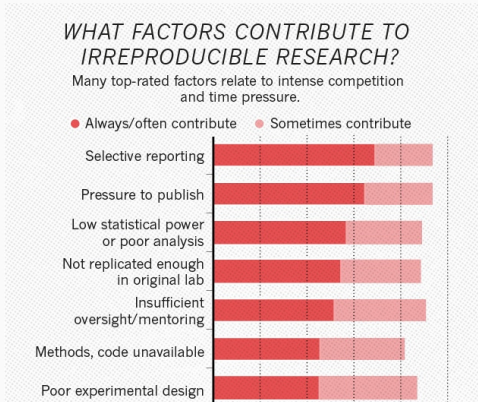
Resultados de um questionário com 1.500 cientistas:



(1,500 Scientists Lift the Lid on Reproducibility, Nature, Maio de 2016)

O que Dificulta a Reprodutibilidade?

Resultados de um questionário com 1.500 cientistas:



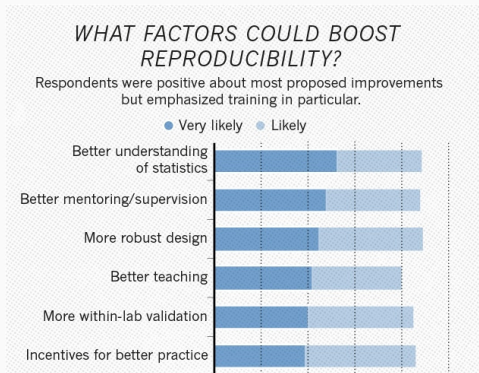
Dificultam a Reprodutibilidade

- Reportagem **seletiva**
- **Pressão** por publicações
- Dificuldades com **estatística**
- Falta de **acesso** aos dados

(1,500 Scientists Lift the Lid on Reproducibility, Nature, Maio de 2016)

O que pode Promover a Reprodutibilidade?

Resultados de um questionário com 1.500 cientistas:



Promovem a Reprodutibilidade

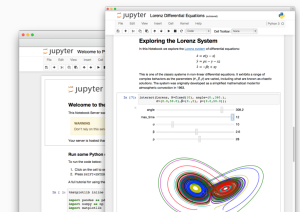
- Estudar **estatística**
- **Colaboração** e **comunidade**
- Melhores **incentivos**

(1,500 Scientists Lift the Lid on Reproducibility, Nature, Maio de 2016)

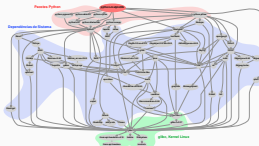
Desafios e Abordagens para se fazer Ciência Reprodutível

Ferramentas Existentes e Padrões Emergentes

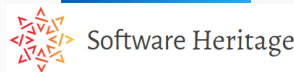
Cadernos de Laboratório



Ambientes de Software



Plataformas de Compartilhamento



1 Documento Computacional

Meu computador me diz que π vale aproximadamente

3.141592653589793

Mas se usarmos o **método** da *Agulha de Buffon*, obteremos a **aproximação**:

```
[8]: import numpy as np

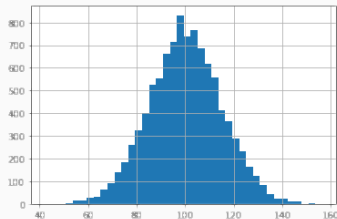
N = 1000000
x = np.random.uniform(size = N, low = 0, high = 1)
theta = np.random.uniform(size = N, low = 0, high = pi / 2)

approx_pi = 2 / (sum(x + np.sin(theta) > 1) / N)

print(approx_pi)
```

3.142712129140327

Podemos também incluir fórmulas matemáticas como $\frac{1}{\sigma\sqrt{2/\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$ e *desenhos* que não têm nada a ver com π (ele ao menos aparece como constante de normalização ☹)



Cadernos de Laboratório

Jupyter exemplo_pi Last Checkpoint: 20 minutes ago (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help

Run

Markdown

```
# Documento Computacional
Meu computador me diz que  $\pi$  vale aproximadamente
```

Código

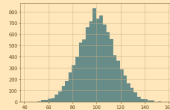
```
In [3]: from math import *
print(pi)
3.141592653589793
```

```
1 Mas se usarmos o "método" da [Agulha de Buffon](https://pt.wikipedia.org/wiki/Agulha_de_Buffon), obteremos a
  "aproximação":
```

```
In [8]: import numpy as np
2
3 N = 1000000
4 x = np.random.uniform(size = N, low = 0, high = 1)
5 theta = np.random.uniform(size = N, low = 0, high = pi / 2)
6
7 approx_pi = 2 / (sum(x + np.sin(theta) > 1) / N)
8 print(approx_pi)
3.142712129140327
```

```
1 Podemos também incluir fórmulas matemáticas como  $\frac{1}{\sigma\sqrt{2/\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$  e "desenhos" que não têm nada a ver com  $\pi$  (ele ao menos aparece como constante de
  normalização 🙄)
```

```
In [10]: import matplotlib.pyplot as plt
2
3 mu, sigma = 100, 15
4 x = mu + (sigma * np.random.randn(10000))
5
6 plt.hist(x, 40)
7 plt.grid(True)
8 plt.show()
```



Resultados

Exportar



1 Documento Computacional

Meu computador me diz que π vale aproximadamente

3.141592653589793

Mas se usarmos o método da Agulha de Buffon, obteremos a aproximação:

```
[8]: import numpy as np

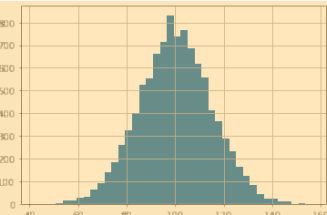
N = 1000000
x = np.random.uniform(size = N, low = 0, high = 1)
theta = np.random.uniform(size = N, low = 0, high = pi / 2)

approx_pi = 2 / (sum(x + np.sin(theta) > 1) / N)

print(approx_pi)
```

3.142712129140327

Podemos também incluir fórmulas matemáticas como $\frac{1}{\sigma\sqrt{2/\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$ e "desenhos" que não têm nada a ver com π (ele ao menos aparece como constante de normalização 🙄)



Cadernos de Laboratório

jupyter exemplo_pi Last Checkpoint: 20 minutes ago (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help

Markdown

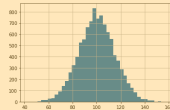
```
# Documento Computacional
Meu computador me diz que  $\pi$  vale aproximadamente
```

Código

```
In [3]: from math import *
print(pi)
3.141592653589793
```

```
1 Mas se usarmos o "método" da [Agulha de Buffon](https://pt.wikipedia.org/wiki/Agulha_de_Buffon), obteremos a
  "aproximação":
In [8]: import numpy as np
2 N = 1000000
3 x = np.random.uniform(size = N, low = 0, high = 1)
4 theta = np.random.uniform(size = N, low = 0, high = pi / 2)
5 approx_pi = 2 / (sum(x + np.sin(theta) > 1) / N)
6 print(approx_pi)
3.142712129140327
```

```
1 Podemos também incluir fórmulas matemáticas como  $\frac{1}{\sigma\sqrt{2/\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$  e "desenhos" que não
  têm nada a ver com  $\pi$  (ele ao menos aparece como constante de
  normalização 🐘)
In [10]: import matplotlib.pyplot as plt
2 mu, sigma = 100, 15
3 x = mu + (sigma * np.random.randn(10000))
4 plt.hist(x, 40)
5 plt.grid(True)
6 plt.show()
```



Exportar



1 Documento Computacional

Meu computador me diz que π vale aproximadamente

3.141592653589793

Mas se usarmos o método da Agulha de Buffon, obteremos a aproximação:

```
[8]: import numpy as np

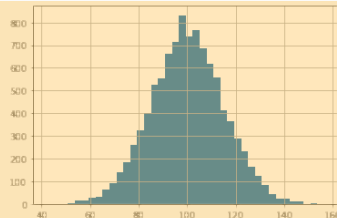
N = 1000000
x = np.random.uniform(size = N, low = 0, high = 1)
theta = np.random.uniform(size = N, low = 0, high = pi / 2)

approx_pi = 2 / (sum(x + np.sin(theta) > 1) / N)

print(approx_pi)
```

3.142712129140327

Podemos também incluir fórmulas matemáticas como $\frac{1}{\sigma\sqrt{2/\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$ e "desenhos" que não têm nada a ver com π (ele ao menos aparece como constante de normalização 🐘)



Studio

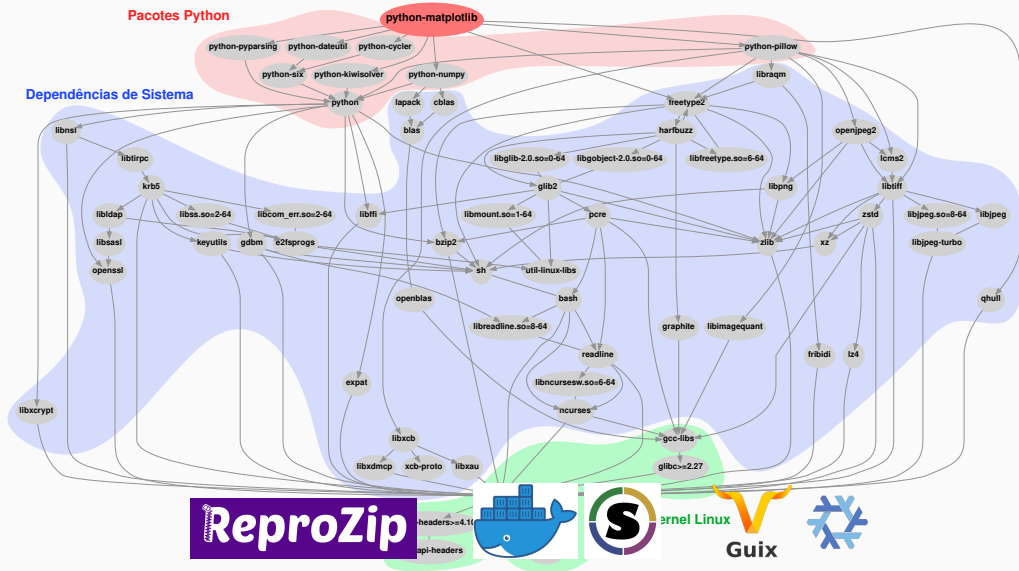
<https://jupyter.org/try>

Ambientes de Software: O que se Esconde nas Dependências?

```
$ pacman -Qi python-matplotlib
```

```
Name           : python-matplotlib
Version        : 3.4.1-2
Depends On     : freetype2 python-cycler python-dateutil python-kiwisolver
                  python-numpy python-pillow python-pyparsing qhull
Optional Deps  : tk: Tk{Agg,Cairo} backends [installed]
                  pyside2: alternative for Qt5{Agg,Cairo} backends
                  python-pyqt5: Qt5{Agg,Cairo} backends [installed]
                  python-gobject: for GTK3{Agg,Cairo} backend [installed]
                  python-wxpython: WX{,Agg,Cairo} backend
                  python-cairo: {GTK3,Qt5,Tk,WX}Cairo backends [installed]
                  python-cairocffi: alternative for Cairo backends
                  python-tornado: WebAgg backend [installed]
                  ffmpeg: for saving movies [installed]
                  imagemagick: for saving animated gifs [installed]
                  ghostscript: usetex dependencies [installed]
                  texlive-bin: usetex dependencies [installed]
                  texlive-latexextra: usetex usage with pdflatex [installed]
                  python-certifi: https support [installed]
```


Ambientes de Software: O que se Esconde nas Dependências?



Plataformas de Compartilhamento e Arquivamento

- D. Spinellis. *The Decay and Failures of URL References*. CACM, 46(1), 2003
"A meia-vida de uma referência em URL é de aproximadamente 4 anos após sua publicação"
- P. Habibzadeh. *Decay of References to Web sites in Articles Published in General Medical Journals: Mainstream vs Small Journals*. Applied Clinical Informatics. 4 (4), 2013
"a meia-vida durou entre 2,2 anos no EMHJ e 5,3 anos no BMJ"

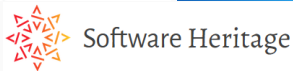
Arquivamento de Artigos



Arquivamento de Dados



Arquivamento de Software



or



= excelentes para colaborações (\neq arquivamento)

Desafios

- Como **planejar** experimentos?
- Como **analisar** resultados?
- Datasaurus Dozen
 - <https://cran.r-project.org/web/packages/datasauRus/>

Abordagens

- Gráficos, antes de qualquer análise
- Análises mais simples primeiro: mas fáceis de interpretar
- Documentos computacionais
- Desenho de Experimentos
- Controle de versão

Estatística: O que é Machine Learning?

Conceito	Estatística	Aprendizado de Máquina
Usar dados para estimar quantidades desconhecidas	Estimação	Aprendizado
Predizer \mathbf{y} discreto a partir de \mathbf{x}	Classificação	Aprendizado Supervisionado
Dividir dados em grupos	Clusterização	Aprendizado Não-Supervisionado
$(\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_N, \mathbf{y}_N)$	Desenho Experimental	Conjunto de Treinamento
$(\mathbf{x}_1, \dots, \mathbf{x}_N)$	Variáveis Preditoras	Características
Intervalo contendo uma estimativa	Intervalo de Confiança	—
...

É Possível Garantir a Reprodutibilidade?

Abordagens da ACM

Mão na Massa: Ferramentas para Reprodutibilidade

Controle de Versão

Hospedagem

Arquivamento

Conclusão

É possível fazer Ciência (mais) Reprodutível!

Ciência Reprodutível para Experimentos em Computação de Alto Desempenho

Pedro Bruel, Lucas Schnorr, Alfredo Goldman

phrb@ime.usp.br

3 de maio de 2021