

# REPRODUCIBILITY CRISIS, OPEN SCIENCE,... AND COMPUTER SCIENCE

---

Arnaud Legrand



LISTIC, Annecy  
February 2023



# WHAT IS SCIENCE?

Question 1: In less than 5 lines give a definition of "Science"

# WHAT IS SCIENCE?

Question 1: In less than 5 lines give a definition of "Science"

## Dictionary of science and technology

1. the study of the physical and natural world and phenomena, especially by using systematic observation and experiment
2. a particular area of study or knowledge of the physical world
3. a systematically organized body of knowledge about a particular subject

**New Oxford Dictionary** the intellectual and practical activity encompassing the systematic study of the structure and behavior of the physical and natural world through observation and experiment : the world of science and technology.

1. a particular area of this : veterinary science | the agricultural sciences.
2. a systematically organized body of knowledge on a particular subject : the science of criminology.
3. archaic knowledge of any kind.

# WHAT IS SCIENCE?

Question 1: In less than 5 lines give a definition of "Science"

## Dictionary of science and technology

1. the study of the physical and natural world and phenomena, especially by using systematic observation and experiment
2. a particular area of study or knowledge of the physical world
3. a systematically organized body of knowledge about a particular subject

**New Oxford Dictionary** the intellectual and practical activity encompassing the systematic study of the structure and behavior of the physical and natural world through observation and experiment : the world of science and technology.

1. a particular area of this : veterinary science | the agricultural sciences.
2. a systematically organized body of knowledge on a particular subject : the science of criminology.
3. archaic knowledge of any kind.

**Building Reliable Knowledge**

# PUBLIC EVIDENCE FOR A LACK OF REPRODUCIBILITY

- J.P. Ioannidis. *Why Most Published Research Findings Are False* PLoS Med. 2005.
- *Lies, Damned Lies, and Medical Science*, The Atlantic. Nov, 2010
- *Reproducibility: A tragedy of errors*, Nature, Feb 2016.
- Steen RG, *Retractions in the scientific literature: is the incidence of research fraud increasing?*, J. Med. Ethics 37, 2011



## Science has lost its way, at a big cost to humanity

Researchers are rewarded for splashy findings, not for double-checking accuracy. So many scientists looking for cures to diseases have been building on ideas that aren't even true.

The screenshot shows the homepage of the journal "Science". At the top, there's a red banner with the AAAS logo and the word "SCIENCE". Below it, a sub-banner says "The World's Leading Journal of Original Scientific Research, Global News, and Commentary". The main navigation menu includes "NEWS", "SCIENCE JOURNALS", "CAREERS", "MULTIMEDIA", and "COLLECTIONS". The main content area features a large headline "Science has lost its way, at a big cost to humanity" and a sub-headline "Researchers are rewarded for splashy findings, not for double-checking accuracy. So many scientists looking for cures to diseases have been building on ideas that aren't even true.". Below this, there's a detailed article summary with a photo of a scientist, the author's name (Marcia McNutt), and the date (January 17, 2014). The article is available in PDF format. The sidebar on the left contains links for "Article Views", "Summary", "Full Text", "Full Text (PDF)", "Editorial", and "Reproducibility". The footer includes a "Related Content" section and a "Save to My Folders" link.

The screenshot shows the homepage of the journal "nature". At the top, there's a red banner with the word "nature" and the subtitle "International weekly journal of science". Below it, a sub-banner says "Announcement: Reducing our irreproducibility". The main navigation menu includes "Home", "News & Comment", "Research", "Careers & Jobs", "Current Issue", and "Archive". The main content area features a large headline "Announcement: Reducing our irreproducibility" and a sub-headline "24 April 2013". Below this, there's a detailed announcement with a photo of a scientist, the author's name (Marcia McNutt), and the date (April 24, 2013). The article is available in PDF format. The sidebar on the left contains links for "Article Views", "Summary", "Full Text", "Full Text (PDF)", "Editorial", and "Reproducibility". The footer includes a "Related Content" section and a "Save to My Folders" link.

A large graphic with the title "HOW SCIENCE GOES WRONG." in a stylized, colorful font. The background features various scientific illustrations like a globe, DNA helixes, and microscopes. The word "SCIENCE" is written vertically in green and blue. The word "WRONG" is written vertically in pink and purple. There are also some smaller text elements like "Economist" and "Investment tips from Nobel economists".

The screenshot shows the homepage of the journal "nature". At the top, there's a red banner with the word "nature" and the subtitle "International weekly journal of science". Below it, a sub-banner says "Announcement: Reducing our irreproducibility". The main navigation menu includes "Menu", "Advanced search", and "Search". The main content area features a large headline "Announcement: Reducing our irreproducibility" and a sub-headline "24 April 2013". Below this, there's a detailed announcement with a photo of a scientist, the author's name (Marcia McNutt), and the date (April 24, 2013). The article is available in PDF format. The sidebar on the left contains links for "Article Views", "Summary", "Full Text", "Full Text (PDF)", "Editorial", and "Reproducibility". The footer includes a "Related Content" section and a "Save to My Folders" link.

The screenshot shows the homepage of the magazine "TheScientist". At the top, there's a large logo with the word "TheScientist" in a bold, serif font. Below it, a sub-banner says "EXPLORING LIFE. INSPIRING INNOVATION". The main content area features a large headline "NIH Tackles Irreproducibility" and a sub-headline "The federal agency speaks out about how to improve the quality of scientific research.". Below this, there's a detailed article summary with a photo of a scientist, the author's name (Jef Akst), and the date (January 28, 2014). The article is available in PDF format. The sidebar on the left contains links for "Article Views", "Summary", "Full Text", "Full Text (PDF)", "Editorial", and "Reproducibility". The footer includes a "Related Content" section and a "Save to My Folders" link.

Courtesy V. Stodden, SC, 2015

# NEWSWORTHY STORIES ABOUT SCIENTIFIC MISCONDUCT

**Dong-Pyou Han** Assistant professor, Biomedical sciences, Iowa State University, 2013

*Falsified blood results to make it appear as though a vaccine exhibited anti-HIV activity*

- Han and his team received  $\approx$  \$19 million from NIH
- 1 retracted publication and resignation of university. Sentenced in 2015 to 57 months imprisonment for fabricating and falsifying data in HIV vaccine trials. \$7.2 million!

# NEWSWORTHY STORIES ABOUT SCIENTIFIC MISCONDUCT

**Dong-Pyou Han** Assistant professor, Biomedical sciences, Iowa State University, 2013

*Falsified blood results to make it appear as though a vaccine exhibited anti-HIV activity*

- Han and his team received  $\approx$  \$19 million from NIH
- 1 retracted publication and resignation of university. Sentenced in 2015 to 57 months imprisonment for fabricating and falsifying data in HIV vaccine trials. \$7.2 million!

**Diederik Stapel** Professor, Social Psychology, Univ. Tilburg, 2011

*I failed as a scientist. I adapted research data and fabricated research. Not once, but several times, not for a short period, but over a longer period of time. [...] I am aware of the suffering and sorrow that I caused to my colleagues... I did not withstand the pressure to score, to publish, the pressure to get better in time. I wanted too much, too fast. In a system where there are few checks and balances, where people work alone, I took the wrong turn.*

58 retracted publications

# NEWSWORTHY STORIES ABOUT SCIENTIFIC MISCONDUCT

**Dong-Pyou Han** Assistant professor, Biomedical sciences, Iowa State University, 2013

*Falsified blood results to make it appear as though a vaccine exhibited anti-HIV activity*

- Han and his team received  $\approx$  \$19 million from NIH
- 1 retracted publication and resignation of university. Sentenced in 2015 to 57 months imprisonment for fabricating and falsifying data in HIV vaccine trials. \$7.2 million!

**Diederik Stapel** Professor, Social Psychology, Univ. Tilburg, 2011

*I failed as a scientist. I adapted research data and fabricated research. Not once, but several times, not for a short period, but over a longer period of time. [...] I am aware of the suffering and sorrow that I caused to my colleagues... I did not withstand the pressure to score, to publish, the pressure to get better in time. I wanted too much, too fast. In a system where there are few checks and balances, where people work alone, I took the wrong turn.*

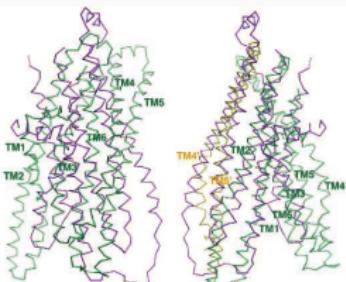
58 retracted publications

**Brian Wansink** Professor, Psychological Nutrition, Cornell, 2016

*I gave her a data set of a self-funded, failed study which had null results. I said "This cost us a lot of time and our own money to collect. There's got to be something here we can salvage because it's a cool (rich & unique) data set." I told her what the analyses should be. [...] Every day she came back with puzzling new results, and every day we would scratch our heads, ask "Why," and come up with another way to reanalyze the data with yet another set of plausible hypotheses*

17 retracted publications

# How COMPUTERS BROKE SCIENCE



Geoffrey Chang (Scripps, UCSD) works on crystallography and studies the structure of cell membrane proteins.

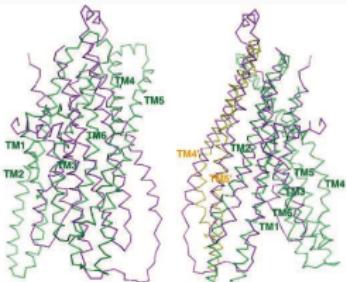
He specialized in structures of **multidrug resistant transporter proteins in bacteria**: MsbA de Escherichia Choli (Science, 2001), Vibrio cholera (Mol. Biology, 2003), Salmonella typhimurium (Science, 2005)

2006: Inconsistencies reveal **a programming mistake**

*A homemade data-analysis program had flipped two columns of data, inverting the electron-density map from which his team had derived the protein structure.*

5 retractions that motivate **improved software engineering practices** in comp. biology

# HOW COMPUTERS BROKE SCIENCE



Geoffrey Chang (Scripps, UCSD) works on crystallography and studies the structure of cell membrane proteins.

He specialized in structures of **multidrug resistant transporter proteins in bacteria**: MsbA de Escherichia Choli (Science, 2001), Vibrio cholera (Mol. Biology, 2003), Salmonella typhimurium (Science, 2005)

2006: Inconsistencies reveal **a programming mistake**

*A homemade data-analysis program had flipped two columns of data, inverting the electron-density map from which his team had derived the protein structure.*

5 retractions that motivate **improved software engineering practices** in comp. biology

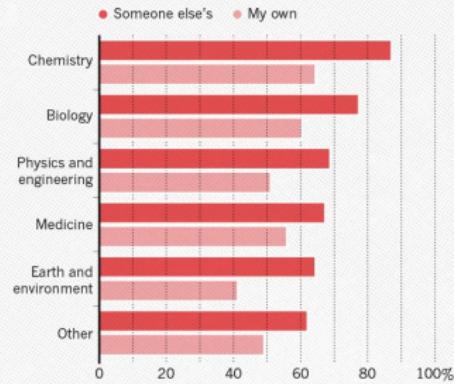
There is **worse!**

- The generalized and intensive use of **spreadsheets** (**COVID tracing**)
- Relying on **black box** statistical methods is infinitely easier than understanding them  
(Learning and Data Analytics frameworks = nuke)
- Numerical errors and software environment unawareness

# A REPRODUCIBILITY CRISIS?

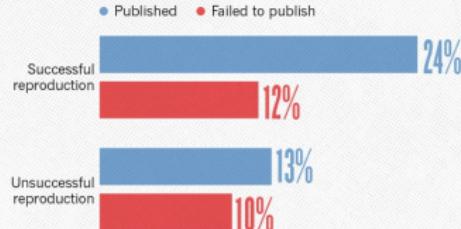
## HAVE YOU FAILED TO REPRODUCE AN EXPERIMENT?

Most scientists have experienced failure to reproduce results.



## HAVE YOU EVER TRIED TO PUBLISH A REPRODUCTION ATTEMPT?

Although only a small proportion of respondents tried to publish replication attempts, many had their papers accepted.



Number of respondents from each discipline:  
Biology 703, Chemistry 106, Earth and environmental 95,  
Medicine 203, Physics and engineering 236, Other 233.

1,500 scientists lift the lid on reproducibility,

Nature, May 2016

## Social causes

- Fraud, conflict of interest (pharmaceutic, ...)
- No incentive to reproduce/check our own work (afap), nor the work of others (big results!), nor to allow others to check (competition)
- Peer review does not scale: 1+ million articles per year!

## Methodological or technical causes

- The many biases (apophenia, confirmation, hindsight, experimenter, ...): bad designs
- Selective reporting, weak analysis (statistics, data manipulation mistakes, computational errors)
- Lack of information, code/raw data unavailable

# NO TRANSPARENCY NO CONSENSUS



# DIFFERENT REPRODUCIBILITY CONCERN IN MODERN SCIENCE

**Social Sciences, Oncology, ...** methodology, statistics, pre-registration

**Genomics** software engineering, computational reproducibility, provenance

**Computational fluid dynamics** numerical issues

**Artificial Intelligence** most of the above

*The processing steps between raw observations and findings have gotten increasingly numerous and complex*

## Authors



# DIFFERENT REPRODUCIBILITY CONCERN IN MODERN SCIENCE

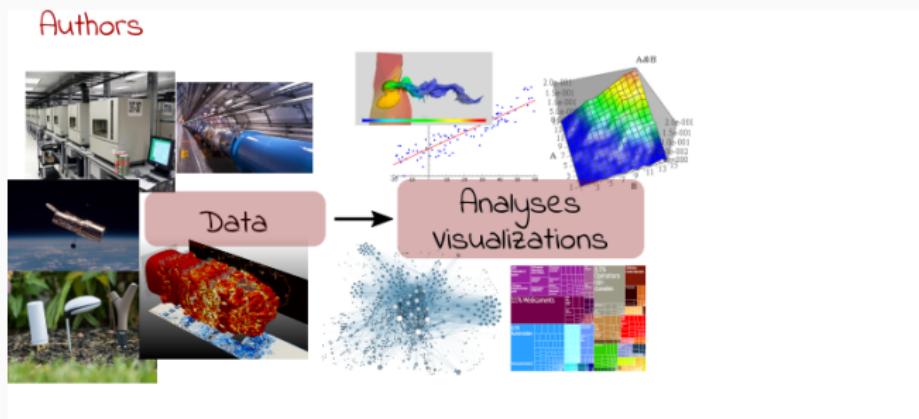
**Social Sciences, Oncology, ...** methodology, statistics, pre-registration

**Genomics** software engineering, computational reproducibility, provenance

**Computational fluid dynamics** numerical issues

**Artificial Intelligence** most of the above

*The processing steps between raw observations and findings have gotten increasingly numerous and complex*



# DIFFERENT REPRODUCIBILITY CONCERN IN MODERN SCIENCE

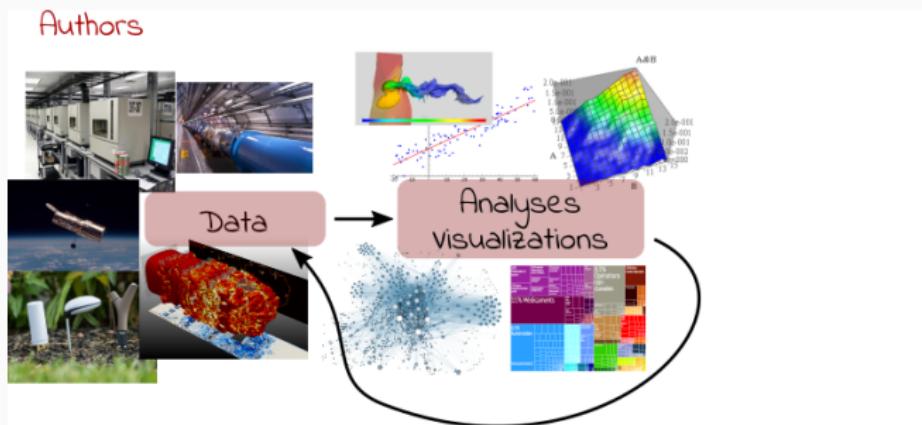
**Social Sciences, Oncology, ...** methodology, statistics, pre-registration

**Genomics** software engineering, computational reproducibility, provenance

**Computational fluid dynamics** numerical issues

**Artificial Intelligence** most of the above

*The processing steps between raw observations and findings have gotten increasingly numerous and complex*



# DIFFERENT REPRODUCIBILITY CONCERN IN MODERN SCIENCE

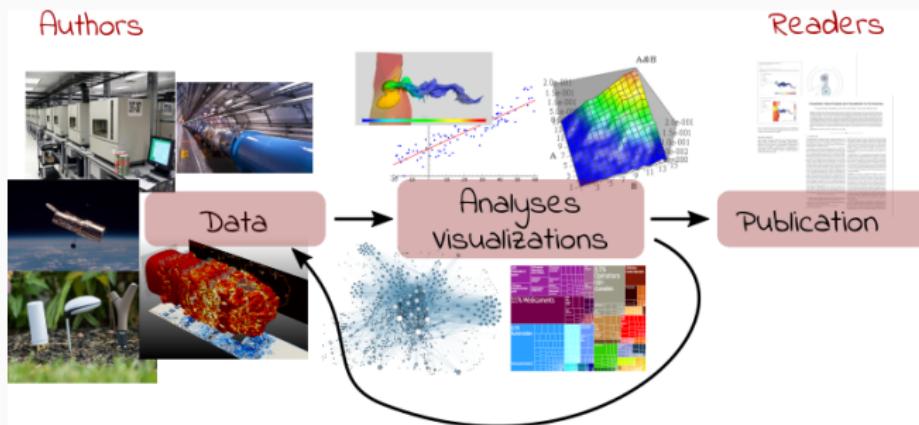
**Social Sciences, Oncology, ...** methodology, statistics, pre-registration

**Genomics** software engineering, computational reproducibility, provenance

**Computational fluid dynamics** numerical issues

**Artificial Intelligence** most of the above

*The processing steps between raw observations and findings have gotten increasingly numerous and complex*



# DIFFERENT REPRODUCIBILITY CONCERN IN MODERN SCIENCE

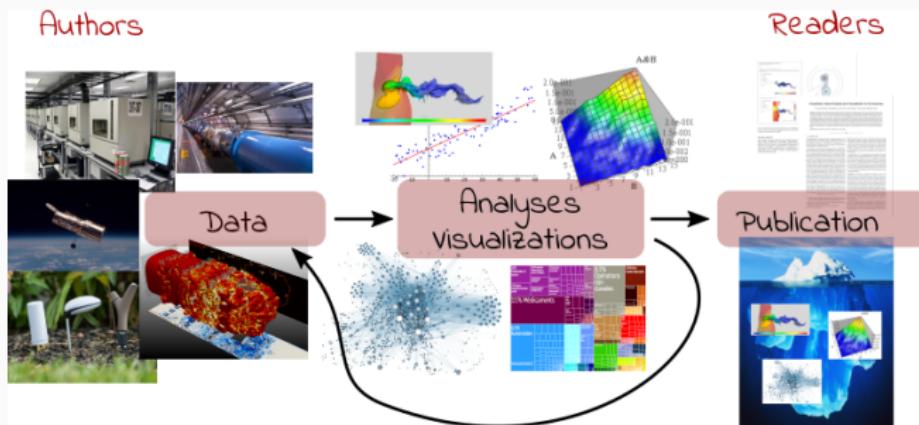
**Social Sciences, Oncology, ...** methodology, statistics, pre-registration

**Genomics** software engineering, computational reproducibility, provenance

**Computational fluid dynamics** numerical issues

**Artificial Intelligence** most of the above

*The processing steps between raw observations and findings have gotten increasingly numerous and complex*



# DIFFERENT REPRODUCIBILITY CONCERN IN MODERN SCIENCE

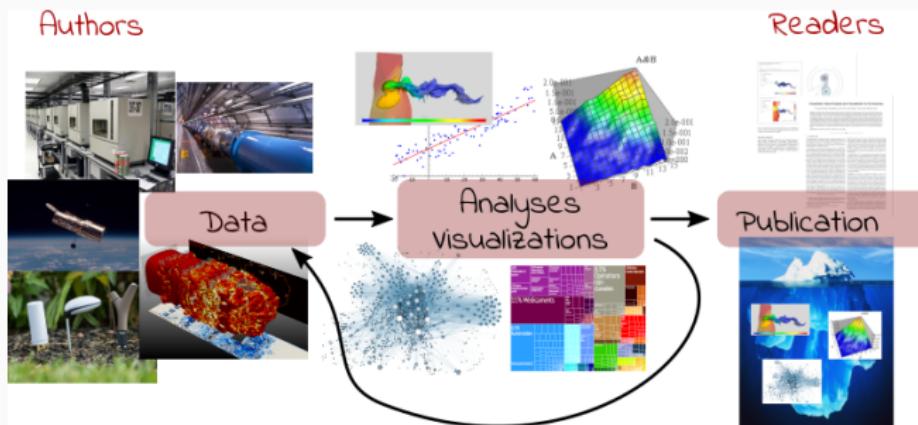
**Social Sciences, Oncology, ...** methodology, statistics, pre-registration

**Genomics** software engineering, computational reproducibility, provenance

**Computational fluid dynamics** numerical issues

**Artificial Intelligence** most of the above

*The processing steps between raw observations and findings have gotten increasingly numerous and complex*



Reproducible Research = Bridging the Gap by working Transparently

# REPRODUCIBLE RESEARCH PRACTICES

---

# REPRODUCIBILITY (GLOSSARY MAY VARY)

Many **definitions** (*replicability, repeatability, reproducibility*), sometimes conflicting  
(*new data, same person, independent researcher*)

|                                         |                                                                             |   |                                        |
|-----------------------------------------|-----------------------------------------------------------------------------|---|----------------------------------------|
| <b>experimental</b><br>reproducibility  | similar input (data) +<br>similar experimental protocol                     | → | <b>similar results</b> <sup>1</sup>    |
| <b>statistical</b><br>reproducibility   | different input (data) +<br>same analysis                                   | → | <b>same conclusions</b> <sup>2</sup>   |
| <b>computational</b><br>reproducibility | similar input (data) +<br>same code/software +<br>same software environment | → | <b>exact same results</b> <sup>3</sup> |

Reproducible Research = A way of doing science so that scientific experiments, discoveries, results, etc. can be easily reproduced (done again), to be confirmed, or to be built on for the next study.

– Courtesy G. Durrif, 2021

<sup>1</sup>Up-to measurement variability and precision

<sup>2</sup>Independently from (random) sampling variability (fight bias)

<sup>3</sup>Bitwise

# "REPRODUCIBLE RESEARCH": FIRST APPEARANCE

Claerbout & Karrenbach, meeting of the Society of Exploration Geophysics, 1992

## Electronic Documents Give Reproducible Research a New Meaning

RE1.3

Jon F. Claerbout and Martin Karrenbach, Stanford Univ.

### SUMMARY

A revolution in education and technology transfer follows from the marriage of word processing and software command scripts. In this marriage an author attaches to every figure caption a pushbutton or a name tag usable to recalculate the figure from all its data, parameters, and programs. This provides a new meaning of reproducibility in computer documents.

In 1990, we set this sequence of goals:

- Learn how to merge a publication with its underlying computational analysis.
- Teach researchers how to prepare a document in a form where they themselves can reproduce their own research results a year or more later by "pressing a single button".
- Learn how to leave finished work in a condition where coworkers can reproduce the calculation including the final illustration by pressing a button in its caption.
- Prepare a complete copy of our local software environment so that graduating students can take their work away with them to other sites, press a button, and reproduce their Stanford work.
- Merge electronic documents written by multiple authors (SEP reports).

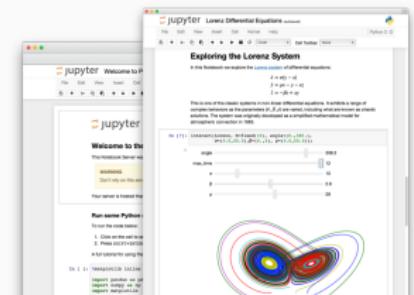
- make incremental improvements in electronic-document software
- seek partners for broadening standards (and making incremental improvements).

Our basic goal is reproducible research. The electronic document is our means to this end. In principle, reproducibility in research can be achieved without electronic documents and that is how we started. Our first nonelectronic reproducible document was a textbook in which the paper document contained the name of a program script in every figure caption. The program scripts were organized by book chapter and section so they could be correlated to an accompanying magnetic tape dump of the file system. The magnetic tape also contained all the necessary data to feed the program script.

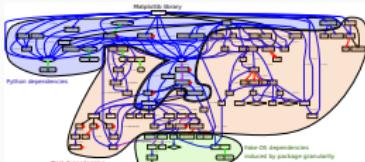
Now that we have begun using CD-ROM publication, we can go much further. Every figure caption contains a pushbutton that jumps to the appropriate science directory (folder) and initiates a figure rebuild command and then displays the figure, possibly as a movie or interactive program. We normally display seismic images of the earth's interior, but to reach wider audiences, Figure 1 shows a satellite weather picture which the pushbutton will animate as seen on commercial television. We include all our plot software as well as freely available software from many sources, including compilers and the L<sup>A</sup>T<sub>E</sub>X word processing systems. Naturally some software includes licensed software, but with the exception

# EXISTING TOOLS, EMERGING STANDARDS

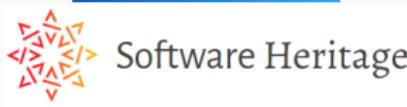
## Notebooks and workflows



## Software environments



## Sharing platforms



## GOOD PRACTICE #1

### TAKING NOTES AND DOCUMENTING

---



## Author

- I thought I used the same parameters but **I'm getting different results!**
- The new student wants to compare with **the method I proposed last year**
- My advisor asked me whether I took care of setting this or this but **I can't remember**
- The damned fourth reviewer asked for a major revision and wants me to change **Figure 3**. Which code and which data set did I use?
- **It worked yesterday!** 6 months later: Why did I do that?

## Reviewer

- As usual, there is **no confidence interval**, I wonder about the variability and whether the difference is **significant** or not
- That can't be true, I'm sure **they removed some points**
- Why is this graph in logscale? **How would it look like otherwise?** I'm not even **sure** of what this value means. If only I could access the generation script

# TOOL 1: COMPUTATIONAL NOTEBOOKS/LITTERATE PROGRAMMING

## Un document computationnel

Mon ordinateur m'indique que  $\pi$  vaut approximativement

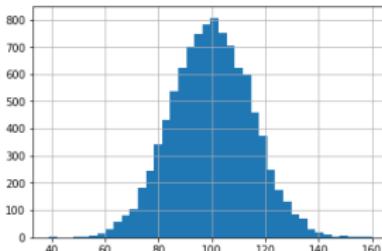
3.141592653589793

Mais calculé avec la méthode des [aiguilles de Buffon](#), on obtiendrait comme approximation :

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

3.1437198694098765

On peut inclure des formules mathématiques comme  $\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$  et des dessins qui n'ont rien à voir avec  $\pi$  (si ce n'est une constante de normalisation... ☺).



# TOOL 1: COMPUTATIONAL NOTEBOOKS/LITTERATE PROGRAMMING

Document initial dans son environnement

The screenshot shows a Jupyter Notebook interface with the following details:

- Title:** # Un document computationnel
- In [1]:** A code cell containing:

```
from math import *
print(pi)
3.141592653589793
```

A note below it says: "Mais calculé avec la [méthode des aiguilles de Buffon](#) ([https://fr.wikipedia.org/wiki/Aiguille\\_de\\_Buffon](https://fr.wikipedia.org/wiki/Aiguille_de_Buffon)), on obtient l'air comme approximation :".
- In [2]:** A code cell containing:

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

A note below it says: "On peut inclure des formules mathématiques comme  $\sqrt{2/\pi} \exp(-x^2/2)$  et des dessins qui n'ont rien à voir avec  $\pi$  (si ce n'est une constante de normalisation...)."
- In [3]:** A code cell containing:

```
%matplotlib inline
import matplotlib.pyplot as plt
mu, sigma = 100, 15
x = mu + sigma*np.random.randn(10000)
plt.hist(x,40)
plt.grid(True)
plt.show()
```

A histogram plot is shown, centered at 100 with a peak around 100.

Document final

## Un document computationnel

Mon ordinateur m'indique que  $\pi$  vaut approximativement

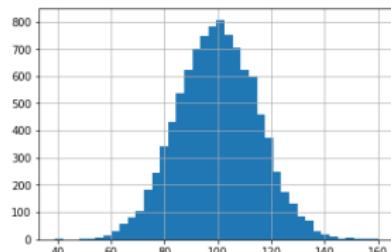
3.141592653589793

Mais calculé avec la [méthode des aiguilles de Buffon](#), on obtiendrait comme approximation :

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

3.1437198694998765

On peut inclure des formules mathématiques comme  $\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$  et des dessins qui n'ont rien à voir avec  $\pi$  (si ce n'est une constante de normalisation...).



# TOOL 1: COMPUTATIONAL NOTEBOOKS/LITTERATE PROGRAMMING

Document initial dans son environnement

The screenshot shows a Jupyter Notebook interface with three code cells:

- In [1]:** A cell containing Python code to print the value of pi.

```
from math import *
print(pi)
3.141592653589793
```
- In [2]:** A cell calculating the value of pi using theBuffon's needle method.

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

Out [2]: 3.14371986944998765
- In [3]:** A cell generating a histogram of random numbers.

Annotations in red:

- A red arrow points from the text "MarkDown" to the first cell, which contains a block of text explaining the Buffon's needle method.
- Two red arrows point from the text "MarkDown" to the second cell, which contains a block of text explaining the Buffon's needle method.

Document final

## Un document computationnel

Mon ordinateur m'indique que  $\pi$  vaut approximativement

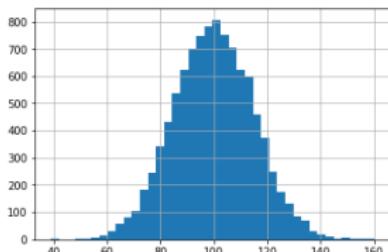
3.141592653589793

Mais calculé avec la [méthode des aiguilles de Buffon](#), on obtiendrait comme approximation :

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

3.14371986944998765

On peut inclure des formules mathématiques comme  $\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$  et des dessins qui n'ont rien à voir avec  $\pi$  (si ce n'est une constante de normalisation... ☺).



# TOOL 1: COMPUTATIONAL NOTEBOOKS/LITTERATE PROGRAMMING

Document initial dans son environnement

A screenshot of a Jupyter Notebook interface. The title bar says "jupyter example\_pi". The notebook has three cells:

- In [1]:** Contains the Python code `from math import \*` and `print(pi)`. The output is "3,141592653589793".

Mon ordinateur m'indique que  $\pi$  vaut approximativement

Mais calculé avec la [méthode des algilles de Buffon](https://fr.wikipedia.org/wiki/Algille_de_Buffon) ([https://fr.wikipedia.org/wiki/Algille\\_de\\_Buffon](https://fr.wikipedia.org/wiki/Algille_de_Buffon)), on obtiendrait comme approximation :
- In [2]:** Contains the Python code for the Buffon's needle simulation. The output is "3,143719869498765".

On peut inclure des formules mathématiques comme  $\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$  et des dessins qui n'ont rien à voir avec  $\pi$  (si ce n'est une constante de normalisation...).
- In [3]:** Contains the Python code for plotting a histogram. The output is a histogram showing a bell-shaped distribution centered around 100.

Document final

## Un document computationnel

Mon ordinateur m'indique que  $\pi$  vaut approximativement

3.141592653589793

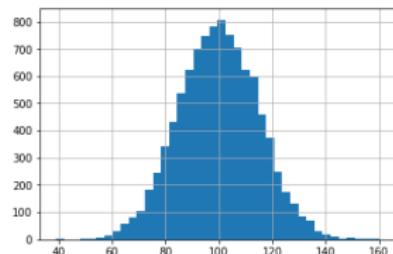
Mais calculé avec la [méthode des algilles de Buffon](#), on obtiendrait comme approximation :

```
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)
2*(sum((x+np.sin(theta))>1))/N
```

3.143719869498765

On peut inclure des formules mathématiques comme  $\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$  et

des dessins qui n'ont rien à voir avec  $\pi$  (si ce n'est une constante de normalisation...).



# TOOL 1: COMPUTATIONAL NOTEBOOKS/LITTERATE PROGRAMMING

Document initial dans son environnement

A screenshot of a Jupyter Notebook interface. The top bar shows 'jupyter example\_pi' and 'Python 3'. The notebook contains three code cells:

- In [1]:** Prints the value of pi: `print(pi)` followed by `3.141592653589793`. A note below explains that this is an approximation.
- In [2]:** Generates random points (x, y) in the unit square [0,1] x [0,1] and calculates the ratio of points where  $y \leq \sin(\pi x)$  to the total number of points. The output is `3.143719869498765`.
- In [3]:** Plots a histogram of 100,000 random points. The x-axis ranges from 40 to 160, and the y-axis ranges from 0 to 800. The distribution is approximately normal.

Annotations with red arrows point from the text 'Résultats' to the output of cell In [1] and the histogram in cell In [3].

Document final

## Un document computationnel

Mon ordinateur m'indique que  $\pi$  vaut approximativement

3.141592653589793

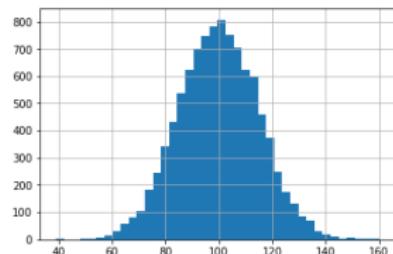
Mais calculé avec la méthode des [aiguilles de Buffon](#), on obtiendrait comme approximation :

```
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)
2*(sum((x+np.sin(theta))>1))/N
```

3.143719869498765

On peut inclure des formules mathématiques comme  $\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$  et

des dessins qui n'ont rien à voir avec  $\pi$  (si ce n'est une constante de normalisation... ☺).



# TOOL 1: COMPUTATIONAL NOTEBOOKS/LITTERATE PROGRAMMING

Document initial dans son environnement

A screenshot of a Jupyter notebook interface. The top bar shows 'jupyter example\_pi' and 'Python 3'. The notebook has three cells:

- In [1]:** Prints  $\pi$  to the console.
- In [2]:** Calculates the value of  $\pi$  using Buffon's needle method. It includes a note about the formula and the result is approximately 3.1437198694098765.
- In [3]:** Plots a histogram of 100,000 random numbers between 0 and 100, showing a bell-shaped distribution centered around 50.

Document final

## Un document computationnel

Mon ordinateur m'indique que  $\pi$  vaut approximativement

3.141592653589793

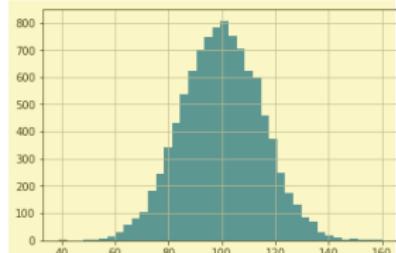
Mais calculé avec la **méthode des aiguilles de Buffon**, on obtiendrait comme approximation :

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

3.1437198694098765

Export

On peut inclure des formules mathématiques comme  $\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$  et des dessins qui n'ont rien à voir avec  $\pi$  (si ce n'est une constante de normalisation... ☺).



# TOOL 1: COMPUTATIONAL NOTEBOOKS/LITTERATE PROGRAMMING

Document initial dans son environnement

The screenshot shows a Jupyter Notebook interface with three code cells:

- In [1]:** Prints the value of  $\pi$  as 3.141592653589793.
- In [2]:** Generates random points and calculates the ratio of points inside a unit circle to the total number of points, which is approximately 3.14198694098765.
- In [3]:** Plots a histogram of 100,000 random numbers between 0 and 1, showing a bell-shaped distribution centered at 0.5.

Document final

## Un document computationnel

Mon ordinateur m'indique que  $\pi$  vaut approximativement

3.141592653589793

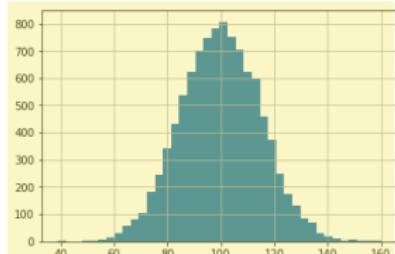
Mais calculé avec la **méthode des aiguilles de Buffon**, on obtiendrait comme approximation :

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

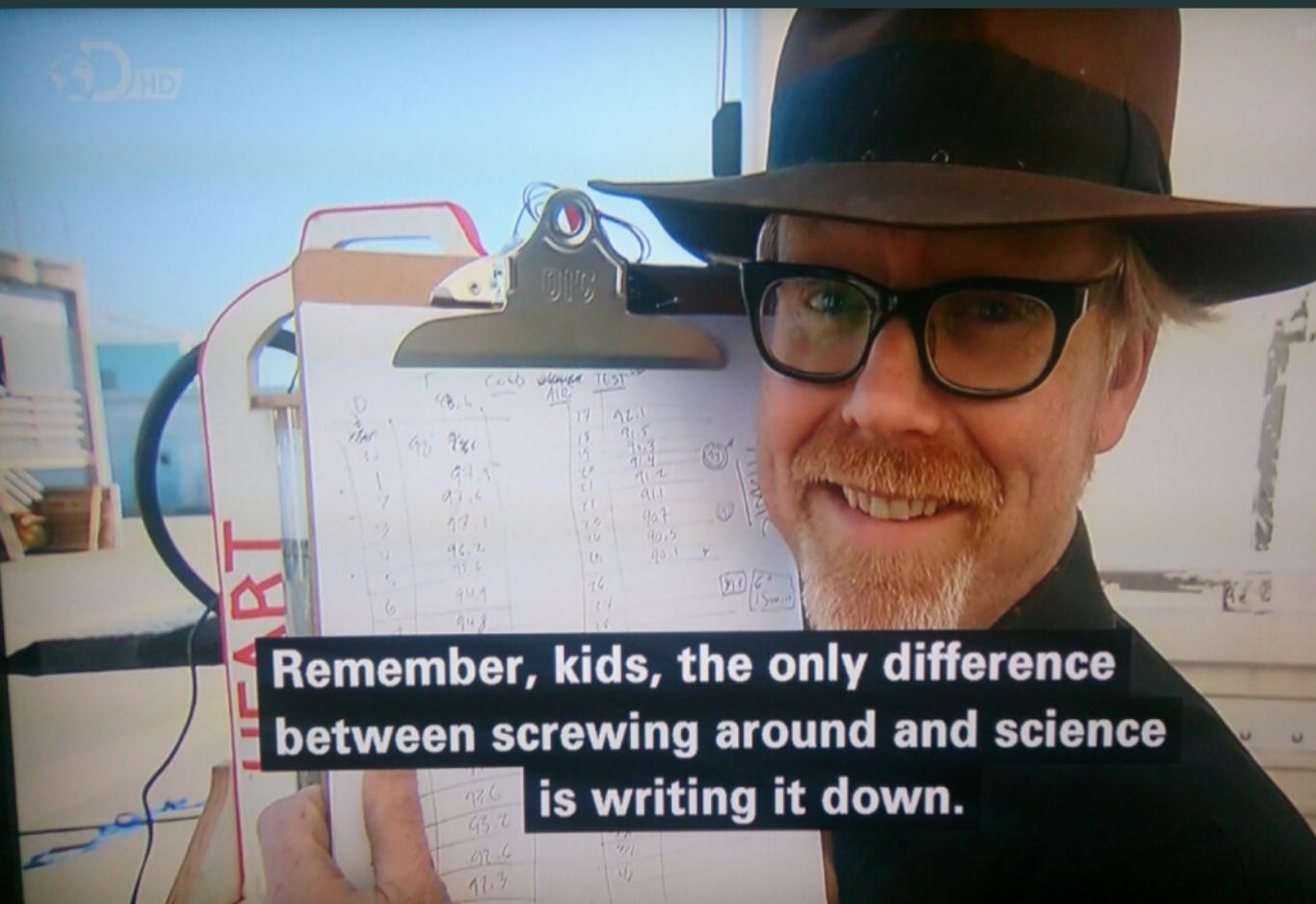
3.1437198694098765

Export

On peut inclure des formules mathématiques comme  $\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$  et des dessins qui n'ont rien à voir avec  $\pi$  (si ce n'est une constante de normalisation... ☺).



## TOOL 1 BIS: LABORATORY NOTEBOOKS, COMPUTATIONAL DOCUMENTS



**Remember, kids, the only difference between screwing around and science is writing it down.**

# TOOL 1 TER: WORKFLOWS

Notebooks are no panacea and do not help developing clean code

jupyter example\_pi [recent]

File Edit View Insert Cell Kernel Widgets Help Hide Code Hide Outputs Python 3 Cell Toolbar

# Un document computationnel

Mon ordinateur m'indique que  $\pi$  vaut "approximativement"

In [1]:

```
from math import *  
print(pi)  
3.141592653589793
```

Mais calculé avec la `__method__` des (ajoutées de Buffet) `__approximation__`, on obtiendrait comme

In [2]:

```
import numpy as np  
n = 1000000  
x = np.random.uniform(0, low=0, high=1)  
theta = np.random.uniform(0, low=0, high=np.pi/2)  
((1/(theta)) * np.sin(theta)) > 1/N
```

Out[2]: 0.1437120864495075

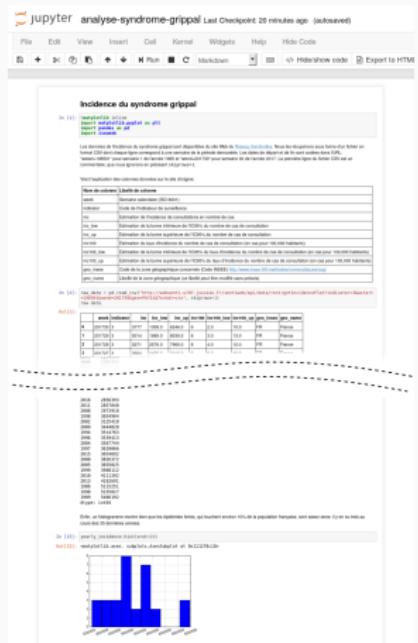
On peut inclure des formules mathématiques comme  $\frac{1}{\theta} \sin(\theta)$  ou  $\frac{1}{\theta}$  dans un cellulaire et l'interpréteur les comprendra et nous donnera le résultat attendu (ce qui n'est rien à voir avec `eval` si ce n'est une constante de normalisation...).

In [3]:

```
%matplotlib inline  
import matplotlib.pyplot as plt  
  
N = 1000000  
x = np.random.uniform(0, high=1)  
theta = np.random.uniform(0, high=np.pi/2)  
y = 1/(theta) * np.sin(theta)  
plt.hist(x, 40)  
plt.hist(y, 40)  
plt.show()
```

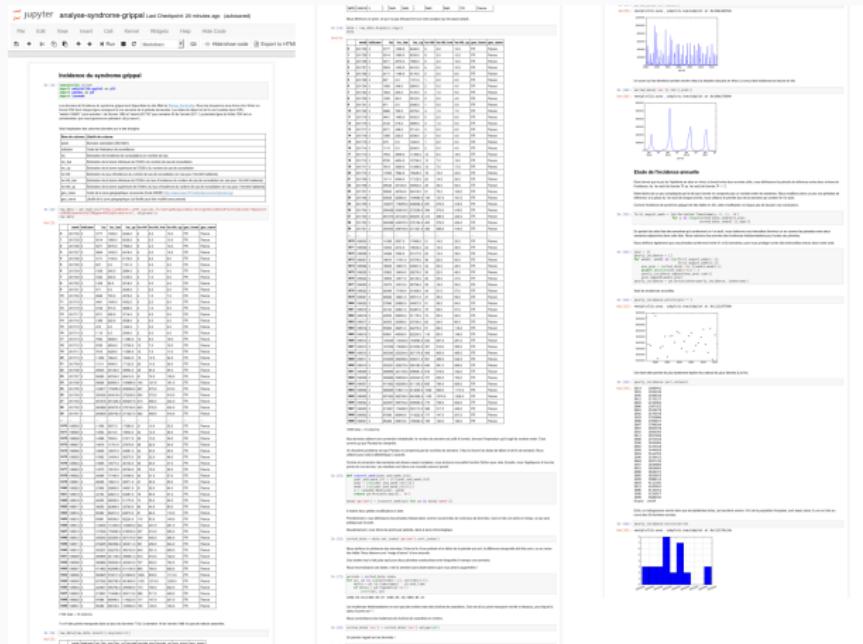
## TOOL 1 TER: WORKFLOWS

Notebooks are no panacea and do not help developing clean code



# TOOL 1 TER: WORKFLOWS

Notebooks are no panacea and do not help developing clean code



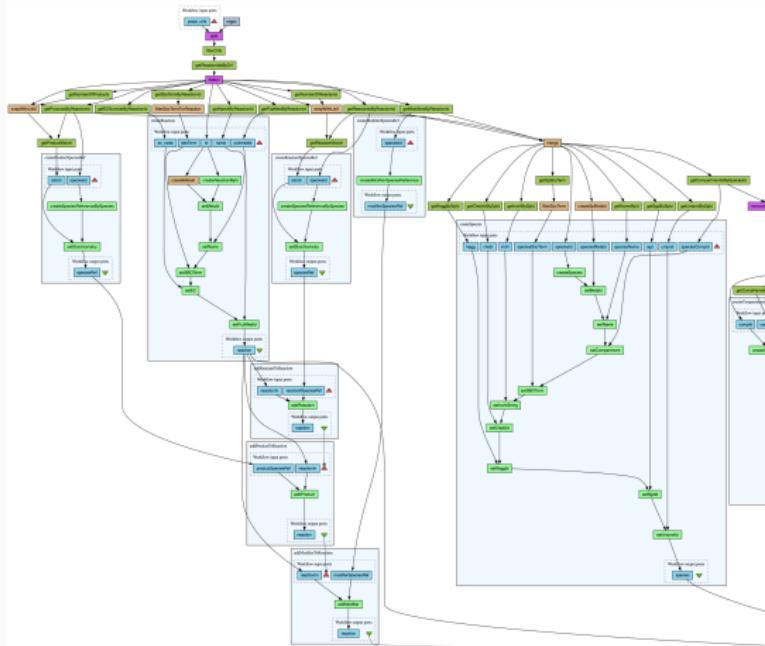
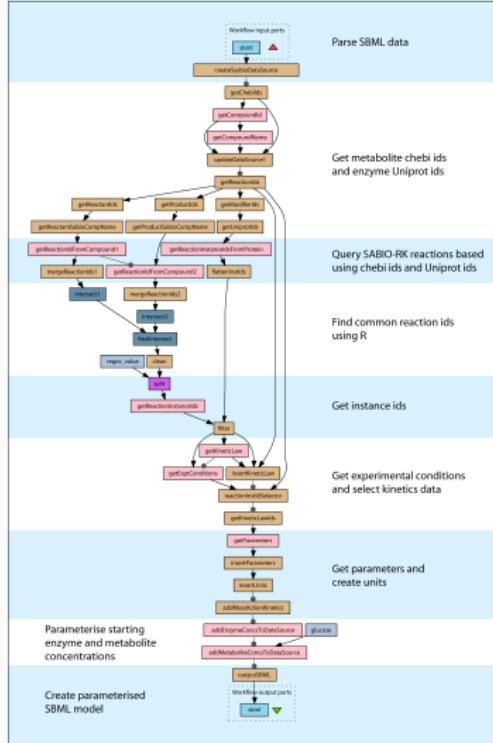
# TOOL 1 TER: WORKFLOWS

Notebooks are no panacea and do not help developing clean code

The image displays a 4x3 grid of Jupyter Notebook screenshots, each illustrating a different step or aspect of a workflow. The notebooks cover topics such as estimating color names from web images, performing k-means clustering, visualizing color distributions, and analyzing machine learning model results.

- Estimating Color Names by Web Image**: A notebook titled "Estimating Color Names by Web Image" shows code for extracting colors from a URL and comparing them with a color palette.
- Color Clustering and K-Means**: A notebook titled "Color Clustering and K-Means" demonstrates how to cluster colors using k-means and visualize the results.
- Color Clustering and K-Means (continued)**: A continuation of the previous notebook, showing more detailed clustering and visualization steps.
- Color Clustering and K-Means (final)**: A final notebook in the series, concluding the color clustering process.
- Color Clustering and K-Means (final) - Results**: A notebook titled "Color Clustering and K-Means (final) - Results" displays the final clustered data and a heatmap of the color distribution.
- Dimensionality Reduction and Model Results**: A notebook titled "Dimensionality Reduction and Model Results" shows dimensionality reduction and model results.
- Dimensionality Reduction and Model Results (continued)**: A continuation of the previous notebook, focusing on dimensionality reduction and model results.
- Dimensionality Reduction and Model Results (final)**: A final notebook in the series, concluding the dimensionality reduction and model results process.
- Dimensionality Reduction and Model Results (final) - Results**: A notebook titled "Dimensionality Reduction and Model Results (final) - Results" displays the final dimensionality reduction and model results.
- Prediction**: A notebook titled "Prediction" shows code for predicting colors based on input features.
- Predicting**: A notebook titled "Predicting" shows code for predicting colors based on input features.
- Visualizing the training data**: A notebook titled "Visualizing the training data" shows a scatter plot of training data points.
- Visualizing the training data (continued)**: A continuation of the previous notebook, showing more details about the training data visualization.
- Visualizing the training data (final)**: A final notebook in the series, concluding the training data visualization process.
- Visualizing the training data (final) - Results**: A notebook titled "Visualizing the training data (final) - Results" displays the final training data visualization results.
- Visualizing the test data**: A notebook titled "Visualizing the test data" shows a scatter plot of test data points.
- Visualizing the test data (continued)**: A continuation of the previous notebook, showing more details about the test data visualization.
- Visualizing the test data (final)**: A final notebook in the series, concluding the test data visualization process.
- Visualizing the test data (final) - Results**: A notebook titled "Visualizing the test data (final) - Results" displays the final test data visualization results.
- Modeling the test data**: A notebook titled "Modeling the test data" shows code for modeling test data.
- Modeling the test data (continued)**: A continuation of the previous notebook, showing more details about the test data modeling.
- Modeling the test data (final)**: A final notebook in the series, concluding the test data modeling process.
- Modeling the test data (final) - Results**: A notebook titled "Modeling the test data (final) - Results" displays the final test data modeling results.
- Testing the model**: A notebook titled "Testing the model" shows code for testing the model.
- Testing the model (continued)**: A continuation of the previous notebook, showing more details about the model testing.
- Testing the model (final)**: A final notebook in the series, concluding the model testing process.
- Testing the model (final) - Results**: A notebook titled "Testing the model (final) - Results" displays the final model testing results.
- Wrapping the results**: A notebook titled "Wrapping the results" shows code for summarizing the results.
- Wrapping the results (continued)**: A continuation of the previous notebook, showing more details about the results wrapping.
- Wrapping the results (final)**: A final notebook in the series, concluding the results wrapping process.
- Wrapping the results (final) - Results**: A notebook titled "Wrapping the results (final) - Results" displays the final results wrapping results.
- Conclusion**: A notebook titled "Conclusion" provides a summary of the findings.
- Conclusion (continued)**: A continuation of the previous notebook, providing more details about the conclusion.
- Conclusion (final)**: A final notebook in the series, concluding the overall workflow.
- Conclusion (final) - Results**: A notebook titled "Conclusion (final) - Results" displays the final conclusion results.

# TOOL 1 TER: WORKFLOWS



## Workflows:

- Clearer high-level view
- **Explicit** composition of codes and data movement
- Safer sharing, reusing, and execution
- Notebooks are a variant that is both impoverished and richer
  - No simple/mature path from a notebook to a workflow

## Examples:

- Galaxy, Kepler, Taverna, Pegasus, Collective Knowledge, VisTrails
- Light-weight: `make`, dask, drake, swift, `snakemake`, ...
- Hybrids: SOS-notebook, ...

## GOOD PRACTICE #2

### CONTROLLING SOFTWARE ENVIRONMENT

---

## ARGH... DAMNED COMPUTERS

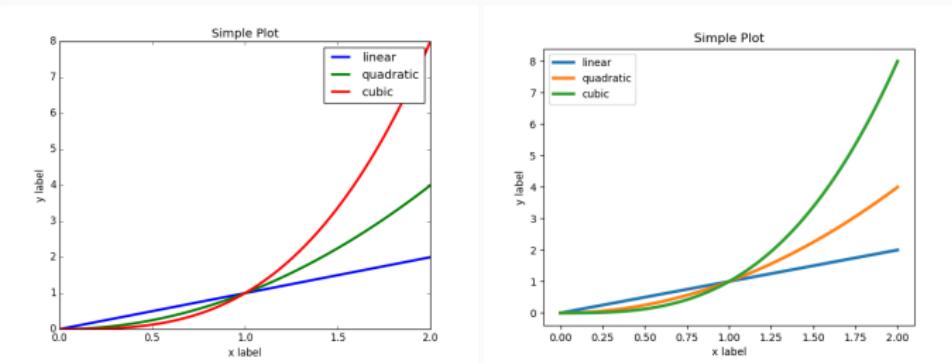
- Alice: I got 3.123123 Bob: I got segfault
  - Damned! It used to work!!! Whenever I upgrade my computer, things break so I try to stay away from this 😞
  - Whenever trying the code of my colleague, I had to install **libFoo-1.5c** but I broke everything and now neither his code nor mine works! 😞
  - But hey! Here is my code. It's on GitHub so feel free to play with it! I'm doing open science 😊
    1. No one will ever run/use your code if it isn't easy to install
    2. No one will ever manage to run your code if you don't document how to run it
    3. Others (even you) are unlikely to get the same results unless you control and share your software environment

## SOFTWARE DEPENDENCIES: HORROR STORIES

---

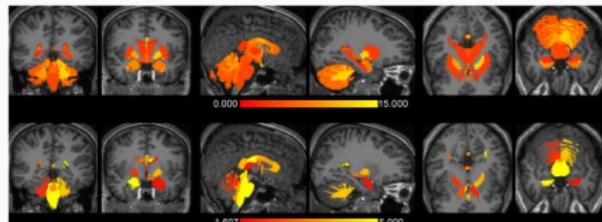
# SOFTWARE DEPENDENCIES: HORROR STORIES

- Software environment evolution



# SOFTWARE DEPENDENCIES: HORROR STORIES

- Software environment evolution
- OS heterogeneity



The Effects of FreeSurfer Version, Workstation Type, and Macintosh Operating System Version on Anatomical Volume and Cortical Thickness Measurements (PLOS ONE, 2012)

*Significant differences in volume and cortical thickness were revealed across FreeSurfer versions:*

- volume:  $8.8 \pm 6.6\%$  (range 1.3-**64.0%**)
- cortical thickness:  $2.8 \pm 1.3\%$  (range 1.1-7.7%)

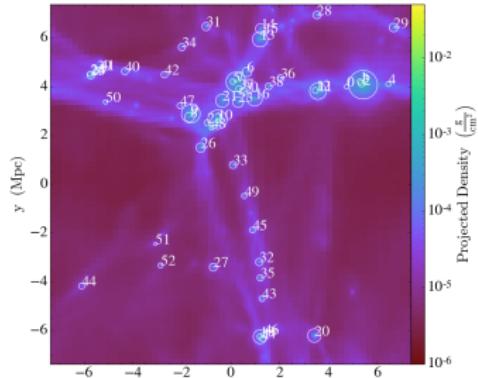
*About a factor two smaller differences were found between the Mac and HP workstations and between Mac OSX 10.5 and OSX 10.6.*

*In the context of an ongoing study, users are discouraged to update to a new major release of either FreeSurfer or operating system.*

*Formal assessment of the accuracy of FreeSurfer is desirable.*

# SOFTWARE DEPENDENCIES: HORROR STORIES

- Software environment evolution
- OS heterogeneity
- Impact of the compiler

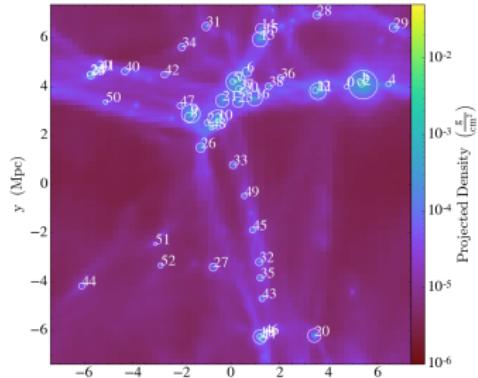


Assessing Reproducibility: An Astrophysical Example of Computational Uncertainty in the HPC Context (ResCuE-HPC, 2018)

| Compiler  | Optim. | Largest Halo<br>Avg Mass. | Std. Err  | Walltime |
|-----------|--------|---------------------------|-----------|----------|
| gcc@6.2.0 | None   | 2.273E 46                 | 1.069E 44 | 22h      |

# SOFTWARE DEPENDENCIES: HORROR STORIES

- Software environment evolution
- OS heterogeneity
- Impact of the compiler

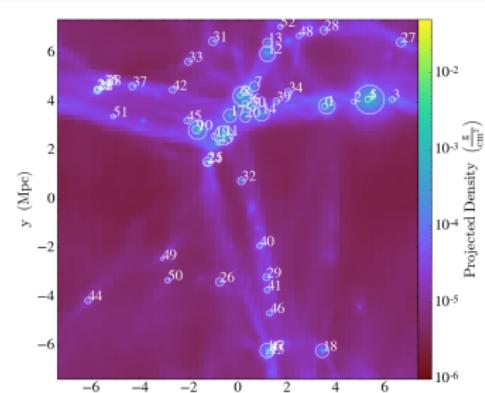


Assessing Reproducibility: An Astrophysical Example of Computational Uncertainty in the HPC Context (ResCuE-HPC, 2018)

| Compiler  | Optim. | Largest Halo<br>Avg Mass. | Std. Err  | Walltime |
|-----------|--------|---------------------------|-----------|----------|
| gcc@6.2.0 | None   | 2.273E 46                 | 1.069E 44 | 22h      |

# SOFTWARE DEPENDENCIES: HORROR STORIES

- Software environment evolution
- OS heterogeneity
- Impact of the compiler

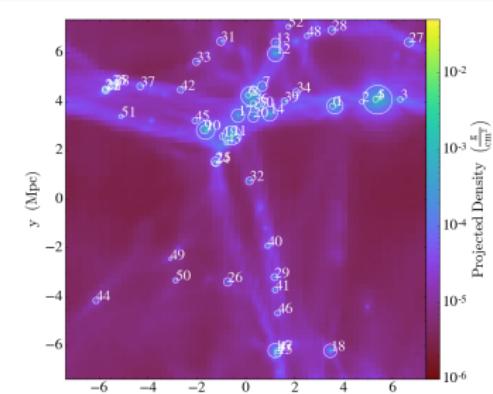


Assessing Reproducibility: An Astrophysical Example of Computational Uncertainty in the HPC Context (ResCuE-HPC, 2018)

| Compiler  | Optim. | Largest Halo |           | Walltime |
|-----------|--------|--------------|-----------|----------|
|           |        | Avg Mass.    | Std. Err  |          |
| gcc@6.2.0 | None   | 2.273E 46    | 1.069E 44 | 22h      |
| gcc@6.2.0 | Normal | 2.266E 46    | 1.218E 44 | 10h      |
| gcc@6.2.0 | High   | 2.275E 46    | 1.199E 44 | 9h       |

# SOFTWARE DEPENDENCIES: HORROR STORIES

- Software environment evolution
- OS heterogeneity
- Impact of the compiler



Assessing Reproducibility: An Astrophysical Example of Computational Uncertainty in the HPC Context (ResCuE-HPC, 2018)

| Compiler     | Optim. | Largest Halo      |           | Walltime |
|--------------|--------|-------------------|-----------|----------|
|              |        | Avg Mass.         | Std. Err  |          |
| gcc@6.2.0    | None   | 2.273E 46         | 1.069E 44 | 22h      |
| gcc@6.2.0    | Normal | 2.266E 46         | 1.218E 44 | 10h      |
| gcc@6.2.0    | High   | 2.275E 46         | 1.199E 44 | 9h       |
| intel@16.0.3 | None   | <b>22.71</b> E 46 | 1.587E 44 | 39h      |
| intel@16.0.3 | Normal | <b>43.30</b> E 46 | 1.248E 44 | 7h       |
| intel@16.0.3 | High   | 2.268E 46         | 1.414E 44 | 6h       |
| cce@8.5.5    | Low    | <b>43.11</b> E 46 | 1.353E 44 | 16h      |
| cce@8.5.5    | Normal | 2.271E 46         | 1.261E 44 | 6h       |
| cce@8.5.5    | High   | 2.272E 46         | 1.341E 44 | 5h       |

# COMPLEX ECOSYSTEMS

```
1 import matplotlib  
2 print(matplotlib.__version__)
```

3.5.1

# COMPLEX ECOSYSTEMS

```
1 import matplotlib  
2 print(matplotlib.__version__)
```

## 3.5.1

```
1 apt show python3-matplotlib
```

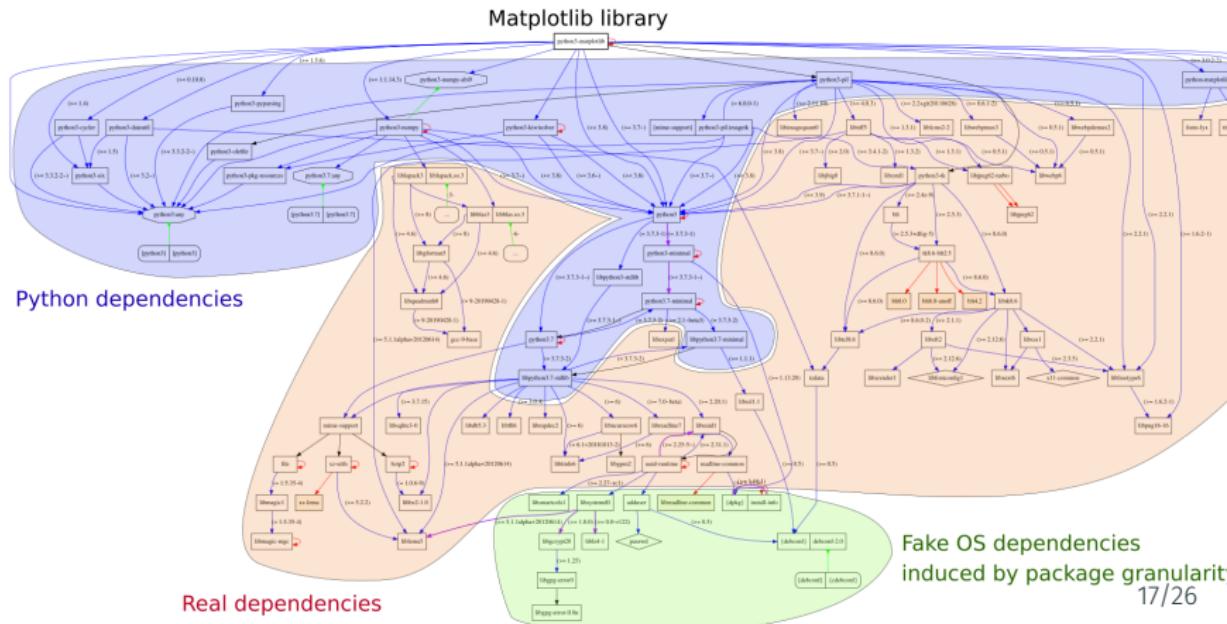
Package: python3-matplotlib  
Version: 3.5.1-2+b1  
Source: matplotlib (3.5.1-2)  
Maintainer: Sandro Tosi <morph@debian.org>  
Installed-Size: 27.6 MB  
Depends: libjs-jquery, libjs-jquery-ui, python-matplotlib-data (>= 3.5.1),  
python3-dateutil, python3-pil.imagetk, python3-pyparsing (>= 1.5.6),  
python3-six (>= 1.4), python3-numpy (>= 1:1.20.0), python3-numpy-abi9,  
python3 (<< 3.11), python3 (>= 3.9~), python3-cycler (>= 0.10.0),  
python3-fonttools, python3-kiwisolver, python3-packaging, python3-pil,  
python3:any, libc6 (>= 2.29), libfreetype6 (>= 2.2.1),  
libgcc-s1 (>= 3.3.1), libqhull-r8.0 (>= 2020.1), libstdc++6 (>= 11)  
Recommends: python3-tk  
Suggests: dvipng, ffmpeg, fonts-staypuft, ghostscript, gir1.2-gtk-3.0, inkscape,  
ipython3, librsvg2-common, python-matplotlib-doc, python3-cairocffi,  
python3-gi, python3-gi-cairo, python3-gobject, python3-pyqt5, 17/26  
python3-scipy, python3-sip, python3-tornado, texlive-extra-utils,  
liblouis0

# COMPLEX ECOSYSTEMS

```
1 import matplotlib  
2 print(matplotlib.__version__)
```

## 3.5.1

```
1 apt show python3-matplotlib
```



## TOOL 2: CONTAINERS AND PACKAGE MANAGERS

The good



The bad



The ugly



Automatic tracking

## TOOL 2: CONTAINERS AND PACKAGE MANAGERS

The good



The bad



The ugly



Automatic tracking

Containers

- Pros: Lightweight, Good isolation, Easy to use
  - Running as easy as `docker run <img> <cmd>`
  - Building images: `docker build -f <Dockerfile>`
  - Sharing through the Docker Hub: `docker pull/push <img>`

## TOOL 2: CONTAINERS AND PACKAGE MANAGERS

The good



The bad



The ugly



Automatic tracking

### Containers

- **Pros:** Lightweight, Good isolation, Easy to use
- **Cons:** Opaque, Container build is generally not reproducible
  - Recipes rarely follow *reproducible good practices*

```
1 FROM ubuntu:20.04
2 RUN apt-get update
3     && apt-get upgrade -y
4     && apt-get install -y ...
```

- Choose a stable image (and the smallest possible)
- Include only the necessary libraries (e.g. no graphics libs)
- Avoid system updates (instead freeze sources)

## TOOL 2: CONTAINERS AND PACKAGE MANAGERS

The good



The bad



The ugly



Automatic tracking

Containers

- Pros: Lightweight, Good isolation, Easy to use
- Cons: Opaque, Container build is generally not reproducible

Package managers (the ugly and the good)

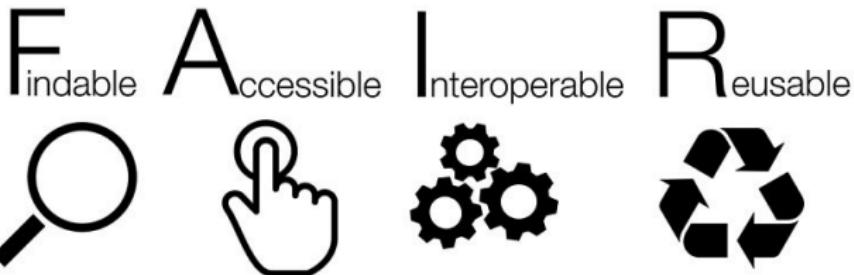
- Language specific: `pip/pipenv/virtualenv`, `conda`, `CRAN/Bioconductor`
  - Limits: version management, durability, permeable, language centric
- **GUIX/NiX** = Full-fledged functional package manager
  - Native support for environment (*à la git*)
  - Isolation through `--pure`
  - Recompile from source (cache recommended)

## GOOD PRACTICE #3

## VERSION CONTROL AND ARCHIVING

---

# FAIR PRINCIPLES



<https://www.go-fair.org/fair-principles/>

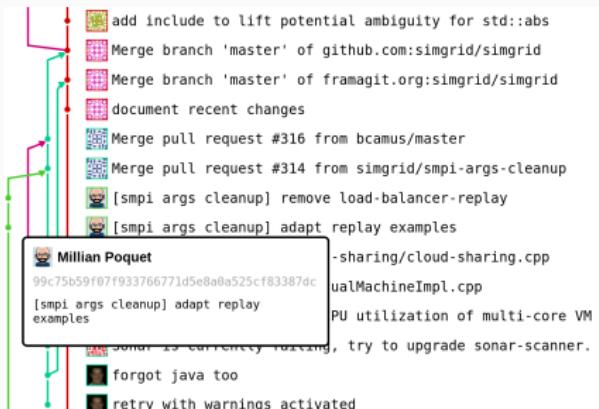
- "Open as much as possible and close as much as necessary"
- Management, publication, annotation (metadata), archiving
- Source code = specific data with specific consideration

Let's go beyond general principles!

# TOOL 3: VERSION CONTROL AND FORGE

## Git = version control

- Developed in 2005 by Linus Torvalds for the kernel development
- Local and efficient rollbacks
- Distributed: everyone has a full copy of the history



## GitHub, GitLab, and Co

- Free hosting of public projects, social network



## Limitation

- Managing large data: **Git LFS**   **Git Annex** (or DataLad)

## TOOL 3BIS: FIGHTING INFORMATION LOSS WITH ARCHIVES



or



= awesome collaborations ( $\neq$  archive)

- D. Spinellis. *The Decay and Failures of URL References*. CACM, 46(1), 2003  
*The half-life of a referenced URL is approximately 4 years from its publication date.*
- 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  
*half life ranged from 2.2 years in EMHJ to 5.3 years in BMJ*
- Discontinued forges: Code Space, Gitorious, Google code, Inria Gforge

## TOOL 3BIS: FIGHTING INFORMATION LOSS WITH ARCHIVES



or



= awesome collaborations ( $\neq$  archive)

- D. Spinellis. *The Decay and Failures of URL References*. CACM, 46(1), 2003  
*The half-life of a referenced URL is approximately 4 years from its publication date.*
- 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  
*half life ranged from 2.2 years in EMHJ to 5.3 years in BMJ*
- Discontinued forges: Code Space, Gitorious, Google code, Inria Gforge

Article archives



Data archives



figshare



Software Archive



Software Heritage

Collect/Preserve/Share

WHAT WILL IT TAKE ?

---

# CHANGING RESEARCH PRACTICES

## Soft. Engineering, Statistics, and Reproducible Research in the curricula

**Manifesto:** "*I solemnly pledge*" ([WSSSPE](#), [Lorena Barba](#), [FAIR](#))

1. I will teach my graduate students about reproducibility
2. All our research code (and writing) is under version control
3. We will always carry out verification and validation
4. We will share data, plotting script & figure under CC-BY
5. We will upload the preprint to arXiv at the time of submission of a paper
6. We will release code at the time of submission of a paper
7. We will add a "Reproducibility" declaration at the end of each paper
8. I will keep an up-to-date web presence



Learn and Teach using online resources like

- [Software Carpentry](#), [The Turing Way](#), ...

## Artifact evaluation and ACM badges



## Major conferences

- Supercomputing: Artifact Description (AD) mandatory, Artifact Evaluation (AE) still optional, Double blind vs. RR
- NeurIPS, ICLR: open reviews, reproducibility challenge



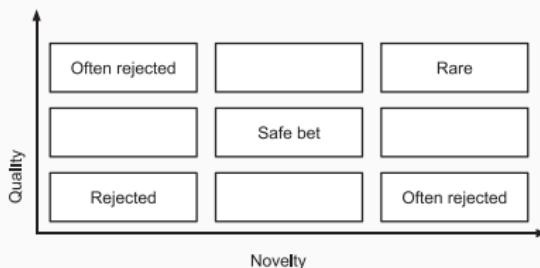
Joelle Pineau @ NeurIPS'18

- ACM SIGMOD 2015-2019, Most Reproducible Paper Award...

Mentalities are evolving people care, make stuff available, errors are found and fixed

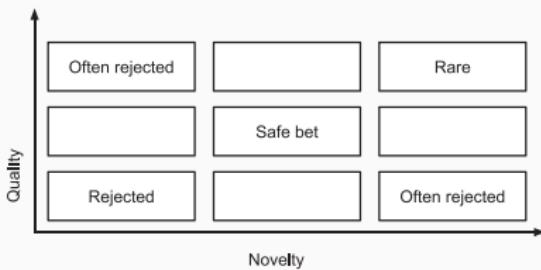
# CHANGING ACADEMIC PRACTICES (PUBLISH OR PERISH)

- Goodhart's Law: Are Academic Metrics Being Gamed?, M. Fire 2019
  - AI: over 1,000 ranked journals ( $\times 10$  in 15 years)
  - Shorter papers with increasing self references
  - More and more papers without any citation
  - Sharp increase in the number of new authors publishing at a much faster rate given their career age
- The Truth, The Whole Truth, and Nothing But the Truth: A Pragmatic, Guide to Assessing Empirical Evaluations, TOPLAS 2016



# CHANGING ACADEMIC PRACTICES (PUBLISH OR PERISH)

- Goodhart's Law: Are Academic Metrics Being Gamed?, M. Fire 2019
  - AI: over 1,000 ranked journals ( $\times 10$  in 15 years)
  - Shorter papers with increasing self references
  - More and more papers without any citation
  - Sharp increase in the number of new authors publishing at a much faster rate given their career age
- The Truth, The Whole Truth, and Nothing But the Truth: A Pragmatic, Guide to Assessing Empirical Evaluations, TOPLAS 2016



- Impact factor abandoned by Dutch university in hiring and promotion, decisions. Nature, June 2021. Faculty and staff members at Utrecht University will be evaluated by their commitment to open science

# WHAT ABOUT OPEN SCIENCE ?

## Plan National pour la Science Ouverte (BSN ↽ CoSO)

- CNRS, Inria, INRAE, ...
- Many flavors: *Citizen Science*

Main pillars:

1. Open access
2. Open data
3. Open source
  - Open hardware



4. Open methodology (**Reproducible Research**)
  - Open-notebook science
  - Open science infrastructures
5. Open peer review (avoid collusion)

6. Open educational resources



**NO TRANSPARENCY  
NO CONSENSUS**

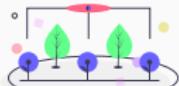


THAT'S ALL FOLKS!

---

# RESOURCES AND ACKNOWLEDGMENTS

Vers une recherche  
reproductible  
Faire évoluer ses pratiques



Loïc Desquillet, Sabrina Granger, Boris Hejblum,  
Arnaud Legrand, Pascal Pernot, Nicolas Rougier

A non-technical introduction to reproducibility issues (in French)

- Loïc Desquillet, Sabrina Granger, Boris Hejblum, Pascal Pernot, Nicolas Rougier

# RESOURCES AND ACKNOWLEDGMENTS



A non-technical introduction to reproducibility issues (in French)

- Loïc Desquillet, Sabrina Granger, Boris Hejblum, Pascal Pernot, Nicolas Rougier

MOOC Reproducible Research: Methodological principles for a transparent science, Learning Lab Inria

- Konrad Hinsen, Christophe Pouzat
- 3rd Edition: March 2020 – March 2023  
(15,000+)



Stay tuned for the MOOC "Advanced RR" planned for 2021 2022 2023

- Managing data (**FITS/HDF5, git annex**)
- Software environment control (**docker, singularity, guix**)
- Scientific workflow (**make, snakemake**)