

REPRODUCIBLE RESEARCH AND COMPUTER SCIENCE

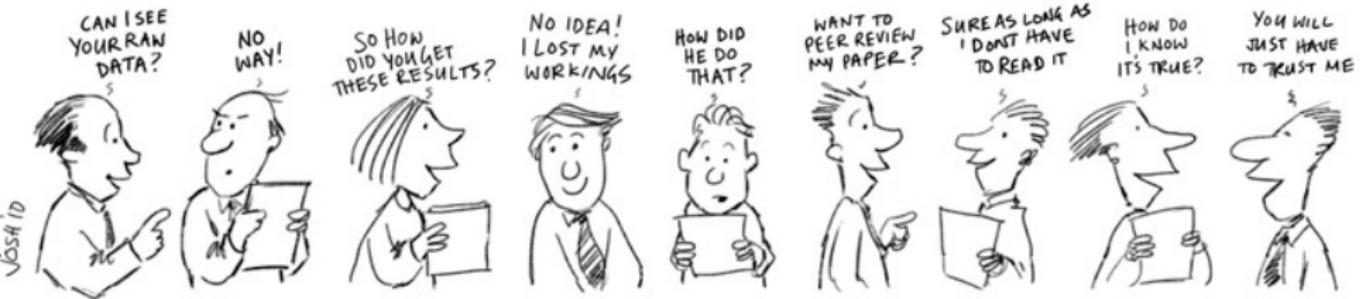
Arnaud Legrand



AVALON days, June 2023



NO TRANSPARENCY NO CONSENSUS



TOWARD OPEN SCIENCE

Plan National pour la Science Ouverte (CoSO)

- France (**CNRS**, **Inria**, **INRAE**, ...) but also Europe and US
- Many flavors: *Citizen Science vs. Ethics and Societal Responsibility*

TOWARD OPEN SCIENCE

Plan National pour la Science Ouverte (CoSO)

- France (CNRS, Inria, INRAE, ...) but also Europe and US
- Many flavors: *Citizen Science vs. Ethics and Societal Responsibility*

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
6. Open educational resources

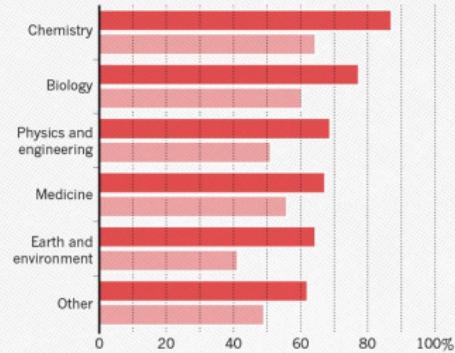


SOCIO-TECHNICAL CHALLENGES

HAVE YOU FAILED TO REPRODUCE AN EXPERIMENT?

Most scientists have experienced failure to reproduce results.

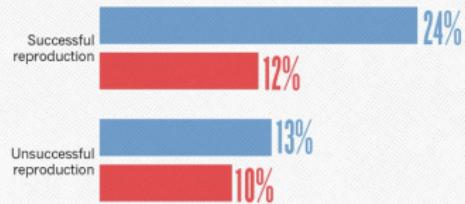
● Someone else's ● My own



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.

● Published ● Failed to publish



Number of respondents from each discipline:

Biology 703, Chemistry 106, Earth and environmental 95,

Medicine 203, Physics and engineering 236, Other 233.

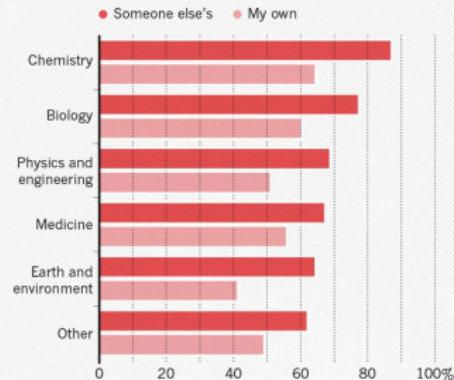
1,500 scientists lift the lid on reproducibility,

Nature, May 2016

SOCIO-TECHNICAL CHALLENGES

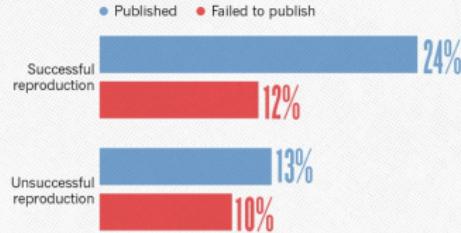
HAVE YOU FAILED TO REPRODUCE AN EXPERIMENT?

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

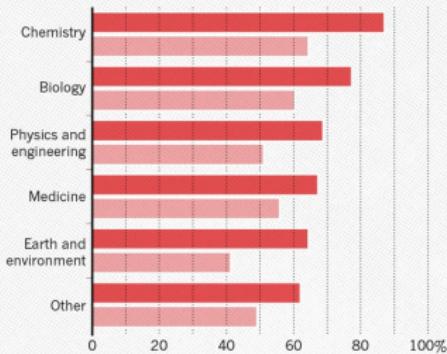
- Fraud, conflict of interest (pharmaceutic, ...)
- No incentive to reproduce/check our own work (afap), nor the work of others (big results!), nor to allow others to check (competition)
- Peer review does not scale: 1M+ articles per year!

SOCIO-TECHNICAL CHALLENGES

HAVE YOU FAILED TO REPRODUCE AN EXPERIMENT?

Most scientists have experienced failure to reproduce results.

● Someone else's ● My own



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.

● Published ● Failed to publish



Number of respondents from each discipline:

Chemistry 106, Biology 703, Physics and engineering 236, Medicine 203, Earth and environmental 95, Other 233.

1,500 scientists lift the lid on reproducibility,

Nature, May 2016

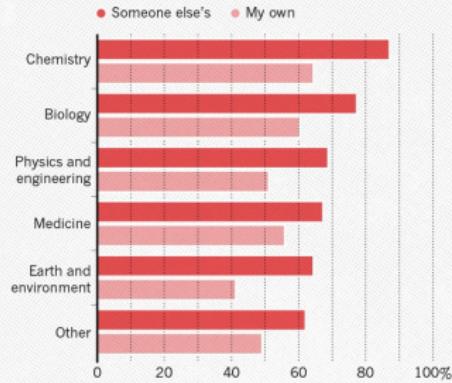
Social causes

- 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: 1M+ articles per year!
- Emerging practices: DORA/Plan S/COARA, DMP and FAIR data, artefact evaluation, reproducibility badges, reproducibility challenges, open reviews, ...

SOCIO-TECHNICAL CHALLENGES

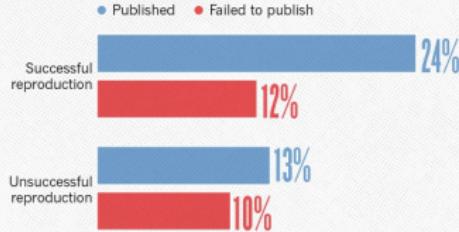
HAVE YOU FAILED TO REPRODUCE AN EXPERIMENT?

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: 1M+ articles per year!
- Emerging practices: DORA/Plan S/COARA, DMP and FAIR data, artefact evaluation, reproducibility badges, reproducibility challenges, open reviews, ...

Methodological/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

DIFFERENT REPRODUCIBILITY CONCERNS IN MODERN SCIENCE

Biology, Oncology sample provenance, clinical trials \rightsquigarrow standardized protocols

Psychology, Nutrition HARKING, p-hacking \rightsquigarrow pre-registration

DIFFERENT REPRODUCIBILITY CONCERNS IN MODERN SCIENCE

Biology, Oncology sample provenance, clinical trials \rightsquigarrow standardized protocols

Psychology, Nutrition HARKING, p-hacking \rightsquigarrow pre-registration

Genomics software engineering, computational reproducibility, provenance

Computational fluid dynamics numerical chaos, parallel architectures

DIFFERENT REPRODUCIBILITY CONCERN IN MODERN SCIENCE

Biology, Oncology sample provenance, clinical trials \rightsquigarrow standardized protocols

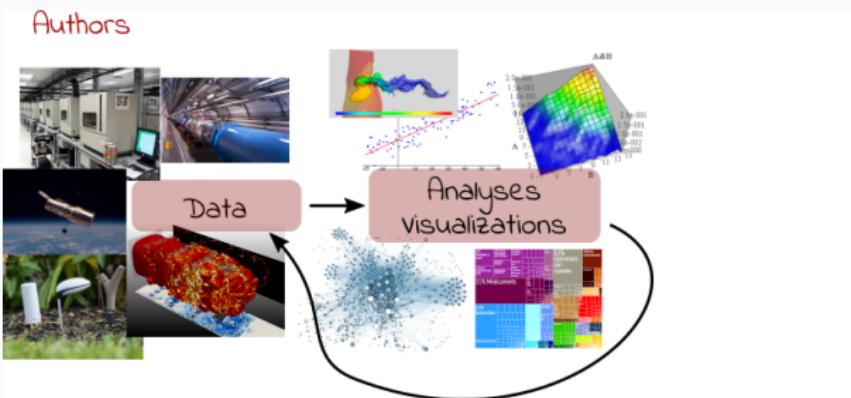
Psychology, Nutrition HARKING, p-hacking \rightsquigarrow pre-registration

Genomics software engineering, computational reproducibility, provenance

Computational fluid dynamics numerical chaos, parallel architectures

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

Biology, Oncology sample provenance, clinical trials \rightsquigarrow standardized protocols

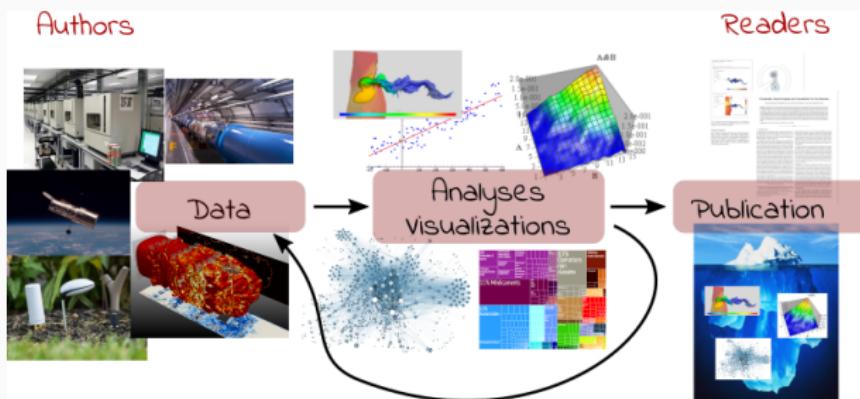
Psychology, Nutrition HARKING, p-hacking \rightsquigarrow pre-registration

Genomics software engineering, computational reproducibility, provenance

Computational fluid dynamics numerical chaos, parallel architectures

Artificial Intelligence most of the above 😊

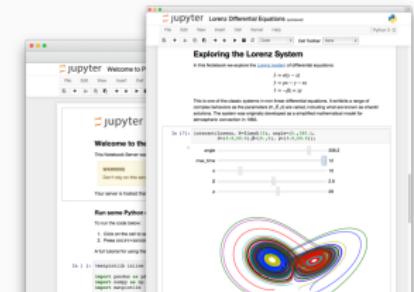
The processing steps between raw observations and findings have gotten increasingly numerous and complex



Reproducible Research = Bridging the Gap by working Transparently

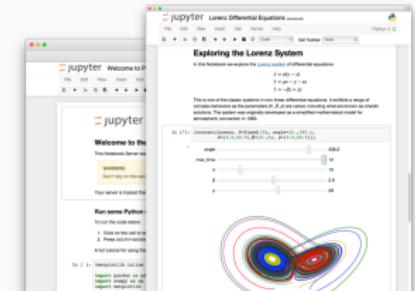
REPRODUCIBILITY ISSUES RELATED TO THE USE OF COMPUTERS

Computation provenance: notebooks and workflows

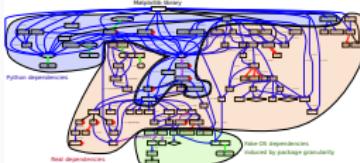


REPRODUCIBILITY ISSUES RELATED TO THE USE OF COMPUTERS

Computation provenance: notebooks and workflows



Software environments

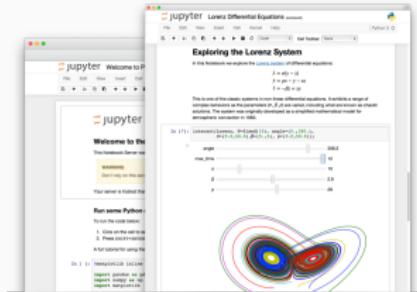


ReproZip



REPRODUCIBILITY ISSUES RELATED TO THE USE OF COMPUTERS

Computation provenance: notebooks and workflows



Software environments



Sharing and Archiving



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 π vaut approximativement

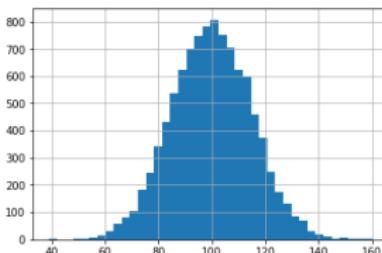
3.141592653589793

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 π (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
```
- Out [1]:** The output of the previous cell: 3.141592653589793
- 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=pi/2)
2/(sum((x*np.sin(theta))>1))/N
```
- Out [2]:** The output of the previous cell: 3.14371986944998765
- 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()
```
- Out [3]:** A histogram plot showing a bell-shaped distribution centered around 100.

Document final

Un document computationnel

Mon ordinateur m'indique que π vaut approximativement

3.141592653589793

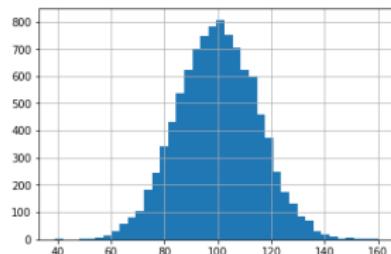
Mais calculé avec la [méthode des aiguilles de Buffon](#), on obtiendrait comme approximation :

```
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
```

3.14371986944998765

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 π (si ce n'est une constante de normalisation... ☺).



TOOL 1: COMPUTATIONAL NOTEBOOKS/LITTERATE PROGRAMMING

Document initial dans son environnement

Jupyter example_pl (modified)

Un document computationnel

Mais mon ordinateur m'indique que π vaut "approximativement"

```
In [1]:
```

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

Mais calculé avec la [méthode des aiguilles de Buffon](#) (https://fr.wikipedia.org/wiki/Aiguille_de_Buffon), on obtient aussi comme approximation :

```
In [2]:
```

```
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

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 π (si ce n'est une constante de normalisation... ☺).

```
In [3]:
```

```
%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()
```

Document final

Un document computationnel

Mon ordinateur m'indique que π vaut approximativement

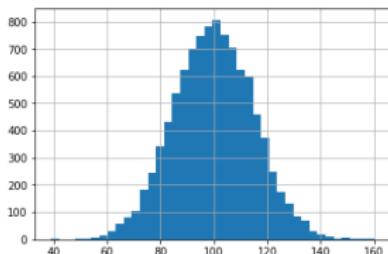
3.141592653589793

Mais calculé avec la [méthode des aiguilles de Buffon](#), on obtient 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 π (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]:** `# Un document computationnel`. The output shows the value of pi as 3,141592653589793.
- In [2]:** A more complex calculation involving numpy and random modules, resulting in the value 3,1437198694098765.
- In [3]:** A histogram plot of a normal distribution with mu=100 and sigma=15, centered at 100 with a peak frequency of approximately 800.

Annotations with red arrows point from the text 'Code' to the code blocks in the first two cells, and from the text 'Figure' to the histogram in the third cell.

Document final

Un document computationnel

Mon ordinateur m'indique que π vaut approximativement

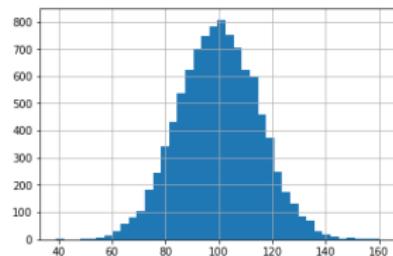
3.141592653589793

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 π (si ce n'est une constante de normalisation... ☺).



TOOL 1: COMPUTATIONAL NOTEBOOKS/LITTERATE PROGRAMMING

Document initial dans son environnement

Un document computationnel

```
In [1]:  
from math import *  
print(pi)  
3,141592653589793
```

Mais calculé avec la `_methodes_ des épinglettes de Buffon` (https://fr.wikipedia.org/wiki/Épinglette_de_Buffon), on obtiendrait comme `approximation` :

```
In [2]:  
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  
Out[2]: 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`, avec `matplotlib` (si ce n'est une constante de normalisation... ☺).

```
In [3]:  
%matplotlib inline  
import matplotlib.pyplot as plt  
  
mu, sigma = 100, 15  
x = mu + sigma*np.random.randn(10000)  
  
plt.hist(x, 99)  
plt.grid(True)  
plt.show()
```

Résultats

Document final

Un document computationnel

Mon ordinateur m'indique que π vaut approximativement

3.141592653589793

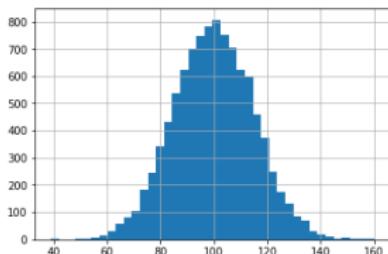
Mais calculé avec la **méthode des aiguilles de Buffon**, on obtiendrait comme approximation :

```
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 π (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 a uniform distribution of points (x, theta) and calculates the ratio of points where x <= mu + sigma * sin(theta) to the total number of points, which approximates pi/4.
- In [3]:** Plots a histogram of x values from 40 to 160, showing a bell-shaped curve centered around 100.

Document final

Un document computationnel

Mon ordinateur m'indique que π vaut approximativement

3.141592653589793

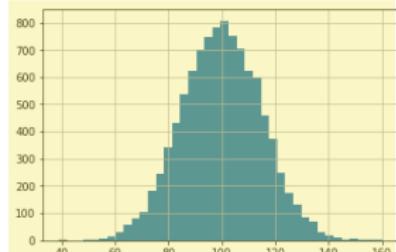
Mais calculé avec la **méthode des aiguilles de Buffon**, on obtiendrait comme approximation :

```
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 π (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 π (3.141592653589793) and includes a note about calculating it with the Buffon's needle method.
- In [2]:** Generates random points and calculates the ratio of points below a line to the total number of points, which is used to approximate π .
- 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 π vaut approximativement

3.141592653589793

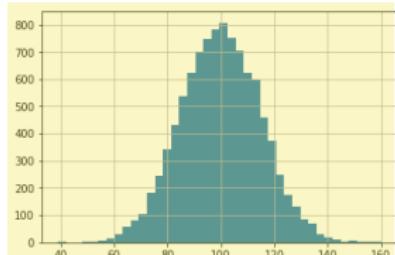
Mais calculé avec la méthode des [aiguilles de Buffon](#), on obtiendrait comme approximation :

```
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)
```

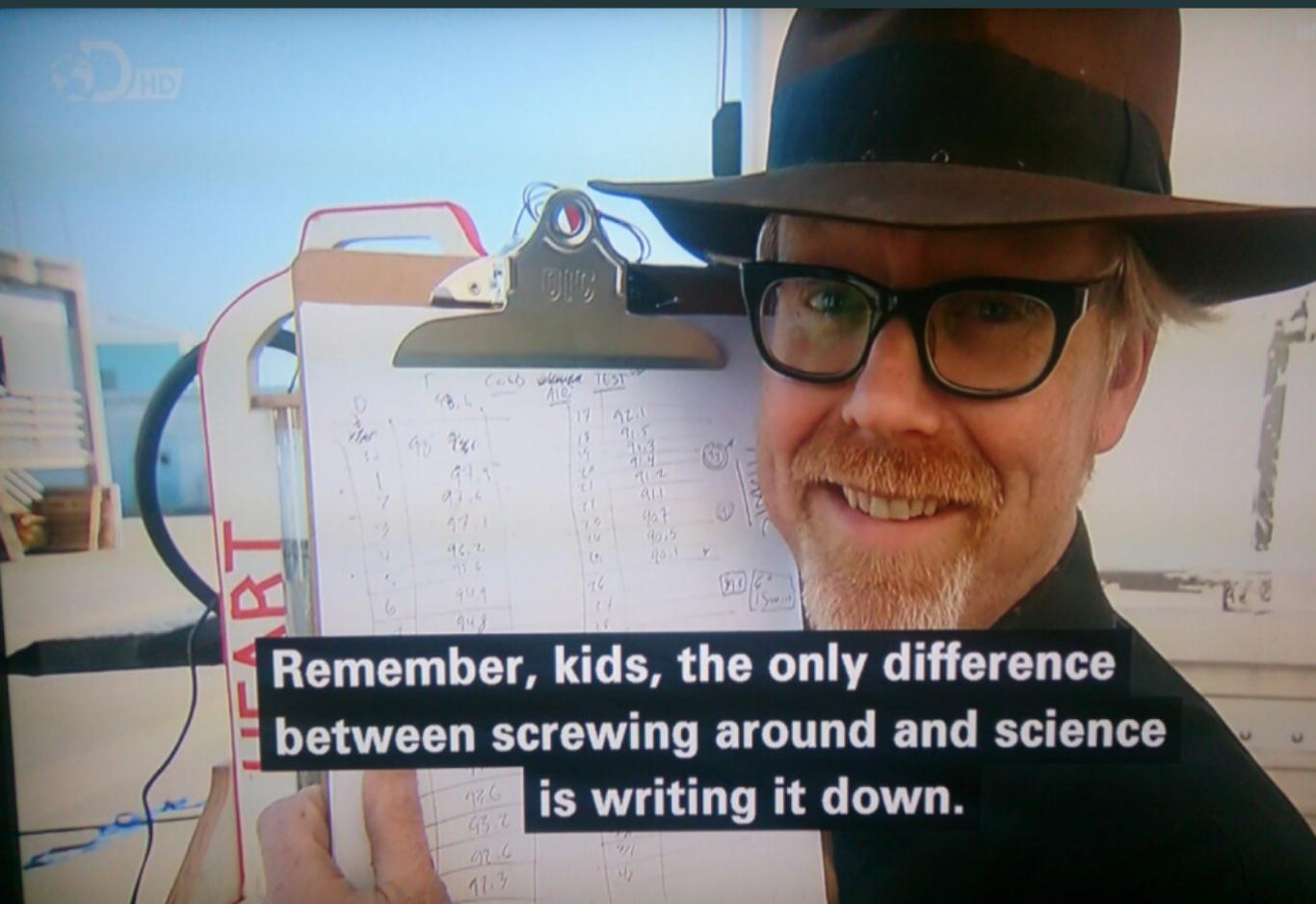
3.1437198694098765

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 π (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

The screenshot shows a Jupyter Notebook interface with the title bar "jupyter example_pi.ipynb". The notebook contains several cells:

- In [1]:** A code cell containing:

```
# Un document computationnel

# Mon ordinateur n'indique que j'suis vers *approximativement*.

In [1]:
```

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

Annotations above this cell say "Hide Prompt" and "Hide Code". Annotations below the output say "Mais calculé avec la __méthode__ des (ajoutées de Buffet) `math.pi_as_double()` ou `math.pi_as_hex()`, on obtientrait comme approximation...".
- In [2]:** A code cell containing:

```
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)
if (x**2 + np.sin(theta)**2) < 1/n
```

Annotations above this cell say "Hide Prompt" and "Hide Code". Annotations below the output say "On peut inclure des formules mathématiques comme `Sqrt(2)/pi` ou `(4*pi/3)*sqrt(3)` dans les cellules de code et elles seront automatiquement dessinées ou n'est rien à voir avec latex (si ce n'est une constante de normalisation...)."
- In [3]:** A code cell containing:

```
In [3]:
```

```
%matplotlib inline
import matplotlib.pyplot as plt

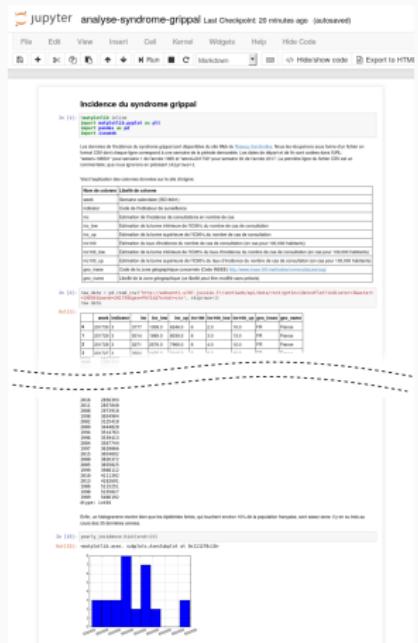
n, sigma = 100, 33
x = np.random.normal(0, sigma, n)
plt.hist(x, 40)
plt.title("Bell curve")
plt.show()
```

Annotations above this cell say "Hide Prompt" and "Hide Code". Annotations below the output say "Hide Output".

The output of this cell is a histogram titled "Bell curve" showing a normal distribution curve centered at zero with a standard deviation of approximately 33.

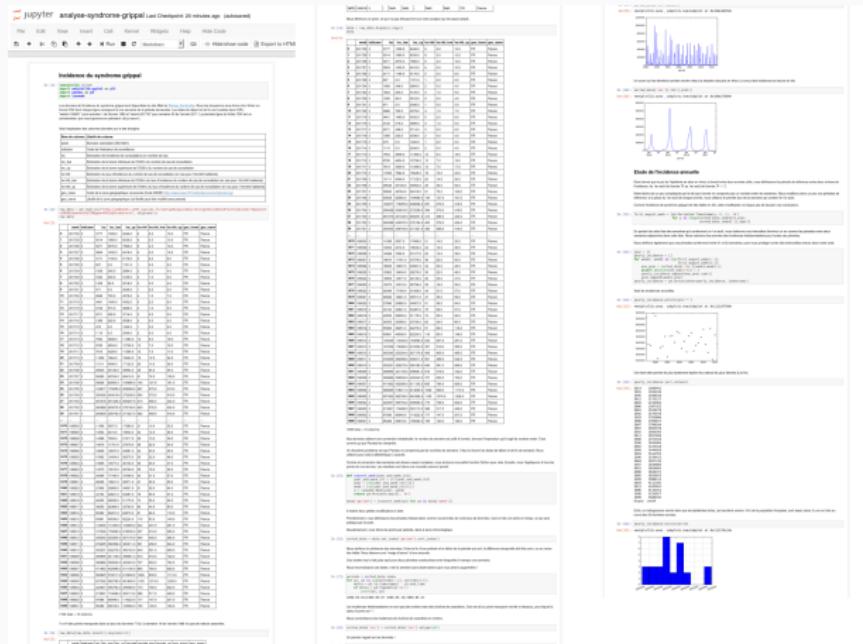
TOOL 1 TER: WORKFLOWS

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



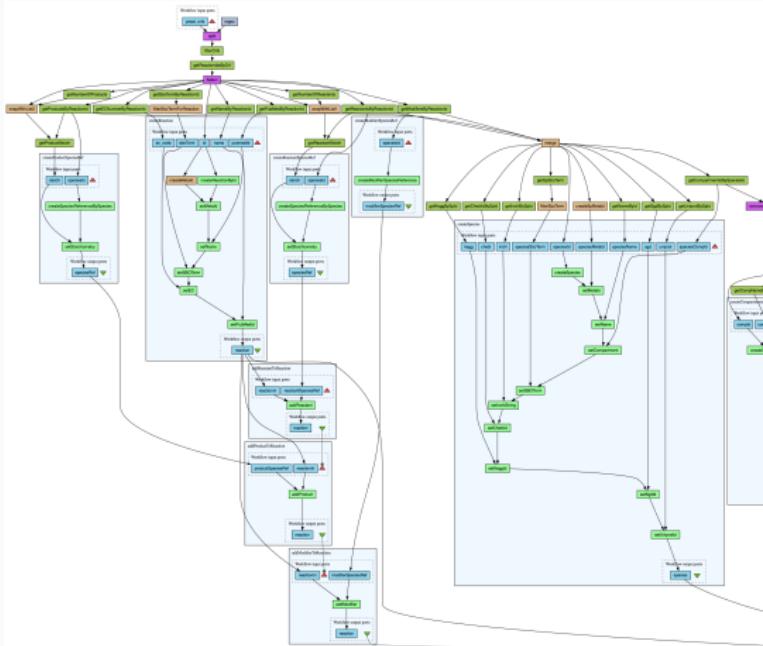
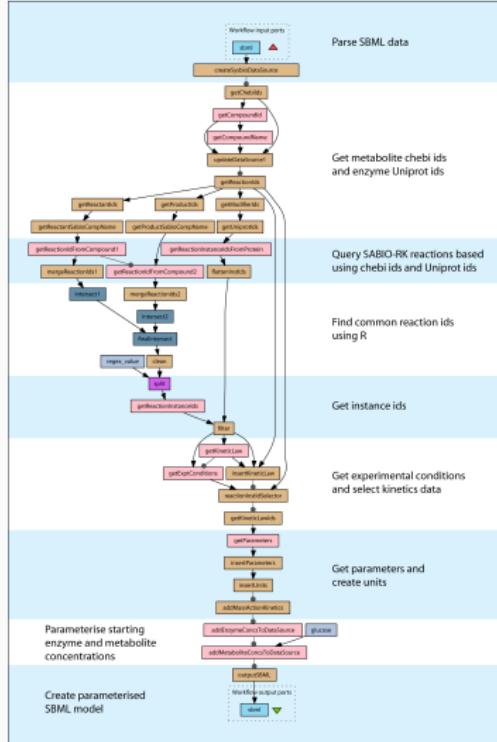
TOOL 1 TER: WORKFLOWS

Notebooks are no panacea and do not help developing clean code

The image displays a 4x3 grid of Jupyter Notebook screenshots, each showing a different step in a data science workflow:

- Row 1:**
 - Extracting Color Names by Web Image Searchers**: A cell with code for extracting color names from web images.
 - Dimensionality reduction and PCA**: A cell with code for dimensionality reduction and PCA.
 - Dimensionality reduction and PCA model results**: A heatmap titled "Dimensionality reduction and PCA model results" showing clustered data points.
- Row 2:**
 - Model building**: A cell with code for building a model.
 - Model building**: Another cell with code for building a model, showing a more complex script.
 - Analysis**: A cell with code for analysis, including a scatter plot of "Prediction error vs. Training sample variance".
- Row 3:**
 - Building the training data**: A cell with code for building training data.
 - Building the training data**: Another cell with code for building training data, showing a scatter plot of "Chronicity distribution of training data".
 - Analysis**: A cell with code for analysis, including a scatter plot of "Prediction error vs. Training sample variance".
- Row 4:**
 - Preparing**: A cell with code for preparing data.
 - Preparing**: Another cell with code for preparing data.
 - Conclusion**: A cell with code for conclusion.

TOOL 1 TER: WORKFLOWS



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

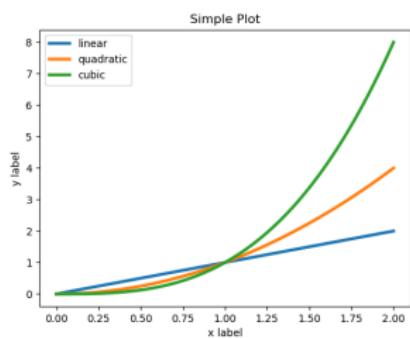
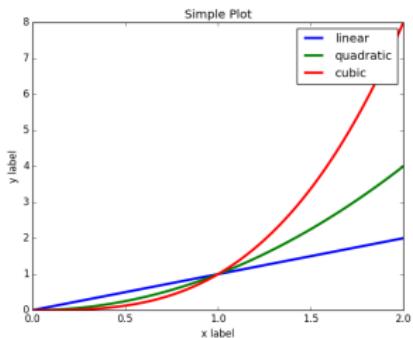
ARGH... DAMNED COMPUTERS

- 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

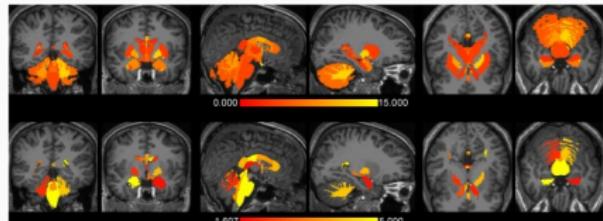
SOFTWARE DEPENDENCIES: HORROR STORIES

- 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%)

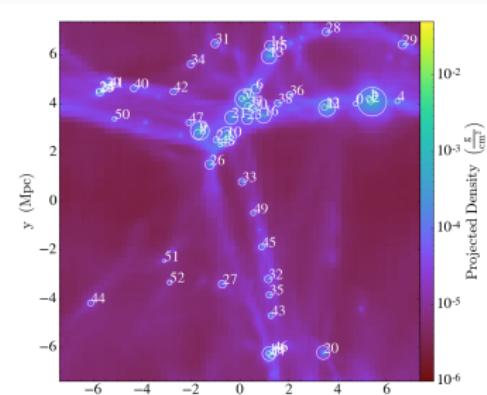
About a factor two smaller differences were found between the Mac and HP workstations and between Mac OSX 10.5 and OSX 10.6.

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

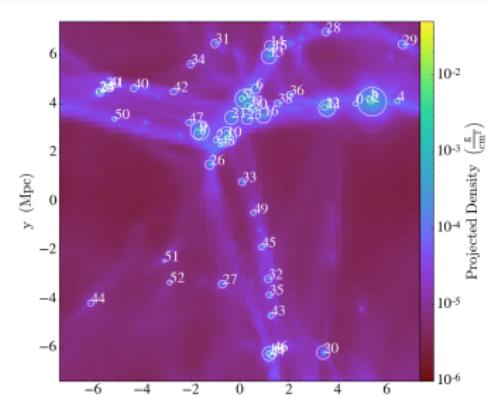


Assessing Reproducibility: An Astrophysical Example of Computational Uncertainty in the HPC Context (ResCuE-HPC, 2018)

| Compiler | Optim. | Largest Halo 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

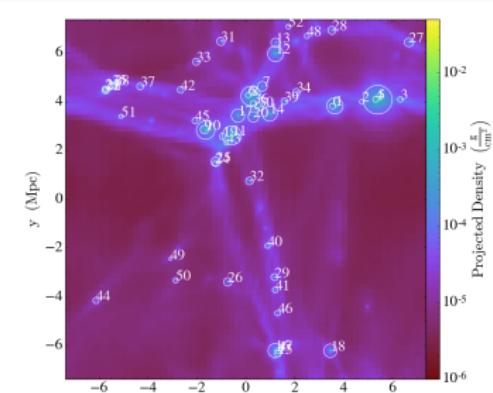


Assessing Reproducibility: An Astrophysical Example of Computational Uncertainty in the HPC Context (ResCuE-HPC, 2018)

| Compiler | Optim. | Largest Halo 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

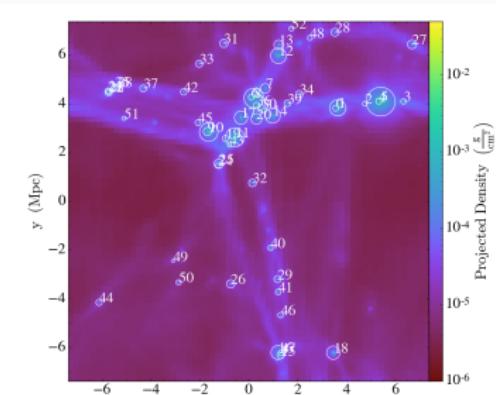


Assessing Reproducibility: An Astrophysical Example of Computational Uncertainty in the HPC Context (ResCuE-HPC, 2018)

| 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 | 22.71 E 46 | 1.587E 44 | 39h |
| intel@16.0.3 | Normal | 43.30 E 46 | 1.248E 44 | 7h |
| intel@16.0.3 | High | 2.268E 46 | 1.414E 44 | 6h |
| cce@8.5.5 | Low | 43.11 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
```

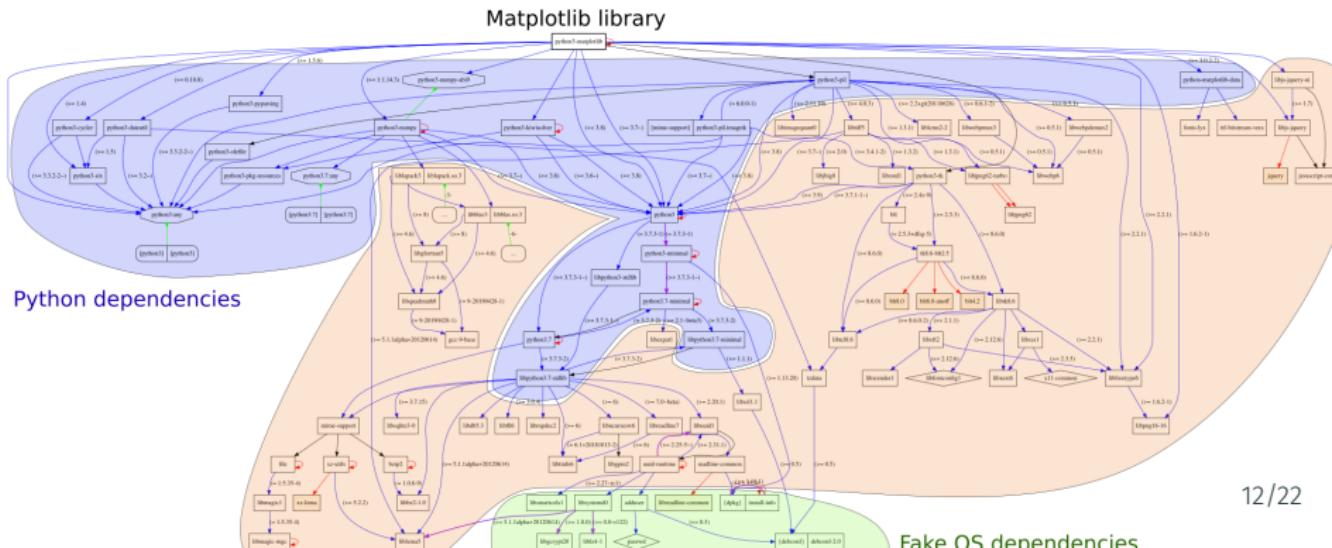
Package: python3-matplotlib
Version: 3.5.1-2+b1
Source: matplotlib (3.5.1-2)
Maintainer: Sandro Tosi <morph@debian.org>
Installed-Size: 27.6 MB
Depends: libjs-jquery, libjs-jquery-ui, python-matplotlib-data (>= 3.5.1),
 python3-dateutil, python3-pil.imagetk, python3-pyparsing (>= 1.5.6),
 python3-six (>= 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,

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 <cmd>`
 - Building images: `docker build -f <Dockerfile>`
 - Sharing through the Docker Hub: `docker pull/push `

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

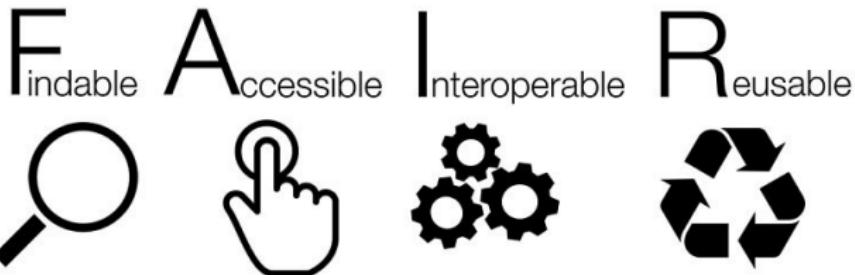
Package managers (the ugly and the good)

- 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/>

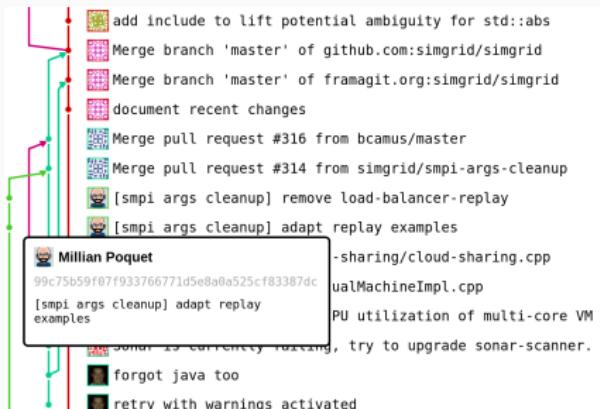
- "*Open as much as possible and close as much as necessary*"
- Management, publication, annotation (metadata), archiving
- Source code = specific data with specific consideration

Let's go beyond general principles!

TOOL 3: VERSION CONTROL AND FORGE

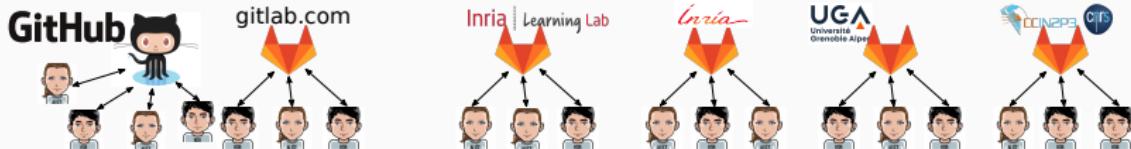
Git = version control

- 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](#), ...

CHANGING PUBLISHING PRACTICES

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



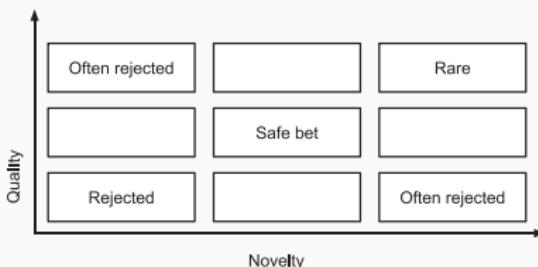
Joelle Pineau @ NeurIPS'18

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

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

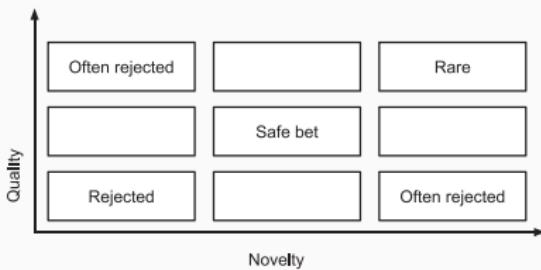
CHANGING ACADEMIC PRACTICES (PUBLISH OR PERISH)

- 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

REPRODUCIBLE RESEARCH = RIGOR AND TRANSPARENCY

Good research requires time and resources

1. Train yourself and your students: RR, statistics, experiments
 - Beware of checklists and norms Understand what's at stake

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

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



REPRODUCIBLE RESEARCH = RIGOR AND TRANSPARENCY

Good research requires time and resources

1. Train yourself and your students: RR, statistics, experiments

- Beware of checklists and norms Understand what's at stake

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

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



MOOC "Advanced RR" planned for Nov. 2023

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

REPRODUCIBLE RESEARCH = RIGOR AND TRANSPARENCY

Good research requires time and resources

1. Train yourself and your students: RR, statistics, experiments
 - Beware of checklists and norms Understand what's at stake

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

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



MOOC "Advanced RR" planned for Nov. 2023

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

2. Change the norm: make publication practices evolve
3. Incentive: consider RR/open science when hiring/promoting

REPRODUCIBLE RESEARCH = RIGOR AND TRANSPARENCY

Good research requires time and resources

1. Train yourself and your students: RR, statistics, experiments

- Beware of checklists and norms Understand what's at stake

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

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



MOOC "Advanced RR" planned for Nov. 2023

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

2. Change the norm: make publication practices evolve

3. Incentive: consider RR/open science when hiring/promoting

4. Prepare the Future: Toward **literate experimentation?**

- Reuse, reuse, reuse!
- Shared and controlled testbeds
- How to share Experiments ?



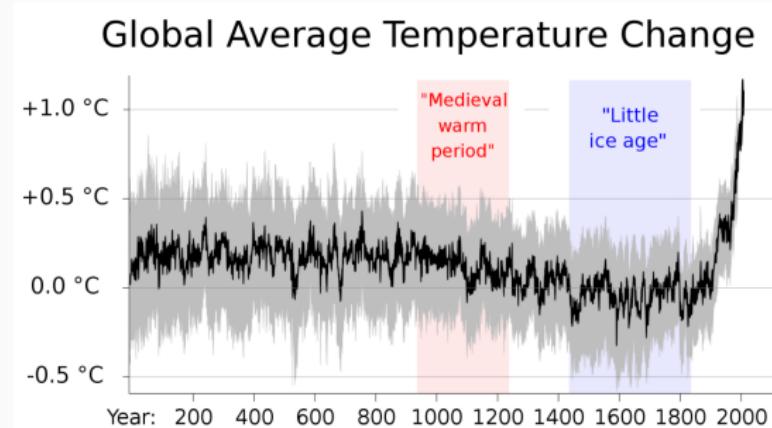
THE SCIENCE IS CLEAR

Why are we
ignoring it?

scientist rebellion

IPCC, IPBES, <https://climate.nasa.gov/>

1. Global climate change is not a future problem



https://en.wikipedia.org/wiki/Global_temperature_record

2023 Alberta wildfires (> 1 Mha)

THE SCIENCE IS CLEAR

scientist rebellion

Why are we ignoring it?

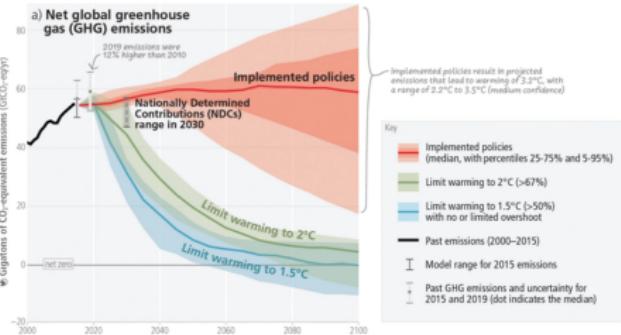


IPCC, IPBES, <https://climate.nasa.gov/>

1. Global climate change is **not** a future problem
2. It is **entirely** due to human activity

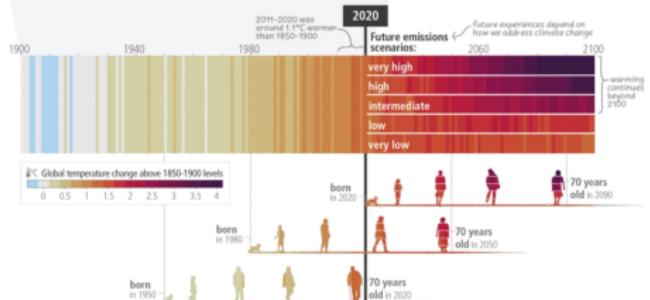
Limiting warming to **1.5°C** and **2°C** involves rapid, deep and in most cases immediate greenhouse gas emission reductions

Net zero: CO₂ and net zero GHG emissions can be achieved through strong reductions across all sectors



Paris Agreement'15 ~ Net Zero by 2050

c) The extent to which current and future generations will experience a hotter and different world depends on choices now and in the near-term



Latest IPCC report

21/22

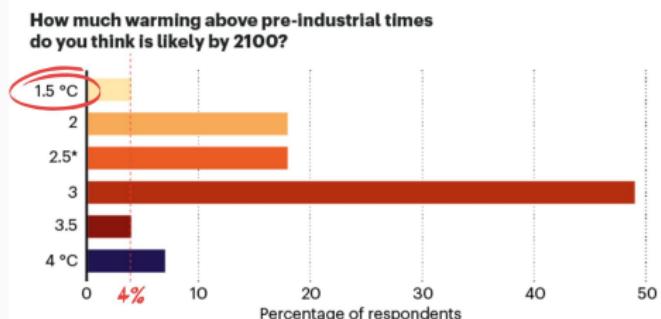
THE SCIENCE IS CLEAR

Why are we
ignoring it?

scientist rebellion

IPCC, IPBES, <https://climate.nasa.gov/>

1. Global climate change is **not** a future problem
2. It is **entirely** due to human activity
3. **9 out of 10 IPCC scientists believe overshoot is likely**

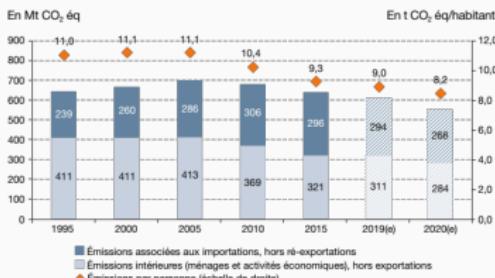


@natu Nature survey, Nov. 2021

THE ELEPHANT IN THE ROOM: CLIMATE CHANGE

Put aside biodiversity loss, pollution, freshwater, land system change...

ÉVOLUTION DE L'EMPREINTE CARBONE DE LA FRANCE



Empreinte carbone moyenne en France
10 tonnes de CO₂e/an/pers.



÷2
d'ici
2030

<2t CO₂e

Objectif d'ici 2050

- de 2 t de CO₂e/an/pers.

+ Faire plus d'activités bas carbone !

Danser, chanter, jardiner, rêver, écire, lire, courir, randonner, planter des arbres, discuter, marcher en forêt, méditer, passer du temps avec ceux qu'on aime, lire...

Bref, inventer nos vies bas carbone désirables !

Par exemple :

0,5 t CO₂e/Annee : à la maison, préférence légumes et fruits et sans produits dérivés

0,5 t CO₂e/Annee : transports : 2000km en voiture (80%) ou 30 ans, risques importants pour la santé et le climat, faire des déplacements en transports en commun, vélo, bus, métro

0,5 t CO₂e/Annee : Consommation : éviter rien de tout, mais faire des achats responsables et éthiques, faire des expérimentations et innovations, se servir dans les magasins de vêtements et accessoires

0,2 t CO₂e/Annee : L'agriculture : choisir bio sur un territoire en permaculture, "l'offrir" et d'agir pour le bien-être des animaux, faire des achats locaux et saisonniers, faire des achats de saison, préparer à la maison, préférer la viande rouge et le poisson

0,2 t CO₂e/Annee : Services publics : faire enseignement, éducation, culture, sport, etc.

<https://www.nosviesbascarbonne.org/>

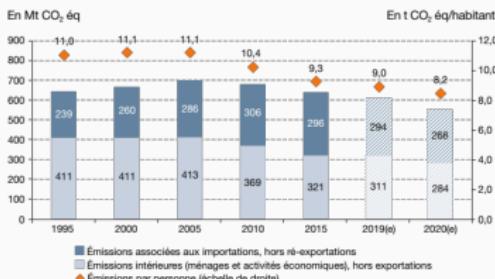
INVENTONS
NOUS VIES
BAS CARBONE

Sources : Kit Inventons nos vies bas carbone (Fév. 2021), Rapport sur l'état de l'environnement en France (Déc. 2020)

THE ELEPHANT IN THE ROOM: CLIMATE CHANGE

Put aside biodiversity loss, pollution, freshwater, land system change...

ÉVOLUTION DE L'EMPREINTE CARBONE DE LA FRANCE



(e) = estimations.

Note : l'empreinte carbone porte sur les trois principaux gaz à effet de serre (CO₂, CH₄, N₂O). En 2021, la méthodologie a été ajustée afin de mieux tenir compte de l'évolution des coûts du pétrole brut, du gaz et du charbon. La dernière édition de la série a ainsi été révisée, l'essentiel des ajustements portant sur les émissions importées de CH₄.

Champ : périmètre Kyoto (Île-de-France et outre-mer appartenant à l'UE).

Sources : Citepa ; AIE ; FAO ; Douanes ; Eurostat ; Insee. Traitement : SDES, 2021



Empreinte carbone moyenne en France
10 tonnes de CO₂e/an/pers.



÷2
d'ici
2030

<2t CO₂e

Objectif d'ici 2050

- de 2 t de CO₂e/an/pers.

+ Faire plus d'activités bas carbone !

Danser, chanter, jardiner, rêver, écire, lire, courir, randonner, planter des arbres, discuter, marcher en forêt, méditer, passer du temps avec ceux qu'on aime, lire...

Bref, inventer nos vies bas carbone désirables !

Par exemple :

0,5 t CO₂e : Alimentation : à tendance végétalienne, hypergourmande en sans produits animaux

0,5 t CO₂e : Transport : 2000km en voiture (800km de fabrication amortie sur 30 ans, importations de carburants, émissions de CO₂ dans les transports en commun)

0,5 t CO₂e : Consommation : Basé sur des ménages moyens, faire diverses expéiences éco-responsables et intemporelles, se déplacer dans le cadre de ses obligations quotidiennes

0,2 t CO₂e : Logement : Chaque fois qu'un m² HPC est construit, 90% de l'effort d'un logement basé isolé dans un quartier périurbain va à la construction d'un bâtiment ou solaire thermique

0,2 t CO₂e : Services publics : faire évoluer, améliorer, optimiser les services publics



<https://www.nosviesbascarbonne.org/>

INVENTONS
NOUS VIES
BAS CARBONE

Sources : Kit Inventons nos vies bas carbone (Fév. 2021), Rapport sur l'état de l'environnement en France (Déc. 2020)

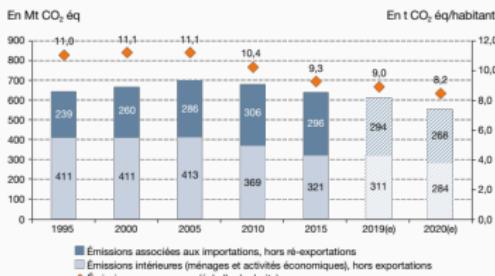
French government response

- Verdissement de l'industrie: « pause » sur les normes environnementales
- Loi de programmation militaire (+41%)
- Nous devons préparer la France à une élévation de la température de 4 °C
- Academia ? PEPR 5G, Cloud, NUMPEX, Quantique, IA, Agroécologie et numérique

THE ELEPHANT IN THE ROOM: CLIMATE CHANGE

Put aside biodiversity loss, pollution, freshwater, land system change...

ÉVOLUTION DE L'EMPREINTE CARBONE DE LA FRANCE



Empreinte carbone moyenne en France
10 tonnes de CO₂e/an/pers.

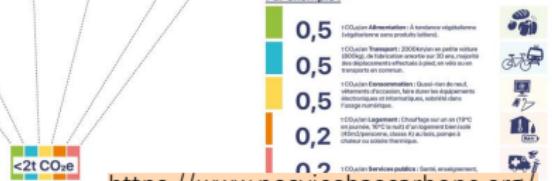


÷2
d'ici
2030

Objectif d'ici 2050
- de 2 t de CO₂e/an/pers.



Par exemple :



<https://www.nosviesbascarbonne.org/>

INVENTONS
NOUS VIES
BAS CARBONE

Sources : Kit Inventons nos vies bas carbone (Fév. 2021). Rapport sur l'état de l'environnement en France (Déc. 2020)



French government response

- Verdissement de l'industrie: « pause » sur les normes environnementales
- Loi de programmation militaire (+41%)
- Nous devons préparer la France à une élévation de la température de 4 °C
- Academia ? PEPR 5G, Cloud, NUMPEX, Quantique, IA, Agroécologie et numérique

Several scenarios on the table

- What will research/CS look like/be used for in such a world?
- Energy optimization/saving ≠ sobriety and frugality