

# Logiciel libre et science ouverte dans la cité

---

Gaël Varoquaux

*Inria*



# Logiciel libre et science ouverte dans la cité

- 1 Science et société
- 2 Quand l'informatique ouvre la science
- 3 Python scientifique, la naissance d'un écosystème
- 4 Scikit-learn : démocratiser l'apprentissage statistique



# 1 Science et société



# La science change la société

## Mécanique Newtonienne

- Hier dynamique céleste
- Aujourd'hui des ponts...

## Mécanique quantique

- Énergie et bombe nucléaire

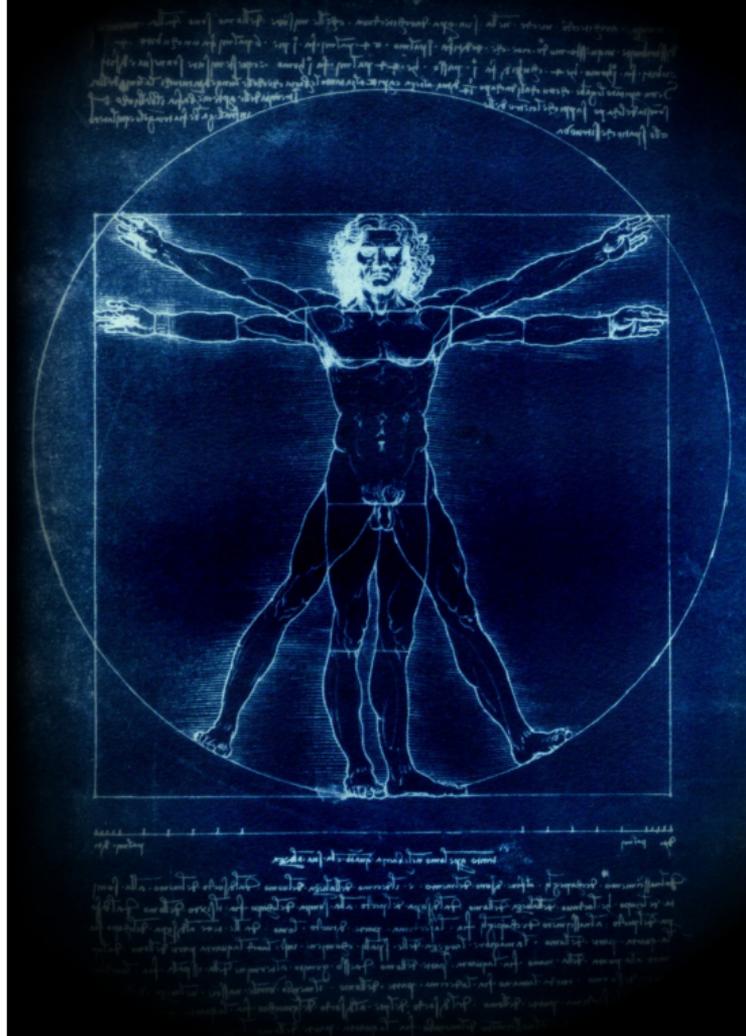
## Psychologie cognitive

- Améliorer l'enseignement

## L'histoire

- Hier éclaire aujourd'hui

*Ceux qui ne peuvent se souvenir du passé sont condamnés à le répéter — Santayana*



# La science : une démarche

## Établissement de connaissance

- Une vérité souvent difficile à établir
- Théorie avec prédictions précises  
⇒ falsification possible
- ⚠ Domaine de validité  
avion en papier  $\neq$  parabole
- Difficulté de la mesure  
(test d'intelligence)

*La science, c'est dur*



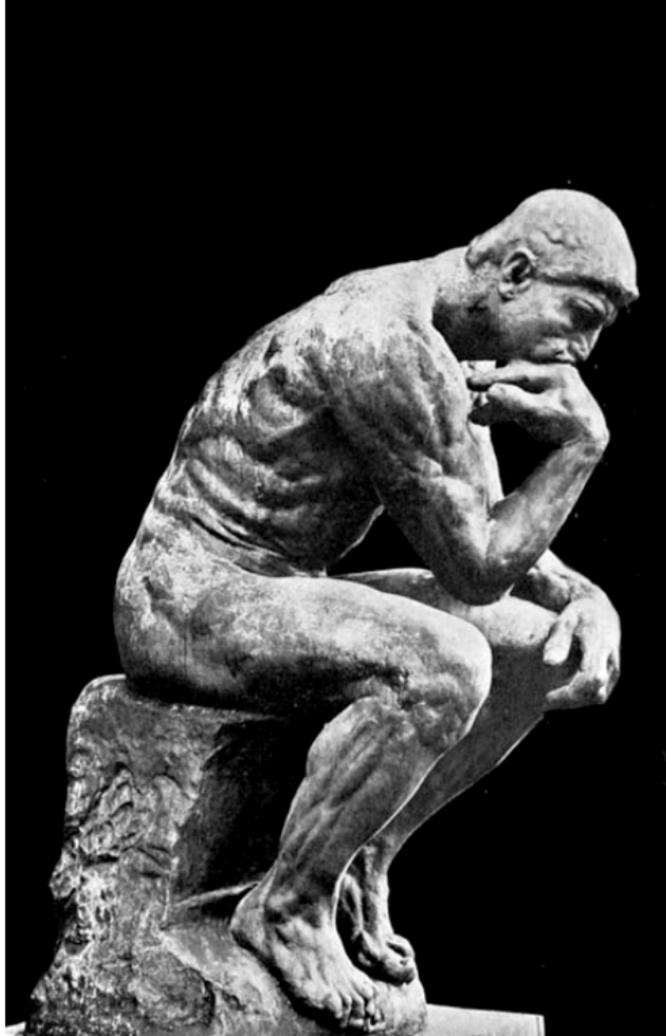
# Délégation épistémique

*L'argument d'autorité*

On ne peut tout savoir

- L'ordinateur fait ce qu'on croit
- Nos cellules ont bien de l'ADN
- La preuve dans [doi :10.2307/2118559](https://doi.org/10.2307/2118559)
- ...

*Notre démarche  
repose sur la confiance*



# Un équilibre fragile

## Des faillites

- Réchauffement climatique
- Vaccins et autisme
- Épidémiologie du dimanche

...



# Un équilibre fragile

## Des faillites

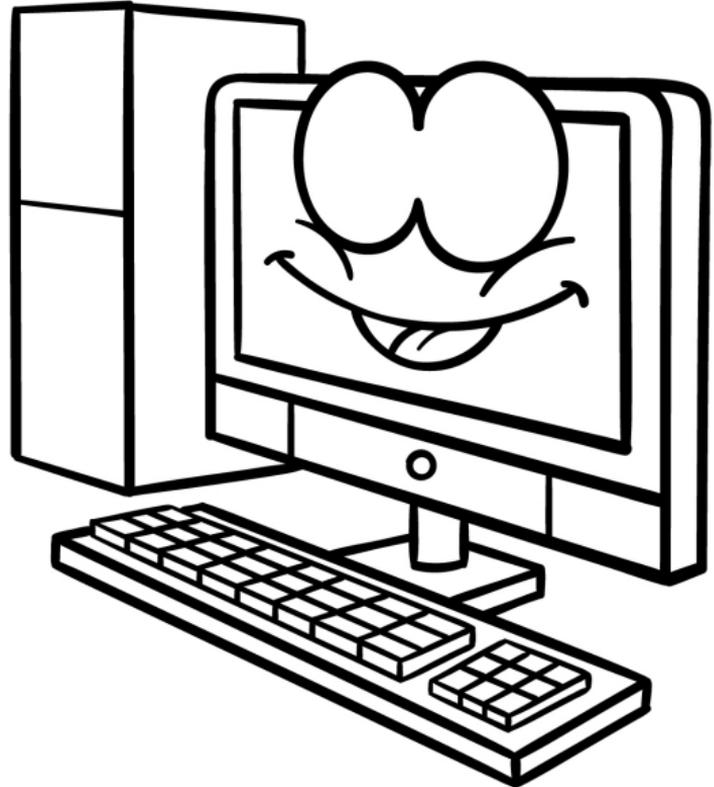
- Réchauffement climatique
- Vaccins et autisme
- Épidémiologie du dimanche
- ...

## Existence de la science dans la société

- Contribuer au discours publique,  
au progrès
- La société doit s'en emparer



## 2 Quand l'informatique ouvre la science



## L'informatique en science

- Modèles (prédictions)
- Données (transparence, partage)
- Expérience numérique (exploration)

Devenir numérique rend la science plus accessible

Reproduire facilement  
S'approprier



# Ouvrir

## Les codes sources

- Licence non contraignantes (BSD)
- Économie de mutualisation de coûts  
dépasser vision simpliste du “transfert” 🗑️

## Les donnés

- Transparence, confiance
- Données scientifiques  
Vers les pairs – post review
- Données d’individus et de société  
santé, transport en commun, ⚠️ vie privé  
Vers la société civile

Rapports de pouvoir



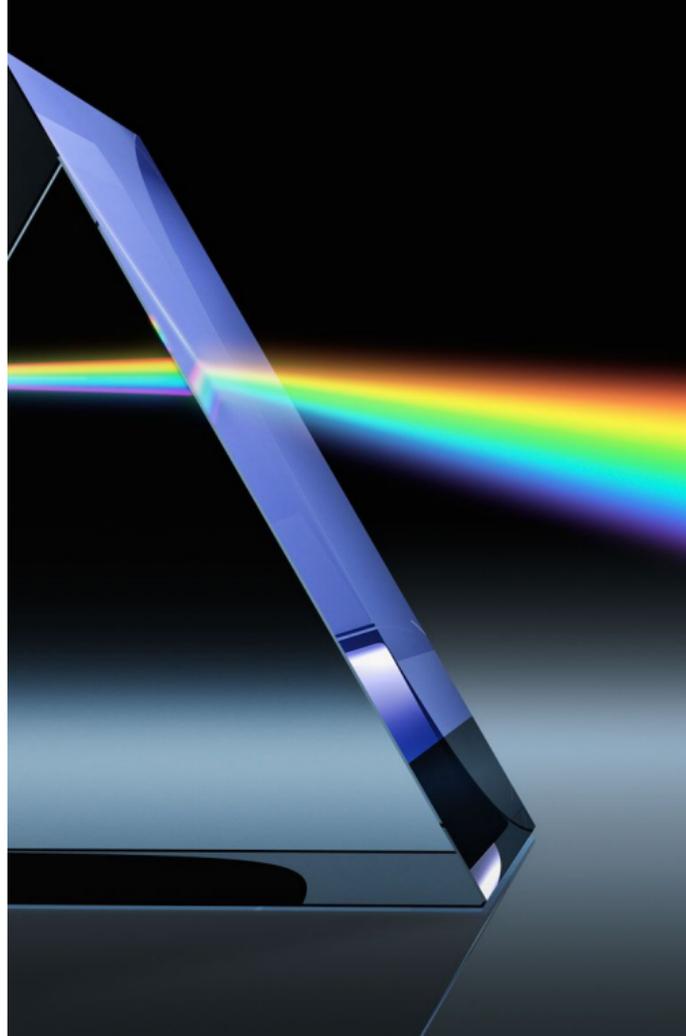
## Plus que la transparence

Dépasser le faux sens

Épidémie déclarée  $< 1$  semaine

⇒ Séjour moyen en réa  $< 1$  semaine  
("censure statistique")

- Pédagogie
- Education
- Appropriation

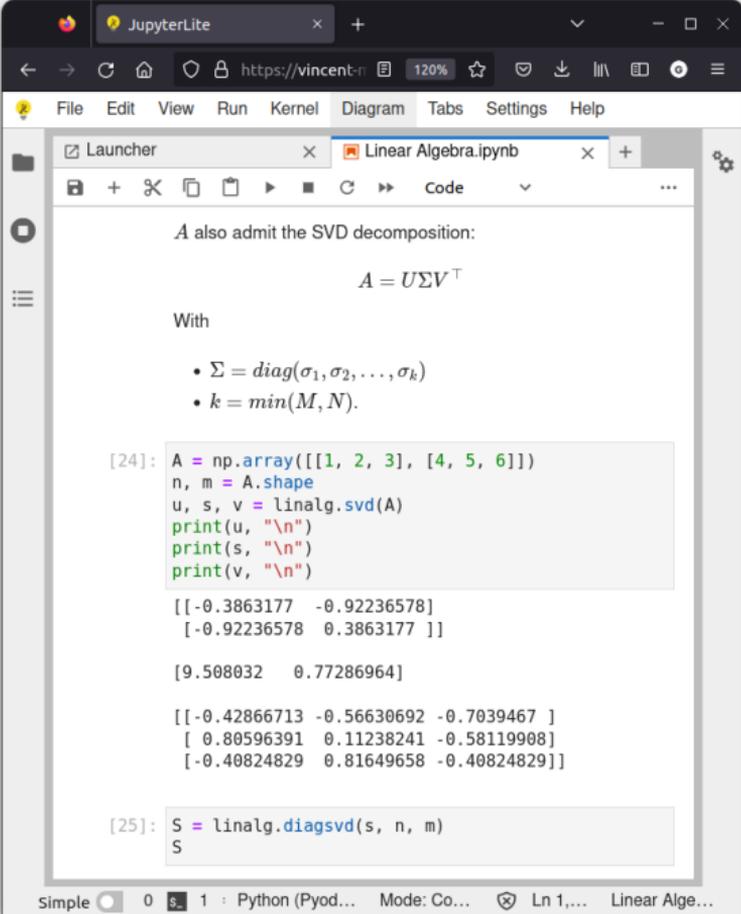


# La science dans le navigateur

## Cours d'algèbre linéaire computationnelle

<https://vincent-maladiere.github.io/scipy-demo/lab/index.html>

Éducation pour tous  
Informatique et sciences du  
numérique au lycée



The screenshot shows a JupyterLite browser window with a single tab titled "Linear Algebra.ipynb". The interface includes a menu bar (File, Edit, View, Run, Kernel, Diagram, Tabs, Settings, Help) and a toolbar with icons for file operations and execution. The notebook content consists of text, a mathematical equation, a list of parameters, and two code cells with their corresponding outputs.

A also admit the SVD decomposition:

$$A = U\Sigma V^T$$

With

- $\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_k)$
- $k = \min(M, N)$ .

```
[24]: A = np.array([[1, 2, 3], [4, 5, 6]])
n, m = A.shape
u, s, v = linalg.svd(A)
print(u, "\n")
print(s, "\n")
print(v, "\n")
```

```
[[ -0.3863177  -0.92236578]
 [ -0.92236578  0.3863177  ]]

[9.508032  0.77286964]

[[ -0.42866713  -0.56630692  -0.7039467 ]
 [ 0.80596391  0.11238241  -0.58119908]
 [-0.40824829  0.81649658  -0.40824829]]
```

```
[25]: S = linalg.diagsvd(s, n, m)
S
```

At the bottom of the window, the status bar shows "Simple" mode, "0" lines of code, "1" cell, "Python (Pyod...)" kernel, "Mode: Co..." mode, and "Ln 1,..." position. The tab title is "Linear Alge...".

A query on neuroscience, cognition, or brain pathologies

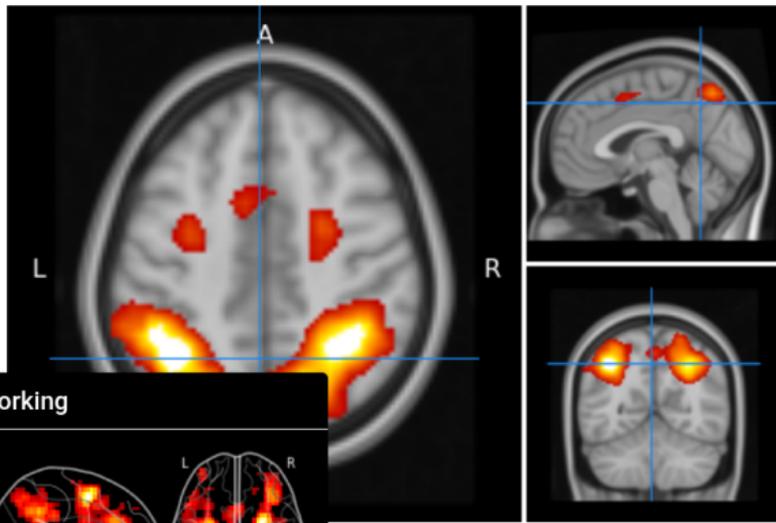
visuospatial attention

Click to edit.

Edit query 

Terms related to the query

	Term	Similarity <sup>†</sup>	Weight in brain map <sup>‡</sup>	N
In query	visuospatial	<div style="width: 80%;"></div>	<div style="width: 10%;"></div>	2001
	attention	<div style="width: 40%;"></div>	<div style="width: 10%;"></div>	9595
In expansion	parietal	<div style="width: 10%;"></div>	<div style="width: 90%;"></div>	10159
	working	<div style="width: 15%;"></div>	<div style="width: 85%;"></div>	5201
	left	<div style="width: 20%;"></div>	<div style="width: 80%;"></div>	12782
	task	<div style="width: 25%;"></div>	<div style="width: 75%;"></div>	12194
	working memory	<div style="width: 30%;"></div>	<div style="width: 70%;"></div>	4424
	visual	<div style="width: 35%;"></div>	<div style="width: 65%;"></div>	10448
	occipital	<div style="width: 40%;"></div>	<div style="width: 60%;"></div>	6796
	memory	<div style="width: 45%;"></div>	<div style="width: 55%;"></div>	8405
	right	<div style="width: 50%;"></div>	<div style="width: 50%;"></div>	13076
	hand	<div style="width: 55%;"></div>	<div style="width: 45%;"></div>	7603



working

Search on [Wikipedia](#)  
Search on [MeSH](#)



...ion of activations in the

Download map 

### 3 Python scientifique, la naissance d'un écosystème



## Python, what else ?

Un langage généraliste

- Techno web, admin système...

Utile pour la science

- Facile à apprendre

- Une compétence utile

Une technologie adapté

- Machine virtuelle peu invasive

- Incorporation codes existants

partage mémoire



# Un écosystème scientifique

numpy

calcul sur tableau et modèle de mémoire

scipy

routines numériques (numerical recipes)

matplotlib

visualization scientifique

jupyter (ipython)

environnement de travail

*Quelques décennies*



# Un développement décentralisé

## Une communauté

### ■ Beaucoup d'acteurs différents

- académiques (certains précaires)
- start-up (certains devenus riches)

## Une croissance organique

- Différents projets naissent
- Dialogue plutôt que hiérarchie



## 4 Scikit-learn : démocratiser l'apprentissage statistique



# Apprentissage statistique

## Apprendre des règles

- Programmer / modéliser différemment
- Ajustement de courbes ++
- Modèles flexibles plutôt que réalistes

## Utile partout

- Science fondamentale (liens statistiques)
- Intelligence artificielle
- Santé, société



## La vision derrière scikit-learn

Démocratiser l'apprentissage statistique

- Algorithmique compliquée
- Expertise statistique pointue

**La première barrière à l'impact  
est l'adoption**

Faciliter

- Baisser les coûts
- Développer des nouvelles applications

*Au sein d'un écosystème*



# Une grande communauté

## Énormément d'utilisateurs

- 1 million / mois consultent la doc
- 66 000 citations académiques

## Beaucoup de développeurs

- > milles contributeurs
- 50 contributeurs différents / mois
- Pleins de background différents
- Rôle d'ingénieurs "cœur"  
Efforts de développement humain

## Financements

- Projets académiques
- Mécénat (fondation Inria)



# Rendre facile est difficile

## Des choix techniques

- Modèles faciles d'utilisations
- Algorithmes stables
- Bon défauts

## Utilisabilité

- Un code utilisateur facile à lire
- Modèle "boite grise"

## Documentation

- Co-concevoir code et documentation
- MOOC : expliquer, à tout le monde



## Efforts en cours

### Performance

- Implémentations plus rapides
- Intégration de GPU

### Utilisabilité

- Intégration de données (pandas)
- Visualisation de modèles
- Validation de modèles



# Logiciel libre et science ouverte dans la cité

## La science s'inscrit dans la société

- L'influence se fait dans les deux sens
- S'ouvrir à la société
- L'informatique facilite l'aller retour
- Un effort pour dépasser son domaine

## Scikit-learn et l'écosystème Python

- Pour la science et au delà
- Beaucoup de compromis : pas la course à l'exploit

