The under-equipped social scientist ?

Why we need more dedicated, flexible and documented Python libraries for social sciences.

Open Research @FODEM 2023

Emilien Schultz - médialab/SESSTIM - France

TL;DR Social sciences need more scientific programming

1/ Scientific programming has the “right” **flexibility** to equip the diversity of practices in social sciences

2/ For the moment, the main language used is R ; Python could benefits of some **impulse**

3/ Dedicated disciplinary packages are a gate to develop Python uses, i.e. **“disciplinary API”**

Disclosures :

1/ I have been trained in physics before moving to sociology but **speak as a sociologist here**

2/ 2 question in 1 : train social scientists to Python/clarify the role of scientific programming

3/ Work and thinking in progress…

To be clear :

Not saying that we are under-equipped in a pejorative way

- Social sciences have well established open source software/platforms
- Give a warm welcome to new strategies for data analysis
- Expand its object to new numeric data

But (in general)

- Have low tech practices
- Use applications for discreet (punctual, unseen) operations
- Need flexibility to adapt to individualized practices/topics

Muller C., Clavert F., 2021, « De la poussière à la lumière bleue », Signata, 12

A word about the specificity of social sciences

- **Plurality** between disciplines
- And within each disciplines
  - a diversity of methodologies/theoretical approaches
  - Weak functional dependency between researchers
- Important national specificities
- Conceptually ladden
  - Importance of individual theoretical frame
- **Limits of one size fits all instruments**
  - Critics against the normativity of tools

Nevertheless, shared instruments are important

Science studies have shown the crucial role of instruments :

- conceptual changes
- disciplinary identity
- coordination between specialties and standardization of practices

(i.e. electronic microscope for biophysics ; high-throughput sequencer for medicine ; few studies about software though)

But : social sciences are divided regarding standardization, especially imposed standardization which can reflect power relationship

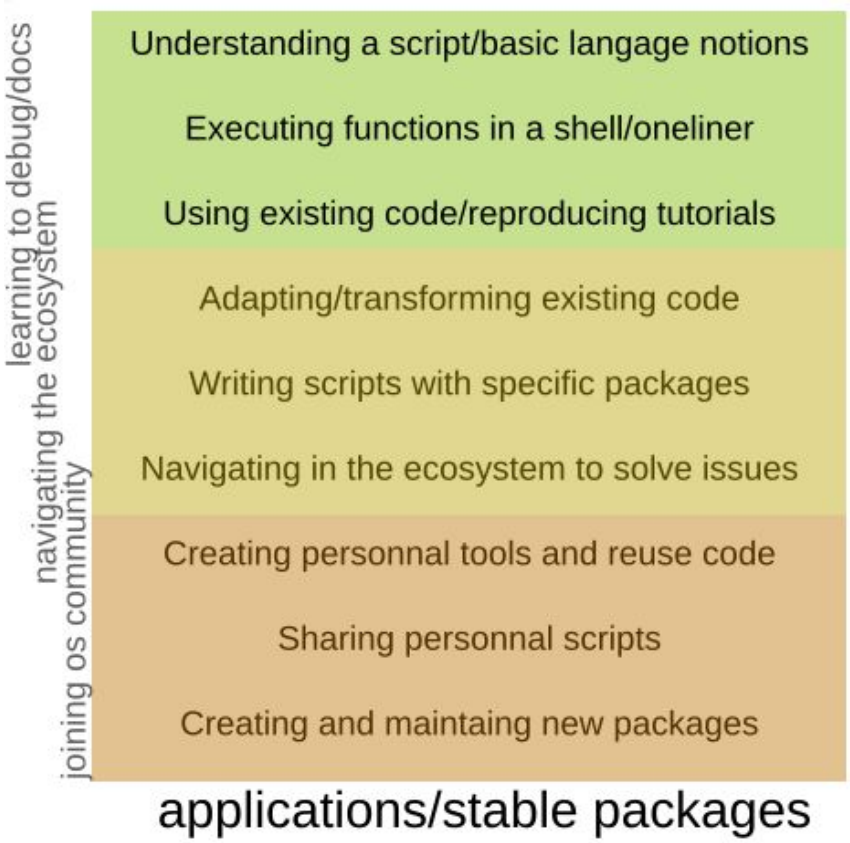
Jérôme Lamy, « Le renouveau de l'histoire des instruments scientifiques », *Artefact*, 2022.
URL : http://journals.openedition.org/artefact/13018 ; DOI :

Scientific programming is a solution

- Scientific programming favors new scientific instruments
  - from the specialties and improve a second-order standardization
- Scientific programming is an entry point for new open source practices
  - Good practices of reproducibility and collaboration
  - New cool stuff from computer sciences (machine learning, etc.)
- **Nevertheless**, still not very common in social sciences
  - For the common people (there are always cool kids)
  - R has a intermediary status : between programming and statistics

Scientific programming

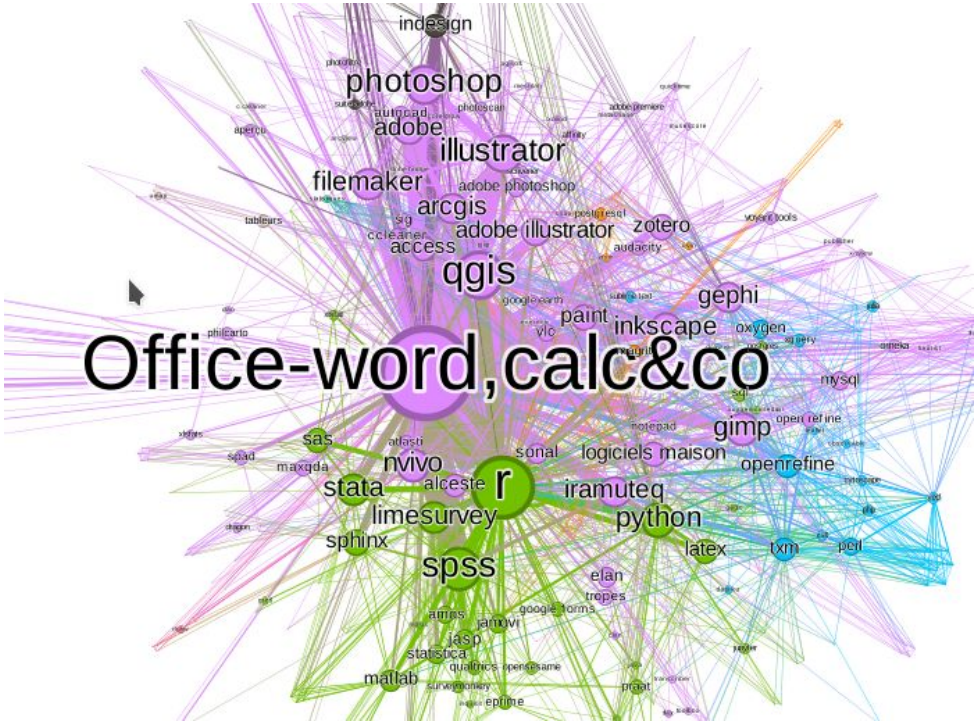
- In reality, a gradient of practices
  - Have in common : Interactivity, exploratory, based on packages
- Priority given to the usefulness regarding uncertain questions to explore
  - Stability, design and other important questions of software development are secondary



The instruments of French social scientists

N = 492 ; SOSP_ State of Open Science Practices in France (Le Béchec et al., 2021)

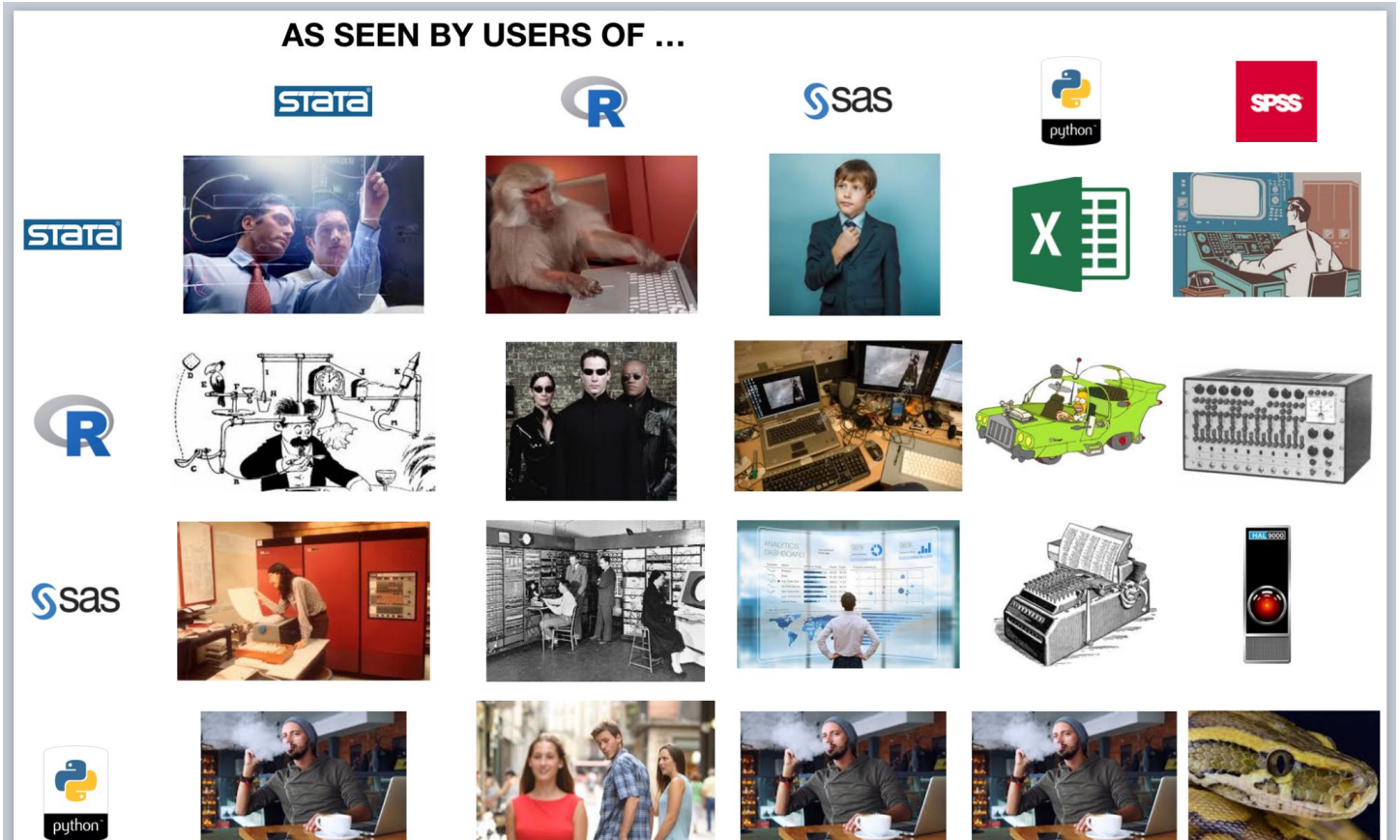
A diversity of softwares ; centrality of standard office software then R ; diversity of profiles



23% use R ; 12% QGIS ; 10% SPSS ; 6% Python

“I generate my data using my body (brain, mouth, and hand) when I'm in the field, and I use some tools (recorder, camera, and notebook/pen) to record these data digitally, I mainly use Pages and Numbers on Mac. I use Adobe for PDFs, when I want to save/protect a photo, I copy it to TIFF. I sometimes make hand-drawn diagrams (not digital). I have used a spreadsheet (Numbers/Excel) to categorize data, I don't really generate figures. I have used GIMP to edit a map.” (comment from SOSP survey)

A fragmented landscape even for “quanti” users



Observation : R developed for good reasons

An **elective affinity** between the diversity of practices and the flexibility of the tool allowed by scientific programming

- script orientated
- small packages specific to subcommunities
- support in French (FactoMineR)

Some limits :

- diversity of specific tools
  - for instance, function in a library specific to a national community or only documented in another language than English, that doesn't exist elsewhere or is not documented in English or other languages
- low documentation / standardization of code
- ambiguity between "statistical language" and "programming language"

The state of Python uses in social sciences

- Let’s say we are not many
- More and more young researchers interested to leverage machine learning in their research
- Difficult to get “basic” analysis with the current stack
    - i.e, the presentation of logistic regression

Dedicated community packages would be a middle ground for researchers to access scientific programming and then are made aware of better OS practices

Need to go **beyond** application development (one package will not be enough)

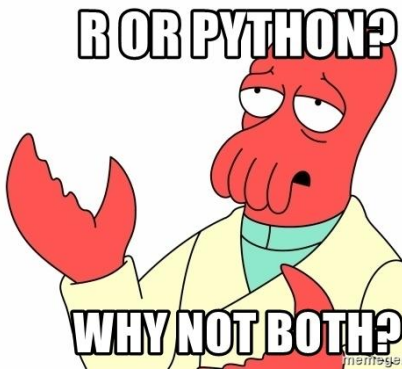
Expected positive benefits of Python's broader adoption

- Enhance scientific programming practices
  - especially thanks to the Notebooks & other tools of the ecosystem
- The diverse tools of the scipy community have the potential of flexibility
- A lingua franca with other communities (computer science, etc.)

But R is already here (what to do with that …)

- Accept that it is a foolish idea to develop Python
- Advocate for polyglotism or shared libraries between R and Python (Apache Arrow)
- Start a transition to Python as the main first language learnt for social scientists

My leap of faith : still advocate Python



How to enable this practice - my PySHS

A double constraint :

- achieve some standardization
- without sacrificing disciplinary specificities

Step 1 : identify quasi-standard practices

Step 2 : build easy to use packages that can find its place in the workflow

Step 3 : prove it can be useful

Step 4 : train colleagues and develop practices

Step 1 : uncovering standard practices

- Identifying the common sense of the specialties
- Not all social scientists are doing machine learning (or statistics)
  - but a lot of them are doing some basics statistics
- There are are some “quasi-standard” operations :
  - descriptive statistics for surveys
  - format transformation
  - generate tables / intermediate documents
  - “usual tools from handbooks” : MCA, etc.
- Need to start from these workflow

Step 2 : facilitating disciplinary use - my try

- in French
- one-liner
- close to the common sense
  - tables
  - statistical tests
- facilitate workflow for survey analysis
- based on well maintained packages

Step 3 : showing the usefulness of both Python/Pyshs

To diffuse the tools, there is a need to do **public demonstration in context**

-> Notebooks have become a perfect vector to combine specific research questions and standard tools.

Thanks to Huma-Num Labs / Datactivist collaboration in France, 5 notebooks for machine learning : https://gitlab.huma-num.fr/io as a starting point

README.md

Analyse de questionnaire : enquête sur les pratiques numériques des chercheurs

V1 complète 2022-07-22

Présentation du Notebook

Démarche d'analyse de données d'une enquête par questionnaire.

L'objectif est de présenter la démarche de chargement, de transformation et d'analyse des données avec pour objectif la production d'une typologie des répondants sur leurs pratiques numériques.

Les données utilisées sont celles collectées en 2020 par le collectif du projet **State of Open Science Practices in France** *Mariannig Le Béchec - Aline Bouchard - Philippe Charrier - Claire Denecker - Gabriel Gallezot - Stéphanie Rennes mise à disposition sur la plateforme Zenodo qui ont donné lieu à un rapport.*

Niveau du Notebook : débutant/intermédiaire

Principales notions vues :

Step 4 : training colleagues and students

The tools need to find their place in research workflow

Translation needed :

- Writing books and academic examples to stabilize shared practices
- Intervening in the laboratories to show the uses of Python/existing tools
- Trainings to give an overview of Python for Social sciences (URFIST Lyon ; CUSO)
- Creating space to discuss our specific practices (seminary ”Behind the scene of the code”)

Concluding ideas

- Scientific programming in Python = third way
  - Between applications/no-code
  - Promote reproducibility and open source practices
  - Promote interdisciplinary collaboration with colleagues
- A need for facilitators:
  - excavating standard practices
  - identifying early users and creating a core of developers
  - demonstrating the concrete efficiency
- Some limits :
  - focus on disciplines is a potential of dispersion from standard libraries
  - maybe there are better languages to promote…

Work in progress… Thanks

Thanks Matthias Bussonnier (Quansight) ; Léo Mignot (CED) ; Sébastien Plutniak (CNRS) and Mathieu Morey (datactivist) for the insights