# Statistical Programming and Open Science Methods

## Statistical programming languages: An overview

Joachim Gassen
Humboldt-Universität zu Berlin

September 02, 2022

SFB/Transregio 266
ACCOUNTING FOR TRANSPARENCY

# Time table October 10

| When? | What? |
|---|---|
| 09:00 | Welcome and Introduction |
| 09:30 | The development environment and project organization |
| 10:30 | Coffee |
| 11:00 | Using Git and Github |
| 12:30 | Lunch |
| 14:00 | Statistical programming languages: An overview |
| 15:30 | Coffee |
| 16:00 | Functional versus object-oriented programming |
| 19:30 | Pizza at Due Forni, Schönhauser Allee 12 |

# Obvious aspects to consider when picking the ‘right’ SPL

- User base
- Range of features
- Ease of learning
- Performance
- Extendability
- License model and pricing

# Not so obvious aspects

- Portability across platforms
- Portability across time
- Verifiability of algorithms
- Quality of documentation
- Interoperability with other languages
- Interoperability with RDBMS
- Dynamic output creation (HTML/Javascript)
- User community

# My take on the Top 4: Julia, Python, R and Stata

- Julia:
  - Pros: Fast, open source, expert user base
  - Cons: Small user base, few packages
- Python:
  - Pros: General purpose, open source, relatively easy to learn, many machine learning packages, large user base
  - Cons: Packaging system, statistic packages have limited interoperabilty, object orientation feels alien when working with data
- R:
  - Pros: Focused on data science, open source, packaging system, interoperability, graphics system
  - Cons: Not really easy to learn, tidyverse helps though
- Stata
  - Pros: Easy to learn, very broad user base in economics, most statisitical methods are quickly implemented
  - Cons: Commercially licensed and closed source, inflexible programming environment

# Activity: Compare our code solution

- Let's compare our solutions
- Do our samples differ? If yes: why?
- Whose code is the fastest?
- Whose code is the most readable?