

WORKSHOP ON OPEN WORKFLOWS (WOOW)

DANI ARRIBAS-BEL & THOMAS DE GRAAFF

OUTLINE

- Introduction: Why discuss this and why this workshop
- What is an open workflow (for social scientists)
- Tools for (open) workflows (Examples)
 - Editing
 - Writing
 - Analysing
 - Saving
- Conclusion

INTRODUCTION

WHY THIS WORKSHOP?

- Interest in workflow tools
- Increasing need for openness & transparency
 - from journals, universities and government
 - increase in cooperation (over wider distances)
 - access to your own files
- In the *social sciences* few attention to what tools to use (and why)

WHAT WE WANT (AND DON'T WANT) WITH THIS WORKSHOP

- We are not advocating particular tools
- instead, we are more interested in the principles behind a good open (scientific) workflow, aware of the facts that
 - there is no final, optimal, set of workflow tools
 - investment is very costly
- aiming for a broader discussion
- and stimulating a wider use

WORKFLOW

OPEN?

- Workflow: *Progression of steps (tasks, events, interactions) that comprise a work process, involve two or more persons, and create or add value to the organization's activities* ([BusinessDictionary](#))
- Open workflow: One that enhances *transparency, collaboration and reproducibility*

EMPIRICAL CYCLE

- Read other papers
- Think of a brilliant idea
- Do:
 1. Collect data
 2. Transform data
 3. Analyze data
 4. Write up results
 5. Present results
 6. Go back to 1. until satisfied
- Send paper to journal and go back once again to i. until referees satisfied
- And... documenting throughout the entire process!!!

THEORETICAL CYCLE

- Read other papers
- Think of a brilliant idea
- Do:
 1. State assumptions
 2. Model (simulate)
 3. Analyze model outcome
 4. Write up results
 5. Present results
 6. Go back to i. until satisfied
- Send paper to journal and go back once again to i. until referees satisfied
- And... documenting throughout the entire process!!!

WHY BOTHER ABOUT A WORKFLOW OR TOOLS?

- Good scientific practice: *document how you have achieved your results*
- Reproducibility
- Transparency
- Modularity
- Portability (across systems and users)
- Efficiency
- Self-sanity

WHY SHOULD IT BE OPEN?

- Open Science
- Reproducibility
- Transparency
- Modularity
- Portability (across systems and users)
- Efficiency

WHEN SHOULD I ADOPT AN OPEN WORKFLOW?

- The sooner the better
- But think twice about which one (switching is costly)
- Start one step at a time

A journey of a thousand miles begins with a single step

Lao-tzu

TOOLS FOR ...

OPEN WORKFLOWS

TEXT EDITTING

- Plain text is **simple**, light, cross-platform, flexible...
- Many **academic** tools are based on plain text (typesetting systems, scripting languages, bibliography managers)
- **Good investment** to learn a rich text editor ("learn once, use for everything"):
 - Efficient typing (command vs. insert modes)
 - Syntax highlighting and indenting
 - Shortcuts, macros and templates
 - Consistent look, feel and behaviour
- Examples: Vim, Emacs, other (TextMate, Sublime text, etc.,...)

TEXT EDITING

Vim demo...

- Command vs Insert mode
- Syntax highlighting
- LaTeX shortcuts
- Python indenting

BEAUTIFUL (AND EFFICIENT) TYPESETTING

- Documentation of progress, presentation of results (paper or slides) and final products depend on this
- **plain text + markup languages** = very powerful
 - Detach content inputting from layout and styling
 - One source, multiple outputs (paper, slides, website...)
- Examples: LaTeX, Markdown, Org

BEAUTIFUL (AND EFFICIENT) TYPESETTING

LaTeX and **Beamer**...

- General template
- Sectioning
- Equations (inline, outside)
- Table

Markdown...

MANAGING LISTS OF PAPERS

- *One reference list to rule them all*
- Create the reference and never worry about proper inserting
 - Bibtex
 - Reference manager
 - Online services (e.g. Mendeley)
- Bibtex demo...

ANALYZING DATA

- **Platforms** for statistical analysis & **scripting languages**
- Examples: Python, R, STATA
- The power of code vs. point-and-click
 - **Flexibility** (Python)
 - Typically **wider range** of methods (STATA)
 - **Extensible** and updated more rapidly (R)
 - **Reproducible** and transparent (remember *exactly* what you did)

ANALYZING DATA

IPython notebook demo...

- Load up data
- Create descriptives
- Scatter plot
- Run a model and simple print
- Print LaTeX output

SAVING THE WORKFLOW

Backup: "You don't need it until you really need it"

- Security copy of all your (valuable) documents
- External drive vs. Cloud solution
- Software to make the process painless or automated
- Many options: TimeMachine, Dropbox, Amazon Glacier...

SAVING THE WORKFLOW

Versioning control: *"How did I get to that table of results?"*

- Save snapshots of a project in an intelligent way
- Allows to trace the *history* of a project/document (very neat [example](#))
- Very well developed for code development
- Examples: DropBox, git, svn...

PUTTING IT ALL TOGETHER

Amsterdam paper example:

- http://darribas.org/buzz_adam



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).