soundsoftware.ac.uk

Learning how to learn, with Software Carpentry

Luis Figueira, Chris Cannam, Mark D Plumbley
Centre for Digital Music
Queen Mary, University of London

EPSRC
Engineering and Physical Sciences Research Council

Overview

- Introduction & Motivation
- What is Software Carpentry
- The bootcamps
- Some lessons learned
- Discussion

Reproducible Research

(Buckheit & Donoho, 1995; Vandewalle et al, 2009)

Idea: researchers should be able to reproduce the work of others.

So, we need

- The paper (ideally Open Access)
- The data (ideally Open Data)
- The software (ideally Open Source)

Well-known example: WaveLab (Buckheit & Donoho, 1995)

But in audio & music research, few people do this. Why?

Research software in practice

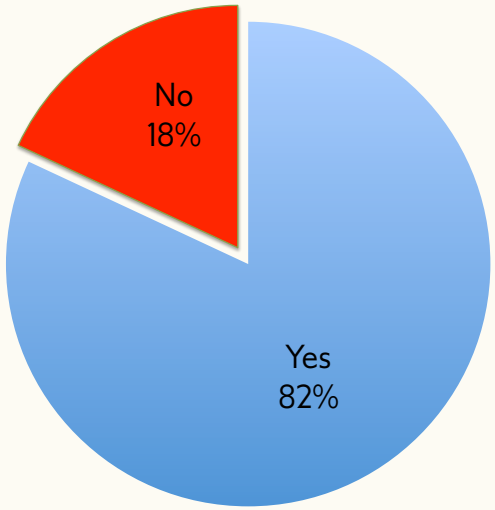
We carried out a **Survey of UK audio and music researchers***.

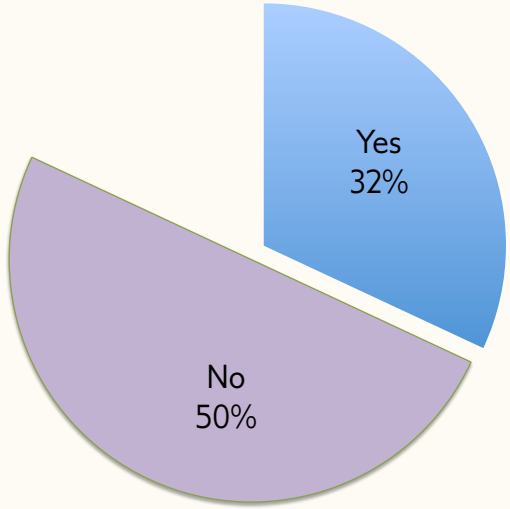
82% developed software, but only 39% of those took steps to reproducibility,

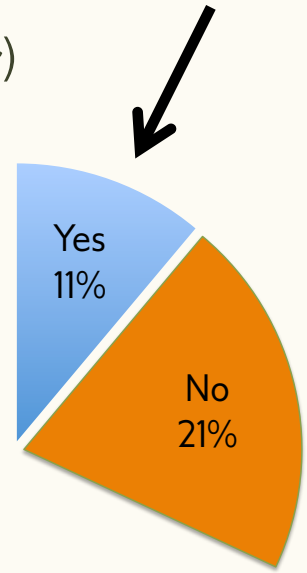
and only 35% of *those* published any code

only **11%** tried to be reproducible and published the code.

(Also: 51% said their code never left their own computer)







* - Oct 2010-Apr 2011, 54 complete + 23 partial responses. For these figures we considered 72 responses.

Why don't we publish code & data?

Our survey suggested:

- Lack of time
- Copyright restrictions
- Potential for future commercial use

Other factors (UK Research Information Network, 2010):

- Lack of evidence of benefits
- Culture of independence or competition
- Quality concerns (self-taught programmers)

Also: it takes effort **early** in the research cycle;
hard to find time/motivation after the paper is published

Barriers to publication and reuse

Barriers to publication and reuse:

- Lack of facilities and tools
- Lack of incentive for publication
- Platform incompatibilities
- **Lack of education and confidence with code**

These are barriers to publication of *code*.

(Related issues for data)

Software Carpentry

Volunteer organisation

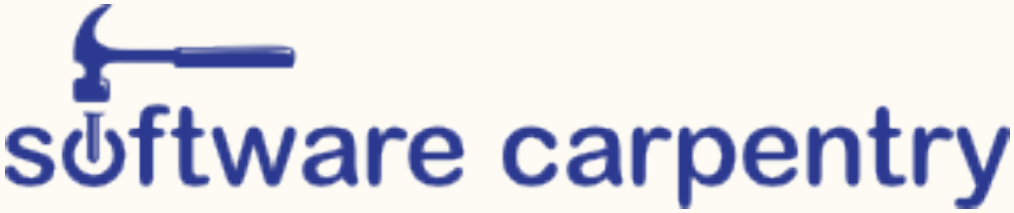
Started by Greg Wilson in 1998

Mission: "*Software Carpentry helps researchers be more productive by teaching them basic computing skills.*"

Funded by the Sloan Foundation and Mozilla

How?

- Online resources (video lectures, presentations)
- Workshops/Bootcamps



Our first workshop

Autumn School for Audio and Music Researchers

- London, one week (Nov 2010)
- Software Carpentry - Greg Wilson
- 10 - 5pm each day
- 22 researchers from 12 UK institutions
- Live coding, many group exercises
- Syllabus:
  - Python, Bash, Databases, Testing, Version Control
  - Last day: audio processing in Python

Our first workshop (cont.)

Post workshop survey (6 months after workshop):

- All liked the format (33% considered it excellent)
- Majority considered that it would be important for their future

Not sustainable:

- Extremely tiring for both instructor and participants
  - "*More wearing than enlightening*"
- Not always easy to take a week from your PhD
- Expensive (instructor, students, venue, ...)

A new approach

January 2012
- 2 days workshop
- more people can attend
- easier to retain what is learnt

Typical syllabus

Day 1

- Introduction to Unix shell (pipes, loops, scripting, ...)
- Introduction to Python (components/reusable functions)

Day 2

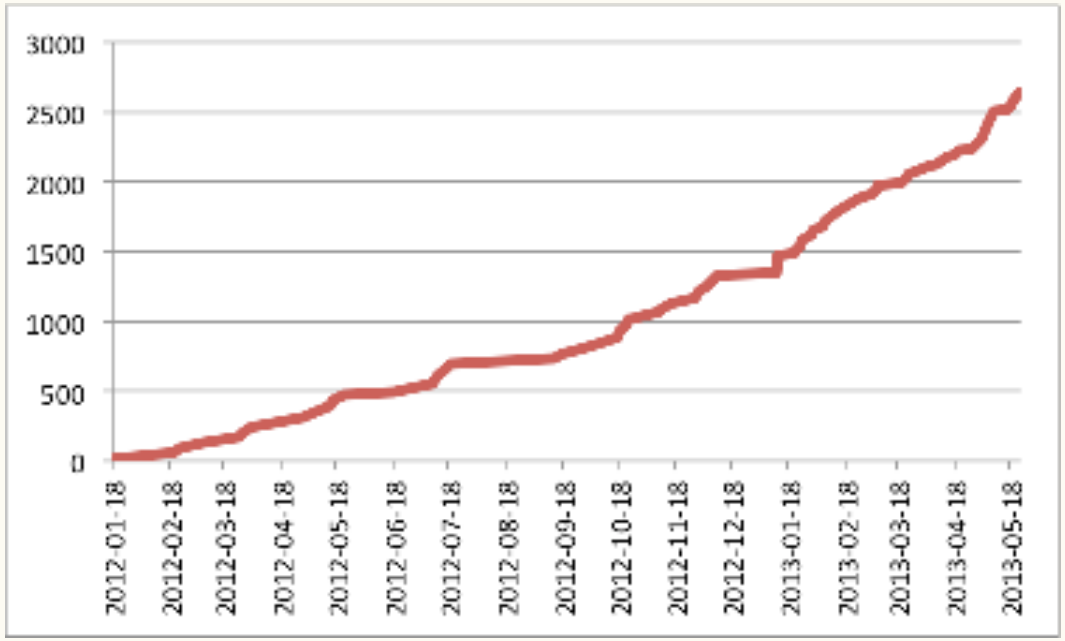
- Version control (co-operative work and reproducibility)
- Testing (not only unit tests!)
- Databases / NumPy

Aim is to teach *computational competence*.

Bootcamps/attendees

Some figures*

- 92 two-day workshops
- 100+ volunteer instructors
- 3000+ attendees

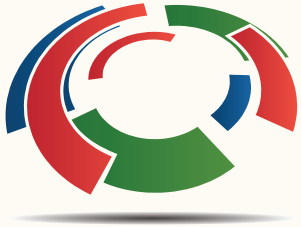


* January 2012 - July 2013

Software Carpentry in the UK

Software Sustainability Institute (SSI)

- Started collaborating in 2012
- 12+ bootcamps (10+ institutions)
- 400+ attendees



Software Sustainability Institute

SoundSoftware collaboration:

- University College London: 30 April - 1 May 2012
  - 4 students from C4DM
- Newcastle University: 14 - 15 May 2012

Created/delivered content for version control (Mercurial)

Audio/Music targeted bootcamps

York Bootcamp for Audio and Music Researchers

- Co-located with DAFx (Digital Audio Effects) conference
- University of York, 13-15 Sep 2012
- 3 day bootcamp
  - 2 days typical SWC (Greg Wilson)
  - 1 day audio content (Codasign)
- 26 attendees,
  - 8 different institutions (York, Surrey, East Anglia, Amsterdam, Edinburgh, Queen Mary, ...)

Audio/Music targeted bootcamps

Internal Queen Mary 2 day bootcamps:

- Media Arts and Technology (Feb 2013)
- Centre for Digital Music (March 2013

ISMIR 2012 (Porto)

- Tutorial focused on testing (hands-on, live coding session)

January 2014, London UK

- Tutorial on the ÆS 53rd Conference on Semantic Audio

Some (good) feedback

- Good atmosphere: “felt comfortable asking stupid questions”
- Live coding
- Open lessons/downloadable content
  - Autumn school (2010) available on YouTube
  - Code available in repositories
- Pair programming/Examples
- Social side (pub, other researchers from similar field)

- Bash, Python and Version Control are usually favorite topics

Some (not so good) feedback

Environmental/external issues

- too warm/cold, lack of coffee, overcrowded

Issues installing tools

- people bring their own computers

Too fast/too slow

- Heterogeneous groups
- Splitting people by level?

Does it work?

Participants express extremely high levels of satisfaction*:

- 85% learned what they hoped to learn
- 95% would recommend the workshop to others

Attendees are learning and applying some of what is taught*:

- Increases computational understanding
- Enhances habits and routines (leads them to adopt standard tools)

* *Aranda2012 (http://bit.ly/11WQz1Q)*

Some open questions

Many find hard to apply what we teach to their own work

- Audio specific days helped.

What happens if the bootcamps become mandatory?

- How to keep students motivated?

Can we actually say it is working?

- Do scientists become more productive?
- Do scientists share code more often?
  - Long term assessment needed (but how to assess?)

Thank you for your attention

Questions welcome!