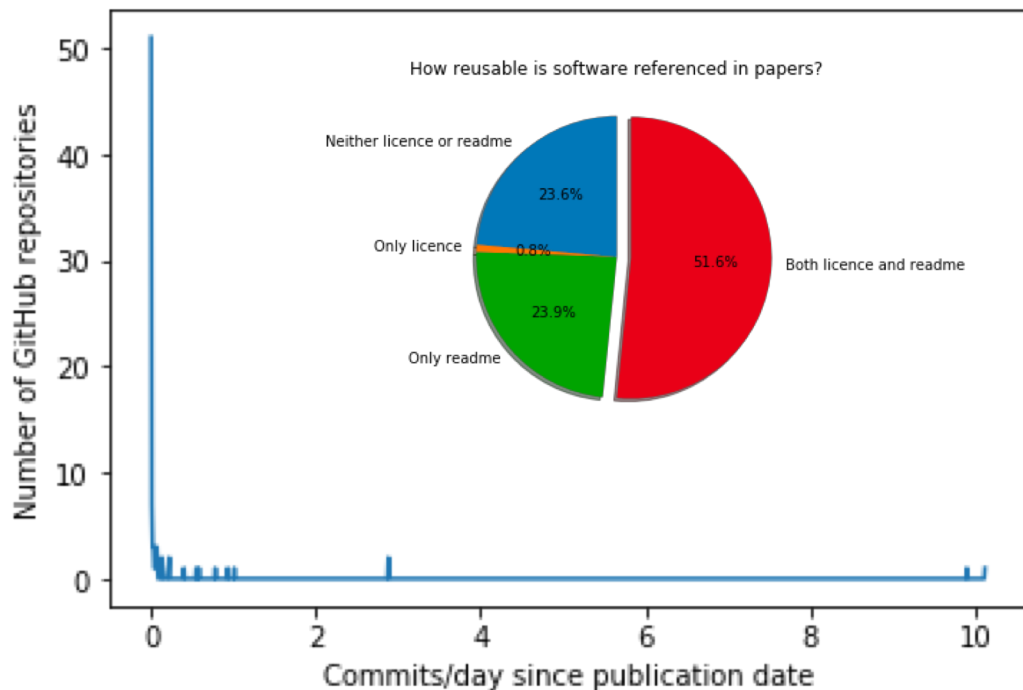


# Published, then forgotten



www.software.ac.uk

Does software referenced in publications get updated?



- Most software referenced in papers never updated after publication
  - Why would others use this?

How reusable is software mentioned in Open Access papers?  
An empirical study using code-cite DOI: 10.5281/zenodo.1209095

# Long, long, long tail



www.software.ac.uk

“What software do you use in your research?”

2958 responses (2014–16, 1261 participants)

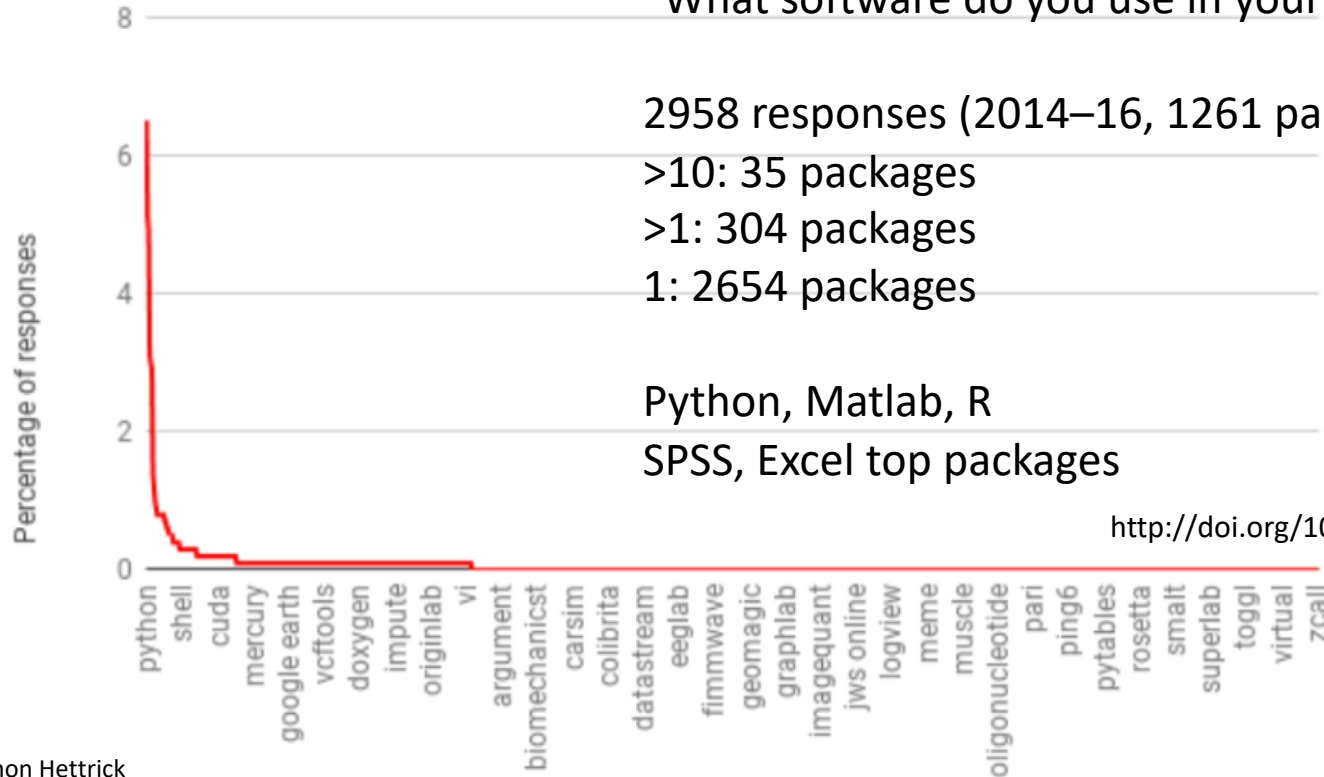
>10: 35 packages

>1: 304 packages

1: 2654 packages

Python, Matlab, R

SPSS, Excel top packages



<http://doi.org/10.5281/zenodo.60276>



# The Funding Paradox



www.software.ac.uk

- How do we incentivise researchers to write the smallest possible amount of new code?
  - The strongest incentive (for most researchers) is funding
  - Research funders would like to fund software that will be used by others
    - Not how peer review currently works
  - No easy way for funders to make decisions around when to fund new work vs when to fund consolidation
  - Can we understand how to make better investment decisions, based on available data?
    - When is a piece of software “on the up” vs “popular platform” vs “in decline”
      - See also <https://chaoss.community/>
    - How do we best fund each stage? And who should money go to?
      - This is the way to create career paths
      - This is also what we need to get right to sustain key pieces of software
    - What should URSSI’s role be in supporting this?

