Questions…

software carpentry

- How many of you test your code or scripts...
  - With a single set of input data, checking this gives an expected set of output data?
  - With multiple sets of input data?
  - With input data you know to be incorrect, checking that code or scripts behave as expected?
  - After every change you've made, to fix a bug or optimise your code or to add a new feature?
  - Using some form of automation e.g. a set of testing scripts or a test framework?

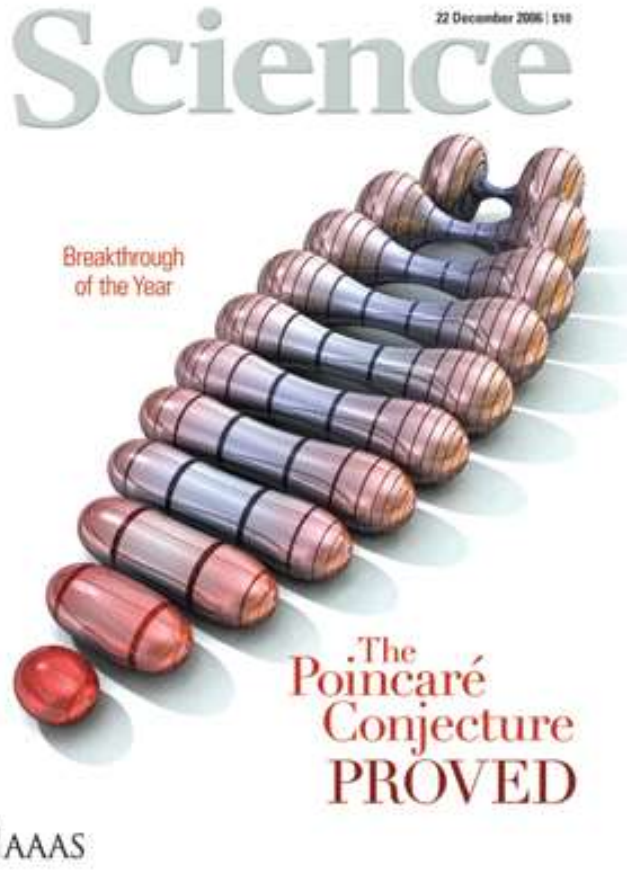
So why don’t we do it?

software carpentry

- “I don’t write buggy code”
- “It’s too hard”
- “It’s not interesting”
- “It takes too much time…”
- “…I’ve research to do!”

Why we should…Geoffrey Chang

software carpentry



We wish to retract our research article … and both of our Reports...
An **in-house data reduction program** introduced a **change in sign** for anomalous differences…
Unfortunately, the use of the multicopy refinement procedure still allowed us to obtain reasonable refinement values for the **wrong** structures.

http://software-carpentry.org

Testing allows us to check that our code…

software carpentry

- Verify that we have written our code correctly
  - …bug free, precise, accurate and repeatable
  - …after we've fixed bugs or made improvements
- Validate that we have written the right code
  - …produces the data we expect
  - …on any set of valid input data
- Fails gracefully…
  - …if given invalid input data
  - …if pushed beyond its limits
- And we can automate it!

Testing…

software carpentry

- Saves us time
- Gives us confidence that our code does what we want and expect it to
- Promotes trust that our code, and so our research, is correct

- Remember Geoffrey Chang!
- “If it’s not tested, it’s broken”
  - Bruce Eckel, Thinking in Java (3rd edition)