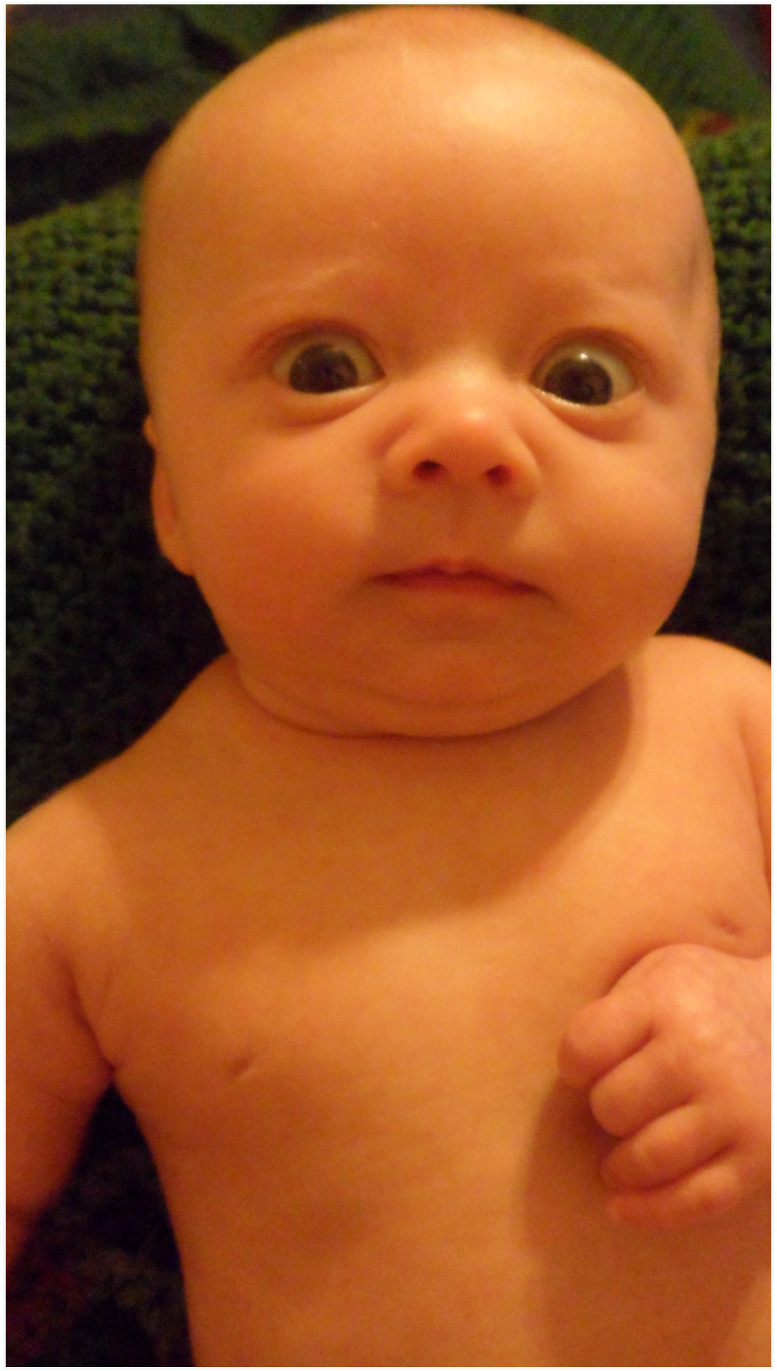
NOVEMBER 27, 2016

OUTRO TO PS239T

WHY ARE WE HERE?



AIMS OF THE COURSE

- **Skills** (the grammar)
    - Bash, Python, R, Git
    - Transfers between many languages
    - Relies on shared logics and structures
    - Constrains what can and can't be done
    - Used to make sense of things: shaping, analyzing, plotting
- **Applications** (the vocabulary)
    - APIs, webscraping, packages in Python and R, etc.
    - Used to observe and obtain things–in our case, data

WHAT WE DID

- ***Learn to learn***
- What we learned:
  - Navigate & operate effectively in a UNIX environment
  - Master basic Git and GitHub workflows
  - Write, execute, and debug Python code for data collection and manipulation, as well as other computing tasks
  - Write, execute, and debug R code for statistical analysis
  - Master the fundamentals of good data visualization in R
  - Find answers to things we won’t cover in this class

OBSTACLES TO LEARNING

- Unspoken/tacit knowledge among coders (“banana banana banana”)
- Learning to think like a computer (1s and 0s not as easy as you think)
- Becoming impatient while learning basic concepts (most errors lie here)
- Low endurance (the diligent bird builds the worm)
- Infrequent practice (fluency doesn’t come from weekends)
- Solo coding (teaching is learning)
- Copy-and-pasting instead of typing (ctrl+c is not your friend)
- Disorganized coding (would you write an essay without an outline?)
- Failing to comment code as you go (## what does this do?)
- Expecting to be Gandalf (10-20 lines a day)

(NON-)OBSTACLES TO GROWING IN YOUR CODING

- **You didn't learn how to do everything (or even how to do everything you want to do)**
  - If you had, it would have been a complete waste of your time. Languages and tools change
- **You'll forget things**
  - So do I. (Let us not forget the many "arghhh" moments in my transition from Python 2 to Python 3)
  - Comments and docstrings are key
- **You're not totally self-sufficient**
  - 2.7 million questions/year on Stack Overflow suggest no one else is, either

MOVING FORWARD

- Build off your final projects (reusable code is capital)
- Experiment now while things are freshest
- When things stop being fresh, go back to basics (remember swirl, codeacademy, D-Lab, etc.)
- Consider teaching

THANKS

“WHAT WOULD YOU DO IF YOU WEREN’T AFRAID?”