Programming for Neuroscientists

two perspectives

Emily Coffey, M.Sc.

Zatorre Lab

Outline

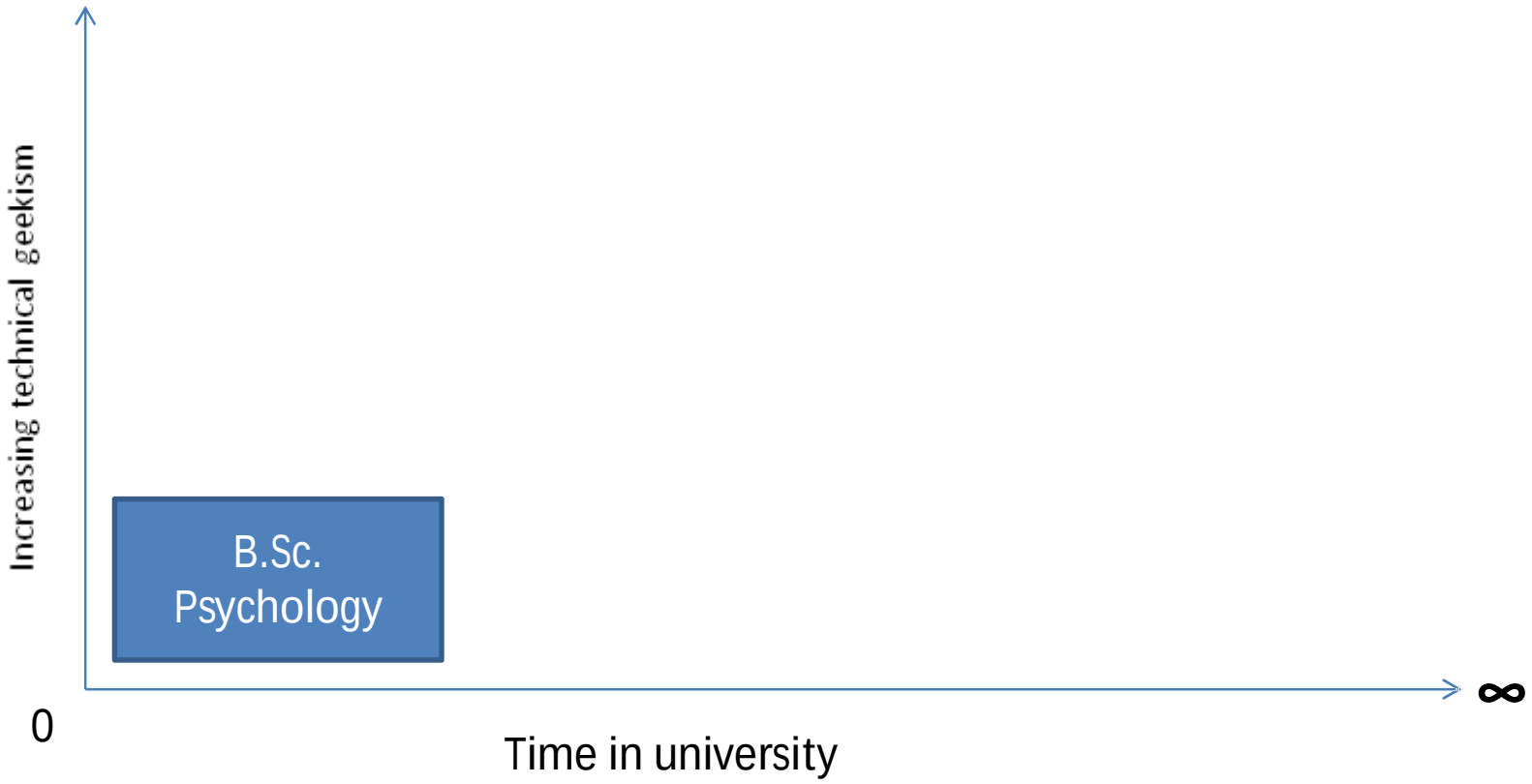
- Two perspectives:
  - Part I: The neuroscientist user without proper training (me)
  - Part II: The expert user (Robb)

Part I: the neuroscientist user

- **Why learn to program?**
- **How to learn?**
- **How long to learn?**
- **What are the key skills?**
- **Considerations for which language to learn (continued by Robb)**
- **Resources**

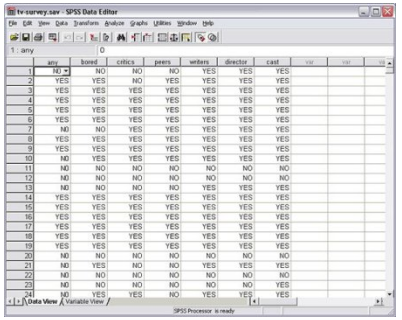
Why learn to program?

- **My trajectory through geekspace**



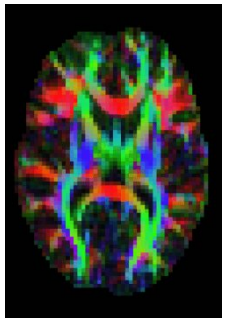
Why learn to program?

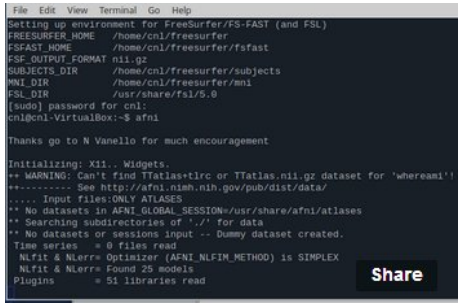
- **BSc thesis**

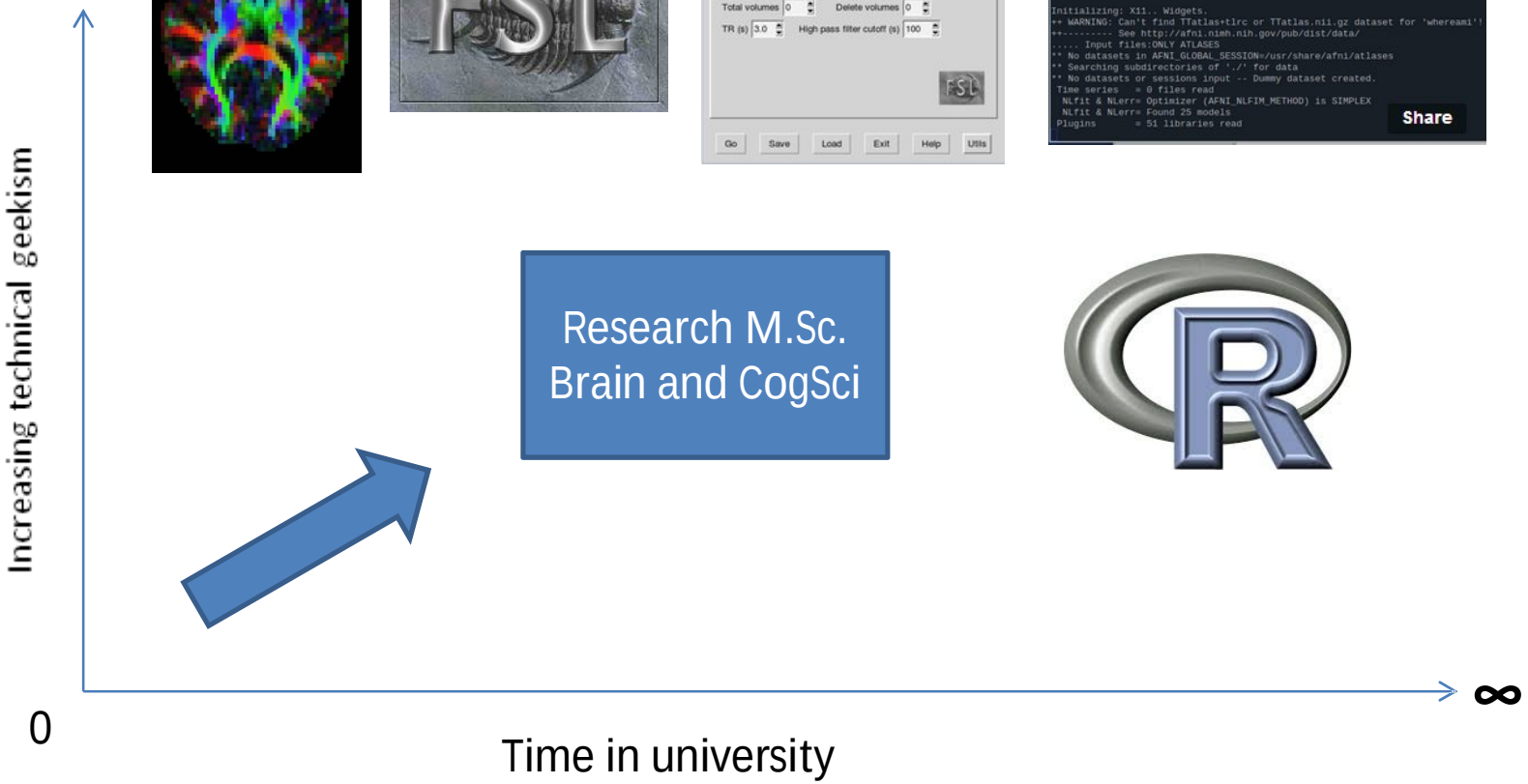


Images: http://elitechoice.org/wp-content/uploads/2008/09/sx02_flight_simulator.jpg
http://www.pilotfriend.com/home_files/summer_splash.jpg
http://i.telegraph.co.uk/multimedia/archive/02563/vw_laptop_2563199b.jpg

Why learn to program?







Why learn to program?

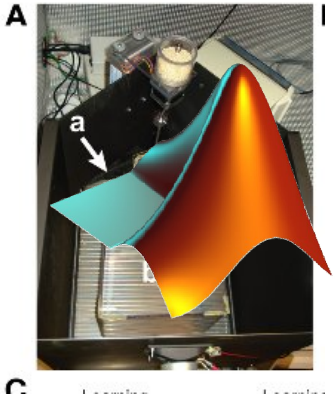
- **Aside: do you know what that is?**

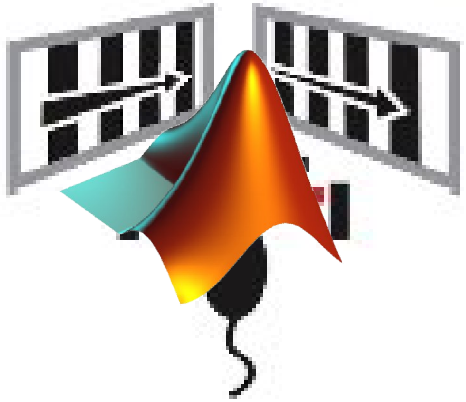
©Jeffrey Jay Foxx / NYC

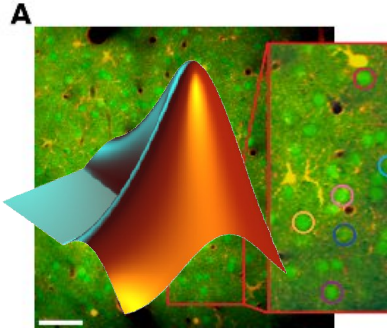
Images: http://worldharitage.com/wp-content/uploads/2013/09/Chichen_.gif
http://www.blogcdn.com/www.engadget.com/media/2012/08/google-street-view-chichen-itza-1345210908.jpg

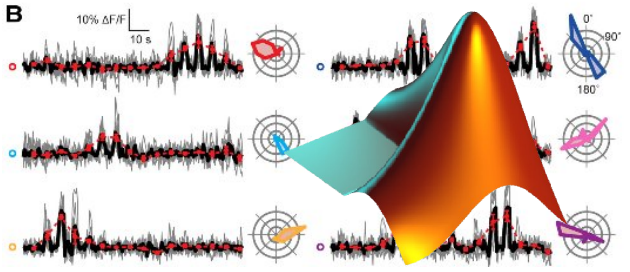
Why learn to program?

- **MSc Internship: two-photon microscopy with Pieter Goltstein, UvA**









Why learn to program?

- **MSc Internship: two-photon microscopy with Pieter Goltstein, UvA**

Why learn to program?

- **MSc Internship 2: NIRS + EEG to measure workload (for BCIs)**

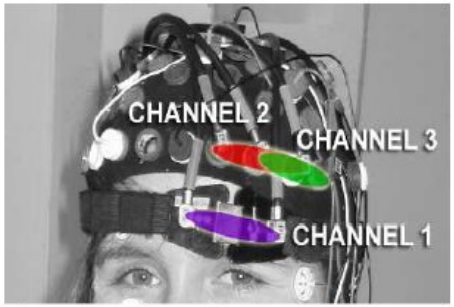


Figure1: optical channel positions (photo used with permission)

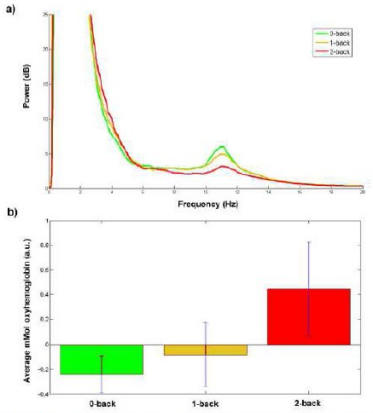
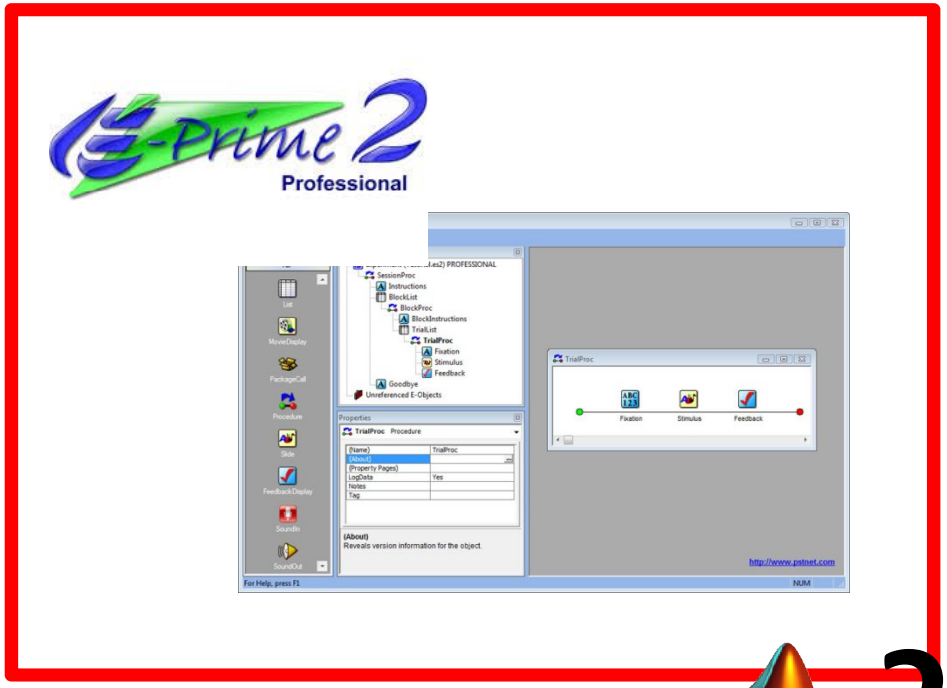
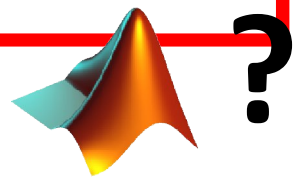


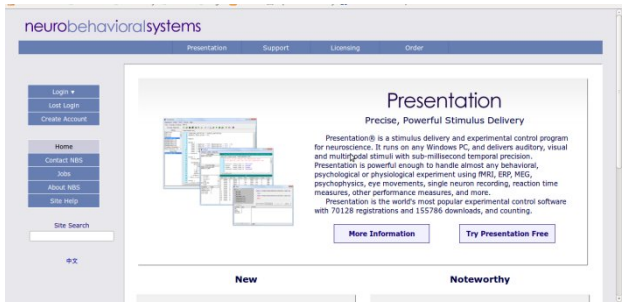
Figure 3: Sample visualization of a)EEG spectra by condition and b) NIRS oxyhemoglobin concentration from subject 2. Error bars represent the standard errorof the averages of the raw data within each block





?

Why learn to program?



Ph.D. Neuroscience

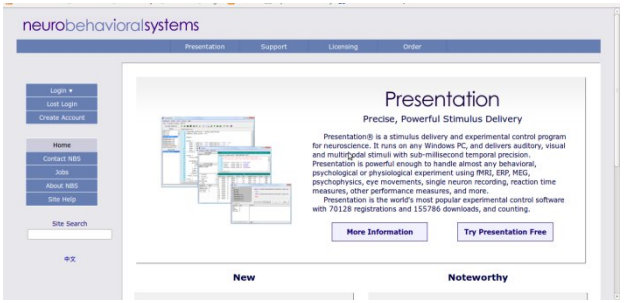
Increasing technical geekism

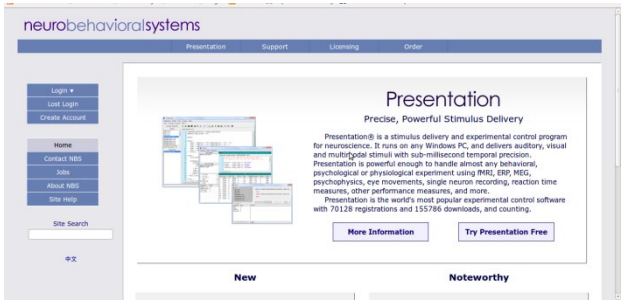
0

∞

Time in university

Why learn to program?





Why learn to program?

Point of no return

Ph.D. Neuroscience (of the auditory cognitive neuroimaging variety)

Increasing technical geekism

0

∞

Time in university

Why learn to program?

- **Benefits**
  - **FREEEEDOM**
  - **Independence** from computer experts, companies
  - **General skills:** don't have to learn a new interface per part of your project
  - **Stimulus generation and presentation:** precise control, more options
  - **Data collection:** automated, improved organization
  - **Analysis:** exploration, flexibility, batching and pipelines for large datasets and using neuroimaging tools
  - **Understanding:** stimulus creation, signal processing, etc.
  - **A useful and valuable skill for your future**

Why learn to program?

- **Drawbacks and challenges**
  - **Inertia:** painful to switch from GUIs
  - **Up front investment** in time and effort
  - **Need to continue practicing**, especially at the beginning, or it's gone
  - **Can be frustrating**
  - **Where to start**? Which language(s), and how to start?

How to learn?

- **Try to do useful work instead of (or simultaneously with) taking a course**
  - Help, online forums, tutorials as needed

[Picture removed]

How to learn?

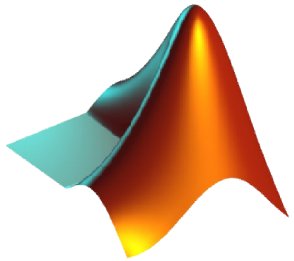
- Try to do useful work instead of (or simultaneously with) taking a course
- Do a little bit regularly
- Expect episodes of frustration and ineptitude

What are the key skills?

- Asking the right questions and specifying them precisely
  - Both for help and for making programs
- Deconstructing a problem into solvable steps
  - flow control
- Basic understanding of the elements, syntax and functions
- Troubleshooting, debugging and testing
- Documentation and version management

What are the key skills?

- **Asking the right questions and specifying them precisely**
  - **Both for help and for making programs**
- **Deconstructing a problem into solvable steps**
  - **flow control**
- **Basic understanding of the elements, syntax and functions**
- **Troubleshooting, debugging and testing**
- **Documentation and version management**



<- Both imperative, interpreted, object oriented languages

How long to learn?

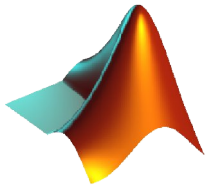
- From scratch to little projects?
- From scratch to pretty fluent?
  - Python may be a little faster

How long to learn?

- **To switch between languages?**
  - **You probably won't**

Considerations for language?

- Watch Robb's face
- MatLab and Python are not all that different
- Consider:
  - $?
  - What are you going to want to do with it? (function and toolboxes)
  - Do you have to share code with colleagues using one of them?
  - Is one more sought-after in your field (CV and future jobs)
  - Are you willing to learn both?
- In general, unless you have strong reasons if you are starting out Python is probably a good investment



Resources

- **MatLab:**
  - **Google "MatLab Tutorials" –** interactive tutorial, trial licence
  - **Prof. Marc Schönwiesner's course at UdM (PSY6976) –** visual and auditory stimulus stuff for cognitive scientists
  - **User community** on MatLab site
  - **YouTube**
- **Python:**
  - **There are regular high quality MOOCS on Python**
    - **Coursera has one starting March 24th, EdX has one that started a week ago**
  - **MNI Intro to Python workshop**
    - **At the BIC on Friday, March 7th, from 12:20 to 3:30 pm**
  - **Software Carpentry Bootcamp ($)**
    - **at PyCon, April 14-15**
  - **Local user communities**
    - **PyLadies**
    - **MNI user group? (Talk to Melanie Segado)**
- **R?**
  - **If you are primarily interested in statistical computing and graphing, this is a very popular specialized language, also many MOOCs and online resources**

Robb will now convince you to use Python.