

Tips and Resources for Scientific Coding

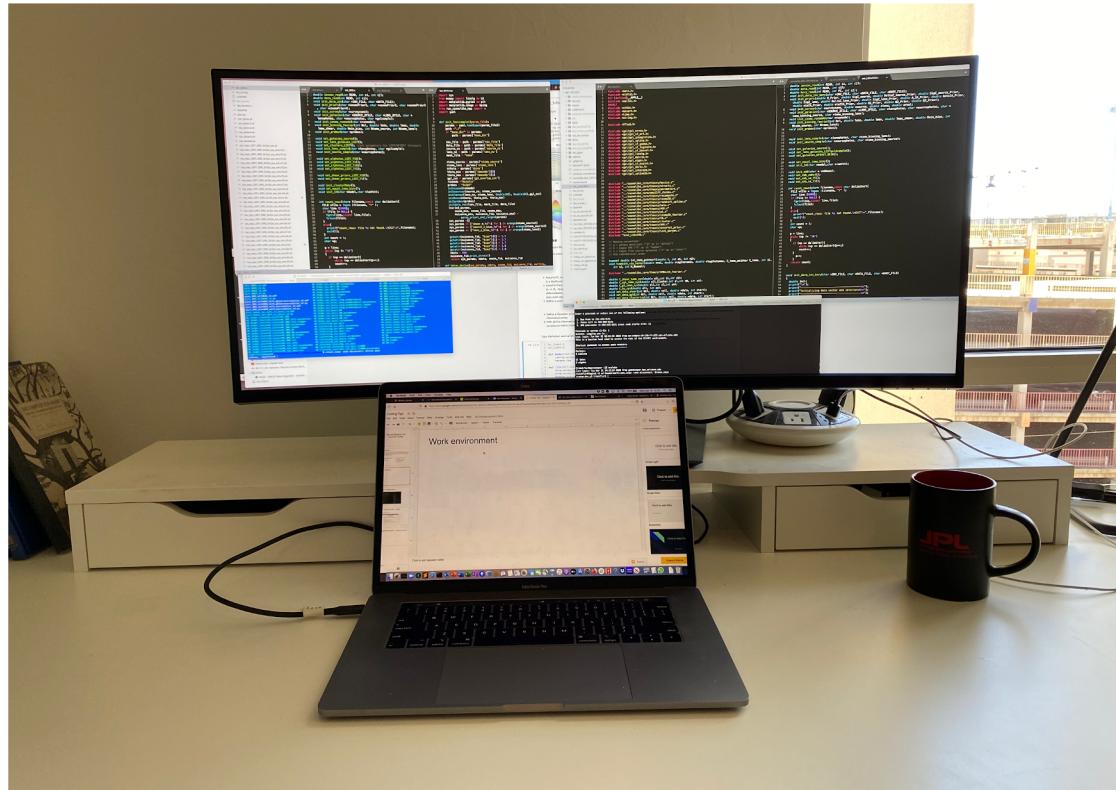
Quick overview

General remarks: This is not supposed to be a presentation but rather an interactive exchange of information. We will cover a variety of topics and hopefully everyone can share their thoughts experiences and questions. Please interrupt at any time and do so frequently.

Topics:

- Work environment, Editors
- Github
- Stackoverflow
- Python, ipython notebooks, numpy, scipy
- C and python interfaces
- Fun with ML/AI scikit learn
- Supercomputing

Work environment (take time to think about it...)



Most general workspace considerations apply to code developers as well:

- Monitor size (I chose a large one)
- Distraction free environment (I usually only code at my desk, sometimes outside)
- Portability (unhook laptop, done)
- Peripherals needed (I opted against mouse, keyboard, etc)
- I find it useful to walk or move around when thinking about code structure. Switching between standing/sitting desk can be useful

Text editors - my personal choice... sublime

A screenshot of the Sublime Text 3 interface on a Mac OS X desktop. The window title bar shows "Sublime Text". The menu bar includes "File", "Edit", "Selection", "Find", "View", "Goto", "Tools", "Project", "Window", and "Help". The status bar at the bottom right shows "Wed Apr 15 14:53" and "master [16] Spaces: 2". There are four tabs open in the editor:

- `init_SRD.c`: Contains C code for initializing SRD, including functions like `invcov_read`, `data_invread`, and `init_data_inv`.
- `init_emu.c`: Contains C code for initializing emulator parameters.
- `run_datav.sh`: A shell script with commands like `mpirun` and file paths.
- `sys_derivs.py`: Python code defining a function `init_lenssample1` that reads parameters from a YAML file and performs various calculations related to lensing samples.

The code in `sys_derivs.py` includes imports for `sys`, `numpy`, `linalg`, `matplotlib.pyplot`, `mpimg`, and `run_cosmolike_mpp`. It uses `yaml` to load parameters from a file. The function `init_lenssample1` takes a parameter file as input and performs operations like setting up source and lens galaxies, calculating shear priors, and initializing IA models.

For other options see
this review

<https://www.techradar.com/best/best-text-editors>

Text editors - my personal choice... sublime

<https://www.sublimetext.com/download>

The screenshot shows the official Sublime Text website's download page. At the top, there is a dark navigation bar with white text containing links for Home, Download, Buy, Blog, Forum, and Support. Below the navigation bar, the word "Download" is prominently displayed in a large, bold, dark font. Underneath this heading, there is a paragraph of text explaining that Sublime Text 3 is the recommended version and is available for Windows, OS X, and Linux. Further down the page, there is information about the latest version (Sublime Text 1.x) and a portable version. At the bottom, there is a note about the evaluation period for Sublime Text.

Home Download Buy Blog Forum Support

Download

Sublime Text 3 may be downloaded from the [Sublime Text 3](#) page. This is the recommended version of Sublime Text to use, and is available for Windows, OS X and Linux.

The latest version of Sublime Text 1.x is [Sublime Text 1.4](#)

It's also available as a [portable](#) version, to run off a USB key.

Sublime Text may be downloaded and evaluated for free, however a license must be purchased for continued use. There is no enforced time limit for the evaluation.

Share your editor experience... what have you tried?

Github

<https://guides.github.com/activities/hello-world/>

Let's try it out! You will learn how to

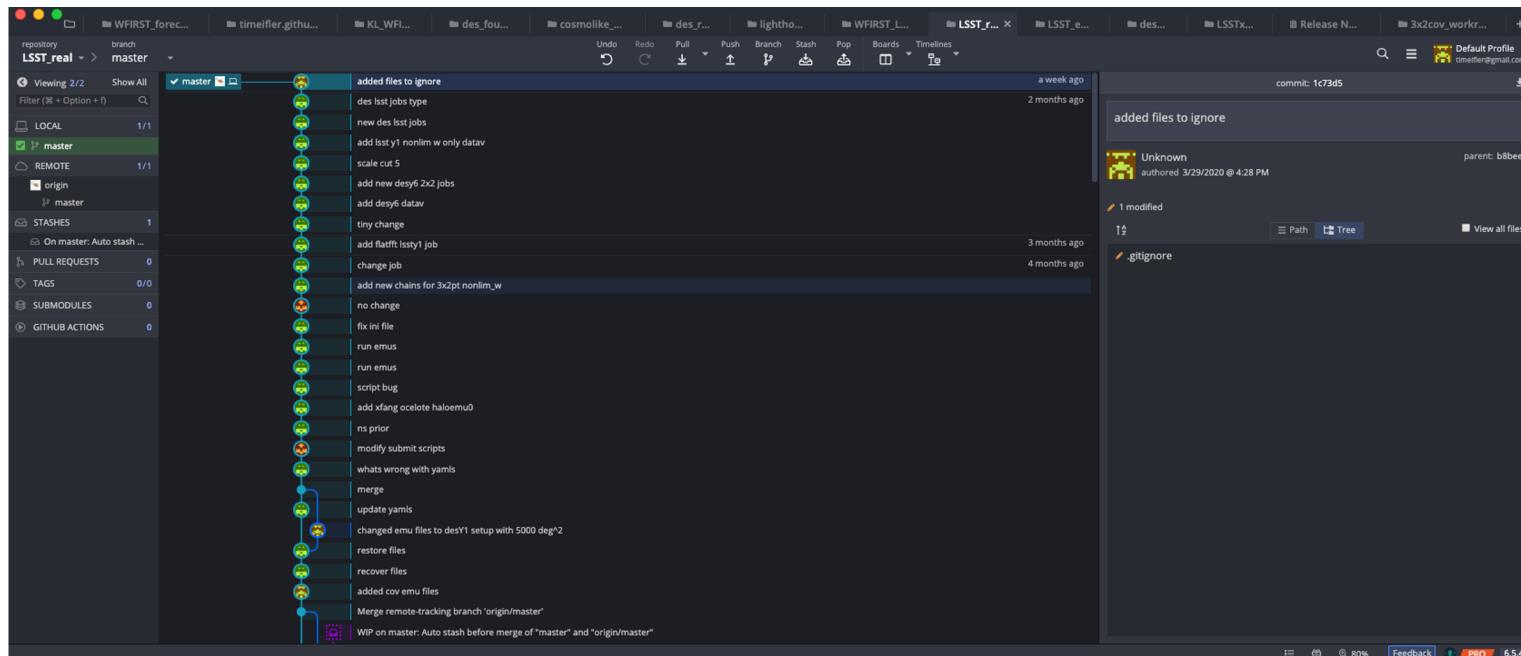
- Create and use a repository
- Start and manage a new branch
- Make changes to a file and push them to GitHub as commits
- Open and merge a pull request

This exercise requires no coding, just a github account. After completion (~20mins), you will be familiar with almost all important github features and can claim that you know your way around in github (job interviews...).

How to manage github efficiently

Try one of the github GUIs: see here <https://acodez.in/git-gui-clients/> for options

My choice: <https://www.gitkraken.com/> (also has a very good free version)



Go through
features in
class.

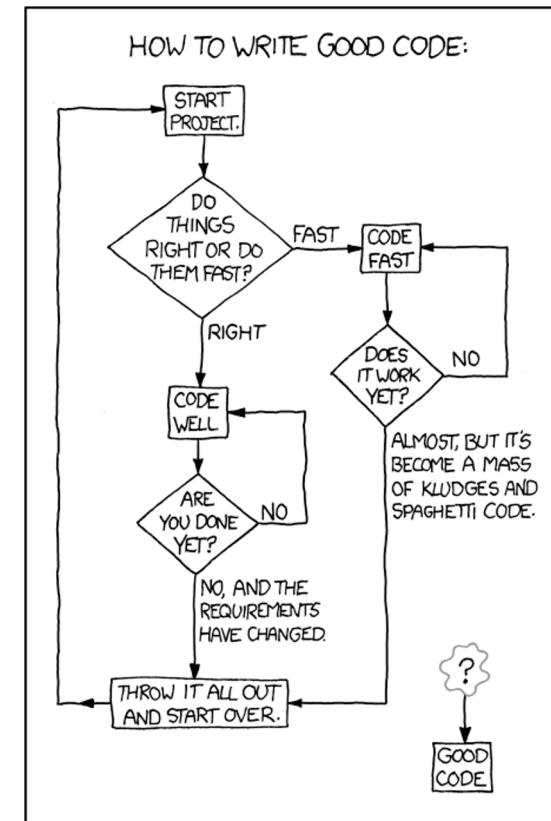
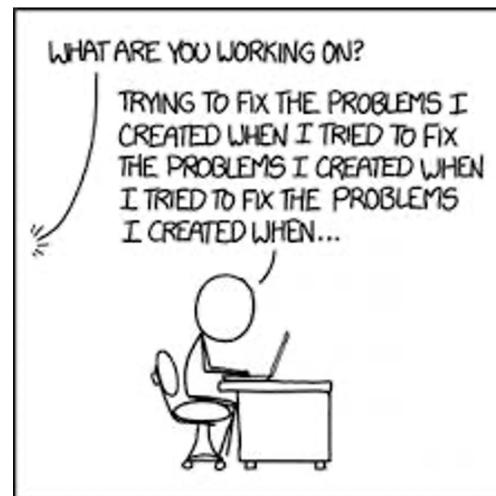
Share your GUI experience... which github GUIs you tried?

Stackoverflow <https://stackoverflow.com/>

The typical way to solve coding problems is:

- 1) start coding, 2) fail, 3) google your error message, 4) which brings you to stackoverflow most of the time, 5) implement the solution, 6) repeat

stackoverflow has a point system and ranking for its problem solvers. High-ranked solvers get really good job offers from top companies.



Python ... argh... where to begin?

Main page:

<https://www.python.org/about/gettingstarted/>

Beginners (really useful):

<https://www.learnpython.org/>

<https://wiki.python.org/moin/BeginnersGuide/NonProgrammers>

Beginners and Advanced:

<https://wiki.python.org/moin/BeginnersGuide/Programmers>

There are a gazillion python tutorial on the web... just pick one from the lists above, and then another one, then another one...



Python ... argh... where to begin?

Really nice Physics/Astro notebooks:

<https://www.numfys.net/>

Stats, CS notebooks+tutorials:

<https://people.duke.edu/~ccc14/sta-663/index.html>

These resources are great to learn about astro/physics/stats concepts and how to solve them with ipython notebooks... learn through looking at those!



Interfacing python and c code

There are many, many ways... my personal solution is rather simple

Ctypes: <https://docs.python.org/3/library/ctypes.html>

Will show example from Cosmolike LSST_emu project in class

Machine Learning and AI - Some fun examples

Supervised learning:

<https://quickdraw.withgoogle.com/#>

3D Visualization of a Convolutional Neural Network:

<https://www.cs.ryerson.ca/~aharley/vis/conv/>

Machine Learning and AI - tutorials

Overview of some really nice tutorials (notebooks you can download from github)

<https://www.kdnuggets.com/2016/04/top-10-ipython-nb-tutorials.html>

Let's look at this one...

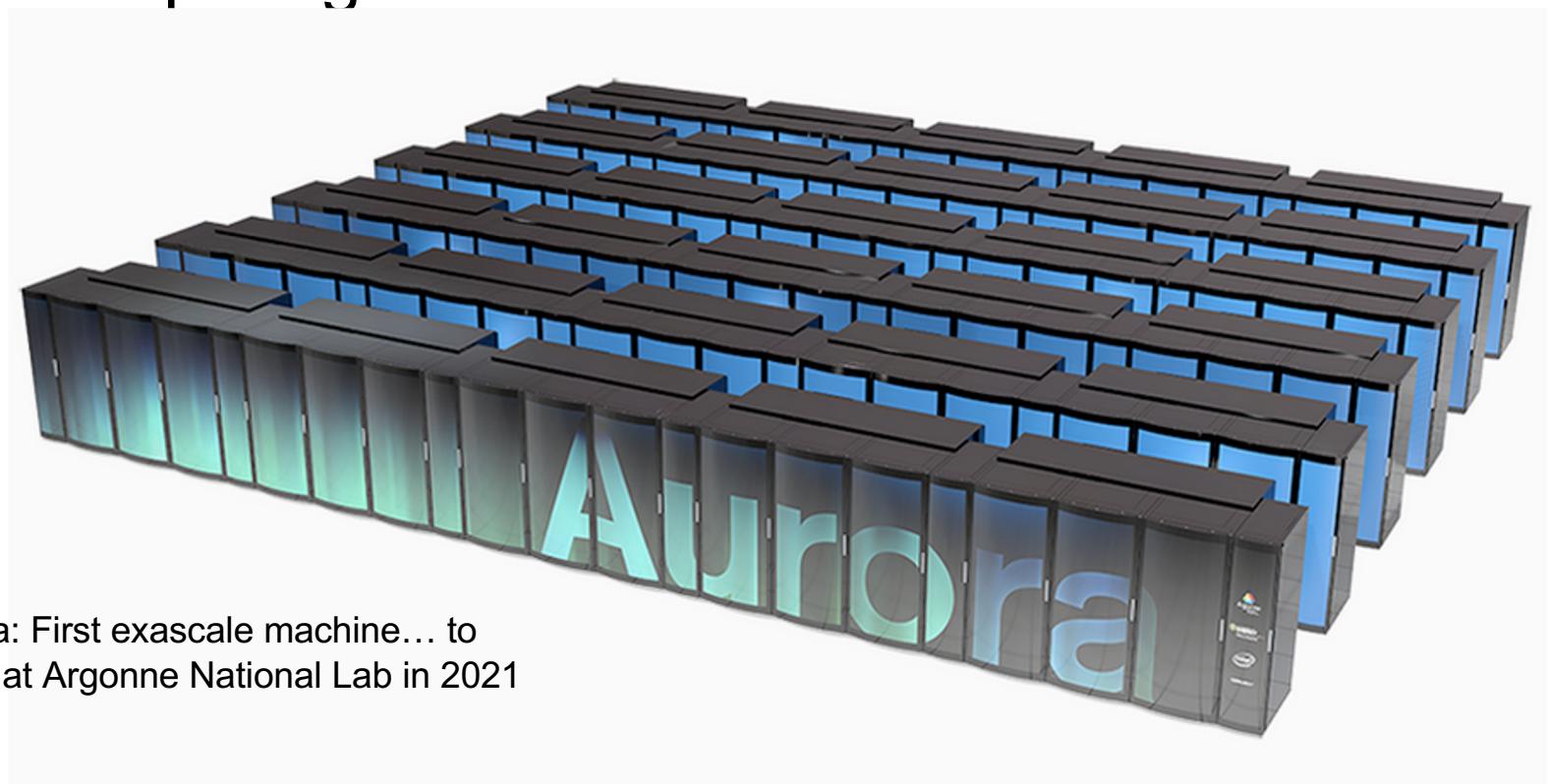
https://github.com/jakevdp/sklearn_tutorial

Getting into AI: my path...

- 1) Watch movie AlphaGo: It's free and pretty great:
<https://www.alphagomovie.com/>
- 2) Instead of news... read this regularly <https://towardsdatascience.com/>
- 3) There is even an explanation of the AlphaGo algorithm:
<https://towardsdatascience.com/understanding-alphago-how-ai-thinks-and-learns-advanced-d70780744dae>
- 4) Original paper by the team: <https://www.nature.com/articles/nature24270.epdf>
- 5) More examples, tutorials for ML/AI : <https://scikit-learn.org/stable/>

The scikit package in python contains most tools AI/ML developers need

SuperComputing



Aurora: First exascale machine... to
arrive at Argonne National Lab in 2021

SuperComputing - Getting involved is easy

- Any UA faculty can get an account and free time on the UA SC ocelote and that's a pretty great system
- Ask your advisor, get added to his/her account, create a simple program and submit
- <https://public.confluence.az.berkeley.edu/display/UAHPC/User+Guide>
- New UA HPC PUMA has arrived...



SuperComputing - Getting involved is easy

- Many Astro and Physics projects have access to National HPC resources, ask about summer projects!
- NERSC (DOE)
<https://www.nersc.gov/>
- Pleiades (NASA)
<https://www.nas.nasa.gov/hecc/resources/pleiades.html>
- XSEDE (NSF)
<https://xsede.org/>



Hopefully this was useful - some takeaways

- 1) Create your work environment
- 2) There are fantastic resources on the web. Go there and look at them, don't overthink, don't be discouraged
- 3) Team up: It's boring to do this alone, get a group together and work through some examples. Just get them to run and explain code to each other. Code in pairs... yep, pair-coding is a thing. Share screen and get a coffee.
- 4) Ask for research projects with code development/supercomputing aspects
- 5) Advanced: Solve problems on stackoverflow - top companies are hiring from the stackoverflow rankings