Welcome to The Carpentries Etherpad!

This pad is synchronized as you type, so that everyone viewing this page sees the same text. This allows you to collaborate seamlessly on documents.

Use of this service is restricted to members of The Carpentries community; this is not for general purpose use (for that, try https://etherpad.wikimedia.org).

Users are expected to follow our code of conduct: https://docs.carpentries.org/topic_folders/policies/code-of-conduct.html

----

Links

Etherpad: https://pad.carpentries.org/2023-ADACS-ECR-WOrkshop
Workshop page: https://adacs-australia.github.io/2023_ASA_ECR_Python_Workshop/
Github for workshop: https://github.com/ADACS-Australia/2023_ASA_ECR_Python_Workshop
Pauls' email for feedback / questions: paul.hancock@curtin.edu.au
Pre-workshop survey: https://forms.gle/cpoHF72b4Ah4k8co9
Post-workshop survey: https://forms.gle/HJR4ERAJRDEHZiFc6

Melbourne

Day 1

EXERCISE — Pylint

Patrick’s example: src/skysim.py:34:0: C0305: Trailing newlines (trailing-newlines)
Problem: there are blank lines at te end of the Python file
Solution: delete the blanks lines
Test if it works: It did!

Problem: "W1514: Using open without explicitly specifying an encoding (unspecified-encoding)"
Solution: Apparently we should put encoding="utf8" or whatever encoding we prefer in our "open" statement

Problem: "R1732: Consider using 'with' for resource-allocating operations (consider-using-with)"
Solution: We can open the file like this instead:

```
with open('catalog.csv','w') as f:
```

and make sure everything under it is indented by one extra level now.

Problem: C0114: Missing module docstring (missing-module-docstring)
Solution: Add three double quotes around a description of the document to get rid of this error. Put it at the very top of the file. Something like as follows:

```
"""
Description of code
"""
```

Test if it works: It did

Problem: C0103: Constant name "ra" doesn't conform to UPPER_CASE naming style (invalid-name)
Solution: change 'ra' to 'RA'

Problem: from math import *
Solution: import math as mh (and then add before the functions cos and pi mh.)

Problem: C0209: Formatting a regular string which could be a f-string (consider-using-f-string)
Solution: make it an f-string, i.e. print(f"{i:07d}, {RAs[i]:12f}, {DECs[i]:12f}", *file*=f)

Problem: C0413: Import "from random import *" should be placed at the top of the module (wrong-import-position)
Solution: move the line to the top of the module

prob: *# ^- Formatting a regular string which could be a f-string (consider-using-f-string)*
sol: could use {0:07f} rather than {0:07d}

Probelm: Constant name "nsrc" doesn't conform to UPPER_CASE naming style (invalid-name)
Solution: change to 'NSRC' as it is a constant variable

Problem: constant dec doesn't conform to standard, change its case

Installed pylint, got error message working, agreed with W0401 discussion that * is not a good convention

MY FAVORITE PYTHON PACKAGES

Patrick: numpy, gstreamer, astropy, multiprocessing
Leonie: pyraf, bokeh, scikit-learn
Rory Elliott: commonly used modules = [numpy,astropy,pandas,threadcount,matplotlib,math,re,lmfit].
Most obsure module = [threadcount]
Numpy, pandas, matplotlib, astropy
Geoff: matplotlib, scikit, os, numpy, pandas,   rare [mayavi, h5py]

Manu: numpy, matplotlib, h5py, seaborn, scipy. Obscure module: cKDTree (scipy.spatial)
Doran: wellpathpy
Gio: opencv, lenstronomy, emcee
Behzad: uncertainties (does error propogation for you!), healpy, TensorFlow
Robert: pandas, scipy (for the weird functions not avaialable in numpy), h5py
Violet: ctapipe ;) , PyROOT , scikit-learn
Simon: pandas, numpy, matplotlib, astropy, tqdm, click
Rami: Astropy , numpy , pandas , matplotlib, scipy, tqdm
Imogen: matplotlib, numpy, astropy, astrodendro
Ryan: MCEq, astropy, partISM, lmfit, scipy
Ella: lmfit, pymc3, pandas
Mark:Pandas, matplotlib, scipy

GROUP PROJECT PITCHES

(Rory's ideals)
-creating a function series that takes edge-on/face-on galaxy fits files, identifies major SB clumps, and makes a catalouge of their properties
-creating a function that allows multiple IFU galaxy fits files to be stacked together without any issues existing where spatial/wavelength dimensions/scales are missaligned
-

- script for bootstrapping / fitting and plotting your favorite function (Leonie)
-calculate gaussian mixture models(Leonie)
-script that sets up new project structures automatically (with documentation) (Leonie)
- plot/calculate active satellite orbits (Ella) - interested (Ryan) interested (Rami)
-Creating a 3D map for pulsars and molecular clouds in our galaxy. -Rami
- write package that generates gamma-ray and neutrino fluxes from some proton distribution and basic gas distribution and plots them against each other, and calculates effective events for IceCube - Ryan (interested in proj= Rory)
- sequence model that takes in a music file of your choice and composes music in that style -Doran
- group project idea creator - Simon
- script that grades your group project idea - Simon
- script that critiques pylint's grievances - Simon
Writing a fast Fourier transform :) — Patrick (interested: Doran)
Writing a slow Fourier transform :(— Simon
Script that reads galaxy spectra and identifies redshift from selected lines - Manu -
Script that reads mag - colour of stars in a star cluster in order to make HR diagrams and derive ages through isochrones. -Manu - (interested in proj= Rory)(interested in project: Chan)
Rewrite my code for statistical lensing into a module - [Gio]
Rewrite my rubbish mixed fortran/python plottiing codes as pure python apps using functions stored in a module: (solo) Geoff
Write some basic scripts for accessing data stored in HDF files and visualise them using python/matplotlib (Geoff)
- Write a mixed C+Python pipeline that does the hard/performance-intensive work in C and the analysis in Python (Robert)
How do you call fortran from python and vice-versa - a few simple examples (Geoff)
Rewrite output of current project fortran program as a HDF file and then write simple python scripts to

extract data and visualise them - Geoff (solo project)

Group project names (and members):

SpaceXYZ - Ella , Ryan, Rami, Jemma

Ecclesia - Geoff (facilitator), members (Geoff)

Description: Ecclesia generates HDF files from the text output files of CATAPULT (a fortran code which generates dust particle trajectories around protostars) and then uses a suite of python scripts to visualise these files.
The python codes are: plt_scalar for ploting scalar quantities (temperature, density etc), plt_stream for plotting streamlines of gas and dust velocity and plt_vector for plotting 3d vector plots.

Goals: (1) Design structure of HDF files, (2) Using fortran bindings of the HDF library generate HDF files from existing output files, (3) Import data from HDF files into test python script; the python script should have command line options; (4) Write plotting functions for contour, scalar and streamline plots and incorporate into a module.; (5) write python codes plt_contour, plt_scalar and plt_stream using this module.

votetally - Violet (facilitator), Imogen, Simon [Remote - Adelaide]
Description: Tally up votes for a university club, for a number of roles each with varable number of positions. The voting implements the Hare-Clark electoral system and declares the incoming committee.
https://github.com/vmharvey/vote-tally/tree/devel
Goals:
0. Funnier name

1. Decide on format of file that the votes are read from.
2. One person for one role (single transferrable vote).
3. Multiple people, one role (Hare-Clark)
4. Varying people, multiple roles
5. Bonus: Candidate preferences for roles
6. Pass pylint.

Roles: president, social media manager, etc.

Isocreams: Chan (facilitator), Rory

myfftpy - Doran(facilitator)
desc: a custom implementation of the fast fourier transform algorithm

DIORAMA (**D**o **I**t y**O**u**R**self simul**A**tion co**M**p**A**rison): Robert (facilitator), Giovanni, Manu, Nicolas
Description: Diorama aims to retrieve some information on galaxy evolution from numerical simulations (we start targeting one between EAGLES/MILLENNIUM/ILLUSTRISTNG/FIRE...).
The code is Object Oriented and aims at learning this programming framework.

DAY 2

PAPER WRITING DISCUSSION EXERCISE
Qn (1) — if you draft an excellent paragraph and delete it, how can you retrieve it?
Qn (2) — how do you manage colloaboration between 5 coäuthors?

Rory: (Q1) retrival can either be done using version control, or through a doccument backup system. (Q2) in word, changes could be tracked temporarially through the comments/suggested changes system (however, these changes would eventually need to be comitted to.

Geoff (Q1) - Keep two directories of Latex files, scratch and final and a script that allows you to move a file from scratch to final, work always on the scratch copy and only move to final when you're happy that its OK. You can then copy back from final to scratch versions if you make an error and need to retrieve the. copy.
Geoff (Q2) - You need proper version control for this! If there is a principal coordinating author you could of course ask the co authors to write their sections. The coordinating author then puts together the draft and then circulates the draft version for comment and controls the repository.

Simon - (1) Depends if i've got a rendered PDF lying around somewhere...otherwise I tend to comment out things I'm not 100% sure I'm deleting (Overleaf. With the free version I only get 24hr of revisions). Also every major milestone I make a new copy of the Overleaf file (currently up to paper2v4).
(2) Other people send me their feedback, I implement it. Stay away from my original.
Violet: Use LaTeX for managing the paper. One line in the .tex file for each sentence. Commit to git at the end of each day's work. Accept comments on the rendered PDF with line numbers.
Doran - think really hard and draft another / use version control or some other collaboration tools
Sergey: Latex (overleaf) version track. Edit one-by-one (e.g. if it is about article writing) or indicate changes by the color. It does but in a limited number of changes. In can be compared in the History, and can be checked which particular changes were done in comparison with necessary verion (even in a free version)
Manu: Qn1 - Overleaf and download. Periodically backup on hard disk. Q2: Overleaf (for writing), GitHub (for coding).

Robert Q1: I usually use Overleaf, since my institution provides paid accounts (which come with a built-in version control system) and some other fancy features. If I didn't have that service I would probably implement version control myself.
Robert Q2: If I'm the main author I'm writing most of the paper anyway. Once my draft is 70-80% done I circulate it among the co-authors and we use latexdiff to see what they added/changed to the paper. It's kind of tedious so Overleaf is better anyway.

git issue (Rory) I've made a repo in a folder that contains src\anothermodule\sky_sim\__init__.py. I've tried git add src\anothermodule\sky_sim\__init__.py, but git returns a message claiming that no files were added. How should I navigate this?

GIT (AND OTHER) WORKFLOW DISCUSSIONS

Things to think about:
Does this workflow scale with team size?
Is it easy to undo mistakes and errors with this workflow?
Does this workflow impose any new unnecessary cognitive overhead to the team?
“Workflows are good and important”. Try to say something wiser than this. :) — Patrick

-(Rory): workflows should be kept low in technical intricacy such that teammembers do not have alot of their time dedicated to simply navigating it. However, it should be complex enough to manage multiple inputs from various individuals. The following questions should also be understood: "does the workflow allow divergent changes in a script (each branch suggested by a separate individual) to be negotiated before being comitted?", "is the history of the workflow eassily accesed?", "is the workflow host(s) secure, and can backups of existing work be easily taken?", "does the workflow update individuals on new commits to the project in a reaonsable and timely manner?, ""
Do you need to have a heirarchy with a group project - that is do you need to have some "authority" that leads the group and resolves conflicts or can you get away with an "anarchist syndicalist collective"? - Geoff
How can you "wind-back" a project to an agreed prior checkpoint?

CODE REVIEW EXAMPLES

"LGTM" (Looks Good To Me) — Patrick
"Euclidean distance metric isn't appropriate here" - Simon
this will mean we end up with less than `nsrc` sources by the end - Violet
Apply filter through make_stars rather than through main() - Chan
"I am not sure if looping through the array elements is the best approach in this situation. It could be completely avoided if instead of lists we use numpy arrays and take advantage of vectorisation." - Robert

Group AND/OR solo project: Tuesday's goals

(1) Get something into your git repository (if you didn't yesterday)
(2) Every group member should create a feature branch and implement a (small!) feature
(3) As many PRs as possible should be made, reviewed (by yourself if solo), (edited if necessary), and merged in
Optional but highly recommended task:
(4) Add a link to your Github repo (if public) to the Etherpad

Patrick/Leonie repository: https://github.com/leoniechevalier/ECR-Workshop
vote-tally repository: https://github.com/vmharvey/vote-tally/tree/devel
myfftpy repo: https://github.com/08dhuh/myfftpy/tree/dev
diorama repo: https://github.com/EMventura/DIORAMA

Day 3

HOW DO YOU ***CURRENTLY*** TEST YOUR CODE?
-(Rory): my methods for testing are using try/except conditionals for different lines to check where unexpected results are occuring, using "print" (python function) to check if data strucutre is what I expect, separating code into various functions such that each can be checked individually, making figures,
Ryan: Within Jupyter notebooks by hand
Plot something and compare with other people results if possible
Violet: Hand-testing, plotting, looking at the physics output and thinking about if it's correct.

Ella: By hand with print statements in c++, with debug mode in VS for python
Manu: Lots of print statements if the code is failing, plotting results for the physics. When in C using the debug mode for the compiler.
Rami: plotting the results as test method
Robert: Using debug mode in python, and manually testing it in C.
Doran: small scale test using Jupyter notebook in separate cells
Chan: Usually use the 'print' function or look at plots to make sure the code is running and gives some form of an output to make sure we are on track

PYTEST COVERAGE LINK:

https://pypi.org/project/pytest-cov/

get_radec() DOCSTRING EXAMPLES

Ryan-

```
    Get coordinates for the Andromeda Galaxy in units of right ascension (deg)
    and declination (deg), based on the (h:m:s) and (deg:m:s) coordinates from Wikipedia.

    Returns
    -------
    ra : float, (deg)
        The right ascenstion of Andromeda in degrees
    dec : float, (deg)
        The declination of Andromeda in degrees
```

Violet:

- Return Andromeda location.
- Returns the location of Andromeda from wikipedia in RA/Dec in units of
- degrees.
- Returns
- -------
- float,float
-   The RA and Dec in degrees
- References
- ----------
- https://en.wikipedia.org/wiki/Andromeda

Rory

```
   r"""
```

```
    Takes the Andromeda galaxy Ra Dec from Wikipedia, and converts the given values into a Ra and Dec value in units of decimal degrees.

    Parameters
    ----------
    None

    Returns
    -------
    ra : float
        ra of andromeda galaxy in decimal degrees
    dec : float
        dec of andromeda galaxy in decimal degrees
    type_without_description

    References
    ----------
    https://en.wikipedia.org/wiki/Andromeda_Galaxy
    """
```

EXAMPLE OF A REALLY WELL DOCUMENTED/HAS DOCUMENTATION YOU LIKE PACKAGE THAT YOU USE

-(Rory) When using it for honours, PPXF (penalized pixel fittitng) has a large pdf with expansive explanations of each input parameter, when to modify them, and what the consequences of modifying them were. This allowed much more detail to be shared compared to what would be in a regular docstring

https://rombus.readthedocs.io/en/latest/getting_started.html --- (as long as you don't look into the "how it works section"

https://ctapipe.readthedocs.io/en/latest/api-reference/index.html

Robert: https://docs.h5py.org/en/stable/ -- a package I end up using almost everytime to interface with HDF5 files
https://docs.streamlit.io/ -- beginner friendly, with lots of concrete examples and easy-to-follow instructions
Chan: https://lscsoft.docs.ligo.org/bilby/
https://dendrograms.readthedocs.io/en/stable/index.html
Rami : https://docs.astropy.org/en/stable/wcs/index.html

POSSIBLE OTHER CLI OPTIONS FOR SKYSIM

--radius (for radius size around the central point)
-n (for number of points you want to generate)
--pi (a boolean for whether you want to use the code to Monte Carlo estimate pi (given we're generating points in a square, then clipping to a circular region within the square))

DEBUGGING HORROR STORIES

(Patrick) Here’s a C one: I had a structure with (let's say) three variables in it: { energy, position_x, position_y }. I then edited it to add a fourth variable: { energy, momentum, position_x, position_y }... but

I compiled half the program with one version of the structure and the other half with the newer version. So some code did point.momentum = ... and the momentum value got written to 'position_x'. (Python *does not* have this problem. Phew) Another (not me): https://en.wikipedia.org/wiki/Knight_Capital_Group#2012_stock_trading_disruption "RLP code repurposed a flag that was formerly used to activate an old function known as 'Power Peg' ... Knight Capital took a pre-tax loss of $[USD]440 million."

-(Rory) I had one with a Javascript 3D environment I was building where a controlable object was not working as intended despute me playing around with various input values. After tones of experimentation, I found that I'd
defined a variable twice in two difference places, and the variable I was modifying (the one listed later) had it's name wrong by one letter (which was difficult to pick out). Of course, Javascript does not see this as an error, but instead another variable entirely.

- (Robert) I needed to read this ASCII file in Python and my usual workflow involved passing it through Pandas, for convenience. Turns out that whoever created that ASCII file, containing a table of properties of halos, decided to setup the header (and the rows) using a combination of spaces, tabs and commas. I couldn't figure out why Pandas wasn't able to parse the header until I opened the file in a reader that could show "hidden" characters. I had to use regex to make it work, which is not something that I want to touch ever again.

Modify your sky_sim.py code to make use of the logging module.

1. import the logging module and set up a logger called mylogger
2. set the default logging level to be logging.INFO
3. within each function write a DEBUG level note when you are entering/exiting the function
4. within the main function, replace the print statements with logs at the INFO level
5. Run your code and view the output
6. Change the logging level to be DEBUG and re-run your code
7. (Extension) Add a command line argument -v which will change the logging level to DEBUG

GROUP/SOLO PROJECT TASKS

1. Write at least one test for your project
2. Think about what you're going to present at "Demo Day" tomorrow
3. (optional) Implement a CLI for your project

Day 4

ONE LINE STATUS REPORT ON YESTERDAY'S PROJECT WORK

Patrick/Leonie report: we did some good teaching. Fixed more typos in course material ;)

vote-tally group: we've succeeded in getting a minimum viable product that tallies up a ranked choice vote and declares a winner, with documented functions and tests, and used github to make issues and pull requests.
Isocreams: We were able to load dataframes of isochrones and start plotting them, and we have documentation as well as initial functions. We got a test committed, which came out well, and looked at loading the isochrones. The next stage is to make sure the functions work together.

DIORAMA: We managed to read a ILLUSTRIS-TNG halo catalogue into our pipeline and make use of some of its properties to pass it to other modules. We have the luminosity function comparison.
myfftpy: wrote tests, implemented basic documentationa and CLI interface to execute basic fft on 1-D signal inputs
Space-XYZ: wrote functions, pushed all invididdual work into the joint depository, ready for integration into single run script
ECCLESIA: Away Wednesday. However have downloaded HDF module and HDF gfortran library, missed testing lecture yesterday. Downloaded test data.

REPORT YOUR MEAN AND MAX skysim EXECUTION TIMES BELOW HERE
https://adacs-australia.github.io/2023_ASA_ECR_Python_Workshop/BenchmarkingAndProfiling/index.html

RORY: real 0.240s, user 000s, sys 000s

PATRICK: Nsrc = 10,000,000, max = 19.986sec, mean = 19.457sec

Manu: Nsrc = 10,000,000, max = 22.140sec, mean = 21.757sec

Geoff: NSRC=1.0E6, max 2.720 s, mean(user) = 2.601 s, (n=6, MacOS=8 GB RAM)
NSRC=1.0E7, max = 25.093s, mean(user) = 24.907s (n=6)

Simon: N=10,000,000, max=60.191s, mean=59.24s (n=3, Surface Pro 3 (2014), i5-4300U, 8GB RAM)

Chan: NSRC = 10,000,000 max = 49.384s mean = 23.737s

Robert: NRSC= 10,000,000, mean=17.6s

Violet: 10 million stars, run 7 times, max = 16.59s, mean = 16.27 s [i7-1260P, 16 GB RAM]

WHAT’S THE SLOWEST PART OF skysim?
Rory- I'm guessing the file writter, as its using itteration, value formating changing, and file writting all in one function.

Patrick’s hypothesis: it runs slowly when you really need to get the work done *now*
Robert - creating the stars?
Gio - save star positions to file (?)
Geoff - writing the list of positions to a file and saving (Why? relying on system calls to access hardware?)

Behzad - I guess writing the list to disk too.

INTERESTING OPTIMISATION IDEAS

Rory-getting rid of itterative element of crop to circle: create an np array of radii (from ras and decs), and use boolean logic to cut out items greater than a specified radii (from ras and decs)(just checked this, and the difference it makes in runtime is marginal)

Patrick - Idea: copy the "faster" code from the course notes. Did it work? No for 1e6, "sort of" for 1e7

Violet - Generate stars within a circle to start with, so we don't have to overcompensate before cropping them.

- Ryan - fantastic idea -> you can use the Box-Muller transform to randomly place points uniformly within some circle

Geoff - Don't no about the Box-Muller transform but can you express the position of a random point as: Dec_0 + radius(rand())*cos(2pi*rand()), radius(rand()*sin(2pi*rand()) ?

GROUP PROJECTS — THE PLAN

(we have lost the Zoom, wasn't me I swear, will be back soon)
PLEASE REJOIN THE ZOOM :)

2.30pm (Melb time) – 3.30pm — Work on your group project
3.30pm – 4.00pm — Tea break
4.00pm – 4.15pm — FINAL CHANGES—make your demo look good
4.15pm – 4.55pm — Demo Day (5mins each)
 * A quick demo of something you can do with your program
 * Show us your documentation
 * Show us one of your pull requests
 * Show us some "future work" (new features or optimisation ideas), which should be a Git(hub/lab/etc) issue

Group presentation schedule (first come first served ;)):
Slot 1 vote-tally - https://github.com/vmharvey/vote-tally/tree/devel
Slot 2 isocreams
Slot 3 DIORAMA
Slot 4
Slot 5
Slot 6 SpaceXYZ

Workshop feedback survey: https://forms.gle/HJR4ERAJRDEHZiFc6

Please complete :)