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>

All content is publicly available under the Creative Commons Attribution License: <https://creativecommons.org/licenses/by/4.0/>

 ----------------------------------------------------------------------------

 FAIR in circadian Practice

## Day 3

22 participants

MODEL ORGANISMS (Arabidopis, Zebra fish etc): 5 people

Ambient-like projects (Somnofy): 9 people

HUMAN DATA (non ambient ones): 8 people

### Your coding experience,

Please, enter +1 if:

0: Not much coding experience:1+1+1+1+1+1+1+1+1

1. You use jupyter notebooks: +1+1+1

2. You sometimes write scripts (bash etc): +1+1+1+1+1+1

3. You sometimes code in R: +1+1+1+1+1+1+1+1+1

4. You sometimes code in Python:+1+1+1+1+1+1

4. You code in other language (type which):

------------------------------------------------------------------------------------------------------------------

### Lesson 10: Jupyter notebooks for reusable data analysis Exercise 1: Basics of Jupyter Notebooks

Open this Jupyter server: <http://mango.bio.ed.ac.uk/jupyter>

We will first show you how to duplicate a notebook, save it and run code.

1.     Select the notebook titled **'demo-analysis.ipynb**' and click 'Duplicate'.

2.    Confirm with **Duplicate** when asked if you are certain that you want to duplicate the notebook.

3.    A copy of the notebook has appeared with the suffix '-Copy' and a number.

4.    Change the title of the notebook from -copy number to your initials e.g. “demo-analysis\_IB”

5.    Explore the anatomy of the notebook (<https://github.com/carpentries-incubator/fair-bio-practice/blob/gh-pages/fig/10-02-jupyter_anatomy.png>)

6.    Save the notebook: Click on the disk symbol in the toolbar

7.    Run the notebook: Select the top cell of the notebook with the title (this is likely pre-selected already and will show with a light-blue bar to its left), and click “Run” in the tool bar.

8.    Click two times. What can you see?

9.    We want to run ALL the code: In the top tool bar click Cell > Run All. What can you see?

-

-

-

-

 DONE: 1+1+1+1+1+1+1+1

 ------------------------------------------------------------------------------------------------------------------

#### Exercise 2: How to add and remove content

You have duplicated a notebook and saved it under your own name. Now we will add text, remove cells and change code.

1.     Change the author name of the document to your name: Double click on the cell containing the author name and change the name.

2.     Press Run again.

3.     Add a new cell: Let us add details about the “light-conditions\_ppa.csv” file that is loaded. The “+” in the tool bar creates new cells below the currently selected cell. Thus select the cell above the code and click “+” in the toolbar.

4.     Ensure the type of the cell is Markdown and enter a description of subsequent analysis e.g.: “Loading of results following short- and long-day light exposure on arabidopsis, to investigate differences in periods between genotypes on short-days and long-days.”

5.     Press Run again.

6.     Experiment with formatting, check the existing cells how they use (# \* - to add formatting)

7.     To remove a cell, select the cell you have just created and click on the scissors icon in the toolbar. (This can be undone under Edit > Undo Delete Cells)

8.     Change colours of your box plot graph (If you do not see the box plot run all cells):

Where the code of the graph reads the comment “#setting colors manually” you can replace the HEX codes (# followed by 6-symbol code), with names of colours (e.g. blue, green…) or other HEX codes if you are familiar with them.

DONE:+1+1+1+1+1+1+1+1+1

 ------------------------------------------------------------------------------------------------------------------

#### Exercise 3: Add another analysis step

We have shown you how to manipulate text and code in Jupyter notebooks. You should be able to add data visualisation (a graph) and stats for short-day light condition including annotations yourself.

1.     Add additional cells including

a.     Titles: analysis of short days

b.     Find the code where data were selected for long days (LD) (before data were plotted), copy it and change selection for (SD)

c.     Copy code from box plot

c.     Edited code to depict graph from short-days

d.     Copy code for statistical testing of difference between genotypes on short-days

e.     Add Interpretation of results of statistical testing

DONE:+1+1+1+1+1+1+1

 ------------------------------------------------------------------------------------------------------------------

If you finished follow the suggestions for other plots in the notebook.

#### Exercise 4: Sharing of your Jupyter Notebook

You have now generated your own analysis and interpretation on top of your collaborators results and want to share this with your colleagues.

1.     Download your Notebook (ensure all code has been run) as .html and .pdf

2.    View the documents and think about why it is important to run all code before download (try Cell > All Output > Clear and download your Notebook, compare the outputs)

**Questions:**

What is the difference between running all code and clearing all run code?

Why is it important to run all code before download?

 - to make sure the code actually works before you share it

 -some information from previous script is needed

 - to show the results in the html or pdf when we are knitting

Why share notebooks in both ipynb and html

DONE:

 ------------------------------------------------------------------------------------------------------------------

 Lets meet at  14:10

#### Exercise 5: Accessibility of Jupyter Notebooks

On a scale from -2 to 2, how do you feel about the following statements (R is interchangeable with Python),

where -2 (strongly disagree), 0 no opinion to +2 strongly agree:

Type a number between -2 to 2 behind the statement:

·         making graphs for a subset of data is easier in R than in Excel:+20000+2+0+2+200+202+200

·         it is easier to filter for data in R than in Excel:+20000+00+2202+202+20

·         it is easier to generate a series of plots with similar layout in R than Excel:+2+2+20+2+0+2+202+212+2+2

·         it is easier to do large scale data processing in R than in Excel:+2+2+20+2+0+20+22+202+2+2

·         using notebooks does not require any programming knowledge:-1-2-2+2-100-10+2-20-1

·         notebooks give you a better overview of your data analysis than Excel:+2+2+120+2+20+1++22+202+2

·         notebooks links laboratory style records with data analysis:0+200+20+020+2+202

·         Jupyter is free, whilst a Microsoft Office (+Excel) suite costs $149.99, this alone is an incentive to use Jupyter:20+10000+0+2+1+1\_202

·         you need to learn R to do any data processing in notebooks:+11+2+1+1-20+0+2000

·         notebooks assures reproducible computing:+2+2+200+2+02+2+2+210

·         wrong inputs or not captures parameters are main reasons for not reproducible analysis:+22+2z=0+1+2+0+2002

DONE:+1+1+1+1+1+1+11+1+1+1+1+1+1

------------------------------------------------------------------------------------------------------------------

MEET 14:45

### Lesson 11: Version Control

#### Exercise 1: Problems with a change

Which of these issues can you relate to? (Type +1 next to each statement if you relate to it)

·         I have fifteen versions of this file - which one do I use?+1+1+1+1+1+1+1

·         I can’t remake this figure from last year.+1+1+1+1+1

·         I slightly modified my code in one place, everything stopped working.+1+1+1+1+1

·         I have several copies of the same directory because I'm worried about breaking something.+1+1+1+1+1+1+1

·         Somebody added erroneous records in a shared file with samples, cannot find who and why+1+1+1+1

·         You remember seeing a data file but cannot find it anymore: is it deleted? Moved away?+1+1+1+1+1+1+1+1+1+1

·         I tried multiple analysis and I don't remember which one I chose to generate my output data+1+1+1+1+1+1

·         I have to merge changes to our manuscript from 10 different emails with collaborators+1+1+1+1+1+1+1

·         I accidentally deleted a part of my work+1++11+1+1

·         I came to an old project and forgot where I left/saved it+1+1+1+1+1+1+1

·         I reorganized my data files, but broke the analysis pipeline and don't know how to fix it+1+1

·         I have trouble to find the source of a mistake in an experiment+1+1+1

·         My directory is polluted with a lot of unused/temporary/old folders because I'm afraid of losing something important+1+1+1+1+1+1+1+1

·         I made a lot of changes to my paper but want to bring back one paragraph+1+1+1

DONE:+1+1+1+1++11+1+1+1+1+1+1+1+1+1

 ------------------------------------------------------------------------------------------------------------------

#### Exercise 2: Manual Versioning 14:55

Which of these issues does manual versioning help to tackle?

(Type +1 next to each statement if manual versioning helps to tackle)

·         I have fifteen versions of this file and I don't know which one to use+1+1+1+1++1+1+1+1+1+1

·         I can't remake this figure from last year+1+1+1+1

·         I slightly modified my code in one place, everything stopped working+1+1+1+++111+1+1

·         I have several copies of the same directory because I'm worried about breaking something+1+1+1+1+1+1+1+1

·         Somebody added erroneous records in a shared file with samples, I cannot find who and why+1+1+1+1+1

·         I tried multiple analysis and I don't remember which one I chose to generate my output data+1+1+1+1+1+1+1+1+1+1+1

·         I have to merge changes to our manuscript from 10 different emails from collaborators+1++11+1+1+1+1+1

·         I made a lot of changes to my paper but want to bring back one paragraph+1+1+1+1+1+1+1+1

DONE:+1+1+1=+11+1+1+1+1+1+1+1+!

 ------------------------------------------------------------------------------------------------------------------

#### Exercise 3: Changelog in action

Have a look at one of the example Github repositories and how they track changes\*:

Room 1 and 2:

·         data from E.R. Ballou et al. 2020

·         <https://github.com/ewallace/pseudonuclease_evolution_2020/commits/master>

Give examples of:

·         what makes them a good changelog

* detailed, author, date, small changes are logged
* simple and clear description of the changes made

·         what could be improved

lacking motivation,

to make it easier to read/follow

 commit messages are not clear sometimes

Which are the most difficult features to replicate with manual version control?:

 include row where change was made

Room 3, 4:

·         data from I. Boehm et al. 2020

·         <https://github.com/BioRDM/nmj-pig/commits/main>

Give examples of:

·         what makes them a good changelog

 frequent and descriptive - even in the titles

 updated readme file

 repetitive subtitle

·         what could be improved

 minor changes aren't logged+1

Which are the most difficult features to replicate with manual version control?

 DONE:

 ------------------------------------------------------------------------------------------------------------------

#### Exercise 4:

Five reasons to use a version control system in research, give +1 to the two most important for you

·         Tell the story: The history of your commit messages will describe your project progress. +1

·         Travel back in time: a version control system makes it easy to compare different time points of your project smoothly. If you want to compare the stage of your project a year ago from now, it only takes one command-line of code +1+1+1+1+1+1+1+1+1+1

·         Experiment with changes: if you want to make changes in a script you can first make a “snapshot” of the project status before experimenting with changes.+1+1+1+1+1+1+1+1+1

·         Backup your work: by being able to link your local repository (folder) to a distant online host (GitHub), a version control system backs up your precious work instantly +1+1+1+1

·         Collaborate easily on projects: having a web-hosted synchronised version of your project will encourage collaboration with other researchers. Think about a colleague of yours being able to add a script to make a figure for your publication for instance. The repository will track their contribution and evidence their participation in the project +1+1+1+1+1+1+1+1+1+1

DONE: +1+1+1+1+1+1+1+1+1+1+1+1+1+1

 ------------------------------------------------------------------------------------------------------------------

### Learning GIT:

 •Software Carpentry git workshop <https://swcarpentry.github.io/git-novice/>

•Edinburgh Carpentries runs courses: <https://edcarp.github.io/>

•<https://ourcodingclub.github.io/tutorials/git/>

•Crash course on youtube: <https://youtu.be/SWYqp7iY_Tc>

•Learn git branching: <https://learngitbranching.js.org/>

•Learn git-game: <https://github.com/git-game/git-game>

 SEE You 15:40

 ------------------------------------------------------------------------------------------------------------------

#### Semantic versioning quiz

Type +1 after the correct answer:

1. Which     of the library version is the latest?

·         a) 0.12.4

·         b) 1.1.0

·         c) 1.12.3-alpha+1+1+1+1+1+1+1+1+1+1+1+1+1

·         d) 1.12.2

1. You     use library fair.2.3.1 to automatically upload your data after analysis.     You found out that there is available new version of the library     fair.4.3.1. Which situation is the most probable?

·         a) fair.4.3.1 uploads data twice as fast as fair.2.3.1

·         b) you need to reserve time to test the new version and adapt your analysis pipeline to work with +1it+1+1+1+1+1+1+1+1+1+1++11+1

·         c) both versions offer the same features and bug fixes as they have the same minor/path version 3.1

1. You     developed code that helps to plot detrended and normalized data, the last     release has version 1.2.1. You added Z-score to the available     normalization methods and fixed the spelling errors in the user messages.     Which version number should you give to the new release:

·         a) 2.0.0

·         b) 1.3.0+1+1+1+1+1+1

·         c) 1.3.2

·         d) 1.2.2+1+1+1+1+1+1+1+1+1

·         e) 2.2.1

1. Your     group releases data about studied cancer patients following semantic     versioning. The last data sets are released as 1.11, you added data from a     new patient,
2. what     version should you use:

·         a) 1.12+1++11+1+1+1+1+1+1+1+1+1+1+1+1

·         b) 2.11

·         c) 2.0

1. You     were asked to better anonymize the above cancer data (v1.11) in future     releases. Instead of data and time of patient visit you release only the     year, the hospital names have been encoded and patient age has been     obscured by random noise.
2. What     version should you use:

·         a) 1.12

·         b) 2.1

·         c) 2.0+1+1+1+1+1+1+1+1+1+1+1+1+1

DONE:+1+1++11+1+1+1++11+1+1+1+1

------------------------------------------------------------------------------------------------

 Lets 16:00

### Lesson 12: Templates for consistency

Exercise 1: Your template 16:20

FOLDER with README template, data examples, and metadata suggestions

<https://drive.google.com/drive/folders/1AHxyKHoJ2ixVWJnfvPCqRHN2BxHqM-lu?usp=sharing>

In your group

1.      Have a look at the example figures with data, imagine you have a dataset with data underling those figures

2.      Extend the README file with sections necessary to capture the essential metadata for the REUSE, lets focus on REUSE rather than reproducibility

3.      Make suggestions what each section should contain

4.      Add example content with such information.

-----------------------------------------------------------------------------------------------------------------

#### Q&A:

Do you have any questions about the topics discussed today? Please write them down here. Use +1 to upvote the ones you are interested in if someone already asked it. We will briefly discuss them before the following set of lessons.

-

-

-----------------------------------------------------------------------------------------------------------------

#### Feedback:

1.      How do you feel about the presented topics after this session (type +1 next to the statement that best describes your feeling):

•       I am more confused:

•       I have a better understanding of them now:+1+1+1+1+1+1+1+1+1+1+1+1

•       My knowledge has not changed much:

2.      How was the pace of the lesson:

•       Too fast: +1+1

•       About right:+1+1+1+1+1+1+1++11+1+1

•       Too slow:

3. From 0 to 5 how useful was the jupyter demo:4444(for beginner user)444433

4. If the lesson could be 5 minutes longer, what would you add or spend more time on:

- More time going through Jupyter+1+1+1+1+1

5. What could be improved:

- an additional bit about common ways of using Jupyter notebook for specific data analysis

-

-

6. What did you like:

- Nice intro to Jupyter, would definitely want to try it out