CMEE Masters: Computing Coursework Assessment

Assignment Objectives: To work on a series of computing/programming exercises and problems in a coherent, modular, reproducible workflow under version control.

Note that:

- *All script/code files, errors and other info mentioned below are in the weekly log/feedback files.*

- *The overall assessment will typically have significantly lesser marks than a simple weighted average of each week's points because the overall assessment is based on not just the "Computing Coursework Assessment Criteria", but also the the "Marking Criteria for Exams, Essays and Coursework". Both sets of marking criteria are in the Assessment Appendix of the online TheMulQuaBio notes and git repository.*

- *In your 1:1 post-assessment feedback session, we will discuss where you gained or lost marks, and what you could have improved further. To the extent possible, please come with questions about specific scripts based upon the overall and weekly feedback you have received. This may require you to compare your code with the solution code in many cases.*

Student's Name: Pablo Alonso

1 Specific feedback

1.1 The Good (what you did well!)

1. Found all the expected weekly directories in your parent directory.

2. Neat, clean project organization and code.

3. Your Git repo size when I checked week 7 was about 94 MB — an OK size, suggesting you did not keep unnecessary binary files under VC, and that you did not commit excessively. I assume/hope you did commit enough, and along the the way did not lose parts of your git history!

4. You had an overall readme file including a list of code files.

5. You also had a Readme within each week which was clear and succinct (but see some suggestions below).

6. Excellent job with the coding overall. Great attention to detail and re-usability. Practically no errors, and minimal warnings.

7. You did practically all extra credit Qs - fantastic.

8. Great job with the shell scripts - you made them more robust and user friendly.

9. Nice job with the Autocorrelation practical code and report. You plotted the correlation pattern, but could also have plotted the histogram/density of the permuted correlation coefficients with the observed one overlaid. Interpretation of the results was good.

10. Scripts from Weeks 4, 5 & 6 were not part of the assessment, but you kept these weeks organized - super!

1.2 The Bad (errors, missing files, etc)

1. A couple of errors in week 7

1.3 The Ugly (niggling issues like commenting, cosmetics, complexity of code, etc)

1. You had a .gitignore throughout, with meaningful exclusions specific to certain weeks. Good, but you can fine tune the exclusions further. You will likely find this useful: `https://www.gitignore.io`.

2. Your overall Readme could have had a description of objectives, conventions and usage.

3. In your weekly Readmes you could have included the language and dependencies requirements - great. Check out this resource: `https://github.com/jehna/readme-best-practices`. As you become a seasoned programmer, you will learn to make the readme file descriptions even more informative yet succinct.

4. You used somewhat non-standard single-quotes for docstrings. Not a major issue (though it threw off my assessment script), but using double quotes is recommended: `https://www.python.org/dev/peps/pep-0257/`.

5. As much as possible, it is a good idea to write scripts to be self-sufficient / modular. For example, align_seqs.py was nicely done, but it could have been written to be a full blown module that also take external inputs optionally (though I did not ask for it specifically). Compare with the solution. Also, you often put a lot of stuff (practically all the main code) in the `main` function. You should try to modularize (literally in terms of Python modules) a bit more.

6. Please do compare as many of your solutions with the ones I have given (e.g., Unix-Prac1.txt, using_os.py) as possible. There are simpler ways to solve some of them, especially the last one, and in general it will be insightful to see how the same code/solution can be written/found. In particular:

   (a) using_os.py: the script could have provided some more meaningful output to screen.
   (b) You did a great job with lc1.py, lc2.py, dictionary.py, and tuple.py, but if you compare with the solutions on the repo, you will notice that you could have make them produce better-formatted output.

2 Overall Assessment

You did an outstanding job overall. I was impressed by your efforts to understand as many details of the programming languages and coding as possible. You clearly love coding!

As this is the first time you have done programming in a heady mix of UNIX, Python, & R with a sprinkling of LaTeXand git, you did very, very well!

It was a tough set of weeks, but I believe your hard work in them has given you a great start towards further training, a quantitative masters dissertation, and ultimately a career in quantitative biology!

Provisional Mark: 90

Signed: Samraat Pawar

March 8, 2020