CMEE Masters: Computing Coursework Assessment

Note that:

All script/code errors and other info mentioned below are in the weekly assessment log files

In the weekly feedback/assessments, please compare with the solution whenever needed to see why I might have taken off points for a particular exercise/script or code file. We can then discuss these in your 1:1 post-assessment feedback session.

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

Student's Name: David Bridgwood

Overall Project workflow

Found all the expected weekly directories in your parent directory.

You had a .gitignore throughout, with meaningful exclusions specific to certain weeks – great. Later on you also included more pattern exclusions – great! You will likely find this useful: `https://github.com/github/gitignore`

You had a readme file with a list of the weeks' content, and then within each week, a readme with a detailed directory/file tree for each week , presumably using the nifty `tree` command. OK, but you may not always want/need to do it in this detail. For example, no need to list files in Sandbox or Data. After all, the Code files would presumably contain sufficient info about what sorts of data are expected as input.

Of equal or more importance is a description of what the overall project structure is and what the language and dependencies requirements are, which would be equally useful for a new user trying to understand and run your workflow. As you become a seasoned programmer, you will learn to make the readme file descriptions more informative yet succinct.

Your Git repo size when I checked week 7 was about 105.54 MB — a OK size, suggesting you did not keep unnecessary binary files under VC, and that you did not commit excessively. It could also mean that you did not commit enough, and/or somehow along the the way lost parts of your git history — but I won't check these possibilities!

WEEK 1

Found directories Data, Results Sandbox, Code

Found 11 code files: ConcatenateTwoFiles.sh, CountLines.sh, variables.sh, CompileLaTeX.sh, csvtospace.sh, FirstExample.tex, MyExampleScript.sh, FirstBiblio.bib, UnixPrac1.txt, tabtocsv.sh, boilerplate.sh

Found a FirstExample.pdf: -0.5pt

UnixPrac1.txt was fine. Each solution was described in a comment, great. You could have broken the description down into the key components of the unix command, but that's OK. Compare with the solutions, some of which you may find are more simple/compact, especially the last one!

csvtospace.sh was fine, but one addition you could have made to the script was to throw an error (with a message) if no input csv file was provided. In general, it is a good idea to add some input checks and return a meaningful message with error for utility files like this, especially in case somebody else uses it. Similar comment for ConcatenateTwoFiles.sh (running without two input files will not work), tabtocsv.sh and CompileLaTeX.sh, CountLines.sh, Variables.sh. But it's OK. No points deleted for this.

Points for this week: 99.5

WEEK 2

Found the Code, Sandbox, Data, Results directories

Found 19 code files: lc2.py, boilerplate.py, basic_csv.py, test_oaks.py, dictionary.py, debugme.py, scope.py, tuple.py, test_controlflow.py, basic_io.py, lc1.py, oaks.py, loops.py, using_name.py, cfexercises.py, align_seqs.py, sysargv.py, align_seqs_fasta.py, control_flow.py

Found no extra files; great!

lc1.py, lc2.py, dictionary.py, tuple.py were all fine. They could have given an better formatted output – Compare with the solution on the repo; -1 pts each.

align_seqs.py was nicely done. Glad you wrote it as a self-sufficient script that could also take external inputs. Also compare with the solution.

You did align_seqs_fasta.py — so +2.5 extra credit points.

All other scripts were fine.

Points for this week: 98.5

WEEK 3

Found directories Practicals, Code, Data, Results

Found 22 code files: browse.R, PP_Regress.R, apply1.R, sample.R, GPDD_maps.R, run_get_TreeHeight.sh, boilerplate.R, TreeHeight.R, PP_Lattice.R, next.R, Vectorize1.R, break.R, basic_io.R, try.R, apply2.R, get_TreeHeight.R, TAutoCorr.R, Vectorize2.R, PP_Regress_loc.R, DataWrang.R, TAutoCorr_Report.tex, control.R

The "extra" Rplots.pdf file is a R quirk — I am not actually deleting points for it, as it only happens when a R script is run externally from unix using `RScript`, and there is no easy solution for this.

Vectorize1.R was fine.

Vectorize2.R was fine, nice job — compare with the solution.

PP_Regress.R: good — also have a look at my solution.

TAutoCorr.R was fine – compare with the solution for a different approach — does your solution capture avery pair of successive years? The report: Nicely done. You could have plotted the histogram of the permuted correlation coefficients as well. Some more interpretation would have been nice!

You did the Mapping and PP_Regress_loc extra credit - +5 pts.

Points for this week: 100 pts

WEEKS 4, 5 & 6

Not assessed, but happy you kept everything organized as much as possible.

WEEK 7

Found directories Code, Data, and Results

Found a README

Found 17 code files: using_os.py, Nets.py, regexs.py, run_LV.sh, timeitme.py, DrawFW.py, TestR.R, Nets.R, LV1.py, fmr.R, re4.py, TestR.py, blackbirds.py, run_fmr_R.py, profileme.py, MyFirstJupyterNb.ipynb, LV2.py

You had a Nets_py_figure.svg: -0.5pt

using_os.py worked, but compare with the solution. The code could have provided some meaningful output to screen. -2pts

Nets.py needed the pandas package. Good job, but note that the goal was to avoid using additional packages unless absolutely necessary ;).

blackbirds.py was fine, except that it required prettytable - Note that the goal was to avoid using additional packages unless absolutely necessary ;). Do look at the solution as well.

You did the two LV* scripts with profiling. Good. Also look at the solution, which is simpler, and covers the other LV challenges as well.

Points for this week: 99.5 pts

Overall Assessment

You did an excellent job overall, including many extra credit Qs.

Practically no errors. You also went just that extra mile in many cases. You clearly like coding!

Overall, You delivered on most fronts, and if 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 well! In particular, you seem to have become quite comfortable with both Python and R – that's great!

I was impressed by your efforts to understand as many details of the programming languages and programming as possible.

It was a tough set of weeks, but I hope it gave you an inkling of why and how you would/could use Python, R, UNIX, etc together or as and when required.

Provisional Grade

A*	85
A	
B	
C	
F	

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 Appendix of the SilBioComp document.

We will discuss where gained or lost marks, and what you could have improved further in your 1:1 post-assessment feedback session. To the extent possible, please come with questions about specific scripts based upon the feedback you have received.

Signed: Samraat Pawar

January 31, 2018