

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: Hira Tanvir

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. You will likely find this useful: <https://github.com/github/gitignore> For example, .gitignore comes handy for extra L^AT_EX files.

Overall readme was there, but did not really have anything useful to guide a new user.

Within each week, you had a readme with a list of files and what they did. 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 useful for a new user trying to understand and run your workflow. As you become a (more!) 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 83.91 MB, suggesting you did not keep too many 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 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

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 especially for the last problem.

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: 100

WEEK 2

Found the Code, Sandbox, Data, Results directories

Found 18 code files: lc2.py, boilerplate.py, basic_csv.py, test_oaks.py, dictionary.py, debugme.py, scope.py, tuple.py, basic_io.py, lc1.py, oaks.py, loops.py, using_name.py, cfexercises.py, align_seqs.py, sysargv.py, control_flow.py, test_control_flow.py

Found no extra files; great!

sysargv.py gave an syntax error: -5pts

lc1.py, lc2.py, dictionary.py, tuple.py were all fine. They could have given nicer-formatted output – compare with the solution on the repo.

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

All other scripts were fine, but docstrings were missing in many many scripts.

Points for this week: 86

WEEK 3

Found directories Practicals, Code, Data, Results

Found 20 code files: run_get_TreeHeight.sh, Control.R, boilerplate.R, TreeHeight.R, PP_Lattice.R, next.R, Sample.R, Vectorize1.R, break.R, basic_io.R, try.R, apply2.R, get_TreeHeight.R, TAutoCorr.R, Vectorize2.R, DataWrang.R, TAutoCorr.tex

Found the following extra files: .Rhistory, TAutoCorrGraph.pdf, TAutoCorr.pdf, .RData: -2 pts (some of these are candidates for .gitignore)

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, but only gave one timing result on screen — compare with the solution.

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

TAutoCorr.R was fine – compare with the solutions for a different approach. The report: Nicely done (but check the compilation of TAutoCorr.tex). You could have plotted the histogram of the permuted correlation coefficients as well. The interpretation was fine. You could have reported the actual calculated p-value.

Points for this week: 98 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

Found the following extra file: .Rhistory -0.5 pt

run_LV.sh, re4.py gave errors: -10pts

using_os.py worked, but compare with the solution. The code could have provided some meaningful output to screen (like a summary of file numbers). -1pt

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 - nice output. Do look at the solution as well.

You did the LV* scripts with profiling, but had an error — look at the solution.

Docstrings were missing in a few scripts.

Points for this week: 80.5 pts

Overall Assessment

You did an good job overall.

Very few errors, but some silly ones. Review them – might seem more obvious now that you have been programming longer in Python and R.

Some of the scripts could have been more compact/short – see the solutions for how you could have found (potentially better) solutions/implementations. You could have gone just that extra mile in some cases — making sure the outputs to screen were nicely formatted, for example.

But overall, You delivered on many fronts, and if this is the first time you have done programming in a heady mix of UNIX, Python, & R with a sprinkling of L^AT_EX and git, you did very well! It was a tough set of weeks, but you kept up, and 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*	
A	
B	68
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