Syllabus

Time and place

- See UBC Calendar

Course description

Data science methods to automate the running and testing of code and analytic reports, manage data analysis software dependencies, package and deploy software for data analysis, and collaborate with others using version control.

Pre-reqs: DSCI 100 and either (a) one of CPSC 203, CPSC 210, CPEN 221 or (b) one of MATH 210, ECON 323 and one of CPSC 107, CPSC 110.

See the Faculty of Science Credit Exclusion Lists: www.calendar.ubc.ca/vancouver/index.cfm?tree=12,215,410,414

Long version: Data Science skills and tools are increasingly in demand across a large variety of disciplines. DSCI 310 aims to further students' existing data science knowledge with reproducible and trustworthy workflows in the areas of creating and deploying data analysis, reports, and software. Particular focus will be placed on teaching the skills and tools currently used in academic research and industry settings.

Without deliberate and conscious effort towards project organization, tool choice, and workflows, complex and large data science projects can quickly grow out-of-hand and become irreproducible and untrustworthy. This course will focus on reproducible and trustworthy workflows for writing computer scripts, analytic reports and data analysis pipelines, as well as packaging, automated testing and deployment of software written for data analysis. An emphasis is also placed on how to collaborate effectively with others using version control tools, such as Git and GitHub. Such workflows act to mitigate chaos and maximize transparency, reproducibility, and productivity.

While the course will be based on the use of the two leading languages in data science, Python and R, and related current tools (conda, Docker, Git, GitHub, Jupyter, etc.), the concepts and skills taught in the course aim to be discipline and tool agnostic, focussing on the importance of reproducible and trustworthy workflows for data analysis and the implications of failing to implement these when performing a data analysis.

Students who have completed this course will be able to complete complex data analysis projects with minimal technical debt – meaning that others can transparently follow how the analysis was done, reproduce the analysis for themselves if desired, and easily pickup on, and further extend the analysis in new areas. Strategies for collaboration on data science projects will also be emphasized.

Textbook

We will be using a collection of resources available online. These include:

- DSCI 310 course notes
- R packages
- Python packages

Hardware & software

Students are required to bring a laptop to both lectures and tutorials. Students who do not own a laptop, chromebook, or tablet may be able to loan a laptop from the UBC library.

Course-level learning outcomes

By the end of the course, students will be able to:

1. Defend and justify the importance of creating data science workflows that are reproducible and trustworthy and the elements that go into such a workflow (e.g., writing clear, robust, accurate and reproducible code, managing and sharing compute environments, defined collaboration strategies, etc).
2. Constructively criticize the workflows and data analysis of others in regards to its reproducibility and trustworthiness.
3. Develop a data science project (including code and non-code documents such as reports) that uses reproducible and trustworthy workflows
4. Demonstrate how to effectively share and collaborate on data science projects and software by creating robust code packages, using reproducible compute environments, and leveraging collaborative development tools.
5. Defend and justify the benefit of, and employ automated testing regimes, continuous integration and continuous deployment for managing and maintaining data science projects and packages.
6. Demonstrate strong communication, teamwork, and collaborative skills by working on a significant data science project with peers throughout the course.

Teaching team

Note that your TAs are students too; they may have class right before their office hours, and they may run a few minutes late. Please be patient!

Position	Name	Email	Office Hours	Office Location
Instructor	Tiffany Timbers	tiffany.timbers[-at-]stat.ubc.ca	Thursdays at 3pm	See Canvas
TA	Jossie Jiang	—-	TBD	See Canvas
TA	Andy Liu	—-	TBD	See Canvas

Assessment

Course breakdown

This course includes a substantial group project component. You will work in randomly assigned groups of four for the project milestones. There are also individual assignments that act as stepping stones to the project milestones. Given that collaboration is so important in data science, a portion of your final grade will be an assessment of the evidence you provide that you were an effective and productive team member. A combination of peer evaluation and GitHub history will be used to evaluate this. Your individual knowledge on the course materials (concepts and practical skills) will be evaluated on two summative assessments (midterm and final exam).

Finally, this course is delivered in a blended format, with some pre-work (video watching or reading) expected to be done before each lecture. These will be provided in the course Canvas shell. Each in class lecture session will start with iClicker cloud questions to probe your understanding of the pre-lecture material and then we will work through demonstrations and exercises in class to build off of this.

Deliverable	Grade	Learning objectives addressed
iClicker cloud	1%	1 - 6
Individual assignments	1%	1, 2, 4, 5
Project milestone 1	5%	3, 6
Project milestone 2	5%	3, 4, 6
Project milestone 3	5%	3, 4, 5, 6
Final project	5%	3, 4, 5, 6

Deliverable	Grade	Learning objectives addressed
Peer review	2.5%	2
Teamwork	5%	6
GitHub username quiz	0.5%	NA
Mid-term Exam	20%	1, 2, part of 4
Final Exam (**see note below**)	50%	1, 2, 4, 5

- **You must pass the final exam to pass the course.**
- We will be using iClicker cloud in the lectures regularly. You will be graded on both participation and performance (50% each). The three lowest iClicker marks will not be counted.

Schedule at a glance

Week	Date	Topic	Assessments due	Notes
1	2023/01/08	How do reproducible and trustworthy workflows impact data analysis?		Start working on your installation instructions
2	2023/01/15	Version control for transparency and collaboration	Individual assignment 1 & GitHub username quiz	
3	2023/01/22	Integrated development environments, filenames and data science project organization	Individual assignment 2	
4	2023/01/29	Managing dependencies using virtual environments		Team assignment for group projects & drafting of team work contract
5	2023/02/05	Managing dependencies using containerization	Individual assignment 3	
6	2023/02/12	Non-interactive scripts and reproducible reports	Mid-term exam	
7	2023/02/19	Reading Break		
8	2023/02/26	Data analysis pipelines	Milestone 1	
9	2023/03/04	Introduction to testing code for data science	Individual assignment 4	

Week	Date	Topic	Assessments due	Notes
10	2023/03/11	Advanced version control workflows	Milestone 2	
11	2023/03/18	Packaging and project work session		
12	2023/03/25	Project work session & documenting code	Milestone 3	
13	2023/04/01	Automated testing and continuous integration	Individual assignment 5 & Peer review	
14	2023/04/06	Deploying and publishing packages, copyright and licenses	Final project & Team work reflection	

Assessment schedule

In general, assignments will be due 11:59 PM on Saturdays. However, in the final week of classes, all assignments need to be submitted by the final day of classes, thus we have two alternative due dates that week.

Assessment	Description	Due date	Due Week
Individual assignment 1	Setting up your computer	2023/01/20 23:59	2
GitHub username quiz		2023/01/20 23:59	2
Individual assignment 2	Version control practice	2023/01/27 23:59	3
Individual assignment 3	Dockerfile practice	2023/02/10 23:59	5
Mid-term exam	The midterm is a summative assessment: https://www.cmu.edu/teaching/assessment/basics/formativesummative.html	2023/02/12-2023/02/16 (Exact date/time TBD)	6
Milestone 1	Question, data & rough draft of analysis in one monolithic literate code document, reproducible environment (`full.ipynb`, `Dockerfile`, `docker-compose.yml`)	2023/03/02 23:59	8
Individual assignment 4	Reproducible reports practice	2023/03/09 23:59	9

Assessment	Description	Due date	Due Week
Milestone 2	literate code document broken into scripts and a report & data analysis pipeline to stitch everything together	2023/03/16 23:59	10
Milestone 3	functions abstracted to a file/module & tests, function documentation	2023/03/30 23:59	12
Peer review	review of another group's project	2023/04/06 23:59	13
Individual assignment 5	Packaging practice	2023/04/06 23:59	13
Final project	package & CI (the full monty package - including docs)	2023/04/11 23:59	14
Team work	Reflection of how the group worked together, as well as individual performance	2023/04/12 23:59	14
Final exam (**You must pass the final exam to pass the course.**)	The Final Exam will include all the material covered in all the components of the course. This is a summative assessment: https://www.cmu.edu/teaching/assessment/basics/formativesummative.html	TBD	

Policies

Code of Conduct

All participants in our course and communications are expected to show respect and courtesy to others. To creating a friendly and respectful place for learning, teaching and contributing, you are expected to read and follow the DSCI 310 Code of Conduct.

Late registration

Students who register for the class late have 1 week from their registration date on Canvas to complete all prior assignments.

Late assignments / mid-term exam absence

Students **must be present** at the invigilation venue (in class, on Zoom, examination centre, etc) to take the mid-term exam; otherwise they will be considered to have missed the mid-term exam and will be assigned a grade of zero.

Students who will miss the mid-term exam **must provide a self-declaration prior to the mid-term exam** and make arrangements (e.g., schedule an oral make-up mid-term exam) with the Instructor. Failing to present a declaration within a reasonable timeframe before the mid-term exam will result in a grade of zero.

A late submission is defined as any work submitted after the deadline. For a late submission, the student will receive a 75% scaling of their grade for the first occurrence, 50% scaling of their grade for the second occurrence, and will receive a grade of 0 for subsequent occurrences.

Students who miss an assignment or quiz can request an academic concession. From the UBC Senate policy on academic concession, grounds for academic concession can be illness, conflicting responsibilities, or compassionate grounds. Examples of compassionate grounds, from the above policy, include "a traumatic event experienced by the student, a family member, or a close friend; an act of sexual assault or other sexual misconduct experienced by the student, a family member, or a close friend; a death in the family or of a close friend."

To request an academic concession, students should immediately email a completed and signed academic concession form to the course Instructor. Upon receiving the form, the Instructor will make a decision about how to proceed. Failure to present valid documentation may result in a failing grade.

Re-grading

If you have concerns about the way your work was graded, please contact the TA who graded it within one week of having the grade returned to you through Piazza. After this one-week window, we may deny your request for re-evaluation. Also, please keep in mind that your grade may go up or down as a result of re-grading.

Missed final exam

Students who miss the final quiz must report to their faculty advising office within 72 hours of the missed exam, and must supply supporting documentation. Only your faculty advising office can grant deferred standing in a course. You must also notify your instructor prior to (if possible) or immediately after the exam. Your instructor will let you know when you are expected to write your deferred exam. Deferred exams will ONLY be provided to students who have applied for and received deferred standing from their faculty.

Academic concession policy

Please see UBC's concession policy for detailed information on dealing with missed coursework, quizzes, and exams under circumstances of an acute and unanticipated nature.

Academic integrity

The academic enterprise is founded on honesty, civility, and integrity. As members of this enterprise, all students are expected to know, understand, and follow the codes of conduct regarding academic integrity. At the most basic level, this means submitting only original work done by you and acknowledging all sources of information or ideas and attributing them to others as required. This also means you should not cheat, copy, or mislead others about what is your work. Violations of academic integrity (i.e., misconduct) lead to the breakdown of the academic enterprise, and therefore serious consequences arise and harsh sanctions are imposed. For example, incidences of plagiarism or cheating may result in a mark of zero on the assignment or exam and more serious consequences may apply if the matter is referred to the President's Advisory Committee on Student Discipline. Careful records are kept in order to monitor and prevent recurrences.

A more detailed description of academic integrity, including the University's policies and procedures, may be found in the Academic Calendar at http://calendar.ubc.ca/vancouver/index.cfm?tree=3,54,111,0.

Plagiarism

Students must correctly cite any code or text that has been authored by someone else or by the student themselves for other assignments. Cases of plagiarism may include, but are not limited to:

- the reproduction (copying and pasting) of code or text with none or minimal reformatting (e.g., changing the name of the variables)
- the translation of an algorithm or a script from a language to another
- the generation of code by automatic code-generation software

An "adequate acknowledgement" requires a detailed identification of the (parts of the) code or text reused and a full citation of the original source code that has been reused.

The above attribution policy applies only to assignments. **No code or text may be copied (with or without attribution) from any source during a quiz or exam. Answers must always be in your own words. At a minimum, copying will result in a grade of 0 for the related question.**

Repeated plagiarism of any form could result in larger penalties, including failure of the course.

Attribution

Parts of this syllabus (particularly the policies) have been copied and derived from the UBC MDS Policies.