DSCI 524: Collaborative Software Development

How to exploit practices from collaborative software development techniques in data scientific workflows. Appropriate use of the software life cycle, unit testing / continuous integration, and packaging for use by others.

Learning Outcomes

By the end of the course, students are expected to be able to:

- complete a project that demonstrates appropriate use of collaborative software development processes and tools, including:
  - packaging code for use by others, including the specification of dependencies/requirements
  - using advanced distributed version control workflows and features (e.g., GitHub flow, master branch protection, code reviews, project boards and issue tracking) to effectively manage a multi-person project
  - writing comprehensive test suites
  - carrying out continuous integration via GitHub Actions
  - deploying packaged software using semantic versioning
  - writing clear and helpful documentation
  - selecting software licenses that best suit the project
  - critically evaluating work of others and providing constructive feedback

See here for lecture-by-lecture learning objectives

Teaching Team

Section	Position	Name	Slack Handle
001	Lecture & Lab Instructor	Tiffany Timbers	`@tiff`
001	Teaching Assistant	Rubia Guerra	`@Rubia (TA)`
001	Teaching Assistant	Tony Liang	`@Tony Liang (TA)`
001	Teaching Assistant	Mohammad Mahdi Asmae	`@Mahdi (TA)`
001	Teaching Assistant	Sky Sheng	`@Sky Sheng (TA)`
002	Lecture & Lab Instructor	Daniel Chen	`@Daniel (Instructor)`
002	Teaching Assistant	Bahar Khodabakhshian	`@Bahar Khodabakhshian`
002	Teaching Assistant	Ailar Mahdizadeh	`@Ailar Mahdizadeh`
002	Teaching Assistant	Nima Hashemi	`@Nima Hashemi`

Section	Position	Name	Slack Handle
002	Teaching Assistant	Eric Lee	`@Eric`

Deliverables

This is a project-based course where you will work collaboratively in groups to develop a Python software package. In lecture and lab, we'll work on the skills and concepts that you need to learn to complete the project. You will work in randomly assigned groups of four for the project milestones. There are also some individual weekly assignments that act as stepping stones to the project milestones. Finally, because collaboration is so important in data science, we will be using a team work multiplier to adjust individual grades. A combination of peer evaluation and GitHub history will be used to evaluate this.

Assessment	weight	Deadline
iClicker	3%	Every lecture
Milestone 1: Coming up with a topic, and setting up package structure & function specification and documentation	15%	2024-01-11 18:00 PT
Individual assignment - Create your own toy Python and R packages (graded for completion)	5%	2024-01-11 18:00 PT
Milestone 2: writing the code and tests for your package	30%	2024-01-18 18:00 PT
Milestone 3: package documentation	15%	2024-01-26 11:59 PT
Milestone 4: continuous integration (and deployment in the case of Python) and improve your package after peer review	25%	**2024-02-03 23:59 PT**
Individual Peer review	5%	**2024-01-30 23:59 PT**
Team work reflection	2%	**2024-02-04 23:59 PT**
Team work multiplier	final grade * multiplier	NA

Lecture Schedule - under construction!

Lecture notes: https://ubc-dsci.github.io/reproducible-and-trustworthy-workflows-for-data-science/

Lecture	Topic	Lecture notes	Additional readings and resources
1	Introduction to R & Python packages	Introduction to R & Python packages	Package structure and state (Python) & Package structure and state (R)
2	Introduction to collaborative software development	Project management using GitHub	GitHub project boards, GitHub project milestones, GitHub pull request reviews, About protected branches
3	PyTest	Pytest	Testing in Python, `pytest` docs
4	Testing coverage	Code coverage	TBD
5	Continuous integration (CI) via GitHub Actions	Continuous Integration CI+Python	GitHub Actions docs, Github actions with R, Continuous integration (Python)
6	Continuous deployment, versioning & package documentation	CD: Python Documentation	Semantic versioning, R: Object Documentation, R: Vignettes, `pkgdown` , Documentation (Python)
7	Publishing your package on the package indices (CRAN & PyPI), and peer review of data science R & Python packages	Package Publishing: Python	rOpenSci Packages: Development, Maintenance, and Peer Review, The pyOpenSci Developer Guide
8	Copyright & Licenses	Copyright and Licenses	Who owns the code?

Textbooks:

- Python packages by Tomas Beuzen and Tiffany Timbers
- R packages by Hadley Wickham and Jenny Bryan

Class Schedule & office hours

See calendar.

Policies

Students are expected to attend 100% of their scheduled lab hours and attendance will be taken at the beginning of each lab session. Students must make formal requests for academic concession in the event that they cannot attend a lab session by filling out this academic concession form and sending it to the course instructor. Applying

for academic concession does not promise the granting of academic concession. Each application will be evaluated in the context of UBC's Academic concession policy. Students missing more than 25% of the total scheduled lab hours for a course without having received an academic concession will receive a possible maximum course grade of 50% and an "F" standing for the course on the transcript.

For all other policies, please see the general MDS policies.

Authors: Tiffany A. Timbers, Varada Koltahar, Florencia D'Andrea, Daniel Chen

License: