Improving students' programming skills through Collaborative Scientific Python

Julien-Pooya Weihs[1,2], Daniel Oddmund Lid[1]
[1]Geophysical Institute, University of Bergen | [2]Bjerknes Centre for Climate Research

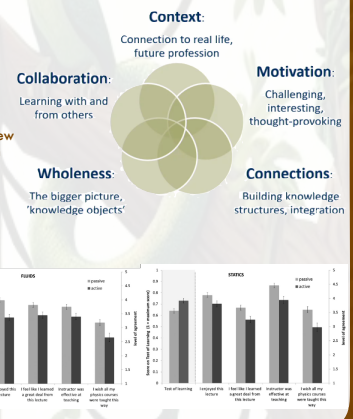
Context

- **Programming education poses challenges** to both students (unclear uses and benefits, lack of training) and staff (goals not clearly defined, complex to teach).
- **Git** is cloud-based *Version Control System* tracking all changes to a file/project, records **who** made **what** change **when** and **why**. Common Git platforms are *GitHub* and *GitLab*.
- **Visual Studio Code** is **free & open-source** code editor-interpreter-compiler synchronising with Git. Supports 100+ programming languages, more than 45000 extensions, most popular coding environment (74% of 71000 respondents in 2022).
- **Git + Visual Studio Code** framework supports thorough collaboration, gives transparency about workflow and individual contributions, keeps shared projects always up to date to everyone.
- Research on using **Git in programming** education [1-4]:

Advantages	Limitations
Collaboration: improves communication and teamwork skills	**Learning curve** for beginners unfamiliar with Git-environment → unthinkable to have degree in science without coding skills → develop shared base of knowledge with other users
Version control: track changes and revert to previous versions if necessary	
Feedback: *pull-request* feature is forum for discussion between users	May not be suitable for **grading assignment** → grade student work on mitt.uib
Transparency of activity: holds users accountable for their contributions	
Opportunity: students develop technical skills at industry-standard	Some may be **uncomfortable sharing** work → possibility to use private repositories at any time

Learning

- **Meaningful learning** [5], where motivation can be split in 2:
  - extrinsic (relating to external drivers, achieving a certain goal).
  - intrinsic (genuine interest, enjoyable).
- **Learning to learn** [6], where goal is not content but development as a learner: capable of finding new approaches and information, and apply these in an effective manner:
  - online resources, communication with peers, creative skills.
  - self-determined learners able to be effective and seek out knowledge.
- Recommendations of *Discipline-Based Educational Research* [7-8]:
  - encourage student **participation**.
  - involve students in **collaborative** activities.
  - supplement instruction with **tutorials**.



Course

- Over **5 days during 2 weeks** in Spring 2024 (17:00-20:00) with free pizza & snacks
- Open to all students from all programs at the Geophysical Institute at UiB (**14 attendees**)
- Activities using **real-life data** and focusing on case-studies in **meteorology** and **oceanography**
- Main objectives:
  1. reach working fluency for **collaborative projects** with **Git**
     - download files, *fork/branch* projects, submit *pull/merge* requests.
     - *commit-push* tracked changes, *fetch-pull* new versions.
     - run and manage projects with peers and collaborators.
  2. have an understanding and practice of basic **scientific Python**
     - discover and use *Pandas*, *Matplotlib*, *Numpy*, *Cartopy*, *Scipy*, etc.
     - be able to read-in, pre-process, analyse and visualise data.
     - groom and utilise these skills all along studies and future career.
- No exam but **final group presentation** to climate expert from Geophysical Institute

Structure

Day 1 – **Introduction** (+ pre-course survey)
- theory and motivation behind [teaching and learning theory] & [Git for programming].
- installation of *Python*, *Visual Studio Code* and *Git* infrastructure.

Day 2 – **Scientific Python**
- demonstration of use cases by invited researcher (in oceanography).
- practice and exercises.
- get everyone past the minimal required experience level.

Days 3 & 4 – **Git & Collaboration**
- basic Git commands.
- scientific group project.
- collaborate in teams and between different teams.

Day 5 – **Reflections** (+ post-course survey)
- present and discuss results with a climate expert.

Resources

All material can be shared on request, please get in touch

- Course slides for 5 lecture days.
- Real-life scientific meteorology and oceanography data.
- Python exercises and possible solutions.
- Pre- and post-course survey instruments.
- Survey data processing suggestion codes.
- Structure of GitLab repository and main routines.
- Link to GitHub repository with all resources.

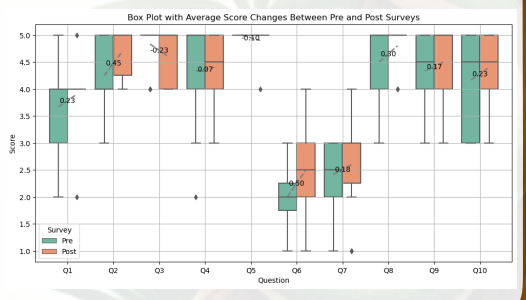
Try this out in your own department next semester!

Educational Research (excerpt)

What are the students' perception of the course on their programming learning?

Survey questions [9-10] code for:
- Intrinsic motivation (Q1 - Q2) **+0.34**
- Motivation (Q3) **-0.23**
- Extrinsic motivation (Q4 - Q5) **-0.02**
- Self-efficacy (Q6 - Q7 - Q8) **+0.33**
- Belonging (Q9 - Q10) **+0.20**

Dataset very limited (12 respondents), but:
- Increased intrinsic motivation can enhance engagement and persistence in learning.
- Greater self-efficacy can lead to increased effort and resilience in facing programming.
- Higher belonging makes students more likely to engage in collaborative activities and foster a supportive environment.



Acknowledgements
This project is run within the HKDir-funded *CoCreatingGFI* project (see QR on the right) at the Geophysical Institute of the University of Bergen.
The authors warmly thank all participating students for their great energy and enthusiasm throughout the course.

References
[1] Zagalsky et al. "The emergence of GitHub as a collaborative platform for Education." (2015)
[2] Kertész. "Using GitHub in the classroom - a collaborative learning experience." (2015)
[3] Feliciano et al. "Student experiences using GitHub in software engineering courses: a case study." (2016)
[4] Glazunova et al. "The effectiveness of GitHub cloud services for implementing a Programming training project: students' point of view." (2021)
[5] Weurlander et al. "Meaningful learning: students' perceptions of a new form of case seminar in pathology." (2009)
[6] Blaschke. "Heutagogy and lifelong learning: a review of heutagogical practice and self-determined learning." (2012)
[7] Singer. "Discipline-Based Education Research: understanding and improving learning in undergraduate Science and Engineering." (2012)
[8] Deslauriers et al. "Measuring actual learning versus feeling of learning in response to being actively engaged in the classroom." (2019)
[9] Boljat et al. "Students' attitudes towards Programming after the first year of implementing a new Informatics curriculum in the elementary school." (2019)
[10] Van Staveren "Integrating Python into a Physical Chemistry lab." (2022)