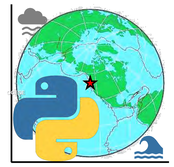
Department of Earth, Ocean and Atmospheric Sciences

TLEF project summary: Embedding Opensource Computational Tools into the Quantitative Earth Science Specializations

Tara Ivanochko (PI), Phil Austin (Lead), Francis Jones (Lecturer / STLF) https://eoas-ubc.github.io *TLEF showcase, May 2023*



1. Project Goals

Two top goals: *Develop open-source computing capacity…*

1. *to increase quantitative learning in EOAS courses, &*
2. *enhance computing & math abilities of EOAS students.*

Five goals in support of the top two:

3. Develop & test sustainable cloud computing facilities
4. Produce documentation, resources, guidelines, tutorials
5. Support faculty adopt consistent opensource practices
6. Support the FoS minor in Data Science and DSCI 100
7. Introduce open education materials & practices

2. Project Contributions

1. **Python & Jupyter Notebooks** (JNBs), new or replacing MatLab.
2. **Dashboards**: Interactive apps for learning or demonstrations.
3. **Data** gathered about students' and instructors' experiences.
4. **Consulting** re. content, learning, pedagogy, or logistics.
5. **Resources**: Guidelines for Python, JNBs, GitHub, dashboards, etc.
6. **Faculty ProD**: COVID → mainly 1-on-1 consulting.
7. **Dissemination**: 6 UBC events; 5 events beyond UBC.

3. Impacts: Courses & Deliverables

20 courses participated; ~2900 students affected, 2020 - 2023.

1. Opensource computing to help expose more students to quantitative Earth Sciences

Course	1. jnb	2. dashb	3.data	4. consult	5. resource	6. FProD	7. dissem
ENVR 300		3	2				
EOSC 112		1	2	2			
EOSC 114				1			
EOSC 116				1	1		
EOSC 116, 326		2					
EOSC 310		1		1			
EOSC 323		1					
EOSC 325		3	4	3			
EOSC 340		1		1			
EOSC 372		1	2	1	1		
EOSC 373				1			
EOSC 429		1					
EOSC 442	y		1	2	1		

Three types of changes to courses & Resources . . .

2. Opensource computing to enhance computing / math abilities of EOAS undergrads

Course	1. jnb	2. dashb	3.data	4. consult	5. resource	6. FProD	7. dissem
ATSC 301	y		2		1		
DSCI 100	y		2	1	4		
EOSC 211	y		12	5	2		
EOSC 350	y		1	1			
EOSC 354	y	1	2	1	2		
EOSC 410/5	y		1	1	1		
EOSC 471	y						

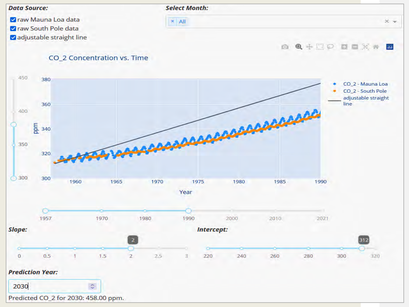
3. Opensource computing capacity in EOAS: Contributions not related to specific courses.

Course	1. jnb	2. dashb	3.data	4. consult	5. resource	6. FProD	7. dissem
Event: eoas						3	1
Event: outside							4
Event: UBC							4
Hub for dashbds					1		
list: Dep't computing needs					1		
project accomplishments					1		
project repository					1		
Project website					1		

4. Engaging with data & concepts (all students)

Targeting goal 1: Build & deploy interactive dashboards using opensource computing techniques.

E.g. → Explore CO2 vs. time at Hawaii & Antarctica, then learn pros & cons of *linear* modelling.



Development

- Opensource code; design & delivery tactics; corresponding pedagogy.
- Team: Instructor, coordinator, student programmer.
- Student feedback gathered with first deployment.
- Time to build: ~40hrs for a known learning task.

Impacts; Instructors

- Need to design interactive, explorative learning tasks.
- Provide feedback on design & deployment.
- Deployment is "transparent" to instructing team.

Impacts; Students

- Students focus on concepts, not data wrangling.
- Engaging with concepts & real data is inspiring.
- Exploring concepts and data sets = "*active learning*".
- Feedback has been positive:

Eg: "*I like how integrated it is and how it connects all the topics dealt with in this course so far*".

Costs & lessons learned

Design/build:

- Instructor(s) & developers need **time** to collaborate.
- **Coding skills** needed are "strong" undergrad level.
- Development is **iterative**.

Deployment and learning

- **Server** needs admin-level computing skills or staff.
- **Students** need time to *explore*; groups work best.
- After exploring, *then* assign concrete tasks.

5. Computing for EOAS students

Targeting goal 2: Transform courses to Python & opensource practices. *(Challenges/costs depend on scope of changes.)*

4. Course conversions from original code environment to Python using Jupyter Notebooks.

OCESE contributions	DSCI 100	EOSC 211	EOSC 442	EOSC 354	EOSC 410/510	EOSC 471
Original	R	Matlab	MatLab	MatLab	MatLab	MatLab
text rewrite	y					
text adopt OER		y				
class materials	y	y				
labs / assigs	y	y	y	y	y	y
auto grading	y	y				
hubs	y	y	y			y
local installs		y		y	y	

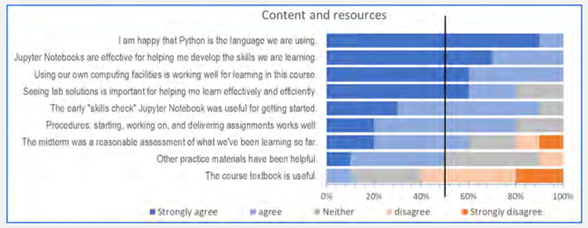
Straightforward course transformations

EOSC442, EOSC354, EOSC410, EOSC471

- Convert all labs or assignments to Python.
- Establish workflows for students & instructors.
- Provide "boot strapping" resources.

Student reactions

- Lower enrolments of 3rd or 4th year students.
- Largely positive; "Math is '*hard*', but Python is '*good*'".
- E.g.: some feedback from EOSC354



Complex / challenging course transformations

EOSC 211, DSCI 100

- Intro courses - hence complete **rebuilds**.
- "Expensive" - many contributors; many hours.
- **Opensource textbooks**.
- Reliable hub **essential** for ~100 students.
  - Initially, hubs were unstable.
- 1/3 students preferred **laptops** to hubs.
- **Auto-grading** can work but is "hard".
- **Support** for students & instructors is essential.
- **Despite challenges**, student **discussions** still focused on course content rather than logistics or hubs.

6. Services & resources delivered

✓**Adjustments to 20 courses**: Tables 1,2,3 below.

✓ **Jupyter notebooks & opensource tactics:** Table 4.

✓ **Dashboards as Open Ed Resources:**

- 17 apps built, 11 courses now use them
- Focuses students on concepts & real data .
- Interest is growing, but slowly (faculty polling)

✓ **Consulting for instructors/TAs** (pedagogy & logistics).

- Course adjustments involved 1-1 support:
  - STLF → Instructor &/or TA
  - TA → instructor
- Workflows for opensource tools & tactics.

✓ **Department and institution**

- J-Hubs reliability & scalability: Opensource community knows how.
- Dashboard servers: in-house, docs being prepared.

✓ **Resources - https://eoas-ubc.github.io**

- Guidelines for students & instructors
- Assessment management
  - Auto-grading for EOSC 211 & DSCI 100
  - Questions: Text/Markdown ←→ Canvas
- Open Education Resources (OERs)

7. Take Home Messages

✓ **Transforming courses to Python:**

- Beginner's courses are **challenging & "costly"**:
- Others **straight forward** if converting assigs only.
- **Hubs** for JNBs must be "bomb-proof" and scalable.

✓ **Dashboards**

- Are attractive & "low-stakes" for most instructors.
- Can be straight forward or challenging to design/build.
- Need in-house server technology and skills to host.

✓ **Educational expertise** keeps the focus on learning and pedagogical best practices.

✓ **Future interest?**

- Of 78 courses in EOAS, 32 use or **hope to** use opensource computing resources.
- Results of polling EOAS faculty:
  - Few who are not already using computing want to.
  - inspiration is needed; showcase successful examples, show details of costs & benefits.

Acknowledgements

*Many thanks for financial support provided by UBC Vancouver students via the **Teaching and Learning Enhancement Fund**, and the UBC **Work Learn Program** for helping employ student contributors.*

Contributors 2020 - 2023:

20 Faculty: T. Ivanochko (P.I.), P. Austin (Lead), F. Jones (STLF), C. Johnson, V. Radic, A. Ameli, S. Waterhouse, M. Maldonado, K. Orians, S. Sutherland, R. Beckie, S. Allen, A. Orsi, M. Bostock, L. Heagey, L. Porritt, K. Hodge, M. Lipsen, T. Timbers (Stats), T. Campbell (Stats)
11 Graduate students: R. Merrill, N. Dahiya, W. Ye, Y. Egorovo, Y. Kuzmenko, A. Loeppky, J. McFarlane, C. Rodell, F. Rossmann, Y. Su, H. Umashankar.
6 Undergraduates: J. Byer, B. Chang, M. Colclough, D. Platonov, C. Zhang, I. Sadeh,