Social and Interactive Learning at Community Colleges: Investigating Data-driven Feedback for Improving Student Outcomes with Project Based Learning (SAIL-CC)

Project-Based Learning and STEM Persistence in Community Colleges

Christopher Bogart
Carnegie Mellon University
cbogart@cs.cmu.edu

Marshall An
Carnegie Mellon University
haokanga@cs.cmu.edu

Huy Nguyen
Carnegie Mellon University
hn1@andrew.cmu.edu

Adam Zhang
Carnegie Mellon University
yufanz@cs.cmu.edu

Jaromir Savelka
Carnegie Mellon University
jsavelka@cs.cmu.edu

Can Kultur
Carnegie Mellon University
ckultur@cs.cmu.edu

Majd Sakr
Carnegie Mellon University
msakr@cs.cmu.edu

Social And Interactive Learning

The Sail() Platform

Sail() is an online learning platform that provides 2-year and 4-year college and university instructors with technology courses created at Carnegie Mellon University that are project-based, collaborative, and use real-time feedback to teach at your own institution.

A CS labor shortage: community colleges can help

Over 350,000 IT jobs will be added in the US in IT, per year through 2034 [1]
Only 65,000 IT undergraduate per year will graduate in CS/IT in the US [2]
31% of CS/IT graduates are from community colleges [3] but they drop out or transfer to other fields at a high rate, especially Black and Latin students [4].

Why Project-based Learning (PBL)?

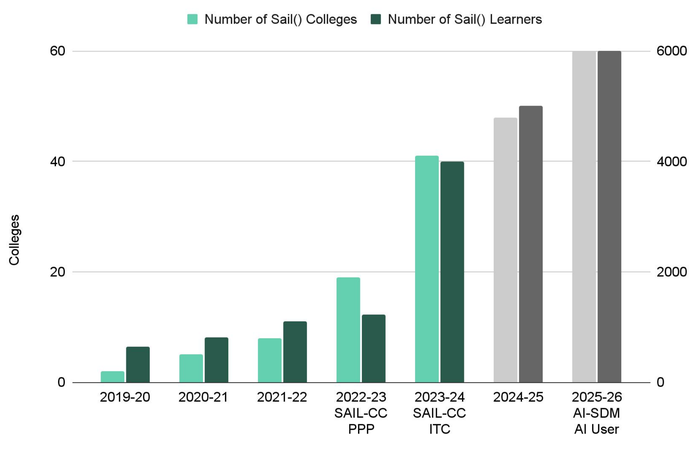
Project-based learning has emerged as a promising educational strategy to engage and retain students in STEM fields, by not only training them in real-world skills, but in imparting the confidence they need to persist in these majors and careers. However, PBL can be resource-intensive for both instructors and students, so there is a need for tools to empower community college instructors to teach in this way.

Making PBL feasible at community colleges

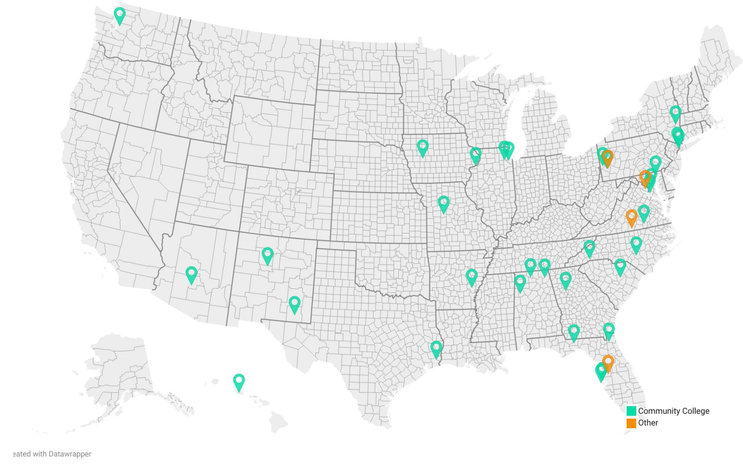
PBL is most successful face-to-face (see below), but colleges cannot always afford to do this. Face-to-face instructors employ strategies such as explicit instruction in how to break up large problems, practical debugging strategies, and individual tutoring for struggling students. These insights are motivating our emerging research into ways of providing scaffolding and a sense of belonging even in institutions that cannot afford to provide on-premises instruction.

[1] CompTIA, 2020. [Online]. Available: https://www.cyberstates.org
[2] P. Loyalka, O. Liu, and I. Chirikov, "Study shows us computer science graduates have the edge. university world news." 2019, April 6. [Online]. Available: https: //www.universityworldnews.com/post.php?story=20190404113158365
[3] A. D'Amico, "Enrollment and Employees in Postsecondary Institutions, Fall 2017; and Financial Statistics and Academic Libraries, Fiscal Year 2017."
[4] C. Riegle-Crumb, B. King, and Y. Irizarry, "Does STEM Stand Out? Examining Racial/Ethnic Gaps in Persistence Across Postsecondary Fields," Educational Researcher, vol. 48, no. 3, pp. 133–144, Apr. 2019,.

A rapidly growing community



of more than 40 community colleges



PBL courses created and offered through this grant

Practical Programming with Python (PPP)

	Concepts	Projects
	Learning with Sail()	development environment, submission, analyze feedback, modify Python code
FUNDAMENTALS	Types, Variables, Functions	expressions, statements, built-in and user defined functions
FUNDAMENTALS	Iteration, Conditionals, Strings, I/O	for and while loops, if statements, relational and logical operators, string manipulation, read and write files
FUNDAMENTALS	Lists, Sets, Tuples, Dictionaries	insert, access, modify and delete elements in various data structures
FUNDAMENTALS	Classes, Objects, Modules	OOP, user-defined classes and modules
	Top-down Design, Testing, Debugging, Style	read code, TDD, poor vs high-quality code, identify application requirements, design
APPLICATIONS	Files and Data Stores	csv/tsv, json, XML, SQL, NoSQL, and related Python libraries
APPLICATIONS	Web Data	HTML, CSS, web scraping, public APIs, office document processing
APPLICATIONS	Data Analysis	pandas, data visualization, regular expressions

Introduction to Computing (ITC)

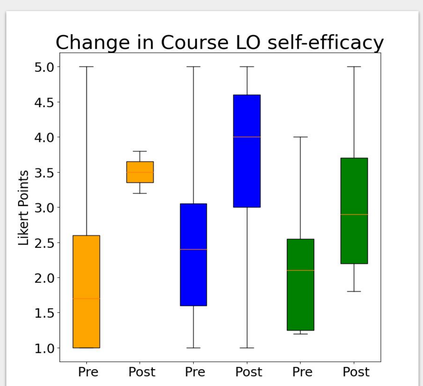
	Concepts	Projects
BASICS	Computing Devices	Computing Devices, Capabilities, Performance, and Cost
BASICS	Operating Systems and Applications	Operating Systems, Applications, Installation, Operation, Maintenance
DATA	Data Representation, Formats, Filesystems	Binary and Text Files, Structured and Unstructured Data Formats, Compression
DATA	Database Systems	Load, Access, and Update Data in a Database System
CLOUD/NET	Networking and Internet	Networking Devices Configuration
CLOUD/NET	Cloud Computing	Cloud Resource Provisioning and Access
CLOUD/NET	Cybersecurity	Phishing, Identity Theft, Wifi Scanner
CODE	Programming and Software Development	Run Code, Troubleshoot a Malfunctioning System

What factors lead to persistence in the Python course?

Community college instructors were free to use the course however they wanted, e.g.:
- Course modality (online or face-to-face; self-paced or with deadlines)
- Which course modules to use,
- How to weigh them as grades
- What scaffolding or other materials to include

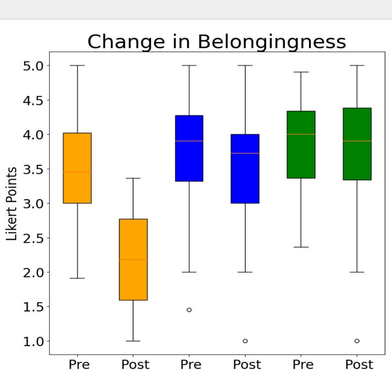
The largest predictor of student persistence (defined as the ordinal number of the *last* assigned project they attempted) was course modality. Teaching face-to-face, with deadlines, resulted in the greatest persistence.

Students in all modalities significantly improved self-efficacy; belongingness dropped in online, self-paced courses.



Sample question:

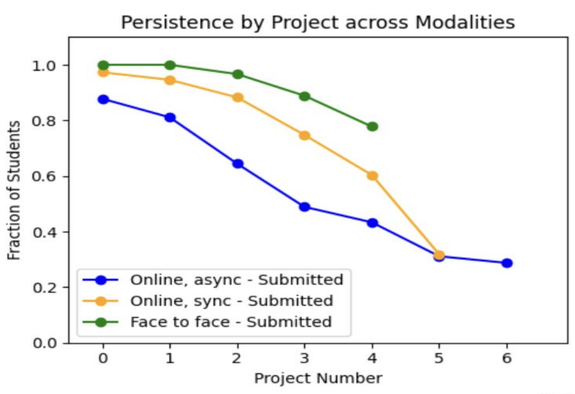
"I could write a small computer program that reads data from several files and produces a summary report stored in a new file."

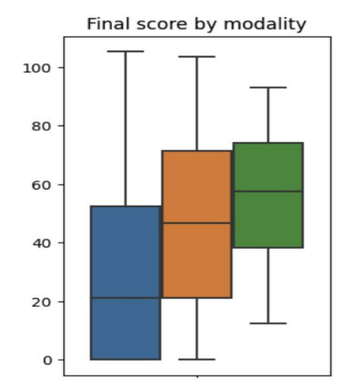


Sample question:

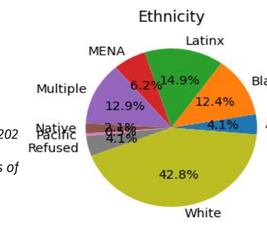
"Other students in this program notice when I am good at something"

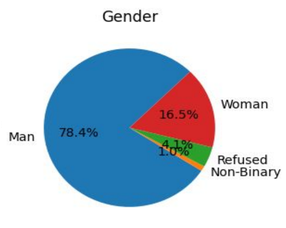
Learning Modality and Persistence





In-person: 140 students in 11 sections
Online (Sync): 205 students in 10 sections
Online (Async): 223 students in 14 sections





Bogart, C., An, M., Keylor, E., Singh, P., Savelka, J. and Sakr, M., 202 March. What Factors Influence Persistence in Project-based Programming Courses at Community Colleges?. In Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1 (pp. 116-122).

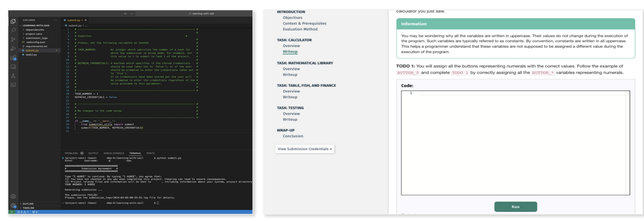
Scaffolding towards a real coding environment

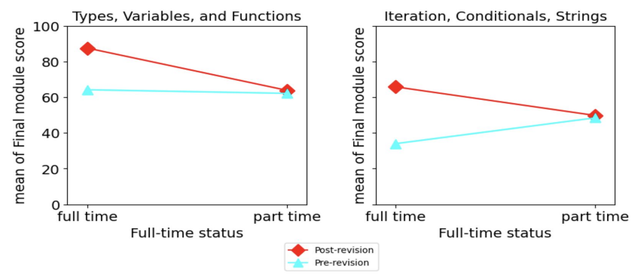
In the first year of the Python course, we insisted students us a real Integrated Development Environment (IDE) for all their submissions. Through interviews and focus groups with instructors we learned that students found this frustrating and demotivating.

In the second year, we introduced an in-browser window for running and submitting projects, for early projects, then introduced the IDE for the latter part of the course. Students performed better on both the in-browser and IDE exercises after the change.

Interestingly, the change only benefitted full-time students. This suggests a need to tailor scaffolding to different kinds of student.

Real vs Simplified Coding Environment





Nguyen, H., Bogart, C., Savelka, J., Zhang, Y, and Sakr, M. Examining the Trade-offs between Simplified and Realistic Coding Environments in an Introductory Python Programming Class. (ECTEL 2024)

This material is based upon work supported by the National Science Foundation under Grant #2111305, "Social and Interactive Learning at Community Colleges: Investigating Data-driven Feedback for Improving Student Outcomes with Project Based Learning (SAIL-CC)". Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.