TIPS, TRICKS & TECHNIQUES

# Lowering Barriers to Coding: Incorporating Generative Artificial Intelligence to Support Data Analysis in a Biology Class


ANDREW A. DAVINACK

**Abstract**

*Integrating generative artificial intelligence (AI) into undergraduate biological data analysis education offers a promising approach to lowering barriers to coding for biology majors. As "big data" becomes increasingly relevant in life sciences, essential programming skills often lead to student frustration due to a lack of prior coding experience. To mitigate barriers to coding, ChatGPT, an AI chatbot, was utilized as a debugging assistant in an undergraduate biological data analysis course. This course involves extensive Python programming, data cleaning, and statistical modeling. ChatGPT provided real-time feedback on coding errors, enhancing student engagement and learning efficiency. Informal feedback suggested that AI integration supported more independent troubleshooting and transformed students' approach to coding by making it more accessible. The implementation highlights the potential of AI-assisted tools in education, suggesting they can improve learning outcomes by reducing cognitive barriers associated with programming.*

**Key Words:** *Python; inclusive; inquiry; AI.*

## ○ Introduction

The integration of programming into biology curricula is becoming increasingly essential due to the rise of "big data" in the life sciences (Auker & Barthelmess, 2020; David, 2021; Emery & Morgan, 2017; Fillinger et al., 2019; Rubinstein & Chor, 2014;). Yet, many biology majors enter undergraduate programs with little to no coding experience, often leading to frustration and discouragement (Auker & Barthelmess, 2020; Emery et al., 2021; Zuvanov et al., 2021). As coding becomes an indispensable part of biological research and data analysis, educators must find ways to make programming more accessible to students from diverse backgrounds, while also recognizing the already substantial load many traditional four-year undergraduate biology programs (both B.A. and B.S.) impose on students. One potential solution is the use of artificial intelligence (AI), specifically large language models (LLMs), as a debugging assistant. These LLM-powered tools have been widely recognized as a disruptive technology in the educational landscape (Frith, 2023; Lo, 2023). While their long-term effects on student learning remain largely unknown and concerns persist among educators (Chan, 2023), they present opportunities to enhance inclusivity.

In the field of computer programming, AI chatbots are increasingly utilized to offer instant feedback on coding errors, suggest alternative solutions, and tailor responses to specific needs (DePalma et al., 2024; Liu et al., 2024). In this section, I will outline a systematic approach that enables biology students to responsibly leverage these AI features for debugging their code. This strategy aims to enhance their learning experience by integrating AI tools effectively and ethically into their programming practice. ChatGPT, the LLM interface developed by OpenAI and powered by the GPT-3.5 architecture at the time of instruction, was introduced into a sophomore-level biology course midway through the semester. Although no formal evaluation protocol was implemented at the end of the course, I will share instructional observations and generalized impressions based on informal student feedback collected during classroom discussions. These reflections are intended to assist instructors considering the integration of AI tools into courses requiring substantial data analysis.

## ○ Class Profile & Description

The course is a sophomore-level undergraduate class introducing biology majors to essential data science concepts, including Python programming, data cleaning, and statistical modeling. It fulfills a quantitative requirement in the Biology major. The course meets three times weekly, combining lectures on statistical theory, lab-style scripting sessions, and a weekly coding practical. As part of a pedagogical experiment, ChatGPT was introduced mid-semester as a debugging assistant. It was chosen for its accessibility; the standard version is free to use, making it ideal for educational contexts. All enrolled students had limited or no prior coding experience. The course is fundamentally an applied statistics course taught by biologists, specifically designed to equip students with the skills needed to perform statistical analyses across various biological data types (course syllabus available as a supplementary document). To simulate the challenges of working with real-world data, all datasets used in the course were sourced from the DRYAD database.

*The American Biology Teacher*, Vol. 87, No. 9, pp. 547–552, ISSN 0002-7685, electronic ISSN 1938-4211.  DOI: https://doi.org/10.1525/abt.2025.87.9.547.

Students are encouraged to frequently utilize this extensive repository to hone their data cleaning, wrangling, visualization, and statistical analysis skills. For their final project, students are tasked with repurposing the dataset from a published study to test new hypotheses and prepare a manuscript suitable for submission to a peer-reviewed journal of their choice.

ChatGPT was formally introduced to students midway through the semester during a dedicated in-class session. Although many students were already familiar with the tool from personal or academic use, I conducted a live demonstration to show how it could be applied for debugging Python code specifically within the course context. This demonstration included examples of effective prompts, interpreting AI output, and verifying accuracy. While ChatGPT use was not graded or required, students were encouraged to experiment with the tool during weekly practical assignments and the final project. At the time this course was taught in early 2023, there were no formal institutional policies on student use of generative AI tools such as ChatGPT. Faculty were encouraged to develop their own course-specific policies based on their teaching goals. In response, I independently created a set of ethical usage guidelines tailored to the course objectives, which emphasized transparency, academic integrity, and the role of AI as a support tool rather than a substitute for student effort. These guidelines (see Figure 1) were shared with students when ChatGPT was introduced and were referenced regularly to help guide responsible use throughout the remainder of the semester. Informal feedback was gathered through in-class discussions near the end of the term, without collecting any identifying information. These informal reflections were intended to guide future pedagogical strategies rather than serve as formal research data.

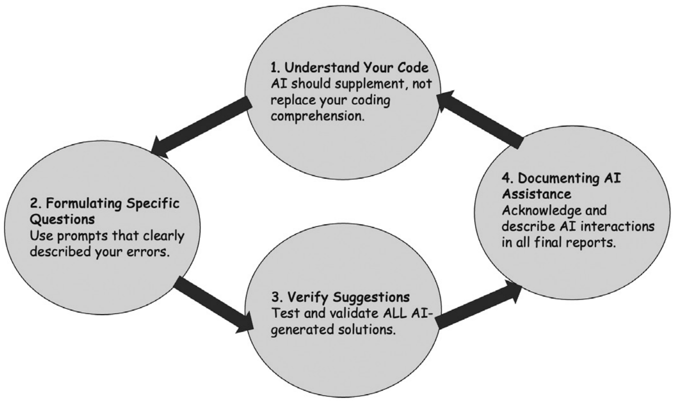


**Figure 1.** Flow chart depicting AI-powered debugging best practices for a biological data analysis course.

## ○ Case Examples of Debugging with ChatGPT

### *Syntax Errors*

A syntax error occurs when the Python interpreter encounters code that does not follow the proper structure or rules of the language. This often results from missing punctuation, incorrect indentation, or improperly formatted statements. Syntax errors prevent the program from running until they are corrected.

1. The following is an example of a syntax error in a simple script designed to perform a student *t*-test on two sets of data (group 1 and group 2) (Figure 2).
2. The following prompt was submitted to ChatGPT to inquire about the error (Figure 3).
3. Validation of suggestions provided by ChatGPT.

ChatGPT suggested that the student forgot to close the parentheses on line 14. After applying the recommendation, the code worked (Figure 4).

4. Documentation of AI Use for Debugging: This information was provided with all data analysis reports that were submitted by each student (Figure 5).


**Figure 2.** Invalid syntax error committed during execution of a student *t*-test script.

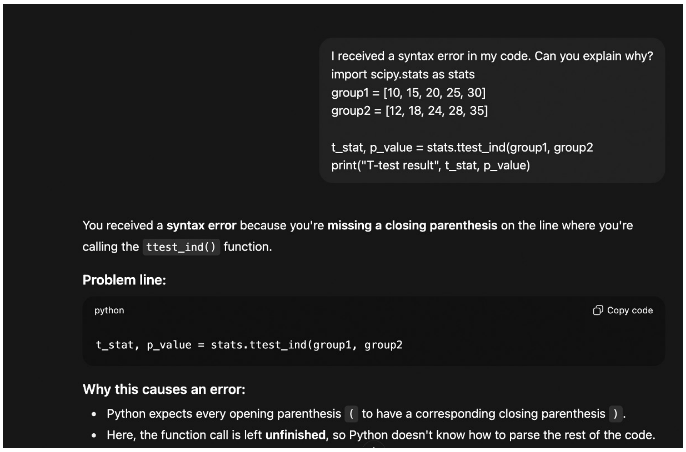


**Figure 3.** ChatGPT prompt provided to rectify syntax error with debugging response shown below.

### *KeyErrors*

A KeyError in Python occurs when trying to access a dictionary or a DataFrame column using a key that does not exist. This error typically happens when a column name in a CSV (comma separated value) file is misspelled or missing or a dictionary key is not found in the dictionary.

For example, the following dataset describes the seasonal parasite load (prevalence and intensity) of a marine snail infected with a trematode parasite (Figure 6). Additionally, water quality parameters are provided as additional data. Students were required to create boxplots of mean intensities and perform a Kruskal-Wallis H test to determine whether there were significant differences in parasite load across seasons. Before conducting statistical analyses, the data file (saved as a CSV) must be loaded into Python as a DataFrame. The relevant column (mean_intensity) must then be sliced and stored under a new variable for further processing.

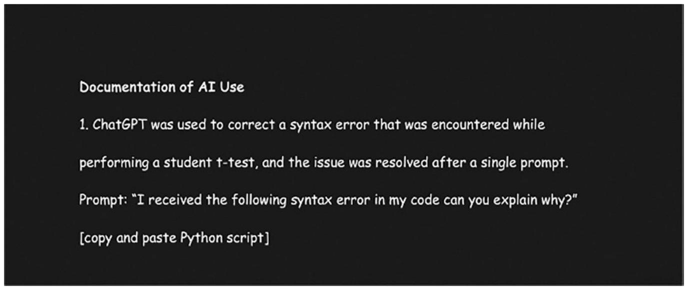


**Figure 5.** Documentation of AI Use required by all students who used ChatGPT in the course.

Because the term *variable* holds a distinct meaning in biology—often referring to measured traits or conditions in experimental design, I encouraged students to use the word *bin* when naming programming objects that store data subsets. This helped reduce confusion between coding terminology and biological concepts, especially for students with limited experience in programming. Instructors in other disciplines facing similar terminology overlap may benefit from adopting alternative labels that align with their students' conceptual frameworks.

Quite a few students encountered KeyErrors not realizing that they needed to type the name of the column in their script exactly as it appeared in the CSV file. To solve this error using ChatGPT, students needed to include three pieces of information in their prompt: (1) the CSV data being analyzed, (2) the error message received, and (3) the script they were using (Figure 7).

## ○ Assessment of ChatGPT Integration

Informal observations and class discussions indicated that the majority of student difficulties were related to syntax errors and KeyErrors—issues common among novice programmers. This was

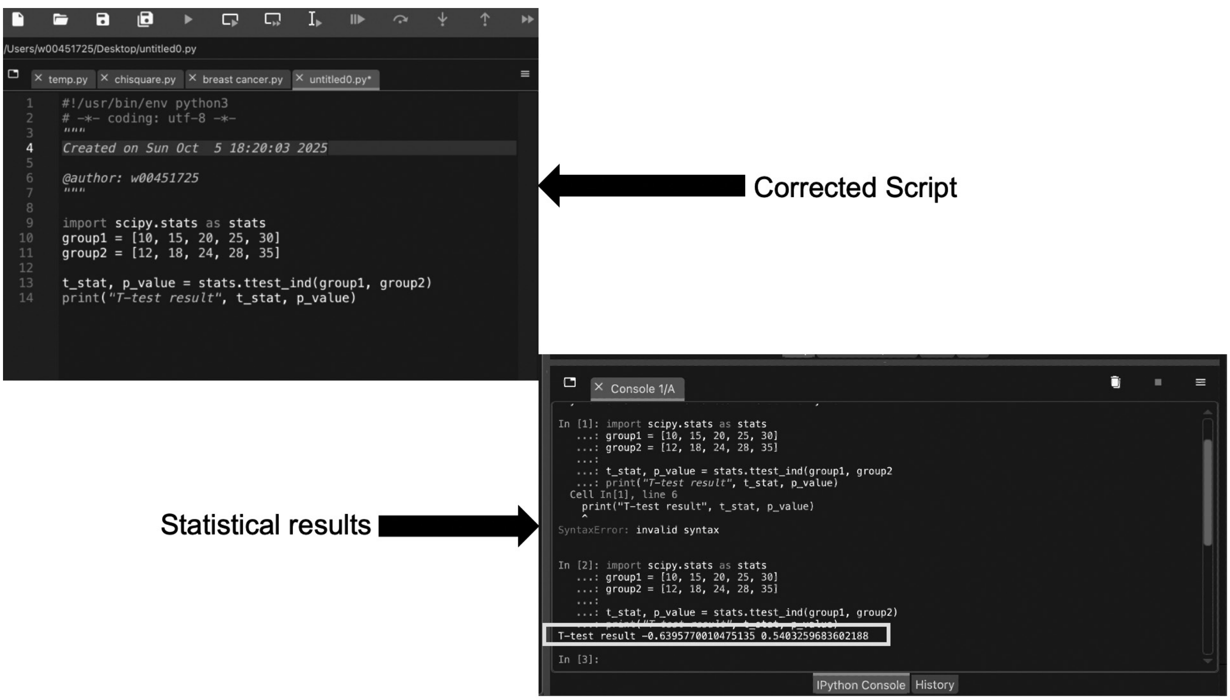


**Figure 4.** Validation of ChatGPT debugging suggestion. Rectangular circle in bottom panel shows *t*-test results (*t*-stat and *p*-value).

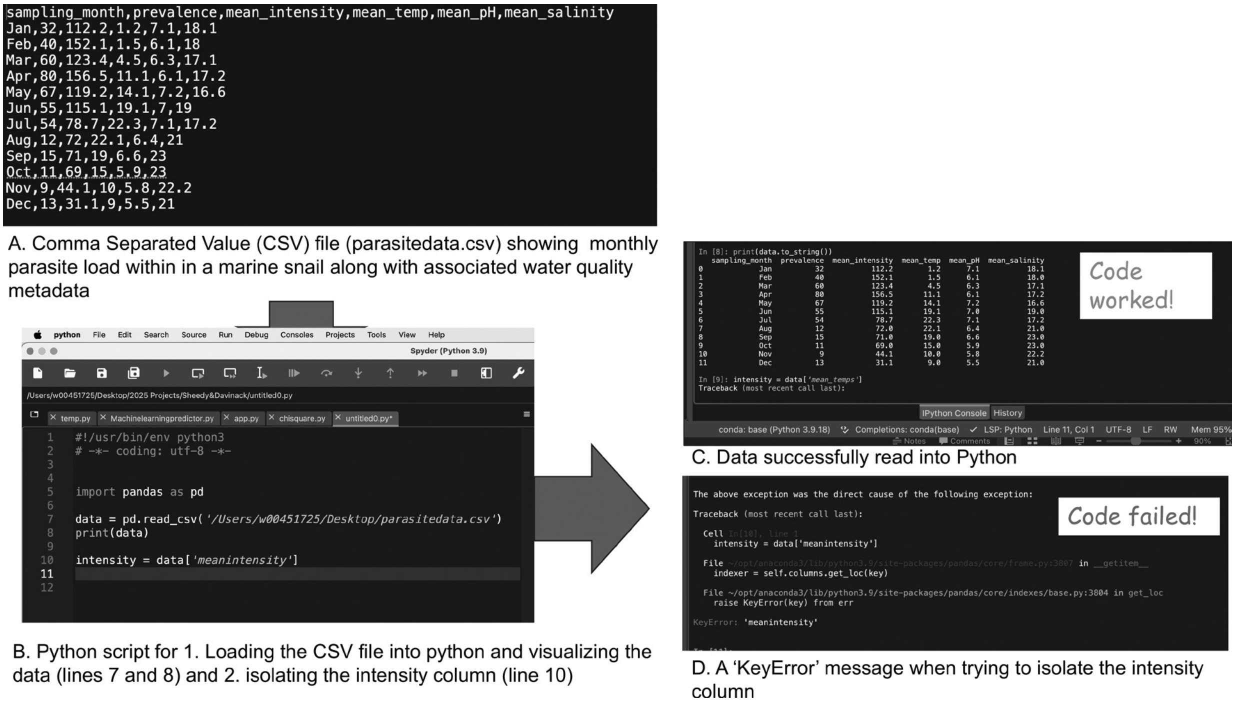


**Figure 6.** (A) Student-submitted Python script to load a CSV file and extract the "mean_intensity" column for analysis; (B) prompt submitted to ChatGPT to troubleshoot a KeyError arising from the incorrect reference to "mean_intensity"; (C) ChatGPT's suggested correction, showing an updated version of the script that resolves the KeyError (lines 7–8); (D) output showing correct slicing and storage of the "mean_intensity" column following ChatGPT's suggestion.


**Figure 7.** ChatGPT successfully debugged the code with a simple explanation and also rewrote the code, which can then be copied or rewritten directly into the student's own Python command line.

not surprising, as these errors are among the most common mistakes made by first-time coders who are still learning the operational rules of the language. One of the most noticeable changes, based on my instructional observations, was that more students began completing the Friday practicals within the 50-minute timeframe, compared with the earlier part of the semester. In the first half of the semester, prior to the intervention of ChatGPT, students often took 2–5 days to complete the practicals. Several students commented during discussions that the ability to receive immediate feedback from ChatGPT helped them iterate solutions more quickly. Many students felt that AI assistance gave them a sense of control over the debugging process, making coding less intimidating.

It is worth noting that ChatGPT did not always provide correct or complete answers on the first attempt. Students often encountered generic or partially incorrect suggestions, especially when their prompts lacked detail. In response, they learned to iterate by refining their questions or requesting clarification, and they documented these exchanges in their submissions. This process reinforced the value of clear communication and helped students build confidence in critically assessing AI output rather than relying on it uncritically. These interactions underscored the importance of guiding students not only in how to use AI tools, but also in how to question and verify their responses.

My experience with incorporating generative AI in this data analysis course suggests that AI-powered debugging assistants can enhance student learning by lowering barriers to coding and creating a more inclusive learning environment. By providing real-time feedback, AI tools empowered students in this course to troubleshoot problems independently, reducing reliance on instructor intervention. It is essential, however, to emphasize that AI should be used as a supplemental tool rather than a substitute for conceptual understanding.

Throughout the semester, I observed students engaging with ChatGPT primarily as a tool for support rather than substitution. Because coding practicals were completed during class, I had the opportunity to monitor student usage in real time. Most students approached debugging with the intent to understand and correct their errors rather than simply copy solutions. Interestingly, several students expressed that AI tools helped them become more independent and confident coders. This experience has shaped my view that structured, guided use of AI—perhaps through a formalized copilot model—could enhance learning while still fostering student ownership of their work. Interestingly, I noticed that students who used AI debugging tools appeared more willing to experiment with different coding approaches and reflect on their errors. While anecdotal, these observations suggest that generative AI has strong potential as a scaffold for developing coding fluency among biology students. By offloading the debugging labor to AI, students appeared to devote more time to interpreting the data and thinking critically about their analyses, based on the depth and clarity of their written reports. While not formally assessed, this shift suggested an increased focus on the biological questions driving the data.

Any STEM discipline that is dependent on science education (such as the biological sciences) stands to benefit significantly from generative AI, as it redistributes the division of labor in a way that enhances learning efficiency and problem-solving. However, while generative AI holds promise for increasing equity among students of different abilities, it also presents new challenges. For instance, disparities may arise as some students might afford premium AI services offering advanced debugging capabilities, thus gaining an advantage over peers using the standard version. For example, the premium version of ChatGPT allows for users to directly attach data files and have the algorithm directly analyze the data while also providing the script as an output for the user to replicate on their machine. This discrepancy could potentially reintroduce inequities within a supposedly leveled learning environment.

I believe that future work should focus on developing a structured, data-driven framework for integrating AI into biological data science education, potentially incorporating a “copilot” model that guides students through debugging while reinforcing key programming principles. This model should actively address common misconceptions about AI capabilities, particularly the belief that tools such as ChatGPT possess reasoning abilities similar to humans. For example, Liffiton et al. (2023) recently deployed a novel copilot LLM model known as “CodeHelp” in a first-year computer science course, which was well received by students who valued its ability to resolve errors and also by instructors for its ability to complement rather than replace instructor-driven support. It is crucial for students to understand that AI does not “reason” or “think”; instead, it generates responses based on patterns and examples from its extensive training data (Mahowald et al., 2024). In addition, a recent study has shown that the use of LLMs has a negative cognitive impact on students who used it as an external support system for writing essays versus a control group, with the former exhibiting reduced functional neural connections (Kosmyna et al., 2025). This is particularly concerning because it suggests that LLMs limit recruitment of brain regions involved in higher-order thinking—a staple in navigating complex problems in higher education and beyond. This understanding will help students critically evaluate AI suggestions and recognize the tool’s limitations.

It is worth noting that because ChatGPT was still a new and rapidly evolving tool at the time of course delivery, a comprehensive discussion of AI ethics—including topics such as data privacy, model transparency, and training bias—was not integrated into the curriculum. Future iterations of the course will include a more structured exploration of these issues, as awareness and resources surrounding responsible AI use in education continue to develop. In addition, ensuring that students are familiar with non-AI sources of help such as Stack Overflow and the actual programming language documentation should be introduced early on for students who prefer more traditional outlets for troubleshooting.

Educational institutions should also consider strategies for providing equitable access to these AI resources, ensuring that all students benefit equally. As data science continues to play a critical role in biology, it is imperative that educators find innovative ways to make coding accessible to all students. AI-assisted debugging presents a promising approach to reducing frustration, enhancing engagement, and improving learning outcomes in biological data analysis courses. By thoughtfully integrating AI tools into the curriculum, and emphasizing the importance of student engagement and oversight, educators can foster a more inclusive and effective learning experience for the next generation of biologists.

Incorporating discussions about the AI’s reliance on training data can also demystify how these tools function and dispel any overestimations of their capabilities. This educational strategy ensures students are aware that while AI can provide immediate feedback and support, it operates within the confines of the data it has been trained on. Hence, students are the ultimate arbiters of the quality and applicability of the code they write. This approach not only enhances learning but also prepares students to use AI responsibly and effectively in their future careers, aware of both its potential and its limits.

## ○ Acknowledgments

This work was supported by the Wheaton Inclusive STEM Excellence (WISE) initiative funded by the Howard Hughes Medical Institute (HHMI).

## References

Auker, L. A., & Barthelmess, E. L. (2020) Teaching R in the undergraduate ecology classroom: Approaches, lessons learned, and recommendation. *Ecosphere, 11*, e03060.

Chan, C. K. Y. (2023) A comprehensive AI policy education framework for university teaching and learning. *International Journal of Educational Technology in Higher Education, 20*: 38. https://doi.org/10.1186/s41239-023-00408-3

David, A. A. (2021). Introducing python programming into undergraduate Biology. *The American Biology Teacher, 83*, 33–41.

DePalma, K., Miminoshvili, I., Henselder, C., Moss, K., & AlOmar, E. M. (2024). Exploring ChatGPT's code refactoring capabilities: An empirical study. *Expert Systematics with Applications*, *249*, 123602.

Emery, L. R., & Morgan, S. L. (2017). The application of project-based learning bioinformatics training. *PLoS Computational Biology*, *13*, e1005620.

Emery, N. C., Crispo, E., Supp, S. R., Farrell, K. J., Kerkhoff, A. J., Bledsoe, E. K., O'Donnell, K. L., McCall, A. C., & Aiello-Lammens, M. E. (2021). Data science in undergraduate life science education: A need for instructor skills training. *BioScience 71*, 1274–1287.

Fillinger, S., de la Garza, L., Peltzer, A., Kohlbacher, O. & Nahnsen, S. (2019). Challenges of big data integration in the life sciences. *Analytical and Bioanalytical Chemistry*, *411*, 6791–6800.

Frith, K. H. (2023) ChatGPT: Disruptive educational technology. *Nursing Education Perspectives*, *44*, 198–199.

Kosmyna, N., Hauptmann, E., Yuan, Y. T., Situ, J., Liao, X-H., Beresnitzky, A. V., Braunstein, I., & Maes, P. (2025). Your brain on ChatGPT: Accumulation of cognitive debt when using an AI assistant for essay writing task. *arXiv preprint arXiv:2506.08872.*

Liffiton, M., Sheese, B., Savelka, J., & Denny, P. (2023). CodeHelp: Using large language models with guardrails for scalable support in programming classes. *Koli Calling '23: Proceedings of the 23rd Koli Calling International Conference on Computing Education Research 8*, 1–11.

Liu, Y., Le-Cong, T., Widyasari, R., Tantithamthavorn, C., Li, L., Le, X-B.D. & Lo, D. (2024). Refining ChatGPT-generated code: characterizing and mitigating code quality issues. *ACM Transactions on Software Engineering and Methodology*, *33*, 1–26.

Lo, C. K. (2023) What is the impact of ChatGPT on education? A rapid review of the literature. *Educational sciences*, *13*, 410.

Mahowald, K., Ivanova, A. A., Blank, I. A., Kanwisher, N., Tenenbaum, J. B., & Fedorenko, E. (2024). Dissociating language and thought in large language models. *Trends in Cognitive Sciences 28*, 517–540.

Rubinstein, A., & Chor, B. (2014) Computation thinking in life science education. *PLoS Computational Biology*, *10*, e1003897.

Zuvanov, L., Basso Garcia, A. L., Correr, F. H., Bizarria Jr, R., Filho, A. P. D. C., Da Costa, A. H., Thomaz, A. T., Pinheiro, A. L. M., Riano-Pachon, D. M., Winck, F. V., Esteves, F. G., Margarido, G. R. A., Casagrande, G. M. S., Frajacomo, H. C., Martins, L., Cavalheiro, M. F., Grachet, N. G., Da Silva, R. G. C., Cerri, R., Ramos, R. T. J., … & Correa dos Santos, R. A. (2021) The experience of teaching introductory programming skills to bioscientists in Brazil. *PLoS Computational Biology* 17: e1009534.

ANDREW A. DAVINACK (davinack_drew@wheatoncollege.edu) is an assistant professor of biology in the Department of Biological, Chemical and Environmental Sciences at Wheaton College in Norton, MA.