WIP: Integrating programming-based modules into a materials characterization laboratory course to reinforce data science and scientific writing

Abstract

The interdisciplinary nature of materials science and engineering (MSE) asks undergraduate majors in MSE to develop materials science domain knowledge in conjunction with complementary skills such as data science (DS) and scientific writing (SW). With little room to pack additional courses into MSE curricula, better integration of these transferable skills into existing courses will help train our students to succeed in the modern workforce. This Work in Progress details the development of a series of programming-based modules to complement the data analysis in a materials characterization laboratory course. We use the Jupyter Book software to design a scaffolded series of Python-based exercises that focus primarily on data visualization, with additional exercises on tabular data analysis, curve fitting, and image processing. Pre- and post-course surveys suggest that these modules had a positive impact on student learning and that students recognize the importance of these skills in MSE.

Keywords

Materials science, Data science, Laboratory reports

Introduction

In the modern age, scientists and engineers must be equipped with not only deep domain expertise, but also several transferable skills if they wish to be successful at their jobs [1]. We focus on two of these skills in particular, data science (DS) and scientific writing (SW), which have been discussed in recent reports from the National Academies [2, 3], ABET [4], and university educators [5–8]. These reports collectively highlight the importance of DS and SW in engineering practice and identify opportunities for students to build these skills through courses and programs, but it remains unclear how we can add more topics into an already-packed materials science and engineering (MSE) curriculum. While many schools offer dedicated DS and SW courses, these courses are often not required and lack examples in the MSE domain, which can leave some students unaware of the applicability of these skills in MSE.

We believe that there is an opportunity, given advancements in computing software and STEM pedagogy, to better integrate DS and SW practices into the MSE curriculum. Many institutions (including UC Berkeley) require an introductory computing course in their engineering curriculum, which provides students with a general introduction to algorithms and computational thinking. This is the foundation on which we introduce DS concepts, facilitated by open-source software such as Python and Jupyter to enhance the accessibility and scalability of this knowledge. Instead of using canonical problems and datasets, we teach these tools using real experimental data collected by undergraduates in an upper-division materials characterization laboratory course

(MSE 104L) at our institution, which is a large, public, research-intensive university in the United States. In MSE 104L, students perform a series of experiments (see Table 1), analyze the data they collect, and write a lab report interpreting their data for each experiment. Student feedback from previous years indicate a desire for more support on data analysis and report writing (partly because the laboratory sessions are focused on machine operation and data collection), which motivated us to design programming-based modules that could easily integrate into the post-experiment procedures, similar to what other instructors have done for physical chemistry labs [9, 10]. The existing course structure remains unchanged and provides a natural environment for modernizing the MSE curriculum as students learn DS and SW skills in context.

Lab #	Experiment / Topic	Python Notebooks
	Intro to modules	- How to use Jupyter Book - Intro to Python
1	X-ray emission	- Intro to plotting - Lab 1 exercises (new: plotting and curve fitting)
2	Powder X-ray diffraction	- Intro to tabular data - Lab 2 exercises (new: pandas DataFrames)
3	Precision X-ray diffraction	- Lab 3 exercises (new: functions and correlations)
4	Energy-dispersive X-ray spectroscopy & Scanning electron microscopy	- Lab 4 exercises (new: annotations and clustering)
5	Self-guided experiments	[None]
6	Transmission electron microscopy	[None]

Table 1: Outline of the six laboratory experiments and the corresponding Python-based learning modules for a materials characterization laboratory course.

Methods

Table 1 shows the six experiments in MSE 104L and the corresponding modules for each one. Each set of exercises is Python-based and written in a Jupyter notebook [11], which allows authors to merge prose, graphics, and code into a single document. Based on our previous work [7], we use the Jupyter Book software [12] to compile the individual notebooks into an interactive digital text that can be stored in a GitHub repository and hosted online (Figure 1), which obviates the need to manage local software installations and is freely accessible by anyone [13]. Moreover, the code on the individual pages can be edited and executed by opening the page in a JupyterHub environment [14], which is generously provisioned by our campus for our students. Students are able to do all of their analysis in the cloud and focus on learning the concepts while having their work saved in one place. Each page can also be opened in Google Colaboratory [15], which is an alternative option for running the Python notebooks in the cloud for those without JupyterHub access.

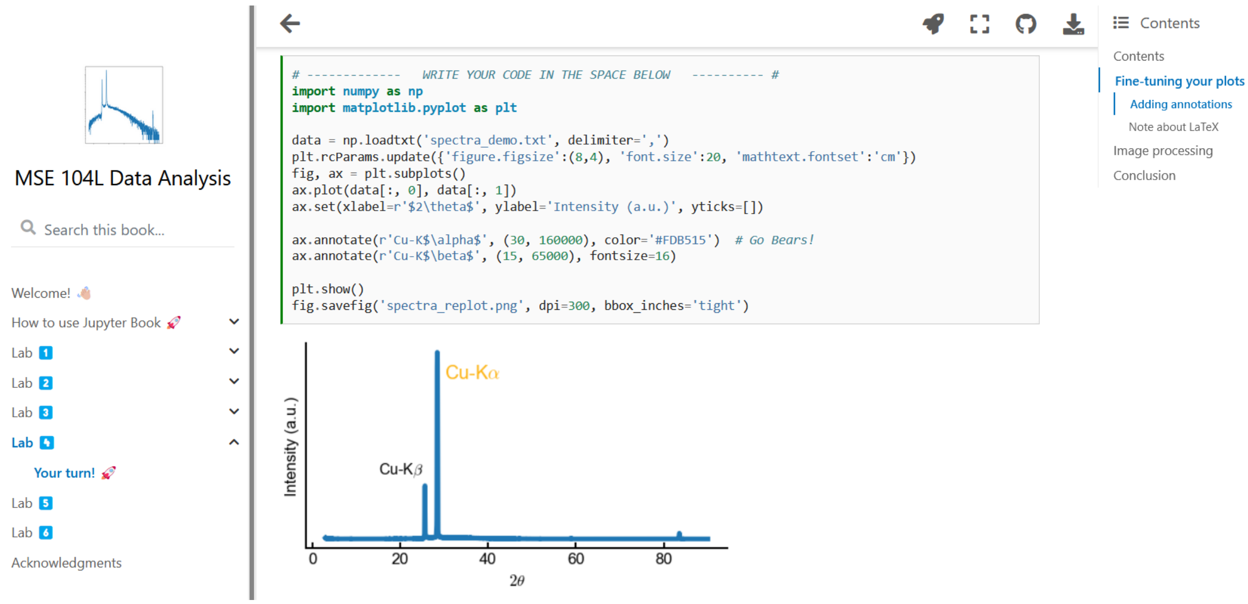


Figure 1: Screenshot of a page of the learning module in the Jupyter Book Web UI [13]. Interactivity is enabled by clicking the rocket symbol which launches the page in JupyterHub [14] or Google Colab [15].

As DS is a broad discipline, for this first iteration we emphasize *data visualization* as it is an important skill in MSE that can easily be misused [16]. Using figures to help structure the narrative is also an important skill for professional communication [17], which we hope enables our modules to support SW development from a "data-first" approach. Past student performance on lab reports in MSE 104L has shown some weaknesses in terms of data interpretation, visualization, and organization, which are consistent with previous reports [5, 6]. We create a series of scaffolded exercises, following the sequence of the laboratory experiments (Table 1), that teach students how to use the versatile Python visualization library Matplotlib [18]. In addition to specific exercises corresponding to each lab report, there are also general tutorials at the beginning for those less familiar with the programming environment. We include discussions of best practices in data visualization and additional exercises on tabular data (e.g., solving for lattice parameters in X-ray diffraction), curve fitting (e.g., Nelson-Riley method [19]), and k-means clustering (e.g., image segmentation). In doing so, we hope to expose students to common scientific computing libraries such as NumPy [20] and pandas [21] so they are aware of how these powerful tools can support their experiments and deepen their understanding of the science. That being said, to avoid overburdening students in this initial rollout, we make the modules optional and let the students decide which tools they wish to use for data analysis in the labs.

At the start of the semester, we explained to all students in MSE 104L the purpose of the learning modules and the details of this research study, which qualified for exempt status through the UC Berkeley Institutional Review Board (CPHS #2022-10-15691). Students are surveyed anonymously at the start of the semester to assess their academic preparation and predispositions toward DS and SW, and then they are surveyed a second time (not linked) near the end to assess the effectiveness of the learning modules and any changes in their beliefs. The two surveys contain a mix of multiple-choice, 7-point Likert-type scale (7 being "Strongly agree"), and short-answer questions whose responses are inductively coded by the lead author (for a list of questions, see SI-B: Survey questions). For this study, we seek to answer the following research questions (RQs):

RQ1: How well are DS and SW skills integrated into the current MSE curriculum?

RQ2: How do MSE students view DS and SW in the context of their work?

Results and Discussion

We will only report and discuss the most salient results in this section, and the rest can be found in SI-A: Additional survey results. All 42 students in MSE 104L consented to participate in the study, with 88 % being juniors and above, and 88 % majoring in MSE. Over half of the students were male (60 %) and a little over a third were female (36 %). Other data such as race and ethnicity were not collected and the full demographic results are found in Table S1.

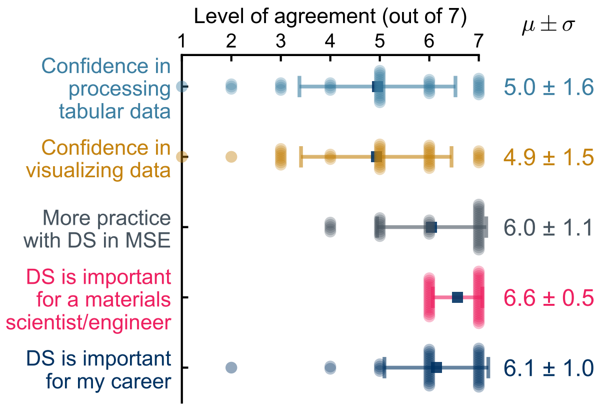
Pre-course survey

Subject	Yes	No
Computing	**33** (**79%**)	9 (21 %)
Data science / Statistics	7 (17 %)	**35** (**83%**)
Scientific writing	3 (7 %)	**39** (**93%**)

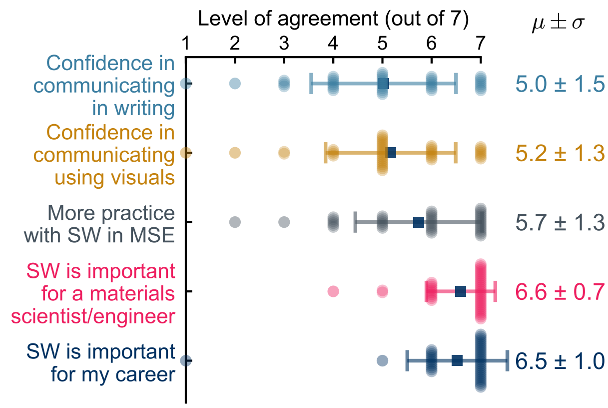
Table 2: Previous or concurrent coursework taken by the students ($n = 42$).

Table 2 shows the academic background of these students, which is largely consistent with our expectations. Nearly 80 % of the students have taken a computing course, but only 17 % of them have taken a DS or statistics course. Even fewer had taken a formal course in SW and it is likely that their only prior exposure to technical writing was in the laboratory component to the gateway MSE course. MSE majors at our institution are required to take a 4-unit introductory computing course, but there is no strict requirement on the timing. Notably, of the nine students who had not taken a computing course, all of them answered "No" to the other two questions, eight of them were male identifying, and none of them were sophomores. This suggests that there does not appear to be a gender disparity in terms of access to computing courses in this sample of students and that later class years may see a benefit to taking computing courses early.

Table S2 details the programming languages used by students in their MSE courses *only*, from which it is clear that MATLAB is the most common required programming language (55 % of students were required to use it at least once) followed by Mathematica (31 %) and Python (26 %). It is interesting to note that Python is the only programming language where the most common reason students used it in MSE courses was by "personal choice" (40 %), although roughly the same proportion of students (38 %) had never used it in their MSE courses. These results support our hypothesis that students generally lack formal training in DS and SW skills and that



(a) Data science (DS)



(b) Scientific writing (SW)

Figure 2: Pre-course survey results for student sentiment on (a) data science and (b) scientific writing. Students ($n = 42$) indicated their level of agreement with five prompts on a Likert-type scale (1 to 7, with 7 being "Strongly agree"). Each square marker and associated error bars correspond to the mean (μ) and one standard deviation (σ), respectively, and each dot is an individual student response.

appropriately scaffolded content in Python would be beneficial toward developing these capabilities.

Theme	Topic	Counts
Academic (27)	Research/expt.	15
	MSE	8
	Courses	4
Data-related (24)	Understanding	10
	Volume	7
	Visualization	5
	Communication	2
Life (18)	Transferable skill	7
	Easier life	5
	Applications	3
	Career skill	3

(a) Data science (DS)

Theme	Topic	Counts
Academic (20)	Research/expt.	19
	MSE	3
	Courses	1
Actions (34)	Communication	27
	Understanding	6
	Instruction	1
Life (15)	Career skill	11
	Transferable skill	3
	Enjoyment	1

(b) Scientific writing (SW)

Table 3: Inductively coded student responses ($n = 42$) to the question, "Why are these skills important (or not important) to you?" for (a) Data science and (b) Scientific writing. Topics were first identified (possibly multiple in a single response) and then grouped into themes.

The two sets of Likert-type scale questions on DS and SW gave similar results (Figure 2). Student confidence regarding four DS and communication tasks hovers around 5/7, indicating potential for improvement. In spite of this (or because of it), students wish to see more opportunities to practice DS and SW skills in their MSE courses as the current amount may not be enough (with respect to RQ1). They also tend to regard both DS and SW as important skills for materials scientists and their own careers (all greater than 6/7 in agreement, consistent with previous studies [22]), which is encouraging to see and signals to us that our learning modules may be appropriately targeted for this audience (RQ2). The first two questions for each topic in Figure 2 have a larger spread in the responses, and we note that the two choices for "1" for the two DS questions are selected by the same student, but a different one from the single student who selected the two choices for "1" for the two SW questions.

We asked students to elaborate on their agreement with the importance of DS and SW (RQ2), and we report the coded results in Table 3. A majority of the students discussed the importance of these skills in the context of scientific research, such as using DS tools to analyze experimental results or using SW to communicate findings. The data-related topics that appeared in the responses to the DS question mirrored those in students' definition of DS, although a greater proportion of students mentioned the importance of handling large volumes of data. It is not surprising that a majority of students felt SW was important for communication and their future careers, although it is interesting to note that fewer students felt that SW was an important transferable skill (11 to 3), whereas for DS the order of career vs. transferable skills was reversed (3 to 7).

Post-course survey

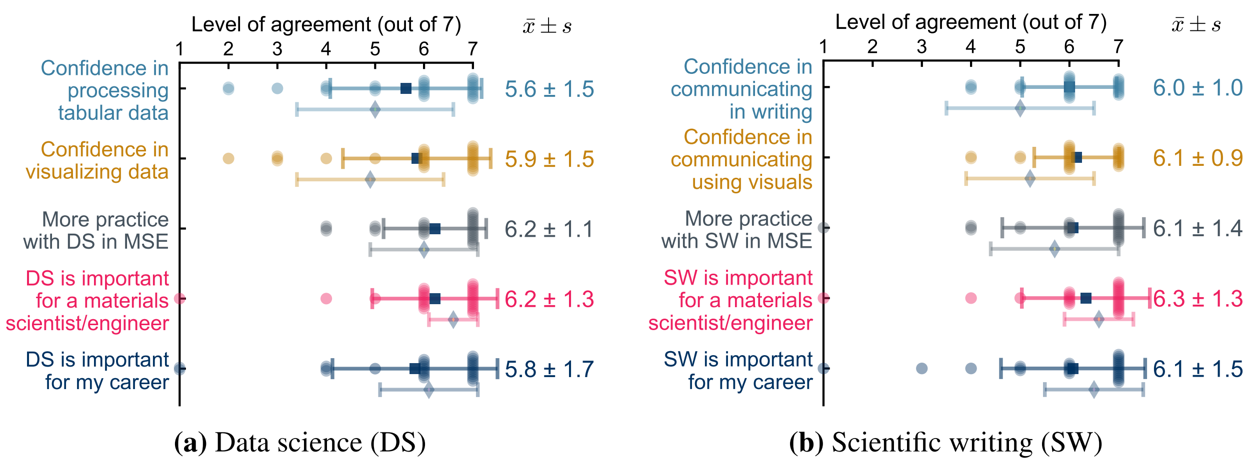


Figure 3: Post-course survey results ($n = 29$) for student sentiment on (a) data science and (b) scientific writing. The diamond marker and second set of error bars correspond to the mean and standard deviation in the pre-course survey results (Figure 2) for ease of comparison.

We surveyed the students a second time towards the end of the course and show the results from the same set of Likert-type scale questions in Figure 3. Unfortunately, we were only able to obtain 29 student responses in this survey instead of the full population of 42 students. Nevertheless, it appears that there is a noticeable increase in average student confidence in DS and SW skills with gains between 0.6 and 1.0 points out of 7. Moreover, the average scores for the questions asking for more practice with DS/SW effectively stayed the same, which suggests that more integration of

DS/SW topics into the MSE curriculum would be welcomed by this audience (RQ1). Figure S1 stratifies the results in Figure 3 by those students who used and did not use the modules, which numbered 18 and 11, respectively. It is interesting to note that among these two groups there is a large difference in confidence when it comes to DS skills but almost equal confidence in SW skills. This provides some evidence that our modules were effective at promoting DS but less effective at promoting SW; however, because our experimental design did not link the survey responses, we caution against drawing stronger conclusions of causation.

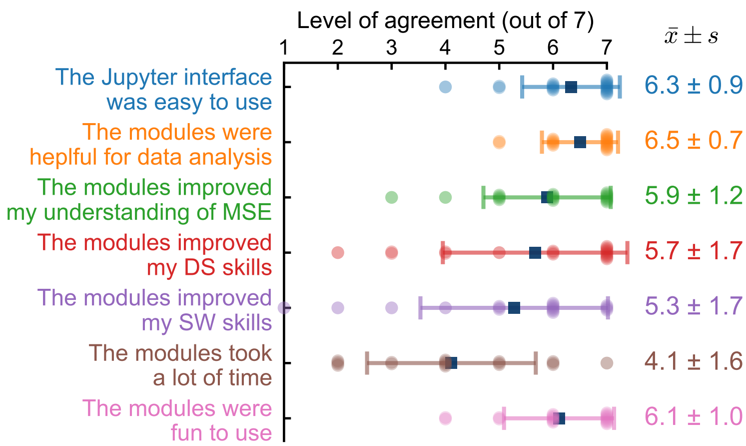


Figure 4: Post-course survey results for student sentiment on the learning modules ($n_{\text{used}} = 18$).

For the ($n_{\text{used}} = 18$) students who used the learning modules, six of them used all four modules and overall usage was higher for the first three labs (Table S3), which aligned with our expectations. Thirteen of those students had never taken a DS course and were likely using Jupyter and Python for the first time. As shown in Table S4, the top motivation for these 18 students was the belief that the modules would help with writing the reports and other primary motivations included the belief that they would support learning MSE and Python, and that the modules were easy to use. It is encouraging to see from another set of Likert-type scale questions (Figure 4) in the post-course survey that students had very positive experiences with the accessibility and utility of the programming-based learning modules. The questions explicitly asking for improvement in DS and SW skills had lower ratings and larger variations, which suggests more improvements could be made to support a greater range of learners, especially when it comes to written communication. Table S5 reveals that many students felt that some sections were lacking in detail and we acknowledge the challenges of using a new UI like Jupyter Book/Hub for the first time. Students felt like the modules took a moderate amount of time, although a more informative assessment could be to compare the amount of time it takes to do data analysis with these modules versus their personal choice.

In fact, personal preference for another method of data analysis was the most common reason given among the ($n_{\text{not}} = 11$) students who did not use any of the modules, followed by the fear that the modules would be too time consuming (Table S4). Most of them also didn't provide ways to encourage adoption (Table S5), but those who did proposed giving a tutorial at the beginning of the course or making the modules a required component of the course. The first suggestion can be implemented by giving a demonstration during the lab sections or even embedding tutorial videos into the Jupyter Book for asynchronous viewing. The second suggestion may very well be a motivation issue [22], but in light of the positive impacts of our learning modules and the large-scale redesign as demonstrated at other institutions [23], there are several merits to including these modules into the course learning objectives more broadly. By introducing these skills earlier and more systematically, we can increase the impact of these tools and better quantify learning gains as we equip students with the necessary skills for professional success.

Conclusion

This Work in Progress paper studies the effectiveness of programming-based learning modules in reinforcing data science and scientific writing in a materials characterization laboratory course. All of the open-source learning modules are freely available online [13], which underscores the scalability and accessibility of this approach, and more exercises can be added in the future in a straightforward manner. We find that MSE students have a strong desire to learn DS and SW in the context of their work (Figure 3) and the modules we designed are effective at guiding this development (Figure 4). The Jupyter ecosystem makes it easy to integrate interactive learning experiences into existing MSE courses in a streamlined way, and the barriers for instructors to acquire new skills [24] may be mitigated by the growing adoption of open-source tools and cooperative efforts between instructors.

Acknowledgments

We thank Chris Kumai, Chad Southard, Min Chen, Ji Guo, and Reed Yalisove for assistance with laboratory operations. E.C. acknowledges a fellowship through the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE-2146752. All figures with data are produced with Matplotlib [18].

Additional information

The authors declare no competing interests.

Correspondence should be addressed to Enze Chen at chenze@berkeley.edu.

References

[1] K. Börner, O. Scrivner, M. Gallant, et al., "Skill discrepancies between research, education, and jobs reveal the critical need to supply soft skills for the data economy," *Proceedings of the National Academy of Sciences*, vol. 115, no. 50, pp. 12630–12637, 2018.

[2] National Academies of Sciences, Engineering, and Medicine, "Communicating science effectively: A research agenda," tech. rep., National Academies Press, Washington, D.C., 2017.

[3] National Academies of Sciences, Engineering, and Medicine, "Data science for undergraduates: Opportunities and options," tech. rep., National Academies Press, Washington, D.C., 2018.

[4] ABET, "Criteria for accrediting engineering programs, 2022–2023," 2022. Accessed Jan. 2023.

[5] E. Wiebe, T. Hare, M. Carter, et al., "Supporting lab report writing in an introductory materials engineering lab," in *2001 ASEE Annual Conference*, pp. 6.917.1–6.917.11, 2001.

[6] A. Genau, "Teaching report writing in undergraduate labs," in *2020 ASEE Annual Conference*, 2020.

[7] E. Chen and M. Asta, "Using Jupyter tools to design an interactive textbook to guide undergraduate research in materials informatics," *Journal of Chemical Education*, vol. 99, no. 10, pp. 3601–3606, 2022.

[8] B. Afflerbach, N. Fathema, A. Gillian-Daniel, W. Crone, and D. Morgan, "Authentic undergraduate research in machine learning with the Informatics Skunkworks: A strategy for scalable apprenticeship applied to materials informatics research," in *2022 ASEE Annual Conference & Exposition*, 2022.

[9] E. J. Menke, "Series of Jupyter notebooks using Python for an analytical chemistry course," *Journal of Chemical Education*, vol. 97, no. 10, pp. 3899–3903, 2020.

[10] M. van Staveren, "Integrating Python into a physical chemistry lab," *Journal of Chemical Education*, vol. 99, no. 7, pp. 2604–2609, 2022.

[11] T. Kluyver, B. Ragan-Kelley, F. Pérez, et al., "Jupyter notebooks - a publishing format for reproducible computational workflows," in *Positioning and power in academic publishing: Players, agents and agendas* (F. Loizides and B. Schmidt, eds.), pp. 87–90, 2016.

[12] Executable Books Project, "Jupyter Book." Zenodo, 2020. v0.12.3.

[13] E. Chen and A. M. Minor, "MSE 104L Data Analysis." GitHub, 2023. `https://enze-chen.github.io/mse104l/`.

[14] Project Jupyter, "JupyterHub," 2021. Accessed Jan. 2023.

[15] Google Research, "Google Colaboratory," 2018. Accessed Jan. 2023.

[16] C. M. Parish and P. D. Edmondson, “Data visualization heuristics for the physical sciences,” *Materials & Design*, vol. 179, p. 107868, 2019.

[17] G. M. Janowski, “Preparing students for communicating in the professional world,” *JOM*, vol. 74, no. 8, pp. 2883–2884, 2022.

[18] J. D. Hunter, “Matplotlib: A 2D graphics environment,” *Computing in Science Engineering*, vol. 9, no. 3, pp. 90–95, 2007.

[19] J. B. Nelson and D. P. Riley, “An experimental investigation of extrapolation methods in the derivation of accurate unit-cell dimensions of crystals,” *Proceedings of the Physical Society*, vol. 57, no. 3, pp. 160–177, 1945.

[20] C. R. Harris, K. J. Millman, S. J. van der Walt, et al., “Array programming with NumPy,” *Nature*, vol. 585, no. 7825, pp. 357–362, 2020.

[21] W. McKinney, “Data structures for statistical computing in Python,” in *Proceedings of the 9th Python in Science Conference* (S. van der Walt and J. Millman, eds.), pp. 56–61, 2010.

[22] S. P. Gentry, “Motivation on programming assignments in materials science and engineering,” in *2019 ASEE Annual Conference & Exposition*, p. 16, 2019.

[23] G. M. Lu, D. R. Trinkle, A. Schleife, et al., “Impact of integrating computation into undergraduate curriculum: New modules and long-term trends,” in *2020 ASEE Annual Conference*, 2020.

[24] N. C. Emery, E. Crispo, S. R. Supp, et al., “Data science in undergraduate life science education: A need for instructor skills training,” *BioScience*, vol. 71, no. 12, pp. 1274–1287, 2021.

Supplementary Information

A: Additional survey results

Major	Responses
MSE	32 (76 %)
Other	5 (12 %)
MSE & Other	5 (12 %)

(a)

Class	Responses
Sophomore	5 (12 %)
Junior	27 (64 %)
Senior & older	10 (24 %)

(b)

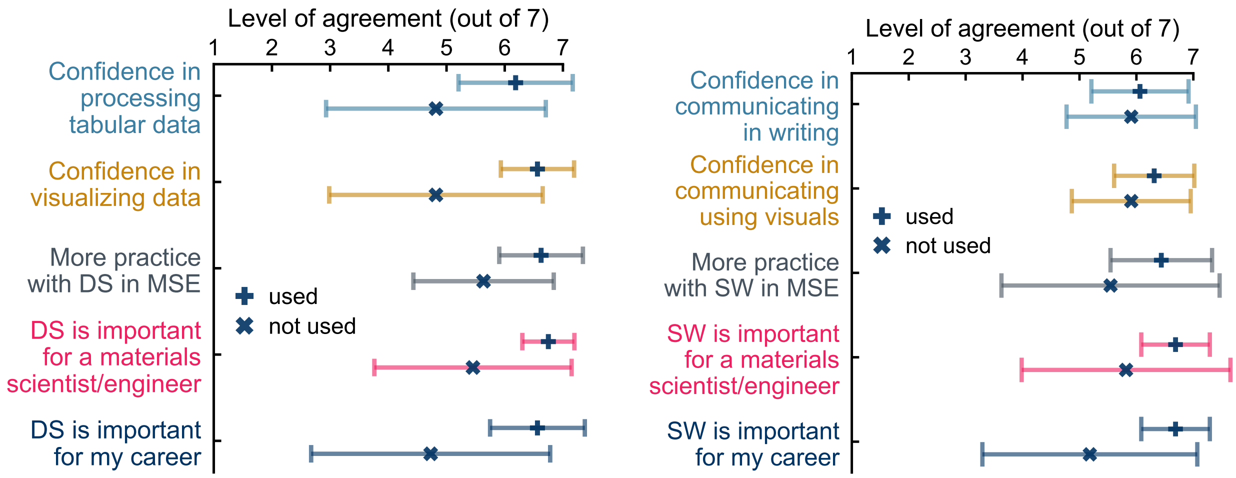
Gender	Responses
Male	25 (60 %)
Female	15 (36 %)
Non-binary	2 (5 %)

(c)

Table S1: Student demographics ($n = 42$) in terms of (a) Major, (b) Class year, and (c) Gender. Note that "Junior" includes 1st-year transfer students and "Senior" includes 2nd-year transfer students, etc.

Language	Reason		
	Required	Choice	Not used
C/C++	2 (5 %)	4 (10 %)	**36 (86%)**
Mathematica	13 (31 %)	8 (19 %)	**21 (50%)**
MATLAB	**23 (55%)**	14 (33 %)	10 (24 %)
Python	11 (26 %)	**17 (40%)**	16 (38 %)
R	2 (5 %)	1 (2 %)	**40 (95%)**
Other	5 (12 %)	8 (19 %)	**29 (69%)**

Table S2: Previous programming languages used by students ($n = 42$) *in their MSE courses only* and whether it was a course requirement, personal choice, or not used at all. The most common reason for each language is in boldface to guide the eye. Note that multiple choices were allowed, so rows may not sum to $100\,\%$.



(a) Data science (DS) **(b)** Scientific writing (SW)

Figure S1: Post-course survey results for student sentiment on (a) data science and (b) scientific writing, where the results from Fig. 3 are stratified by whether students used the modules ($n_{\text{used}} = 18$, upper bars) or not ($n_{\text{not}} = 11$, lower bars).

Lab topic	Responses
Lab 1: X-ray emission	12
Lab 2: Powder XRD	11
Lab 3: Precision XRD	13
Lab 4: EDS in the SEM	7

Table S3: Student responses to the question “Which labs did you use the modules for?”

Motivation	**%**
Helpful for the reports	78 %
Easy to use	56 %
Supported learning MSE concepts	56 %
Improve Python skills	56 %
Already planned to use Python	33 %
Improve DS skills	33 %

(a) Used the modules, $n_{\text{used}} = 18$.

Reason	**%**
Preferred own method	39 %
Too time consuming	22 %
Adapted to MATLAB	11 %
Would distract from MSE	11 %
Too difficult to understand	11 %
Not helpful for the reports	6 %

(b) Did not use the modules, $n_{\text{not}} = 11$.

Table S4: Top reasons for (a) using and (b) not using the Python-based online modules. Percentage is given out of the number of students who identified with each group. Multiple selections were allowed and thus percentages may not sum to $100\,\%$.

Question	Topic	Counts
Most helpful about the notebooks?	Plotting w/ Python	7
	Data processing w/ Python	6
	Connections to lab	4
Least helpful about the notebooks?	Nothing/ I don't know	8
	Need more explanation	7
	Jupyter environment confusing	5
What could we have done differently to encourage use?	Nothing/ I don't know	6
	Required course/assignment	2
	Tutorial videos/slides	2
How have your perspectives on DS/SW changed?	No change	3
	Should learn more	3
	DS/SW should be required	3

Table S5: Inductively coded student responses to free-response questions in the post-course survey. The first two questions are only asked to those who used the notebooks ($n_{\text{used}} = 18$), the third is only asked to those who didn't use the notebooks ($n_{\text{not}} = 11$), and the last question is asked to everyone. Only the most common topics are shown.

B: Survey questions

Pre- and post-surveys

- Personal and academic background
  - (MSE, Other) What is your Major?
  - (2nd, 3rd, 4th, Other) What is your year of study?
  - (Male, Female, Non-binary) What is your gender identity?
  - (Yes, No) Have you taken a course in...? {*Computing, DS/Statistics, Scientific communication/writing*}
  - (Required, Personal choice, Have not used) Please indicate which of the following programming languages you used **in your MATSCI courses** and why. {*C/C++, Mathematics, MATLAB, Python, R, Other*}
- Perspectives on data science (DS)
  - (Likert scale 1 (Strongly disagree) to 7 (Strongly agree)) Please indicate your agreement with the following:
    * I feel confident using programming tools to process tabular data (i.e., data in rows & columns).
    * I feel confident using programming tools to visualize my data (e.g., make plots).
    * There should be more opportunities to practice DS in MATSCI courses at `[University]`.
    * DS is an important skill for a materials scientist/engineer.
    * DS is an important skill for my future career.
  - Briefly, what does "data science" mean to you?
  - Why are DS skills important (or not important) to you?
- Perspectives on scientific writing (SW)
  - (Likert scale 1 (Strongly disagree) to 7 (Strongly agree)) Please indicate your agreement with the following:
    * I feel confident communicating scientific results in writing.
    * I feel confident communicating scientific results using visuals.
    * There should be more opportunities to practice SW in MATSCI courses at `[University]`.
    * SW is an important skill for a materials scientist/engineer.
    * SW is an important skill for my future career.
  - Why are SW skills important (or not important) to you?

Post-survey only, used at least one module

- (Multiple selection) What were some of your *initial* reasons for using the online modules?
  - Believed that the modules were required
  - Believed that the modules would help for the reports
  - Believed that the modules would support learning MSE concepts
  - The modules were easy to use
  - A classmate told me the modules were helpful
  - Already planned to use Python to analyze the data
  - Interested in improving Python skills
  - Interested in improving DS skills
  - Interested in improving SW skills

  - Other: [Enter your own answer]
- (Multiple selection) Which labs did you use the modules for? {*Lab 1, Lab 2, Lab 3, Lab 4*}
- (Likert scale 1 (Strongly disagree) to 7 (Strongly agree)) Please indicate your agreement with the following:
  - The Jupyter Book/Hub interface was easy to use.
  - The modules had helpful instructions for how to analyze the data.
  - The modules improved my understanding of the experiments/MSE.
  - The modules improved my DS skills.
  - The modules improved my SW skills.
  - The modules took a lot of time.
  - The modules were fun to use.
- (2 short-answer questions) For me, the {**most**, **least**} helpful topic of the modules was...
- (2 short-answer questions) How have your perspectives on {DS, SW} changed (if at all) since the beginning of the semester?

<u>Post-survey only, used none of the modules</u>

- (Multiple selection) What were some of your reasons for not using the Python modules for the labs?
  - Preferred my own method of analyzing data
  - Could not get Jupyter Book/Hub to work
  - Did not know that the modules were available
  - Using the modules would take too much time
  - Did not believe that the modules would help for the reports
  - A classmate told me the modules were not helpful
  - The modules were too difficult to understand
  - The modules would distract from learning MSE concepts
  - Not interested in improving Python skills
  - Not interested in improving DS skills
  - Not interested in improving SW skills
  - Other: [Enter your own answer]
- Could we have done anything differently to encourage you to use the materials?
- (2 short-answer questions) How have your perspectives on {DS, SW} changed (if at all) since the beginning of the semester?