# Interdisciplinary Computing Education: An Introductory Programming and Data Science Course for Postdoctoral Researchers in the Biosciences

Arko Barman
*Data to Knowledge Lab*
*Rice University*
Houston TX, USA
arko.barman@rice.edu

Leslie S. Beckman
*Office of Postdoctoral Affairs*
*UTHealth Houston*
Houston TX, USA
Leslie.Beckman@uth.tmc.edu

Yasmin Chebaro
*Office of Postdoctoral Affairs*
*UTHealth Houston*
Houston TX, USA
Yasmin.Chebaro@uth.tmc.edu

***Abstract*—This Innovative-Practice Full Paper presents the curriculum development of an introductory course in programming and data science for postdoctoral researchers (PDRs) in the biosciences. The use of computing software has become ubiquitous and a working knowledge of data science has become increasingly essential for researchers in all domains. However, curriculum development focusing on imparting foundational programming skills and fundamentals of data science for researchers in domains other than computing has been scarce. Thus, there is an unmet need for curriculum development involving computational thinking, programming, and the fundamentals of data science for this audience. Recognizing these growing needs and demands of researchers to learn programming and data science that can then be applied to their area of research or practice, we developed an introductory course in programming and data science for PDRs in biology and medicine. The primary goal of the course was to develop computational thinking skills in PDRs who hail from backgrounds that have traditionally not focused on inculcating computational thinking. This course covered the fundamental concepts of programming using either Python or R - languages that researchers outside the computing community use in numerous ways including the statistical analysis of large datasets that are becoming increasingly common in biomedical research. Further, PDRs enrolled in the course were introduced to some of the broad categories of problems in data science - exploratory data analysis, classification, regression, and clustering - along with relevant algorithms and how they can be applied to real-world datasets in their respective domains using packages or libraries in Python or R. We also report the feedback from the enrolled PDRs, lessons learned, and recommendations for instructors interested in designing similar curricula. Our course focusing on computing and data science education for postdoctoral scholars from a non-computing background demonstrates a promising model for incorporating computing education in other areas of study that do not traditionally have a focus on computing education as well as in continuing education.**

***Index Terms*—computer science education, interdisciplinary computing education, data science education, continuing education**

## I. Introduction

With the rapid growth of computing and data science in every domain of research, there has been an increasing demand for professionals, researchers, and faculty members who are not only knowledgeable in their domain of expertise but who also possess coding and data science skills [1], [2]. The world of biology and medicine is no exception – researchers have to routinely perform analysis of their data by leveraging sophisticated data science tools and algorithms [3]. In fact, several computing libraries, packages, and entire frameworks have been developed in languages like R and Python solely for the analysis of biomedical data [4]–[8]. It has been widely acknowledged that the domain of biosciences has been revolutionized by the growth and incorporation of computing methodologies [2], [9]. However, the incorporation of computing education in the biosciences has failed to keep pace with the ever-increasing advancement in computing research and education. Moreover, there has been an increased use of large genomic, proteomic, metabolomic, and other datasets in biological sciences (such as RNA sequencing and microarrays) in recent years. While wet-lab scientists and clinicians are able to obtain the physical genetic materials used in these analyses, they often cannot efficiently analyze the large dataset results that are obtained due to a lack of training in coding and data science.

While steps have been taken to incorporate computing education in the biosciences in recent years [10]–[12], most of the focus has been on developing supplementary computing curriculum at the undergraduate level [13]. The need for computing education for researchers and professionals with the goal of continuing education has not gained much momentum. Although widely used by the postdoctoral community in the biosciences, the knowledge and expertise of coding and data science algorithms are often gained through a patchwork of informal tutorials and workshops rather than through a structured and well-designed curriculum that is tailor-made with the needs of these researchers in mind [10]. Moreover, most courses in computing education for biologists focus solely on bioinformatics without emphasis on "computational thinking" [14].

Here, we discuss the design and development of an introductory course in coding and data science for postdoctoral researchers (PDRs) in the biosciences. The goal of this course was not only to impart foundational knowledge in coding, computing, and data science, but also to develop computational thinking skills and engage the enrolled PDRs in experiential learning [15] by solving problems using datasets from their own research. For enhanced learning outcomes, we also leverage several evidence-based active learning methodologies, including collaborative learning [16], team-based learning [17], [18], and peer-to-peer learning [19]. Inspired by several prior studies that show the effectiveness of continuing education through postdoctoral training courses and programs [20]–[24], we particularly emphasize the importance of continuing education for PDRs through this course, which helps them develop skills that are gradually becoming essential in both the research and professional world.

## II. Design & Development of the Course

The introductory course in coding and data science described here was offered as a part of a postdoctoral certificate training program that a PDR can undertake over several semesters in a public health science university. The curriculum had to be designed from scratch as no previous reports or publications detailing similar courses were found by the instructor. Further, the overwhelming challenge in developing curricula for PDRs is to keep the time constraint in mind – most PDRs will not be able to take a rigorous course in the traditional style with the conventional assessment and evaluation mechanisms. This difficulty necessitates the design and development of curriculum for this specific target audience from the ground up.

### A. Goals & Objectives

The overarching goal of the course was to develop computational thinking skills in the enrolled researchers through learning how to code and understanding the fundamentals of broad categories of data science problems and applications. It was envisaged that by nurturing and honing computational thinking skills, the researchers would then be able to learn other computational methods, algorithms, and programming languages themselves with minimal effort and guidance, thus enabling a self-sustaining learning journey in computing. In particular, the design of the course was inspired by the "growth mindset" [25], which aims to create a scenario where students can develop their abilities and improve their skills over time, often even after a course has ended. The objectives of designing the course are discussed in detail here.

The primary objective of the course was to introduce researchers to coding through the use of a language that is commonly used by data science practitioners in applied domains. Different iterations of the course used either R or Python to help the enrolled researchers acquaint themselves with the fundamental concepts, ideas, and best practices in coding and software development. We hope that, in the long term, the knowledge of the fundamental concepts of coding would enable researchers to develop their own code base, software, library, or package, which would, in turn, help other researchers to analyze their data.

Our second objective was to introduce the enrolled PDRs to some of the broad categories of data science and machine learning problems along with some relevant algorithms, tools, and methodologies to solve these problems. While researchers are familiar with the questions in their respective research areas, often translating those problems into the appropriate data science questions requires a data science background and perspective. Our aim, in this regard, was to acquaint the researchers with the definitions and the underlying principles of broad machine learning problem categories, such as exploratory data analysis, classification, regression, and clustering, so that they could translate the questions from their individual research areas into these categories.

Another major objective of the course was to engage the researchers in experiential learning. All theoretical concepts and introduction to computing methodologies were supplemented by example applications from the biosciences. These exercises allowed the enrolled scholars to gain firsthand experience in applying algorithms, concepts, and tools to a problem in their area of research. Moreover, working on a final project (that we describe in detail later) allowed the researchers to employ the methods discussed in the class to the datasets from their individual research.

Finally, one of the objectives in the design and development of the course was to improve learning outcomes through the application of several methods for active learning while working in groups. For example, forming teams of researchers to solve a team-based project was inspired from prior use of collaborative learning [16], team-based learning [17], [18], and peer-to-peer learning [19]. In addition, following the best practices in computing education, in-class sessions included several exercises that implemented strategies for enhancing critical thinking skills in computing, such as pair programming [26]–[28] and think-pair-share [29] sessions.

### B. Learning Outcomes

The learning outcomes of the course are presented here. By the end of the course, the enrolled PDRs should be able to:

- write code in R/Python for their research needs
- appreciate the use of data science for their individual research problems
- understand how to load and preprocess data using R/Python
- apply exploratory data analysis tools and interpret the results using R/Python
- understand the fundamental concepts of classification, regression, and clustering
- identify which data analysis tools to use for their specific research needs
- perform experiments involving classification, regression, and clustering on their own data using R/Python libraries

### C. Prerequisites & A Sister Course

This course assumed no prerequisite knowledge except for a college-level familiarity with the fundamental concepts of statistics. It was noted that this level of familiarity with statistics is generally possessed by PDRs in the biosciences for the design of experiments and statistical analyses of their results. However, keeping in mind that some PDRs might need to refresh their knowledge in statistics, a sister course titled "Statistics for Biomedical Researchers" was also offered.

### D. Curriculum

Several courses have been developed with a focus on computing education for students in the biosciences [12], [13]. However, we are not aware of a computing course that covers the fundamentals of coding and data science for PDRs in the biosciences. Consequently, the curriculum had to be developed from scratch with a particular focus on the needs of the postdoctoral community. As a result, there is no literature on the design and development of curricula for a course with these specific requirements. Here, we elaborate on the details of the curriculum design of our course and how the needs of the researchers who enrolled in this course were accommodated.

*1) Coding:* The first half of the course was dedicated entirely to lectures, sessions, and exercises on coding. It was assumed that none of the enrolled researchers had any prior knowledge or experience of coding. Also, keeping in mind the needs and the application areas for this community, the coding module of the curriculum was designed to be taught using R or Python, the two most common programming languages or software environments used by researchers in the biosciences for analyzing their data.

Besides demonstrations and code examples in the classroom, the enrolled researchers were encouraged to participate in pair programming exercises in the classroom to develop their coding skills and reap the benefits of collaborative learning and peer-to-peer learning. Additionally, optional homework assignments were provided for practice.

The topics covered included the basics of coding, common data structures, loops, functions, and the use of libraries and packages. Additionally, given the nature of the course and the target audience, lectures and sessions were held focusing on how to load data into data frames and the fundamentals of handling data frames.

*2) Data Science:* The latter half of the course focused on topics in data science and the broad categories of data science problems that are encountered in research. For this half of the course, it was also assumed that the researchers have no prior knowledge of data science and data analysis tools and methods.

The data science module of the course started with data visualization tools available in R (e.g., `ggplot`, `plotly`) or Python (e.g., `matplotlib`, `plotly`). Subsequently, the researchers were introduced to exploratory data analysis that would enable them to create more effective data visualizations and derive useful insights about the data. Following these lecture sessions, the researchers were introduced to three broad categories of data science & machine learning problems that are frequently encountered in biosciences research – clustering, classification, and regression. The primary focus of these lectures was on how to translate a problem from their domain of research into one or more of these categories.

Besides in-class examples and take-home exercises, the researchers were encouraged to engage in discussions in the classroom following the think-pair-share model to facilitate better learning outcomes through active learning. In addition, they were introduced to the fundamentals of the algorithms and tools for each of these broad categories of data analysis problems.

The final few lectures were focused on how to evaluate the different machine learning models that can be trained on their data and a "floating" lecture on topics of their choice. The PDRs enrolled in the course were particularly appreciative of the floating lecture and were enthusiastic about recommending topics that they have encountered before but were not covered in the lectures. Some of the most common requests included visualization using heatmaps, biclustering, and dimensionality reduction through principal component analysis.

### E. Final Project

The PDRs enrolled in the course were split into teams of 4 to 5. Each team included researchers from different backgrounds, different areas of expertise, different genders, and different ethnicities. In this regard, we emphasized the diversity of the team members to ensure that they reap the most benefits from working collaboratively in diverse teams. Each team was encouraged to meet at least once outside of class hours every week to practice and discuss their final project.

For the final project, the team members were asked to frame data science questions using data from their own research. The objective was to use data from their respective research areas to practice the skills acquired in the classroom and to apply the tools discussed in the lectures. This resulted in enhanced learning outcomes and increased learner motivation and engagement throughout the duration of the course.

On the last day of class, each team presented the findings and results from their final project before their peers. Following each presentation, we facilitated and encouraged discussions meant to enhance the researchers' understanding of the applications of different algorithms to problems in their research domain.

### F. Assessment

The target audience and the nature of the course did not allow for the evaluation of the performance of the enrolled researchers through letter grades. Since the focus was on continuing education with an emphasis on learning coding and data science methods for application to their own research, the PDRs were motivated to learn and participate in class. Feedback from the instructor and peers on the final project of every team was the only assessment that was performed.

### G. Choice of Software

Based on polls and the interest of the community gauged through interactions with the postdoctoral association at the university, the course was offered either in R or Python.

While the first two times the course was offered using R, the third iteration of the course was offered using Python as the programming language.

## III. Feedback & Lessons Learned

The course was launched for the first time in Spring 2019 and continues to be offered to date. Due to the popularity of the course among the postdoctoral community, the course was even offered over the summer. Initially, the course was designed and taught by a PDR with a doctoral degree in computer science and now continues to be taught by senior Ph.D. students in biomedical informatics. It should be noted that while it was started as an in-person course, the mode of instruction was switched to online in the middle of Spring 2020 due to the COVID-19 restrictions.

### A. Feedback

At the end of every semester, feedback was sought from the enrolled PDRs to improve the quality of the course and understand the evolving needs for future semesters. There were two sections in the questionnaire – one section asking for responses on a Likert scale and another section seeking free-text responses to some questions.

*1) Responses on Likert scale:* Responses were collected to questions using a Likert scale of 4 ("Strongly Disagree" "Disagree", "Agree", and "Strongly Agree") as well as open-ended questions on improving the course. The questions included in the questionnaire were:

A The instructor was organized and used class time efficiently.
B The instructor presented course material in a clear manner that facilitated understanding.
C The course supplemented my laboratory training/research.
D The course was worth my time out of the lab.
E My postdoc mentor was supportive of my participation.
F I would encourage other postdoctoral fellows to participate in this course.
G The group project was useful for learning course material.
H Overall, this was a good course.

The survey questions were designed to evaluate a number of different components of the course. In general, the survey questions can be divided into the following categories:

1) Questions A and B were designed to evaluate the effectiveness of the instructor in teaching the course.
2) Questions C and D were designed to evaluate whether the course was useful for the PDRs in terms of application to their research.
3) Question E evaluated whether postdoc mentors are supportive of the postdocs taking time away from their laboratory work to attend the classes.
4) Question F and H evaluated the overall usefulness of the course.
5) Question G was designed to gauge the usefulness of the final project in the learning process.

There were 55 responses to these survey questions over 3 semesters.

In addition, the following question was added in Spring 2020 to evaluate whether the shift from an in-person to an online mode of instruction affected the learning:

I Moving the course to an online platform did not impact my learning of the material.

This question had 18 responses from 1 semester.

The feedback for all 3 semesters was aggregated and percentages of responses in each of the 4 values on the Likert scale were computed. The aggregated results are shown in Figure 1. We noted overwhelmingly positive responses to all the questions in the survey. In particular, we observed that not only did the PDRs find the course useful (question H) and that they would recommend other postdoctoral fellows to take this course (question F), it was also clear from the survey results that the course supplemented the laboratory work (question C) and was worth their time spent outside of the laboratory (question D). It was also seen that mentors of PDRs supported their endeavors in taking a course on coding and data science (E). Further, the positive responses to question G demonstrate the effectiveness and beneficial impact of the group project towards improved learning outcomes. The overwhelmingly positive feedback on the instructor's role (questions A and B) indicate that meticulous planning in the design of the curriculum along with the implementation of active learning techniques in the classroom was well-received by those who took the course.

*2) Free-text responses:* In addition, the enrolled PDRs also provided feedback through a free-text section in the feedback form. This section had three questions:

A The major strengths of this course: The consensus in the responses to this question revolved around the following themes:
- The use of practical real-world examples and datasets from the biosciences used in the classroom for experiential learning
- The usefulness of learning a programming language like Python or R
- The insights and understanding of how to combine biology and computer science
- The instructor's patience, teaching style, availability, and approachability, leading to better-than-expected learning outcomes
- Learning useful tools for data analysis and visualization
- The usefulness of group projects
- Breadth of topics covered
- Appreciation for the optional practice problems

B The major weaknesses of this course: While responses to this question were limited, we noted the following themes mentioned in the responses:
- Difficulty in learning programming
- Requirement of some self-study and practice outside the classroom

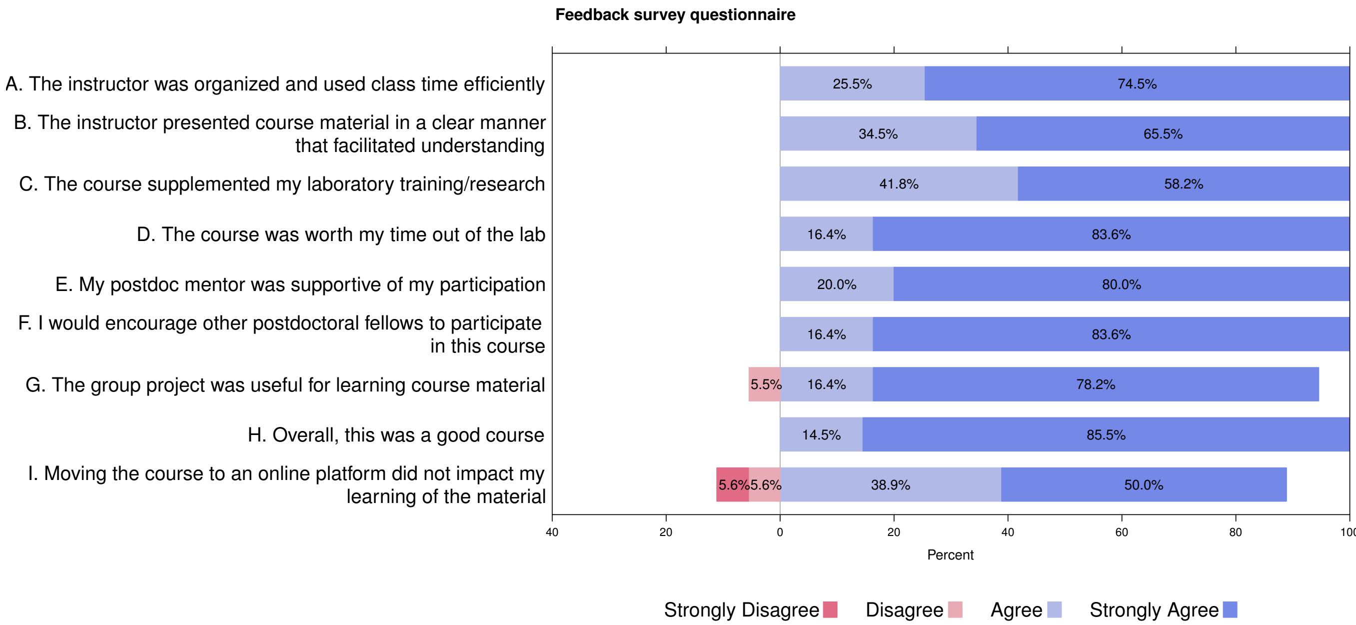


Fig. 1. Visualization of the survey responses on a Likert scale of 4 responses - "Strongly Disagree", "Disagree", "Agree", and "Strongly Agree". Questions A to H had 55 responses over 3 semesters and question I had 18 responses from 1 semester. The survey responses show positive feedback on all survey questions.

- Preference for an individual project instead of a group project
- Virtual nature of the class when the COVID-19 pandemic started
- Short duration of the class time

C Suggestions for improvement of this course: We received several suggestions for improvement, most of which were implemented in the subsequent semesters when the course was offered again. Here, we summarize the most common themes in the suggestions for improvement:

- More practice problems (implemented, see Section III-B2)
- More rigorous final project (implemented, see Section III-B2)
- On-demand lecture on certain topics (implemented in the form of a "floating lecture", see Section III-B4)
- More help in framing the final project (implemented, see Section III-B5)
- More help with software installation (implemented, see Section III-B7)
- Replacing guest lectures in the curriculum with more hands-on learning experience (implemented, see Section III-B6)
- Spreading the course material over two semesters in two different courses
- Better explanation of programming jargon that is widely encountered in online resources
- Including a lecture on data cleaning and pre-processing

### *B. Lessons Learned*

Drawing from our experiences in offering this course over three semesters, we discuss the lessons learned from developing this novel and unique course. These are based on discussions with the enrolled PDRs during and after the classes.

*1) Time Constraints for PDRs:* It is widely known in the PDR community that time is scarce and any time spent outside the laboratory during a work schedule is a challenge. Keeping these considerations and time constraints in mind, the coursework had to be designed in a way that ensured efficient dissemination of the subject matter without compromising on the content of the course. The course had one hour of lecture every week and flexible out-of-class contact time with the instructor. Additionally, the enrolled PDRs were encouraged to spend at least an hour every week outside of class hours for practice and working on the project to reap the greatest benefits of taking the course.

In this regard, we would also like to mention that we also observed that all survey responses indicated that the enrolled PDRs found the course worth their time out of the laboratory.

*2) Homework Exercises & A Rigorous Final Project:* As the course had no formal assessment using letter grades, one potential issue is a lack of motivation of the enrolled PDRs to participate or engage. However, we experienced entirely the opposite. When the course was first offered, there were no optional homework exercises and the team project was not as rigorous. However, according to answers to the open-ended questions in the survey, most PDRs recommended more homework exercises for practice and a more rigorous setup for the final team project. In subsequent semesters, we redesigned these aspects of the curriculum and received more positive feedback. In particular, there was overwhelmingly positive feedback on the effectiveness of the final project on the learning outcomes of the course.

*3) Using Domain-specific Datasets & Problems:* A number of examples, applications, and real-world problems were discussed in class for a better understanding of the material of the course. It was observed that using domain-specific problems and data, in this case from the biosciences, is important to keeping the scholars motivated and focused. In particular, we noted that the PDRs engaged in lively discussions in the classroom on the nature of the datasets and the insights derived from the data analysis examples.

*4) The "Floating" Lecture:* PDRs appreciated the idea of having a floating lecture session. This practice allowed the enrolled researchers to approach the instructor with data science questions they previously encountered but were not yet ready to solve using their newly-learned data analysis skills without additional assistance or supervision. Specifically, most PDRs requested a lecture on principal component analysis, specific clustering methods, such as bi-clustering, and visualization methods, like heatmaps.

*5) Framing the Final Project:* As most of the enrolled PDRs had no prior experience in computing or data science, they needed assistance with setting up suitable data science questions for their final project based on their individual data from research. In general, they needed help in translating their research questions or queries into appropriate data science questions that could be solved using the methods and algorithms discussed in the class. It was envisaged that this would eventually help them later in their research careers where coding and data science are playing an increasingly frequent role. Moreover, the objectives for the final project had to be discussed with the PDRs so that it could be completed in a feasible time frame and with reasonable effort.

*6) Guest Lectures:* For the first iteration of the course, we arranged for some guest lectures by faculty members who were doing hands-on computing and data science research in the biosciences, e.g., genomics, biomedical image analysis, natural language processing for physician notes, data mining of emergency room (ER) data, and the application of machine learning to prescription data. However, it was noticed that the enrolled PDRs appreciated more hands-on experience that would help them in their own research rather than guest lectures on broad topics unrelated to their area of research. Taking this into consideration, the guest lectures were replaced by more hands-on training on coding and applied data science using datasets that the PDRs brought from their own research domains.

*7) Help with Software Installation & Setup:* Due to no prior experience in using software and languages like R and Python, several PDRs needed help in installing and setting up the needed software on their laptops. Detailed instructions were provided before the start of the course and the part of the time on the first day of class needed to be devoted to troubleshooting and helping them to get started.

*8) Implementation of Suggestions for Improvement:* Suggestions for improvement were provided by PDRs through free-text responses in the feedback form, which helped us improve the course and make it more attuned to the needs and demands of the postdoctoral community. Several of the implemented changes were well-received and appreciated by the enrolled PDRs. We are working to incorporate the suggestions that have not been implemented yet to further enhance the experience of PDRs and improve the quality of the course.

## IV. Discussion & Recommendations

With no prior reports published on similar courses in the past, this course serves as a novel and innovative venture in computing education and its possibilities in connecting to other STEM disciplines as well as the social sciences, and offering opportunities for continuing education to postdoctoral researchers. A large pool of talented researchers in various non-computing research domains are often held back from achieving their full scientific potential due to a lack of training in computing and data science. We sought to bridge this gap and introduce the fundamental ideas of computing to motivated researchers set to become future faculty members and industry professionals. A unique combination of domain knowledge, computing, and data science would not only be beneficial to the careers of individual researchers but could also result in a more knowledgeable and productive scientific research workforce.

While the focus of this course was to nurture programming skills and introduce fundamental data science concepts to PDRs, a similar course can be designed for faculty members in disciplines that have not traditionally focused on computing education. An important component of this course could be to have guest lectures from faculty members whose research bridges the gap between computing and other domains for fostering collaborative and interdisciplinary research, e.g., how faculty members in biology and medicine can form collaborations and partnerships with faculty members in computer science.

We would like to strongly emphasize the differences between this course focusing on PDRs and other newly-introduced courses targeting undergraduate and master's students in the biosciences and other non-computing disciplines. While courses designed for undergraduates and master's students are more rigorous and have an extensive evaluation framework, it is not possible to have the same level of rigor or evaluation strategies for this course. This is due to the nature of the audience, i.e., PDRs, who have much less time to devote to the course than regular students. Consequently, this course met only once a week and assessments were done differently compared to traditional coursework. It must be emphasized that due to these significant differences, this course cannot simply be designed by modifying an undergraduate or master's level course but rather needs to be designed from scratch, keeping in mind the specific needs and the constraints of the target audience.

We further wish to highlight that while several PDRs have gained some knowledge and informal experience of programming and data science at this stage of their career through tutorials, workshops, blog posts, and educational videos, these are not a substitute for a structured curriculum designed specifically with this target audience in mind. Neither are these means of learning as effective as the knowledge gained through a tailor-made course, involving the use of active learning, collaborative learning, and experiential learning methods, taught by an instructor. Moreover, the distinct focus on developing computational thinking skills and stress on a "growth mindset" [25] in the design of the curriculum will allow for a sustained learning experience and development of skills beyond the duration of the course.

We hope that other universities and campuses adopt the methods we describe here and develop new curricula to help researchers in advanced stages of their careers through courses focused on continuing education in computing and data science. We would like to emphasize the need for one or more faculty members in universities who would be devoted entirely to these efforts of designing inter-disciplinary curricula for researchers as well as faculty members.

It must be remembered that a course like this can improve substantially through feedback and assessment of the usefulness of the course content to the target audience. For example, our course evolved slightly over time (e.g., discontinuing guest lectures) based on the expectations and needs of the postdoctoral community. It is also important to garner the support of the faculty mentors of PDRs in order to encourage supplementary education that would eventually augment several areas of research.

Finally, we recommend that partnerships be formed between computing-focused university departments and other departments that could benefit from computing education in achieving their missions. While we would emphasize that these collaborations need to reach out to students at an early stage, such as at the undergraduate level, there exists a dire need for such courses among researchers at an advanced stage of their career, such as postdoctoral researchers and doctoral candidates. The ultimate goal of these partnerships should be developing courses and providing opportunities for the self-enrichment of students and researchers to help them achieve their research and career goals.

Future work involves developing similar relevant curricula focusing on developing programming skills and cultivating data science knowledge in other disciplines, such as chemistry and geology. Further, we aim to design similar curricula targeting not just PDRs but graduate students and faculty members.

## Acknowledgment

The authors would like to thank Dr. Kevin A. Morano at UTHealth for his help and support in offering the course described here.

## References

[1] F. Provost and T. Fawcett, "Data Science and its Relationship to Big Data and Data-Driven Decision Making," *Big Data*, vol. 1, no. 1, pp. 51–59, Mar. 2013, publisher: Mary Ann Liebert, Inc., publishers. [Online]. Available: https://www.liebertpub.com/doi/full/10.1089/big.2013.1508

[2] D. Franklin, "Putting the computer science in computing education research," *Communications of the ACM*, vol. 58, no. 2, pp. 34–36, Jan. 2015. [Online]. Available: https://dl.acm.org/doi/10.1145/2700376

[3] N. R. Council, D. o. E. a. P. Sciences, C. S. a. T. Board, and C. o. F. a. t. I. o. C. a. Biology, *Catalyzing Inquiry at the Interface of Computing and Biology*. National Academies Press, Jan. 2006, google-Books-ID: sQOcAgAAQBAJ.

[4] R. C. Gentleman, V. J. Carey, D. M. Bates, B. Bolstad, M. Dettling, S. Dudoit, B. Ellis, L. Gautier, Y. Ge, J. Gentry, K. Hornik, T. Hothorn, W. Huber, S. Iacus, R. Irizarry, F. Leisch, C. Li, M. Maechler, A. J. Rossini, G. Sawitzki, C. Smith, G. Smyth, L. Tierney, J. Y. Yang, and J. Zhang, "Bioconductor: open software development for computational biology and bioinformatics," *Genome Biology*, vol. 5, no. 10, p. R80, Sep. 2004. [Online]. Available: https://doi.org/10.1186/gb-2004-5-10-r80

[5] M. D. Robinson, D. J. McCarthy, and G. K. Smyth, "edgeR: a Bioconductor package for differential expression analysis of digital gene expression data," *Bioinformatics*, vol. 26, no. 1, pp. 139–140, Jan. 2010. [Online]. Available: https://doi.org/10.1093/bioinformatics/btp616

[6] V. A. C. Schoonenberg, M. A. Cole, Q. Yao, C. Macias-Treviño, F. Sher, P. G. Schupp, M. C. Canver, T. Maeda, L. Pinello, and D. E. Bauer, "CRISPRO: identification of functional protein coding sequences based on genome editing dense mutagenesis," *Genome Biology*, vol. 19, no. 1, p. 169, Oct. 2018. [Online]. Available: https://doi.org/10.1186/s13059-018-1563-5

[7] S. J. Spielman and C. O. Wilke, "Pyvolve: A Flexible Python Module for Simulating Sequences along Phylogenies," *PLOS ONE*, vol. 10, no. 9, p. e0139047, Sep. 2015, publisher: Public Library of Science. [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0139047

[8] P. J. A. Cock, T. Antao, J. T. Chang, B. A. Chapman, C. J. Cox, A. Dalke, I. Friedberg, T. Hamelryck, F. Kauff, B. Wilczynski, and M. J. L. de Hoon, "Biopython: freely available Python tools for computational molecular biology and bioinformatics," *Bioinformatics*, vol. 25, no. 11, pp. 1422–1423, Jun. 2009. [Online]. Available: https://doi.org/10.1093/bioinformatics/btp163

[9] P. Pevzner and R. Shamir, "Computing Has Changed Biology—Biology Education Must Catch Up," *Science*, vol. 325, no. 5940, pp. 541–542, Jul. 2009, publisher: American Association for the Advancement of Science Section: Education Forum. [Online]. Available: https://science.sciencemag.org/content/325/5940/541

[10] A. M. Wright, R. S. Schwartz, J. R. Oaks, C. E. Newman, and S. P. Flanagan, "The why, when, and how of computing in biology classrooms," *F1000Research*, vol. 8, p. 1854, Mar. 2020. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6971840/

[11] B. Ekmekci, C. E. McAnany, and C. Mura, "An Introduction to Programming for Bioscientists: A Python-Based Primer," *PLOS Computational Biology*, vol. 12, no. 6, p. e1004867, Jun. 2016, publisher: Public Library of Science. [Online]. Available: https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1004867

[12] R. Libeskind-Hadas and E. Bush, "A first course in computing with applications to biology," *Briefings in Bioinformatics*, vol. 14, no. 5, pp. 610–617, Sep. 2013. [Online]. Available: https://doi.org/10.1093/bib/bbt005

[13] A. A. David, "Introducing Python Programming into Undergraduate Biology," *The American Biology Teacher*, vol. 83, no. 1, pp. 33–41, Jan. 2021. [Online]. Available: https://doi.org/10.1525/abt.2021.83.1.33

[14] A. Rubinstein and B. Chor, "Computational Thinking in Life Science Education," *PLOS Computational Biology*, vol. 10, no. 11, p. e1003897, Nov. 2014, publisher: Public Library of Science. [Online]. Available: https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003897

[15] D. A. Kolb, *Experiential Learning: Experience as the Source of Learning and Development*. FT Press, Dec. 2014, google-Books-ID: jpbeBQAAQBAJ.

[16] M. Laal and S. M. Ghodsi, "Benefits of collaborative learning," *Procedia - Social and Behavioral Sciences*, vol. 31, pp. 486–490,

Jan. 2012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877042811030205

[17] L. Michaelsen, A. Knight, and L. Fink, "Team-Based Learning: A Transformative use of Small Groups in College Teaching," *Centers for Teaching and Technology - Book Library*, Jan. 2004. [Online]. Available: https://digitalcommons.georgiasouthern.edu/ct2-library/199

[18] P. G. Koles, A. Stolfi, N. J. Borges, S. Nelson, and D. X. Parmelee, "The Impact of Team-Based Learning on Medical Students' Academic Performance," *Academic Medicine*, vol. 85, no. 11, pp. 1739–1745, Nov. 2010. [Online]. Available: https://journals.lww.com/academicmedicine/Fulltext/2010/11000/The_Impact_of_Team_Based_Learning_on_Medical.34.aspx

[19] C. Lang, A. Craig, and G. Casey, "A pedagogy for outreach activities in ICT: Promoting peer to peer learning, creativity and experimentation," *British Journal of Educational Technology*, vol. 48, no. 6, pp. 1491–1501, 2017, _eprint: https://bera-journals.onlinelibrary.wiley.com/doi/pdf/10.1111/bjet.12501. [Online]. Available: https://bera-journals.onlinelibrary.wiley.com/doi/abs/10.1111/bjet.12501

[20] M. J. Karel, V. Molinari, D. Gallagher-Thompson, and S. L. Hillman, "Postdoctoral training in professional geropsychology: A survey of fellowship graduates," *Professional Psychology: Research and Practice*, vol. 30, no. 6, pp. 617–622, 1999, place: US Publisher: American Psychological Association.

[21] R. F. Levant, J. E. N. Albino, A. B. Brown, S. A. Feldman, R. A. Folen, P. Kaczmarek, E. S. LeVine, R. E. McGrath, G. D. Pickar, A. E. Shapiro, S. R. Tulkin, and C. VanderPlate, "Training programs," in *Prescriptive authority for psychologists: A history and guide*. Washington, DC, US: American Psychological Association, 2003, pp. 117–140.

[22] J. Moye and E. Brown, "Postdoctoral training in geropsychology: Guidelines for formal programs and continuing education," *Professional Psychology: Research and Practice*, vol. 26, no. 6, pp. 591–597, 1995, place: US Publisher: American Psychological Association.

[23] G. J. Neimeyer and J. M. Taylor, "Continuing education in psychology," in *History of psychotherapy: Continuity and change, 2nd ed*. Washington, DC, US: American Psychological Association, 2011, pp. 663–672.

[24] T. Thierer, "MORE ON EDUCATION," *The Journal of the American Dental Association*, vol. 137, no. 9, p. 1215, Sep. 2006, publisher: Elsevier. [Online]. Available: https://jada.ada.org/article/S0002-8177(14)64300-2/abstract

[25] A. Rattan, K. Savani, D. Chugh, and C. S. Dweck, "Leveraging Mindsets to Promote Academic Achievement: Policy Recommendations," *Perspectives on Psychological Science*, vol. 10, no. 6, pp. 721–726, Nov. 2015, publisher: SAGE Publications Inc. [Online]. Available: https://doi.org/10.1177/1745691615599383

[26] C. McDowell, L. Werner, H. E. Bullock, and J. Fernald, "Pair programming improves student retention, confidence, and program quality," *Communications of the ACM*, vol. 49, no. 8, pp. 90–95, Aug. 2006. [Online]. Available: https://doi.org/10.1145/1145287.1145293

[27] C. McDowell, L. Werner, H. Bullock, and J. Fernald, "The effects of pair-programming on performance in an introductory programming course," in *Proceedings of the 33rd SIGCSE technical symposium on Computer science education*, ser. SIGCSE '02. New York, NY, USA: Association for Computing Machinery, Feb. 2002, pp. 38–42. [Online]. Available: https://doi.org/10.1145/563340.563353

[28] M. Celepkolu and K. E. Boyer, "The Importance of Producing Shared Code Through Pair Programming," in *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*, ser. SIGCSE '18. New York, NY, USA: Association for Computing Machinery, Feb. 2018, pp. 765–770. [Online]. Available: https://doi.org/10.1145/3159450.3159506

[29] M. Kaddoura, "Think Pair Share: A Teaching Learning Strategy to Enhance Students' Critical Thinking," *Educational Research Quarterly*, vol. 36, no. 4, pp. 3–24, Jun. 2013, publisher: Behavioral Research Press. [Online]. Available: https://eric.ed.gov/?id=EJ1061947