

# Successful Integration of Data Science in Undergraduate Biostatistics Courses Using Cognitive Load Theory

Laura Melissa Guzman,\* Matthew W. Pennell, Ellen Nikelski, and Diane S. Srivastava

Department of Zoology and Biodiversity Research Centre, University of British Columbia, Vancouver, BC V6T 1Z4, Canada

## ABSTRACT

Biostatistics courses are integral to many undergraduate biology programs. Such courses have often been taught using point-and-click software, but these programs are now seldom used by researchers or professional biologists. Instead, biology professionals typically use programming languages, such as R, which are better suited to analyzing complex data sets. However, teaching biostatistics and programming simultaneously has the potential to overload the students and hinder their learning. We sought to mitigate this overload by using cognitive load theory (CLT) to develop assignments for two biostatistics courses. We evaluated the effectiveness of these assignments by comparing student cohorts who were taught R using these assignments ( $n = 146$ ) with those who were taught R through example scripts or were instructed on a point-and-click software program (control,  $n = 181$ ). We surveyed all cohorts and analyzed statistical and programming ability through students' lab reports or final exams. Students who learned R through our assignments rated their programming ability higher and were more likely to put the usage of R as a skill in their curricula vitae. We also found that the treatment students were more motivated, less frustrated, and less stressed when using R. These results suggest that we can use CLT to teach challenging material.

## INTRODUCTION

Today, more than ever, biology graduates need to be equipped with statistical and programming skills. This is true both of students going into graduate or professional programs and those who enter the workforce upon graduation. For those entering modern graduate/professional schools, “data literacy” is an essential skill; technological advances have made possible the assembly of large and complex data sets, and a primary challenge for researchers is how to manage and make sense of this data deluge (Marx, 2013). For those entering the job market, for example, in the environmental and conservation sectors, employers list programming and statistical skills as important skills they look for in potential hires (Blickley *et al.*, 2012). More broadly, a career in “data science” is becoming increasingly attractive; the employment site Glassdoor rated data science as the best job in the United States in 2018 (Glassdoor, 2018), and being a data scientist is regarded as the “sexiest job of the 21st century” (Davenport and Patil, 2012). Even farther afield, these programming skills are not only important for jobs in data science or professional biology, but also those in “non-tech” sectors, such as marketing, engineering, finance, manufacturing, design, and healthcare (Dishman, 2016).

The increased need for data science solutions for biological data has resulted in a growing demand for customizable and reproducible approaches to statistical analyses. As a result, the programming language R, which is a free and open source, is now more commonly used in both commercial applications and academic research than

Sarah L. Eddy, *Monitoring Editor*

Submitted Feb 15, 2019; Revised Jul 29, 2019; Accepted Aug 8, 2019

CBE Life Sci Educ December 1, 2019 18:ar49

DOI:10.1187/cbe.19-02-0041

\*Address correspondence to: Laura Melissa Guzman (guzman@zoology.ubc.ca).

© 2019 L. M. Guzman *et al.* CBE—Life Sciences Education © 2019 The American Society for Cell Biology. This article is distributed by The American Society for Cell Biology under license from the author(s). It is available to the public under an Attribution–Noncommercial–Share Alike 3.0 Unported Creative Commons License (<http://creativecommons.org/licenses/by-nc-sa/3.0>).

“ASCB®” and “The American Society for Cell Biology®” are registered trademarks of The American Society for Cell Biology.

point-and-click software packages such JMP, SAS, and SPSS (Muenchen, 2017; Touchon and McCoy, 2016).

Biology education, at undergraduate *and* graduate levels, rarely provides students with the statistical and programming skills that they need for their future careers. One proposed solution to this problem, at the graduate student level, is to provide students with accelerated learning programs at the beginning of their graduate programs (Vale *et al.*, 2012; Stefan *et al.*, 2015). However, a recent study by Feldon *et al.* (2017) found that short format training courses, such as “bootcamps,” do not provide students with the desired skills. One explanation for this result is that students learn quantitative skills best when taught incrementally over a long time frame rather than intensively (Rohrer, 2015). It seems then, that a better place to introduce programming and statistical skills is at the undergraduate level (Michener and Jones, 2012). Teaching data science skills to biology undergraduates will provide them with the skills they need, not only for graduate school, but also for a demanding job market.

Given that biology undergraduates require simultaneous training in statistics and programming, the question is how this can be most effectively achieved. Teaching either statistics or programming alone is challenge enough. Both biostatistics and programming are courses in which students report high levels of anxiety, with debilitating effects on academic performance (Wilson and Shrock, 2001; Onwuegbuzie and Wilson, 2003). For example, the main predictors of student success in introductory programming courses is feeling comfortable while working on computer assignments and being able to ask questions (Wilson and Shrock, 2001; Simon *et al.*, 2006). Statistics and programming courses not only induce high anxiety in students, they also are perceived to be hard courses. Programming, for example, requires that students use both deep (understanding application of concepts) and surface (e.g., memorization of syntax) learning at the same time, and therefore students have trouble learning when instruction is primarily through lectures (Bellaby *et al.*, 2003) or when they do not have adequate support on assignments (Wilson and Shrock, 2001; Jenkins, 2002; Bellaby *et al.*, 2003). The simultaneous instruction of biostatistics and programming will only increase the cognitive load on students. One strategy for this problem is to use cognitive load theory (CLT) to design hands-on assignments (Wilson, 2018). CLT deals with how cognitive resources are distributed during learning and problem solving (Sweller *et al.*, 1990). Specifically, it explains how learning tasks induces an information processing load, and in return, how this load affects the processing of new information (Sweller *et al.*, 2019).

CLT suggests that learners have a limit in their working memory. There are three components of cognitive load: 1) Intrinsic load is the inherent difficulty of the instructional material. It is related to the number of elements that learners need to consider simultaneously to learn a particular procedure and the prior knowledge of the learner (Sweller and Chandler, 1994). 2) Extraneous load is determined by the manner in which the instructional materials are presented. Because students have limited cognitive resources, using cognitive resources to process the extraneous load reduces the available resources for the intrinsic load (Sweller, 1993). 3) Finally, the germane load is the processing and creation of mental models. The germane load can be modified by instructors through the materials presented (Paas *et al.*, 2004). By recognizing these three aspects

of cognitive load, instructors can tailor the scope and nature of their teaching so as to minimize the intrinsic and extrinsic loads while emphasizing the germane load.

We used CLT to design regular homework assignments to teach R programming in two biostatistics courses. In particular, we used three pedagogical methods based on CLT to design our assignments: the worked-example effect, in which studying worked examples results in better performance of the students (Renkl, 2005); the completion effect, in which we required students to complete partially solved problems (Paas and Van Merriënboer, 1994); and the split-attention effect, in which an integrated teaching of multiple concepts can improve learning compared with presenting the concepts separately but concurrently in a “split” format (Ayres and Sweller, 2005). We compared student cohorts who applied R using assignments based on CLT with cohorts who either applied R strictly through reference to example scripts or applied a point-and-click software. We investigated whether 1) the two cohorts were comparable in their initial interest to learn to program and initial skills in R or other programming language, 2) the students learned to use R effectively, 3) the introduction of R programming hindered the learning of statistics, 4) the students felt that they learned a useful skill, 5) the students felt positive or negative emotions when using R, and 6) the students liked the assignments and the way R was taught.

## METHODS

### Target Courses

We implemented this experiment at the University of British Columbia (Canada) with Behavioural Research Ethics Board Approval (H16-02319) in an introductory biostatistics course, Fundamentals of Biostatistics, often a third-year course (hereafter Biostatistics), and an advanced ecological statistics course, Ecological Methodology, often a fourth-year course (hereafter Eco-Methods). Biostatistics introduces the concepts of hypothesis testing, probability, experimental design, and statistical tests such as Student's *t* test, linear regression, and analysis of variance (ANOVA). Biostatistics includes three 50-minute lectures and one 2-hour optional computer laboratory per week. Eco-Methods introduces the concepts of experimental design, statistical power and sample size, mark and recapture methods, metrics of community diversity and composition, as well as statistical tests such as ANOVA, multiple regression, ordination, and clustering. Eco-Methods includes two 60-minute lectures and one 3-hour field and/or computer laboratory per week.

For each course, we had a control and a treatment term (Table 1). All courses included homework assignments, each of which was a relatively small part of the grade. The main difference between the treatment and control terms was the teaching of R using CLT in the homework assignments (see Box 1 and Appendix 1 in the Supplemental Material for the descriptions of the assignments). The assignments taught and tested the ability to apply the statistical concepts in R. In Biostatistics, we aggregated the previous homework assignments and introduced CLT for conceptual questions taken from the textbook and the R questions. In this course, we included two R questions in the midterm and the final exams. The control terms were different for each course. In Biostatistics, the students in the control term learned how to use the point-and-click software JMP. In Eco-Methods, the assignments introduced CLT in the

TABLE 1. Course structure in control vs. treatment terms<sup>a</sup>

	Biostatistics		Eco-Methods	
	Control	Treatment	Control	Treatment
Year	2016	2018	2016	2017
Total number of students	240	185	45	37
Number of students who consented and answered the survey	155	116	26	30
Response rate	65%	63%	58%	81%
Instructor	M.W.P.	M.W.P.	M.K.T.	D.S.S.
Teaching assistants	5	5	2	2
Grade breakdown	Assignments (3): 10% Homework (10): 10% Midterm exam: 30%  Final exam: 50%	Homework assignments (10): 20% Midterm exam: 30%  Final exam: 50%	Homework assignments (5): 25% Formal lab reports (two at 15% each): 30% Research proposal, group project: 10% Group project presentations: 5% Group project written report: 25% Participation: 5%	Homework assignments (7): 28% Formal lab reports (three at 11% each): 33% Research proposal, group project: 11% Group project presentation: 5% Group project written report: 21% Participation: 2%
Labs	Labs used JMP.	Labs used R.	Labs used R/Microsoft Excel.	Labs used R.
Homework assignments	Homework was conceptual problems from the textbook.	Homework assignments used R and CLT.	Homework was R scripts that they had to run on their own time and conceptual statistics.	Homework assignments used R and CLT.

<sup>a</sup>The treatment groups for both courses completed assignments designed using the ideas of CLT as homework.

presentation of R concepts and the subsequent questions to practice those R concepts. The students in the control term learned how to use R using example scripts. In all courses and terms, the in-class sessions consisted of Socratic lecturing.

Although the instructor differed between the control (2016) and treatment (2017) terms for Eco-Methods, both instructors taught from the same lecture slides. We note that, in 2015, instructor D.S.S. taught R from the same example scripts as M.K.T. in 2016, and that their teaching evaluations were comparable between these two years, suggesting that there was not a strong effect of instructor identity.

### Homework Assignments

We designed 10 homework assignments for Biostatistics and seven homework assignments for Eco-Methods. In each of these assignments, we applied CLT hoping to 1) reduce the extraneous load of students by taking advantage of the split-attention effect, the worked-example effect, and the completion effect; 2) reduce the intrinsic load of the material by managing the element interactivity; and 3) increase the germane load by scaffolding the material with self-explanation questions (see examples in Box 1 and Appendix 1 in the Supplemental Material; all materials have been submitted to CourseSource). We scaffolded across assignments in two ways: 1) we scaffolded the content across the assignments to make sure that each assignment had a lower intrinsic load, but by the end of the course, the students had skills on data wrangling, data visualization, and tests; and 2) we provided only partially completed problems when we first introduced a concept to the students.

### Surveys

We evaluated students' perceptions of their self-motivation and ability to use R using a survey at the end of the course (Appendix 2 in the Supplemental Material). Student participation in the survey was requested by L.M.G., who was not a course instructor or teaching assistant, during lecture time. We did not offer any incentive to complete the survey; therefore, it was voluntary, but it also presented no cost to the students. Survey completion was anonymous, occurred during class time, and was conducted with the instructors absent from the room. Both surveys consisted of three open-ended questions and 29 closed-response questions, of which 26 were Likert-scale items measuring different constructs (e.g., self-perception in programming proficiency at the beginning and the end of the course) and three were multiple-response questions measuring self-perception in affect. After aligning the questions to the goals presented in this study, only 18 closed-response questions remained relevant to these goals; therefore, we present only the results of these 18 questions. The survey included questions on the frequency of using R before and after the course took place, as well as attitudes toward the perceived difficulty of the course and students' emotional response to the data science material during the course.

Likert-scale items were analyzed using an ordinal logistic regression to test for differences in the response between control and treatment terms. We did 18 models, one for each Likert-scale item.

For the open-response questions, we developed hierarchical codes for each of these questions using the method described in

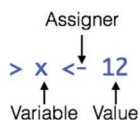
**BOX 1. Homework assignment examples**

Selected examples from the assignments showing how we used CLT to introduce R programming concepts in the statistics exercises.

**1. Reducing the extraneous load**

Split-attention effect:

Code is often presented as multiple sources of information. We incorporated the code and the explanations as a single source to reduce the split-attention effect.



Worked-example effect:

We presented worked examples of simple and complex problems, both involving how to write code and how to use code to run statistical tests. All worked examples were partitioned into different parts.

**Question:** Calculate the mean of a vector of all the integers from 1 to 50.

**First,** we must create the vector.

```
vector <- 1:50
```

**Second,** we must calculate the mean.

```
> mean(vector)
```

```
[1] 25.5
```

**Finally,** we now have our answers calculated by R. The mean of a vector from 1 to 50 is 25.5.

Completion effect:

After presenting worked examples, we presented partially completed problems in which the scaffolding was introduced in the steps to solve a question and the code needed to run a statistical test.

**Third,** construct your box plot using ggplot. Fill in the blanks in the following code to do so:

```
> ggplot (data = _____, aes(x = _____,
y = _____)) + _____()
```

**2. Reducing the intrinsic load**

We reduced the element interactivity of the material by:

- Presenting only one way to do a task. In R, every task can be done by multiple functions. While understanding these different functions is useful for more advanced programming, beginners can be overwhelmed by learning multiple functions simultaneously.
- Presenting only the functions that were needed for a given statistical test.

**3. Increasing the germane load**

In both worked examples and in partially completed problems, we asked the students to reflect on a part of the question to engage in germane load activities such as self-explaining.

Self-explanation questions:

**For you to think:** Why did you use ':' instead of 'c' to create a vector in 'vector <- 1:50'?

Guest *et al.* (2012). Two researchers, E.N. and L.M.G., generated and reviewed the codes, the themes, and the codebook (Appendix 3 in the Supplemental Material). After assigning each response to one of the codes, we counted the responses per code. For every course, we analyzed both the treatment and the control groups.

**Were the Two Cohorts for Each Course Comparable in Their Initial Skills and Interests in Either Programming or R?** To address the question of whether the students cohorts were comparable, we used the following survey questions: "My programming skills (in any programming language) were ...," "My skills in any statistical software (JMP, SPSS, etc) were ...," "Before this course started I used R...", and "Before this course started, I was interested in learning a programming language." We chose these questions, as they established base knowledge that the students had at the beginning of the term. If our cohorts differed in these base questions, we could take this into account for the following questions. For example, if we found that one cohort self-reported a higher initial proficiency in programming, we could then use these data as covariates in the model testing for differences in self-report of proficiency at the end of the course.

**Did the Students Learn to Use R Effectively?** To address the question of whether the students learned R effectively, we used both a self-reported question from the survey and two analyses

of the graphs particular to each course. For the survey question, we asked whether the students would rate their programming proficiency as "high" at the end of the course.

In Biostatistics, we evaluated whether the students learned R by analyzing whether the use of R to create graphs increased over the semester. For this analysis of the graphs in Biostatistics, we asked the students to upload and submit graphs as part of their assignments. The students were required to do nine graphs as the course progressed. For half of these graphs, the assignments walked through the code required to create a similar graph from the same data set, requiring students to only adapt this code by changing a few variables. For this set of graphs, we expected the students would use R as the main method of graphing, as they were following the example from R code. For the other half of the graphs, the students had to import the data from the textbook, then manipulate the data, and they had no walk-through on a related example graph. For this second set of graphs, the students were allowed to use any method. We then examined, for both types of graphs in Biostatistics, whether, as the course progressed, the students were more likely to use R to create their graphs over Microsoft Excel or doing them by hand. We only did this temporal comparison in the treatment group.

In Eco-Methods, we evaluated whether students learned R by analyzing whether the students customized graphs. For this analysis, we evaluated graphs produced by students for their final reports. To do these graphs, the students had to collect their own data, import the data into R, and manipulate them

into the correct format before plotting the graph. For these graphs, we assessed whether students were able to customize graphs relative to the example graph provided in the assignments. For example, we recorded whether the students changed the color, type of lines, font, background of their figures, and so on. For each graph “customization,” we assigned 1 point and added the total number of customizations per student. The students were not expected to emulate any graph, nor did they have specific customizations that were required for their final reports. The students were asked to perform two tests (one univariate and one multivariate) and to represent their results in graphs. The students could choose the software they used to do their analysis and graphs and for the presentation of their results (i.e., the type of graph) and the customization of their graphs. We compared the customization of graphs between the control and the treatment groups. To analyze the degree of customization of the graphs in the Eco-Methods lab reports, we summed the total number of customizations per person and then used a generalized linear mixed-effects model to test for differences between the courses. We used the number of customized elements per person as the response variable and the treatment as the fixed effect. Because the lab reports were done in groups of four students, the reports generated were not independent; therefore, we used the “group id” as the random effect. We used a Poisson family with a logarithmic link function.

**Did the Introduction of R Programming Hinder the Learning of Statistics?** We were also interested in determining whether the introduction of R programming would hinder the learning of statistics. Biostatistics was the only course with a final exam. We ensured that this exam had one question in common between the control and the treatment group. We then compared the scores for this question, which was a multiple-part, multiple-choice question. The students were asked how increasing the sample size affected different statistical estimates such as the SE, the SD, type II error, type I error, and power of a statistical test. This question evaluated core concepts in statistics that the students were required to understand. To analyze whether student scores on this question differed between course sections, we used a generalized linear model using a Poisson family and logarithmic link function.

**Did the Students Feel That They Learned a Useful Skill?** To determine whether students felt that, in learning R, they had learned a useful skill, we used three closed-response survey questions plus the open-response question. For the closed-response questions, we asked whether the student would put the ability to use R as a skill on their curricula vitae (CVs), whether they would continue using R in their own projects for their undergraduate or graduate school at the end of the course, and how often they used the software (R or JMP) outside class. In the open-response questions, we identified themes that were relevant to this aim.

**Did the Students Feel Positive or Negative Emotions When Using R?** We asked the students to assess their emotions toward both the conceptual parts of the course and the use of R or JMP. We transformed all positive feelings (e.g., happy and excited) into values of 1 and all negative feelings (e.g.,

frustrated and stressed) into values of 0. For these types of questions, we used a generalized linear model to test for differences in the response due to the treatment (control vs. treatment) or due to the use of R versus JMP (for Biostatistics). We used a binomial family with a logit link function. To investigate which particular feelings contributed most to this difference, we evaluated, for each feeling separately, the difference between the treatments using a chi-squared contingency test. We corrected the  $p$  values for multiple comparisons using the false discovery rate method (Benjamini and Hochberg, 1995). We excluded from this analysis all feelings that had fewer than 10 responses.

We also looked at the open-response questions and identified themes that were relevant to this aim.

**Did the Students Like the R Assignments and the Way R Was Taught?** We looked at the open-response questions and identified codes that were relevant to this aim.

All analyses were done using the R programming language (R Core Team, 2016). Ordinal regressions were done using the MASS package (Venables and Ripley, 2002).

## RESULTS

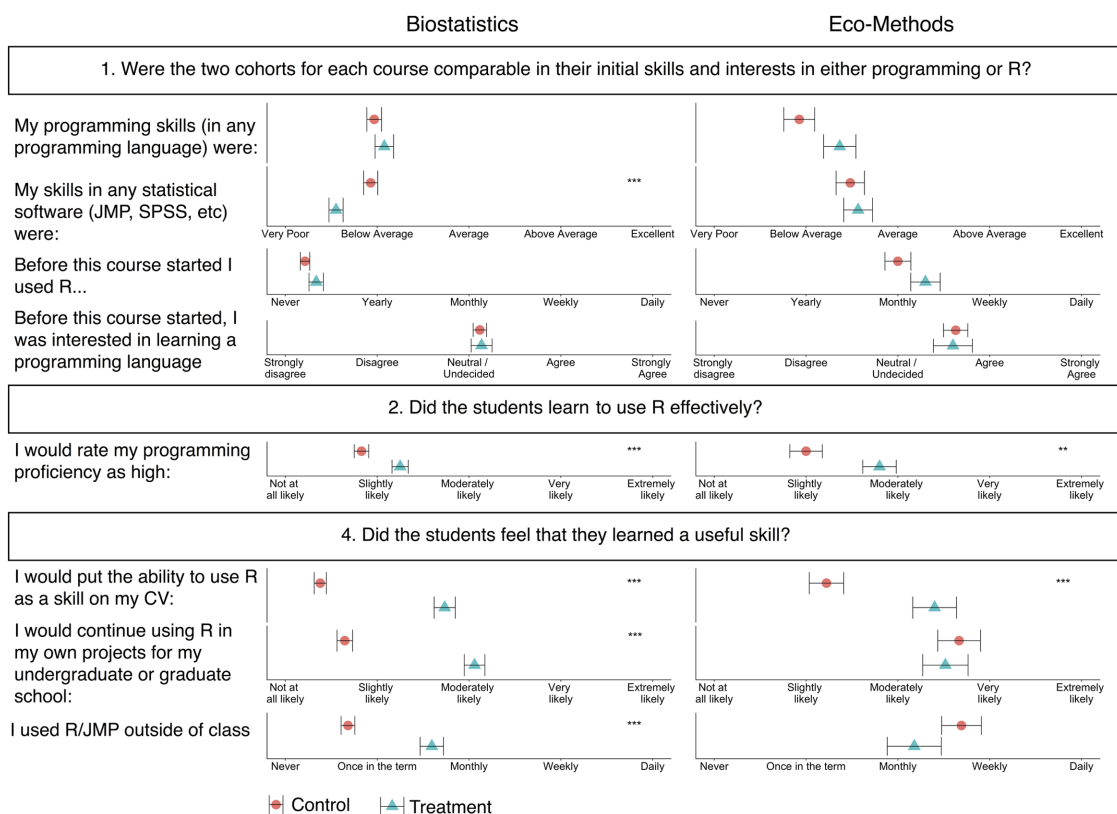
### Were the Two Cohorts for Each Course Comparable in Their Initial Skills and Interests in Either Programming or R?

Students in the control and the treatment cohorts, for both courses, rated similarly their initial programming skills in any language (Biostats:  $\beta = 0.15$ , SE = 0.23,  $p$  value = 0.50, mean control = 1.96, mean treatment = 2.08; Eco-Methods:  $\beta = 0.86$ , SE = 0.49,  $p$  value = 0.08, mean control = 1.93, mean treatment = 2.37; Figure 1), whether they had used R before (Biostats:  $\beta = 0.36$ , SE = 0.35,  $p$  value = 0.31, mean control = 1.21, mean treatment = 1.34; Eco-Methods:  $\beta = 0.78$ , SE = 0.62,  $p$  value = 0.21, mean control = 3, mean treatment = 3.3; Figure 1), and whether they had any interest in a programming language before (Biostats:  $\beta = 0.13$ , SE = 0.23,  $p$  value = 0.59, mean control = 3.12, mean treatment = 3.14; Eco-Methods:  $\beta = 0.16$ , SE = 0.49,  $p$  value = 0.75, mean control = 3.63, mean treatment = 3.6; Figure 1). The Biostatistics students in the control cohort rated their skills on any statistical software higher than the treatment students ( $\beta = -0.83$ , SE = 0.24,  $p$  value = 0.0005, mean control = 1.93, mean treatment = 1.55; Figure 1), while the Eco-Methods students rated their skills similarly ( $\beta = 0.17$ , SE = 0.49,  $p$  value = 0.74, mean control = 2.48, mean treatment = 2.57; Figure 1).

### Did the Students Learn to Use R Effectively?

In Biostatistics and Eco-Methods, by the end of the course, the students who used the R assignments designed with CLT self-rated a higher proficiency in R than the control students (Biostats:  $\beta = 0.86$ , SE = 0.23,  $p$  value = 0.0001, mean control = 1.83, mean treatment = 2.25; Eco-Methods:  $\beta = 1.50$ , SE = 0.51,  $p$  value = 0.003, mean control = 2, mean treatment = 2.8; Figure 1).

The students in the treatment group of Biostatistics had to do two types of graphs: graphs from data provided in the textbook, for which they had to input and graph the data without an example, and graphs from data provided in the labs, for which the data were already formatted and easy to input and the graph was based on an example. We found that, when the students had an example, they were able to produce a graph

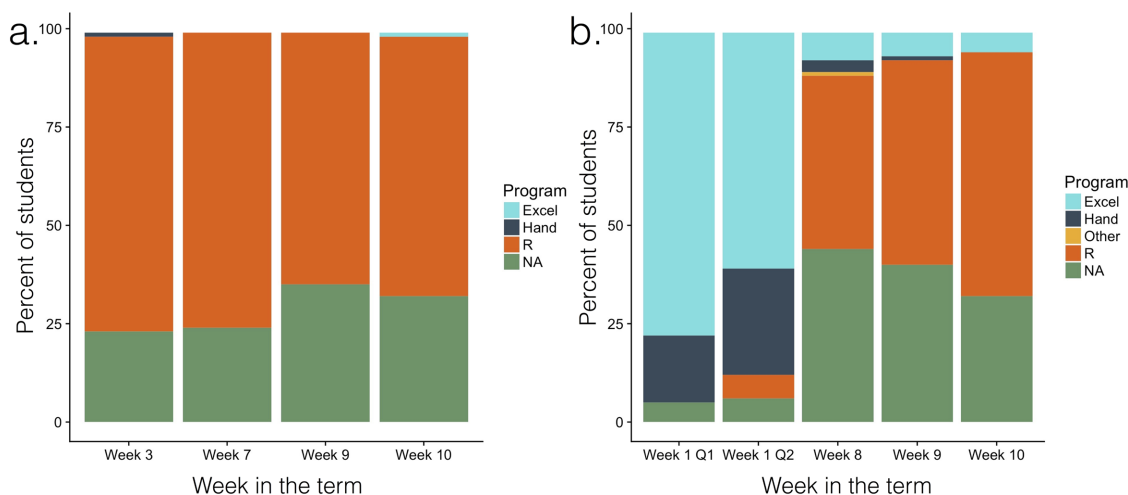


**FIGURE 1.** Students responses to the survey questions in relation to teaching treatment (control vs. CLT treatment) and course identity. Student responses are ranked on a Likert scale. Points and bars represent means and SEs respectively. Control groups are colored red, and treatment groups are colored blue. Significance is noted with asterisks: \*\*,  $p < 0.01$ ; \*\*\*,  $p < 0.001$ .

of their data using R from the beginning of the term (Figure 2a). However, without an example, we found that, at the beginning of the term, they used Microsoft Excel or drew the graph by hand, but by the end of the term, the majority of the

students were able to input and graph their data using R (Figure 2b).

In the analysis of the graphs in Eco-Methods, we found that students in the control and treatment cohorts did not



**FIGURE 2.** (a) Percentage of students in Biostatistics in the treatment cohort who made their graphs using R from previous assignment examples is high from the beginning of the course. (b) Percentage of students in Biostatistics in the treatment cohort who made their graphs using R from the textbook increases as the term progresses and replaces use of Microsoft Excel ("Excel") or hand drawing ("Hand") or other software. "NA" indicates students who did not submit either their homework assignments or this particular question from the homework assignments. The students who did not submit their homework assignments are not the same across all weeks.

differ in the number of customized elements on their graphs ( $\beta = -0.31$ ,  $SE = 0.24$ ,  $p$  value = 0.19, project group as random effect).

### Did the Introduction of R Programming Hinder the Learning of Statistics?

The treatment and control cohorts of Biostatistics did not differ in their scores on the same question in the final exam ( $\beta = 0.05$ ,  $SE = 0.06$ ,  $p$  value = 0.42, mean control = 3.99, mean treatment = 4.20).

### Did the Students Feel That They Learned a Useful Skill?

In Biostatistics and Eco-Methods, we found that the students in the treatment group were more likely to put R as a skill in their CVs (Biostats:  $\beta = 2.55$ ,  $SE = 0.27$ ,  $p$  value =  $3.79 \times 10^{-20}$ , mean control = 1.38, mean treatment = 2.74; Eco-Methods:  $\beta = 1.86$ ,  $SE = 0.54$ ,  $p$  value = 0.0005, mean control = 2.22, mean treatment = 3.4; Figure 1). The students in Biostatistics in the treatment group were also more likely to continue using R in their future graduate and undergraduate studies ( $\beta = 2.30$ ,  $SE = 0.26$ ,  $p$  value =  $7.91 \times 10^{-19}$ , mean control = 1.65, mean treatment = 3.06; Figure 1). In Eco-Methods, the treatment and control students did not differ in how likely they were to continue to use R for their own projects, but both were between moderately and very likely to continue using it ( $\beta = -0.18$ ,  $SE = 0.48$ ,  $p$  value = 0.71, mean control = 3.67, mean treatment = 3.52; Figure 1). In Biostatistics, we found that students who used R were more likely to use the software outside class than students who used JMP ( $\beta = 1.37$ ,  $SE = 0.24$ ,  $p$  value =  $1.34 \times 10^{-08}$ , mean control = 1.68, mean treatment = 2.59; Figure 1). In Eco-Methods, the treatment and control students did not differ in how often they used R outside class, but both used R either monthly or weekly ( $\beta = -0.48$ ,  $SE = 0.53$ ,  $p$  value = 0.36, mean control = 3.69, mean treatment = 3.18; Figure 1).

In the open-response question for Biostats students, in which we asked the students what would they change about the way the course was taught, we identified the theme “Course should use other software (theme A),” which had 47 responses in total (control = 47 out of 157 students, 30%; treatment = 0 out of 117 students, 0%). In Biostatistics, the control cohort learned JMP in the labs, whereas the treatment cohort learned R. All of the responses in this category came from students who learned JMP. Of these students, 30% wanted to use another software; the students mentioned both R and Excel in their answers.

### Did the Students Feel Positive or Negative Emotions When Using R?

We found that both the Biostatistics and the Eco-Methods students in treatment cohorts had more positive feelings than the students who were taught JMP (Biostatistics) or R (Eco-Methods) traditionally (Biostats:  $\beta = 0.94$ ,  $SE = 0.19$ ,  $p$  value =  $6.88 \times 10^{-07}$ , mean control = 0.25, mean treatment = 0.46; Eco-Methods:  $\beta = 1.11$ ,  $SE = 0.33$ ,  $p$  value = 0.0008, mean control = 0.26, mean treatment = 0.53). Specifically, the students in the Biostatistics treatment cohort felt more excited, happy, motivated, proud, and less bored, and the students in the Eco-Methods experimental cohort felt less frustrated (Table 2).

**TABLE 2. Treatment students in Biostatistics significantly felt less bored and more excited, happy, motivated and proud than control students, while treatment students in Eco-Methods felt less frustrated than control students<sup>a</sup>**

Emotion	Biostatistics		Eco-Methods	
	$\chi^2$	$p$ value	$\chi^2$	$p$ value
Angry	0.02	0.98		
Annoyed	0.005	0.98	0.73	0.59
Anxious	0.55	0.62	1.45	0.45
Bored	6.52	0.03**		
Excited	17.94	<<0.001***	0.20	0.66
Frustrated	0.51	0.62	10.89	0.01**
Happy	8.92	0.009***		
Motivated	30.04	<<0.001***	3.73	0.16
Overwhelmed	1.05	0.50	1.33	0.45
Proud	10.66	0.005***	0.26	0.66
Scared	0.001	0.98		
Stressed	3.52	0.13	5.24	0.10
Supported	1.53	0.40	0.29	0.66

<sup>a</sup> $\chi^2$  and  $p$  value of the  $\chi^2$  test are given for each emotion. The  $p$  values were corrected for multiple comparisons using the false discovery rate. Blank cells had fewer than 10 responses. \*\*,  $p < 0.01$ ; \*\*\*,  $p < 0.001$ .

### Did the Students Like the R Assignments and the Way R Was Taught?

The survey had an open-response question, asking the students what would they keep about the way that R was taught. In Biostatistics, we identified a theme wherein the students suggested, “Keep some part of the canvas R assignments (theme K),” which had 49 responses (42% of the students; control = 0 out of 157 students, 0%; treatment = 49 out of 117 students, 42%). In particular, 18 students suggested we keep the walk-throughs; 12 students, the detailed instructions; five students, the step-by-step questions; and three students, the fill-in-the blank questions, the expected codes, and graphs. Four students mentioned the assignments were informative and not overwhelming. In Eco-Methods, we identified multiple themes related to our aim. First, the students suggested that “R was taught well (theme C),” which had 21 responses (control = 4 out of 27 students, 15%; treatment = 17 out of 30 students, 57%). Second, the students also suggested that “they liked having an R Workshop (theme E),” which had 19 responses (control = 15 out of 27, 56%; treatment = 4 out of 30, 13%). Here, we found that the students liked having the first in-lab session devoted to learning to start using R, which occurred for both the treatment and the control groups. Third, the students also suggested that they “liked the R/stats assignments (theme I),” which had 16 responses (control = 2 out of 27, 7%; treatment = 14 out of 30, 47%). Fourth, the students mentioned that they were “grateful to have learned R (theme F),” which had five responses (control = 2 out of 27, 7%; treatment = 3 out of 30, 10%). Finally, the students found that “learning packages/analyses/functions was useful,” which had four responses (control = 2 out of 27, 7%; treatment = 2 out of 30, 6%).

## DISCUSSION

Overall, we found that students not only learned to use R, but also that they themselves felt that this was a valuable skill and were motivated when working on the assignments. In this

study, we did two major interventions. First, we compared two cohorts who used either R (using CLT) or JMP (point-and-click software) in Biostatistics. Second, we compared two cohorts who used either R in a traditional way or R using CLT in Eco-Methods. Our studies support the idea that R can be introduced into biostatistics courses and that CLT seems to be a valuable method to introduce programming into biostatistics. In general, the results from both of our courses support the idea that introducing R in any way (either a traditional way or using CLT) is beneficial to the students. We found no evidence that the students who used R performed worse on the question on the final exam (in Biostatistics), suggesting that their learning of statistics was equivalent. However, we only used one question on the final exam. Future studies should study this question more thoroughly, as the main limitation of introducing R programming into biostatistics is the potential negative effect it could have in the learning of statistics.

Our assessment of the effect of using cognitive theory *per se* provided mixed evidence. On one hand, students in the Eco-Methods class who worked through assignments based on CLT felt less frustrated and were more likely to rate their programming proficiency as high than students who were taught R in more conventional ways (though we could not distinguish with our data alone which concept of CLT had the biggest influence). On the other hand, there was no difference in the learning of R between students taught R using CLT and those taught R using other techniques. There are two possible explanations for these conflicting results: It may be that concepts from CLT made the learning experience less emotionally taxing, but final learning outcomes did not differ, or that our measure of the learning outcomes was not sufficiently nuanced. In our study, we used the ability to input, arrange, and graph data as a measure of proficiency but, of course, recognize that this is a rather limited subset of tasks and a more expansive definition of proficiency might have yielded differences between the CLT group and the conventional methods groups in R skills. We are unaware of an agreed-upon standard for what a novice, intermediate, and advanced R user should know, and we have not yet found a data science concept inventory; we suggest developing this should be a high priority for researchers in the field of data science education.

Overall we found that students appreciated learning R, regardless of the format in which it was taught. For example, a student from the Biostatistics control group (which used the JMP program) wrote, “I wish I learned R because it seems more relevant to my degree and I wish it was part of homework and assignments” (C97).” As well, those students who were taught R generally felt it was valuable; one student wrote: “[I am] glad [I] learned R, as [I]’ve heard it’s very useful in biology especially” (E52, Biostatistics). Another student thought the course could be improved by adding even more R into the class as this was “probably the most useful part of this course moving forward” and “would have liked more assignments that required more problem solving” (E113, Biostatistics).

Self-determination theory states that there are multiple sources of extrinsic motivation (Ryan and Deci, 2000). When a student identifies the value or utility of a task, the extrinsic goal is self-endorsed and thus adopted. Identifying the utility of task is a form of extrinsic motivation that has been associated with greater engagement and performance and higher-quality learn-

ing, among other outcomes (Ryan and Deci, 2000). If the students perceived using R as a useful skill for their future jobs or for their careers, this could provide another source of motivation for them to learn R. From the surveys, we found that the students were more likely to put the ability to use R as a skill in their CVs. Future studies should test this assumption directly and assess whether the students find R as a useful skill and whether this is another source of motivation.

Regarding the student’s affect, we found that the students reported feeling more motivated when learning R than when learning JMP. Additionally, we found that the students felt more positive when using the treatment assignments to learn R than when either learning JMP or using only scripts to learn R. Having a positive affect toward learning can be important, because negative affect can have metacognitive effects, such as feelings of difficulty (Efklides, 2017). For example, a negative mood can increase the self-reported difficulty in math problem solving (Efklides and Petkaki, 2005). Specifically, we found that the students who used the R assignments in Biostatistics felt more excited, happy, motivated, proud, and less bored than the students who used JMP. In Eco-Methods, the students who used the CTL-based R assignments felt less frustrated than the control students who used the R scripts. Previous studies have found that, when teaching novice students, boredom and frustration were negatively correlated with learning, while transitioning between confusion and engagement were positively correlated with learning (Bosch and D’Mello, 2015). Our measure of affect was not continuous throughout the term; future studies could therefore measure affect more frequently to see whether frustration happens at key parts of the term or is evenly distributed.

The positive affect response may be due to the students liking some elements of CLT that we introduced in the assignments. For example, when we asked the students what they would keep about the way the software was taught, they wrote that they liked how the assignments “walked you through the questions almost step-by-step” (E3), how “everything was broken down and explained to a very basic level [as] it made it very enjoyable to learn for someone who really struggles with computer programming” (E19), and how the instructions “made sure your code was right and gave hints too if you were on the right track” (D14). Consistent with the principles behind CLT, we also found that the design of the assignments influenced whether students perceived that they were able to be successful. For example, one student wrote, “I liked the fill-in-the-blanks especially the question with the expected graphs because I could test it out and it gave me some sense of support” (E74); and another wrote, “[I] really liked how the instructions walked us through the process so it was less overwhelming” (D7).

CLT has been used successfully in a variety of courses. For example, Mason *et al.* (2016) used CLT to redesign a course in database systems. They found that the failing rate of mid- to lower-performing students was reduced by 34% after the redesign on identical final exams. Student satisfaction also increased, and feedback was very positive (Mason *et al.*, 2016). Similarly, on an advanced web applications course for graduate students, CLT was used to develop an online programming tool, and researchers found that students performed best when they were able to view examples of code during the learning of new material (Heo and Chow, 2005). When CLT was applied to teaching math to middle school students, researchers found

that student performance was improved by signaling important information, improving the aesthetic of item organization, and removing extraneous content (Gillmor *et al.*, 2015). Previous studies on teaching programming to novice learners have also found that using CLT led to better learning as well as an increase in self-efficacy and reduction in the perception of difficulty (Mason and Cooper, 2013).

When we designed the assignments, we included multiple types of scaffolding, including procedural scaffolding (helps the learners use appropriate resources as well as tools) and metacognitive scaffolding (helps the learners to reflect about what they are learning). Metacognitive scaffolding and self-questioning have been shown to support student learning of programming (Nurulain *et al.*, 2017).

We also note that, while the students improved substantially over the course of a semester, they were still far from mastering programming skills, and this is reflected in students' self-assessments. This is consistent from assessments of the very popular "boot camp" format courses and workshops (Feldon *et al.*, 2017); consistent with our experiences learning R, it appears that a single or concentrated course is unlikely to facilitate students becoming truly proficient. We suggest that a more productive strategy for teaching data science concepts would be to scaffold them throughout a university curriculum such that students are continually exposed to them in a structured and coherent manner.

### Limitations

The perceptions expressed by the students may not be generalizable to a larger population. The students who were surveyed were those present on the last day of class, which may reflect a more motivated subset of the class. Furthermore, many students in the University of British Columbia biology program who take these classes are interested in medical school or graduate school, and this motivation may not extend to students situated in other environments or those enrolled in other programs. We used surveys to assess the previous skills in programming and one component of learning R; these results are based on student self-reports. Self-reports are known to have validity concerns (Fan *et al.*, 2006), and the students in our study may have altered their responses because they knew the main purpose of the study from the consent forms. This study used only one final exam question to assess whether introducing R into biostatistics hindered the learning of R. While this question was an important concept in biostatistics, it does not measure all the concepts in biostatistics. This study was unable to control for the possibility of temporal differences in either course or that instructors differed in teaching ability (Eco-Methods). We view these explanations less likely, as similar effects were seen in both courses.

### CONCLUSIONS

This is the first evidence, to our knowledge, that using CLT increased learning success for the introduction of data sciences practices and the integration of programming and biostatistics, based on two courses in an biology undergraduate program. Each course teaches different concepts in biostatistics, but we found congruent results in terms of affect and performance of the students. The findings presented here suggest that data science is of interest to students, and CLT can be useful in introducing programming not only in biostatistics but also in other

courses. Even though we designed these assignments with biology students (and novice programmers) in mind, other disciplines will face the same data-heavy method demands and challenges of having to teach quantitative skills to novice undergraduate students. We think that these methods can be applied to other disciplines with discipline-specific examples.

### ACKNOWLEDGMENTS

This work is supported by the Public Scholars Initiative from the University of British Columbia awarded to L.M.G. and the Centre for Integrative Research Teaching and Learning. L.M.G. is also supported by Natural Sciences and Engineering Research Council of Canada CGS-D and UBC Four Year Fellowships. D.S.S. and M.W.P. are supported by NSERC Discovery Grants. We thank Alejandra Echeverri and Sarah Otto for comments on the article. We are very grateful to all the students who participated in this study.

### REFERENCES

- Ayres, P., & Sweller, J. (2005). The split-attention principle in multimedia learning. In Mayer, R. E. (Ed.), *The Cambridge handbook of multimedia learning* (pp. 135–146). New York: Cambridge University Press.
- Bellaby, G., McDonald, C., & Patterson, A. (2003). Why lecture? In *4th Annual LTSN-ICS Conference*, NUI Galway, Ireland.
- Benjamini, Y., & Hochberg, Y. (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society, Series B, Statistical Methodology*, 57(1), 289–300.
- Blickley, J. L., Deiner, K., Garbach, K., Lacher, I., Meek, M. H., Porensky, L. M., ... & Schwartz, M. W. (2012). Graduate student's guide to necessary skills for nonacademic conservation careers. *Conservation Biology*, 27(1), 24–34.
- Bosch, N., & D'Mello, S. (2015). The affective experience of novice computer programmers. *International Journal of Artificial Intelligence in Education*, 27(1), 181–206.
- Davenport, T. H., & Patil, D. J. (2012). Data scientist: The sexiest job of the 21st century. *Harvard Business Review*, 90(10), 70–76, 128.
- Dishman, L. (2016). Why coding is still the most important job skill of the future. *Fast Company*, June 14, 2016. Retrieved January 21, 2019, from [www.fastcompany.com/3060883/why-coding-is-the-job-skill-of-the-future-for-everyone](http://www.fastcompany.com/3060883/why-coding-is-the-job-skill-of-the-future-for-everyone)
- Efklides, A. (2017). Affect, epistemic emotions, metacognition, and self-regulated learning. *Teachers College Record*, 119(13), 130305.
- Efklides, A., & Petkaki, C. (2005). Effects of mood on students' metacognitive experiences. *Learning and Instruction*, 15(5), 415–431.
- Fan, X., Miller, B. C., Park, K.-E., Winward, B. W., Christensen, M., Grotevant, H. D., & Tai, R. H. (2006). An exploratory study about inaccuracy and invalidity in adolescent self-report surveys. *Field Methods*. <https://doi.org/10.1177/152822x06289161>
- Feldon, D. F., Jeong, S., Peugh, J., Roksa, J., Maahs-Fladung, C., Shenoy, A., & Oliva, M. (2017). Null effects of boot camps and short-format training for PhD students in life sciences. *Proceedings of the National Academy of Sciences USA*, 114(37), 9854–9858.
- Gillmor, S., Poggio, J., & Embretson, S. (2015). Effects of reducing the cognitive load of mathematics test items on student performance. *Numeracy*, 8(1). <https://doi.org/10.5038/1936-4660.8.1.4>
- Glassdoor. (2018). *50 Best Jobs in America*. Retrieved January 2, 2019, from [www.glassdoor.com/List/Best-Jobs-in-America-LST\\_KQ0,20.htm](http://www.glassdoor.com/List/Best-Jobs-in-America-LST_KQ0,20.htm)
- Guest, G., MacQueen, K., & Namey, E. (2012). Themes and codes. In *Applied thematic analysis* (pp. 49–78). Thousand Oaks, CA: Sage Publications.
- Heo, M., & Chow, A. (2005). The impact of computer augmented online learning and assessment tool. *Journal of Educational Technology & Society*, 8(1), 113–125.
- Jenkins, T. (2002). On the difficulty of learning to program. In *3rd Annual LTSN-ICS Conference*, Loughborough University.
- Marx, V. (2013). The big challenges of big data. *Nature*, 498(7453), 255–260.

- Mason, R., & Cooper, G. (2013). Mindstorms robots and the application of cognitive load theory in introductory programming. *Computer Science Education*, 23(4), 296–314.
- Mason, R., Seton, C., & Cooper, G. (2016). Applying cognitive load theory to the redesign of a conventional database systems course. *Computer Science Education*, 26(1), 68–87.
- Michener, W. K., & Jones, M. B. (2012). Ecoinformatics: Supporting ecology as a data-intensive science. *Trends in Ecology & Evolution*, 27(2), 85–93.
- Muenchen, R. A. (2017). The popularity of data science software. *r4stats*. 2017. Retrieved January 2019 from <http://r4stats.com/articles/popularity>
- Nurulain, S., Rum, M., & Ismail, M. A. (2017). Metacognitive support accelerates computer assisted learning for novice programmers. *Journal of Educational Technology & Society*, 20(3), 170–181.
- Onwuegbuzie, A. J., & Wilson, V. A. (2003). Statistics anxiety: Nature, etiology, antecedents, effects, and treatments—a comprehensive review of the literature. *Teaching in Higher Education*, 8(2), 195–209.
- Paas, F. G. W. C., & Van Merriënboer, J. J. G. (1994). Variability of worked examples and transfer of geometrical problem-solving skills: A cognitive-load approach. *Journal of Educational Psychology*, 86(1), 122–133.
- Paas, F. G. W. C., Renkl, A., & Sweller, J. (2004). Cognitive load theory: Instructional implications of the interaction between information structures and cognitive architecture. *Instructional Science*, 32(1/2), 1–8.
- R Core Team. (2016). *R: A language and environment for statistical computing* (Version 3.3.2). Vienna, Austria: R Foundation for Statistical Computing. [www.r-project.org](http://www.r-project.org)
- Renkl, A. (2005). The worked examples principle in multimedia learning. In Mayer, R. E. (Ed.), *The Cambridge handbook of multimedia learning* (pp. 391–412). New York: Cambridge University Press.
- Rohrer, D. (2015). Student instruction should be distributed over long time periods. *Educational Psychology Review*, 27(4), 635–643.
- Ryan, R. M., & Deci, E. L. (2000). Intrinsic and extrinsic motivations: Classic definitions and new directions. *Contemporary Educational Psychology*, 25(1), 54–67.
- Simon, S. F., Robins, A., Baker, B., Box, I., Cutts, Q., de Raadt, M. et al. (2006). Predictors of success in a first programming course. In *Proceedings of the 8th Australasian Conference on Computing Education* (Vol. 52, pp. 189–196).
- Stefan, M. I., Gutlerner, J. L., Born, R. T., & Springer, M. (2015). The quantitative methods boot camp: Teaching quantitative thinking and computing skills to graduate students in the life sciences. *PLoS Computational Biology*, 11(4), e1004208.
- Sweller, J. (1993). Some cognitive processes and their consequences for the organisation and presentation of information. *Australian Journal of Psychology*, 45(1), 1–8.
- Sweller, J., & Chandler, P. (1994). Why some material is difficult to learn. *Cognition and Instruction*, 12(3), 185–233.
- Sweller, J., Chandler, P., Tierney, P., & Cooper, M. (1990). Cognitive load as a factor in the structuring of technical material. *Journal of Experimental Psychology: General*, 119(2), 176–192.
- Sweller, J., van Merriënboer, J. J. G., & Paas, F. (2019). Cognitive architecture and instructional design: 20 years later. *Educational Psychology Review*. <https://doi.org/10.1007/s10648-019-09465-5>.
- Touchon, J. C., & McCoy, M. W. (2016). The mismatch between current statistical practice and doctoral training in ecology. *Ecosphere*, 7(8), e01394.
- Vale, R. D., DeRisi, J., Phillips, R., Dyché Mullins, R., Waterman, C., & Mitchison, T. J. (2012). Graduate education. Interdisciplinary graduate training in teaching labs. *Science*, 338(6114), 1542–1543.
- Venables, W. N., & Ripley, B. D. (2002). *Modern applied statistics with S* (4th ed.). New York: Springer.
- Wilson, B. C., & Shrock, S. (2001). Contributing to success in an introductory computer science course. In *Proceedings of the thirty-second SIGCSE technical symposium on computer science education—SIGCSE '01*. <https://doi.org/10.1145/364447.364581>
- Wilson, G. (Ed.) (2018). *Teaching tech together*. CRC Press. <http://teachtogether.tech>