HOW ENVIRONMENTAL SCIENCE GRADUATE STUDENTS ACQUIRE STATISTICAL COMPUTING SKILLS

ABSTRACT

Modern environmental science research increasingly requires a substantial amount of computational ability to apply statistics to environmental science problems, but graduate students in the sciences typically lack these integral skills. Consequently, many scientific graduate degree programs expect graduate students to acquire these computational skills in an applied statistics course. However, a gap remains between the computational skills required for scientific research and those taught in statistics courses. This qualitative study examines the strategies and experiences of five environmental science graduate students at one institution in acquiring computational skills in the context of an applied statistics course sequence, and the factors that foster or inhibit learning. In-depth interviews revealed three themes in these students' paths towards computational knowledge acquisition: use of peer support, seeking out a singular "consultant," and learning through independent research experience. These themes provide rich descriptions of graduate student experiences and strategies used while developing computational skills to apply statistics in their own research, thus better informing how to improve instruction, both in and out of the formal classroom.

Keywords: *Statistics education research; Statistical computing; Computational thinking*

1. INTRODUCTION

With the increased focus on data-intensive research, statistical computing has become essential in many scientific fields; however, the gap between science education and computation has become more evident, particularly in the environmental and life sciences. The growth in computational power and the volume and variety of available data has multiplied the computational, mathematical, and statistical expectations of scientific researchers' abilities. Because of the importance of computational skills in particular, some universities have started to require undergraduate science students to enroll in an introductory programming course (Rubinstein & Chor, 2014), while others are providing graduate students with computing bootcamps for quantitative methods (Stefan, Gutlerner, Born, & Springer, 2015) or requiring graduate-level statistics coursework for degree completion. Each of these requirements is intended to help students acquire needed computational skills; however, little is known about what resource paths graduate students actually rely upon when faced with computational problems in order to apply statistics in the context of their research. The intention of this study is to understand and describe these paths. We consider the following research question: Where do graduate students in the environmental sciences gain the computational knowledge necessary to implement statistical analyses for applications in their disciplines?

The subjects of this study were graduate students enrolled in the second semester of a graduate-level applied statistics course sequence at a mid-size university in the Western United States. The target audience is non-statistics graduate students; and, at this institution, this two-semester graduate statistics sequence is either required or highly recommended for the completion of a master's degree in departments such as Ecology, Land Resources and Environmental Sciences, Animal and Range Sciences, and Plant Sciences. The course sequence covers the foundations of statistical inference, including a wide variety of statistical methods, starting from two sample inferences and moving through regression and generalized linear models to mixed models. Taught using an `R` (2018) programming environment, students are typically given code to modify, covering base `R` graphics, summaries, and built-in functions, while also being exposed to a few computational concepts such as loops, and conditional and relational statements. This terminal statistics sequence often serves as graduate students' sole statistical computing course, and thus their only formal preparation for the computational problems they may face when implementing statistics as researchers and practitioners. In examining the experiences these environmental science graduate students face when acquiring computational skills, we seek to capture an in-depth understanding of the successes and shortfalls these students encounter in their computational journey.

Though the term "Environmental Science" refers to a specific discipline in the literature, in this paper we will collectively refer to the large assortment of fields serviced by the graduate-level applied statistics course sequence as "environmental science." We begin by describing areas of the research literature that address the computational and statistical training of undergraduate and graduate students in the environmental sciences. We then describe the qualitative study we implemented to explore where graduate environmental science students acquire the skills necessary to complete applications of statistical computing in their field. The results presented reveal the prevailing resources these students employed when faced with computational problems beyond their understanding, and articulate the paths students follow to gain computational skills for carrying out statistical analyses.

2. COMPUTING AND THE ENVIRONMENTAL SCIENCES

Research in computational abilities of environmental science students is in its infancy, with only a handful of institutions performing research that specifically addresses the computational training necessary to prepare students for careers post undergraduate or graduate degree. Literature related to this area has primarily focused on avenues that students could potentially use to increase their computational abilities, with no studies focusing on the resources graduate students actually employ when wrestling with statistical computing problems of their own.

In this section, we discuss briefly three broad areas of the research literature that informed this study. First, we review the literature on the foundational role computation has in the sciences. We then discuss research efforts on curriculum design for introductory computing courses for non-computer science majors. Finally, we describe the available tools which graduate students may use to acquire these critical technological skills.

2.1. COMPUTING AND STATISTICS IN THE SCIENCES

Over the last decade, the life and environmental science fields have seen a rapid increase in the use of computation and analytical tools to model phenomena across many disciplines of inquiry. In some scientific fields, such as biology and chemistry, the recent ability to collect multitudes of data easily and quickly have made computational abilities vital to researchers and practitioners. Meanwhile, fields previously thought to be niche disciplines, such as computational biology, are now "becoming an integral part of the practice of biology across all fields" (Stefan et al., 2015, p. 2). Across a large sector of scientific domains, computationally heavy applications of mathematical and statistical techniques, such as management of large data sets, dynamic data visualization, and modern data analysis, have become essential computational understandings for field applications (Weintrop et al., 2016). With these advances in computational power, analytical methods, and detailed computational and statistical models, scientific fields are undergoing a renaissance. These advances have, however, created a growing need for scientists to receive an appropriate education in computational methods and techniques (Fox & Ouellette, 2013).

When considering curriculum re-evaluation, we note that, for many scientific fields, statistics preparation is considered vital, and has readily been incorporated into undergraduate and graduate programs across the country. Many chemistry, biochemistry, and bioinformatics programs have additionally begun to incorporate computational training into their programs, however, a similar revolution, affirming the importance of computational proficiency, has yet to be experienced in the environmental sciences. Hence, undergraduate and graduate-level terminal statistics courses potentially act as the sole or final statistical computing training students receive prior to performing independent research.

2.2. COMPUTATIONAL COURSES FOR UNDERGRADUATE SCIENCE MAJORS

Multi-disciplinary efforts have been made at Purdue, Carnegie-Mellon, Harvey Mudd, Princeton, and Winona State universities to create introductory computing courses with a focus on non-computer science undergraduate majors, in particular science students, in fields ranging from physics to chemistry to biology (Cortina, 2007; Sedgewich & Wayne, 2008; Sedgewich & Wayne, 2015; Wilson et al., 2008; Wing 2006). These courses were produced in collaboration with science faculty, and are intended to begin each student's journey into computing. Students are presented examples in a familiar language that allows them to focus on the foundational principles of each computational problem (Hambrusch et al., 2009).

The research on undergraduate level computing courses directly relates to the computational abilities of graduate-level science researchers, as these students have most likely graduated from an undergraduate science program with little to no computational training. The concepts emphasized in the courses developed at Purdue, and elsewhere, can be used to inform the environmental sciences and statistics departments as to which computational skills other scientific disciplines, such as physics, biology, and chemistry, believe to be the most important for students to grasp.

2.3. COMPUTATIONAL TRAINING FOR GRADUATE SCIENCE MAJORS

A variety of research has been published on teaching computational skills to graduate students in biological fields, with limited attention paid directly to environmental science graduate students. Anecdotal accounts of teaching `R` or Python to computational biology students (Eglen, 2009; Ekmekci, McAnany, Mura, 2016) provide insight on instructional methods used, with the intention of informing faculty on the successes and shortfalls of these instructional methods. Alternatively, Hampton et al. (2017) highlight the variety of “extramural options for acquiring critical technological skills” (p. 547), emphasizing the importance of instructional methods for transferring computational skills to individual fields of research. These extramural options range from single online programming lessons to in-person workshops to full university courses, each with their own target audience. Yet, none of the existing prevalent training resources are targeted towards graduate students.

Researchers in the Department of Biological and Biomedical Sciences at Harvard have developed one such intensive workshop that introduces graduate bioinformatics students to the "fundamentals of programming, statistics, and image and data analysis through the use of MATLAB" (Stefan et al., 2015, p. 2). This course is framed not only with the goals of developing programming skills and statistical understandings, but also emphasizing how to algorithmically reason through a computational problem. The structure of the “two-week intensive ‘bootcamp’” consists of five full, mandatory days. The workshop dedicates the first two days to an introduction to programming using MATLAB, where students learn a variety of topics, including creating variables, performing basic variable operations, indexing, logicals, functions, conditionals, and loops. Day 3 is dedicated to developing statistical understandings, including probability distributions, hypothesis testing, p-values, bootstrapping methods, and multiple testing. Day 4 covers topics in image analysis, and Day 5 assists students in working with their own data. These workshops are given twice a year, once prior to the start of the school year as new graduate students are attending orientation, and a second time for upper-level graduate students and post-doctoral fellows (Gutlerner & Van Vactor, 2013).

In introducing beginning graduate students to these concepts, researchers hoped to lower the computational barrier for students taking courses, empower students to learn computational tools on their own, and allow for other courses to "build upon this foundation and integrate quantitative methods throughout the curriculum" (Stefan et al., 2015. p. 2). Survey results from the last five offerings (Spring 2012 to Spring 2014) indicated that “students report significant gains in their self-assessed programming ability,” with students reporting that some of the computational concepts “around statistics [are the] most challenging” (Stefan et al., 2015. p. 8). These surveys also indicated that, following completion of the course, students believed they had acquired practical quantitative and computational skills that would prepare them for research in their field, recognized the importance of computational and quantitative methods in their field, felt confident in the methods they had learned, and would recommend that other graduate students learn these types of computational methods.

This study aims to close the gap in the research between the computational training of graduate environmental science students and the resources environmental science graduate students invoke when reasoning through computational problems in order to apply statistics in their field of research.

3. METHODOLOGY

For this study, a pragmatic phenomenological approach was appropriate, as the intention was to understand and describe common experiences in statistical computational thinking and abilities for environmental science graduate students when applying their computational skills and understandings in their own research. The focus of this study lent itself naturally to a pragmatic framework, since a pragmatic framework allows for an emphasis on the process of finding a working solution, allowing for varied solutions rather than a single solution (Creswell, 2013). A phenomenology formed the appropriate context for this study, because every graduate student selected had experienced the same phenomenon of enrolling in the graduate-level applied statistics course sequence.

We examined factors that influenced how environmental science graduate students gained computational knowledge and the ability to reason through statistical applications in their disciplines. In this study, statistical computing is considered to consist of the computing necessary for the entire process of statistical analysis, from data cleaning to data visualization to data analysis. We chose to align our definition of computational thinking with the taxonomy developed by Weintrop et al. (2016). This definition includes fluency of computer programming, knowledge of data practices, the ability to reason through problems with a given set of tools, and knowledge of resources that could provide assistance in solving a particular problem. This allows for the possiblity that these students could call upon any or all of these skills as they reason through computational problems in order to apply statistics to their research.

3.1. PARTICIPANTS

Students were recruited from the second semester of a graduate-level applied statistics course sequence in spring of 2017. These students were interviewed following their spring break, nearly halfway through the course. Only graduate students taking the course for their respective master's or doctoral programs in environmental science fields were considered.

All of the environmental science graduate students were requested to complete a survey detailing their previous statistics and computer science courses, the computer languages they were familiar with, and their independent research experience. A total of eight graduate environmental science students were enrolled in this course in the spring of 2017, all of which completed the survey. All eight of these students were then asked to participate in an in-depth interview, of which five agreed.

The five students who agreed to be interviewed all identified as women, and all had taken the first course in the sequence in the last two years. Additional details of these five participants are summarized in Table 1.

Of the five interview participants, four had taken or were taking the four statistics courses required for completion of a Graduate Certificate in Applied Statistics; these four courses include the graduate-level applied statistics course sequence, sampling or experimental design, and one additional upper-level statistics course. However, one participant, Catherine's only prior statistics course had been the first semester of the graduate-level applied statistics course sequence. The two participants who had taken general computer science courses had done so in their undergraduate coursework. Every participant voiced familiarity with SQL, either from independent research experiences or from coursework. Stephanie instead had experience with both Python and Java after completing a year's work as a research assistant, prior to enrolling in graduate school. Four of the participants had begun or were nearly finished with their master's thesis, while Robin had just begun to work on the projects associated with her dissertation.

Table 1. Academic demographics of participants: degree seeking, program of study, academic year they took the first semester graduate-level applied statistics (GLAS) course (Fall, Spring), number of statistics and computer science (CS) courses they had taken (undergraduate and graduate), programming languages they were familiar with, and amount of independent research they had completed.

Name (pseudonym)	***Degree***	***Program***	***GLAS I***	***Stat Courses***	***CS Courses***	***CS Languages***	***Independent Research***

Beth	MS	Animal Range Science	F 2015	4	0	SQL	Thesis
Catherine	MS	Environmental Science	F 2015	1	0	SQL	Thesis
Kelly	MS	Ecology	S 2016	3	1	SQL	Thesis
Robin	PhD	Environmental Science	F 2015	2	2	Python, SQL	A few projects
Stephanie	MS	Environmental Science	F 2015	4	0	Python, Java, SQL	Thesis

3.2. DATA COLLECTION

Following the preliminary survey, students who agreed to be interviewed were audio recorded describing their coursework, where and how they acquired their computational knowledge, and discussing their experiences in acquiring these computational understandings. This interview protocol was modified from surveys administered by researchers investigating computational thinking at Harvard using Scratch, an interactive programming language for creating interactive art, stories, and simulations. This rubric provided rich content for assessing students' experiences in performing computational applications ("Interviewing Students," n.d.). The full interview protocol is included as an Appendix.

Following the interview, students were asked to work through a set of ecological applications of statistical computing. These tasks assessed students' abilities to reason through applications of statistical computing, and outlined any gaps noticeable in students' ability to transfer their statistical knowledge to applications of statistical computing. Computational problems were developed in collaboration with statistics faculty at the university. The analysis in this paper is based on the five interviews on students' experiences acquiring the computational knowledge necessary to apply statistics to their research; students' abilities to reason through ecological applications of statistical computing will be used in a future analysis.

3.3. DATA ANALYSIS

Interviews for each participant were transcribed verbatim, with participants' names removed and pseudonyms given. Descriptive coding was then implemented to analyze and describe how the participants had acquired their knowledge of the computational skills they used to complete the statistical computing tasks.

The transcripts were read numerous times in order to segment the data and construct themes specific to each individual's acquisition of statistical computing skills. With these individual themes, we were able to compare commonalities that emerged across participants. Initially, three themes in statististical computing knowledge acquisition emerged. When new variations of knowledge acquistion emerged, they were scrutinized to see if they fit within the developed codebook or if modifcations were necessary. The original three themes remained as the final three themes outlined in Section 4.

To establish validity, participants were provided with an itemized detail of how they completed the problem and the transcription of their interview. This inclusion of member checking allows participants to check for accuracy of their statements. The authenticity of the study, its ability to capture the perceived experiences of students' abilities to think through computational problems, is enhanced with the lack of researcher engagement with students prior to their participation in the study. This helped to ensure that no student felt more comfortable in the interview environment, articulating their experiences, than any other student.

4. RESULTS

When investigating where these students gained their computational knowledge, we expected themes of content and support structure to emerge. However, the themes that emerged from every participant's interview primarily related to the support structures they employed, rather than the content that helped them succeed when performing applications of statistical computing. In this section, we present these themes that developed throughout the participants' interviews: (1) independent research, (2) singular consultant, and (3) peer support. A sub-theme of "coursework" appeared within peer support and independent research, where participants voiced the importance of their coursework on their knowledge of statistical computing. However, this sub-theme was consistently voiced to depend on either peer assistance or independent research in its impact on participants' understanding of statistical computing. The themes and sub-themes are summarized in Table 2.

Table 2. Participants' themes in acquisition of statistical computing knowledge

Theme	***Sub Theme***	***Description***
Independent Research	Coursework	Research experiences that allowed students to take their course knowledge and transfer it to statistical computing applications
Singular Consultant		All-knowing past or current graduate student whom students sought out for computational assistance
Peer Support	Coursework	Assistance from peers with statistical computing tasks

In the sections that follow, we provide a detailed description of each theme, supplemented with quotations from participants to ensure authenticity of descriptions of their experiences.

4.1. INDEPENDENT RESEARCH EXPERIENCE

The first theme in acquiring statistical computing knowledge was participation in independent research. Involvement in independent research helped students transfer their course knowledge to statistical computing applications, seeing the messiness of non-classroom applications. These experiences came predominantly in the form of working as a research assistant prior to entering graduate school, collaborating on a project in the first year of graduate school, or performing research for a master's thesis, or ultimately, a doctoral dissertation.

Catherine, a master's student in Environmental Science, who still faced everyday computational struggles, attributed the majority of her application-specific computational knowledge to her experiences in independent research. She emphasized the importance of understanding how to work in a statistical computing environment, such as `R`, which she learned from performing research, before she was able to begin to transfer the statistical knowledge she had learned in the classroom:

> What I struggled with is [the first graduate-level statistics course] covers theory really well, but since I was new I spent most of my time trying to figure out how to apply that theory in [R]. And even now I struggle transferring from R into actual statistical theory, when I'm writing my thesis. The way I had to approach it was I had to learn the R first, then I was able to look back on what I had actually done, in order to learn the statistics.

Kelly, an Ecology master's student, described her experiences with data management for her master's thesis as having produced the most substantial contributions to her computational abilities. Often she attributed her intuition for solving statistical computing problems to experiences she had "merging data sets" and learning to use conditional statements for her project. She emphasized the importance of her statistical knowledge gained in both graduate-level statistics courses in understanding "what statistical method to use," but she attributed becoming more fluent in statistical computing to her research experiences: "The data management stuff comes from independent research, trial and error, getting myself through." Similar sentiments were voiced by Beth, an Animal Range master's student, with the majority of her computational knowledge stemming from her independent research. With the recommendation of her adviser, she taught herself how to create an Access database to store her data. In storing her project in this manner, she was able to learn important concepts about data structures, subsetting data "using qualifiers and criteria," and sorting data, all using SQL statements.

4.2. SINGULAR CONSULTANT

When describing whom they seek out for computational help, every participant described first seeking out an "all-knowing" past or current graduate student. These figures served as "singular consultants," with whom these students had the "best," most productive, experiences in finding solutions to computational problems that had arisen in their implementation of statistics to their research. For Beth, this singular consultant came in the form of a past graduate student from Animal Range Sciences who was hired to help faculty complete projects:

> We have a guy who used to be a student in our department and then he was hired on again to help finish some projects, after he got his master's in Statistics. He is very helpful with [pointing out what's wrong with your code]. He's very good with code and if I have a quick question he can always answer it.

For Kelly, another graduate student on her same project served as this consultant. Kelly described turning to this particular graduate student for help with computational problems she had encountered in her thesis; she added that other graduate students in their department also used this person as a consultant for their computational problems:

> The other grad student on this project is so well versed in R that he's unofficially become the person that people go to with questions.

Throughout her computational struggles, Catherine found assistance from previous graduate students from the department, but she found the most assistance from a previous graduate student "who had left the department and was off professionally somewhere else, but he still took the time to help walk me through [my code]."

One participant, Stephanie, a Environmental Science master's student, served as this singular computational consultant for many members of the Environmental Science department. With her experiences teaching herself `R`, she was able to "explain code in a way that makes sense," says Robin, a fellow Environmental Science doctoral student who has often sought out help from Stephanie. With an adviser from a computational background and a project which performs sophisticated statistical modeling, Stephanie "has to learn code." Additionally, her laboratory often worked in collaboration with computer science faculty, where she and her lab-mates were taught computer science coding practices and jargon. "Stephanie has gotten good at teaching it, because everyone on our floor is like 'I can't do this, Stephanie help me'," said Robin. Stephanie stated that graduate students have sought her assistance "daily" or "at minimum two to three times a week." In contrast, when Stephanie experiences difficulty in performing computational tasks, she has found solace in her lab-mates and ultimately, when necessary, with her adviser:

> My entire lab works in the same room and my adviser's door is always open. So if someone is having a major issue, whoever is in the room can hear that. If [my adviser] hears me ask [lab-mate] how to do something and he knows how, he just shouts how to do it. So it's a very group oriented dynamic. I've never had to go beyond the people in my lab.

4.3. PEER SUPPORT

The third theme in acquiring computational knowledge which all participants spoke of was the support they had received from fellow graduate students when performing computational tasks related to applications of statistics. The students described how, when they are unsure of how to complete a computational task for their research and their singular consultant is not available to them, they turn to fellow graduate students for help. Participants described instances when the computational tasks required of them were beyond their current knowledge or occasions when they had been unsuccessful at attempting to complete a problem and sought out help from a fellow graduate student. For example, Kelly, an Animal Range Science master's student, shared that when she reached a point in coding when she doesn't know how to do something she turned to one of her lab-mates:

> I've been to a point where I didn't know how to do something with my knowledge or what I can find online, and then I'll go to one of my lab-mates.

Catherine, a master's student in Environmental Science, spoke of the expectations of her advisers that the computational problems she was being asked to perform were "easy, since she had all the information." However, she has had numerous experiences where she did not have the knowledge necessary to perform the task or she was missing "little caveats" that kept her from fully being able to perform the tasks. When faced with these problems, she "reached out to previous students that had taken the course."

Robin, a doctoral student in Environmental Science, reiterated Catherine's experiences, describing how she reached out to other graduate students in other labs for help with computational problems. Stephanie, however, as a singular consultant, voiced that when she was faced with computational problems beyond her knowledge, she had never been forced to "go beyond talking to her lab-mates" for assistance.

However, peer support does not always provide an optimal solution, a potential reason for participants seeking help from peers only when their singular consultant was unavailable. For example, Kelly described negative experiences when seeking computational assistance from graduate students not of close proximity to her:

> When I'm struggling with something and I go to other grad students, they'll say "I did this the other day. I'll send you my code." I've found most of the time I don't understand what they've done enough to plug in what I want and make it work. There have been a few times when making tables and plots and someone sends me their code and I can just plug in my data and it works just fine. I've had less success with that.

5. DISCUSSION

The present study, while exploratory in nature, outlines the paths that some environmental science graduate students employ to gain the computational skills necessary to reason through applications of statistics in their fields. A path diagram, depicting the resources students move through when faced with applications of statistical computing, is shown in Figure 1.

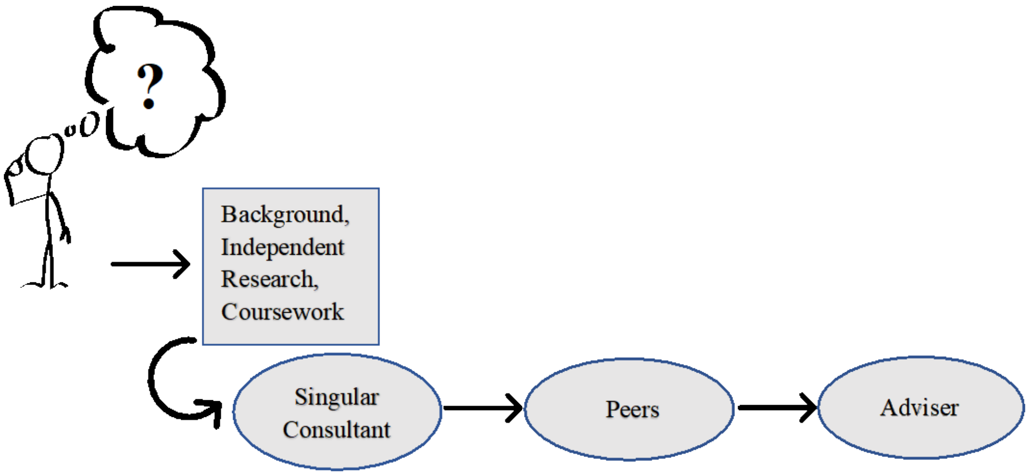

Figure 1. Graduate students' resources used in reasoning through statistical computing applications, corresponding to themes in sections 4.1, 4.2, and 4.3.

The expectation of coursework to be a primary source of statistical computing knowledge was not found for these participants. Indeed, when these graduate students encountered a statistical computing problem, they first pulled upon the knowledge they had acquired through their undergraduate background, graduate coursework and independent research, but this knowledge was often insufficient. Rather, the computational understandings that these students attributed to their statistics coursework were primarily low-level concepts, such as using built-in `R` functions, adding comments to their code, and limited trouble-shooting of error messages. Additionally, these concepts were said to only be fully understood through participants' peer interactions, or as they were being implemented independently, within their own research.

Participants voiced the importance of their experiences performing independent research as having a substantial influence on their abilities to reason through and perform the computational tasks required for various statistical analyses. Through independent research, the participants were able to play with real-world data and applications outside of what they had encountered in the classroom. The programming understandings developed during a student's independent research, in conjunction with peer collaboration, were described largely as high-level concepts, such as conditional statements, loop implementation, and some user-defined functions. Students described their independent research as having opened the door to experiencing the unease that comes when one is asked to perform statistical computing tasks beyond one's knowledge, a feeling they had not experienced in their courses. In these circumstances, students stated that they would ask for help from the people with whom they felt the most comfortable.

In a direct connection to the participants' discomfort in asking for help from an adviser, the theme of a singular consultant emerged. These singular consultants served as an “all-knowing” individual, from whom the participants had either had the “best” experiences with, where the individual spent the necessary time to explain the concepts, or the consultant had always been capable of providing the participant with a solution to their problem. These individuals served as the first line of defense when statistical computing problems arose, where participants were both able to seek computational help and acquire new computational skills and understandings through their interactions. If due to time or physical constraints, this consultant was unavailable to the graduate student, these students then turned to their peers.

Peer support was initially discussed by the participants in their interviews as a mechanism they used when their “code doesn't run” or when they were asked (or needed) to do something beyond their current computational understandings. However, this theme continued to emerge as the participants worked through computational problems, often attributing their knowledge of a computational procedure to a friend or fellow graduate student helping them “do it with their data.” These peers offered an avenue for students to seek help, often voiced to be more comfortable than asking an adviser, where participants described both the fear of asking and “feeling dumb,” or being “brushed off” because their adviser thought they should “be able to figure out how to do it.” However, as opposed to the help participants received from their singular consultant, these students also voiced negative experiences they had encountered when seeking help from their peers, such as a peer sending them “helpful” code that they did not understand.

Lastly, the adviser played an important role in students acquiring the computational knowledge necessary to perform applications. Despite students’ reluctance to seek out computational assistance from their adviser, advisers did often emphasize the importance of statistical computing skills, as well as introduce (or recommend) students to store their data using an `Access` database. The participants’ ability to understand both data structures and sorting or filtering data was largely attributed to their experiences working with these types of databases. Although these interviews found that advisers were often considered as the last line of defense, they were, however, viewed as an accessible way for students to better understand the statistical computing necessary for their independent research projects, which overall contributed to better computational understanding and skills for these students.

6. IMPLICATIONS

The implications for statistics education focus on the importance of graduate students’ acquisition of the computational knowledge needed to apply statistical methods in their own research. Environmental science fields have long understood the importance of a statistics education for their students. However, many of these programs are not actively incorporating computational courses into their degree, instead assuming that students are acquiring these skills in their recommended statistics courses. A restructuring dilemma is faced by both parties involved, with intractable differences between coursework and knowledge expectations.

The impact of an undergraduate education on students' experiences as graduate researchers should be considered by statistics, ecology, and environmental science faculty in higher education when recognizing the importance of developing data-intensive statistical computing skills in undergraduate courses. In this study, students with fewer computational skills and understandings had substantially different independent research experiences than their counterparts with more. The frustrations of simple tasks, such as subsetting data or removing NA's, were felt by the participants who had completed a bachelor's without any computational elements to their coursework, while those who were exposed to even small amounts of computing in their undergraduate coursework, such as a general computer science course, a GIS course, or experience with `Access` databases, were able to begin computational tasks in their research walking and not crawling.

Additionally, statistics educators should consider the power an applied statistics course sequence has to provide graduate students with a year-long introduction to statistical computing. As seen by Stephanie, who entered graduate school after completing a year's work as a research assistant working in `R`, these learning experiences can help to alleviate the power differential students feel when asking their advisers or peers for assistance. However, the content covered by graduate applied statistics sequences is expected to paint a vast picture of the field of statistics, with topics ranging from a difference in means to mixed-models. Consequently, many educators feel they do not have the time to incorporate statistical computing into the classroom, and some feel that they have limited computational expertise to teach these concepts (Hampton et al., 2017; Nolan & Temple Lang, 2010). The inflexibility of graduate programs further complicates this issue, as many graduate students are unable to enroll directly in a statistical computing course due to an already full and demanding course load. Thus, questions should be raised about how to best bridge this gap between coursework and research expectations for statistical computing skills.

The importance of playing with statistical applications on real-world data, as voiced by these participants, should be also considered by statistics educators at all levels. This transition to incorporating authentic, research-like tasks, which engage students in statistical computing, can be supported by online resources, data-discovery tools, example datasets and code, and instructional tools, along with collaborative course designs and the sharing of instruction materials.

6.1. LIMITATIONS

While the methodology used to discover some graduate environmental science students' experiences with statistical computing in the present study provided useful themes, it is not without its limitations. Eliciting descriptions of computational knowledge acquisition yielded varied experiences with each of the main themes, but richer data could be gathered in a future longitudinal study. A study following graduate students throughout their program of study could identify where students are acquiring statistical computing knowledge, as well as instructional methods that best assist students in obtaining these understandings. To better inform science and statistics faculty, a thorough investigation of both the coursework and structure of courses completed by these participants could be performed. This would allow for a discussion of how to best integrate these computational concepts into current coursework requirements, so that students leave the classroom with understandings they can implement immediately in their own research.

Finally, it should be noted that the present study focused on describing five environmental science graduate students' experiences in acquiring statistical computing knowledge, but not in what computational knowledge they possessed. Therefore, we have learned primarily about the resources these five students relied upon when they experienced computational expectations beyond their ability. The present study intended to understand and describe these statistical computing experiences rather than make statistical generalizations to a larger population. Hence, the graduate students interviewed in this study may not be representative of other graduate students from their respective fields of study, and these participants' experiences may not be generalizable to other graduate students within their fields.

7. CONCLUSION

Statistical computing has become a foundational aspect of research in the environmental sciences. This small-scale exploratory study brings forward the experiences of graduate environmental science students in acquiring the computational understandings necessary to successfully perform field-related statistical applications. Participants found the greatest success in acquiring the computational skills required for their research through independent research, a singular consultant, and peers. Whereas others have noted the importance of integrating computing into the undergraduate science curriculum (Cortina, 2007; Sedgewich & Wayne, 2008; Sedgewich & Wayne, 2015; Wilson et al., 2008; Wing, 2006) or how to provide computational training for biological science graduate students (Stefan, Gutlerner, Born, & Springer, 2015; Eglen, 2009; Ekmekci, McAnany, Mura, 2016), we instead explored the computational knowledge acquisition experiences of graduate environmental science students. The computational burdens experienced by these participants when implementing statistics in the context of their research and the computational understanding with which they left the statistics classroom, suggests the need for integration of formal computational training into these programs. The present study helps to emphasize the importance of computing skills necessary for data-intensive environmental science research.

ACKNOWLEDGEMENTS

REFERENCES

Bloom, B. S. (1956). *Taxonomy of educational objectives, handbook I: The cognitive domain.* New York: David McKay Co Inc.

Computational thinking with scratch, developing fluency with computational concepts, practices, and perspectives. (2018, May 26). Retrieved from Interviewing students about scratch programming experiences : http://scratched.gse.harvard.edu/ct/files/Student_Interview_Protocol.pdf

Cortina, T. (2007). An introduction to computer science for non-majors using principles of computation. *Proceedings of the 38th SIGCSE technical symposium on computer science education*, (pp. 218-222). Covington, Kentucky.

Creswell, J. (2013). *Qualitative inquiry & research design* (Vol. 3). Thousand Oaks, California: Sage Publications.

Fox, J. A., & Ouellette, B. F. (2013). Education in computational biology today and tomorrow. *PLOS Computational Biology, 9*(12), 1-2.

Gutlerner, J. L., & Van Vactor, D. (2013). Catalyzing curriculum evolution in graduate science education. *Cell, 153*(4), 731-736.

Hambrusch, S., Hoffmann, C., Korb, J., Kaugan, M., & Hosking, A. (2009). A multidisciplinary approach towards computational thinking for science majors. *Proceedings of the 40th ACM technical symposium on computer science education*, (pp. 183-187). Chattanooga, Tennessee.

Hampton, S. E., Jones, M. B., Wasser, L. A., Schildhauer, M. P., Supp, S. R., Brun, J., . . . Aukema, J. E. (2017). Skills and knowledge for data-intensive envrionmental research. *BioScience, 67*(6), 546-557.

Miles, M., Huberman, . A., & Saladaña, J. (2014). *Qualitative data analysis: A methods sourcebook.* Thousand Oaks, California: Sage Publications.

Nolan, D., & Temple Lang, D. (2010). Computing in the Statistics Curricula. *American Statistician, 64*(2), 97-107.

R Core Team, (2018). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Rubinstein, A., & Chor, B. (2014). Computational thinking in life science education. *PLOS Computational Biology, 10*(11), 1-5.

Sedgewich, R., & Wayne, K. (2008). *Introduction to programming in java.* Boston: Addison Wesley.

Sedgewich, R., & Wayne, K. (2015). *Introduction to programming in python.* Boston: Addison Wesley.

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), 1-12.

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology, 25*(1), 127-147.

Wilson, G., Alvarado, C., Campbell, J., Landau, R., & Sedgewich, R. (2008). CS-1 for scientists. *Proceedings of the 39th SIGCSE technical symposium on computer science education*, (pp. 36-37). Portland, Oregon.

Wing, J. (2006). Computational thinking. *49*(3), 33-35.

APPENDIX

INTERVIEW PROTOCOL

Describe a time when your code didn't run as you wanted.

Describe how you investigated the cause of the problem and fixed the problem.

Describe other ways you could have fixed the problem.

Describe a time when you could not find a way to fix your code.

Where did you turn to for help and why?

Describe how you found advice or support by using someone else's code on a project or homework. Why did you seek out advice or support?

Describe a time you used the code from another homework or project as part of your homework or project. How often do you use previous code on a current project or homework?

Describe a time you modified existing code (either someone else's or your own) to improve or enhance it.

Where have you learned the statistical computing skills necessary for your course work and research?