2021 ASEE ANNUAL CONFERENCE
Virtual Meeting | July 26–29, 2021 | Pacific Daylight Time
Paper ID #34115

Fundamental Theorem Learning With Optimum Pedagogy for Technology Integration in Quality Control Course

Ms. McKenzie Landrum, University of Florida

Ms. McKenzie Landrum is an Instructional Assistant Professor in the Department of Industrial and Systems Engineering at the University of Florida. Both her Bachelors in ISE and her Masters in Management come from UF and she is currently pursuing her PhD in ISE from UF. As an Instructional Assistant Professor, Ms. Landrum has taught Spreadsheet-Based Decision Support Systems, Industrial Quality Control, Lean Production Systems, Introduction to Data Analytics, and Computer Programming for Engineers. Her research interests lie in the development of active learning methods.

Mr. Austin Hayes, University of Florida

Mr. Austin Hayes is an undergraduate student at the University of Florida currently pursuing his B.S. in Industrial and Systems Engineering.

Josefina Giamichelle
Dr. Sima Sabahi, University of Florida

Sima Sabahi is an Instructional Assistant Professor in the Department of Industrial and Systems Engineering at the University of Florida.

Ms. Sanaz Motamedi, University of Florida

Sanaz Motamedi is faculty member at Industrial and System Engineering, the University of Florida after her postdoctoral fellowship at the California Partners for Advanced Transportation Technology program, University of California, Berkeley. Her research interests are human factors, user experience, engineering education, and transportation safety.

Fundamental Theorem Learning with Optimum Pedagogy for Technology Integration in Quality Control Course (Work in Progress)

Abstract

As technology advances and new software is developed, the education system is being challenged to adapt pedagogical approaches for the smooth integration of such tools into the curriculum. These tools can be beneficial for teaching because they allow students to visualize difficult concepts and can be used to execute functions that would otherwise be time prohibitive. However, there is a concerning trend of students depending too heavily on this technology. Technology provides an avenue through which students can feign comprehension and continue advancing in the curriculum. The purpose of this study is to look at different pedagogical approaches and their effects on student's self-efficacy and topic comprehension. To address this, we worked with a required course (ESI4221C: Industrial Quality Control) in the Industrial and Systems Engineering (ISE) curriculum at the University of Florida (UF). This course focuses on quality control and builds on statistical fundamentals while also introducing new theoretical concepts such as tests statistics, confidence intervals, p values, and ANOVA. Not only does this course teach the fundamental concepts, but it includes teaching common software that is used in industry or higher education, specifically RStudio. Different pedagogical approaches were used to teach fundamental concepts along with software to students in this study and each approach was randomly assigned to a single module. The first pedagogical approach is the Instructor Guided Method in which the instructor taught RStudio to students after each topic. The second pedagogical approach is the Think-Pair-Share Method in which students were assigned mandatory readings and instructor dedicated class time for peer-to-peer discussion. Self-efficacy surveys and conceptual/computational assessments were given for each method to determine how students felt while learning the material and their comprehension. Our preliminary data indicates that there is no statistically significant difference between the teaching methods implemented so far. The data did not reveal a difference between the methods used so far, but additional studies need to be done to assess if this result is due to low sample size or differences in difficulty between the modules that each method was assigned to. This research will be continued and expanded on in the fall 2021 semester to collect more data from additional courses and the methods applied to different modules.

Introduction

Due to the rapid growth of technology and the use of software, the importance of teaching theory alongside software has increased drastically. Software programs have become very powerful, with the ability to solve problems efficiently and eliminate the chance for computational error. Despite these benefits, concerns have arisen about students developing a dependence on these software programs. Using certain teaching methods to expose students to software may negatively impact their ability to learn the theoretical concepts.

The common pedagogical approach is to introduce students to the software as they learn the theory; however, this tends to worsen the dependency issue. For example, the theory taught in upper-level statistics courses is difficult to grasp for many students. To compensate for this difficulty, students view the software as a way out. Even if they may not fully understand the theory, students are able to get the correct answer to problems by simply plugging in numbers. This leaves some students with a weak theoretical foundation for future statistics. As educators, we want to minimize this from occurring in our classrooms. This situation exposes a need to determine the optimal way to include software in courses, especially those heavy in statistics and theory.

To address this need, we identified multiple pedagogical methods to test in a quality control classroom at the University of Florida. We wanted to investigate the best way to introduce students to software while a new theoretical concept is being taught. For a student to maximize the learning of new software, they first need to fully understand the theoretical basics. When students have mastered the theory, then they can understand the inputs and outputs of the software and interpret these results.

Background

The inclusion of software in education is an important research topic because many courses require the use of integrated learning tools, such as calculators or programs, to compliment the theory taught [1]. Our literature review is mostly based in research studies that integrated graphing calculators into classroom environments and then noted the impact on students since calculators have been used for a long time. However, research on advanced software tools is less robust.

Most of the literature focused on the use of graphing calculators rather than software and the potential dependence on calculators that could occur. The study populations in the literature were mainly elementary, middle, or high school students [1, 2, 3, 4]. These age groups would likely respond to software integration differently than upperclassmen in college. While seeing how graphing calculators are used in lower-level classrooms is interesting, we are more concerned about software use in upper-level classrooms.

Learning software for the first time is similar to learning calculators for the first time, however, software is often more difficult for students to grasp. One common problem that students face while learning software is visualizing the given problem and connecting it to the software being used. Rather than learning how to input numbers into a preset function, software allows students to develop their own algorithms, requiring creativity and problem-solving skills. Therefore, the use of software during the learning process is beneficial for students as long as it is restricted. Applying restrictions while students are introduced to software will lower the chance of them becoming dependent on it. For example, regarding calculators, putting restrictions on the use of calculators in prior courses was positively related with students' performance in their following courses [2]. With this in mind, there is more reason to believe that a balanced integration of software will be beneficial for students to succeed in the future.

For a student to fully master the use of software, they first need to understand the fundamental concepts behind the software functionality [5]. In investigating how to introduce students to software, one example details that tutorial and computer demonstrations tend to be helpful for many students [6]. Specifically, tutorials were found to be more beneficial if they were integrated into lectures rather than assigned for students to cover on their own time. Although asynchronous tutorials give students flexibility with their studying, the synchronous environment is more conducive to learning. Furthermore, instructors must ensure that the tutorials they share with students are well-structured and properly delivered [6]. Poorly developed tutorials may not share the same benefits as high-quality ones. This study did have limitations, however. The research was originally conducted in 1997 and software tools and applications have grown significantly since then. In addition, this study did not do any statistical analysis to evaluate the results of their work [6].

To measure how students felt about the course, we measured self-efficacy - an individual's belief in their capacity to execute behaviors necessary to produce specific performance attainments [7]. If during the introduction of a new software the student becomes discouraged, they will likely develop a negative attitude towards the use of such software as well as a negative attitude towards learning software in the future [8]. Discouraged students do not feel confident about their knowledge and over time this leads to a decrease in their self-efficacy. On the other hand, students are more confident with material they find engaging [8]. This confidence translates into a greater self-efficacy and more success in the classroom. For these reasons, engineering educators need to be concerned with their students' beliefs about their skills to be able to succeed [8].

In this study, we are addressing the following question: What teaching method is most conducive for students to dominate the theory and be proficient with software?

Methodology

Participants. In this study we targeted students who enrolled in ESI4221C: Industrial Quality Control during the spring 2021 semester. The mode of delivery was synchronous and weekly online Zoom meetings were held for this class. In this class, we cover the use of modern statistical methods for quality control and improvement. We provide comprehensive coverage of the subject from basic principles to state-of-the-art concepts and applications using RStudio.

There were 24 students in this class that agreed to participate in this study. Out of these, 62.5% were females and the other 37.5% were male. Figure 1 shows the ethnic makeup of our sample.

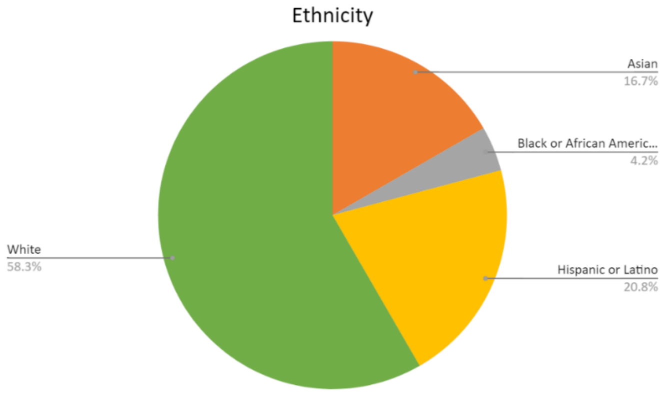

Figure 1: Ethnic makeup of the sample

The majority of the students were seniors (87.5%) and the rest were juniors (12.5%). 87.5% of these students attended high school in the United States and the other 12.5% attended high school in a different country. Furthermore, while in high school, 37.5% of these students took a statistics course but 62.5% did not.

There are two courses that are prerequisites for this course (STA4321: Introduction to Probability and STA4322: Introduction to Statistics Theory). In addition to these prerequisites, some students have taken additional statistics courses, such as STA3032: Engineering Statistics. In our study population, only one student took one additional statistics course, and one other student took two additional statistic courses. Figure 2 shows the self-reported average grade of students over their prerequisite statistics courses.

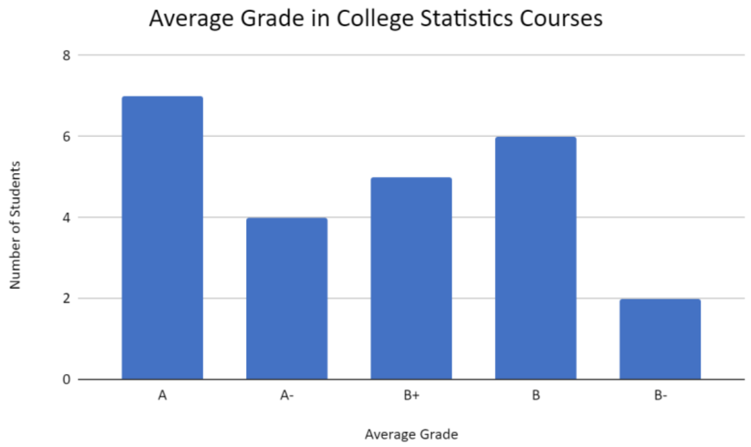

Figure 2: Students' achievement in prerequisite college statistics courses

Before the class began, 16.67% of the students had minor experience with RStudio and 83.33% of the students had no experience. Furthermore, 12.5% of the students had already worked in an internship position using the topics taught in the course and 87.5% had not. However, 41.67% of the students plan on using the topics taught in the course during their career or a future internship.

Course Content. As mentioned previously, we applied different pedagogical methods to a quality control course. This course introduces students to statistical techniques used in the design, control, and improvement of quality. In particular, our research focused on two modules within the course: Modeling Process Quality and Interpreting Process Quality. These modules are a review of students' prior statistics courses with focus on how these concepts are applicable to quality control.

The topics discussed in Modeling Process Quality include constructing and interpreting visual data displays (such as stem-and-leaf plot, histogram, and box plot), computing and interpreting descriptive statistics (such as sample mean, sample variance, sample standard deviation, and sample range). In this module, we also explain the concepts of a random variable and a probability distribution, understanding and interpreting the mean, variance, and standard deviation of a probability distribution, determining probabilities from probability distributions.

We cover the differences between a discrete probability distribution and a continuous probability distribution, select an appropriate probability distribution for use in specific applications, using probability plots, and lastly using approximations for some hypergeometric and binomial distributions. This module takes four 50-minutes classes on average.

Topics discussed in the next module, Interpreting Process Quality, include random sampling, sampling distributions, parameter estimation, confidence intervals on a single mean and on the difference in two means, confidence intervals on a single variance or the ratio of two variances, and confidence intervals on a single proportion and on the difference in two proportions. In this module we also cover test hypotheses, P-value approach, analysis of variance (ANOVA), and finally fitting and interpreting linear regression models. This module takes nine 50 minutes classes on average.

Teaching Methods. The purpose of our research was to understand the benefits and limitations of different pedagogical methods. Each method incorporated theoretical concepts for a module along with the appropriate RStudio functions for that module. All students in this course were exposed to the same teaching method for each module. For this study, the Modeling Process Quality module was taught using the Instructor Guided Method and the module on Interpreting Process Quality was taught using the Think-Pair-Share Method.

- **Instructor Guided Method:** The students were given tutorials to understand the RStudio applications at the beginning of the module. After each topic, the instructor showed the students how to use RStudio to apply these concepts computationally. All the instructional materials were recorded in zoom for students to review later on. This is a traditional method of instruction used in many classrooms.

- **Think-Pair-Share Method:** The students were given tutorials to understand the RStudio applications at the beginning of the module. The lectures were focused on the conceptual material only and students were required to develop their understanding in RStudio using the given readings. There was dedicated class time for students to ask questions and have discussions in groups of 4-5 students with instructor supervision. These discussion groups would not be recorded. This method is utilizing active learning techniques in the classroom that allow students to learn from their peers.

Data Collection Tools: We collected multiple points of data including a survey of the student background, self-efficacy surveys, and assessments of students' conceptual knowledge as well as computational abilities.

> **1) Background Survey:** The students' name and other identifying information were included as well as their statistical self-efficacy and experience in prior statistics courses. We were curious if students had been previously exposed to the topics in this course and if they believed this course would be useful for their future studies and career. This information was used to help us better understand our study population and allow us to draw more accurate conclusions from the data.

2) Self-Efficacy Survey: Self-efficacy statements regarding three different aspects of the course - statistics, RStudio, and theory - were included in the survey. Since each method involved RStudio and theory, it was crucial to look at each of those measures in the survey. Likewise, quality control is part of the statistics curriculum for industrial and systems engineering so we found it important to include a few statements on statistics self-efficacy as well.

A. **Statistics self-efficacy:** The survey asked students to reflect on their comfort with statistics as well as their feelings about learning more statistics. Although quality control is a specific type of statistics, we were interested in how the course changes students' perspectives on statistics in general.

B. **RStudio self-efficacy:** The survey asked students to evaluate their own technical skills in RStudio. It also addressed students' confidence in connecting statistical theory to the RStudio syntax.

C. **Theory self-efficacy:** In addition to general statistics and RStudio, statements are included specifically about the theory learned in class. Not only were we concerned about how well students could learn theory, but we were also interested in if they enjoyed learning and solving these new problem types.

The self-efficacy survey included several statements adapted from a multitude of sources as well as several newly developed statements which were reported in Table 1. For the self-efficacy scales, statements were used from five different sources. Each statement was measured on a 5-point scale with one being "strongly agree" and five being "strongly disagree".

Table 1: References for statements used in self-efficacy survey

Survey Sections	Statements	Reference
Self-efficacy for theory	I can learn the content related to…	[8]
	I am interested to learn more about…	--
	I have enjoyed studying…	[9]
	I can identify when a problem requires the use of…	--
	I like trying to solve new problems using…	[9]
Self-efficacy for RStudio	I can write syntactically correct statements in R	[10]
	I can identify and correct errors in my R code	--
	I am confident I can explain how to approach … problems in R to my classmates	[11]
	I can correctly interpret the outputs of my code in R	--
	The use of R helped my understanding of…	[6]
Self-efficacy for statistics	I can learn the statements taught in my statistics course	--
	I am interested in learning new topics in statistics related to ISE	--
	I have enjoyed studying statistics in school	[9]

3) Conceptual/Computational Assessment: This assessment tested students on the course content after exposing them to the theory and RStudio component of the module they were learning. This was a formative assessment in the form of a low-stakes quiz. Questions for this assessment were either taken from the course textbook [12] or created with the instructor of the course. Questions fell into two categories: Theoretical or RStudio. Although different modules will be addressed, the structure and length of this assessment will be kept at a similar level for each method.

Data Collection Procedure. To remove any links to identifiers, random and unique IDs were assigned to each student. This ID was used for all the surveys collected during this research. One of the research group members received all surveys, assigned a unique ID to each student, and removed the student names prior to sharing with the rest of the research group.

Throughout the semester, data was collected at multiple points. At the beginning of the course, students took the Background Survey. After the Background Survey, the different methods were implemented for different course modules. For each method, once both concepts and RStudio software were taught to the students, the students took both the Self-Efficacy Survey and the Conceptual/Computational Assessment for that module. Table 2 shows the classification of the questions in the assessment for both the Instructor Guided Method and Think-Pair-Share Method.

Table 2: Classification for each question in post exposure assessment

Question #	Instructor Guided Method	Think-Pair-Share Method
1	Theoretical	RStudio
2	RStudio	Theoretical
3	RStudio	RStudio
4	Theoretical	RStudio
5	RStudio	Theoretical

Results

For our research, we collected two types of data. The first dataset focused on student self-efficacy which we broke down into three sections: RStudio, statistics, and theory. All statements for the self-efficacy survey were measured on a 5-point scale with 1 representing “strongly agree” and 5 representing “strongly disagree”. The second dataset focused on the conceptual understanding and computational skills of the students**.**

Self-efficacy Analysis. We ran t-tests for each statement, each type of self-efficacy, and the total self-efficacy between the methods (Table 3).

A. **Statistics self-efficacy:** We assessed self-efficacy for statistics with three statements. As shown in Figure 3, students were more confident in their ability to learn after the Instructor Guided Method. However, there was not a significant difference between the methods for each statement or statistics self-efficacy as a whole.

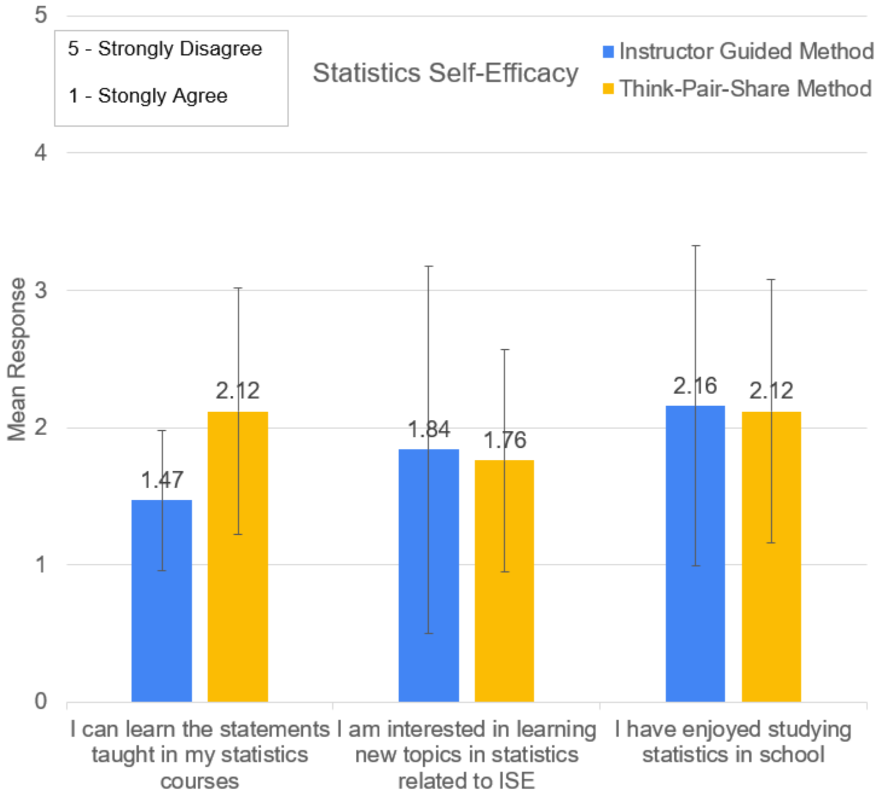

Figure 3: Responses to statistics self-efficacy statements

B. **RStudio self-efficacy:** We assessed self-efficacy for RStudio with seven statements. These statements were tailored for the specific model that students were currently working on. To show the comparison between modules, the module specific phrasing has been removed and the results are shown in Figure 4. As shown, students liked using RStudio to solve problems in both methods. However, for the Instructor Guided Method, they are less confident in their ability to write/interpret the code and no student “strongly agreed” that they could explain topic-specific statements in RStudio to their classmates. On the other hand, after the Think-Pair-Share Method, students were more confident in their ability to write and interpret the code. In fact, a couple of students even “strongly agreed” that they could explain topic-specific statements in RStudio to their classmates. There was not a significant difference between the methods for each statement or RStudio self-efficacy as a whole.

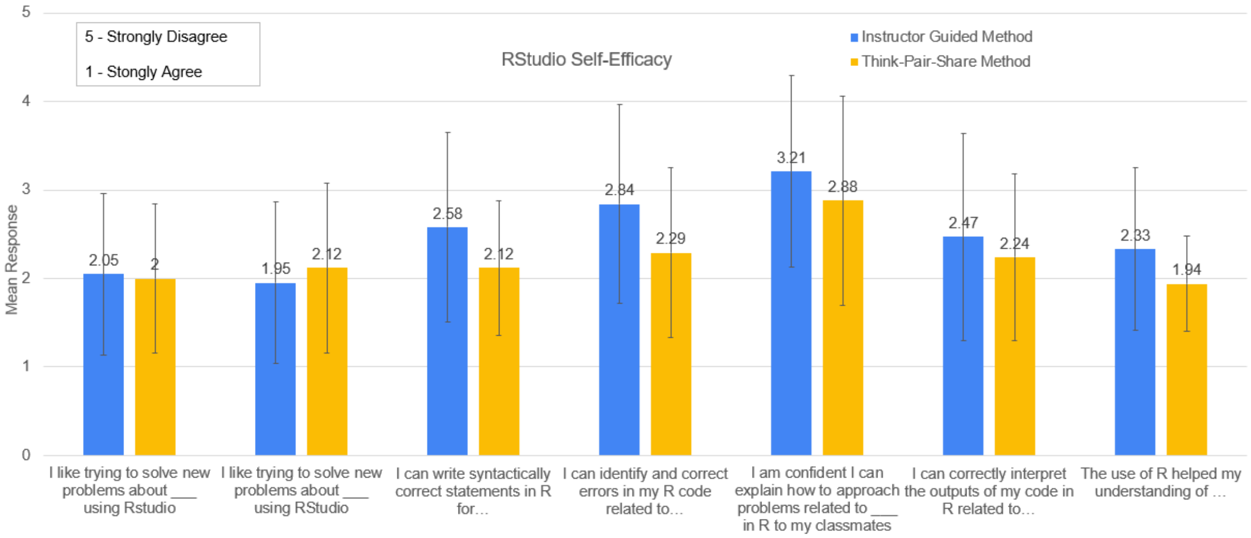

Figure 4: Responses to RStudio self-efficacy statements

C. **Theory self-efficacy:** We assessed self-efficacy for statistics with six statements with the results shown in Figure 5. Students enjoyed studying the topics in each module at about the same level. Although the students did not like solving problems from either module by hand, they were able to identify the correct approach better for the problems after the Think-Pair-Share Method. This is shown in the mean response for "I can identify what _ the problem requires the use of " which is 0.25 lower for the Think-Pair-Share Method. In addition, they felt more able to learn the content using the Think-Pair-Share Method. There was not a significant difference between the methods for each statement or theory self-efficacy as a whole.

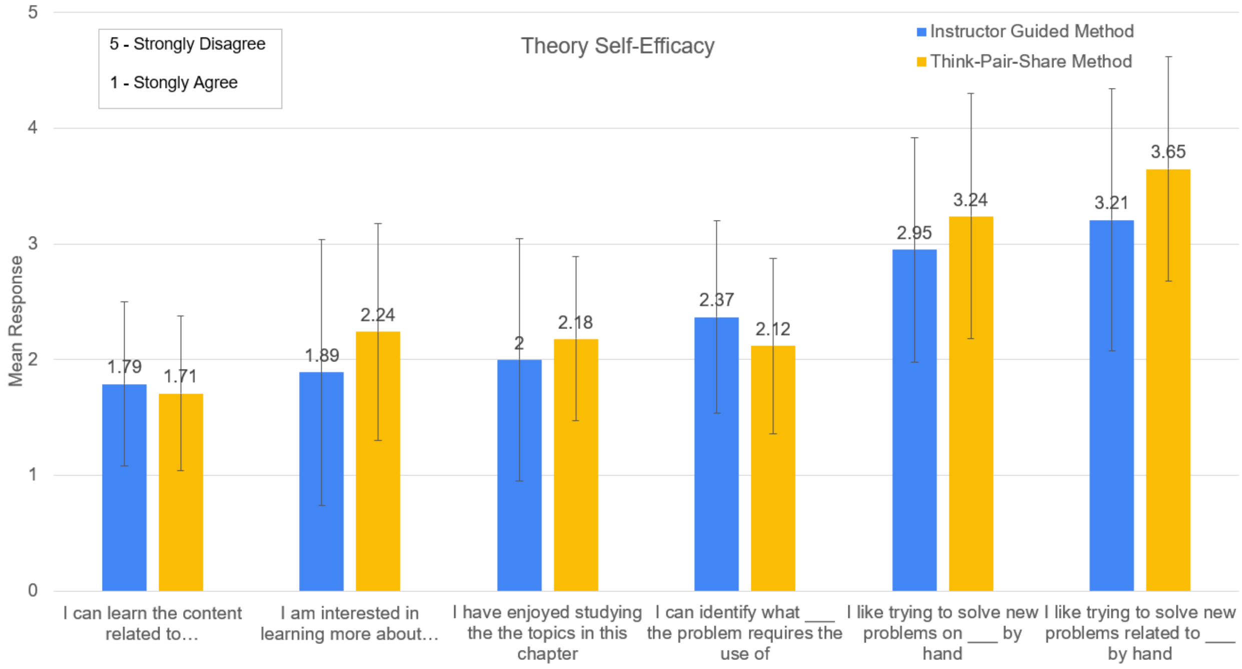

Figure 5: Responses to theory self-efficacy statements

Self-Efficacy Comparison. Table 3 shows the t-test results or each self-efficacy question as well as the total responses for the methods - combining results from theory, statistics, and RStudio self-efficacy. As you can see in the table, there was no significant difference found between the methods.

Table 3: Results of t-tests

		Instructor Guided Method			Think-Pair-Share Method				
Self-Efficacy Type	Question	Sample Size	Mean	Std. Dev.	Sample Size	Mean	Std. Dev.	t_o	p-value
Statistical Self-Efficacy	Q1	19	1.470	0.513	17	2.120	0.900	-0.861	0.272
	Q2	19	1.840	1.344	17	1.760	0.810	0.042	0.396
	Q3	19	2.160	1.167	17	2.120	0.960	0.024	0.396
	Total	19	1.824	1.144	17	1.980	0.860	-0.096	0.394
RStudio Self-Efficacy	Q1	19	2.050	0.911	17	2.000	0.840	0.038	0.396
	Q2	19	1.950	0.911	17	2.120	0.960	-0.130	0.393
	Q3	19	2.580	1.071	17	2.120	0.760	0.301	0.378
	Q4	19	2.840	1.119	17	2.290	0.960	0.344	0.373
	Q5	19	3.210	1.084	17	2.880	1.180	0.212	0.387
	Q6	19	2.470	1.172	17	2.240	0.940	0.137	0.392
	Q7	19	2.330	0.918	17	1.940	0.540	0.299	0.378
	Total	19	2.336	1.027	17	2.227	0.952	0.074	0.395
Theoretical Self-Efficacy	Q1	19	1.790	0.713	17	1.710	0.670	0.078	0.395
	Q2	19	1.890	1.150	17	2.240	0.940	-0.213	0.387
	Q3	19	2.000	1.054	17	2.180	0.710	-0.120	0.393
	Q4	19	2.370	0.831	17	2.120	0.760	0.210	0.387
	Q5	19	2.950	0.970	17	3.240	1.060	-0.208	0.387
	Q6	19	3.210	1.134	17	3.650	0.970	-0.271	0.381
	Total	19	2.313	1.119	17	2.520	1.106	-0.129	0.393
	Total Self-Efficacy	19	2.272	1.110	17	2.297	1.001	-0.012	0.396

Conceptual Data Analysis: The first part of our conceptual analysis was to analyze the correctness of students' responses. For each method, the Conceptual/Computational Assessment had 5 questions, testing the understanding of students of the theoretical and practical concepts. If the question was answered correctly, it received a score of 1 and the question received a score of 0 if it was incomplete or left blank. After this, we calculated the average of correct answers that each student had in each of the methods.

A. **Instructor Guided Method:** The assessment for the Instructor Guided Method was completed by 22 students. On average, students got 2.4 out of the 5 questions correct. Figure 6 shows the percentage of students that had a certain number of correct answers in this assessment overall. For example, 13.6% of students did not get any questions correct and 4.5% got all the questions for this module correct. Per question type, 56.8% of the students got the correct answers for the theoretical questions and 42.4% of students got the correct answers for the RStudio questions.

B. **Think-Pair-Share Method:** The assessment for the Think-Pair-Share Method was completed by 33 students. On average, students got 1.7 questions out of 5 correct. Figure 6 shows the percentage of students that had a certain number of correct answers in this survey. For example, 3% of students that did this survey (1 student) got all questions correct and 15% (5 students) got 0 questions correct. Per question type, 37.9% of students correctly answered the theoretical questions and 31.3% of students correctly answered the RStudio questions.

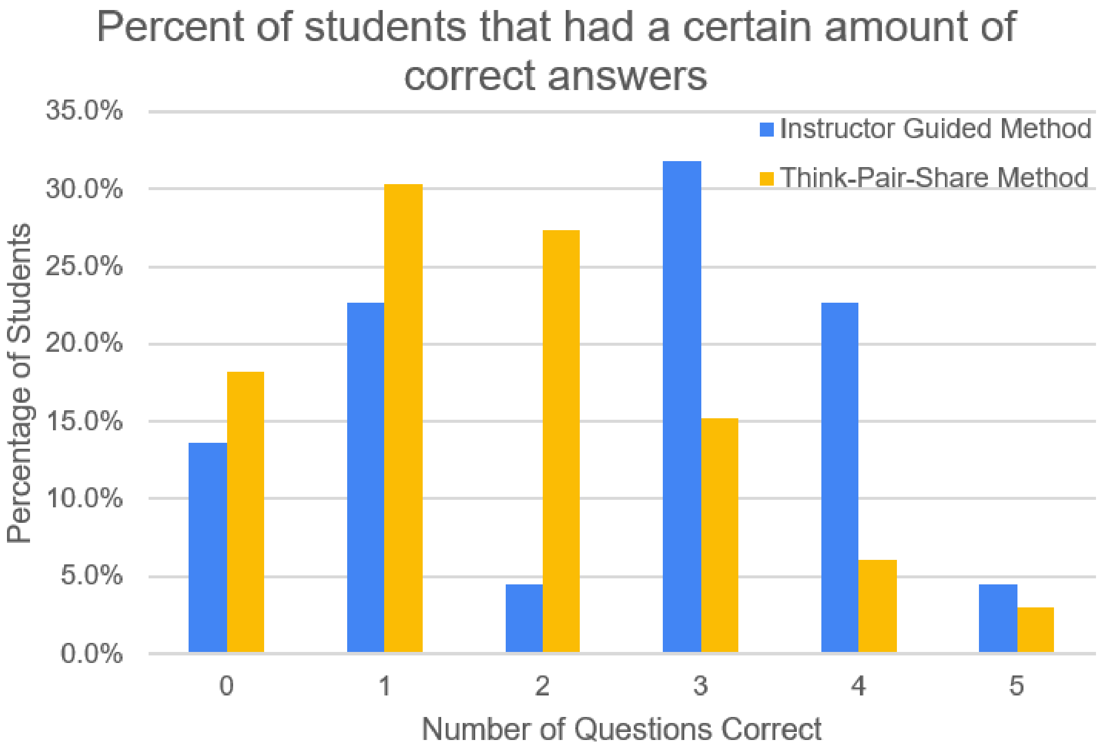

Figure 6: Percentage of students that had a certain amount of correct answers

Conceptual Comparison: We ran t-test for the questions classified as: Theoretical, RStudio questions, as well as both combined. The results from the analysis of the difference in means between the Instructor Guided Method and the Think-Pair-Share Method are shown in Table 4. Three separate analyses were done, one for the questions that where classified as theoretical, another for the computational questions that were solved by students using RStudio, and the last one included both. The test statistic (t_0) and the p-value were calculated using the RStudio software. The p-value in the three scenarios is greater than 0.05 which indicates that there is no statistically significant difference in the methods used. However, it is a point of note that in all three scenarios, the mean for the Instructor Guided Method is higher than the mean for the Think-Pair-Share Method.

Table 4: Results of Inference for a Difference in Means

	Analysis					
	Theory		**R Studio**		**Both**	
n1 = 21 n2 = 32	Instructor Guided Method	Think-Pair-Share Method	Instructor Guided Method	Think-Pair-Share Method	Instructor Guided Method	Think-Pair-Share Method
mu	0.568	0.409	0.386	0.318	0.482	0.385
sigma	0.362	0.320	0.206	0.220	0.061	0.186
t_o	0.346		0.259		1.543	
p-value	0.374		0.384		0.121	

Discussion

Based on our preliminary data, there was no statistically significant difference between the two methods implemented. In general, the Think-Pair-Share Method seems to result in a higher RStudio self-efficacy than the Instructor Guided Method. In fact, the statements with the largest differences in mean response between the two methods were “I can write syntactically correct statements in R for…” and “I can identify and correct errors in my R code related to…”. The mean differences are 0.46 and 0.55, respectively. While the other statements from the survey are still important, these two are the most directly tied to RStudio comprehension. If a student feels comfortable using the RStudio syntax and feels as if they can debug their code, then they are in a good position in the class. A possible explanation of this could be found in the nature of the Think-Pair-Share Method, which is based around discussion and work with peers. This is an active learning technique and allows students to teach and learn from their peers. For learning such difficult material, students may perform better when given the chance to talk things through with other students who are in the same position.

On the other hand, the Instructor Guided Method only allows instructor-student interaction and does not set aside time for group work. In addition, the lectures are recorded and available for students to review on their own time. The reality is that students may procrastinate on reviewing the recorded lectures which is not a good study habit, especially in a class heavy in theory, like quality control. Again, there was no statistically significant difference, but this was generally reflected in the results, as students responded less positively to the self-efficacy statements for the Instructor Guided Method. The lack of self-efficacy is most evident in the statement, “I am confident I can explain how to approach problems related to ___ in R to my classmates", which had a mean response of 3.21. Such a response reflects that students may not have spent as much time studying on their own time. It could also be a reflection of class time being used ineffectively.

The data for theory self-efficacy and statistics self-efficacy does not show a very large difference between the two methods. While the Think-Pair-Share Method was consistently more effective for RStudio self-efficacy, that success did not carry into theory and statistics self-efficacy. Students were more discouraged by the concepts under the Think-Pair-Share Method as the mean response for “I can learn the statements taught in my statistics course” was 0.65 higher than for the Instructor Guided Method. Perhaps that can simply be attributed to the increasing difficulty of the material as the course progresses. Regardless, we want to keep students engaged in class so seeing this negative response is a concern. An interesting trend arises in the statements “I like trying to solve new problems on _ by hand”. Both methods had high average responses for these statements, signaling that students would much prefer using RStudio while solving problems. One of the main difficulties in teaching software is keeping students from becoming dependent on it. It is more likely for students to be dependent on software if they do not enjoy solving the problems by hand, which seems to be the case here.

The data for the conceptual survey does not show a statistically significant difference between the two methods. While the questions given after the Instructor Guided Method presented a higher mean of correct answers than the questions given after the Think-Pair-Share Method, there was a very small difference between them. This could be since the material covered with the Instructor Guided Method is considered to be easier for students as it is an introduction to quality control.

While RStudio and graphing calculators have many differences, they are both technologies that serve a purpose in the classroom. Our findings have some similarities with [2, 3], which both looked at calculators in pre-college settings, in that students in our sample may be developing a dependence on the technology. Also, students’ RStudio self-efficacy appearing to be slightly greater but not significantly for the Think-Pair-Share Method could be explained by the Constructivist Learning Theory (CLT) presented in [1]. CLT is the idea that students learn better when they are active in the learning process rather than receiving information passively. This is the case of the Think-Pair-Share Method where students are given the opportunity to be involved in discussions during class. However, despite some interesting trends in our data, the lack of statistical significance and presence of statistical noise makes it difficult to state any new findings with certainty.

Conclusion

The aim of this ongoing study is to determine the optimal pedagogical approach for introducing RStudio in a quality control classroom. In this study, we applied two methods including Think-Pair-Share Method and Instructor Guided Method. Although the results were not significantly different for all the self-efficacy surveys, for RStudio self-efficacy surveys the Think-Pair-Share Method had slightly higher results than the Instructor Guided Method. Looking at the self-efficacy data and the conceptual/computational assessment data together, we cannot conclude that one method is better than the other. Besides all these findings, this study had some limitations. The first limitation was the sample size. Not all students who enroll in this class, did participate in the self-efficacy surveys and conceptual/computational assessments for both methods. The second limitation was related to the modules assigned to these assessments. Although we assigned them randomly, the difficulty of the class tends to increase throughout the course. Therefore, the teaching method might not be the only reason for getting lower or higher results.

To address these limitations for future research we suggest motivating students to participate by allocating some points to these assessments. Having a larger sample size will allow us to have a better understanding of the impact of the teaching methods in students. To mitigate the second limitation, we suggest that to conduct this study in multiple semesters and reassign the modules to the pedagogical approaches to ensure different coverage. In addition to continuing this research in the quality control course, the research team plans to expand research coverage to the prerequisites of this course to conduct more of a longitudinal study and see how students' progress throughout their statistics courses in the curriculum.

References

[1] Tajuddin, N. M., Tarmizi, R. A., Mohd K., & Wan, Ali. (2009). Instructional efficiency of the integration of graphing calculators in teaching and learning mathematics. *International Journal of Instruction* (2).

[2] Rohrabaugh, A. P. and Cooper, C. M. (2016). Calculators in high school classrooms. *Honors Research Projects*, 313.

[3] Sheets, C. L. (2007). Calculators in the classroom: Help or hindrance? *Action Research Projects*. 33.

[4] Mao, Yi & White, Tyreke & Sadler, Philip & Sonnert, Gerhard. (2017). The association of precollege use of calculators with student performance in college calculus. *Educational Studies in Mathematics*. 94. 10.1007/s10649-016-9714-7

[5] Tiwari, S., & Rathore, S. "Teaching software process models to software engineering students: An exploratory study," in *2019 26th Asia-Pacific Software Engineering Conference (APSEC)*, Putrajaya, Malaysia, pp. 308-315. doi: 10.1109/APSEC48747.2019.00049

[6] Canizares, C. A., & Faur, Z. T. (1997). Advantages and disadvantages of using various computer tools in electrical engineering courses. IEEE Transactions on Education, 40(3), 166–171. https://doi.org/10.1109/13.618025

[7] Carey, M. P., & Forsyth, A. D. (2009). *Teaching tip sheet: Self-efficacy*. American Psychological Association. https://www.apa.org/pi/aids/resources/education/self-efficacy

[8] Mamaril, N. A., Usher, E. L., Li, C. R., Economy, D. R., & Kennedy, M. S. (2016). Measuring undergraduate students' engineering self-efficacy: A validation study. Journal of Engineering Education, 105(2), 366–395. https://doi.org/10.1002/jee.20121

[9] Austin, J. S. (1996). Effect of graphing calculator use on student achievement in college algebra: Gender and age-related differences. (Order No. 9703500, University of Missouri - Kansas City). *ProQuest Dissertations and Theses*, 135.

[10] Korkmaz, Ö., & Altun, H. (2014). Adapting computer programming self-efficacy scale and engineering students' self-efficacy perceptions. *Participatory Educational Research*, 20–31. https://doi.org/10.17275/per.14.02.1.1

[11] Woolcock, A. D., Creevy, K. E., Coleman, A. E., Moore, J. N., & Brown, S. A. (2016). Assessing academic self-efficacy, knowledge, and attitudes in undergraduate physiology students. *American Journal of Educational Research, 4*(9), 652–657.

[12] Montgomery, D. C. (2020). *Introduction to statistical quality control*. Hoboken,NJ: Wiley.