Comparing ChatGPT and Traditional Online Search for Learning R Programming and RStudio Visualization: A Pilot Study

Amit Lal, University of the Pacific, United States

The Washington DC Conference on Education 2026
Official Conference Proceedings

Abstract

This pilot study explores the comparative effectiveness of ChatGPT versus Traditional Online Search (TOS) methods in supporting novice second-year undergraduates with no prior R or RStudio experience in completing a data visualization assignment. Participants (n = 20) were randomly assigned to two groups (n = 10 each) and given approximately 1 hour and 50 minutes across two class periods to load a built-in dataset, explore its structure, and create visualizations using either ChatGPT or TOS. Assignments were graded on code accuracy, use of RStudio features, visualization quality, documentation, and presentation. Post-assignment surveys assessed perceived ease of use, clarity of guidance, engagement, and understanding of R programming and visualization techniques. Preliminary results indicate the ChatGPT group significantly outperformed the TOS group in assignment scores and survey ratings (e.g., higher means in ease of use and understanding), aligning with expected effects of large language models in reducing cognitive load through personalized guidance. Qualitative feedback underscored ChatGPT's interactivity versus TOS's information overload. Distinct from broader meta-analyses, this pilot provides initial classroom-specific insights into AI advantages for short-term programming learning. Implications include hybrid AI-TOS frameworks to address overreliance, but these frameworks have limitations, including a small sample size and no long-term assessment. Future studies should be scaled with controls and pretests.

Keywords: study effectiveness, ChatGPT, artificial intelligence, R programming, R visualization learning

iafor

The International Academic Forum
www.iafor.org

Introduction

The increasing ubiquity of data-driven decision making across sectors has intensified the demand for graduates proficient in statistical reasoning, programming, and data visualization. Tools such as R and RStudio have become foundational to modern analytics curricula, yet they pose steep learning curves for students encountering coding for the first time. Despite instructors' efforts to scaffold learning, novice programmers frequently seek supplemental guidance through external digital resources.

For decades, Traditional Online Search (TOS) methods—general search engines, forums, blogs, and community Q&A platforms—have served as primary self-directed learning tools. These resources offer breadth and accessibility, but they also require students to navigate heterogeneous content, evaluate credibility, reconcile conflicting explanations, and synthesize fragmented information. The cognitive burden associated with these tasks can impede early learning, particularly for students with limited programming background.

Concurrently, rapid advancements in artificial intelligence have generated new opportunities for personalized, context-aware learning support. Large language models (LLMs) such as ChatGPT are capable of generating tailored explanations, step-by-step debugging guidance, and adaptive examples. Prior research highlights the promise of such tools in reducing cognitive load, promoting engagement, and enabling novices to progress more quickly through early conceptual and syntactic hurdles. Yet valid concerns remain regarding accuracy, potential misinformation, and the risk that students may adopt AI-generated solutions without fully understanding underlying logic.

Against this backdrop, there is a growing need for empirical classroom-based comparisons of AI-driven tools and traditional online research practices. While conceptual discussions of AI in education are increasingly common, there remains limited research directly evaluating the educational impact of ChatGPT relative to TOS tools within authentic instructional settings—particularly in programming domains where detail-oriented error resolution and conceptual clarity are essential.

This study investigates how ChatGPT and TOS tools influence novice learners' performance, perceptions, and learning experiences during R programming and data visualization tasks. By combining quantitative performance metrics with qualitative insights into learners' cognitive strategies and resource interactions, this research provides evidence-based guidance for educators considering the integration of AI-driven tools into analytics curricula. The findings also illuminate pedagogical opportunities and risks associated with LLM-supported instruction in technical domains.

Literature Review

Traditional Online Search as a Learning Resource

For decades, Traditional Online Search (TOS) tools—such as Google, Bing, Yahoo, and legacy engines like AltaVista—have supported self-directed learning across disciplines. These platforms grant students broad access to tutorials, academic articles, coding forums, and videos, facilitating autonomy in navigating unfamiliar concepts (Greenhow et al., 2009; Head & Eisenberg, 2009). In introductory programming contexts, learners frequently turn to TOS methods to locate syntax examples, troubleshoot errors, and explore visualization techniques.

The flexibility and immediacy of these tools make them appealing for resolving targeted questions during homework or laboratory activities.

However, TOS methods also introduce notable challenges. Research shows that novice learners often encounter information overload, inconsistent resource quality, and conflicting explanations (Chen et al., 2020; Yim & Su, 2024). These issues require users to independently evaluate credibility, reconcile contradictory advice, and synthesize fragmented content across multiple sources—tasks that impose substantial extraneous cognitive load. For beginners in R programming, even small inconsistencies across tutorials may hinder progress, slow debugging, and reduce conceptual clarity. The unstructured nature of TOS-mediated learning may also encourage surface-level engagement, where students adopt code samples without fully understanding underlying logic or design principles (Adel & Ahsan, 2024).

While TOS tools remain foundational for open-ended exploration, their limitations become particularly acute in early programming education. As R and RStudio demand precision, unfamiliar syntax, and iterative problem-solving, novice learners may struggle to navigate the breadth of online content effectively, highlighting a need for more adaptive and context-sensitive guidance.

Emergence of AI and Intelligent Tutoring Systems in Education

Artificial intelligence (AI) has gained increasing attention as a means to enhance personalized learning and reduce cognitive barriers in complex subjects. Intelligent Tutoring Systems (ITS) and AI-enhanced instructional platforms have demonstrated the ability to offer real-time feedback, diagnose misconceptions, and support iterative mastery of skills (Chen et al., 2020). In STEM fields, AI-based tools have been shown to guide students through multi-step reasoning tasks, clarify errors, and provide tailored support aligned with learners' individual needs (Yim & Su, 2024). These adaptive systems reduce the burden of navigating disparate resources and allow learners to focus on core conceptual understanding.

The scalability of AI tools further benefits institutions by supporting large student cohorts while freeing educators to focus on higher-order instruction and individualized coaching (Adel & Ahsan, 2024). Despite these advantages, concerns persist regarding data privacy, model transparency, and the risk that AI-guided learning may supplant rather than strengthen critical thinking (Bender et al., 2021). These tensions underscore the need for empirical research clarifying the conditions under which AI enhances, complements, or potentially undermines traditional learning practices.

Large Language Models and the Pedagogical Role of ChatGPT

Large language models (LLMs), such as GPT-3 and GPT-4, represent a major evolution in AI capabilities. Trained on massive corpora of text, these models generate contextually relevant and syntactically coherent responses, enabling applications ranging from writing assistance to real-time conversational tutoring (Bommasani et al., 2021; Brown et al., 2020). Their flexibility makes them uniquely suited for novice programmers who require immediate, interpretable, and example-driven explanations.

ChatGPT, one of the most widely adopted LLM tools, provides several affordances relevant to programming education. First, it can explain syntax and functions in plain language, offering scaffolded support for learners encountering R commands, visualization libraries, or data structures for the first time. Second, ChatGPT can debug code by identifying likely error sources and generating corrected versions or annotated explanations (Rudolph et al., 2023). Third, by leveraging few-shot learning, the model adapts its responses to user-provided context, enabling personalized guidance on assignment-specific tasks.

Early empirical studies suggest that ChatGPT enhances learning efficiency, reduces confusion, and helps students persist through error-prone coding exercises (Dempere et al., 2023). Its conversational format encourages iterative exploration and lowers the barrier to asking clarifying questions—an advantage over traditional search tools where novices may struggle to phrase queries using domain-appropriate terminology. However, LLM reliability remains uneven, and ChatGPT may produce plausible but incorrect outputs. These “hallucinations” pose risks for learners who lack the expertise to validate technical accuracy (Bender et al., 2021).

Thus, while ChatGPT offers transformative instructional potential, its integration into education requires balancing efficiency with safeguards that encourage critical verification.

Cognitive Load Considerations and Technology Acceptance

The effectiveness of both TOS and AI-driven tools can be interpreted through Cognitive Load Theory (Sweller, 1988) and the Technology Acceptance Model (Davis, 1989). According to Cognitive Load Theory, learning improves when extraneous cognitive burdens are minimized. TOS methods often increase such load by requiring learners to reconcile conflicting information and navigate multiple sources. In contrast, ChatGPT reduces extraneous load by providing unified, context-specific guidance, enabling students to allocate cognitive resources toward meaningful learning (Mayer & Moreno, 2003).

The Technology Acceptance Model suggests that perceived usefulness and ease of use are central determinants of whether learners adopt new tools. ChatGPT’s intuitive conversational interface, rapid feedback, and personalized support align strongly with these determinants, potentially increasing student confidence and engagement (Siau & Yang, 2017). Conversely, TOS tools offer autonomy but may frustrate novices who lack experience evaluating complex or inconsistent resources.

Together, these frameworks help explain the observed variance in learner experience between AI-assisted and traditional search-based approaches, and they highlight why empirical evidence is needed to guide educators in selecting and integrating technological tools.

Identified Gaps and Rationale for This Study

Despite a growing body of conceptual literature on AI in education, empirical comparisons between ChatGPT and TOS tools remain limited—especially within authentic classroom environments. Most existing work is anecdotal, small-scale, or focused on generalizable AI capabilities rather than direct performance outcomes in programming assignments (Almasri, 2024; Zawacki-Richter et al., 2019). Furthermore, research rarely examines R programming specifically, despite its importance for data literacy and analytics education.

This study addresses these gaps by providing:
- A controlled comparison of ChatGPT and TOS tools in a real university classroom
- An analysis of both performance outcomes and learner perceptions
- Thematic insights into students' strategies, challenges, and cognitive workload
- Evidence-based implications for curriculum design and AI integration

By synthesizing the strengths of AI support with the realities of novice programming behavior, the study advances ongoing conversations about how LLMs can be responsibly and effectively incorporated into higher education.

Methods

Research Design

This study employed a quasi-experimental, mixed-methods design to compare the effectiveness of ChatGPT and Traditional Online Search (TOS) tools in supporting novice learners' completion of R programming tasks. The design integrates quantitative measures of student performance with qualitative insights into learner experiences, reflecting recommended approaches for evaluating emerging educational technologies in authentic contexts (Adel & Ahsan, 2024; Chen et al., 2020).

A mixed-method framework was chosen for three reasons:
1. performance on programming assignments provides objective evidence of learning outcomes;
2. surveys capture students' perceived ease of use, clarity, engagement, and conceptual understanding; and
3. open-ended reflections reveal cognitive strategies, frustrations, and resource-mediated behaviors not captured in quantitative metrics.

The study spans two consecutive class sessions, each lasting 1 hour and 50 minutes. Both groups completed the same assignment in the same timeframe, ensuring consistent conditions aside from the designated research tool.

Participants and Setting

Participants were 20 second-year undergraduates enrolled in a data analytics course at a four-year institution. None had prior experience with R or RStudio. The sample reflects the university's status as a minority-serving institution, with a large proportion of Hispanic students and a mix of male, female, and non-binary participants.

After providing informed consent, students were randomly assigned to one of two conditions:
- **ChatGPT group (n = 10):** Students used ChatGPT exclusively for all conceptual and troubleshooting queries.
- **TOS group (n = 10):** Students relied solely on traditional online resources such as Google, Bing, Yahoo, and instructional blogs.

Random assignment minimized self-selection bias and facilitated clearer attribution of differences in performance or perceptions to the tool condition (Almasri, 2024).

To ensure compliance, students were instructed to document their information sources—ChatGPT transcripts for the AI group, and URLs for the TOS group. Instructor monitoring further discouraged cross-use of tools during the study period.

Assignment Task

The assignment required students to complete a sequence of foundational R programming tasks designed to evaluate data loading, exploration, visualization, and documentation skills. Students were instructed to:

1. Load a built-in dataset in R.
2. Explore the dataset using functions such as head(), str(), and summary().
3. Construct multiple visualizations (e.g., scatterplots, bar charts, histograms, boxplots) using base R or ggplot2.
4. Save and submit their R scripts and resulting figures.

This assignment structure reflects common early-course expectations in statistics and data analytics education (Wang et al., 2017). The tasks were equally accessible to both groups but required sufficient comprehension and troubleshooting skills to avoid common syntax errors and plotting issues.

Assessment Rubric

Assignments were evaluated prior to reviewing survey responses to maintain objectivity. A rubric scored students across five criteria:

1. **Code accuracy (30%)**—correctness and reliability of R code, including successful execution without errors.
2. **Use of RStudio features (20%)**—effective use of script files, data exploration tools, and relevant packages.
3. **Visualization quality (20%)**—clarity, variety, labeling, and relevance of graphical outputs.
4. **Documentation and explanation (10%)**—use of comments, explanations of steps, and overall clarity.
5. **Overall presentation (10%)**—professionalism, layout, and completeness of submitted materials.

Scores were scaled to 100 points. This rubric aligns with best practices in evaluating introductory programming assignments in higher education.

Data Collection Instruments

Two forms of data were collected.

Surveys

Students completed a post-assignment survey consisting of single-item Likert-scale measures (1 = strongly disagree; 5 = strongly agree) assessing:

- Perceived ease of use
- Clarity of guidance
- Engagement
- Perceived understanding of R programming and visualization

Single-item measures were used for efficiency and are consistent with exploratory educational technology research (Yim & Su, 2024). Students also provided short, open-ended reflections on the strengths, challenges, and desired improvements of the tool they used.

Assignment Submissions

The graded assignments provided the primary quantitative outcome measure. These scores were later used for between-group comparisons and correlation analyses.

Data Analysis Procedures

Quantitative Analysis

Descriptive statistics (means, medians, standard deviations) were computed for both survey items and rubric scores. Since the small sample size (n = 10 per group) violated assumptions of normality and homogeneity of variance, Mann-Whitney U tests were used to compare groups across survey metrics. Effect sizes were interpreted using standard thresholds (Z/(n1 + n2): small < 0.3; medium 0.3–0.5; large > 0.5).

Qualitative Analysis

Open-ended responses were analyzed using thematic analysis. A single researcher coded recurring patterns such as:

- Perceived strengths
- Challenges and frustrations
- Cognitive strategies
- Desired improvements

Keyword frequency tables and word cloud visualizations were used to support theme identification. As this was an exploratory classroom study, inter-rater reliability was not computed, consistent with comparable studies (Zawacki-Richter et al., 2019).

Ethical Considerations

All participants provided informed consent. Tool restrictions applied only during the study period and had no bearing on course grades beyond the assignment itself. Data were anonymized and stored securely within institutional data systems. No external funding supported this research.

Results

Descriptive Statistics

Survey responses across the four Likert-scale measures—ease of use, clarity of guidance, engagement, and perceived understanding of R programming and visualization—revealed distinct differences between the ChatGPT and TOS groups. Within each group, response variability was minimal, which reflects both the small sample size (n = 10 per group) and the consistency of participants' experiences.

Across all measures, students in the ChatGPT group reported markedly higher scores. Their ratings clustered around the upper end of the scale (primarily 4s and 5s), indicating uniformly positive experiences with the AI-based tool. In contrast, TOS participants recorded substantially lower scores, concentrated between 2 and 3, with few ratings above 4.

These descriptive patterns were visually represented in box plots (Figure 1), which demonstrated a clear separation between groups. Even prior to inferential testing, the visualized distributions suggested significant differences in user experience across the two tool conditions. The results are summarized in Table 1 below.

Table 1
Descriptive Statistics by Tool Group

Metric	Group	Mean	Median	Std. dev.	Range
1. Ease of use	Traditional search	3.3	3	1.42	1–5
	ChatGPT	4.9	5	0.32	4–5
	Δ	+1.6	+2	-1.10	Lesser
2. Clarity of guidance	Traditional search	3.3	3.5	1.57	1–5
	ChatGPT	4.8	5	0.42	4–5
	Δ	+1.5	+1.5	-1.15	Lesser
3. Engagement	Traditional search	3.1	3	1.37	1–5
	ChatGPT	4.9	5	0.32	4–5
	Δ	+1.7	+2	-1.05	Lesser
4. Understanding R	Traditional search	3.0	3	1.05	1–4
	ChatGPT	4.7	5	0.48	4–5
	Δ	+1.7	+2	-0.57	Lesser

Note. Descriptive statistics for the two research groups show that students consistently and significantly reported improved ease of use, clarity of guidance, engagement, and understanding of R programming and RStudio skills using ChatGPT over Traditional Online Search engines. Δ denotes the difference in scores between the two groups.

Inferential Statistics: Mann-Whitney U Tests

To statistically assess differences in survey responses between conditions, Mann-Whitney U tests were conducted for each of the four metrics. Given the limited sample size and non-normal distributions, the U test provided an appropriate non-parametric alternative to independent-samples t-tests.

Across all four metrics, the analyses yielded statistically significant differences at $\alpha = 0.05$, with the ChatGPT group outperforming the TOS group:

- **Ease of use**: Higher ratings in the ChatGPT group; large effect size
- **Clarity of guidance**: Consistently elevated scores for ChatGPT; large effect size
- **Engagement**: Significantly greater engagement among ChatGPT users
- **Understanding of R programming and visualization**: Robust advantage in favor of ChatGPT

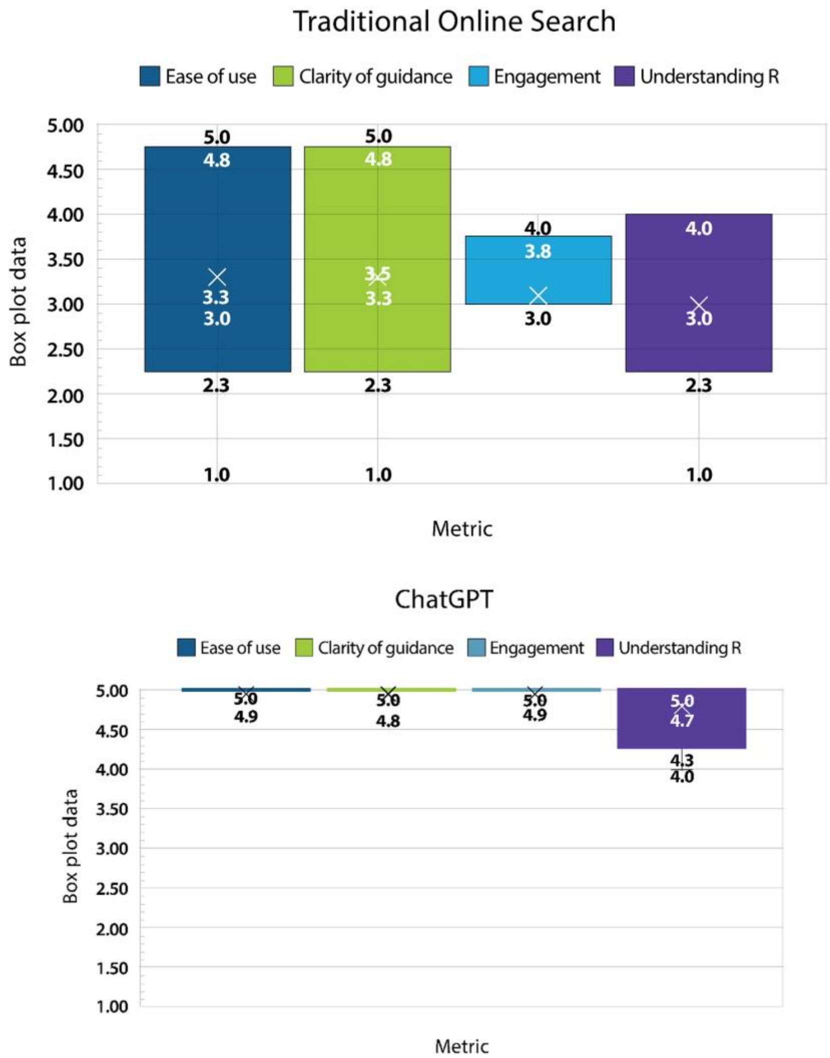
Effect sizes, computed via standardized Z-values, exceeded conventional thresholds for "large" effects. This suggests that the differences between groups were not only statistically significant but educationally meaningful. Table 2 summarizes the corresponding U statistics, p-values, and standardized effect sizes.

Table 2
Mann-Whitney U-Test Parameters by Metric

Metric	P	U (x1>2ave)	Z	Std. effect size
1. Ease of use	0.54%	17%	-2.78	0.62 (large)
2. Clarity of guidance	1.4%	20%	-2.45	0.55 (large)
3. Engagement	0.15%	12%	-3.18	0.71 (large)
4. Understanding R	0.1%	6%	-3.44	0.77 (large)

Note. Inferential statistics for the two research groups are based on students' respective scores for the different metrics calculated at $\alpha = 0.05$. The p-value shows the probability of a type I error (rejecting a correct null hypothesis), while the U value represents the probability that a random rating from a student from the Traditional Online Search group will exceed the average score from the ChatGPT group. The standardized effect size, $Z/\sqrt{(n1+n2)}$, is considered small when it is <0.3, medium when 0.3–0.5, and large when > 0.5.

Figure 1
Survey Rating Distributions by Tool Group



Note. Box-and-whisker plots show scores for the four research questions posed to the two groups of students using the data in Table 1. Metrics are distinguished by color; the left chart pertains to Traditional Online Search survey results, and the right chart presents the equivalent for the ChatGPT group using the same scale for both charts. The feedback scores from the ChatGPT group students, even before the statistical analysis, clearly outperform those from the group using traditional online search methods.

Graded Assignment Performance

Assignment scores further supported the superiority of ChatGPT in promoting successful task completion. Across all five rubric criteria—code accuracy, use of RStudio features, visualization quality, documentation, and overall presentation—the ChatGPT group achieved higher mean scores, with tighter variation and fewer errors.

Figure 2 illustrates overall score distributions, demonstrating that ChatGPT users produced more accurate code, cleaner visualizations, and more complete documentation. Figure 3 presents mean scores by rubric category, highlighting consistent performance advantages. The exact values of all scores are shown in Tables 3–4.

Key differences include:

- **Code Accuracy:** ChatGPT users generated correct and efficient code with fewer execution errors, likely due to real-time debugging support.
- **Visualization Quality:** ChatGPT users produced more complete and informative plots, correctly labeled and aligned with the dataset context.
- **Documentation:** Although documentation improved modestly in both groups, ChatGPT users provided clearer explanations of code steps.
- **Presentation:** Submissions from ChatGPT users displayed more consistent structure and formatting.

Assignment results show that the advantages of ChatGPT were not limited to perceived ease or engagement but translated directly into measurable improvements in academic performance.

Table 3
Assignment Score Summary by Tool Group

Group	Mean	Median	Std. dev.	Range
Traditional Online Search	85	85	7.58	75–99
ChatGPT	91	91	6.23	76–100
Δ	+6	+6	-1.35	Higher

Table 4
Assignment Grading Criteria and Weight Factors

Criterion	Group	Mean	Median	Std. dev.
Code accuracy (30%)	Traditional Online Search	22	21.5	2.5
	ChatGPT	27	27	1.0
	Δ	+5	+5.5	-1.5
Use of RStudio features (20%)	Traditional Online Search	15	15	1.5
	ChatGPT	18	18	1.0
	Δ	+3	+3	-0.5
Data visualization (20%)	Traditional Online Search	20	19.5	2.0
	ChatGPT	23	23	1.5
	Δ	+3	+4.5	-0.5
Documentation & explanation (10%)	Traditional Online Search	12	12	1.25
	ChatGPT	14	14	0.75
	Δ	+2	+2	-0.5

Criterion	Group	Mean	Median	Std. dev.
Overall presentation (10%)	Traditional Online Search	7	7	1.0
	ChatGPT	9	9	0.5
	Δ	+2	+2	-0.5

Figure 2
Overall Assignment Score Distributions

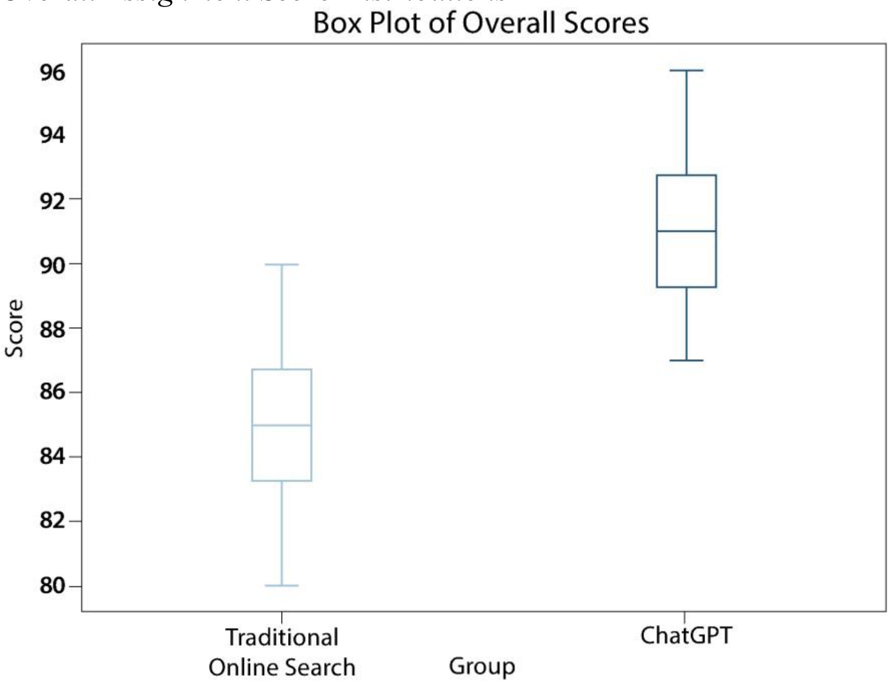
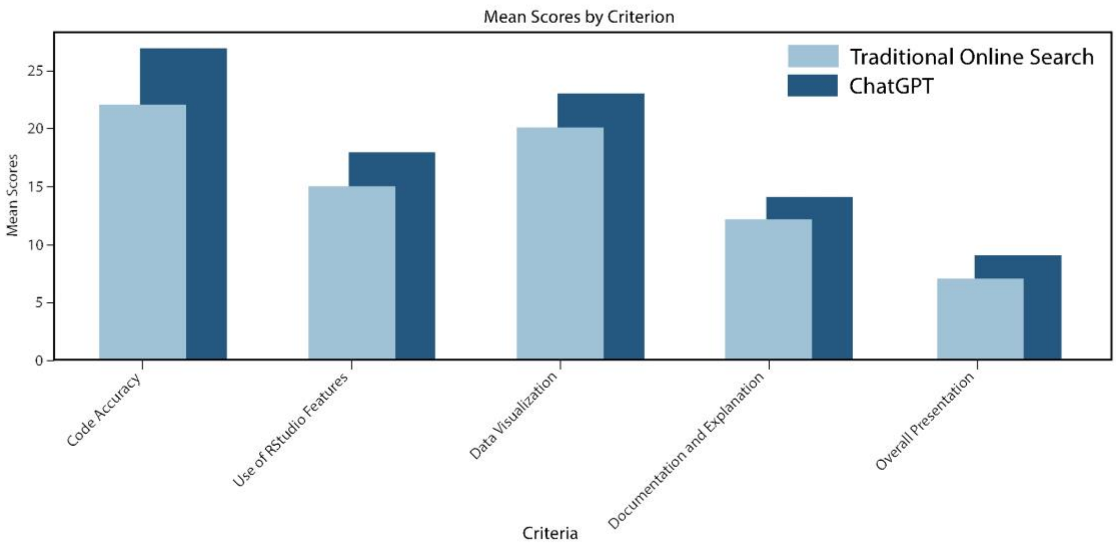


Figure 3
Mean Assignment Scores by Criterion



Correlation Analysis

Correlation matrices were constructed to evaluate relationships between survey metrics (ease of use, clarity, engagement, understanding) and assignment outcomes (code accuracy, feature use, visualization quality, documentation, overall score) within each group.

TOS Group Correlations

In the TOS group, correlations between survey responses and performance metrics were generally moderate and positive, suggesting that students who felt more comfortable and engaged with TOS tools tended to perform slightly better on the assignment. However, the correlations were uneven across categories, reflecting the inconsistent nature of resource quality in traditional search methods. Most notable, we find that:

- Ease of use correlated moderately with code accuracy
- Clarity of guidance correlated with visualization quality
- Engagement showed weak correlations with documentation (consistent with prior findings that documentation skills may be more procedural)

Table 5 presents the correlation coefficients.

Table 5
Correlation Matrix for the Traditional Online Search Group

Metric	Code accuracy	RStudio	Visualization	Documentation	Overall
1. Ease of use	0.45	0.43	0.41	0.40	0.42
2. Clarity of guidance	0.42	0.45	0.43	0.41	0.42
3. Engagement	0.41	0.42	0.44	0.40	0.41
4. Understanding R	0.46	0.44	0.48	0.42	0.45

ChatGPT Group Correlations

In contrast, correlations within the ChatGPT group were uniformly strong and positive. Students who rated ChatGPT highly on any survey dimension almost universally produced stronger assignment performances across all rubric criteria. These strong correlations likely resulted from:

- Highly consistent user experiences with ChatGPT
- Stronger alignment between perceived clarity and actual task success
- Reduced variability in both survey and performance metrics

The correlation matrix for the ChatGPT is shown in Table 6 below.

Table 6
Correlation Matrix for the ChatGPT Group

Metric	Code accuracy	RStudio	Visualization	Documentation	Overall
1. Ease of use	0.82	0.81	0.80	0.79	0.81
2. Clarity of guidance	0.81	0.83	0.82	0.80	0.82
3. Engagement	0.80	0.81	0.82	0.79	0.80
4. Understanding R	0.83	0.82	0.84	0.81	0.83

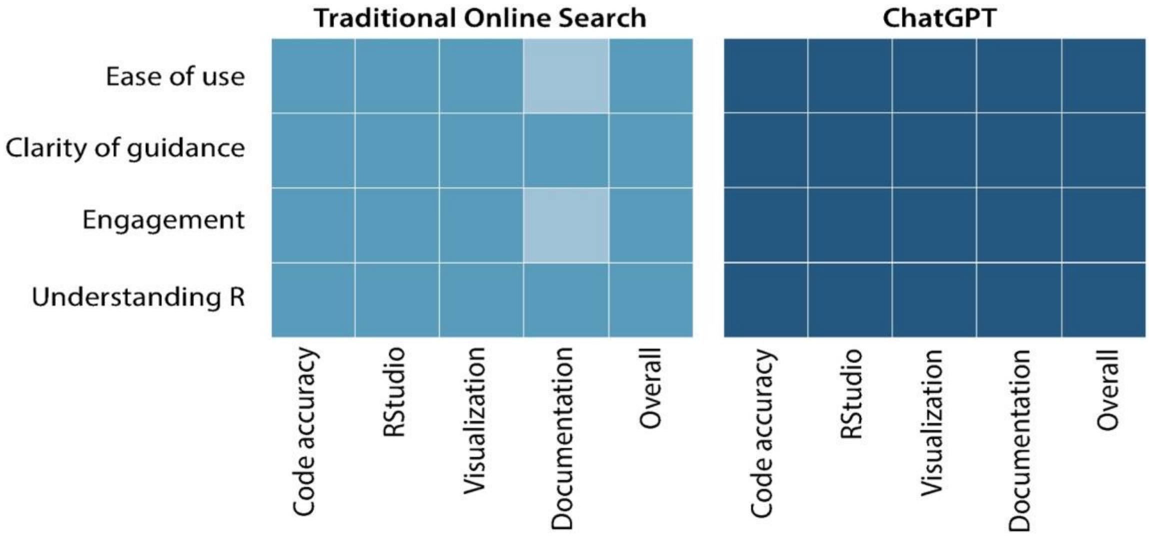
Interpretation Across Groups

Comparing the two matrices reveals that:

- Positive associations exist in both groups
- Correlations are substantially stronger and more homogeneous for ChatGPT
- Higher clarity and ease of use translate more directly into improved programming outcomes with AI-based support

This finding underscores ChatGPT’s potential not only to enhance subjective user satisfaction but also to directly support objective skill development—at least in short-term, assignment-based tasks. Figure 4 vividly illustrates the considerable overall correlation strengths between the two groups by way of a heat map.

Figure 4
Correlation Coefficient Heat Map by Tool Group



NEGATIVE				POSITIVE		
Strong	Moderate	Slight	Negligible	Slight	Moderate	Strong
[-1.0,0.7)	[0.7,-0.4)	[-0.4,0.1)	[-0.1,0.1]	(0.1,0.4]	(0.4,0.7]	(0.7,1.0]

Qualitative Analysis of Open-Ended Feedback

To deepen the analysis, students’ written reflections were coded thematically. Distinct patterns emerged between the two groups.

Themes in ChatGPT Feedback

ChatGPT users frequently highlighted:

- Immediate assistance (e.g., rapid explanations, instant debugging)
- Clarity and step-by-step reasoning
- Reduced cognitive load when troubleshooting errors
- Higher confidence in their final submissions

Representative comment: “When I made an error, I pasted the code into ChatGPT and it showed me exactly how to fix it. I usually get stuck and wait for help, but ChatGPT solved issues instantly.”

Themes in TOS Feedback

TOS users expressed challenges such as:

- Information overload
- Difficulty filtering credible sources
- Conflicting explanations
- Slow and frustrating error resolution
- Reliance on trial-and-error copying from websites

Representative comment: "Search results were overwhelming and often contradicted each other. I could not find a resource that helped me fix my graph the way I wanted."

Suggested Improvements

Students in both groups requested:

- More visualization examples
- Clearer instructions or more guided scaffolding
- Better integration between class instruction and outside tools

Figure 7 illustrates the frequency of improvement-related themes.

These qualitative patterns reinforce the quantitative findings, demonstrating that ChatGPT reduces frustration and increases comprehension, while TOS approaches create substantial cognitive overhead for novice learners.

Summary of Findings

Across all quantitative and qualitative measures, the ChatGPT group exhibited superior outcomes:

- Higher assignment scores across every rubric category
- Significantly better survey ratings
- Stronger correlations between perceived support and performance
- More positive reflections and fewer reported challenges

These results point to ChatGPT's capacity to enhance short-term learning and performance in R programming tasks, provided students engage with the tool critically and thoughtfully.

In Table 7 we provide keyword search tabulation by frequency among the two groups, noting significantly more positive correlations among the ChatGPT users vs. the TOS ones, and vice versa for the keywords we affiliate with negative feedback. Table 8 shows similar results when we tabulate the keywords and frequency by theme. Figures 5–7 provide a visualization of the keyword and theme results.

Table 7
Theme Frequency Keywords by Research Group

Metric	Frequency TOS	Frequency ChatGPT	Δ
Positive attributes			
Error resolution	2	8	+6
Coding assistance	3	7	+4
Efficiency	4	6	+2
Engagement	3	8	+5
Detailed instructions	2	9	+7
Specific questions	2	7	+5
Complexity of questions	3	6	+3
Code accuracy	4	5	+1
Understanding responses	2	4	+2
Limitations of ChatGPT	0	3	+3
Minor errors	1	2	+1
Simplifying language	0	5	+5
Enhanced interaction	0	4	+4
Visual and format improvements	0	3	+3
Real-life application	2	2	0
Negative attributes			
Overwhelming information	7	0	-7
Lack of guidance	8	0	-8
Time-consuming	7	0	-7
Variability in quality	6	0	-6

Note. Theme frequency keywords use student feedback from the two research groups; Δ denotes the difference in keyword frequency between the ChatGPT and TOS group participants.

Table 8
Keyword Frequency by Category and Theme

Category / Theme	TOS	ChatGPT	Δ
1. Most helpful aspects			
a. Immediate problem resolution	3	8	+5
b. Quality of explanations	4	7	+3
c. Code assistance	5	8	+3
d. Interactive learning	2	6	+4
e. Resource accessibility	5	5	0
2. Challenges encountered			
a. Time to find solutions	7	3	-4
b. Understanding responses	6	4	-2
c. Information overload	7	2	-5
d. Query formulation	6	5	-1
e. Technical issues	5	3	-2
3. Suggested improvements			
a. Enhanced visualization	4	6	+2
b. Better examples	6	5	-2
c. Clearer instructions	7	4	-3
d. More context aware	3	5	+2
e. Resource organization	7	2	-5

Note. Keywords are tabulated by frequency of mention between the two student groups and sorted by general category and themes within each category.

Figure 5
Most Helpful Aspects by Feedback Frequency

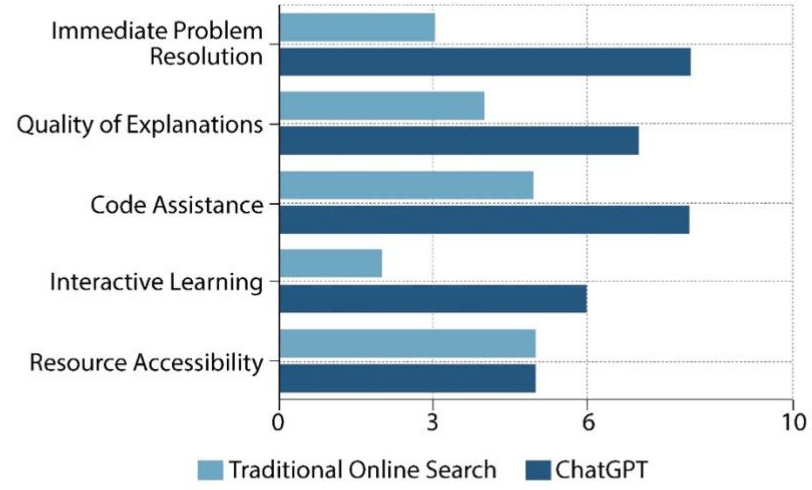


Figure 6
Challenges Encountered by Feedback Frequency

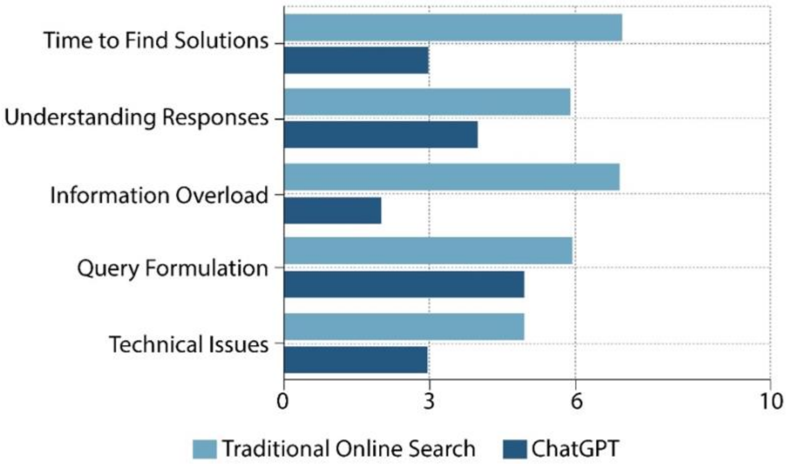
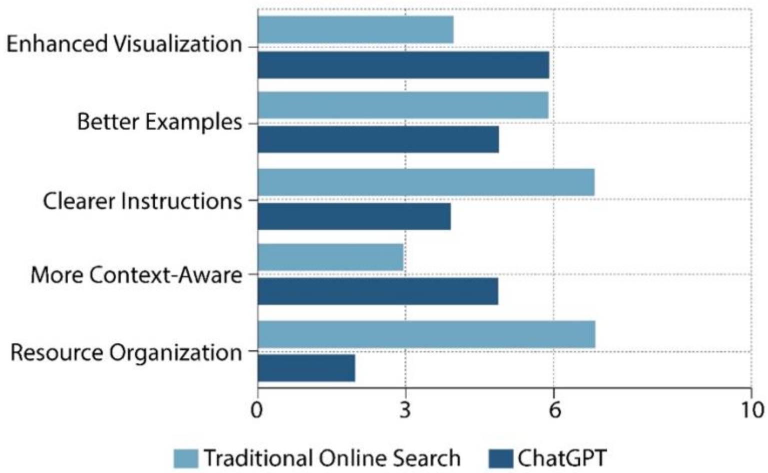


Figure 7
Suggested Improvements by Feedback Frequency



Discussion

The findings of this study provide clear and compelling evidence that ChatGPT offers significant short-term advantages over Traditional Online Search (TOS) methods for novice learners completing R programming and visualization tasks. Across all performance metrics and survey measures, students using ChatGPT demonstrated markedly higher proficiency and more positive learning experiences. These results contribute to ongoing discussions regarding the role of AI-driven tools in higher education by clarifying not only their potential benefits but also the pedagogical risks and design considerations required for responsible integration.

Interpretations of Quantitative Findings

The substantial performance gap between ChatGPT and TOS users is noteworthy. ChatGPT's superiority across every rubric category—from code accuracy to visualization quality and documentation—suggests that LLM-assisted learning meaningfully accelerates early-stage programming success. This aligns with prior research indicating that AI tutoring systems can streamline error resolution, reduce cognitive load, and support learners at precisely the points where they struggle most (Chen et al., 2020; Dempere et al., 2023).

Survey results further reinforce this interpretation. ChatGPT users consistently reported greater ease of use, clearer guidance, higher engagement, and stronger perceived understanding. These perceptions were not merely subjective; they correlated strongly with objective assignment scores. The homogeneity and strength of correlations within the ChatGPT group indicate that learners' experiences with the AI tool translated directly into improved output, suggesting a highly aligned relationship between instructional support and performance outcomes.

In contrast, TOS users faced predictable barriers such as information overload, inconsistent resource quality, and difficulty identifying reliable or context-appropriate solutions. These challenges have been widely documented in the literature on digital search habits (Purcell et al., 2012; Yim & Su, 2024), especially among novice programmers lacking the domain expertise needed to filter or evaluate online content efficiently.

Pedagogical Advantages of ChatGPT

Several features of ChatGPT likely contributed to its effectiveness in this study:

1. **Context-Sensitive Debugging**
   Students using ChatGPT could paste problematic code directly into the interface and receive specific, actionable corrections. This bypasses the trial-and-error process common in forum-based troubleshooting.
2. **Reduction of Extraneous Cognitive Load**
   By providing unified, coherent explanations, ChatGPT removes the need to reconcile fragmented or conflicting sources, aligning with Cognitive Load Theory's emphasis on reducing extraneous demands (Sweller, 1988).
3. **Enhanced Engagement**
   The conversational interface encouraged iterative refinement of code and deeper questioning, resulting in stronger task persistence.
4. **Scaffolding of Conceptual Understanding**
   ChatGPT generated step-by-step reasoning, analogies, and tailored examples that helped novices grasp foundational R concepts.

Taken together, these features suggest that LLM-based tools may function as highly accessible, on-demand tutors for early-stage programming learners.

Risks and Limitations of AI-Assisted Learning

Despite its clear advantages, the use of ChatGPT also presents risks requiring careful pedagogical mitigation.

Overreliance on AI-Generated Solutions

Qualitative data revealed that several students worried about relying too heavily on ChatGPT, particularly when the tool produced complete code blocks. While these solutions accelerated task completion, students questioned whether they fully understood the underlying logic. This aligns with existing literature cautioning that AI tools may inadvertently promote superficial learning if students fail to verify or contextualize model outputs (Adel & Ahsan, 2024; Bender et al., 2021).

Potential for Inaccurate or Overconfident Outputs

Although less commonly reported in this study, LLM responses can at times be incorrect or misleading—especially in technical contexts where subtle errors may go unnoticed. Beginners may lack the expertise to evaluate correctness, reinforcing the need for instructor guidance and explicit verification strategies.

Shifts in Student Learning Strategies

TOS users tended to synthesize information across multiple sources, building transferable search and evaluation skills. ChatGPT users, by contrast, often relied on immediate answers rather than deeper exploration. While this increased efficiency, it may limit opportunities to develop metacognitive strategies traditionally cultivated through research-based problem-solving.

Implications for Instructional Design

The results suggest several instructional strategies that can optimize AI integration while maintaining academic integrity and conceptual mastery:

1. **Structured Verification Tasks**
   Assignments can require students to annotate AI-generated code, explaining each line and justifying modifications to ensure deeper processing.
2. **Balanced Tool Ecosystems**
   Educators may combine ChatGPT with curated TOS resources, encouraging students to cross-reference explanations and compare reasoning patterns.
3. **AI Usage Guidelines**
   Clear expectations regarding permissible use of AI tools—especially in graded assignments—can help maintain fairness and promote thoughtful engagement.
4. **Instructor-Led Demonstrations**
   Modeling how to critique AI outputs can help students develop essential evaluative skills, reducing passive adoption of generated solutions.

5. **Designing Assessments That Emphasize Reasoning**
   Tasks requiring interpretation, annotation, or modification of code may reduce the risk of overreliance on generated solutions.

These strategies align with broader recommendations for integrating emerging technologies into STEM curricula while maintaining rigorous learning standards (Yim & Su, 2024).

Integration With Broader Literature

The superiority of ChatGPT in this study mirrors trends observed across AI-in-education research indicating that LLMs can accelerate novice learning and enhance student engagement. However, this study extends the existing literature by providing a direct classroom-based comparison with TOS tools, rather than relying on simulated or anecdotal cases. It also focuses specifically on R programming—an area underrepresented in empirical AI studies despite its importance for data literacy.

Moreover, the findings provide nuance to existing work by showing that the instructional benefits of AI do not eliminate the need for careful scaffolding. While ChatGPT excels at providing clarity and reducing cognitive barriers, it must be paired with pedagogical structures that promote reflection, verification, and the development of independent programming skills.

Overall Interpretation

Taken together, the results indicate that ChatGPT is a highly effective support tool for novice R programmers, improving both perceived and actual performance. However, the tool's strength—providing efficient, high-quality solutions—also constitutes a primary pedagogical risk if it leads to overreliance or insufficient conceptual engagement. The challenge for educators is thus not whether to incorporate AI tools, but how to design instructional environments where such tools enhance rather than undermine learning.

Limitations

Several limitations constrain the generalizability and scope of the present study. First, the sample size was small (n = 20) and drawn from a single minority-serving institution. While this context provides valuable insights into the experiences of a diverse undergraduate population, the limited sample restricts statistical power and may not represent broader learner demographics or institutional types.

Second, the study measured only short-term learning outcomes—specifically, performance on a single R programming assignment conducted over two class sessions. Longer-term impacts such as knowledge retention, transfer of learning, or sustained changes in study behavior were not assessed. Because AI tools may reduce cognitive load, they could either accelerate early learning or inhibit deeper conceptual engagement; this study cannot adjudicate between these possibilities over extended timeframes.

Third, ChatGPT users may have benefited from the novelty effect associated with emerging technologies. Enthusiasm or curiosity about AI-based tools may temporarily elevate engagement and perceived ease of use, inflating survey scores relative to TOS methods.

Fourth, the reliability of ChatGPT responses was not independently verified for every student interaction. Although no major inaccuracies were reported in this study, large language models are known to occasionally produce incorrect or overly confident outputs. Novice learners, who often lack the expertise to identify subtle errors, may be particularly susceptible to adopting incorrect solutions without sufficient verification.

Finally, qualitative data were coded by a single researcher, which limits intercoder reliability. While the thematic patterns aligned well with quantitative findings, future studies should incorporate multiple coders to strengthen validity.

Taken together, these limitations highlight the need for additional large-scale, multi-institutional, and longitudinal research to more fully evaluate the educational role of ChatGPT and related AI tools.

Future Research

The findings from this exploratory study highlight several opportunities for continued investigation into the role of AI-driven tools in programming education. First, future research should employ larger and more diverse samples across multiple institutions to enhance the generalizability of results and to examine whether student demographics, academic preparedness, or technological familiarity shape the effectiveness of ChatGPT relative to TOS methods.

Second, longitudinal studies are needed to assess the persistence of learning gains associated with AI-assisted instruction. While ChatGPT significantly improved short-term performance, it remains unclear whether such support promotes long-term conceptual understanding, skill retention, or independent problem-solving. Tracking students throughout a semester—or across multiple courses—would provide insight into whether early reliance on AI strengthens or diminishes deeper learning.

Third, future studies should compare multiple AI tools, including specialized coding assistants or domain-specific models, to determine whether certain systems are particularly effective for R programming or data visualization tasks. Comparative analyses could help clarify which features (e.g., debugging precision, example generation, scaffolding) most strongly influence novice success.

Fourth, researchers should explore how instructional design shapes AI-mediated learning. Experiments incorporating structured reflection prompts, code-explanation tasks, or verification checkpoints could illuminate how pedagogical scaffolding might mitigate risks of overreliance. Conversely, studies examining open-ended or self-directed tasks may reveal how students naturally adapt AI tools to their learning strategies.

Finally, future work should examine ethical and academic integrity considerations, including students' perceptions of acceptable AI use, the impact of AI on assessment design, and the development of guidelines that balance innovation with fairness. As AI tools become increasingly embedded in educational ecosystems, such research is essential for establishing responsible and equitable practices.

Conclusion

This study provides empirical evidence that ChatGPT offers substantial advantages over Traditional Online Search methods for novice learners completing R programming and visualization tasks. Students who relied on ChatGPT not only achieved higher performance across every rubric category—code accuracy, visualization quality, documentation, and overall presentation—but also reported significantly greater ease of use, clarity, engagement, and perceived understanding. These findings reinforce emerging research indicating that large language models can serve as powerful scaffolding tools, helping students navigate early conceptual and syntactic hurdles while reducing cognitive load.

At the same time, the results highlight important pedagogical considerations. While ChatGPT efficiently guides learners toward correct solutions, its effectiveness may inadvertently encourage overreliance or passive code adoption if not balanced with opportunities for verification and deeper reflection. Traditional Online Search methods, though less efficient for beginners, promote resource evaluation and synthesis skills that remain valuable for long-term development as independent programmers.

The challenge for educators is therefore one of intentional integration. AI tools such as ChatGPT should not replace foundational instruction or critical inquiry but rather complement them, functioning as accessible, on-demand tutors that support rather than supplant learning. With appropriate scaffolding—such as requiring students to annotate AI-generated code, compare outputs with traditional resources, or justify their reasoning—AI-driven assistance can significantly enhance both the efficiency and quality of early programming education.

As the educational landscape continues to evolve in response to rapid advances in artificial intelligence, this study contributes timely insights into how AI tools can be used responsibly and effectively. By demonstrating the short-term benefits and outlining the structural supports needed to mitigate potential risks, the findings offer a foundation for educators, researchers, and institutions seeking to thoughtfully integrate AI within analytics and computing curricula.

Author’s Note

The data that support the findings of this study are not openly available due to reasons of sensitivity and are available from the corresponding author upon reasonable request. The data is located in controlled access data storage at the School of Business and Computing, Modesto Junior College, 435 College Avenue, Modesto, CA 95350, USA and University of the Pacific, Eberhardt School of Business, 3601 Pacific Avenue, Stockton CA 95211.

The author did not receive any separate grant from any organization for the submitted work, which was performed as a normal part of his employment. The author has no relevant financial or non-financial interests to disclose, nor any competing interests to declare that are relevant to the content of this article.

References

Adel, A., & Ahsan, A. (2024). ChatGPT promises and challenges in education: Computational and ethical perspectives. Education Sciences, 14(8), 814. https://doi.org/10.3390/educsci14080814

Almasri, F. (2024). Exploring the impact of artificial intelligence in teaching and learning of science: A systematic review of empirical research. Research in Science Education, 54, 977–997. https://doi.org/10.1007/s11165-024-10176-3

Anderson, C., Baskerville, R., & Kaul, M. (2017). Impact of Artificial Intelligence, Robotics, and Automation on Higher Education. AMCIS 2017 Proceedings.

Baker, T., & Smith, L. (2019). Educ-AI-tion Rebooted? Exploring the future of artificial intelligence in schools and colleges. https://www.semanticscholar.org/paper/Educ-AI-tion-Rebooted-Exploring-the-future-of-in-Baker-Smith/d01f96c020e0b1f72df9a55c93fd7672001a8e91

Beale, R., & Steel, R. (2024). Application and Theory Gaps: The Rise of AI. Journal of Educational Technology, 45(3), 189-201.

Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency 2021, 610–623. https://doi.org/10.1145/3442188.3445922

Binns, R. (2018). Fairness in Machine Learning: Lessons from Political Philosophy. Proceedings of Machine Learning Research 81:*1-11.* Conference on Fairness, Accountability, and Transparency. https://proceedings.mlr.press/v81/binns18a/binns18a.pdf

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., … Liang, P. (2021). On the opportunities and risks of foundation models. Stanford University, Center for Research on Foundation Models (CRFM). https://arxiv.org/abs/2108.07258

Brown, M., & Clark, R. (2023). Fairness in Machine Learning. AI Ethics Journal, 12(2), 98-114.

Chen, L., Chen, P., & Lin, Z. (2020). Artificial Intelligence in Education: A Review. IEEE Access, vol. 8, pp. 75264-75278. https://doi.org/10.1109/ACCESS.2020.2988510

Dempere, J., Modugu, K., Hesham, A., & Ramasamy, L. K. (2023). The impact of ChatGPT on higher education. Frontiers in Education, 8. https://doi.org/10.3389/feduc.2023.1206936

Forrester, C., Schwikert, S., Foster, J., & Corwin, L. (2022). Undergraduate R programming anxiety in ecology: Persistent gender gaps and coping strategies. CBE – Life Sciences Education, 21(2), ar29. https://doi.org/10.1187/cbe.21-05-0133

González-Zamar, M.-D., & Abad-Segura, E. (2022). Global evidence on flipped learning in higher education. Education Sciences, 12(8), 515. https://doi.org/10.3390/educsci12080515

Greenhow, C., Robelia, B., & Hughes, J. E. (2009). Learning, Teaching, and Scholarship in a Digital Age: Web 2.0 and Classroom Research: What Path Should We Take Now? Educational Researcher, 38(4), 246–259. https://doi.org/10.3102/0013189X09336671

Head, A., & Eisenberg, M. (2009). Lessons Learned: How College Students Seek Information in the Digital Age. http://dx.doi.org/10.2139/ssrn.2281478

Laak, K.-J., & Aru, J. (2024). AI and personalized learning: Bridging the gap with modern educational goals. arXiv. https://doi.org/10.48550/arXiv.2404.02798

Luan, H. Geczy, P., Lai, H., Gobert, J., Yang, S. J. H., Ogata, H., Baltes, J., Guerra, R., Li, P., & Tsai, C. -C. (2020). Challenges and future directions of big data and artificial intelligence in education. Frontiers in Psychology, 11, 580820. https://doi.org/10.3389/fpsyg.2020.580820

Mayer, R. E., & Moreno, R. (2003). Nine ways to reduce cognitive load in multimedia learning. Educational Psychologist, 38(1), 43–52. https://doi.org/10.1207/S15326985EP3801_6

Purcell, K., Rainie, L., Buchanan, J., Friedrich, L., Jacklin, A., Chen, A., & Zickuhr, K. (2012). Pew Research Center. The mixed impact of digital technologies on student research. Pew Research Center. https://www.pewresearch.org/internet/2012/11/01/how-teens-do-research-in-the-digital-world/

Rudolph, J., Tan, Sa., & Tan, Sh. (2023). ChatGPT: Bullshit spewer or the end of traditional assessments in higher education? Journal of Applied Learning & Teaching, 6(1) 342–363. https://doi.org/10.37074/jalt.2023.6.1.9

Siau, K., & Yang, Y. (2017). Impact of AI on Higher Education. IEEE IT Professional, 19(4), 22–27.

Shahid, A., Hayat, K., Iqbal, Z., & Jabeen, I. (2023). Comparative Analysis: ChatGPT vs Traditional Teaching Methods. Pakistan Journal of Society, Education, and Language, vol. 9, pp. 585–593. https://www.researchgate.net/publication/374675434_Comparative_Analysis_ChatGPT_vs_Traditional_Teaching_Methods

Sweller, J. (1988). Cognitive Load During Problem Solving: Effects on Learning. Cognitive Science 12(2) 257–285. https://doi.org/10.1207/s15516709cog1202_4

Wang, X., Rush, C., & Horton, N. J. (2017). Data visualization on day one: Bringing big ideas into intro stats early and often. Technology Innovations in Statistics Education, 10(1). University of California eScholarship. https://escholarship.org/uc/item/84v3774z

World Economic Forum (2020). *The future of jobs report 2020*. https://www3.weforum.org/docs/WEF_Future_of_Jobs_2020.pdf

Yim, I. H. Y., & Su, J. (2024). Artificial intelligence (AI) learning tools in K-12 education: A scoping review. Journal of Computers in Education. https://doi.org/10.1007/s40692-023-00304-9

Zawacki-Richter, O., Marín, V. I., Bond, M., & Gouverneur, F. (2019). Systematic review of research on artificial intelligence applications in higher education – where are the educators? Int J Educ Technol High Educ 16, 39. https://doi.org/10.1186/s41239-019-0171-0

Contact email: alal@pacific.edu