rEFLections
Vol 33, No 3, September - December 2026

Self-Reported AI Use and Quiz Performance in an Introductory Programming Course: An Observational Study

NOUR HAURANE[1]
ERLEND TØSSEBRO[2*]
PATTAMAWAN JIMARKON[1]
[1]*Department of University Pedagogy, University of Stavanger, Norway*
[2]*Department of Electrical Engineering and Computer Science, University of Stavanger, Norway*
***Corresponding author email: erlend.tossebro@uis.no**

Article information

Article history:
Received: 18 Jun 2026
Last revised: 14 Oct 2026
Accepted: 24 Oct 2026
Available online: 29 Oct 2026

Keywords:
Large language models
Introductory programming
AI-assisted programming
Novice programmers
Quiz performance
Student self-assessment

Abstract

Large language models (LLMs) are now widely available to students, but it is still unclear how their use during coursework relates to later quiz performance. This study examines AI use in a bachelor-level Python course, where learners completed programming exercises and follow-up quizzes. To examine how students' self-reported AI use relates to perceived difficulty, confidence, perceived helpfulness, and quiz performance. We conducted a quantitative observational study with supplementary qualitative analysis using 170 anonymous exercise-level survey submissions collected after five programming exercises. Students reported their AI use, perceived difficulty, confidence, perceived helpfulness, and how they used AI. Three exercises included follow-up comprehension quizzes, providing 100 scored submissions. AI assistance during these quizzes was not ruled out. AI was used in 64.7% of submissions. In the scored subset, reported exercise AI use was associated with a 21.0-point lower mean than reported no-AI use on the observed-range 0–100 score, which is not percent correct. Exploratory subgroup differences varied by difficulty and confidence, with some small comparison cells. Higher AI-use intensity was associated with lower scores, while perceived helpfulness had a near-zero association with quiz performance. Open-ended responses suggested that higher-scoring AI-use submissions more often described debugging or checking their own code, while lower-scoring submissions more often described code generation. Student ratings of AI helpfulness should be interpreted cautiously, since they did not track quiz performance. For introductory programming courses, follow-up checks of independent understanding may be important when students use AI during exercises.

INTRODUCTION

Large language models (LLMs) have become widely accessible instructional tools, and their influence is especially visible in programming education. Students now have access to tools that can generate functional code, explain concepts, and debug errors on demand (Denny et al., 2023; Finnie-Ansley et al., 2022; Kasneci et al., 2023; Kazemitabaar et al., 2024).

While these capabilities promise to support learning, a central concern is that the same tools may reduce engagement with the reasoning, debugging, and retrieval processes that build durable skill (Bjork & Bjork, 2020; Koedinger & Aleven, 2007).

Despite rapid growth in the literature, gaps remain for introductory programming course settings. We still have limited evidence about how AI use varies across course tasks, how it relates to students' perceived difficulty and confidence, and whether students' perceptions of AI helpfulness correspond to quiz performance. Controlled studies of designed AI support do not resolve these questions about novices' self-selected use during ordinary coursework.

Two ideas from learning research help frame this tension. First, the assistance dilemma describes a tradeoff between providing help that makes a task easier and creating conditions where students build lasting competence (Koedinger & Aleven, 2007). When help is too direct, students can complete tasks while doing less of the thinking that supports later independent performance, which may prevent them from engaging in the productive struggle necessary for deeper learning. Too little help, however, can leave them frustrated and unable to progress. This becomes especially pressing with AI tools that can provide complete solutions on demand. Unlike a human tutor who might offer hints or ask guiding questions, an LLM can readily produce working code when prompted, potentially bypassing the learning process altogether. Second, the concept of desirable difficulties suggests that effortful practice can slow short-term performance while strengthening long-term learning and transfer (Bjork & Bjork, 2020). From this perspective, the friction students experience when working through a programming problem is part of how skill develops. AI tools that remove this friction may produce an illusion of competence: students complete tasks successfully but fail to develop the underlying understanding necessary to perform independently.

LLMs also raise risks familiar from research on cognitive offloading and automation misuse. When a system produces plausible output with little effort, users may defer cognitive work to external tools (Risko & Gilbert, 2016), a pattern confirmed in AI-education research where offloading correlates with reduced critical thinking (Gerlich, 2025; Lee et al., 2025; Stadler et al., 2024). In programming settings, this can take the form of solution-first use, where students adapt generated code before they understand it. Other uses, such as explanation, debugging, or checking, may place more of the reasoning work with the student. Beyond affecting task performance, AI use may also shape how students judge their own competence. Self-efficacy is known to influence persistence and engagement in programming (Ramalingam et al., 2004), and when tools make tasks feel more manageable, students may develop confidence that outpaces their actual understanding (Zhang & Xu, 2025).

Empirical results on AI support and learning outcomes are mixed and appear sensitive to how AI is designed and used. A meta-analysis of AI chatbots and a systematic review of structured AI tutors report learning gains (Létourneau et al., 2025; Wu & Yu, 2024). One RCT found that AI tutoring produced more than double the learning gains of in-class active learning (Kestin et al., 2025). Guardrails can mitigate harms in some contexts (Bastani et al., 2025; Kazemitabaar et al., 2024; Liffiton et al., 2023). Other programming studies report small or no differences in learning outcomes between AI-assisted and non-assisted groups (Sun et al., 2024). In

contrast, some studies find negative associations in programming learning outcomes (Yang et al., 2025). Meta-analytic work also suggests that average effects vary by context and design (Deng et al., 2025). Recent experimental evidence from a different population points toward skill acquisition as a direct concern. In a randomized trial with 52 professional software engineers learning a new Python library, Shen and Tamkin (2026) found that participants with AI assistance scored 17 percentage points lower on a comprehension quiz than those coding by hand, with the largest gap on debugging questions. Their qualitative analysis identified distinct interaction patterns associated with different outcomes: participants who fully delegated tasks to AI completed them fastest but showed poor comprehension, while those who asked conceptual questions or requested explanations preserved learning. Whether similar patterns appear among novice programmers in naturalistic classroom settings remains an open question. Extending this to classroom settings, Jošt et al. (2024) found a significant negative relationship between reliance on LLMs for code generation and debugging and students' final grades, especially for tasks requiring critical thinking.

Many students nevertheless experience AI as helpful. For teachers, this creates a measurement problem: students' sense of progress during practice may not correspond to their quiz performance. Improved engagement or perceived support without corresponding gains in knowledge measures have been reported (Wang et al., 2026), suggesting that student perceptions may be unreliable indicators of learning. CS education research also points to a widening gap: students with stronger monitoring and self-regulation may be better placed to use AI productively, while others may have more difficulty detecting errors or deciding when to rely on the tool (Darvishi et al., 2024; Prather et al., 2024).

We address these gaps with a quantitative observational study with supplementary qualitative analysis in an introductory programming course at the University of Stavanger (UiS). Across five exercises, we collected 170 voluntary, anonymous survey submissions. Each submission included the student's self-reported AI use for that exercise, perceived difficulty, confidence in understanding, and an open-ended description of how they used AI. Three exercises were followed by quizzes that provided performance scores, yielding 100 scored submissions.

The study asks three research questions: (1) How does AI use vary with perceived difficulty and confidence? (2) How is AI use associated with quiz performance? (3) Do these associations differ by difficulty level and confidence level? The goal is to describe patterns of AI use and performance in a live course and to generate hypotheses that can inform future controlled studies and instructional guidance.

METHODS

Course context

The study took place in DAT120, an introductory programming and data analysis course at UiS. The course uses Python, a language well represented in LLM training data, which likely made these exercises relatively accessible to AI tools at the time of data collection. During

the semester, students completed a series of 10 programming exercises. After five of these exercises (2, 4, 5, 7, and 10), students were invited to complete a brief, voluntary, anonymous survey about their experience. Three exercises (2, 4, and 7) included short quizzes that provided performance scores. AI use during quizzes was not recorded.

The five surveyed exercises covered different topics and structures. Exercises 2 (control structures and loops), 4 (reading text files and exception handling), and 7 (collections including lists, arrays, tuples, sets, and dictionaries) each contained several small programming tasks designed to teach specific constructs. Exercise 5 was a single larger task focused on problem solving and decomposition. Exercise 10 was the final part of a group project requiring students to integrate all course topics including basic classes and objects. The course was designed with exercises roughly increasing in difficulty. In informal instructor testing conducted around the time of data collection, most AI tools could reliably solve the tasks in Exercises 2 and 4, while producing more variable results on the remaining exercises.

Procedure and measures

For each exercise, the survey asked students to report their AI use for that exercise (four levels from "Not at all" to "On all the sub-questions"), how helpful AI was (from "Didn't use" to "Very good help"), perceived difficulty (1–5), confidence in understanding (1–5), and open-ended descriptions of how they used AI and what challenges they faced.

Dataset

The dataset contains 170 survey submissions across five exercises. Because surveys were anonymous, submissions were not linked across exercises, and each submission was treated as one observation. Per-exercise submission counts were: Ex. 2 (n = 49), Ex. 4 (n = 31), Ex. 5 (n = 40), Ex. 7 (n = 20), and Ex. 10 (n = 30). Quiz performance data were available for Ex. 2, Ex. 4, and Ex. 7 (n = 100 scored submissions). In this scored subset, 70 submissions (70%) reported some AI use.

Outcome measure: Quiz score

For scored exercises, quiz items used a penalty scoring rule (+3 for correct, -2 for incorrect, 0 for blank). Raw quiz totals were expressed as a percentage of the maximum possible score ("% from max"), which can be negative if incorrect answers outweighed correct ones. To make scores easier to interpret on a single scale, we rescaled "% from max" using min-max rescaling based on the observed range in the scored subset (minimum = −44.44, maximum = 100.00). The resulting rescaled score ranges from 0 to 100, where 0 corresponds to the lowest observed score and 100 to the highest. This rescaling preserves rank order and relative differences, but it should not be interpreted as percent-correct. On the original percentage-of-maximum scale, mean scores were 19.3 with reported AI use and 49.7 without it, a difference of 30.4 points.

Derived variables

We defined an exercise AI-use indicator as any usage level other than "Not at all." The labels "AI" and "no AI" refer to exercise use, not verified quiz-time assistance. For analyses of intensity, we treated the four AI-use categories as an ordered scale (0–3). For subgroup analyses, we grouped difficulty ratings into Easy (1–2), Moderate (3), and Difficult (4–5), and confidence ratings into Low (1–2) and High (3–5). To assess alignment between confidence and performance, we calculated a relative confidence–performance index as the difference between standardized quiz score and standardized confidence rating. This index describes relative positions on two standardized measures and is not a validated measure of absolute calibration.

Quantitative analysis

We summarized AI-use rates overall and by exercise. To examine how AI use relates to perceived difficulty and confidence, we computed Spearman correlations between AI-use intensity and each rating. To test performance differences between AI users and non-users, we used Mann–Whitney U tests with rank-biserial correlation as the effect size, along with 95% bootstrap confidence intervals (2,000 iterations). To account for differences between quizzes, we estimated linear models with exercise fixed effects and robust (HC3) standard errors. These models treat results as associations, not causal effects, because students chose whether to use AI. For an exploratory weighted description, we estimated inverse-probability weights from a logistic regression of AI use on perceived difficulty and confidence, with propensity scores clipped to [0.05, 0.95]. Mean differences and index means are also reported with 95% bootstrap confidence intervals.

Because difficulty and confidence were measured after the exercise, they may partly reflect students' experience of using AI and should not be treated as fully independent predictors. We therefore treat adjusted estimates as descriptive. Subgroup analyses by difficulty and confidence are exploratory. We report unadjusted p-values and interpret these results cautiously.

Qualitative analysis

Across all exercises, 88 submissions included a description of how AI was used and 26 included a response to the challenges prompt. To focus on substantive responses, we excluded entries shorter than 15 characters, yielding 82 usage descriptions and 18 challenge descriptions.

We first reviewed the open-ended responses to summarize recurring ways students described using AI and where they struggled. To make the exploratory comparisons reproducible, we then used keyword-assisted coding to map responses to a small set of common purposes: debugging/checking, explanation/understanding, code generation or solution-focused use, getting started, and syntax/formulation help. Responses could receive multiple codes. We iteratively refined keyword patterns by reading responses to reduce obvious mismatches. We then connected coded categories to quiz scores for the scored subset. These qualitative-quantitative links are exploratory. They are intended to generate hypotheses about productive and unproductive patterns of AI use, not to identify causal mechanisms.

RESULTS

AI use patterns

Across the 170 submissions, AI was used at least once in 64.7% of submissions. AI use varied across exercises, ranging from 50.0% in Ex. 5 to 80.6% in Ex. 4 (Figure 1). Within each exercise, the most common pattern was using AI on one sub-question or a good deal of the exercise, with fewer submissions reporting use on all sub-questions.

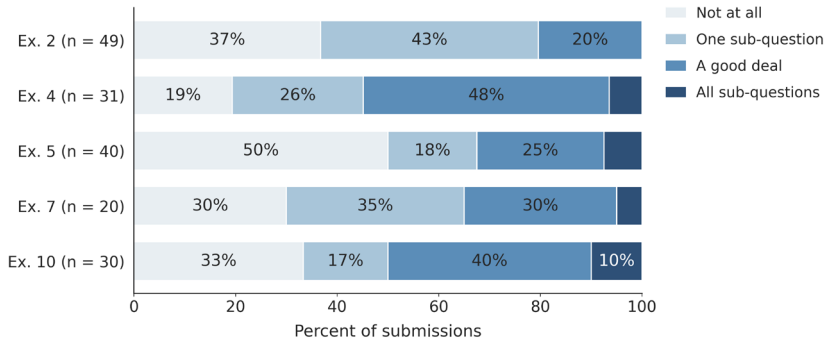


Figure 1 Self-reported AI use across five programming exercises. Sample sizes are shown beside each exercise.

Relationship between AI use, difficulty, and confidence

At the exercise level, exercises that students rated as more difficult tended to be the same exercises where average confidence was lower (r = -0.91, p = 0.032, n = 5 exercises). Across all 170 submissions, AI-use intensity was positively associated with perceived difficulty (ρ = .41) and negatively associated with confidence (ρ = −.36), as shown in Figure 2. The scored-subset correlations were .36 and −.29, respectively (n = 100). Submissions reporting greater difficulty or lower confidence also reported more AI use.

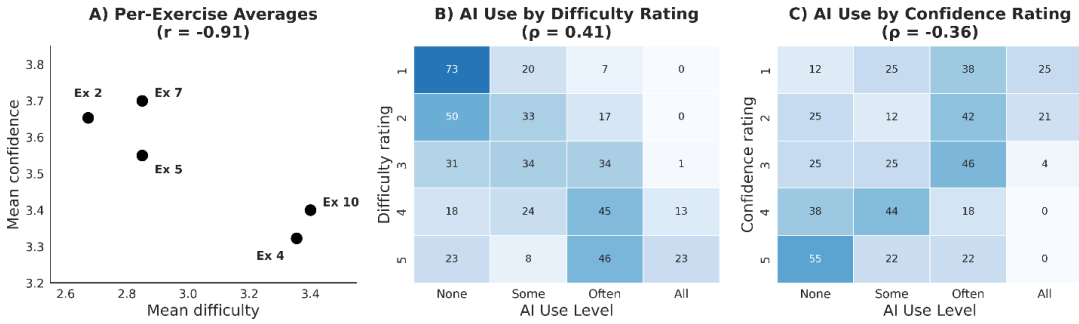


Figure 2 Relationships between AI usage, perceived difficulty, and confidence. (A) Exercise-level averages showing inverse relationship between difficulty and confidence. (B) AI use distribution by difficulty rating. (C) AI use distribution by confidence rating. Panels B and C use all 170 submissions; scored-subset correlations are reported separately in the text.

Performance differences on scored exercises

In the scored subset (n = 100), submissions reporting exercise AI use had lower quiz scores than submissions reporting no AI use (Figure 3). On the 0–100 rescaled score, AI-use submissions averaged 44.1 compared to 65.2 for no-AI submissions, a difference of 21.0 points. This difference was statistically significant (Mann-Whitney U = 592.5, p < .001) with a medium-to-large effect size (rank-biserial r = 0.44, 95% *CI* [0.20, 0.65]). Using a descriptive threshold of 50 on the 0–100 rescaled score (equivalent to 27.78 on the percentage-of-maximum scale, not a course pass threshold), 61% of AI-user submissions fell below 50 compared to 30% of non-AI submissions (risk ratio = 2.05, Fisher's exact p = .005). The gap was consistent across quiz components: AI-use submissions scored lower on both Q1 (1.16 vs 3.03, p = .002) and Q2 (1.00 vs 2.57, p = .004).

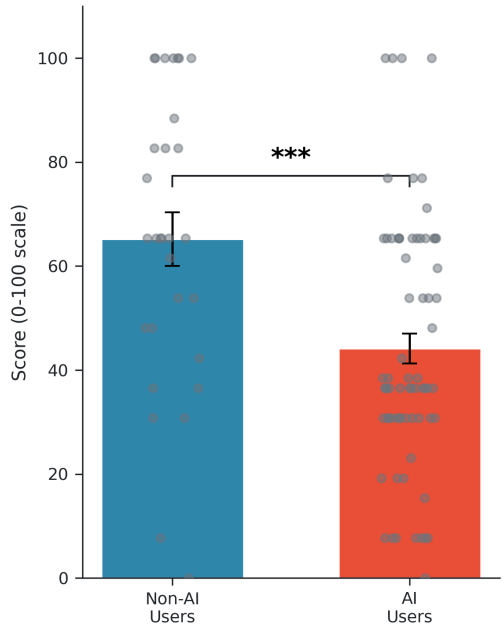


Figure 3 Quiz performance by AI use in the scored subset. Bars show means with standard errors. Points show individual submissions. Significance testing compares AI-use submissions (n = 70) to no-AI submissions (n = 30) using the Mann-Whitney U test. ***p < .001.

Performance by exercise

No-AI submissions had higher mean scores in each scored exercise (Figure 4). The no-AI minus AI differences were 17.9 observed-range points in Ex. 2 (95% bootstrap interval [5.1, 29.9]; 31 AI and 18 no-AI records), 24.2 in Ex. 4 ([−4.2, 51.1]; 25 and 6 records), and 23.6 in Ex. 7 ([−10.8, 51.1]; 14 and 6 records). The direction of the point estimates is descriptive; the broad intervals in Ex. 4 and Ex. 7 limit the precision of comparisons.

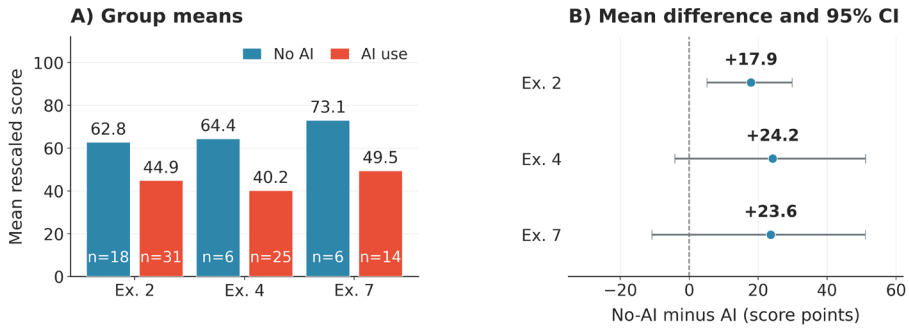


Figure 4 Quiz performance by exercise. (A) Mean rescaled scores, with group sizes inside the bars. (B) No-AI minus AI mean score differences with 95% bootstrap intervals.

AI use intensity and performance

When AI use was treated as an ordered scale, higher reported AI use aligned with lower quiz scores (ρ = −0.36, $p < .001$). A simple linear trend model estimated an average decrease of 10.8 points per increment in AI-use level (Figure 5). Mean scores were 65.2 for "Not at all" ($n = 30$), 47.0 for "One sub-question" ($n = 36$), 41.0 for "A good deal" ($n = 31$), and 42.3 for "All sub-questions" ($n = 3$).

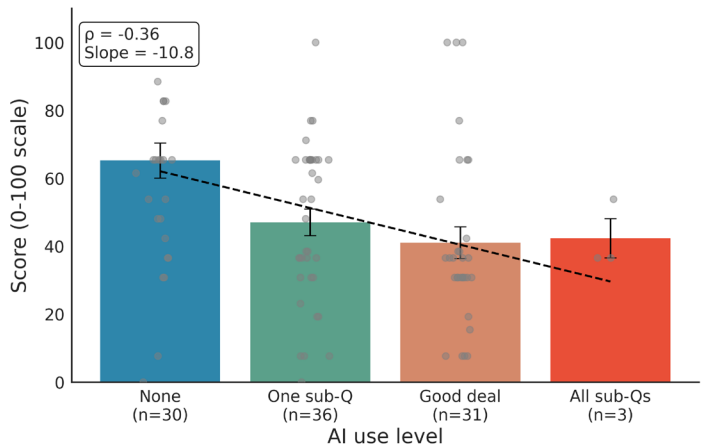


Figure 5 Quiz performance by self-reported AI usage level. Bars show means with standard error. Individual data points shown with jitter. Dashed line shows linear trend (slope = −10.8 points per level).

Performance by perceived difficulty and confidence

The observed score differences varied across difficulty groups. For easy ratings, the no-AI minus AI difference was 30.4 observed-range points (95% bootstrap interval [15.2, 45.3]; 18 AI and 13 no-AI records; rank-biserial $r = .68$). For moderate ratings, it was 17.7 ([0.0, 34.8]; 30 and 13 records; $r = .35$). For difficult ratings, it was −10.0 ([−28.1, 7.7]; 22 and 4 records; $r = -.25$; Figure 6). The reversed point estimate in the difficult group is uncertain, particularly because it contains only four no-AI records. These exploratory comparisons do not establish differences in an AI effect across difficulty groups.

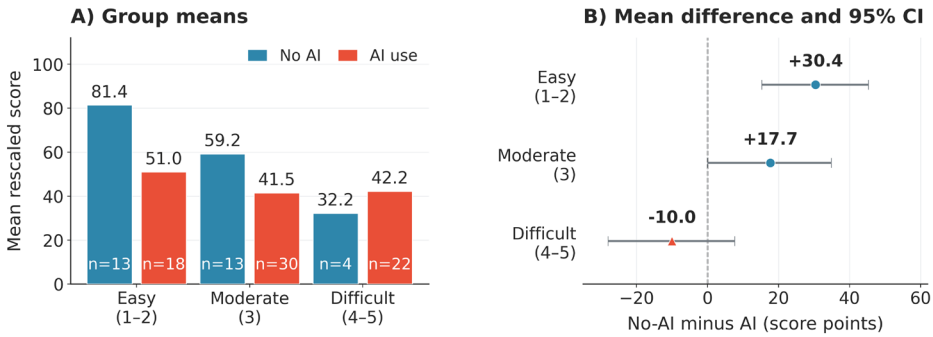


Figure 6 Quiz performance by perceived difficulty. (A) Mean rescaled scores, with group sizes inside the bars. (B) No-AI minus AI mean score differences with 95% bootstrap intervals. Difficulty was rated after the exercise.

Among high-confidence submissions (ratings 3–5), the no-AI minus AI difference was 20.7 observed-range points (95% bootstrap interval [8.1, 34.0]; 55 AI and 27 no-AI records; rank-biserial r = .42). Among low-confidence submissions (1–2), it was 10.9 ([−1.0, 22.7]; 15 and 3 records; r = .47; Figure 7). The three-record no-AI cell limits the low-confidence comparison. These estimates do not establish a reliable difference between confidence groups.

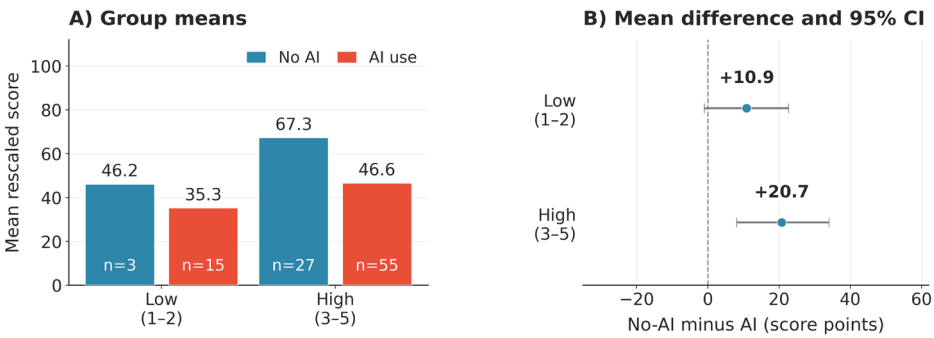


Figure 7 Quiz performance by confidence. (A) Mean rescaled scores, with group sizes inside the bars. (B) No-AI minus AI mean score differences with 95% bootstrap intervals. Low confidence includes ratings 1–2; high confidence includes 3–5.

Relative confidence and performance

To examine how confidence aligned with performance, we compared standardized quiz scores with standardized confidence ratings (Figure 8). The mean relative confidence–performance index was .23 for no-AI submissions and −.10 for AI-use submissions. The difference was .33 standardized units (p = .12, rank-biserial r = .20), without clear evidence of a group difference. These values describe relative positions on the two measures; they do not establish absolute underconfidence or overconfidence.

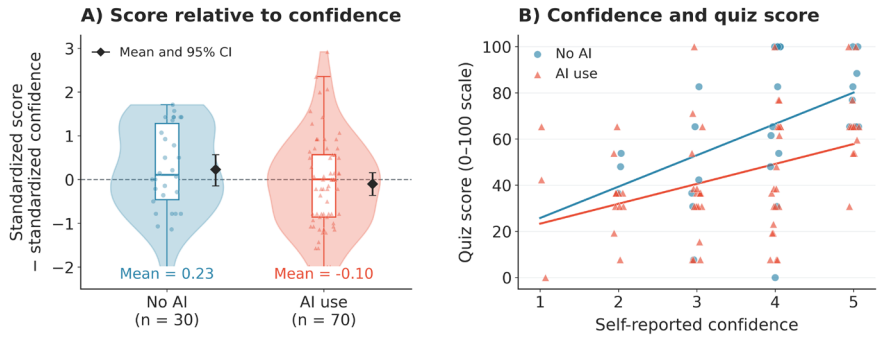


Figure 8 Relative confidence and performance. (A) Distributions of standardized score minus standardized confidence; diamonds show means with 95% bootstrap intervals. (B) Confidence ratings and rescaled quiz scores by reported AI use, with descriptive fitted lines.

Perceived helpfulness versus performance

Among AI users in the scored subset with helpfulness ratings other than "Didn't use" (n = 69), perceived helpfulness was not related to quiz performance (ρ = 0.02, p = .86). Mean scores remained below the non-AI user mean of 65.2 across all helpfulness categories (Figure 9). This pattern suggests that students' perceived helpfulness did not track quiz performance in this setting. Across AI-using submissions with valid helpfulness ratings (n = 107), perceived helpfulness was strongly correlated with usage intensity (ρ = 0.49, p < .001), and students who used AI more rated it as more helpful. Helpfulness ratings may reflect perceived contribution to task completion, but that construct was not measured directly.

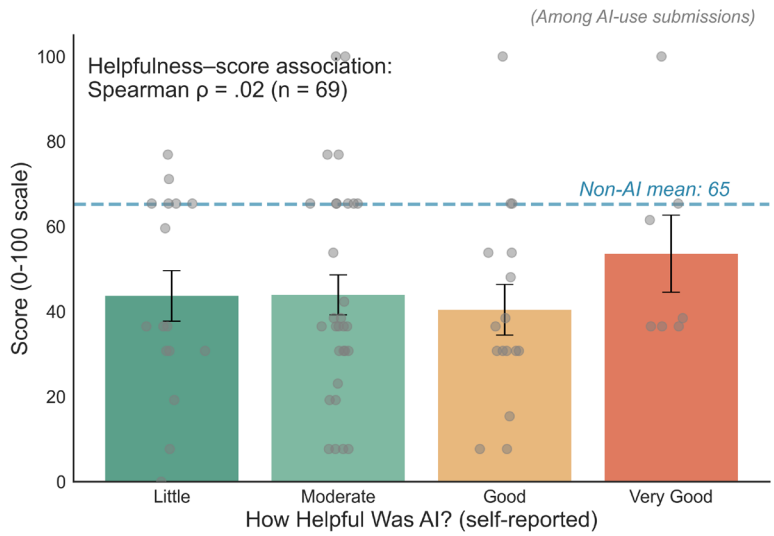


Figure 9 Performance by self-reported AI helpfulness (AI users only). Bars show means with standard errors. Dashed line indicates the non-AI user mean (65.2). The annotation describes the association with recorded quiz scores, not an estimate of actual learning benefit.

Adjusted and exploratory analyses

In a regression model that included perceived difficulty, confidence, and exercise fixed effects, AI use remained associated with lower quiz scores (b = −15.5, p = .010). Confidence was positively associated with performance (b = 8.4, p = .004), while perceived difficulty showed no significant association after adjustment (b = −2.6, p = .38). The model explained 28.9% of variance.

Exploratory inverse-probability weighting produced a smaller negative estimate (b = −12.7), and the uncertainty increased (p = .075). Because difficulty and confidence are self-reports collected after completing the exercise, they may partly reflect consequences of AI use rather than independent factors. We therefore treat these adjusted estimates as descriptive rather than causal.

As an exploratory check of the inverse difficulty pattern, we fit a model with an interaction between AI use and the 1-5 difficulty rating. The interaction term suggested that the negative association between AI use and performance was weaker on exercises that students rated as more difficult (b = 11.3, p = .03).

Thematic and structured content analysis of open-ended responses

Across exercises, 82 usage descriptions met the minimum length criterion, 79 of which came from AI-using submissions. The most common purposes were explanation or understanding (36.7%) and debugging or checking one's own code (35.4%). Code generation or solution-focused language appeared less often (17.7%), as did getting started (13.9%) and syntax help (7.6%). Beyond these categories, open-ended responses revealed differences in how students approached AI. Strategic, learning-focused use was rare (14%) and declined over the semester (from 17% early to 0% late). Critical evaluation of AI outputs appeared in only 5% of responses. Most students described reactive use. They turned to AI when they got stuck, with little evidence of a planned strategy.

In the scored subset, 53 AI-user submissions included a substantive usage description. Submissions mentioning debugging or checking scored higher on average (mean = 60.2, n = 19) than those that did not (mean = 39.7, n = 34). In the top and bottom quartiles of these 53 submissions, debugging language appeared in 63% of top-quartile descriptions and 11% of bottom-quartile descriptions. The open-ended responses showed that top-quartile performers more often described using AI to refine existing code (42% versus 17%). Bottom-quartile performers more often described using AI for code generation (17% versus 5%) and for getting started (22% versus 11%). These patterns raise a question about sequence, but they do not establish that consulting AI after an initial attempt leads to higher scores.

For the challenges question, 14 substantive responses came from AI-using submissions. Most said they struggled to phrase prompts that would elicit explanations; their prompts often produced full solutions instead (71.4%). One student wrote: "Difficult to write questions in a way that doesn't make it solve the problem, but explains what I'm trying to understand."

Over half described receiving incorrect or unhelpful outputs that required further prompting (57.1%). Some also reported that AI produced code using patterns not yet covered in the course.

We treat links between themes and performance as descriptive. The coded comparisons are best read as hypotheses about how students approach AI, not as evidence that any specific strategy causes higher or lower scores.

DISCUSSION

Across the surveyed exercises, AI use was common and patterned. Students reported more AI use when tasks felt difficult and when confidence was lower. In the scored subset, AI use was associated with lower quiz performance, especially on exercises students rated as easy and among high-confidence students. Helpfulness ratings, however, gave little indication of quiz performance.

These results should not be read as evidence that AI use caused lower scores. Students chose whether to use AI, and that choice likely reflects unmeasured factors such as prior programming experience, motivation, time constraints, or help-seeking preferences. However, the pattern is consistent with prior work in which unrestricted AI access improved practice performance but harmed later independent performance when AI was removed (Bastani et al., 2025). The findings also leave open the possibility that AI can support learning under different conditions. The present study therefore describes an association between self-reported AI use and quiz performance in a naturalistic course setting.

The AI-performance gap was largest on exercises students rated as easy and weaker on harder exercises. One possible interpretation, drawing on desirable difficulties theory (Bjork & Bjork, 2020), is that easier tasks leave more room for productive struggle to be bypassed. This also aligns with expertise reversal findings showing that the value of assistance depends on learner knowledge and task demands (Tetzlaff et al., 2025). The assistance dilemma offers a related possible explanation: on tasks that feel manageable, direct assistance may replace practice that would otherwise strengthen independent performance (Koedinger & Aleven, 2007). However, difficulty ratings were collected after the exercise, so AI use may also have influenced how difficult a task felt. The difficult-task group also included very few non-AI submissions, so this pattern should be treated as suggestive.

In our data, helpfulness ratings were unrelated to quiz performance, even though they rose with AI-use intensity across exercises, suggesting that helpfulness may reflect perceived contribution to finishing the exercise. Task contribution was not measured directly. Helpfulness ratings therefore provide weak evidence about whether AI improved learning. This pattern fits concerns that AI can increase confidence or perceived progress without a corresponding gain in competence (Bastani et al., 2025; Ren et al., 2026; Zhang & Xu, 2025). The relative confidence–performance comparison did not provide clear evidence of a group difference.

The open-ended responses suggest a hypothesis about the sequence of AI use: some students appeared to consult AI before doing enough of the problem-solving work themselves. Lower-performing students more often described using AI to get started or generate initial code, whereas higher-performing students more often described debugging or checking work they had already attempted. This distinction resembles the interaction patterns described by Shen and Tamkin (2026), although their study involved a different population and design. The pattern is therefore best read as hypothesis-generating: AI used after an initial attempt may support repair and reflection, while AI used before problem decomposition may reduce opportunities to practice the skills being assessed.

For teaching, the practical implication is to design AI use around sequence and verification. Students may benefit from guidance that asks them to attempt problems first, use AI for explanation or checking rather than initial solution generation, and verify outputs against their own understanding. Low-stakes checks of independent understanding may also be more informative than helpfulness ratings. Guardrails and constraint-based support may help preserve assistance while protecting opportunities to practice core skills (Liffiton et al., 2023; Kazemitabaar et al., 2024).

This study is limited to one course at one institution. Survey participation was voluntary and anonymous, so we could not estimate a response rate relative to course enrollment. AI use and perceptions are self-reported, whereas quiz scores are recorded outcomes. AI assistance during quizzes was not ruled out and was not identified by the exercise-use reports. These scores therefore cannot establish unaided competence or retained learning. Some subgroup analyses were based on small samples, and repeated participation could not be accounted for. The observational design means we cannot determine whether AI use caused lower scores or whether lower-performing students were more likely to seek AI help. Future work should combine measurement of prior skill with designs that vary AI access by task type or difficulty. Process measures such as time-on-task, code snapshots, and interaction logs would help test whether timing, effort, and verification explain the patterns observed here.

CONCLUSION

In this introductory programming course, self-reported AI assistance was associated with lower quiz performance, especially on exercises students experienced as easy. Helpfulness ratings gave little indication of quiz performance, suggesting that students' sense of support during practice should not be treated as evidence of learning. Because the study was observational, these findings should be read as associations. Even so, they point to a practical concern for programming instruction: AI use may need to be structured around first attempts, explanation, checking, and verification if it is to support rather than replace the work through which students learn. The study adds an exercise-level comparison of reported AI use, perceptions and quiz scores within the same course records. It highlights the distinction between perceived support and assessed quiz performance, while the quiz conditions prevent a claim about verified unaided understanding or retained learning.

THE AUTHORS

Nour Haurane is a senior educational technology advisor at Uniped (Department of University of Pedagogy), University of Stavanger, and Professor II at Noroff University, Norway. He designs and facilitates professional development programs for university teachers navigating technological change in assessment and pedagogy. His work focuses on critical AI integration in higher education, emphasizing evidence-based approaches that balance opportunity with limitations. He also specialises in assisting teachers maintaining assessment validity while acknowledging the complex realities students face, and is particularly interested in the role of trust and psychological safety in enabling pedagogical innovation under uncertainty.
nour.haurane@uis.no

Erlend Tøssebro is an associate professor in computer science at the Department of Electrical Engineering and Computer Science, University of Stavanger, Norway. His research interests include geospatial databases, data management, and teaching computer science, including the effects of AI on how to teach and what students need to learn.
erlend.tossebro@uis.no

Pattamawan Jimarkon is a professor of university pedagogy whose research focuses on intercultural pedagogies, language, identity, and communication in university settings. Her work centers on teacher training and professional development, with a particular emphasis on fostering reflective practice and reflexivity.
pattamawan.jimarkon@uis.no

REFERENCES

Bastani, H., Bastani, O., Sungu, A., Ge, H., Kabakcı, Ö., & Mariman, R. (2025). Generative AI without guardrails can harm learning: Evidence from high school mathematics. *Proceedings of the National Academy of Sciences, 122*(26), Article e2422633122. https://doi.org/10.1073/pnas.2422633122

Bjork, R. A., & Bjork, E. L. (2020). Desirable difficulties in theory and practice. *Journal of Applied Research in Memory and Cognition, 9*(4), 475–479. https://doi.org/10.1016/j.jarmac.2020.09.003

Darvishi, A., Khosravi, H., Sadiq, S., Gašević, D., & Siemens, G. (2024). Impact of AI assistance on student agency. *Computers & Education, 210*, Article 104967. https://doi.org/10.1016/j.compedu.2023.104967

Deng, R., Jiang, M., Yu, X., Lu, Y., & Liu, S. (2025). Does ChatGPT enhance student learning? A systematic review and meta-analysis of experimental studies. *Computers & Education, 227*, Article 105224. https://doi.org/10.1016/j.compedu.2024.105224

Denny, P., Kumar, V., & Giacaman, N. (2023). Conversing with Copilot: Exploring prompt engineering for solving CS1 problems using natural language. In *SIGCSE 2023: Proceedings of the 54th ACM Technical Symposium on Computer Science Education* (Vol. 1, pp. 1136–1142). Association for Computing Machinery. https://doi.org/10.1145/3545945.3569823

Finnie-Ansley, J., Denny, P., Becker, B. A., Luxton-Reilly, A., & Prather, J. (2022). The robots are coming: Exploring the implications of OpenAI Codex on introductory programming. In *ACE 2022: Proceedings of the 24th Australasian Computing Education Conference* (pp. 10–19). Association for Computing Machinery. https://doi.org/10.1145/3511861.3511863

Gerlich, M. (2025). AI tools in society: Impacts on cognitive offloading and the future of critical thinking. *Societies, 15*(1), Article 6. https://doi.org/10.3390/soc15010006

Jošt, G., Taneski, V., & Karakatič, S. (2024). The impact of large language models on programming education and student learning outcomes. *Applied Sciences, 14*(10), Article 4115. https://doi.org/10.3390/app14104115

Kasneci, E., Seßler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., Gasser, U., Groh, G., Günnemann, S., Hüllermeier, E., Krusche, S., Kutyniok, G., Michaeli, T., Nerdel, C., Pfeffer, J., Poquet, O., Sailer, M., Schmidt, A., Seidel, T., ... Kasneci, G. (2023). ChatGPT for good? On opportunities and challenges of large language models for education. *Learning and Individual Differences, 103*, Article 102274. https://doi.org/10.1016/j.lindif.2023.102274

Kazemitabaar, M., Ye, R., Wang, X., Henley, A. Z., Denny, P., Craig, M., & Grossman, T. (2024). CodeAid: Evaluating a classroom deployment of an LLM-based programming assistant that balances student and educator needs. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Article 650, pp. 1–20). Association for Computing Machinery. https://doi.org/10.1145/3613904.3642773

Kestin, G., Miller, K., Klales, A., Milbourne, T., & Ponti, G. (2025). AI tutoring outperforms in-class active learning: An RCT introducing a novel research-based design in an authentic educational setting. *Scientific Reports, 15*, Article 17458. https://doi.org/10.1038/s41598-025-97652-6

Koedinger, K. R., & Aleven, V. (2007). Exploring the assistance dilemma in experiments with cognitive tutors. *Educational Psychology Review, 19*(3), 239–264. https://doi.org/10.1007/s10648-007-9049-0

Lee, H.-P., Sarkar, A., Tankelevitch, L., Drosos, I., Rintel, S., Banks, R., & Wilson, N. (2025). The impact of generative AI on critical thinking: Self-reported reductions in cognitive effort and confidence effects from a survey of knowledge workers. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems* (Article 1121, pp. 1–22). Association for Computing Machinery. https://doi.org/10.1145/3706598.3713778

Létourneau, A., Deslandes Martineau, M., Charland, P., Karran, J. A., Boasen, J., & Léger, P. M. (2025). A systematic review of AI-driven intelligent tutoring systems (ITS) in K-12 education. *npj Science of Learning, 10*, Article 29. https://doi.org/10.1038/s41539-025-00320-7

Liffiton, M., Sheese, B., Savelka, J., & Denny, P. (2023). CodeHelp: Using large language models with guardrails for scalable support in programming classes. In *Proceedings of the 23rd Koli Calling International Conference on Computing Education Research* (pp. 1–11). Association for Computing Machinery. https://doi.org/10.1145/3631802.3631830

Prather, J., Reeves, B. N., Leinonen, J., MacNeil, S., Randrianasolo, A. S., Becker, B. A., Kimmel, B., Wright, J., & Briggs, B. (2024). The widening gap: The benefits and harms of generative AI for novice programmers. In *Proceedings of the 2024 ACM Conference on International Computing Education Research* (Vol. 1, pp. 469–486). Association for Computing Machinery. https://doi.org/10.1145/3632620.3671116

Ramalingam, V., LaBelle, D., & Wiedenbeck, S. (2004). Self-efficacy and mental models in learning to program. *ACM SIGCSE Bulletin, 36*(3), 171–175. https://doi.org/10.1145/1026487.1008042

Ren, L., Stephens, J. M., & Lee, K. (2026). The impact of AI on learners' self-efficacy: A meta-analysis. *Behavioral Sciences, 16*(1), Article 158. https://doi.org/10.3390/bs16010158

Risko, E. F., & Gilbert, S. J. (2016). Cognitive offloading. *Trends in Cognitive Sciences, 20*(9), 676–688. https://doi.org/10.1016/j.tics.2016.07.002

Shen, J. H., & Tamkin, A. (2026). *How AI impacts skill formation* [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2601.20245

Stadler, M., Bannert, M., & Sailer, M. (2024). Cognitive ease at a cost: LLMs reduce mental effort but compromise depth in student scientific inquiry. *Computers in Human Behavior, 160*, Article 108386. https://doi.org/10.1016/j.chb.2024.108386

Sun, D., Boudouaia, A., Zhu, C., & Li, Y. (2024). Would ChatGPT-facilitated programming mode impact college students' programming behaviors, performances, and perceptions? An empirical study. *International Journal of Educational Technology in Higher Education, 21*, Article 14. https://doi.org/10.1186/s41239-024-00446-5

Tetzlaff, L., Simonsmeier, B., Peters, T., & Brod, G. (2025). A cornerstone of adaptivity – A meta-analysis of the expertise reversal effect. *Learning and Instruction, 98*, Article 102142. https://doi.org/10.1016/j.learninstruc.2025.102142

Wang, Z., Zou, D., Zhang, R., Lee, L.-K., Xie, H., & Wang, F. L. (2026). ChatGPT-enhanced self-regulated learning in programming education: Impacts on motivation, self-efficacy, and learning outcomes. *Interactive Learning Environments, 34*(5), 3041–3066. https://doi.org/10.1080/10494820.2025.2559919

Wu, R., & Yu, Z. (2024). Do AI chatbots improve students learning outcomes? Evidence from a meta-analysis. *British Journal of Educational Technology, 55*(1), 10–33. https://doi.org/10.1111/bjet.13334

Yang, T.-C., Hsu, Y.-C., & Wu, J.-Y. (2025). The effectiveness of ChatGPT in assisting high school students in programming learning: Evidence from a quasi-experimental research. *Interactive Learning Environments, 33*(6), 3726–3743. https://doi.org/10.1080/10494820.2025.2450659

Zhang, L., & Xu, J. (2025). The paradox of self-efficacy and technological dependence: Unraveling generative AI's impact on university students' task completion. *The Internet and Higher Education, 65*, Article 100978. https://doi.org/10.1016/j.iheduc.2024.100978