Engineering Educators Bringing the World Together
2025 ASEE Annual Conference & Exposition
Palais des congrès de Montréal, Montréal, QC • June 22–25, 2025
Paper ID #49295

BOARD #106: Investigating Factors Influencing Performance in an Introductory Programming Course

Amanda Nicole Smith, University of Florida

Amanda is an undergraduate student pursuing a Bachelor of Science in Computer Science at the University of Florida, with an expected graduation in Spring 2025. Her research interests focus on computer science education, particularly how educators can use machine learning models to provide real time intervention strategies to optimize individual student outcomes. This paper is a reflection of her commitment to improving educational strategies and fostering an inclusive learning atmosphere within the engineering community. As a Computer Science major, Amanda is focused on making positive impact through software engineering and machine learning.

Sage Bachus, University of Florida

Sage Bachus is a fourth-year Mechanical Engineering and Pre-Med student at the Herbert Wertheim College of Engineering, University of Florida. His main research focus is in learning analytics and developing a way to better understand the underlying intricacies of how students learn and perform.

Ashish Aggarwal, University of Florida

Ashish Aggarwal is an Instructional Associate Professor of Computer Science in the Department of Engineering Education at the University of Florida's Herbert Wertheim College of Engineering. His research operates at the intersection of Computer Science Education, Learning Analytics, and Artificial Intelligence, focusing on enhancing programming education through innovative technologies and pedagogical approaches. He is currently developing AI-powered tools for student assistance and assessment in large programming courses. He holds a Master's in Computer Science and Management from the University of Florida and a Bachelor's in Computer Science and Engineering from Jaypee University of Information Technology, India.

Investigating Factors Influencing Performance in an Introductory Programming Course

Abstract

Identifying factors that influence undergraduate students' performance in an introductory programming (CS1) course can enable educators to optimize student success. It has become increasingly important for educators to understand and provide customized aid to their students in formal learning environments. In fast-paced and large-class environments, educators face challenges in identifying students' learning needs based on diverse demographic and academic profiles. This study investigates how Gender, Prior Programming Experience (PPE), and Grade Point Average (GPA) impact student success in CS1 courses. The dataset consists of 836 students from six semesters (Spring 2021 to Fall 2023), with demographic and prior experience information collected through surveys. Although in the past, researchers have predicted student success using interaction with course material, previous exam scores, and a combination of many other factors, there is a gap in the literature for using specifically the combination of the pre-course factors of PPE, GPA, and gender to predict performance. Using correlation analysis, logistic regression, and Chi-Square tests, we explored the relationships between these factors and student performance, particularly in predicting whether a student would achieve above or below 80% in course exams before the student starts the course. The logistic regression model achieved a 76% accuracy, with higher precision and recall for identifying students scoring below 80%. GPA showed the strongest positive correlation with performance, while PPE and Gender also exhibited statistically significant relationships, though Gender's impact was minimal. These findings suggest that GPA and PPE are useful predictors for early identification of students at risk of under performance, helping educators develop targeted strategies to support students in programming courses.

1 Introduction

As the demand for computational skills continue to rise in all engineering disciplines, there is an increased focus on integrating programming courses into curricula for non-programming majors [1]. The introductory programming (CS1) course contains students with a range of majors and prior programming experience, which has led to a need for educators to identify demographics to help predict student success in the course. The ability of educators to understand student success and the factors that influence higher performance in programming courses has been growing in importance, specifically for large-class virtual learning environments. By understanding factors that influence student success in introductory programming (CS1) courses, educators who aim to support diverse learners and improve educational outcomes can use this understanding to assist students who fall behind. There have been multiple studies that have indicating early success in CS1 courses can be attributed to interaction with course material, attending class, and joining study groups [2, 3]. However, the majority of the studies have been conducted on predictive factors that consist of students interaction during the course. Given that there is information on the student's profiles before they even start the CS1 course, exploring the profiles of different students and what profiles can predict success in the CS1 course would be valuable for the educator to provide early intervention.

Determining which factors directly correlate to student success is complex and dynamically changing in virtual learning environments. Studies have ranged from research examining the relationship between psycho social factors in predicting student success, to identifying how interaction with clicker quizzes lead to higher exam performance [4, 5]. All these factors previously studied can help identify a persona that can be predicted to be at risk of poor performance in the CS1 course; however, the importance of early detection is not possible in these types of studies. For an educator, having access to factors that can be used to predict student performance before the student embarks on course material will mean having the ability to provide specific engagement with students immediately. Research indicates that high-performing students often demonstrate greater persistence in pursuing programming-related courses, with a direct correlation between academic success and retention [6]. For non-programming majors in a programming course, high-performance is vital for students to retain attendance in programming courses. If educators can propel academic success, both students and universities will benefit from positive outcomes and achievement.

In this paper, we address the gap in providing insights into pre-course predictors of student success in CS1 course for non-programming engineering students. How does ***Grade point average, gender, and prior programming experience*** impact performance in an introductory programming course? Do these factors interact in ways that predict student success or under-performance? In this paper, we intend to answer these questions and identify how well predetermined factors can be used in creating a prediction model for student exam performance. By taking in three predetermined factors on an analyzed dataset of 836 students from 6 semesters (Spring 2021 to Fall 2023), results will help educators understand how demographic and academic characteristics influence learning outcomes, enabling the development of targeted interventions to improve student success in the CS1 course.

2 Background and Related Work

Various methods have been developed to prove how a large range of factors can be used in prediction models to demonstrate how students will perform in a course or even an entire major [3, 4, 7]. With the increased reliance of virtual learning environments post COVID-19, The study by Mbunge et al. (2020) explores the impact of online learning during the COVID-19 pandemic on the performance of first-year computer programming students. The study identifies key factors like motivation, digital literacy, and supportive learning environments as crucial for enhancing retention rates and success in programming courses, particularly for STEM students [8]. Meanwhile, Quille and Bergin [9] revisit and expand upon a decade's worth of predictive modeling research for introductory programming, culminating in the PreSS (Predict Student Success) model that consistently achieves close to 80% accuracy. Their follow-up work explores additional factors (*e.g.*, students' own final-grade predictions, changes in gaming hours) and demonstrates how these newer indicators can boost model performance further. This highlights the value of continually re-examining which predictors remain most relevant as course formats, technologies, and student populations evolve. While many studies in the past have identified predictive factors of student success, it should be noted that the landscape of CS1 courses have changed due to the enhanced reliance of virtual learning environments and the effect that it has on students' performance. It is still relevant however, to note the importance of accounting for traditional factors in prediction models, especially for providing early insights during the course [10].

Multiple other studies have shown that past academic performance (GPA in this study) reflects academic capability and learning habits, making it a reliable feature to correlate with outcomes [10]. One specific study on engineering dynamics found that GPA, when used as the sole predictor in a multiple linear regression model, provides reliable average performance predictions, while models with additional predictors, such as grades in prerequisite courses, further improve accuracy for individual predictions [11]. While PPE has been consistently identified as a strong predictor of student success in programming courses, showing students with prior exposure to coding concepts perform better in the course [12]. Students with prior programming experience enter engineering courses with an advantage, boosting their confidence and engagement in computational tasks. In contrast, those without such experience often face challenges that can undermine their confidence and sense of belonging in the engineering field, reflecting in performance [13].

Gender-specific motivational factors and preferences for learning strategies contribute to performance differences, highlighting the need to consider gender dynamics when designing predictive models and instructional strategies in programming education [14]. Literature indicates specifically that a persistent gender gap was noted in a lecture-based course, where male students consistently outperformed female students[15]. This gap was attributed to differences in engagement and the instructional format. The study redesigns the course to include active learning and hands-on practicum sessions, resulting in the performance disparity between male and female students to be eliminated [15]. This demonstrates that there is a need and success in creating more equitable environments to prevent gender disparities in performance.

Prior Programming Experience (PPE), GPA, and gender have all been identified by other studies as significant predictors of success in programming studies, however, while there has been exploration of these factors, it is important to explore specifically how these three factors interact with each other and can be used in a machine learning model to predict success in the CS1 course. While various factors can influence student performance in introductory programming courses, identifying specific and measurable predictors can help educators develop student profiles. This approach enables early intervention and targeted support, ultimately reducing the risk of poor performance.

3 Research Questions

With a complete understanding of how predictive factors that influence student performance interact with each other and are able to predict student success, educators can be more informed about the profiles of their students to provide the best learning pathways for individual needs. While there are many different potential factors that can be used in predicting student success, predetermined factors will be best for early intervention and identification of students at risk of low performance. Therefore, this research analyzes the factors that influence students' success in exam performance at the beginning of a CS1 course. Consequently, our research questions are:

- ***How do GPA, Gender, and PPE impact performance in an introductory programming course?***
- ***Do these factors interact in ways that predict success or under-performance?***

4 Course Context

The dataset used in this study is from an introductory programming course for all engineering students. The course is two-credits and focuses on introduction to MATLAB. The class was taught using a flipped classroom model at an R1 research university in the US over six semesters: Spring 2021, Fall 2021, Spring 2022, Fall 2022, Spring 2023, Fall 2023. This course has multiple and varying sections, all offering an in-person class subsection (except the Spring 2021 semester, in which the majority of students attended online due to the pandemic). Generally, each of the semesters had over 100 students with the largest semester having 174 students.

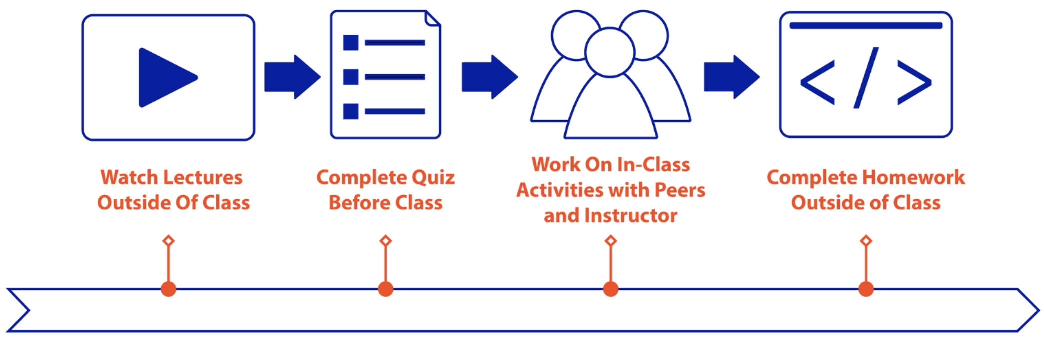

Figure 1: Weekly plan of the CS1 flipped course

Figure 1 shows the standard weekly timeline for the course. The course follows a flipped classroom model where students watch lectures for each module and complete graded quizzes before class. During class time content taught in the video is reviewed and in-class programming problems were completed with the instructor and peers. Homework assignments were assigned weekly and were completed outside of class.

The key programming concepts covered in the modules included input/output, conditionals, while/-for loops, vectors, strings, images, and functions. The course was divided into 14 modules with eight programming-related homework assignments and two exams (a midterm and a final). The last assignment was a cumulative final project.

4.1 Introductory Surveys

Each student was given an introductory survey during the first week of the class. This survey asked for demographic information (gender, prior programming experience (PPE), year in school, GPA, and major), general information (if the student was also taking the lab associated with this course and how many hours of prior programming experience they had), as well as what grade they expected to receive in the course by the end of the semester. Students were given a drop-down menu of achievable grades in the class and asked to choose which grade option they expected themselves to receive. Finally, the students were asked questions about their perceptions of programming and learning preferences. For each question, five options on the Likert scale were given: strongly agree, somewhat agree, somewhat disagree, strongly disagree, and neither agree nor disagree.

5 Demographics

The dataset consists of 856 students whose data was collected across six semesters of the CS1 course. Across the original data, 836 remained after discarding student data with missing information, including missing exam scores or improper GPA scores. The distribution of data across the 6 semesters is 112 from Spring of 2021, 166 from Fall 2021, 135 from Spring 22, 173 from Fall 2022, 108 from Spring 2023, and 142 from Fall 2023. This balanced distribution ensures a comprehensive analysis of different academic periods combined into one larger dataset. Performance score is calculated based on the 4 exam scores in the data.

The dataset captures gender as a binary variable, where 0 represents female students and 1 represents male students. The gender distribution shows that there are 530 male students and 306 female students. Table 1 and Figure 2 shows the breakdown of gender over the 6 semesters recorded in the dataset. While male students consistently outnumber female students, the gap fluctuates across semesters, with the smallest difference observed in Fall 2021 and Fall 2023. This predominance of male students is typical in many engineering courses, which suggests the need to explore how gender dynamics might influence academic performance in this context.

Table 1: Breakdown of Data Based On Gender for Each Semester

	Spring 21	Fall 21	Spring 22	Fall 22	Spring 23	Fall 23
$Female$	38	74	39	63	30	62
$Male$	74	92	96	110	78	80
Total	112	166	135	173	108	142

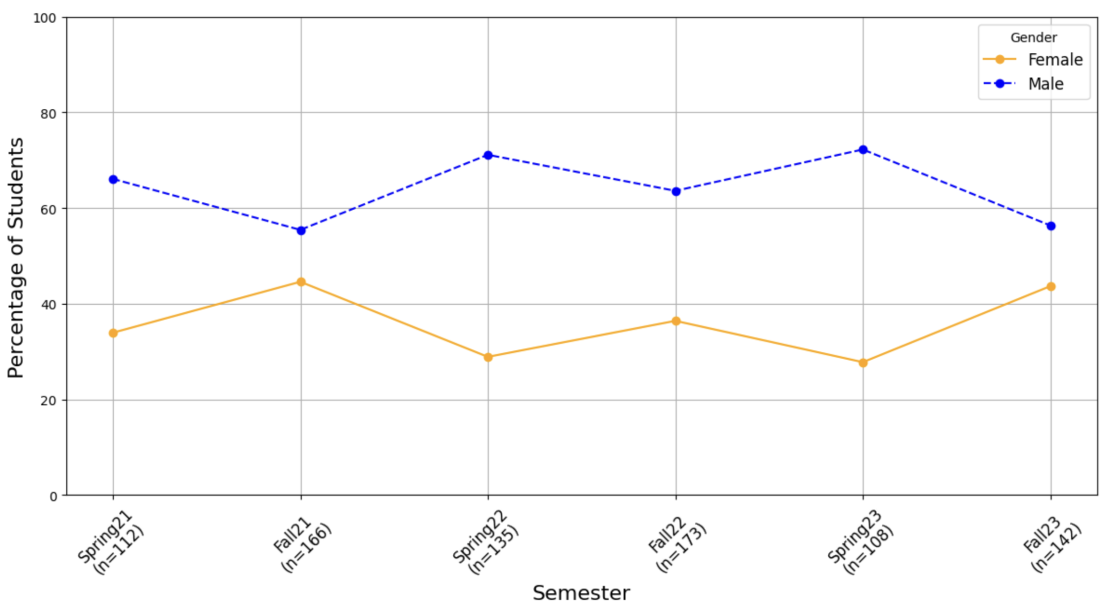

Figure 2: Percentage of Students Enrolled Based on Gender Over 6 Semester (Total n = 836)

Prior programming experience (PPE) is recorded as a binary variable, with 0 indicating the student has no prior experience and 1 representing students with prior programming experience. The dataset reveals that 503 students (60.17%) have no prior programming experience, while 333 students (39.83%) have some form of PPE as shown in Table 2 and Figure 3. This substantial proportion of students without programming experience underscores the importance of examining the role PPE plays in shaping academic success in the course. The fact that 60.17% of students having no prior programming experience highlights the challenge of teaching non-programmers in an introductory course and suggests that PPE may play a significant role in shaping student performance.

Table 2: Breakdown of Data Based On Prior Programming Experience for Each Semester

	Spring 21	Fall 21	Spring 22	Fall 22	Spring 23	Fall 23
$NoPPE$	60	116	68	109	66	84
PPE	52	50	67	64	42	58
Total	112	166	135	173	108	142

350

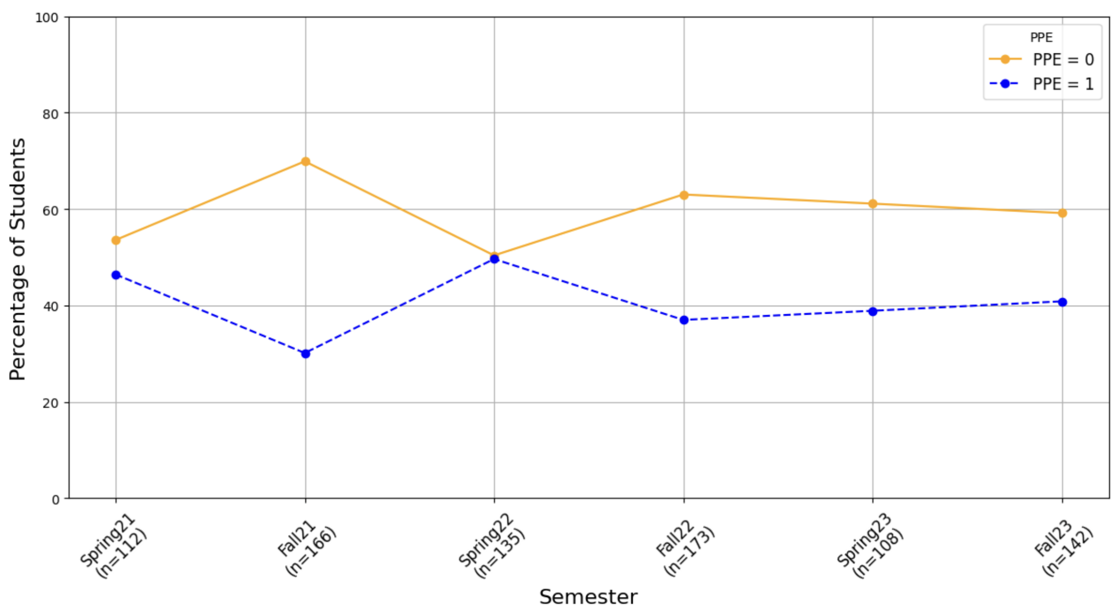

Figure 3: Percentage of Students Enrolled Based on Prior Programming Experience Over 6 Semesters (Total n = 836)

Students were asked to self-report their academic standing as the GPA going into the course. GPA is on a 4.00 scale with a minimum GPA as 0.00. Out of 836 students, the dataset had a mean GPA of 3.61 and a standard deviation of 0.37. The minimum GPA recorded is 0.00, while the 25th, 50th (median), and 75th percentiles are 3.40, 3.69, and 3.91, respectively. The maximum GPA recorded is 4.00 after outliers above 4.00 were dropped. These statistics suggest that the cohort consists of high-achieving students, with relatively low variability in their GPAs, as evidenced by the standard deviation of 0.37. The plots in Figure 4 show the exact distribution over each semester.

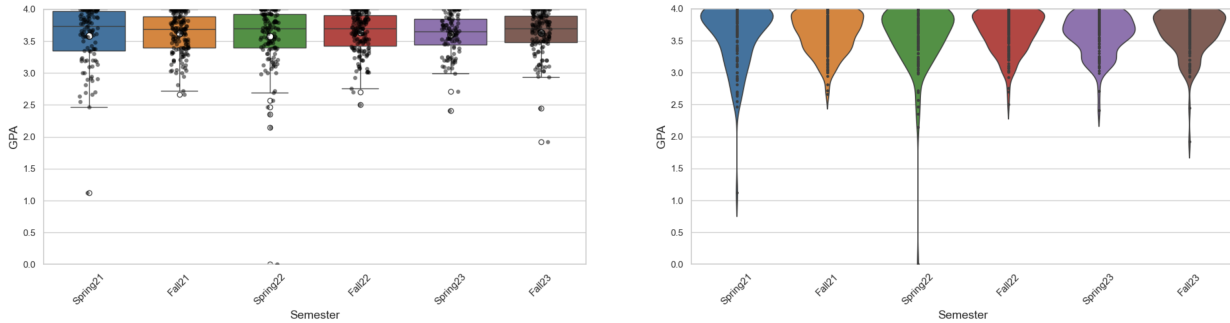

Figure 4: Distribution of Students' GPA each semester

6 Methodology

To understand how gender, PPE, and GPA influence a student's exam performance, data was collected over six semesters via an introductory survey during the first week of class within a CS1 course.

Regarding the PPE, the students were given the options: ”No prior programming experience”, ”Between 1-10 Hours of experience”, ”Between 11-100 hours of experience”, ”Between 101-500 hours of experience”, or ”Created a programming language.” If students selected the first option, they were assigned a 0 for PPE. If they selected the 3rd-5th option, they were assigned a value of 1 and were asked to submit an explanation of their experience. If they selected ”Between 1-10 hours of experience,” their explanations were used to manually determine whether or not they would receive a 0 or 1.

To choose these students' level of experience, a set of rules was applied. If only the following criteria was met, the student received a 0.

- The student's stated experience was only block coding (Scratch, Tinkercad, Arduino, etc.).
- The student insinuated that their knowledge was limited, basic, unsure (a tutorial/entry course would fall under this category as well).
- If they took a non-programming course in which they only used a small amount of programming.

If any of the following criteria was met, the student received a 1:

- If the student mentions a programming language without going into detail about depth of knowledge.
- The student took any level of programming class/online course in middle school, high school, or college including any AP/AICE level computer science class.

- The student received any programming certification.
- If the student is confident in their abilities.
- If the student started a course, but did not finish it. (However, this can be unclear and can be left to the author's discretion)

All other examples, and examples that are listed above but are not clear, were left to the author's discretion.

Regarding gender, students were given the options: "Male", "Female", and "Other", with the addition of an optional comment box. If a student responded with "Male" they were assigned a value of 1, a student who responded with "Female" was assigned a value of 0, and all other responses were dropped from the dataset.

Regarding GPA, students submitted their grade point averages through a comment box. The expected input was a decimal number on the 4.0 GPA scale. Those who reported no GPA or left a comment were filtered out of the data.

Student's exam performance score was calculated based on the average of 4 exam scores. Any students who had a missing exam score E1P1, E1P2, E2P1, E2P2 were removed from the dataset. To handle semester-specific differences in grading, performance was standardized using Min-Max scaling with each semester. Normalizing performance scores within each semester ensure the range of scores for each semester is consistently scaled in order to compare scores across semesters. The min-max scaling was applied separately for each semester to rescale performance scores to a range of 0 to 100. The performance scores were then adjusted for consistency across semesters to account for varying course difficulty. Adjusting the scaled scores helped to balance the average performance across semesters. A correction factor was computed by dividing the overall average performance by the average performance of each semester. Each semester's scaled scores were multiplied by the correction factor to align them with the overall average. Finally, to facilitate logistic regression, performance was converted to a binary outcome. A threshold of 80 was used to create a binary of the adjusted performance scores. If the adjusted performance score is greater than 80, it is labeled as 1 (indicating high performance); otherwise, it is labeled as 0 (indicating low performance). The distribution of performance scores is shown in Figure 5 from before scaling, after scaling, and after adjustments are made.

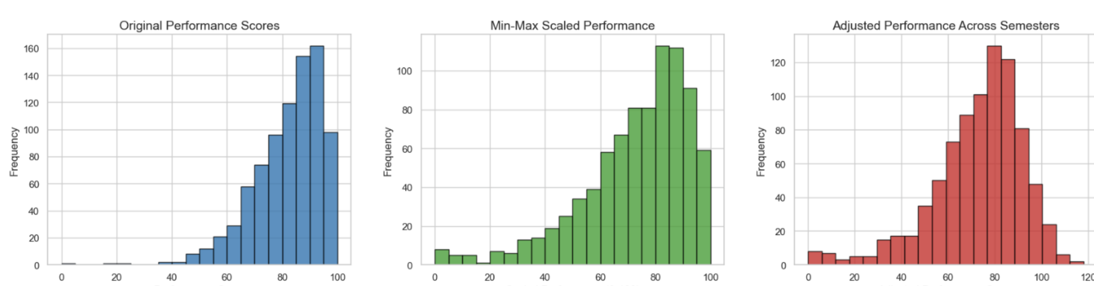

Figure 5: Distribution of Performance Scores: Original, Scaled, and Adjusted

We first analyzed the relationship between each predictor and each predictor's influence on student performance using statistical analysis. The Pearson Correlation Coefficient (r) was used to assess the correlation between GPA and performance given the continuous nature of both variables. Pearson's correlation coefficient helps us understand whether there is a linear relationship between two variables. The Pearson correlation coefficient ranges from -1 to +1, where -1 indicates a perfect negative linear relationship and +1 indicates a perfect positive linear relationship. The Wilcoxon Rank-Sum Test was used on PPE and gender which are two binary variables. The Wilcoxon Rank-Sum Test is a non-parametric test and it does not assume normality of the data. The Chi-square test was used to evaluate independence between gender and PPE and assess if the distribution of PPE differed between male and female.

To understand how each predictor collectively influences performance, a logistic regression model was developed. The model used the surveyed variables (GPA, Gender, and PPE) as independent variables to predict the binary outcome of exam performance (above or bbelow 80%).

7 Findings

The analysis of the dataset provided insights into the relationships between performance and three key predictors: GPA, Gender, and PPE.

RQ1: How do GPA, Gender, and PPE impact performance in an introductory programming course

We used the Pearson Correlation coefficient because it gives a way of measuring a linear correlation. Since GPA is continuous, Pearson's coefficient provides a clear indication of whether higher GPAs are associated with better performance outcomes in the course. The Pearson correlation coefficient between GPA and performance is $r = 0.48$, indicating a moderate positive correlation. This suggests that students with higher GPAs tend to perform better in the course. ***The corresponding p-value is highly significant ($p < 0.001$), and this means that the relationship between GPA and performance is unlikely to be due to chance, and GPA has a notable correlation to student performance.***

We use the Wilcoxon Rank-Sum (Mann-Whitney U) test to compare performance scores across the two interdependent groups (gender and PPE), which is appropriate when data does not meet normality assumptions. The test revealed a statistically significant difference between male and female students' performance $U = 70098$, with an extremely small p-value of ($p < 0.001$), indicating a statistically significant relationship between gender (where 1 represents male and 0 represents female) and performance. Despite the statistical significance, ***this result suggests that while there is a difference in performance based on gender, it may not be a strong predictor of performance in this course on its own.*** Although gender is statistically significant (p ¡ 0.001), its practical impact on student performance is minimal. This indicates that while the relationship is detectable, gender's predictive value is limited, as reflected in the lower ROC AUC and accuracy 4. Figure 6 shows the distribution of gender in relation to performance using a Kernel Density Estimate (KDE) plot. It is important to note the KDE plot shows the probability density of where male and female performance values are most likely to occur, the graph does not display exact count or percentage of data points. It's important to recognize that observed differences between gender groups should not be interpreted as definitive proof of causation and should be approached cautiously to avoid making broad generalizations.

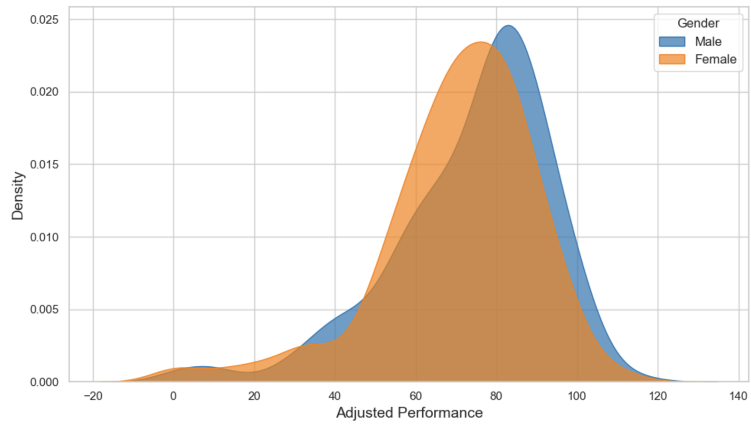

Figure 6: Distribution of Gender and Adjusted Performance

The rank-sum test statistic for PPE vs performance is $U = 67576$, with $p < 0.001$, indicating a statistically significant difference in performance between students with and without prior programming experience. This suggests that students with prior programming experience tend to perform differently in the course. ***The extremely small p-value (p <0.001) confirms that the relationship between PPE and performance is statistically significant, meaning that prior programming experience has a measurable impact on student performance.*** Figure 7 shows this difference in performance between students with and without prior programming experience.

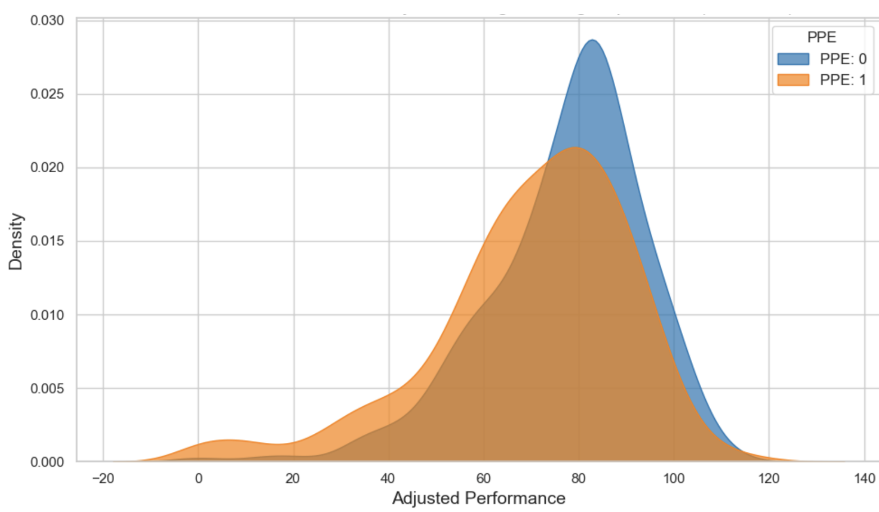

Figure 7: Distribution of Prior Programming Experience and Adjusted Performance

Taken together, these findings align with our broader analysis: GPA is the strongest overall predictor of performance, followed by PPE, while gender shows only a modest association. This suggests that academic standing (GPA) and prior exposure to coding are the most consequential factors for students' success in this introductory programming context.

RQ2: Do these factors interact in ways that predict success or underperformance?

Is There a Statistically Significant Relationship Between Gender and PPE?

The Chi-Square test was conducted to examine the independence between Gender and PPE. The test resulted in a $\chi^2 = 13.86$ with a p-value of 0.0002. The low p-value indicates a statistically significant relationship between Gender and PPE, suggesting that the distribution of students with prior programming experience differs significantly between male and female students. Table 3 and Figure 8 show the distribution of PPE by gender.

Table 3: Breakdown of Prior Programming Experience by Gender

Gender	No PPE (%)	Has PPE (%)	Total
Female	210 (68.63%)	96 (31.37%)	306
Male	293 (55.28%)	237 (44.72%)	530

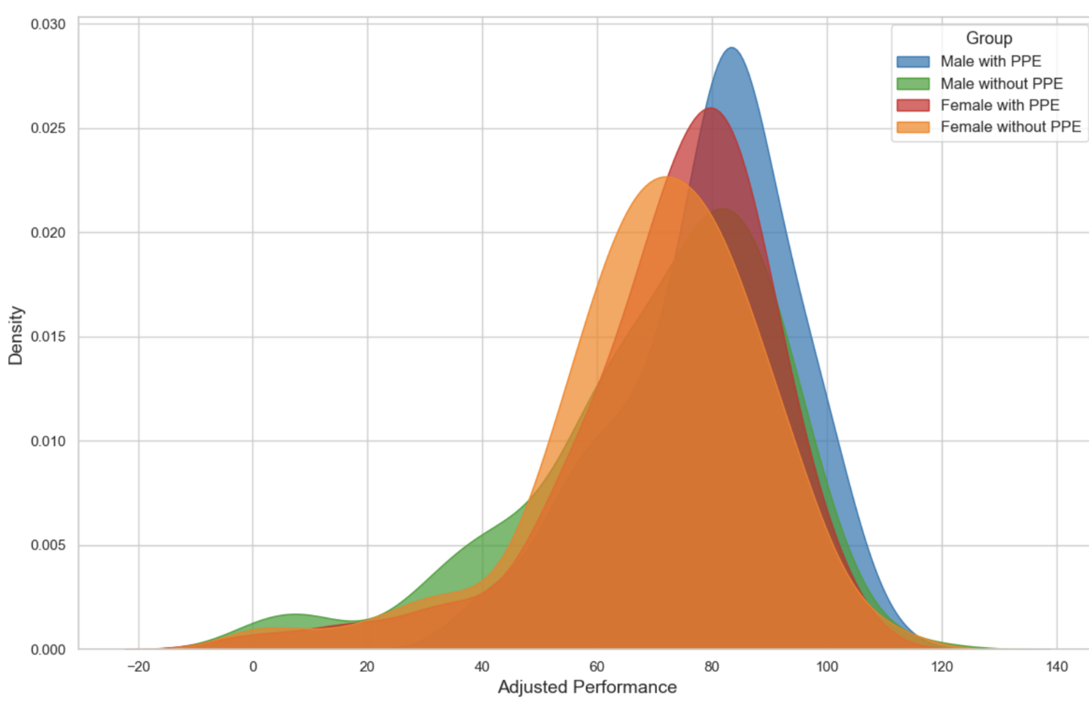

Figure 8: Distribution of Gender, Prior Programming Experience, and Adjusted Performance

Is There a Statistically Significant Relationship Between Gender and GPA?

A Mann-Whitney U test (Wilcoxon Rank-Sum test) was conducted to compare the GPA of male and female students. The test produced a U-statistic of 87,851.5 with a p-value of 0.044. Since the p-value is less than the commonly used threshold of 0.05, we reject the null hypothesis. This suggests that there is a statistically significant difference in GPA between male and female students, indicating that gender influences the GPA distribution. Figure 9 demonstrates the distribution of gender and GPA.

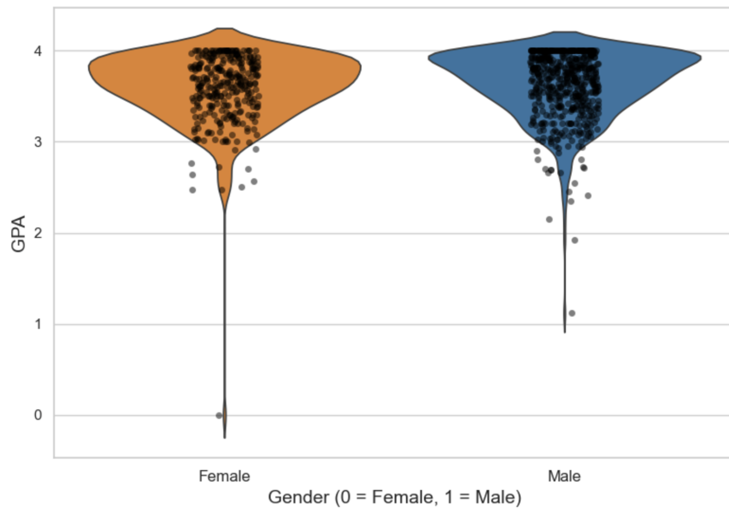

Figure 9: Distribution of Gender and GPA

Is There a Statistically Significant Relationship Between GPA and PPE?

A Mann-Whitney U test was also performed to compare the GPA of students with and without prior programming experience. The test resulted in a U-statistic of 88,953.0 and a p-value of 0.127. Since the p-value is greater than 0.05, we fail to reject the null hypothesis. This indicates that there is no statistically significant difference in GPA between students who have prior programming experience and those who do not. Figure 10 shows a visual of this distribution.

Logistic Regression Findings

Our logistic regression models are of the form $log\left(\frac{P(Performance>80)}{P(Performance\leq 80)}\right)$ i.e., the dependent variable is the binary outcome of student exam performance, coded as 1 if performance is above 80% and 0 if it is 80% or below given the combination of gender, PPE, and GPA. The 80% threshold is used in the logistic regression to distinguish between high and low performance, aligning with traditional grading scales where 80% typically marks a "B" grade, indicating solid competence. It serves as a clear cutoff for identifying students at risk of under-performing, allowing for targeted interventions before significant difficulties arise.

We evaluated seven logistic regression models with different combinations of predictors (GPA, PPE, and Gender) to determine the best predictive model. We assessed the goodness of fit using both the Receiver Operating Characteristic curve (ROC curve) and the Area Under the Curve. The ROC curve evaluates a model's ability to differentiate between classes by plotting the true positive rate against the false positive rate at various thresholds. The AUC quantifies this performance, with values closer to 1 indicating stronger classification accuracy, making a high AUC the desired outcome. We used these metrics and selected the best model based on these criteria. The results are given in Table 4.

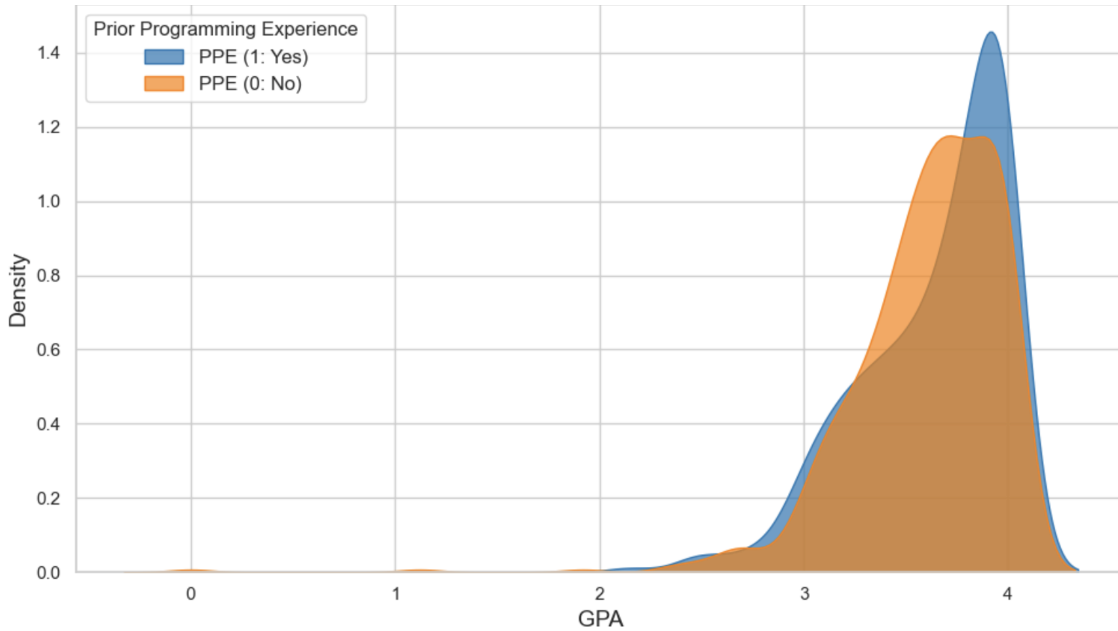

Figure 10: Distribution of Prior Programming Experience and Adjusted Performance

Table 4: Performance Metrics of Different Models

Model	ROC AUC	Accuracy
Gender only	0.5463	0.6011
GPA only	0.8101	0.7738
PPE only	0.5680	0.5833
GPA + Gender	0.8089	0.7619
GPA + PPE	0.8123	0.7559
Gender + PPE	0.5820	0.6309
GPA + Gender + PPE	0.8120	0.7559

Table 5: P-values of Different Models

Model	p-values
Gender only	{`Gender`: 0.003}
GPA only	{`GPA`: 2.9e-26}*
PPE only	{`PPE(Binary)`: 2.3e-05}*
GPA + Gender	{`GPA`: 6.9e-26, `Gender`: 0.016}*
GPA + PPE	{`GPA`: 2.3e-26, `PPE(Binary)`: 5.1e-05}*
Gender + PPE	{`Gender`: 0.014, `PPE(Binary)`: 9.2e-05}*
GPA + Gender + PPE	{`GPA`: 4.9e-26, `Gender`: 0.056, `PPE(Binary)`: 1.6e-04}*

*Statistically significant at $p < 0.05$.

Table 5 indicates that based on p-values of each individual predictor used in the model, most models have statistical significance, indicated by an asterisk. However, in the GPA + Gender + PPE model, gender is not statistically significant ($p = 0.056$), suggesting it contributes less to performance prediction in this setting. Gender, while it is statistically significant, has poor results for the ROC AUC, and accuracy which shows weaker performance with higher p-values, indicating that gender is not as influential in predicting exam outcomes. ***Based on the results presented in Table 4, using the ROC and AUC values with significant coefficients, we find that the "GPA + PPE" model achieves the highest ROC AUC (0.8123) and a high accuracy (0.76), indicating it has the best predictive power among the combinations tested.*** Models using only GPA or combining GPA with gender also perform at a high level indicating that GPA is the strongest predicting feature. The model with only GPA as a predictor has the best performance accuracy (0.77) but a slightly lower ROC AUC (0.8101) compared to the GPA + PPE model. This suggests that while GPA alone is a strong predictor of exam outcomes, incorporating PPE enhances the model's ability to distinguish between performance levels. The GPA + Gender model also performs well, reinforcing GPA's predictive strength, but gender's weaker statistical significance and lower impact on AUC suggest it is not as critical in determining student performance. Overall, these findings highlight that GPA is the most influential factor, while PPE adds meaningful predictive value, making the GPA + PPE model the best choice for performance prediction. Models based solely on gender or PPE have lower accuracy and ROC AUC.

Based on the results presented in Table 4, using the ROC and AUC values with significant coefficients, we find that the fifth model, where GPA and PPE explain a student's outcome expectancy, has the strongest results between the goodness of fit and model complexity, so we select:

$$\log\left(\frac{P(\text{Performance} = 1)}{P(\text{Performance} = 0)}\right) = -15.091 + 3.938 \times \textbf{GPA} + 0.756 \times \textbf{PPE}$$

as our best model of the seven we fit. Although the seventh model, which includes GPA, PPE, and gender, also demonstrated strong predictive power, it was not selected as the final model. The

GPA + PPE model had a slightly higher ROC AUC (0.8123) compared to the combined model's (0.8120) which is nearly identical. The inclusion of gender in the seventh model did not significantly enhance performance but added complexity. By focusing only on GPA and PPE, the model maintains simplicity while still achieving robust prediction. However, it is crucial to note that the ROC AUC and accuracy metrics do not fully capture whether the model is a good fit for our data distribution. Additional tests, such as residual analysis and model validation with external data, are necessary to confirm the robustness of our selected model.

Logistic regression is built on several core assumptions: a binary outcome, independent observations, absence of multicollinearity, linear relationships between independent variables and the log-odds of the outcome, and a sufficiently large sample size. In this study, our binary outcome variable is student performance (pass/fail above or below 80%). We include data from 836 students across six semesters. The Variance Inflation Factors (VIF) for GPA (3.135) and PPE (1.691) suggest that multicollinearity is not a major concern, as both values remain below the commonly used threshold of 5. The Pearson correlation of 0.43 ($p <$ 0.05) between GPA and the outcome indicates a moderate positive relationship, supporting the inclusion of GPA as a meaningful predictor. This model confirms the viability of using GPA and PPE to predict student performance. Table 6 shows the results of the VIF and significance tests.

Table 6: Verifying Logistic Regression Results

Metric	Value	Interpretation
VIF (GPA)	3.135	No multicollinearity
VIF (PPE)	1.691	No multicollinearity
Pearson's r	0.43	Moderate positive correlation
p-value	2.84×10^{-38}	Statistically significant

Satisfying the assumption of linearity of independent variables and log-odds, our model confirms the validity of predicting the expected grade of a student based on their prior programming experience and GPA. Because GPA is on a narrow 0–4 scale, a 0.1 point increase significantly influences the chances of success, increasing them by approximately 1.48. This highlights that students with higher GPAs are much more likely to succeed, reinforcing GPA as the most influential factor in predicting performance. Similarly, students with prior programming experience are 2.13 times more likely to pass than those without PPE.

7.1 Model Diagnostics

To ensure our logistic regression model adhered to underlying assumptions and provided a proper fit, we conducted three additional tests: (1) a Box–Tidwell procedure to evaluate linearity in the log-odds, (2) a residual analysis (deviance and Pearson residuals) to identify possible outliers or misfit patterns, and (3) the Hosmer–Lemeshow (H–L) goodness-of-fit test to compare predicted versus observed outcomes.

Box–Tidwell Test: Since GPA is a continuous predictor, we applied the Box–Tidwell method to investigate whether GPA's relationship with the log-odds is truly linear. The analysis produced a significant interaction term (GPA $\times \ln(\text{GPA})$, $p < 0.001$), suggesting that GPA's effect on performance is *nonlinear*. This indicates a single linear term for GPA may underestimate or overestimate the effect at lower or higher GPA ranges.

Residual Analysis: We fit the model using a generalized linear model (GLM) with a binomial family, which enabled us to compute deviance and Pearson residuals. Plotting these residuals against predicted probabilities showed moderate clustering rather than random scatter, implying the model may be missing a crucial factor or that certain observations act as outliers. Ideally, if the model is well-specified and captures the data patterns accurately, the residuals should cluster randomly around zero without forming systematic 'bands' or curved shapes. In figure 11, the two distinct arcs show that the model may be missing some structure (for example, a non-linear term), because residuals become increasingly positive or negative at certain predicted probabilities rather than remaining close to zero.

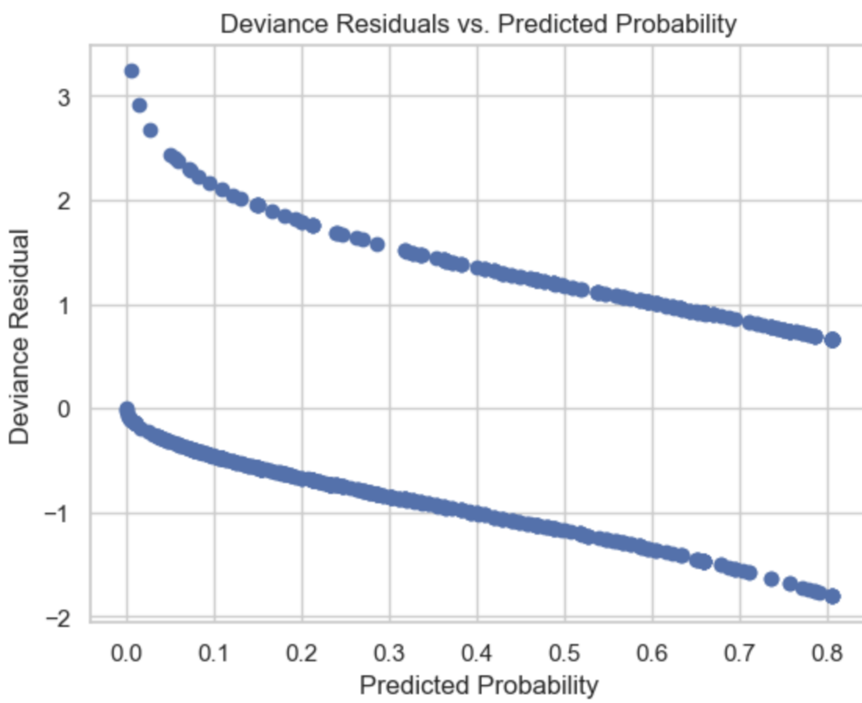

Figure 11: Deviance residuals plotted against the predicted probabilities. The residuals exhibit two notable arcs, indicating a systematic pattern that suggests the model may be over- or underestimating performance in specific probability ranges.

Hosmer–Lemeshow Test: Lastly, the H–L test ($\chi^2 = 24.29,\ p = 0.002$) signaled a statistically significant lack of fit. This finding, coupled with Box–Tidwell results and residual patterns, strongly suggests the current model's linear assumption for GPA may be too restrictive. Introducing additional predictors or modeling GPA with polynomial terms could yield better calculations and predictive performance.

In summary, these diagnostics confirm that while GPA and PPE are robust predictors of exam outcomes, the logistic model's basic linear form may not fully capture GPA's effect. Future research will explore *nonlinear transformations* or expanded modeling approaches to enhance predictive accuracy of the course performance predictions.

8 Discussion

Our first research question asked: ***How do GPA, Gender, and PPE impact performance in an introductory programming course? Results from the logistic regression analysis confirm that GPA and PPE are significant predictors of student performance, while gender has a minor but statistically significant impact.*** This suggests that academic performance (measured by GPA) and prior experience in programming play major roles in determining success, aligning with earlier studies that emphasize the importance of these factors in programming courses [12]. The moderate correlation between GPA and performance ($r = 0.48$) further validates GPA as a reliable predictor.

Prior studies consistently demonstrate that GPA is one of the most robust indicators of student performance across various academic contexts, as it reflects a student's cumulative learning habits, self-regulation, and perseverance. For instance, research on engineering dynamics courses has shown that cumulative GPA alone can serve as a strong predictor of performance, particularly when used in conjunction with other academic metrics like grades in prerequisite subjects[11]. These findings suggest that GPA's predictive power stems from its ability to capture a wide range of skills that are foundational to learning, making it a valuable predictor in programming courses where logical problem-solving and disciplined study habits are crucial for success.

Our second question focused on whether these ***factors interact to predict success or underperformance***. Chi-Square tests indicate a significant relationship between **gender** and **PPE**, implying that male students are more likely to have prior programming experience. However, the lack of significant interaction between **GPA** and **PPE** suggests their effects on performance are largely independent. These results highlight that while academic standing and prior experience are critical, gender's influence is more nuanced and warrants further exploration.

Prior research in programming education highlights the importance of PPE, especially in fast-paced introductory courses. Students with prior programming exposure often report increased confidence and lower cognitive load when learning new concepts, which may help them engage more fully and sustain their performance throughout the course [13]. For students without PPE, research suggests that a lack of early exposure to programming can create additional barriers to success, as they may face steeper learning curves and struggle with foundational skills that others take for granted [13]. This disparity underscores the importance of support mechanisms, such as targeted tutorials or supplemental resources, for students entering programming courses without PPE.

Gender also plays a complex role in academic performance within STEM fields. Studies have shown that when instructional methods in engineering and programming emphasize active learning and collaborative environments, performance gaps between male and female students can be significantly reduced [15]. This suggests that while gender alone may not strongly predict performance, gender-inclusive pedagogical approaches can positively impact learning outcomes. Therefore, while gender itself showed a limited effect in this study, the interaction between gender, prior experience, and course design may present valuable areas for future investigation to create more equitable and supportive learning environments.

9 Limitations

This study has several limitations that must be acknowledged. First, the generalizability of the findings is constrained by the focus on a single institution's CS1 course, which may not represent other universities, programming languages, or instructional methods. Additionally, simplifying gender and prior programming experience to binary variables may overlook the nuanced variations within these categories. The reliance on self-reported GPA and PPE introduces potential biases or inaccuracies. Furthermore, the scope of the analysis is limited to short-term course outcomes, without consideration of longer-term retention or performance in advanced courses. Lastly, the study primarily examines pre-course predictors (GPA, gender, and PPE) while excluding other influential factors, such as motivation, socioeconomic status, and study habits, which could affect student performance in programming courses.

10 Future Work

Since a student's exam performance in a CS1 course can be attributed to many different factors, this study can be expanded to include other factors including student motivation, quiz performance, and interaction with other given course materials. Additionally, it would be interesting to see if the predictors used in this study can be used to predict performance in other classes or universities. Building upon a more complex model to predict specific grades a student will achieve in the course beyond a pass/fail would lead to a greater use of the prediction for more customized intervention from educators.

11 Conclusion

This analysis reveals the complex relationships between GPA, Gender, and PPE and how they influence student performance in an introductory programming course. It is increasingly important for all engineering majors to perform well in programming courses as computational skills grow in essentiality all fields. It is equally important for educators to have a prior knowledge of students to provide early intervention in large classes. While GPA has the strongest correlation with performance, PPE also contributes to shaping learning outcomes, with each factor revealing distinct patterns. ***The model highlights how students with higher GPAs and PPE consistently perform better, while gender differences, though less pronounced, still suggest potential areas for targeted interventions.*** These insights can guide early support strategies tailored to individual student profiles, helping educators enhance success rates in programming courses. The results emphasize the need for targeted interventions, especially for students without prior programming experience or with lower GPAs, as they may be at higher risk of under-performance. The findings of this study can be used by educators and curriculum developers to design more personalized support strategies, ensuring that all students, regardless of their background or experience, are equipped to succeed in programming courses.

References

[1] P. K. Chilana, T. K. Parikh, and A. J. Ko, "How students learn in CS introductory courses: Insights from a management engineering program," in *Proceedings of the 2015 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. Atlanta, GA, USA: IEEE, 2015, pp. 221–229, accessed: 21-Oct-2024. [Online]. Available: https://pg.ucsd.edu/ChilanaVLHCC15CameraReadyFinal.pdf

[2] J. M. Llanos-Mosquera, J. A. Quimbayo-Castro, E. Garcia-Perdomo, and A. H. Alarcon-Lopez, "Effect of early intervention on students in a cs1 programming course," in *Applied Informatics. ICAI 2024*, ser. Communications in Computer and Information Science, H. Florez and H. Astudillo, Eds. Springer, Cham, 2025, vol. 2237, accessed: 21-Oct-2024. [Online]. Available: https://doi.org/10.1007/978-3-031-75147-9_4

[3] K. Shah, P. Lee, D. Barretto, and S. N. Liao, "A qualitative study on how students interact with quizzes and estimate confidence on their answers," in *Proceedings of the 26th ACM Conference on Innovation and Technology in Computer Science Education V. 1*, ser. ITiCSE '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 32–38. [Online]. Available: https://doi.org/10.1145/3430665.3456377

[4] S. N. Liao, "Early identification of at-risk students and understanding their behaviors," Ph.D. dissertation, UC San Diego, 2019, proQuest ID: Liao_ucsd_0033D_18490, Merritt ID: ark:/13030/m5sn5d72. [Online]. Available: https://escholarship.org/uc/item/7t05x04q

[5] C. J. Fong, C. W. Davis, Y. Kim, Y. W. Kim, L. Marriott, and S. Kim, "Psychosocial factors and community college student success: A meta-analytic investigation," *Review of Educational Research*, vol. 87, no. 2, pp. 388–424, 2017.

[6] G. D. Caruth, "Student engagement, retention, and motivation: Assessing academic success in today's college students," *Participatory Educational Research (PER)*, vol. 5, no. 1, pp. 17–30, 2018. [Online]. Available: http://eoi.citefactor.org/10.11203/per.18.4.5.1

[7] S. N. Liao, D. Zingaro, K. Thai, C. Alvarado, W. G. Griswold, and L. Porter, "A robust machine learning technique to predict low-performing students," *ACM Trans. Comput. Educ.*, vol. 19, no. 3, Jan. 2019. [Online]. Available: https://doi.org/10.1145/3277569

[8] E. Mbunge, S. G. Fashoto, and J. Olaomi, "Covid-19 and online learning: Factors influencing students' academic performance in first-year computer programming courses in higher education," *SSRN Electronic Journal*, 2021. [Online]. Available: https://ssrn.com/abstract=3757988

[9] K. Quille and S. Bergin, "Programming: Further factors that influence success," in *Proceedings of the 27th Annual Workshop of the Psychology of Programming Interest Group (PPIG)*. University of Cambridge, Cambridge, UK: PPIG, 2016, https://www.ppig.org.

[10] J. P. J. Pires, F. Brito Correia, A. Gomes, A. R. Borges, and J. Bernardino, "Predicting student performance in introductory programming courses," *Computers*, vol. 13, no. 9, p. 219, 2024. [Online]. Available: https://doi.org/10.3390/computers13090219

[11] S. Huang and N. Fang, “Predicting student academic performance in an engineering dynamics course: A comparison of four types of predictive mathematical models,” *Computers & Education*, vol. 61, pp. 133–145, 2013.

[12] J. Llanos, V. A. Bucheli, and F. Restrepo-Calle, “Early prediction of student performance in cs1 programming courses,” *PeerJ Computer Science*, vol. 9, p. e1655, 2023, open Access under CC-BY 4.0 License. [Online]. Available: https://doi.org/10.7717/peerj-cs.1655

[13] N. V. Mendoza Diaz, D. A. Trytten, R. Meier, and J. M. Moore, “The impact of prior programming experience on computational thinking in first-year engineering experience,” in *2022 American Society for Engineering Education Annual Conference & Exposition*. American Society for Engineering Education, 2022.

[14] R. Mellado, C. Cubillos, R. M. Vicari, and G. Gasca-Hurtado, “Leveraging gamification in ict education: Examining gender differences and learning outcomes in programming courses,” *Applied Sciences*, vol. 14, no. 17, p. 7933, 2024. [Online]. Available: https://doi.org/10.3390/app14177933

[15] N. K. Lape, C. Clark, L. Bassman, M. Spencer, A. Lee, R. E. Spjut, A. M. Dato, L. Palucki Blake, and T. Tsai, “Erasing a gender gap in performance in a multidisciplinary introductory engineering course,” in *2018 CoNECD - The Collaborative Network for Engineering and Computing Diversity Conference*. Crystal City, Virginia: American Society for Engineering Education, 2018.