Estimating the Effect of a Teacher Training Program on Advanced Placement® Outcomes

Richard S. Brown[1]
Emily Anne Brown[2]

[1] West Coast Analytics
[2] University of North Texas

DOI:10.21585/ijcses.v2i4.35

Abstract

This study employs a potential outcomes modeling approach to estimate the effect of Code.org's Professional Learning Program on Advanced Placement (AP) Computer Science Principles test taking and qualifying score earned for a recent cohort of 167 schools compared to a matched group of comparison schools. Results indicate substantial and significant increases in both Computer Science AP test taking and qualifying score earning for all students. In addition, the significant effects were even greater for Computer Science AP test taking and qualifying score earned by female and minority students when impact ratios are analyzed separately. This study provides evidence of a teacher training program that is having a significant and important impact on preparing more students to succeed in computer science and improve the future of computer science education in this country.

Keywords: computer science, professional development, teacher training

1. Introduction

Despite the growing need for qualified workers in STEM fields, there remains a significant under-representation of females in STEM fields (Beede, et al., 2011) and specifically in Computer Science careers (Sax, et al., 2017). Similar gaps exist for minority students. Research has shown that targeted training of teachers to provide Computer Science courses can increase the number of minority students enrolled in advanced Computer Science courses (Goode, 2007). Goode argues that there is a critical need to provide professional development to support and encourage minority participation in Computer Science coursework. This study employs a potential outcomes modeling approach to estimate the causal effect of Code.org's Professional Learning Program.

Code.org, a nonprofit 501(c) (3), works across the education spectrum to expand access to computer science and increase participation by women and underrepresented minority populations in computer science coursework. Code.org believes that every student in every school should have the opportunity to learn computer science, just like biology, chemistry or algebra. In addition to developing curricula for grades K-12, Code.org provides professional development for high school educators. The Code.org Professional Learning Program offers both in-person and online support for teachers before and during their first year teaching the Code.org curriculum. To date, several thousand teachers completed the program, with the majority ranking it as among the best professional development of their careers.

The Code.org Professional Learning Program is a multi-pronged approach to ensure the quality and sustainability of the program at scale. The program represents a coordination of three major Code.org efforts -- Regional Partners, Facilitator Development, and Professional Development Workshops -- all built upon the foundations and principles of Code.org curricula which has been designed to meet learning objectives through engagement with equitable classroom practices.

Taken altogether, the Professional Learning Program can be summarized as a year of ongoing Professional Development Workshops for teachers with agendas and activities designed specifically for the Code.org CS Principles Curriculum and teaching philosophies. Workshops are run by Code.org Professional Development Facilitators who receive training in a separate, year-long program devoted to PD leadership development specifically designed to support the Code.org CS Principles Curriculum. Teachers are supported from the beginning of the program to the end by Code.org Regional Partners who collaborate with facilitators to deliver high quality workshops. Code.org Regional Partners are developed through a multi-year partnership with the aim of building local, sustainable hubs of high quality PD for computer science teachers. Teachers also have additional ongoing supports such as the Code.org Forum, an online professional learning community.

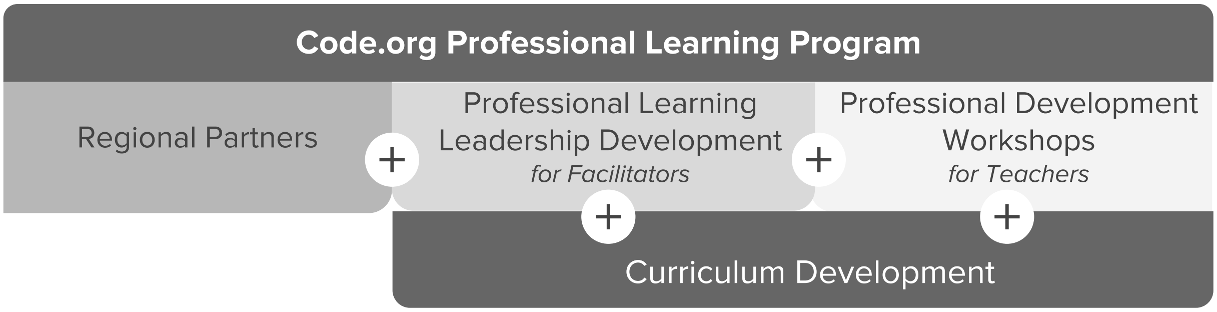

Code.org Professional Learning Program
Regional Partners
+
Professional Learning Leadership Development
for Facilitators
+
Professional Development Workshops
for Teachers
+
+
Curriculum Development

Figure 1. Code.org Professional Learning Program Logic Model

1.1. Goals

The primary goal of the Code.org Professional Learning Program is to support implementation of the Code.org CS Principles Curriculum in schools such that it leads to more students, and a more diverse group of students, taking and earning qualifying scores on the AP Computer Science Principles Exam. Other student goals include generating positive attitudes, self-efficacy, sense of belonging in computer science classrooms, and positive expectation about computer science in their future. A residual outcome would be to increase the number and diversity of students who pursue computing-related opportunities after AP Computer Science Principles, such as taking more computer science classes or seeking employment that requires computer science skills.

The curriculum and associated professional development enable teachers with very little background knowledge in computer science to deliver the course via equitable teaching practices to engage all students. Other goals for teachers include positively affecting teachers' attitudes and self-efficacy toward teaching computer science, as well as their belief-systems about equity in computer science classrooms. The theory underlying these goals is that teachers who engage students with equitable teaching practices coupled with a curriculum rich with resources and activities that support and encourage enactment of those practices will lead to (1) better student learning overall (2) more equitable student engagement and learning.

1.2. Timeline & Implementation

The Code.org Professional Learning Program begins with teachers applying to the program through a Regional Partner starting with January of the year they enter the program. Regional Partners work with Code.org to approve admission to the program based on a number of criteria, the most influential being a stated commitment from the district, or teacher and school principal to offer and teach the course in the upcoming school year. It is important to note that even though the curriculum is designed to support implementation of the AP course, teachers are not required to offer it as an AP course for admission into the Professional Learning Program. In 2016-17 roughly half of the teachers in the program self-reported that they offered CS Principles as an AP course at their school.

The teacher training begins in earnest during the summer with a five-day in-person workshop in which teachers explore the Code.org curriculum and learning tools, practice and discuss classroom management and teaching strategies, and build a community of educators. Modeled after the "five requirements of transformative learning" outlined in Loucks-Horsley, Stiles, Mundry, Love, & Hewson (2010), a major focus of the professional development program is to practice new teaching strategies as part of workshop activities. In the workshop, teachers deliver lessons from the curriculum to an audience of peers that highlight these teaching practices. Afterward teachers debrief the lesson, allowing them time to reflect with peers about how implementation should be tailored for their own classrooms.

Teachers also reflect on enacting equitable teaching practices in light of the historic inequities faced by underrepresented groups in computer science. The workshops devote time to developing strategies for computer science advocacy and student recruitment strategies with a goal of enrolling students in computer science classes that are representative of their school's population in terms of race, gender, and other demographic factors.

The program continues to support teachers throughout the academic school year though workshops hosted locally by Code.org Regional Partners and run by trained Code.org Professional Development Facilitators. Each academic-year workshop combines further curriculum exploration and planning, and revising goals set during the summer (for example: recruiting and retaining a representative set of students, supporting student needs, assessing student learning, etc.). The workshops focus on elements of the curriculum that are essential for effectively teaching the course, such as exploring new computer science content, developing pedagogical strategies to keep the classroom environment equitable and engaging, and doing AP preparation.

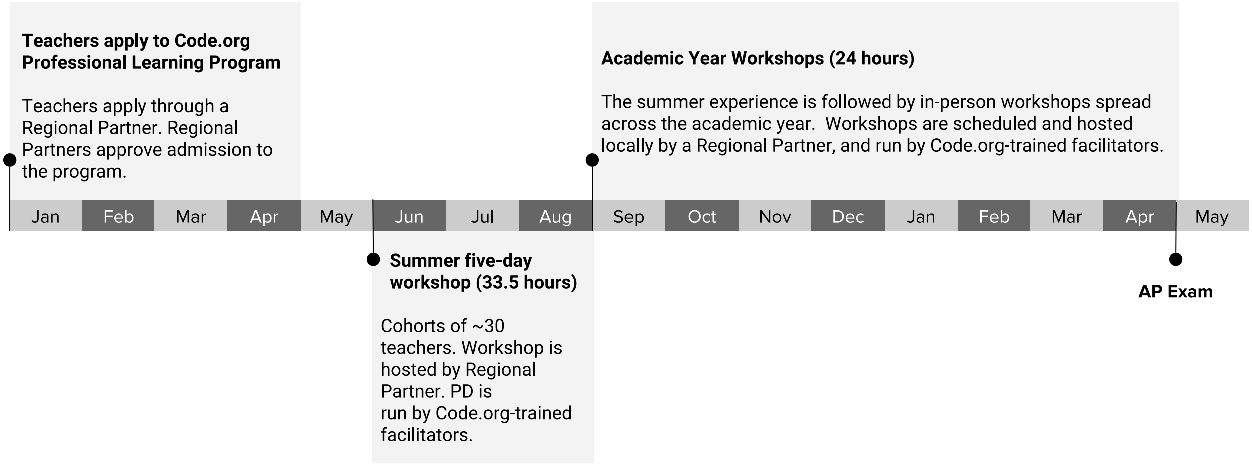

Teachers apply to Code.org Professional Learning Program
Teachers apply through a Regional Partner. Regional Partners approve admission to the program.
Academic Year Workshops (24 hours)
The summer experience is followed by in-person workshops spread across the academic year. Workshops are scheduled and hosted locally by a Regional Partner, and run by Code.org-trained facilitators.
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
Jan
Feb
Mar
Apr
May
Summer five-day workshop (33.5 hours)
Cohorts of ~30 teachers. Workshop is hosted by Regional Partner. PD is run by Code.org-trained facilitators.
AP Exam

Figure 2. Code.org Timeline of Events

2. Methodology

2.1. Data Sources

Advanced Placement test data from a total of 167 treatment schools from the most recent Code.org cohort plus 167 non-treatment schools were analyzed for this study. Data for treatment and matched comparison schools was provided by the College Board by matching the Code.org treatment schools on the state in which the school is located, total school enrollment, percent of students receiving free or reduced priced lunch, and percentage of minority students at each school. The original list of program schools included 383 schools, of which 167 were matched (43.6%). The lower than anticipated match percentage resulting from stringent matching criteria. Matching criteria required that the comparison school be located in the same state as the treatment school and be within +/- 20% of the total student enrollment of the treatment school. Further, each comparison school must also be within one standard error of the mean of the target treatment school in terms of percentage of minority students and percent of students qualifying for free or reduced priced lunch. Thus, all four criteria had to be met to identify an acceptable comparison school.

2.2. Research Design

This study employs a potential outcomes modeling approach (Rubin, 2005) to estimate the causal effect of program participation on first year improvements in AP test taking and AP qualifying score earning in computer science AP subjects. The potential outcomes model, also called the Rubin Causal Model (RCM) (Holland, 1986), allows for the formal identification for causal inference. This approach estimates the average difference between observed outcomes and potential outcomes (counterfactuals) for each unit in the analysis. This is known as the causal estimand. Potential outcomes modeling has been widely used in a number of social science fields, including education, politics, and public health to estimate causal effects of programs or policies (Glass, Goodman, Hernan, & Samet, 2013; Keele, 2015). In fact, Keele (2015) states, "The RCM is the dominant model of causality in statistics at the moment" (p. 315), while acknowledging there are many other approaches to estimating causality in a statistical framework (e.g., Dawid, 2000; Pearl, 2009).

The goal of propensity score matching within the RCM is to construct a sample of comparison schools that are similar to the treatment schools (Rosenbaum & Rubin, 1985) in terms of their likelihood of selection into treatment. This model has gained popularity in recent years and is frequently used to make causal estimates from observational studies. Rubin (2005) has argued, "the potential outcomes formulation of causal effects, whether in randomized experiments or in observational studies, has achieved widespread acceptance" (p. 329). A propensity score is a scalar value that summarizes the likelihood for a unit to receive a treatment, often based on a large set of variables. In this study, we estimate the propensity score and causal estimands using a weighting approach applied in the Toolkit for Weighting and Analysis of Nonequivalent Groups ("twang") package written in the R programming language (Ridgeway, McCaffrey, Morral, Burgette, & Griffin, 2015).

Previous literature suggests that propensity score models should include all confounding variables, that is, variables that are related to the treatment assignment as well as to the outcome (Rubin, 2007; Rubin & Thomas, 1996; West & Thoemmes, 2010), or all variables that are related to the outcome (Rosenbaum, 2002). Stuart (2010) also argues that one should be generous in including predictors in the propensity score model, because the cost of omitting a variable that might predict the outcome is greater than the cost of including a variable that in fact did not predict the outcome (increase in bias versus slight increase in standard errors of propensity scores). In this study, school demographic data such as total enrollment, percent minority enrollment, and percent of enrollment qualifying for free or reduced priced lunch provide ample information that may predict the outcomes of this study (i.e., number of students taking Computer Science AP tests and student performance on Computer Science AP tests). Thus, these three variables will be used to balance the treatment and control conditions.

2.3. Data Analytic Approach

The twang approach to propensity score estimation uses generalized boosted models (GBMs), a multivariate nonparametric regression technique, introduced in McCaffrey, Ridgeway, and Morral (2004). This approach is argued to allow for flexible, nonlinear relationships as well as a large number of variables, and shown to perform well under certain settings (see, e.g., Imai & Ratkovic, 2014). In the GBM approach, instead of matching, a weighting approach is used to estimate the treatment effect. One of the advantages of propensity score approaches is that once non-experimental data are used to "design an observational study" the study achieves balance between treatment and control groups as if it were based on an experimental study (Rubin, 2007). Then, the outcome analysis can proceed in the same way as the analysis that would have been done in an experimental study.

However, note that the effects we seek to obtain can either be the average effect of the treatment on the treated (ATT) or the average treatment effect (ATE). Generally, when we use matching strategies based on the estimated propensity scores, we estimate ATT instead of ATE, because we intentionally select and match control group schools that are like treatment schools. However, when we use weighting strategies (as is done with the twang package), depending on weights that are used, either ATT or ATE can be obtained. For this study, we estimated the effects of the program for both ATT and ATE in order to get a sense of not only what the effect of the program was the participating schools, but also what the effect would have been had the program been provided to the control schools as well.

3. Results

The first step in reviewing the results is to check on the extent to which the propensity score weighting approach results in balance across the treatment and control groups in terms of the balancing variables. As mentioned earlier, several variables were used to balance the treatment and control samples. Along with state in which the schools are located, these included: total school enrollment, percentage of students receiving free or reduced priced lunch, and percentage of total student enrollment that are minority students. These variables were chosen as they are predictive of the outcomes of interest in this study. For example, a regression model using total school enrollment, percentage of total enrollment that are minority, and percentage of total enrollment eligible for free or reduced priced lunch significantly predicted total Computer Science (Computer Science A and Computer Science Principles) tests taken at the school; *F*(3, 333) = 25.12, *p*<.001, R^2 = .19.

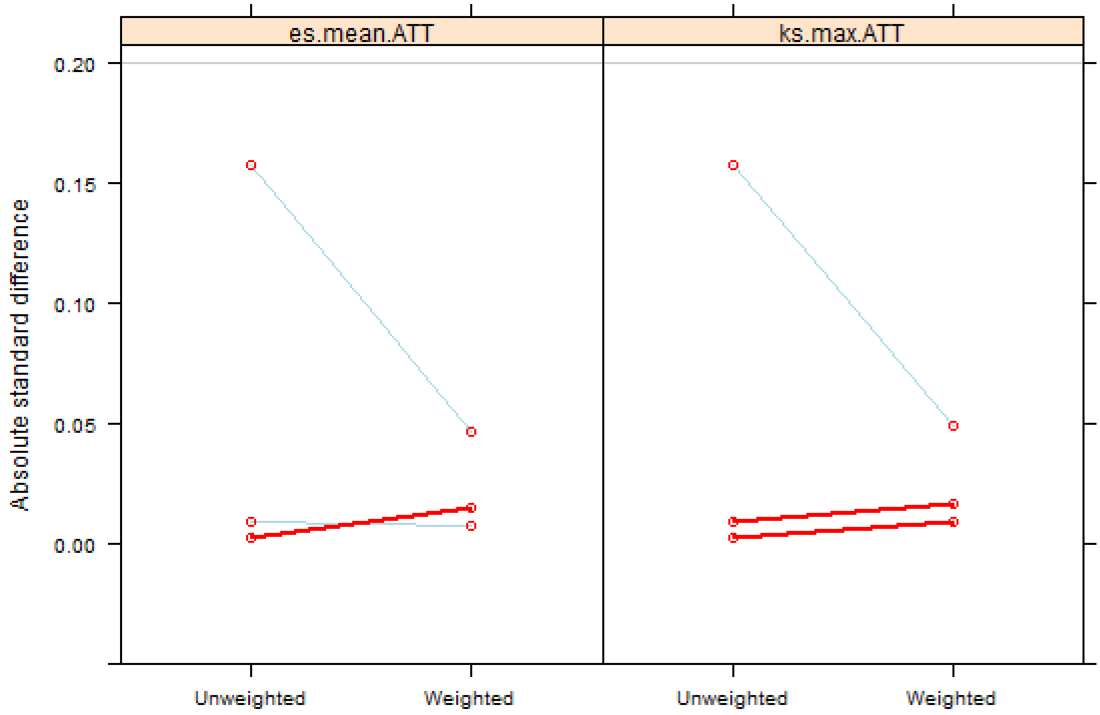


Figure 3. Balance plot for ATT analyses

Treatment and control groups were fairly balanced prior to weighting on total enrollment (*M*=1354.94 for treatment; *M*=1221.48 for controls); *t*(332)=-1.53, *p*=.128. These minor differences were virtually eliminated through weighting (see Figure 3 for balance plot for ATT analyses). No substantial differences between treatment and control schools existed in percentage of minorities (*M*=47.53% for treatment; *M*=47.42% for controls), *t*(332)=-0.03, *p*=.977 or for percent of students qualifying for free or reduced price lunch (*M*=49.56% for treatment; *M*=49.82% for controls); *t*(332)=0.09, *p*=.929. After propensity score weighting (ATT estimation), the treatment and control schools were comparable in terms of all three balancing variables. Specifically, the average total enrollment for the weighted samples was 1354.94 and 1315.26 for treatment and control respectively. Likewise, the average percent minority enrollment was balanced at 47.5 for the treatment schools and 47.0 for the control schools; and the average percent qualifying for free or reduced priced lunch was 49.6 and 49.8 for treatment and control schools, respectively. Perfect balance is not to be expected. Austin cautions, “as with randomization, one should not expect that perfect balance will be achieved for all measured baseline variables between treated and untreated subjects in the matched sample” (Austin, 2008, p. 2040).

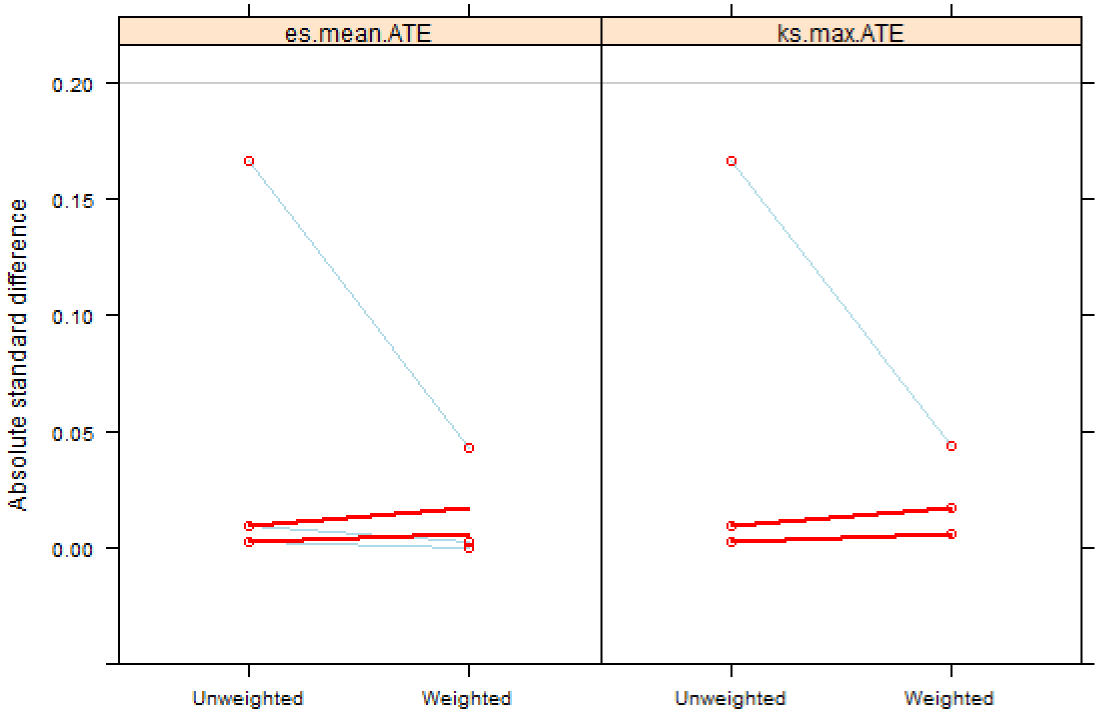


Figure 4. Balance plot for ATE analyses

Treatment and control samples were equally well balanced using the ATE propensity score estimation procedure (see Figure 4). Specifically, for the ATE estimation, the average total enrollment for the weighted samples was 1298.89 and 1263.27 for treatment and control respectively. Likewise, the average percent minority enrollment was balanced at 47.1 for the treatment schools and 47.3 for the control schools; and the average percent qualifying for free or reduced priced lunch was 49.4 and 49.9 for treatment and control schools, respectively. Given the adequately balanced samples with the ATE procedure, we will present the causal estimates from both the ATT and ATE procedures in this report.

Table 1. ATT Estimates for Test Participation by Course

	Estimate	*t* value	*p* value <	Cohen's *d*
All Computer Science				
All Students	17.96	6.72	0.001	0.735
Female Students	5.28	6.07	0.001	0.664
Black Students	1.53	4.08	0.001	0.446
Hispanic Students	5.04	4.95	0.001	0.542
Computer Science Principles				
All Students	16.27	8.03	0.001	0.879
Female Students	5.00	7.10	0.001	0.777
Black Students	1.46	4.18	0.001	0.457
Hispanic Students	4.92	5.62	0.001	0.615
Computer Science A				
All Students	1.69	1.24	**0.215 (NS)**	0.136
Female Students	0.27	0.72	**0.470 (NS)**	0.079
Black Students	0.07	1.07	**0.284 (NS)**	0.117
Hispanic Students	0.12	0.43	**0.668 (NS)**	0.047

The results of the logistic regressions for the average treatment on the treated (ATT) effect are presented in Table 1 above, which shows the impact of the program on average school Computer Science, Computer Science Principles, and Computer Science a Advanced Placement test taking. Table 2 shows the impact of the Code.org program on average number of earned qualifying scores of 3 or better on these same AP tests. Similar analyses were conducted for average treatment effects (ATE), the results of which are provided in Tables 3 and 4.

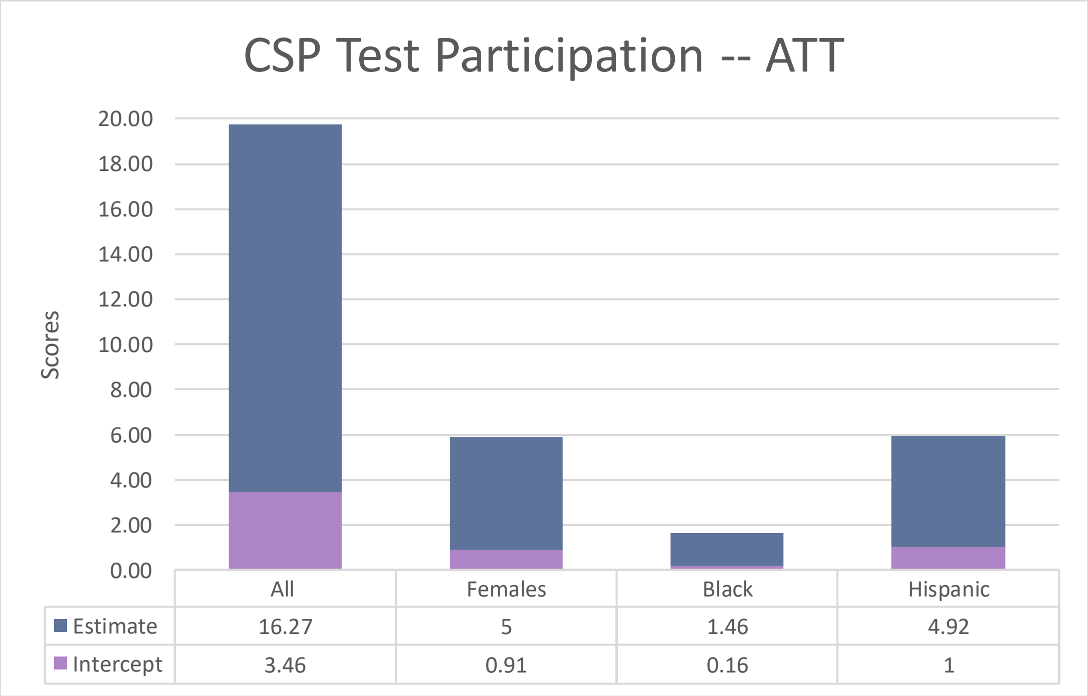


Figure 5. Effect on Computer Science Principles AP Test Participation

As indicated in Table 1, the average number of AP test taking for Computer Science Principles was dramatically higher for all students in the treatment schools following program implementation. On average, participation in the Code.org program generated an average increase of almost 18 additional AP Computer Science tests taken in the 2016-2017 school year; *t*(332) = 6.72, *p* < .001. Moreover, these effects persist when looking at student subgroups. For female students, the increase in Computer Science test taking as a result of program participation is an average of 5.28 tests per school; *t*(332) = 6.07, *p* < .001. For Black students the increase is an average of 1.53 tests; *t*(332) = 4.08, *p* < .001 and for Hispanics it is more than 5 additional tests; *t*(332) = 4.95, *p* < .001. All of the estimates are highly significant statistically, with standardized effect sizes at or above .40 (Cohen's *d*), indicating a moderate to large causal effect of the program on student AP test taking in Computer Science courses. Upon closer inspection, it is clear that virtually all of the effect on increased test participation in Computer Science courses is a function of increasing participation in Computer Science Principles and not in increased participation in Computer Science A, which is consistent with the Code.org model. In fact, there was no discernable impact of program participation on Computer Science A test taking for all students; *t*(332) = 1.24, *p*=.215, female students; *t*(332) = 0.72, *p*=.470, Black students; *t*(332) = 1.07, *p*=.284, or Hispanic students; *t*(332) = 0.43, *p*=.668. In contrast, the effect of the program on Computer Science Principles (CSP) was highly significant for all students and every student subgroup analyzed, thus the effect was not a result of generalized increases in Computer Science participation, but rather a function of targeted Computer Science Principles participation. Moreover, the Cohen's *d* effect sizes ranged from moderate (*d*=.46) to large (*d*=.88).

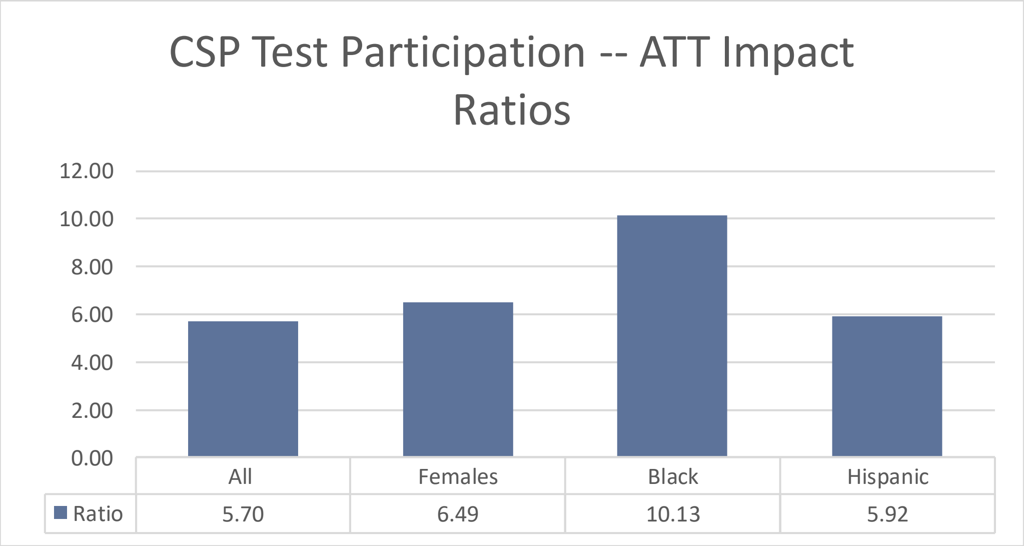


Figure 6. Impact Ratios for Student Subgroups on Computer Science Principles Test Participation

Although the standardized effect size estimates were smaller when viewing minority student test taking effects relative to effects for all students or for female students only, they are nonetheless highly significant and substantial. In fact, Figure 6 shows the impact ratios for Computer Science Principles test taking by student group. This shows that the relative impact is greatest for minority students. Whereas the program effect, in essence, increases test participation for all students by a factor of more than 5, the effect is almost twice that for Black students (10.13). That is to say, the program increased the number of Black students taking Computer Science Principles more than ten-fold on average across the treatment schools. In addition, the program increased the number of Hispanic students taking Computer Science Principles tests nearly six-fold.

Table 2. ATT Estimates for Qualifying Scores earned by Course

	Estimate	*t* value	*p* value <	Cohen's *d*
All Computer Science				
All Students	11.77	5.92	0.001	0.648
Female Students	3.01	5.31	0.001	0.581
Black Students	0.43	4.02	0.001	0.440
Hispanic Students	2.24	4.96	0.001	0.543
Computer Science Principles				
All Students	10.41	6.73	0.001	0.736
Female Students	2.68	5.91	0.001	0.647
Black Students	0.40	4.14	0.001	0.453
Hispanic Students	2.25	5.37	0.001	0.588
Computer Science A				
All Students	1.36	1.35	**0.179 (NS)**	0.148
Female Students	0.39	1.18	**0.239 (NS)**	0.129
Black Students	0.03	1.03	**0.305 (NS)**	0.113
Hispanic Students	-0.01	-0.05	**0.961 (NS)**	-0.005

Similarly impressive results were found for program effects on the number of qualifying scores earned in program schools. In addition to increasing the number of students taking Computer Science AP tests, the Code.org program increased the number of qualifying scores earned by students in Computer Science AP courses. Table 2 demonstrates that program schools reported an average of 11.77 more qualifying scores in all Computer Science courses ($t(332) = 5.92$, $p < .001$) and an average of 10.41 more qualifying scores of Computer Science Principles for all students ($t(332) = 6.73$, $p < .001$), both of which were highly statistically significant. Further, as with test taking effects, the impact on qualifying scores was persistent for each student subgroup, with moderate to large effect sizes demonstrated for Computer Science Principles and no discernable effect on the number of qualifying scores earned in Computer Science A.

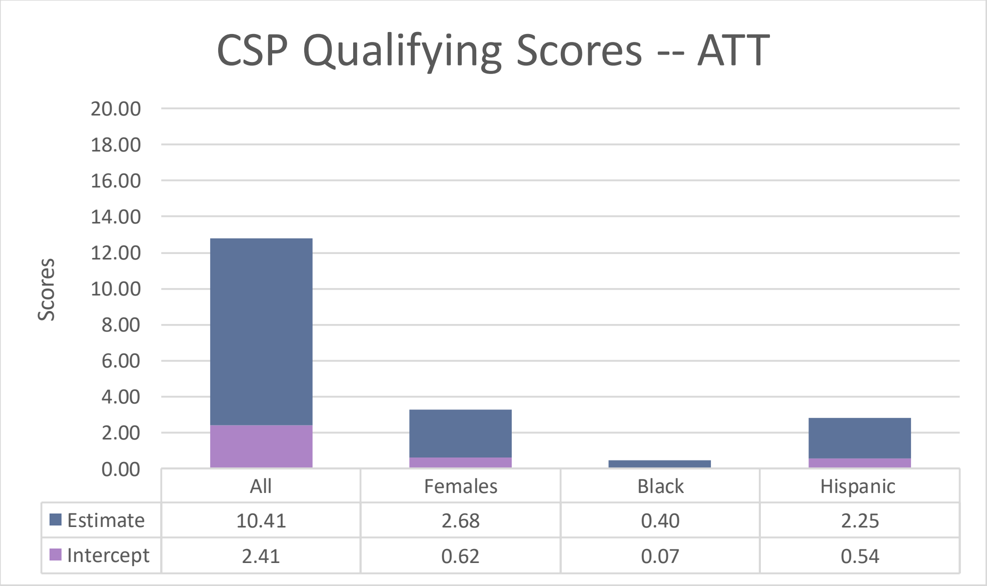
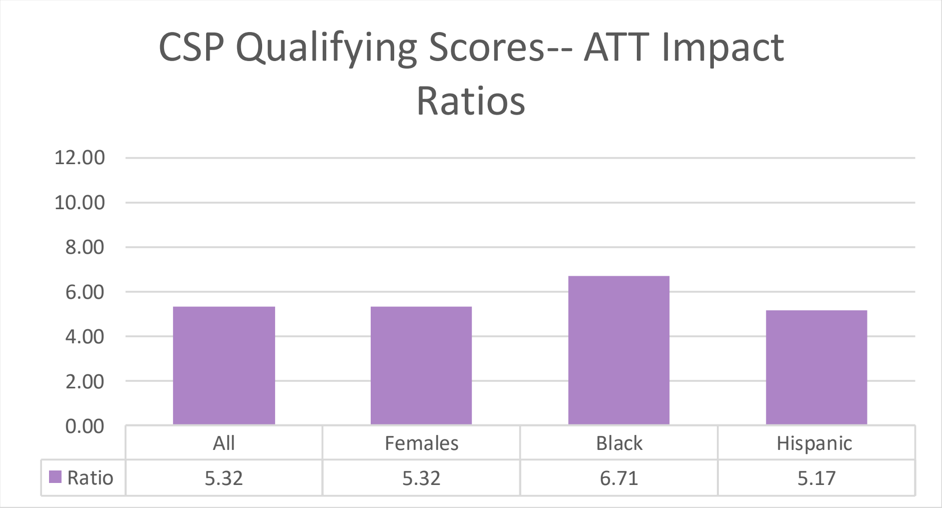


Figure 7. Effect of program on Computer Science Principles qualifying scores earned

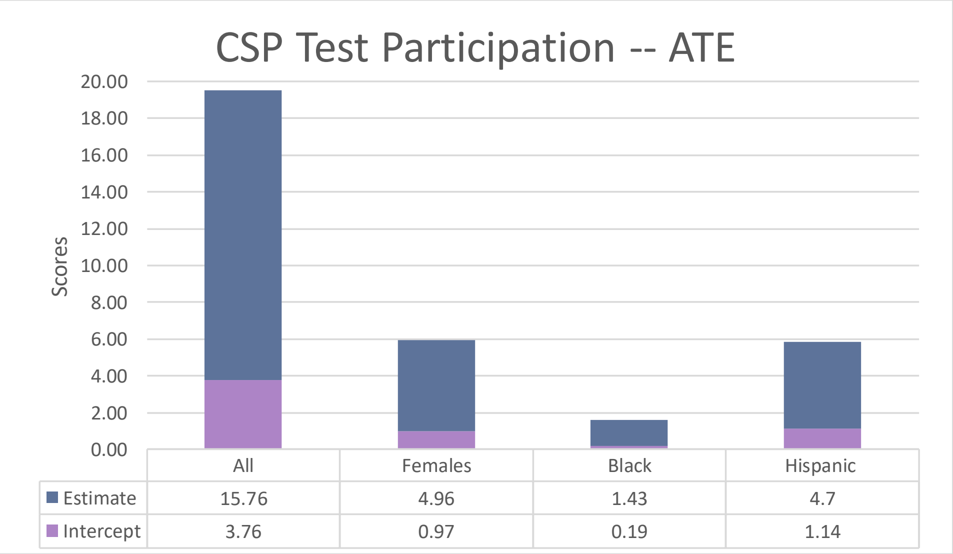
Figure 7 shows the impact of participation in the Code.org program on qualifying scores earned in Computer Science Principles in the treatment schools by student subgroup relative to what would have been expected had the program not been implemented in the treatment schools. On average, the program resulted in 2.68 more qualifying scores for female students; *t*(332) = 5.91, *p* < .001, 0.40 more qualifying scores for Black students; t(332) = 4.14, p < .001, and 2.25 more qualifying scores per school for Hispanic students; *t*(332) = 5.37, *p* < .001. Although these values are smaller compared to the effect for all students, they are nonetheless highly significant substantial effects. The effect sizes for these groups are all in the moderate range (*d*=.45 to *d*=.65).



	All	Females	Black	Hispanic
■ Ratio	5.32	5.32	6.71	5.17

Figure 8. Impact Ratios for Student Subgroups on Computer Science Principles Qualifying Scores

Further, the impact ratios for at least one minority subgroup are greater than for non-minority students. As Figure 8 shows, whereas the program results in a more than five-fold increase in the number of qualifying scores in Computer Science Principles for all students, Black students saw an increase of more than 6.7 times what would have happened without participation in the Code.org program.



	All	Females	Black	Hispanic
■ Estimate	15.76	4.96	1.43	4.7
■ Intercept	3.76	0.97	0.19	1.14

Figure 9. Average Treatment Effect on Computer Science Principles AP Test Participation

These average treatment on the treated (ATT) estimates show that program participation substantially increased the number of Advanced Placement Computer Science Principles tests taken and qualifying scores earned for students in the treatment schools. In addition to these estimates, we estimated the average treatment effect (ATE), which is the expected average effect of the program if it had been presented to the control schools as well. The results of these analyses regarding test participation are presented in Table 3. Consistent with program expectations, program implementation in the full sample would significantly improve Computer Science Principles participation for all students and all student subgroups, but would not impact test participation in Computer Science A for any group. On average, program implementation in all schools in the sample would have resulted in an additional 15.76 Computer Science Principles tests; $t(332) = 7.42$, $p < .001$, an additional 4.96 tests among female students; $t(332) = 6.70$, $p < .001$, an additional 1.43 tests for Black students; $t(332) = 4.03$, $p < .001$, and an additional 4.7 tests for Hispanic students; $t(332) = 5.03$, $p < .001$ (see Figure 9).

Table 3. ATE Estimates for Test Participation by Course

	Estimate	*t* value	*p* value <	Cohen's *d*
All Computer Science				
All Students	17.07	6.57	0.001	0.719
Female Students	5.19	6.00	0.001	0.657
Black Students	1.47	3.85	0.001	0.421
Hispanic Students	4.71	4.33	0.001	0.474
Computer Science Principles				
All Students	15.76	7.42	0.001	0.812
Female Students	4.96	6.70	0.001	0.733
Black Students	1.43	4.03	0.001	0.441
Hispanic Students	4.70	5.03	0.001	0.550
Computer Science A				
All Students	1.31	1.09	**0.275 (NS)**	0.119
Female Students	0.22	0.69	**0.494 (NS)**	0.076
Black Students	0.04	0.59	**0.553 (NS)**	0.065
Hispanic Students	0.01	0.05	**0.961 (NS)**	0.005

As was seen with the ATT estimates, the average treatment effect estimates produced a much greater impact ratio for Black student Computer Science Principles test participation than for the overall collection of students or for Female or Hispanic students. Figure 10 shows that for the full student population, the treatment increased Computer Science Principles test participation more than 500% for all students and for Hispanic students in particular, but the increase for Female students exceeded 600% and for Black students test participation increased more than 800% greater than would be observed without program participation.

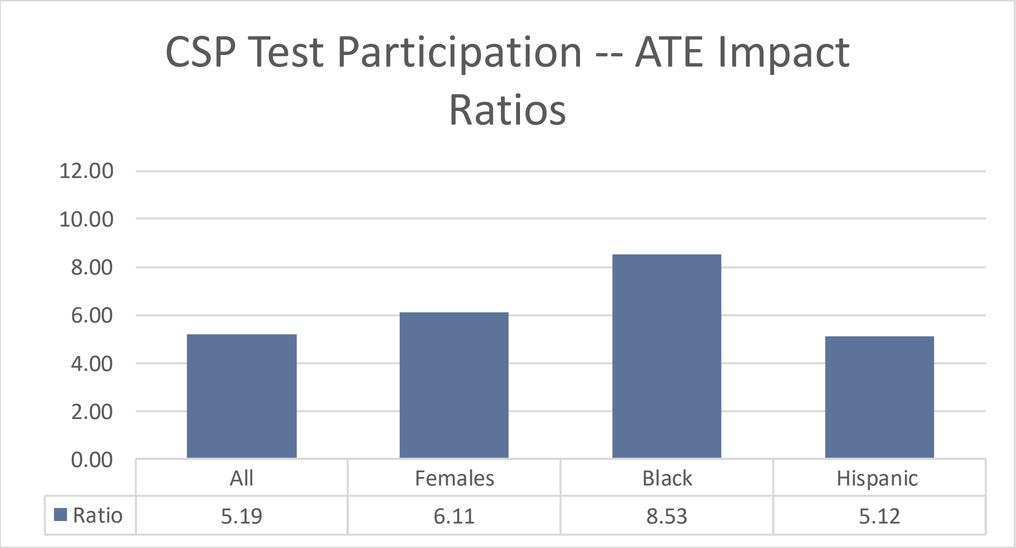


Figure 10. Impact Ratios for Student Subgroups on Computer Science Principles Test Participation.

A comparable pattern of findings was observed for Computer Science Principles qualifying scores using the average treatment effect estimates as was found with test participation using the same ATE estimand (see Table 4). Program participation would increase the average number of qualifying scores by more than 10 per school in the overall sample; $t(332) = 6.05$, $p < .001$, by an average of 2.69 for female students; $t(332) = 5.46$, $p < .001$, by an average of .39 for Black students; $t(332) = 3.83$, $p < .001$, and by an average of more than 2 qualifying scores for Hispanic students; $t(332) = 4.52$, $p < .001$ (see Figure 11). Each of these projected improvements are highly statistically significant. As with the ATT estimates, no significant improvement in Computer Science A qualifying scores is anticipated by program participation.

The impact ratios using the ATE approach, while still substantial, are lower than for the ATT estimation procedure (Figure 12). For all students, the number of qualifying scores is projected to be 4.91 times larger with the ATE approach as compared with 5.32 times larger with the ATT approach. Likewise, the ratio for females is 5.20 for ATE versus 5.32 for ATT. For minority students, the ratios are considerably lower with the average treatment effect approach compared to the average treatment on the treated approach (5.88 vs. 6.71 for Black students; 4.35 vs. 5.17 for Hispanic students). Notwithstanding these discrepancies in estimation procedures, the program effects on the number of Computer Science Principles qualifying scores remain large and significant.

Table 4. ATE Estimates for Qualifying Scores earned by Course

	Estimate	*t* value	*p* value <	Cohen’s *d*
All Computer Science				
All Students	11.14	5.57	0.001	0.610
Female Students	2.97	5.18	0.001	0.567
Black Students	0.41	3.68	0.001	0.403
Hispanic Students	2.05	4.22	0.001	0.462
Computer Science Principles				
All Students	10.08	6.05	0.001	0.662
Female Students	2.69	5.46	0.001	0.598
Black Students	0.39	3.83	0.001	0.419
Hispanic Students	2.08	4.52	0.001	0.495
Computer Science A				
All Students	1.05	1.15	**0.251 (NS)**	0.126
Female Students	0.28	1.19	**0.237 (NS)**	0.130
Black Students	0.02	0.76	**0.448 (NS)**	0.083
Hispanic Students	-0.03	-0.27	**0.785 (NS)**	-0.030

In sum, the results of this study indicate substantial and significant increases in both AP test taking and qualifying score earning for all students following the implementation of the Code.org professional development program. In addition, significant program effects for Computer Science Principles AP test taking and qualifying score earning were found for female students and minority students when analyzed separately. Average effect sizes (Cohen’s *d*) for treatment effects over both average treatment on treated (ATT) and average treatment effects for all students (ATE), all subgroups of students, and both outcomes, and all disciplines was *d*=.62, showing a substantial positive causal impact. The effects are stronger when looking only at the average treatment on the treated (ATT) effects, where the average effect size for first year effects was *d*=.64 across all subsamples and outcomes analyzed. The mean effect size for all analyses with the ATE approach was slightly smaller at *d*=.59, which still indicates a moderate effect size.

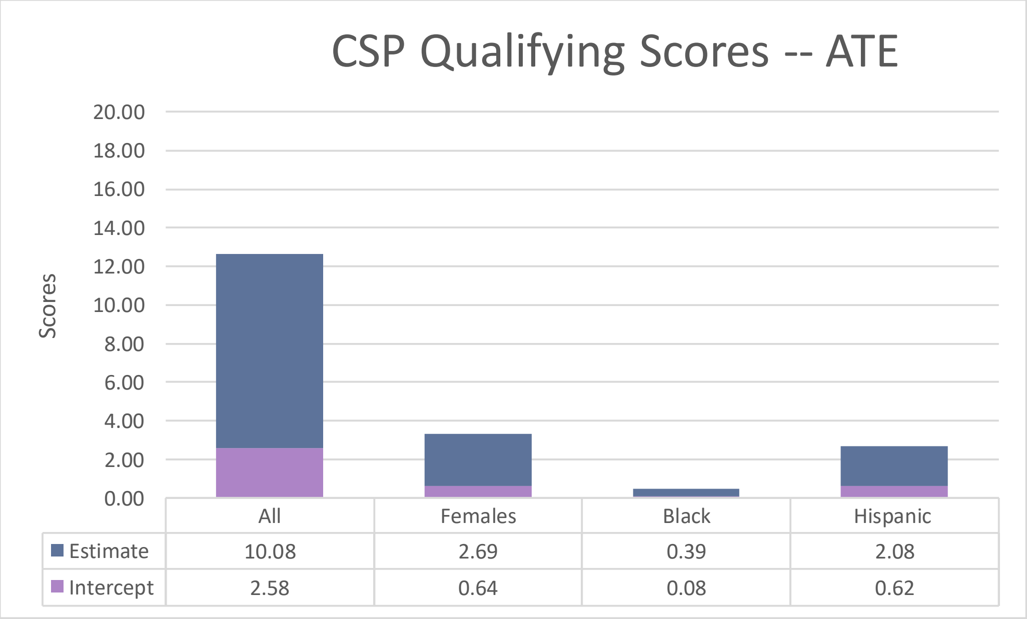


Figure 11. Average Treatment Effect of program on Computer Science Principles qualifying scores earned

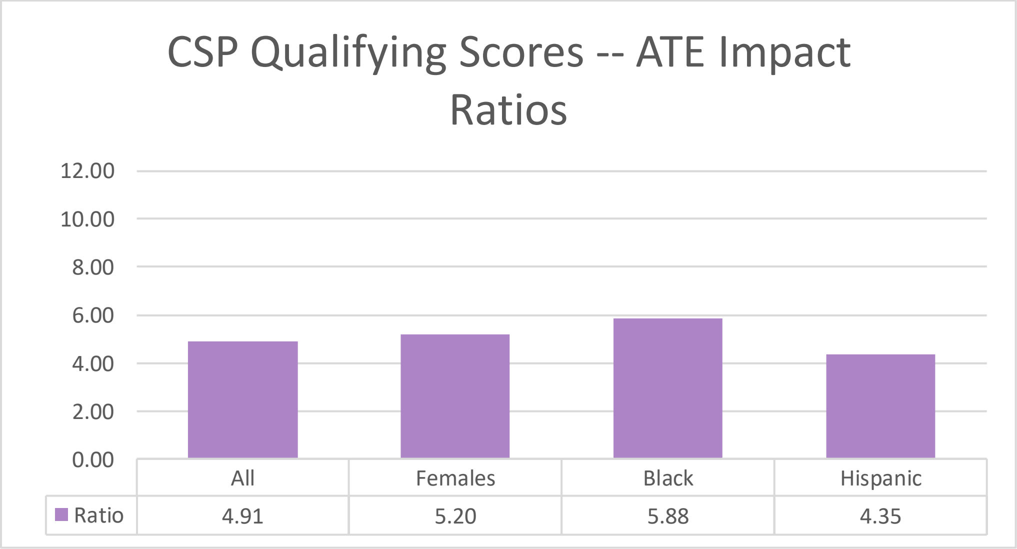


Figure 12. Impact Ratios for Student Subgroups on Computer Science Principles Qualifying Scores

4. Discussion

This study provides evidence that the Code.org teacher preparation program increases the number of AP tests taken and the number of AP qualifying scores earned by the students of the participating teachers. This is consistent with prior research that has shown that teacher professional development can, in certain contexts, positively impact student outcomes generally (Yoon, K. S., Duncan, T., Lee, S. W.-Y., Scarloss, B., & Shapley, K., 2007) and in computer science specifically (Mouza, C., Marzocchi, A, Pan, Y., & Pollock, L., 2016). In and of itself, these results are important, but these increases may lead to additional advantages for these students. Research shows that students who take AP courses have a greater likelihood of attending college (Mattern, Marini, & Shaw, 2013). Mattern, et. al state, "… the odds of enrolling in a four-year institution increased by 171% for students who took one AP Exam compared with students who took no AP exams. The increase in odds was even higher for students who took more than one AP exam" (Mattern, Marini, & Shaw, 2013, p. 5). Students participating in AP classes also earn better grades in college (Shaw, Marini, & Mattern, 2013), and have a greater likelihood of persisting in and graduating from college (Dougherty, Mellor, & Jian, 2006; Hargrove, Godin, & Dodd, 2008). In addition, students who earn qualifying scores on AP tests outperform matched Non-AP students on many college outcome measures (Murphy & Dodd, 2009). Future research should explore these longer term potential impacts of this training program.

This work is significant for many reasons. First, it demonstrates the use of propensity score potential outcomes modeling to observational data to yield meaningful and significant causal estimates of a popular professional development program's effectiveness in a context where randomized assignment to treatment condition is either infeasible or impractical. Secondly, this study provides evidence that Code.org's Professional Development Program for CS Principles is having significant and important impacts on preparing more students to succeed in Computer Science careers and improving the future of Computer Science education in this country. More students, notably female and minority students, are engaging in, and succeeding in, Computer Science Principles as a result of implementing this program in schools across the country. From an impact ratio perspective, the program is having a greater impact for these groups of students.

Acknowledgments

Source: Derived from data provided by the College Board. Copyright © 2017 The College Board. www.collegeboard.org.

References

Austin, P. C. (2008). A critical appraisal of propensity-score matching in the medical literature between 1996 and 2003. Statistics in Medicine, 27, 2037-2049.

Beede, D. N., Julian, T. A., Langdon, D., McKittrick, G., Khan, B., and Doms, M. E., Women in STEM: A Gender Gap to Innovation (August 1, 2011). Economics and Statistics Administration Issue Brief No. 04-11. Available at SSRN: https://ssrn.com/abstract=1964782 or http://dx.doi.org/10.2139/ssrn.1964782

Dawid, A. P. (2000). Causal inference without counterfactuals. Journal of the American Statistical Association, 95(450), 407-424.

Dougherty, C., Mellor, L. & Jian, S. (2006). The relationship between Advanced Placement and college graduation. (National Center for Educational Accountability: 2005 AP Study Series, Report 1). Austin, TX: National Center for Educational Accountability.

Glass, T.A., Goodman, S.N., Hernan, M.A., & Samet, J.M. (2013). Causal inference in public health. Annual Review of Public Health, 34, 61-75.

Goode, J. (2007). If You Build Teachers, Will Students Come? The Role of Teachers in Broadening Computer Science Learning for Urban Youth. Journal of Educational Computing Research, 36(1), 65-88.

Hargrove, L., Godin, D., Dodd, B. (2008). College outcomes comparisons by AP and non-AP high school experiences (College Board Research Report 2008-3). New York: The College Board.

Holland, P. (1986). Statistics and causal inference. Journal of the American Statistical Association, 81(396), 945-960.

Imai, K., & Ratkovic, M. (2014). Covariate balancing propensity score. Journal of the Royal Statistical Society: Series B. 53, 597-610.

Keele, L. (2015). The statistics of causal inference: A view from political methodology. Political Analysis, 23, 313-335.

Louckes-Horsely, S., Stiles, K.E., Mundry, S., Love, N., Hewson, P.W. (2010). Designing Professional Development for Teachers of Science and Mathematics (3rd edition). 69-70.

Mattern, K.D., Marini, J.P., & Shaw, E.J. (2013). The relationship between AP Exam performance and college outcomes. (College Board Research Report 2009-4) New York: The College Board.

Mattern, K.D., Shaw, E.J., & Xiong, X. (2009). Are AP students more likely to graduate from college on time? (College Board Research Report 2013-5) New York: The College Board.

McCaffrey, D. F., Ridgeway, G., & Morral, A. R. (2004). Propensity score estimation with boosted regression for evaluating causal effects in observational studies. Psychological Methods, 9, 403-425.

Morgan, R., & Klaric, J. (2007). AP students in college: An analysis of five-year academic careers (College Board Research Report No. 2007-04). New York: The College Board.

Mouza, C., Marzocchi, A, Pan, Y., & Pollock, L. (2016). Development, Implementation, and Outcomes of an Equitable Computer Science After-School Program: Findings from Middle-School Students. Journal of Research on Technology in Education. 48. 1-21. 10.1080/15391523.2016.1146561.

Murphy, D., & Dodd, B. (2009). A comparison of college performance of matched AP and non-AP student groups. (College Board Research Report No. 2009-6). New York: The College Board.

Pearl, J. (2009). Causality: models, reasoning, and inference. 2nd Edition. New York: Cambridge University Press.

Ridgeway, G., McCaffrey, D., Morral, A., Burgette, L, & Griffin, B. (2015). “twang: Toolkit for weighting and analysis of nonequivalent groups.” Available at http://cran.r-project.org/web/packages/twang/twang.pdf.

Rosenbaum, P. R. (2002). Observational Studies, 2nd ed. Springer, New York.

Rosenbaum, P.R., & Rubin, D. B. (1985). Constructing a control group using multivariate matched sampling methods that incorporate the propensity score. The American Statistician, 39, 33-38.

Rubin, D. B. (2005). Causal inference using potential outcomes: Design, modeling, decisions. Journal of the American Statistical Association, 100(469), 322-331.

Rubin, D. B. (2007). The design versus the analysis of observational studies for causal effects: Parallels with the design of randomized trials. Statistical Medicine, 26, 20-36.

Rubin, D. B., & Thomas, N. (1996). Matching using estimated propensity scores, relating theory to practice. Biometrics, 52, 249-264.

Sax, L. J., Lehman, K. J., Jacobs, J. A., Kanny, M. A., Lim, G., Monje-Paulson, L., & Zimmerman, H. B. (2017). Anatomy of an Enduring Gender Gap: The Evolution of Women's Participation in Computer Science, The Journal of Higher Education, 88:2, 258-293, DOI: 10.1080/00221546.2016.1257306

Shaw, E. J., Marini, J. P., & Mattern, K.D. (2013). Exploring the utility of Advanced Placement participation and performance in college admission decisions. Educational and Psychological Measurement, 73, 229-253.

Stuart, E. A. (2010). Matching methods for causal inference: a review and a look forward. Statistical Science, 25(1), 1-21.

West, S. G., & Thoemmes, F. (2010). Campbell's and Rubin's perspectives on causal inference. Psychological Methods, 15(1), 18-37.

Yoon, K. S., Duncan, T., Lee, S. W.-Y., Scarloss, B., & Shapley, K. (2007). Reviewing the evidence on how teacher professional development affects student achievement (Issues & Answers Report, REL 2007–No. 033). Washington, DC: U.S. Department of Education, Institute of Education Sciences, National Center for Education Evaluation and Regional Assistance, Regional Educational Laboratory Southwest. Retrieved from http://ies.ed.gov/ncee/edlabs