A Study of Stereotype Threat in Computer Science

Amruth N. Kumar
Ramapo College of New Jersey
505 Ramapo Valley Road
Mahwah, NJ 07430, USA
1 201 684 7712
amruth@ramapo.edu

ABSTRACT
A controlled study was conducted to detect stereotype threat on harder topics in introductory Computer Science. Students in the control group were asked to identify their demographic information before taking the test whereas students in the experimental group were asked to do so after completing the test. So, the control group was indirectly reminded of stereotypes before taking the test, when it could affect performance on the test, whereas the experimental group was reminded after the test when it could not affect test performance.

Tests on two different loops were used, along with a partial cross-over design: a random group of students served as the control group on one test, and the experimental group on the other and vice versa. Mixed factor ANOVA analysis of the data showed that all the students scored significantly higher when not reminded of stereotypes before the test regardless of sex or race. In addition, average/below-average students scored significantly higher when not reminded of stereotypes before the test, again, regardless of sex or race. So, on harder topics in Computer Science, stereotype threat affects all the students, and in particular, the less-prepared students. In light of this, some suggestions are offered for avoiding stereotype threat during tests.

Categories and Subject Descriptors
K.3.2 [Computing Milieux]: Computer and Information Science Education

General Terms
Experimentation, Measurement, Human Factors.

Keywords
Stereotype Threat, Online Testing, Computer Science I.

1. INTRODUCTION
When sex and race are associated with positive or negative stereotypes in a discipline, reminding students of the stereotype before a test can positively or negatively affect their performance. The effect has been termed stereotype threat (e.g., [14, 15]). Reminding students of stereotypes could happen explicitly or implicitly. Subtle gestures such as asking students for their race and sex, or holding the test in a classroom that contains “masculine” wall posters [6] are sufficient to elicit stereotype threat. Researchers have studied stereotype threat with women and minorities in mathematics (e.g., [13, 11]), and engineering (e.g., [3]). In Computer Science, stereotype threat is listed as one of the factors that could be contributing to problems with recruitment and retention of female and minority students (e.g., [4, 10]). Interventions have been tried to reduce students’ susceptibility to stereotype threat [1], but have yielded mixed results in Computer Science [12].

A study was conducted earlier to see if stereotype threat could be detected in the context of testing in Computer Science [9]. In the study, stereotype threat was studied among introductory Computer Science students using online tests on logical and assignment expressions. Since stereotype threat was observed on assignment expression test, but not logical expression test, and students found assignment expressions to be harder, the study concluded that stereotype threat may be more readily apparent on harder topics in Computer Science. This result concurred with findings reported in literature for learning Math [13]. Another finding of the earlier study was that stereotype threat was significant on assignment expressions only when students who had scored 100% on the test had been eliminated from consideration. In other words, stereotype threat was more evident among students who had not already mastered the material.

The current study was conducted as a follow-up to see under what conditions stereotype threat could be detected on harder topics in Computer Science. The study was conducted with online tests on loops, which are harder than expression evaluation. In the study, stereotype threat was elicited by simply asking the students to identify their demographic information – sex, race, academic status and major. No other reference was made to positive or negative stereotypes during the test. Software tutors called problets (www.problets.org) were used to conduct the study entirely online.

The study was controlled – in the beginning, all the students entered their name and email address. At this time, the students in the control group were also asked for their demographic information, whereas the students in the experimental group were asked for it on a separate questionnaire presented after the test. So, the students in the control group were reminded of stereotypes
before taking the test, which could affect their test performance, unlike the students in the experimental group. If stereotype threat was indeed detectable, a significant difference in test scores could be expected between control and experimental students. According to stereotype threat, the difference would apply in particular to female and minority students.

2. METHODS

Two tests were administered during the study in spring 2010 – one on while loops and another on for loops. The students answered the while loop test (usually 1-7 days) before the for loop test. During the tests, each limited to 30 minutes, the students solved 9-10 problems on program tracing. The tests and the demographic surveys were administered back to back, online. Computer Science I students participated in the study asynchronously, on their own time after class.

Participants were randomly assigned to one of two groups: A or B. Partial cross-over design was used: group A served as the experimental group on while loop and the control group on for loop test, and vice versa for group B. For analysis purposes, only the students who had answered both the tests were considered. As informed by the earlier study [9], students who had scored 100% on both the tests (N=13) were not considered.

A total of 193 students were included in the study – 129 male, 56 female, 117 Caucasians, and 54 non-Caucasians (including Black/African American, Hispanic/Latino, Asian and Other). Some students did not identify their sex/race and were not included in the corresponding analysis. The students had solved from 7 to 10 problems on the tests. In order to factor out the effect of the differences in the number of problems solved, the average score per problem with a range of 0 → 1.0 was considered for analysis rather than the raw score with a range of 0 → 10. Similarly, the average time spent per problem was considered rather than the total time spent on the test.

3. ANALYSIS AND RESULTS

Unless otherwise noted, 2 X 2 mixed factor ANOVA was used to analyze data in each of the following sections. In each case, the repeated measure and the between-subjects factor are listed. In the analysis, treatment refers to asking for demographic information before versus after the test.

3.1 Treatment versus group

Average score per problem was analyzed, with the treatment as the repeated measure for each student and the group (A versus B) as the between-subjects factor.

A significant main effect was observed for treatment [F(1,191) = 13.910, p < 0.001] - students scored:

- 0.8738 ± 0.02 (at 95% confidence level) when in the control group;
- 0.9105 ± 0.01 when in the experimental group.

The difference was statistically significant [t(192) = -3.950, p < 0.001]. The effect size (Cohen’s d) is 0.31, indicating that the effect of treatment is small, but not negligible. In other words, students scored higher when reminded of stereotypes after rather than before the test. Note that this result applies to the general population, and not just women and minorities.

No significant main effect was observed for group - there was no statistical difference between groups A and B. The two groups of students were comparable, and it did not matter whether the students were in the control group first or in the experimental group first.

A significant interaction was observed between treatment and group [F(1,191) = 4.608, p = 0.033]: Group A scored more on the while test as the experimental group than on the for test as the control group, as shown in Table 1. But the difference between the two scores was not statistically significant. Group B scored less on the while test as the control group than on the for test as the experimental group. The difference between the scores was statistically significant [t(1,106) = -4.642, p < 0.001]. This eliminates the topic of the test (while versus for loops) as a confounding factor.

<table>
<thead>
<tr>
<th>Group (N)</th>
<th>while test</th>
<th>for test</th>
</tr>
</thead>
<tbody>
<tr>
<td>A (86)</td>
<td>0.900 ± 0.022</td>
<td>0.885 ± 0.028</td>
</tr>
<tr>
<td>B (107)</td>
<td>0.865 ± 0.025</td>
<td>0.919 ± 0.020</td>
</tr>
</tbody>
</table>

Average time spent per problem was analyzed, with treatment as the repeated measure and group (A versus B) as the between-subjects factor. A significant main effect was found for treatment [F(1,191) = 10.830, p = 0.001] – students solved problems in:

- 91.739 ± 4.985 seconds per problem in experimental condition;
- 101.165 ± 5.978 seconds in control condition.

The difference was statistically significant [t(192) = 2.940, p = 0.004]. The effect size (Cohen’s d) is 3.39, indicating that the effect of treatment is huge. In other words, students answered the test faster when reminded of stereotypes after rather than before the test.

<table>
<thead>
<tr>
<th>Group (N)</th>
<th>Control group</th>
<th>Experimental group</th>
</tr>
</thead>
<tbody>
<tr>
<td>A (86)</td>
<td>104.992 ± 8.902</td>
<td>87.322 ± 7.424</td>
</tr>
<tr>
<td>B (107)</td>
<td>97.337 ± 7.981</td>
<td>96.156 ± 6.656</td>
</tr>
</tbody>
</table>

No significant main effect was found for group - there was no statistical difference between groups A and B. A significant interaction was found between treatment and group [F(1,191) = 8.285, p = 0.004]:

- Group B spent about the same amount of time per problem on the while loop test as the control group as it did on the for loop test as the experimental group, and any difference between the two was not statistically significant.
- Group A spent more time on the for loop test as the control group than it did on the while loop test as the experimental group. The difference was statistically significant [t(192) = 2.940, p = 0.004].
group as shown in Table 2, and the difference was statistically significant \( t(85) = 3.554, p = 0.001 \).

This eliminates the topic of the test as a confounding factor, while attributing faster testing times to the treatment.

### 3.2 Treatment versus sex

In Computer Science, stereotype threat is expected to negatively affect female students in particular. So, the average score per problem was analyzed, with treatment as the repeated measure and sex as the between-subjects factor.

A marginally significant main effect was observed for sex \( F(1,183) = 3.242, p = 0.073 \): female students scored less than male students in both control and experimental groups as shown in Table 3. The difference between the sexes was marginally significant for the control group \( t(183) = 1.858, p = 0.065 \) and not statistically significant for the experimental group.

<table>
<thead>
<tr>
<th>Sex</th>
<th>Control group</th>
<th>Experimental group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male</td>
<td>0.886 ± 0.023</td>
<td>0.915 ± 0.018</td>
</tr>
<tr>
<td>Female</td>
<td>0.847 ± 0.035</td>
<td>0.897 ± 0.028</td>
</tr>
</tbody>
</table>

The interaction between treatment and sex was not statistically significant since both male and female students scored higher when asked for demographics information after rather than before the test as shown in Table 3. The difference between control and experimental scores was statistically significant for both the sexes:

- Male students - \( t(128) = -2.610, p = 0.010 \);
- Female students - \( t(55) = -2.747, p = 0.008 \).

So, the data did not support any greater influence of stereotype threat on female students than on male students.

Analysis of the average time per problem yielded a significant main effect for sex \( F(1,183) = 14.450, p < 0.001 \) – female students spent more time per problem than male students in control as well as experimental condition as shown in Table 4. The difference between the sexes was statistically significant in both the conditions:

- Control condition - \( t(183) = -2.184, p = 0.030 \);
- Experimental condition - \( t(183) = -4.560, p < 0.001 \).

The difference between control and experimental times was statistically significant for male \( t(128) = 3.164, p = 0.002 \), but not female students. So, the data supported a significant influence of treatment on the speed with which male students answered the test.

<table>
<thead>
<tr>
<th>Sex</th>
<th>Control group</th>
<th>Experimental group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male</td>
<td>96.121 ± 7.200</td>
<td>85.232 ± 5.839</td>
</tr>
<tr>
<td>Female</td>
<td>110.606 ± 10.928</td>
<td>109.761 ± 8.862</td>
</tr>
</tbody>
</table>

### 3.3 Treatment versus race

Stereotype threat is also expected to negatively affect minority students in Computer Science. So, the average score per problem was analyzed, with treatment as the repeated measure and race as the between-subjects factor.

No significant main effect was found for race: non-Caucasians scored the same as Caucasians. The interaction between treatment and race was not statistically significant: both Caucasians and non-Caucasians scored higher when asked for demographic information after rather than before the test, as shown in Table 5. The improvement in score was statistically significant for both the groups:

- Caucasians - \( t(116) = -2.313, p = 0.022 \);
- Non-Caucasians - \( t(53) = -2.979, p = 0.004 \).

Asians are positively stereotyped in quantitative domains such as Computer Science (e.g., [7]). So, the analysis was repeated after combining Caucasians and Asians into one group and the rest into the other group. The results remained the same: no main effect for race since both the groups had similar scores; and no significant interaction between treatment and race since both the groups scored higher as experimental subjects than as control subjects. In other words, the data did not support any greater influence of stereotype threat on minority students.

<table>
<thead>
<tr>
<th>Race</th>
<th>Control group</th>
<th>Experimental group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Caucasians</td>
<td>0.878 ± 0.024</td>
<td>0.907 ± 0.020</td>
</tr>
<tr>
<td>Non-Caucasians</td>
<td>0.871 ± 0.035</td>
<td>0.913 ± 0.029</td>
</tr>
</tbody>
</table>

A similar analysis of the time spent per problem yielded a significant main effect for race \( F(1,169) = 9.336, p = 0.003 \) – Caucasians spent less time per problem than non-Caucasians in both treatments, as shown in Table 6. The difference between the races was statistically significant in both the conditions:

- Control condition - \( t(169) = -2.543, p = 0.012 \);
- Experimental condition - \( t(169) = -2.649, p = 0.009 \).

The decrease in time from the control to the experimental condition was significant for Caucasians \( t(116) = 2.097, p = 0.038 \), but not for non-Caucasians. So, the data supported a significant influence of treatment on the speed with which Caucasian students answered the test.

<table>
<thead>
<tr>
<th>Race</th>
<th>Control group</th>
<th>Experimental group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Caucasians</td>
<td>94.717 ± 7.482</td>
<td>87.779 ± 6.265</td>
</tr>
<tr>
<td>Non-Caucasians</td>
<td>111.870 ± 11.013</td>
<td>102.741 ± 9.222</td>
</tr>
</tbody>
</table>

The pattern that emerges from these analyses is that students scored higher when reminded of stereotypes after rather than before the test. This was true and the increase in score was statistically significant regardless of the sex or race. The data did not support that stereotype threat affected the score of students in any one particular demographic group more than the other. On
the other hand, being reminded of stereotypes after rather than before the test helped male and Caucasian students answer the test significantly faster. It might seem counter-intuitive that even male and Caucasian students who are positively stereotyped in Computer Science were affected by stereotype threat, i.e., they scored lower and solved problems slower when reminded of stereotypes before the test. However, this is explained as “choking effect” in literature [2, 5] – when male/Caucasian students have not mastered a topic, if they are reminded of stereotypes before a test, they may view the positive stereotype associated with their demographic group as being externally imposed. They choke, leading to poorer performance [2]. In other words, stereotype threat is a double-edged sword – it could affect positively stereotyped students just as much as negatively stereotyped students in Computer Science.

3.4 Preparation of the student
Could the level of preparation of students be a factor in eliciting stereotype threat? In order to test this hypothesis, the test performance of the students in the control condition was used as a measure of their preparation. The average test score of the control group was calculated and students were grouped into two – those who had scored above average and those who had not. The question was whether students who scored average/below-average were affected more from treatment than those who scored above average. The average score per problem was again analyzed with treatment as the repeated measure and preparation as the between-subjects factor.

As could be expected, a significant main effect was observed for preparation \[F(1,191) = 123.970, p < 0.001\] - the average was:

- 0.940 for students who scored above-average;
- 0.813 for those who scored average/below average.

Notably, the interaction between treatment and preparation was statistically significant \[F(1,191) = 96.663, p < 0.001\] as shown in Table 7:

- The score of above-average students decreased from control to experimental condition and the decrease was statistically significant \[t(119) = 2.891, p = 0.005\];
- The score of students at or below average increased from control to experimental condition and the increase was statistically significant \[t(72) = -8.312, p < 0.001\].

<table>
<thead>
<tr>
<th>Preparation</th>
<th>Control group</th>
<th>Experimental group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Above-Average</td>
<td>0.951 ± 0.015</td>
<td>0.930 ± 0.019</td>
</tr>
<tr>
<td>Average or Below</td>
<td>0.747 ± 0.020</td>
<td>0.879 ± 0.024</td>
</tr>
</tbody>
</table>

It is not clear why the score of above-average students decreased from control to experimental condition. 20 of the 120 above-average students scored 1.0 on the control test, but scored less than 1.0 on the experimental test. Since the maximum score of 1.0 was observed in the control test and not the experimental test, ceiling effect cannot be blamed for the decrease. The 20 students who scored 1.0 on the control test were almost evenly distributed between groups A and B, which eliminates the topic of the test as contributing to the decrease. It is tempting to argue that in the absence of upfront reinforcement of positive stereotype associated with them, above-average students did not do as well in the experimental condition as they did in the control condition. But above-average student population was not homogeneous – it included female and minority students, whose traits are associated with negative rather than positive stereotypes in Computer Science. This unusual observation, however, concurs with the non-significant tendency reported in literature that “non-stereotyped groups seem to perform slightly worse in low stereotype threat conditions” [13].

A 2 X 2 X 2 mixed factor ANOVA with treatment as the repeated measure and preparation and sex as between-subjects factors did not yield any significant interaction between treatment, preparation and sex: the pattern of change in the score observed was the same for male and female students in the two groups of preparation as it was for the whole population. Similarly, a 2 X 2 X 2 mixed factor ANOVA with treatment as the repeated measure and preparation and race as between-subjects factors did not yield any significant interaction between treatment, preparation and race either. The pattern that emerges from these analyses is that less-prepared students scored higher when reminded of stereotypes after rather than before the test. This is true regardless of sex or race.

The students were asked to fill out a survey before each test. In the survey, the students were asked to respond to statements on how well they knew the topic of the test. They responded on a Likert scale of 1 (Strongly agree) to 5 (Strongly disagree). Their responses on the survey were used as an estimate of their self-confidence. Note that on the chosen Likert scale, the lower the numerical response, the more positive the student, and hence, the greater the student’s self-confidence.

Analysis of the survey response with treatment as the repeated measure and preparation as the between-subjects factor yielded a significant main effect for preparation \[F(1,167) = 4.020, p = 0.047\]: not surprisingly, above-average students were more self-confident in control as well as experimental condition, as shown in Table 8. One hypothesis for the test score pattern observed earlier, which is the same regardless of sex or race, is that when average/below-average students are asked to identify their demographic information before the test, they are reminded that they are not confident in their knowledge of the topic.

<table>
<thead>
<tr>
<th>Preparation</th>
<th>Control group</th>
<th>Experimental group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Above-Ave</td>
<td>2.571 ± 0.161</td>
<td>2.548 ± 0.166</td>
</tr>
<tr>
<td>Ave/Below-Ave</td>
<td>2.836 ± 0.204</td>
<td>2.790 ± 0.210</td>
</tr>
</tbody>
</table>

4. DISCUSSION
This study was a follow-up to an earlier study [9] that concluded that stereotype threat may be more readily apparent on harder topics in Computer Science. The results of the current study indicate that on harder topics such as loops, stereotype threat applies to all the students and not just those with whom negative stereotypes are associated. The reasons may be different
– reminder of negative stereotypes for female/minority students, and choking effect [2] for male/Caucasian students, but the effect is the same – all the students scored higher on tests when they were not reminded of stereotypes before the test. Male/Caucasian students also solved problems significantly faster when not reminded of stereotypes before the test. The take-away message is that Computer Science educators must try to avoid eliciting stereotype threat during tests, and especially so when the tests are on harder topics.

The study found that on harder topics in Computer Science, the level of preparation of students is a strong predictor of stereotype threat – average/below-average students scored higher when they were not reminded of stereotypes before the test, regardless of sex or race. Additional analysis not included here showed that this was also true regardless of the major (Computer Science versus other disciplines), status (freshman versus others) or type of institution (4-year versus 2-year). The dearth of computing professionals is acute all around the world, whether in the United States (where enrollments have dropped significantly since the hey days of dot-com bubble), Europe (where talent is being imported on special visas) or in emerging markets (e.g., India cannot produce enough qualified computing professionals to meet its needs). So, the discipline cannot afford to ignore or lose average/below average students. Avoiding stereotype threat during testing would be an effort in the right direction.

In this study, stereotype threat was elicited implicitly by asking students to identify their demographic information, i.e., sex, race, academic status and major. In literature, other mechanisms have been listed for eliciting stereotype threat, e.g., researchers removed objects stereotypical of Computer Science such as Star Trek posters and video games from a classroom and replaced them with non-stereotypical objects such as nature posters to increase the interest of female undergraduate students to the level of their male peers, through a process called “ambient belonging” [6]. In another study, researchers used the gender of the experimenter as a subtle stereotype threat cue [16], and proposed a dual process for how blatant and subtle stereotype threat cues can impact performance. There are other ways in which stereotype threat can be elicited, and therefore, should be avoided – some are listed below:

- A class is separated by sex either during instruction or testing (as might be done in some cultures)
- Students are seated for a test not randomly but by some demographic criterion (such as major) to discourage unwanted collaboration during the test
- Two different tests are handed out, or students are asked to answer odd or even-numbered questions on a test based not on random order, but on some demographic criterion (such as academic status)
- An extra field is included on the test book for the student to enter “maiden name” or credits earned so far towards a degree.
- The wording of the test questions is not gender-neutral.
- The examples used in the test are known to appeal primarily to say male and not female students (i.e., are “boy-themed” [8]), e.g., a catch-as-they-drop game described as a warship catching falling bombs rather than a monkey catching falling fruits.

In each case, the instructor may inadvertently elicit stereotype threat, which can in turn lead to poorer performance on the test.

Given that stereotype threat can be elicited in any number of ways both overt and subtle, it negatively affects the test performance of both positively and negatively stereotyped students on harder topics in Computer Science, and it especially affects less-prepared students, Computer Science instructors should be cognizant of the potential for stereotype threat and take reasonable precautions to avoid it when administering tests. Such precautions are important to ensure that social factors do not affect student performance on the tests, and tests assess only technical knowledge. This is all the more important given the declined enrollment and continuing attrition problem in Computer Science and the projected increase in the demand for Computer Science graduates over the next decade all around the world.

5. ACKNOWLEDGMENTS

Partial support for this work was provided by the National Science Foundation under grant DUE-0817187.

6. REFERENCES


