DOI:10.1145/3474097

Research shows that White, male, and younger engineers receive less pushback than those in other demographics.

BY EMERSON MURPHY-HILL, CIERA JASPAN, CAROLYN EGELMAN, AND LAN CHENG

The Pushback Effects of Race, Ethnicity, Gender, and Age in Code Review

TECH COMPANIES ARE often criticized for a lack of diversity in their engineering workforce. In recent years, such companies have improved engineering workforce diversity through hiring and retention efforts, according to publicly available diversity reports.^a However, we know little about the day-to-day, on-the-job experiences of traditionally underrepresented engineers once they join an organization.^b

A core activity of software engineers at many companies is *code review*, where one or more engineers provide feedback on another engineer's

code to ensure software quality and spread technical knowledge.¹ Beyond software companies, code review has long been practiced in open source software engineering and is emerging as an important practice for scientists.² Code review is fundamentally a decision-making process, where reviewers must decide if and when a code change is acceptable; thus, code review is susceptible to human biases. Indeed, prior research on open source projects suggests that some code reviews authored by women are more likely to be rejected than those authored by men.¹⁶

This article provides confirmation-al evidence that some demographic groups face more code review pushback than others. There is no published research that has studied such differences in a corporate setting.

Method

This section describes the setting of this study, the theory that we ground it in, the dependent and independent variables we use, our modeling approach, and the dataset. While we briefly describe the variables we use here, a full description can be found in the supplementary material at <https://dl.acm.org/doi/10.1145/3474097>.

Setting. Code review at Google is a process that is used in the company's monolithic codebase.¹⁴ When a software engineer makes a code change—to add a new feature or fix a

>> key insights

- Code review, a common practice in software organizations, is susceptible to human biases, where reviewer feedback may be influenced by how reviewers perceive the author's demographic identity.
- Through the lens of role congruity theory, we show the amount of pushback code authors receive varies based on their gender, race/ethnicity, and age.
- We estimate such pushback costs Google more than 1,000 extra engineer hours every day, or approximately 4% of the estimated time engineers spend responding to reviewer comments, a cost borne by non-White and non-male engineers.

a See <https://www.aboutamazon.com/working-at-amazon/diversity-and-inclusion/ourworkforce-data>, <https://www.apple.com/diversity/>, <https://diversity.fb.com/readreport/>, <https://diversity.google/annual-report/>, and <https://www.microsoft.com/enus/diversity/>.

b In this article, for convenience we refer to a person involved in code review as an “engineer” even though, as we shall see, non-engineers are also involved in the code review process.

defect—that code must be reviewed by at least one other engineer. Reviewers evaluate the fitness for purpose of the change, as well as its quality. If they have concerns or questions, they express those comments in the code review tool. Most reviewers are engineers who are on the same team as the author, but reviews can also be performed across different teams, such as when a software engineer fixes a problem in code that they use but normally do not work on. Authors choose their reviewers, but the code review system can also suggest appropriate reviewers. The code review tool provides authors and reviewers with op-

portunities to learn about each other, including their full names and photos (more in the supplementary material).

Theory and hypotheses. Our study is grounded in role congruity theory, which states that a member of a group will receive negative evaluations when stereotypes about the group misalign with the perceived qualities necessary to succeed in a role.³ Applying this to our context, the theory predicts that code reviews will be evaluated negatively when the author of a code change belongs to a group whose stereotypes do not align with the perceived qualities of a successful programmer or software engineer. We evaluate three

different demographic dimensions across which we predict code review evaluations will vary: gender, race/ethnicity, and age.

With respect to gender, we hypothesize that reviews of women coders will be more negative than reviews of men. The rationale is the role mismatch between “the pervasive cultural associations linking men but not women with raw intellectual talent” and that computer science is perceived by some to require high “innate intellectual talent.”⁷ Likewise, we hypothesize that people who identify as Black, Hispanic, or Latinx have greater odds of facing negative evaluations than those who

identify as White, because, as the General Social Survey suggests, Americans are less likely to view those groups as possessing innate intelligence.¹⁵ On the other hand, we hypothesize that those who identify as Asian will face more positive evaluations than those who identify as White, because Asians are stereotypically viewed as having higher role congruity in engineering fields.⁸ We make no hypothesis about role congruity for Native Americans, due to a lack of prior research literature. Recent research shows that contributions from White developers in open source are more likely to be accepted than those from non-White developers.⁹ With respect to age, we hypothesize that older engineers are more likely to experience negative reviews than younger engineers, because of two major role mismatches:

- While there “is a stereotype that older workers have lower ability... and are less productive than younger workers,”¹¹ a great software engineer is expected to be mentally capable of handling complexity and be highly productive.⁹

- While there is a stereotype that older workers “are harder to train, less adaptable, less flexible, and more resistant to change” and “have a lower ability to learn,”¹¹ great software engineers are expected to be open-minded, continuously self-improving, and to not let their understanding stagnate.⁹

Dependent variable. The dependent variable in our predictive model is *pushback*, defined as “the perception of unnecessary interpersonal conflict in code review while a reviewer is blocking a change request.”⁵ In prior work, where we did not provide demographic breakdowns, we compared several quantitative signals that predicted negative evaluations in two ways relevant to role congruity theory: 1) a reviewer requesting excessive changes, and 2) a reviewer withholding approval. The strongest predictors of individual engineers’ self-reported pushback were when a code review had a high number of rounds (that is, back and forth between the author and reviewers), a high amount of time spent by reviewers, and a high amount of time spent by the author addressing the reviewers’ concerns. In that work, such

reviews with high pushback would be in the 90th percentile of each metric: more than nine rounds of review, 48 minutes reviewing, and 112 minutes spent by the author. In this study, we adopt that composite measure as our independent variable by modeling whether a review is likely to be identified as high pushback, or just “pushback” for short.

Independent variables. The independent variables of primary interest are gender, race/ethnicity, and age. Here, we largely use pre-existing demographic categories that Google maintains as part of reporting requirements under U.S. law. For gender, the reported categories are female or male. Race/ethnicity includes Asian+, Black+, Latinx+, Native American+, and White+, where the “+” denotes the fact that engineers can choose multiple race/ethnic identities. For age, we discretize ages into ranges.

Independent control variables. Drawing on prior research about code review,^{5,13,14,17} the independent variables used as controls are based on properties of the change, properties of the author, and other variables:

- For properties of the change, we model the log value of the number of lines changed, the number of reviewers, whether the change contains at least one modification to a file written in a coding language, and several special properties of a review:

- Did the review require a “readability” reviewer⁴—that is, a reviewer certified as an expert in programming language coding standards?

- Was the review part of the readability certification process, in which the author’s expertise in programming language coding standards was being evaluated?

- Was the review a large-scale change (or LSC), approved either by a local code owner or a globally empowered one?

- For properties of the author of the change, we included the level (seniority) of the reviewer, how long they have been at Google, and their job family—for instance, software engineer, site reliability engineer, etc).

- Other variables we captured were the job family of the main reviewer and the relationship between main reviewer and author. By “main reviewer,” we mean the reviewer who has made

the most comments, or in the case of a tie, the first reviewer to comment. We model relationships as “insider” when the author and main reviewer work on the same team; otherwise, we define them as “outsider” reviews. While insider reviews are more common, outsider reviews are necessary when, for example, an author needs to change another team’s code, such as fixing downstream dependencies on an API. Descriptive statistics for all variables are available in the online supplementary material.

Independent interactions variables. Prior work¹⁶ suggests that the relationship between author and reviewer moderates gender bias effects. To account for such a moderating effect, we model the interaction between relationship (insider or outsider) and each independent variable (gender, race/ethnicity, and age).

Modeling approach. Since the dependent variable was binary—either the change was flagged as receiving pushback or not—we used a mixed-effect binomial logistic regression model. In this model, to attempt to control for the same engineer appearing repeatedly as an author or reviewer across code reviews, we use author and main reviewer identities as random effects. As in our prior work on pushback,⁵ we describe the effect size in terms of odds ratios of the primary independent variables, as well as their statistical significance. We address potential multicollinearity issues, performing variance inflation factor (VIF) and generalized variance inflation factor (GVIF)⁶ checks for independent variables; since all continuous variables’ VIF scores were below 1.3 and GVIF scores for categorical variables were below 1.5, we assume that multicollinearity was not a substantial threat to the interpretation of our model. We also ensured the robustness of our analysis by replicating the study on a different dataset; we found that gender and race/ethnicity effects were very consistent, and age effects were largely consistent (see supplementary material).

Dataset. We analyzed code reviews performed in one of the main code review tools at Google over a six-month period from the beginning of January 2019 through the end of June 2019, subject to the following constraints.

Figure 1. Odds ratios from regression analysis predicting pushback in code review for controls (a), the main demographic predictors of interest (b), and the outsider interaction (c). Odds ratios are omitted for non-significant results.

Reviews must have had at least one reviewer (which excludes some experimental, emergency, and documentation changes), and both the author and all reviewers must be full-time-equivalent Google employees working in the U.S. Changes from authors who had incomplete demographic data were ex-

cluded. In sum, this analysis includes more than two million code reviews from over 30,000 authors.

Results

Figure 1 displays the results of our mixed-effect regression predicting code-review pushback. The left half

of the chart shows the model's independent variables, along with their p values in parentheses. The right half shows the odds ratio for each independent variable. Odds ratios of less than 1.0 mean lower odds; odds ratios larger than 1.0 mean higher odds.

Figure 1(a) displays our control

variables. For instance, the first row indicates that the log of the number of lines changed in the code review is significantly ($p < .001$) correlated with pushback. Changing more lines of code increases the odds of the review being flagged with pushback. On the other hand, the odds of a locally approved, large-scale change (LSC) review—generally a low-risk change—being identified with pushback is substantially lower (0.02) compared to non-LSC reviews. As the figure indicates, each new reviewer increases the odds of pushback (2.73), as does whether the review is part of the readability certification process (1.58) and whether a certified readability reviewer is required (1.58). A review without code—for instance, documentation only—is less likely to be flagged with pushback (0.4) than a change being reviewed with code.

As Figure 1(a) indicates, job-relevant author characteristics also change the odds of pushback. Reviews by more senior-level authors are less likely to receive pushback than those of, for instance, an entry-level engineer (level 3). This confirms findings from prior work⁵ that more senior engineers are less likely to face pushback. Likewise, a review author who has been at Google for less than a year is more likely to face pushback than one who has been with the organization longer. Including such experience covariates in our model helps isolate demographic factors—covariates which might otherwise confound results. For instance, Google’s 2020 diversity report states that women tend to have lower attrition than men, and Native American+ employees have higher attrition than White+ employees.

Compared to the most common software engineering role—software engineer, or `ENG_SOFT`—changes authored by other types of engineers (`ENG_OTHER`, such as research scientist engineers) and non-engineers (`OTHER`, such as technical operations employees) are more likely to receive pushback. We did not detect a statistically significant difference in the odds of pushback for changes from site reliability engineers (`ENG_SRE`) compared to the baseline, regular software engineers. Since pushback for SRE authors was not significant, we omit odds ratios for SREs and in several

Some demographic groups face more code review pushback than others.

other places in Figure 1 for non-significant factors.

Figure 1(b) displays results that evaluate our hypotheses—demographic predictors of pushback. Since our model uses an interaction effect between demographics and the relationship, the first set of demographics should be interpreted as applying to insiders—that is, when the author and main reviewer are on the same team.

With respect to gender, consistent with the gender correlations observed on GitHub,¹⁶ women’s changes have a 1.21 higher likelihood of receiving pushback than changes by men. Likewise, compared to White+ engineers, the odds of pushback are higher on authors who identify as Black+ (1.54), Hispanic or Latinx+ (1.15), and Asian+ (1.42). With respect to age, the results show changes from older engineers have higher odds of pushback compared to younger engineers, even after accounting for seniority and tenure. For instance, a change authored by an engineer who is 60 years old or older is more than three times likely to receive pushback than that of an author at the same level and tenure who is between 18 and 24 years old.

Figure 1(c) shows the results for outsider code reviews. Overall, the results indicate that code reviews from engineers on a different team than the author have higher odds (1.15) of pushback. For race/ethnicity and gender, there are few statistically significant differences for insider and outsider code reviews—that is, unlike in prior work,¹⁶ relationships are not a substantial mediating factor. In the cases where there is a statistically significant interaction, the effect is compounding. For instance, compared to an 18-to-24-year-old insider, the model would naively predict that reviews by outsider authors who are between 30 and 34 years old would have 1.36 (1.18 odds for 30–34 years old $\times$ 1.15 odds for outsiders) greater odds of pushback, but the interaction coefficient indicates that the actual odds of pushback for this group is even higher, at 1.77.

In summary, these results indicate that regardless of team relationship between author and main reviewer, authors from some demographic groups face higher odds of pushback during code review than others. Women au-

more online

Online-only supplementary material for this article can be found at <https://dl.acm.org/doi/10.1145/3474097>.

thors face higher odds of pushback than men; Asian, Black, and Hispanic/Latinx authors face higher odds than White authors; and older authors face higher odds than younger authors.

Finally, we have presented effect sizes in terms of odds ratios, but what do these differences mean in practical terms? We answer this question by approximating the excess cost of pushback during the code review process, particularly in terms of additional rounds of review, one component of pushback.⁵ We do this by modeling the number of review rounds a change undergoes, subtracting that from a prediction of the number of rounds it would have taken had the author been a White male, and then estimating the time spent by authors addressing comments in a round of review (details are in the supplementary material, including caveats). We estimate that the total amount of excess time spent during the study period was 1,050 engineer hours per day, or about 4% of the estimated time engineers spend responding to reviewer comments, a cost borne by non-White and non-male engineers. While this number provides one view of the impact of pushback, we would advise readers to interpret this estimate with caution.

Discussion and Conclusion

Compared with prior work, which found that some women faced less-successful code reviews when their gender was apparent,¹⁶ the results in this paper suggest not only that women authors have greater odds of pushback as both outsiders and insiders, but that this effect extends to other demographic groups.

Unlike in an experimental setting, cross-sectional retroactive studies such as ours cannot conclude with certainty that there's a causal relationship between demographic factors and pushback. Potential third variables that we could not control for may exist. For instance, contrary to what we hypothesized from role congruity theory, we found that Asian engineers faced greater odds of pushback than White engineers. The hidden third variable here may be whether the engineer speaks English as

a first language. Those who speak English as a second language may face more difficulty communicating their intent and rationale during a code review discussion, lengthening the time it takes to successfully defend a code review and manifesting as pushback. More broadly, other hidden variables may exist, such as code quality in the change under review. Our analysis is limited in other ways as well, which we enumerate in the supplementary material.

We estimated that more than 1,000 hours per day is spent at Google responding to "excessive" pushback, a cost borne by non-White, non-male, or older engineers. One way to conceptualize this estimate is as an opportunity; if we can reduce pushback for these groups of engineers, they can spend their time being productive elsewhere. But there's also an inverse way to conceptualize this research: White, male, and younger engineers are privileged to receive less pushback than those in other demographics. In either case, we view reducing the gaps between demographic groups as a worthwhile goal, and we expect our software to improve as we attempt to do so.

At Google, a company-wide objective is to make our workplace equitable, and this paper provides one way to measure progress towards this objective. Our initiatives to this end are wide-ranging, from bias-busting training^c to anonymous author code review.¹⁰ We look forward to seeing whether such initiatives will foster more equitable treatment of different groups of engineers in the workplace.

Acknowledgments

We thank Alison Song, Alyson Palmer, Amir Najmi, Andrea Knight, Annie Jean-Baptiste, Ash Kumar, Asim Husain, Ben Holtz, Caitlin Hogan, Collin Green, Dan Friedland, Danny Berlin, David Patterson, David Sinclair, Diane Tang, Elvin Lee, Jill Dicker, Liz Kammer, Luiz André Barroso, Maggie Hodges, Mark Canning, Matthew Jorde, Melody Meckfessel, Melonie Parker, Nina Chen, Rachel Potvin, Ted Smith, and anonymous reviewers for their assistance throughout this research.

c See <https://rework.withgoogle.com/guides/unbiasing-hold-everyone-accountable/steps/give-your-own-unbiasing-workshop/>

References

1. Bacchelli, A. and Bird, C. Expectations, outcomes, and challenges of modern code review. *International Conf. on Software Engineering* (2013), 712-721.
2. Check Hayden, E. Mozilla plan seeks to debug scientific code. *Nature News* 501, 7468 (2013), 472.
3. Eagly, A.H. and Karau, S.J. Role congruity theory of prejudice toward female leaders. *Psychological Review* 109, 3 (2002), 573.
4. Eby, L.T., McManus, S.E., Simon, S.A., and Russell, J.E. The protege's perspective regarding negative mentoring experiences: The development of a taxonomy. *J. of Vocational Behavior* 57, 1 (2000), 1-21.
5. Egelman, C.D., Murphy-Hill, E., Kammer, E., Hodges, M.M., Green, C., Jaspán, C., and Lin, J. Pushback: Characterizing and detecting negative interpersonal interactions in code review. *Intern. Conf. on Software Engineering* (2020), 174-185.
6. Fox, J. and Monette, G. Generalized collinearity diagnostics. *J. of the American Statistical Association* 87, 417 (1992), 178-183.
7. Leslie, S.J., Cimplan, A., Meyer, M., and Freeland, E. Expectations of brilliance underlie gender distributions across academic disciplines. *Science* 347, 6219 (2015), 262-265.
8. Leong, F.T. and Hayes, T.J. Occupational stereotyping of Asian Americans. *The Career Development Quarterly* 39, 2 (1990), 143-154.
9. Li, P.L., Ko, A.J., and Begel, A. What distinguishes great software engineers? *Empirical Software Engineering* 25, 1 (2020), 322-352.
10. Murphy-Hill, E., Dicker, J., Hodges, M., Egelman, C.D., Jaspán, C.N.C., Cheng, L., Kammer, L., Holtz, B., Jorde, M.A., Dolan, A.M.K., and Green, C. Engineering impacts of anonymous author code review: A field experiment. *Trans. on Software Engineering*. (To appear).
11. Nadri, R., Rodriguez-Perez, G., and Nagappan, M. On the relationship between the developer's perceptible race and ethnicity and the evaluation of contributions in OSS. *Trans. on Software Engineering*. (To appear).
12. Posthuma, R.A. and Campion, M.A. Age stereotypes in the workplace: Common stereotypes, moderators, and future research directions. *J. of Management* 35, 1 (2009), 158-188.
13. Potvin, R. and Levenberg, J. Why Google stores billions of lines of code in a single repository. *Communications of the ACM* 59, 7 (2016), 78-87.
14. Sadowski, C., Söderberg, E., Church, L., Sipko, M., and Bacchelli, A. Modern code review: A case study at Google. *Intern. Conf. on Software Engineering: Software Engineering in Practice* (2018), 181-190.
15. Smith, T.W., Davern, M., Freese, J., and Morgan, S.L. *General Social Surveys* (2019).
16. Terrell, J., Kofink, A., Middleton, J., Raine, C., Murphy-Hill, E., Parnin, C., and Stallings, J. Gender differences and bias in open source: Pull request acceptance of women versus men. *PeerJ Computer Science* 3, e111 (2017).
17. Yu, Y., Wang, H., Filkov, V., Devanbu, P., and Vasilescu, B. Wait for it: Determinants of pull request evaluation latency on Github. *Working Conf. on Mining Software Repositories* (2015), 367-371.

Emerson Murphy-Hill (emersonm@google.com) is a research scientist at Google.

Ciera Jaspán is a software engineer at Google.

Carolyn Egelman is a quantitative user experience researcher at Google.

Lan Cheng is a quantitative user experience researcher at Google.

This work is licensed under a <http://creativecommons.org/licenses/by/4.0/>

Watch the authors discuss this work in the exclusive *Communications* video. <https://cacm.acm.org/videos/the-pushback-effects>