Missing Women in Tech: The Labor Market for Highly Skilled Software Engineers

Raviv Murciano-Goroff,**New York University***

October 2018

Abstract

This paper examines the behavior of job seekers and recruiters in the labor market for software engineers. I obtained data from a recruiting platform where individuals can self-report their computer programming skills and recruiters can message individuals they wish to contact about job opportunities. I augment this dataset with measures of each individual's previous programming experience based on analysis of actual computer source code they wrote and shared within the open-source software community. This novel dataset reveals that candidates' self-reported technical skills are quantitatively one of the most important predictors of recruiter interest. Consistent with social psychology and behavioral economics studies, I also find female programmers with previous experience in a programming language are 9.10% less likely than their male counterparts to self-report knowledge of that programming language on their resume. Despite public pronouncements, however, recruiters do not appear more inclined toward recruiting female candidates who self-report knowing programming languages. Indeed, recruiters are predicted to be 11.56% less likely to message a woman than a man with comparable observable qualifications, even if those qualifications are very strong. Ultimately, neither the labor supply nor the labor demand side is adjusting their behavior with regard to the self-reported technical skills in ways that could increase the representation of women among software engineering recruits.

*E-mail: ravivmg@post.harvard.edu. Thank you to Aaron Mangum for his help with wrangling the data. I would like to thank Jonathan Levin, Timothy Bresnahan, Muriel Niederle, Richard B. Freeman, and Paul Oyer for their mentorship and advice. Thank you to the workers at the hiring and recruiting platform who facilitated this research and provided many helpful conversations and insights into the dynamics of the tech labor market. Thank you to Bledi Taska for providing and supporting my use of the Burning Glass data. Thank you to Andrew Chamberlain at Glassdoor. Thank you to Claudia Goldin, Prasanna Tambe, Adina Sterling, Julia Lane, and the seminar participants in the IO and Labor workshops at Stanford University for their helpful comments. My research has been funded by the Ewing Marion Kauffman Foundation as well as the B.F. Haley and E.S. Shaw Fellowship for Economics through a grant to the Stanford Institute for Economic Policy Research. The contents of this publication are solely the responsibility of myself.

1 Introduction

One of the most frequently discussed questions regarding technology companies is why their workforces are persistently gender imbalanced. Despite concerted efforts, many tech companies have been unable to increase the representation of women among their engineering staff. In 2015, tech giants such as Google, Facebook, and Twitter had a mere 17%, 15%, and 10% of their respective technical staff positions filled by female engineers.[1]

The gender imbalance in tech is often attributed to factors on both the labor supply and labor demand sides. Employers note the relatively small number of female students graduating from engineering programs and entering this labor market.[2] Potential workers cite reports about inequitable treatment of female tech workers, harassment in the workplace, and potential discrimination in hiring and promotions.[3] Determining what can be done to improve diversity in recruiting and hiring at tech firms requires answering two questions: 1) do gender differences in the behaviors of job seekers exist, and 2) do recruiters adjust based on such differences in ways that could increase the diversity of the job applicant pool?

This paper examines the initial screening and recruiting of candidates for software engineering positions. Using unique data from a large online recruiting platform, I investigate if there are gender differences in the decisions of job seekers regarding which technical skills to advertise to potential employers. On the hiring and recruiting platform studied, job seekers post digital resumes with a list of skills they feel proficient in. For a subsample of those candidates, I am able to find actual previous computer code they created and uploaded online. Thus, I am able to compare the programming skills individuals claim proficiency in with some of their actual previous coding work. This comparison enables me to quantify the extent of gender differences in the advertising of programming abilities conditional on measures of candidates' actual previous coding work.

In addition, recruiters from major tech companies subscribe to this platform in order to find and contact potential hires. In my dataset, I observe which candidates on the platform recruiters

[1]The data for these statistics comes from the equal-opportunity data websites for these firms as well as news reports, such as https://www.huffingtonpost.com/2015/03/27/women-in-tech_n_6955940.html.

[2]Georgia Wells, "Facebook Blames Lack of Available Talent for Diversity Problem," *The Wall Street Journal (July 14, 2016), https://www.wsj.com/articles/facebook-blames-lack-of-available-talent-for-diversity-problem-1468526303.*

[3]Deepa Seetharaman, "Facebook's Female Engineers Claim Gender Bias," *The Wall Street Journal* (May 2, 2017), https://www.wsj.com/articles/facebooks-female-engineers-claim-gender-bias-1493737116; Katie Benner, "Women in Tech Speak Frankly on Culture of Harassment," *The New York Times* (June 30, 2017), https://www.nytimes.com/2017/06/30/technology/women-entrepreneurs-speak-out-sexual-harassment.html?mcubz=1; Nitasha Tiku, "Bias Suit Could Boost Pay, Open Promotions for Women at Google," *Wired* (September 14, 2017), https://www.wired.com/story/bias-suit-could-boost-pay-open-promotions-for-women-at-google/; Liz Mundy, "Why is Silicon Valley So Awful to Women?" *The Atlantic* (April 2017), https://www.theatlantic.com/magazine/archive/2017/04/why-is-silicon-valley-so-awful-to-women/517788/, Kubler et al. (2017).

expressed interest in contacting. I can therefore examine if the self-reporting of programming languages predicts similar or different probabilities of recruiters showing interest in male and female candidates. Furthermore, I can test if recruiters adjust to gender differences in the propensity of candidates to self-report their known programming skills.

I find three main empirical results. First, among all of the information recruiters receive from candidates about their background, tech recruiters are most responsive to the technical skills that individuals self-report on their digital profile. Even when recruiters can see objective evidence that an individual has previous coding experience in a programming language, individuals who also self-report knowing that programming language are predicted to be approximately 30% more likely to be recruited. The predicted benefits of self-reporting are more limited, however, for those with higher levels of experience in a programming language. Second, female programmers are 9.10% less likely to self-report knowing programming languages that they have experience in than their male counterparts. Surprisingly, this lower propensity to self-report knowledge of a programming language is also apparent when controlling for the usage of one's code by other programmers, a measure of external validation of ones programming skills. Third, recruiters do not adjust for gender differences in the self-reporting of skills. In particular, I do not find evidence that recruiters are more inclined toward recruiting female candidates who self-report knowing a programming language than male candidates with similar profile information shown on this platform.

This paper contributes to two largely discrete literatures. A first set of papers focuses on decisions made on the labor supply side. This literature highlights the existence of gender differences in self-assessed abilities and self-promotional actions. For example, even after controlling for test scores, female students have been shown to self-assess their level of proficiency as lower than their male classmates (Beyer, 1990; Beyer et al., 2003; Correll, 2001). The bias in self-assessments has been shown to be particularly large in stereotypically gendered topics and fields, including Computer Science and Engineering (Beyer, 1990; Beyer et al., 2003). These differences in self-assessments have been shown to have significant impacts on careers. In particular, female college students are less likely to enroll in advanced mathematics courses compared with their male classmates with similar math test scores and grades (Lantz and Smith, 1981), choose quantitative careers despite similar scores in quantitive courses(Correll, 2001), guess on standardized exams (Baldiga, 2013), contribute to group-work settings (Coffman, 2014; Ivanova-Stenzel and Kubler, 2011), and enter into competitive environments (Niederle and Vesterlund, 2007; Niederle et al., 2012). A second literature emphasizes the inequitable treatment of male and female candidates by employers on the labor demand side. Using audits studies, this research carefully measures the extent of discrimination in recruiting (Bertrand and Mullainathan, 2004; Riach and Rich, 2002; McIntyre et al., 1980).

Few studies, however, have explored the interaction of gender differences in the actions of job seekers with hiring managers' decisions. The unique dataset used in this paper allows me to observe both gender differences in labor demand behavior, as well as to test if recruiters appear to be responding to such differences in their recruitment decisions.

Overall, while a variety of factors contribute to the underrepresentation of female engineers in tech, my results show evidence of the importance self-promotional behavior in this labor market. Importantly, these findings indicate that neither the labor supply side nor the labor demand side utilizes the self-reporting mechanism in ways that could increase the percentage of women recruited for software engineering positions.

2 Setting

Many tech companies actively solicit job applications from qualified individuals. Often, recruiters find these individuals on online recruiting platforms. On these platforms, job seekers post a digital version of their resume.[4] Recruiters who subscribe to the platform can then search through the digital resumes to find candidates for open positions. Upon finding a qualified individual, recruiters typically send the candidate a message regarding a job opening that might be of interest.[5]

The platform whose data I use for my analysis in this paper facilitates such recruiting and is popular among tech company recruiters searching for software engineers. This particular platform has both a website for job seekers and a discrete website for recruiters. On the job seeker website, individuals compose their digital resume. On these resumes, candidates describe their educational credentials and work histories. In addition, the digital resumes on the site have space for job seekers to list their known skills. This list of skills is known as the "Self-Reported Skills" list. Individuals typically list both non-technical skills, such as "Writing" and "Social Media," as well as technical skills, such as "Java" and "MySQL." Job seekers interested in software engineering and programmer positions frequently include the names of programming languages they know and use among their self-reported skills.

On the recruiter website, subscribers search and view "candidate profiles" derived from the digital resumes. These "profiles" display the information that candidates wrote about themselves on their digital resumes. In addition, the profiles show the platform's estimation of candidates' programming abilities. Each profile displays a list of programming languages that the platform claims the candidate is "verified" as knowing. The platform constructs this list of "Verified" languages by examining code that the candidates posted online. Many computer programmers upload open

[4]The individuals who upload their resume might not be actively seeking a job when they created their profile on this platform or when recruiters viewed their profile. Many engineers keep a digital resume online even if they are not actively seeking a new employer.

[5]Contacting individuals who are not actively looking for a job is known as "passive recruiting." Finding and contacting individuals about job openings is also sometimes referred to as "outbound recruiting."

source software (OSS), computer programs made freely available with their source code online for others to see and use. In addition, many coders ask and answer questions about computer programming and coding errors on online question-and-answer forums. The programming languages that a candidate either uploaded open source code in or answered a question about are shown in a list of "Verified" languages on their profile. Furthermore, for each "verified" language the platform also indicates if candidates have a "High" or "Low" level of experience based on proprietary analysis of their open source code and question answers.[6]

The combination of the "Self-Reported Skills" list and the "Verified Languages" list creates five different ways that a programming language can appear on a candidate's profile. If a candidate knows a programming language, he or she can either list the language among their self-reported skills or leave it off that list. In addition, the platform can show that language as "Verified - Low Experience," "Verified - High Experience," or not list it among the verified languages list.

The list of self-reported skills and the list of "verified" languages are constructed independently. From the job seeker website, individuals are unable to see the list of programming languages that are displayed as "verified." In addition, the platform lists any languages that an individual has uploaded open source code in as verified regardless of whether or not that programming language was listed by the individual among their self-reported skills.

Recruiters can search the candidate profiles by a limited set of criteria: the name of the school attended, previous employers, geographic location, and a verified language. Recruiters are not able to search by the gender of the candidate. In addition, recruiters could not search for candidates based on a self-reported programming language. For example, if a recruiter searched for Python programmers and a candidate had self-reported knowledge of that language but did not have the language "verified," then that candidate would not appear in the search results.

After searching for candidates, recruiters would open a candidate's profile. On that profile, the recruiter would see both the list of self-reported skills as well as the list of verified programming languages. After viewing this information, a recruiter could choose to press a button to "save" the profile. The recruiter could then return to the saved candidates and send them messages soliciting job applications or requests for interviews.

In 2015, when the data for this paper was collected, employers used and sought out workers with experience in variety of different programming languages. The most highly demanded programming language skills in this year was JavaScript, a language used primarily for web application development. General purpose scripting languages, such as Python and Ruby were associated with jobs having the highest average wages. Finally, the programming language Java was the most

[6] The algorithm for computing this rating is proprietary, and thus the details of what is incorporated into this measure are unclear. This does not impact my analysis, however, as the recruiters and I are aware that both levels of verification are derived from actual code written by candidates.

frequently taught language at the university level, whereas JavaScript was relatively infrequently learned in school. While I focus on the JavaScript programming language in this paper because of its popularity among recruiters and programmers, I include more details on the other programming languages in the Appendix.

3 Data

This paper utilizes two datasets. The first dataset, which I will refer to as the "Profiles dataset," is a cross-section of the profiles available for recruiters to view on the platform in December 2015. For each profile, I construct 259 variables that summarize the information about the candidate that recruiters would see on the profile. I refer to these as the "attributes" of candidates. Details about these attributes are provided in Table A. Among the 259 profile attributes included in the dataset are variables representing if a programming language appeared as self-reported or "verified" and at what level of experience. In addition to the information shown to recruiters, for each profile I observe if any recruiter saved the profile between March 2014 and November 2016.

I restrict attention to candidates likely to consider jobs involving computer programming. In my analysis, I include only profiles that list a bachelor's degree in Computer Science (CS) or immediately related fields, a previous job involving computer programming, or at least one self-reported skill related to software engineering. In addition, I restrict my attention to candidates located in the United States. I use only profiles where the first name of the candidate is strongly associated with either the male or female gender.[7] Finally, I use only the profiles that appeared on the site throughout 2014 to 2016.[8]

The Profiles dataset contains 3,744,305 candidate profiles. Of the candidates, 20.08% are female. Table 1 shows the mean values of attributes displayed on the profiles of male and female candidates. For many attributes, male and female candidates have similar means. For example, a similar percentage of male and female profiles, approximately 12%, hold a bachelor's degrees in CS or immediately related fields and the average year in which both male and female candidates completed their bachelor's degree is around 2000. There are also contrasts. In particular, female candidates are slightly more likely than their male counterparts to list at least one self-reported skill on their resume: 64.01% of female candidates and 54.34% of the male candidates list at least one skill. The current job titles of 40.13% of male candidates and 29.24% of female candidates are

[7] I drop 14.85% of the observations because I am not able to code the gender of the first name. The most frequent dropped names are Terry, Christian, Chris, Pat, Krishna, Wei, and Kim. Wealhinglon and Bechtel (2012) showed that recruiters examine Facebook and LinkedIn photos to gauge responsibleness. This implies that recruiters do pay attention to the photos of candidates. Thus, recruiters could make inferences about the gender of the candidate from the photos of candidates shown on the platform in addition to the first names displayed.

[8] No profiles were deleted during this time, and my data agreement did not provide me access to any of the new profiles that were added to the site.

strongly associated with coding. Finally, the rates at which recruiters contacted male and female candidates is different: 1.84% of male candidates and 0.69% of the female candidates were saved by at least one recruiter.

In Table 2, I show the frequency of how candidates display the popular programming language JavaScript on their profile. Each cell in Table 2 shows a percentage of all profiles. For example, the top-left cell reveals that 0.30% of profiles show JavaScript in their "Self-Reported Skills" list and their "Verified Languages" list. The bottom-left cell shows that 0.24% of profiles indicate that this language is verified, but not self-reported.

This table shows that candidates with higher "Verified Experience" in a programming language are also more likely to list JavaScript among their self-reported skills. Among those who have the language in their "Verified" list with "Low Experience," 25.59% also list the language within their "Self-Reported Skills." In contrast, among those who have the language in their "Verified" list with "High Experience," 55.56% also list the language within their "Self-Reported Skills."

The low rate of self-reporting JavaScript is partly because not every profile lists any self-reported skills. Only 56.28% of the sample list at least one skill—technical or non-technical. Among the subsample of profiles that have at least one self-reported skill, 60.42% of the "Verified - Low" and 80.00% of "Verified - High" coders self-report knowing that language.

Note that self-reporting this language is not adversely selected. For example, among those with Computer Science bachelors degrees who self-report at least one skill, the percentage of verified coders who also self-report is much higher. Within this subsample, 65.02% of "Verified - Low" and 82.30% of "Verified -High" profiles also self-report.

Based on the "Verified" languages listed on candidate profiles in the Profiles dataset, I construct a second dataset. An observation in this data set is a candidate–language pair. I restrict to pairs where the language is listed as one of the candidate's "Verified Languages." In addition to the profile's attributes, this dataset includes the number of lines of code the candidate uploaded to open source repositories in this programming language,[9] the number of distinct days with open source uploads, the years since the first upload of open source in this language, and the number of questions and answers about this language posted on StackOverflow.

I also collect indications of a coder's reputation within the open source community. On the popular open source platform GitHub coders can "follow" each other and thus receive updates about code contributions made by others. In addition, those interested in a particular open source project can "watch" that project and receive updates about changes to that software. The total number of followers that an individual has is an indication of their overall reputation. The number of watchers that a candidate's open source projects have is an indication of the usage of that software.

[9] Lines of code from "forked" repositories, source code which is copied from other open source projects, is removed from this measure.

This second dataset contains 1,687,796 candidate-language pairs. I use only the set of programming languages that meet the following criteria: they are popular programming languages among open source contributors and recruiters, they have distinct names and source code file formats, and they are not markup or style languages.[10] In addition, I use only profiles that list at least one self-reported skill in order to focus on candidates who had thought about the configuration of the self-reported skills on their profile. Finally, I only use candidate-language pairs where the candidate is "verified" in that language because of having uploaded at least one line of open source code.[11]

After the sample selection, the dataset, which I refer to as the "OS Contributors" dataset, contains 443,288 candidate-language pairs. These pairs are derived from 171,981 distinct candidates. This sample is 7.64% of the main sample, but accounts for 47.50% of the profiles saved on the platform between 2014 and 2016.

Summary statistics for the candidate-language pairs in this dataset are shown in Table 3. The distributions of all of the metrics are extremely right skewed. For example, the mean number of lines of JavaScript code uploaded to open source repositories is 35,945.43, but the median is 2,993. Furthermore, the means and variances of the metrics vary widely across languages. Because of this variation, I compute z-scores for each of these metrics by demeaning within language and dividing by the standard deviation of the metric for the language.

The OS Contributors dataset includes only candidate-language pairs based on candidates with at least one "Verified" language. Table 1 shows descriptive statistics of the attributes of the distinct male and female candidates represented in the OS Contributors dataset. In comparison to the larger sample, the candidates in this subsample are younger, more likely to have completed a bachelor's degree in CS or related fields, more likely to currently be employed as coders, and more likely to receive recruiter attention. In addition, this subsample is more gender skewed: a mere 8% of the open source contributors are female.

Like many other online communities, open source communities and programming question-and-answer websites have been shown to not always treat the contributions of female users equitably (Bohren et al. 2017; Terrell et al. 2017). The female candidates represented in the OS Contributors dataset are therefore likely to be particularly committed programmers as they continue to write and upload source code despite potential discrimination on those platforms.

[10]The exact languages included include, JavaScript, Java, C#, C++, PHP, Python, Perl, and Ruby.

[11]The platform could "verify" a candidate's knowledge of a language based on open source contributions or questions and answers on the website StackOverflow. I focus on the candidate-language pairs "verified" by open source as I can download the open source contributions in order to do my own analysis and because it contains actual code written by the individual.

4 Empirical Framework

The three overarching questions of this paper—do recruiters care about self-reported skills, do female coders self-report their known programming languages at the same rate as male coders, and do recruiters adjust their decisions because of the underreporting of skills—require distinct empirical exercises.

The first exercise identifies the subset of candidate attributes that predict if recruiters will save a profile. Using the Profiles dataset, I regress if any recruiter saved a candidate profile on that candidate's attributes:

$$\begin{aligned} saved_i = &\sum_{l \in \mathcal{L}} \beta_{1,l}\text{SR}_{i,l} + \beta_{2,l}\text{VL}_{i,l} + \beta_{3,l}\text{SRVL}_{i,l} + \beta_{4,l}\text{VH}_{i,l} + \beta_{5,l}\text{SRVH}_{i,l} \\ &+ \alpha + \gamma SRS_i + \theta X_i + \epsilon_i \end{aligned} \tag{1}$$

In this equation, $saved_i$ is an indicator for whether or not any recruiter saved profile i between 2014 and 2016. As covariates, I include indicators representing the five different ways in which programming languages appear in a profile: self-reported only (SR), verified with low (VL) or high experience (VH), or both self-reported and verified with with low (SRVL) or high (SRVH) experience. I include these indicators for each of the follow languages: JavaScript, Java, Python, Ruby, PHP, C++, and C#. This set of languages is denoted by $\mathcal{L}$. I also include a vector of indicators, SRS, that represent the other self-reported skills shown on the profile. This vector includes both non-technical and technical skills; it does not include the self-reported programming languages. In this regression, and in many regressions throughout this paper, I include a set of controls, X_i, for the candidate's attributes displayed on the profile. This vector includes the attributes documented in Table A. I will refer to this set of controls in this paper as the "profile attribute controls."

The estimated coefficients represent the average predicted increase in the probability a profile is saved when that profile displays a particular attribute. For example, $\beta_{1,JavaScript}$ is the predicted difference in the probability that profiles with JavaScript in the "Self-Reported Skills" list are saved when compared to otherwise similar profiles without JavaScript listed anywhere. By comparing the magnitudes of these coefficients, I will explore the relative demand associated with candidates advertising different sets of skills, educational credentials, and previous jobs on their profiles.

One might hypothesize that if a recruiter can observe a strong signal that a candidate is capable of programming in a particular language they would disregard whether or not the candidate listed that language among their self-reported skills. I test if recruiters respond to whether or not candidates self-reported knowing a language among candidates whose profiles display "verified" experience in that language. Using the candidates with verified experience in the programming language JavaScript, I construct "profile groups" by finding profiles that are similar along many different dimensions, including their level of verified experience in the JavaScript language.[12] I then examine the groups that contain profiles where the candidate self-reported JavaScript as a skill as well as profile where the candidate did not. I then compare the rate at which recruiters saved profiles with JavaScript self-reported versus those that do not list the language within the same profile groups. Because the platform's search engine only showed candidates based on their verified level of knowledge of a programming language and did not use whether or not they self-reported that language, the difference in the rates at which recruiters saved a profile can be attributed to recruiters' responses to the self-reported skills listed.

I estimate Equation 2 using the sample of profiles that list the language "JavaScript" among their "Verified" programming languages.[13]

$$saved_i = \alpha_{g(i)} + \beta sr_i + \epsilon_i \tag{2}$$

This equation predicts whether or not a profile was saved by a recruiter between 2014 and 2016, $saved_i$, by a fixed effect for the profile group, $\alpha_{g(i)}$, and an indicator for if the profile lists JavaScript in the self-reported skills section, sr_i.[14] The coefficient on sr_i represents the average difference in the probability that a profile with JavaScript in the self-reported skills list is saved by recruiters relative to those that do not.

I hypothesize that the predicted increase in self-reporting on top of verification will be different depending on both the language and the level of experience that a candidate has in that language. In particular, if self-reporting conveys a candidate's interest in working with a particular technology, recruiters are likely to be more responsive to self-reports about less popular, niche programming languages. I, therefore, estimate this equation separately for "Verified - Low" profiles, "Verified - High" profiles, as well as for a variety of different programming languages.

The second question analyzed in this paper asks if male and female coders self-report programming languages they have previous experience in at the same rate. I examine if a male and a female candidate both have similar previous experience coding in a programming language, do they both list that language among their "Self-Reported Skills"?

[12]The profile groups are constructed by matching precisely attributes listed in Table A as well as which lists the programming languages Java, Python, and C# appeared in on the profile. Typically, attempting to match on large numbers of features is challenging because few observations would share the exact same features, and we would the lack empirical variation necessary for statistical inference. This motivates many researchers to use balancing/propensity scores or experimentally constructed audit studies. In this dataset, however, I am able to balance the sample on the strict matching of all profile attributes because of the large number of profiles in the dataset.

[13]In the appendix, I show results for other languages.

[14]Because there are endogenous indicators included as covariates, I chose to estimate this as a linear probability model. Running the same model as a probit, for example, can create additional biases since the covariate is endogenous (Heckman, 1978). See the Appendix for the probit margins.

I use open source code contributions to find a subset of candidates where I can observe a portion of their previous coding work. Using pairs of candidates and the programming languages they have contributed open source code in from the OS Contributors dataset, I examine the propensity of coders to self-report programming languages they have coded in. In Equation 3, I predict if an open source contributor lists a programming language in their list of self-reported skills.

$$sr_{i,l} = \alpha + \beta female_i + \gamma_1 experience_{i,l} + \gamma_2(female_i \times experience_{i,l}) + \gamma_3 X_i + \epsilon_{i,l} \qquad (3)$$

In this equation, $sr_{i,l}$ is an indicator for if candidate i self-reports programming language l, $female_i$ is an indicator for if the candidate is female, and $experience_{i,l}$ is a measure of candidate i's previous programming experience in language l. Finally, X_i represents controls for the educational background, work history, and indications of the career preferences of the candidate.

There are two coefficients that are useful for answering whether or not female coders self-report the programming languages they code in with the same propensity as their male counterparts. First, β shows the difference in the propensity of male and female coders to self-report programming languages they previously coded in on their resume. Second, γ_2 tells us if male and female coders with higher levels of experience in those languages diverge in their propensity to self-report the programming language. For example, if we find a positive coefficient, female candidates with higher levels of previous experience in a language increase their probability of self-reporting by a larger margin than male programmers.

A number of potential reasons might explain why a gender gap in the self-reporting of programming languages would exist. For example, on average, male and female candidates might switch jobs at different rates, have different occupations, different levels of general self-confidence, or even different preferences over careers. I attempt to rule out many of these potential motivations for the difference in self-reporting by segmenting the dataset to various subpopulations in which such differences are attenuated. In addition, I separately examine programming languages associated with higher paid vs lower paid jobs as well as languages typically learned in school versus self-taught.

Lastly, I investigate if the predicted recruiter response to the skills listed on profiles is different for male versus female candidates. In particular, if female candidates systematically self-report programming languages with a lower probability, the female candidates who do self-report would on average have more coding experience than their male counterparts. I test if recruiters are more likely to save the profiles of female candidates who self-report knowledge of a programming language than male candidates with similar profiles.

I predict whether or not any recruiter will save a profile using Equation 4 on observations from the Profiles dataset:

$$\begin{aligned}
saved_i = &\sum_{l \in \mathcal{L}} \Big(\beta_{1,l}\text{SR}_{i,l} + \beta_{2,l}\text{VL}_{i,l} + \beta_{3,l}\text{VH}_{i,l} + \beta_{4,l}\text{SRVL}_{i,l} + \beta_{5,l}\text{SRVH}_{i,l} \Big) \\
&+ female_i \times \sum_{l \in \mathcal{L}} \Big(\delta_{1,l}\text{SR}_{i,l} + \delta_{4,l}\text{SRVL}_{i,l} + \delta_{5,l}\text{SRVH}_{i,l} \Big) \\
&+ \gamma_1 SRS_i + \gamma_2 female_i \times SRS_i \\
&+ \alpha + \theta X_i + \epsilon_i
\end{aligned} \tag{4}$$

This equation is similar to Equation 1, but allows for gender differences in the responsiveness of recruiters to how self-reported skills appear on a profile.

The coefficients of interest from this regression are those on the interaction of $female_i$ with the variables $\text{SRVL}_{i,l}$ and $\text{SRVH}_{i,l}$. The coefficients on the interaction terms tell us if the average increase in the probability that a recruiter saves a profile when they observe a candidate self-reporting a programming language on their resume is equal, larger, or smaller for female candidates relative to their male counterparts.

If female candidates are reporting the programming languages they code in at a lower rate conditional on the same previous coding experience then we would hypothesize that optimizing recruiters who want the most talented coders would show more interest in the female candidates. Therefore, we would anticipate that the interaction of $female_i$ with the variables $\text{SRVL}_{i,l}$ and $\text{SRVH}_{i,l}$ would be positive. If, however, these coefficients are not significantly different from 0 or are negative then this would imply that recruiters may be overlooking more talented coders.[15]

Finally, I also predict whether or not any recruiter will save a profile using Equation 5 on observations from the Profiles dataset:

$$\begin{aligned}
saved_i = &\sum_{l \in \mathcal{L}} \Big(\beta_{1,l}\text{SR}_{i,l} + \beta_{2,l}\text{VL}_{i,l} + \beta_{3,l}\text{VH}_{i,l} + \beta_{4,l}\text{SRVL}_{i,l} + \beta_{5,l}\text{SRVH}_{i,l} \Big) \\
&+ female_i \times \sum_{l \in \mathcal{L}} \Big(\delta_{1,l}\text{SR}_{i,l} + \delta_{2,l}\text{VL}_{i,l} + \delta_{3,l}\text{VH}_{i,l} + \delta_{4,l}\text{SRVL}_{i,l} + \delta_{5,l}\text{SRVH}_{i,l} \Big) \\
&+ \gamma_1 SRS_i + \gamma_2 female_i \times SRS_i \\
&+ \alpha + \theta_1 X_i + \theta_2 female_i \times X_i + \epsilon_i
\end{aligned} \tag{5}$$

[15]None of the recruiters from major tech companies that I have spoken with have been aware of gender differences in the listing of programming languages on resumes. That being said, all of the recruiters are very much aware of differences in male and female candidate behavior that could impact the information shown on resumes, in cover letters, or in interviews. Many of these recruiters receive training on mitigating conscious and unconscious biases.

This linear prediction model allows for recruiters to show different levels of interest in any of the profile attributes across genders.

Again, we might anticipate heterogeneity how firms respond to the profiles of male and female candidates. While the firms are anonymized, I run the above regression separately for larger and smaller firms, firms recruiting for entry level and more experienced candidates, and firms recruiting with thicker and thinner tech labor markets. In addition, I run the regression among recruiters who used the platform for more than a year in order to check if recruiters learn over time about potential underreporting and eventually adjust.

5 Results

5.1 Recruiters Value Specific Technical Skills

Recruiters could search for candidates based on a variety of attributes prior to deciding whether or not to save their profile. I examine which profile attributes predict the largest increases in the probability of being saved. Specifically, I estimated Equation 1 as a linear probability model by regressing if a profile from the Profiles dataset was saved between 2014 and 2016 on all of the profile attributes visible to recruiters. Table 4 shows the estimated coefficients from that regression in the column on the left.

The coefficients on specific technical skills are large relative to traditionally prominent information on paper resumes, such as educational credentials and work history information. For example, self-reporting knowledge of the database program MongoDB is associated with a 0.040 higher probability of being saved, while the big data tool Apache Hive is associated with a 0.058 higher probability of a candidate being saved. For comparison, candidates who hold a bachelor's degree in Computer Science (CS) are associated with a modest 0.01 higher probability of being saved by a recruiter, while currently being employed by one of the top tier tech companies predicts a 0.052 higher probability of recruiter attention.[16]

The increases in recruiter attention associated with programming languages on profiles are even larger. Candidates who are "Verified" with "Low Experience" in JavaScript have a 0.032 higher probability of being saved that those without the language listed. Candidates who are "Verified - Low" and also self-report knowing this language have a 0.093 higher probability of being saved. Candidates who are "Verified" with "High Experience" in JavaScript are predicted to be saved at a 0.114 higher probability than those without this language on their profile, while those who are "Verified - High" and also self-report knowing the language are associated with a 0.208 higher

[16]I use a 2015 survey of software developers to identify the top tier tech companies. My agreement with the data provider prevents me from naming any of the specific companies or universities.

probability.

The way in which programming languages appear on candidates' profiles explains much of the variation regarding which candidates received messages from recruiters. If the linear probability model is run using only the indicators for how programming languages appeared on a candidate's profile, the resulting R^2 is 0.207 or 90.94% of the R^2 from using all 259 profile attributes. Furthermore, recruiters rarely contacted candidates without verified languages. Even among those who received a bachelors degree from one of the universities ranked as having a top tier Computer Science department, recruiters saved only 1.5% of those without a verified language, while they saved 28.44% of those with a verified language.[17]

These findings imply that recruiters focused on candidates with demonstrated experience in specific technical skills. A number of aspects of this labor market encourages recruiters to search for candidates based on technical skills. First, the labor market for software engineers exhibits very high churn. Candidates in the Profiles dataset report switching employers on average every 2.4 years. From an employer's perspective, if their employees will only be at their firm for a limited period of time then investing in training them in particular technologies is relatively costly. Instead, employers are likely to search for candidates who already possess experience in the technology stack that their workers will need to use on the job. Second, the labor market is relatively tight. Tech companies have frequently bemoaned the "skills gap" in which they are unable to find adequate numbers of job seekers qualified in the particular technical skills they desire. If finding candidates with proficiency in particular skills is the primary constraint for the employers, recruiters are likely to start their searches filtering candidates on the basis of the skills listed on the candidate profiles. Third, tech companies have complained that many traditional Computer Science curricula do not teach the practical skills required for building production-ready, large-scale computer programs. Therefore, employers are less inclined to limit their searches based on pedigree. Finally, some recruiters believe that recruiting based on candidates' skills is more "objective" than recruiting on the basis of the schools candidates attended.[18]

5.2 Recruiters' Responses to Self-Report Programming Language Skills

In addition to valuing demonstrated experience in technical skills, candidates who self-report knowing programming languages are also associated with higher chances of being saved by a recruiter. For example, the results in Table 4 showed candidates with "Low Experience" in JavaScript

[17]From the Profiles dataset, 106,217 of the candidate profiles list having attended a bachelors institution with a top tier C.S. department and have no verified languages. 10,991 candidates attended those same schools but also have a verified language.

[18]This opinion was expressed by multiple recruiters that I spoke with at multiple companies. Indeed, some recruiters referenced systems, such as the "competency matrix" framework, as being a more "objective" hiring systems because of their emphasis on the skills of recruiters.

are predicted to have a 6.1 percentage point higher probability of being saved when they also self-reporting this language. Candidates with "High Experience" who also self-report are associated with a 9.4 percentage point higher probability of being saved than those who do not.

While the large positive coefficients on self-reporting verified languages might reflect recruiters' disposition toward candidates who self-report programming languages, two other factors could also create the large coefficients. First, the estimated recruiter response to self-reported languages could be biased because of misspecification. Instead of valuing candidate attributes in an additively separable manner, recruiters may consider a complex set of interactions of a candidate's attributes when deciding whether or not to save their profile. Therefore, the magnitude of the coefficients on self-reporting might misrepresent the importance of this attribute appearing on a candidate's profile. Second, self-reporting is endogenous, and candidates who self-promote their knowledge of programming languages are likely to list other desirable information on their profile as well. Correlations between the decision to self-report and other desirable attributes might make self-reporting predict more recruiter attention than it would on its own.

I address these concerns and investigate the robustness of the prediction that even when a language is shown as "Verified," self-reporting that language garners more recruiter attention on average. In particular, I find groups of candidates whose profiles are very similar except in whether or not they choose to self-report knowing JavaScript. Within each group, I check if on average the profiles that self-report JavaScript receive more recruiter attention than those that do not. This exercise exploits that the search results on this platform did not condition on or sort by whether or not the candidate self-reported the language.

Equation 2 is estimated using candidates in the Profiles dataset who are "Verified" in the language JavaScript. The dependent variable in that regression is whether or not the candidate was saved between 2014 and 2016. The explanatory variables are whether or not the candidate self-reported the language JavaScript as well as a fixed effect for the candidate's "profile group," a cluster of profiles with the same attributes on many salient dimensions.

Table 5 shows the results of estimating this regression. The first column shows the results using only the 48,226 candidates who are "Verified - Low Experience" in JavaScript and are associated with a profile group that has at least candidate who self-reported knowing Javascript and one who did not list that language. The second column shows the results using the 5,922 candidates who are "Verified - High Experience" and have similar matching profiles. The explanatory variable of interest is the indicator for whether or not the candidate listed JavaScript as one of their "Self-Reported Skills."

The coefficient on self-reporting for candidates who are "Verified - Low Experience" in JavaScript is 0.105. Relative to the probability that a candidate with this level of experience is saved, 0.170, this correspondents to a 61.77% higher predicted rate of being save. Similarly, candidates who are

"Verified - High Experience" in JavaScript and self-report are associated with a 0.161 higher probability of being saved, equivalent to a 28.55% higher rate. These results indicate that self-reporting a programming language is indeed associated with large gains even among candidates who have demonstrated their knowledge of that language.

In the Online Appendix, I run the same regression with profiles that are verified with low experience in other programming languages. The results show a similar pattern of significantly higher rates of being saved among candidates who in addition to being verified in a language also self-report knowledge of that programming language among their self-reported skills. I also show the results are robust to using different sets of controls.

While the sensitivity of recruiters to self-reported languages is surprisingly high, self-reporting a programming language is a way that candidates express their preferences for working with particular technologies. Coders choose jobs partially because the position enables them to work with technologies they enjoy using as well as technologies they predict will help them build experience for finding their next job. In fact, as a recent working paper by Prasanna Tambe, Xuan Ye, and Peter Cappelli shows, "firms in the market for skilled technical labor compete not just on wages, but by offering a combination of technologies and wages" (Tambe et al., 2017). In addition to technical proficiency, recruiters are incentivized to care about whether candidates they contact would be likely to respond and accept job offers if they are extended.[19] Therefore, recruiters are likely to examine a candidate profile for indications of the job seeker's preferences and potential interest in positions using various technologies.

5.3 Gender Differences in the Propensity to Self-Report Programming Language Skills

Given that self-reported technical skills are associated with receiving more attention from recruiters, I investigate if there are differences in the propensity of male and female coders to self-report programming languages they have previously coded in.

Figure 1 shows the observed probability that programmers who have uploaded open source code in the programming language JavaScript also list JavaScript as a self-reported skill. In this figure, the 46,567 programmers in the OS Contributors dataset who have uploaded at least one line of JavaScript code to open source and have at least one self-reported skill listed on their profile are grouped into quintiles according to the total number of lines of code in JavaScript

[19]This is often referred to as the "recruiting funnel" (see, for example, `https://www.jobvite.com/general-recruiting/7-benchmark-metrics-to-help-you-master-your-recruiting-funnel/`). It is important to note that at most large tech companies that I have spoken with, the recruiting functions and the HR functions are separated. This means that recruiters are typically not incentivized by candidates' eventual tenure at the firm or even their on-the-job performance. At smaller firms, however, there is often closer interaction between the hiring and human resources roles.

they have contributed to open source over their lifetime. The probabilities that male and female candidates within each quintile self-reported JavaScript are shown on the vertical axis. The figure reveals that within every quintile the female coders who have uploaded JavaScript code have a lower probability of self-reporting that programming language on their resume than their male counterparts.

When coders are ranked by measures of their reputation in the open source community, a similar result is also seen. In Figure 2, I plot the probability that candidates self-report the programming language JavaScript against the total number of "watchers" of a candidate's JavaScript open source projects on the website GitHub.[20] This figure again shows that female programmers are less likely to self-report JavaScript even when other programmers are validating the usefulness of the code they have written by subscribing to updates.

Using a regression framework, I show the robustness of this finding to the inclusion of various controls. In Table 23, I estimate Equation 3 using the sample of candidates who uploaded JavaScript open source code and self-reported at least one skill. In this regression, I predict whether or not a candidate lists the JavaScript programming language amongst their self-reported skills. Included in the regression is a measure of experience in that language based on the number of lines open source code they have uploaded. Each column includes additional controls: The left most column controls only for previous experience, the next column includes fixed effects for the geographic area the candidate resides, the next column includes fixed effects for the candidate's educational background, and the final column includes fixed effects for the candidate's current employer and occupation type.

The coefficient estimates confirm differences in the propensity of male and female coders to self-report programming languages in which they have previous experience. The regressions show a negative coefficient on the indicator for the candidate being female. This coefficient remains negative even after controls are included. Among all of the regression specifications, I find that female coders who have uploaded code in this language self-report knowing this language at a rate between 4.3 and 6.1 percentage-points lower than male coders. Relative to the average rate of self-reporting this language, 0.67, this is equivalent to female coders self-reporting with 6.42%-9.10% lower propensity.

The coefficient on the interaction between the indicator for the candidate being female and the candidate's experience is positive across the regressions. These coefficients show that with each additional increment of experience the female candidates are predicted to increase their probability of self-reporting a language faster than the male candidates. That being said, the estimated interaction terms are not significantly different from zero. Indeed, as Figure 1 showed, female coders who

[20]"Watchers" receive email updates about changes in the software project. These are typically users of the software who wish to know about bug fixes and new features.

have written and uploaded extremely large amounts are still less likely than their male counterparts to self-report knowing this language.

In order to test if the pattern seen in JavaScript is also present in other programming languages, I estimate Equation 3 pooling languages with similar purposes. Table 24 shows the results of running this regression separately for candidates with open source code in web development languages (JavaScript and PHP), scripting languages (Python, Ruby, and Perl), and compiled languages (Java, C#, and C++).

The estimated coefficients from these three regressions show that female coders with open source code work in web development languages self-report those languages at a rate 6.09% lower than their male counterparts. Female coders with experience writing open source code in scripting languages self-report at a rate 10.00% lower than male coders. In contrast, the male and female candidates with open source coding work in compiled programming languages self-report knowing those languages at statistically indistinguishable rates.

These results are robust to looking exclusively at younger cohorts. I estimate the above equations using only the sample of candidates who received their bachelor's degree between 2010 and 2015. The coefficient estimates are show in Table 22. Similar to the previous regressions, I find that female candidates in this sample are 5.45% less likely to self-report knowledge of web development languages and 9.41% less likely to self-report knowledge of scripting languages.

A number of different possible motivations could explain why female coders are less inclined on average toward self-reporting their known programming language skills. Importantly, if the observed difference in self-reporting reflects differences in average preferences over occupations or careers involving programming or particular programming languages than a gender gap in recruiter attention because of the observed difference in self-reporting might be efficient. If, however, female coders with similar preferences for coding jobs and similar programming experience in a language self-report with lower probabilities than their male counterparts, we would predict that recruiters who wished to contact candidates in an equitable fashion based on previous coding experience would show more interest in female coders self-reporting programming skills all else being equal. I test if differences in the average preferences of male and female coders can explain the differences in self-reporting by segmenting to a subsample in which male and female coders are likely to have similar preferences over occupations and examining if difference in self-reporting persists.

For candidates on this recruiting platform, conditional on having a job involving programming during the first five years after graduating college, the probability of being currently employed as a programmer is statistically indistinguishable for men and women (see Appendix F for details). Therefore, I estimate Equation 2 again using only the coders who were employed as computer programmers during the first five years after they graduate from college.

Table 9 displays the results of these regressions. The coefficients reveal a similar pattern: female candidates are approximately 3.53% less likely to self-report web development languages (significant at the 0.10 level), 7.2% less likely to self-report scripting languages (significant at the 0.01 level), while no statistically significant difference can be detected for compiled languages.

While the gap is slightly attenuated, the existence of differences by gender in the propensity to self-report even after examining a subsample with similar rates of being employed in programming occupations implies that differences in average preferences does not account for the entire self-reporting gap. Some of the motivation for the observed gender gap in self-reporting could be related to two previous lines of research. First, men and women may have on average different beliefs regarding how much previous experience in a programming language recruiters expect when they view a candidate self-reporting. This gender gap in beliefs could be related to differences in average levels of self-confidence identified in social psychology and behavioral economics studies (Gneezy et al., 2003; Bursztyn et al., 2017; Niederle and Vesterlund, 2007; Correll, 2001; **?**; Beyer, 1998, 1990; Beyer et al., 2003). A second line of research states that women and minorities who anticipate potential demand-side discrimination may be less inclined toward taking job seeking actions (Becker, 1965; Kang et al., 2016).[21] While my data does not allow me to say definitively what motivates the observed difference in the self-reporting of programming languages, the results in this paper are consistent with differences by gender in the perceived level of experience required to self-report knowing a language.

I roughly estimate the impact of the lower level of self-reporting of known programming skills by female candidates. I compute the predicted probability that the female candidates are saved based on estimates of Equation 1 from all candidates with at least one open source code contribution as shown in Table 4. I then use the estimated coefficients to create predicted probabilities under the hypothetical situation in which anyone who contributed open source code in JavaScript also self-reported that language on their digital resume (details are shown in Appendix E). For female candidates with open source contributions in at least one language, 14.82% are saved by at least one recruiter between 2014 and 2016. If, however, these candidates always self-reported the language JavaScript when they had uploaded open source in it, the linear probability model predicts that 16.23% would be saved. If the female candidates self-reported all of the languages that they contributed open source in, the linear probability model predicts that 18.90% would be saved. This implies that approximately 27.53% more female candidates would be predicted to be contacted by recruiters if all the candidates self-report all the programming languages they contributed code in. While this is a rough estimate, this exercise indicates that self-reporting programming language knowledge could potentially have a large impact on diversity in this hiring pipeline.

[21] Female candidates who do self-report might be evaluated more critically. Evidence of this can be seen in previous research such as Foschi (1996); Eckel and Grossman (1996); Bowles et al. (2005).

5.4 Gender Differences in Recruiters' Response to Profile Attributes

The previous analysis revealed that recruiters seek candidates with experience in technical skills. Furthermore, female candidates who self-report programming languages have on average more experience than their male counterparts who self-report the same languages. An implication of these results is that recruiters searching for the most experienced employees should be more inclined to save the profiles of female candidates with a self-reported language than the profiles of male candidates with the same self-reported language all else being equal. In this section, I assess if recruiters' actions are consistent with utilizing gender differences in the propensity of candidates to self-report for identifying and soliciting job applications from the most experienced coders. In particular, I examine if self-reporting a technical skill predicts a higher chance of being saved for female candidates as compared to male candidates with similar profile attributes.

Equation 4 predicts if a candidate from the Profiles dataset was saved between 2014 and 2016. I use all of the profile attributes as covariates. In addition, I include the interaction of the indicators for each of the self-reported skills as well as the self-reporting of each programming language with an indicator for the candidate being female. If recruiters are using the information that female candidates on average do not self-report with the same propensity as male candidates in their decisions regarding whom to save, we would expect that these interaction terms would be positive and significant. Estimates of the coefficients on these covariates appear in Table 10 in the first column.

The estimated coefficients on the interaction of self-reporting specific technical skills and the candidate being female are negative. Displaying knowledge of a programming language, such as JavaScript, is associated with a higher probability of a recruiter saving a candidate. The increased probability for female candidates, however, is smaller in magnitude. For example, self-reporting JavaScript and displaying the language as "Verified - Low Experience" is associated with a 0.095 increase in the probability of being saved for male candidates. For the 6,745 female candidates with profiles displaying JavaScript in this configuration, the associated increase in probability is 9.47% lower than for male candidates. This point estimate is not statistically differentiable from 0. Self-reporting JavaScript and displaying the language as "Verified - High Experience" is associated with a 0.211 increase in probability of being saved for male candidates, but a 25.12% lower and statistically significant difference in the probability of being saved for the 985 female candidates in this group of elite JavaScript coders.

Noticeably, the coefficients on self-reported non-technical skills show a contrasting pattern. As most recruiters in my dataset are looking for technology workers, candidates whose profiles highlight non-technical skills are associated with lower probabilities of being saved. In contrast, female candidates who list these skills are predicted to have higher probabilities of being recruited than their male counterparts. For example, in Table 10, the appearance of "Project Management" on a male candidate's list of self-reported skills is associated with a 0.001 lower probability of being saved. For the 112,788 female candidates who list this skill, the associated change in probability of being saved is 0.001 higher than for their male colleagues. Similarly, "Public Speaking," "Customer Service," and "Leadership" are all associated with more positive outcomes for female candidates than male candidates.

These results provide evidence that recruiters are not adjusting their screening process to favor female candidates who self-report technical skills over male candidates with equivalent displayed information. While the estimated interaction terms in the above regressions are mostly not significantly distinguishable from zero, they are almost uniformly negative and most are economically significant. This implies that recruiters might be saving male candidates with slightly lower levels of previous experience in programming languages, while overlooking similar female candidates with more experience.[22]

The positive coefficients on the non-technical skills indicate that recruiters are likely conscious of the gender of candidates they observe. While self-reporting non-technical skills are mostly economically insignificant in the early stages of candidate screening in this labor market, these estimates show that recruiters can make adjustments—consciously or unconsciously—regarding the weight that they put on different attributes when viewing the profiles of male and female candidates.

Why might recruiters not use gender to adjust their screening process and find the most experienced coders? Even recruiters who are striving to make their workforce more gender balanced may not be aware that female coders self-report their technical skills with a lower propensity than male coders. I examine if recruiters with relatively more experience assessing candidates are adjusting based on the gender gap in self-reporting. In Table 11, I show the results of estimating Equation 4 using a dependent variable of whether or not a candidate was saved by a recruiter who had been active on the hiring and recruiting platform studied for at least one year. The estimated coefficients on the interaction of the candidate being female with the self-reporting of knowledge of JavaScript is negative or insignificantly different than zero. These results show that even relatively experienced recruiters do not lean towards female candidates when examining self-reported technical skills.

Another reason why recruiters might not use gender to adjust their screening process is because recruiters may believe that by not incorporating the gender of the candidate into their screening process they are ensuring more equitable treatment of male and female candidates. Finally, recruiters might have an objective other than getting the most experienced coders.[23]

[22] There does exist the possibility that recruiters typically show a bias against female candidates and that after incorporating the information about gender differences in the propensity to self-report then the recruiters are closer to equal treatment.

[23] For example, some companies may be more concerned with employee retention, and thus gravitate toward male

Despite recruiters not favoring female candidates based on their self-reported knowledge of technical skills, the possibility remains that recruiters could still show a preference for female candidates more generally.[24] Therefore, it is possible that recruiters favor female candidates on dimensions other than their self-reported skills and that cumulatively this could create a leaning towards female candidates. I test for this by examining the average predicted probability that candidates are saved when allowing for gender differences in the estimated correlation of all profile attributes with the outcome of being saved.

I perform this analysis by estimating Equation 5. This equation predicts whether or not a candidate profile from the Profiles dataset was saved by any recruiter between 2014 and 2016. As covariates, I include all profile attributes described in Table A, as well as their interaction with an indicator for if the candidate is female. Once the coefficients in Equation 5 are estimated, I predict the average probability that the candidates in my sample would be saved if all the candidates were male versus the average probability if all the candidates were female.

The estimated coefficients from Equation 5 are shown in Table 10 in the second column. Educational credentials and work experiences do not predict differentially higher probabilities of being saved for female candidates. For example, male candidates who posses a bachelor's degree in Computer Science are predicted to have a 0.011 higher probability of being saved. For the 84,516 female candidates with this degree, however, the predicted increase is 45.45% less than their male colleagues experience. Similarly, male candidates currently employed in a job strongly associated with programming are associated with a 0.011 higher probability of being saved by recruiters, while female candidates are predicted to receive almost 50% less attention from potential employers. Only one profile attributes is associated with larger gains for female candidates. The coefficient on the candidate being female is 0.001 or 6.25% higher than the unconditional probability of being saved.

For the average candidate being female predicts a lower chance of being saved by a recruiter. The average predicted probability a candidate in the Profiles dataset is saved under the hypothetical world in which all candidates are male is 0.0162. Under the hypothetical world in which all candidates are female, the average predicted probability of being saved is 0.0144. This amounts to female candidates being predicted to receive 11.56% less attention from recruiters.[25]

The decision of recruiters regarding which candidates to contact can be compared with the reputation of those candidates in the open source community. On the popular open source website GitHub, the reputation of an individual coder can be measured in two ways. First, coders have

candidates because of the perception that male coders have a higher likelihood of remaining in technical roles over their career.

[24] A preference for female workers has been shown for short term, hourly, and project-based hires in Chan and Wang (2018).

[25] This difference is statistically significant at the 0.01 level.

“followers” who subscribe to updates about any code contributions made by an individual. Second, a coder’s project has “watchers” who subscribe to updates about changes to a particular project. The number of followers that an individual has is a measure of their overall reputation, while the number of “watchers” to their projects is an indication of the popularity of their coding projects. In this section, I ask if recruiters save the male and female candidates with similar reputations in the open source community along these two measures at the same rate.

Figure 4 displays the probability that recruiters saved a profile given the number of followers of the associated candidate’s account on GitHub. The chart shows that recruiters were more interested in those with more followers on GitHub. Among each bin of the number of followers, however, the female candidates had a lower probability of being saved.

Figure 3 displays the probability that recruiters saved a profile given the number of watchers of the JavaScript open source projects of the associated candidate’s account on GitHub. The chart shows that recruiters were more interested in candidates whose projects had larger numbers of watchers. Again, however, female candidates had a lower probability of being saved than their male counterparts with similar numbers of watchers.

One potential explanation for the gender gap in recruiters saving candidates by reputation could be if recruiters are concerned about attrition and retention. As female programmers are on average more likely to leave positions involving coding over the course of their careers, recruiters who are concerned about attrition may treat female candidates less favorably. This hypothesis, however, fails to explain why recruiters would differentially treat candidates with large numbers of followers on GitHub. For example, candidates with over 100 followers on GitHub are likely committed to careers in engineering. That being said, if recruiters do have a different objective function that could explain some of the observed gender gap in the probability of being saved.

These results show that recruiters are not finding and recruiting all of the qualified female candidates on this platform. While many recruiters say that they are actively searching for female candidates, the above results suggest that some female candidates received less attention than their male counterparts with similar attributes. A variety of reasons might create this empirical result. First, recruiters may consciously or unconsciously be overlooking female candidates because of biases. Given the extensive campaign to increase the number of women in tech, as well as the training that many recruiters receive on mitigating potential biases in the hiring process, it seems unlikely that this could be the sole reason for the above empirical findings. Second, recruiters may be concerned with factors other than those directly viewable on candidate profiles, such as attrition rates of employees. If female candidates are more likely to leave tech occupations or their employers, recruiters might not choose female candidates at similar rates. Finally, because female candidates are thought to be in high demand, recruiters might worry that they would be unable to convince a female candidate to join their firm, and thus are less likely to contact them.

My dataset does not allow me to differentiate between these possible motivations. Additional empirical studies as well as experiments will be required to carefully disentangle them. That being said, my results do indicate that recruiters should be mindful of the possible gender differences in self-reporting when relying on this as a mechanism for finding experienced coders.

6 Conclusion

The tech workforce remains highly gender imbalanced despite considerable efforts by companies to increase diversity and inclusion. In the labor market for software engineers, many companies regularly contact and recruit candidates regarding job openings. Much of this recruiting takes place using online recruiting platforms where individuals can self-report their skills and recruiters can message qualified individuals. I analyzed the behavior of individuals and recruiters on one such recruiting platform. I focused on quantitatively one of the most important predictors for which candidates are recruited: the self-reporting of technical skills.

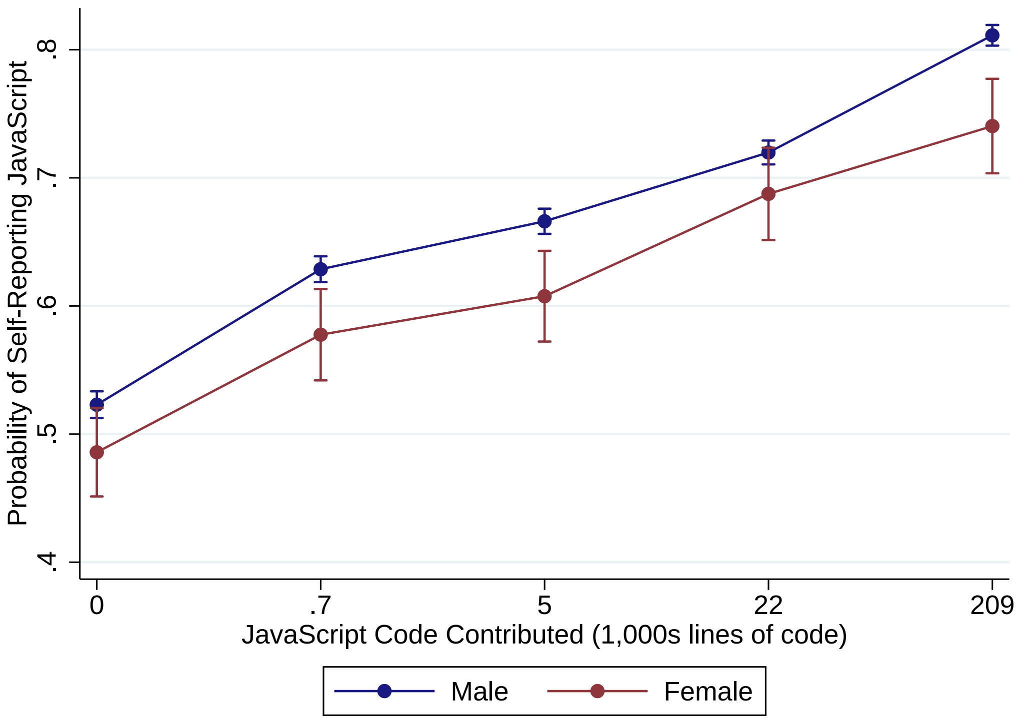
My analysis revealed a gender difference in the propensity to self-report known programming languages on the labor supply side that recruiters on the labor demand side do not appear to adjust for in their decisions. In particular, female coders who contribute to open source software projects in a programming language are on average 9.10% less likely to self-report knowing that same language. While one might have expected that recruiters would therefore be more likely to save female candidates who self-report knowing a language than their male counterparts with similar profiles—anticipating that on average the female candidates would be more experienced in that language than the male candidates—this is not the case. Instead, female candidates are predicted to receive similar benefits from the self-reporting programming languages and on average receive 11.56% less recruiter attention overall after controlling for the candidate attributes shown to recruiters.

Depending on the motivation for the supply-side difference in the propensity to self-report, these findings could have very different implications. If female candidates do not self-report because of a "confidence gap," the lack of adjustment by recruiters for the difference in self-reporting behavior means that employers are likely contacting some male candidates based on self-reported coding skills while overlooking some female candidates with similar actual previous coding experience. If, however, the gender difference in propensity to self-report is primarily due to average differences in preferences for occupations involving coding, the behaviors of both the labor demand and supply sides may be efficient. While my data does not allow me to decompose the precise motivation for the difference in propensity to self-report, the results of this study indicate that neither supply nor demand leverages the self-reporting mechanism by behaving in ways that could increase the percentage of women recruited for software engineering positions.

Finally, these results suggest that employers need to think carefully about the implementation of gender-blind resume reviews as well as algorithmic and machine learning based recruiting systems.[26] While gender-blind resume review can eliminate labor demand biases and potential discrimination, careful consideration must be taken for gender differences in labor supply side behavior.

7 Tables and Figures

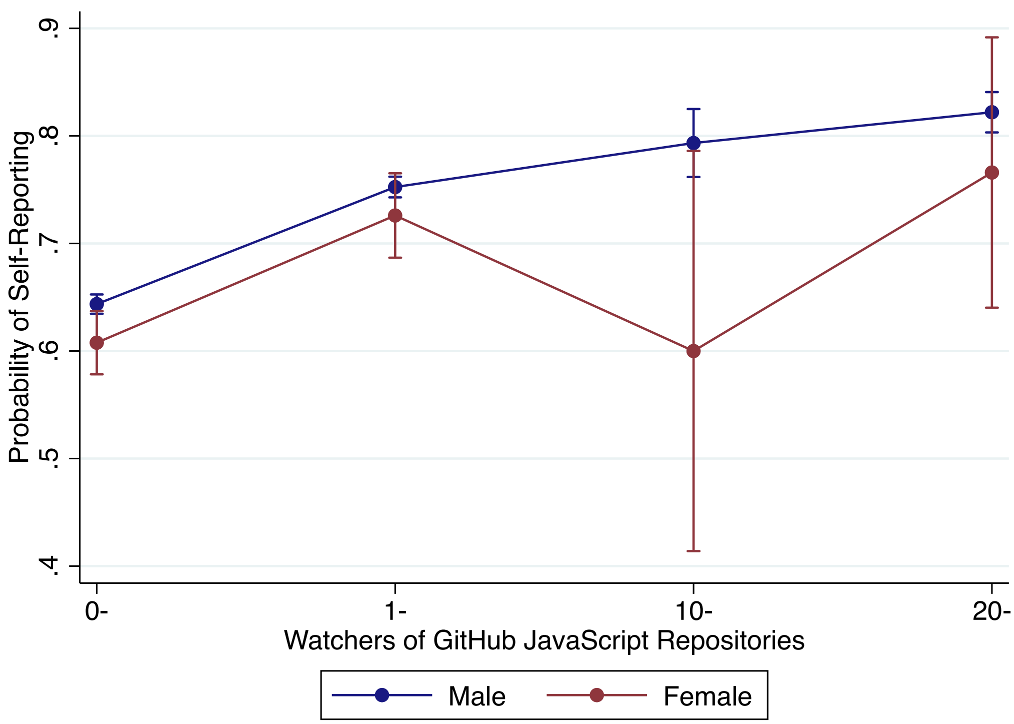
Figure 1: Probability JavaScript is Listed in the Self-Reported Skills Section of the Profiles of JavaScript Open Source Contributing Candidates



Note. Candidates with at least one line of open source code in JavaScript and at least one self-reported skill on their resume are grouped into quintiles according to the total lines of code in JavaScript they have uploaded in JavaScript over their lifetime. The mean probability that male and female coders within each of these quintiles list the language JavaScript amongst the self-reported skills on their resume are plotted. The 95% confidence interval on the means are also plotted. The points are evenly spaced, however, the horizontal axis is labeled with the average number of lines of code uploaded for the corresponding quintiles.

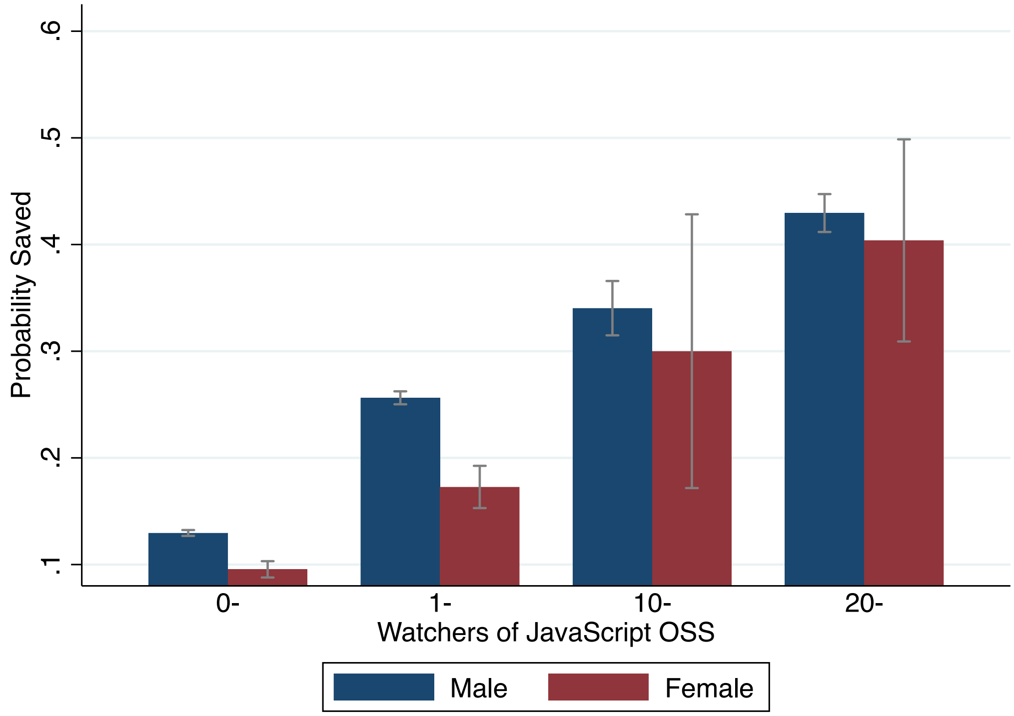
[26]The potential issues that arise can be seen in Amazon's experience building an algorithmic recruiting pipeline (`https://slate.com/business/2018/10/amazon-artificial-intelligence-hiring-discrimination-women.html`).

Figure 2: Probability JavaScript is Listed in the Self-Reported Skills Section of the Profiles Grouped by Number of "Watchers" of JavaScript Repositories



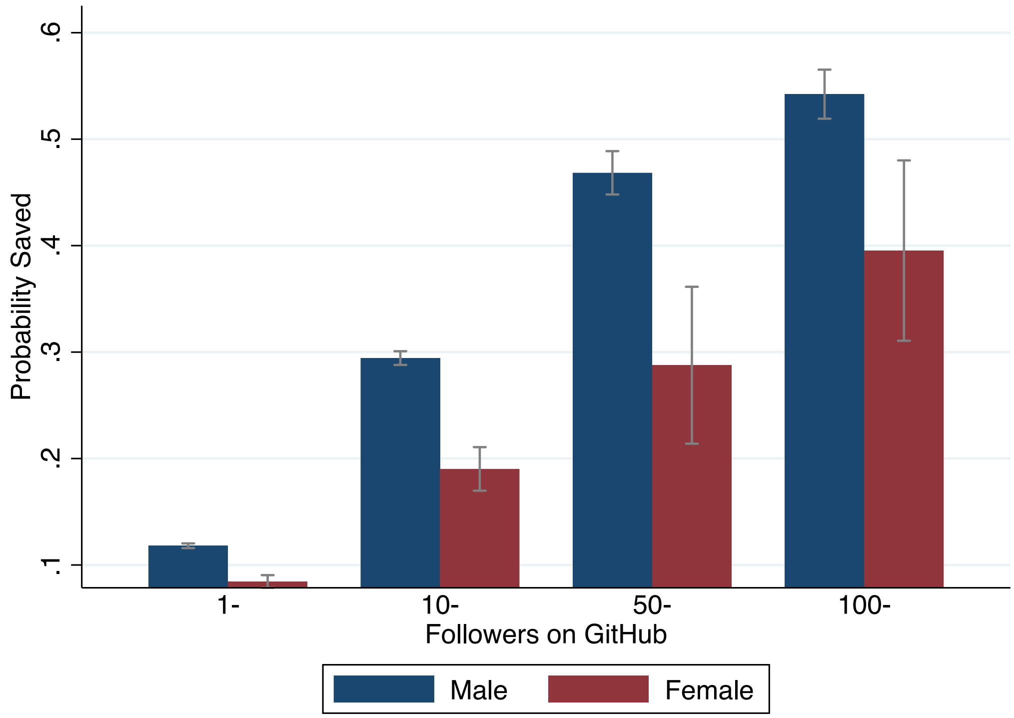
Note. The above graph uses data from candidates with at least one line of open source code in JavaScript and at least one self-reported skill on their resume. These candidates are grouped according to the total number of "watchers" that their JavaScript open source repositories hosted on GitHub have. A "watcher" is someone who subscribes to updates about code changes made in that repository. "Watchers" are typically following these update because they use the code or are interested in the open source project. There are 10,938 male and 1,063 female coders with 0 watchers, 7,776 male and 500 female coders with one to 10 watchers, 634 male and 30 female coders with 10 to 20 watchers, and 1,601 male and 47 female candidates with greater than 20 watchers.

Figure 3: Probability a Profile is Saved versus Number of “Watchers” of JavaScript Repositories



Note. The above graph uses data from candidates with at least one line of open source code in JavaScript and at least one self-reported skill on their resume. On the vertical axis, I show the probability that a profile is saved. These candidates are grouped according to the total number of “watchers” that their JavaScript open source repositories hosted on GitHub have. A “watcher” is someone who subscribes to updates about code changes made in that repository. “Watchers” are typically following these update because they use the code or are interested in the open source project. Blue bars are estimated using only the male candidates; red bar are estimated using only the female candidates.

Figure 4: Probability a Profile is Saved versus Number of "Followers"



Note. The above graph uses data from candidates with at least one line of open source code in JavaScript and at least one self-reported skill on their resume. These candidates are grouped according to the total number of "followers" that they have on GitHub. A "follower" is someone who subscribes to updates about code changes made by this user. The vertical axis shows the probability that a candidate was saved.

Table 1: Mean Value of Attributes on Profiles

	All		OS Contributors		OS Contributors with SR Skills	
	Males	Females	Males	Females	Males	Females
pinned	0.02	0.01	0.20	0.14	0.33	0.23
BA Year	1999.80	2001.49	2005.75	2007.55	2005.84	2007.72
BA in CS	0.14	0.11	0.29	0.27	0.37	0.33
Has Masters	0.18	0.21	0.16	0.23	0.21	0.28
Has Ph.D.	0.03	0.03	0.05	0.05	0.05	0.05
Currently Coder	0.40	0.29	0.46	0.38	0.60	0.51
SR Skills	12.35	14.56	10.94	9.95	26.25	24.30
SR Languages	0.25	0.14	1.09	0.88	2.61	2.15
Verified Languages	0.45	0.11	7.66	5.84	9.41	6.78
Lists BA Degree	0.46	0.49	0.49	0.56	0.61	0.69
Lists SR Skills	0.54	0.64	0.42	0.41	1.00	1.00
SR Prog. Skills	0.42	0.54	0.38	0.37	0.92	0.91
SR JavaScript	0.05	0.03	0.24	0.21	0.58	0.51
SR Python	0.03	0.01	0.15	0.14	0.36	0.34
N	2,992,146	752,159	175,662	14,423	73,213	5,908

Note: An observation is a profile from the Profiles dataset. The means of the attributes of profiles with male and female names are shown in the left and right columns respectively. The first columns show all profiles from the dataset. The second two columns show the profiles of those with at least one verified language. The last column shows means for those with at least one verified language and one verified skill. The above variables are described in detail in Table A

Table 2: Fraction of Profiles Displaying JavaScript as Verified and Self-Reported on their Profile

	Verified - High	Verified - Low	Not Verified
Self-Reported	0.30	0.65	3.81
Not Self-Reported	0.24	1.89	93.11

Note: 266,506 profiles have JavaScript listed as either "Self-Reported" or "Verified" or both. Each cell shows the percentage of all profiles that have the programming language JavaScript listed in either the "Self-Reported Skills" list, the "Verified Languages" list, or both lists. The last line of the table shows the fraction of individuals who are "Verified" at a level of experience who also self-report the language. The left cell shows the fraction of "Verified - High" experience individuals who also self-report knowing JavaScript, while the right cell shows the fraction of "Verified - Low" experience candidates who self-report JavaScript.

Table 3: Summary Statistics of Profile-Language Pairs

	Mean	St.Dev.	P10	P25	P50	P75	P90
Code (lines) C#	6,823.05	34,430.28	33	170	813	3,132	10,823
Code (lines) Java	7,184.66	103,179.60	28	147	721	2,974	10,306
Code (lines) Javascript	35,945.43	164,819.74	24	248	2,993	20,409	75,672
Code (lines) Php	26,561.05	123,161.16	14	80	586	4,567	49,660
Code (lines) Python	5,303.84	31,418.65	12	59	299	1,579	6,897
Code (lines) Ruby	6,057.34	44,057.92	11	74	373	1,542	5,612
Days C#	74.07	427.75	0	0	0	10	108
Days Java	112.42	919.63	0	0	1	22	179
Days Javascript	170.70	1,045.60	0	0	3	54	337
Days Php	130.46	1,094.23	0	0	0	17	196
Days Python	174.67	898.48	0	0	1	37	332
Days Ruby	187.87	1,286.92	0	0	1	33	300
QA Answers C#	3.85	48.14	0	0	0	0	3
QA Answers Java	1.36	30.31	0	0	0	0	0
QA Answers Javascript	0.92	19.10	0	0	0	0	0
QA Answers Php	1.40	20.51	0	0	0	0	0
QA Answers Python	1.03	24.52	0	0	0	0	0
QA Answers Ruby	1.40	19.92	0	0	0	0	0
Profile-Language Pairs	303,341						
Fraction Profile-Language Pairs Female	0.07						
Profiles	141,366						
Fraction Profiles Female	0.08						

Note: An observation in the above table is a candidate profile and a programming language that is listed as "Verified" on that profile. The table shows summary statistics for the metrics on the profile-language pairs. I separate the metrics by programming language. The metrics are "Code", which is the total number of lines of code uploaded to open source repositories, "Days", which are the number of days with an open source code upload, and "QA Answers", which are the total number of answers to questions that are tagged with the programming language on a question-and-answer website about programming. The first line of the table shows the summary statistics of "Code" for profile-language pair observations where the language is "C#". The next line shows the same statistics for profile-language pairs where the language is "JavaScript."

Table 4: Linear Probability Model Predicting if a Profile is Saved

	Saved	
	All	OS Contributors
BA in CS=1	0.010***	0.039***
	(0.000)	(0.003)
Current Employer in Top 10 for Tech	0.052***	0.069***
	(0.002)	(0.007)
SR Public Speaking	0.000	-0.011
	(0.000)	(0.007)
SR MongoDB	0.040***	0.047***
	(0.003)	(0.006)
SR Node.js	0.071***	0.081***
	(0.020)	(0.029)
SR Hive	0.058***	0.059***
	(0.007)	(0.017)
SR Machine Learning	0.012***	0.034***
	(0.002)	(0.007)
Javascript SR, No V	0.008***	0.013***
	(0.001)	(0.005)
Javascript V-Lo, No SR	0.032***	0.009***
	(0.001)	(0.002)
Javascript SR and V-Lo	0.093***	0.039***
	(0.003)	(0.004)
Javascript V-Hi, No SR	0.114***	0.078***
	(0.005)	(0.005)
Javascript SR and V-Hi	0.208***	0.153***
	(0.006)	(0.006)
Controls	Yes	Yes
N	3,744,305	190,085
Dep. Mean.	0.016	0.191
R^2	0.228	0.264

Note: The above table shows a subset of coefficients from estimating Equation 1. An observation in this regression is a profile in the Profiles dataset. The dependent variable is whether or not a recruiter saved the profile between 2014 and 2016. The covariates are described in detail in Table A and include all of the candidate attributes visible to recruiters. Robust standard errors are shown in parentheses.

Table 5: OLS Regression of Indicator for if Profile is Saved on Indicator for JavaScript Appearing in Self-Reported Skills List

	Saved	
	Verified - Low	Verified - High
SR JavaScript	0.105***	0.157***
	(0.004)	(0.012)
Group Fixed Effect	Yes	Yes
N	48,226	5,922
N Groups	6,539	1,492
Dep. Mean	0.17	0.55
R^2	0.020	0.036

Note: The above table shows the estimated coefficient from two OLS regressions. Column (1) uses profiles that list JavaScript as "Verified - Low Experience" while Column (2) uses that list JavaScript as "Verified - High Experience." Profiles are put into groups by those having exact matches for the following features: bachelor's degree year, whether or not the bachelor's degree was in Computer Science, whether or not they have a masters degree, whether or not they have a Ph.D., whether or not they are currently in a job associated with coding, the platform computed "Overall" scores as displayed on the profile, whether or not they self-reported knowing agile methods, Git/SVN, machine learning, REST, and whether or not JavaScript, Java, Python, Ruby, and C# are verified. Note that I do not include the gender of the name of the individual. I also do not include whether or not the programming language JavaScript appears in the self-reported skills section. SR JavaScript is an indicator for if the programming language JavaScript appears amongst the self-reported skills on the profile. A fixed effect for each profile group is included in the regression. Standard errors are clustered at the profile group level.

Table 6: Predicted Probability a Programmer Lists a JavaScript Language in the Self-Reported Skills on their Resume

	Self-Reported			
	(1)	(2)	(3)	(4)
Female	-0.061***	-0.058***	-0.054***	-0.043***
	(0.009)	(0.009)	(0.009)	(0.009)
Code (z-score)	0.034***	0.034***	0.034***	0.033***
	(0.003)	(0.003)	(0.003)	(0.003)
Female x Code (z-score)	0.006	0.006	0.004	0.002
	(0.012)	(0.012)	(0.012)	(0.012)
Geo. FE	No	Yes	Yes	Yes
College x BA Year x Major FE	No	No	Yes	Yes
Work Controls	No	No	No	Yes
N	46,567	46,567	46,567	46,567
N Programmers	46,567	46,567	46,567	46,567
Dependent Mean	0.67	0.67	0.67	0.67
R^2	0.01	0.01	0.05	0.08

Standard errors in parentheses

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Note: The above table shows the estimates of Equation 3 using the 46,567 candidates who uploaded JavaScript open source code. Each columns adds additional controls. The first column run the regression with only the measure of experience (lines of code uploaded to open source in that language, z-score transformed by language) and the interaction with the candidate being female. The second column adds controls for the geographic area in which the candidate resides (state fixed effects). The third column adds controls for the college the candidate attended interacted with the year in which they graduated and whether or not the candidate majored in Computer Science. Indicators are included for if the year and school of the candidate's bachelor's degree is missing on their profile.

Table 7: Predicted Probability a Programmer Lists a Programming Language in the Self-Reported Skills on their Resume – Grouped

	Self-Reported		
	(1) Web	(2) Scripting	(3) Compiled
Female	-0.039***	-0.047***	0.008
	(0.009)	(0.008)	(0.011)
Code (10k lines)	0.002***	0.008***	0.001*
	(0.000)	(0.001)	(0.001)
Female x Code (10k lines)	0.000	-0.000	0.003
	(0.001)	(0.002)	(0.002)
Geo. FE	Yes	Yes	Yes
College x BA Year x Major FE	Yes	Yes	Yes
Work Controls	Yes	Yes	Yes
N	64,460	64,764	47,263
N Programmers	48,200	42,961	34,535
Dependent Mean	0.64	0.47	0.61
R^2	0.07	0.04	0.07

Standard errors in parentheses

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Note: The above table shows the estimates of Equation 3 using the candidates who uploaded open source code. The first column uses only observations where the programming language of the code is either JavaScript or PHP. The second column uses only observations where the programming language of the code is Python, Ruby, or Perl. The third column shows the languages Java, C#, and C++. Included in the regressions are geographic controls (state fixed effects), education controls (college interacted with year of bachelor's degree interacted with whether or not the candidate majored in Computer Science fixed effects), and work controls (whether or not the candidate is currently in a coding occupation and whether or not the candidate works for an employer ranked in the top ten most desirable tech firms).

Table 8: Predicted Probability a Programmer Lists a Programming Language in the Self-Reported Skills on their Resume – Grouped, Recent Grads

	Self-Reported		
	(1) Web	(2) Scripting	(3) Compiled
Female	-0.036**	-0.048***	0.015
	(0.015)	(0.015)	(0.017)
Code (10k lines)	0.002***	0.008***	0.001
	(0.000)	(0.001)	(0.001)
Female x Code (10k lines)	-0.001	-0.002	0.006
	(0.002)	(0.003)	(0.017)
Geo. FE	Yes	Yes	Yes
College x BA Year x Major FE	Yes	Yes	Yes
Work Controls	Yes	Yes	Yes
N	14,085	14,064	12,069
N Programmers	10,444	9,460	8,569
Dependent Mean	0.66	0.51	0.70
R^2	0.07	0.05	0.08

Standard errors in parentheses

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Note: The above table shows the estimates of Equation 3 using the candidates who uploaded open source code and who graduated from college between 2010 and 2015. The first column uses only observations where the programming language of the code is either JavaScript or PHP. The second column uses only observations where the programming language of the code is Python, Ruby, or Perl. The third column shows the languages Java, C#, and C++. Included in the regressions are geographic controls (state fixed effects), education controls (college interacted with year of bachelor's degree interacted with whether or not the candidate majored in Computer Science fixed effects), and work controls (whether or not the candidate is currently in a coding occupation and whether or not the candidate works for an employer ranked in the top ten most desirable tech firms).

Table 9: Predicted Probability a Programmer Lists a Programming Language in the Self-Reported Skills on their Resume – Grouped, 5 year coders

	Self-Reported		
	(1) Web	(2) Scripting	(3) Compiled
Female	-0.024*	-0.036***	0.007
	(0.013)	(0.013)	(0.015)
Code (10k lines)	0.003***	0.006***	0.002*
	(0.000)	(0.002)	(0.001)
Female x Code (10k lines)	-0.000	-0.000	0.003
	(0.001)	(0.003)	(0.003)
Geo. FE	Yes	Yes	Yes
College x BA Year x Major FE	Yes	Yes	Yes
Work Controls	Yes	Yes	Yes
N	27,784	28,122	22,258
N Programmers	20,787	18,469	16,083
Dependent Mean	0.68	0.50	0.67
R^2	0.07	0.05	0.09

Standard errors in parentheses

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Note: The above table shows the estimates of Equation 3 using the candidates who uploaded open source code and who worked as programmers during the first five years after they graduated from college. The first column uses only observations where the programming language of the code is either JavaScript or PHP. The second column uses only observations where the programming language of the code is Python, Ruby, or Perl. The third column shows the languages Java, C#, and C++. Included in the regressions are geographic controls (state fixed effects), education controls (college interacted with year of bachelor's degree interacted with whether or not the candidate majored in Computer Science fixed effects), and work controls (whether or not the candidate is currently in a coding occupation and whether or not the candidate works for an employer ranked in the top ten most desirable tech firms).

Table 10: Predicted Probability a Profile is Saved Using Profile Attributes and Interactions of Self-Reporting

	Saved - All Profiles		Saved - OSS Contributors	
	SR Interacted	All Interacted	SR Interacted	All Interacted
Female	-0.001***	0.001***	-0.008***	0.004
	(0.000)	(0.000)	(0.003)	(0.005)
Javascript SR, No V	0.008***	0.008***	0.017***	0.017***
	(0.001)	(0.001)	(0.005)	(0.005)
Female × Javascript SR, No V	-0.001	-0.000	-0.033**	-0.037**
	(0.002)	(0.002)	(0.017)	(0.017)
Javascript V-Lo, No SR	0.031***	0.048***	0.009***	0.021***
	(0.001)	(0.001)	(0.002)	(0.002)
Female × Javascript V-Lo, No SR		-0.007*		-0.007
		(0.004)		(0.005)
Javascript SR and V-Lo	0.095***	0.121***	0.041***	0.056***
	(0.003)	(0.003)	(0.004)	(0.004)
Female × Javascript SR and V-Lo	-0.009	-0.002	-0.010	-0.012
	(0.011)	(0.011)	(0.013)	(0.014)
Javascript V-Hi, No SR	0.113***	0.146***	0.078***	0.102***
	(0.005)	(0.005)	(0.005)	(0.005)
Female × Javascript V-Hi, No SR		0.001		-0.005
		(0.020)		(0.020)
Javascript SR and V-Hi	0.211***	0.259***	0.157***	0.189***
	(0.006)	(0.005)	(0.006)	(0.006)
Female × Javascript SR and V-Hi	-0.053**	-0.031	-0.059**	-0.050**
	(0.024)	(0.023)	(0.025)	(0.024)
Controls	Yes	Yes	Yes	Yes
N	3,744,305	3,744,305	190,085	190,085
Prob. Saved	0.016	0.016	0.191	0.191
R^2	0.230	0.226	0.266	0.262

pinned

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Note: The above table shows a subset of the coefficients from an OLS regression on profiles in the sample. The dependent variable is an indicator for whether or not a profile was saved by any recruiter between 2014 and 2016. JavaScript SR, No V is defined as an indicator that JavaScript appeared in the self-reported skills list, but was not listed as a "Verified" language. JavaScript V-Lo, No SR is defined as an indicator that JavaScript appeared as a "Verified - Low Experience" language, but did not appear on the list of self-reported skills. JavaScript SR and V-Lo is defined as an indicator that JavaScript appeared as a "Verified - Low Experience" language as well as in the list of self-reported skills. JavaScript V-Hi, No SR is defined as an indicator that JavaScript appeared as a "Verified - High Experience" language, but did not appear on the list of self-reported skills. JavaScript SR and V-Hi is defined as an indicator that JavaScript appeared as a "Verified - High Experience" language as well as in the list of self-reported skills. Controls include the variables shown in Table A. Robust standard errors are used.

Table 11: Predicted Probability a Profile is Saved Using Profile Attributes and Interactions of Self-Reporting, Experienced Recruiters

	Saved - All Profiles		Saved - OSS Contributors	
	SR Interacted	All Interacted	SR Interacted	All Interacted
Female	0.000	0.000	0.002	0.000
	(0.000)	(0.000)	(0.001)	(0.002)
Javascript SR, No V	0.000	-0.000	0.000	-0.000
	(0.000)	(0.000)	(0.002)	(0.002)
Female × Javascript SR, No V	-0.001	-0.000	-0.007	-0.005
	(0.001)	(0.001)	(0.007)	(0.007)
Javascript V-Lo, No SR	0.002**	0.005***	0.002***	0.005***
	(0.001)	(0.001)	(0.001)	(0.001)
Female × Javascript V-Lo, No SR		0.004**		0.002
		(0.002)		(0.002)
Javascript SR and V-Lo	0.003*	0.009***	0.001	0.004*
	(0.002)	(0.001)	(0.002)	(0.002)
Female × Javascript SR and V-Lo	0.001	0.003	-0.002	-0.000
	(0.005)	(0.005)	(0.006)	(0.006)
Javascript V-Hi, No SR	0.017***	0.026***	0.017***	0.023***
	(0.003)	(0.003)	(0.003)	(0.003)
Female × Javascript V-Hi, No SR		0.002		0.000
		(0.012)		(0.012)
Javascript SR and V-Hi	0.041***	0.056***	0.038***	0.049***
	(0.003)	(0.003)	(0.004)	(0.004)
Female × Javascript SR and V-Hi	-0.036***	-0.034***	-0.038***	-0.036***
	(0.013)	(0.013)	(0.013)	(0.013)
Controls	Yes	Yes	Yes	Yes
N	3,744,305	3,744,305	190,085	190,085
Prob. Saved	0.002	0.002	0.033	0.033
R^2	0.091	0.088	0.132	0.130

pinned_gte12months_rec

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Note: The above table shows a subset of the coefficients from an OLS regression on profiles in the sample. The dependent variable is an indicator for whether or not a profile was saved by a recruiter with at least 12 months using the platform between 2014 and 2016. JavaScript SR, No V is defined as an indicator that JavaScript appeared in the self-reported skills list, but was not listed as a "Verified" language. JavaScript V-Lo, No SR is defined as an indicator that JavaScript appeared as a "Verified - Low Experience" language, but did not appear on the list of self-reported skills. JavaScript SR and V-Lo is defined as an indicator that JavaScript appeared as a "Verified - Low Experience" language as well as in the list of self-reported skills. JavaScript V-Hi, No SR is defined as an indicator that JavaScript appeared as a "Verified - High Experience" language, but did not appear on the list of self-reported skills. JavaScript SR and V-Hi is defined as an indicator that JavaScript appeared as a "Verified - High Experience" language as well as in the list of self-reported skills. Controls include the variables shown in Table A. Robust standard errors are used.

Table 12: Average Predicted Probability a Profile is Saved if All Profiles are Male versus Female

	All Male	All Female	Difference	t-stat	p-value
Average Predicted Probability Saved	0.016	0.014	0.002	44.35	0.000

Note: The above table shows the average predicted probability that candidates would be saved by recruiters. I estimate Equation 5 using OLS and then use the estimated coefficients to computer the average predicted probabilities. In the left-most column, I show the average predicted probability of being saved if all candidates were male. The next column shows the predicted probability if all candidates were female. The third column shows the difference between these predicted values. Finally, the last two columns show the statistic from a t-test that these two predicted probabilities are equal as well as the p-value for that test.

References

Babcock, Linda, *Women Don't Ask: Negotiation and the Gender Divide*, Princeton, N.J.: Princeton University Press, 2003.

Baldiga, Katherine, "Gender Differences in Willingness to Guess," *Management Science*, October 2013, *60* (2), 434–448.

Becker, Gary S., *Human Capital: A Theoretical and Empirical Analysis, with Special Reference to Education*, New York: National Bureau of Economic Research, 1965.

Bertrand, Marianne and Sendhil Mullainathan, "Are Emily and Greg More Employable than Lakisha and Jamal? A Field Experiment on Labor Market Discrimination," *American Economic Review*, 2004, *94* (4), 991–1013.

Beyer, Sylvia, "Gender Differences in the Accuracy of Self-Evaluations of Performance," *Journal of Personality and Social Psychology*, 1990, *59* (5), 960–970.

_ , "Gender Differences in Self-Perception and Negative Recall Biases," *Sex Roles*, 1998, *38* (1/2), 103–133.

_ , Kristina Rynes, Julie Perrault, Kelly Hay, and Susan Haller, "Gender Differences in Computer Science Students," in "Proceedings of the 34th SIGCSE Technical Symposium on Computer Science Education," Vol. 35 Reno, Nevada February 2003, pp. 49–53.

Bohren, J. Aislinn, Alex Imas, and Michael Rosenberg, "The Dynamics of Discrimination: Evidence from a Natural Field Experiment," 2017.

Bowles, Hannah Riley, Linda C. Babcock, and Lei Lai, "It Depends Who Is Asking and Who You Ask: Social Incentives for Sex Differences in the Propensity to Initiate Negotiation," *SSRN Electronic Journal*, 2005.

Bursztyn, Leonardo, Thomas Fujiwara, and Amanda Pallais, "'Acting Wife' Marriage Market Incentives and Labor Market Investments," *American Economic Review*, 2017.

Chan, Jason and Jing Wang, "Hiring Preferences in Online Labor Markets: Evidence of a Female Hiring Bias," *Management Science*, July 2018, *64* (7), 2973–2994.

Cheryan, Sapna and Victoria C. Plaut, "Explaining Underrepresentation: A Theory of Precluded Interest," *Sex Roles*, October 2010, *63* (7-8), 475–488.

_ , _ , Caitlin Handron, and Lauren Hudson, "The Stereotypical Computer Scientist: Gendered Media Representations as a Barrier to Inclusion for Women," *Sex Roles*, July 2013, *69* (1-2), 58–71.

Coffman, Katherine Baldiga, "Evidence on Self-Stereotyping and the Contribution of Ideas," *The Quarterly Journal of Economics*, November 2014, *129* (4), 1625–1660.

Cohoon, J. McGrath, "Recruiting and Retaining Women in Undergraduate Computing Majors," *ACM SIGCSE Bulletin*, June 2002, *34* (2), 48.

Correll, Shelley J, "Gender and the Career Choice Process: The Role of Biased Self-Assessments," *American Journal of Sociology*, 2001, *106* (6), 1691–1730.

Croson, Rachel and Uri Gneezy, "Gender Differences in Preferences," *Journal of Economic Literature*, May 2009, *47* (2), 448–474.

Eckel, Catherine C. and Philip J. Grossman, "The Relative Price of Fairness: Gender Differences in a Punishment Game," *Journal of Economic Behavior & Organization*, August 1996, *30* (2), 143–158.

Foschi, Martha, "Double Standards in the Evaluation of Men and Women," *Social Psychology Quarterly*, September 1996, *59* (3), 237.

_ , "Gender and Double Standards in the Assessment of Job Applicants Author (s): Martha Foschi , Larissa Lai and Kirsten Sigerson Source : Social Psychology Quarterly , Vol . 57 , No . 4 (Dec ., 1994), Pp . 326-339 Published by : American Sociological Asso," 2017, *57* (4), 326–339.

Gneezy, Uri, Muriel Niederle, and Aldo Rustichini, "Performance in Competitive Environments: Gender Differences," *The Quarterly Journal of Economics*, 2003, *118* (3), 1049–1074.

Goldin, Claudia and Cecilia Rouse, "Orchestrating Impartiality: The Impact of "Blind" Auditions on Female Musicians," *The American Economic Review*, January 1997, *90* (4), 715–741.

Hadass, Yael S, "The Effect of Internet Recruiting on the Matching of Workers and Employers," 2004.

Heckman, J, "Dummy Endogenous Variables in a Simultaneous Equation System," *Econometrica*, 1978, *46* (4), 931–959.

Ivanova-Stenzel, Radosveta and Dorothea Kubler, "Gender Differences in Team Work and Team Competition," *Journal of Economic Psychology*, October 2011, *32* (5), 797–808.

Kang, Sonia K., Katherine A. DeCelles, András Tilcsik, and Sora Jun, "Whitened Résumés: Race and Self-Presentation in the Labor Market," *Administrative Science Quarterly*, September 2016, *61* (3), 469–502.

Kubler, Dorothea, Julia Schmid, and Robert Stüber, "Be a Man or Become a Nurse: Comparing Gender Discrimination by Employers across a Wide Variety of Professions," April 2017.

Kumar, Amruth N., "A Study of Stereotype Threat in Computer Science," in "in" ACM Press 2012, p. 273.

Lantz, Alma E and Gregory P Smith, "Factors Influencing the Choice of Nonrequired Mathematics Courses," *Journal of Educational Psychology*, 1981, *73* (6), 825–837.

McIntyre, Shelby, Dennis J. Moberg, and Barry Z. Posner, "Preferential Treatment in Preselection Decisions According to Sex and Race," *The Academy of Management Journal*, 1980, *23* (4), 738–749.

Mobius, Markus M, Muriel Niederle, Paul Niehaus, and Tanja S Rosenblat, "Managing Self-Confidence," *Working Paper*, 2014.

Niederle, Muriel and Lise Vesterlund, "Do Women Shy Away from Competition? Do Men Compete Too Much?," *The Quarterly Journal of Economics*, January 2007, *112* (3), 1067–1101.

_ , Carmit Segal, and Lise Vesterlund, "How Costly Is Diversity? Affirmative Action in Light of Gender Differences in Competitiveness," *Management Science*, September 2012, *59* (1), 1–16.

Oyer, Paul and Scott Schaefer, *Personnel Economics: Hiring and Incentives*, Vol. 4, Elsevier B.V., 2011.

Papastergiou, Marina, "Are Computer Science and Information Technology Still Masculine Fields? High School Students' Perceptions and Career Choices," *Computers & Education*, September 2008, *51* (2), 594–608.

Riach, PA and J Rich, "Field Experiments of Discrimination in the Market Place," *Economic Journal*, 2002, *112* (483), F480–F518.

Rynes, Sara L., Amy E. Colbert, and Kenneth G. Brown, "HR Professionals' Beliefs about Effective Human Resource Practices: Correspondence between Research and Practice," *Human Resource Management*, 2002, *41* (2), 149–174.

Singh, Kusum, Katherine R. Allen, Rebecca Scheckler, and Lisa Darlington, "Women in Computer-Related Majors: A Critical Synthesis of Research and Theory From 1994 to 2005," *Review of Educational Research*, December 2007, *77* (4), 500–533.

Smith, Brittany N., Mamta Singh, and Vetle I. Torvik, "A Search Engine Approach to Estimating Temporal Changes in Gender Orientation of First Names," in "in" ACM Press 2013, p. 199.

Spence, Michael, "Job Market Signaling," *The Quarterly Journal of Economics*, August 1973, *87* (3), 355–374.

Spencer, Steven J., Claude M. Steele, and Diane M. Quinn, "Stereotype Threat and Women's Math Performance," *Journal of Experimental Social Psychology*, January 1999, *35* (1), 4–28.

Steele, Claude M, "How Stereotypes Shape Intellectual Identity and Performance," *American Psychologist*, 1997, p. 17.

Tambe, Prasanna and Lorin M. Hitt, "Job Hopping, Information Technology Spillovers, and Productivity Growth," *Management Science*, October 2013, *60* (2), 338–355.

_ , Xuan Ye, and Peter Cappelli, "Paying To Program? Engineering Brand And High-Tech Wages," 2017.

Terrell, Josh, Andrew Kofink, Justin Middleton, Clarissa Rainear, Emerson Murphy-Hill, Chris Parnin, Jon Stallings, Josh Terrell, Andrew Kofink, Justin Middleton, Clarissa Rainear, Chris Parnin, and Jon Stallings, "Gender Differences and Bias in Open Source: Pull Request Acceptance of Women Versus Men," 2017.

Thoms, Peg, Rosemary McMasters, Melissa R. Roberts, and Douglas A. Dombkowski, "Resume Characteristics as Predictors of an Invitation to Interview," *Journal of Business and Psycology*, 1999, *13* (3), 339–356.

Wealhinglon, Bart L. and Autumn R. Bechtel, "Alternative Sources of Information and the Selection Decision Making Process," *Journal of Behavioral and Applied Management; Glendale*, January 2012, *13* (2), 108–120.

A Data and Description of Variables

The data for this paper comes from a hiring and recruiting platform used by both job seekers and employers. The platform is effectively composed of two different sites. The job seeker site enables those interested in jobs to create a digital resume. While some of the individuals who created a digital resume were actively seeking new jobs, many created a profile simply to be visible to potential employers. The job seeker site is general and many workers who were not software engineers or programmers also created a profile. The recruiter site provides a means for employers to find individuals to contact about potential jobs. This site is particularly geared towards finding engineers and technical workers. A range of employers subscribed to the recruiter site including both large and medium sized firms. The majority of subscribers serve as recruiters for a particular employer.

In the following, I describe each variable on each profile record in the dataset:
Saves. The number of times any recruiter who subscribed to the platform pressed a button on the profile indicating that they wished to contact this candidate between March 2014 and November 2016.

Saved. An indicator for if any recruiter pressed a button on the candidate's profile indicating that they wished to contact this candidate between March 2014 and November 2016.

Lists Bachelor's Degree. An indicator for if the candidate described completing a Bachelor's degree and information about that degree appeared on their profile.

Bachelor's Year. The year in which the bachelor's degree listed on the profile was completed. If multiple bachelor's degree are listed then year of the first degree is used.

Bachelor's in CS. Whether or not any of the completed bachelor's degrees listed on the profile have majors in Computer Science, Information Science, Information Systems, Information Technology, Artificial Intelligence, Data Processing, Databases, or System Administration. This list of majors comes from the Department of Education's IPEDS database as being part of Computer and Information Sciences.

Bachelor's School. The school attended for which the first listed bachelor's degree was completed. Only schools within the top 25 colleges for Computer Science according to U.S. News & World Report are included. Many schools have a variety of names and acronyms. Therefore, I limit the number of schools that are distinctly identified. The schools are Carnegie Mellon University, Massachusetts Institute of Technology, Stanford University, University of California-Berkeley, University of Illinois-Urbana-Champaign, Cornell University, University of Washington, Princeton University, Georgia Institute of Technology, University of Texas-Austin, California Institute of Technology, University of Wisconsin-Madison, University of California-Los Angeles, University of Michigan-Ann Arbor, Columbia University, University of California-San Diego, University of Maryland-College Park, Harvard University, University of Pennsylvania, Brown University, Purdue University-West Lafayette, Rice University, University of Southern California, Yale University, and Duke University.

Rank of BA School in CS. The U.S. New & World Report Ranking of the college or university in which the candidate received their bachelor's degree.

Schools Attended. Schools attended for any degree. I include only schools within the top 25 colleges for Computer Science according to U.S. News & World Report. The schools are Carnegie Mellon University, Massachusetts Institute of Technology, Stanford University, University of California-Berkeley, University of Illinois-Urbana-Champaign, Cornell University, University of Washington, Princeton University, Georgia Institute of Technology, University of Texas-Austin, California Institute of Technology, University of Wisconsin-Madison, University of California-Los Angeles, University of Michigan-Ann Arbor, Columbia University, University of California-San Diego, University of Maryland-College Park, Harvard University, University of Pennsylvania, Brown University, Purdue University-West Lafayette, Rice University, University of Southern California, Yale University, and Duke University.

Masters Degree. Indicator for if a master's degree appears on the profile.

Ph.D. Degree. Indicator for if a Ph.D. degree appears on the profile.

Currently Coder. Indicator for if the profile lists a current job with a title associated with a job involving programming. These include any job title with the words, "software", "sde", "coder", “programmer”, “developer”, “engineer”, or “hacker.”

Past Employers. Employers listed in the employment history of the candidate. Only the top 25 employers for tech workers according to Glassdoor’s survey in 2015 are included as separate indicators in the regressions.

Internship. Employers listed in the employment history of the candidate where the job title included the phrase “intern.” Only the top 25 employers for tech workers according to Glassdoor’s survey in 2015 are included as separate indicators in the regressions.

Geographic location. Indicators for the state in which candidate currently resides.

“Overall Candidate Scores”. The platform estimates two scores for each candidate’s predicted relative level of technical skill and potential as an employee. These two scores are between one and five and displayed prominently on their profile. They used a proprietary method for constructing this score that incorporated analysis of the candidate’s work history, education, and open source contributions.

Lists SR Skills. An indicator for if the candidate listed at least one self-reported skill on their profile.

SR Programming. An indicator for if the candidate listed “Programming” as a self-reported skill.

SR Software Dev./Engineering. An indicator for if the candidate listed either “Software Development” or “Software Engineering” as a self-reported skill.

SR Programming. An indicator for if the candidate listed “Web Applications” as a self-reported skill.

SR Git/SVN. An indicator for if the candidate listed either “Git” or “SVN” as a self-reported skill.

SR REST. An indicator for if the candidate listed “REST” as a self-reported skill.

SR Web Dev. An indicator for if the candidate listed “Web Development” as a self-reported skill.

SR Agile. An indicator for if the candidate listed either “Agile Methodologies” or “Agile Practices” as a self-reported skill.

SR Project Management. An indicator for if the candidate listed “Project Management” as a self-reported skill.

SR Program Management. An indicator for if the candidate listed “Program Management” as a self-reported skill.

SR Management. An indicator for if the candidate listed “Management” as a self-reported skill.

SR Leadership. An indicator for if the candidate listed "Leadership" as a self-reported skill.

SR Customer Service. An indicator for if the candidate listed "Customer Service" as a self-reported skill.

SR Social Media. An indicator for if the candidate listed "Social Media" as a self-reported skill.

SR Public Speaking. An indicator for if the candidate listed "Public Speaking" as a self-reported skill.

SR Team Building. An indicator for if the candidate listed "Team Building" as a self-reported skill.

SR JavaScript. An indicator for if the candidate listed either "JavaScript" or various JavaScript libraries as a self-reported skill.

JavaScript in Work Descriptions. An indicator for if the candidate wrote "JavaScript" in a description of their employment history. Note that candidate's descriptions of their previous jobs were not shown to recruiters. Only job titles, employer names, and employment dates were shown to recruiters.

Note that many of these fields are missing because a candidate did not fill them in on their digital resume. Missing values are treated as a distinct value since recruiters would see a blank field when information was missing.

B Predicting Self-Reporting

I explore additional predictors of whether or not an individual who has uploaded open source code in the programming language JavaScript self-reported that language on their digital resume. I estimate Equation 3 separately for candidates based on their quartile of code uploaded in JavaScript. The results of this estimation are displayed in Table 13.

Table 13: Predicted Probability a Programmer Lists a JavaScript Language in the Self-Reported Skills on their Resume

	Self-Reported			
	(1)	(2)	(3)	(4)
Female	0.015	-0.016	-0.080**	-0.055**
	(0.024)	(0.031)	(0.041)	(0.022)
Code (10k lines)	5.463***	0.106***	0.024***	0.001***
	(0.529)	(0.039)	(0.007)	(0.000)
Female x Code (10k lines)	-3.096	-0.000	0.034	-0.000
	(1.948)	(0.141)	(0.027)	(0.001)
coder_5yr	0.124***	0.112***	0.075***	0.091***
	(0.016)	(0.015)	(0.015)	(0.014)
Geo. FE	Yes	Yes	Yes	Yes
College x BA Year x Major FE	Yes	Yes	Yes	Yes
Work Controls	Yes	Yes	Yes	Yes
N	11,962	11,961	11,961	11,961
N Programmers	11,962	11,961	11,961	11,961
Dependent Mean	0.54	0.64	0.69	0.79
R^2	0.13	0.12	0.12	0.12

Standard errors in parentheses

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Note: The above table shows the estimates of Equation 3 using the 46,592 candidates who uploaded JavaScript open source code split into quartiles. Each columns estimates the equation using just the candidates who uploaded the number of lines of code associated with that column's quartile. The left-most column shows the estimates for those with the smallest contributions of code, while the right-most column shows those in the highest quartile of code contributions. The dependent variable is whether or not the language JavaScript was self-reported on the profile.

These results show statistically insignificant coefficients on all but the highest quartile. This implies that the largest detectable difference in candidates' propensity to self-report knowledge of particular languages occurs among the most prolific coders. Given that those in the highest quartile are elite coders with 80% choosing to self-report their particular knowledge of JavaScript, it is somewhat surprising that the gap is most noticeable for this group.

I also estimate Equation 3 adding additional covariates. In particular, I include if the individual self-reported being skilled in leadership and software development. The results of this estimation are displayed in Table 14.

Table 14: Predicted Probability a Programmer Lists a JavaScript Language in the Self-Reported Skills on their Resume

	Self-Reported
	(1)
Female	-0.024***
	(0.009)
Code (z-score)	0.033***
	(0.003)
Female x Code (z-score)	0.003
	(0.011)
BA in CS	0.045***
	(0.005)
bio_coder_zerocoded=1	0.069***
	(0.006)
SR Leadership	-0.101***
	(0.011)
SR Public Speaking	-0.030***
	(0.012)
SR Software Dev.	0.045***
	(0.005)
Geo. FE	Yes
Edu. Controls	Yes
Work Controls	Yes
N	45,470
N Programmers	45,470
Dependent Mean	0.67
R^2	0.06

Standard errors in parentheses

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Note: The above table shows the estimates of Equation 3 using the 46,567 candidates who uploaded JavaScript open source code. In addition to controlling for experience in the language using lines of code uploaded to open source (z-score transformed), I also include controls for the geographic area in which the candidate resides (state fixed effects), the college the candidate attended interacted with the year in which they graduated. Finally, I include whether or not the candidate self-reported skills such as Leadership, Public Speaking, and Software Development, as well as if the candidate majored in Computer Science for a Bachelor's degree and if they describe themselves as a "coder" or "programmer." An indicator is also included for profiles with no educational information available.

These results show that self-reporting the skill "Leadership" is negatively correlated with the probability of self-reporting the programming language JavaScript. In contrast, self-reporting knowledge of "Software Development," is positively correlated with self-reporting the programming language.

One might wonder if individuals who display more confidence in their general skills are also more likely to self-report knowing programming languages that they have previously written code in. Self-reporting "Leadership" may be an indication of a general level of self-confidence. It may, however, also be correlated with the desire to work in non-coding roles. In that case, interpreting the sign and magnitude of the coefficient estimate on this term is ambiguous.

The coefficient on self-reporting "Software Development" as a skill is more clear: those who feel confident in their general software engineering skills are also more likely to enumerate the particular languages they feel proficient in.

Finally, I relax the restriction that the candidates must have at least one self-reported skill on their digital resume. Estimates of Equation 3 using all 94,675 candidates who uploaded JavaScript open source code are shown in Table 15 below.

The results show estimates of the coefficient on the female indicator between -0.037 and -0.052. Relative to a base rate of self-reporting this language of 0.33, this represents a 11.21% to 15.75% difference in the propensity to self-report knowledge of this language.

Table 15: Predicted Probability a Programmer Lists a JavaScript Language in the Self-Reported Skills on their Resume

	Self-Reported			
	(1)	(2)	(3)	(4)
Female	-0.041***	-0.051***	-0.052***	-0.037***
	(0.006)	(0.006)	(0.006)	(0.005)
Code (z-score)	0.042***	0.041***	0.041***	0.038***
	(0.011)	(0.011)	(0.011)	(0.010)
Female x Code (z-score)	0.004	0.004	0.005	0.002
	(0.018)	(0.018)	(0.018)	(0.016)
Geo. FE	No	Yes	Yes	Yes
College x BA Year x Major FE	No	No	Yes	Yes
Work Controls	No	No	No	Yes
N	94,675	94,675	94,675	94,675
N Programmers	94,675	94,675	94,675	94,675
Dependent Mean	0.33	0.33	0.33	0.33
R^2	0.01	0.03	0.07	0.12

Standard errors in parentheses

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Note: The above table shows the estimates of Equation 3 using the 94,675 candidates who uploaded JavaScript open source code. Each columns adds additional controls. The first column run the regression with only the measure of experience (lines of code uploaded to open source in that language, z-score transformed by language) and the interaction with the candidate being female. The second column adds controls for the geographic area in which the candidate resides (state fixed effects). The third column adds controls for the college the candidate attended interacted with the year in which they graduated and whether or not the candidate majored in Computer Science. Indicators are included for if the year and school of the candidate's bachelor's degree is missing on their profile.

I also show the rates of self-reporting for those who self-report at least one skill on their resume as well as those with bachelor's degrees in Computer Science.

Table 16: Fraction of Candidates with Self-Reported Skills Displaying JavaScript as Verified and Self-Reported on their Profile

	Verified - High	Verified - Low	Not Verified
Self-Reported	0.52	1.16	6.76
Not Self-Reported	0.13	0.76	90.66

Note: The above table examines only candidates who self-report at least one skill on their resume. 2,107,461 profiles have JavaScript listed as either "Self-Reported" or "Verified" or both. Each cell shows the percentage of all profiles that have the programming language JavaScript listed in either the "Self-Reported Skills" list, the "Verified Languages" list, or both lists. The last line of the table shows the fraction of individuals who are "Verified" at a level of experience who also self-report the language. The left cell shows the fraction of "Verified - High" experience individuals who also self-report knowing JavaScript, while the right cell shows the fraction of "Verified - Low" experience candidates who self-report JavaScript.

Table 17: Fraction of Candidates with Bachelor's Degrees in Computer Science Displaying JavaScript as Verified and Self-Reported on their Profile

	Verified - High	Verified - Low	Not Verified
Self-Reported	1.86	4.35	23.17
Not Self-Reported	0.40	2.34	67.88

Note: The above table examines only candidates who list a bachelor's degree in Computer Science on their resume and list at least one self-reported skill. 219,521 profiles have JavaScript listed as either "Self-Reported" or "Verified" or both. Each cell shows the percentage of all profiles that have the programming language JavaScript listed in either the "Self-Reported Skills" list, the "Verified Languages" list, or both lists. The last line of the table shows the fraction of individuals who are "Verified" at a level of experience who also self-report the language. The left cell shows the fraction of "Verified - High" experience individuals who also self-report knowing JavaScript, while the right cell shows the fraction of "Verified - Low" experience candidates who self-report JavaScript.

C Measuring Predicted Returns to Self-Reporting with Propensity Score Matching

An alternative means to measure the predicted returns to self-reporting knowledge of a programming language can be computed using propensity score matching. I estimate the first stage matching by estimating the a logit model predicting if a candidate self-reports knowledge of JavaScript on an indicator for the candidate being female, an indicator for the candidate currently being a coder, the candidate scores provided by the platform, indicators for the state of residence of the candidate, indicators for the previous companies the candidate worked at, and indicators for the schools the candidate attended. I then use the logit predict probabilities of self-reporting as propensity scores to find a single nearest-neighbor for each candidate. I then compute the average difference in the probability of being saved.

The results of the propensity score matching procedure reveals similar magnitude to the exact matching procedure described in this paper. The estimated average difference in the propensity to be saved for self-reporters in the verified low category is 0.110 probability points or 57.89% higher than the base saved rate. For verified high experience coders, this procedure predicts that the self-reporters will have a 29.79% higher probability of being saved by a recruiter.

Table 18: Predicted Probability Candidate is Saved Based on Self-Reported Skills Using Propensity Score Matching

	Saved	
	Verified - Low	Verified - High
ATE		
r1vs0.SR JavaScript	0.098***	0.161***
	(0.008)	(0.018)
N	18,974	3,988
Dep. Mean	0.19	0.49

Note: The above table shows the estimated coefficient from two propensity score matching procedures. In both columns, a propensity score is computed for probability that a candidate self-reports knowledge of the programming language JavaScript. This propensity score is computed based on a logit model with the following covariates: an indicator for the candidate being female, an indicator for the candidate currently being a coder, the candidate scores provided by the platform, indicators for the state of residence of the candidate, indicators for the previous companies the candidate worked at, and indicators for the schools the candidate attended. Each candidate is then assigned a nearest neighbor based on their propensity score. Finally, the average difference in the probability of being saved is computed and displayed above. The left column shows this procedure run only candidates with the Verified - Low experience in JavaScript. The right column shows the same procedure when run on Verified - High experience candidates in JavaScript. In both cases, when candidates who attended a particular school or worked for a previous employer have no variation in their self-reporting of JavaScript, I drop those schools or employers. In total, I drop 27 candidates because of lack of variation.

One might hypothesize that recruiters would respond more to self-reported skills when they are listed on digital resumes with shorter employment histories. The reason for this hypothesis is that recruiters would have fewer other indications of a candidate's interest in working with particular technologies.

I examine subsample of those who received their bachelor's degree between 2010 and 2015. These more recent graduates have shorter employment track-records, and thus I test the above hypothesis by checking if the predicted increase in being saved by self-reporting is larger or smaller among recent graduates versus the broader sample. The estimates reveal that candidates who are in the Verified - Low category for JavaScript programming are predicted to have 0.111 probability points higher change of being saved, while those in the Verified - High group are predicted to have 0.137 probability point higher chances of being saved. Compared to the baseline, this translates to

52.86% and 25.85% higher chances of being saved. Thus, these predicted gains are very similar to those found for the general sample, and we do not see a particular different for recent graduates.

Table 19: Predicted Probability Candidate is Saved Based on Self-Reported Skills Using Propensity Score Matching Among Recent College Graduates

	Saved	
	Verified - Low	Verified - High
ATE		
r1vs0.SR JavaScript	0.111***	0.137***
	(0.007)	(0.019)
N	17,127	3,762
Dep. Mean	0.21	0.53

Note: The above table shows the estimated coefficient from two propensity score matching procedures. In both columns, a propensity score is computed for probability that a candidate self-reports knowledge of the programming language JavaScript. This propensity score is computed based on a logit model with the following covariates: an indicator for the candidate being female, an indicator for the candidate currently being a coder, the candidate scores provided by the platform, indicators for the state of residence of the candidate, indicators for the previous companies the candidate worked at, and indicators for the schools the candidate attended. Each candidate is then assigned a nearest neighbor based on their propensity score. Finally, the average difference in the probability of being saved is computed and displayed above. The left column shows this procedure run only candidates with the Verified - Low experience in JavaScript. The right column shows the same procedure when run on Verified - High experience candidates in JavaScript. In both cases, when candidates who attended a particular school or worked for a previous employer have no variation in their self-reporting of JavaScript, I drop those schools or employers. Only profiles that list graduating from college between 2010 and 2015 are used in this estimation procedure.

D Other Measures of Programming Language Experience

In addition to using lines of code as a measure of previous experience in a programming language, I show the robustness of the previous findings to using the number of days with open source uploads and the number of StackOverflow Answers to questions about a programming language as proxies for previous experience. I show estimates of Equation 3 using these as measures of previous experience. The results displayed in the tables below reveal that the basic pattern and magnitudes for the gender gap in self-reporting of programming language knowledge are robust to these alternative measures of past experience.

Table 20: Predicted Probability a Programmer Lists a JavaScript Language in the Self-Reported Skills on their Resume

	Self-Reported			
	(1)	(2)	(3)	(4)
Female	-0.064***	-0.062***	-0.058***	-0.046***
	(0.009)	(0.009)	(0.009)	(0.009)
Years	0.008***	0.008***	0.008***	0.008***
	(0.002)	(0.002)	(0.002)	(0.002)
Female x Years	0.018***	0.018***	0.018***	0.015***
	(0.006)	(0.006)	(0.006)	(0.005)
Geo. FE	No	Yes	Yes	Yes
College x BA Year x Major FE	No	No	Yes	Yes
Work Controls	No	No	No	Yes
N	46,567	46,567	46,567	46,567
N Programmers	46,567	46,567	46,567	46,567
Dependent Mean	0.67	0.67	0.67	0.67
R^2	0.01	0.01	0.05	0.08

Standard errors in parentheses

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Note: The above table shows the estimates of Equation 3 using the 46,567 candidates who uploaded JavaScript open source code. The measure of experience in this regression is the number of days in which a candidate uploaded open source code in a given language converted to years. Each columns adds additional controls. The first column run the regression with only the measure of experience (lines of code uploaded to open source in that language, z-score transformed by language) and the interaction with the candidate being female. The second column adds controls for the geographic area in which the candidate resides (state fixed effects). The third column adds controls for the college the candidate attended interacted with the year in which they graduated and whether or not the candidate majored in Computer Science. Indicators are included for if the year and school of the candidate's bachelor's degree is missing on their profile.

Table 21: Predicted Probability a Programmer Lists a Programming Language in the Self-Reported Skills on their Resume – Grouped

	Self-Reported		
	(1)	(2)	(3)
	Web	Scripting	Compiled
Female	-0.042***	-0.059***	0.008
	(0.008)	(0.008)	(0.011)
Years	0.010***	0.007***	0.013***
	(0.002)	(0.003)	(0.003)
Female x Years	0.020***	0.063***	0.023*
	(0.006)	(0.009)	(0.012)
Geo. FE	Yes	Yes	Yes
College x BA Year x Major FE	Yes	Yes	Yes
Work Controls	Yes	Yes	Yes
N	64,460	64,764	47,263
N Programmers	48,200	42,961	34,535
Dependent Mean	0.64	0.47	0.61
R^2	0.06	0.05	0.08

Standard errors in parentheses

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Note: The above table shows the estimates of Equation 3 using the candidates who uploaded open source code. The first column uses only observations where the programming language of the code is either JavaScript or PHP. The second column uses only observations where the programming language of the code is Python, Ruby, or Perl. The third column shows the languages Java, C#, and C++. Included in the regressions are geographic controls (state fixed effects), education controls (college interacted with year of bachelor's degree interacted with whether or not the candidate majored in Computer Science fixed effects), and work controls (whether or not the candidate is currently in a coding occupation and whether or not the candidate works for an employer ranked in the top ten most desirable tech firms).

Table 22: Predicted Probability a Programmer Lists a Programming Language in the Self-Reported Skills on their Resume – Grouped, Recent Grads

	Self-Reported		
	(1)	(2)	(3)
	Web	Scripting	Compiled
Female	-0.043***	-0.052***	0.015
	(0.015)	(0.015)	(0.017)
Years	0.008**	0.041***	0.030***
	(0.003)	(0.005)	(0.006)
Female x Years	0.018	0.092***	0.072**
	(0.012)	(0.029)	(0.029)
Geo. FE	Yes	Yes	Yes
College x BA Year x Major FE	Yes	Yes	Yes
Work Controls	Yes	Yes	Yes
N	14,085	14,064	12,069
N Programmers	10,444	9,460	8,569
Dependent Mean	0.66	0.51	0.70
R^2	0.07	0.06	0.09

Standard errors in parentheses

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Note: The above table shows the estimates of Equation 3 using the candidates who uploaded open source code and who graduated from college between 2010 and 2015. The first column uses only observations where the programming language of the code is either JavaScript or PHP. The second column uses only observations where the programming language of the code is Python, Ruby, or Perl. The third column shows the languages Java, C#, and C++. Included in the regressions are geographic controls (state fixed effects), education controls (college interacted with year of bachelor's degree interacted with whether or not the candidate majored in Computer Science fixed effects), and work controls (whether or not the candidate is currently in a coding occupation and whether or not the candidate works for an employer ranked in the top ten most desirable tech firms).

Table 23: Predicted Probability a Programmer Lists a JavaScript Language in the Self-Reported Skills on their Resume

	Self-Reported			
	(1)	(2)	(3)	(4)
Female	-0.063***	-0.061***	-0.057***	-0.046***
	(0.009)	(0.009)	(0.009)	(0.009)
StackOverflow Answers	0.000**	0.000**	0.000***	0.000***
	(0.000)	(0.000)	(0.000)	(0.000)
Female x StackOverflow Answers	0.009***	0.009***	0.009***	0.007**
	(0.003)	(0.003)	(0.003)	(0.003)
Geo. FE	No	Yes	Yes	Yes
College x BA Year x Major FE	No	No	Yes	Yes
Work Controls	No	No	No	Yes
N	46,567	46,567	46,567	46,567
N Programmers	46,567	46,567	46,567	46,567
Dependent Mean	0.67	0.67	0.67	0.67
R^2	0.00	0.00	0.05	0.07

Standard errors in parentheses

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Note: The above table shows the estimates of Equation 3 using the 46,567 candidates who uploaded JavaScript open source code. The measure of experience in this regression is the number of answers to questions about a programming language that the candidate posted on StackOverflow. Each columns adds additional controls. The first column run the regression with only the measure of experience (lines of code uploaded to open source in that language, z-score transformed by language) and the interaction with the candidate being female. The second column adds controls for the geographic area in which the candidate resides (state fixed effects). The third column adds controls for the college the candidate attended interacted with the year in which they graduated and whether or not the candidate majored in Computer Science. Indicators are included for if the year and school of the candidate's bachelor's degree is missing on their profile.

Table 24: Predicted Probability a Programmer Lists a Programming Language in the Self-Reported Skills on their Resume – Grouped

	Self-Reported		
	(1) Web	(2) Scripting	(3) Compiled
Female	-0.043***	-0.048***	0.007
	(0.008)	(0.008)	(0.011)
StackOverflow Answers	0.001***	0.001***	0.000***
	(0.000)	(0.000)	(0.000)
Female x StackOverflow Answers	0.011***	0.004**	0.005***
	(0.003)	(0.002)	(0.001)
Geo. FE	Yes	Yes	Yes
College x BA Year x Major FE	Yes	Yes	Yes
Work Controls	Yes	Yes	Yes
N	64,460	64,764	47,263
N Programmers	48,200	42,961	34,535
Dependent Mean	0.64	0.47	0.61
R^2	0.06	0.04	0.07

Standard errors in parentheses

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Note: The above table shows the estimates of Equation 3 using the candidates who uploaded open source code. The first column uses only observations where the programming language of the code is either JavaScript or PHP. The second column uses only observations where the programming language of the code is either Python or Ruby. The third column shows the languages Java, C#, and C++.

E Predicted Probability Candidates Saved If All Languages are Self-Reported

I estimate the coefficients of Equation 1 on the candidates who uploaded open source code in at least one programming language. Using the estimated coefficients, I predict the probability that each candidate was saved by recruiters between 2014 and 2016. I also predict the probability that each candidate would be saved if all candidates who uploaded any open source code in JavaScript also self-reported that language on their profile. The mean predicted probability of a female candidate being saved under these two scenarios is shown in Table 25.

Table 25: Predicted Probability Candidates Saved If JavaScript Contributes Also Self-Reported

	Unadjusted	JavaScript Adjusted	Difference	t-stat	p-value
Predict Probability Saved	0.148	0.163	-0.014	-7.136	0.000
N	28,846				

Note: The above table shows a t-test comparison. The first column, labeled Unadjusted, shows the mean predicted probability that a female candidate is save based on estimates of Equation 1 using all candidates with at least one programming language in which they uploaded open source code. The column labeled "JavaScript Adjusted" shows the mean predicted probability of being saved for female candidates if those candidates self-reported the programming language whenever they had uploaded open source code in that language. The column labeled "Difference" shows the difference in means, while the remaining columns shows the t-statistic from comparing these means as well as the p-value.

I also predict the probability that each candidate would be saved if all candidates who uploaded any open source code in any programming language also self-reported that language on their profile. The mean predicted probability of a female candidate being saved under these two scenarios is shown in Table 26.

Table 26: Predicted Probability Candidates Saved If All Languages with Open Source Contributions Are Also Self-Reported

	Unadjusted	All Languages Adjusted	Difference	t-stat	p-value
Predict Probability Saved	0.148	0.189	-0.041	-19.650	0.000
N	28,846				

Note: The above table shows a t-test comparison. The first column, labeled Unadjusted, shows the mean predicted probability that a female candidate is save based on estimates of Equation 1 using all candidates with at least one programming language in which they uploaded open source code. The column labeled "All Languages Adjusted" shows the mean predicted probability of being saved for female candidates if those candidates self-reported all programming languages that they had uploaded open source code in. The column labeled "Difference" shows the difference in means, while the remaining columns shows the t-statistic from comparing these means as well as the p-value.

F Predicting if Currently Coder

In order to identify subsamples of individuals who are highly likely to have similar preferences over occupations involving computer programming, I predict which candidates are currently in coding occupations. In particular, I regress an indicator for wether or not the candidate is currently in a coding occupation on an indicator for the candidate being female as well as fixed effects for the candidate's geographic location as well the college they attended interacted with the year of their bachelors degree interacted with wether or not they were a Computer Science major. In Table 27, I show the results for this OLS regression on those with JavaScript open source contributions. The first column shows the results for candidates who graduated with the bachelors degree between 2010 and 2015. The second column shows the sample of candidates who held coding occupations during the first five years after they graduated from college.

Table 27: Predicted Probability a Candidate's Current Occupation Involves Coding

	Current Coder	
	(1)	(2)
	Recent BA	Coder After College
Female	-0.083***	-0.011
	(0.016)	(0.011)
Geo. FE	Yes	Yes
College x BA Year x Major FE	Yes	Yes
N	10,143	20,180
N Programmers	10,143	20,180
Dependent Mean	0.72	0.81
R^2	0.09	0.10

Standard errors in parentheses

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

Note: The above table shows the results of a linear probability model. The dependent variable is an indicator for if the candidate is currently employed in a coding occupation. Included as covariates are an indicator for the candidate being female as well as state fixed effects and fixed effects for the school that the candidate graduated from interacted with the year they graduated interacted with whether or not they majored in Computer Science. This regression is only run on the sample of candidates with at least one line of JavaScript open source code uploaded and at least one self-reported skill on their profile. The first column shows the results for those that graduated college between 2010 and 2015. The second column shows the regression estimated on those who were in coding positions during the five years after they graduated from college.

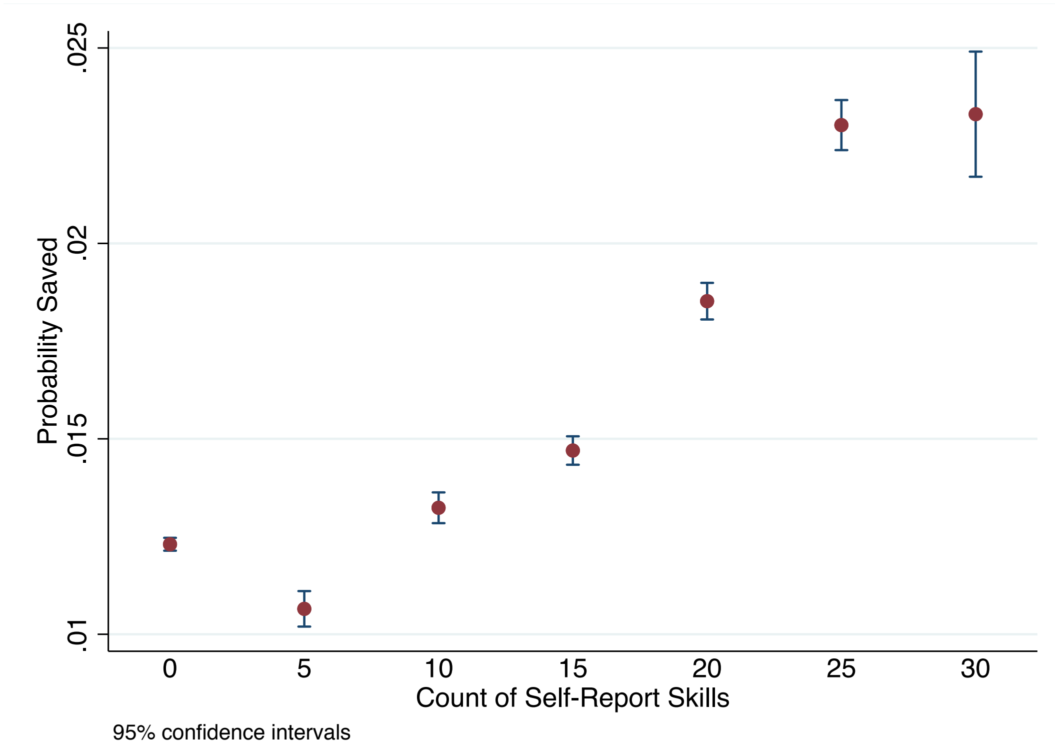
The results show that among recent college graduates with open source contributions in JavaScript on this platform, female candidates are 8.3 probability points less likely to currently be employed in coding occupations. In contrast, male and female candidates who were in occupations involving coding during the five years after they graduates have statistically indistinguishable rates of being in coding occupations when the data was collected.

G Probability of Being Saved Given the Self-Reported Skills and Languages

In the following plots, I show the probability that a candidate is saved relative to the number of self-reported skills and the number of programming languages listed among their self-reported skills. The plots show relatively linear increases in the probability of being saved relative to the number of skills. In particular, the plots do not show an obvious concavity.

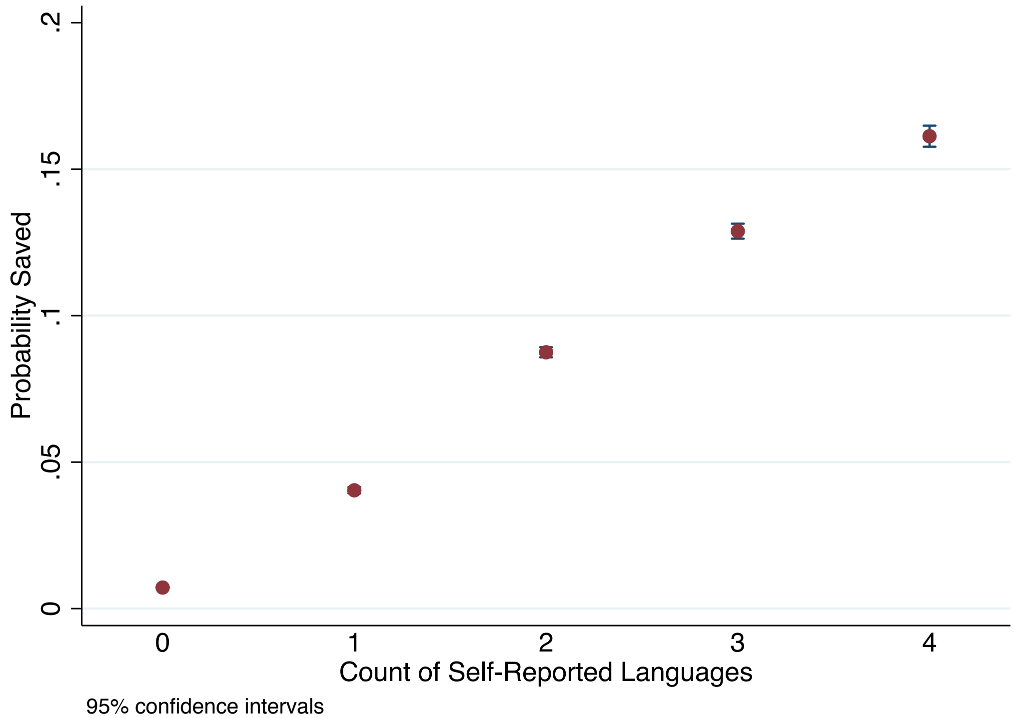
Because the plots are not particularly concave, candidates likely do not face diminishing returns or increasing costs from listing additional skills on their profiles. One might have suspected that upon seeing a large number of self-reported skills on a profile, recruiters would discount the candidate's proficiency in those skills. These plots do not show evidence that recruiters behave that way. One reason for this is because recruiters are increasingly using keyword searches for particular words and phrases on profiles.

Figure 5: Probability A Candidate is Saved Given the Number of Self-Reported Skills



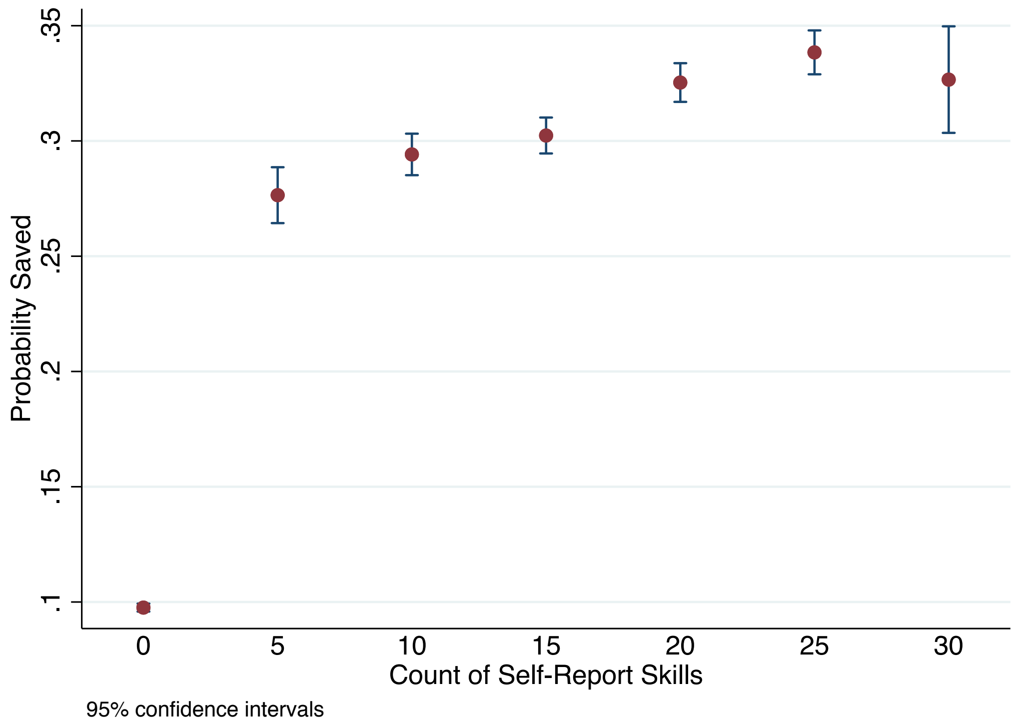
Note. Candidates are grouped by the number of self-reported skills on their profile in bins of 5. The probability that a candidate is saved is then computed for candidates within each bin. The probability is shown on the vertical axis, and the horizontal axis shows the number of self-reported skills. Along with the probability of being saved, a 95% confidence interval on the probability is displayed. All profiles are used in the above figure.

Figure 6: Probability A Candidate is Saved Given the Number of Self-Reported Languages



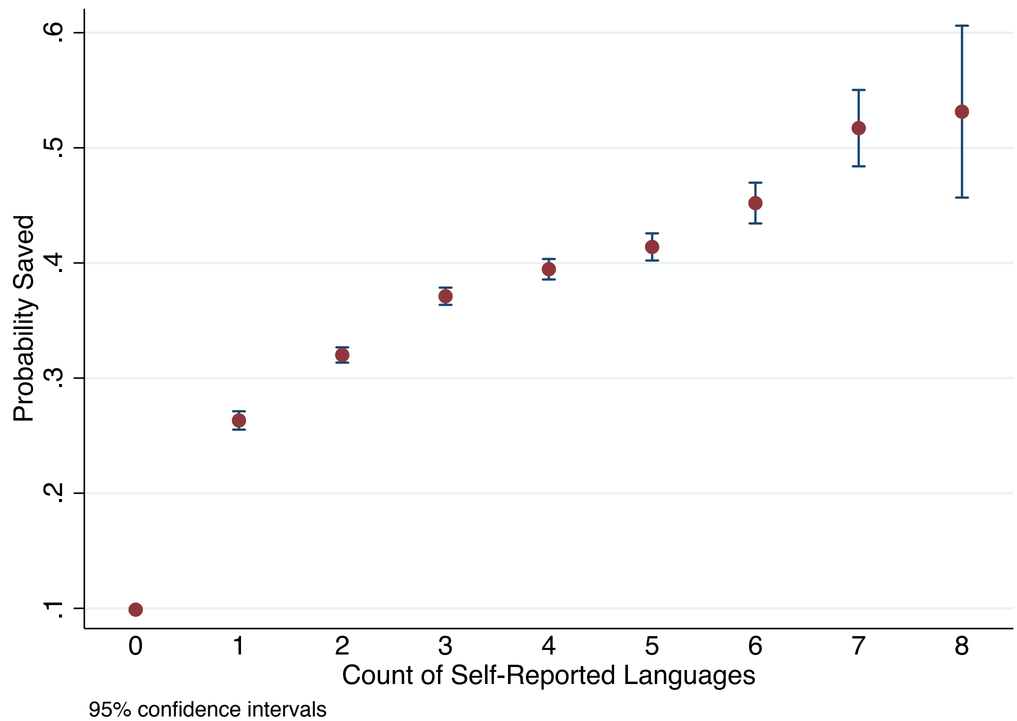
Note. Candidates are grouped by the number of self-reported languages listed within their self-reported skills. The probability that a candidate is saved is then computed for candidates for each number of languages. The probability is shown on the vertical axis, and the horizontal axis shows the number of self-reported skills. Along with the probability of being saved, a 95% confidence interval on the probability is displayed. All profiles are used in the above figure.

Figure 7: Probability A Candidate is Saved Given the Number of Self-Reported Skills



Note. Candidates are grouped by the number of self-reported skills on their profile in bins of 5. The probability that a candidate is saved is then computed for candidates within each bin. The probability is shown on the vertical axis, and the horizontal axis shows the number of self-reported skills. Along with the probability of being saved, a 95% confidence interval on the probability is displayed. Only profiles with at least one verified language are used in the above figure.

Figure 8: Probability A Candidate is Saved Given the Number of Self-Reported Languages



Note. Candidates are grouped by the number of self-reported languages listed within their self-reported skills. The probability that a candidate is saved is then computed for candidates for each number of languages. The probability is shown on the vertical axis, and the horizontal axis shows the number of self-reported skills. Along with the probability of being saved, a 95% confidence interval on the probability is displayed. Only profiles with at least one verified language are used in the above figure.

H Additional Covariates from Gender Differences in Recruiters' Response to Profile Attributes Regressions

The following shows additional coefficients from Tables 10 and 5 based on estimates of Equation 4.

Table 28: Predicted Probability a Profile is Saved Using Profile Attributes and Interactions of Self-Reporting

	Saved - All Profiles		Sa
	SR Interacted	All Interacted	SR I
BA in CS=1	0.010***	0.011***	0
	(0.000)	(0.000)	(
Female × BA in CS=1		-0.005***	
		(0.001)	
Currently Coder=1	0.010***	0.011***	0
	(0.000)	(0.000)	(
Female × Currently Coder=1		-0.005***	
		(0.000)	
Currently Coder=1 × Current Employer in Top 10 for Tech=1	0.040***	0.035***	
	(0.005)	(0.005)	(
Female × Current Employer in Top 10 for Tech=1		-0.017**	
		(0.007)	
Female × Currently Coder=1 × Current Employer in Top 10 for Tech=1		0.012	
		(0.011)	
SR Leadership=1	-0.000	-0.000	-
	(0.000)	(0.000)	(
Female × SR Leadership=1	0.001***	0.001***	-
	(0.000)	(0.000)	(
SR Social Media=1	-0.001***	-0.001***	-
	(0.000)	(0.000)	(
Female × SR Social Media=1	0.001*	0.000	-
	(0.000)	(0.000)	(
SR Project Management=1	-0.001***	-0.001***	-
	(0.000)	(0.000)	(
Female × SR Project Management=1	0.001***	0.001**	
	(0.000)	(0.000)	(
SR Public Speaking=1	-0.000	-0.000	-
	(0.000)	(0.000)	(
Female × SR Public Speaking=1	0.001***	0.001**	
	(0.000)	(0.000)	(
SR Customer Relations=1	-0.001***	-0.001***	-
	(0.000)	(0.000)	(
Female × SR Customer Relations=1	0.002***	0.001***	
	(0.000)	(0.000)	(
Controls	Yes	Yes	
N	3,744,305	3,744,305	1
Prob. Saved	0.016	0.016	
R^2	0.230	0.226	

Table 29: Predicted Probability a Profile is Saved Using Profile Attributes and Interactions of Self-Reporting, Experienced Recruiters

	Saved - All Profiles		Sa
	SR Interacted	All Interacted	SR I
BA in CS=1	0.000***	0.000**	
	(0.000)	(0.000)	(
Female × BA in CS=1		0.000	
		(0.000)	
Currently Coder=1	0.000***	0.000***	0
	(0.000)	(0.000)	(
Female × Currently Coder=1		-0.000	
		(0.000)	
Currently Coder=1 × Current Employer in Top 10 for Tech=1	-0.001	-0.001	
	(0.002)	(0.002)	(
Female × Current Employer in Top 10 for Tech=1		0.001	
		(0.003)	
Female × Currently Coder=1 × Current Employer in Top 10 for Tech=1		-0.001	
		(0.004)	
SR Leadership=1	0.000	0.000	-
	(0.000)	(0.000)	(
Female × SR Leadership=1	-0.000	-0.000	-
	(0.000)	(0.000)	(
SR Social Media=1	-0.000	-0.000	
	(0.000)	(0.000)	(
Female × SR Social Media=1	0.000	0.000	-
	(0.000)	(0.000)	(
SR Project Management=1	-0.000*	-0.000**	-
	(0.000)	(0.000)	(
Female × SR Project Management=1	-0.000	-0.000	
	(0.000)	(0.000)	(
SR Public Speaking=1	0.000**	0.000**	0
	(0.000)	(0.000)	(
Female × SR Public Speaking=1	-0.000**	-0.000**	-
	(0.000)	(0.000)	(
SR Customer Relations=1	-0.000**	-0.000**	-
	(0.000)	(0.000)	(
Female × SR Customer Relations=1	0.000*	0.000*	
	(0.000)	(0.000)	(
Controls	Yes	Yes	
N	3,744,305	3,744,305	1
Prob. Saved	0.002	0.002	
R^2	0.091	0.088	